

# TRABAJO DE FIN DE GRADO

## DESARROLLO DE APLICACIONES WEB



**LUCKYLOOP**

*Luckyloop es ese rincón donde el juego se siente como en casa. Aquí no solo apostamos, nos divertimos, aprendemos y sobre todo nos fiamos el uno del otro, ¿te unes?*

**Alumnos:** Darío Collar, Alejandro Cernada y Omar Daza

**Tutor:** Gustavo Millán

**Curso académico:** 2024-2025

**Centro:** IES Cañaveral

**Ciclo:** Desarrollo de Aplicaciones Web

**Fecha de presentación:** junio 2025

## RESUMEN PROYECTO

*Luckyloop es más que un casino, es el desarrollo completo de una plataforma de un casino online, implementando desde los juegos básicos hasta los sistemas de seguridad y gestión de usuarios. El objetivo es crear una experiencia de juego segura, atractiva y totalmente funcional que demuestre el dominio de las tecnologías web.*

## ÍNDICE

<b>1. INTRODUCCIÓN</b> .....	3
• 1.1. Descripción y contexto del proyecto .....	3
• 1.2. Motivación del proyecto .....	3
• 1.3. Beneficios esperados .....	3

<b>2. OBJETIVOS DEL PROYECTO .....</b>	<b>3</b>
<b>3. CONTEXTO ACTUAL .....</b>	<b>4</b>
<b>4. ANÁLISIS DE REQUISITOS .....</b>	<b>5</b>
• 4.1. Estado del arte .....	5
• 4.2. Conceptos clave .....	5
<b>5. DISEÑO DE LA APLICACIÓN .....</b>	<b>6</b>
• 5.1. Diagrama de casos de uso .....	6
• 5.2. Requisitos funcionales principales .....	6
• 5.3. Requisitos no funcionales .....	6
• 5.4. Descripción de los usuarios y sus necesidades .....	6
<b>6. DESARROLLO DE LA APLICACIÓN .....</b>	<b>7</b>
• 6.1. Mockups y prototipos .....	7
• 6.2. Arquitectura del sistema .....	7
• 6.3. Diagramas de clases y entidad-relación .....	7
• 6.4. Diseño de la base de datos .....	7
<b>7. PLANIFICACIÓN DEL PROYECTO .....</b>	<b>12</b>
• 7.1. Tecnologías y herramientas utilizadas .....	12
• 7.2. Descripción de las funcionalidades implementadas .....	12
<b>8. PRUEBAS Y VALIDACIÓN .....</b>	<b>13</b>
• 8.1. Acciones .....	13
• 8.2. Temporalización y secuenciación .....	13
<b>9. PRUEBAS Y VALIDACIONES .....</b>	<b>14</b>
<b>10. RELACION DEL PROYECTO CON LOS MODULOS .....</b>	<b>15</b>
<b>11. CONCLUSIONES .....</b>	<b>16</b>
<b>12. PROYECTOS FUTUROS .....</b>	<b>17</b>
<b>13. BIBLIOGRAFIA .....</b>	<b>18</b>
<b>14. ANEXOS .....</b>	<b>19</b>

## 1. INTRODUCCION:

En el mundo del entretenimiento los casinos online han experimentado una transformación y crecimiento a nivel online. Y este proyecto nace de la fascinación por crear experiencias que combinen ocio, entretenimiento, seguridad y la tecnología más moderna.

## 1.1 DESCRIPCION Y CONTEXTO DEL PROYECTO:

Consiste en el desarrollo de una plataforma de casino online, abarcando desde la implementación de juegos clásicos de toda la vida como ruleta, blackjack hasta creación de juegos actuales de los casinos convencionales hoy en día, ya que el gaming online ha crecido.

Además, creación de un sistema de gestión de usuarios, transacciones seguras, y administración de la plataforma.

Aportamos una plataforma buscando la experiencia digital y ocio disponible 24/7.

## 1.2 MOTIVACION DEL PROYECTO:

La motivación principal en este proyecto fue la necesidad de enfrentarse a un desafío técnico de alta complejidad que unía tanto un complejo sistema backend con gestión de usuarios, seguridad, fiabilidad de nuestro sistema, etc. Por otra parte, en el frontend una gran variedad y compleja visualización de cada juego, interfaces atractivas e interactivas, etc.

Además el proyecto nos permitió profundizar y aprender más de los casinos online actuales, ya que tuvimos que investigar cómo hacer muchos de los juegos que queríamos implementar en nuestro proyecto, así como el backend para estos, o para transacciones

## 1.3 BENEFICIOS ESPERADOS:

Podríamos decir varios beneficios que queríamos tener, pero a nivel educativo el dominio de tecnologías modernas y su implementación, comprensión de los procesos de desarrollo de un proyecto, experiencia en manejo de seguridad en proyectos. Por otra parte, a nivel profesional ojalá un posicionamiento gaming en la industria que hoy en día tiene una gran demanda, demostrar competencias de tener una capacidad de trabajar con tecnologías modernas. Y a nivel proyecto que se puede tener una gran confianza en cuanto seguridad de gestión de usuarios, password, y transacciones.

## 2. OBJETIVO/S GENERALES DEL PROYECTO:

Nuestro objetivo fue desarrollar una plataforma de un casino online que ofrezca una experiencia de juego seguro, y sea atractiva al usuario. Implementando tecnologías modernas e intentando mejores prácticas de desarrollo para crear un sistema confiable y escalable. Englobando la creación digital que no solo sea entretenimiento de calidad si no que demuestre un dominio de las tecnologías trabajadas. Por otra parte, somos estudiantes que nos gusta los desafíos complicados.

## 3. OBJETIVOS ESPECIFICOS:

Para alcanzar esos objetivos propuestos hemos desarrollado la siguiente estructura.

### 3.1 ANALISIS Y PLANIFICACION:

Analizar y definir los requisitos funcionales y no funcionales de nuestro sistema, identificando las necesidades de usuarios y estableciendo un desarrollo sostenible de nuestra plataforma.

Investigar y evaluar tecnologías más adecuadas para el desarrollo de este.

### 3.2 DISEÑO DEL SISTEMA:

Diseñar la arquitectura de entidad-relación y modelos de datos de nuestro proyecto para que este se vea de forma clara y eficiente todas las relaciones que nuestro proyecto contiene.

Desarrollar el diseño de la base de datos que sea optima, y represente de forma clara la integridad de datos y facilidad para realizar consultas entre las diferentes entidades de nuestro proyecto.

### 3.3 DESARROLLO DE LA INTERFAZ DE USUARIO:

Diseñar y desarrollar la interfaz de usuario que sea atractiva para estos y fácil de utilizar, que proporcione una experiencia fluida.

Implementar un diseño responsive que garantice la experiencia de dispositivos tanto de escritorio, como móviles.

Crear interfaces para cada tipo de juego que contiene nuestra plataforma manteniendo la coherencia, pero también adaptando a las necesidades de entretenimiento que queremos aportar.

### 3.4 DESARROLLO DEL BACKEND Y LOGICA DEL PROYECTO:

Implementar la lógica del casino desarrollando la parte de atrás, la parte “oculta” de nuestro proyecto.

Desarrollar un sistema de gestión de usuarios robusto que incluya registro y login, como autorización y privacidad.

Crear u sistema de gestión financiera dentro de nuestra plataforma con la que trabajamos con tecnologías nuevas, que suelen usar en proyectos grandes. En el cual contiene retiros, depósitos, saldos de usuario etc.

### 3.5 SEGURIDAD Y CALIDAD:

Implementar medidas de seguridad avanzadas con las cuales también hemos trabajado con otra tecnología moderna que suelen trabajar en pymes y empresas grandes. Previniendo SQL injection, XSS entre otros.

### 3.6 TESTING Y VALIDACION:

Realizar pruebas de nuestro sistema para garantizar que todas las funcionalidades operen correctamente.

Validar que la experiencia del usuario sea fácil y entretenida.

## 4. CONTEXTO ACTUAL:

### 4.1 ESTADO DEL ARTE:

El mercado de casinos online ha experimentado un gran crecimiento y ha dado lugar a varias plataformas en el sector.

Plataformas líderes como:

- Bet365, que se convirtió en una de las plataformas más completas del mercado, que incluye casino y deportes.
- 888Casino que ha llegado a implementar IA en su experiencia de usuario.
- PokerStars que revolucionó el póker teniendo funcionalidades a tiempo real.

Actualmente las plataformas existentes garantizan transparencia en los juegos, IA para determinados juegos etc.

### 4.2 CONCEPTOS CLAVE:

Hemos llegado a conclusión que los conceptos que tendría nuestro proyecto serían:

- Juegos y mecánicas de casino: juegos clásicos, incluyendo ruleta, blackjack, etc..., que requieren implementaciones para las probabilidades en cuanto juego.
- Sistema de apuestas y gestión financiera del usuario: manejando depósitos, retiros y el balance de fondos usuario y juegos.
- Futuras implementaciones de promociones por juegos, registros, días de recarga... etc.
- Experiencia de usuario e interfaz: diseño responsive en nuestra plataforma para que se pueda ver correctamente como en dispositivos de escritorio como móviles.
- Seguridad de nuestra plataforma: con sistema de autenticación y autorización con verificación de identidad de usuario y contraseña propia. Además de usar un sistema de autenticación con JWT.
- Juego responsable

## 5. ANALISIS DE REQUISITOS:

### 5.1 DIAGRAMA DE CASOS DE USO:

Diagrama uml casos de uso:

### 5.2 REQUISITOS FUNCIONALES PRINCIPALES: FUNCIONALIDADES QUE DEBE TENER LA APLICACION:

### 5.3 REQUISITOS NO FUNCIONALES: RENDIMIENTO, SEGURIDAD, USABILIDAD, ETC:

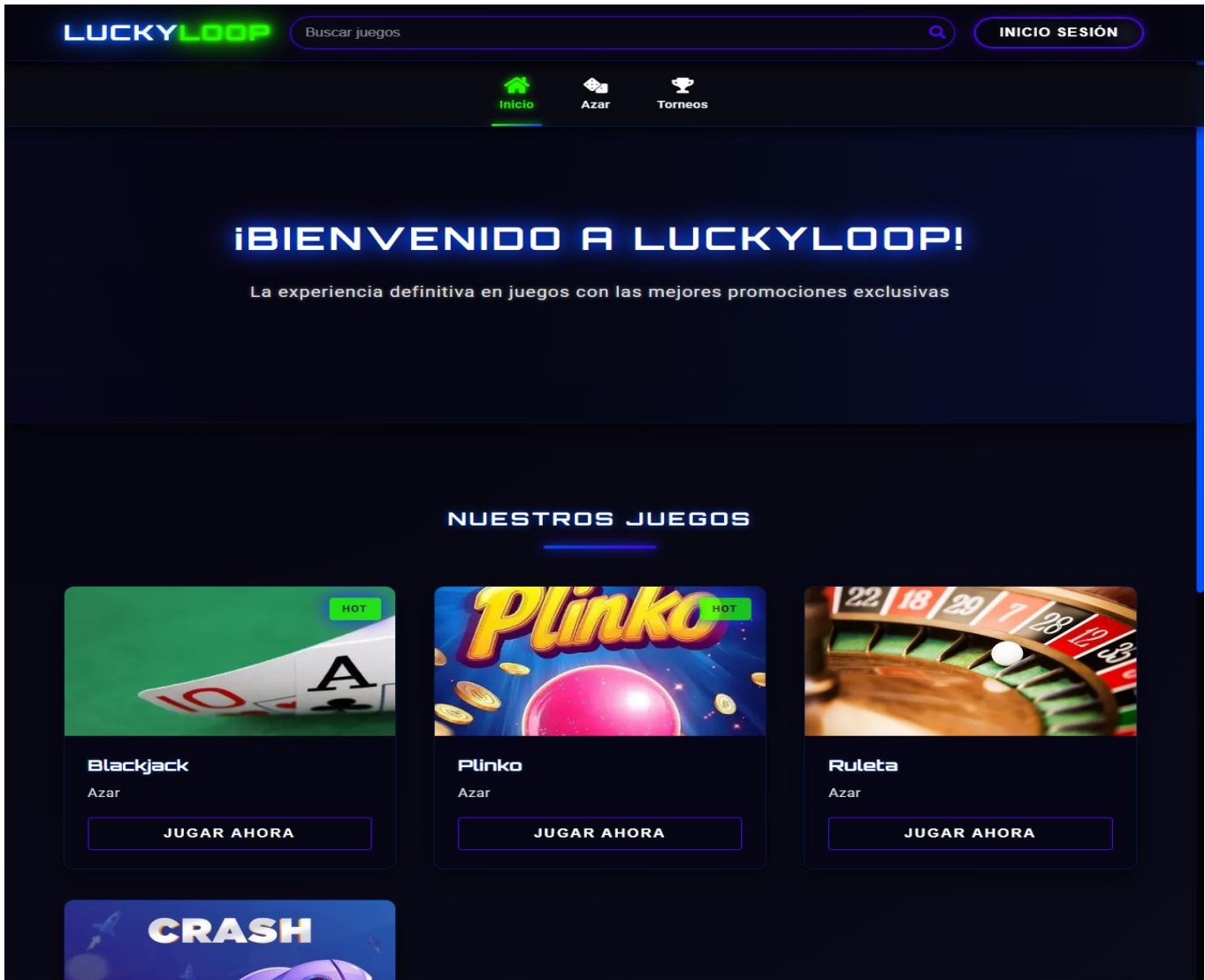
#### 5.4 DESCRIPCION DE LOS USUARIOS Y SUS NECESIDADES:

## 6. DISEÑO DE LA APLICACION:

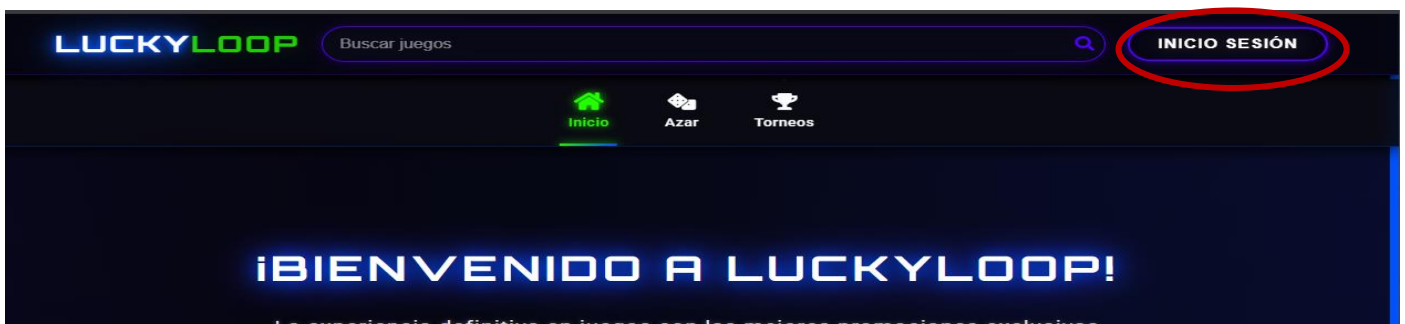
### 6.1 MOCKUPS O WIREFRAMES O PROTOTIPOS DE LA INTERFAZ GRAFICA DE USUARIO:

Diagrama general de nuestra plataforma seria:

- El menú principal de nuestra página al encontrarnos



- Al querer acceder a tu cuenta en nuestra plataforma a través del inicio de sesión



- Podrás iniciar sesión en nuestra plataforma siempre que tengas cuenta de usuario. En caso de no tener, tendrás que registrarte, siendo fácil e intuitivo.

## Iniciar Sesión

ENTRAR

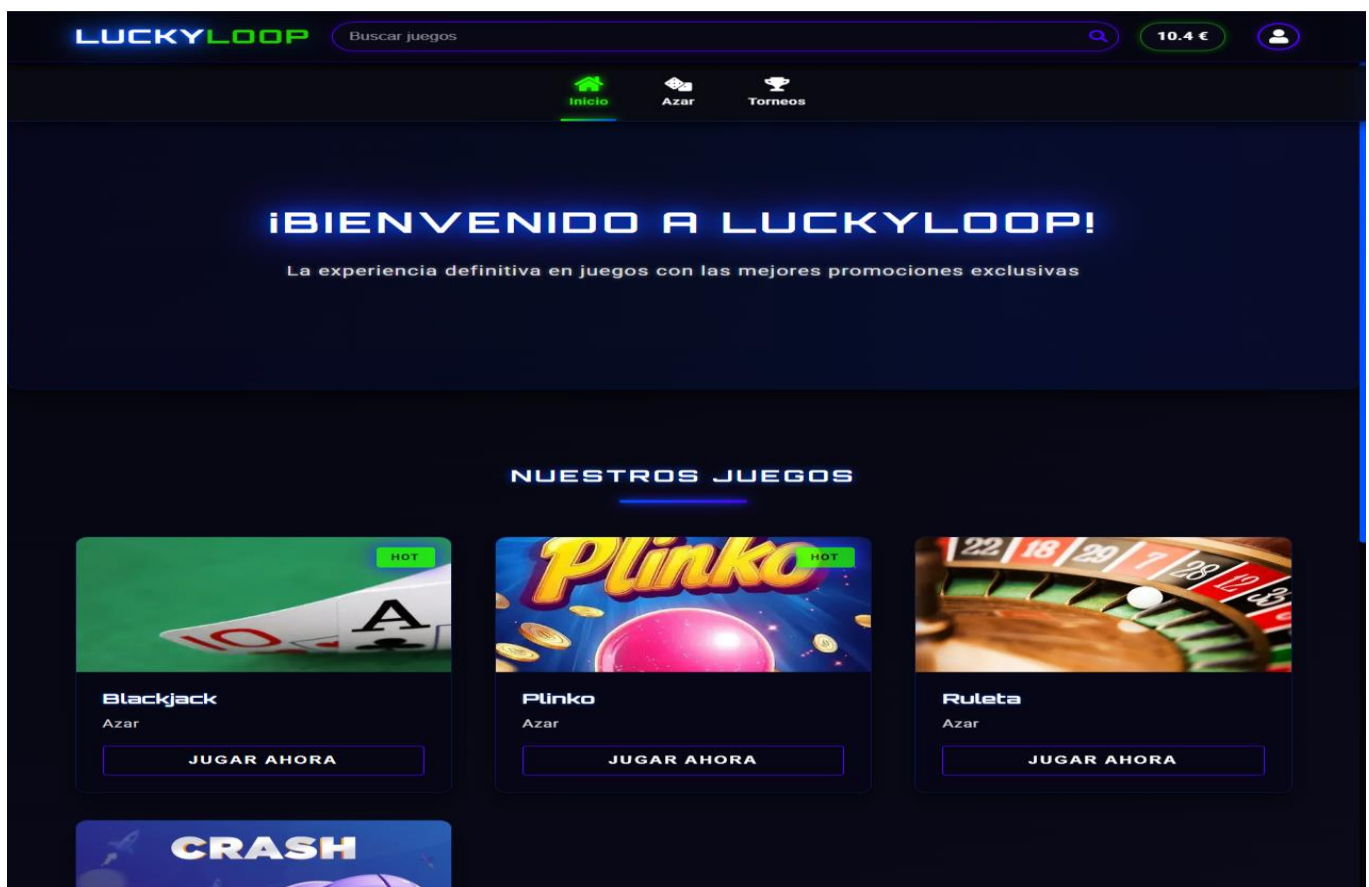
¿No tienes cuenta todavía? [Registrarse](#)  
[Olvidé mi contraseña](#)

## Registro

REGISTRARSE

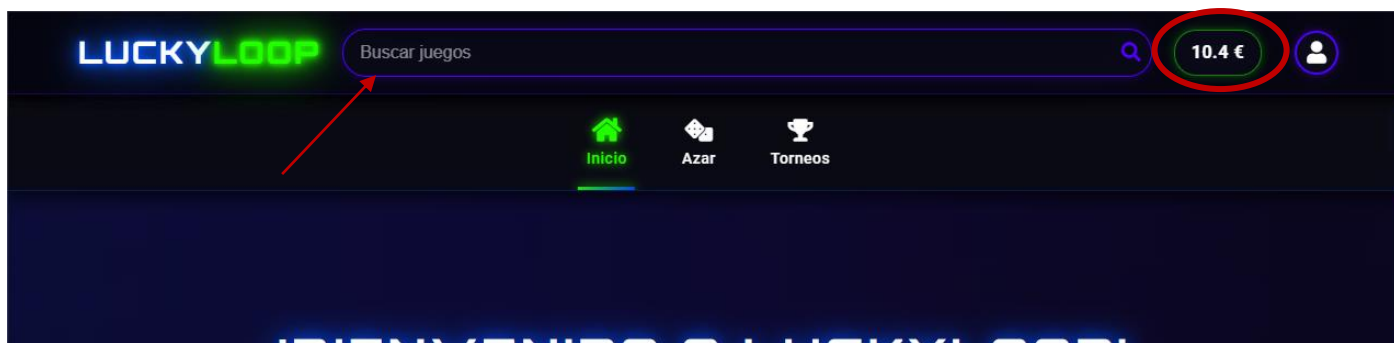
¿Ya tienes cuenta? [Iniciar sesión](#)

- Cuando inicies sesión nuestra plataforma tiene un menú principal hecho para ti con los juegos.

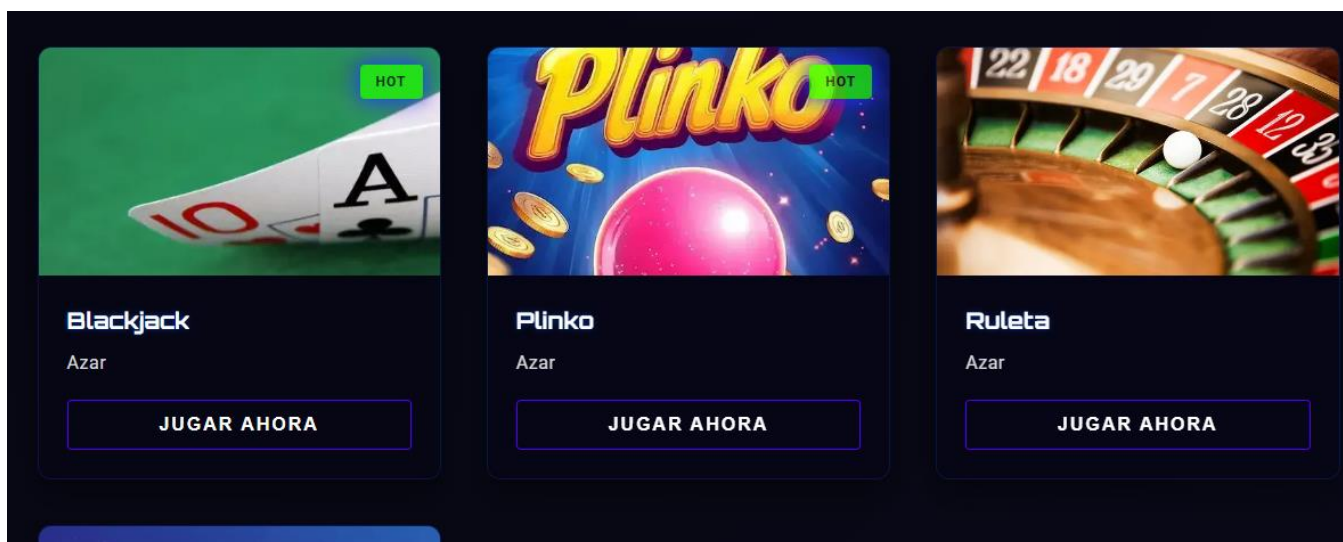


- La barra espaciadora podrás filtrar los juegos que quieras buscar, y tendrás tu balance como usuario (dinero depositado para jugar). Y en tu usuario tendrás varias opciones interesantes.

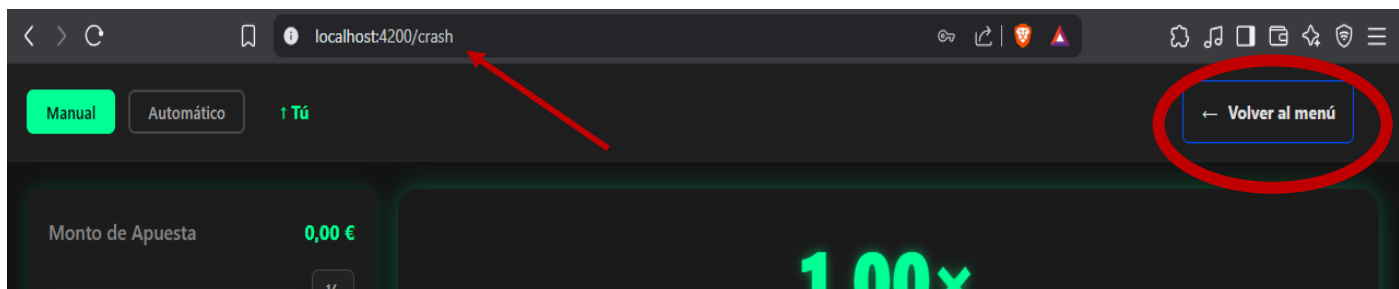




- Si el usuario quiere jugar a un juego determinado solo tiene que pulsar el botón de jugar y será redirigido al juego en específico



- Dentro del juego en específico podrás jugar siempre que tengas dinero depositado ya que este estará basado en tu balance personal de dinero. Y siempre podrás salir del juego, volver atrás al menú principal



## 6.2 ARQUITECTURA DEL SISTEMA:

Nuestro proyecto sigue la arquitectura cliente-servidor que contiene el MVC, el cual el front anteriormente comentado contendría html,css, js y ts hecho con framework de angular. Nuestro back estaría formado por una estructura de Symfony, un framework de php. Dentro de este, tenemos:

- entidades que representan las tablas de la base de datos y definen la estructura de nuestro back y maneja la aplicación.
- Controladores que se encargan de recibir las peticiones del cliente y procesarlas para devolverlas al front
- Servicios que serían las clases auxiliares que encapsulan la lógica del proyecto y mantiene todo limpio y reutilizable.

### LuckyLoop API 1.0.0 OAS 3.0

Documentación de la API del proyecto LuckyLoop

Authorize

#### Estadísticas

POST	/api/getEstadisticas	Obtiene las estadísticas del usuario actual		▼
POST	/api/getEstadisticas/juego/{id}	Obtiene las estadísticas de un juego específico		▼

#### Juego

GET	/api/getJuegos	Obtiene la lista de juegos disponibles		▼
GET	/api/getCategorias	Obtiene la lista de categorías disponibles		▼

#### Partida

POST	/api/finPartida	Inserta los datos de la partida		▼
------	-----------------	---------------------------------	--	---

#### Ranking

POST	/api/getRanking			▼
------	-----------------	--	--	---

#### Autenticación Autenticación

POST	/api/login	Obtener token JWT		▼
------	------------	-------------------	--	---

#### Usuario Usuario

POST	/api/usuario/emailToken	Genera un token a un usuario si su email existe y le envía un email		▼
GET	/api/usuario/comprobarToken/{token}	Comprueba si el token del usuario es válido		▼
POST	/api/usuario/cambiarPassword	Cambia la contraseña del usuario		▼
POST	/api/usuario/registrarse	Registra un nuevo usuario		▼
GET	/api/usuario/getSaldo	Obtener el saldo actual de un usuario por ID		▼
POST	/api/usuario/updateSaldo	Resta o suma saldo a un usuario		▼
POST	/api/usuario/cambiarNombre	Cambiar el nombre del usuario		▼

La imagen anterior contiene todo nuestro api/doc que contiene como manejamos las APIs en nuestro proyecto, las cuales están en back, y que las recoge después el front. Además, nos muestra cómo funciona cada endpoint, ejemplo una de las APIs y muestra lo que devuelven.

## Juego

GET /api/getJuegos Obtiene la lista de juegos disponibles

Devuelve todos los datos de los juegos disponibles en el sistema

### Parameters

Try it out

No parameters

### Responses

Code	Description	Links
200	<p>Lista de juegos devuelta correctamente</p> <p>Media type application/json</p> <p>Controls Accept header.</p> <p>Example Value   Schema</p> <pre>{   "status": "success",   "data": [     {       "id": 1,       "name": "BlackJack",       "image": "assets/images/blackjack.webp",       "category": "Juego de Azar",       "isHot": true,       "url": "/blackjack"     }   ] }</pre>	No links
404	<p>No se encontraron juegos</p> <p>Media type application/json</p> <p>Example Value   Schema</p> <pre>{   "message": "No se encontraron juegos" }</pre>	No links

▼ LuckyLoop

- ▼ BACKEND
  - > bin
  - ▼ config
    - ▼ jwt
      - private.pem
      - public.pem
    - > packages
    - > routes
      - bundles.php
      - preload.php
      - routes.yaml
      - services.yaml
    - > migrations
    - > public
    - ▼ src
      - ▼ AuthBundle
        - > Controller
        - > Services
          - AuthBundle.php
      - > Controller
      - > Entity
      - > EstadisticasBundle
      - > JuegoBundle
      - > PartidaBundle
      - > RankingBundle
      - > Repository
      - > Service

Y la estructura de nuestro backend estructurado con sus diferentes entidades, controllers, paquetes, repositorio y configuraciones que en ellas está el JWT más adelante explicado.

## 6.3 DIAGRAMA DE CLASES Y ENTIDAD-RELACION:

