Model Program Book

LONG TERM INTERNSHIP (On-Site/Virtual)

Designed & Developed by

APSCHE

ANDHRA PRADESH
STATE COUNCIL OF HIGHER EDUCATION
(A STATUTORY BODY OF GOVERNMENT OF ANDHRA PRADESH)

PROGRAM BOOK FOR

LONG-TERM INTERNSHIP

(Onsite/Virtual)

Name of the Student : MADDIPATI VINEELA CHOWDARY
SUTAPALLI JAGADEESH
BIRREBALASAMPATH
KOLLAPU CHAITANYA

Name of the College : GIET Engineering College

Registration Number : 20T91A0421
20T91A0439
21T95A0401
21T95A0403

Period of Internship : 6 Months

Name & Address of the Organization : DATAVALLEY India Pvt Ltd, VIP Hills,
Jaihind enclave, Madhapur, Hyderabad,
Telangana - 500081

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY,KAKINADA.

2024

An Internship Report on

WHEATHER IO:WHEATHER APP

Submitted in accordance with the requirement for the degree of Under the

Faculty Guidance of

Mr.A.G.V.KARTHIK RAJU,M.Tech.,

Assistant Professor

Submitted by:

M . VINEELA CHOWDARY	-	20T91A0421
S.JAGADEESH	-	20T91A0439
B. BALA SAMPATH	-	21T95A0401
K. CHAITANYA	-	21T95A0403

Department of ELECTRONICS AND COMMUNICATION ENGINEERING

GIET ENGINEERING COLLEGE

Accredited by NAAC, Affiliated to JNTUK, Kakinada, Chaitanya Knowledge City,

Velugubanda,Rajamahendravaram ,Andhra Pradesh, India-522396.

GIET ENGINEERING COLLEGE

NH-16, Chaitanya Knowledge city, Rajahmundry -533296

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

CERTIFICATE

This is to certify that the LMS Internship project work entitled " WEATHER.IO:WEATHER APP " is being submitted for partial fulfillment of **BACHELOR OF TECHNOLOGY** in **ELECTRONICS AND COMMUNICATION ENGINEERING** to **GIET ENGINEERING COLLEGE**, Rajahmundry, A.P. affiliated to the **J.N.T.U**, Kakinada, is Bonafied work done by **M.VINEELA CHOWDARY** Bearing Roll No: **20T91A0421**, **A.JAGADEESH** Bearing Roll No: **20T91A0439**, **B.BALA SAMPATH** Bearing Roll No: **21T95A0401**, **K.CHAITANYA** Bearing Roll No: **21T95A0403**under our guidance during the academic year 2023- 2024 and it has been found suitable for acceptance according to the requirement of University.

These results embodied in the LMS Internship project report have not been submitted to any other university or institute for the award of degree.

Ms. P. AKHILA, M.Tech
LMS Coordinator,
Department of ECE

Dr. T. SRINIVAS, M.Tech, Ph.D
Head of the Department,
Department of ECE

External Examiner

Student’s Declaration

We **M.VINEELA CHOWDARY (20T91A0421)**, **S.JAGADEESH (20T91A0439)**, **B.BALA SAMPATH (21T95A0401)**, **K. CHAITANYA (21T95A0403)**, are the students of **UG** Program, of the Department of **Electronics And Communication Engineering**, **GIET ENGINEERING COLLEGE** do hereby declare that we have completed the mandatory 6 months of internship in **Smartinternz** under the faculty guidance of **Mr. A.G.V.Karthik Raju M.Tech,** Department of **Electronics And Communication Engineering**, **GIET ENGINEERING COLLEGE** .

M.VINEELA CHOWDARY	**-**	**20T91A0421**
S.JAGADEESH	**-**	**20T91A0439**
B.BALA SAMPATH	**-**	**21T95A0401**
K. CHAITANYA	**-**	**21T95A0403**

Official Certification

This is to certify that **M.VINEELA CHOWDARY (20T91A0421)**, **S.JAGADEESH (20T91A0439)**, **B.BALA SAMPATH (21T95A0401)**, **K. CHAITANYA (21T95A0403)** has completed their internship in **Data Valley** on **WEATHER IO:WEATHER APP** under my supervision as a part of partial fulfillment of the requirement for the Degree of BACHELOR OF TECHNOLOGY in the Department of **Electronics And Communication Engineering**, **GIET ENGINEERING COLLEGE** .

Mr.A.G.V. Karthik Raju ,M.Tech
Internal Guide,
Department of ECE

Dr. T. SRINIVAS, M.Tech, Ph.D
Head of the Department,
Department of ECE

External Examiner

Certificate from Intern Organization

This is to certify that **M.VINEELA CHOWDARY (20T91A0421)**, **S.JAGADEESH (20T91A0439)**, **B.BALA SAMPATH (21T95A0401)**, **K. CHAITANYA (21T95A0403)**, of **GIET ENGINEERING COLLEGE** underwent internship in DATA VALLEY **with** time period of 6 MONTHS

The overall performance of the intern during their internship is found to be **Good.**

Acknowledgement

We would like to gratefully acknowledge our Head of the Department of Electronics And Communication Engineering, **Dr. T. SRINIVAS,** M.Tech, Ph.D., has been abundantly helpful and has assisted us in numerous ways.

We would like to express our sincere thanks to our Google Android Application Development project guide **A.G.V.KARTHIK RAJU** M.Tech.., for her valuable guidance, best suggestions and constant encouragement right to inception to the end of this report which is also enabled to complete the service project successfully in time.

We wish to thank all the staff members in the department for their kind cooperation for giving support throughout the project report.

We would like to express our deep sense of gratitude to **Dr.M.V.SEKHAR BABU**,Ph.D,Principal, for providing me a chance to undergo the course in the prestigious institute.

Finally, we acknowledge sincerely the effective services rendered by one and all involveddirectly and indirectly in the entire project.

M.VINEELA CHOWDARY	**-**	**20T91A0421**
S.JAGADEESH	**-**	**20T91A0439**
B.BALA SAMPATH	**-**	**21T95A0401**
K. CHAITANYA	**-**	**21T95A0403**

1. Introduction to Java

- Java is a popular programming language, created in 1995.
- It is owned by Oracle, and more than 3 billion devices run Java.
- Java works on different platforms (Windows, Mac, Linux, Raspberry Pi, etc.).
- It is easy to learn and simple to use.
- It is open source and free.
- It is secure, fast and powerful.
- Java is an object oriented language which gives a clear structure to programs allows code to be reused, lowering development costs.
- It is used for:
  - ✓ Mobile applications (specially Android apps)
  - ✓ Desktop applications and Web applications
  - ✓ Web servers and application servers
  - ✓ Games

Syntax:

```
public class Main{
     public static void main(String[] args){
         System.out.println("Hello World");}
}
```

- Every line of code that runs in Java must be inside a class. In our example, we named the class Main. A class should always start with an uppercase first letter.
- Java is case-sensitive: "MyClass" and "myclass" has different meaning.
- The name of the java file must match the class name. When saving the file, save it using the class name and add ".java" to the end of the filename.Any code inside the main() method will be executed.Inside the main() method, we can use the println() method to print a line of text to the screen.

1.1 Java OOP(Object Oriented Programing)

OOP stands for Object-Oriented Programming.Procedural programming is about writing procedures or methods that perform operations on the data, while object- oriented programming is about creating objects that contain both data andmethods.Object-oriented programming has several advantages over proceduralprogramming:

- ✓ OOP is faster and easier to execute.
- ✓ OOP provides a clear structure for the programs.
- ✓ OOP helps to keep the Java code DRY "Don't Repeat Yourself",and makesthe code easier to maintain, modify and debug.
- ✓ OOP makes it possible to create full reusable applications with less codeand shorter development time

Object:

Any entity that has state and behavior is known as an object. For example, a chair, pen, table, keyboard, bike, etc. It can be physical or logical. An Object can be defined as an instance of a class.

Example: A dog is an object because it has states like color, name, breed, etc. as well as behaviors like wagging the tail, barking, eating, etc.

Class:

Collection of objects is called class. It is a logical entity. A class can also be defined asa blueprint from which you can create an individual object. Class doesn't consume any space.

1.2 Inheritance

When one object acquires all the properties and behaviors of a parent object, it is known as inheritance. It provides code reusability. It is used to achieve runtime polymorphism.

Syntax:

```
class Subclass-name extends Superclass-name

 {

 //methods and fields

}
```

Types of inheritance:

- ✓ Single inheritance
- ✓ Mutliple inheritance
- ✓ Mutli-Level inheritance
- ✓ Hierarchical inheritance
- ✓ Hybrid inheritance

Polymorphism:

If one task is performed in different ways, it is known as polymorphism. For example: to convince the customer differently, to draw something, for example, shape, triangle, rectangle..,

Encapsulation :

Binding (or wrapping) code and data together into a single unit are known as encapsulation. For example, a capsule, it is wrapped with different medicines. A java class is the example of encapsulation. Java bean is the fully encapsulated class because all the data members are private here.

Interface in Java:

- ✓ An interface in Java is a blueprint of a class. It has static constants and abstract methods.
- ✓ The interface in Java is a mechanism to achieve abstraction. There can be only abstract methods in the Java interface, not method body. It is used to achieve abstraction and multiple inheritance in Java.
- ✓ In other words, you can say that interfaces can have abstract methods and variables. It cannot have a method body.

Syntax:

```
interface <interface_name>{

        // declare constant fields

       // declare methods that abstract

      // by default.

}
```

2. Introduction to JavaScript

- *JavaScript* was initially created to “make web pages alive”.
- The programs in this language are called *scripts*. They can be written right in a web page’s HTML and run automatically as the page loads.
- Scripts are provided and executed as plain text. They don’t need special preparation or compilation to run.
- In this aspect, JavaScript is very different from another language called Java.

2.1 Why is it called JavaScript?

When JavaScript was created, it initially had another name: “LiveScript”. But Java was very popular at that time, so it was decided that positioning a new language as a “younger brother” of Java would help.But as it evolved, JavaScript became a fully independent language with its own specification called ECMAScript, and now it has no relation to Java at all.

Today, JavaScript can execute not only in the browser, but also on the server, or actually on any device that has a special program called the JavaScript engine.

The browser has an embedded engine sometimes called a “JavaScript virtual machine”.

Different engines have different “codenames”.

For example:

- V8 – in Chrome, Opera and Edge.
- SpiderMonkey – in Firefox.
- There are other codenames like “Chakra” for IE, “JavaScriptCore”, “Nitro” and “SquirrelFish” for Safari, etc.

The terms above are good to remember because they are used in developer articles on the internet. We’ll use them too. For instance, if “a feature X is supported by V8”, then it probably works in Chrome, Opera and Edge.

How do engines work?

Engines are complicated. But the basics are easy.

1. The engine (embedded if it’s a browser) reads (“parses”) the script.
2. Then it converts (“compiles”) the script to machine code.

3. And then the machine code runs, pretty fast.

What CAN’T in-browser JavaScript do?

JavaScript’s abilities in the browser are limited to protect the user’s safety. The aim is to prevent an evil webpage from accessing private information or harming the user’s data.

Examples of such restrictions include:

- JavaScript on a webpage may not read/write arbitrary files on the hard disk, copy them or execute programs. It has no direct access to OS functions.
- Modern browsers allow it to work with files, but the access is limited and only provided if the user does certain actions, like “dropping” a file into a browser window or selecting it via an <input> tag.
- There are ways to interact with the camera/microphone and other devices, but they require a user’s explicit permission. So a JavaScript-enabled page may not sneakily enable a web-camera, observe the surroundings and send the information to the NSA.
- Different tabs/windows generally do not know about each other. Sometimes they do, for example when one window uses JavaScript to open the other one. But even in this case, JavaScript from one page may not access the other page if they come from different sites (from a different domain, protocol or port).
- This is called the “Same Origin Policy”. To work around that, *both pages* must agree for data exchange and must contain special JavaScript code that handles it. We’ll cover that in the tutorial.
- This limitation is, again, for the user’s safety. A page from http://anysite.com which a user has opened must not be able to access another browser tab with the URL http://gmail.com, for example, and steal information from there.
- JavaScript can easily communicate over the net to the server where the current page came from. But its ability to receive data from other sites/domains is crippled. Though possible, it requires explicit agreement (expressed in HTTP headers) from the remote side. Once again, that’s a safety limitation.

2.2 Languages "over" JavaScript

The syntax of JavaScript does not suit everyone's needs. Different people want different features.

That's to be expected, because projects and requirements are different for everyone. So, recently a plethora of new languages appeared, which are *transpiled* (converted) to JavaScript before they run in the browser.

Modern tools make the transpilation very fast and transparent, actually allowing developers to code in another language and auto-converting it "under the hood".

Examples of such languages:

- CoffeeScript is "syntactic sugar" for JavaScript. It introduces shorter syntax, allowing us to write clearer and more precise code. Usually, Ruby devs like it.
- TypeScript is concentrated on adding "strict data typing" to simplify the development and support of complex systems. It is developed by Microsoft.
- Flow also adds data typing, but in a different way. Developed by Facebook.
- Dart is a standalone language that has its own engine that runs in non-browser environments (like mobile apps), but also can be transpiled to JavaScript. Developed by Google.
- Brython is a Python transpiler to JavaScript that enables the writing of applications in pure Python without JavaScript.
- Kotlin is a modern, concise and safe programming language that can target the browser or Node.

There are more. Of course, even if we use one of these transpiled languages, we should also know JavaScript to really understand what we're doing.

- javaScript was initially created as a browser-only language, but it is now used in many other environments as well.
- Today, JavaScript has a unique position as the most widely-adopted browser language, fully integrated with HTML/CSS.
- There are many languages that get "transpiled" to JavaScript and provide certain features. It is recommended to take a look at them, at least briefly, after mastering JavaScript.

2.3 Applications of JavaScript

- **Web Development:** Adding interactivity and behavior to static sites JavaScript was invented to do this in 1995. By using AngularJS that can be achieved so easily.
- **Web Applications:** With technology, browsers have improved to the extent that a language was required to create robust web applications. When we explore a map in Google Maps then we only need to click and drag the mouse. All detailed view is just a click away, and this is possible only because of JavaScript. It uses Application Programming Interfaces(APIs) that provide extra power to the code. The Electron and React are helpful in this department.
- **Server Applications:** With the help of Node.js, JavaScript made its way from client to server and Node.js is the most powerful on the server side.
- **Games:** Not only in websites, but JavaScript also helps in creating games for leisure. The combination of JavaScript and HTML 5 makes JavaScript popular in game development as well. It provides the EaseJS library which provides solutions for working with rich graphics.
- **Smartwatches:** JavaScript is being used in all possible devices and applications. It provides a library PebbleJS which is used in smartwatch applications. This framework works for applications that require the Internet for their functioning.
- **Art:** Artists and designers can create whatever they want using JavaScript to draw on HTML 5 canvas, and make the sound more effective also can be used **p5.js** library.
- **Machine Learning:** This JavaScript ml5.js library can be used in web development by using machine learning.
- **Mobile Applications:** JavaScript can also be used to build an application for non-web contexts. The features and uses of JavaScript make it a powerful tool for creating mobile applications. This is a Framework for building web and mobile apps using JavaScript. Using React Native, we can build mobile applications for different operating systems. We do not require to write code for different systems. Write once use it anywhere!

3. Introduction Database Management System

The database is a collection of inter-related data which is used to retrieve, insert and delete the data efficiently. It is also used to organize the data in the form of a table, schema, views, and reports, etc.

- Database management system is a software which is used to manage the database. For example: MySQL, Oracle, etc are a very popular commercial database which is used in different applications.
- DBMS provides an interface to perform various operations like database creation, storing data in it, updating data, creating a table in the database and a lot more.
- It provides protection and security to the database. In the case of multiple users, it also maintains data consistency.

DBMS allows users the following tasks:

- **Data Definition:** It is used for creation, modification, and removal of definition that defines the organization of data in the database.
- **Data Updation:** It is used for the insertion, modification, and deletion of the actual data in the database.
- **Data Retrieval:** It is used to retrieve the data from the database which can be used by applications for various purposes.
- **User Administration:** It is used for registering and monitoring users, maintain data integrity, enforcing data security, dealing with concurrency control, monitoring performance and recovering information corrupted by unexpected failure.

3.1 Characteristics of DBMS

- It uses a digital repository established on a server to store and manage the information.
- It can provide a clear and logical view of the process that manipulates data.
- DBMS contains automatic backup and recovery procedures.
- It contains ACID properties which maintain data in a healthy state in case of failure.
- It can reduce the complex relationship between data.
- It is used to support manipulation and processing of data.

- It is used to provide security of data.
- It can view the database from different viewpoints according to the requirements of the user.

3.2 Advantages of DBMS

- **Controls database redundancy:** It can control data redundancy because it stores all the data in one single database file and that recorded data is placed in the database.
- **Data sharing:** In DBMS, the authorized users of an organization can share the data among multiple users.
- **Easily Maintenance:** It can be easily maintainable due to the centralized nature of the database system.
- **Reduce time:** It reduces development time and maintenance need.
- **Backup:** It provides backup and recovery subsystems which create automatic backup of data from hardware and software failures and restores the data if required.
- **multiple user interface:** It provides different types of user interfaces like graphical user interfaces, application program interfaces

3.3 Disadvantages of DBMS

- **Cost of Hardware and Software:** It requires a high speed of data processor and large memory size to run DBMS software.
- **Size:** It occupies a large space of disks and large memory to run them efficiently.
- **Complexity:** Database system creates additional complexity and requirements.
- **Higher impact of failure:** Failure is highly impacted the database because in most of the organization, all the data stored in a single database and if the database is damaged due to electric failure or database corruption then the data may be lost forever.

3.4 RDBMS (Relational Database Management System)

- **RDBMS** stands for *Relational Database Management System.*
- All modern database management systems like SQL, MS SQL Server, IBM DB2, ORACLE, My-SQL, and Microsoft Access are based on RDBMS.

- It is called Relational Database Management System (RDBMS) because it is based on the relational model introduced by E.F. Codd.

1. What is table/Relation?

Everything in a relational database is stored in the form of relations. The RDBMS database uses tables to store data. A table is a collection of related data entries and contains rows and columns to store data. Each table represents some real-world objects such as person, place, or event about which information is collected. The organized collection of data into a relational table is known as the logical view of the database.

4. HTML

HTML stands for **HyperText Markup Language**. It is the standard markup language used to create web pages. HTML is a combination of Hypertext and Markup language. Hypertext defines the link between web pages. A markup language is used to define the text document within the tag to define the structure of web pages.

This language is used to annotate (make notes for the computer) text so that a machine can understand it and manipulate text accordingly. Most markup languages (e.g. HTML) are human-readable. The language uses tags to define what manipulation has to be done on the text.

What is HTML?

HTML stands for HyperText Markup Language and it is used to create webpages. It uses **HTML tags** and **attributes** to describe the structure and formatting of a web page.HTML consists of various elements, that are responsible for telling search engines how to display page content. For example, headings, lists, images, links, and more.

HTML Example

```
<!DOCTYPE html>
<html>
  <head>
    <title>First HTML Code</title>
  </head>
  <body>
    <h2>Welcome To GFG</h2>
    <p>Hello Geeks</p>
  </body>
</html>
```

4.1 Features of HTML

- It is easy to learn and easy to use.

- It is platform-independent.
- Images, videos, and audio can be added to a web page.
- Hypertext can be added to the text.
- It is a markup language.

4.2 HTML Elements and Tags

HTML uses predefined **tags** and **elements** that tell the browser how to display the content. HTML elements include an opening tag, some content, and a closing tag.

Remember to include closing tags. If omitted, the browser applies the effect of the opening tag until the end of the page.

4.3 HTML Page Structure

The basic structure of an HTML page is shown below. It contains the essential building-block elements (i.e. doctype declaration, HTML, head, title, and body elements) upon which all web pages are created.

- **<!DOCTYPE html>** – This is the document type declaration (not technically a tag). It declares a document as being an HTML document. The doctype declaration is not case-sensitive.
- **<html>** – This is called the HTML root element. All other elements are contained within it.
- **<head>** – The head tag contains the "behind the scenes" elements for a webpage. Elements within the head aren't visible on the front end of a webpage. HTML elements used inside the <head> element include:
- **<style>** – This HTML tag allows us to insert styling into our web pages and make them appealing to look at with the help of CSS.
- **<title>** – The title is what is displayed on the top of your browser when you visit a website and contains the title of the webpage that you are viewing.
- **<base>** – It specifies the base URL for all relative URL's in a document.
- **<noscript>** – Defines a section of HTML that is inserted when the scripting has been turned off in the user's browser.
- **<script>** – This tag is used to add functionality to the website with the help of JavaScript.

- **<meta>** – This tag encloses the metadata of the website that must be loaded every time the website is visited. For eg:- the metadata charset allows you to use the standard UTF-8 encoding on your website. This in turn allows the users to view your webpage in the language of their choice. It is a self-closing tag.
- **<link>** – The 'link' tag is used to tie together HTML, CSS, and JavaScript. It is self-closing.
- **<body>** – The body tag is used to enclose all the visible content of a webpage. In other words, the body content is what the browser will show on the front end.

An HTML document can be created using an **HTML text editor**. Save the text file using the "**.html"** or "**.htm"** extension. Once saved as an HTML document, the file can be opened as a webpage in the browser.

4.4 Advantages of HTML

- HTML is used to build websites.
- It is supported by all browsers.
- It can be integrated with other languages like CSS, JavaScript, etc.

4.5 Disadvantages of HTML

- HTML can only create static web pages. For dynamic web pages, other languages have to be used.
- A large amount of code has to be written to create a simple web page.
- The security feature is not good.

5. PYTHON

Python is a widely used general-purpose, high level programming language. It was created by Guido van Rossum in 1991 and further developed by the Python Software Foundation. It was designed with an emphasis on code readability, and its syntax allows programmers to express their concepts in fewer lines of code.

Python is a programming language that lets you work quickly and integrate systems more efficiently.

There are two major Python versions: **Python 2 and Python 3**. Both are quite different.

Beginning with Python programming:

1. Finding an Interpreter:

Before we start Python programming, we need to have an interpreter to interpret and run our programs. There are certain online interpreters like **https: //ide.geeksforgeeks.org/** that can be used to run Python programs without installing an interpreter.

Windows**:** There are many interpreters available freely to run Python scripts like IDLE (Integrated Development Environment) that comes bundled with the Python software downloaded from **http://python.org/**.

Linux**:** Python comes preinstalled with popular Linux distros such as Ubuntu and Fedora. To check which version of Python you're running, type "python" in the terminal emulator. The interpreter should start and print the version number.

macOS**:** Generally, Python 2.7 comes bundled with macOS. You'll have to manually install Python 3 from **http://python.org/**.

2. Writing our first program:

Just type in the following code after you start the interpreter.

```
# Script Begins

print("GeeksQuiz")

# Scripts Ends
```

Output:

Let’s analyze the script line by line.

***Line 1:* [# Script Begins]** In Python, comments begin with a #. This statement is ignored by the interpreter and serves as documentation for our code.

Line 2: [print(“GeeksQuiz”)] To print something on the console, print() function is used. This function also adds a newline after our message is printed(unlike in C). Note that in Python 2, “print” is not a function but a keyword and therefore can be used without parentheses. However, in Python 3, it is a function and must be invoked with parentheses.

***Line 3:* [# Script Ends]** This is just another comment like in Line 1.

Python designed by Guido van Rossum at CWI has become a widely used general-purpose, high-level programming language.

Prerequisites:

Knowledge of any programming language can be a plus.

Reason for increasing popularity

1. Emphasis on **code readability, shorter codes**, ease of writing
2. Programmers can express logical concepts in **fewer lines** of code in comparison to languages such as C++ or Java.
3. Python supports **multiple** programming paradigms, like object-oriented, imperative and functional programming or procedural.
4. There exists inbuilt functions for almost all of the frequently used concepts.
5. Philosophy is “Simplicity is the best”.

5.1 LANGUAGE FEATURES

- **Interpreted**
  - There are no separate compilation and execution steps like C and C++.
  - Directly *run* the program from the source code.
  - Internally, Python converts the source code into an intermediate form called byte codes which is then translated into native language of specific computer to run it.
  - No need to worry about linking and loading with libraries, etc.

- **Platform Independent**
  - Python programs can be developed and executed on multiple operating system platforms.
  - Python can be used on Linux, Windows, Macintosh, Solaris and many more.
- **Free and Open Source;** Redistributable
- **High-level Language**
  - In Python, no need to take care about low-level details such as managing the memory used by the program.
- **Simple**
  - Closer to English language;Easy to Learn
  - More emphasis on the solution to the problem rather than the syntax
- **Embeddable**
  - Python can be used within C/C++ program to give scripting capabilities for the program's users.
- **Robust**:
  - Exceptional handling features
  - Memory management techniques in built
- **Rich Library Support**
  - The Python Standard Library is very vast.
  - Known as the **"batteries included"** philosophy of Python ;It can help do various things involving regular expressions, documentation generation, unit testing, threading, databases, web browsers, CGI, email, XML, HTML, WAV files, cryptography, GUI and many

6. EXECUTIVE SUMMARY

6.1 INTRODUCTION

Weather.io is a sleek and intuitive weather app designed by front-end developers to provide users with real-time weather information. With a user-friendly interface, interactive maps, and dynamic visuals, it delivers a seamless experience for staying updated on current weather conditions and forecasts.

6.2 Overview

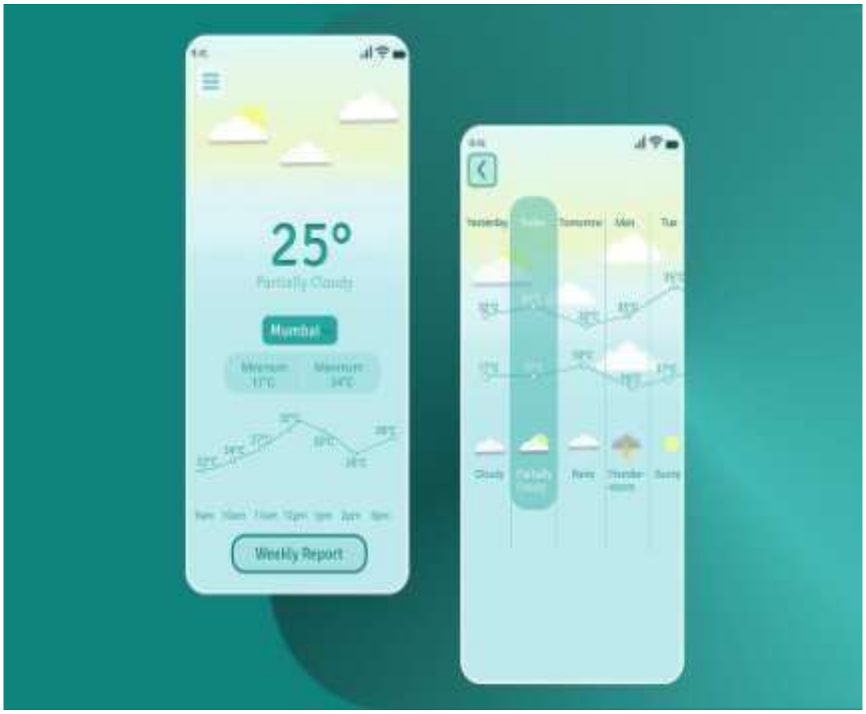


FIG 6.1:Weather App

Weather forecasting is the application of current technology and science to predict the state of the atmosphere for a future time and a given location. People can download thousands of weather apps on Apple App Store Android Play Store nowadays. Those apps show about present weather information and weather forecasts, with sleek and gorgeous interfaces. It seems it’s unnecessary to design more weather applications, but I designed new one few days ago. Weather apps enable users to get instant alerts regarding weather conditions. Many people use weather forecasts to determine what to wear on a given day. Since outdoor activities are severely curtailed by heavy rain, snow and wind chill, forecasts can be used to plan activities around these events, and to plan ahead and survive them.

6.3 Purpose

Weather forecasting is the application of science and technology to predict the conditions of the atmosphere for a given location and time. People have attempted to predict the weather informally for millennia and formally since the 19th century. Weather forecasts are made by collecting quantitative data about the current state of the atmosphere, land, and ocean and using meteorology to project how the atmosphere will change at a given place.

The inaccuracy of forecasting is due to the chaotic nature of the atmosphere, the massive computational power required to solve the equations that describe the atmosphere, the land, and the ocean, the error involved in measuring the initial conditions, and an incomplete understanding of atmospheric and related processes. Hence, forecasts become less accurate as the difference between current time and the time for which the forecast is being made increases.

There are a vast variety of end uses for weather forecasts. Weather warnings are important because they are used to protect life and property. Forecasts based on temperature and precipitation are important to agriculture, and therefore to traders within commodity markets.

Temperature forecasts are used by utility companies to estimate demand over coming days.

7. OVERVIEW OF THE ORGANIZATION

7.1 Introduction of the Organization

Datavalley is an Experiential Learning & Remote Externship Platform to bring academia & industry very close for a common goal of talent creation. "Datavalley" is a platform that offers internships and skill development programs. They focus on providing opportunities for students and young professionals to gain practical experience in various industries. In Datavalley is collaborated with stack next us to provide the effective knowledge in full stack web development.

"Nexus" could refer to various things depending on the context. One common usage is in the field of software development, where Nexus might be associated with the Nexus Framework, an agile framework that helps organizations scale and manage large software development projects.

As a certified partner of some of the world’s most impactful technology companies, Stack

Nexus builds new platforms or supplements existing ones to elevate your business’ offerings. By harnessing the powerful capabilities of these leading platforms, Stack Nexus provides meaningful technology solutions that endure the dynamic technological landscape.

7.2 Vision, Mission, and Values of the Organization

Vision:

Empowering the workforce of tomorrow.

Mission:

Datavalley mission is to establish a sustainable talent pipeline for the emerging industry by fostering Strong industry-academia connections. Through their project-based learning and virtual internship Programs, they equip students with in-demand skills in cutting-edge technologies, empowering them to succeed in their careers.

Values:

- To provide innovative learning aids and services in the education sector.
- To connect industry-ready professionals, researchers, advanced learners, educators and entrepreneurs who can take best care of stake holders.

- To extend cutting-edge research, publications and consultancy.

7.3 Existing Problem

Weather.io: weather apps are valuable for organizing tasks, but they come with challenges. Users may feel overwhelmed with too many tasks, struggle to prioritize effectively, or neglect the app over time. Real-time updates and integration issues can lead to confusion, while poor reminder management may result in task delays.

- **Synchronization Issues:** Users may experience delays or inconsistencies in task synchronization across devices, leading to discrepancies in their to-do lists.
- **Notification Overload:** Users might feel overwhelmed by a high volume of notifications, potentially leading to distraction rather than productivity.
- **Learning Curve for Advanced Features:** Some users may find it challenging to grasp and utilize the app's advanced features effectively.
- **Integration Challenges:** Users may encounter difficulties integrating the To-Do App with other tools or platforms they regularly use.
- **Limited Offline Functionality:** Users may face limitations when trying to access and edit tasks in offline mode.

7.4 Proposed Solution

To create a weather app, consider integrating features such as real-time weather updates, location-based forecasts, a user-friendly interface, and perhaps additional features like radar maps or severe weather alerts.

- **Synchronization Issues:** Verify your internet connection and ensure that the app is updated to the latest version. If the issue persists, manually refresh the app or log out and log back in to trigger a synchronization.
- **Notification Overload:** Customize notification settings within the app. Tailor notifications to only alert you about high-priority tasks or upcoming deadlines. Adjusting these settings can help maintain focus while staying informed.
- **Learning Curve for Advanced Features:** Offer comprehensive tutorials or tooltips within the app, guiding users through advanced functionalities. Providing user-friendly documentation and video tutorials can enhance the learning experience for users seeking to leverage the app's full potential.

- **Integration Challenges:** Improve integration capabilities and provide clear documentation on how to connect the app with popular tools. Offering dedicated support for integration-related queries can further assist users in streamlining their workflows.
- **Limited Offline Functionality:** Enhance offline functionality by allowing users to make changes to tasks without an internet connection. Implement an automatic sync feature when the device reconnects to the internet to ensure data consistency.

8. INTERNSHIP PART

8.1 INTRODUCTION

Overview:

Weather.io: Weather app is a comprehensive weather app designed to provide users with accurate and real-time weather information. With a user-friendly interface, the app offers features such as location-based forecasts, live weather updates, and customizable preferences. Users can personalize their experience by choosing preferred units and adding favourite locations for quick access. The app may include advanced features like radar maps, offering a visual representation of weather patterns.

Purpose:

To create a weather app, consider integrating features such as real-time weather updates, location-based forecasts, a user-friendly interface, and perhaps additional features like radar maps or severe weather alerts. Ensure a responsive design for various devices, and prioritize data accuracy from reliable sources.

- **Real-time Weather Updates:** Users can access up-to-the-minute weather information, ensuring they stay informed about the current conditions at their location or any specified location.
- **Location-Based Forecasts:** The app provides personalized weather forecasts based on the user's current location, allowing for accurate and relevant information tailored to their surroundings.
- **Customizable Preferences:** Users can personalize their experience by setting preferences such as preferred units (e.g., Celsius or Fahrenheit) and adding favourite locations for quick and easy access to weather information for frequently visited areas.
- **Advanced Features:** weather.io may include advanced features like radar maps, providing users with a visual representation of weather patterns and helping them track changes over time.
- **Severe Weather Alerts:** To ensure user safety, the app may offer alerts for severe weather conditions, allowing users to take timely precautions when faced with potentially hazardous weather events.

DIAGRAMMATIC OVERVIEW OF THE PROJECT

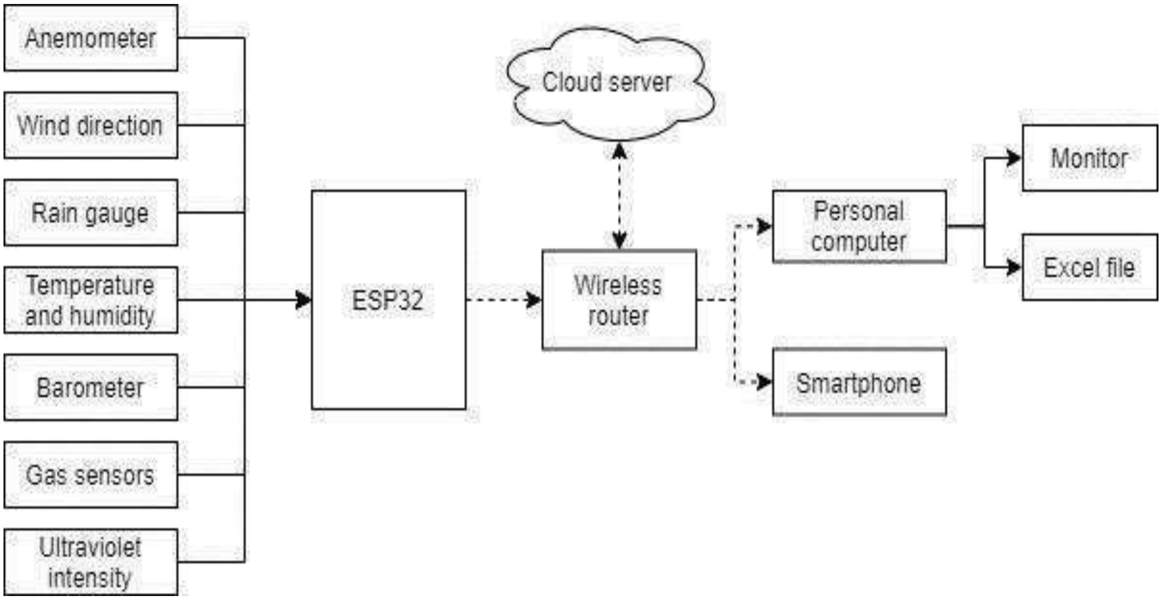


FIG 8.1:Overview of the project

8.2 HARDWARE REQUIREMENTS

The firmware for the ESP32 was built according to the workflow. Based on the ESP32 workflow installed in the field station together with all these sensors starts with the initialization of the pins used, the library, the connection to the sensor, and the connection to Wi-fi. After that, the device is connected to a local Wi-fi network with the SSID and password that has been previously set. After a successful connection, the device will activate the server. Get into the main program, that runs in an infinite loop to read all sensor data, combine all readings into one string, then send it to the server if there any request from client device. The hardware design of the system. It uses ESP 32 development kit C as the main processor. Both processor and all sensors supplied by 2 DC/DC step down converters. The system uses 16x2 LCD to show the connected wi-fi said and its IP address on local connection.

SOFTWARE USED

Visual Studio Code - Integrated Development Environment (IDE)

9. OUTCOMES DESCRIPTION

Managerial Skills that I have acquired

- **Planning:** Planning is a vital aspect within an organization. The planning process includes identifying and setting achievable goals, developing necessary strategies, and outlining the tasks and schedules on how to achieve the set goals. Without a good plan, little can be achieved.
- **Decision Making:** For the organization to run effectively and smoothly, clear and right decisions should be made. A manager must be accountable for every decision that they make and also be willing to take responsibility for the results of their decisions. A good manager needs to possess great decision-making skills, as it often dictates his/her success in achieving organizational objectives.
- **Problem - solving:** Problem-solving in management involves identifying a certain problem or situation and then finding the best way to handle the problem and get the best solution. It is the ability to sort things out even when the prevailing conditions are not right.
- **Motivating:** Motivation helps bring forth a desired behaviour or response from the employees or certain stakeholders.
- **Delegation:** Delegation is the act of passing on work-related tasks and/or authorities to other employees or subordinates. It involves the process of allowing your tasks or those of your employees to be reassigned or reallocated to other employees depending on current workloads.
- **Communication:** It can determine how well information is shared throughout a team, ensuring that the group acts as a unified workforce. How well a manager communicates with the rest of his/her team also determines how well outlined procedures can be followed, how well the tasks and activities can be completed, and thus, how successful an organization will be.

How Can I Improve My Communication Skills:

- By being an “Active Listener”.
- Speaking up about my Thoughts and Ideas.
- Thinking Before Speaking.

- Being brief and Specific about The Topic.
- Writing down the Topic that I want to speak.
- Changing the Way of speaking Based on the person whom I was taking too.

Technological Developments that I have observed:

Technology is changing in the way organizations and the employees need to accomplish their work. Empirical evidence on this topic is scarce. The aim of this study is to provide an overview of the effects of technological developments on work characteristics and to derive the implications for work demands and continues vocational education and training (CVET).

Technologies, defined a digital, electrical or mechanical tool that effect the accomplishment of work tasks, are considered in various disciplines, such as sociology or psychology. Authentically frame works based on theories from these disciplines was developed on the relationships between technology and work characteristics, such as complexity autonomy or meaningfulness, were divided. Empirical evidence was extracted and its implications for work demands and CVET were divided by using a model that illustrates the components of learning environments.

10. SOURCE CODE

10.1 INDEX.HTML

```
<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Weather.io</title>

    <link rel="stylesheet" href="./style.css">

</head>

<body>

    <div class="container">

        <div class="weather_header">

            <form class="weather_search">

                <input type="text" placeholder="Search for a city..." class="weather
          searchform">

                <i class="fa-solid fa-magnifying-glass"></i>

            </form>

            <div class="loc weather_____location-btn"><i class="fa-solid fa-
      location"></i></div>

            <div class="weather_units">

                <span class="weather_unit_celsius">&#176C</span>
                <span class="weather_unit_farenheit">&#176F</span>

            </div>

        </div>

        <div class="weather_body">

            <h1 class="weather_city"></h1>

            <div class="weather_datetime">

            </div>

            <div class="weather_forecast"></div>

            <div class="weather_icon">

            </div>

            <p class="weather_temperature">

            </p>

            <div class="weather_minmax">

                      <p>Min: 12&#176</p>

                       <p>Max: 16&#176</p>

            </div>

        </div>

        <div class="weather_info">

                  <div class="weather_card">

                <i class="fa-solid fa-temperature-full"></i>

                <div>

                    <p>Feels like</p>

                    <p class="weather_realfeel">18&#176</p>

                </div>

            </div>

                 <div class="weather_card">
```

```
                    <i class="fa-solid fa-droplet"></i>

                    <div>

                        <p>Humidity</p>

                        <p class="weather_humidity">18&#176</p>

                    </div>

                </div>

                <div class="weather
                        card">

                    <i class="fa-solid fa-wind"></i>

                    <div>

                        <p>Wind</p>

                        <p class="weather_wind">18&#176</p>

                    </div>

                </div>

                <div class="weather_card">

                    <i class="fa-solid fa-gauge-high"></i>

                    <div>

                        <p>Pressure</p>

                        <p class="weather_pressure">18&#176</p>

                    </div>

                </div>

            </div>

        </div>

    <script
src="https://kit.fontawesome.com/a692e1c39f.js"
```

```
crossorigin="anonymous"></script>

        <script src="./script.js"></script>

    </body>

    </html>
```

10.2 Style. CSS

```
@import url("https://fonts.googleapis.com/css2?family=Raleway&display=swap");

* {

margin: 0;

padding: 0;

box-sizing: border-box;

font-family: "Raleway", sans-serif;

}

body {

  background-image: url('https://e1.pxfuel.com/desktop-wallpaper/443/448/desktop-
wallpaper-rainy-weather-mobile-rain.jpg');
background-repeat: no-repeat;

background-size: cover;

}

. loc {

font-size: 1rem;

padding: 0.5rem 1rem;

background-color: #1e1e1e;

cursor: pointer;

border-radius: 5px;
```

```
color: #000000;

background: rgba (255, 255, 255, 0.21);

}

.container {

/* background: rgba (0, 0, 0, 0.71);

*/border-radius: 10px;

border: 1px solid rgba (255, 255, 255, 0.38);

background: rgba (255, 255, 255, 0.292);

backdrop-filter: blur(15px);

color: #fff; padding: 2rem;width: 40%;

margin: 2rem auto;

border-radius: 10px;

}

. weather_header {

  display: flex;

  justify-content: space-between;

  align-items: center;

  flex-wrap: wrap;

  }

input  {

  border: none;

  background: #1e1e1efc;

  outline: none;
  color: #fff;

  padding: 0.5rem 2.5rem;

  border-radius: 5px;

  }

Input: placeholder {

color: #fff;

  }

. weather_search {

    position: relative;

  }

. weather_search i {

   position: absolute;

   left: 10px;

   top: 10px;

   font-size: 15px;

    color: #fff;
  }

. weather_units {

    font-size: 1.5rem;

  }

. weather_units span {

     cursor: pointer;

  }

. weather_units span: first-child {
     margin-right: 0.5rem;
  }
. weather_body {
    text-align: center;
    margin-top: 3rem;
  }
. weather_datetime {
     margin-bottom: 2rem;
     font-size: 14px;
  }
  . weather_forecast {
    background: #1e1e1e;
    display: inline-block;
    padding: 0.5rem 1rem;
    border-radius: 30px;
    margin-bottom: 1rem;
  }
  . weather_temperature {
     font-size: 5rem;
  }
  . weather_minmax {
    display: flex;
    justify-content: center;
```

```
}

. weather_minmax p {

    font-size: 14px;

    margin: 0.5rem;

}

. weather_info {

    display: grid;

    grid-template-columns: repeat (2, 1fr);

    grid-gap: 1rem;

    margin-top: 3rem;

}

. weather_card {

    display: flex;

    align-items: center;

    background: #1e1e1e;

    padding: 1rem;

    color: #000000;

     border-radius: 10px;

    border: 1px solid rgba (255, 255, 255, 0.38);

    background: rgba (255, 255, 255, 0.21);
    backdrop-filter: blur(15px);

}

. weather_card i {

    font-size: 1.5rem;

    margin-right: 1rem;

}

. weather_card p {

     font-size: 14px;

}

@media (max-width: 936px) {

.container { width: 90%;

}

. weather_header {

    flex-direction: column;

    gap: 1rem;

    flex-wrap: wrap;


 }

 . weather_units {

     margin-top: 1rem;
 }

 }

 @media (max-width: 400px) {

 . weather_info {

     grid-template-columns: none;

 }

 }
```

10.3 Java Script

```
const cityElement = document. querySelector (". weather_city");

const datetimeElement = document. querySelector weather_datetime");

const forecastElement = document. querySelector (". weather_forecast");

const iconElement = document. querySelector (". weather_icon");

const temperatureElement = document. querySelector (". weather_temperature");

const minMaxElement = document. querySelector (". weather_minmax");

const realFeelElement = document. querySelector (". weather_realfeel");

const humidityElement = document. querySelector (". weather_humidity");

const windElement = document. querySelector (". weather_wind");

const pressureElement = document. querySelector (". weather_pressure");

const searchForm = document. querySelector (". weather_search");

const searchInput = document. querySelector (". weather__searchform");

 const locationBtn = document. querySelector (". weather_location-btn");

 const celsiusUnit = document. querySelector (". weather_unit_celsius");

 const fahrenheitUnit = document. querySelector (". weather_unit_farenheit");
```

```
  const apiKey = "e9b258b833970b94a95b9403213f3b7c";

  const baseUrl = "https://api.openweathermap.org/data/2.5/weather";

  let units = "metric";
  searchForm.addEventListener("submit", (e) => {
   e. preventDefault ();
   const city = searchInput. value.
 Trim ();if (city! == "") {
 fetchWeatherData(city);
   }
   searchInput.value = "";
  });
  locationBtn.addEventListener("click", getCurrentLocationWeather);
function getCurrentLocationWeather () {

   if (navigator. geolocation) {
     navigator. geolocation.
   getCurrentPosition ((position) => {
      const latitude = position. coords.
    latitude; const longitude = position.
    coords. longitude;
      fetchWeatherByCoordinates (latitude, longitude);
     },
     (error) => {
      console.log (error. message);
     }
    );
   } else {
    console.log ("Geolocation is not supported by this browser.");
   }
  }


  async function fetchWeatherByCoordinates (latitude, longitude) {
```

```
  try {
    const response = await fetch (
      `${baseUrl}?lat=${latitude}&lon=${longitude}&appid=${apiKey}&units=${units}`
    );
    if (! response) {
      throw new Error ("Weather data not available.");
    }
    const data = await response.
  Json ();
  updateWeatherInfo(data);
    } catch
  (error) {
  console.log(er
  ror);
  }
}
  celsiusUnit.addEventListener("click", () => {
  if (units!== "metric") {
    units = "metric";
    fetchWeatherData (cityElement.textContent); // Update weather data with new unit
  }
});
  fahrenheitUnit.addEventListener("click",
() => {if (units! == "imperial") {
    units = "imperial";
    fetchWeatherData (cityElement.textContent); // Update weather data with new unit
```

```
   }

  });
   async function
  fetchWeatherData(city) {try {
     const response = await fetch (

       `${baseUrl}?q=${city}&appid=${apiKey}&units=${units}`

     );

  if (! response. ok) {

     throw new Error ("Weather data not available.");

   }

   const data = await
   response.json ();
   updateWeatherInfo(data);
  } catch (error)
   {
   console.log(er
   ror);
  }

}

function updateWeatherInfo(data) {

  cityElement.textContent = data.name;

  datetimeElement.textContent = getCurrentTime ();
  forecastElement.textContent = data. Weather [0]. description;

  iconElement.innerHTML = `<img src="http://openweathermap.org/img/wn/$ {data.
weather[0]. icon}.png" alt="Weather Icon">`;
  temperatureElement.innerHTML = `${Math. Round (data. main. temp)}
   &#176; ${units === "metric"? "C": "F"
  }`;
  minMaxElement.innerHTML = `<p>Min: ${Math.round(data. main. temp_min)}
    &#176; ${units === "metric"? "C": "F"
  } </p><p>Max: ${Math.round(data. main. temp_min)}
    &#176; ${units === "metric"? "C": "F"
  } </p>`;

  realFeelElement.innerHTML = `<p>${Math.round(data. main. feels_like)}
    &#176; ${units === "metric"? "C": "F"
  } </p>`;

  humidityElement.textContent = `${data. main.
  humidity}%`;windElement.textContent = `${data. wind.
  Speed} ${
    units === "imperial"? "mph": "m/s"

  }`;

  pressureElement.textContent = `${data. main. Pressure} hPa`;

}
function getCurrentTime () {
  const date = new Date ();
  return date.
  toLocaleString ();
}
window. addEventListener ("load", () => { fetchWeatherData("Hyderabad");
 datetimeElement.textContent = getCurrentTime ();
});
```

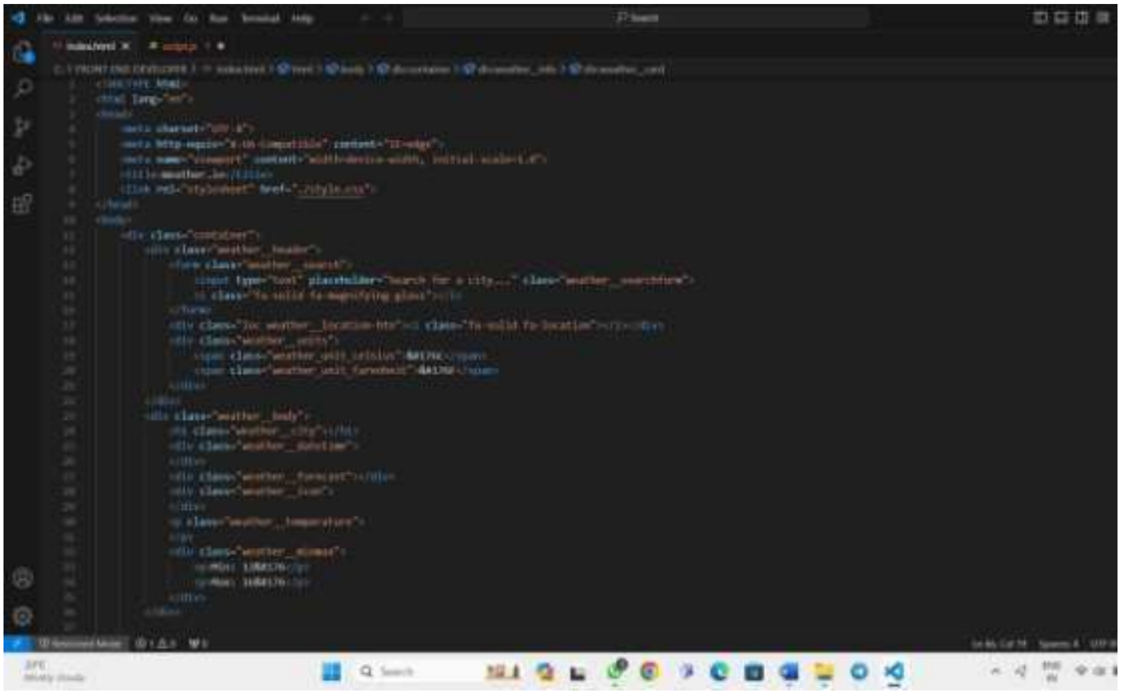
PYTHON CODE

```
import requests

def get_weather(city, api_key):

    base_url = "http://api.openweathermap.org/data/2.5/weather?"
    complete_url = f"{base_url}q={city}&appid={api_key}&units=metric"
    response = requests.get(complete_url)
    data = response.json()
    if data["cod"] != "404":
        main = data["main"]
        weather = data["weather"][0]
        temperature = main["temp"]
        feels_like = main["feels_like"]
        description = weather["description"]
        return f"Temperature: {temperature}°C\nFeels like: {feels_like}°C\nDescription: {description}"
    else:
        return "City not found"
if__name__== "_main_":
    city = input("Enter city name: ")
    api_key = "YOUR_API_KEY_HERE"
    print(get_weather(city, api_key))
```

11. PHOTOS

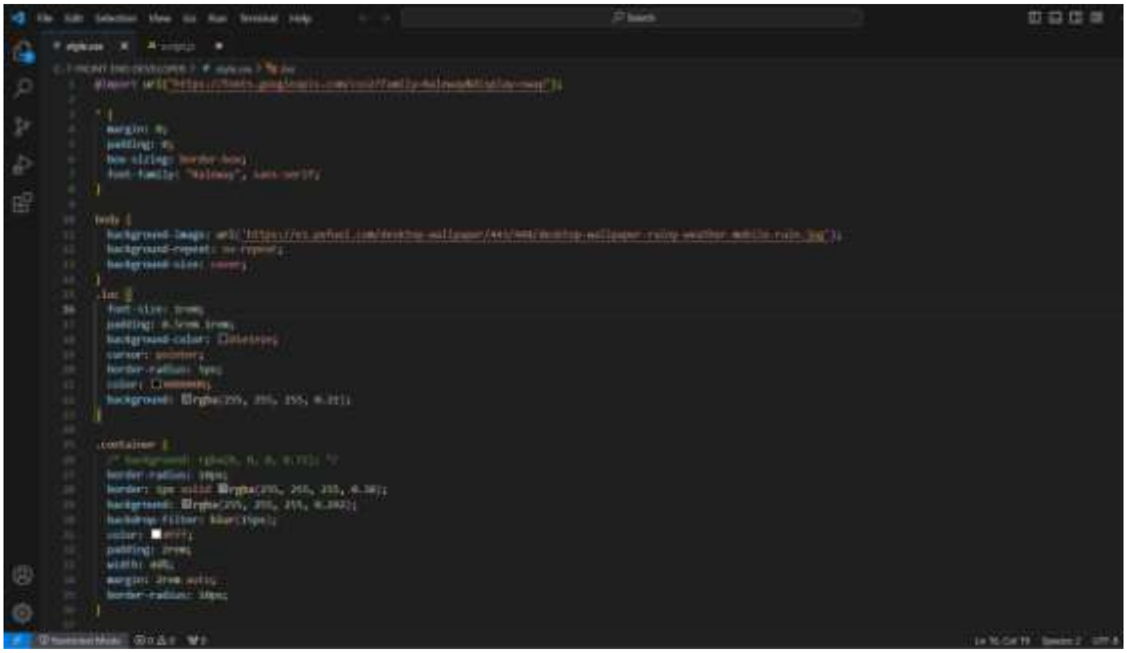
11.1 Web page file locations

1. Html file location:


File Edit Selection View Go Run Terminal Help
Search
index.html
script.js
C: > FRONT END DEVELOPER > index.html > html > body > div.container > div.weather_info > div.weather_card
Restricted Mode
Search

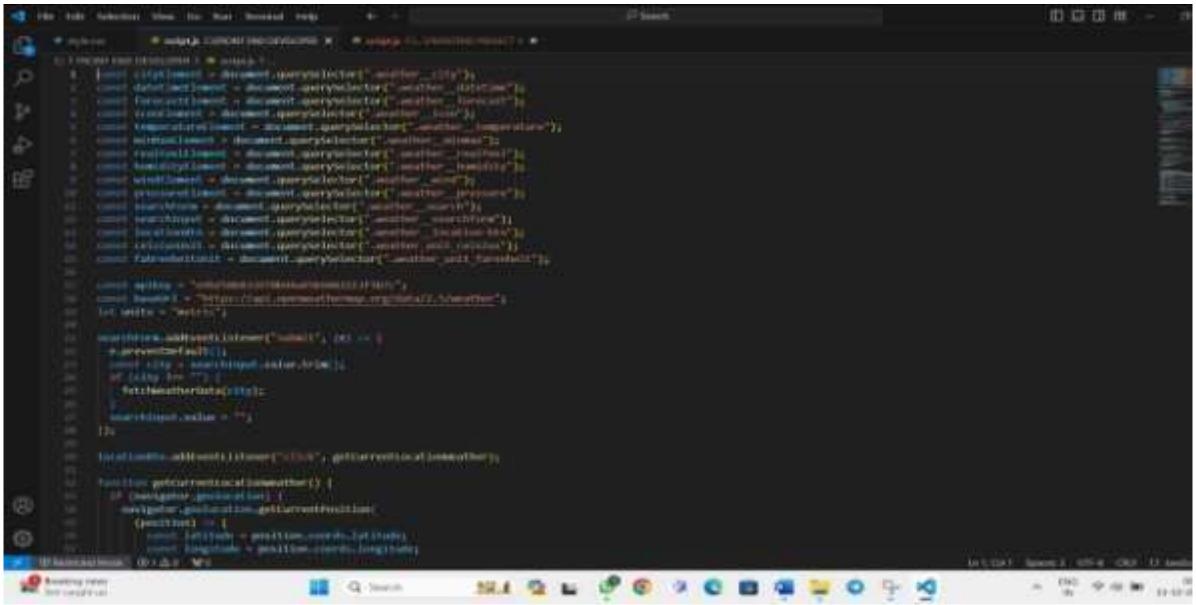
```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>weather.io</title>
    <link rel="stylesheet" href="./style.css">
</head>
<body>
    <div class="container">
        <div class="weather__header">
            <form class="weather__search">
                <input type="text" placeholder="Search for a city..." class="weather__searchform">
                <i class="fa-solid fa-magnifying-glass"></i>
            </form>
            <div class="loc weather__location-btn"><i class="fa-solid fa-location"></i></div>
            <div class="weather__units">
                <span class="weather_unit_celsius">&#176C</span>
                <span class="weather_unit_farenheit">&#176F</span>
            </div>
        </div>
        <div class="weather__body">
            <h1 class="weather__city"></h1>
            <div class="weather__datetime">
            </div>
            <div class="weather__forecast"></div>
            <div class="weather__icon">
            </div>
            <p class="weather__temperature">
            </p>
            <div class="weather__minmax">
                <p>Min: 12&#176</p>
                <p>Max: 16&#176</p>
            </div>
        </div>
```

2. CSS file location:


File Edit Selection View Go Run Terminal Help
Search
style.css
script.js
C: > FRONT END DEVELOPER > style.css > .loc
Restricted Mode

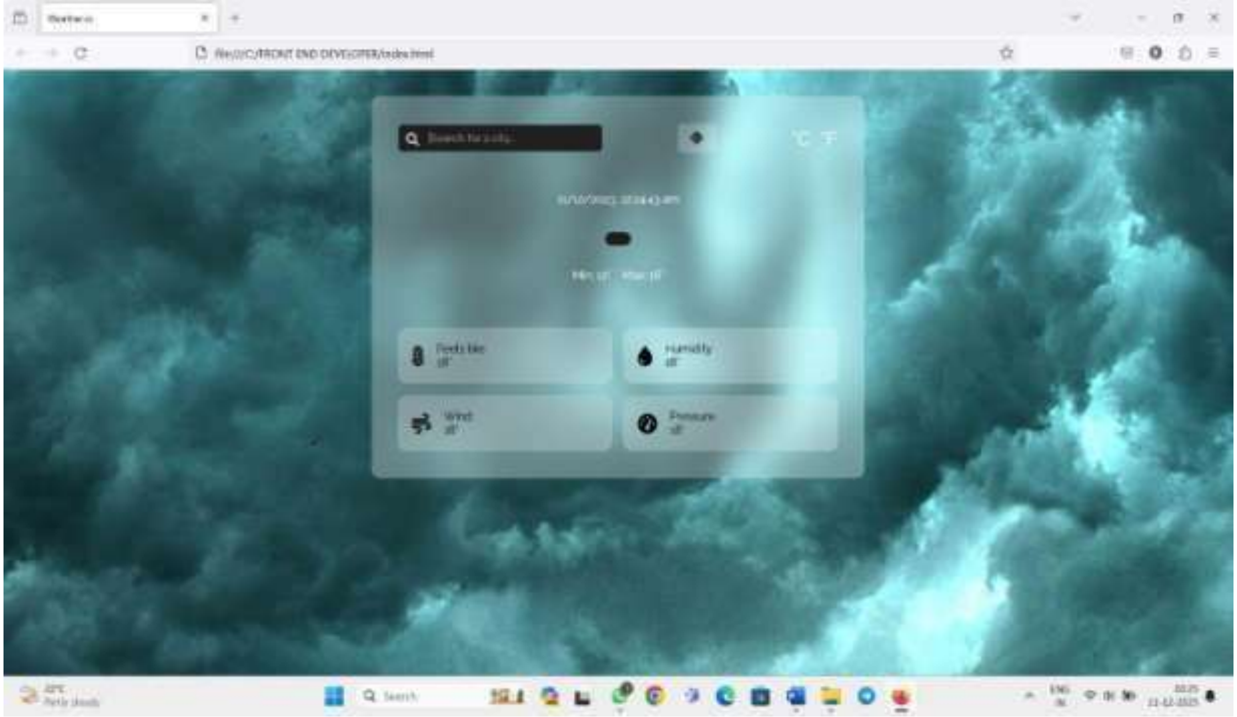
```
@import url('https://fonts.googleapis.com/css2?family=Raleway&display=swap');

* {
  margin: 0;
  padding: 0;
  box-sizing: border-box;
  font-family: "Raleway", sans-serif;
}

body {
  background-image: url('https://r1.pxfuel.com/desktop-wallpaper/441/448/desktop-wallpaper-rainy-weather-mobile-rain.jpg');
  background-repeat: no-repeat;
  background-size: cover;
}
.loc {
  font-size: 1rem;
  padding: 0.5rem 1rem;
  background-color: [illegible];
  cursor: pointer;
  border-radius: 5px;
  color: #000000;
  background: rgba(255, 255, 255, 0.21);
}

.container {
  /* background: rgba(0, 0, 0, 0.71); */
  border-radius: 10px;
  border: 1px solid rgba(255, 255, 255, 0.30);
  background: rgba(255, 255, 255, 0.202);
  backdrop-filter: blur(15px);
  color: #fff;
  padding: 2rem;
  width: 40%;
  margin: 2rem auto;
  border-radius: 10px;
}
```

3. JavaScript file location:



```
const cityElement = document.querySelector(".weather__city");
const datetimeElement = document.querySelector(".weather__datetime");
const forecastElement = document.querySelector(".weather__forecast");
const iconElement = document.querySelector(".weather__icon");
const temperatureElement = document.querySelector(".weather__temperature");
const minmaxElement = document.querySelector(".weather__minmax");
const realfeelElement = document.querySelector(".weather__realfeel");
const humidityElement = document.querySelector(".weather__humidity");
const windElement = document.querySelector(".weather__wind");
const pressureElement = document.querySelector(".weather__pressure");
const searchForm = document.querySelector(".weather__search");
const searchInput = document.querySelector(".weather__searchform");
const locationBtn = document.querySelector(".weather__location-btn");
const celsiusUnit = document.querySelector(".weather_unit_celsius");
const fahrenheitUnit = document.querySelector(".weather_unit_farenheit");

const apiKey = "[illegible]";
const baseURL = "https://api.openweathermap.org/data/2.5/weather";
let units = "metric";

searchForm.addEventListener("submit", (e) => {
  e.preventDefault();
  const city = searchInput.value.trim();
  if (city !== "") {
    fetchWeatherData(city);
  }
  searchInput.value = "";
});

locationBtn.addEventListener("click", getCurrentLocationWeather);

function getCurrentLocationWeather() {
  if (navigator.geolocation) {
    navigator.geolocation.getCurrentPosition(
      (position) => {
        const latitude = position.coords.latitude;
        const longitude = position.coords.longitude;
```

11.2 Output Page:



12. ADVANTAGES AND DISADVANTAGES

Advantages:

- Farmers can know when to plant or harvest their crops
- People can choose where and when to take their holidays to take advantages of good weather
- Surfers known when large waves are expected
- Regions can be evacuated if hurricanes or floods are expected
- Aircraft and shipping rely heavily on accurate weather forecasting
- Preparedness for temperature and precipitation changes
- Early warnings for severe weather
- Agricultural planning
- Transportation and aviation support
- Farmers can know when to plant or harvest their crops
- People can choose where and when to take their holidays to take advantages of good weather
- Surfers known when large waves are expected
- Regions can be evacuated if hurricanes or floods are expected
- Aircraft and shipping rely heavily on accurate weather forecasting

Disadvantages:

- Weather is extremely difficult to forecast correctly.
- It is expensive to monitor so many variables from many sources
- The computers needed to perform the millions of calculations necessary are expensive
- The weather forecasters get blamed if the weather is different from the forecast
- The terminology used in weather forecasting can be confusing, making it difficult for some people to understand the predictions.
- Weather forecasting relies heavily on technology, and if the technology fails or is unavailable, accurate predictions cannot be made.
- Weather forecasts are not available for many remote or sparsely populated areas, making it difficult for people in these areas to prepare for severe weather.

- Forecasting models can only make predictions based on existing data and are limited by the quality and quantity of that data.
- Forecasts are usually only accurate for a short time frame, making it difficult to plan ahead.
- Weather is extremely difficult to forecast correctly
- It is expensive to monitor-so many variables from so many sources
- The computers needed to perform the millions of calculations necessary are expensive
- The weather forecasters get blamed if the weather is different from the forecast

13. APPLICATIONS

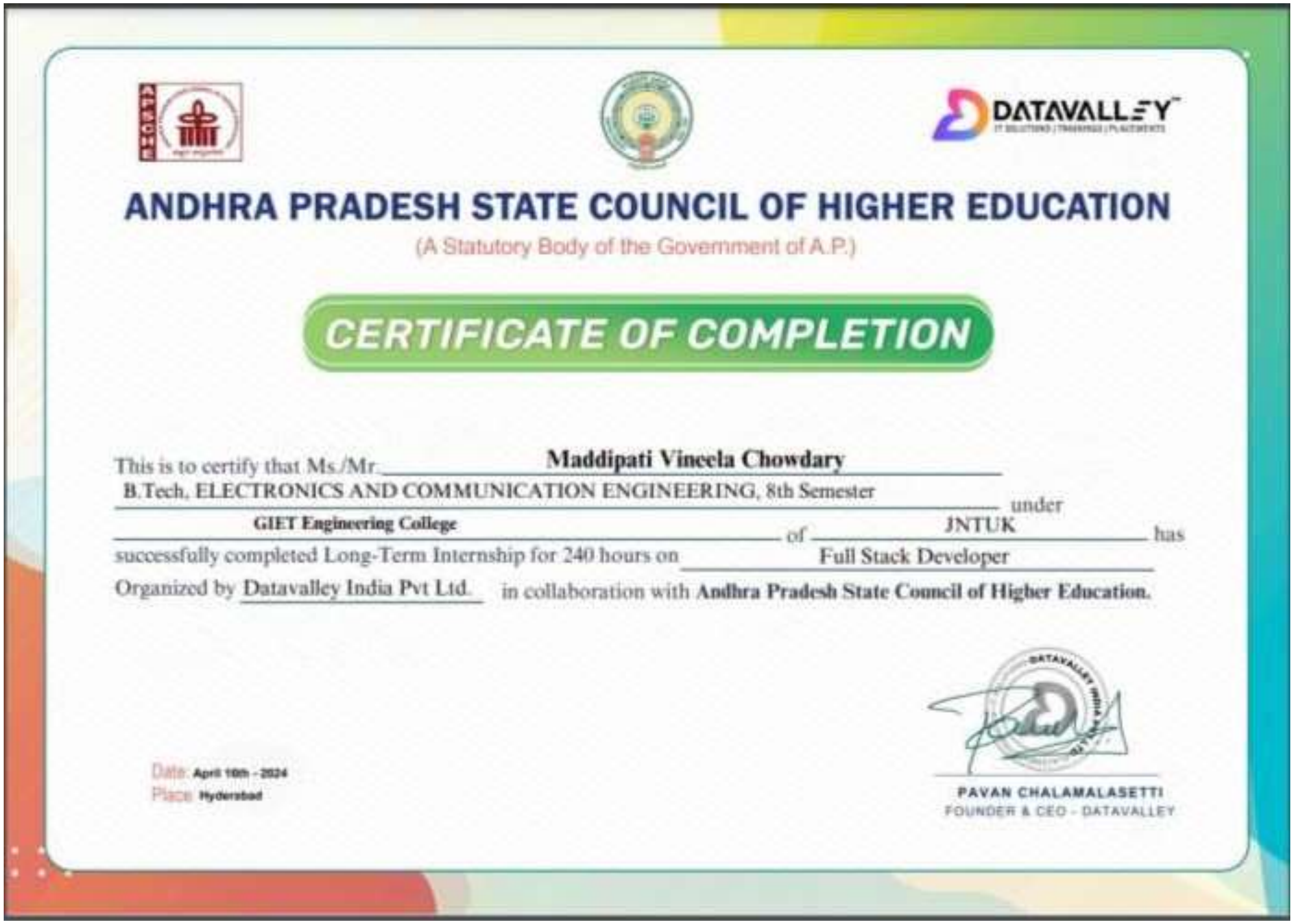
- **Agriculture:** Farmers use weather forecasts to plan irrigation, planting, and harvesting, optimizing crop yields.
- **Transportation:** Airlines, shipping companies, and road maintenance crews rely on forecasts to plan routes and schedules, minimizing disruptions due to adverse weather.
- **Energy Management:** Utilities use forecasts to anticipate demand for heating or cooling, helping them manage energy resources more efficiently.
- **Disaster Preparedness:** Early warnings of severe weather events like hurricanes, tornadoes, and floods allow communities to evacuate and take protective measures.
- **Construction:** Builders use forecasts to schedule outdoor work, reduce downtime due to weather-related delays, and ensure worker safety.
- **Tourism:** Travel companies and tourist destinations use weather forecasts to attract visitors and offer suitable activities.
- **Retail and Marketing:** Businesses use weather data to tailor marketing strategies, such as promoting weather-appropriate products.
- **Emergency Services:** First responders rely on forecasts during emergencies to plan and allocate resources effectively.

14. CONCLUSION

In the era of the global warming, research in weather measurement, monitoring and forecasting are become more and more relevant. This research demonstrates the design and implementation of an affordable mini weather monitoring system that ensures flexibility, portability, scalability and user-friendly operations which can provide data of some weather variables including temperature, humidity and pressure. With the advancement of technology weather forecasting has developed to its level best, but there is yet to develop, as far as a nature is so unpredictable. Weather forecasts are increasingly accurate and useful, and their benefits extend widely across the economy. While much has been accomplished in improving weather forecasts, there remains much room for improvement. Simultaneously, they are developing new technologies and observational networks that can enhance forecaster skill and the value of their services to their users.

15. REFERENCE

- ✓ https://www.google.com/
- ✓ https://lms.datavalley.ai/student/dashboard
- ✓ https://chat.openai.com/



DATAVALLEY

ANDHRA PRADESH STATE COUNCIL OF HIGHER EDUCATION

(A Statutory Body of the Government of A.P.)

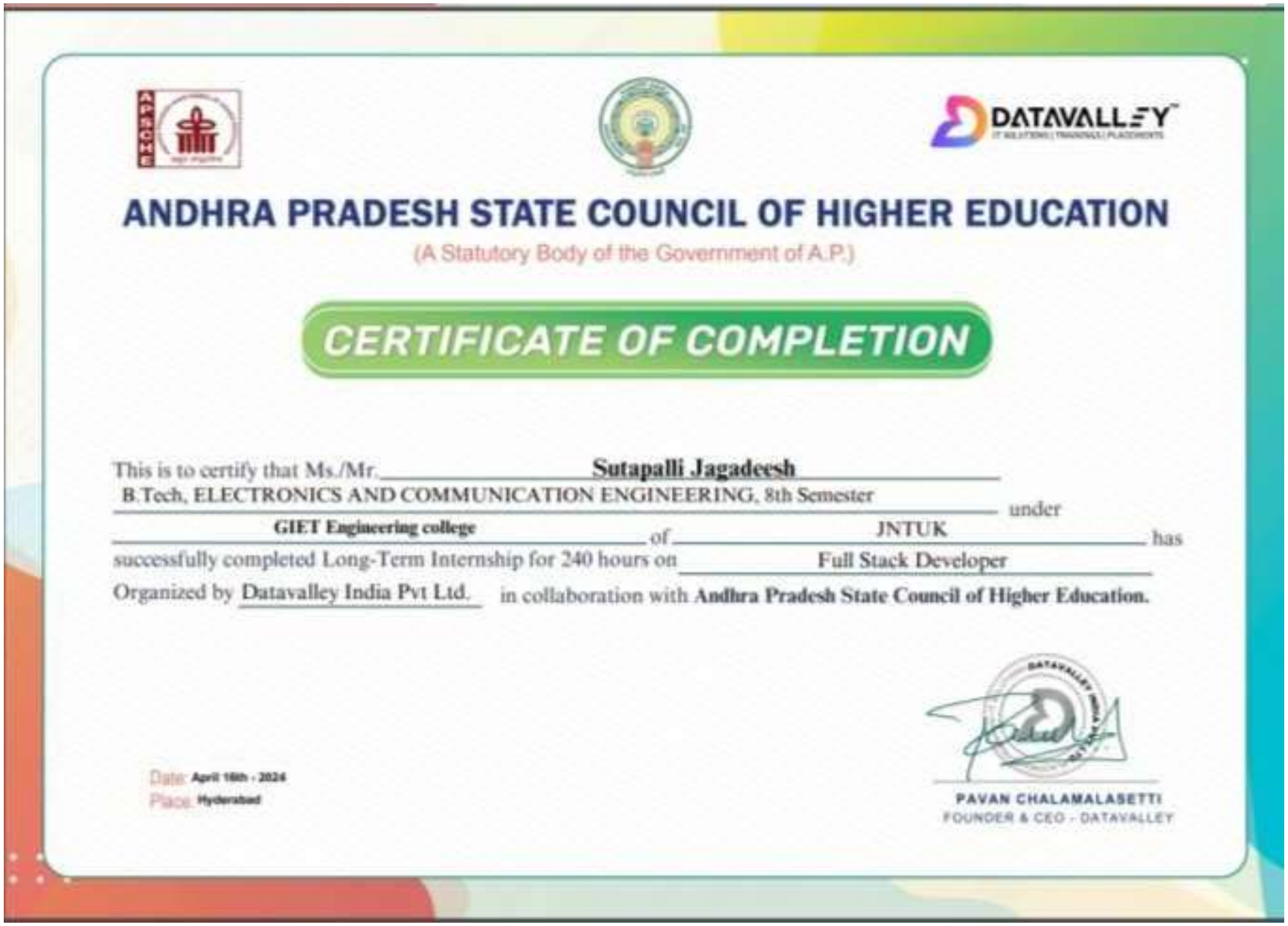
CERTIFICATE OF COMPLETION

This is to certify that Ms./Mr. Maddipati Vineela Chowdary
B.Tech, ELECTRONICS AND COMMUNICATION ENGINEERING, 8th Semester under
GIET Engineering College of JNTUK has
successfully completed Long-Term Internship for 240 hours on Full Stack Developer
Organized by Datavalley India Pvt Ltd. in collaboration with **Andhra Pradesh State Council of Higher Education.**

Date: April 10th - 2024
Place: Hyderabad

DATAVALLEY INDIA PVT LTD

PAVAN CHALAMALASETTI
FOUNDER & CEO - DATAVALLEY



ANDHRA PRADESH STATE COUNCIL OF HIGHER EDUCATION

(A Statutory Body of the Government of A.P.)

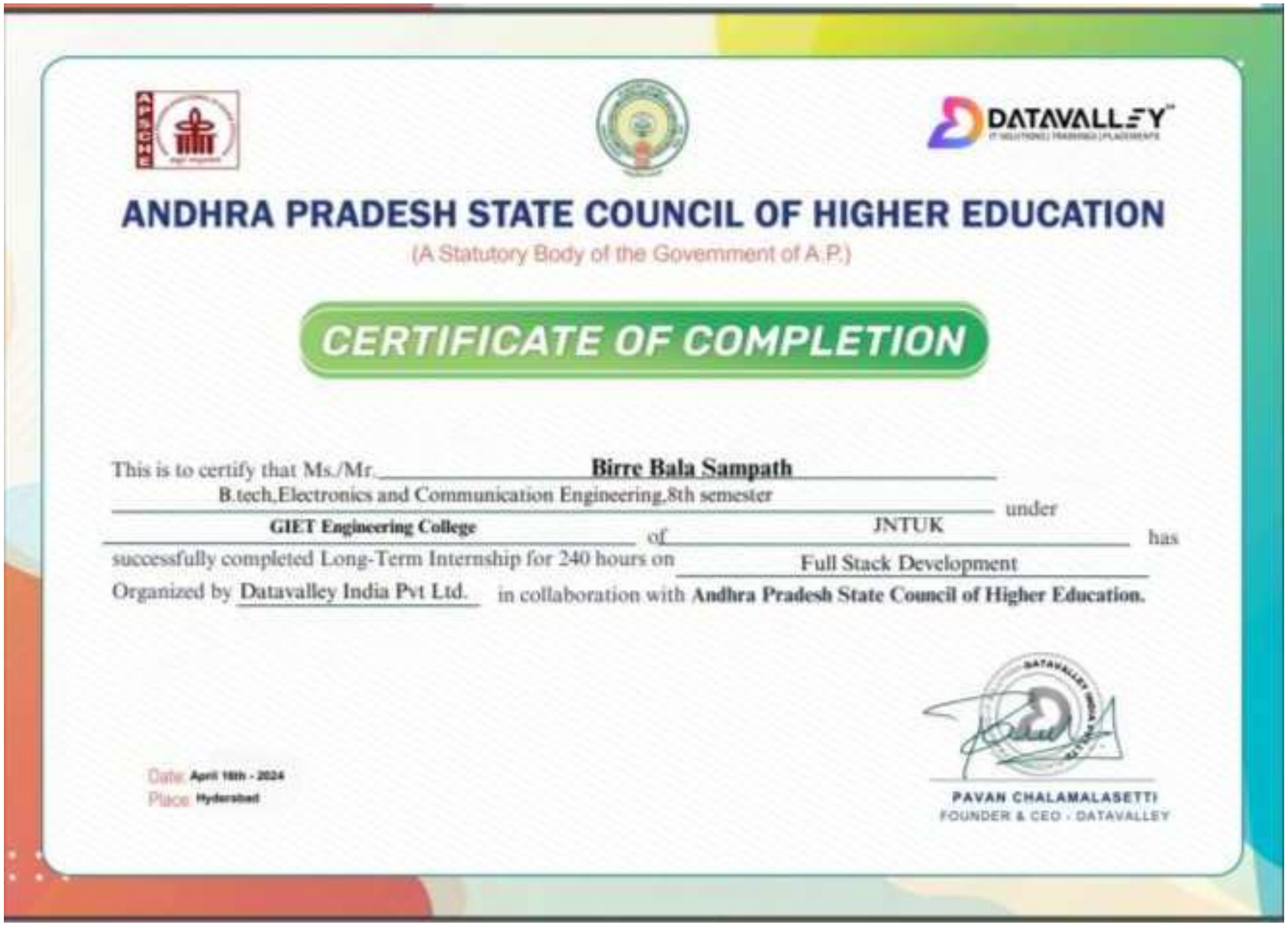
CERTIFICATE OF COMPLETION

This is to certify that Ms./Mr. **Sutapalli Jagadeesh**
B.Tech, ELECTRONICS AND COMMUNICATION ENGINEERING, 8th Semester under
GIET Engineering college of JNTUK has
successfully completed Long-Term Internship for 240 hours on Full Stack Developer
Organized by Datavalley India Pvt Ltd. in collaboration with **Andhra Pradesh State Council of Higher Education.**

Date: April 16th - 2024
Place: Hyderabad

DATAVALLEY INDIA PVT LTD

PAVAN CHALAMALASETTI
FOUNDER & CEO - DATAVALLEY



DATAVALLEY
IT SOLUTIONS | TRAINING | PLACEMENTS

ANDHRA PRADESH STATE COUNCIL OF HIGHER EDUCATION

(A Statutory Body of the Government of A.P.)

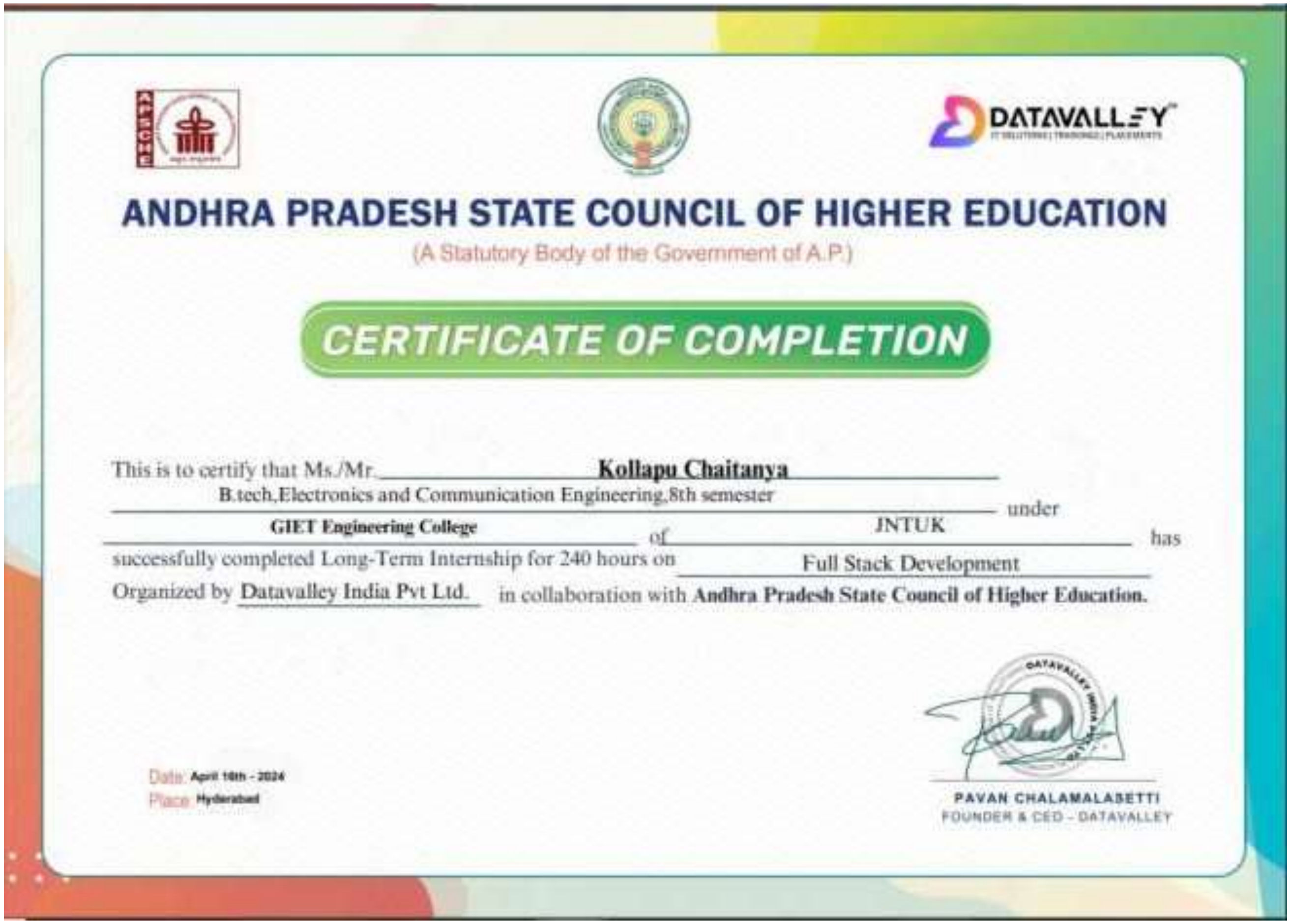
CERTIFICATE OF COMPLETION

This is to certify that Ms./Mr. **Birre Bala Sampath**
B.tech,Electronics and Communication Engineering,8th semester under
GIET Engineering College of JNTUK has
successfully completed Long-Term Internship for 240 hours on Full Stack Development
Organized by Datavalley India Pvt Ltd. in collaboration with **Andhra Pradesh State Council of Higher Education.**

Date: April 18th - 2024
Place: Hyderabad

DATAVALLEY INDIA PVT LTD

PAVAN CHALAMALASETTI
FOUNDER & CEO - DATAVALLEY



DATAVALLEY

ANDHRA PRADESH STATE COUNCIL OF HIGHER EDUCATION

(A Statutory Body of the Government of A.P.)

CERTIFICATE OF COMPLETION

This is to certify that Ms./Mr. **Kollapu Chaitanya** B.tech,Electronics and Communication Engineering,8th semester under **GIET Engineering College** of JNTUK has successfully completed Long-Term Internship for 240 hours on Full Stack Development Organized by Datavalley India Pvt Ltd. in collaboration with **Andhra Pradesh State Council of Higher Education.**

Date: April 16th - 2024
Place: Hyderabad

DATAVALLEY INDIA PVT. LTD.

PAVAN CHALAMALASETTI
FOUNDER & CEO - DATAVALLEY

ANDHRA PRADESH
STATE COUNCIL OF HIGHER EDUCATION
(A Statuory Body of the Government of Andhra Pradesh)

2nd, 3rd, 4th and 5th floors, Neeladri Towers, Sri Ram Nagar, 6th Battalion Road
Atmakur (V)Mangalagiri (M), Guntur, Andhra Pradesh, Pin - 522 503
www.apsche.ap.gov.in