|  |  |  |  |
| --- | --- | --- | --- |
| # | Use Cases | Requirement | Due |
| 10 | [Start single-player game](#h.1fob9te) | S10 | 12/2/2015 |
| 20 | [Play single-player game](#h.3znysh7) | S11 | 12/2/2015 |
| 30 | [Pause & Resume](#h.2et92p0) | M20 | 12/2/2015 |
| 40 | [Add User](#h.tyjcwt) | S20 | 12/2/2015 |
| 50 | [Edit User](#h.3dy6vkm) | S20 | 12/2/2015 |
| 60 | [Delete User](#h.1t3h5sf) | S20 | 12/2/2015 |
| 70 | [Add Game Levels](#h.4d34og8) | S42 | 12/2/2015 |
| 80 | [Add Assets](#h.2s8eyo1) | S41 | 12/2/2015 |
| 90 | [Register Account](#h.17dp8vu) | S10 | 12/2/2015 |
| 100 | [Log In](#h.3rdcrjn) | S10 | 12/2/2015 |
| 110 | [View High Scores](#h.26in1rg) | M10 | 12/2/2015 |
| 120 | [Start Multi-Player Game](#h.35nkun2) | S10 | 12/2/2015 |
| 130 | [Play Multi-Player Game](#h.1ksv4uv) | S11 | 12/2/2015 |

Use case #10: Start single-player game

|  |  |
| --- | --- |
| Use Case Name | Start single-player game session |
| Scope | Plumbers on ICE Game App |
| Level | User goal |
| Primary Actors | Player |
| Stakeholders and Interests | - Player: Wants to begin playing a level in single-player mode |
| Preconditions | User has installed and started the game application, has created an account, and signed in with that account within the game app. |
| Success Guarantee | The player has chosen the level he wants to play, the level loads successfully and gameplay begins. |
| Main Success Scenario | 1. System asks the player to choose single-player or multi-player mode.  2. Player chooses single-player mode.  3. System prompts the user to choose from the available levels, with an indication as to which ones are unlocked.  4. Player chooses one of the unlocked and available levels.  5. System initiates a single-player game session on the selected level and gameplay begins. |
| Extensions | 4a. Player chooses a level he has not unlocked yet.  i. The system ignores the player’s choice and allows him another chance to choose a level |
| Special Requirements |  |
| Technology and Data Variations List |  |
| Frequency of Occurrence | High |
| Miscellaneous |  |

Use case #20: Play single-player game

|  |  |
| --- | --- |
| Use Case Name | Play single-player game |
| Scope | Plumbers on ICE Game App |
| Level | User goal |
| Primary Actors | Player |
| Stakeholders and Interests | - Player: Has the goal of reaching the end of the current level in the shortest amount of time that he can, while collecting coins to increase his score. |
| Preconditions | User has initiated a single-player game session |
| Success Guarantee | The player is able to reach the end of the level, with a certain level of difficulty depending on the particular level, and his score is recorded if it exceeds his previous best score for the level. |
| Main Success Scenario | 1. The player's character starts at the predefined starting point for the level.  2. The player has two basic actions available to him: Walk forward, and jump.  i. **Walk forward**: The player's character proceeds horizontally, to the right. If this input is held down, this action continues until it is released.  ii. **Jump**: The player's character 'jumps' in the upward direction and is pulled back down due to gravity.  iii. These two actions can be **combined** by using Jump while holding Walk forward. The character will leap upward while maintaining its horizontal velocity – allowing for the player to leap between platforms and over obstacles.  3. The player navigates through the level by using appropriate actions while avoiding hazards and collecting coins.  4. The player succeeds in reaching the end of the level.  5. The system calculates the player's score, based on elapsed time and coins collected.  6. A message is displayed, along with the player's score. |
| Extensions | 3a. Player allows the character to come into contact with an 'enemy' or other hazard, or the character falls to the bottom edge of the play area.  i. The character 'dies' and is placed again at the current level's starting point. The main scenario begins again. |
| Special Requirements |  |
| Technology and Data Variations List |  |
| Frequency of Occurrence | High |
| Miscellaneous |  |

Use case #30: Pause & Resume

|  |  |
| --- | --- |
| Use Case Name | Pause & Resume |
| Scope | Plumbers on ICE Game App |
| Level | User goal |
| Primary Actors | Player |
| Stakeholders and Interests | Player: pause and resume game play |
| Preconditions | Game play has been started |
| Success Guarantee | Game play can be paused anytime and resumed anytime |
| Main Success Scenario | 1. The player will be able to pause the game play in any instance 2. The player will be able to resume the game play from the pause |
| Extensions |  |
| Special Requirements |  |
| Technology and Data Variations List |  |
| Frequency of Occurrence | Medium/Low |
| Miscellaneous |  |

Use case #40: Add User

|  |  |
| --- | --- |
| **Use Case Name** | Add User |
| **Scope** | User Administration System |
| **Level** | User goal |
| **Primary Actor** | Site Administrator |
| **Stakeholders and Interests** | User – | to be added to the system.  Player - |once added will have access to game(s).  Site Administrator -| responsible for the task of adding new users. |
| **Preconditions** | The administrator has already logged in. |
| **Success Guarantee** | The administrator has submitted the information and confirmed its accuracy. The information entered has been validated by the system. The database and server have connected successfully. |
| **Main Success Scenario** | 1. The administrator navigates to the User Management interface. 2. The system displays the list of available options. 3. The administrator selects the option to add a user. 4. The system prompts the administrator to enter information about the user. 5. The administrator enters the required information about the user. 6. The administrator submits the information to the system. 7. The system prompts the administrator to confirm the information entered. 8. The administrator confirms the information. 9. The system displays a message that the new user has been added successfully. |
| **Extensions** | 9a. (i) The system displays a message informing the administrator that there was an error processing the information submitted.  (ii) The system prompts the administrator to reenter erroneous information about the user.  9b. The system displays a message that there is no internet connection and the server cannot be reached.  9c. The system displays a message that there was an error and the database cannot be accessed. |
| **Special Requirements** | 1. There is a user management interface that is distinct from other administrative operations. 2. The list of options only allows the selection of one option for each user management operation (as a counter example, an administrator would not be able add a user and delete a user simultaneously with one submit operation). 3. If the server cannot be reached the system should timeout and display an application generated error. 4. The default font-size is visible to readers with less than 20/20 vision. |
| **Technology and Data Variations List** | 9bc. The configuration of the server engine will set the desired timeout and required handling operations on the event that it occurs. |
| **Frequency of Occurrence** | Infrequent. Either from the cycling of administrators from time to time or unhandled system error (which may be discontinued by a patch). |
| **Questions/Concerns** | * Multiple types of administrators? One for user mgmt. and one for game mgmt? |

Use case #50: Edit User

|  |  |
| --- | --- |
| **Use Case Name** | Edit User |
| **Scope** | User Management Subsystem |
| **Level** | User goal |
| **Primary Actor** | Site Administrator |
| **Stakeholders and Interests** | User – | to be added to the system.  Player - |once added will have access to game(s).  Site Administrator -| responsible for the task of editing user information. |
| **Preconditions** | The system has failed to allow the user to edit their information, so they have contacted the administrator for assistance OR a user has simply contacted the administrator to change information OR the user has forgot their password and needs to have it reset. The administrator has already logged in. |
| **Success Guarantee** | The information is valid after it has been changed. The updated information about the user has been submitted. The server and database have been successfully connected. |
| **Main Success Scenario** | 1. The administrator navigates to the User Management interface. 2. The system displays the list of available options. 3. The administrator selects the option to edit a user. 4. The system displays a list of all current users of the site. 5. The administrator selects a user. 6. The system prompts the administrator to edit the information of that user. 7. The administrator changes information about the user. 8. The administrator submits the information to the system. 9. The system prompts the administrator to confirm the information entered. 10. The administrator confirms the information. 11. The system displays a message that information about the user has been updated successfully. |
| **Extensions** | 11a. (i) The system displays a message informing the administrator that there was an error processing the information submitted.  (ii) The system prompts the administrator to reenter erroneous information about the user.  11b. The system displays a message that there is no internet connection and the server cannot be reached.  11c. The system displays a message that there was an error and the database cannot be accessed. |
| **Special Requirements** | 1. There is a user management interface that is distinct from other administrative operations. 2. The list of options only allows the selection of one option for each user management operation (as a counter example, an administrator would not be able add a user and delete a user simultaneously with one submit operation). 3. If the server cannot be reached the system should timeout and display an application generated error. 4. The default font-size is visible to readers with less than 20/20 vision. |
| **Technology and Data Variations List** | 11bc. The configuration of the server engine will set the desired timeout and required handling operations on the event that it occurs. |
| **Frequency of Occurrence** | Frequent. Until an automatic password reset via email becomes available, the administrator will be required to manually reset passwords for users that have forgotten their password. |
| **Questions/Concerns** | * Multiple types of administrators? One for user mgmt. and one for game mgmt? |

Use case #60: Delete User

|  |  |
| --- | --- |
| **Use Case Name** | Delete User |
| **Scope** | User Management Subsystem |
| **Level** | User goal |
| **Primary Actor** | Site Administrator |
| **Stakeholders and Interests** | User – | whose login information is being handled.  Player - |once deleted will no longer be able to access games or alternatively to administer the site depending on privilege level.  Site Administrator -| responsible for the task of deleting user information. |
| **Preconditions** | The administrator has already logged in. A user has been added to the system via user registration or the user management subsystem. The database and server connections are successful. |
| **Success Guarantee** | The delete user request has been submitted. The server and database have been successfully connected. |
| **Main Success Scenario** | 1. The administrator navigates to the User Management interface. 2. The system displays the list of available options. 3. The administrator selects the option to delete a user. 4. The system displays a list of all current users of the site. 5. The administrator selects a user. 6. The administrator submits the request to the system. 7. The system prompts the administrator to confirm the delete request. 8. The administrator confirms the delete request. 9. The system displays a message that the user has been successfully deleted. |
| **Extensions** | 9a. (i) The system displays a message informing the administrator that there was an error processing the information submitted.  (ii) The system prompts the administrator to reenter erroneous information about the user.  9b. The system displays a message that there is no internet connection and the server cannot be reached.  9c. The system displays a message that there was an error and the database cannot be accessed. |
| **Special Requirements** | 1. There is a user management interface that is distinct from other administrative operations. 2. The list of options only allows the selection of one option for each user management operation (as a counter example, an administrator would not be able add a user and delete a user simultaneously with one submit operation). 3. If the server cannot be reached the system should timeout and display an application generated error. 4. The default font-size is visible to readers with less than 20/20 vision. |
| **Technology and Data Variations List** | 9bc. The configuration of the server engine will set the desired timeout and required handling operations on the event that it occurs. |
| **Frequency of Occurrence** | Frequent. When a user account has been inactive for a time, that account will need to be deleted by the administrator, either that or at the request of a user or when an administrator retires. |
| **Questions/Concerns** | * Multiple types of administrators? One for user mgmt. and one for game mgmt? |

Use case #70: Add Game Levels

|  |  |
| --- | --- |
| Use Case Name | Add Game Levels |
| Scope | Server backend |
| Level | Administrator goal |
| Primary Actors | Administrator |
| Stakeholders and Interests | Administrator: update and/or upgrade game levels |
| Preconditions | Server has been set up and configured |
| Success Guarantee | The server stores the added game levels |
| Main Success Scenario | 1. The administrator logs into the server 2. The administrator selects the appropriate directory 3. The administrator uploads game levels to the directory |
| Extensions |  |
| Special Requirements |  |
| Technology and Data Variations List |  |
| Frequency of Occurrence | Low |
| Miscellaneous |  |

Use case #80: Add Assets

|  |  |
| --- | --- |
| Use Case Name | Add Assets |
| Scope | Server backend |
| Level | Administrator goal |
| Primary Actors | Administrator |
| Stakeholders and Interests | Administrator: update and/or upgrade assets |
| Preconditions | Server has been set up and configured |
| Success Guarantee | The server stores the added assets |
| Main Success Scenario | 1. The administrator logs into the server 2. The administrator selects the appropriate directory 3. The administrator uploads assets to the directory |
| Extensions |  |
| Special Requirements |  |
| Technology and Data Variations List |  |
| Frequency of Occurrence | Low |
| Miscellaneous |  |

Use case #90: Register Account

|  |  |
| --- | --- |
| Use Case Name | Register Account |
| Scope | Plumbers on Ice Game App |
| Level | User goal |
| Primary Actors | Player |
| Stakeholders and Interests | Player: wants to register account to play game |
| Preconditions | Player has already downloaded app and launched it |
| Success Guarantee | Player successfully registers an account for the app. |
| Main Success Scenario | 1. User chooses to register account 2. System prompts user with registration requirements 3. User inputs necessary information and submits data 4. System sends data to server and creates account 5. System prompts user with success account creation message |
| Extensions | 3a. User does not enter in available user name  i. System prompts username error message  ii. User enters a new user name  3b. User password does not meet the requirements  i. System prompts password error message  ii. User enters in a new password  4a. System cannot connect to webserver  i. System prompts user with connection error |
| Special Requirements |  |
| Technology and Data Variations List | Internet connection |
| Frequency of Occurrence | high |
| Miscellaneous |  |

Use case #100: Log In

|  |  |
| --- | --- |
| Use Case Name | Log in |
| Scope | Plumbers on ICE Game App |
| Level | user goal |
| Primary Actors | Player |
| Stakeholders and Interests | Player: wants to login successfully to play game. |
| Preconditions | User has already registered an account and launched the app |
| Success Guarantee | User enters username and password. System authenticates user’s session. |
| Main Success Scenario | 1. User inputs username and password 2. User hits submit 3. System sends input to web server for verification 4. System authenticates user’s session 5. Systems launches user’s profiled session |
| Extensions | 3a. System cannot connect to web server  i. System prompts error message associated with connection issues  4a. User input does not match with any account  i. System prompts user with authentication failed prompt  5a. System files are corrupted and cannot open  i. System terminates and closes application |
| Special Requirements | none |
| Technology and Data Variations List | internet connection |
| Frequency of Occurrence | high |
| Miscellaneous | none |
|  |  |
|  |  |

Use case #110: View High Scores

|  |  |
| --- | --- |
| Use Case Name | View High Scores |
| Scope | Plumbers on ICE Game App |
| Level | user goal |
| Primary Actors | Player |
| Stakeholders and Interests | Player: wants to view highscores to compare performance against other players |
| Preconditions | The app has already been installed |
| Success Guarantee | User launches high scores |
| Main Success Scenario | 1. User launches high scores 2. System launches highscore webpage |
| Extensions | 2a. System cannot connect to webserver  i. System prompts connection error prompt |
| Special Requirements |  |
| Technology and Data Variations List |  |
| Frequency of Occurrence | high |
| Miscellaneous |  |

Use case #120: Start multi-player game

|  |  |
| --- | --- |
| Use Case Name | Start multi-player game |
| Scope | Game app, Game server |
| Level | User goal |
| Primary Actors | Player |
| Stakeholders and Interests | - Player: Wants to begin a two-player match against an opponent over the internet |
| Preconditions | - User has installed and started the game application, has created an account, and is signed in with that account within the game app.  - The game application (client) has an internet connection and can connect to the server. |
| Success Guarantee | The player will be matched with another player also requesting to start a multi-player game, a level will be chosen by the system, and a two-player game session (match) will begin. |
| Main Success Scenario | 1. System asks the player to choose single-player or multi-player mode.  2. Player chooses multi-player mode.  3. The client creates a connection to the game server and requests to start a multi-player match.  4. The system performs a test to measure network latency between the client and the server. If this is within an acceptable range, the system continues to the next step.  5. (At least one other player with their own client has been following these same steps.)  6. The system chooses one of the other players also requesting to play a multi-player match and matches them with the first player.  7. The system chooses one of the levels available for multi-player at random for this match to be played on.  8. An appropriate message is displayed to each player, the chosen level appears and a countdown to the start begins.  9. When the countdown finishes, play begins. |
| Extensions | 4a. If the latency is found to be outside the range deemed acceptable for multi-player, the player is shown a message. The player may try again, but cannot proceed as long as the test fails.  6a. If there are no other players waiting for a match, the player is given the option to wait, or cancel. |
| Special Requirements |  |
| Technology and Data Variations List |  |
| Frequency of Occurrence | High |
| Miscellaneous |  |

Use case #130: Play multi-player game

|  |  |
| --- | --- |
| Use Case Name | Play multi-player game |
| Scope | Plumbers on ICE Game App |
| Level | User goal |
| Primary Actors | First player  Second player |
| Stakeholders and Interests | - Players: Have the goal of reaching the end of the level first, before the other player, their opponent, is able to do so. |
| Preconditions | Both game clients have a connection to the server, and a multi-player game session between the two players has been initiated |
| Success Guarantee | Both players have a chance to reach the end of the level first, and once one of them does the match ends and the system records the results of the match. |
| Main Success Scenario | 1. Each player's character starts at the predefined starting point for the level.  2. As in single-player mode, each player has two basic actions available to him: Walk forward, and jump.  i. **Walk forward**: The player's character proceeds horizontally, to the right. If this input is held down, this action continues until it is released.  ii. **Jump**: The player's character 'jumps' in the upward direction and is pulled back down due to gravity.  iii. These two actions can be **combined** by using Jump while holding Walk forward. The character will leap upward while maintaining its horizontal velocity – allowing for the player to leap between platforms and over obstacles.  3. Each player navigates through the level by using appropriate actions while avoiding hazards.  4. One player succeeds in reaching the end of the level (before the other).  5. The system records some details of the match, including the identities of the two players, the level on which it was played, the winner, the number of times each player 'died', and the elapsed time from the start until the winning player finished.  6. An appropriate message is displayed to each player and results of the match are shown. |
| Extensions | 3a. A player allows his character to come into contact with an 'enemy' or other hazard, or the character falls to the bottom edge of the play area.  i. The character 'dies' and is placed back at the level's starting point. |
| Special Requirements |  |
| Technology and Data Variations List |  |
| Frequency of Occurrence | High |
| Miscellaneous |  |