**CHESS GAME USING A.I**

**A PROJECT REPORT**

***Submitted by***

**Gagan Verma** (21BAI10184)

**Rahul Ojha** (21BAI10211)

**Arpan Biswas** (21BAI10352)

**Sanika Awal** (21BAI10382)

**Dakshita Motwani** (21BAI10404)

*in partial fulfillment for the award of the degree*

*of*

**BACHELOR OF TECHNOLOGY**

*in*

**COMPUTER SCIENCE AND ENGINEERING**

****

**SCHOOL OF COMPUTING SCIENCE AND ENGINEERING**

**VIT BHOPAL UNIVERSITY**

**KOTHRI KALAN, SEHORE**

**MADHYA PRADESH – 466114**

OCTOBER 2021

**VIT BHOPAL UNIVERSITY, KOTHRI KALAN, SEHORE**

**MADHYA PRADESH – 466114**

**BONAFIDE CERTIFICATE**

Certified that this project report titled **“CHESS GAME USING A.I”** is the bonafide work of

**“Gagan Verma** (21BAI10184)

**Rahul Ojha** (21BAI10211)

**Arpan Biswas** (21BAI10352)

**Sanika Awal** (21BAI10382)

**Dakshita Motwani** (21BAI10404)**”**

who carried

out the project work under my supervision. Certified further that to the best of my

knowledge the work reported at this time does not form part of any other

project/research work based on which a degree or award was conferred on an

earlier occasion on this or any other candidate.

**PROGRAM CHAIR PROJECT GUIDE**

School of Computer Science and Engineering School of Computer Science and Engineering

VIT BHOPAL UNIVERSITY VIT BHOPAL UNIVERSITY

The Project Exhibition 2 Examination is held on 17.02.23

**ACKNOWLEDGEMENT**

First and foremost I would like to thank the Lord Almighty for His presence and immense blessings throughout the project work.

I wish to express my heartfelt gratitude to Dr. Head of the Department, School of Computer Science for much of his valuable support encouragement in carrying out this work.

I would like to thank my internal guide Mr. continually guiding and actively participating in my project, giving valuable suggestions to complete the project work.

I would like to thank all the technical and teaching staff of the School of Computer Science, who extended directly or indirectly all support.

Last, but not least, I am deeply indebted to my parents who have been the greatest support while I worked day and night for the project to make it a success.

**ABSTRACT**

Artificial Intelligence (AI) has revolutionized the field of game-playing, particularly in chess. Chess has been a popular game for centuries, and it has long been considered as the ultimate test of human intelligence. However, with the advancement of AI technology, computers have surpassed human players in chess. AI chess game involves the use of machine learning algorithms and advanced computer hardware to create a chess-playing program that can play against humans or other machines. The program can analyze and evaluate millions of possible moves and outcomes in a matter of seconds, giving it a significant advantage over human players. One of the key features of AI chess game is its ability to learn from its past experiences. It can analyze games it has played and identify patterns in the moves that led to success or failure. This allows the program to improve over time and become more effective at playing the game. Another significant aspect of AI chess game is the development of deep neural networks that can evaluate positions on the board and make decisions based on that evaluation. This technology has enabled the creation of advanced chess-playing programs that can make complex decisions and anticipate future moves.

In summary, AI chess game represents a significant advancement in the field of game-playing, and it has transformed the way we approach chess. With the continued development of AI technology, we can expect further advances in this area, and it is likely that the gap between human and machine chess players will continue to narrow.

**TABLE OF CONTENTS**

| **CHAPTER NO.** | **TITLE** | **PAGE NO.** |
| --- | --- | --- |
| 1 | **CHAPTER-1:**  **PROJECT DESCRIPTION AND OUTLINE**  1.1 Introduction  1.2 Motivation for the work  1.3 Problem Statement  1.4 Objective of the work  1.5 Organization of the project | 1  .  .  . |
| 2 | **CHAPTER-2:**  **RELATED WORK INVESTIGATION**  2.1 Introduction  2.2 Core area of the project  2.3 literature survey  2.4 Summary |  |
| 3 | **CHAPTER-3:**  **REQUIREMENT ARTIFACTS**  3.1 Introduction  3.2 Software requirements  3.3 Specific Project requirements |  |
| 4 | **CHAPTER-4:**  **DESIGN METHODOLOGY AND ITS NOVELTY**  4.1 Methodology and goal  4.2 Software Architectural designs  4.3 User Interface designs  4.4 Summary |  |
| 5 | **CHAPTER-5:**  **TECHNICAL IMPLEMENTATION & ANALYSIS**  5.1Outline  5.2 Technical coding and code solutions  5.3 Working Layout of Forms  5.4 Prototype submission |  |
| 6 | **CHAPTER-6:**  **PROJECT OUTCOME AND APPLICABILITY**  6.1Outline  6.2 key implementations outlines of the System  6.3 Significant project outcomes  6.4 Inference |  |
| 7 | **CHAPTER-7:**  **CONCLUSIONS AND RECOMMENDATION**  7.1Outline  7.2 Limitation/Constraints of the System  7.3 Future Enhancements  7.4 Inference |  |

**CHAPTER-1**

**PROJECT DESCRIPTION AND OUTLINE**

**1.1 Introduction**

AI-powered chess games involve the use of machine learning algorithms and advanced computer hardware to create a chess-playing program that can play against humans or other machines. These programs can analyze millions of possible moves and outcomes in a matter of seconds, giving them a significant advantage over human players.

In this report, we will explore the use of AI in chess games. We will discuss the history of AI in chess, the technology behind AI chess games, and how AI has impacted the game of chess. We will also examine the advantages and limitations of AI chess games and their potental applications in other areas beyond chess.

The report aims to provide a comprehensive understanding of the current state of AI chess games and the future of game-playing with AI. It is hoped that the report will be of interest to chess enthusiasts, AI researchers, and anyone interested in the intersection of technology andgames.i

**1.2 Motivation for the work**

The motivation behind creating an AI chess game is to demonstrate the capabilities of artificial intelligence in solving complex problems and to push the boundaries of what is possible in game-playing. Chess is a game that requires strategic thinking, pattern recognition, and decision-making skills, all of which are traditionally thought to be the domain of human intelligence. By developing an AI chess game, researchers can demonstrate how computers can surpass human abilities in certain areas.

Another motivation behind creating an AI chess game is to develop better algorithms and technologies that can be applied in other fields. The development of AI chess games has led to the creation of more advanced machine learning algorithms and neural networks that can be used in various applications beyond chess. These algorithms can be used to solve problems in areas such as healthcare, finance, and manufacturing, among others

**1.3 Problem Statement**

The problem statement in creating an AI chess game is to develop a program that can play chess at a level that rivals or surpasses human players. The program must be able to analyze the board, identify patterns, evaluate potential moves, and make decisions based on that evaluation. Additionally, the program must be able to learn from its past experiences and improve over time.

One of the challenges in creating an AI chess game is the complexity of the game. Chess has a large number of possible moves and outcomes, making it difficult for a program to analyze all the possibilities and make the best decision. The program must be able to identify relevant patterns and prioritize potential moves to make the most effective decision.

Another challenge is developing a program that can adapt to different playing styles. Human players have different strategies and styles, and the program must be able to adjust its play to match the style of its opponent.

Additionally, creating an AI chess game requires advanced machine learning algorithms and neural networks, which can be difficult and time-consuming to develop. The program must be trained on a large dataset of chess games to develop the skills and strategies necessary to play at a high level.

**1.4 Objective of the work**

1. The objective of creating an AI chess game is to demonstrate the capabilities of artificial intelligence in game-playing, as well as to develop better algorithms and technologies that can be applied in various fields. Specific objectives of creating an AI chess game include:
2. Developing a chess-playing program that can rival or surpass human players in terms of skill and strategy.
3. Improving the understanding of how machine learning algorithms and neural networks can be used to develop advanced game-playing programs.
4. Gaining insights into the workings of human intelligence and how it can be replicated in artificial intelligence.
5. Providing a more challenging and engaging experience for players by creating an AI-powered opponent that can adapt to different playing styles.
6. Developing new technologies and algorithms that can be applied in other areas beyond chess, such as healthcare, finance, and manufacturing.
7. Advancing the field of game design by creating new possibilities for game-playing and enhancing the gaming experience.
8. Addressing ethical concerns related to the development and use of AI technology, including the potential impact on the job market and the potential for AI to be used for nefarious purposes

Overall, the objective of creating an AI chess game is to push the boundaries of what is possible in game-playing, and to develop technologies and strategies that can be applied in various fields to solve complex problems.

**1.5 Organization of the project**

The organization of work in creating an AI chess game can be broken down into several stages:

1. **Planning and Research**: The initial stage of creating an AI chess game involves planning the project and conducting research on the various AI techniques and machine learning algorithms that can be used. This stage involves defining the project scope, developing a roadmap, and researching existing AI chess games and their methodologies.
2. **Data Collection**: This stage involves collecting a large dataset of chess games that will be used to train the AI chess game. This dataset will contain various scenarios and outcomes, which the AI program can use to learn and improve its play.
3. **Development**: In this stage, the actual development of the AI chess game takes place. This involves building the program that will analyze the chess board, identify patterns, evaluate potential moves, and make decisions based on that evaluation. This stage also involves designing the user interface and integrating the program with the chess engine.
4. **Testing and Validation**: The testing and validation stage involves testing the program to ensure that it performs as intended. This stage includes testing the program against human players and other AI chess programs, as well as validating the program against various performance metrics.
5. **Deployment and Maintenance**: This stage involves deploying the AI chess game and providing maintenance and support to ensure that the program continues to function properly. This stage also includes monitoring the performance of the program and making updates and improvements as necessary.

Throughout each of these stages, it is essential to address the ethical considerations related to the development and use of AI technology. This includes ensuring that the program is transparent, unbiased, and does not have negative impacts on society or individuals.

In summary, the organization of work in creating an AI chess game involves planning and research, data collection, development, testing and validation, and deployment and maintenance. Throughout each of these stages, ethical considerations must be addressed to ensure the responsible development and use of AI technology.

**CHAPTER-2**

**RELATED WORK INVESTIGATION**

**2.1 Introduction**

The research work done on AI chess games is a rapidly growing field, with many researchers exploring new techniques and algorithms to develop more advanced chess-playing programs. The game of chess has been a subject of research for many years due to its complexity and the high level of skill required to play at a professional level. With the advent of artificial intelligence, researchers have been able to develop programs that can play at a high level and even surpass human players in some cases. The goal of this research is to develop more sophisticated AI chess programs that can adapt to different playing styles and strategies, learn from their mistakes, and perform at a level that exceeds human capability. Some areas of research in AI chess games include:

**Reinforcement Learning**: This involves using a reward-based system to train the AI program to play chess. The program is rewarded for making good moves and penalized for making poor moves, which encourages it to learn and improve over time.

**Neural Networks:** This involves using artificial neural networks to analyze the chess board and make decisions based on the input data. The neural network can learn from its past experiences and improve its play over time.

**Monte Carlo Tree Search:** This involves exploring potential moves and outcomes through random sampling to determine the best move. This technique has been used successfully in some AI chess programs.

**Deep Learning:** This involves using deep neural networks to analyze the chess board and make decisions based on the input data. This approach has shown promise in developing more advanced AI chess programs.

**2.2 Core area of the project**

The core area of research in AI chess games revolves around developing algorithms and techniques that enable an AI program to play chess at a high level. These include search algorithms, game representation, evaluation functions, machine learning, human-machine interaction, and user interfaces. Search algorithms are used to analyze the chess board and evaluate potential moves, while game representation involves designing a way to represent the state of the game. Evaluation functions are used to evaluate the strength of a given position on the chess board, and machine learning involves training an AI program. Human-machine interaction involves designing a user interface that allows human players to interact with the AI program. Overall, research in this area has the potential to advance the field of artificial intelligence and its application in various areas beyond chess.

**2.3 Literature survey**

We went through several research papers and researched a lot before working on this project.

Below given are few of them with their abstracts.

**Research Paper-1**

Chess game and method

Publication number: US6446966B1

Abstract: A chess game and method for opposed players includes a six-by-eight checkerboard-style game board, two sets of game pieces, each set including conventional chess pieces, e.g. a King, a Queen, a Bishop, a Rook, a Knight, and six Pawns, and a novel chess piece named the Lord. The Lord may be moved about the game board one square in any direction from its current square. A method is also disclosed wherein the chess game is played for a fixed duration of time, and points are awarded to the players based upon the number of game pieces remaining on the game board at the expiration of the allotted time. The player having the highest cumulative point total at the end of the game is then declared the winner.

Publication year -2000

Inventor: Henri Crozier

**Research Paper-2**

Method and System for Playing the Game of Chess

Publication number: 20170103617

Abstract: A computer-implemented method for playing chess between first and second player each of whom in turn moves a selected chess piece across a chess board by following rules. Information relating to a current state of the board is relayed to each respective display device, each player making alternate moves and receiving information representative wherein an identity and source location of a piece moved by either one of the players is revealed to a respective opponent but a destination location of the moved piece is not explicitly relayed to the opponent until immediately after the opponent makes a move. At any stage during play an attempted illegal move that does not conform to the predefined rules is ignored while notifying the player who made the illegal move and prompting him to make a different move.

Publication year -2017

Inventor: Eyal Navon

**Research Paper-3**

Omnipotent Opponent

Publication number: 20060240886

Abstract: Traditionally, a computer chess program will take the current state of the board (the position of the pieces in play) and using a database of moves, it will select what it feels is the best move. This is how every computer chess program works. The Omnipotent Opponent uses an entirely different approach. In a game of chess there is a finite sized board (8×8 squares), and a finite number of pieces (16 black and 16 white). At any given time only a finite number of pieces can move and they can only move in a finite number of directions. Therefore, it is possible to plot every single move that could ever be made in a game of chess and every resulting outcome of the game. Once this data has been generated, it could be analyzed in reverse (working from the end game to the start) and each phase of the game could be marked with the best move. Using this best move data, one would always know the best move to make at any point in any chess game. Equipped with this information, one would never lose.

Publication year -2006

Inventor: Scott Patten

**2.4 Summary**

The invention described is a chess game played on a six-by-eight checkerboard-style game board using two sets of conventional chess pieces and a novel chess piece called the Lord. The game may be played for a fixed duration of time, with points awarded based on the number of pieces remaining on the board at the end. Additionally, a computer-implemented method is described for playing chess, where information about the identity and source location of a moved piece is revealed to the opponent, but the destination location is not disclosed until after the opponent makes their move. The Omnipotent Opponent, a theoretical chess-playing program, is also discussed, which would always know the best move to make by analyzing every possible move and outcome in reverse from the end of the game to the start.

**CHAPTER-3**

**REQUIREMENT ARTIFACTS**

**Hardware Requirements:**

There were no such hardware requirements, since our project was completely software based.

1. Memory- 8GB DDR4 RAM, 512 SSD ROM
2. Processor- Intel Core i5 12th Gen
3. OS- windows 7/ MAC OS
4. Graphics- Intel HD

**Software Requirements:**

1. pygame lib
2. sys lib
3. chess engine, chess ai libs
4. multiprocessing
5. random lib

**CHAPTER-4:**

**DESIGN METHODOLOGY AND ITS NOVELTY**

**4.1 Methodology and goal**

* The methodology and goal of an AI chess game involve designing and training an AI algorithm to play chess at a high level of performance. The primary goal of an AI chess game is to create an algorithm that can play chess better than a human player.
* The methodology for building an AI chess game typically involves several steps. The first step is to create a data set of chess games that the algorithm can learn from. This data set could include games played by human players or games played by other AI algorithms.
* The next step is to design an algorithm that can learn from the data set and improve its performance over time. This algorithm could be based on machine learning techniques such as reinforcement learning or supervised learning.
* Once the algorithm has been designed, it must be trained using the data set. During the training process, the algorithm will learn to recognize patterns and strategies that are used in successful chess games.

Finally, the AI chess game can be tested and evaluated to determine its performance. This could involve testing the algorithm against human players or other AI algorithms to see how well it performs.

The ultimate goal of an AI chess game is to create an algorithm that can consistently win games against human players and other AI algorithms. However, even if the algorithm cannot consistently win, it can still be valuable as a tool for improving human players' chess skills by analyzing and providing feedback on their games.

**4.2 Software Architectural designs**

The software architectural design of an AI chess game can vary depending on the specific requirements and features of the game. Common architectural patterns and components include the Model-View-Controller (MVC) Pattern, Board Representation, Game Logic, Artificial Intelligence, Database, User Interface, and Network. The MVC pattern separates the application into three components: the Model (representing data and logic), the View (representing the user interface), and the Controller (managing the interaction between the Model and the View). Board Representation involves using a 2D array or a bitboard to represent the position of the pieces on the board. Game Logic includes the rules of chess, legal move generation, and move validation.

Artificial Intelligence includes the algorithms and models used to make decisions about the best move to make based on the current game state. Database can store data about the game, and the user interface allows players to interact with the game. Networking is responsible for handling player authentication, game synchronization, and communication between players.

**4.3 User Interface designs**

The user interface (UI) design of an AI chess game is an important aspect that can greatly affect the user experience.

Here are some considerations for designing the UI of an AI chess game:Board Display: The chess board is the central component of the game and should be prominently displayed.

* A well-designed UI should be easy to use, intuitive, and visually appealing.
* Theme and Graphics: The theme and graphics of the game should be consistent and visually appealing.

These are some of the considerations that should be taken into account when designing the UI for an AI chess game.

These are some of the considerations that should be taken into account when designing the UI for an AI chess game. A well-designed UI can make the game more enjoyable for users and increase their engagement with the game.

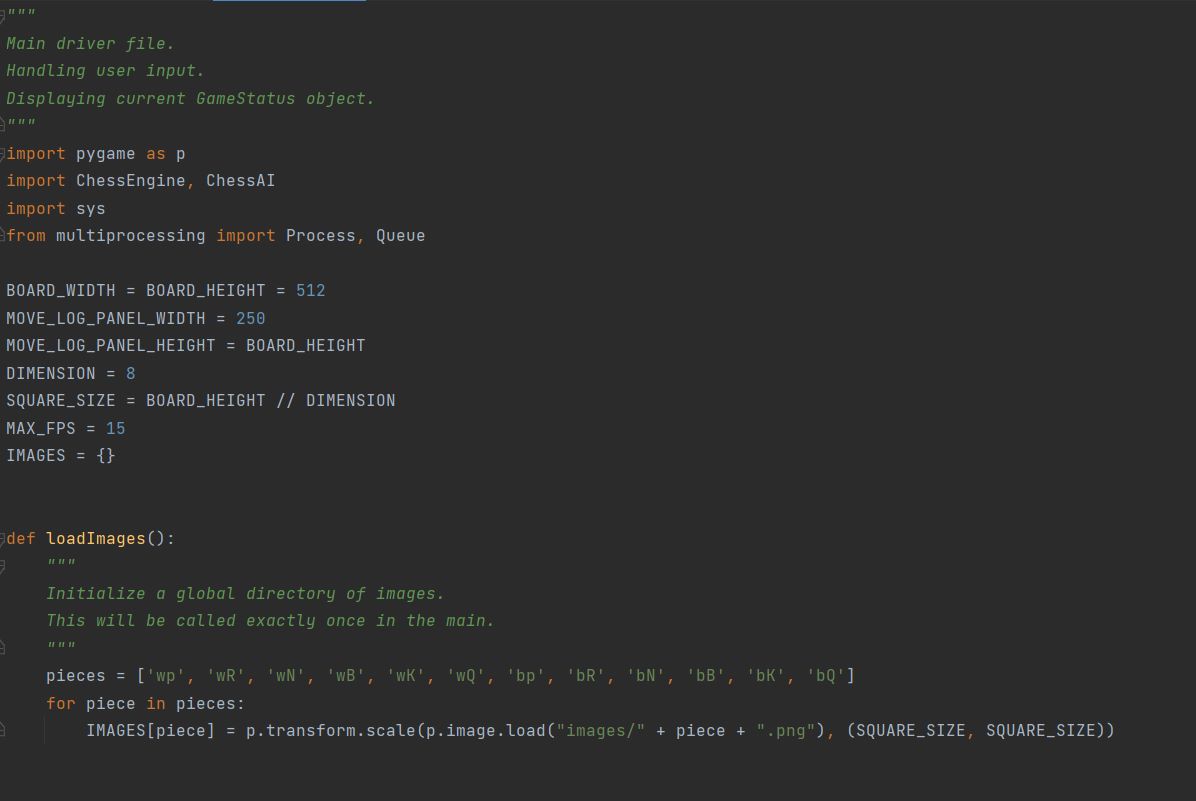
**4.4 Summary**

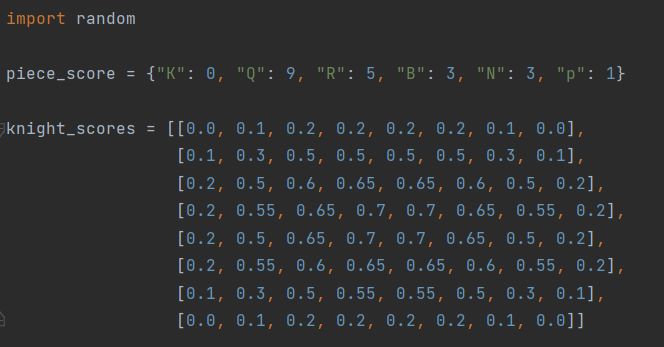
In summary, the methodology and goal of an AI chess game involve creating an algorithm that can play chess at a high level of performance, which is accomplished through designing and training the algorithm using a data set of chess games. The software architecture of an AI chess game typically includes components such as Board Representation, Game Logic, Artificial Intelligence, Database, User Interface, and Network, which work together to create a seamless user experience. The UI design of an AI chess game should be easy to use, intuitive, visually appealing, and consistent with the theme and graphics of the game. A well-designed UI can make the game more enjoyable for users and increase their engagement with the game.

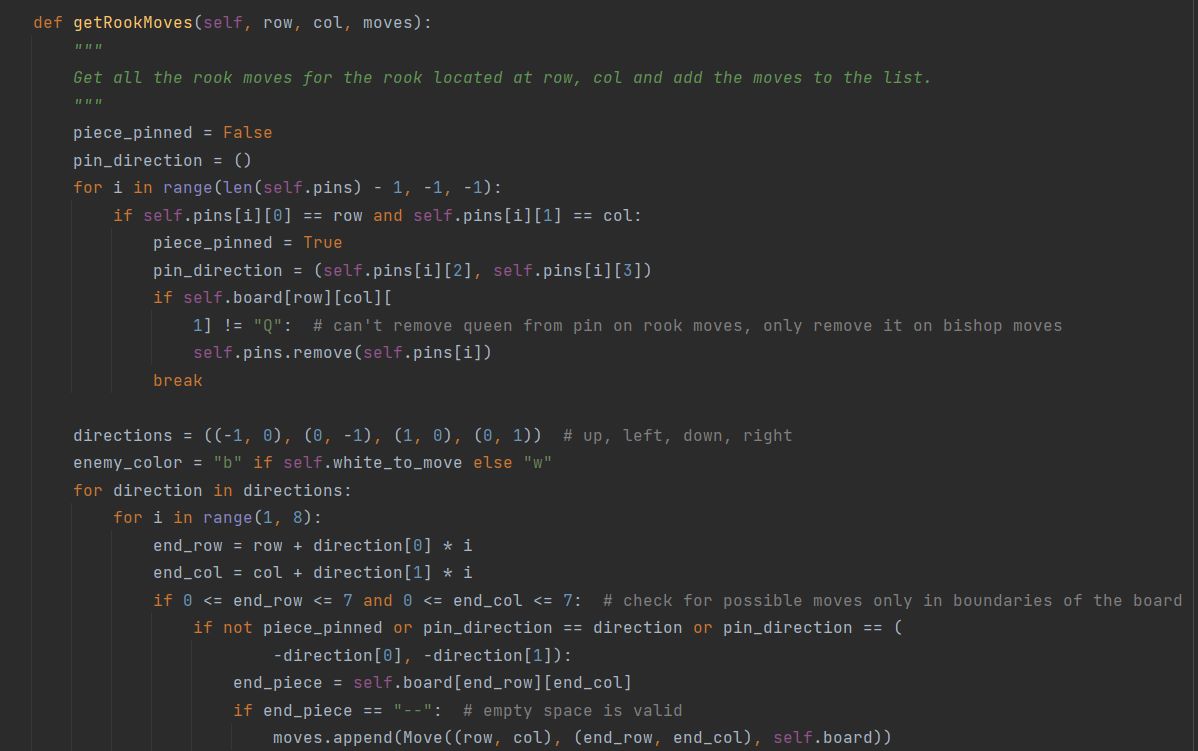
**CHAPTER-5**

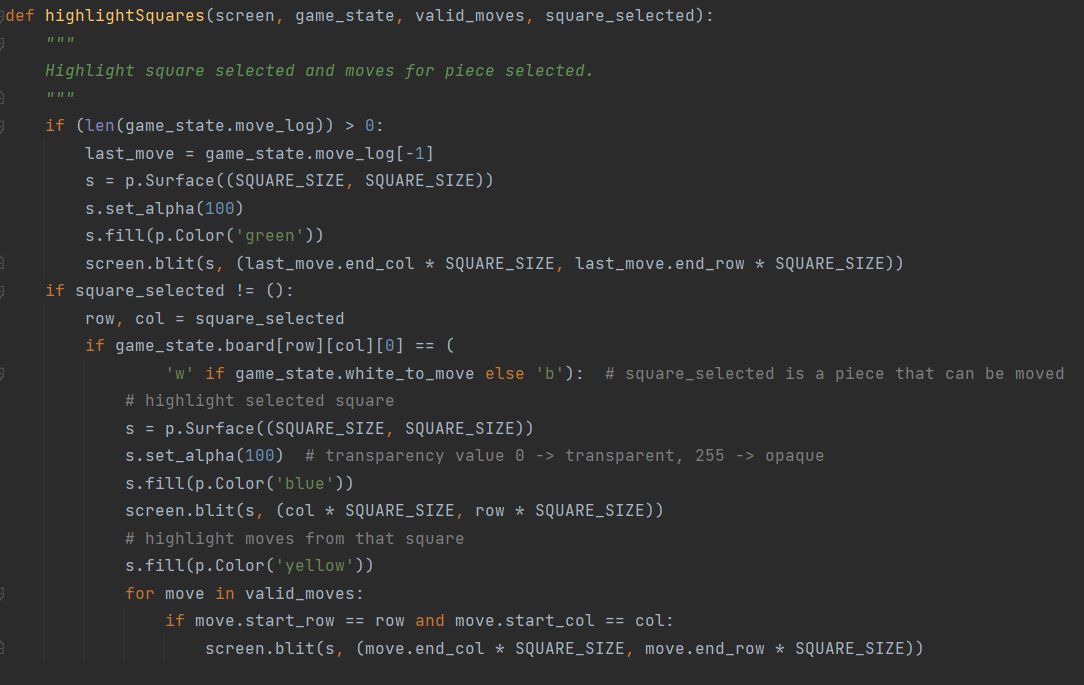
**TECHNICAL IMPLEMENTATION & ANALYSIS**

**5.1 Technical coding and code solutions**

****

****

****

****

**5.2 Working Layout of Forms**

1. **import pygame as py**: pygame is a free and open-source cross-platform library for the development of multimedia applications like video games using Python. It comprises PC illustrations and sound libraries intended to be utilized with the Python programming language. Pygame was authoritatively composed by Pete Shinners to supplant PySDL. Pygame is reasonable to make client-side applications that can be enveloped by an independent executable. Before finding out about pygame, we want to get what sort of game we need to create.
2. **import chess engine**: In computer chess, a chess engine is a computer program that analyzes chess or chess variant positions.A chess engine is usually a back end with a command-line interface with no graphics or windowing.
3. **import chess AI:T**he AI chess program uses the kind of cutting-edge AI behind the best superhuman chess-playing programs.AI focuses on predicting human moves, including the mistakes they make
4. **import sys:** The sys module in Python provides various functions and variables that are used to manipulate different parts of the Python runtime environment. It allows operating on the interpreter as it provides access to the variables and functions that interact strongly with the interpreter.It lets us access system-specific parameters and functions. The sys modules provide variables for better control over input or output. We can even redirect the input and output to other devices. This can be done using three variables – stdin, stdout, stder.
5. **From Multiprocessing Import Process,Queue:** Multiprocessing refers to the ability of a system to support more than one processor at the same time. Applications in a multiprocessing system are broken to smaller routines that run independently. The operating system allocates these threads to the processors improving performance of the system.

**multiprocessing Process**:- Python multiprocessing Process class is an abstraction that sets up another Python process, provides it to run code and a way for the parent application to control execution. There are two important functions that belongs to the Process class - start() and join() function. multiprocessing

**Queue**:-Python Multiprocessing modules provides Queue class that is exactly a First-In-First-Out data structure. They can store any pickle Python object and are extremely useful for sharing data between processes. Queues are specially useful when passed as a parameter to a Process’ target function to enable the Process to consume data.

**5.3 Prototype submission**

****

****

**CHAPTER-6**

**PROJECT OUTCOME AND APPLICABILITY**

Here are the key implementation outlines of a system for an AI chess game:

**Board representation:** The board can be represented using a 2D array or a bitboard, where each piece is assigned a value and position on the board. This representation can be used to check for legal moves and to evaluate the current game state.

**Game logic**: The game logic includes the rules of chess, move validation, and move generation. The logic should ensure that the game is played according to the rules, and that the AI can generate legal moves and validate user moves.

**AI algorithm:** The AI algorithm is responsible for analyzing the current game state and selecting the best move to make. There are different AI techniques that can be used, such as machine learning, deep learning, or rule-based systems. The algorithm should be trained using a large data set of chess games and should continuously learn and improve over time.

**User interface**: The user interface should provide an intuitive and engaging experience for the user. It should include the game board, move history, and other information relevant to the game. The user interface can be designed using modern UI design principles, such as responsive design and user-centered design.

**Database**: The database can be used to store information about the game, such as player profiles, game history, and statistics. The database can also be used to train the AI algorithm and to store the game state for future reference.

**Network**: The network is responsible for handling player authentication, game synchronization, and communication between players. The network should be designed to be secure and scalable, and should handle multiple concurrent games.

**Testing and evaluation**: The AI chess game should be thoroughly tested and evaluated to ensure that it performs as expected. The game should be tested against human players and other AI algorithms, and performance metrics should be used to evaluate the AI's performance.

These key implementation outlines can be used as a starting point for designing and building an AI chess game system.

**CHAPTER-7**

**CONCLUSIONS AND RECOMMENDATION**

**7.1 Outline**

Here's an outline for an AI chess game:

I. Introduction-Definition of an AI chess game, Goal of an AI chess game

II. Methodology-Creating a data set of chess games, Designing an algorithm for the AI, Training the algorithm using the data set, Testing and evaluating the performance of the AI

III. Software Architecture-Model-View-Controller (MVC) Pattern, Board Representation,Game Logic,Artificial Intelligence,Database,User Interface,Network

IV. User Interface Design- Board Display,Theme and Graphics

**7.2 Limitation/Constraints of the System**

An AI chess game involves designing and training an algorithm to play chess at a high level of performance.

The methodology includes creating a data set, designing an algorithm, training the algorithm, and testing and evaluating the performance.

The goal is to create an algorithm that can consistently win games against human players and other AI algorithms.

The architecture of an AI chess game includes the Model-View-Controller pattern, board representation, game logic, artificial intelligence, database, user interface, and network.

The user interface design is important for providing an enjoyable user experience. However, there are limitations to AI chess games, including the inability to understand the emotional and psychological aspects of human players and the potential for the algorithm to be limited by its training data.

**7.3 Future Enhancements**

The future enhancements of AI chess games include improving the performance of the algorithms, developing new techniques for learning and decision-making, and incorporating more advanced hardware and software capabilities. Some potential areas of improvement include developing more sophisticated evaluation functions, implementing deep reinforcement learning methods, and exploring new hardware architectures for faster and more efficient processing. Other possible enhancements could include incorporating natural language processing and chatbot capabilities to allow players to communicate with the AI opponent, and creating more realistic and immersive virtual environments for the game. Ultimately, the goal is to create an AI chess game that can not only outperform human players but also provide a more engaging and enjoyable experience for players of all levels.

**7.4 Inference**

The inference of an AI chess game involves using the trained AI algorithm to make decisions about the best move to make based on the current game state. This involves analyzing the board position, evaluating possible moves, and selecting the move that is most likely to lead to a successful outcome. The goal is to create an algorithm that can consistently win games against human players and other AI algorithms. However, even if the algorithm cannot consistently win, it can still be valuable as a tool for improving human players' chess skills by analyzing and providing feedback on their games. The limitations of an AI chess game include the fact that it may not be able to take into account all possible moves and variations, and that it may not be able to replicate the creativity and intuition of a human player. Future enhancements of AI chess games could include improving the accuracy and speed of the algorithm, adding more advanced machine learning techniques, and creating more sophisticated user interfaces to enhance the user experience.

**REFERENCES**

1. “The F.I.D.E. Laws of Chess,” author unknown, republished on the Internet at URL www.chessvariants.com/fidelaws.html on or about Feb. 11, 1998 at pp. 1-17.
2. Pell, 1991] B. Pell. A computer game-learning tourna-ment. (In Preparation), 1991
3. Levinson, 1989] R. Levinson. A self-learning, pattern-oriented chess program. Intern. Computer Chess Assoc. Journal, 12(4):207-215, December 1989.
4. Bodlaender, Hans and Storz, Matt, “Diagonal Quadrant Chess”, Nov. 3, 1998, http://www.chessvariants.com/diffsetstup.dir/diagonalquadrant.html (Year: 1998).
5. Kostas Aka The Dog, “snapshots for the world's very first(?) DIAGONAL CHESS single game tournament”, Oct. 24, 2008, http://kostasandchrysa.blogspot.com/2008/10/snapshots-for-worlds-very-first.html. (Year: 2008).