PROJECT REPORT

Date	19/11/2022
Team ID	PNT2022TMID18498
Name	Natural Disasters Intensity Analysis and Classification Using Artificial Intelligence

Table of Contents:

1. INTRODUCTION

a. PROJECT OVERVIEW

Natural disasters not only disturb the human ecological system but also destroy the properties and critical infrastructures of human societies and even lead to permanent change in the ecosystem. Disaster can be caused by naturally occurring events such as earthquakes, cyclones, floods, and wildfires. Many deep learning techniques have been applied by various researchers to detect and classify natural disasters to overcome losses in ecosystems, but detection of natural disasters still faces issues due to the complex and imbalanced structures of images. To tackle this problem, we developed a multilayered deep convolutional neural network model that classifies the natural disaster and tells the intensity of natural disaster.The model uses an integrated webcam to capture the video frame and the video frame is compared with the Pre-trained model and the type of disaster is identified and showcased on the OpenCV window.

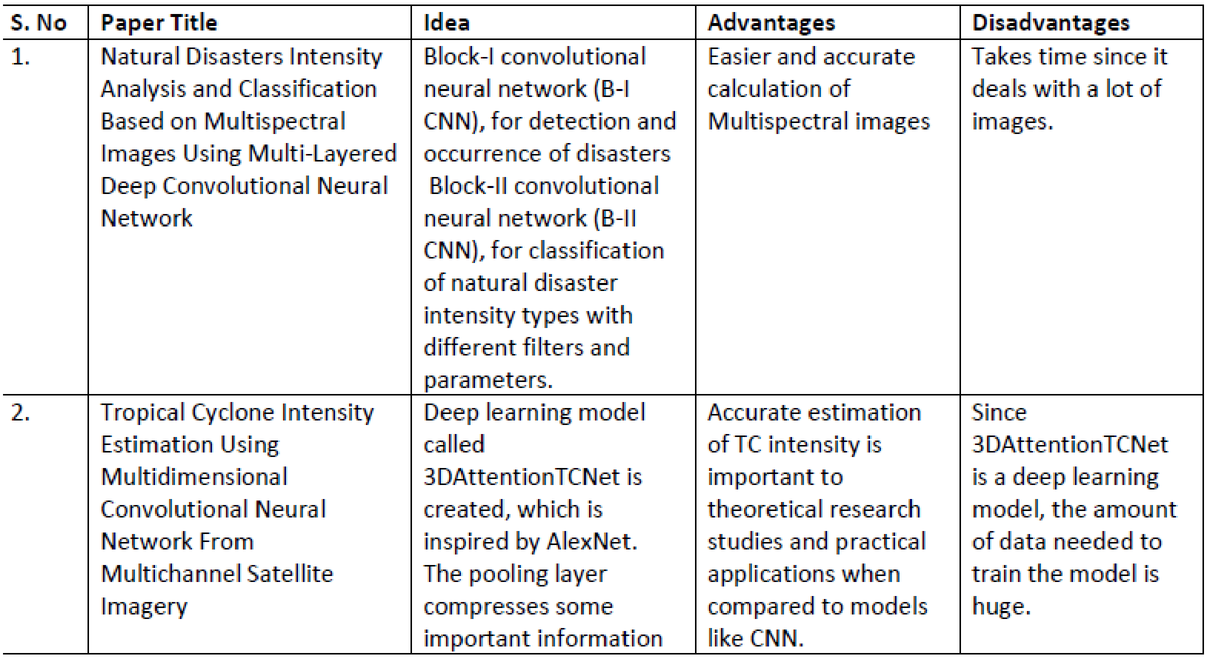
b. PURPOSE:

The purpose of the project is to classify the natural disaster based on the image given as input. The image shown in the Webcam is analyzed and algorithms like CNN are applied to the image to produce the classification result. The project is useful to NDRF(National Disaster Response Force) employees and the students.

2. LITERATURE SURVEY

a. EXISTING PROBLEM:

The research papers related to the Natural Disaster Intensity Analysis and Classification are given below where the advantages and disadvantages are listed.



S. No	Paper Title	Idea	Advantages	Disadvantages
1.	Natural Disasters Intensity Analysis and Classification Based on Multispectral Images Using Multi-Layered Deep Convolutional Neural Network	Block-I convolutional neural network (B-I CNN), for detection and occurrence of disasters Block-II convolutional neural network (B-II CNN), for classification of natural disaster intensity types with different filters and parameters.	Easier and accurate calculation of Multispectral images	Takes time since it deals with a lot of images.
2.	Tropical Cyclone Intensity Estimation Using Multidimensional Convolutional Neural Network From Multichannel Satellite Imagery	Deep learning model called 3DAttentionTCNet is created, which is inspired by AlexNet. The pooling layer compresses some important information	Accurate estimation of TC intensity is important to theoretical research studies and practical applications when compared to models like CNN.	Since 3DAttentionTCNet is a deep learning model, the amount of data needed to train the model is huge.

		resulting in the loss of some intensity features, we remove the pooling layers		
3.	Designing Deep-Based Learning Flood Forecast Model With ConvLSTM Hybrid Algorithm	A robust mathematical tool used to determine the flood state at a particular time for a given area is the Flood Index (IF). A model is developed using ConvLSTM, as an objective model, with alternative methods of LSTM, CNN-LSTM and SVR that can also determine the flood state.	Early detection of natural disasters such as floods can greatly assist humans in reducing the extent of the damage caused by such events. The accuracy is high when compared to other models.	Since model developed using ConvLSTM is a deep learning model, the amount of data needed to train the model is huge and also time and processor consuming.
4.	A Conformal Regressor With Random Forests for Tropical Cyclone Intensity Estimation	A multiple linear regression (MLR) model was constructed based on the extraction of the most significant signals and parameters from satellite infrared images.	It is considered an excellent way to extract features from satellite images to estimate TC intensity. The Dvorak technique tried to estimate the TC intensity using visible or infrared images based on the cloud structure.	The MLR regression technique is exactly not suitable for all the scenarios of images.
5.	Rainformer: Features Extraction Balanced Network for Radar-Based Precipitation Nowcasting	Framework: Rainformer Rainformer consists of an encoder (green box) and decoder (blue box). They both have four stages. When the stage goes deeper, the feature size becomes smaller. Both encoder and decoder include FEBM. FEBM enhances the low to medium and high-intensity rainfall features at every stage.	It can extract global and local features from radar echo maps separately, and fuses balanced these two features to enhance the model's ability to predict heavy rain or rainstorm.	The Rainformer model is processor complex and also the encoding may not be very efficient.
6.	Quantifying change after natural disasters to estimate infrastructure damage with mobile phone data	It indicates that how mobility patterns are changing, in the post disaster timeframe, is crucial in order to settle rescue centers and send help to the most affected areas. We describe the approach taken to work with aggregated CDR data.	We analyzed the relationship between the reach score changes and the damage index of the earthquake in urban areas, and it showed that the correlation was negative on the day after the natural disaster.	The mobile phone data is sometimes not sufficient for better quantification.

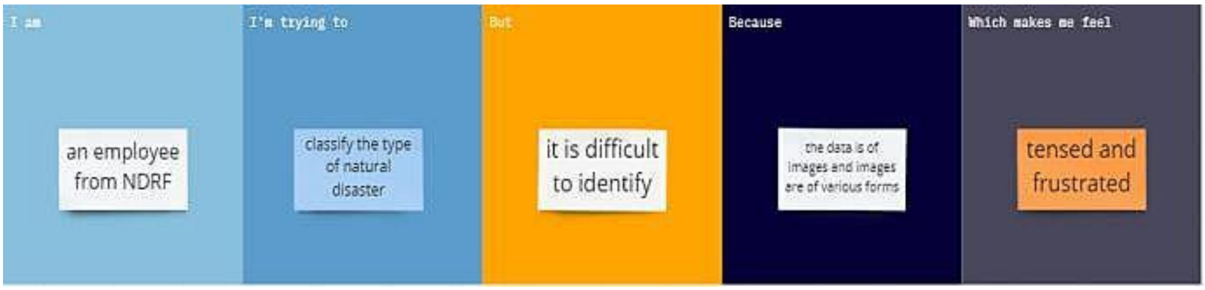
b. REFERENCES

1. Tonini M., D’Andrea M., Biondi G., Degli Esposti S., Trucchia A., Fiorucci P. A Machine Learning- Based Approach forWildfire Susceptibility Mapping. The Case Study of the Liguria Region in Italy.
2. Amit S.N.K.B., Aoki Y. Disaster detection from aerial imagery with convolutional neural network; Proceedings of the2017 International Electronics Symposium on Knowledge Creation and Intelligent Computing (IES-KCIC); Surabaya,Indonesia.
3. Padmawar P.M., Shinde A.S., Sayyed T.Z., Shinde S.K., Moholkar K. Disaster Prediction System using Convolution NeuralNetwork; Proceedings of the 2019 International Conference on Communication and Electronics Systems (ICCES);Coimbatore, India.
4. Nguyen D.T., Ofli F., Imran M., Mitra P. Damage assessment from social media imagery data during disasters;Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining; Sydney,NSW, Australia.
5. D. Han, L. Chan, and N. Zhu, ``Flood forecasting using support vector machines''
6. X. H. Le, H. V. Ho, G. Lee, and S. Jung, ``Application of long short-term memory (LSTM) neural network for flood forecasting”
7. M. F. Piñeros, E. A. Ritchie, and J. S. Tyo, “Estimating tropical cyclone intensity from infrared image data”
8. T. L. Olander and C. S. Velden, “Tropical cyclone convection and intensity analysis using differenced infrared and water vapor imagery”.
9. X. Shi et al., “Deep learning for precipitation nowcasting: A benchmark and a new model”.

c. PROBLEM STATEMENT DEFINITION

PROBLEM STATEMENT 1:

I am a student and my aim is to classify the natural disaster but the It is difficult to identify the disaster because the data is of images and images are of various forms. So it makes me feel tensed and frustrated.



I am

an employee from NDRF

I'm trying to

classify the type of natural disaster

But

it is difficult to identify

Because

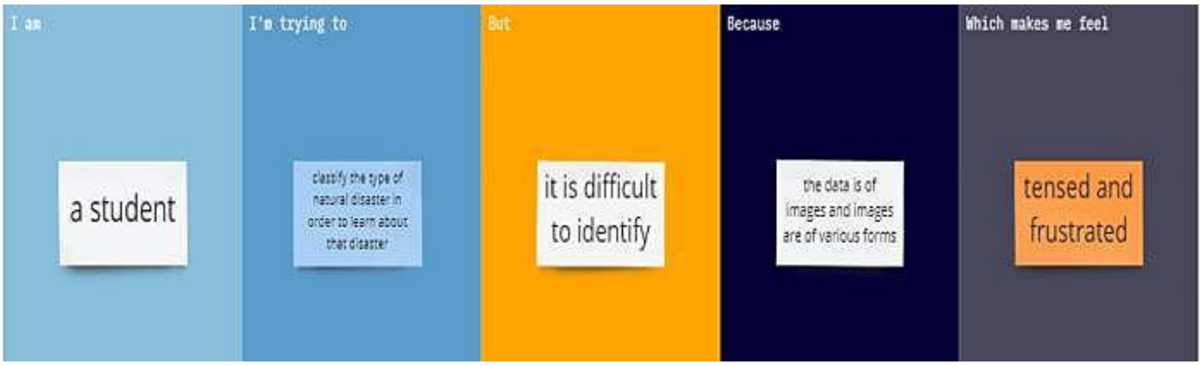
the data is of images and images are of various forms

Which makes me feel

tensed and frustrated

PROBLEM STATEMENT 2:

I am a NDRF Employee and my aim is to classify the natural disaster but the It is difficult to identify the disaster because the data is of images and images are of various forms. So it makes me feel tensed and frustrated.



I am

a student

I'm trying to

classify the type of natural disaster in order to learn about that disaster

But

it is difficult to identify

Because

the data is of images and images are of various forms

Which makes me feel

tensed and frustrated

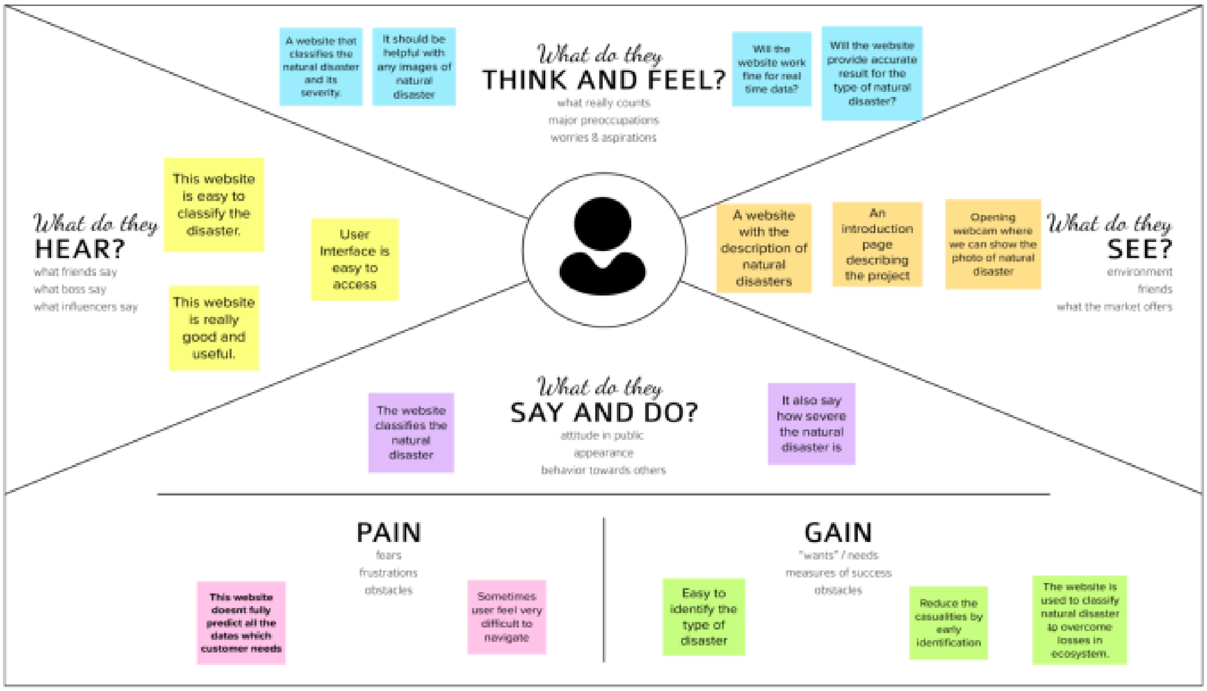
3. IDEATION & PROPOSED SOLUTION:

A. EMPATHY MAP CANVAS:

An empathy map is a simple, easy-to-digest visual that captures knowledge about a user’s behaviours and attitudes.

It is a useful tool to helps teams better understand their users.
Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user’s perspective along with his or her goals and challenges.

An empathy map shows the perspective of the employee from NDRF while using the website



A website that classifies the natural disaster and its severity.

It should be helpful with any images of natural disaster

Will the website work fine for real time data?

Will the website provide accurate result for the type of natural disaster?

This website is easy to classify the disaster.

User Interface is easy to access

This website is really good and useful.

A website with the description of natural disasters

An introduction page describing the project

Opening webcam where we can show the photo of natural disaster

The website classifies the natural disaster

It also say how severe the natural disaster is

This website doesnt fully predict all the datas which customer needs

Sometimes user feel very difficult to navigate

Easy to identify the type of disaster

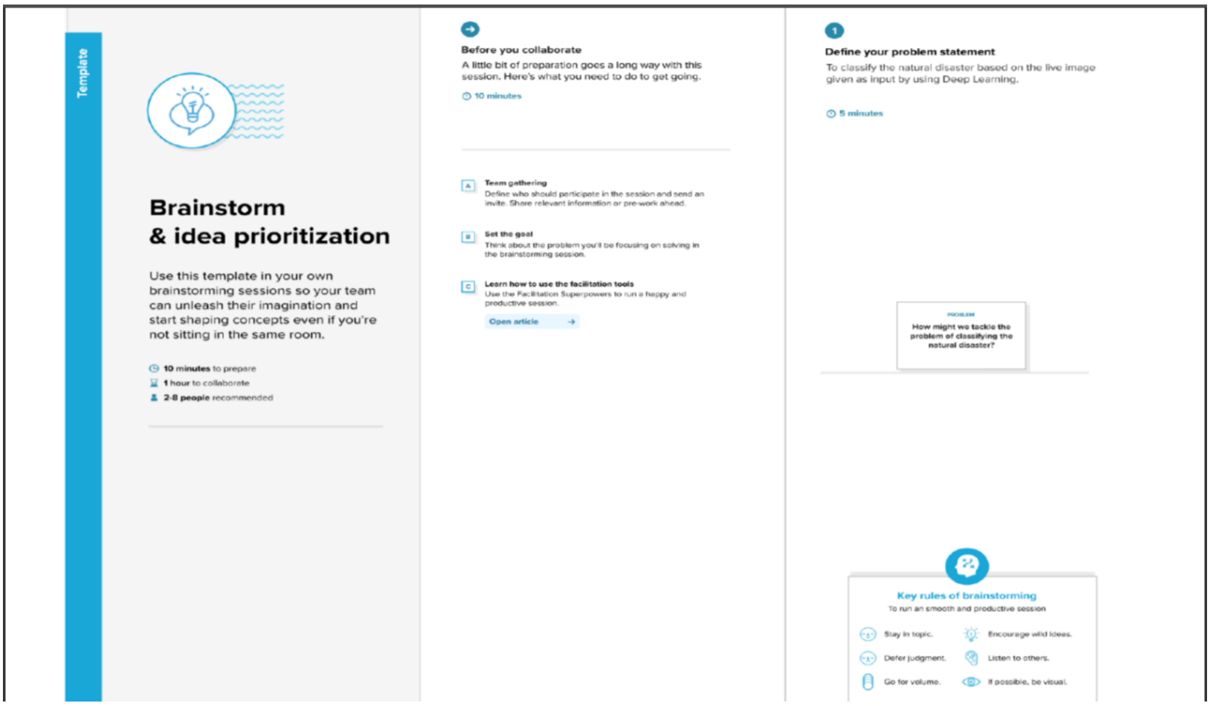
Reduce the casualities by early identification

The website is used to classify natural disaster to overcome losses in ecosystem.

B. IDEATION & BRAINSTORMING:

Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem solving. Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich amount of creative solutions.

STEP1:TEAM GATHERING,COLLABORATION AND SELECT THE PROBLEM STATEMENT



Brainstorm
& idea prioritization

Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

10 minutes to prepare
1 hour to collaborate
2-8 people recommended

Before you collaborate
A little bit of preparation goes a long way with this session. Here's what you need to do to get going.

10 minutes

A Team gathering
Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.

B Set the goal
Think about the problem you'll be focusing on solving in the brainstorming session.

C Learn how to use the facilitation tools
Use the Facilitation Superpowers to run a happy and productive session.

Define your problem statement
To classify the natural disaster based on the live image given as input by using Deep Learning.

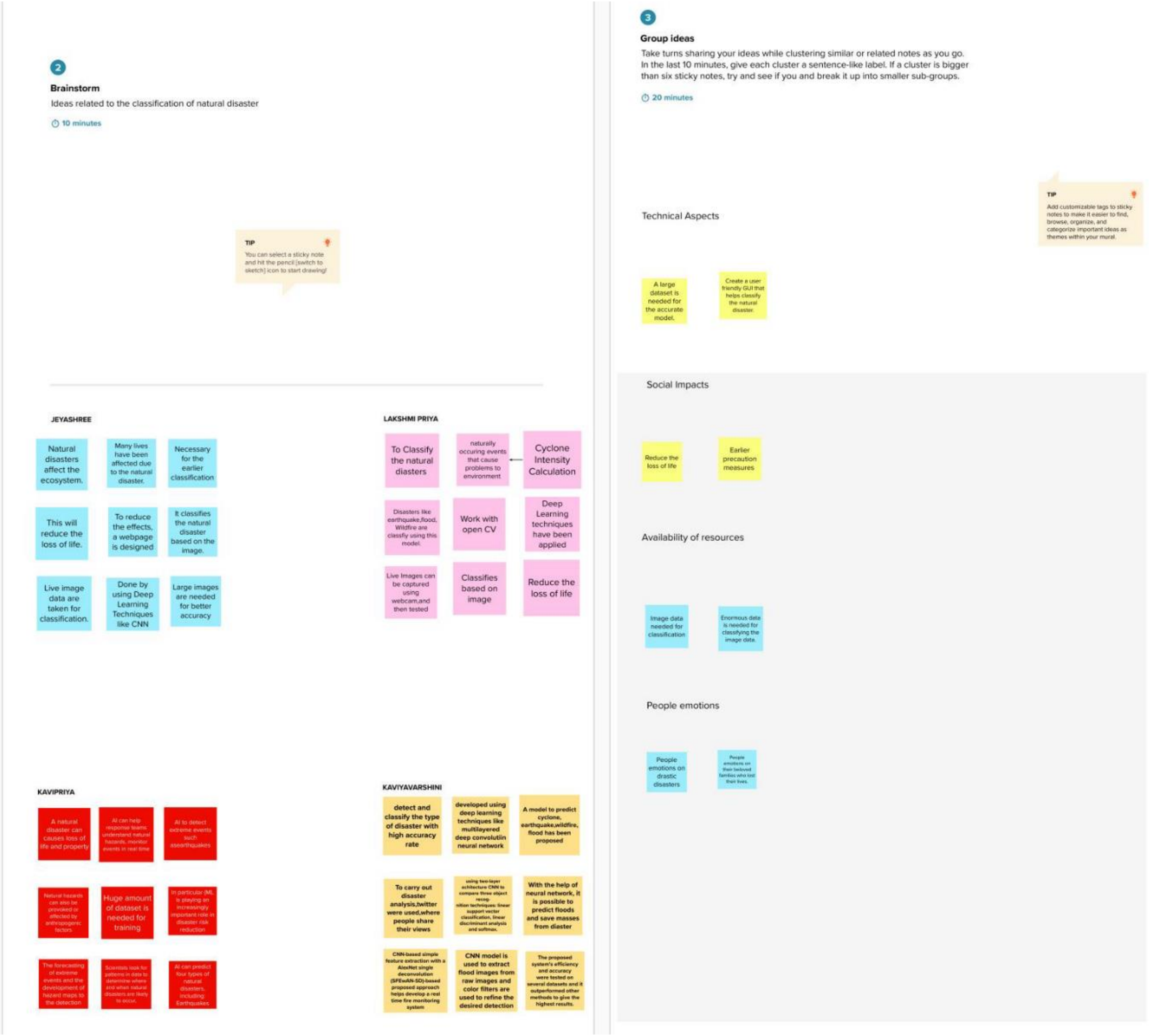
5 minutes

PROBLEM
How might we tackle the problem of classifying the natural disaster?

Key rules of brainstorming
To run an smooth and productive session

Stay in topic.
Encourage wild ideas.
Defer judgment.
Listen to others.
Go for volume.
If possible, be visual.

STEP-2: BRAINSTORM, IDEA LISTING AND GROUPING



2

Brainstorm

Ideas related to the classification of natural disaster

10 minutes

TIP

You can select a sticky note and hit the pencil [switch to sketch] icon to start drawing!

JEYASHREE

Natural disasters affect the ecosystem.

Many lives have been affected due to the natural disaster.

Necessary for the earlier classification

This will reduce the loss of life.

To reduce the effects, a webpage is designed

It classifies the natural disaster based on the image.

Live image data are taken for classification.

Done by using Deep Learning Techniques like CNN

Large images are needed for better accuracy

LAKSHMI PRIYA

To Classify the natural diasters

naturally occuring events that cause problems to environment

Cyclone Intensity Calculation

Disasters like earthquake,flood, Wildfire are classfiy using this model.

Work with open CV

Deep Learning techniques have been applied

Live Images can be captured using webcam,and then tested

Classifies based on image

Reduce the loss of life

KAVIPRIYA

A natural disaster can causes loss of life and property

AI can help response teams understand natural hazards, monitor events in real time

AI to detect extreme events such asearthquakes

Natural hazards can also be provoked or affected by anthropogenic factors

Huge amount of dataset is needed for training

In particular (ML is playing an increasingly important role in disaster risk reduction

The forecasting of extreme events and the development of hazard maps to the detection

Scientists look for patterns in data to determine where and when natural disasters are likely to occur.

AI can predict four types of natural disasters, including Earthquakes

KAVIYAVARSHINI

detect and classify the type of disaster with high accuracy rate

developed using deep learning techniques like multilayered deep convolutiin neural network

A model to predict cyclone, earthquake,wildfire, flood has been proposed

To carry out disaster analysis,twitter were used,where people share their views

using two-layer achitecture CNN to compare three object recog- nition techniques: linear support vector classification, linear discriminant analysis and softmax.

With the help of neural network, it is possible to predict floods and save masses from diaster

CNN-based simple feature extraction with a AlexNet single deconvolution (SFEwAN-SD)-based proposed approach helps develop a real time fire monitoring system

CNN model is used to extract flood images from raw images and color filters are used to refine the desired detection

The proposed system's efficiency and accuracy were tested on several datasets and it outperformed other methods to give the highest results.

3

Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. In the last 10 minutes, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

20 minutes

TIP

Add customizable tags to sticky notes to make it easier to find, browse, organize, and categorize important ideas as themes within your mural.

Technical Aspects

A large dataset is needed for the accurate model.

Create a user friendly GUI that helps classify the natural disaster.

Social Impacts

Reduce the loss of life

Earlier precaution measures

Availability of resources

Image data needed for classification

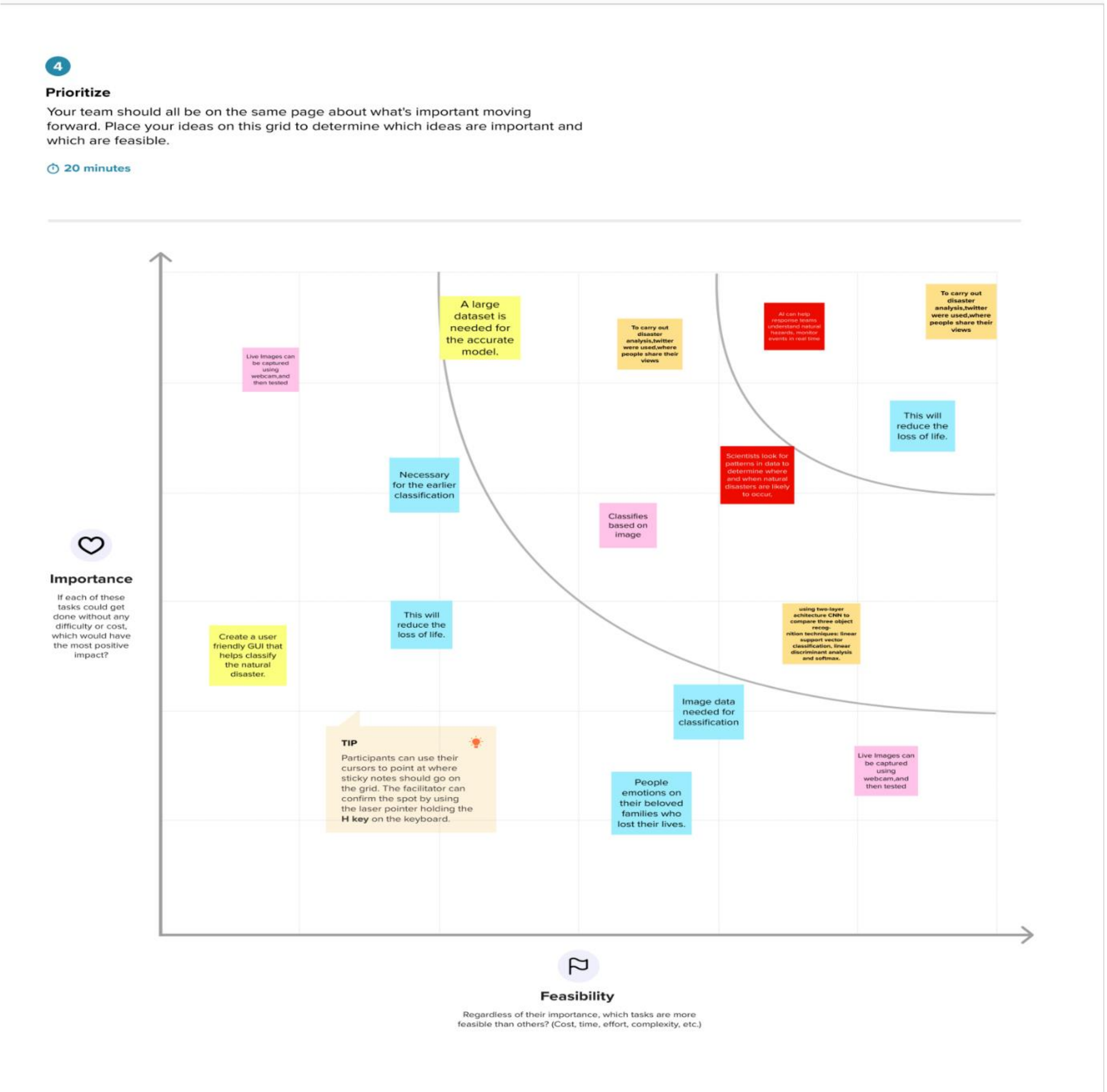
Enormous data is needed for classifying the image data.

People emotions

People emotions on drastic disasters

People emotions on their beloved families who lost their lives.

STEP-3: IDEA PRIORITIZATION



4

Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

20 minutes

Importance

If each of these tasks could get done without any difficulty or cost, which would have the most positive impact?

Live Images can be captured using webcam,and then tested

A large dataset is needed for the accurate model.

To carry out disaster analysis,twitter were used,where people share their views

AI can help response teams understand natural hazards, monitor events in real time

To carry out disaster analysis,twitter were used,where people share their views

This will reduce the loss of life.

Necessary for the earlier classification

Classifies based on image

Scientists look for patterns in data to determine where and when natural disasters are likely to occur,

Create a user friendly GUI that helps classify the natural disaster.

This will reduce the loss of life.

using two-layer achitecture CNN to compare three object recog-
nition techniques: linear support vector classification, linear discriminant analysis and softmax.

Image data needed for classification

Live Images can be captured using webcam,and then tested

TIP

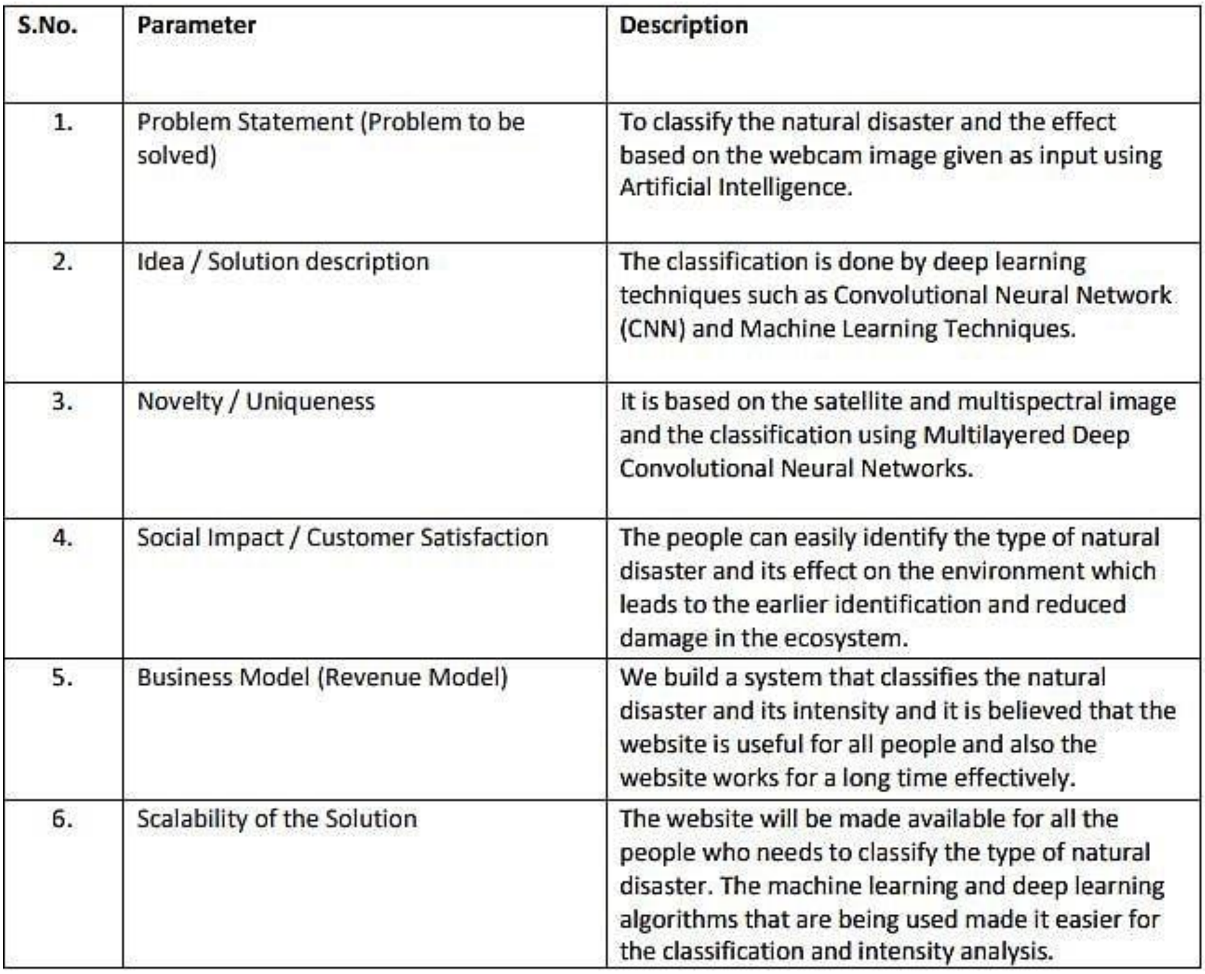
Participants can use their cursors to point at where sticky notes should go on the grid. The facilitator can confirm the spot by using the laser pointer holding the **H key** on the keyboard.

People emotions on their beloved families who lost their lives.

Feasibility

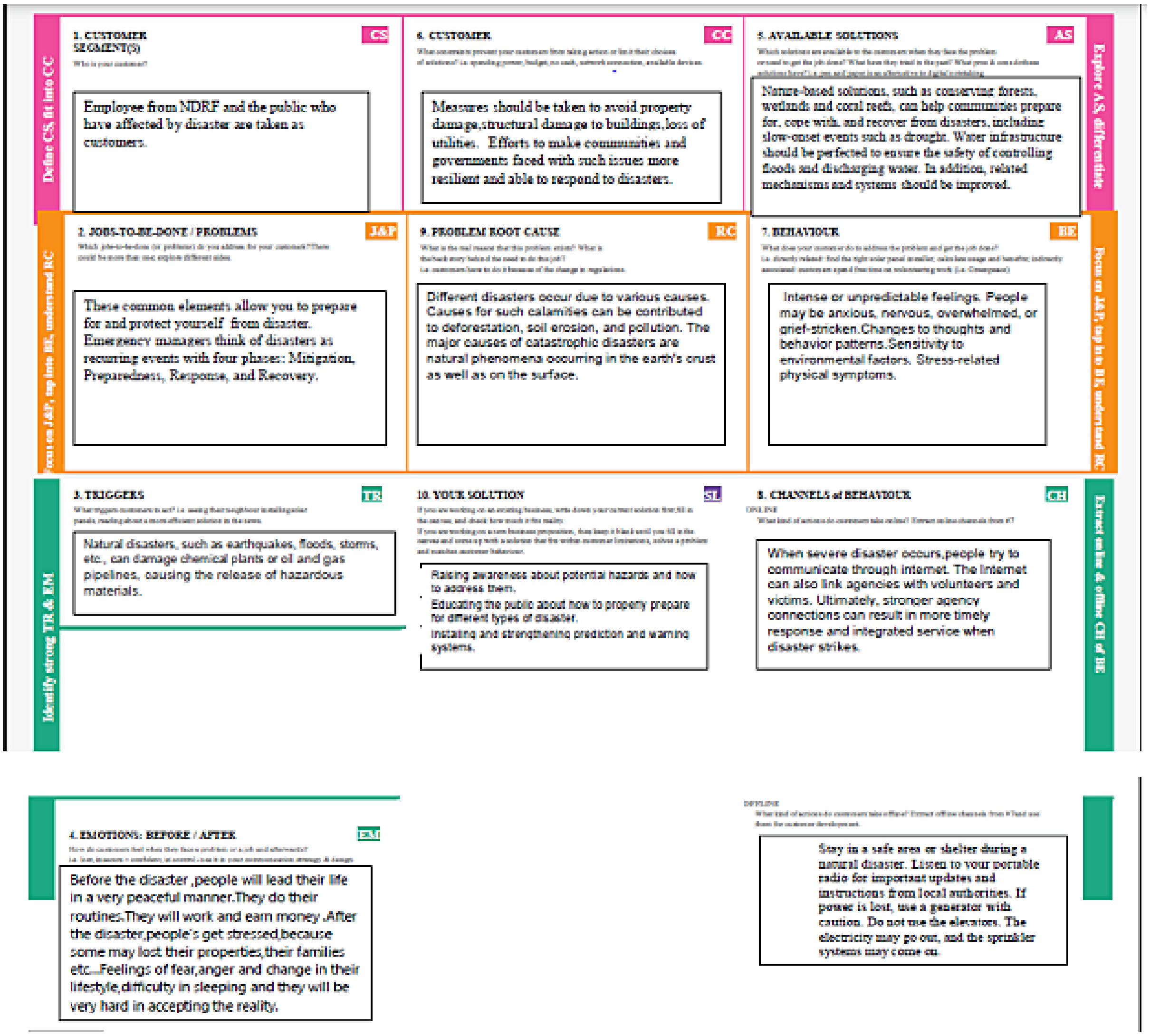
Regardless of their importance, which tasks are more feasible than others? (Cost, time, effort, complexity, etc.)

C. PROPOSED SOLUTION:



S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	To classify the natural disaster and the effect based on the webcam image given as input using Artificial Intelligence.
2.	Idea / Solution description	The classification is done by deep learning techniques such as Convolutional Neural Network (CNN) and Machine Learning Techniques.
3.	Novelty / Uniqueness	It is based on the satellite and multispectral image and the classification using Multilayered Deep Convolutional Neural Networks.
4.	Social Impact / Customer Satisfaction	The people can easily identify the type of natural disaster and its effect on the environment which leads to the earlier identification and reduced damage in the ecosystem.
5.	Business Model (Revenue Model)	We build a system that classifies the natural disaster and its intensity and it is believed that the website is useful for all people and also the website works for a long time effectively.
6.	Scalability of the Solution	The website will be made available for all the people who needs to classify the type of natural disaster. The machine learning and deep learning algorithms that are being used made it easier for the classification and intensity analysis.

D. PROBLEM SOLUTION FIT


Define CS, fit into CC
Explore AS, differentiate
Focus on J&P, tap into BE, understand RC
Focus on J&P, tap into BE, understand RC
Identify strong TR & EM
Extract online & offline CH of BE

1. CUSTOMER SEGMENT(S) CS

Who is your customer?

Employee from NDRF and the public who have affected by disaster are taken as customers.

6. CUSTOMER CC

What constraints prevent your customers from taking action or limit their choices of solutions? i.e. spending power, budget, no cash, network connection, available devices.

Measures should be taken to avoid property damage,structural damage to buildings,loss of utilities. Efforts to make communities and governments faced with such issues more resilient and able to respond to disasters.

5. AVAILABLE SOLUTIONS AS

Which solutions are available to the customers when they face the problem or need to get the job done? What have they tried in the past? What pros & cons do these solutions have? i.e. pen and paper is an alternative to digital notetaking

Nature-based solutions, such as conserving forests, wetlands and coral reefs, can help communities prepare for, cope with, and recover from disasters, including slow-onset events such as drought. Water infrastructure should be perfected to ensure the safety of controlling floods and discharging water. In addition, related mechanisms and systems should be improved.

2. JOBS-TO-BE-DONE / PROBLEMS J&P

Which jobs-to-be-done (or problems) do you address for your customers? There could be more than one; explore different sides.

These common elements allow you to prepare for and protect yourself from disaster. Emergency managers think of disasters as recurring events with four phases: Mitigation, Preparedness, Response, and Recovery.

9. PROBLEM ROOT CAUSE RC

What is the real reason that this problem exists? What is the back story behind the need to do this job?
i.e. customers have to do it because of the change in regulations.

Different disasters occur due to various causes. Causes for such calamities can be contributed to deforestation, soil erosion, and pollution. The major causes of catastrophic disasters are natural phenomena occurring in the earth's crust as well as on the surface.

7. BEHAVIOUR BE

What does your customer do to address the problem and get the job done?
i.e. directly related: find the right solar panel installer, calculate usage and benefits; indirectly associated: customers spend free time on volunteering work (i.e. Greenpeace)

Intense or unpredictable feelings. People may be anxious, nervous, overwhelmed, or grief-stricken.Changes to thoughts and behavior patterns.Sensitivity to environmental factors. Stress-related physical symptoms.

3. TRIGGERS TR

What triggers customers to act? i.e. seeing their neighbour installing solar panels, reading about a more efficient solution in the news.

Natural disasters, such as earthquakes, floods, storms, etc., can damage chemical plants or oil and gas pipelines, causing the release of hazardous materials.

10. YOUR SOLUTION SL

If you are working on an existing business, write down your current solution first, fill in the canvas, and check how much it fits reality.
If you are working on a new business proposition, then keep it blank until you fill in the canvas and come up with a solution that fits within customer limitations, solves a problem and matches customer behaviour.

- Raising awareness about potential hazards and how to address them.
- Educating the public about how to properly prepare for different types of disaster.
- Installing and strengthening prediction and warning systems.

8. CHANNELS of BEHAVIOUR CH

ONLINE

What kind of actions do customers take online? Extract online channels from #7

When severe disaster occurs,people try to communicate through internet. The Internet can also link agencies with volunteers and victims. Ultimately, stronger agency connections can result in more timely response and integrated service when disaster strikes.

4. EMOTIONS: BEFORE / AFTER EM

How do customers feel when they face a problem or a job and afterwards?
i.e. lost, insecure > confident, in control - use it in your communication strategy & design.

Before the disaster ,people will lead their life in a very peaceful manner.They do their routines.They will work and earn money .After the disaster,people's get stressed,because some may lost their properties,their families etc...Feelings of fear,anger and change in their lifestyle,difficulty in sleeping and they will be very hard in accepting the reality.

OFFLINE

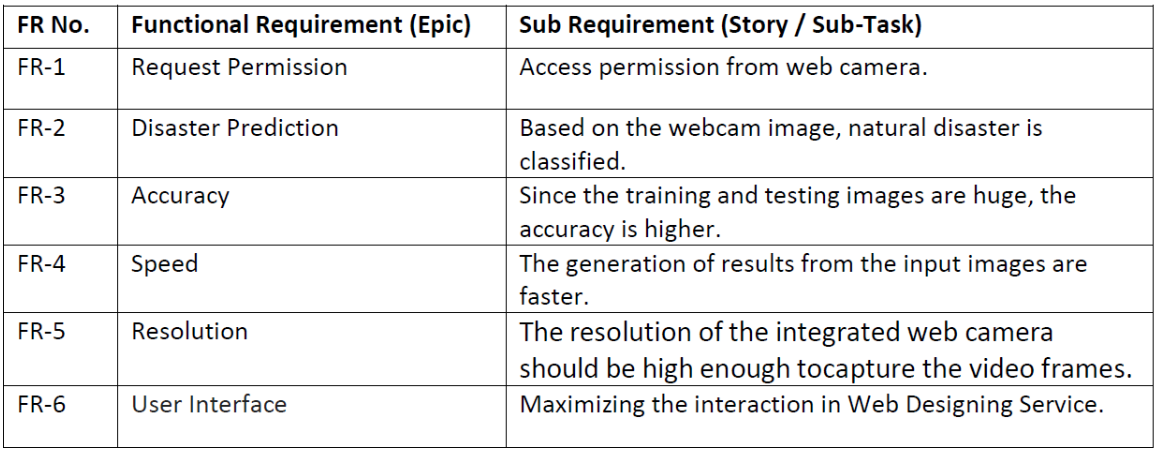
What kind of actions do customers take offline? Extract offline channels from #7 and use them for customer development.

Stay in a safe area or shelter during a natural disaster. Listen to your portable radio for important updates and instructions from local authorities. If power is lost, use a generator with caution. Do not use the elevators. The electricity may go out, and the sprinkler systems may come on.

4. REQUIREMENT ANALYSIS

a. FUNCTIONAL REQUIREMENT

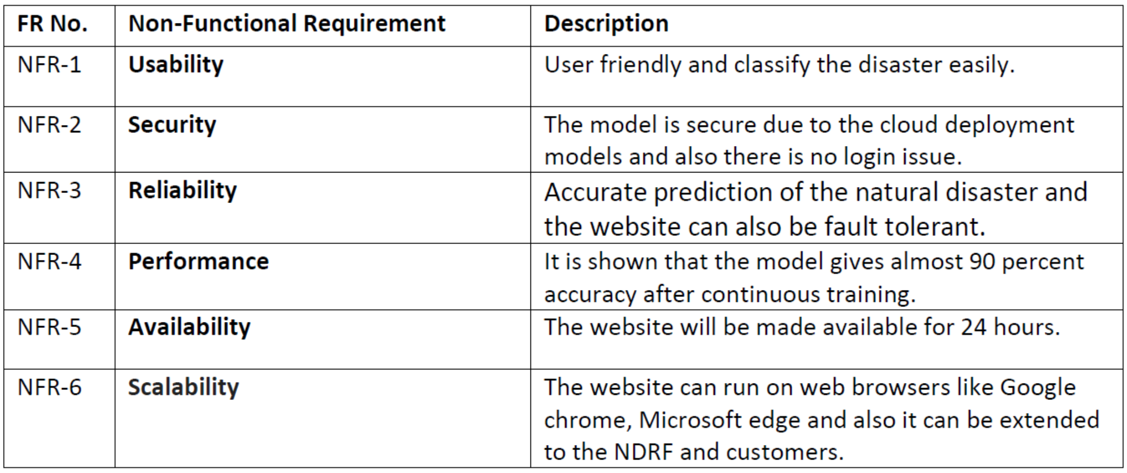
Following are the functional requirements of the proposed solution.



FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Request Permission	Access permission from web camera.
FR-2	Disaster Prediction	Based on the webcam image, natural disaster is classified.
FR-3	Accuracy	Since the training and testing images are huge, the accuracy is higher.
FR-4	Speed	The generation of results from the input images are faster.
FR-5	Resolution	The resolution of the integrated web camera should be high enough tocapture the video frames.
FR-6	User Interface	Maximizing the interaction in Web Designing Service.

b. NON FUNCTIONAL REQUIREMENTS

Following are the non-functional requirements of the proposed solution.

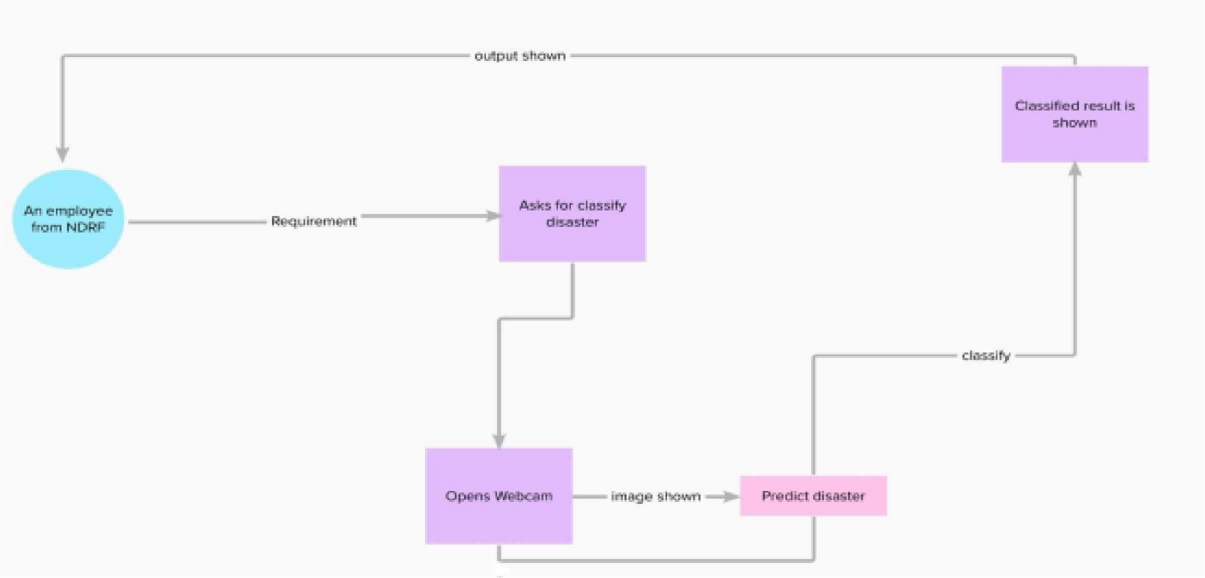


FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	User friendly and classify the disaster easily.
NFR-2	**Security**	The model is secure due to the cloud deployment models and also there is no login issue.
NFR-3	**Reliability**	Accurate prediction of the natural disaster and the website can also be fault tolerant.
NFR-4	**Performance**	It is shown that the model gives almost 90 percent accuracy after continuous training.
NFR-5	**Availability**	The website will be made available for 24 hours.
NFR-6	**Scalability**	The website can run on web browsers like Google chrome, Microsoft edge and also it can be extended to the NDRF and customers.

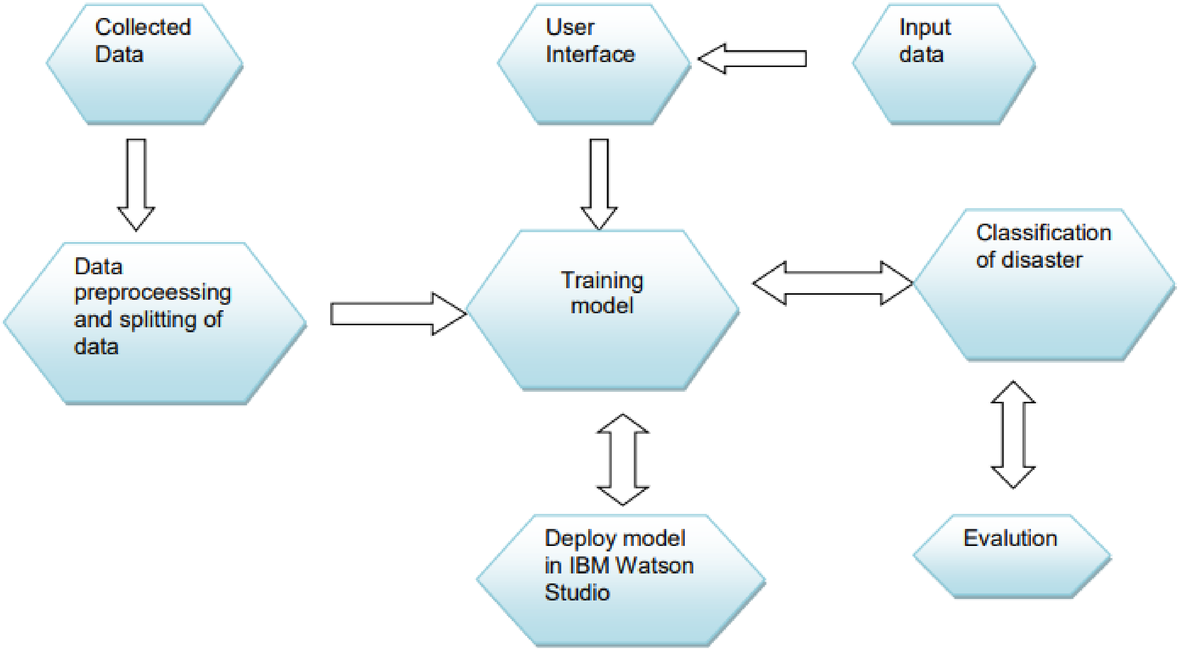
5. PROJECT DESIGN

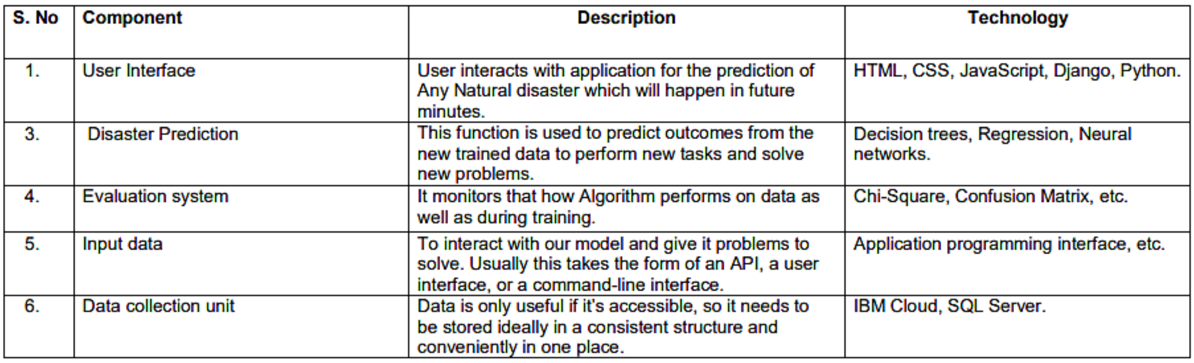
A. DATA FLOW DIAGRAMS:

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored



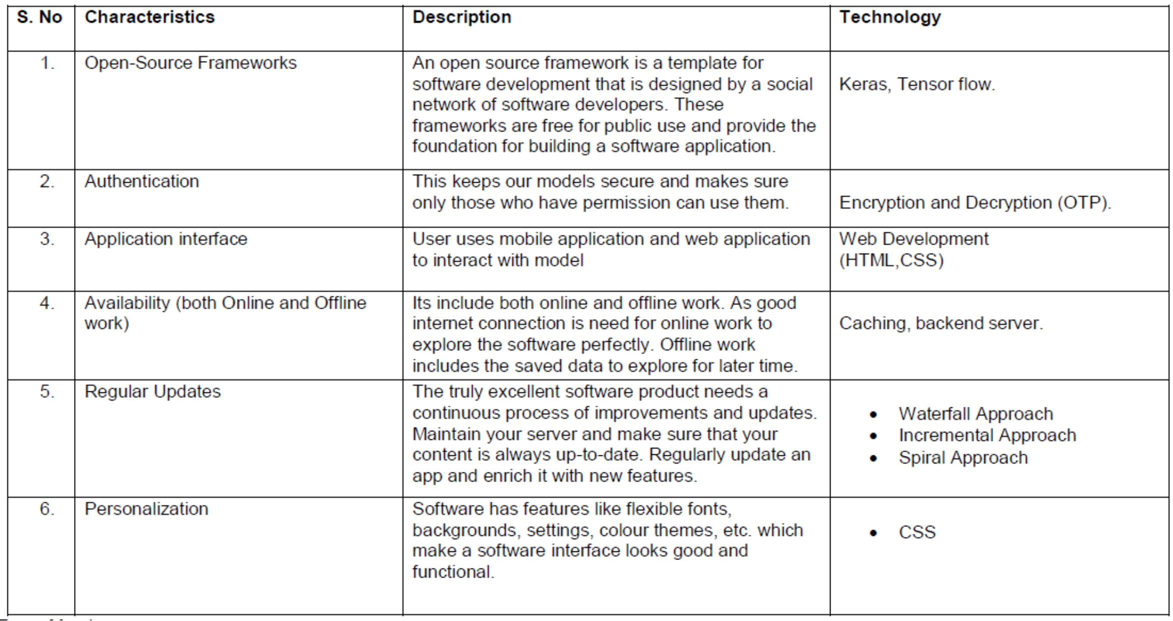
B. SOLUTION & TECHNICAL ARCHITECTURE


Collected Data
User Interface
Input data
Data preproceessing and splitting of data
Training model
Classification of disaster
Deploy model in IBM Watson Studio
Evalution



S. No	Component	Description	Technology
1.	User Interface	User interacts with application for the prediction of Any Natural disaster which will happen in future minutes.	HTML, CSS, JavaScript, Django, Python.
3.	Disaster Prediction	This function is used to predict outcomes from the new trained data to perform new tasks and solve new problems.	Decision trees, Regression, Neural networks.
4.	Evaluation system	It monitors that how Algorithm performs on data as well as during training.	Chi-Square, Confusion Matrix, etc.
5.	Input data	To interact with our model and give it problems to solve. Usually this takes the form of an API, a user interface, or a command-line interface.	Application programming interface, etc.
6.	Data collection unit	Data is only useful if it's accessible, so it needs to be stored ideally in a consistent structure and conveniently in one place.	IBM Cloud, SQL Server.

Table-2: Application Characteristics:



S. No	Characteristics	Description	Technology
1.	Open-Source Frameworks	An open source framework is a template for software development that is designed by a social network of software developers. These frameworks are free for public use and provide the foundation for building a software application.	Keras, Tensor flow.
2.	Authentication	This keeps our models secure and makes sure only those who have permission can use them.	Encryption and Decryption (OTP).
3.	Application interface	User uses mobile application and web application to interact with model	Web Development (HTML,CSS)
4.	Availability (both Online and Offline work)	Its include both online and offline work. As good internet connection is need for online work to explore the software perfectly. Offline work includes the saved data to explore for later time.	Caching, backend server.
5.	Regular Updates	The truly excellent software product needs a continuous process of improvements and updates. Maintain your server and make sure that your content is always up-to-date. Regularly update an app and enrich it with new features.	• Waterfall Approach • Incremental Approach • Spiral Approach
6.	Personalization	Software has features like flexible fonts, backgrounds, settings, colour themes, etc. which make a software interface looks good and functional.	• CSS

c. USER STORIES:

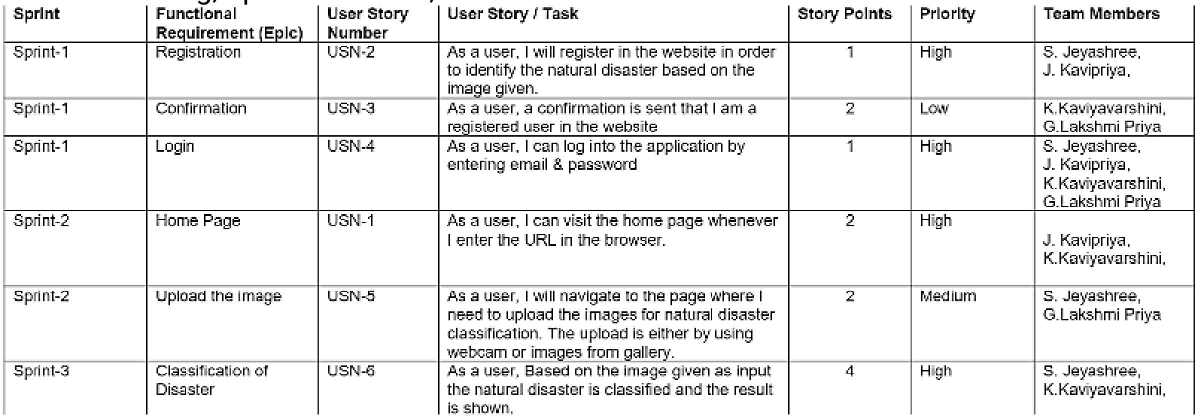
Here the list all the user stories for the project “Natural Disaster Intensity Analysis and Classification Using Artificial Intelligence”

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer	Registration	USN-1	As a user, registration should be done	Proper email id and password is accepted	High	Sprint-1
Customer	Area to be monitored	USN-2	As user ,I can particularly select the area to be continuously checked and analyzed	The areas should be checked and selected without lapse.	Medium	Sprint-1
Customer	Safety	USN-3	As a user,I should monitor the device is in the secured place which should cover wide area	Safety measures should be done to prevent disaster	High	Sprint-2
Customer	Examination of Natural anamoly	USN-4	As a user,I should analyse the depth of the occurrence of the phenomena	I should monitor the factors which causes disaster	High	Sprint-1
Customer	Battery Backup	USN-5	As a user,I want to check the battery to prevent from power loss	Aware to always keep battery backup .Sometimes it may help in any crucial situations.	Low	Sprint-3
Customer	Algorithm to be used	USN-6	As a user,I should be very conscious in selecting required algorithm	Algorithm provides a correct understanding about the model designed.	Medium	Sprint-4
Customer(Web user)	Internet Connectivity	USN-7	As a user,I should monitor the internet connection periodically	Strong internet connection is required in emergency situations.	High	Sprint-2
Customer(web User)	Social media	USN-8	As a user ,I will be active in social media sites to know more updates about specific diasaster	Active in social media sites to know updates	Medium	Sprint-4
Customer	Prediction and analysis of data	USN-9	As a user,I can ale to predict and visualize data	Using algorithms and some visualization	High	Sprint-3

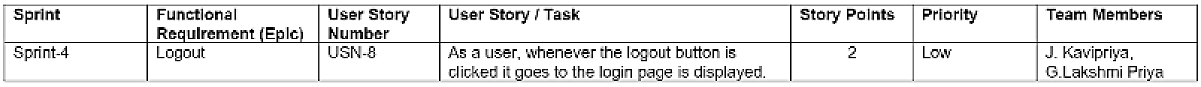
6. PROJECT PLANNING & SCHEDULING

A. SPRINT PLANNING & ESTIMATION:

Product Backlog, Sprint Schedule, and Estimation



Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-2	As a user, I will register in the website in order to identify the natural disaster based on the image given.	1	High	S. Jeyashree, J. Kavipriya,
Sprint-1	Confirmation	USN-3	As a user, a confirmation is sent that I am a registered user in the website	2	Low	K.Kaviyavarshini, G.Lakshmi Priya
Sprint-1	Login	USN-4	As a user, I can log into the application by entering email & password	1	High	S. Jeyashree, J. Kavipriya, K.Kaviyavarshini, G.Lakshmi Priya
Sprint-2	Home Page	USN-1	As a user, I can visit the home page whenever I enter the URL in the browser.	2	High	J. Kavipriya, K.Kaviyavarshini,
Sprint-2	Upload the image	USN-5	As a user, I will navigate to the page where I need to upload the images for natural disaster classification. The upload is either by using webcam or images from gallery.	2	Medium	S. Jeyashree, G.Lakshmi Priya
Sprint-3	Classification of Disaster	USN-6	As a user, Based on the image given as input the natural disaster is classified and the result is shown.	4	High	S. Jeyashree, K.Kaviyavarshini,



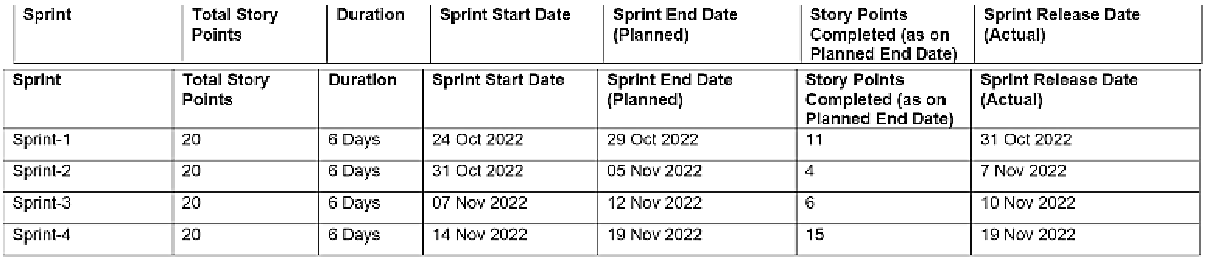
Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-4	Logout	USN-8	As a user, whenever the logout button is clicked it goes to the login page is displayed.	2	Low	J. Kavipriya, G.Lakshmi Priya

Velocity:
Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let’s calculate the team’s average velocity (AV) per iteration unit (story points per day)

$$AV = \frac{sprint\ duration}{velocity} = \frac{20}{10} = 2$$

Burndown Chart:
A burn down chart is a graphical representation of work left to do versus time. It is often used in agile software development methodologies such as Scrum. However, burn down charts can be applied to any project containing measurable progress over time.

Project Tracker, Velocity & Burndown Chart:



Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	11	31 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	4	7 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	6	10 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	15	19 Nov 2022

B. SPRINT DELIVERY SCHEDULE

Sprint 1:

The Sprint 1 of our project mainly involves preprocessing the data using data augmentation in Python. The Sprint Start Date is 24 Oct 2022 and the end date is 29 Oct 2022. The actual sprint release is 31 Oct 2022.

Sprint 2:

The Sprint 2 of our project mainly involves model building using CNN in Python. The Sprint Start Date is 31 Oct 2022 and the end date is 05 Nov 2022. The actual sprint release is 7 Nov 2022.

Sprint 3:

The Sprint 3 of our project mainly involves Front end development using Html and CSS and the integration of HTML page and Python using Flask. The Sprint Start Date is 07 Nov 2022 and the end date is 12 Nov 2022. The actual sprint release is 19 Nov 2022.

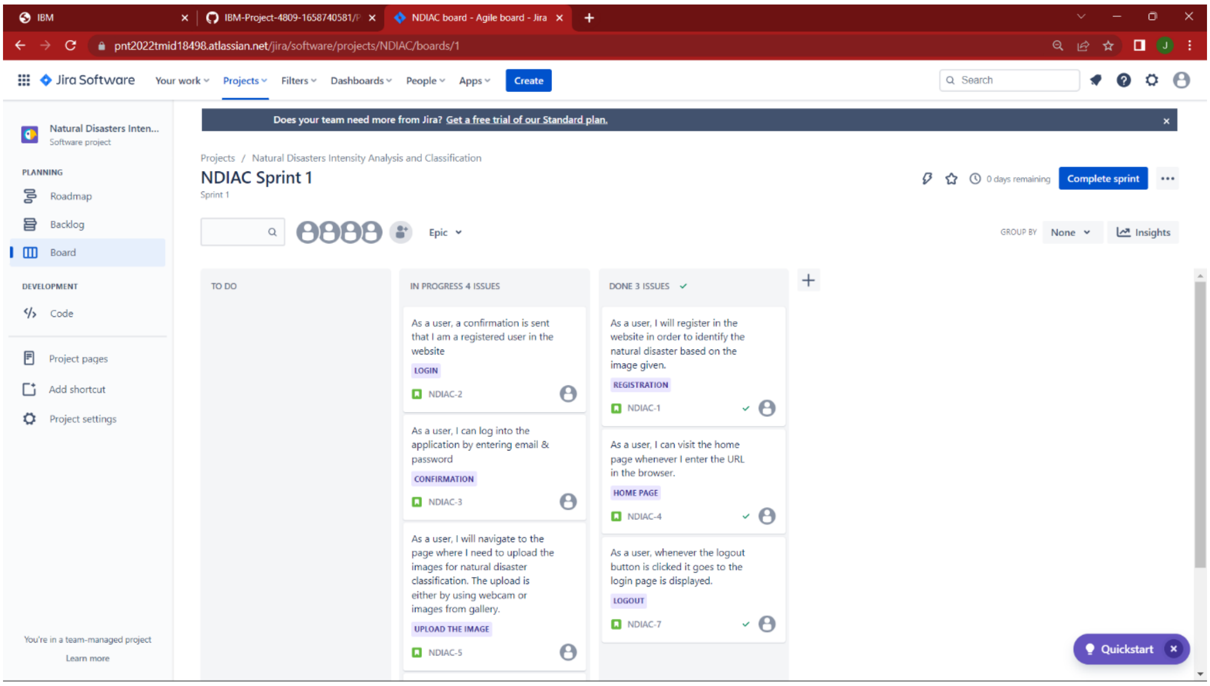
Sprint 4:

The Sprint 4 of our project mainly involves Deployment of Python cloud in IBM Watson Studio. The Sprint Start Date is 07 Nov 2022 and the end date is 12 Nov 2022. The actual sprint release is 19 Nov 2022.

C.REPORTS FROM JIRA

Sprint Delivery Plan for the project on Jira - On Progress

Sprint Board:



Does your team need more from Jira? Get a free trial of our Standard plan.

Projects / Natural Disasters Intensity Analysis and Classification

NDIAC Sprint 1

Sprint 1

TO DO

IN PROGRESS 4 ISSUES

As a user, a confirmation is sent that I am a registered user in the website

LOGIN

NDIAC-2

As a user, I can log into the application by entering email & password

CONFIRMATION

NDIAC-3

As a user, I will navigate to the page where I need to upload the images for natural disaster classification. The upload is either by using webcam or images from gallery.

UPLOAD THE IMAGE

NDIAC-5

DONE 3 ISSUES ✓

As a user, I will register in the website in order to identify the natural disaster based on the image given.

REGISTRATION

NDIAC-1 ✓

As a user, I can visit the home page whenever I enter the URL in the browser.

HOME PAGE

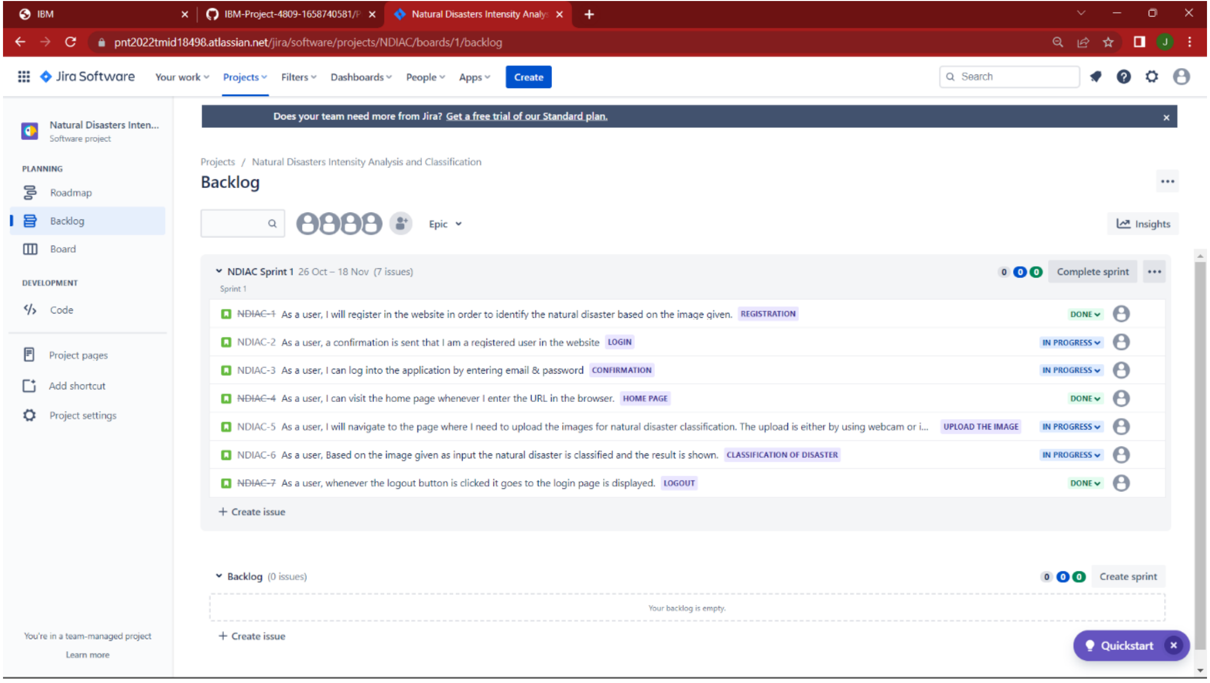
NDIAC-4 ✓

As a user, whenever the logout button is clicked it goes to the login page is displayed.

LOGOUT

NDIAC-7 ✓

Backlog:



Does your team need more from Jira? Get a free trial of our Standard plan.

Projects / Natural Disasters Intensity Analysis and Classification

Backlog

NDIAC Sprint 1 26 Oct – 18 Nov (7 issues) 0 0 0

Sprint 1

~~NDIAC-1~~	As a user, I will register in the website in order to identify the natural disaster based on the image given.	REGISTRATION	DONE
NDIAC-2	As a user, a confirmation is sent that I am a registered user in the website	LOGIN	IN PROGRESS
NDIAC-3	As a user, I can log into the application by entering email & password	CONFIRMATION	IN PROGRESS
~~NDIAC-4~~	As a user, I can visit the home page whenever I enter the URL in the browser.	HOME PAGE	DONE
NDIAC-5	As a user, I will navigate to the page where I need to upload the images for natural disaster classification. The upload is either by using webcam or i...	UPLOAD THE IMAGE	IN PROGRESS
NDIAC-6	As a user, Based on the image given as input the natural disaster is classified and the result is shown.	CLASSIFICATION OF DISASTER	IN PROGRESS
~~NDIAC-7~~	As a user, whenever the logout button is clicked it goes to the login page is displayed.	LOGOUT	DONE

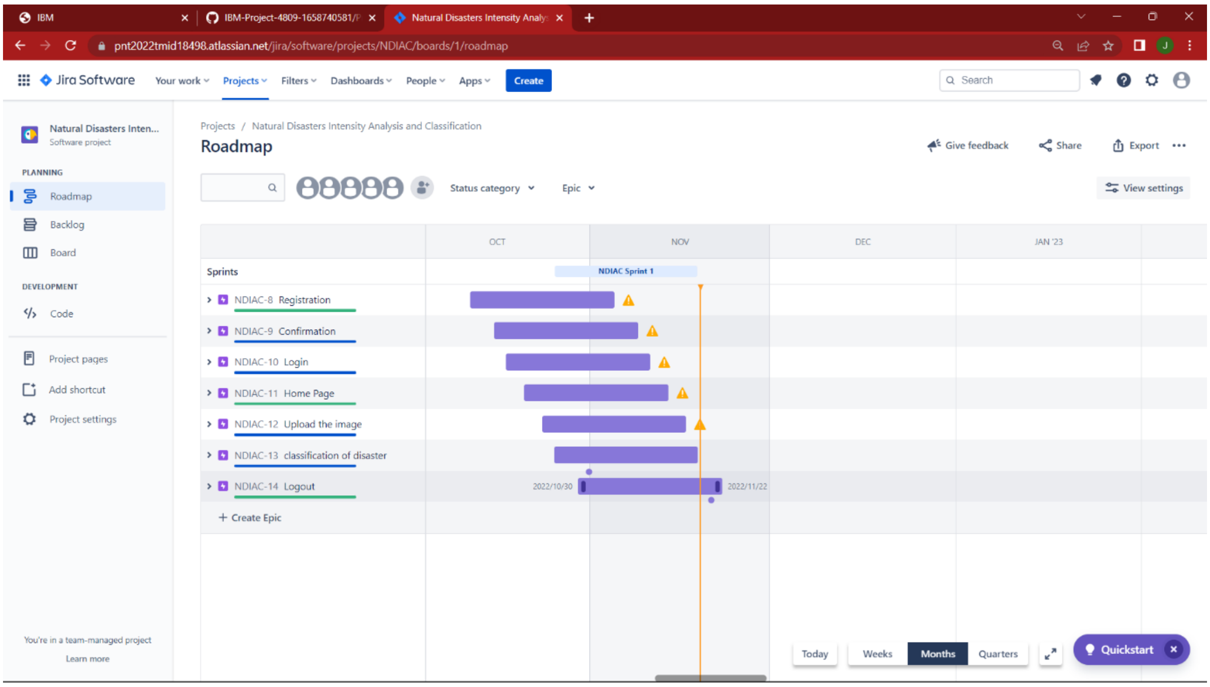
+ Create issue

Backlog (0 issues) 0 0 0

Your backlog is empty.

+ Create issue

Roadmap:



Projects / Natural Disasters Intensity Analysis and Classification

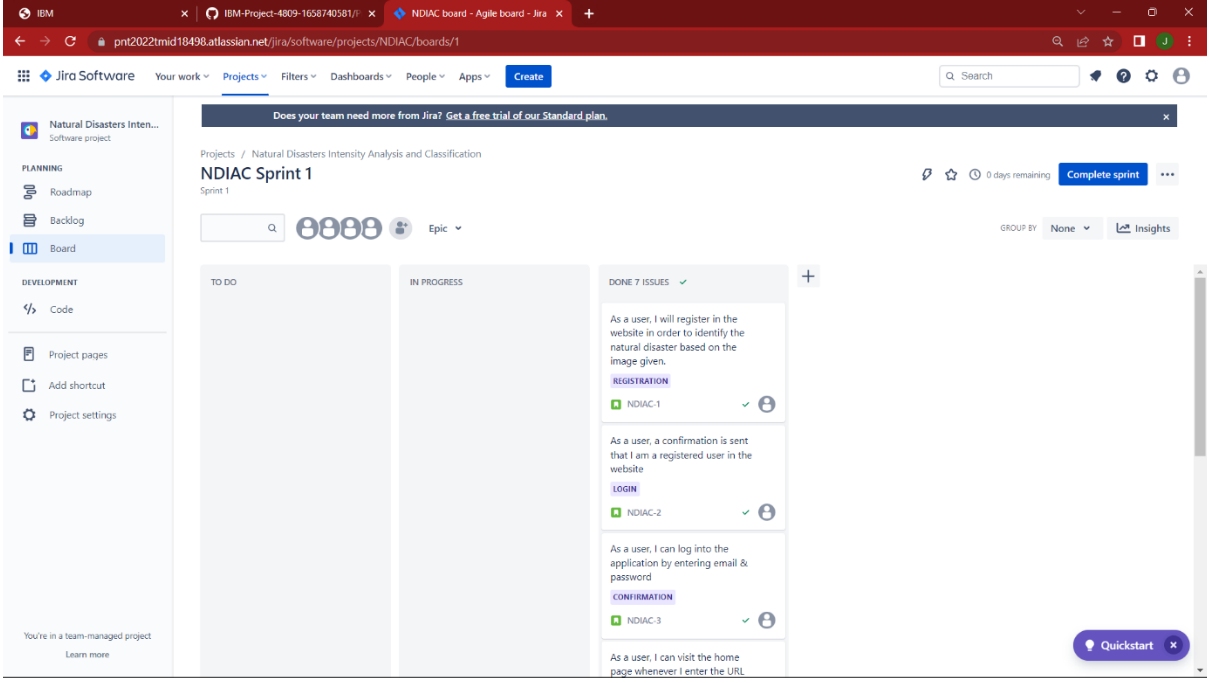
Roadmap

- NDIAC-8 Registration
- NDIAC-9 Confirmation
- NDIAC-10 Login
- NDIAC-11 Home Page
- NDIAC-12 Upload the image
- NDIAC-13 classification of disaster
- NDIAC-14 Logout

+ Create Epic

Sprint Delivery Plan on Jira – Completed

Sprint Board:



Does your team need more from Jira? Get a free trial of our Standard plan.

Projects / Natural Disasters Intensity Analysis and Classification

NDIAC Sprint 1

Sprint 1

TO DO

IN PROGRESS

DONE 7 ISSUES

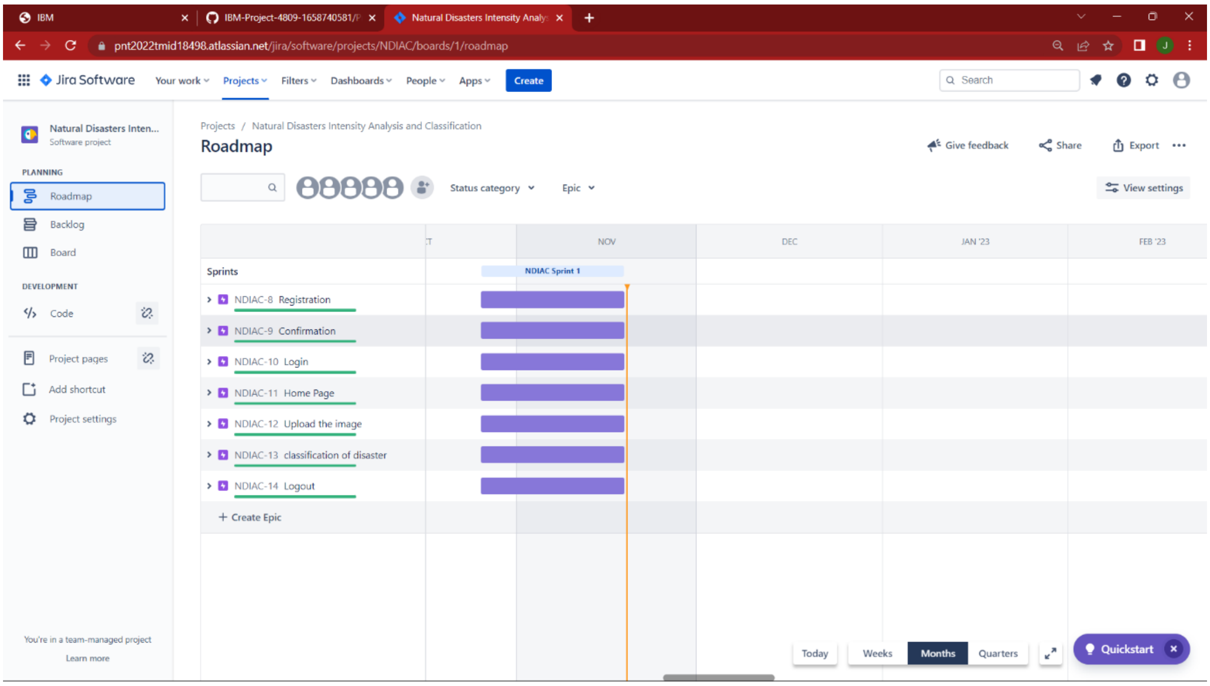
As a user, I will register in the website in order to identify the natural disaster based on the image given.
REGISTRATION
NDIAC-1

As a user, a confirmation is sent that I am a registered user in the website
LOGIN
NDIAC-2

As a user, I can log into the application by entering email & password
CONFIRMATION
NDIAC-3

As a user, I can visit the home page whenever I enter the URL

Roadmap:



Projects / Natural Disasters Intensity Analysis and Classification

Roadmap

	T	NOV	DEC	JAN '23	FEB '23
Sprints		NDIAC Sprint 1			
NDIAC-8 Registration					
NDIAC-9 Confirmation					
NDIAC-10 Login					
NDIAC-11 Home Page					
NDIAC-12 Upload the image					
NDIAC-13 classification of disaster					
NDIAC-14 Logout					
+ Create Epic					

7. CODING AND SOLUTIONING

a. DATA PREPROCESSING AND MODEL BUILDING

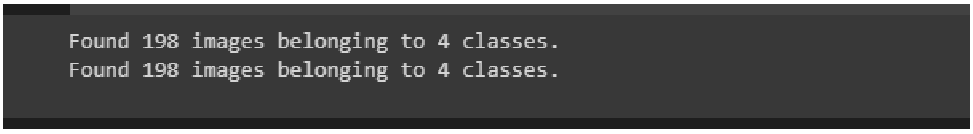
```
 import numpy as np#used for numerical analysis
import tensorflow #open source used for both ML and DL for computation
from tensorflow.keras.models import Sequential #it is a plain stack of layers
from tensorflow.keras import layers #A layer consists of a tensor-in tensor-out computation function
#Dense layer is the regular deeply connected neural network layer
from tensorflow.keras.layers import Dense,Flatten
#Faltten-used fot flattening the input or change the dimension
from tensorflow.keras.layers import Conv2D,MaxPooling2D #Convolutional layer
#MaxPooling2D-for downsampling the image
from keras.preprocessing.image import ImageDataGenerator
tensorflow.__version__
```

```
'2.8.0'
```

```
tensorflow.keras.__version__
```

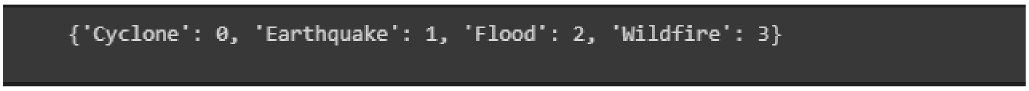
```
'2.8.0'
```

```
#setting parameter for Image Data agumentation to the training data
train_datagen = ImageDataGenerator(rescale=1./255,shear_range=0.2,zoom_range=0.2,horizontal_flip=True)
#Image Data agumentation to the testing data
test_datagen=ImageDataGenerator(rescale=1./255)
#performing data agumentation to train data
x_train = train_datagen.flow_from_directory(r'C:\Users\hp\Desktop\IBM\dataset\test_set',target_size=(64, 64),batch_size=5,
color_mode='rgb',class_mode='categorical')
#performing data agumentation to test data
x_test = test_datagen.flow_from_directory(r'C:\Users\hp\Desktop\IBM\dataset\test_set',target_size=(64, 64),batch_size=5,
color_mode='rgb',class_mode='categorical')
```



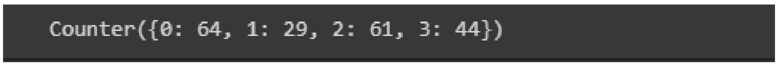
```
Found 198 images belonging to 4 classes.
Found 198 images belonging to 4 classes.
```

```
print(x_train.class_indices)#checking the number of classes
```



```
{'Cyclone': 0, 'Earthquake': 1, 'Flood': 2, 'Wildfire': 3}
```

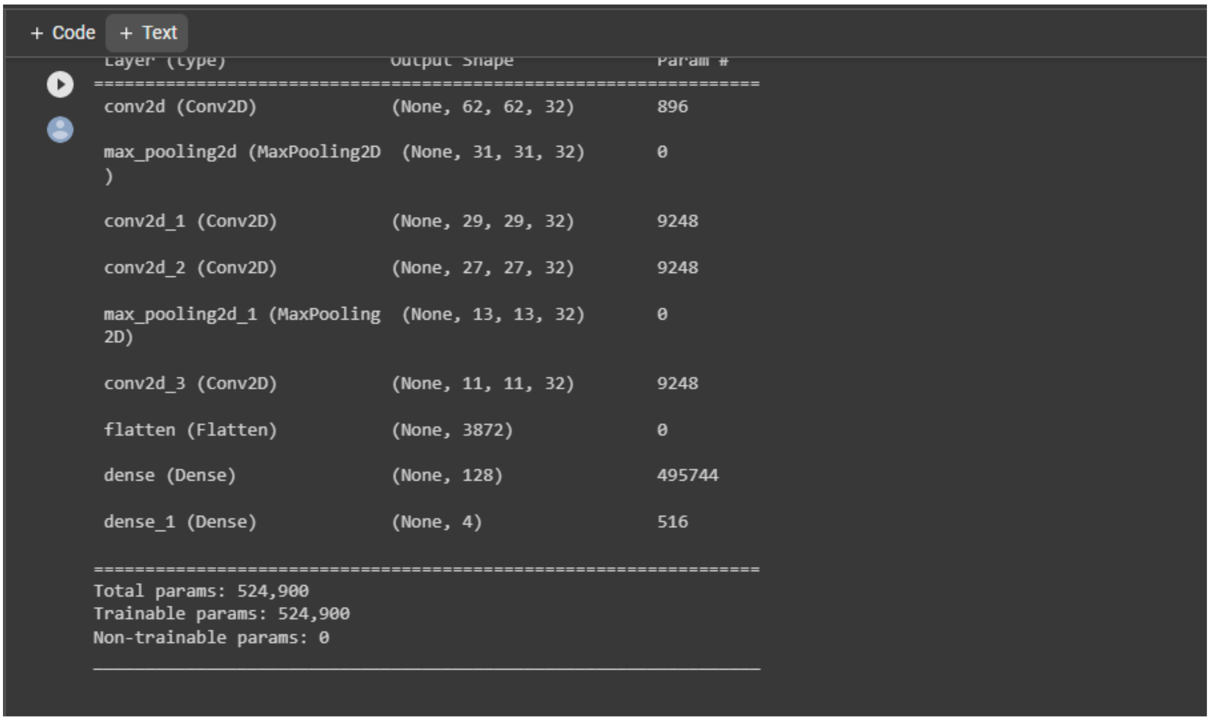
```
from collections import Counter as c
c(x_train .labels)
```



```
Counter({0: 64, 1: 29, 2: 61, 3: 44})
```

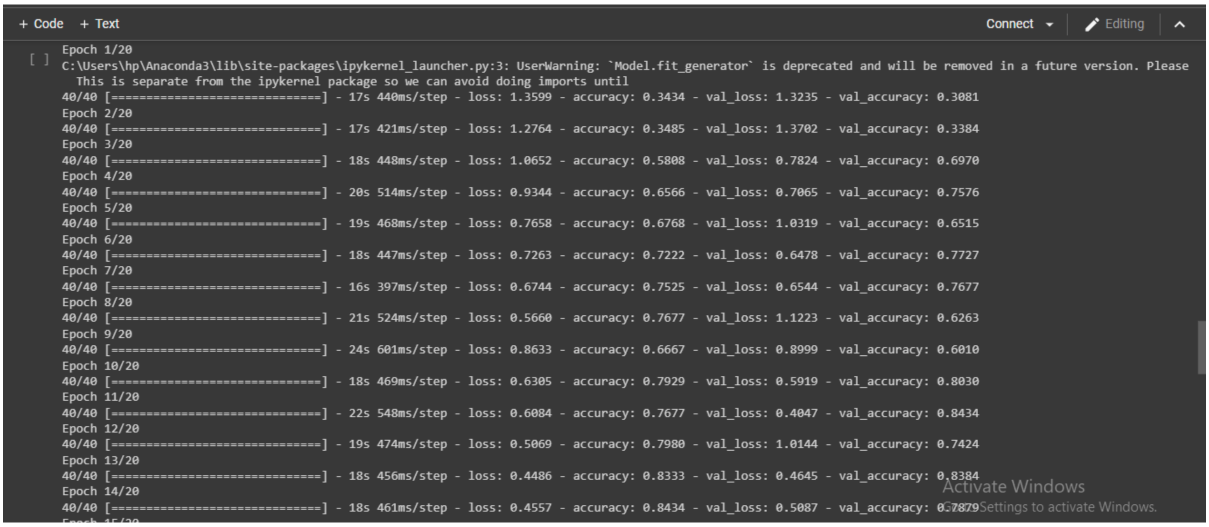
```
 Initializing the CNN
classifier = Sequential()
# First convolution layer and poolingo
classifier.add(Conv2D(32, (3, 3), input_shape=(64, 64, 3), activation='relu'))
classifier.add(MaxPooling2D(pool_size=(2, 2)))
classifier.add(Conv2D(32, (3, 3), input_shape=(64, 64, 3), activation='relu'))
# Second convolution layer and pooling
classifier.add(Conv2D(32, (3, 3), activation='relu'))
# input_shape is going to be the pooled feature maps from the previous
convolution layer
classifier.add(MaxPooling2D(pool_size=(2, 2)))
classifier.add(Conv2D(32, (3, 3), input_shape=(64, 64, 3), activation='relu'))
# Flattening the layers
classifier.add(Flatten())
# Adding a fully connected layer
classifier.add(Dense(units=128, activation='relu'))
classifier.add(Dense(units=4, activation='softmax')) # softmax for more than 2
```

```
classifier.summary()#summary of our model
```



```
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 62, 62, 32)        896

 max_pooling2d (MaxPooling2D  (None, 31, 31, 32)       0
 )

 conv2d_1 (Conv2D)           (None, 29, 29, 32)        9248

 conv2d_2 (Conv2D)           (None, 27, 27, 32)        9248

 max_pooling2d_1 (MaxPooling  (None, 13, 13, 32)       0
 2D)

 conv2d_3 (Conv2D)           (None, 11, 11, 32)        9248

 flatten (Flatten)           (None, 3872)              0

 dense (Dense)               (None, 128)               495744

 dense_1 (Dense)             (None, 4)                 516

=================================================================
Total params: 524,900
Trainable params: 524,900
Non-trainable params: 0
_________________________________________________________________
```

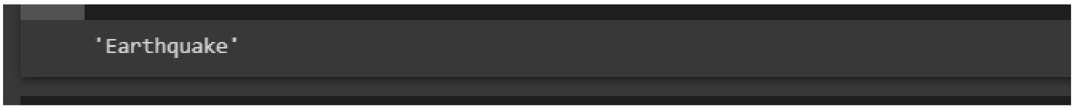
# Compiling the CNN
# categorical_crossentropy for more than 2
classifier.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])
classifier.fit_generator(
generator=x_train,steps_per_epoch = len(x_train),
epochs=20, validation_data=x_test,validation_steps = len(x_test))# No of images in test set



```
Epoch 1/20
C:\Users\hp\Anaconda3\lib\site-packages\ipykernel_launcher.py:3: UserWarning: `Model.fit_generator` is deprecated and will be removed in a future version. Please
  This is separate from the ipykernel package so we can avoid doing imports until
40/40 [==============================] - 17s 440ms/step - loss: 1.3599 - accuracy: 0.3434 - val_loss: 1.3235 - val_accuracy: 0.3081
Epoch 2/20
40/40 [==============================] - 17s 421ms/step - loss: 1.2764 - accuracy: 0.3485 - val_loss: 1.3702 - val_accuracy: 0.3384
Epoch 3/20
40/40 [==============================] - 18s 448ms/step - loss: 1.0652 - accuracy: 0.5808 - val_loss: 0.7824 - val_accuracy: 0.6970
Epoch 4/20
40/40 [==============================] - 20s 514ms/step - loss: 0.9344 - accuracy: 0.6566 - val_loss: 0.7065 - val_accuracy: 0.7576
Epoch 5/20
40/40 [==============================] - 19s 468ms/step - loss: 0.7658 - accuracy: 0.6768 - val_loss: 1.0319 - val_accuracy: 0.6515
Epoch 6/20
40/40 [==============================] - 18s 447ms/step - loss: 0.7263 - accuracy: 0.7222 - val_loss: 0.6478 - val_accuracy: 0.7727
Epoch 7/20
40/40 [==============================] - 16s 397ms/step - loss: 0.6744 - accuracy: 0.7525 - val_loss: 0.6544 - val_accuracy: 0.7677
Epoch 8/20
40/40 [==============================] - 21s 524ms/step - loss: 0.5660 - accuracy: 0.7677 - val_loss: 1.1223 - val_accuracy: 0.6263
Epoch 9/20
40/40 [==============================] - 24s 601ms/step - loss: 0.8633 - accuracy: 0.6667 - val_loss: 0.8999 - val_accuracy: 0.6010
Epoch 10/20
40/40 [==============================] - 18s 469ms/step - loss: 0.6305 - accuracy: 0.7929 - val_loss: 0.5919 - val_accuracy: 0.8030
Epoch 11/20
40/40 [==============================] - 22s 548ms/step - loss: 0.6084 - accuracy: 0.7677 - val_loss: 0.4047 - val_accuracy: 0.8434
Epoch 12/20
40/40 [==============================] - 19s 474ms/step - loss: 0.5069 - accuracy: 0.7980 - val_loss: 1.0144 - val_accuracy: 0.7424
Epoch 13/20
40/40 [==============================] - 18s 456ms/step - loss: 0.4486 - accuracy: 0.8333 - val_loss: 0.4645 - val_accuracy: 0.8384
Epoch 14/20
40/40 [==============================] - 18s 461ms/step - loss: 0.4557 - accuracy: 0.8434 - val_loss: 0.5087 - val_accuracy: 0.7879
```

classifier.save('disaster.h5')
model_json = classifier.to_json()
with open("model-bw.json", "w") as json_file:
json_file.write(model_json)
from tensorflow.keras.models import load_model

```
from keras.preprocessing import image
#model = load_model("disaster.h5") #loading the model for testing
img = image.load_img(r"C:\Users\hp\Downloads\e2.jpg",grayscale=False,
target_size= (64,64))#loading of the image
x = image.img_to_array(img)#image to array
x = np.expand_dims(x,axis = 0)#changing the shape
#pred = classifier.predict_classes(x)#predicting the classes
predict=classifier.predict(x)
classes_x=np.argmax(predict,axis=1)
classes_x

index=['Cyclone','Earthquake','Flood','Wildfire']
result=str(index[classes_x[0]])
result
```



```
'Earthquake'
```

b. FRONT END DEVELOPMENT AND INTEGRATION WITH FLASK

HOME.html

```
<html>
<script>


</script>


<style>
.header {        position: relative;
                        top:0;
                        margin:0px;
                        z-index: 1;
                        left: 0px;
                        right: 0px;
                        position: fixed;
                        background-color: #FCAD98 ;
                        color: white;
```

```
                              box-shadow: 0px 8px 2px grey;
                              overflow: hidden;
                              padding-left:20px;
                              font-family: 'Josefin Sans';
                              font-size: 2vw;
                              width: 100%;
                              height:8%;
                              text-align: center;
                        }
                        .topnav {
  overflow: hidden;
  background-color: #FCAD98;
}

.topnav-right a {
  float: left;
  color: black;
  text-align: center;
  padding: 14px 16px;
  text-decoration: none;
  font-size: 18px;
}

.topnav-right a:hover {
  background-color: #FCAD98;
  color: black;
}

.topnav-right a.active {
  background-color: #FCAD98;
  color: white;
}
```

```
.topnav-right {
  float: right;
  padding-right:100px;
}

body {
background-image: -webkit-linear-gradient(90deg, skyblue 0%, steelblue 100%);
  background-image: url("");
    background-size: cover;
  background-attachment: fixed;
  background-size: 100% 100%;
  background-color: ;
  background-repeat: no-repeat;
  background-size:cover;
  background-position: 0px 0px;
  }
  .button {
  background-color: #091425;
  border: none;
  color: white;
  padding: 15px 32px;
  text-align: center;
  text-decoration: none;
  display: inline-block;
  font-size: 12px;
  border-radius: 16px;
}
.button:hover {
  box-shadow: 0 12px 16px 0 rgba(0,0,0,0.24), 0 17px 50px 0 rgba(0,0,0,0.19);
}
form {border: 3px solid #f1f1f1; margin-left:400px;margin-right:400px;}

input[type=text], input[type=password] {
  width: 100%;
  padding: 12px 20px;
  display: inline-block;
  margin-bottom:18px;
  border: 1px solid #ccc;
  box-sizing: border-box;
}

button {
  background-color: #091425;
  color: white;
  padding: 14px 20px;
  margin-bottom:10px;
  border: none;
  cursor: pointer;
  width: 17%;
  border-radius:4px;
  font-family:Montserrat;
}

button:hover {
  opacity: 0.8;
}

.cancelbtn {
  width: auto;
  padding: 10px 18px;
  background-color: #f44336;
}

.imgcontainer {
  text-align: center;
  margin: 24px 0 12px 0;
```

```
}

img.avatar {
  width: 30%;
  border-radius: 50%;
}

.container {
  padding: 16px;
}

span.psw {
  float: right;
  padding-top: 16px;
}

/* Change styles for span and cancel button on extra small screens */
@media screen and (max-width: 300px) {
  span.psw {
     display: block;
     float: none;
  }
  .cancelbtn {
     width: 100%;
  }
}

.home{
	margin:80px;

  width: 84%;
  height: 500px;
  padding-top:10px;
  padding-left: 30px;

}
.login{
          margin:80px;
          box-sizing: content-box;
  width: 84%;
  height: 420px;
  padding: 30px;
  border: 10px solid blue;
}
.left,.right{
box-sizing: content-box;
height: 400px;
margin:20px;
border: 10px solid blue;
}

.mySlides {display: none;}
img {vertical-align: middle;}

/* Slideshow container */
.slideshow-container {
  max-width: 1000px;
  position: relative;
  margin: auto;
}

/* Caption text */
.text {
  color: #f2f2f2;
  font-size: 15px;
  padding: 8px 12px;
  position: absolute;
  bottom: 8px;
  width: 100%;
  text-align: center;
}
/* The dots/bullets/indicators */
.dot {
  height: 15px;
  width: 15px;
  margin: 0 2px;
  background-color: #bbb;
  border-radius: 50%;
  display: inline-block;
  transition: background-color 0.6s ease;
}

.active {
  background-color: #FCAD98;
}

/* Fading animation */
.fade {
  -webkit-animation-name: fade;
  -webkit-animation-duration: 1.5s;
  animation-name: fade;
  animation-duration: 1.5s;
}

@-webkit-keyframes fade {
  from {opacity: .4}
  to {opacity: 1}
}
```

```
@keyframes fade {
  from {opacity: .4}
  to {opacity: 1}
}

/* On smaller screens, decrease text size */
@media only screen and (max-width: 300px) {
  .text {font-size: 11px}
}
```

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

* {
  box-sizing: border-box;
}

body {
  min-height: 100vh;
  margin: 0;
  color: #fff;
  font-family: 'Poppins',sans-serif;
  display: flex;
  align-items: center;
  justify-content: center;
  background-color: #f5f5f5;
}

.container {
  max-width: 1376px;
  margin: auto;
  padding: 2rem 1.5rem;
}

.cards {
  display: flex;
  flex-wrap: wrap;
  align-items: center;
  justify-content: center;
}

.card {
  cursor: pointer;
  background-color: transparent;
  height: 300px;
  perspective: 1000px;
  margin: 1rem;
  align-items: center;
  justify-content: center;
}

.card h3 {
  border-bottom: 1px #fff solid;
  padding-bottom: 10px;
  margin-bottom: 10px;
  text-align: center;
  font-size: 1.6rem;
  word-spacing: 3px;
```

```
}

.card p{
  opacity: 0.75;
  font-size: 0.8rem;
  line-height: 1.4;
}

.card img {
  width: 360px;
  height: 300px;
  object-fit: cover;
  border-radius: 3px;
}

.card-inner {
  position: relative;
  width: 360px;
  height: 100%;
  transition: transform 0.9s;
  transform-style: preserve-3d;
}

.card:hover .card-inner {
  transform: rotateY(180deg);
}

.card-front,
.card-back {
  position: absolute;
  width: 360px;
  height: 100%;
  -webkit-backface-visibility: hidden;
  backface-visibility: hidden;
}

.card-back {
  background-color: #222;
  color: #fff;
  padding: 1.5rem;
  transform: rotateY(180deg);
}
.text-block {
  position: absolute;
  bottom: 20px;
  right: 20px;
  background-color: black;
  color: white;

  padding-left: 20px;
  padding-right: 20px;
}
</style>

<body>
<div class="header">
<div style="width:50%;float:left;font-size:2vw;text-align:left;color:black; padding-
top:1%;padding-left:5%;">AI based Natural disaster analysis</div>
  <div class="topnav-right"style="padding-top:0.5%;">

    <a class="active" href="/home">Home</a>
    <a href="/intro">Introduction</a>
    <a href="/upload">Open Web Cam</a>
  </div>
</div>
```

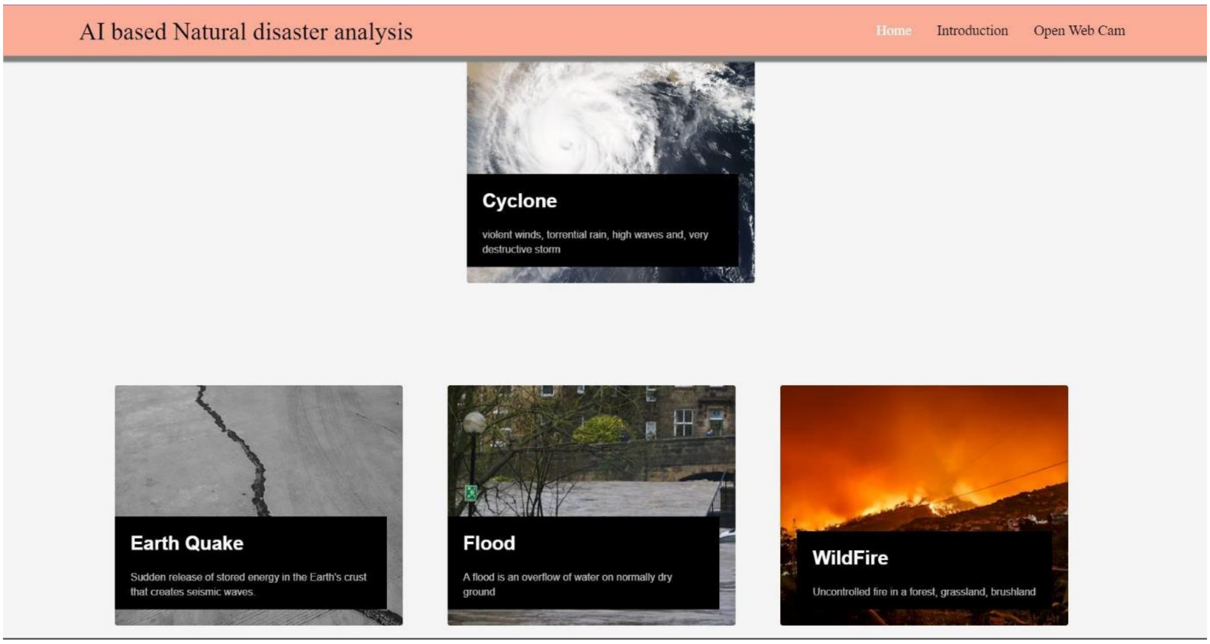
```
<div class="container">

    <div class="cards">

      <div class="card">
        <div class="card-inner">
          <div class="card-front">
            <img src="https://images.unsplash.com/photo-1454789476662-
53eb23ba5907?ixid=MXwxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHw%3D&ixlib=rb-
1.2.1&auto=format&fit=crop&w=689&q=80"
              alt="">
<div class="text-block">
  <h2>Cyclone</h2>
  <p>violent winds, torrential rain, high waves and, very destructive storm</p>
 </div>

          </div>
          <div class="card-back">
            <h3>Cyclone</h3>
            <p>The effects of tropical cyclones include heavy rain, strong wind, large storm
surges near landfall, and tornadoes. The destruction from a tropical cyclone, such as a
hurricane or tropical storm, depends mainly on its intensity, its size, and its location.</p>
          </div>
        </div>
      </div>


<div class="container">

    <div class="cards">

      <div class="card">
```

```
        <div class="card-inner">

          <div class="card-front">

            <img src="https://images.unsplash.com/photo-1603869311144-
66b03d340b32?ixid=MXwxMjA3fDB8MHxzZWFyY2h8M3x8ZWFydGhxdWFrZXxlbnwwfHww
fA%3D%3D&ixlib=rb-1.2.1&auto=format&fit=crop&w=500&q=60"

              alt="">

<div class="text-block">

  <h2>Earth Quake</h2>

  <p>Sudden release of stored energy in the Earth's crust that creates seismic waves.</p>

 </div>


          </div>

          <div class="card-back">

            <h3>Earth Quake</h3>

            <p>Earthquakes are usually caused when rock underground suddenly breaks
along a fault. This sudden release of energy causes the seismic waves that make the ground
shake. ... During the earthquake and afterward, the plates or blocks of rock start moving, and
they continue to move until they get stuck again.</p>

          </div>

        </div>

      </div>




<div class="container">


    <div class="cards">


      <div class="card">

        <div class="card-inner">

          <div class="card-front">

            <img src="https://images.unsplash.com/photo-1547683905-
f686c993aae5?ixid=MXwxMjA3fDB8MHxzZWFyY2h8MXx8Zmxvb2R8ZW58MHx8MHw%3D
&ixlib=rb-1.2.1&auto=format&fit=crop&w=500&q=60"

              alt="">
```

```
<div class="text-block">
  <h2>Flood</h2>
  <p>A flood is an overflow of water on normally dry ground</p>
 </div>

          </div>
          <div class="card-back">
            <h3>Flood</h3>
            <p>During heavy rain, the storm drains can become overwhelmed or plugged by debris and flood the roads and buildings nearby. Low spots, such as underpasses, underground parking garages, basements, and low water crossings can become death traps. Areas near rivers are at risk from floods.</p>
          </div>
        </div>
      </div>



<div class="container">

    <div class="cards">

      <div class="card">
        <div class="card-inner">
          <div class="card-front">
            <img src="https://images.unsplash.com/photo-1473260079709-83c808703435?ixid=MXwxMjA3fDB8MHxzZWFyY2h8NHx8d2lsZGZpcmV8ZW58MHx8MHw%3D&ixlib=rb-1.2.1&auto=format&fit=crop&w=500&q=60"
              alt="">
<div class="text-block">
  <h2>WildFire</h2>
  <p>Uncontrolled fire in a forest, grassland, brushland</p>
 </div>

          </div>
```

```
                    <div class="card-back">

                        <h3>Wildfire</h3>

                        <p>Wildfires can be caused by an accumulation of dead matter (leaves, twigs, and trees) that can create enough heat in some instances to spontaneously combust and ignite the surrounding area. Lightning strikes the earth over 100,000 times a day. 10 to 20% of these lightning strikes can cause fire.</p>

                    </div>

                </div>

            </div>


</body>

<html>
```

SCREENSHOT:



Cyclone

violent winds, torrential rain, high waves and, very destructive storm

Earth Quake

Sudden release of stored energy in the Earth's crust that creates seismic waves

Flood

A flood is an overflow of water on normally dry ground

WildFire

Uncontrolled fire in a forest, grassland, brushland

INTRO.html

```
<html>

<script>



</script>



<style>
```

```
.header {          position: relative;
                          top:0;
                          margin:0px;
                          z-index: 1;
                          left: 0px;
                          right: 0px;
                          position: fixed;
                          background-color: rgba(100, 100, 100, 0.5) ;
                          color: white;
                          box-shadow: 0px 8px 4px grey;
                          overflow: hidden;
                          padding-left:20px;
                          font-family: 'Josefin Sans';
                          font-size: 2vw;
                          width: 100%;
                          height:8%;
                          text-align: center;
                    }
                    .topnav {
  overflow: hidden;
  background-color: #FCAD98;
}

.topnav-right a {
  float: left;
  color: black;
  text-align: center;
  padding: 14px 16px;
  text-decoration: none;
  font-size: 18px;
}

.topnav-right a:hover {
  background-color: #FCAD98;
  color: black;
}

.topnav-right a.active {
  background-color: #FCAD98;
  color: white;
}

.topnav-right {
  float: right;
  padding-right:100px;
}

body {

  background-color: ;
  background-repeat: no-repeat;
  background-size:cover;

    background-size: cover;
  background-position: 0px 0px;
  }
  .button {
  background-color: #091425;
  border: none;
  color: white;
  padding: 15px 32px;
  text-align: center;
  text-decoration: none;
  display: inline-block;
  font-size: 12px;
  border-radius: 16px;
```

```
}
.button:hover {
  box-shadow: 0 12px 16px 0 rgba(0,0,0,0.24), 0 17px 50px 0 rgba(0,0,0,0.19);
}
form {border: 3px solid #f1f1f1; margin-left:400px;margin-right:400px;}

input[type=text], input[type=password] {
  width: 100%;
  padding: 12px 20px;
  display: inline-block;
  margin-bottom:18px;
  border: 1px solid #ccc;
  box-sizing: border-box;
}

button {
  background-color: #091425;
  color: white;
  padding: 14px 20px;
  margin-bottom:10px;
  border: none;
  cursor: pointer;
  width: 17%;
  border-radius:4px;
  font-family:Montserrat;
}

button:hover {
  opacity: 0.8;
}

.cancelbtn {
  width: auto;
  padding: 10px 18px;
  background-color: #f44336;
}

.imgcontainer {
  text-align: center;
  margin: 24px 0 12px 0;
}

img.avatar {
  width: 30%;
  border-radius: 50%;
}

.container {
  padding: 16px;
}

span.psw {
  float: right;
  padding-top: 16px;
}

/* Change styles for span and cancel button on extra small screens */
@media screen and (max-width: 300px) {
  span.psw {
     display: block;
     float: none;
  }
  .cancelbtn {
     width: 100%;
  }
}
```

```
.home{
        margin:80px;

  width: 84%;
  height: 500px;
  padding-top:10px;
  padding-left: 30px;

}
.login{
        margin:80px;
        box-sizing: content-box;
  width: 84%;
  height: 420px;
  padding: 30px;
  border: 10px solid blue;
}
.left,.right{
box-sizing: content-box;
height: 400px;
margin:20px;
border: 10px solid blue;
}

.mySlides {display: none;}
img {vertical-align: middle;}

/* Slideshow container */
.slideshow-container {
  max-width: 1000px;
  position: relative;
  margin: auto;
```

```
}

/* Caption text */
.text {
  color: #f2f2f2;
  font-size: 15px;
  padding: 8px 12px;
  position: absolute;
  bottom: 8px;
  width: 100%;
  text-align: center;
}
/* The dots/bullets/indicators */
.dot {
  height: 15px;
  width: 15px;
  margin: 0 2px;
  background-color: #bbb;
  border-radius: 50%;
  display: inline-block;
  transition: background-color 0.6s ease;
}

.active {
  background-color: #FCAD98;
}

/* Fading animation */
.fade {
  -webkit-animation-name: fade;
  -webkit-animation-duration: 1.5s;
  animation-name: fade;
  animation-duration: 1.5s;
```

```
}

@-webkit-keyframes fade {
  from {opacity: .4}
  to {opacity: 1}
}

@keyframes fade {
  from {opacity: .4}
  to {opacity: 1}
}

/* On smaller screens, decrease text size */
@media only screen and (max-width: 300px) {
  .text {font-size: 11px}
}



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

* {
  padding: 0;
  margin: 0;
}

body {
  height: 100vh;
  display: flex;
  flex-direction: column;
  justify-content: center;
  align-items: center;
}
```

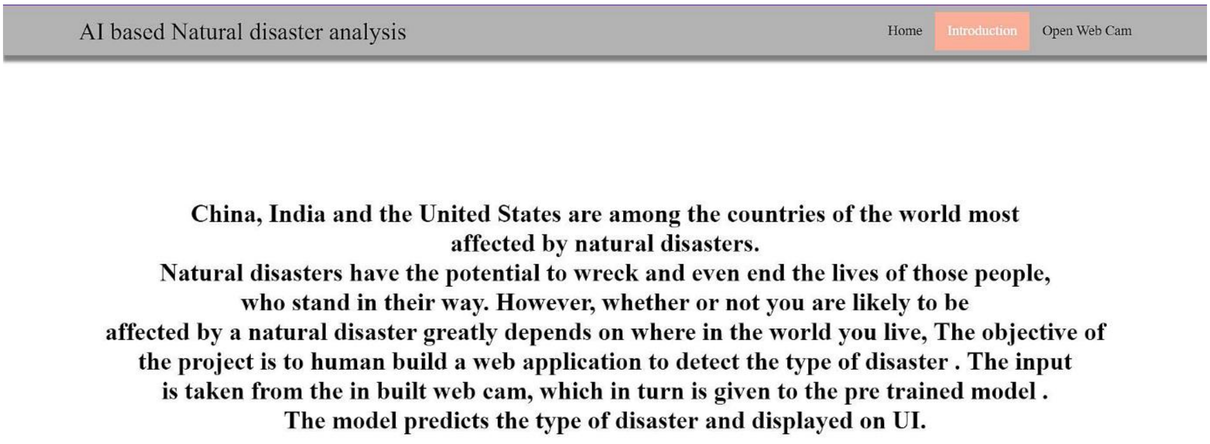
```
h1 {
  font-family: "Montserrat Medium";
  max-width: 90ch;
  text-align: center;
  transform: scale(0.94);
  animation: scale 3s forwards cubic-bezier(0.5, 1, 0.89, 1);
}
@keyframes scale {
  100% {
    transform: scale(1);
  }
}

span {
  display: inline-block;
  opacity: 0;
  filter: blur(4px);
}

span:nth-child(1) {
  animation: fade-in 1s 0.1s forwards cubic-bezier(0.11, 0, 0.5, 0);
}

span:nth-child(2) {
  animation: fade-in 0.8s 0.2s forwards cubic-bezier(0.11, 0, 0.5, 0);
}

span:nth-child(3) {
  animation: fade-in 0.8s 0.3s forwards cubic-bezier(0.11, 0, 0.5, 0);
}

span:nth-child(4) {
  animation: fade-in 0.8s 0.4s forwards cubic-bezier(0.11, 0, 0.5, 0);
}

span:nth-child(5) {
  animation: fade-in 0.8s 0.5s forwards cubic-bezier(0.11, 0, 0.5, 0);
}

span:nth-child(6) {
  animation: fade-in 0.8s 0.6s forwards cubic-bezier(0.11, 0, 0.5, 0);
}

span:nth-child(7) {
  animation: fade-in 0.8s 0.7s forwards cubic-bezier(0.11, 0, 0.5, 0);
}

span:nth-child(8) {
  animation: fade-in 0.8s 0.8s forwards cubic-bezier(0.11, 0, 0.5, 0);
}

span:nth-child(9) {
  animation: fade-in 0.8s 0.9s forwards cubic-bezier(0.11, 0, 0.5, 0);
}

span:nth-child(10) {
  animation: fade-in 0.8s 1s forwards cubic-bezier(0.11, 0, 0.5, 0);
}

span:nth-child(11) {
  animation: fade-in 0.8s 1.1s forwards cubic-bezier(0.11, 0, 0.5, 0);
}

span:nth-child(12) {
  animation: fade-in 0.8s 1.2s forwards cubic-bezier(0.11, 0, 0.5, 0);
```

```
}

span:nth-child(13) {
  animation: fade-in 0.8s 1.3s forwards cubic-bezier(0.11, 0, 0.5, 0);
}

span:nth-child(14) {
  animation: fade-in 0.8s 1.4s forwards cubic-bezier(0.11, 0, 0.5, 0);
}

span:nth-child(15) {
  animation: fade-in 0.8s 1.5s forwards cubic-bezier(0.11, 0, 0.5, 0);
}

span:nth-child(16) {
  animation: fade-in 0.8s 1.6s forwards cubic-bezier(0.11, 0, 0.5, 0);
}

span:nth-child(17) {
  animation: fade-in 0.8s 1.7s forwards cubic-bezier(0.11, 0, 0.5, 0);
}

span:nth-child(18) {
  animation: fade-in 0.8s 1.8s forwards cubic-bezier(0.11, 0, 0.5, 0);
}
span:nth-child(19) {
  animation: fade-in 0.8s 1.9s forwards cubic-bezier(0.11, 0, 0.5, 0);
}
span:nth-child(20) {
  animation: fade-in 0.8s 2.0s forwards cubic-bezier(0.11, 0, 0.5, 0);
}
span:nth-child(21) {
  animation: fade-in 0.8s 2.1s forwards cubic-bezier(0.11, 0, 0.5, 0);
```

```
}
span:nth-child(22) {
  animation: fade-in 0.8s 2.2s forwards cubic-bezier(0.11, 0, 0.5, 0);
}
span:nth-child(23) {
  animation: fade-in 0.8s 2.3s forwards cubic-bezier(0.11, 0, 0.5, 0);
}span:nth-child(24) {
  animation: fade-in 0.8s 2.4s forwards cubic-bezier(0.11, 0, 0.5, 0);
}span:nth-child(25) {
  animation: fade-in 0.8s 2.5s forwards cubic-bezier(0.11, 0, 0.5, 0);
}span:nth-child(26) {
  animation: fade-in 0.8s 2.6s forwards cubic-bezier(0.11, 0, 0.5, 0);
}span:nth-child(27) {
  animation: fade-in 0.8s 2.7s forwards cubic-bezier(0.11, 0, 0.5, 0);
}span:nth-child(28) {
  animation: fade-in 0.8s 2.8s forwards cubic-bezier(0.11, 0, 0.5, 0);
}
@keyframes fade-in {
  100% {
    opacity: 1;
    filter: blur(0);
  }
}

</style>

<body>
<h1>
  <span> China, India and the United States </span> <span> are among the countries of the
world most </span> <span> affected by natural disasters. </span > <span> Natural disasters
have the potential to wreck and even end the lives of those people,</span> <span>who
stand in their way.</span> <span>  However, whether or not you are likely to be  </span>
<span> affected by a natural disaster greatly depends</span > <span>  on where in the
world you live, </span>
```

```
 <span> The objective of </span> <span> the project is to</span> <span>human build a
</span > <span> web application </span> <span> to detect the </span> <span> type of
disaster .</span> <span> The input </span > <span> is taken from the in built web
cam,</span>

<span> which in turn </span> <span> is </span> <span> given to the  </span > <span>pre
trained model .</span> <span> The model predicts the  </span> <span> type of disaster
</span> <span> and displayed</span > <span> on UI.</span>




</h1>

<!–Brian Tracy–>


<div class="header">

<div style="width:50%;float:left;font-size:2vw;text-align:left;color:black; padding-
top:1%;padding-left:5%;">AI based Natural disaster analysis</div>

 <div class="topnav-right"style="padding-top:0.5%;">


   <a  href="/home">Home</a>

   <a class="active" href="/intro">Introduction</a>

   <a href="/upload">Open Web Cam</a>

 </div>

</div>

</body>


</html>
```

SCREENSHOT:



China, India and the United States are among the countries of the world most affected by natural disasters.
Natural disasters have the potential to wreck and even end the lives of those people, who stand in their way. However, whether or not you are likely to be affected by a natural disaster greatly depends on where in the world you live, The objective of the project is to human build a web application to detect the type of disaster . The input is taken from the in built web cam, which in turn is given to the pre trained model .
The model predicts the type of disaster and displayed on UI.

UPLOAD.html

```
<html lang="en">


<head>
  <title>Register</title>
  <link href="https://cdn.bootcss.com/bootstrap/4.0.0/css/bootstrap.min.css"
rel="stylesheet">
<style>
.header {        position: relative;
                      top:0;
                      margin:0px;
                      z-index: 1;
                      left: 0px;
                      right: 0px;
                      position: fixed;
                      background-color: #F36262 ;
                      color: white;
                      box-shadow: 0px 8px 4px grey;
                      overflow: hidden;
                      padding-left:20px;
                      font-family: 'Josefin Sans';
                      font-size: 2vw;
```

```
                                        width: 100%;
                                        height:8%;
                                        text-align: center;
                              }
                              .topnav {
  overflow: hidden;
  background-color: #FCAD98;
}

.topnav-right a {
  float: left;
  color: black;
  text-align: center;
  padding: 14px 16px;
  text-decoration: none;
  font-size: 18px;
}

.topnav-right a:hover {
  background-color: #FCAD98;
  color: black;
}

.topnav-right a.active {
  background-color: #FCAD98;
  color: white;
}

.topnav-right {
  float: right;
  padding-right:100px;
}
```

```
body {


  background-color: ;
  background-repeat: no-repeat;
  background-size:cover;
  background-image:
url("https://i.pinimg.com/originals/b2/1d/c6/b21dc69346915015bc4e19bd502f401b.gif");
    background-size: cover;
  background-position: 0px 0px;
  }
  .button {
  background-color: #091425;
  border: none;
  color: white;
  padding: 15px 32px;
  text-align: center;
  text-decoration: none;
  display: inline-block;
  font-size: 12px;
  border-radius: 16px;
}
.button:hover {
  box-shadow: 0 12px 16px 0 rgba(0,0,0,0.24), 0 17px 50px 0 rgba(0,0,0,0.19);
}
form {border: 3px solid #f1f1f1; margin-left:400px;margin-right:400px;}


input[type=text], input[type=password] {
  width: 100%;
  padding: 12px 20px;
  display: inline-block;
  margin-bottom:18px;
  border: 1px solid #ccc;
  box-sizing: border-box;
}

button {
  background-color: #091425;
  color: white;
  padding: 14px 20px;
  margin-bottom:10px;
  border: none;
  cursor: pointer;
  width: 17%;
  border-radius:4px;
  font-family:Montserrat;
}

button:hover {
  opacity: 0.8;
}

.cancelbtn {
  width: auto;
  padding: 10px 18px;
  background-color: #f44336;
}

.imgcontainer {
  text-align: center;
  margin: 24px 0 12px 0;
}

img.avatar {
  width: 30%;
  border-radius: 50%;
}
```

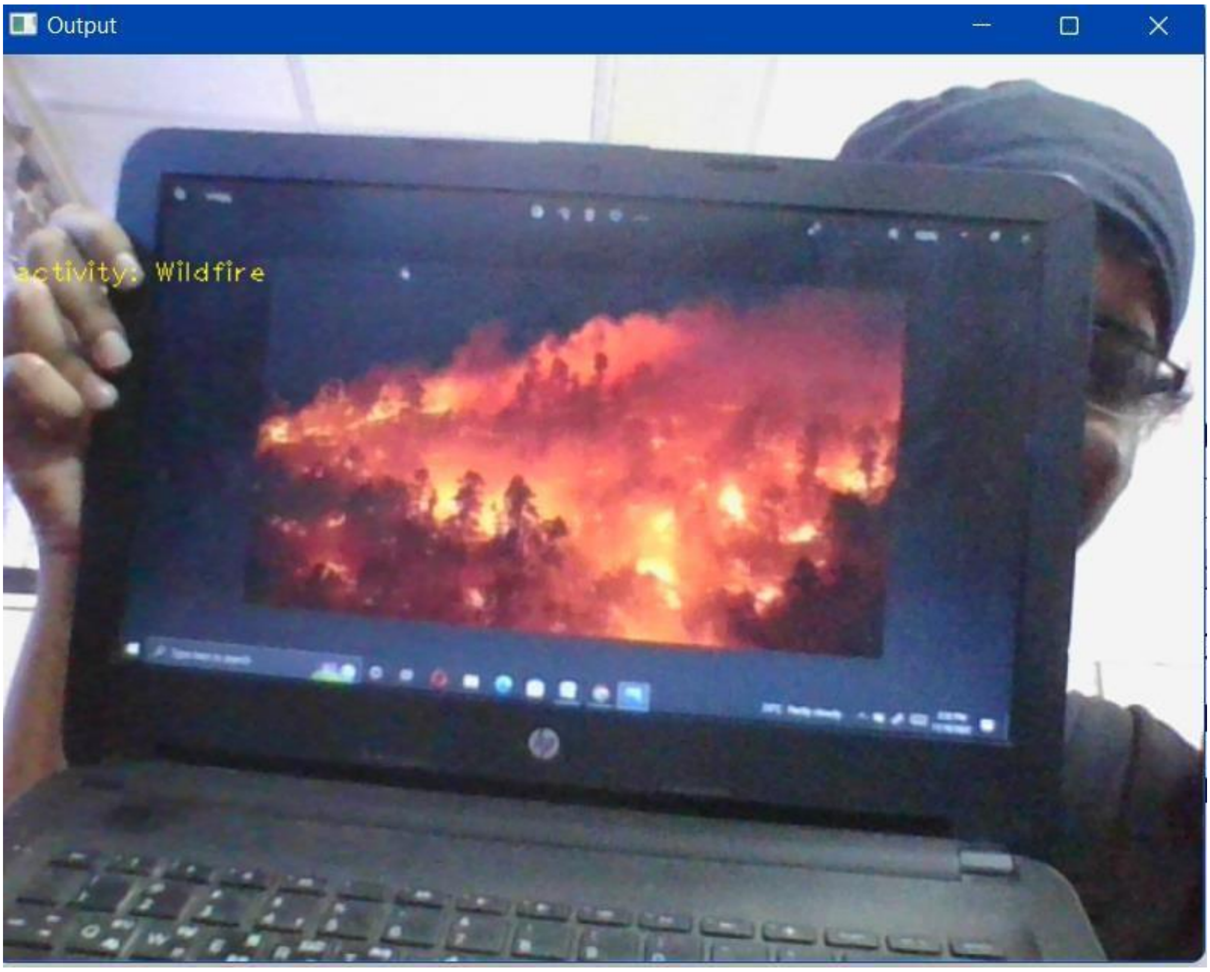
```
.container {
  padding: 16px;
}

span.psw {
  float: right;
  padding-top: 16px;
}

/* Change styles for span and cancel button on extra small screens */
@media screen and (max-width: 300px) {
  span.psw {
     display: block;
     float: none;
  }
  .cancelbtn {
     width: 100%;
  }
}

.home{
            margin:80px;

  width: 84%;
  height: 500px;
  padding-top:10px;
  padding-left: 30px;

}
.login{
            margin:80px;
            box-sizing: content-box;
  width: 84%;
  height: 420px;
  padding: 30px;
  border: 10px solid blue;
}
.left,.right{
box-sizing: content-box;
height: 400px;
margin:20px;
border: 10px solid blue;
}

.mySlides {display: none;}
img {vertical-align: middle;}

/* Slideshow container */
.slideshow-container {
  max-width: 1000px;
  position: relative;
  margin: auto;
}

/* Caption text */
.text {
  color: #f2f2f2;
  font-size: 15px;
  padding: 8px 12px;
  position: absolute;
  bottom: 8px;
  width: 100%;
  text-align: center;
}
/* The dots/bullets/indicators */
```

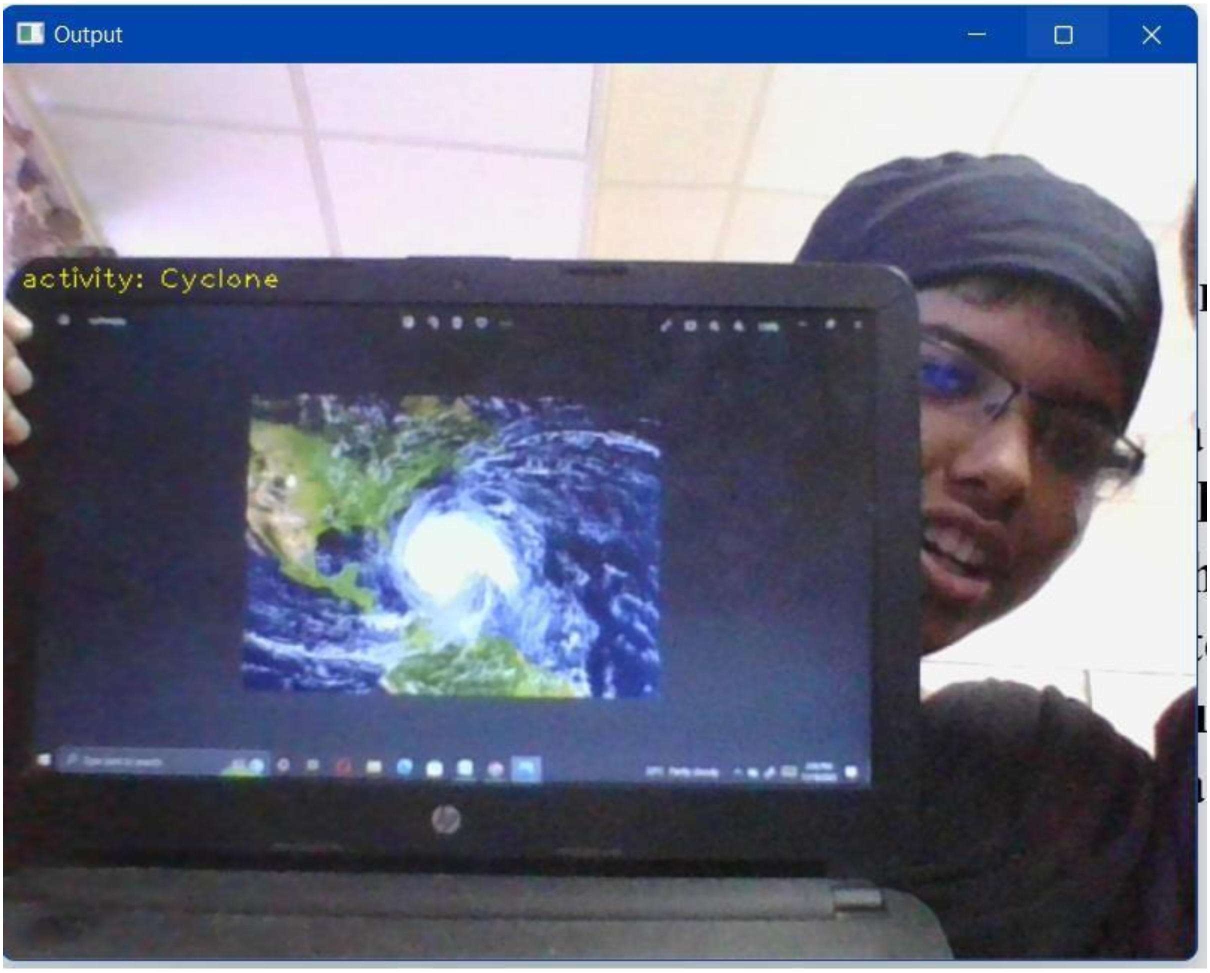
```
.dot {
  height: 15px;
  width: 15px;
  margin: 0 2px;
  background-color: #bbb;
  border-radius: 50%;
  display: inline-block;
  transition: background-color 0.6s ease;
}

.active {
  background-color: #FCAD98;
}

/* Fading animation */
.fade {
  -webkit-animation-name: fade;
  -webkit-animation-duration: 1.5s;
  animation-name: fade;
  animation-duration: 1.5s;
}

@-webkit-keyframes fade {
  from {opacity: .4}
  to {opacity: 1}
}

@keyframes fade {
  from {opacity: .4}
  to {opacity: 1}
}

/* On smaller screens, decrease text size */
```

```
@media only screen and (max-width: 300px) {
  .text {font-size: 11px}
}







.bar
{
margin: 0px;
padding:20px;
background-color:white;
opacity:0.6;
color:black;
font-family:'Roboto',sans-serif;
font-style: italic;
border-radius:20px;
font-size:25px;
}
a
{
color:grey;
float:right;
text-decoration:none;
font-style:normal;
padding-right:20px;
}
a:hover{
background-color:black;
color:white;
border-radius:15px;0
font-size:30px;
padding-left:10px;
}
body
{
   background-image: url("https://images.unsplash.com/photo-1532883130016-
f3d311140ba8?ixid=MXwxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHw%3D&ixlib=rb-
1.2.1&auto=format&fit=crop&w=1050&q=80");
   background-size: cover;
}
p
{
color:white;
font-style:italic;
font-size:30px;
}
</style>
</head>

<body>

<div class="header">
<div style="width:50%;float:left;font-size:2vw;text-align:left;color:black; padding-
top:1%;padding-left:5%;">AI based Natural disaster analysis</div>
  <div class="topnav-right"style="padding-top:0.5%;">


    <a  href="/home">Home</a>
    <a href="/intro">Introduction</a>
    <a class="active" href="/upload">Open Web Cam</a>
  </div>
</div>
```

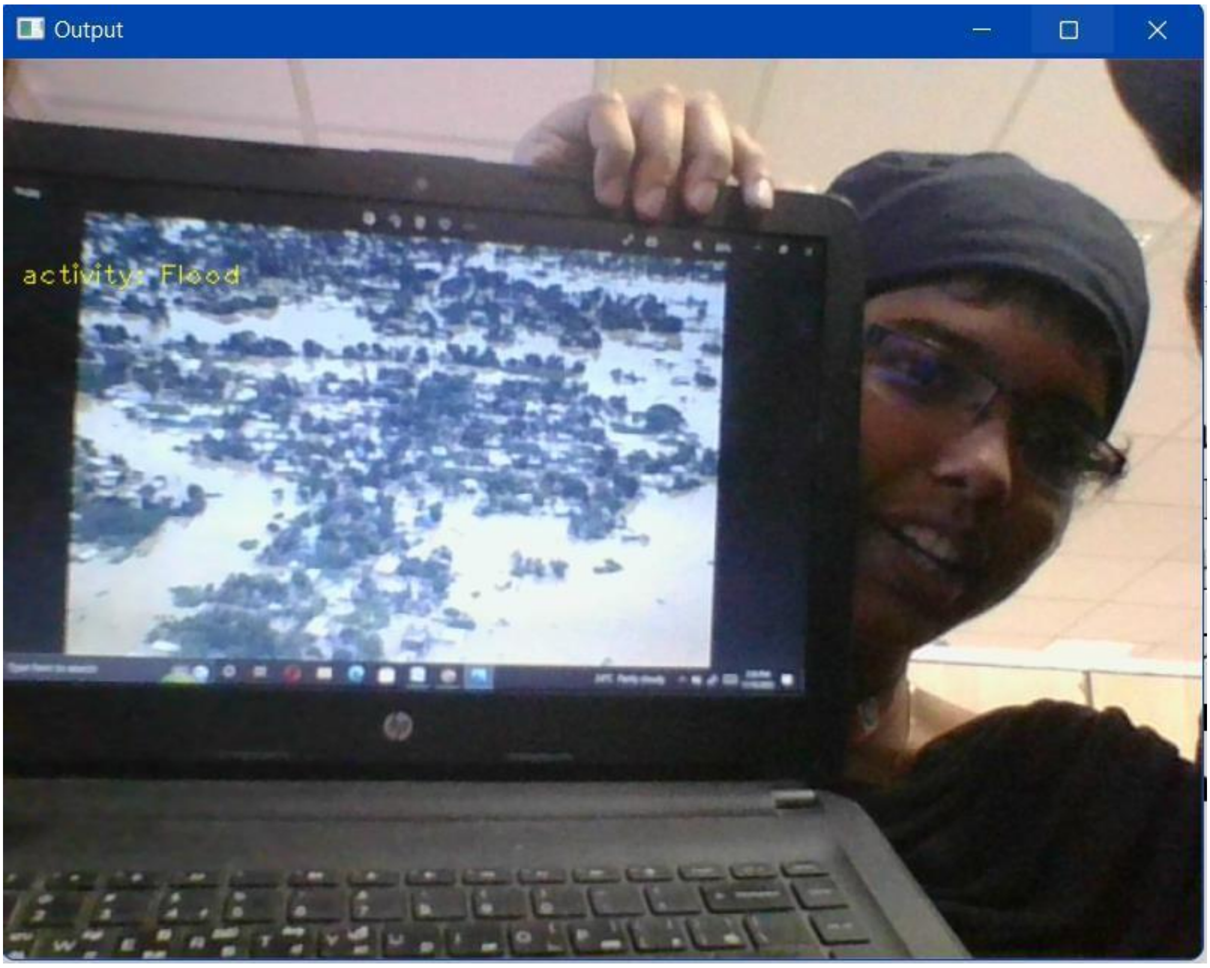
```
</body>
```

SCREENSHOT:





```
activity: Cyclone
```



FLASK FILE- app.py

```
from flask import Flask, render_template, request, redirect, url_for
import cv2
from tensorflow.keras.models import load_model
import numpy as np
from werkzeug.utils import secure_filename
app = Flask(__name__, template_folder="templates")
model = load_model('disaster.h5')
print("Loaded model from disk")
@app.route('/', methods=['GET'])
def index():
   return render_template('home.html')
@app.route('/home', methods=['GET'])
def home():
   return render_template('home.html')
```

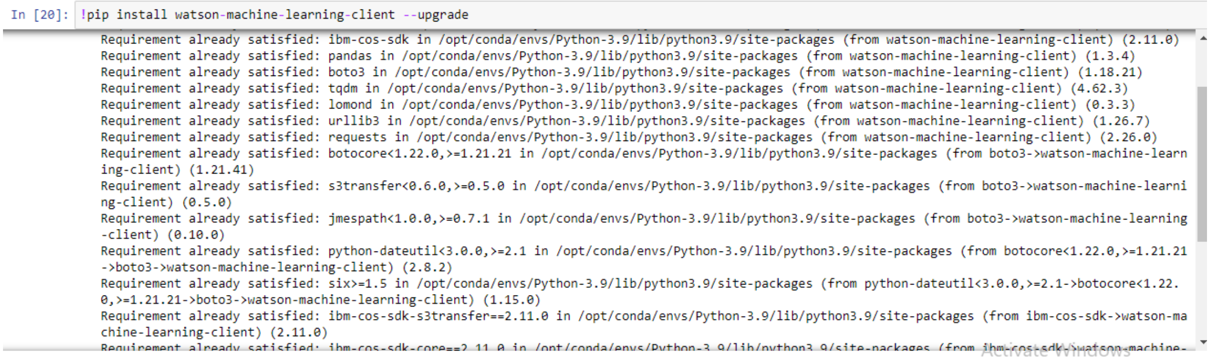
```
@app.route('/intro', methods=['GET'])
def about():
    return render_template('intro.html')



@app.route('/upload', methods=['GET', 'POST'])
def predict():
    print("[INFO] starting video stream...")
    vs = cv2.VideoCapture(0)
    (W, H) = (None, None)
    while True:
        (grabbed, frame) = vs.read()
        if not grabbed:
            break
        if W is None or H is None:
            (H, W) = frame.shape[:2]
        output = frame.copy()
        frame = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
        frame = cv2.resize(frame, (64, 64))
        x = np.expand_dims(frame, axis=0)
        result = np.argmax(model.predict(x), axis=-1)
        index = ['Cyclone', 'Earthquake', 'Flood', 'Wildfire']
        result = str(index[result[0]])

        cv2.putText(output, "activity: {}".format(result), (10, 120), cv2.FONT_HERSHEY_PLAIN,
                    1, (0, 255, 255), 1)
        cv2.imshow("Output", output)
        key = cv2.waitKey(1) & 0xFF
        if key == ord("q"):
            break
    print("[INFO] cleaning up...")
    vs.release()
    cv2.destroyAllWindows()
    return render_template("upload.html")


@app.route('/file', methods=['POST', 'GET'])
def video():
    if request.method == 'POST':
        uploaded_file = request.files['file1']
        if uploaded_file.filename != '':
            vid_name = str(uploaded_file.filename)
            print(vid_name + "Uploaded_Succesfully")
            uploaded_file.save(uploaded_file.filename)
            vs = cv2.VideoCapture(vid_name)
            if (vs.isOpened() == False):
                print("Error opening video stream or file")

            (W, H) = (None, None)
            while True:
                (grabbed, frame) = vs.read()
                if not grabbed:
                    break
                if W is None or H is None:
                    (H, W) = frame.shape[:2]
                output = frame.copy()
                frame = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
                frame = cv2.resize(frame, (64, 64))
                x = np.expand_dims(frame, axis=0)
                result = np.argmax(model.predict(x), axis=-1)
                index = ['Cyclone', 'Earthquake', 'Flood', 'Wildfire']
                result = str(index[result[0]])
                cv2.putText(output, "activity: {}".format(
                    result), (10, 120), cv2.FONT_HERSHEY_PLAIN, 1, (0, 255, 255), 1)
            cv2.imshow("Output", output)
            key = cv2.waitKey(1) & 0xFF
            if key == ord("q"):
              break
          print("[INFO] cleaning up...")
          vs.release()
          cv2.destroyAllWindows()
    return render_template("file.html")


  @app.route('/image', methods=['POST', 'GET'])
  def image():
    resulttext = ''
    if request.method == 'POST':
      uploaded_file = request.files['imgfile']
      if uploaded_file.filename != '':
        img_name = str(uploaded_file.filename)
        print(img_name + "Uploaded Succesfully")
        uploaded_file.save(uploaded_file.filename)
        from tensorflow.keras.models import load_model
        from keras.preprocessing import image
        model = load_model("disaster.h5")
        img = image.load_img(img_name, grayscale=False,
                   target_size=(64, 64))
        x = image.img_to_array(img)
        x = np.expand_dims(x, axis=0)
        pred = model.predict_classes(x)
        index = ['Cyclone', 'Earthquake', 'Flood', 'Wildfire']
        result = index[pred[0]]
        resulttext = result
    return render_template('image.html', result_text=resulttext)
```

```
if __name__ == '__main__':

   app.run(host='127.0.0.1', port=8000, debug=True)
```

c. DEPLOYMENT OF THE PROJECT ON IBM WATSON STUDIO

CODE IN IBM WATSON STUDIO:

```
 #Install Watson Machine Learning Client
!pip install watson-machine-learning-client –upgrade
```

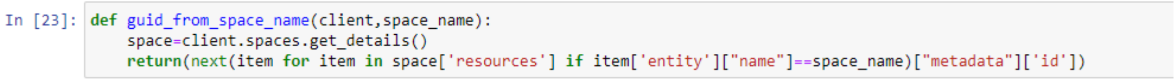


```
In [20]: !pip install watson-machine-learning-client --upgrade

         Requirement already satisfied: ibm-cos-sdk in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine-learning-client) (2.11.0)
         Requirement already satisfied: pandas in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine-learning-client) (1.3.4)
         Requirement already satisfied: boto3 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine-learning-client) (1.18.21)
         Requirement already satisfied: tqdm in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine-learning-client) (4.62.3)
         Requirement already satisfied: lomond in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine-learning-client) (0.3.3)
         Requirement already satisfied: urllib3 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine-learning-client) (1.26.7)
         Requirement already satisfied: requests in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine-learning-client) (2.26.0)
         Requirement already satisfied: botocore<1.22.0,>=1.21.21 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from boto3->watson-machine-learn
         ing-client) (1.21.41)
         Requirement already satisfied: s3transfer<0.6.0,>=0.5.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from boto3->watson-machine-learni
         ng-client) (0.5.0)
         Requirement already satisfied: jmespath<1.0.0,>=0.7.1 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from boto3->watson-machine-learning
         -client) (0.10.0)
         Requirement already satisfied: python-dateutil<3.0.0,>=2.1 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from botocore<1.22.0,>=1.21.21
         ->boto3->watson-machine-learning-client) (2.8.2)
         Requirement already satisfied: six>=1.5 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from python-dateutil<3.0.0,>=2.1->botocore<1.22.
         0,>=1.21.21->boto3->watson-machine-learning-client) (1.15.0)
         Requirement already satisfied: ibm-cos-sdk-s3transfer==2.11.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-cos-sdk->watson-ma
         chine-learning-client) (2.11.0)
         Requirement already satisfied: ibm-cos-sdk-core==2.11.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-cos-sdk->watson-machine-
```

```
from ibm_watson_machine_learning import APIClient
wml_credentials={
"url":"https://us-south.ml.cloud.ibm.com",
"apikey":"tLvqhgailCsiFTL8LX5pDMTvUdAvXCiEyMFCLaTT8_mv"
}
client=APIClient(wml_credentials)
```

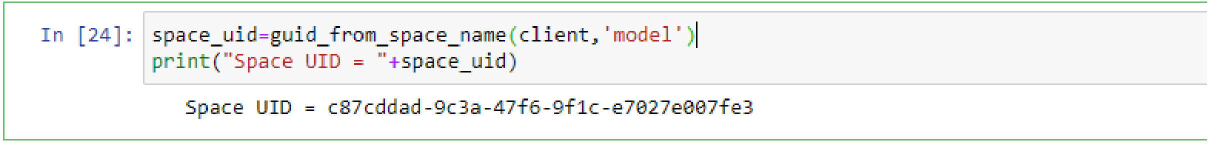
```
In [22]: from ibm_watson_machine_learning import APIClient
         wml_credentials={
                "url":"https://us-south.ml.cloud.ibm.com",
                "apikey":"tLvqhgailCsiFTL8LX5pDMTvUdAvXCiEyMFCLaTT8_mv"
         }
         client=APIClient(wml_credentials)
```

```
def guid_from_space_name(client,space_name):
space=client.spaces.get_details()
return(next(item for item in space['resources'] if
item['entity']["name"]==space_name)["metadata"]['id'])
```



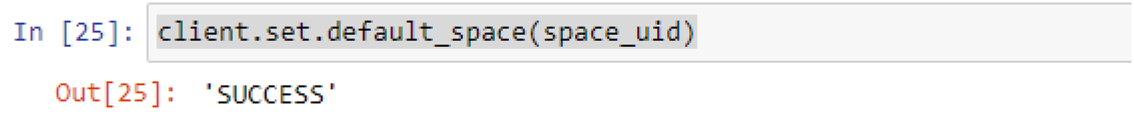
```
In [23]: def guid_from_space_name(client,space_name):
             space=client.spaces.get_details()
             return(next(item for item in space['resources'] if item['entity']["name"]==space_name)["metadata"]['id'])
```

```
space_uid=guid_from_space_name(client,'model')
print("Space UID = "+space_uid)
```



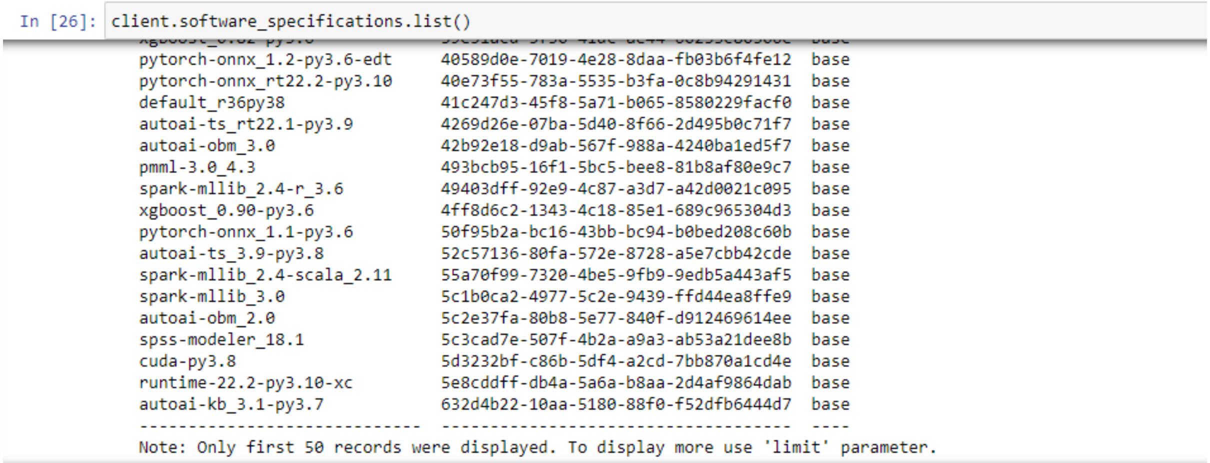
```
In [24]: space_uid=guid_from_space_name(client,'model')
         print("Space UID = "+space_uid)

            Space UID = c87cddad-9c3a-47f6-9f1c-e7027e007fe3
```

```
client.set.default_space(space_uid)
```



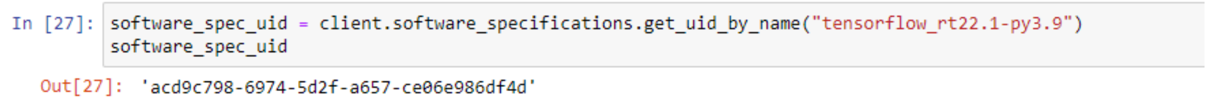
```
In [25]: client.set.default_space(space_uid)

   Out[25]: 'SUCCESS'
```

```
client.software_specifications.list()
```



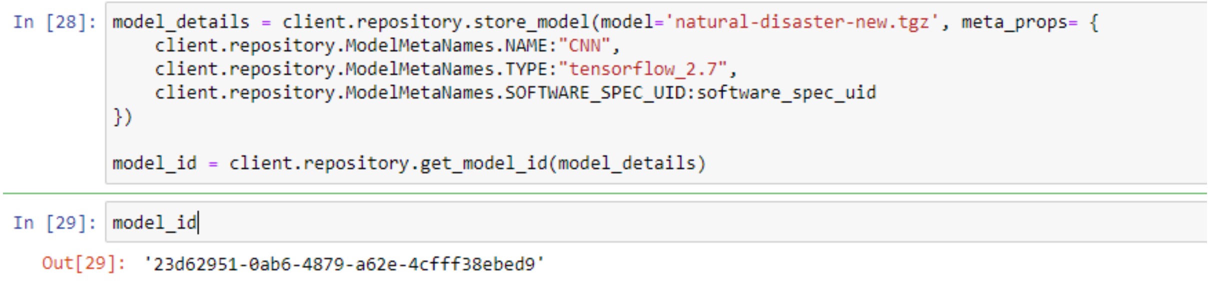
```
In [26]: client.software_specifications.list()
          pytorch-onnx_1.2-py3.6-edt      40589d0e-7019-4e28-8daa-fb03b6f4fe12  base
          pytorch-onnx_rt22.2-py3.10      40e73f55-783a-5535-b3fa-0c8b94291431  base
          default_r36py38                 41c247d3-45f8-5a71-b065-8580229facf0  base
          autoai-ts_rt22.1-py3.9          4269d26e-07ba-5d40-8f66-2d495b0c71f7  base
          autoai-obm_3.0                  42b92e18-d9ab-567f-988a-4240ba1ed5f7  base
          pmml-3.0_4.3                    493bcb95-16f1-5bc5-bee8-81b8af80e9c7  base
          spark-mllib_2.4-r_3.6           49403dff-92e9-4c87-a3d7-a42d0021c095  base
          xgboost_0.90-py3.6              4ff8d6c2-1343-4c18-85e1-689c965304d3  base
          pytorch-onnx_1.1-py3.6          50f95b2a-bc16-43bb-bc94-b0bed208c60b  base
          autoai-ts_3.9-py3.8             52c57136-80fa-572e-8728-a5e7cbb42cde  base
          spark-mllib_2.4-scala_2.11      55a70f99-7320-4be5-9fb9-9edb5a443af5  base
          spark-mllib_3.0                 5c1b0ca2-4977-5c2e-9439-ffd44ea8ffe9  base
          autoai-obm_2.0                  5c2e37fa-80b8-5e77-840f-d912469614ee  base
          spss-modeler_18.1               5c3cad7e-507f-4b2a-a9a3-ab53a21dee8b  base
          cuda-py3.8                      5d3232bf-c86b-5df4-a2cd-7bb870a1cd4e  base
          runtime-22.2-py3.10-xc          5e8cddff-db4a-5a6a-b8aa-2d4af9864dab  base
          autoai-kb_3.1-py3.7             632d4b22-10aa-5180-88f0-f52dfb6444d7  base
          ----------------------------    ------------------------------------  ----
          Note: Only first 50 records were displayed. To display more use 'limit' parameter.
```

software_spec_uid = client.software_specifications.get_uid_by_name("tensorflow_rt22.1-py3.9")
software_spec_uid



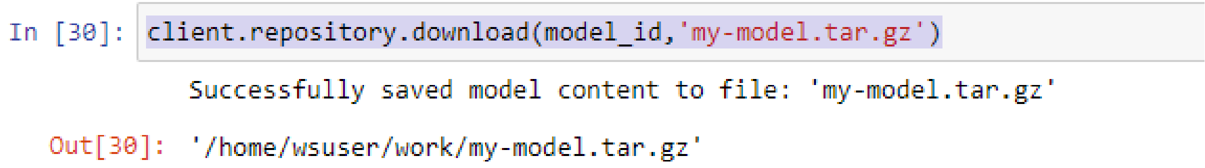
```
In [27]: software_spec_uid = client.software_specifications.get_uid_by_name("tensorflow_rt22.1-py3.9")
         software_spec_uid

Out[27]: 'acd9c798-6974-5d2f-a657-ce06e986df4d'
```

model_details = client.repository.store_model(model='natural-disaster-new.tgz',
meta_props= {
client.repository.ModelMetaNames.NAME:"CNN",
client.repository.ModelMetaNames.TYPE:"tensorflow_2.7",
client.repository.ModelMetaNames.SOFTWARE_SPEC_UID:software_spec_uid
})
model_id = client.repository.get_model_id(model_details)
model_id



```
In [28]: model_details = client.repository.store_model(model='natural-disaster-new.tgz', meta_props= {
             client.repository.ModelMetaNames.NAME:"CNN",
             client.repository.ModelMetaNames.TYPE:"tensorflow_2.7",
             client.repository.ModelMetaNames.SOFTWARE_SPEC_UID:software_spec_uid
         })

         model_id = client.repository.get_model_id(model_details)

In [29]: model_id

Out[29]: '23d62951-0ab6-4879-a62e-4cfff38ebed9'
```

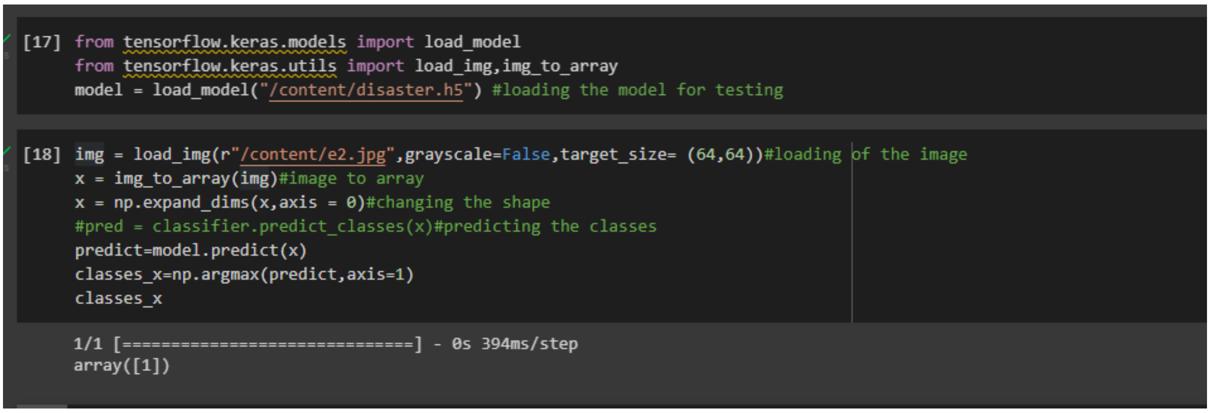
client.repository.download(model_id,'my-model.tar.gz')



```
In [30]: client.repository.download(model_id,'my-model.tar.gz')

         Successfully saved model content to file: 'my-model.tar.gz'

Out[30]: '/home/wsuser/work/my-model.tar.gz'
```

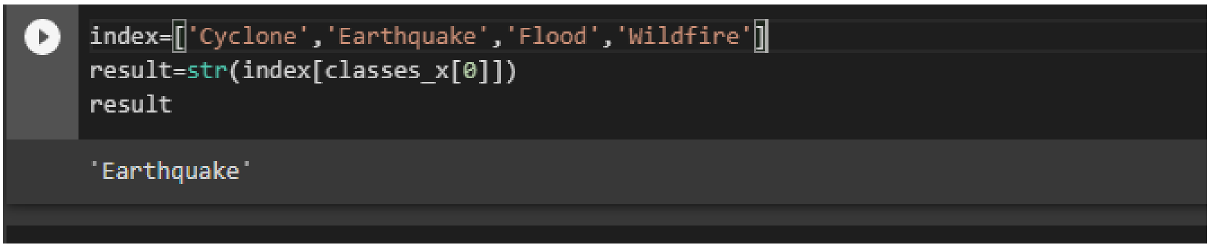
LOAD THE MODEL IN JUPYTER NOTEBOOK:
from tensorflow.keras.models import load_model
from tensorflow.keras.utils import load_img,img_to_array
model = load_model("/content/disaster.h5") #loading the model for testing
img = load_img(r"/content/e2.jpg",grayscale=False,target_size= (64,64))#loading of the image
x = img_to_array(img)#image to array
x = np.expand_dims(x,axis = 0)#changing the shape
predict=model.predict(x)

classes_x=np.argmax(predict,axis=1)
classes_x



```
[17] from tensorflow.keras.models import load_model
     from tensorflow.keras.utils import load_img,img_to_array
     model = load_model("/content/disaster.h5") #loading the model for testing

[18] img = load_img(r"/content/e2.jpg",grayscale=False,target_size= (64,64))#loading of the image
     x = img_to_array(img)#image to array
     x = np.expand_dims(x,axis = 0)#changing the shape
     #pred = classifier.predict_classes(x)#predicting the classes
     predict=model.predict(x)
     classes_x=np.argmax(predict,axis=1)
     classes_x

     1/1 [==============================] - 0s 394ms/step
     array([1])
```

index=['Cyclone','Earthquake','Flood','Wildfire']
result=str(index[classes_x[0]])
result



```
index=['Cyclone','Earthquake','Flood','Wildfire']
result=str(index[classes_x[0]])
result

'Earthquake'
```

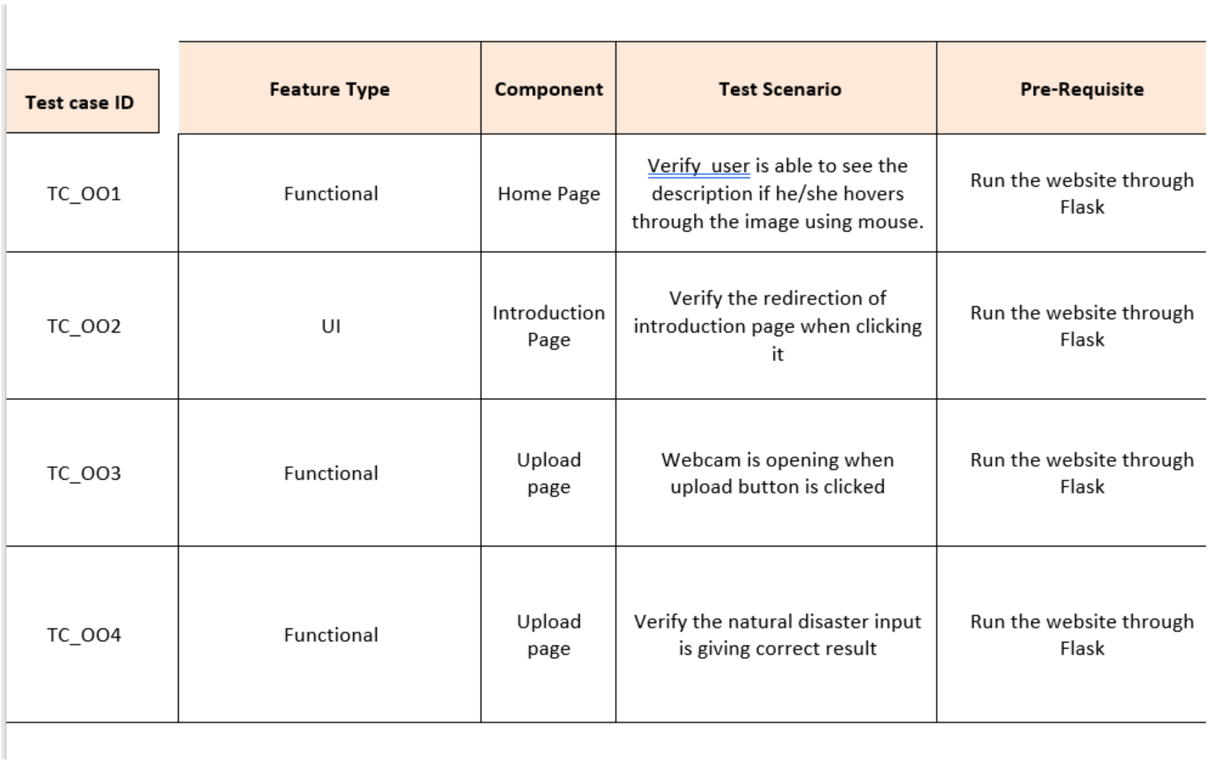
8. TESTING

A. TEST CASES

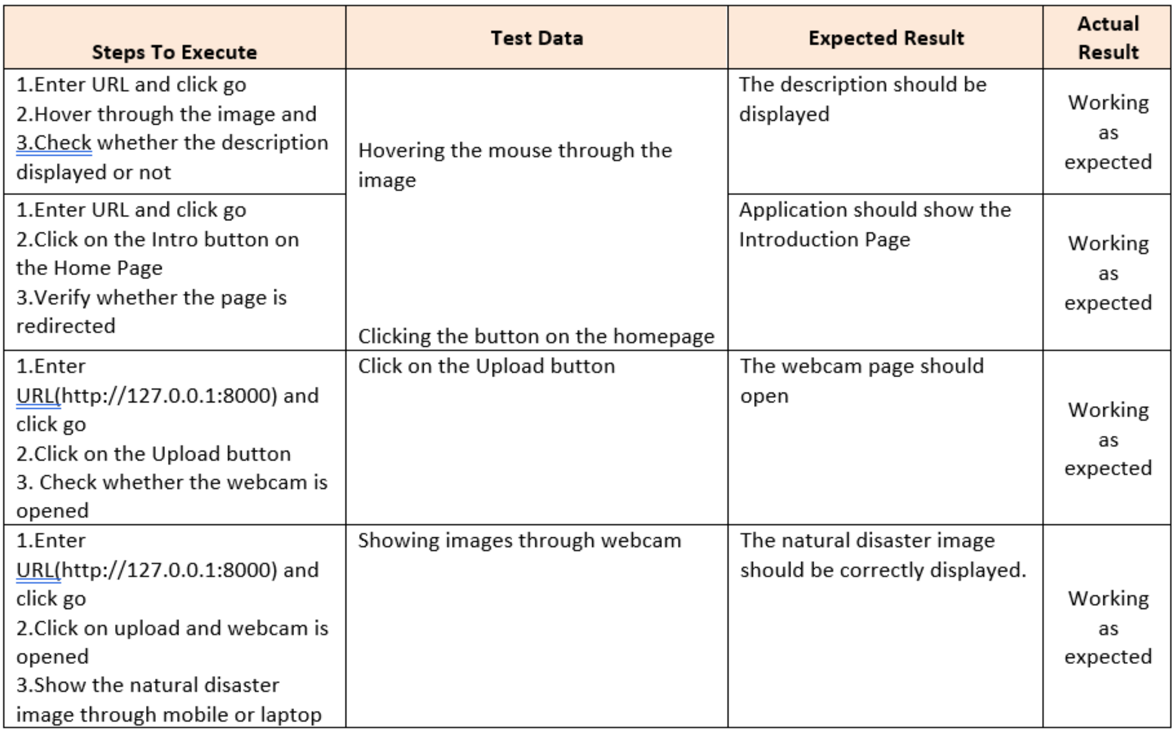
1. Verify user is able to see the description if he/she hovers through the image using mouse.
2. Verify the redirection of introduction page when clicking it
3. Webcam is opening when upload button is clicked
4. Verify the natural disaster input is giving correct result

B. USER ACCEPTANCE TESTING

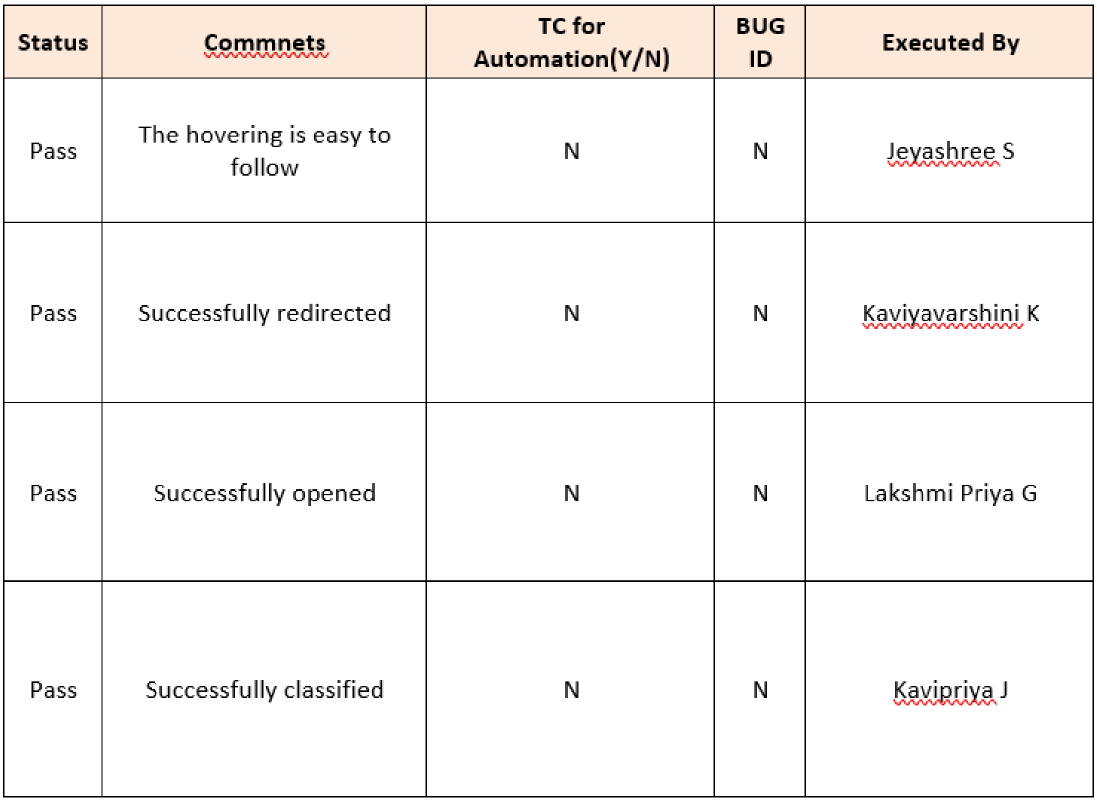
The purpose of this document is to briefly explain the test coverage and open issuesof the Natural Disasters Intensity Analysis and Classification using Artificial Intelligence project at the time of the release to User Acceptance Testing (UAT).



Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite
TC_OO1	Functional	Home Page	Verify user is able to see the description if he/she hovers through the image using mouse.	Run the website through Flask
TC_OO2	UI	Introduction Page	Verify the redirection of introduction page when clicking it	Run the website through Flask
TC_OO3	Functional	Upload page	Webcam is opening when upload button is clicked	Run the website through Flask
TC_OO4	Functional	Upload page	Verify the natural disaster input is giving correct result	Run the website through Flask



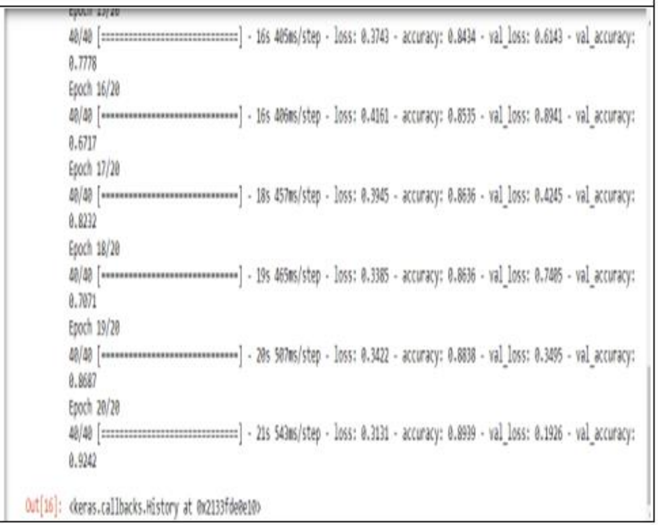
Steps To Execute	Test Data	Expected Result	Actual Result
1.Enter URL and click go 2.Hover through the image and 3.Check whether the description displayed or not	Hovering the mouse through the image	The description should be displayed	Working as expected
1.Enter URL and click go 2.Click on the Intro button on the Home Page 3.Verify whether the page is redirected	Clicking the button on the homepage	Application should show the Introduction Page	Working as expected
1.Enter URL(http://127.0.0.1:8000) and click go 2.Click on the Upload button 3. Check whether the webcam is opened	Click on the Upload button	The webcam page should open	Working as expected
1.Enter URL(http://127.0.0.1:8000) and click go 2.Click on upload and webcam is opened 3.Show the natural disaster image through mobile or laptop	Showing images through webcam	The natural disaster image should be correctly displayed.	Working as expected

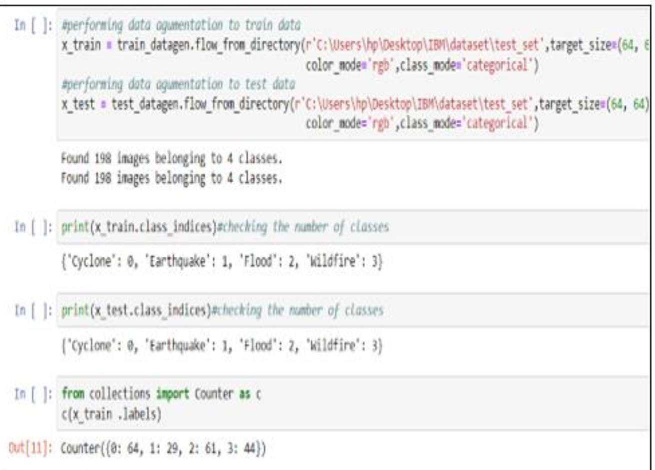


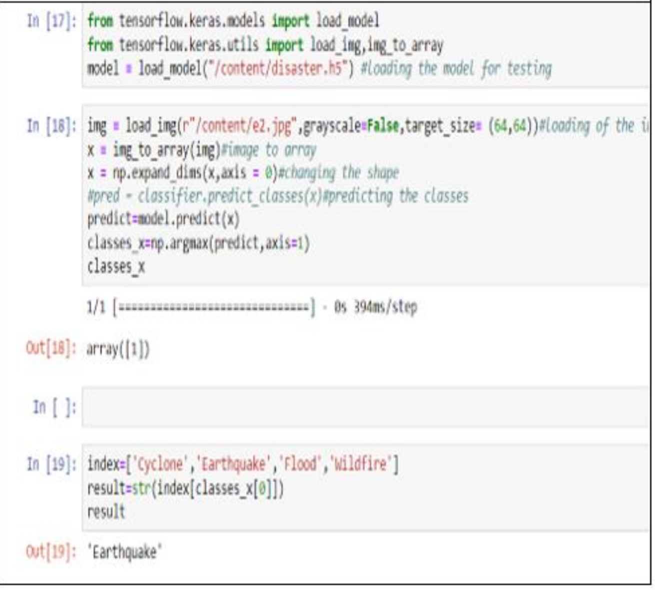
Status	Commnets	TC for Automation(Y/N)	BUG ID	Executed By
Pass	The hovering is easy to follow	N	N	Jeyashree S
Pass	Successfully redirected	N	N	Kaviyavarshini K
Pass	Successfully opened	N	N	Lakshmi Priya G
Pass	Successfully classified	N	N	Kavipriya J

9. TESTING

a. PERFORMANCE TESTING

S.No.	Parameter	Values	Screenshot
1.	Metrics	**Deep Learning Model – CNN** Accuracy through epoch	 Epoch 15/20 40/40 [==============================] - 16s 405ms/step - loss: 0.3743 - accuracy: 0.8434 - val_loss: 0.6143 - val_accuracy: 0.7778 Epoch 16/20 40/40 [==============================] - 16s 406ms/step - loss: 0.4161 - accuracy: 0.8535 - val_loss: 0.8941 - val_accuracy: 0.6717 Epoch 17/20 40/40 [==============================] - 18s 457ms/step - loss: 0.3945 - accuracy: 0.8636 - val_loss: 0.4245 - val_accuracy: 0.8232 Epoch 18/20 40/40 [==============================] - 19s 465ms/step - loss: 0.3385 - accuracy: 0.8636 - val_loss: 0.7405 - val_accuracy: 0.7071 Epoch 19/20 40/40 [==============================] - 20s 507ms/step - loss: 0.3422 - accuracy: 0.8838 - val_loss: 0.3495 - val_accuracy: 0.8687 Epoch 20/20 40/40 [==============================] - 21s 543ms/step - loss: 0.3131 - accuracy: 0.8939 - val_loss: 0.1926 - val_accuracy: 0.9242 Out[16]: <keras.callbacks.History at 0x2133fde0e10>

2.	Preprocess and Train the model	Using data augmentation and train test split method	
			In [ ]: #performing data agumentation to train data
			x_train = train_datagen.flow_from_directory(r'C:\Users\hp\Desktop\IBM\dataset\test_set',target_size=(64, 6
			color_mode='rgb',class_mode='categorical')
			#performing data agumentation to test data
			x_test = test_datagen.flow_from_directory(r'C:\Users\hp\Desktop\IBM\dataset\test_set',target_size=(64, 64)
			color_mode='rgb',class_mode='categorical')
			Found 198 images belonging to 4 classes.
			Found 198 images belonging to 4 classes.
			In [ ]: print(x_train.class_indices)#checking the number of classes
			{'Cyclone': 0, 'Earthquake': 1, 'Flood': 2, 'Wildfire': 3}
			In [ ]: print(x_test.class_indices)#checking the number of classes
			{'Cyclone': 0, 'Earthquake': 1, 'Flood': 2, 'Wildfire': 3}
			In [ ]: from collections import Counter as c
			c(x_train .labels)
			Out[11]: Counter({0: 64, 1: 29, 2: 61, 3: 44})

3.	Test the model	By loading the pretrained model and predicting the results	
			In [17]: from tensorflow.keras.models import load_model
			from tensorflow.keras.utils import load_img,img_to_array
			model = load_model("/content/disaster.h5") #loading the model for testing
			In [18]: img = load_img(r"/content/e2.jpg",grayscale=False,target_size= (64,64))#loading of the i
			x = img_to_array(img)#image to array
			x = np.expand_dims(x,axis = 0)#changing the shape
			#pred = classifier.predict_classes(x)#predicting the classes
			predict=model.predict(x)
			classes_x=np.argmax(predict,axis=1)
			classes_x
			1/1 [==============================] - 0s 394ms/step
			Out[18]: array([1])
			In [ ]:
			In [19]: index=['Cyclone','Earthquake','Flood','Wildfire']
			result=str(index[classes_x[0]])
			result
			Out[19]: 'Earthquake'

10. ADVANTAGES AND DISADVANTAGES

ADVANTAGES:

1. This will be useful for the NDRF (National Disaster Response Force) employees to easily classify and identify the natural disaster.
2. It will be useful for the students who need to know and study about the type of natural disaster.
3. It predicts results with the images given with high accuracy.
4. It is user friendly.
5. The website is interactive to the user.

DISADVANTAGES:

1. Sometimes the images with poor quality give wrong results.
2. The website is not available to everyone.
3. The poor camera quality can give poor results.
4. The customers find it difficult to search and find the website since it is not extended globally.

11. CONCLUSION

Thus the natural disaster has been classified when the natural disaster image is given as input through the webcam. The model was trained using Deep Learning Algorithms like Convolutional Neural Network and the model was trained using high accuracy. The website was created using HTML and CSS and it was integrated using Flask successfully. The results are shown live through the webcam.

12. FUTURE SCOPE

In future, the website can be extended to all the users across the world. The images with poor quality can also be reconstructed using OpenCV algorithms and the accuracy can be increased. The website can also be made more user friendly. The social media data can also be gathered and the location can be identified so that the help to the rescues can be done with the help of this webpage in future.

13. APPENDIX

SOURCE CODE – GITHUB LINK:

https://github.com/IBM-EPBL/IBM-Project-4809-1658740581

DEMO LINK – GITHUB: