

Universidad ORT Uruguay  
Facultad de Ingeniería

# Diseño de proceso de outsourcing de software para startups

Entregado como requisito para la obtención del  
título de Ingeniera en Sistemas

Ana Clara Rodríguez Ferragut - 210472

Tutores:  
Martín Solari  
Gerardo Matturro

2021

# Declaración de autoría

Yo, Ana Clara Rodríguez declaro que el trabajo que se presenta en esta obra es de mi propia mano. Puedo asegurar que:

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



Ana Clara Rodríguez

18 de marzo de 2021

“En primer lugar, quisiera agradecer a mis padres por incentivar y apoyarme desde siempre. A mi hermano, por saber comprender las largas noches de estudio y días de poca calma.

A Camila, Julieta y María que me brindaron apoyo moral y me acompañaron durante todo este recorrido. A mi futuro colega Pale, por su amistad y sinceridad en los momentos más claves en estos cinco años. A Vinicio, amigas y amigos que me han comprendido por tantas ausencias y alentado a no bajar los brazos. Sin su incentivo este transcurso hubiera sido distinto.

A mis tutores, profesores y compañeros de curso por brindarme las herramientas y conocimientos para poder formarme y desarrollarme tanto en el ámbito académico como personal.”

- Ana Clara

# Abstract

Una startup de software es una empresa recientemente creada sin historial operativo y rápida en la creación de tecnologías de punta. Este tipo de empresas, desarrollan software bajo altas condiciones de incertidumbre, abordando mercados de rápido crecimiento bajo una grave falta de recursos. Debido a estas características, las mismas tienen una gran tendencia a subcontratar empresas donde tercerizan el desarrollo de software, técnica también conocida como outsourcing.

Bajo el marco de proyectos de fin de carrera, la universidad ORT Uruguay ha generado una línea de investigación que aborda temas ligados a este tipo de startups en particular. Estudiantes de las carreras de Ingeniería y Licenciatura en Sistemas han llevado a cabo proyectos que involucran tanto emprendimientos que pertenecen al Centro de Innovación y Emprendimientos de la universidad (CIE), como a la incubadora Ingenio del Laboratorio Tecnológico del Uruguay (LATU). Dado que las startups continúan siendo emergentes y más aún en el área del desarrollo de software, consideramos necesario avanzar en el conocimiento sobre las mismas.

El objetivo del presente proyecto es poder determinar de qué manera se desarrollan y qué dificultades enfrentan las startups a efectos de generar insumos para elaborar una nueva propuesta de proceso a ser implementada entre las startups y las empresas donde tercerizan su código.

Para cumplir con este objetivo, se ha optado por el estudio de caso como instrumento de investigación ya que el mismo se centra en examinar un fenómeno en su entorno natural, empleando múltiples métodos de recopilación de datos para reunir información de una o unas pocas entidades (personas, grupos u organizaciones).

Se ha utilizado la entrevista como herramienta para la recolección de datos. Se entrevistaron cuatro informantes calificados que cumplían un rol de expertos en empresas especializadas en desarrollo de software y venta de servicios para startups.

Entre los principales resultados, se destaca la elaboración de una propuesta preliminar de proceso que tiene como particularidad la generación de una instancia inicial de intercambio y comunicación efectiva entre las startups y el equipo de outsourcing. Cabe destacar que esta nueva propuesta debe ser puesta en práctica y validada en futuras investigaciones.

# Palabras clave

Startup, startup de software, outsourcing, equipo de outsourcing, proceso, metodología ágil, ingeniería de software, comunicación.

# Índice general

Índice de tablas	8
<b>1 Introducción</b>	<b>9</b>
<b>2 Marco teórico</b>	<b>13</b>
2.1 Startup . . . . .	13
2.2 Startup de software . . . . .	14
2.3 Metodología ágil . . . . .	15
2.4 Lean Startup . . . . .	16
2.5 Lean Software Development . . . . .	17
2.6 Scrum . . . . .	18
2.7 Outsourcing . . . . .	19
2.8 Buenas prácticas . . . . .	21
2.9 Trabajo previo . . . . .	22
2.9.1 Antecedentes nacionales . . . . .	22
2.9.2 Antecedentes internacionales . . . . .	23
<b>3 Metodología de investigación</b>	<b>26</b>
3.1 Objetivos . . . . .	26
3.1.1 Objetivos generales . . . . .	26
3.1.2 Objetivos específicos . . . . .	26
3.2 Preguntas de investigación . . . . .	27
3.3 Enfoque . . . . .	27
3.3.1 Estudio de caso . . . . .	27
3.3.2 La entrevista como herramienta . . . . .	30
3.3.3 Muestra . . . . .	31
3.3.4 Guión de entrevista . . . . .	31
<b>4 Resultados del trabajo de campo</b>	<b>34</b>
4.1 Realización de entrevistas . . . . .	34

4.2 Síntesis de resultados . . . . .	37
4.2.1 Preguntas sobre el proceso de outsourcing . . . . .	37
4.2.2 Preguntas sobre metodologías ágiles . . . . .	39
4.2.3 Preguntas sobre la comunicación . . . . .	40
4.2.4 Preguntas sobre dificultades . . . . .	41
<b>5 Discusión y síntesis del proceso</b>	<b>43</b>
5.1 Proceso Bentos-Pera . . . . .	43
5.2 Propuesta de extensión del proceso para outsourcing . . . . .	48
5.3 Propuesta de buenas prácticas para el desarrollo del proceso . . . . .	54
<b>6 Conclusiones</b>	<b>59</b>
6.1 Lecciones aprendidas . . . . .	59
6.2 Amenazas a la validez . . . . .	59
6.3 Trabajos futuros . . . . .	60
<b>7 Referencias bibliográficas</b>	<b>62</b>

# Índice de tablas

3.1 Relación entre preguntas y objetivos de investigación . . . . .	33
4.1 Descripción de empresas entrevistadas . . . . .	35
4.2 Información de entrevistas realizadas . . . . .	36
4.3 Categorización de preguntas del guión de entrevista . . . . .	37
5.1 Descripción de etapas del proceso B-P . . . . .	47
5.2 Roles necesarios en el proceso liviano de software . . . . .	48
5.3 Hitos de la etapa Requerimientos y descripción de roles involucrados	51
5.4 Hitos de la etapa Validación y descripción de roles involucrados . . .	53
5.5 Flujo de interacción entre las herramientas de comunicación . . . . .	58

# 1 Introducción

Un emprendimiento es un negocio creado a partir de una idea nueva o novedosa con el fin de generar ganancias económicas. Generalmente, estas ideas se caracterizan por querer transformar el mundo resolviendo problemas existentes mediante soluciones innovadoras. Cada vez es más frecuente que las universidades promuevan la cultura emprendedora no solo como forma de explorar nuevas vías profesionales, sino como modo de enfrentar nuevos desafíos donde los alumnos pueden volcar sus habilidades, aptitudes y conocimientos en escenarios que generan impacto positivo en la sociedad.

La ingeniería de software es una disciplina que se enfoca en ofrecer métodos y técnicas estructuradas para desarrollar y mantener software de calidad de manera costeable. Sommerville [1] la define como “una actividad interdisciplinaria que conjunta equipos de personas con diferentes bases de conocimiento. Los equipos de ingeniería de sistemas son necesarios debido al amplio conocimiento requerido para considerar todas las implicaciones de las decisiones en el diseño del sistema.” Por otro lado, asegura que el contexto organizacional de un sistema afecta a su diseño y uso, y es por esto que la ingeniería de software enfatiza que los sistemas de software no existen de forma aislada: se utilizan en un contexto social y organizacional, como por ejemplo las startups, donde los requerimientos se pueden derivar y restringir según ese contexto.

En los últimos años en la Universidad ORT Uruguay, bajo el marco de proyectos de fin de carrera, se han realizado diferentes proyectos de investigación orientados a estudiar los emprendimientos tecnológicos de software. Los mismos han investigado sobre diferentes áreas entre las cuáles podemos destacar las prácticas y procesos de ingeniería de software que son aplicadas, las metodologías de trabajo utilizadas (tanto metodologías tradicionales como ágiles), o las herramientas necesarias para tener un buen desempeño en el emprendimiento.

Dos de los principales antecedentes que justifican la realización de este proyecto son las investigaciones de Galperin - González sobre habilidades técnicas y

blandas en emprendimientos tecnológicos de software [2] y de Bentos y Pera sobre Gestión de Producto Mínimo Viable y modelo de proceso de software ágil para emprendimientos tecnológicos de software [3]. Si bien estos trabajos aportan nuevo conocimiento significativo sobre los emprendimientos de software, una dimensión que ha sido poco estudiada es la que refiere al lugar de la comunicación en el outsourcing.

En esta investigación, consideramos que las startup de software desarrollan software bajo altas condiciones de incertidumbre, abordando mercados de rápido crecimiento bajo una grave falta de recursos. Este tipo de empresas, son caracterizadas por tener poco o nulo historial operativo, explorar los segmentos innovadores del mercado, tener un producto único y ser altamente reactivas, es decir, son capaces de reaccionar rápidamente a cambios en el mercado, tecnologías o productos. Otra dimensión relevante analizada en este estudio son los modelos de tercerización más conocidos: multisourcing, microsourcing y crowdsourcing.

La tercerización, o outsourcing refiere a la actividad de derivar el desarrollo de software a terceros en lugar de desarrollarlo internamente dentro de una empresa.

En base a la literatura revisada, las principales razones por las cuales las empresas deciden utilizar la técnica de outsourcing son la falta de conocimientos técnicos internos y la capacidad de transferir los riesgos al proveedor. Además deben tener en cuenta la reducción de costos, flexibilidad, aplicación de mejores prácticas e innovación.

En síntesis, según los documentos académicos y reportes de investigación consultados, si bien existe evidencia sobre la importancia y el rol que cumple la comunicación en proyectos donde se utiliza el outsourcing, hay escaso conocimiento acumulado sobre las técnicas, buenas prácticas o procesos a llevar a cabo para que la misma se haga de forma exitosa.

Dado que las startups continúan siendo emergentes y más aún en el área del desarrollo de software, consideramos necesario avanzar en el conocimiento sobre la relación de las startups con las empresas donde realizan el outsourcing, desde una perspectiva que profundice sobre las ceremonias, metodologías y procedimientos que se llevan a cabo al momento de establecer la comunicación entre ambas partes.

Luego de ahondar por las distintas metodologías de investigación, se ha optado por el estudio de caso como instrumento de investigación ya que el mismo se centra en examinar un fenómeno en su entorno natural, empleando múltiples métodos de recopilación de datos para reunir información de una o unas pocas entidades

(personas, grupos u organizaciones). Los límites del fenómeno no son claramente evidentes al principio de la investigación y no hay manipulación o control experimental (Benbasat *et al.* [4]).

Así mismo, se ha utilizado la entrevista como herramienta para la recolección de datos. Para las mismas, se han buscado informantes calificados que cumplieran un rol de expertos en empresas especializadas en desarrollo de software y venta de servicios para startups. Como paso inicial, se generaron unas primeras preguntas de investigación y se realizó una entrevista preliminar con uno de los entrevistados con el fin de ver cómo llevarla a cabo y redefinir las preguntas. Luego de dicha instancia, se contactaron el resto de los entrevistados y se aplicó el guión definitivo. Las entrevistas fueron de tipo semi estructuradas y se buscó que las preguntas fueran abiertas ya que las mismas no delimitan al entrevistado, permitiendo que el mismo se remita a vivencias y experiencias personales.

Las tareas de campo y relevamiento de datos se realizaron en el año 2020, en un período afectado por la pandemia del COVID 19 que impactó negativamente tanto en el desarrollo de la actividad empresarial como en la dinámica organizacional tanto de las universidades como de las empresas. Es por esto, que los entrevistados fueron contactados mediante videollamadas a través de la herramienta Zoom y se solicitó el consentimiento de los mismos para poder grabarlas con el fin de tener una recolección de datos más precisa. Para la transcripción de los datos, se optó por, primero recaudar la información de cada entrevista en particular según las preguntas referidas a los objetivos del proyecto para luego, poder generar conclusiones más generales.

El objetivo general de este estudio fue generar una propuesta de proceso de outsourcing basado en metodologías ágiles para las startups de software y analizar las estrategias de comunicación entre las mismas y las empresas donde tercerizan su código.

Las principales preguntas que orientaron la investigación fueron las siguientes:

- ¿Cuáles son los procesos que utilizan a la hora de desarrollar software para startups?
- ¿Utilizan actividades de metodologías ágiles o se caracterizan por la realización de actividades de Ingeniería de software tradicional?
- ¿Cuáles son las principales dificultades que enfrentan las empresas a la hora de desarrollar software para startups?

- ¿Cómo se desarrolla y con que estrategias llevan a cabo la comunicación con las startups?
- ¿Qué características debería tener una propuesta de proceso de lenguaje no técnico que sea compatible con los procesos de desarrollo de las startups?

Para responder estas preguntas se ha desarrollado este proyecto de investigación. La información que se presenta en este documento se ha organizado en 6 capítulos. Luego de esta introducción se presenta el capítulo 2 con el marco teórico, y posteriormente el capítulo 3 referido a la metodología de investigación. En el capítulo 4 se presentan los resultados del trabajo de campo y en el capítulo 5 se detalla la síntesis del proceso. Por último, en el capítulo 6 se incluyen las conclusiones y se discuten los resultados. Así mismo, se comentan las amenazas de la validez del estudio y se reflexiona sobre posibles líneas de investigación a futuro.

## 2 Marco teórico

Para la revisión de la literatura se analizaron múltiples fuentes y, antecedentes teóricos y empíricos referidos a las empresas cuyo servicio principal es el desarrollo de software. El análisis de los antecedentes teóricos y la revisión de bibliografía realizada se ha organizado en diferentes ejes conceptuales del marco teórico: definición de startup y startup de software, metodologías ágiles y la tercerización o outsourcing y sus problemas. Así mismo, también se realizó una breve revisión de la literatura sobre buenas prácticas.

### 2.1 Startup

La literatura académica acerca de las características que definen a una empresa como startup es muy amplia y presenta una variada gama de matices según los diferentes autores y fuentes consultadas.

Para Ries *et al.* [5], una startup es una “Institución diseñada para crear nuevos productos y servicios bajo extremas condiciones e incertidumbres”, mientras que para Paternoster [6], “las startups son un tipo de empresa que buscan tener como objetivo crear productos innovadores y de alta tecnología, crecen al expandir agresivamente su negocio en mercados altamente escalables”. Sin embargo, Gardino *et al.* [7] manifiestan que hasta no hace muchos años atrás el término startup podía ser asociado a diferentes significados, pero hoy en día el mismo, luego de ser adoptado por investigadores y profesionales: “un startup es una empresa pequeña que explora nuevas oportunidades de negocio, resolviendo problemas a los cuales no hay soluciones conocidas y el mercado es volátil”. Recalcan además, que el hecho de que una empresa haya sido fundada recientemente no la hace startup. Las dos características claves que diferencian a un startup de las empresas más establecidas son la alta incertidumbre y la rápida evolución.

En síntesis para este proyecto, se ha elaborado la siguiente definición de startup considerando los aportes mencionados.

Una startup es una empresa pequeña que explora nuevas oportunidades de negocio, resolviendo problemas a los cuales no hay soluciones conocidas y el mercado es volátil. Buscan tener como objetivo crear productos innovadores y de alta tecnología, crecen al expandir agresivamente su negocio en mercados altamente escalables.

## 2.2 Startup de software

Entre las diferentes fuentes consultadas, tanto Paternoster *et al.* [6] y Giardino *et al.* [7] coinciden en que las startups de software son empresas recientemente creadas sin historial operativo y rápidas en la creación de tecnologías de punta. Desarrollan software bajo altas condiciones de incertidumbre, abordando mercados de rápido crecimiento bajo una grave falta de recursos. Por lo tanto, presentan una combinación única de características que plantean varios desafíos a las actividades de desarrollo de software.

Así mismo, en “What do we know about software development in startups?” [7] se listan las siguientes características propias de este tipo de startups:

- Bajos recursos
- Innovación
- Incertidumbre
- Equipos pequeños
- Poca experiencia
- Dependencia en terceros

Las startup de software desarrollan software bajo altas condiciones de incertidumbre, abordando mercados de rápido crecimiento bajo una grave falta de recursos. Son caracterizadas por tener poco o nulo historial operativo, explorar los segmentos innovadores del mercado, tener un producto único y ser altamente reactivas, es decir, son capaces de reaccionar rápidamente a cambios en el mercado, tecnologías o productos.

## 2.3 Metodología ágil

Es el tipo de metodología más utilizado en los últimos tiempos principalmente en el desarrollo de software ya que la misma busca reducir el tiempo de desarrollo y entrega del mismo, en proyectos donde los requisitos del sistema suelen cambiar rápidamente durante el proceso de desarrollo. Según Sommerville [1], “su objetivo es entregar rápidamente un software que funcione a los clientes, quienes pueden proponer requisitos nuevos y modificaciones a ser incluidas en iteraciones posteriores del sistema”. Por otro lado recalca que, este tipo de metodologías apuntan a reducir la burocracia del proceso evitando trabajo de dudoso valor a largo plazo y eliminando la documentación que probablemente nunca se utilizará.

La filosofía de los métodos ágiles se refleja en el manifiesto ágil [8] el cual afirma los siguientes ítems:

- Individuos e interacciones sobre procesos y herramientas
- Software funcionando sobre documentación extensiva
- Colaboración con el cliente sobre negociación contractual
- Respuesta ante el cambio sobre seguir un plan

Basándose en dicho manifiesto, estos tipos de metodologías son afianzadas a los siguientes cinco principios:

### **Participación del cliente**

Los clientes deben participar ininterrumpidamente en todo el proceso de desarrollo. Su papel es proporcionar y priorizar los nuevos requisitos del sistema y evaluar las iteraciones del sistema.

### **Entrega incremental**

El software debe ser desarrollado de manera incremental, teniéndose en cuenta los requisitos especificados por el cliente en cada iteración-incremento.

### **Valorizar a las personas**

Las habilidades del equipo de desarrollo deben ser reconocidas y aprovechadas. Los miembros del mismo, deben poder desarrollar sus propias formas de trabajo.

### **Adoptar el cambio**

Hay que esperar que los requisitos del sistema cambien y diseñar el sistema para que pueda adaptarse a los mismos.

### **Mantener la simplicidad**

Centrarse en la simplicidad tanto en el software que se desarrolla como en el proceso de desarrollo. Siempre que sea posible, trabajar activamente para eliminar la complejidad del sistema.

## **2.4 Lean Startup**

Es una metodología que se basa en la validación de productos a través de la experiencia, evaluando si los mismos cubren o no las necesidades y expectativas de los clientes o usuarios finales. La misma hace énfasis en acortar los ciclos de desarrollo, lanzando distintas propuestas por un período de tiempo y obteniendo una retroalimentación continua con el fin de mejorar las distintas versiones del producto.

El libro *El método Lean Startup* de Ries [9] especifica los siguiente cinco principios como los más importantes dentro de esta metodología:

1. Un emprendedor es cualquier persona que trabaje en el ámbito de una startup independientemente del tamaño de la misma o sector.
2. El emprendimiento es gestión, una startup es una institución por lo que debe ser gestionada. Requiere nuevos métodos y herramientas especialmente adaptadas a las incertidumbres que la misma conlleva.
3. Aprendizaje validado, con la validación correcta, se pueden identificar y abordar los riesgos claves que están involucrados en el producto o idea y poder así realizar los ajustes necesarios.
4. Construir-Medir-Aprender, esto significa que el objetivo principal de la startup es transformar ideas en diferentes versiones de un producto, medir las métricas en función de los comentarios de los clientes y saber si es necesario realizar modificaciones o continuar igual.

5. Contabilidad innovadora, este principio responsabiliza a los fundadores o emprendedores por las acciones realizadas por ellos y su resultado a través de la descripción de hitos, la medición del progreso y la priorización del trabajo.

## 2.5 Lean Software Development

La metodología Lean Software Development es una síntesis de un sistema de prácticas, principios y filosofía para construir sistemas de software teniendo como base los principios Lean [10]. Poppendieck [11] expresa los siguientes siete principios como los principales contribuyentes de esta metodología.

- **Eliminar los desperdicios:** se debe tratar de evitar desperdicios cuanto más sea posible. Los más comunes son: codificaciones innecesarias, inicio de más tareas de las que pueden ser completadas, retrasos en el propio proceso de desarrollo, requisitos poco claros o sujetos a constantes cambios, comunicación lenta o ineficaz, defectos y problemas de calidad, cambios de tareas injustificados.
- **Aprender constantemente:** es la clave para dotar de mayor flexibilidad a los equipos al tiempo que se sientan las bases de la productividad a largo plazo. Algunas de las acciones que permiten generar este valor son: las revisiones y notas sobre los códigos, la creación de documentación, la elaboración de una base de datos informativa sujeta a actualizaciones y las puestas en común de conocimiento.
- **Retrasar las toma de decisiones:** las prácticas de desarrollo que prevén una toma de decisiones tardía son efectivas en dominios que implican incertidumbre, porque proporcionan un enfoque basado en las opciones. Retrasar las decisiones es valioso porque se pueden tomar mejores decisiones cuando se basan en hechos, no en especulaciones. En un mercado en evolución, mantener las opciones de diseño abiertas es más valioso que comprometerse antes de tiempo.
- **Entregar rápido:** tras un inicio tan temprano como sea posible el objetivo debe ser evitar la complejidad y simplificar al máximo. Esta metodología va en contra de los excesos (generalmente se cometen en términos de arquitectura y requerimientos de negocio).

- **Involucrar a todos:** de esta forma se consigue una mayor motivación que tiene un impacto directo en los niveles de productividad alcanzados. Este mayor rendimiento individual responde al reconocimiento y la autonomía, dos elementos que el responsable de la gestión de proyectos de lean software development debe procurar.
- **Construir integridad:** El software necesita un nivel adicional de integridad: debe mantener su utilidad a lo largo del tiempo. Normalmente se espera que el software evolucione a medida que se adapta al futuro. El software íntegro tiene una arquitectura coherente, tiene una puntuación alta en cuanto a usabilidad y adecuación al propósito, y es mantenible, adaptable y extensible. Investigaciones han demostrado que la integridad proviene del liderazgo, la experiencia pertinente, la comunicación eficaz y una sana disciplina.
- **Optimizar el conjunto:** se debe tratar de evitar que el foco de atención recaiga sobre un elemento en particular ya sean los plazos, los costes o los procesos. Cuando esto sucede la tendencia a compensar aumenta la carga en otras áreas que comienzan a perder ajuste. Conseguir aplicar este principio de forma correcta requiere de un modelo de liderazgo comprometido con la gestión ajustada y consciente de la necesidad de buscar un equilibrio.

Así mismo, se expresa que estos principios se utilizan para enmarcar el debate sobre las prácticas de desarrollo de productos que pueden ser apropiadas, dependiendo de la situación única de cada organización.

## 2.6 Scrum

Scrum es un framework para organizar y gestionar el trabajo. A diferencia de las metodologías tradicionales, no es un proceso estandarizado en el que se sigue metódicamente una serie de pasos secuenciales que garantizan la producción, a tiempo y dentro del presupuesto, de un producto de alta calidad que satisfaga a los clientes.

El mismo se basa en un conjunto de valores, principios y prácticas que proporcionan la base a la que una organización agregará su implementación única de las prácticas de ingeniería pertinentes y sus enfoques específicos para realizar las prácticas de Scrum. El resultado será entonces una versión de Scrum que es propia de la organización.

Rubin [12] para un mejor entendimiento de este marco, manifiesta análogamente: “es como los cimientos y las paredes de un edificio. Los valores, principios

y prácticas de Scrum serían los componentes estructurales claves. No se puede ignorar o cambiar fundamentalmente un valor, principio o práctica sin correr el riesgo de colapso. Lo que se puede hacer, sin embargo, es personalizar dentro de la estructura de Scrum, añadiendo accesorios y características hasta tener un proceso que funcione para nuestras condiciones”.

## 2.7 Outsourcing

El término outsourcing refiere a la actividad de derivar el desarrollo de software a terceros en lugar de desarrollarlo internamente dentro de una empresa.

Los modelos de tercerización más conocidos son multisourcing, microsourcing y crowdsourcing.

En el multisourcing se utiliza a un único proveedor para todos los proyectos. Esto puede llevar a un presupuesto elevado y poco entusiasmo por falta de competencia. En el microsourcing se contrata a freelancers. En este caso, un aspecto en contra es que a veces los proyectos comienzan siendo pequeños y luego escalan, por lo que pueden sobrepasar las capacidades del freelancer. En el crowdsourcing lo que se terceriza son tareas más pequeñas, generalmente las que, si bien pueden ser realizadas fácilmente por una persona, aún no pueden ser automatizadas.

Krym [13] manifiesta que uno de los puntos clave para obtener buenos resultados al aplicar esta técnica de trabajo es definir objetivos bajo el acrónimo SMART (Specific, Measurable, Actionable, Result-Focused, Time-bound).

Así mismo, afirma que, tercerizar a veces conlleva enfrentar riesgos, los cuales se caracterizan en: riesgos externos, internos y personales.

Los riesgos externos son aquellos asociados a factores que están por fuera de la organización (inestabilidad geopolítica, seguridad, privacidad y pérdida de propiedad intelectual, capacidades inadecuadas de los proveedores, incumplimiento de las responsabilidades, etc.). Los riesgos internos son aquellos asociados a factores que si se encuentran dentro de la organización. Los más comunes son: falsas expectativas, falta de preparación frente a cambios organizacionales, impacto negativo en el personal. Por último, los riesgos personales son aquellos más subjetivos, relacionados a la carrera, estilo de vida o reputación de cada persona que forma parte del equipo que va a tercerizar.

El autor asegura que una vez que se decide si un proyecto será tercerizado,

se debe especificar un plan de alto nivel en el cual los aspectos más importantes son: definir el alcance general del trabajo a tercerizar y dividir dicho trabajo en componentes concretos.

En el libro *CMMI for Acquisition Guidelines for Improving the Acquisition of Products and Services* [14] se sostiene que hay estudios que afirman que, en la mayoría de los casos, los beneficios de tercerizar no se asemejan a las expectativas iniciales. Las principales razones por las cuales las empresas deciden tercerizar son la falta de conocimientos técnicos internos y la capacidad de transferir los riesgos al proveedor. Además de tener en cuenta la reducción de costos, flexibilidad, aplicación de mejores prácticas e innovación. No existe una sola razón por la cual dichas expectativas no se asemejan, sino que son varias. Una de ellas es atribuida a los procesos de contratación. En las contrataciones con terceros, es importante tener en cuenta que los ciclos de vida de los contratos pasan por distintas etapas, como, por ejemplo, la solicitud, evaluación de la respuesta del proveedor, negociación y finalización. La comunicación tiene un rol importante a lo largo de estos ciclos de vida. Generalmente sucede que la reducción en los costos no es tan grande como se espera debido a que existen costos “ocultos” asociados a servicios que no fueron considerados apropiadamente a la hora de escribir el contrato.

El artículo *Exploring the outsourcing relationship in software startups – A multiple case study* [15] por su parte, expresa que las startups generalmente al tener pocos recursos optan por tercerizar las actividades de ingeniería: el prototipado, diseño, desarrollo, etc. Además, al ser empresas emergentes, prefieren subcontratar a desarrolladores con experiencia. Así mismo, existe una minoría que también terceriza las tareas del área de marketing y finanzas. Se considera que para las startups es positivo tercerizar no solo por los beneficios de minimizar costos, sino que se genera una proximidad con el mercado, se mejora el trabajo en equipo y la realización de procesos, aspectos en los cuales generalmente al no tener experiencia operativa son nulos. Generalmente se evalúa si la tercerización tuvo buenos resultados o no desde una perspectiva técnica, según: la calidad del código, la comunicación y la satisfacción del cliente.

En algunos casos particulares, luego de que se termina el proyecto, el equipo de tercerización puede quedar ligado a la startup, ya sea con prioridad para futuras tercerizaciones o como en una de las startups mencionadas en este artículo, donde se cedieron parte de las acciones de esta al team leader del equipo de tercerización.

## 2.8 Buenas prácticas

Kristjansson *et al.* [16] manifiestan que, recurrir al outsourcing como herramienta para reducir costos es una alternativa atractiva, sin embargo, existen varias barreras que hay que superar para realmente conseguir que los mismos sean los más bajos. Uno de estos obstáculos es el intercambio adecuado de conocimientos. Estos autores, plantean un proceso a partir de una investigación basada en empresas que proveen servicios de nearshore outsourcing. El modelo reconoce que es necesario intercambiar conocimientos en las distintas fases de un proyecto de desarrollo de software y no sólo en una única instancia al inicio del mismo. Se establecen tres fases primarias las cuales se describen a continuación.

### Inicio del proyecto

La primera fase del modelo marca el inicio de la colaboración, la misma propone un alineamiento de conocimientos globales respecto al proyecto. Así mismo, en esta instancia, deben establecerse y ajustarse los medios de comunicación a ser utilizados.

### Desarrollo continuo

Siendo esta la fase principal del modelo, se establecen cinco prácticas/focos:

- Colaboración funcional: comunicación y colaboración continua diaria sobre las funcionalidades deseadas y especificaciones.
- Specification grooming: se enfoca en los requerimientos funcionales con especial énfasis en la documentación de los mismos. De igual manera, se centra en las alteraciones realizadas a los requerimientos.
- Asistencia técnica: consiste en que un experto técnico por parte del cliente ayuda al equipo desarrollador en cuestiones emergentes como el diseño técnico, integraciones o la arquitectura.
- Documentación técnica: hace hincapié en escribir lo que se sabe sobre la implementación técnica.
- Product Knowledge Center: vincula y apoya los cuatro focos mencionados anteriormente.

## Entrega del equipo

Esta última fase hace referencia a la transición en la que el proyecto pasa a manos de otro equipo. Se enfatiza en la necesidad de transmitir el máximo conocimiento necesario a este nuevo equipo.

## 2.9 Trabajo previo

### 2.9.1 Antecedentes nacionales

Respecto a los antecedentes de investigación a nivel nacional, se han identificado diferentes estudios previos que aportan conocimientos, resultados y hallazgos que sirvieron de apoyo y complemento en este proyecto. En todos los casos, estos antecedentes fueron elaborados en el marco de los proyectos de fin de carrera de la Universidad ORT Uruguay para las carreras Licenciatura en Sistemas e Ingeniería en Sistemas.

En primer lugar, se ha identificado el trabajo *Habilidades técnicas y blandas en las startups de software y su evolución* por Buffa y Febles [17]. En el mismo, se destaca que el outsourcing aparece como una consecuencia del crecimiento continuo.

Según las evidencias recolectadas en este estudio realizado en el año 2019, las tres formas comunes de incorporar conocimiento son las de realizar cursos, contratar personal y subcontratar empresas. Según varios de los entrevistados en dicha tesis, el outsourcing aparece en una primera instancia, luego de darse cuenta de que había mucho conocimiento que no tenían y que les era necesario. Debido al corto tiempo tenían que concentrarse en otras tareas y no podían adquirirlo ellos mismos por lo que decidieron contratar a terceros.

Otro de los estudios revisados fue el trabajo de Nieto y González, *El proceso de creación y evolución del Producto Mínimo Viable en las startups de software* [18]. En este documento se destaca que, si bien cuatro de las startups participantes del trabajo contrataron un equipo externo, las tareas a realizar por dichos equipos no pertenecían a una una misma área, sino más bien, fueron muy variadas, desde el desarrollo de un MVP hasta solamente la parte mobile de una aplicación. Una de estas startups manifestó que aplicar la estrategia de outsourcing no dio resultado ya que les fue mal con todas las empresas con las que trabajaron (una de las causas

principales fue que tuvieron problemas con las entregas, siempre demoradas). Otra de las startups participante contrató un servicio de encuestas tercerizadas como herramienta para recolectar información sobre qué mejoras realizar en el MVP.

Así mismo, el estudio realizado por Perez y Peralta en *Inicio y evolución de las prácticas y actividades de ingeniería en las startups de software* [19] aporta conocimiento sobre la etapa de incorporación del conocimiento para el desarrollo de la empresa. En este trabajo se halló que los momentos en los cuales las startups decidieron contratar un equipo externo difieren mucho y dependen de múltiples factores. Por ejemplo, una de las empresas relevadas decidió desde un principio, focalizarse en la versión web y tercerizar la versión mobile una vez que se hayan conseguido fondos para poder hacerlo. En otro caso, la idea de recurrir al outsourcing apareció luego de intentar y tener dificultades con el manejo de mapas de geolocalización. Por último, para otra empresa que participó de este estudio la idea apareció luego de que el equipo fundador comenzara a realizar pruebas, y luego de un período corto de tiempo se dieran cuenta que les llevaría una cantidad determinada de tiempo con la cual no contaban.

Por último, en los trabajos de Galperin y González [2] y de Bentos y Pera [3] se destaca el outsourcing como herramienta para tercerizar no solo desarrollo de software, sino que también, tareas relacionadas a la documentación de los proyectos, como por ejemplo un informe con los aspectos que no funcionan como deberían para tener un registro de los mismos y tenerlos en cuenta.

En estos antecedentes se afirma que el outsourcing no solo es una herramienta para incorporar conocimiento, sino que también, es de gran ayuda para poder delegar un conjunto de tareas y poder así invertir el tiempo en otras, recurso generalmente escaso. Así mismo, en el estudio de Bentos-Pera se menciona como problema recurrente entre los entrevistados la complejidad que tienen las empresas en la integración de sus sistemas con los servicios de outsourcing.

## 2.9.2 Antecedentes internacionales

Abrahamsson y Duc [20] manifiestan que el outsourcing es considerado un beneficio para las empresas pequeñas tanto de forma directa como indirecta, ya sea en: reducción de costos, proximidad al mercado, improvisación en el trabajo en equipo y en los procesos. Por otro lado, la colaboración y coordinación son reconocidos como factores de éxito. Para estos autores, las prácticas de gestión de la coordinación, comunicación, la confianza y el uso eficaz de los recursos son aspectos importantes en la gerencia de startups. En la investigación que presentan se analiza un estudio realizado en Europa que demostró que los problemas principales

para establecer confianza son los siguientes: la poca socialización y las diferencias socioculturales, inconsistencia y disparidades en las prácticas laborales, la poca comunicación y la falta de reuniones cara a cara, habilidades lingüísticas y poca experiencia con el manejo de problemas.

Las startups entrevistadas en este artículo afirman que a medida que su negocio crecía, aprendieron acerca de la capacidad de los socios subcontratados para prestar servicios tanto en cantidad como en calidad. Una de las startups entrevistadas expresó una mala experiencia al elegir un servicio freelance barato anunciado en un portal de empleos en línea, ya que el CEO manifestó sus dudas sobre la experiencia y el historial del socio. El diseñador era bueno, pero considero que los desarrolladores no tanto ya que no hicieron tests con ambas plataformas, iPhone y Android.

Establecer una coordinación y comunicación fluida, tal como se hace constar en este estudio, es muy difícil. Debido a que se intenta reducir los costos, la mayoría de las personas subcontratadas se localizan en diferentes entornos geográficos, temporales y culturales. A menudo es un desafío establecer una relación de subcontratación con una persona encontrada en un portal de freelance. Por otro lado, también es un desafío establecer buenas prácticas de desarrollo remoto.

Verner *et al.* [21], en el artículo *Establishing and maintaining trust in software outsourcing relationships: An empirical investigation* afirman que trabajar con empresas con diferencias socioculturales no es fácil, no sólo por dichos factores, sino que, cada empresa tiene su forma propia de desempeñar el trabajo, por lo que generalmente surgen problemas. Por otro lado, se afirma que la confianza brindada por la empresa que terceriza ha sido identificada como un factor importante para poder tener éxito en la colaboración, las alianzas estratégicas y las redes de empresas.

Por último, en el artículo *Technology innovation and risks in outsourcing* [22] se hace referencia a que las literaturas académicas han acordado que las empresas pueden obtener los beneficios esperados del outsourcing, como la reducción de los costos operacionales, la capacidad de transformar los costos fijos en costos variables, la capacidad de centrarse en las competencias básicas y el acceso a las competencias y los conocimientos especializados externos más importantes de la industria. Sin embargo, también hay algunas literaturas que resaltan las debilidades y los riesgos asociados al outsourcing en gran escala. El riesgo de depender de un solo proveedor y de los costos ocultos existentes del outsourcing, la posibilidad de una pérdida de tecnología vital con respecto a las competencias básicas en materia de contratación externa y el problema de la selección del proveedor/proveedor de servicios más adecuado y apto para el rápido cambio de tecnología.

En conclusión, las razones o dificultades que conducen a la tercerización son generalmente la falta de tiempo y/o conocimiento suficiente para llevar a cabo determinadas tareas. A su vez, una de las startups entrevistadas remarca que se intentó contratar la menor cantidad de personas posibles ya que eso conlleva tiempo extra de gestión, además de dependencias legales; en este caso, el prototipo base fue lo tercerizado. La gran mayoría de las startups no tuvieron buenas experiencias en la tercerización. Por ejemplo, la que tercerizó las pruebas dijo que las mismas quedaron inconclusas. Por otro lado, otra de las startups afirma que siempre se obtuvieron resultados tardíos. Uno de los autores afirma que la mala experiencia con la tercerización se debe a que no existió un buen monitoreo del proceso de esta. Los autores concluyen que varios de los entrevistados “destacan la necesidad de contratar o tercerizar personal para incorporar conocimiento cuando se llega un punto en el que el fundador no puede incorporarlo porque tiene que enfocarse en otras tareas, o porque no es viable por el esfuerzo o tiempo necesarios (conocimiento financiero, contable o legal)”. También destacan que acudir a terceros forma parte de incorporar conocimiento.

## 3 Metodología de investigación

En este apartado se describen en primer lugar las preguntas de investigación y los objetivos específicos y posteriormente se desarrolla el enfoque metodológico y el instrumento de investigación.

### 3.1 Objetivos

#### 3.1.1 Objetivos generales

Generar una propuesta de proceso basado en metodologías ágiles para el outsourcing de startups de software.

#### 3.1.2 Objetivos específicos

1. Identificar y clasificar los criterios que se utilizan para realizar outsourcing.
2. Determinar la estrategia de comunicación predominante entre la startup y el equipo de outsourcing.
3. Conocer las áreas y actividades de desarrollo dónde se realiza el outsourcing.
4. Conocer las actividades de metodologías ágiles que se aplican con empresas de outsourcing.
5. Conocer las actividades específicas de ingeniería de software tradicional que se aplican con empresas de outsourcing.
6. Generar una propuesta de proceso que sea compatible con los procesos de las startups.
7. Generar una propuesta de proceso de lenguaje no técnico.

## 3.2 Preguntas de investigación

A continuación se listan las preguntas para la etapa de trabajo de campo.

1. ¿Se especializan en el desarrollo tercerizado o desarrollan productos propios?
2. ¿Se centran en un tipo específico de desarrollo, como por ejemplo MVP o testing?
3. ¿Cuáles son los procesos que utilizan a la hora de desarrollar software para startups?
4. ¿De qué manera se especifican las tareas a realizar?
5. ¿Dichas tareas están formalizadas en algún documento?
6. ¿Cuáles son las principales dificultades que enfrentan a la hora de desarrollar software para startups?
7. ¿Utilizan algún proceso para llevar a cabo la comunicación con las startups?
8. ¿Utilizan actividades de metodologías ágiles?
9. ¿Utilizan actividades de Ingeniería de software tradicional?

## 3.3 Enfoque

Considerando los aportes de Hernández Sampieri [23], se ha optado por un enfoque cualitativo ya que el mismo se orienta al análisis de fenómenos sociales y experiencias de vida. Desde esta perspectiva, se busca comprender el significado que las personas tienen acerca del objeto de estudio, siendo entonces un enfoque comprensivo.

### 3.3.1 Estudio de caso

Runeson *et al.* [24] manifiestan que el estudio de caso es una estrategia de investigación utilizada no solo para aumentar el conocimiento, ya sea sobre individuos, grupos y organizaciones, fenómenos sociales, políticos, etc., sino que también para

generar cambios sobre los fenómenos que están siendo estudiados. La investigación en ingeniería de software tiene, en alto nivel, objetivos similares. Es decir, comprender mejor cómo y por qué debe realizarse la ingeniería de software y, con este conocimiento, mejorar los procesos de ingeniería y los productos de resultantes.

Hay diferentes taxonomías utilizadas para clasificar la investigación en ingeniería de software. El término estudio de caso se utiliza en paralelo con términos como estudio de campo y observación. Cada uno de ellos se centra en un aspecto particular de la metodología de investigación.

Yin [25] define el estudio de caso como una investigación empírica que estudia un fenómeno contemporáneo dentro de su contexto, especialmente cuando los límites entre el fenómeno y el contexto no son claramente evidentes.

Por otro lado, Benbasat *et al.* [4] lo definen como: un estudio de caso examina un fenómeno en su entorno natural, empleando múltiples métodos de recopilación de datos para reunir información de una o unas pocas entidades (personas, grupos u organizaciones). Los límites del fenómeno no son claramente evidentes al principio de la investigación y no hay manipulación o control experimental.

Una de las directrices preliminares para la investigación empírica en la ingeniería de software afirma que se debe especificar tanto como sea posible el contexto. En particular, definir claramente las entidades, atributos y medidas que están captando la información contextual.

Los estudios de casos ofrecen un enfoque que no requiere un límite estricto entre el objeto de estudio y su entorno. Los estudios de casos proporcionan una comprensión más profunda de los fenómenos que están siendo estudiados respecto a otras metodologías de investigación .

El estudio de caso en la ingeniería de software es una investigación empírica que se basa en múltiples fuentes de pruebas para investigar un caso (o un pequeño número de casos) de un fenómeno contemporáneo de ingeniería de software dentro de su contexto, especialmente cuando no se puede especificar claramente el límite entre el fenómeno y el contexto.

El libro *Case Study Research in Software Engineering: Guidelines and Examples* [24] afirma que existen tres estrategias de investigación importantes relacionadas con los estudios de caso: encuesta, experimento e investigación de acción.

La encuesta, es la recolección de información estandarizada de una población específica generalmente mediante entrevistas o cuestionarios. El experimento o experimento controlado se caracteriza por medir los efectos de manipular una va-

riable sobre otra variable y que los sujetos se asignan a los tratamientos al azar. Por último, la investigación de acción con su propósito de influenciar o cambiar algún aspecto del centro de la investigación está sumamente relacionado con el estudio de casos. En algunas definiciones, el estudio de caso es puramente observacional mientras que la investigación de acción se centra en el proceso de cambio.

Según Robson [26] se distinguen cuatro tipos generales de propósitos de investigación:

### **Exploratorio**

Averiguar lo que está sucediendo, buscando nuevos conocimientos, y generando ideas e hipótesis para nuevas investigaciones.

### **Descriptivo**

Describir el estado actual de una situación o fenómeno.

### **Explicativo**

Generar una explicación para una situación o un problema.

### **Mejorando**

Tratar de mejorar cierto aspecto del fenómeno estudiado.

Para la realización de este trabajo, se ha optado entonces, por el tipo exploratorio.

### **Proceso**

Para llevar a cabo el caso de estudio existen cinco grandes pasos a considerar:

1. Diseño del estudio de caso: se definen los objetivos y el estudio de caso es planificado.
2. Preparación para la recolección de datos: se definen procedimientos y protocolos de recolección de datos.
3. Recolección de datos: se ejecutan los procedimientos previamente definidos.
4. Análisis de los datos recolectados: se aplican procedimientos de análisis de datos.
5. Reporte: el estudio y sus conclusiones son presentados en un informe.

### 3.3.2 La entrevista como herramienta

Se ha utilizado como técnica de recolección de datos la entrevista. Esta herramienta, según Hernández Sampieri [23] permite realizar la recaudación cuando el problema de estudio no se puede observar o es muy difícil hacerlo por ética o complejidad. Asimismo, indica que las mismas cumplen con las siguientes características:

1. El principio y final de la entrevista se definen con claridad. De hecho, tal definición se integra en el cuestionario.
2. El mismo instrumento se aplica a todos los participantes, en condiciones lo más similares posible (estandarización).
3. Se busca que sea individual, sin la intrusión de otras personas que pueden opinar o alterar de alguna manera la entrevista.
4. Es poco a nada anecdótica (aunque en algunos casos es recomendable que el entrevistador anote cuestiones fuera de lo común como ciertas reacciones y negativas a responder).
5. La mayoría de las preguntas suelen ser cerradas, con mínimos elementos rebatibles, ampliaciones y sondeos.
6. El entrevistador y el propio cuestionario controlan el ritmo y la dirección de la entrevista.
7. El contexto social no es un elemento a considerar, lo es solamente el ambiental.
8. El entrevistador procura que su patrón de comunicación sea similar (su lenguaje, instrucciones, etcétera).

Se ha buscado que las preguntas fueran abiertas ya que las mismas no delimitan al entrevistado, permitiendo que el mismo se remita a experiencias personales y verbalice pensamientos brindando de esta forma respuestas más amplias.

En este caso, las entrevistas fueron realizadas mediante videollamadas a través de la herramienta Zoom. Las mismas fueron del tipo semi-estructuradas, teniendo como base el guión descrito en el apartado 3.3.4 de este documento. En los momentos previos a la entrevista se solicitó el consentimiento de los entrevistados para poder grabarlas con el fin de tener una recolección de datos más fiable y en correspondencia con los procedimientos éticos de la investigación.

### 3.3.3 Muestra

Para poder definir el perfil de los integrantes de la muestra se ha buscado entrevistar informantes calificados que cumplieran un rol de expertos en empresas especializadas en desarrollo de software y venta de servicios para startups. Así mismo, se ha buscado que dichas empresas no solo se encuentren dentro del país, sino que también, tuvieran conocimiento sobre los proyectos de fin de carrera realizados dentro de la Universidad ORT Uruguay.

Las empresas entrevistadas tienen la particularidad de contar con varios años de experiencia en el mercado, además de clasificar como pequeñas/medianas empresas. Todas realizan la comunicación interna en español, mientras que, la comunicación con los clientes es dada en Inglés debido a que la gran parte de los mismos residen en Estados Unidos.

Se generaron unas primeras preguntas de investigación y se realizó una entrevista preliminar con uno de los entrevistados con el fin de ver cómo llevarla a cabo y redefinir las preguntas. Luego se contactaron el resto de los entrevistados y se aplicó el guión definitivo. Dado que las entrevistas fueron de tipo semi estructurada, algunos aspectos no fueron contestados explícitamente por lo que se volvieron a contactar a los entrevistados vía mail para recabar sus respuestas.

Luego de haber elaborado el guión definitivo se realizaron cuatro entrevistas en profundidad.

### 3.3.4 Guión de entrevista

A continuación se detalla el guión estándar de entrevista elaborado teniendo en cuenta los objetivos del proyecto.

- Fecha de realización:
- Duración:
- Empresa:
- Entrevistado:
- Modalidad de la entrevista:
- Tipo de entrevista:

**Contextualización de la entrevista. Preguntas generales:**

1. ¿En qué año fue fundada la empresa?

---

2. ¿Desde qué año desarrollan software para startups?

---

**Preguntas referidas a los objetivos del proyecto:**

1. ¿Se especializan en el desarrollo tercerizado o desarrollan productos propios?

---

2. ¿Se centran en un tipo específico de desarrollo, como por ejemplo MVP o testing?

---

3. ¿Cuáles son los procesos que utilizan a la hora de desarrollar software para startups?

---

4. ¿Se especializan en el desarrollo tercerizado o desarrollan productos propios?

---

5. ¿Utilizan algún proceso para llevar a cabo la comunicación con las startups?

---

6. ¿De qué manera se especifican las tareas a realizar?

---

7. ¿Dichas tareas están formalizadas en algún documento?

---

8. ¿Utilizan actividades de metodologías ágiles?

---

9. ¿Utilizan actividades de Ingeniería de software tradicional?

---

La relación entre preguntas y objetivos se detallan en la siguiente tabla.

Objetivos específicos	Pregunta
1- Identificar y clasificar los criterios que se utilizan para realizar outsourcing.	<p>¿Se especializan en el desarrollo tercerizado o desarrollan productos propios?</p> <p>¿Cuáles son los procesos que utilizan a la hora de desarrollar software para startups?</p> <p>¿Se centran en un tipo específico de desarrollo, como por ejemplo MVP o testing?</p>
2 - Determinar la estrategia de comunicación predominante entre la startup y el equipo de outsourcing.	<p>¿Utilizan algún proceso para llevar a cabo la comunicación con las startups?</p> <p>¿De qué manera se especifican las tareas a realizar?</p> <p>¿Dichas tareas están formalizadas en algún documento?</p>
3 - Conocer las áreas y actividades de desarrollo dónde se realiza el outsourcing.	¿Cuáles son las principales dificultades que enfrentan a la hora de desarrollar software para startups?
4 - Conocer las actividades de metodologías ágiles que se aplican con empresas de outsourcing.	¿Utilizan actividades de metodologías ágiles?

Tabla 3.1: Relación entre preguntas y objetivos de investigación

## 4 Resultados del trabajo de campo

### 4.1 Realización de entrevistas

Debido a la emergencia sanitaria declarada en el país a causa del virus COVID-19, las entrevistas no pudieron ser realizadas de forma presencial. Los entrevistados debieron ser contactados en una primera instancia vía mail dándose a conocer el propósito del proyecto con el fin de poder coordinar un encuentro para realizar las preguntas.

Una vez coordinado dicho encuentro, se dieron a conocer las preguntas a ser realizadas para quienes lo habían solicitado.

A continuación se detalla una breve descripción de cada una de las empresas entrevistadas.

Empresa	Descripción
Vairix	La empresa Vairix surgió en 2010 ofreciendo servicios tanto de desarrollos de software completos como de staff augmentation. Se centra en la ciudad de Montevideo y el total de sus clientes radica en Estados Unidos.
Kreitech	Se especializan en el desarrollo de software web y mobile enfocado en IoT, surgió luego de un intento fallido por fundar una startup y de ser contratados en varias instancias como freelancers por parte de empresas donde ya habían trabajado. Se estableció como una empresa que se especializa en el outsourcing donde sus principales clientes se encuentran también en Estados Unidos.
Nimacloud	La empresa Nimacloud fundada en 2013 en sus principios tenía como idea eventualmente poder generar un producto propio, pero si bien se realizaron lluvias de ideas y se contaba con personal capacitado, se priorizaron los proyectos demandados por los clientes, es entonces que decidieron dedicarse a la consultoría, específicamente de la plataforma Salesforce. En este caso, los clientes son tanto de Estados Unidos como de Uruguay.
Moove-it	Fundada en 2005, siendo esta la empresa con mayor trayectoria, ha variado sus servicios a lo largo de los años. En sus inicios generaba productos y ofrecía servicios de consultoría, pero luego de que los clientes basados en Estados Unidos comenzaron a multiplicarse, hoy en día se brindan distintos servicios. Por un lado, el desarrollo de aplicaciones web y mobile, transformación digital, que se basa en el asesoramiento para incorporación digital dentro de la empresa, product recovery donde se trabaja con startups que se encuentran con una idea semi validada o validada generando descubrimiento del producto, y servicio de staff augmentation.

Tabla 4.1: Descripción de empresas entrevistadas

En la siguiente tabla se especifica para cada empresa los datos de la entrevista realizada: duración, fecha y modalidad.

Empresa	Duración	Fecha de realización	Modalidad
Vairix	21 minutos	07/08/2020	videollamada
Kreitech	26 minutos	24/08/2020	videollamada
Nimacloud	19 minutos	03/09/2020	videollamada
Moove-it	17 minutos	22/09/2020	videollamada

Tabla 4.2: Información de entrevistas realizadas

## 4.2 Síntesis de resultados

A continuación se realiza un agrupamiento de las preguntas en función de las categorías de análisis.

Categoría	Preguntas
Proceso de outsourcing	1- ¿Cuáles son los procesos que utilizan a la hora de desarrollar software para startups?  2 - ¿Se centran en un tipo específico de desarrollo, como por ejemplo MVP o testing?  3 - ¿Se especializan en el desarrollo tercerizado o desarrollan productos propios?
Metodologías ágiles	6 - ¿De qué manera se especifican las tareas a realizar?  7 - ¿Dichas tareas están formalizadas en algún documento?  8 - ¿Utilizan actividades de metodologías ágiles?
Comunicación	5 - ¿Utilizan algún proceso para llevar a cabo la comunicación con las startups?
Dificultades	4 - ¿Cuáles son las principales dificultades que enfrentan a la hora de desarrollar software para startups?

Tabla 4.3: Categorización de preguntas del guión de entrevista

### 4.2.1 Preguntas sobre el proceso de outsourcing

Dentro de las empresas participantes de este estudio existe una gran variedad en lo que refiere a los servicios que brindan y los procesos que utilizan para ofrecer los mismos. De igual manera, el tipo de desarrollo que proporcionan es diverso y depende de la demanda de los diferentes clientes.

### **1- ¿Cuáles son los procesos que utilizan a la hora de desarrollar software para startups?**

La empresa 2 indicó que a lo largo de los años han trabajado sobre un proceso que si bien sufre modificaciones según el proyecto en el que se esté trabajando, está basado en la metodología Scrum. Por otro lado, la empresa 3 señaló que tienen un proceso con tres grandes instancias, la primera donde se realiza el acuerdo con el cliente sobre cuál es el sistema a desarrollar y los costos, tanto de tiempo como económicos. La segunda, son las reiteradas daily meetings que se realizan con el fin de tener un chequeo constante y diario de lo que se va haciendo y cuáles han sido las dificultades, y por último la realización de una sprint demo para realizar validaciones con el cliente.

### **2 - ¿Se centran en un tipo específico de desarrollo, como por ejemplo MVP o testing?**

La empresa 3 señaló que las propuestas que reciben son muy variadas en cuanto al tipo de desarrollo, por otro lado, las empresas 1, 2 y 4 destacan que si bien en un principio si trabajaban en el desarrollo de MVPs con el fin de conseguir inversionistas, hay una tendencia a que los proyectos que reciben ya vienen con MVPs probados y su objetivo es trabajar sobre estos. Uno de los entrevistados indicó al respecto: “Antes, si pasaba que los proyectos que llegaban eran MVPs, pero hoy en día son primeros productos con espalda. Los proyectos que nos llegan están con un inversionista fuerte queriendo sacar una primera versión del producto pero todavía no es un producto 100 por ciento desarrollado... Si tuviera que ponerlo en porcentaje sería capaz un treinta por ciento de esos y el otro setenta ya viene con espalda”.

### **3 - ¿Se especializan en el desarrollo tercerizado o desarrollan productos propios?**

En base a las respuestas de los entrevistados podemos observar que, si bien cada una de las empresas ofrecen diferentes servicios, todas brindan servicios de outsourcing en distintas modalidades. La Empresa 1 indicó que trabajan con dos grandes ramas, el desarrollo de sistemas completos y staff augmentation. La empresa 2 informó que luego de un intento fallido de establecerse como startup, esperado por la definición propia de startup (ver 2.1), y de ser contratados en varias instancias como freelancers por parte de empresas donde ya habían trabajado, decidieron establecerse como empresa que se especializa en el outsourcing. Asimismo, como la empresa 1, la empresa 4 brinda el servicio de staff augmentation además de otros servicios como transformación digital y product recovery. Por último, la Empresa 3 se distingue de las anteriores ya que se especializa solamente en el outsourcing en

una tecnología en particular, el entrevistado, al respecto expresó: “Siempre la idea fue eventualmente tener un producto y en un momento trabajamos para empezar a hacer brainstorming para ver qué tipo de producto podíamos hacer porque tenemos gente de calidad, etcétera, pero nunca llegamos a nada porque los proyectos siempre fueron más prioritarios.”

## 4.2.2 Preguntas sobre metodologías ágiles

Si bien las metodologías ágiles son conocidas como las más apropiadas para desarrollar software, a la hora de ponerlas en práctica en empresas con grandes trayectorias, puede resultar costoso y poco atractivo cambiar el desarrollo de productos tradicional, con un enfoque lineal por técnicas ágiles e iterativas. Del mismo modo, es habitual que cuando estas metodologías son aplicadas, sean adaptadas según las necesidades de la empresa.

### 6 - ¿De qué manera se especifican las tareas a realizar?

Las tareas son especificadas en diferentes modalidades según la empresa y el proyecto. La empresa 2 señaló que generalmente se trabaja sobre un tablero compartido en la herramienta Trello donde se van colocando notas con cada tarea. Por su parte, la empresa 3 expresó respecto a las tareas: “Lo vamos traqueando en la herramienta que usemos, generalmente Jira.” De igual manera, indicó que, como indica SCRUM, luego de asignar a cada sprint su velocidad, se ponderan las tareas con story points.

### 7 - ¿Dichas tareas están formalizadas en algún documento?

Todas las empresas expresaron que el generar un documento formal depende cien por ciento de la demanda del cliente. A su vez, indicaron que la mayor parte de las veces, este no es generado debido a la naturaleza de las startups donde hay una tendencia muy grande a cambiar los requerimientos en el transcurso del tiempo. La empresa 2 sobre esto señaló: “Es complejo tener una documentación actualizada cuando se trabaja con startups porque pivotan, prueban en el mercado y si no les funcionan cambian sus funcionalidades, así reiteradas veces”. A su vez, esta misma empresa, destacó que por lo general, con las startups se establece una relación de confianza con la intención de seguir trabajando en conjunto quitando entonces la necesidad de tener un documento bien formalizado. Por otro lado, la empresa 1 manifestó: “Depende mucho del proyecto, cuando se acuerdan los entregables es ahí donde se acuerda si simplemente se quiere el código en una cuenta de GitHub, el código y documentación de uso, en el caso de que sea un proyecto de backend el código y la documentación de la API para poder consumirla, etc.” También

resaltó que, al igual que la empresa 3, normalmente mucha información queda en las herramientas utilizadas para definir las tareas (Jira y Trello) actuando como una especie de documentación más informal. Por último, la empresa 2 señaló que en algunas instancias se han producido documentos específicos sobre las reglas de negocio o arquitectura del sistema a desarrollar pero eso se ha dado en casos muy puntuales.

### **8 - ¿Utilizan actividades de metodologías ágiles?**

En cada una de las empresas se aspira a utilizar diferentes metodologías ágiles, las mismas son utilizadas como base ya que son modificadas según las necesidades de la empresa. La empresa 1 señaló que utilizan como base Scrum, pero no en todos los proyectos pueden trabajar de esa forma debido a que hay algunos donde el cliente tiene muy definido lo que quiere hacer y se pierde la esencia que tiene esta metodología de ir descubriendo el producto a medida que transcurre el tiempo. También mencionó el caso en el que los proyectos son demasiado cortos, de dos o tres meses, donde no se puede madurar un concepto de velocidad o velocidad media. Además agregó: “Sí se trabaja con iteraciones, si hay daily, si hay planning, si hay retrospectiva. Pero el manejo del tiempo y los costos muchas veces se dan antes de que exista el proyecto, se intenta hacer una estimación y se da un presupuesto normalmente de un precio cerrado y se trabaja en esas condiciones”. La empresa 2 al igual que la empresa 1 utiliza Scrum. Indicó que una vez que la relación comercial comienza y se pasa a una etapa de desarrollo se define un product backlog así como también sprints. La empresa 3 señaló que se utilizan instancias donde se dialoga con el cliente para obtener feedback y stand-up meetings para mantener al equipo actualizado. Por último, la empresa 4 indicó que según el proyecto utilizan Kanban, Scrum o XP todas con sus debidas modificaciones según sea necesario.

## **4.2.3 Preguntas sobre la comunicación**

Un aspecto adicional que ha resultado de interés investigar es el lugar de la comunicación como factor de desarrollo de las empresas. En la sección se analizan las respuestas de los participantes relacionadas con esta sub dimensión. La pregunta que orientó el relevamiento fue la siguiente:

### **5 - ¿Utilizan algún proceso para llevar a cabo la comunicación con las startups?**

Si bien ninguna de las empresas expresó tener un proceso definido para la comunicación si nombraron varios puntos que llevan a cabo con frecuencia entre los cuales se encuentran: especificar roles, estos varían según la empresa y el tipo de

proyecto pero los más nombrados fueron project manager, product manager y team leader. Definir un canal compartido en la herramienta Slack donde se comunican los desarrolladores y el cliente, y definir la frecuencia. Si bien esta se especifica ya sea semanalmente o quincenalmente, la gran mayoría de los casos se remarca que también puede ser on demand. La empresa 1 señaló al respecto: “Además de las instancias formales que son una vez por semana, instancias donde nos juntamos con el cliente, releemos los requerimientos o simplemente vemos como esta de conforme con el rendimiento del equipo, hay comunicación on demand en cualquier momento del día”. La empresa 2 también agregó: “Intentamos que si surge una duda o se quiere validar algo al instante se utilicen videollamadas mediante Zoom o Google Meet y para ello tratamos de conocer bien los horarios, días de descanso y feriados de nuestros clientes”. La empresa 3 destacó que dependiendo de la formalidad del cliente y del grado de seniority de los desarrolladores se define un líder para que interactúe como mediador entre los desarrolladores y el cliente. La empresa 4 agregó que para cada proyecto tienen una figura denominada principal la cual cumple un rol comercial a muy alto nivel, mientras que de la parte más operativa del proyecto se encarga un team leader y este es quien dialoga con los distintos clientes. Sin embargo, la empresa 1 indico que tratan que no haya una comunicación una o uno, si no que, sea siempre por canales compartidos donde todos sepan de que se está hablando.

## 4.2.4 Preguntas sobre dificultades

Otra de las subdimensiones de análisis de interés para este proyecto ha sido identificar cuál o cuáles son las principales dificultades encontradas por las empresas cuando deben desarrollar software para startups.

### **4 - ¿Cuáles son las principales dificultades que enfrentan a la hora de desarrollar software para startups?**

Uno de los aspectos altamente nombrado fue el de la poca certeza por parte de los clientes en cuanto a lo que realmente quieren. La empresa 1 lo catalogó como un problema clásico en cualquier desarrollo de software y a su vez agregó: “al no saber lo que quiere el cliente y por ejemplo, contratar un servicio de staff augmentation, capaz que se pierden roles importantes que ni los tiene la empresa ni los tiene este tipo de servicio.” La empresa 2, reafirmando lo anterior señaló que, en su caso, una gran cantidad de startups llegan con poco o nulo conocimiento en el área de tecnología y esto conlleva a no desarrollar directamente lo que en un principio quieren y por el contrario, guiarlas a llegar a una solución más efectiva y adaptada a sus necesidades. Por otro lado, todas las empresas mencionaron la comunicación

como un gran desafío resaltando la definición de lo que hay que hacer y en cuanto tiempo debe ser desarrollado como los dos puntos que con más frecuencia son propulsores a generar conflictos. Finalmente, la empresa 2 mencionó que para mitigar esto último, han incorporado en la empresa cursos para el desarrollo de habilidades blandas, mientras que la 4 debido a su experiencia, destacó que cada vez se enfrentan menos a problemas de este tipo.

En conclusión, podemos resumir entonces de estas entrevistas los siguientes hallazgos:

- Si bien la metodología ágil predominante a la hora de realizar los proyectos es SCRUM, cada una de las empresas la adopta y toma aspectos de la misma según su conveniencia.
- Salvo por una de ellas, las empresas no se rigen por un tipo de desarrollo en específico, más bien, esto es determinado por cada uno de los diferentes clientes.
- Las actividades de la metodología ágil más utilizadas son las daily meetings.
- No hay una manera formal de especificar las tareas, las herramientas más utilizadas son Jira y Trello y son utilizadas como forma de documentar las mismas.
- Si bien las daily meetings son el tipo de reunión mas utilizado, también se recurre a la la comunicación bajo demanda a medida que se necesita.
- Uno de los problemas más frecuentes es el mal interpretar que es lo que realmente quiere el cliente.

# 5 Discusión y síntesis del proceso

## 5.1 Proceso Bentos-Pera

En la investigación presentada por Bentos y Pera [3] podemos observar los resultados de su trabajo respecto a la propuesta de Proceso liviano de software para emprendimientos tecnológicos incubados (citado a partir de este momento como proceso B-P). Según este proyecto de investigación, la gran mayoría de las startups utilizan Scrum como metodología adaptando la misma según sus necesidades. Esta metodología es utilizada debido a la naturaleza que tienen los emprendimientos estudiados: bajos recursos, incertidumbre relacionada con el producto y el segmento de clientes y poca capacidad técnica específica. Según se indica en esa investigación, esta elección por parte de las empresas apunta a una mejor organización del trabajo y a evitar desperdicios, siendo estos dos elementos fundamentales de la metodología Scrum.

En el proceso B-P se han contemplado entre otros aspectos: la forma de priorizar y estimar las tareas, la correcta planificación y la gestión del tiempo. Uno de los hallazgos de la tesis presentada por Bentos y Pera es haber identificado que el proceso de software que deben utilizar las startups debe adaptarse a su realidad de forma coherente. Considerar tanto la relación entre la cantidad de miembros y los roles que estos cumplen en la empresa, como el contexto de bajo recursos e incertidumbre en la cual se mueven.

En ese estudio se definieron tres aspectos fundamentales que debe soportar el proceso:

- La creación de la pieza de software con la calidad que el emprendimiento requiera.
- Debe evitar el desperdicio de trabajo tanto como sea posible.

- Debe amalgamarse de forma correcta con el proceso de descubrimiento, validación y creación de productos y servicios que la startup realice.

El proceso B-P ha sido planteado siguiendo el modelo Lean startup [9], el cual se basa en “Invertir poco y aprender mucho”, y ha sido apoyado en el marco de trabajo Scrum para organizar de forma ágil y sencilla las distintas actividades y la comunicación con el equipo.

Se definieron 5 grandes etapas que conforman el proceso: Relevación de Requerimientos, Planificación, Construcción, Validación y, Organización y Gestión, los cuales se ilustran en la figura 5.1.

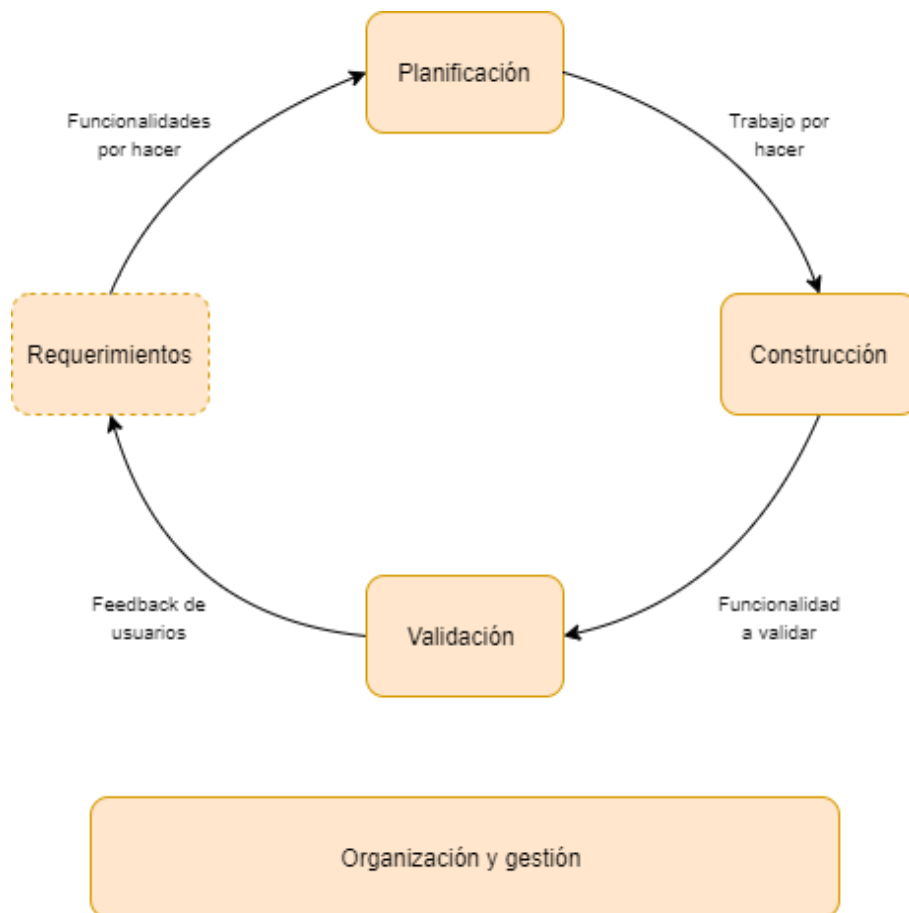


Figura 5.1: Proceso B-P

A continuación se especifica para cada etapa, una breve descripción, su owner y la participación de los usuarios o clientes si así corresponde.

<b>Etapa</b>	<b>Descripción</b>	<b>Owner</b>	<b>Participación de usuario-cliente</b>
Relevación de Requerimientos	Punto de entrada al proceso cíclico. Se transforman los requisitos de los usuarios en directivas de trabajo. Se obtiene como resultado el Product Backlog con las Historias de Usuario.	Product owner.	Obligatorio.
Planificación	El objetivo de este proceso es organizar el trabajo a partir de los requisitos obtenidos y priorizar aquellos que agreguen más valor al usuario. Se obtiene como resultado un Sprint Backlog que contiene las tareas a realizar en la siguiente etapa de construcción.	El equipo del emprendimiento en su conjunto. Equipo de construcción y diseño.	Opcional.
Construcción	Tiene como objetivo el diseño y creación del producto que el emprendimiento requiere. Se obtiene un incremento del MVP.	El equipo del emprendimiento en su conjunto. Equipo de construcción y diseño.	Opcional.
Validación	El objetivo de este proceso es conocer si el producto construido cumple con las expectativas de los usuarios.	No se especifica.	Obligatorio.
Organización y Gestión	Único subproceso transversal. Tiene como objetivo la organización del proyecto en su totalidad, abarcando sus diferentes aspectos. Se buscan mejoras a nivel de equipo y organización siguiendo los lineamientos de Scrum, y se gestiona la comunicación dividiéndola en dos tipos: comunicación colaborativa e informativa.	El equipo del emprendimiento en su conjunto.	Opcional.

Tabla 5.1: Descripción de etapas del proceso B-P

A continuación se listan los roles especificados como necesarios dentro del proceso B-P:

Rol	Descripción
Ventas/Comercial	Es el nexo entre el equipo emprendedor y los interesados, early adopters, clientes, etc. Busca mejorar y fortalecer como es el relacionamiento con ellos. Debe identificar y localizar a nuestros clientes o potenciales clientes.
Administración	Gestionar y llevar adelante las tareas subyacentes a la ejecución del emprendimiento. Debe manejar costos, estimaciones de recursos financieros y demás. También debe dar soporte a Ventas / Comercial en las actividades de gestión de clientes.
Tecnológico	La principal responsabilidad es construir el producto o servicio. Debe conocer los aspectos tecnológicos relacionados con la solución propuesta por el emprendimiento y conocer cómo integrarlo según los requisitos planteados.
Diseñador UX	La principal responsabilidad es construir el producto o servicio. Debe conocer los aspectos tecnológicos relacionados con la solución propuesta por el emprendimiento y conocer cómo integrarlo según los requisitos planteados.

Tabla 5.2: Roles necesarios en el proceso liviano de software

## 5.2 Propuesta de extensión del proceso para outsourcing

El modelo previamente descrito, planteado por Bentos y Pera ha sido orientado para el estudio de la interacción entre startups y cliente final, en cambio, este proyecto ha sido enfocado en la interacción entre startups y empresas donde tercerizan el desarrollo de software.

El proceso B-P ha sido un antecedente valioso para la elaboración de esta propuesta. Por un lado, con la información recolectada se ha podido confirmar la relevancia de los cinco elementos considerados (Requerimientos, Planificación, Construcción, Validación y Organización y gestión). No obstante, esta investigación ha constatado la existencia de una nueva área que, si bien ha sido mencionada en el modelo anterior, debería ser jerarquizada o tratada con mayor atención y prioridad. Esta es, la comunicación.

Si bien el proceso de Organización y Gestión es transversal y debe llevarse a cabo durante todas las etapas, uno de los hallazgos demostró que es de vital importancia prever y planificar diferentes instancias de intercambio y comunicación efectiva entre las partes. Es por esto que se ha generado un nuevo modelo de proceso incluyendo este último aspecto como foco central.

En este modelo de proceso la comunicación ocupa un lugar destacado ya que tal como lo definen Tuten y Urban [27] la comunicación es uno de los factores de riesgo altamente influenciados en el éxito de la colaboración con empresas que brindan servicios de outsourcing.

Se han identificado dos etapas en el proceso ya existente donde la comunicación juega un papel importante a la hora de que las mismas sean llevadas a cabo con éxito y que no tengan repercusión negativa en el resto del proceso cíclico. Estas dos etapas son Requerimientos (siendo este el punto de entrada al proceso cíclico) y la etapa de Validación. Las mismas se encuentran en momentos claves del desarrollo, donde la toma de decisiones y la participación del cliente son sumamente influyentes y determinantes.

De todas formas, no se debe dejar de lado que la comunicación debe ser abordada con importancia durante todo el proceso B-P ya que la obtención de buenos resultados se logra de una buena interacción y entendimiento entre los interesados que forman parte del proyecto.

Se presentan entonces, para cada una de estas dos etapas (Requerimientos y Validación) una nueva propuesta de modelo donde se agregan hitos referentes a la buena implementación de la comunicación.

### **Requerimientos**

Como se especifica en la figura 5.2, en la etapa de relevamiento de requerimientos se destacan tres grandes hitos de comunicación que se abordan en diferentes fases. La comunicación inicial, es un hito que debe ser realizado previamente a la ejecución del proceso cíclico para poder definir y dejar en claro los métodos y la frecuencia de comunicación a llevar a cabo.

Quienes deben participar de esta instancia son el scrum master por parte de la empresa que brinda servicios de outsourcing y el product owner por parte de la startup. La comunicación continua, debe llevarse a cabo durante toda la etapa de relevamiento de requerimientos, es decir, a medida que surjan dudas y se presenten aspectos que no se entiendan con claridad, se debe mantener una comunicación periódica.

Por último, la etapa de obtención de feedback, refiere a una etapa donde se evacuan dudas y reafirman supuestos, esta etapa debe realizarse al finalizar cada reunión con la finalidad de que ambas partes se encuentran alineadas respecto a lo conversado en dicha reunión.

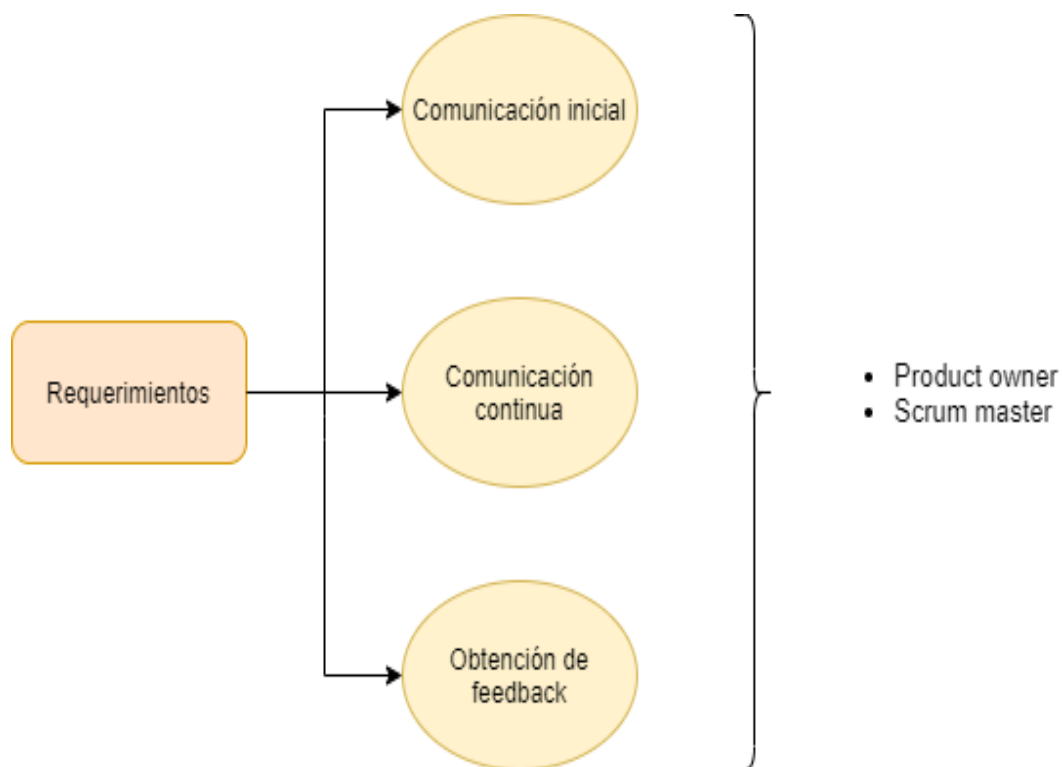


Figura 5.2: Hitos de la etapa Requerimientos

Hito	Roles	Descripción
Comunicación inicial	Product owner - scrum master.	Se establecen medios de comunicación.
Comunicación continua	Equipo de desarrollo - product owner. Product owner - scrum master.	Al surgir dudas o inquietudes por parte del equipo de desarrollo, el mismo debe recurrir al product owner para evacuarlas. De ser necesario, este último debe apelar al scrum master.
Obtención de feedback	Equipo de desarrollo - product owner. Product owner - scrum master.	Esta interacción debe darse al finalizar cada reunión, de ser necesario, para evacuar dudas.

Tabla 5.3: Hitos de la etapa Requerimientos y descripción de roles involucrados

\*el equipo de desarrollo es comprendido por desarrolladores en su amplia gama, front-end y back-end, arquitectos, diseñadores UX/UI, equipo de QA, etc.

## **Validación**

Para esta etapa, como se muestra en la figura 5.3, se ha identificado un nuevo hito denominado triangulación de conocimientos. El mismo refiere a una instancia integrada por: el product owner, quien es representante de la startup y posee los conocimientos del negocio. El scrum master, quien es representante de la empresa que brinda el servicio de outsourcing y es responsable de la gestión del desarrollo, y, el usuario final.

De esta forma, se puede recolectar información respecto a posibles mejoras y modificaciones en los requerimientos. El usuario toma un rol importante al ser incluido como una nueva fuente de información, donde mediante técnicas de retroalimentación se busca alinear las perspectivas de las tres partes involucradas.

Así mismo, es importante la presencia del product owner en esta instancia, ya que debido a la definición de startup, es posible que el scrum master no cuente con la experiencia o conocimiento necesario para poder evaluar esa información recolectada luego de la prueba con el usuario.

Se listan a continuación actividades y técnicas que para poder detectar estos problemas de usabilidad y evaluar la experiencia del usuario.

- Prototipación
- A/B testing
- Alpha testing
- Beta testing
- Demos
- Observación
- Pruebas con usuarios



Figura 5.3: Hitos de la etapa Validación

Hito	Roles	Descripción
Triangulación de conocimientos	Product owner - scrum master - usuario final.	Se establece una interacción entre estos los tres interesados con el fin de poder obtener un feedback por parte del usuario teniendo en cuenta tanto, el conocimiento técnico del scrum master como la experiencia del product owner.

Tabla 5.4: Hitos de la etapa Validación y descripción de roles involucrados

## 5.3 Propuesta de buenas prácticas para el desarrollo del proceso

Tal como se ha desarrollado en el marco teórico, el modelo de Kristjansson *et al.* [16] es un buen ejemplo de buenas prácticas. En él se reconoce que es necesario intercambiar conocimientos en las distintas fases de un proyecto de desarrollo de software y no sólo en una única instancia al inicio del mismo. Las tres fases principales de su propuesta refieren al inicio del proyecto donde deben establecerse claramente los medios de comunicación a ser utilizados, al desarrollo continuo de las 5 prácticas consideradas exitosas (colaboración funcional, specification grooming, asistencia técnica, documentación técnica, product knowledge center - ver 2.8) y a la entrega del equipo. Esta última fase hace referencia a la transición en la que el proyecto pasa a manos de otro equipo.

En base a la literatura revisada se han identificado las siguientes buenas prácticas a tener en cuenta a la hora de establecer la comunicación entre ambas partes.

1. **Establecer comunicación fluida y en tiempo real:** al subcontratar empresas, puede suceder que las mismas no se encuentren en la misma zona horaria, por lo que es importante que se identifiquen las herramientas adecuadas para poder establecer comunicación sincrónica. Establecer canales de comunicación donde todas las personas que forman parte del proyecto puedan interactuar, evacuar dudas y estar al día con el desarrollo del mismo.
2. **Organizar reuniones regulares de actualización y demo:** asegurarse de que el equipo que conforma la startup de software tiene el hábito de programar reuniones regulares para poder así demostrar el progreso y recibir comentarios. La retroalimentación debe ser incorporada a la estructura del funcionamiento del equipo. Esto además, ayuda a que la startup esté al día con lo que se va haciendo y se puedan detectar tanto errores como mejoras en el menor tiempo posible.
3. **Promover la utilización flexible de herramientas tecnológicas:** de ser necesario, se debe recurrir a un amplio repertorio de herramientas para la obtención de feedback como por ejemplo videollamadas o demostraciones en vivo. Este tipo de actividades permitirían profundizar y resolver problemas de manera más eficiente que si se hicieran por medio de una conversación en herramientas como Slack. Se debe tener en cuenta, que se debe recurrir a este tipo de reuniones de manera consciente, ya que el uso excesivo de las mismas pueden ralentizar el proceso de desarrollo en su totalidad.

Para tener una comunicación eficiente, es necesario utilizar herramientas que permitan y habiliten que la misma sea llevada a cabo de manera dinámica y continua, donde todos los participantes puedan inferir y participar de manera proactiva.

Cada vez con más tendencia y dinamismo las empresas están optando por la implementación de modalidades de trabajo remoto (más aún ahora debido al contexto de emergencia sanitaria que muchos países tienen establecidos) por lo que es importante detectar cuales son estas herramientas que mejoran y permiten el buen entendimiento de equipos distribuidos.

Luego de una revisión de los recursos tecnológicos disponibles para la comunicación se han detectado cuatro herramientas consideradas muy útiles para establecer diferentes estrategias de comunicación que favorecen la sinergia entre las empresas participantes. Las mismas son detalladas en la siguiente sección.

### **Google calendar**

Utilizar esta agenda y calendario electrónico para marcar reuniones, deadlines y estar ordenados ya sea diaria, semanal o mensualmente. Permite crear reuniones, sincronizando las mismas con los calendarios de otros usuarios.

Asimismo, se pueden crear calendarios múltiples, los mismos pueden ser compartidos en modo solo lectura o con control completo generando una colaboración efectiva entre las personas que lo utilicen. Por ejemplo, si un colaborador sale de licencia puede marcarlo en el calendario para que el resto del equipo esté al tanto. Por otro lado, esta herramienta, puede ser sincronizada con otras aplicaciones, como por ejemplo Slack. De esta forma se puede por ejemplo, enviar notificaciones a un canal en específico cuando se aproxima la fecha de determinada reunión o evento.

También es útil para identificar los días festivos y feriados de otros países, para de esta manera tenerlos en cuenta al momento de establecer deadlines o reuniones. Al crear una reunión, permite que, luego de ser aceptada por los invitados, la misma sea incorporada de manera inmediata en los calendarios de los mismos. Se sugiere que las reuniones pautadas con esta herramienta no deben ser menores a 20 minutos, de lo contrario utilizar otras herramientas. Además de deadlines y reuniones, puede ser utilizada para la realización de demostraciones para obtención de feedback.

### **Asana**

Esta aplicación web y móvil ayuda a los equipos a organizar, rastrear y administrar su trabajo. Proporciona una interfaz que posibilita la gestión de proyectos y tareas

entre dos o más personas. Permite crear subgrupos de tareas para ordenarlas como por ejemplo, por proyecto o departamento. Para una mejor especificación de las tareas las mismas deben componerse de por lo menos:

- Descripción breve en el título.
- Descripción detallada en el cuerpo.
- Owner.
- Colaboradores.
- Fecha de finalización.

Así mismo, se debe tener en cuenta que si una tarea no puede ser realizada para la fecha esperada, se debe volver a estimar y notificar etiquetando a quienes correspondan. Esta herramienta, también permite ser sincronizada con otras aplicaciones como Slack y Gmail, por lo que es recomendable para tener una mejor trazabilidad de las tareas mediante notificaciones.

### **Slack**

Es una herramienta de chat para mantener conversaciones con no más de 2 horas de respuesta. Permite generar canales de comunicación, proporcionando que los mismos sean destinados a diferentes proyectos, temas o tópicos. Por otro lado, se pueden realizar llamadas y videollamadas.

Se aconseja que se realicen siempre y cuando se quieran evacuar dudas o afirmar supuestos en reuniones de no más de 15 a 20 minutos. Suele ser usada para compartir artículos e información de interés de manera informal y para generar vínculos entre los miembros del equipo. Suele ser útil separar distintos canales por temas, por ejemplo:

- Desarrollo.
- Testing.
- UX/UI.
- Arquitectura.

### **Gmail**

Esta herramienta si bien es similar a Slack, permite redacciones más complejas. Así

mismo, las diferentes notificaciones generadas por parte de las otras herramientas pueden ser centralizadas en esta plataforma. Por otro lado, se pueden generar distintos grupos para ordenar y descentralizar la bandeja de entrada creando una especie de filtro según las necesidades del usuario.

También, los mails pueden ser marcados mediante etiquetas para identificar la importancia o prioridad que los caracteriza, y, de igual manera, pueden ser archivados para despejar la contaminación visual sin la necesidad de borrarlos. Se aconseja que, para temas importantes, las respuestas deben darse en 24 horas como máximo, mientras que para el resto no deben superarse las 48 horas.

Cabe destacar que, para lograr una comunicación más efectiva, es necesaria la utilización e interacción de todas estas herramientas en su conjunto. En la tabla 5.5 se pone en evidencia la relación entre estas herramientas junto con una breve descripción del flujo de interacción entre las mismas y las posibles etapas donde pueden darse dichas interacciones.

Herramienta de comunicación	Flujo	Posibles etapas
Mail - Asana	Luego de un intercambio mediante mails, si así se amerita, se determinan las tareas en Asana, de esta forma queda ordenado y se puede hacer seguimiento de lo hablado previamente.	Requerimientos.
Slack- Asana	Si se discuten aspectos de las tareas a realizar mediante el chat de slack, indicar dichos acuerdos en las tareas correspondientes en Asana de modo que los colaboradores se mantengan al tanto de los mismos.	Requerimientos. Construcción
Asana - Slack	Si al tomar una tarea de asana se generan dudas sobre la misma, contactar a los colaboradores mediante Slack, y de ser necesario realizar llamadas cortas para evacuar dudas o afirmar supuestos.	Requerimientos. Construcción
Asana - Google calendar	Una vez que se finaliza un conjunto de tareas en Asana, marcar una instancia de demostración y obtención de feedback mediante un evento en el calendario de los colaboradores.	Validación

Tabla 5.5: Flujo de interacción entre las herramientas de comunicación

# 6 Conclusiones

## 6.1 Lecciones aprendidas

La realización de este proyecto ha sido comprendida en su totalidad en tiempos de emergencia sanitaria debido a la pandemia por coronavirus COVID-19. El realizar un trabajo de esta magnitud sin previo conocimiento en el área de la investigación, no solo ha sido, en lo personal, un desafío importante sino que, esta situación atípica ha hecho que el trabajo se caracterice por particularidades y contratiempos que de no ser por este contexto no hubieran surgido.

Por un lado, las medidas de aislamiento social preventivo han afectado directamente la recolección de datos. El realizar las entrevistas por medio de videollamadas, han quitado aspectos esenciales a la hora de establecer contacto con los entrevistados, como por ejemplo, el contacto visual o la charla previa para “romper el hielo”. Esto ha llevado a que las entrevistas fueran no solo menos fluidas sino que también se reduzcan los tiempos de las mismas.

Así mismo, conseguir entrevistas con empresas que cumplieran el perfil buscado ha sido dificultoso. La tarea de obtener un primer contacto con las empresas se realizó en el mes de Septiembre, y si bien en un principio las mismas se mostraron a disposición, luego, la etapa de llevar a cabo la entrevista debió ser suspendida debido a que en la mayoría de los casos, las empresas se encontraban en un periodo de adaptación a la semi-presencialidad. Esto derivó en que las mismas se encontraran con otras prioridades y escasa disponibilidad. Este último aspecto, conlleva a que se tengan entonces, pocos casos en la muestra.

## 6.2 Amenazas a la validez

Las amenazas a la validez de los resultados de los estudios cualitativos han sido discutidas por varios autores y referentes de la investigación a lo largo de los

años. Tal como lo indica Hernández Sampieri [23], las mismas pueden referirse a la confiabilidad, la transferencia de los resultados, entre otros aspectos. En este caso, la credibilidad y validez de resultados de este proyecto, podría verse debilitada debido a los siguientes tres aspectos detectados.

En primer lugar, el número reducido de las entrevistas realizadas. Este aspecto es una limitante ya que los resultados deben ser interpretados en función de las perspectivas y las particularidades de los casos entrevistados, restringiendo poder atribuir los análisis en ámbitos generalizados.

En segundo lugar, la recolección y análisis de los datos ha sido realizada teniendo en cuenta solamente una de las entidades que conforman el objeto de estudio, las empresas que brindan servicios de outsourcing. Esto limita una revisión profunda que podría verse enriquecida si se realizara también un análisis desde la perspectiva de las startups. Así mismo, las entrevistas nos han dado una visión generalizada del tema de estudio pero sin bajar a nivel de proyectos concretos. Ahondar en otros contextos, considerando dificultades y particularidades a la hora de llevar los proyectos a la práctica podría ser de gran utilidad para potenciar esta investigación.

Por último, tanto la nueva propuesta de proceso como el conjunto de buenas prácticas sugerido, han sido basados en la perspectiva de los sujetos pero no se sometieron a validaciones en la práctica. Este último podría ser un nuevo desafío a asumir por futuras investigaciones que profundicen en la temática.

## 6.3 Trabajos futuros

El proceso planteado en este proyecto presenta un nuevo hito adicional respecto al sugerido por Bentos y Pera. Un aporte distintivo de esta tesis es la necesidad de jerarquizar el intercambio de conocimiento y la comunicación efectiva entre los participantes. De igual manera, entre los resultados de esta tesis se detalla una lista de buenas prácticas a llevar a cabo durante toda la colaboración entre la startup y la empresa donde tercerizan su código. Vale destacar que, si bien estas propuestas han sido elaboradas luego de un amplio trabajo de campo, las mismas deberían ser analizadas y validadas por expertos.

En este sentido, en futuros trabajos sería conveniente que estas propuestas sean puestas en práctica para, por un lado, evaluar su impacto cuando son puestas en ejecución, y por el otro, obtener potenciales cambios o mejoras. Esto podría hacerse dentro del ámbito académico, como por ejemplo, siguiendo el mismo hilo

de este tipo de proyectos, en startups universitarias. O, podrían aprovecharse las instancias de validación del proceso B-P para validar también, de forma paralela, esta nueva propuesta en aquellos casos donde las startups utilicen la técnica de outsourcing.

## 7 Referencias bibliográficas

- [1] I. Sommerville, *Software Engineering*, 9th ed., México: Addison-Wesley, 2011.
- [2] F. Galperin and E. González, “Replicación de un estudio sobre habilidades técnicas y blandas en emprendimientos tecnológicos de software,” Tesis de grado - Licenciatura en Sistemas, Universidad ORT Uruguay, Montevideo, Uruguay, 2020.
- [3] C. Pera and A. Bentos, “Gestión de producto mínimo viable y modelo de software ágil para emprendimientos tecnológicos de software,” Tesis de grado - Ingeniería en Sistemas, Universidad ORT Uruguay, Montevideo, Uruguay, 2020.
- [4] I. Benbasat, D. Goldstein, and M. Mead, “The case research strategy in studies of information systems,” *MIS Quarterly*, vol. 11, no. 3, Set., 1987.
- [5] E. Ries, *The lean startup: How today’s entrepreneurs use continuous innovation to create radically successful businesses*, London, United Kingdom: Crown books, 2011.
- [6] N. Paternoster, C. Giardino, M. Unterkalmsteiner, T. Gorschek, and P. Abrahamsson, “Software development in startup companies: A systematic mapping study,” *Elsevier*, vol. 56, no. 10, Oct., 2014.
- [7] C. Giardino, M. Unterkalmsteiner, N. Paternoster, T. Gorschek, and P. Abrahamsson, “What do we know about software development in startups?” *IEEE Software*, vol. 32, no. 5, Set., 2014.
- [8] Agilemanifesto, “Manifiesto por el desarrollo Ágil de software,” 2001. [Online]. Available: <http://agilemanifesto.org/iso/es/manifesto.html>
- [9] E. Ries, *El método Lean Startup*, London, United Kingdom: Crown books, 2002.

- [10] M. Poppendieck and M. A. Cusumano, “Lean software development: A tutorial,” *IEEE Software*, vol. 29, no. 5, Jun., 2012.
- [11] M. Poppendieck and T. Poppendieck, *Lean Software Development. An Agile Toolkit*, 1st ed., New York, USA: Addison-Wesley, 2003.
- [12] K. Rubin, *Essential Scrum. A practical guide to the most popular agile process*, 1st ed., New York, USA: Addison-Wesley, 2013.
- [13] N. Krym, *Outsource It! A No-Holds-Barred Look at the Good, the Bad, and the Ugly of Offshoring Tech Projects*, 1st ed., California, USA: Pragmatic bookshelf, 2013.
- [14] B. Gallagher, M. Phillips, K. Richter, and S. Shrum, *CMMI for Acquisition Guidelines for Improving the Acquisition of Products and Services*, 2nd ed., USA: Addison-Wesley, 2011.
- [15] B. Vebjørn, B. Jørgen, N.-D. Anh, P. Ilias, and J. Letizia, “Exploring the outsourcing relationship in software startups – a multiple case study,” Conference: the 21st International Conference, Jun., 2017.
- [16] B. Kristjansson, R. Helms, and S. Brinkkemper, “Integration by communication: knowledge exchange in global outsourcing of product software development,” *Expert Systems: The Journal of Knowledge Engineering*, Jul., 2014.
- [17] A. Buffa and D. Febles, “Habilidades técnicas y blandas en las startups de software y su evolución,” Tesis de grado - Licenciatura en Sistemas, Universidad ORT Uruguay, Montevideo, Uruguay, 2019.
- [18] A. González and G. Nieto, “El proceso de creación y evolución del producto mínimo viable en las startups de software,” Tesis de grado - Licenciatura en Sistemas, Universidad ORT Uruguay, Montevideo, Uruguay, 2019.
- [19] S. Peralta and G. Pérez, “Inicio y evolución de las prácticas y actividades de ingeniería en las startups de software,” Tesis de grado - Licenciatura en Sistemas, Universidad ORT Uruguay, Montevideo, Uruguay, 2019.
- [20] A. N. Duc and P. Abrahamsson, “Exploring the outsourcing relationship in software startups – a multiple case study,” Conference: the 21st International Conference, Jun., 2017.
- [21] M. A. Babar, J. M. Verner, and P. T. Nguyen, “Establishing and maintaining trust in software outsourcing relationships: An empirical investigation,” *Elsevier*, vol. 80, no. 9, Sep., 2007.

- [22] Y. Yong, L. NengSheng, and B. Hong, “Technology innovation and risks in outsourcing,” *IEEE Software*, vol. 25, no. 5, Nov., 2008.
- [23] R. Sampieri, C. Fernandez, and P. B. Lucio, *Metodología de la investigación*, 6th ed. México: McGraw Hill, 2014.
- [24] P. Runeson, M. Host, A. Rainer, and B. Regnell, *Case study research in Software Engineering. Guidelines and examples*, 1st ed., New York, USA: Wiley, 2012.
- [25] R. Yin, *Case Study Research: Design and Methods*, 5th ed., California, USA: SAGE Publications, 2013.
- [26] C. Robson, *Real world research*, 4th ed., New York, USA: Wiley, 2002.
- [27] T. Tuten and D. Urban, “An expanded model of business-to-business partnership formation and success,” *Elsevier*, vol. 30, no.3, Feb., 2001.