Documento Principal

Plataforma de Aprendizaje para Todos

|  |  |  |
| --- | --- | --- |
| **Revisor** | **Versión** | **Fecha** |
| Fiorella Mirano | V1.0 | 4 sept 2024 |
| Max Saavedra | V1.0 | 4 sept 2024 |
| Fiorella Mirano | V1.0 | 27 sept 2024 |

**Historial de Revisiones**

|  |  |  |  |
| --- | --- | --- | --- |
| **Revisor** | **Descripción** | **Versión** | **Fecha** |
| **Fiorella Mirano** | **Creación del**  **Documento** | **1.0** | **4 sept 2024** |
|  |  |  |  |

# Introducción

## Propósito

Este documento tiene como objetivo brindar una visión global y comprensible del diseño general de la plataforma, permitiendo al lector comprender su funcionamiento y la forma en que se relacionan los diferentes actores dentro de la misma.

## Descripción de la Plataforma

* **Usuarios:** Los usuarios podrán acceder a una amplia variedad de materiales textuales, participar en foros para realizar preguntas y buscar recursos adicionales.
* **Administrador:** El administrador se encarga de aprobar los materiales textuales publicados por los usuarios, asegurando la calidad y pertinencia del contenido disponible en la plataforma.

## Objetivo

La Plataforma de Aprendizaje para Todos tiene como objetivo crear un entorno dinámico y colaborativo, donde los usuarios trabajen juntos para facilitar el acceso a la información y el aprendizaje.

## Beneficio

* **Acceso a información:** Facilita el acceso a una amplia gama de materiales textuales, incluyendo libros y materiales digitales.
* **Comunidades de aprendizaje:** Fomenta la colaboración entre usuarios a través de los foros de preguntas y respuestas.
* **Control de calidad:** El proceso de aprobación de materiales por parte del administrador asegura la calidad y la pertinencia del contenido.

# Arquitectura de la Plataforma

Para el diseño de la Plataforma de Aprendizaje para Todos, se ha elegido una arquitectura de tres capas (Presentación, Aplicación y Persistencia), la cual se adapta a las necesidades específicas de una plataforma web educativa.

## Arquitectura de Tres Capas

* **Presentación:** Esta capa se encarga de la interfaz gráfica de usuario (GUI), la cual es accesible a través del navegador web. Esta capa se encarga de la interacción con el usuario, la visualización de la información y la recepción de las solicitudes del usuario.
* **Aplicación:** Esta capa se encarga de la lógica de negocio de la plataforma. Gestiona las solicitudes provenientes de la capa de presentación, interactúa con la capa de persistencia para acceder a la base de datos y realiza las operaciones necesarias para procesar la información.
* **Persistencia:** Esta capa se encarga de almacenar y recuperar datos de la base de datos. La capa de aplicación se comunica con esta capa para acceder a la información necesaria para realizar las operaciones del sistema.

## Ventajas de la Arquitectura de Tres Capas

* **Independencia de capas:** Cada capa es independiente de las demás, lo que facilita el desarrollo, la mantenibilidad y la actualización del sistema.
* **Reutilización de código:** La lógica de negocio en la capa de aplicación puede ser reutilizada en diferentes interfaces de usuario (por ejemplo, una aplicación móvil o un sistema de escritorio).
* **Escalabilidad:** Se puede escalar cada capa de forma independiente para satisfacer las demandas cambiantes de la plataforma.

## Desarrollo Web

Se toma en cuenta la naturaleza web del sistema para implementar una interfaz gráfica de usuario accesible a través de un navegador web.

# Metas y restricciones arquitectónicas

## Metas

* **Seguridad:** La plataforma requerirá autenticación de usuarios con código de usuario y contraseña para acceder al sistema. Además, se implementarán controles de acceso basados en el perfil del usuario, limitando las opciones disponibles y protegiendo la información confidencial según el rol del usuario (usuario o administrador).
* **Usabilidad:** La plataforma debe ser fácil de usar e intuitiva para todos los usuarios, independientemente de su experiencia previa con plataformas digitales.
* **Escalabilidad:** El sistema debe ser capaz de manejar un número creciente de usuarios y materiales textuales sin afectar el rendimiento.

## Restricciones

* **Base de Datos:** Para asegurar un sistema escalable y eficiente, se recomienda utilizar una base de datos NoSQL como MongoDB. MongoDB es ideal para el almacenamiento de grandes volúmenes de datos no estructurados, como la información de los usuarios, los materiales textuales, los foros, etc. Además, MongoDB es escalable horizontalmente, lo que significa que se puede agregar fácilmente más capacidad de procesamiento y almacenamiento a medida que aumenta la demanda.
* **API:**
  + **Google Cloud Platform (GCP):** Ofrece una variedad de APIs gratuitas para integrar funcionalidades como almacenamiento, traducción, reconocimiento de voz, etc.
  + **Firebase:** Es una plataforma de desarrollo para aplicaciones web y móviles que ofrece una amplia gama de servicios, incluyendo bases de datos (Firestore), almacenamiento de archivos (Storage), autenticación, etc.
  + **AWS (Amazon Web Services):** Ofrece una gama de APIs gratuitas

para diferentes servicios, incluyendo almacenamiento, reconocimiento de voz, traducción y más.

* + **Amazon S3 (Simple Storage Service):** Ofrece almacenamiento de objetos escalable y duradero para datos como archivos, imágenes y videos.
  + **AWS Cognito:** Brinda servicios de autenticación y autorización.
* **Consideraciones Para la Nube:**
  + **Escalabilidad:** Se debe considerar una base de datos y APIs que sean escalables en la nube, como MongoDB y los servicios de GCP o AWS mencionados anteriormente.
  + **Seguridad:** Se debe implementar una estrategia de seguridad robusta en la nube, incluyendo control de acceso, encriptación de datos y medidas contra ataques.
  + **Coste:** Es importante elegir servicios en la nube con un modelo de precios que se ajuste al presupuesto del proyecto.
* **Requisitos Mínimos del Equipo:**
  + **Procesador:** Core i3 de novena generación o superior.
  + **RAM:** Al menos 4 GB de RAM.
  + **Espacio en disco:** 1 GB o más de espacio libre. - **Sistema operativo:** Windows 10 o superior.
* **Adaptación a la Plataforma de Aprendizaje para Todos:**
  + - **Autenticación:** La autenticación de usuarios será fundamental para proteger la información sensible, controlar el acceso a recursos y gestionar las diferentes funcionalidades del sistema.
    - **Base de Datos:** MongoDB es una opción adecuada para gestionar

información de usuarios, materiales textuales, foros, actividades y otras funciones de la plataforma.

* + - **APIs:** Se recomienda utilizar las APIs de GCP o AWS para integrar funcionalidades como almacenamiento de archivos, reconocimiento de voz (para subtítulos en videos), traducción (para ofrecer contenido en varios idiomas) y otros servicios que mejoren la experiencia del usuario.
    - **Requisitos del Equipo:** Se deben considerar los requisitos mínimos del equipo para asegurar que la plataforma funcione correctamente y que los usuarios tengan una buena experiencia.

# Caso de Uso

## Descripción del Negocio

La lógica de negocio de la Plataforma de Aprendizaje para Todos define cómo interactúan los usuarios con el sistema, y cómo se gestionan los materiales textuales para lograr los objetivos de aprendizaje.

### Gestión de Usuarios

* **Registro:** Los usuarios deben registrarse en la plataforma proporcionando información personal.
* **Autenticación:** Los usuarios deben autenticarse con un nombre de usuario y contraseña para acceder al sistema.
* **Perfiles:** Cada usuario tiene un perfil con información específica
* **Usuario:** Nombre, correo electrónico, contraseña, historial de cursos, calificaciones.
* **Administradores:** Nombre, correo electrónico, contraseña, acceso a todas las funcionalidades de la plataforma, control de usuarios y contenido.
* **Administración de Usuarios:** El administrador puede crear, modificar y eliminar usuarios.

### Gestión de materiales textuales

* **Publicación de Recursos:** Los usuarios pueden publicar materiales textuales como libros, artículos o textos parecidos. La plataforma debe validar la información básica del recurso (título, autor, descripción, etc.) y el tipo de contenido.
* **Aprobación de Recursos:** El administrador revisa y aprueba los materiales textuales publicados por los usuarios antes de que sean visibles para todos.
* **Categorización de Recursos:** Los materiales textuales se clasifican en materias (por ejemplo, literatura, historia, matemáticas) para facilitar la búsqueda.
* **Acceso a Recursos:** Los usuarios pueden acceder a los materiales textuales aprobados por el administrador.

### Gestión de Foros

* **Creación de foros:** Los usuarios pueden crear foros específicos para debatir sobre los libros o artículos que encuentren.
* **Publicación de Preguntas:** Los usuarios pueden publicar preguntas en los foros.
* **Respuestas:** Otros usuarios pueden responder a las preguntas en los foros.
* **Moderación:** El administrador puede moderar los foros para garantizar un entorno adecuado.

### Flujo de Trabajo

* **Usuario:** Se registra, busca materiales textuales, crea foros, participa en foros, publica material textual.
* **Administrador:** Gestiona usuarios, aprueba materiales textuales, modera los foros, etc.

### Integración con APIs

* **Integración con Otras Plataformas:** La plataforma puede integrar APIs para conectar con otras plataformas educativas, como bibliotecas digitales o plataformas de aprendizaje en línea, para ampliar la oferta de materiales textuales.
* **Servicios de la Nube:** Se pueden integrar APIs para utilizar servicios de la nube como almacenamiento (Amazon S3, Google Cloud Storage), reconocimiento de voz (Amazon Transcribe, Google Cloud Speech-to-Text), traducción (Google Translate, Microsoft Translator) y otros servicios que mejoren la experiencia del usuario.

## Identificación de los Procesos de Negocio

Se identifican tres procesos principales del negocio que rigen el funcionamiento de la Plataforma de Aprendizaje para Todos.

### Gestión de Contenidos

* **Objetivo:** Controlar la creación, aprobación, organización y disponibilidad de los materiales textuales en la plataforma.
* **Flujo:**

○ Los usuarios publican material textual.

○ El administrador revisa y aprueba los materiales textuales.

○ Los materiales textuales se categorizan para facilitar su búsqueda.

○ Los usuarios acceden a los recursos aprobados.

○ Los usuarios pueden gestionar el acceso a sus recursos.

### Gestión del Foro

* **Objetivo:** Facilitar la interacción entre usuarios a través de foros de preguntas y respuestas.
* **Flujo:**

○ Los usuarios crean foros específicos de cada material textual que se requiera.

○ Otros usuarios publican preguntas o comentarios en los foros.

**○** Otros usuarios pueden responder a las preguntas.

**○** El administrador modera los foros para mantener un ambiente apropiado.

**Importancia de estos procesos:**

* Estos dos procesos del negocio son fundamentales para el correcto funcionamiento de la Plataforma de Aprendizaje para Todos, asegurando la calidad del contenido, la interacción entre usuarios y la administración eficiente del sistema.

## Identificar Actores

### Usuarios

* **Objetivo:** Acceder a material textual, compartir material textual, participar en foros de aprendizaje y mejorar su conocimiento en diferentes áreas.
* **Funciones:**

**○ Registrarse:** Crean una cuenta para acceder a la plataforma.

**○ Buscar y acceder a materiales compartidos:** Buscan libros y otros materiales textuales categorizados por materia.

**○ Participar en foros:** Realizan preguntas, responden a otras preguntas y participan en debates.

**○ Gestionar su perfil:** Modifican sus datos personales, consultan su historial de textos.

### Administradores

* **Objetivo:** Gestionar la plataforma en su totalidad, asegurando la calidad del contenido, la seguridad del sistema y el funcionamiento general.
* **Funciones:**

**○ Aprobar materiales textuales:** Revisan y aprueban los materiales textuales publicados por los usuarios antes de que estén disponibles para todos.

**○ Moderar los foros:** Aseguran que los foros sean un espacio adecuado para el debate, eliminando contenido inapropiado o spam.

**○ Controlar el sistema:** Supervisan el funcionamiento general de la plataforma, resuelven problemas técnicos, implementan nuevas funcionalidades y actualizan el sistema.

## Descripción de los casos de uso relevantes para la arquitectura

|  |  |
| --- | --- |
| **ID** | CUS - 1 |
| **Caso de Uso** | Registrar Material Textual |
| **Actor** | Usuario |
| **Descripción** | Permitir al usuario publicar nuevo material textual en la plataforma. |
| **Precondición** | El usuario debe estar autenticado en la plataforma. |
| **Flujo Principal** | 1. El usuario selecciona la opción "Publicar Material". 2. Ingresa la información del material (título, autor, descripción, enlace, tipo de contenido, etc.).   1. Selecciona la categoría(materia) correspondiente. 2. Envía la información para su revisión. |
| **Post-condición** | El material se envía a la cola de revisión del administrador. |

|  |  |
| --- | --- |
| **Flujos Alternos** | 1. Si el usuario no completa la información obligatoria, se le muestra un mensaje de error. 2. Si el material ya existe en la plataforma, se le muestra un mensaje de advertencia. |
| **ID** | CUS - 2 |
| **Caso de Uso** | Registrar Material Recibido |
| **Actor** | Administrador |
| **Descripción** | Permitir al administrador registrar nuevos materiales que se han recibido para la biblioteca de la plataforma. | |
| **Precondición** | El administrador debe estar autenticado en la plataforma. |
| **Flujo Principal** | 1. El administrador selecciona la opción "Registrar Libros Recibidos". 2. Ingresa la información del libro (título, autor, editorial.). 3. Selecciona la categoría(materia) correspondiente. 4. Confirma la información del libro. |
| **Post-condición** | El libro se agrega a la base de datos de la biblioteca. |
| **Flujos Alternos** | 1. Si el administrador no completa la información obligatoria, se le muestra un mensaje de error. |
| **ID** | CUS - 3 |
| **Caso de Uso** | Consultar Material |
| **Actor** | Usuario |
| **Descripción** | Permitir al usuario buscar y consultar la información de los libros disponibles en la plataforma. |
| **Precondición** | El usuario debe estar autenticado en la plataforma. |
| **Flujo Principal** | 1. El usuario selecciona la opción "Buscar Libros". 2. Ingresa los criterios de búsqueda (título o categoría). 3. Se muestra una lista de libros que coinciden con los criterios de búsqueda.   4. El usuario selecciona un libro para visualizar su |

|  |  |
| --- | --- |
|  | información detallada. |
| **Post-condición** | El usuario puede visualizar la información del libro seleccionado. |
| **Flujos Alternos** | 1. Si no se encuentran libros que coincidan con los criterios de búsqueda, se muestra un mensaje informando al usuario. |
| **ID** | CUS - 4 |
| **Caso de Uso** | Publicar Contenido |
| **Actor** | Usuario |
| **Descripción** | Permitir al usuario publicar contenido (libros, artículos, noticias, etc.) en la plataforma, sometiéndolo a la aprobación del administrador. |
| **Precondición** | El usuario debe estar autenticado en la plataforma. |
| **Flujo Principal** | 1. El usuario selecciona la opción "Publicar Contenido". 2. Ingresa el contenido, incluyendo título, autor, descripción y otros campos relevantes. 3. Selecciona la categoría correspondiente. 4. Envía el contenido para su revisión. |
| **Post-condición** | El contenido se envía a la cola de revisión del administrador. |
| **Flujos Alternos** | 1. Si el usuario no completa la información obligatoria, se le muestra un mensaje de error. 2. Si el contenido ya existe en la plataforma, se le muestra un mensaje de advertencia. |
| **ID** | CUS - 5 |
| **Caso de Uso** | Registrar Preguntas y Temas de Discusión |
| **Actor** | Usuario |
| **Descripción** | Permitir al usuario crear un nuevo tema de discusión o publicar una pregunta en un foro específico. |

|  |  |
| --- | --- |
| **Precondición** | El usuario debe estar autenticado en la plataforma y debe haber seleccionado un foro. |
| **Flujo Principal** | 1. El usuario selecciona la opción "Nueva pregunta/tema" dentro del foro. 2. Redacta la pregunta o el tema de discusión. 3. Pública la pregunta o el tema. |
| **Post-condición** | La pregunta o el tema se publican en el foro seleccionado. |
| **Flujos Alternos** | 1. Si el usuario no completa la información obligatoria, se le muestra un mensaje de error. |
| **ID** | CUS - 6 |
| **Caso de Uso** | Registrar Respuestas y Comentarios |
| **Actor** | Usuario |
| **Descripción** | Permitir al usuario registrar respuestas a las preguntas o comentarios a los temas de discusión en un foro. |
| **Precondición** | El usuario debe estar autenticado en la plataforma y debe haber seleccionado un foro. |
| **Flujo Principal** | 1. El usuario selecciona la opción "Responder/Comentar" dentro del foro. 2. Redacta la respuesta o el comentario. 3. Pública la respuesta o el comentario. |
| **Post-condición** | La respuesta o el comentario se publican en el foro seleccionado. |
| **Flujos Alternos** | 1. Si el usuario no completa la información obligatoria, se le muestra un mensaje de error. |
| **ID** | CUS - 7 |
| **Caso de Uso** | Mantener Información del Foro |
| **Actor** | Administrador |

|  |  |
| --- | --- |
| **Descripción** | Permitir al administrador actualizar, eliminar o bloquear temas de discusión, respuestas o comentarios dentro de los foros de la plataforma. |
| **Precondición** | El administrador debe estar autenticado en la plataforma. |
| **Flujo Principal** | 1. El administrador selecciona la opción "Gestionar Foros". 2. Selecciona el foro que desea gestionar. 3. Selecciona la acción a realizar (aprobar o eliminar) sobre el tema, respuesta o comentario. 4. Confirma la acción. |
| **Post-condición** | Se realiza la acción seleccionada sobre el tema, respuesta o comentario en el foro. |
| **Flujos Alternos** | 1. Si el administrador no tiene los permisos necesarios para realizar la acción, se le muestra un mensaje de error. |
| **ID** | CUS - 8 |
| **Caso de Uso** | Registrar Usuario |
| **Actor** | Usuario |
| **Descripción** | Permitir al usuario registrarse en la plataforma, creando una nueva cuenta de usuario. |
| **Precondición** | Ninguna. |
| **Flujo Principal** | 1. El usuario selecciona la opción "Registrarse". 2. Ingresa los datos personales (nombre completo, correo electrónico, contraseña) 4. Confirma los datos. |
| **Post-condición** | Se crea la cuenta de usuario. |
| **Flujos Alternos** | 1. Si el correo electrónico ya está registrado, se le muestra un mensaje de error. 2. Si el usuario no completa la información obligatoria, se le muestra un mensaje de error. |
| **ID** | CUS - 9 |
| **Caso de Uso** | Actualización de Perfil |
| **Actor** | Usuario |
| **Descripción** | Permitir al usuario actualizar sus datos personales y preferencias en la plataforma. |
| **Precondición** | El usuario debe estar autenticado en la plataforma. |
| **Flujo Principal** | 1. El usuario selecciona la opción "Mi Perfil". 2. Realiza los cambios en su información (nombre, correo electrónico, contraseña, descripción). 3. Guarda los cambios. |
| **Post-condición** | La información del perfil se actualiza. |
| **Flujos Alternos** | 1. Si el usuario no completa la información obligatoria, se le muestra un mensaje de error. |

## Sección de restricciones

**4.5.1. Tecnología**

|  |  |  |
| --- | --- | --- |
| **Capa** | **Tecnología** | **Descripción** |
| Front-end | React  [**https://devdocs.io/re act/**](https://devdocs.io/react/) | Framework JavaScript para desarrollo de interfaces de usuario web. Ofrece un enfoque declarativo y componentes reutilizables, lo que facilita la creación de aplicaciones web interactivas. |
| Front-end | Visual Studio Code [**https://code.visualst udio.com/docs**](https://code.visualstudio.com/docs) | IDE (Entorno de  Desarrollo Integrado) ligero y popular para desarrollar código en varios lenguajes, incluyendo JavaScript. Ofrece funciones de autocompletado, depuración, integración con Git y extensiones que facilitan el desarrollo. |

|  |  |  |
| --- | --- | --- |
| **Capa** | **Tecnología** | **Descripción** |
| Front-end | Figma | Herramienta de diseño colaborativa que se utiliza para crear interfaces de usuario (UI) y experiencia de usuario (UX). Permite a los diseñadores crear prototipos interactivos, maquetas y diseños visuales para aplicaciones web y móviles. |
| Front-end | Whimsikal | Herramienta de colaboración visual que se utiliza para crear diagramas, wireframes, mapas mentales y flujos de trabajo. Ayuda a los equipos a visualizar ideas y procesos de manera rápida y sencilla. |
| Back-end | Node.js  [https://nodejs.org/doc s/latest/api/](https://nodejs.org/docs/latest/api/) | Entorno de ejecución de JavaScript de lado del servidor. Permite crear aplicaciones web robustas y escalables utilizando JavaScript. |
| Back-end | Express.js  <https://expressjs.com/> | Framework web para Node.js que facilita la creación de aplicaciones web con rutas, middleware y gestión de peticiones HTTP. Se adapta perfectamente a la naturaleza de la plataforma. |
| Base de Datos | MongoDB  [https://www.mongod b.com/docs/](https://www.mongodb.com/docs/) | Base de datos NoSQL  que utiliza  documentos JSON para almacenar la información. Es ideal |

|  |  |  |
| --- | --- | --- |
| **Capa** | **Tecnología** | **Descripción** |
|  |  | para gestionar grandes volúmenes de datos no estructurados y se adapta a la naturaleza dinámica de la plataforma. |
| Nube | Google Cloud  Platform (GCP)  [https://cloud.google.c](https://cloud.google.com/docs)  [om/docs](https://cloud.google.com/docs) | Plataforma de servicios en la nube que ofrece una amplia gama de servicios como almacenamiento (Cloud Storage), bases de datos (Cloud SQL,  Cloud Firestore), computación (Compute Engine, App Engine), etc. Es ideal para desplegar y escalar la plataforma. |
| APIs | Firebase  [https://firebase.googl e.com/docs](https://firebase.google.com/docs) | Plataforma de desarrollo móvil y web que ofrece una amplia gama de  servicios, incluyendo una base de datos NoSQL (Firestore), almacenamiento de archivos (Storage), autenticación, etc. Se integra bien con React. |
| APIs | Google Cloud  Platform (GCP) | Ofrece APIs para  integrar diferentes servicios como reconocimiento de voz, traducción, análisis de texto, etc. Se puede usar para mejorar la accesibilidad y la funcionalidad de la plataforma. |
| Gestión | GitHub  [GitHub](https://docs.github.com/es) [Docs](https://docs.github.com/es) | Plataforma para alojar y gestionar proyectos |
| **Capa** | **Tecnología** | **Descripción** |
|  |  | de software utilizando control de versiones con Git. Sirve para colaborar en el desarrollo de código, gestionar versiones y almacenar repositorios de forma segura. |
| Gestión | Jira [https://www.atlassian. com/es/software/jira/ guides/getting-started](https://www.atlassian.com/es/software/jira/guides/getting-started/introduction#dig-into-specific-features)  [/introduction#dig-into](https://www.atlassian.com/es/software/jira/guides/getting-started/introduction#dig-into-specific-features)  [-specific-features](https://www.atlassian.com/es/software/jira/guides/getting-started/introduction#dig-into-specific-features) | Herramienta utilizada para la gestión de proyectos y seguimiento de tareas, especialmente en equipos que usan metodologías ágiles como Scrum o Kanban. |

* **Recomendación para el Back-end:**
  1. **Express.js:** Es una excelente opción para el back-end de la plataforma. Es un framework ligero y flexible que se integra perfectamente con Node.js. Su facilidad de uso, el gran ecosistema de módulos y su capacidad de escalar hacen que sea una elección ideal para un proyecto web como este.
* **Justificación:**
  1. **React:** React es la tecnología de elección para el desarrollo del front-end, ya que permite crear interfaces de usuario interactivas, componentes reutilizables y mantiene un alto rendimiento.

**○ Visual Studio Code:** Visual Studio Code es un IDE popular y eficiente que ofrece herramientas de desarrollo útiles para React y JavaScript.

**○ MongoDB:** La naturaleza dinámica de la plataforma y la necesidad de gestionar datos no estructurados hacen que MongoDB sea la mejor opción para la base de datos.

**○ Express.js:** Su sencillez, flexibilidad y el gran ecosistema de módulos lo convierten en una excelente opción para construir el back-end de una plataforma web moderna.

**○ GCP:** GCP ofrece una plataforma completa de servicios en la nube, incluyendo bases de datos, almacenamiento, computación y APIs. Esto facilita el despliegue y la escalabilidad de la plataforma.

**○ APIs de Google:** Firebase y GCP ofrecen una variedad de APIs gratuitas que pueden mejorar la experiencia del usuario, como almacenamiento de archivos, reconocimiento de voz, traducción y análisis de texto.

## Sección de QoS

### Usabilidad

* La plataforma debe ser fácil de usar e intuitiva para todos los usuarios, sin importar su experiencia con plataformas digitales.
* La navegación debe ser simple y clara, con un diseño que facilite la búsqueda de información y la participación en los foros.

### Eficiencia

* La plataforma debe ser rápida y eficiente en su funcionamiento, con tiempos de carga rápidos para los materiales y una respuesta rápida en las interacciones con el sistema.
* Se deben optimizar las consultas a la base de datos para asegurar un rendimiento óptimo.

### Confiabilidad

* La plataforma debe ser estable y segura, garantizando la disponibilidad de la información y la protección de los datos de los usuarios.
* Se debe implementar una estrategia de respaldo y recuperación de datos para minimizar el impacto de posibles fallos.

### Mantenimiento

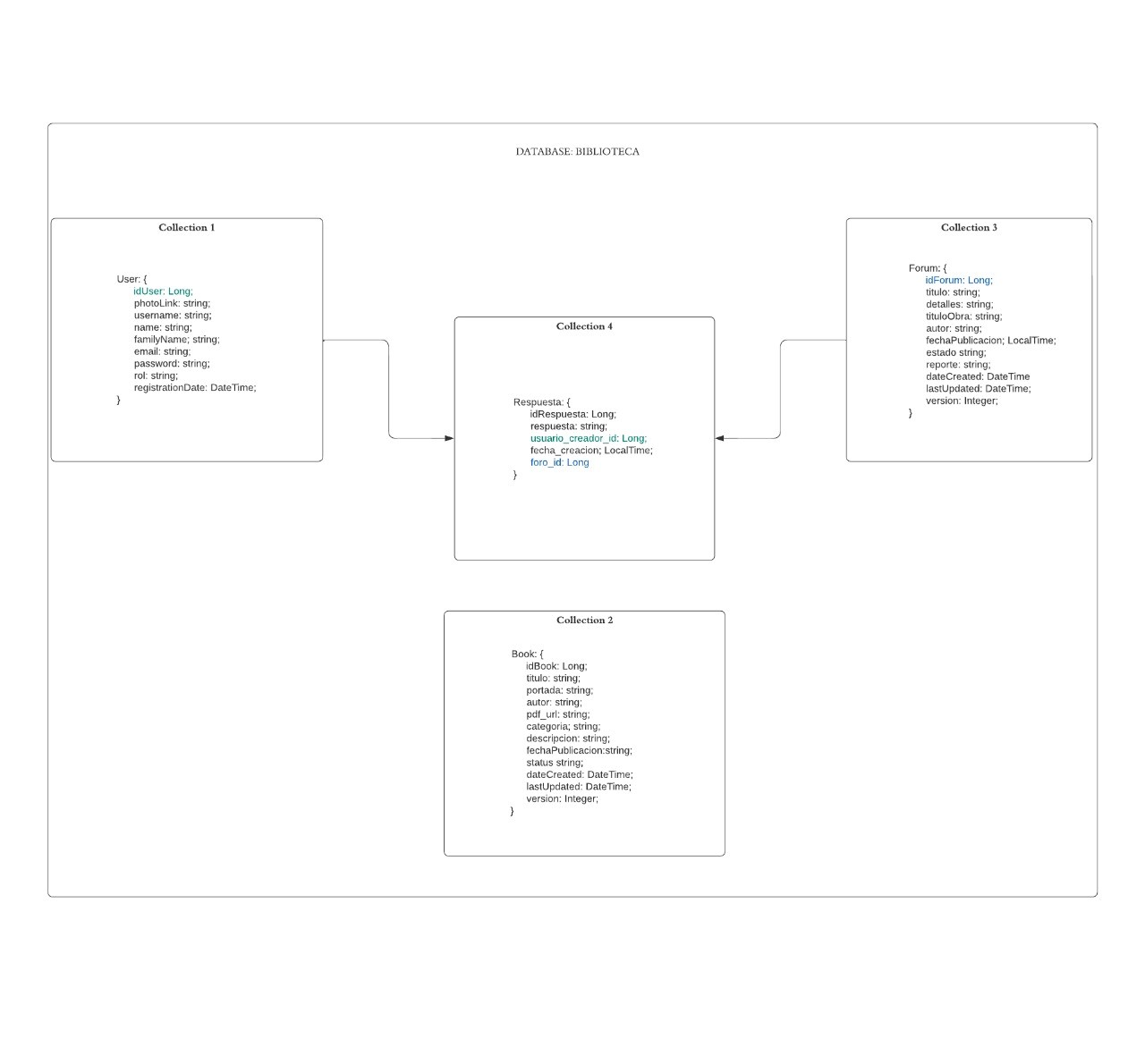
* La plataforma debe ser fácil de mantener y actualizar, permitiendo la adición de nuevas funcionalidades, la corrección de errores y la adaptación a las necesidades cambiantes.
* Se debe utilizar un sistema de versionamiento para el código fuente para facilitar la gestión de las actualizaciones.

### Estándares

* La plataforma debe cumplir con las normas de accesibilidad web (WCAG) para que sea accesible a todos los usuarios, independientemente de sus capacidades.
* Se deben seguir las mejores prácticas de desarrollo web para asegurar un código limpio, eficiente y de alta calidad.

**5. Vista lógica**

# - Diagrama de Base de Datos



# - Diagrama de Interfaces