

**REQUIREMENT ANALYSIS & USABILITY**

**REPORT**

**<Project Title>**

**Home Automation System**

**<submitted by>**

Group Number # 32

**<Client>**

Dr. Mansoor Ebrahim

**<Submitted to>**

Dr. Mansoor Ebrahim

**<GitHub Repository Link>**

https://github.com/Muhammad-Rujail-hamza-Shaikh/IOT-BASED-HOME-AUTOMATION-SYSTEM.git

* List of team members and email addresses.

**Group Members Name & Email:**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.NO** | **NAME** | **ID** | **EMAIL** |
| 1 | Muhamad Rujail Hamza Shaikh | 14092 | rujail.hamza@gmail.com |
| 2 | S. M. Ali Jaffry | 14022 | alijaffry19@gmail.com |
| 3 | Hunain Ahmad | 14118 | hunainahmad11@gmail.com |
| 4 | Muhammad Ahmad Faheem | 14119 | muhammadahmadfaheem@gmail.com |

* **FUNCTIONAL REQUIREMENTS:**

**Software System Attributes:**

Some system attributes are given below:

**1. Reliability:**

**1.1 Mean Time between Failures:**

The Home Automation System shall not fail on average more than once per week.

**1.2 Mean Time to Repair:**

The Home Automation System shall not take on average longer than two days to repair after a system failure.

**2. Availability:**

The system shall be available 95% of the time unless previously announced for scheduled maintenance or backup.

**2.1 Degraded Mode of Operations:**

In the event of the unavailability of the system, the customers shall see a temporary unavailable screen or response message when requesting the system services.

**2.2 System Maintenance Notification:**

In the event that the system will undergo foreknown system maintenance or data backup, users of the system shall be notified during next interaction with the system.

**3. Security:**

Security points are given below:

**3.1 Customer Confidentiality:**

The system shall not reveal any personal data in correlation to usage scenarios or statistics.

**3.2 Development Access:**

All development work shall be password protected at or above the rigor of the Iowa State Computer Science password standards at web server (Adafruit).

**4. Maintainability:**

Maintenance points are given below:

**4.1 System Reset:**

The system shall provide the ability to reset all settings to the default or a saved configuration.

**4.2 Developer Access:**

The system administrator shall be able to grant developer privileges to users.

**4.3 Code Maintainability:**

The programming code developed for the system shall conform to programming style standards and shall be commented thoroughly as determined by a code review team.

**4.4 Portability:**

The system shall allow all information and applications to be run from the mobile or local infrastructure.

* **NON-FUNCTIONAL REQUIREMENTS:**

To make this project, basic requirements are given below:

**Relay module 4 channel**



This relay module allows you to combine the processing power of the Arduino to devices that use higher current and voltage.  It does so by providing four relays that are rated for 7A at either 28VDC or 10A at 125VAC.

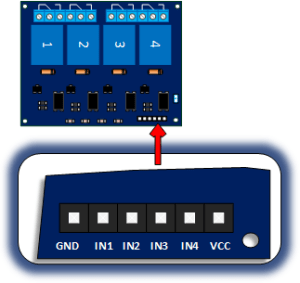
Each relay has a Normally Open (NO) and a Normally Closed (NC) contact.

With these relays you can control:

 Appliances:

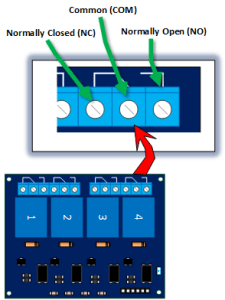
* Motors
* Lights
* Other Relay

**Relay Module Inputs:**

* The module is supplied with power via the pin labeled VCC and ground via the pin labeled GND.
* **[](https://i0.wp.com/henrysbench.capnfatz.com/wp-content/uploads/2015/04/Four-Channel-Relay-Inputs.png)**The relays are energized with low inputs to the IN1, IN2, IN3 and IN4 inputs

**Relay Module Outputs.**

There are four relays that each provide dry contact outputs.  That is to say that each relay provides a common (COM) , normally open (NO) and a normally closed (NC) terminal.

**[](https://i2.wp.com/henrysbench.capnfatz.com/wp-content/uploads/2015/04/Four-Channel-Relay-Outputs.png)**

**WIFI Module:**

The ESP8266 WIFI Module is a self-contained SOC with integrated TCP/IP protocol stack that can give any microcontroller access to your WIFI network. The ESP8266 is capable of either hosting an application or offloading all Wi-Fi networking functions from another application processor.

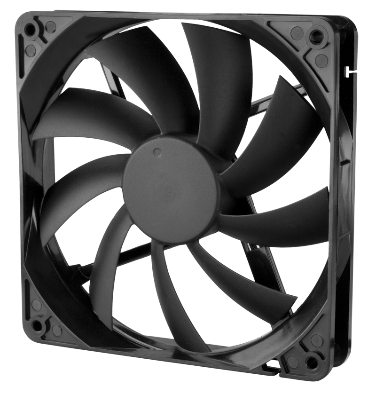
**Relay Module:**

This relay module allows you to combine the processing power of the Arduino to devices that use higher current and voltage.  It does so by providing four relays that are rated for 7A at either 28VDC or 10A at 125VAC.

**PIR sensor:**

A passive infrared sensor (PIR sensor) is an electronic sensor that measures infrared (IR) light radiating from objects in its field of view. They are most often used in PIR-based motion detectors.

**FAN Cooler’s:**

A computer fan is any [fan](https://en.wikipedia.org/wiki/Fan_(machine)) inside, or attached to, a [computer case](https://en.wikipedia.org/wiki/Computer_case) used for [active cooling](https://en.wikipedia.org/wiki/Active_cooling). Fans are used to draw cooler air into the case from the outside, expel warm air from inside, and move air across a [heat sink](https://en.wikipedia.org/wiki/Heat_sink) to cool a particular component. Both [axial](https://en.wikipedia.org/wiki/Axial_fan) and sometimes [centrifugal](https://en.wikipedia.org/wiki/Centrifugal_fan) (blower/squirrel-cage) fans are used in computers. Computer fans commonly come in standard sizes, and are powered and controlled using 3- or 4-pin fan [connectors](https://en.wikipedia.org/wiki/Electrical_connector).

Jumper wires:

Jumper wires are simply wires that have connector pins at each end, allowing them to be used to connect two points to each other without soldering. Jumper wires are typically used with [breadboards](https://blog.sparkfuneducation.com/what-is-a-breadboard) and other prototyping tools in order to make it easy to change a circuit as needed. Fairly simple. In fact, it doesn’t get much more basic than jumper wires.

**Led 12v:**

A light-emitting diode (LED) is a [semiconductor](https://en.wikipedia.org/wiki/Semiconductor) [light source](https://en.wikipedia.org/wiki/Light_source) that emits light when [current](https://en.wikipedia.org/wiki/Electric_current) flows through it.