**Project Report Template**

**Project Title:** 2D Chess with Adaptive AI Opponent

**Submitted By:** Shanza Noor

**Course:** AI

**Instructor:** Alishba Subhani

**Submission Date:** 11-05-2025

**1. Executive Summary**

● **Project Overview:**

This project aims to implement a traditional 2D chess game with an integrated AI opponent. The goal is to simulate intelligent behavior in a competitive environment using the Minimax algorithm enhanced with Alpha-Beta Pruning. The project focuses on building a strong AI capable of analyzing board positions and selecting optimal moves using custom heuristic evaluations.

**2. Introduction**

● **Background:**

Chess is a classic two-player board game played on an 8x8 grid, known for its deep strategy and tactical complexity. It is widely used in AI development due to its deterministic nature and well-defined rules. This project selects chess to explore AI decision-making and game-playing strategies.

● **Objectives of the Project:**

* To develop a fully functional 2D chess game.
* To implement a competitive AI using Minimax with Alpha-Beta pruning.
* To evaluate the AI’s effectiveness against human players.
* To build a clean and interactive user interface for gameplay.

**3. Game Description**

● **Original Game Rules:**

Chess is played on an 8x8 board with each player starting with 16 pieces: 8 pawns, 2 rooks, 2 knights, 2 bishops, 1 queen, and 1 king. Each piece has specific movement rules. The game’s objective is to checkmate the opponent's king.

* **Innovations and Modifications:**

This project does not modify traditional chess rules but focuses on integrating an intelligent AI opponent using established game theory algorithms.

**4. AI Approach and Methodology**

● **AI Techniques Used:**

The AI uses the Minimax algorithm, enhanced with Alpha-Beta pruning to reduce computation time by eliminating suboptimal branches.

● **Algorithm and Heuristic Design:**

Heuristics include:

* **Material Value:** Assigns scores to pieces (e.g., queen = 9, rook = 5).
* **Positional Advantage:** Rewards central control and piece activity.
* **King Safety:** Penalizes exposed kings or weak defensive setups.

● **AI Performance Evaluation:**

The AI’s performance was evaluated based on decision speed, move accuracy, and ability to compete with human players.

**5. Game Mechanics and Rules**

● **Modified Game Rules:**

No modifications were made to the standard rules of chess.

● **Turn-based Mechanics:**

Players alternate turns. White moves first. The game ends in checkmate, draw, or resignation.

● **Winning Conditions:**

A player wins by checkmating the opponent's king. Draw conditions include stalemate, threefold repetition, and insufficient mating material.

**6. Implementation and Development**

● **Development Process:**

The game was developed using modern frontend and backend technologies. The AI logic was designed using NumPy, and the user interface was built using React.js.

● **Programming Languages and Tools:**

○ Programming Language: JavaScript (Frontend), Python (Backend & AI)

○ Libraries: NumPy (AI), React.js (UI)

○ Tools: GitHub (Version Control), Flask (Backend Server)

● **Challenges Encountered:**

* Tuning heuristics to make the AI competitive.
* Maintaining performance during deeper search depths.
* Integrating AI decisions smoothly with the UI flow.

**7. Team Contributions**

● **Team Members and Responsibilities:**

This project was completed individually.

* **Shanza Noor:** Responsible for game logic, AI implementation, UI development, and testing.

**8. Results and Discussion**

● **AI Performance:**

The AI performed competitively, demonstrating logical moves and maintaining consistent decision-making. It was able to make most decisions within 2–3 seconds and successfully challenged human players during testing. The Alpha-Beta pruning significantly improved efficiency.

**9. References**

* Russell, S., & Norvig, P. (2020). Artificial Intelligence: A Modern Approach.
* Chess Programming Wiki – Heuristics and Evaluation.
* React.js Documentation
* NumPy Documentation
* Chess.com – Game Rules and Strategies