TED UNIVERSITY

Computer Engineering Department

CMPE 491 Senior Project – High-Level Design Report - AgroAutomaTED

by

Korhan Deniz Akın

Ahmet Can Öztürk

Kaan BUDAK

07/01/2024

Table of Contents

1. Introduction

In terms of this "High-Level Design Report", a brief initial implementation demonstration will be provided which includes hardware-software relationships, the initial usage of the hardware components such as connection to the boards and receiving sensor data, used software architecture, database structure which enables the communication between the mobile application and the hardware system, ideas for the rest of the project such as machine learning models. In the introduction, the purpose of the system, the goals of the project, and the design goals are provided.

1.1 Purpose of the System

The AgroAutomaTED project is designed to propose a smart irrigation system that helps farmers and homeowners grow plants effectively. With this project, agriculture is backed by reducing excessive watering, monitoring the soil quality to see if it fits the desired conditions of the planted herb and providing users with a user-friendly application.

The system consists of 1 soil moisture sensor, humidity and temperature sensor (DHT11), to check the water level of the water tank proximity sensor (HC-SR04) was placed upper of the water tank, and a WiFi module (NodeMCU 1.0) was used on to collect data from the Arduino UNO board over Serial Communication and send data to the web server and the database server. Additionally, an Artificial Intelligence model shall be provided concerning some additional sensors such as the Soil NPK sensor which is the tester of soil efficiency with extremely high precision, pH sensor and salinity sensor. With data provided from given sensors, a beneficial machine learning model can be evaluated.

Users who are willing to get information about their soil, who have no time to irrigate their plants at home while they are away, and farmers who are willing to benefit from the AI model to choose which type of plant should they plant on the ground. With the proposed hardware system and the mobile application, irrigation over the application and soil condition monitoring are possible.

1.2 Design goals

The design of AgroAutomaTED is guided by a set of specific goals and objectives that address challenges related to water management in agriculture and aim to contribute to the overall goal of zero hunger. The system is designed to achieve the following key goals:

1.2.1 Efficient Water Use:

It aims to minimise water waste in irrigation by precisely measuring soil moisture levels and providing water only when necessary. To provide real-time data on soil conditions, the system uses the soil moisture sensor to make smart irrigation decisions.

1.2.2 Optimised Soil Moisture Management:

It implements a feedback loop between the soil moisture sensor and the irrigation system to prevent soil damage caused by overwatering or underwatering, as well as to maintain optimum soil moisture levels for plant growth.

1.2.3 Real-Time Monitoring and Notification:

It allows farmers and users to receive instant updates on critical conditions such as low water levels in the tank or abnormal soil moisture levels. It aims to develop a mobile application interface that receives and displays real-time data from the Arduino device and sends instant notifications to users when necessary.

1.2.4 User-friendly interface:

It features a mobile application interface design that is intuitive and accessible to users with varying levels of technical expertise. The aim is to provide a visually appealing and user-friendly application that provides clear information about soil moisture, water levels and irrigation activities status.

1.2.5 Artificial Intelligence-Driven Automation:

It is planned to apply artificial intelligence algorithms to automate irrigation decisions based on plant type, soil conditions and environmental factors. It is planned to manage the system with AI by integrating machine learning models that analyse historical data and current conditions into the mobile application to optimise irrigation planning.

1.2.6 Resource and Labour Efficiency:

It utilises automation features to reduce the need for constant manual monitoring and intervention in irrigation processes and minimise labour in irrigation management, allowing farmers to focus on other aspects of agriculture.

1.3 Definitions, acronyms, and abbreviations

DHT11: Temperature and Humidity Sensor

HC-SR04: Ultrasonic Distance Sensor

NodeMCU 1.0: WIFI Module

Soil NPK Sensor: Nitrogen, Phosphorus, and Potassium Sensor

Firebase: It is a platform that provides a set of backend cloud computing services and applications, we used it for database management.

1.4 Overview

This project outlines the AgroAutomaTED project, a comprehensive solution that aims to eliminate food shortages and contribute to the United Nations' Sustainable Development Goal of zero hunger. The project focuses on optimizing water management in agriculture through the integration of mobile application and Arduino-based physical devices.

2. Current software architecture

In the current system, a mobile application has started to be implemented based on the general outline which is specified in the Project Analysis Report. Flutter was used as a workspace to develop a cross-platform project. The Flutter project has integrated with Firebase. A page titled “My Plant” has been created, which pulls data from Firebase and displays the current information (water, soil humidity, temperature) about the plant. Also, a distance sensor (HC-SR04) has been placed to measure the tank water level.

Arduino IDE and sensors were used to get information. Integrates sensors including soil moisture level and water level sensors to gather real-time data. Data was collected and processed from the sensors to assess soil and water conditions. Also, implements logic for controlling the irrigation system based on data received from sensors. Ensures efficient water usage and prevents over-irrigation. On the other hand establishes communication with the mobile app transmitting data securely. Data will be stored in the cloud for historical analysis and remote monitoring. With this method, the system enables users to access data from multiple devices and locations. The mobile app processes incoming data and displays real-time information to the user. Notifications are sent to users based on predefined events or critical conditions. Users can interact with the mobile app to manually control irrigation settings.

3. Proposed software architecture

3.1 Overview

This part provides detailed information about the AgroAutomaTED project. First, the subsystem decomposition of the system is discussed. Then, hardware and software mapping and persistent data management will be demonstrated. Next, important data handling and global software control topics are discussed. Finally, all the boundary conditions that need consideration are reviewed.

3.2 Subsystem decomposition

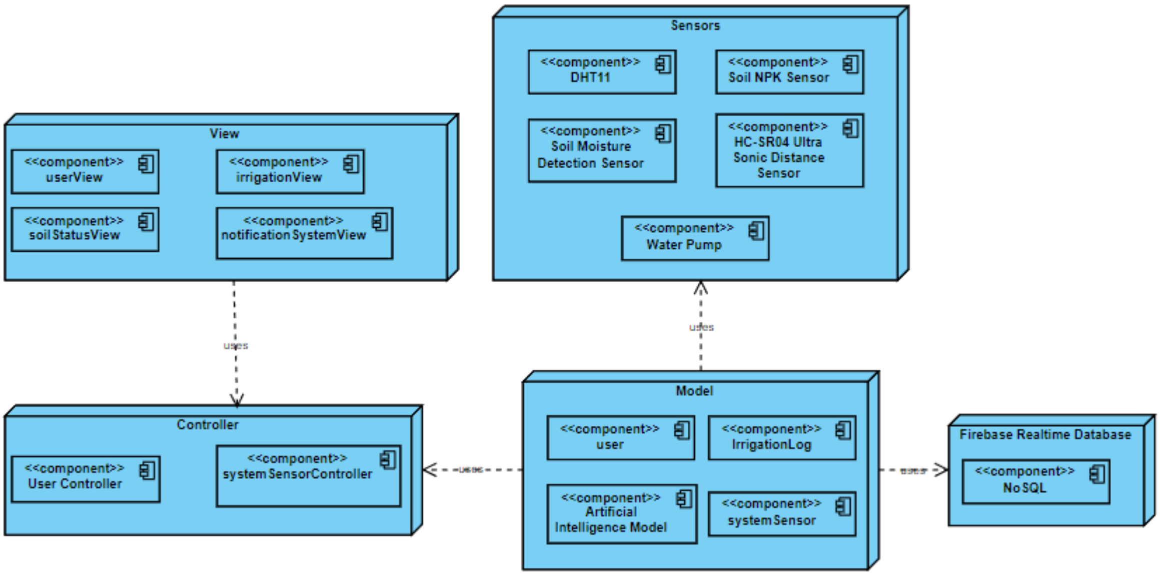

Diagram 1: System Decomposition Diagram

3.3 Hardware and software mapping

In the AgroAutomated project, there are two different paths for the transition of data as shown in the figure. Therefore, our software and hardware mappings are structured to handle this. In-depth mapping is specifically designed to ensure seamless integration between the Firebase Realtime Database, Flutter-based mobile applications for iOS and Android, and Arduino-based hardware equipped with essential sensors. The Firebase database serves as the main repository, enabling real-time data synchronization and communication with mobile applications and Arduino components. Through defined protocols and secure connections facilitated by the NodeMCU 1.0 WiFi module, sensor data is transmitted to the Firebase database, ensuring accurate monitoring and control functionalities. This integrated mapping strategy supports efficient data flow, user-friendly interaction, and system reliability, giving users useful perspectives and enhanced agricultural management capabilities.

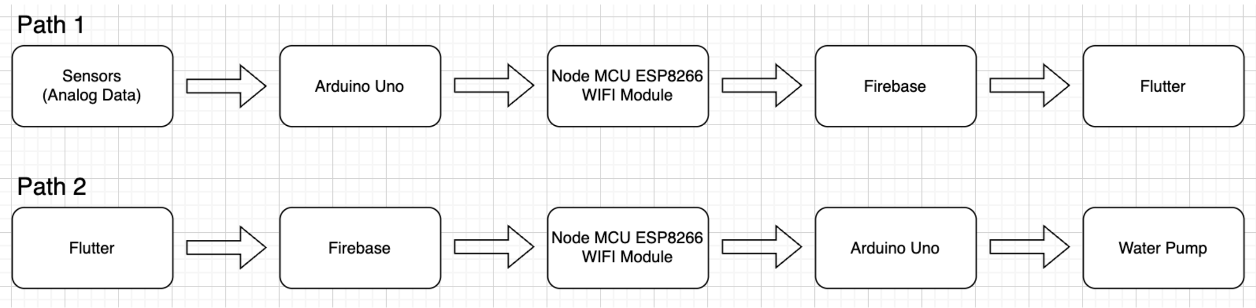

Diagram 2: Hardware and Software Diagram.

3.4 Persistent data management

Persistent data management was crucial for the AgroAutomaTED system. The reason for this is that sensor data is sent to the database continuously within a specific time interval. Therefore, it was not possible to store this amount of data in a hardware system or a database system as memory is not infinite. For this reason, it was planned to use sensor data to generate predictions and not to store them.

Together with the sensor data, irrigation log information and user data are managed. Excluding continuous sensor data, they are stored in the Firebase Realtime Database System. This is an applicable solution since retainment of data from the Firebase System to the Flutter application and also, sending data from the Wi-Fi module is relatively easy.

3.5 Access control and security

In the AgroAutomaTED project, it is extremely important to provide robust access control and security measures to protect sensitive agricultural data and preserve the integrity of the irrigation system, so the following components contribute to the access control and security aspects of the software architecture.

3.5.1 User verification:

The mobile application includes a user authentication system that requires users to sign in with secure credentials. This prevents unauthorized interventions by ensuring that only authorized individuals can access the system. For example, these one-time keys can be used to increase the security layer with one-time keys.

3.5.2 Secure Data Transmission:

The communication module of the Arduino device uses secure protocols to transmit data to the mobile application. Encryption mechanisms are implemented to protect the integrity and confidentiality of the information exchanged between the device and the application. A more secure data transmission is ensured by using HTTPS protocol or AES encryption in communication between the Arduino device and the mobile application.

3.5.3 Notification Security:

The notification system is designed to send secure push notifications to the user. Security measures are in place to prevent notification tampering and ensure users receive real alerts of critical events.

3.5.4 Device Authentication:

It is planned to have authentication protocols to verify the Arduino device's identity before communicating with the mobile application. This prevents unauthorized devices from accessing the system. For example, HMAC can be used to authenticate the device. HMAC generates an authentication code by hashing the data with a private key. This verification code is used to authenticate the device.

3.6 Global software control

The AgroAutomaTED project requires a global software control mechanism to facilitate centralized monitoring and management of the irrigation system. The software architecture includes the following elements to ensure global control.

3.6.1 Cloud Integration:

Optionally, the system allows cloud integration, allowing users to access the software globally. This facilitates remote monitoring and control, allowing farmers or agronomists to manage the irrigation system from anywhere in the world.

3.6.2 Central Data Storage:

The software architecture includes a central data storage system within the Firebase Realtime Database. This ensures that all relevant data, including soil moisture levels, water storage levels and historical analysis, is stored in a central location for easy access and analysis.

3.6.3 Remote Accessibility:

The mobile app is designed for remote accessibility, allowing users to monitor and control the irrigation system remotely. This enables both the status of the crops to be checked, especially when they are away from the field, and the system to be managed remotely by giving the necessary commands through the system.

3.7 Boundary conditions

3.7.1 Initizilation

To initiate the utilization of the AgroAutomaTED system, the application must be downloaded by users on their iPhone or Android devices from either the App Store or Google Play. Once the application has been successfully downloaded and configured, users need to establish an account. During this process, an email address and password must be entered for registration purposes. Following account creation and login procedures, the Arduino device is to be connected to the user's account using both WI-FI connectivity and Device ID for seamless integration. That enables users to know when a plant is irrigated and to get parameter values, which are provided by sensors. Remember, the internet is essential to use all the features and get updates on your plants.

3.7.2 Termination

Once the AgroAutomaTED application is launched, and a user logs in, the system stays active, facilitating the transmission of notifications. The application will persist in the background, retaining cached data, even if users do not close the app tab until it is fully terminated. Upon logging out, notifications cease, and history data is deleted.

3.7.3 Failure

In the AgroAutomaTED system, both the mobile app and the hardware with Arduino can face issues that need special fixes. First, the mobile app needs a good internet connection to work. If there's no stable internet, users can't use the app until they get a good connection. On the hardware side (Arduino Uno), it's crucial to have steady power. If the power goes out or acts up, it can mess with the sensors and how the system works. Plus, sometimes the sensors might not work perfectly due to things like wear and tear or software problems. To keep things running smoothly, it's important to check and fix these issues regularly. Overall, keep an eye out for these problems and have plans to handle them.

4. Subsystem services

4.1 Subsystem Services and Descriptions:

The ones that are mentioned in this section are considered the major components that our system benefits. They are treated as the gateway to the analog world in terms of our hardware structure. These sensors combined with some useful boards are there to collect beneficial information about given properties such as soil condition, temperature of the current environment together with the humidity sensor, a proximity sensor and a tiny water pump. These sensors incorporate an AVR microcontroller named "Arduino UNO". Once this data is assembled into this microcontroller, there is a Wi-Fi module to send data to the corresponding database server. All of these are mentioned in the corresponding section in detail.

4.1.1 DHT11 Humidity and Temperature Sensor

To talk about them one by one, let us start with "DHT11 Humidity and Temperature Sensor" which is demonstrated in Figure x. This sensor is responsible for getting the temperature data from the environment together with its humidity. The corresponding sensor is tiny, working very accurately and fast.

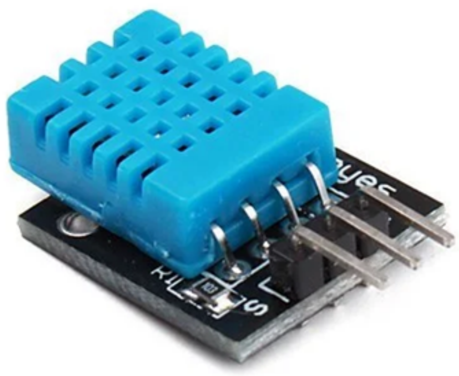

Figure 1: DHT11 Humidity and Temperature Sensor.

Data from the environment is received with the help of the "dht11.h" library. Which includes "temperature" and "humidity" attributes of the dht11 object declared from the given library.

4.1.2 Soil Hygrometer Moisture Detection Sensor

Another useful sensor was the "Soil Hygrometer Moisture Detection Sensor". The Soil Hygrometer Moisture Detection Sensor measures moisture in soil continuously. By putting the conductive tip of the sensor into the soil. This data is used to inform the user through the mobile app such that all situations are covered about the soil moisture. For example, when the level is critically low there is: "Water level is critically low! Please water the plants.", when the level is balanced: "Soil moisture is at a moderate level.", lastly when the water level is acceptable: "Soil moisture is sufficient." messages are sent to the related database and then to the user. The aforementioned sensor is shown in Figure x+1.

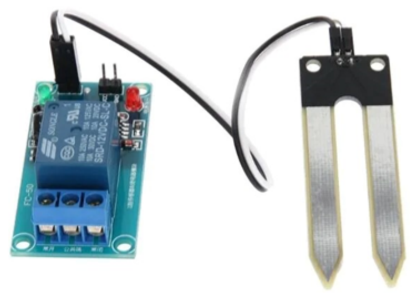

Figure 2: Soil Hygrometer Moisture Detection Sensor.

4.1.3 Mini Submersible Water Pump

To water the soil, water was crucial. This component has a pumping mechanism which takes water from the tank and sends it to the soil through a pipe. Combined with the Arduino UNO board and the Wi-Fi it is possible to generate this pump from the mobile application. The related component is demonstrated in Figure x+2.

Figure 3: Mini Submersible Water Pump. The image is taken from

4.1.4 HC-SR04 Ultrasonic Distance Sensor

As our system consists of a water tank, its water level must be looked after. It was done so by using "HC-SR04 Ultrasonic Distance Sensor". The HC-SR04 Ultrasonic Distance Sensor uses ultrasonic sound waves to measure the distance between the sensor and an object or in our case the water. It operates by sending out an ultrasonic pulse and calculating the time it takes for the pulse to bounce back after hitting an object. This time measurement is then used to determine the distance. The corresponding image is illustrated in Figure x+3.

Figure 4: HC-SR04 Ultrasonic Distance Sensor. The image is taken from

4.1.5 Arduino UNO Board

Finally, to talk about the main responsibilities of computational power and Wi-Fi connectivity, the Arduino UNO board and NodeMCU ESP8266 Development Card are worth mentioning.

Arduino UNO Board (can be seen in Figure x+4) uses an ATmega328P microcontroller which includes a sufficient number of GPIO pins. This is the reason that it was decided to use such a board rather than only using the NodeMCU module (which also consists of a built-in microcontroller). With its 6 analog pins (https://www.circuito.io/blog/arduino-uno-pinout/), a lifesaver for a system with a lot of sensors, this board can be considered as the bridge between sensors and corresponding Wi-Fi module.

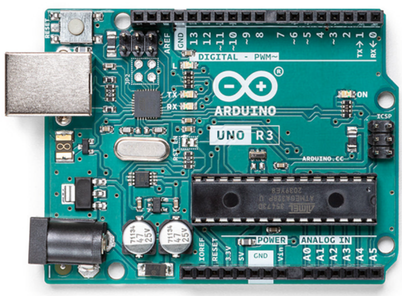

Figure 5: Arduino UNO Board.

4.1.6 NodeMCU ESP8266 Development Card

Lastly, the connection provided to the Firebase is achieved with the help of the “NodeMCU ESP8266 Development Card”. The NodeMCU ESP8266 Development Card (can be seen in Figure x+5) consists of an ESP8266 Wi-Fi module that provides wireless connectivity. Even though it consists of GPIO pins and analog pins the amount of them was not sufficient to obtain data from sensors as mentioned previously. The data transfer was achieved through the “USART” protocol between the Arduino UNO board and the NodeMCU Module.

Figure 6: NodeMCU ESP8266 WiFi Module.

4.2 Connections and Dependencies:

In the AgroAutomaTED project, various subsystems collaborate to create a harmonious and effective irrigation management system. The connections and dependencies between these subsystems are crucial for the proper functioning of the overall solution.

4.2.1 Mobile Application - Arduino Device Connection:

The Mobile app relies on the Arduino device to provide real-time data on soil moisture levels and water tank status. The mobile application communicates with the Arduino device via established protocols to receive sensor data and send control commands.

4.2.2 Arduino Device - Soil and Water Sensors:

The Arduino device depends on soil moisture and water level sensors for accurate data on environmental conditions. The Arduino device integrates soil and water sensors to collect data, allowing informed irrigation decisions to be made.

4.2.3 Mobile Application - Cloud Storage:

The mobile application may rely on cloud storage for expanded data storage and analysis capabilities. The mobile app can provide global accessibility and analytics by securely communicating with cloud servers to upload and retrieve historical data.

4.2.4 Arduino Device - Mobile Application - User Interface:

The mobile app's user interface depends on data from the Arduino device for real-time monitoring. The user interface of the mobile application provides users with information about soil and water conditions by displaying information received from the Arduino device.

4.2.5 Mobile Application - Artificial Intelligence Algorithms:

AI-driven automation features in the mobile app depend on accurate data from the Arduino device. The mobile app processes data using artificial intelligence algorithms to automate irrigation decisions based on plant, soil and environmental conditions.

4.3 Interfaces

The AgroAutomaTED Project has two main communication paths, which are Sensors to Mobile Application User Interface and Mobile Application User Interface to Water Pump.

4.3.1 Path 1: Sensors to Mobile Application User Interface

4.3.1.1 Analog Data:

The process initiates with the collection of analog data from various sensors, such as the soil moisture sensor and DHT11 humidity and temperature sensor. The analogRead() function provides to Analog-to-Digital Converter (ADC) process.

4.3.1.2 Arduino UNO:

Arduino employs the analogRead() function to convert the received analog signals into digital format. Then, using the USART method that allows two-way communication, it sends this data to the Node MCU ESP8266 WiFi module.

4.3.1.3 Node MCU ESP8266 WiFi Module:

The Node MCU module receives the processed digital data and establishes a two-way communication link with the Web Connection, ensuring seamless data transmission.

4.3.1.4 **Firebase**:

In Firebase, the Realtime Database service receives data from the NodeMCU ESP8266 WiFi Module via Web Connection. A Realtime database provides synchronization data in real-time across all clients or devices connected to the database. Also, data is stored systematically and organized in a structured way to make pulling data to the UI effortless.

4.3.1.5 **Flutter**:

Finally, the Flutter-based mobile application retrieves and displays the data from Firebase. It provides users with real-time alerts and irrigation services based on the sensor data supported by Machine Learning (AI Model).

4.3.2 Path 2: Mobile Application User Interface to Water Pump

4.3.2.1 **Flutter**:

Flutter provides the main screen or face of our mobile app. It shows you information about your plants and lets you give them water when they need it. Users control the irrigation system's working condition, this data will be sent to Firebase.

4.3.2.2 **Firebase**:

The real-time database gets data from the user interface, stores it, and then sends it to the Node MCU ESP8266 WiFi Module via Web Connection.

4.3.2.3 **Node MCU ESP8266 WiFi Module**:

With receiving control signals, the Web Connection interfaces with the Node MCU ESP8266 WiFi module, initiating USART communication with the Arduino UNO board to relay the signals.

4.3.2.4 **Arduino UNO**:

The Arduino UNO board converts the processed digital data back to analog format suitable for actuation tasks such as activating the water pump, based on user-defined parameters and control signals.

4.3.2.5 **Water Pump**:

The converted analog data is used to turn off or turn off the water pump, ensuring the irrigation needs of the plant requirements.

5. Appendix

Task Name	Story	Sprint Ready	Priority	Status	Story Points	Assigned to Sprint
1. Sensor Integration	Integrate soil moisture and water level sensors with Arduino device.	Yes	High	In Progress	5	Sprint 1
2. Mobile App UI Design	Design the user interface for the mobile app.	Yes	Medium	To Do	3	Sprint 2
3. Data Encryption Implementation	Implement data encryption between Arduino and mobile app for secure communication.	Yes	High	To Do	8	Sprint 1
4. Cloud Integration	Explore and integrate cloud storage for historical data and analytics.	Yes	Medium	To Do	5	Sprint 3
5. Notification System	Develop push notification system for alerting users about critical events.	Yes	High	To Do	5	Sprint 2
6. AI Algorithm Implementation	Implement AI algorithms for automated irrigation based on sensor data.	Yes	High	To Do	8	Sprint 3
7. Testing - Arduino Device	Conduct testing for Arduino device functionality and reliability.	Yes	High	To Do	5	Sprint 2
8. Testing - Mobile App	Perform testing for the mobile app, ensuring usability and bug fixing.	Yes	Medium	To Do	5	Sprint 3
9. Documentation	Document system architecture, user guides, and API documentation.	Yes	Low	To Do	3	Sprint 4
10. Final System Testing	Conduct comprehensive testing for the entire system.	No	High	To Do	8	Sprint 4

6. Glossary

Arduino UNO Board: A microcontroller board used in the AgroAutomaTED system to collect data from various sensors and control the irrigation process.

NodeMCU 1.0: A Wi-Fi module that facilitates communication between the Arduino UNO board and the Firebase Realtime Database, enabling data transmission and control functionalities.

Firebase: A cloud-based platform used for database management in the AgroAutomaTED project. It ensures real-time data synchronization and facilitates the storage and retrieval of data.

DHT11: A temperature and humidity sensor utilized in the system to monitor environmental conditions, specifically temperature and humidity levels.

HC-SR04: An ultrasonic distance sensor is employed to measure the water level in the tank, ensuring optimal water supply for irrigation purposes.

Soil NPK Sensor: A sensor that measures the levels of Nitrogen, Phosphorus, and Potassium in the soil, providing insights into soil fertility and efficiency.

Flutter: A cross-platform framework used to develop the mobile application interface for the AgroAutomaTED system, facilitating real-time data visualization and user interaction.

AI Model: Artificial intelligence algorithms integrated into the system to automate irrigation decisions based on plant type, soil conditions, and environmental factors.

Subsystem Decomposition: The division of the AgroAutomaTED system into smaller subsystems to facilitate design, development, and integration tasks, ensuring efficient functionality and performance.

Mobile Application: An application developed using Flutter and integrated with Firebase, allowing users to monitor soil moisture, water levels, and irrigation activities, receive real-time notifications, and control irrigation settings remotely.

Sensors: Devices integrated into the AgroAutomaTED system, including DHT11, HC-SR04, Soil NPK Sensor, and Soil Hygrometer Moisture Detection Sensor, to collect data on environmental conditions, soil moisture levels, and water levels for optimal plant growth and water management

This glossary provides definitions for key terms and components related to the AgroAutomaTED project, facilitating a better understanding of the system's architecture, functionalities, and objectives.

7. References

- World Bank. (2023). World Bank Group. Retrieved from https://www.worldbank.org/
- United Nations. (2023). Sustainable Development Goals. Retrieved from https://sdgs.un.org/
- United States Department of Agriculture. (2023). USDA. Retrieved from https://www.usda.gov/
- Flutter. (n.d.). Flutter - Beautiful native apps in record time. Retrieved from https://flutter.dev/
- Firebase. (n.d.). Firebase - Google. Retrieved from https://firebase.google.com/
- Robotistan. (n.d.). Robotistan - Electronic Components & Robot Parts. Retrieved from https://www.robotistan.com/
- Arduino. (n.d.). Arduino - Home. Retrieved from https://www.arduino.cc/
- Circuito. (n.d.). Arduino Uno Pinout Guide. Retrieved from https://www.circuito.io/blog/arduino-uno-pinout/
- Loox. (n.d.). Arduino Uno Rev3 - Store. Retrieved from https://store.arduino.cc/products/arduino-uno-rev3#looxReviews
- NodeMCU. (n.d.). NodeMCU - Lolin ESP8266 Development Board. Retrieved from https://www.robotistan.com/nodemcu-lolin-esp8266-gelistirme-karti
- Robotistan. (n.d.). Mini Dalgıç Pompa 6V 120 Litre/Saat. Retrieved from https://www.robotistan.com/mini-dalgic-pompa-6v-120-litresaat
- Robotistan. (n.d.). HC-SR04 Ultrasonik Mesafe Sensörü. Retrieved from https://www.robotistan.com/hc-sr04-ultrasonik-mesafe-sensoru
- Robotistan. (n.d.). DHT11 Isı ve Nem Sensörü Kart. Retrieved from https://www.robotistan.com/dht11-isi-ve-nem-sensoru-kart
- Robotistan. (n.d.). Nem Algılama Nem İzleme Modülü. Retrieved from https://www.robotistan.com/nem-algilama-nem-izleme-modulu
- Robotistan. (n.d.). Soil NPK Sensor - Analogue. Retrieved from https://www.robotistan.com/soil-npk-sensor-analog