

# Documento de Propuesta de Diseño de Software I, II y III.

**Tutor:** Alexander Enrique Toscano Ricardo

**Estudiantes:**

Marlon Yesid Cobos Villalobos

Nair José Madera Tardecilla

Felipe Miguel Patrón De La Ossa

Cristian Javier Castaño Martínez



## Breve reseña

DARC: Descubre, Aprende Regiones de Colombia, es una página web educativa dirigida a estudiantes de primaria (7-12 años) que transforma el aprendizaje sobre las regiones colombianas en una experiencia lúdica e interactiva. A través de juegos didácticos, mapas interactivos y actividades que fomentan el aprendizaje autónomo y la colaboración, DARC busca que los niños aprendan sobre la geografía, cultura, historia y biodiversidad de su país donde podrán conectar imágenes, audios y videos arrastrándolas en diferentes regiones de Colombia, ofreciendo un recursos académico y didáctico para docentes.

<b>ETAPA 1 DISEÑO DE LA APLICACIÓN Y ANÁLISIS DE REQUISITOS</b>	<b>5</b>
<b>Introducción</b>	<b>5</b>
Propósito del Documento	5
Alcance del Proyecto	5
Definiciones y Acrónimos	5
<b>Descripción General</b>	<b>5</b>
Objetivos del Sistema	5
Funcionalidad General	5
Usuarios del Sistema	5
Restricciones	5
<b>Requisitos Funcionales</b>	<b>5</b>
Mockup de la Interfaz de Usuario (UI)	6
Casos de Uso	6
Descripción detallada de cada caso de uso	6
Diagramas de Flujo de Casos de Uso	6
Prioridad de Requisitos	6
<b>Requisitos No Funcionales</b>	<b>6</b>
Requisitos de Desempeño	6
Requisitos de Seguridad	6
Requisitos de Usabilidad	6
Requisitos de Escalabilidad	6
<b>Modelado E/R</b>	<b>6</b>
Diagrama de Entidad-Relación	6
Diagrama relacional	7
Descripción de Entidades y Relaciones	7
Reglas de Integridad	7
<b>Anexos (si es necesario)</b>	<b>7</b>
Diagramas Adicionales	7
Referencias	7
<b>ETAPA 2: PERSISTENCIA DE DATOS CON BACKEND</b>	<b>8</b>
<b>INTRODUCCIÓN</b>	<b>8</b>
Propósito de la Etapa	8
Alcance de la Etapa	8
Definiciones y Acrónimos	8
<b>Diseño de la Arquitectura de Backend</b>	<b>8</b>
Descripción de la Arquitectura Propuesta	8
Componentes del Backend	8
Diagramas de Arquitectura	8
<b>Elección de la Base de Datos</b>	<b>8</b>
Evaluación de Opciones (SQL o NoSQL)	8
Justificación de la Elección	8
Diseño de Esquema de Base de Datos	9
<b>Implementación del Backend</b>	<b>9</b>
Elección del Lenguaje de Programación	9
Creación de la Lógica de Negocio	9
Desarrollo de Endpoints y APIs	9

Autenticación y Autorización	9
<b>Conexión a la Base de Datos</b>	<b>9</b>
Configuración de la Conexión	9
Desarrollo de Operaciones CRUD	9
Manejo de Transacciones	9
<b>Pruebas del Backend</b>	<b>9</b>
Diseño de Casos de Prueba	9
Ejecución de Pruebas Unitarias y de Integración	10
Manejo de Errores y Excepciones	10
<b>ETAPA 3: CONSUMO DE DATOS Y DESARROLLO FRONTEND</b>	<b>11</b>
<b>INTRODUCCIÓN</b>	<b>11</b>
Propósito de la Etapa	11
Alcance de la Etapa	11
Definiciones y Acrónimos	11
<b>Creación de la Interfaz de Usuario (UI)</b>	<b>11</b>
Diseño de la Interfaz de Usuario (UI) con HTML y CSS	11
Consideraciones de Usabilidad	11
Maquetación Responsiva	11
<b>Programación Frontend con JavaScript (JS)</b>	<b>11</b>
Desarrollo de la Lógica del Frontend	11
Manejo de Eventos y Comportamientos Dinámicos	11
Uso de Bibliotecas y Frameworks (si aplicable)	12
<b>Consumo de Datos desde el Backend</b>	<b>12</b>
Configuración de Conexiones al Backend	12
Obtención y Presentación de Datos	12
Actualización en Tiempo Real (si aplicable)	12
<b>Interacción Usuario-Interfaz</b>	<b>12</b>
Manejo de Formularios y Validación de Datos	12
Implementación de Funcionalidades Interactivas	12
Mejoras en la Experiencia del Usuario	12
<b>Pruebas y Depuración del Frontend</b>	<b>12</b>
Diseño de Casos de Prueba de Frontend	12
Pruebas de Usabilidad	12
Depuración de Errores y Optimización del Código	13
<b>Implementación de la Lógica de Negocio en el Frontend</b>	<b>13</b>
Migración de la Lógica de Negocio desde el Backend (si necesario)	13
Validación de Datos y Reglas de Negocio en el Frontend	13
<b>Integración con el Backend</b>	<b>13</b>
Verificación de la Comunicación Efectiva con el Backend	13
Pruebas de Integración Frontend-Backend	13

# **Etapas 1 Diseño de la Aplicación y Análisis de Requisitos**

## **Introducción**

DARC: Descubre, Aprende Regiones de Colombia, es una aplicación web educativa dirigida a estudiantes de primaria (7-12 años) que transforma el aprendizaje sobre las regiones colombianas en una experiencia lúdica e interactiva. A través de juegos didácticos, mapas interactivos y actividades que fomentan el aprendizaje autónomo y la colaboración, DARC busca que los niños aprendan sobre la geografía, cultura, historia y biodiversidad de su país donde podrán conectar imágenes, audios y videos arrastrándose en diferentes regiones de Colombia, ofreciendo un recursos académico y didáctico para docentes.

## **Propósito del Documento**

- Definir los requisitos funcionales y no funcionales de DARC.
- Describir la interfaz de usuario y la experiencia del usuario.
- Modelar la estructura de datos de la aplicación.
- Servir como base para la planificación y el desarrollo del software.

## **Alcance del Proyecto**

DARC es una aplicación web educativa que busca hacer del aprendizaje sobre las regiones de Colombia una experiencia divertida e interactiva para niños de primaria. El proyecto abarca el desarrollo de una plataforma que permita a los usuarios explorar un mapa interactivo, acceder a información detallada sobre cada región, participar en juegos didácticos y realizar actividades interactivas de arrastrar y soltar. DARC se enfocará en la usabilidad y accesibilidad para niños, con el objetivo de complementar los recursos educativos existentes y ofrecer herramientas adicionales para docentes.

Registrarse con información básica.

Explorar un mapa interactivo de Colombia.

Acercar y alejar el mapa.

Desplazarse por el mapa.

Acceder a información detallada sobre cada región, incluyendo su ubicación geográfica, cultura y tradiciones, historia y biodiversidad.

Visualizar imágenes, reproducir audios y ver videos de cada región.

Jugar a juegos didácticos con barras de progreso, sistemas de puntos y recompensas.

Realizar actividades interactivas de arrastrar y soltar.

Obtener validación de respuestas y recibir retroalimentación visual y sonora en las actividades.

Monitorear el progreso y la actividad de los usuarios.

Generar reportes y estadísticas sobre el uso de la plataforma.

Definir sistemas de puntos y recompensas (logros) para los juegos.

## **Definiciones y Acrónimos**

DARC: Descubre, Aprende Regiones de Colombia

UI: Interfaz de usuario

UX: Experiencia de usuario

CRUD: Crear, Leer, Actualizar, Eliminar (operaciones de base de datos)

RIC (Regiones, Identidad y Cultura) CRIE (Culturas Regionales Interactivas de Educación)

RECI (Regiones Educativas de Colombia Interactivas)

CRIS (Culturas Regionales de Colombia en un Solo clic)

RIDE (Regiones, Identidad y Desarrollo Educativo)

CRECE (Culturas Regionales de Colombia para la Educación)

REIN (Regiones Educativas de Colombia Interactivas y Nutridas)

CIRE (Culturas e Identidad Regional Educativa)

CITREC (Culturas e Identidad Regional con Tecnología Educativa de Colombia)

TICREC (Tecnologías de la Información para la Cultura Regional Educativa de Colombia)

API: Interfaz de Programación de Aplicaciones (Application Programming Interface).

DBMS: Sistema de Gestión de Bases de Datos (Database Management System).

SQL: Lenguaje de Consulta Estructurada (Structured Query Language).

HTTP: Protocolo de Transferencia de Hipertexto (Hypertext Transfer Protocol).

REST: Transferencia de Estado Representacional (Representational State Transfer).

JSON: Notación de Objetos de JavaScript (JavaScript Object Notation).

JWT: Token de Web JSON (JSON Web Token).

CRUD: Crear, Leer, Actualizar y Borrar (Create, Read, Update, Delete).

ORM: Mapeo Objeto-Relacional (Object-Relational Mapping).

MVC: Modelo-Vista-Controlador (Model-View-Controller).

API RESTful: API que sigue los principios de REST.

CI/CD: Integración Continua / Entrega Continua (Continuous Integration / Continuous Delivery).  
SaaS: Software como Servicio (Software as a Service).  
SSL/TLS: Capa de sockets seguros/Seguridad de la Capa de Transporte (Secure Sockets Layer/Transport Layer Security).  
HTML: Lenguaje de Marcado de Hipertexto (Hypertext Markup Language).  
CSS: Hojas de Estilo en Cascada (Cascading Style Sheets).  
JS: JavaScript.  
DOM: Modelo de Objeto del Documento (Document Object Model).  
UI: Interfaz de Usuario (User Interface).  
UX: Experiencia del Usuario (User Experience).  
SPA: Aplicación de Página Única (Single Page Application).  
AJAX: Asíncrono JavaScript y XML (Asynchronous JavaScript and XML).  
CMS: Sistema de Gestión de Contenido (Content Management System).  
CDN: Red de Distribución de Contenido (Content Delivery Network).  
SEO: Optimización de Motores de Búsqueda (Search Engine Optimization).  
IDE: Entorno de Desarrollo Integrado (Integrated Development Environment).  
CLI: Interfaz de Línea de Comandos (Command Line Interface).  
PWA: Aplicación Web Progresiva (Progressive Web App).

## **Descripción General**

DARC se desarrollará como una aplicación web accesible desde cualquier dispositivo con conexión a internet. El sitio web estará diseñado con un enfoque en la usabilidad para niños de 7 a 12 años, con el objetivo de ofrecer un recurso académico y didáctico tanto para estudiantes como para docentes.

## Objetivos del Sistema

Educativo: Facilitar el aprendizaje sobre las regiones de Colombia de forma divertida e interactiva.

Motivador: Despertar el interés de los niños por la geografía, cultura e historia de su país.

Accesible: Estar disponible para cualquier estudiante con acceso a internet.

Fácil de usar: Contar con una interfaz intuitiva y amigable para niños.

Recurso didáctico: Proporcionar herramientas y materiales de apoyo para docentes.

## Funcionalidad General

Gestión de Perfiles de Usuario:

- Registrar usuario.
- Almacenar progreso.

Mapa Interactivo:

- Renderizar mapa
- Realizar zoom
- Hacer desplazamiento (panning).
- Resaltar región seleccionada

Información por Región:

- Mostrar información detallada.
- Mostrar imágenes de cada región.
- Reproducir audios de cada región.
- Reproducir videos de cada región.

Juegos Didácticos:

- Mostrar progreso.
- Mostrar sistema de puntos.
- Obtener recompensa (logros).
- Mostrar recompensas (logros).

Actividades Interactivas (Drag and Drop):

- Arrastrar y soltar elementos en el mapa.

- Validar las respuestas.
- Mostrar mensaje resultado.
- Reproducir sonido resultado.

## Usuarios del Sistema

Funcionalidad	Administradores	Docente Investigador	Docente Invitado	Alumno	Invitado
Registrar usuario		X		X	
Almacenar el progreso				X	
Renderizado de mapa		X	X	X	X
Hacer Zoom		X	X	X	X
Hacer desplazamiento		X	X	X	X
Resaltar regiones		X	X	X	X
Mostrar información		X	X	X	X



<b>Mostrar imágenes</b>	<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>
<b>Reproducir audios</b>	<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>
<b>Reproducir videos</b>	<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>
<b>Mostrar progreso.</b>		<b>X</b>		<b>X</b>	
<b>Mostrar sistema de puntos</b>				<b>X</b>	<b>X</b>
<b>Obtener recompensa</b>				<b>X</b>	<b>X</b>
<b>Mostrar recompensas</b>				<b>X</b>	<b>X</b>
<b>Arrastrar y soltar elementos</b>				<b>X</b>	<b>X</b>
<b>Validar respuestas</b>				<b>X</b>	<b>X</b>

<b>Mostrar alerta de resultado</b>				<b>X</b>	<b>X</b>
<b>Reproducir alerta de resultado</b>				<b>X</b>	<b>X</b>

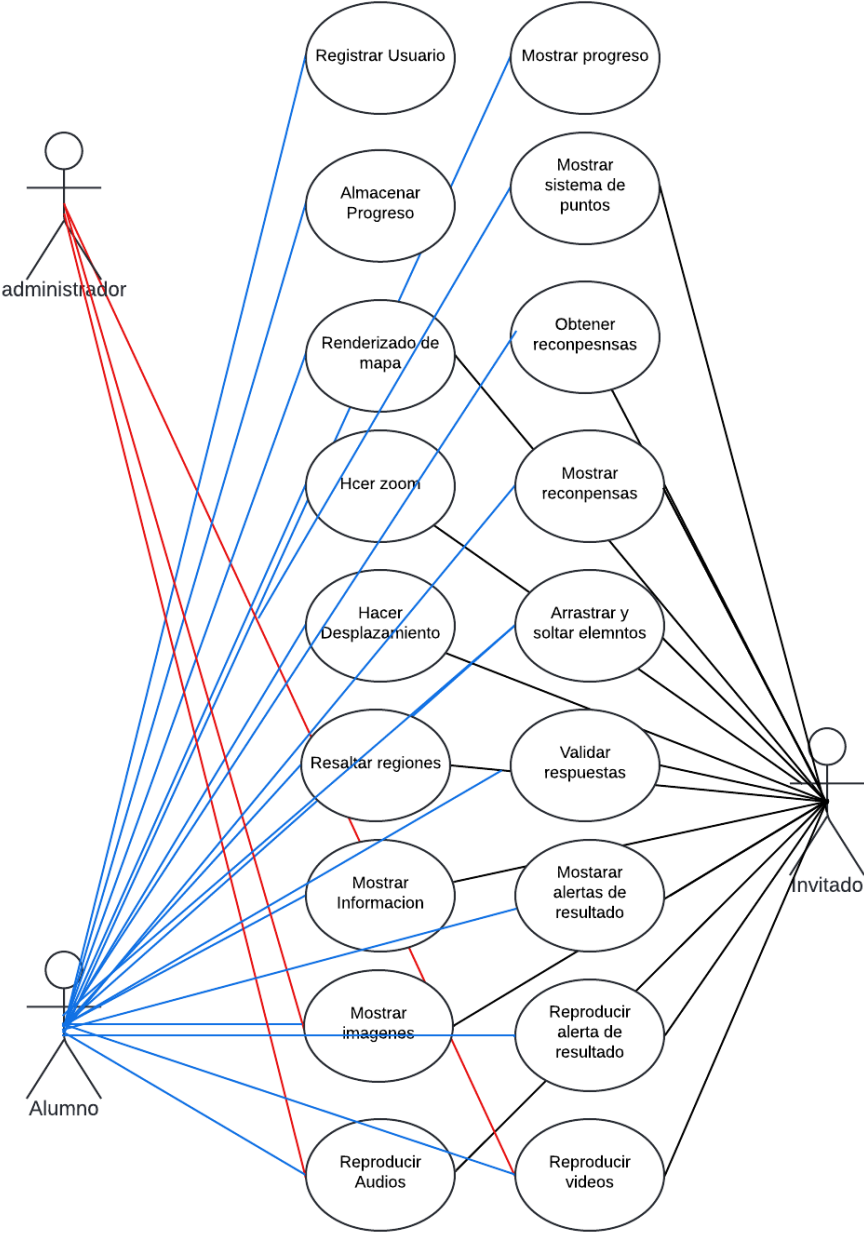
## Restricciones

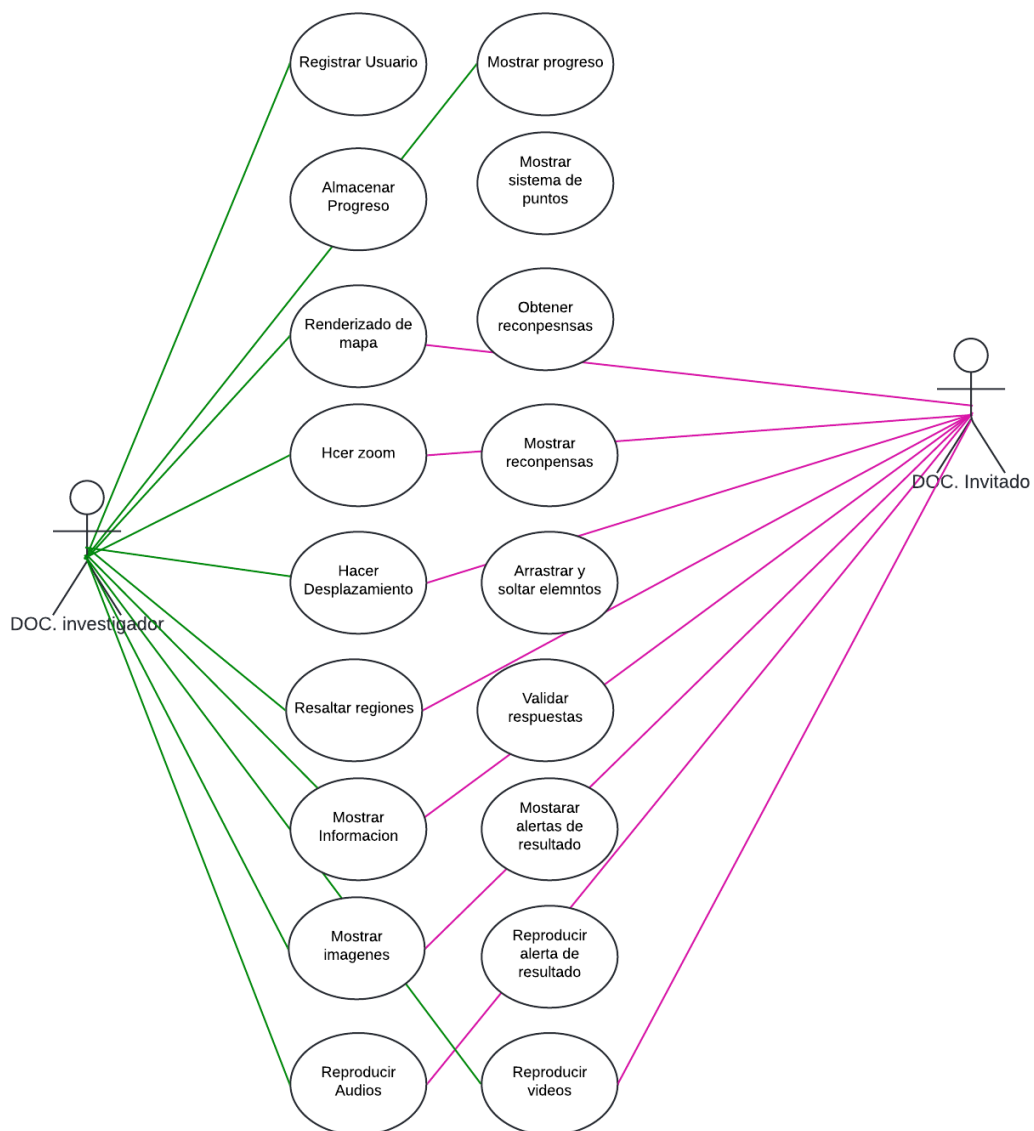
## Requisitos Funcionales

- CU-1: Permitir el registro de usuarios con información básica.
- CU-2: Almacenar el progreso del usuario, incluyendo puntajes, logros y estadísticas.
- CU-3: Mostrar un mapa interactivo de Colombia dividido por regiones.
- CU-4: Implementar funciones de zoom in/out y restablecer zoom en el mapa.
- CU-5: Permitir el desplazamiento (panning) en el mapa.
- CU-6: Resaltar visualmente la región seleccionada en el mapa.
- CU-7: Mostrar información detallada de cada región.
- CU-8: Mostrar imágenes de cada región.
- CU-9: Reproducir audios de cada región.
- CU-10: Reproducir videos de cada región.
- CU-11: Ofrecer juegos didácticos con barras de progreso, sistema de puntos y recompensas.
- CU-12: Implementar actividades interactivas de arrastrar y soltar.
- CU-13: Validar las respuestas del usuario en las actividades.
- CU-14: Proporcionar retroalimentación visual y sonora al usuario.
- CU-15: Permitir a los administradores gestionar usuarios.
- CU-16: Permitir a los administradores gestionar contenido (regiones, información, multimedia).
- CU-17: Permitir a los administradores gestionar juegos.
- CU-18: Permitir a los administradores generar reportes y estadísticas.

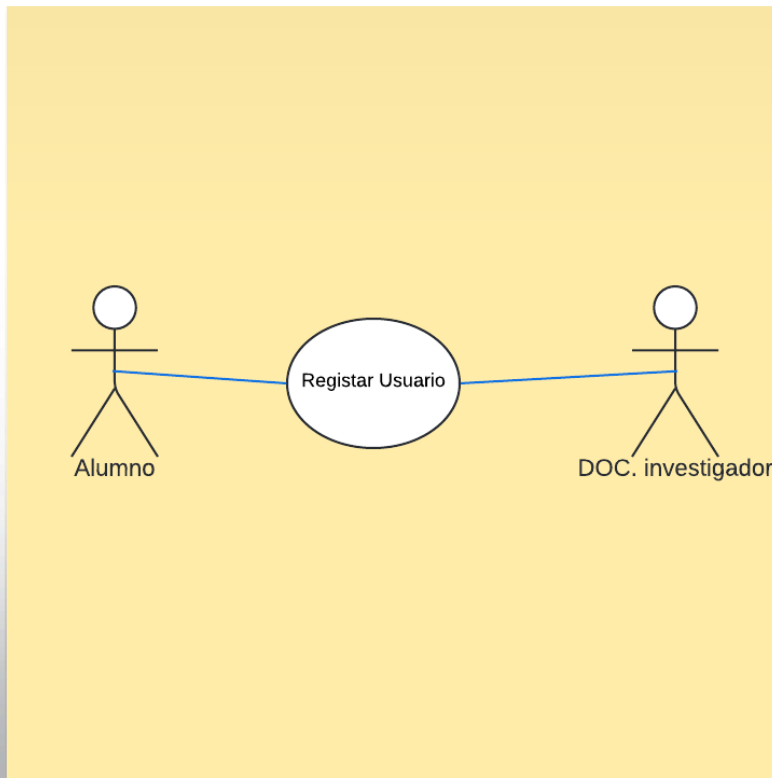
# Mockup de la Interfaz de Usuario

## (UI) Casos de Uso



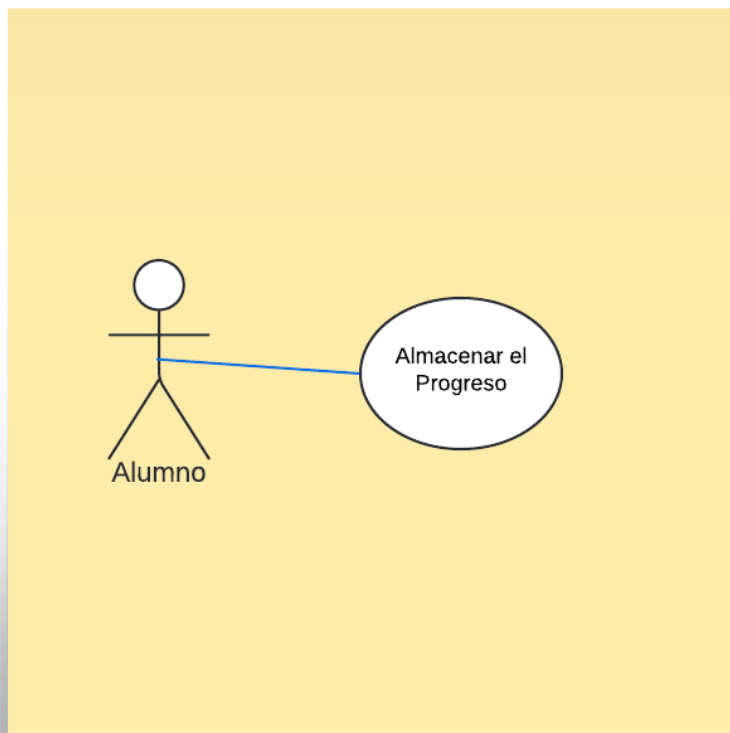


## Descripción detallada de cada caso de uso



## CASO No. 1 Registrar Usuario

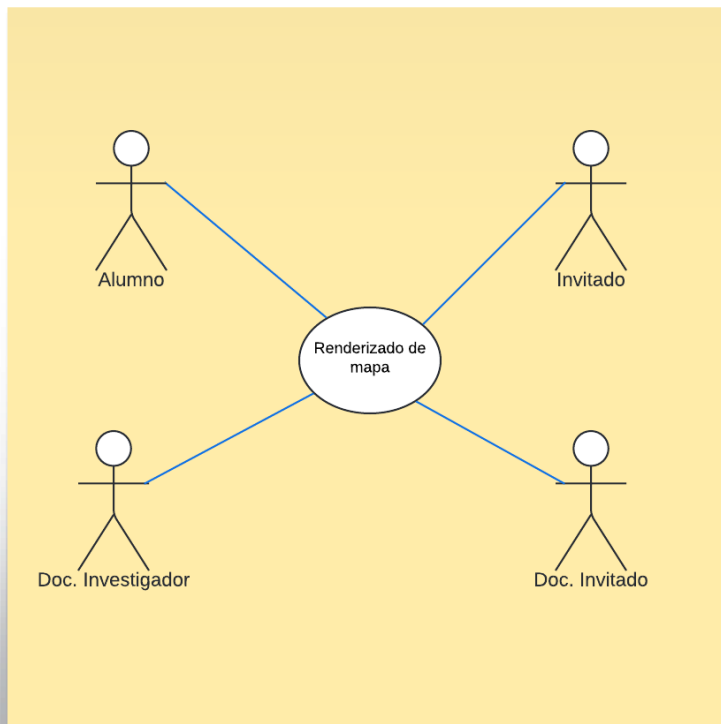
ID:	CU-1	
Nombre	Registrar Usuario	
Actores	Docente investigador, Alumno	
Objetivo	Este caso debe registrar usuario	
Urgencia	5	
Esfuerzo	5	
Pre-condiciones		
Flujo Normal	Actores	Sistema
	Seleccionar botón registrarse	
		Desplegar pestaña registro de usuario
	Ingresar datos	
	Seleccionar botón guardar	
		Guardar datos de usuario
		Retornar mensaje de guardado



---

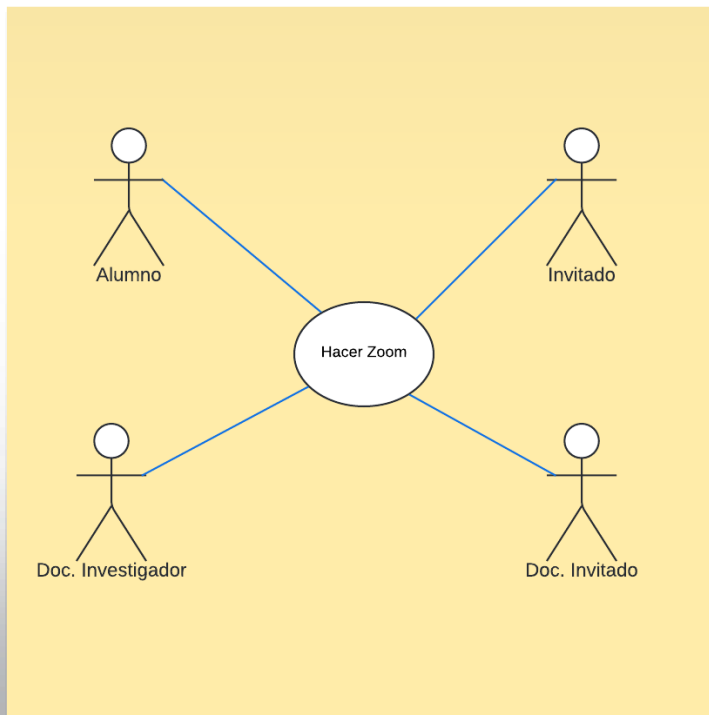
## CASO No. 2 Almacenar Progreso

ID:	CU-2	
Nombre	Almacenar Progreso	
Actores	Docente investigador, Docente invitado, Alumno, Invitado	
Objetivo	Este caso debe registrar usuario	
Urgencia	5	
Esfuerzo	3	
Pre-condiciones		
Flujo Normal	Actores	Sistema
		Actualización automática de progreso



## CASO No. 3 Renderizado De Mapa

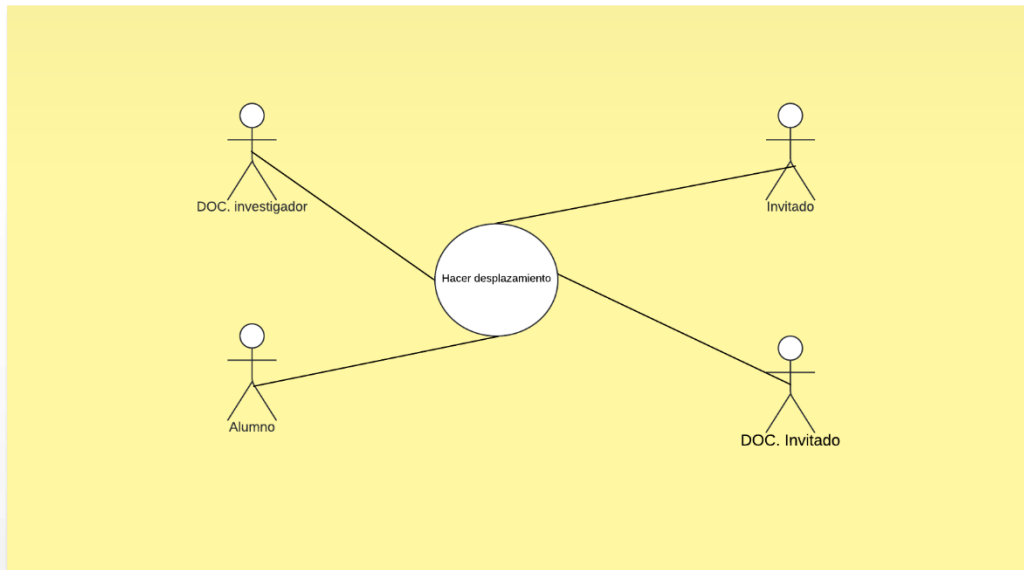
ID:	CU-3	
Nombre	Renderizado de mapa	
Actores	Docente investigador, Docente invitado, Alumno, Invitado	
Objetivo	Este caso debe renderizar mapa	
Urgencia	5	
Esfuerzo	4	
Pre-condiciones		
Flujo Normal	Actores	Sistema
		Renderizado automático del mapa



## CASO No. 4 Hacer Zoom

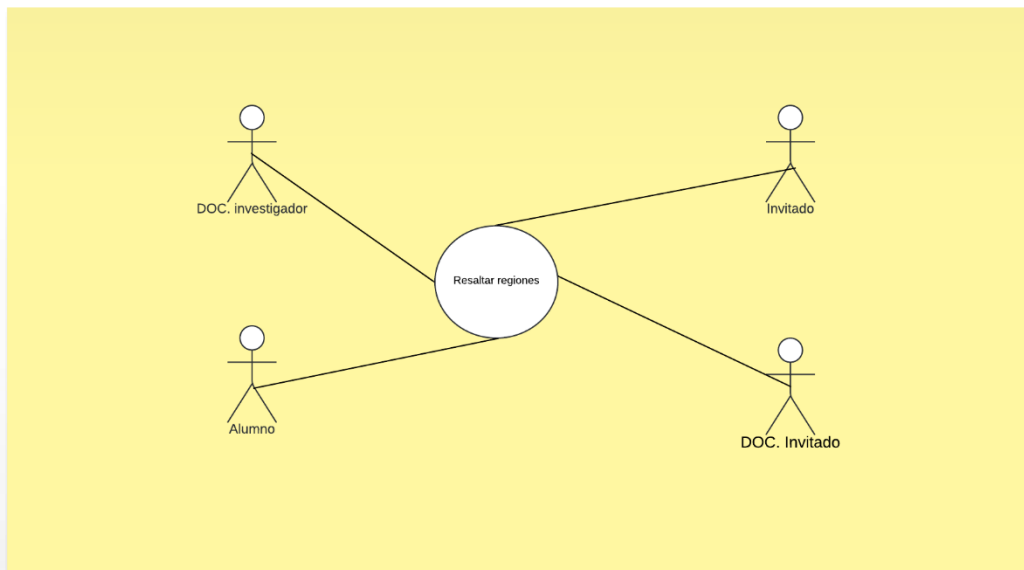
ID:	CU-4	
Nombre	Registrar Usuario	
Actores	Docente investigador, Docente invitado, Alumno, Invitado	
Objetivo	Este caso debe hacer zoom in y zoom out	
Urgencia	2	
Esfuerzo	2	
Pre-condiciones	Renderizado de mapa CU-3	
Flujo Normal	Actores	Sistema
	Hacer Scroll hacia abajo con el click en el mapa	
		Hacer zoom in en el mapa
	Hacer Scroll hacia arriba con el click en el mapa	
		Hacer zoom out en el mapa





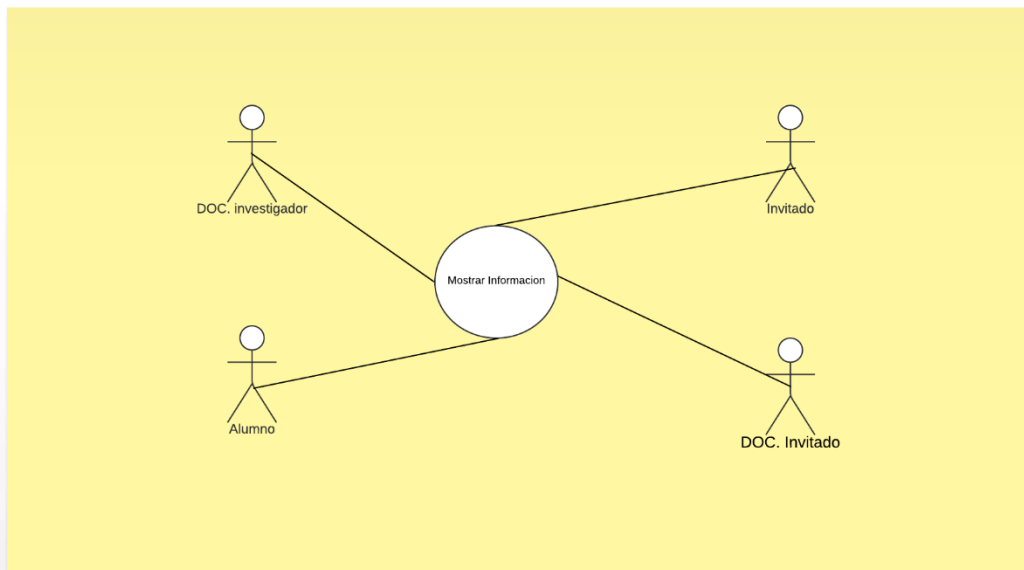
## CASO No. 5 Hacer Desplazamiento

ID:	CU-5	
Nombre	Hacer desplazamiento	
Actores	Docente investigador, Alumno, Docente Invitado, Invitado.	
Objetivo	Este caso debe permitir y detener el desplazamiento por el mapa.	
Urgencia	4	
Esfuerzo	3	
Pre-condiciones	Renderizado de mapa CU-3	
Flujo Normal	Actores	Sistema
	Hacer click sostenido en el mapa y arrastrar para desplazarse.	Realizar desplazamiento por el mapa actualizando información.
	Soltar click sostenido para detener desplazamiento del mapa.	
		Detener el desplazamiento del mapa.



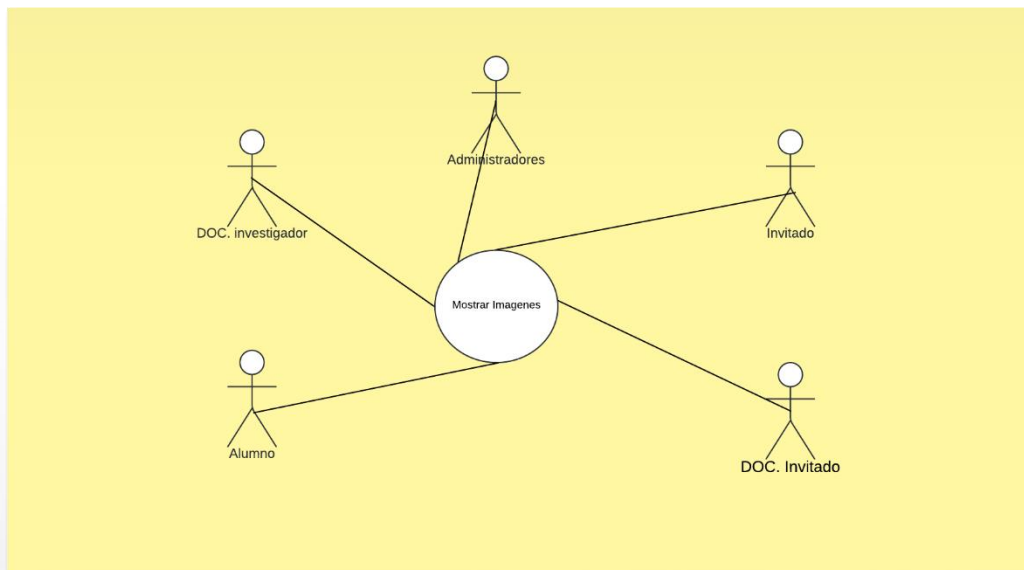
## CASO No. 6 Resaltar Regiones

ID:	CU-6	
Nombre	Resaltar regiones	
Actores	Docente investigador, Alumno, Docente invitado, Invitado.	
Objetivo	Permite resaltar regiones dándoles click.	
Urgencia	4	
Esfuerzo	2	
Pre-condiciones	Renderizado de mapa CU-3	
Flujo Normal	Actores	Sistema
	Hacer click sobre el mapa para resaltar zona	
		Resalta la zona marcada.



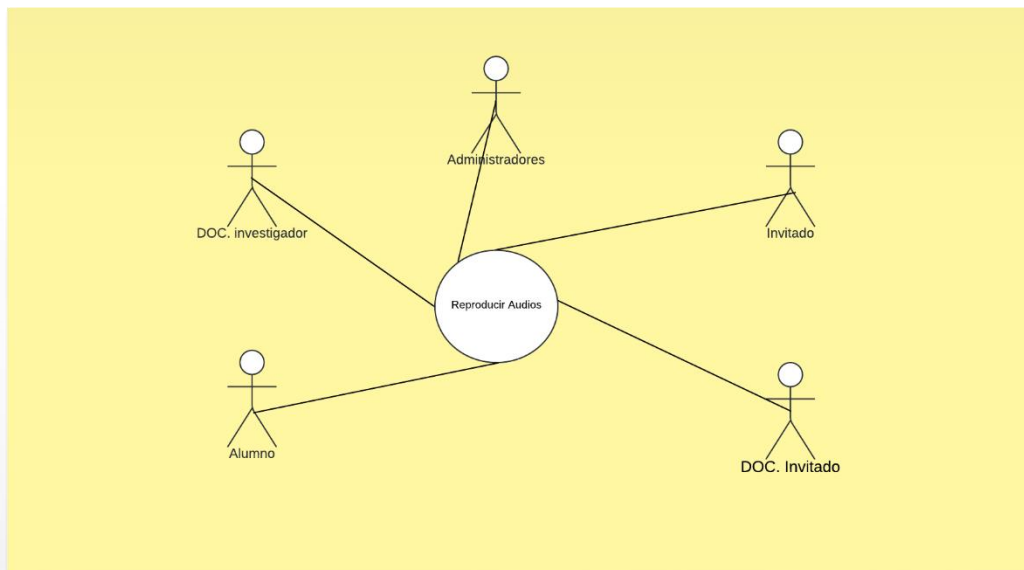
## CASO No. 7 Mostrar Información

ID:	CU-7	
Nombre	Mostrar Información.	
Actores	Docente investigador, Alumno, Docente invitado, Invitado.	
Objetivo	Permite mostrar información de regiones seleccionadas.	
Urgencia	3	
Esfuerzo	5	
Pre-condiciones	Renderizado de mapa CU-3, Guardar progreso CU-2	
Flujo Normal	Actores	Sistema
	Seleccionar zona del mapa	
		Mostrar información correspondiente
Flujo alternativo 1	Hacer click en el botón progreso	
		Mostrar información correspondiente



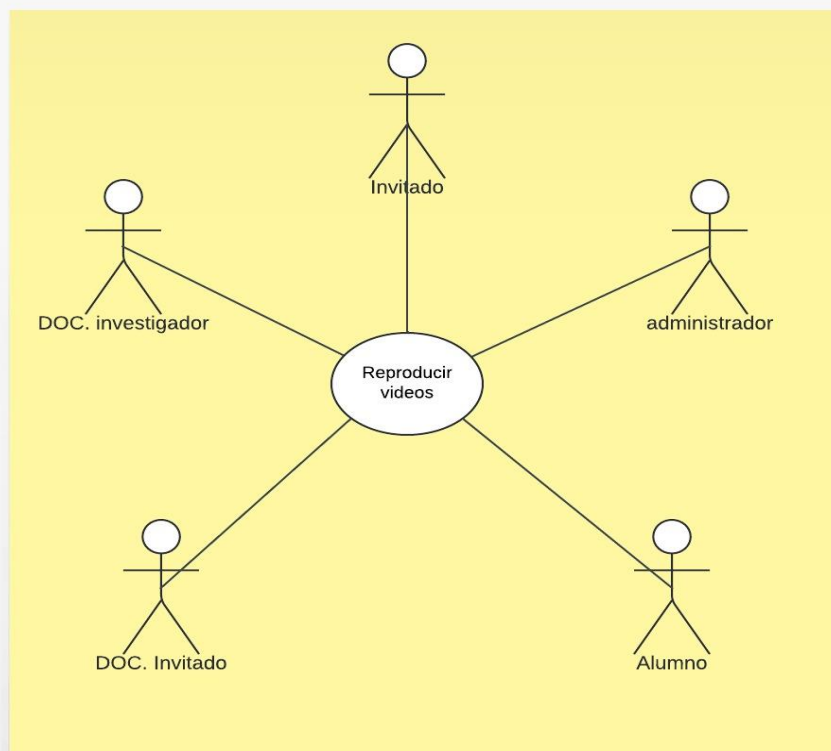
## CASO No. 8 Mostrar Imágenes

ID:	CU-8	
Nombre	Mostrar Imágenes	
Actores	Docente investigador, Alumno, Docente invitado, Invitado, Administradores.	
Objetivo	Permitir mostrar imágenes del mapa.	
Urgencia	2	
Esfuerzo	2	
Pre-condiciones		
Flujo Normal	Actores	Sistema
	Hacer click sobre zona del mapa	
		Mostrar imagen relacionada sobre la zona
	Hacer click fuera del mapa	
		Cierra la imagen abierta



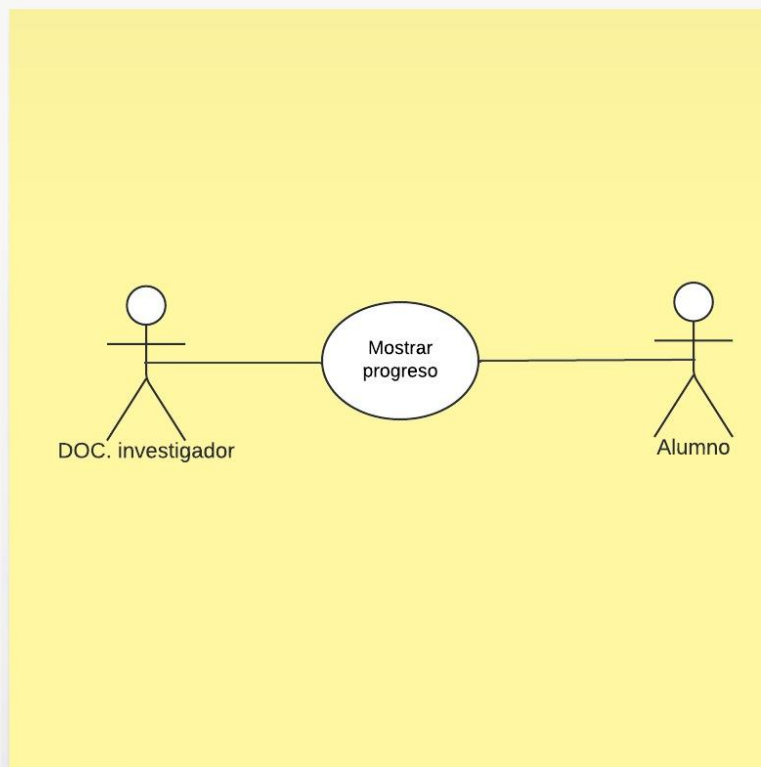
## CASO No. 9 Reproducir Audios

<b>ID:</b>	CU-9	
<b>Nombre</b>	Reproducir Audios.	
<b>Actores</b>	Docente investigador, Alumno, Docente invitado, Invitado, Administradores.	
<b>Objetivo</b>	Permite reproducir sonidos del sistema.	
<b>Urgencia</b>	1	
<b>Esfuerzo</b>	3	
<b>Pre-condiciones</b>		
<b>Flujo Normal</b>	<b>Actores</b>	<b>Sistema</b>
	Hacer click sobre pestaña de objetos	
		Abrir pestaña de objetos
	Hacer click en minijugo Audios	
		Reproducir audio



## CASO No. 10 Reproducción de videos

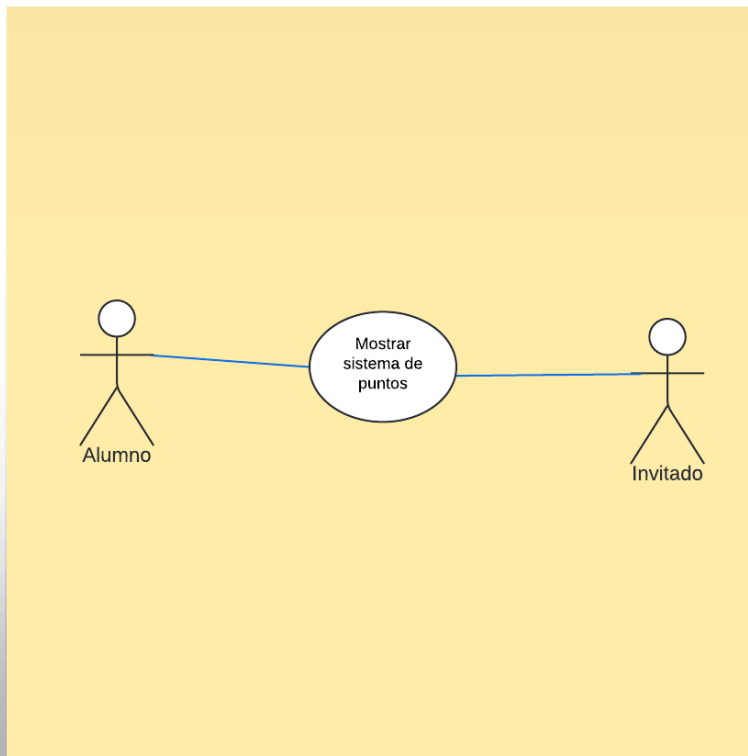
ID:	CU-10	
Nombre	Reproducción de videos	
Actores	Administrador, Docente investigador, Docente invitado, Alumno, Invitados	
Objetivo	Este caso debe permitir reproducir videos	
Urgencia	4	
Esfuerzo	3	
Pre-condiciones		
Flujo Normal	Actores	Sistema
	Hacer click sobre pestaña de objetos	
		Abrir pestaña de objetos
	Hacer click en minijugo Videos	
		Reproducir video



---

## CASO No. 11 Mostrar progreso

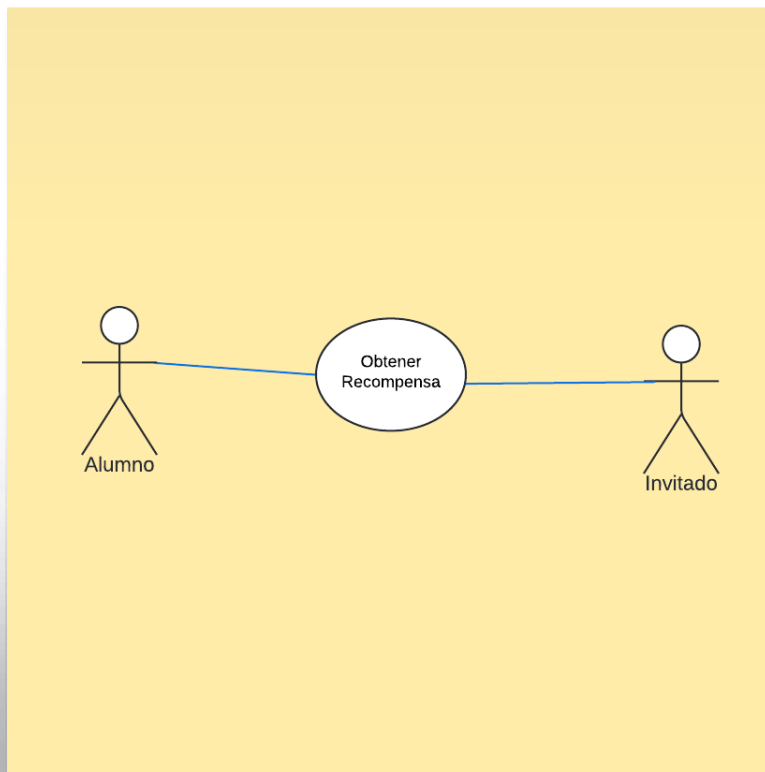
ID:	CU-11	
Nombre	Mostrar progreso	
Actores	Docente investigador, Alumno	
Objetivo	Este caso debe permitir mostrar el progreso	
Urgencia	4	
Esfuerzo	3	
Pre-condiciones	- Necesita del almacenar progreso CU-2	
Flujo Normal	Actores	Sistema
		Muestra el progreso actual



## CASO No. 12 Mostrar sistema de puntos

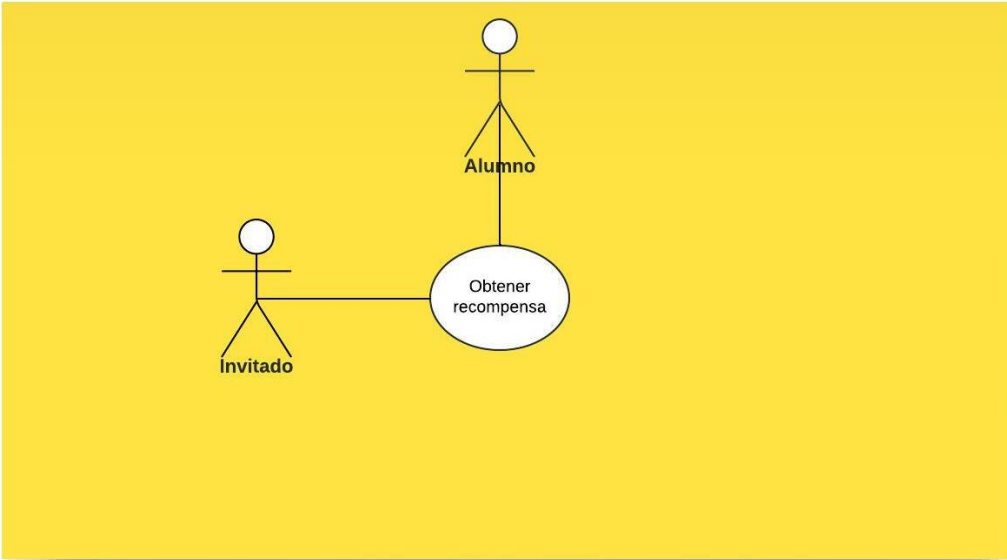
ID:	CU-12	
Nombre	Mostrar sistema de puntos	
Actores	Alumno, Invitado	
Objetivo	Este caso debe permitir mostrar el progreso	
Urgencia	2	
Esfuerzo	2	
Pre-condiciones		
Flujo Normal	Actores	Sistema
	Hacer click en el botón Puntos	
		Muestra el progreso actual





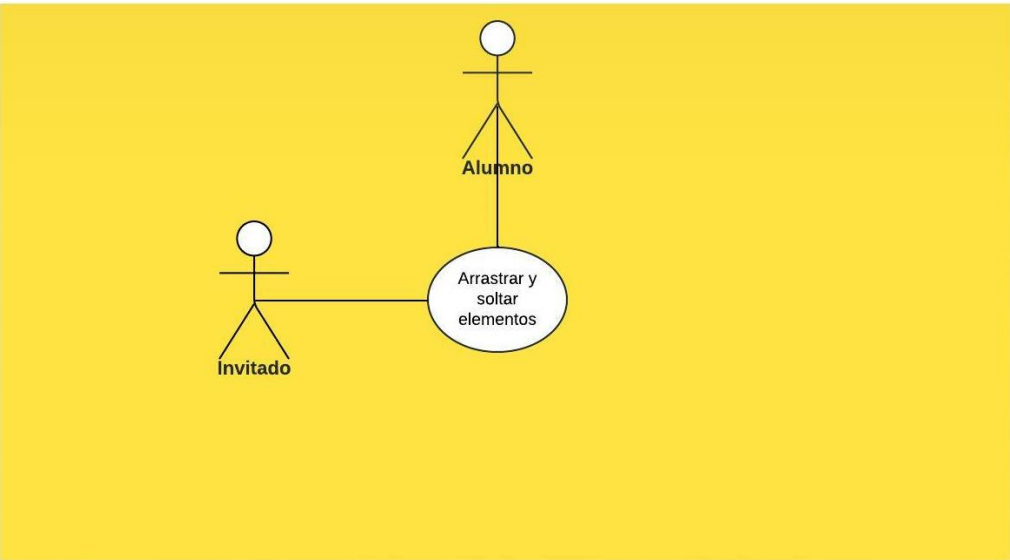
## CASO No. 13 Obtener recompensa

ID:	CU-13	
Nombre	Mostrar progreso	
Actores	Alumno, Invitado	
Objetivo	Este caso debe permitir mostrar el progreso	
Urgencia	3	
Esfuerzo	2	
Pre-condiciones		
Flujo Normal	Actores	Sistema
	Completa minijuego con éxito	
		Actualizar progreso



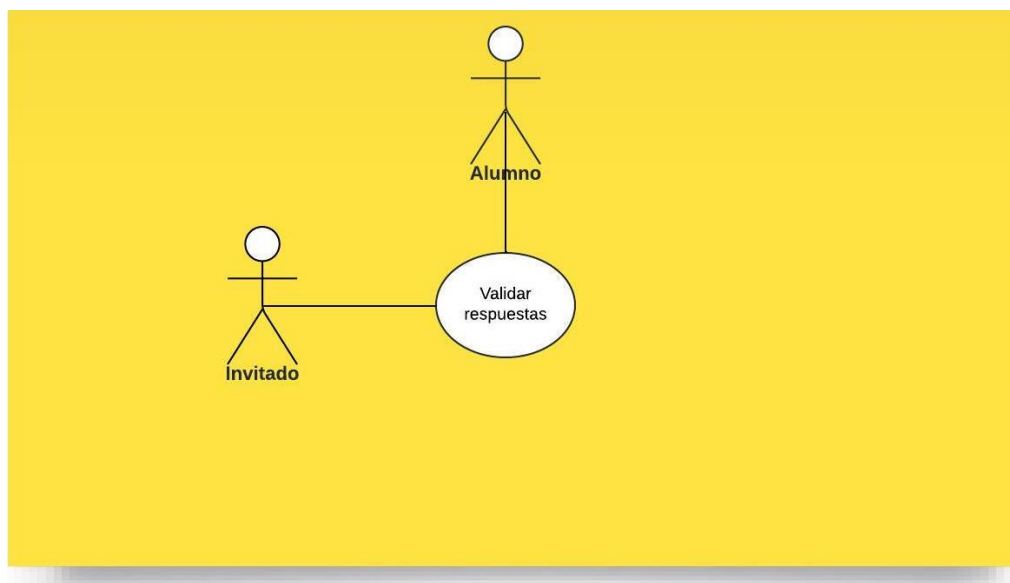
# CASO No. 14 Mostrar Recompensa

ID:	CU-14	
Nombre	Obtener Recompensa	
Actores	Invitado, Alumno	
Objetivo	Este caso el invitado y el alumno deben validar una respuesta	
Urgencia	5	
Esfuerzo	4	
Pre-condiciones		
Flujo Normal	Actores	Sistema
	Completa minijuego con éxito	
		Mostrar recompensa



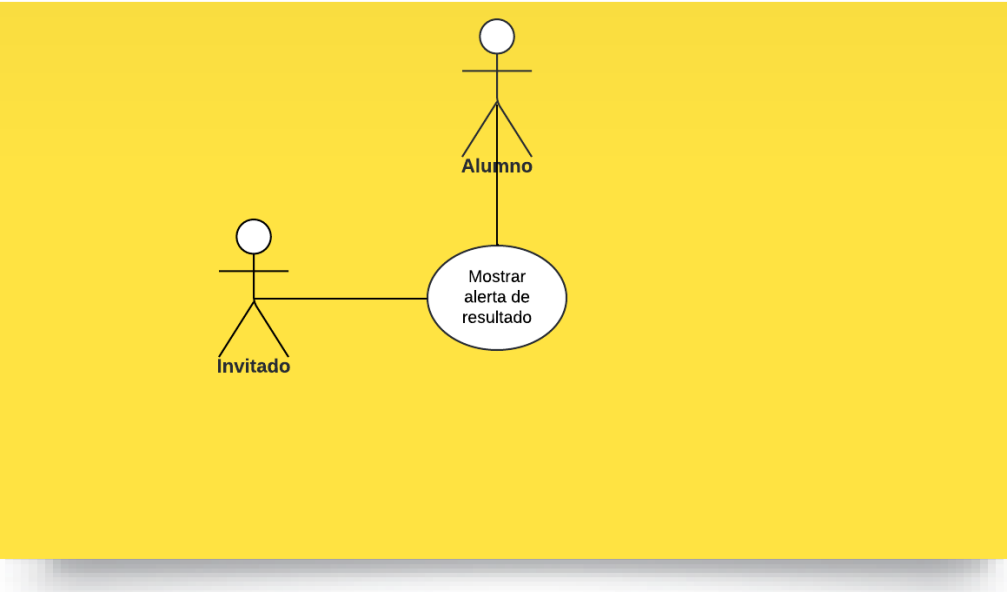
## CASO No. 15 Arrastrar y soltar elementos

ID:	CU-15	
Nombre	Arrastrar y soltar elementos	
Actores	Invitado, Alumno	
Objetivo	Este caso el invitado y el alumno deben arrastrar y soltar elementos	
Urgencia	5	
Esfuerzo	5	
Pre-condiciones	- Deben haber abierto la página web y darle a jugar	
Flujo Normal	Actores	Sistema
	Click sostenido sobre objeto	
		Muestra objeto sobre el click
	Suelta click sobre objeto	



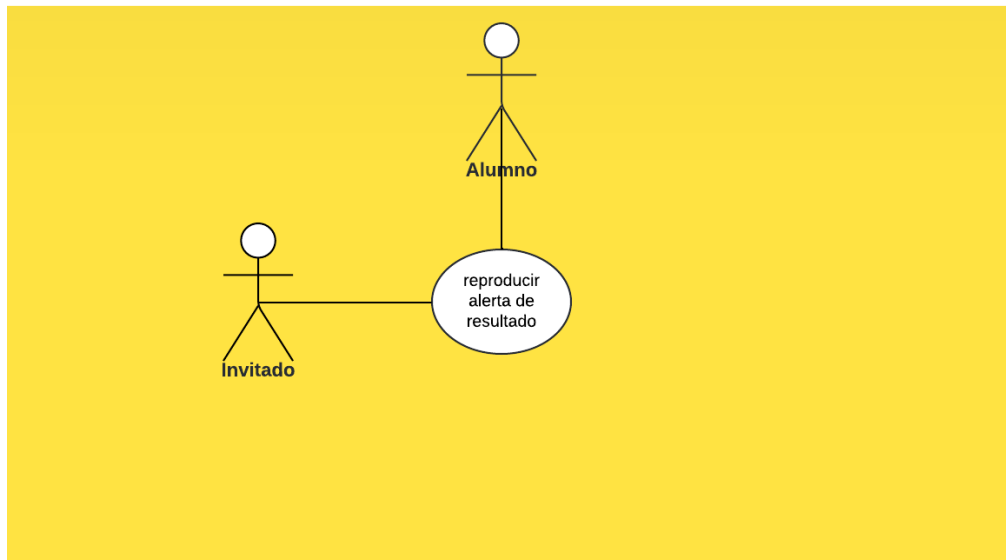
## CASO No. 16 Validar respuesta

ID:	CU-16	
Nombre	Validar respuesta	
Actores	Invitado, Alumno	
Objetivo	Este caso el invitado y el alumno deben arrastrar y soltar elementos	
Urgencia	5	
Esfuerzo	5	
Pre-condiciones		
Flujo Normal	Actores	Sistema
	Resolver minijuego	
		Validación de respuesta



# CASO No. 17 Mostar alerta de resultado

ID:	CU-17	
Nombre	Mostar alerta de resultado	
Actores	Invitado, Alumno	
Objetivo	Este caso el invitado y el alumno deberán ver una alerta con el resultado de su respuesta	
Urgencia	4	
Esfuerzo	3	
Pre-condiciones	Validar respuesta CU-16	
Flujo Normal	Actores	Sistema
	Resolver minijuego	
		Mostrar alerta de resultado



## CASO No. 18 Reproducir alerta de resultado

ID:	CU-18	
Nombre	Reproducir alerta de resultado	
Actores	Invitado, Alumno	
Objetivo	Este caso el invitado y el alumno deberán escuchar la alerta de resultado	
Urgencia	4	
Esfuerzo	3	
Pre-condiciones	Validar respuesta CU-16	
Flujo Normal	Actores	Sistema
	Resolver minijuego	

### Diagramas de Flujo de Casos de Uso

### Prioridad de Requisitos

<b>Urgencia</b>						
<b>Impacto</b>		<b>1-Baja</b>	<b>2-Menor</b>	<b>3-Moderada</b>	<b>4-Alta</b>	<b>5-Obligatoria</b>
	<b>5-Muy alto</b>	<b>5</b>	<b>10</b>	<b>15</b>	<b>20</b>	<b>25</b>
				<b>CU-7</b>		<b>CU-1 CU-15 CU-16</b>
	<b>4-Alto</b>	<b>4</b>	<b>8</b>	<b>12</b>	<b>16</b>	<b>20</b>
			<b>CU-8</b>			<b>CU-3 CU-14</b>
	<b>3-Medio</b>	<b>3</b>	<b>6</b>	<b>9</b>	<b>12</b>	<b>15</b>
		<b>CU-9</b>			<b>CU-5 CU-10 CU-11 CU-17 CU-18</b>	<b>CU-2</b>
	<b>2-Bajo</b>	<b>2</b>	<b>4</b>	<b>6</b>	<b>8</b>	<b>10</b>
			<b>CU-4 CU-12</b>	<b>CU-13</b>	<b>CU-6</b>	
	<b>1-Muy bajo</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>

## Requisitos No Funcionales

### Seguridad:

- Todos los datos sensibles, como información personal de los usuarios y credenciales de acceso, deben ser cifrados utilizando algoritmos de cifrado.

### Rendimiento:

- La página web debe ofrecer un rendimiento óptimo, permitiendo la colaboración en tiempo real incluso con un gran número de usuarios, El sistema debe garantizar que el tiempo de respuesta para las solicitudes de los usuarios no exceda los 2 segundos en condiciones normales de carga.

### Escalabilidad:

- La página web debe ser escalable para manejar un aumento en el número de usuarios y la cantidad de contenido, Los requisitos de estabilidad incluyen un manejo efectivo de errores,

recuperación rápida ante fallos, consistencia de datos y pruebas de estrés para asegurar un rendimiento estable.

**Disponibilidad:**

- La página web debe estar disponible y funcionando de manera constante, minimizando el tiempo de inactividad.

**Compatibilidad con Dispositivos:**

- La página web debe ser compatible con una variedad de dispositivos, incluyendo computadoras de escritorio, tabletas y teléfonos móviles.

**Usabilidad:**

- La interfaz de usuario de la página web debe ser intuitiva y fácil de usar para usuarios de diferentes niveles de habilidad, debe permitir que los nuevos usuarios completen tareas básicas, como registrarse y navegar por el sistema, en menos de 5 minutos. Además, se debe incluir un sistema de ayuda accesible que ofrezca tutoriales y respuestas a preguntas frecuentes, mejorando así la experiencia del usuario y reduciendo la curva de aprendizaje.

**Accesibilidad:**

- La página web debe ser accesible para personas con discapacidades, cumpliendo con estándares de accesibilidad web.

**Cumplimiento Normativo:**

- La página web debe cumplir con regulaciones y normativas de privacidad y seguridad de datos.

**Tiempo de Respuesta:**

- La página web debe tener tiempos de respuesta rápidos para mantener una experiencia de usuario fluida.

## Requisitos de Desempeño

- **Rendimiento en Tiempo Real:** Capacidad de renderizar en tiempo real, interacción y reacción manera fluida en múltiples dispositivos a la vez.
- **Tiempo de Carga Rápido:** Dar prioridad a los recursos que permita entrar de forma rápida, cargando primero así el mapa general renderizado y el resto de los elementos a medida que se requieran.
- **Optimización de Recursos:** Implementar imágenes y videos en alta definición usando formatos audiovisuales compactos, permitiendo así ejecutarse en equipos de gama baja.

## Requisitos de Seguridad

- Se implementará un sistema de autenticación segura para proteger el acceso a la plataforma.
- Los datos sensibles de los usuarios se almacenarán de forma encriptada.
- Se implementarán medidas de seguridad para prevenir ataques de inyección SQL y cross-site scripting (XSS).

## Requisitos de Usabilidad

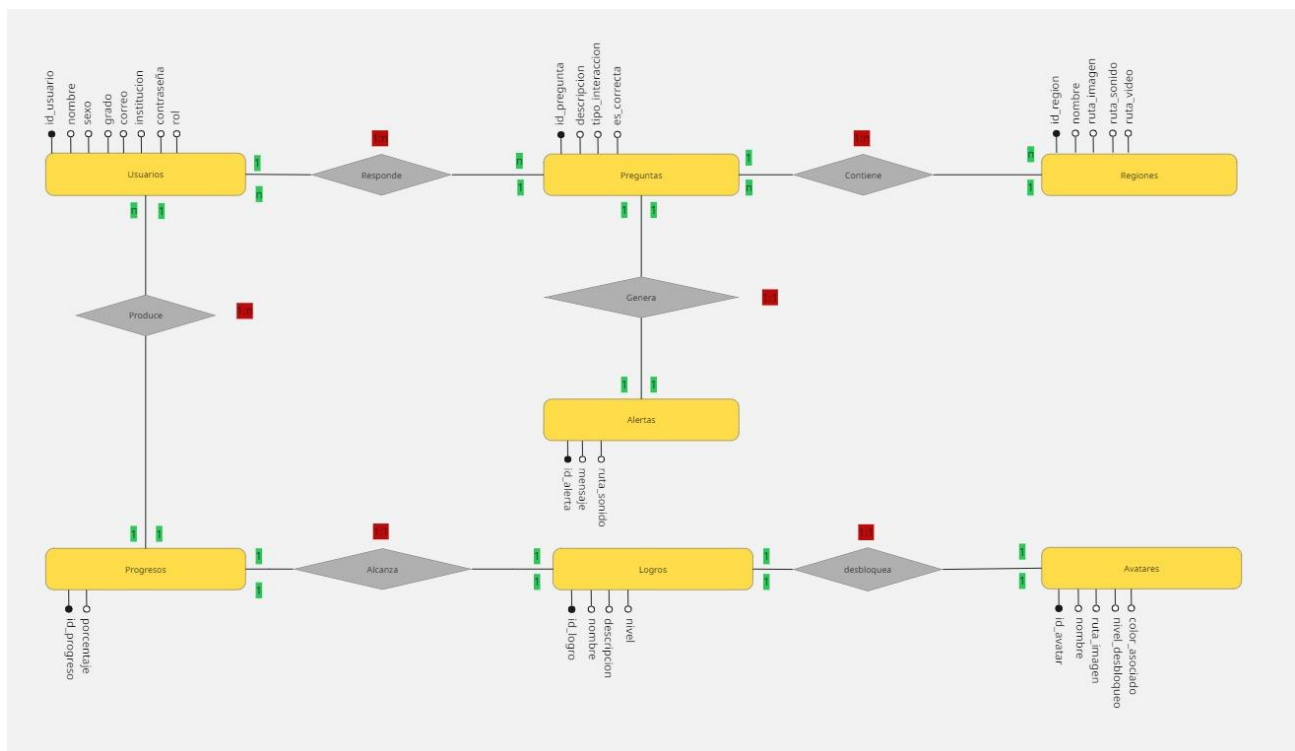
- **Interfaz Intuitiva:** Interfaz de usuario intuitiva y fácil de usar para los estudiantes, siendo así colorida con imágenes, audio, video, de modo que se les haga una experiencia amigable.
- **Compatibilidad con Dispositivos:** La plataforma debe ser capaz de ejecutarse en todo tipo de computadores y navegadores web, incluyendo equipos de gama baja, permitiendo su ejecución en la mayor cantidad de dispositivos.
- **Documentación y Ayuda en Línea:** Proporcionar ayuda, documentación y tutoriales en línea dentro de la plataforma, que sean intuitivos y les guíen a los estudiantes usar la plataforma.

## Requisitos de Escalabilidad

- La arquitectura de la aplicación debe ser escalable para poder manejar un número creciente de usuarios y datos.
- Se utilizará una base de datos que permita la escalabilidad horizontal.

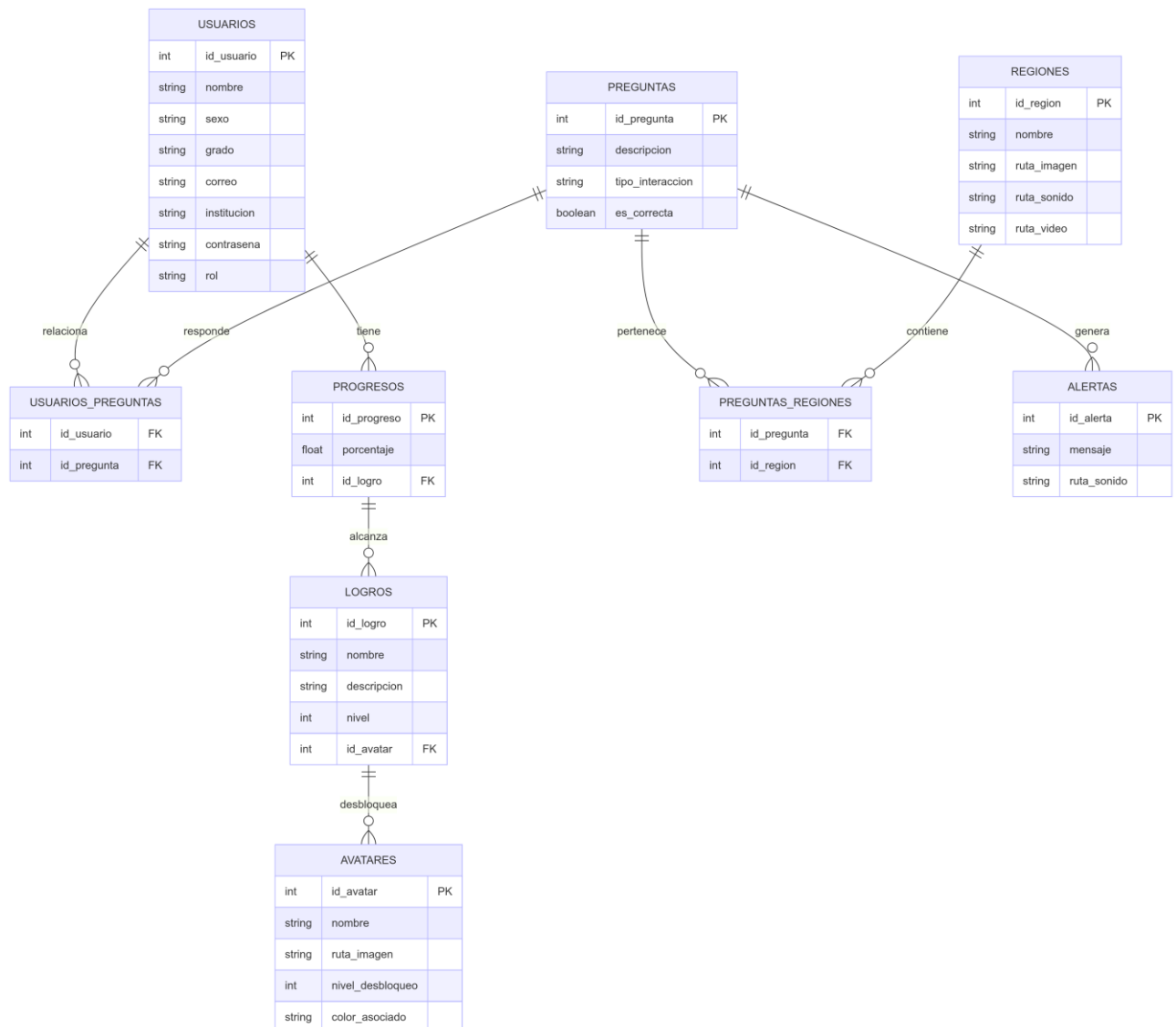
## Modelado E/R

### Diagrama de Entidad-Relación





## Diagrama relacional



## Descripción de Entidades y Relaciones

**Usuario:** Representa a un usuario de la aplicación, identificado por ID\_Usuario. Almacena información como NombreUsuario, Contraseña, TipoUsuario (que define su rol), y opcionalmente su Email e ID\_Region (estableciendo una relación con la entidad Región).

**Región:** Representa una región de Colombia, con ID\_Region como identificador. Contiene información como Nombre, Descripción, Imagen, Audio y Video. Se relaciona con Usuario (un usuario pertenece a una región) y con Actividad (una región contiene múltiples actividades).

**Juego:** Representa un juego didáctico, identificado por ID\_Juego. Incluye Nombre, Descripción y PuntajeMaximo. Se relaciona con Usuario a través de la entidad Usuario\_Juego, indicando qué usuarios han jugado qué juegos y sus puntajes. También se relaciona con Logro (un juego permite obtener logros).

**Actividad:** Representa una actividad de arrastrar y soltar, con ID\_Actividad como identificador. Contiene una Descripción, ID\_Region (indicando a qué región pertenece) y RespuestaCorrecta. Se

relaciona con ElementoArrastrar (una actividad tiene varios elementos para arrastrar).

**ElementoArrastrar:** Representa un elemento individual dentro de una actividad de arrastrar y soltar, identificado por ID\_Elemento. Incluye ID\_Actividad (para asociarlo a la actividad correcta), Contenido (el contenido del elemento, como texto o URL de imagen) y PosicionCorrecta.

**Usuario\_Juego:** Es una entidad asociativa que representa la relación muchos a muchos entre Usuario y Juego. Almacena el ID\_Usuario, ID\_Juego, el Puntaje obtenido por el usuario en ese juego y la Fecha en que se jugó.

**Usuario\_Logro:** Es una entidad asociativa que representa la relación muchos a muchos entre Usuario y Logro. Registra el ID\_Usuario, ID\_Logro y la FechaObtencion del logro.  
**Logro:** Representa un logro que un usuario puede obtener, identificado por ID\_Logro. Incluye Nombre y Descripción.

## Reglas de Integridad

Regla	Entidad(es) Afectada(s)	Descripción	Implementación en DBMS
Integridad Referencial			
RI-1	Usuario, Región	'ID_Región' en Usuario debe referenciar un 'ID_Región' válido en Región. Permite NULL.	'ALTER TABLE Usuario ADD CONSTRAINT FK_Usuario_Región FOREIGN KEY (ID_Región) REFERENCES Región(ID_Región);'
RI-2	Actividad, Región	'ID_Región' en Actividad debe referenciar un 'ID_Región' válido en Región.	'ALTER TABLE Actividad ADD CONSTRAINT FK_Actividad_Región FOREIGN KEY (ID_Región) REFERENCES Región(ID_Región);'
RI-3	Usuario_Juego, Usuario	'ID_Usuario' en Usuario_Juego debe referenciar un 'ID_Usuario' válido en Usuario.	'ALTER TABLE Usuario_Juego ADD CONSTRAINT FK_UsuarioJuego_Usuario FOREIGN KEY (ID_Usuario) REFERENCES Usuario(ID_Usuario);'
RI-4	Usuario_Juego, Juego	'ID_Juego' en Usuario_Juego debe referenciar un 'ID_Juego' válido en Juego.	'ALTER TABLE Usuario_Juego ADD CONSTRAINT FK_UsuarioJuego_Juego FOREIGN KEY (ID_Juego) REFERENCES Juego(ID_Juego);'
RI-5	ElementoArrastrar, Actividad	'ID_Actividad' en ElementoArrastrar debe referenciar un 'ID_Actividad' válido en Actividad.	'ALTER TABLE ElementoArrastrar ADD CONSTRAINT FK_ElementoArrastrar_Actividad FOREIGN KEY (ID_Actividad) REFERENCES Actividad(ID_Actividad);'
RI-6	Usuario_Logro, Usuario	'ID_Usuario' en Usuario_Logro debe referenciar un 'ID_Usuario' válido en Usuario.	'ALTER TABLE Usuario_Logro ADD CONSTRAINT FK_UsuarioLogro_Usuario FOREIGN KEY (ID_Usuario) REFERENCES Usuario(ID_Usuario);'
RI-7	Usuario_Logro, Logro	'ID_Logro' en Usuario_Logro debe referenciar un 'ID_Logro' válido en Logro.	'ALTER TABLE Usuario_Logro ADD CONSTRAINT FK_UsuarioLogro_Logro FOREIGN KEY (ID_Logro) REFERENCES Logro(ID_Logro);'
Integridad de Entidad			
IE-1	Usuario	'ID_Usuario' debe ser clave primaria única, no	'ALTER TABLE Usuario ADD CONSTRAINT PK_Usuario PRIMARY

		nula.	KEY (ID_Usuario);`
IE-2	Región	`ID_Región` debe ser clave primaria única, no nula.	`ALTER TABLE Region ADD CONSTRAINT PK_Región PRIMARY KEY (ID_Región);`
IE-3	Juego	`ID_Juego` debe ser clave primaria única, no nula.	`ALTER TABLE Juego ADD CONSTRAINT PK_Juego PRIMARY KEY (ID_Juego);`
IE-4	Actividad	`ID_Actividad` debe ser clave primaria única, no nula.	`ALTER TABLE Actividad ADD CONSTRAINT PK_Actividad PRIMARY KEY (ID_Actividad);`
IE-5	ElementoArrastrar	`ID_Elemento` debe ser clave primaria única, no nula.	`ALTER TABLE ElementoArrastrar ADD CONSTRAINT PK_ElementoArrastrar PRIMARY KEY (ID_Elemento);`
IE-6	Usuario_Juego	`(ID_Usuario, ID_Juego)` debe ser clave primaria única, no nula.	`ALTER TABLE Usuario_Juego ADD CONSTRAINT PK_Usuario_Juego PRIMARY KEY (ID_Usuario, ID_Juego);`
IE-7	Usuario_Logro	`(ID_Usuario, ID_Logro)` debe ser clave primaria única, no nula.	`ALTER TABLE Usuario_Logro ADD CONSTRAINT PK_Usuario_Logro PRIMARY KEY (ID_Usuario, ID_Logro);`
IE-8	Logro	`ID_Logro` debe ser clave primaria única, no nula.	`ALTER TABLE Logro ADD CONSTRAINT PK_Logro PRIMARY KEY (ID_Logro);`
Integridad de Dominio			
	Restricciones sobre tipos de datos, rangos y longitudes de los campos.	(Depende del DBMS, se usarían comandos CHECK, constraints de tipo de datos, etc.)	
Integridad de Concurrencia	Todas las tablas	Mecanismos para manejar la concurrencia, ej: transacciones, bloqueo de filas.	(Depende del DBMS, se usarían comandos BEGIN TRANSACTION, COMMIT, ROLLBACK, etc.)

## Referencias

1. Cookieclicker. (n.d.). *Cookieclicker*. <https://cookieclicker.com/> La progresión basada en recompensas de *Cookieclicker* se analiza como un modelo para la implementación de un sistema de gamificación en DARC. Su sistema de feedback visual y auditivo, así como la curva de aprendizaje gradual, son ejemplos de buenas prácticas para el diseño de experiencias de usuario (UX) en entornos educativos.
2. Didactalia. (n.d.). *Mapas interactivos*. <https://mapasinteractivos.didactalia.net/comunidad/mapasflashinteractivos/MapasDidactalia> La arquitectura de la interfaz de usuario (UI) del mapa interactivo de *Didactalia* se estudia como un ejemplo de buenas prácticas para la implementación de mapas cartográficos interactivos con información contextual. Se analizan particularmente sus mecanismos de zoom, panning, y la presentación de datos geoespaciales en formato accesible.
3. Educaplay. (n.d.). *Bogotá*. <https://es.educaplay.com/recursos-educativos/6656215-bogota.html> Las actividades interactivas de *Educaplay* se toman como referencia para el desarrollo de micro-aprendizajes y actividades de tipo drag-and-drop. La implementación de elementos de feedback, la validación de respuestas y la gestión del progreso del usuario se consideran puntos importantes para el diseño de las actividades de DARC.
4. GeoGuessr. (n.d.). *GeoGuessr*. <https://www.geoguessr.com/> La arquitectura de backend y frontend de *GeoGuessr* se analiza para la implementación de un sistema de

mapas interactivos en DARC. Se toma como referencia su sistema de gestión de datos geoespaciales, la integración de imágenes de alta resolución y la eficiencia del renderizado cartográfico para diferentes niveles de zoom.

5. Poki. (n.d.). *¿Dónde está tu país?*. <https://poki.com/es/g/wheres-your-country> La funcionalidad de geolocalización y las mecánicas de juego de "¿Dónde está tu país?" se consideran como ejemplos para el diseño de la interacción usuario-mapa en DARC. El análisis se centra en la simplificación de la interfaz, la intuitividad de la mecánica de juego y la integración de elementos de gamificación para el aprendizaje.
6. World Geography Games. (n.d.). *Juegos de geografía mundial*. <https://world-geography-games.com/es/index.html> Se estudian diversos juegos de *World Geography Games* para el diseño de la arquitectura de los minijuegos de DARC. El análisis se enfoca en las diferentes mecánicas de juego, las estrategias de feedback, el diseño de las interfaces de usuario y la implementación de la lógica de juego para garantizar la usabilidad y una experiencia de aprendizaje efectiva.

# **Etapla 2: Persistencia de Datos con Backend**

## **Introducción**

### **Propósito de la Etapa**

### **Alcance de la Etapa**

### **Definiciones y Acrónimos**

## **Diseño de la Arquitectura de Backend**

### **Descripción de la Arquitectura Propuesta**

### **Componentes del Backend**

### **Diagramas de Arquitectura**

## **Elección de la Base de Datos**

### **Evaluación de Opciones (SQL o NoSQL)**

### **Justificación de la Elección**

**Diseño de Esquema de Base de Datos**

**Implementación del Backend**

**Elección del Lenguaje de Programación**

**Creación de la Lógica de Negocio**

**Desarrollo de Endpoints y APIs**

**Autenticación y Autorización**

**Conexión a la Base de Datos**

**Configuración de la Conexión**

**Desarrollo de Operaciones CRUD**

**Manejo de Transacciones**

**Pruebas del Backend**

**Diseño de Casos de Prueba**

## **Ejecución de Pruebas Unitarias y de Integración**

## **Manejo de Errores y Excepciones**

## **Etapas de Desarrollo**

### **Etapas de Desarrollo**

#### **Frontend Introducción**

##### **Propósito de la Etapa**

##### **Alcance de la Etapa**

##### **Definiciones y Acrónimos**

#### **Creación de la Interfaz de Usuario (UI)**

##### **Diseño de la Interfaz de Usuario (UI) con HTML y CSS**

##### **Consideraciones de Usabilidad**

##### **Maquetación Responsiva**

#### **Programación Frontend con JavaScript (JS)**

##### **Desarrollo de la Lógica del Frontend**

##### **Manejo de Eventos y Comportamientos Dinámicos**



**Uso de Bibliotecas y Frameworks (si aplicable)**

**Consumo de Datos desde el Backend**

**Configuración de Conexiones al Backend**

**Obtención y Presentación de Datos**

**Actualización en Tiempo Real (si aplicable)**

**Interacción Usuario-Interfaz**

**Manejo de Formularios y Validación de Datos**

**Implementación de Funcionalidades Interactivas**

**Mejoras en la Experiencia del Usuario**

**Pruebas y Depuración del Frontend**

**Diseño de Casos de Prueba de Frontend**

**Pruebas de Usabilidad**

**Depuración de Errores y Optimización del Código**

**Implementación de la Lógica de Negocio en el Frontend**

**Migración de la Lógica de Negocio desde el Backend (si necesario)**

**Validación de Datos y Reglas de Negocio en el Frontend**

**Integración con el Backend**

**Verificación de la Comunicación Efectiva con el Backend**

**Pruebas de Integración Frontend-Backend**