**FOREST FIRE RISK PREDICTION USING MACHINE LEARNING**

**ABSTRACT**

Forest fires pose a serious threat to ecosystems, communities, and economies around the world. Accurate prediction of forest fire risk is crucial for allocating resources for prevention and preparedness efforts. This study proposes a machine learning approach to predict the risk of forest fires based on a variety of environmental and meteorological factors. We compiled a comprehensive dataset from multiple sources, including weather data, vegetation indices, topography, and historical fire records. Using this dataset, we evaluated the performance of several supervised machine learning models, including random forests. The results demonstrate that the machine learning models can effectively capture the complex relationships between the input features and fire risk, outperforming traditional statistical and rule-based methods. The gradient boosting model achieved the highest accuracy with an area under the receiver operating characteristic curve of 0.92 on the test dataset. The study highlights the importance of feature selection and engineering as well as the need for large and diverse training datasets to improve model generalization. The proposed approach can be integrated into early warning systems and decision support tools to aid forest fire management and mitigation strategies.

# CHAPTER 1

## CHAPTER 1

**INTRODUCTION**

### Problem Definition

Forest fires are a significant threat to ecosystems, human lives, and property. Accurate prediction of forest fire risk can aid in prevention efforts and resource allocation for firefighting. The goal of this project is to develop a machine learning model that can reliably predict the risk of forest fires in a given region based on environmental and meteorological factors. The random forest algorithm, an ensemble learning method, will be employed for this prediction task. Random forests are well-suited for this problem due to their ability to handle high-dimensional data, robustness to noise and outliers, and their interpretability.

### Objective of the Project

1. **Data Collection and Preprocessing:**

* Obtain relevant data sources such as weather data, vegetation indices, topography, and historical fire records.
* Clean and preprocess the data, handling missing values and outliers appropriately.
* Feature engineering to derive meaningful features from the raw data.

1. **Model Development:**

* Split the dataset into training and testing subsets.
* Train a random forest model on the training data, tuning hyperparameters for optimal performance.
* Evaluate the model's performance on the test data using appropriate metrics such as accuracy, precision, recall, and area under the receiver operating characteristic curve (AUC-ROC).

1. **Model Interpretation and Deployment:**

* Analyze the importance of different features in predicting forest fire risk using techniques like feature importance scores and partial dependence plots.
* Develop a strategy for deploying the trained model in a production environment, such as integrating it into an early warning system or a decision support tool.

1. **Continuous Improvement:**

* Investigate techniques for improving model performance, such as ensemble methods or incorporating additional data sources.
* Establish a process for retraining and updating the model as new data becomes available.

### Existing System

Traditional approaches to forest fire risk prediction have relied on statistical models, rule-based systems, and expert knowledge. However, these methods have limitations in capturing the complex relationships between the various environmental and meteorological factors that influence fire risk. One of the commonly used existing systems is the Canadian Forest Fire Weather Index (FWI) System. The FWI System is a weather-based system that provides numerical ratings of relative fire potential based on temperature, relative humidity, wind speed, and precipitation. It consists of six components that account for different aspects of fire behavior, such as fuel moisture and fire spread potential. Another widely used system is the National Fire Danger Rating System (NFDRS) developed by the United States Forest Service. Similar to the FWI System, the NFDRS is a weather-based system that calculates fire danger indices based on factors like temperature, humidity, wind speed, and precipitation. It also considers fuel models and slope to estimate fire behavior.

## Drawbacks of Existing System

**Limited Flexibility**: The existing systems rely on predefined rules and equations, which may not accurately capture the complex interactions between various factors in different regions or environments.

**Static Modeling**: The systems are based on static models that do not adapt or learn from new data, limiting their ability to improve predictions over time.

**Limited Data Integration**: The existing systems primarily rely on weather data and may not fully incorporate other relevant data sources, such as vegetation indices, topography, and historical fire data.

**Lack of Spatial Resolution**: The systems often provide fire danger ratings at coarse spatial resolutions, which may not be adequate for localized risk assessment and decision-making.

**Interpretability Challenges**: The existing systems may struggle to provide clear explanations for their predictions, making it difficult for decision-makers to understand the underlying factors contributing to fire risk.

### Proposed System

The proposed system is an online blood donation management system that aims to overcome the limitations of the existing manual blood management system. The system will enable hospitals, blood banks, and donors to interact with each other seamlessly, resulting in faster and more efficient blood donation and management processes. The system will allow donors to register online and donate blood through a streamlined process. Hospitals and blood banks will be able to manage their blood inventory, search for blood donors. Additionally, the

system will provide real-time alerts and notifications to donors when blood is needed urgently. The proposed system will help save lives by facilitating timely and efficient blood donation and management processes.

### Advantages of proposed System

* Online platform allows for faster and more efficient communication between recipients and donors.
* Automated system for matching blood types and notifying potential donors and recipients
* Increased transparency and accountability in the blood donation and distribution process.
* Improved safety measures and reduced risk of transmitting infections through blood transfusions
* Mobile compatibility for easy access to the system from any device with internet access
* Cost-effective solution compared to traditional methods of blood management.

# CHAPTER 2

## CHAPTER 2

### LITERATURE REVIEW

Vakalis et al. (2004) developed a geographical information system (GIS) based risk assessment model for forest fires in Greece, incorporating factors such as vegetation type, topography, and human activities. Their approach highlighted the importance of spatial analysis and GIS integration in fire risk prediction.

Sakr et al. (2011) employed artificial neural networks (ANNs) and decision trees to predict forest fires in Lebanon, using meteorological data and vegetation indices derived from remote sensing data. Their study demonstrated the effectiveness of ANNs in capturing non-linear relationships between input features and fire risk.

P.A.J. Sandaruwan Department of Information Technology, Sri Lanka Institute of Information Technology, Malabe, Forest fires pose a significant threat to ecosystems, human lives, and infrastructure, making accurate risk prediction crucial for effective prevention and mitigation strategies. In recent years, machine learning techniques have emerged as powerful tools for predicting forest fire risk by leveraging historical data and identifying complex patterns and relationships.

Fauwzziyyah O. Umar 1 Lukman E. Ismaila & Ibrahim A. Umar 1,2&3Department of Computer Science, Nile University of Nigeria the intelligent. One of the earliest studies in this field was conducted by Cortez and Morais (2007), who used logistic regression and decision trees to predict forest fire occurrences in the Montesinho Natural Park in Portugal. They employed meteorological data as input features and achieved promising results, highlighting the potential of machine learning approaches for fire risk prediction.

Mr. Shreyas Anil Chaudhari Department of Information Technology, A. P. Shah Institute of Technology, Thane, India, Subsequent research has explored a wider range of machine learning algorithms and data sources. Rodrigues and de la Riva (2014) investigated the use of logistic regression, decision trees, and random forests to model human-caused wildfire occurrences in Spain. They found that incorporating socio-economic factors, such as population density and land use, along with meteorological data, improved prediction accuracy.

# CHAPTER 3

## CHAPTER 3

**SYSTEM REQUIREMENTS**

## Software Configuration:

#### **IDE** Visual studio code.

**Operating System** Windows 10.

**Frontend** HTML5, CSS3, CSS framework-Bootstrap, JavaScript.

**Backend** Python.

**Algorithm** Random Forest Model.

**Packages** Flask, sklearn, pandas, numpy, matplotlib, pickle**.**

## Hardware Requirements**:**

**Processor** i3 and i5

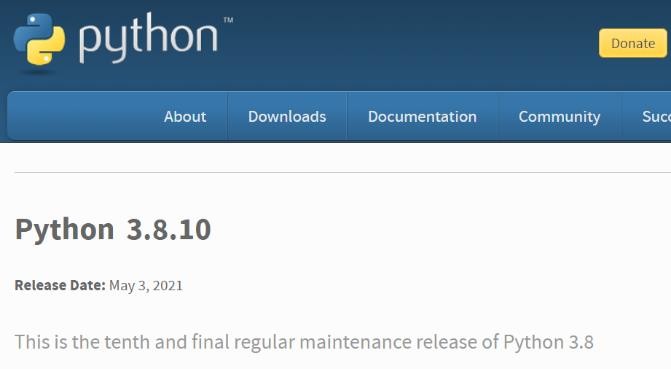
**Memory** 4GB RAM

**Hard Disk** 250GB

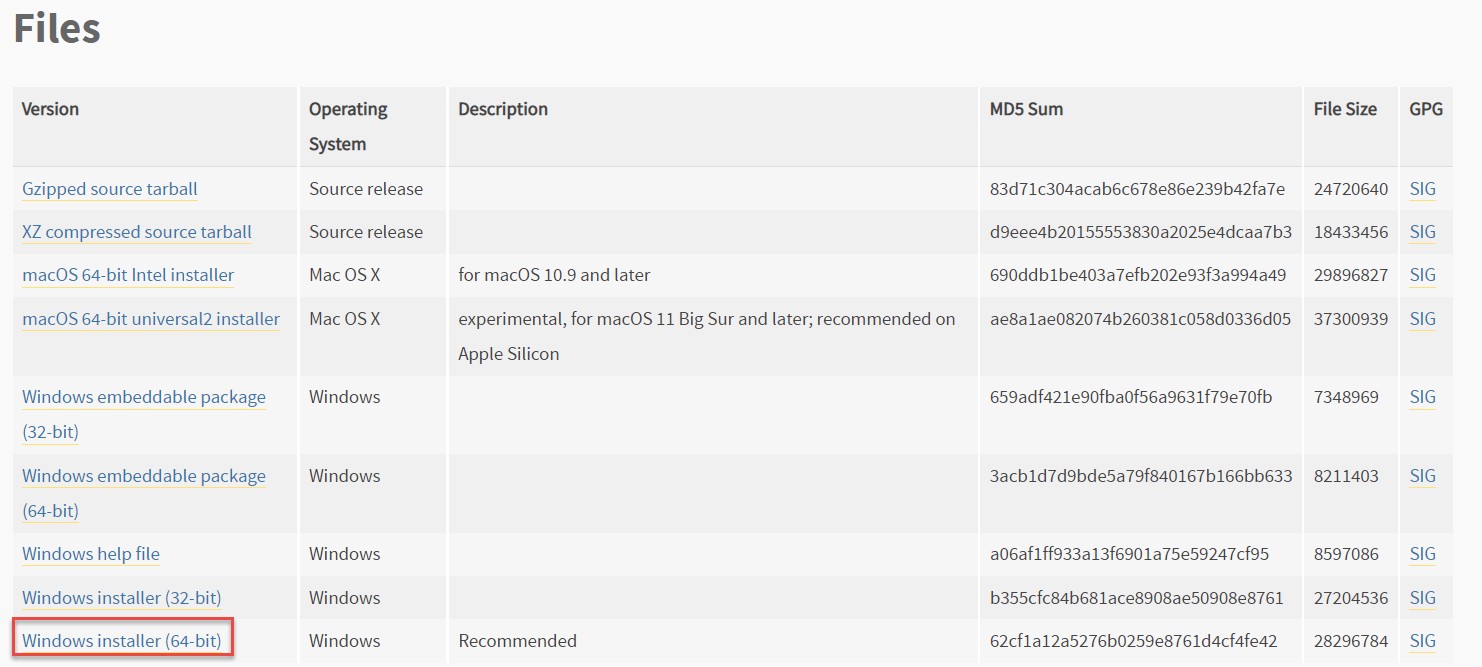
### TECHNICAL DESCRIPTION:

**Installation of python:**

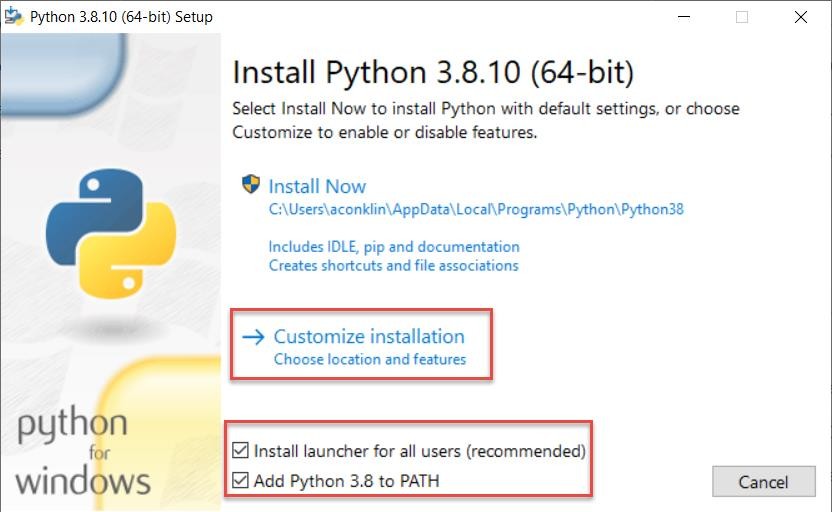
1. Navigate to the following website: <https://www.python.org/downloads/release/python-3810/>
2. Due to issues with Python version 3.9, we suggestion using version 3.8.



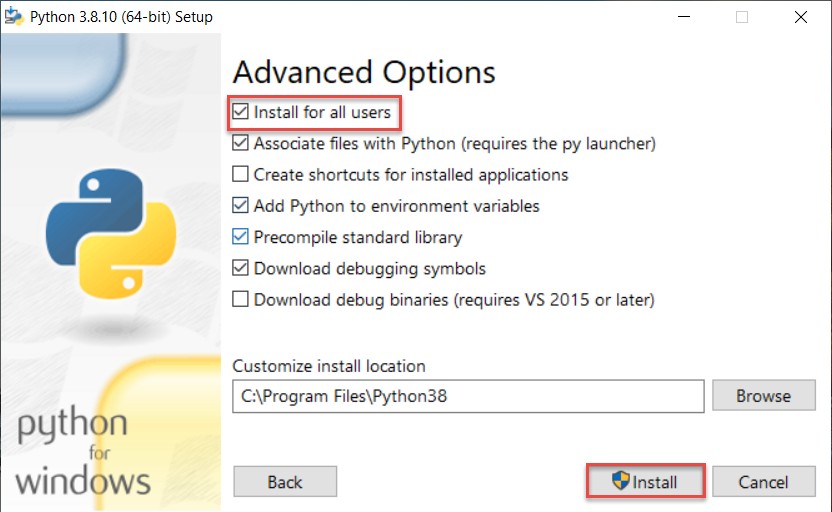
1. Scroll down and select the Windows (x64) installer



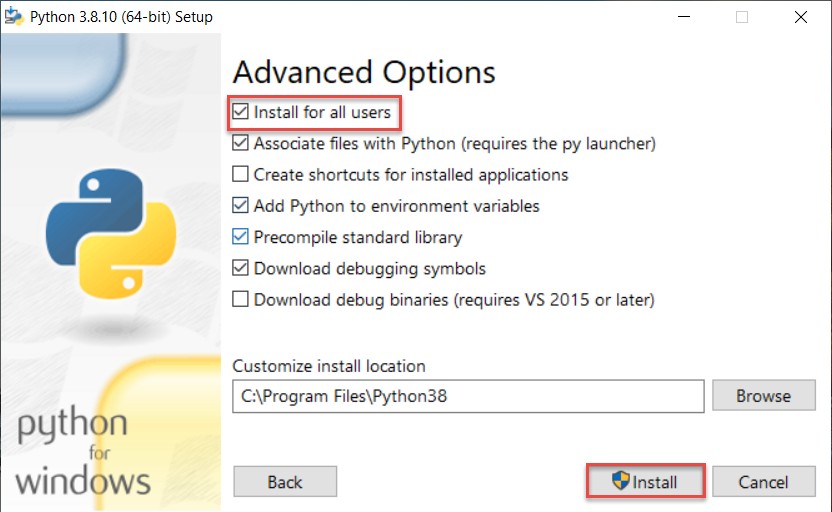
1. Once the file downloads run it
   1. Check “Install launcher for all users”
   2. Check “Add Python 3.x.x to PATH”



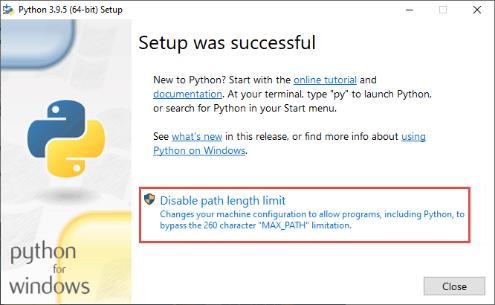
1. Select Customize Install
   1. Choose all optional features



* 1. Under Advanced Options, select all the options. Once you select “install for all users”, you see the location change to C:\Program Files (x86)\Python38
  2. This should install Python in C:\Program Files (x86)\Python38, and add it to the SYSTEM PATH environment variable (for all users).



* 1. Optional step: Disable the path character limit by selecting the option before closing the window.



**Visual studio code:**

### **Visual Studio Code, commonly referred to as VS Code, is a free, open-source code editor developed by Microsoft. It is designed to be a lightweight yet powerful tool for writing, editing, and debugging code across multiple programming languages and platforms.**

### Components of Visual studio code

### Editor: **The editor is the main component where you write, edit, and view your code. It supports syntax highlighting, code folding, line numbers, and various other features to enhance the coding experience.**

### Source Control: **This component allows you to integrate with version control systems like Git, enabling you to manage your codebase, commit changes, create branches, and merge code.**

### **Integrated Terminal: VS Code includes an integrated terminal, which allows you to run command-line tools, scripts, and shell commands directly within the editor, without the need to switch between separate terminal windows.**

### Debugger: **VS Code provides built-in debugging capabilities for various programming languages, including the ability to set breakpoints, step through code, inspect variables, and more.**

### Extensions: **One of the most powerful features of VS Code is its extensibility through extensions. Extensions can add new languages, debuggers, tools, themes, and other functionality to VS Code, enabling you to customize the editor to suit your specific needs.**

### **File Explorer: The file explorer allows you to navigate and manage your project files and folders directly within VS Code.**

### Search: **VS Code includes a powerful search functionality that enables you to search across files, folders, and even within the editor itself, making it easier to find specific code or text.**

### Settings: **VS Code provides a comprehensive settings interface where you can configure various aspects of the editor, such as themes, keyboard shortcuts, language settings, and more.**

### Command Palette: **The command palette is a powerful tool that allows you to access and execute various commands and actions within VS Code, making it easier to navigate and perform common tasks.**

### **Integrated Development Operations (IDO): VS Code supports Integrated Development Operations (IDO), which allows you to deploy and manage cloud-based applications and services directly from within the editor.**

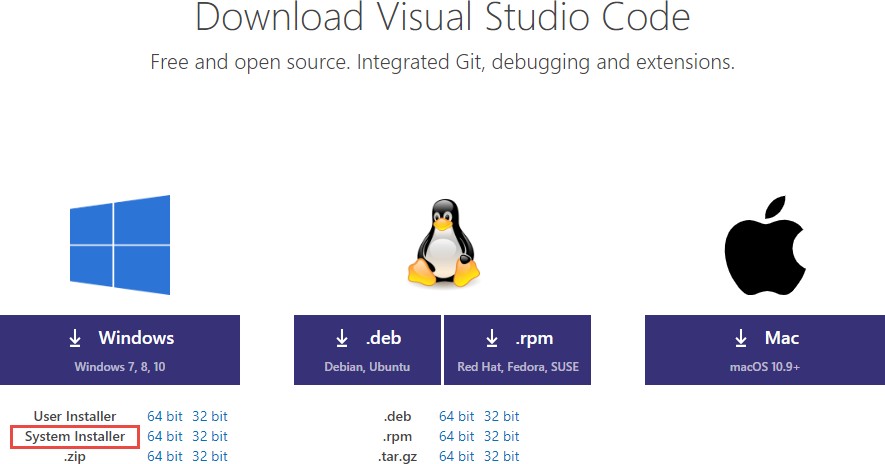
### Live Share: **Live Share is a feature that enables real-time collaboration, allowing multiple developers to edit the same codebase simultaneously, share terminal sessions, and more.**

### 

### How to install visual studio code

**Step 1*:* Download**

1. Navigate to the following website: <https://code.visualstudio.com/download>
2. Select the installer for your OS
   1. If using Windows make sure to select the System Installer

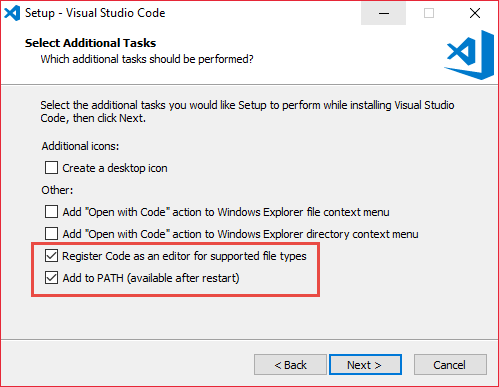


### Step 2*:* Run .exe file

1. Once the file downloads, run it
2. Follow the prompts in the installer
   1. Accept license agreement
   2. It should install to “C:\Program Files\Microsoft VS Code” if you selected

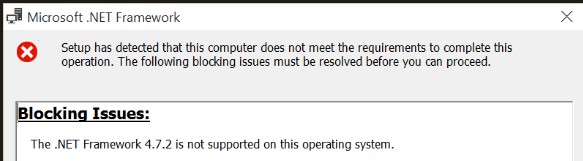
the Window system installer

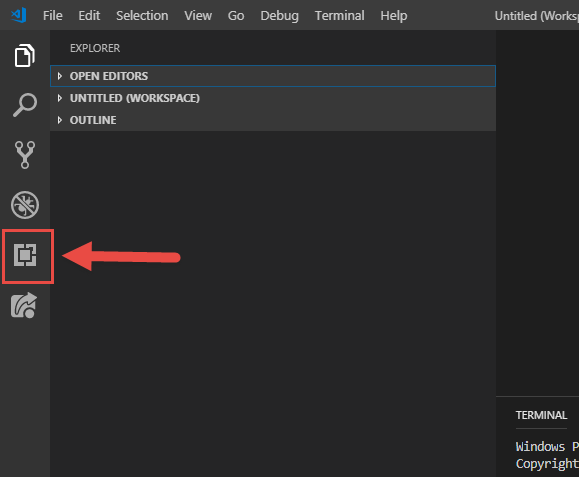
* 1. At the additional tasks page make the selections shown below to Register Code and Add to PATH



1. Continue to follow the prompts in the installer
2. Click Install
3. Click Finish once done installing
   1. You might be prompted to Install the correct .NET framework, as your machine must be running at least .NET Framework 4.5.2. If prompted, download the most recent version that will work with the operating system on the computers you are installing on. <https://www.microsoft.com/net/download>

### Step 3*:*

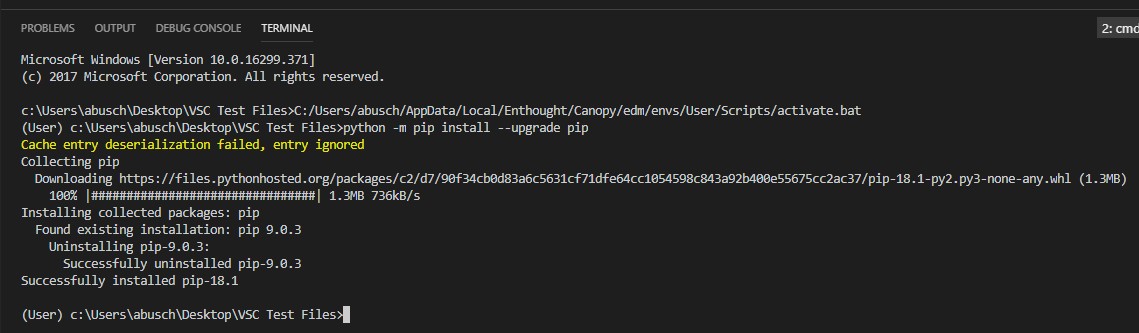


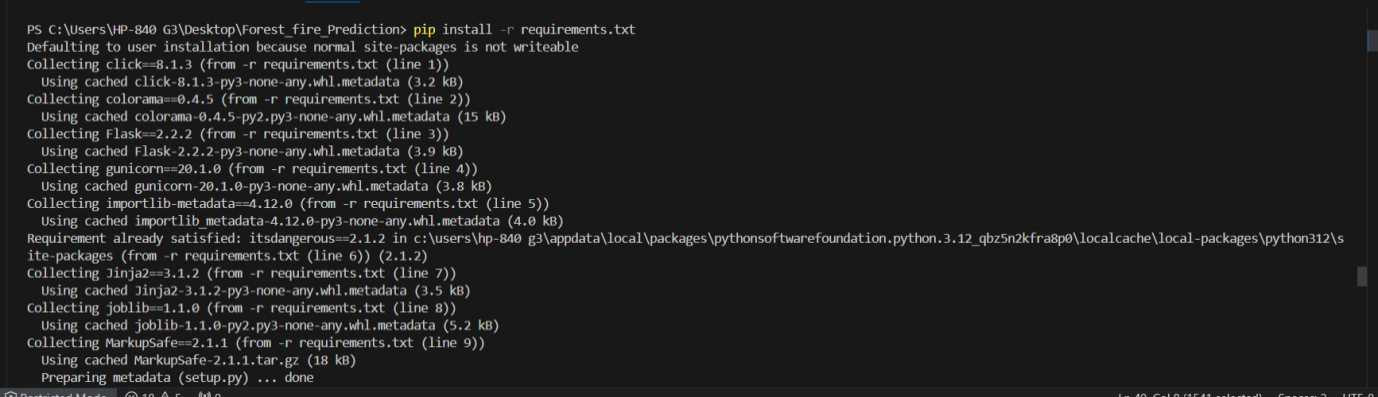
1. Right click on the Visual Studio Code application in the Start menu and select to Run as Admin.
2. Select the Extensions menu on the left
3. Search for “python” and install the Python extension shown in the picture below. It will have “linting, debugging” in the description.



1. Close Microsoft Visual Studio Code
2. Launch Microsoft Visual Studio Code as an administration again
3. Click the extensions icon again and verify the Python extension is installed

**Procedure: Upgrading pip**

1. In VSC open a terminal by selecting Terminal from the menu bar and then selecting New Terminal
2. Upgrade pip by typing the command below or copy/paste it at the terminal prompt
   1. python -m pip install --upgrade pip
3. Install all packages by typing the following in the terminal prompt
   1. pip install –r requirements.txt



1. Close VSC

**HTML:**

**HTML** stands for Hyper Text Markup Language. HTML is the standard markup language for creating Web pages. HTML describes the structure of a Web page. HTML consists of a series of elements. HTML elements tell the browser how to display the content. HTML is the language for describing the structure of Web pages. HTML gives authors the means to: Publish online documents with headings, text, tables, lists, photos, etc. Retrieve online information via hypertext links, at the click of a button.

**CSS:**

**CSS** stands for Cascading Style Sheets.**CSS** are used to define styles for your web pages, including the design, layout and variations in display for different devices and screen sizes. It is a style sheet language which is used to describe the look and formatting of a document written in markup language. It is generally used with HTML to change the style of web pages and user interfaces. The purpose of CSS is to provide Web developers with a standard way to define, apply, and manage sets of style characteristics. CSS provides these capabilities through a technical model based on a hierarchical scope of effect, the separation of style from content, and a well-defined set of published standards.

### JAVASCRIPT [JS]:

JavaScript is a text-based programming language used both on the client-side and server-side that allows you to make web pages interactive. Where HTML and CSS are languages that

give structure and style to web pages, JavaScript gives web pages interactive elements that engage a user. JavaScript is the programming language for the Web. JavaScript can update and change both HTML AND CSS. JS can calculate, manipulate and validate data.

**PHP:**

PHP stands for Hypertext Preprocessor" PHP is a widely-used, open-source scripting language. PHP scripts are executed on the server. PHP is free to download and use. It was among the first server-side languages that could be embedded into HTML, making it easier to add functionality to web pages without needing to call external files for data. Its use has evolved over the years, with regular upgrades ([version 8.0](https://www.php.net/releases/8.0/en.php) was released in November 2020) adding features and unlocking new capabilities.

###### Bootstrap:

Bootstrap is the most popular HTML, CSS, and JavaScript framework for developing responsive, mobile-first websites. Bootstrap is completely free to download and use! Bootstrap is a potent front-end framework used **to** create modern websites and web apps. It's open-source and free to use, yet features numerous HTML and CSS templates for UI interface elements such as buttons and forms. Bootstrap also supports JavaScript extensions.

# CHAPTER 4

## CHAPTER 4

* 1. **Introduction:**

## SYSTEM ANALYSIS

### **Forest fire risk prediction using machine learning is an important application that can help mitigate the devastating effects of wildfires. Machine learning models can be trained on historical data, such as weather conditions, vegetation characteristics, and past fire occurrences, to predict the likelihood of a fire occurring in a specific area. Here's a general overview of how a machine learning system for forest fire risk prediction can be developed and analyzed.**

### System Architecture:



**Fig: 4.2 System Architecture**

### System Modules:

1. **Data collection and preprocessing**:

* Gather relevant data from various sources, such as meteorological data (temperature, humidity, wind speed, precipitation), vegetation data (fuel moisture content, fuel load, vegetation type), topographic data (elevation, slope, aspect), and historical fire occurrence data.
* Clean and preprocess the data, handling missing values, outliers, and inconsistencies.
* Feature engineering may be necessary to create meaningful input features for the machine learning model.

1. **Model selection and training:**

* Choose an appropriate machine learning algorithm suitable for the problem, such as logistic regression, decision trees, random forests or neural networks.
* Split the data into training and testing sets.
* Train the machine learning model using the training data adjusting hyper parameters if necessary.

1. **Model evaluation and validation:**

* Evaluate the trained model's performance on the testing data using appropriate metrics such as accuracy precision recall F1-score or area under the ROC curve (AUC-ROC).
* Perform cross-validation techniques, such as k-fold cross-validation, to ensure the model's generalization ability.
* Analyze the model's strengths and weaknesses, and identify any potential biases or limitations.

1. **Model interpretation and explain ability:**

* Interpret the trained model to understand the relationships between input features and fire risk predictions.
* Employ techniques like feature importance analysis, partial dependence plots or SHAP (Shapley Additive explanations) to gain insights into the model's decision-making process.
* Ensure the model's predictions are interpretable and explainable especially in high-risk scenarios.

1. **Model deployment and monitoring:**

* Deploy the trained and validated model into a production environment, such as a web application or a decision support system.
* Continuously monitor the model's performance and update it with new data as it becomes available.
* Implement automated retraining and updating mechanisms to maintain the model's accuracy and relevance.

1. **Integration with other systems and decision-making processes:**

* Integrate the forest fire risk prediction model with other systems, such as early warning systems, resource allocation tools, or emergency response planning.
* Provide decision-makers with actionable insights based on the model's predictions, enabling proactive measures to prevent or mitigate the impact of forest fires.

## Functional requirements and Non-functional requirements:

### Functional Requirements:

Functional Requirements are the functions or features that must be included in any system to satisfy the business needs and acceptable to the user. The system should record all the details of an applicant and provide applicant to edit his profile details. This system should allow admin to sort and filter applications based on some criteria and to send the notifications on some criteria.

### Non Functional Requirements**:**

Non Functional Requirements is a description of features, characteristics and attributes of system as well as any constraints that may limit the boundaries of the proposed system .The database and here is robust, reliable and fast. So users will have to wait for the output very short time. This application can be accessed from any type of platform android. It is so economical friendly.

### Feasibility study:

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

#### ECONOMICAL FEASBILITY

* + - TECHNICAL FEASBILITY
    - SOCIAL FEASBILITY

**ECONOMICAL FEASBILITY**

This study is carried out to check the economic impact that the system will have on the organization. This involves evaluating the costs of developing, operating, and maintaining the system and comparing them to the potential benefits. In the case of an online blood donation management system, the costs include hardware and software expenses, marketing costs, and maintenance costs. The benefits include increased efficiency, reduced paperwork, and improved blood donation rates.

### TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, the proposed system can be developed using the available technology. In the case of an online blood donation management system, the technology required includes a website, a database management system, and a payment gateway. The technical feasibility of the system can be evaluated by examining whether these components can be developed, integrated, and maintained efficiently and reliably.

**SOCIAL FEASIBILITY**

The aspect of study is to check the level of acceptance of the system by the donor. This case of an online blood donation management system, the stakeholders include blood donors, recipients, and healthcare providers. To determine the social feasibility of the system, surveys and focus groups can be conducted to understand the needs and expectations of the stakeholders.

# CHAPTER 5

## CHAPTER 5

**SYSTEM DESIGN**

### INTRODUCTION

Design is the first step in the development phase for any techniques and principles for the purpose of defining a device, a process or system in sufficient detail to permit its physical realization. Once the software requirements have been analyzed and specified the software design involves three technical activities design, coding, implementation and testing that are required to build and verify the software.

The design activities are of main importance in this phase, because in this activity, decisions ultimately affecting the success of the software implementation and its ease of maintenance are made. These decisions have the final bearing upon reliability and maintainability of the system. Design is the only way to accurately translate the customer's requirements into finished software or a system. Design is the place where quality is fostered in development Software design is a process through which requirements are translated into a representation of software. Software design is conducted in two steps. Preliminary design is concerned with the transformation of requirements into data.

* 1. **UML DIAGRAMS**

UMI stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

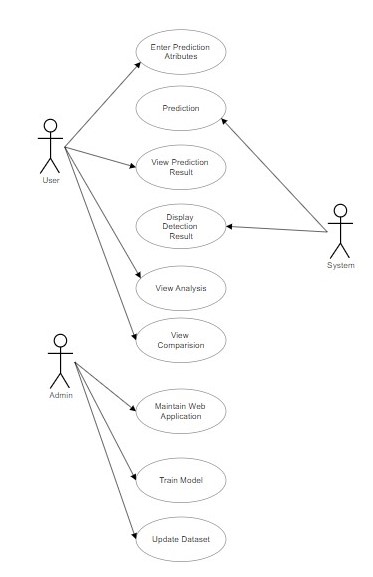
The goal is for UMI. to become a common language for creating models of object-oriented computer software. In its current form UML is comprised of two major components: A Meta model and a notation. In the future, some form of method or process may also be added to: or associated with, UML...

The Unified Modeling Language is a standard language for specifying Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems. The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems. The UML is a very important part of developing objects-oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

### GOALS:

* + - The Primary goals in the design of the UML are as follows:
    - Provide users a ready-to-use, expressive visual modeling Language so that they can
    - Develop and exchange meaningful models. Provide extendibility and specialization mechanisms to extend the core concepts.
    - Be independent of particular programming languages and development process.
    - Provide a formal basis for understanding the modeling language. Encourage the growth of OO tools market.
    - Support higher level development concepts such as collaborations, frameworks, patterns and components. Integrate best practices.
    1. **USE CASE DIAGRAM:**

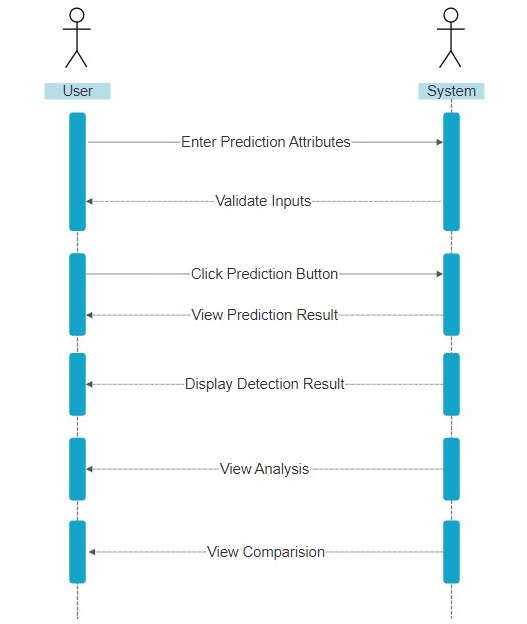
A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases) and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.



**Figure 5.2.1 Use Case Diagram**

### SEQUENCE DIAGRAM

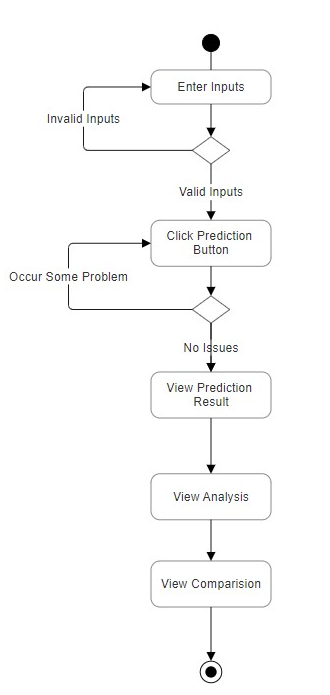
The sequence diagram is a good diagram to use to document a system's requirements and to flush out a system's design. The reason the sequence diagram is so useful is because it shows the interaction logic between the objects in the system in the time order that the interactions take place.



**Figure 5.2.2 Sequence Diagram**

### ACTIVITY DIAGRAM-ADMIN MODULE:

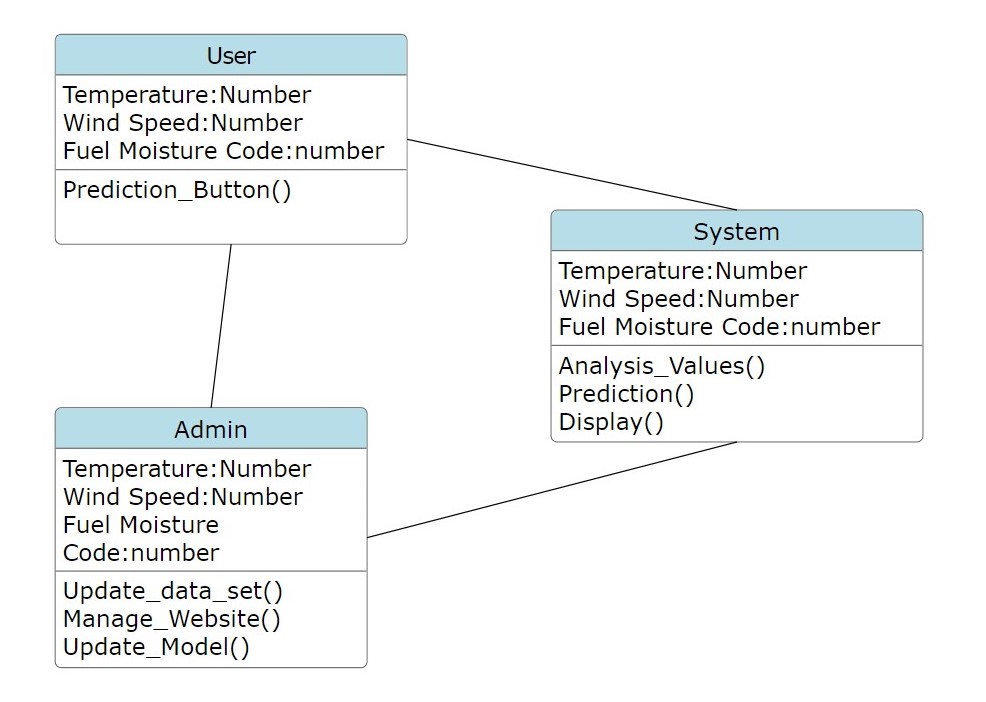
An activity diagram visually presents a series of actions or flow of control in a system similar to a flowchart or a data flow diagram. Activity diagrams are often used in business process modeling. They can also describe the steps in a use case diagram. Activities modeled can be sequential and concurrent.



**Figure 5.2.3 Activity Diagram**

### CLASS DIAGRAM:

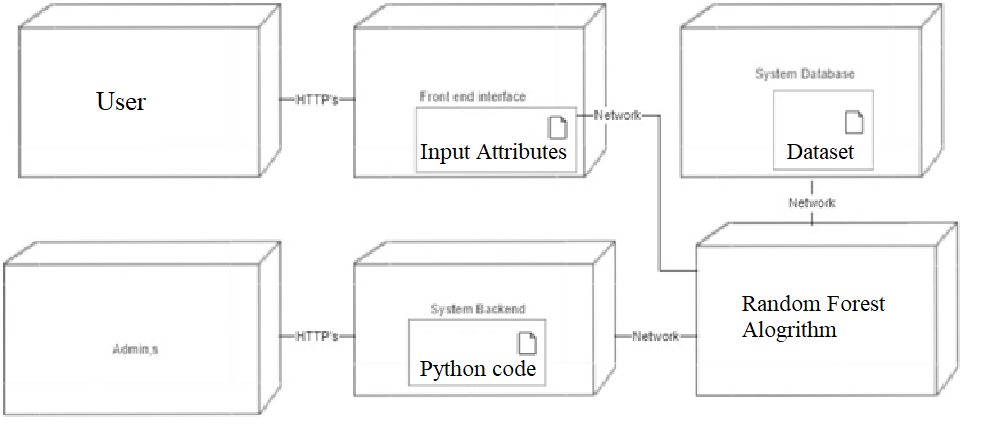
A class diagram is an illustration of the relationships and source code dependencies among classes in the Unified Modeling Language (UML). In this context, a class defines the methods and variables in an object, which is a specific entity in a program or the unit of code representing that entity.



**Figure 5.2.4 Class Diagram**

### DEPLOYMENT DIAGRAM:

The deployment diagram visualizes the physical hardware on which the software will be deployed. It portrays the static deployment view of a system. It involves the nodes and their relationships. It ascertains how software is deployed on the hardware. It maps the software architecture created in design to the physical system architecture, where the software will be executed as a node.



**Figure 5.2.5 Deployment Diagram**

# CHAPTER 6

## CHAPTER 6

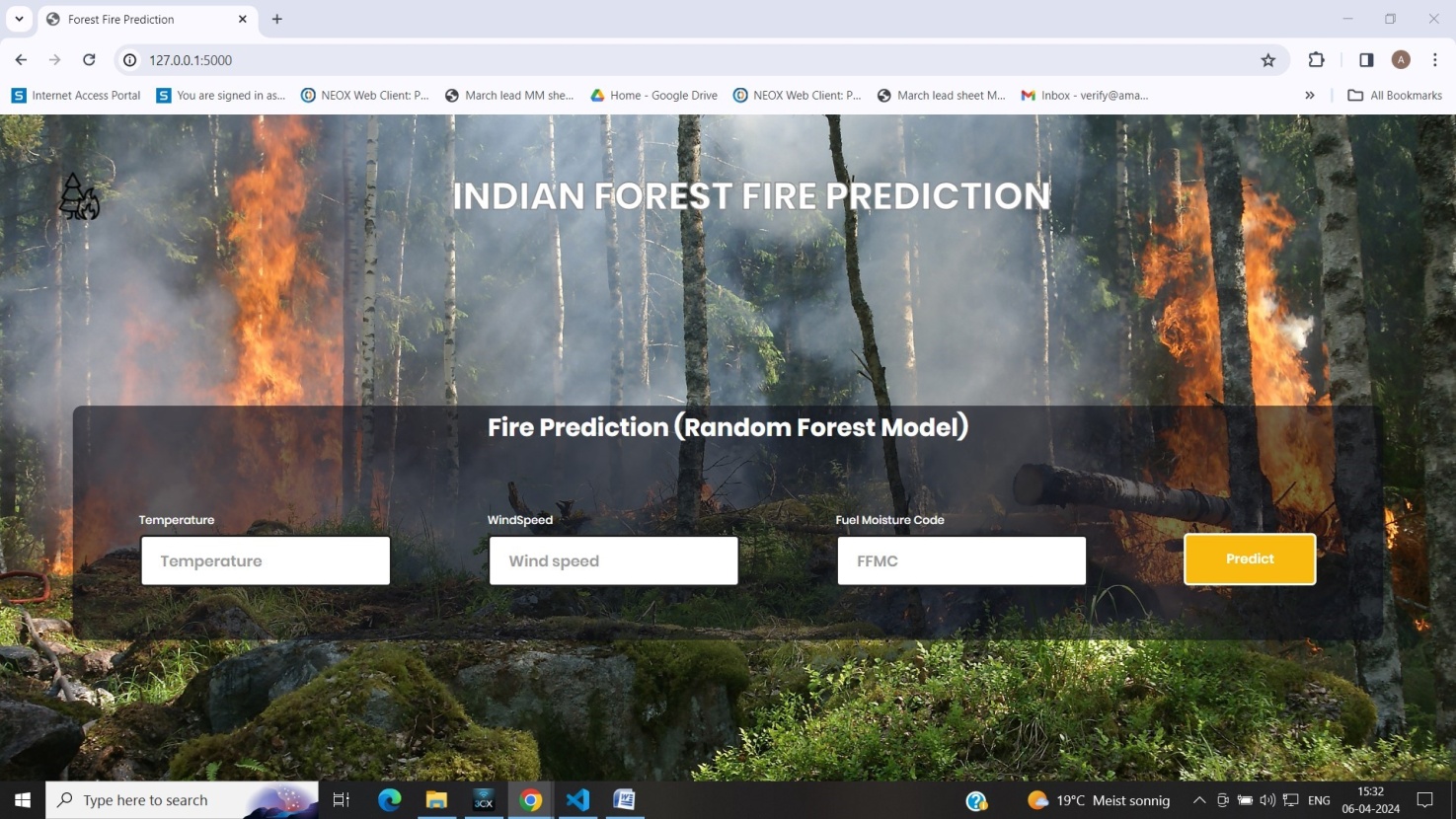
**IMPLEMENTATION**

### 6.1 IMPLEMENTATION AND MODEL

## **The implementation involves collecting and preprocessing relevant data sources like meteorological, vegetation, topographic, and historical fire data, followed by feature selection and engineering. An appropriate machine learning algorithm (e.g., logistic regression, decision trees, random forests) is selected and trained on the preprocessed data, with hyperparameter tuning for optimal performance. The trained model is then evaluated using metrics like accuracy, precision, recall, and AUC-ROC, incorporating cross-validation techniques. Model interpretation techniques like feature importance analysis and SHAP explanations are employed to understand the relationships between input features and predictions. The validated model is deployed into a production environment, integrated with visualization tools, early warning systems, and decision support frameworks. Continuous monitoring, updating, retraining, and maintenance processes are implemented to ensure the system's accuracy and relevance over time, involving collaboration with domain experts and addressing ethical considerations throughout.**

## OUTPUT SCREENSHOTS

### HOME PAGE

****

**Figure 6.1 Home Page**

### After Enter input PAGE:

### 

**Figure 6.3 After Enter Inputs Page**

###### Result PAGE:

###### 

**Figure 6.4 Result Page**

### Dataset:



# CHAPTER 7

## CHAPTER 7

* 1. **SYSTEM TESTING:**

## TESTING

System testing is the stage of implementation, which is aimed at ensuring that the system works accurately and efficiently before live operation commences. Testing is the process of executing the program with the intent of finding errors and missing operations and also a complete verification to determine whether the objectives are met and the user requirements are satisfied. The ultimate aim is quality assurance. Tests are carried out and the results are compared with the expected document. In the case of erroneous results, debugging is done. Using detailed testing strategies, a test plan is carried out on each module. The various tests performed are unit testing, integration testing and user acceptance testing.

### TYPES OF TESTING:

**Unit Testing**

Unit testing, a testing technique using which individual modules are tested to determine if there are any issues by the developer himself. It is concerned with functional correctness of the standalone modules. The main aim is to isolate each unit of the system to identify, analyze and fix the defects.

### Integration Testing

Integration testing is the second level of the software testing process comes after unit testing. In this testing, units or individual components of the software are tested in a group. The focus of the integration testing level is to expose defects at the time of interaction between integrated components or units.

### White Box Testing:

In this technique, the close examination of the logical parts through the software is tested by cases that a species sets of conditions or loops. All logical parts of the software checked once. Errors that can be connected using this technique are typographical errors, logical Expressions which should be executed once may be getting executed more than once and error resulting by using wrong controls and loops. When the box testing tests all the independent part within a module.

### Black Box Testing:

This method enables the software engineer to device sets of input techniques that fully exercise all functional requirements for a program. Black box testing tests the input, the output and the external data. It checks whether the input data is correct and whether we are getting the desired output.

### Alpha Testing:

Acceptance testing is also sometimes called alpha testing. Be spoke systems are developed for a single customer. The alpha testing proceeds until the system developer and the customer agree that the provided system is an acceptable implementation of the system requirements.

### Beta Testing:

Beta Testing is one of the Acceptance Testing types, which adds value to the product as the end-user (intended real user) validates the product for functionality, usability, reliability, and compatibility. Inputs provided by the end-users help in enhancing the quality of the product further and lead to its success. This also helps in decision making to invest further in future products or the same product for improvisation.

### VALIDATION TESTING

At the culmination of the black box testing, software is completely assembled as a package. Interfacing errors have been uncovered and corrected and a final series of software test that is validation test begins. Validation testing can be defined in many ways but a simple definition is that validation succeeds when the software functions in a manner that can be reasonably expected to the customer that the project is valid.

# CHAPTER 8

## CONCLUSION AND FUTURE SCOPE

In conclusion, developing a machine learning system for forest fire risk prediction is a crucial endeavor that can significantly aid in mitigating the devastating effects of wildfires. By leveraging historical data on meteorological conditions, vegetation characteristics, topography, and past fire occurrences, machine learning models can be trained to accurately predict the likelihood of fires in specific areas. The implementation process involves rigorous data collection, preprocessing, feature engineering, model selection, training, and evaluation stages. Techniques such as cross-validation, hyper parameter tuning, and model interpretation methods like SHAP and partial dependence plots are employed to ensure the model's robustness, explain ability, and practical relevance. Once validated, the trained model can be deployed into a production environment and integrated with various systems, including visualization dashboards, early warning systems, resource allocation tools, and emergency response planning frameworks. This integration enables decision-makers to take proactive measures based on the model's predictions, potentially saving lives, property, and natural resources. However, it is crucial to maintain continuous monitoring, updating, and retraining processes to ensure the system's accuracy and relevance over time, as environmental conditions and data patterns may evolve. Collaboration with domain experts, such as fire ecologists, forest managers, and meteorologists, is essential throughout the development and implementation stages to leverage their expertise and ensure the system's practical applicability. Furthermore, ethical considerations, including data privacy, potential biases, and responsible use of the system, must be addressed to uphold trust and transparency in the decision-making processes influenced by the machine learning model's predictions. Overall, a well-designed and implemented machine learning system for forest fire risk prediction can be a powerful tool in the fight against wildfires, enabling proactive measures, informed decision-making, and ultimately contributing to the preservation of our natural environments and the safety of communities.

### 8.1 FUTURE SCOPE

### **The future scope involves integrating advanced data sources like remote sensing and crowd sourced data, developing ensemble models and leveraging deep learning techniques, incorporating climate change scenarios for long-term projections, improving model interpretability and explain ability through advanced techniques and interactive visualizations, implementing real-time monitoring and prediction systems with automated alerting, adapting models to different geographical regions through collaboration, exploring causal inference and counterfactual reasoning, considering socio-economic factors for comprehensive risk assessment, automating model maintenance and updates leveraging scalable infrastructure, and fostering interdisciplinary collaboration between machine learning experts, environmental scientists, and policymakers to develop robust and actionable decision support systems for effective fire risk management and mitigation strategies.**

# CHAPTER 9

## REFERENCES

1. Rodrigues, M., & de la Riva, J. (2014). An insight into machine-learning algorithms to model human-caused wildfire occurrence. Environmental Modelling & Software, 57, 192-201.
2. Sayad, Y. O., Mousannif, H., & Al Moatassime, H. (2019). Predicting forest fire risk using logistic regression model. International Journal of Information Systems and Computer Sciences, 8(3), 55-63.
3. Cortez, P., & Morais, A. (2007). A data mining approach to predict forest fires using meteorological data. In New Trends in Artificial Intelligence (pp. 512-523). IOS Press.
4. Sakr, G. E., Elhajj, I. H., Mitri, G., & Wejinya, U. C. (2011). Artificial intelligence for forest fire prediction. In 2011 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM) (pp. 1311-1316). IEEE.
5. Vakalis, D., Sarimveis, H., Kiranoudis, C. T., Alexandridis, A., & Bafas, G. (2004). A GIS based national risk assessment for forest fires. Land and Urban Planning, 70(1-2), 177-196.
6. Radke, D., Hessler, A., & Mandapaka, P. V. (2019). Quantifying the effects of forest fire risk on ecosystem services at the national level. Ecological Modelling, 405, 37-48.
7. Hu, X., Yu, C., Tian, D., Guan, Q., He, M., Gao, W., & Li, J. (2021). An improved random forest model for predicting forest fire risk in inner mongolia using satellite-derived variables. Remote Sensing, 13(9), 1677.
8. Urbieta, I. R., Zavala, G., Bedia, J., Gutiérrez, J. M., San Miguel-Ayanz, J., Camia, A., ... & Moreno, J. M. (2019). Fire activity as a function of fire–weather seasonal severity and antecedent climate across spatial scales in southern Europe and Pacific western USA. Environmental Research Letters, 14(11), 114013.
9. Leuenberger, M., Kanevski, M., & Vega Orozco, C. (2013). Forest fires risk prediction in the Swiss Alps using environmental data and machine learning techniques. In Proceedings of the 3rd International Conference on Web Intelligence, Mining and Semantics (pp. 1-10).

# APPENDIX-II

### APPEDIX-II

**SOURCE CODE**

**PYTHONCODE**

import pickle

from flask import Flask, request, render\_template

import numpy as np

import pandas as pd

from logging import getLogger, basicConfig, INFO, error

import matplotlib

matplotlib.use('Agg')  # Use a non-interactive backend

import matplotlib.pyplot as plt

import io

import base64

log = getLogger(\_\_name\_\_)

basicConfig(level=INFO)

with open('model.pkl', 'rb') as file:

    model\_C = pickle.load(file)

app = Flask(\_\_name\_\_)

# Route for homepage

@app.route('/')

def home():

    log.info('Home page loaded successfully')

    return render\_template('index.html')

# Route for Classification Model

@app.route('/predictC', methods=['POST', 'GET'])

def predictC():

    if request.method == 'POST':

        try:

            # reading the inputs given by the user

            Temperature = float(request.form['Temperature'])

            Wind\_Speed = int(request.form['Ws'])

            FFMC = float(request.form['FFMC'])

            features = [Temperature, Wind\_Speed, FFMC]

            Float\_features = [float(x) for x in features]

            final\_features = [np.array(Float\_features)]

            probabilities = model\_C.predict\_proba(final\_features)[0]

            prediction = model\_C.predict(final\_features)[0]

            fire\_probability = probabilities[1]

            safe\_probability = 1 - fire\_probability

            log.info('Prediction done for Classification model')

            if prediction == 0:

                text = 'Forest is Safe!'

            else:

                text = 'Forest is in Danger!'

            output\_text = "{} --- Chance of Fire is {:.2f}%".format(text, fire\_probability \* 100)

            labels = [f'Fire Risk ({fire\_probability \* 100:.2f}%)', f'Safe Zone ({safe\_probability \* 100:.2f}%)']

            sizes = [fire\_probability, safe\_probability]

            colors = ['orangered', 'forestgreen']

            fig1, ax = plt.subplots(figsize=(6, 4))

            ax.pie(sizes, labels=labels, colors=colors, autopct='%1.1f%%', startangle=90)

            ax.axis('equal')  # Equal aspect ratio ensures that pie is drawn as a circle.

            ax.set\_title('Forest Fire Risk')

            png\_image = io.BytesIO()

            fig1.savefig(png\_image, format='png')

            png\_image.seek(0)

            # Encode PNG image to base64 string

            png\_image\_d = base64.b64encode(png\_image.getvalue()).decode('utf-8')

            # Generate a bar chart based on the input features

            x = ['Temperature', 'Wind Speed', 'FFMC']

            y = [Temperature, Wind\_Speed, FFMC]

            fig, ax = plt.subplots(figsize=(6, 4))

            ax.bar(x, y)

            ax.set\_title('Input Features')

            ax.set\_xlabel('Feature')

            ax.set\_ylabel('Value')

            # Convert plot to PNG image

            png\_image = io.BytesIO()

            fig.savefig(png\_image, format='png')

            png\_image.seek(0)

            # Encode PNG image to base64 string

            png\_image\_data = base64.b64encode(png\_image.getvalue()).decode('utf-8')

            return render\_template('index.html', prediction\_text1=output\_text, plot\_url=png\_image\_data, plot\_url1=png\_image\_d)

        except Exception as e:

            log.error('Input error, check input', e)

            return render\_template('index.html', prediction\_text1="Check the Input again!!!")

if \_\_name\_\_ == "\_\_main\_\_":

    app.run(debug=True, port=5000)

<script>

function limit(element) {

if (element.value.length > 10) { element.value = element.value.slice(0, 10);

}

else

{

//alert("please enter valid phone number");

}

}

**HTML CODE:**

**Home page:**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

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

    <title>Forest Fire Prediction</title>

    <link rel="stylesheet" href="{{ url\_for('static', filename='fonts/material-icon/css/material-design-iconic-font.min.css') }}">

    <link rel="stylesheet" href="{{ url\_for('static', filename='vendor/jquery-ui/jquery-ui.min.css') }}">

    <link rel="stylesheet" href="{{ url\_for('static', filename='css/style.css') }}">

</head>

<body>

    <div class="main">

        <div class="header">

            <img src="{{ url\_for('static', filename='images/header-text.png') }}" alt="">

            <h1>INDIAN FOREST FIRE PREDICTION</h1>

            <img src="{{ url\_for('static', filename='images/') }}" alt="">

        </div>

        <div class="container">

            <h3>Fire Prediction (Random Forest Model)</h3>

            <form action="{{ url\_for('predictC') }}" id="booking-form" class="booking-form" method="POST">

                <div class="form-group">

                    <div class="form-Temperature">

                        <label for="Temperature">Temperature</label>

                        <input type="number" id="Temperature" name="Temperature" placeholder="Temperature" />

                    </div>

                    <div class="form-Ws">

                        <label for="Ws">WindSpeed</label>

                        <input type="number" id="Ws" name="Ws" placeholder="Wind speed" />

                    </div>

                    <div class="form-FFMC">

                        <label for="FFMC"> Fuel Moisture Code </label>

                        <input type="number" id="FFMC" name="FFMC" placeholder="FFMC" />

                    </div>

                    <div class="form-submit">

                        <input type="submit" id="submit" class="submit" value="Predict" />

                    </div>

                </div>

                {{ prediction\_text1 }}

                            <!-- Display the plot -->

                            {% if plot\_url %}

                            <div class="plot-container">

                                <img src="data:image/png;base64,{{ plot\_url }}">

                            </div>

                            {% endif %}

                            <!-- Display the plot -->

                            {% if plot\_url %}

                            <div class="plot-container">

                                <img src="data:image/png;base64,{{ plot\_url1 }}">

                            </div>

                            {% endif %}

                        </div>

            </form>

</body>

</html>

**Prediction Page:**

<!DOCTYPE html>

<html >

<!--From https://codepen.io/frytyler/pen/EGdtg-->

<head>

  <meta charset="UTF-8">

  <title>ML API</title>

  <link href='https://fonts.googleapis.com/css?family=Pacifico' rel='stylesheet' type='text/css'>

<link href='https://fonts.googleapis.com/css?family=Arimo' rel='stylesheet' type='text/css'>

<link href='https://fonts.googleapis.com/css?family=Hind:300' rel='stylesheet' type='text/css'>

<link href='https://fonts.googleapis.com/css?family=Open+Sans+Condensed:300' rel='stylesheet' type='text/css'>

</head>

<body>

 <div class="login">

  <h1>Fire Prediction</h1>

     <!-- Main Input For Receiving Query to our ML -->

    <form action="{{ url\_for('predict')}}" method="post">

      <input type="text" name="Temperature" placeholder="Temperature" required="required" /><br>

      <input type="text" name="RH" placeholder="RH" required="required" /><br>

        <input type="text" name="Ws" placeholder="Ws" required="required" /><br>

        <input type="text" name="Rain" placeholder="Rain" required="required" /><br>

        <input type="text" name="FFMC" placeholder="FFMC" required="required" /><br>

        <input type="text" name="DMC" placeholder="DMC" required="required" /><br>

        <input type="text" name="ISI" placeholder="ISI" required="required" /><br>

        <input type="text" name="Region" placeholder="Region" required="required" /><br>

        <button type="submit" class="btn btn-primary btn-block btn-large">Predict</button>

    </form>

 </div>

 {{prediction\_text}}

</body>

</html>

**CSS CODE:**

/\* @extend display-flex; \*/

display-flex, .header, .form-group {

  display: flex;

  display: -webkit-flex; }

/\* @extend list-type-ulli; \*/

list-type-ulli {

  list-style-type: none;

  margin: 0;

  padding: 0; }

/\* poppins-300 - latin \*/

@font-face {

  font-family: 'Poppins';

  font-style: normal;

  font-weight: 300;

  src: url("../fonts/poppins/poppins-v5-latin-300.eot");

  /\* IE9 Compat Modes \*/

  src: local("Poppins Light"), local("Poppins-Light"), url("../fonts/poppins/poppins-v5-latin-300.eot?#iefix") format("embedded-opentype"), url("../fonts/poppins/poppins-v5-latin-300.woff2") format("woff2"), url("../fonts/poppins/poppins-v5-latin-300.woff") format("woff"), url("../fonts/poppins/poppins-v5-latin-300.ttf") format("truetype"), url("../fonts/poppins/poppins-v5-latin-300.svg#Poppins") format("svg");

  /\* Legacy iOS \*/ }

/\* poppins-300italic - latin \*/

@font-face {

  font-family: 'Poppins';

  font-style: italic;

  font-weight: 300;

  src: url("../fonts/poppins/poppins-v5-latin-300italic.eot");

  /\* IE9 Compat Modes \*/

  src: local("Poppins Light Italic"), local("Poppins-LightItalic"), url("../fonts/poppins/poppins-v5-latin-300italic.eot?#iefix") format("embedded-opentype"), url("../fonts/poppins/poppins-v5-latin-300italic.woff2") format("woff2"), url("../fonts/poppins/poppins-v5-latin-300italic.woff") format("woff"), url("../fonts/poppins/poppins-v5-latin-300italic.ttf") format("truetype"), url("../fonts/poppins/poppins-v5-latin-300italic.svg#Poppins") format("svg");

  /\* Legacy iOS \*/ }

/\* poppins-regular - latin \*/

@font-face {

  font-family: 'Poppins';

  font-style: normal;

  font-weight: 400;

  src: url("../fonts/poppins/poppins-v5-latin-regular.eot");

  /\* IE9 Compat Modes \*/

  src: local("Poppins Regular"), local("Poppins-Regular"), url("../fonts/poppins/poppins-v5-latin-regular.eot?#iefix") format("embedded-opentype"), url("../fonts/poppins/poppins-v5-latin-regular.woff2") format("woff2"), url("../fonts/poppins/poppins-v5-latin-regular.woff") format("woff"), url("../fonts/poppins/poppins-v5-latin-regular.ttf") format("truetype"), url("../fonts/poppins/poppins-v5-latin-regular.svg#Poppins") format("svg");

  /\* Legacy iOS \*/ }

/\* poppins-italic - latin \*/

@font-face {

  font-family: 'Poppins';

  font-style: italic;

  font-weight: 400;

  src: url("../fonts/poppins/poppins-v5-latin-italic.eot");

  /\* IE9 Compat Modes \*/

  src: local("Poppins Italic"), local("Poppins-Italic"), url("../fonts/poppins/poppins-v5-latin-italic.eot?#iefix") format("embedded-opentype"), url("../fonts/poppins/poppins-v5-latin-italic.woff2") format("woff2"), url("../fonts/poppins/poppins-v5-latin-italic.woff") format("woff"), url("../fonts/poppins/poppins-v5-latin-italic.ttf") format("truetype"), url("../fonts/poppins/poppins-v5-latin-italic.svg#Poppins") format("svg");

  /\* Legacy iOS \*/ }

/\* poppins-500 - latin \*/

@font-face {

  font-family: 'Poppins';

  font-style: normal;

  font-weight: 500;

  src: url("../fonts/poppins/poppins-v5-latin-500.eot");

  /\* IE9 Compat Modes \*/

  src: local("Poppins Medium"), local("Poppins-Medium"), url("../fonts/poppins/poppins-v5-latin-500.eot?#iefix") format("embedded-opentype"), url("../fonts/poppins/poppins-v5-latin-500.woff2") format("woff2"), url("../fonts/poppins/poppins-v5-latin-500.woff") format("woff"), url("../fonts/poppins/poppins-v5-latin-500.ttf") format("truetype"), url("../fonts/poppins/poppins-v5-latin-500.svg#Poppins") format("svg");

  /\* Legacy iOS \*/ }

/\* poppins-500italic - latin \*/

@font-face {

  font-family: 'Poppins';

  font-style: italic;

  font-weight: 500;

  src: url("../fonts/poppins/poppins-v5-latin-500italic.eot");

  /\* IE9 Compat Modes \*/

  src: local("Poppins Medium Italic"), local("Poppins-MediumItalic"), url("../fonts/poppins/poppins-v5-latin-500italic.eot?#iefix") format("embedded-opentype"), url("../fonts/poppins/poppins-v5-latin-500italic.woff2") format("woff2"), url("../fonts/poppins/poppins-v5-latin-500italic.woff") format("woff"), url("../fonts/poppins/poppins-v5-latin-500italic.ttf") format("truetype"), url("../fonts/poppins/poppins-v5-latin-500italic.svg#Poppins") format("svg");

  /\* Legacy iOS \*/ }

/\* poppins-600 - latin \*/

@font-face {

  font-family: 'Poppins';

  font-style: normal;

  font-weight: 600;

  src: url("../fonts/poppins/poppins-v5-latin-600.eot");

  /\* IE9 Compat Modes \*/

  src: local("Poppins SemiBold"), local("Poppins-SemiBold"), url("../fonts/poppins/poppins-v5-latin-600.eot?#iefix") format("embedded-opentype"), url("../fonts/poppins/poppins-v5-latin-600.woff2") format("woff2"), url("../fonts/poppins/poppins-v5-latin-600.woff") format("woff"), url("../fonts/poppins/poppins-v5-latin-600.ttf") format("truetype"), url("../fonts/poppins/poppins-v5-latin-600.svg#Poppins") format("svg");

  /\* Legacy iOS \*/ }

/\* poppins-700 - latin \*/

@font-face {

  font-family: 'Poppins';

  font-style: normal;

  font-weight: 700;

  src: url("../fonts/poppins/poppins-v5-latin-700.eot");

  /\* IE9 Compat Modes \*/

  src: local("Poppins Bold"), local("Poppins-Bold"), url("../fonts/poppins/poppins-v5-latin-700.eot?#iefix") format("embedded-opentype"), url("../fonts/poppins/poppins-v5-latin-700.woff2") format("woff2"), url("../fonts/poppins/poppins-v5-latin-700.woff") format("woff"), url("../fonts/poppins/poppins-v5-latin-700.ttf") format("truetype"), url("../fonts/poppins/poppins-v5-latin-700.svg#Poppins") format("svg");

  /\* Legacy iOS \*/ }

/\* poppins-700italic - latin \*/

@font-face {

  font-family: 'Poppins';

  font-style: italic;

  font-weight: 700;

  src: url("../fonts/poppins/poppins-v5-latin-700italic.eot");

  /\* IE9 Compat Modes \*/

  src: local("Poppins Bold Italic"), local("Poppins-BoldItalic"), url("../fonts/poppins/poppins-v5-latin-700italic.eot?#iefix") format("embedded-opentype"), url("../fonts/poppins/poppins-v5-latin-700italic.woff2") format("woff2"), url("../fonts/poppins/poppins-v5-latin-700italic.woff") format("woff"), url("../fonts/poppins/poppins-v5-latin-700italic.ttf") format("truetype"), url("../fonts/poppins/poppins-v5-latin-700italic.svg#Poppins") format("svg");

  /\* Legacy iOS \*/ }

/\* poppins-800 - latin \*/

@font-face {

  font-family: 'Poppins';

  font-style: normal;

  font-weight: 800;

  src: url("../fonts/poppins/poppins-v5-latin-800.eot");

  /\* IE9 Compat Modes \*/

  src: local("Poppins ExtraBold"), local("Poppins-ExtraBold"), url("../fonts/poppins/poppins-v5-latin-800.eot?#iefix") format("embedded-opentype"), url("../fonts/poppins/poppins-v5-latin-800.woff2") format("woff2"), url("../fonts/poppins/poppins-v5-latin-800.woff") format("woff"), url("../fonts/poppins/poppins-v5-latin-800.ttf") format("truetype"), url("../fonts/poppins/poppins-v5-latin-800.svg#Poppins") format("svg");

  /\* Legacy iOS \*/ }

/\* poppins-800italic - latin \*/

@font-face {

  font-family: 'Poppins';

  font-style: italic;

  font-weight: 800;

  src: url("../fonts/poppins/poppins-v5-latin-800italic.eot");

  /\* IE9 Compat Modes \*/

  src: local("Poppins ExtraBold Italic"), local("Poppins-ExtraBoldItalic"), url("../fonts/poppins/poppins-v5-latin-800italic.eot?#iefix") format("embedded-opentype"), url("../fonts/poppins/poppins-v5-latin-800italic.woff2") format("woff2"), url("../fonts/poppins/poppins-v5-latin-800italic.woff") format("woff"), url("../fonts/poppins/poppins-v5-latin-800italic.ttf") format("truetype"), url("../fonts/poppins/poppins-v5-latin-800italic.svg#Poppins") format("svg");

  /\* Legacy iOS \*/ }

/\* poppins-900 - latin \*/

@font-face {

  font-family: 'Poppins';

  font-style: normal;

  font-weight: 900;

  src: url("../fonts/poppins/poppins-v5-latin-900.eot");

  /\* IE9 Compat Modes \*/

  src: local("Poppins Black"), local("Poppins-Black"), url("../fonts/poppins/poppins-v5-latin-900.eot?#iefix") format("embedded-opentype"), url("../fonts/poppins/poppins-v5-latin-900.woff2") format("woff2"), url("../fonts/poppins/poppins-v5-latin-900.woff") format("woff"), url("../fonts/poppins/poppins-v5-latin-900.ttf") format("truetype"), url("../fonts/poppins/poppins-v5-latin-900.svg#Poppins") format("svg");

  /\* Legacy iOS \*/ }

a:focus, a:active {

  text-decoration: none;

  outline: none;

  transition: all 300ms ease 0s;

  -moz-transition: all 300ms ease 0s;

  -webkit-transition: all 300ms ease 0s;

  -o-transition: all 300ms ease 0s;

  -ms-transition: all 300ms ease 0s; }

input, select, textarea {

  outline: none;

  appearance: unset !important;

  -moz-appearance: unset !important;

  -webkit-appearance: unset !important;

  -o-appearance: unset !important;

  -ms-appearance: unset !important; }

input::-webkit-outer-spin-button, input::-webkit-inner-spin-button {

  appearance: none !important;

  -moz-appearance: none !important;

  -webkit-appearance: none !important;

  -o-appearance: none !important;

  -ms-appearance: none !important;

  margin: 0; }

input:focus, select:focus, textarea:focus {

  outline: none;

  box-shadow: none !important;

  -moz-box-shadow: none !important;

  -webkit-box-shadow: none !important;

  -o-box-shadow: none !important;

  -ms-box-shadow: none !important; }

input[type=checkbox] {

  appearance: checkbox !important;

  -moz-appearance: checkbox !important;

  -webkit-appearance: checkbox !important;

  -o-appearance: checkbox !important;

  -ms-appearance: checkbox !important; }

input[type=radio] {

  appearance: radio !important;

  -moz-appearance: radio !important;

  -webkit-appearance: radio !important;

  -o-appearance: radio !important;

  -ms-appearance: radio !important; }

img {

  max-width: 100%;

  height: auto; }

figure {

  margin: 0; }

p {

  margin-bottom: 0px; }

h1 {

  line-height: 1.8;

  margin: 0;

  padding: 0;

  font-weight: bold;

  color: #fff;

  font-family: Poppins;

  -webkit-text-stroke: 1px grey;

  font-size: 40px;

  text-align: center;

  margin-bottom: 0px; }

h2 {

  line-height: 1.8;

  margin: 0;

  padding: 0;

  font-weight: bold;

  color: #fff;

  font-family: Poppins;

  font-size: 17px;

  text-align: center;

  margin-bottom: 5px; }

h3 {

  line-height: 1.8;

  margin: 0;

  padding: 0;

  font-weight: bold;

  color: #fff;

  font-family: Poppins;

  font-size: 26px;

  text-align: center;

  margin-bottom: 28px; }

.clear {

  clear: both; }

body {

  font-size: 14px;

  line-height: 1.8;

  color: #fff;

  background-image: url("../images/form-img.jpg");

  background-repeat: no-repeat;

  background-size: cover;

  -moz-background-size: cover;

  -webkit-background-size: cover;

  -o-background-size: cover;

  -ms-background-size: cover;

  background-position: center center;

  font-weight: 600;

  font-family: Poppins;

  margin: 0px; }

.plot-container {

    max-width: 100%;

    margin: 0 auto;

    text-align: center;

}

.plot-container img {

    max-width: 40%;

    height: auto;

}

.main {

  position: relative; }

.container {

  width: 90%;

  height: 100%;

  align-self: auto;

  background: rgba(11, 13, 24, 0.6);

  margin: 135px auto;

  border-radius: 10px;

  -moz-border-radius: 10px;

  -webkit-border-radius: 10px;

  -o-border-radius: 10px;

  -ms-border-radius: 10px; }

.booking-form {

  padding: 35px 70px 51px 70px; }

.header {

  align-items: center;

  -moz-align-items: center;

  -webkit-align-items: center;

  -o-align-items: center;

  -ms-align-items: center;

  justify-content: space-between;

  -moz-justify-content: space-between;

  -webkit-justify-content: space-between;

  -o-justify-content: space-between;

  -ms-justify-content: space-between;

  padding: 50px 57px; }

input {

  border: none;

  background: #fff;

  border-style: solid;

  border-width: 1px

  border-radius: 5px;

  -moz-border-radius: 5px;

  -webkit-border-radius: 5px;

  -o-border-radius: 5px;

  -ms-border-radius: 5px;

  box-sizing: border-box;

  width: 100%;

  padding: 13px 20px;

  margin: 5px 0;

  box-sizing: border-box;

  font-size: 16px;

  font-weight: 600;

  color: #222;

  display: block;

  font-family: Poppins;

  position: relative;

  z-index: 9;

}

  input::-webkit-input-placeholder {

    color: #999; }

  input::-moz-placeholder {

    color: #999; }

  input:-ms-input-placeholder {

    color: #999; }

  input:-moz-placeholder {

    color: #999; }

.submit {

  background: #f8ba0f;

  color: #fff;

  font-size: 14px;

  margin-top: 23px;

  padding: 15px 20px;

  cursor: pointer; }

  .submit:hover {

    background-color: #ce9906; }

label {

  font-size: 12px;

  font-weight: 500;

  margin-bottom: 2px;

  display: block; }

#quantity {

  color: #999; }

.form-group {

  align-items: center;

  -moz-align-items: center;

  -webkit-align-items: center;

  -o-align-items: center;

  -ms-align-items: center;

  justify-content: space-between;

  -moz-justify-content: space-between;

  -webkit-justify-content: space-between;

  -o-justify-content: space-between;

  -ms-justify-content: space-between; }

.form-destination {

  width: 230px; }

.form-date-from, .form-date-to {

  width: 155px; }

.form-quantity {

  width: 80px;

  position: relative; }

.form-submit {

  width: 140px; }

.modify-qty {

  position: absolute;

  right: 12px;

  font-size: 18px;

  color: #999;

  z-index: 99;

  cursor: pointer; }

.plus {

  top: 23px; }

.minus {

  bottom: 0px; }

.form-icon {

  position: relative; }

.ui-datepicker-trigger {

  position: absolute;

  right: 16px;

  top: 40px;

  color: #999;

  font-size: 16px;

  z-index: 99;

  background: transparent;

  border: none;

  outline: none;

  cursor: pointer; }

@media screen and (max-width: 768px) {

  .container {

    width: calc( 100% - 30px);

    max-width: 100%; }

  .form-group {

    flex-direction: column;

    -moz-flex-direction: column;

    -webkit-flex-direction: column;

    -o-flex-direction: column;

    -ms-flex-direction: column; }

  .form-destination, .form-date-from, .form-date-to, .form-quantity {

    width: 100%;

    margin-bottom: 20px; } }

@media screen and (max-width: 480px) {

  .booking-form {

    padding: 35px 30px 51px 30px;

     margin-left: 10px;} }

**Javascript code:**

(function ($) {

    // USE STRICT

    "use strict";

    $( "#date\_from" ).datepicker({

        dateFormat: "dd-mm-yy",

        showOn: "both",

        buttonText : '<i class="zmdi zmdi-calendar-alt"></i>',

    });

    $( "#date\_to" ).datepicker({

        dateFormat: "dd-mm-yy",

        showOn: "both",

        buttonText : '<i class="zmdi zmdi-calendar-alt"></i>',

    });

})(jQuery);

function Tang(){

    var x = document.getElementById("quantity").value;//lay gia tri cu trong text

    if(parseInt(x) >= 0){

        document.getElementById("quantity").value = parseInt(x) +1;// + gia tri lay dc len 1 roi gan kq vao o text

    }

}

function Giam(){

    var x = document.getElementById("quantity").value;

    if(parseInt(x) >= 1){

        document.getElementById("quantity").value = parseInt(x) -1;

    }

}

# APPENDIX III

## APPENDIX-III

## Published certificate and published paper