



Entrega de Avance del Reto. Ciclo 1

“Mate mágicos”

Equipo 6

Sebastián Elorduy Martínez Manzanero A01655930

Jorge Rodrigo Colín Rubio A01662960

Miguel Sebastián Reyes Moguel A01663071

Diego Samperio Arce A01662935

PROFESORES:

Sergio Ruiz Loza

Christian Rolando Vallejo Villamil

José Ángel Martínez Navarro

Cuarto Semestre

Construcción de software y toma de decisiones (TC2005B.441)

ENTREGA:

19 de abril del 2024

Índice

| | |
|--|---------------------------|
| Entrega de Avance del Reto. Ciclo 1..... | 1 |
| Índice..... | 2 |
| Introducción..... | 3 |
| Módulo 1..... | 4 |
| Modelo relacional de la base de datos versión beta en dbdiagram.io..... | 4 |
| Módulo 2..... | 5 |
| Licencia..... | 5 |
| ● <u>Lista de requerimientos funcionales.....</u> | <u>5</u> |
| ○ <u>Tabla 1.1 Requerimientos funcionales webapp.....</u> | <u>5</u> |
| ○ <u>Tabla 1.2 Requerimientos funcionales juego.....</u> | <u>5</u> |
| ● <u>Lista de requerimientos no funcionales.....</u> | <u>7</u> |
| ○ <u>Tabla 2.1 Requerimientos no funcionales webapp.....</u> | <u>7</u> |
| ○ <u>Tabla 2.2 Requerimientos no funcionales juego.....</u> | <u>7</u> |
| UML..... | 9 |
| ○ <u>Casos de uso juego.....</u> | <u>9</u> |
| Diagramas..... | 18 |
| Diagrama de componentes..... | 18 |
| Diagrama de clases..... | 19 |
| Definición de historias de usuario..... | 19 |
| ○ <u>Usuario 1 web.....</u> | <u>19</u> |
| ○ <u>Usuario 1 journey map.....</u> | <u>20</u> |
| ○ <u>Usuario 2 web.....</u> | <u>21</u> |
| ○ <u>Usuario 1 juego.....</u> | <u>23</u> |
| ○ <u>Usuario 1 journey map.....</u> | <u>23</u> |
| ○ <u>Usuario 2 juego.....</u> | <u>25</u> |
| ○ <u>Usuario 2 juego journey map.....</u> | <u>25</u> |
| ● Diagrama de despliegue señalando las partes del Modelo, Vista y Controlador..... | 26 |
| ● Diagrama de componentes señalando la arquitectura cliente – servidor..... | 28 |
| Caso de prueba..... | 28 |
| Caso de prueba Nivel 1..... | 29 |
| Caso de prueba Nivel 2..... | 30 |
| Caso de prueba Nivel 3..... | 31 |
| Módulo 3..... | 33 |
| Este endpoint permite hacer los request a la base de datos para poder hacer el login con un usuario admin pre registrado y saber si existe y darle los permisos de loguearse..... | 35 |
| Módulo 4..... | 38 |
| Storyboard..... | 38 |
| Referencias:..... | 43 |

| | |
|--|-----------|
| Módulo 1(25%)..... | 48 |
| Módulo 2(25%)..... | 48 |
| Módulo 3(25%)..... | 48 |
| Módulo 4(25%)..... | 48 |
| Entregables finales por ELumen:..... | 50 |
| Ponderación de la evidencia para el curso:..... | 50 |

Introducción

En el contexto actual de la enseñanza matemática, el equipo de desarrollo se ha embarcado en la etapa inicial de investigación y diseño, con el objetivo principal de optimizar el proceso de aprendizaje matemático en estudiantes de quinto grado de la Escuela Metropolitana "La Luz". Este esfuerzo se centra en la mejora de la comprensión y aplicación de conceptos relacionados con superficies geométricas. Mediante un exhaustivo análisis del entorno educativo y las necesidades específicas de los alumnos, estamos comprometidos en diseñar una solución que armonice el entretenimiento proporcionado por los videojuegos con el avance académico indispensable.

El proyecto en cuestión pretende desarrollar una experiencia de aprendizaje interactiva y estimulante que facilite a los estudiantes la revisión eficaz de conceptos matemáticos y fortalezca su comprensión de estos. El videojuego, que se desarrollará utilizando la plataforma Unity, está dirigido específicamente a alumnos de quinto grado de la escuela urbana "La Luz", aunque se diseñará de manera que también sea relevante y ajustable para estudiantes de otros niveles, acorde a los estándares educativos establecidos por la Secretaría de Educación Pública (SEP) para cada grado escolar.

La estructura del juego se compone de tres niveles distintos:

1. En el **primer nivel**, los estudiantes serán introducidos a las definiciones de superficies y reconocerán formas geométricas básicas.
2. En el **segundo nivel**, los alumnos realizan operaciones de suma y multiplicación que involucran superficies, reforzando simultáneamente conceptos fundamentales de geometría.
3. En el **tercer nivel**, los estudiantes progresan en el cálculo del área de formas geométricas y explorarán temas más avanzados relacionados con el cálculo de superficies.

Este enfoque didáctico no solo atenderá a los alumnos de quinto grado, sino que también permitirá a estudiantes de otros niveles involucrarse de manera significativa y divertida con los conceptos matemáticos. Los estudios previos realizados sobre iniciativas similares validan la viabilidad de este proyecto. Un caso de referencia es el videojuego "Moon the alien" de Álvaro Rangel Arranz, que combina elementos educativos con actividades lúdicas, demostrando que la integración de tecnologías de aprendizaje interactivo es una estrategia eficaz en el contexto educativo moderno.

Módulo 1

Modelo relacional de la base de datos versión beta en dbdiagram.io.

Este esquema describe la estructura y relación entre las entidades 'Users' y 'Alumnos', las cuales conforman la versión preliminar de la base de datos:

```
1  Table Users {
2    id INTEGER [pk, increment]
3    username TEXT [not null]
4    password TEXT [not null]
5  }
6
7  Table Alumnos {
8    id INTEGER [pk, increment]
9    grupo TEXT
10   lista TEXT
11   genero TEXT
12   puntuacion1 INTEGER [default: 0]
13   puntuacion2 INTEGER [default: 0]
14   puntuacion3 INTEGER [default: 0]
15 }
```

| Users | | Alumnos | |
|----------|---------|-------------|---------|
| id | INTEGER | id | INTEGER |
| username | TEXT NN | grupo | TEXT |
| password | TEXT NN | lista | TEXT |
| | | genero | TEXT |
| | | puntuacion1 | INTEGER |
| | | puntuacion2 | INTEGER |
| | | puntuacion3 | INTEGER |

La tabla `Users` está diseñada para almacenar la información de inicio de sesión de los usuarios, mientras que la tabla `Alumnos` está destinada a registrar datos académicos y personales relevantes de los estudiantes. Ambas tablas tienen un campo de identificación único (`id`) que se incrementa automáticamente con cada nuevo registro. Los campos `username` y `password` son obligatorios para garantizar que no se creen usuarios sin las credenciales esenciales. En cuanto a la tabla `Alumnos`, se ha previsto un valor por defecto de 0 para las puntuaciones, lo que permite la inicialización de estos campos en ausencia de datos.

```

@app.route('/')
def informacion():
    return render_template('inicio.html')

```

Ruta principal donde te manda a la página de inicio y te muestra la pagina principal de nuestra página web

```

@app.route('/leaderboard')
def paginaprincipal():
    conn = get_db_connection()

    # Puntuaciones por grupo
    data = conn.execute('SELECT grupo, AVG(puntuacion1) as avg_puntuacion1, AVG(puntuacion2) as avg_puntuacion2, AVG(puntuacion3) as avg_pu
grupos = [row['grupo'] for row in data]
avg_puntuaciones1 = [row['avg_puntuacion1'] for row in data]
avg_puntuaciones2 = [row['avg_puntuacion2'] for row in data]
avg_puntuaciones3 = [row['avg_puntuacion3'] for row in data]

# 10 mejores puntuaciones por grupo
top_puntuaciones_por_grupo = []

for grupo in grupos:
    top_puntuaciones = conn.execute('SELECT * FROM alumnos WHERE grupo = ? ORDER BY puntuacion1 DESC LIMIT 10', (grupo,)).fetchall()
    top_puntuaciones_por_grupo.append({'grupo': grupo, 'top_puntuaciones': top_puntuaciones})

# Puntuaciones por género
top_puntuaciones_por_genero = {}

generos = ['Masculino', 'Femenino', 'Otro']

for genero in generos:
    top_puntuaciones = conn.execute('SELECT * FROM alumnos WHERE genero = ? ORDER BY puntuacion1 DESC LIMIT 10', (genero,)).fetchall()
    top_puntuaciones_por_genero[genero] = top_puntuaciones

# Gráfico de barras
fig = go.Figure(data=[
    go.Bar(name='Nivel 1 Media', x=grupos, y=avg_puntuaciones1),
    go.Bar(name='Nivel 2 Media', x=grupos, y=avg_puntuaciones2),
    go.Bar(name='Nivel 3 Media', x=grupos, y=avg_puntuaciones3)
])

# Cambiar el diseño del gráfico
fig.update_layout(barmode='group', title_text='Puntuaciones Medias por Grupo y Nivel', xaxis_title='Grupo', yaxis_title='Media')

bar_div = plotly.offline.plot(fig, output_type='div')

```

Esta ruta crea la conexión de la base de datos y el flask para poder proyectar las gráficas en leaderboard. Pide a la base de datos las puntuaciones, grupo y número de lista para poder crear las gráficas de la media de cada grupo, los 10 mejores por grupo y los mejores por género.

```
@app.route('/login', methods=['GET', 'POST'])
def login():
    error = None

    if request.method == 'POST':
        username = request.form.get('username')
        password = request.form.get('password')

        user = get_admin(username)

        if user and user['password'] == password:

            session['logged_in'] = True
            return redirect(url_for('paginaprincipal'))
        else:

            error = 'Usuario o contraseña incorrectos'

    return render_template('login.html', error=error)
```

Este endpoint permite hacer los request a la base de datos para poder hacer el login con un usuario admin pre registrado y saber si existe y darle los permisos de loguearse.

```
@app.route('/index')
def index():
    if not session.get('logged_in'):
        return redirect(url_for('login'))
    return render_template('index.html')
```

Revisa si el login es correcto y una vez ingresado te renderiza el template index que es donde están los leaderboards.

```
@app.route('/logout')
def logout():
    session['logged_in'] = False
    return redirect(url_for('login'))
```

Registra si el usuario que está logueado quiere salir de la sesión.

```
@app.route('/alumnos', methods=['GET', 'POST'])
def alumnos():
    conn = get_db_connection()

    if request.method == 'POST':
        id = request.form.get('id')
        grupo = request.form.get('grupo')
        lista = request.form.get('lista')
        genero = request.form.get('genero')

        # Si el ID existe, actualiza el alumno
        if id:
            conn.execute('UPDATE alumnos SET grupo = ?, lista = ?, genero = ? WHERE id = ?',
                        (grupo, lista, genero, id))
            conn.commit()
        else:
            # Si el ID no existe, inserta un nuevo alumno
            conn.execute('INSERT INTO alumnos (grupo, lista, genero) VALUES (?, ?, ?)',
                        (grupo, lista, genero))
            conn.commit()

    alumnos = conn.execute('SELECT * FROM alumnos').fetchall()
    conn.close()

    return render_template('alumnos.html', alumnos=alumnos)
```

Este endpoint sirve para ingresar en un forms y mandar los nuevos registros de alumnos, también ayuda a poder editar los alumnos existentes y cambiarles las propiedades.

```

@app.route('/grupo', methods=['POST'])
def adsa():
    lista = request.form.get('lista')
    group = request.form.get('group')

    print(f'Recibido lista: {lista}, group: {group}')

    if not grupo_lista_existe(group, lista):
        response = {"error": "El grupo o lista no existe"}
        print("Error: Grupo o lista no existe")
        return jsonify(response), 400

    response = {"group": group}
    return jsonify(response)

@app.route('/id', methods=['POST'])
def CrearUsuario():
    data = request.form
    id = data.get('id')
    group = data.get('group')

    print(f'Recibido id: {id}, group: {group}')

    if not grupo_lista_existe(group, id):
        response = {"error": "El grupo o lista no existe"}
        print("Error: Grupo o lista no existe")
        return jsonify(response), 400

    response = {"id": id}
    return jsonify(response)

```

Estas rutas son las que reciben la información de unity y ven si la información existe en la base de datos y regresan a el servidor si existen o si existe algun error

```
@app.route('/existe', methods=['POST'])
def existe_grupo_lista():
    data = request.form
    grupo = data.get('grupo')
    lista = data.get('lista')

    print(f'Recibido grupo: {grupo}, lista: {lista}')

    if grupo_lista_existe(grupo, lista):
        response = {"existe": True}
    else:
        response = {"existe": False}

    return jsonify(response)
```

Este endpoint busca en la base de datos y compara si los datos ingresados son reales y existen en la base de datos y si existen regresa un true y permite iniciar sesión en el juego.

Módulo 2

Licencia

[Mate mágicos](#) © 2024 by Sebastián Elorduy Martínez Manzanero, Jorge Rodrigo Colín Rubio, Miguel Sebastián Reyes Moguel and Diego Samperio Arce is licensed under

[Creative Commons Attribution-NonCommercial 4.0 International](#) 



La licencia Creative Commons Atribución-NoComercial 4.0 Internacional (CC BY-NC 4.0) permite compartir (copiar y redistribuir el material en cualquier medio o formato) y adaptar (remixar, transformar y construir a partir del material) para cualquier propósito no comercial, siempre y cuando se dé crédito adecuadamente.

- Lista de requerimientos funcionales.

Verde = Completado.

- Tabla 1.1 Requerimientos funcionales webapp

| Requerimiento | Descripción |
|---------------|---|
| RFW01 | La aplicación web mostrará estadísticas obtenidas en el juego. |
| RFW02 | La aplicación web deberá conectarse a una base de dato. |
| RFW03 | La aplicación web deberá desplegar el tipo de licencia del juego. |
| RFW04 | La aplicación web deberá ser intuitiva. |

- Tabla 1.2 Requerimientos funcionales juego

| Requerimiento | Descripción |
|---------------|---|
| RF01 | La aplicación contará con un menú didáctico donde se encontrarán los créditos, leaderboard, inicio sesión y botón de play |
| RF02 | La aplicación tendrá dos leaderboards, top global y top por género. |
| RF03 | La aplicación tendrá los créditos donde se enlistan todos los creadores |
| RF04 | La aplicación contará con tres niveles los cuales se irán desbloqueando progresivamente. |

| | |
|------|--|
| RF05 | La aplicación deberá tener un sistema que monitoree y guarde la puntuación de los jugadores. |
| RF06 | La aplicación deberá aumentar progresivamente de dificultad conforme avanza en nivel. |
| RF07 | La aplicación tendrá que tener un tutorial o pistas para asistir al jugador en entender las mecánicas del juego. |
| RF08 | El juego contará con un límite de tiempo por nivel. |
| RF09 | El jugador tendrá 3 vidas por nivel y se acumularán si llegan a sobrar. |

- Lista de requerimientos no funcionales.

- Tabla 2.1 Requerimientos no funcionales webapp

| Requerimiento | Descripción |
|---------------|--|
| RFNW01 | La aplicación web normalizará sus tablas para tener un mejor control. |
| RFNW02 | El servidor de la aplicación web será en la nube. |
| RFNW03 | La aplicación web tendrá que tener el certificado HTTPS para asegurar el paso seguro de datos. |
| RFNW04 | La aplicación web usará el framework flask para backend. |

| | |
|--------|--|
| RFNW05 | La aplicación web usará el framework de REACT para frontend. |
|--------|--|

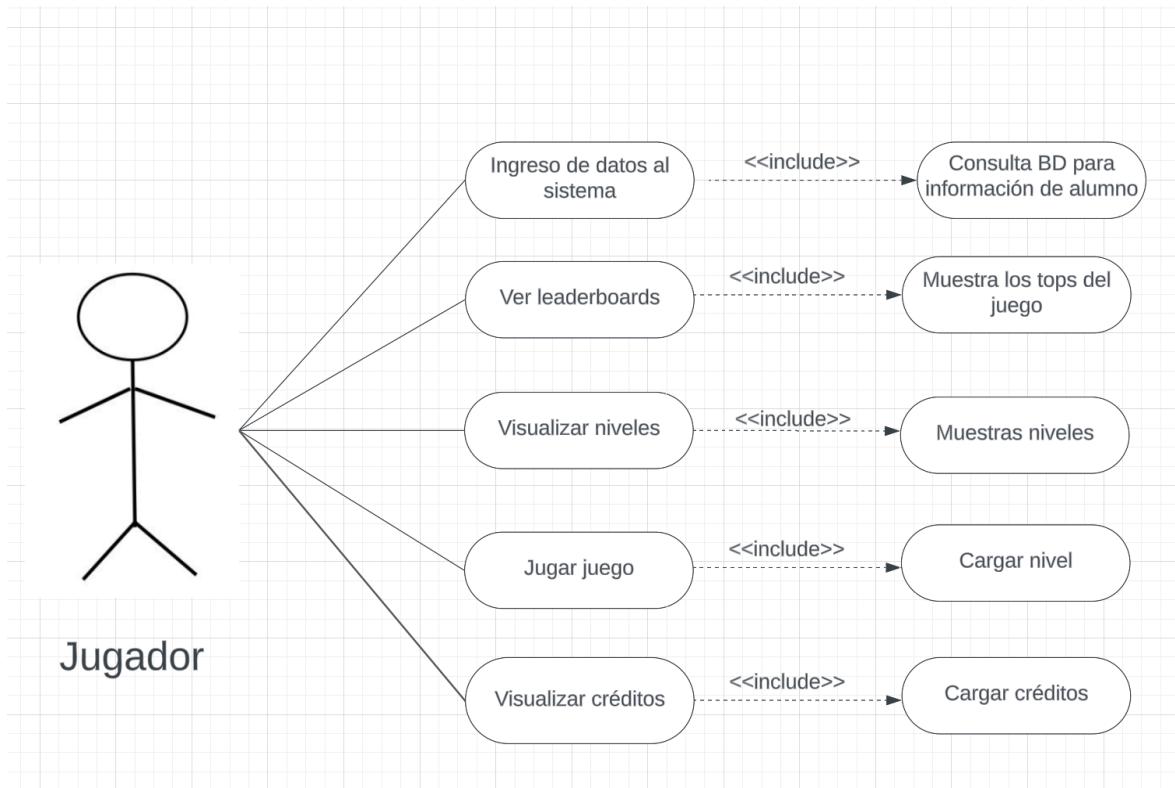
o Tabla 2.2 Requerimientos no funcionales juego

| Requerimiento | Descripción |
|---------------|---|
| RFN01 | La aplicación tendrá que ser accesible e intuitiva para los alumnos. |
| RFN02 | La aplicación deberá poder ejecutarse en un sistema windows 10 con especificaciones mínimas al menos con 30 fps. |
| RFN03 | La aplicación deberá contar con un tiempo de carga menor a 5 segundos. |
| RFN04 | La aplicación deberá ser desarrollada en unity siguiendo las mejores prácticas y asegurando que el código sea mantenible. |
| RFN05 | La aplicación deberá tener sistemas de seguridad SSL/TLS para transmisión de datos. |
| RFN06 | El servidor de la aplicación deberá ser implementado en la nube. |
| RFN07 | La aplicación deberá tener un sistema de seguridad AUTH que permita el login seguro de los usuarios. |

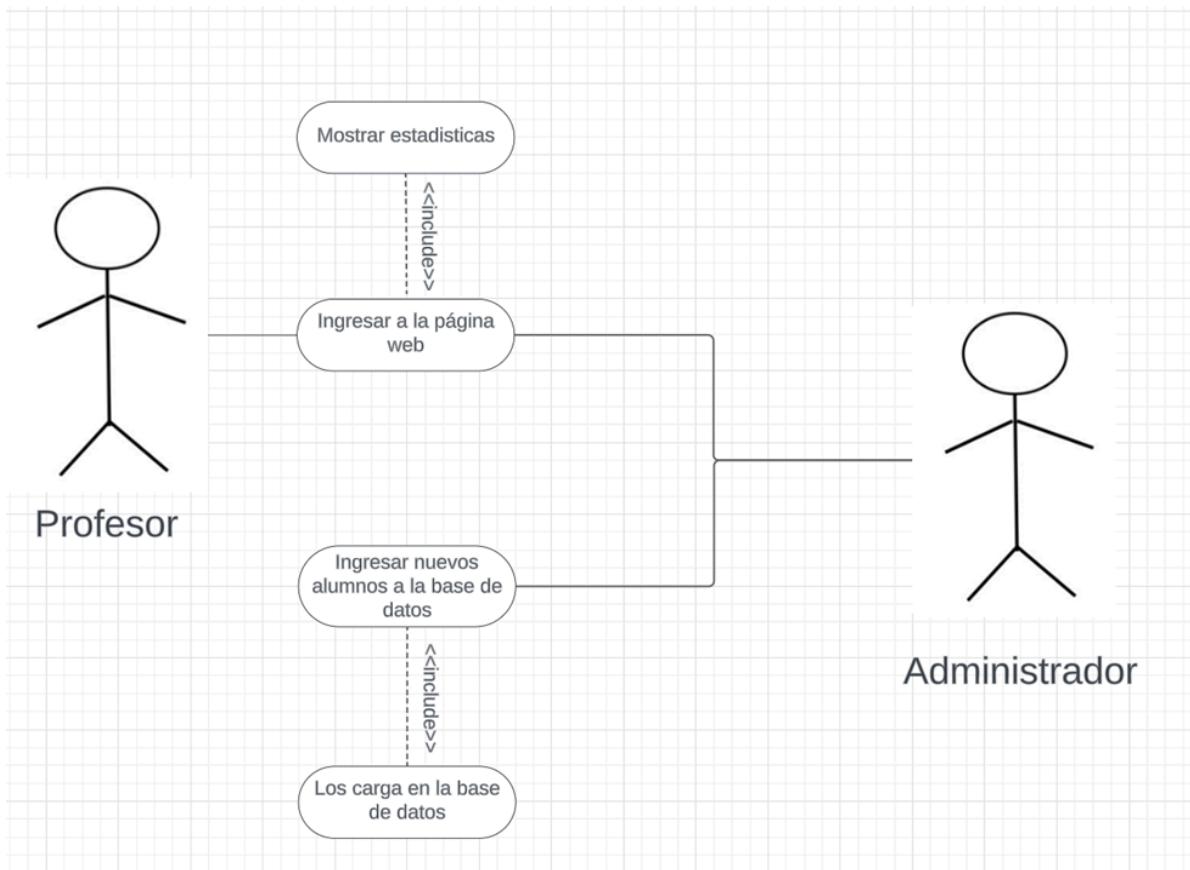
UML

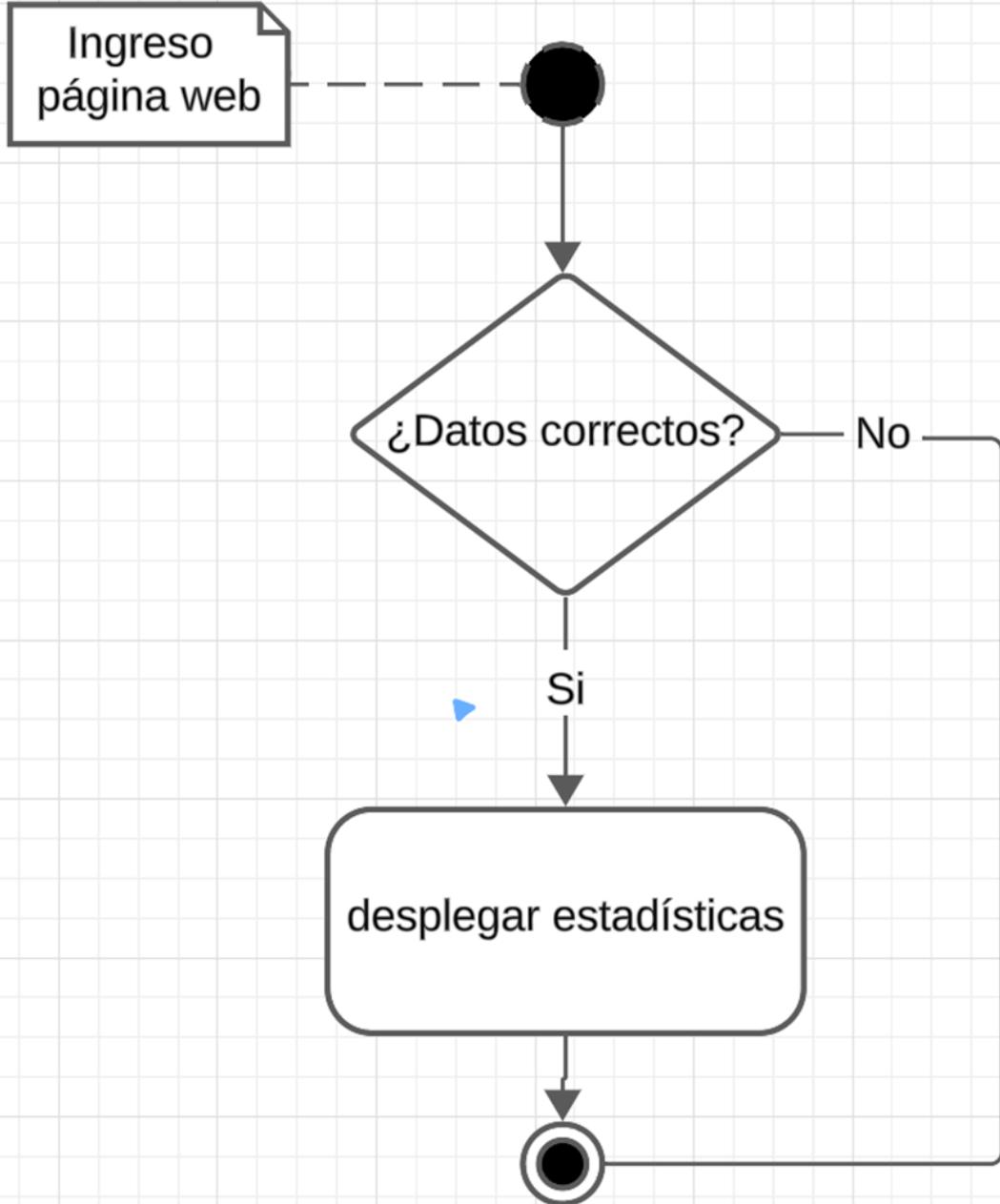
o Casos de uso juego

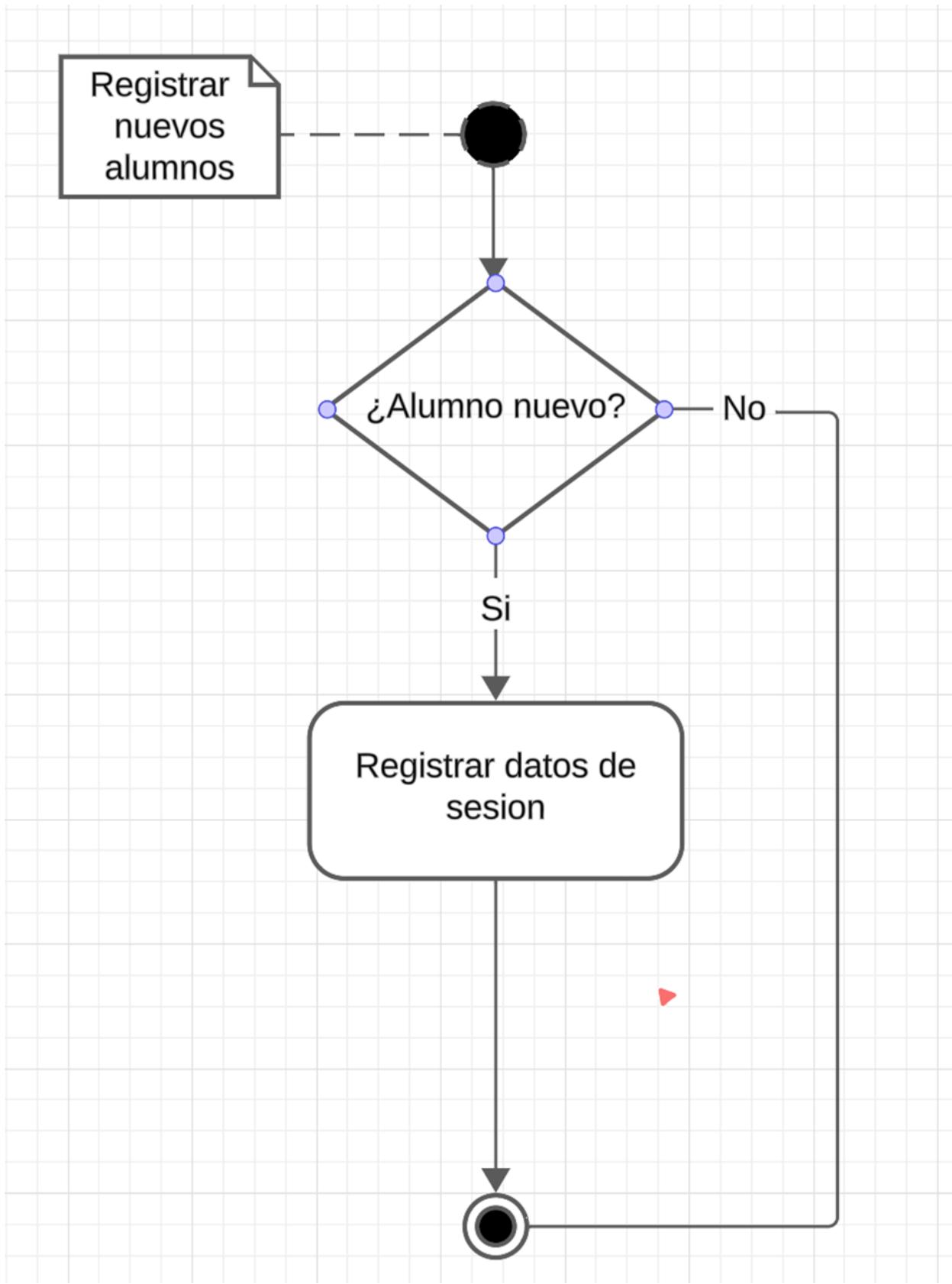
Caso 1.0



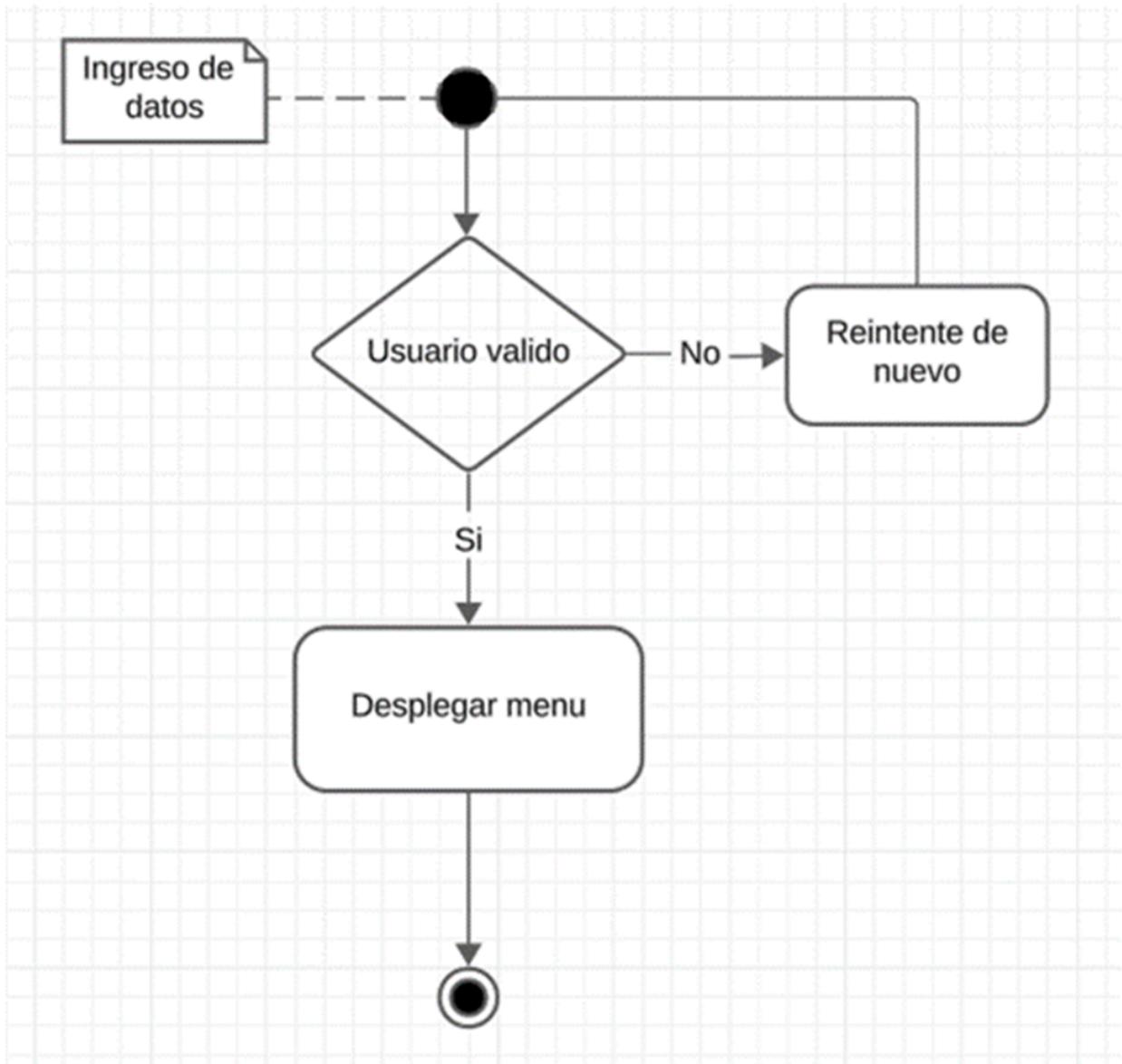
Caso 2.0

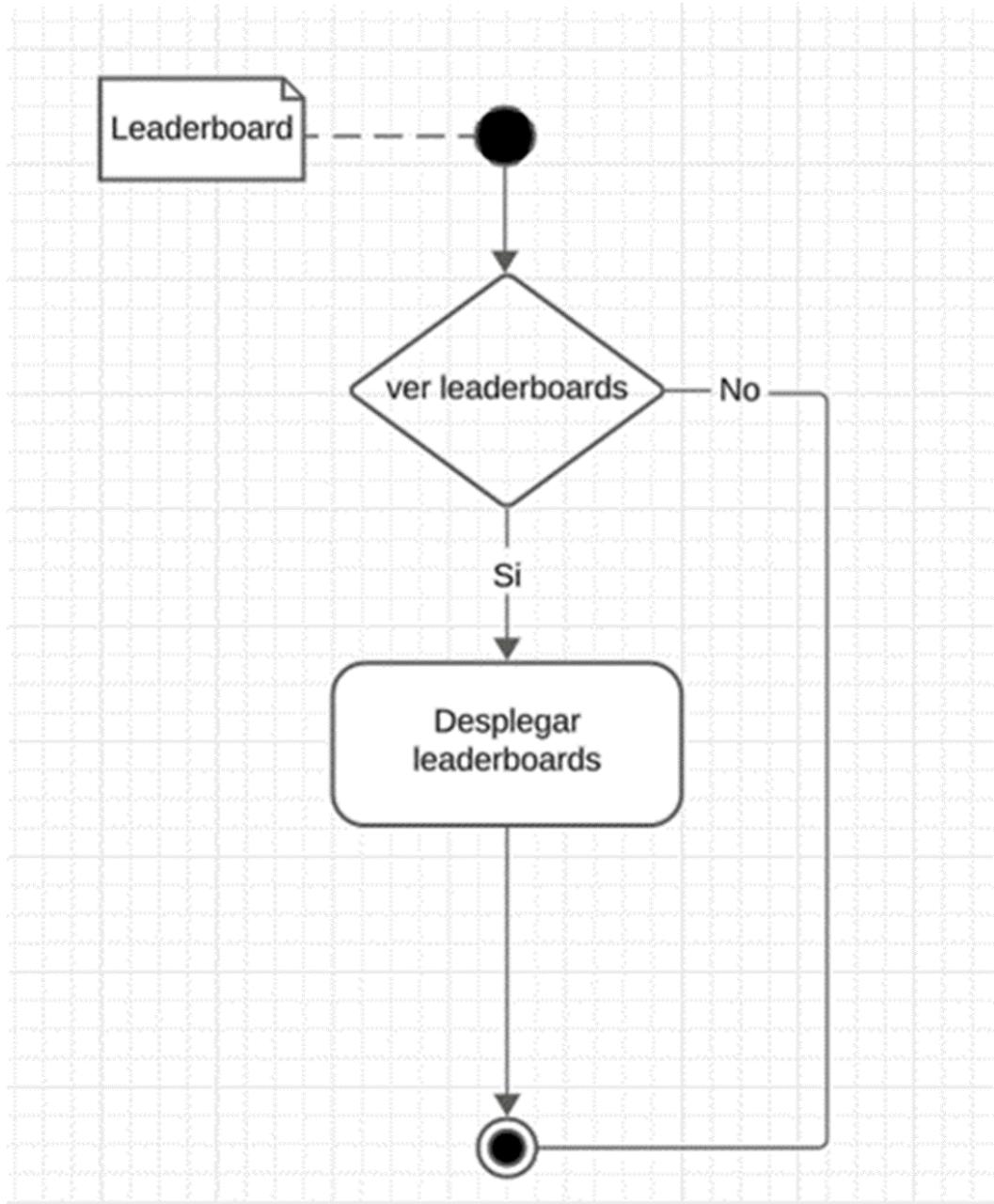


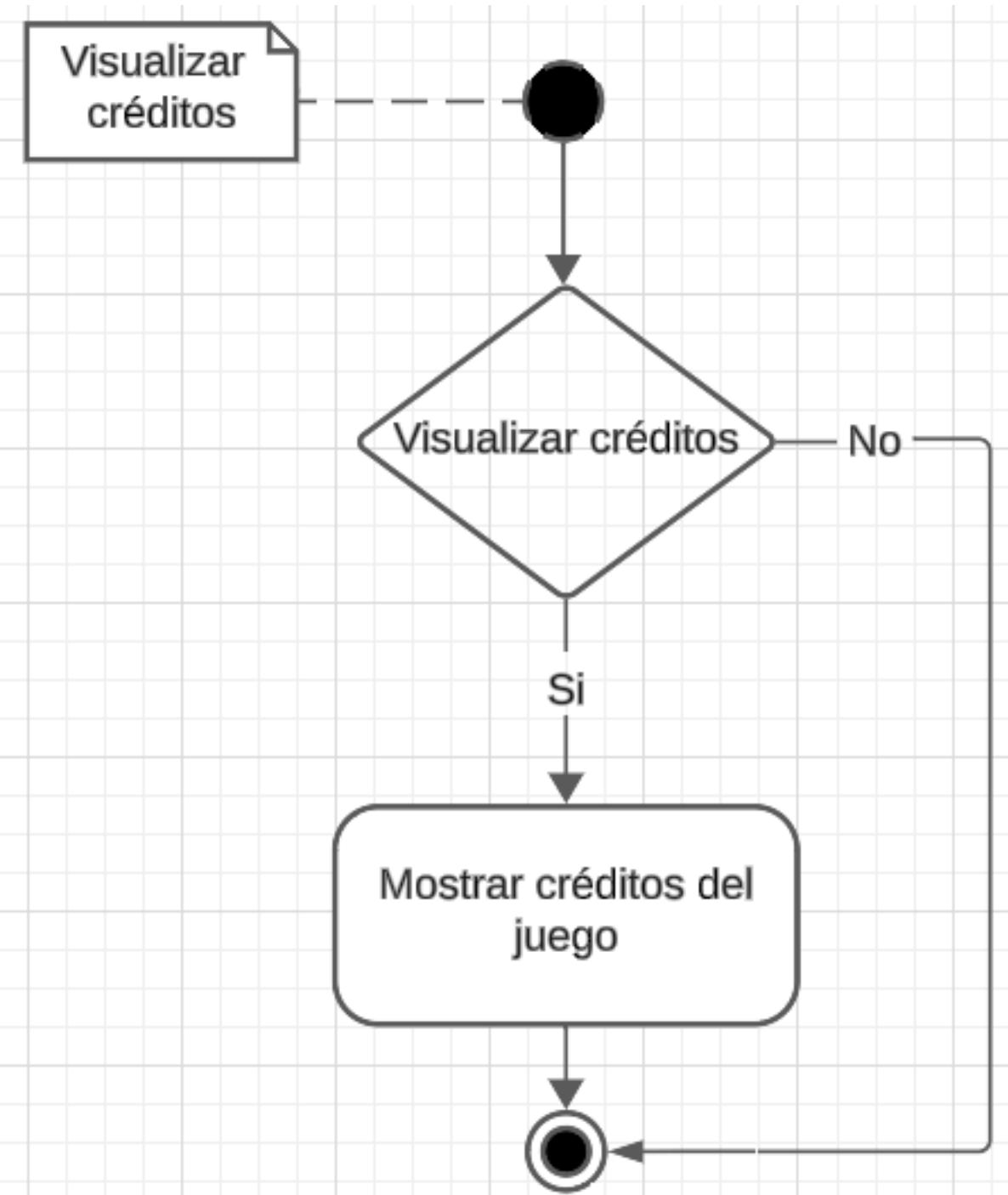


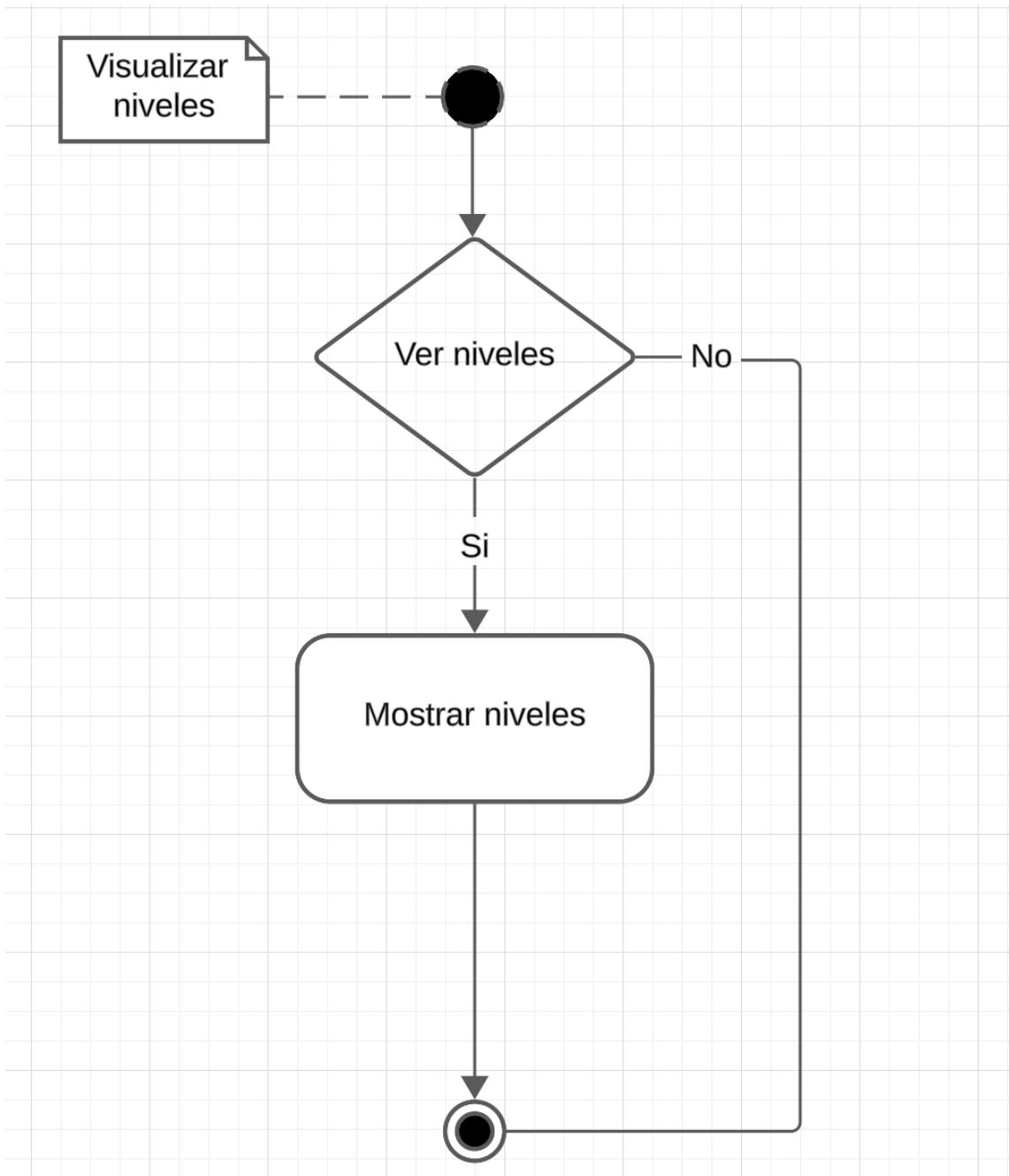


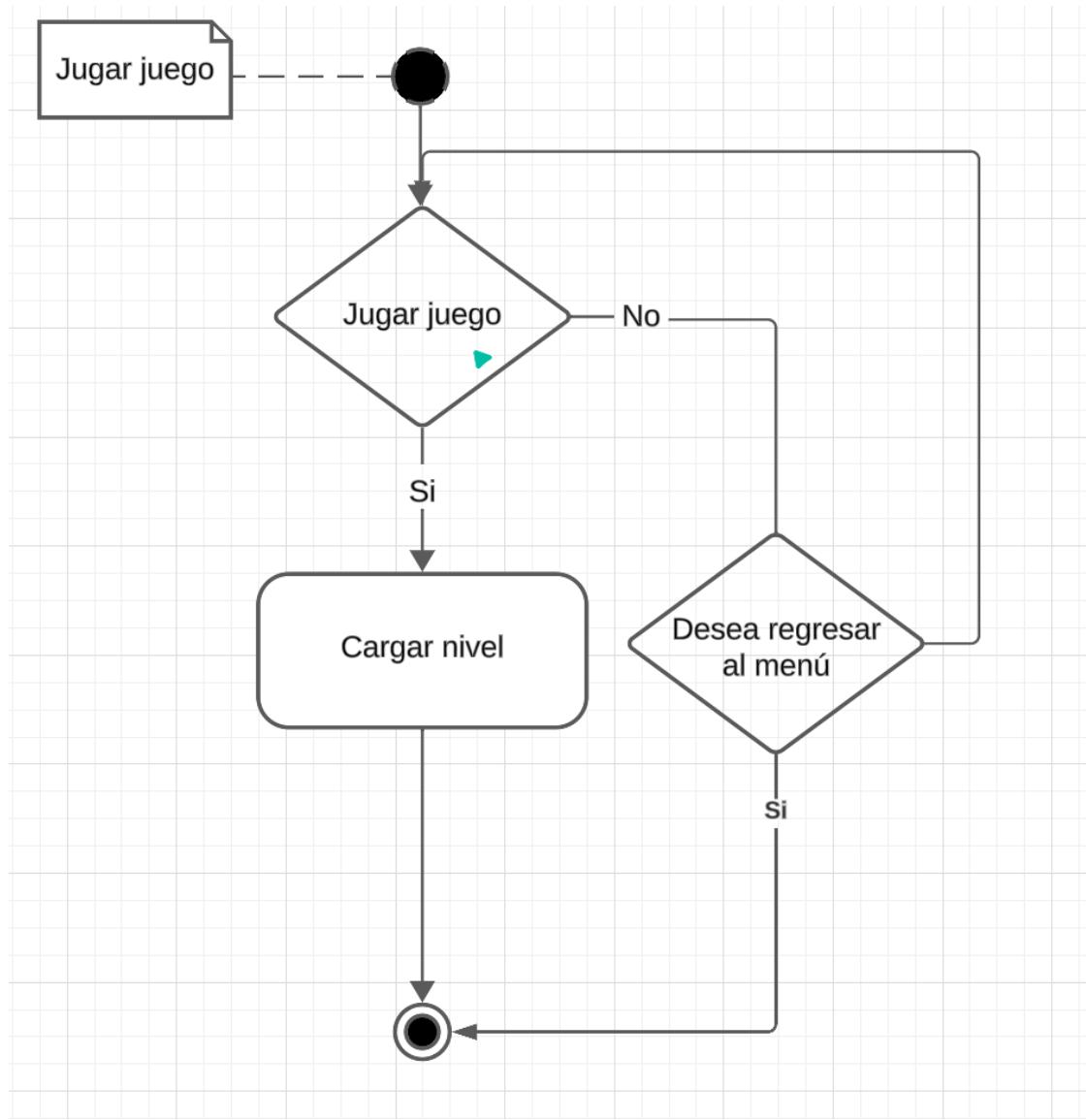
o Actividad juego





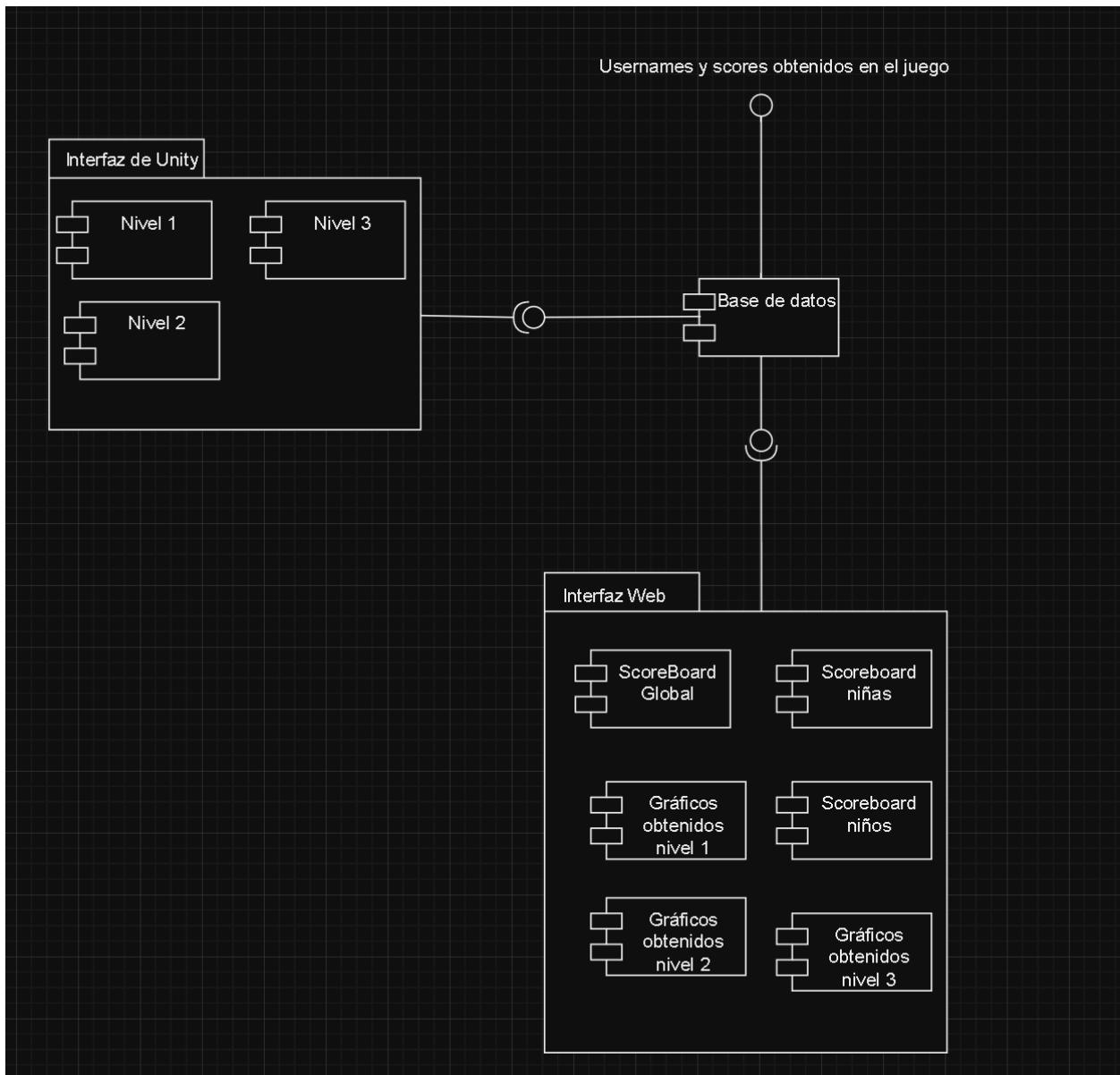






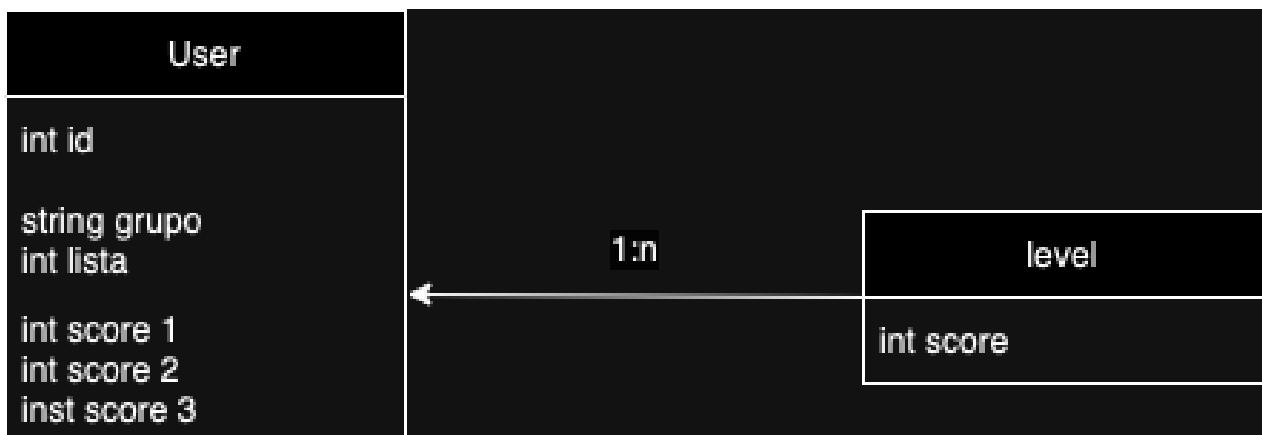
Diagramas

Diagrama de componentes



La imagen muestra un esquema de arquitectura de sistema con componentes principales: Una interfaz de unity donde se ingresan los datos de usuario para poder ingresar y se verifica en la base de datos, después de registrar los scores de en la base de datos se vuelven a mandar a la pagina web donde son utilizados para hacer gráficas y mostrar resultados.

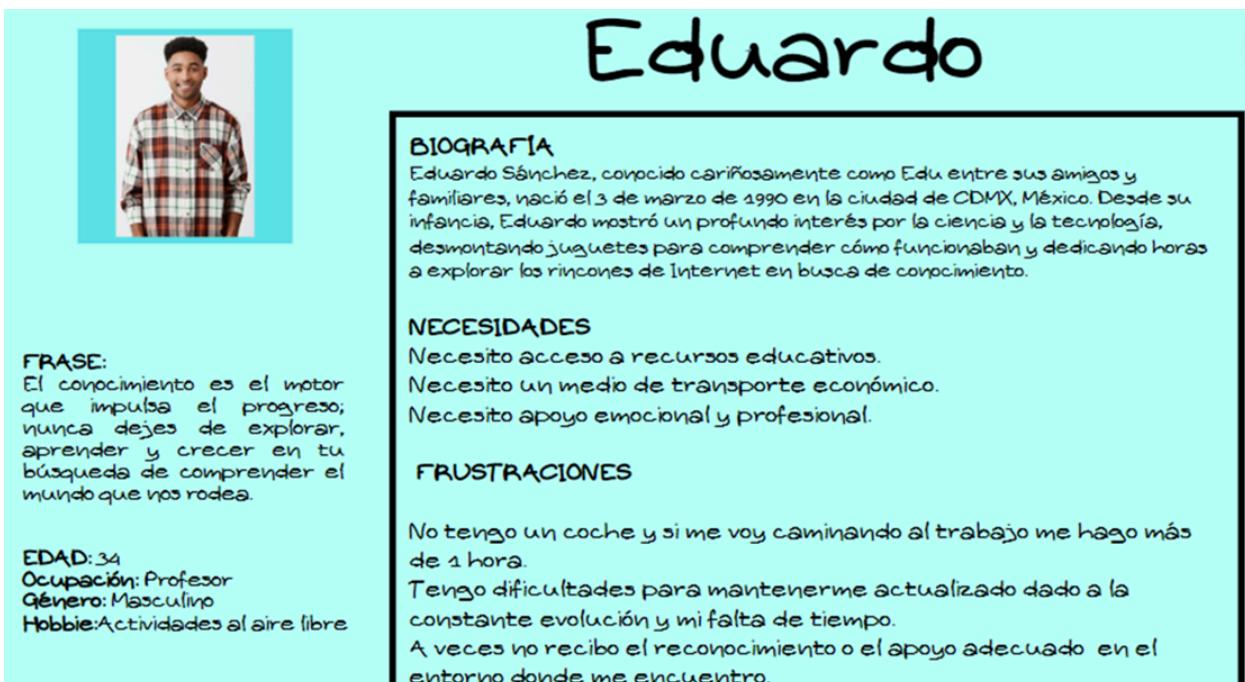
Diagrama de clases



Nuestro diagrama de clases cuenta con un “user” que es identificado mediante un id, grupo y número de lista. Además de que definimos un score diferente para cada nivel que vamos a subir a la base de datos .

Definición de historias de usuario.

o Usuario 1 web



Eduardo

BIOGRAFIA
Eduardo Sánchez, conocido cariñosamente como Edu entre sus amigos y familiares, nació el 3 de marzo de 1990 en la ciudad de CDMX, México. Desde su infancia, Eduardo mostró un profundo interés por la ciencia y la tecnología, desmontando juguetes para comprender cómo funcionaban y dedicando horas a explorar los rincones de Internet en busca de conocimiento.

NECESIDADES
Necesito acceso a recursos educativos.
Necesito un medio de transporte económico.
Necesito apoyo emocional y profesional.

FRUSTRACIONES
No tengo un coche y si me voy caminando al trabajo me hago más de 1 hora.
Tengo dificultades para mantenerme actualizado dado a la constante evolución y mi falta de tiempo.
A veces no recibo el reconocimiento o el apoyo adecuado en el entorno donde me encuentro.

FRASE:
El conocimiento es el motor que impulsa el progreso; nunca dejes de explorar, aprender y crecer en tu búsqueda de comprender el mundo que nos rodea.

EDAD: 34
Ocupación: Profesor
Género: Masculino
Hobbie: Actividades al aire libre

o Usuario 1 journey map

Journey Map Profesor

| | Registro | Visualización de estadísticas | Busqueda de estudiantes |
|--------------|---|--|---|
| Necesidades | Una medio para poder visualizar los resultados de sus alumnos | Ver los resultados obtenidos por sus alumnos | Poder encontrar un alumno en específico |
| Acciones | Agregar cuenta y contraseña | Visualizar las graficas mostradas en la página web | Buscar a un alumno por su nombre y número de lista |
| Herramientas | Internet, dispositivo electrónico y correo | Dispositivo electrónico, internet y pagina web | Dispositivo electrónico, internet y pagina web |
| Storyboard |  |  |  |
| Experiencia |  |  |  |

| | Registro | Visualización de estadísticas | Busqueda de estudiantes |
|-----------|---|--|---|
| Problemas | Validación de correo y fallas en el internet. | Problemas de visualización | Alumno no encontrado y fallas en el internet |
| Ganancias | Más herramientas educativas | Ver el resultado de sus enseñanzas en las graficas | Encontrar al alumno para visualizar su progreso |

0 Usuario 2 web



Henry

BIOGRAFIA

Soy un apasionado de las actividades al aire libre, me encanta ver la naturaleza, soy un programador, trabajo en Samsung por la Roma y tengo 40 años.

NECESIDADES

Necesito mejorar la calidad de las actividades escolares
Necesito que las nuevas generaciones aprendan
Necesito que sea una app funcional que ayude al desarrollo de los infantes

FRUSTRACIONES

Las anticuadas formas de enseñanza tienen que ser modernizadas con las nuevas tecnologías que poseemos, es por ello que quiero crear nuevas formas de aprendizaje para que a las nuevas generaciones les interese la escuela y aprender un poco mas

FRASE:
Soy un apasionado del ejercicio y los videojuegos, creo que es la base del éxito y tener un buen equilibrio entre tu vida laboral y extraoficial es lo mejor que puedes hacer por ti mismo.

TELÉFONO: 55-1218-1234
EDAD: 40
Ocupación: Programador
Género: Masculino
Hobby: Videojuegos

Journey Map Director

| | Registro | Visualización de estadísticas | Busqueda de estudiantes | Agregar administrativos y/o eliminar |
|--------------|---|--|--|---|
| Necesidades | Una medio para poder visualizar los resultados de sus alumnos | Ver los resultados obtenidos por sus alumnos | Poder encontrar un alumno en específico | Poder agregar nuevas personas al registro |
| Acciones | Agregar cuenta y contraseña | Visualizar las graficas mostradas en la página web | Buscar a un alumno por su nombre y número de lista | Agregar personas para poder hacer uso de la app |
| Herramientas | Internet, dispositivo electrónico y correo | Dispositivo electrónico, internet y página web | Dispositivo electrónico, internet y página web | Dispositivo electrónico, internet y página web |
| Storyboard | | | | |
| Experiencia | | | | |

| | Registro | Visualización de estadísticas | Busqueda de estudiantes | Agregar administrativos y/o eliminar |
|-----------|---|--|---|--------------------------------------|
| Problemas | Validación de correo y fallas en el internet. | Problemas de visualización | Alumno no encontrado y fallas en el internet | Problemas de registro |
| Ganancias | Más herramientas educativas | Ver el resultado de sus enseñanzas en las gráficas | Encontrar al alumno para visualizar su progreso | Agregar nuevas personas al sistema |

O Usuario 1 juego



Isaac

BIOGRAFIA

Le encanta jugar videojuegos, desde chico con su hermano se pasaba horas jugando, también le encanta jugar fútbol y sueña con ser youtuber.

FRASE:
Soy el mayor fan de los superhéroes y me encantan los videojuegos

EDAD: 10
Ocupación: Estudiante de sto de primaria
Género: Masculino
Hobby: Jugar videojuegos como fall guys, fortnite y roblox.

NECESIDADES

Necesita aprender más de como se hacen las operaciones básicas de resta y multiplicación.
Necesita comprender los temas de área y perímetro de figuras.
Necesita una forma de aprender más dinámica.

FRUSTRACIONES

Se aburre mucho en la escuela, siente que la forma de enseñanza es muy aburrida, la clase de matemáticas no le gusta ya que solo dejan ejercicios y no hacen la clase más divertida.

O Usuario 1 journey map

Journey Map usuario juego

| | Registro | Muestra de top | Nivel 1 | Nivel 2 | Nivel 3 |
|--------------|--|---|---|---|---|
| Necesidades | Poder ingresar al juego para jugar niveles y/o ver el top de jugadores | Ver comparativa de avance con compañeros y motivación | Comprender bien los temas de operaciones básicas, como resta y multiplicación | Aprender las temáticas de figuras, junto con sus características y fórmulas | Aprender sobre los temas área y perímetro |
| Acciones | Ingresar el número de lista y grupo al que pertenece | Seleccionar la opción del trofeo | Seleccionar el botón de jugar y el nivel 1 | Seleccionar el botón de jugar y el nivel 2, además de haber pasado el nivel 1 | Seleccionar el botón de jugar y el nivel 3, además de haber pasado el nivel 1 y 2 |
| Herramientas | Internet, computadora, número de lista y grupo | Internet, conexión con base de datos | Mouse teclado, papel y pluma | Mouse teclado, papel y pluma | Mouse teclado, papel y pluma |
| Storyboard | | | Level 1 | Level 2 | Level 3 |
| Experiencia | | | | | |

Journey Map usuario juego

| | Registro | Muestra de top | Nivel 1 | Nivel 2 | Nivel 3 |
|-----------|--|--|--|--|--|
| Problemas | Validación de número de lista y grupo | Fallo de puntuación y actualización del top | Fallo de respuestas correctas en el nivel. | Fallo de respuestas correctas en el nivel. | Fallo de respuestas correctas en el nivel. |
| Ganancias | Acceso a sistema de niveles y sistema de top | Posibilidad de ver estadística de género y general | Nivel 2 desbloqueado si aprueba nivel 1 | Nivel 3 desbloqueado si aprueba nivel 2 | Posibilidad de poder repetir los niveles y mejorar puntaje |

O Usuario 2 juego



Regina

BIOGRAFIA

Es una niña que le encanta hacer actividades deportivas como patinar y bailar, es muy comprometida en la escuela, siempre intenta estudiar en su casa temas que verán en un futuro y su materia favorita son las matemáticas.

FRASE:
Amo patinar, sueño con en algún momento ser una profesional y competir en las olimpiadas.

EDAD: 11
Ocupación: Estudiante de 5to de primaria
Género: Femenino
Hobby: Patinar y bailar

NECESIDADES
 Necesito reforzar temas de área y perímetro de figuras.
 Necesito entender más como se hacen las divisiones.
 Necesito conocer nuevas maneras y más rápidas de hacer operaciones.

FRUSTRACIONES
 Aunque ha repasado mucho los temas de figuras, todavía es algo que le falla un poco, además de que a veces se tarda mucho en los exámenes, porque no cuenta con mucha agilidad para hacer operaciones matemáticas.

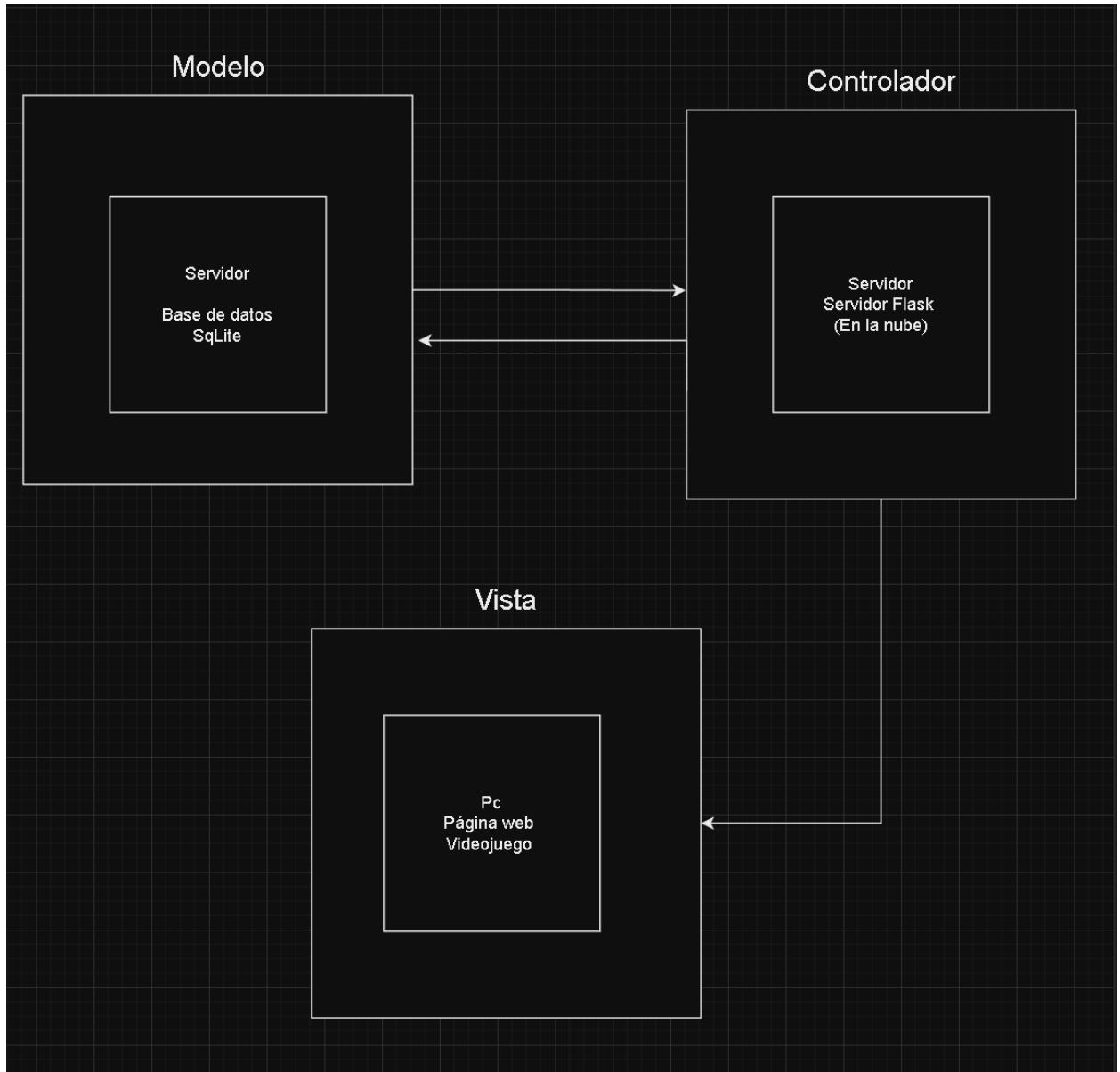
O Usuario 2 juego journey map

| Journey Map usuario juego | | | | | |
|---------------------------|---|---|---|---|---|
| | Registro | Muestra de top | Nivel 1 | Nivel 2 | Nivel 3 |
| Necesidades | Poder ingresar al juego para jugar niveles y/o ver el top de jugadores | Ver comparativa de avance con compañeros y motivación | Ganar agilidad en las operaciones básicas | Comprender los temas de figuras, junto con sus características y fórmulas | Entender como resolver los temas área y perímetro |
| Acciones | Ingresar el número de lista y grupo al que pertenece | Seleccionar la opción del trofeo | Seleccionar el botón de jugar y el nivel 1 | Seleccionar el botón de jugar y el nivel 2, además de haber pasado el nivel 1 | Seleccionar el botón de jugar y el nivel 3, además de haber pasado el nivel 1 y 2 |
| Herramientas | Internet, computadora, número de lista y grupo | Internet, conexión con base de datos | Mouse teclado, papel y pluma | Mouse teclado, papel y pluma | Mouse teclado, papel y pluma |
| Storyboard | <div style="display: inline-block; vertical-align: middle;"> Iniciar Sesión  </div> | <div style="display: inline-block; vertical-align: middle;">  </div> | Level 1 | Level 2 | Level 3 |
| Experiencia |  |  |  |  |  |

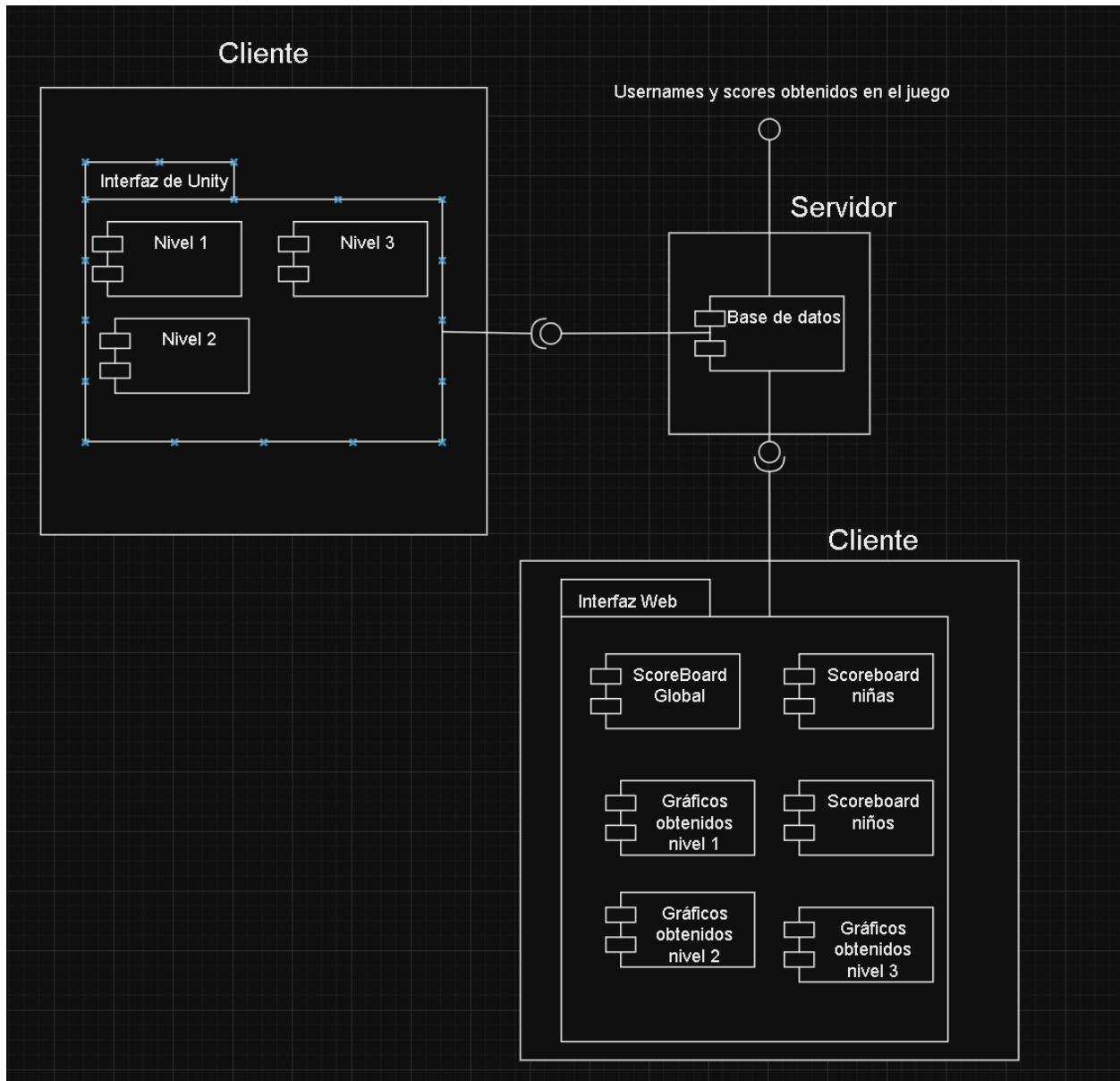
Journey Map usuario juego

| | Registro | Muestra de top | Nivel 1 | Nivel 2 | Nivel 3 |
|-----------|--|--|--|--|--|
| Problemas | Validación de número de lista y grupo | Fallo de puntuación y actualización del top | Fallo de respuestas correctas en el nivel. | Fallo de respuestas correctas en el nivel. | Fallo de respuestas correctas en el nivel. |
| Ganancias | Acceso a sistema de niveles y sistema de top | Posibilidad de ver estadística de género y general | Nivel 2 desbloqueado si aprueba nivel 1 | Nivel 3 desbloqueado si aprueba nivel 2 | Posibilidad de poder repetir los niveles y mejorar puntaje |

- Diagrama de despliegue señalando las partes del Modelo, Vista y Controlador.



- Diagrama de componentes señalando la arquitectura cliente – servidor.



Caso de prueba

Caso de prueba Nivel 1

| <u>ID Escenario de prueba</u> | <u>Videojuego</u> | | <u>ID Caso de prueba</u> | | <u>Nivel1</u> | |
|-------------------------------|---|---|---|---|--------------------------------------|--|
| <u>Descripción</u> | <u>Probar las mecánicas del primer nivel del juego.</u> | | <u>Prioridad</u> | | <u>Alta</u> | |
| <u>Probado por</u> | <u>Miguel Sebastian Reyes Moguel</u> | | <u>Revisado por</u> | | <u>Miguel Sebastian Reyes Moguel</u> | |
| <u>No.</u> | <u>Acción</u> | <u>Entradas</u> | <u>Salidas esperadas</u> | <u>Salidas obtenidas</u> | <u>Resultado</u> | <u>Comentarios</u> |
| 1 | <u>Caminar</u> | <u>Presionar cualquiera de las teclas "w", "a", "d" o "s"</u> | <u>Movimiento del jugador</u> | <u>Movimiento correcto del jugador.</u> | <u>Pass</u> | <u>Hacer el movimiento más fluido.</u> |
| 2 | <u>Entrar a todos los colliders de figuras.</u> | <u>Colisión con collider de figuras.</u> | <u>Actualización de pregunta</u> | <u>Correcta actualización de pregunta</u> | <u>Pass</u> | <u>Poner un delay al collider.</u> |
| 3 | <u>Finalización de nivel al término del contador.</u> | <u>Paso del tiempo a 0.</u> | <u>Finalización de juego a pantalla de fin.</u> | <u>Correcta finalización de nivel.</u> | <u>Pass</u> | <u>Mejor la pantalla final.</u> |

Caso de prueba Nivel 2

| <u>ID</u> <u>Escenario de prueba</u> | <u>Videojuego</u> | | <u>ID Caso de prueba</u> | <u>Nivel2</u> | | |
|---|---|--|--|--|------------------|---|
| <u>Descripción</u> | <u>Probando mecánicas y funcionalidades del nivel</u> | | <u>Prioridad</u> | <u>Alta</u> | | |
| <u>Probado por</u> | <u>Jorge Rodrigo Colín Rubio</u> | | <u>Revisado por</u> | <u>Jorge Rodrigo Colín Rubio</u> | | |
| <u>No.</u> | <u>Acción</u> | <u>Entradas</u> | <u>Salidas esperadas</u> | <u>Salidas obtenidas</u> | <u>Resultado</u> | <u>Comentarios</u> |
| 1 | <u>Pop Up de la pregunta</u> | <u>Conseguir 10 puntos en el score</u> | <u>Pregunta con operaciones matemáticas</u> | <u>Pregunta con operaciones matemáticas correctas.</u> | <u>Pass</u> | <u>La pregunta se muestra con las 4 operaciones suma,resta, multiplicacion y división</u> |
| 2 | <u>Colisión con los obstáculos</u> | <u>Colisión con los obstáculos</u> | <u>Reiniciar el nivel al momento de colisionar</u> | <u>Correcto reinicio del nivel</u> | <u>Pass</u> | <u>Funciona correctamente</u> |
| 3 | <u>Finalización de nivel al término del contador.</u> | <u>Paso del tiempo a 0.</u> | <u>Finalización de juego a pantalla de fin.</u> | <u>Correcta finalización de nivel.</u> | <u>Pass</u> | <u>Mejor la pantalla final.</u> |

Caso de prueba Nivel 3

| <u>ID Escenario de prueba</u> | <u>Videojuego</u> | | <u>ID Caso de prueba</u> | <u>Nivel3</u> | | |
|-------------------------------|--|--|--|--|----------------------------|--|
| <u>Descripción</u> | <u>Probando mecánicas y funcionalidades del nivel</u> | | | <u>Prioridad</u> | <u>Alta</u> | |
| <u>Probado por</u> | <u>Diego Samperio Arce</u> | | | <u>Revisado por</u> | <u>Diego Samperio Arce</u> | |
| <u>No.</u> | <u>Acción</u> | <u>Entradas</u> | <u>Salidas esperadas</u> | <u>Salidas obtenidas</u> | <u>Resultado</u> | <u>Comentarios</u> |
| 1 | <u>Respuestas correctas de los túneles.</u> | <u>Respuesta correcta a la pregunta.</u> | <u>Suma de 10 puntos al marcador final.</u> | <u>10 puntos sumados al marcador final.</u> | <u>Pass</u> | <u>La actualización del marcador se ve arriba a la derecha y se actualiza al instante.</u> |
| 2 | <u>Respuestas incorrectas de los túneles.</u> | <u>Respuesta incorrecta a la pregunta.</u> | <u>Resta de 10 puntos al marcador final.</u> | <u>10 puntos restados al marcador final.</u> | <u>Pass.</u> | <u>La actualización del marcador se ve arriba a la derecha y se actualiza al instante.</u> |
| 3 | <u>Finalización de nivel al llegar al último piso.</u> | <u>Detección de colisión del personaje con el último piso.</u> | <u>Detención del puntaje y tiempo de la partida.</u> | <u>Correcta finalización de nivel.</u> | <u>Pass.</u> | <u>El marcador y puntaje si se detienen y muestran en el canvas final.</u> |

Caso de prueba Web

| <u>ID Escenario de prueba</u> | | <u>WEB</u> | | <u>ID Caso de prueba</u> | | <u>WEB</u> | |
|-------------------------------|---|--|--|--|--|---|--|
| <u>Descripción</u> | | <u>Probando funcionalidad de la página web</u> | | <u>Prioridad</u> | | <u>Alta</u> | |
| <u>Probado por</u> | | <u>Sebastian Elorduy Martinez Manzanero</u> | | <u>Revisado por</u> | | <u>Sebastian Elorduy Martinez Manzanero</u> | |
| <u>No.</u> | <u>Acción</u> | | <u>Entradas</u> | <u>Salidas espera-das</u> | <u>Salidas obtenidas</u> | <u>Resultado</u> | <u>Comentarios</u> |
| 1 | <u>Envio datos de nuevo alumno</u> | | <u>inputs de salon y numero de lista</u> | <u>Creación de un nuevo alumno en la base de datos</u> | <u>Nuevo alumno en base de datos que se puede editar</u> | <u>Pass</u> | <u>Se agrega nuevo alumno a la base de datos y es posible modificarlo</u> |
| 2 | <u>Enviar datos nuevo administrador</u> | | <u>Nombre de usuario y contraseña del nuevo admin</u> | <u>Se crea el nuevo admin en la base de datos</u> | <u>Se crea nuevo admin y permite hacer login</u> | <u>Pass.</u> | <u>Se crea nuevo admin y puede hacer login</u> |
| 3 | <u>Editar administrador</u> | | <u>Detección de botón de edición y que se selecciona</u> | <u>Se selecciona el admin y es posible editarlo y cambiar contraseña</u> | <u>Se cambia contraseña correctamente y se actualiza en la base de datos</u> | <u>Pass.</u> | <u>Se actualiza admin en base de datos y se puede hacer login con los nuevos datos</u> |

Módulo 3

Leadboard Inicio Login

Puntuaciones Medias por Grupo

A bar chart titled "Puntuaciones Medias por Grupo y Nivel". The Y-axis is labeled "Media" and ranges from 0 to 500. The X-axis is labeled "Grupo" and shows two groups, A and B. Each group has three bars representing different levels: Nivel 1 Media (blue), Nivel 2 Media (orange/red), and Nivel 3 Media (green). Group A has approximately 270, 530, and 470 respectively. Group B has approximately 410, 210, and 410 respectively. The legend indicates: Nivel 1 Media (blue), Nivel 2 Media (orange/red), and Nivel 3 Media (green).

| Grupo | Nivel 1 Media | Nivel 2 Media | Nivel 3 Media |
|-------|---------------|---------------|---------------|
| A | ~270 | ~530 | ~470 |
| B | ~410 | ~210 | ~410 |

Mejores 10 Puntajes por Grupo

Lista: 5 - Nivel 1: 753, Nivel 2: 125, Nivel 3: 675
Lista: 3 - Nivel 1: 523, Nivel 2: 286, Nivel 3: 426
Lista: 20 - Nivel 1: 431, Nivel 2: 612, Nivel 3: 724
Lista: 7 - Nivel 1: 421, Nivel 2: 591, Nivel 3: 235

Mejores Puntajes por Género

Masculino

| Lista: | Nivel 1: | Nivel 2: | Nivel 3: |
|--------|----------|----------|----------|
| 5 | 905 | 193 | 461 |
| 5 | 753 | 125 | 675 |
| 4 | 634 | 295 | 452 |

Esta es la pestaña de leaderboard donde se proyectan las diferentes tablas separadas por los 10 mejores puntajes, mejores puntajes por género y las gráficas que representan las puntuaciones medias por grupo y nivel.

Leadboard Inicio Login

Login

Usuario:

Contraseña:

Esta es la página de login donde los administradores(profesores) se loguean para poder acceder a ciertas funciones de la página web.

The screenshot shows a web application interface. At the top, there is a navigation bar with links: Leadboard, Inicio, Logout, Alumnos, and Admin. Below the navigation bar, the main content area has a title "Lista de Alumnos". Underneath the title is a form with three input fields: "Grupo:" (with a dropdown menu), "Lista:" (with a dropdown menu), and "Género:" (with a dropdown menu set to "Masculino"). A blue button labeled "Añadir Alumno" is at the bottom of the form. Below the form, the title "Lista de Alumnos" appears again, followed by a list of three student entries: "Grupo: B, Lista: 50, Género: Masculino" with an "Editar" button; "Grupo: A, Lista: 20, Género: Femenino" with an "Editar" button; and "Grupo: A, Lista: 22, Género: Masculino" with an "Editar" button.

Esta página es donde se registran a los alumnos y se agrega el grupo y número de lista con el que van a poder ingresar al juego aparte de que igual forma se pueden editar para cambiar lista, grupo y/o género

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "elorduy50.pythonanywhere.com/admin". The page content is titled "Administradores". It lists three users: "Matematicos" (with Editar and Borrar buttons), "sebastian" (with Editar and Borrar buttons), and "Sebastian" (with Editar and Borrar buttons). Below this is a section titled "Agregar Administrador" containing two input fields: "Nombre de usuario:" and "Contraseña:", followed by a blue "Agregar" button. Further down is a section titled "Editar Administrador" containing the same two input fields: "Nombre de usuario:" and "Contraseña:".

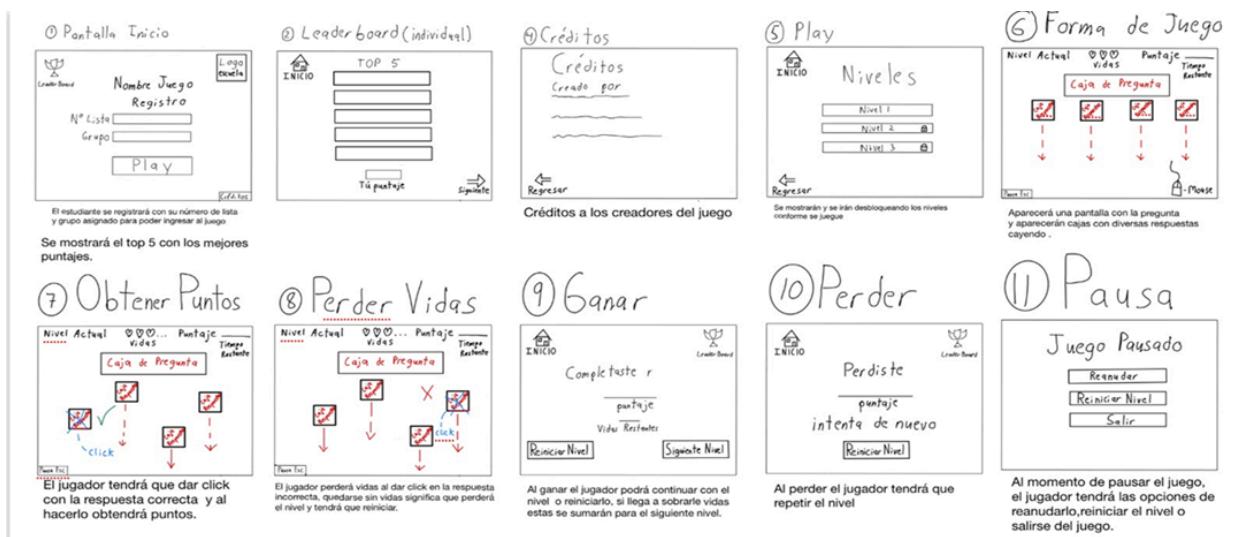
En esta pagina es donde se editan los administradores se pueden agregar nuevos administradores y se pueden borrar administradores



Esta es la página de inicio donde se va a poder descargar el juego .

Módulo 4

Storyboard



El storyboard muestra el proceso general del funcionamiento del juego. Cada recuadro es una escena conectada mediante botones. El juego contará con menú, leaderboard, selección de niveles y sus respectivos niveles con sus pantallas de victoria y derrota.

Se realizaron dos mockups de cómo se vería nuestro juego, el primero de estos es uno basado en el juego de Fortnite, con tipos de letras, colores, fondos y música basadas en dicho

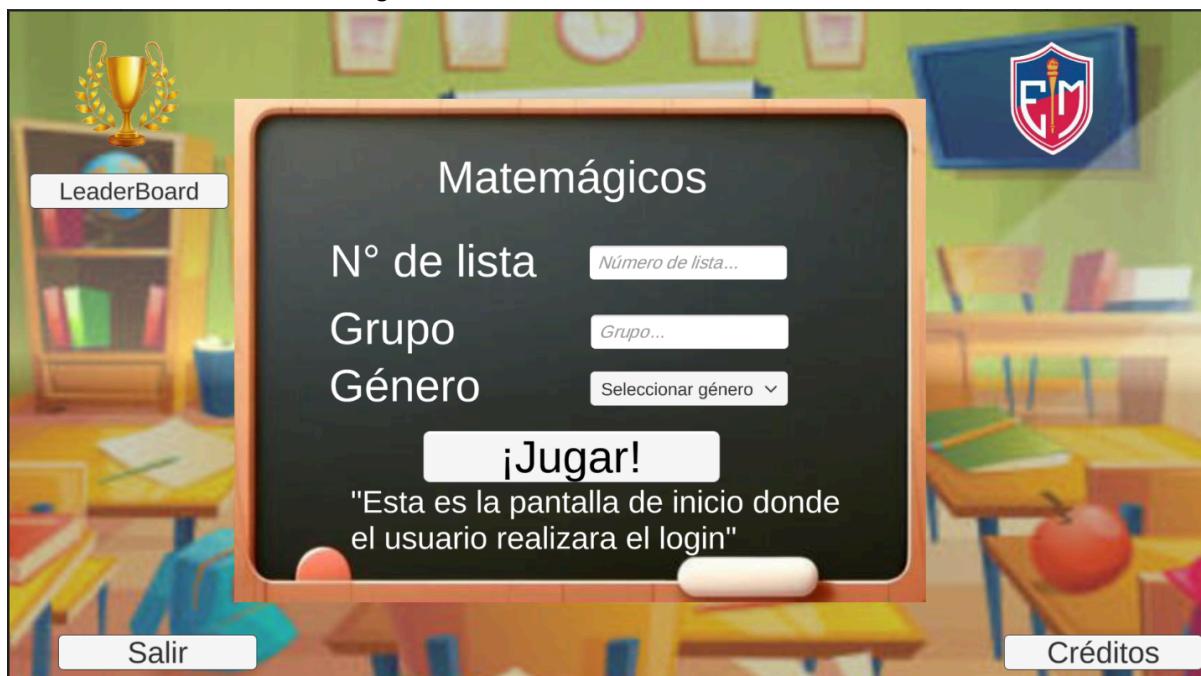
juego, esto con base en un estudio en el que vimos que actualmente es uno de los juegos más populares entre los niños de ese rango de edad.

La muestra del canva es el siguiente:



El segundo mockup y que escogimos, cuenta con fondos mucho más vivos y dinámicos en los que se observa como un aula de clase, con un diseño más propositivo, original y con detalles más únicos, es por esto que lo elegimos como diseño final para el videojuego.

La muestra del canva es el siguiente:



Al poner int en los diferentes input fields estos serán mandados al servidor donde el servidor los recibirá en formato json y nuevamente el servidor regresa los datos en formato json y en unity

en la consola se puede ver los datos y a que pertenecen. Gracias a esto podemos ver que las conexiones entre nuestro unity y el servidor son correctas.

```
IEnumerator SendId(User u)
{
    WWWForm form = new WWWForm();
    form.AddField("user", JsonUtility.ToJson(u));
    using (UnityWebRequest ww = UnityWebRequest.Post("https://elorduy45.pythonanywhere.com/id", form))
    {
        yield return ww.SendWebRequest();

        if (ww.result != UnityWebRequest.Result.Success)
        {
            Debug.Log(ww.error);
        }
        else
        {
            string response = ww.downloadHandler.text;
            User temp = JsonUtility.FromJson<User>(response);
            Debug.Log("Id asignado" + " " + temp.id);
        }
    }
}

IEnumerator SendGrupo(User u)
{
    WWWForm form = new WWWForm();
    form.AddField("user", JsonUtility.ToJson(u));
    using (UnityWebRequest ww = UnityWebRequest.Post("https://elorduy45.pythonanywhere.com/grupo", form))
    {
        yield return ww.SendWebRequest();

        if (ww.result != UnityWebRequest.Result.Success)
        {
            Debug.Log(ww.error);
        }
        else
        {
            string response = ww.downloadHandler.text;
            User temp = JsonUtility.FromJson<User>(response);
            Debug.Log("Grupo asignado" + " " + temp.group);
        }
    }
}

string grupo = tmp_grupo.text;
string lista = tmp_lista.text;

User u = new User();
u.id = int.Parse(lista);
u.group = int.Parse(grupo);

StartCoroutine(SendId(u));
StartCoroutine(SendGrupo(u));

IEnumerator CambiarDespuesDeEspera()
{
    yield return new WaitForSeconds(5);

    SceneManager.LoadScene("Niveles");
}

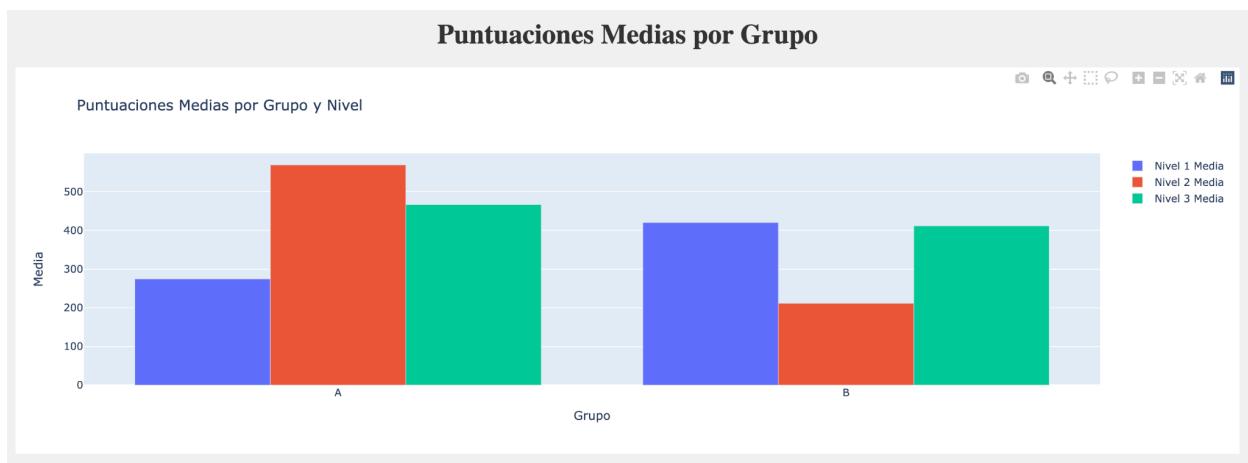
StartCoroutine(CambiarDespuesDeEspera());
```

Este es nuestro código en c# el cual nos ayuda a mandar los datos ingresados al servidor y de regreso a unity. También es donde colocamos el url del server que ya está subido en la nube y está corriendo de forma permanente.

Mejores 10 Puntajes por Grupo

| | |
|----------|--|
| Grupo: A | Lista: 5 - Nivel 1: 753, Nivel 2: 125, Nivel 3: 675 |
| | Lista: 3 - Nivel 1: 523, Nivel 2: 286, Nivel 3: 426 |
| | Lista: 20 - Nivel 1: 431, Nivel 2: 612, Nivel 3: 724 |
| | Lista: 7 - Nivel 1: 421, Nivel 2: 591, Nivel 3: 235 |
| | Lista: 1 - Nivel 1: 345, Nivel 2: 163, Nivel 3: 732 |
| | Lista: 4 - Nivel 1: 235, Nivel 2: 928, Nivel 3: 245 |
| | Lista: 8 - Nivel 1: 235, Nivel 2: 592, Nivel 3: 235 |
| | Lista: 10 - Nivel 1: 126, Nivel 2: 965, Nivel 3: 362 |
| | Lista: 2 - Nivel 1: 125, Nivel 2: 759, Nivel 3: 754 |
| | Lista: 9 - Nivel 1: 125, Nivel 2: 592, Nivel 3: 632 |
| Grupo: B | Lista: 5 - Nivel 1: 905, Nivel 2: 193, Nivel 3: 461 |
| | Lista: 4 - Nivel 1: 634, Nivel 2: 295, Nivel 3: 452 |
| | Lista: 8 - Nivel 1: 623, Nivel 2: 105, Nivel 3: 367 |
| | Lista: 6 - Nivel 1: 544, Nivel 2: 195, Nivel 3: 341 |
| | Lista: 10 - Nivel 1: 462, Nivel 2: 184, Nivel 3: 236 |
| | Lista: 33 - Nivel 1: 423, Nivel 2: 23, Nivel 3: 346 |
| | Lista: 9 - Nivel 1: 423, Nivel 2: 158, Nivel 3: 237 |
| | Lista: 7 - Nivel 1: 324, Nivel 2: 104, Nivel 3: 235 |
| | Lista: 1 - Nivel 1: 245, Nivel 2: 582, Nivel 3: 734 |
| | Lista: 3 - Nivel 1: 234, Nivel 2: 391, Nivel 3: 373 |

Esta tabla refleja los mejores 10 puntajes por grupo separando los diferentes grupos



Esta gráfica refleja las medias de grupo y los separa por niveles que conectan con un color en específico.

Mejores Puntajes por Género

Masculino

| | | | |
|-----------|--------------|--------------|--------------|
| Lista: 5 | Nivel 1: 905 | Nivel 2: 193 | Nivel 3: 461 |
| Lista: 5 | Nivel 1: 753 | Nivel 2: 125 | Nivel 3: 675 |
| Lista: 4 | Nivel 1: 634 | Nivel 2: 295 | Nivel 3: 452 |
| Lista: 8 | Nivel 1: 623 | Nivel 2: 105 | Nivel 3: 367 |
| Lista: 33 | Nivel 1: 423 | Nivel 2: 23 | Nivel 3: 346 |
| Lista: 1 | Nivel 1: 345 | Nivel 2: 163 | Nivel 3: 732 |
| Lista: 7 | Nivel 1: 324 | Nivel 2: 104 | Nivel 3: 235 |
| Lista: 1 | Nivel 1: 245 | Nivel 2: 582 | Nivel 3: 734 |
| Lista: 4 | Nivel 1: 235 | Nivel 2: 928 | Nivel 3: 245 |
| Lista: 8 | Nivel 1: 235 | Nivel 2: 592 | Nivel 3: 235 |

Femenino

| | | | |
|-----------|--------------|--------------|--------------|
| Lista: 6 | Nivel 1: 544 | Nivel 2: 195 | Nivel 3: 341 |
| Lista: 3 | Nivel 1: 523 | Nivel 2: 286 | Nivel 3: 426 |
| Lista: 10 | Nivel 1: 462 | Nivel 2: 184 | Nivel 3: 236 |
| Lista: 20 | Nivel 1: 431 | Nivel 2: 612 | Nivel 3: 724 |

Esta tabla refleja los mejores puntajes por género separándolos por masculino y femenino

LINKS:

Pagina funcionando: <https://elorduy50.pythonanywhere.com/>
github unity: <https://github.com/tc2005b-202411-441/team6-team6Unity>
github web: <https://github.com/tc2005b-202411-441/team6-team6Web>

Referencias:

Rangel Arranz, Á. (2022, July 1). Desarrollo de un videojuego Educativo en unity: Implementación Y Almacenamiento de la base de datos. Inicio. <https://oa.upm.es/73253/>

Metropolitana La Luz. (2023, September 12). <https://metropolitana-la-luz.webnode.mx/>

A free database designer for developers and analysts. Database Relationship Diagrams Design Tool. (n.d.). <https://dbdiagram.io/home>

Aportaciones:

Historial de versiones

Todas las versiones ▾

Hoy

15 de marzo, 3:26 p.m.

⋮

Versión actual

● Jorge Rodrigo Colín Rubio

▶ 15 de marzo, 1:21 p.m.

- Miguel Sebastián Reyes Moguel
- Sebastián Elorduy Martínez Manzanero
- Diego Samperio Arce
- Jorge Rodrigo Colín Rubio

▶ 15 de marzo, 12:27 p.m.

- Miguel Sebastián Reyes Moguel
- Jorge Rodrigo Colín Rubio
- Sebastián Elorduy Martínez Manzanero
- Diego Samperio Arce

15 de marzo, 12:10 p.m.

● Jorge Rodrigo Colín Rubio