Semester Project  
Mobile Chess Game

Sidney Akers and Will Maxwell

|  |  |  |
| --- | --- | --- |
| Title: Open Menu  Description: Users want an title menu with various selections |  | Title: Local Game  Description – Users may want to play a local game on a single phone |
|  |  |  |
| Title: Random Pairing  Description: Users may want to pair randomly with other users online |  | Title: Specific Pairing  Description: Users may want to pair with specific users online |
|  |  |  |
| Title: Notifications  Description: Users may want to be notified when an opponent plays |  | Title: Available Moves  Description: Users may want to see all available moves for any given piece |
|  |  |  |
| Title: Checkmate notify  Description: Users may want to be notified when they are in check |  | Title: Game Timer  Description: Users may want to play a timed game |
|  |  |  |
| Title: Points  Description: Users may want to keep track of points |  | Title: Chess  Description: Users may want to play chess |
|  |  |  |
| Title: On screen tutorial  Description: Users may want a quick “Howto” on chess |  | Title: Rule varients  Description: Users may wish to enable or disable rule variants |
|  |  |  |
| Title: Remote Server  Description: Users will require remote servers to communicate each other |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Time Estimates

Chess – 10 days – The most essential requirement, the program must run a chess game, this includes board management, rules movements, and win conditions

Points – 1 day – This is a slight modification to the set of rule movements. Each piece if given a value, and when that piece is taken, the point value is added to the users score

Checkmate notify – 1 day – This is a slight modification to the game loop. Before each turn, check to see if either of the kings is in check. Also, prevent users from placing themselves in check via invalid movements.

Available moves – ½ day – This is simple, since the set of rule movements is already defined, simply highlight all available moves when a piece is tapped.

Game Timer – ½ day – Simply start a time for a user when it is his turn. The question is should the timer be running when the app Is in the background.

Open Menu – ½ day – This a simple menu that allows the user to do a couple of things – Execute the game loop, select local or network game, select random or specific partner, allow or disallow rule variants ,or view the how-to. Can’t be that hard.

Local Game – 1 day – This will be built in the “chess” requirement. It will simply allow one color of piece to move, then the other.

Rule variants – ½ day – This is a set of Boolean values that will allow or disallow specific rules. It will include (But is not limited to) allowing castling, first pawn moves twice, and knight jumps pieces.

Specific Pairing – 5 days – This will take extensive examination and practice with the android API; it may also take a server depending on how the api allows two devices to communicate. It appears that there is a utility called google cloud messaging, which may be useful here.

Random pairing – 5 days – This will again take examination and practice with the android api.

Notifications – ½ day – This seems to be built into the android API

Remote Server – 3 days – Google has an implementable server for this

Iteration Schedule

21 days  
 v1.0 – Will hopefully include the Chess requirement, the checkmate notify, the available moves, local game, rule variants, howto, and local game.

42 days

v2.0 – Will include any bugfixes needed in 1.0 as well as specific pairing, random pairing, notifications, game timer, remote server, and points.