**Carta de Proyecto ágil.**

**Project Charter Scrum**

**Proyecto Rhea**

**Cliente: [Jesus Cortes Urriola]**

**Nombres:**

* **Franco Fuentes**
* **Miguel Garate**

**Sección: 001D**

**Docente : Christian Lazcano**

**Historial de Revisiones**

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Versión** | **Descripción** | **Autor/es / Integrantes** |
| 17/09/2024 | 1.0 | Se realizo los puntos: 2,3,5 | Miguel G |
| 18/09/2024 | 1.1 | Se realizó los puntos: 4,11,12 | Miguel G |
| 19/09/2024 | 1.2 | Se realizó los puntos: 6,10 | Miguel G |
| 20/09/2024 | 1.3 | Se realizó los puntos: 7,8,9 | Miguel G |
| 22/09/2024 | 1.4 | Se realizó los puntos: 13, 15, 16, 17 ,18. Etapa2: 1, 2, 3, 4, 5, 6 ,7 ,8, 9 | Franco Fuentes |
| 22/09/2024 | 1.5 | Se realizó los puntos, Etapa2: 11,12,13 | Miguel G |

# Contenido

Contenido

[2. Contenido 3](#_Toc177988764)

[3. DEFINICIÓN DEL PROYECTO 5](#_Toc177988765)

[3. Propósito de este documento 5](#_Toc177988766)

[4. Problemática por resolver 5](#_Toc177988767)

[5. Visión 5](#_Toc177988768)

[6. Modelo de Negocios Canva 6](#_Toc177988769)

[7. Objetivos 7](#_Toc177988770)

[7.  Objetivos Técnicos: 7](#_Toc177988771)

[7.  Objetivos Comerciales: 7](#_Toc177988772)

[7.  Objetivos de Producto: 7](#_Toc177988773)

[8. Estimación del tamaño del proyecto 7](#_Toc177988774)

[9. Estimación de la complejidad del proyecto 8](#_Toc177988775)

[10. Alcances 8](#_Toc177988776)

[11. Organización del proyecto 8](#_Toc177988777)

[12. Recursos 9](#_Toc177988778)

[13. Metodología de trabajo 9](#_Toc177988779)

[14. Criterios de éxito (Definition of Done) 11](#_Toc177988780)

[15. Prioridades (Product Backlog) 11](#_Toc177988781)

[16. Hoja de ruta del producto (Roadmap) 11](#_Toc177988782)

[17. Supuestos y limitaciones (Alcances) 12](#_Toc177988783)

[17. Supuestos: 12](#_Toc177988784)

[17. Limitaciones 12](#_Toc177988785)

[18. Riesgos y problemas 13](#_Toc177988786)

[19. Valores de trabajo con Scrum 13](#_Toc177988787)

[20. Personas y roles del proyecto. 13](#_Toc177988788)

[21. Épicas a cubrir 15](#_Toc177988789)

[22. Perfil de Usuarios relacionados 16](#_Toc177988790)

[23. Principales historias de usuarios por Épicas 17](#_Toc177988791)

[24. Definición de responsabilidades del Equipo por Rol definido 18](#_Toc177988792)

[25. Visión del producto 19](#_Toc177988793)

[26. Tecnologías de Desarrollo e Implementación 19](#_Toc177988794)

[27. Componentes y Principales Artefactos por épicas 19](#_Toc177988795)

[28. Product Backlog 20](#_Toc177988796)

[29. Definición del Done 22](#_Toc177988797)

[30. RoadMap 23](#_Toc177988798)

[31. Diseño inicial del software 24](#_Toc177988799)

[31. Diagramas de Caso de Uso General 24](#_Toc177988800)

[31. Diagramas de Caso de Uso extendido(Reservar Cita) 25](#_Toc177988801)

[31. Diagrama de Caso de Uso extendido(Crear pyme) 26](#_Toc177988802)

[31. Diagramas de Actividad (Reservar Cita) 26](#_Toc177988803)

[31. Diagramas de Actividad (Crear Pyme) 28](#_Toc177988804)

[31. Modelo de Datos (MER) 29](#_Toc177988805)

[32. Herramientas de gestión ágil 29](#_Toc177988806)

[33. Anexos. 30](#_Toc177988807)

[33. Mapa de Actores 30](#_Toc177988808)

[33. Modelos UML de la solución 31](#_Toc177988809)

[33. Prototipado la solución 32](#_Toc177988810)

# DEFINICIÓN DEL PROYECTO

## Propósito de este documento

Este documento describe el propósito del proyecto, sus objetivos, alcances y la implementación de la metodología de trabajo scrum en la empresa para la gestión del desarrollo del proyecto.

Incluye junto con la descripción de este ciclo de vida iterativo e incremental para el proyecto, la propuesta solución y los artefactos o documentos anexos con los que se gestionarán las tareas de adquisición y suministro: requisitos, monitorización y seguimiento del avance, así como las responsabilidades y compromisos de los participantes en el proyecto.

# Problemática por resolver

Las pequeñas y medianas empresas (pymes) suelen enfrentar dificultades en la gestión de reservas y pagos, lo que genera pérdida de clientes, errores en la programación manual, y falta de opciones de pago en línea. Rhea surge para resolver estos problemas, ofreciendo una plataforma digital accesible y personalizada que automatiza la gestión de citas, pagos y organización del personal, mejorando la eficiencia operativa y la experiencia del cliente, contribuyendo así al crecimiento y éxito de las pymes.

# Visión

Nuestra visión es ofrecer una plataforma que satisfaga las necesidades de las pequeñas y medianas empresas, facilitando su gestión de reservas, pagos y empleados. Queremos aliviar la carga operativa de estos negocios, ayudándolos a simplificar y automatizar sus procesos, permitiéndoles concentrarse en ofrecer un mejor servicio a sus clientes, contribuyendo a su crecimiento sostenible y mejorar su relación con los clientes, impulsando su éxito en un entorno cada vez más digitalizado. ￼

# Modelo de Negocios Canva



# Objetivos

##  Objetivos Técnicos:

1. Desarrollar una plataforma web escalable y personalizable basada en microservicios para gestionar reservas, pagos y personal de manera eficiente.
2. Integrar sistemas de pago en línea (como Transbank) para facilitar transacciones seguras y confiables.
3. Implementar notificaciones automáticas (por correo o SMS) para la confirmación de citas, recordatorios y cancelaciones.

##  Objetivos Comerciales:

* Facilitar a las pymes la adopción de una herramienta accesible que automatice su gestión diaria, reduciendo errores y aumentando su productividad.
* Mejorar la satisfacción del cliente final al ofrecer una experiencia de reserva fluida y sin complicaciones.

##  Objetivos de Producto:

* Crear una plataforma que permita a los negocios configurar su perfil, servicios, y empleados de manera sencilla y ajustada a sus necesidades particulares.
* Garantizar que la plataforma sea fácil de usar, tanto para el dueño del negocio como para los empleados y clientes.
* Permitir que los usuarios finales puedan reservar servicios, seleccionar profesionales y horarios, y pagar en línea o en tienda.

# Estimación del tamaño del proyecto

El proyecto Rhea se estima como un desarrollo de tamaño mediano. Implicará el desarrollo de una plataforma web con múltiples funcionalidades, incluyendo frontend y backend, integración de una base de datos, y sistemas de pago y notificaciones.

**Equipo de Desarrollo**:

* **Desarrolladores** (2): Para implementar tanto el frontend como el backend.
* **Tester/QA** (2): Para realizar pruebas manuales y automatizadas, asegurando que el producto funcione sin errores.
* **Scrum Master** (1): Para organizar y coordinar el trabajo del equipo.
* **Administrador de Bases de Datos** (1): para garantizar que el diseño, optimización, y mantenimiento de la base de datos cumpla con los requisitos del sistema.

**Infraestructura:**

* **Servicios de host:** permite que la aplicación esté disponible en línea para pruebas reales, acceso de usuarios, y conexión con servicios externos como APIs de pago.
* **Sistemas de Notificaciones y Pago**: Herramientas para notificaciones, e integración con un sistema de pago en línea.

# Estimación de la complejidad del proyecto

El proyecto tiene una complejidad moderada a alta, ya que involucra múltiples módulos, como la gestión de reservas, pagos en línea, notificaciones automáticas y administración de usuarios, lo que requiere una interacción constante entre componentes. Además, se integrarán sistemas externos para pagos y notificaciones. Aunque el proyecto iniciará con un volumen pequeño de usuarios, deberá estar preparado para escalar, optimizando tanto el backend como la base de datos para soportar transacciones simultáneas. También será crucial asegurar altos estándares de seguridad para proteger los datos personales y las transacciones de pago.

# Alcances

Necesidades conocidas/asumidas del cliente, requisitos funcionales y no funcionales anticipados.

**Requisitos Funcionales**

* Los clientes deben poder ver, seleccionar y reservar servicios ofrecidos por las pymes, eligiendo fecha, hora y profesional.
* Los empleados deben poder visualizar sus citas diarias
* Los dueños deben tener la capacidad de supervisar todas las reservas
* El sistema debe permitir a los dueños crear y personalizar el perfil de su negocio, agregar empleados, y definir los servicios ofrecidos.
* Los clientes deben poder realizar pagos en línea al momento de reservar
* El sistema debe registrar el estado del pago.
* El sistema debe enviar notificaciones.

**Requisitos No Funcionales**

* El sistema debe poder escalar para manejar un número creciente de pymes, clientes y transacciones
* El tiempo de respuesta del sistema debe ser rápido, asegurando que las reservas, pagos y consultas de datos se procesen
* Las contraseñas deben almacenarse de forma segura
* La interfaz debe ser intuitiva y fácil de usar, tanto para clientes como para dueños de pymes

# Organización del proyecto

El proyecto Rhea será guiado por Jesús Cortes Urriola, dueño de la barbería Black Buffalo Men's Salon. Como dueño y empleado del local, Jesús aportará su valiosa experiencia en el negocio, brindando feedback constante para asegurar que la plataforma se ajuste a las necesidades reales del sector. Su conocimiento del mercado y de los procesos operativos de su barbería será fundamental para el éxito del proyecto..

# Recursos

El proyecto Rhea comenzará en un ambiente de pruebas, utilizando únicamente los recursos propios del equipo. Este entorno de pruebas servirá para desarrollar y validar todas las funcionalidades clave de la plataforma, asegurando que todo funcione correctamente antes de migrar a un servicio de hosting profesional en el futuro.

**Recursos Humanos:**

* **Desarrolladores Full-Stack:** Implementarán tanto el frontend como el backend de la aplicación.
* **Administrador de Bases de Datos:** Encargado del diseño, optimización y seguridad de la base de datos.
* **Tester/QA:** Realizarán pruebas manuales y automatizadas para asegurar la calidad y funcionalidad de la plataforma.

**Recursos Tecnológicos**:

* **Visual Studio Code:** IDE principal para el desarrollo de frontend y backend.
* **GitHub:** Control de versiones y colaboración en el código
* **Cypress:** Herramienta para pruebas automatizadas de la plataforma.
* **Figma**: Herramienta para el diseño y prototipado de interfaces de usuario.
* **MySQL**: Base de datos relacional para almacenar datos
* **Transbank**: Usaremos Transbank para procesar pagos en línea de forma segura en la plataforma Rhea, permitiendo a los clientes pagar sus reservas directamente en la web.

# Metodología de trabajo

**Estrategias**

* **Enfoque Centrado en el Cliente**: El desarrollo se enfocará en resolver las necesidades de los dueños de pymes y sus clientes, asegurando que la plataforma sea intuitiva, eficiente y funcional para todos los usuarios.

**Metodología**

* **Scrum**: Se utilizará Scrum como metodología ágil para el desarrollo del proyecto, permitiendo una entrega continua e incremental de valor. El equipo trabajará en ciclos cortos, adaptándose de manera flexible a los cambios y priorizando siempre el valor entregado al usuario final. La revisión y mejora constante del proceso será clave en cada iteración.

**Procesos**

* **Historias de Usuario**: Las funcionalidades serán planificadas a través de historias de usuario, enfocando cada tarea en la solución de problemas específicos de los usuarios finales y asegurando que el desarrollo esté alineado con sus necesidades.

**Herramientas**

* **Jira**: Se usará para organizar tareas, gestionar el backlog y realizar el seguimiento del progreso del equipo, proporcionando visibilidad sobre el estado del proyecto.
* **Google Drive**: Se empleará para la colaboración y el almacenamiento compartido de documentos, facilitando el acceso y edición en tiempo real por parte del equipo.
* **GitHub**: Lo utilizamos para la gestión de versiones tanto en el código como en los archivos por cada fase.
* **Figma**: Herramienta para diseñar prototipos de las interfaces de usuario, permitiendo validar el diseño antes de su desarrollo.
* **Visual Studio Code**: Entorno de desarrollo que utilizará el equipo para escribir y depurar el código de la plataforma.

**Técnicas**

* **Historias de Usuario**: Se utilizarán para estructurar y priorizar las tareas del equipo, manteniendo el enfoque en las necesidades del usuario final.
* **Cypress**: Se utilizará para automatizar las pruebas de la plataforma, asegurando la correcta funcionalidad de los flujos clave y facilitando una rápida detección de errores en el proceso de desarrollo.

# Criterios de éxito (Definition of Done)

1. Todas las funcionalidades deben estar completamente implementadas, probadas y sin errores críticos.
2. Pruebas automatizadas con Cypress deben cubrir al menos el 80% de los flujos clave.
3. La interfaz de usuario debe ser intuitiva y validada mediante pruebas de usuario.
4. La experiencia del usuario debe ser clara y fácil de usar.
5. La integración con un sistema de pagos en línea debe funcionar correctamente, permitiendo transacciones fluidas.
6. Las notificaciones automáticas (confirmaciones, recordatorios, cancelaciones) deben ejecutarse sin fallos.
7. El producto debe ser probado por usuarios reales y recibir feedback positivo.
8. Todos los problemas identificados deben estar resueltos.

# Prioridades (Product Backlog)

Como equipo decidimos priorizar el Product Backlog siguiendo la técnica **MoSCoW**. Las historias de usuario están organizadas según su importancia para el producto y el impacto que tienen en los usuarios finales.

***Must Have (M):*** Estas funcionalidades son críticas para el sistema. Incluyen el sistema de reservas, la autenticación de usuarios, la visualización de reservas para clientes y empleados, y la integración de pagos en línea.

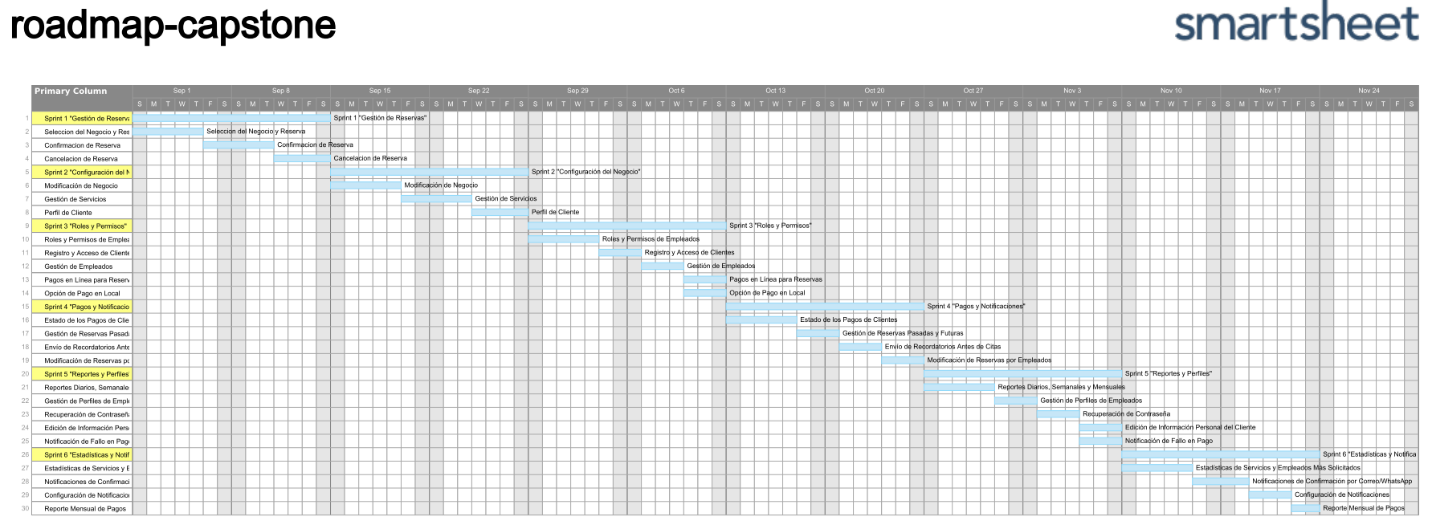
***Should Have (S):*** Funcionalidades importantes, pero no críticas, que se desarrollarán después de los Must Have. Incluyen la gestión de perfiles de clientes para fidelización y los reportes básicos.

***Could Have (C)***: Funcionalidades que aportan valor, pero que no son prioritarias para el funcionamiento del sistema. Incluyen la personalización de notificaciones y otros pequeños ajustes.

***Won’t Have (W)***: Funcionalidades que no se desarrollarán en esta versión.

En términos de compensaciones y objetivos, se priorizarán los módulos de reservas y pagos, ya que son esenciales para el funcionamiento del sistema. Las funcionalidades adicionales como la gestión avanzada de clientes o las estadísticas se implementarán en fases posteriores, dependiendo del tiempo y los recursos disponibles. (más adelante se adjuntará el Product Backlog priorizado)

# Hoja de ruta del producto (Roadmap)



[[Link RoadMap Drive]](https://drive.google.com/file/d/1kUA9JIEYOrFcmdRo99ad0wzIsxVWM1hH/view?usp=drive_link)

# Supuestos y limitaciones (Alcances)

## Supuestos:

* El equipo tendrá acceso a todas las herramientas necesarias, como Jira y GitHub, para gestionar el desarrollo del proyecto.
* Los recursos necesarios para el desarrollo y pruebas del sistema estarán disponibles durante 1 o 2 meses luego del inicio de la fase de desarrollo.
* Los usuarios clave (dueños de negocios y empleados) podrán participar en pruebas y revisiones del sistema durante el desarrollo para validar las funcionalidades.
* El sistema lo utilizarán inicialmente pequeñas y medianas empresas que buscan una solución simple y eficiente para gestionar reservas y servicios.

## Limitaciones

* No se incluirá la funcionalidad de gestión de promociones y descuentos en esta versión del proyecto.
* El desarrollo estará limitado a una **aplicación web**, sin incluir aplicaciones móviles nativas.
* El proyecto tendrá un enfoque en la **gestión básica de reservas, pagos y notificaciones**, priorizando las funcionalidades esenciales.
* Como el tiempo es acotado, se priorizaron las tareas más críticas para poder asegurar un lanzamiento funcional.

# Riesgos y problemas

**Riesgo 1**: Falta de tiempo para implementar todas las funcionalidades planeadas.

**Plan de mitigación**: Priorizar las tareas más importantes (Must Have) y dejar las funcionalidades menos críticas para después.

**Riesgo 2**: Problemas de integración con la API de pagos (Transbank).

**Plan de mitigación**: Iniciar la integración de pagos lo antes posible para resolver problemas tempranamente, y tener un plan alternativo de pago en local.

**Riesgo 3**: Dificultades técnicas con el despliegue en producción.

**Plan de mitigación**: Utilizar plataformas confiables de hosting como Heroku o Vercel, con un enfoque de despliegue continuo para detectar problemas con antelación.

# Valores de trabajo con Scrum

1. **Individuos e interacciones sobre procesos y herramientas**: El equipo se enfoca en la colaboración efectiva y continua entre desarrolladores, stakeholders y usuarios, priorizando la comunicación directa.
2. **Software funcionando sobre documentación extensiva**: El objetivo principal es entregar un sistema funcional al final de cada sprint, con iteraciones constantes para mejorar el producto.
3. **Colaboración con el cliente sobre negociación contractual**: Los dueños de los negocios que usarán "Rhea" estarán involucrados durante el desarrollo para garantizar que el producto cumpla con sus expectativas.
4. **Respuesta ante el cambio sobre seguir un plan**: El equipo estará preparado para adaptarse a cambios en las necesidades del usuario o requisitos del negocio durante el desarrollo del proyecto.

# Personas y roles del proyecto.

|  |  |  |
| --- | --- | --- |
| **Persona** | **Contacto** | **Rol** |
| PRODUCT OWNER | Jesus Cortes Urriola  +56 975345101 | Responsable de identificar y priorizar las necesidades del cliente, además de asegurar que el equipo esté enfocado en brindar valor al cliente y alcanzar los objetivos comerciales. |
| SCRUM MASTER | Franco Fuentes Mellado  Fr.fuentesm@duocuc.cl | Encargado de administrar el proceso Scrum, superar obstáculos y garantizar que el equipo siga los principios y prácticas de Scrum. |
| SCRUM DEV 1 | Franco Fuentes Mellado | - |
| SCRUM DEV 2 | Miguel Garate Sanhueza  Mig.garate@duocuc.cl | Responsables de crear el producto. Esto incluye:   * Desarrollo y pruebas de funcionalidades del sistema según las historias de usuario. * Colaborar con el Product Owner para clarificar requisitos. * Desplegar el producto de manera incremental, asegurando que cada sprint entregue una versión funcional. * Solucionar problemas técnicos y optimizar el código.   Su principal objetivo es transformar los requisitos del Product Backlog en un producto funcional. |

1. **ORGANIZACIÓN ÁGIL DEL PROYECTO**

# Épicas a cubrir

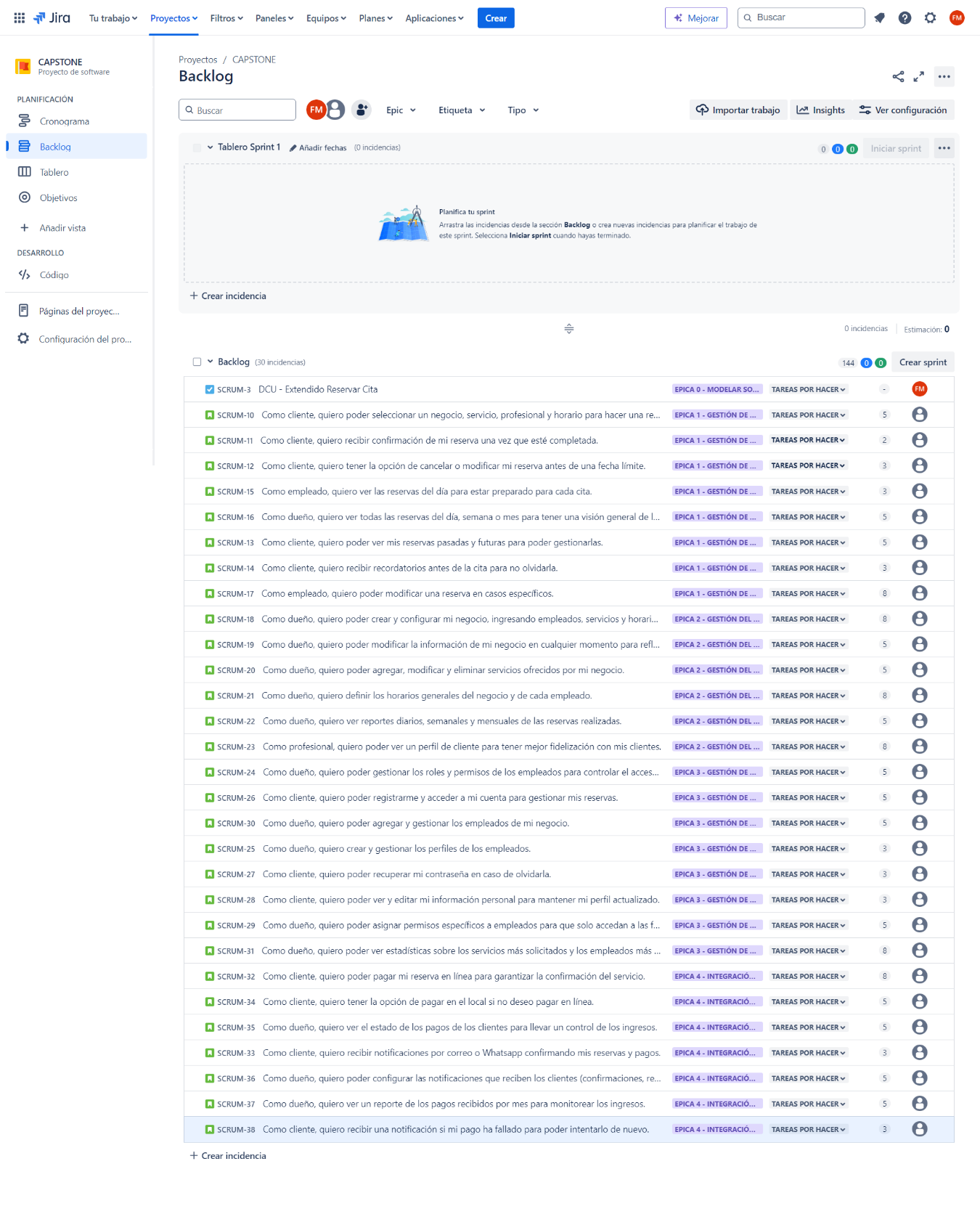
|  |  |  |
| --- | --- | --- |
|  | **NOMBRE** | **DECLARACIÓN DE EPICA** |
| E.0 | Iniciación del proyecto | Definición visión del producto y planificación del product backlog. Implementar ambiente de Desarrollo y producción. S1-S5 |
| E1 | Gestión de Reservas | Implementación del sistema de reservas para que los clientes puedan seleccionar un servicio, profesional y horario. |
| E2 | Gestión del Negocio | Los dueños podrán configurar su negocio, gestionando empleados, horarios y servicios. |
| E3 | Gestión de Usuarios y Roles | Gestión de clientes, empleados y permisos de acceso. Los profesionales podrán ver perfiles de clientes. |
| E4 | Integración de Pagos y Notificaciones | Los clientes podrán realizar pagos en línea y recibir notificaciones automáticas. |

# Perfil de Usuarios relacionados

A continuación, se detalla una tabla donde se relacionan los diferentes tipos de usuarios con las épicas en las que interactúan.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Usuario** | **Gestión de Reservas** | **Gestión del Negocio** | **Gestión de Usuarios y Roles** | **Integración de Pagos y Notificaciones** |
| *Cliente (El que reserva)* | ✔ |  | ✔ | ✔ |
| *Profesional (Empleado)* | ✔ |  | ✔ |  |
| *Dueño PYME (El que crea el negocio)* | ✔ | ✔ | ✔ | ✔ |

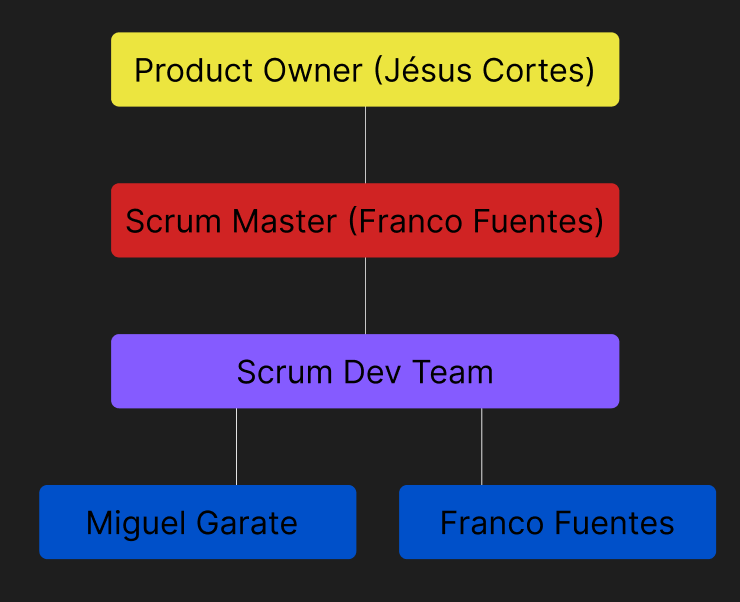
# Principales historias de usuarios por Épicas



# Definición de responsabilidades del Equipo por Rol definido

* ***Product Owner*** (Jesús): Responsable de definir la visión del producto y priorizar las historias de usuario en el backlog.
* ***Scrum Máster*** (Franco Fuentes): Facilita el proceso Scrum y asegura que el equipo siga las buenas prácticas ágiles.
* ***Scrum Dev Team*** (Miguel Garate / Franco Fuentes): Desarrollan las funcionalidades del sistema, realiza pruebas, y colabora con el Product Owner para clarificar los requisitos.

Organigrama del Equipo Scrum:



# Visión del producto

Nuestra visión es ofrecer una plataforma que satisfaga las necesidades de las pequeñas y medianas empresas, facilitando su gestión de reservas, pagos y empleados. Queremos aliviar la carga operativa de estos negocios, ayudándolos a simplificar y automatizar sus procesos, permitiéndoles concentrarse en ofrecer un mejor servicio a sus clientes, contribuyendo a su crecimiento sostenible y mejorar su relación con los clientes, impulsando su éxito en un entorno cada vez más digitalizado.

# Tecnologías de Desarrollo e Implementación

Las tecnologías que utilizaremos para el desarrollo e implementación de la solucion serán las siguientes:

|  |  |
| --- | --- |
| Front End | React JS |
| Back End | Node.js / Express.js |
| Notificaciones | Send Grid / Twilio Whatsapp API |
| Base de Datos | MySQL |
| Pagos | Transbank API |

# Componentes y Principales Artefactos por épicas

|  |  |  |  |
| --- | --- | --- | --- |
| N° | Épica | Componentes | Artefactos |
| E.0 | Iniciación del Proyecto | Definición de la visión y planificación | Modelamiento inicial del sistema, diagramas preliminares |
| E.1 | Gestión de Reservas | Frontend del cliente, Backend de reservas | Diagrama de Casos de Uso (DCU), MER, Diagrama de Actividades, DCU Extendido "Reservar Hora" |
| E.2 | Gestión del Negocio | Backend de configuración, Panel de reportes | Diagrama de Clases, Documento de Requisitos, Código fuente, Pruebas de Integración |
| E.3 | Gestión de Usuarios y Roles | Sistema de autenticación, Gestión de permisos | Diagrama de Actividades, Código fuente, Pruebas Funcionales |
| E.4 | Pagos y Notificaciones | API de pagos, Sistema de notificaciones | Diagrama de Secuencia, Especificación de la Integración de API, Código fuente, Pruebas de Aceptación |

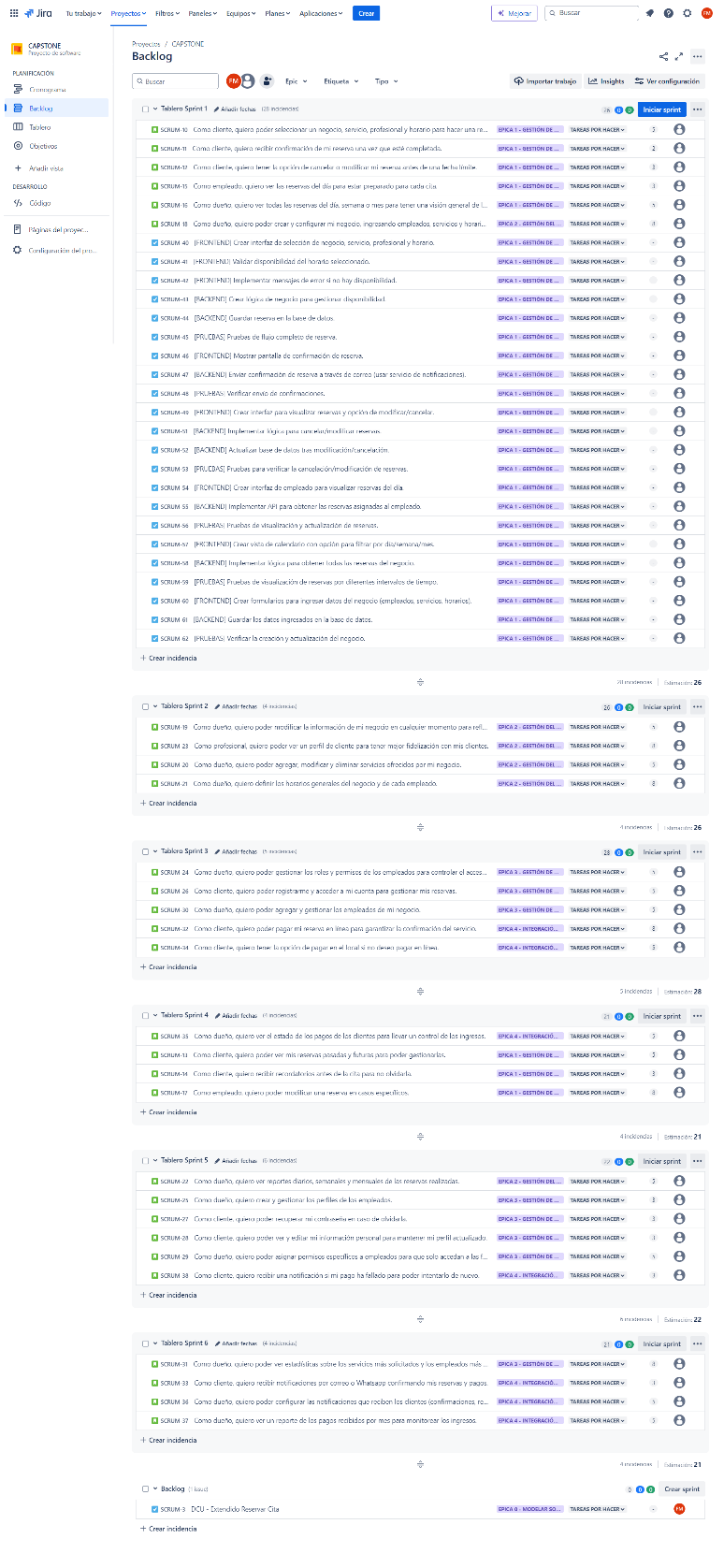
# Product Backlog

La priorización de componentes se ha realizado utilizando la técnica MoSCoW, para asegurar que las funcionalidades esenciales del sistema estén disponibles en las primeras entregas del proyecto.

Los **componentes Must Have** son aquellos relacionados con la gestión de reservas, configuración del negocio y autenticación de usuarios. Los **Should Have** incluyen funcionalidades importantes como la visualización de perfiles y el envío de notificaciones, mientras que los **Could Have** son funcionalidades que se desarrollarán si hay tiempo disponible, como los reportes avanzados y pagos en línea.

Esta priorización asegura que el sistema será funcional y útil para los usuarios desde las primeras fases de desarrollo.

*[Abajo se encuentra una imagen del product backlog priorizado de mayor a menor, dividido en épicas.]*



[Link Product Backlog](https://capstone-fuentes-garate.atlassian.net/jira/software/projects/SCRUM/boards/1/backlog?atlOrigin=eyJpIjoiNzhkMjBjNTc0ZWQxNDIzZjhjNWNlMDY1OWYwMmUzZGEiLCJwIjoiaiJ9)

# Definición del Done

Para cada componente definido como entregable en el Product Backlog, los criterios de aceptación serán los siguientes:

***Gestión de Reservas***: La funcionalidad permite a los clientes hacer, modificar y cancelar reservas. Las reservas se guardan correctamente en la base de datos y se reflejan en el calendario del profesional. El cliente recibe una confirmación de su reserva por correo electrónico.

***Gestión del Negocio***: Los dueños pueden agregar, modificar y eliminar servicios y empleados. Los horarios y servicios se reflejan correctamente en el sistema y son accesibles para los clientes. El panel de reportes muestra datos de reservas y empleados de forma clara.

***Gestión de Usuarios y Roles:*** Los clientes pueden registrarse, iniciar sesión y modificar su información personal. Los empleados pueden acceder a sus reservas y ver el perfil de los clientes. Los roles y permisos se gestionan correctamente y limitan el acceso a funcionalidades.

***Pagos y Notificaciones***: El cliente puede realizar pagos en línea con éxito y recibir notificaciones de confirmación. Las notificaciones se envían de forma automática para confirmar reservas, recordatorios y pagos. Los dueños pueden visualizar el estado de los pagos en su panel.

# RoadMap

A continuación, se detalla la secuencia de tareas por semana y componente, distribuidas a lo largo de los sprints:

Sprint 1: Gestión de Reservas

- Reserva de Servicios (Cliente):

Semana 1: Desarrollo del frontend para la selección de servicios y profesionales. Semana 2: Implementación del backend y pruebas de integración para reservas.

Sprint 2: Configuración del Negocio y Perfil de Clientes

- Modificación de Negocio (Dueño):

Semana 1: Desarrollo de la interfaz para modificar la información.

Semana 2: Pruebas y actualización de datos en el backend.

Sprint 3: Roles y Pagos

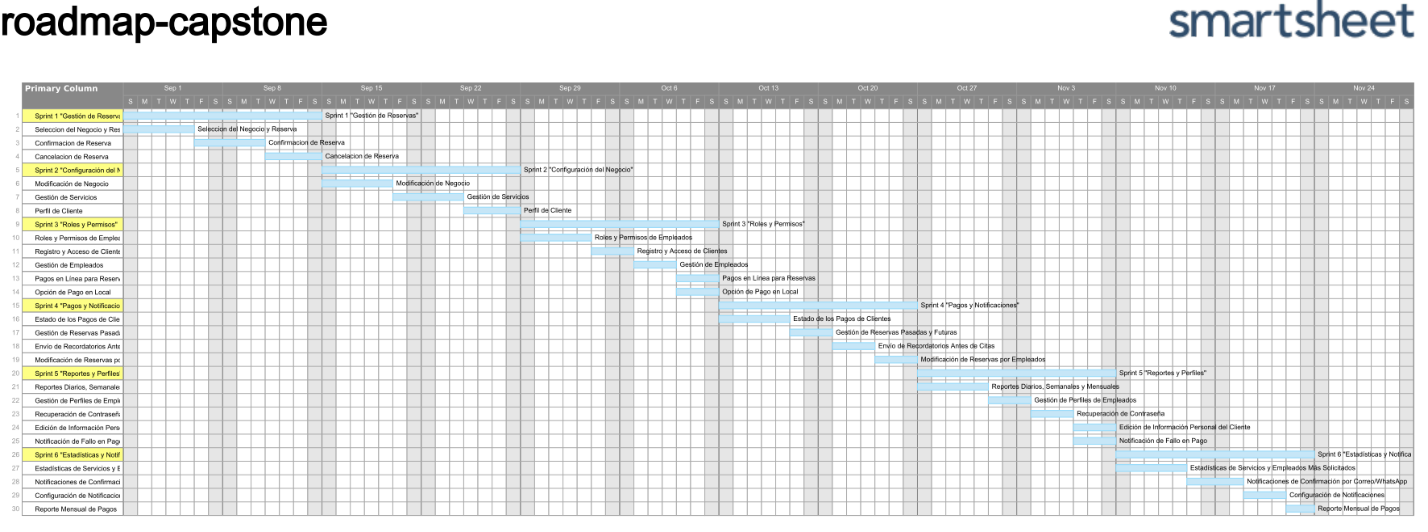
- Roles y Permisos:

Semana 1: Asignación de roles y permisos.

Semana 2: Pruebas y funcionalidad.

- Pagos en Linea

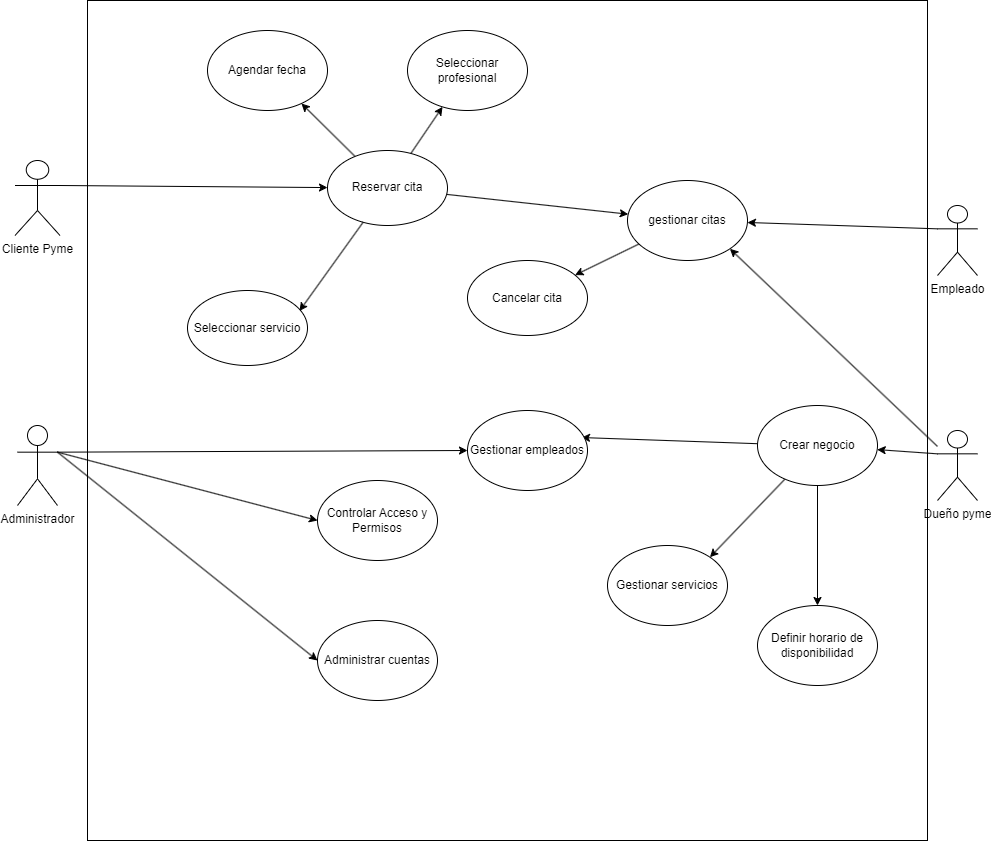
Semana 2: Integración del sistema de pagos.



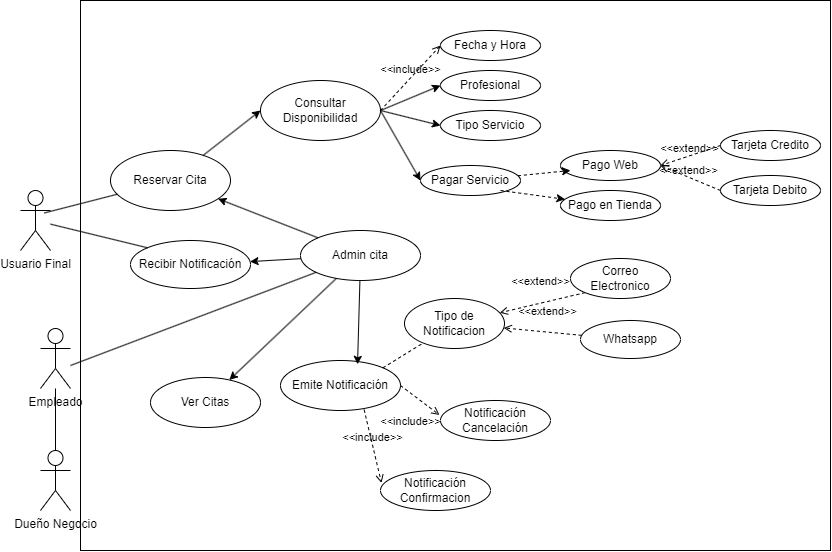
[[Link RoadMap Drive]](https://drive.google.com/file/d/1kUA9JIEYOrFcmdRo99ad0wzIsxVWM1hH/view?usp=drive_link)

# Diseño inicial del software

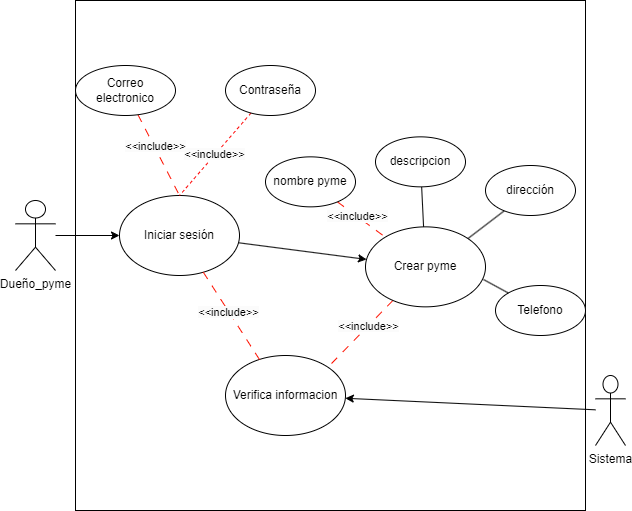
## Diagramas de Caso de Uso General



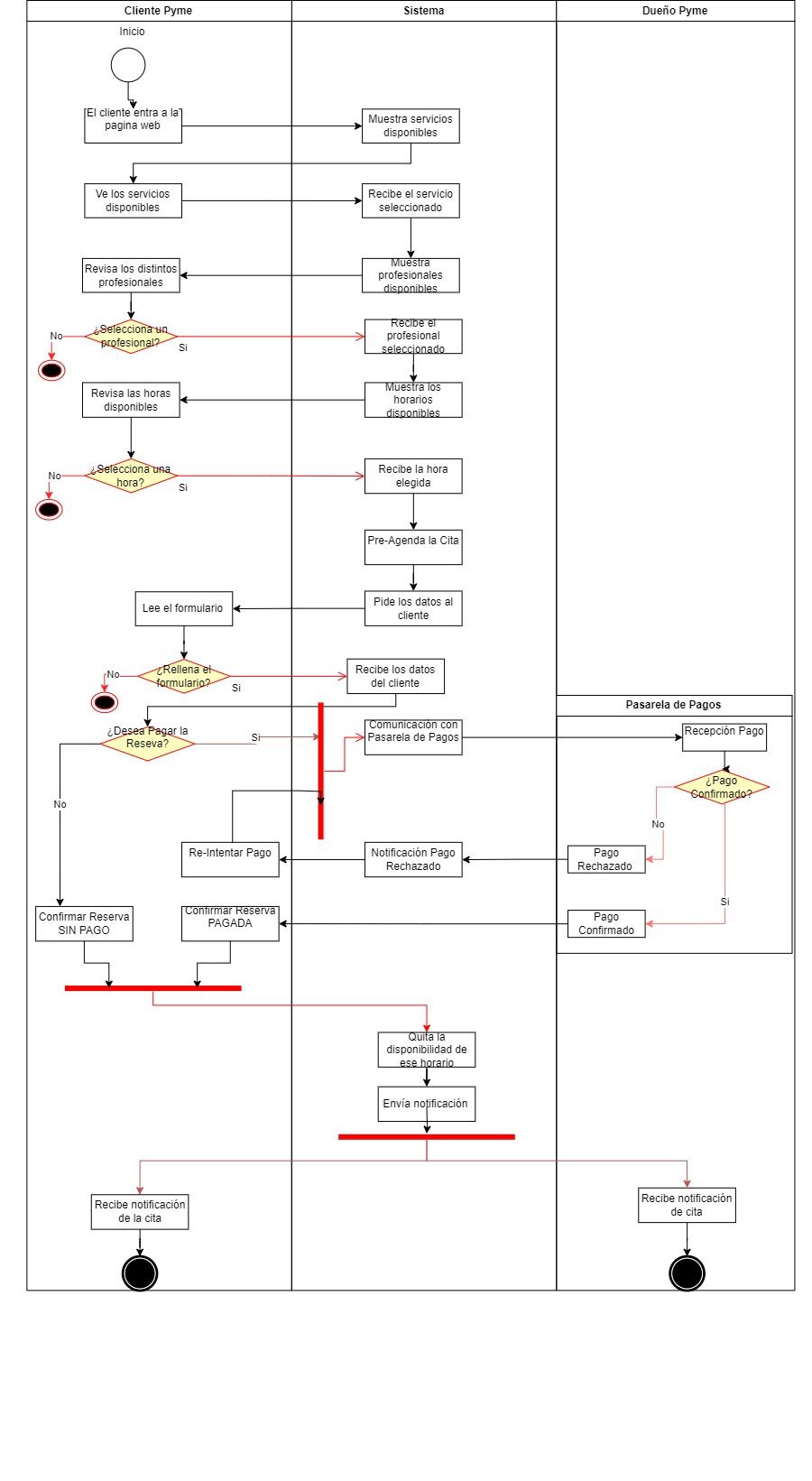
## Diagramas de Caso de Uso extendido(Reservar Cita)



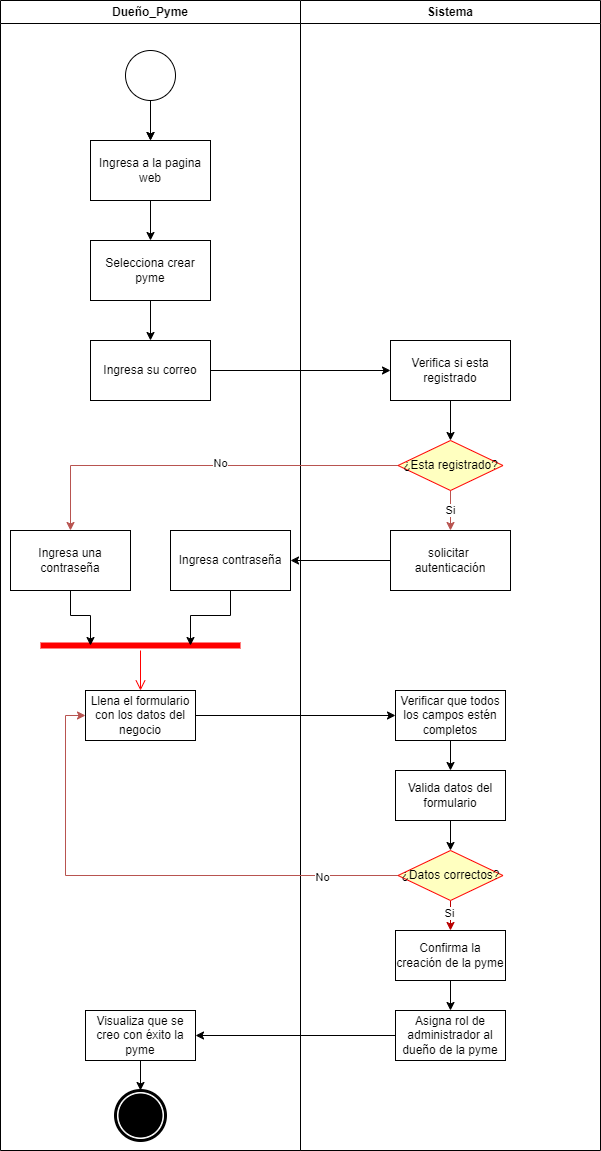
## Diagrama de Caso de Uso extendido(Crear pyme)



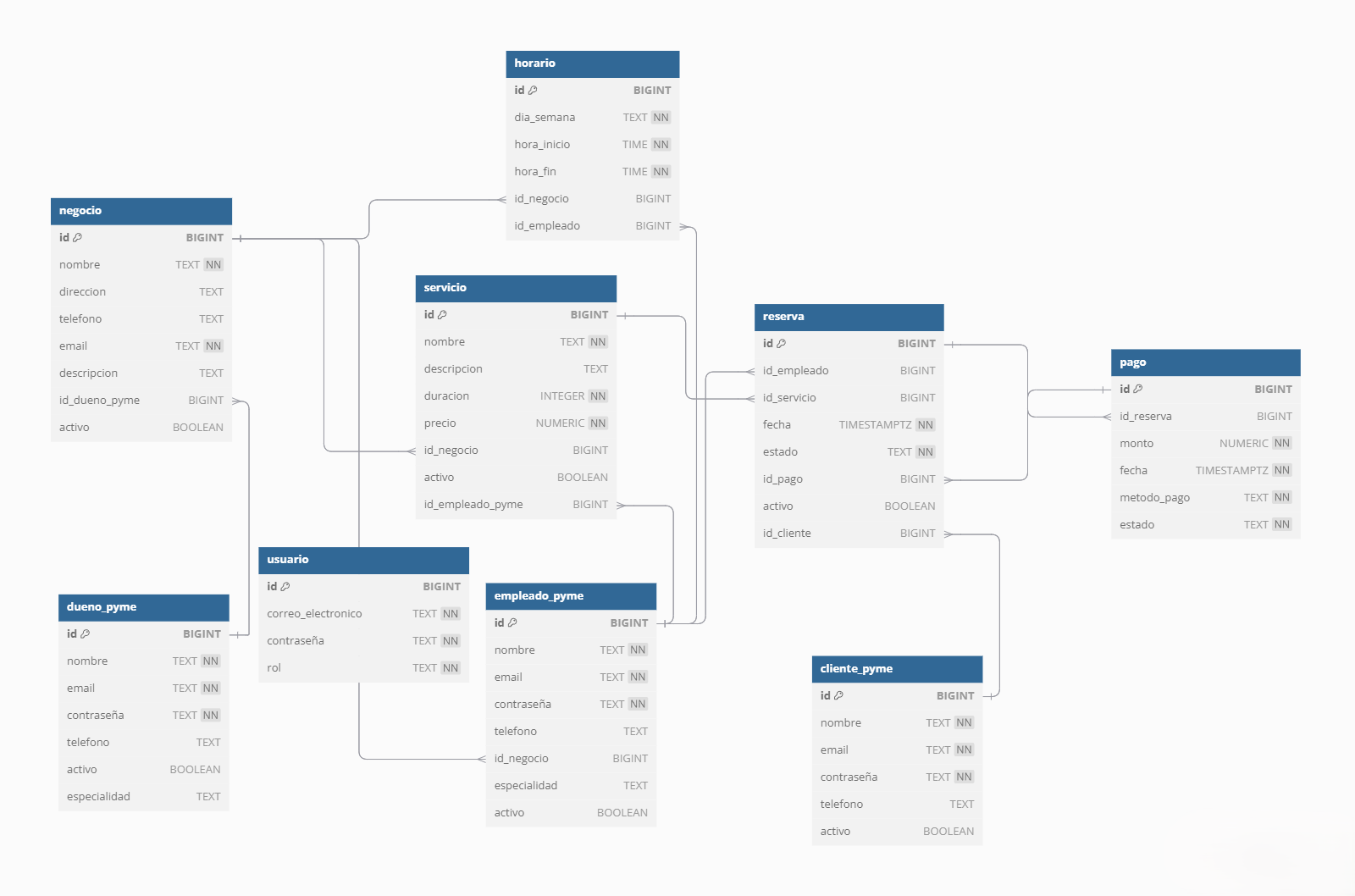
## Diagramas de Actividad (Reservar Cita)



## Diagramas de Actividad (Crear Pyme)



## Modelo de Datos (MER)



# Herramientas de gestión ágil

Descripción de las principales herramientas de gestión y administración del proyecto y los accesos a los recursos:

* **Carpeta Drive**: Para almacenar y compartir documentos importantes del proyecto de forma colaborativa.

<https://drive.google.com/drive/folders/1vVrGqnaKgpUdSvU60d9N1jqzlhGfho25?usp=sharing>

* **GitHub**: Para el control de versiones del código y la colaboración en el desarrollo del proyecto.

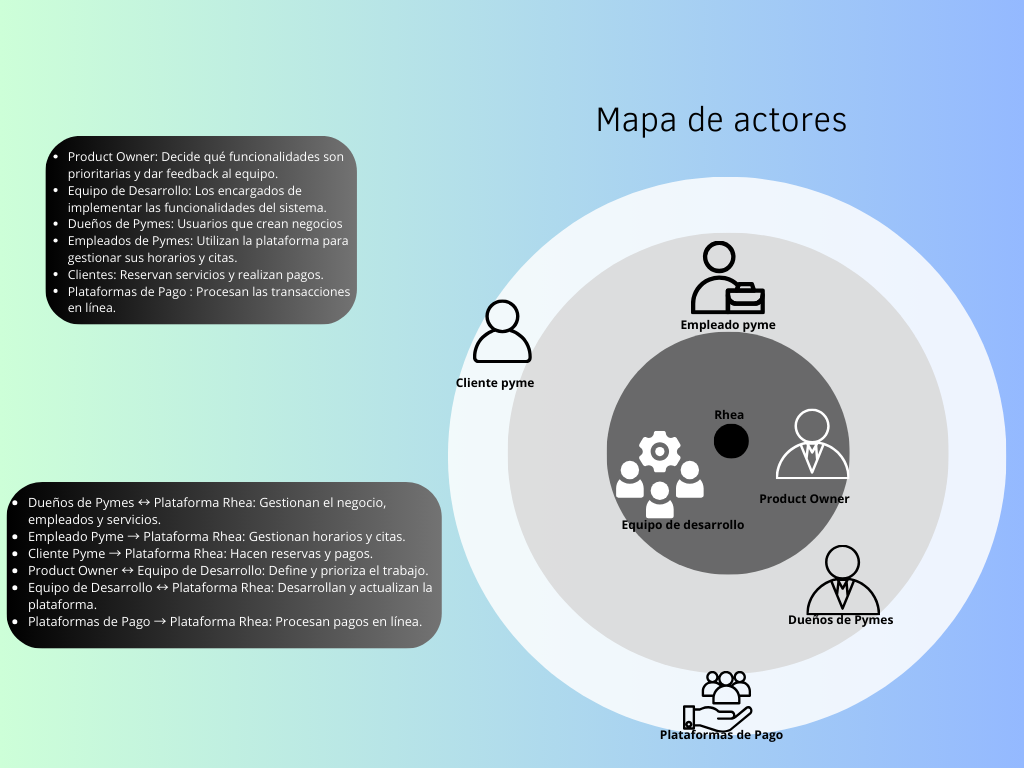
<https://github.com/FrxncoF/2024_2_PO_CAPSTONE_001D_GRUPO_001>

* **Tablero Jira**: Para gestionar tareas, organizar sprints y hacer seguimiento del progreso del equipo.

https://capstone-fuentes-garate.atlassian.net/jira/software/projects/SCRUM/boards/1?atlOrigin=eyJpIjoiY2U1MzgyN2IxODRiNDM3ZDljMjBhYjNlNzRmY2VlNzAiLCJwIjoiaiJ9

# Anexos.

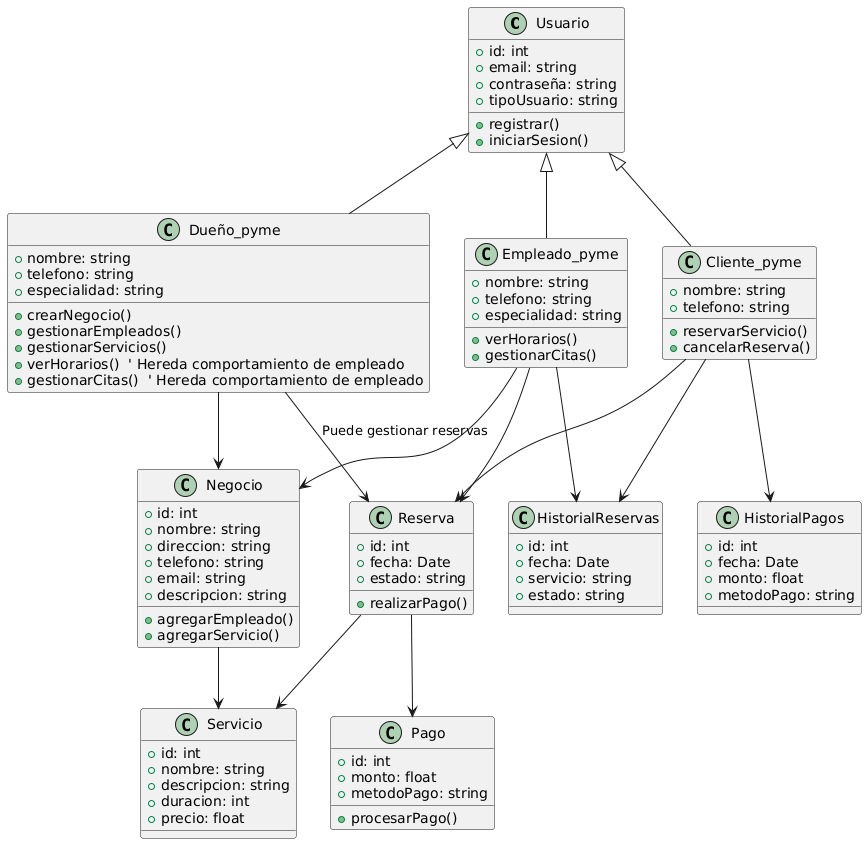
## Mapa de Actores



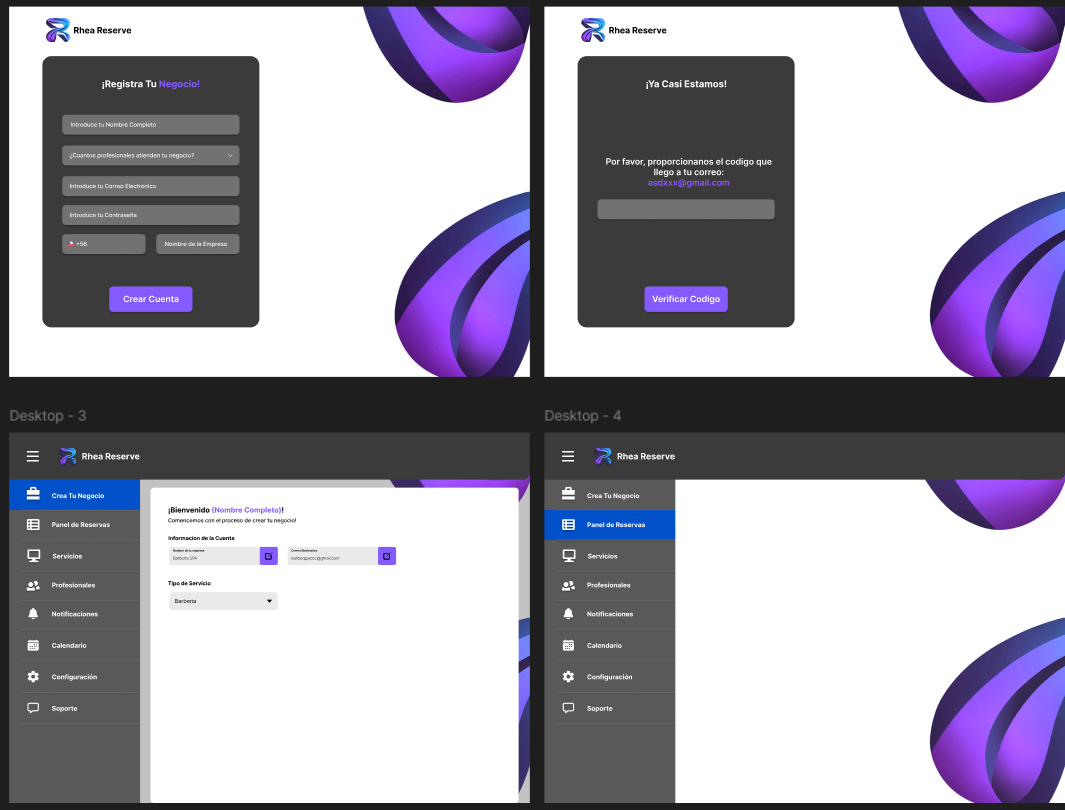
## Modelos UML de la solución

Diagrama

Descripción generada automáticamenteDiagrama de Clases



## Prototipado la solución



# Raci

