**Reporte de desarrollo de software**

**Proyecto EvaEduca: Plataforma educativa con asistente virtual y autoevaluación AI**

**Versión 1.1**

**Historial de Revisiones**

| **Fecha** | **Versión** | **Descripción** | **Autor** |
| --- | --- | --- | --- |
| 7/05/2024 | 1.0 | Versión preliminar como una propuesta de desarrollo. | Jefe de proyecto |
| 20/06/2024 | 1.1 | Informar sobre los puntos avanzados | Jefe de proyecto |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

### 

### **Índice**

### 1. Introducción

### 2. Descripción del Proyecto

#### 3. Requisitos

### 4. Arquitectura del Software

### 5. Planificación del Desarrollo

### 6. Gestión de Riesgos

### 7. Conclusiones

#### **1. Introducción**

* **Objetivo del documento**:

El propósito de este reporte es proporcionar una visión detallada del desarrollo del proyecto EvaEduca. Este documento abarca desde la planificación inicial hasta el estado actual del proyecto, describiendo los objetivos, alcance, arquitectura, planificación, gestión de riesgos y próximos pasos.

* **Alcance del proyecto**:

El proyecto EvaEduca tiene como objetivo desarrollar una plataforma educativa con un asistente virtual y autoevaluación basada en inteligencia artificial. Esta plataforma permitirá a los docentes realizar evaluaciones más eficientes y precisas, proporcionando retroalimentación automática a los estudiantes y facilitando el acceso a sus calificaciones.

#### **2. Descripción del Proyecto**

* **Visión y objetivos**:

La visión del proyecto EvaEduca es revolucionar el proceso de evaluación educativa mediante la integración de inteligencia artificial, mejorando la eficiencia y precisión en la revisión de exámenes y evaluaciones. Los principales objetivos son:

* Automatizar la revisión de exámenes utilizando IA.
* Proporcionar retroalimentación detallada y personalizada a los estudiantes.
* Agilizar el proceso de evaluación y gestión de cursos.
* **Descripción general del sistema**:

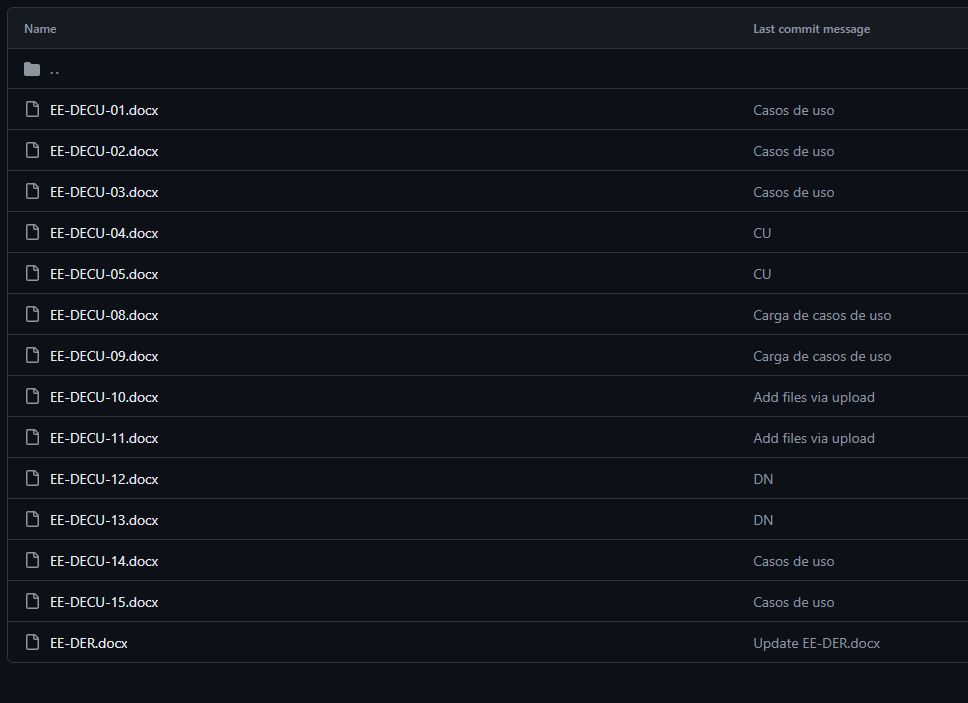
EvaEduca es una plataforma educativa que permite a los profesores gestionar cursos y evaluaciones, subir rúbricas de evaluación y respuestas de los alumnos, y realizar revisiones automáticas mediante IA. Los estudiantes pueden visualizar sus evaluaciones, recibir retroalimentación y realizar reclamos en caso de discrepancias.

* **Stakeholders**:
* **Superadministrador:** Gestiona la creación de instituciones educativas y administradores.
* **Administrador:** Gestiona cuentas de profesores y alumnos, y cursos dentro de su institución.
* **Profesor:** Gestiona cursos, crea evaluaciones, y revisa exámenes.
* **Alumno:** Visualiza cursos y evaluaciones, y puede realizar reclamos.

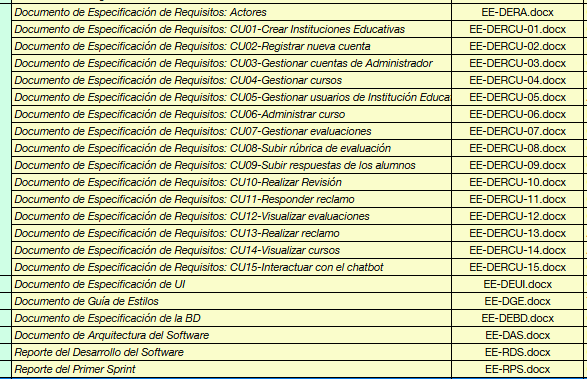
#### **3. Requisitos**

* **Requisitos funcionales**:

Los requisitos funcionales se encuentran de manera detallada en el repositorio del proyecto:



De la misma forma, se encuentran los nombres explicados en el cronograma del proyecto:



* **Requisitos no funcionales**:
* **Rendimiento:** El sistema debe procesar las evaluaciones en un tiempo razonable, con una latencia mínima.
* **Seguridad:** Los datos de los usuarios y evaluaciones deben estar protegidos contra accesos no autorizados.
* **Usabilidad:** La plataforma debe ser intuitiva y fácil de usar tanto para profesores como para alumnos.

#### **4. Arquitectura del Software**

El documento de arquitectura se encuentra en la carpeta de Diseño del repositorio, entre lo resultante o lo más importante están las capas de nuestro software:

* **Capas arquitectónicas**
  + **Capa de presentación**

Es la capa que ve el usuario y con la que interactúa. Se encarga de presentar el sistema al usuario final, recopilando y mostrando los datos necesarios para la aplicación. En el proyecto se usarán herramientas como HTML, CSS y JavaScript.

* **Capa de lógica de negocios**

Es el nivel medio donde se procesa la información obtenida del nivel superior mediante las reglas del negocio. Se desarrollará utilizando NodeJS y se incluirán las APIs necesarias.

* **Capa de datos**

Es el nivel de backend donde se almacenarán los datos mediante tablas de las diferentes entidades de la aplicación. El gestor de base de datos elegido será PostgreSQL.

#### **5. Planificación del Desarrollo**

La metodología adoptada para el desarrollo del proyecto es Scrum, un marco ágil que facilita la entrega incremental y la mejora continua del producto. Más detalles de la planificación se visualizan en el cronograma de proyecto del repositorio.

#### **6. Gestión de Riesgos**

*Identificación de riesgos:*

* **Retrasos en el desarrollo:** Riesgo de no cumplir con los plazos debido a problemas técnicos o de coordinación.
* **Fallas de seguridad:** Riesgo de vulnerabilidades que podrían comprometer la información de los usuarios.
* **Desempeño de IA:** Riesgo de que la IA no proporcione revisiones precisas o consistentes.

*Mitigación de riesgos:*

* **Retrasos en el desarrollo:** Implementar revisiones regulares de sprint y mantener una comunicación constante dentro del equipo.
* **Fallas de seguridad:** Realizar auditorías de seguridad periódicas y aplicar mejores prácticas de codificación.
* **Desempeño de IA:** Realizar pruebas exhaustivas y ajustar los algoritmos según los resultados de las pruebas.

#### **7. Conclusiones**

* **Estado actual del proyecto:**

El proyecto EvaEduca ha completado varias fases clave, incluyendo la definición de requisitos, diseño inicial, y desarrollo de los primeros módulos. Se ha implementado el código necesario para la evaluación del documento mediante IA.

* **Próximos pasos:**
* Desarrollar la capa de FrontEnd y BackEnd para culminar el proyecto