

# **Universidad ORT Uruguay**

## **Facultad de Ingeniería**

### **WareUP: Plataforma de gestión de almacenamiento**

Entregado como requisito para la obtención del título Licenciada en Sistemas

Marcela Abi-Saab - 186587  
Romina Areco - 206874  
Leticia Presa - 198050

Tutor: Gerardo Matturro  
2023

## Declaración de autoría

Nosotras, Leticia Presa, Romina Areco y Marcela Abi-Saab declaramos que el trabajo que se presenta en esta obra es de nuestra propia mano. Podemos asegurar que:

- La obra fue producida en su totalidad mientras realizamos el proyecto de grado de la carrera de Licenciatura en Sistemas.
- Cuando hemos consultado el trabajo publicado por otros, lo hemos atribuido con claridad.
- Cuando hemos citado obras de otros, hemos indicado las fuentes. Con excepción de estas citas, la obra es enteramente nuestra.
- En la obra, hemos acusado recibo de las ayudas recibidas.
- Cuando la obra se basa en trabajo realizado conjuntamente con otros, hemos explicado claramente qué fue contribuido por otros, y qué fue contribuido por nosotras.
- Ninguna parte de este trabajo ha sido publicada previamente a su entrega, excepto donde se han realizado las aclaraciones correspondientes.



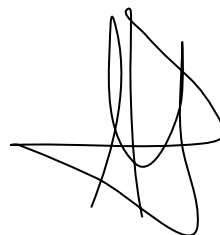
Leticia Presa

19/10/2023



Romina Areco

19/10/2023



Marcela Abi-Saab

19/10/2023

# Agradecimientos

En primer lugar, agradecer al equipo de WareUP, cliente para el que trabajamos, agradecemos la disposición que mostraron con el equipo así como la inquietud y el entusiasmo que mantuvieron a lo largo del proyecto.

Por otro lado, queremos agradecer a nuestras familias, parejas, amigos y allegados por el apoyo brindado durante estos meses de proyecto.

Finalmente también queremos agradecerle a nuestro tutor Gerardo Matturro, por mantenerse disponible y atento a nuestras consultas y dudas en el transcurso del proyecto.

Para nosotras ha sido un gran camino recorrido, donde nos nutrimos de una gran experiencia y contar con el apoyo de todas estas personas fue de suma importancia.

A todos, muchas gracias.

# Abstract

El sistema 'WareUP: Plataforma de gestión de almacenamiento', es una plataforma web desarrollada para gestionar espacios de almacenamiento que son publicados por personas/empresas y por otro lado, ayudar a aquellos usuarios que buscan arrendar espacios, por lo que el sistema sirve de intermediario.

La solución desarrollada es un *MVP* escalable que servirá como puntapié inicial para generar una solución final y lanzarla en ambientes productivos. Su diseño se basa en patrones que favorecen la flexibilidad y extensibilidad del producto.

Este sistema fue desarrollado a pedido de la empresa WareUP y fue construido en base a un conjunto de requerimientos planteados por la misma.

En cuanto a las tecnologías, se desarrolló una aplicación web en React que interactúa mediante una API REST con un backend desarrollado en NodeJs. Para el almacenamiento de datos se utilizó el gestor de base de datos MySQL.

El Frontend sirve como portal de búsqueda para aquellos usuarios que quieren arrendar un espacio, ofreciendo también la posibilidad de publicar.

El Backend funciona atendiendo las peticiones del Frontend, resolviendo problemas de lógica de negocio y el acceso a los datos.

El sistema está diseñado para correr en ambientes distribuidos, comunicándose a través del protocolo HTTPS y utilizando prácticas de seguridad para garantizar la integridad de los datos.

El ciclo de vida del desarrollo fue iterativo incremental y fue desarrollado utilizando la metodología ágil Scrum para la gestión del proyecto y BDD para el desarrollo del *frontend* y *backend*.

Durante el proceso el equipo se enfrentó a distintos obstáculos que gracias a la planificación y gestión de riesgos se superaron con éxito.

Como resultado se obtuvo una solución funcional con gran potencial de crecimiento, que cumple con los requerimientos definidos en el alcance, que resulta escalable y ha sido validada por la empresa.

## Palabras Clave

BDD, MVP, MySQL, NodeJs, React, Scrum, Typescript, WareUP, arrendamiento, logística, espacio de almacenamiento, metodologías ágiles.

# Glosario

## A

**API:** Interfaz de programación de aplicaciones que brinda servicios para ser utilizados por otro software [1].

**AWS:** Amazon Web Services es una colección de servicios de computación en la nube pública que en conjunto forman una plataforma de computación en la nube, ofrecidas a través de Internet.

**Azure:** Microsoft Azure es un servicio de computación en la nube creado por Microsoft para construir, probar, desplegar y administrar aplicaciones y servicios mediante el uso de sus centros de datos.

## B

**Backend:** Es la porción del sistema que encapsula la lógica de negocio y el acceso a datos. Tiene la responsabilidad de atender las solicitudes del front end y resolverlas según las reglas del negocio [2].

**BDD (Behaviour Driven Development):** Metodología de desarrollo de software guiada por la definición de los requerimientos, y la construcción de pruebas en un lenguaje estandarizado (generalmente se utiliza Gherkin).

**Bug:** Refiere a un error o falla en cualquier programa de computadora o sistema de hardware.

## D

**Definition of Done:** Conjunto de criterios definidos por el equipo que determinan si una funcionalidad puede ser considerada como hecha [3].

## F

**Frontend:** Es la porción del sistema que se encarga de resolver la interacción con el usuario a través de interfaces gráficas [4].

## G

**Gherkin:** Lenguaje lógico para la especificación de requerimientos. Se utiliza en la metodología BDD (Behaviour Driven Development) para unificar los requerimientos tanto para los desarrolladores o interesados, como para el software que corra los escenarios automatizados.

**Git Flow:** Es un modelo alternativo de creación de ramas en Git en el que se utilizan ramas de función y varias ramas principales [5].

## M

**Material Design:** Es una guía de diseño con pautas, componentes y herramientas que respaldan las mejores prácticas de diseño de interfaces de usuario. [6]

**MVP:** Minimum Viable Product, en español Producto Mínimo Viable, es un producto inicial con suficientes características para satisfacer al cliente [7].

## O

**One Piece Flow:** Flujo de trabajo donde se busca trabajar sobre un único producto o tarea, en lugar de avanzar en varias en paralelo.

## P

**Planning Poker:** Es una técnica basada en el consenso del equipo utilizada para estimar el esfuerzo o tamaño relativo del trabajo a realizar.

**Product Backlog:** Artefacto de *Scrum* donde se ponen todas las funcionalidades del proyecto, por lo general con un formato entendible para todos los involucrados.

**Pull request:** Funcionalidad que facilita la comunicación entre desarrolladores en Git.

## R

**Release:** Refiere a liberar una versión del software que se está desarrollando.

## S

**Scrum:** Es una metodología ágil, adaptable, flexible y efectiva para el desarrollo de software. Es un *framework* basado en un proceso incremental e iterativo con el principal objetivo de satisfacer las necesidades del cliente.

**Story point:** Se trata de unidades de medida que permiten expresar una estimación del esfuerzo total que deberá hacer el equipo para implementar íntegramente un elemento del backlog del producto o cualquier otro trabajo.

**SQA:** El SQA (Software Quality Assurance) se compone por un conjunto de actividades planificadas y sistematizadas que aseguran que el proceso de desarrollo de un software sea evaluado con respecto a su calidad, la correcta implementación de procesos, requerimientos y estándares de desarrollo.

## U

**UI:** Parte de una aplicación o sistema que permite la interacción entre el usuario y la máquina y juega un papel fundamental en la experiencia del usuario y la eficacia de la interacción con la tecnología.

# Índice

<b>1 Introducción .....</b>	<b>12</b>
1.1 Selección del proyecto.....	12
1.2 Descripción del cliente.....	13
1.3 Equipo del proyecto .....	13
1.4 Objetivos .....	14
1.4.1 Objetivos académicos .....	14
1.4.2 Objetivos del producto .....	15
1.5 Estructura del documento.....	15
<b>2 Problema y contexto .....</b>	<b>17</b>
2.1 Contexto actual.....	17
2.2 Descripción del Problema.....	18
2.3 Motivación .....	18
2.4 Entregables .....	18
<b>3 Ingeniería de requerimientos .....</b>	<b>20</b>
3.1 Introducción .....	20
3.2 Técnicas de relevamiento.....	20
3.2.1 Estudio de las soluciones existentes .....	20
3.2.2 Ingeniería inversa en la web existente de WareUP .....	24
3.2.3 Entrevista con el cliente.....	24
3.3 Especificación de requerimientos.....	25
3.3.1 Requerimientos funcionales .....	26
3.3.2 Requerimientos no funcionales .....	32
<b>4 Diseño arquitectónico .....</b>	<b>33</b>
4.1 Introducción .....	33
4.2 Descripción de la arquitectura .....	33
4.2.1 Patrones arquitectónicos .....	33
4.2.2 Patrones de diseño.....	35
4.2.3 Principios de diseño.....	38
4.3 Atributos de calidad .....	38
4.3.1 Modificabilidad .....	38
4.3.2 Usabilidad .....	39
4.3.3 Portabilidad .....	45
4.3.4 Interoperabilidad .....	48
4.3.5 Seguridad.....	48
4.3.6 Escalabilidad.....	49
4.3.7 Disponibilidad.....	49
4.4 Herramientas y tecnologías de desarrollo.....	49
4.5 Estructura de la solución .....	53
4.6 Conclusión .....	62
<b>5 Gestión del proyecto .....</b>	<b>63</b>
5.1 Introducción .....	63
5.2 Marco metodológico .....	63

5.2.1 Marco de gestión .....	63
5.2.2 Metodología Scrum.....	64
5.2.3 Metodología BDD .....	69
5.3 Herramientas de Gestión.....	74
5.4 Planificación y cronograma .....	76
5.4.1 Estimación de etapas del proyecto .....	78
5.4.2 Planificación y estimación de iteraciones.....	79
5.4.3 Resultados obtenidos de las estimaciones iniciales .....	82
<b>6 Gestión de Riesgos .....</b>	<b>96</b>
6.1 Introducción .....	96
6.2 Planificación de la gestión de los riesgos .....	96
6.2.1 Clasificación de riesgos.....	97
6.3 Cuantificación de los riesgos.....	97
6.3.1 Ocurrencia del riesgo.....	97
6.3.2 Impacto del riesgo .....	98
6.3.3 Ocurrencia en el tiempo .....	98
6.4 Ponderación de los riesgos .....	99
6.5 Riesgos identificados.....	99
6.5.1 Descripción de los riesgos.....	101
6.6 Análisis cualitativo de los riesgos.....	104
6.7 Definición de estrategias genéricas .....	105
6.8 Plan de respuesta y contingencia .....	105
6.9 Evolución de riesgos principales .....	109
<b>7 Gestión de cambios e incidentes .....</b>	<b>113</b>
7.1 Introducción .....	113
7.2 Gestión de cambios .....	114
7.3 Gestión de incidentes .....	114
7.3 Plan de gestión .....	115
7.5 Resultados del plan de gestión .....	115
7.5.1 Gestión de incidentes .....	116
7.5.2 Relación de incidentes con funcionalidades .....	122
7.5.3 Gestión de cambios .....	123
<b>8 Gestión de la calidad.....</b>	<b>127</b>
8.1 Introducción .....	127
8.2 Objetivos de calidad .....	127
8.2.1 Del producto.....	127
8.2.2 Del proyecto.....	128
8.3 Aseguramiento de la calidad (SQA).....	128
8.4 Principales actividades de SQA .....	129
8.4.1 Detalle de actividades de SQA.....	131
8.5 Definición de métricas .....	138
8.5.1 Asociados a objetivos de producto.....	138
8.5.2 Asociados a objetivos de proyecto.....	141
8.6 Evolución y análisis de métricas .....	142

8.6.1 Producto.....	142
8.6.2 Proyecto.....	147
<b>9 Gestión de la configuración .....</b>	<b>149</b>
9.1 Introducción .....	149
9.2 Herramientas utilizadas .....	149
9.3 Repositorio.....	149
9.4 Versionado.....	151
9.5 Ambientes de trabajo.....	152
9.5.1 Local.....	152
9.5.2 Testing .....	152
9.5.3 Pre-Prod.....	152
9.5.4 Producción .....	153
9.6 Despliegue de la aplicación.....	154
<b>10 Conclusiones y lecciones aprendidas .....</b>	<b>155</b>
10.1 Análisis de resultados según objetivos planteados .....	155
10.2 Conclusiones y lecciones aprendidas nivel de equipo .....	156
<b>Referencias Bibliográficas .....</b>	<b>158</b>
<b>Anexos .....</b>	<b>160</b>
Anexo 1 - Requerimientos iniciales relevados enviados al cliente .....	160
Requerimientos funcionales .....	160
Anexo 2 - Product Backlog - Versión 1 .....	163
Product Backlog - Modificaciones posteriores a la versión 1 .....	178
Anexo 3 - Prototipos iniciales .....	190
Anexo 4 - Capturas de Artefactos de Scrum.....	201
Anexo 5 - Casos de prueba funcionales .....	202
Anexo 6 - Registro de incidentes.....	228
Anexo 7 - Reuniones con tutor .....	239
Anexo 8 - Reuniones con cliente.....	240
Anexo 9 - Sprint Retrospectives .....	241
Anexo 10 - Planificación y resultado de los sprints.....	244
Anexo 11 - Endpoints REST del backend.....	249
Anexo 12 - UnitTest a nivel de backend.....	262
Anexo 13 - Sendgrid en el backend .....	265
Anexo 14 - Reportes Clockify - Gestión de horas.....	268
Anexo 15 - Integración con Google Analytics .....	270
Anexo 16 - Integración con Google Maps.....	284
Anexo 17 - Resultados de encuesta de satisfacción del usuario final .....	289
Anexo 18 - Resultados de encuesta de grado de satisfacción del cliente WareUP .....	292

# 1 Introducción

La presente obra tiene como objetivo describir el proceso de construcción y desarrollo del proyecto final de carrera denominado “WareUP - Plataforma de gestión de almacenamiento” con el fin obtener el título de Licenciado en Sistemas de la Universidad ORT Uruguay.

WareUP es el primer marketplace de logística en Uruguay, provee soluciones de logística a empresas de pequeño y mediano porte siendo el nexo entre empresas con capacidad ociosa para ofrecer y aquellas con necesidades de capacidad.

En la actualidad el cliente cuenta con una plataforma web desarrollada en WordPress la cual cumple con las funcionalidades básicas necesarias del negocio, pero presenta la limitante de verse acotado a lo que ofrece este sistema en particular. Esto implica que la viabilidad del desarrollo de una nueva funcionalidad dependa de terceros, ajenos a la empresa, lo que no es deseable.

Es por esto que surge este proyecto, con el fin de crear una plataforma web versátil y personalizada, que permita una mayor extensibilidad a posibles requerimientos y funcionalidades a futuro.

Para el desarrollo y gestión del proyecto se optó por metodologías ágiles, puntualmente la aplicación de Scrum como marco principal. Se llevaron a cabo diferentes prácticas relacionadas al desarrollo de software las cuales se detallan en este documento.

## 1.1 Selección del proyecto

El proyecto lo conocimos a partir de la feria de proyectos presentados por la Universidad ORT Uruguay, e inmediatamente supimos que sería el proyecto ideal, porque si bien es un proyecto definido por una empresa, implicaba investigación y libertad en la toma de decisiones a la hora de construirlo.

Por una parte, al tratarse de un rubro desconocido, implicaba la necesidad de informarnos al respecto, buscar información y aprender sobre un negocio nuevo y en auge.

Además, la libertad tanto en diseño como en tecnología nos daba el espacio para investigar diferentes soluciones, evaluar cada una, identificar ventajas y desventajas y tomar decisiones al respecto.

Esa necesidad de conocimiento y libertad en la toma de decisiones nos resultó desafiante y atractiva, ya que nos brinda de un gran pilar y una base sólida para enfrentarnos a varios proyectos tecnológicos del mercado.

## 1.2 Descripción del cliente

La empresa WareUP es una empresa que provee soluciones de logística a empresas de pequeño y mediano porte, ayudándoles a ingresar en un sector donde es muy difícil acceder.

A través de WareUP se cubren dos tipos de necesidades, por un lado a aquellas empresas con necesidades de capacidad, a quienes se les ofrece una amplia oferta de depósitos con la flexibilidad necesaria para adaptarse ágilmente a sus necesidades. Por otro lado, brinda servicios a las empresas con capacidad ociosa, a quienes ayudan a capitalizar el espacio ocioso, transformándolo en una nueva fuente de ingresos.

Lo que busca WareUP es ser el nexo entre ambos segmentos, brindándoles soluciones de almacenamiento.

En la actualidad la empresa tiene únicamente un socio y trabaja con 3 colaboradores/consultores.

## 1.3 Equipo del proyecto

El equipo está conformado por Leticia, Romina y Marcela. Se formó a raíz de haber compartido materias en común y encontrar buena sinergia personal y sobre todo complementación de las características de cada individuo.

Entendemos que diferentes cualidades y diversidad puede ser beneficioso a la hora de llevar a cabo el proyecto final, ya que nos complementamos entre nosotros.

Yendo a cada individuo en particular y los diferentes roles, presentamos al equipo junto con sus principales cualidades y roles que tomará dentro del marco de trabajo *SCRUM*.

Marcela Abi-Saab: *Product Owner* y *Scrum Master*, debido a la experiencia obtenida en gestión de proyectos, será el principal nexo entre el cliente y el equipo pero también será la encargada de gestionar el equipo de manera interna.

Leticia Presa: *Developer*, debido a la experiencia como desarrolladora.

Romina Areco: *Technical Leader* del proyecto, en virtud de su larga trayectoria como desarrolladora de software y conocimiento de tecnologías.

Cabe destacar que debido a lo reducido que es el equipo para aplicar de la mejor forma Scrum, es que las tres integrantes si bien tienen su rol principal podrán desempeñar tareas correspondientes a otros roles.

## 1.4 Objetivos

El objetivo del proyecto es la construcción de una plataforma funcional, versátil y escalable que permita a empresas de pequeño y mediano porte ofrecer o alquilar almacenamiento, aplicando en el proceso buenas prácticas de la ingeniería de software.

### 1.4.1 Objetivos académicos

Id	Objetivo	Descripción
OA1	Aplicar conocimientos adquiridos y desarrollados durante la carrera	Se busca aplicar metodologías y conceptos aprendidos a lo largo de la carrera en un proyecto real. Tanto aspectos de ingeniería y diseño de software, como la gestión de proyectos.
OA2	Adquirir experiencia en desarrollo y gestión de proyectos.	Se tiene como finalidad adquirir experiencia en gestión y desarrollo de proyectos para el crecimiento profesional de las integrantes del equipo.

Tabla 1: Objetivos académicos

## 1.4.2 Objetivos del producto

Id	Objetivo	Descripción
OP1	Entregar un mínimo producto viable	Desarrollar un <i>MVP</i> que aporte valor a la empresa WareUP.
OP2	Desarrollar software de calidad y extensible	Implementar teniendo en cuenta aspectos de calidad de software que aporten valor a la plataforma y que pueda ser extensible a futuras funcionalidades.
OP3	Entregar documento adicional al cliente	Entregar un documento que detalle la arquitectura de la plataforma junto con otros aspectos importantes a considerar de la misma.

Tabla 2: Objetivos del producto

## 1.5 Estructura del documento

A continuación, se describe brevemente lo que se presentará en cada sección numerada del cuerpo de la obra.

En el **capítulo 2:** Problema y contexto, se describen los problemas a resolver, el contexto de la empresa y las motivaciones que llevaron a desarrollar la solución.

En el **capítulo 3:** Ingeniería de Requerimientos, se incluye el proceso de relevar, identificar y documentar los requerimientos, junto con el listado de los requerimientos funcionales y no funcionales.

En el **capítulo 4:** Arquitectura y Diseño de la Solución, se detallan las decisiones tomadas a nivel de diseño para construir el sistema, desde la arquitectura a las tecnologías utilizadas.

En el **capítulo 5:** Metodología de Gestión, se explican las decisiones tomadas a nivel de proyecto para atacar el problema, incluyendo el ciclo de vida, las metodologías utilizadas, la planificación y el cronograma.

En el **capítulo 6:** Gestión de Riesgos, se detallan los riesgos identificados así como el plan que se implementó para mitigarlos y enfrentarlos.

En el **capítulo 7:** Gestión de cambios e incidentes, se detalla el plan que se llevó a cabo para gestionar los cambios e incidentes que surgieron en el proyecto. Se detallan los mismos, junto con métricas de análisis.

En el **capítulo 8:** Gestión de la Calidad, se reportan los objetivos, las actividades de aseguramiento de calidad planificadas y realizadas a lo largo del proyecto.

En el **capítulo 9:** Gestión de la configuración, se detallan las actividades planificadas y realizadas de gestión de la configuración del software, incluyendo líneas base, versionado y control de repositorios.

En el **capítulo 10:** Conclusiones y lecciones aprendidas, se mencionan todas las conclusiones y lecciones aprendidas por el equipo a lo largo del proyecto y si se logró cumplir con los objetivos pautados inicialmente.

## 2 Problema y contexto

### 2.1 Contexto actual

En el contexto actual de la empresa se cuenta con algunos clientes fijos, se trabaja en aumentar la base de datos existente de m<sup>3</sup> en oferta y el foco hoy está puesto en la búsqueda de financiación para hacer crecer la empresa en términos comerciales, continuar validando la propuesta y comenzar a validar otros aspectos de la propuesta de valor. El mercado se encuentra en continua expansión, con cierta opacidad, al que no todas las empresas acceden plenamente y con grandes cambios de paradigma impulsados por los cambios en los hábitos de consumo del cliente final.

El sistema actual del cliente es una solución a la gestión de logística que busca mejorar la eficiencia de la cadena de suministro. Dicha solución es una plataforma web desarrollada a través de *WordPress*, la cual es un *CMS* (Sistema de Gestión de Contenidos), una herramienta que ayuda a las empresas a crear, publicar y gestionar el contenido de sus páginas web.

La plataforma puede ser utilizada tanto por empresas que necesitan de espacio, como por empresas que tienen espacio ocioso para ofrecer. Para esto las empresas/personas deben registrarse mediante un usuario con correo electrónico y una contraseña.

A través de la plataforma las empresas pueden registrar espacios ociosos habilitados para su alquiler, detallando varios aspectos de importancia para el negocio como lo son: la ubicación, el perímetro, el tipo de zona (temperatura controlada, zona descubierta, zona techada), tipo de piso, entre otras.

Por otro lado, aquellas personas o empresas que tengan la necesidad de arrendar espacio podrán acceder a una búsqueda mediante fotos y filtros (correspondientes a los aspectos mencionados anteriormente).

### 2.2 Descripción del Problema

La problemática que se presenta actualmente en la plataforma existente es que al estar, la solución, desarrollada mediante *WordPress* se ve limitada a utilizar

únicamente los servicios y funciones que la misma provee. La logística es un negocio que se encuentra en auge por lo que surgen diferentes necesidades de forma constante y que deben poder ser resueltas en la plataforma, idealmente, sin la dependencia de WordPress.

Por otro lado, a partir de conversaciones con el cliente, nos encontramos con que el sector se está digitalizando de gran manera, pero que aun así es un rubro bastante conservador donde la confianza con el interlocutor es uno de los valores más importantes por lo que, WareUP, el desafío al que se enfrenta principalmente es lograr recrear ese valor a través de una plataforma que muestre transparencia y destaque su propuesta de valor.

## 2.3 Motivación

La principal motivación del cliente para determinar los objetivos es el deseo de resolver el problema descrito en la sección anterior y lograr una plataforma lo más amigable posible con el usuario que motive su uso.

Por parte del equipo la motivación principal es poder aportar valor al cliente y ser de real ayuda en el desarrollo de su sistema.

## 2.4 Entregables

El análisis de requerimientos sirvió como punto de partida para desglosar los entregables al cliente que se mencionan a continuación:

- Entrega del código fuente de la plataforma web.
- Documentación correspondiente con información valiosa sobre el proyecto: diagramas de la arquitectura, análisis de un posible servicio de hosting de la plataforma y especificaciones particulares que ameriten ser detalladas para el conocimiento del cliente.
- Instructivo para la integración de la plataforma con Google Maps. Ver [Anexo 16 - Guía integración con Google Maps](#) para visualizar la guía proporcionada al cliente de cómo utilizarla.

- Instructivo para la integración de la plataforma con Google Analytics. Ver [Anexo 15 - Integración con Google Analytics](#) para visualizar la guía proporcionada al cliente de cómo utilizarla.

Es de importancia destacar que lo relacionado a la implementación del hosting de la plataforma queda del lado del cliente, es por eso que como se menciona, en relación a este tema, se le entregará un análisis de comparación entre los dos servicios de hosting AWS y Azure, para que el cliente tenga una guía y un panorama de qué servicio podría servirle más.

## 3 Ingeniería de requerimientos

### 3.1 Introducción

A través de esta sección se presenta el proceso de Ingeniería de Requerimientos. Dentro de este proceso se utilizaron diferentes técnicas de relevamiento como ingeniería inversa de la web existente, entrevista al cliente e investigación de soluciones similares existentes.

El cliente WareUP ya contaba con una solución web para su servicio es por esto que gran foco del proceso estuvo en identificar cuales funcionalidades de su sistema actual podrían aplicarse a la nueva solución, para luego lograr aplicar reingeniería de software sobre la información recolectada.

Por otro lado, al tratarse de un nuevo desarrollo, el cliente brindó al equipo la posibilidad de plantear nuevas funcionalidades que parecieran interesantes o que pudiesen aportar valor. Para esto se realizó una investigación de soluciones existentes así como también al equipo le surgieron ideas propias que fueron puestas en consideración.

### 3.2 Técnicas de relevamiento

#### 3.2.1 Estudio de las soluciones existentes

Se realizó un estudio de dos soluciones existentes que presentan ciertas similitudes con la plataforma a desarrollar, como son la plataforma Airbnb [8], que si bien no forma parte del negocio de logística, funciona como intermediario entre dos interesados, al igual que nuestro proyecto, y por otro lado se analizó la plataforma Warehouse Exchange [9], una plataforma extranjera que ofrece el mismo servicio que WareUP.

Este análisis permitió tener una mirada más amplia de posibles filtros y diagramación de pantallas entre otros aspectos. Inspirarnos en soluciones que consideramos pioneras en los modelos de negocio de mercados comunitarios nos genera confianza de que estamos basándonos en estándares altos y que a futuro la solución desarrollada pueda escalar a una plataforma simil Airbnb, por ejemplo.

## Airbnb:

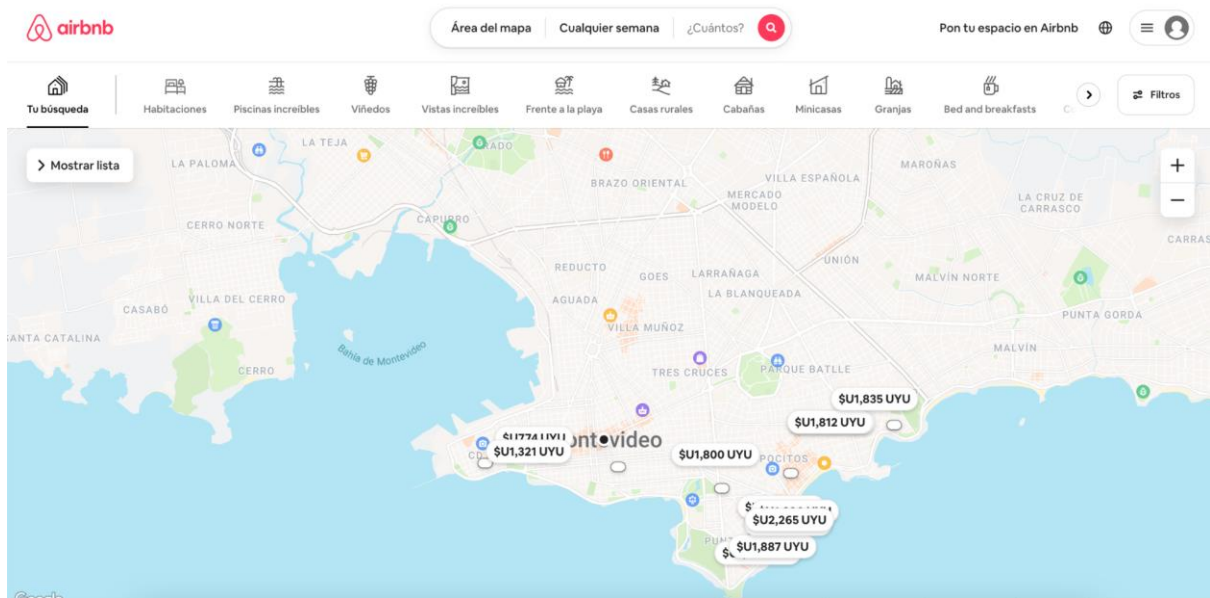


Ilustración 1: Airbnb - Interfaz gráfica 1

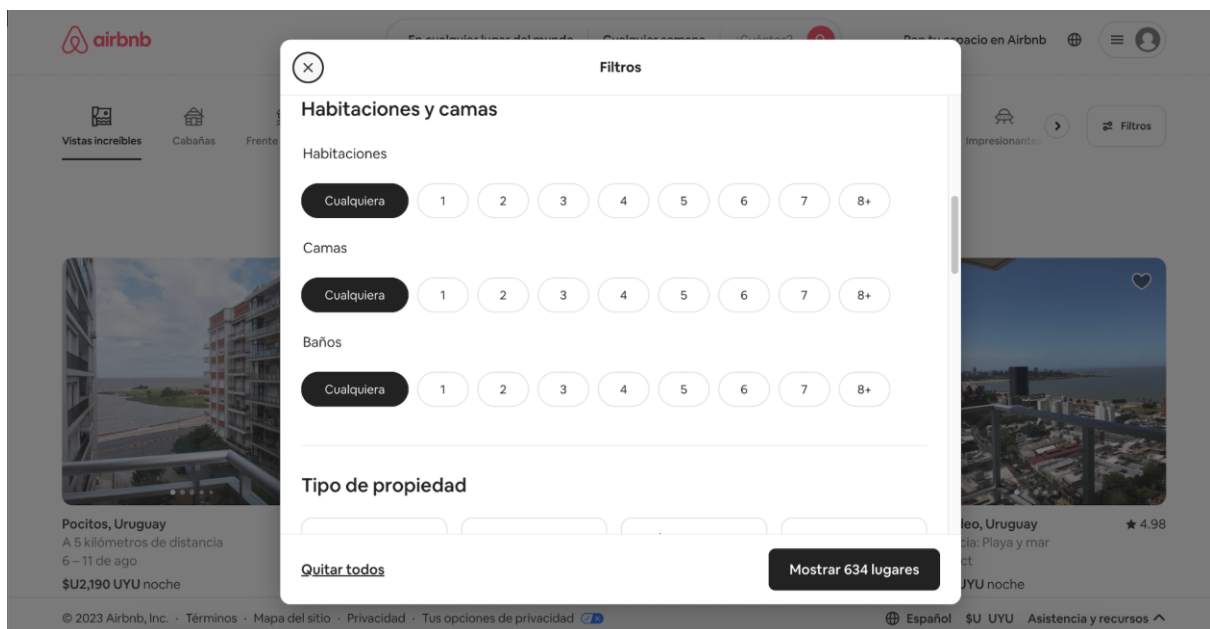


Ilustración 2: Airbnb - Interfaz gráfica 2

Airbnb es una plataforma que sirve para publicar, dar publicidad y reservar alojamiento en diferentes países del mundo a través de internet. Quien tiene alojamiento para ofrecer, lo publica en la web y por otro lado están los usuarios que necesitan de alojamiento.

De esta plataforma recabamos los siguientes puntos:

- **Utilización del mapa:** consideramos que resulta amigable para el usuario la utilización de un mapa para ubicar los lugares geográficamente.
- **Filtros ubicados arriba del mapa:** Destacamos el uso de los filtros que se muestran arriba del mapa ofreciendo rápidamente la filtración directa al clickear sobre cada uno de ellos. También ofrece otra ventana en formato “pop up” para setearlos manualmente.

## Warehouse Exchange:

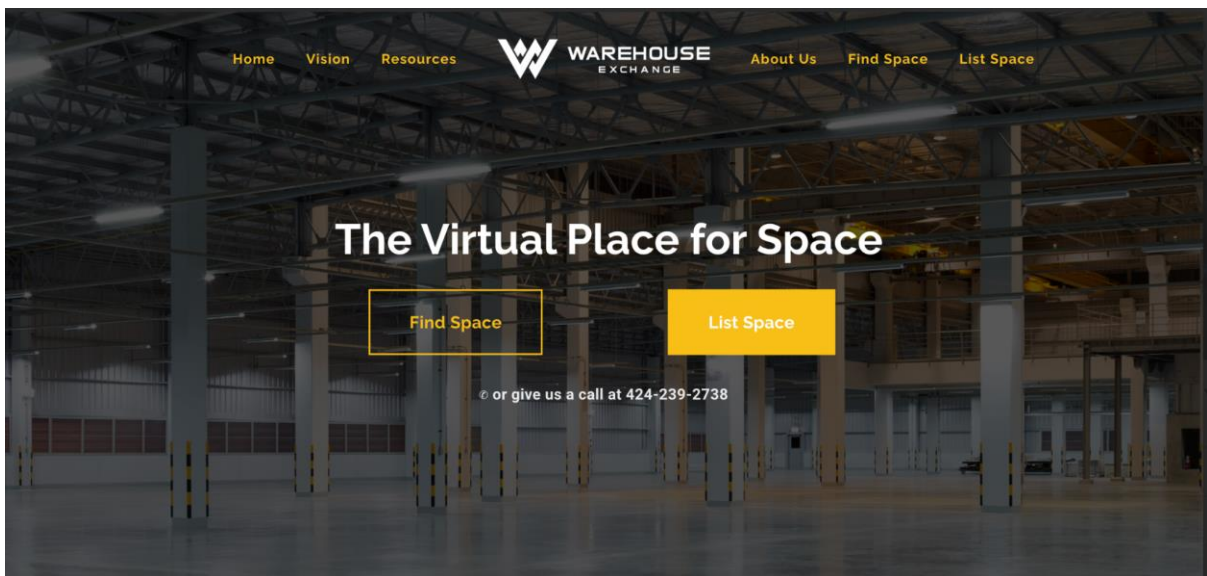


Ilustración 3: Warehouse Exchange - Interfaz gráfica 1

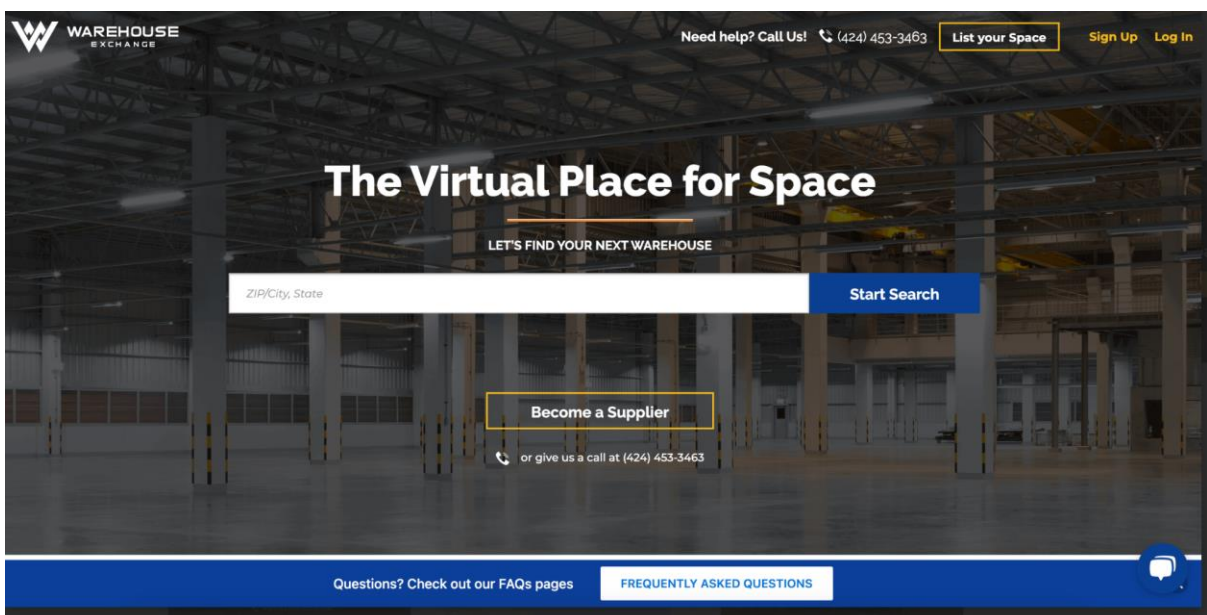


Ilustración 4: Warehouse Exchange - Interfaz gráfica 2

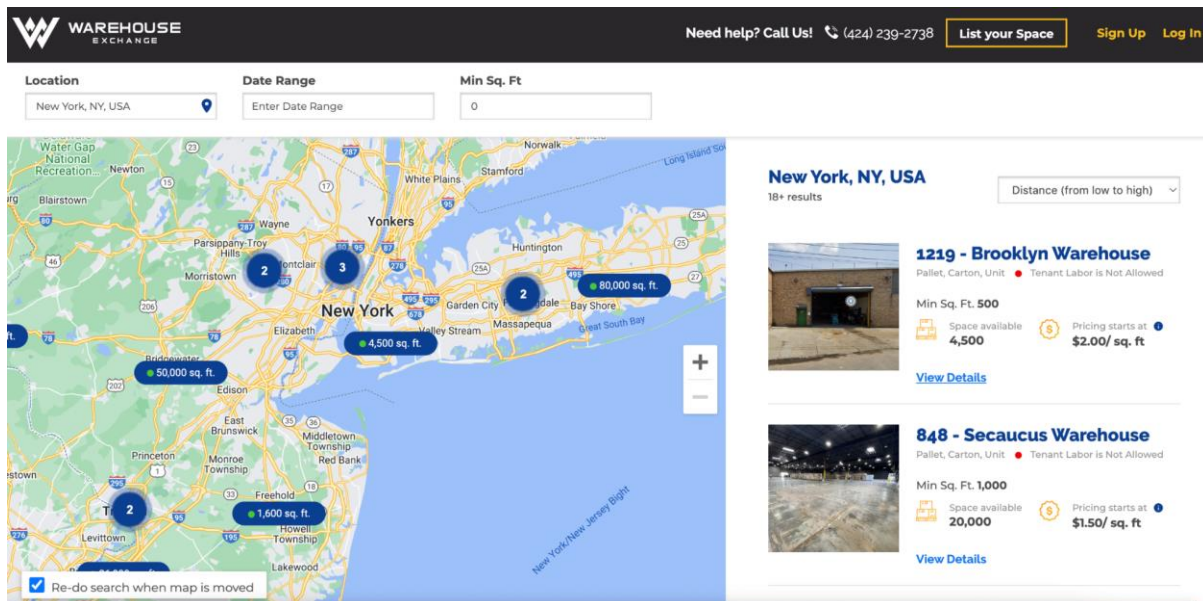


Ilustración 5: Warehouse Exchange - Interfaz gráfica 3

Warehouse Exchange, es una plataforma web que funciona como motor de búsqueda para el mercado de espacios/almacenes. Los proveedores utilizan la plataforma para publicar sus almacenes libres mientras que las empresas que buscan un espacio logístico pueden encontrarlo en la web.

De esta plataforma recabamos los siguientes puntos:

- **Pantalla principal:** La pantalla principal destaca dos botones, Encontrar Espacio (*Find Space*) y Publicar Espacio (*List Space*), estos dos botones presentados en la pantalla central consideramos son muy intuitivos para el usuario y rápidamente el usuario puede seleccionar ser proveedor de almacenamiento o arrendar un espacio.
- **Pantalla con motor de búsqueda:** Nos pareció de utilidad la barra de búsqueda que presentan, pudiendo buscar por Ciudad o Estado. Podría aportar valor a nuestro proyecto presentarlo de la misma manera, pudiendo buscar por ejemplo por Barrio.
- **Utilización del mapa:** Al igual que Airbnb esta plataforma utiliza un mapa para ubicar los espacios geográficamente, donde se destacan también los metros cuadrados. También, en la misma pantalla sobre la derecha, se puede ver el listado de espacios.

### 3.2.2 Ingeniería inversa en la web existente de WareUP

Se realizó el proceso de ingeniería inversa sobre la web que tiene actualmente nuestro cliente [18], este proceso es llevado a cabo con el objetivo de obtener información a partir de un producto existente.

Esta técnica fue la de mayor utilidad debido a que la página cuenta con una base de lo que se esperaría en el nuevo desarrollo. Así como hay funcionalidades que las anotamos a ser implementadas, encontramos otras funcionalidades o aspectos de la interfaz web que consideramos deben ser eliminadas o modificadas. Toda esta información relevada en una primera instancia fue documentada para luego ser discutida con el cliente y posteriormente modificada. Se puede ver esta información en el [Anexo 1 - Requerimientos iniciales relevados enviados al cliente](#).

### 3.2.3 Entrevista con el cliente

Como último paso para el relevamiento de requerimientos se llevó a cabo una reunión con el cliente donde asistió el director de la empresa, el encargado del área de tecnología y la encargada del área comercial. La asistencia de representantes de todas las áreas fue crucial dado que se presentaron intercambios de ideas interesantes, aportando distintos puntos de vista sobre la necesidad de un posible requerimiento.

En este encuentro se analizó el documento que fue comentado en el punto anterior, se leyó cada requerimiento en conjunto, se discutió y, en caso de ser necesario, se hicieron las modificaciones pertinentes.

Consideramos que esta técnica de relevamiento, es decir la entrevista con los interesados, fue de la que más nos nutrimos y más información pudimos extraer. Si bien ya contábamos con una base debido a la ingeniería inversa realizada sobre la plataforma existente, en la entrevista es donde se dieron conversaciones interesantes, pudimos intercambiar sobre tecnologías, necesidades del negocio, entre otros.

### 3.3 Especificación de requerimientos

El registro de requerimientos se llevó a cabo entre el 24 de abril de 2023 hasta el 22 de mayo de 2023. Los requerimientos fueron inicialmente registrados en un documento con el siguiente formato:

**REF id - nombre del requerimiento - descripción del requerimiento**

(Siendo REF id un identificador numérico)

Con información básica y necesaria para luego poder profundizar más en cada uno. El impacto y esfuerzo para completar cada uno fue definido en las *Sprint Plannings*.

En una segunda instancia procedimos a la utilización del formato *user story*. El utilizar *Scrum* como metodología de trabajo surgió como la opción más natural. Las user stories referentes al desarrollo tuvieron la siguiente sintaxis:

Como <TIPO DE USUARIO> quiero <OBJETIVO> para tener <BENEFICIO>

Para calificar los requerimientos se usó la siguiente escala de valores:

- **Alta:** Estos son requerimientos que el cliente quiere que sin falta se encuentren en la plataforma.
- **Media:** Requerimientos que el cliente tiene gran interés de que se desarrollen, pero no son de máxima prioridad.
- **Baja:** Requerimientos que al cliente le resultan deseables de tener pero están últimos en prioridad.

Estas calificaciones fueron brindadas por el *Product Owner* del equipo luego de tener definido el alcance del proyecto.

Por otro lado, en cuanto a los actores del sistema, se identifican dos posibles roles de usuarios:

- **Usuario cliente:** usuarios y/o empresas que utilizan los servicios de WareUP para publicar sus espacios de almacenamiento disponibles o pueden arrendar espacios de almacenamiento en caso de lo que necesiten. Cada usuario cliente es responsable de mantener la información de sus publicaciones al día.
- **Usuario backoffice:** usuarios que tienen acceso a las mismas funcionalidades que los usuarios cliente pero también cuentan con el poder de gestionar todas las entidades del sistema. Son también conocidos como **administradores del sistema**.

Según lo pautado por el estándar **729** de la IEEE [10], separamos los requerimientos de nuestro proyecto entre funcionales y no funcionales.

A continuación se presentan los requerimientos funcionales. La lista completa de User Stories asociadas a los requerimientos está disponible en el [Anexo 2 - Product Backlog - Versión 1](#)

### 3.3.1 Requerimientos funcionales

#### 3.3.1.1 Requerimientos generales

##### **REF1 - Inicio de sesión de usuario**

- Tanto usuarios clientes como usuarios backoffice deberán de poder loguearse en la web de WareUP utilizando mail y contraseña.
- Prioridad - Alta

##### **REF2 - Unicidad de identidad de usuario**

- Cada usuario será único dentro del sistema
- Prioridad - Alta

##### **REF3 - Recuperación de contraseña**

- El sistema deberá tener la habilidad de cambiar o recuperar la contraseña de un usuario (se debe hacer mediante envío de un mensaje o un email al usuario).
- Prioridad - Alta

#### **REF4 - Autogestión de perfil**

- El usuario deberá poder gestionar su perfil pudiendo cambiar sus datos personales.
- Prioridad - Media

#### **REF5 - Roles del usuario**

- El sistema maneja **2** roles de usuarios distintos. Un usuario podrá ser Backoffice o Usuario cliente. Siendo el primero capaz de gestionar todos los usuarios del sistema y gestionar los espacios que se publiquen mientras que el usuario cliente podrá publicar o arrendar espacios disponibles.
- Prioridad - Media

#### **REF6 – Niveles de usuario Backoffice**

- El sistema deberá permitir que distintos usuarios de Backoffice tengan distintos niveles de privilegios dentro de la Web. Siendo estas de tener acceso a todas las funcionalidades de la misma, o como tener acceso a un conjunto básico de funcionalidades.
- Prioridad - Media

#### **REF7 - Logout**

- El usuario podrá cerrar la sesión.
- Prioridad - Alta

#### **REF8 - Registro de usuarios**

- El sistema deberá permitir el registro de usuarios nuevos al sistema. Donde como primera versión solo será necesario los siguientes campos: email, nombre y contraseña.
- Prioridad - Alta

#### 3.3.1.2 Requerimientos para el usuario que busca espacio

#### **REF9 - Buscador de espacios publicados**

- El usuario podrá acceder a un listado de todos los espacios disponibles que estén publicados en la plataforma.
- Prioridad - Alta

### **REF10 - Filtros para mejorar la búsqueda de espacios**

- El usuario podrá disponer de distintos filtros para realizar la búsqueda de espacios de acuerdo a sus necesidades.
- Prioridad - Alta

### **REF11 - Ordenar resultados de búsqueda**

- El usuario podrá ordenar los resultados de la búsqueda de espacios de acuerdo al criterio que le parezca mejor.  
Ejemplo: Menor precio/Mayor precio, Más cercano geográficamente/Más alejado geográficamente, Mayor completitud/Mejor completitud de la información.
- Prioridad - Baja

### **REF12 - Mapa con las publicaciones en base a la cercanía del usuario**

- El usuario puede acceder a un mapa donde se muestran todos los espacios de almacenamiento que tenga en un rango aproximado indicando si están disponibles o no.
- Este requerimiento al equipo le implica análisis de tecnologías y otras cuestiones, se plantea como un requerimiento que puede ser bueno de implementar pero no es prioritario.
- Prioridad - Baja

### **REF13 - Significado de los filtros**

- Un usuario podrá pararse sobre un popup y visualizar una breve definición de lo que significa el filtro de búsqueda.
- Prioridad - Baja

#### **REF14 - Visualización rápida de un espacio de almacenamiento**

- El usuario puede visualizar de forma sencilla información básica de un espacio publicado donde se muestra una imagen, descripción, título y precio.
- Prioridad - Media

#### **REF15 - Visualización completa de una publicación de espacio de almacenamiento**

- El usuario puede seleccionar una publicación del buscador y acceder a la misma para obtener el detalle del espacio de almacenamiento completo.
- Prioridad - Alta

#### **REF16 - Comprobar disponibilidad de un espacio de almacenamiento**

- El usuario podrá seleccionar un periodo de tiempo y metros cuadrados necesarios para comprobar si el espacio de almacenamiento se encuentra disponible para ser arrendado o no.
- Prioridad - Alta

#### **REF17 - Generar solicitud de arrendamiento de un espacio de almacenamiento**

- El usuario podrá generar una solicitud de arrendamiento de espacio de almacenamiento si éste se encuentra disponible en la fecha y metros cuadrados indicados. Al momento de confirmar la solicitud para arrendar un espacio, se solicitará la información de la empresa que desea confirmar esta operación.
- Prioridad - Alta

#### **REF18 - Compartir publicación de espacio de almacenamiento**

- Un usuario podrá compartir en sus redes sociales una publicación de un espacio de almacenamiento que se encuentre en la plataforma.
- Prioridad - Baja

### 3.3.1.3 Requerimientos para el usuario que ofrece espacio

#### **REF19 - Registrar solicitud de espacio de almacenamiento**

- Un usuario cliente puede generar una solicitud para registrar el espacio disponible con el que cuenta acompañado de la información de la empresa. Luego de procesar esta solicitud, se registrará el espacio de almacenamiento en la plataforma.
- Prioridad - Alta

#### **REF20 - Cambiar estado de solicitud de registro de espacio de almacenamiento**

- Un usuario cliente o backoffice puede cambiar el estado de la solicitud de registro de un espacio de almacenamiento.
- Prioridad - Alta

#### **REF21 - Alta de espacio de almacenamiento**

- Un usuario backoffice puede dar de alta una publicación de espacio de almacenamiento que esté pendiente de aprobación.
- Prioridad - Alta

#### **REF22 - Baja de espacio de almacenamiento**

- Un usuario cliente o backoffice puede dar de baja la publicación de un espacio de almacenamiento.
- Prioridad - Alta

#### **REF23 - Modificar información del espacio de almacenamiento**

- Un usuario cliente o backoffice puede modificar la información de un espacio de almacenamiento.
- Prioridad - Alta

#### **REF24 - Validar información modificada de un espacio de almacenamiento existente**

- Un usuario backoffice puede aprobar o rechazar las modificaciones que se intenten hacer sobre un espacio de almacenamiento ya existente en el sistema.
- Algunas modificaciones serán de forma automática, aunque para el usuario toda modificación que se haga debe ser validada por un usuario backoffice. La definición de qué características pueden ser modificadas de forma automática será provista por parte del cliente.
- Un usuario backoffice tiene como máximo 72 horas hábiles para aprobar o rechazar la solicitud del usuario en su publicación, en caso de que este tiempo se termine, la publicación se aprobará correctamente.
- Prioridad - Baja

#### **REF25 - Crear menú principal**

- Un usuario cliente o backoffice puede visualizar una homepage con la información general del negocio de la empresa.
- Prioridad - Alta

### 3.3.2 Requerimientos no funcionales

A continuación, se presentan los requerimientos no funcionales que fueron relevados y validados con el cliente:

<b>Id</b>	<b>Descripción</b>	<b>Prioridad</b>
RNF1	Se debe desarrollar el backend de la plataforma utilizando NodeJS	Alta
RNF2	Se debe desarrollar el frontend de la plataforma utilizando React	Alta
RNF3	Se debe utilizar MySQL como motor de Base de Datos	Alta
RNF4	La solución debe ser sencilla e intuitiva de utilizar por los distintos autores	Alta
RNF5	La solución debe permitir solo a usuarios registrados y que hayan iniciado sesión, acceder al contenido de la Web a excepción del buscador y/o publicaciones	Alta
RNF6	Toda entidad que se registra en el sistema, ya sean usuarios, espacios de almacenamiento o arrendamientos, deberán ser datos protegidos en una base de datos.	Alta
RNF7	Adaptabilidad de la Web para ser utilizada en un navegador Web común y en un navegador de dispositivo móvil.	Media
RNF8	Ninguna de las funcionalidades disponibles en el sistema debe tardar más de 1 minuto en ser ejecutada.	Media
RNF9	Cumplimiento de estándares de calidad de código y buena programación para que la solución sea mantenida por el cliente en el futuro	Alta
RNF10	Disponibilidad 24/7 del sistema para que los usuarios puedan operar sobre el mismo sin importar el horario con la salvedad de definir eventuales períodos de baja por mantenimiento.	Alta
RNF11	Que la plataforma sea desarrollada de forma que pueda ser escalable en cuanto a nivel de funcionalidades e infraestructura.	Alta

Tabla 3: Requerimientos no funcionales

## 4 Diseño arquitectónico

### 4.1 Introducción

En este capítulo se presenta la arquitectura que se implementó para la solución del cliente. Luego de analizar la plataforma que WareUP tiene desarrollada en WordPress y de relevar los requerimientos funcionales y no funcionales se diseñó una arquitectura acorde a las necesidades del cliente.

### 4.2 Descripción de la arquitectura

La solución se compone de un *frontend*, desarrollado en React, un *backend* implementado en una API Rest mediante NodeJs y una base de datos desarrollada utilizando MySQL.

La solución fue implementada de tal forma de poder hacer un despliegue distribuido, teniendo en foco la escalabilidad de la plataforma, el cual es un requerimiento no funcional con prioridad alta (RNF11).



Ilustración 6: Diagrama arquitectura propuesta

#### 4.2.1 Patrones arquitectónicos

##### 4.2.1.1 Arquitectura en capas

Se utilizó el patrón de arquitectura en capas, un patrón muy conocido y aplicado en el rubro, por lo que está ampliamente validado. El patrón consiste en separar la plataforma en diferentes capas, donde cada una tiene una responsabilidad y razón de existir.

Yendo a nuestro proyecto puntual, definimos las capas *Controller*, *Service* y *Repository*. Cada capa tiene entonces una función que cumplir y no debe realizar

nada más que para lo que fue creada. Por ejemplo, la capa negocios (*Service*) no necesita preocuparse por cómo formatear los datos, ni si vienen la información requerida, solo debe ejecutar el pedido siguiendo la lógica de negocio, apoyándose de la capa *Repository*, quien le brinda la información existente. Una vez aplicada la lógica de negocio sobre la información, solo debe retornarla a la capa *Controller*.

Cada capa se comunica únicamente con la capa que se encuentra debajo. A su vez, se cuenta con una capa transversal que contiene funcionalidades comunes a todas las capas.

Se presenta a continuación un diagrama donde se plasman las diferentes capas e interacciones.

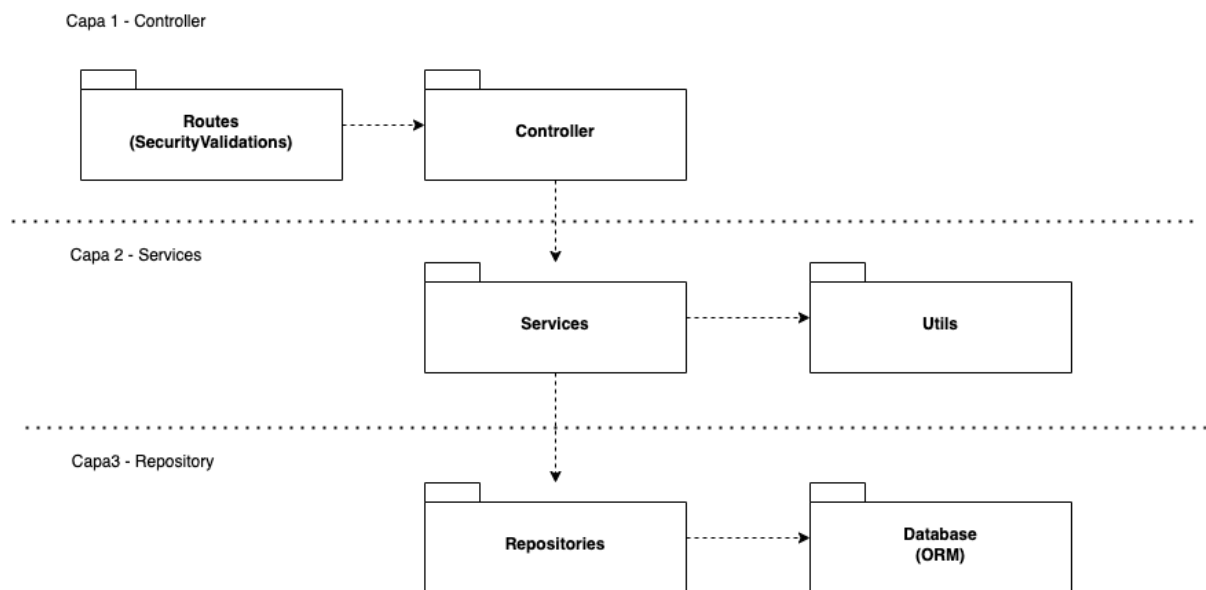


Ilustración 7: Arquitectura en capas implementada

#### 4.2.1.2. Arquitectura cliente-servidor

Este patrón se utilizó para separar la implementación del *frontend* de lo que es el *backend*, permitiendo fácilmente intercambiar el *frontend* o anexar nuevas implementaciones escalando la solución.

Hoy en día se cuenta únicamente con una aplicación web desarrollada con React, pero la arquitectura propuesta permite a futuro poder incorporar aplicaciones móviles nativas o cualquier otro sistema, reutilizando la lógica de negocio.

Se presenta a continuación un diagrama con un ejemplo de cómo podría escalar la aplicación, incluyendo una aplicación Mobile.

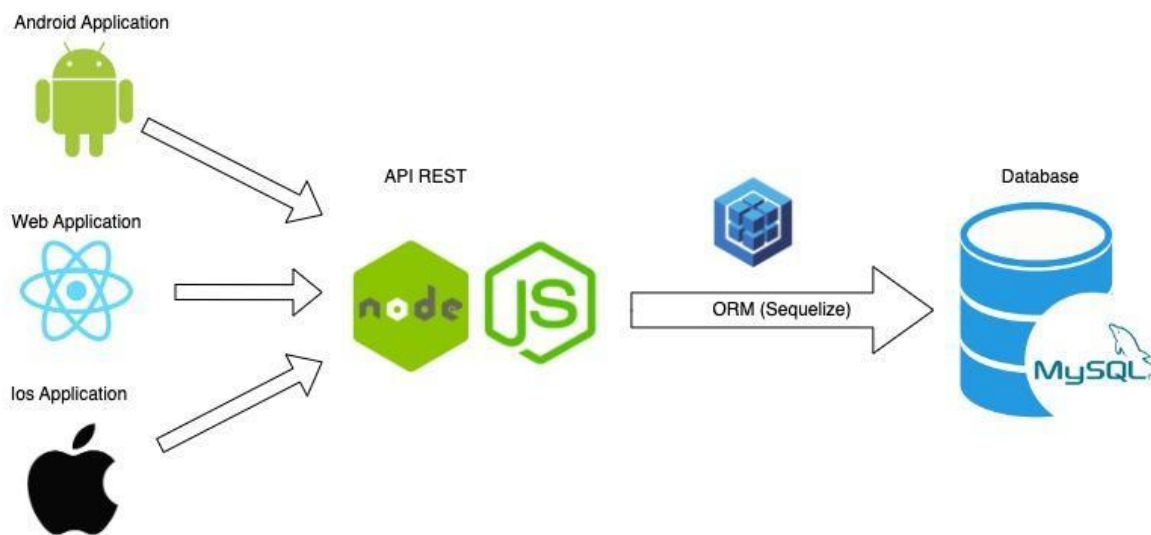


Ilustración 8: Escalabilidad posible

## 4.2.2 Patrones de diseño

### 4.2.2.1. Inyección de dependencias

La solución a nivel de API utiliza la inyección de dependencias. La inyección de dependencias es un patrón de diseño orientado a objetos, en el que se suministran objetos a una clase en lugar de ser la propia clase la que cree dichos objetos.

Para implementarlo, creamos una clase llamada dependency, encargada de instanciar todas las clases.

```
JS dependency.js X
src > routes > JS dependency.js > ...
23
24 const logRepository = new LogRepository();
25 const userRepository = new UserRepository(logRepository);
26 const companyRepository = new CompanyRepository(logRepository);
27 const depositRepository = new DepositRepository(logRepository);
28 const serviceGroupRepository = new ServiceGroupRepository(logRepository);
29 const depositRequestRepository = new DepositRequestRepository(logRepository);
30 const depositServiceRepository = new DepositServiceRepository(logRepository);
31 const departmentRepository = new DepartmentRepository(logRepository);
32 const cityRepository = new CityRepository(logRepository);
33 const depositImageRepository = new DepositImageRepository(logRepository);
34 const depositCalendarRepository = new DepositCalendarRepository(logRepository);
35 const bookingRequestRepository = new BookingRequestRepository(logRepository);
36
37 const mailService = new MailService(logRepository);
38 const userService = new UserService(userRepository, logRepository, mailService);
39 const authService = new AuthService(userRepository, logRepository);
40 const companyService = new CompanyService(companyRepository, logRepository, userRepository);
41 const depositRequestService = new DepositRequestService(depositRequestRepository, logRepository, companyRepository);
42 const depositService = new DepositService(depositRepository, logRepository, companyRepository, depositServiceRepository);
43 const serviceGroupService = new ServiceGroupService(serviceGroupRepository, logRepository);
44 const commonService = new CommonService(departmentRepository, cityRepository, logRepository);
45
46 const bookingRequestService = new BookingRequestService(bookingRequestRepository, logRepository);
47 const depositCalendarService = new DepositCalendarService(depositCalendarRepository, logRepository, depositRepository);
48
```

Ilustración 9: Inyección de dependencias

#### 4.2.2.2 Fachada Controller

También se implementó el patrón *Fachada controller*, como punto de entrada a nuestra aplicación. Se definió un controlador por entidad, a modo de agrupar todas las funcionalidades de un mismo objeto en una misma clase, manteniendo la alta cohesión.

```
▼ controllers
JS auth.controller.js
JS auth.controller.test.js
JS bookingRequest.controller.js
JS bookingRequest.controller.test.js
JS common.controller.js
JS common.controller.test.js
JS company.controller.js
JS company.controller.test.js
JS deposit.controller.js
JS deposit.controller.test.js
JS depositCalendar.controller.js
```

Ilustración 10: Fachada Controller

#### 4.2.2.3 Patron alta cohesión

Dentro de cada capa, se separa el código en diferentes clases, donde la asignación de responsabilidad se da a quien tiene la información para procesar los pedidos. Esto da como resultado una alta cohesión. Todos los métodos que se encuentran dentro de un módulo/clase tienen una relación interna, debido a que manipulan o consumen un mismo tipo objeto. Cuanto más cohesionados se encuentran los miembros de un módulo, mayor reutilización, mantenimiento y evolución podría llegar a tener, en términos generales.

Yendo a ejemplos puntuales, toda la lógica referente a la empresa se encuentra dentro de la clase `company.service`. Así ocurre con todas las diferentes entidades, donde el manejo de cada una de ellas se encuentra dividido en clases separadas, generando una alta cohesión.

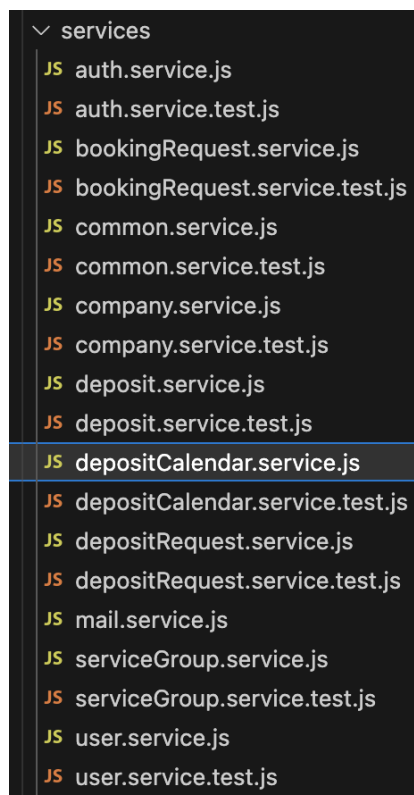


Ilustración 11: Alta Cohesión en la solución

#### 4.2.2.4 Patron bajo acoplamiento

Otro concepto aplicado es el de acoplamiento, que tiene que ver con el número de dependencias que un módulo tiene con otros módulos. Cuantas más dependencias

tiene un módulo más difícil de testear y más expuesto a cambios en el futuro podría estar, es por eso que se optó por mantener el acoplamiento al mínimo posible, reduciendo las dependencias.

### 4.2.3 Principios de diseño

Para el desarrollo de nuestra solución, nos basamos en los principios S.O.L.I.D [11], los cuales plantean principios que ayudan a que la solución sea más mantenible.

Se hizo gran foco en mantener la alta cohesión y el bajo acoplamiento, reduciendo los motivos que tiene para cambiar las diferentes clases, buscando cumplir con el principio de *Single Responsibility*.

Cada operación se colocó debidamente en la clase con la información para ejecutarla, evitando generar dependencias innecesarias que no promueven la mantenibilidad del sistema.

Además, mediante la creación de una arquitectura en capas, se logró una separación lógica que fomenta la baja cohesión, y cuya comunicación se da mediante la inyección de dependencias, logrando desacoplar las diferentes clases y módulos.

## 4.3 Atributos de calidad

A continuación se describen los atributos de calidad más importantes para el equipo, y las estrategias que se utilizan a nivel de arquitectura para satisfacerlas en la solución planteada.

### 4.3.1 Modificabilidad

Debido a que es un sistema que planea ser extendido y modificado a futuro por terceros, se definió la modificabilidad como un atributo de calidad principal.

Se especificó que el sistema debe ser entendible a fin de poder ser modificado por personas ajenas al equipo. Es por eso que se realizaron prácticas de refactor para dejar el código lo más claro posible y eliminar cualquier código duplicado o redundante. Además, resulta menos costoso cualquier cambio solicitado por el cliente

en cuanto más prolijo sea el código por lo que este atributo presenta un beneficio interno durante la ejecución del proyecto, y no solo a futuro.

### 4.3.2 Usabilidad

Un atributo de interés por el cliente fue la usabilidad por lo que se hizo énfasis en él. Actualmente el cliente cuenta con una plataforma no tan amigable, por lo que se realizó un esfuerzo por mejorar la experiencia del usuario en la nueva plataforma.

Para dicha mejora, se utilizó *Material Design* [6], un estilo de diseño desarrollado por Google, ya que provee un diseño conocido y por tanto intuitivo para el usuario final. Se buscó dejar claridad en lo que cada elemento presente en la pantalla es. Es decir, si se presenta un botón, el mismo debe tener el estilo correspondiente acorde a un botón, para que el usuario esté claro.

Además, la ubicación de los mismos, se respeto en toda la plataforma, manteniendo el orden y la coherencia de las diferentes pantallas.

Se siguieron estándares como ser:

- Los campos requeridos con asterisco y reborde rojo al no completarlos.
- Mensaje indicado que un campo es requerido, seguido al campo en cuestión.
- Botones deshabilitados hasta no completar los datos.
- Los campos numéricos solo permiten ingresar números.

Formulario de registro con los siguientes campos:

- Nombre
- Apellido
- Email\*
- Contraseña\*
- Repetir contraseña\*
- Industrias con la que estoy familiarizada... ?
- He leído y acepto las políticas de privacidad.\*

Botones: Registrarse (deshabilitado), Cancelar

Ilustración 12: Botones deshabilitados

Formulario de registro con los siguientes campos:

- Nombre: Romina
- Apellido: Areco
- Email\*: romina.areco@hotmail.com
- Contraseña\*: .....
- Repetir contraseña\*: .....
- Industrias con la que estoy familiarizada/o\*: Informatica ?
- He leído y acepto las políticas de privacidad.\*

Botones: Registrarse (activo), Cancelar

Ilustración 13: Completado de datos

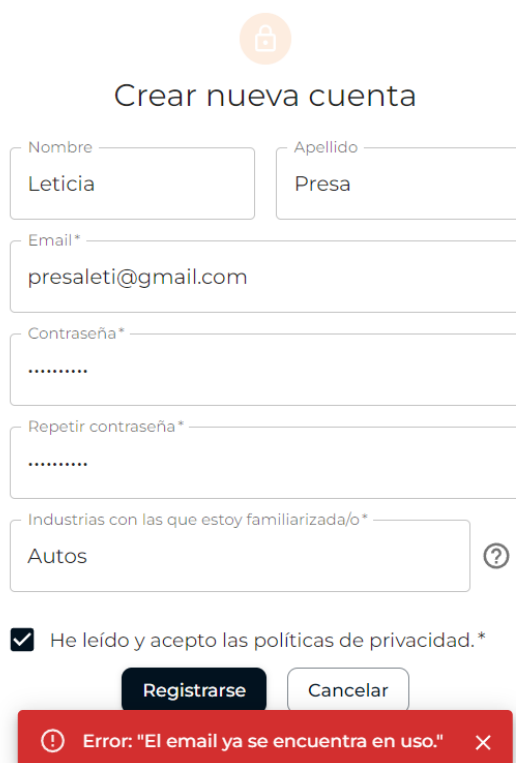
Se buscó que el uso de la plataforma sea lo más fácil y amigable para incentivar su uso y participación de los diferentes usuarios, con el fin de que pudieran cargar la información necesaria y mantenerla actualizada con facilidad.

Se realizaron validaciones de datos requeridos y formatos, previniendo que el usuario cometa errores. Todos los formularios de ingreso fueron creados de forma de prevenir el error del usuario.

A su vez, si ocurre un error, siempre se informa el motivo del mismo para dar claridad y transparencia al usuario.

Se buscó cumplir con las heurísticas de Nielsen [12], las cuales detallamos a continuación y ejemplificamos.

Visibilidad del estado del sistema: Se buscó que siempre que se presione un botón, sea transparente al usuario el resultado de la acción. Por ejemplo, para búsquedas, se muestra un loading indicando que se está buscando y procesando el pedido. Para operaciones de creación, modificación y eliminación, se muestra un icono de *loading*, pero además, se muestra un mensaje indicando el resultado de la operación.



Crear nueva cuenta

Nombre Leticia Apellido Presa

Email\* presaleti@gmail.com

Contraseña\* .....

Repetir contraseña\* .....

Industrias con las que estoy familiarizada/o\* Autos ?

He leído y acepto las políticas de privacidad.\*

Registrar Cerrar

Error: "El email ya se encuentra en uso." X

Ilustración 14: Estado del sistema - Error



Crear nueva cuenta

Nombre Apellido

Email\*

Contraseña\*

Repetir contraseña\*

Industrias con la que estoy familiarizada... ?

He leído y acepto las políticas de privacidad.\*

Registrar Cerrar

Usuario registrado exitosamente X

Ilustración 15: Estado del sistema - OK

Relación entre el sistema y el mundo real: Se usó la terminología indicada por el cliente, ya que es la utilizada en el rubro.

El término arrendamiento, por ejemplo, es el utilizado en cuanto a espacios de almacenamiento refiere, y no alquiler, como uno pensaría.

Lo mismo ocurre con el término depósitos y espacios de almacenamiento, que son palabras ampliamente conocidas en el sector.

Se buscó entonces la mayor claridad en la aplicación, de modo que resulte fácil e intuitivo su uso y prevenga errores, ya sea, poniendo foco en la terminología, como también en los iconos y la visual.

Por ejemplo, la acción de **“Tengo espacio”**, va acompañada de un icono con un +, lo cual sugiere una operación de “crear/agregar”.



Ilustración 16: Plataforma WareUP - Menú superior

Control y libertad del usuario: El usuario tiene posibilidad de modificar o eliminar toda la información que ingresa. Nos cercioramos que tenga control sobre la información que aporta, ya sea del deposito, la empresa o sus datos personales. Siempre tiene la oportunidad de corregirlos y/o complementar la información existente.



Ilustración 17: Plataforma WareUP - Mis empresas

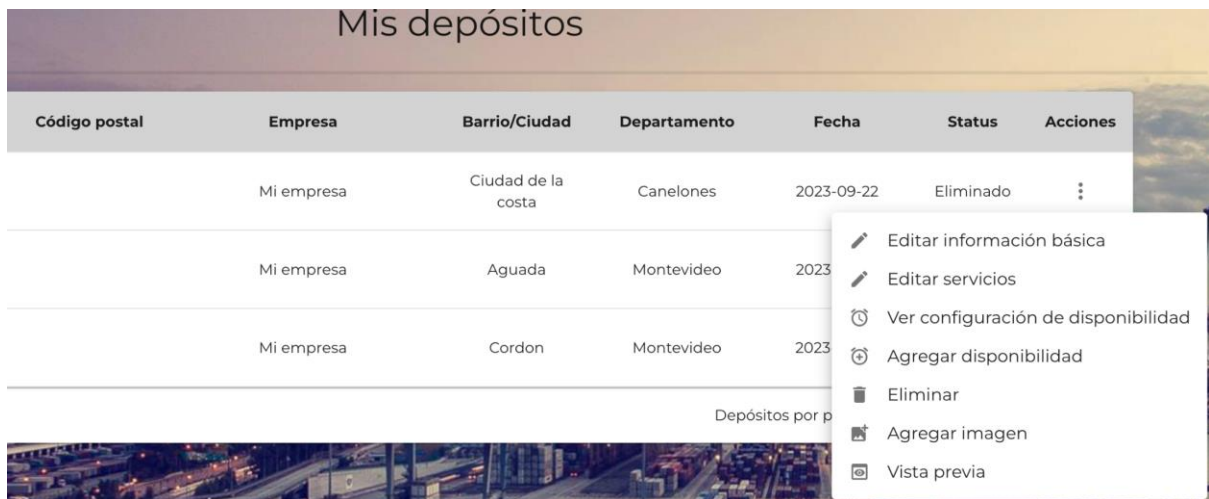


Ilustración 18: Plataforma WareUP - Mis depósitos

Datos personales

Nombre  
Romina

Apellido  
Areco

Email  
romina.areco@hotmail.com

Industrias con la que estoy familiarizada/o  
Informática

[✎ Editar perfil](#)

Ilustración 19: Plataforma WareUP - Editar perfil

Consistencia y estándares: Al utilizar la librería de Material UI [13], nos aseguramos de cumplir con los estándares y lo esperado por los usuarios. Tres puntitos representan un menú desplegable. El mensaje verde lo asociamos a una acción culminada exitosamente, mientras rojo representa un error.



Ilustración 20: Plataforma WareUP - Menu tres puntitos



Ilustración 21: Plataforma WareUP - Mensajes resultado

Prevención de errores: Evitar los errores del usuario fue otro punto importante. Para eso, incorporamos varias ayudas al usuario como ser:

- Ayudas en los textos libres, como ser, sugerencias de textos a utilizar.
- Posibilidad de reutilizar los datos guardados de Google.
- Ingreso doble de contraseña para validar que sea la deseada por el usuario



Ilustración 22: Plataforma WareUP - Buscador

<p>Crear nueva cuenta</p> <input type="text" value="Nombre"/> <input type="text" value="Apellido"/> <input type="text" value="Email*"/> <input type="password" value="Contraseña"/> <input type="password" value="Repetir contraseña"/>	<p>Crear nueva cuenta</p> <input type="text" value="Nombre"/> <input type="text" value="Apellido"/> <input type="text" value="Email*"/> <input type="password" value="Contraseña*"/> <input type="password" value="Repetir contraseña*"/>
---	---

Ilustración 23: Sugerencia autocompletar

Ilustración 24: Validación contraseña

Flexibilidad y eficiencia de uso: La plataforma se desarrolló de modo que fuera simple de utilizar. Al ingresar se cuenta con un buscador donde uno simplemente ingresa una ubicación y ya tiene acceso a los diferentes depósitos.

Además, el registro solicita datos mínimos y sencillos, para agilizar y simplificar el proceso de enrolamiento en el sistema.

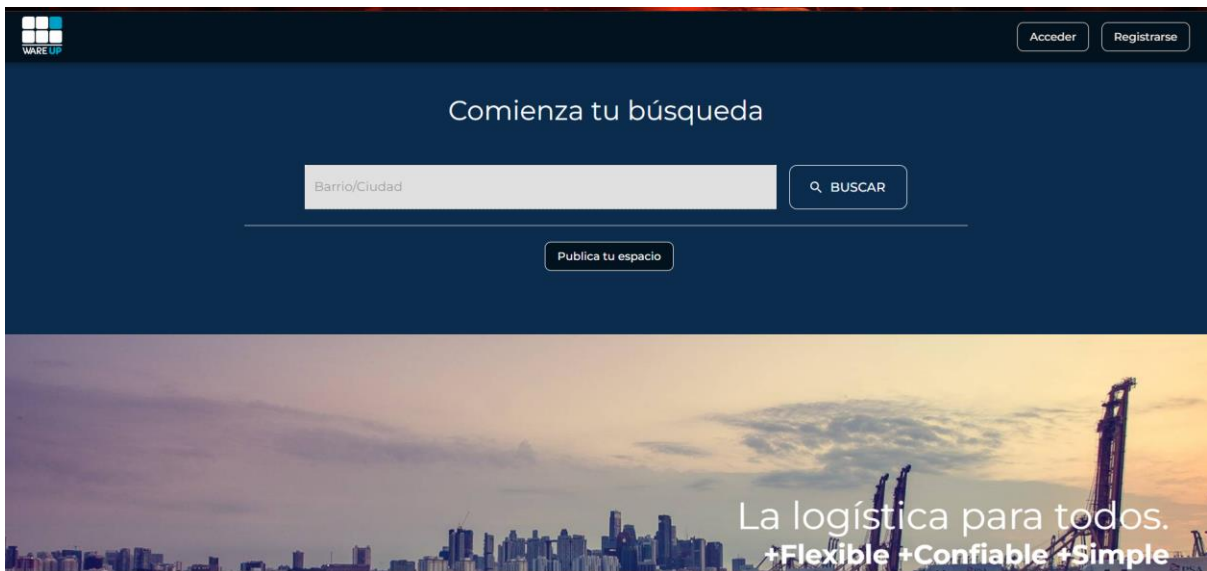


Ilustración 25: Plataforma WareUp - Home

Diseño estético y minimalista: Las pantallas se diseñaron para contener sólo la información relevante y necesaria. No se hicieron agregados extra que no aporten valor y complejizan la usabilidad.

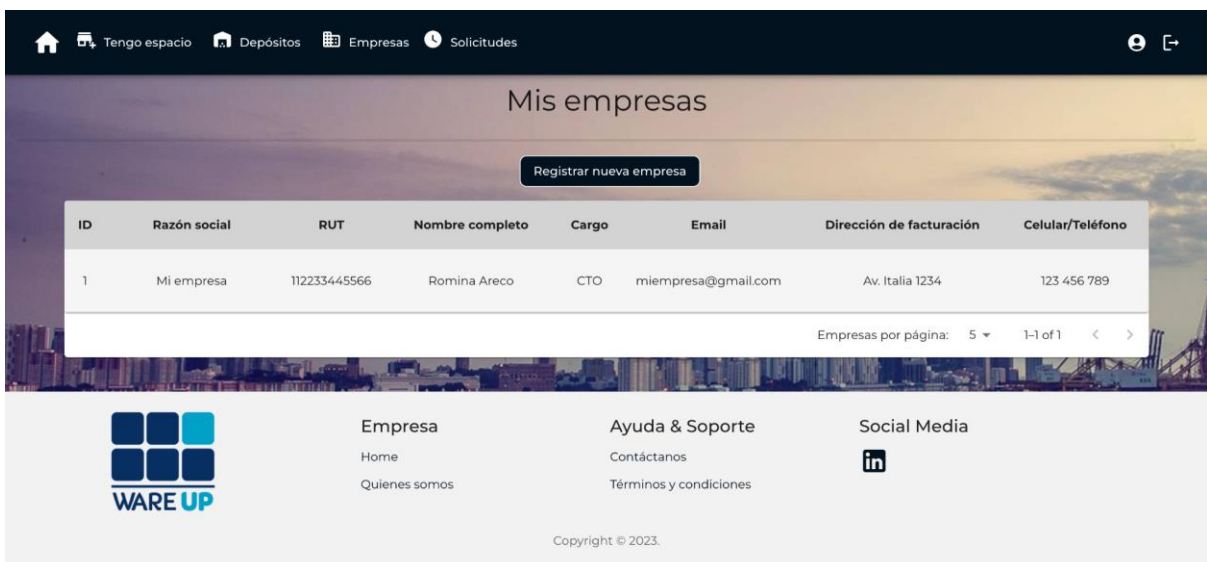


Ilustración 26: Plataforma WareUP - Mis empresas

Ayudar a los usuarios a reconocer, diagnosticar y corregir los errores: Todos los errores capturados por el sistema se muestran en pantalla con un mensaje amigable y claro para el usuario.

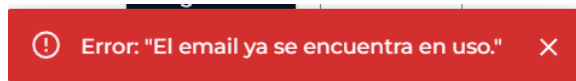


Ilustración 27: Error mail



Ilustración 28: Error contraseña

### 4.3.3 Portabilidad

El desarrollo de la aplicación web se implementó de tal manera que pueda ser soportada por diferentes navegadores, en diferentes sistemas operativos, por lo que puede ser accedida desde el sistema operativo Windows, los, Android, Linux, y soportando los navegadores Google Chrome, Mozilla, Edge o Safari.

Al momento de la implementación se tuvo en cuenta los diseños para que pudiera, la plataforma, correr en los agentes mencionados sin dificultad o errores.

Se utilizó a nivel de diseño librerías CSS para customizar los diseños y poder ajustar las vistas con el objetivo de una correcta visualización y, por consiguiente, conseguir la portabilidad de la plataforma desarrollada.

Además, se hizo gran énfasis en el atributo *responsive*, a modo de que pueda accederse a través de un browser, ya sea desde una computadora, tablet o celular.

Se realizaron las diferentes pruebas en los diferentes dispositivos, validando el correcto funcionamiento de la misma en ellos. La plataforma se visualizó según lo esperado en cada uno de ellos, sin presentar errores.

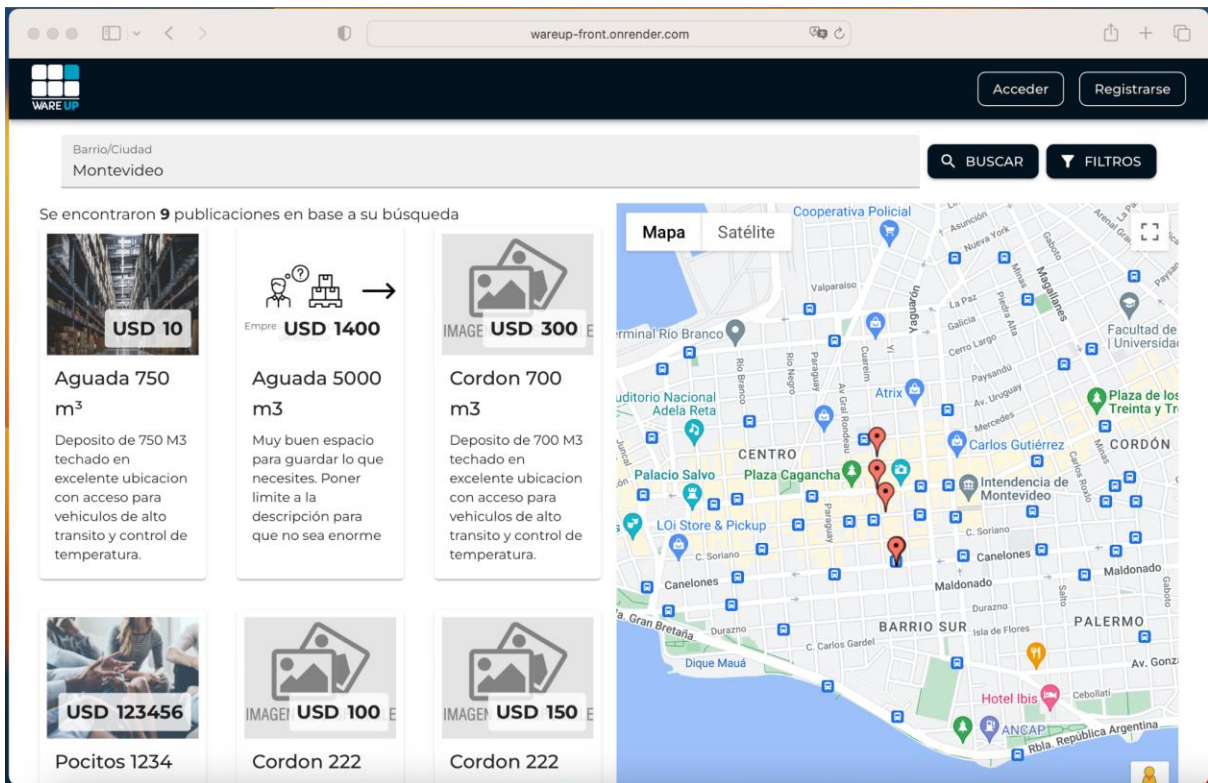


Ilustración 29: Buscador en navegador Safari

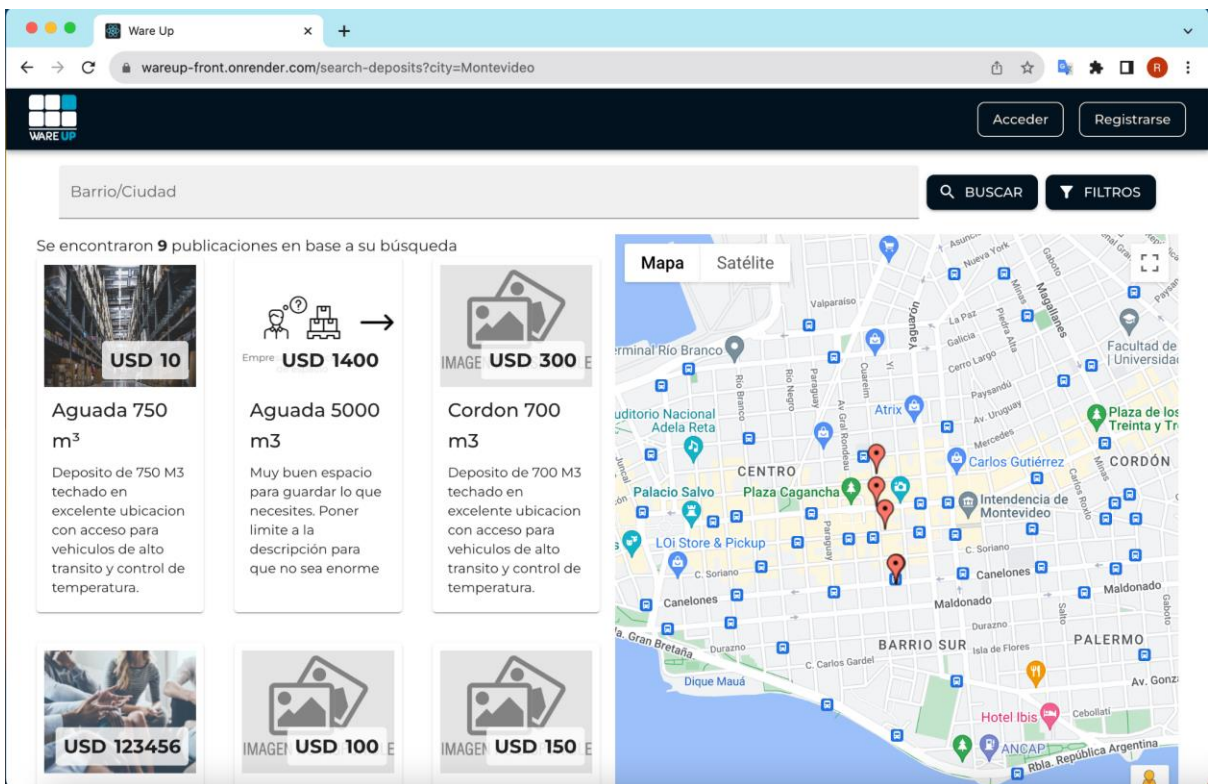


Ilustración 30: Buscador en navegador Google Chrome



Ilustración 31: Buscador en mobile

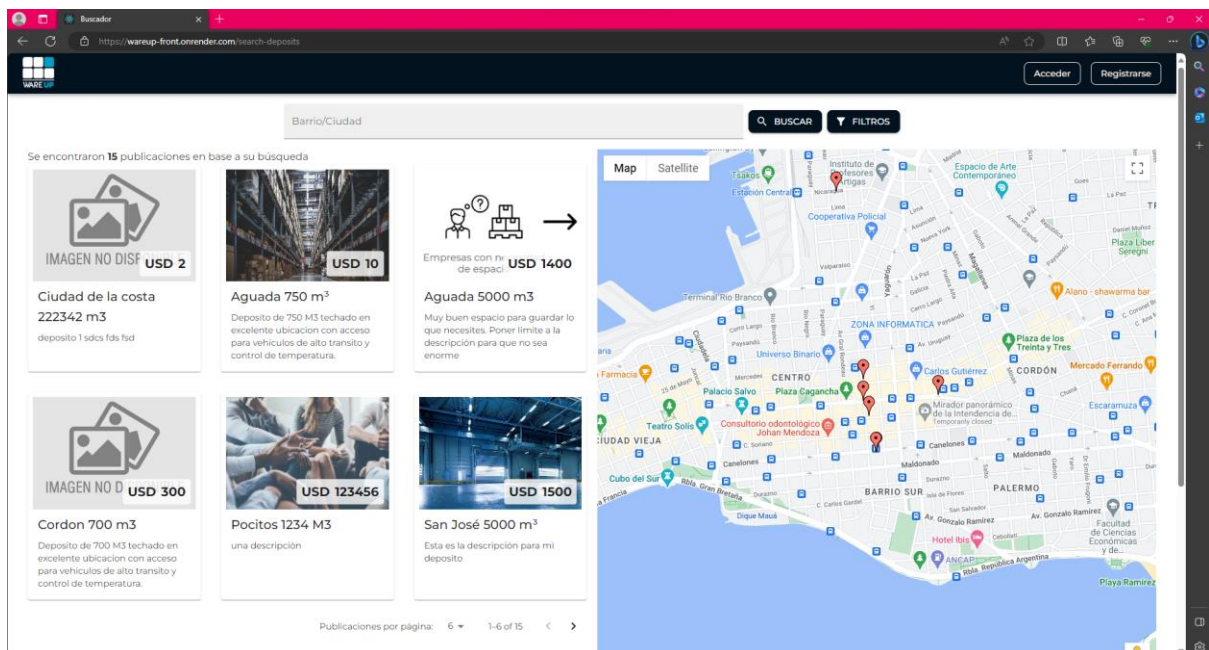


Ilustración 32: Buscador en navegador Microsoft Edge

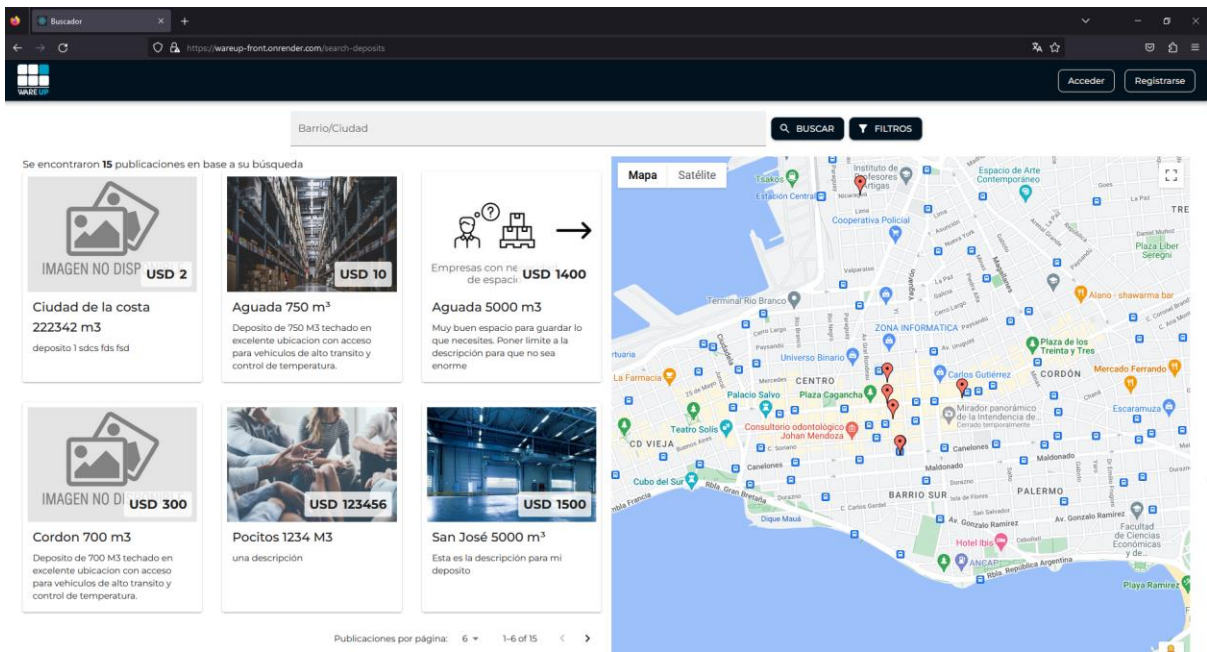


Ilustración 33: Buscador en navegador Mozilla

#### 4.3.4 Interoperabilidad

La interoperabilidad es un atributo cuyo propósito es medir la capacidad de intercambio de información de la aplicación con otros sistemas o con el entorno donde opera.

En ese sentido, mediante el patrón cliente-servidor se creó una API Rest que se encuentra disponible, no solo para la aplicación web desarrollada, si no que para cualquier otro sistema que quiera conectarse, cumpliendo con este atributo.

#### 4.3.5 Seguridad

Otro atributo de calidad relevante es la seguridad. Para implementarla se adiciona una validación sobre las llamadas recibidas donde se valida la autenticación y autorización por parte del usuario sobre el llamado, validando que proviene de un usuario autenticado y, además, de un usuario autorizado al acceso de la información. Además, la información sensible en la base de datos se encuentra encriptada o hasheada, manteniendo la confidencialidad de la información.

#### 4.3.6 Escalabilidad

Se desarrolló un producto siguiendo una arquitectura distribuida que permite escalar fácilmente los servidores según sus demandas.

Si bien no se implementó, como parte del proyecto se incluyen dos análisis para el despliegue de la aplicación en Amazon y en Azure DevOps. Ambas plataformas permiten fácilmente agregar recursos a los servidores, aumentar cantidad de servidores, utilizar balanceadores de cargas, entre otros. Todo esto enfocado en obtener una escalabilidad, en relación a la demanda recibida, reduciendo los costos a los necesarios.

#### 4.3.7 Disponibilidad

Al igual que se mencionó en el punto de seguridad, si bien no se llegó a implementar, como parte del proyecto se le brindó un instructivo de despliegue en Amazon y en Azure DevOps. Ambas plataformas, siguiendo la documentación entregada, garantizan la disponibilidad de la aplicación. Las mismas ofrecen dentro de sus características configuraciones de, por ejemplo, mínimo de servidores activos, que garantiza que, si falla un servidor, se apaga o rompe, inmediatamente se crea uno nuevo poniéndose en funcionamiento para atender a la demanda. Con esto se garantiza que siempre esté operativa la web. Esta característica es aplicable tanto a servidor de *frontend*, como *backend*, como base de datos.

Por otra parte, se cuentan con alertas sobre la capacidad de los servidores, y una configuración de rendimiento mínimo que garantiza que, en caso de saturarse un servidor, inmediatamente incremente sus capacidad o se agregue otro servidor, de forma de que no se produzcan timeout al atender a la demanda.

### 4.4 Herramientas y tecnologías de desarrollo

Se definieron las herramientas y tecnologías a utilizar para llevar a cabo el proyecto en base a las necesidades del cliente y a las funciones y características de cada una.

**Node js - v18.17.1:** NodeJs es un entorno de ejecución en tiempo real con la capacidad de ejecutar un programa escrito en JavaScript. Permite que, código

javascript que antes solo podía ejecutarse en el navegador, ahora pueda correr en las computadoras, como aplicaciones independientes

Tanto JavaScript como Node.js se ejecutan en el motor de tiempo de ejecución JavaScript V8. Este motor toma el código JavaScript y lo traduce en código de máquina. El código de máquina es el código de nivel más bajo, lo que la computadora puede ejecutar sin necesidad de interpretarlo primero, ignorando la compilación y por lo tanto aumentando su velocidad.

Todo nuestro *backend* está desarrollado utilizando esta tecnología.

**React - v18.2.0:** Es una librería JavaScript de código abierto, en la cual se puede escribir tanto en Javascript como en JSX. El JSX permite combinar HTML y Javascript en el código, facilitando el desarrollo. Maneja capas de vista de aplicaciones web y móviles por lo que facilita el desarrollo multiplataforma. Además, proporciona una estructura independiente basada en componentes fomentando la reutilizabilidad del código.

Se decidió desarrollar el *frontend* de la aplicación web en este lenguaje de acuerdo a los requerimientos del cliente.

**Material UI - v5.14.1:** Es una biblioteca de componentes de interfaz de usuarios para react. Fue diseñada para facilitar los desarrollos webs dando una imagen moderna. Sigue los principios de *Material Design* [6], que son un conjunto de especificaciones de diseño que definen cómo deben ser las interfaces de usuario, que se debe hacer y que no. Además define qué componentes se pueden utilizar.

**MySQL:** Es un sistema de gestión de base de datos de código abierto, lo que significa que el código fuente es de público acceso y que cualquiera puede inspeccionar, modificar y mejorar. MySQL nos permite almacenar y acceder a datos utilizando diferentes motores de almacenamiento, como ser InnoDB, CSV y NDB.

Dentro de las funcionalidades a destacar de MySQL tenemos:

- Es multiplataforma, soporta Mac, Windows, Linux y Unix.
- El RDBMS soporta grandes bases de datos con millones de registros.

- Soporta gran variedad de tipos de datos como enteros, decimal, float, double, char, varchar, date, time y blob.
- Utiliza sistemas de acceso a través de contraseñas encriptadas
- Incluye varios protocolos de conexión.

**Sequelizer - v6.32.1:** Es un ORM (*Object relational mapping*), que nos facilita el llamado a la base de datos brindándonos funciones javascript para ello. Con el, no es necesario escribir consultas de SQL para manipular los datos de la base de datos si no que podemos usar consultas comunes de javascript, reduciendo los tiempos de desarrollo, ya que el ORM se encarga de conexión, actualización, consultas y mapeo de datos en entidades o modelos. Además, crea las tablas, las actualiza y las popula con información, todo eso utilizando programación orientada a objetos y sin necesidad de escribir una sola sentencia de base de datos.

**Render:** Es una nube unificada para crear y ejecutar aplicaciones y sitios web con certificados TLS gratuitos, una CDN (*Content Delivery Network*) global, protección ante denegación de servicios, redes privadas e implementaciones automáticas desde git. Nos permite tener disponibles varias publicaciones, actualizadas con lo último en la rama configurada, pudiendo implementar integración continua y deploy continuo.

**Git y GitHub:** Se utilizó Git y GitHub para llevar adelante el control de las versiones del producto, comparar las versiones y las mejoras, saber qué integrante hizo qué cosa y cuándo. Recibir *feedback* de los pull request y colaborar, permitiendo a varios integrantes trabajar a la vez en el producto en diferentes branches.

**Jira:** Es un potente software de gestión de proyectos para organizar el proyecto y sus diferentes etapas. Dentro de sus funcionalidades está la posibilidad de organizar el trabajo en sprints, medir el rendimiento y eficacia del equipo. Es flexible y se adapta fácilmente al flujo de trabajo definido.

**Postman:** Esta aplicación permite el testeo de la API Rest al enviar peticiones con diferentes request, y visualizando la respuesta. Brinda una interfaz gráfica muy intuitiva y fácil de utilizar. A su vez permite grabar proyectos y compartirlos, de modo que, no solo sirva para testear el *backend*, si no como apoyo para el desarrollo del *frontend*, al ver el *request* y *response* esperado.

**SendGrid:** Es una solución para el envío de email basada en la nube, que permite además definir templates a utilizar con variables enviadas desde la API, centralizando todo lo relacionado a la comunicación vía mail desde allí. Almacena los diferentes templates y permite tener versionado de los mismos. Se destaca la facilidad tanto para su uso como para la integración en la API REST. Ver [Anexo 13 - SendGrid en el backend](#).

**REST API:** La plataforma desarrollada en cuestión implica una API REST donde se siguieron los lineamientos y principios del diseño REST.

**Google Draw:** Esta aplicación permite la creación de diagramas para el desarrollo de la aplicación. Brinda todos los componentes necesarios para diagramar y resulta fácil de utilizar, además de ser gratuita.

**Google Meet:** Para las reuniones internas, con el cliente y con el tutor se utilizó Google Meet, debido a la facilidad de su uso y agilidad, permitiendo unirse y participar desde cualquier dispositivo, ya sea web o mobile.

**Google Maps:** Es un servicio de mapas en línea desarrollado por Google que ofrece mapas interactivos y funcionalidades relacionadas con la geolocalización, como la visualización de mapas, direcciones, información sobre lugares de interés, navegación paso a paso, imágenes satelitales y mucho más. La integración de este servicio con la aplicación en React fue de mucha ayuda para disponibilizar una funcionalidad que aporta valor a la plataforma. Ver [Anexo 16 - Guía integración con Google Maps](#) para visualizar la guía proporcionada al cliente de cómo utilizarla.

**Google Analytics:** Google Analytics es una plataforma de análisis web desarrollada por Google que permite a los propietarios de sitios web y aplicaciones rastrear y analizar datos relacionados con el tráfico, el comportamiento de los usuarios y otras métricas relevantes. Gracias a este servicio se puede monitorear el comportamiento de los usuarios en la plataforma y encontrar posibles mejoras resultando en incrementar la buena experiencia de los usuarios. Ver [Anexo 15 - Guía integración con Google Analytics](#) para visualizar la guía proporcionada al cliente de cómo utilizarla.

## 4.5 Estructura de la solución

El proyecto se desarrolló en dos soluciones diferentes, una para el *frontend* y otra para el *backend*.

La aplicación de frontend fue desarrollada en React. Esta aplicación se comunica con el servidor a través de una API REST. Ver [Anexo 11 - Endpoints REST del backend](#). El servidor una vez realizadas las validaciones necesarias en la capa de negocio se comunica con la base de datos y realiza las acciones necesarias para retornar datos, actualizar, dar de alta o borrar.

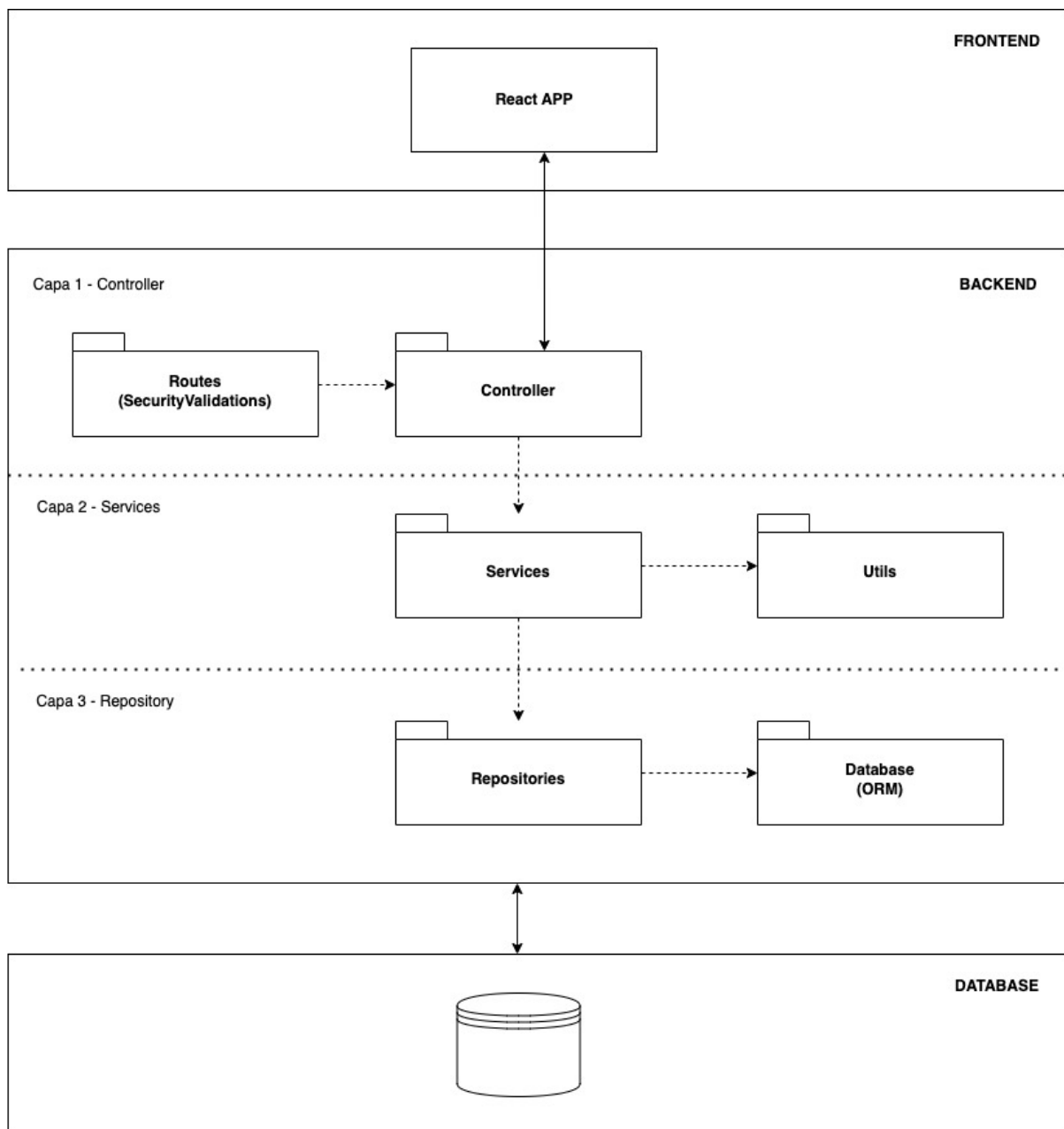


Ilustración 34: Diagrama de capas a nivel de plataforma

## Frontend:

El frontend fue desarrollado utilizando *React*, una librería de JavaScript para el desarrollo de interfaces de usuario. *React* presenta como ventaja la reutilización de componentes, es por eso que se definió la siguiente estructura a nivel de solución:

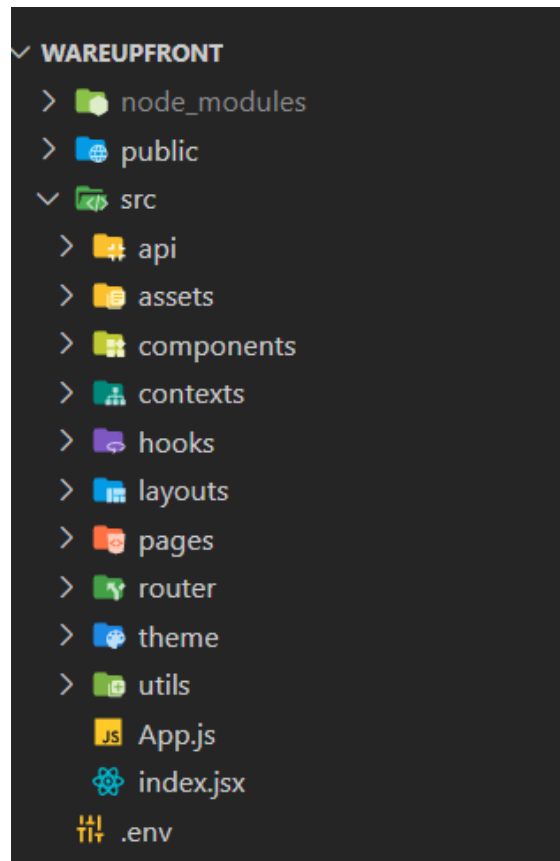


Ilustración 35: Estructura frontend

Dentro de la solución propuesta, tenemos la carpeta *components*, donde se definen los componentes que son reutilizados a lo largo de toda la aplicación. Aquí se definieron componentes desde lo más básico, como ser botones y etiquetas, hasta lo más complejo como ser formularios de creación o edición.

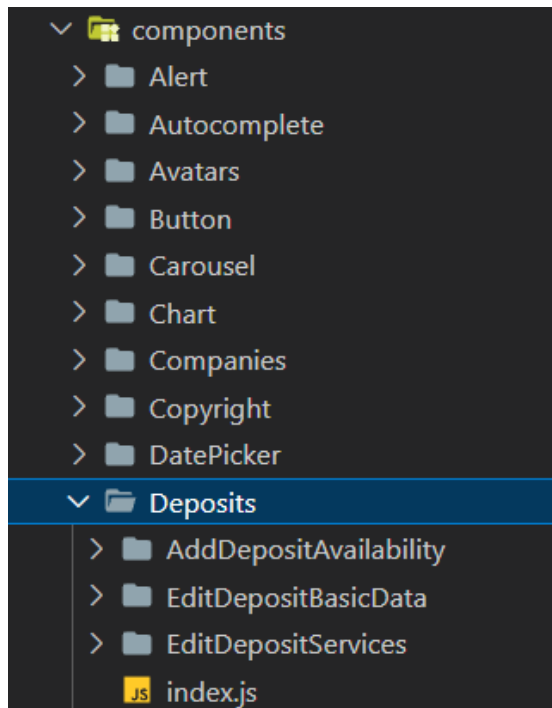


Ilustración 36: Estructura frontend - componentes

En la carpeta *context* se maneja todo lo referente al usuario logueado, los datos, permisos, y demás.

Dentro de *pages*, se definen todas las paginas a ser navegadas. A nivel de *pages* es que se genera la reutilizaron de los componentes, previamente definidos.

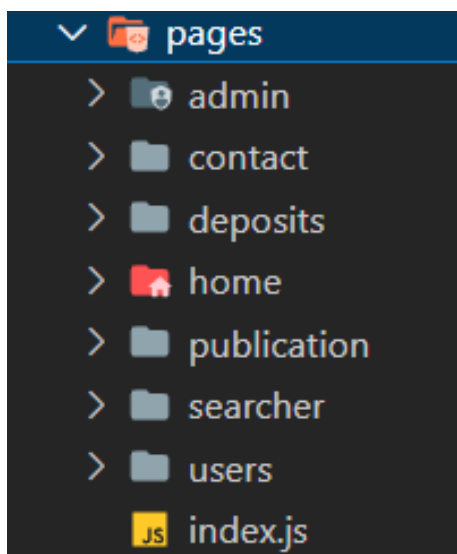


Ilustración 37: Estructura frontend - pages

En la carpeta *layouts* tenemos la definición de qué barra de navegación se utilizará en base al usuario que haya iniciado sesión.

En la carpeta *router* tendremos la definición de rutas para toda la web. Además del control de privilegios asociados a cada usuario, si es un usuario backoffice puede acceder a ciertas rutas de la web que el usuario cliente no.

Finalmente tenemos la carpeta *api* donde se encuentran integrados los métodos expuestos por el *backend*, es decir, en este nivel se resuelve la comunicación con la API REST.

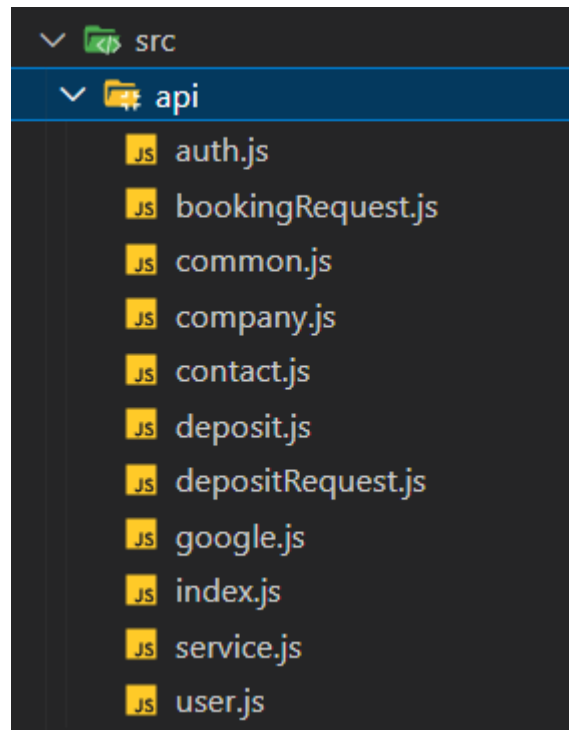


Ilustración 38: Estructura frontend - api

### **Backend:**

A continuación detallaremos la solución de la API REST, indicando en qué consiste cada capa de la solución propuesta:

#### Controller

En esta capa se encuentra el punto de acceso a nuestra API REST desarrollada en *React*. Tiene la responsabilidad de recepcionar las peticiones, y brindar la respuesta. Es el punto de entrada a la API y en ella se realizan las validaciones del request, corroborando la estructura y existencia de los campos.

```
async getByCompany(req, res, next) {
  const { companyId } = req.params;
  if (!companyId || typeof companyId !== "string") {
    return res.status(400).json({ message: "Invalid Params" });
  }
  const result = await this.service.getByCompany(companyId);
  return res.status(200).json(result);
}
```

Ilustración 39: Estructura backend - controller

Si no se encuentra la información requerida, se retorna un error 400 indicando *BadRequest*.

Si el request está bien se procede entonces con la operación retornando 200-OK si se completa la operación sin errores o 500-ServerError si se presenta un error en las siguientes capas.

En esta capa entonces solo se hacen validaciones de formato. No se hacen validaciones de reglas de negocio.

### Services

La capa de servicio es la encargada de realizar toda la lógica de negocio antes de realizar cambios o consultas a la base de datos u otros servicios. Aquí es donde se implementan los algoritmos, modificación y cálculos necesarios según la funcionalidad lo requiera.

Se procesan las peticiones según la lógica de negocio. Se recibe la información de la capa controller y, con la ayuda de los métodos expuestos en la capa repository, se completa el pedido retornando un resultado que indica si se procesó correctamente u ocurrió un error.

Para el caso donde ocurrió un error se retorna un mensaje de error que indica el motivo de la falla. Los errores que retorna pueden ser errores de lógica de negocio (usuario no existente) o de servidor (error al conectarse al servidor SMTP para envío de mails).

```

async getByCompany(companyId){
  let hasError = false;
  let message = null;
  let resultCode = enums.resultCodes.OK;
  let depositRequests = null;

  try{
    depositRequests = await this.repository.getByCompany(companyId);
  }
  catch (error) {
    resultCode = enums.resultCodes.genericError;
    hasError = true;
    message = 'Ha ocurrido un error obteniendo las solicitudes de depositos de la compania';

    this.log.create('Error in getByCompany: '+ error, enums.logsType.service);
  }

  return { message, hasError, resultCode, depositRequests };
}

```

Ilustración 40: Estructura backend - services - get

En todos los métodos de esta capa se implementó un *try-catch* de forma de siempre retornar un error controlado. Además, dentro del *catch* se realiza un proceso de logueo, para dejar registro del error real. Este proceso de logueo, según el modo en que se encuentre configurado (dev-noDev) realiza diferentes comportamientos. Si o si, todo error es logueado a nivel de base de datos, en una tabla llamada log. Y dependiendo de si está configurado modo dev-noDev o si no hay acceso a la base de datos, se adiciona además un log a nivel de consola, para fácil visualización.

```

async create(description, type) {
  try {
    if(config.mode == 'dev'){
      console.log(description);
    }
    return this.log.create({
      description,
      type
    });
  }
  catch (error) {
    console.log(error);
  }
  return null;
}

```

Ilustración 41: Estructura backend - service - create

## Repository

La responsabilidad de esta capa es únicamente el acceso a base de datos. No se agrega ninguna lógica extra aquí sino simplemente realizar acciones de lectura, escritura, modificación o borrado. En ella se encuentran las operaciones básicas de *insert*, *update*, *delete* y *get*.

```
async create(request) {
  try {
    return this.model.create(request);
  }
  catch (error) {
    this.log.create('Error in create: '+error, enums.logsType.database);
  }
  return null;
}
```

Ilustración 42: Estructura backend - repository - create

```
async getByCompany(companyId) {
  try {
    return this.model.findAll({
      where: {companyId: companyId},
      include: [CompanyModel, CityModel]
    });
  }
  catch (error) {
    this.log.create('Error in getByCompany: '+error, enums.logsType.database);
  }
  return null;
}
```

Ilustración 43: Estructura backend - repository - get

Las entidades, representan una capa transversal a la solución, y se ve reflejada en el siguiente diagrama de clases.

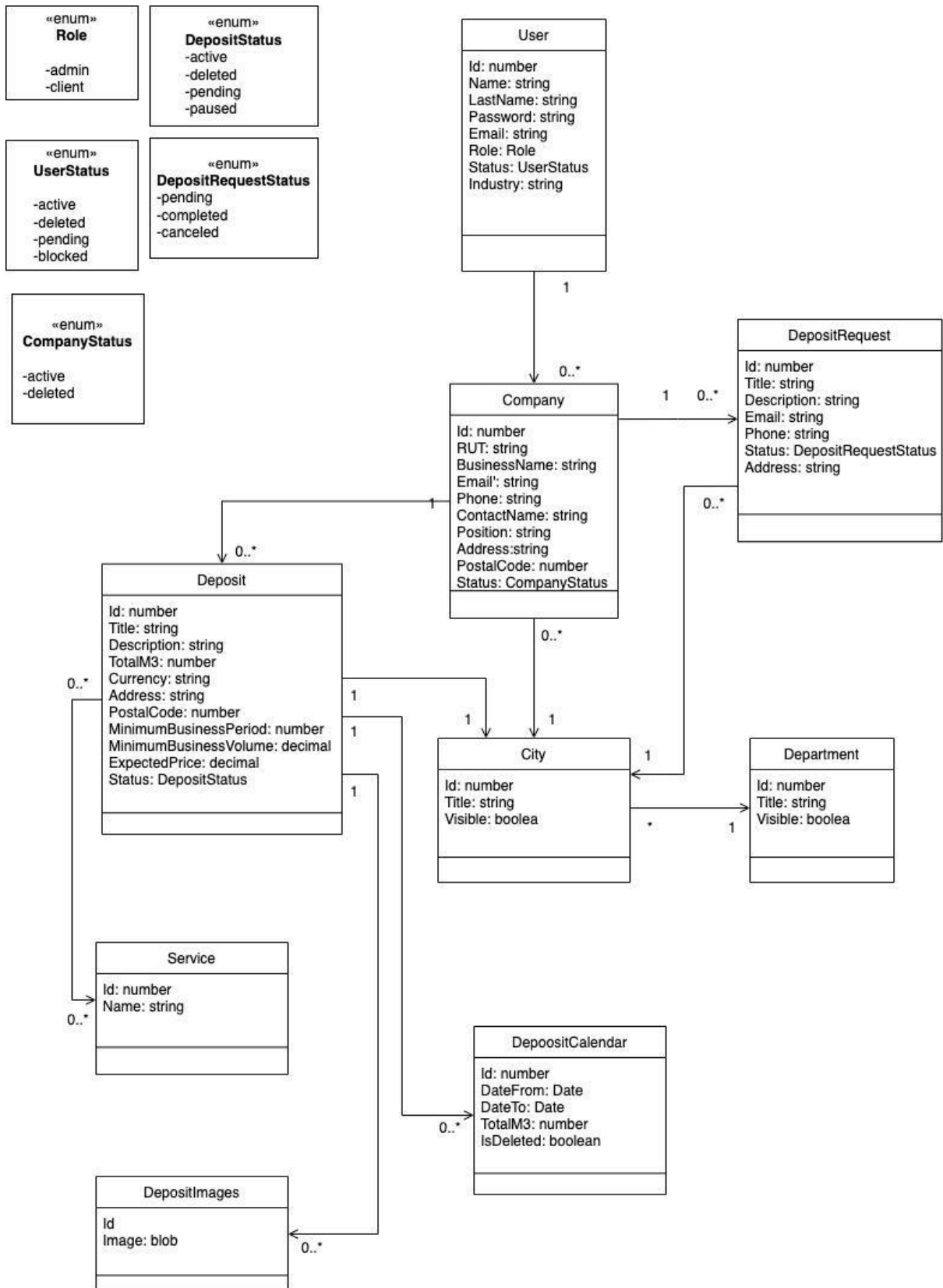


Ilustración 44: Diagrama de clases

## Base de datos

Sobre la solución propuesta para base de datos, se implementó creando una base de datos relacional, que responde a la siguiente estructura:

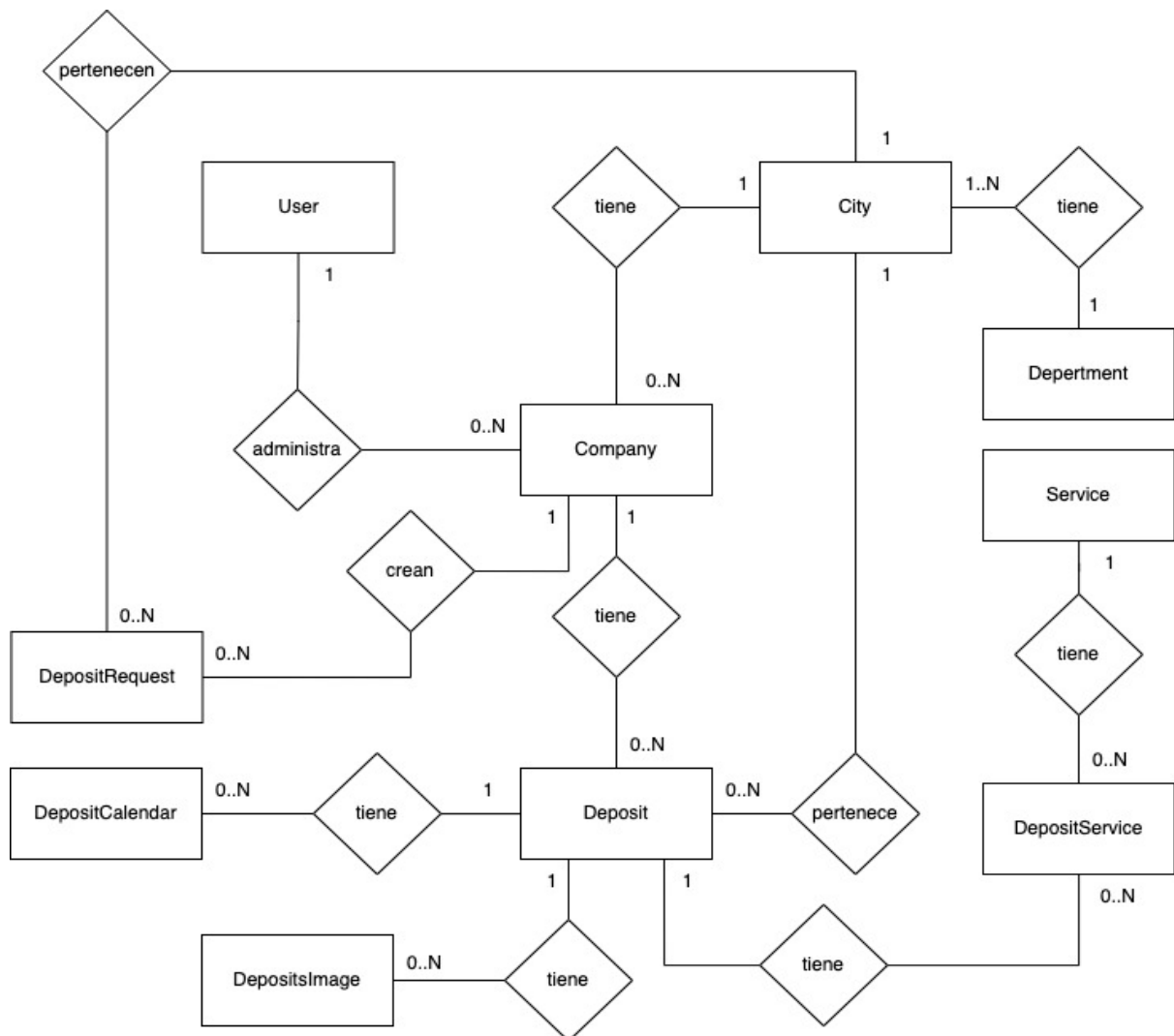


Ilustración 45: Diagrama MER

Para la creación de las tablas, se utilizó la estrategia CodeFirst, que consiste en definir las clases a nivel de *backend* y, mediante la utilización de un ORM, realizar el pasaje a tablas generando la base de datos automáticamente, sin necesidad de escribir sentencias SQL.

## 4.6 Conclusión

Como se ha mencionado en secciones anteriores, uno de los requisitos fundamentales era la escalabilidad de la solución planteada.

La estructura de la solución planteada tiene como foco entonces que se puedan agregar, modificar o eliminar fácilmente entidades, incorporando nuevas funcionalidades y escalando la aplicación a nivel de lógica.

La arquitectura propuesta, por su parte, también plantea la posibilidad de escalar, ya sea incorporando una nueva aplicación, o escalando a nivel de servidores, ya sea horizontal o verticalmente, aumentando la cantidad o los recursos de estos.

# 5 Gestión del proyecto

## 5.1 Introducción

Este capítulo describe cómo fue realizada la gestión del proyecto. Se explica el marco metodológico, justificando los motivos por los que se seleccionó Scrum como marco de trabajo para la gestión del proyecto y BDD (*Behavior Driven Development*) para el desarrollo de ingeniería de software. También se detalla cómo fue realizada la adopción a dicha metodología, como se realizó la planificación y seguimiento del proyecto y que herramientas de gestión se utilizaron.

Cabe destacar que el proyecto comenzó con un periodo de investigación de un mes y medio aproximadamente donde el objetivo principal fue entender el contexto en el que vamos a estar trabajando y reducir la incertidumbre que el equipo tenía respecto a los requerimientos del cliente.

## 5.2 Marco metodológico

En esta sección se detallan las características principales que llevaron al equipo a tomar la decisión de adoptar *Scrum* como marco de trabajo para gestionar el proyecto y BDD para el desarrollo de ingeniería de software.

### 5.2.1 Marco de gestión

Fue necesario definir un marco de gestión que se adecuara a la situación del proyecto. Fue por esto que se analizaron las principales características del escenario del proyecto y en base a eso se definió el marco de gestión. Para definir el ciclo de vida a utilizar durante este proyecto, entendemos que es importante tener en cuenta los siguientes aspectos.

1. Consideramos que tenemos que ser capaces de llevar de forma rápida distintos entregables al cliente, que permitan observar la evolución del producto y obtener un *feedback* temprano y oportuno, de forma tal que podamos validar los avances y realizar las mejoras necesarias en caso de que lo amerite.
2. Dado que hay una fecha fija de entrega, se decidió ordenar el desarrollo desde lo más prioritario para tener un producto funcional hacia lo menos prioritario

o necesario, de manera que si no se logra cumplir con la fecha de entrega al menos se cumple con lo más relevante. Vale la pena mencionar que el cliente cuenta con un perfil técnico que puede seguir desarrollando y tomar la responsabilidad del mantenimiento del sistema.

3. La base de requerimientos es bastante estable, dado que contamos con alto entendimiento de la realidad y la necesidad del cliente, lo cual reduce la posibilidad de que estos cambien de forma radical, mitigando el riesgo de realizar grandes modificaciones sobre la marcha del proyecto, de todas formas, existe la posibilidad de que se hagan refinamientos a lo largo del proyecto.

En base a estos puntos, entendemos que el ciclo de vida que mejor se adapta a la ejecución del proyecto es el iterativo incremental, con el objetivo de definir iteraciones para separar la etapa de desarrollo, y en base a estas iteraciones generar los entregables para validar con el cliente y seguir adaptándonos a sus necesidades.

### 5.2.2 Metodología Scrum

Considerando el ciclo de vida elegido, las características y necesidades del proyecto que fueron mencionadas anteriormente, así como el conocimiento por parte de los integrantes del equipo con relación a metodologías ágiles, entendemos que la mejor opción para la ejecución de este proyecto es utilizar el marco de trabajo propuesto por *Scrum*.

*Scrum* pertenece a la rama de las metodologías ágiles y es un *framework* para el manejo de proyectos que tienen como fin el desarrollo de productos complejos. Es una metodología que encaja perfectamente en grupos con un número reducido de integrantes y, además, es ideal para aquellos grupos que precisan de una comunicación directa y de constantes validaciones del producto. El funcionamiento de esta metodología es muy sencillo: se definen *sprints* o iteraciones. Antes de iniciar un *sprint* se determinan las funcionalidades o entregables a realizar en el mismo y, luego de finalizado se llevan a cabo instancias de validación y evaluación del producto y proceso. Este ciclo se repite hasta que se logra satisfacer la necesidad del cliente.

## SCRUM PROCESS

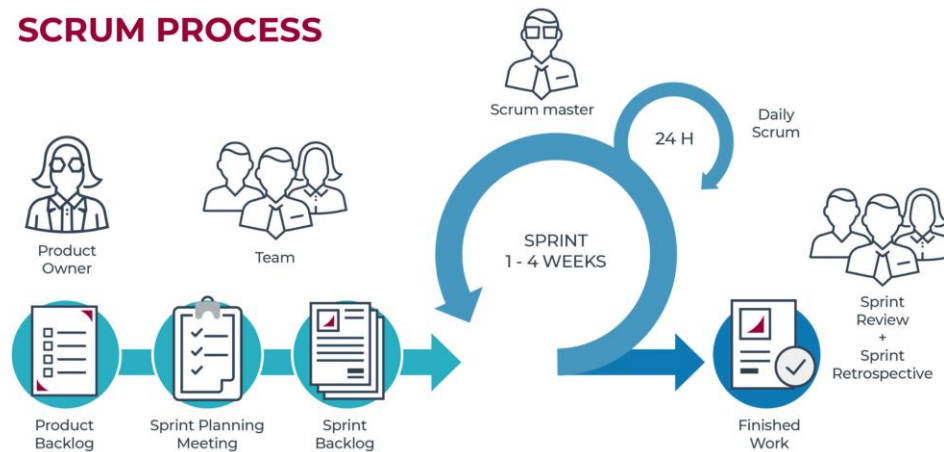


Ilustración 46: Proceso de Scrum

En la Ilustración 45 se pueden apreciar los artefactos y ceremonias que describe Scrum. Estos son el *Product Backlog*, el *Sprint Backlog*, la *Sprint Planning Meeting*, la *Daily Scrum Meeting*, la *Sprint Review* y la *Retrospective*.

En cuanto a los roles *Scrum* define tres:

- **Product Owner:** Es quien debe conocer todos los requerimientos del cliente y, entre otras cosas, debe validar que lo realizado al finalizar cada *sprint* cumpla con lo definido al principio.
- **Scrum Master:** Es quien se encarga de resolver cualquier problema que surja en el equipo. Sirve de facilitador, debe asegurarse que el equipo esté motivado para lograr los objetivos.
- **Team:** Es un grupo multidisciplinario, auto gestionado que la función principal es planificar y confeccionar el producto entregar.

El equipo utilizó la metodología *Scrum*, pero con algunas modificaciones en cuanto a las ceremonias celebradas. Debido a que la mayor cantidad de horas de esfuerzo del equipo se centran en los fines de semana, se omitió la utilización de las ceremonias de reuniones diarias o como se las conoce *daily meeting*, llevando a cabo en su lugar, una reunión semanal, "*Weekly meeting*", para mantenernos actualizadas de los avances realizados en base a lo planificado en cada *sprint*.

### 5.2.2.1 Ceremonias

Durante el desarrollo del proyecto, el equipo participó en diversas ceremonias provistas por Scrum que ayudan a los interesados a conocer el avance, brindándoles transparencia y confiabilidad del producto que se les va a entregar. Además, dichas ceremonias también sirvieron para identificar desvíos en la planificación y como resultado de eso, generar los planes de acción adecuados para solucionar la situación. Siempre se buscó encontrar la mejora continua no solo para el desarrollo del producto final sino también para el proceso de trabajo dentro del equipo. A continuación se detallan las diferentes ceremonias y su contenido:

- **Sprint Planning**: se realiza al comienzo de cada *sprint* con el fin de definir el objetivo y alcance del mismo. Se definen qué *user stories* se estarán tomando, su esfuerzo y responsable a llevarlas a cabo.
- **Sprint Review**: se realiza al culminar el *sprint*, donde el equipo se reúne con el *product owner* y valida las *user stories* hechas en ese *sprint*. En caso de que sea necesario el cliente participa de esta reunión con el fin de obtener su retroalimentación y validación. Entendemos que para todos los *sprints* no es necesaria la participación activa del cliente, ya que en algunos casos pueden ser *sprints* de gestión interna o investigación de alguna tecnología o fase dentro del proyecto.
- **Sprint Retrospective**: al igual que la anterior, se realiza al culminar el *sprint*, el equipo analiza qué cosas estuvieron bien en él, qué problemas tuvimos que enfrentar y cómo lo hicimos, qué cosas podemos mejorar y qué acciones podemos tomar al respecto. Todos estos puntos con el fin de mejorar el proceso dentro del equipo y que la forma de trabajar sea a gusto de todas las integrantes del mismo. En el [Anexo 9 - Sprint Retrospectives](#) están disponibles los resultados de las distintas retrospectivas realizadas durante el proyecto.
- **Weekly Scrum Meeting**: como se menciona anteriormente, en vez de realizar una *daily meeting*, se optó por una reunión semanal, debido a que se adecuaba mejor al contexto de nuestro proyecto. Semanalmente el equipo se reunía por Google Meets para informar sobre avances en los que se estuvo trabajando.

- **Backlog Grooming:** Esta ceremonia siempre se realizó antes de comenzar con el *Sprint Planning Meeting*. De esta forma se preparaba y arreglaba el *backlog*, agregando nuevas *user stories*, revisando estimaciones, y otros.

#### 5.2.2.2 Roles

Cada miembro del equipo cuenta con diferentes niveles de experiencia y capacidades. Teniendo en cuenta las características de cada integrante, así como alguna preferencia personal, se le asignó un rol principal y uno secundario a cada uno. De esta forma se tiene un apoyo en cada área, en caso que el encargado principal estuviera ausente por cualquier motivo.

Se definieron 4 roles de trabajo en el proyecto: *Technical Leader*, *Scrum Master*, *Product Owner* y *Developer*. Dentro de estos, solo los primeros tres se asignaron de forma individual, mientras que el rol de *Developer* fue ocupado por todas las integrantes.

La motivación principal por la cual se asignaron roles fue para mantener un orden dentro del equipo y así centralizar la información u organización de las distintas áreas. Un integrante del equipo que tiene un rol sobre un área es el principal encargado de mantener lo relacionado a ese rol en particular, sin embargo, no será la única responsable pero debe estar al tanto constantemente de forma de poder dividir las tareas adecuadamente. Se optó por definir un segundo responsable por área para contemplar las posibles ausencias de algún integrante.

Rol	Principal	Secundario
Scrum Master	Marcela	Leticia
Product Owner	Marcela	Romina
Technical Leader	Romina	Leticia
Developer	Leticia	Romina

Tabla 4: Roles de las integrantes del proyecto

Cabe destacar que todas las integrantes tienen igual peso de decisión en las áreas definidas. Muchas de las decisiones tomadas son hechas en grupo, más allá del rol asignado a cada integrante. La idea principal es que todas puedan colaborar en el resto de las áreas, ya que es un proyecto a realizar en conjunto.

#### 5.2.2.3 Sprints

Los *sprints* tuvieron una duración corta de dos semanas, ya que era necesario que el cliente validara de forma rápida los resultados. Además el equipo se vio beneficiado ya que se lograba ver de forma temprana el avance de las funcionalidades y si estaban cumpliendo el objetivo final. Cabe destacar que al comienzo del proyecto se definió el Sprint 0 de planificación de una duración de 6 semanas con foco principal en la planificación e investigación, y por otro lado el Sprint 10 tuvo una duración de 10 días, dedicado principalmente a terminar la documentación.

#### 5.2.2.4 Artefactos

A continuación se detallan los distintos artefactos que se utilizaron a lo largo del proyecto:

- **Product Backlog**: es un listado conformado por todas las *user stories* a realizar que describen los requerimientos del producto final. A partir de este *backlog* es que se realiza la planificación de todos los *sprints*.
- **Sprint Backlog**: son el conjunto de *user stories* a realizar en un *sprint* dado. Esto nos permite tener una visión clara de las tareas que se estarán trabajando en las siguientes dos semanas que conforman el *sprint*. Dicho *backlog* será inicializado en la ceremonia *Sprint Planning* que se mencionará más adelante.
- **Incremento**: Es un conjunto de elementos del *Product Backlog* completados en un *sprint* lo cual implica un avance hacia el objetivo del producto final.

#### 5.2.3 Metodología BDD

Se utilizó la metodología *Behavior Driven Development* para parte del proceso de desarrollo, esta metodología permite a todos los implicados, no solo el equipo de desarrollo, entender el proceso desarrollado y el contenido del código fuente.

Se decidió implementar solo en algunas de las funcionalidades principales del sistema debido al esfuerzo que requiere y el tamaño del equipo ya que gran parte de nuestro tiempo tuvo que ser destinado a la capacitación de las tecnologías como se explicará más adelante. Para el resto de las funcionalidades se implementaron y probaron a través de pruebas unitarias y un gran conjunto de casos de pruebas funcionales que fueron repetidos por todas las integrantes del equipo con el fin de no tener una visión sesgada del resultado de las pruebas.

Para mantener un único flujo de trabajo, se optó por incorporar el ciclo de BDD en el contexto de Scrum como se puede apreciar a continuación:

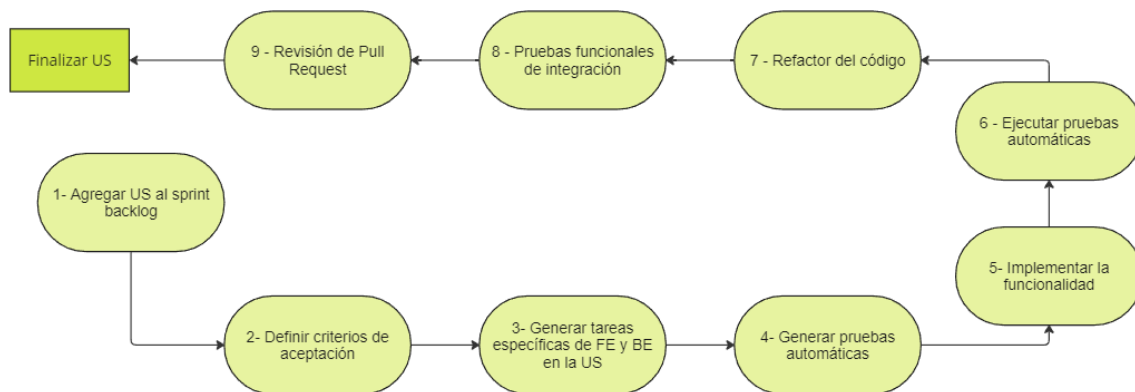


Ilustración 47: Ciclo de trabajo metodolgia BDD

La forma en la que se procedió fue la siguiente, se comenzaba cada *sprint* con todas las *user stories* a ser desarrolladas en un *sprint backlog* (**paso 1**), se revisaban o definían en caso de que fuese necesario nuevos escenarios para la narrativa de la *user story* (**paso 2**). Con estos escenarios definidos, separamos la *user story* en tareas según sus componentes (*frontend* y *backend*) (**paso 3**) para que puedan ser trabajadas en paralelo por distintas integrantes del equipo y se procede a implementar las pruebas para los escenarios en Gherkin (**paso 4**).

Una vez que las pruebas de BDD estaban implementadas, procedemos a implementar el código de la aplicación (**paso 5**) y a utilizar las pruebas anteriormente escritas para realizar la etapa de testing (**paso 6**). Luego de probado el componente, procedíamos con la etapa de refactoro, en caso de que fuese necesario (**paso 7**), donde se pulía el código, siguiendo los estándares definidos y otras prácticas en [8](#)

[Gestión de la calidad](#). Luego se realizan pruebas de funcionales manualmente (**paso 8**) con el fin de tener aún más seguridad del correcto funcionamiento de la feature implementada. Por último, pero no menos importante, se crea un Pull Request (**paso 9**) en GitHub para ser revisado por el otro integrante. Generalmente tendremos como mínimo dos *Pull Requests* por *user story* siendo estas una en el proyecto de *frontend* y la otra en *backend*.

Una vez que todos los *Pull Requests* quedaban mergeados a **main**, la *user story* se pasaba a *Done* y se consideraba el trabajo como completado.

El haber fusionado el ciclo de BDD con *Scrum* nos permitió generar confianza en el desarrollo de nuestras funcionalidades generando resultados de calidad sin tener que hacer mucho re trabajo sobre las mismas, sin embargo, esto nos generó que el ciclo de cada *user story* fuera más largo, por lo cual decidimos aplicar la técnica *One Piece Flow* donde cada integrante solamente podía tener una única tarea en progreso en el *sprint*. De esta manera nos aseguramos de no estar trabajando en simultáneo en varias tareas a la vez y poder poner el foco correspondiente en la tarea que debemos.

#### 5.2.3.1 Desarrollo de funcionalidades utilizando BDD

Para ciertas funcionalidades claves del producto, tanto el frontend como el backend fueron desarrollados utilizando la metodología BDD en el contexto de Scrum. Tomamos como base los criterios de aceptación de cada una de las *user stories* que fueron especificados con Gherkin que nos sirvieron para escribir las *features* y *step definitions* que fueron ejecutados en la herramienta Cucumber JS [14].

A modo de definición, "*features*" son las descripciones legibles por humanos de los escenarios de prueba, mientras que los "*step definitions*" son las implementaciones técnicas que hacen que esos escenarios se ejecuten y se prueben de manera efectiva. Los "*step definitions*" traducen el lenguaje natural de las "*features*" en acciones y verificaciones concretas.

Estas pruebas automatizadas nos sirvieron para reforzar nuestros criterios de aceptación y tener evidencia de las pruebas que posteriormente se repitieron de forma manual y fueron detallados en los casos de prueba funcionales. Ver [Anexo 5 - Casos de pruebas funcionales](#).

A continuación se deja un ejemplo de las funcionalidades implementadas utilizando BDD acompañado de feature definida y sus step definitions.

#### 5.2.3.1.1 User Story

<b>Título</b>	REF1 - Inicio de sesión de usuario
<b>Narrativa</b>	<p><b>Como</b> usuario registrado en el sistema</p> <p><b>Quiero</b> poder iniciar sesión con mi email y contraseña al sistema registrado</p> <p><b>Para</b> hacer uso de las funcionalidades que el mismo provee.</p>
<b>Criterios de Aceptación</b>	<p><b>Dado</b> un usuario no identificado,  <b>Y</b> el usuario está en la pantalla de iniciar sesión,  <b>Y</b> agrega un Email válido en el campo de email,  <b>Y</b> una Contraseña correcta en el campo de contraseña,  <b>Cuando</b> acepta el ingreso de datos (aceptar)  <b>Entonces</b> el sistema despliega la pantalla de inicio.</p> <p><b>Dado</b> un usuario no identificado,  <b>Y</b> el usuario está en la pantalla de iniciar sesión,  <b>Y</b> agrega un Email no válido en el campo de email,  <b>Y</b> una Contraseña correcta o no en el campo de contraseña,  <b>Cuando</b> acepta el ingreso de datos (aceptar)  <b>Entonces</b> el sistema muestra un error.</p> <p><b>Dado</b> un usuario no identificado,  <b>Y</b> el usuario está en la pantalla de iniciar sesión,  <b>E</b> ingresa un Email válido en el campo de email,  <b>Y</b> una Contraseña incorrecta para el email provisto en el campo de contraseña,  <b>Cuando</b> acepta el ingreso de datos (aceptar)  <b>Entonces</b> el sistema muestra un error.</p>

Tabla 5: User Story de REF1 - Inicio de sesión de usuario

### 5.2.3.1.2 Feature

```
features > login.feature > ...
1 Feature: Inicio de sesión de usuario
2   Scenario: Inicio de sesión exitoso
3     Given un usuario no identificado
4     And el usuario está en la pantalla de iniciar sesión
5     And agrega un email válido en el campo email
6     And una contraseña correcta en el campo contraseña
7     When acepta el ingreso de datos correctos
8     Then el sistema despliega la pantalla de inicio
9
10    Scenario: Ingreso de email no registrado
11     Given un usuario no identificado
12     And el usuario está en la pantalla de iniciar sesión
13     And agrega un email no válido en el campo email
14     And una contraseña correcta en el campo contraseña
15     When acepta el ingreso de datos con email no válido
16     Then el sistema despliega un mensaje de error sobre el email incorrecto
17
18    Scenario: Ingreso de contraseña incorrecta
19     Given un usuario no identificado
20     And el usuario está en la pantalla de iniciar sesión
21     And agrega un email válido en el campo email
22     And una contraseña incorrecta para el email provisto en el campo contraseña
23     When acepta el ingreso de datos con contraseña incorrecta
24     Then el sistema despliega un mensaje de error sobre la contraseña incorrecta
```

Ilustración 48: CucumberJS - Feature

### 5.2.3.1.3 Step definitions

Se deja detalle de parte de los *steps definitions* generados para ejecutar las pruebas automáticas.

```
s > step_definitions > login.js > ...
const { Before, Given, When, Then } = require("@cucumber/cucumber");
const assert = require("chai").assert;

const AuthController = require("../src/controllers/auth.controller");
const AuthService = require("../src/services/auth.service");
const UserRepository = require("../src/repositories/user.repository");
const LogRepository = require("../src/repositories/log.repository");

let authController;
let authService;
let userRepository;
let logRepository;

Before(function () {
  logRepository = new LogRepository();
  userRepository = new UserRepository(logRepository);
  authService = new AuthService(userRepository, logRepository);
  authController = new AuthController(authService);
});

Given("un usuario no identificado", function () {
});

Given("el usuario está en la pantalla de iniciar sesión", function () {
  return "OK";
});

Given("agrega un email válido en el campo email", function () {
  return "email_valido@gmail.com";
});
```

Ilustración 49: CucumberJS - Steps definition 1

```

// Scenario 1: Inicio de sesión exitoso
When("acepta el ingreso de datos correctos", { timeout: 100000 }, async function () {
  const email = "email_valido@gmail.com";
  const password = "password_valida";

  const request = {
    body: {
      email,
      password,
    },
  };

  const response = {
    status: (statusCode) => {
      return {
        json: (data) => (response.data = data),
      };
    },
  };

  await authController.login(request, response);

  assert.equal(response.data.hasError, false);
});

Then("el sistema despliega la pantalla de inicio", function () {
  return "SUCCESS";
});

```

Ilustración 50: CucumberJS - Steps definition 2

```

When("acepta el ingreso de datos con contraseña incorrecta", { timeout: 100000 }, async function () {
  const email = "email_valido@gmail.com";
  const password = "contrasena_incorrecta";

  const request = {
    body: {
      email,
      password,
    },
  };

  const response = {
    status: (statusCode) => {
      return {
        json: (data) => (response.data = data),
      };
    },
  };

  await authController.login(request, response);

  assert.equal(response.data.hasError, true);
  assert.equal(response.data.message, "Contraseña incorrecta.");
});

Then("el sistema despliega un mensaje de error sobre el email incorrecto", function () {
  return "El email no es valido.";
});

Then("el sistema despliega un mensaje de error sobre la contraseña incorrecta", function () {
  return "Contraseña incorrecta.";
});

```

Ilustración 51: CucumberJS - Steps definition 3

#### 5.2.3.1.4 Resultado de la ejecución de las pruebas

Como se puede apreciar en la siguiente imagen, los casos de prueba corrieron exitosamente.

```
PS C:\Users\Leti\Desktop\Proyecto\wareupAPI> npx cucumber-js
.....Connection has been established successfully.
.....

3 scenarios (3 passed)
18 steps (18 passed)
0m15.372s (executing steps: 0m15.338s)
```

Ilustración 52: CucumberJS - Ejecución de pruebas

### 5.3 Herramientas de Gestión

Para poder ejecutar un proyecto de forma adecuada es necesario poder contar, entre otras cosas, con una herramienta de gestión que permita el seguimiento y monitoreo de las tareas. Es por esto que el equipo analizó diferentes herramientas en base a la experiencia de los miembros del equipo y se definieron las siguientes a ser utilizadas:

- **Jira:** herramienta que se adapta a la metodología Scrum, ya que permite planificar y ejecutar sprints, confeccionar el product backlog con historias de usuario e ingresar incidentes. Tiene un *sprint* board con diferentes columnas que indican el estado de la tarea o historia de usuario.
- **Google Drive:** se crearon documentos como las minutas de reunión enviadas al cliente, se llevaron también registros de notas generales, se utilizaron hojas de cálculo para la creación de gráficas. Se seleccionó esta herramienta por ser conocida por todos los integrantes, permite trabajar de forma simultánea y ofrece facilidades para compartir documentos.
- **Clockify:** con esta herramienta se llevó adelante el registro de horas. Permite tener un usuario cada una bajo un mismo proyecto. Se seleccionó ya que es fácil de usar, no requiere de mayores esfuerzos y permite la visualización gráfica del registro de Horas según diferentes categorías. En el [Anexo 14 - Reportes Clockify - Gestión de horas](#), se puede observar el reporte final que

permitió extraer esta herramienta y también se muestra un detalle de cómo se registraban las horas con su descripción y categoría correspondiente.

- **Reetro.io:** con esta herramienta se llevaron adelante las reuniones de retrospectiva de los sprints. Esta herramienta ofrece diferentes templates para llevar adelante la reunión invitándonos a hacernos preguntas como: ¿Qué salió bien? ¿Qué no se puede repetir? ¿Qué se puede mejorar? En el [Anexo 9 - Sprint Retrospectives](#), se pueden observar capturas utilizando la herramienta.
- Para la **comunicación interna** del equipo se optó por utilizar principalmente el servicio de mensajería instantánea WhatsApp ya que es un canal inmediato y nos permite una comunicación efectiva, en línea, además de ser confiable.
- Para la **comunicación con el tutor** se utilizó tanto la casilla de correo como Microsoft Teams, para coordinar y realizar reuniones o intercambiar documentación. Ver [Anexo 7 - Reuniones con el tutor](#).
- La **comunicación con el cliente** fue a través de correo electrónico, para evacuar dudas puntuales y conversaciones más informales se utilizó Whatsapp, las reuniones coordinadas se realizaron presencialmente y/o a través de *Google Meet*. Ver [Anexo 8 - Reuniones con el cliente](#).

## 5.4 Planificación y cronograma

En esta sección se explica la forma en la que se llevó a cabo la planificación y el seguimiento de los *sprints* a lo largo del proyecto.

Una vez que nos fue asignado oficialmente el proyecto de WareUP, el equipo evaluó el tiempo restante del que se disponía hasta la fecha de la entrega final, tomando como inicio el 24 de Abril y finalización el 19 de octubre se contabilizaron 26 semanas en total. Por otro lado se estimó que cada integrante le dedique semanalmente un promedio de 16 horas. Resultando en el siguiente análisis:

Horas disponibles	Por integrante	Equipo completo
Semanales	16	48
Proyecto (26 semanas)	416	<b>1248</b>

Tabla 6: Horas del proyecto

A partir de la **Tabla 6** concluimos que se cuenta con una disponibilidad aproximada de 1248 horas para completar la totalidad del proyecto.

Sin embargo, se tomará un total de **1240** horas para realizar una planificación inicial del proyecto dejando 8 horas como posible margen de error inicial. Es importante resaltar que la estimación de la distribución de horas se hizo en base a analogía de otros proyectos similares y bajo juicio de experto.

Para la planificación del proyecto y con el objetivo de lograr obtener luego información valiosa se decidió agrupar las horas dentro de 4 categorías:

- **Investigación y Capacitación:** en esta categoría se espera que se registren todas las horas que se destinan a la investigación del negocio así como horas destinadas tanto a la capacitación de tecnologías como de cualquier otro punto que requiera el proyecto.
- **Desarrollo y Testing:** en esta categoría se registran todas las horas destinadas a tareas de desarrollo, así como horas designadas a tareas testing.
- **Documentación:** las horas que se dediquen a tareas de documentación irán bajo esta categoría.
- **Reuniones y Gestión:** en esta categoría se espera que se registren todas las horas que se invirtieron en reuniones, ya sea internas, con el tutor, o con el cliente. También aquellas horas que se designan a tareas de gestión como control de horas o chequeo de correos.

Luego de hacer esta categorización, procedimos a ubicar las **1240** horas estimadas de la siguiente forma:

Categoría	Porcentaje	Horas
Investigación y Capacitación	15%	186
Desarrollo y Testing	50%	620
Documentación	20%	248
Reuniones y Gestión	15%	186

Tabla 7: Horas del proyecto por categoría

Nos basamos principalmente en porcentaje, obteniendo así el resultado en horas.

### Estimación de horas por categoría

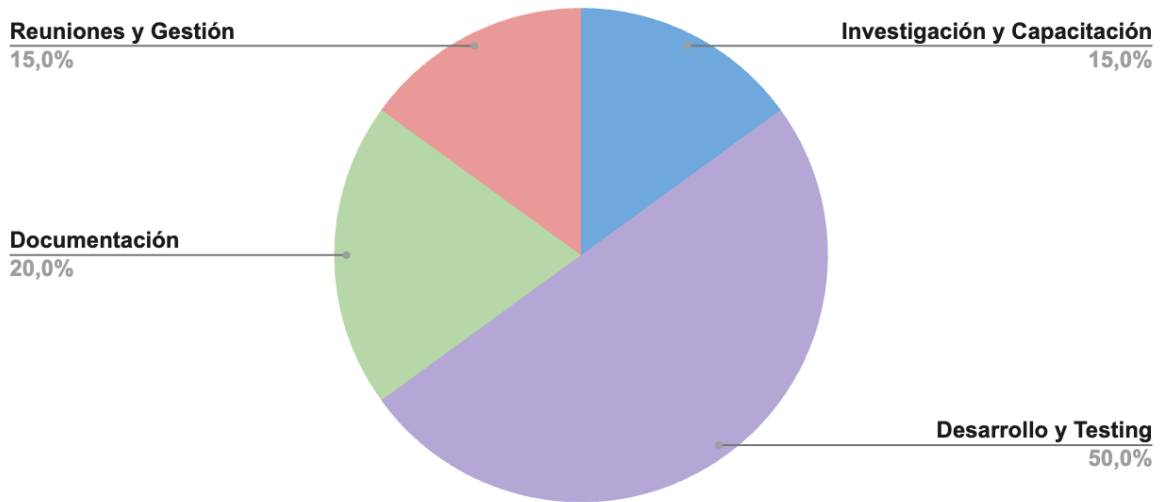
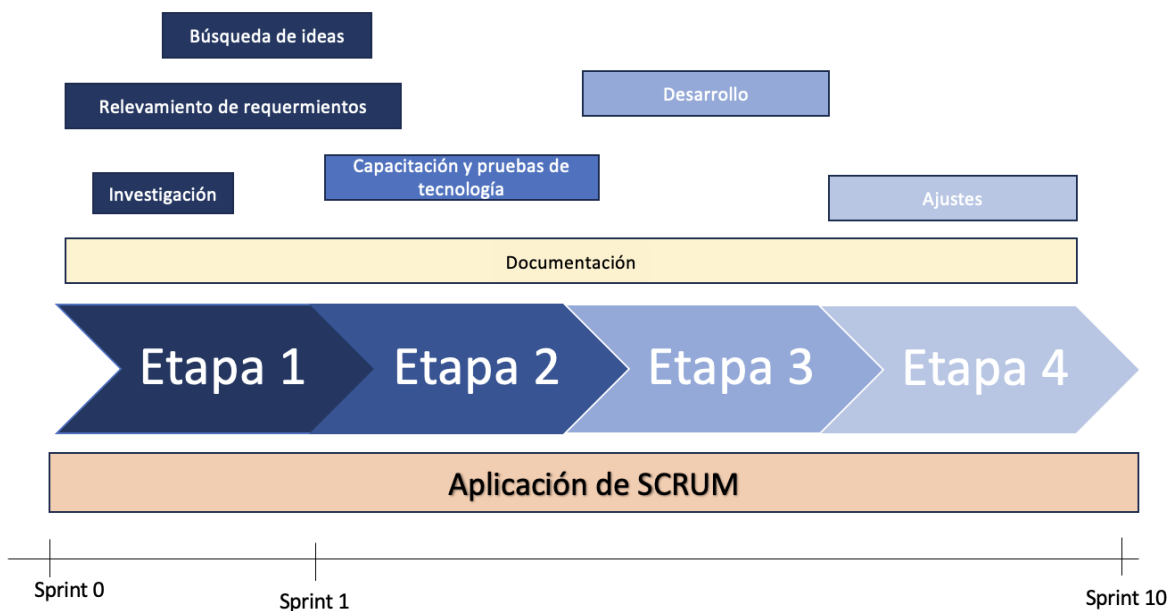


Ilustración 53: Estimación de horas por categoría

En la sección [5.4.3 Resultados obtenidos de las estimaciones iniciales](#) se puede observar un análisis de la comparación entre la estimación inicial y lo finalmente ejecutado.

#### 5.4.1 Estimación de etapas del proyecto



#### Ilustración 54: Etapas del proyecto

A nivel macro el proyecto se dividió en cuatro etapas, que si bien son independientes se solaparon en algún momento.

**Etapa 1 - Investigación:** Se priorizó la definición del proceso, la investigación sobre el negocio, relevamiento de requerimientos, búsqueda de ideas e investigación de tecnologías.

**Etapa 2 - Capacitación y pruebas de tecnologías:** En esta etapa el equipo se capacitó en los lenguajes a ser utilizados, principalmente en *React*, ya que no se contaba con experiencia previa. Se realizó el desarrollo de una funcionalidad puntual a modo de prueba, para luego poder estimar con mayor precisión el restante de funcionalidades así como para reevaluar el alcance en caso de ser necesario para que se ajuste al tiempo disponible. Cabe destacar que si bien se puso mayor énfasis en la capacitación de tecnologías al comienzo del desarrollo, durante todo el proceso de desarrollo se fue aprendiendo acerca de los lenguajes que fueron utilizados.

**Etapa 3 - Desarrollo:** En esta etapa la prioridad fue el cumplimiento de las user stories relativas al desarrollo de la solución.

**Etapa 4 - Ajustes de la solución:** En esta etapa final se puso foco en finalizar el desarrollo, poniendo atención en posibles ajustes que puedan surgir para lograr obtener la solución que se tenía planificada. En paralelo fue en la etapa donde más foco se puso en finalizar la documentación.

**Etapa transversal al proyecto - Documentación:** Con el propósito de no acumular una gran cantidad de trabajo al final del proyecto, es que se fue documentando lo que era posible, y planificando las futuras partes del documento.

#### 5.4.2 Planificación y estimación de iteraciones

Para la planificación de las iteraciones, el equipo utilizó la técnica de *Planning Poker* para tener una estimación inicial en donde primero se presentaron las distintas *user stories* y se discutieron brevemente las tareas necesarias a llevar a cabo para cada implementación.

Basándose en lo discutido, cada integrante asignó el tamaño que consideraba en base a la experiencia en proyectos anteriores, teniendo en cuenta la escala de Fibonacci (1, 3, 5, 8, 13) tomando como referencia superior el valor 13, reservado para aquellas tareas que se deberían analizar con mayor profundidad, en caso de que una user story supere el valor 13, se consideraba una épica. En caso de que todos coincidan con el puntaje seleccionado, se le asigna el mismo a la tarea. En caso contrario, se discuten nuevamente las razones por las cuales se seleccionó ese puntaje hasta llegar a un acuerdo.

SP	¿Dificultad?	¿Tamaño?	Referencia
1	Baja	Pequeño	Modificar un mensaje de texto o pantalla
3	Baja	Medio	Listados básicos
5	Media	Pequeño	Integración con Sendgrid
8	Media	Medio	ABM de usuarios
13	Alta	Grande	Filtros de depósitos

Tabla 8: Escala de Fibonacci

Siguiendo los lineamientos de Scrum, al final de cada *sprint* se realizaron las ceremonias *sprint review* y *sprint retrospective*. Estas se llevaron a cabo previo al comienzo del siguiente *sprint*. En estos encuentros de equipo, para cada user story del *sprint backlog* se evalúa si fueron o no satisfechos los criterios de aceptación, se actualiza y se vuelve a estimar el *product backlog* para el siguiente *sprint*. Luego se procede a planificar el siguiente *sprint* comenzando con el *Backlog Grooming* y finalizando con la *sprint planning meeting*, teniendo en cuenta también la velocidad del equipo.

Al comienzo del desarrollo no se tenía una velocidad de *story points* calculada para el equipo, es por esto que los primeros dos *sprints* sirvieron para tomar un promedio de puntos realizados, resultando así en una velocidad de 42 puntos de desarrollo por *sprint*. En la sección [5.4.3 Análisis de resultados obtenidos](#) se detalla sobre los resultados obtenidos.

#### 5.4.2.1 Gestión del Product Backlog

Como mencionamos anteriormente, los sprints comenzaron con el *Backlog Grooming*. Durante el mismo se realiza una revisión del *product backlog*, eliminando *user stories* obsoletas, agregando nuevas si corresponde y volviendo a priorizar las existentes. Los criterios de priorización dependen de la etapa según se encontraba el proyecto y según lo que se conversaba con el cliente.

El *product backlog* fue compuesto por *user stories* relacionadas a la documentación y al desarrollo de las funcionalidades. Al inicio del proyecto se tuvieron principalmente historias referentes a la documentación e investigación. A medida que el proyecto fue avanzando se incorporaron historias referentes al desarrollo de funcionalidades, utilizando el formato *user story* que se describe en el [Capítulo 3 Ingeniería de Requerimientos](#). Se pueden ver las capturas del *product backlog* en el [Anexo 4 - Capturas de Artefactos de Scrum](#).

#### 5.4.2.2 Gestión del Sprint Backlog

En esta sección se describe detalladamente cómo fue la elaboración y el mantenimiento del *sprint backlog* durante la ejecución del proyecto. Se pueden ver las capturas de algunos *sprints backlogs* en el [Anexo 4 - Capturas de Artefactos de Scrum](#).

##### 5.4.2.2.1 Elaboración de un nuevo Sprint Backlog

Para la creación del *sprint backlog* del próximo *sprint*, se tomaron las *user stories* de mayor prioridad, evaluando que la suma del tamaño de todas las *user stories* seleccionadas fueran cercano a la velocidad del equipo por *sprint* (42). Fue importante también considerar el número de horas disponibles para el *sprint*, debido a que en ciertas etapas del proyecto se dispuso de más tiempo, y en otras la disponibilidad fue menor.

##### 5.4.2.2.2 Tareas del Sprint Backlog

Cada *user story* que formaba parte del *sprint backlog* fue desglosado en tareas.

Al comienzo la estimación de esfuerzo de cada tarea resultó difícil ya que el proyecto se encontraba en una etapa temprana y no se tenían métricas para comparar, luego

de identificar mejor el esfuerzo requerido por user story se pudo realizar una estimación más precisa.

Para poder realizar un seguimiento preciso del avance de las tareas durante el proyecto se hizo uso de un Scrum Board mediante la herramienta Jira.

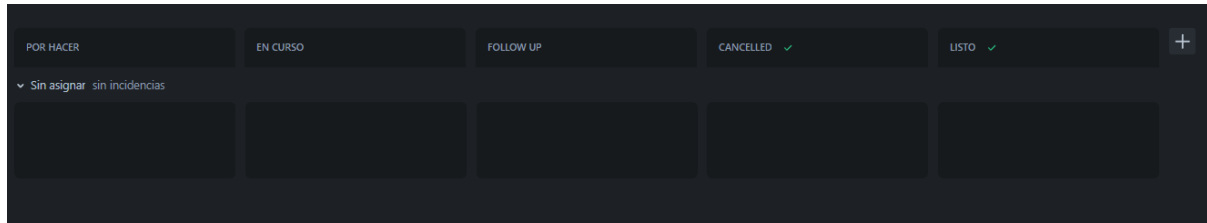


Ilustración 55: Scrum Board utilizado en Jira

Los estados utilizados fueron:

- **To Do:** La tarea no se ha comenzado.
- **In Progress:** La tarea está en proceso de ejecución.
- **Follow up/Blocked:** La tarea está bloqueada por otra o se requiere que otro integrante realice code review cruzado.
- **Done:** La tarea ha sido terminada.
- **Cancelled:** La tarea se cancela.

Se actualizó manualmente el estado de las user stories a Done solo cuando cumplían los criterios de aceptación correspondientes.

### 5.4.3 Resultados obtenidos de las estimaciones iniciales

#### 5.4.3.1 Cronograma detallado

Etapa/Evento	Actividades principales	Fecha de inicio	Fecha de fin
Sprint 0	Definiciones iniciales y requerimientos	24/04/23	04/06/23
Sprint 1	Definiciones de arquitectura y base de datos. Desarrollo	05/06/23	18/06/23
Sprint 2	Desarrollo y testing	19/06/23	02/07/23
Sprint 3	Desarrollo y testing	03/07/23	16/07/23

Sprint 4	Desarrollo y testing	17/07/23	30/07/23
Sprint 5	Desarrollo y testing	31/07/23	13/08/23
Release I	Primera versión del producto	14/08/23	14/08/23
Sprint 6	Desarrollo y testing	14/08/23	27/08/23
Sprint 7	Desarrollo y testing	28/08/23	10/09/23
Sprint 8	Documentación, desarrollo y testing en caso de atrasos	11/09/23	24/09/23
Sprint 9	Documentación y testing	25/09/23	08/10/23
Release II	Segunda versión del producto	09/10/23	09/10/23
Sprint 10	Documentación	09/10/23	19/10/23
Cierre del proyecto	Entrega final	19/10/23	19/10/23
Release final	Entrega final MVP	19/10/23	19/10/23

Tabla 9: Cronograma del proyecto

#### 5.4.3.2 Análisis de línea del tiempo

A continuación se muestra una línea del tiempo que abarca los meses previos y posteriores a la asignación oficial del proyecto. El objetivo de esta gráfica es ubicar a lo largo de los meses las diferentes etapas por las que el equipo transcurrió mostrando también los hitos principales.

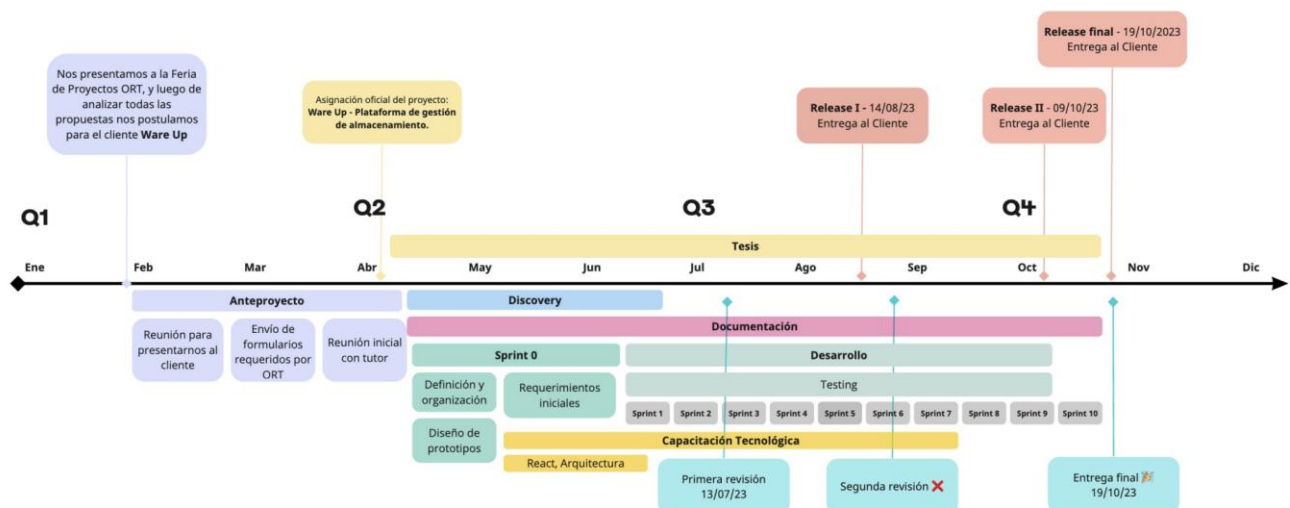


Ilustración 56: Línea del tiempo del proyecto

En febrero se dió comienzo al trabajo de fin de carrera, comenzando en ese mes lo que se conoce como la etapa “Anteproyecto”. Esta etapa consistió principalmente en elegir el proyecto a realizar, asegurarnos tener un tutor asignado y presentar varios formularios solicitados por ORT.

Para la elección del proyecto, el equipo se presentó a la Feria de Proyectos de ORT, escuchando todas las propuestas posibles y luego seleccionando las de nuestro mayor interés, entre ellas se encontró la propuesta de WareUp. Es así como finalmente en Abril nos notifican que el proyecto nos fue asignado oficialmente.

El 24 de Abril, dimos comienzo oficial al proyecto. Previo a detallar las etapas, nos resulta de importancia comentar la situación previa a la planificación. Principalmente lo relacionado al desconocimiento del negocio, que lógicamente esto trajo una etapa de *discovery*.

Ninguna de las integrantes del equipo contaba con experiencia en el rubro de la logística. Debido a esto tuvimos que recurrir a lecturas [15] [16] sobre el tema para tener el contexto necesario y así entender las distintas particularidades del rubro como ayudarnos a definir los intereses del cliente.

A continuación se detallan algunos de los puntos que eran originalmente desconocidos por las integrantes del equipo y eran factores fundamentales para el producto a entregar:

- Distintas características que pueden tener los espacios de almacenamiento
- El término “depósitos” es tan usado como “espacios de almacenamiento”
- La posibilidad de arrendar un cierto porcentaje de metros cúbicos y no todo un depósito en general
- La diferencia de términos como “alquiler” y “arrendamiento”
- Entender que aporta valor extra a los usuarios

A partir de esta información, se comenzó a trabajar en el *sprint 0*, cuyo objetivo fue organizar el proyecto lo máximo que se pueda, definiendo la metodología de trabajo y los roles. También se comenzó a bajar a tierra los requerimientos del sistema, se validaron con el cliente y a partir de esto se comenzó a diseñar posibles prototipos.

Otro punto importante a mencionar en esta etapa, fue la elección de tecnologías a ser utilizadas en el desarrollo. Originalmente las integrantes del equipo propusieron proactivamente al cliente utilizar un set de tecnologías en particular, sin embargo, por preferencia del cliente, se decidió utilizar React para el desarrollo *frontend* y NodeJs para el *backend* ya que el perfil técnico del cliente ya maneja estas tecnologías y así el mantenimiento y continuación del producto no serán un desafío para él. Esta elección de tecnologías le requirió al equipo destinar una gran cantidad de horas a la capacitación de las mismas puesto que no se encontraban dentro de las que más dominaban las integrantes. El comienzo de capacitación se dió en esta etapa de planificación pero se mantuvo casi a lo largo de todo el proyecto teniendo diferentes picos de dedicación de horas según el momento y las funcionalidades que iban siendo atacadas.

Mencionar también que la elección de tecnologías desconocidas, nos obligó a re evaluar el alcance del proyecto, esto en acuerdo con el cliente, ya que se entendió que se debían destinar una mayor cantidad de horas a capacitación por lo que nos vimos obligadas a quitar horas de desarrollo para compensar la carga total. Lo que se quitó del alcance entendemos que son funcionalidades o detalles que si bien es deseable que estén, no son prioritarias y podrían quedar para implementarse en una segunda versión del producto.

Durante esta etapa inicial, la gestión y definición de tareas se iba dando a medida que éstas surgían. La distribución de las tareas entre las integrantes del equipo se daba asignando las que estuvieran disponibles para realizar, buscando que la carga fuese lo más equitativa posible según nuestro criterio y sentido común.

Este *sprint* 0, según experiencias en otros proyecto similares, estimamos que nos podría llevar alrededor de 5 semanas, pero finalmente nos requirió una semana más de trabajo, se decidió concederle este tiempo extra a esta etapa ya que entendíamos era crucial para el correcto entendimiento del proyecto. Entendemos que este tiempo extra que nos requirió fue debido al desconocimiento de las tecnologías por lo que no estábamos 100% preparadas para comenzar con el desarrollo sin antes investigar un poco más en ciertos aspectos principalmente del lenguaje React. Aprovechamos el tiempo para hacer cursos e implementar funcionalidades simples con el fin de poder

dar una estimación que se pudiera ajustar a la realidad de las funcionalidades que íbamos a implementar para el producto final.

Por otro lado, en paralelo a este *sprint* 0, se dió comienzo a la etapa de Documentación, la cual fue transversal a todo el proyecto. Si bien el foco mayor estuvo sobre el final del proyecto, el equipo desde el comienzo creó un esqueleto de las secciones a desarrollar y siempre y cuando se tuviese la información requerida, se iba documentando, asegurándonos así no acumular una gran carga de trabajo de documentación para el final. Demás está decir, que lo que se iba documentando jamás quedaba intacto, sino que siempre se iba modificando y agregando más información.

A partir del resultado obtenido en esta primera etapa, se intentó generar el resto de los *sprints* para poder estimar el tiempo que llevaría completar el desarrollo. Esto derivó en dividir el cronograma en un total de 10 *sprints*, siendo 9 de ellos de dos semanas de duración cada uno, y la excepción del último *sprint* que tuvo la duración de diez días debido a la fecha final pautada. Este último *sprint* se planificó para que tenga prioridad total la documentación y que no queden tareas de desarrollo. Esto no se cumplió del todo, puesto que para el *sprint* 10 quedó pendiente el desarrollo de la funcionalidad: REF32 - Métricas internas del sistema para usuario backoffice y el trabajo en algunos *bugs* que quedaron de sprints anteriores, de todos modos no generó controversias sobre lo planificado.

Una vez finalizada esta primera etapa de planificación, se dio comienzo al proceso iterativo de análisis, desarrollo y pruebas de las funcionalidades asociadas a cada *sprint* de trabajo.

Los primeros sprints fueron los más costosos para el equipo, ya que aún no nos encontrábamos familiarizados con las tecnologías y aún nos estábamos adaptando a todo lo que implica el flujo y entendimiento de trabajo. Tampoco se tenía claro una velocidad determinada por *story points*, por lo que este dato hubo que generarlo.

A medida que se avanzaba, se iba generando y arraigando el conocimiento, es así que luego de los dos primeros *sprints*, el equipo concluyó que 42 *story points* era la velocidad promedio, con el pasar de los *sprints*, el equipo logró apegarse a la velocidad pautada y generar un buen ritmo de trabajo.

Durante los últimos *sprints* el equipo puso foco en estabilizar el producto y destinar un mayor esfuerzo a la corrección tanto de incidentes como a los aspectos mencionados en el *feedback* del cliente.

En el *sprint* 10 se puso un esfuerzo superior a los anteriores a documentar, se acudió al tutor de manera continua para la correcta validación de las partes del documento. El equipo intentó asegurarse de estar cumpliendo con el formato, las secciones y la información requerida para la completitud de un buen trabajo final.

En cuanto a hitos importantes a lo largo de esta línea del tiempo, podemos mencionar por un lado, las *releases* al cliente, y por otro lado las revisiones de ORT con un tutor externo.

En relación al primer punto, se planificaron tres *releases*, donde se le brindaba al cliente la url de la página para que pruebe y registre *feedback* en una planilla de Google Sheets para luego ser conversado en conjunto e implementar los cambios necesarios. Estas entregas se planificaron para que a la hora de entregar el producto se tengan desarrolladas un conjunto de funcionalidades que aporten valor al negocio.

El primer *release* se hizo al finalizar el *sprint* 5, el segundo al finalizar el *sprint* 9, y por último la entrega final al cliente se planificó para que coincida con la entrega final del proyecto, siendo el 19 de Octubre.

En relación al segundo punto, las revisiones, tuvimos la primera el día 13 de Julio, con presencia de nuestro tutor Gerardo Maturro y un tutor externo, en esta instancia se hizo una presentación del trabajo que se tenía hasta el momento y también del trabajo planificado a futuro, el equipo destaca el *feedback* valioso que se llevó de esta instancia para avanzar con mayor confianza y claridad. Lamentablemente, la segunda instancia de revisión se vió suspendida por ORT a causa de la falta de un tutor externo, siendo esto razones ajenas al equipo.

Finalmente, el 19 de Octubre se dió por finalizado el proyecto, entregando el documento requerido por ORT y los entregables correspondientes al cliente.

### 5.4.3.3 Velocidad durante el proyecto

La velocidad promedio por *sprint* del equipo es una métrica importante en la gestión del proyecto. Indica la cantidad de *story points* que el equipo logra implementar en promedio en cada iteración, esto permite definir con mayor precisión el *sprint backlog*, así como mejorar las estimaciones.

Como se mencionó anteriormente, al comienzo del desarrollo el equipo no sabía cuál era la velocidad promedio, es por eso que los primeros dos sprints sirvieron para recolectar este dato y sacar un promedio, es así que concluimos que **42 *story points*** es la velocidad promedio del equipo. Cabe destacar que esta velocidad refiere a todas las tareas pertenecientes a la categoría de Desarrollo ya que las horas de Documentación se registraron y estimaron en formato de horas.

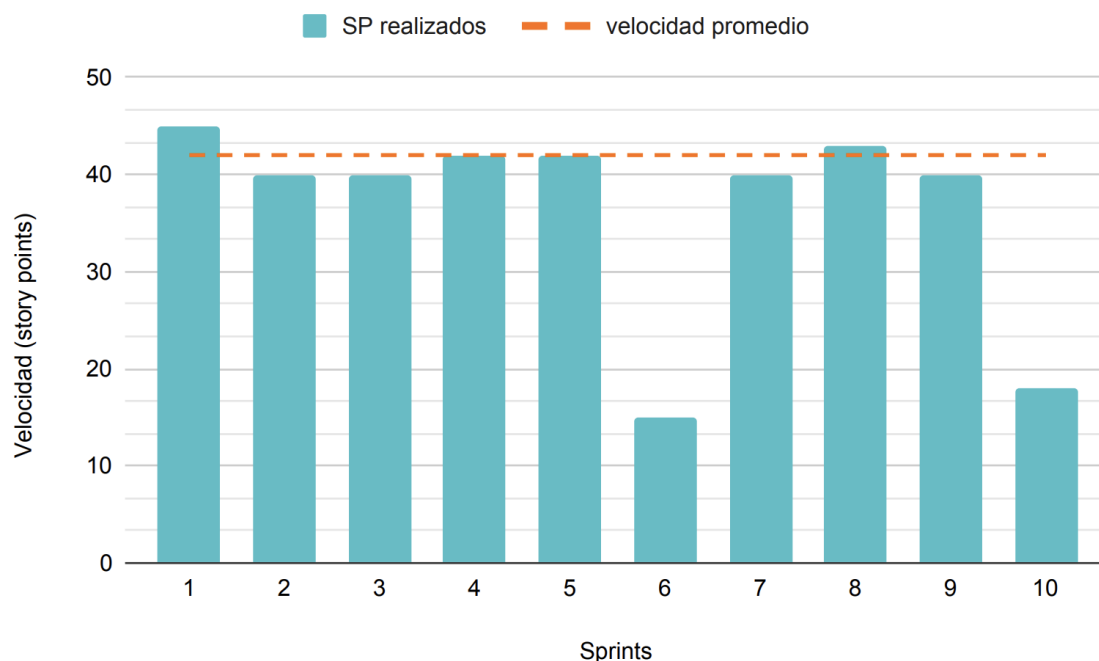


Ilustración 57: sprints velocity

Como se puede apreciar en la gráfica, en el correr de los 10 *sprint* planificados se mantuvo una gran cercanía a la velocidad promedio con la excepción de dos casos notorios en los que no se cumplió que son en el *sprint* 6 y 10.

En el *sprint* 6, ejecutado en Agosto, se observa un descenso importante de la velocidad, esto se dió a causa de una intervención quirúrgicas con una integrante del equipo y nuevo comienzo de semestre en la carrera, provocando que el equipo le dedique menos horas al proyecto. Para compensar esta falta, en el *sprint* 7 se lograron

completar los story points del *sprint* anterior junto con los correspondientes a ese *sprint*.

Por otro lado, en el *sprint* 10, el mayor esfuerzo estuvo dedicado a la documentación del proyecto, es por esto que la velocidad del equipo fue baja, ya que únicamente se implementó una nueva funcionalidad. En este *sprint* el equipo se propuso la posibilidad de implementar ciertos requerimientos que hasta el momento no pudieron ser desarrollados, pero finalmente a causa del tiempo se priorizó la documentación y el correcto cierre del proyecto. La información detallada referente a los requerimientos que fueron desarrollados en cada *sprint* y los hitos más importantes se pueden ver en el [Anexo 10 - Planificación y resultado de los sprints](#).

Como conclusión consideramos que la velocidad promedio pautada al inicio de los sprints resultó acercarse al esfuerzo que el equipo terminó ejecutando en el desarrollo.

#### 5.4.3.4 Sprint Burndown Chart

Las *sprints burndown chart* son gráficas de gran ayuda para ir visualizando y analizando el avance del equipo sobre un mismo *sprint*, pudiendo corroborar si se podría completar el trabajo comprometido a tiempo según el esfuerzo que se viene haciendo. La identificación de un desvío notorio entre los *story points* reales y los ideales puede llevar a que el equipo decida tomar acciones a tiempo para corregir el proceso.

En cuanto a la velocidad del equipo dentro de cada *sprint*, la misma fue evolucionando de forma positiva *sprint* a *sprint*.

A modo de resumir la información, mostraremos las *burndown chart* de dos *sprints*, uno del comienzo del proyecto y otro sobre el final para que se pueda apreciar la mejora que el equipo tuvo en el ritmo de avance del proyecto.

## SPRINT 4

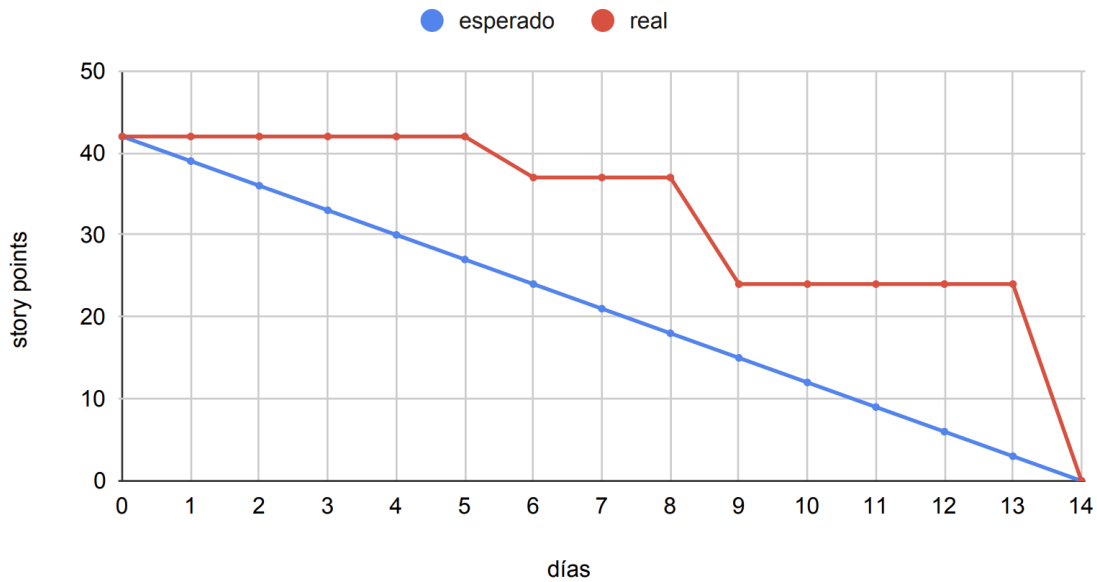


Ilustración 58: sprint 4 - burndown chart

En relación a la *burndown chart* del *sprint 4*, podemos observar que se muestra un escenario conocido como *"The cliff"* [17]. En este *sprint* el equipo no estaba familiarizado del todo con la actualización de los *story points* repercutiendo esto de gran manera en el resultado de la gráfica, impidiendo ver de manera real el avance que se iba teniendo. Se puede ver que hasta el día 5 el equipo de desarrollo parece no haber ejecutado ningún *story points*, cuando probablemente lo sucedido es que no se haya contabilizado como corresponde. El scrum master luego de observar esta problemática puso el tema sobre la mesa en la *sprint retrospective* del equipo logrando que se tome más responsabilidad sobre este asunto.

## SPRINT 8

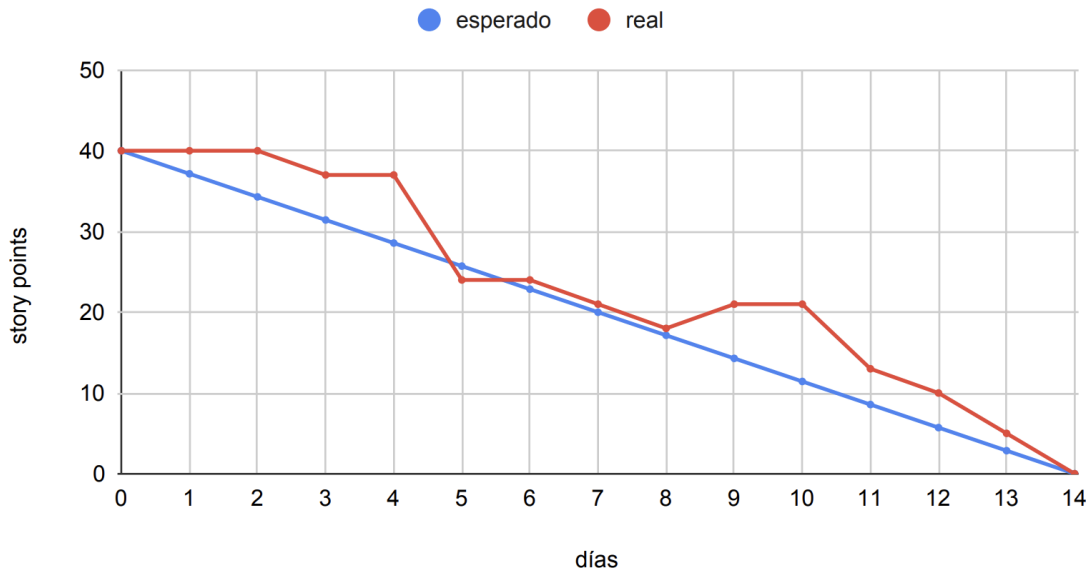


Ilustración 59: sprint 8 - burndown chart

En el caso del *sprint 8*, el escenario es más positivo que la gráfica anterior. Se puede observar que lo real se acerca mucho más a lo esperado. Este caso, se conoce como “*The bubbler*” [17]. En este momento del proyecto el equipo contaba con más experiencia y tenía automatizado el cierre de las user story una vez que se finalizaban aportando a que la gráfica se pueda construir mostrando la realidad. Un punto de gran importancia a destacar, es que en el día 8 de este *sprint* se agregó una nueva user story referente a un *bug*, generando un incremento en los story points estimados inicialmente, es por eso que se puede ver que la línea roja aumenta.

### 5.4.3.5 story points estimados VS story points realizados

Esta gráfica busca reflejar los story points que se estimaron por *sprint* versus los que efectivamente se realizaron.

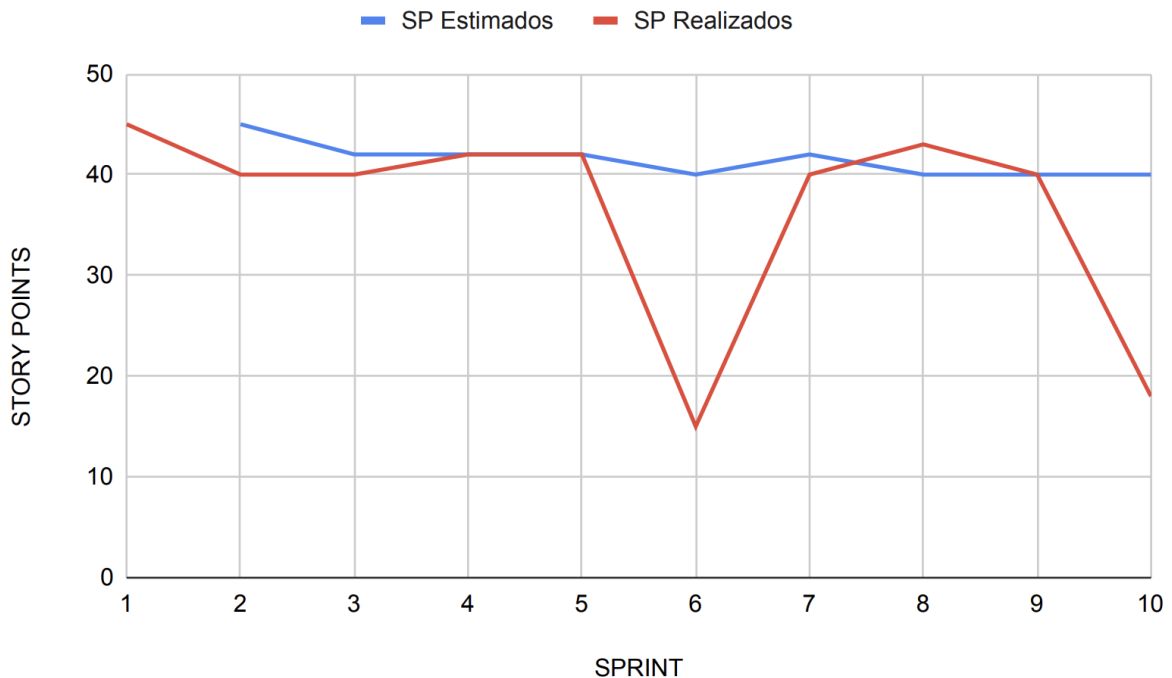


Ilustración 60: sprints story points

Al comienzo del proyecto no se tenía aún información confiable que permita estimar con precisión la cantidad de trabajo que le equipo podía lograr por *sprint*, es por eso que en la primera iteración no se hizo una estimación sino que sirvió para ir recolectando este dato.

Al comienzo de los *sprint* se puede observar que en general la estimación supera lo realizado, esto entendemos que sucedió por nuestra inexperiencia en etapas iniciales, luego se puede ver que a medida que se avanzó en el proyecto se logra una cercanía mayor entre lo que se estimaba y se termina ejecutando.

En el *sprint* 6, se puede ver que el equipo logró realizar mucho menos de lo que se estimó inicialmente, esto sucedió debido a que, como se mencionó anteriormente, fue un *sprint* que presentó complicaciones a niveles personales por temas quirúrgicos y de estudio, generando así que el equipo no pueda completar los story points estipulados. Pasado este *sprint* de trabajo, el equipo logró equiparar nuevamente el esfuerzo de trabajo, inclusive en el *sprint* 8 se observa un aumento en el trabajo realizado sobre lo que se estimó, esto se debió a que se agregaron story points una vez comenzado el *sprint*, ya que se encontraron incidentes que debían ser trabajados en esa iteración.

Otro *sprint* que merece una explicación más detallada es el 10, última iteración de trabajo, donde el equipo estimó 40 *story points* pero finalmente se hicieron 18. La causa de esto fue debido a que se propuso que en caso de que el tiempo fuere suficiente se iba a intentar de implementar requerimientos que no pudieron ser implementados hasta el momento y que tenían prioridad baja, pero finalmente se tuvo que poner mayor esfuerzo en tareas de documentación viéndose el equipo obligado a dejar estos requerimientos para una próxima versión del producto.

Concluimos que el esfuerzo realizado por sobre lo estimado fue mejorando a medida que se fue avanzando en los sprints y que en general se vió un escenario positivo donde el equipo logró en todo momento acercarse siempre al cálculo inicial que se hacía. También destacar que siempre que no se puede cumplir con lo pautado el equipo encontró las justificaciones y era un tema que se abordaba en las *sprint retrospective* para analizar lo sucedido y mejorar a futuro.

#### 5.4.3.6 Registro de horas invertidas

El equipo mantuvo un registro de horas dedicadas al proyecto por parte de cada uno de sus integrantes por medio de la herramienta *Clockify* descrita anteriormente. Cada integrante fue el responsable de registrar las horas correctamente. Fue tarea del *product owner* llevar un control de que las horas se están cargando a medida que avanza el proyecto.

##### 5.4.3.6.1 Distribución por categoría

Como se mencionó anteriormente el registro de horas del proyecto se agrupó en 4 categorías:

- Investigación y Capacitación
- Desarrollo y Testing
- Documentación
- Reuniones y Gestión

## Distribución de horas por categoría

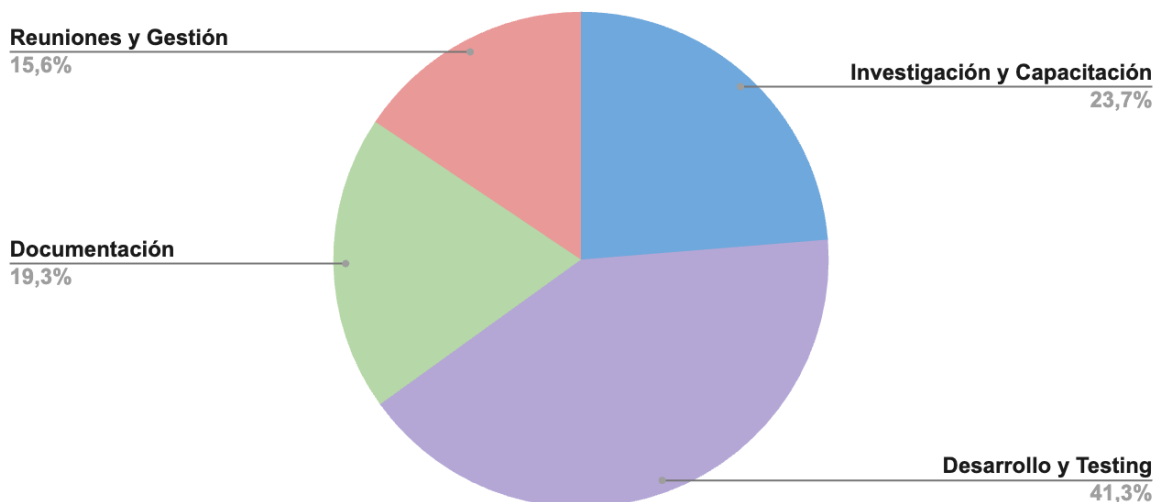


Ilustración 61: Distribución de horas del proyecto

A partir de la gráfica se puede observar que la mayor parte del esfuerzo fue dedicada al desarrollo de la plataforma web, esto incluye tareas de desarrollo de código como también las tareas implicadas a *testing*.

En segundo lugar, las tareas de investigación y capacitación tuvieron un esfuerzo del **23,7%**, es la segunda categoría con más horas y esto tiene sentido, ya que se destinó un gran esfuerzo en la capacitación de las tecnologías y en el entendimiento del negocio.

En tercer lugar, la documentación tuvo un esfuerzo del **19,3%**, ejecutando la mayor cantidad de horas sobre el final del proyecto. Y por último las tareas relacionadas a la gestión y reuniones tanto internas como con el cliente consumieron el **15,6%** de la totalidad de horas.

Si comparamos la estimación inicial con lo finalmente ejecutado podemos notar que se asemeja de gran manera por más que presenta ciertas diferencias.

Categoría	Estimado	Ejecutado	Diferencia
Investigación y Capacitación	15%	23,7%	(+) 8,7%

Desarrollo y Testing	50%	41,3%	(-) 8,7%
Documentación	20%	19,3%	(-) 0,7%
Reuniones y Gestión	20%	15,6%	(+) 0,6%

Tabla 10: Desviación de estimaciones

Una de las mayores diferencias se encuentra en la categoría Investigación y Capacitación, se ejecutó un **8,7%** más de horas de lo estimado, esto es completamente justificable ya que el equipo tuvo que invertir mayor cantidad de horas de lo pensado en la etapa de discovery y principalmente en la capacitación de las tecnologías, en compensación con esta diferencia de horas es que se puede ver que el Desarrollo y Testing requirió un **8,7%** menos de esfuerzo de lo planificado, creemos que al invertir en capacitación luego a la hora de tener que efectivamente implementar el código el equipo pudo desenvolverse de mejor manera.

Con respecto a las categorías Documentación y Reuniones y Gestión, el esfuerzo fue muy cercano a lo estimado inicialmente presentando mínimas diferencias. Concluimos que la estimación inicial fue acertada, y que nuestras experiencias previas en la ejecución de proyectos resultó de gran ayuda finalmente para efectuar estimaciones.

#### 5.4.3.6.2 Horas totales

En la siguiente gráfica se muestran la cantidad de horas hombre trabajadas por el equipo durante todo el proyecto:

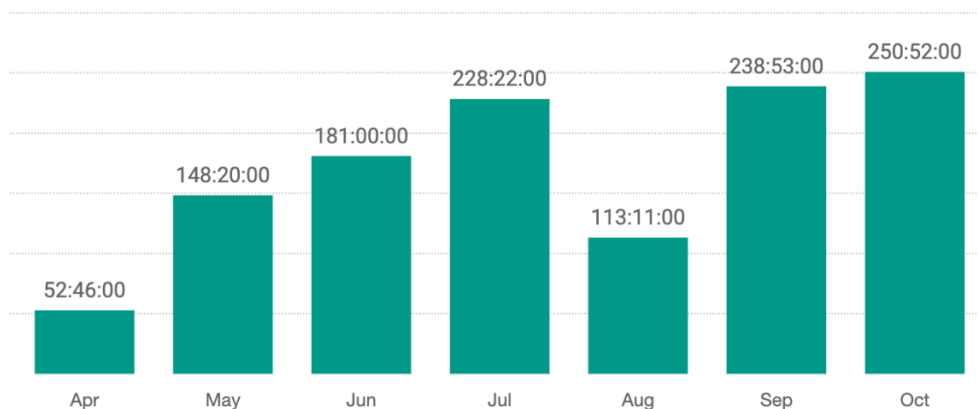


Ilustración 62: Horas mensuales del proyecto

Se le dio comienzo oficial al proyecto el 24 de Abril, es por esto que las horas pertenecientes a este mes son pocas. Luego se puede observar que las horas van en aumento con el correr de los meses, esto es esperable ya que a medida que se fue confeccionando el *backlog* y avanzando en definiciones de gestión del proyecto cada integrante pudo tomar mayor autonomía y dedicarle mayor esfuerzo.

En el mes de Agosto, se puede notar un descenso notorio las horas ejecutadas, esto se dio principalmente por dos hitos principales que sucedieron ese mes, en primer lugar, una de las integrantes se sometió a una operación médica, viéndose obligada a hacer reposo y por otro lado, se dio comienzo al segundo semestre de la carrera donde las tres integrantes comenzaron a cursar al menos una materia. Para compensar esta falta de horas, en el mes de septiembre se hizo un esfuerzo extra y se logró completar el trabajo que se tenía pautado.

En los meses de Julio y Octubre son los que más horas se ejecutaron. En Julio se puso gran foco en el desarrollo web, logrando que el equipo tenga un primer incremento valioso del *MVP*, y por otro lado en Octubre, finalizando la última etapa, se destinó una gran cantidad de horas a la documentación del proyecto.

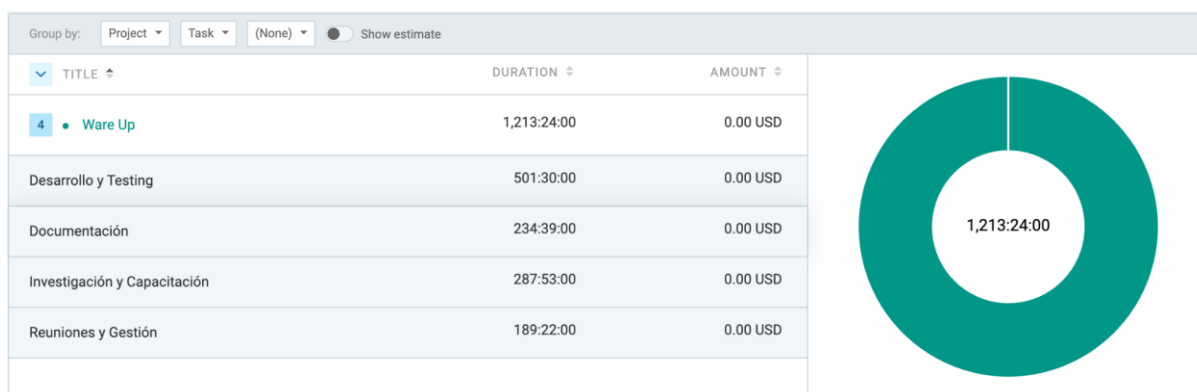


Ilustración 63: Desglose horas del proyecto

Como dato final, concluimos en que el proyecto nos llevó un total de **1213:24 horas**.

# 6 Gestión de Riesgos

## 6.1 Introducción

La gestión de riesgos es un componente esencial en el desarrollo de un proyecto ya que nos permite estar mejor preparados ante esto, mitigando las consecuencias.

Dentro de la gestión de riesgo realizamos actividades de identificación, análisis, evaluación de consecuencias, ponderación e identificación de acciones tanto preventivas como correctivas.

Tener los riesgos identificados, y sobre todo, un plan de acción, reducen el impacto en el proyecto en caso de que ocurran, favoreciendo el éxito en el desarrollo del mismo. En nuestro análisis de riesgos identificamos tanto los que pueden llegar a tener un impacto negativo como positivo.

## 6.2 Planificación de la gestión de los riesgos

La planificación de riesgos implicó definir un plan inicial de gestión de los mismos al comienzo del proyecto, donde se identificaron y evaluaron los principales riesgos detectados, y se definieron mecanismos de respuesta y contingencia. Además, se definió un plan para actualizar las magnitudes de los riesgos.

Por otro lado, se definió una clasificación cualitativa de categorías de riesgo, definiendo una matriz de estrategia para utilizar dependiendo de la categoría en la que un riesgo pertenezca.

### 6.2.1 Clasificación de riesgos

Las categorías de riesgos utilizadas para su clasificación fueron las siguientes:

**Riesgos del cliente:** Dentro de esta categoría englobamos todos los riesgos asociados a WareUP como cliente: la relación, comunicación, disponibilidad, colaboración, exigencias.

**Riesgos técnicos y tecnológicos:** Se evaluó bajo esta categoría las tecnologías seleccionadas para la realización del producto, evaluando nuestro conocimiento sobre ellas, madurez, acceso a la información.

**Riesgos humanos:** Se detallan los riesgos referentes a las personas como recursos en la ejecución del proyecto, como ser, licencias, enfermedades, disponibilidad en general.

**Riesgos de gestión:** Errores en la planificación, desconocimiento de procesos, malas definiciones, errores de requerimientos/especificaciones, error de estimaciones.

**Otros:** Para todos aquellos riesgos que no pueden ser identificados en alguna de las categorías antes mencionadas.

## 6.3 Cuantificación de los riesgos

La cuantificación de los riesgos fue realizada en base a tres unidades de medida: la probabilidad de que ocurran, una escala de impacto en caso de que sucedan, y el tiempo estimado de posible ocurrencia del riesgo.

### 6.3.1 Ocurrencia del riesgo

Indica que tan probable es que un riesgo ocurra. Para su evaluación se seleccionó una escala entre el 0 y 1, con incrementos de 0 a 0,2.

Probabilidad de ocurrencia	Descripción
0	No probable
0,2	Poco probable
0,4	Probable
0,6	Muy probable
0,8	Altamente probable
1	Es un hecho

Tabla 11: Ocurrencia del riesgo

### 6.3.2 Impacto del riesgo

Se evalúa cuál es el impacto en el proyecto en caso de que ocurra el riesgo. La escala manejada fue entre 0 y 5 con incrementos de a 1.

Nivel de impacto	Descripción
0	Ninguno
1	Marginal
2	Poco importante
3	Importante: puede retrasar el proyecto
4	Crítico: puede provocar el fracaso del proyecto
5	Catastrófico: el fracaso del proyecto es un hecho

Tabla 12: Impacto del riesgo

### 6.3.3 Ocurrencia en el tiempo

Se evalúa el tiempo de ocurrencia del riesgo a lo largo de todo el proyecto.

Tiempo de ocurrencia	Descripción
Inmediata	Ocurrencia en etapas tempranas
Mediano plazo	Ocurrencia en etapas medianas
Largo plazo	Ocurrencia al final del proyecto

Tabla 13: Ocurrencia en el tiempo

## 6.4 Ponderación de los riesgos

Para poder identificar y controlar aquellos riesgos que en caso de suceder pudieran afectar el proyecto de forma considerable, se calculó la magnitud de cada uno. Siendo la magnitud como el producto entre el impacto y la probabilidad de ocurrencia.

Los niveles son tres y se desprenden la magnitud de los riesgos, calculada como:

Magnitud (M) = Impacto \* Probabilidad de Ocurrencia.

- Bajo (verde): **M ≤ 0,4** - Prioridad baja debido a baja probabilidad de ocurrencia y bajo impacto

- Moderado (amarillo):  $0,6 < M \leq 1,2$  - Prioridad media, se debe tener en consideración aunque si ocurriera su resultado no sería catastrófico.
- Alto (rojo):  $M \geq 1,6$  - Prioridad alta, es un riesgo a atacar inmediatamente ya que es crítico para el proyecto puede provocar su fracaso total

	Ninguno (0)	Marginal (1)	Poco importante (2)	Importante (3)	Crítico (4)	Catastrófico (5)
<b>No probable (0)</b>	0	0	0	0	0	0
<b>Poco probable (0,2)</b>	0	0,2	0,4	0,6	0,8	1
<b>Probable (0,4)</b>	0	0,4	0,8	1,2	1,6	2
<b>Muy probable (0,6)</b>	0	0,6	1,2	1,8	2,4	3
<b>Altamente probable (0,8)</b>	0	0,8	1,6	2,4	3,2	4
<b>Se convierte en un hecho (1)</b>	0	1	2	3	4	5

Tabla 14: Matriz de riesgos

## 6.5 Riesgos identificados

A continuación se listan los riesgos que el equipo identificó y fueron clasificados según su impacto positivo o negativo.

Tipo	Riesgo	Descripción	Impacto
Técnico	RT1 - Pérdida de electricidad	Pérdida de electricidad en alguna de las ubicaciones de trabajo.	Negativo
Técnico	RT2 - Pérdida de conectividad de internet	Pérdida de conectividad de internet en alguna de las ubicaciones de trabajo.	Negativo
Técnico	RT3 - Pérdida, destrucción o robo de notebook	Pérdida, destrucción o robo de notebook.	Negativo
Técnico	RT4 - Destrucción del disco duro	Destrucción del disco duro de algunos de los equipos.	Negativo
Técnico	RT5 - Problemas con la nube	Problemas de conectividad con la nube.	Negativo

Técnico	RT6 - Problemas con la tecnología	Problemas con las tecnologías utilizadas.	Negativo
Técnico	RT7 - Desconocimiento de la tecnología	Tener desconocimientos de las tecnologías que se vayan a utilizar.	Negativo
Técnico	RT8 - Falta de diseñador UI/UX	No tener conocimiento ni experiencia en el área de UI/UX.	Negativo
Humano	RH1 - Abandono de un recurso	Abandono de alguno de los integrantes del equipo	Negativo
Humano	RH2 - Pérdida de capacidad momentánea de un recurso	Abandono momentáneo de alguno de los integrantes del equipo	Negativo
Gestión	RG1 - Cambios en requerimientos	Modificación en los requerimientos del proyecto impactando el alcance.	Negativo
Gestión	RG2 - Mala definición del cronograma	Definición errónea del cronograma de trabajo.	Negativo
Gestión	RG3 - Falta experiencia ejecutando proyectos	Poca experiencia de parte de los integrantes ejecutando proyectos similares.	Negativo
Gestión	RG4 - Equipo multidisciplinario	Formar parte de un equipo multidisciplinario	Positivo
Cliente	RC1 - Baja disponibilidad para reuniones	El cliente presenta poca disponibilidad para reuniones con el equipo.	Negativo

Tabla 15: Riesgos

### 6.5.1 Descripción de los riesgos

Se establece información acerca de los riesgos para aclarar el contexto de cada uno de ellos.

- **RT1 - Pérdida de electricidad**

- Al trabajar con equipos electrónicos, la pérdida de electricidad puede repercutir de manera negativa, imposibilitando avanzar en el proyecto ya que se depende de gran manera de contar con electricidad e internet.
- Objetivo afectado: Cronograma

- **RT2 - Pérdida de conectividad de internet**

- Al igual que el riesgo anterior, el internet es crucial para el correcto desarrollo del proyecto. La posible pérdida de conectividad en alguno

de los equipos es un riesgo a tener en cuenta.

- Objetivo afectado: Cronograma
- **RT3 - Pérdida, destrucción o robo de notebook**
  - Al ser un proyecto desarrollado utilizando software, las notebooks corren el riesgo de ser robadas, dañadas o perdidas. Es por esto que es un riesgo a tomar en cuenta, ya que se puede llegar a perder todo el trabajo realizado en caso de no contar con un backup correspondiente.
  - Objetivo afectado: Cronograma
- **RT4 - Destrucción del disco duro**
  - La posible destrucción de un disco duro podría repercutir en la pérdida de información.
  - Objetivos afectados: Alcance, cronograma
- **RT5 - Problemas con la nube**
  - Al utilizar la nube se podrían presentar problemas como un bloqueo de accesos o problemas de seguridad.
  - Objetivos afectados: Alcance, cronograma
- **RT6 - Problemas con la tecnología**
  - Se podrían presentar posibles problemáticas a la hora de utilizar tecnologías, ya sean nuevas para el equipo o no, como por ejemplo la necesidad de contar con ciertas licencias.
  - Objetivos afectados: Cronograma, calidad
- **RT7 - Desconocimiento de la tecnología**
  - En caso de desconocer alguna de las tecnologías a ser utilizadas, la curva de aprendizaje de ciertos aspectos tecnológicos del proyecto puede ser muy grande para alguno de los integrantes del proyecto. Esto genera que el avance del desarrollo sea lento.
  - Objetivos afectados: Cronograma
- **RT8- Falta de diseñador UI/UX**
  - Ninguno de los integrantes del equipo tiene conocimientos o experiencia en el área de UI/UX por lo que para mitigar este riesgo se plantea

capacitación y utilización de templates ya prontos, como la utilización de un framework de UI/UX.

- Objetivo afectado: Cronograma, calidad
- **RH1 - Abandono de un recurso**
  - Cualquiera de los integrantes del equipo podría abandonar el proyecto durante la ejecución. En dicho caso, los recursos del proyecto disminuirán y generará un impacto crítico en todos los objetivos del proyecto.
  - Objetivos afectados: Alcance, cronograma, calidad
- **RH2 - Pérdida de capacidad momentánea de un recurso**
  - Cualquiera de los integrantes del equipo podría abandonar de forma momentánea el proyecto por diferentes razones como puede ser enfermedad o trabajo. En dicho caso, los recursos del proyecto disminuirán por un tiempo determinado y generará un impacto crítico en todos los objetivos del proyecto.
  - Objetivos afectados: Alcance, cronograma, calidad
- **RG1 - Cambios en requerimientos**
  - Si bien inicialmente se pactaron los requerimientos con el cliente, puede ocurrir que en etapas de desarrollo el cliente decida cambiar un requerimiento por restricciones de negocio. Un cambio en etapas tardías puede afectar no solo el cronograma sino que también el alcance.
  - Objetivos afectados: Alcance, cronograma
- **RG2 - Mala definición del cronograma**
  - Una mala definición del cronograma puede saberse por malas estimaciones, imposibilidad de paralelización de tareas, entre otros, esto puede generar un impacto fuerte en el alcance del proyecto.
  - Objetivos afectados: Alcance, cronograma, calidad
- **RG3 - Falta experiencia ejecutando proyectos**
  - Las integrantes del equipo pueden contar con poca experiencia ejecutando proyectos similares, repercutiendo esta falta de experiencia

en la fluidez a la hora de apegarse a las metodologías de trabajo entendiéndose así los tiempos pautados.

- Objetivos afectados: Cronograma, calidad

- **RG4 - Equipo multidisciplinario**

- Los integrantes del equipo poseen experiencia laboral en rubros y roles distintos. Tener enfoques variados colabora con la toma de decisiones y las tareas que serán ejecutadas a lo largo del proyecto.
- Objetivo afectado: Calidad

- **RC1 - Baja disponibilidad para reuniones**

- Coordinar reuniones con el cliente puede ser complejo en caso que no cuenten con una disponibilidad para reuniones. Esto puede ocasionar que el equipo no pueda recolectar el feedback necesario para avanzar alineados a ellos.
- Objetivo afectado: Cronograma

## 6.6 Análisis cualitativo de los riesgos

Se determinó la magnitud de cada riesgo negativo para poder determinar qué estrategia usar frente a cada uno:

Riesgo	Probabilidad de ocurrencia	Nivel de impacto	Magnitud	Nivel
RT1 - Pérdida de electricidad	0,2	2	0,2	Bajo
RT2 - Pérdida de conectividad de internet	0,2	2	0,4	Bajo
RT3 - Pérdida, destrucción o robo de notebook	0,2	2	0,4	Bajo
RT4 - Destrucción del disco duro	0,2	2	0,4	Bajo
RT5 - Problemas con la nube	0,2	3	0,6	Moderado
RT6 - Problemas con la tecnología	0,2	4	0,8	Moderado

RT7 - Desconocimiento de la tecnología	0,4	4	1,6	Alto
RT8 - Falta de diseñador UI/UX	0,8	3	2,4	Alto
RH1 - Abandono de un recurso	0,2	5	1	Moderado
RH2 - Pérdida de capacidad momentánea de un recurso	0,4	3	1,2	Moderado
RG1 - Cambios en requerimientos	0,6	3	1,8	Alto
RG2 - Mala definición del cronograma	0,4	3	1,2	Moderado
RG3 - Falta de experiencia ejecutando proyectos	0,4	4	1,6	Alto
RC1 - Baja disponibilidad para reuniones	0,2	3	0,6	Moderado

Tabla 16: Análisis de riesgos

## 6.7 Definición de estrategias genéricas

Según el nivel de los riesgos, se definieron estrategias genéricas a seguir para actuar en consecuencia.

Nivel de impacto	Estrategia a utilizar	Descripción
Bajo	Aceptar pasivamente	El equipo reconoce el riesgo y decide no tomar ninguna medida. Es necesario documentar la estrategia para que el equipo aborde los riesgos conforme se presentan.
Moderado	Mitigar	Se debe actuar para disminuir la probabilidad de ocurrencia.
Alto	Evitar	Implica realizar cambios en el plan del proyecto con el fin de eliminar la amenaza de manera temprana.

Tabla 17: Estrategia sobre impacto

Las estrategias ayudan a disminuir los costos que se podrían generar en el proyecto si ocurren los eventos.

Para los riesgos bajos se pretende aceptarlos pasivamente dado que como su afectación en los objetivos del proyecto es menor, tomar alguna acción podría terminar repercutiendo de peor manera.

La estrategia para los moderados es mitigarlos y así reducir su probabilidad de ocurrencia. Dependiendo del objetivo afectado por el riesgo la acción de mitigación que será utilizada.

La intención de la estrategia para los altos es evitarlos, puesto que son los que tienen una mayor repercusión en los objetivos del proyecto. En caso de que algún riesgo de esta categoría no pueda ser evitado, se buscarán otras alternativas con el fin de disminuir sus efectos.

## 6.8 Plan de respuesta y contingencia

Una vez identificados y categorizados los riesgos, y habiendo definido las estrategias genéricas, el equipo elaboró un plan para poder hacer un seguimiento de los riesgos a lo largo del proyecto.

Con respecto a los planes de respuesta, estas son las actividades que el equipo considera preventivas de materialización de algún riesgo. La rigurosidad con que las llevemos a cabo puede evitar su materialización o disminuir el impacto de los mismos.

También se describe un plan de contingencia en caso de una crisis o emergencia en donde haya que accionar de forma rápida.

Riesgo	Estrategia	Plan de respuesta	Plan de contingencia
RT1 - Pérdida de electricidad	Aceptar	Lograr siempre tener un margen de carga en las notebooks de modo que si la electricidad se pierde se pueda seguir trabajando por un lapso de tiempo.	Al trabajar en ubicaciones de trabajo, que la electricidad falle en otro punto de trabajo abastecerse de electricidad.
RT2 - Pérdida de conectividad de internet	Aceptar	Asegurarse tener más de una opción de acceso a internet, como puede ser desde un celular fijo, además de la red wifi.	Se tienen identificadas ubicaciones diferentes con internet por lo que si se puede solventar trasladarse a otro punto. También se puede avanzar en el proyecto de manera local, y una vez

			el acceso se resincronización de lo trab
RT3 - Pérdida, destrucción o robo de notebook	Aceptar	Asegurarse tener la sincronización y almacenamiento de todos los documentos y código en un repositorio en la nube.	Contamos con una nota de reserva con el trabajo ya preparado, de pueda usarse inmediatamente no afectar los tiempos.
RT4 - Destrucción del disco duro	Aceptar	Asegurarse de tener el proyecto almacenado en la nube.	Al trabajar con repositorio en la nube evitamos que la información del disco duro tenga copia sobre el proyecto y la información.
RT5 - Problemas con la nube	Mitigar	Tener sincronizado de forma local los repositorios.	Si hubiera un error en la nube nos bloqueara los trabajos, como tenemos sincronizado de forma local los repositorios a poder seguir trabajando. Se hace una coordinación para que todo siga funcionando los trabajos, no pisar los trabajos realizados en los diferentes dispositivos.
RT6 - Problemas con la tecnología	Mitigar	Realizar una investigación profunda previo a la elección de la tecnología, consultando con especialistas, evaluando las ventajas y desventajas, curva de aprendizaje e información disponible.	Se solicitará ayuda de un experto externo que pueda resolver rápidamente y evacuar dudas.
RT7 - Desconocimiento de la tecnología	Evitar	Se acuerda dedicar horas extra de investigación y capacitación del equipo. Compartir el conocimiento con el resto del equipo.	Si algún integrante del equipo no puede avanzar por falta de conocimiento se agenda una reunión para la programación de a parte de otro desarrollador.
RT8 - Falta de diseñador UI/UX	Evitar	Se acuerda dedicar horas de investigación, utilizar plantillas y frameworks existentes y reconocidos. Se acordará con el cliente de antemano el nivel esperado de diseño.	Se hará una primera versión con foco en lo funcional, con mejoras de usabilidad y una segunda instancia de mejoras pactadas por el cliente.
RH1 - Abandono de un recurso	Mitigar	Generar un ambiente de trabajo transparente, que fomente la comunicación del equipo para que cualquier integrante se sienta libre de manifestar	Se tienen identificadas funcionalidades menos importantes que podrían quitarse con el fin de compensar de horas del recurso.

		disconformidades.	
RH2 - Pérdida de capacidad momentánea de un recurso	Mitigar	Se deberá de notificar con antelación de un mes para que el equipo pueda planificarse con suficiente tiempo y que el avance del proyecto se vea afectado lo mínimo posible.	En caso de que el avance sea afectado, se deberán hacer ajustes en el cronograma, aumentando horas de dedicación en el sprint.
RG1 - Cambios en requerimientos	Evitar	Se intentará consolidar la mayor cantidad de requerimientos en etapas tempranas, generando el mayor entendimiento posible del negocio y del cliente, para que los requerimientos sufran la menor cantidad de cambios posibles.	Si el cliente solicita un cambio en alguno de los requerimientos en etapas tardías, el equipo evaluará lo posible implementar dicho cambio sin afectar el alcance. En caso de que no se vea afectado con dicho cambio se informará al cliente y se ofrecerá una alternativa en caso de que podrá ser quitar algún requerimiento con prioridad baja.
RG2 - Mala definición del cronograma	Mitigar	Se optará por una estimación pesimista sobre los requerimientos con el fin de generar un margen en caso de un error.	Se compensará con horas por parte del equipo.
RG3 - Falta experiencia ejecutando proyectos	Evitar	Previo a comenzar el proyecto el equipo se compromete a tener profundo conocimiento sobre la modalidad Scrum y estar informado sobre pautas estipuladas por ORT para la realización de la tesis.	En caso de que el proyecto sea afectado en tiempo o calidad el equipo deberá compensar el resultado ejecutando horas.
RC1 - Baja disponibilidad para reuniones	Mitigar	Se acuerda previo al comienzo del proyecto las condiciones para las reuniones, dejando fijas ya una cantidad mínima y especificando disponibilidad tanto del equipo como del cliente.	Se contactará al cliente por medios, como lo es correo o llamadas desistiendo así de reuniones grupales.

Tabla 18: Plan de respuesta y contingencia sobre riesgos

## 6.9 Evolución de riesgos principales

A continuación se muestra la evolución a lo largo del proyecto de los cuatro riesgos principales, estos son los que presentan una magnitud alta.

Evolución de Riesgo: RT7 - Desconocimiento de la tecnología

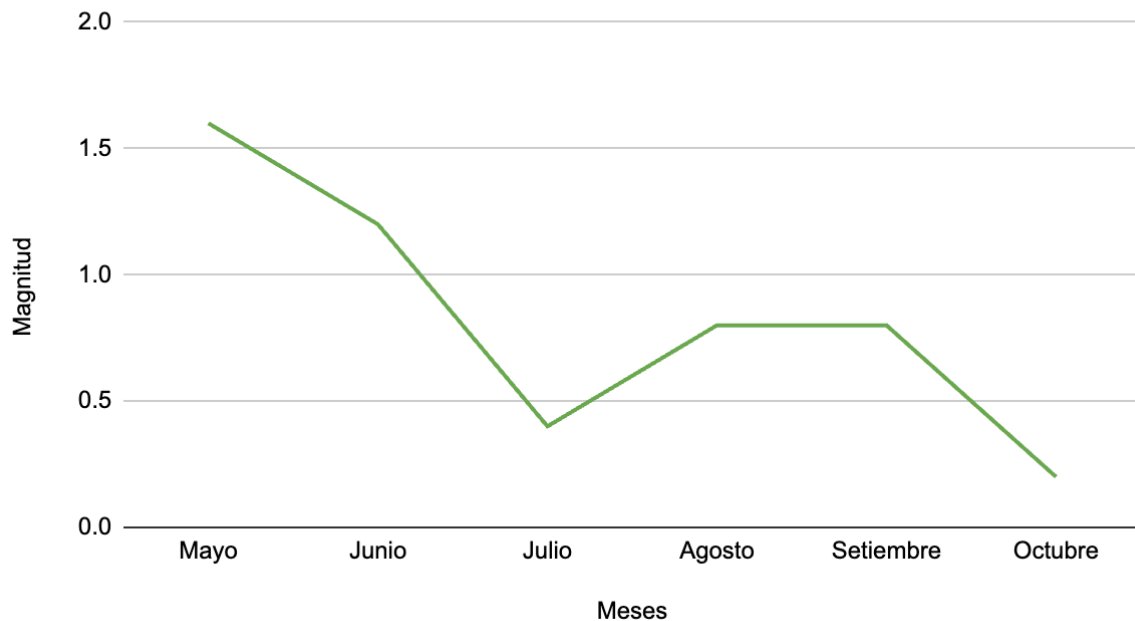


Ilustración 64: Evolución del riesgo: RT7 - Desconocimiento de la tecnología

En lo que respecta al desconocimiento de la tecnología, se puede observar que la magnitud del riesgo fue un tanto variable, al comienzo se tenía una magnitud alta ya que el equipo contaba con poco conocimiento de React principalmente, lo que obligó a destinar horas en capacitación, en los primeros *sprints* el riesgo fue disminuyendo ya que se fue incorporando conocimiento y el equipo se fue desarrollando con mayor fluidez a la hora de desarrollar. En Agosto el riesgo subió un poco debido a que se comenzó a trabajar en requerimientos que necesitaron nuevamente de una mayor capacitación, también se comenzó a evaluar aspectos de posibles servicios de hosting con mayor detalle e integraciones con otras herramientas como Google Analytics lo cual requirió de investigación sobre estos temas desconocidos. Sobre el final del proyecto el riesgo volvió a disminuir ya que se había finalizado de gran manera con el desarrollo por lo que el desconocimiento de las tecnologías era mínimo a esa altura.

Evolución de Riesgo: RT8 - Falta de diseñador UI/UX

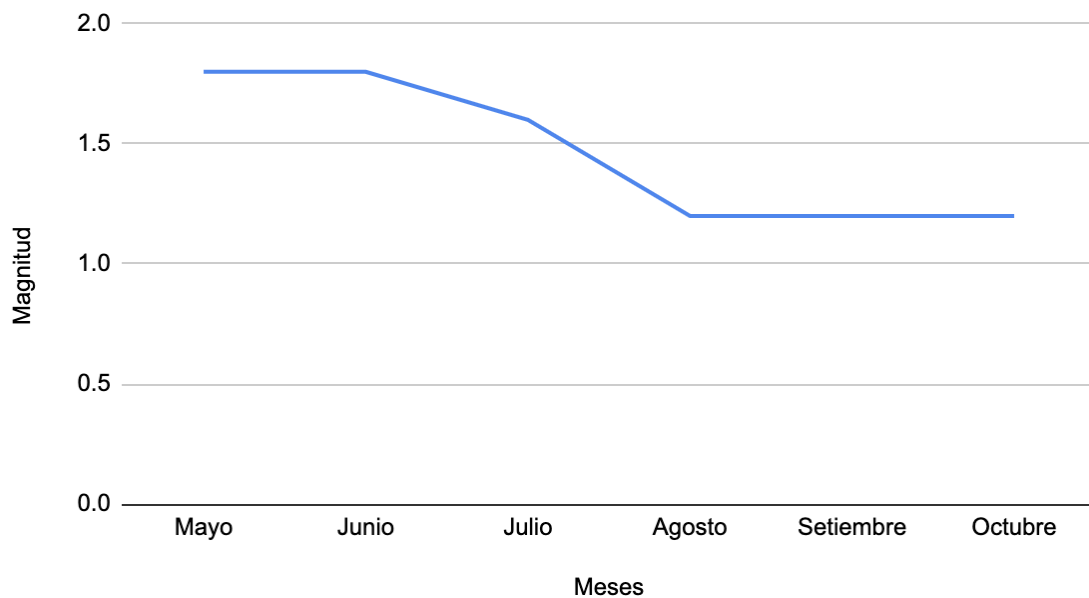


Ilustración 65: Evolución del riesgo: RT8 - Falta de diseñador UI/UX

El riesgo RT8 - Falta de diseñador *UI/UX*, tuvo una evolución estable, disminuyendo *sprint* en los sprints iniciales y estabilizando al final. Al principio, resultó ser un riesgo alto ya que no se contaba con conocimientos ni experiencia en relación a la interfaz gráfica y la usabilidad. En la medida que se fue avanzando en el proyecto, fue disminuyendo a base de cursos y capacitaciones, y en gran medida gracias al uso de *frameworks* específicos para *UI/UX*, como ser Material UI [13].

### Evolución de Riesgo: RG1 - Cambios en los requerimientos

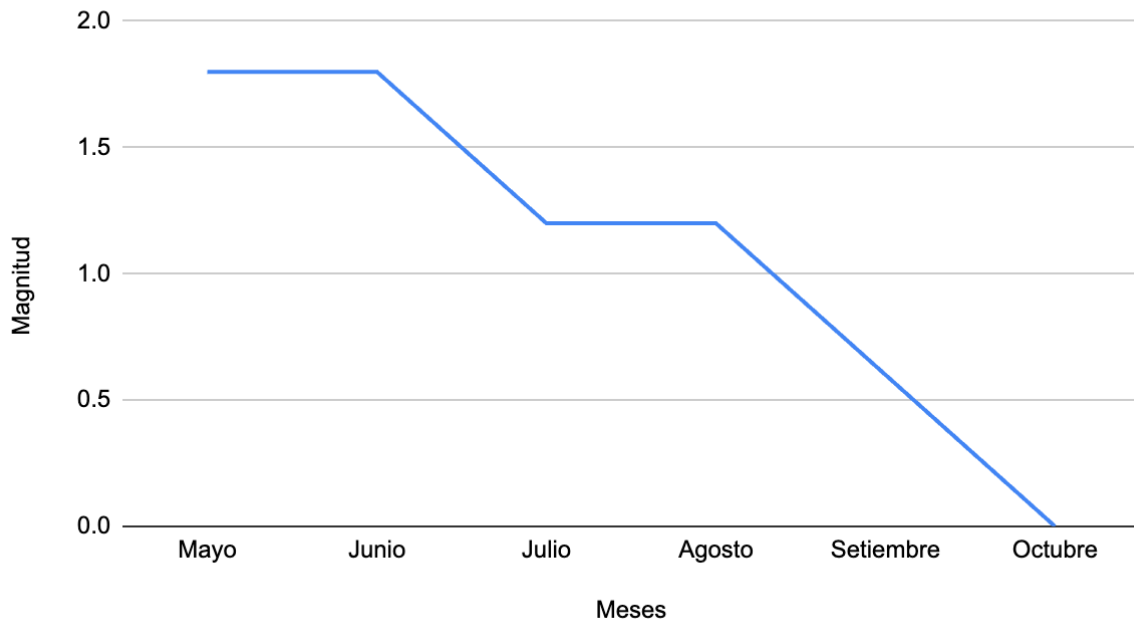


Ilustración 66: Evolución del riesgo: RG1 - Cambios en los requerimientos

A partir de la gráfica podemos concluir que la magnitud del riesgo relacionado a los cambios en los requerimientos fue disminuyendo a medida que se avanzaba en el proyecto, esto es así ya que se puso gran empeño en que los requerimientos variarían lo mínimo posible en etapas avanzadas del proyecto, intentando de tener lo más definido posible las necesidades de la plataforma en etapas tempranas. Los primeros meses donde se hizo el relevamiento y el intercambio con el cliente fue el momento donde más cambios se realizaron a la lista de requerimientos. Sobre Julio hubieron algunas modificaciones al *product backlog* en común acuerdo con el cliente, y ese riesgo de cambio se mantuvo hasta fines de Agosto cuando finalmente el riesgo comenzó a descender ya que se logró un gran entendimiento con el cliente sobre lo que se esperaba para etapas finales, logrando así que los posibles cambios sean mínimos.

Evolución de Riesgo: RG3 - Falta de experiencia ejecutando proyectos

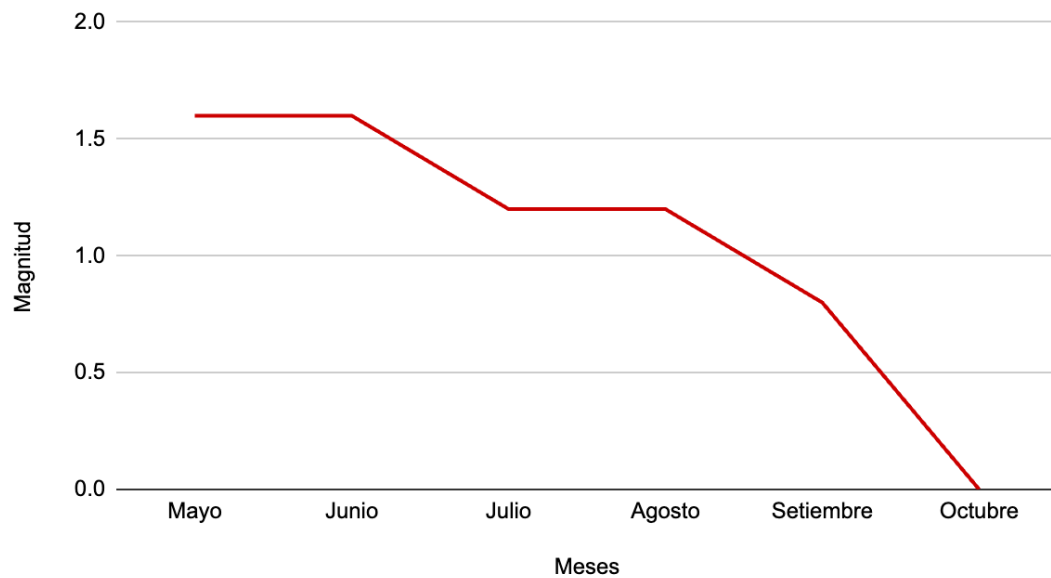


Ilustración 67: Evolución del riesgo: RG3 - Falta de experiencia ejecutando proyectos

Al comienzo del proyecto, estaba latente el riesgo relacionado a la falta de experiencia ejecutando proyectos, si bien las tres integrantes cuentan con experiencia laboral ejecutando proyectos de tecnología, se tenía cierta incertidumbre de cómo sería la metodología de trabajo del proyecto de fin de carrera y que otros posibles aspectos nuevos abarcaría. La magnitud fue disminuyendo a medida que se avanzaba en los

# 7 Gestión de cambios e incidentes

## 7.1 Introducción

El objetivo de la gestión de incidentes fue encontrar posibles errores y mejoras que existiesen en el producto, ya sean comunicadas por parte del cliente o detectados internamente en el equipo, para luego poder corregirlos y siempre velar por obtener un producto final de calidad que aporte valor al cliente.

Se decidió armar un plan para la gestión de cambios e incidentes para evitar posibles desvíos en el cronograma y el alcance causados por la implementación de mejoras a la solución.

Nuestro foco principal es definir un proceso que permita diferenciar claramente los incidentes de los cambios y facilite la gestión de los mismos.

Para poder diferenciarlos, se tomaron las siguientes definiciones para cada uno:

- **Incidente** → comportamiento no deseado que genera problemas en el uso de la aplicación.
- **Cambio** → comportamiento del sistema que no fue tenido en cuenta en el alcance pero puede ser tomado en cuenta para mejorar la experiencia del usuario.

El relevamiento de mejoras e incidentes fue llevado a cabo durante toda la ejecución del proyecto, a partir de las pruebas que se fueron realizando, puestas en común internas con el equipo y también en reuniones con el cliente.

El registro inicial se llevó a cabo en Google Sheets. Los datos que se registraron son los siguientes:

- **Requerimiento asociado:** Indica a qué requerimiento está asociado el elemento.
- **Categoría:** Se detalla si se trata de un incidente o un cambio/mejora.
- **Tipo:** Interno: del equipo, Externo: enviado por el cliente.
- **Área:** Si es un problema de *backend*, de *frontend* o del ambiente de prueba.

- **Nombre:** Nombre descriptivo.
- **Descripción:** Descripción del caso, detallando los pasos a seguir para poder reproducirlo.
- **Severidad:** Si es un incidente, se detalla la criticidad: Crítica, Alta, Media o Baja.
- **Detectado:** *Sprint* en el cual se detectó.
- **Resuelto:** *Sprint* en el cual se corrigió.

Adicionalmente, cada integrante del equipo que agregaba un registro en esta planilla, se encargaba de copiar al resto de las integrantes para que estuvieran al tanto del mismo.

## 7.2 Gestión de cambios

Como los cambios no fueron tenidos en cuenta en la planificación inicial del proyecto, el equipo se mantuvo en una comunicación constante con el cliente para evaluar si deberían ser parte del alcance del proyecto o podía ser considerado para una futura versión del *MVP*. En varias ocasiones, el cliente optó por priorizar los cambios planeados y despriorizar alguna de las funcionalidades originalmente planificadas. Los cambios siempre serán estimados y enviados al *product backlog* para ser tomados en el siguiente *sprint* al cual son reportados. En varias ocasiones si disponíamos de tiempo en el *sprint* que eran reportados, se hacía la excepción y se toman igual, saliendo así del flujo planificado.

## 7.3 Gestión de incidentes

Los incidentes fueron categorizados según su severidad y se definió un tiempo de resolución para los mismos.

Severidad	Descripción	Resolución
Crítica	Bloquea totalmente el uso del sistema	Inmediata en el <i>sprint</i> que se reporta
Alta	Bloquea una funcionalidad	Inmediata en el <i>sprint</i> que se reporta
Media	No generan bloqueos, pero están asociados a la interacción con el cliente. Ejemplo: mensajes de	Puede esperar al próximo <i>sprint</i> de ser reportado

	error confusos	
Baja	No influyen en el flujo normal de la funcionalidad pero si en uno alternativo	Puede esperar hasta dos sprints de ser reportado

Tabla 19: Categorización de incidentes

### 7.3 Plan de gestión

Una vez definidos los conceptos generales para diferencias tanto los cambios de los incidentes, se genera el siguiente plan que resumirá el proceso a llevarse a cabo.

Se debe destacar que este proceso será puesto en práctica para cada incidente/mejora reportado.

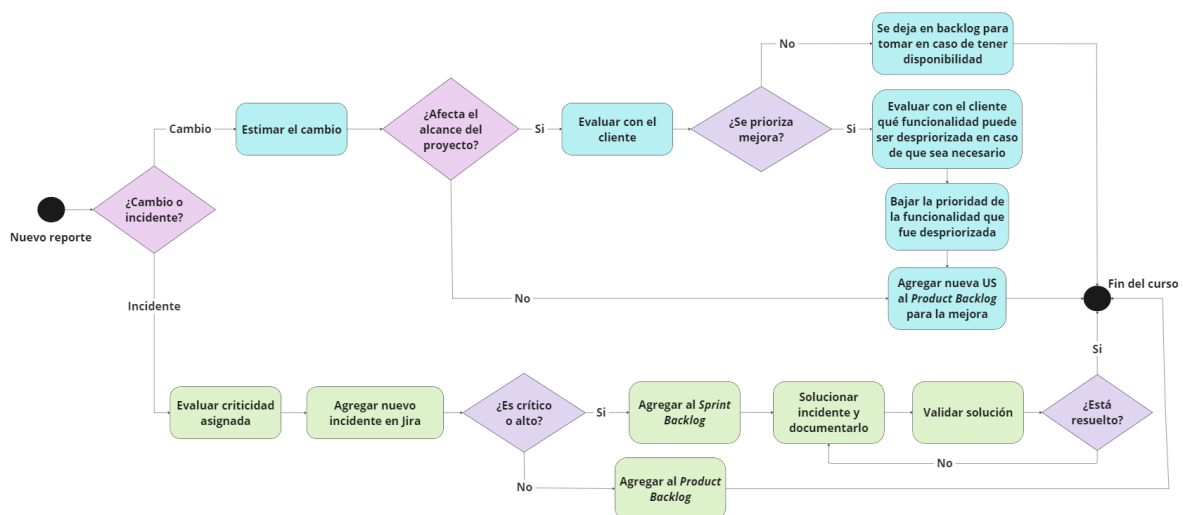


Ilustración 68: Proceso de gestión de cambios e incidentes

WAREUP-223 Al querer cambiar contraseña me salta error: "Usuario no autorizado"	FINALIZADA	3	RA
WAREUP-224 [Editar depósito] no se actualizan address y postalCode	FINALIZADA	1	LP
WAREUP-226 [Agregar imágenes a un deposito existente] Falla con Failed to fetch data	FINALIZADA	1	RA

Ilustración 69: Evidencia de registro de incidentes en Jira

### 7.5 Resultados del plan de gestión

En cada *weekly meeting* el equipo revisaba si existían nuevos incidentes o mejoras reportadas en la planilla, más allá de estar enteradas porque nos notificabamos cuando se agregaba el registro, nos servía para poner sobre la mesa todos los casos que teníamos para revisar y eventualmente trabajar sobre ellos. En caso del *feedback*

obtenido por el cliente, revisamos diariamente la planilla en caso de que el mismo haya dejado algún comentario que fuese necesario responder y mantener una charla fluida.

Por otro lado, a lo largo de los *sprints* se realizó la ejecución de pruebas funcionales para validar el comportamiento de las funcionalidades implementadas.

Durante la ejecución, se detectaron comportamientos inesperados que fueron categorizados en incidentes o cambios según el plan que se mencionó anteriormente.

Se detectaron en total **53** reportes, donde **34** fueron categorizados como incidentes y **19** como mejoras.



Ilustración 70: Gráfico porcentual de incidentes y mejoras detectadas

En el [Anexo 6 - Registro de incidentes](#) se encuentra disponible la tabla completa de reportes que se generaron.

### 7.5.1 Gestión de incidentes

En el caso de los incidentes, optamos por categorizarlos por severidad para tener un mayor entendimiento de la gravedad de los incidentes detectados.



Ilustración 71: Gráfico porcentual de incidentes por severidad

Como se puede apreciar en la gráfica de arriba, la mayor parte de los incidentes fueron de severidad media por lo cual no se generó una afectación total del sistema ni de ninguna funcionalidad en particular. El **2,9%** crítico corresponde a la primera release. El ambiente que generamos para compartirle la web al cliente era un ambiente gratuito y limitado en ciertos recursos, por ende, se presentaba cierta inestabilidad a la hora de realizar las pruebas. Lo mismo ocurrió con la base de datos temporal que utilizamos ya que al ser gratis tiene algunos inconvenientes al guardar las imágenes que son muy grandes. Esto fue comunicado al cliente para que lo tenga en cuenta al momento de realizar sus pruebas ya que era un error conocido.

Para tener un mejor contexto, analizamos cuántos fueron registrados por el equipo mismo y cuantos fueron reportados por el cliente. Por otra parte, también entender el origen de estos incidentes, si fueron por el *frontend*, *backend* o el ambiente de prueba generado para que el cliente pudiera realizar sus pruebas.

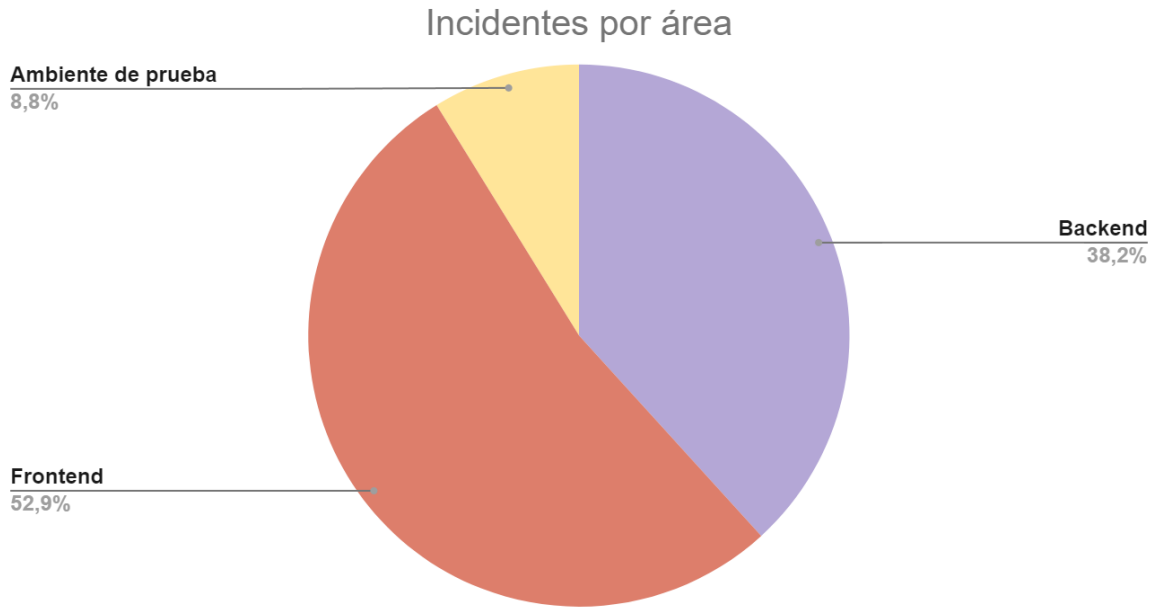


Ilustración 72: Gráfica porcentual de incidentes por área

De acuerdo a lo que podemos ver en la gráfica anterior, el mayor porcentaje **52,9%** de los incidentes fue ocasionado en el *frontend*. Esto se debió al desconocimiento mismo del entorno de React por parte de las integrantes del equipo, por ende tuvimos que poner mucho foco en revisar y generar todas las pruebas posibles con el fin de asegurarnos de que el producto funcione como esperamos.

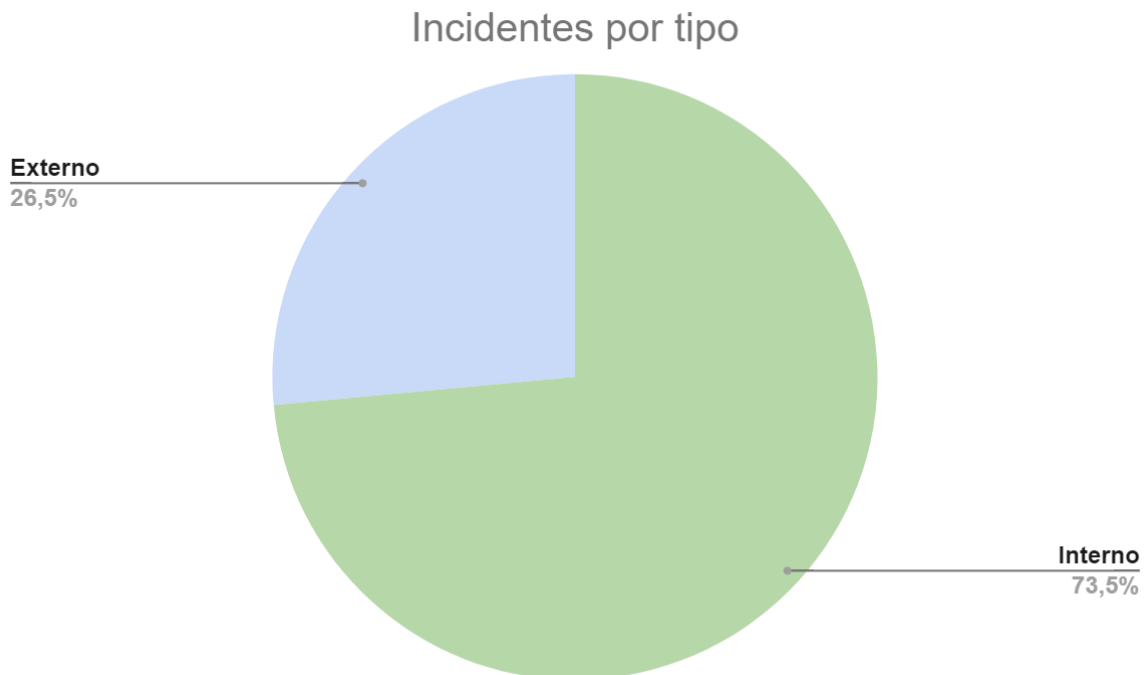


Ilustración 73: Gráfica porcentual de incidentes por tipo

El **73,5%** de los incidentes reportados fueron por el equipo. Esto demuestra el grado de detalle y dedicación que pusimos en las pruebas para asegurarnos de que el producto sea lo acordado antes de enviarle avances al cliente.

#### 7.5.1.1 Resolución de los incidentes

Los incidentes detectados y resueltos se distribuyeron de la siguiente forma a lo largo de los *sprints*:

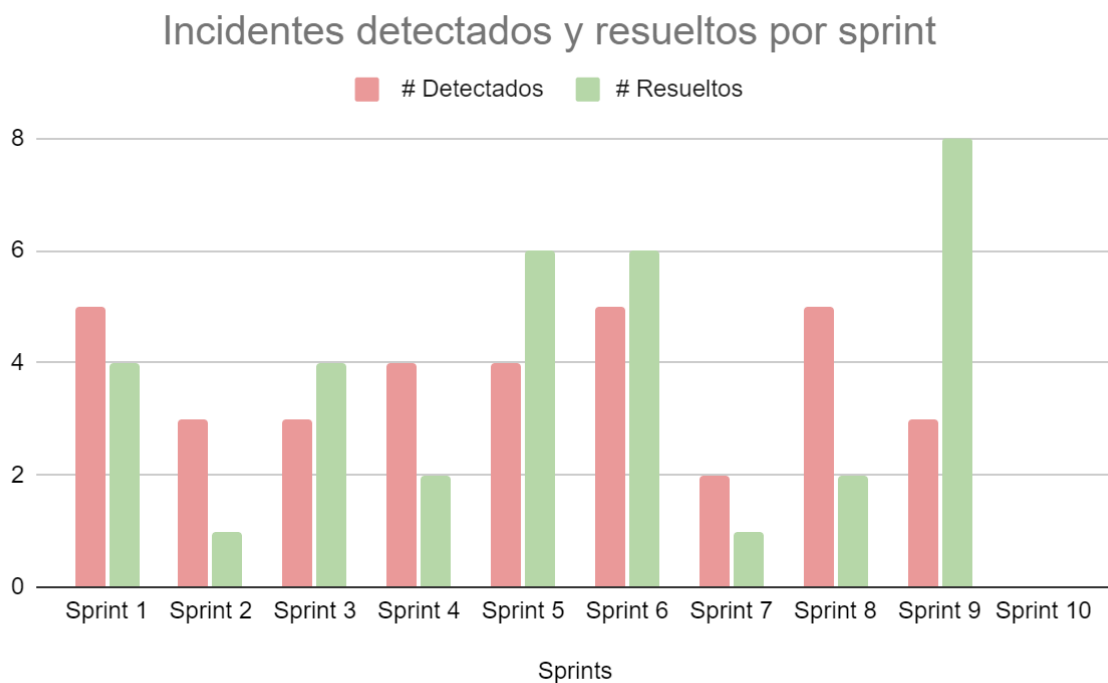


Ilustración 74: Gráfica con cantidad de incidentes detectados y resueltos por sprint

Como se puede visualizar en la gráfica de arriba, el **Sprint 5** fue donde se resolvió mayor cantidad de incidentes ya que fue el *sprint* antes de la release I, y en el **Sprint 6** tuvimos el *feedback* del cliente, por ende, era de esperarse que incrementara la cantidad de incidentes pero también se esperaba tener una acción más proactiva y rápida en la resolución de los mismos. Por este motivo, el **Sprint 6** no se trabajó mucho en implementaciones nuevas y el poco tiempo que el equipo le pudo dedicar, estuvo más enfocado en la resolución de incidentes. Lo mismo pasó con el **Sprint 9**, que fue previo a la *release* II.

Profundizando un poco más en las severidades de los incidentes, analizamos el tiempo de resolución de los mismos para comprobar si se cumplió con el tiempo de resolución definido anteriormente.

Los incidentes catalogados con severidad crítica y alta, en el **100%** de los casos, fueron resueltos en el mismo *sprint* en el que fueron detectados, cumpliendo así con el tiempo de resolución definido en el plan de gestión de incidentes.

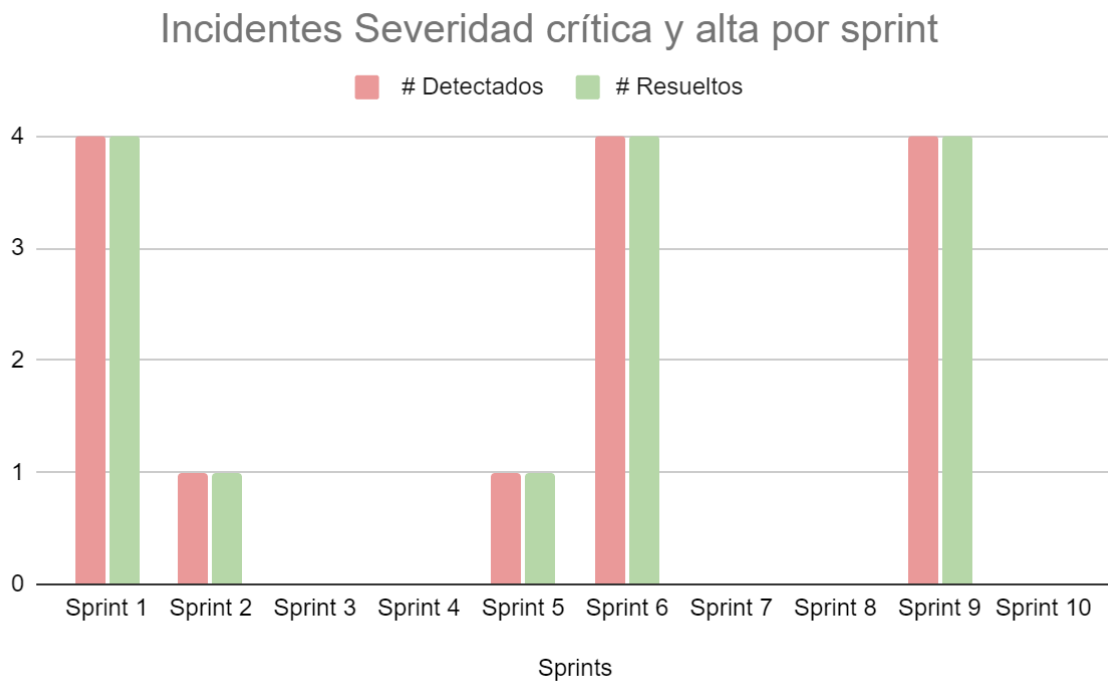


Ilustración 75: Cantidad de incidentes con severidad crítica y alta por sprint

Con lo que respecta a los incidentes con severidad media y baja, podemos visualizar a continuación que también se cumplió con el tiempo de resolución definido, incluso en algunos casos resolviéndose antes de la fecha límite.

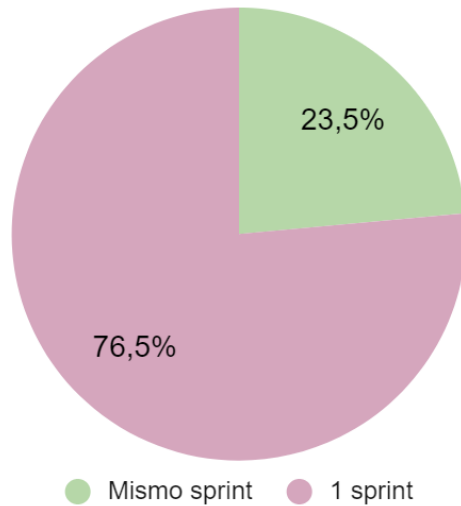


Ilustración 76: Gráfico porcentual de tiempo de resolución de incidentes con severidad media

En el caso de los incidentes con severidad media se resolvieron tal cual lo planificado y en un **23,5%** se superó la estimación haciendo el cambio mismo en el *sprint* que se reportó.

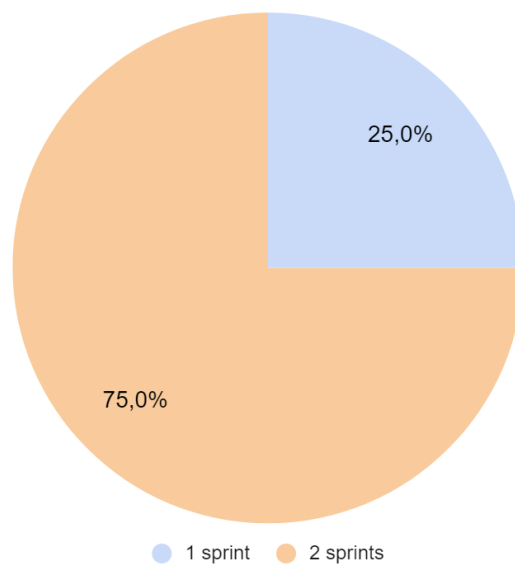


Ilustración 77: Gráfico porcentual de tiempo de resolución de incidentes con severidad baja

Por otro lado, con los incidentes de severidad baja tuvimos la misma situación. Se cumplió en un **100%** el tiempo de resolución definido e incluso se superó en algunos casos donde el **25%** de los casos fueron resueltos un *sprint* siguiente al reportado cuando el tiempo de resolución era de hasta 2 *sprints* para la resolución.

Como resolución final, podemos observar que todos los incidentes reportados fueron resueltos exitosamente y tomados en cuenta para no volver a cometerlos en futuras iteraciones del producto.

De este análisis se desprende que el hecho de tener un plan de incidentes enfocado en la calidad del producto tuvo efectos positivos. La definición de las reglas generales para detectar y solucionar los incidentes fueron cumplidas a los largo del proceso, obteniendo así buenos resultados.

### 7.5.2 Relación de incidentes con funcionalidades

Como mencionamos anteriormente, las pruebas funcionales fueron de gran importancia para la gestión de los incidentes. En el [Anexo 5 - Casos de prueba funcionales](#) se pueden encontrar los casos de prueba y resultados obtenidos.

Se realizará un análisis de la relación entre los incidentes y las distintas implementaciones, para evaluar la calidad de las implementaciones.

El total de pruebas funcionales ejecutadas fue **107** y se distribuyó de la siguiente manera:

<b>Sprint</b>	<b>Cantidad de pruebas ejecutadas</b>	<b>Cantidad de incidentes reportados</b>	<b>Cantidad de incidentes resueltos</b>
<b>1</b>	13	5	4
<b>2</b>	15	3	1
<b>3</b>	23	3	4
<b>4</b>	15	4	2
<b>5</b>	14	4	6
<b>6</b>	5	5	6
<b>7</b>	6	2	1
<b>8</b>	8	5	2
<b>9</b>	7	3	8
<b>10</b>	1	0	0
<b>Totales</b>	<b>107</b>	<b>34</b>	<b>34</b>

Tabla 20: Pruebas funcionales ejecutadas

Como se puede apreciar en la tabla anterior, el mayor foco de pruebas estuvo destinado en la primera mitad de sprints, ya que se desarrollaron las funcionalidades claves para el producto. Luego, las funcionalidades que fueron quedando a lo largo de los sprints requerían pruebas pero eran más concretas y no tan variables como los sprints anteriores. Por ejemplo, el registro de las distintas entidades fue de las funcionalidades que más pruebas requirió debido a la cantidad de campos a completar y velar por su correcta completitud.

### 7.5.3 Gestión de cambios

Se detectaron **19** mejoras a lo largo de todo el proyecto, siendo estas sugerencias por parte interna del equipo como también por parte del cliente. Las mejoras fueron encontradas y resueltas con la siguiente distribución a lo largo de los *sprints*.

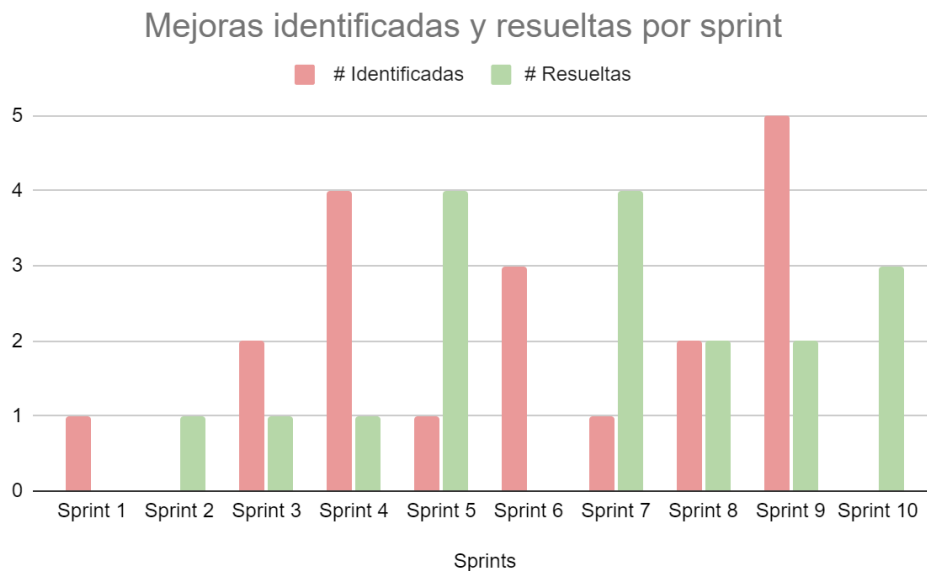


Ilustración 78: Cantidad de mejoras identificadas y resueltas por sprint

Las mejoras iban siendo tomadas a lo largo de todo el proyecto, como mencionamos anteriormente, una vez que eran identificadas se analizaron y se estimaron, se evaluaba con el cliente en caso de que fuese necesario despriorizar algún otro requerimiento y se agregaba al *Product Backlog*. Intentábamos en todos los sprints posibles tomar alguna mejora reportada con el fin de no generar un backlog infinito y despriorizado. En los casos donde no se tomaron mejoras fueron en los **Sprint 1** y

**Sprint 6** ya que en el primero priorizamos la resolución de incidentes importantes y en el **Sprint 6** el esfuerzo del equipo se vió reducido en el proyecto por motivos médicos de una integrante.

A continuación, se detalla cómo fue la distribución de los mismos en términos de tipo y de área.

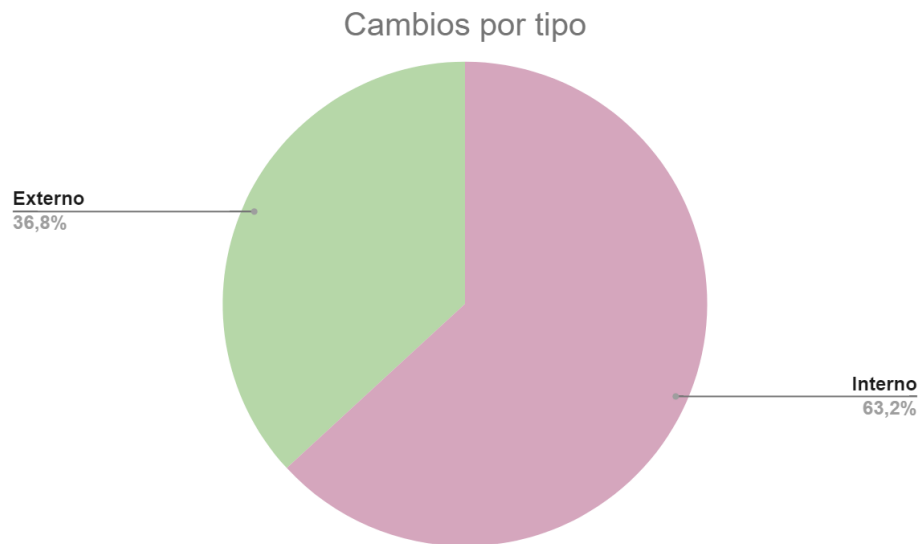


Ilustración 79: Gráfica porcentual de cambios por tipo

Podemos visualizar que la mayor cantidad de cambios que se reportaron fueron internos en el equipo. Esto demuestra el querer siempre mejorar nuestra propia solución con el fin de tener el mejor producto posible.

## Cambios por área

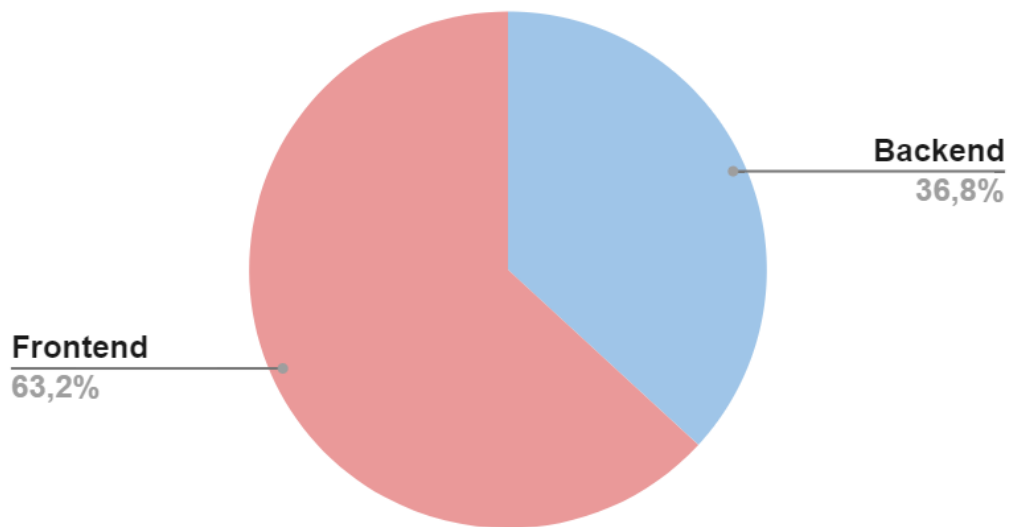


Ilustración 80: Gráfica porcentual de cambios por área

Con lo que respecta al área de reporte de los cambios, la mayoría fueron correspondientes al *frontend* esto demuestra el no conocimiento de las integrantes en el área de UX por ende, a medida que íbamos aprendiendo sobre el tema, intentábamos buscar mejoras para la web.

Con respecto a la resolución de las mejoras, podemos ver que no se cumplió con un 100% sino que se cumplió con un **95%** quedando por fuera una única mejora sin resolver. Esto fue discutido con el cliente y se entendió que no era fundamental para la primera versión del *MVP*.

### Resolución de mejoras detectadas

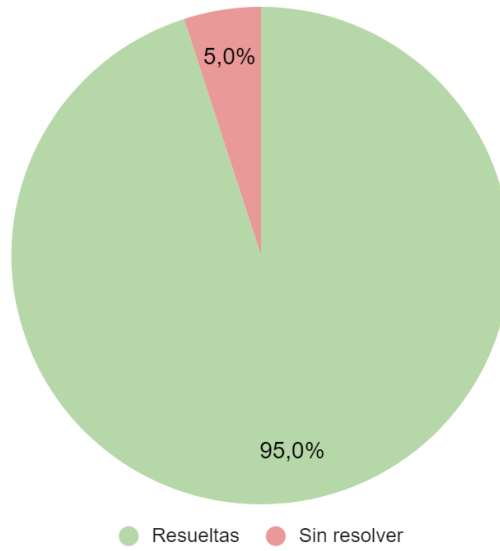


Ilustración 81: Gráfica porcentual con la resolución de las mejoras detectadas

## 8 Gestión de la calidad

### 8.1 Introducción

A lo largo del proyecto se realizaron distintas actividades para poder asegurar tanto la calidad del producto, como la calidad del proceso.

Para poder asegurar la calidad del proceso, se buscaron mecanismos que permitan definir un proceso ordenado, que se adecúe a las características del proyecto. Debido a que se decidió utilizar la metodología *Scrum*, se tienen un conjunto de artefactos y ceremonias que, al ser utilizadas y monitoreadas, permiten obtener un proceso controlado de buena calidad.

Por otro lado, para asegurar la calidad el producto se puso énfasis en una buena calidad de código, siendo buenos estándares de codificación, llevando a cabo también la realización de pruebas. Por otro lado, se fue haciendo un seguimiento con el cliente para asegurarnos estar cumpliendo con sus expectativas y estar desarrollando el producto deseado por el cliente.

### 8.2 Objetivos de calidad

#### 8.2.1 Del producto

Objetivo general asociado	Objetivo	Requerimientos a los que aplica
OP1	<b>OCPR1:</b> Lograr un mínimo producto viable que cumpla con lo requerimientos funcionales y no funcionales	<b>Funcionales:</b> Todos los requeridos <b>No funcionales:</b> Todos
OP2	<b>OCPR2:</b> Solucionar los incidentes de prioridad “Crítica” y “Alta” al menos un 100% y los de prioridad “Baja” y “Media” en un 70%.	<b>Funcionales:</b> Todos los requeridos
OP2	<b>OCPR3:</b> Tener un porcentaje de cobertura de pruebas unitarias que supere el 80% en el paquete de lógica de negocios.	<b>Funcionales:</b> Todos los requeridos

Tabla 21: Objetivos de calidad

El objetivo general del proyecto “OP3 - Entregar documento adicional al cliente” se omitió del plan de calidad ya que nos resultó difícil de medir puesto que al documento se le fueron agregando secciones y realizando modificaciones.

### 8.2.2 Del proyecto

Objetivo general asociado	Objetivo	Motivación
OA1, OA2	<b>OCPY1:</b> Tener indicadores para evaluar si el avance de desarrollo del producto es el adecuado.	Si la velocidad de desarrollo disminuye es posible que se generen desvíos relacionados con el tiempo. Si se tienen indicadores que miden la velocidad del desarrollo, se pueden tomar acciones correctivas para lograr la construcción del producto final en el tiempo definido.
OA1, OA2	<b>OCPY2:</b> Tener indicadores para detectar desviaciones entre lo solicitado por el cliente y lo comprendido por el equipo de desarrollo.	Se deben detectar dichas desviaciones conceptuales para tomar acciones correctivas en caso de tener diferencias entre lo solicitado y lo entendido.

Tabla 22: Objetivos del proyecto

### 8.3 Aseguramiento de la calidad (SQA)

Se muestran en la tabla los atributos definidos por el equipo según su importancia para asegurar la calidad y así lograr los objetivos descritos anteriormente.

Atributo	Motivación	Objetivo/s de calidad asociado/s
<b>SQA1:</b> Usabilidad	Dado que se trata de una plataforma para un usuario consumidor, es importante que la misma sea intuitiva y fácil de usar.	OCPR1
<b>SQA2:</b> Extensibilidad	La idea es que la empresa pueda agregar cualquier cambio en la lógica. Por lo tanto es imprescindible que el sistema pueda extenderse o cambiarse fácilmente.	OCPR1
<b>SQA3:</b> Seguridad	Es importante tratar los datos de forma adecuada para evitar posibles ataques informáticos.	OCPR1

<b>SQA4:</b> Complejidad de código	El código de backend será una base para que WareUP pueda modificarlo según nuevas lógicas de negocio a futuro. Por lo tanto se apuntará a que se cumplan ciertos estándares definidos por el equipo.	OCPR3
<b>SQ5:</b> Correctitud	El equipo se compromete a entregar un producto final que no tenga <i>bugs</i> críticos ni altos y que tenga una baja cantidad de <i>bugs</i> medios o bajos.	OCPR2
<b>SQA6:</b> Complejidad funcional	El equipo se compromete a entregar un producto con todos los requerimientos funcionales requeridos por el sistema.	OCPR1, OCPY2
<b>SQA7:</b> Complejidad técnica	Se debe contar con un control para evaluar el cumplimiento de los requerimientos no funcionales definidos para el producto final.	OCPR1
<b>SQA8:</b> Cumplimiento de actividades definidas por ORT	Si bien este no es un atributo de calidad en sí mismo, el equipo debe cumplir ciertas fechas que fueron estipuladas por la Universidad ORT para lograr un proyecto exitoso.	OCPY1

Tabla 23: Atributos de SQA

Tanto las actividades como las métricas que se describen en las siguientes secciones están relacionadas con los atributos y los objetivos de calidad definidos.

## 8.4 Principales actividades de SQA

Con el fin de cumplir los objetivos planteados, se realizaron varias actividades que se definen a continuación:

Atributo	Actividad	Motivación
<b>SQA1:</b> Usabilidad	<b>ACT01</b> - Analizar otras web similares	Permite identificar patrones en común e identificar aspectos que puedan ser de valor para el usuario final.
	<b>ACT02</b> - Realizar un prototipo de la web	Tener una idea del producto a construir validada por el cliente.
	<b>ACT03</b> - Realizar "User Persona"	Permite identificar los diferentes tipos de clientes que pueden hacer uso del sitio.
	<b>ACT04</b> - Seguir heurísticas de Nielsen	Seguir heurísticas de usabilidad reconocidas para el diseño de las interfaces de usuario.
<b>SQA2:</b> Extensibilidad	<b>ACT05</b> - Usar patrones de diseño que aporten	Implementar el desarrollo utilizando patrones de diseño para que se pueda agregar

	a la extensibilidad	comportamiento de manera fácil.
<b>SQA3:</b> Seguridad	<b>ACT06</b> - Aportar seguridad de datos a nivel de base	Las contraseñas de los usuarios no deben estar en claro en la base de datos.
	<b>ACT07</b> - Manejar el concepto de identificación de usuarios	Exigir a los usuarios registrarse en el sistema, aportando un grado de seguridad a los datos del usuario.
<b>SQA4:</b> Complejidad de código	<b>ACT08</b> - Utilizar estándares de codificación	Seguir estándares de codificación para tener un código claro y de calidad.
	<b>ACT09</b> - Revisar código periódicamente	Detectar errores de codificación para tener un código claro y de calidad.
	<b>ACT10</b> - Programación de a pares	Asegurar que el código sea revisado por más de un programador y mejorarlo durante su implementación.
<b>SQA5:</b> Correctitud	<b>ACT11</b> - Desarrollar pruebas unitarias	Construir un producto con la menor cantidad de errores, así como también detectar posibles errores luego de un cambio.
	<b>ACT12</b> - Armar y ejecutar casos de pruebas funcionales.	Controlar el correcto funcionamiento de cada uno de los requerimientos implementados.
<b>SQA6:</b> Complejidad funcional	<b>ACT13</b> - Planificar actividades	Evitar desvíos entre lo solicitado por el cliente y lo implementado.
<b>SQA7:</b> Complejidad técnica	<b>ACT14</b> - Definir herramientas de programación a utilizar junto con su versión.	Definir herramientas a utilizar teniendo en cuenta la preferencia del cliente.
	<b>ACT15</b> - Armar un diseño arquitectónico, de base de datos que aporte seguridad y robustez.	Aportar mantenibilidad y seguridad al sistema.
	<b>ACT16</b> - Definir aspectos no funcionales de diseño que sean transversales a toda la aplicación	Definir aspectos de usabilidad y la paleta de colores en conjunto con el cliente.
<b>SQA8:</b> Cumplimiento de actividades definidas por	<b>ACT17</b> - Planificar <i>sprints</i> definiendo un objetivo	Definir objetivos con el fin de reforzar la capacidad para próximos <i>sprints</i> en caso de ser necesario.

ORT		
-----	--	--

Tabla 24: Principales actividades de SQA

### 8.4.1 Detalle de actividades de SQA

A continuación se detalla en qué consistió cada una de las tareas definidas para asegurar la calidad:

#### **SQA1: Usabilidad - OCPR1**

La usabilidad es un atributo difícil de medir antes de tener una web corriendo en ambientes productivos, dado que en ese escenario es que se pueden estudiar aspectos como las malinterpretaciones que puede tener el usuario, el no entendimiento de la navegación o mensajes de error mostrados, entre otros. Dado que esto no es posible, se implementaron otras alternativas para evaluar este aspecto.

#### **Actividades asociadas:**

##### **ACT01 - Analizar otras web similares**

En la etapa de extracción y análisis inicial de requerimientos se destinaron varias horas para analizar varias páginas web similares a la de WareUP con el fin de recabar información de los patrones en común que tienen las mismas. El detalle de este análisis se encuentra en la sección [3.2.1 Estudio de las soluciones existentes](#).

##### **ACT02 - Realizar un prototipo de la web**

Luego de analizar otras páginas web se realizaron prototipos iniciales mediante la aplicación Figma [19] y existieron reuniones con el cliente en donde se compartieron estos prototipos, se validaron y se tomó el feedback del cliente. Los prototipos se presentan en el [Anexo 3 - Prototipos iniciales](#)

##### **ACT03 - Realizar “User Persona”**

Se utilizó la técnica de *User Persona*, con el fin de crear dos personajes ficticios basados en los clientes que representan a los usuarios objetivos de la página de

WareUP para comprender mejor al público objetivo y en consecuencia, tomar decisiones de diseño.

Se construyeron dos tipos de personajes ficticios, uno que representa a los usuarios que desean publicar sus espacios libres para ser arrendados (Gustavo Ripol), y por otro lado, uno que representa a los usuarios que desean arrendar (Sebastian López).

**Gustavo Ripol**  
55 años

**DEMOGRAFÍA**

Edad: 55 años  
Profesional, Abogado y dueño de un depósito.  
Vive en Montevideo - Carrasco

**TECNOLOGÍA:**

Utiliza su smartphone diariamente para trabajar y comunicarse principalmente por asuntos laborales. No utiliza redes sociales. La computadora la utiliza solo si la necesita, principalmente acude a su celular.

**DESCRIPCIÓN:**

Me considero una persona que esta siempre adaptándose a los nuevos cambios tecnológicos. Me gusta trabajar pero disfruto mucho de tener tiempo libre para pasar con mi familia.

**PERSONAL:**

Casado con 3 hijos. Vive en familia en una casa en el barrio Carrasco.  
Si bien ejerce como abogado, tiene un depósito sin uso para ofrecer espacio libre.

**MOTIVACIONES:**

Tiene el depósito sin uso hace mas de 6 meses y entiende que al ofrecerlo podría beneficiarse tanto él como el que lo alquila.

**OBJETIVO**

Tiene un depósito vacío y quiere encontrar una forma de darle uso y sacarle ganancia.

**ESCENARIO:**

Entra a la página de Ware Up para ofrecer espacio ocioso.

**FRUSTRACIONES:**

No tener un punto de contacto con personas que necesiten espacio. Utiliza el boca en boca, eso lo cansa además de que le trae pocas oportunidades.

Ilustración 82: User persona - Gustavo Ripol



Ilustración 83: User Persona - Sebastian Lopez

## ACT04 - Seguir heurísticas de Nielsen

Al momento de desarrollar el *frontend* el equipo decidió basarse en algunas heurísticas de Nielsen [12]:

### 1. Visibilidad del estado del sistema

El usuario debe entender de manera natural toda acción siendo capaz de comprender el flujo de la web. Es fundamental que el usuario sepa en qué paso está al cargar un nuevo depósito en el sistema, como también tener un feedback de que su registro fue completado con éxito.

### 2. Coincidencia entre el sistema real y el mundo real

A veces sucede que los títulos, las disposiciones o los distintos tipos de mensaje no son comprensibles para un usuario que nunca ingresó a la web.

Es importante ser conscientes de que para que la web resulte atractiva para los usuarios, los textos, y la disposición de los componentes deben ser fáciles e intuitivos.

### 3. Estándares y consistencia

Para la consistencia del sistema, es importante que el diseño sea intuitivo en términos de botones, links, imágenes clickeables, etc. logrando que el usuario sepa dónde hacer click y no dedique tiempo en entender cómo funciona la web.

#### 4. Estética y diseño minimalista

No sobrecargar la Web de mucha información que sea sencilla pero que tenga todo lo que el usuario necesita. Para esto, trabajaremos en conjunto con el equipo de **Comunicación** del cliente donde nos ayudará a mejorar la experiencia de *UX* de la Web.

#### 5. Ayudar al usuario a reconocer, diagnosticar y recuperarse de los errores

Se debe informar al usuario en lenguaje natural y comprensible cualquier situación de error que le impida realizar una acción o que impida que visualice determinados datos en la web.

#### 6. Ayuda y documentación

Brindar ayuda a los usuarios a tener el mejor entendimiento posibles de las distintas acciones a realizar dentro de la Web y cómo sacar el mejor resultado posible de las mismas. Además, brindar la posibilidad de tener una guía intuitiva a seguir en caso de generarse un error.

### **SQA2 Extensibilidad - OCPR1**

#### **ACT05 - Usar patrones de diseño que aporten a la extensibilidad**

En la investigación inicial del proyecto, y el análisis de las plataformas existentes, al ver distintas lógicas de negocio se detectaron puntos que deberían ser extensibles, como por ejemplo poder agregar más filtros.

En estos casos se buscó que la implementación tenga las características necesarias con el uso de patrones que promuevan la extensibilidad, permitiendo agregar comportamiento de manera fácil a futuro sin modificar el actual.

### **SQA3: Seguridad - OCPR1**

#### **ACT06 - Aportar seguridad de datos a nivel de base**

Se evalúan distintos métodos de hashing para que las contraseñas se almacenen hasheadas en la base de datos. Se observa el resultado en la sección [8.6 Evolución y análisis de métricas: MPR2 - Seguridad de contraseñas](#)

### **ACT07 - Manejar el concepto de identificación de usuarios**

En el registro de usuarios, se solicitan datos de seguridad como el mail y la contraseña, que deberá cumplir con determinados aspectos de seguridad.

### **SQA4: Complejidad del código - OCPR3**

### **ACT08 - Utilizar estándares de codificación**

Se desarrolla siguiendo estándares de codificación definidos por el equipo que aporten a la facilidad de la lectura y mantenibilidad del mismo. Se tomaron algunas de las prácticas recomendadas por Clean Code.

Prácticas principales:

- Nombres mnemotécnicos para variables y métodos.
- Nombres de variables comienzan en minúscula, a menos que se trate de una propiedad. Además, son siempre en singular.
- Nombres de métodos comienzan en mayúscula.
- Nombres con más de una palabra implican uso de CamelCase.
- Nombres de variables no incluyen el tipo de variable.
- Métodos hacen únicamente lo que su nombre indica y no reciben una gran cantidad de parámetros. Es preferible que reciban un objeto.
- Los TEST están claramente divididos en tres partes: CONSTRUCCIÓN - OPERACIÓN - VALIDACIÓN.
- En lo posible, un único ASSERT por test, a no ser que para probar un concepto se precisen más de uno.

Por otro lado, para asegurar la calidad del código se utilizó la metodología *BDD* para parte del proceso de desarrollo, esta metodología permite a todos los implicados, no

solo el equipo de desarrollo, entender el proceso desarrollado y el contenido del código fuente. Se describe en detalle en la sección [5.2.3 Metodología BDD](#) la forma en la que fue implementada.

#### **ACT09 - Revisar código periódicamente**

El objetivo de las revisiones periódicas es detectar errores y corregirlos lo más temprano posible. Aquellos bloques de código que no cumplen con los estándares de codificación planteados anteriormente, son modificados.

#### **ACT10 - Programación de a pares**

En esa práctica, dos programadores trabajan en pareja en una sola máquina resolviendo el mismo problema. Asegura que todo el código sea revisado por más de un programador y da como resultado un mejor diseño, mejores pruebas, y mejor código.

#### **SQA5: Correctitud - OCPR2, OCPR3**

#### **ACT11 - Desarrollar pruebas unitarias**

Con el objetivo de disminuir la cantidad de errores que se pueden dar en la plataforma y promover la mantenibilidad se desarrollan pruebas unitarias en el paquete de lógica. Ayudan a detectar de manera temprana si los cambios que se van realizando en el código generan nuevos errores o no. El resultado de las pruebas se detalla en la sección [8.6 Evolución y análisis de métricas: MPR5 - Porcentaje de cobertura de pruebas unitarias](#).

#### **ACT12 - Armar y ejecutar casos de pruebas funcionales.**

Para cada uno de los requerimientos desarrollados se procede con el armado de casos de pruebas funcionales para corroborar su correcto funcionamiento.

Dichas pruebas luego se ejecutan con el fin de detectar incidentes que no hayan sido detectados durante la ejecución de las pruebas unitarias. Cada uno será catalogado con severidad Crítica, Alta, Media o Baja.

Todos los casos de prueba funcionales que se implementaron se detallan en el [Anexo 5 - Casos de pruebas funcionales](#).

#### **SQA6: Completitud funcional - OCPR1, OCPY2**

### **ACT13 - Planificar actividades**

Tal como se detalla en [5 Gestión de Proyecto](#), se decidió utilizar *scrum* para la gestión. En este marco de trabajo se dispone de un *backlog* donde se encuentran las tareas a realizar del proyecto.

Existe una planificación inicial, donde se intenta de armar un plan ideal. Sin embargo, en las reuniones de planificación de cada *sprint* el equipo define qué tareas serán llevadas a cabo en esa iteración teniendo en cuenta la capacidad del equipo, la estimación de horas de cada uno de ellas y el tiempo restante para la finalización del proyecto. El cronograma inicial se observa en la sección [5.4.3.1 Cronograma detallado](#).

### **SQA7: Completitud técnica - OCPR1**

#### **ACT14 - Definir herramientas de programación a utilizar junto con su versión.**

En el inicio del proyecto, se definieron junto con el cliente las herramientas y versiones a utilizar tanto para programar como para modelar la base de datos. Se optó por herramientas OpenSource. Las herramientas se listan en la sección [4.4 Herramientas y tecnologías de desarrollo](#).

#### **ACT15 - Armar un diseño arquitectónico, de base de datos que aporte seguridad y robustez.**

Se definió el diseño de la arquitectura del sistema y de la base de datos aplicando patrones y buenas prácticas que aportan a la seguridad del sistema, así como también a su mantenibilidad. Todo lo relacionado a la arquitectura del sistema se puede leer en la sección [4 Diseño arquitectónico](#).

#### **ACT16 - Definir aspectos no funcionales de diseño que sean transversales a toda la aplicación**

En las reuniones llevadas a cabo con el cliente se detallaron aspectos no funcionales que refieren al diseño de la web como por ejemplo la paleta de colores a utilizar.

### **SQA8: Cumplimiento de actividades definidas por ORT**

#### **ACT17 - Planificar sprints definiendo un objetivo**

Durante la planificación de cada *sprint* se define un objetivo general a seguir para ese *sprint*. De esta forma, al finalizar el *sprint*, en la actividad de retrospectiva se puede ver si el equipo logró cumplir el objetivo. En caso de no haber cumplido el objetivo se deberá reforzar la capacidad para el próximo *sprint*, para que el avance del desarrollo sea el adecuado. En el [Anexo 10 - Planificación y resultado de los sprints](#) se muestra el detalle de trabajo en cada *sprint*.

## 8.5 Definición de métricas

Las actividades descritas anteriormente sirven de apoyo para cumplir con las métricas definidas y así lograr los objetivos de calidad planteados. A continuación, se definen las métricas para cada objetivo de calidad.

### 8.5.1 Asociados a objetivos de producto

#### 8.5.1.1 OCPR 1

**Objetivo OCPR1:** Lograr un mínimo producto viable que cumpla con los requerimientos funcionales y no funcionales.

#### **MPR1 - Grado de satisfacción del cliente final**

<b>ID</b>	MPR1
<b>Nombre</b>	Grado de satisfacción del usuario final
<b>Atributo asociado</b>	SQA1 - Usabilidad
<b>Información que brinda</b>	Que tan usable es el sistema desde la perspectiva del usuario
<b>Objetivo</b>	Generar un producto final que sea aceptado por usuarios que utilicen el servicio de la web.
<b>Frecuencia de medición</b>	Al final del proyecto
<b>Herramienta de cálculo</b>	Encuesta - formulario de google
<b>Unidad</b>	%
<b>Meta</b>	Lograr como mínimo un 80% de encuestas positivas.

Tabla 25: Grado de satisfacción del cliente final

#### **MPR2 - Seguridad de contraseñas**

<b>ID</b>	MPR2
<b>Nombre</b>	Seguridad de contraseñas
<b>Atributo asociado</b>	SQA3 - Seguridad
<b>Información que brinda</b>	Seguridad de las contraseñas
<b>Objetivo</b>	Detectar si a nivel de base de datos las contraseñas se guardan de forma segura
<b>Frecuencia de medición</b>	Al final del proyecto
<b>Herramienta de cálculo</b>	Base de datos
<b>Unidad</b>	Booleano: Si están hasheadas - Sí, de lo contrario - No
<b>Meta</b>	Las contraseñas se encuentran hasheadas a nivel de base de datos

Tabla 26: Seguridad de contraseña

### MPR3 - Cumplimiento de los requerimientos funcionales

<b>ID</b>	MPR3
<b>Nombre</b>	Cumplimiento de los requerimientos funcionales
<b>Atributo asociado</b>	SQA6 - Completitud funcional
<b>Información que brinda</b>	Cumplimiento de requerimientos funcionales
<b>Objetivo</b>	Todos los requerimientos funcionales requeridos deben de estar implementados.
<b>Frecuencia de medición</b>	Al final del proyecto
<b>Herramienta de cálculo</b>	Manual
<b>Unidad</b>	Númerica
<b>Meta</b>	Cantidad de requerimientos funcionales requeridos = Cantidad de requerimientos funcionales implementados

Tabla 27: Cumplimiento de los requerimientos funcionales

#### 8.5.1.2 OCPR 2

**Objetivo OCPR2:** Solucionar todos los incidentes de prioridad 'Crítica' y 'Alta' y al menos un 70% de los de prioridad 'Media' y 'Baja'.

### MPR7 - Porcentaje de corrección de incidentes detectados

<b>ID</b>	MPR4
<b>Nombre</b>	Porcentaje de corrección de incidentes detectados
<b>Atributo asociado</b>	SQA5 - Correctitud
<b>Información que brinda</b>	Porcentaje de corrección de incidentes detectados durante los <i>sprints</i> .
<b>Objetivo</b>	Entregar un producto sin incidentes graves y con una baja cantidad de incidentes de prioridad media y baja.
<b>Frecuencia de medición</b>	Al finalizar cada sprint
<b>Herramienta de cálculo</b>	Manual
<b>Unidad</b>	%
<b>Meta</b>	Incidentes Críticos o Altos corregidos: 100% del total de reportados Incidentes Medios o Bajos corregidos: 80% del total de reportados

Tabla 28: Porcentaje de corrección de incidentes detectados

### 8.5.1.3 OCPR 3

**Objetivo OCPR3:** Tener un porcentaje de cobertura de pruebas unitarias que supere el 80% en el paquete de lógica de negocios.

#### **MPR8 - Porcentaje de cobertura de pruebas unitarias**

<b>ID</b>	MPR5
<b>Nombre</b>	Porcentaje de cobertura de pruebas unitarias en la capa de lógica de negocio
<b>Atributo asociado</b>	SQA5 - Correctitud
<b>Información que brinda</b>	Porcentaje de cobertura de pruebas unitarias
<b>Objetivo</b>	Detectar desvíos en el testing unitario y proponer planes de acción.
<b>Frecuencia de medición</b>	En cada Pull Request
<b>Herramienta de cálculo</b>	Ejecución de unitTest
<b>Unidad</b>	%
<b>Meta</b>	Porcentaje de cobertura de pruebas unitarias mayor que el 80% en todos los paquetes

Tabla 29: Porcentaje de cobertura de pruebas unitarias

## 8.5.2 Asociados a objetivos de proyecto

### 8.5.2.1 OCPY1

**Objetivo OCPY1:** Tener indicadores para evaluar si el avance de desarrollo del producto es el adecuado.

#### MPY1 - Cumplimiento de objetivos de *sprint*

<b>ID</b>	MPY1
<b>Nombre</b>	Cumplimiento de objetivos de <i>sprint</i>
<b>Atributo asociado</b>	SQA8 - Cumplimiento de actividades definidas
<b>Información que brinda</b>	Desvíos de metas propuestas en <i>sprints</i>
<b>Objetivo</b>	Lograr que los objetivos del sprint se cumplan en un alto porcentaje para evitar esfuerzos mayores en los siguientes.
<b>Frecuencia de medición</b>	Al finalizar cada sprint
<b>Herramienta de cálculo</b>	Actividades de <i>sprints</i> . <u>Cálculo:</u> $100 - (\text{Cantidad de elementos realizados} / \text{Cantidad de elementos estimados} * 100)$
<b>Unidad</b>	%
<b>Meta</b>	Lograr un porcentaje de desvío menor que 20%

Tabla 30: Cumplimiento de objetivos del sprint

### 8.5.2.2 OCPY2

**Objetivo OCPY2:** Tener indicadores para detectar desviaciones entre lo solicitado por el cliente y lo comprendido por el equipo de desarrollo.

#### MPY2 - Grado de satisfacción de WareUP

<b>ID</b>	MPY2
<b>Nombre</b>	Grado de satisfacción de WareUP
<b>Atributo asociado</b>	SQA6 - Completitud funcional
<b>Información que brinda</b>	El grado de satisfacción de WareUP con el proyecto

<b>Objetivo</b>	Generar un producto final que sea aceptado por WareUP
<b>Frecuencia de medición</b>	Al final del proyecto
<b>Herramienta de cálculo</b>	Formulario de google - Encuesta (Se brindará aplicación al cliente para que realice pruebas y de <i>feedback</i> )
<b>Unidad</b>	%
<b>Meta</b>	Lograr como mínimo un 85% de respuestas positivas en encuesta

Tabla 31: Grado de satisfacción de WareUP

## 8.6 Evolución y análisis de métricas

### 8.6.1 Producto

#### 8.6.1.1 OCPR1

**Objetivo OCPR1:** Lograr un mínimo producto viable que cumpla con requerimientos funcionales y no funcionales.

#### **MPR1 - Grado de satisfacción del usuario final**

El grado de satisfacción del usuario final fue medido mediante una encuesta realizada al final del desarrollo. Fue destinada a **diez usuarios** siendo tres de ellos pertenecientes a la empresa WareUP, utilizando la plataforma bajo un usuario backoffice, y luego se le hicieron a siete posibles usuarios clientes.

La encuesta incluyó preguntas de franja etaria para tener información sobre los usuarios que iban a estar completando la encuesta, preguntas relacionadas a la usabilidad de la página, de cómo ven el diseño de la misma, incluso se indicó un ejemplo particular para que todos los usuarios intenten reproducirlo y así entender el nivel de dificultad que esto les concierne. Por otro lado, se evaluó el entendimiento de las distintas funcionalidades que el sistema tiene, si el objetivo de las mismas es claro, si ven alguna mejora a realizar, entre otros aspectos.

Para evaluar el nivel de acuerdo de los usuarios con las situaciones planteadas en la encuesta, se implementó una escala del **1 al 5**, que abarcaba desde "**Nada de acuerdo**" (equivalente a 1) hasta "**Muy de acuerdo**" (equivalente a 5).

En base a los resultados obtenidos, podemos concluir que es **positiva** la valoración de los usuarios, ya que en la mayoría de las preguntas el porcentaje de respuesta positiva fue superior a las respuestas negativas.

Sin embargo, hay algunos casos a destacar donde podemos continuar trabajando y así seguir mejorando la experiencia de los usuarios:

Algunas funcionalidades resultan complejas de utilizar

10 respuestas

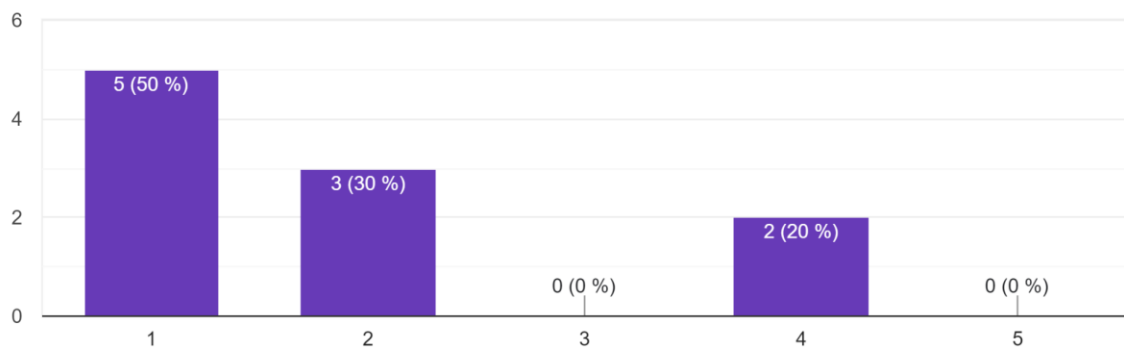


Ilustración 84: Respuesta de encuesta de grado de satisfacción del usuario final

Un **20%** de los usuarios respondió que algunas funcionalidades pueden ser complejas. A raíz de esto, podemos sacar accionables donde se incluyan más textos descriptivos o íconos de ayuda con breves descripciones de la acción que se espera por parte del usuario. Entendemos que mantenernos en una mejora continua es fundamental para seguir generando el mejor producto posible.

Los resultados de la encuesta se pueden visualizar en el [Anexo 17 - Resultados de encuesta de satisfacción del usuario final](#).

## **MPR2 - Seguridad de contraseñas**

Las contraseñas se encuentran hasheadas a nivel de base de datos por lo que esta métrica fue cumplida.

```

password
$2a$10$WN2yAmQd484zZ4550tO.RebE7zRmOVuJ6F0igU6sVh5...
$2a$10$BwPltaoD91YcA1HfaYrNZORI3x76.S2ZqBLxqBhb5t6...
$2a$10$uTYqe7gmPFaKBqFPhoBw1OG5txFyWojZ0xzkeoaPa/7...
$2a$10$QGt0ZAeHln1lgAvsD88cAu7EuVvj7qokHFMbnu.AhL8...

```

Ilustración 85: Password hasheada en base de datos

### MPR3 - Cumplimiento de los requerimientos funcionales

No se cumplió con esta métrica ya que de los 40 requerimientos definidos, se implementaron 34. Cabe destacar, que los 6 requerimientos que quedaron por fuera, se quitaron del alcance en común acuerdo con el cliente.

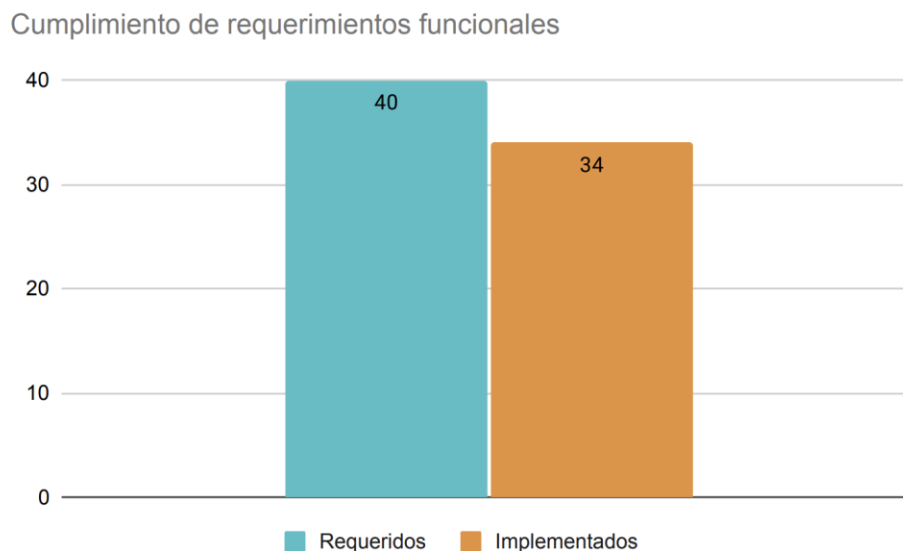


Ilustración 86: Cumplimiento de requerimientos funcionales

#### 8.6.1.2 OCPR2

**Objetivo OCPR2:** Solucionar todos los incidentes de severidad “Crítica” y “Alta” y al menos un 70% de los de severidad “Media” y “Baja”.

### MPR4 - Porcentaje de corrección de incidentes detectados

Como se puede apreciar en la sección [7.5.1.1 Resolución de los incidentes](#), los incidentes con severidad crítica y alta fueron resueltos en un **100%**, incluso cumpliendo con el tiempo de resolución establecido.

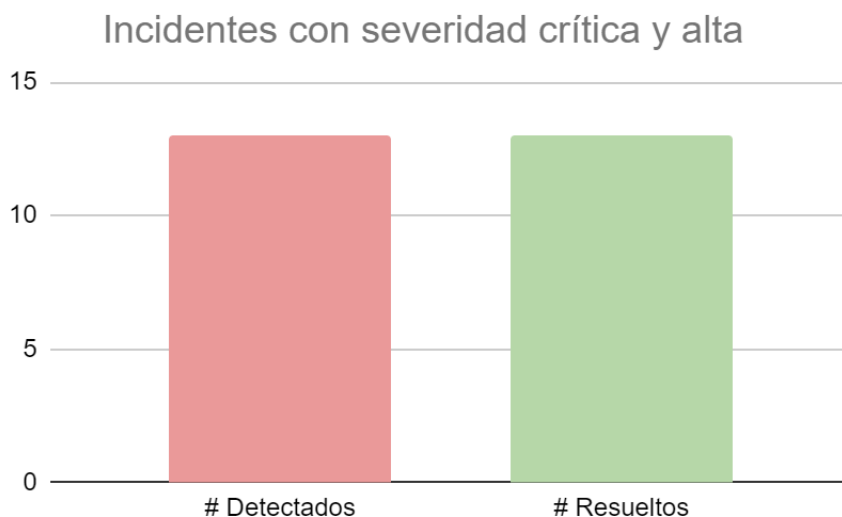


Ilustración 87: Resolución de incidentes con severidad crítica y alta

En cuanto a los incidentes de severidad media y baja, también se cumplió con la resolución del **100%** de los incidentes como se puede apreciar en la siguiente gráfica. Al igual que los incidentes de severidad crítica y alta, el resultado se puede apreciar en la sección [7.5.1.1 Resolución de los incidentes](#).

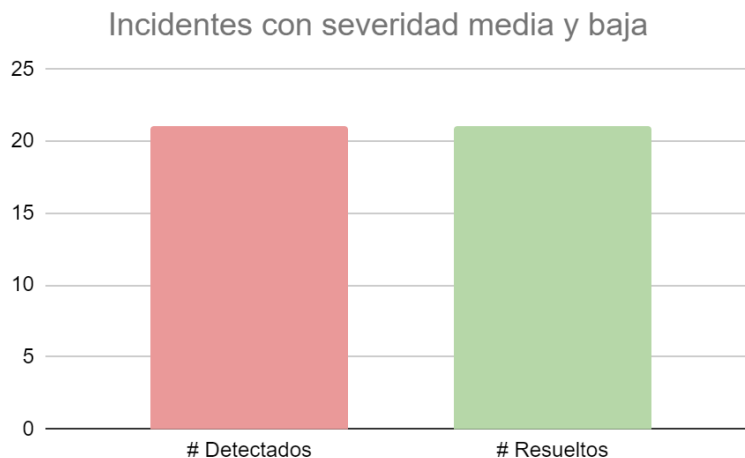


Ilustración 88: Resolución de incidentes con severidad media y baja

### 8.6.1.3 OCPR3

**Objetivo OCPR3:** Tener un porcentaje de cobertura de pruebas unitarias que supere el 80% en paquete *BusinessLogic*.

### MPR5 - Porcentaje de cobertura de pruebas unitarias

File	% Stmts	% Branch	% Funcs	% Lines
All files	85.62	73.23	85.1	85.73
wareupAPI	100	100	100	100
config.js	100	100	100	100
constants.js	100	100	100	100
wareupAPI/src/controllers	87.92	85.71	100	87.92
auth.controller.js	92.85	83.33	100	92.85
bookingRequest.controller.js	84.37	72.22	100	84.37
common.controller.js	88.88	75	100	88.88
company.controller.js	91.89	90.69	100	91.89
deposit.controller.js	80.35	79.59	100	80.35
depositCalendar.controller.js	84	81.81	100	84
depositRequest.controller.js	90.62	89.65	100	90.62
serviceGroup.controller.js	88.88	75	100	88.88
user.controller.js	94.11	93.87	100	94.11
wareupAPI/src/database	95.55	100	50	95.55
index.js	95.55	100	50	95.55
wareupAPI/src/models	97.95	0	94.44	97.95
bookingRequest.model.js	100	100	100	100
city.model.js	100	100	100	100
company.model.js	100	100	100	100
department.model.js	100	100	100	100
deposit.model.js	100	100	100	100
depositCalendar.model.js	100	100	100	100
depositImage.model.js	66.66	0	50	66.66
depositRequest.model.js	100	100	100	100
depositService.model.js	100	100	100	100
log.model.js	100	100	100	100
serviceGroups.model.js	100	100	100	100
services.model.js	100	100	100	100
user.model.js	100	100	100	100
wareupAPI/src/repositories	90.9	50	90.76	90.78
bookingRequest.repository.js	100	100	100	100
city.repository.js	100	100	100	100
company.repository.js	82.35	50	85.71	82.35
department.repository.js	100	100	100	100
deposit.repository.js	100	100	100	100
depositCalendar.repository.js	84	100	83.33	84
depositImage.repository.js	89.47	50	83.33	88.88
depositRequest.repository.js	89.28	100	85.71	89.28
depositService.repository.js	80.95	100	80	80.95
log.repository.js	80	50	100	80
serviceGroup.repository.js	100	100	100	100
user.repository.js	87.09	100	85.71	87.09
wareupAPI/src/services	81.6	47.72	75.29	81.85
auth.service.js	10.52	0	16.66	10.52
bookingRequest.service.js	82.05	50	85.71	82.05
common.service.js	100	100	100	100
company.service.js	85.45	50	100	85.45
deposit.service.js	82.69	40.9	68.42	83.09
depositCalendar.service.js	88.17	71.42	100	88.17
depositRequest.service.js	93.02	50	100	93.02
mail.service.js	36	100	30	36
serviceGroup.service.js	93.93	100	71.42	100
user.service.js	91.66	80	100	91.66

Ilustración 89: Cobertura de Pruebas unitarias

En la imagen se pueden observar la cobertura de las pruebas en las diferentes capas del proyecto.

Si bien, el requerimiento estaba asociado a la capa de negocios, se extendieron las pruebas unitarias a las capas *controller* y *repository*, de cada a asegurar un mayor nivel de calidad.

Por cada clase, se incorporó una clase de test asociada a ella donde se generó, como mínimo, un *unitTest* por cada operación existente con resultado satisfactorio y uno con resultado de error.

Para más detalle sobre las pruebas realizadas, ver [Anexo 12 - UnitTest a nivel de backend](#).

## 8.6.2 Proyecto

### 8.6.2.1 OCPY1

**Objetivo OCPY1:** Tener indicadores para evaluar si el avance de desarrollo del producto es el adecuado.

#### MPY1 - Cumplimiento de objetivos de sprint

#### Requerimientos estimados vs implementados

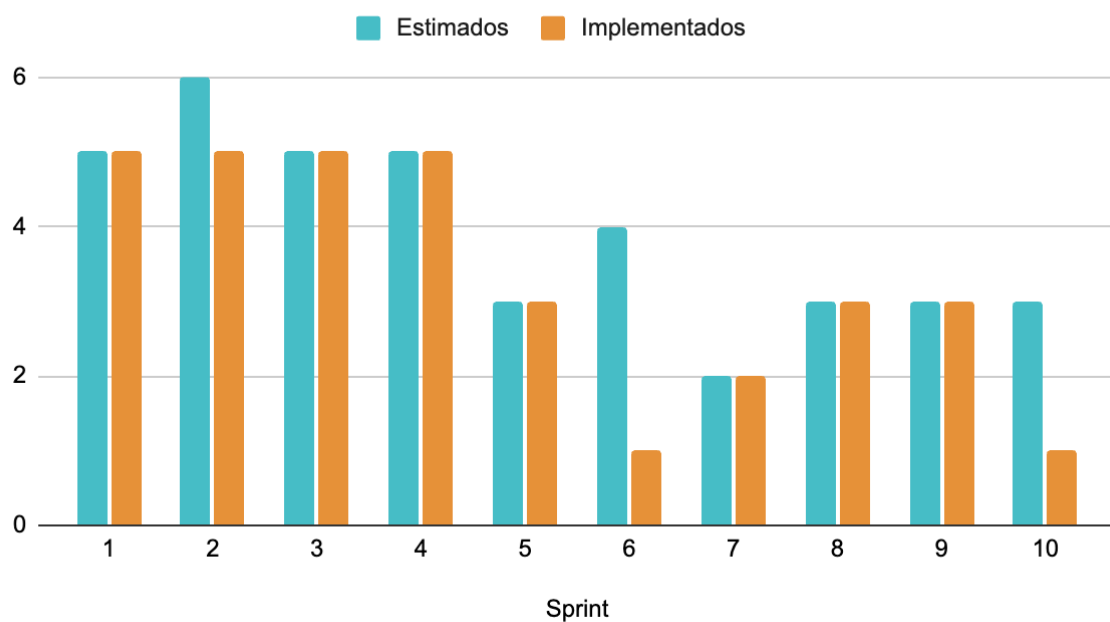


Ilustración 90: Gráfica Requerimientos estimados vs implementados

En la ilustración se puede observar que existieron los siguientes atrasos:

- En el sprint 2 del 16,6% -  $(100 - (5/6 * 100))$
- En el sprint 6 del 75% -  $(100 - (1/4 * 100))$
- En el sprint 10 del 66,6% -  $(100 - (1/3 * 100))$

Estos atrasos se encuentran justificados y mencionados en el [Anexo 10 - Planificación y resultados de los sprints](#), a raíz de estos atrasos el equipo realizó ajustes en el próximo sprint para así lograr cumplir con las metas de desarrollo en los posteriores

sprints. Podemos concluir que en los sprint que no se logró cumplir con el objetivo de la métrica son en el 6 y 10 donde superó la meta de **20%**.

#### 8.6.2.2 OCPY2

**Objetivo OCPY2:** Tener indicadores para detectar desviaciones entre lo solicitado por el cliente y lo comprendido por el equipo de desarrollo.

#### **MPY1 - Grado de satisfacción de WareUP**

A lo largo del proyecto el equipo se mantuvo en continua comunicación con el cliente, validando los requerimientos y haciendo los ajustes necesarios.

Al finalizar el *sprint* 10, se mantuvo una reunión donde se hizo una demo de la aplicación final. Luego se proporcionó una encuesta a Andres Fleitas, CEO de la empresa y a Nicolas López, perfil técnico de la empresa, con el objetivo de evaluar el grado de satisfacción con el *MVP*.

La encuesta estuvo compuesta por cuatro preguntas con opciones del 1 al 5, en donde 5 es la opción que indica el mayor grado de satisfacción, y 1 el menor grado de satisfacción. También se incluyeron dos preguntas abiertas sobre qué características les gustaron más y menos del producto.

Todas las respuestas del cliente fueron positivas, por lo que la meta fue cumplida.

El resultado de la encuesta puede consultarse en el [Anexo 18 - Encuesta de grado de satisfacción de WareUP](#).

## 9 Gestión de la configuración

### 9.1 Introducción

Bajo esta sección se encuentran los procesos destinados a asegurar la calidad de los diferentes elementos de la configuración de *software* (ECS), durante todo el desarrollo del proyecto. Detallamos a continuación los distintos ECS identificados.

## 9.2 Herramientas utilizadas

Para la **documentación** utilizamos toda la suite disponible de Google: Google drive para su almacenamiento, y Google Docs y Google Sheet para la creación y mantenimiento de la misma. Esto nos permitió tener toda la documentación centralizada, sincronizada en línea, accesible para todos los miembros del equipo. A su vez nos permitió tener un historial de cambios y un buen manejo de concurrencia.

Para el **código** utilizamos GitHub como servicio en la nube debido a su conocimiento y gran popularidad, permitiendo a su vez la integración con Jira, la herramienta utilizada para la gestión del proyecto.

## 9.3 Repositorio

La **documentación** se almacenó en Google Drive. Dentro de la misma se incluyen informes de avance, minutas de reuniones, documentación técnica y funcional, diagramas de software y arquitectura.

Se siguió la siguiente estructura de carpetas:

- **Alcance:** Se encuentra la documentación relacionada al alcance del proyecto, donde se detallan las funcionalidades del producto, entre otros.
- **Planificación:** Cronogramas.
- **Desarrollo:** Documentación necesaria para el desarrollo del producto.
- **Calidad:** Se encuentran los casos de prueba, resultados de ejecución de los mismos, métricas, reportes de calidad.
- **Reuniones:** Se encuentra toda la información referente a las reuniones, como ser notas y minutas de las mismas.
- **Otros:** Diagramas, información, registro de horas, etc.

El **código** se almacenó en dos repositorios en GitHub.

- wareupAPI: contiene el código del *backend* (API REST)

- wareupFront: contiene todo lo relacionado al *frontend* (React Web Application)

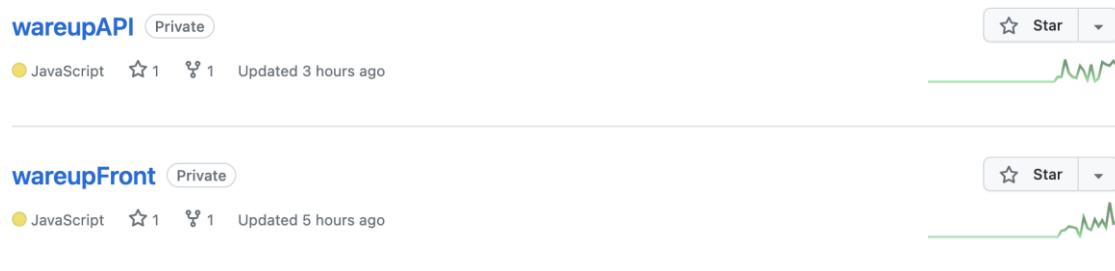


Ilustración 91: Repositorios en GitHub

Se definió a la hora de trabajar una estructura de ramas siguiendo *GitFlow*, una estrategia de branches que implica una rama principal y luego subramas por funcionalidades. Optamos por esta estrategia ya que es ágil, contribuye a la paralelización de los desarrollos y eliminar la dependencia de las ramas, ya que se separan por funcionalidad. Apoya a la metodología ágil, permitiendo la entrega continua de funcionalidades.

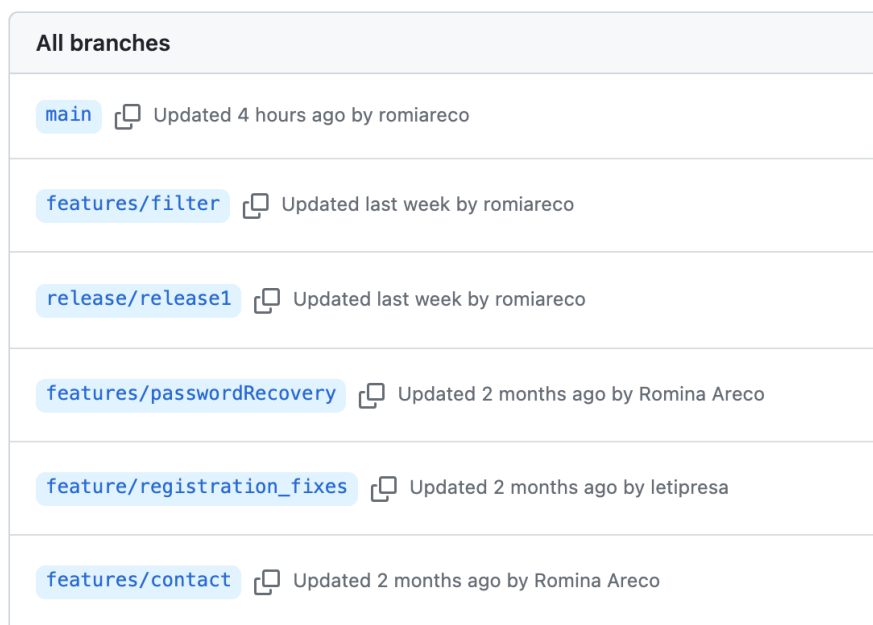


Ilustración 92: Ramas creadas en GitHub

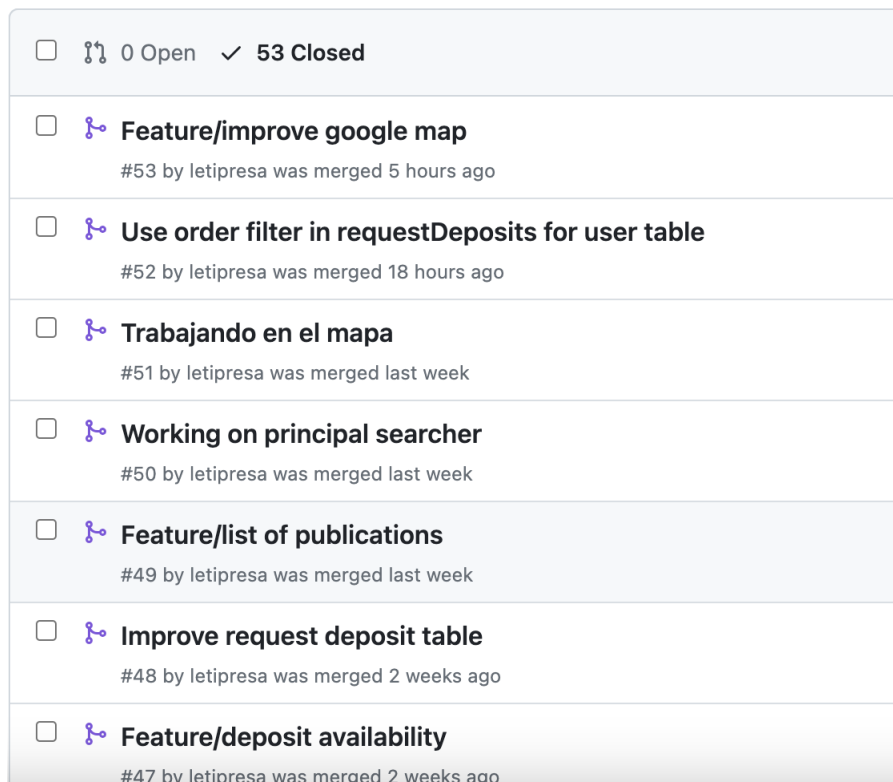


Ilustración 93: Pull Request creados en GitHub

## 9.4 Versionado

El versionado de la **documentación** se resolvió de forma sencilla, al trabajar con documentos separados en fases del proyecto. Para cada documento, cuando se realizó una modificación, se realizó una copia con nombre y fecha actual, de modo de poder recuperar y tener accesible la información anterior.

Para el versionado del **producto** seguimos el estándar *Semantic Versioning*, que consiste en el siguiente formato: `major . minor . patch`

- **Major:** Incluye cambios disruptivos donde se pierde la compatibilidad con las versiones previas.
- **Minor:** Consiste en la incorporación de nuevas funcionalidades. Se sigue manteniendo la compatibilidad con versiones previas.
- **Patch:** Incluye *fixes* sobre *bugs* o cambios mínimos. Se mantiene la compatibilidad.

## 9.5 Ambientes de trabajo

### 9.5.1 Local

Para el desarrollo, cada integrante debía clonar el repositorio localmente y realizar las configuraciones de conectividad, de cara a paralelizar los desarrollos.

### 9.5.2 Testing

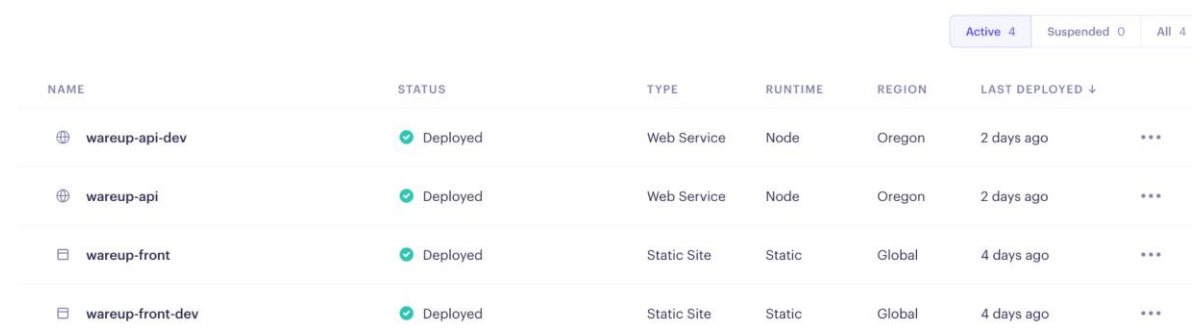
Para validar la integración de los diferentes desarrollos y probar las funcionalidades, se generaron ambientes en Render [20] (**wareup-api-dev** y **wareup-front-dev**), una plataforma en la nube para el deploy de aplicaciones desde GIT, permitiendo el *continuous integration continuous deployment (CI/CD)*.

Respecto a la base de datos, se generó una instancia pública en *db4free* [21], una plataforma gratuita para el hosting de base de datos MySQL.

### 9.5.3 Pre-Prod

Este ambiente se generó también utilizando la plataforma Render [20], para las liberaciones al cliente y su certificación. La misma incluye, además de la automatización de los procesos de entrega, la posibilidad de monitorear los usos de la plataforma y tener visibilidad sobre la participación del cliente en la validación del entregable, al registrarse cada interacción.

El ambiente de *PreProd* también contaba con una base independiente, hospedada en **db4free**.



NAME	STATUS	TYPE	RUNTIME	REGION	LAST DEPLOYED ↓	
wareup-api-dev	Deployed	Web Service	Node	Oregon	2 days ago	...
wareup-api	Deployed	Web Service	Node	Oregon	2 days ago	...
wareup-front	Deployed	Static Site	Static	Global	4 days ago	...
wareup-front-dev	Deployed	Static Site	Static	Global	4 days ago	...

Ilustración 94: Publicaciones creadas en Render

## Usage

The historical usage for this service (up to 31 days).

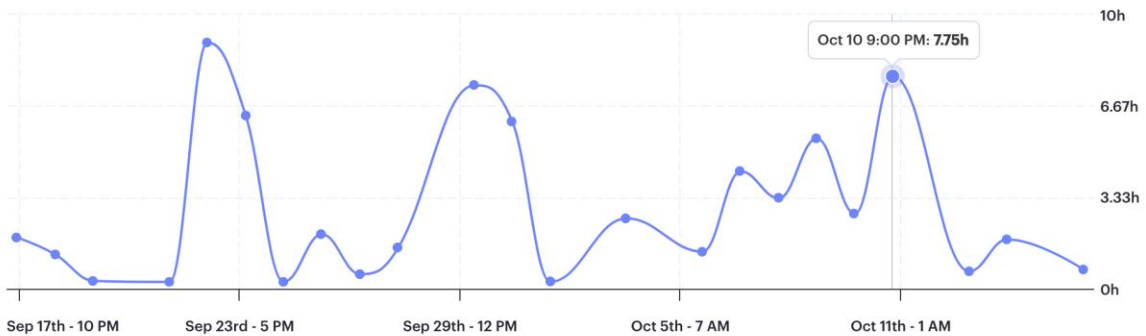


Ilustración 95: Actividad de las publicaciones en Render

### 9.5.4 Producción

El ambiente de producción, tal como se acordó al inicio del proyecto en conjunto con el cliente, queda como responsabilidad de WareUP la creación y publicación del mismo. Como alcance en el proyecto, en relación a este punto, se incluye un análisis sobre *AWS* y *Azure* para la publicación de la plataforma en la nube, donde se evalúan los pros y cons, quedando del lado del cliente la decisión final sobre cuál de ellas utilizar y su posterior implementación.

### 9.6 Despliegue de la aplicación

Se utilizó CI/CD para el despliegue de la solución en los diferentes ambientes, mediante la utilización de Render [20], que permite publicar automáticamente, ramas pre-definidas, en los diferentes ambientes.

WEB SERVICE

**wareup-api-dev** Node Free Connect Manual Deploy

romiareco / wareupAPI main  
<https://wareup-api-dev.onrender.com>

Events

Free instance types will spin down with inactivity. [Upgrade to a paid instance type to prevent this behavior.](#) [Learn more.](#)

Deploy live for c3c0884: Merge pull request #38 from romiareco/feature/fix\_url\_bug Change url for pass...  
August 14, 2023 at 7:04 PM

Deploy started for c3c0884: Merge pull request #38 from romiareco/feature/fix\_url\_bug Change url for pass...  
New commit  
August 14, 2023 at 7:03 PM

Server healthy  
August 10, 2023 at 3:37 PM

Logs  
Disks  
Environment  
Shell  
Previews  
Jobs  
Metrics

Ilustración 96: Eventos de publicación en Render

# 10 Conclusiones y lecciones aprendidas

En este capítulo se presentan las conclusiones y lecciones aprendidas por parte del equipo en este proyecto, tanto desde el punto de vista técnico como humano.

## 10.1 Análisis de resultados según objetivos planteados

### **Objetivos académicos:**

- Aplicar conceptos aprendidos en la carrera en un proyecto real
- Adquirir experiencia en gestión y desarrollo de proyectos

### **Resultados:**

Finalizado el proyecto consideramos que cumplimos con los objetivos planteados. Para el desarrollo del proyecto se aplicaron metodologías y conocimientos aprendidos a lo largo de la carrera. Otro aspecto positivo a considerar es que no solo pudimos aplicar estos conceptos conocidos, sino que también aprendimos acerca de tecnologías que hasta el momento no habíamos utilizado.

Consideramos que adquirimos experiencia en gestión y desarrollo de proyectos y concientizamos la importancia de realizar los distintos planes como el de calidad y riesgos y seguirlos proactivamente para minimizar los impactos negativos y poder controlar el alcance.

Usar como metodología de trabajo *scrum* fue un aspecto muy positivo para el tipo de proceso que se ejecutó. Se adaptó muy bien al ciclo de trabajo del equipo, y permitió tener una buena sincronía de feedback con el cliente. Esto permitió mantener al cliente actualizado sobre el estado del proyecto y poder tener una visibilidad real del avance del producto.

También mencionar que valoramos la experiencia de trabajar en conjunto con el cliente WareUP, quienes en todo momento nos acompañaron y colaboraron en el proceso.

### **Objetivos del producto:**

- Entregar un mínimo producto viable

- Desarrollar software de calidad y extensible
- Entregar documento adicional al cliente

### **Resultados:**

Uno de los objetivos más importantes del proyecto fue lograr un producto de calidad que cumpliera con las expectativas del cliente.

Generamos un MVP amigable con el usuario, de calidad y aceptado por nuestro cliente y los usuarios finales.

Dicho MVP fue desarrollado de manera escalable y cuenta con potencial como para poder crecer y lanzarse en producción si el cliente continúa con el desarrollo.

Se le entregó un documento adicional al cliente, que contiene aspectos del negocio, investigaciones e información acerca de la arquitectura del sistema.

Consideramos que fuimos capaces de planificar y gestionar un producto de software de calidad que aporta valor al cliente y a los usuarios finales.

## **10.2 Conclusiones y lecciones aprendidas nivel de equipo**

Concluimos que logramos consolidar gran parte de los conocimientos y habilidades aprendidas durante la carrera.

Las siguientes son algunas de las lecciones aprendidas en el transcurso del proyecto.

- Importancia de la comunicación para formar equipos y el desarrollo de habilidades blandas
- Definir procesos, acciones y herramientas
- Realizar estimaciones
- Gestionar riesgos, incidentes, alcance y calidad
- Negociar con los clientes

La comunicación entre el equipo fue de gran importancia para el correcto desarrollo del proyecto, no solo entre el equipo sino, con el cliente. Se logró negociar en todo momento e intercambiar perspectivas sobre lo que se esperaba obtener como producto final. Sobre este punto concluimos que un equipo con una comunicación transparente, donde predomine el respeto, la escucha y la claridad del mensaje, evita

conflictos y malentendidos así como también aumenta la productividad y eficiencia en las tareas, por lo que es un aspecto que no debe ser subestimado en ningún momento.

En relación a la conformación del equipo, creemos que las fortalezas de cada una fueron un factor diferenciador que fue aprovechado y enriqueció al equipo. A su vez, contar con perfiles tan diferenciados y tener un clima transparente fomentó el trabajo colaborativo en donde cada una pudo exponer sus debilidades y fortalecerlas mediante ayuda de otros.

Fue de vital importancia definir de manera clara la metodología de trabajo a utilizar, detallando las acciones puntuales y herramientas a emplear. Este paso concluimos que es crucial para el correcto desempeño de un equipo y para cumplir con los objetivos y metas que se proponen ya que entendemos permiten dirigir la atención a lo importante.

Se aprendió la importancia de detectar los riesgos de forma temprana, ideando planes de mitigación para prevenir su ocurrencia y planes de contingencia en caso de que un riesgo se convierta en un problema. Realizar reuniones frecuentes para evaluar la aparición de nuevos riesgos y analizar la evolución de los existentes le permite al equipo poner foco en los más problemáticos en cada etapa del proyecto, evitando desperdiciar esfuerzos en riesgos que ya fueron mitigados o que dejaron de ser amenazantes en la etapa actual del proyecto.

También, se tomó conciencia sobre la importancia de generar un plan de calidad, y sobre todo en este proyecto donde el cliente planea continuar con el desarrollo y el mantenimiento de la plataforma. El equipo considera que el seguimiento de métricas de calidad a lo largo del desarrollo facilitó la construcción de un producto robusto. Asimismo la gestión de incidentes aportó transparencia para el cliente con respecto a la situación actual del producto.

## Referencias Bibliográficas

- [1] TechTerms. "API Definition". [Online]. Disponible en: <https://techterms.com/definition/api> [Accedido: 15 de Junio 2023].
- [2] Wikipedia. "Front-end y back-end". [Online]. Disponible en: [https://es.wikipedia.org/wiki/Front-end\\_y\\_back-end](https://es.wikipedia.org/wiki/Front-end_y_back-end) [Accedido: 15 de Junio 2023].
- [3] LeadingAgile "Definition Of Done". [Online]. Disponible en: <https://www.leadingagile.com/2017/02/definition-of-done/> [Accedido: 10 de Octubre 2023].
- [4] Wikipedia. "Front-end y back-end". [Online]. Disponible en: [https://es.wikipedia.org/wiki/Front-end\\_y\\_back-end](https://es.wikipedia.org/wiki/Front-end_y_back-end) [Accedido: 15 de Junio 2023].
- [5] Atlassian, "Flujo de trabajo de git flow," Atlassian. [Online]. Disponible en: <https://www.atlassian.com/es/git/tutorials/comparing-workflows/gitflow-workflow> [Accedido: 1 de Octubre 2023].
- [6] Google, "Material Design". [Online]. Disponible en: <https://m3.material.io/get-started> [Accedido: 19 de Setiembre 2023].
- [7] Wikipedia. "Producto viable mínimo". [Online]. Disponible en: [https://es.wikipedia.org/wiki/Producto\\_viable\\_m%C3%ADnimo](https://es.wikipedia.org/wiki/Producto_viable_m%C3%ADnimo) [Accedido: 4 de Mayo 2023].
- [8] Airbnb, Airbnb. [Online]. Disponible en: <https://es-l.airbnb.com/> [Accedido: 3 de Junio 2023].
- [9] Warehouse Exchange, Warehouse Exchange. [Online]. Disponible en: <https://www.warehouseexchange.com/> [Accedido: 3 de Junio 2023].
- [10] IEEE-Standard-729-1983-Engineering-Terminology. [Online]. Disponible en: <https://standards.ieee.org/ieee/729/967/> [Accedido: 3 de Junio 2023].
- [11] Wikipedia, "Principios S.O.L.I.D". [Online]. Disponible en: <https://es.wikipedia.org/wiki/SOLID> [Accedido: 4 de Junio 2023].

[12] J. Nielsen, "10 Usability Heuristics for User Interface Design". Nielsen Norman Group, 24 abril 1994, actualizado 15 noviembre 2020. [Online]. Disponible en: <https://www.nngroup.com/articles/ten-usability-heuristics/> [Accedido: 25 de Abril 2023].

[13] MUI, Material UI. [Online]. Disponible en: <https://mui.com/> [Accedido: 4 de Abril 2023].

[14] Cucumber, "Cucumber for JavaScript." [Online]. Disponible en: <https://github.com/cucumber/cucumber-js> [Accedido: 10 de Mayo 2023].

[15] Gwynne Richards, "Warehouse Management - A complete guide to improving efficiency and minimizing costs in the modern warehouse". [Online]. Disponible en: <https://shorturl.at/wFUVZ>

[16] David E. Mulcahy, Joachim Sydow, "A Supply Chain Logistics Program for Warehouse Management". [Online]. Disponible en: <https://shorturl.at/eswE8>

[17] Modus Create, "Things Your Burndown Chart is Burning to Tell You". [Online]. Disponible en: <https://moduscreate.com/blog/burndown-chart/>

[18] WareUP, WareUP. [Online]. Disponible en: <https://wareup.com.uy/> [Accedido: 3 de Abril 2023].

[19] Figma. "Figma". [Online]. Disponible en: <https://www.figma.com/> [Accedido: 13 de Abril 2023].

[20] Render [Online]. Disponible en: <https://render.com/> [Accedido: 27 de Mayo 2023].

[21] db4free, db4free.net. [Online]. Disponible en: <https://db4free.net/> [Accedido: 6 de Abril 2023].

# Anexos

## Anexo 1 - Requerimientos iniciales relevados enviados al cliente

A través de este documento se pretende detallar ciertos requerimientos que el equipo relevó como importantes para el desarrollo de la plataforma WareUP. Se espera que sea analizado por el cliente para continuar agregando y modificando los necesarios.

### Requerimientos funcionales

**REF1** - Inicio de sesión de usuario: Un usuario registrado podrá ingresar al sistema utilizando mail y contraseña.

**REF2** - Unicidad de identidad de usuario: Cada usuario será único dentro del sistema.

**REF3** - Recuperación de contraseña: El sistema deberá tener la habilidad de cambiar o recuperar la contraseña de un usuario (se debe hacer mediante envío de un mensaje o un email al usuario).

**REF4** - Autogestión de perfil: El usuario deberá poder gestionar su perfil pudiendo cambiar sus datos personales.

**REF5** - Roles del usuario: El sistema maneja **2** roles de usuarios distintos. Un usuario podrá ser Backoffice o Usuario cliente. Siendo el primero capaz de gestionar todos los usuarios del sistema y gestionar los espacios que se publiquen mientras que el usuario corriente podrá publicar o contratar espacios disponibles.

**REF6** - Acerca de: Generar una opción de menú donde se pueda visualizar más información sobre los creadores de la empresa.

**REF7** - Listado de espacios publicados. El usuario podrá acceder a un listado de todos los espacios disponibles que estén publicados en la plataforma.

**REF8** - Filtros para mejorar la búsqueda de espacios. El usuario podrá disponer de distintos filtros para realizar su búsqueda de espacios de acuerdo a sus necesidades. Definir si desean mantener los filtros actuales o si deberíamos agregar nuevos.

**REF9** - Ordenar resultados de búsqueda. El usuario podrá ordenar los resultados de

la búsqueda de espacios de acuerdo al criterio que le parezca mejor.

Ejemplo: Menor precio/Mayor precio, Más cercano geográficamente/Más alejado geográficamente, etc.

**REF10** - Mapa con las publicaciones en base a la cercanía del usuario. El usuario puede acceder a un mapa donde se muestran todos los espacios de almacenamiento que tenga en un rango aproximado indicando si están disponibles o no.

Este requerimiento al equipo le implica análisis de tecnologías y otras cuestiones, se plantea como un requerimiento que puede ser bueno de implementar pero no es prioritario.

**REF11** - Significado de los filtros. Un usuario podrá pararse sobre un popup y visualizar una breve definición de lo que significa el filtro de búsqueda.

**REF12** - Visualización rápida de un espacio. El usuario puede visualizar de forma sencilla información básica de un espacio publicado donde se muestran sus imágenes, descripción, título y *precio*? Donde en caso de interesarle, puede entrar a la información completa de la publicación.

**REF13** - Visualización completa de una publicación de espacio. El usuario puede seleccionar una publicación del buscador y acceder a la misma para obtener el detalle del espacio de almacenamiento completo.

**REF14** - Comprobar disponibilidad de un espacio. El usuario podrá seleccionar un periodo de tiempo y comprobar si el espacio de almacenamiento se encuentra disponible para ser alquilado o no.

**REF15** - Adquisición de un espacio de almacenamiento. El usuario podrá adquirir el espacio de almacenamiento si éste se encuentra disponible en la fecha indicada.

**REF16** - Registro de empresas. Un usuario puede ser representante de una o más empresas y por ende, tiene la posibilidad de registrar la información de las mismas.

**REF17** - Registro de espacio. Un usuario puede registrar la información del espacio disponible con el que cuenta previamente seleccionando a qué empresa corresponde dicho espacio.

**REF18** - Registro de depósito adicional. Un usuario puede registrar información sobre un depósito adicional, en caso de que exista, a un depósito de almacenamiento previamente registrado o en el momento de registrar el mismo.

**REF19** - Alta de espacio de almacenamiento. Un usuario administrador puede dar de alta un almacenamiento que esté pendiente de aprobación.

**REF20** - Baja de espacio de almacenamiento. Un usuario administrador puede dar de baja o pausar la publicación de un almacenamiento que incumpla alguna normativa o que no se desee mantener en el sitio.

**REF21** - Baja de espacio de almacenamiento propio. Un usuario puede dar de baja sus propias publicaciones de espacios de almacenamiento.

**REF22** - Modificar información del espacio de almacenamiento propio. Un usuario puede modificar la información de sus propias publicaciones de espacios de almacenamiento que luego quedarán pendientes de validación por parte de un usuario administrador.

**REF23** - Validar información modificada de un espacio de almacenamiento existente. Un usuario administrador puede aprobar o rechazar las modificaciones que se intenten hacer sobre un espacio de almacenamiento ya existente en el sistema.

## Anexo 2 - Product Backlog - Versión 1

<b>Título</b>	REF1 - Inicio de sesión de usuario
<b>Narrativa</b>	<p><b>Como</b> usuario registrado en el sistema</p> <p><b>Quiero</b> poder iniciar sesión con mi email y contraseña al sistema registrado</p> <p><b>Para</b> hacer uso de las funcionalidades que el mismo provee.</p>
<b>Criterios de Aceptación</b>	<p><b>Dado</b> un usuario no identificado,  <b>Y</b> el usuario está en la pantalla de iniciar sesión,  <b>Y</b> agrega un Email válido en el campo de email,  <b>Y</b> una Contraseña correcta en el campo de contraseña,  <b>Cuando</b> acepta el ingreso de datos (aceptar)  <b>Entonces</b> el sistema despliega la pantalla de inicio.</p> <p><b>Dado</b> un usuario no identificado,  <b>Y</b> el usuario está en la pantalla de iniciar sesión,  <b>Y</b> agrega un Email no válido en el campo de email,  <b>Y</b> una Contraseña correcta o no en el campo de contraseña,  <b>Cuando</b> acepta el ingreso de datos (aceptar)  <b>Entonces</b> el sistema muestra un error.</p> <p><b>Dado</b> un usuario no identificado,  <b>Y</b> el usuario está en la pantalla de iniciar sesión,  <b>E</b> ingresa un Email válido en el campo de email,  <b>Y</b> una Contraseña incorrecta para el email provisto en el campo de contraseña,  <b>Cuando</b> acepta el ingreso de datos (aceptar)  <b>Entonces</b> el sistema muestra un error.</p>
<b>Prioridad</b>	Alta

Tabla 32: Requerimiento funcional 1

<b>Título</b>	REF2 - Unicidad de identidad de usuario
---------------	---

<b>Narrativa</b>	<p><b>Como</b> usuario no registrado en el sistema</p> <p><b>Cuando</b> estoy en la sección de registro de nuevo usuario <b>y</b> completo el campo de email con un correo ya existente, <b>Entonces</b> me indique que ese email ya se encuentra en uso <b>y</b> debo elegir uno nuevo</p>
<b>Criterios de Aceptación</b>	<p><b>Dado</b> un usuario no identificado,</p> <p><b>Y</b> el usuario está en la pantalla de registrar usuario,</p> <p><b>Y</b> agrega un Email ya registrado en el campo de email,</p> <p><b>Y</b> una Contraseña en el campo de contraseña,</p> <p><b>Cuando</b> presiona el botón “Registrar”,</p> <p><b>Entonces</b> el sistema despliega un mensaje de error.</p>
<b>Prioridad</b>	Alta

Tabla 33: Requerimiento funcional 2

<b>Título</b>	REF3 - Recuperación de contraseña
<b>Narrativa</b>	<p><b>Como</b> usuario ya registrado en el sistema</p> <p><b>Quiero</b> poder cambiar o recuperar mi contraseña en caso de que la haya olvidado</p> <p><b>Para</b> continuar utilizando las funcionalidades del sistema.</p>
<b>Criterios de Aceptación</b>	<p><b>Dado</b> un usuario registrado,</p> <p><b>Y</b> el usuario se encuentra en la pantalla Iniciar Sesión,</p> <p><b>Cuando</b> el usuario selecciona el botón “¿Olvidaste tu contraseña?” <b>e</b> ingresa su correo electrónico</p> <p><b>Entonces</b> el sistema muestra un mensaje de éxito <b>y</b> deberá enviar un email al usuario para recuperar su clave</p> <p><b>Dado</b> un usuario no registrado,</p> <p><b>Y</b> el usuario se encuentra en la pantalla iniciar sesión</p> <p><b>Cuando</b> el usuario selecciona recuperar contraseña e ingresa un email no válido</p> <p><b>Entonces</b> el sistema notificará mediante un mensaje de error que se email no se encuentra registrado</p>

<b>Prioridad</b>	Alta
------------------	------

Tabla 34: Requerimiento funcional 3

<b>Título</b>	REF4 - Autogestión de perfil
<b>Narrativa</b>	<b>Como</b> usuario ya registrado e iniciado sesión en el sistema, <b>Quiero</b> poder editar mi información personal desde mi perfil <b>Para</b> tener mis datos actualizados en caso de ser necesarios.
<b>Criterios de Aceptación</b>	<b>Dado</b> un usuario registrado y logueado en el sistema, <b>y</b> el usuario se encuentra en la pantalla Autogestión de perfil, <b>y</b> decide modificar sus datos, <b>Y</b> selecciona el botón “Guardar”, <b>Entonces</b> el sistema modificará sus datos sin ninguna restricción <b>Y</b> se mostrará un mensaje “Usuario actualizado exitosamente”
<b>Prioridad</b>	Media

Tabla 35: Requerimiento funcional 4

<b>Título</b>	REF5 - Roles del usuario
<b>Narrativa</b>	<b>Como</b> usuario que estoy logueado al sistema <b>Quiero</b> entrar al menú de inicio <b>y</b> acceder a las funcionalidades correspondientes a mi rol de usuario ya sea backoffice o usuario cliente
<b>Criterios de Aceptación</b>	<b>Dado</b> un usuario con rol backoffice logueado en el sistema, <b>Cuando</b> entro a la pantalla de inicio <b>Entonces</b> el sistema me permitirá gestionar todos los usuarios del sistema y gestionar los espacios  <b>Dado</b> un usuario con rol usuario cliente logueado en el sistema, <b>Cuando</b> entro a la pantalla de inicio <b>Entonces</b> el sistema me permitirá publicar o arrendar espacios disponibles.
<b>Prioridad</b>	Media

Tabla 36: Requerimiento funcional 5

<b>Título</b>	REF6 – Niveles de usuario Backoffice
<b>Narrativa</b>	<p><b>Como</b> usuario backoffice con full permisos a todas las funcionalidades del sistema</p> <p><b>Quiero</b> gestionar los permisos del resto de los usuarios backoffice</p> <p><b>Para</b> restringir el acceso a ciertas funcionalidades que no deben ser accedidas por todos por igual.</p>
<b>Criterios de Aceptación</b>	<p><b>Dado</b> un usuario backoffice,  <b>Y</b> con permisos restringidos para únicamente aprobar nuevas publicaciones,  <b>Entonces</b> el sistema solo muestra la posibilidad de aprobar las publicaciones</p> <p><b>Dado</b> un usuario backoffice,  <b>Y</b> con todos los permisos habilitados,  <b>Entonces</b> el sistema le permite hacer las modificaciones y aprobaciones que desee</p>
<b>Prioridad</b>	Media

Tabla 37: Requerimiento funcional 6

<b>Título</b>	REF7 - Logout
<b>Narrativa</b>	<p><b>Como</b> usuario</p> <p><b>Quiero</b> cerrar la sesión en el sistema</p> <p><b>Para</b> poder salir del mismo.</p>
<b>Criterios de Aceptación</b>	<p><b>Dado</b> un usuario identificado y logueado,  <b>Cuando</b> presiona el botón “cerrar sesión”,  <b>Entonces</b> el sistema desloguea al usuario</p> <p><b>Y</b> muestra la pantalla de iniciar sesión</p>
<b>Prioridad</b>	Alta

Tabla 38: Requerimiento funcional 7

<b>Título</b>	REF8 - Registro de usuarios
<b>Narrativa</b>	<p><b>Como</b> usuario cliente</p> <p><b>Quiero</b> registrarme al sistema indicando un email no existente en el sistema, nombre y contraseña</p> <p><b>Para</b> hacer uso de las funcionalidades del sistema</p>
<b>Criterios de aceptación</b>	<p><b>Dado</b> un usuario no identificado,  <b>Y</b> se encuentra en la pantalla “Registrarse”,  <b>Cuando</b> ingresa un email no registrado en el sistema <b>y</b> completa los campos restantes obligatorios <b>y</b> acepta las políticas de privacidad,  <b>Entonces</b> el sistema muestra un mensaje de éxito</p> <p><b>Y</b> el usuario queda registrado</p> <p><b>Dado</b> un usuario no identificado,  <b>Y</b> se encuentra en la pantalla “Registrarse”,  <b>Cuando</b> ingresa un email ya registrado en el sistema <b>y</b> completa los campos restantes obligatorios <b>y</b> acepta las políticas de privacidad,  <b>Entonces</b> el sistema muestra un mensaje de error</p> <p><b>Dado</b> un usuario no identificado,  <b>Y</b> se encuentra en la pantalla “Registrarse”,  <b>Cuando</b> ingresa los datos requeridos <b>y</b> una contraseña que no cumple con los requisitos,  <b>Entonces</b> el sistema muestra un mensaje de error</p>
<b>Prioridad</b>	Alta

Tabla 39: Requerimiento funcional 8

<b>Título</b>	REF9 - Buscador de espacios publicados
<b>Narrativa</b>	<p><b>Como</b> usuario</p> <p><b>Quiero</b> visualizar todos los espacios de almacenamientos disponibles que tiene la Web</p> <p><b>Para</b> arrendar uno en caso de que tenga disponibilidad</p>
<b>Criterios de Aceptación</b>	<p><b>Dado</b> un usuario cliente identificado,  <b>Cuando</b> seleccionó la opción “Buscar depósitos”,</p>

	<b>Entonces</b> el sistema me muestra en pantalla un listado con los espacios disponibles.
<b>Prioridad</b>	Alta

Tabla 40: Requerimiento funcional 9

<b>Título</b>	REF10 - Filtros para mejorar la búsqueda de espacios
<b>Narrativa</b>	<b>Como</b> usuario <b>Quiero</b> poder filtrar por distintas características en el buscador de espacios <b>Para</b> mejorar los resultados en base a mis preferencias.
<b>Criterios de Aceptación</b>	<b>Dado</b> un usuario cliente identificado, <b>Y</b> se encuentra en la pantalla de espacios disponibles, <b>Cuando</b> seleccione un filtro y apriete el botón “Buscar”, <b>Entonces</b> el sistema me muestra únicamente los espacios que cumplen esa condición
<b>Prioridad</b>	Media

Tabla 41: Requerimiento funcional 10

<b>Título</b>	REF11 - Ordenar resultados de búsqueda
<b>Narrativa</b>	<b>Como</b> usuario <b>Quiero</b> ordenar el resultado del buscador de depósitos <b>Para</b> mejorar la búsqueda en base a mis intereses.
<b>Criterios de Aceptación</b>	<b>Dado</b> un usuario cliente identificado, <b>Y</b> se encuentre en la pantalla de espacios disponibles, <b>Y</b> quiera ordenar los espacios según el criterio seleccionado, <b>Cuando</b> apriete el botón “Ordenar”, <b>Entonces</b> el sistema mostrará los espacios ordenados según el criterio seleccionado por el cliente
<b>Prioridad</b>	Baja

Tabla 42: Requerimiento funcional 11

<b>Título</b>	REF12 - Mapa con las publicaciones en base a la cercanía del usuario
<b>Narrativa</b>	<b>Como</b> usuario <b>Quiero</b> poder visualizar los distintos depósitos publicados en un mapa <b>Para</b> tener información de cuáles son los que se encuentran más cercanos a mi ubicación.
<b>Criterios de Aceptación</b>	<b>Dado</b> un usuario cliente identificado, <b>Y</b> se encuentre en la pantalla de espacios disponibles, <b>Y</b> quiera visualizar los espacios en un mapa de forma gráfica, <b>Cuando</b> apriete el botón “Mapa”, <b>Entonces</b> el sistema mostrará en un mapa los espacios disponibles más cercanos al usuario indicando si están o no disponibles <b>y</b> el usuario podrá navegar por el mapa y descubrir el resto de los espacios en la zona que él prefiera.
<b>Prioridad</b>	Baja

Tabla 43: Requerimiento funcional 12

<b>Título</b>	REF13 - Significado de los filtros
<b>Narrativa</b>	<b>Como</b> usuario <b>Quiero</b> leer el nombre de un filtro en el buscador <b>y</b> saber a qué hace referencia para tener un mejor entendimiento <b>y</b> <b>Poder</b> hacer uso de los filtros con mejor sabiduría en mi búsqueda .
<b>Criterios de Aceptación</b>	<b>Dado</b> un usuario cliente identificado, <b>Y</b> en la pantalla de espacios disponibles, <b>Y</b> quiere filtrar la búsqueda, <b>Cuando</b> posicione el cursor sobre determinado filtro, <b>Entonces</b> el sistema mostrará un pop up con el significado de ese filtro.
<b>Prioridad</b>	Baja

Tabla 44: Requerimiento funcional 13

<b>Título</b>	REF14 - Visualización rápida de un espacio de almacenamiento
---------------	--

<b>Narrativa</b>	<p><b>Como</b> usuario registrado</p> <p><b>Quiero</b> ver de forma rápida información básica de un espacio de almacenamiento en el listado que me aparece en el buscador,</p> <p><b>Para</b> tener una breve descripción del mismo y ver si meramente cubre las necesidades que tengo.</p>
<b>Criterios de Aceptación</b>	<p><b>Dado</b> un usuario cliente identificado,</p> <p><b>Y</b> quiera visualizar la información básica de un espacio,</p> <p><b>Cuando</b> se encuentre en la pantalla de espacios disponibles,</p> <p><b>Entonces</b> el sistema mostrará en pantalla los diferentes espacios con la imagen, descripción, título, precio.</p>
<b>Prioridad</b>	Media

Tabla 45: Requerimiento funcional 14

<b>Título</b>	REF15 - Visualización completa de una publicación de espacio de almacenamiento
<b>Narrativa</b>	<p><b>Como</b> usuario,</p> <p><b>Quiero</b> visualizar todas las características e información de interés de un espacio de almacenamiento,</p> <p><b>Para</b> conocer más sobre el mismo.</p>
<b>Criterios de Aceptación</b>	<p><b>Dado</b> un usuario cliente identificado,</p> <p><b>Y</b> se encuentre en la pantalla de espacios disponibles,</p> <p><b>Y</b> quiera visualizar la información completa de un espacio,</p> <p><b>Cuando</b> seleccione el botón “ver más información” de un espacio en particular,</p> <p><b>Entonces</b> el sistema mostrará en pantalla la página de la publicación con toda la información de la misma.</p>
<b>Prioridad</b>	Alta

Tabla 46: Requerimiento funcional 15

<b>Título</b>	REF16 - Comprobar disponibilidad de un espacio de almacenamiento
---------------	--

<b>Narrativa</b>	<p><b>Como</b> usuario registrado</p> <p><b>Quiero</b> validar si un espacio de almacenamiento se encuentra disponible en determinada fecha y determinados metros cuadrados</p> <p><b>Para</b> poder arrendarlo.</p>
<b>Criterios de Aceptación</b>	<p><b>Dado</b> un usuario cliente identificado,</p> <p><b>Y</b> quiera validar si un espacio se encuentra disponible en una determinada fecha y metros cuadrados,</p> <p><b>Cuando</b> se encuentre en la pantalla del espacio deseado</p> <p><b>Y</b> seleccione la fecha en un calendario,</p> <p><b>Y</b> la cantidad de metros cuadrados que necesita,</p> <p><b>Entonces</b> el sistema indicará si el espacio se encuentra disponible o no.</p>
<b>Prioridad</b>	Alta

Tabla 47: Requerimiento funcional 16

<b>Título</b>	REF17 - Arrendamiento de un espacio de almacenamiento
<b>Narrativa</b>	<p><b>Como</b> usuario registrado</p> <p><b>Quiero</b> disponer de la opción para arrendar un espacio de almacenamiento cuando se encuentre disponible en el rango de fechas y metros cuadrados deseados</p> <p><b>Para</b> poder hacer uso del mismo.</p>
<b>Criterios de Aceptación</b>	<p><b>Dado</b> un usuario cliente identificado,</p> <p><b>Y</b> quiera arrendar un espacio disponible en la fecha y mts cuadrados deseados por el usuario,</p> <p><b>Cuando</b> seleccione la opción “solicitar arrendamiento”,</p> <p><b>Entonces</b> el sistema mostrará en pantalla un formulario solicitando la información básica de la empresa que desea confirmar esa operación</p> <p><b>Y</b> enviará un mensaje de confirmación notificando que se comunicarán con la empresa.</p>
<b>Prioridad</b>	Alta

Tabla 48: Requerimiento funcional 17

<b>Título</b>	REF18 - Compartir publicación de espacio de almacenamiento
<b>Narrativa</b>	<p><b>Como</b> usuario cliente identificado</p> <p><b>Quiero</b> tener la opción de compartir un espacio de almacenamiento en mis redes sociales</p> <p><b>Para</b> compartirlo con otros conocidos <b>y</b> viralizar el contenido de la página</p>
<b>Criterios de Aceptación</b>	<p><b>Dado</b> un usuario cliente identificado,</p> <p><b>Y</b> quiera compartir un espacio en sus redes sociales,</p> <p><b>Cuando</b> se encuentre en la pantalla del espacio a ser compartido</p> <p><b>Y</b> seleccione el icono de la red social</p> <p><b>Entonces</b> el sistema se redireccionará a la red social seleccionada permitiendo compartir el espacio</p>
<b>Prioridad</b>	Baja

Tabla 49: Requerimiento funcional 18

<b>Título</b>	REF19 - Registrar solicitud de espacio de almacenamiento
<b>Narrativa</b>	<p><b>Como</b> usuario cliente identificado,</p> <p><b>Quiero</b> generar una solicitud para publicar un espacio de almacenamiento que tengo disponible,</p> <p><b>Para</b> que otras empresas puedan hacer uso de el mismo.</p>
<b>Criterios de Aceptación</b>	<p><b>Dado</b> un usuario cliente identificado,</p> <p><b>Y</b> quiera registrar un espacio de almacenamiento en la web,</p> <p><b>Cuando</b> se encuentre en la pantalla de "Tengo Espacio"</p> <p><b>Y</b> complete el formulario de solicitud con los datos básicos</p> <p><b>Y</b> seleccione el botón "Registrar"</p> <p><b>Entonces</b> el sistema creará una solicitud de registro de espacio de almacenamiento y mostrará un mensaje de éxito</p>
<b>Prioridad</b>	Alta

Tabla 50: Requerimiento funcional 19

<b>Título</b>	REF20 - Cambiar estado de solicitud de registro de espacio de almacenamiento
<b>Narrativa</b>	<p><b>Como</b> usuario backoffice,  <b>Quiero</b> tener la posibilidad de aceptar o cancelar una solicitud de registro de espacio de almacenamiento,  <b>Para</b> que no se procese la solicitud de crear un nuevo espacio de almacenamiento.</p> <p><b>Como</b> usuario cliente identificado,  <b>Quiero</b> tener la posibilidad cancelar una solicitud de registro de espacio de almacenamiento,  <b>Para</b> que no se procese la solicitud de crear un nuevo espacio de almacenamiento.</p>
<b>Criterios de Aceptación</b>	<p><b>Dado</b> un usuario backoffice identificado,  <b>Y</b> quiera cancelar o completar una solicitud de registro de un espacio de almacenamiento,  <b>Cuando</b> se encuentre en la pantalla que muestra el listado de solicitudes de registro de espacios de almacenamientos  <b>Y</b> seleccione la opción "Cancelar" o "Completar" en uno de ellos  <b>Entonces</b> el sistema mostrar un mensaje notificando que la solicitud fue cancelada o completada de forma exitosa</p> <p><b>Dado</b> un usuario cliente identificado,  <b>Y</b> quiera cancelar una solicitud de registro de un espacio de almacenamiento,  <b>Cuando</b> se encuentre en la pantalla que muestra el listado de sus solicitudes de registro de espacios de almacenamientos  <b>Y</b> seleccione la opción "Cancelar" en uno de ellos  <b>Entonces</b> el sistema mostrar un mensaje notificando que la solicitud fue cancelada de forma exitosa</p>
<b>Prioridad</b>	Alta

Tabla 51: Requerimiento funcional 20

<b>Título</b>	REF21 - Alta de espacio de almacenamiento
<b>Narrativa</b>	<p><b>Como</b> usuario backoffice identificado,</p> <p><b>Quiero</b> habilitar que un espacio de almacenamiento esté visible en la web mientras que cumpla con las condiciones necesarias y definidas por parte de la empresa</p> <p><b>Para</b> que otras empresas interesadas puedan tener conocimiento del mismo.</p>
<b>Criterios de Aceptación</b>	<p><b>Dado</b> un usuario backoffice,</p> <p><b>Y</b> quiera dar de alta un almacenamiento que esté pendiente de aprobación en la lista de solicitudes,</p> <p><b>Cuando</b> se encuentre en la pantalla que muestra el listado de solicitudes de espacios de almacenamientos a ser aprobados</p> <p><b>Y</b> seleccione la opción "Aprobar"</p> <p><b>Entonces</b> el sistema mostrará un mensaje de que fue aprobado correctamente.</p> <p><b>Dado</b> un usuario backoffice identificado,</p> <p><b>Y</b> quiera dar de alta un almacenamiento que esté pendiente de aprobación en la lista de solicitudes,</p> <p><b>Cuando</b> se encuentre en la pantalla que muestra el listado de espacios de almacenamientos a ser aprobados</p> <p><b>Y</b> seleccione la opción "Rechazar"</p> <p><b>Entonces</b> el sistema mostrará un mensaje de que la publicación fue rechazada correctamente.</p>
<b>Prioridad</b>	Alta

Tabla 52: Requerimiento funcional 21

<b>Título</b>	REF22 - Baja de espacio de almacenamiento
<b>Narrativa</b>	<p><b>Como</b> usuario backoffice o cliente identificado,</p> <p><b>Quiero</b> tener la posibilidad de dar de baja una publicación en el sitio para que esta no se muestre más o no esté disponible</p> <p><b>Para</b> ser arrendada por otros usuarios.</p>

<b>Criterios de Aceptación</b>	<p><b>Dado</b> un usuario backoffice identificado,  <b>Y</b> quiera dar de baja un almacenamiento registrado  <b>Cuando</b> se encuentre en la pantalla que muestra el listado de espacios de almacenamientos activos  <b>Y</b> seleccione la opción “Cancelar” en uno de ellos  <b>Entonces</b> el sistema mostrar un mensaje notificando que la publicación se dio de baja de forma exitosa</p> <p><b>Dado</b> un usuario cliente identificado,  <b>Y</b> quiera dar de baja un almacenamiento registrado  <b>Cuando</b> se encuentre en la pantalla que muestra el listado de sus espacios de almacenamientos activos  <b>Y</b> seleccione la opción “Cancelar” en uno de ellos  <b>Entonces</b> el sistema mostrar un mensaje notificando que la publicación se dio de baja de forma exitosa</p>
<b>Prioridad</b>	Alta

Tabla 53: Requerimiento funcional 22

<b>Título</b>	REF23 - Modificar información del espacio de almacenamiento
<b>Narrativa</b>	<p><b>Como</b> usuario backoffice o cliente identificado,  <b>Quiero</b> poder modificar los datos de un espacio de almacenamiento,  <b>Para</b> corregir información que no es correcta o  <b>Para</b> agregar nuevos datos que den más completitud.</p>
<b>Criterios de Aceptación</b>	<p><b>Dado</b> un usuario cliente identificado,  <b>Y</b> quiera modificar una/s característica/s de alguno de sus espacios de almacenamiento.</p> <p><b>Cuando</b> se encuentre en la pantalla del listado de sus espacios de almacenamientos  <b>Y</b> seleccione el espacio de almacenamiento a modificar,  <b>Y</b> modifique alguna/s de las características  <b>Entonces</b> el sistema mostrará un mensaje de que el cambio fue realizado con éxito</p> <p><b>Dado</b> un usuario backoffice identificado,</p>

	<p><b>Y</b> quiera modificar una/s característica/s de alguno de los espacios de almacenamientos registrados,</p> <p><b>Cuando</b> se encuentre en la pantalla de todos los espacios de almacenamientos registrados,</p> <p><b>Y</b> seleccione el espacio de almacenamiento a modificar,</p> <p><b>Y</b> modifique alguna/s de las características</p> <p><b>Entonces</b> el sistema mostrará un mensaje de que el cambio fue realizado con éxito</p>
<b>Prioridad</b>	Alta

Tabla 54: Requerimiento funcional 23

<b>Título</b>	REF24 - Validar información modificada de un espacio de almacenamiento existente
<b>Narrativa</b>	<p><b>Como</b> usuario backoffice</p> <p><b>Quiero</b> tener la posibilidad de aceptar o rechazar una modificación a realizarse en una publicación del sitio</p> <p><b>Para</b> preservar la integridad de los datos e información que mostramos a los usuarios.</p>
<b>Criterios de Aceptación</b>	<p><b>Dado</b> un usuario backoffice identificado,</p> <p><b>Y</b> quiera validar información modificada de un almacenamiento identificado</p> <p><b>Cuando</b> se encuentre en la pantalla que muestra las modificaciones pendientes a ser aprobadas y seleccione la opción “Aprobar” en una de ellas,</p> <p><b>Entonces</b> el sistema mostrará un mensaje de que fue aprobado correctamente.</p> <p><b>Dado</b> un usuario backoffice identificado,</p> <p><b>Y</b> quiera validar información modificada de un almacenamiento identificado</p> <p><b>Cuando</b> se encuentre en la pantalla que muestra las modificaciones pendientes a ser aprobadas y seleccione la opción “Rechazar” en una de ellas,</p>

	<p><b>Entonces</b> el sistema mostrará un mensaje de que fue rechazada correctamente.</p> <p><b>Dado</b> un usuario backoffice identificado,  <b>Y</b> quiera validar información modificada de un almacenamiento identificado  <b>Y</b> hayan pasado más de 72 hs de la modificación  <b>Cuando</b> se dirija a aprobar la modificación en cuestión  <b>Entonces</b> el sistema ya habrá hecho la modificación automáticamente</p>
<b>Prioridad</b>	Media

Tabla 55: Requerimiento funcional 24

<b>Título</b>	REF25 - Crear menú principal
<b>Narrativa</b>	<p><b>Como</b> usuario backoffice o cliente identificado,  <b>Quiero</b> ingresar a la web  <b>Y</b> visualizar información de la empresa  <b>Para</b> tener entendimiento del negocio</p>
<b>Criterios de Aceptación</b>	<p><b>Dado</b> un usuario cliente identificado,  <b>Cuando</b> ingrese a la web  <b>Entonces</b> el sistema mostrará una home page  <b>Y</b> se mostrará información general de WareUP entre otras cosas</p> <p><b>Dado</b> un usuario backoffice identificado,  <b>Cuando</b> ingrese a la web  <b>Entonces</b> el sistema mostrará una home page  <b>Y</b> se mostrará información general de WareUP entre otras cosas.</p>
<b>Prioridad</b>	Alta

Tabla 56: Requerimiento funcional 25

## Product Backlog - Modificaciones posteriores a la versión 1

**REF6 – Niveles de usuario Backoffice:** se decidió en conjunto con el cliente que en la primera versión haya únicamente un nivel de usuario backoffice que tenga habilitado todos los permisos posibles.

**REF11 - Ordenar resultados de búsqueda:** Requerimiento que finalmente no fue implementado ya que al igual que el REF13, sobre el final del desarrollo se priorizo la corrección de *bugs*, entendiendo que era de mayor importancia. Este requerimiento quedaría para implementar en una próxima versión del producto.

**REF12 - Mapa con las publicaciones en base a la cercanía del usuario:** Este requerimiento fue modificado de manera que se muestran los depósitos ubicados en el mapa pero sin considerar la cercanía del usuario.

**REF13 - Significado de los filtros:** Este requerimiento fue dado de baja sobre el final del cronograma ya que el tiempo que restaba se priorizo para la corrección de cambios e incidentes y se consideró que al ser un requerimiento de baja prioridad no generaba mayor impacto en el MVP final.

**REF16 - Comprobar disponibilidad de un espacio:** En conjunto con el cliente se decidió que este requerimiento quedará para una segunda versión. Ofreciendo en la primera versión del MVP una solicitud de arrendamiento sin la necesidad de corroborar previamente la fecha disponible. Esta decisión se tomó por motivos de tiempos, entendiendo que la implementación requería una lógica más compleja y nos consumía una mayor cantidad de tiempo y esfuerzo de la permitida.

**REF17- Generar solicitud de arrendamiento de un espacio de almacenamiento:** En relación a este requerimiento, se modificaron dos puntos, por un lado la solicitud de arrendamiento será hecha sin la necesidad de comprobar la disponibilidad, quedando del lado del cliente en todo caso rechazar la solicitud. Por otro lado, se omite la necesidad de solicitar los datos de la empresa, ya que la solicitud quedará registrada bajo el usuario logueado.

**REF18 - Compartir publicación de espacio de almacenamiento:** Bajo mutuo acuerdo con el cliente se decide suprimir este requerimiento debido a que se llegó a la conclusión de que no es prioritario. Se decide destinar este tiempo en el desarrollo de otras funcionalidades.

**REF24 - Validar información modificada de un espacio de almacenamiento existente:** Inicialmente en este requerimiento se planteó la posibilidad de que un usuario backoffice tuviese un máximo de 72 hs hábiles para aprobar o rechazar la solicitud de modificación de una publicación de un usuario, finalmente para mayor

fluidez se decidió que las modificaciones se hagan de forma automática sin la necesidad de que el backoffice interfiera en el proceso. Se conversó con el cliente que cuando la fluidez y el flujo de usuarios sea mayor sería conveniente reanudar este requerimiento para tener un mayor control de información de la web.

**REF26 - Registrar una empresa:** En conjunto con el cliente, y basándonos en el funcionamiento del negocio, se decidió que la entidad empresa sea apartada de la entidad usuario, permitiendo así a un usuario registrar más de una empresa. Es por esto, que se agrega este requerimiento que hace referencia al registro de una nueva empresa.

**REF27 - Baja de una empresa:** Como usuario registrado quiero poder eliminar o modificar los datos de una empresa registrada. Se agrega este requerimiento.

**REF28 - Modificar la información de una empresa:** Como usuario registrado quiero poder eliminar o modificar los datos de una empresa registrada. Se agrega este requerimiento.

**REF29 - Cargar disponibilidad de un espacio de almacenamiento:** Esta funcionalidad deberá ser tomada como un requerimiento aparte. Como usuario común o backoffice quiero poder cargar la disponibilidad que tiene un espacio de almacenamiento para que otro usuario pueda arrendarlo.

**REF30 - Cargar imágenes de un espacio de almacenamiento:** Como usuario común o backoffice quiero poder cargar las imágenes correspondientes a un espacio de almacenamiento.

**REF31 - Contactar empresa:** Como usuario quiero tener la posibilidad de contactar con la empresa WareUP para evacuar posibles dudas. Se agrega esto como un requerimiento.

**REF32 - Métricas internas del sistema para usuario backoffice:** Como usuario backoffice quiero saber cuantas entidades tengo registradas en mi base de datos, en qué estado están y cuando se crearon para tener un panorama del estado actual de mi empresa.

A continuación, se detallan nuevos requerimientos que fueron surgiendo como necesidad a partir del desarrollo que íbamos implementando:

**REF33 - Listado de usuarios registrados:** Como usuario backoffice quiero poder visualizar un listado de los usuarios que están registrados en mi web.

**REF34 - Listado de espacios de almacenamiento registrados:** Como usuario backoffice o usuario cliente quiero poder visualizar el listado de todos los espacios de almacenamiento registrados en el sistema o el listado de espacios de almacenamiento que el propio usuario haya registrado.

**REF35 - Listado de solicitudes de registro de espacios de almacenamiento:** Como usuario backoffice o usuario cliente quiero poder visualizar el listado de solicitudes de registro de espacios de almacenamiento.

**REF36 - Listado de solicitudes de arrendamiento:** Como usuario backoffice o usuario cliente quiero poder visualizar el listado de todas las solicitudes de arrendamiento que se hicieron en el sistema o el listado de solicitudes de arrendamiento que haya registrado el propio usuario.

**REF37 - Listado de empresas de un usuario:** Como usuario cliente quiero poder visualizar el listado de todas las empresas que haya registrado en el sistema.

**REF 38 - Modificar información de un usuario registrado:** Como usuario backoffice quiero poder modificar la información personal de cualquiera de los usuarios que se encuentran registrados.

**REF 39 - Dar de baja un usuario registrado:** Como usuario backoffice quiero poder eliminar de forma lógica cualquier usuario que esté registrado en el sistema.

**REF 40 - Listado de rangos de disponibilidad agregados para un espacio de almacenamiento:** Como usuario backoffice o cliente quiero ver la configuración de disponibilidad. Esto muestra los distintos rangos configurados de disponibilidad para un mismo espacio de almacenamiento.

Nuevas user stories

A continuación se detallan las historias de usuarios de los requerimientos que fueron desarrollados posteriormente a la primera versión del Product Backlog.

<b>Título</b>	REF26 - Registrar una empresa
---------------	-------------------------------

<b>Narrativa</b>	<p><b>Como</b> usuario identificado</p> <p><b>Quiero</b> registrar una empresa</p> <p><b>Para</b> luego publicar un espacio perteneciente a esa empresa</p>
<b>Criterios de Aceptación</b>	<p><b>Dado</b> un usuario cliente identificado,</p> <p><b>Y</b> quiera registrar un empresa,</p> <p><b>Cuando</b> se encuentre en la pantalla de “Registrar empresa”</p> <p><b>Y</b> complete el formulario con los datos solicitados</p> <p><b>Y</b> apriete el botón “Registrar”</p> <p><b>Entonces</b> el sistema mostrará un mensaje de éxito</p> <p><b>Dado</b> un usuario cliente identificado</p> <p><b>Y</b> quiera registrar un empresa</p> <p><b>Cuando</b> se encuentre en la pantalla de “Registrar empresa”</p> <p><b>Y</b> complete el formulario con los datos solicitados</p> <p><b>Y</b> coloqué un RUT de una empresa que ya está registrada</p> <p><b>Y</b> apriete el botón “Registrar”</p> <p><b>Entonces</b> el sistema mostrará un mensaje de error notificando que el RUT ya se encuentra registrado</p>
<b>Prioridad</b>	Alta

Tabla 57: Requerimiento funcional 26

<b>Título</b>	REF27 - Baja de una empresa
<b>Narrativa</b>	<p><b>Como</b> usuario backoffice o cliente identificado</p> <p><b>Quiero</b> eliminar una empresa</p> <p><b>Para</b> que no forme más parte de las empresas registradas</p>
<b>Criterios de Aceptación</b>	<p><b>Dado</b> un usuario cliente identificado</p> <p><b>Y</b> quiera eliminar una empresa registrada</p> <p><b>Cuando</b> se encuentre en la pantalla de “Mis empresas”</p> <p><b>Y</b> selecciona eliminar una empresa</p> <p><b>Entonces</b> el sistema mostrará un mensaje “Empresa eliminada exitosamente”</p> <p><b>Dado</b> un usuario backoffice identificado</p>

	<p><b>Y</b> quiera eliminar una empresa registrada</p> <p><b>Cuando</b> se encuentre en la pantalla de todas las empresas registradas</p> <p><b>Y</b> selecciona eliminar una empresa</p> <p><b>Entonces</b> el sistema mostrará un mensaje “Empresa eliminada exitosamente”</p>
<b>Prioridad</b>	Alta

Tabla 58: Requerimiento funcional 27

<b>Título</b>	REF28 - Modificar información de una empresa
<b>Narrativa</b>	<p><b>Como</b> usuario backoffice o cliente identificado</p> <p><b>Quiero</b> modificar los datos de una empresa</p> <p><b>Para</b> para actualizar los datos de ella</p>
<b>Criterios de Aceptación</b>	<p><b>Dado</b> un usuario cliente identificado,</p> <p><b>Y</b> quiera modificar una empresa registrada,</p> <p><b>Cuando</b> se encuentre en la pantalla de “Mis empresas”</p> <p><b>Y</b> selecciona modificar una empresa</p> <p><b>Y</b> modifique alguno de los datos</p> <p><b>Entonces</b> el sistema mostrará un mensaje “Empresa actualizada exitosamente”</p> <p><b>Dado</b> un usuario backoffice identificado,</p> <p><b>Y</b> quiera modificar una empresa registrada,</p> <p><b>Cuando</b> se encuentre en la pantalla de todas las empresas registradas</p> <p><b>Y</b> selecciona modificar una empresa</p> <p><b>Y</b> modifique alguno de los datos</p> <p><b>Entonces</b> el sistema mostrará un mensaje “Empresa actualizada exitosamente”</p>
<b>Prioridad</b>	Alta

Tabla 59: Requerimiento funcional 28

<b>Título</b>	REF29 - Cargar disponibilidad de un espacio de almacenamiento
---------------	---

<b>Narrativa</b>	<p><b>Como</b> usuario backoffice o cliente identificado</p> <p><b>Quiero</b> cargar la disponibilidad de un espacio de almacenamiento</p> <p><b>Para</b> que otros tenga visibilidad sobre la disponibilidad y puedan arrendarlo</p>
<b>Criterios de Aceptación</b>	<p><b>Dado</b> un usuario cliente identificado</p> <p><b>Cuando</b> se encuentre en la pantalla de “mis espacios”</p> <p><b>Y</b> selecciona la opción “Agregar disponibilidad” en uno de sus espacios de almacenamiento</p> <p><b>Y</b> seleccione la disponibilidad de fechas</p> <p><b>Y</b> de metros cúbicos</p> <p><b>Entonces</b> el sistema mostrará un mensaje “Disponibilidad agregada exitosamente”</p> <p><b>Dado</b> un usuario backoffice identificado</p> <p><b>Cuando</b> se encuentre en la pantalla de todos los espacios de almacenamientos registrados</p> <p><b>Y</b> selecciona la opción “Agregar disponibilidad” en uno de los espacios de almacenamiento</p> <p><b>Y</b> seleccione la disponibilidad de fechas</p> <p><b>Y</b> de metros cúbicos</p> <p><b>Entonces</b> el sistema mostrará un mensaje “Disponibilidad agregada exitosamente”</p>
<b>Prioridad</b>	Alta

Tabla 60: Requerimiento funcional 29

<b>Título</b>	REF30 - Cargar imágenes en un espacio de almacenamiento
<b>Narrativa</b>	<p><b>Como</b> usuario backoffice o cliente identificado</p> <p><b>Quiero</b> cargar imágenes de un espacio de almacenamiento</p> <p><b>Para</b> que quienes quiera arrendarlo puedan tener visibilidad previa</p>
<b>Criterios de Aceptación</b>	<p><b>Dado</b> un usuario backoffice identificado</p> <p><b>Cuando</b> se encuentre en la pantalla de todos los espacios de almacenamientos registrados</p>

	<p>Y selecciona la opción “Agregar imágenes” en uno de los espacios de almacenamiento</p> <p>Y seleccione las imágenes a ser cargadas</p> <p>Y seleccione el botón “Subir”</p> <p><b>Entonces</b> el sistema mostrará un mensaje “Imagenes cargadas exitosamente”</p> <p><b>Dado</b> un usuario cliente identificado</p> <p><b>Cuando</b> se encuentre en la pantalla de “Mis espacios”</p> <p>Y selecciona la opción “Agregar imágenes” en uno de sus espacios de almacenamiento</p> <p>Y seleccione las imágenes a ser cargadas</p> <p>Y seleccione el botón “Subir”</p> <p><b>Entonces</b> el sistema mostrará un mensaje “Imagenes cargadas exitosamente”</p>
<b>Prioridad</b>	Alta

Tabla 61: Requerimiento funcional 30

<b>Título</b>	REF31 - Contactar empresa
<b>Narrativa</b>	<p><b>Como</b> usuario</p> <p><b>Quiero</b> tener la opción de contactar a la empresa</p> <p><b>Para</b> evacuar posibles dudas</p>
<b>Criterios de Aceptación</b>	<p><b>Dado</b> un usuario cliente identificado</p> <p><b>Cuando</b> se encuentre en la pantalla de “Mis empresas”</p> <p><b>Entonces</b> el sistema mostrará un mensaje “Empresa eliminada exitosamente”</p>
<b>Prioridad</b>	Alta

Tabla 62: Requerimiento funcional 31

<b>Título</b>	REF32 - Métricas internas del sistema para usuario backoffice
---------------	---

<b>Narrativa</b>	<p><b>Como</b> usuario backoffice identificado</p> <p><b>Quiero</b> saber cuantas entidades tengo registradas en mi base de datos</p> <p><b>Y</b> en qué estado están</p> <p><b>Y</b> cuando se crearon</p> <p><b>Para</b> tener un panorama del estados actual de mi empresa</p>
<b>Criterios de Aceptación</b>	<p><b>Dado</b> un usuario backoffice identificado</p> <p><b>Cuando</b> se encuentre en la pantalla “Métricas”</p> <p><b>Entonces</b> el sistema mostrará de manera gráfica diferentes datos correspondientes a las entidades del sistema</p>
<b>Prioridad</b>	Alta

Tabla 63: Requerimiento funcional 32

<b>Título</b>	REF33 - Listado de usuarios registrados
<b>Narrativa</b>	<p><b>Como</b> usuario backoffice identificado</p> <p><b>Quiero</b> visualizar un listado de los usuarios registrados en la web</p> <p><b>Para</b> tener control sobre los mismos</p>
<b>Criterios de Aceptación</b>	<p><b>Dado</b> un usuario backoffice identificado</p> <p><b>Cuando</b> se encuentre en la pantalla “Usuarios”</p> <p><b>Entonces</b> el sistema mostrará una lista de los usuarios registrados en el sistema</p>
<b>Prioridad</b>	Alta

Tabla 64: Requerimiento funcional 33

<b>Título</b>	REF34 - Listado de espacios de almacenamiento registrados
<b>Narrativa</b>	<p><b>Como</b> usuario cliente identificado</p> <p><b>Quiero</b> visualizar un listado de mis espacios de almacenamientos registrados</p> <p><b>Para</b> tener control sobre los mismos</p> <p><b>Como</b> usuario backoffice identificado</p>

	<p><b>Quiero</b> visualizar un listado de todos los espacios de almacenamientos registrados en la web</p> <p><b>Para</b> tener control sobre los mismos</p>
<b>Criterios de Aceptación</b>	<p><b>Dado</b> un usuario cliente identificado</p> <p><b>Cuando</b> se encuentre en la pantalla “Listado de espacios de almacenamiento”</p> <p><b>Entonces</b> el sistema mostrará una lista de sus espacios registrados en el sistema</p> <p><b>Dado</b> un usuario backoffice identificado</p> <p><b>Cuando</b> se encuentre en la pantalla “Listado de espacios de almacenamiento”</p> <p><b>Entonces</b> el sistema mostrará una lista de todos los espacios registrados en el sistema</p>
<b>Prioridad</b>	Alta

Tabla 65: Requerimiento funcional 34

<b>Título</b>	REF35 - Listado de solicitudes de registro de espacios de almacenamiento
<b>Narrativa</b>	<p><b>Como</b> usuario backoffice identificado</p> <p><b>Quiero</b> visualizar un listado de todas las solicitudes de registro de espacios de almacenamientos</p> <p><b>Para</b> tener control sobre los mismos</p>
<b>Criterios de Aceptación</b>	<p><b>Dado</b> un usuario backoffice identificado</p> <p><b>Cuando</b> se encuentre en la pantalla “Solicitudes de registro de espacios de almacenamiento”</p> <p><b>Entonces</b> el sistema mostrará una lista de todos las solicitudes de registro de espacios en el sistema</p>
<b>Prioridad</b>	Alta

Tabla 66: Requerimiento funcional 35

<b>Título</b>	REF36 - Listado de solicitudes de arrendamiento
---------------	---

<b>Narrativa</b>	<p><b>Como</b> usuario backoffice identificado</p> <p><b>Quiero</b> visualizar un listado de todas las solicitudes de arrendamiento en el sistema</p> <p><b>Para</b> tener control sobre las mismos</p> <p><b>Como</b> usuario cliente identificado</p> <p><b>Quiero</b> visualizar un listado de todas mis solicitudes de arrendamiento</p> <p><b>Para</b> tener control sobre las mismos</p>
<b>Criterios de Aceptación</b>	<p><b>Dado</b> un usuario backoffice identificado</p> <p><b>Cuando</b> se encuentre en la pantalla “Solicitudes de arrendamiento”</p> <p><b>Entonces</b> el sistema mostrará una lista de todos las solicitudes de arrendamientos de espacios que existen en el sistema</p> <p><b>Dado</b> un usuario cliente identificado</p> <p><b>Cuando</b> se encuentre en la pantalla “Solicitudes de arrendamiento”</p> <p><b>Entonces</b> el sistema mostrará una lista de todos las solicitudes de arrendamientos de espacios que hizo el usuario</p>
<b>Prioridad</b>	Alta

Tabla 67: Requerimiento funcional 36

<b>Título</b>	REF37 - Listado de empresas de un usuario
<b>Narrativa</b>	<p><b>Como</b> usuario cliente identificado</p> <p><b>Quiero</b> visualizar un listado de todas las empresas que tengo registradas bajo mi nombre de usuario</p> <p><b>Para</b> tener control sobre los mismos</p>
<b>Criterios de Aceptación</b>	<p><b>Dado</b> un usuario cliente identificado</p> <p><b>Cuando</b> se encuentre en la pantalla “Mis empresas”</p> <p><b>Entonces</b> el sistema mostrará una lista de todos las empresas que se encuentran registradas bajo su nombre de usuario</p>
<b>Prioridad</b>	Alta

Tabla 68: Requerimiento funcional 37

<b>Título</b>	REF 38 - Modificar información de un usuario registrado
<b>Narrativa</b>	<p><b>Como</b> usuario backoffice identificado</p> <p><b>Quiero</b> poder modificar la información de alguno de los usuarios registrados en el sistema</p> <p><b>Para</b> hacer las modificaciones necesarias en caso de que se requiera</p>
<b>Criterios de Aceptación</b>	<p><b>Dado</b> un usuario backoffice identificado</p> <p><b>Cuando</b> se encuentre en la pantalla “Listado de usuarios”</p> <p><b>Y</b> seleccione la opción “Editar” sobre uno de los usuarios</p> <p><b>Entonces</b> el sistema mostrará una pantalla para que se puedan modificar sus datos</p>
<b>Prioridad</b>	Alta

Tabla 69: Requerimiento funcional 38

<b>Título</b>	REF 39 - Dar de baja un usuario registrado
<b>Narrativa</b>	<p><b>Como</b> usuario backoffice identificado</p> <p><b>Quiero</b> poder eliminar de forma lógica cualquier usuario que se encuentre registrado en el sistema</p> <p><b>Para</b> quitarlo del sistema en caso de que no se encuentre más activo</p>
<b>Criterios de Aceptación</b>	<p><b>Dado</b> un usuario backoffice identificado</p> <p><b>Cuando</b> se encuentre en la pantalla “Listado de usuarios”</p> <p><b>Y</b> seleccione la opción “Eliminar” sobre uno de los usuarios</p> <p><b>Entonces</b> el sistema borrará de forma lógica al usuario seleccionado</p> <p><b>Y</b> mostrará un mensaje de que fue eliminado de forma exitosa</p>
<b>Prioridad</b>	Alta

Tabla 70: Requerimiento funcional 39

<b>Título</b>	REF 40 - Listado de rangos de disponibilidad agregados para un espacio de almacenamiento
---------------	--

<b>Narrativa</b>	<p><b>Como</b> usuario cliente identificado  <b>Quiero</b> visualizar el listado de rangos de disponibilidad agregados a uno de mis espacios de almacenamiento registrados  <b>Para</b> tener control sobre los mismos</p> <p><b>Como</b> backoffice identificado  <b>Quiero</b> visualizar el listado de rangos de disponibilidad agregados en todos los espacios de almacenamiento registrados en el sistema  <b>Para</b> tener control sobre los mismos</p>
<b>Criterios de Aceptación</b>	<p><b>Dado</b> un usuario cliente identificado  <b>Cuando</b> se encuentre en la pantalla “Mis espacios de almacenamiento”  <b>Y</b> seleccione la opción “Ver configuración de disponibilidad” sobre uno de ellos  <b>Entonces</b> el sistema mostrará en forma de lista los distintos rangos configurados de disponibilidad para ese mismo espacio de almacenamiento.</p> <p><b>Dado</b> un usuario backoffice identificado  <b>Cuando</b> se encuentre en la pantalla “Espacios de almacenamiento”  <b>Y</b> seleccione la opción “Ver configuración de disponibilidad” sobre uno de ellos  <b>Entonces</b> el sistema mostrará en forma de lista los distintos rangos configurados de disponibilidad para ese mismo espacio de almacenamiento.</p>
<b>Prioridad</b>	Alta

Tabla 71: Requerimiento funcional 40

## Anexo 3 - Prototipos iniciales

A continuación, se detallan algunos prototipos iniciales que se realizaron a través de la aplicación Figma [19] y fueron revisados en conjunto con el cliente.

### Login de la página

Inicialmente se presentó el login como una nueva página, donde se requería la información básica para iniciar sesión y se brindaban las opciones de recuperar contraseña y/o registrarse en caso de que el usuario no estuviera registrado previamente.

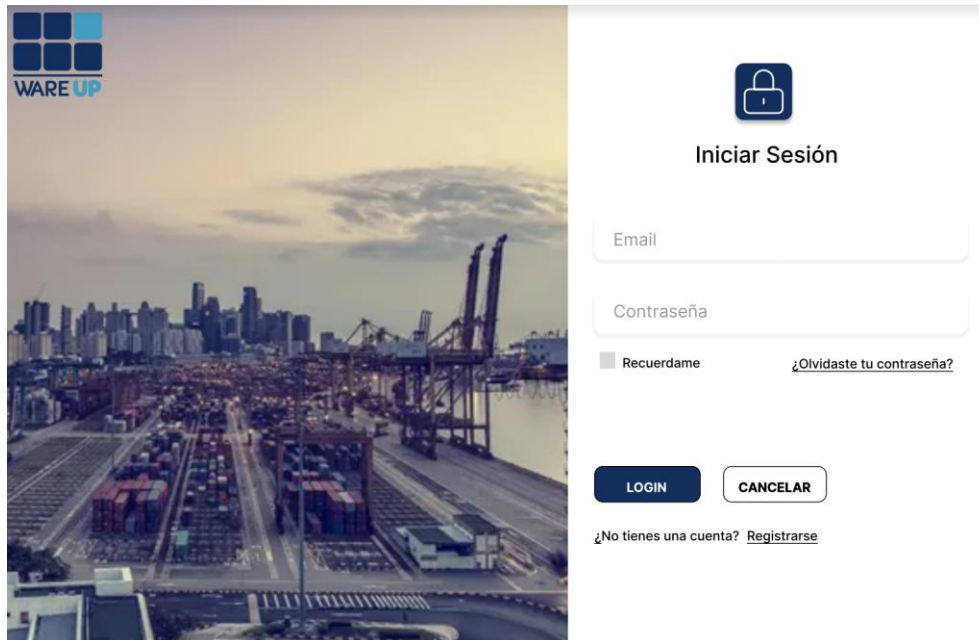


Ilustración 97: Prototipo de Login

Sin embargo, este prototipo cambió ya que el login pasó a ser un popup de la página donde se encuentre el usuario para facilitar su uso y evitar tener que redirigir al usuario a otra página particular.

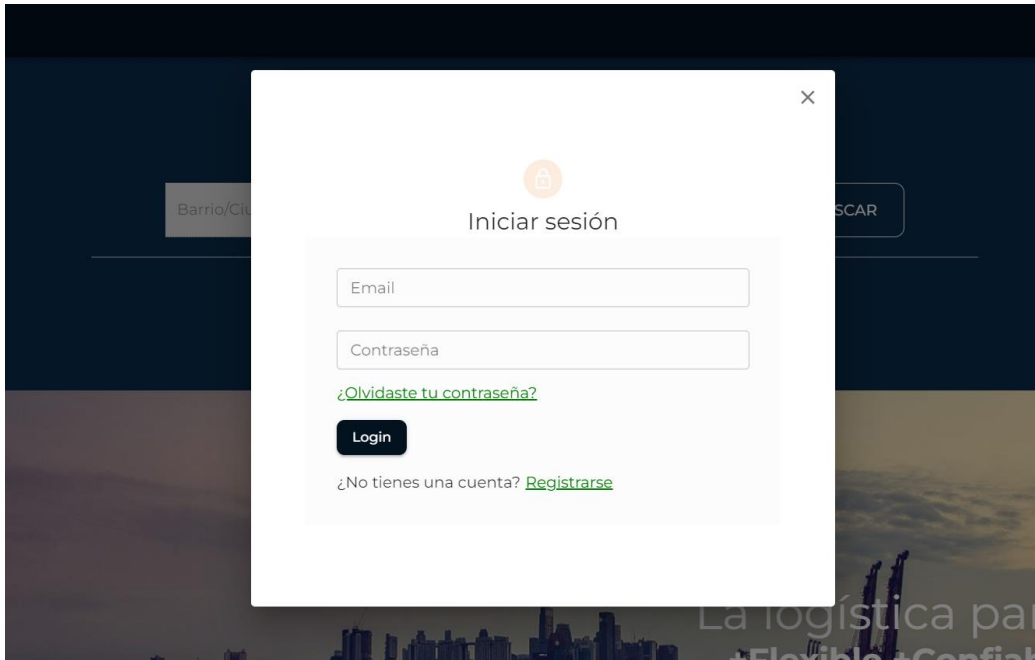


Ilustración 98: Login de la página

## Recuperar contraseña

Pantalla sencilla e intuitiva, el usuario deberá ingresar su email para recuperar su contraseña.

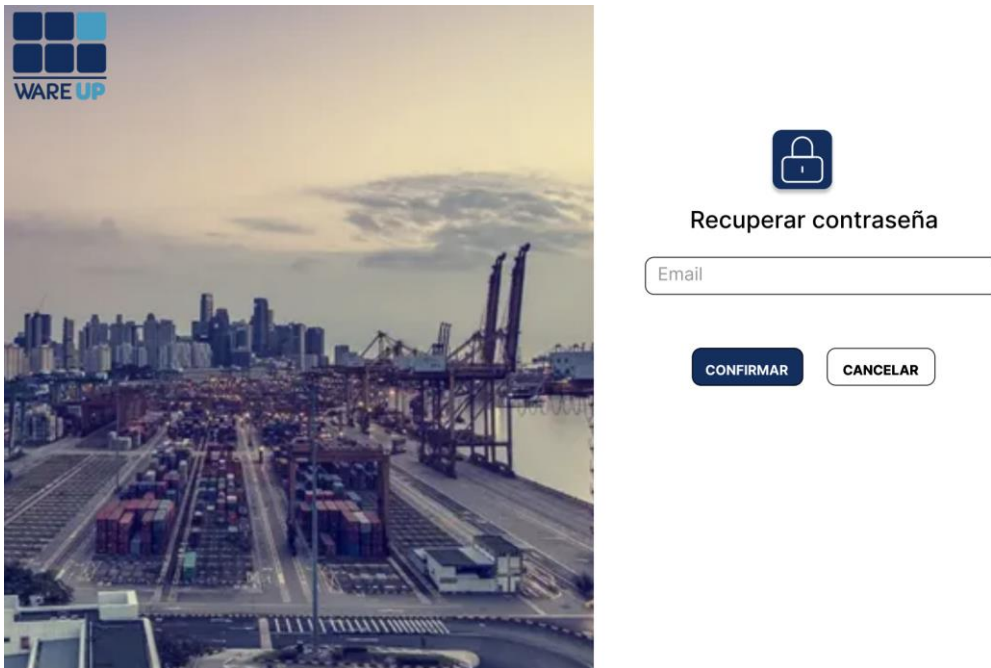


Ilustración 99: Prototipo recuperar contraseña

Este prototipo se mantuvo tal cual fue planificado ya que se consideró que era sencillo de entender y mantenía la estética y el concepto de la página en general.

## Crear nuevo usuario

En este prototipo se puede visualizar la información que inicialmente se pretendía cargar de un usuario nuevo a ser registrado en el sistema.



### Crear nueva cuenta

Nombre	Apellido
Email *	
Contraseña *	
Repetir contraseña *	
<input type="checkbox"/> He leído y acepto las políticas de privacidad	
<input type="button" value="REGISTRARSE"/>	<input type="button" value="CANCELAR"/>

Ilustración 100: Prototipo registrar usuario

Sin embargo, este prototipo se vió modificado ya que finalmente se agregó un nuevo campo requerido para la creación de un usuario. Este nuevo campo no había sido tenido en cuenta inicialmente porque en la página actual del cliente no era un campo existente pero se vió que podíamos obtener información valiosa de los usuarios en caso de que lo tuviéramos.

  
Crear nueva cuenta

Nombre	Apellido
--------	----------

Email\*

Contraseña\*

Repetir contraseña\*

Industrias con las que estoy familiarizad... 

He leído y acepto las políticas de privacidad.\*

Registrarse Cancelar

Ilustración 101: Registro de usuario

### Home con usuario que haya iniciado sesión al sistema

Inicialmente se pensaba tener un menú lateral con las distintas opciones que el usuario podía tener en el sistema y un panel principal con información de la empresa.



Ilustración 102: Prototipo home de usuario logueado

Sin embargo, no nos quedaba claro en qué nos podía aportar dicha página. Por ende,

se decidió no tener una home por usuario logueado sino que mantener la home principal de bienvenida a la página, y en caso de que el usuario esté logueado, cambiarán las opciones a mostrarse en la barra principal. Por otro lado, en base a la sugerencia del cliente, utilizar un panel lateral no era de su preferencia por lo que optamos por tener una barra horizontal con las opciones del sistema.

### Home sin usuario que haya iniciado sesión

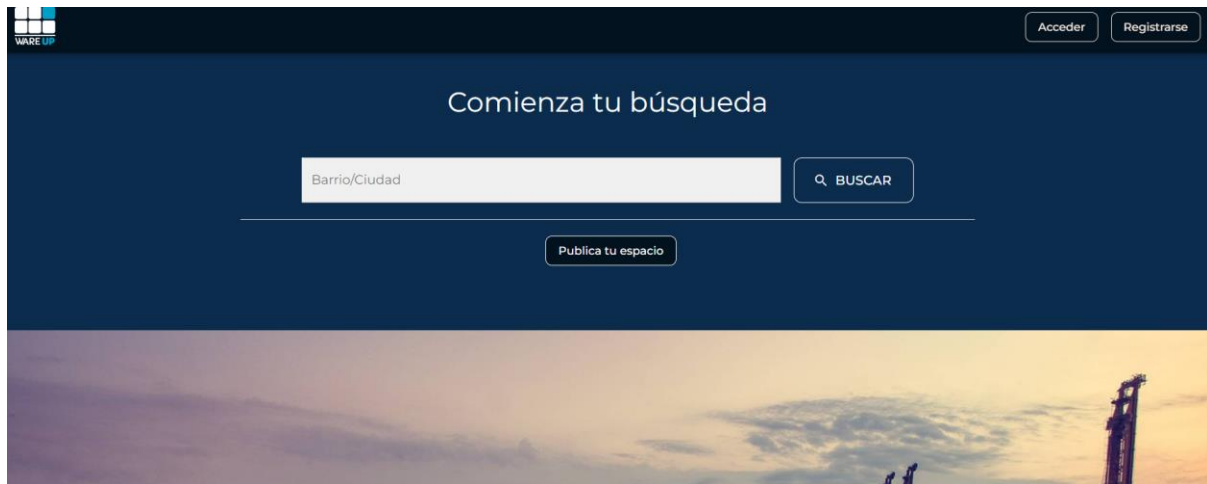


Ilustración 103: Página principal con usuario nos iniciar sesión

### Home con usuario que sí haya iniciado sesión

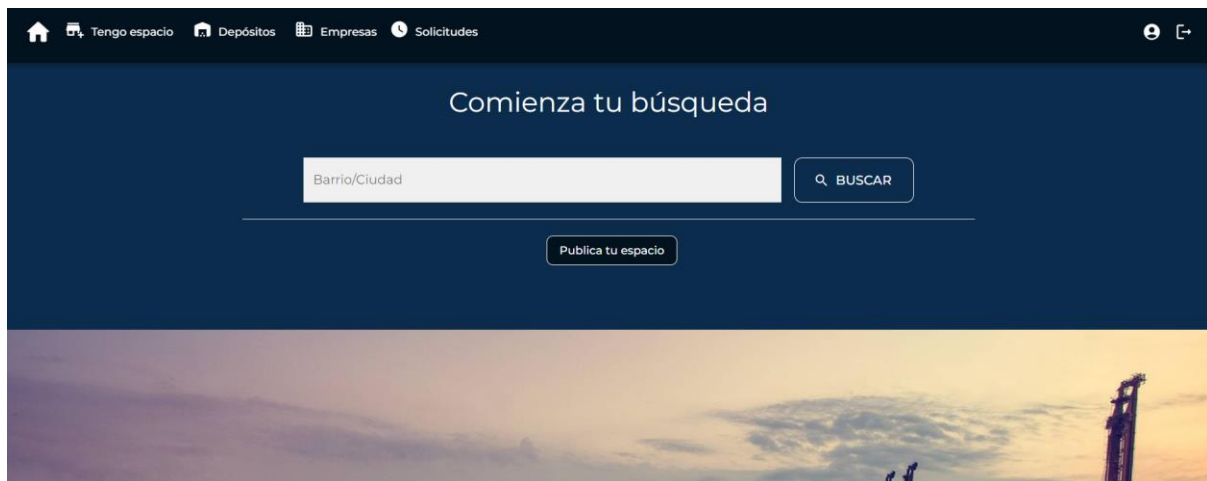


Ilustración 104: Página principal con usuario que inició sesión

Por otro lado, las opciones “Sobre nosotros” y “Contactanos” no serán parte de la barra principal, sino que serán parte del pie de página. Además se agregaron otras

opciones que no habían sido tenidas en cuenta en los prototipos iniciales como pueden ser “Términos y condiciones” y acceso a las redes sociales del cliente.



Ilustración 105: Pie de página

## Formulario “Tengo espacio”

El usuario cliente tiene la opción de solicitar el registro de su espacio de almacenamiento en el sistema y para concretar este requerimiento, debe completar el formulario que se muestra a continuación. La información que inicialmente se pedía fue de acuerdo a la que hoy en día se pide en la página del cliente.

**Referente**

EMPRESA  
Razón social

RUT

Nombre fantasía

Celular

Contacto  
Nombre

Celular

Cargo en la empresa

Correo

**Datos depósito**

Dirección

Telefono

Barrio

¿Cuenta esta ubicación con operativa?  
 SI  NO

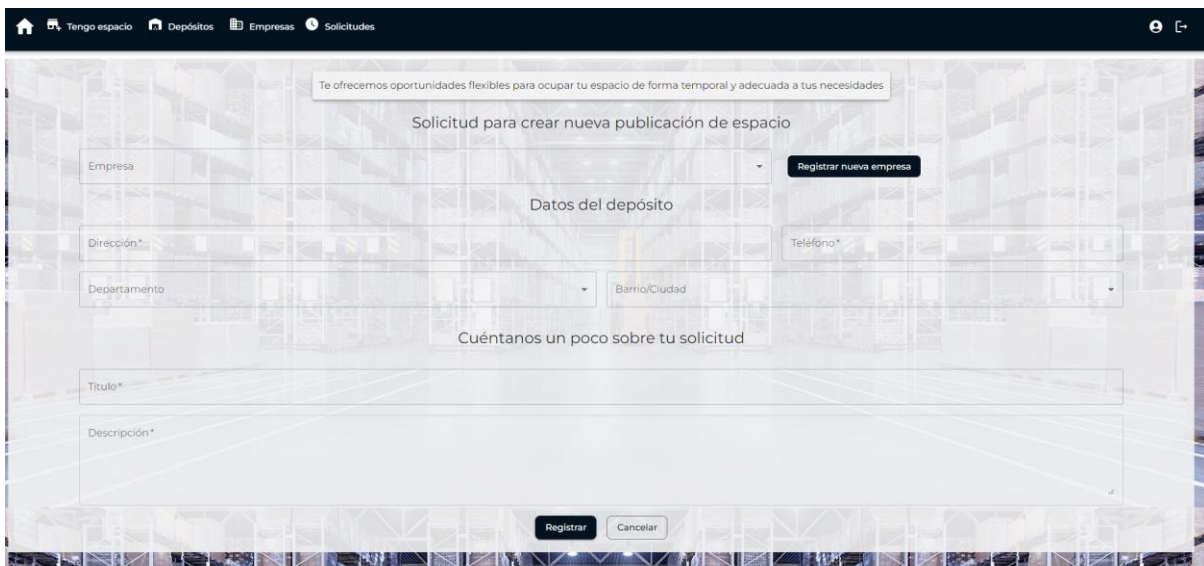
Mensaje (opcional)

ENVIAR

Ilustración 106: Prototipo de formulario “Tengo espacio”

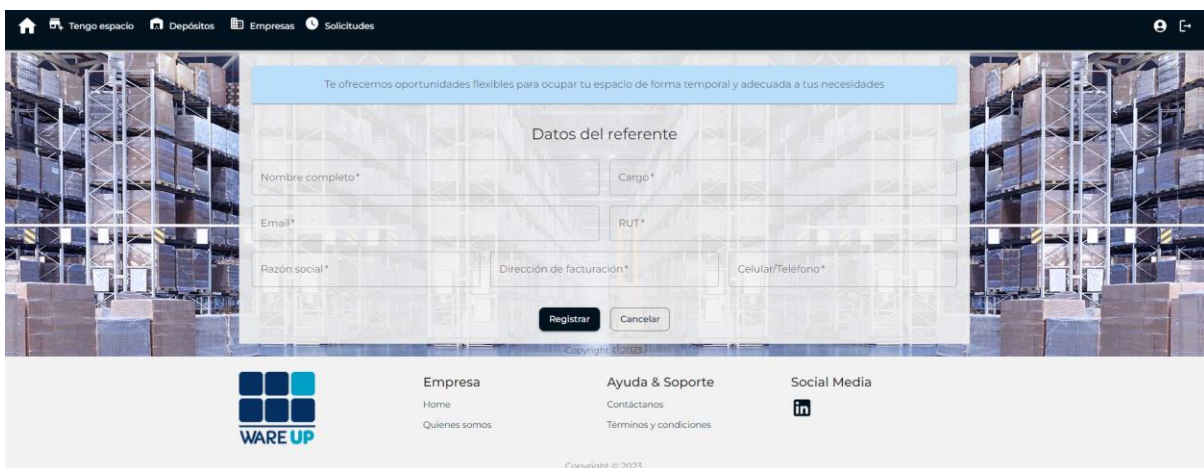
Sin embargo, este formulario sufrió algunas modificaciones de acuerdo a lo discutido con el cliente. Por un lado, se registran las empresas del usuario y por el otro, se realiza el registro del formulario. Como precondition para completar este formulario, se deberá tener registrada la empresa del usuario. Se tomó esa decisión con el fin de

poder reutilizar la información de las empresas de un usuario y que no tenga que reescribir cada vez que necesite registrar un espacio de almacenamiento lo cual podría tornarse repetitivo y no simplifica la usabilidad de la funcionalidad.



The screenshot shows a web form titled "Solicitud para crear nueva publicación de espacio". At the top, there is a navigation bar with icons for "Tengo espacio", "Depósitos", "Empresas", and "Solicitudes". Below the navigation bar, a message reads: "Te ofrecemos oportunidades flexibles para ocupar tu espacio de forma temporal y adecuada a tus necesidades". The form is divided into several sections: 1. "Empresa": A dropdown menu and a "Registrar nueva empresa" button. 2. "Datos del depósito": Fields for "Dirección\*", "Departamento", "Barrio/Ciudad", and "Teléfono\*". 3. "Cuéntanos un poco sobre tu solicitud": Fields for "Título\*" and "Descripción\*". At the bottom of the form are "Registrar" and "Cancelar" buttons.

Ilustración 107: Formulario “Tengo espacio”



The screenshot shows a web form titled "Datos del referente". At the top, there is a navigation bar with icons for "Tengo espacio", "Depósitos", "Empresas", and "Solicitudes". Below the navigation bar, a message reads: "Te ofrecemos oportunidades flexibles para ocupar tu espacio de forma temporal y adecuada a tus necesidades". The form is divided into several sections: 1. "Datos del referente": Fields for "Nombre completo\*", "Cargo\*", "Email\*", "RUT\*", "Razón social\*", "Dirección de facturación\*", and "Celular/Teléfono\*". 2. At the bottom of the form are "Registrar" and "Cancelar" buttons. The footer of the page includes the "WARE UP" logo, navigation links for "Empresa" (Home, Quiénes somos), "Ayuda & Soporte" (Contáctanos, Términos y condiciones), and "Social Media" (LinkedIn icon). A copyright notice "Copyright © 2023" is visible at the bottom.

Ilustración 108: Formulario “Registrar empresa”

## Buscador principal en la home page

Una de las funcionalidades principales con las cuales debía contar el sistema a desarrollar, era el buscador de espacios de almacenamiento. Este se diseñó a partir de la página Warehouse Exchange, la cual fue sugerida por el cliente. En el prototipo se puede visualizar un cuadro de texto donde nos indica que podemos ingresar un barrio o ciudad, acompañado de un botón Buscar y además, otro botón que indica “Publica tu espacio”. Este último botón se pensó como un “atajo” para llegar al

formulario de “Tengo espacio”. En caso de que el usuario haya iniciado sesión, nos redirigirá al formulario, y por otro lado, si el usuario no inició sesión aún, nos mostrará el popup con el login.



Ilustración 109: Prototipo buscador principal

Finalmente, se mantuvo la esencia principal del buscador quedando de la siguiente manera.

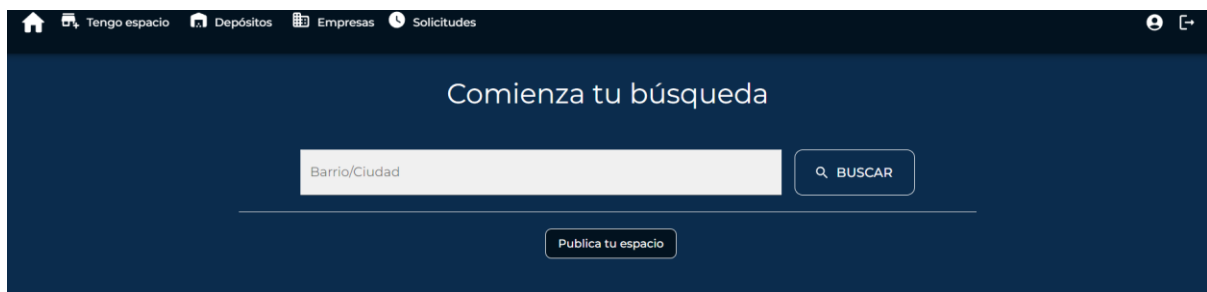
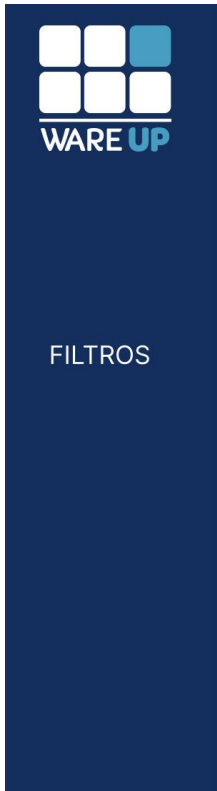


Ilustración 110: Buscador principal

### Resultado de la búsqueda

El buscador fue uno de los prototipos que se realizó avanzada la definición del producto, por ende, ya se puede visualizar la barra horizontal con las opciones del sistema. Originalmente se pensó tener un panel lateral con los filtros al igual que la página actual del cliente y mostrar simplemente el título de los espacios de almacenamiento y una foto como también se muestra hoy en día en la página actual.



[Tengo espacio](#) [Gestionar mis espacios](#) [Sobre nosotros](#) [Contactanos](#)

Fernando



Descripción del espacio  
Ver que titular



Descripción del espacio  
Ver que titular



Descripción del espacio  
Ver que titular



Descripción del espacio  
Ver que titular



Descripción del espacio  
Ver que titular



Descripción del espacio  
Ver que titular

Ilustración 111: Prototipo resultado de la búsqueda

Sin embargo, se quiso mejorar esta experiencia a una vista más moderna y más informativa. Por ende, se eligió tener un botón con todos los filtros disponibles para uso de los usuarios y por otro lado, agregar más datos a la vista rápida (así llamada por el cliente) de los distintos espacios de almacenamiento, no solo incluir su título e imagen sino también incluir información de los metros cúbicos y precio de cada uno.

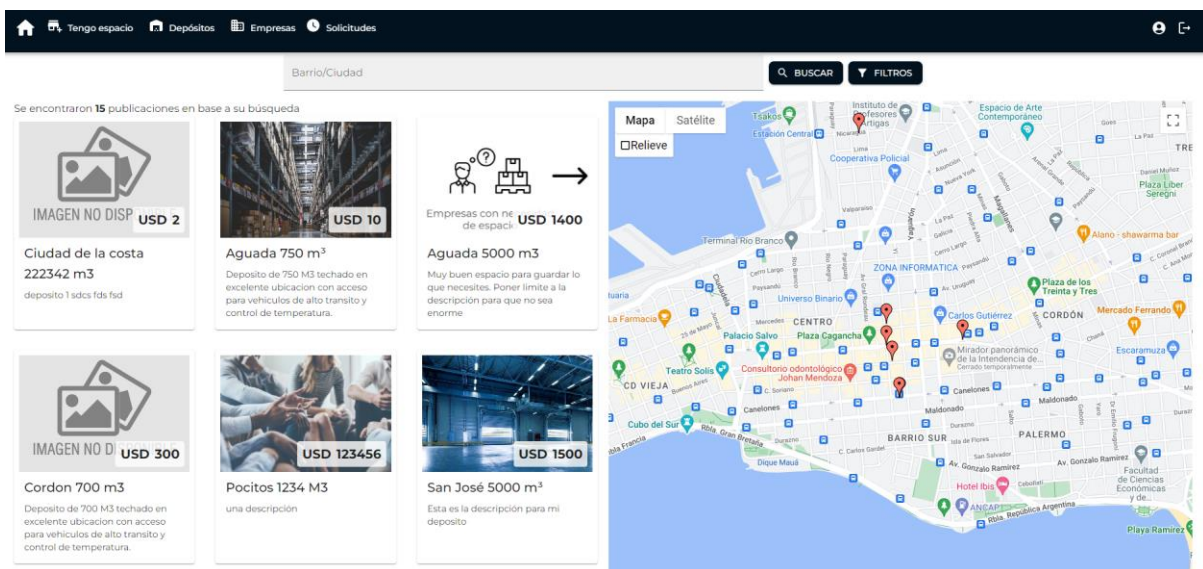


Ilustración 112: Resultado de la búsqueda

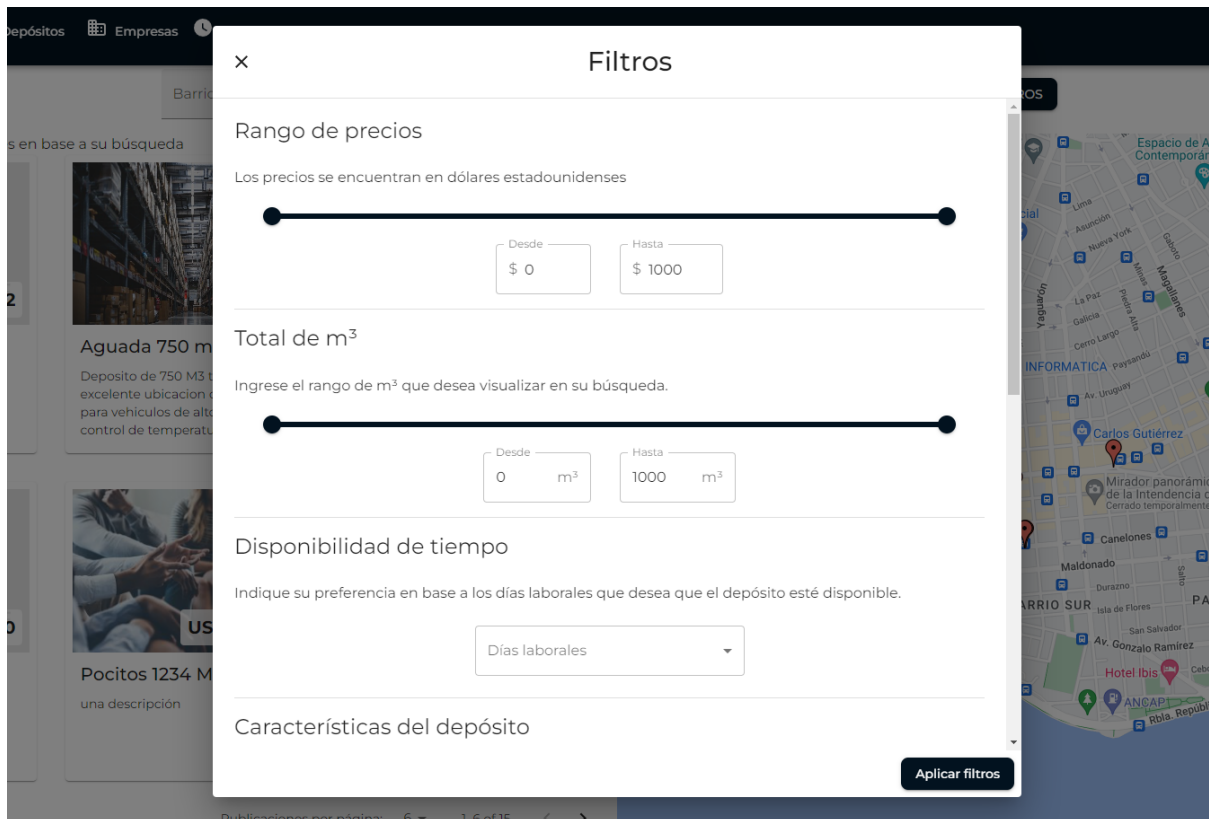


Ilustración 113: Filtros en el resultado de la búsqueda

## Publicaciones

Para visualizar la información de una publicación de espacio de almacenamiento, se intentó mantener el diseño simple de la página actual pero agregando más información de acuerdo a lo requerido por el cliente.



2.500 m<sup>3</sup>, en  
eje de ruta 5,  
Canelones

Depósito en zona de  
gran demanda, amplia  
disponibilidad de  
espacio y servicios

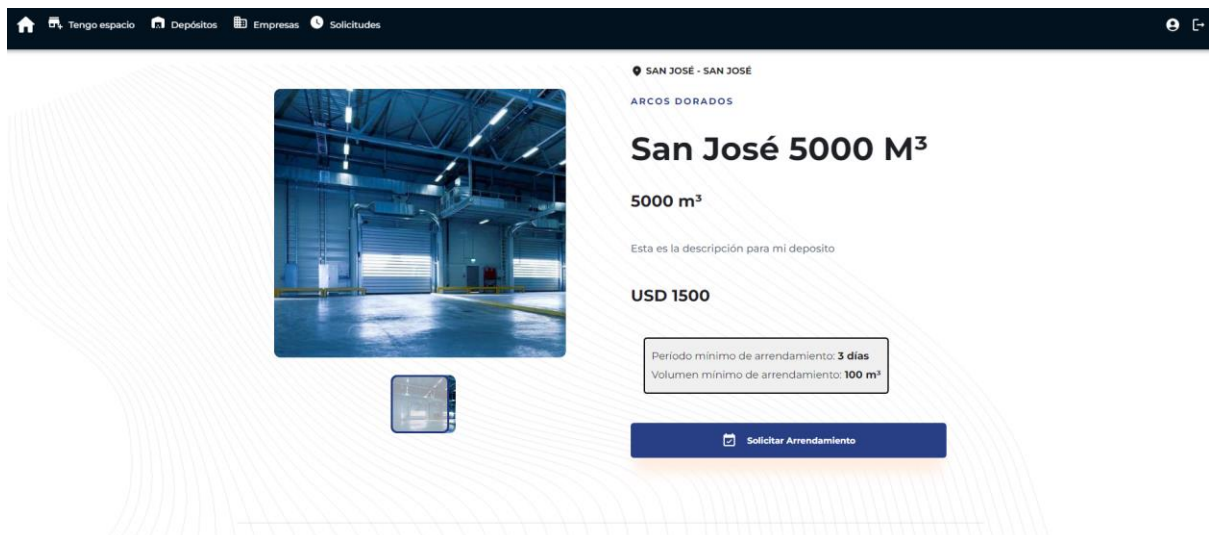
Solicitar arrendamiento

Servicios disponibles:

- Seguridad 24 hs
- Control de plagas

Ilustración 114: Prototipo de vista de publicación

A continuación se puede visualizar cómo finalmente quedó la vista de una publicación.



The screenshot shows a navigation bar with icons for 'Tengo espacio', 'Depósitos', 'Empresas', and 'Solicitudes'. The main content area features a large image of a warehouse interior with a blue floor and a mobile device displaying a map below it. To the right of the image, the listing details are as follows:

- Location: SAN JOSÉ - SAN JOSÉ
- Company: ARCOS DORADOS
- Title: **San José 5000 M<sup>3</sup>**
- Volume: **5000 m<sup>3</sup>**
- Description: Esta es la descripción para mi deposito
- Price: **USD 1500**
- Periodo mínimo de arrendamiento: **3 días**
- Volumen mínimo de arrendamiento: **100 m<sup>3</sup>**
- Call to Action: **Solicitar Arrendamiento**



Ilustración 115: Vista de una publicación

## Anexo 4 - Capturas de Artefactos de Scrum

### Product Backlog

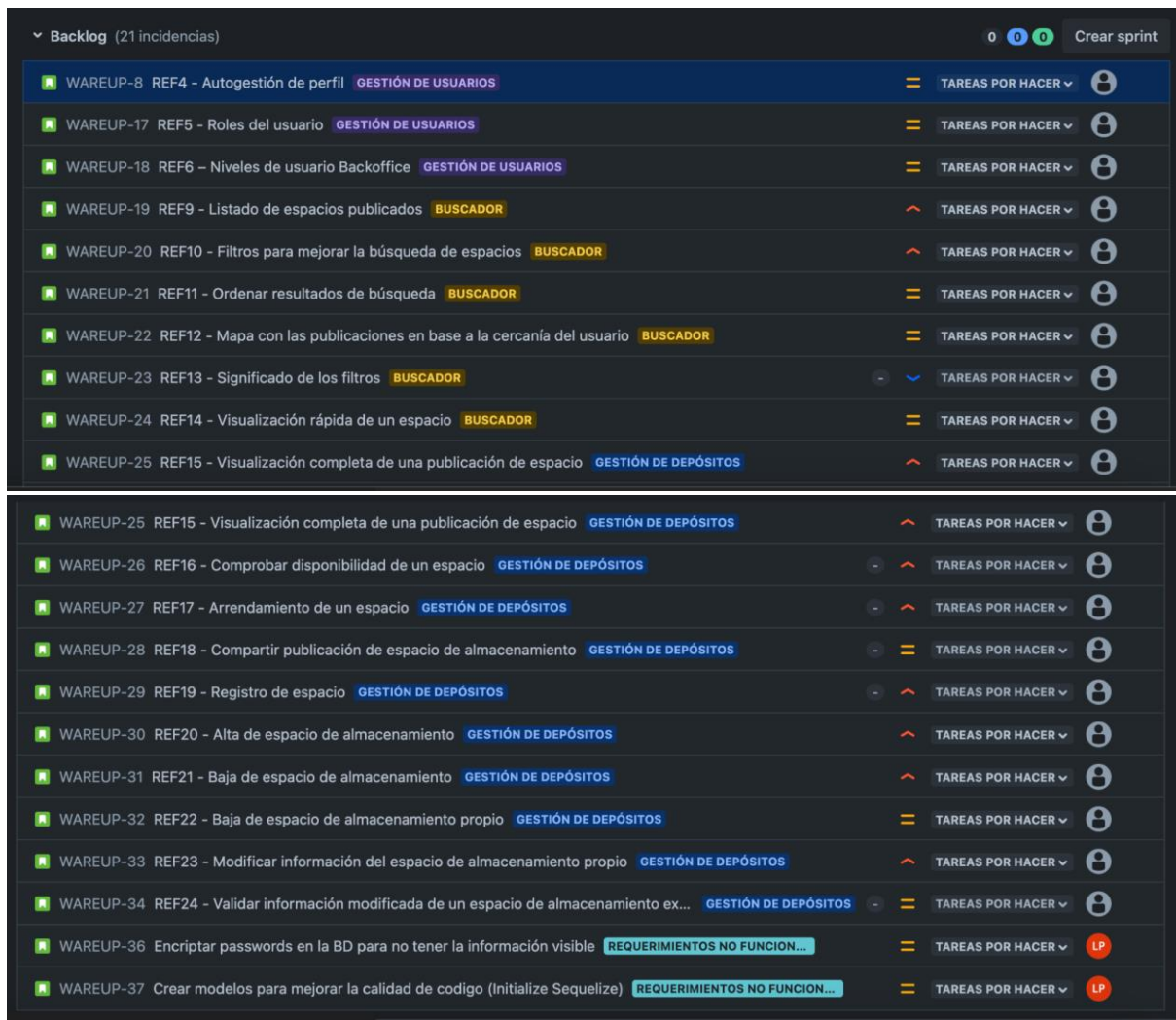


Ilustración 116: Product Backlog en Jira

## Sprint 1 Backlog

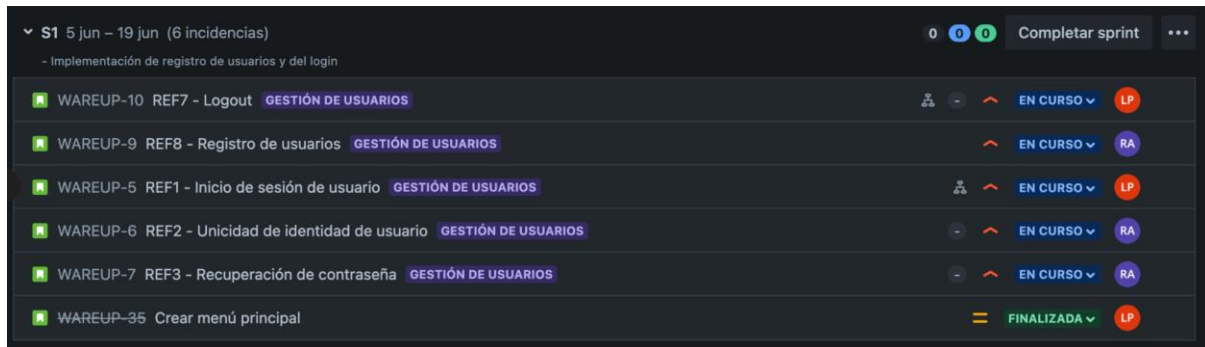


Ilustración 117: Sprint 1 Backlog en Jira

## Anexo 5 - Casos de prueba funcionales

A continuación se detallan los casos de prueba funcionales que se realizaron a lo largo de todo el proyecto. Estos fueron desarrollándose a medida que las funcionalidades se iban implementando.

Las columnas definidas fueron las siguientes:

- **Req:** corresponde al requerimiento funcional del producto.
- **Usuario:** es el tipo de usuario que tiene acceso a dicha funcionalidad o hace uso de la misma.
- **Caso de prueba:** el caso que se desea probar.
- **Pasos a seguir:** los pasos que se deben ejecutar para llevar a cabo el caso de prueba.
- **Resultado esperado:** el resultado que queremos obtener de los pasos que seguimos.
- **Ejecución inicial:** primera ejecución luego de haber realizado la funcionalidad y haber pasado las pruebas unitarias. En algunos casos puede visualizarse **FAIL** estos casos podrán verse relacionados al reporte de incidentes que se fue gestionando a lo largo de todo el proyecto.
- **Ejecución final:** el resultado obtenido post solucionar cualquier incidente que se haya encontrado.

Req.	Usuario	Caso de prueba	Pasos a seguir	Resultado esperado	Ejecución inicial	Ejecución final
REF25 - Crear menú principal	Todos	Visualizar la home de la página sin haber iniciado sesión	Acceder a la página y visualizar su información	Se mostrará información sobre el negocio de la empresa y las opciones que brinda a los usuarios. También se visualizarán los clientes que ya trabajan con la empresa.	OK	OK
REF25 - Crear menú principal	Todos	Visualizar la home de la página con sesión iniciada	Acceder a la página y visualizar su información	Se mostrará información sobre el negocio de la empresa y las opciones que brinda a los usuarios. También se visualizarán los clientes que ya trabajan con la empresa.	FAIL	OK
REF1 - Inicio de sesión de usuario	Todos	Iniciar sesión en el sistema con usuario registrado previamente	Ingresar un email y contraseña válidos	Se iniciará sesión en el sistema y se mostrará la home de la página	OK	OK
REF1 - Inicio de sesión de usuario	Todos	Iniciar sesión en el sistema sin haberse registrado	Ingresar un email y contraseña al azar	Se informará al usuario que el email no es válido.	OK	OK
REF1 - Inicio de sesión de usuario	Todos	Iniciar sesión en el sistema con contraseña incorrecta	Ingresar un email ya registrado previamente y una contraseña al azar	Se informará al usuario que la contraseña es incorrecta	OK	OK
REF1 - Inicio de sesión de usuario	Todos	Iniciar sesión con un usuario eliminado	Ingresar un email que haya sido eliminado recientemente y la contraseña correspondiente	Se mostrará el mensaje: "El usuario se encuentra eliminado" y no nos permitirá continuar con el inicio de sesión	FAIL	OK
REF3 - Recuperación de contraseña	Todos	Intentar recuperar la contraseña de	Desde el Login, debemos seleccionar la opción <b>¿Olvidaste tu contraseña?</b> e	Se mostrará el siguiente mensaje "Solicitud de reinicio de contraseña enviado exitosamente"	FAIL	OK

		un email ya registrado	ingresar un email que haya sido registrado en el sistema	y el usuario recibirá un correo a su email con el link para recuperar la contraseña		
REF3 - Recuperación de contraseña	Todos	Generar nueva contraseña a partir del link enviado en el correo	Ingresar al link enviado por email y completar los campos "Contraseña" y "Repetir contraseña" con valores que coincidan	Se mostrará el siguiente mensaje "Contraseña reiniciada exitosamente" y la contraseña será reiniciada correctamente.	OK	OK
REF3 - Recuperación de contraseña	Todos	Al intentar generar nueva contraseña no completamos los campos "Contraseña" y "Repetir contraseña" con el mismo valor	Ingresar al link enviado por email y completar los campos "Contraseña" y "Repetir contraseña" con valores diferentes	Se mostrará el siguiente mensaje "Las contraseñas tienen que ser iguales" y no permitirá continuar con el cambio requerido	OK	OK
REF2 - Unicidad de identidad de usuario REF8 - Registro de usuarios	Todos	Intentar registrar un usuario con un email ya en uso	Ingresar a "Registrarse", completar el form agregando un email que ya está en uso por otro usuario	Se mostrará el siguiente mensaje "El email ya se encuentra en uso." y no permitirá continuar con el registro.	OK	OK
REF8 - Registro de usuarios	Todos	Intentar registrar un usuario sin aceptar las políticas de privacidad	Ingresar a "Registrarse", completar el form pero no seleccionar el checkbox de "He leído y acepto las políticas de privacidad"	No permitirá continuar con el registro. El botón aparecerá deshabilitado	FAIL	OK
REF8 - Registro de usuarios	Todos	Intentar registrar un usuario sin completar uno de los campos requeridos	Ingresar a "Registrarse", completar Email, Contraseña, Repetir Contraseña, Industrias con las que estoy	Se informará al usuario que el campo Nombre es un campo obligatorio	OK	OK

			familiarizada/o pero no completar el Nombre.			
REF8 - Registro de usuarios	Todos	Intentar registrar un usuario y los campos "Contraseña" y "Repetir contraseña" no coinciden	Ingresar a "Registrarse", completar todos los campos que nos piden pero "Contraseña" y "Repetir contraseña" no coinciden en su valor.	Se informará al usuario que deben coincidir los campos	OK	OK
REF8 - Registro de usuarios	Todos	Registrar un nuevo usuario	Ingresar a "Registrarse", completar todos los campos que nos piden correctamente y aceptar las políticas de privacidad.	Se mostrará el siguiente mensaje "Usuario registrado exitosamente" y se abrirá la ventana para loguearnos	FAIL	OK
REF31 - Contactar empresa	Todos	Intentar contactar a la empresa llenando un form con la sugerencia/consulta	Desde la home, ir al pie de página y seleccionar la opción "Contáctanos", llenar los campos requeridos y seleccionar <b>Enviar</b> .	Se mostrará el mensaje "Mensaje enviado exitosamente".	OK	OK
REF31 - Contactar empresa	Todos	Intentar contactar a la empresa llenando el form con un email inválido	Desde la home, ir al pie de página, seleccionar la opción "Contáctanos" e ingresar un email inválido que no cumpla con los estándares de email.	Se mostrará mensaje "El email no es válido"	OK	OK
REF31 - Contactar empresa	Todos	Intentar contactar a la empresa sin cargar alguno de los campos requeridos	Desde la home, ir al pie de página, seleccionar la opción " <b>Contáctanos</b> " e intentar seleccionar <b>Enviar</b> sin cargar el campo <b>mensaje</b> que es requerido.	Se mostrará mensaje "Campo obligatorio"	OK	OK

REF31 - Contactar empresa	Todos	Intentar contactar a la empresa cargando texto en el campo de "Celular/Teléfono"	Desde la home, ir al pie de página, seleccionar la opción " <b>Contáctanos</b> ", ingresar un texto en el campo "Celular/Teléfono" e intentar seleccionar <b>Enviar</b> .	Se mostrará mensaje "El número de teléfono contiene caracteres no válidos"	FAIL	OK
REF25 - Crear menú principal	Cliente	Usuario intenta publicar su propio espacio sin haber iniciado sesión	En la home seleccionar el botón "Publica tu espacio"	Se mostrará la pantalla para iniciar sesión	OK	OK
REF25 - Crear menú principal	Cliente	Usuario intenta publicar su propio espacio y ya inició sesión	En la home seleccionar el botón "Publica tu espacio"	Se redirigirá a la pantalla de "Tengo espacio" para generar la solicitud de registro de un depósito.	OK	OK
REF5 - Roles de usuario	Cliente	Usuario intenta acceder a URLs que deberían solo ser vistas por los usuarios admin.	El usuario intenta acceder a <b>/admin/manage-users</b>	Se mostrará en pantalla una imagen con mensaje "404" y un botón para volver atrás	OK	OK
REF4 - Autogestión de perfil	Todos	Editar información personal de un usuario logueado	Ir a la opción " <b>Perfil</b> " sobre la barra superior, seleccionar <b>Editar perfil</b> y cambiar el nombre actual. Finalmente seleccionar <b>Guardar</b>	Se mostrará mensaje "Usuario actualizado exitosamente"	OK	OK
REF4 - Autogestión de perfil	Todos	Eliminar información personal de un usuario logueado que es requerida	Ir a la opción " <b>Perfil</b> " sobre la barra superior, seleccionar <b>Editar perfil</b> e intentar eliminar el valor en el campo <b>nombre</b> .	Se mostrará el mensaje "Campo obligatorio"	OK	OK

REF7 - Logout	Todos	Cerrar sesión cuando el usuario está logueado	Seleccionar la opción de <b>"Cerrar sesión"</b> que aparece en la barra superior.	A nivel de usuario, aparecerán nuevamente las opciones de Acceder / Registrarse. Y a nivel de navegador del usuario se eliminará el token que identificaba la sesión del usuario.	OK	OK
REF19 - Registrar solicitud de espacio de almacenamiento	Cliente	Registrar una solicitud de espacio de almacenamiento	Con un usuario logueado y una empresa ya creada previamente, seleccionar la opción <b>"Tengo espacio"</b> y completar el formulario con la información que nos piden.	Se mostrará el siguiente mensaje: <b>¡Muchas gracias por elegirnos!</b> <b>Se ha registrado exitosamente la solicitud para publicar su depósito.</b> <b>El equipo de WareUP en breve se pondrá en contacto contigo para continuar con el proceso.</b>	FAIL	OK
REF19 - Registrar solicitud de espacio de almacenamiento	Cliente	Intentar registrar una solicitud de espacio de almacenamiento sin haber registrado una empresa previamente	Con un usuario logueado seleccionar la opción <b>"Tengo espacio"</b> y al querer seleccionar una empresa, nos aparece: <b>No hay empresas registradas.</b> Intentamos continuar con el registro de todas formas.	Cuando seleccionamos el botón Registrar, nos aparecerá el mensaje de error <b>"Debe seleccionar una empresa"</b>	OK	OK
REF19 - Registrar solicitud de espacio de almacenamiento	Cliente	Intentar registrar una solicitud de espacio de almacenamiento ingresando un texto en un campo numérico	Con un usuario logueado seleccionar la opción <b>"Tengo espacio"</b> e ingresar texto en el campo <b>teléfono.</b>	Se mostrará el mensaje: "El número de teléfono contiene caracteres no válidos"	FAIL	OK
REF19 - Registrar solicitud de espacio de almacenamiento	Cliente	Intentar registrar una solicitud de espacio de	Con usuario logueado, seleccionar la opción <b>"Tengo</b>	Se mostrará el mensaje: "Debe cargar una breve descripción de su solicitud."	OK	OK

		almacenamiento sin completar uno de los campos requeridos	<b>espacio"</b> y dejar vacía la <b>descripción.</b>			
REF35 - Listado de solicitudes de registro de espacios de almacenamiento	Cliente	Visualizar las solicitudes de registro de espacios de almacenamientos hechas por el usuario	Con usuario logueado, ingresar al menú <b>Solicitudes -&gt; Solicitudes de registro.</b>	Se mostrará un listado de todos los espacios de almacenamiento que se solicitó registrar por parte del usuario que está logueado.	OK	OK
REF35 - Listado de solicitudes de registro de espacios de almacenamiento	Backoffice	Visualizar las solicitudes de registro de espacios de almacenamientos hechas por todos los usuarios	Con usuario logueado, ingresar al menú <b>Solicitudes -&gt; Solicitudes de registro de depósito.</b>	Se mostrarán todas las solicitudes que se hayan registrado hasta el momento con su correspondiente status y acciones a hacer sobre ellas, ya sea cancelar o completar la solicitud.	OK	OK
REF35 - Listado de solicitudes de registro de espacios de almacenamiento	Todos	Ordenar los registros por una columna en particular	Con usuario logueado, ingresar al menú <b>Solicitudes -&gt; Solicitudes de registro de depósito.</b> Seleccionar el nombre de la columna "Fecha creación" para ordenar los registros por dicho campo.	Se mostrarán los registros ordenados por la columna seleccionada.	FAIL	OK
REF35 - Listado de solicitudes de registro de espacios de almacenamiento	Todos	Cambiar la cantidad de registros que se muestran por página	Con usuario logueado, ingresar al menú <b>Solicitudes -&gt; Solicitudes de registro de depósito.</b> Cambiar la cantidad de solicitudes que se muestran por página a 10. (por defecto siempre son 5)	Se comenzarán a mostrar de a 10 los resultados de registros	OK	OK

REF20 - Cambiar estado de solicitud de registro de espacio de almacenamiento	Cliente	Dar de baja una solicitud que el propio usuario haya realizado y su estado es pendiente	Con usuario logueado, ingresar al menú <b>Solicitudes -&gt; Solicitudes de registro</b> . Visualizar la solicitud que desea cancelar y si su estado es Pendiente, seleccionar la opción de cancelar	Se mostrará en pantalla un diálogo donde se pide confirmación por parte del usuario para cancelar la solicitud de registro.	OK	OK
REF20 - Cambiar estado de solicitud de registro de espacio de almacenamiento	Cliente	Dar de baja una solicitud que ya fue cancelada anteriormente	Con usuario logueado, ingresar al menú <b>Solicitudes -&gt; Solicitudes de registro</b> . Visualizar la solicitud que desea cancelar y si su estado es Cancelada, seleccionar la opción de cancelar	Se mostrará en pantalla un mensaje indicando que la solicitud ya fue cancelada anteriormente	OK	OK
REF20 - Cambiar estado de solicitud de registro de espacio de almacenamiento	Cliente	Dar de baja una solicitud que ya fue completada	Con usuario logueado, ingresar al menú <b>Solicitudes -&gt; Solicitudes de registro</b> . Visualizar la solicitud que desea cancelar y si su estado es Completada, seleccionar la opción de cancelar	Se mostrará en pantalla un mensaje indicando que no se puede cambiar el estado de una solicitud ya confirmada	OK	OK
REF20 - Cambiar estado de solicitud de registro de espacio de almacenamiento	Backoffice	Dejar registro de que la solicitud de registro de espacio ya se completó y la empresa se contactó con la persona para continuar con el registro del espacio de	Con usuario logueado, ingresar al menú <b>Solicitudes -&gt; Solicitudes de registro de depósito</b> . Visualizar una solicitud que esté en estado Pendiente y seleccionar la acción <b>Aceptar</b>	Se mostrará en pantalla un diálogo que nos pedirá confirmación para completar la solicitud. Cuando seleccionemos <b>Aceptar</b> nos aparecerá un mensaje "Solicitud de registro actualizada exitosamente"	OK	OK

		almacenamiento correspondiente				
REF20 - Cambiar estado de solicitud de registro de espacio de almacenamiento	Backoffice	Intentar cancelar una solicitud que ya fue cancelada anteriormente	Con usuario logueado, ingresar al menú <b>Solicitudes</b> -> <b>Solicitudes de registro</b> . Visualizar la solicitud que desea cancelar y si su estado es Cancelada, seleccionar la opción de <b>cancelar</b>	Se mostrará en pantalla un mensaje indicando que la solicitud ya fue cancelada anteriormente	OK	OK
REF20 - Cambiar estado de solicitud de registro de espacio de almacenamiento	Backoffice	Intentar completar una solicitud que ya fue completada anteriormente	Con usuario logueado, ingresar al menú <b>Solicitudes</b> -> <b>Solicitudes de registro</b> . Visualizar la solicitud que desea completar y si su estado es Completada, seleccionar la opción de <b>aceptar</b>	Se mostrará en pantalla un mensaje indicando que la solicitud ya fue completada anteriormente	OK	OK
REF20 - Cambiar estado de solicitud de registro de espacio de almacenamiento	Backoffice	Intentar completar una solicitud que fue cancelada anteriormente	Con usuario logueado, ingresar al menú <b>Solicitudes</b> -> <b>Solicitudes de registro</b> . Visualizar la solicitud que desea completar y si su estado es Cancelada, seleccionar la opción de <b>aceptar</b>	Se mostrará en pantalla un mensaje indicando que la solicitud fue cancelada y no se puede aprobar.	OK	OK
REF36 - Listado de solicitudes de arrendamiento	Backoffice	Visualizar las solicitudes de arrendamiento que se hayan	Con usuario logueado, ingresar al menú <b>Solicitudes</b> -> <b>Solicitudes de arrendamiento</b> .	Se mostrarán todas las solicitudes que se hayan registrado en el sistema hasta el momento. Incluyendo un link a la publicación desde donde se hizo la solicitud	OK	OK

		registrado en el sistema				
REF36 - Listado de solicitudes de arrendamiento	Cliente	Visualizar las solicitudes de arrendamiento que el usuario haya hecho	Con usuario logueado, ingresar al menú <b>Solicitudes -&gt; Solicitudes de arrendamiento.</b>	Se mostrarán todas las solicitudes que el usuario haya hecho hasta el momento. Incluyendo un link a la publicación desde donde se hizo la solicitud	OK	OK
REF26 - Registrar una empresa	Cliente	Registrar una nueva empresa	Con usuario logueado, ingresa a <b>Empresas -&gt; Registrar nueva empresa.</b> Completar el formulario con información correcta y seleccionar <b>Registrar.</b>	Se mostrará el mensaje "Empresa registrada exitosamente"	OK	OK
REF26 - Registrar una empresa	Cliente	Intentar registrar una nueva empresa con un email inválido	Con usuario logueado, ingresa a <b>Empresas -&gt; Registrar nueva empresa.</b> Completar el campo <b>email</b> con un texto no válido para ser email y seleccionar <b>Registrar.</b>	Se mostrará el mensaje "El email no es válido" y no nos permitirá seguir con el registro	OK	OK
REF26 - Registrar una empresa	Cliente	Intentar registrar una nueva empresa sin completar un campo requerido	Con usuario logueado, ingresa a <b>Empresas -&gt; Registrar nueva empresa.</b> Se deja vacío el campo <b>RUT</b> y se selecciona <b>Registrar.</b>	Se mostrará el mensaje "Campo obligatorio" y no nos permitirá seguir con el registro	OK	OK
REF26 - Registrar una empresa	Cliente	Intentar registrar una nueva empresa cargando texto en el campo <b>Celular/Teléfono</b>	Con usuario logueado, ingresa a <b>Empresas -&gt; Registrar nueva empresa.</b> Completar el campo <b>celular/teléfono</b> con un texto y seleccionar <b>Registrar.</b>	Se mostrará el mensaje "El número de teléfono contiene caracteres no válidos" y no nos permitirá seguir con el registro	OK	OK

REF37 - Listado de empresas de un usuario	Cliente	Visualizar todas las empresas creadas por el usuario	Con usuario logueado, ingresa a <b>Empresas</b>	Se visualizarán todas las empresas que el usuario haya registrado con sus correspondientes acciones de editar o eliminar.	OK	OK
REF37 - Listado de empresas de un usuario	Cliente	Ordenar las empresas por una columna en particular	Con usuario logueado, ingresa a <b>Empresas</b> . Seleccionar el nombre de la columna "Fecha creación" para ordenar las empresas por dicho campo.	Se mostrarán las empresas ordenadas por la columna seleccionada.	FAIL	OK
REF37 - Listado de empresas de un usuario	Cliente	Cambiar la cantidad de empresas que se muestran por página	Con usuario logueado, ingresa a <b>Empresas</b> . Cambiar la cantidad de empresas que se muestran por página a 10. (por defecto siempre son 5)	Se comenzarán a mostrar de a 10 los resultados de empresas	OK	OK
REF27 - Baja de una empresa	Cliente	Dar de baja una empresa existente	Con usuario logueado, ingresa a <b>Empresas</b> . Visualizar la empresa que se desea dar de baja y seleccionar la acción de eliminar.	Se mostrará un diálogo que nos informa que estamos intentando eliminar una empresa y que todos los espacios de almacenamiento que la misma tenga asociados se eliminarán lógicamente. Al seleccionar Aceptar, nos mostrará un mensaje de que la empresa se eliminó exitosamente	FAIL	OK
REF27 - Baja de una empresa	Cliente	Intentar dar de baja una empresa que ya fue eliminada anteriormente	Con usuario logueado, ingresa a <b>Empresas</b> . Visualizar la empresa que se desea dar de baja donde su estado es Eliminada y seleccionar la acción de eliminar.	Se mostrará el mensaje "La empresa ya se encuentra eliminada"	FAIL	OK

REF28 - Modificar la información de una empresa	Cliente	Intentar modificar una empresa que su estado es Eliminada	Con usuario logueado, ingresa a <b>Empresas</b> . Visualizar la empresa que se desea modificar su información y su estado es eliminada. Seleccionar la acción de editar.	Se mostrará el mensaje "No se puede editar una empresa ya eliminada".	FAIL	OK
REF28 - Modificar la información de una empresa	Cliente	Modificar la información de una empresa activa	Con usuario logueado, ingresa a <b>Empresas</b> . Visualizar la empresa que se desea modificar su información y su estado es Activa. Seleccionar la acción de editar y cambiar el <b>nombre completo</b> del representante de la empresa.	Se mostrará el mensaje "Empresa actualizada exitosamente" y se podrá visualizar el cambio al actualizar la página en la información de la empresa.	OK	OK
REF28 - Modificar la información de una empresa	Cliente	Intentar eliminar un campo requerido al modificar la información de una empresa activa	Con usuario logueado, ingresa a <b>Empresas</b> . Visualizar la empresa que se desea modificar su información y su estado es Activa. Seleccionar la acción de editar y eliminar el valor que tiene en el campo <b>RUT</b> .	Se mostrará el mensaje "Campo obligatorio"	OK	OK
REF33 - Listado de usuarios registrados	Backoffice	Visualizar todos los usuarios registrados en el sistema	Con usuario logueado, ingresa a <b>Usuarios</b> .	Se visualizarán todos los usuarios que se hayan registrado hasta el momento, con sus correspondientes acciones de editar o eliminar.	OK	OK
REF33 - Listado de usuarios registrados	Backoffice	Ordenar los usuarios por una	Con usuario logueado, ingresa a <b>Usuarios</b> . Seleccionar el nombre de la	Se mostrarán los usuarios ordenados por la columna seleccionada.	OK	OK

		columna en particular	columna "Fecha creación" para ordenar los usuarios por dicho campo.			
REF33 - Listado de usuarios registrados	Backoffice	Cambiar la cantidad de usuarios que se muestran por página	Con usuario logueado, ingresa a <b>Usuarios</b> . Cambiar la cantidad de usuarios que se muestran por página a 10. (por defecto siempre son 5)	Se comenzarán a mostrar de a 10 los resultados de usuarios.	OK	OK
REF21 - Alta de espacio de almacenamiento	Backoffice	Registrar espacio de almacenamiento	Con usuario logueado, y al menos una empresa ya creada, seleccionar el menú <b>Depósitos -&gt; Publicar depósito</b> . Completar la información del espacio de almacenamiento con información correcta, luego seleccionar al menos un servicio y seleccionar <b>Finalizar</b> .	Se mostrará el mensaje "Depósito registrado exitosamente" y nos mostrará las opciones para agregar imágenes o agregar disponibilidad al espacio de almacenamiento.	OK	OK
REF21 - Alta de espacio de almacenamiento	Backoffice	Se intenta registrar un espacio de almacenamiento sin tener empresas registradas	Con usuario logueado, seleccionar el menú <b>Depósitos -&gt; Publicar depósito</b> . Completar la información del espacio de almacenamiento menos la empresa ya que no hay empresas registradas y se intenta pasar al siguiente paso.	El botón <b>Siguiente</b> estará deshabilitado y se mostrará el mensaje "Debe seleccionar una empresa".	OK	OK
REF21 - Alta de espacio de almacenamiento	Backoffice	Se intenta registrar un espacio de almacenamiento	Con usuario logueado, seleccionar el menú <b>Depósitos -&gt; Publicar</b>	El botón <b>Siguiente</b> estará deshabilitado y se mostrará el mensaje "Debe cargar el precio".	OK	OK

		sin completar uno de los campos requeridos en la información del espacio.	<b>depósito.</b> Se deja vacío el precio y se intenta pasar al siguiente paso.			
REF21 - Alta de espacio de almacenamiento	Backoffice	Se intenta registrar un espacio de almacenamiento con una descripción que supere los 200 caracteres.	Con usuario logueado, seleccionar el menú <b>Depósitos -&gt; Publicar depósito.</b> Se agrega una descripción que supere los 200 caracteres y se intenta pasar al siguiente paso.	El botón <b>Siguiente</b> estará deshabilitado y se mostrará el mensaje "La descripción no debe superar los 200 caracteres".	OK	OK
REF21 - Alta de espacio de almacenamiento	Backoffice	Se intenta registrar un espacio de almacenamiento sin seleccionar un servicio	Con usuario logueado, seleccionar el menú <b>Depósitos -&gt; Publicar depósito.</b> Se completa la información del espacio de almacenamiento, se selecciona <b>Siguiente</b> y no se selecciona ningún servicio.	El botón <b>Finalizar</b> estará deshabilitado	OK	OK
REF34 - Listado de espacios de almacenamiento registrados	Cliente	Visualizar los espacios de almacenamiento propios del usuario	Con usuario logueado, ingresar a <b>Depósitos -&gt; Listado de depósitos.</b>	Se mostrarán los espacios de almacenamiento registrados para el usuario logueado.	OK	OK
REF34 - Listado de espacios de almacenamiento registrados	Backoffice	Visualizar los espacios de almacenamiento registrados	Con usuario logueado, ingresar a <b>Depósitos -&gt; Listado de depósitos.</b>	Se mostrarán todos los espacios de almacenamiento registrados en el sistema hasta el momento.	OK	OK
REF34 - Listado de espacios de	Todos	Ordenar los espacios de almacenamiento	Con usuario logueado, ingresar a <b>Depósitos -&gt; Listado de depósitos.</b>	Se mostrarán los espacios de almacenamiento ordenados por la columna seleccionada.	OK	OK

almacenamiento registrados		por una columna en particular	Seleccionar el nombre de la columna "Fecha creación" para ordenar los espacios de almacenamiento por dicho campo.			
REF34 - Listado de espacios de almacenamiento registrados	Todos	Cambiar la cantidad de espacios de almacenamiento que se muestran por página	Con usuario logueado, ingresar a <b>Depósitos -&gt; Listado de depósitos.</b> Cambiar la cantidad de espacios de almacenamiento que se muestran por página a 10. (por defecto siempre son 5)	Se comenzarán a mostrar de a 10 los resultados de espacios de almacenamiento.	OK	OK
REF32 - Métricas internas del sistema para usuario backoffice	Backoffice	Visualizar el estado actual de las entidades registradas en el sistema	Con usuario logueado, ingresar a <b>Métricas.</b>	Se mostrarán gráficas con la cantidad de entidades registradas por status, cantidad de entidades registradas por mes para cada tipo de entidades que hay en el sistema.	OK	OK
REF22 - Baja de espacio de almacenamiento	Todos	Dar de baja un espacio de almacenamiento con estado diferente de Eliminado	Con usuario logueado, ingresar a <b>Depósitos -&gt; Listado de depósitos.</b> Visualizar el espacio de almacenamiento que se desea dar de baja y seleccionar la acción <b>Eliminar.</b>	Se mostrará un diálogo que nos informa de que estamos eliminando un espacio de almacenamiento y nos pide confirmación. Al seleccionar <b>Aceptar</b> se mostrará el mensaje "Depósito eliminado exitosamente", se eliminará el espacio de almacenamiento y el estado se verá reflejado al actualizar la página.	OK	OK
REF22 - Baja de espacio de almacenamiento	Todos	Dar de baja un espacio de almacenamiento	Con usuario logueado, ingresar a <b>Depósitos -&gt; Listado de depósitos.</b> Visualizar el espacio de	Se mostrará el mensaje "No se puede eliminar un depósito que ya fue eliminado"	OK	OK

		con estado Eliminado	almacenamiento que se desea dar de baja y seleccionar la acción <b>Eliminar.</b>			
REF23 - Modificar información del espacio de almacenamiento	Todos	Modificar la información básica de un espacio de almacenamiento	Con usuario logueado, ingresar a <b>Depósitos.</b> Seleccionar uno de los espacios de almacenamiento que queramos modificar, ir a las <b>Acciones</b> y seleccionar <b>editar información básica.</b> Modificar la dirección del espacio de almacenamiento por una correcta y su correspondiente Barrio/Ciudad en caso de que lo amerite. Seleccionar <b>Guardar.</b>	Se mostrará el mensaje "Depósito actualizado exitosamente" y se verá el cambio al actualizar la página.	OK	OK
REF23 - Modificar información del espacio de almacenamiento	Todos	Eliminar información requerida de un espacio de almacenamiento	Con usuario logueado, ingresar a <b>Depósitos.</b> Seleccionar uno de los espacios de almacenamiento que queramos modificar, ir a las <b>Acciones</b> y seleccionar <b>editar información básica.</b> Eliminar la dirección del espacio de almacenamiento y seleccionar <b>Guardar.</b>	Se mostrará el mensaje "Debe completarse la dirección del depósito"	OK	OK
REF23 - Modificar información del espacio de almacenamiento	Todos	Modificar los servicios actuales de un espacio de almacenamiento	Con usuario logueado, ingresar a <b>Depósitos.</b> Seleccionar uno de los espacios de almacenamiento	Se mostrará el mensaje "Servicios actualizados exitosamente" y se actualizaron los servicios del espacio de almacenamiento	FAIL	OK

			que queramos modificar, ir a las <b>Acciones</b> y seleccionar <b>editar servicios</b> . Agregar un nuevo servicio y seleccionar <b>Guardar cambios</b> .			
REF23 - Modificar información del espacio de almacenamiento	Todos	Eliminar los servicios actuales de un espacio de almacenamiento	Con usuario logueado, ingresar a <b>Depósitos</b> . Seleccionar uno de los espacios de almacenamiento que queramos modificar, ir a las <b>Acciones</b> y seleccionar <b>editar servicios</b> . Eliminar todos los servicios seleccionados e intentar seleccionar <b>Guardar cambios</b> .	El botón <b>Guardar cambios</b> estará deshabilitado y se mostrará el mensaje "Debe seleccionar al menos un servicio"	OK	OK
REF23 - Modificar información del espacio de almacenamiento	Todos	Modificar la información básica de un espacio de almacenamiento con estado Eliminado	Con usuario logueado, ingresar a <b>Depósitos</b> . Seleccionar uno de los espacios de almacenamiento que queramos modificar, ir a las <b>Acciones</b> y seleccionar <b>editar información básica</b> .	Se mostrará el mensaje "No se puede editar un depósito ya eliminado"	OK	OK
REF23 - Modificar información del espacio de almacenamiento	Todos	Modificar los servicios de un espacio de almacenamiento con estado Eliminado	Con usuario logueado, ingresar a <b>Depósitos</b> . Seleccionar uno de los espacios de almacenamiento que queramos modificar, ir a las <b>Acciones</b> y seleccionar <b>editar servicios</b> .	Se mostrará el mensaje "No se puede editar un depósito ya eliminado"	OK	OK

REF38 - Modificar información de un usuario registrado	Backoffice	Modificar el nombre de un usuario con estado activo	Con usuario logueado, ingresa a <b>Usuarios</b> . Visualizar el usuario que queremos modificar, seleccionar <b>Editar perfil</b> y cambiar el nombre del usuario.	Se mostrará el mensaje "Usuario actualizado exitosamente" y se actualizará la información del usuario	OK	OK
REF38 - Modificar información de un usuario registrado	Backoffice	Intentar eliminar un campo requerido al modificar la información de un usuario activo	Con usuario logueado, ingresa a <b>Usuarios</b> . Visualizar el usuario que queremos modificar, seleccionar <b>Editar perfil</b> y eliminar el nombre del usuario.	Se mostrará el mensaje "Campo obligatorio" y no nos dejará continuar con la modificación.	OK	OK
REF38 - Modificar información de un usuario registrado	Backoffice	Intentar modificar un usuario eliminado	Con usuario logueado, ingresa a <b>Usuarios</b> . Visualizar el usuario que queremos modificar con estado Eliminado y seleccionar <b>Editar perfil</b> .	Se mostrará el mensaje "No se puede editar un usuario que fue eliminado".	OK	OK
REF39 - Dar de baja un usuario registrado	Backoffice	Dar de baja un usuario activo	Con usuario logueado, ingresa a <b>Usuarios</b> . Visualizar el usuario que queremos dar de baja y seleccionar <b>Eliminar</b> .	Se mostrará un diálogo de confirmación donde nos mencionan que estaremos eliminando un usuario. En caso de confirmar, el usuario se eliminará lógicamente del sistema.	OK	OK
REF39 - Dar de baja un usuario registrado	Backoffice	Dar de baja un usuario ya eliminado	Con usuario logueado, ingresar a <b>Usuarios</b> . Visualizar el usuario que queremos dar de baja y seleccionar <b>Eliminar</b> .	Se mostrará el mensaje "El usuario ya se encuentra eliminado."	OK	OK

REF29 - Cargar disponibilidad de un espacio de almacenamiento	Todos	Agregar un nuevo rango de disponibilidad para un espacio de almacenamiento	Con usuario logueado, ingresar a <b>Depósitos</b> . Visualizar el espacio de almacenamiento que queremos gestionar y seleccionar la acción <b>Agregar disponibilidad</b> . Ingresar los campos "Desde", "Hasta" y "Total metros cúbicos" con valores correctos.	Se mostrará el mensaje "Disponibilidad agregada exitosamente"	OK	OK
REF29 - Cargar disponibilidad de un espacio de almacenamiento	Todos	Agregar un nuevo rango de disponibilidad para un espacio de almacenamiento cuando ya existe otro y las fechas coinciden.	Con usuario logueado y rango de disponibilidad ya ingresado anteriormente para las fechas (25-10-2023 hasta 11-01-2024) ingresar a <b>Depósitos</b> . Visualizar el espacio de almacenamiento que queremos gestionar y seleccionar la acción <b>Agregar disponibilidad</b> . Ingresar los siguientes datos: Desde: 16-10-2023 Hasta: 11-09-2023 Total metros cúbicos: 4000	Se mostrará el mensaje "Ya hay un calendario existente" y no se registrará el rango de disponibilidad	FAIL	OK
REF29 - Cargar disponibilidad de un espacio de almacenamiento	Todos	Intentar agregar disponibilidad para un espacio de almacenamiento pero el rango de fechas no es correcto	Con usuario logueado e ingresar a <b>Depósitos</b> . Visualizar el espacio de almacenamiento que queremos gestionar y seleccionar la acción <b>Agregar disponibilidad</b> . Ingresar los siguientes datos:	Se mostrará el mensaje "La fecha de fin debe ser posterior o igual a la fecha de inicio" y no nos permitirá continuar con el registro.	OK	OK

			Desde: 16-10-2023 Hasta: 10-10-2023 Total metros cúbicos: 4000			
REF29 - Cargar disponibilidad de un espacio de almacenamiento	Todos	Intentar agregar disponibilidad para un espacio de almacenamiento pero el total de metros cúbicos supera lo configurado para el espacio de almacenamiento	Con usuario logueado e ingresar a <b>Depósitos</b> . Visualizar el espacio de almacenamiento que queremos gestionar y seleccionar la acción <b>Agregar disponibilidad</b> . Ingresar los siguientes datos: Desde: 16-10-2023 Hasta: 19-10-2023 Total metros cúbicos: 6000	Se mostrará el mensaje "Total M³ no puede ser superior a 5000" (donde 5000 fue lo que se configuró originalmente como Total de metros cúbicos. No nos permitirá continuar con el registro	FAIL	OK
REF29 - Cargar disponibilidad de un espacio de almacenamiento	Todos	Intentar agregar disponibilidad para un espacio de almacenamiento pero la fecha inicial "Desde" es anterior a la fecha actual.	Con usuario logueado e ingresar a <b>Depósitos</b> . Visualizar el espacio de almacenamiento que queremos gestionar y seleccionar la acción <b>Agregar disponibilidad</b> . Intentar seleccionar una fecha anterior a la actual.	El calendario mostrará deshabilitados los días anteriores a la fecha actual	OK	OK
REF29 - Cargar disponibilidad de un espacio de almacenamiento	Todos	Intentar agregar disponibilidad a un espacio de almacenamiento con estado Eliminado	Con usuario logueado e ingresar a <b>Depósitos</b> . Visualizar el espacio de almacenamiento que queremos gestionar y	Se mostrará el mensaje "No se puede editar un depósito ya eliminado." y no nos dejará continuar con la acción	OK	OK

			seleccionar la acción <b>Agregar disponibilidad.</b>			
REF30 - Cargar imágenes en un espacio de almacenamiento	Todos	Visualizar las imágenes actuales de un espacio de almacenamiento con imágenes ya cargadas	Con usuario logueado e ingresar a <b>Depósitos.</b> Visualizar el espacio de almacenamiento que queremos gestionar y seleccionar la acción <b>Gestionar imágenes.</b>	Se mostrará en un carrusel donde van pasando todas las imágenes del espacio de almacenamiento acompañado del botón de <b>Seleccionar imágenes</b> para cargar nuevas imágenes	OK	OK
REF30 - Cargar imágenes en un espacio de almacenamiento	Todos	Visualizar las imágenes actuales de un espacio de almacenamiento que no tiene imágenes cargadas.	Con usuario logueado e ingresar a <b>Depósitos.</b> Visualizar el espacio de almacenamiento que queremos gestionar y seleccionar la acción <b>Gestionar imágenes.</b>	Se mostrará el mensaje "No se han agregado imágenes aún..." acompañado del botón <b>Seleccionar imágenes</b> para cargar imágenes	OK	OK
REF30 - Cargar imágenes en un espacio de almacenamiento	Todos	Agregar nuevas imágenes a un espacio de almacenamiento	Con usuario logueado e ingresar a <b>Depósitos.</b> Visualizar el espacio de almacenamiento que queremos gestionar y seleccionar la acción <b>Gestionar imágenes.</b> Hacer click en el botón <b>Seleccionar imágenes</b> y seleccionar una o más imágenes que estén guardadas en nuestra pc. Por último, hacer click en <b>Subir.</b>	Se mostrará el mensaje "Se han agregado imágenes al depósito exitosamente" y se agregarán las imágenes al espacio de almacenamiento existente.	FAIL	OK
REF30 - Cargar imágenes en un	Todos	Intentar agregar nuevas imágenes a un espacio de	Con usuario logueado e ingresar a <b>Depósitos.</b> Visualizar el espacio de	Se mostrará el mensaje "No se puede agregar imágenes a un depósito que fue eliminado."	OK	OK

espacio de almacenamiento		almacenamiento con estado Eliminado	almacenamiento que queremos gestionar y seleccionar la acción <b>Gestionar imágenes.</b>			
REF40 - Visualizar la disponibilidad configurada para un espacio de almacenamiento	Todos	Visualizar la disponibilidad configurada para un espacio de almacenamiento	Con usuario logueado e ingresar a <b>Depósitos.</b> Visualizar el espacio de almacenamiento que queremos gestionar y seleccionar la acción <b>Ver configuración de disponibilidad.</b>	Se mostrará una tabla con los distintos rangos de fechas configuradas de disponibilidad y su correspondiente total de metros cúbicos.	OK	OK
REF9 - Buscador de espacios publicados	Todos	Visualizar todos los espacios publicados sin ninguna especificación	En el buscador principal de la home de la página seleccionar <b>Buscar</b> sin ingresar ningún Barrio/Ciudad.	Se mostrarán todas las publicaciones existentes en el sistema (espacios de almacenamientos) acompañado de un mapa donde se marcarán las ubicaciones de los espacios de almacenamiento.	OK	OK
REF9 - Buscador de espacios publicados	Todos	Visualizar todos los espacios publicados para un barrio en particular y tiene resultados	En el buscador principal de la home de la página, ingresar un barrio donde sabemos que ya hay espacios registrados y seleccionar <b>Buscar.</b>	Se mostrarán todas las publicaciones existentes para ese barrio ingresado acompañado de un mapa donde se marcarán las ubicaciones de los espacios de almacenamiento.	OK	OK
REF9 - Buscador de espacios publicados	Todos	Buscar espacios publicados en un barrio donde aún no se hayan registrado espacios de almacenamiento.	En el buscador principal de la home de la página, ingresar un barrio donde sabemos que no hay espacios registrados aún y seleccionar <b>Buscar.</b>	Se mostrará el mensaje: "No se encontraron resultados para su búsqueda. ¿No encuentras lo que buscas? Explora todas nuestras publicaciones"	OK	OK

				Donde Explora todas nuestras publicaciones nos redirigirá al buscador con todas las publicaciones existentes en el sistema sin ningún filtro.		
REF12 - Mapa con información de las publicaciones	Todos	Visualizar el mapa con las direcciones de los espacios publicados	En el buscador principal de la home de la página, ingresar un barrio donde sabemos que ya hay espacios registrados y seleccionar <b>Buscar</b> .	Se mostrará un mapa acompañado del listado de las publicaciones donde se marcarán las direcciones de los espacios de almacenamiento y cada marca tendrá información básica de la publicación: <b>título, descripción y precio</b> .	OK	OK
REF14 - Visualización rápida de un espacio	Todos	Visualizar la información básica de un espacio publicado al utilizar el buscador	En el buscador principal de la home de la página, seleccionar el botón <b>Buscar</b> y en el listado de publicaciones visualizar la información básica de las mismas.	Por cada publicación, se mostrará el <b>título</b> , la <b>descripción</b> , una <b>imagen</b> base y su <b>precio</b> dando un indicio a los usuarios de los datos más básicos de la publicación.	OK	OK
REF10 - Filtros para mejorar la búsqueda de espacios	Todos	Aplicar filtros de rangos a la búsqueda realizada y encontrar publicaciones que cumplan con los filtros aplicados	En el buscador de la home de la página, seleccionar el botón <b>Buscar</b> , y cuando nos aparece el listado de las publicaciones, seleccionar <b>Filtros</b> y agregar un <b>rango de precios</b> para que nos aparezcan las publicaciones que estén dentro de dicho rango. Por último, seleccionar <b>Aplicar filtros</b> .	Se actualizará el listado de publicaciones y el mapa con los espacios que coinciden con el filtro realizado y en la URL del navegador se podrán visualizar los filtros aplicados.	OK	OK

REF10 - Filtros para mejorar la búsqueda de espacios	Todos	Aplicar filtros de características a la búsqueda realizada y encontrar publicaciones que cumplan con los filtros aplicados	En el buscador de la home de la página, seleccionar el botón <b>Buscar</b> , y cuando nos aparece el listado de las publicaciones, seleccionar <b>Filtros</b> y seleccionar alguno o algunos de los checkboxes de características de un espacio para que nos aparezcan las publicaciones que contengan dicha o dichas característica/s. Por último, seleccionar <b>Aplicar filtros</b> .	Se actualizará el listado de publicaciones y el mapa con los espacios que coinciden con el filtro realizado y en la URL del navegador se podrán visualizar los filtros aplicados.	OK	OK
REF10 - Filtros para mejorar la búsqueda de espacios	Todos	Aplicar filtros a la búsqueda realizada y no encontrar ninguna publicación que cumpla con los resultados	En el buscador de la home de la página, seleccionar el botón <b>Buscar</b> , y cuando nos aparece el listado de las publicaciones, seleccionar <b>Filtros</b> y seleccionar todos los filtros posibles. Por último, seleccionar <b>Aplicar filtros</b> .	Se mostrará en pantalla el mensaje: "No se encontraron resultados para su búsqueda. ¿No encuentras lo que buscas? Explora todas nuestras publicaciones" Donde <b>Explora todas nuestras publicaciones</b> nos redirigirá al buscador sin filtros.	OK	OK
REF15 - Visualización completa de una publicación de espacio	Todos	Visualizar la publicación de un espacio de almacenamiento desde el listado de depósitos registrados	Con usuario logueado, ingresar a <b>Depósitos</b> . Visualizar el espacio de almacenamiento que queremos revisar y seleccionar la acción <b>Vista previa</b> .	Se abrirá una nueva pestaña en el navegador con la publicación del espacio de almacenamiento donde se podrá visualizar la siguiente información: imágenes del espacio, ubicación (ciudad y departamento), empresa que lo gestiona, título, total de metros cúbicos, descripción, precio, información de período y volumen mínimo de	OK	OK

				arrendamiento, servicios y la opción de arrendamiento.		
REF15 - Visualización completa de una publicación de espacio	Todos	Visualizar la publicación de un espacio de almacenamiento desde el buscador	En el buscador principal de la home de la página, seleccionar el botón <b>Buscar</b> y en el listado de publicaciones seleccionar una de las publicaciones.	Se abrirá una nueva pestaña en el navegador con la publicación del espacio de almacenamiento donde se podrá visualizar la siguiente información: imágenes del espacio, ubicación (ciudad y departamento), empresa que lo gestiona, título, total de metros cúbicos, descripción, precio, información de período y volumen mínimo de arrendamiento, servicios y la opción de arrendamiento.	OK	OK
REF15 - Visualización completa de una publicación de espacio	Todos	Visualizar una publicación de espacio de almacenamiento que no tiene imágenes	En el buscador principal de la home de la página, seleccionar el botón <b>Buscar</b> y en el listado de publicaciones seleccionar una de las publicaciones.	Se abrirá una nueva pestaña en el navegador con la publicación del espacio de almacenamiento donde se podrá visualizar la siguiente información: a diferencia de los casos anteriores aparecerá una imagen por defecto que dice " <b>Imagen no disponible</b> ", ubicación (ciudad y departamento), empresa que lo gestiona, título, total de metros cúbicos, descripción, precio, información de período y volumen mínimo de arrendamiento, servicios y la opción de arrendamiento.	OK	OK

REF15 - Visualización completa de una publicación de espacio	Todos	Visualizar mejor las imágenes de la publicación	Acceder a una publicación y seleccionar las imágenes	Se mostrarán las imágenes en un tamaño mayor y tendremos la opción de ir pasando a través de todas las imágenes.	OK	OK
REF17 - Generar solicitud de arrendamiento de un espacio de almacenamiento	Cliente	Solicitar arrendar un espacio de almacenamiento	Acceder a una publicación y seleccionar "Solicitar arrendamiento". Agregar las fechas de inicio y fin y el espacio que deseamos arrendar.	Se mostrará el mensaje <b>"¡Muchas gracias por elegirnos! Se ha registrado exitosamente la solicitud para arrendar el espacio. El equipo de WareUP en breve se pondrá en contacto contigo para continuar con el proceso."</b>	OK	OK
REF17 - Generar solicitud de arrendamiento de un espacio de almacenamiento	Cliente	Intentar solicitar el arrendamiento un espacio de almacenamiento ingresando un total de metros cúbicos mayor al existente en el espacio de almacenamiento	Acceder a una publicación y seleccionar " <b>Solicitar arrendamiento</b> ". En espacio a arrendar agregar 1000 (donde el total de metros cúbicos de la publicación es 500). Seleccionar <b>Confirmar</b> .	Se mostrará el mensaje "El valor debe ser menor o igual a X" donde X es el total de metros cúbicos definidos en el espacio de almacenamiento	OK	OK
REF17 - Generar solicitud de arrendamiento de un espacio de almacenamiento	Cliente	Intentar solicitar el arrendamiento un espacio de almacenamiento ingresando un rango de fechas inferior al periodo mínimo definido en la publicación	Acceder a una publicación y seleccionar " <b>Solicitar arrendamiento</b> ". En las fechas de inicio y fin ingresar: Desde: 10-10-2023 Hasta: 15-10-2023 (El período mínimo de arrendamiento es 10 días de acuerdo a la información de la publicación)	Se mostrará el mensaje "El período de reserva debe ser mayor o igual a X" donde X es el período mínimo de reserva configurado en el espacio de almacenamiento	OK	OK

			Seleccionar <b>Confirmar</b> .			
REF17 - Generar solicitud de arrendamiento de un espacio de almacenamiento	Cliente	Intentar solicitar el arrendamiento un espacio de almacenamiento ingresando un total de metros cúbicos inferior al definido en la publicación	Acceder a una publicación y seleccionar " <b>Solicitar arrendamiento</b> ". En espacio a arrendar ingresar: 50. (El volumen mínimo de arrendamiento es 100 metros cúbicos de acuerdo a la información de la publicación) Seleccionar <b>Confirmar</b> .	Se mostrará el mensaje "El valor debe ser mayor o igual a X" donde X es el volumen mínimo de reserva configurado en el espacio de almacenamiento	OK	OK
REF17 - Generar solicitud de arrendamiento de un espacio de almacenamiento	Cliente	Intentar solicitar el arrendamiento un espacio de almacenamiento que está pausado o eliminado	Acceder a una publicación y buscar el botón " <b>Solicitar arrendamiento</b> "	En la publicación no se mostrará el botón "Solicitar arrendamiento" sino que aparecerá como "Publicación pausada" y no se podrá realizar ninguna acción sobre la misma	OK	OK

Tabla 72: Casos de prueba funcionales

## Anexo 6 - Registro de incidentes

Los datos que se registraron para cada incidente o mejora son los siguientes:

- **Requerimiento asociado:** Indica a qué requerimiento está asociado el elemento.
- **Categoría:** Se detalla si se trata de un incidente o un cambio/mejora.

- **Tipo:** Interno: del equipo, Externo: enviado por el cliente.
- **Área:** Si es un problema de backend, de frontend o del ambiente de prueba.
- **Nombre:** Nombre descriptivo.
- **Descripción:** Descripción del caso, detallando los pasos a seguir para poder reproducirlo.
- **Severidad:** Si es un incidente, se detalla la criticidad: Crítica, Alta, Media o Baja. Si es un cambio/mejora, no se define.
- **Detectado:** Sprint en el cual se detectó.
- **Resuelto:** Sprint en el cual se corrigió.

Requerimiento asociado	Categoría	Tipo	Área	Nombre	Descripción	Severidad	Detectado	Resuelto
REF31 - Contactar empresa	Mejora	Interno	Backend	Falta de campo "Asunto" en pantalla de contáctenos	Se desea agregar un nuevo campo al formulario de contactanos donde el usuario podrá agregar un asunto a su consulta	-	Sprint 3	Sprint 3
REF3 - Recuperación de contraseña	Incidente	Interno	Backend	Error al obtener la información de RecoverPassword	RecoverPassword tiene un campo data incorrecto se desea eliminarlo para mantener todos las responses de los endpoints con la misma estructura	Baja	Sprint 1	Sprint 3
REF1 - Inicio de sesión de usuario	Incidente	Interno	Backend	Usuario eliminado intenta loguearse	Si el usuario fue eliminado, no se puede loguear al sistema	Alta	Sprint 1	Sprint 1
REF21 - Alta de espacio de almacenamiento	Mejora	Interno	Backend	Mejorar grupos de servicios	Se desea mejorar la experiencia de los grupos de servicios ya que	-	Sprint 4	Sprint 5

					tenemos muchos en "Varios" y tal vez se podrían clasificar mejor			
REF21 - Alta de espacio de almacenamiento	Mejora	Interno	Frontend	Poner en rojo cuando falla algo del form Agregar depósito en los selectBox	Hoy en día se pone el mensaje de error pero quedan en negro, no en rojo. Se desea mejorar esto para que sea más visible el error	-	Sprint 4	Sprint 5
REF1 - Inicio de sesión de usuario	Incidente	Interno	Frontend	Error 404 cuando usuario no está logueado	Cuando un usuario no está logueado y quiere acceder a alguna de las páginas que necesita login, mostrarles mensaje de error	Alta	Sprint 1	Sprint 1
REF21 - Alta de espacio de almacenamiento	Mejora	Interno	Backend	Guardado de userId en register deposit	Se desea que el userId no sea pasado desde el FE sino que sea buscado a través del companyId	-	Sprint 4	Sprint 4
REF3 - Recuperación de contraseña	Incidente	Interno	Backend	Email con link para recuperar contraseña	No se está enviando el email correspondiente	Alta	Sprint 1	Sprint 1
REF8 - Registro de usuarios	Incidente	Interno	Frontend	Interacción dudosa en el registro de usuario. No es intuitivo.	Si no aceptamos el checkbox y le damos click en registrar no hace nada. Quizás se podría mostrar como campo requerido o no habilitar el botón hasta que se acepte.	Alta	Sprint 1	Sprint 1
REF26 - Registrar una empresa	Mejora	Interno	Backend	Estados de entidad Company	Se desea solo tener dos estados, ACTIVE o DELETED con el fin de simplificar la solución	-	Sprint 3	Sprint 5
REF3 - Recuperación de contraseña	Incidente	Interno	Backend	Email no registrado quiere recuperar contraseña	Cuando pongo un correo inexistente o no registrado y selecciono confirmar, muestra el mensaje "Solicitud de reinicio de contraseña enviado exitosamente", no debería.	Baja	Sprint 2	Sprint 3

REF4 - Autogestión de perfil	Incidente	Interno	Frontend	Seleccionar dos veces seguidas editar usuario no funciona	Si selecciono editar usuario en gestionar usuarios, cierro la ventana, y vuelvo a querer abrir la misma, no me deja. Tengo que hacer otra acción para poder volver a editarlo	Baja	Sprint 2	Sprint 4
REF28 - Modificar la información de una empresa	Mejora	Interno	Frontend	Mejorar usabilidad en editar empresa	El lápiz se entiende que es el editar, sin embargo cuando le doy click me lo muestra readonly. Tengo que darle de nuevo en editar, lo cual queda raro. Si le doy click en el lápiz, espero que ya se abra en modo edición. Si quiero ver los datos, pondría el icono de un ojo. Ahi si levantaría el popup en modo readonly. Quizás podemos hacer que al hacer click en el lápiz ya se abra el popup en modo edición y si se hace click en la fila, se abra el popup en modo visualización (readonly)	-	Sprint 5	Sprint 7
Aplica a todos los requerimientos por igual	Mejora	Interno	Frontend	Loading en botones de carga	Cuando le doy click en guardar cambios, parece como que queda congelado sin hacer nada, aunque en realidad está haciendo. Ahí sería bueno bloquear el botón y agregar un loading, como mostrando que se está trabajando. Esto lo haría en todos los botones.	-	Sprint 6	Sprint 8
REF19 - Registrar solicitud de espacio de almacenamiento	Incidente	Interno	Frontend	No se muestra mensaje de éxito	Mostrar mensaje de que la empresa se contactará con la persona	Media	Sprint 3	Sprint 4

REF25 - Crear menú principal	Mejora	Interno	Frontend	Estética en la página principal	<p>Dejar más "aire" al bloque de casillero de búsqueda. Que quede más espacio arriba y abajo . aprox 3cm (Que las dos primeras secciones tengan la misma altura). Eliminar el botón "empezar ahora"</p> <p>Sección 4: Entre "propuesta de valor" y "nuestra propuesta de valor se apoya en 5 pilares", dejar menos interlineado.</p> <p>Borrar fondo de líneas onduladas, poner fondo azul oscuro (igual que la barra de menú) y los iconos y párrafos en blanco.</p> <p>Dejar 1 cm más de aire en la altura de la sección 4.</p> <p>Sección 5 (necesito espacio, tengo espacio): Eliminar.</p> <p>Sección 6: centrar título y logos dentro de la sección. Mantener altura de la sección.</p> <p>Footer (Sección 7): Centrar el logo de WareUP en cuanto a la altura. Social Media y el logo de LinkedIn subirlo un poquito, alinearlos a los otros títulos.+</p>	-	Sprint 8	Sprint 8
REF1 - Inicio de sesión de usuario	Mejora	Interno	Frontend	Wording de botón Iniciar sesión	ACEDER en vez de INICIAR SESIÓN en la barra de menú	-	Sprint 7	Sprint 7
REF19 - Registrar solicitud de espacio de almacenamiento	Incidente	Interno	Frontend	No se controla el campo numérico correctamente	Me deja cargar un texto, deberíamos controlar que sea un teléfono que acepte las características numéricas solo	Media	Sprint 3	Sprint 3

REF3 - Recuperación de contraseña	Mejora	Interno	Backend	Unificar los errores que devolvemos desde el backend	Para el login devolvemos un *msg* mientras que para el *recover-password* devolvemos un message y otra variable más *hasError.* Deberíamos unificarlos para siempre recibir lo mismo en el frontend	-	Sprint 1	Sprint 2
REF3 - Recuperación de contraseña	Incidente	Interno	Backend	Usuario no autorizado al querer cambiar contraseña	Url: 'https://wareup-api-dev.onrender.com/api/v1/users/update-password' Body: '{"password":"XXXXXX","linkEncrypt":"74eb6f90215e664ba7fb397e8d685b9c2ab1aa4c7aeddee214f6d1a980441d83"}' Esta fue la url desde donde lo hice: http://localhost:3000/users/password - recovery?74eb6f90215e664ba7fb397e8d685b9c2ab1aa4c7aeddee214f6d1a980441d83 Me está dando 402 y no puedo cambiar la contraseña	Alta	Sprint 2	Sprint 2
REF35 - Listado de solicitudes de registro de espacios de almacenamiento	Incidente	Interno	Backend	GET de requestDeposit byUser devuelve objeto y no lista de objetos	<a href="http://localhost:3001/api/v1/depositRequests/byUser/3">http://localhost:3001/api/v1/depositRequests/byUser/3</a> Me devuelve un único objeto a diferencia de <a href="http://localhost:3001/api/v1/deposits/byUser/4">http://localhost:3001/api/v1/deposits/byUser/4</a> que me devuelve una lista. Devolver ambos de igual forma	Media	Sprint 3	Sprint 3
REF21 - Alta de espacio de almacenamiento	Incidente	Interno	Frontend	No se visualiza correctamente la ciudad al volver hacia atrás	Si estoy en el agregando servicios pero quiero volver a la info de datos	Media	Sprint 4	Sprint 5

					básicos de mi depósito, la ciudad no se carga correctamente			
REF23 - Modificar información del espacio de almacenamiento	Incidente	Interno	Backend	[Editar depósito] no se actualizan address y postalCode	Puedo modificar todos los campos menos esos, es como que ignore las modificaciones que le mando.  Body: { "description": "una descripción", "totalM3": 1234, "minimumBusinessPeriod": 3, "minimumBusinessVolume": 100, "address": "Canelones 1228 apto 301", "expectedPrice": 123456, "cityId": 4, "companyId": 2, "postalCode": 113000, "currency": "USD", "status": 1 }	Alta	Sprint 5	Sprint 5
REF1 - Inicio de sesión de usuario	Incidente	Externo	Ambiente de prueba	Iniciar sesión en el sistema	No me pude loguear ni como admin ni como usuario	Crítica	Sprint 6	Sprint 6
REF21 - Alta de espacio de almacenamiento	Mejora	Interno	Backend	[Formación de título para un depósito] Cambiar M3 por m³	Con este cambio, van a quedar más estéticos los títulos	-	Sprint 9	Sprint 10
REF21 - Alta de espacio de almacenamiento	Incidente	Interno	Backend	Inconsistencia entre data_type de total M3 en entidad deposit	En el controller validamos si es string pero en el modelo está definido como integer	Baja	Sprint 4	Sprint 6
REF1 - Inicio de sesión de usuario	Incidente	Externo	Frontend	Cambios de layouts entre usuario logueado y no logueado	Si estás logueado no accede al home. Al intentar ver el home con un usuario logueado demora en cargar y termina llevando a una página de error 404.	Alta	Sprint 6	Sprint 6
REF21 - Alta de espacio de almacenamiento	Mejora	Externo	Frontend	Configuración de los campos en el formulario	En el formulario Agregar información del depósito, el campo moneda	-	Sprint 6	Sprint 7

				Agregar información del depósito	quedó chico y los demás campos de la línea no rellena el gap.			
REF31 - Contactar empresa	Incidente	Interno	Frontend	Formulario Contactanos, interacción dudosa y no intuitiva	Luego de llenar el formulario de contáctenos y hacer click en enviar, se envía y muestra un mensaje de que se envió correctamente, pero queda en la misma pantalla. Hice click en el botón cancelar para volver para atrás pero no hace nada. Quizás luego de enviar se pueda redireccionar al home de nuevo, o que el cancelar te lleve al home.	Media	Sprint 4	Sprint 5
REF20 - Cambiar estado de solicitud de registro de espacio de almacenamiento	Incidente	Externo	Frontend	Aceptar solicitud de registro	Intentando aceptar la solicitud de registro de una empresa me dio error pongo captura a la derecha	Alta	Sprint 6	Sprint 6
REF34 - Listado de espacios de almacenamiento registrados	Mejora	Externo	Frontend	Agregar más información a las tablas de entidades	En <a href="https://wareup-front.onrender.com/admin/manage-requests">https://wareup-front.onrender.com/admin/manage-requests</a> podría estar bueno agregar una columna fecha de solicitud	-	Sprint 6	Sprint 7
REF31 - Contactar empresa	Incidente	Interno	Frontend	No se controla el campo numérico correctamente	Me deja cargar un texto, deberíamos controlar que sea un teléfono que acepte las características numéricas solo	Media	Sprint 4	Sprint 5
REF8 - Registro de usuarios	Mejora	Externo	Backend	Agregar nuevo campo de "Industria" al registro de usuario básico	Es un campo que nos puede servir para tener estadísticas del tipo de usuario que accede al sistema	-	Sprint 9	Sprint 9

REF15 - Visualización completa de una publicación de espacio	Incidente	Interno	Backend	Consulta: traer mas info en el GET de deposit	<p>"Hoy traemos:</p> <pre> "city": {   "id": 1,   "title": "Ciudad de la costa",   "visible": true,   "departmentId": 1,   "createdAt": "2023-07-18T17:33:35.000Z",   "updatedAt": "2023-07-18T17:33:35.000Z" }, {noformat} </pre> <p>Traer la información entera de la entidad department para poder mostrarla en la publicación de un depósito.</p>	Media	Sprint 5	Sprint 5
REF23 - Modificar información del espacio de almacenamiento	Incidente	Interno	Backend	Devolver información del departamento en deposits	Cómo devolvemos la info de la city, podríamos devolver la del department también? Esto nos ayuda para la edición del depósito	Media	Sprint 5	Sprint 5
REF23 - Modificar información del espacio de almacenamiento	Incidente	Interno	Backend	No se están actualizando los servicios de un depósito	Cuando quiero eliminar servicios de un depósito, me devuelve OK pero luego vuelvo a revisarlos, y veo que siguen existiendo	Media	Sprint 5	Sprint 6
REF1 - Inicio de sesión de usuario	Mejora	Externo	Frontend	Deslogueo por inactividad	No sé si está previsto, pero luego de 5 minutos inactivo se debería desloguear el usuario, les parece bien?	-	Sprint 8	No se implementó

REF17 - Generar solicitud de arrendamiento de un espacio de almacenamiento	Mejora	Externo	Frontend	Al intentar reservar sin estar logueado, me da un mensaje de error.	Estaría bueno que te abriera el cuadro de diálogo de logueo.	-	Sprint 9	Sprint 10
REF17 - Generar solicitud de arrendamiento de un espacio de almacenamiento	Mejora	Externo	Frontend	Cuando se realiza una reserva el cuadro de diálogo dice "Reserva exitosa" y la reserva aún no se concretó.	De todos modos entiendo que es un tema de definición de texto, que se cambia fácilmente. Me suena que habíamos hablado algo como que "El equipo de WareUP en breve se pondrán en contacto contigo" o algo del estilo	-	Sprint 9	Sprint 10
Es una mejora general, no aplica a ningún requerimiento en particular	Incidente	Externo	Frontend	Visibilidad de la web en mobile	Cambiar los atributos xs de las columnas por md para mejorar la vista mobile	Media	Sprint 6	Sprint 7
REF21 - Alta de espacio de almacenamiento	Mejora	Externo	Frontend	Luego de cargadas las fotos y disponibilidad, te avisa que fue exitosa, pero te deja en esa página	debería llevarte nuevamente a registrar o poner un botón "finalizar" y que te lleve al home no? o algo así...	-	Sprint 9	Sprint 9
REF27 - Baja de una empresa	Incidente	Interno	Frontend	Me deja eliminar una empresa que ya fue eliminada	No tenemos control de si una empresa fue eliminada o no. Si queremos eliminar una empresa que ya tiene estado eliminada, deberíamos mostrar un mensaje de error, así es más intuitivo para el usuario.	Media	Sprint 7	Sprint 8
REF8 - Registro de usuarios	Mejora	Interno	Frontend	Acción no intuitiva al finalizar el registro de usuario	Cuando finaliza, te muestra el mensaje de OK pero no hace más nada, estaría bueno que te mostrará el login	-	Sprint 4	Sprint 5

REF27 - Baja de una empresa	Incidente	Externo	Frontend	Eliminar empresa	Como usuario cliente, no me permitió eliminar mi empresa	Alta	Sprint 6	Sprint 6
REF21 - Alta de espacio de almacenamiento	Incidente	Externo	Backend	Faltan departamentos para elegir	Queriendo cargar un depósito real ubicado en San José, el menú desplegable de departamento no me trae San José, imagino que porque hay que cargarlo.	Alta	Sprint 9	Sprint 9
REF21 - Alta de espacio de almacenamiento	Incidente	Externo	Ambiente de prueba	Terminar carga del depósito	No me permitió terminar la carga del depósito y me apareció un cartel que dice "Próximamente", antes no me permitió subir fotos, etc	Alta	Sprint 9	Sprint 9
REF28 - Modificar la información de una empresa	Incidente	Interno	Frontend	Me deja modificar una empresa que ya fue eliminada	No tenemos control de si una empresa fue eliminada o no. Si queremos modificar una empresa que ya tiene estado eliminada, deberíamos mostrar un mensaje de error, así es más intuitivo para el usuario.	Media	Sprint 7	Sprint 8
REF15 - Visualización completa de una publicación de espacio	Incidente	Externo	Frontend	Mostrar más información de los depósitos	Mostrar los campos de minimumBusinessDate y minimumBusinessMts manejar el periodo en "días"	Media	Sprint 8	Sprint 9
REF29 - Cargar disponibilidad de un espacio de almacenamiento	Incidente	Interno	Backend	No se están controlando los rangos de tiempos registrados previamente	Al registrar un rango de días de disponibilidad del depósito no estamos controlando si ya existe otro para las fechas que estoy ingresando. Esto nos genera registros duplicados	Media	Sprint 8	Sprint 9

REF29 - Cargar disponibilidad de un espacio de almacenamiento	Incidente	Interno	Frontend	No estamos controlando el total de metros cúbicos para cargar la disponibilidad del depósito	Si ingreso un valor mayor al total de M3 que tiene el deposito, me deja continuar igual.	Media	Sprint 8	Sprint 9
REF35 - Listado de solicitudes de registro de espacios de almacenamiento	Incidente	Interno	Frontend	Falla al ordenar la tabla por una columna numérica	Puedo ordenar bien por las columnas que son texto pero no ordena 100% bien por una numérica	Media	Sprint 8	Sprint 9
REF37 - Listado de empresas de un usuario	Incidente	Interno	Frontend	Falla al ordenar la tabla por una columna numérica	Puedo ordenar bien por las columnas que son texto pero no ordena 100% bien por una numérica	Media	Sprint 8	Sprint 9
REF30 - Cargar imágenes en un espacio de almacenamiento	Incidente	Externo	Ambiente de prueba	Subir imágenes a un depósito	No me permite subir foto, me dice "failed to fetch"	Alta	Sprint 9	Sprint 9

Tabla 73: Registro de incidentes y mejoras

## Anexo 7 - Reuniones con tutor

Fecha	Motivo	Participantes	Notas
21/04/23	Reunión inicial	Marcela Abi-saab Romina Areco Leticia Presa Gerardo Maturro	Orientación y consideraciones sobre el proyecto. Identificación de temas relevantes y definición de prioridades a la hora de ejecutar el proyecto.
04/05/23	Presentación avances	Marcela Abi-saab Romina Areco Leticia Presa Gerardo Maturro	Se comentan los avances y se validan actividades en el desarrollo del proyecto. Se revisan las fechas de entrega y se corroboran los próximos pasos.
24/05/23	Definición de tecnologías	Marcela Abi-saab Romina Areco Leticia Presa Gerardo Maturro	Se plantean dudas sobre la propuesta del cliente de la tecnología y cómo afecta al proyecto.
13/07/23	Primera revisión	Marcela Abi-saab Romina Areco Leticia Presa Gerardo Maturro Rafael Bentancur	Se presenta al tutor y al tutor externo una presentación con los avances del proyecto hasta el momento y se recibe feedback valioso para continuar con el proyecto.
14/09/23	Presentación avances	Marcela Abi-saab Romina Areco Leticia Presa Gerardo Maturro	Reunión para mostrar avances del proyecto. Consultamos sobre secciones de la documentación que teníamos dudas sobre cómo presentar la data.
04/10/23	Presentación avances	Marcela Abi-saab Romina Areco Leticia Presa Gerardo Maturro	Se muestra la documentación avanzada, recta final del proyecto. Se conversa sobre los puntos de la documentación que hay que tener mayor consideración.

Tabla 74: Registro de reuniones con tutor

## Anexo 8 - Reuniones con cliente

Fecha	Motivo	Participantes Equipo	Participantes Cliente	Notas
08/03/2023	Presentarnos al cliente	Marcela Abisaab Romina Areco Leticia Presa	Andrés Fleitas	Nos presentamos al cliente con el propósito de que nos conozcan y sepan nuestro interés en su proyecto.
16/05/23 En el CIE	Presentación de requerimientos	Marcela Abisaab Romina Areco Leticia Presa	Andrés Fleitas Nicolas López Leticia Lamaita Gonzalo Reboledo	Se validan y se hacen correcciones sobre los requerimientos según comentarios del cliente. Se obtiene una priorización inicial de los mismos.
29/05/23	Tecnologías a utilizar	Marcela Abisaab Leticia Presa	Andrés Fleitas Nicolas López	Se plantean las posibles tecnologías, se acuerda realizar una prueba de concepto para validar tiempos y afectaciones sobre el alcance original.
01/07/23	Evaluar prototipos y evacuar dudas del equipo	Marcela Abisaab Romina Areco Leticia Presa	Andrés Fleitas Nicolas López	Se sacan notas sobre los prototipos realizados, se evacúan dudas puntuales, queda pendiente el cliente enviar banners y logos para utilizar.
31/07/23	Avances y evacuar dudas	Marcela Abisaab Romina Areco Leticia Presa	Andrés Fleitas Nicolas López	Se vuelven a analizar los prototipos y se recaba información a ser utilizada en la diagramación de la web.
16/09/23	Avances y recibir feedback	Marcela Abisaab Romina Areco Leticia Presa	Andrés Fleitas Nicolas López	Conversamos sobre la entrega de la página. Recibimos feedback para seguir avanzando.
18/10/23	Entrega final	Marcela Abisaab Romina Areco Leticia Presa	Andrés Fleitas Nicolas López	Se conversó sobre cuándo y cómo se entregará el código. Se hizo una revisión final y un cierre de proyecto en conjunto.

Tabla 75: Registro de reuniones con el cliente

## Anexo 9 - Sprint Retrospectives

Sprint	¿En qué nos fue bien?	¿En qué no nos fue bien?	¿Qué deberíamos intentar para el próximo sprint?
<b>0</b>	<p>Buena comunicación del equipo</p> <p>Disposición de todos los miembros</p> <p>Productividad y entusiasmo para avanzar con el proyecto</p>	<p>Nos pisamos con tareas iguales</p> <p>No hay carga horaria</p> <p>No hay especificación clara de las tareas</p>	<p>Mayor organización en la distribución de tareas</p> <p>Familiarizarnos con las nuevas tecnologías</p> <p>Puntualidad en las reuniones</p>
<b>1</b>	<p>Se logró definir la arquitectura</p> <p>Se definió esquema de base de datos</p> <p>Se mejoró en la autonomía a la hora de ejecutar tareas</p>	<p>Gestión de horas aún necesita mejora</p> <p>Reportar status al equipo</p>	<p>Aumentar la dedicación semanal</p> <p>Mejorar carga de horas</p> <p>Dejar registro de las reuniones que se tiene</p>
<b>2</b>	<p>Relación del equipo es muy positiva</p> <p>Se avanzó en el aprendizaje de las nuevas tecnologías</p>	<p>Atraso en el desarrollo</p> <p>Impuntualidad en la reuniones</p>	<p>Apegarnos al cronograma</p> <p>Ser efectivas con las reuniones</p>
<b>3</b>	<p>Se trabajo en equipo para avanzar con lo atrasado</p> <p>Estamos apegandonos a la forma de trabajo estipulada</p>	<p>No se están registrando los incidentes</p>	<p>No descuidar las calidad de la solución por avanzar</p> <p>Registro de incidentes</p> <p>Mejorar al velocity del equipo</p>
<b>4</b>	<p>Testing y solución de <i>bugs</i></p> <p>Buen manejo del tablero de Jira</p>	<p>No se cargaron bien las horas en este sprint</p>	<p>Recordar sacar captura y guardar toda documentación que nos sirva para la doc final</p> <p>Evacuarnos dudas lo antes posible, no demorar en contestarnos</p> <p>Mantener actualizado el board en Jira para ver el avance real sobre el sprint</p>

<b>5</b>	<p>Hubo compromiso con la carga de horas</p> <p>Cliente proporcionó feedback valioso</p>	<p>El equipo discutió sobre temas repetitivos</p>	<p>Poner foco en lo importante!</p> <p>Seguir avanzando a ritmo fuerte en el desarrollo</p>
<b>6</b>	<p>Se avanzó de buena manera en la documentación</p> <p>Comunicación con el cliente viene siendo muy positiva</p>	<p>Se pospusieron algunas meetings por temas individuales</p> <p>Cumplimiento con tareas de desarrollo</p>	<p>Mantener reuniones aunque no asistan todas.</p> <p>Poner foco en el continuar con el desarrollo de funcionalidades de alto valor para el negocio</p>
<b>7</b>	<p>Se cerraron temas que se venían arrastrando</p>		<p>Priorizar testing</p> <p>Comenzar a poner mayor foco en documentación</p>
<b>8</b>	<p>Se pudo hacer la integración con Google Maps</p> <p>Se finalizó el buscador</p>	<p>No pudimos resolver todos los <i>bugs</i> reportados por el cliente</p>	<p>Poner foco en solucionar <i>bugs</i></p> <p>Mantener ritmo de avance en la documentación</p>
<b>9</b>	<p>Pudimos resolver incidentes y atacar feedback del cliente</p> <p>Hubo un gran apoyo entre el equipo para avanzar en esta última etapa</p>	-	<p>Terminar la documentación y comenzar a realizar validaciones con Tutor</p> <p>Coordinar con cliente entrega de código</p>
<b>10</b>	<p>Trabajo en equipo para ver incidentes</p> <p>Demo final con el cliente</p> <p>Encuesta muy positiva del cliente para con el MVP</p>	-	-

Tabla 76: Registro de Sprint Retrospectives

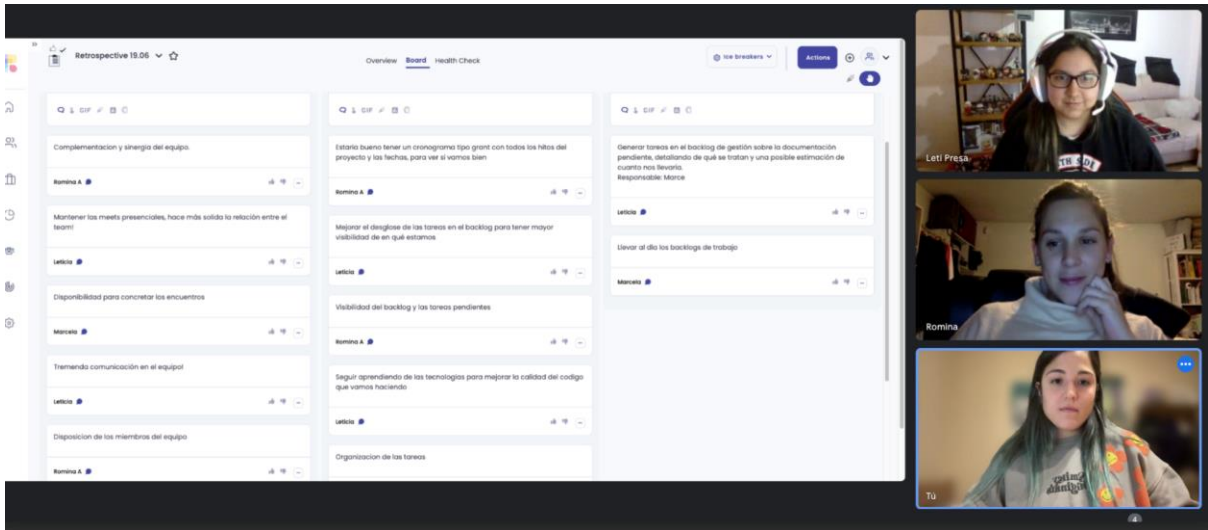


Ilustración 118: Retro S1 con herramienta Reetro.io

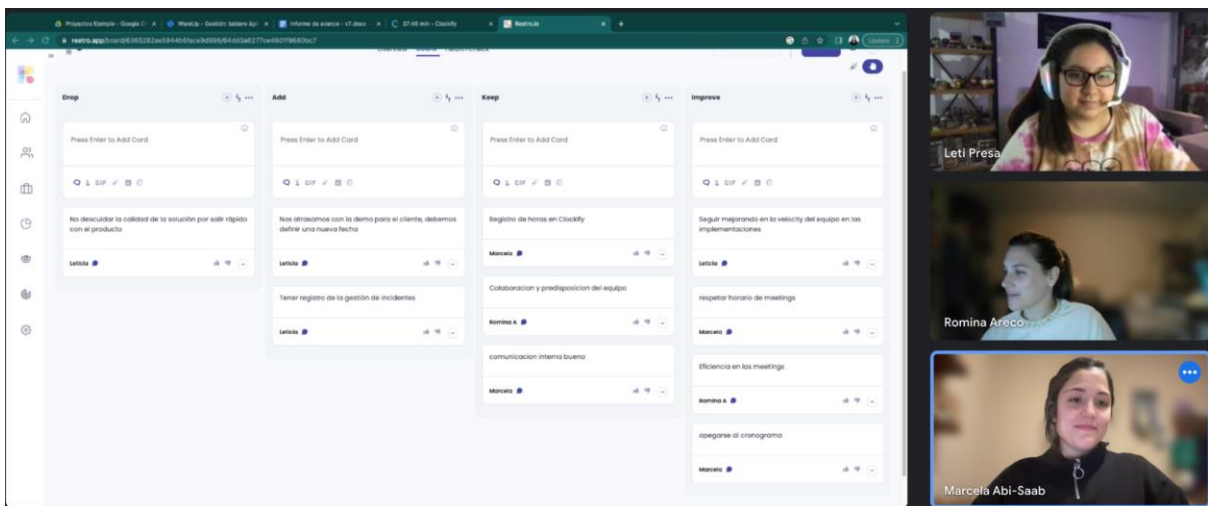


Ilustración 119: Retro S5 con herramienta Reetro.io

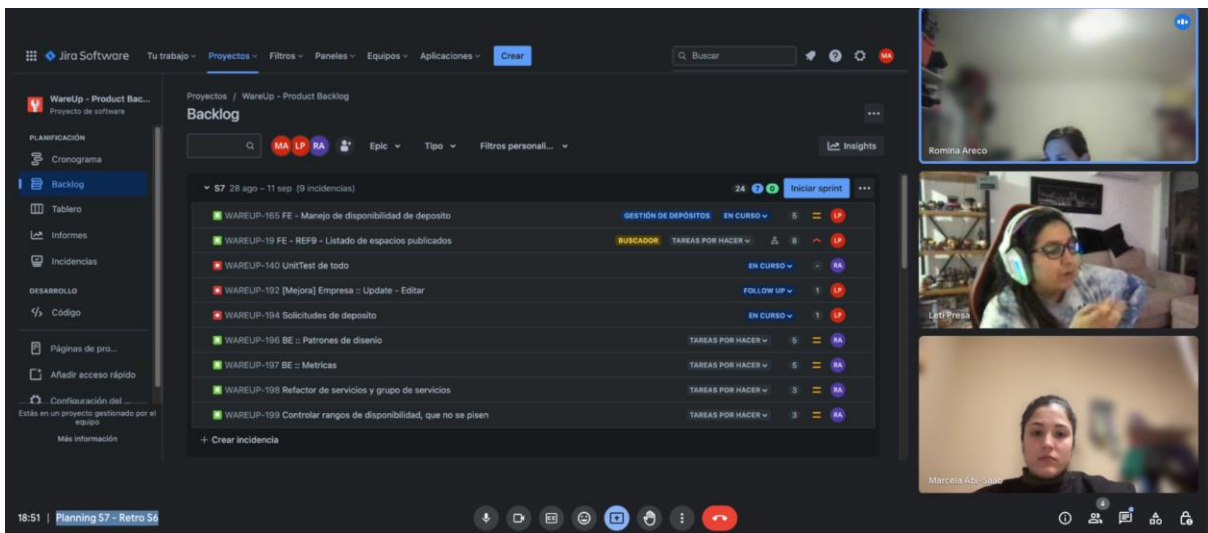


Ilustración 120: Planning S7 - Retro S6

## Anexo 10 - Planificación y resultado de los *sprints*

Sprint	Fecha	SP estimados	SP que se hicieron	US que se hicieron
1	5/6 - 18/6	-	45	REF1 - REF7 - REF2 - REF3 - REF8
2	19/6 - 02/7	45	40	REF4 - REF5 - REF25 - REF33 - REF38 - REF39
3	3/7 - 16/7	42	40	REF31 - REF26 - REF 19 - REF20 - REF35
4	17/7 - 30/7	42	42	REF21 - REF22 - REF27 - REF28 - REF37
5	31/7 - 13/8	42	42	REF15 - REF23 - REF34
<b>RELEASE 1</b>				
6	14/8 - 27/8	40	15	REF30
7	28/8 - 10/9	42	40	REF9 - REF10
8	11/9 - 24/9	40	43	REF12 - REF14 - REF29
9	25/9 - 08/10	40	40	REF17 - REF36 - REF40
<b>RELEASE 2</b>				
10	9/10 - 19/10	40	18	REF32
<b>RELEASE FINAL</b>				

Tabla 77: Registro de SP estimados y que se hicieron por sprint

### **Sprint 0**

**Duración:** 24/04/2023 al 04/06/2023

**Descripción del *sprint*:** Requerimientos, definiciones genéricas del proyecto, análisis del negocio

#### **Objetivos:**

- Especificación y priorización de requerimientos (funcionales y no funcionales).
- Definición de objetivos del proyecto.
- Definición de roles.
- Evaluación y definición de herramientas y tecnologías.

- Definición del ciclo de vida del proyecto.
- Definición de metodologías a utilizar.
- Gestión de riesgos.
- Análisis del entorno del negocio
- Armado de prototipos de la web.

### ***Sprint 1***

**Duración:** 05/06/2023 al 18/06/2023

**Descripción del *sprint*:** Arquitectura y Base de datos

#### **Objetivos:**

- Definir arquitectura y esquema de bases de datos del sistema.
- Implementación de la arquitectura y BD
- Definir bases y actividades de plan de calidad
- Desarrollo de tres requerimientos con Prioridad Alta
  - REF1 - Inicio de sesión de usuario
  - REF2 - Unicidad de identidad de usuario
  - REF3 - Recuperación de contraseña
  - REF7 - Logout
  - REF8 - Registro de usuarios

### ***Sprint 2***

**Duración:** 19/06/2023 al 02/07/2023

**Descripción del *sprint*:** Desarrollo de funcionalidades e investigación de UX

#### **Objetivos:**

- Desarrollo de funcionalidades
  - REF4 - Autogestión de perfil
  - REF19 - Registrar solicitud de espacio de almacenamiento
  - REF25 - Crear menú principal
  - REF5 - Roles de usuario
- Investigación de UX para el diseño de la UI
- Trabajo en *bugs* reportados internamente de las funcionalidades implementadas en sprint anterior.

### ***Sprint 3***

**Duración:** 03/07/2023 al 16/07/2023

**Descripción del *sprint*:** Desarrollo de funcionalidades y base de datos

**Objetivos:**

- Desarrollo de funcionalidades
  - REF19 - Registrar solicitud de espacio de almacenamiento
  - REF20 - Baja de solicitud de alta de almacenamiento
  - REF26 - Registrar una empresa
  - REF31 - Contactar empresa
  - REF35 - Listado de solicitudes de registro de espacios de almacenamiento
- Trabajo en *bugs* reportados internamente de las funcionalidades implementadas en sprint anterior
- Carga masiva de datos en la base de datos

### ***Sprint 4***

**Duración:** 17/07/2023 al 30/07/2023

**Descripción del *sprint*:** Desarrollo de funcionalidades

**Objetivos:**

- Desarrollo de funcionalidad
  - REF21 - Alta de espacio de almacenamiento
  - REF22 - Baja de espacio de almacenamiento
  - REF27 - Baja de una empresa
  - REF28 - Modificar información de una empresa
  - REF37 - Listado de empresas de un usuario
- Trabajo en *bugs* reportados internamente de las funcionalidades implementadas en sprint anterior

### ***Sprint 5***

**Duración:** 31/07/2023 al 13/08/2023

**Descripción del *sprint*:** Desarrollo de funcionalidades y mejora en el diseño de la UI

**Objetivos:**

- Desarrollo de funcionalidad
  - REF15 - Visualización completa de una publicación de espacio
  - REF23 - Modificar información del espacio de almacenamiento
  - REF34 - Listado de espacios de almacenamiento registrados
- Trabajo en *bugs* reportados internamente de las funcionalidades implementadas en sprint anterior
- Mejora en el diseño de la UI
- Investigación de servidor online gratis para realizar el primer release para el cliente
- Armado de Release I

### ***Sprint 6***

**Duración:** 14/07/2023 al 27/08/2023

**Descripción del *sprint*:** Desarrollo de funcionalidades y atacar feedback del cliente

**Objetivos:**

- Desarrollo de funcionalidades
  - REF30 - Cargar imagenes en un deposito
  - REF9 - Buscador de espacios publicados
  - REF10 - Filtros para mejorar la búsqueda de espacios
  - REF11 - Ordenar resultados de búsqueda
- Trabajar en el feedback del cliente de la release I

En esta etapa del proyecto el equipo tuvo un atraso en la completitud de requerimientos debido a temas médicos de una de las integrantes y obligaciones estudiantiles de las demás integrantes, esto generó que se pudiera finalizar únicamente el REF30.

### ***Sprint 7***

**Duración:** 28/08/2023 al 10/09/2023

**Descripción del *sprint*:** Desarrollo: Compensar el atraso del sprint anterior.

**Objetivos:**

- Lograr implementar las funcionalidades que quedaron atrasadas del Sprint 6
- Desarrollo de funcionalidades
  - REF 9 - Listado de espacios publicados

- REF10 - Filtros para mejorar la búsqueda de espacios
- Poner foco en testing.
- Trabajamos sobre una primera revisión del feedback recibido del cliente luego de entregar la primera versión
- En esta etapa del proyecto se despriorizó la implementación del REF11 ya que se tenían pendientes requerimientos de mayor prioridad. Esto fue de común acuerdo con el cliente.

## ***Sprint 8***

**Duración:** 11/09/2023 al 24/09/2023

**Descripción del *sprint*:** Desarrollo: *Backlog* / Documentación

### **Objetivos:**

- Desarrollo de funcionalidades
  - REF12 - Mapa con las publicaciones en base a la cercanía del usuario
  - REF14 - Visualización rápida de un espacio
  - REF29 - Cargar disponibilidad de un depósito
- Investigación de integración del *frontend* con Google Maps para REF12
- Trabajo en *bugs* reportados internamente de las funcionalidades implementadas en sprint anterior
- Comenzar proceso final de documentación
- Trabajamos sobre una segunda revisión del *feedback* proporcionado por el cliente
- En este sprint se superó la estimación de *story points* por 3, debido a que se agregó al sprint backlog la resolución de un *bug* que no se había considerado en la *sprint planning*.

## ***Sprint 9***

**Duración:** 25/09/2023 al 8/10/2023

**Descripción del *sprint*:** Documentación y testing

### **Objetivos:**

- Desarrollo de funcionalidades
  - REF17 - Generar solicitud de arrendamiento de un espacio de almacenamiento

- REF36 - Listado de solicitudes de arrendamiento
- REF40 - Listado de rangos de disponibilidad agregados para un espacio de almacenamiento
- Trabajo en *bugs* reportados internamente de las funcionalidades implementadas en sprint anterior
- Se investiga cómo integrar *Google Analytics* a la web
- Finalizar documentación
- Gran foco en refinar la UX de la web

### **Sprint 10**

**Duración:** 09/10/2023 al 19/10/2023

**Descripción del *sprint*:** Documentación y atacar feedback del cliente

#### **Objetivos:**

- Desarrollo de funcionalidades
  - REF32 - Métricas internas del sistema para usuario backoffice
  - REF11 - Ordenar resultados de búsqueda
  - REF13 - Significado de los filtros
- Se trabajó sobre el feedback del cliente de la Release II
- Finalizar documentación y realizar validaciones con tutor
- En este sprint se intentó finalizar los requerimientos que no fueron implementados pero no se logró ya que hubo que poner el mayor esfuerzo en la documentació

## Anexo 11 - Endpoints REST del backend

### **Users**

<b>Descripción</b>	Lista de usuarios con el status indicado o todos
<b>Endpoint</b>	/users
<b>Verbo</b>	GET
<b>Parámetros</b>	status (opcional)
<b>Body</b>	-

<b>Requerida</b>	{ message, hasError, resultCode, users: [ {id, name, lastname, password, email, role, status, createdAt, updatedAt} ] }
------------------	---

Tabla 78: Obtener todos los usuarios

<b>Descripción</b>	Obtener usuario logueado
<b>Endpoint</b>	/users/me
<b>Verbo</b>	GET
<b>Parámetros</b>	-
<b>Body</b>	-
<b>Requerida</b>	{ message, hasError, resultCode, user: {id, name, lastname, password, email, role, status, createdAt, updatedAt} }

Tabla 79: Obtener el usuario que haya iniciado sesión

<b>Descripción</b>	Crea un usuario nuevo en el sistema
<b>Endpoint</b>	/users
<b>Verbo</b>	POST
<b>Parámetros</b>	-
<b>Body</b>	{ name, lastname, password, email }
<b>Requerida</b>	{ message, hasError, resultCode, user: {id, name, lastname, password, email, role, status, createdAt, updatedAt} }

Tabla 80: Crear nuevo usuario

<b>Descripción</b>	Actualiza los datos de un usuario existente
<b>Endpoint</b>	/users/:userId
<b>Verbo</b>	PUT
<b>Parámetros</b>	userId
<b>Body</b>	{ name, lastname, password, email }
<b>Requerida</b>	{ message, hasError, resultCode, user: {id, name, lastname, password, email,

	role, status, createdAt, updatedAt} }
--	---------------------------------------

Tabla 81: Actualizar un usuario

<b>Descripción</b>	Elimina lógicamente un usuario existente
<b>Endpoint</b>	/users/:userId
<b>Verbo</b>	DELETE
<b>Parámetros</b>	userId
<b>Body</b>	-
<b>Requesta</b>	{ message, hasError, resultCode, user: {id, name, lastname, password, email, role, status, createdAt, updatedAt} }

Tabla 82: Eliminar lógicamente un usuario

<b>Descripción</b>	Recupera la password de un usuario existente
<b>Endpoint</b>	/users/recover-password
<b>Verbo</b>	POST
<b>Parámetros</b>	-
<b>Body</b>	{ email }
<b>Requesta</b>	{ message, hasError, resultCode }

Tabla 83: Recupera la password de un usuario existente

<b>Descripción</b>	Envía un formulario de contacto a WareUP con datos del usuario que lo envió
<b>Endpoint</b>	/users/contact
<b>Verbo</b>	POST
<b>Parámetros</b>	-
<b>Body</b>	{ email, phone, clientName, message, subject }
<b>Requesta</b>	{ message, hasError, resultCode }

Tabla 84: Envía un formulario de contacto con datos del usuario que lo envió

<b>Descripción</b>	Obtener los datos de un usuario
<b>Endpoint</b>	/users/:id
<b>Verbo</b>	GET
<b>Parámetros</b>	id
<b>Body</b>	-
<b>Requerida</b>	{ message, hasError, resultCode, user: {id, name, lastname, password, email, role, status, createdAt, updatedAt} }

Tabla 85: Obtener los datos de un usuario

## Companies

<b>Descripción</b>	Crea una nueva empresa
<b>Endpoint</b>	/companies
<b>Verbo</b>	POST
<b>Parámetros</b>	
<b>Body</b>	{RUT, businessName, email, phone, contactName, cityId, address, postalCode, position}
<b>Requerida</b>	{ message, hasError, resultCode, company: {RUT, businessName, email, phone, contactName, cityId, address, postalCode, position, status} }

Tabla 86: Crear nueva empresa

<b>Descripción</b>	Modificar una empresa existente
<b>Endpoint</b>	/companies/:id
<b>Verbo</b>	PUT
<b>Parámetros</b>	id
<b>Body</b>	{RUT, businessName, email, phone, contactName, cityId, address, postalCode, position}

<b>Requesta</b>	{ message, hasError, resultCode, company: {RUT, businessName, email, phone, contactName, cityId, address, postalCode, position, status } }
-----------------	--

Tabla 87: Modificar una empresa existente

<b>Descripción</b>	Obtener listado de empresas según su estado
<b>Endpoint</b>	/companies?status
<b>Verbo</b>	GET
<b>Parámetros</b>	status (opcional)
<b>Body</b>	-
<b>Requesta</b>	{ message, hasError, resultCode, companies: [{RUT, businessName, email, phone, contactName, cityId, address, postalCode, position, status } ] }

Tabla 88: Obtener listado de empresas según su estado

<b>Descripción</b>	Obtener listado de empresas según usuario
<b>Endpoint</b>	/companies/byUser/:userId
<b>Verbo</b>	GET
<b>Parámetros</b>	userId
<b>Body</b>	-
<b>Requesta</b>	{ message, hasError, resultCode, companies: [{RUT, businessName, email, phone, contactName, cityId, address, postalCode, position, status} ] }

Tabla 89: Obtener listado de empresas según usuario

<b>Descripción</b>	Elimina lógicamente una empresa existente
<b>Endpoint</b>	/companies/:id
<b>Verbo</b>	DELETE
<b>Parámetros</b>	id
<b>Body</b>	-

<b>Requerida</b>	{ message, hasError, resultCode, company: {RUT, businessName, email, phone, contactName, cityId, address, postalCode, position, status } }
------------------	--

Tabla 90: Elimina lógicamente una empresa existente

<b>Descripción</b>	Obtiene una empresa por id
<b>Endpoint</b>	/companies/:id
<b>Verbo</b>	GET
<b>Parámetros</b>	id
<b>Body</b>	-
<b>Requerida</b>	{ message, hasError, resultCode, company: {RUT, businessName, email, phone, contactName, cityId, address, postalCode, position, status } }

Tabla 91: Obtiene una empresa por id

### DepositRequests

<b>Descripción</b>	Crea una nueva solicitud de registro de depósito
<b>Endpoint</b>	/depositRequest
<b>Verbo</b>	POST
<b>Parámetros</b>	
<b>Body</b>	{companyId, address, phone, cityId, email, title, description}
<b>Requerida</b>	{ message, hasError, resultCode, depositRequest: {companyId, address, phone, cityId, email, title, description } }

Tabla 92: Crea una nueva solicitud de registro de depósito

<b>Descripción</b>	Modificar una solicitud de depósito existente
<b>Endpoint</b>	/depositRequest/:id
<b>Verbo</b>	PUT

<b>Parámetros</b>	id
<b>Body</b>	{status}
<b>Requesta</b>	{ message, hasError, resultCode, depositRequest: {companyId, address, phone, cityId, email, title, description } }

Tabla 93: Crea una nueva solicitud de registro de depósito

<b>Descripción</b>	Obtener listado de solicitudes de depósito según empresa
<b>Endpoint</b>	/depositRequest/byCompany/:id
<b>Verbo</b>	GET
<b>Parámetros</b>	id
<b>Body</b>	-
<b>Requesta</b>	{ message, hasError, resultCode, depositRequests: [{companyId, address, phone, cityId, email, title, description }]} }

Tabla 94: Obtener listado de solicitudes según empresa

<b>Descripción</b>	Obtener listado de solicitudes de depósito según usuario
<b>Endpoint</b>	/depositRequest/byUser/:userId
<b>Verbo</b>	GET
<b>Parámetros</b>	id
<b>Body</b>	-
<b>Requesta</b>	{ message, hasError, resultCode, depositRequests: [{companyId, address, phone, cityId, email, title, description }]} }

Tabla 95: Obtener listado de solicitudes según usuario

<b>Descripción</b>	Obtener todas las solicitudes de depósito
<b>Endpoint</b>	/depositRequest/
<b>Verbo</b>	GET

<b>Parámetros</b>	-
<b>Body</b>	-
<b>Requesta</b>	{ message, hasError, resultCode, depositRequests: [{companyId, address, phone, cityId, email, title, description}] }

Tabla 96: Obtener todas las solicitudes de depósito

<b>Descripción</b>	Obtiene una solicitud de depósito por id
<b>Endpoint</b>	/depositRequest/:id
<b>Verbo</b>	GET
<b>Parámetros</b>	id
<b>Body</b>	-
<b>Requesta</b>	{ message, hasError, resultCode, depositRequest: {companyId, address, phone, cityId, email, title, description} }

Tabla 97: Obtiene una solicitud de depósito por id

## Deposits

<b>Descripción</b>	Crear un nuevo depósito en el sistema
<b>Endpoint</b>	/deposits
<b>Verbo</b>	POST
<b>Parámetros</b>	
<b>Body</b>	{ description, totalM3, currency, minimumBusinessPeriod, minimumBusinessVolume, expectedPrice, cityId }
<b>Requesta</b>	{ message, hasError, resultCode, deposit: { id, description, totalM3, currency, minimumBusinessPeriod, minimumBusinessVolume, expectedPrice, cityId, status } }

Tabla 98: Crear un nuevo depósito en el sistema

<b>Descripción</b>	Editar un depósito existente
<b>Endpoint</b>	/deposits/:id
<b>Verbo</b>	PUT
<b>Parámetros</b>	id
<b>Body</b>	{ description, totalM3, currency, minimumBusinessPeriod, minimumBusinessVolume, expectedPrice, cityId }
<b>Requesta</b>	{ message, hasError, resultCode, deposit: { id, description, totalM3, currency, minimumBusinessPeriod, minimumBusinessVolume, expectedPrice, cityId, status } }

Tabla 99: Editar un depósito existente

<b>Descripción</b>	Eliminar un depósito existente
<b>Endpoint</b>	/deposits/:id
<b>Verbo</b>	DELETE
<b>Parámetros</b>	id
<b>Body</b>	-
<b>Requesta</b>	{ message, hasError, resultCode, deposit: { id, description, totalM3, currency, minimumBusinessPeriod, minimumBusinessVolume, expectedPrice, cityId, status } }

Tabla 100: Eliminar un depósito existente

<b>Descripción</b>	Obtener listado de depósitos
<b>Endpoint</b>	/deposits
<b>Verbo</b>	GET
<b>Parámetros</b>	applyFilter, servicesId, status, city
<b>Body</b>	-
<b>Requesta</b>	{ message, hasError, resultCode, deposits: [{ id, description, totalM3,

	currency, minimumBusinessPeriod, minimumBusinessVolume, expectedPrice, cityId, status ]} }
--	--

Tabla 101: Obtener listado de depósitos

<b>Descripción</b>	Obtener listado de depósitos de un usuario
<b>Endpoint</b>	/deposits/byUser/:id
<b>Verbo</b>	GET
<b>Parámetros</b>	id
<b>Body</b>	-
<b>Requesta</b>	{ message, hasError, resultCode, deposits: [{ id, description, totalM3, currency, minimumBusinessPeriod, minimumBusinessVolume, expectedPrice, cityId, status }]} }

Tabla 102: Obtener listado de depósitos de un usuario

<b>Descripción</b>	Obtener listado de depósitos de la empresa
<b>Endpoint</b>	/deposits/byCompany/:id
<b>Verbo</b>	GET
<b>Parámetros</b>	id
<b>Body</b>	-
<b>Requesta</b>	{ message, hasError, resultCode, deposits: [{ id, description, totalM3, currency, minimumBusinessPeriod, minimumBusinessVolume, expectedPrice, cityId, status }]} }

Tabla 103: Obtener listado de depósitos de la empresa

<b>Descripción</b>	Registra imágenes de un depósito
<b>Endpoint</b>	/deposits/images/:id
<b>Verbo</b>	POST
<b>Parámetros</b>	id

<b>Body</b>	{ images: [ imageBase64 ] }
<b>Requesta</b>	{ message, hasError, resultCode, deposit: { id, description, totalM3, currency, minimumBusinessPeriod, minimumBusinessVolume, expectedPrice, cityId, status } }

Tabla 104: Registra imágenes de un depósito

<b>Descripción</b>	Obtener las imágenes de un depósito
<b>Endpoint</b>	/deposits/images/:id
<b>Verbo</b>	GET
<b>Parámetros</b>	id
<b>Body</b>	-
<b>Requesta</b>	{ message, hasError, resultCode, depositImages: [ {image} ] }

Tabla 105: Obtener las imágenes de un depósito

<b>Descripción</b>	Registro del calendario de un depósito
<b>Endpoint</b>	/depositCalendar
<b>Verbo</b>	POST
<b>Parámetros</b>	-
<b>Body</b>	{ dateFrom, dateTo, totalM3, depositId }
<b>Requesta</b>	{ message, hasError, resultCode, depositCalendar: { id, dateFrom, dateTo, totalM3, depositId } }

Tabla 106: Registro del calendario de un depósito

<b>Descripción</b>	Obtener calendarios de un depósito
<b>Endpoint</b>	/depositCalendar/byDeposit/:id
<b>Verbo</b>	GET
<b>Parámetros</b>	id

<b>Body</b>	-
<b>Requesta</b>	{ message, hasError, resultCode, depositCalendars: [{ id, dateFrom, dateTo, totalM3, depositId }] }

Tabla 107: Obtener calendarios de un depósito

<b>Descripción</b>	Actualizar el calendario de un depósito
<b>Endpoint</b>	/depositCalendar/:id
<b>Verbo</b>	PUT
<b>Parámetros</b>	id
<b>Body</b>	{ dateFrom, dateTo, totalM3, depositId, isDeleted }
<b>Requesta</b>	{ message, hasError, resultCode, depositCalendar: { id, dateFrom, dateTo, totalM3, depositId } }

Tabla 108: Actualizar el calendario de un depósito

<b>Descripción</b>	Solicitud de reserva de depósito
<b>Endpoint</b>	/bookingRequests
<b>Verbo</b>	POST
<b>Parámetros</b>	-
<b>Body</b>	{ depositId, userId, description, totalM3, dateFrom, dateTo }
<b>Requesta</b>	{ message, hasError, resultCode, depositCalendar: { id, depositId, userId, description, totalM3, dateFrom, dateTo, status } }

Tabla 109: Solicitud de reserva de depósito

<b>Descripción</b>	Actualizar solicitud de reserva de depósito
<b>Endpoint</b>	/bookingRequests/:id
<b>Verbo</b>	PUT
<b>Parámetros</b>	id

<b>Body</b>	{ depositId, userId, description, totalM3, dateFrom, dateTo }
<b>Requesta</b>	{ message, hasError, resultCode, depositCalendar: { id, depositId, userId, description, totalM3, dateFrom, dateTo, status } }

Tabla 110: Actualizar solicitud de reserva de depósito

<b>Descripción</b>	Obtener las solicitudes de reservas de un depósito
<b>Endpoint</b>	/bookingRequests/byDeposit/:id
<b>Verbo</b>	GET
<b>Parámetros</b>	id
<b>Body</b>	-
<b>Requesta</b>	{ message, hasError, resultCode, depositCalendars: [{ id, depositId, userId, description, totalM3, dateFrom, dateTo, status }] }

Tabla 111: Obtener las solicitudes de reservas de un depósito

<b>Descripción</b>	Obtener las solicitudes de reservas de un usuario
<b>Endpoint</b>	/bookingRequests/byUser/:id
<b>Verbo</b>	GET
<b>Parámetros</b>	id
<b>Body</b>	-
<b>Requesta</b>	{ message, hasError, resultCode, depositCalendars: [{ id, depositId, userId, description, totalM3, dateFrom, dateTo, status }] }

Tabla 112: Obtener las solicitudes de reservas de un usuario

<b>Descripción</b>	Obtener las solicitudes de reservas
<b>Endpoint</b>	/bookingRequests
<b>Verbo</b>	GET

<b>Parámetros</b>	-
<b>Body</b>	-
<b>Requesta</b>	{ message, hasError, resultCode, depositCalendars: [{ id, depositId, userId, description, totalM3, dateFrom, dateTo, status } ] }

Tabla 113: Obtener las solicitudes de reservas

### Common

<b>Descripción</b>	Obtener los servicios y los grupos de servicios
<b>Endpoint</b>	/servicegroups
<b>Verbo</b>	GET
<b>Parámetros</b>	-
<b>Body</b>	-
<b>Requesta</b>	{ message, hasError, resultCode, serviceGroups: [{ id, code, title, order, showTitle, status, services: [ id, title, order, status] } ] }

Tabla 114: Obtener los servicios y los grupos de servicios

<b>Descripción</b>	Obtener un grupo de servicio y sus servicios
<b>Endpoint</b>	/servicegroups/:id
<b>Verbo</b>	GET
<b>Parámetros</b>	id
<b>Body</b>	-
<b>Requesta</b>	{ message, hasError, resultCode, serviceGroup: { id, code, title, order, showTitle, status, services: [ id, title, order, status] } }

Tabla 115: Obtener un grupo de servicio y sus servicios

<b>Descripción</b>	Obtener los departamentos
<b>Endpoint</b>	/common/departments
<b>Verbo</b>	GET

<b>Parámetros</b>	-
<b>Body</b>	-
<b>Requesta</b>	{ message, hasError, resultCode, departments:[ { id, title, visible, cities: [id, title, visible] } ] }

Tabla 116: Obtener los departamentos

<b>Descripción</b>	Obtiene las ciudades de un departamento
<b>Endpoint</b>	/common/cities/:id
<b>Verbo</b>	GET
<b>Parámetros</b>	id
<b>Body</b>	-
<b>Requesta</b>	{ message, hasError, resultCode, cities: [ { id, title, visible } ] }

Tabla 117: Obtiene las ciudades de un departamento

<b>Descripción</b>	Valida las credenciales de un usuario
<b>Endpoint</b>	/auth/login
<b>Verbo</b>	POST
<b>Parámetros</b>	
<b>Body</b>	{email, password}
<b>Requesta</b>	{ message, hasError, resultCode, user: {id, name, lastname, password, email, role, status, createdAt, updatedAt } }

Tabla 118: Valida las credenciales de un usuario

## Anexo 12 - UnitTest a nivel de backend

### Capas cubiertas

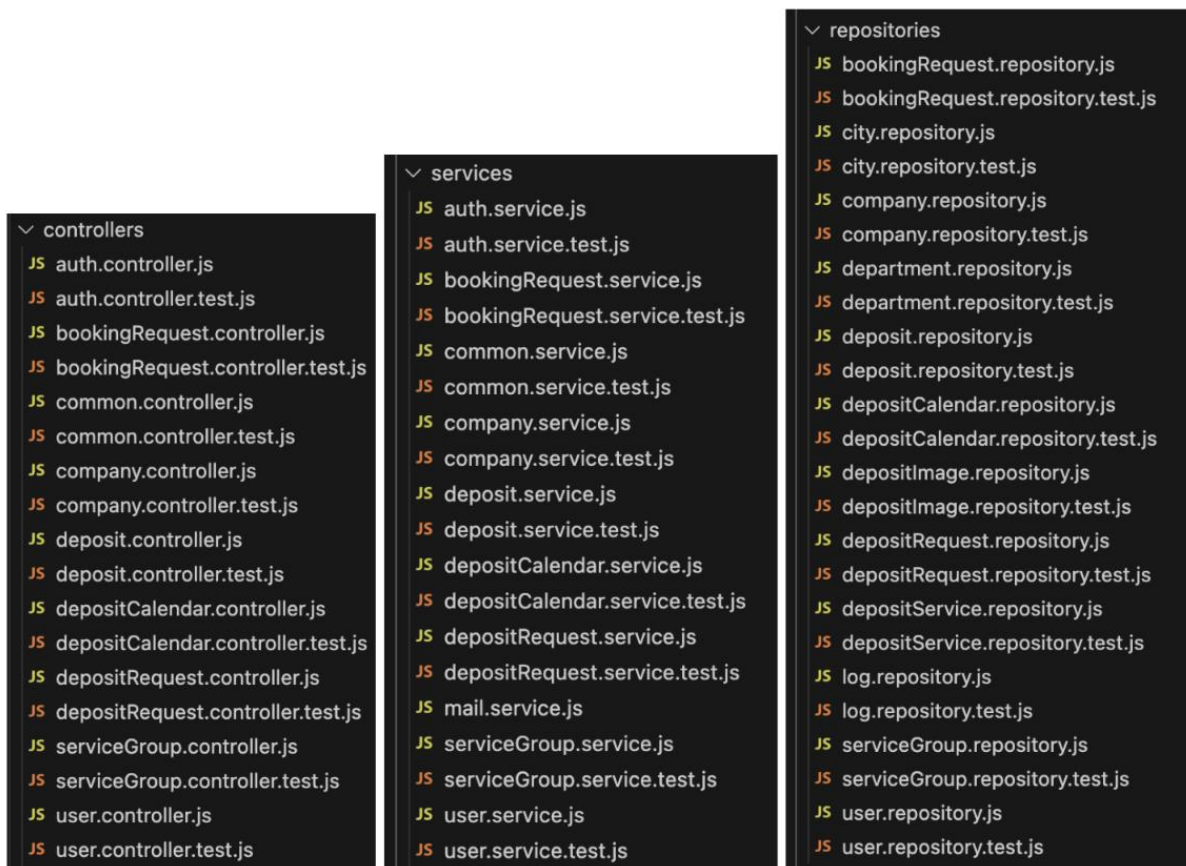


Ilustración 121: UnitTest - Capas de la solución

## Operaciones básicas

### Obtener Empresa

```
it("should return all company by user", async function() {
  const repository = new CompanyRepository();
  const stub = sinon.stub(repository, "getByUser").returns([stubValue]);

  const service = new CompanyService(repository);
  const result = await service.getByUser(stubValue.id);

  expect(stub.calledOnce).to.be.true;
  expect(result.companies[0].id).to.equal(stubValue.id);
});
```

Ilustración 122: UnitTest - Obtener empresa - OK

```
it("should return an error when getByUsers throws an exception", async function() {
  sinon.restore();
  const logRepository = new LogRepository();
  const repository = new CompanyRepository();
  var stub = sinon.stub(repository, "getByUser").throwsException();
  stub = sinon.stub(logRepository, "create").returns();

  const service = new CompanyService(repository, logRepository);
  const result = await service.getByUser(stubValue.id);

  expect(stub.calledOnce).to.be.true;
  expect(result.hasError).to.equal(true);
});
```

Ilustración 123: UnitTest - Obtener empresa - Error

## Obtener Empresas

```
it("should return all companies", async function() {
  const repository = new CompanyRepository();
  const stub = sinon.stub(repository, "getAll").returns([stubValue]);

  const service = new CompanyService(repository);
  const result = await service.getAll(stubValue.id);

  expect(stub.calledOnce).to.be.true;
  expect(result.companies[0].id).to.equal(stubValue.id);
});
```

Ilustración 124: UnitTest - Obtener empresas - OK

```
it("should return an error when getAalls throws an exception", async function() {
  sinon.restore();
  const logRepository = new LogRepository();
  const repository = new CompanyRepository();
  var stub = sinon.stub(repository, "getAll").throwsException();
  stub = sinon.stub(logRepository, "create").returns();

  const service = new CompanyService(repository, logRepository);
  const result = await service.getAll(stubValue.id);

  expect(stub.calledOnce).to.be.true;
  expect(result.hasError).to.equal(true);
});
```

Ilustración 125: UnitTest - Obtener empresas - Error

## Crear Empresa

```
it("should create a company when the data is okey", async function() {
  sinon.restore();
  const repository = new CompanyRepository();
  var stubCreate = sinon.stub(repository, "create").returns(stubValue);
  sinon.stub(userRepository, "get").returns({ id: 1, });
  sinon.stub(repository, "getByRUT").returns(null);

  const service = new CompanyService(repository, logRepository, userRepository);
  const result = await service.create(stubValue);

  expect(result.hasError).to.equal(false)
  expect(stubCreate.calledOnce).to.be.true;
  expect(result.company.id).to.equal(stubValue.id);
});
```

Ilustración 126: UnitTest - Crear empresa - OK

```
it("should return an error when create method throws an exception", async function() {
  sinon.restore();
  const logRepository = new LogRepository();
  const repository = new CompanyRepository();
  var stub = sinon.stub(repository, "create").throwsException();
  var stubGeUser = sinon.stub(userRepository, "get").returns({ id: 1, });
  stub = sinon.stub(logRepository, "create").returns();
  sinon.stub(repository, "getByRUT").returns(null);
  const service = new CompanyService(repository, logRepository, userRepository);
  const result = await service.create(stubValue.id);

  expect(stub.calledOnce).to.be.true;
  expect(result.hasError).to.equal(true);
});
```

## Ilustración 127: UnitTest - Crear empresa - Error

### Modificar Empresa

```
it("should update a company", async function() {
  sinon.restore();
  const repository = new CompanyRepository();
  sinon.stub(repository, "update").returns(stubValue);
  sinon.stub(repository, "get").returns(stubValue);

  const service = new CompanyService(repository, logRepository, userRepository);
  const result = await service.update(stubValue);

  expect(result.hasError).toEqual(false);
  expect(result.company.id).toEqual(stubValue.id);
});
```

## Ilustración 128: UnitTest - Modificar empresa - OK

```
it("should return an error", async function() {
  sinon.restore();
  const logRepository = new LogRepository();
  const repository = new CompanyRepository();
  var stub = sinon.stub(repository, "update").throwsException();
  var stubGeUser = sinon.stub(userRepository, "get").returns({ id: 1, });
  stub = sinon.stub(logRepository, "create").returns();
  const service = new CompanyService(repository, logRepository, userRepository);
  const result = await service.update(stubValue.id);

  expect(stub.calledOnce).toBe(true);
  expect(result.hasError).toEqual(true);
});
```

## Ilustración 129: UnitTest - Modificar empresa - Error

### Eliminar Empresa

```
it("should delete a company", async function() {
  sinon.restore();
  const repository = new CompanyRepository();
  sinon.stub(repository, "update").returns(stubValue);
  sinon.stub(repository, "get").returns(stubValue);

  const service = new CompanyService(repository, logRepository, userRepository);
  const result = await service.delete(stubValue);

  expect(result.hasError).toEqual(false);
  expect(result.company.id).toEqual(stubValue.id);
});
```

## Ilustración 130: UnitTest - Eliminar empresa - OK

```

it("should return an error", async function() {
  sinon.restore();
  const logRepository = new LogRepository();
  const repository = new CompanyRepository();
  var stub = sinon.stub(repository, "update").throwsException();
  sinon.stub(repository, "get").returns({ id: 1 });
  stub = sinon.stub(logRepository, "create").returns();
  const service = new CompanyService(repository, logRepository, userRepository);
  const result = await service.delete(stubValue.id);

  expect(stub.calledOnce).to.be.true;
  expect(result.hasError).to.equal(true);
});

```

Ilustración 131: UnitTest - Eliminar empresa - Error

## Anexo 13 - Sendgrid en el backend

Para el envío de mails utilizamos sendgrid. *Sendgrid* permite el envío de *emails* y además la creación de templates, entre otras funcionalidades. Es una plataforma muy potente. Se pueden crear campañas de marketing también, entre otros.

### Sobre la configuración para WareUP

En la parte de *Email API* está la configuración para integrarlo con nuestra API con NodeJS.

Ahí nos importan dos cosas:

Primero los **legacy transactional** templates. Acá podemos diseñar *templates* de emails para luego usarlos desde la api. Lo interesante es que se administran desde la plataforma misma y no necesitamos descargarlos ni llevar el HTML a nuestra API o llevar los templates. Simplemente podemos usarlos a través del ID que *Sendgrid* le asocia.

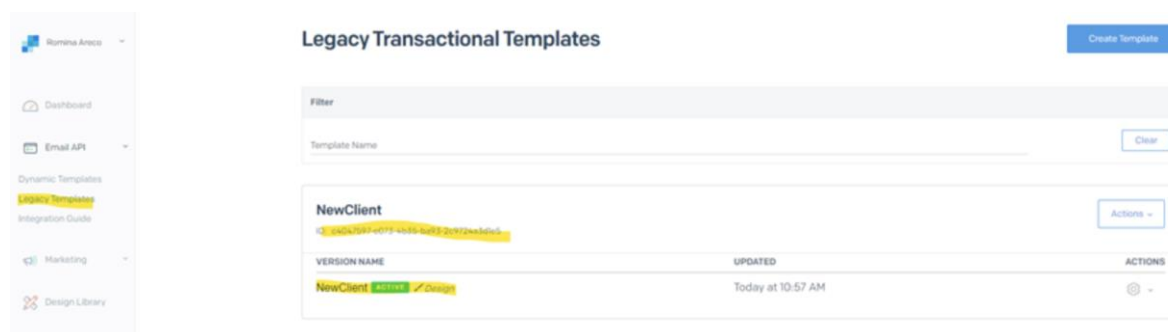


Ilustración 132: Configuración Sendgrid - paso 1

Ese id se coloca directamente en nuestra API para referenciar a este template.

```
const msg = {
  to: email,
  from: 'romiareco@gmail.com', // Change to your verified sender
  subject: 'Welcome to WareUp',
  template_id: 'c4047597-e073-4b35-ba93-2c9724a3d1e5'
}
sgMail
  .send(msg)
  .then((response) => {
    console.log('Email sent')
  })
  .catch((error) => {
    console.error(error)
  })
})
```

Ilustración 133: Configuración Sendgrid - paso 2

Lo otro relevante es el *API Key*, que es la forma en que nuestra cuenta de sendgrid se integra con nuestra API. Ahí se generan en Sendgrid y se copian a nuestro código.

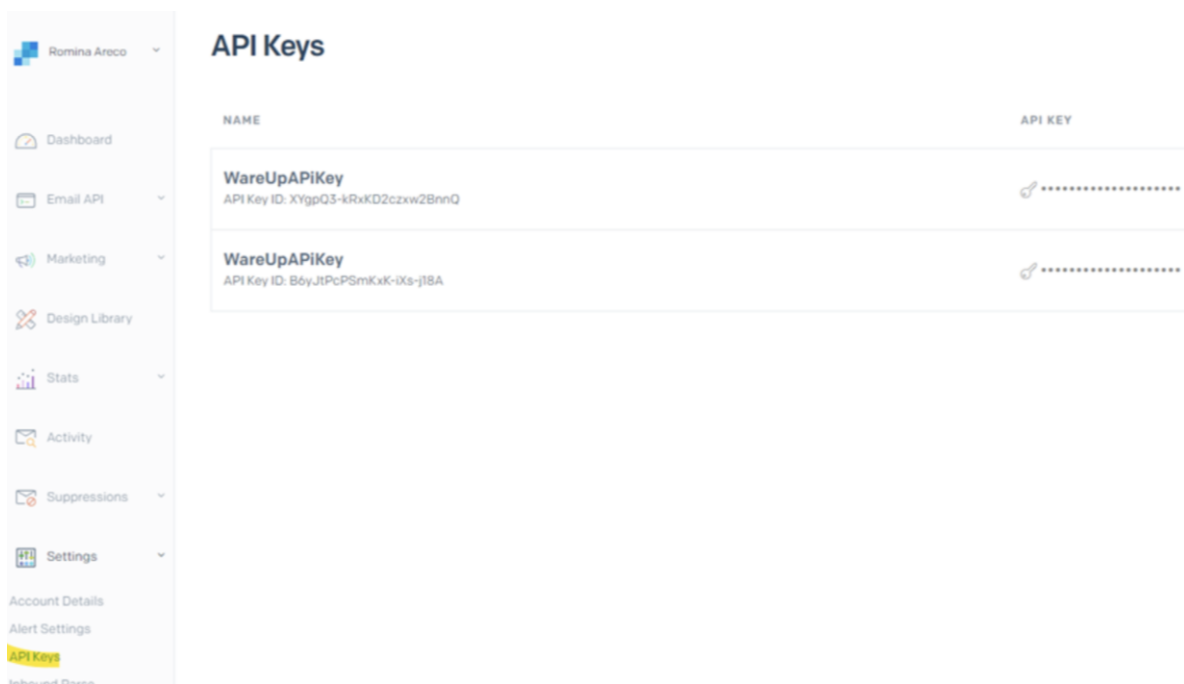


Ilustración 134: Configuración Sendgrid - paso 3

```
const sgMail = require('@sendgrid/mail')
sgMail.setApiKey('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')//Take from config

const msg = {
  to: email,
  from: 'romiareco@gmail.com', // Change to your verified sender
  subject: 'Welcome to WareUp',
  template_id: 'c4047597-e073-4b35-ba93-2c9724a3d1e5'
}
sgMail
```

Ilustración 135: Configuración Sendgrid - paso 4

Con esta configuración, ya queda integrado a nuestra API y queda funcionando el envío de *emails*, utilizando los *templates* indicados.

# Anexo 14 - Reportes Clockify - Gestión de horas

## Summary report

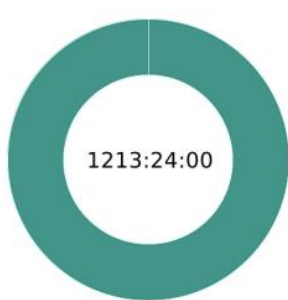
01/01/2023 - 31/12/2023



Total: 1213:24:00 Billable: 1213:24:00 Amount: 0.00 USD

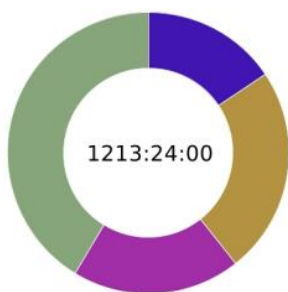


## Project



• Ware Up	1213:24:00	100.00%
-----------	------------	---------

## Task



• Desarrollo y Testing - Ware Up	501:30:00	41.33%
• Documentación - Ware Up	234:39:00	19.34%
• Investigación y Capacitación - Ware Up	287:53:00	23.73%
• Reuniones y Gestión - Ware Up	189:22:00	15.61%

Ilustración 136: Resumen Clockify

## Reporte de horas detallado del mes de Agosto

### Detailed report

01/08/2023 - 31/08/2023

Total: 113:11:00 Billable: 113:11:00 Amount: 0.00 USD



Date	Description	Duration	User
30/08/2023	Gestión de riesgos WareUp - Documentación	01:56:10 17:24:02 - 19:20:12	Marcela Abi-Saab 0.00 USD
30/08/2023	Documentando riesgos WareUp - Documentación	02:11:00 11:46:00 - 13:57:00	Marcela Abi-Saab 0.00 USD
29/08/2023	Desarrollo WareUp - Desarrollo y Testing	08:00:00 20:27:00 - 04:27:00 <sup>+1</sup>	Romina Areco 0.00 USD
28/08/2023	Se continua trabajando en la vista de publicación + miscellaneous WareUp - Desarrollo y Testing	03:03:37 17:48:16 - 20:51:53	Leti Presa 0.00 USD
26/08/2023	se trabaja en vista de publicacion de depositos WareUp - Desarrollo y Testing	04:00:28 16:21:18 - 20:21:46	Leti Presa 0.00 USD
26/08/2023	Desarrollo WareUp - Desarrollo y Testing	02:08:00 14:13:00 - 16:21:00	Romina Areco 0.00 USD
26/08/2023	Investigando para el listado de publicaciones WareUp - Investigación y Capacitación	01:39:00 11:32:03 - 13:11:03	Leti Presa 0.00 USD
26/08/2023	cambiar layout de admin WareUp - Desarrollo y Testing	03:09:40 08:10:22 - 11:20:02	Leti Presa 0.00 USD
25/08/2023	Desarrollo WareUp - Desarrollo y Testing	08:00:00 20:27:00 - 04:27:00 <sup>+1</sup>	Romina Areco 0.00 USD
25/08/2023	trabajando en las imágenes de los depositos WareUp - Desarrollo y Testing	02:00:00 19:00:00 - 21:00:00	Leti Presa 0.00 USD
25/08/2023	conectando logica de registro deposito + agregado de imagenes WareUp - Desarrollo y Testing	01:48:36 10:14:47 - 12:03:23	Leti Presa 0.00 USD
25/08/2023	Agregando listado de depositos para editarlos WareUp - Desarrollo y Testing	02:29:00 10:00:00 - 12:29:00	Leti Presa 0.00 USD

Ilustración 137: Detalle Clockify

## Anexo 15 - Integración con Google Analytics

A continuación se detalla la guía proporcionada al cliente para realizar la integración con Google Analytics.

En la era digital actual, la toma de decisiones informadas es esencial para el éxito de cualquier proyecto en línea. En nuestro caso, al tener la aplicación desarrollada en React, la integración de Google Analytics se convierte en un recurso invaluable para comprender el comportamiento de los usuarios y el rendimiento de la aplicación. El objetivo de esta guía es proporcionar un enfoque paso a paso para llevar a cabo esta integración.

### ¿Qué beneficios puedo obtener de la Integración con Google Analytics?

- **Seguimiento del Tráfico Web:** Google Analytics te permite realizar un seguimiento detallado del tráfico en la aplicación. Esto incluye información sobre la cantidad de visitantes, su ubicación geográfica, dispositivos utilizados y cómo llegaron a la web.
- **Comprender el Comportamiento del Usuario:** Podemos obtener información sobre cómo los usuarios interactúan con la aplicación. Esto incluye el tiempo que pasan en la web, las páginas que visitan con más frecuencia y las acciones que realizan.
- **Optimización de Contenido:** Con Google Analytics, podemos evaluar el rendimiento del contenido y las características de la aplicación. Esto nos permite identificar áreas de mejora y ajustar la estrategia de contenido.
- **Seguimiento de Conversiones:** Nos permite saber qué usuarios se convierten en cliente ya que Google Analytics nos permite rastrear estos eventos y evaluar qué estrategias son más efectivas.

### Comencemos con la integración

Antes de empezar a configurar nuestra página, debemos asegurarnos de haber iniciado sesión con una cuenta de Google. Luego, debemos dirigirnos a Google Analytics y **crear una cuenta**.

Para crear una cuenta, debemos dirigirnos a **Admin** o **Administración** en caso de tener el contenido en español.

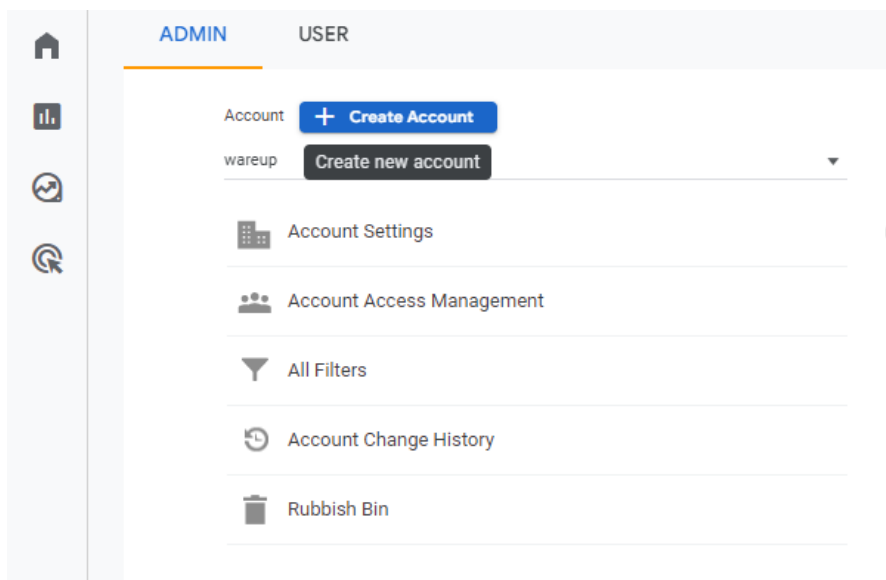


Ilustración 138: Crear cuenta en Google Analytics

Este paso nos va a pedir información de la empresa. Y en el segundo paso, nos va a pedir **generar una propiedad** como se puede visualizar en la siguiente imagen.

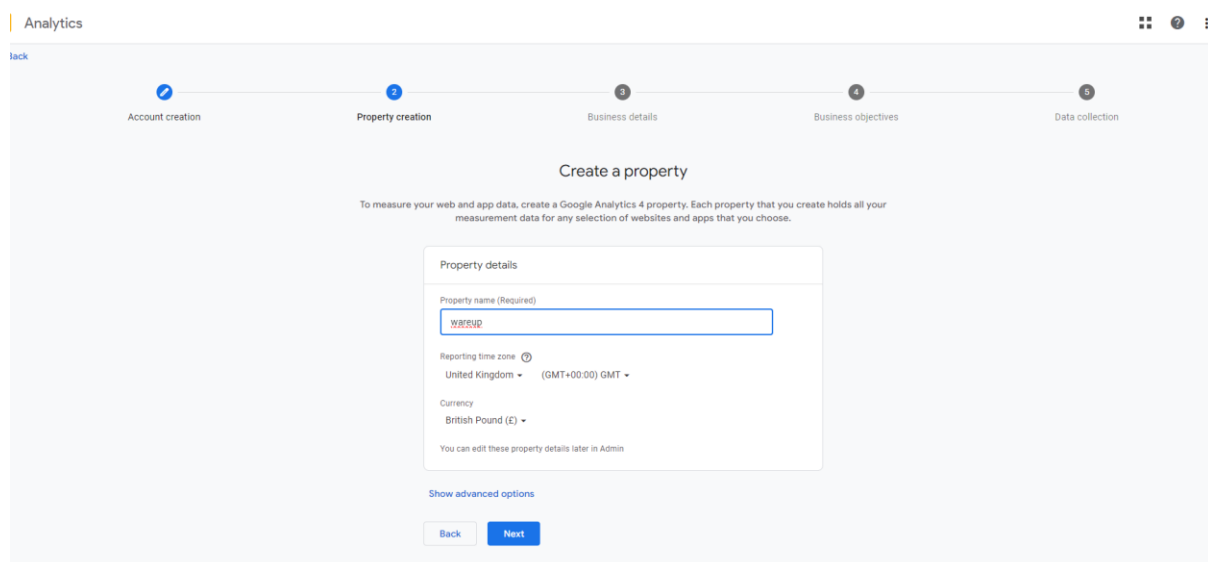


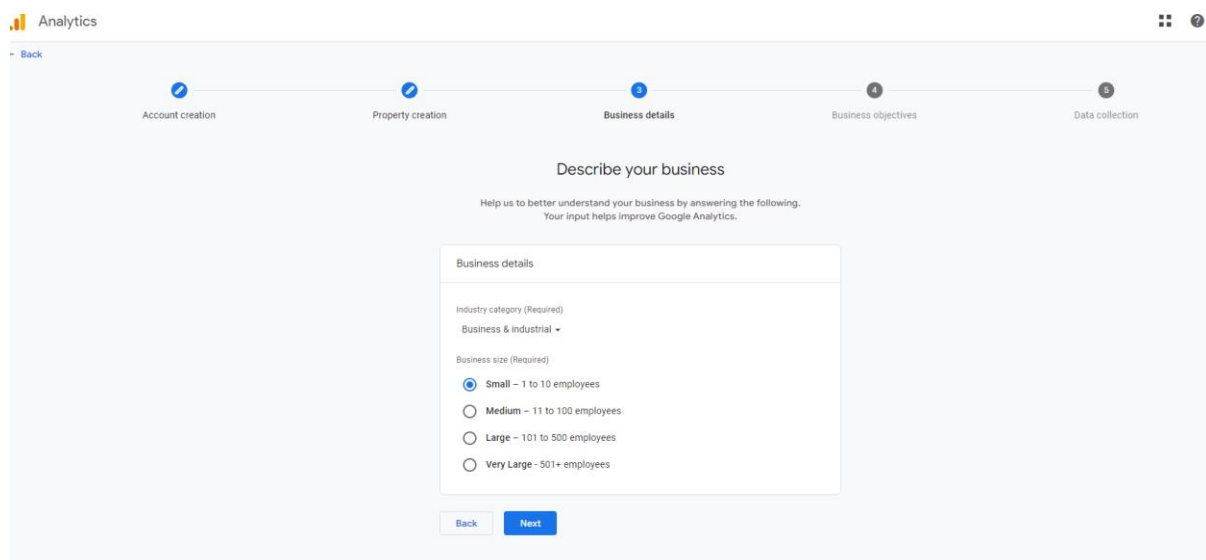
Ilustración 139: Crear *property* en Google Analytics

Debemos configurar nuestra zona horaria para que los registros que veamos luego estén en el horario correcto.

Para mayor contexto, una **propiedad de Google Analytics 4 (GA4)** es una instancia independiente de seguimiento de datos. Esto significa que cada sitio web o aplicación que deseamos rastrear debe configurarse como su propia propiedad dentro de nuestra cuenta de GA4. Cada propiedad tiene su propio conjunto de datos y configuraciones específicas. Podemos considerar una propiedad como un contenedor que almacena los datos recopilados para un sitio web o aplicación en particular.

Y **Google Analytics 4 (GA4)** es la última versión de la plataforma de análisis de Google que proporciona información detallada sobre el comportamiento de los usuarios en un sitio web o una aplicación. En GA4, una "*property*" se refiere a una entidad específica que representa un sitio web o una aplicación móvil que se está rastreando y analizando.

Luego de definir el nombre de esta propiedad que será fundamental para el seguimiento de datos como mencionamos anteriormente, tendremos que cargar más información del negocio:



The screenshot shows the 'Business details' step of the Google Analytics 4 setup wizard. At the top, there is a progress bar with five steps: 'Account creation', 'Property creation', 'Business details' (current step), 'Business objectives', and 'Data collection'. Below the progress bar, the heading 'Describe your business' is followed by the instruction: 'Help us to better understand your business by answering the following. Your input helps improve Google Analytics.' The main content area is titled 'Business details' and contains two sections: 'Industry category (Required)' with a dropdown menu set to 'Business & industrial', and 'Business size (Required)' with four radio button options: 'Small - 1 to 10 employees' (selected), 'Medium - 11 to 100 employees', 'Large - 101 to 500 employees', and 'Very Large - 501+ employees'. At the bottom of the form, there are 'Back' and 'Next' buttons.

Ilustración 140: Ingresar información del negocio

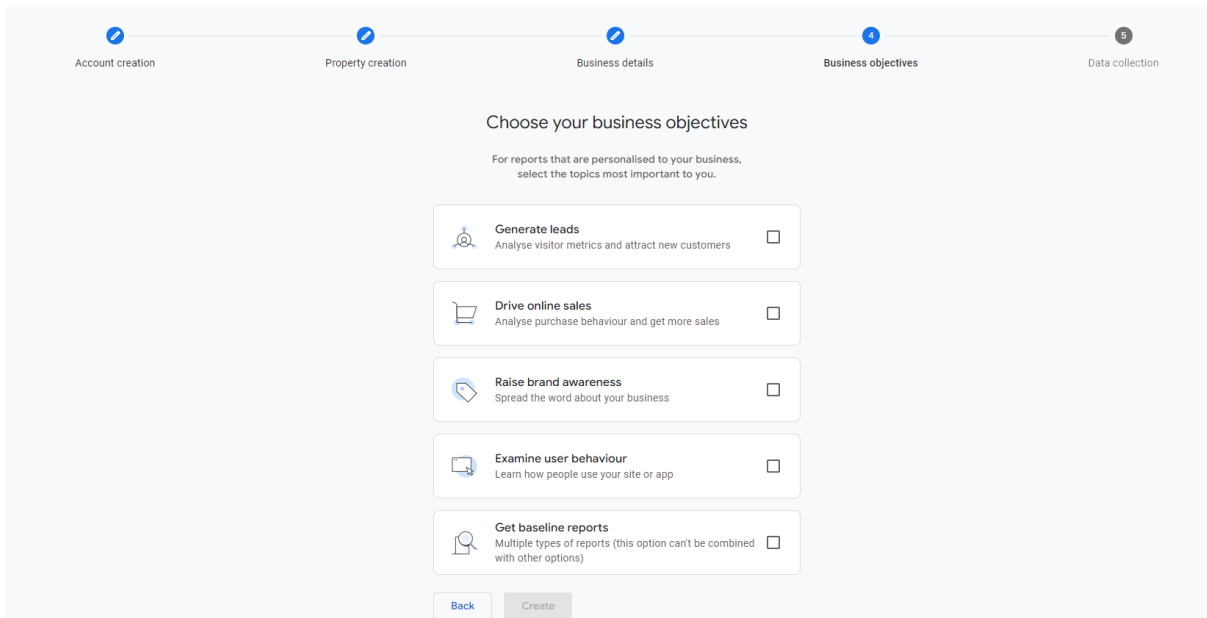


Ilustración 141: Ingresar los objetivos del negocio con Google Analytics

Antes de finalizar nuestro registro tendremos que aceptar los términos y condiciones:

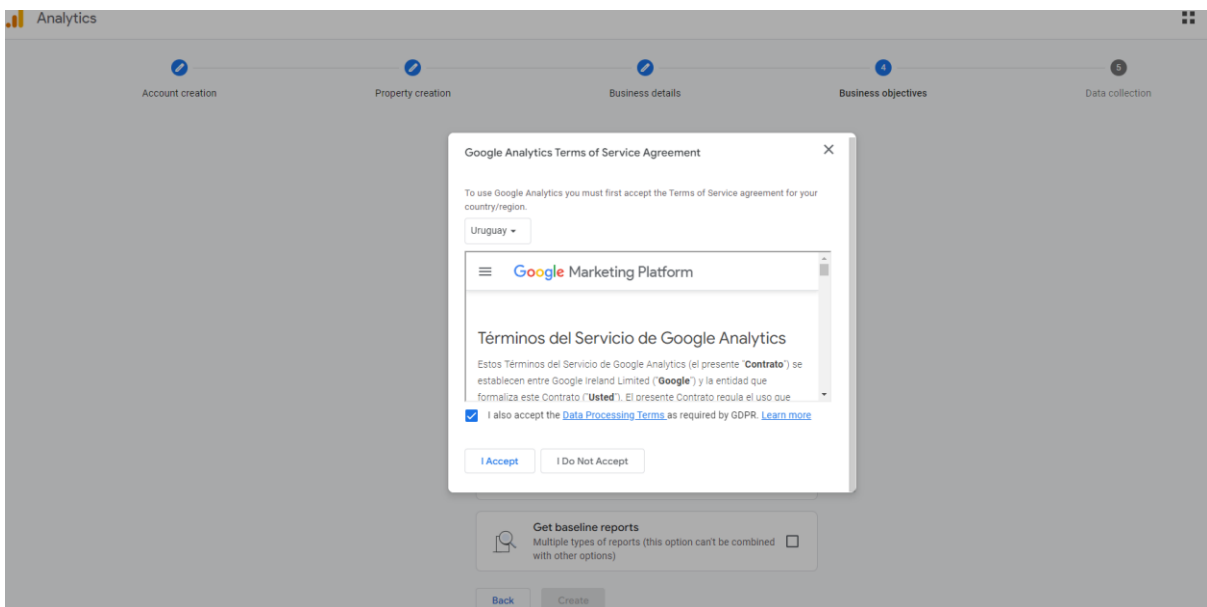


Ilustración 142: Aceptar términos y condiciones del registro

Y en caso de que queramos ya comenzar a trackear la información nos permite seleccionar la plataforma que vamos a estar monitoreando y posteriormente que ingresamos la URL a monitorear:

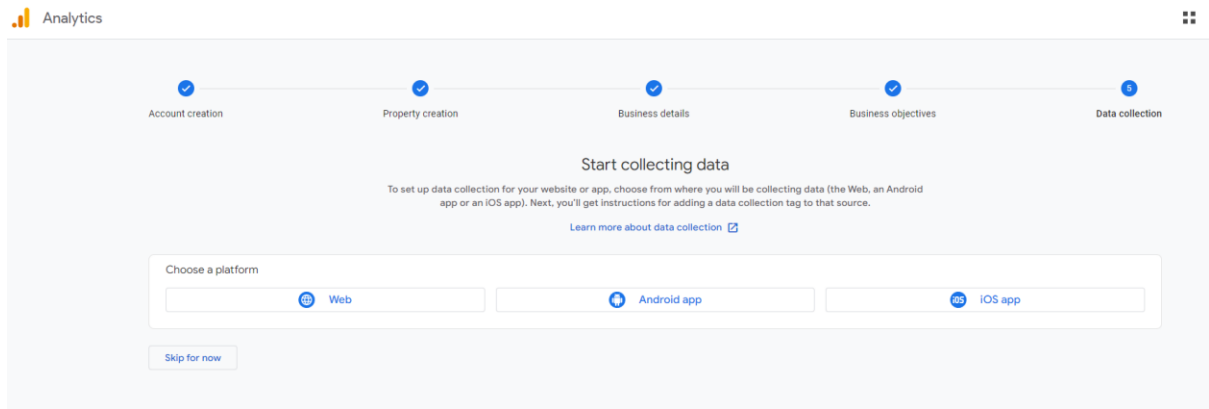


Ilustración 143: Seleccionar plataforma a monitorear

× Set up data stream

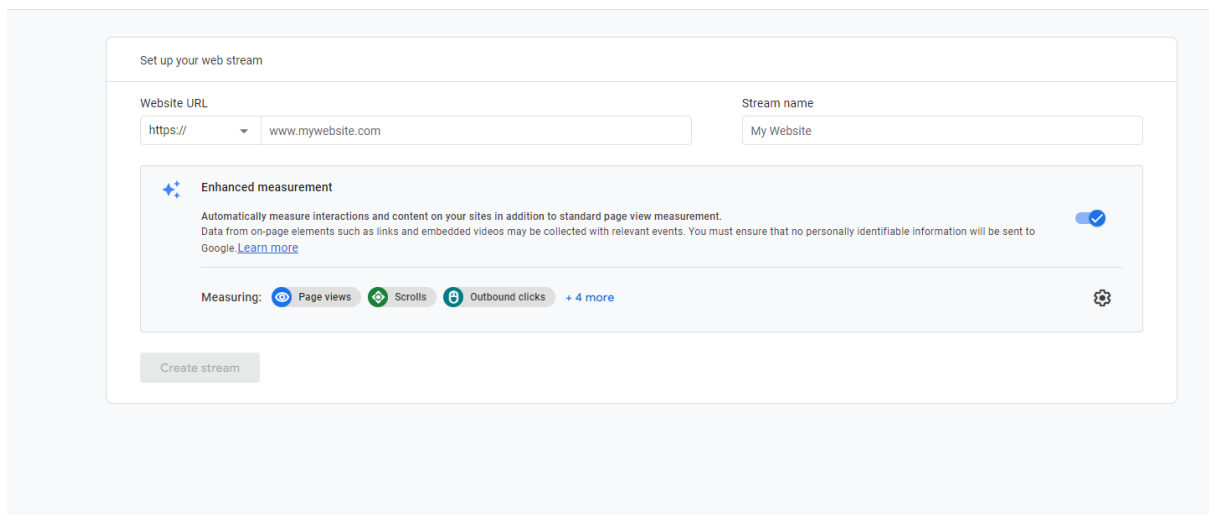


Ilustración 144: Ingresar URL a monitorear

Una vez que confirmamos los datos, nos aparecerá un resumen como el siguiente:

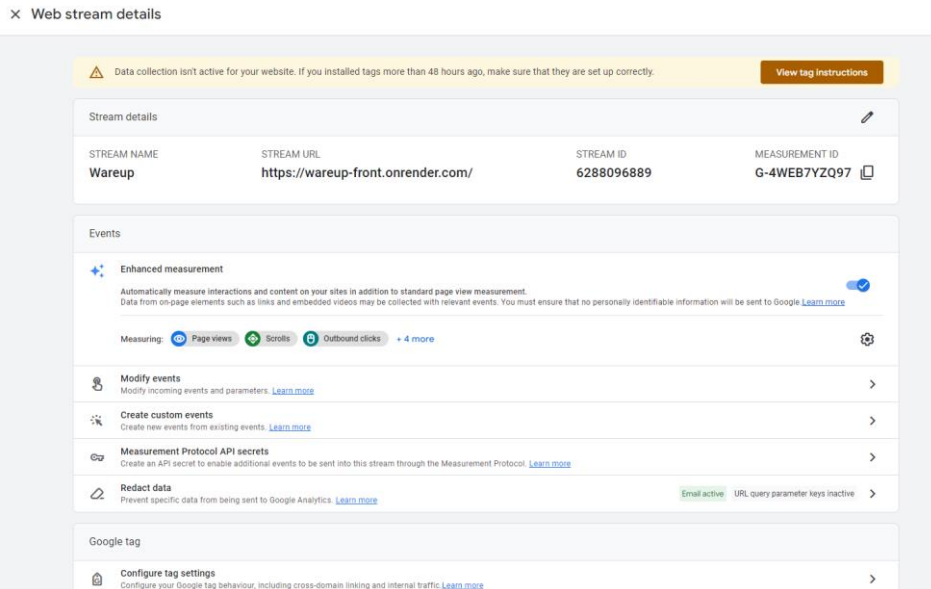


Ilustración 145: Resumen de la creación

**Importante:** debemos guardar el **MEASUREMENT ID** ya que es el código G4 que debemos configurar en nuestra aplicación web para comenzar a trackear la información y visualizarla en Google Analytics. En este ejemplo, el código es **G-4WEB7YZQ97**. Este código es a modo de prueba y visualización en la guía de integración, no se debe usar para consumo propio.

### ¿Qué vemos en la home?

Cuando recién generamos nuestra propiedad GA4 nuestra home se verá de la siguiente manera:

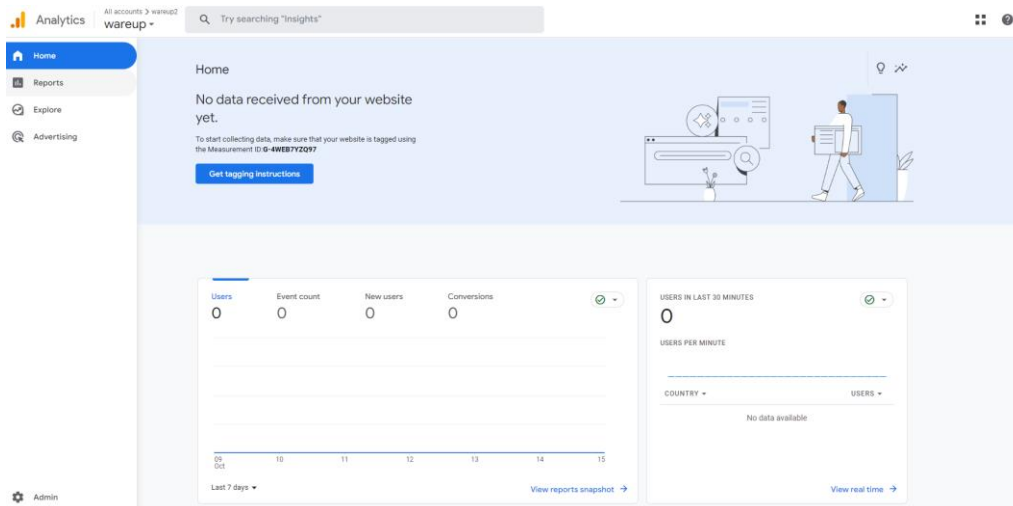


Ilustración 146: Home Google Analytics

Para empezar a ver nuestro tráfico tenemos que ir a nuestra aplicación del frontend y hacer algunas configuraciones.

### Configuración en aplicación frontend

Antes de comenzar, para toda integración de Google Analytics es necesario instalar la librería necesaria. En nuestro caso, ya está instalada por ende no es necesario realizar este paso.

```
npm install react-ga --save
```

Luego, en src/App.js archivo principal de nuestro *frontend*, tenemos la siguiente configuración que es la que nos permite trackear los datos de la aplicación y se muestren en Google Analytics.

```
7
8 export default function App() {
9   useEffect(() => {
10     window.dataLayer = window.dataLayer || [];
11     function gtag() {
12       window.dataLayer.push(arguments);
13     }
14     gtag("js", new Date());
15     gtag("config", "G-XXXXXXXXXX");
16   }, []);
17
18   return (
19     <AuthProvider>
20       <ThemeProvider theme={theme}>
21         <BrowserRouter>
22           <WebRouter />
23         </BrowserRouter>
24       </ThemeProvider>
25     </AuthProvider>
26   );
27 }
28
```

Ilustración 147: Código en App.js

**Importante:** en la línea 15 donde tenemos un código **G-XXXXXXXXXX** se debe completar con el **MEASUREMENT ID** que nos devolvió al crear la propiedad en el paso anterior.

Como se puede apreciar en la siguiente imagen, con esta simple configuración si vamos a **Reports - Real-time** ya podemos tener información en Google Analytics:

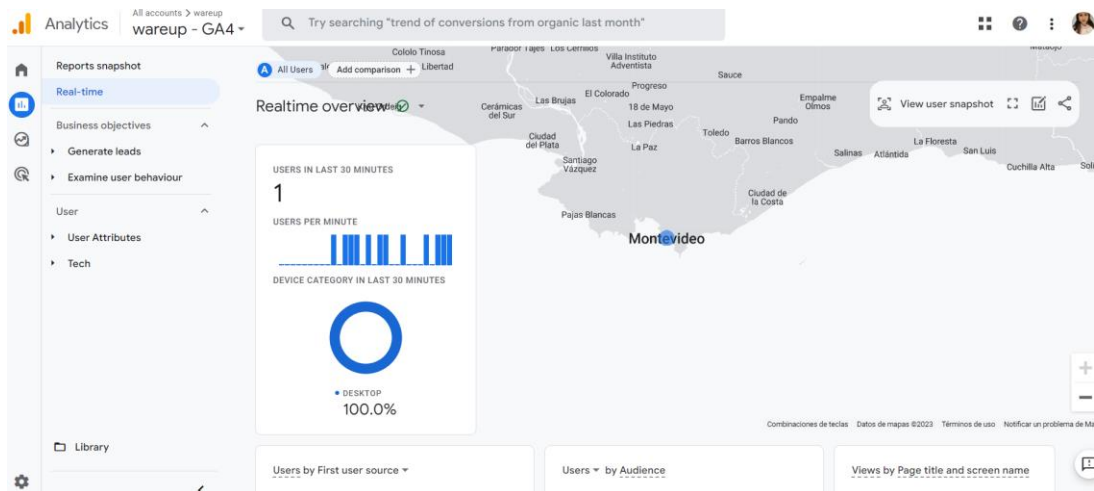


Ilustración 148: Reporte Real-time en Google Analytics

Ahora, queremos tener datos más personalizados para nuestra aplicación. Para hacer esto, se configuraron eventos para las acciones que consideramos fundamentales en ser trackeadas.

Estas acciones son:

- Ingreso a las distintas páginas de la web
- Registros de entidades
- Seleccionar el botón “Tengo espacio” para ver cuántas personas tienen un interés inicial por registrar sus espacios.

Quisimos mejorar la experiencia de cuando se ingresa a las páginas y les configuramos el título para que se vean más amigables en Google Analytics. Para hacer esto, en cada una de las páginas definimos el título de la siguiente manera:

```
useEffect(() => {
  document.title = "Ware Up";
}, []);
```

Quedando el código como se muestra a continuación:

```

7   export function UserProfile() {
8     const { user } = useAuth();
9     useEffect(() => {
10      document.title = "Mi perfil";
11    }, []);
12
13    return (
14      <Box>
15        <UserInformationProfile user={user} />
16        <Footer />
17      </Box>
18    );
19  }
20

```

Ilustración 149: Agregado de título en las páginas de la web

### ¿Cómo se visualiza esta información en Google Analytics?

Veremos el título de la página acompañado por la cantidad de visitas. Cabe destacar que esta vista es en base a los resultados de **Real-time** ya que los datos en modo reporte demoran en aparecer. Esto será detallado más adelante en [Demoras al actualizar la información.](#)

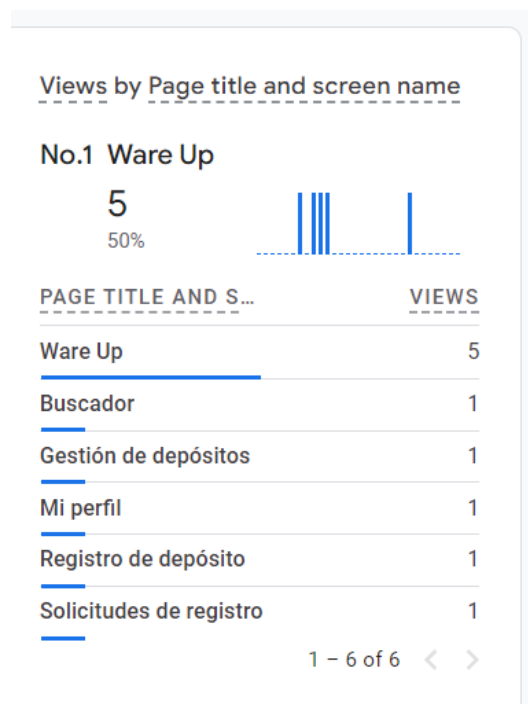


Ilustración 150: Páginas vistas y cantidad de views en Google Analytics

Además de visualizar las páginas que se visitan, podemos ver los distintos eventos que se realizan sobre la web.

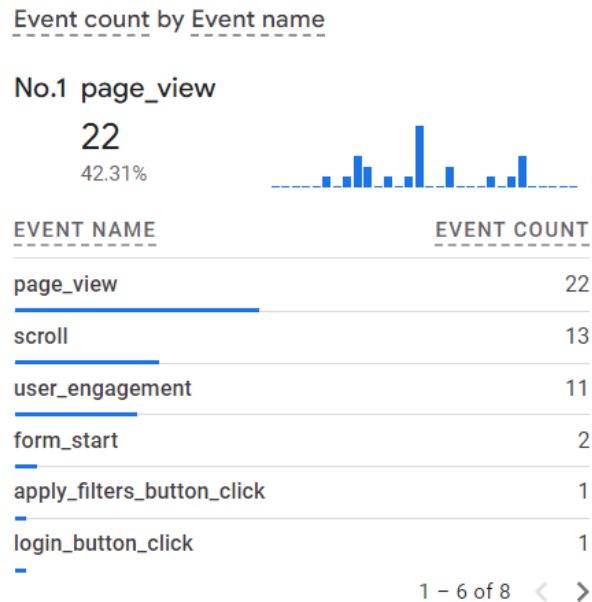


Ilustración 151: Eventos realizados en Google Analytics

### ¿Cómo podemos tener un registro personalizado de los botones que se seleccionan?

Nos importaba saber cuando un botón en particular se seleccionaba, y no solo tener eventos de “click” genéricos, para hacer esto tuvimos que agregar el siguiente código en las acciones de los botones:

```
const handleClickOpen = () => {
  window.gtag("event", "request_booking_button_click", {
    method: "click",
  });
  setOpen(true);
};
```

Quedando el código de la siguiente manera:

```

export function Description({ depositId }) {
  const { accessToken } = useAuth();
  const [deposit, setDeposit] = useState(null);
  const [open, setOpen] = useState(false);

  const handleClickOpen = () => {
    window.gtag("event", "request_booking_button_click", {
      method: "click",
    });
    setOpen(true);
  };

  const handleClose = () => {
    setOpen(false);
  };
}

```

Ilustración 152: Registro de acciones de “click”

A continuación se muestra cómo se visualiza en Google Analytics estos eventos:

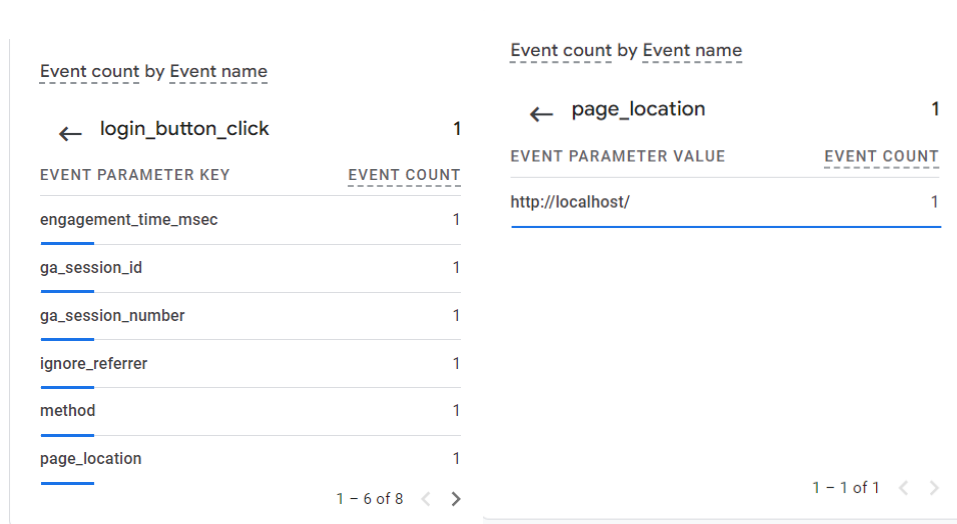


Ilustración 153: Evidencia del evento click en el botón de Login

### ¿Cómo podemos monitorear las entidades que se registran en el sistema?

Para tener registro de este tipo de eventos, fue necesario agregar el siguiente código cuando se confirma que un registro se realizó exitosamente:

```

window.gtag("event", "register", {
  event_category: "Registrations",
  event_label: "User registration",
});

```

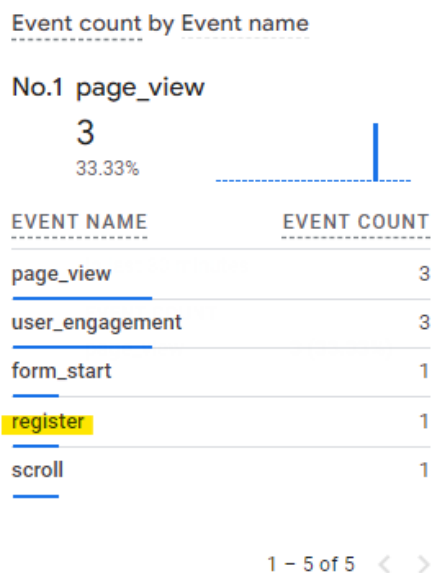
Quedando el código de la siguiente manera:

```
const formik = useFormik({
  initialValues: initialValues(),
  validationSchema: validationSchema(),
  validateOnChange: false,
  onSubmit: async (formValue, { resetForm }) => {
    try {
      await companyController.register(accessToken, user, formValue);
      setNotificationMessage("Empresa registrada exitosamente");
      setNotificationSeverity("success");
      setNotificationOpen(true);

      window.gtag("event", "register", {
        event_category: "Registrations",
        event_label: "Company registration",
      });
      resetForm();
    } catch (error) {
      const errorMessage = "Error: " + JSON.stringify(error.message);
      setNotificationMessage(errorMessage);
      setNotificationSeverity("error");
      setNotificationOpen(true);
    }
  },
});
```

Ilustración 154: Registro de acciones de “click”

Y en Google Analytics podemos visualizarlos de la siguiente manera:



Event count by Event name

← register 1

EVENT PARAMETER...	EVENT COUNT
engagement_time_m...	1
event_category	1
event_label	1
ga_session_id	1
ga_session_number	1
ignore_referrer	1

1 - 6 of 8 < >

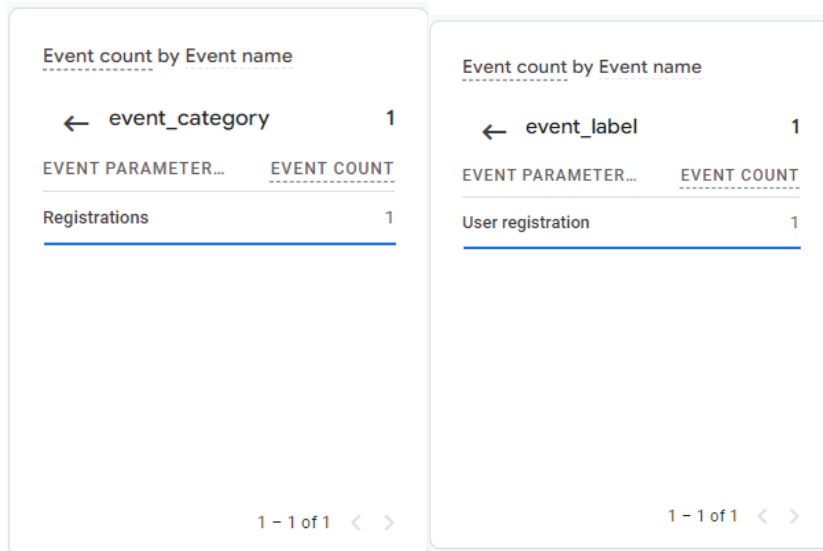


Ilustración 155: Evidencia de eventos de registro en Google Analytics

### Demoras al actualizar la información

El reporte **Real Time** nos asegura que tiene como máximo 1 minuto de demora en actualizarse.

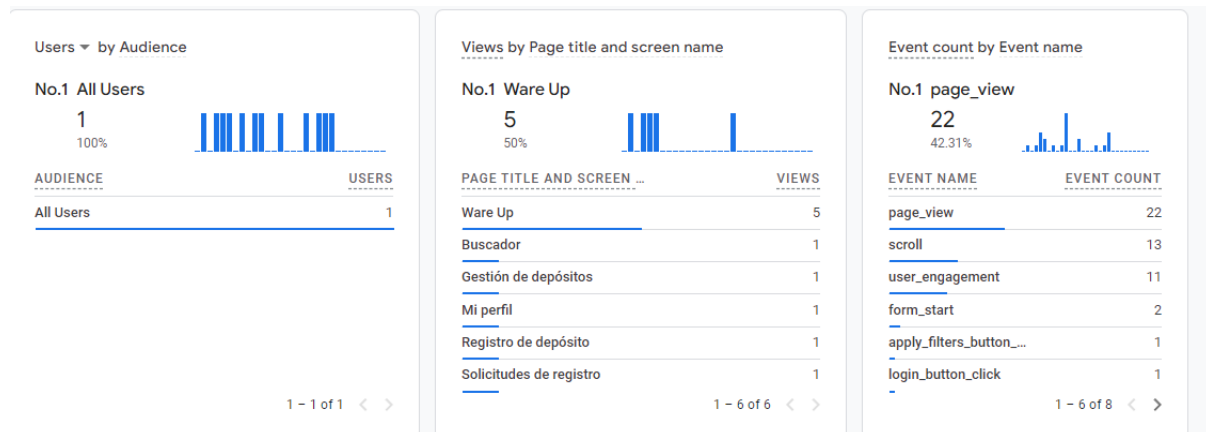


Ilustración 156: Evidencia del reporte Real-time en Google Analytics

Sin embargo, para obtener más reportes y poder hacer comparativa de información, estos son los tiempos que Google maneja:

Interval	Typical processing time	Properties	Data limits per property	Query coverage
Realtime	Less than 1 minute	360, Standard	None	Limited to a few dimensions and metrics
360 intraday	About 1 hour	360	Premium Normal and Premium Large as defined <a href="#">here</a>	All reports and API queries, except <a href="#">these</a>
Standard intraday	4-8 hours	Standard	Standard Normal	All reports and API queries, except <a href="#">these</a>
Daily	12 hours	360, Standard	Standard, Premium Normal	All reports and API queries
Daily	18 hours	360, Standard	Premium Large	All reports and API queries
Daily	24+ hours	360, Standard	Premium XLarge	All reports and API queries

Ilustración 157: Demoras en actualizar información en Google Analytics

A continuación se puede visualizar información en Google Analytics pasado el tiempo necesario para que se actualice la información en el resto de los reportes.

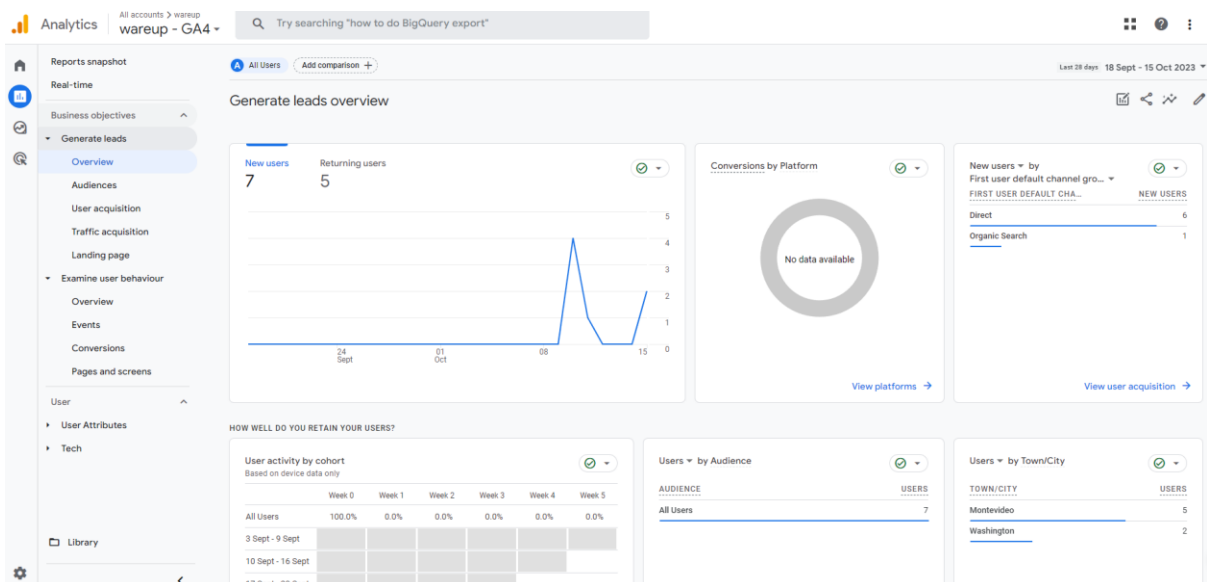


Ilustración 158: Reporte general con información de los últimos 28 días

Para cada reporte se puede indicar un rango de tiempo que queremos visualizar, por ejemplo, registros del último mes.

Algunos reportes interesantes a tener en cuenta:

- Cantidad de eventos realizados
- Páginas más visitadas de la web

- Información demográfica de los usuarios

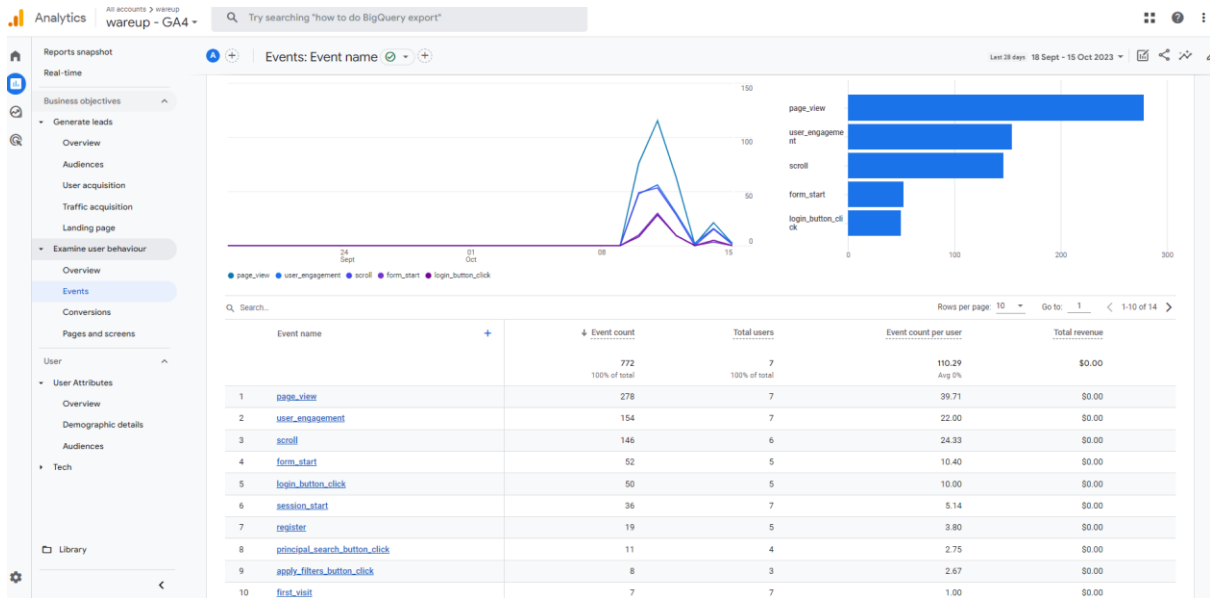


Ilustración 159: Reporte general de eventos realizados en los últimos 28 días

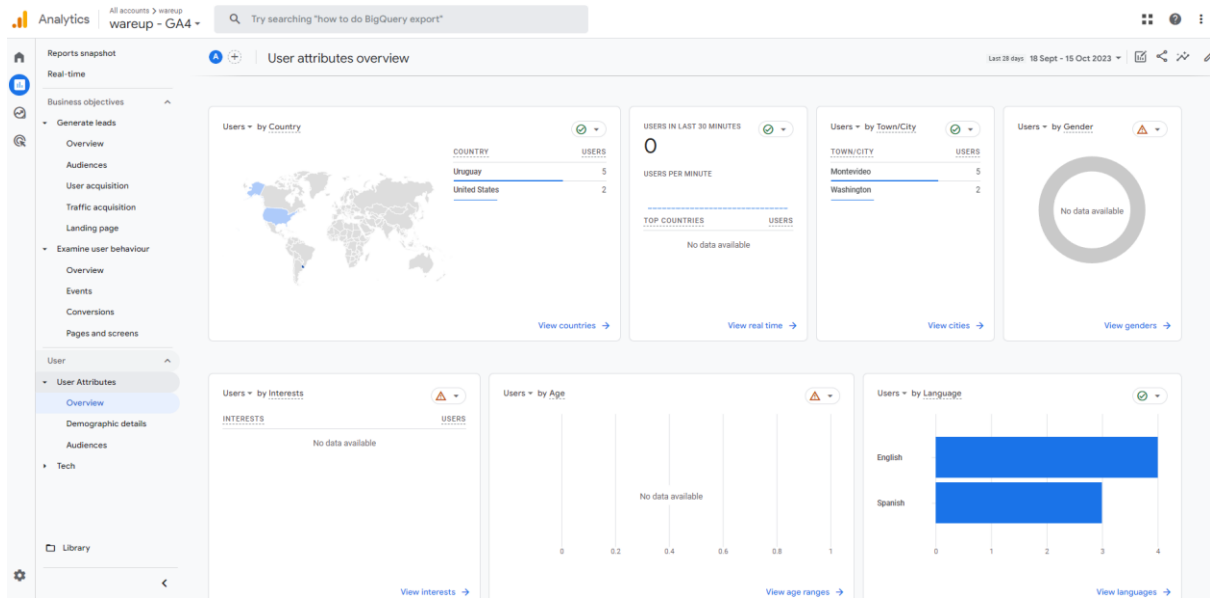


Ilustración 160: Reporte de información usuarios en los últimos 28 días

## Anexo 16 - Integración con Google Maps

### Pre requerimientos

- Contar con una cuenta en gmail

- Crear un proyecto en [Google Developer Console](#)

Para hacer la integración, necesitamos hacer dos configuraciones importantes:

1. Google API Key
2. Maps JavaScript API

## Google API Key

En Google Developer Console, luego de haber creado nuestro proyecto, vamos a **APIs & Services** → **Credentials**.

Seleccionamos **CREATE CREDENTIALS** y elegimos **API key**:

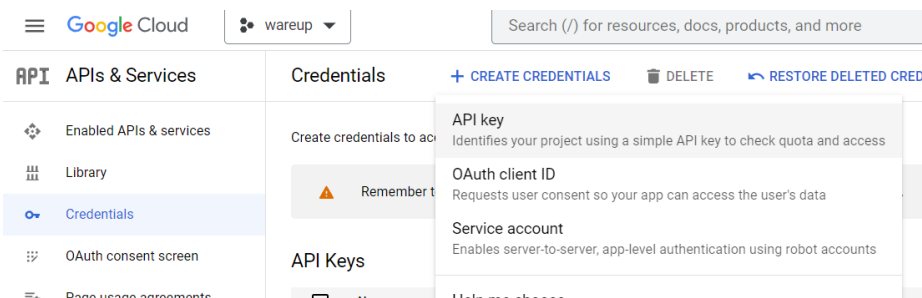


Ilustración 161: Crear nueva API Key en Google Developer Console

Nos aparecerá un mensaje indicando que la API Key fue creada y mencionando cuál es. Si queremos ya la podemos guardar, sino vamos a poder verla más adelante.

### API key created

Use this key in your application by passing it with the `key=API_KEY` parameter.

Your API key

AIzaSyDG1Gy1LHtC3KiHHVNCr2W2ecwQ0SMwb74

⚠ This key is unrestricted. To prevent unauthorized use, we recommend restricting where and for which APIs it can be used. [Edit API key](#) to add restrictions. [Learn more](#)

CLOSE

Ilustración 162: API Key generada exitosamente

Vamos a ver que la **API Key** creada nos queda con un mensaje de warning. Esto nos indica que no tenemos restricciones a cómo se puede utilizar la API key, como lo tenemos hoy podemos tener tráfico no autorizado o sobrepasar la cuota de uso. Para esto, debemos modificar la configuración de la API Key seleccionando **Edit settings**.

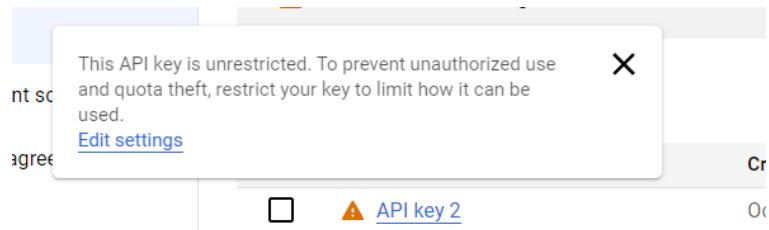


Ilustración 163: Advertencia de API Key pública sin restricciones

Sugerimos considerar la siguiente configuración para hacer uso de la API Key:

#### Set an application restriction

Application restrictions limit an API key's usage to specific websites, IP addresses, Android applications, or iOS applications. You can set one application restriction per key.

- None
- Websites
- IP addresses
- Android apps
- iOS apps

#### API restrictions

API restrictions specify the enabled APIs that this key can call

- Don't restrict key  
This key can call any API
- Restrict key

5 APIs

#### Selected APIs:

Geocoding API  
Geolocation API  
Maps Embed API  
Maps JavaScript API  
Places API

Note: It may take up to 5 minutes for settings to take effect

Ilustración 164: Configuración recomendada para API Key

Para el uso de Google Maps es necesario consumir las APIs que se pueden visualizar en la imagen, en caso de utilizar esta API Key para otra funcionalidad de Google, se sugiere revisar qué APIs deberían ser habilitadas en esta sección.

Una vez configurada correctamente, la API Key nos aparecerá con un check en verde:

## API Keys

<input type="checkbox"/>	Name	Creation date ↓
<input type="checkbox"/>	✓ <a href="#">API key 1</a>	Sep 16, 2023

Ilustración 165: Confirmación de API Key configurada correctamente

## Maps JavaScript API

Ahora debemos habilitar las APIs que queremos utilizar en nuestro proyecto. Para esto, nos dirigimos a **Enabled APIs & Services**.

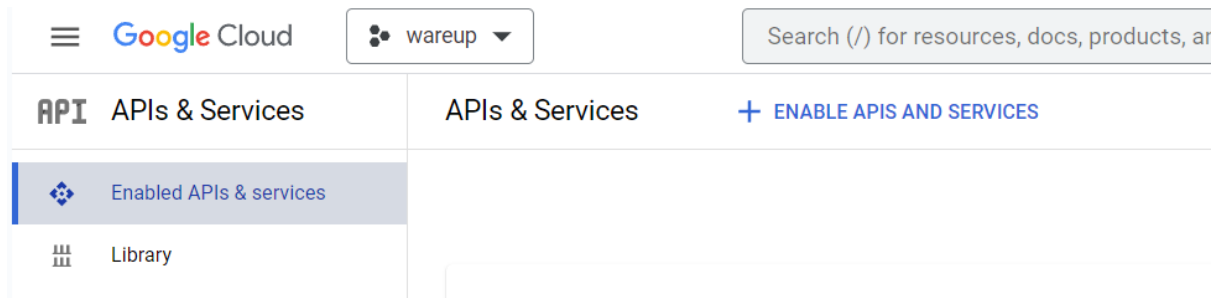


Ilustración 166: Enables APIs & Services en Google Developer Console

Seleccionamos **ENABLE APIS AND SERVICES** y buscamos **Maps JavaScript API**.

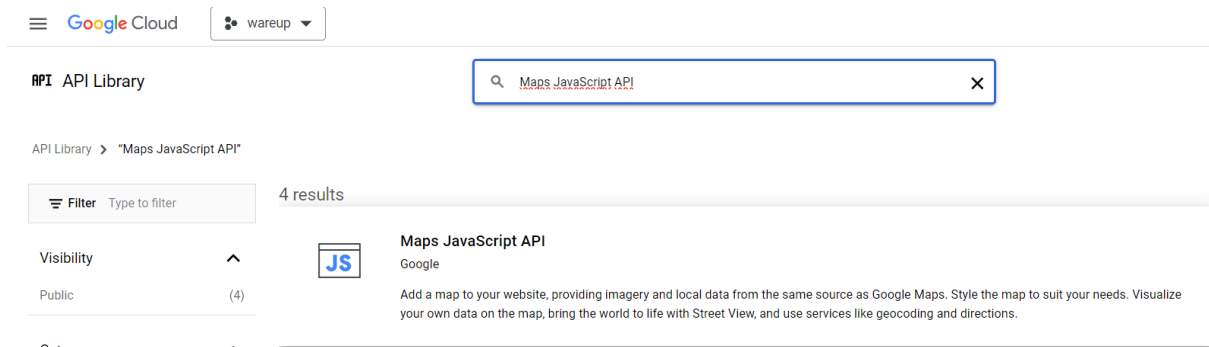


Ilustración 167: Habilitar Maps JavaScript API

Accedemos a la API del resultado y seleccionamos **ENABLE** quedando de la siguiente manera:

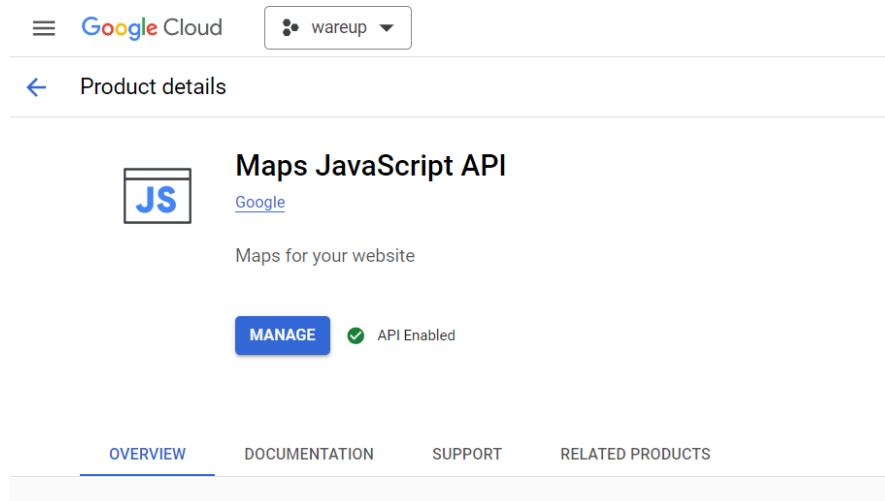


Ilustración 168: Confirmación Maps JavaScript API habilitada

Debemos repetir este paso con **Geocoding API** que nos permitirá obtener las coordenadas para una dirección en particular.

### ¿Cómo integramos Google Maps con nuestro proyecto en React?

Una vez configurado el proyecto en Google Developer Console y habilitado las APIs correspondientes, llega la parte de integrarlo con nuestro código React.

#### Configurar Google API Key

Vamos a `src/utis/constants.js` tenemos una constante definida `API_KEY` → `GOOGLE_MAPS` que debe llevar la API Key que generamos en la Google Developer Console. La API Key que tenemos ahora es meramente para pruebas, una vez que el proyecto sea entregado esta API Key será deshabilitada.

La generación del mapa con los distintos depósitos se puede encontrar en `src/components/Map/DepositsMap.js`

# Anexo 17 - Resultados de encuesta de satisfacción del usuario final

Seleccione su franja etaria

10 respuestas

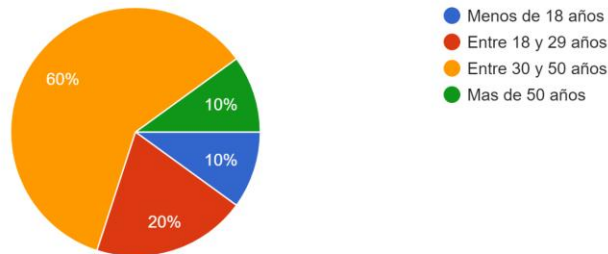


Ilustración 169: Pregunta 1 de encuesta satisfacción del usuario final

Te pedimos que ingreses a la plataforma y busques un almacenamiento en el barrio Aguada, que cuente con cámaras de seguridad. Cuanto tiempo te tomo realizar esta acción?

10 respuestas

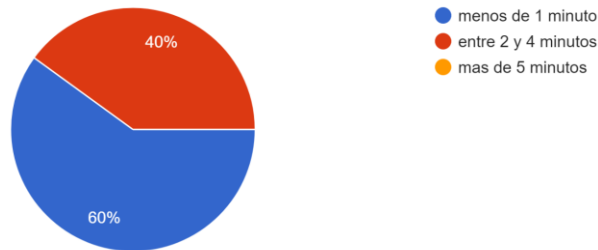


Ilustración 170: Pregunta 2 de encuesta satisfacción del usuario final

Que oportunidades de mejora le encuentras al flujo anterior?

10 respuestas

Ninguna
Ninguna, me parece muy buena
Quizás se puede agregar filtro de cámara de seguridad
.
Parece facil de usar, no veo nada para mejorar
Nada
Elegir otros colores?
Las imagenes de los depositos un poco mas grandes.
Nada

Ilustración 171: Pregunta 3 de encuesta satisfacción del usuario final

Me siento cómodo/a usando la plataforma

10 respuestas

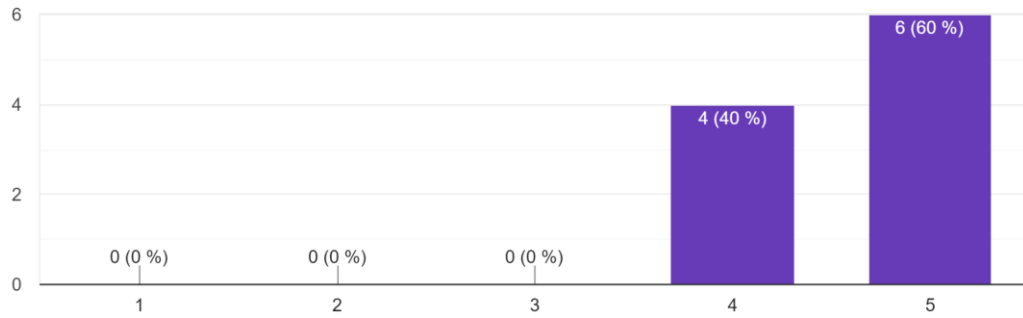


Ilustración 172: Pregunta 4 de encuesta satisfacción del usuario final

Me resulta fácil navegar entre las diferentes funcionalidades.

10 respuestas

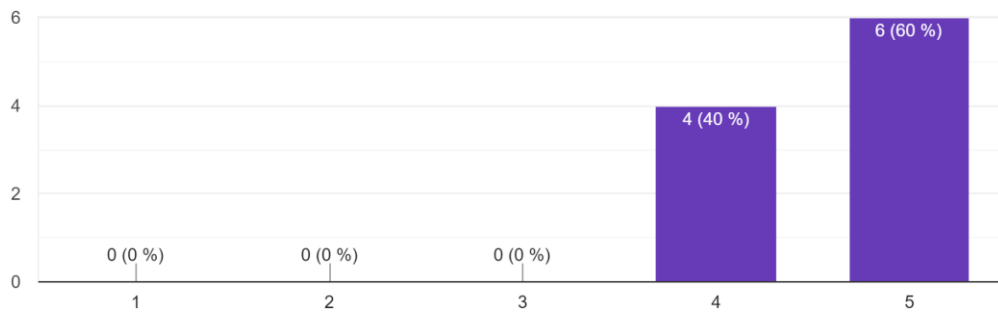


Ilustración 173: Pregunta 5 de encuesta satisfacción del usuario final

Se requiere ayuda para completar una acción

10 respuestas

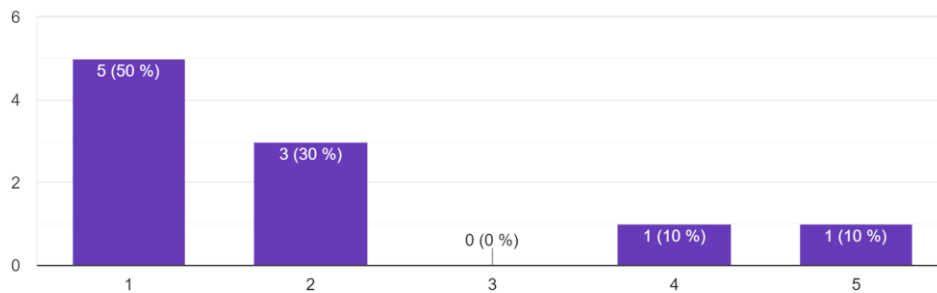


Ilustración 174: Pregunta 6 de encuesta satisfacción del usuario final

Algunas funcionalidades resultan complejas de utilizar

10 respuestas

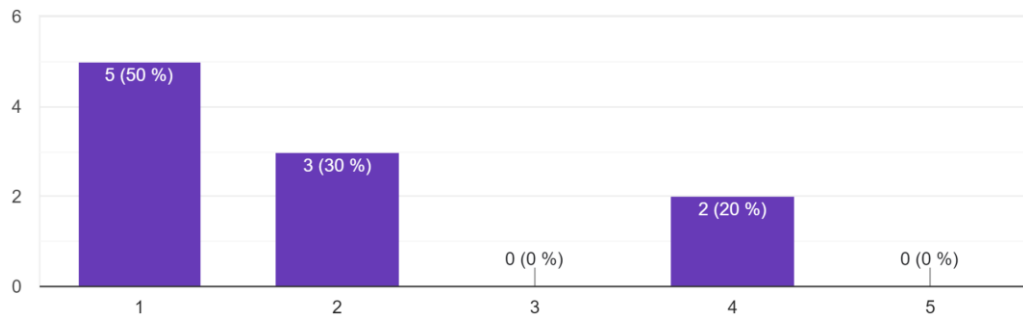


Ilustración 175: Pregunta 7 de encuesta satisfacción del usuario final

Reconozco todos los elementos que aparecen en pantalla

10 respuestas

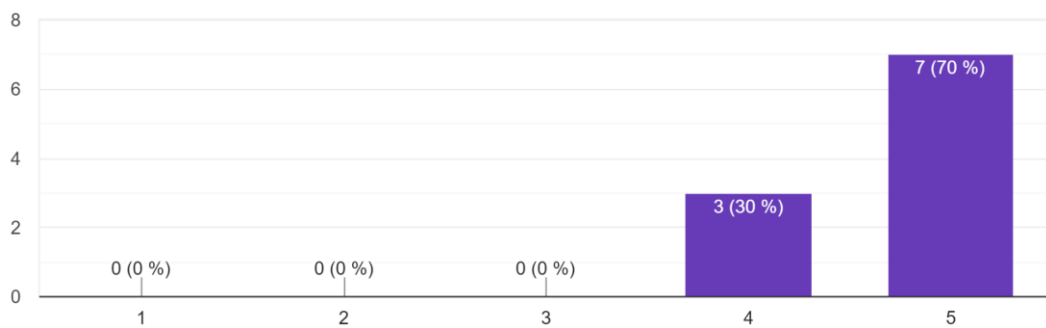


Ilustración 176: Pregunta 8 de encuesta satisfacción del usuario final

Nivel de satisfacción general con el diseño

10 respuestas

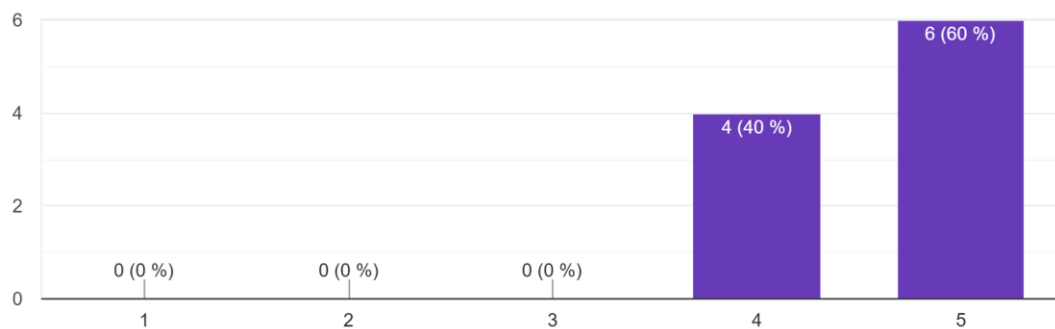


Ilustración 177: Pregunta 9 de encuesta satisfacción del usuario final

## Anexo 18 - Resultados de encuesta de grado de satisfacción del cliente WareUP

¿Qué tan satisfechos están con el MVP final?

2 respuestas

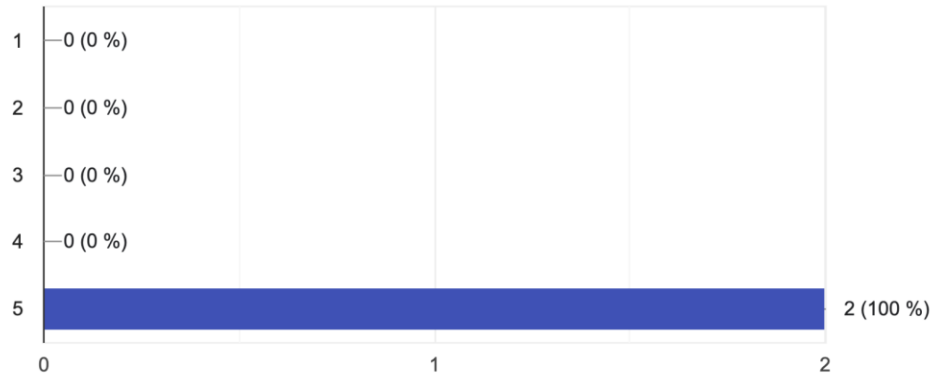


Ilustración 178: Pregunta 1 de encuesta de satisfacción al cliente

Califiquen el nivel de comprensión que el equipo tuvo de sus necesidades empresariales.

2 respuestas

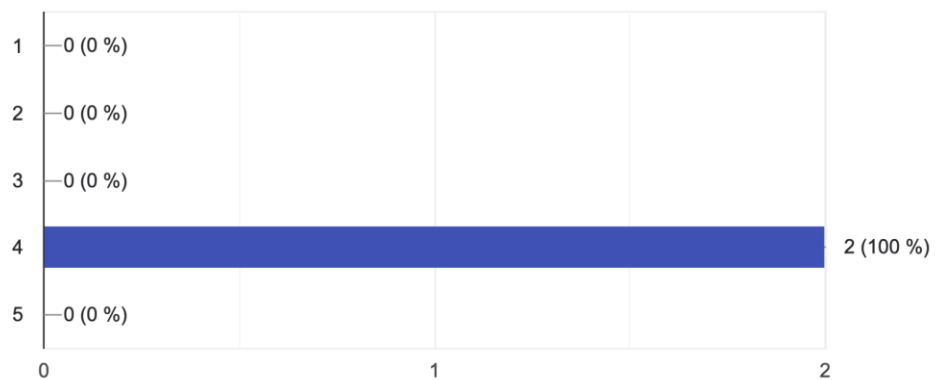


Ilustración 179: Pregunta 2 de encuesta de satisfacción al cliente

¿Qué tan útil creen que les resulte tener como base este MVP para continuar con el desarrollo y poder lanzarlo en producción?

2 respuestas

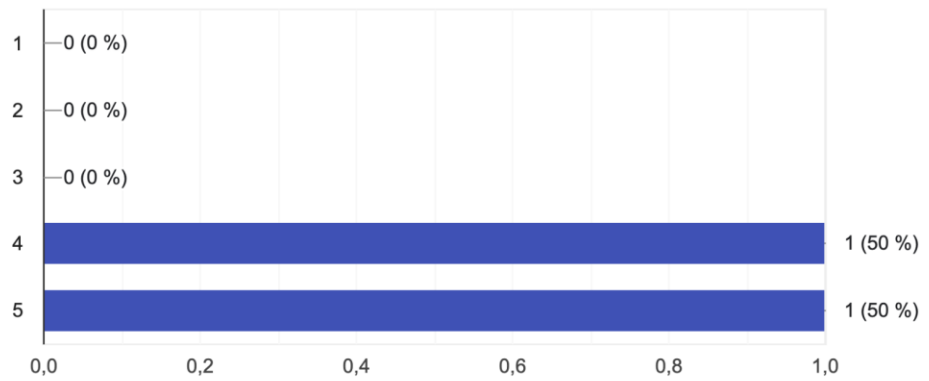


Ilustración 180: Pregunta 3 de encuesta de satisfacción al cliente

¿Cómo consideran que fue la comunicación con el equipo del proyecto a lo largo de estos meses de trabajo?

2 respuestas

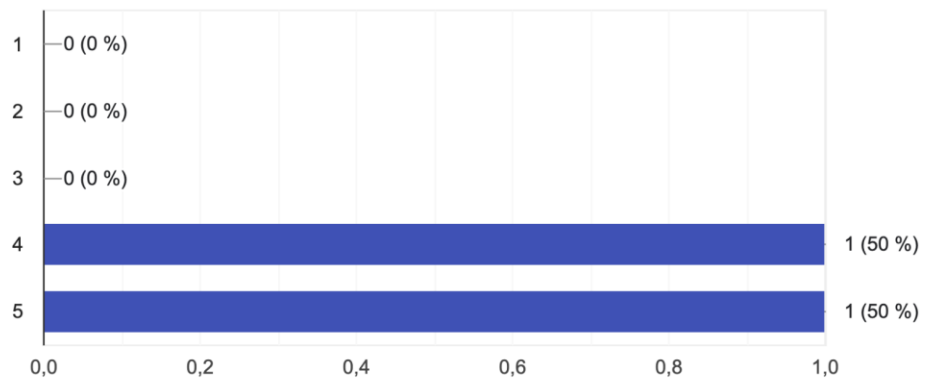


Ilustración 181: Pregunta 4 de encuesta de satisfacción al cliente

¿Cómo consideran que fue la comunicación con el equipo del proyecto a lo largo de estos meses de trabajo?

2 respuestas

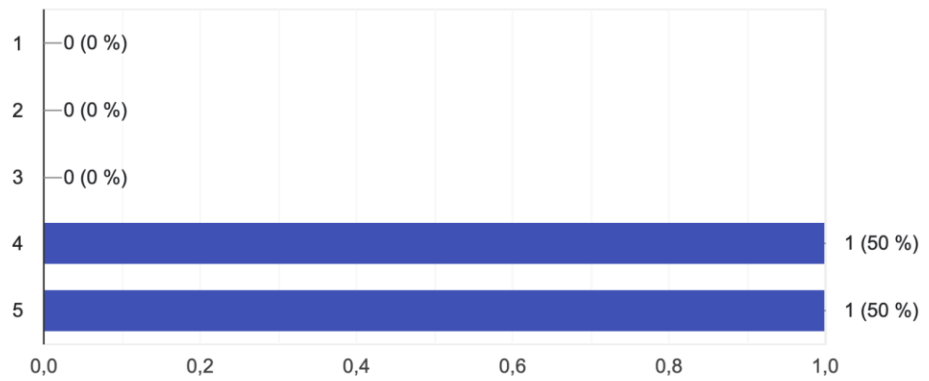


Ilustración 182: Pregunta 5 de encuesta de satisfacción al cliente

¿Qué características les gustaron más del producto?

2 respuestas

El mapa con las ubicaciones

El buscador

Ilustración 183: Pregunta 6 de encuesta de satisfacción al cliente

¿Qué características les gustaron menos del producto?

2 respuestas

La página web creo que se podría mejorar un poco pero funcionalmente me parece bien.

No poder restaurar la contraseña desde el autoperfil

Ilustración 184: Pregunta 7 de encuesta de satisfacción al cliente

Si desean, pueden agregar comentarios adicionales.

2 respuestas

Nada para agregar. Muy buen trabajo chicas!

Muy buen trabajo

Ilustración 185: Pregunta 8 de encuesta de satisfacción al cliente