



Communications and Computers Engineering

Faculty of Engineering

Mansoura University

Graduation Project Report 2022/2023

Project Title : Smart Greenhouse

Team Members:

Name	ID
Fares Ali Mohamed	800166887
Rawan Emad El-Ghali	800136730
Mahmood Farahat Ali	800166997
Hams Ahmed Zahran	800139088
Omar Khaled Amin	800168618
Zeinab Tharwat Shouman	800140311
Aya Mostafa Abo Essa	800136789
Heba Adel Bahy Eldien	800139106
Marc Atef Habeeb	800136956
Eslam El-Saied El-Shafie	800136668

Supervisors:

DR. John Fayez Wanis

ENG. Ahmed Saad Radwan

Table of Contents

<u>Abstract</u>	4
<u>Chapter 1: Introduction</u>	5
• 1.1: Motivation	5
• 1.2: Novelty	5
• 1.3: Technical Context	6
• 1.4: Structure of Report	7
<u>Chapter 2: Background</u>	8
• 2.1: Hardware	8
• 2.2: Database	8
• 2.3: Brand Identity	9
• 2.4: UX	9
• 2.5: UI	10
• 2.6: Flutter	11
• 2.7: Communication Protocols	12
• 2.8: Web Application	12
• 2.9: Security	13
• 2.10: IOT	13
• 2.11: Related work done before	13

<u>Chapter 3: Design, Implementation and Results</u>	14
● 3.1: Overview	14
● 3.2: Database	15
● 3.3: Dashboard BI.....	17
● 3.4: Brand Identity	19
● 3.5: UX	22
● 3.6: UI	24
● 3.7: Hardware	30
● 3.8: Communication Protocols	33
● 3.9: Flutter	33
● 3.10: Web Application	41
● 3.11: Security	47
● 3.12: Business Model	50
<u>Chapter 4: Conclusion and Future Work</u>	54
● 4.1: Future work	56
● 4.2: References	56
<u>Acknowledgment</u>	57

Abstract

A smart greenhouse is a type of greenhouse that utilizes advanced technology to optimize growing conditions for plants. This can include automated systems for controlling temperature, humidity, lighting, irrigation, and fertilization. Smart greenhouses also often include sensors and data analysis tools to monitor and adjust these conditions in real-time, as well as remote monitoring and control capabilities. The goal of a smart greenhouse is to improve crop yields, reduce water and energy use, and improve the overall efficiency of greenhouse farming.

Chapter 1: Introduction

1.1 Motivation

The motivation for creating a smart greenhouse is to optimize the growing conditions for plants in order to increase yields and improve the overall efficiency of the greenhouse operation. This can be achieved by automating tasks such as temperature and humidity control, light intensity, and irrigation, as well as monitoring the environment and the health of the plants in real-time. Additionally, a smart greenhouse can also help to reduce energy consumption and decrease the environmental impact of traditional greenhouses.

There are several reasons to create a smart greenhouse, including:

1. Increased yields: By automating and optimizing the growing conditions, smart greenhouses can help to increase crop yields and improve the overall efficiency of the greenhouse operation.
2. Resource management: Smart greenhouses can help to conserve resources such as water and energy by automating systems such as irrigation and temperature control.
3. Climate control: Smart greenhouses can help to maintain a stable and optimal growing environment for plants, regardless of the outside weather conditions.
4. Plant monitoring: Smart greenhouses can monitor the health and growth of plants in real-time, allowing for early detection of issues and prompt action to address them.
5. Cost savings: By automating many of the tasks involved in greenhouse operations, smart greenhouses can help to reduce labor costs and increase overall efficiency.
6. Climate change: Smart greenhouses can help to reduce the environmental impact of traditional greenhouses by increasing the energy efficiency and reducing the carbon footprint.

1.2 Novelty

Our smart greenhouse is distinguished from others by its advanced technology and automation features. It utilizes sensors and monitoring systems to track and optimize the growing conditions for the plants inside. This includes monitoring temperature, humidity, soil moisture, and light levels, as well as controlling the irrigation and ventilation systems. The data collected by these sensors is analyzed by our proprietary software to make adjustments and recommendations for

optimal plant growth. Additionally, our smart greenhouse is also equipped with a remote monitoring and control system, which allows for real-time monitoring and adjustments to be made remotely, via a mobile app or web portal. This allows for greater convenience and flexibility for the user, as well as the ability to monitor multiple greenhouses from one location. Overall, our smart greenhouse is designed to provide the optimal environment for plants to thrive, while also making it easy and convenient for the user to manage and maintain.

1.3 Technical Context

The project consists of the use of Firebase real time database, Flutter, dart, provider, MS Power BI, Canva, Adobe Illustrator, Adobe Photoshop and Figma.

Firebase is a mobile and web application development platform that provides a set of tools and services to help developers build high-quality apps. It includes features such as authentication, storage, and real-time databases. The Firebase Real-time Database is a cloud-hosted NoSQL database that allows developers to store and sync data between users in real-time. It provides a simple and flexible API that allows for easy integration into web and mobile apps and supports offline access to data. With its powerful features and ease of use, Firebase and the Real-time Database have become a popular choice for many developers building real-time, collaborative applications. Flutter is an open-source framework by Google for building beautiful, natively compiled, multi-platform applications from a single codebase. Dart is a programming language designed for client development such as for the web and mobile apps. It is developed by Google and can also be used to build server and desktop applications. Adobe xd is a vector-based user experience design tool for web apps and mobile apps, developed to help preview the result of work directly on mobile devices. Adobe XD enables website wireframing and creating click-through prototypes.

- **Operating System: Windows 10, Windows 11 & Debian GNU Linux.**
- **Database: Firebase real time database**
- **Mobile application: Flutter 3.0.0, Dart, provider.**
- **UI, UX & any visuals: Adobe Illustrator, Adobe Photoshop, Canva and Figma.**
- **Dashboard: MS Power BI**
- **Hardware**
- **Additional service : Notion, Google Drive and One Drive.**

- **Operating System:** Windows 10, Windows 11 & Debian GNU Linux.
- **Database:** Firebase real time database
- **Mobile application:** Flutter 3.0.0, Dart, provider.
- **UI, UX & any visuals:** Adobe Illustrator, Adobe Photoshop, Canva and Figma.
- **Dashboard:** MS Power BI
- **Hardware**

1.4 Structure of Report

- **Chapter 1** is an introduction to the project. It leads to the motivation of the project and analyses the novelty of the methods.
- **Chapter 2** is the background of the project. it introduces the concept of Hardware & Database, UX, UI, Dashboard and BI, flutter and website ending with the related work done before our project.
- **Chapter 3** is the design, implementation, and Results of the project, which gives the overall review of the whole system, and the way to realize it.
- **Chapter 4** is the conclusion and further work of the project, which concludes the whole work and further work that aims to optimize the project.

Chapter 2: Background

2.1 HARDWARE

Hardware refers to the physical components of a computer system or other electronic device. These components include things like the central processing unit (CPU), memory (RAM), storage devices (hard drive or solid-state drive), input/output devices (keyboard, mouse, monitor), and peripherals (printers, scanners, etc.). Hardware can also refer to the physical components of other types of technology, such as smartphones, televisions, and gaming consoles. Hardware is the physical aspect of technology, while software is the non-physical or digital aspect of technology. Together, hardware and software make up a complete system or device that can perform a specific task or set of tasks.

2.2 DATABASE

- A database is a collection of data that is organized in a specific way, allowing for efficient storage, retrieval, and manipulation of the data. The data is typically stored in tables, which consist of rows (records) and columns (fields). Databases can be used for a wide variety of purposes, such as storing customer information for a business, tracking inventory for a warehouse, or keeping track of financial transactions. They can be accessed by a variety of programs and applications, using a language called SQL (Structured Query Language).
- Firebase is a comprehensive mobile and web application development platform that provides a wide range of tools and services to help developers create high-quality, feature-rich apps. One of the key features of Firebase is its real-time database, which allows developers to easily store, retrieve, and sync data in real-time across multiple devices and users. The real-time database uses a cloud-based NoSQL data model, which makes it easy to scale and manage large amounts of data. Additionally, Firebase provides powerful security features, including user authentication and access controls, to help keep data safe and secure. Overall, Firebase is an essential tool for any developer looking to build fast, responsive, and feature-rich apps that can handle large amounts of data and users.

2.3 Brand Identity

Brand identity refers to the visual and verbal elements that make up a brand's image and reputation. This includes things like the company's logo, color scheme, messaging, and overall aesthetic. A strong brand identity helps to establish a company's unique identity and differentiate it from competitors, making it more memorable and recognizable to consumers. It also helps to create a sense of trust and loyalty with customers. In summary, Brand Identity is the visual and verbal elements that make up a brand's image and reputation and it helps to establish a company's unique identity and differentiate it from competitors, making it more memorable and recognizable to consumers, and helps to create a sense of trust and loyalty with customers.

2.4 UX

The UX process typically involves several stages, including:

- **Research:** This involves gathering information about the users and their needs, as well as the context in which the product or service will be used. This can include user interviews, surveys, focus groups, and other methods of gathering user feedback.
- **Design:** This involves creating prototypes, wireframes, and other design elements that will be used to build the final product or service. The goal is to create a design that is intuitive, easy to use, and aesthetically pleasing.
- **Testing:** This involves testing the product or service with users to gather feedback and make improvements. This can include usability testing, where users are asked to perform specific tasks with the product or service, and user acceptance testing, where users are asked to evaluate the overall experience of using the product or service.
- **Implementation:** This involves building the final product or service based on the design, and making any necessary changes based on the feedback gathered during testing.
- **Evaluation:** This involves gathering feedback from users after the product or service has been released, and using this feedback to make ongoing improvements to the user experience.

2.5 UI

The user interface (UI) process is the process of designing and developing the visual and interactive elements of a product or service that users interact with. It involves creating a visual design that is aesthetically pleasing and easy to use and implementing this design in the form of a user interface.

The UI process typically involves several stages, including:

- **Research:** This involves gathering information about the users and their needs, as well as the context in which the product or service will be used. This can include user interviews, surveys, focus groups, and other methods of gathering user feedback.
- **Design:** This involves creating prototypes, wireframes, and other design elements that will be used to build the final product or service. The goal is to create a design that is intuitive, easy to use, and aesthetically pleasing.
- **Testing:** This involves testing the product or service with users to gather feedback and make improvements. This can include usability testing, where users are asked to perform specific tasks with the product or service, and user acceptance testing, where users are asked to evaluate the overall experience of using the product or service.
- **Implementation:** This involves building the final product or service based on the design, and making any necessary changes based on the feedback gathered during testing.
- **Evaluation:** This involves gathering feedback from users after the product or service has been released and using this feedback to make ongoing improvements to the user interface.

2.6 FLUTTER

Flutter is an open-source framework by Google for building beautiful, natively compiled, multi-platform applications from a single codebase. Flutter code compiles to ARM or Intel machine code as well as JavaScript, for fast performance on any device. Flutter is supported and used by Google, trusted by well-known brands around the world, and maintained by a community of global developers. Flutter is powered by Dart, a language optimized for fast apps on any platform. Flutter uses firebase which is a Backend-as-a-Service (BaaS) app development platform that provides hosted backend services such as a Realtime database, cloud storage, authentication, crash reporting, machine learning, remote configuration, and hosting for your static files.

The application used to create flutter:

VS Code is a lightweight editor that has great Flutter support, good extensibility and sees widespread use across the developer community.

Android studio which provides:

- | | |
|--|-----------------------------------|
| 1-Faster Coding and Quick Iteration. | 2-Fast and Feature-rich Emulator. |
| 3-Boost Your Self-belief while Coding. | 4-Robust Testing Mechanisms. |

2.7 Communication Protocols

Global System for Mobile Communications (GSM) is a widely-used mobile telecommunications technology that was first developed in Europe in the late 1980s and early 1990s. It is the most popular standard for mobile communications in the world, with over 5 billion people using GSM-compatible devices. GSM is a digital cellular technology that operates in the 900 MHz and 1.8 GHz bands, and uses time division multiplexing (TDM) to divide the available bandwidth into time slots. This allows multiple users to share the same frequency band without interfering with each other.

GSM is a second-generation (2G) mobile technology, which means that it is the second generation of mobile phone systems to be developed after the original analog systems. It was designed to provide improved voice quality and better security than the first generation of mobile phones. GSM also introduced a number of new features that are now taken for granted, such as text messaging, voicemail, and call forwarding.

One of the key features of GSM is its use of a SIM card (Subscriber Identity Module), which stores the subscriber's account information and allows the phone to be used on any compatible network. This means that users can change their phone or switch to a different network without losing their phone number or contacts. The SIM card also provides additional security features, such as a PIN code that must be entered before the phone can be used, and a SIM lock that prevents the phone from being used on other networks.

2.8 Web Application

WordPress is a content management system (CMS) that allows users to create and manage websites. It is open-source software, which means that it is free to use and can be modified by anyone.

WordPress is based on PHP and MySQL and is most used as a blogging platform, but it can also be used to create more complex websites, including e-commerce sites and portfolios. It is known for its ease of use and flexibility and has a large community of developers who create and share themes and plugins, which are add-ons that can be used to add functionality to a WordPress site. Many hosting providers offer one-click installation of WordPress, making it easy for users to set up and start using it.

2.9 Security

Security refers to the protection of information, systems, and assets from unauthorized access, use, disclosure, disruption, modification, or destruction. This can include physical security, such as locks and security cameras, as well as cyber security, which involves protecting networks and computer systems from cyber-attacks. The goal of security measures is to ensure the confidentiality, integrity, and availability of information and systems.

2.10 Internet of Things (IOT)

IoT (Internet of Things) refers to the interconnectedness of everyday devices, such as smartphones, appliances, and vehicles, through the internet. These devices are equipped with sensors, software, and network connectivity, allowing them to collect and share data with one another, and with central systems, to perform tasks automatically. This allows for greater efficiency and convenience in areas such as home automation, transportation, and healthcare.

2.11 Related work done before.

We all know that traditional green houses are very important for Agricultural productivity as we can grow and produce seasonal kinds of plants all along the year. Even though it sounds easy when you first think about preparing traditional green houses, but we wish it was that easy. Green houses are facing lot of critical difficulties such as unpredicted weather, unstable wind speed, the bad interaction between human resources and our plants and finally Control Infection and Avoid Disease Outbreak In our we are seeking for solving the issues by integrating different Artificial intelligence techniques like (‘ Machine Learning ‘ , ‘ Deep learning’ & ‘ Computer Vision’) with different electronic components and sensors not only to make the system able to control the green house but to improvise urgent situation and disaster faster than human.

Chapter 3: Design, Implementation and Result

3.1 Overview

The first step of the project is to outline the main stages of the application. Since not all of us were familiar with how to implement the application and we had limited knowledge regarding hardware, BI, Data, flutter, and marketing so we searched on Internet, some of us took summer trainings and, we asked our tutors and supervisors for help. During the process of preparation of the work, we also sketched a script for the whole project and set up correlation among tasks.

In Business model: We face problems like the dwindling size of agricultural land, particularly due to past land use changes, has led to a reduction in arable land and soil erosion on a large scale, and scarce water resources, particularly as the agricultural sector is the largest consumer of Egypt's water.

In flutter: The first challenge was how to configure power bi with mobile application and make it available for android and ios, connect mobile application with firebase real time database to get live data. we faced challenges in UI to meet the UI team needs.

The overview of the project can be described as follow:

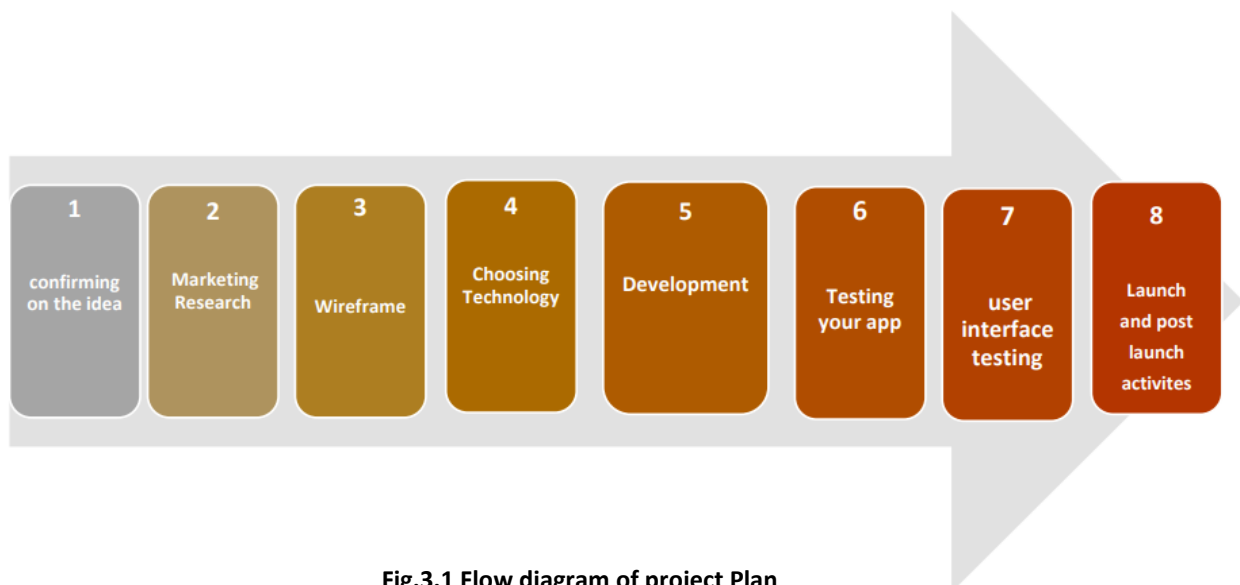
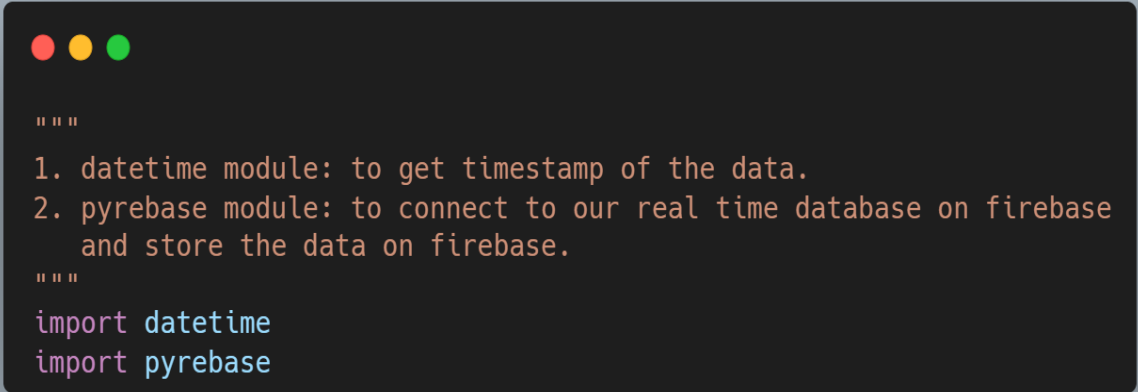


Fig.3.1 Flow diagram of project Plan

3.2 Database

1. The script will continuously read data from the sensors and upload it to the Firebase real-time database. The data that is read and uploaded includes Moisture, Temperature, Humidity, Brightness, Air Quality, and a Timestamp.
2. Make sure to have the correct wiring of the sensors with raspberry pi. And make sure your firebase project is set to public read and write.



```
"""  
1. datetime module: to get timestamp of the data.  
2. pyrebase module: to connect to our real time database on firebase  
   and store the data on firebase.  
"""  
import datetime  
import pyrebase
```

- **Installation:**
 1. Install the Pyrebase library by running `pip install pyrebase`.
 2. Clone the repository to your Raspberry Pi.
 3. Connect the sensors to the Raspberry Pi.
 4. Replace the config variable in the script with your Firebase project's configuration details.
 5. Run the script with `python sensor_data.py`.

```

#Storing the configuration data to connect to firebase
config = {
    "apiKey": "AIzaSyCqGcmWnc1vw2aU4EL9XaqAX5HJ6FiVdv0",
    "authDomain": "greenhouse-bcd96.firebaseio.com",
    "databaseURL": "https://greenhouse-bcd96-default-rtdb.firebaseio.com",
    "storageBucket": "greenhouse-bcd96.appspot.com"
}

#Initialize our connection to firebase
firebase = pyrebase.initialize_app(config)
storage = firebase.storage()
database = firebase.database()

```

```

"""
1. We read moisture, temperature, humidity, brightness, and air quality
from the raspberry pi through the sensors connected to it.
2. We get the timestamp to know every record time for future analysis.
"""

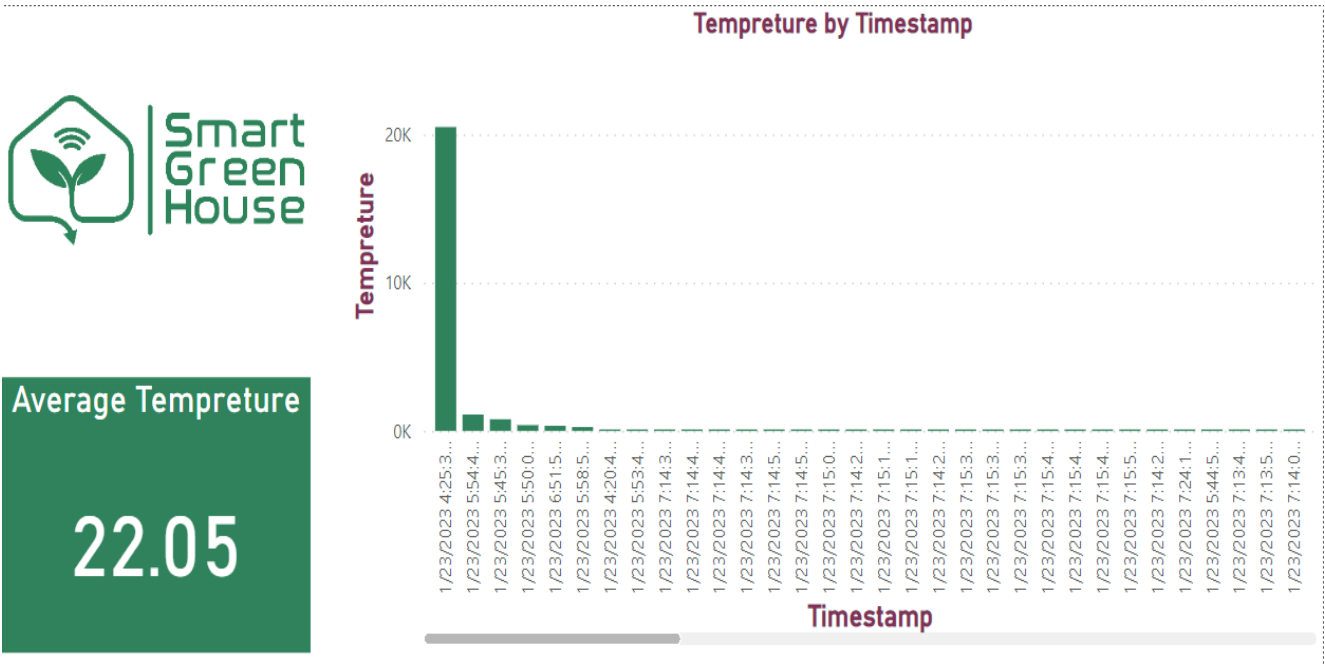
gas = mq135(ADC0834.getResult(0))
gas_measurement = gas.get_ppm()/8
humid, temp = Adafruit_DHT.read_retry(temp_sensor, dht11_pin)
moist = ADC0834.getResult(1)
TimeStamp = datetime.datetime.now().strftime("%Y-%m-%d %H:%M:%S")

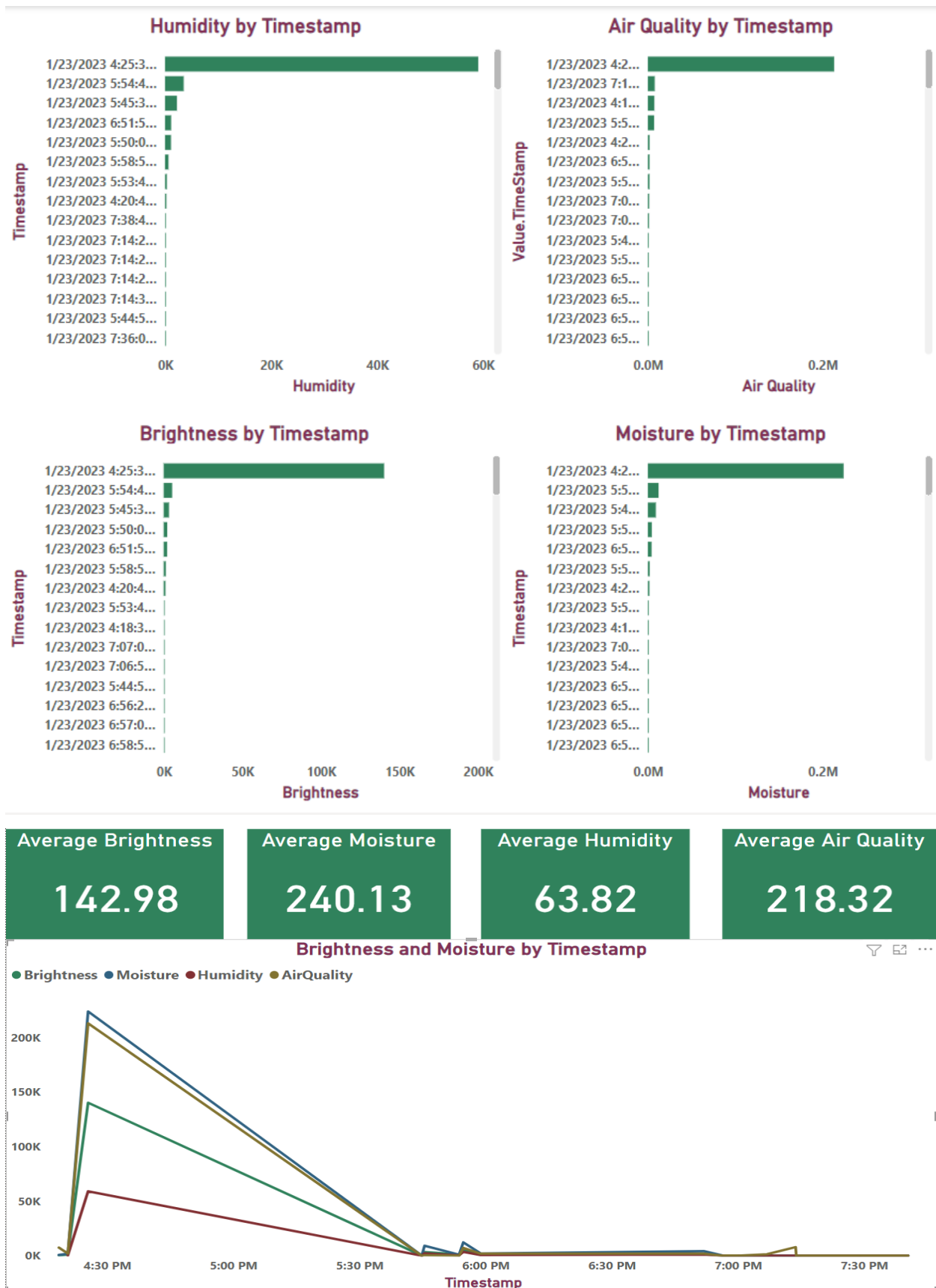
#Sending data to the real time database on firebase
res = {
    "Moisture": moist,
    "Temperature": temp,
    "Humidity": humid,
    "Brightness": bright,
    "AirQuality": gas_measurement,
    "TimeStamp": TimeStamp
}
database.push(res)

```


3.3 Dashboard and BI

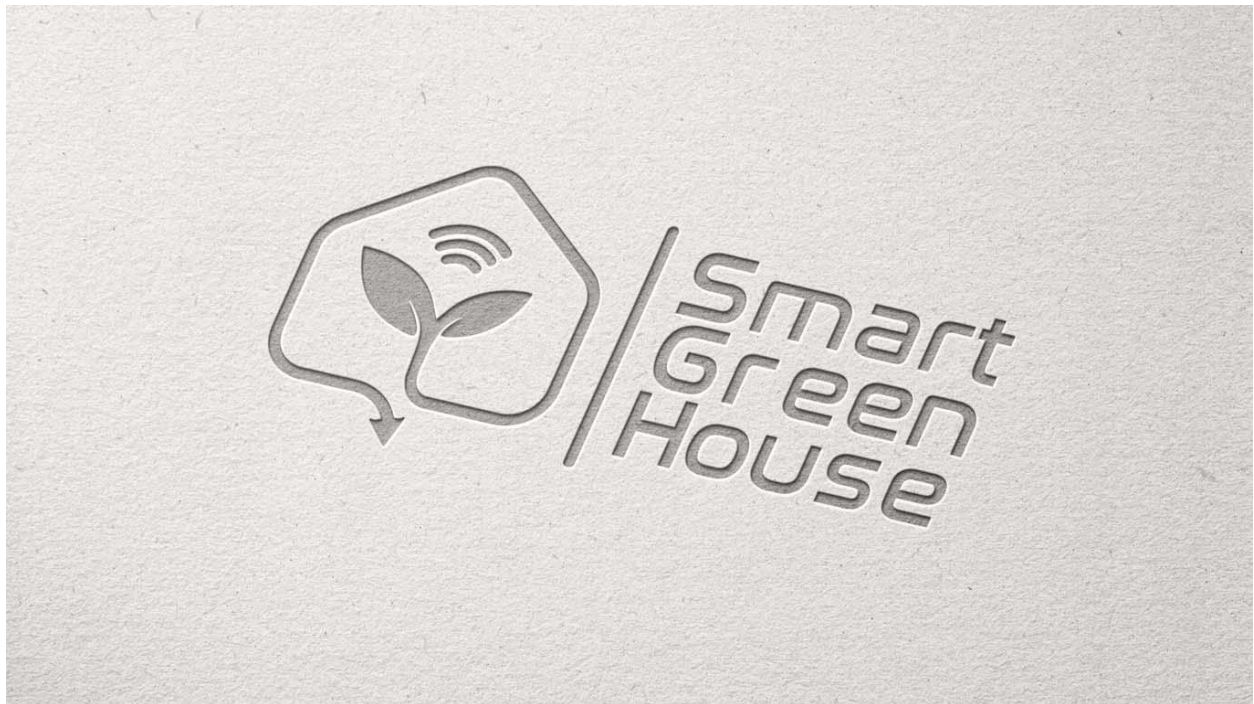
- Power BI is a powerful business intelligence tool that allows users to create interactive dashboards and visualizations from a variety of data sources. With Power BI, users can connect to data from a wide range of sources, including Excel, SQL Server, and cloud-based services such as Azure and Salesforce. Once connected, users can use Power BI's drag-and-drop interface to create a variety of charts, tables, and visualizations, as well as build interactive reports and dashboards. Power BI also provides a wide range of customization options, including the ability to add filters, drill-through actions, and custom visuals. Additionally, Power BI provides robust data modeling and analysis capabilities, including the ability to perform calculations, create measures, and create calculated columns. Power BI also provides the ability to share and collaborate on dashboards and reports with others. Overall, Power BI is a powerful tool for creating interactive, data-driven dashboards and visualizations that can help organizations gain insights and make data-driven decisions.





3.4 Brand Identity

first, we started to make some research over internet and brain storming then sketching and make our brand identity and we reach to good point at the project.





LOGO OUTLINES



LOGO IDEA



HOUSE

House element refer to greenhouse whice is main element in our project.



PLANT

Plant element refer to main elemnt in our project and what we are trying to make changes.



WIRELESS

Wireless element refer to our sensors and its ability to send data outside greenhouse to our server.




3.5 UX | User Experience

User experience (UX) for a product refers to the overall experience a user has while interacting with a product, including the design, usability, and overall satisfaction. A positive UX can lead to increased customer loyalty, while a negative UX can lead to poor reviews and a decrease in sales.

Measuring UX for a product can involve a variety of methods, such as user testing, surveys, and analytics. User testing involves having real users interact with a product and providing feedback on their experience. Surveys can also be used to gather feedback on a product's UX, while analytics can provide information on how users are interacting with a product, such as which features are being used the most and which are causing the most difficulty.

Also, we get in touch with some potential users and made personas:



**DR. AHMED
EBRAHIM
ABDULKADER
ABO ELMAATY**

AGE: 53

GENDER: Male

LOCATION: Ahmed Orabi Association ,
Obour City

ABOUT

He is a creative person highly interested in developing his knowledge, skills, expertise, and career in a manner that fulfils his enthusiasm and leads to professional recognition and career progression thus allowing him to enhance and develop his background, experiences, and talents .

PERSONALITY

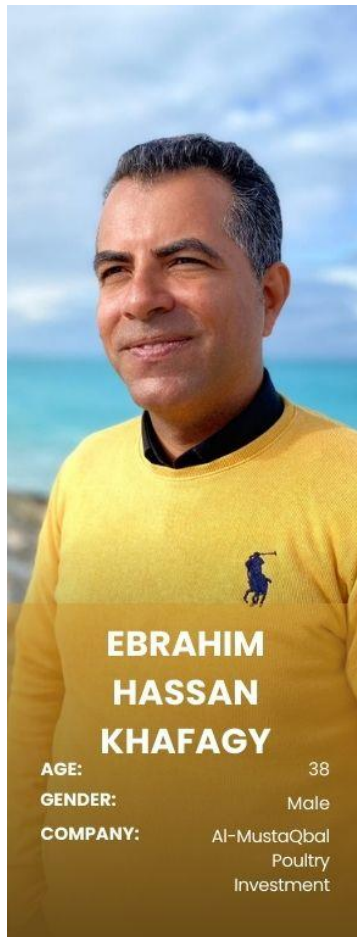
Analytical	<div><div></div><div></div><div></div><div></div><div></div></div>
Problem-Solving	<div><div></div><div></div><div></div><div></div><div></div></div>
Public Speaking	<div><div></div><div></div><div></div><div></div><div></div></div>
Adaptable	<div><div></div><div></div><div></div><div></div><div></div></div>

GOALS

- To improve his creativity and knowledge, develop his skills and expertise, advance his career, and gain professional recognition.
- To enhance his background, experiences, and talents in order to achieve these goals.

SKILLS

Communication	<div><div></div></div>
Leadership experience	<div><div></div></div>
Computer proficiency	<div><div></div></div>



**EBRAHIM
HASSAN
KHAFAQY**

AGE: 38
GENDER: Male
COMPANY: Al-MustaQbal
Poultry
Investment

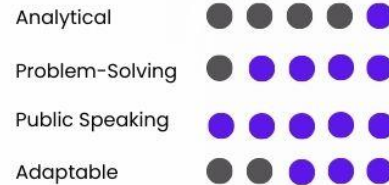
ABOUT

He studied agriculture at Cairo University in an open education program while working in the field of agriculture and poultry investment and production of fertilizers at his own company.

GOALS

- Maximizing profits and financial success for their company
- Expanding their business by acquiring new customers and increasing their market share
- Improving efficiency and productivity of their poultry operations
- Maintaining high standards of animal welfare and maintaining a good reputation in the industry
- Continuously researching and implementing new technologies and industry best practices to improve their operations
- Meeting or exceeding industry standards for food safety and quality

PERSONALITY



PAIN POINT

- Difficulty in forecasting the demand for poultry products and managing inventory accordingly.
- High operational costs, such as feed, labor, and energy costs.
- Difficulty in securing financing and funding for the business.

SKILLS



One of the key elements of a positive UX is usability, or how easy it is for a user to accomplish their goals while using a product. A product that is easy to use and navigate will have a higher level of usability than one that is difficult to understand or use. Additionally, good design can also contribute to a positive UX, as it can make a product more visually appealing and enjoyable to use.

Another important aspect of UX is accessibility, which refers to how well a product can be used by people with disabilities. A product that is designed with accessibility in mind can be used by a wider range of users and can lead to increased customer satisfaction.

3.6 UI | User Interface

User interface (UI) for a product refers to the way a user interacts with and controls the product. It is the visual and interactive elements that allow a user to interact with a product, including buttons, menus, and other controls. A well-designed UI can greatly enhance the user experience of a product and make it more enjoyable and easy to use.

Designing a good UI for a product involves a number of factors, such as layout, visual design, and functionality. The layout of a UI should be intuitive and easy to navigate, with clear and consistent visual design elements. The functionality of a UI should be designed to make it easy for a user to accomplish their goals, such as completing a task or accessing a specific feature.

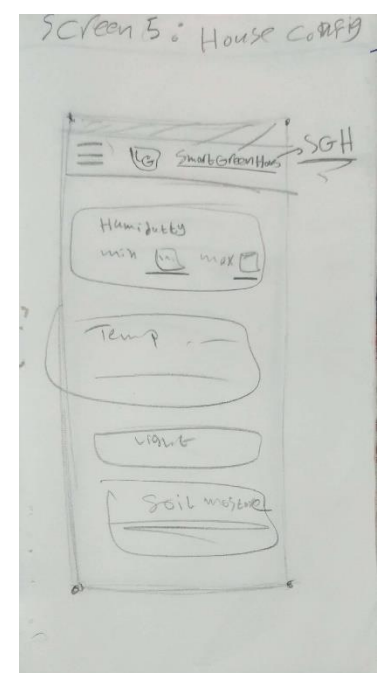
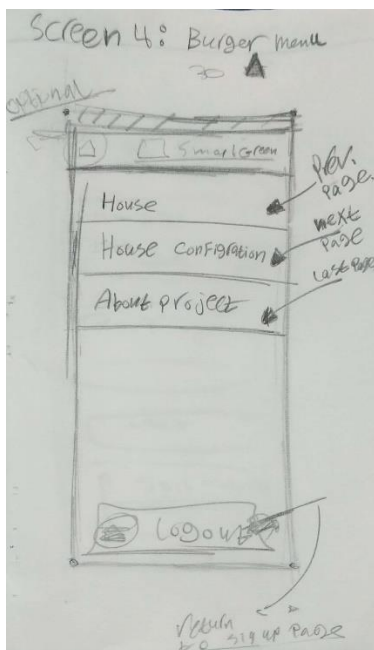
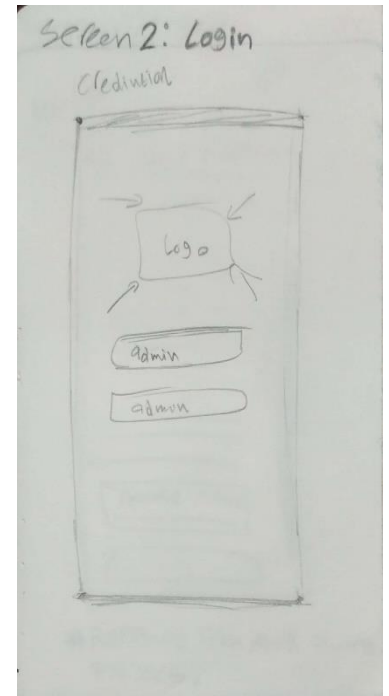
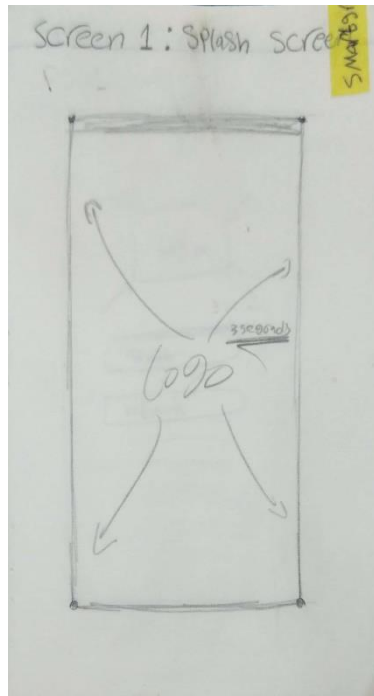
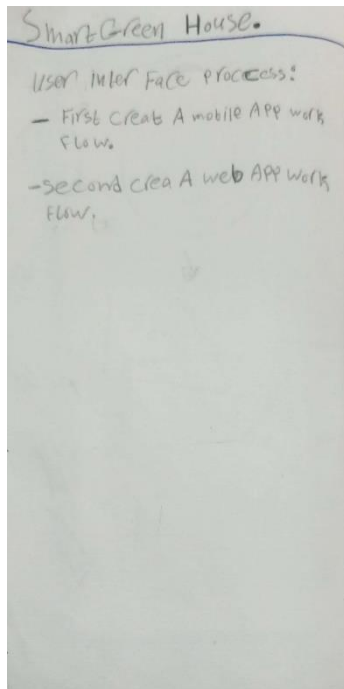
One important aspect of UI design is usability, which refers to how easy it is for a user to accomplish their goals while using a product. A product with a good UI should be easy to understand and use, with clear and consistent visual design elements. Additionally, good UI design should also consider accessibility, making sure that the interface can be used by people with disabilities.

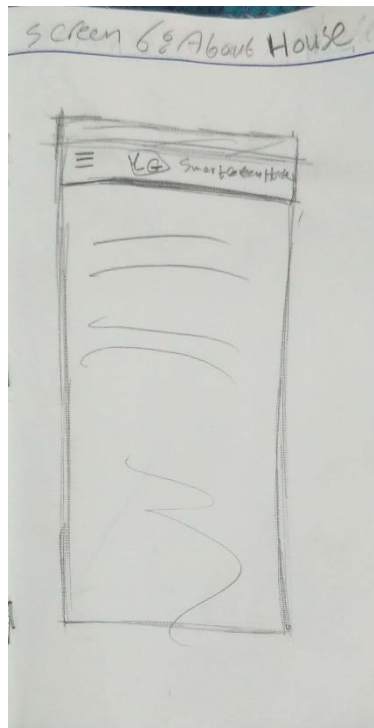
Another important aspect of UI design is consistency. Consistency in UI design helps users to understand and navigate the product, as the elements and interactions are familiar and predictable. Consistency can be achieved through the use of common design patterns, such as navigation menus and buttons, as well as through the use of consistent visual design elements, such as typography and color.

Testing the UI is also an important step in the design process. User testing can provide valuable feedback on the usability and effectiveness of the UI, and help identify areas for improvement.

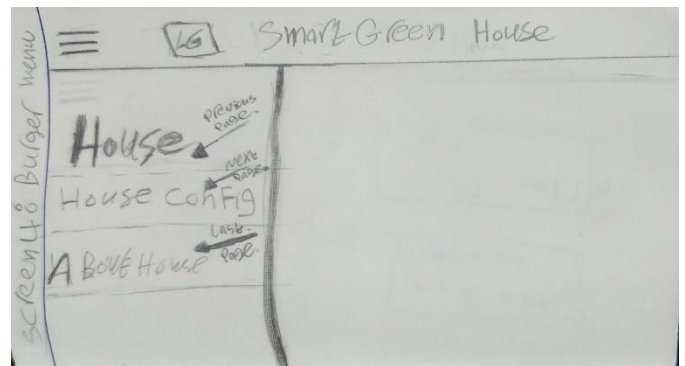
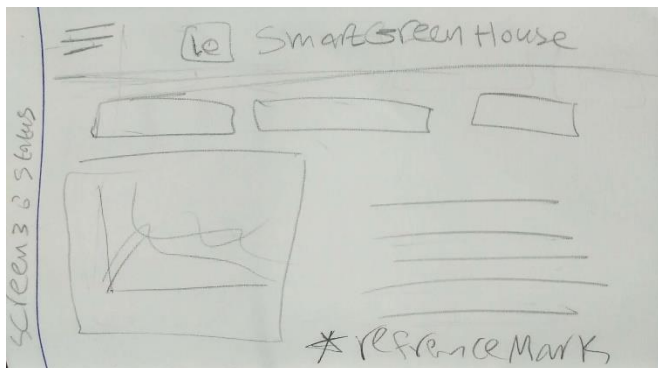
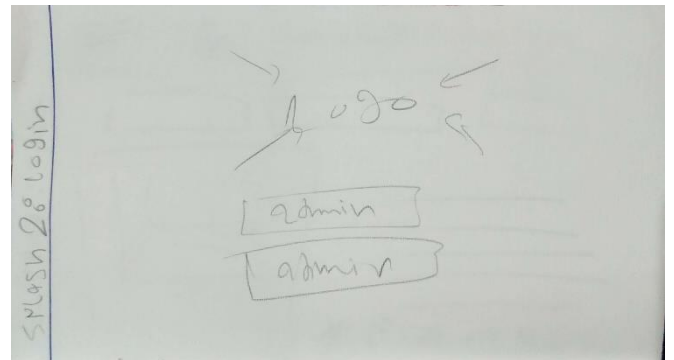
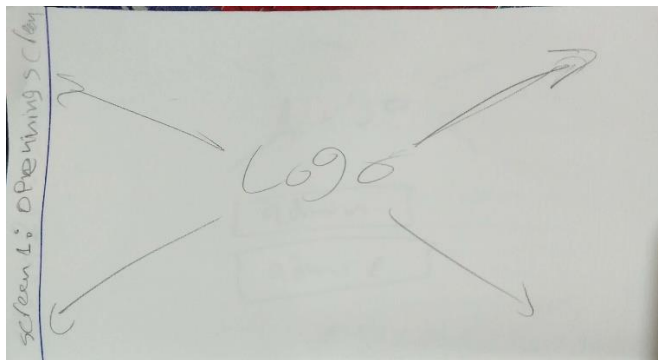
Overall, a good UI design should be intuitive, easy to use, consistent, and accessible. It should be designed with the user in mind, and should be tested to ensure that it meets their needs and expectations.

First we make a hand sketch called wireframes For mobile App:

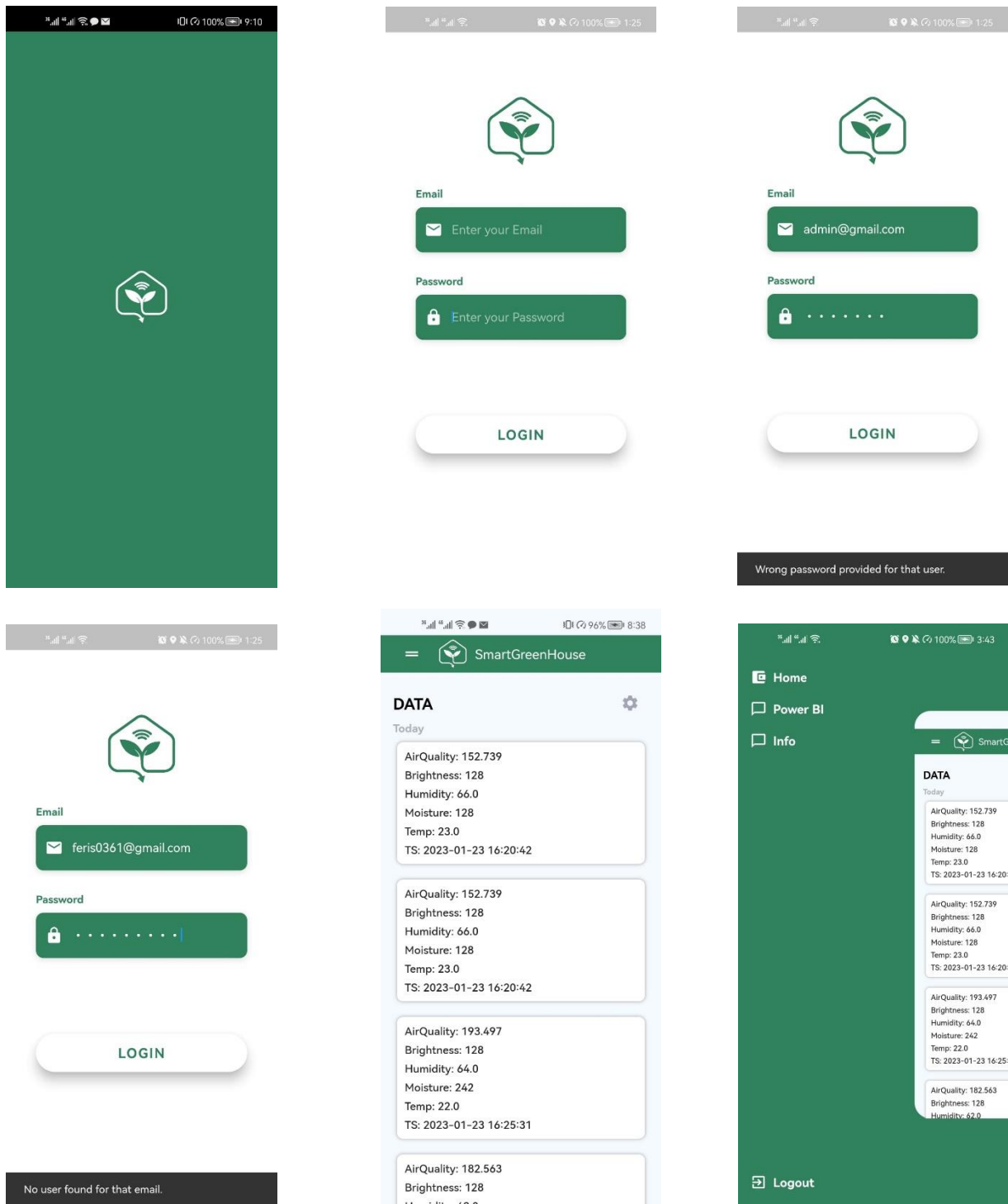




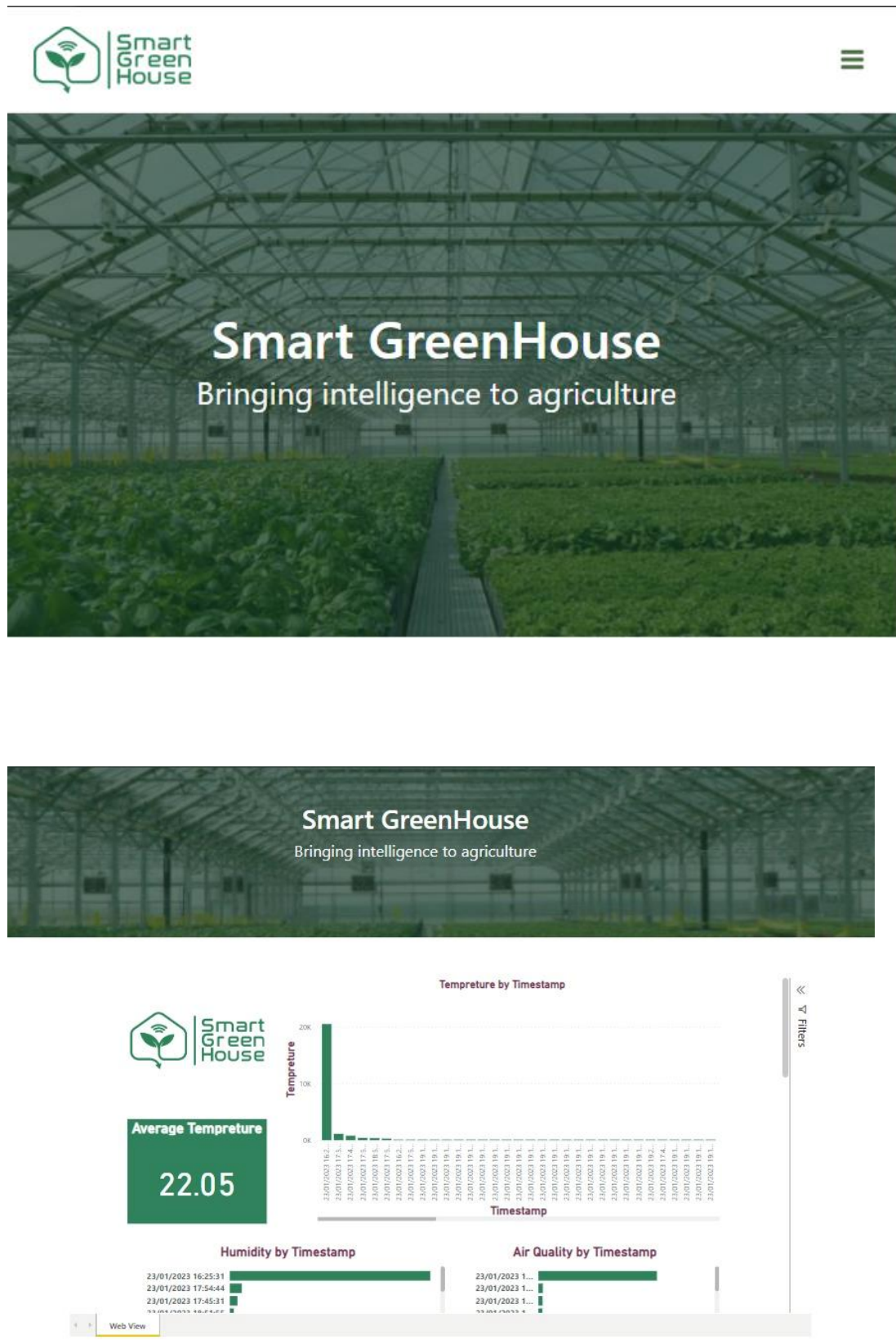
Wireframes For Web App:



Second turn all these wireframes into high fidelity designs using Figma:



Second turn all these wireframes into high fidelity designs using Figma:



Our services

Consultants

Our team of experts can help you optimize your greenhouse for maximum efficiency and productivity, using the latest technology and techniques.

Implementation

Our team of experts can help you design, install and maintain the latest technology and techniques for your greenhouse.

Prototyping

Our team of experts can help you think outside the box and come up with unique and innovative solutions for your greenhouse.

Design

Our team of experts can help you create a customized greenhouse solution that meets your specific needs and goals. We can help you design a greenhouse that is efficient, sustainable, and productive, using the latest technology and techniques.

Development

Our team of experts can help you design, develop and implement a customized solution for your greenhouse that utilizes the latest technology and techniques.

Branding

Our team of experts can help you create a unique and effective brand for your greenhouse business that sets you apart from the competition. We can help you develop a brand strategy, create a visual identity, and develop marketing materials that will help you promote your greenhouse and attract customers.



3.7 Hardware

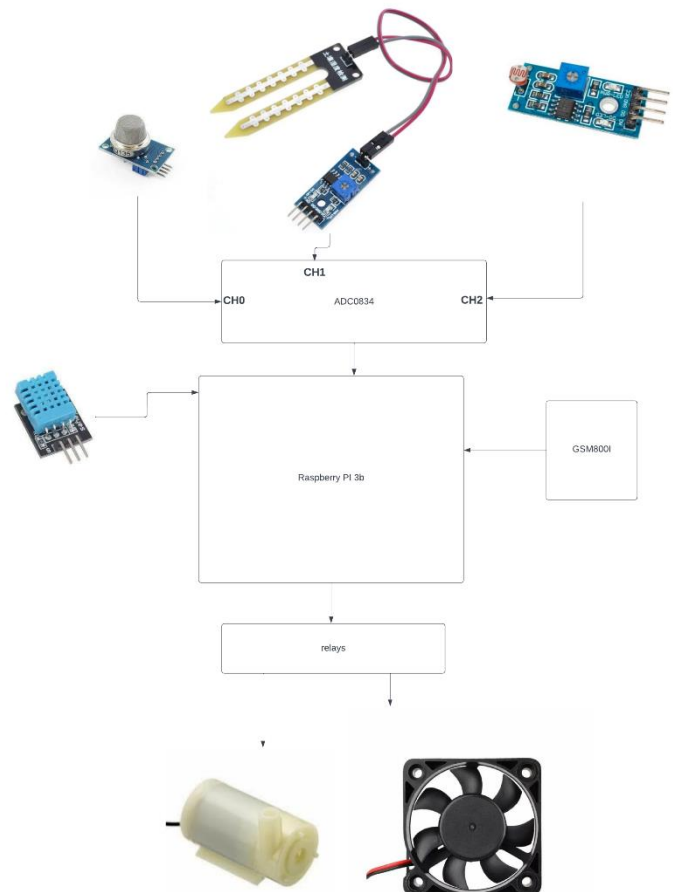
Production in a protected area is an important and fast growing component of the agricultural industry. In B&H, this production is still in development, but it can be noticed that more and more populations have the need for this type of production.

Protected areas are usually made of glass. In recent times, a greenhouse plastic film for the archive greenhouse effect is often used. In this case, the growth of plants is affected by: the thickness of the greenhouse plastic film, the permeability of the amount of light and the possibility of heat retention.

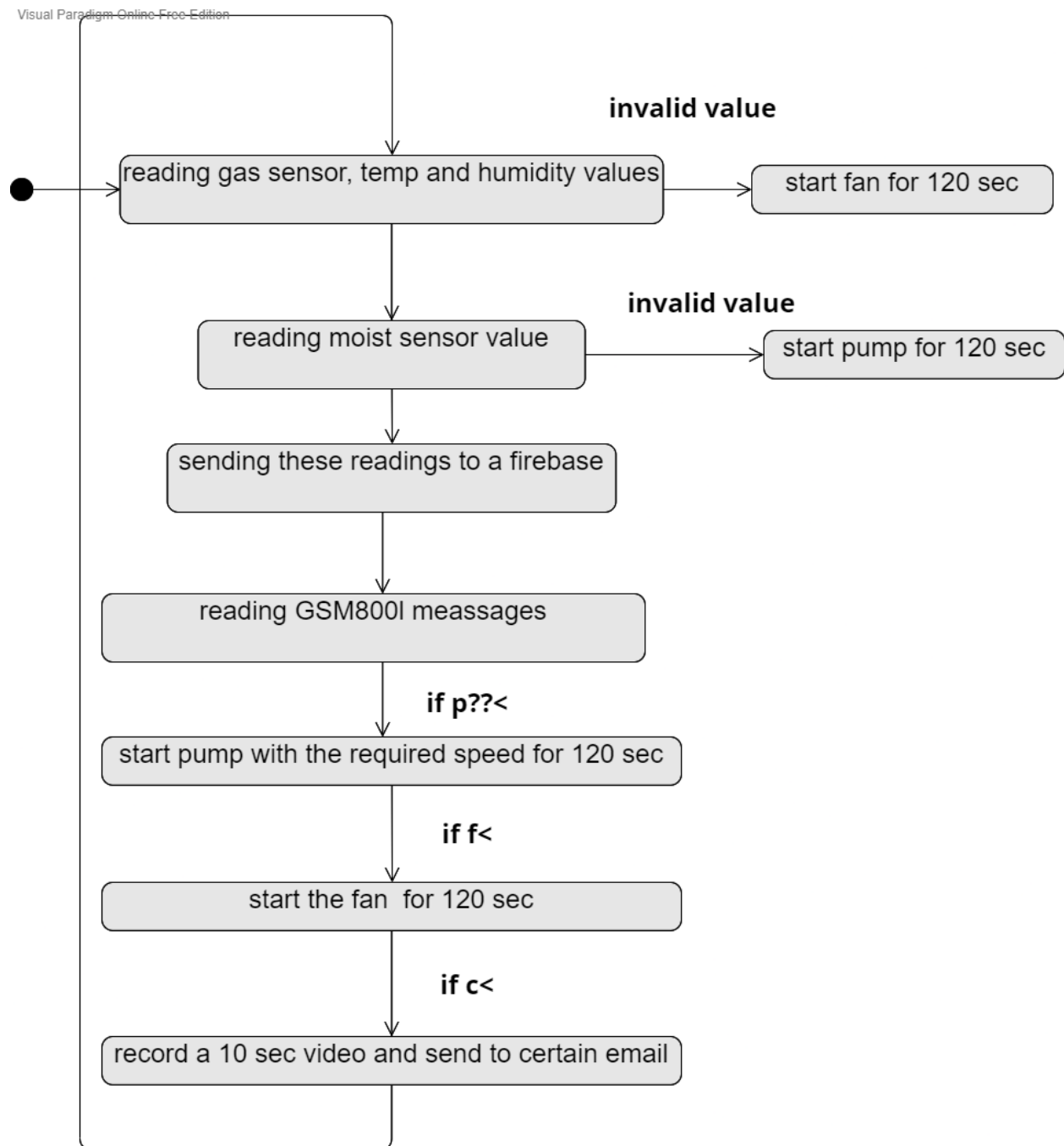
The permeability of the amount of light is essential for plant growth. Clear greenhouse plastic film is used for growth lower plants because they require direct light.

While the diffusion greenhouse plastic film is used for the growth of higher plants. For these plants, scattered light is suitable. Diffuse greenhouse plastic film reduces sunburn and the temperature inside the greenhouse. In addition to these there are anti-condensing greenhouse plastic films to prevent condensation inside the greenhouse.

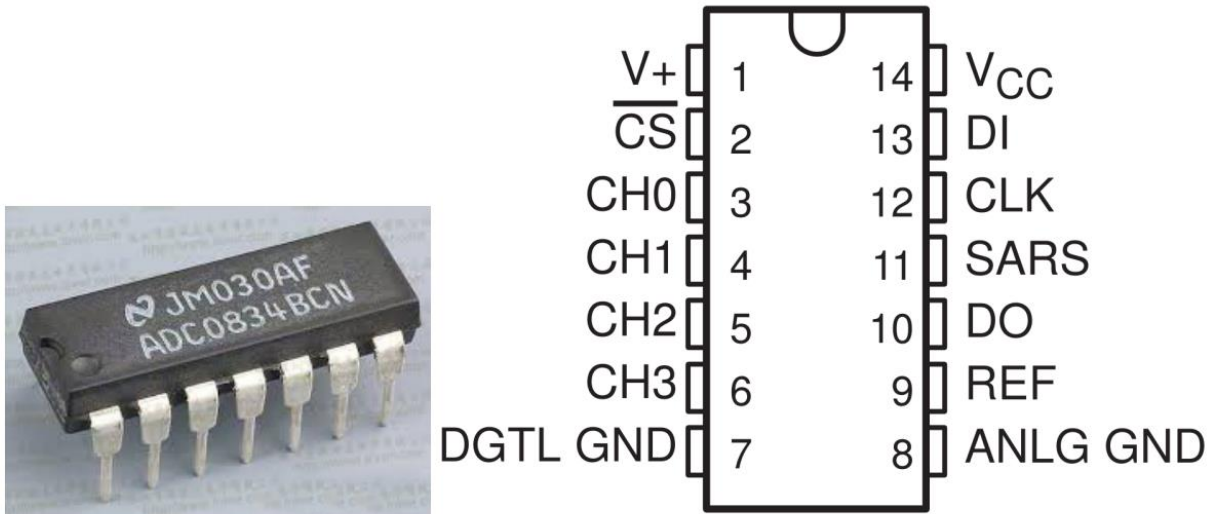
For easier production, the greenhouse can be automated. In this way, the user of day-to-day duties is released, which are crucial for maintaining optimal conditions in the greenhouse. In automated greenhouse, the user can, regardless of his location, use a computer or some other smart device to control the lighting, temperature, soil moisture and air quality.



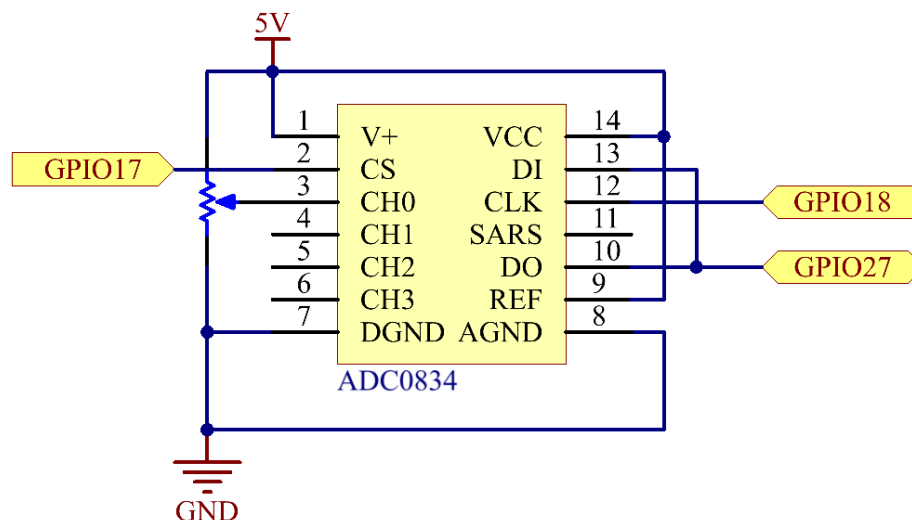
Program state diagram:



The ADC0834 is an 8-bit analog-to-digital converter (ADC) that is commonly used in a variety of applications, including data acquisition and signal processing. It is a successive-approximation type ADC, which means that it uses a binary search algorithm to determine the digital output code that most closely represents the input analog signal. The ADC0834 has a wide input voltage range of 2.5V to 5.5V and can convert an input signal with a maximum frequency of 200kHz. It also features a low power consumption of typically less than 1mA, making it suitable for battery-powered applications.



ADC and its connection with raspberry pi :



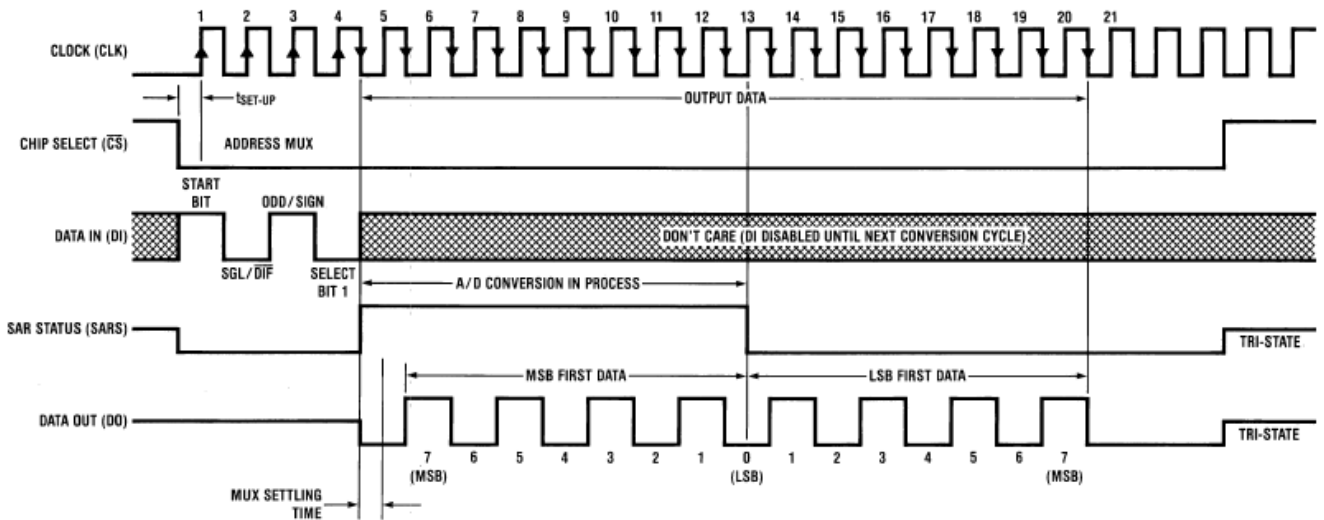
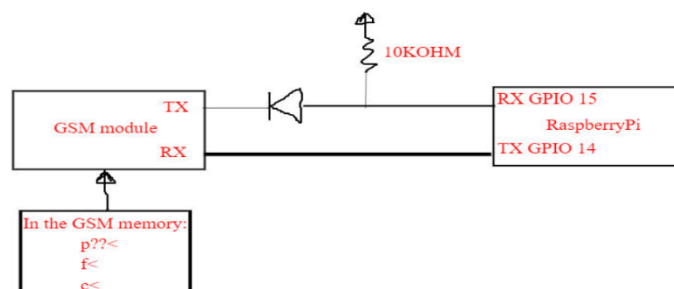


Figure 20. ADC0834-N Timing

3.8 Communication Protocols

GSM800I Module

The GSM module is connected with the Raspberry pi through UART protocol and connected with a level shifter to convert 5v from GSM to 3.3 volts from the Raspberry Pi



sim800l=SIM800L('/dev/ttyS0')

The sim800i object starts communication between Pi and GSM,

Once communication started we can send any commands from Pi to GSM from the below list of commands

Command	Description
AT+CMGD	DELETE SMS MESSAGE
AT+CMGF	SELECT SMS MESSAGE FORMAT
AT+CMGL	LIST SMS MESSAGES FROM PREFERRED STORE
AT+CMGR	READ SMS MESSAGE
AT+CMGS	SEND SMS MESSAGE
AT+CMGW	WRITE SMS MESSAGE TO MEMORY
AT+CMSS	SEND SMS MESSAGE FROM STORAGE
AT+CMGC	SEND SMS COMMAND
AT+CNMI	NEW SMS MESSAGE INDICATIONS
AT+CPMS	PREFERRED SMS MESSAGE STORAGE
AT+CRES	RESTORE SMS SETTINGS
AT+CSAS	SAVE SMS SETTINGS
AT+CSCA	SMS SERVICE CENTER ADDRESS
AT+CSCB	SELECT CELL BROADCAST SMS MESSAGES
AT+CSDH	SHOW SMS TEXT MODE PARAMETERS
AT+CSMP	SET SMS TEXT MODE PARAMETERS
AT+CSMS	SELECT MESSAGE SERVICE

3.9 Flutter

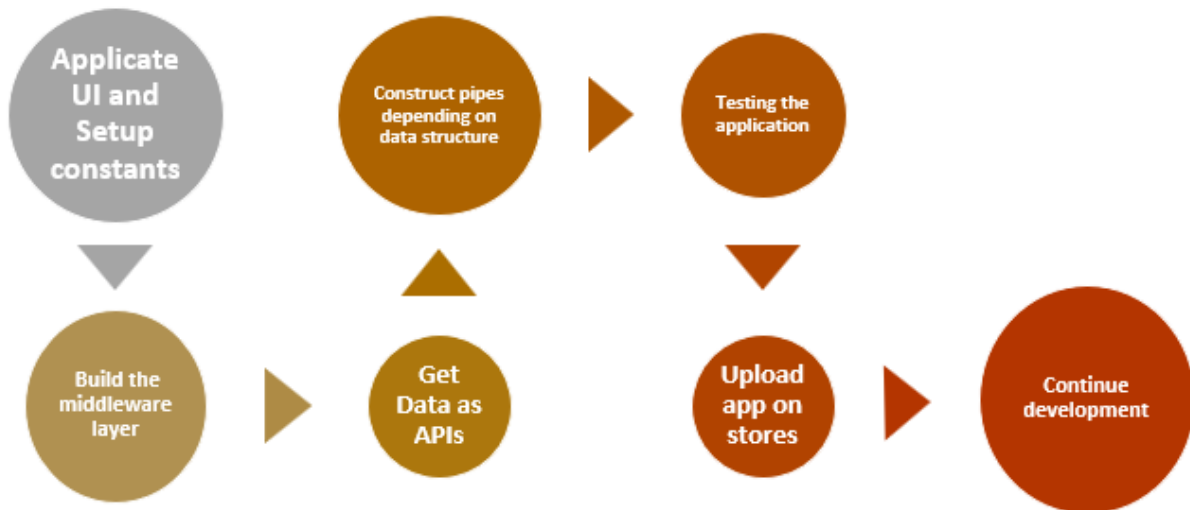


Fig.3.2 Flow diagram of Flutter

3.9.1 Applicate UI and Setup constants

User interface (UI) components encapsulate a unit of user interface interaction through a defined contract. The Web Component contract exposes the functionality of a component through the user interface, enabling the component to interact with other parts of the application. Visual Builder supports the Oracle JavaScript Extension Toolkit (JET) components contract, which adds data binding, component metadata, and dependencies on top of the Web Component contract.

3.9.2 Build the middleware layer (Business Layer)

An automotive middleware layer masks the heterogeneity of platforms, and provides high level communication services to applicative tasks. In addition, this layer is a [software architecture](#), shared

between car makers and third-part suppliers, ensuring the portability and [interoperability](#) of the applicative tasks.

3.9.3 Get Data from the backend team as APIs

API stands for [Application Programming Interface](#). It is impossible to imagine modern development without APIs. This term has many definitions, but we will try to explain it as simple as possible is like an open language, the rules of which are shared by a certain service. You can teach your application the rules of this language, so it can communicate with the service and access all the functions and data that the service is ready to share.

3.9.4 construct the pipes depending on data structure

Use [pipes](#) to transform strings, currency amounts, dates, and other data for display. Pipes are simple functions to use in [template expressions](#) to accept an input value and return a transformed value. Pipes are useful because you can use them throughout your application, while only declaring each pipe once.

3.9.5 Testing the application

In a software or application testing process there are different types of mobile application testing that help us to ensure the application works as per the user requirements or not. During testing, Most of the testing steps are conducted with the help of testing tools. Below is the list of top mobile testing tools to consider while doing app testing

1. Appium
2. Robot Framework
3. Espresso
4. Xamarin.UITest

3.9.6 Upload the app on stores.

Once you have **built an app** successfully all that remains is to submit it on the app store. We have come up with a step-by-step process of publishing an app to the Apple store. If followed in the right way, you will successfully upload your app for sure with seamless approval.

3.9.7 Continue development.

Research and development are a concept that applies to almost any product or application, and our application is no exception; therefore, we will update it constantly and carefully to satisfy the needs of end-users and vendors, as well as market and general orientation.

Flutter code:

Model Class : the structure of data that comes from API, exists to replace the runtime error with compile-time error and make code more readable and easier to develop

```
import 'dart:convert';

import 'package:firebase_database/firebase_database.dart';
import 'package:http/http.dart' as http;

class GetData {
  List<dynamic> funToReturnList(List? data) {
    Future<List> fetchAlbum() async {
      List listOldData = [];
      final response = await http.get(Uri.parse(
        'https://greenhouse-bcd96-default-rtdb.firebaseio.com/.json'));

      if (response.statusCode == 200) {
        // If the server did return a 200 OK response,
        // then parse the JSON.
        Map<String, dynamic> map = json.decode(response.body);
        map.forEach((key, value) {
          listOldData.add(value);
        });
        return listOldData;
      } else {
        // If the server did not return a 200 OK response,
        // then throw an exception.
        throw Exception('Failed to load album');
      }
    }

    return data!;
  }
}

class AutoGenerate {
  AutoGenerate({
    required this.id1,
  });
  late final id1;

  AutoGenerate.fromJson(Map<String, dynamic> json) {
    id1 = id1.fromJson(json['-NJjilGt_NMgMIB7dS-F']);
  }
}

class id1 {
  id1({
    required this.AirQuality,
    required this.Brightness,
    required this.Humidity,
    required this.Moisture,
    required this.Temperature,
    required this.TimeStamp,
  });
  late final double AirQuality;
  late final int Brightness;
  late final int Humidity;
  late final int Moisture;
  late final int Temperature;
  late final String TimeStamp;

  id1.fromJson(Map<String, dynamic> json) {
    AirQuality = json['AirQuality'];
    Brightness = json['Brightness'];
    Humidity = json['Humidity'];
    Moisture = json['Moisture'];
    Temperature = json['Temperature'];
    TimeStamp = json['TimeStamp'];
  }
}
```

Business and presentation layer: that responsible for controlling UI status, making any operation or checking then getting data from the data layer and handling errors as well

```
ListView(
  shrinkWrap: true,
  physics: ClampingScrollPhysics(),
  children: <Widget>[
    Container(
      child: Text(
        '$strDate',
        style: TextStyle(
          color: Color(0xffadb2be),
          fontWeight: FontWeight.w600,
        ),
      ),
    ),
    ListView.builder(
      padding: EdgeInsets.fromLTRB(5, 10, 5, (lastElement) ? 40 : 5),
      physics: ClampingScrollPhysics(),
      shrinkWrap: true,
      itemBuilder: (context, index) {
        return Container(
          padding: EdgeInsets.all(10),
          margin: EdgeInsets.only(
            bottom: 20,
          ),
          decoration: BoxDecoration(
            color: Colors.white,
            borderRadius: BorderRadius.circular(8),
            boxShadow: [
              BoxShadow(
                color: Colors.grey,
                blurRadius: 3,
              ),
            ],
          ),
          // padding: EdgeInsets.fromLTRB(15, 10, 15, 10),
          child: Column(
            mainAxisAlignment: MainAxisAlignment.start,
            crossAxisAlignment: CrossAxisAlignment.start,
            children: [
              Text(
                'AirQuality: ${listOfData[index]['AirQuality'].toString().substring(0,
listOfData[index]['AirQuality'] > 0 ? 7 : listOfData[index]['AirQuality'])}',
                overflow: TextOverflow.ellipsis,
                style: TextStyle(
                  fontWeight: FontWeight.w500,
                ),
              ),
              SizedBox(
                height: 8,
              ),
              Text(
                'Brightness: ${listOfData[index]['Brightness'].toString()}', // substring(0,5)
                overflow: TextOverflow.ellipsis,
                style: TextStyle(
                  fontWeight: FontWeight.w500,
                ),
              ),
              SizedBox(
                height: 8,
              ),
              Text(
                'Humidity: ${listOfData[index]['Humidity'].toString()}', // substring(0,5)
                overflow: TextOverflow.ellipsis,
                style: TextStyle(
                  fontWeight: FontWeight.w500,
                ),
              ),
              SizedBox(
                height: 8,
              ),
              Text(
                'Moisture: ${listOfData[index]['Moisture'].toString()}',
                overflow: TextOverflow.ellipsis,
                maxLines: 1,
                style: TextStyle(
                  fontWeight: FontWeight.w500,
                ),
              ),
              SizedBox(
                height: 8,
              ),
              Text(
                'Temp: ${listOfData[index]['Temperature'].toString()}', // substring(0,5)
                overflow: TextOverflow.ellipsis,
                maxLines: 1,
                style: TextStyle(
                  fontWeight: FontWeight.w500,
                ),
              ),
              SizedBox(
                height: 8,
              ),
              Text(
                'TS: ${listOfData[index]['TimeStamp'].toString()}', // substring(0,5)
                overflow: TextOverflow.ellipsis,
                maxLines: 1,
                style: TextStyle(
                  fontWeight: FontWeight.w500,
                ),
              ),
            ],
          ),
        );
      },
      itemCount: listOfData.length,
    ),
  ],
);
```

sample of translation file (locally) that comes with (.arb) extension the translation depends on the language that user already chooses and saves at local storage.

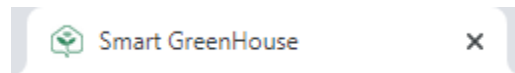


3.10 Website

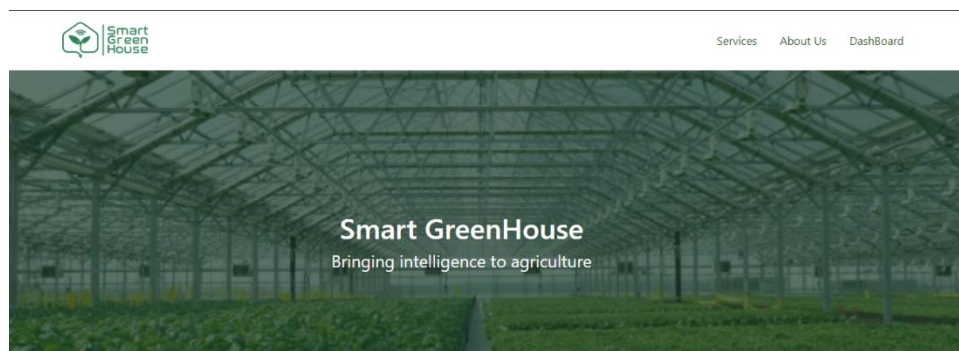
In this section we will cover the website and the tools we used to build it

The website consists of :

- Home page
- Dashboard



We used our brand name and logo as a name and logo to the website.

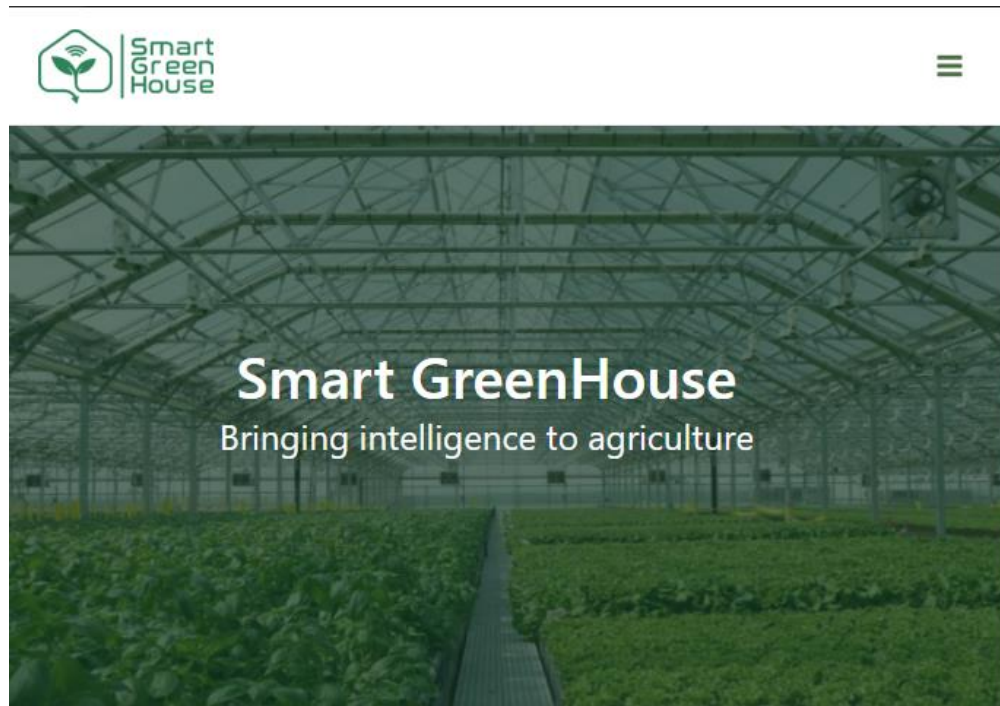


Website header consists of a navigation bar that have the content to our website.

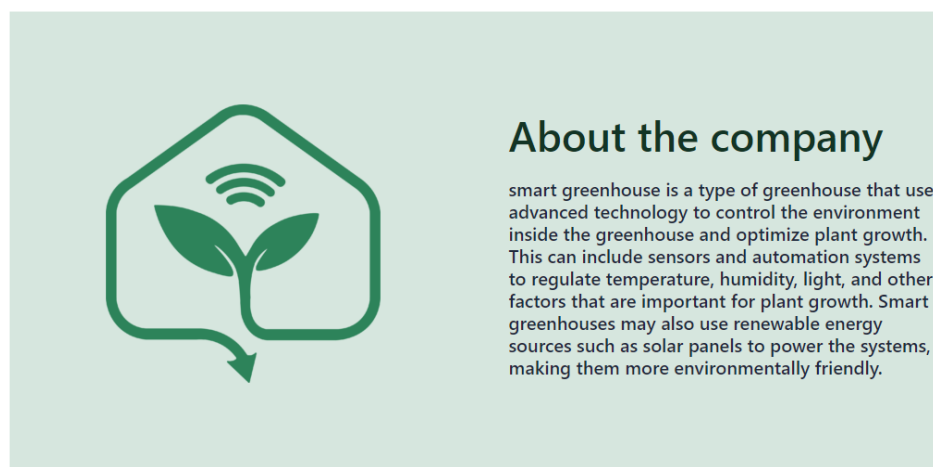
This links will scroll smoothly to their intended sections.

The header is responsive as it will fit the mobile and tablet screens. Navigation bar will be a menu contains our links.

And The Header itself has a background of a greenhouse -represents our brand- and our slogan.



The "About Us" section of a website is typically a page or section that provides information about Our Smart Greenhouse website. It is meant to give visitors an understanding of who the website belongs to, what it does, and what its goals are. The About Us section is usually one of the most visited pages on a website, as it helps visitors to understand the website's purpose, who is behind the website and what they do, and how the website can help them.



It is also animated as when the website reloaded the "About Us" will be faded in left, And the About Section will be faded in right.

The section at mobile, Tablet screens:



The "Services" section of a website is typically a page or section that provides information about the products or services offered by the organization, business, or individual that operates the website. It is meant to give visitors an understanding of what the website can offer them, and how it can help them.

Our services

Consultants

Our team of experts can help you optimize your greenhouse for maximum efficiency and productivity, using the latest technology and techniques.

Implementation

Our team of experts can help you design, install and maintain the latest technology and techniques for your greenhouse.

Prototyping

Our team of experts can help you think outside the box and come up with unique and innovative solutions for your greenhouse.

Design

Our team of experts can help you create a customized greenhouse solution that meets your specific needs and goals. We can help you design a greenhouse that is efficient, sustainable, and productive, using the latest technology and techniques.

Development

Our team of experts can help you design, develop and implement a customized solution for your greenhouse that utilizes the latest technology and techniques.

Branding

Our team of experts can help you create a unique and effective brand for your greenhouse business that sets you apart from the competition. We can help you develop a brand strategy, create a visual identity, and develop marketing materials that will help you promote your greenhouse and attract customers.

It's also Responsive.

Our services

Consultants

Our team of experts can help you optimize your greenhouse for maximum efficiency and productivity, using the latest technology and techniques.

Implementation

Our team of experts can help you design, install and maintain the latest technology and techniques for your greenhouse.

Prototyping

Our team of experts can help you think outside the box and come up with unique and innovative solutions for your greenhouse.

Design

Our team of experts can help you create a customized greenhouse solution that meets your specific needs and goals. We can help you design a greenhouse that is efficient, sustainable, and productive, using the latest technology and techniques.

A footer is a section of a website that typically appears at the bottom of every page and contains information such as copyright information.

An "up button" in a website typically refers to a button or link that allows the user to quickly navigate back to the top of the page. This is often implemented as a small arrow or icon that appears at the bottom of the screen and becomes visible when the user scrolls down the page. Clicking the button will smoothly scroll the page back to the top. This feature is useful for long web pages, as it makes it easy for users to return to the top without having to manually scroll back up.

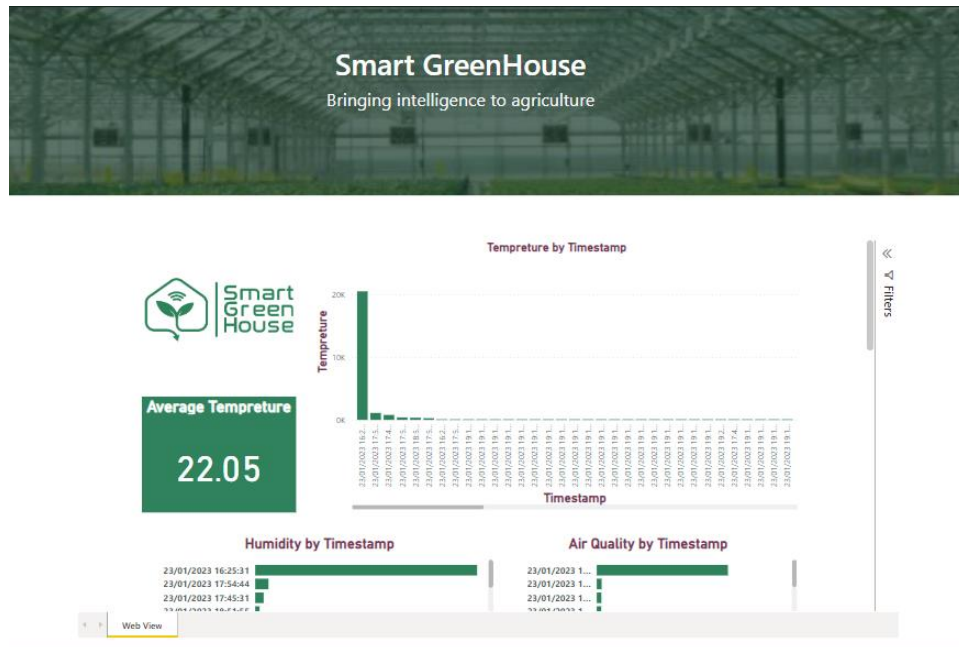
The code in JavaScript:

```
// Get the button
const upButton = document.getElementById("up-button");

// Listen for scroll events
window.addEventListener("scroll", () => {
  if (window.scrollY > 300) { // Show the button if user has scrolled down 300
    pixels
    upButton.style.display = "block";
  } else { // Hide the button if user has scrolled up
    upButton.style.display = "none";
  }
});
```

```
upButton.addEventListener("click", () => {
  window.scrollTo({
    top: 0,
    behavior: 'smooth'
  });
});
```

The Second Page of website is the Dashboard Page than contains all the data extracted from the smart greenhouse.



To configure a Power BI dashboard to a WordPress website, we followed these steps:

Created our Power BI dashboard and made sure it is published to the web. We did this by clicking on the "Share" button in Power BI Desktop and selecting "Publish to web". given a link to the dashboard that we used to embed it in our website.

Installed the "Power BI Embed" plugin on our WordPress website. This plugin allowed us to easily embed Power BI content on our website.

Once the plugin is installed and activated, we entered the link to our dashboard.

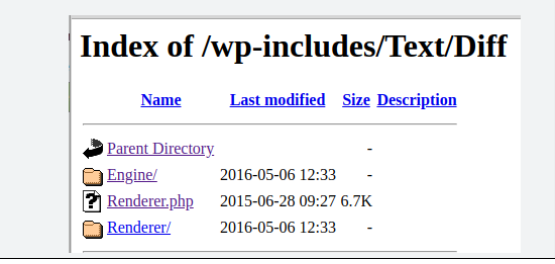
we can also configure the size of the dashboard and other options by clicking on the "Embed options" button.

Once we had entered the link to our dashboard and configured the options, we had a short code that we pasted into any page or post on our website to embed the dashboard.

Lastly, we made sure that the Power BI service is accessible from our web server. Some companies may block external access to Power BI service, so we may need to check with your IT team to make sure the service is accessible.

3.11 Security

- WordPress website up-to-date version [6.1.1]
- No 0days exploits (explain 0days)
- Vulnerabilities
 - Vulnerable to CVE-2017-5487 (the REST API implementation in WordPress doesn't properly restrict listings of post authors, which allows remote attackers to obtain sensitive information via a wp-json/wp/v2/users request.) (Request: https://green-house.website/?rest_route=/wp/v2/users/) exposes the admin email which can be used in brute force attacks against <http://green-house.website/wp-admin/>)
- WordPress uses plugins and themes that could be vulnerable to 0days attacks.
- After Enumerating the website plugins using a wordlist <https://raw.githubusercontent.com/RandomRobbieBF/wordpress-plugin-list/main/wp-plugins.lst>
- It turns out the website has 6 Plugins (all are up to date and not vulnerable to any kind of attacks)
- Defenses :
 - Disable User Enumeration
 - Limited Access to the wp-admin area
 - Change the Login Page location (default: /wp-admin) changed to a random path that cannot be brute-forced eg (<https://green-house.website/?admin-Only>)
 - Remove Wordpress Version Number (To Avoid Targeted attacks based on the version number)
 - **Remove WordPress Meta Generator Tag**
 - **Remove Slider Revolution Meta Generator Tag**
 - **Remove WPBakery Page Builder Meta Generator Tag**
 - **Remove Version from Stylesheet**
 - **Remove Version from Stylesheet (CSS Files disclosing the version numbers)**
 - **Remove Version from Stylesheet (JS Files disclosing the version numbers)**
 - Server Hardening
 - **Hide Directory Listing of WP includes**
 - Example of Directory Listing



Name	Last modified	Size	Description
Parent Directory	-	-	-
Engine/	2016-05-06 12:33	-	-
Renderer.php	2015-06-28 09:27	6.7K	-
Renderer/	2016-05-06 12:33	-	-

- Mobile App
 - Static Analysis
 - Dynamic Analysis
- Static Analysis
 - analyzing software code or other data without executing

- Hardcoded API Keys - Passwords - URLs
 - Dynamic Analysis
 - analyzing software by executing it and observing its behavior. In the context of mobile apps
 - API Testing
 - Static Analysis
 - Review the App permissions
 - android.permission.ACCESS_NETWORK_STATE >> Allows an application to view the status of all networks.
 - android.permission.INTERNET >> Allows an application to create network sockets.
 - Vulnerabilities
 - Extracting the URLs in the apk resulted in
 - api.flutter.dev
 - app.powerbi.com
 - bit.ly
 - console.firebase.google.com
 - firebase.google.com
 - flutter.dev
 - greenhouse-bcd96-default-rtdb.firebaseio.com
 - github.com
 - plus.google.com
 - file:///
 - <https://www.youtube.com/>
 - <https://api.flutter.dev/flutter/material/Scaffold/of.html>
 - <https://api.flutter.dev/flutter/dart-ui/ChannelBuffers-class.html>
 - file:///C:/Users/faris/AndroidStudioProjects/dashboard_gh/.dart_tool/flutter_build/dart_plugin_registrant.dart
 - <https://app.powerbi.com/reportEmbed?reportId=0caf4f22-abd5-40a8-969d-919e0cc5f24d&autoAuth=true&ctid=5fcc9d9b-e3d3-4e19-ac0c-90aaca677cf>
 - <https://flutter.dev/docs/release/breaking-changes/network-policy-ios-android>.
 - <https://github.com/flutter/flutter/issues/new>. greenhouse-bcd96-default-rtdb.firebaseio.com was the most interesting one . adding .json to the URI resulted in revealing all the data
 - Attacker can use `curl -X PUT -d '{"data": "value"}' 'http://[project-name].firebaseio.com/[node-name].json'` to remove and override all the data and write his own content
 - `curl -X PATCH -d '{"newData": "newValue"}' 'http://[project-name].firebaseio.com/[node-name].json'` to add to the data
 - Application signed with a debug certificate. Production application must not be shipped with a debug certificate.
 - When an Android app is developed, it is typically signed with a "debug" certificate, which is used for testing and debugging purposes. A debug certificate is created automatically by the Android SDK and is intended for use during the development

process, it is not meant for production use. A production application is an app that is intended for release to the public, and it should be signed with a "release" certificate, which is a certificate that is issued by a trusted certificate authority (CA).

- When an application is signed with a debug certificate, it can be easily compromised by attackers since the certificate is not secure, and it is intended for testing only. A production application that is shipped with a debug certificate is considered to be insecure and can be easily hacked or modified by attackers.
- To mitigate this risk, it is important to sign the production application with a release certificate, and make sure that the debug certificate is not included in the final build. This ensures that the app is secure and that its integrity is protected.
- It's also important to mention that, the debug certificates are usually self-signed, and the Android operating system does not trust them by default. This means that the app will not be able to access certain protected areas of the system, such as the external storage, and it will show a warning to the user that the app is not from a trusted source.
- Insecure Implementation of SSL. Trusting all the certificates or accepting self signed certificates is a critical Security Hole. This application is vulnerable to MITM attacks
 - "Insecure Implementation of SSL" refers to a vulnerability in the way that an app handles Secure Socket Layer (SSL) connections. SSL is a protocol that is used to secure communication over the internet by encrypting the data that is sent between a client (e.g. a web browser) and a server (e.g. a web server).
 - Trusting all the certificates or accepting self-signed certificates is considered a critical security hole because it allows attackers to intercept and modify the communication between the app and the server. This means that an attacker could potentially intercept and modify sensitive data, such as login credentials or financial information, without the user being aware.
 - When an application trusts all the certificates or accepts self-signed certificates, it is vulnerable to Man-in-the-Middle (MITM) attacks. A MITM attack is when an attacker intercepts the communication between the app and the server and impersonates the server in order to steal sensitive information.
 - The recommended practice for secure SSL implementation is to validate the server certificate using a trusted certificate authority (CA). This ensures that the certificate is legitimate and that the server is who it claims to be. Additionally, it is important to check the certificate's expiration date, common name, and issuer.
 - It's important to note that, when an app trusts all the certificates or accepts self-signed certificates, it can also be a sign of poor implementation or lack of security testing during the development process.
- Application is signed with v1 signature scheme, making it vulnerable to Janus vulnerability on Android 5.0-8.0, if signed only with v1 signature scheme. Applications running on Android 5.0-7.0 signed with v1, and v2/v3 scheme is also vulnerable.

3.12 Business Model

In this section, we will outline the plan for how our company will generate revenue and make a profit from our smart greenhouse technology. We will discuss the products and services that we will offer, the target market we will sell to, and the methods we will use to distribute and market those products and services. Additionally, we will define the costs and revenue streams associated with our operations, and the ways in which we will make money. Our business model serves as a blueprint for how our company will operate and make money from this innovative and sustainable technology.

- What We have learned:

1. How to develop a suitable business model that copes with what our country aims for EGYPT 2030.
2. How-to do-good research to know our problem well with evidence and data to find the best solution and have the greatest value from our project.
3. Know information about the different business terms such as **CT (Customer Traction)**, **CLV (Customer Life Value)**, and **RFM (recency, frequency, and monetary)** and how to deal with a business user with non-technical words.
4. Have knowledge about management terms in the organizations like CEO, COO, CTO, and CMO. And also know the difference between them and the responsibilities for each position.
- 5.

1) What is meant by business model?

- A business model is a plan for how a company will generate revenue and make a profit. It outlines the products or services the company will offer, the target market it will sell to, and the methods it will use to distribute and market those products or services. The business model also defines the costs and revenue streams associated with the company's operations and the ways in which the company will make money. Essentially, a business model is a blueprint for how a company will operate and make money.
- There are three essential questions you should answer while preparing the business model which is about the problem that we faced, the solution we came up with to solve that problem, and the value of our solution to benefit our country.

2) What is the problem?

- The dwindling size of agricultural land, particularly due to past land use changes, has led to a reduction in arable land and soil erosion on a large scale.
- Scarce water resources, particularly as the agricultural sector is the largest consumer of Egypt's water.
- The rapid population growth, which is not proportionate to the growth in agricultural land, has resulted in a decrease in the per capita share of agricultural land to less than 2 acres, compared to past periods where it was more than an acre. This requires the conference to discuss methods and solutions to control population growth in order to reap the benefits of economic growth.
- The fragmentation of agricultural land ownership, with over 3.4 million holders of less than an acre, equating to around 70% of all landholders, results in poor efficiency in land utilization, difficulty in using agricultural machinery, and increased production costs.
- The recent changes in climate, affecting various countries including the agricultural sector, has a significant impact on this sector. The increased evaporation rates and water requirements of plants, along with the effects on flowering, knots, diseases, and insects, are some of the issues that need to be addressed.

3) What is our solution?

- A smart greenhouse is a type of greenhouse that uses advanced technology to control the environment inside the greenhouse and optimize plant growth. This can include sensors and automation systems to regulate temperature, humidity, light, and other factors that are important for plant growth. Smart greenhouses may also use renewable energy sources such as solar panels to power the systems, making them more environmentally friendly.
- Some examples of the types of technology that may be used in a smart greenhouse include:
 - Temperature control: Smart greenhouses may use heating and cooling systems to maintain optimal temperature ranges for different types of plants.
 - Humidity control: Humidity sensors and systems may be used to maintain optimal humidity levels for different types of plants.
 - Lighting control: Smart greenhouses may use LED lights or other types of lighting systems that can be programmed to provide the optimal spectrum and intensity of light for different types of plants.
 - Watering systems: Smart greenhouses may use irrigation systems that can be programmed to provide the right amount of water for different types of plants at different stages of growth.
 - Soil monitoring: Smart greenhouses may use sensors to monitor soil moisture, pH, and nutrient levels, and may use automation systems to adjust these factors as needed to optimize plant growth.
- Overall, a smart greenhouse is designed to provide a controlled and optimized environment for plant growth, while also being more energy efficient and environmentally friendly than traditional greenhouses.

4) What is the value of our solution?

- The value of our project is in its ability to control and optimize the environment for plant growth, leading to increased crop yields and potentially more efficient use of resources such as water and energy.
- The use of advanced technology such as sensors, automation systems, and renewable energy sources also makes the greenhouse more environmentally friendly compared to traditional greenhouses.
- Additionally, the use of smart technology allows for more precise control of factors such as temperature, humidity, light, and soil conditions, which can lead to higher quality and more consistent crops.
- Overall, the smart greenhouse project offers a sustainable and efficient solution for plant cultivation.

5) Who is our Target market?

- There are several potential target markets for our smart greenhouse project, including:
 - Commercial farmers: Our smart greenhouse technology can be used by large-scale commercial farmers to increase crop yields and improve crop quality.
 - Small-scale farmers: Smart greenhouses can also be used by small-scale farmers to grow crops in a controlled environment, regardless of the weather conditions.
 - Urban farmers: Smart greenhouses can be used in urban areas by individuals or community groups to grow food in a small footprint and with less dependence on traditional farming methods.
 - Research institutions: Universities and research institutions can use smart greenhouses for plant research, breeding, and testing new crop varieties.
 - Government organizations: Governments and international organizations can use smart greenhouses for food security, crop development, and research.
 - Indoor agriculture: Our smart greenhouse technology can be used to create controlled environment agriculture (CEA) to grow fruits, vegetables, and herbs in indoor spaces such as warehouses, shipping containers, etc.
 - Nurseries and garden centers: Smart greenhouses can be used by nurseries and garden centers to propagate and grow plants for sale to the public.
 - Educational institutions: Smart greenhouses can be used in schools, colleges, and universities for educational purposes and for growing food for campus cafeterias.
- It is important to note that, our target market may depend on our goals, resources, and the specific features and capabilities of our smart greenhouse technology. Identifying and focusing on specific target markets can help us to better understand the needs of our customers, and to develop and market our products and services more effectively.

6) What are our expansion plans?

- € There are several potential expansion plans that we could consider for the smart greenhouse project:
 - Increasing the scale of the operation: One way to expand the project would be to increase the number of greenhouses or the size of existing greenhouses. This could include building new greenhouses or retrofitting traditional greenhouses with smart technology.
 - Diversifying the types of crops grown: Another potential expansion plan could be to diversify the types of crops grown in the smart greenhouses. This could include experimenting with different types of plants that are well suited for the controlled environment provided by the greenhouse.
 - Expanding into new geographic markets: We could look to expand into new geographic markets, different regions then countries. This could include building new greenhouses in new locations or partnering with local farmers to retrofit their greenhouses with smart technology.
 - Offering additional services: We could also consider offering additional services such as consulting, training, and technical support to farmers and other greenhouse operators to help them implement smart technology in their own greenhouses.
 - Developing new technology: We could invest in research and development to further improve the technology used in the smart greenhouse, such as using artificial intelligence to optimize the growth parameters of the plants.
 - Using the smart greenhouse technology in other sectors: We could explore the possibility of using the smart greenhouse technology in other sectors such as agriculture, horticulture, research, and education.
- € These are a few potential expansion plans, but it's important to keep in mind that the best strategy will depend on our goals, resources, and target market

Chapter 4: Conclusion and Future

In conclusion, a Smart Greenhouse is a highly efficient and sustainable living space that utilizes a variety of cutting-edge technologies to minimize its environmental impact and reduce energy consumption. Some of the key features of a Smart Greenhouse include energy-efficient appliances and lighting, solar panels for electricity generation, and smart home automation systems for monitoring and controlling various systems within the house.

One of the major benefits of a Smart Greenhouse is the significant reduction in energy consumption and associated cost savings. This can be achieved through a combination of energy-efficient appliances and lighting, as well as the use of renewable energy sources such as solar power. Additionally, Smart Greenhouses often feature advanced insulation and air sealing to reduce heat loss and improve thermal efficiency.

Another important aspect of a Smart Greenhouse is its ability to monitor and control various systems within the house, such as heating and cooling, lighting, and appliances. This can be done through the use of smart home automation systems, which can be controlled remotely via a smartphone or other device. This allows homeowners to easily monitor and adjust their energy usage, helping to further reduce their environmental impact and save money on utility bills.

Despite the benefits, there are some drawbacks to Smart Greenhouses. One major drawback is the initial cost of building or retrofitting a house to be Smart Greenhouse. This can be a significant investment and may not be feasible for some homeowners.

However, with the long-term cost savings and environmental benefits, a Smart Greenhouse can be a wise investment in the long run.

In summary, Smart Greenhouses are a highly efficient and sustainable living space that utilizes a variety of cutting-edge technologies to minimize its environmental impact and reduce energy consumption. They feature energy-efficient appliances and lighting, solar panels for electricity generation, and smart home automation systems for monitoring and controlling various systems within the house. While the initial cost may be high, Smart Greenhouses offer long-term cost savings and environmental benefits that make them a wise investment.

The project consists of

- Figma, Canva, Adobe Illustrator and Adobe Photoshop to create UI/UX design interface and all visuals of the project
- Flutter to create the mobile application that deal with end user.
- Python to deal with raspberry Pi.
- Firebase and Power BI To deal with data.
- WordPress for mobile application

4.1 Future work

- Add AI models & Machine Learning Models (Diseases classification, Location security like face recognition).
- Add Sensors for fertilizers, specifically for Nitrogen-Phosphorus-Potassium (NPK) are devices that measure the levels of these essential plant nutrients in the soil.

4.2 References

- Flutter Resources:
 - [Receive Firebase Dynamic Links in a Flutter app | Firebase Documentation \(google.com\)](#)
 - Web Resources:
 - [Mastering Wordpress](#)
 - Business Model Resources:
 - [Data Analysis Foundations - Egypt FWD \(egfwd.com\)](#)
 - [Python for Data Analysts | Pluralsight](#)
 - Database Resources:
 - [Course: Database Fundamentals \(maharatech.gov.eg\)](#)
 - UI-UX Resources:
 - [Introduction to User Experience Design Essentials - Adobe XD UI UX Design](#)
- [Google UX Design Professional Certificate | Coursera](#)

Acknowledgement

During our final project done in MU, Department of Computer and Communication, it is really appreciated to meet so many warm people and professors in the office. Our supervisors, who are a kind and enthusiastic professor that not only helps us with the project, but also leads us to our future area and builds up confidence about our further Project. It is truly an unforgettable experience to have the chance to work here!