



**UNIVERSIDAD PRIVADA DE TACNA**

**FACULTAD DE INGENIERIA**

**Escuela Profesional de Ingeniería de Sistemas**

**Informe Final**

**Sistema de Conversión y Organización de  
Documentos Técnicos en Markdown con  
Estructuración Automática y Control de Versiones  
para los estudiantes en la facultad de Ingeniería de  
Sistemas**

Curso: Patrones de Software

Docente: *Patrick Cuadros Quiroga*

Integrantes:

<b><i>Chambi Cori, Jerson Roni</i></b>	<b><i>(2021072619)</i></b>
<b><i>Flores Quispe, Jaime Elias</i></b>	<b><i>(2021070309)</i></b>
<b><i>Leyva Sardon, Elvis Ronald</i></b>	<b><i>(2021072614)</i></b>
<b><i>Chite Quispe, Brian Danilo</i></b>	<b><i>(2021070015)</i></b>

**Tacna – Perú  
2025**

# **Sistema de Conversión y Organización de Documentos Técnicos en Markdown**

**Versión 1.0**

CONTROL DE VERSIONES					
Versión	Hecha por	Revisada por	Aprobada por	Fecha	Motivo
1.0	MPV	ELV	ARV	10/10/2020	Versión Original

## INDICE GENERAL

<b>1. Antecedentes.....</b>	<b>4</b>
<b>2. Planteamiento del Problema.....</b>	<b>5</b>
<b>a. Problema .....</b>	<b>5</b>
b. Justificación.....	6
c. Alcance.....	7
<b>3. Objetivos.....</b>	<b>8</b>
<b>4. Marco Teórico.....</b>	<b>8</b>
<b>5. Desarrollo de la Solución .....</b>	<b>10</b>
<b>a. Análisis de Factibilidad (técnico, económica, operativa, social, legal, ambiental) .....</b>	<b>11</b>
Factibilidad Operativa .....	11
Factibilidad Legal.....	12
<b>b. Tecnología de Desarrollo.....</b>	<b>14</b>
<b>c. Metodología de implementación.....</b>	<b>16</b>
<b>6. Cronograma .....</b>	<b>18</b>
<b>7. Presupuesto .....</b>	<b>20</b>
Costos de personal.....	21
<b>8. Conclusiones.....</b>	<b>23</b>
Recomendaciones .....	24
Bibliografía.....	25
Anexos.....	26

## 1. Antecedentes

El desarrollo del Sistema de Conversión y Organización de Documentos Técnicos en Markdown surge como respuesta a la creciente necesidad de estandarizar y optimizar la gestión de la documentación académica en entornos digitales, particularmente en carreras técnicas como Ingeniería de Sistemas. En los últimos años, diversas iniciativas académicas y tecnológicas han puesto de manifiesto la importancia de contar con herramientas que faciliten la creación, estructuración y control de versiones de documentos técnicos, permitiendo a estudiantes y docentes una mejor organización de su producción académica.

Experiencias previas en instituciones educativas y comunidades tecnológicas han adoptado plataformas como GitHub y el uso del formato Markdown para la documentación colaborativa, debido a su flexibilidad, compatibilidad y ligereza. Sin embargo, la transición desde formatos tradicionales (Word, PDF, HTML, TXT) hacia el entorno Markdown ha representado un desafío recurrente, especialmente por la falta de herramientas que automaticen dicho proceso sin comprometer la estructura y el contenido original de los documentos.

La evolución de tecnologías como la inteligencia artificial y los entornos de control de versiones ha abierto nuevas posibilidades para gestionar y mejorar la documentación. Algunos proyectos previos han demostrado que la integración de estas tecnologías puede facilitar la navegación, trazabilidad y colaboración en entornos académicos, estableciendo así las bases para soluciones más robustas y adaptadas a las necesidades actuales del entorno educativo digital.

Por otro lado, las lecciones aprendidas de iniciativas similares en el ámbito educativo han resaltado la importancia de aspectos como la escalabilidad, la interoperabilidad con plataformas existentes y la adopción de estándares abiertos. Estos antecedentes han influido directamente en el diseño del sistema propuesto, que no solo busca resolver las limitaciones técnicas existentes, sino también fomentar buenas prácticas de documentación y colaboración entre los estudiantes, preparándolos para entornos profesionales donde estas habilidades son cada vez más valoradas.

## 2. Planteamiento del Problema

En el ámbito académico, particularmente en carreras como Ingeniería de Sistemas, la gestión de documentos técnicos se ve afectada por múltiples limitaciones relacionadas con la estandarización, el control de versiones y la accesibilidad. La falta de plataformas especializadas que integren funcionalidades clave como la conversión automatizada de archivos, la estructuración coherente del contenido y la trazabilidad del historial de documentos constituye una barrera significativa para estudiantes y docentes que necesitan manejar gran cantidad de información técnica de manera ágil y colaborativa.

Uno de los principales desafíos identificados es la ausencia de una plataforma integral que permita convertir, organizar, mejorar y versionar documentos de forma eficiente y segura. Actualmente, muchos usuarios deben recurrir a múltiples herramientas y procesos manuales para adaptar sus documentos a entornos como GitHub, lo cual genera inconsistencias de formato, pérdida de estructura y dificultad para mantener un control adecuado sobre las versiones previas.

Asimismo, la falta de herramientas de navegación como sidebar.md y footer.md limita la capacidad de los usuarios para estructurar sus documentos en secciones claras, lo cual complica la experiencia de lectura y comprensión de materiales complejos. Esta carencia afecta tanto a la calidad de los entregables académicos como a la posibilidad de reutilizar contenido en otros contextos.

La gestión de cuentas y documentos también representa un aspecto crítico. La ausencia de funcionalidades que permitan a los usuarios visualizar sus historiales de conversión, acceder a versiones anteriores o administrar sus archivos con precisión, limita la trazabilidad y la mejora continua de los trabajos realizados.

Finalmente, desde la perspectiva de la seguridad y privacidad, es esencial garantizar el control sobre la información cargada en el sistema. La inexistencia de mecanismos robustos para proteger los datos de los usuarios, como cierre de sesión seguro y autenticación adecuada, puede generar desconfianza y comprometer la aceptación de la plataforma como una solución confiable en el ámbito académico.

### a. Problema

El proyecto surge en respuesta a una problemática común en el entorno académico: la falta de herramientas unificadas que permitan la conversión automatizada y organizada de documentos técnicos al formato Markdown, con control de versiones y estructuración inteligente. Esta carencia afecta tanto el proceso de aprendizaje como la colaboración entre estudiantes y la calidad final de los proyectos académicos.

- **Conversión manual ineficiente:** La transformación de archivos en formatos como Word, PDF, HTML o TXT a Markdown se realiza de forma manual, lo cual consume un tiempo considerable, introduce errores humanos y dificulta la estandarización de los documentos, afectando la calidad del trabajo final.
- **Ausencia de navegación estructurada:** Los documentos convertidos a Markdown, en su mayoría, carecen de elementos de navegación automática (como sidebar.md, footer.md), lo cual dificulta su comprensión, especialmente cuando se alojan en plataformas colaborativas como GitHub o sistemas de gestión de proyectos académicos.

- **Falta de trazabilidad y control de versiones:** Actualmente no existe un sistema que permita a los usuarios guardar y recuperar versiones anteriores de sus documentos. Esto obstaculiza el trabajo colaborativo, ya que no se puede rastrear el progreso o los cambios realizados a lo largo del tiempo.
- **Limitada gestión de cuentas y archivos:** Los estudiantes no cuentan con herramientas adecuadas para monitorear y administrar sus documentos, editar el historial de versiones o gestionar las conversiones realizadas. Esto hace que la organización de los trabajos académicos sea aún más difícil y desordenada.
- **Riesgos de seguridad y privacidad:** La falta de un mecanismo claro para cerrar sesión de manera segura y proteger los datos almacenados plantea riesgos de seguridad y privacidad, especialmente cuando se maneja información sensible relacionada con proyectos académicos.

## b. Justificación

La justificación del presente proyecto se fundamenta en la creciente necesidad del entorno académico de contar con herramientas digitales que faciliten la creación, conversión y organización de documentos técnicos de manera eficiente, estandarizada y segura. En documentaciones en Ingeniería de Sistemas, en donde la documentación es esencial en el proceso de aprendizaje y desarrollo de proyectos, se observa una creciente demanda por soluciones tecnológicas que optimicen la gestión de archivos en diversos formatos.

Actualmente, los estudiantes y docentes enfrentan dificultades al trabajar con múltiples tipos de documentos (Word, PDF, HTML, TXT), especialmente cuando estos necesitan ser migrados a plataformas de control de versiones como GitHub, donde el formato Markdown (.md) se ha consolidado como un estándar.

El proyecto también responde a la necesidad de superar las barreras técnicas y de conocimiento que muchos usuarios enfrentan al intentar integrar sus documentos en entornos colaborativos más avanzados y profesionales. Al ofrecer una plataforma accesible y automatizada, se elimina la dependencia de procesos manuales, que suelen ser complejos y propensos a errores. De esta forma, los estudiantes pueden generar documentación estructurada, clara y compatible con los estándares modernos, sin importar el dispositivo que utilicen, siempre que tengan acceso a internet.

Una de las características clave que distingue a este proyecto es la incorporación de funcionalidades avanzadas. Estas incluyen la generación automática de elementos de navegación (como `sidebar.md`, `footer.md`), la mejora del contenido mediante inteligencia artificial y el control de versiones. Estas características no solo enriquecen la experiencia del usuario, sino que promueven una cultura de documentación profesional, brindando a los estudiantes herramientas

que les ayudarán a desarrollar competencias esenciales tanto en su vida académica como en su futura carrera profesional.

Además, la seguridad y la protección de los datos académicos son aspectos cruciales dentro de este proyecto. El sistema integrará mecanismos de autenticación, gestión segura de sesiones y protección contra accesos no autorizados, asegurando la integridad de la información procesada. Esta implementación refuerza la confianza en la herramienta y garantiza que se adhiera a las mejores prácticas de desarrollo y protección de datos en el entorno educativo.

### **c. Alcance**

El alcance del proyecto de desarrollo del Sistema de Conversión y Organización de Documentos Técnicos en Markdown se fundamenta en la necesidad creciente dentro del entorno académico de contar con herramientas digitales más accesibles, eficientes y estandarizadas para la gestión de documentación técnica. En un contexto educativo cada vez más digitalizado, estudiantes y docentes requieren soluciones que les permitan convertir, organizar y controlar versiones de sus documentos desde cualquier lugar y en cualquier momento, utilizando únicamente sus dispositivos personales.

Este sistema busca eliminar las limitaciones asociadas al uso de múltiples formatos (Word, PDF, HTML, TXT) y la dependencia de herramientas manuales, centralizando el proceso en una plataforma web unificada que automatiza la conversión de documentos a Markdown. Al estandarizar este formato, ampliamente adoptado en entornos profesionales como GitHub, se facilita la colaboración académica, la reutilización de contenidos y la integración con plataformas de control de versiones.

Asimismo, el proyecto abarca la incorporación de funcionalidades clave que mejoran significativamente la experiencia del usuario, tales como la generación automática de archivos de navegación (sidebar.md, footer.md), la mejora de contenido mediante inteligencia artificial, y la visualización estructurada del historial de versiones. Estas capacidades permiten optimizar no solo la presentación de los documentos, sino también su mantenimiento y seguimiento en el tiempo.

Otro componente esencial del alcance es la seguridad. El sistema está diseñado para garantizar la protección de los datos académicos, implementando mecanismos de autenticación, cierre de sesión seguro y control de accesos, de modo que se salvaguarde la confidencialidad de los documentos gestionados por los usuarios.

En conjunto, el proyecto no solo responde a una necesidad técnica específica, sino que también promueve el desarrollo de competencias digitales relevantes, como el uso de herramientas de documentación moderna y el trabajo colaborativo en entornos distribuidos. Su implementación impactará directamente en la eficiencia académica, la calidad del trabajo documental y la preparación de los estudiantes para los entornos profesionales actuales.

### 3. Objetivos

#### Objetivo Principal:

Diseñar e implementar un sistema web que facilite la conversión automática de documentos a formato Markdown, mejorando la organización, estructuración y accesibilidad de la documentación técnica en entornos académicos.

#### Objetivos Específicos:

- Automatizar la conversión de documentos: Desarrollar un módulo que permita transformar archivos en distintos formatos a Markdown sin perder su estructura.
- Implementar un sistema de gestión de versiones: Permitir a los usuarios almacenar y recuperar versiones previas de sus documentos.
- Integrar una funcionalidad para mejorar documentos con deepseek: Permitir la mejora del contenido del documento implementando inteligencia artificial.
- Garantizar la seguridad del sistema: Implementar autenticación y control de accesos para proteger la información procesada.

### 4. Marco Teórico

El desarrollo del presente sistema se sustenta en diversos conceptos teóricos que permiten comprender la necesidad, el enfoque y la viabilidad de implementar una plataforma para la conversión y organización de documentos técnicos en formato Markdown, con estructuración automática y control de versiones. Estos fundamentos abarcan desde aspectos tecnológicos hasta educativos y de gestión de la información.

#### 4.1 Documentación Técnica y Estándares Académicos

La documentación técnica es esencial en el desarrollo de proyectos académicos y de ingeniería, ya que proporciona una base estructurada que facilita la comprensión y replicabilidad de los trabajos realizados. Según el IEEE (Institute of Electrical and Electronics Engineers), una buena práctica de documentación debe garantizar aspectos como la claridad, trazabilidad, estandarización y accesibilidad, asegurando que los documentos sean comprensibles, fáciles de seguir y que la evolución del proyecto pueda ser rastreada. Sin embargo, en el ámbito educativo, la multiplicidad de formatos utilizados (Word, PDF, HTML, TXT) y la falta de lineamientos claros dificultan la creación de documentos consistentes y colaborativos. Esta fragmentación de formatos genera inconvenientes a la hora de integrar y compartir trabajos entre diferentes plataformas. En este contexto, Markdown se ha consolidado como un estándar ligero y ampliamente adoptado para estructurar contenido técnico de manera clara y accesible, siendo compatible con plataformas de colaboración como GitHub o GitLab. Su simplicidad, junto con su capacidad para mantener una estructura ordenada y legible, lo convierte en una opción ideal para la creación de documentos técnicos en entornos académicos y profesionales, promoviendo la interoperabilidad y facilitando la colaboración.



## 4.2 Conversión Automatizada y Procesamiento de Documentos

La automatización de tareas repetitivas y propensas a error, como la conversión de documentos entre diferentes formatos, es un principio central en la ingeniería de software. Herramientas modernas basadas en bibliotecas de Python, como `markdownify`, `pdfminer` y `python-docx`, permiten extraer, interpretar y transformar el contenido de los documentos manteniendo su estructura semántica. Estas bibliotecas facilitan la conversión de formatos como DOCX, PDF, HTML y TXT a Markdown, garantizando que la información se conserve de manera precisa y ordenada. Este enfoque responde al paradigma de eficiencia, al reducir la intervención humana y minimizar los errores asociados a tareas manuales. La automatización inteligente, al agilizar procesos y mejorar la precisión, se alinea con los principios fundamentales de la ingeniería de software, buscando optimizar el trabajo y mejorar la calidad del resultado final.

## 4.3 Control de Versiones y Gestión Colaborativa

Git y GitHub han revolucionado la manera en que se gestiona la evolución de archivos en entornos colaborativos. La integración de un sistema de control de versiones en plataformas académicas permite a los usuarios seguir los cambios realizados en los documentos, recuperar versiones anteriores y facilitar la colaboración en proyectos grupales. Este enfoque se fundamenta en la teoría de sistemas distribuidos y en el control de versiones semántico, garantizando la integridad de la información y proporcionando un historial detallado del proceso documental. De esta forma, los usuarios pueden trabajar de manera conjunta, gestionando eficazmente los cambios sin perder la trazabilidad de los mismos, lo cual es crucial en el contexto académico, donde la precisión y la colaboración son esenciales.

## 4.4 Inteligencia Artificial para Mejora de Contenido

El uso de modelos de inteligencia artificial en la mejora automática de textos técnicos se fundamenta en técnicas avanzadas de procesamiento de lenguaje natural (PLN), tales como la generación de resúmenes, corrección gramatical y estandarización del lenguaje técnico. Estas herramientas permiten enriquecer los documentos sin necesidad de intervención humana directa, optimizando la calidad y coherencia del contenido final. Al aplicar estas técnicas, los documentos se alinean con las mejores prácticas de documentación profesional, como las utilizadas en la industria del software, donde la claridad, precisión y consistencia son esenciales. Este enfoque no solo mejora la presentación del material, sino que también ahorra tiempo en el proceso de revisión, garantizando que los documentos sean más accesibles y efectivos para su uso académico y profesional.

## 4.5 Usabilidad y Experiencia del Usuario

El diseño centrado en el usuario (DCU) y los principios de usabilidad propuestos por Jakob Nielsen y otros expertos del área, son fundamentales para crear interfaces intuitivas y accesibles. Estos principios guían el desarrollo de la plataforma, garantizando que tanto estudiantes como docentes, sin importar su nivel técnico, puedan utilizar el sistema de manera fácil y eficiente. Al enfocarse en la simplicidad y la claridad en el diseño, la plataforma busca minimizar la curva de aprendizaje y maximizar la adopción del sistema, permitiendo que los usuarios se concentren en sus tareas académicas sin complicaciones adicionales. La implementación de estos principios asegura una experiencia fluida y agradable, lo que facilita la integración del sistema en entornos educativos y mejora la eficiencia de los usuarios.

#### 4.6 Seguridad y Protección de la Información Académica

En entornos digitales, la seguridad de los datos es una prioridad, especialmente en el contexto de Perú. Este proyecto contempla principios de seguridad informática fundamentales, como la autenticación, el control de accesos y la protección de sesiones activas, con el objetivo de resguardar la información procesada en el sistema. Normativas como la Ley de Protección de Datos Personales en Perú (Ley N° 29733) y los estándares internacionales ISO/IEC 27001 sirven como base para establecer protocolos de seguridad robustos que aseguren la confidencialidad e integridad de la información académica. En conjunto, estos fundamentos teóricos respaldan el diseño y la implementación del sistema propuesto, garantizando no solo su viabilidad técnica y relevancia académica, sino también su alineación con estándares profesionales.

El marco teórico no solo justifica el desarrollo del sistema, sino que también lo posiciona como una herramienta innovadora capaz de responder a los desafíos actuales de la documentación académica y la formación en tecnologías emergentes en el contexto peruano.

### 5. Desarrollo de la Solución

1. El desarrollo del Sistema de Conversión y Organización de Documentos Técnicos en Markdown se estructura mediante el uso de herramientas y tecnologías modernas que garantizan una plataforma funcional, segura y con una experiencia de usuario optimizada.
2. En el backend se ha utilizado Flask, un microframework liviano y potente para Python que permite gestionar eficientemente la lógica de negocio, la autenticación de usuarios, la gestión de documentos y el control de versiones. Flask facilita la implementación de arquitecturas limpias, como el patrón Modelo-Vista-Controlador (MVC), adoptado en este proyecto para mantener una estructura clara y mantenible.
3. El sistema utiliza SQL Server como base de datos, accediendo a ella mediante el conector pyodbc, lo que proporciona una integración robusta y un rendimiento confiable para el manejo de documentos, usuarios y registros de versiones. La estructura del backend se organiza en carpetas dedicadas como controllers, models y utils, permitiendo modularidad y escalabilidad del sistema.
4. En cuanto al frontend, se emplean tecnologías estándar como HTML, CSS y JavaScript, organizadas dentro de los directorios static/css y static/js. Estas tecnologías permiten construir interfaces intuitivas y responsivas, donde los usuarios pueden cargar documentos, visualizar versiones anteriores, mejorar contenido y navegar en documentos convertidos con facilidad. Además, se han implementado scripts específicos como document.js y version_history.js para gestionar la interacción en tiempo real con las funcionalidades clave del sistema.

5. Para la interfaz gráfica, se utilizan archivos de plantilla HTML bajo el directorio templates, como document_conversion.html, improved_document.html y version_history.html, que definen las vistas principales del sistema. Estas plantillas trabajan de forma integrada con Flask y el motor de plantillas Jinja2 para renderizar contenido dinámico y personalizado para cada usuario.
6. El entorno de desarrollo se gestiona desde Visual Studio, que proporciona herramientas integradas para la depuración, pruebas y despliegue del sistema. Esta elección permite un flujo de trabajo eficiente y organizado, aprovechando extensiones útiles para el desarrollo en Python y conexión con bases de datos.
7. La solución adopta una arquitectura monolítica con MVC, que facilita la comprensión del sistema y su despliegue inicial. El sistema será implementado en la nube utilizando Azure, lo que garantiza una infraestructura escalable, segura y de alta disponibilidad. El uso de herramientas de automatización como Terraform permite gestionar los recursos en la nube de forma reproducible y eficiente, contribuyendo a una operación estable y alineada con prácticas modernas de desarrollo DevOps.

#### a. Análisis de Factibilidad (técnico, económica, operativa, social, legal, ambiental)

La factibilidad económica se fundamenta en un análisis detallado de los costos asociados con el desarrollo, implementación y mantenimiento del sistema. Este análisis abarcará varios aspectos, como las licencias de software, el hardware necesario, los costos de capacitación para los usuarios y los posibles gastos operativos continuos. Además, se llevará a cabo un análisis de retorno de inversión (ROI), considerando tanto los beneficios a corto como a largo plazo. Este análisis permitirá garantizar que la implementación del sistema sea financieramente viable, generando un impacto positivo en la eficiencia y rentabilidad de las instituciones académicas. A través de esta evaluación económica, se busca asegurar que el sistema no solo sea sostenible, sino que también aporte valor en términos de optimización de recursos y mejora de procesos.

#### Factibilidad Operativa

El sistema de conversión y organización de documentos técnicos en Markdown está diseñado para mejorar la gestión de la documentación académica, optimizando el tiempo y esfuerzo que los estudiantes y docentes invierten en estructurar archivos en formatos tradicionales como Word, PDF, HTML y TXT. Su implementación permitirá una mayor eficiencia en la conversión de documentos, asegurando que estos sean compatibles con plataformas colaborativas como GitHub.

Beneficios del sistema	Contexto
Automatización del proceso de conversión	Reduce la necesidad de intervención manual, minimizando errores en la conversión de formatos y asegurando una documentación estructurada.

Optimización de la organización académica	Los documentos serán generados en un formato estándar, facilitando su integración con sistemas de control de versiones y plataformas de documentación.
Accesibilidad y usabilidad	La plataforma estará diseñada con una interfaz intuitiva para que cualquier usuario pueda utilizarla sin necesidad de conocimientos avanzados.
Reducción de costos operativos	Al eliminar la dependencia de herramientas de pago para la conversión de documentos, se reducen costos para los estudiantes y la institución.
Compatibilidad y escalabilidad	El sistema podrá adaptarse a nuevas necesidades, permitiendo la incorporación de más funcionalidades en el futuro.

### Factibilidad Legal

El desarrollo del sistema deberá cumplir con las regulaciones legales y normativas vigentes en Perú, especialmente en lo que respecta a la protección de datos personales y la seguridad digital, asegurando que la documentación académica sea gestionada de manera ética y conforme a la legislación vigente en el país.

Principales regulaciones a considerar:

Ley de Protección de Datos Personales (LPDP)	La plataforma debe garantizar la confidencialidad de la información académica procesada. Esto implica la implementación de protocolos de seguridad para evitar accesos no autorizados y garantizar la protección de los datos de los usuarios.
Derechos de autor y propiedad intelectual	Se establecerán términos y condiciones claras para evitar conflictos relacionados con la conversión y almacenamiento de documentos protegidos por derechos de autor. La plataforma no podrá ser utilizada para plagio o distribución de contenido sin la debida autorización.
Normativas de seguridad informática	El sistema implementará cifrado de datos y autenticación de usuarios para prevenir filtraciones de información y accesos indebidos.
Uso responsable de la información	Se fomentará el uso ético de la plataforma, asegurando que los documentos convertidos sean utilizados con fines académicos legítimos y no con propósitos indebidos. No se identifican restricciones legales que impidan la implementación del sistema, siempre y cuando se cumplan con estas regulaciones y se establezcan mecanismos de cumplimiento normativo.

No se identifican restricciones legales que impidan la implementación del sistema, siempre y cuando se cumpla con estas regulaciones y se implementen mecanismos de cumplimiento normativo para asegurar que el sistema opere dentro del marco legal de Perú.

### Factibilidad Social

El impacto social del proyecto es positivo, ya que la implementación del sistema no solo mejorará la documentación académica, sino que también fomentará la digitalización y el uso de herramientas modernas en la educación.

<b>Accesibilidad y democratización de la documentación</b>	<ul style="list-style-type: none"> <li>● Al ofrecer una herramienta gratuita y de fácil acceso, se permitirá a los estudiantes de diferentes niveles mejorar la organización de sus documentos sin necesidad de software especializado de pago.</li> <li>● La plataforma será intuitiva y estará diseñada para reducir la curva de aprendizaje, facilitando su uso por parte de cualquier estudiante sin conocimientos técnicos avanzados.</li> </ul>
<b>Fomento de la educación digital</b>	<ul style="list-style-type: none"> <li>● La implementación de este sistema incentivará a los estudiantes a familiarizarse con estándares modernos de documentación, como Markdown y el uso de plataformas de control de versiones como GitHub.</li> <li>● Facilitará la adopción de mejores prácticas en la redacción y organización de documentos técnicos.</li> </ul>
<b>Impacto en la comunidad académica</b>	<ul style="list-style-type: none"> <li>● Docentes y estudiantes podrán mejorar la presentación de proyectos e informes académicos, fomentando una cultura de documentación estructurada y bien organizada.</li> <li>● Se promoverá la colaboración y el trabajo en equipo, al permitir que los estudiantes trabajen con documentación estándar compatible con sistemas de versionado.</li> </ul>
<b>Ética y buenas prácticas digitales</b>	<ul style="list-style-type: none"> <li>● La plataforma fomentará el uso adecuado de la documentación, promoviendo la transparencia y evitando malas prácticas como el plagio o la desorganización en la presentación de trabajos técnicos.</li> <li>● Se establecerán lineamientos claros sobre el uso responsable de la información y la importancia de mantener una documentación académica bien estructurada.</li> </ul>

### Factibilidad Ambiental

El proyecto tiene un impacto ambiental positivo al promover la digitalización de la documentación académica, eliminando la necesidad de imprimir materiales físicos y reduciendo así el uso de papel y la generación de residuos. La automatización de la conversión y organización de documentos optimiza los recursos tecnológicos, disminuyendo el consumo energético en comparación con procesos manuales. Además, al facilitar el acceso remoto a los documentos, se reduce la necesidad de transporte y desplazamientos, contribuyendo a la disminución de emisiones de CO<sub>2</sub>. El sistema también minimiza la

duplicación de archivos, optimizando el almacenamiento digital y evitando el desperdicio de recursos. Con el cumplimiento de normativas ambientales, como las iniciativas ecológicas en el ámbito educativo, se promueve el almacenamiento digital eficiente. Finalmente, la implementación del sistema sensibiliza a los estudiantes sobre la importancia del uso de tecnologías sostenibles y la reducción del impacto ambiental en el contexto académico.

<b>Reducción del Uso de Papel</b>	La digitalización y conversión de documentos técnicos a Markdown elimina la necesidad de imprimir materiales físicos, promoviendo la conservación de recursos naturales y reduciendo la generación de residuos.
<b>Eficiencia Energética</b>	La automatización de la conversión y organización de documentos optimiza el uso de recursos tecnológicos, disminuyendo el consumo energético en comparación con procesos manuales de formateo y estructuración.
<b>Impacto en la Huella de Carbono</b>	Al permitir el acceso remoto a documentos organizados en la plataforma, se reduce la necesidad de transporte y desplazamientos para la consulta de material académico, contribuyendo a la disminución de emisiones de CO <sub>2</sub> .
<b>Gestión de Residuos Digitales</b>	La estructuración automática y el control de versiones minimizan la duplicación innecesaria de archivos, optimizando el almacenamiento y evitando el desperdicio de recursos digitales.
<b>Cumplimiento de Normativas Ambientales</b>	El sistema fomenta prácticas sostenibles al promover el almacenamiento digital eficiente, alineándose con iniciativas ecológicas para la reducción del impacto ambiental en el ámbito educativo.
<b>Conciencia y Educación Ambiental</b>	La implementación del sistema impulsa el uso de tecnologías sostenibles, sensibilizando a los estudiantes sobre la importancia de la documentación digital estructurada y la reducción del impacto ambiental.

## b. Tecnología de Desarrollo

El desarrollo del Sistema de Conversión y Organización de Documentos Técnicos en Markdown se sustenta en una combinación de tecnologías modernas que permiten ofrecer una plataforma robusta, funcional y accesible para usuarios académicos. Este sistema está diseñado para facilitar la conversión, organización y gestión eficiente de documentos técnicos, optimizando el flujo de trabajo para los estudiantes de la Facultad de Ingeniería de Sistemas.

<b>Arquitectura de Estilo: Monolítica</b>	Se adopta una arquitectura monolítica que integra todos los componentes del sistema (interfaz, lógica de negocio y base de datos) dentro de una misma aplicación, lo cual simplifica el proceso de desarrollo, despliegue y mantenimiento inicial del sistema.
<b>Patrón de Diseño: Modelo-Vista-Controlador (MVC)</b>	El sistema se estructura siguiendo el patrón arquitectónico MVC, lo que facilita una clara separación de responsabilidades entre los módulos encargados de manejar la lógica del sistema (controllers), la interacción con los datos (models) y la interfaz de usuario (templates). Esta organización mejora la escalabilidad, mantenibilidad y claridad del código.
<b>FastApi</b>	Se utiliza FastAPI como microframework principal del backend. Su rapidez, eficiencia y soporte nativo para la tipificación permiten implementar funcionalidades como autenticación de usuarios, gestión de sesiones, control de versiones y procesamiento de archivos de manera eficiente y segura.
<b>Pyodbc</b>	Para la conexión a la base de datos SQL Server, se emplea el módulo pyodbc, que proporciona una integración confiable para realizar operaciones CRUD y mantener registros de usuarios, documentos y versiones generadas.
<b>Werkzeug</b>	Como toolkit subyacente a Flask, Werkzeug gestiona las solicitudes y respuestas HTTP, asegurando una comunicación fluida entre el cliente y el servidor dentro del entorno WSGI.
<b>HTML, CSS y JavaScript</b>	<p>La interfaz de usuario se construye utilizando tecnologías estándar del desarrollo web:</p> <p>HTML para la estructura de las páginas</p> <p>CSS para el diseño visual y responsivo</p> <p>JavaScript para las funcionalidades interactivas, como la gestión de versiones, carga de archivos y mejoras de contenido.</p>

<b>Visual Studio</b>	El proyecto se desarrolla dentro del entorno Visual Studio, el cual ofrece herramientas integradas para depuración, control de versiones, pruebas y despliegue, agilizando el ciclo de desarrollo en Python y tecnologías web.
<b>Despliegue en Azure (Arquitectura Monolítica + SQL Server)</b>	El sistema se despliega en la nube utilizando Microsoft Azure, lo que proporciona una infraestructura segura, escalable y de alta disponibilidad. Se incluye una instancia de SQL Server para el almacenamiento estructurado de documentos, versiones y datos de usuario.

Estas tecnologías trabajan en conjunto para ofrecer una plataforma académica eficiente que responde a las necesidades actuales en la gestión de documentación técnica. El sistema no solo optimiza procesos como la conversión y mejora automática de documentos, sino que también promueve buenas prácticas de documentación, versionado y seguridad digital.

### c. Metodología de implementación

La metodología de implementación del sistema Doc2Markdown se organiza en fases claramente definidas, con el objetivo de garantizar un desarrollo eficiente, modular y alineado con las necesidades del entorno académico. Cada fase del proceso sigue principios de ingeniería de software que aseguran tanto la calidad del producto final como su capacidad de adaptación a futuras necesidades. Esta estructura metodológica permite un enfoque flexible, asegurando que cada etapa cumpla con los estándares de calidad y que el sistema evolucione conforme a los requisitos del usuario.

<b>Planificación y Análisis</b>
Definición de Requerimientos: Se identificaron los requerimientos funcionales y no funcionales del sistema, incluyendo la conversión automatizada de documentos (Word, PDF, HTML, TXT) a Markdown, la mejora automática del contenido, el control de versiones, la generación de archivos de navegación (sidebar.md, footer.md) y la gestión de usuarios.
Análisis del Contexto Académico: Se evaluaron las necesidades actuales de los estudiantes y docentes de la Facultad de Ingeniería de Sistemas, así como las limitaciones existentes en cuanto al uso de herramientas de documentación técnica y plataformas de control de versiones.



Diseño
Arquitectura del Sistema: Se definió una arquitectura monolítica basada en el patrón Modelo-Vista-Controlador (MVC), con separación de responsabilidades en módulos como controllers, models, utils y templates, lo que facilita la escalabilidad y el mantenimiento del código.
Diseño de Interfaz (UI/UX): Se elaboraron las plantillas HTML que conforman la interfaz principal del sistema (conversión, historial de versiones, mejora de documentos), aplicando principios de usabilidad para garantizar una experiencia accesible a todo tipo de usuarios.

Desarrollo
Backend y Lógica del Negocio: Se desarrolló la lógica de conversión, navegación, autenticación y versionado utilizando Flask. Se integró pyodbc para gestionar la base de datos SQL Server, almacenando usuarios, documentos, logs de conversión y versiones.
Frontend Dinámico: Se implementó la interfaz del usuario con HTML, CSS y JavaScript, complementada con archivos como document.js, improved_document.js y version_history.js, para brindar interacción en tiempo real, carga dinámica y retroalimentación visual.

Pruebas
Pruebas Unitarias y de Integración: Se realizaron pruebas automatizadas y manuales para verificar que cada componente (conversor, historial, autenticación, navegación automática) funcione correctamente de manera aislada y en conjunto.
Pruebas de Usabilidad: Se efectuaron sesiones de prueba con estudiantes, simulando tareas reales, a fin de evaluar la facilidad de uso de la plataforma, detectar errores de navegación y ajustar la interfaz según la retroalimentación obtenida.

<b>Despliegue</b>
Implementación en Azure: La solución se desplegó en Microsoft Azure bajo un marco arquitectura monolítica con base de datos SQL Server, utilizando herramientas como Terraform para aprovisionar automáticamente la infraestructura en la nube.
Monitoreo y Rendimiento: Se establecieron mecanismos de monitoreo para verificar el rendimiento del sistema, tiempo de respuesta en la conversión de documentos y uso eficiente de recursos en la nube.

<b>Mantenimiento y Evolución</b>
Actualizaciones Periódicas: Se planificó un ciclo de mantenimiento para incorporar mejoras futuras como soporte para nuevos formatos, integración con Google Drive, y opciones avanzadas de edición colaborativa.
Escalabilidad y Soporte Técnico: La arquitectura y modularidad del sistema permiten su evolución hacia versiones más complejas o distribuidas en caso de expansión institucional.
Esta metodología garantiza que el desarrollo del sistema responda efectivamente a las necesidades reales del entorno académico, promoviendo prácticas modernas de documentación y colaborando con la formación técnica de los estudiantes.

## 6. Cronograma

El cronograma de desarrollo del proyecto, que iniciará el 20 de marzo de 2024, está diseñado para guiar de manera estructurada las distintas fases de implementación. La primera semana se dedica a la Planificación y Análisis, donde se definen los requisitos funcionales y no funcionales del proyecto, se realiza un análisis detallado del mercado objetivo y la competencia, y se establecen claramente los objetivos y el alcance del proyecto. La segunda semana se centra en el Diseño, enfocándose en la arquitectura de la aplicación utilizando Flask para el backend y HTML/CSS para el frontend, así como en la creación de diseños de interfaz de usuario intuitivos y atractivos.

Semana	Fechas	Fase / Actividades
Semana 1	20 – 26 marzo	<b>Planificación y Análisis</b> - Requisitos funcionales y no funcionales- Análisis del mercado y competencia- Definición de objetivos y alcance
Semana 2	27 marzo – 2 abril	<b>Diseño</b> - Arquitectura técnica (Flask + HTML/CSS)- Wireframes y diseño de interfaz de usuario- UX/UI
Semanas 3-5	3 – 23 abril	<b>Desarrollo del Backend</b> - Configuración del entorno- Lógica del servidor (Flask)- Rutas, modelos, controladores- Pruebas unitarias
Semanas 6-7	24 abril – 7 mayo	<b>Desarrollo del Frontend</b> - Maquetación con HTML/CSS- Integración con el backend- Validación de formularios
Semanas 8-9	8 – 21 mayo	<b>Integración de Componentes</b> - Conexión frontend-backend- Validación de flujos- Pruebas funcionales
Semana 10	22 – 28 mayo	<b>Pruebas Internas y Ajustes</b> - Corrección de bugs- Refactorización- Validación técnica
Semanas 11-12	29 mayo – 11 junio	<b>Pruebas de Usuario (UAT)</b> - Feedback de usuarios- Registro de mejoras- Validación funcional
Semanas 13-14	12 – 25 junio	<b>Mejoras y Documentación</b> - Implementación de mejoras- Documentación técnica y de usuario- Preparación para producción
Semanas 15-16	26 junio – 5 julio	<b>Despliegue y Cierre</b> - Pruebas finales- Despliegue de la aplicación- Presentación del proyecto- Evaluación final

## 7. Presupuesto

Hablar de presupuestos manejables en el contexto del desarrollo del Sistema de Conversión y Organización de Documentos Técnicos en Markdown implica establecer una planificación financiera clara y eficiente que permita optimizar los recursos disponibles sin comprometer la calidad técnica ni el cumplimiento de los objetivos académicos del proyecto.

Dado que este sistema está orientado a mejorar los procesos de documentación en entornos universitarios, se busca mantener un equilibrio entre funcionalidad, sostenibilidad y bajo costo operativo. Para ello, es necesario definir y proyectar los siguientes costos principales:

### Costos Generales

CONCEPTO	CANTIDAD	COSTO UNITARIO	COSTO TOTAL
Material de oficina (cuadernos, lápices, papel, etc.)	-	-	S/. 100
Transporte (reuniones técnicas)	-	-	S/. 200
<b>Total</b>			<b>S/. 300</b>

### Costos operativos durante el desarrollo

<i>Concepto</i>	<i>Cantidad</i>	<i>Costo mensual (S/)</i>	<i>Total (6 meses) (S/)</i>
<i>Servicios básicos (agua, luz, internet)</i>	<i>1</i>	<i>300</i>	<i>1800</i>
<b>Total</b>			<b>1800</b>

### Costos del ambiente

Concepto	Costo estimado (S/)
Dominio web (.com o .org)	50
Hosting para plataforma	230
<b>Total</b>	<b>280</b>

Costos de infraestructura

Informe de Costos de Infraestructura - Proyecto

Este informe detalla los costos mensuales estimados de los recursos desplegados en la infraestructura del proyecto.

Total Estimado

Total mensual estimado: \$12.985 USD/mes

47 soles al mes.

Este reporte fue generado automáticamente el día: 2025-05-31

Recursos con Costo

Recurso	Componente	Cantidad	Unidad	Costo Mensual
azurerm_service_p...	Instance usage (B1)	730	hours	\$12.41
azurerm_mssql_dat...	Compute (serverless,	Variable	vCore-hours	Depende del uso
azurerm_mssql_dat...	GP_S_Gen5_2) Storage	5	GB	\$0.575

Costos de personal

Costos de sueldo del equipo de desarrollo durante la duración del proyecto.

Rol	Cantidad	Salario mensual (S/) por 5 horas al dia	Duración (meses)	Subtotal (S/)
Desarrollador Backend	1	600	6	3600
Desarrollador Frontend	1	600	6	3600
Analista de Requerimientos	1	600	3	1800
Total				9000

**Costos totales del desarrollo del sistema**

<b><i>Categoría</i></b>	<b><i>Costo total (S/)</i></b>
<i>Costos generales</i>	<i>300</i>
<i>Costos operativos</i>	<i>1800</i>
<i>Costos del ambiente</i>	<i>280</i>
<i>Costos de infraestructura</i>	<i>280</i>
<i>Costos de personal</i>	<i>9000</i>
<b><i>Costo total del proyecto</i></b>	<b><i>11,660</i></b>

## 8. Conclusiones

- El desarrollo del Sistema de Conversión y Organización de Documentos Técnicos en Markdown ha logrado cumplir de manera satisfactoria los objetivos generales y específicos planteados en el inicio del proyecto. La plataforma resultante representa una solución funcional, confiable y técnicamente sólida para mejorar la gestión de documentación académica en entornos universitarios.
- Uno de los logros más destacados ha sido la implementación de un sistema automatizado de conversión de documentos desde formatos tradicionales (Word, PDF, HTML, TXT) hacia el formato Markdown, ampliamente utilizado en plataformas como GitHub. Esta funcionalidad ha permitido a los usuarios mantener la estructura y semántica del contenido original, ahorrando tiempo y reduciendo errores en el proceso de migración documental.
- Además, el sistema integra de forma efectiva herramientas avanzadas como la mejora de contenido mediante inteligencia artificial, la generación automática de archivos de navegación (sidebar.md, footer.md) y el control de versiones. Estas características han fortalecido la utilidad de la plataforma, ofreciendo una experiencia moderna y alineada con las prácticas profesionales actuales de documentación.
- En materia de seguridad, se han incorporado mecanismos esenciales como la autenticación de usuarios, cierre de sesión seguro y control de acceso a los documentos. Estas medidas aseguran la protección de la información académica procesada, promoviendo un entorno confiable y ético para los usuarios.

En resumen, el proyecto ha superado las expectativas iniciales al no solo cumplir con los requerimientos técnicos, sino también establecer una base escalable para futuras mejoras. La experiencia adquirida en su desarrollo aporta valor significativo a la formación de los estudiantes involucrados y demuestra la viabilidad de aplicar buenas prácticas de ingeniería de software en soluciones destinadas al ámbito educativo.

## Recomendaciones

- ✓ Con base en la experiencia adquirida durante la ejecución del proyecto, se proponen las siguientes recomendaciones para fortalecer proyectos similares en el futuro:
- ✓ Incorporar herramientas de gestión de proyectos (como Trello, Jira o GitHub Projects) que faciliten la colaboración entre los miembros del equipo y el seguimiento de tareas en cada fase del desarrollo.
- ✓ Implementar pruebas automatizadas y revisiones de código periódicas, con el fin de mantener altos estándares de calidad y detectar errores desde etapas tempranas.
- ✓ Fomentar la capacitación continua del equipo en nuevas tecnologías, frameworks y metodologías ágiles, lo que permitirá una adaptación rápida a contextos técnicos cambiantes.
- ✓ Establecer métricas de desempeño claras, tanto funcionales como de experiencia del usuario, que permitan evaluar el éxito del proyecto y tomar decisiones informadas en futuras iteraciones.
- ✓ Promover una cultura de retroalimentación constructiva y mejora continua, que favorezca el aprendizaje colaborativo y la evolución constante del sistema y del equipo de desarrollo.



## **Bibliografía**

Novotný, V. (2022). Markdown 2.15.0: What's new?. TUGboat.  
<https://doi.org/10.47397/tb/43-1/tb133novotny-markdown>.

Pandey, P. (2025). Python and Its Implications. INTERANTIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING AND MANAGEMENT.  
<https://doi.org/10.55041/ijsrem41465>.

Relan, K. (2019). Beginning with Flask. Building REST APIs with Flask.  
[https://doi.org/10.1007/978-1-4842-5022-8\\_1](https://doi.org/10.1007/978-1-4842-5022-8_1).

Shao, Z., Dai, D., Guo, D., Liu, B., Wang, Z., & Xin, H. (2024). DeepSeek-V2: A Strong, Economical, and Efficient Mixture-of-Experts Language Model. ArXiv, abs/2405.04434.  
<https://doi.org/10.48550/arXiv.2405.04434>.

**Anexos**

Anexo 01 Informe de Factibilidad

Anexo 02 Documento de Visión

Anexo 03 Documento SRS

Anexo 04 Documento SAD

Anexo 05 Manuales y otros documentos