# Acta de Constitución de Proyecto

| **Empresa:** | Universidad Continental |
| --- | --- |
| **Nombre del proyecto:** | ReviDoc |
| **Tipo de proyecto:** | Proyecto académico de desarrollo de software |
| **Patrocinador:** | Universidad Continental |
| **Dueño del Producto:** | Camargo Ambicho Joussepe Josué, Lorenzo Masgo Josue Eli |
| **Gerente de Proyecto:** | Camargo Ambicho Joussepe Josué, Lorenzo Masgo Josue Eli |
| **Scrum Master:** | Camargo Ambicho Joussepe Josué, Lorenzo Masgo Josue Eli |

| **Propósito del documento** |
| --- |
| Este documento define la descripción general, los objetivos y los participantes del proyecto. Se relaciona principalmente con la autorización del inicio del proyecto.  Asimismo este documento brinda una descripción de la situación actual, los requisitos de alto nivel, criterios de éxito, riesgos y oportunidades. |

| **Propósito / Justificación:** |
| --- |
| Ofrecer una plataforma inteligente que facilite la revisión académica automática, mejorando la calidad de los trabajos y optimizando el tiempo de estudiantes y docentes. |
| **Breve descripción del proyecto:** |
| El proyecto ReviDoc consiste en el desarrollo de una plataforma web full-stack con inteligencia artificial y flujos automatizados que permitirá a estudiantes y docentes cargar documentos académicos, recibir correcciones de gramática, coherencia, estilo, citas y plagio, así como generar informes automáticos, garantizando calidad, eficiencia y sostenibilidad en los procesos de revisión académica. |
| **Alcance preliminar del proyecto:** |
| El proyecto ReviDoc comprende el diseño y desarrollo de una plataforma web con inteligencia artificial para la revisión de documentos académicos, incluyendo análisis de gramática, coherencia, citación y plagio, con integración de flujos automatizados y metodologías ágiles. No abarca la creación de sistemas de calificación docente ni integraciones con plataformas externas, limitándose al entorno definido de la universidad. |

| **Resultados esperados del proyecto / Beneficios:** | |
| --- | --- |
| Se espera contar con una plataforma funcional que permita a estudiantes y docentes revisar documentos académicos de manera rápida y confiable, reduciendo errores de forma y citación, asegurando mayor calidad en los trabajos presentados y optimizando tiempos de corrección. Como beneficios adicionales, el proyecto fortalece la transparencia académica, fomenta buenas prácticas de investigación y aporta a la mejora continua de la formación universitaria. | |
| **Requisitos de alto nivel del proyecto:** | |
| **Requisito** | **Criterio de Éxito** |
| Desarrollar una plataforma web full-stack utilizando el stack MERN (MongoDB, Express.js, React.js, Node.js) | Desarrollar la plataforma ReviDoc con el stack MERN, integrando funcionalidades de inteligencia artificial y automatización para la revisión académica. |
| Implementar funcionalidades de inteligencia artificial para corrección gramatical, coherencia, estilo y detección de plagio en documentos académicos. | Implementar los módulos principales: carga de documentos, revisión automática, verificación de citas, detección de plagio, generación de informes y notificaciones. |
| Incorporar un módulo de verificación de citas y referencias en formatos como APA e IEEE. | Asegurar que el sistema sea escalable, accesible y sostenible, con contenerización en Docker y base de datos en MongoDB. |
| Integrar flujos de automatización con n8n para gestionar la carga de documentos, revisiones, generación de informes y notificaciones. | Completar el desarrollo en un plazo máximo de 16 semanas (ciclo académico). |
| Asegurar que el sistema sea responsivo y accesible desde diferentes dispositivos (PC, tablets, móviles). | Mantener el cumplimiento del plan de trabajo y asegurar que el esfuerzo del equipo se mantenga dentro del presupuesto y recursos asignados por la universidad. |
| Implementar pruebas automatizadas (unitarias y de extremo a extremo) que garanticen calidad y confiabilidad del producto. | Alcanzar una cobertura mínima del 70% en pruebas automatizadas y superar el 90% en funcionalidad crítica. |
| Contenerizar la aplicación con Docker y docker-compose para facilitar despliegues escalables y sostenibles. | Elaborar la documentación técnica y de usuario junto con un informe final que refleje el impacto académico y ambiental del proyecto. |
| Documentar el proyecto de forma clara, incluyendo manual de usuario, diagramas de arquitectura y guía de instalación. | Presentar un video demostrativo funcional al final del proyecto. |
| Gestionar el desarrollo bajo una metodología ágil (Scrum o Kanban) con sprints, tableros, historias de usuario y retrospectivas. |  |
| Evaluar el impacto ambiental positivo del sistema, destacando la reducción en el uso de papel y la eficiencia en el consumo de recursos digitales. |  |
| **Hitos** | |
| * Inicio del proyecto y planificación (Semana 1). * Definición de requerimientos y diseño de la arquitectura del sistema (Semana 2–3). * Desarrollo del backend (API con Node.js + Express + MongoDB) (Semana 4–5). * Desarrollo del frontend (React con integración inicial al backend) (Semana 6–7). * Implementación de funcionalidades de IA (corrección, coherencia y detección de plagio) (Semana 8–9). * Integración de flujos de automatización con n8n (Semana 10). * Pruebas unitarias y E2E con Jest/Cypress (Semana 11–12). * Contenerización con Docker y despliegue en entorno de pruebas (Semana 13). * Documentación técnica y manuales de usuario (Semana 14). * Pruebas finales, ajustes y preparación del demo (Semana 15). * Entrega final y presentación del proyecto (video demostrativo + informe técnico) (Semana 16). | |
| **Riesgos** | |
| * Disponibilidad de personal calificado en el uso del stack MERN, inteligencia artificial y flujos de automatización (n8n). * Complejidad técnica en la integración de IA para revisión gramatical, coherencia y detección de plagio, lo que podría generar retrasos en el desarrollo. * Limitaciones de tiempo, dado que el proyecto debe completarse en un ciclo académico de 16 semanas. * Dependencia de servicios externos (APIs de IA, bases de datos en la nube, herramientas de automatización) que podrían presentar cambios en políticas, costos o disponibilidad. * Falta de recursos tecnológicos adecuados (infraestructura de pruebas, capacidad de cómputo para IA, servidores). * Riesgo de baja adopción por parte de usuarios finales (estudiantes y docentes) si la interfaz no es lo suficientemente intuitiva o confiable. * Problemas en la coordinación del equipo bajo metodología ágil si no se cumplen los roles, compromisos y comunicación efectiva. | |
| **Costo Preliminar Estimado / Presupuesto** | |
| | ***Categoría*** | ***Detalle*** | ***Monto total (S/)*** | ***Mensual (S/)*** | | --- | --- | --- | --- | | ***Servicios en la nube*** | *Dominio, MongoDB Atlas, servidor* | *1,200* | *300* | | ***APIs de Inteligencia Artificial*** | *Consultas masivas (~1000/día con modelo económico GPT-4o mini)* | *600 – 1,800* | *150 – 450* | | ***Herramientas de automatización*** | *n8n (suscripción/servidor)* | *500* | *125* | | ***Infraestructura y contenerización*** | *Docker, entornos de prueba y despliegue* | *400* | *100* | | ***Software y utilitarios*** | *GitHub Pro, Trello, librerías premium, Canva* | *400* | *100* | | ***Documentación y presentación*** | *Informe técnico, video demo, material gráfico* | *500* | *125* |   Total estimado del proyecto (4 meses): S/ 3,600 – 5,000  Promedio mensual: S/ 900 – 1,250 | |

| ***Interesados en el proyecto*** | | | | |
| --- | --- | --- | --- | --- |
| ***Interesado*** | ***Cargo*** | ***Rol*** | ***Teléfono*** | ***Email*** |
| *Estudiantes* | *Usuario final* | *Usuario final (beneficiario)* | *N/A* | *N/A* |
| *Docentes* | *Docentes universitarios* | *Usuario clave* | *N/A* | *N/A* |
| *Administrador académico* | *Área académica / Coordinador* | *Supervisor de procesos académicos* | *N/A* | *N/A* |
| *Equipo de desarrollo* | *Desarrolladores / QA* | *Equipo ejecutor* | *N/A* | *N/A* |
| *Gerente de Proyecto* | *Dirección de TI / Jefe de Proyecto* | *Responsable de gestión* | *N/A* | *N/A* |
| *Área de TI (Soporte)* | *Infraestructura y soporte técnico* | *Soporte técnico* | *N/A* | *N/A* |
| *Dirección académica* | *Alta dirección* | *Patrocinador (sponsor)* | *N/A* | *N/A* |
| *Proveedores de IA* | *API NLP (Hugging Face / OpenAI - Gemini)* | *Proveedor externo* | *N/A* | *N/A* |
| *Proveedor de Automatización* | *Plataforma n8n* | *Proveedor externo* | *N/A* | *N/A* |
| *Área legal / Ética* | *Comité académico / Ética* | *Aseguramiento de calidad ética y legal* | *N/A* | *N/A* |

| **Supuestos** |
| --- |
| | ID | Categoría | Supuesto/Restricción | Estado | Comentarios | | --- | --- | --- | --- | --- | | A1 | Habilidades | Suponemos que habrá suficiente talento en IA, NLP y automatización (n8n) para el desarrollo del sistema. | Pendiente | Existen especialistas en estas áreas en el mercado, pero la disponibilidad depende de costos. | | A2 | Operaciones | Suponemos que la infraestructura tecnológica de la universidad soportará la integración con la plataforma. | Pendiente | Puede requerirse ampliar servidores o servicios en la nube para procesamiento de textos | | A3 | Recursos | Suponemos que las licencias de herramientas (OpenAI o Gemini, Hugging Face, n8n) estarán disponibles y accesibles. | Pendiente | El costo de las licencias puede variar y afectar el presupuesto asignado. | | A4 | Tiempo | Suponemos que los usuarios (docentes y estudiantes) entregarán sus trabajos en formato digital estándar. | | Confirmado | | --- |  |  | | --- | | La mayoría de entregas académicas ya se realizan en Word o PDF, lo que facilita la integración. | |
| **Restricciones** |
| | ID | Categoría | Supuesto/Restricción | Estado | Comentarios | | --- | --- | --- | --- | --- | | C1 | Regulatorio | La plataforma debe cumplir con la normativa de protección de datos (GDPR/LPDP) en almacenamiento seguro. | Activo | El sistema almacenará trabajos académicos que contienen datos personales de los estudiantes. | |

| **Autorización de proyecto** | | | |
| --- | --- | --- | --- |
| **Nombre** | **Cargo** | **Firma** | **Fecha** |
|  | Patrocinador |  |  |
|  | Patrocinador |  |  |