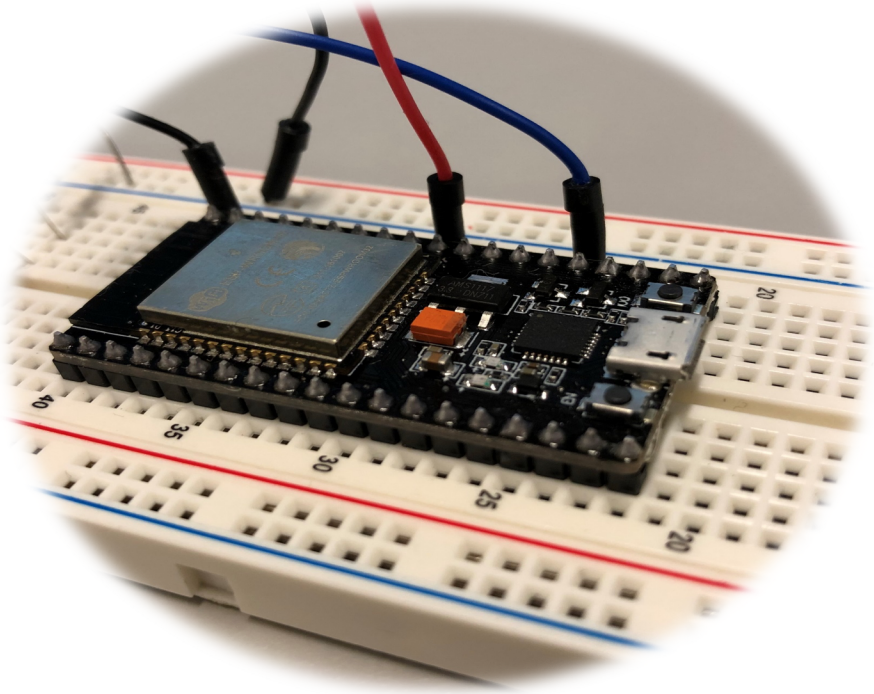
University of Houston Clear Lake

# Mavericks Capstone Project "Offline Monitoring System"



Hajera Nasreen
Vidhi Sharma
Divya Kamboj
Priyankaa Rajendran

University of Houston Clear Lake

# Mavericks Capstone Project "Offline Monitoring System"

---


**Instructor**

Dr. Pradeep Buddharaju
Assistant Professor
Computer Science

Email: Buddharaju@uhcl.edu

**Mentors**

Thor Arne Lovland
COO
Email: tal@optimesubsea.com

Channa Weeratunge
Senior Software Engineer
Email: cw@optimesubsea.com

# Outline

- Introduction
- Project Overview
- Requirement Specification
- Design
- Implementation
- Design for the Mobile Application
- Demo
- Project Timeline
- Project Dynamics

# Introduction


- Optime Subsea is a company which offers Subsea engineering.
- Optime Subsea has an industrial control software which monitors and performs operations.
- Our project is to design, build and implement Offline Monitoring System to monitor pressure, temperature and location.

# Project Overview

- Sometimes it is necessary to monitor environmental conditions in closed systems. A closed system is defined as one where the system does not have a connection to an external environment to transmit its data in real-time.

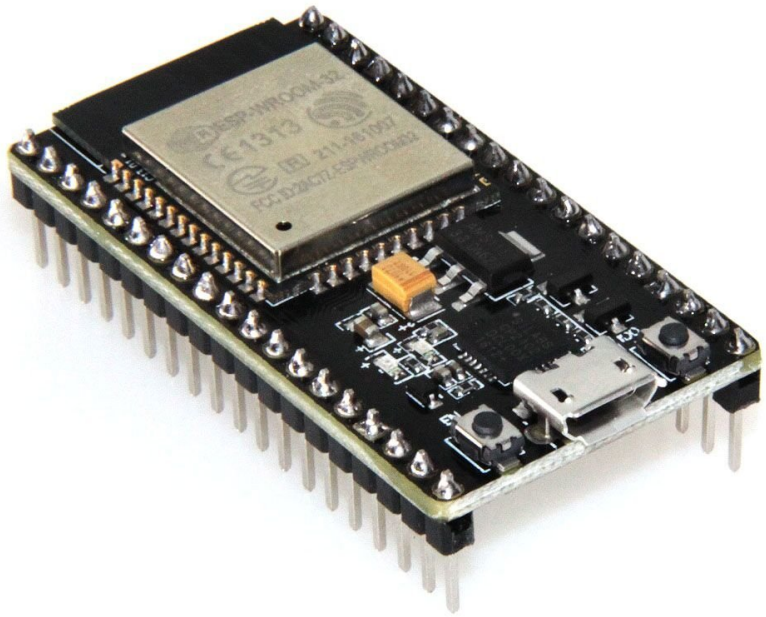
- For example, monitoring the pressure inside a canister as it floats above the earth's surface. For such systems, the data should be logged and when possible, downloaded from the system for further processing.

# Project Overview

- The Internet of things (IoT) is the network of physical devices, vehicles, home appliances and other items embedded with electronics, software, sensors, actuators, and connectivity which enables these objects to connect and exchange data.

- Through the use of an ESP32, it is possible to use Bluetooth to connect to the external environment. The preferred external environment shall be a smartphone running on Android or iOS operating system.

# ESP 32



- This is a compact SOC that has built-in capabilities to connect to external sensors through Wi-Fi and Bluetooth.

- ESP 32 Features:
  - Integrated 802.11BGN HT40 Wi-Fi transceiver.
  - Integrated
  - Flash Frequency: 80MHz
  - Flash Size: 4 MB
  - Low battery consumptions

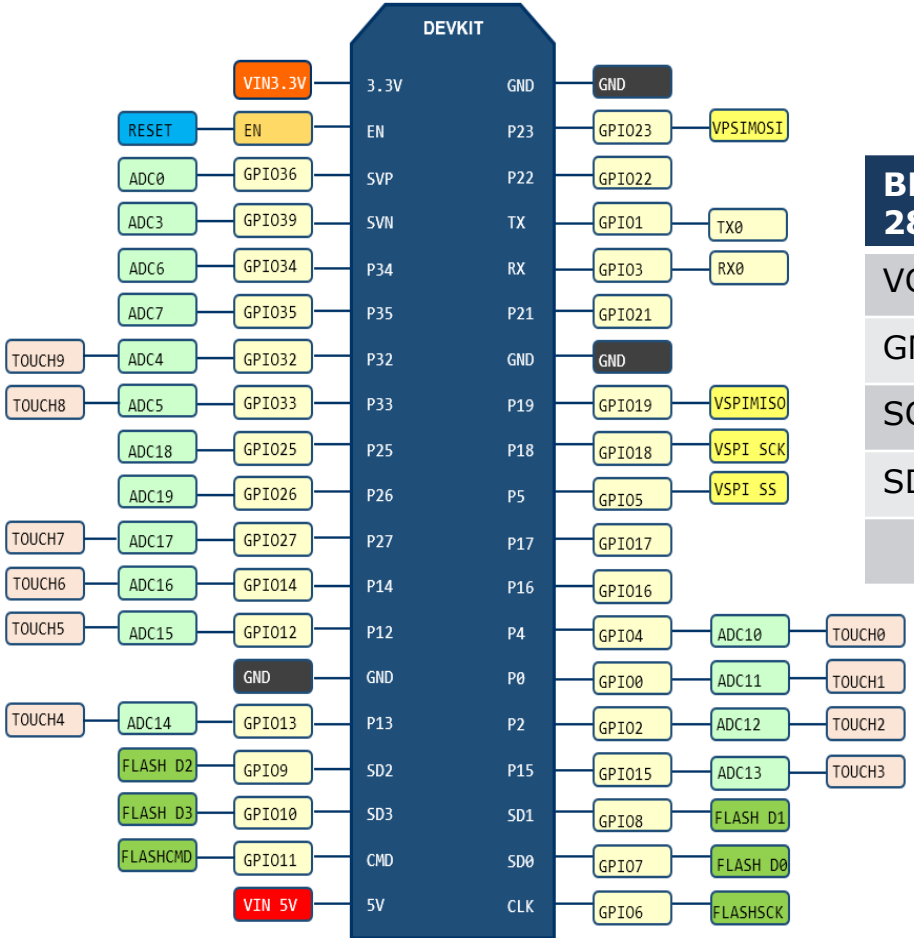
# Temperature and Pressure Sensor


- We are using BME 280 sensor to monitor temperature and pressure.
- Using BME 280,Get more precise temperature, atmospheric pressure values fast.
- Operation Range:
  - Temperature Range: -40…85 °C
  - Pressure Range: 300…1100 hPa

# GPS Module


- We are using Ublox GPS Module to monitor location.
- Built-in fast 50 satellite positioning and tracking ability ublox chip, built-in high gain LNA.
- Easy to setup, very fast lock(locks signal in 10-30 second)
- 1-5Hz position refresh rate

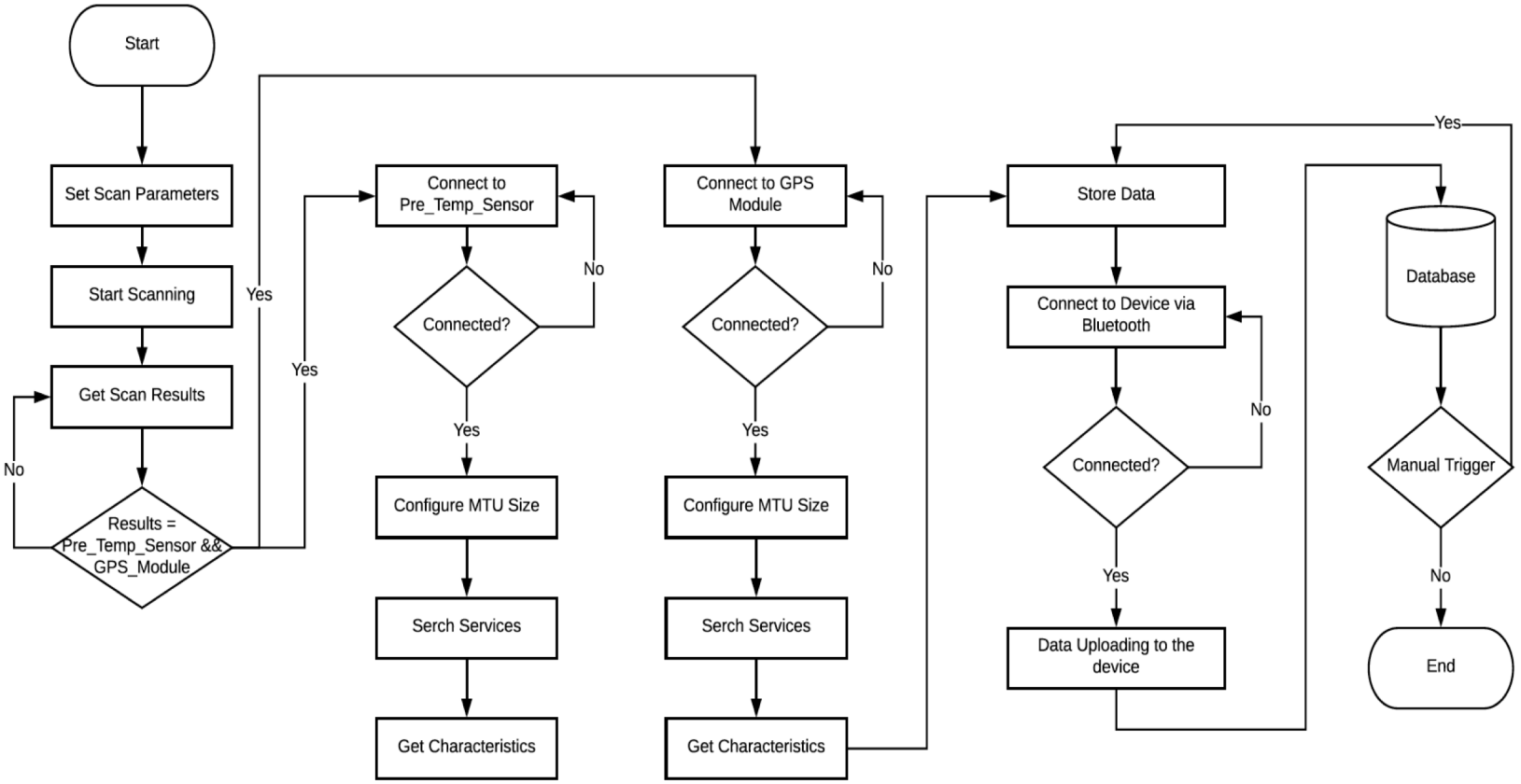
# Design




## Pin Connections

| BME 280 | ESP 32 | Ublox | ESP 32 |
|---|---|---|---|
| VCC | 5 V | E | 5 V |
| GND | GND | V | 5 V |
| SCL | GPIO 22 | G | GND |
| SDA | GPIO 21 | T | GPIO3 |
| | | R | GPIO1 |

# Flow Chart

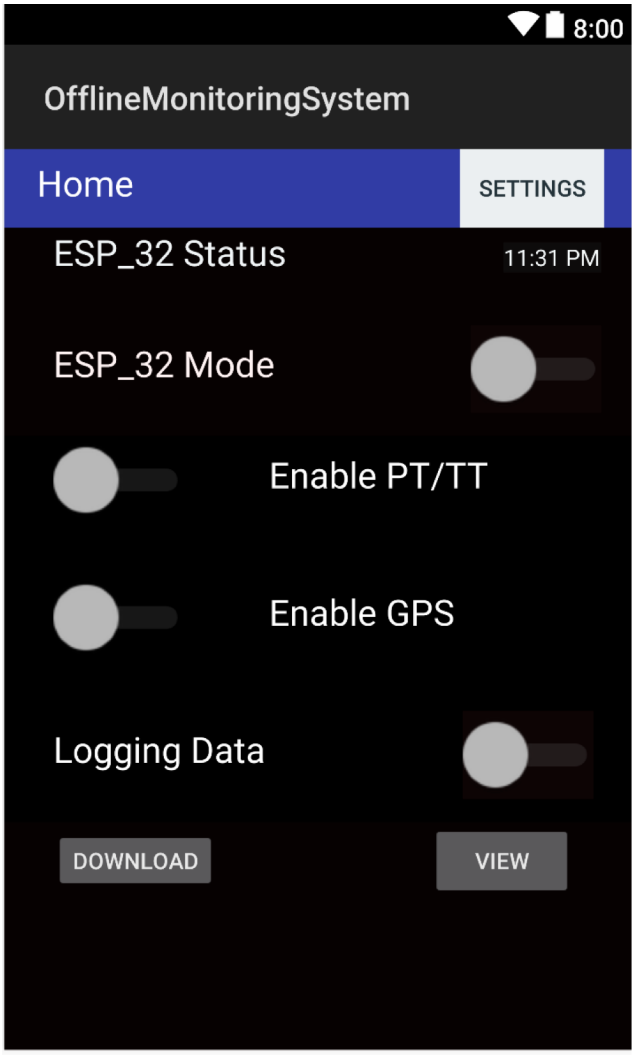

Start
Set Scan Parameters
Start Scanning
Get Scan Results
Results = Pre_Temp_Sensor && GPS_Module
No
Yes
Yes
Connect to Pre_Temp_Sensor
Connected?
No
Yes
Configure MTU Size
Serch Services
Get Characteristics
Connect to GPS Module
Connected?
No
Yes
Configure MTU Size
Serch Services
Get Characteristics
Store Data
Connect to Device via Bluetooth
Connected?
No
Yes
Data Uploading to the device
Yes
Database
Manual Trigger
No
End

# Implementation


- We are using the Arduino IDE for ESP 32 coding.

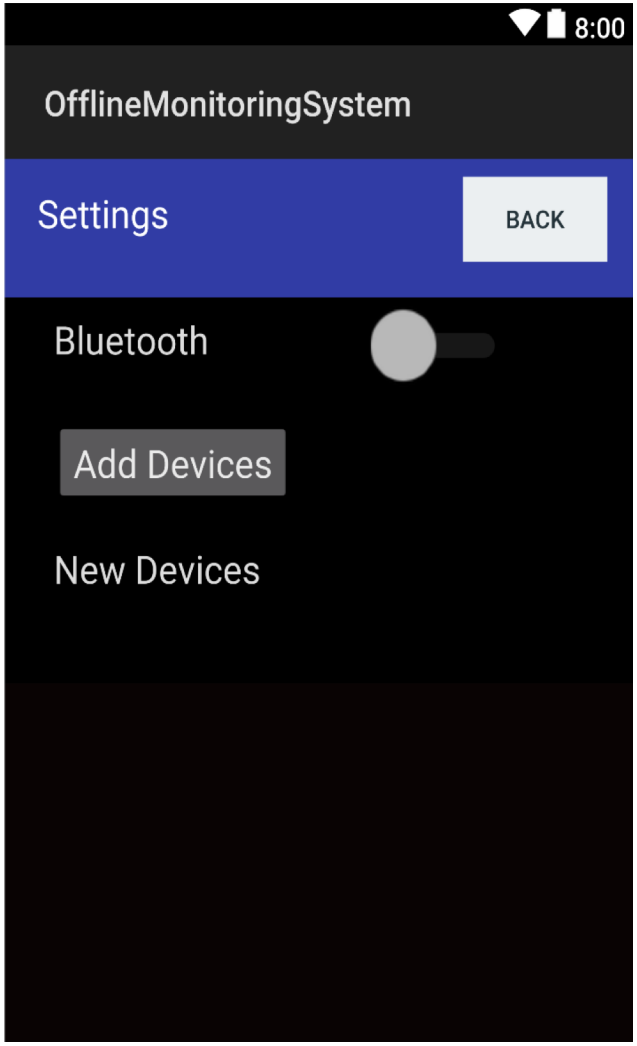

- We are using the Xamarin to create mobile application.

# Mobile Application Design



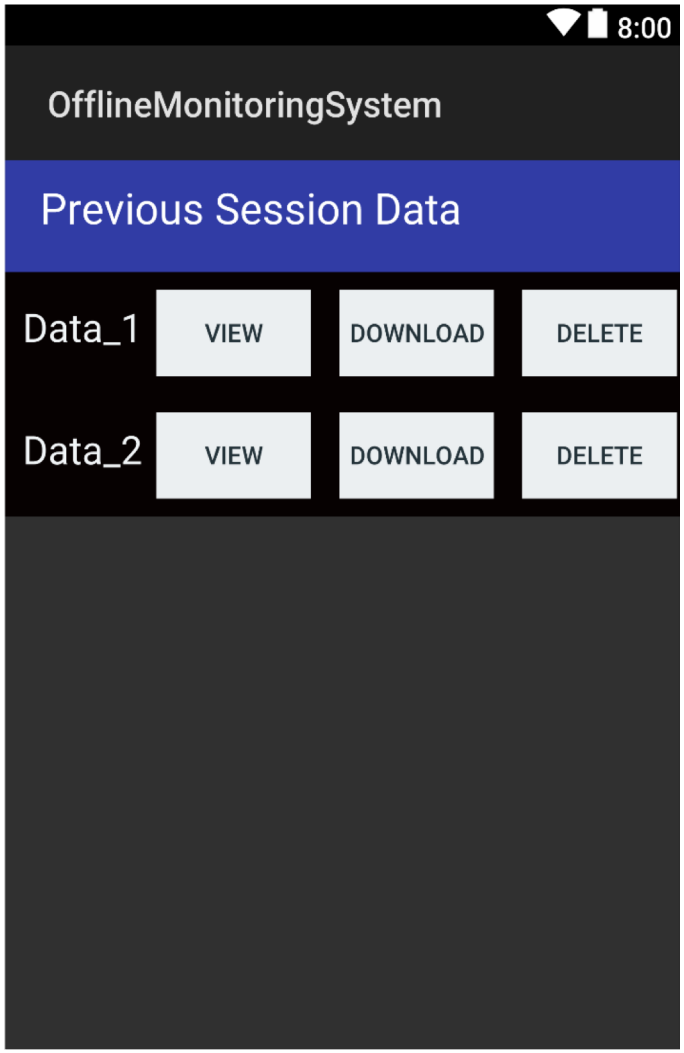

- ESP_32 is connected and is either waiting/collecting data from the sensor, when the user enables PT/TT or GPS sensor ESP_32 status changes to collecting data, once the data is collected it is available to download or view for the user.

# Mobile Application Design




- The settings page displays the devices that is available to connect with the Bluetooth, the user enters a four digit password to connect ESP_32 with the Bluetooth. After connecting with the Bluetooth, the application stores the Bluetooth connection information in a file and then directs the user to the home screen.

# Mobile Application Design



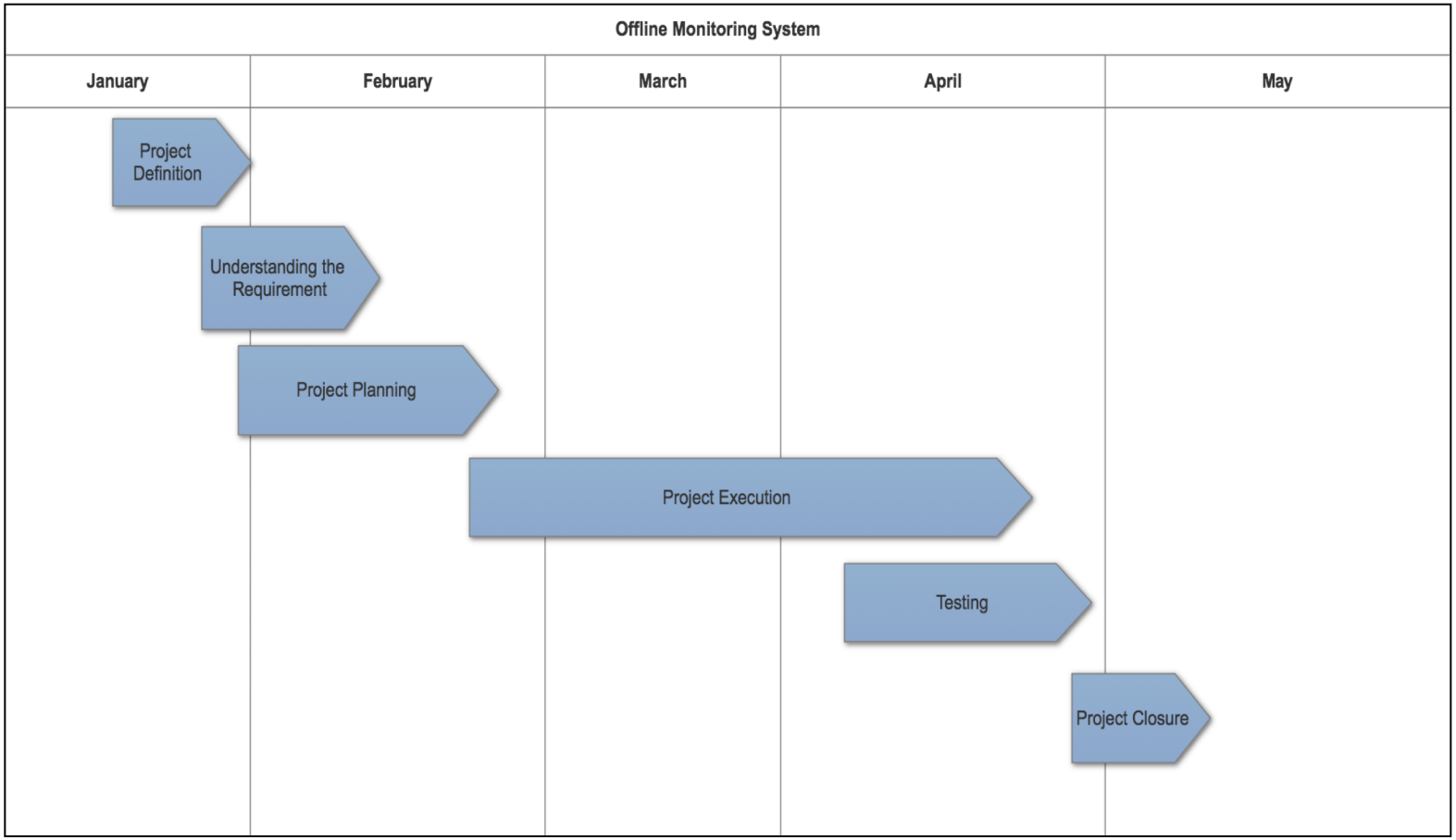

- The information of the previously logged data is stored in xml file format and is available for the user to view/download/delete.

# Communication

- The communication between the ESP32 and the mobile app is done by using the generic attributes.
- (GATT) that define a hierarchical data structure that is exposed to connected Bluetooth Low Energy (LE) devices.
- GATT server stores the data transported over the attribute protocol (ATT) and accepts ATT requests, commands and confirmations from the GATT client.
- The GATT server sends responses to requests and sends indications and notifications asynchronously to the GATT client when specified events occur on the GATT server.

# It's Time for the Demo

# Project Timeline



# Project Dynamics

- Hajera Nasreen:
  - Gathered information, working with Arduino code and modifying mobile application code.
- Vidhi Sharma:
  - Gathered information, working with ESP 32 connections and Xamarin.
- Divya Kamboj:
  - Gathered information, working with ESP 32 connections and Xamarin.
- Priyankaa Rajendran:
  - Gathered information, working with Arduino code and modifying mobile application code.


THANK
YOU!