

SWE3011 – IoT Programming

GROUP ASSIGNMENT

TOPIC: SMART HOME LED CONTROL SYSTEM

Vu Duc Quan – 103432334

Ta Quang Huy – 103423516

Nguyen Gia Huy – 103441107

Nguyen Minh Tu – 103441055

Contents

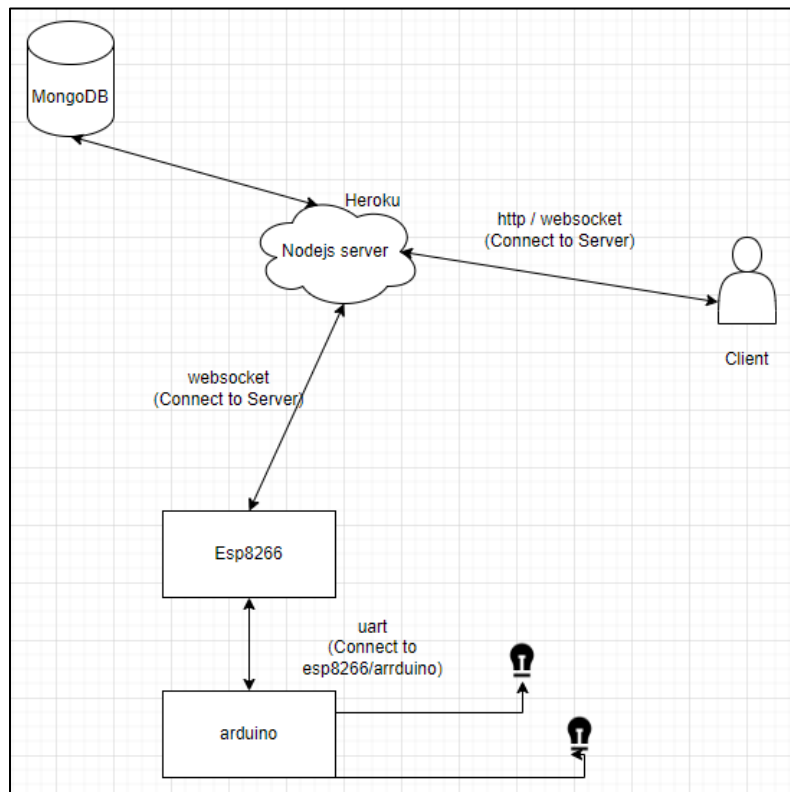
Introduction	2
Conceptual Design	2
Our project system’s workflow:	2
Tasks Breakdown:	3
Implementation	3
Sensing system:	4
Communication Protocols:	4
API or Website Detail + Cloud Computing:	4
Database:	5
User Manual:	5
Limitations:	6
Resources:	6
Appendix:	6

Introduction

In today's world where energy management and conservation is a talked subject and smartphones are reshaping our lives, "Smart Home Lighting System" is one such area which is gaining lot of attraction in lighting industry. For high efficiency, long lifetime, better light quality and reliability, LED is already a winner in lighting technology. Adding smartness to LED light further boost its superiority. Selective turn on/turn off and brightness control results in increased system life and reduced energy consumption.

The proposed system that we designed functions as a smart solution to control lighting system in smart home via wifi connection through a web interface with a database system that record state histories of the leds.

Conceptual Design



Our project system's workflow:

As a first step, whenever the Client interacts with an interface by pressing the "Toggle" button, a signal or an input is delivered immediately to the server from the client computer.

The signal from the server (the Nodejs Server) will be received by the ESP8266 (ESP8266 is connected with Wifi Service through the Websocket protocol).

The ESP8266 is linked to the Arduino Uno R3 that has connected to the 2 LED lights.

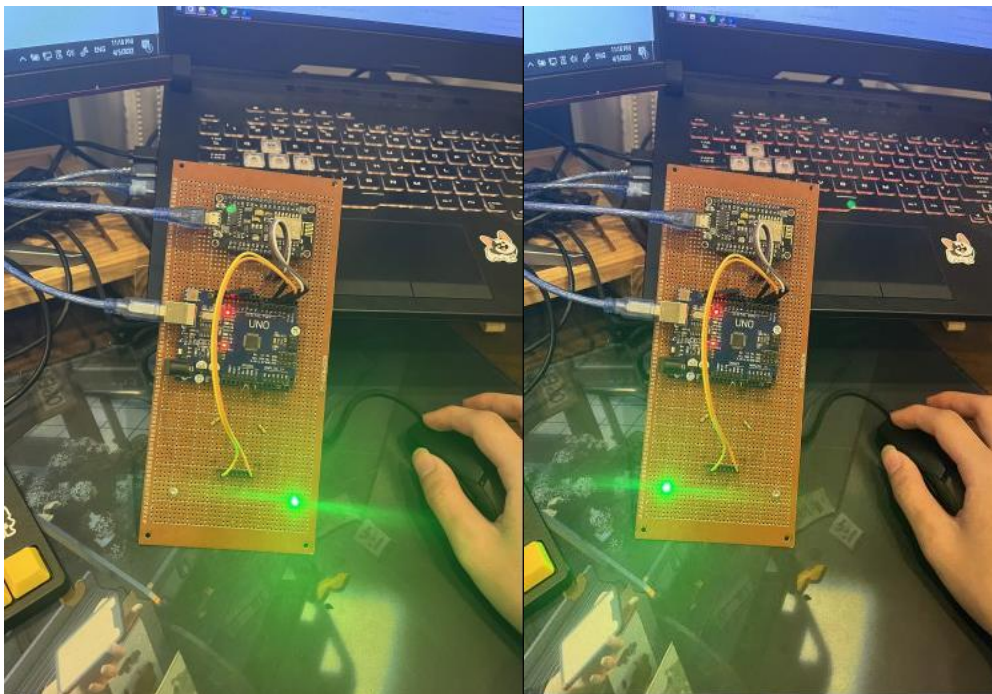
In order to successfully complete the "Toggle" function, we must give an input signal to the ESP8266, which will then transmit data back to the client, informing them that their request was received and processed successfully by the ESP8266 and server.

At the same time, The state of led data and time recorded data is sent directly to the Database system (MongoDb).

Tasks Breakdown:

Tasks	Hardware Preparation	Code Arduino + ESP8266	Code Web API + Database	Deploy Cloud Computing Platform
Vu Duc Quan	X		X	X
Ta Quang Huy		X		X
Nguyen Gia Huy	X	X		
Nguyen Minh Tu			X	X

Implementation



Sensing system:

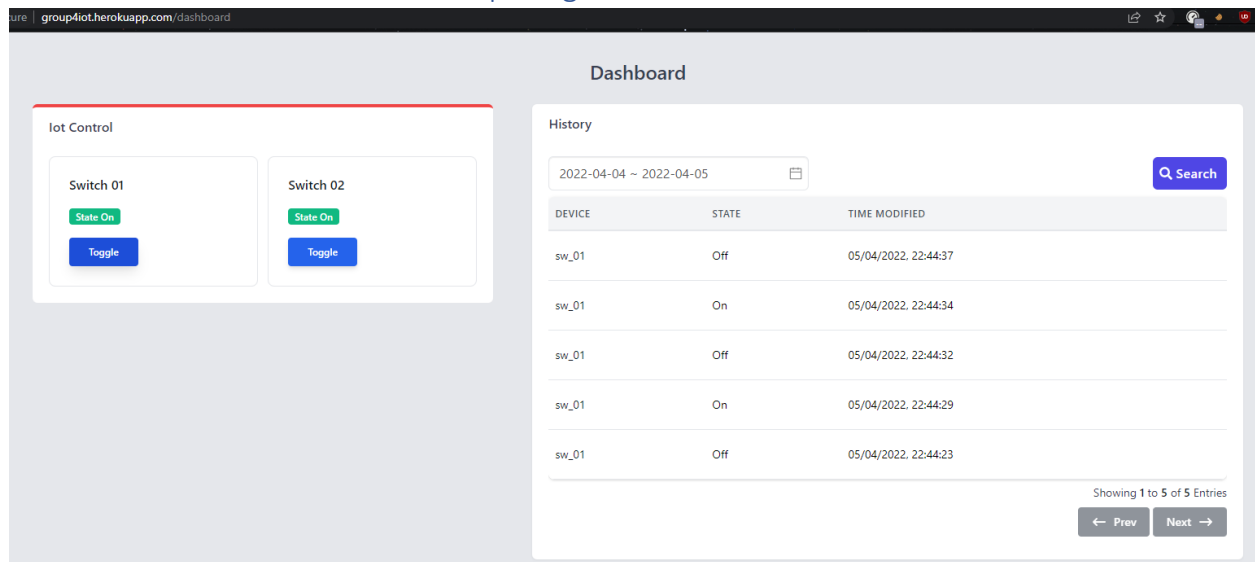
Led lights

Communication Protocols:

Websocket Protocol (Setup environment for Wifi to connect to the Website)

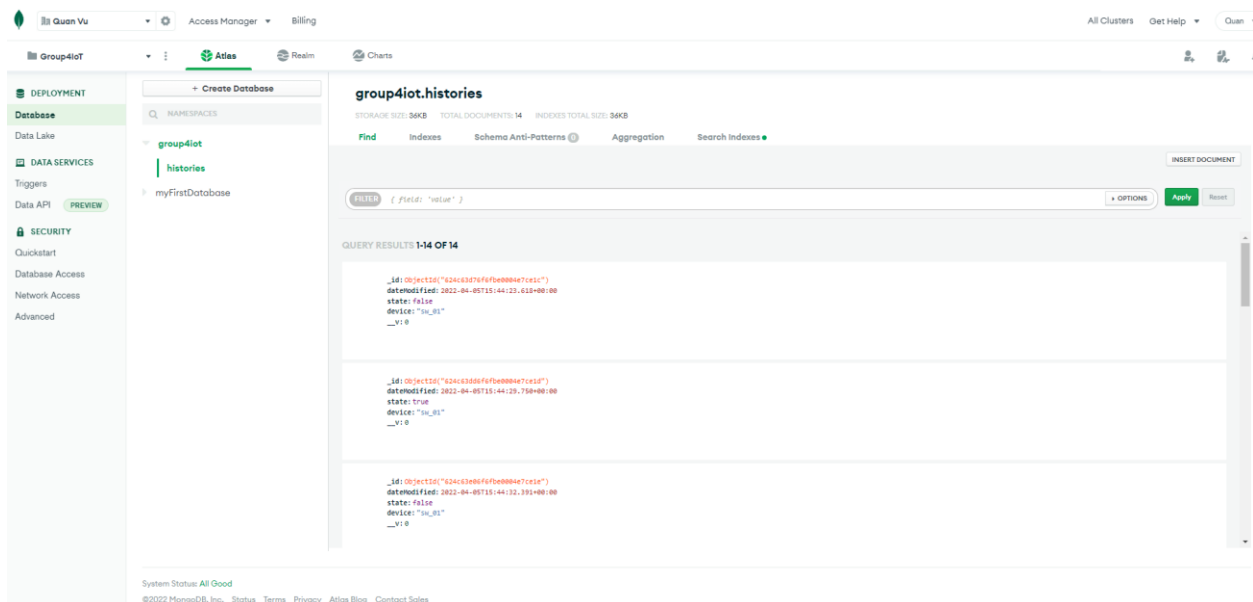


API or Website Detail + Cloud Computing:



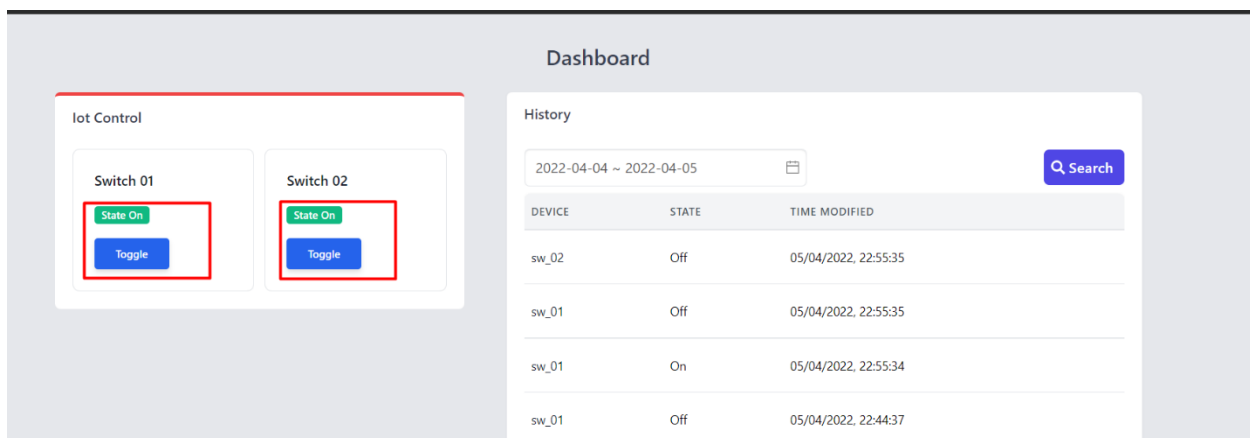
We use Heroku app service in order to develop our website API and the Cloud Computing System altogether. With the aid of this service, we can access the website and control the system via multiple devices without them being conflicted.

Database:



For the database system, we use MongoDB Service in order to push the state histories of led control onto the database.

User Manual:



Client can interact with our system through the Website. To control the Led system, the Client simply need to click on the “Toggle” button to turn on or off the desirable light. For simplicity, we only implement 2 led lights into our project, for future plan, we can implement more devices such as AC, stove, or door system, etc.

Limitations:

For now, our limitations are:

- Simple Idea
- Only 2 main functions: Led Toggle and Histories Record

Resources:

We are inspired by this project linked above and from this one we decided to improve and expand the functions and implement Website API as well as the Cloud Computing platform.

[ESP8266 Make your own LED control web server in Arduino IDE | IoT project](#)

[ESP8266 Web Server and Websockets Using Arduino IDE \(Mac OSX and Windows\) | vs. AJAX and HTTP](#)

[Node.js Crash Course Tutorial #9 - MongoDB](#)

[Heroku Tutorial For Beginners - Deploy Your App to Heroku Under 5 Minutes! \(Heroku Tutorial\)](#)

Appendix: