



Carrera

**Analista Programador
Computacional**

Semana 3

Ingeniería de Software (PRY3211)

Formato de respuesta

| | |
|--|--|
| Nombre estudiante: Camilo Arriagada, Pamela Azúa, Katherin Gastelumendi | |
| Asignatura: : Ingeniería de Software | Carrera: Analista Programador Computacional |
| Profesor: Patricio Aranedi Oliva | Fecha: 26/08/2024 |

Descripción de la actividad

En esta semana realizarás la actividad sumativa grupal con encargo de la Experiencia 1, llamada "Definiendo la visión del producto inicial bajo una propuesta de planificación ágil", la cual estará dividida en dos partes. En la primera parte, entregaran de manera grupal la propuesta del tipo de software a desarrollar con un documento Project Burndown basado en la lista de Requisitos del Sistema y aportando un **Diagrama de Caso de Usos** para representar la vista de escenario principal. Además, deberán aportar una **Planificación ágil Scrum** basada en Scrum basado en una Planilla Product Backlog con principales épicas e historias de usuarios y sus sprint, aportando una Planilla de **Planificación del tiempo RoadMap** distribuidas en semanas y la implementación de la herramienta Trello para la organización de principales tareas y entregables.

En la segunda parte, realizarán una presentación a través de un video de no más de 7 minutos, usando la herramienta de Teams.

En la tercera parte y de manera individual, deberás escribir una reflexión personal sobre los aprendizajes obtenidos en la Experiencia 1; cómo estos impactarán en tu desarrollo personal e influirán en el ámbito profesional y consecución de tus metas.

Documento Project Burndown (versión 2)

Sistema RoomBooker

Descripción de la metodología de trabajo (Scrum)

Versión 1.0

Historial de Revisiones

| Fecha | Versión | Descripción | Autores |
|------------|---------|---|---------|
| 12/08/2024 | 1.0 | Primera versión definición de la Visión del Proyecto con los apartados y contenidos asociados | |
| 19/08/2024 | 1.1 | Primera versión de project Burndown para Sistema Roombooker | |
| | | | |

Tabla de Contenidos

Contenido

| | |
|---|----|
| 1. Introducción | 3 |
| 1.1.1 Propósito de este documento | 4 |
| 1.1.2 Problemática que resolver | 4 |
| 1.1.3 Objetivo del Proyecto | 4 |
| 1.1.4 Alcances | 5 |
| 2. Descripción General de la Metodología ágil a adoptar. | 5 |
| 2.1. Fundamentación | 6 |
| 2.2. Valores de trabajo | 6 |
| 3. Personas y roles del proyecto. | 6 |
| 4. Product Backlog. Lista de Componentes y artefactos para Construir. | 7 |
| 4.1 Épicas e historias de usuarios. | 8 |
| 5. Definición del Done | 9 |
| 5. Definición tecnologías de Desarrollo a utilizar | 9 |
| 7. Descripción de herramientas de gestión en modalidad Scrum. | 11 |

1. Introducción

Este documento describe la implementación de la metodología de trabajo Scrum para el desarrollo del proyecto RoomBooker destinado a HOTEL PACIFIC REEF. Este proyecto tiene como objetivo resolver una serie de desafíos relacionados con la gestión de reservas y la experiencia del cliente en el proceso de hospedaje. HAPPY CODERS, la empresa encargada del desarrollo, se compromete a entregar un producto de software que no solo cumpla con los requisitos técnicos y funcionales, sino que también mejore significativamente la eficiencia operativa y la satisfacción del cliente.

1.1.1 Propósito de este documento

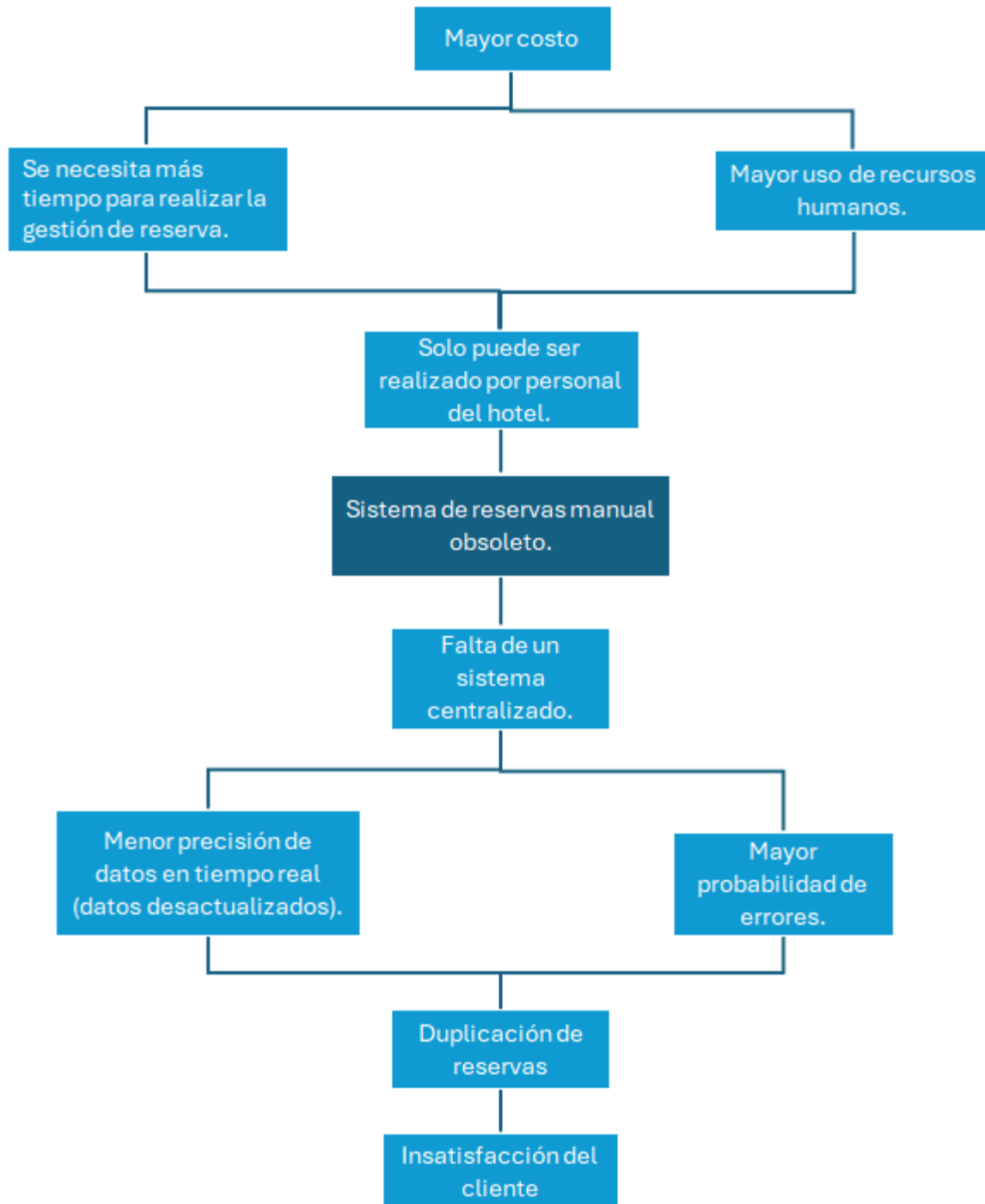
El propósito de este documento es facilitar la información de referencia necesaria a las personas involucradas en el desarrollo del sistema RoomBooker. Se explicará cómo se ejecutará el desarrollo del producto de software utilizando un ciclo de vida adaptativo e integrativo, siguiendo la metodología Scrum. Además, se presentarán las Épicas y sus Historias de Usuarios, junto con los componentes y artefactos que se construirán. Se detallarán los roles dentro del equipo Scrum, especificando las competencias técnicas necesarias y las funciones que cada rol desempeñará en el proyecto.

1.1.2 Problemática que resolver

El HOTEL PACIFIC REEF enfrenta varios desafíos en la gestión de sus reservas y la administración general de su operación hotelera. Actualmente, el proceso de reserva es manual y requiere de una gran cantidad de tiempo y esfuerzo tanto por parte del personal como de los clientes. Esto no solo reduce la eficiencia del hotel, sino que también puede generar errores en la asignación de habitaciones y en la gestión de pagos, afectando negativamente la experiencia del cliente. Además, la falta de un sistema centralizado de gestión de reservas dificulta la toma de decisiones basada en

datos. Sin un sistema adecuado, es complicado para la administración obtener informes precisos y en tiempo real sobre la ocupación del hotel, las tendencias de reservas y la rotación de habitaciones. Esta falta de visibilidad impide que el hotel optimice sus operaciones y mejore la satisfacción del cliente, especialmente en un entorno tan competitivo como el de la industria hotelera.

Mapa del Problema



1.1.3 Objetivo del Proyecto

Desarrollar un sistema de gestión de reservas hoteleras para HOTEL PACIFIC REEF que optimice el proceso de reserva y registro de hospedaje, permitiendo una gestión eficiente de las habitaciones disponibles y mejorando la experiencia del cliente a

través de un sistema intuitivo y accesible tanto para turistas como para el personal del hotel. Este sistema permitirá automatizar la consulta de disponibilidad, facilitar el registro de reservas, calcular y procesar pagos, y generar reportes detallados sobre la ocupación y gestión hotelera.

1.1.4 Alcances

Limitaciones del Sistema:

- **Idiomas Soportados:** El sistema estará disponible en inglés y español únicamente.
- **Plataformas:** La aplicación será accesible mediante dispositivos móviles y computadoras de escritorio, pero no se desarrollarán aplicaciones nativas específicas para iOS o Android en esta fase.
- **Métodos de Pago:** Solo se implementarán métodos de pago en línea comunes como tarjetas de crédito y débito. Otros métodos de pago, como transferencias bancarias o pagos en efectivo, no serán soportados en esta versión.
- **Integración con Sistemas Externos:** No se realizará integración con otros sistemas de gestión hotelera existentes en esta fase del proyecto.

Restricciones en la Implementación Futura:

- **Capacidad de Escalabilidad:** La versión inicial del sistema está diseñada para manejar la capacidad actual del HOTEL PACIFIC REEF (38 habitaciones). La escalabilidad para soportar un mayor número de habitaciones o múltiples establecimientos requerirá una futura reingeniería del sistema.
- **Funcionalidades Avanzadas:** Funcionalidades como la integración con sistemas de marketing digital, personalización avanzada para clientes VIP, o soporte para múltiples monedas, no están contempladas en la versión inicial y se considerarán en futuras fases de desarrollo.
- **Soporte Multilenguaje:** Cualquier expansión para incluir otros idiomas o configuraciones culturales será parte de fases de implementación futuras.

2. Descripción General de la Metodología ágil a adoptar.

2.1. Fundamentación

El ciclo de vida adoptado para este proyecto es **iterativo e incremental**. Este enfoque permite la construcción del sistema a través de múltiples iteraciones, cada una de las cuales agrega nuevas funcionalidades al producto, con base en las prioridades establecidas por el Product Owner. Al final de cada Sprint, se entregará un incremento funcional del software, que será revisado y evaluado por el equipo y los stakeholders.

Sprints Cortos: Cada Sprint tendrá una duración de 2 semanas, permitiendo revisiones frecuentes y la posibilidad de realizar ajustes rápidos según los comentarios del cliente.

Revisión y Retroalimentación Continua: Al final de cada Sprint, se presentará un incremento del producto al cliente y stakeholders para obtener retroalimentación inmediata, lo que permitirá ajustar el desarrollo en función de las necesidades reales del negocio.

Flexibilidad en los Requisitos: Dado que los requisitos pueden evolucionar durante el proyecto, Scrum permite la re-priorización de las tareas en el backlog para asegurarse de que se trabaje en las funcionalidades más importantes primero.

Transparencia y Colaboración: Scrum promueve la transparencia en el proceso de desarrollo, con reuniones diarias (Daily Standups) donde el equipo discute el progreso, identifica obstáculos, y coordina los próximos pasos.

2.2. Valores de trabajo

Describir las competencias técnicas que deberá poseer el equipo Scrum y las habilidades blandas que debe poseer el equipo para trabajar en esta modalidad (ágil) de manera efectiva.

3. Personas y roles del proyecto.

| Persona | Rol | Función |
|------------------------|---------------|---------------------------|
| Juan Caro | Stakeholder/s | Validación técnica. |
| Joseph Bruna | Product Owner | Definición estratégica. |
| Martina Blaster | Scrum Master | Facilitación ágil. |
| Camilo Arriagada | Developer 1 | Desarrollador Back end. |
| Pamela Azua | Developer 2 | Desarrolladora Front end. |
| Katherine Gastelumendi | Developer 3 | Desarrolladora Front end. |

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

4.1 Épicas e historias de usuarios.

- Tabla resumen con Principales Épicas e Historias de Usuarios v/s Componentes.

[Insertar Tabla Product Backlog ampliada a Sprintbacklog y describir]

(Insertar link de archivo planilla ubicado en su repositorio)

5. Definición tecnologías de Desarrollo a utilizar

Herramientas de Desarrollo:

- **NEXT.js:** Usaremos NEXT.js para manejar tanto el front-end como el back-end. Es una herramienta poderosa que combina React para construir interfaces de usuario

con Express para manejar la lógica del servidor. Esto nos permite desarrollar la aplicación de manera rápida y eficiente, todo en un solo marco de trabajo.

- **React:** En el front-end, React nos ayuda a crear una interfaz de usuario fluida y rápida. Es ideal para hacer que la aplicación sea interactiva y fácil de usar, y sus componentes reutilizables nos permiten desarrollar más rápido.
- **Express:** En el back-end, Express se encargará de gestionar las solicitudes y la lógica del servidor. Es simple, flexible y perfecto para desarrollar APIs que se conectan con el front-end.
- **PostgreSQL:** Elegimos PostgreSQL como nuestra base de datos porque es robusta y confiable. Maneja grandes cantidades de datos de forma segura, lo cual es crucial para gestionar reservas y detalles de clientes sin problemas.

Usar **NEXT.js** nos permite unificar el desarrollo del front-end y el back-end, lo que simplifica el proceso y mejora la eficiencia. Con React, aseguramos que la aplicación sea rápida y responsiva, mientras que Express nos proporciona un back-end flexible y ágil.

PostgreSQL es nuestra elección para la base de datos porque es capaz de manejar datos de manera segura y eficiente, lo cual es clave para un sistema de reservas en un hotel. En resumen, estas tecnologías juntas nos permiten crear una aplicación moderna, confiable y fácil de usar.

6. Definición de tecnologías de implementación.

DigitalOcean: Vamos a usar DigitalOcean para alojar y desplegar la aplicación. Es una plataforma en la nube que nos permite configurar y manejar servidores de manera sencilla y rápida.

¿Por qué?

Fácil de Usar: Su interfaz es amigable y fácil de configurar, lo que nos ahorra tiempo y esfuerzo.

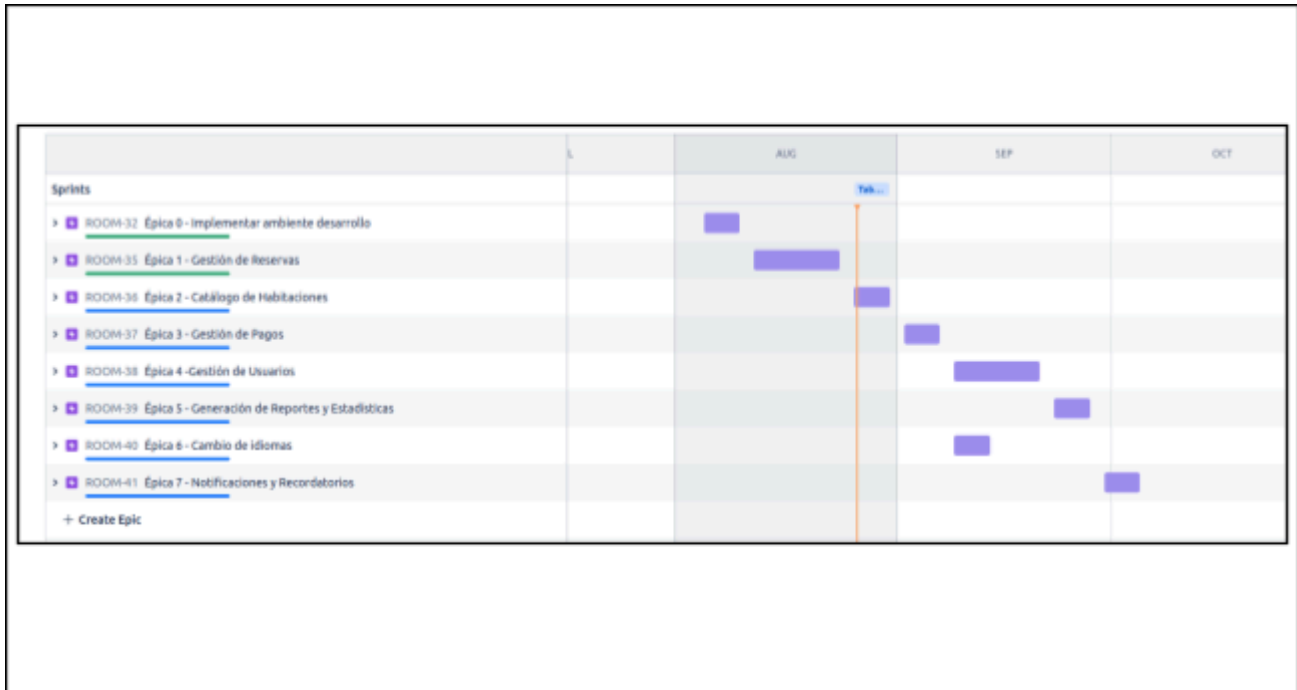
Escalable: Podemos ajustar los recursos según lo que necesitemos, asegurando que la aplicación funcione bien incluso si el tráfico aumenta.

Costo Asequible: Ofrece buenos precios y una excelente relación calidad-precio.

Soporte y Herramientas: Tiene herramientas útiles para manejar la base de datos y otros aspectos técnicos sin complicaciones.

Roadmap

1. Adjunta la imagen de tu Roadmap:



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

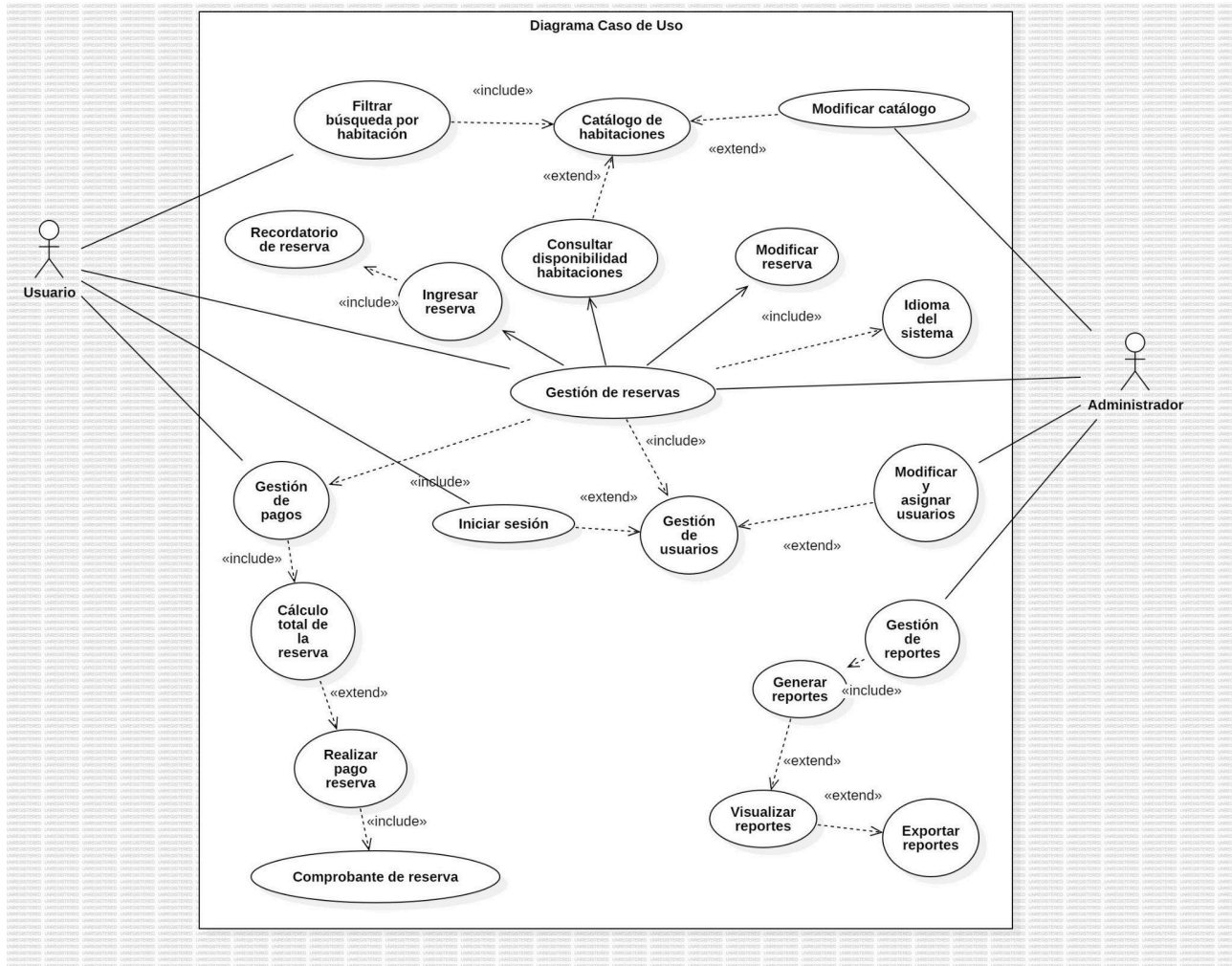
[Repositorio Glithub](#)

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

[Jira](#)

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

[Drive](#)



Parte II: Presentación

Pega el link que se generó al culminar la grabación de tu presentación en Teams

[Presentación Teams](#)



DuocUC[®] ONLINE

Duoc UC