|  |  |
| --- | --- |
| **Flow of Events for the *Pre-game Setup* use-case** | |
| **Objective** | To register each player’s name and initialize their status. |
| **Pre-condition** | The game is successfully loaded. |
|  |  |
| **Main Flow (for each player)** | 1. The system display current player’s number. |
| 2. The player type in their name. |
| 3. The player is given a designated amount of money and security value. |
| 4. Next player repeat the process until every player is registered. |
|  |  |
| **Alternative** **Flows** | At 2, if the name typed in is illegal (contain illegal character or too long), the system displays a warning, player is asked to type in again. |
|  |  |
| **Post-condition** | Player is ready for gameplay. Game-play menu will display. |

|  |  |
| --- | --- |
| **Flow of Events for the *Game Play* use-case** | |
| **Objective** | To allow player select one of the game-play actions. |
| **Pre-condition** | The player has done pre-game registration. |
|  |  |
| **Main Flow** | 1. The system display all available game-play options, plus “next player” option. |
|  |  |
| **Alternative** **Flows** | At 1, if the player selects “next player” option, the Game-play Menu will display for the next player. |
|  |  |
| **Post-condition** | Player will start the selected game-play action. |

|  |  |
| --- | --- |
| **Flow of Events for the *Roll Dice* use-case** | |
| **Objective** | To determine the next tile the player will land on. The player could go to jail if unlucky. |
| **Pre-condition** | The player selected “Roll Dice” in the Game-play Menu. |
|  |  |
| **Main Flow** | 1. The player choose to start rolling dice. |
| 2. The system returns the result (1 to 6). |
| 3. The player lands on a new tile according to the result. |
| 4. The player pays rent and lose security value if the new tile is owned by another player. The more servers there is on the new tile, the more rent the player should pay, the more security value the player will lose. |
| 5. The system display the game-play menu again based on the updated location. |
|  |  |
| **Alternative** **Flows** | At 3, if the player lands on a jail tile, Go Jail use case will be utilized. |
| At 4, if the player loses all their security value, they will be out of the game. |
|  |  |
| **Post-condition** | Player’s location is updated. |

|  |  |
| --- | --- |
| **Flow of Events for the *Go jail* use-case** | |
| **Objective** | To stop the player from doing anything or moving for three rounds. |
| **Pre-condition** | The player is sent to jail by landing on a jail tile. |
|  |  |
| **Main Flow** | 1. The system shows a message saying the player is now in jail and explaining the rule of jail. |
| 2. The player will be asked if they want to pay bribe to get out of jail |
| 3. If bribe is not paid, the player will remain in jail for three rounds. |
|  |  |
| **Alternative** **Flows** | At 2, if bribe is paid, the player will be allowed to roll dice to go out of jail. |
|  |  |
| **Post-condition** | Player will be assigned “in jail” status. |

|  |  |
| --- | --- |
| **Flow of Events for the *Buy Company* use-case** | |
| **Objective** | To buy a branch of the company (represented by a tile) |
| **Pre-condition** | The player has rolled a dice and landed on the tile. |
|  |  |
| **Main Flow** | 1. The system check if the player has already brought the company branch. |
| 2. The price for the company branch is displayed. |
| 3. The system ask the player if they want to buy the company branch or not. |
| 4. The company branch is marked as owned by the player. The player’s money is deduced. |
|  |  |
| **Alternative** **Flows** | At 1, If the player already owns the company branch, the system displays a message and return to the game-play menu. |
| At 3, If the player does not have enough money, the system displays a warning and return to the game-play menu. |
|  |  |
| **Post-condition** | Player will be able to build servers on the tile if they own every branch of the company. |

|  |  |
| --- | --- |
| **Flow of Events for the *Auction Company* use-case** | |
| **Objective** | To put a branch of the company (represented by a tile) on auction. |
| **Pre-condition** | The player has rolled a dice and landed on the tile. |
|  |  |
| **Main Flow** | 1. The system check if the player has already brought the company branch. |
| 2. The player is asked to put a starting price |
| 3. Any player may put a higher bid within a certain amount of time. |
| 4. The last bidder when the time is up is the winner |
| 4. The company branch is marked as owned by the winner. The winner’s money is deduced. |
|  |  |
| **Alternative** **Flows** | At 1, If the player already owns the company branch, the system displays a message and return to the game-play menu. |
| At 3, If the bidder enters an amount that is not greater than the current bid, or it is exceeding their total money, the bid will not be allowed. The system displays a warning, and players are asked to bid again. |
|  |  |
| **Post-condition** | The winner will be able to build servers on the tile if they own every branch of the company. |

|  |  |
| --- | --- |
| **Flow of Events for the *Build Servers* use-case** | |
| **Objective** | To build a server on the selected tile. |
| **Pre-condition** | The player has rolled a dice and landed on the property. |
|  |  |
| **Main Flow** | 1. The system lists every tile the player owns. |
| 2. The player select one of the tiles. |
| 3. The system prints out how much money and security level it will cost to build a server and ask for confirmation. |
| 4. The player confirms the decision. |
| 5. The number of servers on the selected tile increases by one. |
|  |  |
| **Alternative** **Flows** | At 1, If the player does not own any tile. The system displays a warning. Player will return to the game-play menu. |
| At 2, If the selected tile already has 4 servers or a mainframe built onto it. The system displays a warning. Player will return to the game-play menu. |
| At 4, if the player regrets the decision, Player will return to the game-play menu. |
|  |  |
| **Post-condition** | Player will return to the game-play menu. |

|  |  |
| --- | --- |
| **Flow of Events for the *Build Mainframe* use-case** | |
| **Objective** | To build a mainframe on the selected tile.  (note: a mainframe can only be built if there are already 4 servers, and the player cannot build more than one mainframe) |
| **Pre-condition** | The player has rolled a dice and landed on the property. |
|  |  |
| **Main Flow** | 1. The system lists every tile the player owns. |
| 2. The player select one of the tiles. |
| 3. The system prints out how much money and security level it will cost to build a mainframe and ask for confirmation. |
| 4. The player confirms the decision. |
| 5. The number tile now has a mainframe. |
|  |  |
| **Alternative** **Flows** | At 1, If the player does not own any tile. The system displays a warning. Player will return to the game-play menu. |
| At 2, If player has less than 4 servers already built on the tile, or there is already a mainframe, Player will return to the game-play menu. |
| At 4, if the player regrets the decision, Player will return to the game-play menu. |
|  |  |
| **Post-condition** | Player will return to the game-play menu. |

|  |  |
| --- | --- |
| **Flow of Events for the *Lease Property* use-case** | |
| **Objective** | To regain half of the money and security level that has spent of the property, with the cost of no longer able to hack other players when they land on it. |
| **Pre-condition** | The player has rolled a dice and landed on the property. |
|  |  |
| **Main Flow A** | 1. The system list every property (servers or mainframe) the player owns. |
| 2. The player select one of the properties. |
| 3. The system display the money and security level the player will regain and ask for player’s confirmation. |
| 4. The player confirms the decision. |
| 5. The tile is marked as “leased”. |
|  |  |
| **Alternative** **Flows** | At 1, if the player does not own any property. The system will display a warning. The player cannot do anything until next round. |
| At 4, if the player regrets the decision, nothing will change. The player can do nothing until next round. |
|  |  |
| **Post-condition** | The player gained money and security level. Player will return to the game-play menu. |

|  |  |
| --- | --- |
| **Flow of Events for the *Sell Properties* use-case** | |
| **Objective** | To sell properties (server and mainframe) the player owns. |
| **Pre-condition** | The player selected “Sell Properties” in the Game-play Menu. |
|  |  |
| **Main Flow** | 1. The system lists every server the player owns. |
| 2. The player select one of the servers. |
| 3. The system prints out how much money and security level the player will gain by selling the server and ask for confirmation. |
| 4. The player confirms the decision. |
| 5. The number of servers on the selected tile decreases. |
|  |  |
| **Alternative** **Flows** | At 1, If the player does not own any servers. The system displays a warning. Player will return to the game-play menu. |
| At 4, if the player regrets the decision, Player will return to the game-play menu. |
|  |  |
| **Post-condition** | The player gained money and security level but lost the property being sold. Player will return to the game-play menu. |

|  |  |
| --- | --- |
| **Flow of Events for the *Trade with Others* use-case** | |
| **Objective** | To allow the player sell any of property they own by making an offer to other players |
| **Pre-condition** | The player selected “Trade with Others” in the Game-play Menu. |
|  |  |
| **Main Flow** | 1. The system list every property (servers or mainframe) the player owns. |
| 2. The player select one of the properties and enter a price to make an offer. |
| 3. The system list every other player |
| 4. The player select a target player to send an offer. |
| 5. The target player either accept or decline the offer. If the offer is accepted, money is transfer to the target player, while the property is transferred to the current player. |
|  |  |
| **Alternative** **Flows** | At 1, If the player doesn’t own any property. The system displays a warning. Player return to the game-play menu. |
| At 2, If the player enters an illegal amount of money (e.g., £-1000). The system displays a warning. Player will return to the game-play menu. |
| At 4, If the target player declines the offer. The system displays a message. Player will return to the game-play menu. |
|  |  |
| **Post-condition** | Player will return to the game-play menu. |

|  |  |
| --- | --- |
| **Flow of Events for the *Use Wildcards* use-case** | |
| **Objective** | To allow the player select one type of wildcards to use. |
| **Pre-condition** | The player selected “Use Wildcards” in the Game-play Menu. |
|  |  |
| **Main Flow** | 1. The system display five types of wildcards, plus “return to game-play menu” option. |
|  |  |
| **Alternative** **Flows** | At 1, if the player selects “return to game-play menu” option, the Game-play Menu will display again. |

|  |  |
| --- | --- |
| **Flow of Events for the *Use Bonus Card* use-case** | |
| **Objective** | To award the player a random amount of money and security value. |
| **Pre-condition** | The player selected “Use Bonus Card” in the wildcards selection menu. The player has a Bonus Card. |
|  |  |
| **Main Flow** | 1. The system generate two random numbers according to pre-set rules, then display the numbers. |
| 2. The player gained the generated amount of money and security value. |
|  |  |
| **Post-condition** | The player gained a random amount of money and security value. Player will return to the game-play menu. |

|  |  |
| --- | --- |
| **Flow of Events for the *Use Damage Card* use-case** | |
| **Objective** | To allow the player to attack another player by reducing their money and security value by a random amount. |
| **Pre-condition** | The player selected “Use Damage Card” in the wildcards selection menu. The player has a Damage Card. |
|  |  |
| **Main Flow** | 1. The system list every other player, asking the player to select a target. |
| 2. The system generate two random numbers according to pre-set rules, then display the numbers. |
| 3. The targeted player loose the generated amount of money and security value. |
|  |  |
| **Alternative Flow** | At 3, the targeted player could lose the game if the randomly generated security value is more than the value they already have. |
|  |  |
| **Post-condition** | The target player loses money and security value. Player will return to the game-play menu. |

|  |  |
| --- | --- |
| **Flow of Events for the *Use Upgrade Card* use-case** | |
| **Objective** | To allow the player to build a server or mainframe for free. |
| **Pre-condition** | The player selected “Use Upgrade Card” in the wildcards selection menu. The player has an Upgrade Card. |
|  |  |
| **Main Flow** | 1. The system list every tile the player owns, asking the player to select. |
| 2. The server numbers on the tile is added by one. If there are already 4 servers, then it is replaced by a mainframe. |
|  |  |
| **Alternative Flow** | At 1. If the player owns no tile, the system displays a warning with nothing changed. The card is wasted |
| At 2. If there is already a mainframe on the tile, the system displays a warning with nothing changed. The card is wasted. |
|  |  |
| **Post-condition** | The player gained a server or mainframe for free. Player will return to the game-play menu. |

|  |  |
| --- | --- |
| **Flow of Events for the *Use Downgrade Card* use-case** | |
| **Objective** | To allow the player to attack another player by downgrading their server or mainframe. |
| **Pre-condition** | The player selected “Use Downgrade Card” in the wildcards selection menu The player has a Downgrade Card. |
|  |  |
| **Main Flow** | 1. The system list every other player, asking the player to select a target. |
| 2. If the target player has a mainframe, it is downgraded to four servers. Otherwise, the target player will lose one server. |
|  |  |
| **Post-condition** | The target player loses money and security value. Player will return to the game-play menu. |

|  |  |
| --- | --- |
| **Flow of Events for the *Use Move Card* use-case** | |
| **Objective** | To allow the player jump to any tiles. |
| **Pre-condition** | The player selected “Use Move Card” in the wildcards selection menu. The player has a Move Card. |
|  |  |
| **Main Flow** | 1. The system list every tile on the map, asking the player to select. |
| 2. The system display a menu of action the player can perform on the tile. The list is exactly the same as if the player as rolled a dice and landed on the tile. |
|  |  |
| **Post-condition** | The player is move to the selected tile. The player is ready for actions on the tile. |