Green House Manager

Cover Page

**Table of Contents**

**Team Roles**

**A person wearing glasses

Description automatically generated with low confidence**A person wearing a hat

Description automatically generated with medium confidence

**HIMANSH ARORA Tolani Animasahun**

**(Database) (UI/Front End)**

**Hardware**

Raspberry pi model zero

Soil moisture sensors

Jumper cables

Insulation tape

Large LED light

DHT11  - humidity sensor

Temperature sensor

Power Bank

**Data**

**Data collected by the device** :

Device identification (device ID, Raspberry Pi model and version of   
software)

Time log of when sensor data was registered

temperature value in Celsius, humidity percentage from DHT11 sensor, soil moisture

percentage value

**Third party APIs**

Device will receive weather data in JSON format  -

Local temperature in Celsius (int)

Local humidity as percentage (int) sourced from the weatherapi.com.

**Data Storage and Processing**

Sensor data will be stored in a database. The database will be a MySQL database.

Data Storage

Why Store Data?

Today's businesses rely heavily on data. As a company grows, so does the amount of data it collects. And given the importance of data in today's business world, having a proper data storage system is a must-have for any company. This does not, however, imply establishing an old-fashioned paper-based data storage system. The current era is the digital era, which necessitates the implementation of appropriate digital data storage systems for improved operations, accessibility, and data security.

A Database is a perfect means to store and access any data electronically.

Why Database?

* Databases are capable of storing a vast amount of records in an effective manner, they take very less space.
* It's simple to add new data and amend or delete existing data in a database.
* It is extremely easy and quick to locate information.
* Data can simply be sorted and grouped, for example, into alphabetical order, numerical order, group by date, etc.
* Databases have multi-access which means they can be accessed by multiple people at the same time.
* The scope of accessibility can also be controlled by applying different security levels.
* The security of electronic data storage is obviously superior to that of paper storage.

For this project, a Relational Database will be used.

Why Relational Database?

* The data can be simply organized into categories.
* The data is well-organized, has a clear meaning, and is really simple to navigate.
* Relational Databases have a broad eco-system and built-in data integrity.
* Specific users can have direct access to data in tables within an RDBMS.
* SQL, the major query language used with relational databases, makes it simple for users to run complex queries.

A MySQL Database, which is a really simple to operate Relational Database, will be used for this project.

Database Scenario

A user will have one and only one login, however a login system will have many users (1:M). One plant can be planted by many users and a user may have multiple plants(M:N), hence the junction table ‘user\_plant’. A user will have one raspberry pi and vice-versa(1:1). If one user has multiple plants then their raspberry pi can be used to display conditions of more than one plant(1:M).

Entity-Relationship Diagram

Diagram

Description automatically generated

Sample Data

**user**: Primary Key = ‘user\_id’ and Foreign Key = ‘raspi\_id’

**plants**: Primary Key = ‘plant\_name’ and no Foreign Key

**user\_plant**: Primary Key = ‘plant\_id’ and Foreign Key = ‘user\_id’, ’plant\_name’

**raspberry\_id**: Primary Key = ‘raspi\_id’ and Foreign Key = ‘plant\_id’

**login**:  no Primary Key and Foreign Key = ‘user\_id’

Table

Description automatically generated                    Table

Description automatically generated

Table

Description automatically generated

Table

Description automatically generated

Table

Description automatically generated

Database Schema

Table

Description automatically generated with medium confidence

Sample Queries

Following are some of the queries that can be used to access or alter data from the database:

* Display the user ID of a user using their registered username

= select user\_id from login where username = “peter123”;

* Display the plants by their plant ID owned by a particular user

= select plant\_id from user join user\_plant using (user\_id) where name = “Bryan”;

* Get the name of the user who live in either Dundalk or Drogheda and own either tomatoes or cucumbers

= select name from user join user\_plant using (user\_id) where (location like "%Dundalk%" OR location like "%Drogheda%") and (plant\_name = "tomato" OR plant\_name = "cucumber");

* User Mike has moved their house from Galway to Dublin

= update user set location = ”Dublin” where name = “Mike”;

**SECURITY**

Database Security

- Information will be encrypted

- Permissions to write and read will be assigned to users who access the database

- Strong passwords will be used

When transferring data, secure protocols HTTPS and SFTP will be used

Data will be encrypted with SHA256 encryption

**User Interface**

Graphical user interface, application

Description automatically generated

This is our login page for users to log into our app.

They enter their username and password.

Graphical user interface, text, application, chat or text message

Description automatically generated

This is the main page for our user.

This page shows the statistics of the greenhouse overall in terms of Humidity, temperature, and moisture.

From this page the user can see what plants are in their green house and can also add some from an existing database of plants we’ve created.

Certain features have been coloured coded, for users whose strongest language may not be English. E.g., Red box for what plants needs to be watered urgently and orange for what is close after that.

In the “plants to water” section of our paper prototype we plan on changing it from the user needs to water that plant in “30 minutes” to “this plant is below the ideal soil moisture by x amount”.

Diagram

Description automatically generated

This page shows the user what plants they have in their greenhouse. This page displays to the user the soil moisture of the plant when it was watered and the day it was planted.

There is a watering button at the top colour coded in blue so the user can click on it to water their plants. Originally, we planned on having the plants in the greenhouse grouped but we may go back to that idea depending on the scale of the greenhouse we are able to build.

A drop down menu full of the plants in their garden.

Diagram

Description automatically generated

This page allows the user to add plants to their My garden page. The plants from this page will be taken from a database we built full of basic plants so users who haven’t much experience in gardening can find different plants and have the ideal conditions for them right at their fingertips. The information displayed for the user will be the Ideal humidity for the plant, the temperature, and the soil moisture. The add button has been coloured green so users can associate it with buying/adding. Once the user clicks “add” the plant will be added to the my garden page.

Diagram

Description automatically generated

This page is another way of adding plants for users who are more experienced in gardening. Users can add plants on this page that they wouldn’t find in our database like for example a type of exotic plant. The user would need to know the ideal conditions for that specific plant themselves since it wouldn’t be in our database. We had the idea of trying to push these plants to our public database which all users would have access to therefore continuing expanding our ever-growing database.

**User Personas**

**Graphical user interface, website

Description automatically generated**

**Graphical user interface, text, application, website

Description automatically generated**

**Testing**

**User Testing**

The user testing will be done with multiple users. The users we will try and focus our testing on will be experienced gardeners so our user at the garden centre, a classmate who has no experience of gardening and a user who wouldn’t have much experience with technology, from there I believe we can get an accurate representation of what we can add and fix in our project.

We talked to a potential user who works in the black’s garden centre Drogheda as our main tester. They told us it was a great idea and provided us with feedback with the types of data we could display and show the user.

We will also test the UI to make sure it is clear concise and user friendly for all potential users.

All users found it relatively straight forward running through our UI paper prototype. They found the information given to be straight to the point and understandable. The other bits of feedback we received from the user testing of the UI was, that the custom plants page can be added into a better location instead of within the search bar to find it. Not many users would know to find it there unless by pure accident. Another criticism was the colour of the red background with black text over it. This was mainly a concern with our user who isn’t familiar with technology as they were an elderly person, they found it hard to make out what the text was saying. They know it was important piece of information due to the red background behind it.

**Hardware Testing**

We plan on testing the physical device in a small-scale green house with a few plants and from there we will see what the possibilities of expansion are.

We will be testing a multitude of things from the sensors to check how to collect different variables needed to give the user the data they need.

Out of the hardware needed for the project I have managed to successfully finish testing the DHT11 Humidity Sensor.