Testing

We preformed testing on our program by checking if it met each of the requirements. Requirements specific to the user interface were tested by running the entire program and testing by acting as the user.

The first requirement was to set up the checkers board. The requirements specified an 8-by-8 board of dark and light squares (a light square in the bottom right corner), labels running from A-H along the columns (left to right) and labels 1-8 along the rows (from top to bottom). Testing that these requirements were met was straightforward; the program was executed and the appearance of the board on the interface was compared to what was outlined in the requirements specification.

The next set of requirements concerns the user’s input on setting up the board. The user is given the option to do a standard or custom board setup. This option to the user was tested by running the program and clicking on both the standard setup and custom setup options, ensuring each button put the program into the appropriate state awaiting the next user input. In the state of custom setup, the user must be able to specify custom opening positions for the pieces. This option was tested by selecting the custom setup option and placing the pieces in various arrangements. Arrangements included using only red pieces, only black pieces, mixes of regular pieces and king pieces, and varying number of pieces on the board at once. The requirements also specify the user be warned when attempting to place a piece on an illegal (light) square. This requirement was met by having an alert pop up telling the user he/she cannot place a piece on the square that was clicked. This was tested by running the program as the user and attempting to place a piece on an illegal square. A limit on how many pieces of each colour is also specified. There may never be more than 12 of each colour on the board at once. The requirement was met by implementing a counter (hidden to the user) of how many of each coloured piece is on the board. When the user tries to put a 13th piece of one colour on the board, an alert is shown notifying the user he/she has reached the piece limit. This implementation was tested by running the program as a user and attempting to place a 13th black piece and a 13th red piece, ensuring the desired result was reached in both cases.

The final requirement was to implement a way for the user to indicate that the custom board setup is complete when outside of the standard setup mode. This requirement was met by implementing a button on the user interface labelled “Complete Setup” that when pressed would begin the game in whatever state the board is in. This button’s functionality was tested by running the program as a user and pressing the button after various piece setups and pressing it after a custom setup prior to a standard setup being executed.