Documentación del Proyecto: Sistema de Partidas de Ajedrez con DDD

# 1. Introducción

Este proyecto implementa un sistema de partidas de ajedrez aplicando los principios de Domain-Driven Design (DDD). Está construido con Spring Boot, siguiendo una arquitectura limpia que separa claramente el dominio, la aplicación y la infraestructura.

# 2. Análisis del Dominio

## 2.1 Conceptos Clave

- PartidaDeAjedrez: Agregado raíz que orquesta el comportamiento del juego.  
- Jugador: Entidad con identidad única (UUID), color y nombre.  
- Pieza: Objeto compuesto por tipo y color, sin identidad.  
- TipoPieza: Enum que representa las reglas de movimiento de cada pieza.  
- Tablero: Objeto que contiene el estado del juego, incluyendo casillas y piezas.  
- Coordenada: Objeto de valor que representa una posición en el tablero.  
- Color: Enum para diferenciar entre BLANCO y NEGRO.

## 2.2 Reglas de Negocio Modeladas

- Validación de movimientos por tipo de pieza.  
- Control de turnos (alternancia entre jugadores).  
- Detección de jaque y jaque mate.  
- Finalización automática de partida en caso de mate.

# 3. Estructura del Proyecto

El proyecto está organizado en las siguientes capas:  
  
- domain/model: Contiene todas las entidades, objetos de valor y lógica del dominio.  
- application: Casos de uso como crear partida y mover pieza.  
- infrastructure/rest: Controladores HTTP para exponer la API.  
- resources: Configuración del proyecto.

# 4. Uso de la API

## 4.1 Crear una Partida

POST http://localhost:8080/partidas  
  
Parámetros:  
- jugadorBlanco: nombre del jugador blanco  
- jugadorNegro: nombre del jugador negro  
  
Ejemplo:  
http://localhost:8080/partidas?jugadorBlanco=Alice&jugadorNegro=Bob

## 4.2 Mover una Pieza

PUT http://localhost:8080/partidas/{id}/mover  
  
Parámetros:  
- colOrigen: columna (ej. e)  
- filaOrigen: fila (ej. 2)  
- colDestino: columna de destino  
- filaDestino: fila de destino  
  
Ejemplo:  
http://localhost:8080/partidas/{id}/mover?colOrigen=e&filaOrigen=2&colDestino=e&filaDestino=4

# 5. Ejemplos para Probar

- Crear partida con nombres: Alice (blanco) y Bob (negro).  
- Mover Reina Blanca de d1 a h5 (si disponible).  
- Mover Peón Negro de e7 a e5.  
- Mover Reina Blanca a e8 para hacer jaque mate (si es posible según el setup).  
- Consultar que la partida está finalizada tras el mate.

# 6. Conclusión

Este proyecto demuestra cómo aplicar DDD de manera práctica en un dominio complejo como el ajedrez. La estructura modular permite evolucionar la aplicación agregando reglas como promoción de peón, enroque y captura al paso, manteniendo la lógica centralizada en el modelo del dominio.

# 7. Ejemplo de Partida Completa Usando Endpoints (10 movimientos)

A continuación se muestra un ejemplo completo de cómo crear una partida y realizar al menos 10 movimientos válidos usando la API REST del sistema.

## 7.1 Crear la Partida

Endpoint:  
POST http://localhost:8080/partidas?jugadorBlanco=Alice&jugadorNegro=Bob  
  
Respuesta:  
<UUID de la partida> (lo llamaremos {id} en los siguientes pasos)

## 7.2 Realizar Movimientos

1. Peón blanco de e2 a e4  
 PUT /partidas/{id}/mover?colOrigen=e&filaOrigen=2&colDestino=e&filaDestino=4

2. Peón negro de e7 a e5  
 PUT /partidas/{id}/mover?colOrigen=e&filaOrigen=7&colDestino=e&filaDestino=5

3. Caballo blanco de g1 a f3  
 PUT /partidas/{id}/mover?colOrigen=g&filaOrigen=1&colDestino=f&filaDestino=3

4. Caballo negro de b8 a c6  
 PUT /partidas/{id}/mover?colOrigen=b&filaOrigen=8&colDestino=c&filaDestino=6

5. Alfil blanco de f1 a c4  
 PUT /partidas/{id}/mover?colOrigen=f&filaOrigen=1&colDestino=c&filaDestino=4

6. Peón negro de d7 a d6  
 PUT /partidas/{id}/mover?colOrigen=d&filaOrigen=7&colDestino=d&filaDestino=6

7. Dama blanca de d1 a e2  
 PUT /partidas/{id}/mover?colOrigen=d&filaOrigen=1&colDestino=e&filaDestino=2

8. Peón negro de g7 a g6  
 PUT /partidas/{id}/mover?colOrigen=g&filaOrigen=7&colDestino=g&filaDestino=6

9. Enroque corto blanco (manual, Rey de e1 a g1)  
 PUT /partidas/{id}/mover?colOrigen=e&filaOrigen=1&colDestino=g&filaDestino=1

10. Alfil negro de c8 a g4  
 PUT /partidas/{id}/mover?colOrigen=c&filaOrigen=8&colDestino=g&filaDestino=4