**C.A. Fashion**

Nombre: Cesar Camilo Bernal Quiñonez

Programa: Ingeniería de Software

Proyecto PIA 2024

El presente documento describe la metodología empleada para desarrollar una página web de venta de ropa, detallando las técnicas y enfoques aplicados para lograr un producto de alta calidad. La metodología seleccionada para este proyecto es Scrum, un marco ágil adecuado para proyectos de desarrollo de software, que se adapta a las necesidades y a los constantes cambios que puedan surgir durante el desarrollo del sistema. Este documento se enriquece con un vocabulario técnico de un estudiante universitario especializado en ingeniería de software.

2. Metodología Ágil Scrum

Scrum es un marco ágil que organiza el trabajo en sprints cortos y entregables incrementales. Cada sprint tiene una duración de entre 1 y 4 semanas, permitiendo iterar y mejorar el producto de manera constante. Para este proyecto, se optó por aplicar sprints de 2 semanas para cada entrega.

2.1 Roles en Scrum

En un equipo de Scrum, se identifican los siguientes roles:

Scrum Master: Responsable de eliminar obstáculos, asegurar que se siga el proceso de Scrum y facilitar la comunicación entre el equipo.

Product Owner: Responsable de definir los requerimientos del sistema, priorizar el backlog del producto y asegurar que el proyecto entregue valor al cliente.

Equipo de Desarrollo: Compuesto por programadores, diseñadores de la interfaz y otros roles técnicos, que son responsables de la implementación y entrega del producto.

2.2 Artefactos de Scrum

Los artefactos más importantes utilizados durante el desarrollo de la página web fueron:

Product Backlog: Una lista priorizada de las funcionalidades y requerimientos que el Product Owner quiere incluir en el proyecto. En nuestro caso, el backlog incluía elementos como: registro de usuarios, integración con carrito de compras, módulo de pago, etc.

Sprint Backlog: Un subconjunto del product backlog que se selecciona para desarrollarse durante cada sprint.

Incremento: El resultado del sprint, que incluye todas las funcionalidades desarrolladas, probadas e integradas al sistema.

2.3 Sprints Utilizados en el Proyecto

El proyecto se dividió en varios sprints, donde se trabajó en la implementación de diferentes módulos de la página web:

Sprint 1: Diseño e implementación de la interfaz de usuario (UI) y navegación básica.

Sprint 2: Implementación del registro e inicio de sesión de usuario.

Sprint 3: Desarrollo del catálogo de productos y carrito de compras.

Sprint 4: Implementación del sistema de pago y comunicación con la pasarela de pago.

Sprint 5: Pruebas, correcciones de bugs, y optimización del rendimiento.

Cada sprint comenzó con una reunión de planificación del sprint (Sprint Planning), donde se determinó el conjunto de tareas a realizar, y finalizó con una revisión del sprint (Sprint Review) y una retrospectiva del sprint (Sprint Retrospective) para evaluar lo que se puede mejorar.

3. Desarrollo y Tecnologías Utilizadas

Para la construcción de la página web se emplearon tecnologías web actuales, lo que permitió crear un sistema robusto, escalable y fácil de mantener. Las tecnologías y herramientas clave utilizadas incluyen:

HTML5 y CSS3: Para el diseño de la interfaz de usuario (UI) responsiva y atractiva.

JavaScript (ES6): Para el comportamiento interactivo del sitio, como la actualización del carrito de compras y la validación de formularios.

Frameworks y Librerías:

Bootstrap: Para un diseño responsivo y consistente.

Node.js: Para el backend, donde se gestionan las solicitudes de los usuarios y la lógica de negocio.

Express.js: Framework minimalista de Node.js utilizado para construir la API REST del sitio.

Base de Datos:

MySQL: Utilizada para almacenar la información de usuarios, productos, órdenes, y carritos.

4. Estructura del Sistema

La página web sigue una arquitectura basada en MVC (Modelo-Vista-Controlador), donde:

Modelo: Se encarga de la lógica de acceso y manipulación de los datos. Está conectado a la base de datos para gestionar la información de los usuarios, productos, carritos y órdenes.

Vista: Son las interfaces de usuario que interactúan con el cliente. Se diseñaron utilizando HTML, CSS y JavaScript.

Controlador: Es responsable de recibir las peticiones HTTP, procesarlas y devolver una respuesta adecuada a la vista.

5. Modelo Entidad-Relación (ERD)

El sistema está basado en un diseño de base de datos relacional que sigue el siguiente modelo entidad-relación:

Usuario: Contiene la información básica de los usuarios.

Producto: Incluye los detalles de los productos disponibles para la venta.

Orden: Registra las compras realizadas por los usuarios.

Carrito: Mantiene la relación entre los productos y las órdenes hasta que se completa el proceso de pago.

6. Control de Versiones y Gestión del Proyecto

Durante el desarrollo, se empleó Git como herramienta de control de versiones para colaborar en equipo de forma eficiente. La plataforma GitHub fue utilizada para mantener un repositorio centralizado del código y administrar los issues del proyecto.

7. Pruebas y Validación

Las pruebas se realizaron utilizando los enfoques de Test Driven Development (TDD), donde se escribieron pruebas automatizadas antes de desarrollar las funcionalidades clave. Estas pruebas garantizaron que las funciones trabajaran de manera correcta y se detectaran errores en las primeras etapas de desarrollo.

Tipos de pruebas aplicadas:

Pruebas Unitarias: Para validar el comportamiento de funciones individuales.

Pruebas de Integración: Para verificar que los distintos módulos del sistema funcionen correctamente en conjunto.

Pruebas de Usuario: Simulaciones del flujo del cliente desde el registro hasta la compra.

8. Conclusión

El desarrollo de la página web de venta de ropa se llevó a cabo utilizando Scrum, una metodología ágil que permitió la adaptación a los cambios y la entrega de valor constante. Con sprints iterativos, un enfoque en el usuario final, y el uso de tecnologías modernas, se logró construir una plataforma de ventas funcional, escalable y optimizada.

Glosario de Términos

Backlog: Lista priorizada de tareas y características por implementar.

Sprint: Ciclo corto de desarrollo en Scrum para completar un conjunto de tareas.

Framework: Conjunto de herramientas y bibliotecas que proporcionan una estructura estándar para el desarrollo de software.

Git: Sistema de control de versiones distribuido.

API REST: Interfaz de programación de aplicaciones que permite la comunicación entre sistemas de forma sencilla y escalable.