

SMART ROOM APPLICATION

REPORT



Shoaib Ahmed

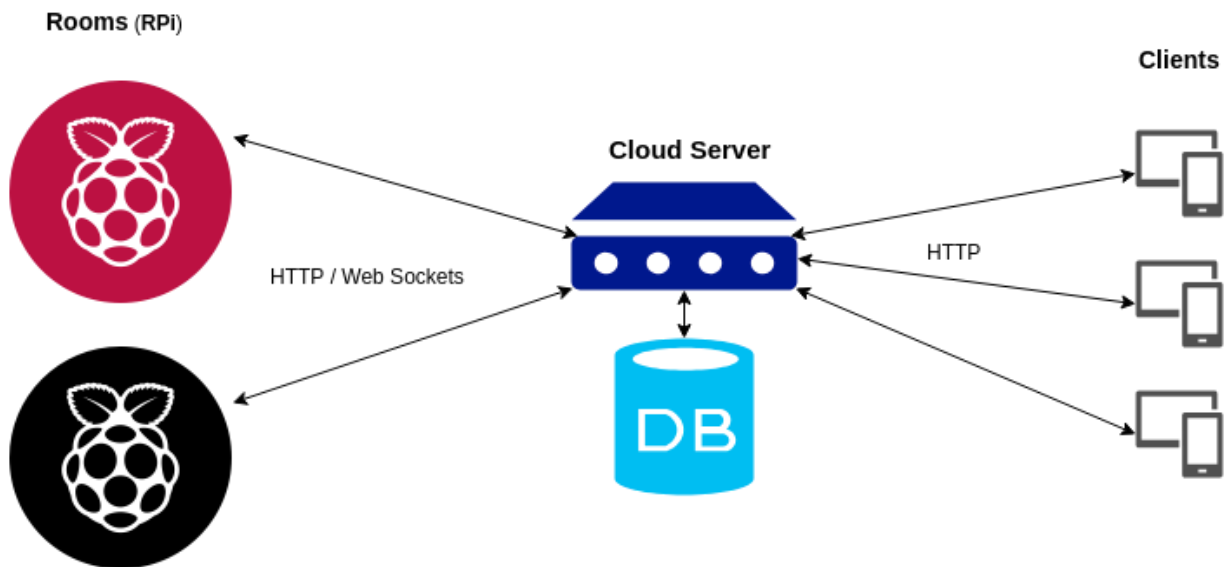
Rahul Korthiwada

Sai Kumar Immadi

IIIT Guwahati | [Github Repo](#)

VISION

Smart Room System is designed with the objective of providing remote control of electrical appliances based on real-time sensing of room conditions. The users can interact with this cloud-based system through a web-based, real-time interface, which also shows the room conditions in real-time. The system uses Wi-fi for communication with the cloud.



Technologies Used

The Smart Room Application uses the following technologies. All the Software Technologies used in this project were chosen keeping in mind their Scaling Capability, Reliability, Security features, Speed, and Popularity. The Hardware Components used in the project were of prototypic nature.

→ Software

- **Node.js:** A Javascript Runtime Environment for server-side programming.
- **Socket.io:** To enable real-time Websockets-like data transfer to and from the Pi device. Powered by Node.js.
- **Express.js:** A Node.js server that runs on both RPi as well as on the cloud server.
- **Heroku:** A cloud platform-as-a-service used to host the cloud server.
- **mLab:** A cloud database service used to host the MongoDB instance for the cloud server.
- **React.js:** A Javascript library used to build single page frontend applications.
- **JWT:** A JSON based Web token system for Server Authentication used to maintain Login information of the user on the React app in a stateless manner.

→ Hardware

- **Raspberry Pi:** To collect and send sensor data to the server and to receive and execute control information from the cloud.
- **PIR Sensor:** Passive InfraRed sensor to detect the presence of humans in a room
- **DHT-11:** Temperature and Humidity Sensor
- LEDs, Pushbutton, Bread Board, and Jumper Wires.

Working of the application :

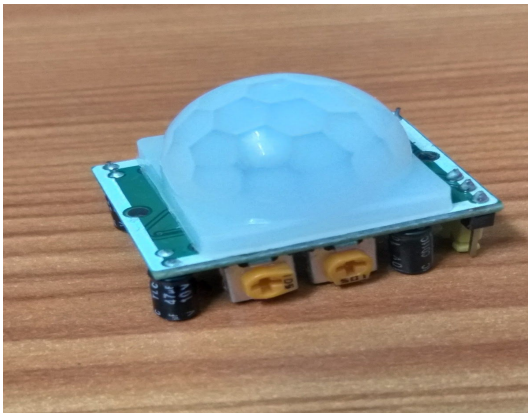
The application has two modes- Manual and Auto. The Manual mode of the application allows for remote control of appliances through the internet. The Auto mode allows for smart control of the room's appliances by the cloud server. The prototype that we built uses a PIR sensor to detect human presence in the room and send this information to the cloud server for decision-making. The RPi device is in charge of sending sensed information to the cloud and receiving and actuating control information from the cloud.

The flow of information in the Auto mode is listed below:

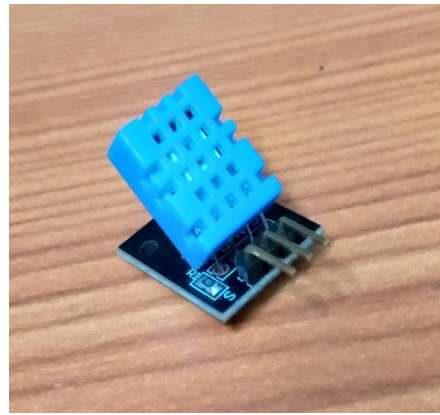
1. The User enables Auto Mode remotely through the Web Application.

2. The Cloud server goes into auto mode and starts sensing a human's presence in the room using the PIR sensor.
3. If human presence is detected, the cloud server sends a message to the RPi asking it to switch On the appliance.
4. If the human leaves the room, the cloud server waits for about 10 seconds before asking the RPi to switch-off the appliance.
5. All the state information from the room i.e. On/Off status of the switch, On/Off status of the auto mode, an temperature and humidity statistics of the room can be viewed from the Web App in real-time.

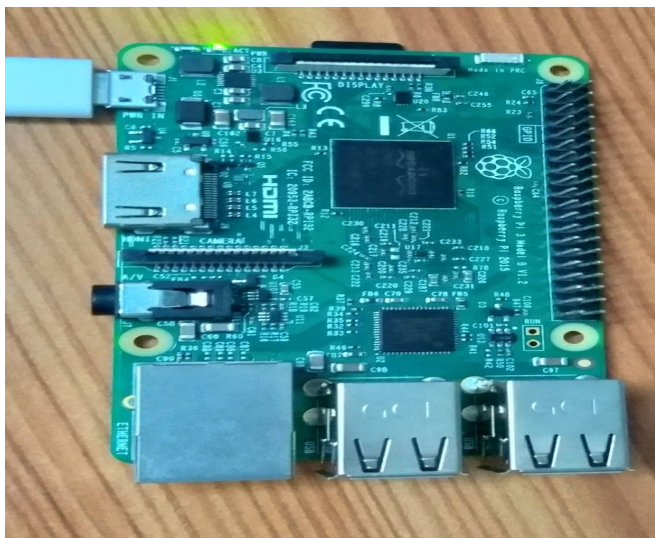
Images of Hardware and working:



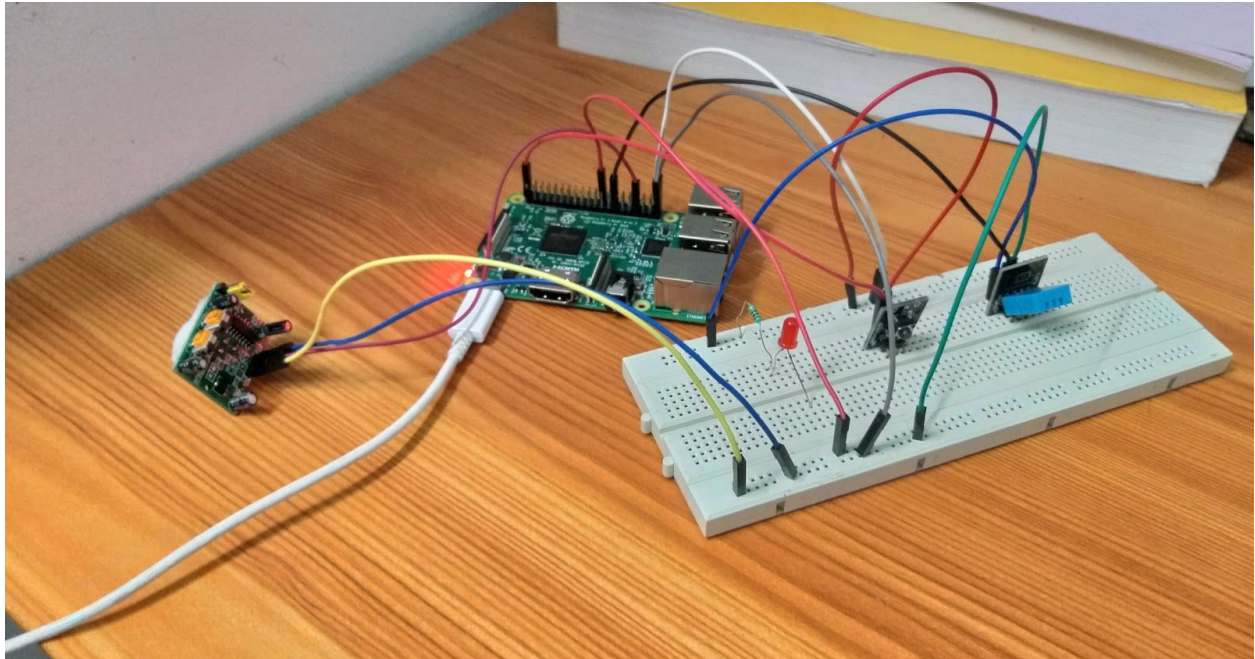
PIR Sensor



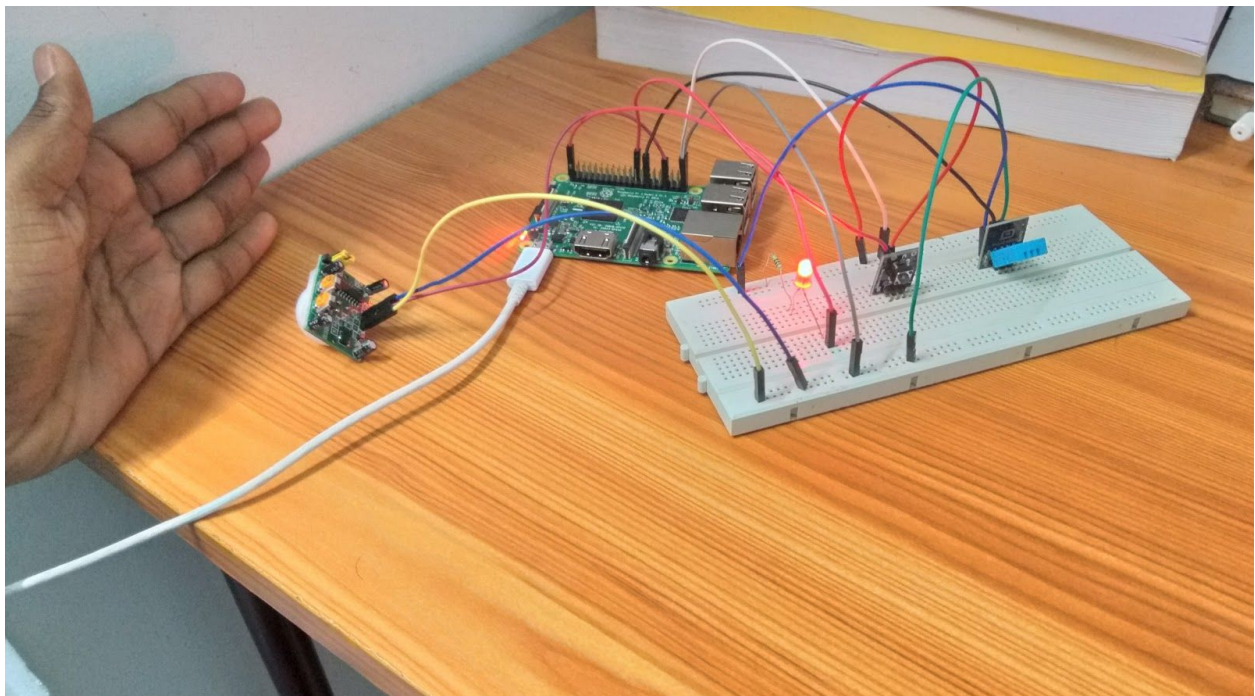
DHT-11 Sensor



Raspberry Pi Device



The LED is used in place of an actuator to show the basic working of the device. When in Auto mode, if there is no human presence detected, the LED remains Off.



When a human's presence is detected, the device switches On the LED. The PIR sensor has a range of 5-10 meters.

Links :

1. The code can be found [here](#).
2. The Web Application can be found at: <https://smarty-room.surge.sh>
3. Login details :- uname: test@test.com pass : Testtest1
4. Some initial python scripts to operate the sensors on RPi can be found [here](#).
5. The design documents of the Smart Room application can be found [here](#).