**Documento de Análisis del Proyecto de Ajedrez**

**I. Finalidad del Programa:** Este programa tiene como finalidad simular un tablero de ajedrez en una interfaz gráfica. Los usuarios podrán realizar movimientos de piezas en el tablero siguiendo las reglas del ajedrez. El programa también registra y muestra y almacena el historial de movimientos realizados por los jugadores.

**II. Funcionalidades del Programa:** El programa permite a los usuarios:

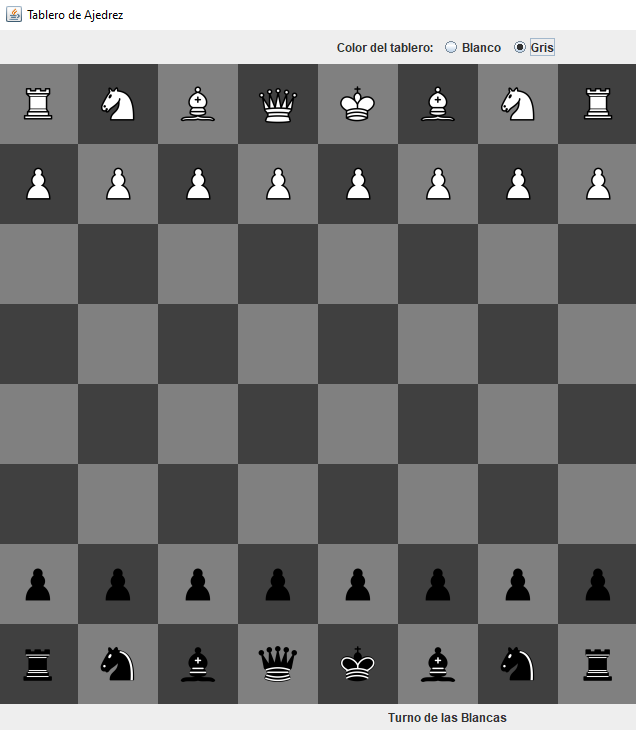
* Visualizar un tablero de ajedrez con todas las piezas posicionadas correctamente al inicio del juego.
* Seleccionar y mover las piezas siguiendo las reglas del ajedrez.
* Capturar las piezas del oponente cuando sea posible y mantener un historial de movimientos.
* Cambiar el turno automáticamente después de cada movimiento válido.
* Almacenar información de la partida.
* Mostrar mensajes de error en caso de movimientos inválidos.

**Algoritmos y Lógica del Programa:**

1. **Movimientos de Piezas:** El programa verifica la validez de los movimientos de las piezas en función de su tipo (peón, torre, caballo, alfil, reina y rey). Para cada tipo de pieza, se implementan algoritmos específicos que comprueban si el movimiento es legal según las reglas del ajedrez.
2. **Control de Turnos:** El programa maneja el cambio de turnos. Los turnos cambian automáticamente después de cada movimiento válido.
3. **Interfaz Gráfica:** La interfaz gráfica se construye utilizando la biblioteca Swing de Java. Se utiliza un diseño de tablero de ajedrez 8x8, donde cada casilla es un componente de JPanel. Las imágenes de las piezas se cargan desde archivos y se escalan para adaptarse al tamaño de las casillas.
4. **Gestión de Eventos:** Se utilizan listeners de eventos de mouse para detectar clics en las casillas del tablero y procesar los movimientos de las piezas.

**III. Modelo de Datos y Base de Datos:** El programa utiliza una base de datos donde se genera el almacenamiento de datos. Además, utiliza una estructura de datos en memoria para representar el estado del tablero y las piezas. Se utiliza un **HashMap** para asociar cada casilla del tablero con el nombre de la pieza que ocupa esa casilla.

**IV. Diagrama de Flujo Funcional:** El flujo funcional del programa se puede describir de la siguiente manera:

1. **Inicialización:** El programa crea y muestra la interfaz gráfica del tablero de ajedrez, posicionando las piezas en sus lugares iniciales. Se puede abrir reproductor de video con tutorial de curso de ajedrez y bloc de notas para tomar apuntes de la partida. También da la posibilidad de cambiar color de la interfaz y acceder desde un link que se cliquea desde la imagen de la interfaz principal a un curso de ajedrez desde una pagina web
2. **Espera de Movimientos:** El programa espera a que el usuario haga clic en una pieza y luego en una casilla de destino.
3. **Validación del Movimiento:** Cuando se hace un clic en una casilla, el programa valida si el movimiento es legal para la pieza seleccionada.
4. **Realizar el Movimiento:** Si el movimiento es válido, el programa mueve la pieza a la nueva casilla y actualiza el estado del tablero y el historial de movimientos.
5. **Cambio de Turno:** Después de cada movimiento válido, el programa cambia automáticamente al turno del otro jugador.
6. **Captura de Piezas:** Si el movimiento implica la captura de una pieza del oponente, se elimina esa pieza del tablero y se registra en el historial.
7. **Mostrar Mensajes de Error:** Si se intenta realizar un movimiento inválido, se muestra un mensaje de error y se espera otro movimiento.
8. **Almacena la información**: El programa a través de las clases BaseDeDatos.java y Registro.java almacena los datos de los movimientos de la partida en una base de datos de Access y un fichero txt.

Este diagrama describe el flujo funcional del programa, centrándose en cómo los usuarios interactúan con el tablero y cómo se validan y registran los movimientos en el historial del juego.

