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

MULTIDISCIPLINARY PROJECT (CO3109)

Assignment

WEAFORE - Weather forecast

Class: CC02
Students: Phạm Nguyễn Quang Nhân - 2152206.
Nguyễn Kiều Bảo Khánh - 2152654.
Lâm Gia Khánh - 2152651.
Nguyễn Minh Khoa - 2152131.
Bùi Đăng Khoa - 2152668.
Advisor: Trương Tuấn Anh

HO CHI MINH CITY, MARCH 2024

Contents

Member list & Workload

No.	Fullname	Student ID	Problems	Contribution
1	Phạm Nguyễn Quang Nhân	2152206	Code and test Yolo:Bit device	100%
2	Nguyễn Kiều Bảo Khánh	2152654	Code and design web application	100%
3	Lâm Gia Khánh	2152651	Writing report and leading team	100%
3	Nguyễn Minh Khoa	2152131	Code and test Yolo:Bit device	100%
3	Bùi Đăng Khoa	2152668	Code and design web application	100%

1 Introduction

Weafore is an Iot Yolo:Bit project focused on giving accurate weather prediction in the local area using real-time environmental data from the surrounding area. Based on the prediction, Weafore will also provide assistance in daily life management such as scheduling an effective plan for outing and staying, automatically controlling many house compartments like roof, lighting,... . However, for a Yolo:Bit project, controlling large house compartments is not possible. Therefore, a small model is made instead to demonstrate the possibility of this project.

2 Requirement

2.1 Functional requirement

- The Yolo:bit device collect data about the environment including light level, humidity and temperature
- Data about light level, humidity and temperature is displayed on an LCD screen or on the phone application.
- When it is dark (light level $\leq 20\%$) outside, the LED light, in place of the lighting system, is automatically turned on.
- When the weather changes into rain (humidity $\geq 80\%$) or sunny (light level $\geq 80\%$), the servo motor is activated representing the roof expanding and covering everything below it
- The user can manually control the roof and light by using the phone application.
- The program can forecast weather and display that data on the phone application.
- Only authorized personnel may access the phone application.

2.2 Non-functional requirement

2.2.1 Yolo:Bit device

- Data collected from the environment should be sent to the phone application in less than 3 seconds.
- The roof should open and close at 30 degree/s in order not to damage the roof or the motor.

2.2.2 Phone application

- Weather forecasting function can be used for the next 5 days.
- Controlling roof and light must be responsive (ideally with less than 3 second delay)
- The phone application is a web application.

3 Devices

For the devices to work according to the requirements, components below are required.

3.1 Light sensor

Measure and report data about light in the environment to Yolo:Bit. The component is placed under a clear sky where there is no shadow in the morning.

3.2 Humidity Sensor

Measure and report data about humidity in the environment to Yolo:Bit. The component is placed outdoors, ideally not near any form of water source such as faucet, sprinkler,....

3.3 Temperature sensor

Measure and report data about temperature in the environment to Yolo:Bit. The component is placed under a clear sky where it is fully exposed to the sun.

3.4 LCD screen

Display environmental data on the LCD screen

3.5 Servo Motor

Used for controlling the roof. Speed is set at 30 degree/s to avoid any damage to the roof and ensure it's opened fast enough. This is used in place of the devices that need to control a real house roof described in section 1.

3.6 LED lights

Used for lighting up the surrounding area when night falls. The component emits white light when it's turned on. This is used in place of the lighting system described in section I.

3.7 Yolo:Bit

The main component of the device. This is used for executing all logic regarding home automation, time scheduling and data communication to the homeowner's phone application.

3.8 Extended circuit

The number of other components required an extended circuit to be installed to have enough connections to the device.

4 Use-case details

4.1 Use case 1: Automatically open/close roof

Use-case name	Automatically open/close the roof
Summary	The system opens or close the roof when the environment data gathered by Yolo:Bit sensors satisfy certain conditions. The roof is represented by a servo motor.
Actor	Yolo:Bit sensors
Trigger	None
Pre-condition	The Yolo:Bit device is plugged in and turned on.
Post-condition	None.
Normal flow	If it is sunny (light level $\geq 80\%$) or rainy (humidity $\geq 80\%$): + Yolo:Bit: Send signal to servo motor to open the roof at 30 degree/s.
Alternative flow	If it is not sunny and rainy: + Yolo:Bit: Send signal to servo motor to close the roof at 30 degree/s.
Exception	If the light sensor, the humidity sensor or the servo motor is not connected or broken, this automatic function is disabled.

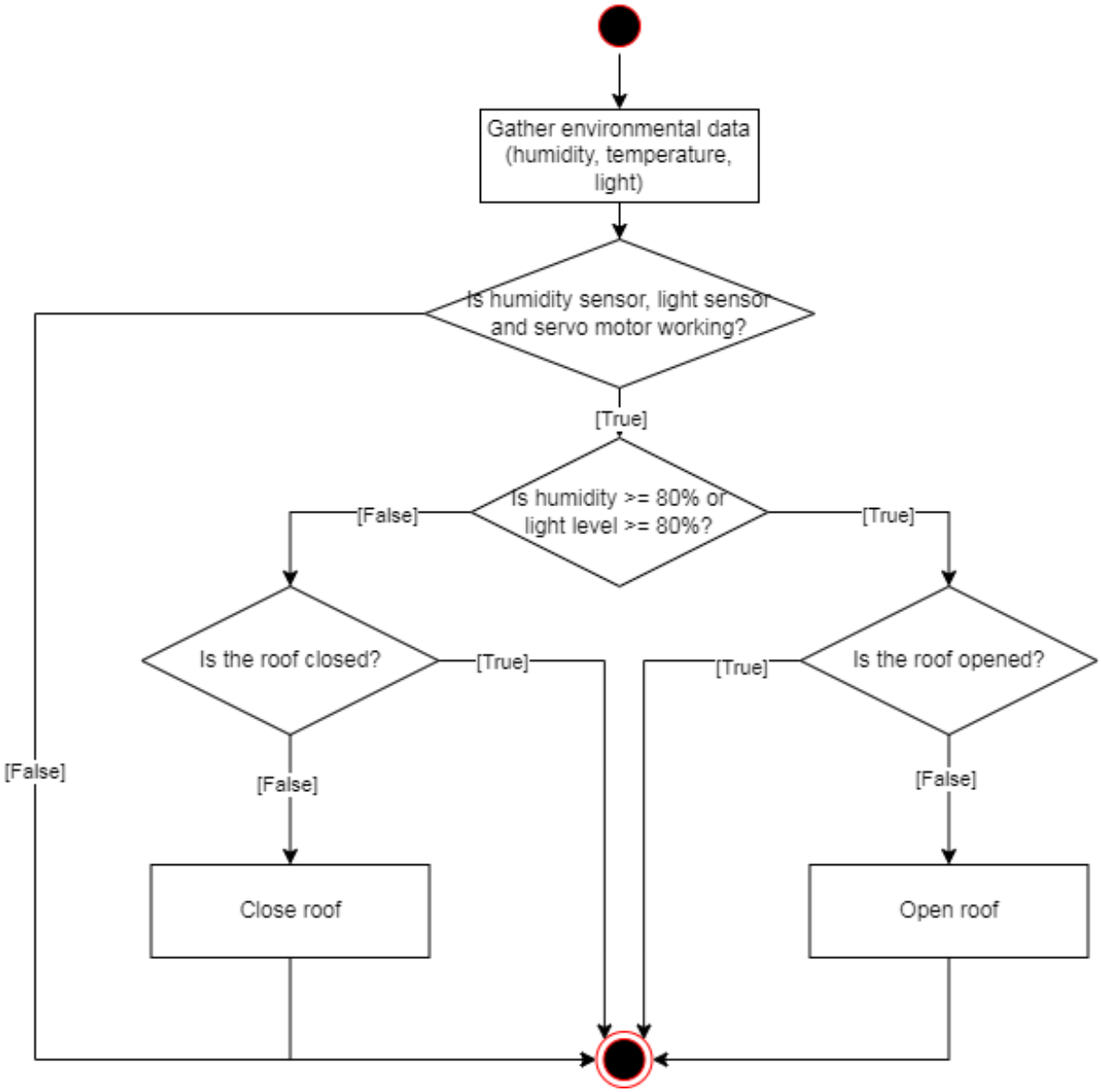


Figure 1: Activity diagram of use case "Automatically open/close roof"

4.2 Use case 2: Manually open/close roof

Use-case name	Manually open/close the roof
Summary	The user manually opens or close the roof
Actor	User
Trigger	The user presses the “Roof” slider on the phone application
Pre-condition	The roof is functioning. User is logged in.
Post-condition	Usage is recorded.
Normal flow	If the roof is currently in close state: + Web application: Send signal to open the roof to Yolo:Bit through Adafruit server. + Yolo:Bit: Receive signal and make the servo motor spin at 30 degree/s to open the roof
Alternative flow	If the roof is currently in open state: + Web application: Send signal to close the roof to Yolo:Bit through Adafruit server. + Yolo:Bit: Receive signal and make the servo motor spin at 30 degree/s to close the roof .
Exception	If the servo motor is disconnected or broken, nothing will happen

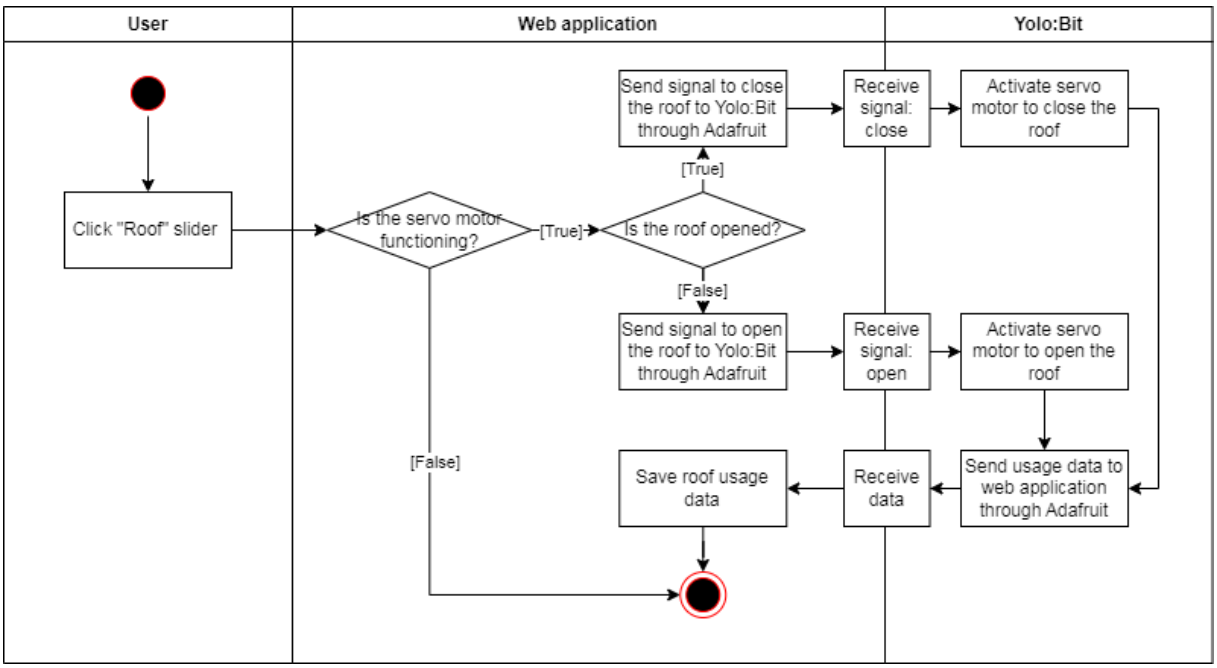


Figure 2: Activity diagram of use case "Manually open/close roof"

4.3 Use case 3: Automatically turn on/off LED light

Use-case name	Automatically turn on/off LED light
Summary	The system turn on or turn off LED light when the environment data gathered by Yolo:Bit sensors satisfy certain conditions
Actor	Yolo:Bit sensors
Trigger	None
Pre-condition	The Yolo:Bit device is plugged in and turned on.
Post-condition	None.
Normal flow	If it's dark outside (light level $\leq 20\%$): + Yolo:Bit: Send signal to LED light to turn on.
Alternative flow	If it's not dark outside (light level $> 20\%$). + Yolo:Bit: Send signal to LED light to turn off.
Exception	If the light sensor or the LED light is disconnected or broken, this automatic function is disabled and the LED light will be turned off.

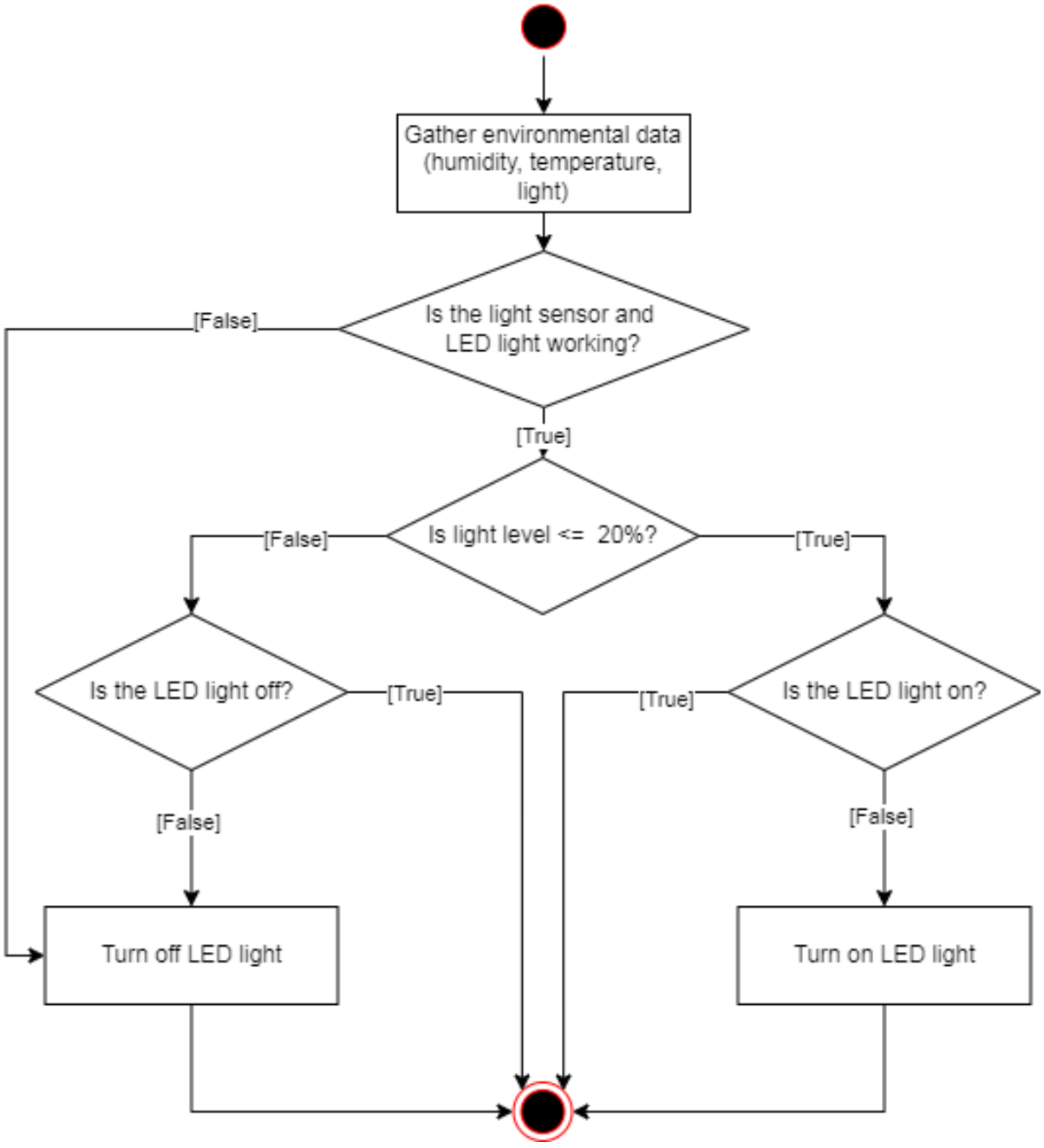


Figure 3: Activity diagram of use case "Automatically turn on/off LED light"

4.4 Use case 4: Manually turn on/off LED light

Use-case name	Manually turn on/off LED light
Summary	The user turns on or turns off the light manually
Actor	User
Trigger	The user presses the “Light” button on the phone application
Pre-condition	The LED light is functioning. User is logged in.
Post-condition	Usage is recorded.
Normal flow	If the LED light is currently off: + Web application: Send signal to turn on LED light through Adafruit server. + Yolo:Bit: Receive the signal and turn on the LED light.
Alternative flow	If the LED light is currently on: + Web application: Send signal to turn off LED light through Adafruit server. + Yolo:Bit: Receive the signal and turn off the LED light.
Exception	If the LED light is disconnected or broken, the LED light will be turned off.

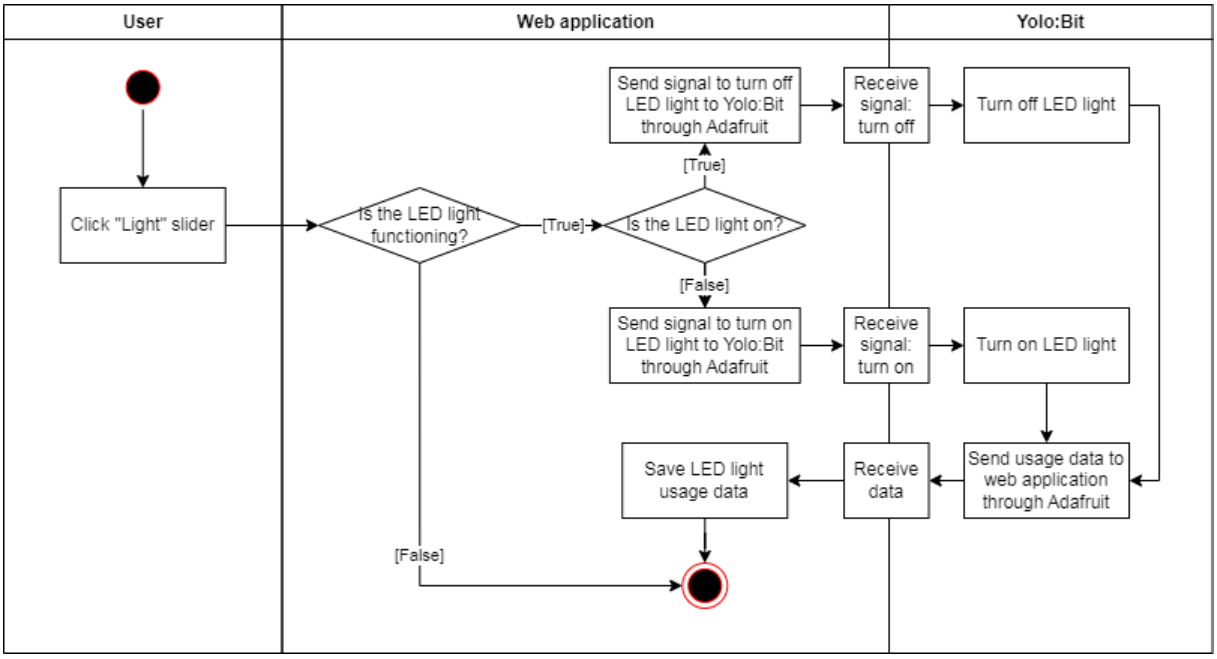


Figure 4: Activity diagram of use case "Manually turn on/off LED light"

4.5 Use case 5: View schedule

Use-case name	View schedule
Summary	The user presses the "Timetable" button on the web application to see the schedule that the system makes based on the environmental data.
Actor	User
Trigger	The user presses the "Time table" button on the phone application.
Pre-condition	User is logged in.
Post-condition	None
Normal flow	+ Web application: Loading the application page for viewing time table and send GET request to Weather API. + Weather API: Respond back to the application with weather data. + Web application: After done loading page and getting a response from Weather API, display data on the timetable.
Alternative flow	None
Exception	None

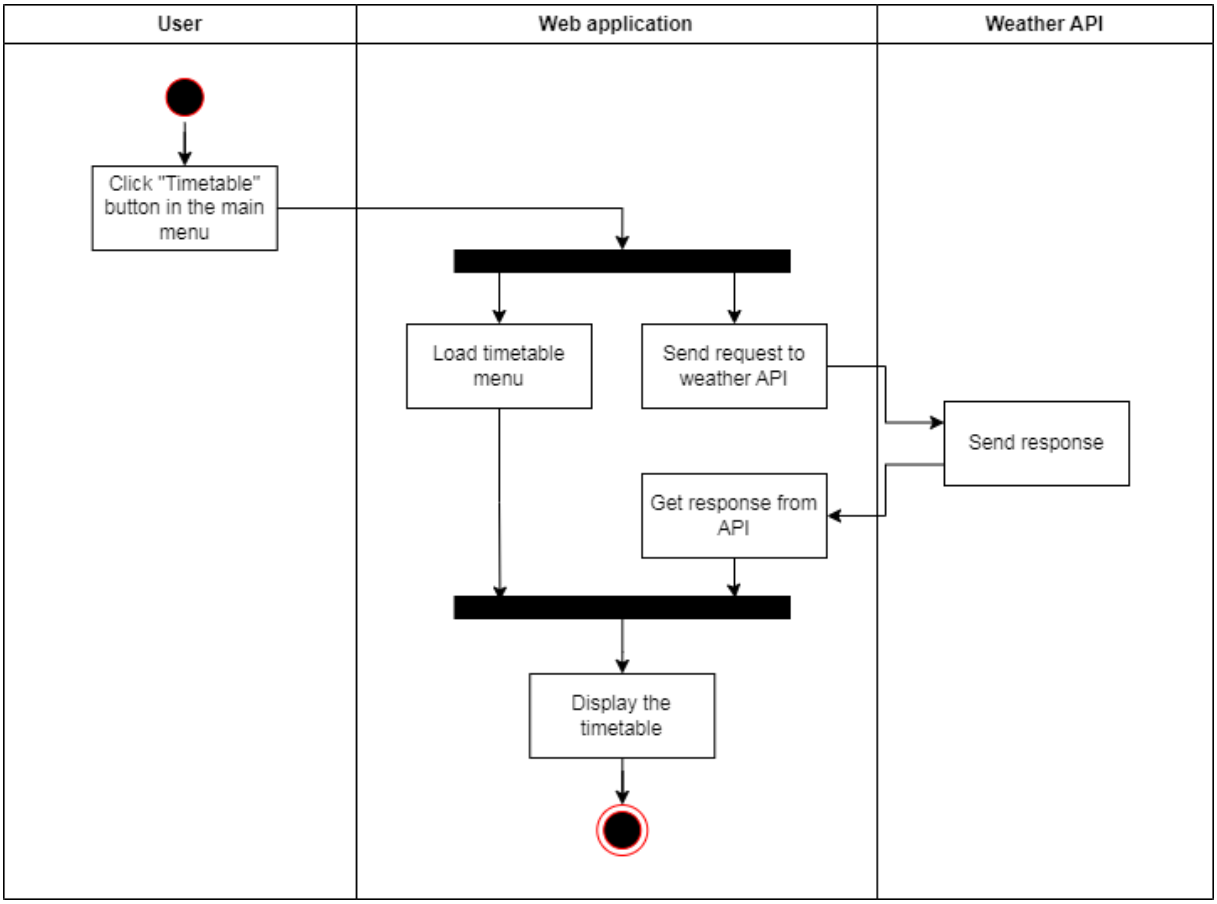


Figure 5: Activity diagram of use case "View schedule"

5 Use-case diagram

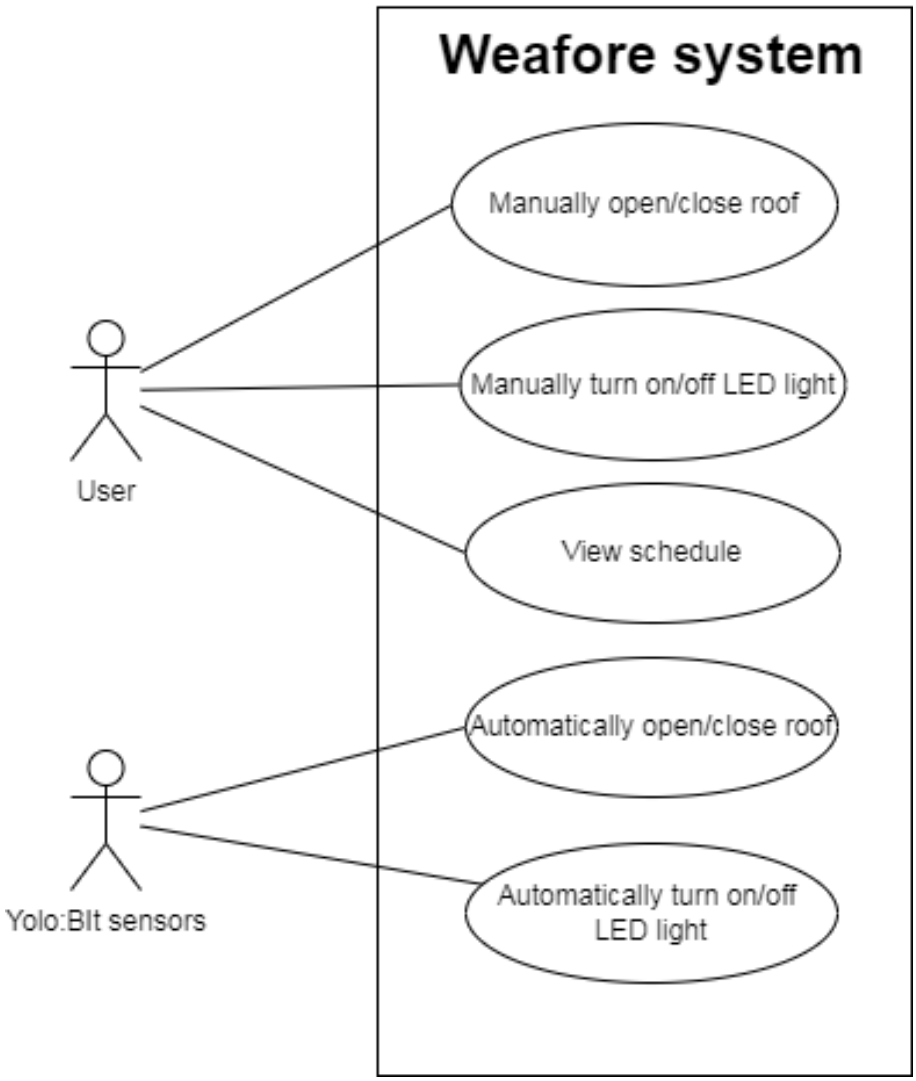


Figure 6: Weafore's use case diagram

6 UI design

Interface for the web application is designed to be simple, intuitive for any user. Most flow for the web page is single flow. For color and design, we modify and use already available template on the internet to simplify the design process. And to be able to reusing template easily, the UI will be implemented in React.

6.1 Welcome page

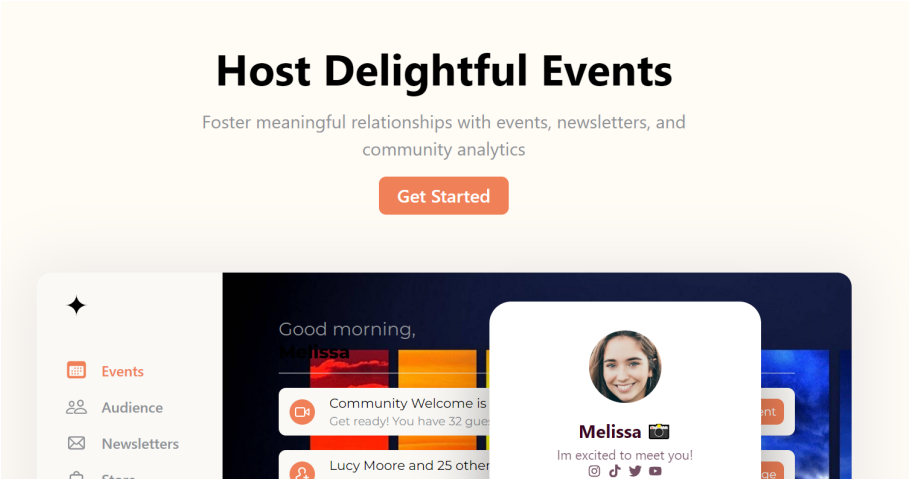


Figure 7: Welcome page screen

When open the web, user is first greeted by the screen above. To be able to use the web, user has to be authorized. Therefore, by clicking "Get Started", the web will be redirected to sign in screen.

6.2 SignIn

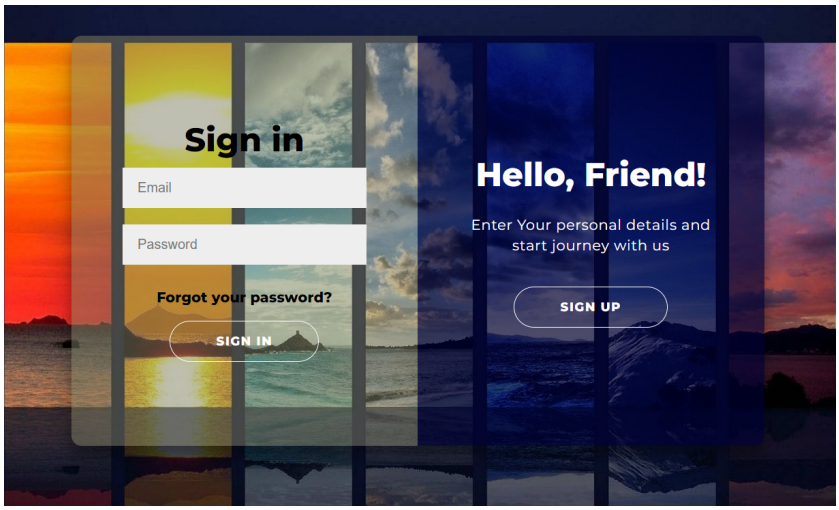


Figure 8: Sign in screen

User can either use input their email and password to sign in to their account or create a new one by clicking "SIGN UP". Both email field and password must be correct to access to the web.

6.3 SignUp

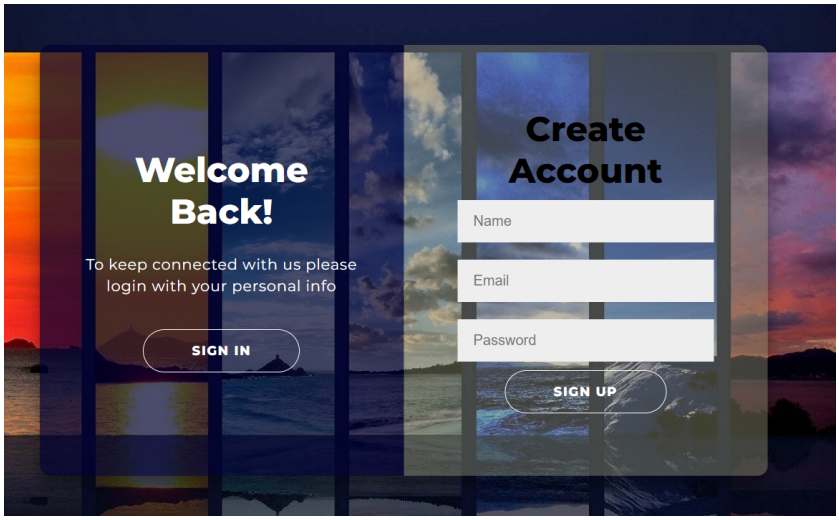


Figure 9: Sign up screen

User can either use input their username, email and password to create a new account or sign in an existing one by clicking "SIGN IN"

6.4 Homepage

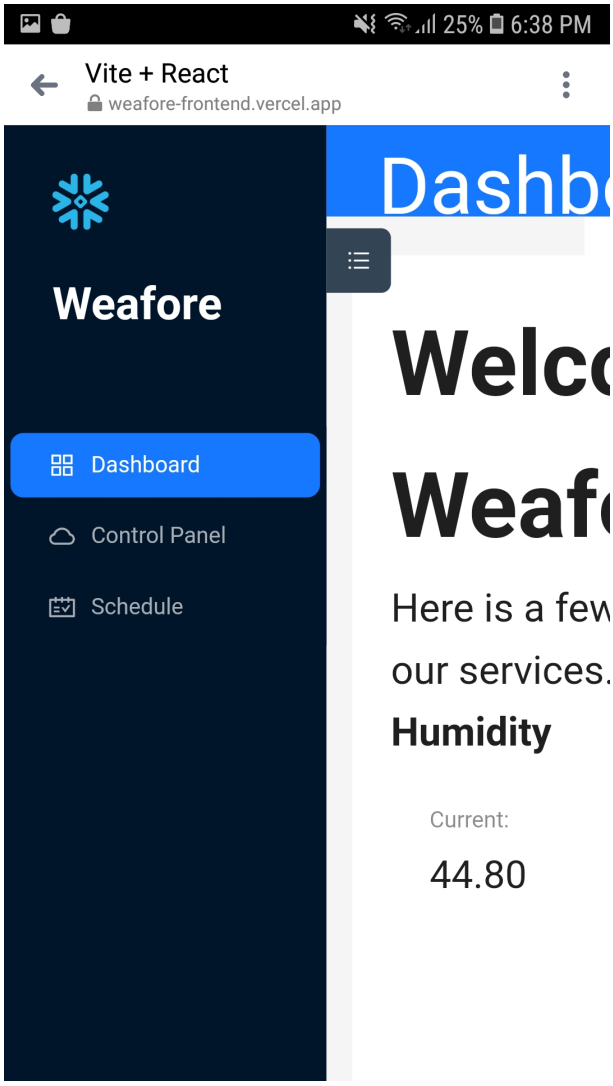
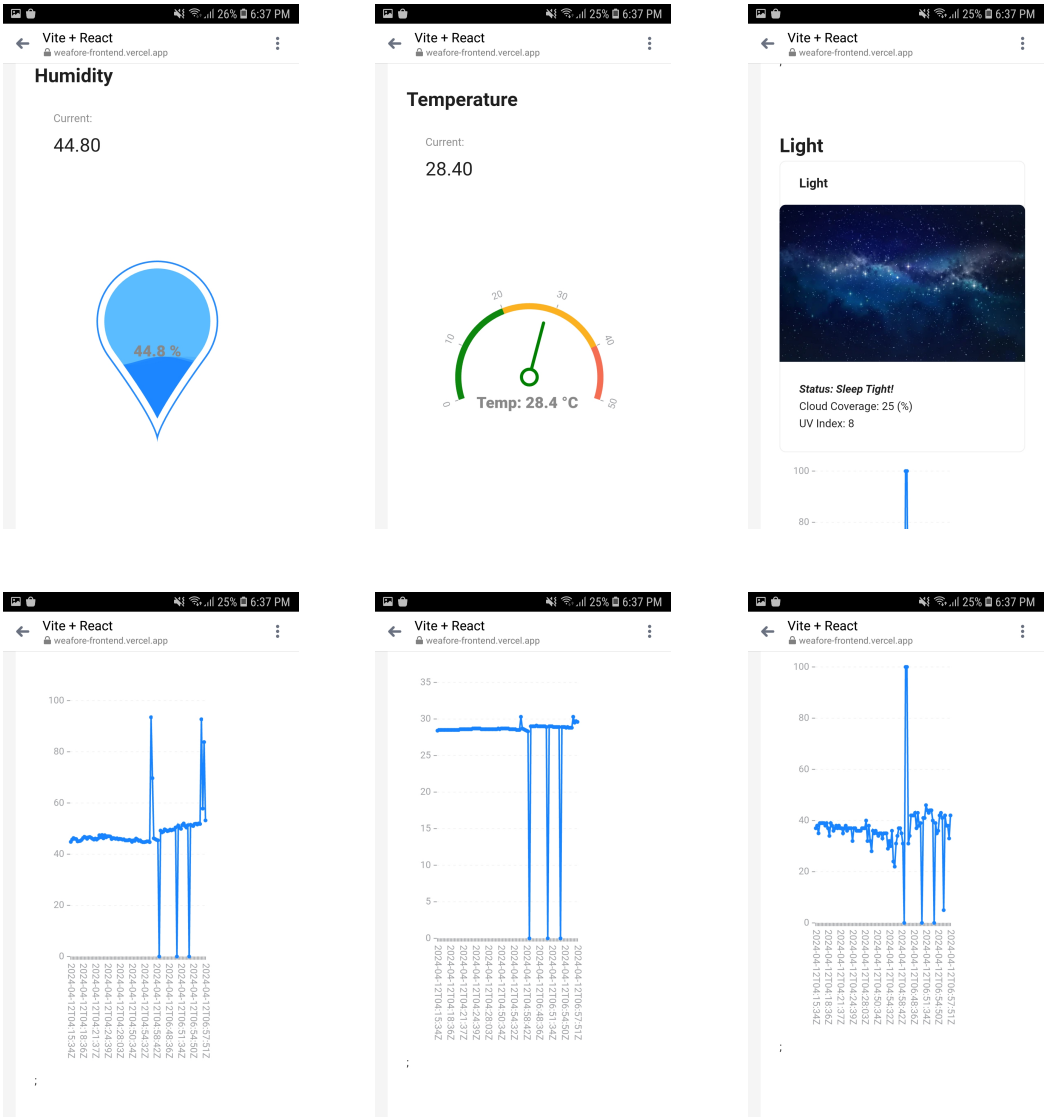


Figure 10: Sidebar

After sign in, user can choose any the services that the web provided through the sidebar. There are 3 services:

- Dashboard: Display environmental data gathered by Yolo:Bit devices
- Control panel: Allow user to control roof and LED light
- Schedule: Display weather forecast for the next 5 days and weather report in the last 5 days, including today.

6.4.1 Dashboard



(a) Humidity (b) Temperature (c) Light

Figure 12: Environmental data report

After signing in, user can view environmental data (humidity, temperature, light level) collected from Yolo:Bit device's sensors for today and many days before that with the graph.

6.4.2 Control panel

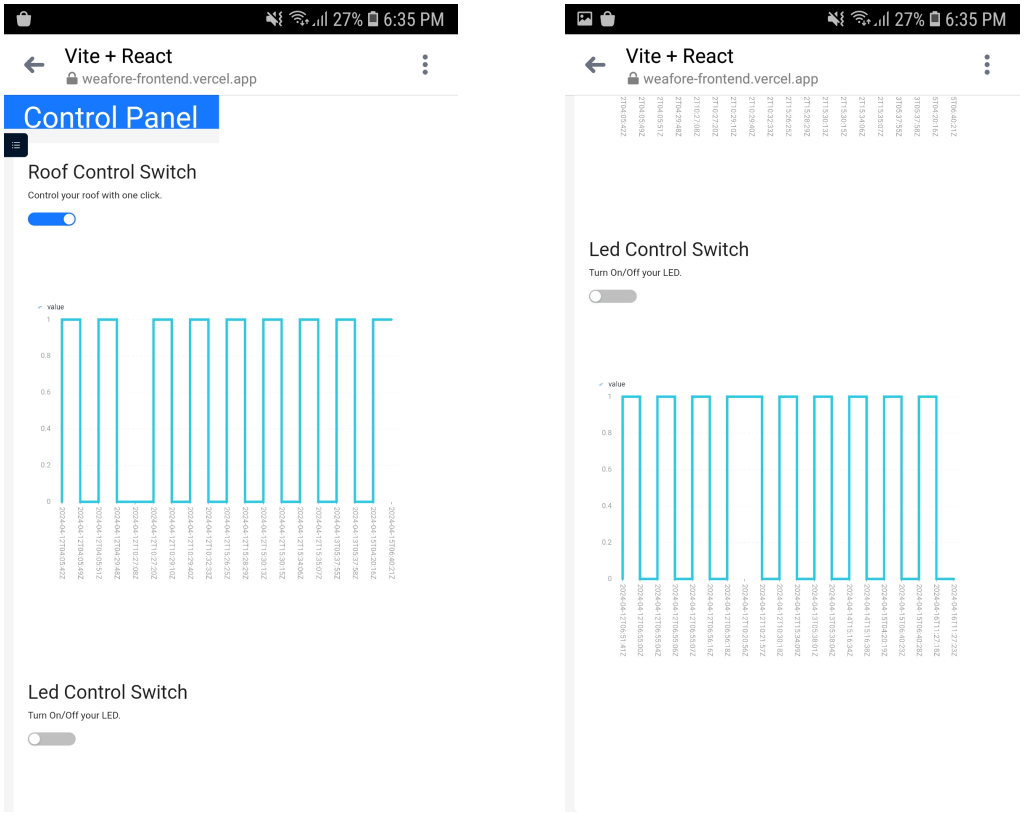
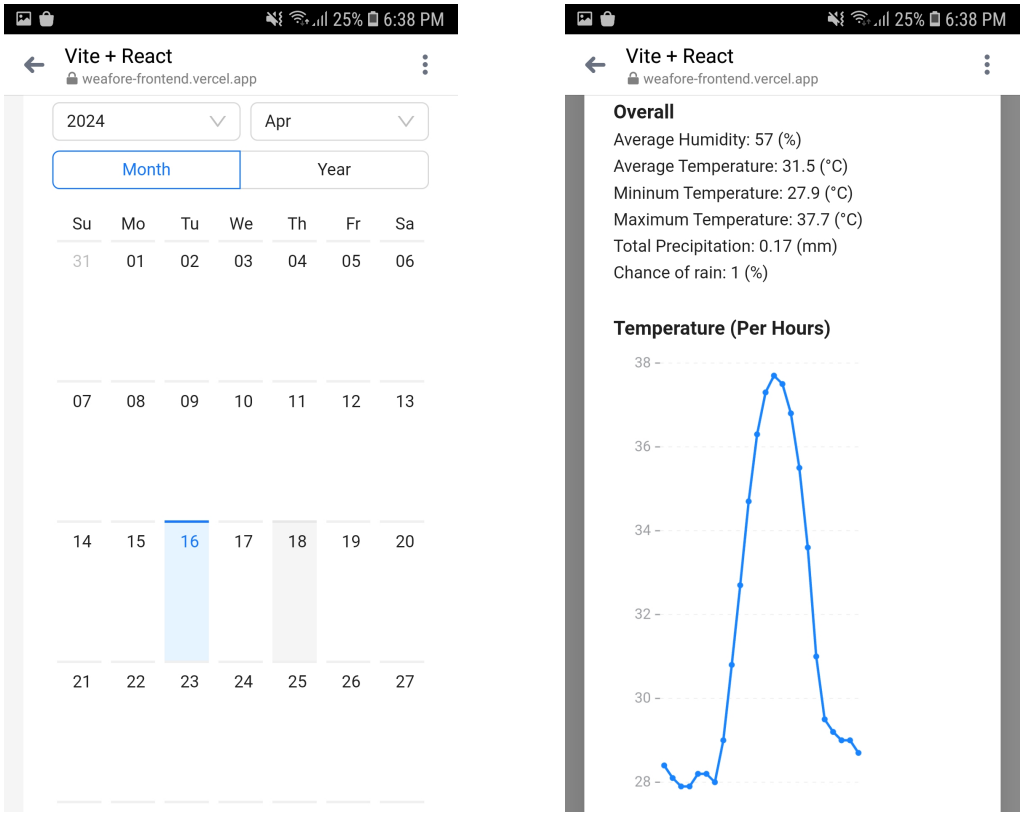


Figure 13: Control panel

User can control both the roof and LED lights by clicking on the slider. Every usage for each component is recorded and displayed separately in two graphs.

6.4.3 Schedule



(a) Schedule (b) Schedule detail

Figure 14: Viewing schedule

User can click on any day in the last 5 days or the next 5 days on the schedule to view the weather report or forecast. Detail of a day includes data about average humidity(%), average temperature(°C), minimum temperature(°C), maximum temperature(°C), total precipitation (mm) and chance of rain(%).

7 Database design(MongoDB)

The database is used for storing data about authorization. A record has the following fields: username, email, password

8 Deployment view

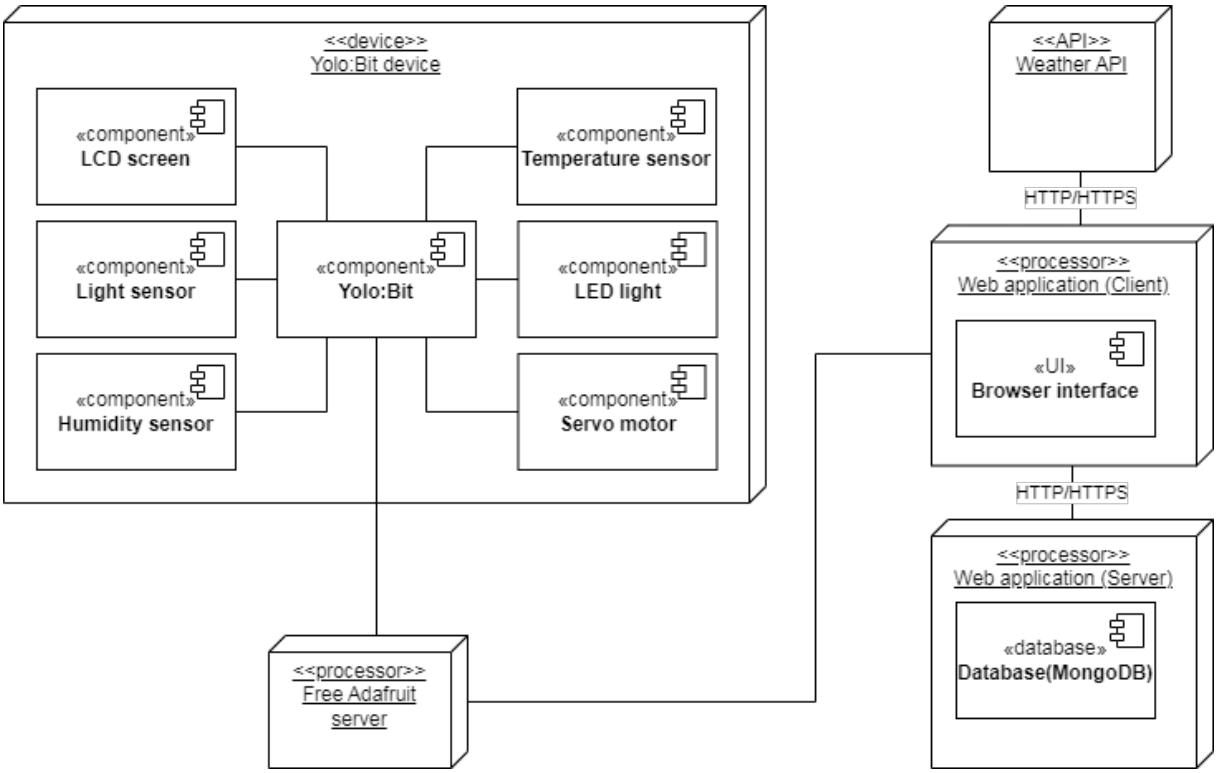


Figure 15: Weafore deployment view

9 Implementation

After planning and assembling Yolo:Bit, the device is put together in the configuration shown in Figure 16

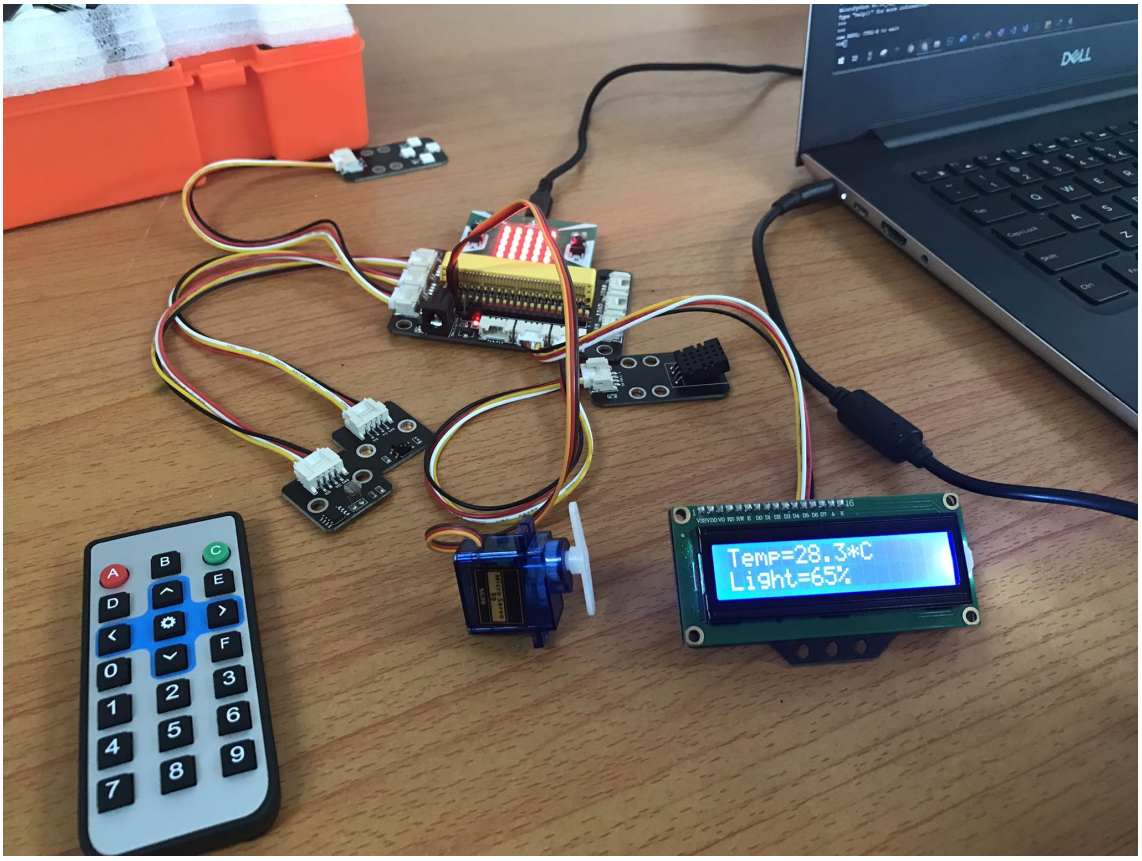


Figure 16: The Yolo:Bit device

The Yolo:Bit device should ideally be always plugged in to ensure that the data about humidity, light intensity and temperature of the surroundings is always fed to the main compartment to process and control the servo motor, light. However, as the device can only be borrowed for limited time, Yolo:bit will process based on the latest data received from the sensors. For monitoring purpose, the data is displayed on the LCD screen and on the phone application.

The automation for each house compartment is implemented on Yolo:Bit itself by programming on https://app.ohstem.vn/. As already described in the requirements, the device can observed the weather in many states as follows:

- Daytime: The light level is greater to 20%
- Nighttime: The light level is lower or equal to 20%
- Raining: The humidity is greater or equal to 80% (60% for testing)
- Sunny: The light level is greater or equal to 80% (60% for testing)

Based on the observation above, Yolo:Bit decides whether to influence the house compartment. If the weather changes to sunny or raining, the motor is activated as a representation for opening the roof to cover. Otherwise, the motor activates again to close the roof. For the LED light, it's only turned on when it's dark outside. Aside from automation, the homeowner may take manual control of both the motor and the LED light by using the phone application. Control house compartment in this manner makes the phone application send signal to Yolo:Bit through Adafruit server to let it knows when to change the state of the house compartments according to the user's will.

On the other hand, regarding the web application, the UI is implemented as discussed in "UI design" section above. Upon deploying, the web is usable on any platform including mobile. For security reason, user has to sign in by inputting email and password in "Sign in" screen. The information is sent to the database to verify using POST request. Database then responses back with POST request after verification in two cases. If it's matched, the user is authorized to use the web application. Otherwise, the user is required to input their email and password again. Additionally, user can also sign up to the service. By inputting username, email and password on the "Sign up" screen, the application will create a new entry in the database, allow user to access the web at any time in the future. Data displayed on the "Dashboard" is updated constantly if the program keeps receiving environmental data from Yolo:Bit via the Free Adafruit server.

For weather forecasting, the web system uses an external weather forecast API, "Weather API". Whenever the user view the schedule, the system make a GET request to Weather API to forecast weather and display it on screen. Data of the weather report and forecast is only available from the last 5 days to the next 5 days.

10 Test case demonstration

Here, we will demonstrate our device and all of its component in the following testcase. All demo is put in the Google Drive file at: https://drive.google.com/drive/folders/1JwBBpEEYb70hUtATq9u1Ir69V
usp=drive_link

10.1 Module 1: Gather environment data

- Testcase: The Yolo:Bit sensors collect environmental data and show it on the LCD screen.
  Result: https://drive.google.com/file/d/1my6glUw4THbEaKbmdeP60PHNvwWZpEWL/view?usp=drive_link
  https://drive.google.com/file/d/1MndrtNQURMc35qQvhC0fspDcXtEc13rr/view?usp=drive_link
- Testcase: The Yolo:Bit sensors collect environmental data and show it on the web.
  Result: https://drive.google.com/file/d/1jpo5l8pfpku2Iw0ed_IX3ldpTM-8uHaC/view?usp=drive_link
- Testcase: The environmental data is updated on the web in less than 3 seconds
  Result: https://drive.google.com/file/d/1jpo5l8pfpku2Iw0ed_IX3ldpTM-8uHaC/view?usp=drive_link

10.2 Module 2: Automatically open/close roof

- Testcase: The system automatially activates the servo motor to open the roof when the weathe changes light level $\geq 60\%$ or humidity $\geq 60\%$

Result: https://drive.google.com/file/d/1GokN3Tt4cOv_JGUdzZl2hjHlvWAFWdam/view?usp=drive_link

- Testcase: The system automatially activates the servo motor to close the roof when the weather changes to light level $< 60\%$ and humidity $< 60\%$
  Result: https://drive.google.com/file/d/1mb3t0bBssFoevzl_Vscr7IXzPFLeUV2P/view?usp=drive_link

*Note: For demonstration, we use 60% instead of 80% as described in the requirement due to difficulty in replicating the condition.

- Testcase: The servo motor turns at 30 degree/s when activated.
  Result: https://drive.google.com/file/d/1mb3t0bBssFoevzl_Vscr7IXzPFLeUV2P/view?usp=drive_link

10.3 Module 3: Automatically turn on/off LED light

- Testcase: The system automatically turn on LED light when the light level $\leq 20\%$
  Result: https://drive.google.com/file/d/1XMIRVkHTrw4pjn7GrhoOkKJyK2AdwShU/view?usp=drive_link
- Testcase: The system automatically turn off LED light when the light level $> 20\%$
  Result: https://drive.google.com/file/d/1XMIRVkHTrw4pjn7GrhoOkKJyK2AdwShU/view?usp=drive_link

10.4 Module 4: Manually open/close roof

- Testcase: User can manually control the servo motor by using the phone application.
  Result:
  Open roof: https://drive.google.com/file/d/1SLax4Zv7SSGG9fMKgwm4t4iDXArfX0Vy/view?usp=drive_link
  Close roof: https://drive.google.com/file/d/1JlUjsM-hbiBk5wTPLuFFmLYLoPiUmHPf/view?usp=drive_link
- Testcase: The servo motor activates in less than 3 seconds counting from the moment the user click the "Roof" slider on the web.
  Result: https://drive.google.com/file/d/1SLax4Zv7SSGG9fMKgwm4t4iDXArfX0Vy/view?usp=drive_link

10.5 Module 5: Manually turn on/off LED light

- Testcase: User can manually control the LED light by using the phone application.
  Result: https://drive.google.com/file/d/1cUEjnaSuuz4B9Klr8nuVflC--wFxHMHb/view?usp=drive_link
- Testcase: The LED light changes state in less than 3 seconds counting from the moment the user click the "Light" slider on the web.
  Result: https://drive.google.com/file/d/1cUEjnaSuuz4B9Klr8nuVflC--wFxHMHb/view?usp=drive_link

10.6 Module 6: View schedule

- Testcase: Sign in (Authorization)
  Result: https://drive.google.com/file/d/1K5Phx_y4E5m-WALDejiTDfRr18O7bA40/view?usp=drive_link
- Testcase: Sign up (Authorization)
  Result: https://drive.google.com/file/d/1K5Phx_y4E5m-WALDejiTDfRr18O7bA40/view?usp=drive_link
- Testcase: Viewing schedule from the last 5 days to the next 5 days.
  Result: https://drive.google.com/file/d/1UAnXyxqR_qhsMofdIYjhIEwhojqFY4ze/view?usp=drive_link