



UNIVERSIDAD PRIVADA BOLIVIANA

GESTIÓN DE PROYECTOS INFORMÁTICOS

E-commerce de FENDERMED

Informe

Estudiante: Alexia Marin, Mateo Michel & Luciano Vargas

Códigos: 60855, 60707, 60872

La Paz – Bolivia – 04 de Noviembre 2024

- 1. Planificación**
 - a. Cronograma**
 - b. Herramientas**
 - c. Recursos**
- 2. Diseño y desarrollo**
 - a. Metodología de desarrollo**
 - b. Herramientas**
 - c. Mockups**
- 3. Control e implementación**
 - a. Pruebas**
 - b. Selección de servidor**
 - c. Como se lanzara a la web**

Planificación del Proyecto de E-Commerce para FENDERMED

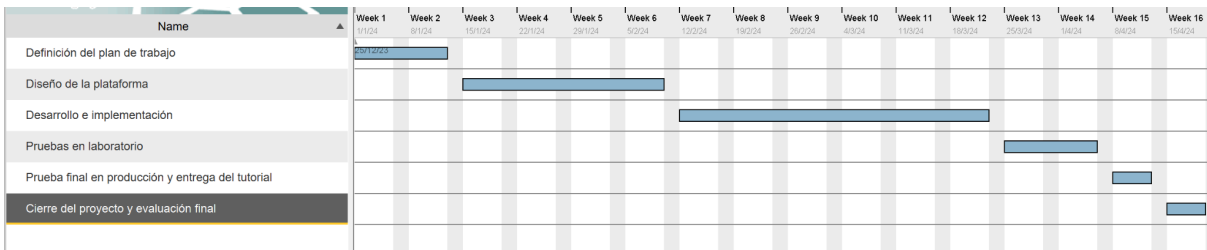
1. Planificación General

Este documento detalla la planificación del proyecto de desarrollo de una plataforma de E-Commerce para FENDERMED, orientada a facilitar la compra de equipos médicos y accesorios para establecimientos de salud. La plataforma estará diseñada para mejorar la experiencia de los clientes y optimizar los procesos de venta.

1.1 Cronogramas

El cronograma del proyecto se divide en las siguientes fases principales:

- Fase 1: Recolección de Requisitos (2 semanas)
 - Reunión con stakeholders.
 - Análisis de necesidades del cliente.
- Fase 2: Diseño y Planificación (3 semanas)
 - Diseño UX/UI de la plataforma.
 - Selección de herramientas de desarrollo y tecnologías.
- Fase 3: Desarrollo (8 semanas)
 - Implementación del frontend y backend.
 - Integración con pasarelas de pago y sistemas externos.
- Fase 4: Pruebas y Verificación (3 semanas)
 - Pruebas de usabilidad.
 - Corrección de errores detectados.
- Fase 5: Implementación y Lanzamiento (2 semanas)
 - Despliegue de la plataforma en el hosting seleccionado.
 - Capacitación al personal de FENDERMED.
- Fase 6: Mantenimiento Inicial (1 mes)
 - Soporte y resolución de problemas post-lanzamiento.



1.2 Herramientas

1.2.1 Para Planificación y Desarrollo

- **Hosting a Utilizar:** Se recomienda Amazon Web Services (AWS) para el hosting de la plataforma, dado que proporciona escalabilidad, disponibilidad del 99.9%, y seguridad con certificados SSL, características esenciales para un E-Commerce.
- **Herramientas de Desarrollo y Levantamiento del Sistema:**
 - **Visual Studio Code:** Como entorno de desarrollo integrado (IDE) para la programación del sistema.
 - **Node.js y Vue.js:** Se utilizarán para el desarrollo del frontend y backend de la plataforma, ya que ofrecen flexibilidad y buena integración.
 - **Postman:** Para probar y validar las APIs desarrolladas.
 - **GitHub:** Para la gestión del control de versiones y la colaboración en el desarrollo.

1.3 Recursos

- **Trello:** Para organizar las tareas del equipo, establecer objetivos claros y facilitar el seguimiento del avance del proyecto mediante tableros de planificación.
- **Vue.js:** Será la librería JavaScript principal para el desarrollo del frontend, proporcionando una interfaz interactiva y dinámica.
- **Amazon Web Services (AWS):** Para el hosting seguro y escalable de la plataforma.

1.4 Gestión de Riesgo

Para asegurar el éxito del proyecto, se debe realizar una gestión continua de riesgos. Cada riesgo se documentará utilizando el siguiente formato:

- **Problema:** Identificación del riesgo.
- **Probabilidad:** Bajo, medio o alto.
- **Impacto:** Alto, moderado o bajo.
- **Descripción:** Descripción detallada del riesgo y sus posibles efectos.

Ejemplos de Riesgos:

1. Riesgo de Atrasos en el Desarrollo

- **Probabilidad:** Medio
- **Impacto:** Alto
- **Descripción:** Retrasos en la implementación debido a problemas técnicos o cambios en los requisitos. Esto podría afectar el cronograma general y retrasar el lanzamiento.

2. Riesgo de Fallas en la Seguridad

- **Probabilidad:** Bajo
- **Impacto:** Alto
- **Descripción:** Vulnerabilidades en la seguridad del sistema que podrían exponer datos sensibles de los clientes. Se mitigará implementando protocolos de seguridad estrictos y realizando auditorías periódicas.

3. Riesgo de Baja Adopción por los Clientes

- **Probabilidad:** Medio
- **Impacto:** Moderado
- **Descripción:** Los clientes podrían no adoptar la plataforma de inmediato, lo que disminuiría el retorno de inversión esperado. Se mitigará mediante estrategias de marketing y soporte continuo al cliente.

4. Riesgo de Problemas en la Integración con Pasarelas de Pago

- **Probabilidad:** Medio
- **Impacto:** Alto
- **Descripción:** Dificultades técnicas al integrar las pasarelas de pago podrían afectar la capacidad de realizar transacciones en línea. Se mitigará realizando pruebas exhaustivas antes del lanzamiento y asegurando soporte continuo para la integración.

5. Riesgo de Falta de Capacitación del Personal

- **Probabilidad:** Medio
- **Impacto:** Moderado
- **Descripción:** El personal de FENDERMED podría tener dificultades para usar la plataforma debido a una capacitación insuficiente. Esto se mitigará proporcionando un tutorial detallado y sesiones de capacitación.

6. Riesgo de Cambios en los Requisitos del Cliente

- **Probabilidad:** Alto
- **Impacto:** Alto

- **Descripción:** Cambios solicitados por el cliente durante el desarrollo podrían requerir modificaciones importantes, afectando el cronograma y el presupuesto. Se mitigará mediante una gestión rigurosa de los cambios y la aprobación previa de los mismos.

2. Desarrollo

2.1 Metodologías de Desarrollo

Para el desarrollo de la plataforma de E-Commerce de FENDERMED, se utilizará la metodología Agile. Esta metodología permite una mayor flexibilidad y adaptación a cambios durante el desarrollo del proyecto. Algunas características clave de Agile para este proyecto incluyen:

- **Sprints:** El trabajo se dividirá en iteraciones cortas, conocidas como sprints, para asegurar entregas parciales y la revisión continua del avance.
- **Reuniones diarias (Daily Stand-ups):** Se realizarán reuniones cortas diarias para discutir el progreso, identificar bloqueos y garantizar que todos los miembros del equipo estén alineados.
- **Revisión y Retrospectiva:** Al final de cada sprint, se realizará una revisión del trabajo completado y una retrospectiva para identificar mejoras en el proceso.

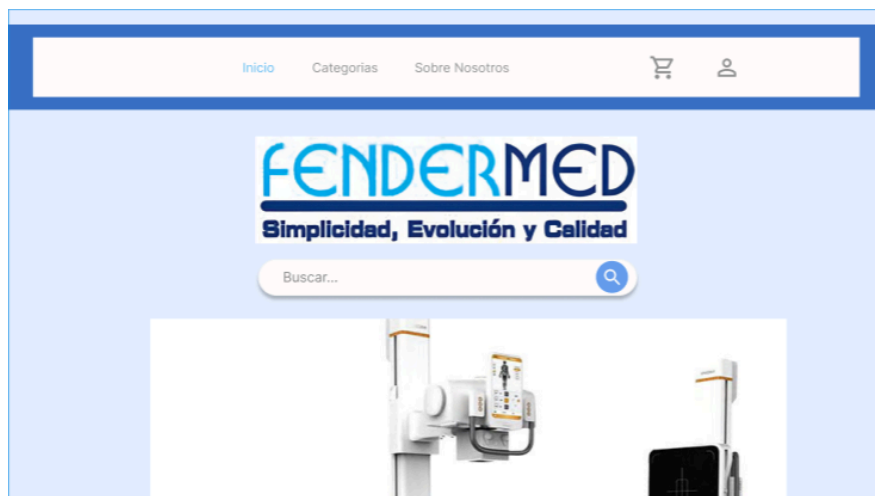
2.2 Herramientas

- **Jira:** Para la gestión de tareas y seguimiento del avance de los sprints. Permite gestionar de manera eficiente el backlog del proyecto y las tareas específicas para cada miembro del equipo.
- **Slack:** Como herramienta de comunicación interna del equipo de desarrollo, facilitando la colaboración y la comunicación rápida.
- **Docker:** Para la creación de entornos de desarrollo consistentes, lo cual facilita la integración continua y el despliegue en diferentes entornos (prueba, producción).
- **Jenkins:** Para la integración continua (CI), permitiendo la compilación y prueba automática del código a medida que se desarrollan nuevas funcionalidades.

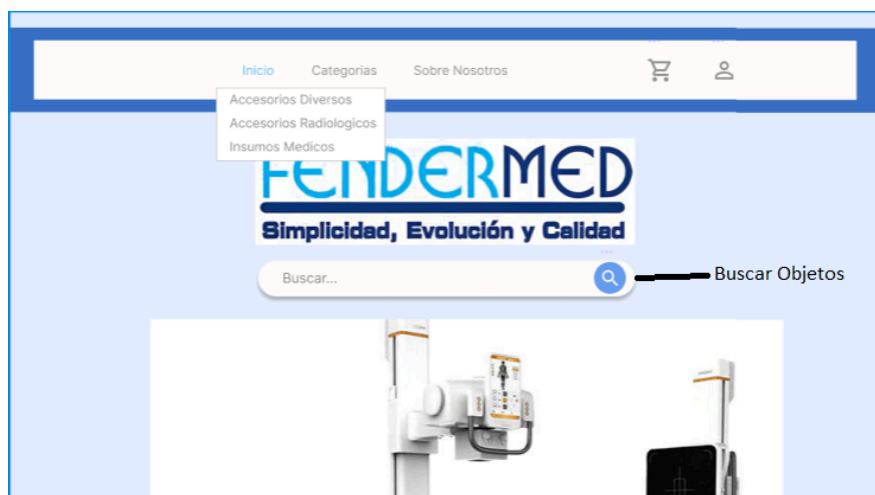
3. Diseño

3.1 Mockups

3.1.1 Inicio



3.1.2 Categorías



3.1.3 Detalles del proyecto

Login form

☐ remember me
[reset password](#)

Correo Electronico o usuario

Contraseña

Recuerda al usuario

Cambiar la contraseña a otra

dont have an account? [register page](#)

Registrarse a la pagina web

De vuelta al inicio de sesion con la cuenta iniciada

ADMINNNNNNNNNNNNN

4. Control

4.1 Cómo Hacer las Pruebas de Control

Las pruebas de control se realizarán siguiendo un enfoque exhaustivo para asegurar que la plataforma funcione correctamente bajo diferentes condiciones. Los tipos de pruebas a realizar incluyen:

- **Pruebas Unitarias:** Validar la funcionalidad de componentes individuales del sistema.
- **Pruebas de Integración:** Verificar la correcta interacción entre distintos módulos de la plataforma.
- **Pruebas de Usabilidad:** Asegurarse de que la plataforma sea fácil de usar para los usuarios finales.
- **Pruebas de Carga:** Evaluar el rendimiento de la plataforma bajo alta demanda.
- **SERVIDOOOOOOOR**

Estas pruebas se realizarán utilizando herramientas como Selenium para la automatización de pruebas y JMeter para las pruebas de carga.

4.2 Servidor a Usar

Se utilizará Amazon Web Services (AWS) como servidor principal. AWS ofrece la escalabilidad y seguridad necesarias para soportar una plataforma de E-Commerce. Los servicios específicos a utilizar incluyen Amazon EC2 para servidores virtuales, Amazon S3 para el almacenamiento de archivos, y Amazon RDS para la base de datos.

4.3 Cómo Lanzarlo a la Web

El lanzamiento a la web se realizará mediante un enfoque de Despliegue Gradual para minimizar riesgos. El proceso será el siguiente:

- **Fase de Prueba Interna:** Desplegar la plataforma en un entorno de prueba para su evaluación por el equipo interno de FENDERMED.
- **Fase de Beta Cerrada:** Permitir que un grupo limitado de usuarios pruebe la plataforma y proporcione retroalimentación antes del lanzamiento oficial.
- **Despliegue en Producción:** Una vez validados todos los aspectos, la plataforma será desplegada en el entorno de producción utilizando AWS Elastic Beanstalk para gestionar la infraestructura y simplificar el proceso de despliegue.
- **Monitoreo Post-Lanzamiento:** Utilizar AWS CloudWatch para monitorear el rendimiento de la plataforma y asegurar la estabilidad en el entorno de producción.