

**PROJECT REPORT**

**ON**

***“Weather Forecasting”***

# Bachelor of Technology (CSE)

(Batch: 2021-2025)

**Submitted To:**

Mr. Brijesh Kumar

**Submitted By:**

Ajay Kumar (2115000090)

Alok Rai (2115000120)

Aman pratap Singh (2115000133)

Nilanshu Rai (2115000673)

1

CERTIFICATE

This is to certify that we Aman Pratap, Ajay kumar, Alok rai, Nilanshu Rai of BTech (CSE) 5th Semester from GLA University, Mathura has presented this Mini project work entitled “Weather Forecasting”, a website in partial fulfilment of the requirements for the award of the degree of Bachelor of Technology under our supervision and guidance.

# ACKNOWLEDGEMENT

It is our proud privilege to express our profound gratitude to the entire management of GLA University and the teachers of the institute for providing us with the opportunity to avail ourselves of the excellent facilities and infrastructure. The knowledge and values inculcated have proved to be of immense help at the very start of my career. Special thanks to the Hon’ble Founder, GLA University, Mathura for having provided us with an excellent infrastructure.

Sincere thanks to all my family members, seniors and friends for their support and assistance throughout the project.

1. **INTRODUCTION**

#### Objective of the System

* 1. **Justification and need for the system**

#### Advantage of system

* 1. **Problem Identification With Previous Systems**

### REQUIREMENT ANALYSIS

#### Analysis Study

* 1. **User Requirement**
  2. **Final Requirements**

### DESIGN OF THE SYSTEM

#### Hardware , Software requirements

* 1. **System requirements**

#### Design Requirement

* 1. **DFDs**

### ` 4. IMPLEMENTATION & CODING

#### Operating System

* 1. **Languages**

### TESTING & TEST RESULTS

#### Implementation and Testing

* 1. **Validation Testing**

### CONCLUSION

#### Conclusion

* 1. **Biblography**

# Chapter 1 INTRODUCTION

In the domain of weather forecast websites, several previous works related systems have been developed to provide accurate and timely weather information. These systems often employ advanced technologies and methodologies to enhance the accuracy and usability of weather forecasts. Here are some notable aspects of previous work and related systems in the context of weather forecast websites. For example, it might be observed that if the sunset was particularly red, the following day often brought fair weather. However, not all of these predictions prove reliable.

# OBJECTIVE OF THE SYSTEM:-

This project will serve the following objectives:-

* + 1. User friendly interface and easy to use
    2. User can check the temperature of any particular region
    3. It will also show humidity, wind speed and cloud

# JUSTIFICATION AND NEED FOR THE SYSTEM:-

A weather forecasting website is essential for informed decision-making in public safety, agriculture, transportation, and various industries. It provides critical data for planning and safety measures in the face of natural disasters, optimizing agricultural practices, ensuring safe travel routes, and aiding energy grid management. The website also plays a key role in event planning, tourism, and community awareness. Beyond practical applications, it serves as a vital resource for meteorologists, researchers, and educational purposes, contributing to overall preparedness and resilience in the face of changing weather patterns.

# Advantages of the system:-

1. Life-Saving Alerts:-

- The system issues crucial early warnings, saving lives during natural disasters.

1. Global Food Security Champion:-

- Optimizing agriculture, it ensures global food security by empowering farmers to strategically plan and adapt to weather challenges, fostering a resilient food supply.

1. Tourism And Hospitality Impact:-

Influences the tourism industry by helping tourists plan trips based on expected weather and assisting hospitality businesses in managing operations effectively.

# Problem Identification With Previous Systems:-

Before we begin a new system it is important to study the systems available , that will be improved or replaced. We need to analyze how this system uses hardware, software, network and peopleresources to convert data resources, such as transaction data, into information products.

Following are the problems associated with the previous project which led to the creation of the proposed project:-

1. **Not user-friendly:** The existing system is not user-friendly because the information like humidity cloud and wind etc. are not in one place.
2. **Not a good UI**: The user interface of the previous systems are not that good.

**6**

# Chapter 2 REQUIREMENT ANALYSIS

## Functional Requirements:-

Functional requirements are the requirements that describe the functionalities of the system elements. It may involve functional user requirements or functional system requirements.

For example:

The operator shall be able to input the region to the system to view the desired weather parameters.

The system shall provide the following weather parameters: temperature, pressure, wind speed ,date / time and humidity.

## ANALYSIS STUDY:-

#### Lower Installation Charges:-

We neither require any high-configuration systems for the smooth running of the server program nor do we require any high-configuration systems for the smooth running of a client program. This website is designed with ease to support any ordinary system having an internet connection.

#### Availability:-

The availability of the system is that the website will be active on the Internet and people will be able to browse it.

## Feasibility Study:-

All projects are feasible if they have unlimited resources and infinite time. But the development of software is plagued by the scarcity of resources and difficult delivery rates. It is necessary and prudent to evaluate the feasibility of a project at the earliest possible time. The three considerations are involved in the feasibility analysis.

## Technical Feasibility:-

Technical feasibility centres on the existing mobile system (hardware, software…etc) and to what extent it can support the proposed addition if the budget is a serious constraint, then the project is judged not feasible. The technical feasibilities are an important role in our project because here we’re using HTML,CSS and JavaScript . It requires Visual Studio Code(software) to develop this application. A easily available software and easy to use.

## Economical Feasibility:-

This procedure is to determine the benefits and savings that are expected from a candidate system and compare them with cost. If the benefits outweigh the cost then the decision is made to design and implement the system. Otherwise, further justification or alterations in proposed systems have to be made if it is having a chance of being approved. This is an ongoing effort that improves any feasibility costs spent on this project because here we’re using open-source environments.

## Operational Feasibility:-

People are inherently resistant to change and mobiles have been known to facilitate change. There is no need of technical background is required to work on this Website . All the information needed can be seen with just one click .

# USER REQUIREMENTS:-

The system specifications that a user may want are as follows:-

* + 1. It should be easy to use and understand
    2. Accuracy and Reliability
    3. Should provide a good user interface
    4. Real Time Weather Alerts

# Final Requirements:-

#### Precision and Trust:- Accurate and reliable forecasts for confident decision-making

1. **Intuitive Accessibility:- User-friendly interface and mobile access for seamless interaction.**

#### Real-Time Updates:- Timely, real-time information on changing weather conditions.

1. **Personalized Alerts:- Customizable location-specific alerts for individual preferences.**

#### Comprehensive Data:- Detailed weather information supporting nuanced decision-making.

1. **Severe Weather Alerts:- Immediate and accurate alerts for enhanced safety measures.**

#### Historical Insights:- Access to historical weather data and trends for research and planning.

1. **Integration Capabilities:- Seamless integration with other systems for diverse user needs.**

#### Educational Resources:- Inclusion of educational content for enriched user understanding.

1. **Multilingual Support:- Accessibility for users in diverse linguistic regions.**
2. **Reliable Support:- Responsive customer support for a positive user experience.**

# Chapter 3

**DESIGN OF THE SYSTEM**

## Software requirements:-

|  |  |
| --- | --- |
| Platform | Platform Independent |
| The Operating System | Independent |
| Location & Services | Dependent |
| Front-End Tool | HTML, CSS |
| API | Open Weather MapAPI |
| Extra Technologies | Java Script |

* 1. **System Requirements:-**

To know the detailed system requirements an SRS has to be prepared. Software requirement specification abbreviated as SRS is a means of translating the idea of files into a formal document. The main features of SRS include:-

* + - Establishing the basis for an agreement between the client and the developer.
    - Producing a reference for validation of the final product. SRS assist clients in determining if the software meets their requirement.

Mainly there are six requirements which an SRS must satisfy:-

1. It should specify the external behaviour.
2. It should specify the constraints.
3. It should be easy to change.
4. It should be a reference tool.
5. It should record throughout the lifecycle.
6. It should have the capacity to expect an undesired event.

## Functional Requirements :-

Functional requirements are the requirements that describe the functionalities of the system elements. It may involve functional user requirements or functional system requirements.

For example:-

The operator shall be able to input the region to the system to view the desired weather parameters.

The system shall provide the following weather parameters: temperature, pressure, wind speed & direction, rainfall, and humidity.

## Design Requirements :-

The main objectives of **input design** are:-

1. Controlling the amount of input.
2. Keeping the process simple.
3. The best thing in the input design is to achieve all the objectives mentioned in the simplest manner possible.

The main objectives of **output design** are:-

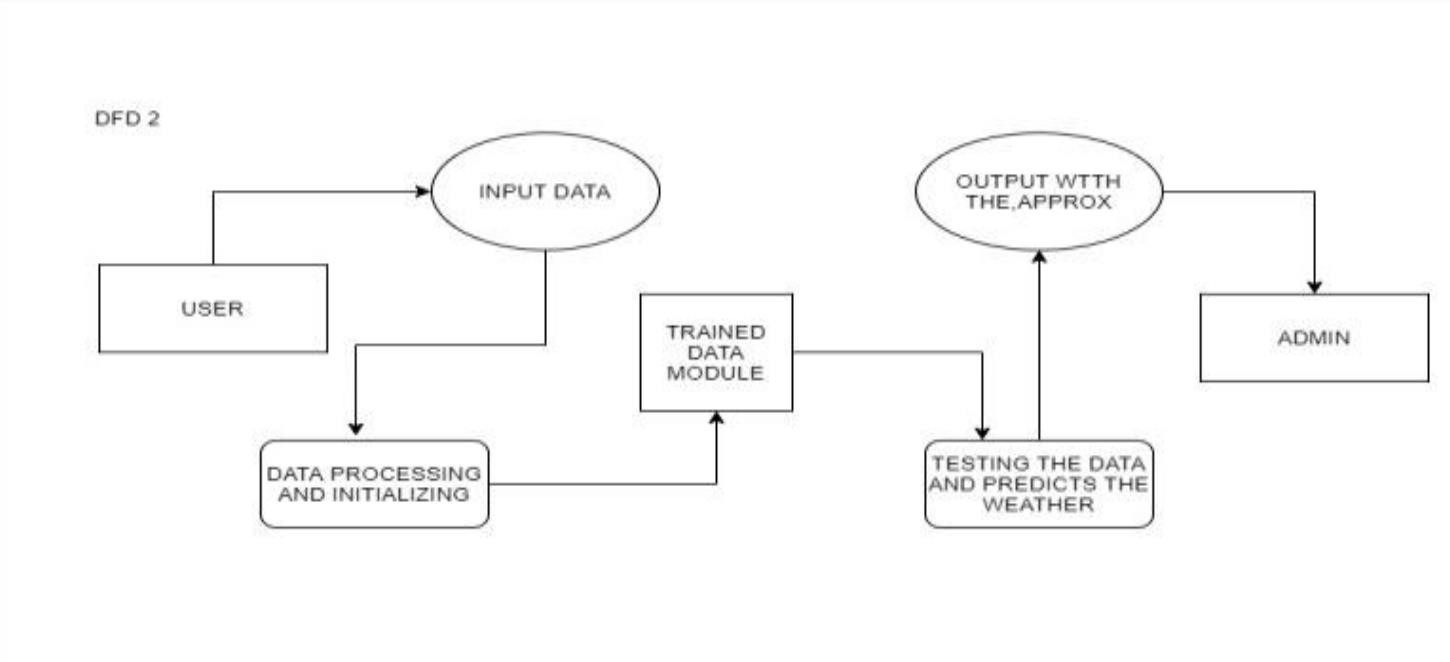
(a) Identifying the specific outputs.

The primary goal of the system analysis is to improve the efficiency of the existing system. For that the study of specification of the requirements is very essential. For the development of the new system, a preliminary survey of the existing system will be conducted. Investigation done whether the upgradation of the system into an application program could solve the problems and eradicate the inefficiency of the existing system

# DATA FLOW DIAGRAM (DFD):-

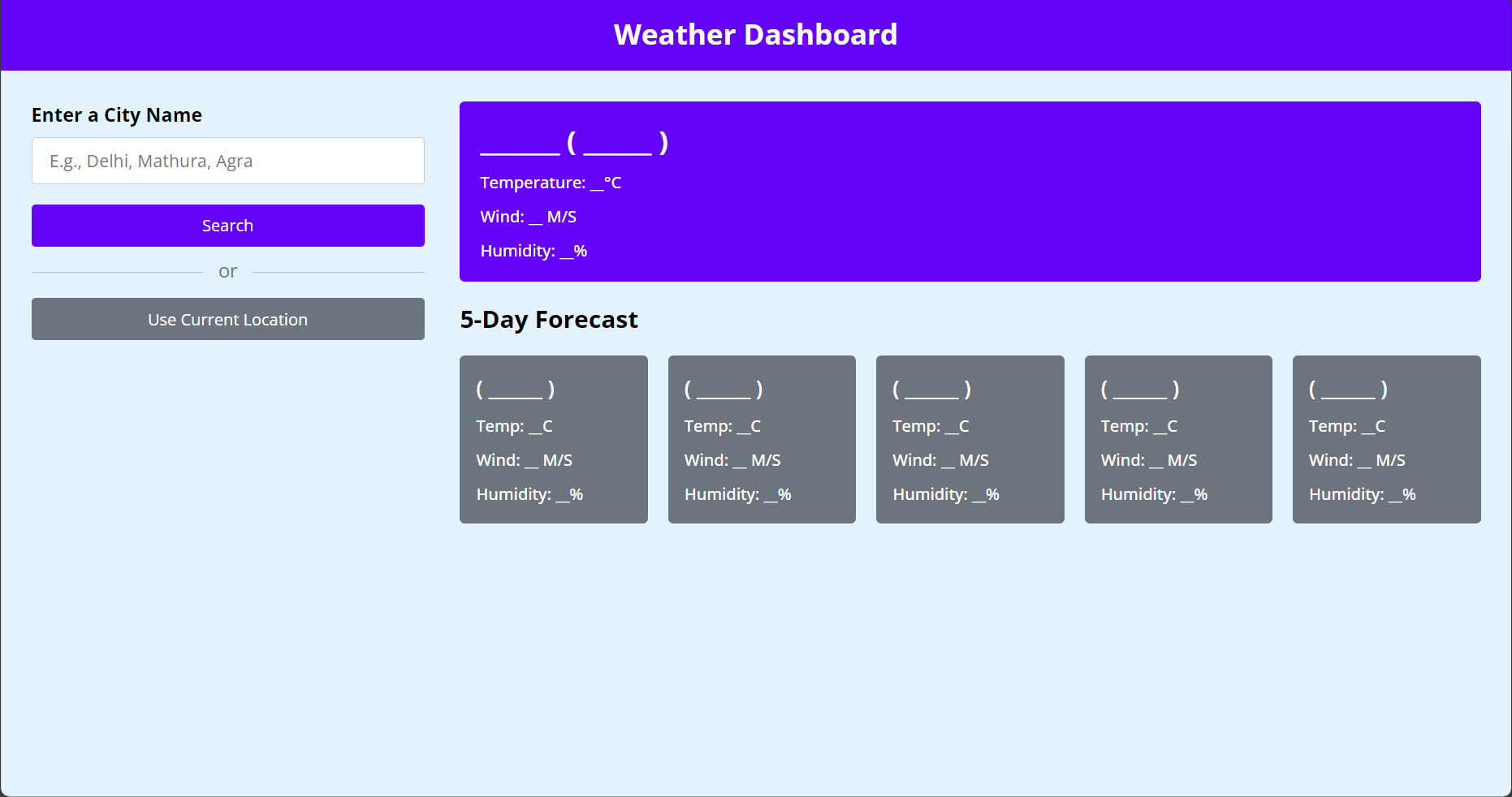
A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system, modelling its process aspects. Often they are a preliminary step used to create an overview of the system which can later be elaborated. DFDs can also be used for the visualization of data processing (structured design).

A DFD shows what kinds of information will be input into and output from the system, where the data will come from and go to, and where the data will be stored. It does not show information about the timing of processes, or information about whether processes will operate in sequence or in parallel (which is shown on a flowchart).



12

# Chapter 4 IMPLEMENTATION AND CODING



* 1. **Operating System:-**

**Platform Independent:** Since the project is done completely in H tml , C ss , J a v a S c r i p t and A pi , it also executes the main properties of the language. The website is platform-independent.

# Languages used:-

## HTML:-

HTML (HyperText Markup Language) plays a crucial role in creating the structure and content of a weather forecasting website. HTML is used to define the layout of web pages and organize the various elements that make up the content. Here are some key roles of HTML in the context of a weather forecasting website

**Forms:** HTML includes form elements such as <input> and <button>. In a weather forecasting website, you might have a form where users can input the city name to get the weather forecast. This allows for user interaction on the webpage.

**Links:** HTML includes the <a> tag for creating hyperlinks. You can use this to link to other pages, external weather APIs, or additional resources related to weather forecasting.

**Embedding Media:** HTML allows you to embed media elements like images and videos. You can use this to display weather-related images, maps, or any visual content.

**Lists:** HTML provides tags for creating ordered <ol> or unordered <ul> lists. This can be useful for presenting information in a structured manner, such as a list of upcoming weather forecasts.

## CSS:-

CSS (Cascading Style Sheets) plays a crucial role in a weather forecasting website by providing the means to control the presentation and layout of the HTML elements. Here are some key roles of CSS in the context of a weather forecasting website:

**Layout and Positioning:** CSS is used to control the layout and positioning of elements on the webpage. This includes setting the size and positioning of headers, sections, and other elements to create a visually appealing and organized layout.

**Typography and Fonts:** CSS is used to control the typography on the website, including font size, color, and style. This ensures that text content, such as weather information, is presented in a readable and visually appealing manner.

**Colors and Styling:** CSS allows you to define colors for various elements on the page, helping to create a cohesive and visually pleasing design. This can include background colors, text colors, borders, and other stylistic elements.

**Responsive Design:** CSS is crucial for making a website responsive, ensuring that it looks good and functions well on various devices and screen sizes. Media queries can be used to adjust the styling based on factors like screen width.

**Animations and Transitions:** CSS allows for the creation of animations and transitions, enhancing the user experience. This can be used to add subtle effects to elements on the page, making the website more engaging.

**Flexibility and Grid Layouts:** CSS provides flexbox and grid layout systems, allowing for more sophisticated and flexible page layouts. This is especially useful for arranging weather information in a structured and visually appealing manner.

## API:-

API is an abbreviation for Application Programming Interface which is a collection of communication protocols and subroutines used by various programs to communicate between them. A programmer can make use of various API tools to make their program easier and simpler. Also, an API facilitates pro- grammers with an efficient way to develop their software programs. Thus in simpler terms, an API helps two programs or applications to communicate with each other by providing them with the necessary tools and functions. It takes the request from the user and sends it to the service provider and then again sends the result generated from the service provider to the desired user.

A developer extensively uses APIs in his software to implement various features by using an API call without writing complex codes for the same. We can create an API for an operating system, database system, hardware system, JavaScript file, or similar object-oriented files. Also, an API is similar to a GUI(Graphical User Interface) with one major difference. Unlike GUIs, an API helps software develop- ers to access web tools while a GUI helps to make a program easier to understand by users.

# Chapter 5

**TESTING & TEST RESULTS**

#### Implementation and Testing

A software system test plan is a document that describes the objectives, scope, approach and focus of software testing effort. The process of preparing a test plan is a usual way to think the efforts needed to validate the acceptability of a software product. The complete document will help people outside the test group understand the "WHY" and "HOW" product validation. It should be through enough to be useful but not so through that no one outside the test group will read it.

#### Introduction

Testing is the process of running a system with the intention of finding errors. Testing enhances the integrity of a system by detecting deviations in design and errors in the system. Testing aims at detecting error-prone areas. This helps in the prevention of errors in a system. Testing also adds value to the product by conforming to the user requirements.

The main purpose of testing is to detect errors and error prone areas in a system. Testing must be through well planned. A partially tested system is to detect errors and error prone areas in a system. Testing must be through well planned. A partially tested system is as bad as an untested system. And the price of an untested and under tested system is high.

#### Objectives of Testing

The objective our test plan is to find and report as many bugs as possible to improve the integrity of our program. Although exhaustive testing is not possible, we will exercise a broad range of tests to achieve our goal. Our user interface to utilize these functions is designed to be user-friendly and provide easy manipulation of the tree. The application will only be used as a demonstration tool, but we would like to ensure that it could be run from a variety of platforms with little impact on performance or usability.

#### Process Overview

The following represents the overall flow of the testing process: Identify the requirements to be tested. All test cases shall be derived using the current Program Specification. Identify which particular test(s) will be used to test each module. Review the test data and test cases to ensure

that the unit has been thoroughly verified and that the test data and test cases are adequate to verify proper operation of the unit Identify the expected results for each test. Document the test case configuration, test data, and expected results. Document the test data, test cases, and test configuration used during the testing process. This information shall be submitted via the Unit/System Test Report (STR). Successful unit testing is required before the unit is eligible for component integration/system testing. Unsuccessful testing requires a Bug Report Form to be generated. This document shall describe the test case, the problem encountered, its possible cause, and the sequence of events that led to the problem. It shall be used as a basis for later technical analysis.

#### Test Cases

Test case is an object for execution for other modules in the architecture does not represent any interaction by itself. A test case is a set of sequential steps to execute a test operating on a set of predefined inputs to produce certain expected outputs. There are two types of test cases:- manual and automated. A manual test case is executed manually while an automated test case is executed using automation.

In system testing, test data should cover the possible values of each parameter based on the requirements. Since testing every value is impractical, a few values should be chosen from each equivalence class. An equivalence class is a set of values that should all be treated the same.

Ideally, test cases that check error conditions are written separately from the functional test cases and should have steps to verify the error messages and logs. Realistically, if functional test cases are not yet written, it is ok for testers to check for error conditions when performing normal functional test cases. It should be clear which test data, if any is expected to trigger errors.

#### Testing Steps

A strategy for software testing must accommodate low-level tests that are necessary to verify that a small source code segment has been correctly implemented as well as high-level tests that validate major system functions against customer requirements. A strategy must provide guidance for the practitioner and a set of milestones for the manager. Because the steps of the test strategy occur at a time when deadline pressure begins to rise, progress must be measurable and problems must surface as early as possible. Following testing techniques are well known and the same strategy is adopted during this project testing.

#### Unit testing

Unit testing focuses verification effort on the smallest unit of software design- the software component or module. The unit test is white-box oriented. The unit testing implemented in every module of Weather Website. By giving correct manual input to the system, the data’s are stored in database and retrieved. If you want required module to access input or gets the output from the End user. any error will accrued the time will provide handler to show what type of error will accrued.

#### Integration Testing

Data can be lost across an interface. One module can have an adverse effect on another, sub functions, when combined, may not be linked in desired manner in major functions. Integration testing is a systematic approach for constructing the program structure, while at the same time conducting test to uncover errors associated within the interface.

#### Performance Testing

Performance testing is designed to test the run-time performance of software within the context of an integrated system. Performance testing occurs throughout all steps in the testing process. Even at the unit level, the performance of an individual module may be assessed as white-box tests are conducted. This project reduce attendance table, codes. It will generate report fast. No have extra time or waiting of results .entered correct data will show result few millisecond. Just used only low memory of our system. Automatically do not getting access at software.

#### Validation

At the culmination of the integration testing, Software is completely assembled as a package. Interfacing errors have been uncovered and corrected and a final series of software test begin in validation testing. Validation testing can be defined in many ways, but a simple definition is that the validation succeeds when the software functions in a manner that is expected by the customer. After validation test has been conducted, one of the three possible conditions exists.

1. The function or performance characteristics confirm to specification and are accepted.
2. A deviation from specification is uncovered and a deficiency lists is created.
3. Proposed system under consideration has been tested by using validation test and found to be working satisfactory.

#### White Box Testing

In white box testing, the UI is bypassed. Inputs and outputs are tested directly at the code level and the results are compared against specifications. This form of testing ignores the function of the program under test and will focus only on its code and the structure of that code. Test case designers shall generate cases that not only cause each condition to take on all possible values at least once, but that cause each such condition to be executed at least once. To ensure this happens, we will be applying Branch Testing. Because the functionality of the program is relatively simple, this method will be feasible to apply.

#### Black box testing

Black box testing typically involves running through every possible input to verify that it results in the right outputs using the software as an end-user would. We have decided to perform Equivalence Partitioning and Boundary Value Analysis testing on our website.

#### System Testing

The goals of system testing are to detect faults that can only be exposed by testing the entire integrated system or some major part of it. Generally, system testing is mainly concerned with areas such as performance, security, validation, load/stress, and configuration sensitivity.

#### Output Testing

After performing the validation testing, the next step is output testing of the proposed system, since no system could be useful if it does not produce the required output in a specific format. The output format on the screen is found to be correct. The format was designed in the system design time according to the user needs. For the hardcopy also, the output comes as per the specified requirements by the user. Hence output testing did not result in any correction for the system.

#### User Acceptance Testing

User acceptance of a system is the key factor for the success of any system. The system under consideration is tested for the user acceptance by constantly keeping in touch with the prospective system users at the time of developing and making changes whenever required.

This is done in regard to the following point:

1. Input Screen Design.
2. Output Screen Design.
3. Format of reports and other outputs

#### Functional Testing

These are the points concerned during the stress test:

Nominal input: character is in putted in the place of digits and the system has to flash the message "Data error" Boundary value analysis: exhaustive test cases have designed to create an output report that produces the maximum (and minimum) allowable number of table entries

# CHAPTER 6

* 1. **CONCLUSION:-**

By this system weather forecasting report generation becomes easy. Less chances of malfunctioning are there. The system has reached a steady state but still improvements are to be made. The system is operated at a high level of efficiency and all the work and user associated with the system understand its advantage. It was intended to solve as requirement specification. In future this system can be implemented to all over the world and will be designed for cross platform.

# BIBLIOGRAPHY:-

<http://www.w3.org/2000/svg> <https://openweathermap.org/api> <https://www.youtube.com/>