Blackbear-Consultants Use Case Models Document

Use Case Models

A close up of a map

Description automatically generated

A close up of text on a white background

Description automatically generated

A close up of a map

Description automatically generatedA close up of text on a white background

Description automatically generated

A close up of a map

Description automatically generatedA close up of text on a white background

Description automatically generated

Use Case Model Descriptions

|  |  |
| --- | --- |
| **Title:** | **View User Statistics** |
| **Description**: | Allows the user to view the gameplay statistics of the user that are logged into the system. |
| **System Under Design:** | Applications Statistics Menu |
| **Primary Actor**: | Instructor |
| **Participants**: | Parent |
| **Goal**: | The instructor will be able to see the data collected from the user for all of their tasks. |
| **Following Use Cases**: | Extends: Select Game Mode |
| **Precondition**: | The user must complete or fail a task for data to be stored.  They must also be logged into the user’s account. |
| **Success Postcondition**: | A user completed or failed a task, and the instructor was able to view if the user was successful in the task or not in the statistics menu for that client and given task. |

|  |  |
| --- | --- |
| **STEPS**:  1. The user selects the ‘View Statistics’ menu button.  3. The user selects the game they wish to view data for.  5. The user selects a task they wish to view the data for. | **SYSTEM RESPONSES**:  2. The system displays the ‘View Statistics' menu which displays the different games available in the application.  4. The system displays the tasks within that game.  6. The data collected by the application for the task is displayed by graphical image and raw values.  7. The Instructor or Parent can swipe through different menus to have the data collected in any format they wish. |

|  |  |
| --- | --- |
| **Title:** | **Display Task Progress** |
| **Description**: | The application will display the current progress for the task being played in the selection menu. |
| **System Under Design:** | Game Selection |
| **Primary Actor**: | N/A |
| **Participants**: | Parent/Instructor |
| **Goal**: | Show task progress through star rating. |
| **Following Use Cases**: | Extension of: Select Object |
| **Precondition**: | A task must have been started by the user.  The user is also logged into their account. |
| **Success Postcondition**: | The instructor will be able to have immediate information regarding the user’s progress on the task selected. |

|  |  |
| --- | --- |
| **STEPS**:  1. The user selects the task selection menu.  3. The user swipes between tasks that have either been completed, or not completed. | **SYSTEM RESPONSES**:  2. The system opens the task selection menu.  4. The Task Selection menu displays the current task, and progress of task by user along with a start button to start the task.   1. If no progress has been completed the system will display that the task has not been started or 2. The system will display collected data if the task was failed in order to indicate there was a failure in completing the task instead of displaying the successful completion of the task. |

|  |  |
| --- | --- |
| **Title:** | **Enter Password** |
| **Description**: | The system will log a user in if a correct password is entered, and deny them otherwise.. |
| **System Under Design:** | Account login |
| **Primary Actor**: | N/A |
| **Participants**: | Parent/Instructor |
| **Goal**: | They will be able to access the account by entering the password when prompted by the system. |
| **Following Use Cases**: | Included in: Click Username |
| **Precondition**: | The account must be created. |
| **Success Postcondition**: | The account will be accessed upon successfully entering the password. |

|  |  |
| --- | --- |
| **STEPS**:  1. The user clicks on a username.  3. User enters a password.  5. User either:   1. The user is logged into the account and sees the Game Selection Menu. 2. The user is rejected and sees a message that the account is locked. | **SYSTEM RESPONSES**:  2. System displays the username followed by a text field to enter a password.  4. System authenticates password:   1. If the password is entered incorrectly the application will prompt for a correct password or 2. Within three attempts of entering a wrong password the application will lock out the user. |

|  |  |
| --- | --- |
| **Title:** | **Select Game Task** |
| **Description**: | Users select a task in the game selection menu and press the “Start” button to start. |
| **System Under Design:** | Game Selection |
| **Primary Actor**: | N/A |
| **Participants**: | Parent/Instructor |
| **Goal**: | The actor will successfully complete the current game task. |
| **Following Use Cases**: | Extension of: Select Object |
| **Precondition**: | The user must be on the game selection menu.  There is a pre-selected task to play if start is initially hit. |
| **Success Postcondition**: | The task will be started and gameplay can commence. |

|  |  |
| --- | --- |
| **STEPS**:  1.   1. The user selects the dropdown menu to go between task categories or 2. The user uses arrows to go between tasks or 3. User hits the start button. | **SYSTEM RESPONSES**:  2.   1. The system displays the selected task category or 2. System will display the different selected task or 3. System starts the selected task. |

|  |  |
| --- | --- |
| **Title:** | **Suggest Starting Tasks** |
| **Description**: | Using data from the user, the system suggests a task the user can continue at login. |
| **System Under Design:** | Game Selection |
| **Primary Actor**: | N/A |
| **Participants**: | Parent/Instructor |
| **Goal**: | The system will suggest a task the user should work on. |
| **Following Use Cases**: | Extension of: Select Minigame |
| **Precondition**: | The user is logged in. |
| **Success Postcondition**: | The user has a suggested task to work on when they log in. |

|  |  |
| --- | --- |
| **STEPS**:  1. The user selects the game selection menu.  4a. The user will press “start” and begin the selected task.  4b. The user will select a different task in the select minigame menu. | **SYSTEM RESPONSES**:  2. The system opens the game selection menu.  3. The system displays a suggested task to work on. |

|  |  |
| --- | --- |
| **Title:** | **Give Prompt (Audio/Visual)** |
| **Description**: | The system will produce an audio or visual cue for the user to complete the given task. |
| **System Under Design:** | Gameplay |
| **Primary Actor**: | N/A |
| **Participants**: | Child |
| **Goal**: | The user will complete the task given the correct audio or visual cue. |
| **Following Use Cases**: | Extension Of: Click Object |
| **Precondition**: | User must be playing a minigame |
| **Success Postcondition**: | System displays/plays appropriate visual/audio cue |

|  |  |
| --- | --- |
| **STEPS**:  1. The user selects a game object, which is the wrong game object.  2a. User successfully follows prompt.  2b. User fails to follow prompt. | **SYSTEM RESPONSES**:  1. The system will prompt the user with a visual and audio cue when a wrong answer is given.  3a. Prompts the user with a positive audio/visual cue.  3b. Prompts the user to follow the prompt with additional help. |

|  |  |
| --- | --- |
| **Title:** | **Take Pictures** |
| **Description**: | The user will be able to take pictures of real world objects and put them into their game, to highlight certain aspects such as shape, color, etc. |
| **System Under Design:** | Game Selection |
| **Primary Actor**: | N/A |
| **Participants**: | Instructor/Parent |
| **Goal**: | The user(s) will be able to use the device’s camera to take a picture and use the image in the selected task. |
| **Following Use Cases**: | None |
| **Precondition**: | The user has to be on the game selection menu, and the System has permission to use the device’s camera. |
| **Success Postcondition**: | After the user(s) have selected the camera button, the system will be able to take a photo and retrieve it to use as an object in the task. |

|  |  |
| --- | --- |
| **STEPS**:  1. The user(s) will press the ‘Camera’ button.  3. The user takes a picture.  5. When the user(s) decides to use a photo from the device they will navigate through the device’s storage system and select the proper image. | **SYSTEM RESPONSES**:  2. The System will activate the camera so the user(s) can use the device’s camera to snap a photo of a real world object.  4. The System will store the photo in the device’s storage device.  6. The image will be imported to the Gameplay Screen to allow the user(s) to adjust the image to fit the task. |

|  |  |
| --- | --- |
| **Title:** | **Get User Permissions** |
| **Description**: | In order to access the device’s camera and storage system for photo and file use, the system will prompt the user to allow or deny access to the following: the device’s camera, and the device’s storage system, etc. |
| **System Under Design:** | Settings Display |
| **Primary Actor**: | N/A |
| **Participants**: | Instructor/Parent |
| **Goal**: | The system will be given access and permission to use the device’s camera, storage system, etc. |
| **Following Use Cases**: | N/A |
| **Precondition**: | The Application is on, and the user is logged in. |
| **Success Postcondition**: | The System will be given access and permission to the device’s camera, storage system, etc. |

|  |  |
| --- | --- |
| **STEPS**:  1. User presses the camera button.  3. The user:   1. Clicks allow, giving the app permission to use camera and file storage or 2. clicks deny, now allowing the app permission to use camera and file storage. | **SYSTEM RESPONSES**:  2. The system will prompt the user(s) to click “allow” to give the system permission to use the device's camera and file storage.  4. The system:   1. Updates permissions and lets users store photos and display settings menu or 2. display settings menu |

|  |  |
| --- | --- |
| **Title:** | **Click Object** |
| **Description**: | User clicks on a game object (shape, number, letter, etc.) to progress and get feedback. |
| **System Under Design:** | Gameplay |
| **Primary Actor**: | Child |
| **Participants**: | Child |
| **Goal**: | User will complete the task by clicking on the appropriate object. |
| **Following Use Cases**: | Extends: Track Progress  Extends: Give Prompt (Audio/Visual) |
| **Precondition**: | User is logged in.  Minigame has been selected. |
| **Success Postcondition**: | After user has selected correct object, system will positively prompt the user and move to next task |

|  |  |
| --- | --- |
| **STEPS**:  1. The user clicks an object. | **SYSTEM RESPONSES**:  2. The system processes clicked object:   1. If object selected was correct, sends that information to progress tracker and prompter and new minigame task is given or 2. Object selected was incorrect, sends information to the progress tracker and prompter or 3. Object selected was indicated by prompt and correct. That information is sent to the progress tracker and a new minigame task is given. 4. Object selected was indicated by prompt and incorrect. Information is sent to the progress tracker and a new minigame task is given. |

|  |  |
| --- | --- |
| **Title:** | **Track Progress** |
| **Description**: | The system will record/update the user's progress as the user works on tasks. |
| **System Under Design:** | Gameplay |
| **Primary Actor**: | N/A |
| **Participants**: | Child |
| **Goal**: | The user’s progress will be recorded and updated as they work on tasks. |
| **Following Use Cases**: | Extension of: Click Object |
| **Precondition**: | The user has clicked an object in a minigame. |
| **Success Postcondition**: | The system updated the user's progress on tasks with success or failure. |

|  |  |
| --- | --- |
| **STEPS**:  1. The click object functionality gives track progress data. | **SYSTEM RESPONSES**:  2. The System records the which object the user clicks on.   1. If the user clicked on the correct game object, the system updates the database with success. 2. If the user clicked on an incorrect object, the system updates the database with a failure. |

|  |  |
| --- | --- |
| **Title:** | **Select Game Mode** |
| **Description**: | Allows the user to select which task’s progress to view. |
| **System Under Design:** | Statistics Display |
| **Primary Actor**: | N/A |
| **Participants**: | Parent/Instructor |
| **Goal**: | System will display the user's progress for the selected task. |
| **Following Use Cases**: | Extension of: View User Statistics |
| **Precondition**: | User is logged in and selects the Statistics Menu. |
| **Success Postcondition**: | System will display the user’s progress for the selected task. |

|  |  |
| --- | --- |
| **STEPS**:  1. The user selects a list of tasks button  3. The User selects which tasks they wish to view statistics for. | **SYSTEM RESPONSES**:  2. The System displays the list of tasks.  4. The System will display the statistics for the selected task.   1. If the task has not been played, a no data sign will be displayed over statistics graphs and charts or 2. The system will display successes and failures for each task selected. |

|  |  |
| --- | --- |
| **Title:** | **Select Minigame** |
| **Description**: | Allows the user to decide which task to work on. |
| **System Under Design:** | Game Selection |
| **Primary Actor**: | N/A |
| **Participants**: | Parent/Instructor |
| **Goal**: | The user is able to pick which task and begin working on it. |
| **Following Use Cases**: | Extends: Suggest Starting Task  Extends: Display Task Progress  Extends: Select Game Task |
| **Precondition**: | The user is logged in and on the Game Selection Menu.  The user has already selected a game task. |
| **Success Postcondition**: | The system will start the selected task for the user to complete. |

|  |  |
| --- | --- |
| **STEPS**:  1. The user swipes through different minigames that are displayed using arrows:   1. The user hits the right arrow or 2. The user hits the left arrow.   3. The user hits the submit button with the minigame they want displayed | **SYSTEM RESPONSES**:  2. The system:   1. Displays the game to the left or 2. Displays the game to the right.   4. The system starts the minigame selected by the user. |

|  |  |
| --- | --- |
| **Title:** | **Adjust Audio** |
| **Description**: | The user will be able to mute or unmute the sound |
| **System Under Design:** | Settings Display |
| **Primary Actor**: | N/A |
| **Participants**: | Instructor/Parent |
| **Goal**: | The system will increase or decrease the current level of audio and brightness based on the user’s preferences. |
| **Following Use Cases**: | N/A |
| **Precondition**: | The user is logged into the application and on Settings Display Menu. |
| **Success Postcondition**: | The user mutes or unmutes the sound the app makes |

|  |  |
| --- | --- |
| **STEPS**:  1. The user taps the checkbox. | **SYSTEM RESPONSES**:  2. The system:   1. Displays a check mark in the checkbox if the sound is muted 2. Displays just an empty checkbox if the sound is unmuted.   3. The system:   1. Mutes the sound the application makes 2. Unmutes the sound the application makes. |

|  |  |
| --- | --- |
| **Title:** | **Log Out User** |
| **Description**: | Allows the user to log out of their current session. |
| **System Under Design:** | Settings Display |
| **Primary Actor**: | N/A |
| **Participants**: | Parent/Instructor |
| **Goal**: | The user is logged out of the currently logged in account. |
| **Following Use Cases**: | N/A |
| **Precondition**: | The user is logged into an account and on the Settings Display menu. |
| **Success Postcondition**: | The user is logged out of the current account and returned to the Account Login Menu. |

|  |  |
| --- | --- |
| **STEPS**:  1. The user presses the “Logout” button. | **SYSTEM RESPONSES**:  2. The system saves all user data, clears current session information, and brings the application to the log in landing menu. |

|  |  |
| --- | --- |
| **Title:** | **Add User** |
| **Description**: | Allows the creation of an account for a new user. |
| **System Under Design:** | Account Registration |
| **Primary Actor**: | Instructor |
| **Participants**: | Instructor |
| **Goal**: | The user will have an account created with space to store progress on tasks, and system preferences. |
| **Following Use Cases**: | Extension of: Edit User |
| **Precondition**: | There are no users logged in.  Users are on the log in landing page  The user is in the edit user page |
| **Success Postcondition**: | A new account is created for the new user. |

|  |  |
| --- | --- |
| **STEPS**:  1. The User selects the add button  2. The user enters a username and password. | **SYSTEM RESPONSES**:  2. The system displays text fields for username and password.  3.  a. If the username is taken, the registration is denied and a “username taken” message is displayed or  b. If the username and password aren’t taken, the user is added to the registry. |

|  |  |
| --- | --- |
| **Title:** | **Delete User** |
| **Description**: | Deletes the progress of a user and their registry information |
| **System Under Design:** | Account Registration |
| **Primary Actor**: | Instructor |
| **Participants**: | Instructor |
| **Goal**: | The selected user’s name, password, progress, and settings are removed from application. |
| **Following Use Cases**: | Extension: Edit User |
| **Precondition**: | There are no users logged in.  Users are on the log in landing page  The user is in the edit user page |
| **Success Postcondition**: | Selected user’s account is removed from application. |

|  |  |
| --- | --- |
| **STEPS**:  1. The user clicks the delete icon.  3. The user clicks the minus button and deletes a user. | **SYSTEM RESPONSES**:    2. The system displays a minus button next to every user in the list of users.  4. The system erases user information from the registry and displays the edit screen without any minus signs. |

|  |  |
| --- | --- |
| **Title:** | **Edit User** |
| **Description**: | Allows the user with proper permission to edit another user’s account information such as, their name, password, settings, etc. |
| **System Under Design:** | Account Registration |
| **Primary Actor**: | Instructor |
| **Participants**: | Instructor |
| **Goal**: | The user is able to modify selected user’s name, password, settings, etc. |
| **Following Use Cases**: | Extends: Add User  Extends: Delete User |
| **Precondition**: | User must not be logged in  User must be on the users landing page. |
| **Success Postcondition**: | User is able to edit their account information |

|  |  |
| --- | --- |
| **STEPS**:  1. User hits the edit button.  3. User taps on the name of the user they want to edit.  5. The user edits some of the selected user’s account information and hits the save button. | **SYSTEM RESPONSES**:    2. The system displays the edit menu, full of registered users.  4. The selected user’s information is displayed in text fields, where the user can edit information of that user.  6. The system edits that user information and saves it in the user registry. |

|  |  |
| --- | --- |
| **Title:** | **Click Username** |
| **Description**: | Initial Menu for Application to allow the user to select which account to begin using. |
| **System Under Design:** | Account Login |
| **Primary Actor**: | N/A |
| **Participants**: | Parent/Instructor |
| **Goal**: | The user is able to select a username to log into the application and begin tasks. |
| **Following Use Cases**: | Includes: Enter Password |
| **Precondition**: | Application is on |
| **Success Postcondition**: | The user is able to select the desired username and the system displays the password box and login button. |

|  |  |
| --- | --- |
| **STEPS**:  1. The User selects a username to log into the associated user account. | **SYSTEM RESPONSES**:  2. The System displays the Username selected with a text box to enter the password for the user account. |