



UNIVERSIDAD PRIVADA DE TACNA

FACULTAD DE INGENIERIA

Escuela Profesional de Ingeniería de Sistemas

Proyecto *DocuCode-AI*

Curso: *PATRONES DE SOFTWARE*

Docente: *PATRICK JOSE CUADROS QUIROGA*

Integrantes:

Jose Luis Jarro Cachi - 2020067148

Farley Rodrigo Eduardo Viveros Blanco - 2020066896

Ronal Daniel Lupaca Mamani - 202006146

**Tacna – Perú
2025**

CONTROL DE VERSIONES					
Versión	Hecha por	Revisada por	Aprobada por	Fecha	Motivo
1.0	MPV	ELV	ARV	3/04/2025	Versión Original
2.0	MPV	ELV	ARV	30/05/2020	mejora

Sistema *DocuCode-AI* Informe de Factibilidad

Versión 2.0

CONTROL DE VERSIONES					
Versión	Hecha por	Revisada por	Aprobada por	Fecha	Motivo
1.0	MPV	ELV	ARV	3/04/2025	Versión Original
2.0	MPV	ELV	ARV	30/05/2020	mejora

1. Introducción

1.1 Propósito

El presente documento describe la visión del proyecto DocuCode-AI, un sistema web orientado a estudiantes universitarios de ingeniería de software y programación, cuyo objetivo principal es facilitar la documentación técnica de proyectos mediante la generación automática de diagramas UML. A partir del código fuente cargado por el usuario, la plataforma permite obtener diagramas como clases, casos de uso, secuencia, actividad, componentes y paquetes, necesarios para los informes académicos y presentaciones finales. El sistema busca reducir el esfuerzo manual y los errores en la elaboración de documentación, mejorando así la productividad y el aprendizaje.

1.2 Alcance

DocuCode-AI permite a los usuarios cargar archivos de código fuente o carpetas comprimidas (ZIP), y obtener como resultado una representación estructurada del sistema en forma de diagramas UML. De manera opcional, el sistema podrá analizar el contenido del código para mejorar la precisión de los diagramas y brindar comentarios automáticos, aunque esta funcionalidad no es prioritaria en la primera versión.

Las funcionalidades clave incluyen:

- Generación automática de **diagramas UML**: clases, casos de uso, secuencia, actividad, componentes y paquetes.
- Exportación de los diagramas en formatos listos para usar en informes: **PDF, imagen (PNG/SVG) y Markdown**.
- Interfaz accesible y amigable, compatible con navegadores modernos.

1.3 Definiciones, Siglas y Abreviaturas

- IA: Inteligencia Artificial
- UML: Unified Modeling Language
- API: Interfaz de Programación de Aplicaciones
- VPS: Servidor Privado Virtual
- SSL: Secure Socket Layer

1.4 Referencias

- Fowler, M. (2002). Patterns of Enterprise Application Architecture.
- <https://marp.app/>
- <https://www.terraform.io/>
- Documentación de OpenAI API y PlantUML

1.5 Visión General

DocuCode-AI surge como una solución académica orientada a estudiantes universitarios que necesitan documentar sus proyectos de programación. El sistema permite generar automáticamente diagramas UML a partir del código fuente, eliminando la necesidad de crear estos elementos manualmente. A través de una

interfaz intuitiva y accesible, los estudiantes podrán visualizar la estructura lógica de sus sistemas y mejorar la presentación técnica de sus trabajos. Aunque opcionalmente el sistema puede realizar análisis de código, su enfoque principal es la generación precisa y automatizada de diagramas UML para tareas como informes, tesis y entregables de cursos.

2. Posicionamiento

2.1 Oportunidad de negocio

En el contexto académico actual, los estudiantes enfrentan dificultades al documentar sus proyectos, especialmente al generar diagramas UML conforme a los estándares solicitados por docentes. Muchos recurren a herramientas complejas o a crear los diagramas manualmente, lo que incrementa el tiempo invertido y la posibilidad de errores.

DocuCode-AI responde a esta necesidad proporcionando una herramienta ágil y automatizada que genera diagramas precisos a partir del código fuente, facilitando la creación de documentación técnica de calidad sin requerir experiencia previa en herramientas especializadas.

2.2 Definición del problema

Actualmente, los estudiantes enfrentan los siguientes desafíos:

- Dificultad en la elaboración manual de diagramas UML (clases, casos de uso, actividad, etc.).
- Falta de retroalimentación visual inmediata sobre la estructura de su código.
- Limitado acceso a herramientas intuitivas para generar documentación técnica estándar.
- Desconocimiento de las buenas prácticas de documentación exigidas en el entorno académico.

3. Descripción de los interesados y usuarios

3.1 Resumen de los interesados

- **Estudiantes universitarios:** Son los principales beneficiarios del sistema, ya que utilizan la plataforma para generar diagramas UML y mejorar la documentación de sus proyectos académicos.
- **Docentes:** Actúan como facilitadores y revisores. Pueden recomendar el uso del sistema como parte de la evaluación de prácticas y trabajos académicos.
- **Universidad Privada de Tacna (UPT):** Institución promotora del uso de herramientas tecnológicas en el proceso de enseñanza-aprendizaje. Puede integrar la plataforma como parte de su ecosistema académico digital.

3.2 Resumen de los usuarios

- **Usuarios autenticados:** El sistema permite iniciar sesión para registrar un historial de análisis por estudiante, facilitando el seguimiento y la mejora continua.

- Usuarios ocasionales: También se contempla el uso sin registro, orientado a generar diagramas rápidos sin almacenar información.
- Acceso multiplataforma: El sistema es accesible desde navegadores modernos (como Chrome, Firefox, Edge) en computadoras de escritorio, laptops, tablets y dispositivos móviles.

3.3 Entorno de usuario

DocuCode-AI es un sistema web responsive que funciona en cualquier dispositivo con acceso a internet. No requiere instalación previa y su interfaz está diseñada para ser intuitiva, incluso para estudiantes sin experiencia técnica avanzada. La generación de diagramas es automática tras subir un archivo comprimido con el código fuente del proyecto.

3.4 Perfiles de los interesados

Interesado	Rol	Necesidad
Estudiante	Usuario principal	Generar automáticamente diagramas UML para incluir en la documentación de sus proyectos académicos.
Docente	Facilitador	Supervisar la calidad de los entregables documentados y promover el uso de herramientas que fomenten buenas prácticas.
Universidad	Promotor institucional	Impulsar el uso de tecnologías innovadoras que optimicen el aprendizaje en la formación profesional.

3.5 Perfiles de los Usuarios

- Usuario estudiante: Principal usuario del sistema, orientado a quienes necesitan documentar proyectos con diagramas como parte de su trabajo universitario. No requiere conocimientos técnicos avanzados.
- Usuario docente: Puede utilizar el sistema para generar ejemplos, recomendar su uso a estudiantes o validar la documentación presentada.
- Usuario administrador: Encargado de gestionar la infraestructura del sistema, mantener la plataforma operativa y actualizar funcionalidades.

3.6 Necesidades de los interesados y usuarios

- Acceso rápido y sencillo desde cualquier dispositivo.
- Generación automática de diagramas UML (clases, casos de uso, actividades, componentes, etc.).
- Interfaz amigable para usuarios no técnicos.
- Opcionalmente, posibilidad de obtener retroalimentación básica del código fuente.

- Capacidad de exportar la documentación en formatos comunes (PDF, Markdown).
- Historial de generación por usuario para seguimiento y control.

4. Vista General del Producto

4.1 Perspectiva del producto

DocuCode-AI es un sistema web accesible desde cualquier navegador moderno, diseñado para asistir a estudiantes y docentes en la generación automatizada de diagramas UML a partir de archivos de código fuente. La plataforma se ejecuta en un backend desarrollado en PHP y Python, e integra tecnologías de inteligencia artificial como OpenAI (de forma opcional) para interpretar el contenido del código. Los resultados generados se almacenan por usuario y pueden ser exportados en formatos estándar como PDF y Markdown, facilitando su incorporación en la documentación académica de proyectos.

4.2 Resumen de capacidades

- Subida de archivos de código fuente o carpetas comprimidas.
- Generación automática de diagramas UML:
 - Diagrama de clases
 - Diagrama de casos de uso
 - Diagrama de actividad
 - Diagrama de componentes
 - Diagrama de paquetes
 - Diagrama de secuencia
- Exportación de diagramas en PDF y Markdown.
- Historial de generación asociado a cada usuario autenticado.
- Opcional: análisis semántico del código con comentarios automáticos y evaluación de calidad.
- Opcional: detección de código duplicado.

4.3 Suposiciones y dependencias

- El sistema requiere acceso estable a la API de OpenAI para funciones de análisis opcionales.
- El despliegue se realizará en un servidor VPS (como Elastika) con soporte para PHP y Python.
- Se asume que los usuarios utilizarán navegadores compatibles (Chrome, Firefox, Edge).
- Los archivos subidos deben contener código en lenguajes compatibles como Java, Python, PHP, JavaScript, entre otros.

4.4 Costos y precios

Se estima que los costos operativos anuales del sistema DocuCode-AI oscilan entre **S/ 150 y S/ 250**, cubriendo servicios como infraestructura VPS (Elastika), nombre de dominio, certificados SSL y uso moderado de la API de OpenAI (para funciones opcionales).

El sistema será de **uso gratuito para estudiantes y docentes en entornos académicos**, con posibilidad de ampliar funcionalidades bajo un modelo freemium en el futuro.

4.5 Licenciamiento e instalación

El sistema será distribuido bajo la **licencia MIT**, permitiendo su libre uso, modificación y redistribución para fines educativos.

La instalación se realizará de forma automatizada mediante scripts de **Terraform**, desplegando el sistema completo en un **servidor VPS autogestionado**, incluyendo todas las dependencias necesarias.

5. Características del producto

- Subida de archivos individuales o carpetas comprimidas (.zip, .rar).
- Generación automática de diagramas UML dinámicos con **PlantUML**:
 - Clases
 - Casos de uso
 - Actividad
 - Secuencia
 - Componentes
 - Paquetes
- Exportación de resultados en **PDF** y **Markdown**.
- Historial de proyectos y diagramas por usuario autenticado.
- Interfaz web intuitiva adaptada a usuarios sin experiencia técnica.
- Panel administrativo para gestión de usuarios y configuración del sistema.
- **Opcional:** análisis semántico del código fuente, evaluación de calidad y detección de duplicados.

6. Restricciones

- El sistema requiere conexión a internet para su uso.
- El análisis automatizado avanzado depende de servicios externos como la API de OpenAI.
- El sistema se aloja en un VPS autogestionado, por lo que requiere conocimientos mínimos para mantenimiento técnico.
- Los resultados generados pueden estar limitados por la calidad del código fuente proporcionado.

7. Rangos de calidad

- Precisión en el análisis > 90%.
- Tiempo de respuesta < 40 segundos por análisis.
- Accesibilidad en dispositivos móviles.
- Alta disponibilidad en servidor VPS.

8. Precedencia y Prioridad

- Generación automática de diagramas UML.
- Historial por usuario de archivos y diagramas generados.
- Exportación de documentación en PDF/Markdown.
- Panel administrativo básico para gestión.
- Análisis de código, evaluación de calidad y detección de duplicados (opcional y futura mejora).

9. Otros requerimientos del producto

- **Estándares legales:** cumplimiento de protección de datos.
- **Estándares de comunicación:** HTTPS, cifrado SSL.
- **Estándares de plataforma:** Navegadores compatibles y arquitectura cliente-servidor.
- **Estándares de calidad y seguridad:** Pruebas unitarias, escaneo con SonarQube, integración continua con GitHub Actions.

CONCLUSIONES

DocuCode-AI es una solución efectiva, viable y académicamente útil. Mejora el proceso de evaluación de código con inteligencia artificial, reduce el tiempo de revisión y aporta valor al entorno educativo con herramientas automatizadas.

RECOMENDACIONES

- Extender la cobertura a nuevos lenguajes de programación.
- Implementar métricas de análisis más detalladas.
- Evaluar integración con LMS como Moodle.

BIBLIOGRAFIA

- Fowler, M. *Patterns of Enterprise Application Architecture*.
- IEEE Std 830-1998. *Recommended Practice for Software Requirements Specifications*.

WEBGRAFIA

- <https://www.terraform.io/>
- <https://marp.app/>
- <https://platform.openai.com/docs>
- <https://plantuml.com/>
- <https://elastika.pe/>

10. GitHub Wiki y Roadmap del Proyecto

10.1 Contenido del Wiki del Proyecto

El repositorio oficial de DocuCode-AI en GitHub contiene un [Wiki](#) donde se detalla:

- Visión general del proyecto y objetivos.
- Guía de instalación del sistema.
- Descripción de funcionalidades principales.
- Manual de uso para docentes.
- Estructura del sistema y tecnologías utilizadas.
- Preguntas frecuentes (FAQ).

10.2 Roadmap del Proyecto

A continuación se presenta el cronograma tentativo para el desarrollo iterativo del sistema:

Fase	Tarea	Fecha Estimada	Estado
F1	Definición de requerimientos	15/04/2025	<input checked="" type="checkbox"/> Completado
F2	Diseño del sistema y estructura	20/04/2025	<input checked="" type="checkbox"/> Completado
F3	Implementación del backend PHP/Python	01/04/2025	<input checked="" type="checkbox"/> Completado
F4	Integración con OpenAI API	08/04/2025	<input checked="" type="checkbox"/> Completado
F5	Generación automática de UML	15/04/2025	<input checked="" type="checkbox"/> Completado
F6	Evaluación de código y duplicidad	28/05/2025	<input type="checkbox"/> Pendiente
F8	Pruebas y documentación final	28/05/2025	<input type="checkbox"/> En progreso
F9	Publicación del sistema en VPS	28/05/2025	<input type="checkbox"/> Pendiente