**Plataforma educativa con asistente virtual y autoevaluación AI**

**Documento de Arquitectura**

**DA**

Versión 1.0

**Lima, 24 de mayo del 2024**

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Versión** | **Descripción** | **Autor** |
| 24/05/2024 | 1.0 | Creación del documento. | Román Suyo, André Fabrizzio |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Control de Versiones**

**ÍNDICE**

**Introducción**

**Propósito**

**Alcance**

**Objetivos**

**Consideraciones generales**

**Definición de la arquitectura**

**Tecnologías usadas**

**Capas arquitectónicas**

**Capa de presentación**

**Capa de lógica de negocios**

**Capa de datos**

**Diseño de software**

**Propósito**

**Alcances**

**Objetivos**

**Impacto del software por módulos**

**Módulo Registrar nueva cuenta**

**Módulo Visualizar cursos**

**Módulo Gestionar evaluaciones**

**Módulo Subir respuestas**

**Módulo realizar reclamo**

**Módulo Responder reclamo**

**Módulo Visualizar evaluaciones**

**Módulo Administrar curso**

1. **Introducción**
   1. **Propósito**

El propósito del presente documento es mostrar la arquitectura del software que se ha planteado para el sistema EvaEduca. Identificamos todas las tecnologías a emplear en el desarrollo del proyecto tanto para la parte del backend, frontend y el control de versiones. También encontramos las diferentes capas arquitectónicas desde la capa de presentación hasta la de datos. Así mismo los procesos necesarios para la implementación. Finalmente, se muestran las diferentes interfaces para un mayor conocimiento del usuario acerca del producto a entregar.

* 1. **Alcance**
     + Proveer una visión general de la arquitectura de software del proyecto EvaEduca.
     + Indicar las tecnologías a emplear para el proyecto EvaEduca.
     + Desarrollar la arquitectura del proyecto EvaEduca.
     + Mostrar el diseño elaborado para el proyecto, resultante del análisis de los requerimientos del sistema.
  2. **Objetivos**
     + Establecer y especificar las tecnologías tanto para la parte backend y frontend con la que los analistas y desarrolladores podrán implementar el sistema.
     + Establecer una base para los analistas con la que podrán verificar la arquitectura del software planteada.
     + Establecer una plataforma de desarrollo colaborativo con la que el equipo podrá alojar el proyecto y hacerle el correcto seguimiento.
     + Permitir a los analistas y desarrolladores modificar la arquitectura planteada para que estos logren adaptarse de manera más adecuada a los requerimientos.

1. **Consideraciones generales**
   1. **Definición de la arquitectura**

Para el desarrollo del programa EvaEduca se trabajó con los niveles de presentación, aplicación y datos. De esta manera se puede trabajar de forma modular y mantenible. Separar las funciones de la aplicación en su interfaz, su nivel lógico y el backend hace que la aplicación sea muy ligera.

* 1. **Tecnologías usadas**
     + **HTML 5:** Especifica la estructura de las páginas web y sus contenidos.
     + **CSS 3:** Maneja el diseño y presentación de las páginas web.
     + **JavaScript:** Añade interactividad en la página web.
     + **React:** Biblioteca de JavaScript para construir interfaces de usuario.
     + **NodeJS:** Entorno de ejecución para JavaScript.
     + **PostgreSQL:** Sistema de gestión de bases de datos relacional.
     + **OAuth 2.0:** Protocolo para la autenticación y autorización.
     + **Git:** Software de control de versiones.
     + **GitHub:** Plataforma para alojar repositorios de Git.

1. **Capas arquitectónicas**
   1. **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.

* 1. **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.

* 1. **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.

1. **Diseño de software**
   1. **Propósito**

Contiene los módulos de diseño junto al nivel de impacto identificado en cada uno provocado por su implementación.

* 1. **Alcances**
     + Especificar los impactos dentro del sistema de acuerdo con los módulos establecidos.
     + Identificar los procesos que serán afectados luego de la implementación del sistema.
     + Detallar las entidades existentes en cada módulo y la información sobre sus campos.
  2. **Objetivos**

Documentar el impacto del software por módulos y las entidades presentes en cada uno con el propósito de simplificar futuros procesos de implementación del sistema.

1. **Impacto del software por módulos**
   1. **Módulo Registrar nueva cuenta**

**Medio:** Entidad que contiene la información de los usuarios que se registran en la plataforma.

* 1. **Módulo Visualizar cursos**

**Medio:** Entidad que contiene la información de los cursos disponibles en la plataforma.

* 1. **Módulo Gestionar evaluaciones**

**Alto:** Entidad que permite la gestión de evaluaciones por parte de los profesores.

* 1. **Módulo Subir respuestas**

**Alto:** Entidad que permite a los alumnos subir sus respuestas a las evaluaciones.

* 1. **Módulo realizar reclamo**

**Alto:** Entidad que permite a los alumnos realizar reclamos sobre sus evaluaciones.

* 1. **Módulo Responder reclamo**

**Alto:** Entidad que permite a los profesores responder a los reclamos realizados por los alumnos.

* 1. **Módulo Visualizar evaluaciones**

**Medio:** Entidad que permite a los alumnos visualizar sus evaluaciones.

* 1. **Módulo Administrar curso**

**Alto:** Entidad que permite a los profesores administrar sus cursos.