# Design Documentation

# Smart House Project

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 10/09/2015 | 0.1 | Description of device development design. | Jabir Al Fatah, Liaquath Hassan |
| 23/09/2015 | 0.2 | Description of device development design. | Jabir Al Fatah, Paul Damon, Liaquath Hassan |
| 17/10/2015 | 0.3 | Description of device development design. | Jabir Al Fatah |
|  |  |  |  |

Design item List

|  |  |
| --- | --- |
| **Requirement Name** | **Priority** |
| D1. Connecting the device(s) with microcontroller. | Essential |
| D2. Writing sketch that can communicate with the devices back and forth. | Essential |
| D3. Implementing and integrating Xbee radio module for Wi-Fi communication. | Essential |
| D4. Passing the information to the server back and forth via Wi-Fi. | Essential |

Design Item Descriptions

### D1

Smart house contains devices (both built in and virtual). Our Microcontroller i.e. Arduino will be connected to the devices.

### D2

For controlling the devices, updating information to the server/database and implementing Xbee radio module we need to write program in Arduino environment. Program will be written such a way that it can exchange data either ways. For example before controlling any device the Arduino has to know the current status of the device. That is, before sending any command to the device; the microcontroller must be aware whether the device is already turned on or off.

**D3**

Because we have to send the update to the server over the internet, we have chosen Wi-Fi for effective communication. Selected module for wireless communication is Xbee. We need to have necessary setup and program for the communication through Xbee.

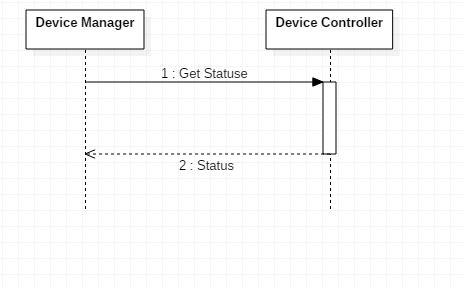
**D4**

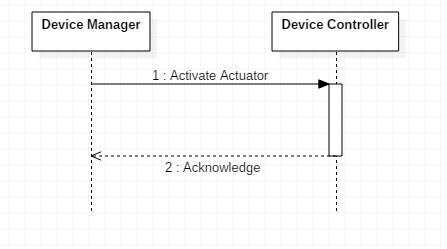
All the information about the changes in devices must be sent to the server in order for dynamic communication.

**UML diagram**

1. **Sequence diagram**

We use sequence diagram to show the actions. The diagrams are given below:





1. **State diagram**

