Table of Contents

Abstract

Wellbeing is fundament requirement. What's more, it is human appropriate to get quality health care. These days, India is confronting numerous medical problems in light of fewer assets. This survey article displays the idea of solving health issues by utilizing a recent innovation, the Internet of Things (IOT). The Internet of Things with their developing interdisciplinary applications has changed our lives. Smart health care being one such IoT application interfaces brilliant gadgets, machines, patients, specialists, and sensors to the web. IoT based mobile phone health monitoring gives users a better way to monitor their health and the cost is also less compared to staying in hospitals. The objective of this project work is to develop a mobile platform that has the capability to collect sensor data from heterogeneous wearable sensors, process the data, analyze the data and trigger the warning messages to users. The system will provide the capability to visualize the signal in user mobile. The system is able to collect the sensor data to monitor the basic vital parameters such as pulse rate and body temperature from the wireless body sensors to the patient's Smartphone using wi-fi module. The collected data is sent to the cloud database for the phone to retrieve it. The data can be analyzed in the phone itself to find the warning level.

Acknowledgements

We owe our deep gratitude to our mentor, Mr. Mohd Abdul Qadeer, from Faculty of Computer Engineering, who took keen interest in our project and guided us all along, till the completion of our project work by providing all the necessary information. We are extremely thankful to them for providing support and guidance. We are also thankful to all the Professors and Faculty Members in the department for their teachings and academic support.

Ahsan Waseem
Mohd Azhan Umar Kamil
Mohd Tanveer Hasan
Date: 12 December 2022

List of Figures

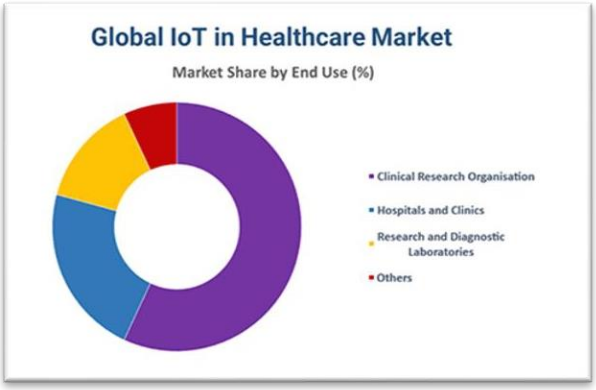
Chapter 1: Introduction

This chapter

1.1 Motivation

Wearable sensors and the Internet of Things (IoT) can become as essential for monitoring our health as sensors are for monitoring a car's performance. Moreover, they have the potential to completely change the way healthcare is delivered, improving patient services and saving lives. By 2030 there will be over 3 billion wearable sensors in use providing a huge amount of additional data that can be used for health monitoring. We are already seeing smart watches that can monitor activity, as well as vital signs such as heart rate and blood oxygen levels. Earlier this year, Google announced that it will start testing a medical device that tracks health data. The device would be able to track cardiac signals of heart attack patients or monitor how the health of Parkinson's disease sufferers' changes if they spend less time moving around or going outdoors.

1.2 Objectives and Scope



The objective of this semester's project was to design an easy to use health monitoring app using sensors and IoT devices. The whole system consists of three components: mobile app (front-end), cloud database (back-end), and IoT devices for collecting sensitive user data. Specifically, we used Flutter for developing and designing the Android mobile app, Google's Firebase for user authentication and IoT devices such as Pulse Sensor (name) and Temperature Sensor (name) for measuring the heart pulse rates and body temperature respectively.

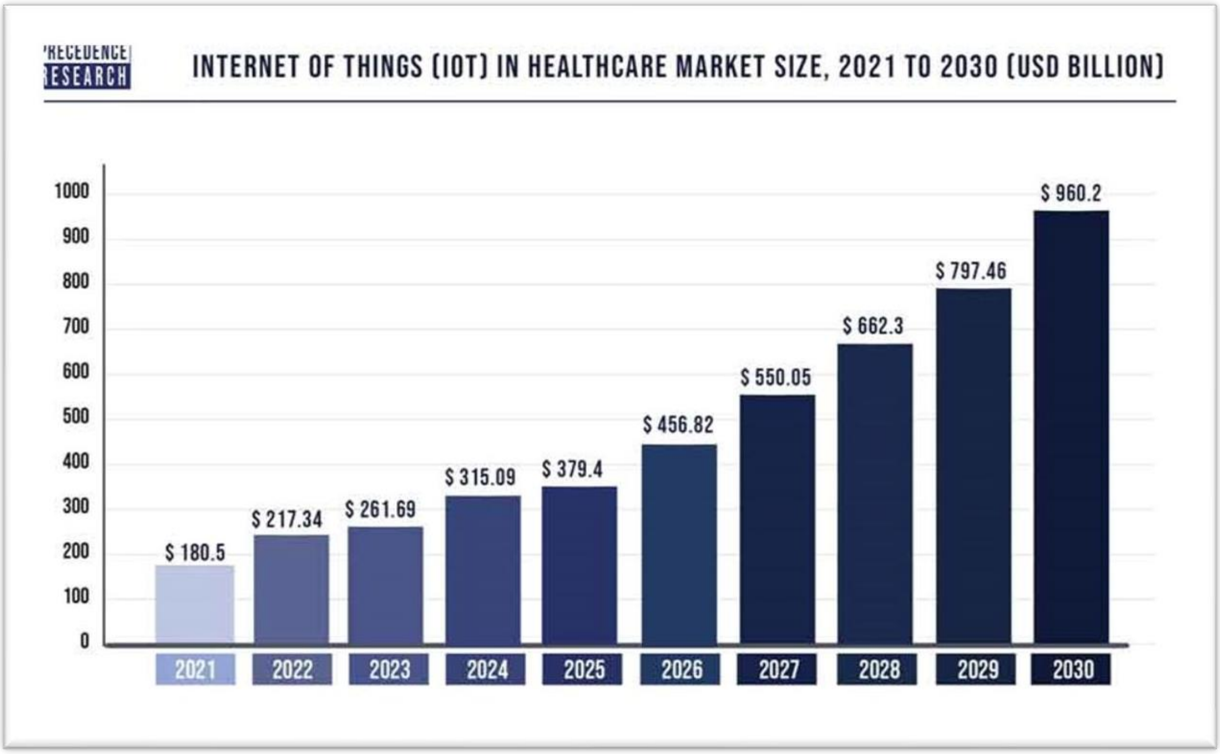


Figure 1.2: IOT in healthcare market

1.3 Organization

The report begins with an abstract and a list of figures used. After that there is a summary that briefly tells the scope and objective of the project as well as the importance of the topic. This is followed by the methodology used for implementation, and the results obtained. The report ends with a conclusion and the scope for improvement in the project.

Chapter 1 explains the importance of the topic and scope of our work.

Chapter 2 provides the insight to some of the research work that has already been done and helped us to understand the topic deeply.

Chapter 3 discusses about the project overview.

Chapter 4 discusses the tools used in the project.

Chapter 5 explains the overall proposed system and then the flow of work for this project i.e. how the ultimate goal was achieved.

Chapter 6 discusses the results obtained at the end of the project.

Chapter 7 gives the conclusion and ideas for improvement of the project.

The main report is followed by the references that have been used while understanding the project and working on it. Wherever these references have been used, they are cross referred by their serial number in the list of references.

Chapter 2: Literature Review

[1] A new lot-based health monitoring approach in which collected medical sensor data is sent to an analysis module via low-cost, low-power and secure communication links provided by a LoRa WAN network infrastructure. It mainly focusses on monitoring blood pressure, glucose and temperature in rural areas where cellular network coverage is either absent or does not allow data transmission. The main objective is to reduce the burden of long trips for people living in these areas to visit healthcare facilities, while minimizing the communication cost. Several experiments have been conducted to evaluate the area covered by the LoRa network and the power consumption of our system.

[2] This paper presents an IoT architecture customized for healthcare applications. The proposed architecture collects the data and relays it to the cloud where it is processed and analyzed. Feedback actions based on the analyzed data can be sent back to the user. A prototype of the proposed architecture has been built to demonstrate its performance advantages.

[3] While Sensor is one such technology which can be used to enable Internet of Things based health-care monitoring system. In this paper, implementation of such a system is described. In a real time health monitoring system, there are many sensors connected to a local data processing unit through a shared channel having a fixed bandwidth. These sensors have a wide variety of channel access requirements. The access to the channel should be discrete, so that each and every sensor avails the required bandwidth and delay in the shared channel. Implementation of this technique in our prototype health-care monitoring application is discussed in this report.

Chapter 3: Project Overview

3.1 Mobile App (Front End)

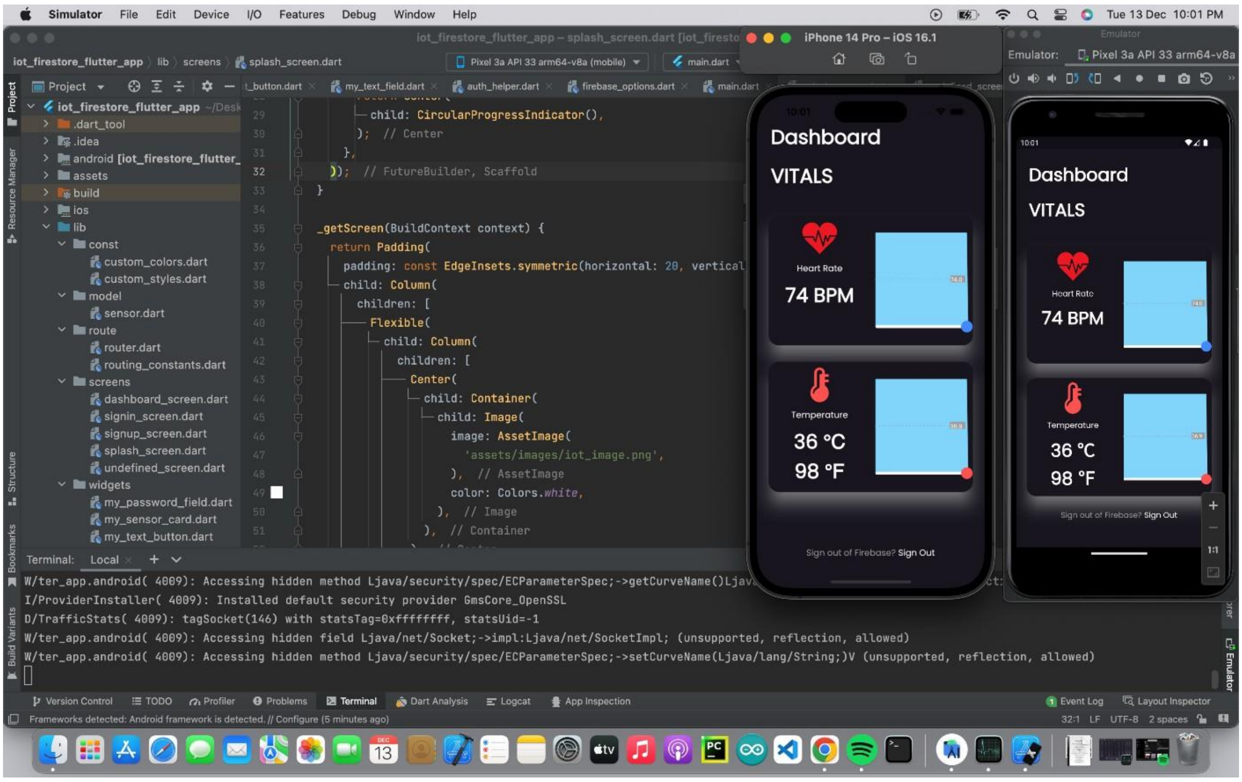


FIGURE 3.1: Android Studio IDE Interface

The whole UI design and development was done in Android Studio using Flutter. We utilized the concept of MVC (Model View Controller) design pattern and dividing the whole project into three segments for handling different functionalities.

The completed app was made to run in simulator provided by Android Studio neural networks tend to be static and symbolic, while the biological brain of most living organisms is dynamic (plastic) and analog.

The simulator shown runs the app on a Google PIXEL and iPhone 14 Pro.

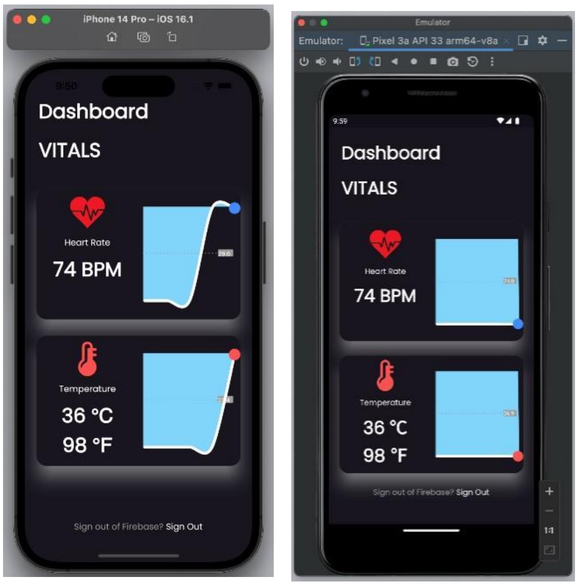


FIGURE 3.2: iPhone and Pixel Emulator

3.2 Google's Firebase (Back-end)

Firebase is a toolset to "build, improve, and grow our app", and the tools it gives us cover a large portion of the services that developers would normally have to build themselves, but don't really want to build, because they'd rather be focusing on the app experience itself. This includes things like analytics, authentication, databases, configuration, file storage, push messaging, and the list goes on. The services are hosted in the cloud, and scale with little to no effort on the part of the developer.

;

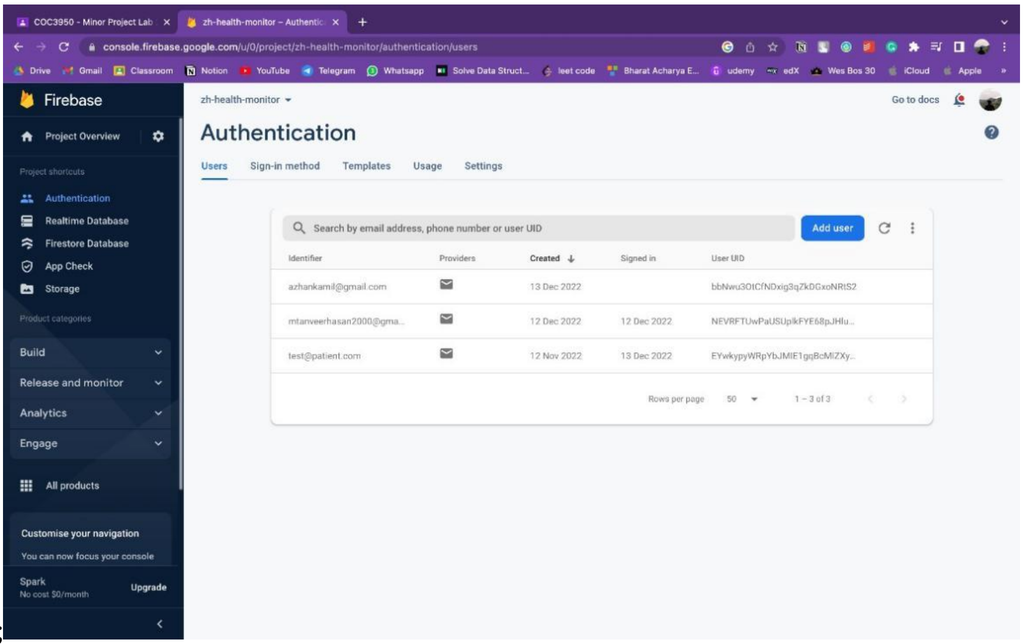


FIGURE 3.3: Google Firebase Console

In this project we've utilized the feature to authenticate users while registering and signing in. As it can be seen that we've already registered two users and three passwords are secured through encryption by Firebase

3.3 : IoT Architecture and Sensor

The basic IoT architecture diagram below defines how the IoT devices work. Taking our project case into consideration the data collected by the Arduino from various sensors is sent to the free cloud platform through ESP8266 Wi-Fi module and then data is collected from the developed iOS application for the health monitoring purpose.

Chapter 4: Tools Used

4.1 Android Studio

Android Studio is the official Integrated Development Environment (IDE) for Android app development, based on IntelliJ idea. On top of IntelliJ's powerful code editor and developer tools, Android Studio offers even more features that enhance your productivity when building Android apps, such as:

FIGURE 4.1: Android Studio

- A flexible Gradle-based build system
- A fast and feature-rich emulator
- A unified environment where you can develop for all Android devices
- Apply Changes to push code and resource changes to your running app without restarting your app
- Code templates and GitHub integration to help you build common app features and import sample code
- Extensive testing tools and frameworks
- Lint tools to catch performance, usability, version compatibility, and other problems
- C++ and NDK support
- Built-in support for Google Cloud Platform, making it easy to integrate Google Cloud Messaging and App Engine

4.2 Flutter

Flutter is an open source UI software development kit created by Google. It is used to develop cross-platform applications for Android, iOS, Linux, MacOS, Windows, Google Fuchsia and the web from a single codebase. Flutter apps are written in the Dart language and make use of many of the language's more advanced features.

FIGURE 4.2: Flutter

While writing and debugging an application, Flutter runs in the Dart virtual machine, which features a just-in-time execution engine. This allows for fast compilation times as well as "hot reload", with which modifications to source files can be injected into a running application. Flutter extends this further with support for stateful hot reload, where in most cases changes to source code are reflected immediately in the running app without requiring a restart or any loss of state.

4.3 Firebase

Firebase is a set of hosting services for any type of application.
It offers NoSQL and real-time hosting of databases, content, social authentication, and notifications, or services, such as a real-time communication server.

The Firebase Realtime Database is a cloud-hosted database. Data is stored as JSON and synchronized in Realtime to every connected client. When you build cross-platform apps with our Apple platforms, Android, and JavaScript SDKs, all of your clients share
one Realtime Database instance and automatically receive updates with the newest data.

Firebase

FIGURE 4.3: Firebase

4.4 Arduino IDE

The Arduino Integrated Development Environment - or Arduino Software (IDE) - contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions and a series of menus. It connects to the Arduino hardware to upload programs and communicate with them. Programs written using Arduino Software (IDE) are called sketches. These sketches are written in the text editor and are saved with the file extension .ino. The editor has features for cutting/pasting and for searching/replacing text. The message area gives feedback while saving and exporting and also displays errors. The console displays text output by the Arduino Software (IDE), including complete error messages and other information. The bottom righthand corner of the window displays the configured board and serial port. The toolbar buttons allow you to verify and upload programs, create, open, and save sketches, and open the serial monitor.

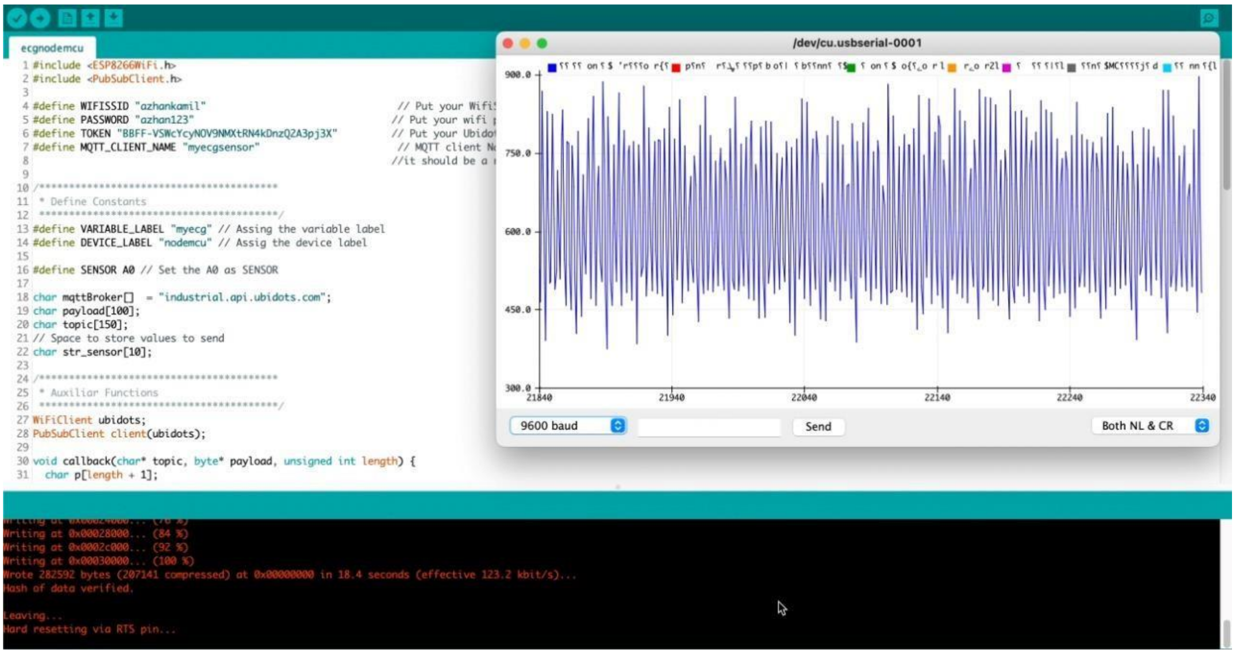


FIGURE 4.4: Arduino IDE with serial monitor

4.5 NodeMCU ESP8266

NodeMCU is an open source firmware for which open source prototyping board designs are available. Both the firmware and prototyping board designs are open source.
The firmware uses the Lua scripting language. The firmware is based on the eLua project, and built on the Espressif Non-OS SDK for ESP8266. It uses many open source projects, such as lua-cjson and SPIFFS. Due to resource constraints, users need to select the modules relevant for their project and build a firmware tailored to their needs. Support for the 32-bit ESP32 has also been implemented.
The prototyping hardware typically used is a circuit board functioning as a dual in-line package (DIP) which integrates a USB controller with a smaller surface-mounted board containing the MCU and antenna.

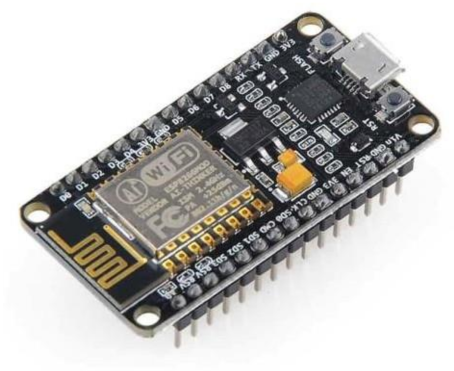


FIGURE 4.5: NodeMCU

The choice of the DIP format allows for easy prototyping on breadboards. The design was initially based on the ESP-12 module of the ESP8266, which is a Wi-Fi SoC integrated with a Tensilica Xtensa LX106 core, widely used in IoT applications

4.6 MLX90614 Temperature Sensor

This temperature sensor is used in this project to measure the human body temperature. This sensor uses infrared rays tp measure the human body temperature. This sensor can measure the temperature from -70˚C to 382˚C. The main advantage of this sensor is that it measures the body temperature without touching which is. very much beneficial in this CoVID situations.

FIGURE 4.6: MLX90614 Sensor

4.7 MAX30102 Heart Rate Sensor

The MAX30102 is an integrated pulse oximetry and heart-rate monitor module. It includes internal LEDs, photodetectors, optical elements, and low-noise electronics with ambient light rejection. The MAX30102 provides a complete system solution to ease the design-in process for mobile and wearable devices. The MAX30102 operates on a single 1.8V power supply and a separate 3.3V power supply for the internal LEDs. Communication is through a standard I2C-compatible interface. The module can be shut down through software with zero standby current, allowing the power rails to remain powered at all times.

FIGURE 4.7 MAX30102 Sensor

Chapter 5: Implementation

5.1 Overview

The sensors detect the temperature and heart rate and send the date to the cloud database, i.e., firebase. The information transfer from sensors to database is done with the help of NodeMCU via ESP8266 Wi-Fi module. The app then retrieves the data from the cloud database and displays it.

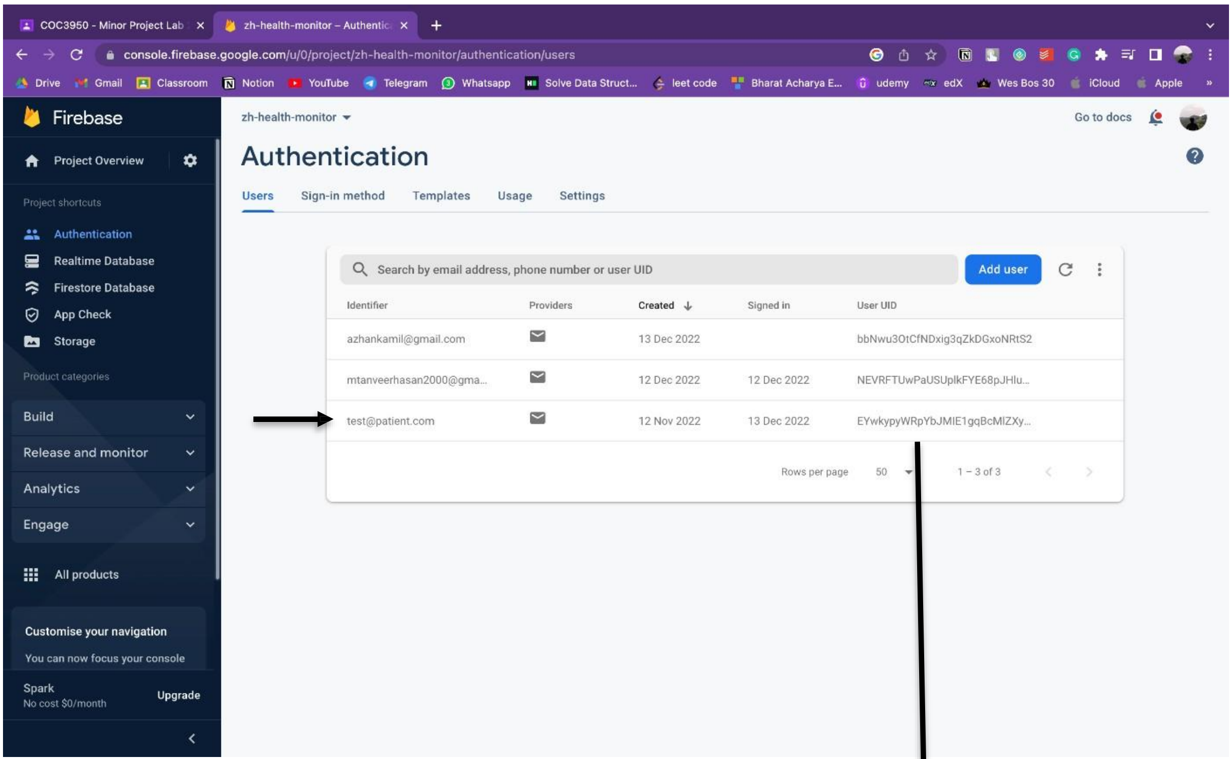
5.2 Sign in and Register

To use the app, the user must be registered in database. Users can use the registering feature to create their account and once the account is created the user can login.

FIGURE 5.1: Sign in and register

After registering, the user will be added to database and can Sign in.

The registered user added to database is show below database.



The user can then use his credentials to login into the app.

Login Information

Sign in button

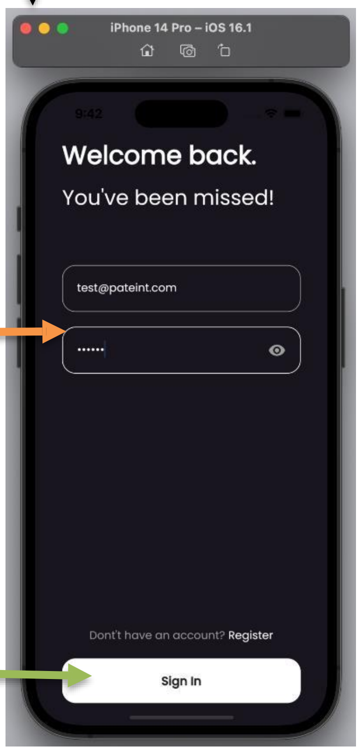


FIGURE 5.2: User database

5.3 Display User Data.

The data sent by sensor to cloud is retrieved by the app and displayed.

.

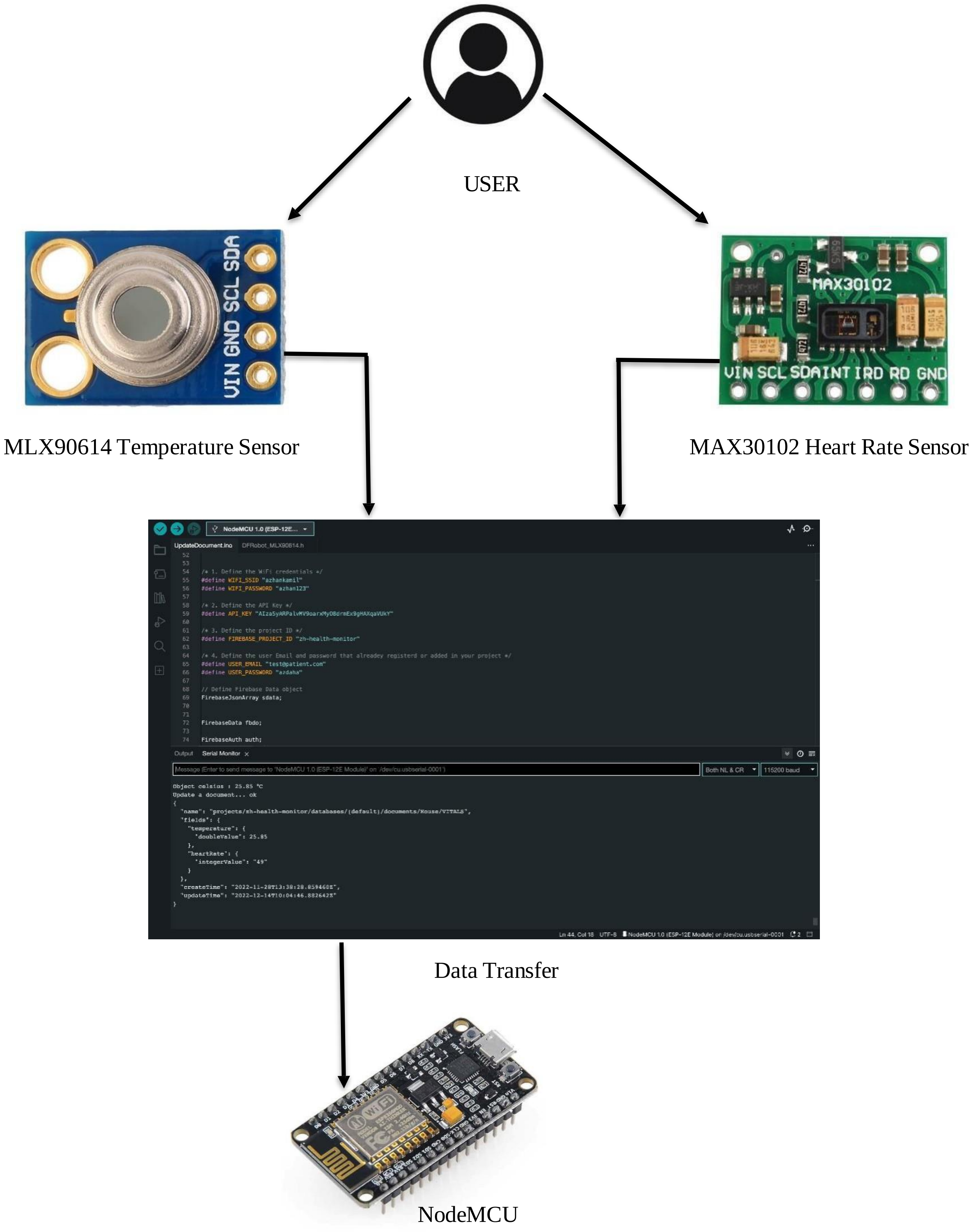


FIGURE 5.3: Data Collection from Sensor

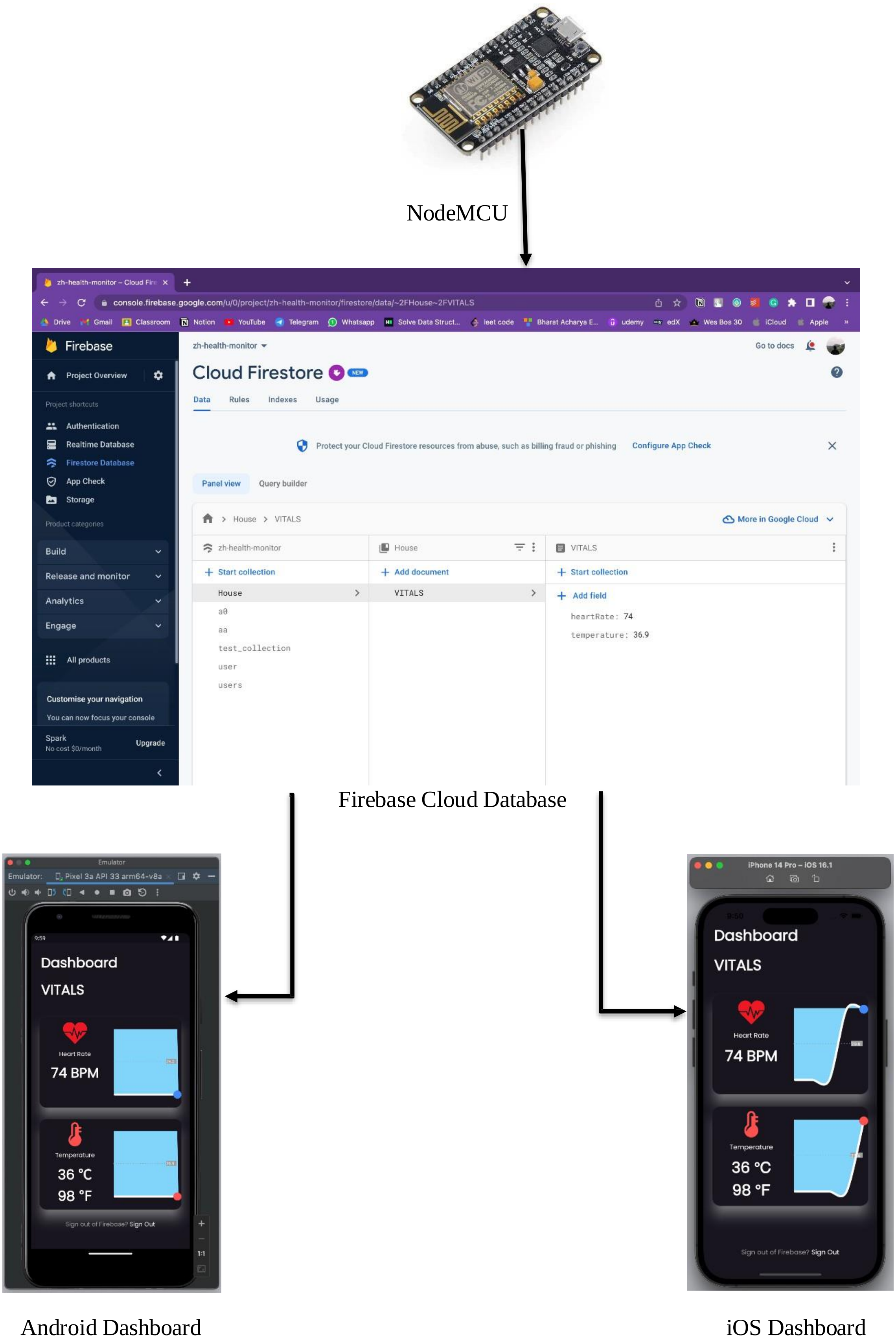


FIGURE 5.4: Displaying data on app

Chapter 6: Conclusions and Future Work

6.1 Conclusions

In summary, one vision of the future is that IoT becomes a utility with increased sophistication in sensing, actuation, communications, control, and in creating knowledge from vast amounts of data. This will result in qualitatively different lifestyles from today. What the lifestyles would be is anyone ‘s guess. It would be fair to say that we cannot predict how lives will change. We did not predict the Internet, the Web, social networking, Facebook, Twitter, millions of apps for smartphones, etc., and these have all qualitatively changed societies ‘lifestyle. Especially for healthcare system, it is most useful to every human. So, health problems are easily predicted at the beginning stage based on IoT healthcare system.

6.2 Future Work

IoT has a long way to go not only in healthcare, but on all the paths it has tread so far and will be traversing in future. Throughout the healthcare industry, its use is not as widespread as it has potential to achieve, but it will come. As the sizes and prices of the devices go down, rapid scaling of these technologies will take place. IoT in healthcare is set to radically change the way healthcare sees the management of inventory, the optimization of workflow and the integration of devices. The digital transformation of healthcare industry, among others, will most certainly be brought about by what is now being called the Internet of Healthcare Things (IoHT).

We have got a pretty good result but still there is a huge scope of improvement.

1. Incorporate a messaging system that alerts nearby hospitals or registered doctors in case of some serious health issue.
2. Adding the feature of measuring daily calorie intake and diet analysis.
3. Integrate a greater number of sensors to provide more insight regarding the user’s health.

References

1. Atzori Luigi, "Antonio Iera and Giacomo Morabito. -The Internet of Things: A Survey", IEEE EUROCON 2017 -17th International Conference on Smart Technologies https://ieeexplore.ieee.org/abstract/document/8011165/authors#authors

2. Abdelgawad A., Yelamarthi K., Khattab A. (2017), “IoT-Based Health Monitoring System for Active and Assisted Living”. https://link.springer.com/chapter/10.1007/978-3-319-61949-1_2.

3. S.K.Dhar, S.S.Bhunia, Nandini Mukherji, “Interference Aware Scheduling of Sensors in IoT Enabled Health-Care Monitoring System”. 2014 Fourth International Conference of Emerging Applications of Information Technology

4. Google’s firebase toolkit for authentication and data management https://firebase.google.com/products/firestore