**Use Case Template**

**Project Name: Project ID: Executive Sponsor: Project Manager: Business Analyst:**

Date: September 12, 20

**Table of Contents**

**APPROVALS ……………………………………………………………………………………………………..…4**

**USE CASE LIST ……………………………………………………………………………………….………..….5**

**1 FEATURE NAME (EXAMPLE: ATM TRANSACTION) ……..…………………………………...….….….5**

1.1 FEATURE PROCESS FLOW / USE CASE MODEL …………………………………………….5

1.2 USE CASE(S) …………………………………………………………………………………..……5

**2 FEATURE NAME (REPEAT FOR EACH USE CASE) ……………………………………………………..7**

2.1 FEATURE PROCESS FLOW / USE CASE MODEL …………………………………………….7

2.2 USE CASE(S) ………………………………………………………………………………………..7

Revision History

|  |  |  |
| --- | --- | --- |
| **Version** | **Date** | **Revision Description** |
| .01 |  |  |
| .02 |  |  |
| .03 |  |  |
| .04 |  |  |
| 1.0 |  | Approved Use Case |
|  |  |  |
|  |  |  |

**Approvals**

We have carefully assessed the Use Cases for this project. This document has been completed in accordance with the requirements of the System Development Methodology.

MANAGEMENT CERTIFICATION - Please check the appropriate statement.

the document is accepted.

the document is accepted pending the changes noted.

the document is not accepted.

We fully accept the changes as needed improvements and authorize initiation of work to proceed. Based on our authority and judgment, the continued operation of this system is authorized.

(\*=Required \*\*= Submit for Review Approval Not Required)

Executive Sponsor\*\* DATE

Project Sponsor\* DATE

Quality Assurance Manager / Team Lead\* DATE

Business Analyst Manager / Team Lead\* DATE

Project Manager DATE

**Use Case List**

|  |  |  |
| --- | --- | --- |
| Use Case ID | Primary Actor | Use Cases |
| UC-1.2.1 |  |  |
| UC-1.2.2 |  |  |

**1 Feature Name:** User Profile Access

* 1. **Feature Process Flow / Use Case Model**
  2. **Use Case(s)**

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC-1.2.1 | | |
| Use Case Name: | Create Profile | | |
| Created By: | Dylan Sporrer | Last Updated By: | Dylan Sporrer |
| Date Created: | 9/13/2018 | Last Revision Date: | 9/13/2018 |
| **Actors:** | Primary: Profile-less User  Secondary: E-Mail Service | | |
| **Description:** | This case is for the event that a user new to the system wishes to create a profile, allowing them to access the full feature set | | |
| **Trigger:** | A user not logged in to any existing profile chooses to create one | | |
| **Preconditions:** | 1. User is not logged in to any existing profile | | |
| **Postconditions:** | Success Guarantees:   1. User login info is stored securely in database 2. User is logged in to new profile | | |
| **Normal Flow:** | 1. User selects “Create New Profile”  2. System prompts user to enter email address, password, and password confirmation  3. System validates no profile exists with given email address  4. System validates password matches restrictions  5. System sends profile confirmation message to provided email address  6. System accepts response from confirmation page  7 User is logged in to new profile | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Profile with Given Email Exists]** | 3a. In step 3 of the normal flow, if a profile exists with the given email  1. System will prompt user to login to existing profile or attempt to create new profile  2. User selects existing profile login  3. User enters UC-1.2.2  3b. In step 3 of the normal flow, if a profile exists with the given email  1. System will prompt user to login to existing profile or attempt to create new profile  2. User selects create new profile  3. Use Case resumes on step 2 of normal flow | | |
| **[Alternative Flow 2 – New Password Invalid]** | 4a. In step 4 of the normal flow, if the password constraints are not met  1. System informs user of issue  2. Use case resumes on step 2 of normal flow | | |
| **Exceptions:** | 5a. In step 4 of the normal flow, if the user enters and invalid email  1. Customer informed that email is non-existent  2. Use Case resumes on step 2 of normal flow | | |
| **Includes:** | N/A | | |
| **Frequency of Use:** | On Demand; High volume early in life of product, low volume later | | |
| **Special Requirements:** | 1. Ability to store login information securely  2. Ability to interact with common email platforms | | |
| **Assumptions:** | 1. User has existing email address and is capable of utilizing it  2. User is capable of creating good-strength password | | |
| **Notes and Issues:** | 1. Maximum password length and contents TBD | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC-1.2.2 | | |
| Use Case Name: | Access Profile | | |
| Created By: | Dylan Sporrer | Last Updated By: | Dylan Sporrer |
| Date Created: | 9/13/2018 | Last Revision Date: | 9/17/2018 |
| **Actors:** | Primary: Any User | | |
| **Description:** | This case is for the event that a user wishes to access an existing user profile. | | |
| **Trigger:** | A user not logged in to any existing profile chooses to log into one. | | |
| **Preconditions:** | 1. User is not logged in to any existing profile.  2. At least one user profile exists in the database which is not in use | | |
| **Postconditions:** | Success Guarantees  1. User is logged in to existing profile  2. Login info remains secure in database | | |
| **Normal Flow:** | 1. User selects “Log In”  2. System prompts user to enter email and associated password  3. System validates that email and password uniquely identify an existing and not-in-use user profile  4. User gains access to existing profile | | |
| **Alternative Flows:**  **[Alternative Flow 1 – No Existing Profile Found]** | 3a. In step 3 of the normal flow, if the email and password do not uniquely identify an existing user profile  1. System informs user of inability to locate expected profile  2. Provided password field is cleared  3. Resume normal flow at step 2 | | |
| **Alternative Flows:**  **[Alternative Flow 2 – Specified Profile in Use]** | 3b. In step 3 of the normal flow, if the uniquely identified profile is currently in use  1. System informs user profile is in use  2. Provided login fields are cleared  3. Resume normal flow at step 2 | | |
| **Exceptions:** | N/A | | |
| **Includes:** | N/A | | |
| **Frequency of Use:** | On Demand | | |
| **Special Requirements:** | 1. Ability to store login data securely | | |
| **Assumptions:** | 1. User knows the login information to a pre-existing user profile | | |
| **Notes and Issues:** | N/A | | |

**2 Feature Name (Repeat for multiple features)**

**2.1 Feature Process Flow / Use Case Model**

**2.2 Use Case(s)**

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC-1.2.3 | | |
| Use Case Name: | Add Pin to Map | | |
| Created By: | David Rowan | Last Updated By: |  |
| Date Created: | 9/23/18 | Last Revision Date: |  |
| **Actors:** | Primary: Any User | | |
| **Description:** | This case is for the event in which the user adds a pin to the map to list a occurrence in Oakland. | | |
| **Trigger:** | A user chooses the add pin action | | |
| **Preconditions:** | 1.User is logged into the system  2. The user is allowed to post things to the map i.e. no permissions revoked | | |
| **Postconditions:** | 1. Pin is added to map  2. Pin location is added into the database  3. Other users can see the pin | | |
| **Normal Flow:** | 1. User Selects “Add Pin” on map 2. User the fills in the create a pin boxes(Location, Occurrence, etc) 3. User clicks “Create” and the pin is added to the map. | | |
| **Alternative Flows:** | 1. In Step 3 a error message appears because a the pin has already been added to that location by another user. 2. In Step 3 the user enters a location outside the bounds of the program. | | |
| **Exceptions:** | N/A | | |
| **Includes:** | N/A | | |
| **Frequency of Use:** | On Demand | | |
| **Special Requirements:** | 1. Ability to store data and retrieve it 2. Access to google maps api | | |
| **Assumptions:** | 1. Google maps is working 2. The database is connected and running | | |
| **Notes and Issues:** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC-1.2.4 | | |
| Use Case Name: | Upvote and downvote pin on map | | |
| Created By: | David Rowan | Last Updated By: |  |
| Date Created: | 9/23/18 | Last Revision Date: |  |
| **Actors:** | Primary: Any User | | |
| **Description:** | This case is the case where a user can upvote/downvote a existing pin on the map to allows other users to see how helpful the pin is. | | |
| **Trigger:** | A user upvotes a pin | | |
| **Preconditions:** | 1.User is logged into the system  2. The user is allowed to post things to the map i.e. no permissions revoked | | |
| **Postconditions:** | 1. Pin has updated count | | |
| **Normal Flow:** | 1. User selects Upvote/Downvote 2. Selection is added to pin | | |
| **Alternative Flows:** | 1. In Step 4 the user has revoked permission and is not allowed to upvote or downvote pins. | | |
| **Exceptions:** | N/A | | |
| **Includes:** | N/A | | |
| **Frequency of Use:** | On Demand | | |
| **Special Requirements:** | 1. Ability to store data and retrieve it 2. Access to google maps api | | |
| **Assumptions:** | 1. Google maps is working 2. The database is connected and running | | |
| **Notes and Issues:** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: |  | | |
| Use Case Name: |  | | |
| Created By: |  | Last Updated By: |  |
| Date Created: |  | Last Revision Date: |  |
| **Actors:** |  | | |
| **Description:** |  | | |
| **Trigger:** |  | | |
| **Preconditions:** |  | | |
| **Postconditions:** |  | | |
| **Normal Flow:** |  | | |
| **Alternative Flows:** |  | | |
| **Exceptions:** | N/A | | |
| **Includes:** | N/A | | |
| **Frequency of Use:** |  | | |
| **Special Requirements:** |  | | |
| **Assumptions:** |  | | |
| **Notes and Issues:** |  | | |