|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | |  |  |  |  |
|  |  |  |  |  |  |  |  |
| **Student ID Number**  *(Do not include student name as anonymous marking is implemented)* | AMA19481452 | | | | | | |
| **Programme Title** | Computing Technologies Extended Degree EVE | | | | | | |
| **Module Title** | Designing and Developing Products for the Internet of Everything (IoE) | | | | | | |
| **Module Code** *(listed on Moodle and in LTAFP)* | QAC020X355H | | | | | | |
| **Module Convenor** | Dr Ajmal Gharib | | | | | | |
| **Coursework Title** | Individual Assignment | | | | | | |
| **Academic Declaration:**  *Students are reminded that the electronic copy of their essay may be checked, at any point during their degree, with Turnitin or other plagiarism detection software for plagiarised material.* | | | | | | | |
| **Word Count** | 1813 | | | **Date Submitted** | | **04/07/2022** | |
|  |  | | |  | |  | |
|  | | | | | | | |

Table of Contents

[Introduction 4](#_Toc100089449)

[System Description 4](#_Toc100089450)

[System functional requirements 4](#_Toc100089451)

[Features 5](#_Toc100089452)

[Required resources 5](#_Toc100089453)

[Circuit Diagram 6](#_Toc100089454)

[The chosen frameworks 7](#_Toc100089455)

[Execution 16](#_Toc100089456)

[Conclusion 17](#_Toc100089457)

[References 18](#_Toc100089458)

Appendix…………………………………………………………………………………………19

Introduction

According to Mohammad Masoud (2019), the Internet of Everything (IoE) is an I.T. term that refers to a device with detecting, processing, information extraction, and communication capabilities. IoE allows diverse electronic devices with differing capabilities to observe their environment and communicate with one another for data exchange. The term Internet of Everything relates to wireless sensor networks in terms.

IoE is a complicated method that includes several applications, dreams, and myths. It has several uses in health, engineering, computer science, marketing, and even social sciences; yet, it has several difficulties that need to be investigated further like security and privacy.

KhattakH (2019) states that the application of IoE is reliant on multidisciplinary technological developments like sensor and integrated technologies, low-power networking, and big data analytics.

System description

The suggested software is essential that uses an ESP32 board and a DHT11 sensor to collect humidity and temperature from the surroundings. The collected data may be seen in real time via the web application, or it can be viewed in charts for historical purposes. The web app and data are saved on Firebase and may be accessed with an account from anywhere.

System functional requirements

* Take humidity and temperature readings in your environment.
* Upload the collected data to a database online.
* Allow different users to obtain data from multiple devices by creating user accounts.
* A real-time web application that shows collected data.

Features

* The user logs in using their email address and password.
* Display the preceding sensor update's timestamp.
* Display the most recent readings in cards and gauges.
* Charts with timestamps display historical data.
* The user may select the number of records to display in the chart.
* Use checkboxes to enable or disable the displays.
* Table displaying all database records
* Button for removing obtained data Responsive to various screen sizes

Required resources

*Software resources:*

* Arduino IDE: programme the board
* Write the web application code in Visual Studio Code.
* Node.JS was selected as the framework for the web application.
* Firebase Tools: a package that allows you to connect your web application to Firebase

*Equipment assets:*

* The ESP8266 is the board which will gather sensor data and send it to a database on the internet.
* The temperature and relative humidity in the environment are measured via the DHT11 sensor.
* Wi-Fi router: connects the board to the internet.
* Write the relevant code on the computer/laptop

*Other materials:*

* Firebase account
* Arduino C, HTML, CSS, JavaScript, and database expertise are required.

Circuit diagram

Diagram, schematic

Description automatically generated with medium confidence

Fig.1 The circuit diagram

Diagram

Description automatically generated

Fig.2 System high-level diagram

Figure 2 shows a high-level representation of the entire system. This is how it works:

* When the software runs on the ESP8266, it checks the validity and obtains the user UID from Firebase.
* The ESP8266 receives a timestamp, followed by sensor readings.
* The sensor readings, together with the timestamp, are saved in the database.
* In a loop, the readings are delivered to the database on a regular basis.
* The user can gain access to the web application through the public domain.
* The user connects to the web application, and data from the database is retrieved and presented.

The chosen frameworks

1. *Arduino IDE*

The Arduino IDE, according to Mohamed Fezari (2018), is a truly open piece of software used mostly for writing and generating code for the Arduino Module. This is a real Arduino approach that allows code assembly so simple that just a non-technical layman can get started.

The word "IDE" refers to an official Arduino.cc software that is primarily used during editing, compiling, and posting code to the Arduino Device. With this open-source software, almost all Arduino elements are functioning, and it is simple to install and start compiling code on the fly.

It is compatible with MAC, Windows, and Linux operating systems, and it runs on the Java Platform, which includes constructed functions and instructions for debugging, editing, and producing code in the environment.

1. *Node.js*

Node.js, according to Priyesh Patel (2018), is a JavaScript runtime framework that provides everything needed to run a JavaScript script. When the original developers of JavaScript moved it from a browser-only software to a native app that might run on your PC, they developed Node.js. It has an event-driven, non-blocking I/O architecture and runs on the V8 JavaScript runtime engine, which makes it efficient and suitable for real-time applications.

Because Node.js employs fewer threads, it consumes less resources/memory, resulting in quicker job completion. As a result, this single-threaded design is comparable to multi-threaded architecture for our needs. When doing data-intensive operations, multi-threaded languages such as Java make considerably more sense. However, Node.js is the logical choice for real-time applications.

1. *Firebase*

According to Nilanjan Chatterjee (2018), Firebase is a platform for designing mobile and web apps for organisations that demand a real-time database, that implies that when one user edits a data file, the change should be immediately notified to all users. It provides a straightforward and uniform platform for countless apps, but also a bevvy of other Google services. When it comes to creating applications, Firebase handles the majority of the server-side work.

Firebase Tools is a command-line client for launching and managing Firebase projects. It is loaded via the Node Package Manager (npm) and is used in this project to publish the web app to Firebase.

1. *Visual Studio Code*

Chika C Uchendu (2021) says that Visual Studio (VS) Code is a fully accessible code editor that is largely used to fix and repair coding problems in cloud and online applications. The tools provided by Visual Studio Code could be used to improve the functionality of every written code. Users do not need to download any additional software to synchronise code between both the server and the editor.

Python, C++, Go, JavaScript, and Node.js are among the programming languages supported by the Visual Studio Code. Aside from fundamental features like as code folding, syntax highlighting, and bracket matching. Other characteristics may vary depending on the programming language. VS Code also includes IntelliSense for TypeScript, CSS, HTML, and other languages, as well as Node.js debugging. Users may use the free extension to get extra translation services, themes, and debuggers. Users may also install a code linter and do static code analysis using the language server protocol.

One of the main benefits of Visual Studio Code is that programmers can quickly switch between tools to make changes to their code. VS Code, in addition to its simple customization choices, provides keyboard shortcuts for typical key combinations and repetitive activities. In addition to website development, programmers may utilise Visual Studio Code on desktop apps. Also, front-end developers use the integrated development environment in Visual Studio Code to create their code highly efficient and error-free.

Execution

The code that runs on the ESP8266 board will be shown in the screenshots below, along with some explanation.

Graphical user interface, text, application, email

Description automatically generated

Fig.3 Importing libraries and defining variables

The first portion of code, seen in Figure 3, begins with the importing of the required libraries and the definition of global variables. The Wi-Fi network connection credentials, the database URL and API key, as well as the password and username used to login to the Firebase platform, are all variables.

Text, letter

Description automatically generated

Fig.4 Connect to a Wi-Fi network function

The WiFi library is used by the function shown in Figure 4 to attach the board to the Wi-Fi network.

Graphical user interface, text

Description automatically generated

Fig.5 Obtain the current time with this function.

The method that retrieves the current timestamp and sends it to the database with the sensor values is shown in Figure 5.

Text

Description automatically generated

Fig.6 The configuration

Figure 6 shows the setup function, which is executed once when code execution begins on the board. It configures the sensor, links to the Wi-Fi network, and validates the account in Firebase.

Graphical user interface, text, application, email

Description automatically generated

Fig.7 The loop technique

Figure 7 depicts the loop technique, which runs continually on the board and receives sensor measurements and sends the data to the database.

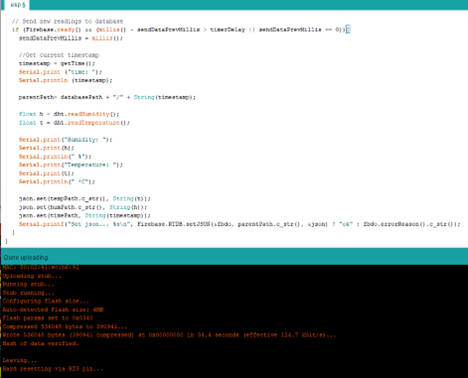


Fig.8 Transfer the code to the ESP8266 board



Fig.9 The data is sent to the database by the ESP8266 hardware.

The methods for configuring Firebase and uploading the web application will be detailed next.

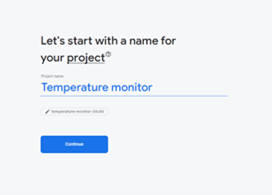


Fig.10 Make a fresh Firebase project.

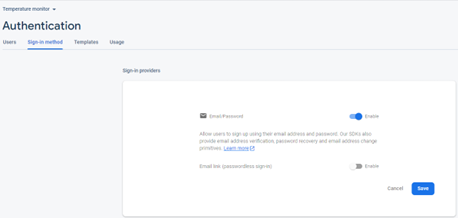


Fig.11 Firebase authentication should be enabled.

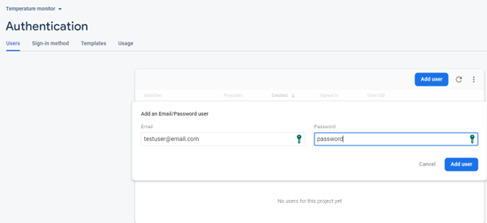


Fig.12 Create a new Firebase user.

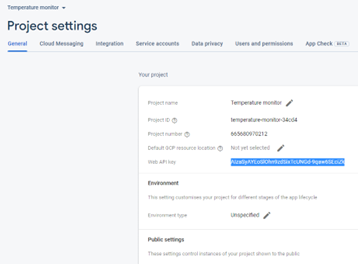


Fig.13 Obtain the API key for the project

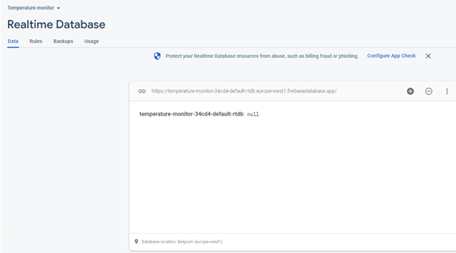


Fig.14 Make a database

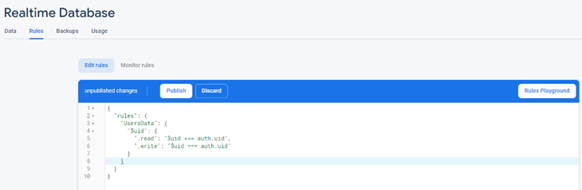


Fig.15 Set up the database rules

Graphical user interface, text, application

Description automatically generated

Fig.16 Make a new Firebase web application

Text

Description automatically generated

Fig.17 The code for the web application

Text

Description automatically generated

Fig.18 Set up the Firebase tools for web application deployment.

Text

Description automatically generated

Fig.19 Firebase the web application

The HTML, CSS, and JavaScript code for the web application is included in the Appendix. To link to the database, JavaScript libraries including "firebasejs" are used, as well as "highcharts" and "canvas-gauges" to show the charts and gauge.

Graphical user interface

Description automatically generated

Fig.20 Real-time measurements from the online application are shown in gauges

Graphical user interface, chart, application, line chart

Description automatically generated

Fig.21 The charts of temperature and humidity

Conclusion

Users may access and see all released data on an online platform by publishing temperature and relative humidity sensor values to an online database. The system currently only supports temperature and humidity data, but the capability may easily be expanded to accommodate measurements from other sensors like pressure, quality of air, gas, or biometric sensors.

There are no major security concerns because the system derives the majority of its functionality from the very secure Firebase platform. The main issue is that an attacker may breach the Wi-Fi network as well as collect data sent from the board to the database, which again is reflected in sensor readings.

There are similar online platforms on the industry that support a variety of sensors, but the majority of them are limited to the use of sensors and boards provided by the platform's proprietor.

References

Mohammad Masoud (2019) *“Sensors of Smart Devices in the Internet of Everything (IoE) Era: Big Opportunities and Massive Doubts”.* Available at:<https://www.hindawi.com/journals/js/2019/6514520/> (Accessed: 26/06/2022);

KhattakH (2019) *“Computer Communications”.* Available at:<https://www.sciencedirect.com/science/article/abs/pii/S0140366419318754#preview-section-references> (Accessed: 27/06/2022);

Mohamed Fezari (2018) *“Integrated Development Environment "IDE" For Arduino”.* Available at: <https://www.researchgate.net/profile/Mohamed-Fezari-2/publication/328615543_Integrated_Development_Environment_IDE_For_Arduino/links/5bd8c6d24585150b2b9206df/Integrated-Development-Environment-IDE-For-Arduino.pdf> (Accessed: 30/06/2022);

Priyesh Patel (2018) *“What exactly is Node.js?”.* Available at:<https://www.freecodecamp.org/news/what-exactly-is-node-js-ae36e97449f5/> (Accessed: 29/06/2022);

Nilanjan Chatterjee (2018*) “Real-time Communication Application Based on Android Using Google Firebase”.* Available at:<https://www.researchgate.net/publication/324840628_Real-time_Communication_Application_Based_on_Android_Using_Google_Firebase> (Accessed:30/06/2022);

Chika C Uchendu (2021) *“Visual Studio Code”.* Available at: <https://www.webopedia.com/definitions/visual-studio-code/> (Accessed 30/06/2022);

Appendix

Here it is the code for the project.

