SimpleChess

Rohan Kumar, Michael Terekhov, Roger Finnerty, Joshua Caban, Harlan Jones

Idea

- The idea for this project is an app that allows a user to play games of chess. It will feature :
  - user vs user capability
  - pieces obeying laws of chess
  - proper timing for moves
  - a bot player to play against if there is only one player
- Potential features include :
  - customizable pieces/themes/backgrounds
  - sound effects for piece movements and takes as well as animations for moves
  - predictive move feature
  - ranking play histories
  - different bot player difficulties and tactics

Basic Requirements

- Local device user vs user gameplay
- Local device user vs computer gameplay
- Chess pieces follow traditional movement rules
- Game recognizes whose turn it is
- Game recognizes when a checkmate or stalemate is reached
- Game prompts user to keep playing, quit playing, or change modes upon game ending

Implementation

- Created class called Piece, which has several subclasses (Rook, Queen, etc)
- In main activity, creates 8x8 matrix of Pieces, as well as two 8x8 matrices of TextViews, layered on top of each other (one for background tiles and one for pieces)
- Logic is done on the Piece matrix, and then reflected on the TextView matrices
- Each click program checks whose turn, whether a win has happened, or whether the king is in check
- Bot can be supplemented after each player turn if mode is set to Bot

Additional Features

- Predictive move feature (able to see possible moves)
- Sound effects/animation
- Customizable UI (pieces, board colors, etc)
- Leaderboard ranking system
- Different levels of bot strength/playing types

Implementation

- Predictive move feature is found by finding all possible moves and displaying on board
- Animation done by moving piece square by square over time
    - speed will be able to be set, etc
- Leaderboard is a count of how many wins a certain user has

Bots

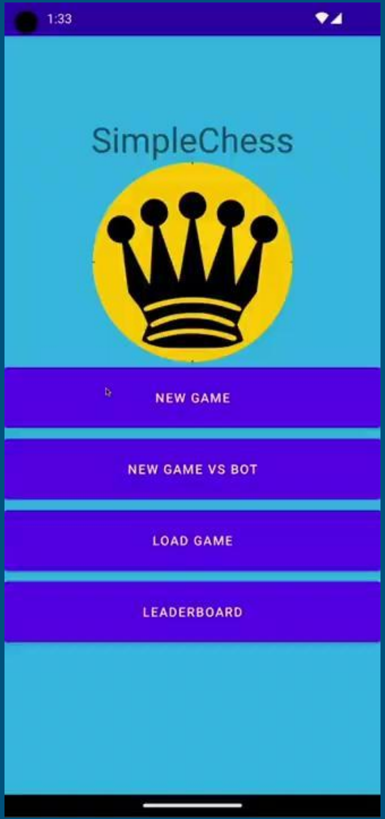
- Random bot
    - finds all possible moves, randomly selects one
    - still can maneuver out of a check
- Aggressive bot
    - prioritizes moves that take pieces, higher value pieces are worth more

- If time permits:
- Location bot
    - ranks each spot on board according to how effective piece is there (e.g.. King is not effective in the middle, but a Queen might be)
    - using coefficient matrix for each piece

Design Decision

- Players should not be able to move in a way that puts them in check
- Leaderboard is propagated by users putting in their name before matches begin
- Being able to return to menu after a game and start another round
- how to save Leaderboard/Settings
    - SharedPreference vs Room (internal storage)

Video Demo if Needed


1:33
SimpleChess
NEW GAME
NEW GAME VS BOT
LOAD GAME
LEADERBOARD