



Universidad Nacional De Asunción

Facultad Politécnica

Ingeniería en Informática

Ingeniería de Software II

Profesor:

- Ms. Guillermo Jacobo González Rodas

Materia: Sistemas Distribuidos.

Alumnos:

- Fabrizio Amadeo Rejala Galeano.
- Alexander Rojas
- Marcos Raúl Flores Duarte
- Ever Rolando Caballero Mercado
- Enzo Sebastian Vidallet Monges

Sección: TQ.

2024

- Introducción

El presente informe final describe las características principales del proyecto de desarrollo de software realizado en el marco de la asignatura. Este documento aborda el proceso de desarrollo utilizado, basado en metodologías ágiles, y sintetiza las experiencias, aprendizajes y resultados obtenidos durante la ejecución del proyecto.

El proyecto se centró en la construcción de un Sistema de Administración de Contenidos Web (CMS) con el objetivo de aplicar y profundizar los conocimientos teóricos y prácticos adquiridos, a través del uso de la metodología SCRUM, se buscó fortalecer las habilidades de trabajo en equipo, planificación, y ejecución de tareas, con un enfoque en el desarrollo iterativo y la entrega de incrementos funcionales.

En el desarrollo del proyecto se utilizaron herramientas modernas de soporte al ciclo de vida del software, tales como plataformas para la gestión de requisitos, planificación de tareas y control de versiones. Asimismo, se promovió la utilización de buenas prácticas de programación y aseguramiento de la calidad, con un énfasis especial en el cumplimiento de los hitos establecidos según la guía metodológica provista por la cátedra.

A lo largo del informe, se presentarán los detalles de la organización del equipo de trabajo, las metodologías empleadas, las oportunidades de mejora identificadas, así como las lecciones aprendidas y conclusiones derivadas de esta experiencia académica y profesional.

- El equipo de trabajo, descripción de perfiles:

1. Alex Rojas - Desarrollador Frontend y Soporte Backend

- **Rol en el equipo:** Responsable del desarrollo de la interfaz de usuario y colaborador en tareas del backend.
- **Tareas principales:**
 - Desarrollo de la interfaz en **React** para integrar los servicios backend.
 - Implementación de las vistas y controles de acceso según los roles de los usuarios.
 - Diseño e implementación del tablero **Kanban** para la gestión del flujo de trabajo.
 - Apoyo en la implementación de tareas del backend relacionadas con la integración y validación de permisos.

2. Fabrizio Rejala - Desarrollador Full Stack

- **Rol en el equipo:** Líder técnico en configuración y desarrollo de infraestructura.
- **Tareas principales:**
 - Configuración inicial del proyecto, incluyendo contenedores **Docker** para entornos de desarrollo y producción.
 - Implementación de la pasarela de pagos mediante **Stripe**.

- Desarrollo de servicios backend para la gestión y asignación de permisos.
- Apoyo en tareas del frontend y backend según las necesidades del proyecto.

3. Enzo Vidallet - Desarrollador Backend

- **Rol en el equipo:** Encargado del desarrollo y optimización de servicios backend.
- **Tareas principales:**
 - Implementación de la carga de imágenes mediante el contenedor **Pictshare**.
 - Desarrollo de operaciones CRUD y servicios GET para la gestión de contenido.
 - Implementación de la vigencia y control de contenido publicado.
 - Creación de un script para automatizar el despliegue del proyecto usando los tags de **GitHub**.
 - Diseño y ejecución de pruebas unitarias para asegurar la calidad del software.

4. Ever Caballero - Scrum Master y Desarrollador Backend

- **Rol en el equipo:** Líder en la planificación y coordinación del proyecto, además de desarrollador backend.
- **Tareas principales:**
 - Asignación y seguimiento de tareas mediante **Jira**, asegurando el cumplimiento de las metodologías ágiles.
 - Desarrollo de servicios GET y CRUD para la gestión de etiquetas y categorías.
 - Implementación del módulo de reportes estadísticos para el análisis del contenido.
 - Desarrollo del sistema de historial de cambios de contenido y reportes asociados.

5. Marcos Flores - Desarrollador Backend y Gestión de Documentación

- **Rol en el equipo:** Encargado de servicios Backend y elaboración de documentación del sitio.
- **Tareas principales:**
 - Desarrollo de operaciones CRUD y servicios GET para la gestión de contenido.
 - Implementación del sistema de comentarios mediante **Disqus**.
 - Creación del módulo de notificaciones por correo para informar sobre cambios en el flujo de publicación.
 - Documentación técnica del proyecto usando **Sphinx**.

- Descripción del proceso utilizado en el proyecto

El desarrollo del proyecto se llevó a cabo utilizando la **metodología ágil SCRUM**, adaptada al entorno académico. Este enfoque permitió una gestión efectiva del tiempo, los recursos y las tareas asignadas al equipo, fomentando la colaboración y la entrega continua de incrementos funcionales del sistema.

- Oportunidades de mejora en el desempeño del equipo

Uno de los principales desafíos enfrentados fue la tendencia del equipo a reunirse a última hora, es decir, pocos días antes de las fechas de entrega de los hitos. Esta práctica limitó la capacidad de identificar y resolver problemas con suficiente antelación, además de generar presión innecesaria sobre los miembros del equipo. teniendo la oportunidad de mejorar en establecer reuniones regulares durante cada sprint (al inicio, a la mitad y antes de la entrega), permitiendo revisar avances y ajustar tareas según sea necesario. La falta de reuniones frecuentes dificulta la sincronización entre los integrantes del equipo, resultando en posibles redundancias o falta de integración entre las tareas desarrolladas.

La ejecución tardía de las tareas dejó al equipo con poco margen para manejar imprevistos o realizar ajustes necesarios.

- Lecciones aprendidas del proceso

- Adaptación y Colaboración en Equipo: Aprendimos a adaptarnos a los estilos de trabajo y habilidades de cada miembro, lo que permitió una mejor colaboración y distribución de tareas. Esto fue esencial para superar desafíos y cumplir con los objetivos en tiempo y forma.
- Implementación de Pruebas Unitarias: Incorporar pruebas unitarias en el desarrollo ayudó a detectar errores de forma temprana, mejorando la calidad del software entregado y reduciendo el tiempo dedicado a la corrección de errores en etapas avanzadas del proyecto.
- Uso Básico de Contenedores Docker: La experiencia con Docker permitió comprender cómo configurar y gestionar contenedores, facilitando la portabilidad y consistencia del entorno de desarrollo y despliegue.
- Utilización de Herramientas de Gestión como Jira: El uso de Jira para asignar tareas, gestionar el tablero Kanban y dar seguimiento al progreso del equipo fue clave para organizar el trabajo y garantizar que todos estuvieran alineados con los objetivos del proyecto.

- Experiencias importantes a mencionar

El uso de herramientas como Git para el control de versiones y Jira para la gestión de tareas representó una experiencia enriquecedora. Esto permitió comprender la importancia de documentar cada cambio en el código. Experimentar los beneficios de una planificación centralizada y accesible para todos los integrantes.

La configuración de scripts de despliegue para ambientes de desarrollo y producción fue una experiencia técnica significativa. Permitió entender la importancia de la automatización en procesos repetitivos y garantizar la estabilidad del sistema en diferentes entornos.

Contraste de Requerimientos Realizados y No Realizados de Ingeniería de Software 1

Requerimientos Funcionales

- Inicio de Sesión:

Realizados:

- Registro de usuarios con datos básicos.
- Inicio de sesión de usuarios registrados.

No Realizados:

- Opción para recordar las credenciales del usuario.
- Recuperación de contraseña mediante correo electrónico.

- Navegación por el Sistema:

Realizados:

- Visualización de publicaciones y posibilidad de realizar comentarios.
- Organización del contenido por categorías.
- Acceso para usuarios registrados y no registrados.
- Buscador para filtrar por título y categoría.
- Creación, edición y publicación de contenido según permisos.

No Realizados:

- Visualización de perfiles de usuarios, incluyendo sus publicaciones recientes, aprobadas o rechazadas.

- Administración de Usuarios:

Realizados:

- Modificación de permisos de roles por parte de administradores.
- Asignación y modificación de roles de usuarios.

No Realizados:

- Baja de usuarios por parte de administradores.

- Manejo de Contenido:

Realizados:

- Creación y modificación de contenido por usuarios con permisos.
- Creación y modificación de categorías por administradores.
- Sección de comentarios para los usuarios.
- Edición, autorización y publicación de contenido según permisos.
- Establecimiento de un período de vigencia para contenido.
- Registro de visualizaciones por publicación y categoría.
- Módulo de notificaciones sobre cambios y comentarios en publicaciones.

No Realizados:

- Plantillas para personalizar el aspecto del contenido según su tipo.
- Eliminación de comentarios por usuarios con permisos (esto se realiza externamente).

Requerimientos No Funcionales

Realizados:

- Uso con mouse y teclado.
- Escalabilidad para soportar incremento de tráfico.
- Navegación intuitiva.
- Medidas de seguridad para proteger la privacidad de datos.
- Compatibilidad con múltiples navegadores.
- Adaptabilidad de la pantalla a distintas resoluciones.

No Realizados:

Tiempo de respuesta ≤ 1 segundo para solicitudes.

Robustez contra ataques de denegación de servicio.

Disponibilidad 24/7 (falta aclarar tiempos de inactividad planificados).

Sistemas de respaldo y recuperación ante pérdida de datos.

Funcionalidades de optimización para motores de búsqueda (SEO).

Resumen General

Requerimientos Funcionales:

- Realizados: 17
- No Realizados: 7

Requerimientos No Funcionales:

- Realizados: 7
- No Realizados: 4

El sistema cumple con una mayoría de los requerimientos planteados inicialmente, destacando en áreas como navegación, manejo de contenido, y compatibilidad técnica. Sin embargo, hay áreas por abordar, como la recuperación de contraseñas, medidas de seguridad avanzadas (robustez frente a ataques y sistemas de respaldo), y mejoras en la optimización para motores de búsqueda. Que no fueron implementadas por cuestiones de tiempo para priorizar otros apartados.

Conclusiones

El desarrollo del proyecto representó una experiencia académica y profesional integral, en la cual se lograron los objetivos planteados, tanto en la creación de un sistema funcional como en la adquisición de conocimientos prácticos sobre ingeniería de software.

El equipo logró desarrollar un Sistema de Administración de Contenidos Web (CMS) funcional, cumpliendo con los requerimientos establecidos y aplicando metodologías y herramientas modernas. Esto permitió consolidar los conceptos teóricos estudiados en la asignatura y adquirir experiencia práctica en un entorno simulado de desarrollo profesional.

El proyecto no solo permitió cumplir con los objetivos académicos establecidos, sino que también preparó al equipo para enfrentar los retos del desarrollo de software en un contexto profesional, sentando una base sólida para proyectos futuros. La experiencia resaltó la importancia de combinar conocimientos técnicos con habilidades de gestión y colaboración para alcanzar el éxito en proyectos complejos.