Institute of Technology of Cambodia

Departement of Telecommunications and Network Engineering

Report Internship

Topic:

"IoT Platform Cloud Storage for Agricultural Application"

Student : **VANNAK Sovannroth**

ID : **e20171016**

Group : **I4-GTR**

Advisor : **HEL Chanthan**

ACADEMIC YEAR

2020-2021

ACKNOWLEDGEMENTS

First and foremost, I would like to thank my parents for their tremendous amount of support, encouragement, I would not have made it this far without their sacrifices. I want to express my acknowledgment to:

H.E. Dr. OM Romny, Director of the ITC, for his good management in facilitating the mechanism of this higher education establishment.

Dr. SRENG Sokchenda, Head of Telecommunications and Network Department (GTR), for his encouragement, instruction and precious pieces of advice to me.

Dr. THOURN Kosorl, Co-Head of Telecommunications and Network Department (GTR) and all the teachers for their teaching, constructive recommendation, explanations, advises, and their professionalism in teaching.

Mr. HEL Chanthan, my supervisor, who has been a tremendous mentor and spent his priceless time for me. I would like to thank him for his good explanation, suggestion, encouragement and allowing me to completed my project. Without his advices and guidance, this achievement would not have been possible.

I would like to express my deep gratitude to all the lecturers and staffs of Institute of Technology of Cambodia that I have studied from the first to the third year, for their professionally teaching and valuable advices guiding me to learn and practice my knowledge.

ABSTRACT

Currently, technology has become one of the most important elements in today's societies. From industry, agriculture to daily life data management is used to manage data and make it possible for end-users to view, searching, and modify the data. When talking about the Internet of Things (IoT), it refers to the billions of physical devices around the world that are now connected to the internet, all collecting and sharing data. This project is designed and implemented to assist in the study and create data management systems. Therefore, Progressive Web Application is very important in SPAF because it provides users and farmers to monitor the state of crops production, manage, and control action when needed in the farmland. This project aims to the realization of Progressive Web Application that responsive, compatible with both mobile and computer and communicates with IoT cloud. The Progressive Web Application should be able to add projects and assigned sensors node that the project needs. It can display information is the environmental sensor such as temperature, humidity, soil moisture, solar radiation, webcam, and pressure.

TABLE OF CONTENTS

LIST OF FIGURES

1. INTRODUCTION

1.1. Presentation of Internship

Due to Corona virus or Covid 19, the end of third year have delayed nearly one month. So, on my vacation, I have only 2 to nearly 3 months for internship. Because of this problem and not enough time, most companies don't want to get internees to internship, I decided to have an internship at school (ITC) with lecturer HEL Chanthan. My first day start from 06 August to 15 October 2020. This internship based on researching project, it focuses on IoT platform cloud storage for agricultural application that it is a part of smart irrigation project, it will be designed for managing data of projects, user, device information and data collected from Sensor Node in the farm. All data are stored in database and will be monitored and managed on the frontend. On "Figure 1.1", diagram illustrates the system overview of system architecture of smart irrigation.

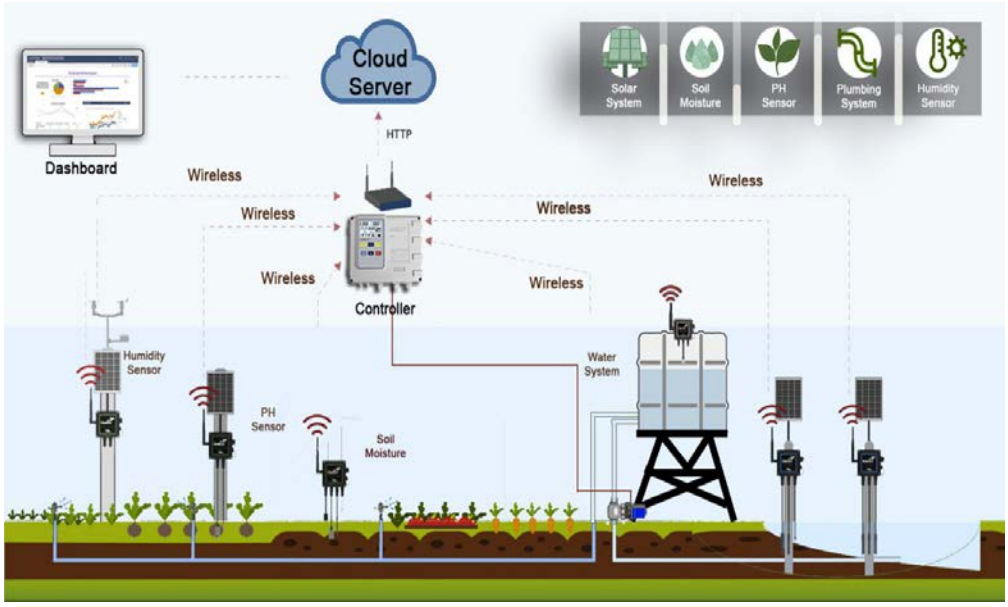


Figure 1.1: System Architecture of Smart Irrigation

I had been developing web/mobile application that it is a IoT analytics platform service that allows us to aggregate, visualize and analyze live data streams in the cloud, it provides instant visualizations of data posted by devices. It will view data value of temperature, soil moisture, humidity, PH, and solar radiation on dashboard as line charts and it will view videos record of plant growth from webcam in greenhouse on frontend and backend (web/mobile application). The web application was developed by using Laravel 7 framework (PHP language) with Bootstrap 4 and storage data on database using MySQL. In this web application has backend and frontend, backend uses for user login/logout, view data on dashboard as line charts, and edit information and frontend

uses only for viewing data from database. On dashboard will view data value of temperature, soil moisture, humidity, and solar radiation. Frontend and backend will view videos record of plant growth from webcam in Greenhouse. For getting data from sensor nodes to database, it must use API with NodeMCU/ESP8266 for connect them. “Fig. 1.2”, illustrates the proposed smart farm that it will overview communication between web application, mobile application, and microcontroller.

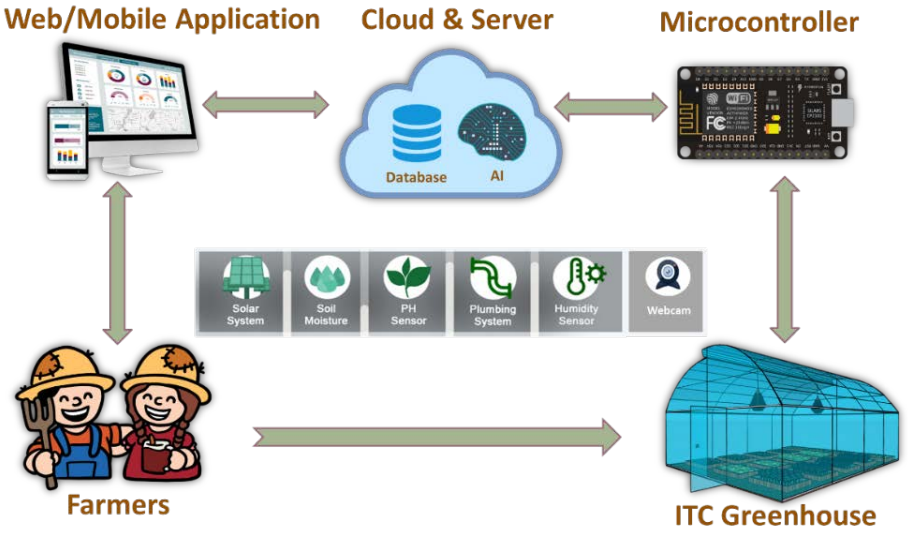


Figure 1.2: The Proposed Smart Farm

Due to this project is the type of a long time researching, after finish internship, I will still continue to research it. I will continue to develop mobile application using Flutter framework that it can develop both Android and iOS application for viewing line charts data and control irrigation on smart phone and using API connect it to web application with the same database.

1.2. Objective of Internship

The farmers will be able maintain, monitor, and control real-time environment information of the smart farm via mobile/web application remotely or can be locally either. This will ensure an environment conducive to the growth of plants at any time, from anywhere, through the Internet. On “Figure 1.3” illustrates viewing of live data on dashboard.

Figure 1.3: viewing of live data on dashboard

Famers will able use IoT devices to measure temperature, humidity, moisture, webcam, and water pH values in the greenhouse. Those captured environmental parameters can be displayed on mobile/web application through the internet cloud service. On “Figure 1.4” illustrates IoT device is used in agriculture application to control, monitor, and maintain smart farm.

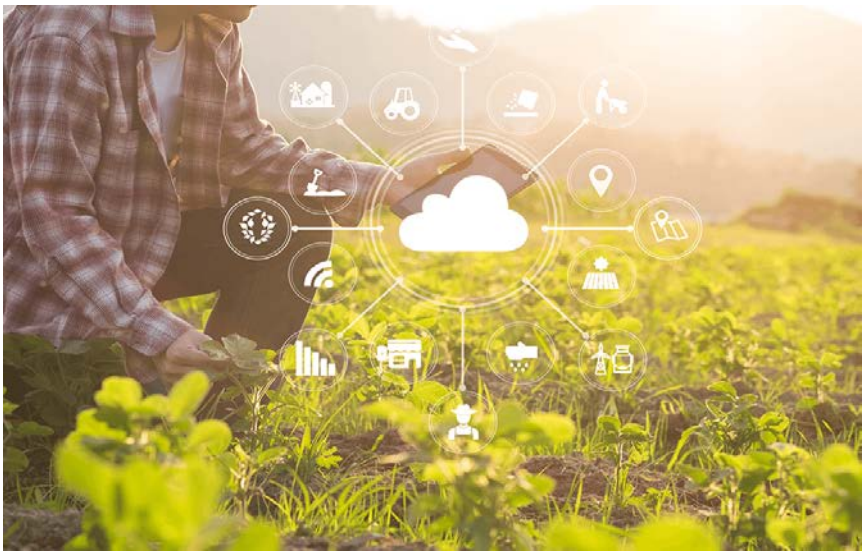


Figure 1.4: IoT device is used in Agriculture Application

Farmers will able to manually or automatically control the watering pump, and/or airflow fan to create suitable environment for growing the plants in the greenhouse. The real-time environmental parameters are able to be displayed and controlled off-line or on-line. On “Figure 1.5”, illustrates famers can remote control sensors in greenhouse, over the Internet.

Figure 1.5: Remote Control Sensors

1.3. Scope of Study

- Web Application: The web application was developed by using Laravel 7 framework (using PHP language) with Bootstrap 4, and to develop a web application the code is written in HTML, CSS, and JS. The main goal in the web application is to communicate with the server, be able to send and collect data from database.
- Server: The main purpose of the server is to connect to sensor nodes, irrigation controller, web/mobile application by using API, and store data received from API to the database. The server be able to send information back to the web application or users that collected from the database and the users have the ability to send or control the state of sensor nodes.
- Database: The database should contain the values collected from API. The database should be made as simple as possible, using relations that are a common technique in database connections. Relations are the connections between the collections to another collection in the database. SQL is a standard relational database language used to communicate with DBMS, and stands for Structured Query Language.
- Mobile Application: The mobile application has been developing by using Flutter framework with Dart language that it can develop both iOS and Android.

- API: API is used for communication between web application, mobile application, database, and sensor nodes that uses HTTP requests to access and use data. That data can be used to GET, PUT, POST and DELETE data types, which refers to the reading, updating, creating and deleting of operations concerning resources.
    - GET to retrieve a resource
    - PUT to change the state of or update a resource, which can be an object, file or block
    - POST to create that resource
    - DELETE to remove it

1.4. Weekly Activities

Week	Activities
1	• Installation Nodejs & MongoDB (Shell and Compass) • Performing CRUD on Shell and Compass • Testing connection locally/cloud on MongoDB
2	• Starting to build sample backend • Transmitting and fetching data to/from the Database • Creating Login and Signup (Email and password)
3	• Learning HTML, CSS, and JS language • Learning SQL and PHP language • Learning Bootstrap 4
4	• Learning Laravel 7 • Install wampserver & Run Laravel 7 with MySQL (Database) • Choose free template Bootstrap 4 for Frontend & Backend
5	• Installation Laravel Module for Backend and Frontend • Starting to build module backend for project. • Creating users Login, Signup, & reset password
6	• Create CRUD for Users login • Create CRUD for Lab information • Create CRUD for upload file (picture)
7	• Starting design dashboard • Apply sample dashboard from bootstrap 4 • Learning how to use platform bootstrap 4
8	• Testing sample line charts on dashboard • Getting data from database view on dashboard
9	• Starting to build sample frontend • Connect frontend to backend • Choose Flutter or React Native for build Mobile application.
10	• Installation Flutter & Android Studio and coding • Start build sample UI on project • Starting build line charts on Mobile application

Figure 1.6: Table of Weekly Activities

1.5. Outline of project

In this report there are 4 chapters:

- Chapter 1: Introduction of project and internship, Objective of work
- Chapter 2: Methodology or Materials and Methods
- Chapter 3: Result and Discussion
- Chapter 4: Conclusions

2. METHODOLOGY AND METHODS

IoT platform cloud storage is a part of smart irrigation project, it will be designed for managing data of projects, user, device information and data collected from Sensor Node in the farm. All data are stored in database and will be monitored and managed on the front-end. On “Figure 2.1” show main of architecture of IoT cloud storage, there is two parts of development in system that the project is developed on the backend and frontend:

- Backend is developed for communication to frontend, data collections, and manages the data in the database that it also allows users to access information from the server.
- Frontend is developed to monitor the data of each project/farm through web application that is compatible on both mobile and computer and plot the information from the database.

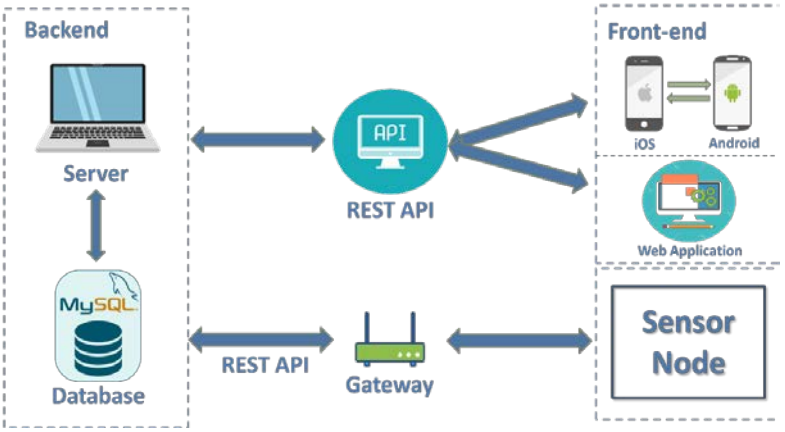


Figure 2.1: Architecture of IoT Cloud Storage

2.1. Backend Development

Backend Development refers to the server-side development. It is the term used for the behind-the-scenes activities that happen when performing any action on a website. It can be logging in to your account or purchasing a watch from an online store. Backend developer focuses on databases, scripting, and the architecture of websites. Code written by backend developers helps to communicate the database information to the browser.

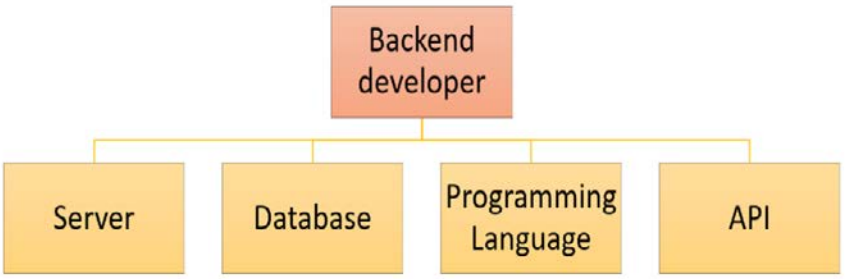


Figure 2.2: Architecture of Backend Development

- SQL Server: is a relational database management system (RDBMS) developed by Microsoft. It is primarily designed and developed to compete with MySQL and Oracle database.
- Database: is a collection of objects such as tables, views, stored procedures, triggers, functions, etc. The project uses Query and SQL language as database.
- Programming Language: using PHP language as main programming language.
- Rest API: allows frontend to do CRUD (CREATE/READ/UPLOAD/DELECT) operations on the data information and API carry out the info through HTTP methods.

2.2. Frontend Development

Frontend development focuses on develop web and mobile application, the project will develop dashboard as line charts on web and mobile application for viewing data collection from database, and it also allow users able to create projects, provides data from the database to plot graphs, allow the user to get or update the information of the projects, and extract the data for further use.

The first step of development started with the search on how to retrieve and transfer information from the application into the backend or server. We choose JSON as the data format because it allows the developer easy both backend and frontend.

The second step of development was to send the information using a REST API. REST is a standards-based architecture that uses HTTP protocol and it allows the developer easy to communicate between application and server.

The final step of development is to add Progressive Web Application features to our frontend web application which allows the users to use the web application similar to the mobile app.

In this section, we will discuss the concept of designing the web application and the Progressive Web App features. To make this project easy for explanation, we decided to

divide into two parts:

2.2.1. Web Application Design

In Web Application Design, we going to discuss the Service for the backend to connect to front-end, essential components, and implementation of data plot for the graph, and we will develop the service that it has multiple service methods, we use the service methods to manipulate CRUD operations to the REST API through HTTP methods. In our case, we run our server locally on 4000 then use the service methods to communicate the port using the local port URI and HTTP methods. Then after the server is connected to the database, we export the URI to the service as URI = http://localhost or HOST = 127.0.0.1.

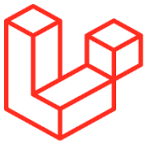
- **Software and platform**

  The Software has used in this project:

Google Chrome is a cross-platform web browser developed by Google. It was first released for Microsoft Windows, and was later ported to Linux, macOS, iOS, and Android where it is the default browser built into the OS. The browser is also the main component of Chrome OS, where it serves as the platform for web applications.

Visual Studio Code is a source-code editor it has features for debugging support, syntax highlighting, intelligent code completion, snippets, code refactoring and embedded Git. The program also customizable and user can install extensions that add additional functionality.

WampServer refers to a solution stack for the Microsoft Windows operating system, created by Romain Bourdon and consisting of the Apache web server, OpenSSL for SSL support, MySQL database and PHP programming language.



Laravel is a free open-source PHP web framework, some of the features of Laravel are a modular packaging system with a dedicated dependency manager, different ways for accessing relational databases, utilities that aid in application deployment and maintenance, and its orientation toward syntactic sugar.

- **Programming Language**

PHP is a general-purpose scripting language especially suited to web, PHP originally stood for Personal Home Page, but it now stands for the recursive initialism PHP.

Bootstrap is a powerful toolkit - a collection of HTML, CSS, and JavaScript tools for creating and building web pages and web applications. It is a free and open-source project, hosted on GitHub, and originally created by Twitter.

2.2.2. Android and iOS Application Design

In Android and iOS Application Design, we will develop Progressive Mobile Application features and connect Rest API to Web application and database, next step we will develop multifunction of controller on mobile application to control, monitor, and maintain real-time environment information of the smart farm remotely or can be locally either. This will ensure an environment conducive to the growth of plants at any time, from anywhere, through the Internet. To measure temperature, humidity, moisture, webcam, and water pH values in the greenhouse. Those captured environmental parameters can be displayed on mobile/web application through the internet cloud service. To manually or automatically control the watering pump, and/or airflow fan to create suitable environment for growing the plants in the greenhouse. The real-time environmental parameters are able to be displayed and controlled off-line or on-line.

- **Software and platform**

The Software has used in this project:

Android Studio is the official integrated development environment (IDE) for Google's Android operating system, designed specifically for Android development. It is a replacement for the Eclipse Android Development Tools as the primary IDE for native Android application development.

Flutter is an SDK that makes building high-performing, modern and beautiful apps easy works for both Android and iOS, is an open-source toolkit, developed by Google.

Firebase is a platform developed by Google for creating mobile and web applications.

Xcode is Apple's integrated development environment (IDE) for macOS, used to develop software for macOS, iOS, iPadOS, watchOS, and tvOS.

- **Programming Language**

  The programming language has used in this project:

Dart is an open-source, scalable programming language, with robust libraries and runtimes, for building web, server, and mobile apps.

A RESTful API is an architectural style for an application program interface (API) that uses HTTP requests to access and use data. That data can be used to GET, PUT, POST and DELETE data types, which refers to the reading, updating, creating and deleting of operations concerning resources.

3. RESULTS AND DISCUSSION

3.1. Backend Development

In backend development, we have developed authentication as Users Login/Logout & Register access into MySQL database. On “Figure 3.1”, illustrates User Login by using Email & Password in Authentication.

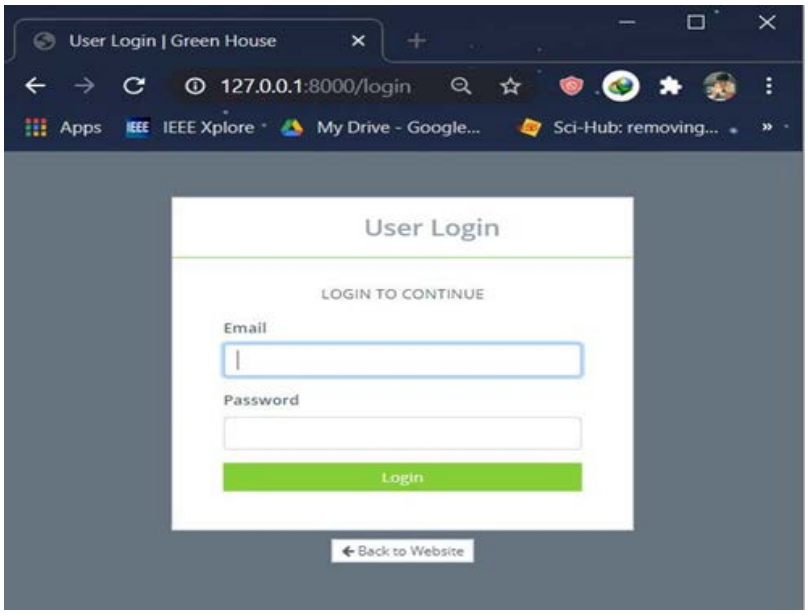


Figure 3.1: Email & Password in Authentication

We have developed sample User Interface (UI) on backend such as Users info, Dashboard, ITC-charts (Node1 & Node2), Teams info, and Laboratory info. On Users info, we can create, update, read, and delete information of users. On ITC-charts, we have developed line charts that it will show all data of temperature, humidity, soil moisture, and solar radiation from database. On “Figure 3.2”, illustrates interface of Users information and on “Figure 3.3”, illustrates viewing data on line charts of ITC-charts.

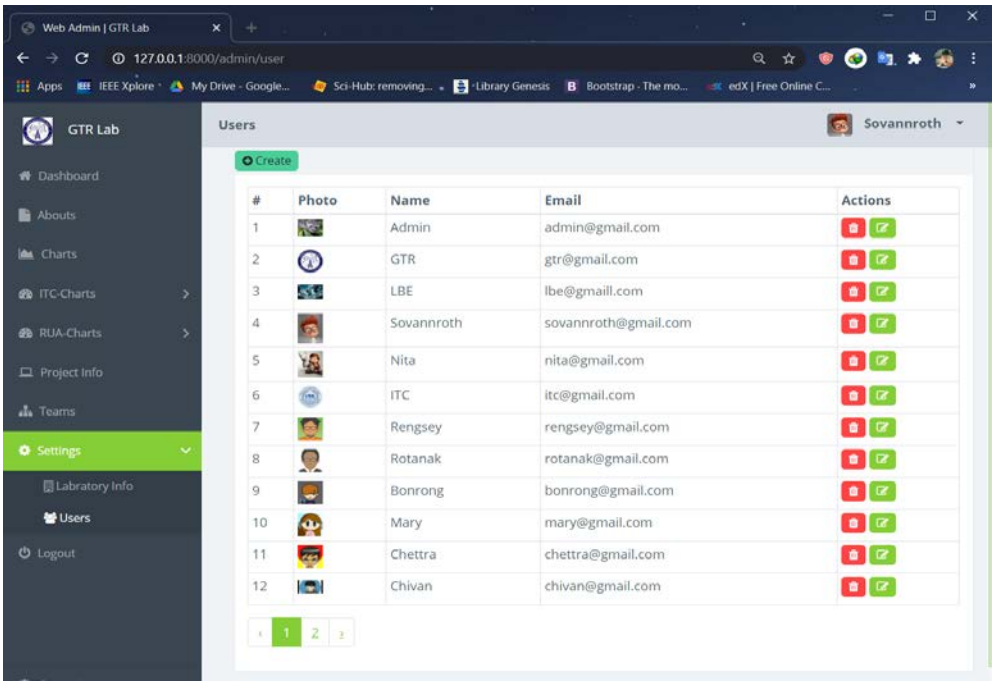


Figure 3.2: Interface of Users information

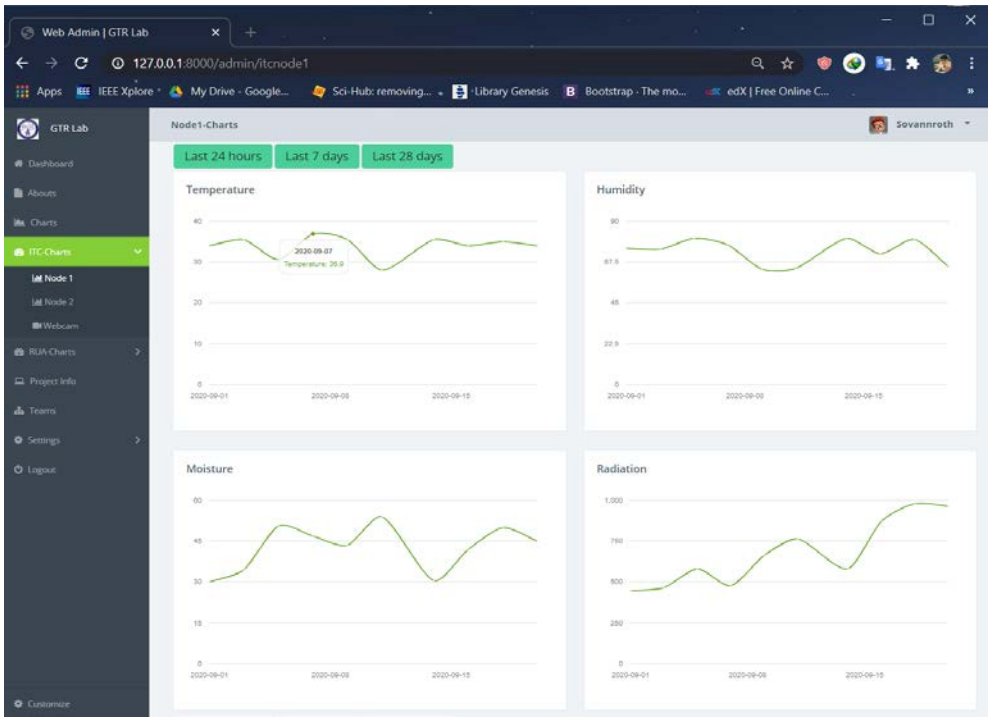


Figure 3.3: Viewing data on line charts of ITC-charts

In this implementation of the backend, the main goal is to focus on communication with the frontend by using REST API. Express routes are the module for PHP. The implementation is on retrieving and transfer data through REST API. We have achieved using Schema to store in the database, receive data from Sensor Node, fully responsive to the data and request from the frontend.

3.2. Frontend Development

3.2.1. Web Application

In frontend web development, we developed web page by following backend but in frontend we cannot create, update, read, and delete information of UI that just only can view data on line charts and information of UI. on “Figure 3.4”, illustrates viewing Team info on frontend.

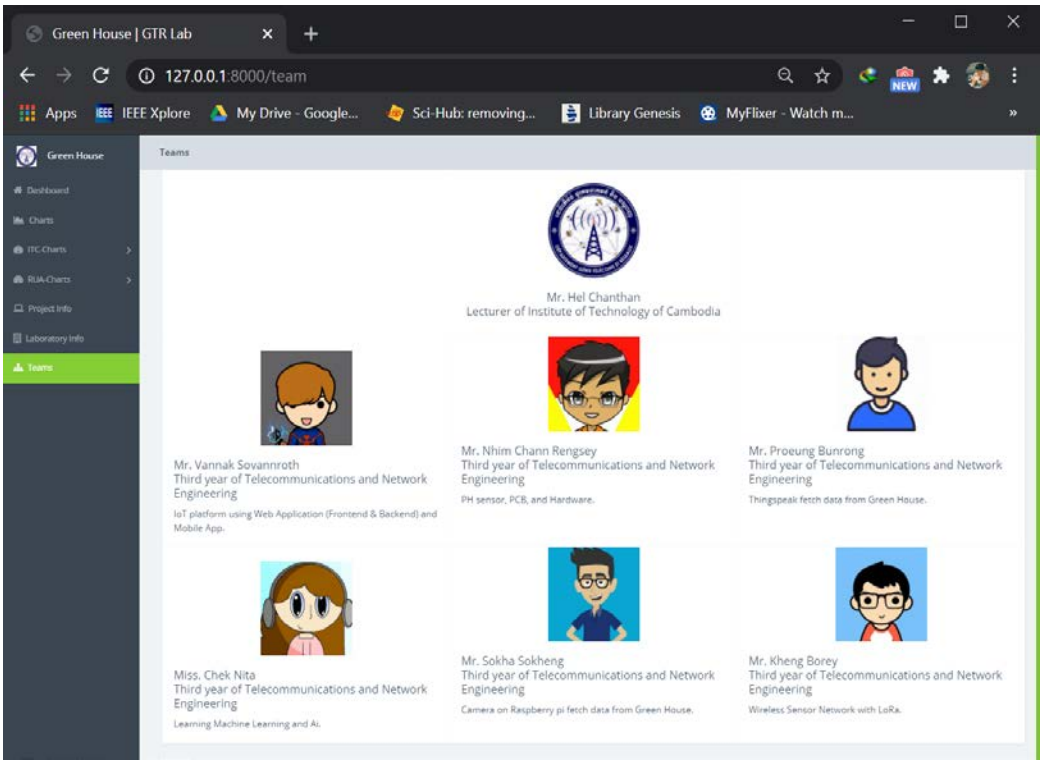


Figure 3.4: Viewing Team info on frontend

3.2.2. Mobile Application

On mobile application, we have developed sample User interface follow web application by on “Figure 3.5”, illustrates UI of Mobile Application.

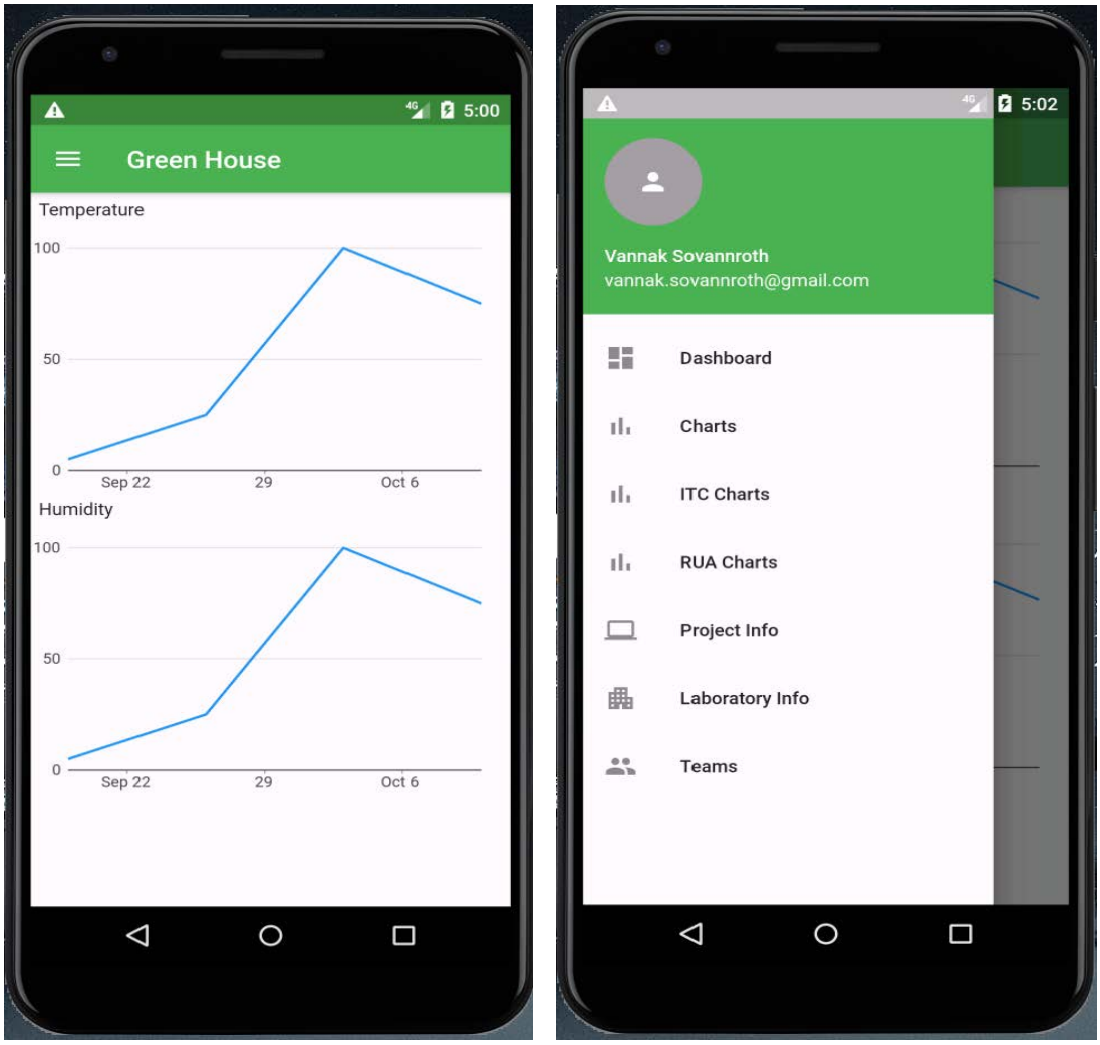


Figure 3.5: Viewing UI of Mobile Application

4. CONCLUSIONS

This project is developing a smart irrigation for smart farm by deploying IoT technology. It based on monitoring systems are in great demand and gives a precise extraction and analysis of data. Using this system, a farmer can share/access the innovative techniques used by him/other farmers to improve the yield, can get the details about irrigation based on moisture control and temperature, crop maintaining information, pesticide details for his farm. Through this framework a farmer can get latest updates via smart-phone/web application.

Future Development

- Planning
  - Backend Development: need to discuss to finalize its feature (Ability of the backend. EX. login, Create account, Manage, Authentication, display…...)

- Frontend Development (Web and Mobile App): Select Framework: Android studio, Flutter; Structure or interface.
  - View data
  - Command/control
- Connection between Backend-Frontend (Develop API)
- Connection between Backend-Sensor Nodes (Develop API)

❖ Feature

- Real time monitoring: Connect to devices and data collection (Development board: Arduino, RPI and new hardware design)
- Real time controlling: ON/OFF and Scheduling
- Data storage: Organization data in database
  - 1 database for 1 project for data storage
  - Database for user information (username, password)
  - Database for device information
  - Database for project API key/Token
- Dashboard for control and monitoring both WEB and APP: (Should discuss in back or frontend)
  - Allow user registration and login via frontend
  - Display project information: Overview, Device list, graph, setting, API key and API link to server (Will discuss)
  - Display will be automatically assigned to the user from the platform (Admin assign)
- Communication protocol: Device to gateway and gateway to Platform
  - Device to gateway: Lora, Zigbee, WiFi
  - Gateway to Platform: HTTP only
- Security
  - Secured device while connect to platform: use device registration (Serial number, MAC address)
  - Secure the platform: Not consider yet
  - Secure the data and network: use Token an API for authentication
- API server script
  - Used for communicating with API client code from device

REFERENCES

[1] Node.js & MongoDB library https://mongodb.github.io/node-mongodb-native/3.6/api/

[2] Node.js learning https://www.w3schools.com/nodejs/default.asp

[3] Node.js SQL server https://www.tutorialsteacher.com/nodejs/access-sql-server-in-nodejs

[4] Node.js coding https://www.geeksforgeeks.org/nodejs-tutorials/

[5] Brad Dayley, “Node.js, MongoDB, and AngularJS Web Development”

[6] Ason Krol, “Web Development with MongoDB and Node.js”

[7] David Herron, “Node.js web development: server-side development with Node 10 made easy”

[8] Bootstrap templates https://bootstrapmade.com

[9] Backend templates https://adminlte.io and https://modularcode.io/modular-admin-html

[10] Laravel Modules https://github.com/nWidart/laravel-modules

[11] Laravel Authentication https://www.itsolutionstuff.com/post/laravel-6-auth-login-with-username-or-email-tutorialexample.html

[12] Charts flutter library https://pub.dev/packages/charts_flutter/install

[13] Charts flutter template https://github.com/imaNNeoFighT/fl_chart and https://google.github.io/charts/flutter/gallery.html

[14] Craig Grannell, “The Essential Guide to CSS and HTML Web Design”

[15] Richard Mansfield, “CSS Web Design For Dummies”

[16] Welling L., Thomson L, “PHP and MySQL Web Development”

[17] Terry Matula, “Laravel Application Development Cookbook”

[18] Jorg Krause, “Introducing Bootstrap 4”

[19] Silvio Moreto, Matt Lambert, Benjamin Jakobus, Jason Marah, “Bootstrap 4 Responsive Web Design”

[20] KIV Sakobyly’s thesis, “Development of Progressive Web Application for Sensor Node Project Management and Monitoring”

[21] Piti Phanthasombath, Sinkerd Phoutthavong, Pawith Chanthavong,“The Implementation of Smart Environment Monitoring System for Strawberry Farm”