

IoT Project : Phillips Hue Manager

Robin Camarasa¹

¹Ecole des mines de Saint-Etienne, Saint-Etienne, France

1 Introduction

This project is an practical application of an 160 hours Computer Sciences lesson. 3 main fields were studied to build this project : Internet of Things, Web Programming and Mobile Programming. The first aim is to create a fully operational a Android application and a website and a local hardware controller to manage a Phillips Hue lamp. The second aim is to create a mobile and a web front-end to manage a full building with Phillips Hue lamps and sound managers. For the second aim no hardware realization is required because of the cost. You will find an illustration of the project in figure 1 at page 1.

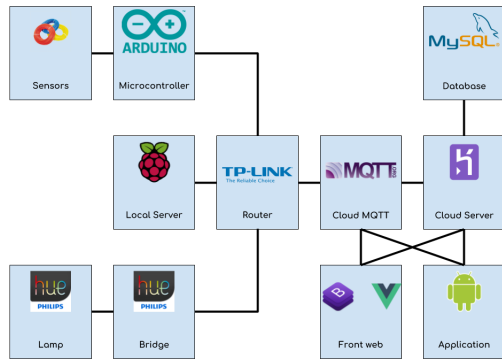


Figure 1: Project structure

2 Hardware

2.1 Local wiring

A Phillips Hue lamp was connected to a Phillips Hue bridge which was wired to the router. Because of the security protocol of our school security and privacy policy the router was connected with an RJ45 cable to a Raspberry Pi that transformed the wifi into Ethernet.

2.2 Arduino

We used an Arduino Uno as a micro-controller. A switch button and a potentiometer were wired to input ports of the micro-controller. The switch button allowed us to switch the light on or off and the potentiometer helped us to adjust the hue value of the lamp. The hardware controllers always had the priority on the software controllers in order to tackle external attacks.

2.3 Communication Protocols

The communication protocols used were MQTT[1] and HTTP. The HTTP was used to create a connection between the front and the back-end. The MQTT allowed communication between the hardware and the cloud. In the hardware part, the Arduino Uno was published

on different topics and the Cloud MQTT subscribed and published to those topics. The Raspberry Pi had a Java program that subscribed to those topics and sent HTTP requests to the Phillips Hue bridge API to control the lamp.

3 Back-end

3.1 Database

The database was really simple and contained only one table LIGHT that contains the state of the lamp, the hue value, the brightness value and the saturation value.

3.2 Implementation

To implement that database we use Springboot[3], a Java Framework. With this framework we created a REST API that was hosted on an heroku[2] server. A MySQL[4] addon were added on the heroku server to store the database.

4 Front-end

4.1 Front Web

The web interface was hosted on a personal github.io page. This page were coded in HTML, CSS and Javascript. Two frameworks were used : Vuejs a JS framework that allowed us to have easily an interactive page and Bootstrap a module that gave us ready-to-use components.

4.2 Front Android

We created the Android application with Android Studio. Two main language were used :

XML to organize the graphical structure of the application and Java to manage the shaping of the data extracted from the database. You will find the render of the application in figure 2 at page 2.

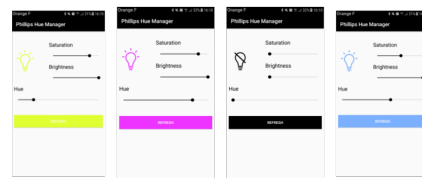


Figure 2: Application render

5 Generalization

5.1 Backend

For the back-end technology used was the same technology described in the Back-end section but the data structure were more sophisticated with multiple tables linked together with relation like one to one or one to many. You will find a description of this data structure in figure 3 at page 3.

5.2 Front-end

Like for the back-end the front-end used the same technology described in the section Front-end. This part of the project made us improve our self in the use of Android and Bootstrap components. You will find the render of the application in figure 4 at page 3.

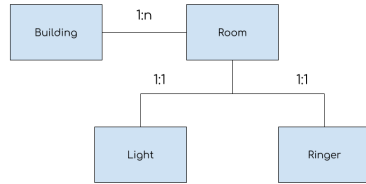


Figure 3: Database structure

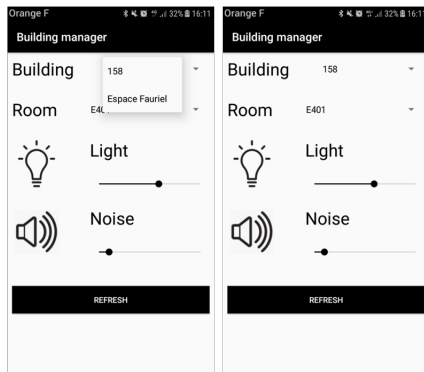


Figure 4: General application render

6 Conclusion

This project led to the realization of a multi-plate-form IoT domotic solution that could be generalized to the management of a full building. One of the important limits of our system is the security.

References

- [1] Light, R.A., 2017. Mosquitto: server and client implementation of the MQTT proto-

col. *Journal of Open Source Software*, 2(13).

- [2] Middleton, N. and Schneeman, R., 2013. *Heroku: Up and Running: Effortless Application Deployment and Scaling*. " O'Reilly Media, Inc."
- [3] Arthur, J. and Azadegan, S., 2005, May. Spring Framework for rapid open source J2EE Web Application Development: A case study. In *Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing*, 2005 and First ACIS International Workshop on Self-Assembling Wireless Networks. SNPD/SAWN 2005. Sixth International Conference on (pp. 90-95). IEEE.
- [4] MySQL, A.B., 2001. MySQL.