# Use Case # [Turning off speaker by leaving a room]

|  |  |
| --- | --- |
| GENERAL CHARACTERISTICS | |
| **Author** | John Clarke |
| **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 for their music to follow them as they change rooms by connecting to different speakers depending on location. |
| **Preconditions** | User has Mobile application installed and registered.  User is in a room that has a speaker currently playing music  User has another speaker in separate room |
| **Success Post Condition** | Speaker in room that User is leaving is no longer playing music |
| **Failed Post Condition** | The speaker in room that User has left is still playing music. |

|  |  |
| --- | --- |
| MAIN SUCCESS SCENARIO (or basic flow) | |
| **Step** | **Action -** description in words of each step in success scenario |
| **1**  **2**  **3**  **4** | The User is in a room with a speaker that is currently playing music  The User leaves room that has speaker currently playing music  The mobile app connects to a speaker in the new room that is closer to the User than the previous speaker  Speaker in previous room stops playing music |
|  |  |

|  |  |
| --- | --- |
| 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 |
| 3a. No speaker is found for that room, speaker to play to is already in use, speaker is turned off  1. The system tells the mobile app that no new speaker is available  2. The mobile app remains connected to the old speaker |
|  |  |

|  |  |
| --- | --- |
| 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* | 2a. Communications are done over a wireless network so server would need internet and mobile device would need mobile data or be connected to wifi  2b. 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 will device be authentication? 2. How will the speaker’s information be stored? |