Use Case Descriptions

Take a Turn:

Primary Actor

Player of game

Stakeholders and Interest

Player(who is making turn)- wants to take a turn with all legal moves, wants the option to keep making moves after a move is made, wants the game to be user friendly.

Other Players- wants to see how many move it toke the user to reach the destination, wants there to be no cheating involved, wants all moves to be clear and visible to see.

Developers- wants the game to be fun and easy to play, along with being easy to maintain.

Pre conditions

1. A game piece destination is flipped.

2. Timer starts.

Post Conditions

Players are aware of each move that is made and how many move it takes to get to the correct destination on the game board. Points of each player is updated.

Main Success scenario

1. The user selects how many moves it will take to reach the destination.

2. The system checks if any other players or computer players can do this in fewer moves.[Alt 1, Another player can do it in fewer move, Use case ends]

3. The system displays that you can do this in the fewest amount of moves.

4. The user selects which robot it would like to move.

5. The user selects which direction it would like to move.

6. The system checks to see if the move is legal.(ex. Going up with a robot directly above you).[Alt 2, an illegal move was made, system tells the user to move again]

7. The system moves the robot and updates the gameboard.

8. The system updates the move counter by 1.

9. The system checks to see if the correct robot has reached the destination in the correct amount of moves. [Alt 3, Correct robot reached it's destination in correct amount of moves or you exceeded the amount of moves that you initially said, use case ends]

10. The system informs the user that they have not reached it's destination and offers the user to end move [Alt 4 ,end case], or to move the same robot or a different robot.

11. Back to step 1

Alternate Flow

Alt 1, Another player can win this turn in fewer moves, use case ends.

Alt 2, the user informs the user that they try to move the robot where the robot could go and tells the user to move again. Back to step 6

Alt 3, Correct robot reached it's destination in correct amount of moves(user score goes up by one) or you exceeded the amount of moves that you initially said, use case ends.

Alt 4, The user elects to quit or save the game, Use case ends.

Exceptions

Exceptions,

1. if at any time the system can not retrieve the gameboard or the input from the user, the system informs the user that there is a problem and exits the game. Use case ends.

2. If the system can not retrieve the move direction indicated by the user, the system will ask the user to try again. Back to step 5.

Special Requirements:

Colours and sizes of text fonts used must provide - or be able to provide - for the visually impaired. For example color blindness.

Open Issues

Can a user select to move a robot that it already moved during that same turn?

What is the user is very young and is a poor reader along with being color blind?

Who gets the point if the player exceeds the amount of moves that they initially said?

Set up a Game:

Primary Actor

Player

Stakeholders and interest

Player – wants to get the game setup easily and quickly, wants to have a option of how many people can play, wants to be able to choose the difficulty of the game.

Developer – wants the setup process to be understandable and zero issues starting the game.

Preconditions

Players are identified.

Postconditions

Players are aware of the difficulty chosen. Players are aware of how many human and computer players are playing.

Main success scenario

1. The user selects to start game, to resume game, help or exit the game.[Alt 1, The user exits ,ask for help or resumes a game].

2. The system starts the game process.

3. The system asks for the number of human players.

4. The user confirms the number of human players.

5. The system checks if the number of players are valid and fills the remaining players as AI.[Alt 2, user entered more players then 4 or less players then one]

6. The system displays an option to play with text on top of the robots(text is needed for color blind).

7. The user selects which option it wants to play with.

8. The system displays the two different board difficulties.

9. The user selects which difficulty that it would like to play.

10. The system displays the game board depending on the difficulty selected.

11.The system randomly places the robots on the board.

Alternate Flow

Alt 1, The user chooses to exit the game or to resume a previous game. Use case ends.

Alt 2, user entered more players then 4 or less players then one. The system tells the user to select a valid number of players. Resume from step 4.

Exceptions

If at any time the system is unable to retrieve, record or provide details then the system informs the user of the problem, attempts to record the time and nature of the failure and the use case ends.

If the system can not load, save, or resume a game. The system will communicate the error to the user and exit the game. Use case ends.

Special Requirements

Colours and sizes of text fonts used must provide - or be able to provide - for the visually impaired. For example color blindness.

Open Issues

What if there are no saved games, would the option be given to resume a game?

Should there be a option to click single player rather then enter 1 human players?