

|  |  |
| --- | --- |
| **Flow of events for the Chance time** **use case** | |
| **Objective** | Player is given opportunity/setback |
| **Pre-Condition** | Player must land on chance square |
| **Main Flow** | 1. Player lands on square  2. Player finds out what affect chance can have **include(analyse space)**  3. Player moves cursor to randomise chance button  4. Player presses mouse button  5. Player receives chance  6. Chance takes effect |
| **Alternative flows** | At 5, if chance requires to be stored for later use it is stored |
| **Post-Condition** | Player ends their turn |

|  |  |
| --- | --- |
| **Flow of events for the Get rules use case** | |
| **Objective** | Player is given list of rules |
| **Pre-Condition** | Player must click on rules option |
| **Main Flow** | 1. Player selects the get rules option  2. Rules appear on screen  3. Player moves mouse button  4. Player clicks on exit rules |
| **Alternative flows** | - |
| **Post-Condition** | Player returns to game menu |

|  |  |
| --- | --- |
| **Flow of events for the Fundraise Money use case** | |
| **Objective** | Player to fundraise money |
| **Precondition** | The player lands on a fundraising tile |
| **Main Flow** | 1. Player rolls dice  2. Player lands on a fundraise money tile  3. Player finds out about the space **include (analyse space)**  4. Player enters how much time they will fundraise money  5. Game adds money earned to the player |
| **Alternative Flows** | At 5 The Player enters no time then no fund raising will occur |
| **Post-condition** | Player has successfully collected money |

|  |  |
| --- | --- |
| **Flow of events for the Win Game use case** | |
| **Objective** | Player to win the game |
| **Precondition** | Player reaches x amount of percentage bar |
| **Main Flow** | 1. Player takes turn.   1. 2. Player ends turn. 2. 3. Player completes the percentage bar. 3. 4. Game displays “Game won”. 4. 5. Player exits the game. |
| **Alternative Flows** | - |
| **Post-condition** | Player wins the game and game ends |

|  |  |
| --- | --- |
| **Flow of events for the *Exit* *Game use case*** | |
| **Objective** | The game comes to an end |
| **Precondition** | The Player is in control of the game |
| **Main Flow** | 1.  1. Player clicks exit.  2.  2. The Player is asked if they want to exit.  3.  3. The Player clicks ‘yes’.  4.  4. Application closes. |
| **Alternative Flows** | At 3, if the player clicks no Game does not end and continues |
| **Post-condition** | The game has ended |

|  |  |
| --- | --- |
| **Flow of events for the Start Game use case** | |
| **Objective** | To begin a game. |
| **precondition** | There are several Players already created. |
| **Main flow** | 1. The Player runs the program.  2. The home menu appears.  3. The Player is asked do they want to start.  4. The Player presses the start button.  5. The game asks if they want to add additional players **include(Create Player)** or continue.  6. The Player presses continue |
| **Alternative Flows** | At 4 the Player can enter a different option leading to create player.  At 4 the Player can enter a different option leading to remove player.  At 4 the Player can enter a different option exiting the program. |
| **Post-Condition** | The game starts and the board appears. |

|  |  |
| --- | --- |
| **Flow of events for the Create Player use case** | |
| **Objective** | To Create a Player |
| **precondition** | The Player is required to start the program. |
| **Main flow** | 1. The Player runs the program.  2. The home menu appears.  3. The Player is presses the create player button.  4. The Player enters the player’s name.  5. The Player will press enter.  6. The Player is returned to the home menu. |
| **Alternative Flows** | At 4 the Player can enter a different option leading to start game.  At 4 the Player can enter a different option leading to remove player.  At 4 the Player can enter a different option exiting the program. |
| **Post-Condition** | The Player is asked to enter player details. |

|  |  |
| --- | --- |
| **Flow of events for the spend resources use case** | |
| **Objective** | Increase the progress of a space |
| **Precondition** | Landed on an interactable tile |
| **Main Flow** | 1. the Player finds out about the space **include(analyse space)**  2. The Player selects spend resources  3. The Player enters the amount  4. Player presses spend  5. The spaces progress gets increased relative to the amount spent  6.The resources spent are taken away from the Player  7.Points will be added to the completion goal |
| **Alternative Flows** | At 3, if Player doesn’t have enough time or money, error message appears and the space remains the same  At 6, if certain squares are landed on they will affect the board instead of adding points to the goal |
| **Postconditions** | The spaces progress is increased |

|  |  |
| --- | --- |
| **Flow of events for the analyse space use case** | |
| **Objective** | Find out about the space being landed on |
| **Precondition** | Player is in control |
| **Main Flow** | 1. Player moves mouse over desired square  2. Player clicks on space  3. Player presses analyse button  4. window appears containing the information about that space  5. Player presses close button to return to the board |
| **Alternative Flows** | At 3, if no square is selected nothing happens |
| **Postconditions** | Player knows more about the space |

|  |  |
| --- | --- |
| **Flow of events for the Start turn use case** | |
| **Objective** | Player starts their turn |
| **Pre-Condition** | The previous Players turn must have ended |
| **Main Flow** | 1.  1. Previous Players turn has ended  2.  2. It is now the Players turn  3.  3. Player clicks to roll dice for turn  4.  4. Player moves tile  5.  5. Player spends resources  6.  6. Players turn ends |
| **Alternative flows** | At 2, If the Players turn must be skipped it will be the next Players turn  At 4, Player can choose to nothing on their turn |
| **Post-Condition** | Player’s progress from their turn is recorded |

|  |  |
| --- | --- |
| **Flow of events for the Lose game use case** | |
| **Objective** | End a lost game |
| **Precondition** | Player has run out of money |
| **Main Flow** | 1.Player ends their turn  2.Money is taken from the player resulting in a negative balance  3.Game displays “game over”  4.game says click to exit  5.Player clicks exit |
| **Alternative Flows** | - |
| **Postconditions** | Player is returned to start menu |

|  |  |
| --- | --- |
| **Flow of events for the end turn use case** | |
| **Objective** | End the current players turn |
| **Precondition** | Player has finished everything they want to do that turn |
| **Main Flow** | 1. Player clicks end turn button  2. Popup asks if the player is sure  3. Player clicks yes  4. Game takes money away from player  5. Game says next player’s name  6. Control of the game is given to the next player |
| **Alternative Flows** | At 3, if the player clicks no then they can interact with the board again |
| **Postconditions** | A new player is taking their turn |

