# Use Case # [Disable/ remove speaker]

|  |  |
| --- | --- |
| GENERAL CHARACTERISTICS | |
| **Author** | Yibing Zhang |
| **Last Update:** | 9/25/2017 |
| **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 remove or disable a speaker |
| **Preconditions** | User has Mobile application installed and registered.  Music source is in the system. |
| **Success Post Condition** | The speaker is successfully removed |
| **Failed Post Condition** | The speaker remains in the system |

|  |  |
| --- | --- |
| MAIN SUCCESS SCENARIO (or basic flow) | |
| **Step** | **Action -** description in words of each step in success scenario |
| 1 | The user uses the mobile application and navigates to the “remove a device” menu |
| 2 | The mobile app displays the options of devices to remove |
| 3 | The user selects the speaker option. |
| 4 | The mobile app instructs the user to navigate the speaker and then click remove. |
| 5 | The mobile app sends the information to the server. |
| 6 | The server removes the speaker from its database. |
| 7 | The speaker is now removed. |

|  |  |
| --- | --- |
| 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.  2a. Mobile application crashes  1. Mobile app sends information about the cause of the crash  2. App attempts to auto restart  2a. Fails to auto restart  1. User manually restarts the app  2. App sends information to the server  3. Mobile app sends its information to the server and reestablishes connections based on what the server sends back |
| 4. The user cannot find the speaker  1. The user must ensure that the speaker is in the system |

|  |  |
| --- | --- |
| SPECIAL REQUIREMENTS | |
| **Req Num** | **Requirement** |
| *n* | 1. Speaker connections restricted to users 2. Server communications require authentication 3. Server communication allowed outside of the local network |

|  |  |
| --- | --- |
| TECHNOLOGY AND DATA VARIATIONS LIST | |
| **Var Num** | **Variation** |
| *n* | 7a. Communications are done over a wireless network so server would need internet and mobile device would need mobile data or be connected to wifi  7b. Keyboard is required to restart server as admin passwords would be needed |

***FREQUENCY OF OCCURRENCE***: Not often. It only occurs when the speaker is broken or the user gets a new speaker

|  |  |
| --- | --- |
| OTHER ISSUES | |
| **Issue Num** | **Issue** |
| *n* | 1. How will device be located? 2. What will we do once we remove a device? Should we have an alternative or reset the room boundaries? 3. What if we remove a device that is currently used by another user? |