AI Chess Positional Analysis Engine v.1.1

First Phase of the ChessCo (Chess Companion) Web Application
V 1.1 Implemented Dec 2022 - Jan 2023

Personal software development project implemented by Boris Tomov

under the supervision of Radoslava Slavova - Chief Software Development Architect

1. ChessCo Application Overview

ChessCo is an online web platform dedicated to providing comprehensive chess education and training for users from all ages and skill levels. It serves as a companion, which slowly but surely guides users through the complexities of the game and provides exposure to a variety of chess concepts and winning strategies. While using the application, people have the choice to either browse through the learning panels and understand more on how to play tactical positional chess or engage in various gameplay modes and quizzes aimed to develop their intuition about the game.

2. Ideas and Motivation

The idea for developing this project has been heavily influenced by my personal passion for developing socially impactful computer software. In this regard, ChessCo is designed to be easily accessible for low-income communities with underdeveloped educational systems that do not have access to any extra-curricular learning activities or social clubs.

With chess being widely renowned for its positive effects on creativity, memory function and brain development, the ChessCo learning companion would hopefully allow young people to improve their problem solving skills and discover new interests. Furthermore, thanks to the ever growing popularity of the game, users would be able to connect with many more chess enthusiasts and feel welcomed in the online chess community.

3.Project Timeline

Phase & Version	Deadline
Phase I - v1.1 (Pseudo-Legal Filtering Mechanisms, Board Render & Basic Functionality, Board Analysis)	January 20, 2023
Phase I - v1.2 (Evaluation, Engine Recommendations)	March 31, 2023
Phase II - v2.1 (Learning Panel & Theory Courses)	May 1, 2023
Phase II - v2.2 (Gameplay & Exercises) Project Deployment	July 1, 2023

4.Application Content and Features

Theory Courses

In line with ChessCo's learning focused domain, the app is going to contain a Learning Panel section containing well-structured and detailed documents/online presentations guiding users step by step in the process of learning a new chess opening, strategy or a concept that could be useful in their development as a chess player.

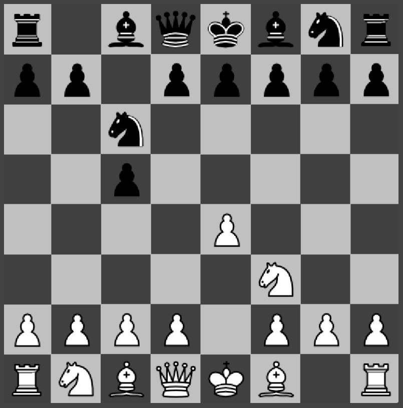
The Learning Panel will furthermore be equipped with a search browser that could retrieve information about a specific topic the user wants to review and would suggest subsequent learning actions one can proceed with from that point on.

Different Modes of Gameplay

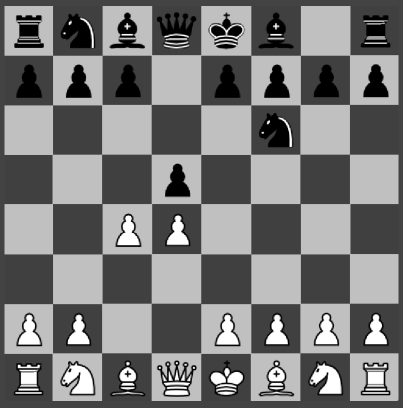
Opening Challenge

One of ChessCo's most exciting educational features is the online Opening Challenge game mode, which allows users to extend their experience playing any of chess' most popular game openings. To ensure stable matchmaking timelapse, ChessCo's Opening Challenge will only feature about 5 to 8 separate opening game modes at a time, which will be updated on a weekly basis, so that users could experience a diverse variety of chess positions.

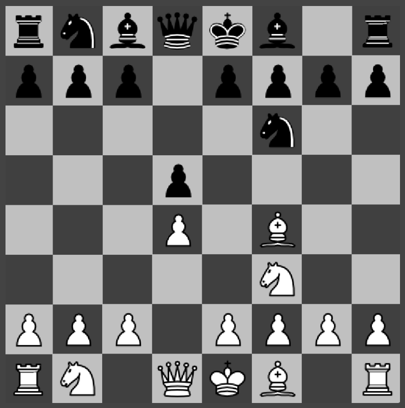
Here are some examples of popular chess openings:



Sicilian Defense



Queen's Gambit



London System

Custom Challenge

In case, however, a user decides they need more practice playing a selected opening position, the Custom Challenge game mode will allow them to personalize the board however they like and host private online tournaments that allow groups of friends to play against each other at a desired board setting. This could be done using the FEN Parsing Feature explained later in the project spec.

Classical Challenge

ChessCo's Classical Challenge is a gameplay mode which allows users to play online 1v1 games against each other in a classical chess board configuration. This is the part of the application where users can really showcase the knowledge obtained through reviewing the Theory Courses and playing the Opening Challenge game mode.

Furthermore, the Classical Challenge game mode will rely on an elo-based matchmaking algorithm which ensures that users will always be matched against opponents from a similar skill level. This would also mean that players could always find appropriate challenges which insure that they can maintain a steady positive progress in their chess journey.

Interactive Chess Engine

Another unique element of ChessCo is its interactive chess move-calculating engine capable of executing in-game positional analyses, which evaluate a given player's probabilistic winning coefficient and compile a selection of current best possible moves.

Furthermore, this feature also provides information about potential opponent responses, as well as constructive feedback to the user explaining why a certain move results in a better or worse evaluation of their winning probability.

Upon request from the user, the engine will also be able to compile a list of focus areas, containing links to suggested exercises and theory courses, which the player can visit in order to learn and achieve better results.

For more information on the technical side - the engine's complex construction relies on the implementation of the following underlying algorithms and functionalities:

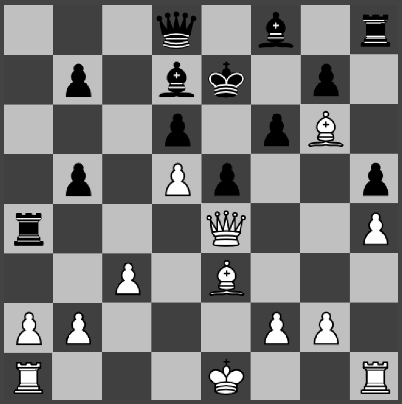
- Pseudo-legal filtering mechanism
  - Ensures that the suggested moves comply with the official chess rules about legality and pseudo-legality. Utilizes defined concepts of pinned pieces, checks and checkmate in order to optimize the branch search time algorithms which determine the best move in a position.

- Database References
  - The chess engine is equipped with a **FEN string parsing feature**, which allows quick imports of particular board configurations encoded in FIDE chess archives or online databases like chesstempo.com and chessdatabase.com. These streams of data are also essential in the AI learning phase of the engine development because they allow it to utilize a wide range of grandmaster games and speed up the process of developing general chess intuition. More on that in version 1.2 of the document.

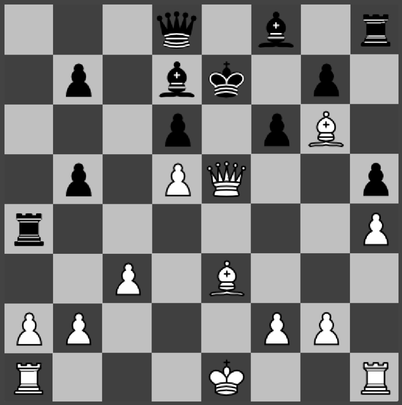
Exercises

As another learning tool that could benefit users' learning experience in the application, ChessCo will contain a separate section for chess puzzles. For a general idea, puzzles in chess represent abstract board positions where many moves seem to be good for a player to make, but in reality only one is truly winning. In that regard, the so-called Puzzle Challenge panel will focus on exercising users' creativity by providing access to a big stock of chess puzzles, which users can try to beat.

As an example of a chess puzzle note the image below:

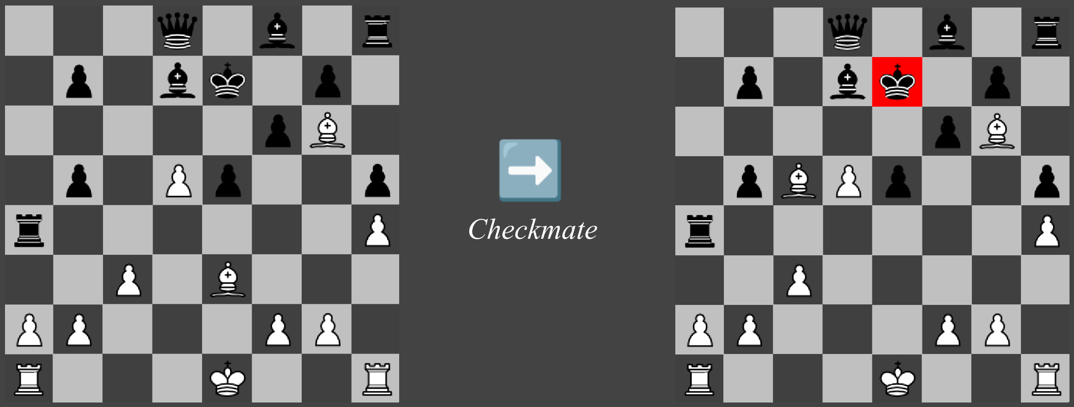


This is a very complex chess position with many pieces scattered across the board. Assuming that there is a person trying to find the best move with the white pieces, they would want to spend a significant period of time trying to analyze the board and figure out what move to play. An obvious first remark would be noticing that the white queen is currently under an attack from the enemy rook on the 4th rank. Logically, a regular chess player would move their queen out of danger. This move, however, while not necessarily bad, is definitely not the best possibility for white in the current position. It might seem ridiculous, but the actual best move in this position is sacrificing the queen by taking the black pawn sitting right in-front.



Queen Kamikaze attack

While it is a fact that the queen could be completely lost when the opponent captures it back with one of his pawns on the 6th rank, all of these opponent responses would lead to a forced checkmate delivered by the white bishop.



5. ChessCo significance and importance for chess learners

What separates ChessCo from other online chess applications?

The unique part about ChessCo is its dedication to creating a personalized online environment for chess learning, which could help users from all different age groups develop a better understanding about the game. The main feature which allows ChessCo to be a better player companion than other apps is the Opening Challenge game mode, which allows users to choose the focus of their learning by engaging in online gameplay modes that start from a chosen by the user opening configuration. By providing such an opportunity, ChessCo actually makes focused learning possible as users are able to play a similar chess position multiple times and thereby remember the theory about it a lot more easier.

Structure and Implementation: What is different and unique about ChessCo's interactive engine compared to other chess engines?

Most chess engines capable of filtering pseudo-legal moves utilize the method of subsequent move simulation in order to determine whether a move is legal or not. In other words, they allow the computer to play all possible moves on the board one by one and then after a long iteration process among all enemy pieces, verify whether the king could be captured on the next move by one of them. If this condition is true, then the given move is excluded from the set of possible destinations for the selected piece to go to and the verification process continues until there are no more possibilities left.

While on a move to move basis this seems to work fine, such implementation can result in significant runtime inefficiencies when later integrated in the evaluation process of a position. In order to return the current best move or retrieve an accurate information about who is winning, the application needs to examine extreme amounts of possible player moves and opponent responses. For a general reference, after the first four moves in a chess game, the amount of possible positions on the board is more than 288 billion! Even with optimizations like alpha beta pruning, the computer still needs to evaluate large numbers of possible scenarios, which is why the illegal moves filtering algorithms need to be highly runtime efficient and easy to execute.

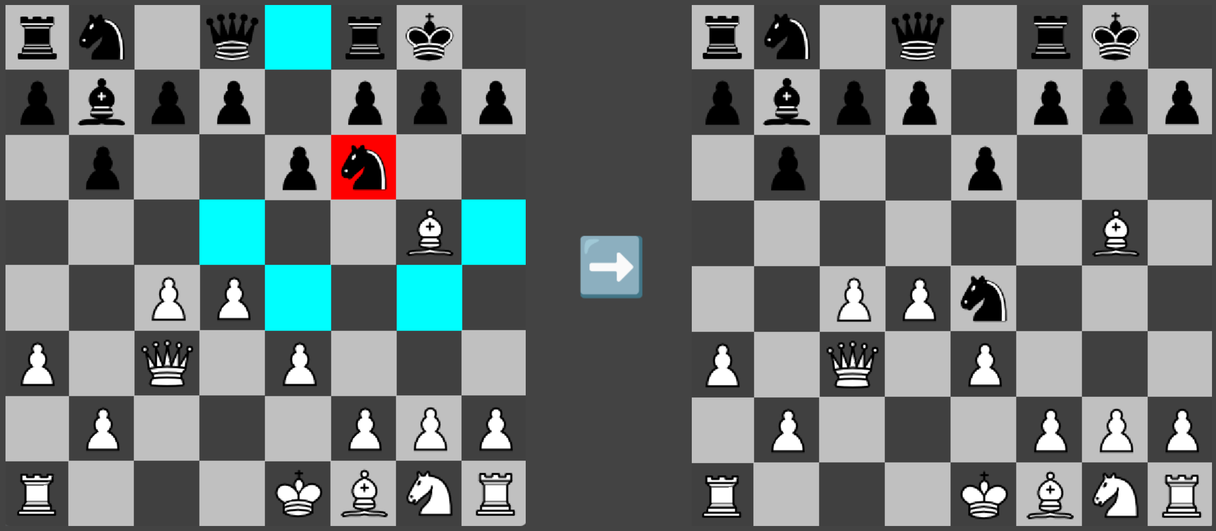
Considering the mentioned above runtime concerns, my chess engine focuses on defining key chess concepts like pinned pieces, checks and checkmate, which greatly optimize the runtime of move generating algorithms and prevent illegal moves from being generated in the first place. This is done mainly by keeping track of both kings' current location on the board and using that information to advance analysis algorithms which detect any of the concepts mentioned above (pinned pieces and potential king threats). More specifically, these algorithms operate by projecting arrays originating from the king's square on the board towards all possible directions and utilize hashing functions and data structures to store and later invoke (in constant time) information about enemy pieces that might be potentially targeting the king or pinning a piece. This way the run time of all pseudo-legal filtering mechanisms is cut more than four times allowing the gameplay and evaluation features to function a lot more smoothly, quickly and efficiently.

6. Scope for version 1.1

The scope for version one of the project includes the implementation of the fundamental game mechanics, render for the board visuals, a FEN parsing feature and also an innovative analysis system for prevention of pseudo-legal moves.

Basic Functionality and Visualization

Board render and fully implemented moving mechanisms for each piece.



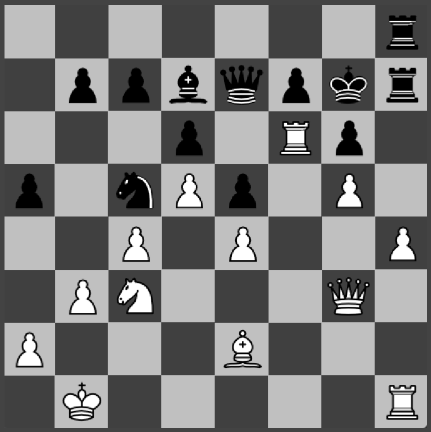
Includes visual representation effects for selected pieces (RED) and projects possible piece destinations (CYAN). The board render functions with any given 60 x 60 pixel PNG image, which allows for customization to the selected theme.

Furthermore, upon clicking on a highlighted (CYAN) square a currently selected piece is moved to that destination. In accordance to the rules of the game, each piece possesses a unique movement pattern, which is why the (say that cyan squares consider pins, friendly pieces, etc)

FEN String Parsing Feature

Function with a FEN string input and generates a board with the given piece configuration on it in constant time O(1).

```
“7r/1ppbqpkr/3p1Rp1/p1nPp1P1/2P1P2P/1PN3Q1/P3B3/1K5R b - - 0 26”
```



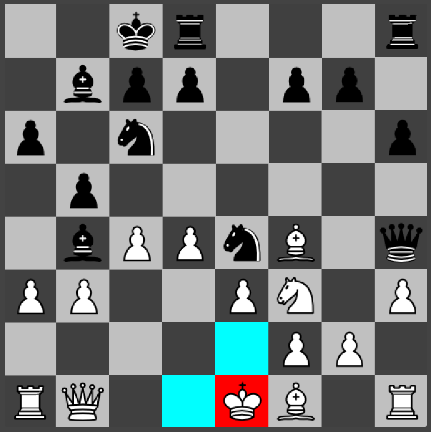
Game: Bobby Fisher - Boris Spassky 1992

System for Filtering Out Pseudo-legal Moves

While in a classical game of chess played on a physical board, it is possible that a player can make a pseudo-legal move like moving a pinned piece, disregarding a checked king or placing the king in scope of enemy piece attack. As I way prevent such actions, this chess engine features intelligent in-game mechanisms, which analyze potential threats in order to disallow the user from committing any of the above mentioned moves.

I. King in Check

Checking the king is a classical game of chess is a move which is heavily utilized by masters as a way for them to gain control over the game as well as to win tempo and develop attacking momentum.
Here in this example, the king is under attack by the enemy bishop, which is why it cannot move to its NW adjacent square. (In this particular example, even if there was no bishop, the enemy's knight would have still prevented the king from going to this square.)



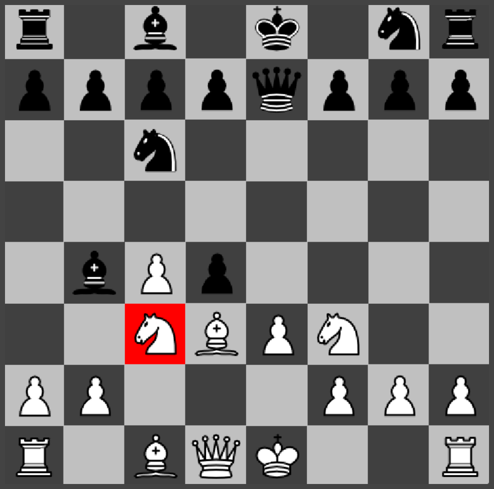
II. Pinned Piece

Pinning pieces is another widely used strategy by high level chess players, which allows them to restrict the movement of their opponent's pieces and really put their opponent's defensive skills to the test.

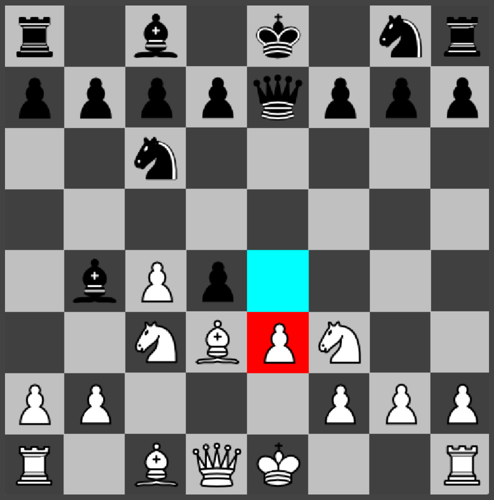
As a reminder, a pinned piece is a friendly piece who is laying in the direction of an attack aimed towards the king. In a scenario when a piece is pinned, the piece cannot move in any direction that would reveal a direct line of attack on the king and allow the enemy piece to capture it.

Note that doing so would furthermore result in a pseudo-legal move. Let's look at some examples of how the pseudo-legal filtering mechanism responds to such circumstances:

Situation 1: Here in the given position below, the knight cannot move anywhere on the board. This is because in the given situation there is an enemy bishop "pinning" the knight at its position as any of its movements will result in a capture of the king.

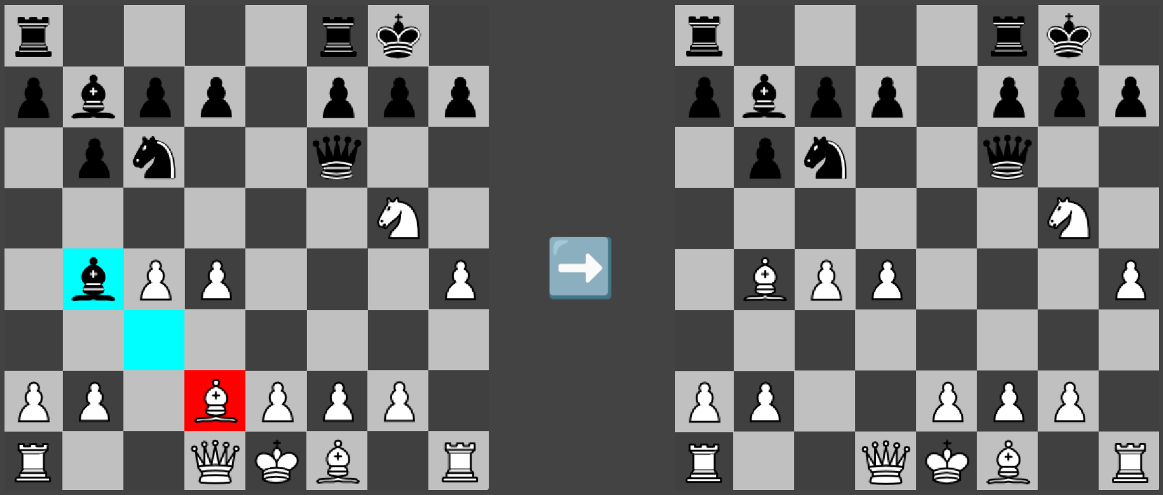


Situation 2: In this scenario a pawn is selected. According to the image, however, it can move upwards but not capture the NW(north-west) adjacent pawn despite being in the right proximity to execute the capture.



This is because the pseudo-legal move filtering mechanism detects the presence of the enemy's queen at a position which could potentially endanger the king's safety. Furthermore the algorithm also understands that the selected pawn is the only piece laying in the direction of the enemy's queen attack and therefore restricts its movement in any direction that would expose the king.

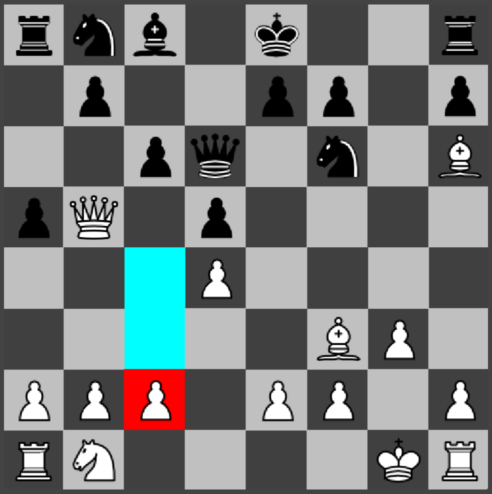
Situation 3: Lastly, the board configuration presented in the image below shows a scenario where the pinned piece can actually capture the attacker and thereby eliminate the threat that is posed on the white king.



Special Movement:

Double Pawn Moves

One of chess' more interesting moves that represents an exception to the basic movement rules is the pawn's ability to move two squares at once on its first ever move. For example:



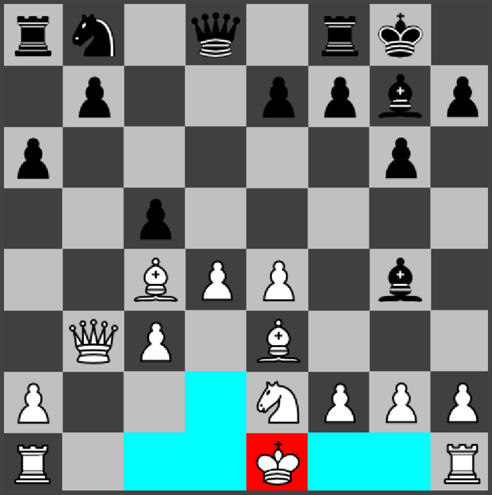
In this case the player can decide whether to move the selected pawn once or twice. The pawn on the 7th file, however, would only be able to move once since it has already left its starting position.

Castling

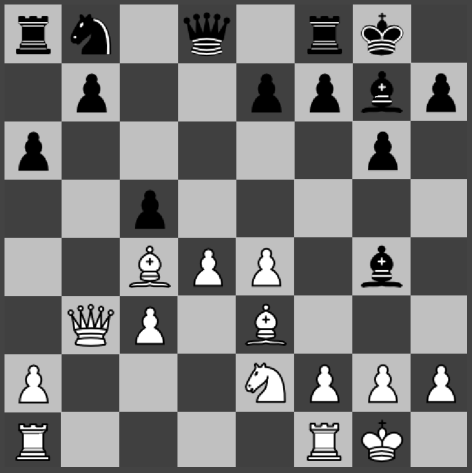
Another one of chess' special moves in Queen and King side castling, which is a rule allowing the King and Rook to swap sides of the board. This move was developed during the 14th and 15th centuries with the intention of increasing the game intensity and speed. Nowadays, castling is firmly integrated in the repertoire of any intermediate chess player as it allows them to quickly prevent the king from being exposed in the center of the board by hiding it in either one of the corners of the board .

Castling, however, requires a set of conditions to be met in order for the move to be considered legal. Those conditions inquire that neither of the pieces (King or Rook) have been moved since the beginning of the game, there are no other friendly or opponent's pieces separating them, the king is not in check and there also is not any enemy piece that can intercept the movement of the pieces along the path of castling.

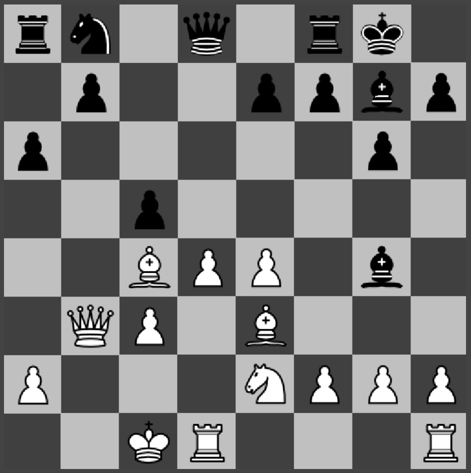
Here is an example of a chess board where both castling options are legal. Notice that the king has two highlighted squares on both of his sides. The castling move itself is defined in the king sliding two squares towards the rook and the rook jumping right besides the king on its opposite side.



After choosing which way to perform the castling maneuver, the board looks like this:

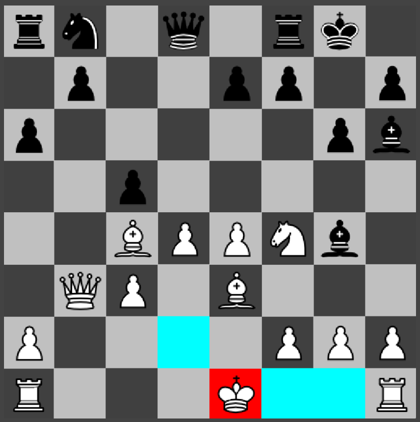


King Side Castle

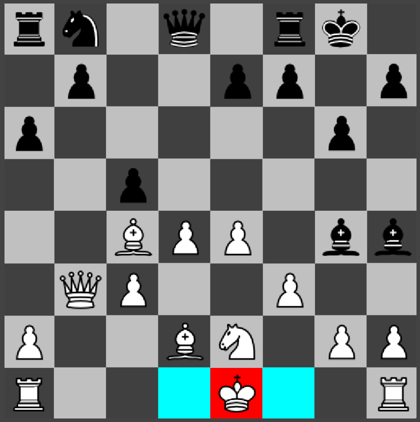


Queen Side Castle

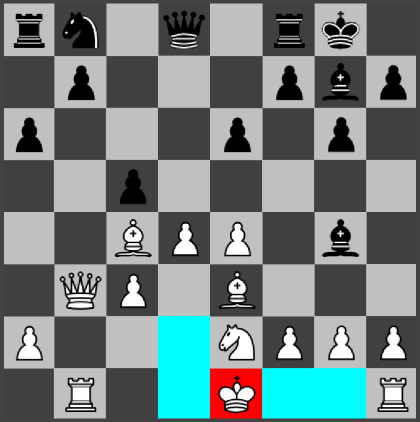
As mentioned above, there are many cases where castling is not actually available to the player. Let’s take a look at some of the more complex situations:



Bishop intercepts Queen Side Castle



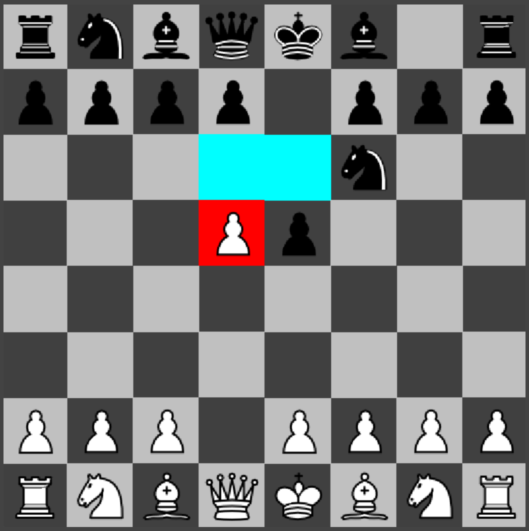
King is in Check



Rook has moved

En Passant

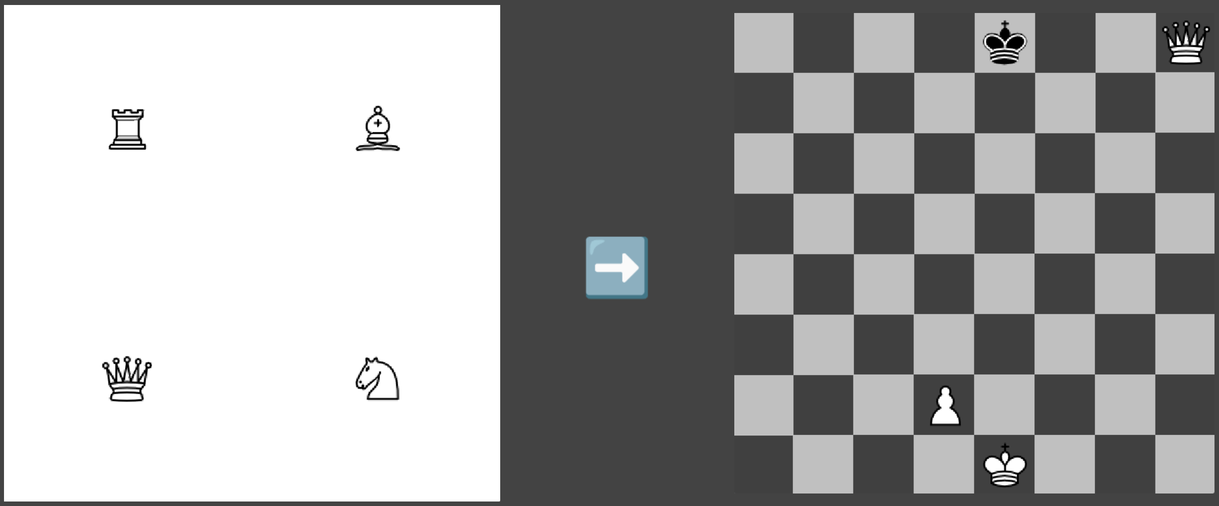
En passant is another special chess move which can happen whenever two opposite color pawns align on either the 4rth or 5th rank and the one of them has just moved two squares up on the previous move. This allows the attacking player to capture the adjacent pawn diagonally from the square behind it. Here is an example:



En passant

Pawn Promotion

Finally, pawn promotion is another special pawn move which allows a pawn to turn into a Queen, Rook, Knight or a Bishop once reaching the final rank on the board. When this happens a pop-up menu appears allowing the user to select a desired promotion piece, while later replaces the pawn at its position on the last rank.



Case where the player has clicked on the Queen promoting option

7. Out-of-scope

The out-of-scope content features all of the functionalities mentioned in the spec but not implemented during v1.1 . Stay tuned for upcoming updated versions of this document.

8.Table of important rules, terms and definitions

- **Check** - a condition that occurs when a player's king is under threat of capture on the opponent's next turn.
- **Checkmate** - any game position in chess and other chess-like games in which a player's king is in check (threatened with capture) and there is no possible escape, nor ways to prevent the capture of the king on the following move.
- **Pinned Piece** - a friendly piece who is laying on the direction path of an attack aimed towards the king. In a scenario when a piece is pinned, the piece cannot move in any direction that would reveal a direct line of attack on the king and allow the enemy piece to capture it.
- **Rank** - Rows
- **File** - Columns

- **Castle -** a special move in chess which allows a king and a rook of the same color to swap their relative sides to each other with the king moving two squares in the direction of the rook and the rook jumping right besides the king in the opposite direction. There are two different types of castling **- queen side** and **king side**, depending on the direction of where the maneuver is happening.
- **Promotion -** a special move in chess, which occurs when a pawn has reached the end of the board. In the event of that happening, the pawn is substituted with a piece chosen by the player, which could be a queen, a rook, a knight or a bishop.
- **Opening -**
- **FEN string -** an official string representation of a chess position, which further brings information about right to castling, possible en passant moves, current player turn as well as current half move and full move count.
- **FIDE -** world international chess federation

[1]

[1] All images in the document have been extracted using ChessCo's rendering feature and are from actual positions runned on my software. User interface and game modes, however, are yet to be implemented. Stay tuned for the next version of the project.