COMP 2005

Iteration 1

Submitted by: Francois Low

Actors and goals

Actor	Goals
Player (Primary)	Choose difficulty levels, Choose board settings, Save game, Load game, Pause game, Resume game, Choose color settings, Enable/Disable hint settings, Flip target chip, Choose number of robot moves, Select number of human players, Restart game, Move robot piece
Computer system (Supporting)	Keep track of moves, Record points earned by player, Countdown timer, Start/end gameplay, Record players' move selection, Select the lowest robot moves, Assign points to player, Apply barrier effect on particular color of the robot, Verify move is legal, Verify target chip match the symbol on the board, Reveal moves of each player, Display hint of each round, Deploy respective computer players based on difficulty levels chosen, Save the last checkpoint, Halt the execution from computer players, Restrict interaction of GUI from human players, Select the winner

Set up a game

Primary Actor: Player

Stakeholders and Interests:

- Player: wants to set up a game with pre-specified settings
- Computer system: wants to ensure that the gameplay is set up with the predefined settings defined by the player.

Precondition:

- Game application has been successfully booted up

Success Guarantee (Post Conditions):

- The user is able to see the expected specifications in the gameplay. The user is able to start with the gameplay.

Main Success Scenario:

1. The system provides the user with the options of creating a new game, continue from the previous saved game or to exit the game. [*Use Case Ends*]

2. The user selects create a new game to start a new game.

3. The system provides the user the options of selecting the number of human players. [*Alt1: Select only 1 human player*] [*Alt2: Select 4 human players*]

4. The user selects the number of human players for which he or she wants.

5. The system provides the user with two difficulty options to choose from, Normal or Extreme mode. [*Alt1: Select Normal mode*] [*Alt2: Select Extreme mode*]

6. The user selects the difficulty level that he or she wants

7. The system provides the user with the two options of selecting the color settings, Default or Altered color settings (Zesty/Elegant/Retro). [*Alt1: Select Default setting*] [*Alt2: Select Altered color setting*]

8. The user selects the color setting that he or she wants.

9. The system provides the user with two board options to choose from, Simple or Complex. [*Alt1: Select Simple board*] [*Alt2: Select Complex board*]

10. The user selects the board option that he or she wants.

11. The system provides the user with the options of enabling/disabling the hint setting. [*Alt1: Enable hint*] [*Alt2: Disable hint*]

12. The user selects the hint setting that he or she wants.

13. The system set up the gameplay with the predefined setting by the user. [*Use Case Ends*]

Alternatives

Alt 1: User creates a new game, wants only 1 human player, extreme mode, default color setting, played on a simple board, with hint setting enabled.

1. Resume at Step 3:
2. After Step 3: The user selects the option of having 1 human player.
3. After Step 5: The user selects extreme mode.
4. After Step 7: The user selects default color setting.
5. After Step 9: The user selects simple board.
6. After Step 11: The user enables hint setting.
7. Flow resumes at Main Success Scenario at Step 11.

Alt 2: User wants 4 human players, normal mode, altered color settings (Zesty/Elegant/Retro), played on a complex board, with hint setting disabled.

1. Resume at Step 3:
2. After Step 3: The user selects the option of having 4 human players.
3. After Step 5: The user selects normal mode.
4. After Step 7: The user selects one of the altered color settings.

5. After Step 9: The user selects complex board.
6. After Step 11: The user disables hint setting.
7. Flow resumes at Main Success Scenario at Step 11.

Exceptions:

- If at any time the system is unable to display options, start the gameplay or crash in the middle of selecting options, the system informs the user of the problem, attempts to reboot the program and the use case ends.

Special Requirements:

- Colors of the robots and board elements used must provide for the visually impaired (color blindness).

Open Issues:

- Is the difficulty level of the game based on the complexity of the board or the computer players?

- Does the difficulty level affect the game if the user chooses 4 human players?

- If loading a previous saved game is allowed, what type/format of the game file is being stored? And how is the file being stored ?

Take a Turn

Primary Actor: Player

Stakeholders and Interests:

- Player: wants to be able to specify the number of moves required for the robot piece to reach the target symbol on the board, wants to be able to demonstrate the steps to reach the target symbol on the board by moving the robot piece, wants to be allocated the correct amount of time during one's turn.

- Computer System: wants to accurately allocate current amount of time for every player's turn, wants to accurately record every player's moves and specified number of moves, wants to ensure player's move is legal, wants to provide correct hints to the players when hint setting is enabled and omit the display of hints when hint is disabled, wants to display different names of the players to distinguish between human players and also computer players.

Preconditions:

- Game settings has been predefined by the player and gameplay has started.

- Target chip has been flipped.

Success Guarantee (Postconditions):

- Player is aware of his/her chosen moves and specified number of moves. Robot piece is moved according to the player's

predetermined steps. Each player's moves are recorded and revealed to other players.

Main Success Scenario:

1. The system displays the name of the player during his/her/its turn, and countdown timer is initialized.

2. The player moves the robot piece to the target symbol on the board. [*Alt 2: Player ran out of time*]

3. The system display and deduct the number of moves that the player has left at every step that the robot piece takes.

4. The player has successfully moved the robot piece to the target symbol on the board. [*Alt 1: Player fails to move robot piece to target symbol on the board*]

5. The system records and displays the number of steps that the player has taken.

6. The system chooses and display the player with the lowest number of steps. [*Alt 3: End of the first/second/third player's turn*]

7. The system increments the player with the lowest number of steps by one.

8. The system display and informs the user the game is proceeding to the next round.

9. The system flips and displays the target chip. [Use case ends]

Alternatives:

Alt 1: Player fails to move robot piece to target symbol on the board

1. The system informs the user that he/she/it has failed to move the robot piece to the target symbol on the board.

2. The system removes the player from selection of the winner for that round.

3. The system switch to the next player's turn.

4. Flow resumes on Step 1.

Alt 2: Player ran out of time

1. The system informs the user that he/she/it ran out of time.

2. The system switch to the next player's turn.

3. Flow resumes on Step 1.

Alt 3: End of the first/second/third player's turn

1. The system records the current player's number of moves.

2. The system switch to the next player's turn.

3. Flow resumes on Step 1.

Exceptions:

- If at any time the system is unable to switch between players' turns, keep track of the gameplay, the system informs the user of the problem, attempts to reboot the program and the use case ends.

Special Requirements:

- Colors of the target chip and board elements must be distinguishable to the visually impaired users (color blindness)

Open Issues:

- Does the system prompt the players to specify the number of steps needed to reach a target symbol on the board first and then proceed with moving the robot piece on the board or does the system allows the players to move the robot piece during their turns while it keeps track of the number of steps that the robot piece moves and at the same time records the path that the players took ?