# Use Case # [4: Changing a user’s password]

|  |  |
| --- | --- |
| GENERAL CHARACTERISTICS | |
| **Author** | Jordan Persson |
| **Last Update:** | 9/23/2017 |
| **Scope** | Home Automation System |
| **Level** | User-goal |
| **Status** | Incomplete Conceptualization |
| **Primary Actor** | Mobile Application User |
| **Secondary Actors** | Server; Mobile Application |
| **Stakeholders and Interests** | Mobile App User: Wants to change their user password |
| **Preconditions** | User has Mobile application installed and registered. |
| **Success Post Condition** | The password has successfully been reset |
| **Failed Post Condition** | The password was not reset. |

|  |  |
| --- | --- |
| MAIN SUCCESS SCENARIO (or basic flow) | |
| **Step** | **Action -** description in words of each step in success scenario |
| 1 | The user navigates to the profile settings page on the mobile application and clicks “Change Password” |
| 2 | The mobile app displays the change password screen and asks the user for their old password, the new password, and to re-enter the new password. |
| 3 | The user inputs this information and clicks submit. |
| 4 | The mobile app sends the old password and new password to the server. |
| 5 | The server confirms that the password matches and sets the new user password. |
| 6 | The server sends a confirmation success to the mobile app. |
| 7 | The mobile app informs the user that the password has been updated. |

|  |  |
| --- | --- |
| EXTENSIONS or Alternate Flows | |
| **Step** | **Branching Action** |
| *n..m* | \*a At any time the server fails:   1. Server attempt to find and fix issues   1a. Server is offline  1. App informs the user that no server is detected  2. User restarts the server  3. App reconnects to the server    1b. Server crash  1. Server auto restart  1a. Server fails to auto restart  1. Mobile app informs the user after 1 minute of no response from the server  2. User manually restarts server   1. Server requests information from the mobile application and resumes normal functionality   \*b At any time the mobile application fails:  1. Mobile app searches for issue  1a. Mobile app has no network connection  1. App attempts to connect to wifi to restore connection  1a. App fails to connect to wifi  1. The information is stored on the app and queue to send when connection is restored  2. The app informs the user that it has no network connection.  2. App sends information to the server  2. Mobile application crashes  1a. Mobile app fails to auto restart  1. Mobile app sends information about the cause of the crash  2. User manually restarts the app  3. Mobile app sends its information to the server and reestablishes connections based on what the server sends back |
| 1a. The user is not logged in to the mobile app.  1. The user selects the forgot password option  2. The mobile app asks for the user’s email associated with the accout.  3. The user inputs the email.  3a. The user forgot the email.  1. The user clicks forgot email.  2. The mobile asks for the phone number or username associated with the account.  2a. The user doesn’t know this information.  1. The user must create a new account.  3. The user inputs the information and clicks submit.  4. The mobile app sends the information to the server.  5. The server confirms the information and sends a reset password link to the email for the account.  5a. Account not found.  1. The server will tell the mobile app that the information was not associated with a known account.  2. The mobile app will inform the user that the account was not found.  3. The user can repeat step 1a.  6. The user follows that link and resets their password.  3a. The new password and conformation entry do not match  1. The mobile app informs the user that the passwords do not match.  2. The user must reenter the passwords and click submit.  3b. The password doesn’t met security standards  1. The mobile app checks the password against its security standards and tells the user what criteria the password is missing.  2. The user inputs a new password and submits.  5a. The old password doesn’t match what is on the server  1. The server tells the mobile app that the passwords didn’t match  2. The mobile app tells the user that an invalid password was used.  3. The users inputs new password and submits.  5b. The new password and the old password were the same  1. The server tells the mobile app that the new and old passwords were the same.  2. The mobile app tells the user that the new password cannot match the old one.  3. The user puts in a new password and submits. |
|  |  |

|  |  |
| --- | --- |
| SPECIAL REQUIREMENTS | |
| **Req Num** | **Requirement** |
| *n* | 1. Server communications require authentication 2. Multiple failed reset attempts will cause the server to send a warning email to the email associated with the account. |

|  |  |
| --- | --- |
| TECHNOLOGY AND DATA VARIATIONS LIST | |
| **Var Num** | **Variation** |
| *n* | \*a. Communications are done over a wireless network so server would need internet and mobile device would need mobile data or be connected to wifi  \*b. Keyboard is required to restart server as admin passwords would be needed |

***FREQUENCY OF OCCURRENCE***: Often. It would occur as often as the user changes rooms

|  |  |
| --- | --- |
| OTHER ISSUES | |
| **Issue Num** | **Issue** |
| *n* | 1. How many failed attempts should we allow before locking out an account? 2. How will we encrypt the passwords? |