****

**UNIVERSIDAD PRIVADA DE TACNA**

**FACULTAD DE INGENIERÍA**

**Escuela Profesional de Ingeniería de Sistemas**

***Proyecto: Sistema Web de Aprendizaje Autoguiado con IA para el Desarrollo de la Competencia “Gestiona proyectos de emprendimiento económico o social” del Área de Educación para el Trabajo en los Estudiantes del VI Ciclo de Educación Secundaria de la I.E. Marcelino Champagnat, Tacna***

Curso: *Construcción de Software I*

Docente: *Ing. ALBERTO JOHNATAN FLOR RODRIGUEZ*

Integrantes:

***Japura Quispe, herminia Aurelia (2018060912)***

***Concha Llaca, Gerardo Alejandro                      (2017057849)***

**Tacna – Perú**

***2025***

| CONTROL DE VERSIONES | | | | | |
| --- | --- | --- | --- | --- | --- |
| Versión | Hecha por | Revisada por | Aprobada por | Fecha | Motivo |
| 1.0 | MPV | ELV | ARV | 10/10/2020 | Versión Original |

Sistema *de Reportes de Problemas Viales*

Documento de Arquitectura de Software

Versión *1.0*

| CONTROL DE VERSIONES | | | | | |
| --- | --- | --- | --- | --- | --- |
| Versión | Hecha por | Revisada por | Aprobada por | Fecha | Motivo |
| 1.0 | MPV | ELV | ARV | 10/10/2020 | Versión Original |

INDICE GENERAL

1. Introducción [4](#_heading=h.3foe5nok2kzd)

1.1 Propósito [4](#_heading=h.8qizw7ae0u8v)

1.2 Alcance [4](#_heading=h.y2zg7wxazjz)

1.3 Definición, siglas y abreviaturas [4](#_heading=h.lmpu4gmdfcu)

1.4 Referencias [4](#_heading=h.auo3voxd82mv)

1.5 Visión General [5](#_heading=h.gg01o6cnio5j)

2. Representación Arquitectónica [5](#_heading=h.9ydf1f5x4lp1)

2.1 Escenarios [5](#_heading=h.tpnshwka5t8h)

2.2 Vista Lógica [6](#_heading=h.95zb4c37qvrz)

2.3 Vista del Proceso [6](#_heading=h.iw4c9e3jqo27)

2.4 Vista del desarrollo [6](#_heading=h.dcdr1wpa34nj)

2.5 Vista Física [6](#_heading=h.ud9sekdhlmx6)

3. Objetivos y limitaciones arquitectónicas [7](#_heading=h.59sz88kkb9tx)

3.1 Seguridad [7](#_heading=h.7gz5iglnxh2p)

3.2 Escalabilidad [7](#_heading=h.rg984mdyggxn)

3.3 Tiempos de Respuesta Rápidos [7](#_heading=h.iyc8ttewbgrn)

3.4 Optimización de Carga [7](#_heading=h.u62q3mh55ywf)

4. Análisis de Requerimientos [8](#_heading=h.bur6gxo9tvud)

4.1 Requerimientos funcionales [7](#_heading=h.7gz5iglnxh2p)

4.2 Requerimientos no funcionales [7](#_heading=h.rg984mdyggxn)

5. Vistas de Caso de Uso [9](#_heading=h.fgrxvwgu7d7u)

6. Vista Lógica [17](#_heading=h.26ec37huwfw4)

6.1 Diagrama Contextual [17](#_heading=h.8r5bm1pgheaj)

7. Vista de Procesos [18](#_heading=h.mextw1e5ftkc)

7.1 Diagrama de Proceso Actual 18

7.2 Diagrama de Proceso Propuesto [18](#_heading=h.6mfkzqfq85jc)

8. Vista de Despliegue [19](#_heading=h.57871c80zlr9)

8.1 Diagrama de Contenedor [19](#_heading=h.lrcf3qe1xjnc)

9. Vista de Implementación [20](#_heading=h.hh6379l8l2iy)

9.1 Diagrama de Componentes [20](#_heading=h.vcmtjsrtol8r)

10. Vista de Datos [22](#_heading=h.yegh4ofj2302)

10.1 Diagrama Base de Datos [22](#_heading=h.1r0575venk7e)

11. Calidad [24](#_heading=h.gema11mixjx4)

11.1 Escenario de Seguridad 25

11.2 Escenario de Escalabilidad 26

11.3 Escenario de Tiempos de Respuesta Rápidos 27

11.4 Optimización de Carga 28

1. Introducción

1.1 Propósito

El propósito de este documento es describir la arquitectura del **Sistema**, el cual permite a los estudiantes acceder a materiales académicos, interactuar con un asistente de IA y realizar evaluaciones; a los docentes publicar contenidos y gestionar calificaciones; y a los administradores administrar usuarios, cursos y permisos.

Este documento detalla las vistas arquitectónicas del sistema (lógica, de procesos, de desarrollo y física), con el objetivo de:

* Facilitar la comprensión técnica del sistema.
* Guiar el desarrollo y mantenimiento del código.
* Garantizar escalabilidad para futuras funcionalidades.
* Establecer estándares para integración con servicios externos (Telegram, Gemini AI).

1.2 Alcance

Este sistema web estará dirigido en su versión piloto a la Institución Educativa Marcelino Champagnat. Incluirá funcionalidades de asignación de contenido, evaluación, retroalimentación y seguimiento del avance académico del área de EPT. El sistema estará disponible para acceso remoto a través de cualquier dispositivo con conexión a Internet y será compatible con los principales navegadores web.

1.3 Definición, siglas y abreviaturas

| **Término** | **Definición** |
| --- | --- |
| **JWT** | JSON Web Token: Estándar para autenticación basada en tokens. |
| **MVC** | Modelo-Vista-Controlador: Patrón arquitectónico usado en el frontend. |
| **PO** | Product Owner: Rol encargado de priorizar requerimientos. |
| **SSO** | Single Sign-On: Autenticación unificada (ej: Google Auth - RF-02). |

1.4 Referencias

* **Documentos:**
  + [Requerimientos Funcionales (RF-01 a RF-14)].
  + [Manifiesto Ágil](http://agilemanifesto.org/).
* **Tecnologías:**
  + Documentación oficial de [ASP.NET Core](https://docs.microsoft.com/en-us/aspnet/core/).
  + API de [Telegram Bots](https://core.telegram.org/bots/api).

1.5 Visión General

El sistema sigue una arquitectura **multicapa** con:

* **Frontend:** Razor.
* **Backend:** API REST con ASP.NET Core y Entity Framework.
* **Base de Datos:** PostgreSQL (relacional) para datos estructurados.
* **Servicios Externos:**
  + Gemini AI para el chat educativo.
  + Telegram para notificaciones en tiempo real.

2. Representación Arquitectónica

2.1 Escenarios

*Tabla N°1: Escenario-Iniciar sesión*

| **Escenario de caso de uso: Iniciar Sesión** | |
| --- | --- |
| **Tipo** | **Obligatorio** |
| **Versión** | V 1.0 |
| **Autor(es)** | Concha Llaca, Gerardo Alejandro  Japura Quispe, Herminia Aurelia |
| **Actores** | Administrador, Docente  , Estudiante |
| **Descripción** | Permite a los usuarios acceder al sistema mediante credenciales de email y contraseña, asignando permisos según su rol (Administrador, Docente o Estudiante). |
| **Precondiciones** | El usuario debe estar registrado en el sistema.  El servicio de autenticación debe estar operativo. |
| **Narrativa de caso de uso** | |
| **Acción del Actor** | **Respuesta del Sistema** |
| 1. El usuario ingresa a la plataforma y selecciona "Iniciar sesión". | 2. El sistema muestra el formulario de login (email y contraseña). |
| 3. El usuario ingresa su email y contraseña. | 4. El sistema valida el formato del email |
|  | 5. El sistema verifica las credenciales en la base de datos |
|  | 6. El sistema identifica el rol asociado al usuario. |
|  | 7. El sistema redirige al dashboard correspondiente (Admin/Docente/Estudiante). |
| **Flujo de Excepciones - FE1** | |
| **Acción del Actor** | **Respuesta del Sistema** |
| 1. El usuario ingresa credenciales inválidas. | 2. El sistema muestra: "Usuario o contraseña incorrectos". |

*Fuente: Elaboración propia*

*Tabla N°2: Escenario de caso de uso-Autenticar con Google*

| **Escenario de caso de uso: Autenticar con Google** | |
| --- | --- |
| **Tipo** | **Obligatorio** |
| **Versión** | V 1.0 |
| **Autor(es)** | Concha Llaca, Gerardo Alejandro  Japura Quispe, Herminia Aurelia |
| **Actores** | Estudiante |
| **Descripción** | Permite a los usuarios autenticarse mediante SSO (Single Sign-On) de Google, asignando automáticamente el rol Estudiante si el correo es institucional |
| **Precondiciones** | El usuario debe tener una cuenta Google válida.  El servicio de Google OAuth debe estar activo. |
| **Narrativa de caso de uso** | |
| **Acción del Actor** | **Respuesta del Sistema** |
| 1. El usuario selecciona "Iniciar con Google". | 2. El sistema redirige a la página de autenticación de Google. |
| 3. El usuario ingresa sus credenciales de Google. | 4. Google valida las credenciales y devuelve el token de acceso al sistema. |
|  | 5. El sistema verifica el dominio del correo |
|  | 6. Si el dominio es válido, asigna rol Estudiante automáticamente. |
|  | 7. El sistema redirige al dashboard de Estudiante. |
| **Flujo de Excepciones – FE2** | |
| **Acción del Actor** | **Respuesta del Sistema** |
| 1. El usuario ingresa con un correo personal (ej: @gmail.com). | 2. El sistema bloquea el acceso y muestra: "Solo correos @instituto.edu permitidos". |

*Fuente: Elaboración propia*

*Tabla N°3: Escenario de caso de uso-Recuperar contraseña*

| **Escenario de caso de uso: Recuperar contraseña** | |
| --- | --- |
| **Tipo** | **Obligatorio** |
| **Versión** | V 1.0 |
| **Autor(es)** | Concha Llaca, Gerardo Alejandro  Japura Quispe, Herminia Aurelia |
| **Actores** | Actores: Usuario (Administrador/Docente/Estudiante) |
| **Descripción** | Permite a los usuarios restablecer su contraseña mediante un código alfanumérico. |
| **Precondiciones** | El usuario debe tener un correo valido |
| **Narrativa de caso de uso** | |
| **Acción del Actor** | **Respuesta del Sistema** |
| 1. Selecciona "¿Olvidó su contraseña?" | 2. Muestra formulario para ingresar correo. |
| 3. Ingresa su correo. | 4. Verifica que el correo exista en la BD. |
|  | 5. Genera código alfanumérico y lo envía por email. |
| 7. Ingresa el código recibido | 8. Válida coincidencia y vigencia del código. |
| 9. Ingresa nueva contraseña | 11. Actualiza contraseña en la BD |
|  | 12. Notifica: "Contraseña actualizada correctamente". |

*Fuente: Elaboración propia*

*Tabla N°4: Escenario de caso de uso-Gestionar usuarios*

| **Escenario de caso de uso: Gestionar usuarios** | |
| --- | --- |
| **Tipo** | **Obligatorio** |
| **Versión** | V 1.0 |
| **Autor(es)** | Concha Llaca, Gerardo Alejandro  Japura Quispe, Herminia Aurelia |
| **Actores** | Administrador |
| **Descripción** | Permite al Administrador realizar operaciones CRUD (Crear, Leer, Actualizar, Eliminar) sobre los usuarios del sistema, incluyendo la asignación de roles (Administrador/Docente/Estudiante). |
| **Precondiciones** | El administrador debe haber iniciado sesión.  El módulo de gestión de usuarios debe estar activo. |
| **Narrativa de caso de uso** | |
| **Acción del Actor** | **Respuesta del Sistema** |
| 1. El usuario ingresa a la plataforma y selecciona "Iniciar sesión". | 2. El sistema muestra el formulario de login (email y contraseña). |
| 3. El usuario ingresa su email y contraseña. | 4. El sistema verifica las credenciales en la base de datos |
|  | 5. El sistema identifica el rol asociado al usuario. |
|  | 6. El sistema redirige al dashboard de Administrador |
| 7. Accede a la pestaña "Usuarios". | 8. Muestra tabla con: Nombre, Apellido, Email, Rol y botones "Editar/Eliminar" |
| 9. Haz clic en "+" o "Crear Usuario". | 10. Abre formulario con campos: Nombre, Apellido, Email, Contraseña, Rol. |
| 11. Completa el formulario. | 12. Valida Datos. |
| 13. Selecciona "Guardar". | 14. El sistema guarda el usuario creado y lo muestra en la lista |
| **Flujo Alternativo - FA1: Edición desde tabla** | |
| **Acción del Actor** | **Respuesta del Sistema** |
| 1. El administrador hace clic en "Editar" de un usuario. | 2. Abre el mismo formulario con datos precargados. |
| 3. Modifica el rol del usuario | 4. Valida que el Admin tenga permisos para cambiar roles. |
|  | 5. Actualiza la fila en la tabla. |
| **Flujo Alternativo - FA2: Eliminacion desde tabla** | |
| **Acción del Actor** | **Respuesta del Sistema** |
| 1. El administrador hace clic en "Eliminar" de un usuario. | 2. Verifica que el administrador tenga permisos para eliminar. |
|  | 3. Busca el usuario a eliminar en la base de datos. |
|  | 4. El sistema elimina el usuario y actualiza la lista. |

*Fuente: Elaboración propia*

*Tabla N°5: Escenario de caso de uso-Registrar estudiantes*

| **Escenario de caso de uso: Registrar estudiantes** | |
| --- | --- |
| **Tipo** | **Obligatorio** |
| **Versión** | V 1.0 |
| **Autor(es)** | Concha Llaca, Gerardo Alejandro  Japura Quispe, Herminia Aurelia |
| **Actores** | Docente |
| **Descripción** | Permite al Docente realizar operaciones CRUD (Crear, Leer, Actualizar, Eliminar) sobre los Estudiantes. |
| **Precondiciones** | El Docente debe haber iniciado sesión.  El módulo de gestión de usuarios debe estar activo. |
| **Narrativa de caso de uso** | |
| **Acción del Actor** | **Respuesta del Sistema** |
| 1. Accede a la pestaña "Usuarios". | 2. Muestra tabla con filtro automático "Estudiantes" y botón "+" habilitado. |
| 3. Haz clic en "+" o "Crear Usuario". | 4. Abre formulario con campos: Nombre, Apellido, Email, Contraseña, Rol (Solo estudiante). |
| 5. Completa el formulario. | 6. Valida Datos. |
| 7. Selecciona "Guardar". | 8. El sistema guarda el usuario creado y lo muestra en la lista |
| **Flujo Alternativo - FA1: Edición desde tabla** | |
| **Acción del Actor** | **Respuesta del Sistema** |
| 1. Haz clic en "Editar" de un usuario. | 2. Abre el mismo formulario con datos precargados. |
| 3. Modifica el rol de Estudiante a Docente. | 4. Valida que el Admin tenga permisos para cambiar roles. |
|  | 5. Actualiza la fila en la tabla. |
| **Flujo Alternativo - FA2: Registro masivo** | |
| **Acción del Actor** | **Respuesta del Sistema** |
| 1. Selecciona "Importar CSV**".** | 2. Muestra modal para subir archivo con columnas |
|  | 3. Valida todos los emails |
|  | 4. Asigna rol Estudiante automático a todos los registros válidos. |

*Fuente: Elaboración propia*

*Tabla N°6: Escenario de caso de uso-Publicar materiales*

| **Escenario de caso de uso: Publicar materiales** | |
| --- | --- |
| **Tipo** | **Obligatorio** |
| **Versión** | V 1.0 |
| **Autor(es)** | Concha Llaca, Gerardo Alejandro  Japura Quispe, Herminia Aurelia |
| **Actores** | Docente |
| **Descripción** | Permite a los docentes publicar recursos educativos en los cursos asignados por el administrador. |
| **Precondiciones** | El docente debe tener cursos asignados |
| **Narrativa de caso de uso** | |
| **Acción del Actor** | **Respuesta del Sistema** |
| 1. El docente inicia sesión. | 2. El sistema valida las credenciales y muestra el panel docente. |
| 3. El docente accede a "Materiales" y selecciona "Agregar Material". | 4. El sistema muestra el formulario para crear un nuevo material, con la lista de cursos asignados al docente. |
| 5. El docente completa los campos del formulario (título, descripción, URL, curso, etc.). | 6. El sistema valida los datos ingresados (campos obligatorios, formato correcto). |
| 7. El docente hace clic en "Guardar". | 8. El sistema guarda el material en la base de datos y muestra confirmación. Envia notificación vía Telegram. |
|  | 9. El sistema redirige a la lista de materiales del docente. |
| **Flujo Alternativo - FA1: Edición de material** | |
| **Acción del Actor** | **Respuesta del Sistema** |
| 1. En la lista de materiales, el docente selecciona "Editar" en un material existente. | 2. El sistema abre el formulario con los datos del material precargados. |
| 3. El docente modifica la descripción o la URL, pero mantiene el curso asignado. | 4. El sistema valida los datos actualizados. |
| 5. El docente hace clic en "Guardar". | 6. El sistema actualiza el material y muestra confirmación. |
|  | 7. El sistema redirige a la lista de materiales |
| **Flujo de Excepción (FE1) - Error en validación de datos** | |
| 1. El docente ingresa datos inválidos o deja campos obligatorios vacíos. | 2. El sistema muestra mensajes de error en los campos inválidos y no guarda los datos. |
| 3. El docente corrige los errores. | 4. El sistema vuelve a validar y, si es correcto, continúa con el flujo principal o alternativo. |

*Fuente: Elaboración propia*

*Tabla N°7: Escenario de caso de uso-Administrar Cursos*

| **Escenario de caso de uso: Administrar Cursos** | |
| --- | --- |
| Tipo | Obligatorio |
| Versión | V 1.0 |
| Autor(es) | Concha Llaca, Gerardo Alejandro  Japura Quispe, Herminia Aurelia |
| Actores | Administrador (Principal)  Docente (Secundario, para edición limitada) |
| Descripción | Permite a los administradores crear, editar, eliminar y asignar docentes/estudiantes a cursos. Los docentes solo pueden editar cursos asignados a ellos. |
| Precondiciones | El administrador debe estar registrado e iniciar sesión en la plataforma.  Debe tener permisos de rol "Administrador".  Los docentes y estudiantes deben estar previamente registrados en el sistema. |
| Narrativa de caso de uso | |
| Acción del Actor | Respuesta del Sistema |
| 1. El administrador inicia sesión. | 2. El sistema valida las credenciales y muestra el panel de administración. |
| 3. Selecciona "Gestionar Cursos". | 4. El sistema muestra la lista de cursos existentes con opciones (Crear/Editar/Eliminar). |
| 5. Elige "Crear Curso". | 6. Muestra un formulario con campos: Nombre, Descripción y lista de docentes. |
| 7.Completa los datos y asigna un docente. | 8. Valida los campos y guarda el curso en la base de datos. |
| **Flujo Alternativo – FA1: Edición por docente** | |
| Acción del Actor | Respuesta del Sistema |
| 1. Docente ingresa a "Gestionar Cursos". | 2. El sistema muestra solo cursos asignados a este docente, sin opciones para editar cursos de otros docentes. |
| 3. Selecciona "Editar" en curso asignado. | 4. El sistema muestra el formulario para modificar nombre y descripción. |
| 5. el docente edita los campos y da en “guardar” | 6. el sistema guarda los cambios y actualiza la informacion. |

*Fuente: Elaboración propia*

*Tabla N°8: Escenario de caso de uso-Gestionar Evaluaciones*

| **Escenario de caso de uso: Gestionar Evaluaciones** | |
| --- | --- |
| **Tipo** | **Obligatorio** |
| **Versión** | V 1.0 |
| **Autor(es)** | Concha Llaca, Gerardo Alejandro  Japura Quispe, Herminia Aurelia |
| **Actores** | Docente |
| **Descripción** | Permite al docente gestionar evaluaciones dentro de los cursos asignados, incluyendo crear, editar, eliminar y visualizar evaluaciones. Además, puede ver el detalle de las respuestas de los estudiantes después de rendirlas. |
| Precondiciones | El docente debe estar registrado, iniciar sesión y tener cursos asignados.  El curso destino debe existir. |
| **Narrativa de caso de uso** | |
| **Acción del Actor** | **Respuesta del Sistema** |
| 1. El docente inicia sesión. | 2. El sistema valida las credenciales y muestra el panel docente. |
| 3. Selecciona "Evaluaciones" en un curso. | 4. El sistema muestra una tabla con las evaluaciones registradas para los cursos del docente. |
| 5. El docente hace clic en el botón “Crear Evaluación”. | 6. El sistema valida que cada pregunta tenga al menos una respuesta correcta y guarda la evaluación. Envía notificacion por Telegram |
| **Flujos Alternativos FA1: Eliminar Evaluación** | |
| 1. En la tabla, el docente hace clic en el botón “Eliminar” | 2. El sistema solicita confirmación. Si se acepta, elimina la evaluación y sus preguntas asociadas. |
| **Flujos Alternativos FA2: Ver Detalle de Evaluación** | |
| 1. En la tabla de evaluaciones, el docente hace clic en el botón “Ver Detalle”. | 2. El sistema muestra todas las preguntas y respuestas del examen, con opción para ver el rendimiento de los estudiantes. |
| **Flujos Alternativos FA3: Eliminar Evaluación** | |
| 1. En la tabla, el docente hace clic en el botón “Eliminar”. | 2. El sistema solicita confirmación. Si se acepta, elimina la evaluación y sus preguntas asociadas. |
| **Flujo Alternativo FA4: Ver Respuestas de los Estudiantes** | |
| 1. Desde el detalle de una evaluación rendida, selecciona “Ver respuestas de estudiantes”. | 2. El sistema muestra por cada estudiante las respuestas marcadas, indicando si fueron correctas o no, junto con la calificación. |

*Fuente: Elaboración propia*

*Tabla N°9: Escenario de caso de uso-Acceder a materiales.*

| Escenario de caso de uso: Acceder a materiales | |
| --- | --- |
| Tipo | Obligatorio |
| Versión | V 1.0 |
| Autor(es) | Concha Llaca, Gerardo Alejandro  Japura Quispe, Herminia Aurelia |
| Actores | Estudiante |
| Descripción | Permite a los estudiantes acceder a materiales educativos (PDF, DOC, etc.) publicados por docentes en cursos donde están inscritos. Los docentes y administradores pueden visualizar todos los materiales, pero solo los docentes asignados pueden editarlos. |
| Precondiciones | El estudiante debe estar registrado, iniciar sesión y estar inscrito en al menos un curso.  El curso debe tener materiales publicados. |
| Narrativa de caso de uso | |
| Acción del Actor | Respuesta del Sistema |
| 1. El estudiante inicia sesión. | 2. El sistema valida las credenciales y muestra el panel de estudiante. |
| 3. Selecciona un curso inscrito. | 4. Muestra la lista de materiales disponibles (ordenados por fecha de publicación). |
| 5. Elige un material (ej: "Guía de estudio"). | 6. El sistema permite visualizarlo en línea (según formato). |
| 7. El administrador o docente guarda el quiz. | 8. El sistema almacena la información y la asigna al curso correspondiente. |
| 9. Un estudiante completa el quiz. | 10. El sistema registra las respuestas y, si es automático, muestra la calificación inmediata. |

*Fuente: Elaboración propia*

*Tabla N°10: Escenario de caso de uso-Interactuar con Chat IA .*

| **Escenario de caso de uso: Interactuar con Chat IA** | |
| --- | --- |
| **Tipo** | **Obligatorio** |
| **Versión** | V 1.0 |
| **Autor(es)** | Concha Llaca, Gerardo Alejandro  Japura Quispe, Herminia Aurelia |
| **Actores** | Estudiante |
| **Descripción** | Permite a los estudiantes realizar consultas académicas a través de un Chat IA, que responde basándose en los materiales y contexto de los cursos en los que están inscritos. |
| **Precondiciones** | El estudiante debe estar registrado, iniciar sesión y estar inscrito en al menos un curso.  El curso debe tener materiales publicados (para contexto). |
| **Narrativa de caso de uso** | |
| **Acción del Actor** | **Respuesta del Sistema** |
| 1. El estudiante inicia sesión. | 2. El sistema valida credenciales y muestra la lista de cursos inscritos. |
| 3. Selecciona "Chat IA" en un curso. | 4. Muestra la interfaz de chat con historial vacío (si es primera vez). |
| 5. Escribe una pregunta | 6. El sistema:  a. Recupera el contexto del curso (nombre, descripción, materiales).  b. Envía la pregunta y contexto a la API de Gemini IA. |
|  | 7. Muestra la respuesta generada por la IA. |
|  | 8. Formatea la respuesta en un chat amigable (markdown soportado) |

*Fuente: Elaboración propia*

*Tabla N°11: Escenario de caso de uso-Notificar por Telegram*

| **Escenario de caso de uso: Notificar por Telegram** | |
| --- | --- |
| **Tipo** | **Obligatorio** |
| **Versión** | V 1.0 |
| **Autor(es)** | Concha Llaca, Gerardo Alejandro  Japura Quispe, Herminia Aurelia |
| **Actores** | Docente / Estudiante / Administrador |
| **Descripción** | Este caso de uso describe cómo, tras una acción del usuario (crear evaluación, rendir examen o publicar material), el sistema envía un mensaje automático a un grupo de Telegram a través de su API oficial. |
| **Precondiciones** | El sistema debe tener configurado correctamente el token del bot de Telegram y el ID del grupo o canal. |
| **Narrativa de caso de uso** | |
| **Acción del Actor** | **Respuesta del Sistema** |
| 1. El actor (Docente, Estudiante o Administrador) ejecuta una acción (crear evaluación, rendir examen o publicar material). | 2. El sistema registra y valida la acción. |
|  | 3. Se genera un mensaje con la información relevante. |
|  | 4. El sistema invoca el Telegram Service, que actúa como intermediario para comunicarse con la API externa. |
|  | 5. El servicio de Telegram recibe la solicitud y publica el mensaje en el grupo. |

*Fuente: Elaboración propia*

*Tabla N°12: Escenario de caso de uso-Realizar Examen*

| **Escenario de caso de uso: Realizar Examen** | |
| --- | --- |
| **Tipo** | **Obligatorio** |
| **Versión** | V 1.0 |
| **Autor(es)** | Concha Llaca, Gerardo Alejandro  Japura Quispe, Herminia Aurelia |
| **Actores** | Estudiante |
| **Descripción** | Permite a los estudiantes realizar un examen previamente asignado por el docente o administrador. El sistema califica automáticamente el examen al finalizar y muestra el resultado al estudiante. Solo se permite un intento |
| **Precondiciones** | El estudiante debe haber iniciado sesión.  Debe existir un examen asignado a su curso.  El estudiante no debe haber rendido anteriormente dicha evaluación. |
| **Narrativa de caso de uso** | |
| **Acción del Actor** | **Respuesta del Sistema** |
| 1. El estudiante inicia sesión. | 2. El sistema valida credenciales y muestra la lista de cursos inscritos. |
| 3. El estudiante accede al módulo de evaluaciones | 4. El sistema valida que el estudiante no haya rendido ya dicha evaluación. |
| 5. El estudiante selecciona una evaluación disponible. |  |
| 6. El estudiante inicia el examen. | 7. El sistema presenta las preguntas con sus respectivas opciones de respuesta. |
| 8. El estudiante responde todas las preguntas y envía el examen. | 9. El sistema muestra al estudiante su puntaje final y estado (aprobado o desaprobado). |
|  | 10. El sistema muestra al estudiante su puntaje final y estado (aprobado o desaprobado). |
| **Flujo Alternativo FA1: Estudiante intenta rendir una evaluación ya rendida** | |
| 1. El estudiante intenta seleccionar una evaluación ya realizada. | 2. El sistema detecta que el estudiante ya completó la evaluación y bloquea el acceso |
| **Flujo de Excepción FE4: Estudiante intenta enviar la evaluación con preguntas sin responder** | |
| 1. El estudiante intenta enviar el examen sin responder todas las preguntas. | 2. El sistema valida las respuestas, detecta preguntas sin respuesta, y muestra un mensaje de advertencia: "debe seleccionar una respuesta." |
|  | 3. El sistema permite al estudiante completar las preguntas restantes y volver a enviar. |

*Fuente: Elaboración propia*

*Tabla N°13: Escenario de caso de uso-Subir Entregable*

| **Escenario de caso de uso: Subir Entregable** | |
| --- | --- |
| **Tipo** | **Obligatorio** |
| **Versión** | V 1.0 |
| **Autor(es)** | Concha Llaca, Gerardo Alejandro  Japura Quispe, Herminia Aurelia |
| **Actores** | Estudiante |
| **Descripción** | Permite a los estudiantes subir entregables de tareas o exámenes en formato PDF para cada material asignado dentro de los cursos. |
| **Precondiciones** | El estudiante debe estar autenticado en el sistema.  El material debe estar disponible para entrega.  El archivo debe estar en formato PDF. |
| **Narrativa de caso de uso** | |
| **Acción del Actor** | **Respuesta del Sistema** |
| 1. El estudiante inicia sesión. | 2. El sistema valida credenciales y muestra la lista de cursos inscritos. |
| 3. El estudiante accede al curso donde está el material para entregar. |  |
| 5. El estudiante selecciona el botón "Subir entregable" correspondiente al material. | 6. El sistema muestra un formulario para subir el archivo. |
| 7. El estudiante selecciona el archivo PDF que desea entregar. |  |
| 8. El estudiante confirma la selección y presiona "Subir". | 9. El sistema verifica que el archivo sea un PDF. |
|  | 10. El sistema guarda el archivo y asocia el entregable con el estudiante y el material correspondiente. |
| **Flujo Alternativo FA1: Reemplazo de entregable existente** | |
| 1. El estudiante intenta subir un nuevo archivo PDF para un material donde ya existe una entrega previa. | 2. El sistema elimina la entrega anterior del estudiante para ese material. |
|  | 3. El sistema guarda el nuevo archivo PDF, lo asocia al estudiante y al material, y muestra el mensaje: "Tu entrega fue registrada correctamente." |
| **Flujo de Excepción FE1: No se selecciona un archivo** | |
| 1. El estudiante intenta subir un entregable sin haber seleccionado un archivo. | 2. El sistema muestra el mensaje: " selecciona un archivo." |
| **Flujo de Excepción FE2: Formato de archivo no permitido** | |
| 1. El estudiante selecciona un archivo que no es PDF. | El sistema muestra el mensaje: "error al subir archivo." y no permite subir el archivo. |

*Fuente: Elaboración propia*

*Tabla N°14: Escenario de caso de uso-Visualizar Entregable*

| **Escenario de caso de uso: Visualizar Entregable** | |
| --- | --- |
| **Tipo** | **Obligatorio** |
| **Versión** | V 1.0 |
| **Autor(es)** | Concha Llaca, Gerardo Alejandro  Japura Quispe, Herminia Aurelia |
| **Actores** | Docente |
| **Descripción** | Permite a los docentes ver los entregables enviados por los estudiantes para cada material de curso, incluyendo la opción de revisar los archivos enviados. |
| **Precondiciones** | El docente debe estar autenticado en el sistema.  Los estudiantes deben haber subido entregables previamente. |
| **Narrativa de caso de uso** | |
| **Acción del Actor** | **Respuesta del Sistema** |
| 1. El docente inicia sesión. | 2. El sistema valida credenciales y muestra la lista de cursos inscritos. |
| 3. El docente accede al curso donde está el material para revisión. |  |
| 5. El docente selecciona el material para revisar las entregas de los estudiantes. | 6. El sistema muestra una lista de los estudiantes que han entregado el material. |
| 7. El docente selecciona un estudiante de la lista. | 8. El sistema muestra el entregable subido por el estudiante, incluyendo el nombre del archivo y el contenido del mismo. |
| **Flujo Alternativo FA2: No existen entregables** | |
| 1. El docente accede al material, pero ningún estudiante ha subido entregas. | 2. El sistema muestra un mensaje: "No hay entregas registradas para este material." |

*Fuente: Elaboración propia*

*Tabla N°15: Escenario de caso de uso-Girar ruleta diaria*

| **Escenario de caso de uso: Girar ruleta diaria** | |
| --- | --- |
| **Tipo** | **Obligatorio** |
| **Versión** | V 1.0 |
| **Autor(es)** | Concha Llaca, Gerardo Alejandro  Japura Quispe, Herminia Aurelia |
| **Actores** | Docente |
| **Descripción** | Permite a los estudiantes girar una ruleta una vez al día. Si el estudiante inicia sesión por primera vez en el día, el sistema muestra automáticamente una ruleta sobrepuesta a la vista principal (Index) como parte del sistema de gamificación. |
| **Precondiciones** | El estudiante debe iniciar sesión en el sistema.  No debe haber girado la ruleta en la fecha actual. |
| **Narrativa de caso de uso** | |
| **Acción del Actor** | **Respuesta del Sistema** |
| 1. El estudiante inicia sesión. | 2. El sistema valida las credenciales e identifica al usuario. |
|  | 3. El sistema verifica si el estudiante ya giró la ruleta en el día actual (VerificarJuegoHoy). |
|  | 4. Si no ha girado la ruleta hoy, el sistema muestra automáticamente una ruleta sobrepuesta a la vista principal (Index). |
| 5. El estudiante gira la ruleta. | 6. El sistema selecciona una pregunta aleatoria desde un archivo JSON (ObtenerPregunta).. |
| 7. El estudiante responde la pregunta. | 8. El sistema evalúa la respuesta y asigna un puntaje si es correcta. |
|  | 9. 12. El sistema registra el puntaje obtenido (GuardarPuntaje) y oculta la ruleta hasta el día siguiente. |
| **Flujo Alternativo FA2: El estudiante ya giró la ruleta hoy** | |
|  | 1. El sistema verifica y detecta que el estudiante ya giró la ruleta en el día actual. |
|  | 2. La ruleta no se muestra y el sistema continúa con la carga habitual del Index. |

*Fuente: Elaboración propia*

*Tabla N°26: Escenario de caso de uso-Mostrar Ranking*

| **Escenario de caso de uso: Mostrar Ranking** | |
| --- | --- |
| **Tipo** | **Obligatorio** |
| **Versión** | V 1.0 |
| **Autor(es)** | Concha Llaca, Gerardo Alejandro  Japura Quispe, Herminia Aurelia |
| **Actores** | Docente |
| **Descripción** | Permite al docente visualizar un ranking con los tres alumnos que han acumulado mayor puntaje diario mediante la ruleta, como parte del módulo de Evaluaciones. |
| **Precondiciones** | El docente debe estar autenticado en el sistema.  Debe existir al menos un alumno con puntaje registrado. |
| **Narrativa de caso de uso** | |
| **Acción del Actor** | **Respuesta del Sistema** |
| 1. El docente inicia sesión. | 2. El sistema valida las credenciales e identifica al usuario. |
| 3. El docente accede al módulo evaluaciones | 4. El sistema consulta los puntajes acumulados por los alumnos. |
| 5. El docente visualiza la sección de ranking. | 6. El sistema muestra el Top 3 de alumnos con mayor puntaje diario. |
| **Flujo de Excepciones – FE1** | |
| 1. El docente accede al módulo, pero no hay alumnos con puntaje registrado. | 2. El sistema muestra el mensaje: "Aún no hay alumnos con puntaje disponible". |

*Fuente: Elaboración propia*

*Tabla N°25: Escenario de caso de uso-administrar Juegos Recreativos*

| **Escenario de caso de uso: administrar Juegos Recreativos** | |
| --- | --- |
| **Tipo** | **Obligatorio** |
| **Versión** | V 1.0 |
| **Autor(es)** | Concha Llaca, Gerardo Alejandro  Japura Quispe, Herminia Aurelia |
| **Actores** | Administrador, Docente |
| **Descripción** | Permite al Administrador y al Docente crear, editar y eliminar juegos recreativos (actualmente Sopa de Letras), definir su contenido (palabras) y asociarlos a un curso específico. |
| **Precondiciones** | El usuario (Administrador o Docente) debe estar autenticado en el sistema.  Deben existir cursos registrados en el sistema para poder asociar los juegos. |
| **Narrativa de caso de uso** | |
| **Acción del Actor** | **Respuesta del Sistema** |
| 1. El Administrador o Docente inicia sesión en el sistema. | 2. El sistema valida las credenciales e identifica al usuario. |
| 3. El actor navega a la sección "Juegos Recreativos". | 4. El sistema muestra una lista con los juegos creados con su curso asignado |
| 5. El actor selecciona la opción "Crear Nuevo Juego" | 6. El sistema muestra los campos "Título", "Tipo de Juego" (con un selector que por defecto o única opción es "Sopa de Letras") y un selector de "Curso" con los cursos disponibles. Para "Sopa de Letras", se presenta un campo de texto para "Contenido" |
| 7. El actor ingresa el "Título" del juego, selecciona el "Tipo de Juego" (actualmente "Sopa de Letras") y el "Curso" al que se asociará. |  |
| 8. Para "Sopa de Letras", el actor ingresa las palabras para la sopa de letras separadas por comas en el campo "Contenido". | 9. El sistema procesa las palabras ingresadas. |
| 10. El actor confirma la creación del juego. | 11. El sistema registra el nuevo juego en la base de datos y lo añade a la lista de juegos, mostrando un mensaje de éxito. |
| **Flujo Alternativo – FA1: Editar Juego** | |
| 1. El actor selecciona un juego existente de la lista. | 2. El sistema carga los detalles del juego seleccionado en un formulario de edición. |
| 3. El actor modifica el "Título", "Tipo de Juego", "Curso" o el "Contenido"del juego | 4. El sistema permite la edición de los campos |
| 5. El actor confirma los cambios. | 6. El sistema actualiza la información del juego en la base de datos, refresca la lista y muestra un mensaje de éxito. |
| **Flujo Alternativo – FA2: Eliminar Juego** | |
| 1. El actor selecciona un juego existente de la lista. | El sistema solicita confirmación antes de eliminar. |
| El actor selecciona la opción "Eliminar" | El sistema procesa la solicitud de eliminación. |
| El actor confirma la eliminación del juego. | El sistema elimina el juego de la base de datos. |

*Fuente: Elaboración propia*

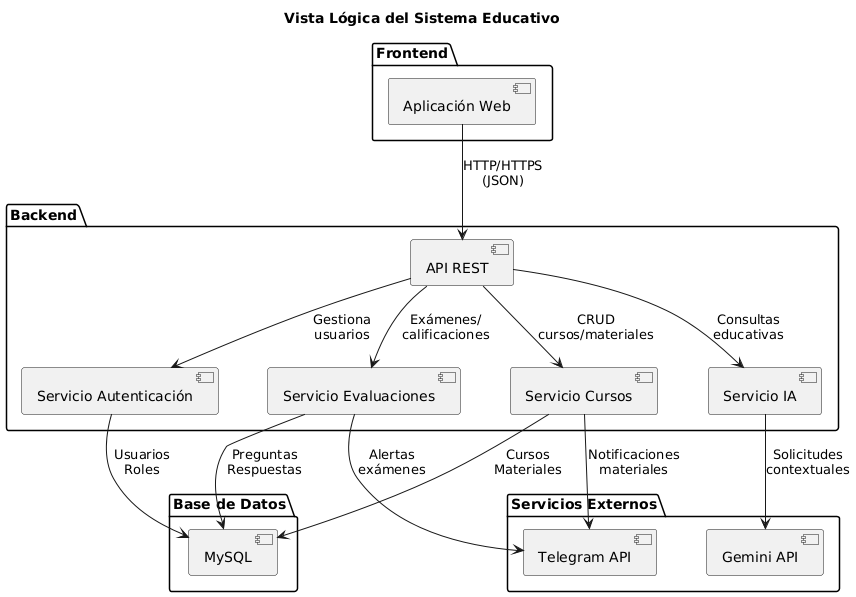
*Tabla N°25: Escenario de caso de uso-Jugar Juegos Recreativos*

| **Escenario de caso de uso: Jugar Juegos Recreativos** | |
| --- | --- |
| **Tipo** | **Obligatorio** |
| **Versión** | V 1.0 |
| **Autor(es)** | Concha Llaca, Gerardo Alejandro  Japura Quispe, Herminia Aurelia |
| **Actores** | Estudiante |
| **Descripción** | Permite a los estudiantes acceder a la lista de juegos recreativos disponibles para sus cursos y seleccionar uno para jugar. El sistema debe presentar la interfaz del juego seleccionada (ej. la sopa de letras con sus palabras). |
| **Precondiciones** | El estudiante debe estar autenticado en el sistema.  Debe haber al menos un juego recreativo asignado a alguno de los cursos en los que el estudiante está inscrito. |
| **Narrativa de caso de uso** | |
| **Acción del Actor** | **Respuesta del Sistema** |
| 1. El Estudiante inicia sesión en el sistema. | 2. El sistema valida las credenciales y redirige al Estudiante al dashboard, mostrando su perfil y cursos asociados. |
| 3. El Estudiante navega a la sección "Juegos Recreativos" en el módulo de Gamificación. | 4. El sistema muestra una interfaz que lista los juegos recreativos disponibles, filtrados por los cursos a los que el estudiante está inscrito. Cada juego muestra su título y tipo. |
| 5. El Estudiante visualiza la lista de juegos disponibles para sus cursos. | 6. El sistema presenta la lista de juegos, indicando claramente el nombre y el curso al que pertenecen.de Letras") y un selector de "Curso" con los cursos disponibles. Para "Sopa de Letras", se presenta un campo de texto para "Contenido" |
| 7. El Estudiante selecciona un juego de la lista (ej. "Sopa de Letras: Vocabulario de Emprendimiento"). | 8. El sistema carga y muestra la interfaz específica del juego seleccionado. Para una "Sopa de Letras", el sistema genera la cuadrícula con las letras y la lista de palabras a encontrar. |
| 8. El Estudiante interactúa con el juego (ej. busca palabras en la sopa de letras). | 9. El sistema registra las interacciones del Estudiante dentro del juego (ej. palabras encontradas, tiempo transcurrido).. |
| 10. Al finalizar el juego o al encontrar todas las palabras, el Estudiante observa la retroalimentación. | 11. El sistema muestra un mensaje de finalización, un resumen del rendimiento (ej. número de palabras encontradas, tiempo, puntaje), y registra el resultado en la base de datos |
| **Flujo Alternativo – FA2: No hay juegos disponibles** | |
| 1. El Estudiante navega a la sección "Juegos Recreativos", pero no hay juegos asociados a sus cursos. | 2. El sistema muestra un mensaje indicando que no hay juegos recreativos disponibles para el Estudiante. |

*Fuente: Elaboración propia*

2.2 Vista Lógica

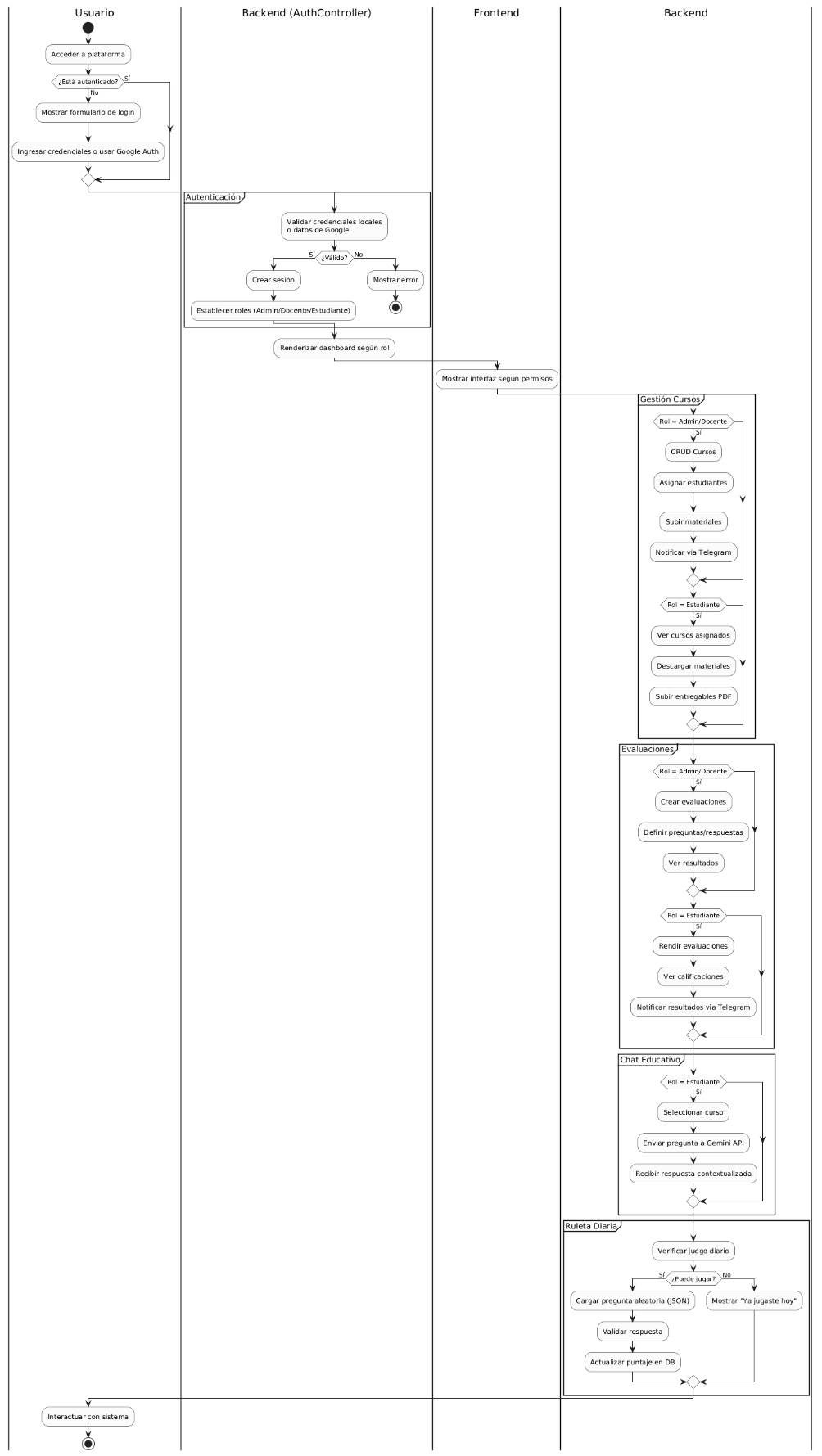
*Figura N°1: Vista Lógica*



*Fuente: Elaboración propia*

2.3 Vista del Proceso [–](https://www.plantuml.com/plantuml/png/dPJFRYD54CRlynIbS8aFAFUC4ElOsMWHXCXc8wt4fT9J7jTkJvTLtPrbGtYOVG1EtBWWuHUZslPuRF0Ys8lTwlfJl-zhJRs82JNbpiCzsn4QWZiLHr4YV0bpkwCEuRh93XC_IbN5TnnMgDX19q7SKgKZI9hf81ANsCXx3YqiqCV323hhmkd35RfthKeEpKIyA7pnBRVBz8Qyb_UdqaKRqctgbRVQWnpxJPnOmZ-JgkVNCQEoF5SRcTNbs3co941OUKou4EsmhdW1vt_-SHtJ-YDWJXGIErFmzGZIaWAStyge0hZyJc9IL2XLsPU-nIulBOUwn5-5LYbQWbCR4XoZfmW2kTpTYBIUO9pJigx8N87p-VhNKKMXuOMH5aVi3ESbOo8XgNXFEdgkzWBXRDp3ELx_5ixWvpBs7ZqtVv_gnPN_7uCnOf9ef3k4WheLVB_-w7cGkUK1i9OJ9MiOAPOnzVRExC47JuuKLAoraJOTXukfE6EY2uldXj6Eer9H9FRE7NX7ggBRZZ79oWvR0twfgikP7Kdv2TMOu_91K1k3QDU_XpAoDfTUgZbZQS_LlYFRXIxmgSzSaNOS9TQ3gKSUtb1CHHPCibgQcRWpP2OUle8ZKIUScRsUxYftlemZjw4es-ll8_Fym0eT6X-Ztr_VIk85EojvP8GVo5EhsDLxAm2EW7fVJz7SauBRm01kC9ezrfIYZMZn1C2Mo_GfjUL98znDNzO7Jt5er_KZ-epEplI_lL8glWmD1hO51qDO6K0E2UE5KbnbYdQi1sLscNqgcZxBd5cfE3L_QnlwyWf3uFXfXjb7h_GfboPBJCEMD9_-C-NSEhi2WRkdx3-tO7C7Ym3XXinjXl7TgxeNwBYyvDuyS68pVahPj3rXWv-YdMLFbZsr0Opz5Y7TIN-RgHLeIb26PN6NgI6BjQYd5WPgUqJh2PDOTpZ_PlxzxQZq6LRLGDn_Gsx7R2rA_Li2wp-z-fT5SlO6DpGn4IpbmzcHrc7lsr9Ef5ZwQx478bj5MVgRzLEzi9hS-Ry0)

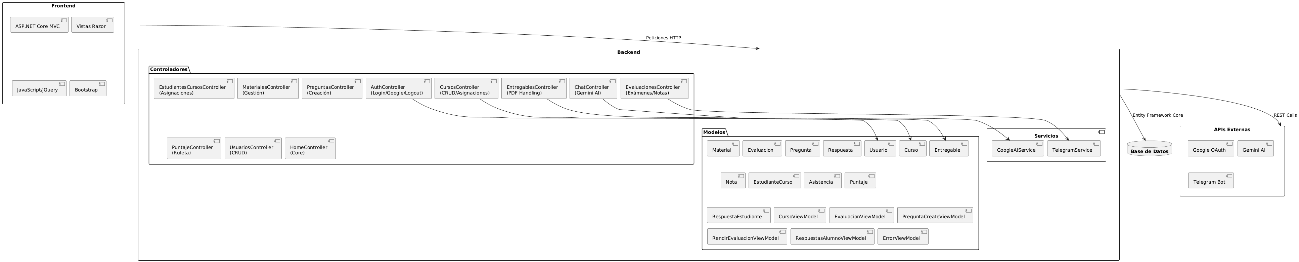
*Figura N°2: Vista del Proceso*



*Fuente: Elaboración propia*

2.4 Vista del desarrollo [-](https://www.plantuml.com/plantuml/png/bLLBSjj64DtZ55FSn6QLmsIHJHOfGnJrIKKEGzBQ95eqCMrehC4qghj7afFIONA0h784NIpu20m0gYnAAtM_rt_CutjHO8s5DvTE58vHW9cyftLbOW4B6pDGTqj98ZSkbC1Gc882PTTC1HhbY3r4hi7IdGkv-GHU-aWWYuAb-VwxdfCXt3JE7vBa6pFzsb-3cnEce1ZiYsp6J27a7iraEksYfjE90J6VQjFybHYJKL5ImCgQfElbjnyM6pCdHdDnEPzKyE_DJiIiu4_YgzhnCzp2EcDNwkppRn7vI-Cz8b9HXl8gUJXma2F8RbuzHnlKZR6jh6QAile7yeenfw1C7YmnogJ1gkxIgDTFY4V-8xpvXN8NPgT4LTfPPL3KjrSjTnvPI0Ri-UhZyImLbmV870MK7VSQTC0ynS85PzBpZh6eGCnXwt6OSdbyOiuWMEz2lkFUWezjXI5vSV_uTu6LT_Q156J75utMGL2KPvf-hjyBK6G7usPEKTGz_Xiwsf8nZq5Xj0L6o0QqclCP1wHLzAZGCJvA17RFRBCZd561Gv0O6_0XwP_rWYnwgWywIDm_sMZ4lPsEPdioLoXbHDdPzOBtJhUtxtx1L9meXip1S2tZ0lz7zxk-T7XN3xVVxnZf6g_Fe3X6LnYiu-UZ1sr8wcCHzkio4m-S3ybG7TR8jovplIDiq6FEKBH8DrRxgDBpdlVX5PeWQ2oQOr0IyqR8az77VpHwUljY1WiAMv1M9_g9djJ2RW_i8bsUYrdSAt80UPq-zKExkb2wUcNTQioljHmrghcJYzhQxTCSaHukeolqJw_yHNwh-k_U_TGAvuzcYUhQM7EssIoJsjtWTbkXYw1ElvWJXWBlY6-Q7uOTfnwh8gqMwusPW_TIzpkdWFTjGekYa845IGxKtpfhxsaVfiHjL2S1NrFabZdwF9F3LRUD7ttqoO4ANySE9ILvZy76mYV_0G00)

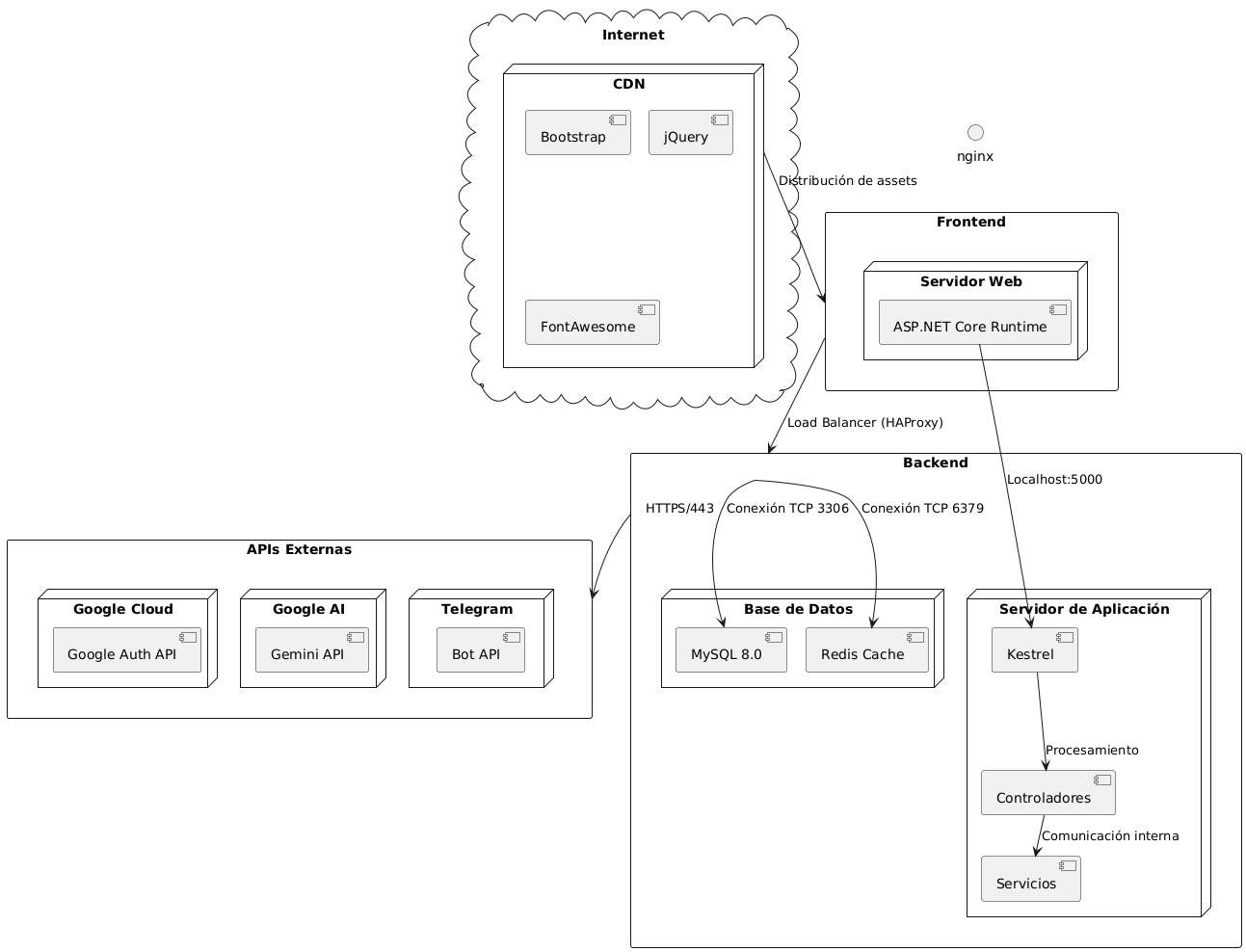
*Figura N°3: Vista del desarrollo*



*Fuente: Elaboración propia*

2.5 Vista Física

*Figura N°4: Vista Física*

*Fuente: Elaboración propia*

3. Objetivos y limitaciones arquitectónicas

3.1 Seguridad

El sistema implementa un enfoque de defensa en profundidad para garantizar la protección de datos sensibles y cumplir con normativas como GDPR/LOPD. Aunque la integración con APIs externas (Google, Telegram, Gemini) introduce riesgos, se mitigan mediante:

* Autenticación robusta: JWT con OAuth2 y doble factor (2FA) opcional.
* Cifrado: AES-256 para datos en tránsito (TLS 1.3) y en reposo (BBDD).
* Control de accesos: Modelo RBAC (Roles: Admin, Docente, Estudiante) con políticas granulares.
* Monitorización: Detección de intrusiones con WAF (Web Application Firewall) y auditorías trimestrales.
* Limitación: La complejidad de gestionar múltiples roles y permisos puede ralentizar el desarrollo de nuevas funcionalidades.

3.2 Escalabilidad

Diseñado para soportar un crecimiento anual del 50% en usuarios y contenido, la arquitectura combina:

* Microservicios estratégicos: Servicios independientes para autenticación, evaluaciones y chat (IA).
* Contenedores Docker: Escalado horizontal automático en Kubernetes basado en demanda.
* BBDD escalable: MySQL con particionamiento horizontal (sharding) para tablas críticas (ej: Entregables).

3.3 Tiempos de Respuesta Rápidos

Diseñado para soportar un crecimiento anual del 50% en usuarios y contenido, la arquitectura combina:

Microservicios estratégicos: Servicios independientes para autenticación, evaluaciones y chat (IA).

3.4 Optimización de Carga

Para manejar picos de tráfico (ej: entregas masivas de PDFs o exámenes simultáneos):

* Compresión: Gzip para APIs y WebSockets.
* Tests regulares: Simulaciones de carga con JMeter (500+ RPS).
* Limitación: Almacenamiento de archivos grandes (ej: videos) requiere ajustes en la infraestructura de almacenamiento (S3 compatible).

4. Análisis de Requerimientos

4.1 Requerimientos funcionales

*Tabla N°1: Cuadro de requerimientos funcionales final*

| **ID** | **Requerimiento Funcional** | **Descripción** | **Prioridad** | **Módulo** | **Regla de Negocio** |
| --- | --- | --- | --- | --- | --- |
| **RF-01** | Iniciar sesión | Permitir acceso con email/contraseña para Administrador, Docente y Estudiante. | Alta | Autenticación | RN-01 |
| **RF-02** | Autenticar con Google | Permitir SSO con Google solo para correos institucionales. | Alta | Autenticación | RN-02 |
| **RF-03** | Gestionar usuarios | Administrador puede crear/editar/eliminar usuarios y asignar roles. | Alta | Usuarios | RN-03 |
| **RF-04** | Registrar estudiantes | Docentes pueden crear usuarios con rol Estudiante. | Media | Usuarios | RN-04 |
| **RF-05** | Administrar cursos | Crear cursos y asignar Docentes/Estudiantes (solo Admin). | Alta | Cursos | RN-05 |
| **RF-06** | Publicar materiales | Docentes pueden subir material a sus cursos. | Alta | Materiales | RN-06 |
| **RF-07** | Gestionar evaluaciones | Los docentes pueden crear, editar, eliminar y visualizar evaluaciones (exámenes o quizzes) asociadas a sus cursos. | Alta | Evaluaciones | RN-07 |
| **RF-08** | Acceder a materiales | Estudiantes pueden acceder a recursos subidos. | Media | Materiales | RN-08 |
| **RF-09** | Interactuar con Chat IA | Estudiantes hacen consultas basadas en materiales del curso. | Alta | Chat IA | RN-09 |
| **RF-10** | Notificar por Telegram | El sistema debe enviar mensajes automáticos a un grupo de Telegram cuando se crea una evaluación, un estudiante rinde un examen, o se publica nuevo material. | Media | Notificaciones | RN-10 |
| **RF-11** | Realizar Examen | El estudiante podrá rendir un examen asignado solo una vez. Al finalizar, el sistema mostrará su calificación de forma automática. | Alta | Evaluaciones | RN-11 |
| **RF-12** | Subir entregables | Los estudiantes pueden subir un archivo PDF como entregable para cada material asignado. Si ya existe uno, se reemplaza. | Alta | Entregables | RN-12 |
| **RF-13** | Visualizar entregables | Los docentes pueden ver una lista de los estudiantes que realizaron una entrega para cada material, incluyendo el nombre del archivo y la posibilidad de descargarlo. | Alta | Entregables | RN-13 |
| **RF-14** | Girar ruleta diaria | Al iniciar sesión, el estudiante puede girar una ruleta 1 vez por día. | Alta | Gamificación | RN-14 |
| **RF-15** | Mostrar ranking | El sistema debe mostrar en el módulo de Evaluaciones a los tres alumnos con mayor puntaje acumulado obtenido mediante la ruleta diaria. | Alta | Evaluaciones | RN-15 |
| **RF-16** | Administrar Juegos Recreativos | Permitir al Administrador y Docente crear, editar y eliminar juegos recreativos. Se debe poder especificar el tipo de juego (actualmente "Sopa de Letras"), definir su contenido (ej. lista de palabras para sopa de letras) y asociar el juego a un curso específico. | Alta | Gamificación | RN-16 |
| **RF-17** | Jugar Juegos Recreativos | Permitir a los estudiantes acceder a la lista de juegos recreativos disponibles para sus cursos y seleccionar uno para jugar. El sistema debe presentar la interfaz del juego seleccionada (ej. la sopa de letras con sus palabras). | Alta | Gamificación | RN-17 |
| **RF-18** | Jugar Juegos de Terceros | El sistema debe permitir a los estudiantes acceder y visualizar una colección predefinida de juegos educativos incrustados desde plataformas de terceros (ej. Wordwall), presentados mediante iframes. | Alta | Gamificación | RN-18 |

*Fuente: Elaboración propia*

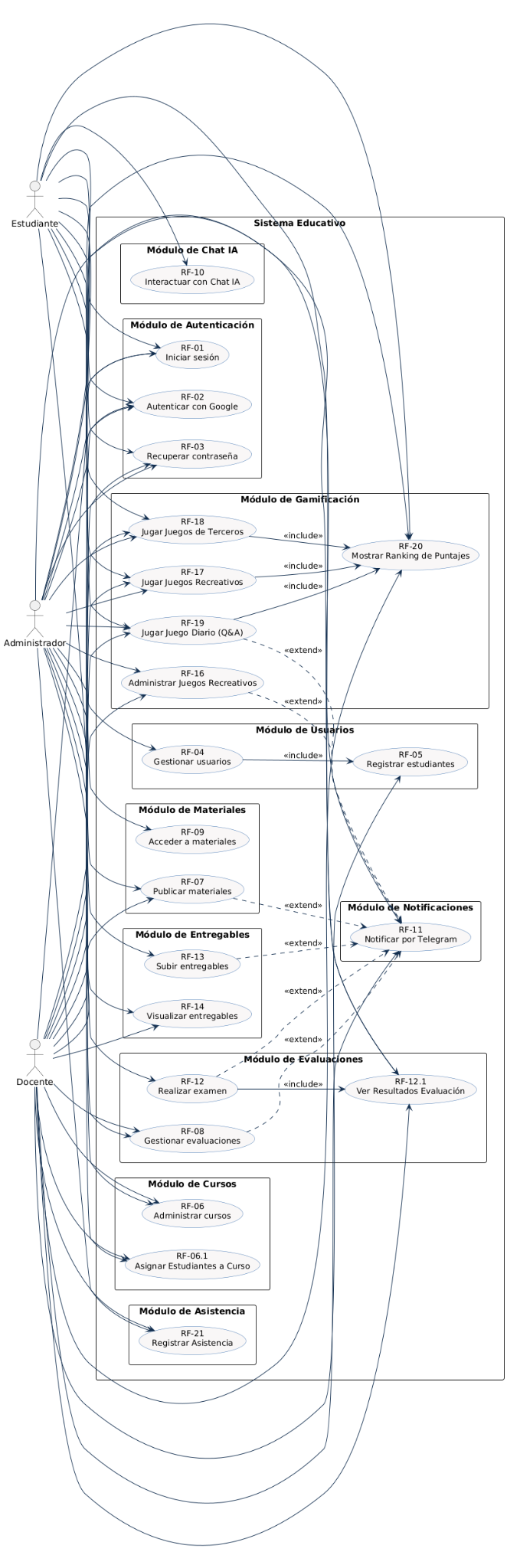
4.2 Requerimientos no funcionales

Requerimientos No Funcionales – Atributos de Calidad

| **Nro.** | **Requerimiento No Funcional** | **Descripción** | **Prioridad** |
| --- | --- | --- | --- |
| **RNF – 01** | Seguridad | La plataforma debe implementar cifrado de datos (SSL) para garantizar la seguridad de la información personal y académica de los estudiantes y docentes. | 3 |
| **RNF – 02** | Escalabilidad | El sistema debe ser capaz de manejar un gran número de usuarios (estudiantes y docentes) sin que afecte el rendimiento, especialmente durante períodos de alta demanda, como la entrega de tareas o evaluaciones. | 3 |
| **RNF-07** | Tiempos de Respuesta Rápidos | Los cambios en los contenidos educativos, las calificaciones deben reflejarse en tiempo real para evitar retrasos en la actualización de la información y asegurar la sincronización de datos entre todos los usuarios. | 2 |
| **RNF-9** | Optimización de Carga | Las imágenes y recursos interactivos deben estar optimizados para evitar tiempos de carga lentos, especialmente en conexiones móviles, asegurando que los estudiantes puedan acceder a los materiales educativos sin problemas. | 2 |

5. Vistas de Caso de Uso

*Figura N°18: Diagrama de caso de uso general* [*-*](https://www.plantuml.com/plantuml/png/ZPRTRjis5CVlzHH41kpZee4bEPOz544zVmGfbg1Bcc8N1emJwbJX8f41FxgkGnzgw2FanKR94kCmP1DVIR_3Fy__71v9VgCqI6sQEgdneoPQ4Ccg6qrA9f5g9dYYRXc_0maDkGDw2nLUwdzg96qOU5NZtWAZa892ycz2o6zsSIM5uUL2r4AI7zQpTR4kseYG9SgUvkiYcwyjdKif_kvXccRBySh2jU3wadr1akR9roG1gcruNZQCCwKbb4BsR2aeSetztKffKp9eGU9yaeDBAy86oAeq53Jx90wiKvk5aBr5P_VVIbCBKYAP6sqtPHGek__63mWe0aRVTELrFrVmmSNwrIZTy5FEA0D952edkLgCqh0Wst2NGX8gE3aHmfeOP5bObc_u1L9pXtAdqjA6xl-3GPPtig_Vg-nA6P1Cg4vY-fjONUCDFq7LpW98jtX8DWvhZbgFLNT8ag0x3YSxUjxZmaZLvw7sCkvlik4F8o5tYrsYIKHpc5gLOXM7_N5H17PvdNoRFk_q33HA1dLVNkDkOuwB3NzdhklksDrgPxa8YsRMCALOeYGGKCsUzxdYMc85ruCAtNtCQMfdxT9SC-dMFcHCyx34Zii7fWpKx0i4TUCNEFq4jG7A11-iEX3lwdH_JlS4haNJiE6idTN1xcTea0zEiuYW7PuFACa5AbDhA8LoZjrJRzKlcPt53MXoEkyqr5x73cBKlbmqIg3Qx1vx9-tJZPvFTW8D-_ZeVLOv8dWijVTylJLO2MLhfnAxjwWxs4bORwVzhQc-gop2ogcdB96yHqbHFYYdMzsJy1QpHrkG9GF91FdvZn_dlmnRpBPbHuCRPBRnPs9N-GNmMyQhriOxmpNyvKOhUq7ppuLshUw5_16AzTyEsw2Lv4v8suCQAmdDKC0BncokcDB8AODE1EusaZHBzr_WdjhMEwIqIN-o1reF9TXKtISOLVBeCqrUlJhkleHXd8Ln7iRZC9v4y3QIj0ZZQH2dQHYFmtWIne6KtX27ucv2Wz5i5CO1myDRAUd_BhbJyK7cWzm7HpwOF07Rr4U53wOUI5CVZ7qmyO7RDD1DBpBqqUFPo0UUZRvh3_yJNEC2B0kmFC1cJraw2Z1lly5EWEO1LZWMQb0Ww7hq99IDl8Vzz7pn-zLobRIFf9iByYjv_PfnMfiIZu-JrkoUwwVHmktkn_QDnXSvmv4bdkVLd-zNvylMSa4E3pjRErFuMIClkor7qsWepUAXF1wQn4FEVdX5ahon5wQfa_y1)

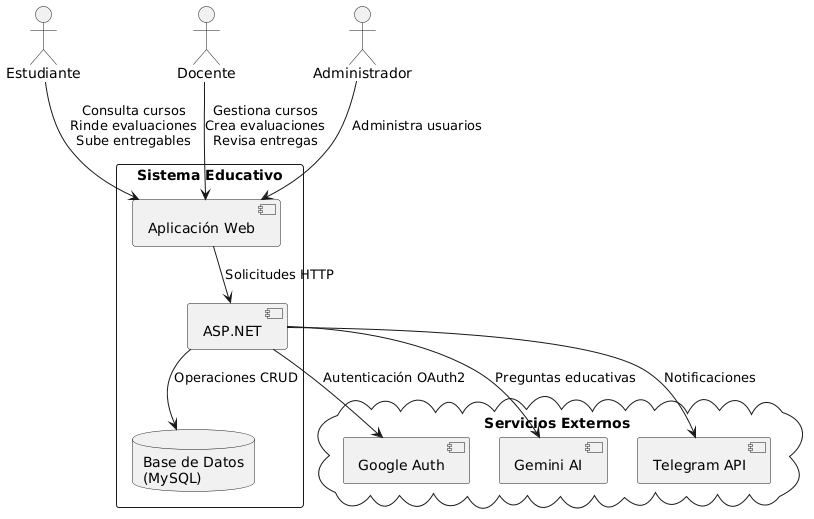


*Fuente: Elaboración propia*

6. Vista Lógica

6.1 Diagrama Contextual

*Figura N°5: Diagrama Contextual*

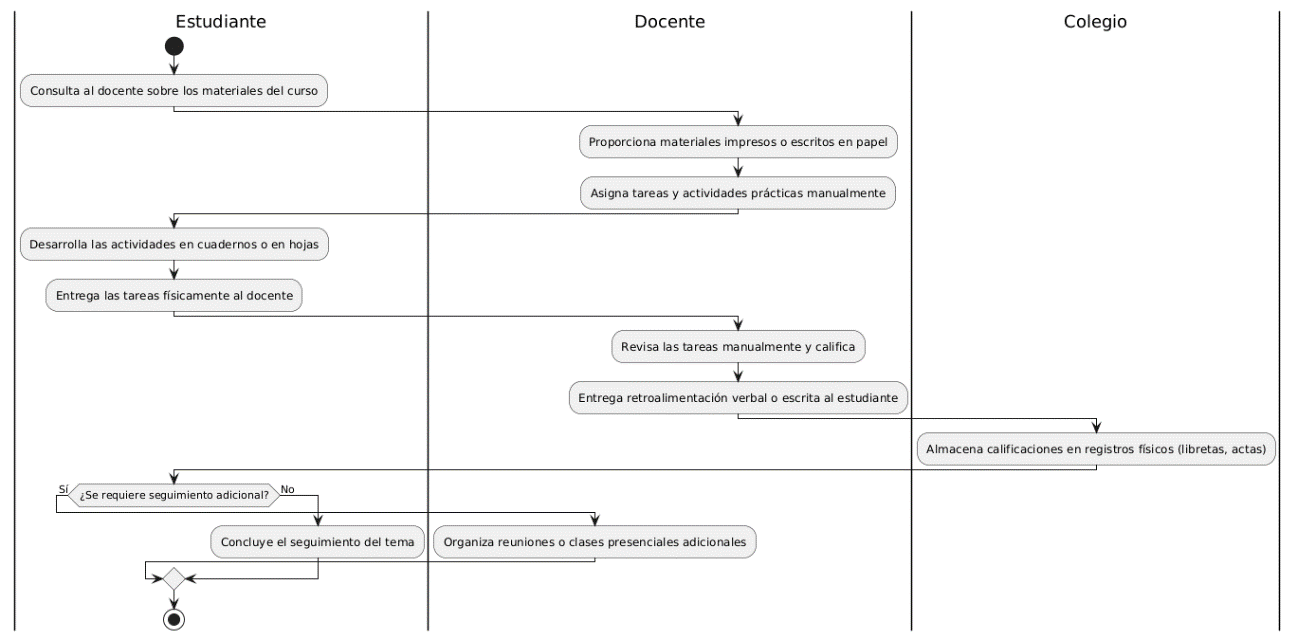


*Fuente: Elaboración propia*

7. Vista de Procesos

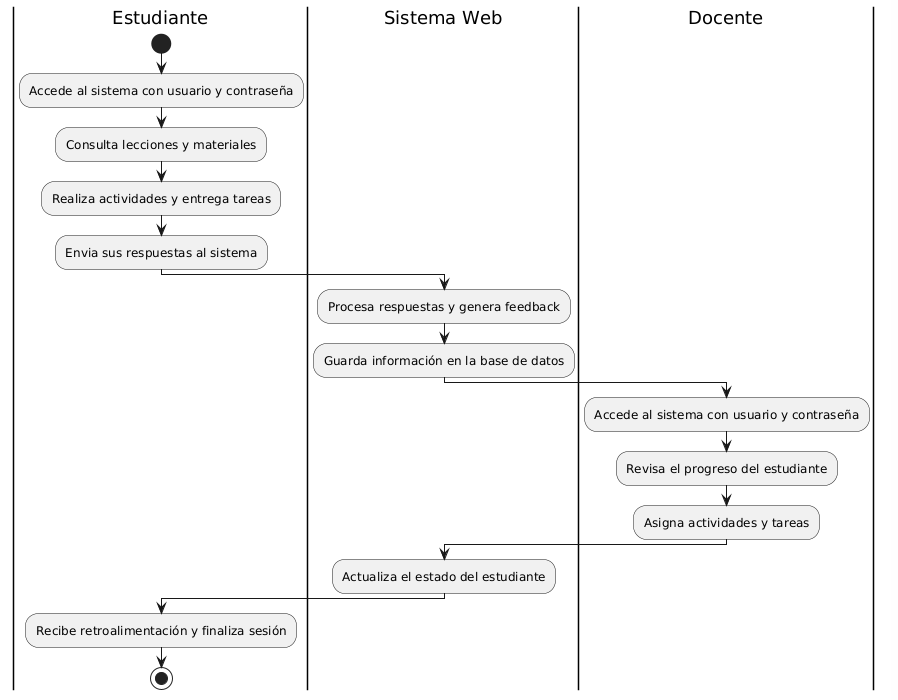
7.1 Diagrama de Proceso Actual

*Figura N°6: Diagrama de Actividades Proceso Actual*

*Fuente: Elaboración propia*

7.2 Diagrama de Proceso Propuesto

*Figura N°7: Diagrama de Actividades Proceso propuesto*

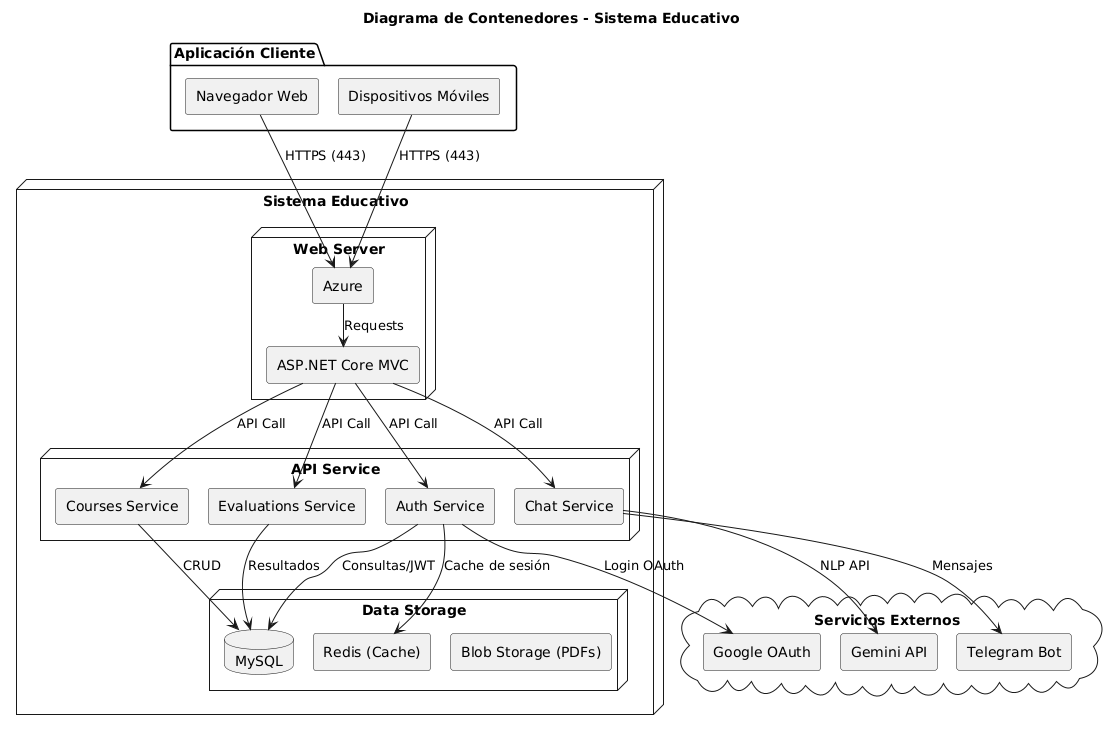


*Fuente: Elaboración propia*

8. Vista de Despliegue

8.1 Diagrama de Contenedor

*Figura N°8: Diagrama de contenedor*

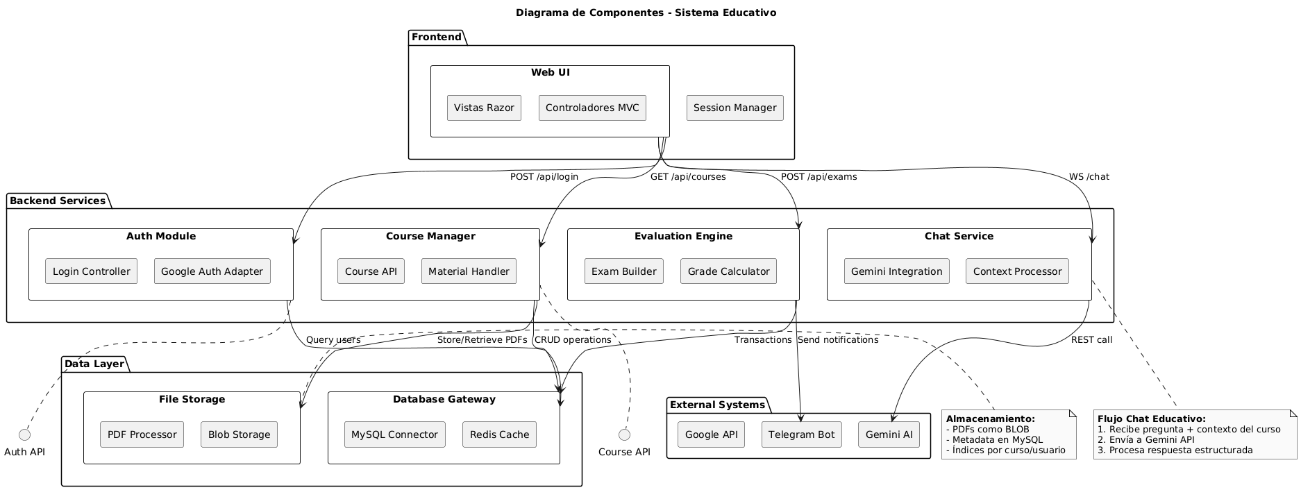


*Fuente: Elaboración propia*

9. Vista de Implementación

9.1 Diagrama de Componentes

*Figura N°8: Diagrama de componentes*

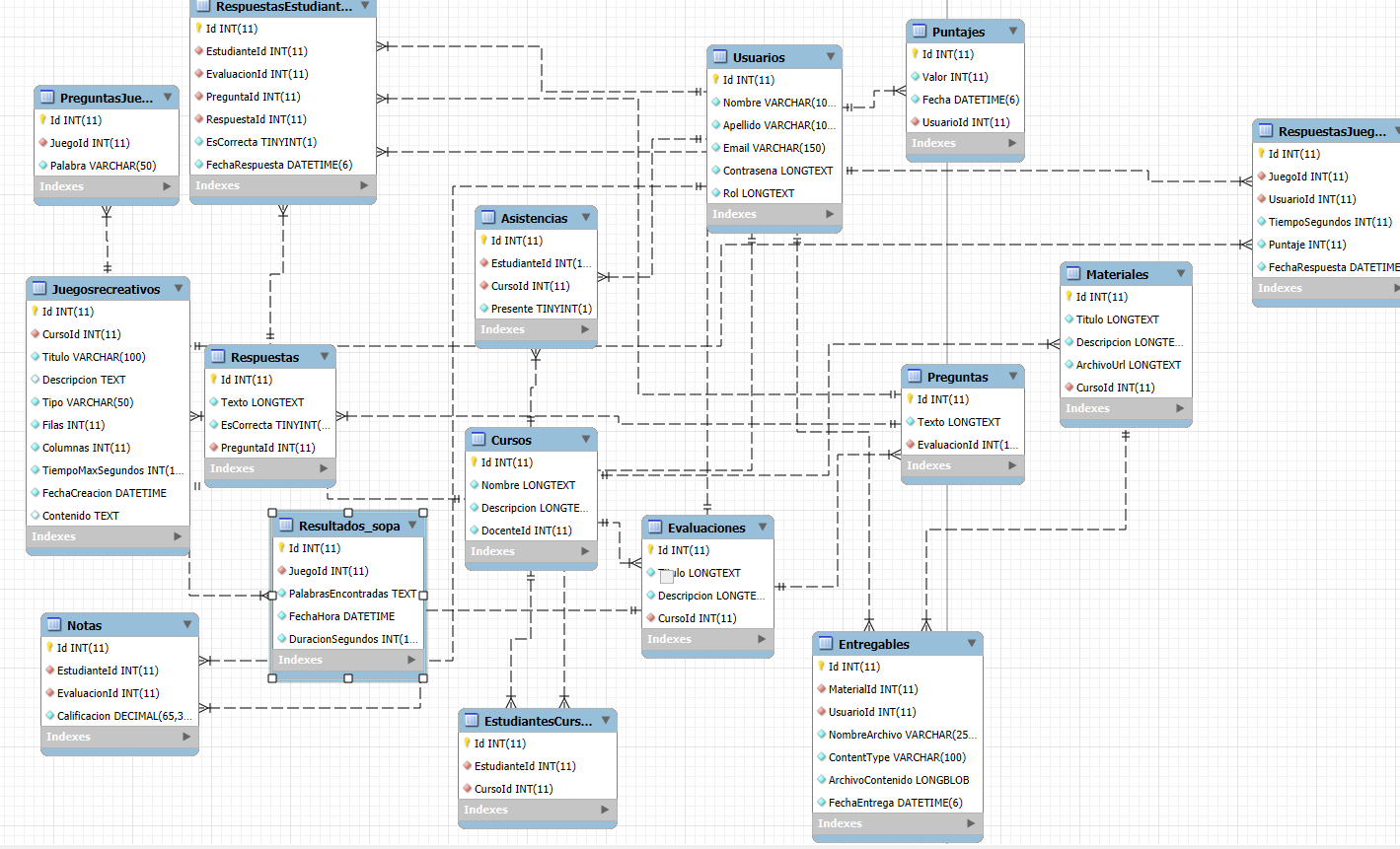


*Fuente: Elaboración propia*

10. Vista de Datos

10.1 Diagrama de Base de Datos

*Figura N°8: Diagrama de Base de Datos*



*Fuente: Elaboración propia*

11. Calidad

11.1 Escenario de Seguridad

**Estímulo**: Intento de acceso no autorizado a datos sensibles (ej: notas de estudiantes).  
**Fuente**: Usuario malintencionado o script automatizado.  
**Artefacto**: Endpoints de API (/api/evaluaciones, /api/usuarios).  
**Entorno**: Sistema en producción con autenticación activa.  
**Respuesta**:

* Bloqueo de IP después de 5 intentos fallidos (rate limiting).
* Registro de auditoría con timestamp e IP.
* Notificación inmediata al administrador vía Telegram.

**Métrica**:

* Tiempo de detección <2 minutos.
* 0 accesos no autorizados exitosos.

11.2 Escenario de Escalabilidad

**Estímulo**: Incremento del 200% en solicitudes concurrentes (ej: inicio de semestre).  
**Fuente**: Usuarios (estudiantes/docentes) accediendo simultáneamente.  
**Artefacto**: Servidores de aplicación y base de datos.  
**Entorno**: Horas pico (8:00 AM - 10:00 AM).  
**Respuesta**:

* Auto-escalado horizontal en Kubernetes (2 → 5 pods).
* Caché de Redis para consultas frecuentes (ej: materiales de cursos).
* Balanceo de carga con HAProxy.  
  **Métrica**:
* Latencia estable (<1s) con 500+ usuarios concurrentes.
* Tiempo de escalado <3 minutos.

11.3 Escenario de Tiempos de Respuesta Rápidos

**Estímulo**: Solicitud de recursos críticos (ej: página principal del curso).  
**Fuente**: Usuario final desde dispositivo móvil.  
**Artefacto**: Frontend y API de cursos.  
**Entorno**: Red 4G con latencia variable.  
**Respuesta**:

* Precarga de assets estáticos (CDN Cloudflare).
* Compresión Gzip para APIs.
* Consultas optimizadas con índices en MySQL.  
  **Métrica**:
* 95% de solicitudes respondidas en <500ms.
* TTFB (Time To First Byte) <200ms.

11.4 Optimización de Carga

**Estímulo**: Envío masivo de archivos PDF (ej: 100+ entregables simultáneos).  
**Fuente**: Estudiantes subiendo tareas antes del plazo.  
**Artefacto**: Servidor de archivos y cola de procesamiento.  
**Entorno**: Horario límite (11:00 PM).  
**Respuesta**:

* Procesamiento asíncrono con RabbitMQ.
* Almacenamiento temporal en S3 compatible.
* Limitación de ancho de banda por usuario (1MB/s).  
  **Métrica**:
* Procesamiento de 10 PDFs/segundo por nodo.
* 0% de pérdida de datos durante los picos.