# **1.** **Introducción**

Este documento describe la implementación de la metodología de trabajo Scrum para el desarrollo del proyecto "Gestión de Reserva Hotelera" destinado a optimizar el proceso de reservas y administración de hospedajes en el Hotel Pacific Reef. Se busca mejorar la experiencia de los clientes y facilitar la gestión interna del hotel a través de una plataforma en línea.  
  
**1.1.1** **Propósito de este documento**

El propósito de este documento es facilitar la información de referencia necesaria a las personas involucradas en el desarrollo del **sistema HPR APP**.  
 Se detallará la ejecución del desarrollo del producto de software mediante un **ciclo de vida adaptativo e integrativo**, además de presentar:

● Las **épicas** e **historias de usuario**.

● Los **componentes y artefactos** a construir.

● Los **roles del equipo Scrum**, sus competencias técnicas y funciones en el proyecto.

**1.1.2** **Problemática que resolver**

### **Situación Actual**

El Hotel Pacific Reef gestiona sus reservas de manera manual, lo que ocasiona problemas como:

● Errores en la disponibilidad de habitaciones.

● Procesos de reserva lentos y poco eficientes.

● Falta de integración con pasarelas de pago.

● Dificultad para generar reportes y gestionar clientes.

### **Solución Propuesta**

Para abordar estos problemas, se desarrollará una plataforma digital que permita:

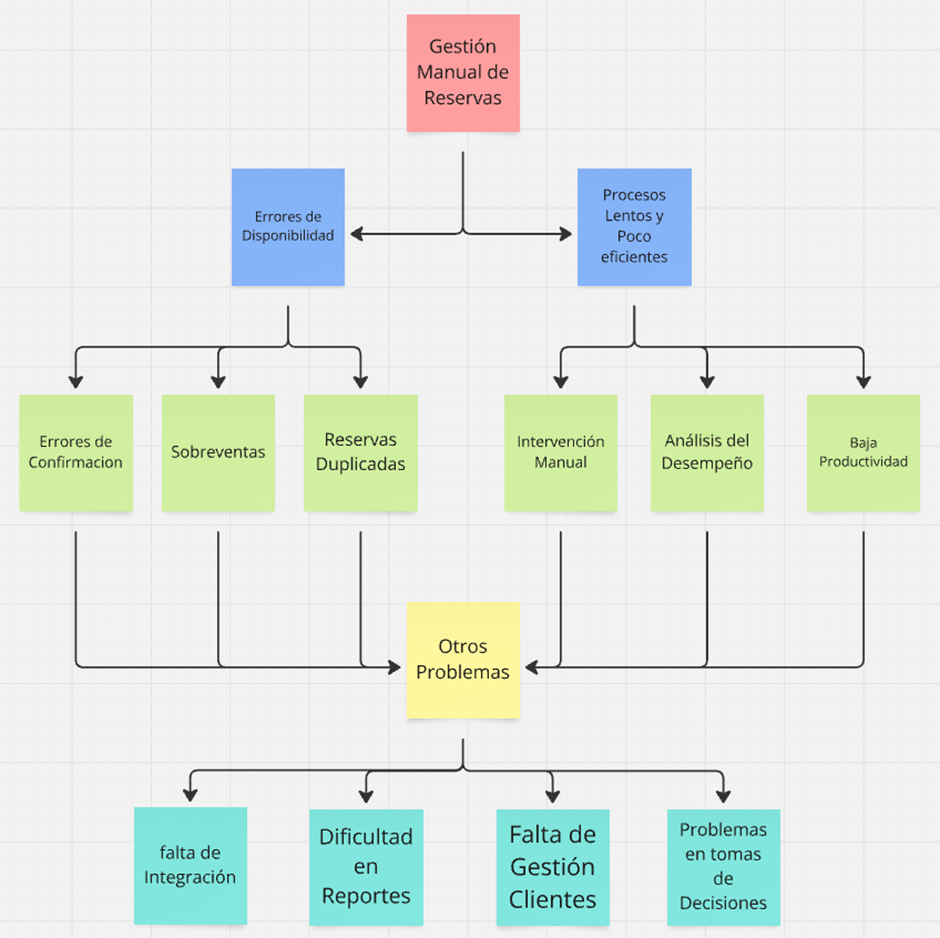
● Consultar disponibilidad y realizar reservas en línea.

● Gestionar pagos de manera automatizada.

● Proporcionar herramientas de administración para empleados y gerentes.

● Generar reportes en tiempo real.

El Hotel Pacific Reef enfrenta múltiples desafíos debido a la gestión manual de sus reservas, lo que genera ineficiencias operativas y una experiencia deficiente para los clientes. La falta de un sistema automatizado provoca errores en la disponibilidad de habitaciones, lo que puede llevar a sobreventas o reservas duplicadas. Además, los procesos manuales de reserva son lentos y requieren intervención constante del personal, lo que reduce la productividad y aumenta el tiempo de espera para los clientes. Otro problema crítico es la ausencia de integración con pasarelas de pago, lo que limita las opciones de cobro y puede generar demoras en la confirmación de reservas, afectando la confianza del cliente en el servicio del hotel.

Asimismo, la falta de herramientas digitales dificulta la generación de reportes y la gestión eficiente de los clientes, lo que impide tomar decisiones estratégicas basadas en datos en tiempo real. Sin un sistema automatizado, el hotel tiene dificultades para analizar patrones de ocupación, evaluar el desempeño del negocio y optimizar la asignación de recursos. Para solucionar estos problemas, es necesario implementar una plataforma digital que agilice las reservas, automatice los pagos, facilite la administración para empleados y gerentes, y proporcione reportes actualizados. Esto permitirá mejorar la eficiencia operativa, optimizar la experiencia del cliente y aumentar la competitividad del hotel en el mercado.  


## **1.1.3** **Objetivo del Proyecto**

Implementar y desarrollar una plataforma web y móvil que optimice la gestión de reservas en el Hotel Pacific Reef, permitiendo una experiencia intuitiva y eficiente para los usuarios y administradores.

# 

# **2.** **Descripción General de la Metodología ágil a adoptar.**

## **2.1.** **Fundamentación**

El proyecto seguirá un **ciclo de vida iterativo e incremental**, basado en la metodología Scrum. Se trabajará en sprints de **2 semanas**, entregando incrementos funcionales en cada iteración.

## **2.2.** **Valores de trabajo**

Para que el equipo Scrum pueda trabajar de manera efectiva en un entorno ágil, debe contar con un conjunto de competencias técnicas y habilidades blandas que le permitan adaptarse a los desafíos del proyecto.

#### **Competencias Técnicas**

El equipo de desarrollo debe poseer conocimientos y experiencia en:

● Desarrollo Frontend: Uso de frameworks modernos como React.js o Angular para la creación de interfaces de usuario responsivas y dinámicas.

● Desarrollo Backend: Implementación de APIs REST utilizando Node.js con Express o Django en Python para manejar la lógica de negocio y comunicación con la base de datos.

● Bases de Datos: Manejo de sistemas de gestión de bases de datos como MySQL o PostgreSQL, garantizando la eficiencia en la gestión de información.

● Seguridad en Aplicaciones Web: Implementación de estándares modernos de autenticación como OAuth 2.0 y JWT, además de encriptación de datos sensibles.

● Control de Versiones: Uso de Git y GitHub/GitLab para la gestión del código fuente, control de cambios y colaboración en equipo.

● Integración y Despliegue Continuo (CI/CD): Automatización de pruebas y despliegues en plataformas en la nube como AWS o Google Cloud.

#### **Habilidades Blandas**

Para trabajar en un entorno ágil y colaborativo, el equipo debe poseer las siguientes habilidades:

● Comunicación efectiva: Capacidad de compartir información de manera clara y concisa en reuniones diarias (Daily Scrum) y retrospectivas.

● Trabajo en equipo: Colaboración y apoyo mutuo para alcanzar los objetivos del sprint.

● Adaptabilidad: Flexibilidad ante cambios en los requerimientos del cliente y ajustes en las prioridades del backlog.

● Resolución de problemas: Habilidad para analizar situaciones y proponer soluciones rápidas y eficientes.

● Gestión del tiempo: Organización de tareas dentro del sprint para cumplir con los entregables dentro del tiempo estipulado.

● Proactividad: Capacidad de tomar iniciativa para mejorar procesos y optimizar el desarrollo del proyecto.

# **3.** **Personas y roles del proyecto.**

| **Persona** | **Rol** | **Función** |
| --- | --- | --- |
| Hotel Pacific Reef | Stakeholder/s | Define requerimientos y valida avances |
| Jorge Rivera Toloza | Product Owner | Prioriza el backlog y asegura que el producto cumpla con los requerimientos |
| Jorge Canales Soto | Scrum Master | Facilita el proceso Scrum, elimina obstáculos y asegura la metodología |
| Matías Jorquera Valdés | Developer 1 | Implementa la lógica de negocio y la API |
| Manuel Soto Lagos | Developer 2 | Desarrolla la interfaz de usuario. |
| Jorge Frick Mendez | Developer 3 | Prueba la funcionalidad y reporta errores. |

# **4.** **Product Backlog. Lista de Componentes y artefactos para Construir.**

**4.1 Épicas e historias de usuarios.**

## 

## El Product Backlog para el proyecto de Gestión de Reserva Hotelera del Hotel Pacific Reef define las principales funcionalidades necesarias para desarrollar un sistema eficiente y ágil. El backlog está estructurado en historias de usuario, organizadas en iteraciones (Sprints), considerando la prioridad, el esfuerzo y la complejidad de cada tarea.

### Usuarios del sistema

## El sistema está diseñado para ser utilizado por los siguientes actores:

## ● Clientes: Personas que desean reservar habitaciones, consultar disponibilidad y gestionar su estadía.

## 

## ● Administrador del hotel: Responsable de gestionar disponibilidad, precios y reportes del hotel.

## 

## ● Empleados del hotel: Consultan reservas, disponibilidad de habitaciones y preparan las habitaciones.

## 

## ● Asesor tecnológico: Supervisa el desarrollo del sistema y valida su correcto funcionamiento.

## 

### Principales funcionalidades

## El backlog contiene historias de usuario agrupadas en distintas épicas, abordando aspectos clave como:

## 1. Gestión de reservas: Permite a los clientes reservar habitaciones en línea y recibir confirmaciones automáticas.

## 

## 2. Pago en línea: Integración de una pasarela de pago para asegurar transacciones rápidas y seguras.

## 

## 3. Administración de tarifas y disponibilidad: El administrador puede modificar precios y gestionar la ocupación del hotel.

## 

## 4. Visualización y consulta de habitaciones: Los clientes pueden explorar el catálogo de habitaciones con imágenes y detalles de cada una.

## 

## 5. Gestión de empleados y procesos internos: Los empleados pueden visualizar reservas y organizar la limpieza de las habitaciones.

## 

## 6. Reportes y supervisión: El sistema genera reportes de ocupación y estadísticas para ayudar en la toma de decisiones.

## 

### Iteraciones y prioridades

## El desarrollo del sistema se dividirá en tres sprints, priorizando las funciones esenciales para una primera versión funcional en ambiente de prueba.

## ● Sprint 1: Reserva en línea, confirmaciones por correo, gestión de disponibilidad y acceso al panel administrativo.

## 

## ● Sprint 2: Implementación de pagos en línea, catálogo de habitaciones y consulta de disponibilidad por parte de los empleados.

## 

## ● Sprint 3: Generación de reportes, ajuste de tarifas por el administrador y supervisión del sistema por parte del asesor tecnológico.

## 

## El objetivo final es entregar un prototipo funcional en nueve semanas, asegurando que el sistema esté preparado para pruebas y validaciones antes de su implementación definitiva.

|  | **Enunciado de la Historia (Usuarios Clientes)** | **USUARIO** | **Artefacto** |
| --- | --- | --- | --- |
| 1 | Como cliente, quiero poder reservar una habitación en línea para asegurar mi estadía. | Cliente | Plataforma Web |
| 2 | Como administrador, quiero gestionar las reservas desde un panel de control para administrar la ocupación. | Cliente | Plataforma Web |
| 3 | Como cliente, quiero recibir una confirmación por correo electrónico después de reservar para tener certeza de mi reserva. | Cliente | Plataforma Web |
| 4 | Como empleado del hotel, quiero poder visualizar las reservas en un calendario para organizar mejor la recepción. | Administrador del hotel | Panel de Administración |
| 5 | Como cliente, quiero poder pagar mi reserva en línea para evitar pagos presenciales. | Cliente | Plataforma Web |
| 6 | Como administrador, quiero poder ver un informe de ocupación semanal para evaluar la demanda del hotel. | Empleado del hotel | Panel de Administración |
| 7 | Como cliente, quiero poder seleccionar la cantidad de días de hospedaje para planificar mejor mi estadía. | Cliente | Pasarela de Pago |
| 8 | Como cliente, quiero poder ver fotos y detalles de cada habitación antes de reservar para tomar una mejor decisión. | Empleado del hotel | Panel de Administración |
| 9 | Como asesor tecnológico, quiero poder acceder a reportes de uso del sistema para evaluar su rendimiento. | Administrador del hotel | Panel de Administración |
| 10 | Como administrador, quiero poder modificar los precios de las habitaciones para ajustarlos a la temporada. | Asesor tecnológico | Plataforma Web |
| 11 | Como empleado del hotel, quiero poder consultar la disponibilidad de habitaciones para coordinar la limpieza y preparación. | Administrador del hotel | Panel de Administración |

| **Componente** | **PTS por Historia Complejidad** | **Esfuerzo**  **Tiempo/Días** | **Duración Sprint/Semanas** | **Iteración (Sprint)** | **Prioridad** | **Estado** |
| --- | --- | --- | --- | --- | --- | --- |
| Notificaciones | 3 | 4 | 2 | Sprint 1 | Media | Pendiente |
| Reservas | 4 | 6 | 2 | Sprint 1 | Alta | Pendiente |
| Reservas | 5 | 6 | 2 | Sprint 1 | Alta | Pendiente |
| Gestión de Reservas | 8 | 10 | 2 | Sprint 1 | Alta | Pendiente |
| Catálogo de Habitaciones | 5 | 6 | 2 | Sprint 2 | Alta | Pendiente |
| Gestión de Reservas | 5 | 6 | 2 | Sprint 2 | Media | Pendiente |
| Pagos | 8 | 10 | 2 | Sprint 2 | Alta | Pendiente |
| Gestión de Reservas | 5 | 6 | 2 | Sprint 2 | Media | Pendiente |
| Reportes | 6 | 8 | 2 | Sprint 3 | Media | Pendiente |
| Supervisión y Validación | 6 | 8 | 2 | Sprint 3 | Media | Pendiente |
| Gestión de Tarifas | 7 | 8 | 2 | Sprint 3 | Alta | Pendiente |

# **5.** **Definición tecnologías de Desarrollo a utilizar**

Para el desarrollo del sistema, se utilizarán las siguientes herramientas y tecnologías:

| Tecnología | Versión | Propósito |
| --- | --- | --- |
| React.js / Angular | Última estable | Desarrollo frontend responsivo. |
| Node.js (Express) / Django | Última estable | Desarrollo backend y API. |
| MySQL / PostgreSQL | Última estable | Almacenamiento estructurado. |
| OAuth 2.0 / JWT | Última estable | Seguridad y autenticación. |

### **Justificación de la Selección de Tecnologías**

Se seleccionaron estas tecnologías debido a las siguientes razones:

-Escalabilidad: Permiten la expansión futura del sistema sin afectar el rendimiento.

-Seguridad: Implementan estándares modernos de autenticación y encriptación.

-Interoperabilidad: Facilitan la integración con otros servicios y APIs.

-Facilidad de Desarrollo: Frameworks ampliamente documentados y con gran soporte comunitario.

-Compatibilidad Multiplataforma: Soporta múltiples dispositivos y sistemas operativos.

Describir las herramientas de desarrollo definidas para construir el sistema y sus versiones. Además, justificar el valor de desarrollar con ese tipo de tecnologías.

# **6.** **Definición de tecnologías de implementación.**

Para garantizar una infraestructura robusta y escalable, se utilizarán las siguientes tecnologías y servicios:

| Tecnología | Uso |
| --- | --- |
| AWS EC2 / Google Cloud Compute Engine | Servidor de aplicaciones para alojar la plataforma. |
| AWS RDS / Google Cloud SQL | Base de datos gestionada en la nube para almacenamiento de información. |
| AWS S3 / Google Cloud Storage | Almacenamiento de archivos, imágenes y documentos. |
| Stripe / PayPal | Pasarela de pago segura para transacciones en línea. |
| Firebase / Auth0 | Servicio de autenticación y seguridad de usuarios. |
| Cloudflare / AWS WAF | Protección contra ataques DDoS y optimización de carga. |

### **Justificación de la Selección de Infraestructura**

-Alta Disponibilidad: Los servicios en la nube garantizan operación 24/7 sin interrupciones.

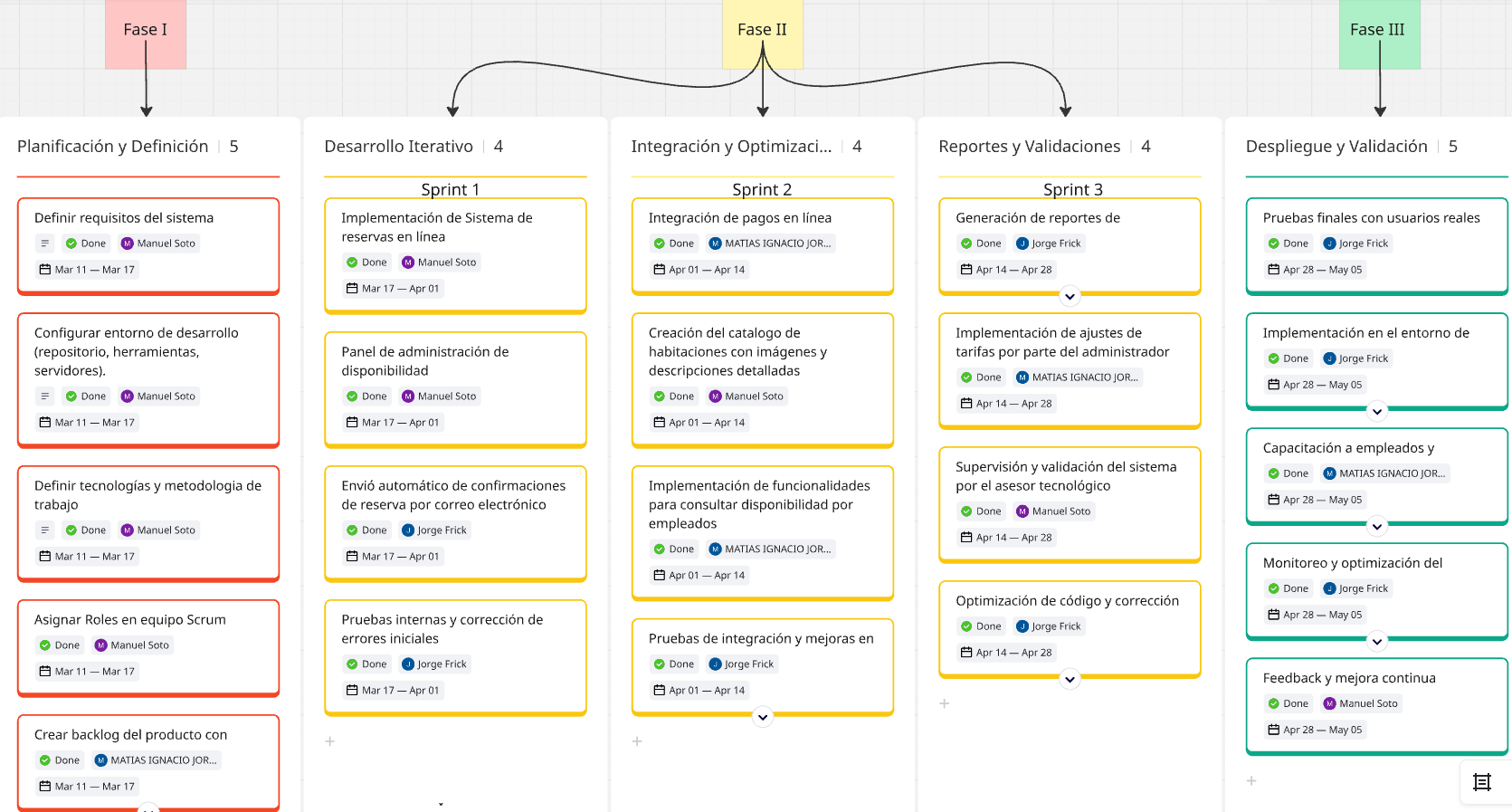
-Escalabilidad: La infraestructura permite aumentar o reducir recursos según la demanda del sistema.

-Seguridad: Se utilizan tecnologías con cifrado de datos, control de acceso y mitigación de amenazas cibernéticas.

Optimización de Costos: Se eligen servicios gestionados para reducir costos operativos y de mantenimiento.

**Roadmap**

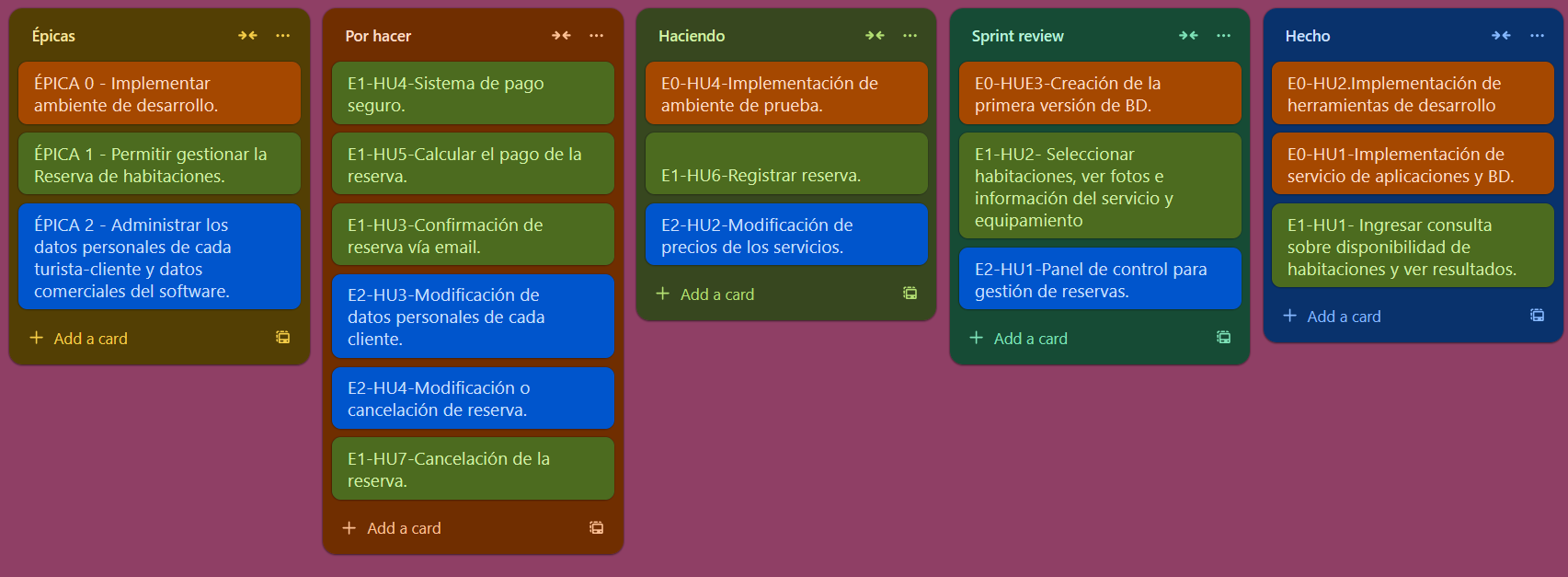
1. Adjunta la imagen de tu Roadmap:



2. Adjunta el link de acceso a archivo original guardado en el repositorio del proyecto:

<https://github.com/ManuelSotoL/Ing.-en-Software-Duoc-UC>

3. Adjunta el link de acceso a tu tablero Trello:

<https://trello.com/invite/b/67e53dc5e759afe4831994c8/ATTIb1d53bfc762644a53d900b791574bdfb7FCA33CF/ing-de-software>  
  


4. Adjunta el link de acceso al drive de tu proyecto:

<https://docs.google.com/document/d/1YybHo1RLrFUFcitxviD6eUteMz5LVODp6pBhfZoghFc/edit?tab=t.0>

**Parte II: Presentación**

<https://youtu.be/MebyQ7KDH9w>