1: Register for website

|  |  |
| --- | --- |
| Use Case Name | Register for website |
| Participating Actors | Not logged in user |
| Goal | Obtain a new personal account for this particular user. |
| Trigger | User chooses to register for a new account from the login page. |
| Precondition | User does not have an account and desires a new one. |
| Postcondition | User's information for a new account is sent to an administrator for approval. |

## Basic Flow:

|  |  |
| --- | --- |
| 1 | Website prompts for the user’s first and last name, their email, and a password. |
| 2 | User enters their information |
| 3 | User submits their information. |
| 4 | System creates a new user with the given information. |
| 5 | System sends the information to an administrator for approval. |

## Exceptions:

|  |  |
| --- | --- |
| 2 | Email is already in use |
| 2.1 | System displays a notification to the user informing them that the email is already taken. |
| 2.2 | System returns to step 1. |

## Tests:

|  |  |
| --- | --- |
| 1 | Upon arriving at the login screen, confirm that the prompt is displayed. |
| 2 | Test that the user can input text in (1) first name field (2) last name field (3) email field (4) password field. |
| 3 | Test that the user can successfully submit their information to the System. |
| 4 | [u] Test that the database (1) makes a new user (2) that the new user has the information that the User submitted (3) that this account CAN be accessed with the email and password tied to it. |
| 5 | Test that the information is sent to an administrator’s notifications. |

2: Log into website

|  |  |
| --- | --- |
| Use Case Name | Log into website |
| Participating Actors | Not logged in user |
| Goal | Obtain logged in status on website with a registered account. |
| Trigger | User desires to be logged into the website. |
| Precondition | User is not logged into the website. |
| Postcondition | User is logged into the website. |

## Flow:

|  |  |
| --- | --- |
| 1 | System Prompts for email and password entry. |
| 2 | User supplies the email and password. |
| 3 | User confirms submission. |
| 4 | System logs in the User with the credentials provided. |

## Exceptions:

|  |  |
| --- | --- |
| 3 | Incorrect email and password combination. |
| 3.1 | System displays a notification to the user stating that they entered incorrect email or password. |
| 3.2 | System returns to step 1. |

## Tests:

|  |  |
| --- | --- |
| 1 | Test that the System displays the prompt for email and password entry. |
| 2 | Test that the User can input their email and password. |
| 3 | Test that the User can submit their entered email and password to the System. |
| 4 | [u] Test that the System successfully logs in the user with the supplied credentials. |

3: Ability to Log Out

|  |  |
| --- | --- |
| Use Case Name | Ability to Log Out |
| Participating Actors | Logged in Users |
| Goal | User is logged out of the System. |
| Trigger | User desires to log out of the System. |
| Precondition | User is logged into the System. |
| Postcondition | User is logged out of the System. |

## Flow:

|  |  |
| --- | --- |
| 1 | User hits the log out button. |
| 2 | System logs the User out. |
| 3 | System returns User to the default entry page of the website. |

## Exception Template:

|  |  |
| --- | --- |
| 3 | User is not connected to the internet. |
| 3.1 | System displays a notification that the User is not connected to the internet. |
| 3.2 | System returns to the original page where the user clicked the link (No change). |

## Tests:

|  |  |
| --- | --- |
| 1 | Test that Users are logged into the System. |
| 2 | Test that the System displays a method to the User to log out of the System. |
| 3 | Test that the method displayed to log out of the System is functional. |
| 4 | Test that the method, once pressed, will inform the server to log the User out of the system. |
| 5 | Test that the User is properly logged out (such as reloading the page won’t log them in). |
| 6 | Test that upon being logged out that the User is brought to the main entry page of the System. |
| 7 | Test that if the User is not connected to the internet that they are notified of it by the System. |
| 8 | Test that if the User is not connected to the internet that they remain on the same page. |

4: Visualization

|  |  |
| --- | --- |
| Use Case Name | Visualization |
| Participating Actors | Users |
| Goal | Display a fantastic visual representation of the database data. |
| Trigger | User desires to see a visual representation of the website’s data. |
| Precondition | The user doesn’t see a visual representation of the data. |
| Postcondition | The user now sees a visual representation of the data. |

## Flow:

|  |  |
| --- | --- |
| 1 | User clicks to the visualization page. |
| 2 | System displays the visualization page and all search options. |
| 3 | User has the information displayed for a particular node of the graph where visualizations like graph are represented by the node-vertex relationship, contributed relationships, etc. are presented on the page for the node the user is currently on. |

## Exception:

|  |  |
| --- | --- |
| 2 | User is not connected to the internet. |
| 2.1 | System displays a notification that the User is not connected to the internet. |
| 2.2 | System returns to the original page where the user clicked the link (No change). |

## Tests:

|  |  |
| --- | --- |
| 1 | Test to see that there is an option to go to Visualize the data. |
| 2 | Test if the method can be interacted with by the User. |
| 3 | Test if the User is taken to the Visualization page. |
| 4 | Test if the Visualizations and search options are displayed. |

5: Search

|  |  |
| --- | --- |
| Use Case Name | Search |
| Participating Actors | Users |
| Goal | To search for a particular person on the website and obtain the results. |
| Trigger | User wants to find someone through the search page. |
| Precondition | User does not see the person through the search. |
| Postcondition | User is presented with the data of the search. |

## Flow:

|  |  |
| --- | --- |
| 1 | User navigates to the home page where the search option is in. |
| 2 | User selects the search field that they desire to search in. |
| 3 | User types in what they want to search for into the box and initiates the search. |
| 4 | The System takes in the supplied entry and then searches the database for related results. |
| 5 | The System presents the results from the query to the User. |

## Exceptions:

|  |  |
| --- | --- |
| 2 | User selects wrong search field |
| 2.1 | The search will run as normally, but is very unlikely to return the desired results. |
| 2.2 | Do not flag the user, continue as if nothing was wrong. |
| 4 | The field desired to be searched for does not exist. |
| 4.1 | Proceed to the same final search page (step 5) however (refer to 4.2): |
| 4.2 | System displays a notification displaying to the User their search query had no results. |

## Tests:

|  |  |
| --- | --- |
| 1 | Test that the User has a method to navigate to the page. |
| 2 | Test that the User can select search fields. |
| 3 | Test that the User can type input data into the search tool. |
| 4 | [u] Test that the User can begin a search with the query they chose. |
| 5 | [u] Test that the System searches the database for the related results. |
| 6 | [u] Extensively test that the database returns accurate results to the User’s queries. |
| 7 | [u] Test that the System presents the accurate results to the User. |
| 8 | [u] Test that if the query has no results, the User is informed of it. |
| 9 | Test that incorrect field and query information can still work and proceeds as if normal. |

6: Detail Information

|  |  |
| --- | --- |
| Use Case Name | Detail Information |
| Participating Actors | Users |
| Goal | To obtain information of a particular person. |
| Trigger | User desires to learn more about a particular person. |
| Precondition | User does not know information about the other person. |
| Postcondition | Information of the person is displayed to the User. |

## Flow:

|  |  |
| --- | --- |
| 1 | User navigates through the search results and chooses to view more information about a person. |
| 2 | The System presents the User with the person’s information. |

## Exception:

|  |  |
| --- | --- |
| 2 | User is not connected to the internet. |
| 2.1 | System displays a notification that the User is not connected to the internet. |
| 2.2 | System returns to the original page where the user clicked the link (No change). |

## Tests:

|  |  |
| --- | --- |
| 1 | Test that there is a method for Users to select to view the information of a person. |
| 2 | Test that the method is something a User can successfully interact with. (Is it functional?) |
| 3 | Test that upon utilizing the method, that the User is successfully shown the information. |
| 4 | [u] Test that the User is viewing the correct information. |
| 5 | Test that if the User is not connected to the internet that they are notified of it. |
| 6 | Test that if the User is not connected to the internet then they remain on the same page (No 404, etc.) |

7: Submit information

|  |  |
| --- | --- |
| Use Case Name | Submit Information |
| Participating Actors | Logged in user |
| Goal | User submits new information. |
| Trigger | User would like to submit information. |
| Precondition | User knows what information they would like to submit. |
| Postcondition | Submitted information is sent to an administrator for approval. |

## Flow:

|  |  |
| --- | --- |
| 1 | User chooses the option to submit information. |
| 2 | System displays the page for submitting new information about a person. |
| 3 | User enters the new information. |
| 4 | User confirms the new information they would like to add. |
| 5 | System creates a new save object of the edited data and stores it for auditing. |
| 6 | System sends the new information to an administrator for approval. |

## Exceptions:

|  |  |
| --- | --- |
| 5 | User loses internet connection when attempting to confirm their submission. |
| 5.1 | System displays notification to the user stating that they have lost internet connection. |
| 5.2 | System will remain on the same page with the text stored in it until the user resubmits. |
| 5.3 | User may alternatively leave the page and then the System will discard their information. |
| 6 | No database storage space left. |
| 6.1 | System will notify administrator of the issue. |
| 6.2 | System will display notification of the error to the user, stating inability to make the submission. |

## Tests:

|  |  |
| --- | --- |
| 1 | Test that the submit button is present on the page. |
| 2 | Test that the submit button is functional. |
| 3 | Test that the page presents to the User the submit information page. |
| 4 | Test that the User can successfully confirm the way they wish to submit the page’s information. |
| 5 | Test that the User has a method to input the data. |
| 6 | [u] Test that the User has a method to confirm the information, and that this commits them. |
| 7 | [u] Test that when the User confirms their commit that the System accepts it. |
| 8 | [u] Test that the System creates a new save object of data. |
| 9 | [u] Test that the System successfully maintains this new save object of data in the database. |
| 10 | Test that the information is sent to an administrator’s notifications. |

8: Edit information

|  |  |
| --- | --- |
| Use Case Name | Edit Information |
| Participating Actors | Logged in user |
| Goal | User chooses to edit information. |
| Trigger | User desires to edit information. |
| Precondition | User knows what information they would like to edit. |
| Postcondition | Information is changed in the system. |

## Flow:

|  |  |
| --- | --- |
| 1 | User locates piece of data that they wish to modify on the page. |
| 2 | User clicks the edit button. |
| 3 | System displays methods to edit the data on the page. |
| 4 | User decides what edits that they wish to make to the page and types them in. |
| 5 | User confirms that they want these changes made to the page. |
| 6 | System creates a new save object of the edited data and stores it for audits. |
| 7 | System displays the newly edited information to the user. |

## Exceptions:

|  |  |
| --- | --- |
| 4 | User loses internet connection when attempting to confirm their edits. |
| 4.1 | System displays notification to the user stating that they have lost internet connection. |
| 4.2 | System will remain on the same edit page with the text stored in it until the user resubmits |
| 4.3 | User may alternatively leave the page and then the System will discard their typed in edits. |
| 5 | No database storage space left. |
| 5.1 | System will notify administrator of the issue. |
| 5.2 | System will display notification of the error to the user, stating inability to make edit. |
| 5.3 | No edit will be done to the data on the page, as if nothing ever happened. |

## Tests:

|  |  |
| --- | --- |
| 1 | Test that data is present on the page, as is the edit button. |
| 2 | Test that the edit button is functional. |
| 3 | Test that the page presents to the User the ways to edit the data on the page. |
| 4 | Test that the User can successfully confirm the way they wish to edit the page’s information. |
| 5 | Test that the User has a method to input the data modifications |
| 6 | [u] Test that the User has a method to confirm their modifications, and that this commits them. |
| 7 | [u] Test that when the User confirms their commit that the System accepts the modifications. |
| 8 | [u] Test that the System accepts the changes that the System creates a new save object of data. |
| 9 | [u] Test that the System successfully maintains this new save object of data in the database. |

9: View Audit Trail of Edits

|  |  |
| --- | --- |
| Use Case Name | View Audit Trail of Edits |
| Participating Actors | Administrator |
| Goal | Administrator can see the edits made to a person’s information. |
| Trigger | Administrator would like to view edits made to a person’s page. |
| Precondition | The person’s information exists in the system. |
| Postcondition | System displays a log of edits to the Administrator. |

## Flow:

|  |  |
| --- | --- |
| 1 | Administrator navigates to the information of a particular person. |
| 2 | Administrator chooses to view the log of edits made to the information. |
| 3 | System displays the log of edits for the information. |

## Exceptions:

|  |  |
| --- | --- |
| 3 | Administrator is not connected to the internet. |
| 3.1 | System displays a notification that the Administrator is not connected to the internet. |
| 3.2 | System returns to the original page where the Administrator clicked the link (No change). |

## Tests:

|  |  |
| --- | --- |
| 1 | Test that there is a method for Administrators to select to view the log of information about a person. |
| 2 | Test that the method is something the Administrator can successfully interact with. |
| 3 | Test that the Administrator is shown the correct information. |
| 4 | Test that if the Administrator is not connected to the internet, that they are notified of it. |
| 5 | Test that if the Administrator is not connected to the internet, then they remain on the same page (No 404, etc.) |

10: Notifications

|  |  |
| --- | --- |
| Use Case Name | Notifications |
| Participating Actors | Administrator |
| Goal | Administrator views notifications for new user and data information. |
| Trigger | Administrator would like to view their notifications. |
| Precondition | A new user or data entry is submitted to the system. |
| Postcondition | The System displays the Administrator's new notifications. |

## Flow:

|  |  |
| --- | --- |
| 1 | Administrator navigates to their notifications. |
| 2 | System displays the Administrator’s list of notifications that require approval. |

## Exceptions:

|  |  |
| --- | --- |
| 2 | Administrator is not currently online (no internet access or otherwise). |
| 2.1 | System displays a notification that the Administrator is not connected to the internet. |
| 2.2 | System returns to the original page where the Administrator clicked the link (No change). |
| 2.3 | System saves messages in the administrator inbox to display when they come online. |
| 2.4 | Until administrator approval, all modifications made to the new data are not committed. |

## Tests:

|  |  |
| --- | --- |
| 1 | [u] Test that the System recognizes when a new user or data information is added. |
| 2 | [u] Test that when the new information is added, the System notifies all Administrators. |
| 3 | [u] Test that all Administrator accounts receive a notification from the System. |
| 4 | Test that the Administrators can view these messages. |
| 5 | [u] Test that when the Administrators aren’t online, that messages are still received. |

11: Information Verification

|  |  |
| --- | --- |
| Use Case Name | Information Verification |
| Participating Actors | Administrator |
| Goal | Administrator approves new information coming into the System. |
| Trigger | Administrator would like to approve or disapprove new information while checking their notifications. |
| Precondition | A new user or data entry is submitted to the system and the Administrators are notified. |
| Postcondition | The System is modified depending on the Administrator's decision. |

## Flow:

|  |  |
| --- | --- |
| 1 | Administrator chooses the information they would like to approve or disapprove. |
| 2 | If the Administrator decides to approve the modification. |
| 2.1 | Administrator confirms the data modification. |
| 2.2 | The System makes the changes to the database and the webpage. If the change is a new user, then the user is notified that their account has been approved. |
| 3 | If the Administrator decides to disapprove the modification. |
| 3.1 | Administrator denies the data modification. |
| 3.2 | The System drops any possible modifications that could have been made to the System. |

## Exceptions:

|  |  |
| --- | --- |
| 2 | There is an issue with the database being too full or some other error in the commit. |
| 2.1 | The system displays a notification to the Administrator stating the database problem. |
| 2.2 | No changes are committed to the database. |

## Tests:

|  |  |
| --- | --- |
| 1 | Test that the Administrators have a functional method to confirm or deny changes. |
| 2 | [u] Test that the confirmation or denial of changes is committed by the System. |
| 3 | [u] Test that the modifications if denied are dropped completely by the System. |
| 4 | [u] Test that if the modifications are accepted the database of the System is properly changed. |
| 5 | Test that once a decision is made that the Administrators have the messages marked as read. |
| 6 | [u] Test that changes are NOT committed to the database without Administrator permission. |