Software Engineering Group Project
Use Case Document Group 18

Author:	Archie Malvern [arm36]
Config. Ref.:	UIGroup18
Date:	2023-03-09
Version:	1.2
Status:	Release

CONTENTS

1 INTRODUCTION

1.1 Purpose of this Document

This document aims to specify the use cases of the chess tutor program, to assist in the design of its user interface.

1.2 Scope

Document includes each step different types of users would take during operation of the program, hypothetical background information for said users, and potential error conditions that should be considered in the design of the user interface. This document was produced following the User Interface Specification Standards [1], and is based on the LaTeXtemplate [2].

1.3 Objectives

The objective of this document is to assist in the design and development of the user interface for the chess tutor, ensuring that a wide variety of use cases and error conditions are accounted for.

2 TYPICAL USERS

2.1 Casual Chess Players

This includes players that are completely new to the game or do not play often. These players would want to learn the rules of the game, or at least gain a better understanding. Two subclasses of this general class could be users that are completely new and need to learn the rules from scratch, and users that have played the game before and have a rough understanding of the rules, but may need a reminder or may not be familiar with certain obscure rules. Below is a hypothetical example of a typical user of this class:
Sarah and Gareth are both casual chess players. Sarah has played before, and has a rough understanding of the game's rules, while Gareth has never played before. Sarah uses the tutor's display to quickly remind herself which moves are valid with which pieces, and to learn about new manoeuvres that are otherwise quite obscure. Gareth carefully looks at the display to work out which moves he can use with which pieces to memorise the roles of each different piece.

2.2 Advanced Chess Players

This includes players that play the game frequently, and are highly competent at chess. These players would want to use the program as a standard chess program, perhaps using it to remind themselves of some more obscure rules if need be. Below is a hypothetical example of a typical user of this class:
Morgan and Sam are two competitive chess players. They use the program to play a game of chess against each other, then use the replay functions to analyse their strategies and improve their performance in future matches. They also upload their configuration files online for other players to view and take inspiration for their own chess matches, and download them to review themselves.

3   USE CASES

3.1   UC-1.1 Starting a game (FR1)

One player opens the program and clicks the *New Game* menu option. One player enters their name and picks whether they're playing white or black, and clicks *Next*. The other player types their name and is assigned the opposite colour to the first player. The players are then presented with a chess board interface.

3.2   UC-2.1 Playing a game (FR3, FR4, FR5)

The player who has selected white goes first in accordance to the rules of chess, and selects a piece. The potential spaces said piece can move are highlighted with grey dots. The player can then click on one of these spaces to move the piece. The current player's name is shown in the corner alongside the colour they have chosen.

3.3   UC-2.2 Player attempts to move a piece to an invalid space (FR5)

If the player selects an invalid space, the piece will not move.

3.4   UC-2.3 The player is put in check or checkmate (FR6, FR7)

When a player is in check, it highlights the background of the king piece alongside the piece putting it in check in red; the same is done for checkmate, but the game is ended and the word *Checkmate* is displayed on screen, alongside the name of the winner, i.e. "*Checkmate! Mary Wins!*". The player is then prompted to start a new game, replay the game or quit.

3.5   UC-2.4 A player wishes to offer a draw or resign (FR8)

On the game screen there are draw and resign buttons, where a player can choose to offer a draw or resign the game. When the draw button is selected, the other player is prompted to either accept or reject the draw.

3.6   UC-2.5 Castling (FR5)

If neither the king or one of the rooks have moved, the king can move two spaces towards an unmoved rook, and the rook will then move to the space the king passed over. This can be done kingside or queenside, meaning the rook can move two or three spaces. This is indicated using grey dots to denote it as a valid move, as in *UC-2.1*.

3.7   UC-2.6 En Passant (FR5)

If an enemy pawn moved two squares on the previous turn, the player's pawn can capture the enemy pawn over the square it passed over. This is visually indicated using the valid moves display established in *UC-2.1*.

3.8 UC-2.7 Pawn Promotion (FR5)

If a pawn moves to the other side of the board, it can promote to any piece except a king. This is done through a pop-up menu, which lists each piece the pawn can promote to. The player then selects the piece they wish to have the pawn promote to.

3.9 UC-3.1 Continuing a game (FR11)

A player can click the *Continue* menu option when the program starts, to select a previously saved game. The player can select one of these files and click *Load* to load the game back to it's exact previous state, or select *Back* to return to the previous screen.

3.10 UC-3.2 Viewing a previous game (FR10)

A player can select the *Replay* option to open a file explorer window and find a previous game. The player can then select a save and click the *View* button to open a chess board interface with a *Pause*, *Play*, *Previous* and *Next* button. The *Play* button plays each turn in order, the *Pause* button pauses on the current turn, the *Previous* button rewinds to a previous turn and the *Next* button advances to the next turn.

3.11 UC-3.3 Exiting the game (FR9)

The player can exit a running game by clicking on the cross icon in the top right corner of the program window, or by clicking the exit button on the menu screen. A pop up dialogue will ask the player to confirm they would like to quit, and if the *Confirm* button is clicked, the program will cease. The player can also press the *Escape* key to bring up a dialogue asking if the player would like to quit, and whether they would like to save the game or not.

4 ERROR CONDITIONS

4.1 EC-1.1 The player attempts to move a piece to an invalid space

This error is triggered when the player tries to make an invalid move. The piece will not move to the selected position. A message is not necessary for this error.

4.2 EC-1.2 Two players pick the same name

When two players try to pick the same name, a hyphen and the player's colour is added to the end of the name, i.e. ”*James-White*” and ”*James-Black*”. This is in order to avoid conflicts.

4.3 EC-1.3 File reading error

If the save file being read in is invalid, an error message is displayed then the program ceases to read in the file and returns to the menu.

4.4 EC-1.4 Player tries go to a turn that does not exist

The *Next* button on the last turn of playback, or *Previous* button on the first turn of playback, is greyed out so the user cannot view a turn that does not exist.

REFERENCES

[1] *Software Engineering Group Projects* User Interface Specification Standards. C. W. Loftus, SE.QA.04. 1.4 Release.

[2] *Software Engineering Group Project* LaTeX Document Example. N.W. Hardy, C.W. Loftus, SE-N66-TEST. 1.4 Release.

DOCUMENT HISTORY

Version	Issue No.	Date	Changes made to Document	Ch
0.1	N/A	2023-02-07	Initial draft document	arn
0.2	N/A	2023-02-07	Added an additional use case	arn
0.3	N/A	2023-02-14	Added some early error conditions	arn
0.4	N/A	2023-02-20	Made the document more consistent with the presentation	arn
0.5	18	2023-02-26	Prepared the document for review	arn
1.0	18	2023-02-28	Initial release	arn
1.1	N/A	2023-03-08	Made improvements in accordance to Chris' feedback	arn
1.2	N/A	2023-03-09	Added a mid-game quit dialogue	arn
1.3	N/A	2023-05-10	Changed the specification to be more consistent with the final program	arn