

Tecnológico de Costa Rica
Escuela de Ingeniería en Computación



Proyecto IV “luiKi Kart”

Lenguajes de Programación

Profesor:
Rodríguez Dávila, Allan

Presentado por:
Bonilla Espinoza, Alina [2016248502]
Lara Oses, Froylan [2018276191]

Noviembre, 2022

Tabla de Contenido

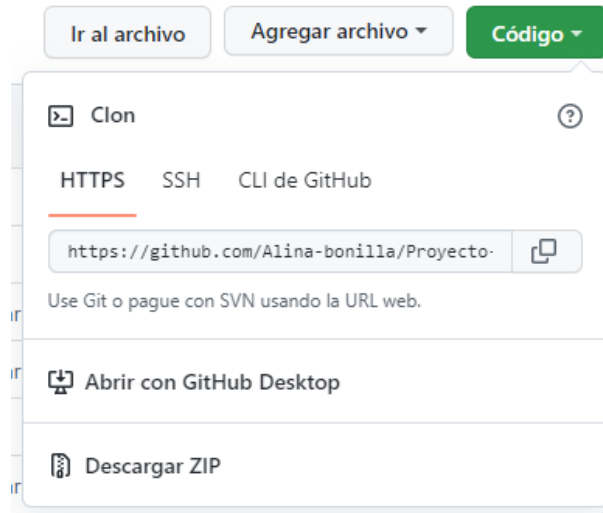
Manual de usuario.....	3
1. Descargar el proyecto:	3
2. Compilación:.....	3
3. Ejecución el proyecto:	3
4. Autenticación:	4
5. Cliente:	4
6. Crear Partida	5
7. Ver estadisticos	6
8. Ver ranking	6
Pruebas de funcionalidad.....	7
1. Autenticación	7
2. Cliente	7
3. Crear Partida	8
4. Vencimiento de partida	9
5. Ver Ranking.....	9
6. Ver estadisticos	10
7. Unirse a una partida	10
Descripción del problema	11
Diseño del programa	12
Librerías usadas.....	12
Análisis de resultados	13
Bitácora.....	13

Manual de usuario

A continuación, se presentan los pasos para hacer uso correcto del programa:

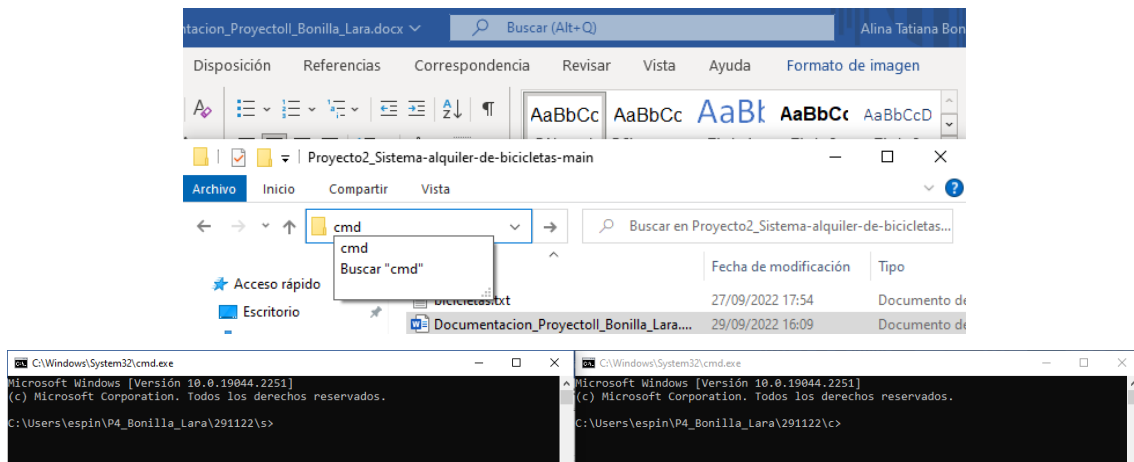
1. Descargar el proyecto:

Se debe ingresar al repositorio git y descargar el **.zip** con el proyecto. Se debe seleccionar la pestaña **código** y posteriormente hacer click en **Descargar ZIP**. Después se debe descomprimir el **.zip**



2. Compilación:

Ingresa a la carpeta del proyecto, en cada una de las subcarpetas (cliente, servidor), realice la siguiente acción: en la dirección coloque **"cmd"** y presione enter. Se abrirá una ventana del cmd.



3. Ejecución el proyecto:

Seguidamente, ingrese el siguiente comando en la ventana del cmd servidor **"node index.js"** y presione enter, se desplegará el siguiente mensaje.

```
C:\Windows\System32\cmd.exe - node index.js

C:\Users\espin\P4_Bonilla_Lara\291122\s>node index.js
Server started on 3001
```

Seguidamente, ingrese el siguiente comando en la ventana del cmd cliente “npm start” y presione enter, se desplegará el siguiente mensaje.

```
Windows PowerShell

Compiled successfully!

You can now view cliente in the browser.

Local:      http://localhost:3000
On Your Network:  http://192.168.56.1:3000

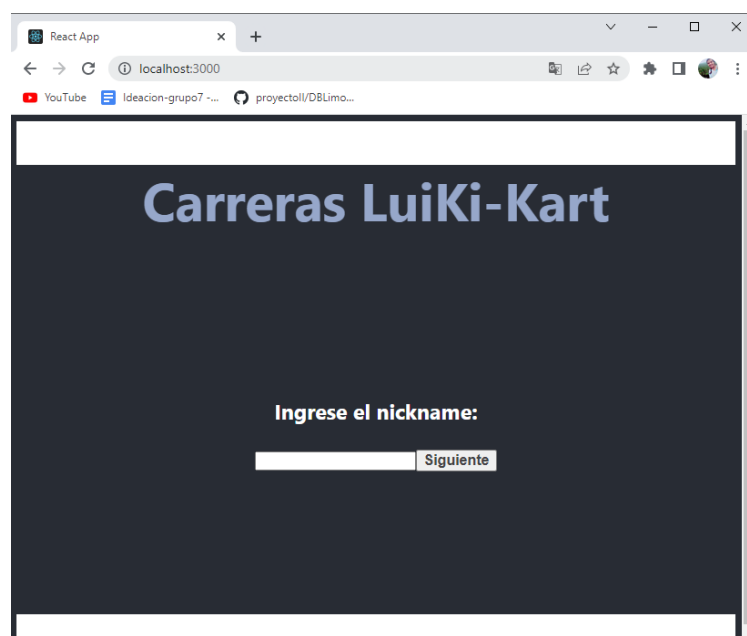
Note that the development build is not optimized.
To create a production build, use npm run build.

webpack compiled successfully
```

4. Autenticación:

Seguidamente, se le abrirá en su navegador predeterminado la página principal del proyecto.

Acá podrá ingresar su nombre de nickname para la partida a realizar.



5. Cliente:

El usuario pueda elegir entre las siguientes opciones:

- a) Crear partida

b) Unirse a una partida (jugar)

c) Ver ranking

Ver estadísticas

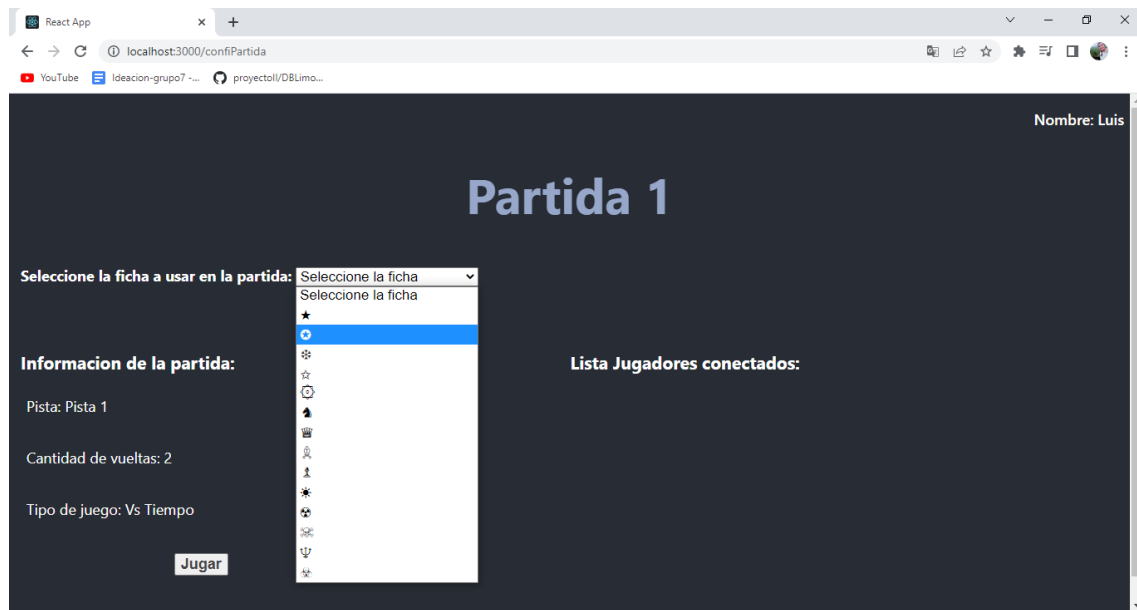


6. Crear Partida

Se le solicitará al usuario seleccionar la pista, cantidad de vueltas y modo de juego. Seguidamente deberá dar click en el botón de “Crear Partida”.



Seguidamente, se le solicitara seleccionar una ficha de juego



7. Ver estadisticos

Por cada juego se debe indicar lo siguiente:

- Nombre de jugadores y su tiempo o posición, ordenado ascendentemente
- Pista
- Identificador de partida

8. Ver ranking

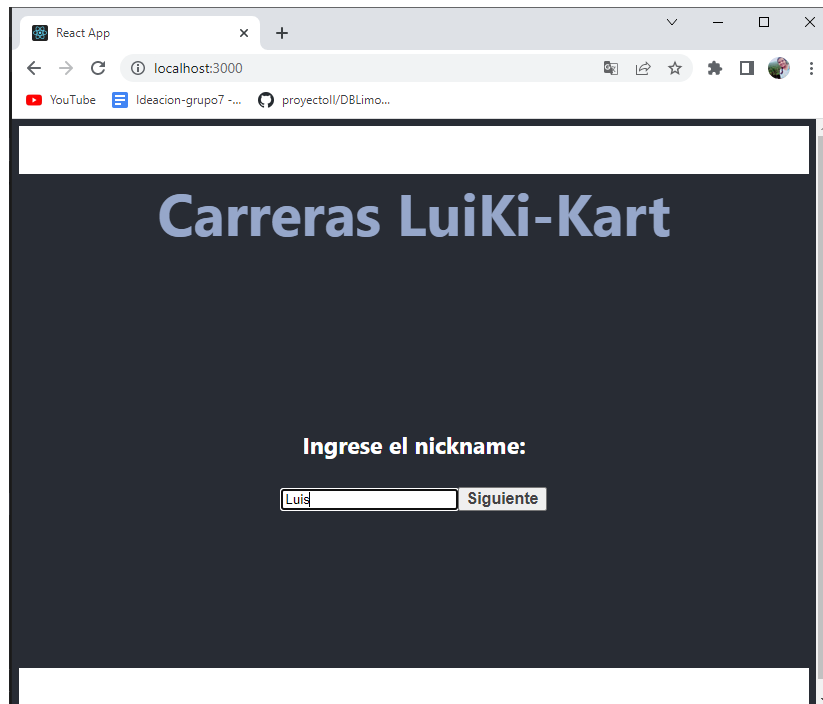
Se muestra la información de los ganadores de cada partida.

- Nombre del ganador
- Tiempo
- Pista
- Vueltas
- Identificador de partida

Pruebas de funcionalidad

1. Autenticación

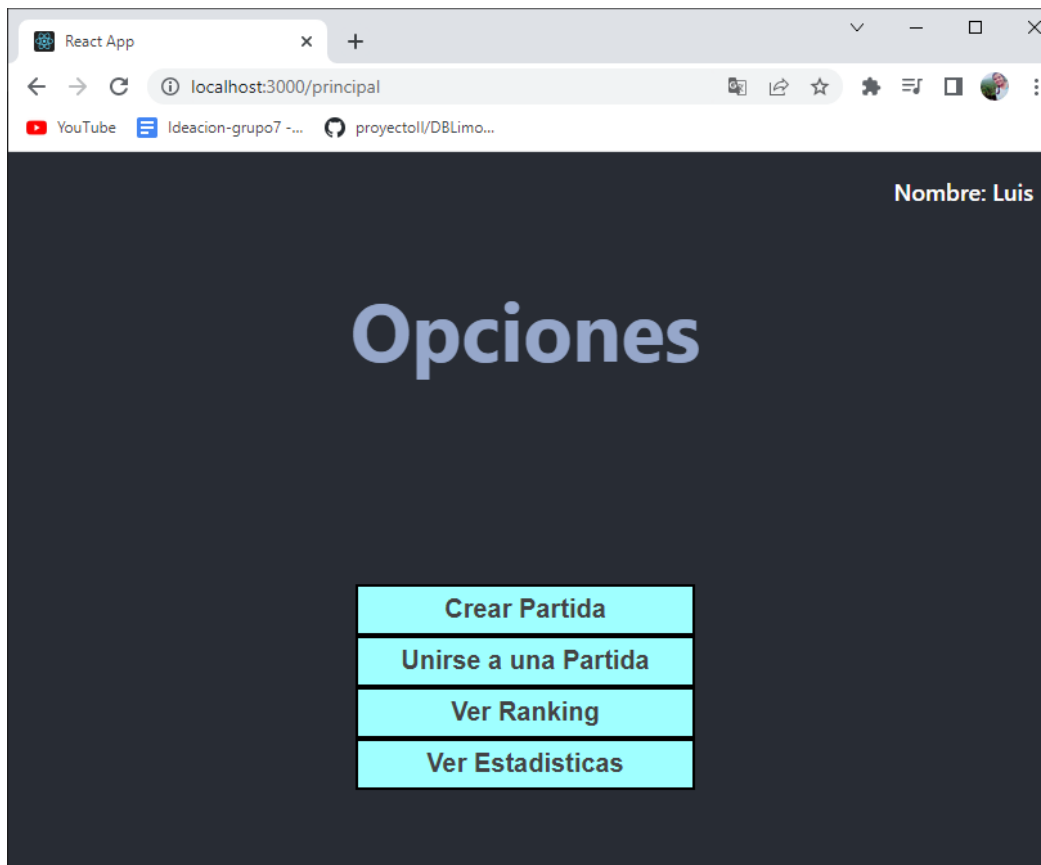
El sistema le solicita y guarda el nombre de nickname ingresado por el usuario para ser identificado durante la partida bajo dicho nombre.



The screenshot shows a web browser window with the title 'React App' and the address bar displaying 'localhost:3000'. The browser's address bar also shows several tabs: 'YouTube', 'Ideacion-grupo7 - ...', and 'proyectoII/DBLimo...'. The main content of the page is a dark blue rectangle with the text 'Carreras LuiKi-Kart' in a large, bold, white font. Below this, the text 'Ingresa el nickname:' is displayed in a smaller white font. Underneath, there is a text input field with the value 'Luis' and a button labeled 'Siguiente'.

2. Cliente

El usuario podrá escoger cuál de las siguientes opciones desea realizar y posteriormente se le desplegará una nueva ventana según su elección.



3. Crear Partida

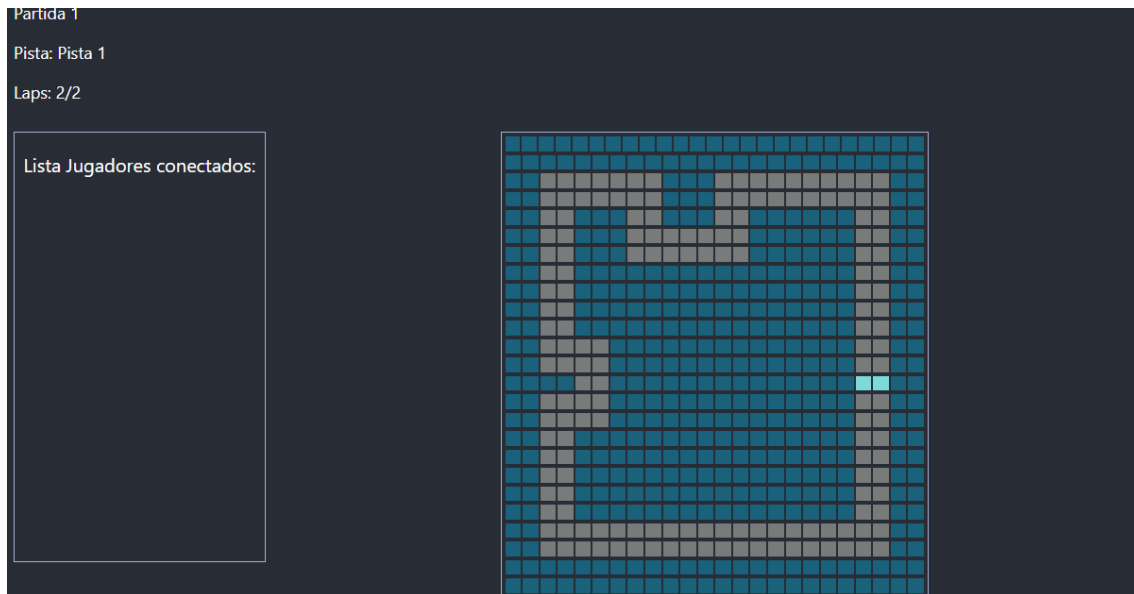
El sistema guarda todos los datos correspondientes a la partida en la base de datos del sistema.

```
Select * from Partida;  
Select * from Jugador;
```

100 %

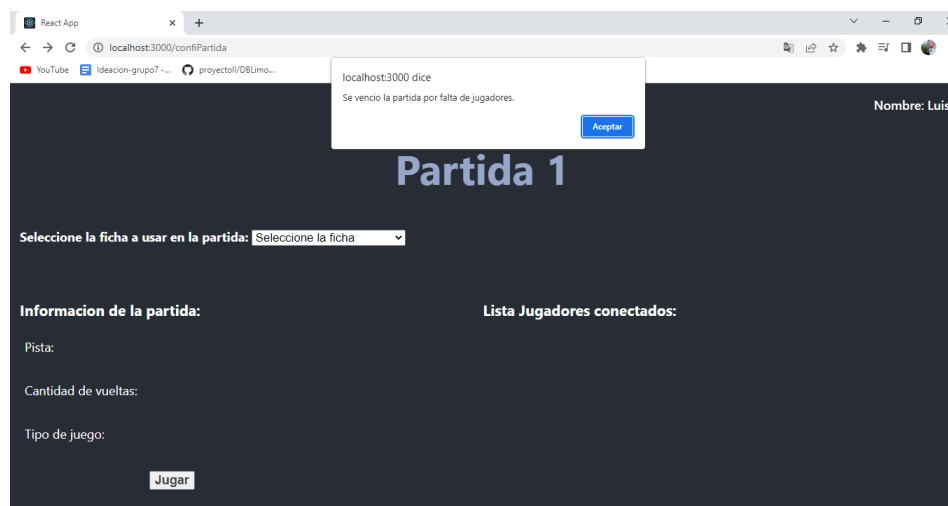
Results Messages

	IdPartida	Pista	Vueltas	TipoPartida
1	1	Pista 1	2	1



4. Vencimiento de partida

Si una partida supera los 3 minutos y posee menos de 2 jugadores, esta será cerrada por el sistema, ya que no cumple con los requerimientos.



5. Ver Ranking

Se le mostrara en pantalla la información de los ganadores de cada partida.



6. Ver estadisticos

Por cada juego se debe indicar lo siguiente:

- Nombre de jugadores y su tiempo o posición, ordenado ascendentemente
- Pista
- Identificador de partida

Nombre: Luis

Ver Estadisticas

Estadisticas por juego

Posicion	Tiempo	Nickname	Pista	Id Partida	Tipo de Partida
1	00:03:00	Fernanda	Pista1	2	2
1	00:04:50	Marcos	Pista 1	1	1
2	00:04:20	Felipe	Pista 1	1	1
4	00:04:50	Antonio	Pista1	2	2

7. Unirse a una partida

Se solicita la ficha y el identificador de la partida a la que desea unirse.

Nombre: Luis

Unirse a Partida

Seleccione el ID de la partida deseada:

Informacion de la partida:

Pista: Pista 1

Cantidad de vueltas: 2

Tipo de juego: Vs Tiempo

Lista Jugadores conectados:

1. Luis

Descripción del problema

Los juegos de carreras han sido desarrollados y distribuidos por diferentes empresas, y estos cuentan con la aparición de personajes, que pueden ser adaptados a los gustos de los jugadores. A lo largo de la historia de los video juegos, los juegos de carreras han sido de gusto de los gamers, que utilizan diferentes consolas.

Para nuestro caso, el plano del juego será una matriz, que deberá tomarse de archivos (.txt, .csv, .xml) y generar la representación de las pistas de juego y se deben generar las estructuras correspondientes para el desarrollo de la funcionalidad por medio de una interfaz web.

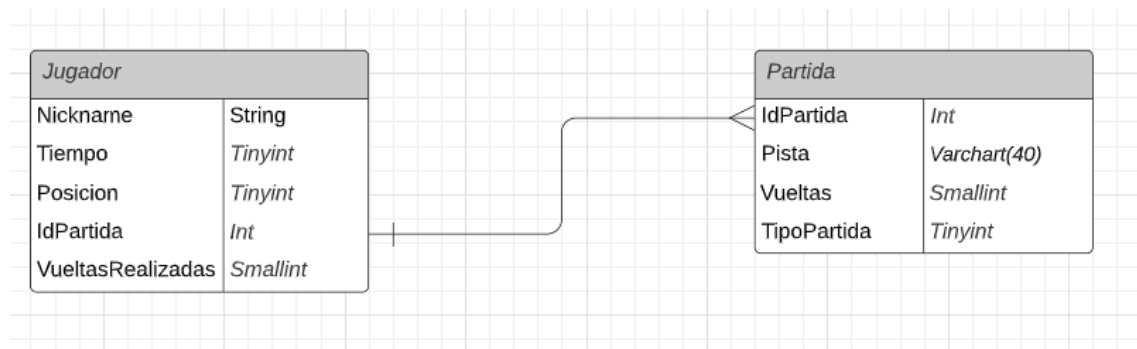
El objetivo general de este proyecto es desarrollar una mejor herramienta Web para el desarrollo de un juego en línea desde la perspectiva de la programación orientada a objetos, la investigación de estrategias de manejo de concurrencia y multiusuarios; de manera que las partes de la lógica y datos (Backend) que lo enmarcan sean diseccionadas minuciosamente para desarrollar una solución web (Frontend). El Backend deberá desarrollarse con .Net Core (C#) o Node js (Express) aplicando orientación a objetos y el Frontend deberá desarrollarse con React, Vue.js, Angular o Net –Asp.Net– (manteniendo la separación de proyectos).

Puede utilizar un motor de base de datos libre para mantener la información persistente. La información de configuración debe estar en un archivo específico para esta finalidad. El servidor “gaming” (Backend) y el Frontend se publicarán por medio de las herramientas Ngrok, Localtunnel o similar para que puedan ser accesadas por medio de enlaces públicos.

Diseño del programa

Para la creación de este proyecto se hizo uso de una serie de archivos de tipo .txt para almacenar las pistas y una base de datos en SQL para mantener la permanencia de datos. Para el manejo de los archivos del sistema (persistencia de datos) se utilizó el formato presentado en el enunciado del proyecto.

Para la elaboración de la base de datos en SQL, el diseño de esta es el siguiente:



Librerías usadas

Para la ejecución de este proyecto se hizo de:

- Node.js: Para la elaboración del Backend.
- Frontend: Para la elaboración del Frontend.
- Ngrok: Enlaces públicos
- Express: Para la conexión entre la base de datos (sql) y node.js.

Análisis de resultados

Objetivos alcanzados:

- Autenticación
- Creación de partidas
- Creación de jugadores
- Unirse a partida
- Ver ranking
- Ver estadísticos

Objetivos no alcanzados:

- Publicación de enlace publico

Bitácora

