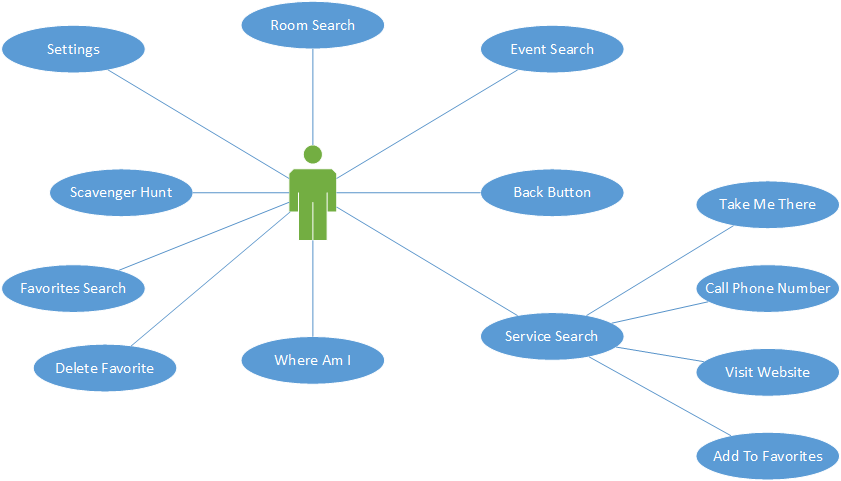
****

**Use Cases:**

**Title: Add to Favorites**

**Description:**  Allows the User to create a link to the Service Data of a Service they plan to utilize often.

**Pre-conditions:**

* The App is open and has focus.
* The App is displaying the Service Results Screen with the Service Data of a Service.

**Post-conditions:**

* The App displays the Favorites Screen showing a the User’s favorite Services including the newly added Service.

**Actors:**

* User
* App

**Main Success Scenario:**

1. The User presses the Add to Favorites Button.
2. The App displays a Prompt that asks the User if they would like to add the displayed Service to their favorites.
3. The User presses on the Prompt's yes button.
4. The App sets the Favorites Flag to true for the Service Data in the Service Database.
5. The App prepares the Favorites Screen.
6. The App queries all Services from the Service Database with a Favorites Flag set to true.
7. The App displays the Favorites Screen showing all the User’s Favorite Service List Items.

**Alternate Flow 1:**

3A. The User presses on the Prompt's no button.

4A. The App displays the Service Results Screen.

**Alternate Flow 2:**

1B. The App receives an interrupt signal from the operating system and loses focus.

2B. The App regains focus and resumes at the step where it had lost focus.

**Title: Call Phone Number**

**Description:** Allows the User to easily call a Service's contact phone number.

**Pre-conditions:**

* The App is open and has focus.
* The App is displaying the Service Results Screen.

**Post-conditions:**

* The App opens the Operating System's default phone dialing application and dials the number that the User clicked on.

**Actors:**

* User
* App

**Main Success Scenario:**

1. The User presses on the Service's phone number.
2. The App displays a Prompt that asks the User if they would like to call the phone number.
3. The User presses on the Prompt's yes button.
4. The App requests that the Operating System open its default phone dialing application and call the Service's phone number.
5. The phone dialing application launches and takes focus.

**Alternate Flow 1:**

3A. The User presses on the Prompt's no button.

4A. The App displays the Service Results Screen.

**Alternate Flow 2:**

1B. The App receives an interrupt signal from the operating system and loses focus.

2B. The App regains focus and resumes at the step where it had lost focus.

**Title: Delete Favorite**

**Description:** Allows the User to navigate to the Favorites Screen and remove a Favorite Service.

**Pre-conditions:**

* The App is open and has focus.
* The App is displaying the Favorites Screen.

**Post-conditions:**

* The App displays the Favorites Screen containing the updated Favorites List Items.

**Actors:**

* User
* App

**Main Success Scenario:**

1. The User swipes left over a Service List Item on the Favorites Screen.
2. The App moves the Service List Item left and displays a Delete Button to the right of the Service List Item.
3. The User presses the Delete Button.
4. The App finds the Service Data from the Service Database for the selected Service.
5. The App sets the Favorites Flag to false.
6. The App queries all Services from the Service Database with a Favorites Flag set to true.
7. The App displays the Favorites Screen showing all the User’s Favorited List Items.

**Alternate Flow 1:**

6A. The User swipes right on the Service List Item that has been swiped left.

7A. The App removes the Delete Button and moves the Service List Item to its original location on the Favorites Screen.

**Alternate Flow 2:**

1A. The App receives an interrupt signal from the operating system and loses focus.

2A. The App regains focus and resumes at the step where it had lost focus.

**Title: Event Search**

**Description:** Allows the User to search a list of available Events at Penn State Harrisburg in the selected month.

**Pre-conditions:**

* The User has successfully installed the App.
* The App is open and displaying the Home Screen.

**Post-conditions:**

* The App displays the Event Results Screen for the User.

**Actors:**

* User
* App

**Main Success Scenario:**

1. The User presses the Events Home List Item.
2. The App displays the Event Search Screen.
3. The User select a month with the provided Dropdown Menu.
4. The App queries the Event Database for a list of Event names for the selected month.
5. The App displays the list of Events.
6. The User presses an Event List Item on the Event Search Screen.
7. The App queries the Event Database for the Event Data of the selected Event.
8. The App displays the Event Results Screen with the Event Data that was found.

**Alternate Flow 1**:

5A. The App didn't find any Events the selected month from the Event Database.

6A. The App displays an Error Message indicating that no Events were found for that month.

7A. The User presses the OK Button on the Error message.

8A. The App displays the Event Search Screen.

**Alternate Flow 2**:

1B. The App receives an interrupt signal from the operating system and loses focus.

2B. The App regains focus and resumes at the step where it had lost focus.

**Title: Favorites Search**

**Description:** Allows the User to navigate to the Favorites Screen and select Favorite Services.

**Pre-conditions:**

* The App is open and has focus.

**Post-conditions:**

* The App displays the Service Results Screen containing Service Data for the Service selected by the User.

**Actors:**

* User
* App

**Main Success Scenario:**

1. The User presses the Favorites Menu Button.
2. The App prepares the Favorites Screen.
3. The App queries all Services from the Service Database with a Favorites Flag set to true.
4. The App displays the Favorites Screen with the User’s Favorite Service List Items.
5. The User presses a Favorites List Item on the Favorites Screen.
6. The App finds the Service Data from the Service Database for the selected Service List Item.
7. The App displays the Service Results Screen with the Service Data that was found.

**Alternate Flow 1:**

1A. The App receives an interrupt signal from the operating system and loses focus.

2A. The App regains focus and resumes at the step where it had lost focus.

**Title: Room Search**

**Description:** Allows the User to search for information about a specific Room on the PSU Harrisburg campus.

**Pre-conditions:**

* The App is open and has focus.

**Post-conditions:**

* The App displays the Room Results Screen for the User.

**Actors:**

* User
* App

**Main Success Scenario:**

1. The User presses the Room Menu Button
2. The App displays the Room Search Screen.
3. The User enters a Room Number with the provided Text Box and Building Name with the provided Dropdown Menu.
4. The User presses the Search Button.
5. The App finds the Room Data from the Room Database.
6. The App formats the found Room Data.
7. The App displays the Room Results Screen with the formatted Room Data.

**Alternate Flow 1**:

3A. The App cannot find the Room Data from the Room Database.

4A. The App displays an Error Message indicating that the room wasn't found.

5A. The User presses the ok button on the Error message.

6A. The App displays the Room Search Screen.

**Alternate Flow 2**:

1B. The App receives an interrupt signal from the operating system and loses focus.

2B. The App regains focus and resumes at the step where it had lost focus.

**Title: Scavenger Hunt**

**Description:** Enables the User to begin self-guided tour throughout the PSU Harrisburg campus learning about all the key information that is needed to be a successful member of the Penn State society.

**Pre-conditions:**

* The App is open and has focus.

**Post-conditions:**

* The App finds a Scavenger Hunt Clue that is not marked as solved and makes it the Current Clue.
* The App displays the Scavenger Hunt Screen with the Current Clue's Clue Description and, if it exists, the Current Clue's Clue Picture.

**Actors:**

* User
* App

**Main Success Scenario:**

1. The User presses the Scavenger Hunt Menu Button.
2. The App prepares the Scavenger Hunt Screen.
3. The App finds the first unsolved Scavenger Hunt Clue.
4. The App displays the Scavenger Hunt Screen.
5. The User enters their answer into the Text Box, if it was provided by the Screen.
6. The User presses the Scavenger Hunt Check Button.
7. The App finds that the Current Clue's Clue Solution requires GPS Coordinates.
8. The App requests the Device's GPS Status from the Operating System and determines that the GPS is enabled and the Device has a GPS signal.
9. The App gathers the user's current GPS Coordinates.
10. The App determines that current GPS Coordinates are within a 16 meter radius of the Current Clue's Clue Solution's GPS Coordinates.
11. The App marks Current Clue as solved.
12. The App finds the Scavenger Hunt Clue that follows the Current Clue.
13. The App sets the Current Clue to the found Scavenger Hunt Clue.
14. The App displays the Scavenger Hunt Screen with the Current Clue's Clue Description and if it exists, the Current Clue's Clue Picture.

**Alternate Flow 1:**

7A. The App finds that the Current Clue's Clue Solution requires a String.

8A. The App determines that the String in the Text Box matches (case-insensitive) the Current Clue's Clue Solution's String.

9A. The App resumes Main Success Scenario at step 11.

**Alternate Flow 2:**

7B. The App finds that the Current Clue's Clue Solution requires a String.

8B. The App determines that the String in the Text Box does not match (case-insensitive) the Current Clue's Clue Solution's String.

9B. The App displays an Error Message indicating the User's input was incorrect.

10B. The User presses the ok button on the Error Message.

11B. The App resumes Main Success Scenario at step 5.

**Alternate Flow 3**:

10C. The App determines that current GPS Coordinates are not within a 16 meters of the Current Clue's Clue Solution's GPS Coordinates.

11C. The App displays an Error Message indicating that the User is in the wrong location.

12C. The User presses the ok button on the Error Message.

13C. The App resumes Main Success Scenario at step 5.

**Alternate Flow 4**:

12D. The App cannot find the Scavenger Hunt Clue that follows the Current Clue.

13D. The App displays an Error Message indicating that the User has completed the Scavenger Hunt.

14D. The User presses the ok button on the Error Message.

15D. The App resets the Scavenger Hunt.

16D. The App sets the Current Clue to the first Scavenger Hunt Clue.

17D. The App resumes Main Success Scenario at step 14.

**Alternate Flow 5**:

5E. The User presses the Scavenger Hunt Hint Button.

6E. The App determines the next Clue Description that will be displayed for the Current Clue.

7E. The App displays the new Clue Description on the Scavenger Hunt Screen.

8E. The App resumes Main Success Scenario at step 5.

**Alternate Flow 6**:

8F. The App requests the Device's GPS Status from the Operating System and determines that the GPS is either disabled or the Device does not have a GPS signal.

9F. The App displays an Error Message indicating that the User should check their GPS settings.

10F. The User presses the ok button on the Error Message.

11F. The App resumes Main Success Scenario at step 5.

**Alternate Flow 7**:

1G. The App receives an interrupt signal from the operating system and looses focus.

2G. The App regains focus and resumes at the step where it had lost focus.

**Title: Service Search**

**Description:** Allows the User to filter through a list of Services and select one to learn more about.

**Pre-conditions:**

* The App is open and has focus.

**Post-conditions:**

* The App displays the Service Results Screen information about the Service the User pressed on.

**Actors:**

* User
* App

**Main Success Scenario:**

1. The User presses the Service Menu Button.
2. The App gathers a list of Service names from the Service Database.
3. The App displays the Service Search Screen.
4. The User presses a Service List Item on the Service Search Screen.
5. The App finds the Service Data from the Service Database for the selected Service.
6. The App displays the Service Results Screen with the Service Data that was found.
7. If the Service is not a Favorite Service, the App displays the Add to Favorites Button.

**Alternate Flow 1:**

1A. The User presses on the Search Bar at the top of the Service Search Screen.

2A. The App opens the User's Device's default On-Screen Keyboard.

3A. The User enters input into the Search Bar with the On-Screen Keyboard.

4A. The App filters, each time the Search Bar input is changed, the available Service List Items by those whose Button Text contains a substring of the Search Bar's inputted text.

5A. The User hits the enter key on the On-Screen Keyboard.

6A. The App displays the Service Search Screen.

**Alternate Flow 2**:

1B. The App receives an interrupt signal from the operating system and loses focus.

2B. The App regains focus and resumes at the step where it had lost focus.

**Title: Take Me There**

**Description:** Allows the User to request navigation to the displayed Service Data's GPS Coordinates.

**Pre-conditions:**

* The App is open and has focus.
* The App is displaying the Service Results Screen.
* The Take Me There Button has been provided by the Service Results Screen.

**Post-conditions:**

* The App opens the Operating System's default navigation application.

**Actors:**

* User
* App

**Main Success Scenario:**

1. The User presses the Take Me There Button.
2. The App requests the Device's GPS Status from the Operating System and determines that the User's Device has GPS enabled and has GPS signal.
3. The App finds the GPS Coordinates for the displayed Service Data.
4. The App requests that the Operating System open the default Navigation Application with directions to the found GPS Coordinates.
5. The Navigation Application launches and gains focus.

**Alternate Flow 1:**

2A. The App determines that the User's Device doesn't have GPS enabled.

3A. The App displays an Error Message indicating that the User must enable GPS in the Operating System's settings.

4A. The User presses the Error Message's ok button.

5A. The App displays the Service Results Screen.

**Alternate Flow 2**:

2B. The App determines that the User's Device has GPS enabled and the Device doesn't have a GPS signal.

3B. The App displays an Error Message indicating that a GPS signal couldn't be found.

4B. The User presses the Error Message's ok button.

5B. The App displays the Service Results Screen.

**Alternate Flow 3**:

1C. The App receives an interrupt signal from the operating system and loses focus.

2C. The App regains focus and resumes at the step where it had lost focus.

**Title: Visit Website**

**Description:** Allows the User to easily access and view a Service's website.

**Pre-conditions:**

* The App is open and has focus.
* The App is displaying the Service Results Screen.

**Post-conditions:**

* The App opens the Operating System's default web browsing application to the Service's website URL.

**Actors:**

* User
* App

**Main Success Scenario:**

1. The User presses on the Service's website URL.
2. The App displays a Prompt that asks the User if they would like to visit the Service's website.
3. The User presses on the Prompt's yes button.
4. The App requests that the Operating System opens its default web browsing application to the Service's website URL.
5. The web browsing application launches and takes focus.

**Alternate Flow 1:**

3A. The User presses on the Prompt's no button.

4A. The App displays the Service Results Screen.

**Alternate Flow 2:**

1B. The App receives an interrupt signal from the operating system and loses focus.

2B. The App regains focus and resumes at the step where it had lost focus.

**Title: Where am I**

**Description:** Allows the User to view information about the Building they are closest to.

**Pre-conditions:**

* The User has successfully installed the App.
* The App is open and displaying the Home Screen.

**Post-conditions:**

* The App displays the Service Results Screen with information about the User's Current Location.

**Actors:**

* User
* App

**Main Success Scenario:**

1. The User presses the Where Am I Home List Item.
2. The App displays the Where Am I Screen.
3. The App requests the Device's current GPS Status from the Operating System and the Device's GPS is enabled and has signal.
4. The App requests the User's current GPS Coordinates from the Operating System.
5. The App queries the Service Database for the Building closest to the User's current GPS Coordinates.
6. The App sets the User's Current Location to the returned Building granted that its distance is less than 1600 meters away.
7. The App displays the Service Results Screen with the Service Data that is mapped to in the Service Directory by the User's Current Location.

**Alternate Flow 1**:

6A. The App cannot find a Service whose determined distance from the User is less than 1600 meters.

7A. The App displays an Error Message indicating that the User is too far away.

8A. The User presses the Error Message's ok button.

9A. The App displays the Home Screen.

**Alternate Flow 2**:

3B. The App requests the Devices's current GPS Status from the Operating System, but the User's Device can't find a GPS signal.

4B. The App displays an Error Message that indicates there was an issue gathering requested data.

5B. The User presses the Error Message's ok button.

6B. The App displays the Home Screen.

**Alternate Flow 3**:

3C. The App requests the Devices's current GPS Status from the Operating System, but the User's Device does not have GPS enabled.

4C. The App displays an Error Message that the User must enable GPS in the Operating System's settings.

5C. The User presses the Error Message's ok button.

6C. The App displays the Home Screen.

**Alternate Flow 4**:

1D. The App receives an interrupt signal from the operating system and loses focus.

2D. The App regains focus and resumes at the step where it had lost focus.