“**Año de la Recuperación y Consolidación de la Economía Peruana**”



**TALLER DE PROYECTOS II**

**NRC: 62100**

**PROYECTO:**

**Tutor Virtual de Lectura Crítica**

**DOCENTE:**

**Daniel GAMARRA MORENO**

**INTEGRANTES:**

* ***Castro Velarde Anyulieth Nayeli***
* ***Contreras Bullon Daniel***
* ***Iturrizaga Campean Geraldine***
* ***Llacza Isidro José Miguel***
* ***Palacios Quincho Melany Alana***
* ***Quispe Ubaldo Alfredo***
* ***Roman Briceño Jhonn Arturo***
* ***Villena Ochoa Yorshyo Artemio***

**HUANCAYO - PERÚ   
*2025-20***

**Acta de Constitución del Proyecto**

**Proyecto:** Tutor Virtual de Lectura Crítica  
**Fecha:** 26/08/2025  
**Equipo:** Taller de Proyectos 2 - Ingeniería de Sistemas e Informática

1. **Alcance**

El proyecto consiste en el desarrollo de una aplicación web full-stack con tecnología MERN (MongoDB, Express.js, React.js, Node.js) que funcionará como un tutor virtual para fortalecer habilidades de lectura crítica.

El sistema permitirá a los usuarios leer textos, responder preguntas generadas automáticamente por IA, recibir retroalimentación y visualizar su progreso. Además, integrará automatización con n8n para gestión de sesiones, recordatorios y notificaciones.

El alcance inicial incluye:

* Desarrollo frontend con React.js, diseño responsive y manejo de estado con Redux o Context API.
* Backend con Node.js + Express.js, API REST documentada.
* Base de datos MongoDB Atlas para almacenamiento de textos, usuarios y resultados.
* Funcionalidades de IA:
* Generación de preguntas con NLP (Hugging Face).
* Clasificación de sesgos en textos.
* Automatización con **n8n** para:
* Envío de recordatorios diarios.
* Registro automático del avance.
* Panel de administración y reportes.

1. **Objetivos**

**Objetivo General**

Implementar un tutor virtual interactivo basado en inteligencia artificial y automatización que fomente la lectura crítica en estudiantes, docentes y usuarios interesados.

**Objetivos Específicos**

1. Diseñar una interfaz intuitiva y accesible con [React.js](http://react.js).
2. Implementar un backend con Node.js y Express que gestione actividades, usuarios y resultados.
3. Incorporar dos funcionalidades de IA: generación automática de preguntas y detección de sesgos.
4. Integrar flujos de automatización con n8n para mejorar la gestión del aprendizaje.
5. Garantizar la calidad mediante pruebas automatizadas (unitarias y E2E).
6. Documentar el proyecto siguiendo buenas prácticas de ingeniería de software y sostenibilidad.
7. **Entregables**

* Acta de Constitución del Proyecto.
* Diagrama de Stakeholders.
* Código fuente en GitHub (con Git Flow aplicado).
* Prototipo navegable de la aplicación web.
* Pruebas automatizadas (Jest, Cypress).
* Contenerización con Docker (docker-compose con frontend, backend, DB y n8n).
* Informe técnico final (con sección de impacto ambiental).
* Presentación oral y video demostrativo.

1. **Roles del Equipo (8 integrantes)**

**Stakeholders internos:**

* **Patrocinador Da:** Docente del curso.
* **Project Manager:** [Integrante 1] – planificación, control, cronograma.
* **Product Owner:** [Integrante 2] – define historias de usuario y prioriza backlog.
* **Scrum Master:** [Integrante 3] – facilita ceremonias ágiles y seguimiento.
* **Frontend Developer (UI/UX):** *Alfredo Quispe Ubaldo* – React.js, UX/UI.
* **Backend Developer:** [Integrante 4] – API REST con Node.js + Express.
* **Database Admin:** [Integrante 5] – modelado y gestión de MongoDB.
* **QA/Tester:** [Integrante 6] – pruebas unitarias y E2E.
* **DevOps:** *Alfredo Quispe Ubaldo* – CI/CD y despliegue con Docker.
* **Documentación & Presentación:** [Integrante 8] – informe final y exposición.

**Stakeholders externos:**

* Usuarios finales (estudiantes, docentes): beneficiarios principales.
* Clientes (instituciones educativas): potencial adopción del sistema.
* Proveedores: herramientas tecnológicas (Hugging Face, MongoDB Atlas, Docker).
* Reguladores: estándares de accesibilidad y seguridad.

1. **Recursos**

* **Humanos:** 8 integrantes con roles asignados.
* **Tecnológicos:** MERN Stack, Hugging Face, n8n, Docker, GitHub, Trello/GitHub Projects, Jest, Cypress.Materiales: Laptops personales, internet, bibliografía del curso.
* **Materiales:** Computadoras, internet, bibliografía del curso.

1. **Plazos Preliminares**

| **Semana** | **Actividad Principal** | **Entregable** |
| --- | --- | --- |
| 1–2 | Elaboración del Acta y Diagrama de Stakeholders | Acta inicial |
| 3–4 | Diseño de arquitectura y base de datos | Diagramas, mockups |
| 5–7 | Desarrollo inicial Frontend + Backend | Versión preliminar |
| 8 | Entrega primera versión (MVP) | Prototipo navegable |
| 9–11 | Integración de IA (preguntas, sesgos) | Funcionalidades IA |
| 12–13 | Automatización con n8n + pruebas automatizadas | Flujo automatizado + test |
| 14 | Contenerización con Docker y optimización | docker-compose |
| 15 | Versión final del sistema | Producto completo |
| 16 | Presentación y defensa del proyecto | Exposición |

1. **Riesgos Iniciales**

* Integración compleja del stack MERN.
* Modelos de IA con baja precisión en preguntas o sesgos.
* Retrasos en la coordinación frontend-backend.
* Dificultades en la automatización con n8n.
* Errores en contenerización y despliegue.
* Baja participación de miembros del equipo.