

Instituto Tecnológico de Costa Rica

Lenguajes de Programación

Damas Chinas

PY #04

Integrantes:

Bayron Rodriguez Centeno - 2020114659

Gadir Calderón Díaz - 2022327328

Profesor:

Allan Rodríguez Dávila

Grupo 60

II Semestre

Año 2024

Tabla de contenidos

Tabla de contenidos.....	2
Manual de Usuario.....	3
1. Instrucciones de Instalación.....	3
a. Requisitos previos.....	3
b. Configuración del Backend.....	3
c. Configuración del frontend.....	3
2. Uso.....	3
3. Pruebas de Funcionalidad.....	3
Descripción del problema.....	3
Diseño del programa.....	4
1. Diagrama de paquetes.....	4
2. Diagrama de distribución.....	5
3. Diagrama de comunicación.....	6
Librerías usadas.....	6
Frontend.....	6
Backend.....	7
Comunicación.....	8
Eventos Socket.io.....	8
API REST.....	8
Análisis de Resultados.....	9
Bitácora.....	9

Manual de Usuario

1. Instrucciones de Instalación

a. Requisitos previos

- i. Node.js (v14.0.0 o superior)
- ii. npm (v6.0.0 o superior)
- iii. MongoDB (v4.0.0 o superior)

b. Configuración del Backend

- i. Navegar hasta el directorio del backend
- ii. Instalar las dependencias
- iii. Crear el archivo .env en la raíz del backend
- iv. Iniciar el servidor

c. Configuración del frontend

- i. Navegar hasta el directorio del frontend
- ii. Instalar las dependencias
- iii. Iniciar la aplicación

2. Pruebas de Funcionalidad

a. Inicio



Damas Chinas

jess

Crear Partida ID de la partida Unirse

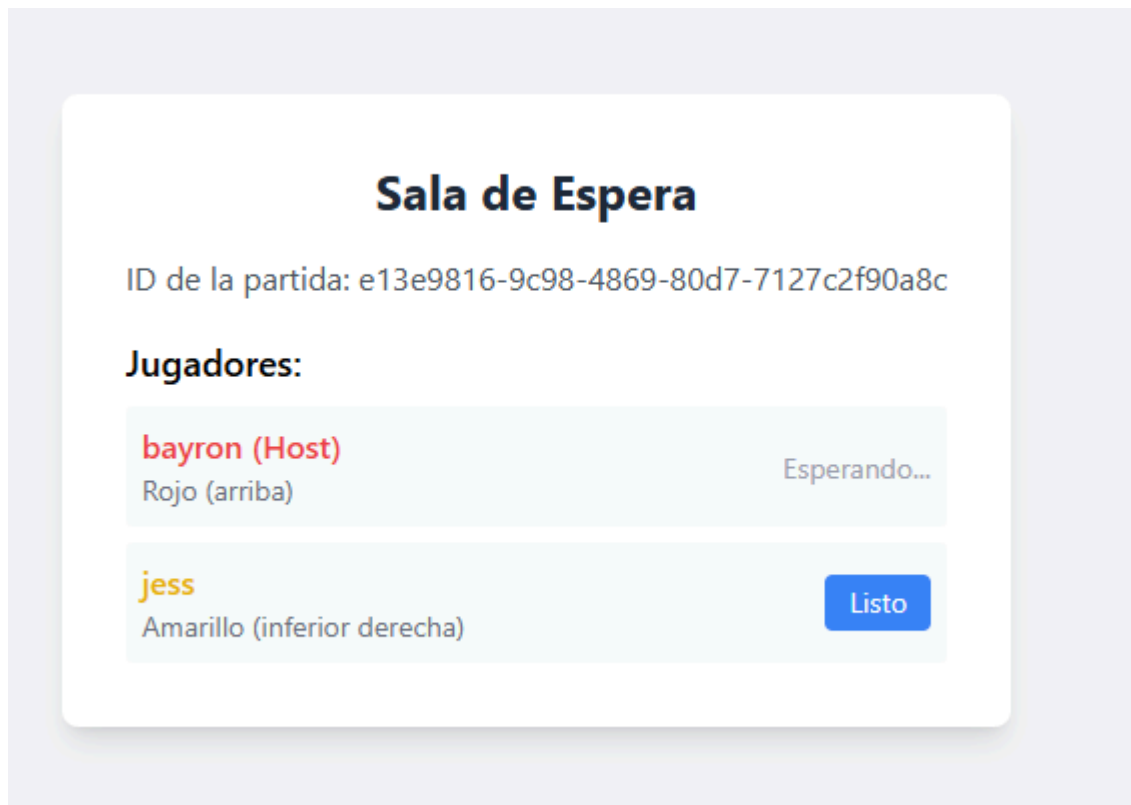
b. Sala de espera



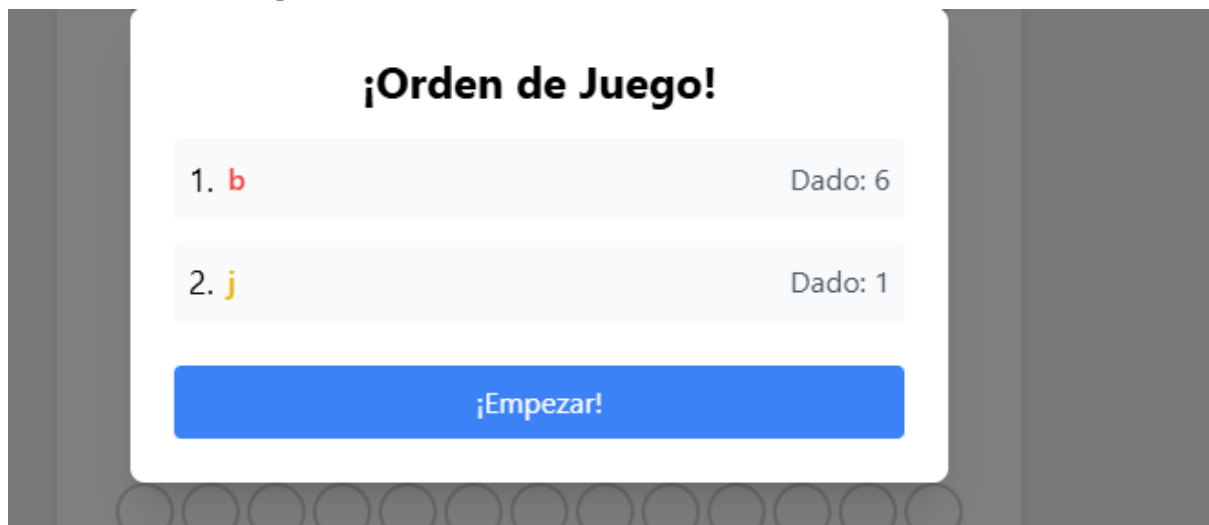
c. Unirse a la partida(host)



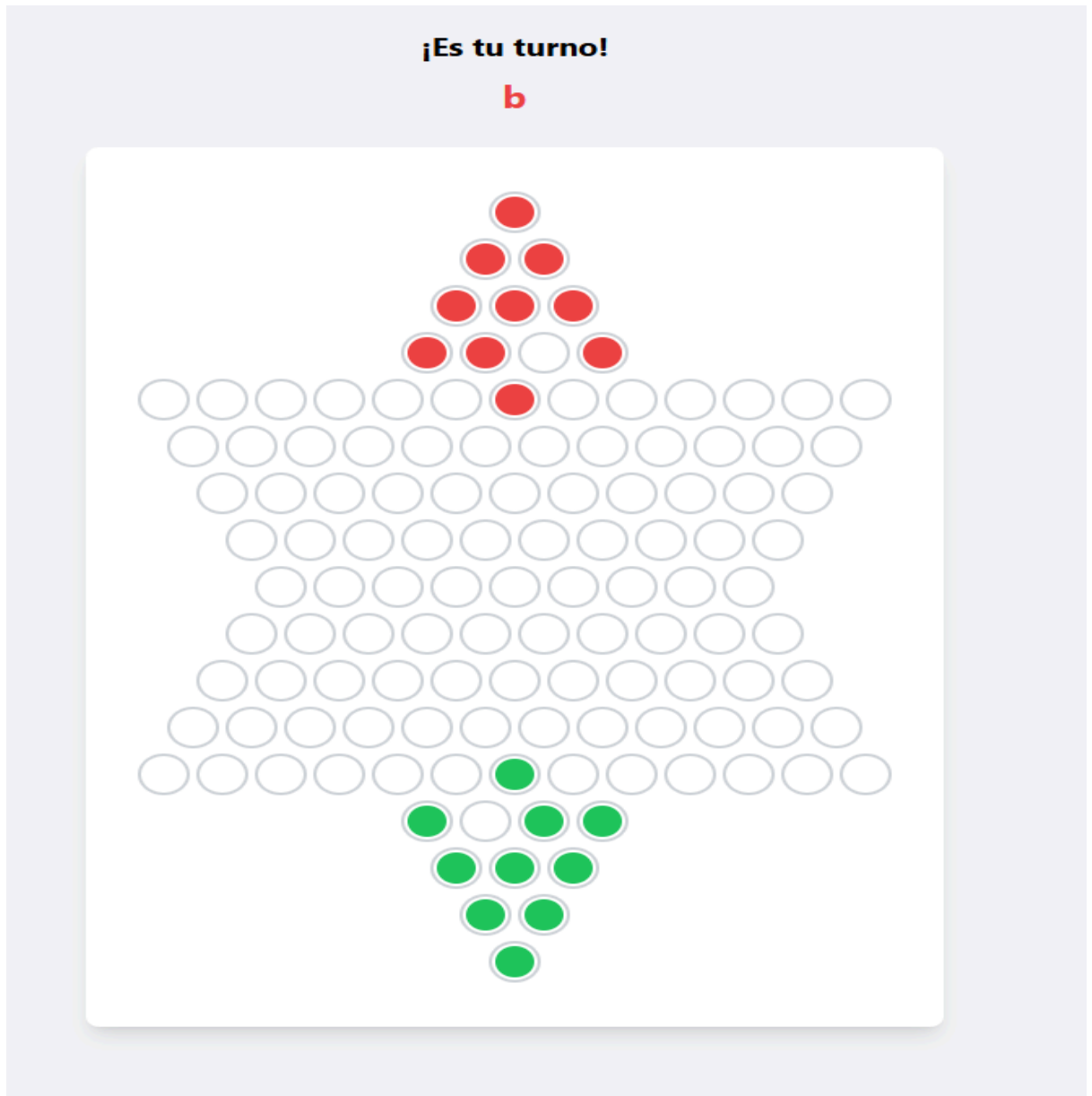
d. Unirse a partida(invitado)



e. Orden por dado



f. Partida(dos jugadores)



Descripción del problema

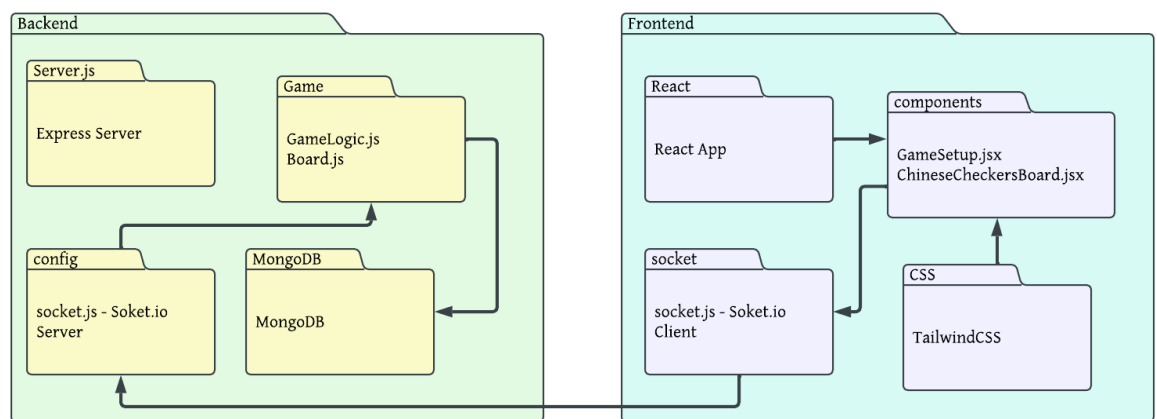
Los juegos de mesa han sido desarrollados y distribuidos por diferentes empresas, y estos cuentan con diferentes variantes, que pueden ser adaptados a los gustos de los jugadores y diferentes regiones. A lo largo de la historia de los videojuegos, los juegos de mesa han sido de gusto de las personas que requieren espacios de ocio. Para nuestro caso, el plano del juego será una estructura con algunas características matriciales y se deben generar las estructuras

correspondientes para adaptarlas al desarrollo de la funcionalidad por medio de una interfaz web.

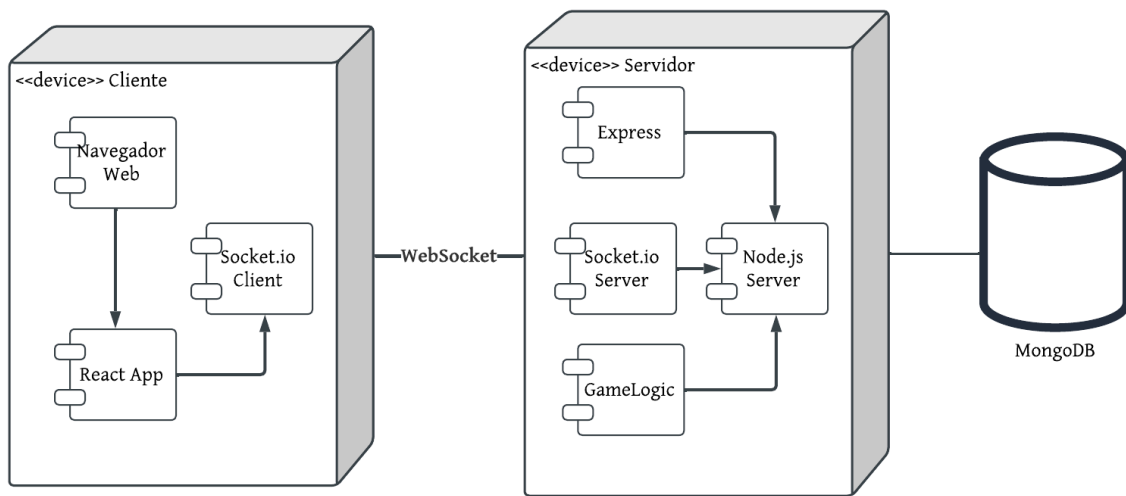
Las Damas Chinas es un juego de mesa de estrategia de origen alemán para dos, tres, cuatro o seis jugadores. El juego es una variante moderna y simplificada del juego estadounidense Halma.

Diseño del programa

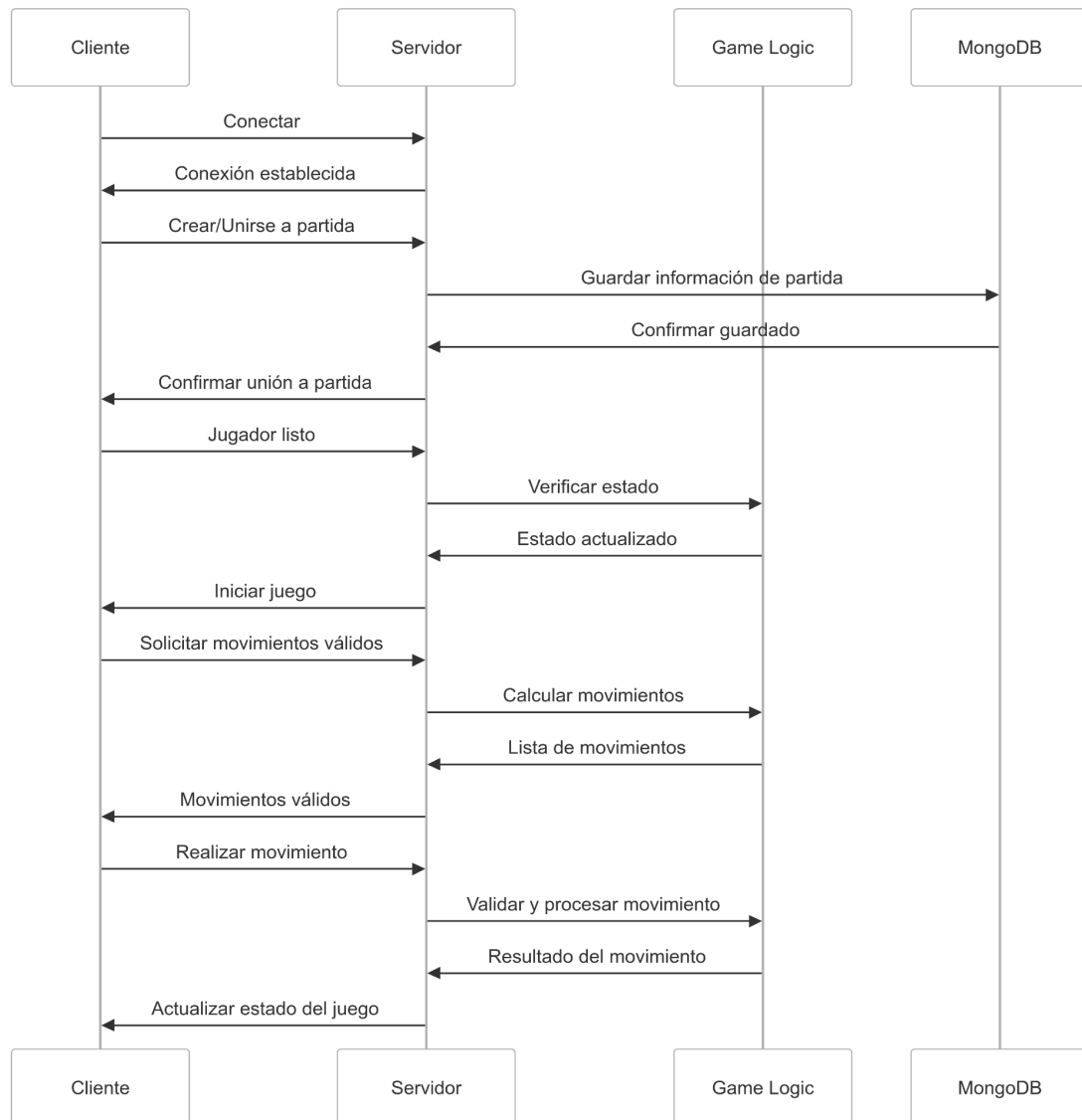
1. Diagrama de paquetes



2. Diagrama de distribución



3. Diagrama de comunicación



Librerías usadas

Frontend

1. React (v18.2.0)
 - a. Framework principal para la interfaz de usuario
 - b. Manejo de estado y componentes

2. Socket.io-client (v4.7.2)
 - a. Comunicación en tiempo real con el servidor
 - b. Manejo de eventos del juego
3. TailwindCSS (v3.3.0)
 - a. Framework CSS para estilos
 - b. Diseño responsive y componentes
4. React Router Dom (v6.8.2)
 - a. Manejo de rutas en la aplicación
 - b. Navegación entre componentes

Backend

1. **Express (v4.18.2)**
 - a. Framework web para Node.js
 - b. Manejo de rutas y middleware
2. **Socket.io (v4.7.2)**
 - a. Servidor WebSocket
 - b. Comunicación bidireccional en tiempo real
3. **Mongoose (v7.0.3)**
 - a. ODM para MongoDB
 - b. Modelado de datos y operaciones CRUD
4. **Cors (v2.8.5)**
 - a. Middleware para manejo de CORS
 - b. Seguridad en comunicación cross-origin
5. **Dotenv (v16.0.3)**
 - a. Manejo de variables de entorno

b. Configuración del entorno

Comunicación

Eventos Socket.io

1. Conexión y Gestión de Salas

- a. connection: Conexión inicial del cliente
- b. joinGame: Unirse a una sala de juego
- c. playerReady: Jugador listo para comenzar
- d. disconnect: Desconexión del cliente

2. Juego

- a. gameStarted: Inicio del juego
- b. getValidMoves: Solicitud de movimientos válidos
- c. validMoves: Respuesta con movimientos válidos
- d. movePiece: Realizar movimiento
- e. pieceMove: Notificación de movimiento realizado
- f. turnChange: Cambio de turno
- g. gameOver: Fin del juego

API REST

1. Gestión de Juegos

POST /api/games/create //crea nueva partida

POST /api/games/join //unirse a partida existente

POST /api/games/:id //Obtener información de partida

2. Gestión de Jugadores

POST /api/players/register // Registrar nuevo jugador

GET /api/players/:id // Obtener información de jugador

Análisis de Resultados

Objetivo	Alcanzado	No alcanzado	¿Por qué?
Autenticación	✓		
Crear Partida	✓		
Unirse a partida	✓		
Área de Juego	✓		
Turnos del juego	✓		
Validación de gane	✓		
Ranking			

Bitácora

https://github.com/Bayronjrc/PP4_Bayron_Gadir.git