VIETNAM NATIONAL UNIVERSITY HO CHI MINH CITY
HO CHI MINH UNIVERSITY OF TECHNOLOGY
FACULTY OF COMPUTER SCIENCE & ENGINEERING

SOFTWARE ENGINEERING (CO3109)

PROJECT

Home Automation System Automatic Light & Fan

TA: M.E Vu Trong Thien
Students: Nguyen Ngoc Tan - 1915059
Nguyen Duy Phuoc - 1914768
Nguyen Dang Tu - 1912384
Bui Nguyen Duc Tung - 1910676
Le Quang Dat - 1913032

Ho Chi Minh City, April 2022

Contents

1 Introduction

Home Automation System (Smart Home) is an ubiquitous topic nowadays. People become hard and harder on choosing technology devices - Everything has to be automatic, one-touch, that can control home devices: Air conditioner; Lights; Electricity system... with smartphones, voice commands, or even without command at all.

Currently, there are some applications can be listed[1]:

- Ring video doorbell with WiFi camera - Notifies the smartphones or other electronic device of the home owner when a visitor arrives at the door. It activates when the visitor presses the button of the doorbell, or when the doorbell senses a visitor with its built-in motion sensors. The smart doorbell lets the home owner use a smartphone app to watch and talk with the visitor.

- Nest Learning Thermostat - Showing weather's impact on energy usage: this is a smart thermostat that optimizes heating and cooling of homes and businesses to conserve energy.

- Smart Automatic Cat Feeder - Gives the food in scheduled-time and adjust the amount of food after every time the cat eats.

In this project, we will implement a system which can turn on/off lights and fans automatically based on environment condition. Moreover, we can remotely control lights and fans. In details:

- Automatic system: Placing sensors around the house for the purpose of sending data to server, which in turn sending to the controller for it to make decision whether turning on/off the light. Simultaneously, server sends a notification to the home owner to notify that the light state has been changed.
- Remote control: The home owner can use our mobile app to turn on/off, change the power of lights and fans.

Automatic Home Lighting Control System

2 Requirement

Functional Requirement

1. Show the state of lights and fans in the house.
   - Display on the mobile app the state of light: *ON* or *OFF*.
   - Display on the mobile app the state of the fan: *OFF* or *ON* (LOW, MEDIUM, HIGH).
2. Automatically turn the lights on and off.
   - Turn the light ON when light intensity is greater than *600 lux*.
   - Turn the light OFF otherwise
3. Automatically turn the fans on and off.
   - Temperature is less than or equals to 32°C: turn the fan OFF.
   - Temperature is greater than 32°C: turn the fan ON at LOW degree.
   - Temperature is greater than 35°C: turn the fan ON at MEDIUM degree.
   - Temperature is greater than 37°C: turn the fan ON at HIGH degree.
4. Allow the user to turn lights and fans on/off by mobile application: Users is allowed to turn ON/OFF or change amplitude if they want to.
5. Display the weather forecast with advice on app UI (extend): Display the temperature, humidity and light intensity on the mobile app, like a weather forecast engine.
6. Automatically measure and send data: Data from sensors is sent to server every 30 seconds.

Non-Functional Requirement

1. Sensors and mobiles must connect to the Internet *24 hours a day*.
2. Data must be transferred in *real time*.

3 List Devices

See *References*[2]

1. ChipI - Humidity & Temperature Sensor.

This sensor is popular used in household devices. It can measure temperature 0 - 50°C, humidity 20 - 90%.

2. ChipI - Light Sensor.

This sensor has a wide area that can sense the environment. It is high sensitive to change.

3. ChipI - Motor DRV.

Provide ability to change the fan direction.

4. Propeller.

5. ChipI - 2-Color LED.

Red and Blue colors.

4 Use case

4.1 Definitions

4.1.1 Actors

Actor is behaviour classifier which specifies a role played by an external entity that interacts with the subject (e.g., by exchanging signals and data), a human user of the designed system, some other system or hardware using services of the subject [3]. Types of actors including: users, database systems, clients, servers, cloud platforms, devices,... [4].
Actors in our system:

- User (home owner): interact with mobile application
- Temperature and humidity sensor: interact with MQTT server
- Light sensor: interact with MQTT server
- Back-end server: interact with mobile application
- MQTT server: interact with I/O system and mobile application
- Mobile application: interact with back-end server

4.1.2 Use case

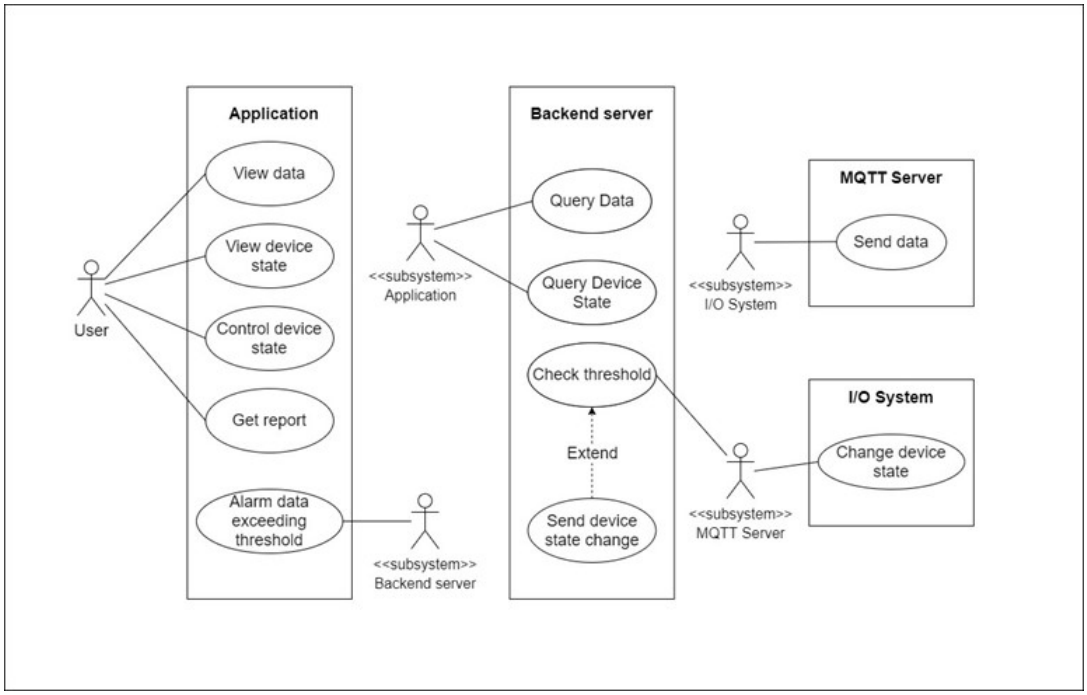
A **use case** is a kind of behaviour classifier that specifies a [complete] unit of [useful] functionality performed by [one or more] subjects to which the use case applies in collaboration with one or more actors, and which [for complete use cases] yields an observable result that is of some value to those actors [or other stakeholders] of each subject [9].

Use cases in our system:

- MQTT server:
  - Send temperature and humidity data
  - Send light intensity data
- Back-end server:
  - Check the temperature
  - Check the light intensity
- Mobile application:
  - Notify the temperature exceed/fall below the threshold
  - Notify the light intensity exceed/fall below the threshold
  - Notify the fan's state has changed
  - Notify the light's state has changed
  - Change the fan's state
  - Change the light's state
  - Display the light's current state

  - Display the fan's current state
  - Display the current temperature
  - Display the current light intensity
  - Generate report
- I/O system:
  - Change the fan's state
  - Change the light's state

4.2 Use case diagram



4.3 Use case description

Term Notes: *device* means both fan and light; *data* means both temperature, humid and light intensity.

Author	Nguyen Duy Phuoc
Date created	Wed, 2 Mar 2022
Use case name	Send data
Actor	I/O system (sensor)
Description	Every 30 seconds, the sensor measures and sends new data to MQTT
Trigger	The timer measures time intervals and sends trigger signal every 30 seconds
Preconditions	The sensor must subscribe to a topic in MQTT server
Postconditions	Backend server receives new data
Normal flow	1. The sensor measures the new data; 2. The sensor publishes new data to the MQTT server; 3. Back-end server receives data from MQTT server.
Alternative flow	None
Exception	Exception 1 at step 1: 1.a. The sensor is broken so it doesn't measure temperature every interval. Use case stops; Exception 2 at step 2: 2.a. The sensor is not connected to the internet. Use case stops.
Non-functional requirements	Data must be transferred in at most 1 second

Table 1: *Send data*

Author	Nguyen Duy Phuoc
Date created	Wed, 2 Mar 2022
Use case name	Change device state
Actor	MQTT server
Description	Send message to change the device state
Trigger	MQTT server sends message to I/O system (device controller)
Preconditions	I/O system (device controller) must subscribe to the MQTT server
Postconditions	Device state changes
Normal flow	1. I/O system receives the message from MQTT server; 2. I/O system change the device to new state included in the message.
Alternative flow	None
Exception	Exception 1 at step 1: 1.a. The device controller is not connected to the internet. Use case stops; Exception 2 at step 2: 2.a. The device controller is broken so it doesn't change the device state. Use case stops.
Non-functional requirements	Device state must be changed in at most 1 second

Table 2: *Change device state*

Author	Nguyen Duy Phuoc
Date created	Wed, 2 Mar 2022
Use case name	Alarm data exceeding threshold
Actor	Backend server
Description	Send alarm to mobile app that data exceeds the threshold
Trigger	Check threshold in Backend server returns EXCEED THRESHOLD flag
Preconditions	App must register token with the Notification service
Postconditions	An alarm notification is sent to mobile app
Normal flow	1. Backend server sends message to the Firebase Cloud Messaging service; 2. Firebase Cloud Messaging service sends message to mobile app; 3. Mobile app receives and display the alarm.
Alternative flow	None
Exception	Exception 1 at step 3: 3.a. Mobile isn't connected to the internet. The alarm will be display when it is reconnected.
Non-functional requirements	Message must be transferred in at most 1 second; Mobile must be connected to the internet

Table 3: *Alarm data exceeding threshold*

Author	Nguyen Dang Tu
Date created	Wed, 2 Mar 2022
Use case name	View data
Actor	User
Description	User can view the current data in mobile app
Trigger	User open mobile app in home page (or control page)
Preconditions	Mobile must connect to the internet
Postconditions	Current data is displayed in the home page or control page
Normal flow	1. Mobile app gets data from query return value in Backend server; 2. Current data is show in the mobile screen.
Alternative flow	None
Exception	Exception 1 at step 1: 1.a. Mobile isn't connected to the internet. The mobile displays an error.
Non-functional requirements	Mobile must be connected to the internet; Mobile must get result in at most 1 second.

Table 4: *View data*

Author	Nguyen Dang Tu
Date created	Wed, 2 Mar 2022
Use case name	View device state
Actor	User
Description	User can view the current state of device in mobile app
Trigger	User open mobile app in control page
Preconditions	Mobile must connect to the internet
Postconditions	Current device state is displayed in the control page
Normal flow	1. Mobile app gets device state from query return value in Backend server; 2. Current data is show in the mobile screen.
Alternative flow	None
Exception	Exception 1 at step 1: 1.a. Mobile isn't connected to the internet. The mobile displays an error.
Non-functional requirements	Mobile must be connected to the internet; Mobile must get result in at most 1 second.

Table 5: *View device state*

Author	Bui Nguyen Duc Tung
Date created	Wed, 2 Mar 2022
Use case name	Control device state
Actor	User
Description	User can control the device remotely
Trigger	User clicks the switch or button on mobile screen
Preconditions	Mobile must connect to the internet
Postconditions	A new state message will be published to MQTT server
Normal flow	1. A new state is returned when user clicks the switch or button; 2. Compare this state with device current state; 3. If (new state != current state): Mobile app will publish a message containing the new state to MQTT server.
Alternative flow	None
Exception	Exception 1 at step 3: 3.a. Mobile isn't connected to the internet. Use case stops.
Non-functional requirements	Mobile must be connected to the internet; Data must be transferred in at most 1 second.

Table 6: *Control device state*

Author	Bui Nguyen Duc Tung
Date created	Wed, 2 Mar 2022
Use case name	Get report
Actor	User
Description	User can get report about average of data record and device usage in a particular of time
Trigger	User opens app in page Report
Preconditions	The mobile must connect to the Internet
Postconditions	Mobile app will display a report screen
Normal flow	1. User chooses from date and to date in Report page; 2. User clicks button GO; 3. Data and device usage in an interval of time (from date, to date) is return from Backend server query; 4. Average data and device usage is calculate in mobile app; 5. A report screen is display
Alternative flow	Alternative 1 at step 1: 1.a. User clicks button that is predefined interval time, which is in list Today, Yesterday, This Week, This Month. Continue to step 3.
Exception	Exception 1 at step 2: 2.a. User chooses invalid date. An error will be displayed.
Non-functional requirements	Report screen must be displayed in at most 0.5 second.

Table 7: *Get report*

Author	Le Quang Dat
Date created	Wed, 2 Mar 2022
Use case name	Query sensor records
Actor	Mobile application
Description	The mobile app can query sensor records
Trigger	Mobile app calls the function in Backend server to query
Preconditions	The mobile must connect to the Internet
Postconditions	Result from the query is returned to mobile app
Normal flow	1. Backend server performs the query on collection Records; 2. Mobile app receives the result of the query from Backend server.
Alternative flow	None.
Exception	Exception 1 at step 2: 2.a. Mobile isn't connected to the internet. Use case stops.
Non-functional requirements	Mobile must be connected to the internet; Mobile app must get result in at most 1 second.

Table 8: *Query sensor records*

Author	Le Quang Dat
Date created	Wed, 2 Mar 2022
Use case name	Query device state
Actor	Mobile application
Description	The mobile app can query device state
Trigger	Mobile app calls the function in Backend server to query
Preconditions	The mobile must connect to the Internet
Postconditions	Result from the query is returned to mobile app
Normal flow	1. Backend server performs the query on collection States; 2. Mobile app receives the result of the query from Backend server.
Alternative flow	None.
Exception	Exception 1 at step 2: 2.a. Mobile isn't connected to the internet. Use case stops.
Non-functional requirements	Mobile must be connected to the internet; Mobile app must get result in at most 1 second.

Table 9: *Query device state*

Author	Le Quang Dat
Date created	Wed, 2 Mar 2022
Use case name	Check threshold
Actor	MQTT Server
Description	Check a data from sensor whether it exceeds a threshold or not
Trigger	MQTT server sends a message containing new data from sensor
Preconditions	Threshold must be pre-defined
Postconditions	A flag in EXCEED THRESHOLD, NOT EXCEED THRESHOLD will be returned
Normal flow	1. Backend server query and return threshold of device; 2. Backend server query and return current data of sensor; 3. If data is from light sensor: return EXCEED THRESHOLD if (new data threshold current data) or (new data threshold current data) else return NOT EXCEED THRESHOLD
Alternative flow	Alternative 1 at step 3: 3.a. If data is from temp-humid sensor: return EXCEED THRESHOLD if new data and current data not in the same one of these intervals (0, threshold_low), (threshold_low, threshold_medium), (threshold_medium, threshold_high), (threshold_high, inf) else return NOT EXCEED THRESHOLD.
Exception	Exception 1 at step 2: 2.a. If there is no data of sensor (e.g. in first time app run). NOT EXCEED.
Non-functional requirements	Result of query must returned in at most 1 second.

Table 10: *Check threshold*

Author	Le Quang Dat
Date created	Wed, 2 Mar 2022
Use case name	Send device state change
Actor	Backend server
Description	Back end server will send device new state
Trigger	An EXCEED THRESHOLD flag is returned from check threshold function of Backend server
Preconditions	Backend server must subscribe to MQTT server
Postconditions	A message containing new device state is sent to MQTT server
Normal flow	1. Create a message containing the new device state; 2. 2. Publish the message to the MQTT server.
Alternative flow	None.
Exception	None.
Non-functional requirements	Message must be transferred in at most 1 second.

Table 11: *Send device state change*

5 Database Design

The mobile app UI/UX include these modules:

- Our back-end server subscribes the topic of temperature, humidity and light intensity to get data from sensors as soon as it is published. The temperature, humidity and intensity data will be stored to our database.
- Fan's state will be stored to our database when the temperature exceeds a threshold or the home owner controls by mobile application.
- Light's state will be stored to our database when the light intensity exceeds a threshold or the home owner controls by mobile application.

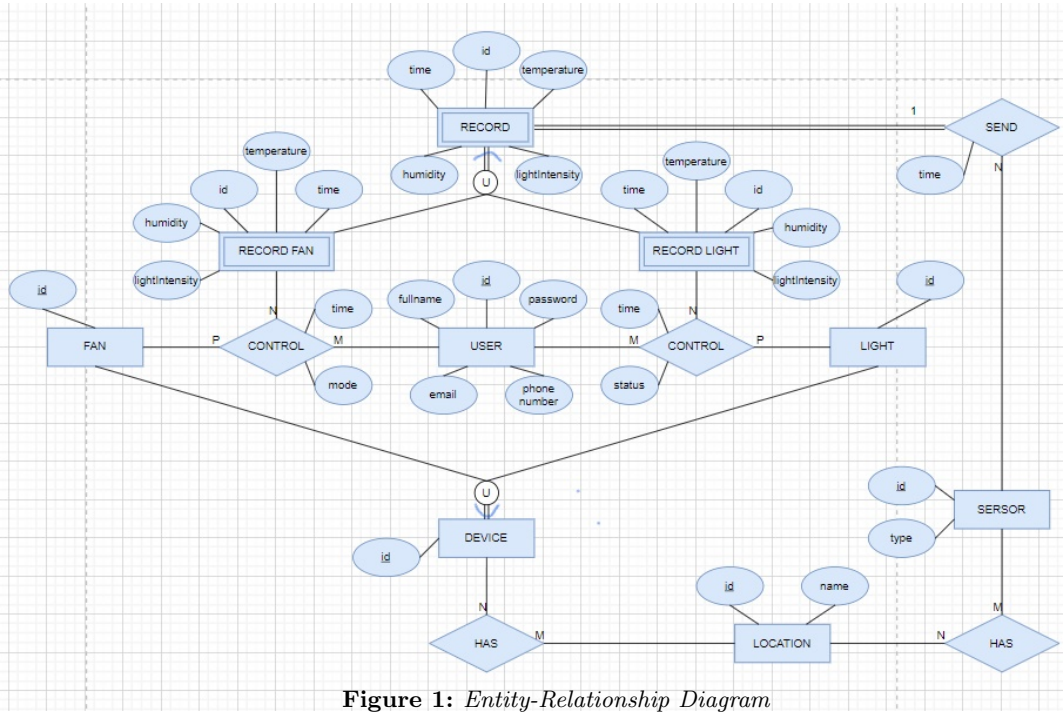


Figure 1: *Entity-Relationship Diagram*

6 UI/UX Design

Live demo: https://drive.google.com/file/d/1JCW5n9UJ405DK93MXqW5r76moJF_lBtL/view

The mobile app UI/UX include these modules:

- Login and sign up: allow users sign up an account to control devices (fans and lights) in their home.
- Data display: display data (temperature, humidity and light intensity) of the environment in the interval of 5 minute users open that screen.
- Device control: allow user to turn ON/OFF or change the mode of devices actively.
- Device's state display: show to user the current states of the devices because their states not only changed by the user but also changed automatically.
- Notification: notify the user when a threshold exceeds and the devices change their states.

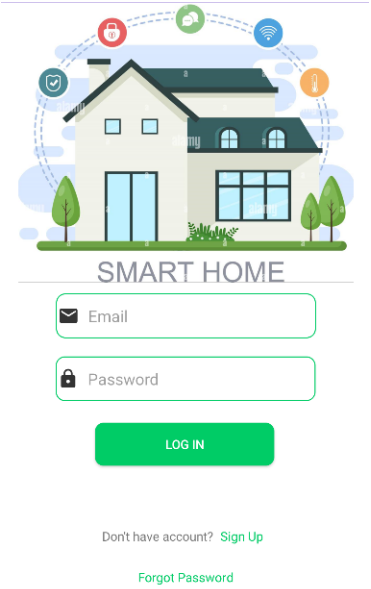


Figure 2: *Login screen*

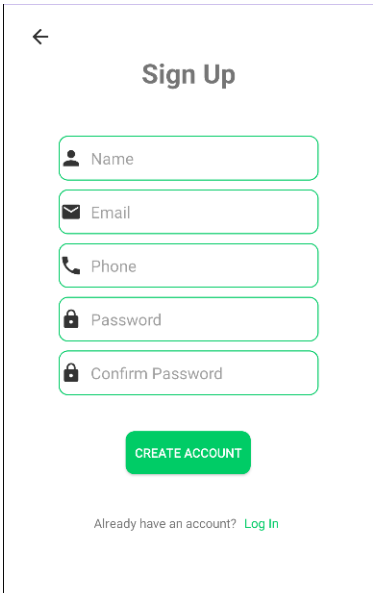


Figure 3: *Sign up screen*

Figure 4: *Home screen*

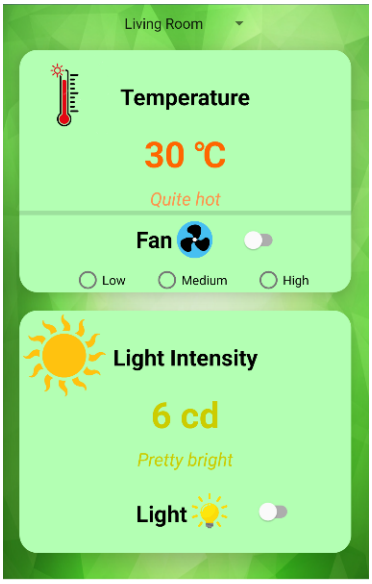


Figure 5: *Control screen*

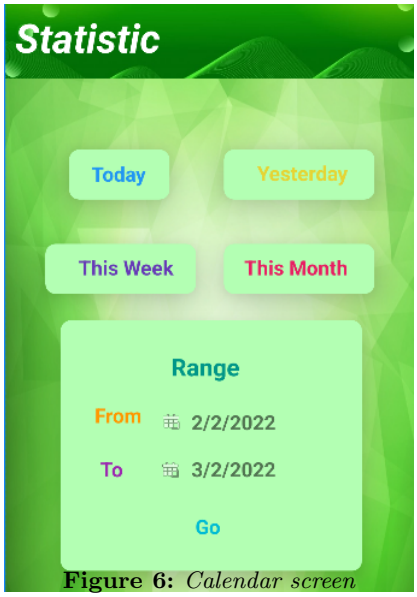


Figure 6: *Calendar screen*

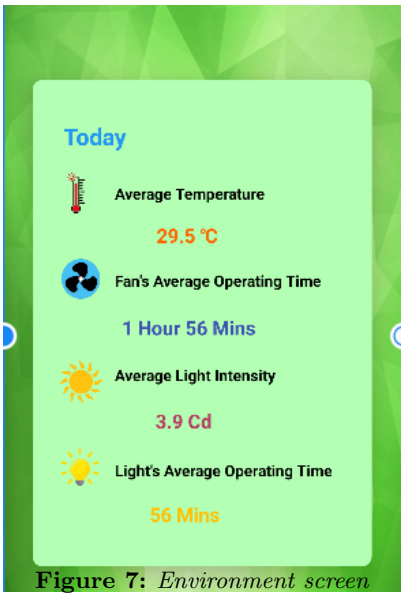


Figure 7: *Environment screen*

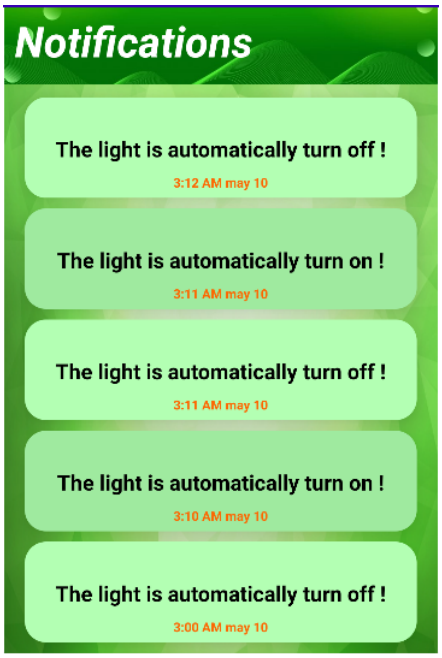


Figure 8: *Notifications screen*

References

[1] Home Automation definition. *https://en.wikipedia.org/wiki/Home_automation*

[2] ChipFC devices. *https://wiki.chipfc.com/index.php?title=Thể_loại:Chipi_Series*

[3] Actor definition. *https://www.uml-diagrams.org/use-case-actor.html*

[4] Actor types. *https://www.visual-paradigm.com/guide/uml-unified-modeling-language/how-to-identify-actors/*

[5] Use case definition. *https://www.uml-diagrams.org/use-case.html*

[6] Android Studio. *https://developer.android.com*

[7] MQTT Server. *https://mqtt.org*

[8] Adafruit. *https://io.adafruit.com*

[9] Firebase. *https://firebase.google.com/docs*