1.INTRODUCTION

1.1 Project Overview

In recent decades, several forecasting methods have been proposed so as to aid in selecting from all optimal alternatives in the demand of emergency resources. Academic research in the field of emergency management has increasingly focused on artificial intelligence. However, more attention has been paid to attempts at simulating the human brain, with little focus on addressing intelligent information processing techniques based on machine learning, big data and smart devices. In this paper, a comprehensive literature review is presented in order to classify and interpret current research on demand forecasting methodologies and applications. A total of 1235 academic papers from 1980 to 2018 in the SpringerLink and Elsevier ScienceDirect databases are categorized as follows: time series analysis, case-based reasoning (CBR), mathematical models, information technology, literature reviews, and discussion and analysis. Application areas from business source premier include papers on the topics of emergency management, decision-making, decision relief, logistics, fuzzy sets and other topics.

1.2 Purpose

There has been an unsettling rise in the intensity and frequency of natural disasters due to climate change and anthropogenic activities. Artificial intelligence (AI) models have shown remarkable success and superiority to handle huge and nonlinear data owing to their higher accuracy and efficiency, making them perfect tools for disaster monitoring and management. Accordingly, natural disaster management (NDM) with the usage of AI models has received increasing attention in recent years, but there has been no systematic review so far. This paper presents a systematic review on how AI models are applied in different NDM stages based on 278 studies retrieved from Elsevier Science, Springer LINK and Web of Science. The review: enables increased visibility into various disaster types in different NDM stages from the methodological and content perspective, obtains many general results including the practicality and gaps of extant studies and provides several recommendations to develop innovative AI models and improve the quality of modeling.

2.LITERATURE SURVEY

PAPER.1

Author: Muhammad Aamir , Tariq Ali, Muhammad Irfan, Ahmad Shaf, Muhammad Zeeshan Azam, Adam Glowacz, Frantisek Brumercik, Witold Glowacz, Samar Alqhtani and Saifur Rahman

Year: 2021

Population growth is disrupting the environment, leading to global warming and increased crop failures. He also notes how some developing countries do not have enough money to damage infrastructure and drive people into depressing situations. None of the regions can be reached, and victims cannot be identified because of their exclusive regional geography. Technology is pursued. These UAE-derived data allow us to recognize the facial expressions of affected people, the depth of the situation, and their desires in published disaster scenarios. These are then processed and a neural community-based fully featured extraction strategy is performed to study this depth. The authors say that plant bug detection using deep learning strategies still faces various problems due to confusion and critical beauty imbalance problems. It is addressed by proposing deep convolutional neural communities. This approach involves a block of convolutional neural communities. The first block recognizes the herbal hazards that are occurring and the second defines the depth shape of the herbal hazards. Meanwhile, the second block further contains three layered mini-convolution blocks, consisting of a photo input layer and an absolute link layer. According to Block-I, the recognition process runs optimally in this phase. There are 3 small batches of 4 layers each. Block II takes the output of the primary block and adds depth to the various herbal disasters. The proposed multilayer deep convolutional neural community was simulated on the computer and the exclusive episodes were calculated. The dataset used by the author is collected from her m PyImage Search readers. The dataset is categorized into four classes: cyclones, earthquakes, floods, and wildfires. Then preprocessing is done to remove noise. To compare performance, the authors used a train-check validation scheme. The entire frame is computed based on the test dataset. Four data variants were created for valid and invalid values from the specified data record of the class. The authors followed the statistics of specificity (SP), sensitivity

(SE), precision (RR), and precision (PRE) as criteria for calculating performance. they used F1 ratings

PAPER.2

Author: Mohsin Raza, Muhammad Awais, Kamran Ali, Nauman Aslam, Vishnu Vardhan Paranthaman, Muhammad Imran, Farman Ali

Year: 2020

During natural hazards, disaster management plays a significant role wherever communication system plays a really important role like giving info to governmental institutes, non-governmental organizations (NGOs), initial responders and rescue staff to isolate people, execute facilitate and rescue service on the affected region. Sometimes the communication get failure, which build the rescue activities troublesome for the social activist, volunteers and organisation. To overcome this result on communication the authors bring the solutions on a way to make the communication effective to the disaster affected and communication outage areas by exploitation a manmade intelligence based mostly detection with social media platform.Here,they used the user central approach that type a adhoc network formation for restoring the mandatory communication of UE\MS once there's a failure in core communication. save for hand} the adhoc network formation they additionally projected novel cluster formation with single and multi-hop communication framework which supplies the communication link to the core network wherever the communication infrastructure collapse. The overall info that is created within the cluster is maximized exploitation hogged optimization. In addition to the other proposed theme they designed a intelligent system which is employed for labelling completely different clusters and their localities into affected and non affected area. the authors proposed a machine learning based mostly disaster severity analysis by exploitation social media platform that's wont to perform pre disaster vulnerability and localize fatal areas. The role of AI is to help government officials and emergency services to spot the threats and overall vulnerability. the acceptable outcomes of the urged machine learning schemes indicate their usage in conjunction with the suggested cluster methods to revive communications in disaster-affected areas and to categorize the disaster's impact for numerous locations in disaster-prone areas.

PAPER.3

Author: Rienna Oktarina, Senator Nur Bahagia, Lucia Diawati, Krisha S. Pribadi

Year: 2022

This technique, supported the applying of artificial neural networks, it's been accustomed predict earthquakes in Indonesia. This model may be anticipated to be ready to provide the types and numbers of things for help that the wedged folks can want throughout the emergency phase. during this study, supervised learning paradigm and backpropagation learning method are used to build ANN. The applied ANN spec is a multi-layer system with one somatic cell employed in each the input and output layer as a result. The impact of the earthquake, that consists of six factors, is that the study's output variable. whereas the study's input variables (predictors) were comprised of eight. As a guide for the accountable stakeholders' contingency coming up with Associate in Nursingd in lightweight of the consequences that the disaster caused by the earthquake can wear the wedged society, it's needed to conduct an assessment procedure on the particular losses and damages. For instance, working out what number destroyed homes there are will facilitate decide how many tents are required to accommodate the homeless. Another example would be to estimate the quantity of abraded folks when an earthquake disaster so as to manage the capability of the hospitals nearby, or to estimate the quantity of displaced folks in order to see the amount of help provides that should incline to minimise their suffering. therefore it's become crucial to form a model which will forecast the losses and damages caused by an earthquake disaster. The earthquake parameters that were instantly known whereas the earthquake was occurring were utilized as input variables during this investigation. The correctness of the sort of input variables can have an effect on the efficaciousness or length of the coaching process, so deciding the input variables is crucial once developing ANN. The created model may be accustomed estimate the number of earthquake-related building damages and fatalities. Thus, the result quickens the implementation of the provision evaluation, making certain prompt distribution of aid throughout the emergency response section to reduce the suffering of the wedged individuals. In distinction to the RADIUS programme, the ANN model is powerful and needs less input file despite having a better degree of prognostic capability.

PAPER.4

Author: Abu Reza Md Towfiqul Islam, Swapan Talukdar , Susanta Mahato, Sonali Kundu, Kutub Uddin Eibek, Quoc Bao Pham, Alban Kuriqi, Nguyen Thi Th

Year: 2021

Flooding is one of nature's most devastating disasters, as it causes enormous damage to land, buildings and casualties. Due to the dynamic and complex nature of flash floods, it is difficult to predict which areas are prone to flash floods. Thus, advanced machine learning models can be used to identify flash flood prone sites early to manage flood disasters. In this work, we apply and evaluate two new hybrid ensemble models: artificial neural networks (ANN), random forests (RF), and dagging and random subspace (RS) combined with support vector machines (SVM), The other three states represent state-of-the-art machine learning models that model the flood vulnerability map of the Teesta River Basin in northern Bangladesh. Application of these models includes 12 flood impact factors, including 413 current and historical flood points transferred to the GIS environment. Information gain ratios, a multicollinearity diagnostic test, were used to determine the relationship between events and flood impact factors. For validation and comparison of these models, measures such as Freidman, Wilcox on signed-rank test, t-pair test, and receiver operating characteristic curve (ROC) were used for predictive power of statistical evaluation. Area under the curve (AUC) values for ROC were greater than 0.80 for all models. For flood vulnerability modelling, the Dagging model excels, followed by RF, ANN, SVM, RS, and various benchmark models. The approach and solutionoriented results outlined in this document will help state and local governments and policy makers to mitigate flood-related threats and implement effective mitigation strategies to mitigate future damage.

PAPER.5

Author: Marj Tonini, Mirko D'Andrea, Guido Biondi, Silvia Degli Esposti, Andrea Trucchia

Year: 2020

Wildfire vulnerability maps show the spatial probability that an area will burn in the future based solely on site-specific local properties. Current research in this area often relies on statistical models, often augmented by expertise in data collection and processing. In recent years, machine learning algorithms have proven their prowess in this area thanks to their ability to learn from data by modelling hidden relationships. In this study, the authors present a randomized His forest-based approach that allows the refinement of forest fire vulnerability maps for the Ligurian region of Italy. The area is highly susceptible to forest fires due to its dense and uneven vegetation, over 70% forested, and favourable climate conditions. Vulnerability considers a dataset of fire perimeters mapped over a 21-year period (1997 to 2017) and various geo-ecological predispositions (land cover, vegetation type, road network, elevation, and runoff). was evaluated. The main objective was to compare different models to determine the effects of (i) including or excluding adjacent vegetation types as additional predispositions and (ii) using increasing convolution in the spatial cross-validation procedure. was to evaluate. Finally, two fire season susceptibility maps were created and validated. This result highlights the ability of the proposed approach to identify areas likely to be affected by wildfires in the near future and its usefulness in assessing the effectiveness of fire fighting measures.

PAPER.6

Author: Jo Schlemper, Jose Caballero, Joseph V. Hajnal, Anthony Price, Daniel Rueckert

Year: 2017

Inspired by recent advances in deep learning, we use deep cascades of convolutional neural networks (CNNs) to reconstruct dynamic sequences of 2D cardiac magnetic resonance (MR) images from under sampled data, We propose a framework to streamline and accelerate the data acquisition process. Especially if the data is collected using aggressive Cartesian under sampling. First, we show that the proposed method improves current 2D compressed acquisition approaches, such as reconstruction speed, if each 2D image frame is reconstructed independently. Next, by jointly reconstructing the frames of the sequence, we show that CNNs can efficiently learn spatio-temporal correlations by combining convolution and data exchange approaches. We show that the proposed method consistently outperforms the state-of-the-art and can preserve anatomical structures more faithfully up to 11-fold under sampling. Moreover, reconstruction is very fast. Each complete dynamic sequence can be reconstructed in less than 10 seconds, and in 2D each image frame can be reconstructed in 23 ms, enabling real-time applications.

PAPER.7

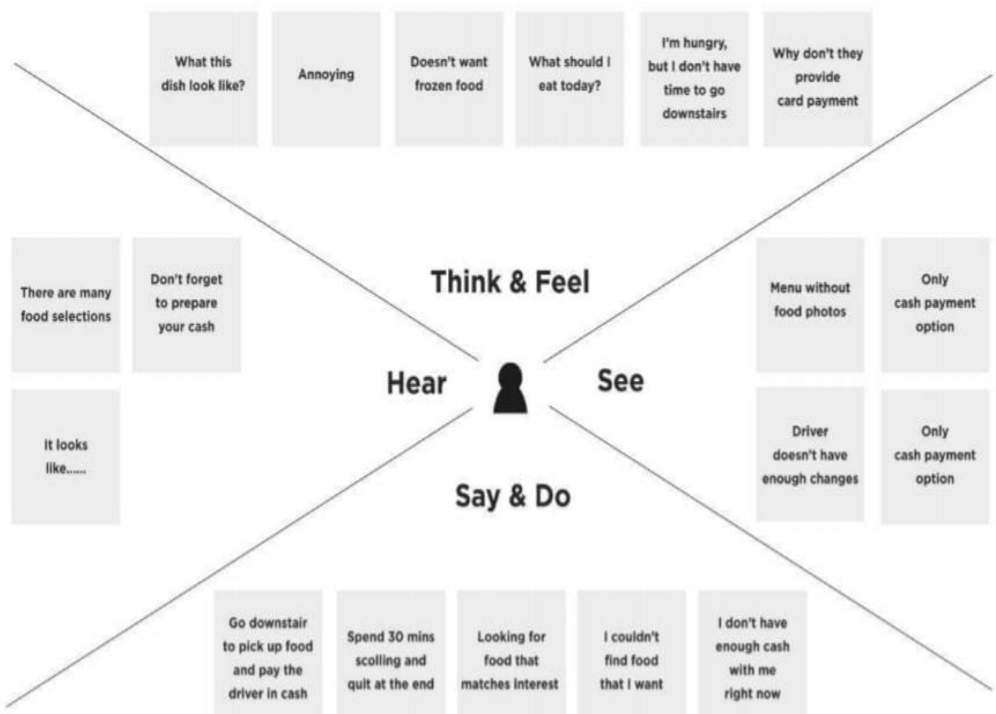
Author: Chaohui Tang, Qingxin Zhu, Wenjun Wu, Wenlin Huang, Chaoqun Hong, Xinzheng Niu

Year: 2020

In recent years, deep learning has become a research hotspot and has had a major impact on computer vision. Deep CNN has proven to be the most important and effective model for image processing, but it is prone to over fitting due to lack of training examples and large number of learning parameters. In this work, we propose a new his two-stage CNN image classification network named ``Improved Convolutional Neural Networks with Image Enhancement for Image Classification", PLANET for short. Image training example. Inspired by computer vision “object motion” scenes, Inner Move can improve the generalization power of deep CNN models for image classification tasks. Ample experimental results show that PLANET using his Inner Move for image enhancement outperforms comparison algorithms, and that Inner Move has significant effects over comparative data enhancement methods for image classification tasks.

3.IDEATION AND PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS



3.2 IDEATION AND BRAINSTORMING

1

Problem Statement

Natural catastrophes not only disrupt the ecology that supports human life, but they also obliterate vital facilities and properties in human society, changing the ecosystem permanently. Natural occurrences like earthquakes, cyclones, floods, and wildfires can bring disaster. To mitigate ecological losses from natural disasters, several deep learning approaches have been used by numerous researchers. However, identification of natural disasters still has difficulties because of the complex and unbalanced image structures. In order to address this issue, we created a multilayered deep convolutional neural network model that detects natural disasters and estimates their intensity.

3

Group ideas

Accurate intensity prediction	Previous history can be used for data analytics which is more effective for future predictions	Artificial intelligence is playing increasingly important role in disaster risk reduction	currently AI can predict four types of natural disasters accurately.
With enormous amounts of good quality datasets, AI can predict the occurrence of numerous natural disasters, which can be the difference between life and death for thousands of people	AI analyses the data to learn about the patterns of various earthquakes and predict where the aftershock might hit..	Artificial intelligence can improve disaster response, from reducing the time to assess damage and effectively deliver aid.	With an AI strategy in place, disaster response will be quick as possible, also the amount of unplanned downtime could be reduced to virtually nothing.
tsunami can be predict through combining global navigation satellite system data with AI.	AI based algorithms can organize disaster data in the order of severity.	The system would use AI to analyze images of disaster and predict the damage they could cause.	It is important to analyze and assess the extent of damage and ensure the right aid goes first those who need it most.

3.3 PROPOSED SOLUTION

Proposed Solution Template:

S.No	Parameter	Description
1.	Problem Statement (Problem to be solved)	Natural disasters are inevitable, but their impact can be mitigated with the help of deep learning and neural networks. Providing timely and effective rescue, assistance and rehabilitation at disaster sites can keep people and property out of danger.
2.	Idea / Solution description	Based on the image dataset, catastrophe are recognized and what catastrophe are classified. A built-in webcam that captures video frames and the video frames are compared with a pre-trained model to identify the nature of the disaster and display it in the OpenCV window.
3.	Novelty / Uniqueness	Accurate prediction of disaster intensity without physical human intervention. It will function more effectively and without error amid a disaster.
4.	Social Impact / Customer Satisfaction	AI helps rescue teams understand immediate situations. It can predict ongoing disasters and help people be aware of upcoming disasters.
5.	Business Model (Revenue Model)	Organizations benefit from subscribers paying for specific features. Governments can also use this to generate future income.
6.	Scalability of the Solution	Improve collaboration between current initiatives focused on specific use cases from a few partners into a more impact-oriented network of Alenabled disaster relief.

3.4 PROBLEM SOLUTION FIT

Project Title: Natural Disasters Intensity Analysis and Classification using Artificial Intelligence

Project Design Phase-I - Solution Fit Template

Team ID: PNT2022TMID26770

Define CS, fit into CC

1. CUSTOMER SEGMENT(S) — CS

Who is your customer?
i.e. working parents of 0-5 y.o. kids

- Government
- NDRF
- Meteorologist
- Climatologist
- Seismologist
- People who have affected by disaster

6. CUSTOMER CONSTRAINTS — 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.

- Cost
- Inaccessibility to the Internet
- Communication breakdown
- Limited resources
- Uncertain climate change

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

- By protecting forests and coral reefs, we can lessen the likelihood of landslides, hurricanes, and rising sea levels.
- Neglecting other underlying issues that may be causing this event
- recognizing the contrast between indirect and direct impacts
- outcomes that are precise and effective lessen severe harm

Explore AS, differentiate

Focus on J&P, tap into BE, understand RC

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.

Although intensity is significant, it is not always simple to recognize it. It is difficult to identify the causes of natural disasters. For instance, earthquakes are difficult to detect but can be used to detect tsunamis. Although plate tectonic theory is supposed to be able to detect it, it is not always reliable.

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.

- Moon activities
- Plate Tectonic movement
- Mining
- Global warming
- Ocean currents
- instability in the lower atmosphere.

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)

- Discover the root reasons to be able to prevent it.
- Offering training programs for professional growth
- Gaining adoption skills and reconstructing one's life and career
- Avoid and neutralize the causes of calamity.
- Acquiring information about disaster relief
- Gaining a better understanding about what to do and what not to do in the event of a disaster

Focus on J&P, tap into BE, understand RC

Identify strong TR & EM

3. TRIGGERS — TR

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

If people who live in disaster-prone locations learned about the items that allow them to foresee danger before it occurs, they would buy them at any price. To be safe, other people will also want to possess it.

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.

Even if their lives may have been idyllic before the accident, they may now be unhappy, frightened, furious, or afraid because they have lost their loved ones, their jobs, or their homes. Additionally, this undermines their confidence. However, if they are aware of it ahead, even if they may be afraid, they will be confident and prepared to face and rebuild.

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 behavior.

To assist AI in tracking and foretelling the influence of diverse environmental conditions and their effects, we want to include reinforcement learning algorithms. This lets the rescue crew take quick and efficient action in addition to minimizing the damage.

8. CHANNELS of BEHAVIOUR — CH

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

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

ONLINE:

- In an effort to learn more about the calamity or how to avoid it, they seek out technical assistance or professional advice online.
- If they are feeling down about the situation, they seek professional help.
- They strive for more specific information regarding the disaster's effects.

OFFLINE:

- They participate in relief efforts or develop initiatives to lessen the effects of imminent disasters or prevent them altogether

Identify strong TR & EM

4.REQUIREMENT ANALYSIS

4.1 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**	Obtain authorization from the web camera.
FR-2	**Disaster Prediction**	Natural disaster is categorized based on the webcam image.
FR-3	**Accuracy**	The accuracy is higher since the training and testing images are huge.
FR-4	**Speed**	Results are generated from the input images more efficiently.
FR-5	**Resolution**	The resolution of the integrated web camera should be high enough to capture the video frames.
FR-6	**User Interface**	Enhancing interaction in web design services.

4.2 NON- FUNCTIONAL REQUIREMENTS

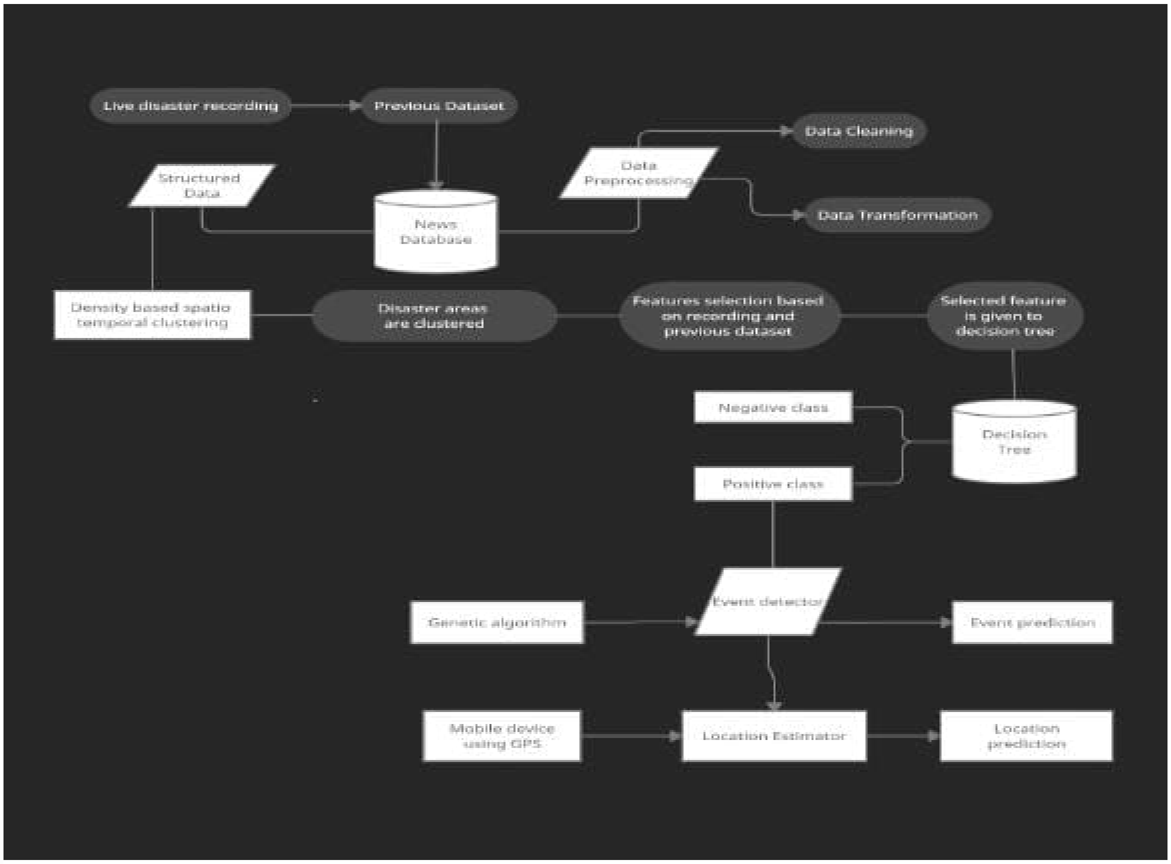
Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Collecting information from reliable sources in addition to gathering research.
NFR-2	**Security**	When a system freezes or disconnects caused due to, too many users using it at once, it should save all of the user's previous actions up until the abnormal event.
NFR-3	**Reliability**	Both the website and the accurate disaster prediction can be fault tolerant.
NFR-4	**Performance**	It is demonstrated that the model provides accuracy of over 90% following continuous training.
NFR-5	**Availability**	This website will be accessible at any situation of disaster for 24 hours.
NFR-6	**Scalability**	The website may be extended to the NDRF and users, and it can be used with web browsers like Microsoft Edge and Google Chrome.

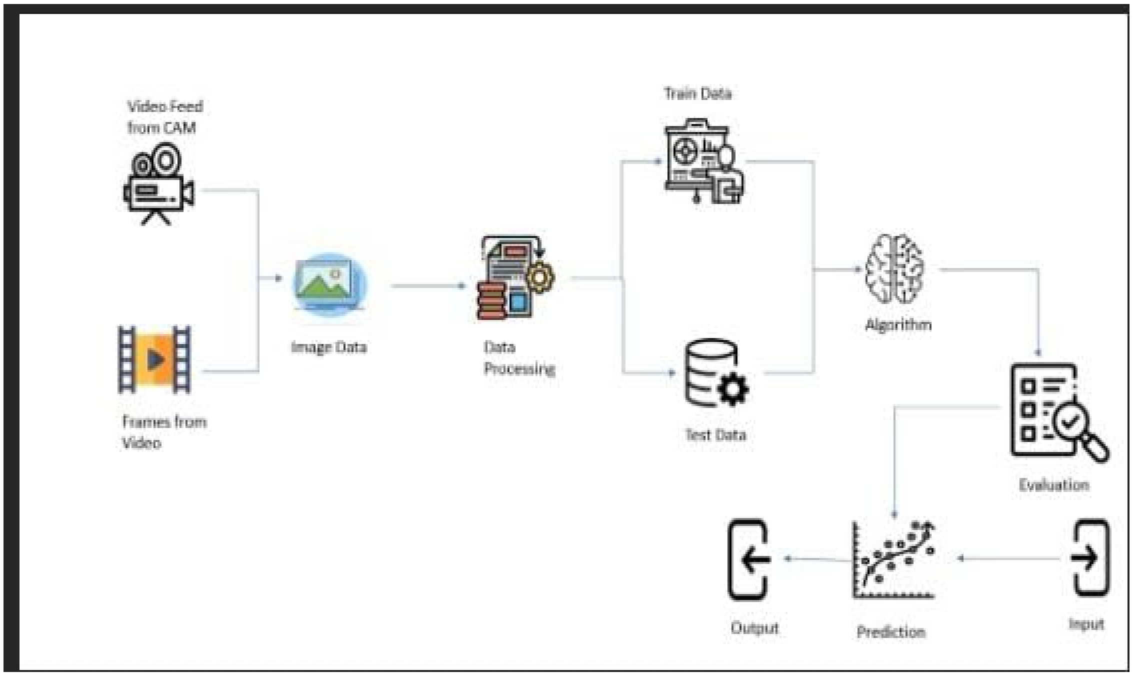
5.PROJECT DESIGN

5.1 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 shoes how data enters and leaves the system,what changes the information,and where data is stored.



5.2 SOLUTION AND TECHNICAL ARCHITECTURE



5.3 USER STORIES

User Stories

Use the below template to list all the user stories for the product.

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
	Login /Dashboard	USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1
Customer (Web user)	Registration	USN-3	As a user, I can register for the application through Gmail	I can register & access the dashboard wih google	Low	Sprint-1
	Login /Dashboard	USN-4	As a user, I can log into the application by entering email & password	I can access my account / dashboard	Medium	Sprint-1
Admin	Login	USN-5	Admin can change the data and monitor the functionality of the programme.	Admin can carry out actions and make modifications.	High	Sprint-1
Seismologist	Disaster prediction	USN-6	Seismologists use the website to review the previous dataset and contrast it with the current data.	They can draw the conclusion about the seismic influence based on the results.	High	Sprint-2
		USN-7		They can forewarn of impending events in advance.	High	Sprint-2
Meteorologists	Disaster prediction	USN-8	Meteorologists use this website to monitor global weather and climate fluctuations.	They are able to give early warning of imminent events.	Medium	Sprint-2
NDRF	Rescue	USN-9	In order to save more lives, they can expedite the rescue effort thanks to the early warning.	Their fast response can reduce casualties.	High	Sprint-2

6. PROJECT PLANNING AND SCHEDULING

6.1 SPRINT PLANNING AND ESTIMATION

Milestone and Activity List:

TITLE	DESCRIPTION	DATE
Literature Survey & Information Gathering	Literature survey on the selected project & gathering information by referring the, technical papers, research publications etc.	3 SEPTEMBER 2022
Prepare Empathy Map	Prepare Empathy Map Canvas to capture the user Pains & Gains, Prepare list of problem statements	10 SEPTEMBER 2022
Ideation	List the by organizing the brainstorming session and Prioritize the top 3 ideas based on the feasibility & importance.	10 SEPTEMBER 2022

Proposed Solution	Prepare the proposed solution document, which includes the novelty, feasibility of idea, business model, social impact, scalability of solution, etc.	2 OCTOBER 2022
Problem Solution Fit	Prepare problem - solution Fit document.	29 SEPTEMBER 2022
Solution Architecture	Prepare solution Architecture document.	20 OCTOBER 2022
Solution Requirements	Prepare Solution Requirements (Functional and Non-functional)	17 OCTOBER 2022
Customer Journey	Prepare the customer journey maps to understand the user interactions & experiences with the application	8 OCTOBER 2022
Data Flow Diagrams	Draw the data flow Diagrams and submit for review.	17 OCTOBER 2022
Technology Architecture	Architecture diagram.	15 OCTOBER 2022
Prepare Milestone & Activity List	Prepare the milestones & Activity list of the project.	27 OCTOBER 2022
Project Development - Delivery of Sprint- 1, 2, 3 & 4	Develop & submit the developed code by testing it.	IN PROGRESS....

6.2 SPRINT DELIVERY SCHEDULE

Product Backlog, Sprint Schedule, and Estimation (4 Marks)

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming that.	2	Low	Priyatharshini
Sprint-1		USN-2	As a user, I will receive confirmation email once I have registered for the application.	3	High	Annapoorani
Sprint-1	Login	USN-3	As a user, I adapt to logging into the system with credentials.	2	Low	Ranjani
Sprint-1	Designati on of Region	USN-4	As a user, I can collect the dataset and select the region of interest to be monitored and analyzed.	5	Medium	Priyanka
Sprint-1	Analysis of required phenomenon	USN-5	As a user, I can regulate certain factors influencing the action and report on past event analysis.	4	High	Priyatharshini
Sprint-1	Algorithm selection	USN-6	As a user, I can choose the required algorithm for specific analysis.	4	Medium	Annapoorani
Sprint-1	Training and Testing	USN-7	As a user, I can train and test the model using the algorithm.	4	High	Ranjani
Sprint-1	Detection and analysis of data	USN-8	As a user, I can detect and visualize the data effectively.	4	High	Priyanka

Sprint 2	Create and configure IBMcloud services	USN-9	As a user I need to enrolL thecloud registration	3	Medium	Priyatharshini
Sprint 2		USN-10	As a user, I will create IBM cloud Account Watson AI platformby accessing cloud account	7	High	Annapoorani
Sprint 2		USN-11	After creating node get device type and id	1	Low	Ranjani
Sprint 2		USN-12	Simulate the node created	3	Medium	Priyanka
Sprint 3	Create a database in Cloud DB	USN-13	Launch the cloud DB andcreate database to store the location data	4	High	Priyatharshini
Sprint 3	Develop the Python script	USN-14	Install the python software	2	Low	Annapoorani
Sprint 3		USN-15	Develop the python flask topublish details to IBM AI platform	6	High	Ranjani
Sprint 3		USN-16	Integrate the device ID , authentication token in python flask	2	Low	Priyanka
Sprint 3		USN-17	Develop the python code for publishing the location (latitude & longitude) to IBM AI platform	8	High	Priyatharshini
Sprint 4	Create the Web application using node Red	USN-18	As a user, I can buildwith the web application.	5	High	Annapoorani
Sprint 4		USN-19	Connect the IBM AI platform and get the location and store the data in the cloud	2	Medium	Ranjani
Sprint 4		USN-20	Create the multilayered deep convolution neural network mode that tells the intensity ofdisaster a	8	High	Annapoorani

Sprint 4		USN-21	Send the notification is the webcam to capture the video frame	4	High	Priyatharshini

Project Tracker, Velocity & Burndown Chart: (4 Marks)

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	31 Oct 2022	20	31 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

Velocity:

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

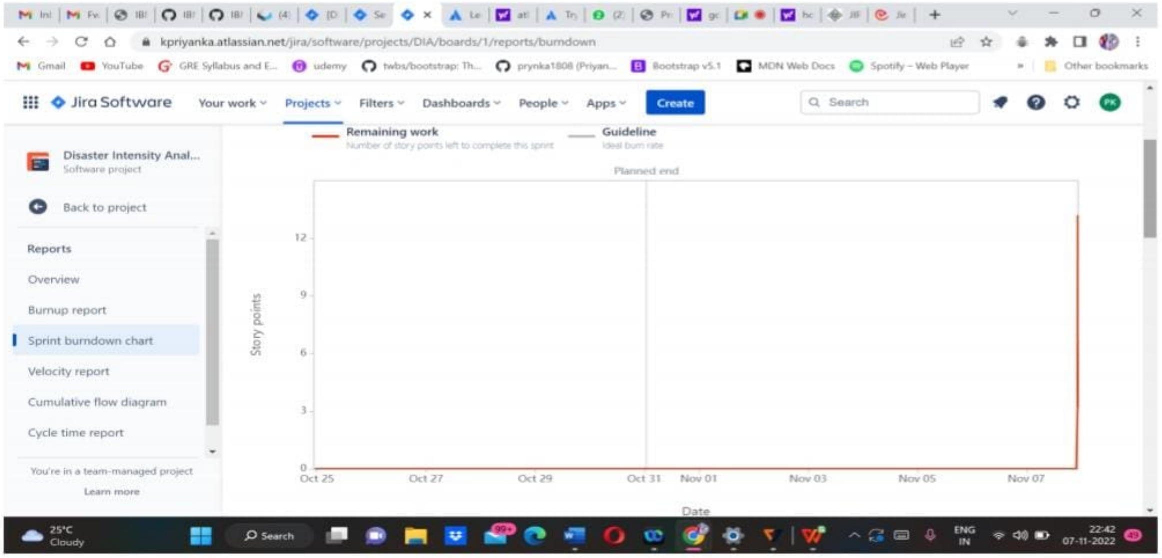
Average velocity=sprint duration/velocity=20/6=3

Sprint	Total Story Points	Duration	Average Velocity
Sprint-1	28	6	4.6
Sprint-2	14	6	2.3
Sprint-3	22	6	3.6
Sprint-4	19	6	3.1
Total Sprint	83	24	3.4

Burndown chart:

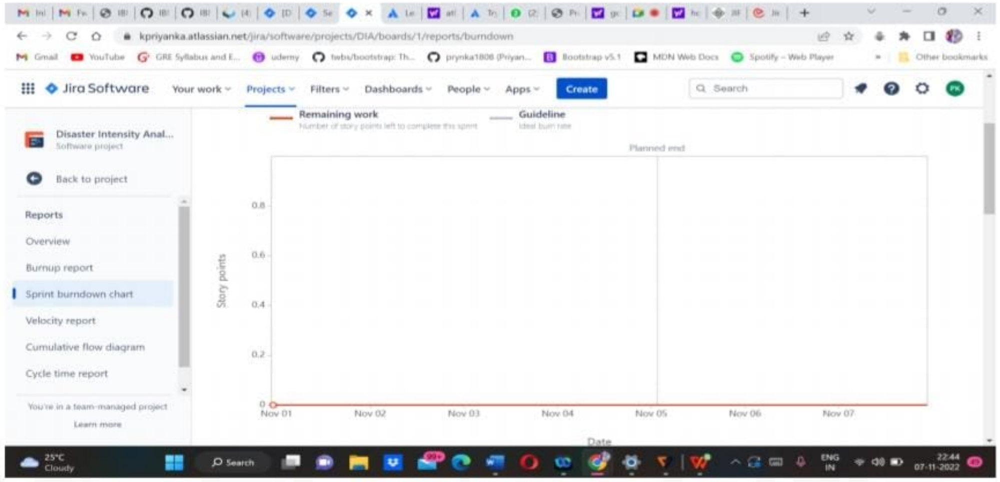
A burn down chart is a graphical representation of work left to do versus time.It is often used in agile software developement methologies such as scrum. However, burn down charts can be applied to any project containing measurable progress over time.

Sprint-1:

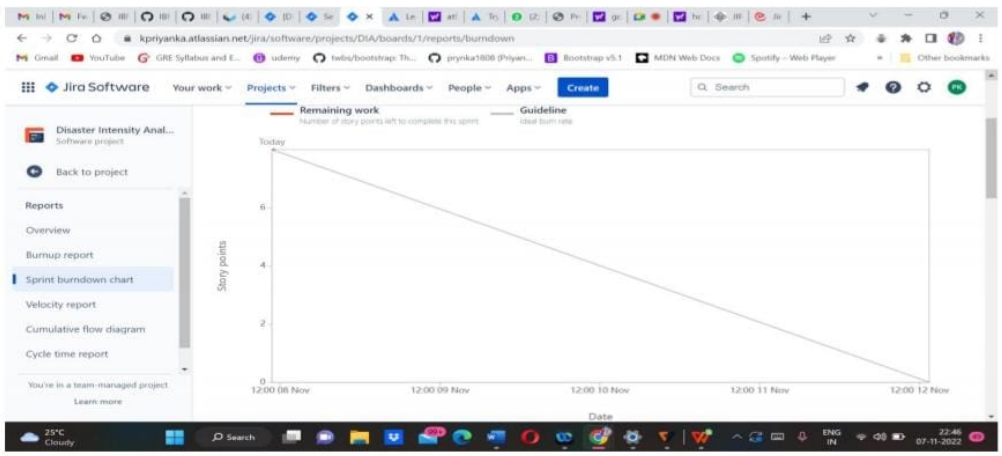


kpriyanka.atlassian.net/jira/software/projects/DIA/boards/1/reports/burndown
Jira Software
Your work
Projects
Filters
Dashboards
People
Apps
Create
Search
Disaster Intensity Anal...
Software project
Back to project
Reports
Overview
Burnup report
Sprint burndown chart
Velocity report
Cumulative flow diagram
Cycle time report
You're in a team-managed project
Learn more
Remaining work
Number of story points left to complete this sprint
Guideline
Ideal burn rate
Planned end
Story points
Date

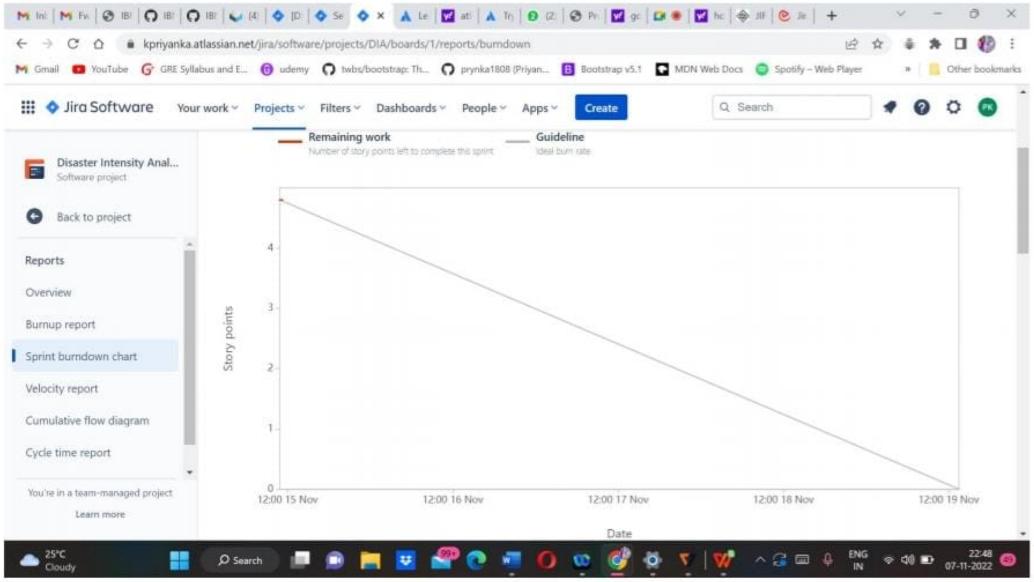
Sprint-2:



Sprint-3:

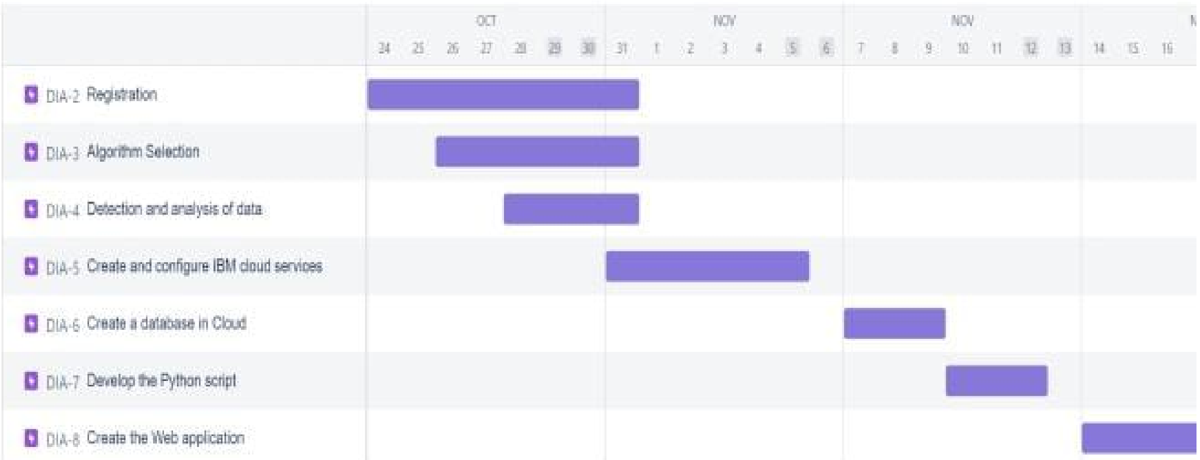


Sprint-4:



kpriyanka.atlassian.net/jira/software/projects/DIA/boards/1/reports/burndown
Jira Software
Your work
Projects
Filters
Dashboards
People
Apps
Create
Search
Disaster Intensity Anal...
Software project
Back to project
Reports
Overview
Burnup report
Sprint burndown chart
Velocity report
Cumulative flow diagram
Cycle time report
You're in a team-managed project
Learn more
Remaining work
Number of story points left to complete this sprint
Guideline
Ideal burn rate
Story points
Date
12:00 15 Nov
12:00 16 Nov
12:00 17 Nov
12:00 18 Nov
12:00 19 Nov

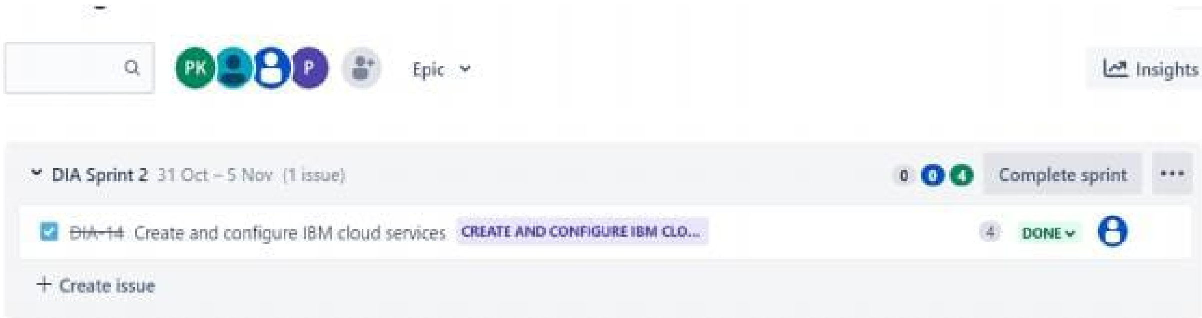
RoadMap:



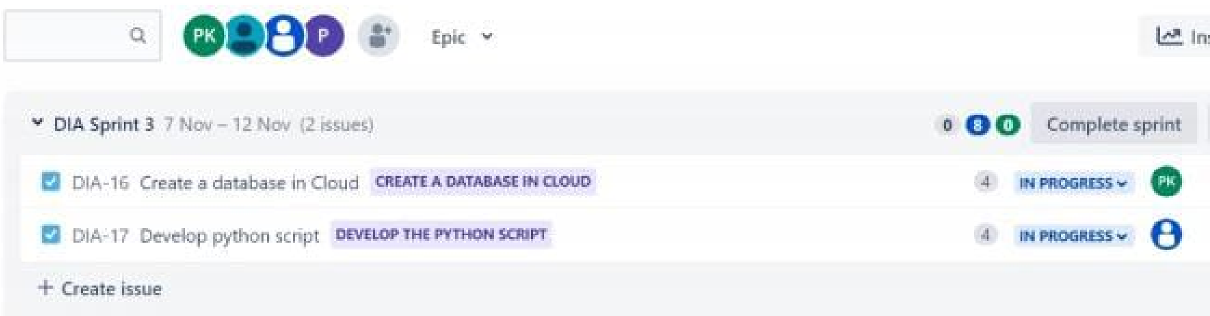
Back Log:

Sprint-1:

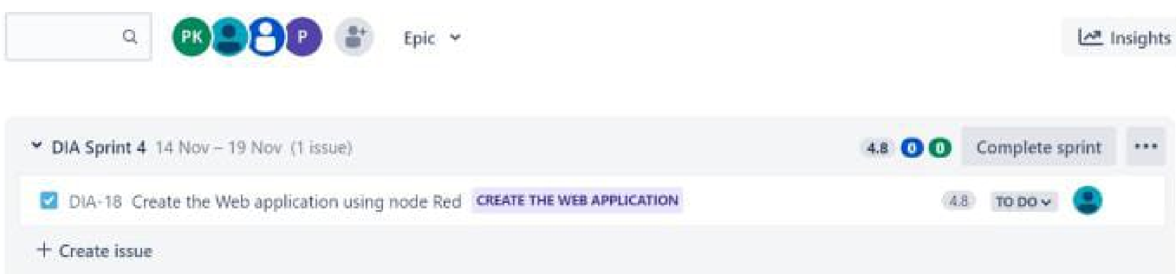
Sprint-2:



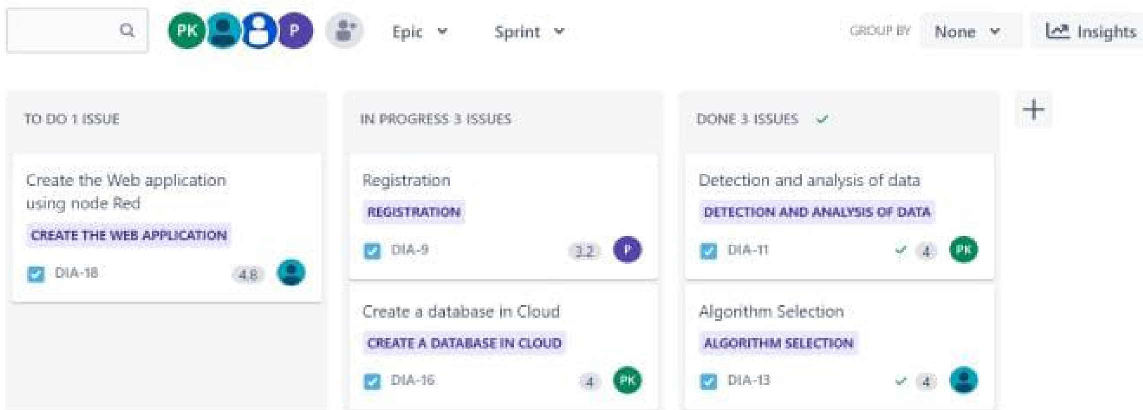
Sprint-3:

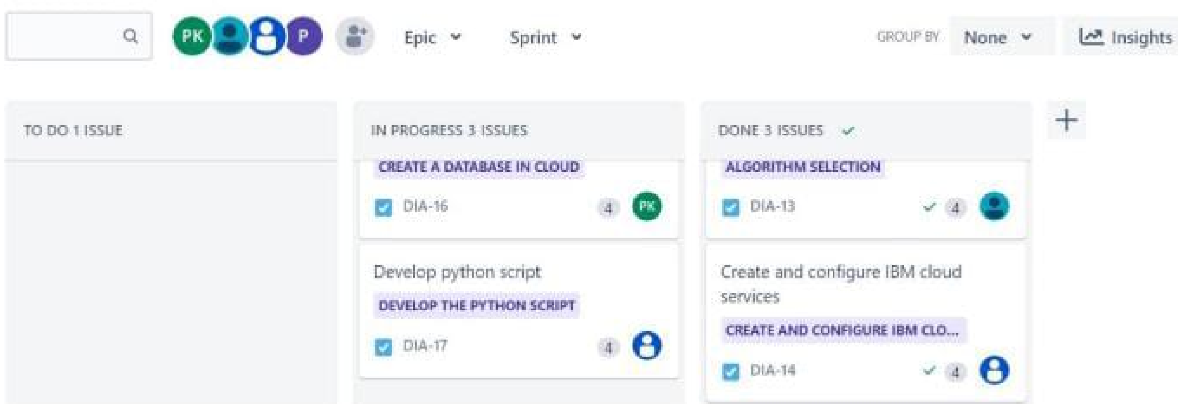


Sprint-4:



Board:





6.3 REPORTS FROM JIRA

7. CODING AND SOLUTIONING

7.1 FEATURE 1

INSERTING NECESSARY LIBRARIES:
Numpy:It is an open source numerical python library.
Scikit-learn:It is a machine learning library for python.
OpenCV:OpenCV is a library of programming funtions mainly aimed at real-time

computer vision.
Flask:Web framework used for building web application

Inserting necessary libraries

```
In [1]: 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

        Using TensorFlow backend.

In [2]: tensorflow.__version__

Out[2]: '2.5.0'

In [3]: tensorflow.keras.__version__

Out[3]: '2.5.0'
```

CREATING THE MODEL:

Creating the model a classifier sequential.classifier is a machine learning algorithm that determines the class of the input element based on the set of the feature.

Creating the Model

```
In [9]: # 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
```

```
In [10]: classifier.summary()#summary of our model

Model: "sequential"
_________________________________________________________________
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 (MaxPooling2 (None, 13, 13, 32)        0
_________________________________________________________________
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
```

LOADING DATA AND PERFORMING DATA AUGUMENTATION:

model using convolution2D function.Convolution2D parameter is an number of filters that convolution layer from.Then,using a flatten() function that flatten the multidimensional input denser into the denser.

Loading our data and performing Data Augumentation

```
In [5]: #performing data agumentation to train data
x_train = train_datagen.flow_from_directory(r'C:\Users\ELCOT\Downloads\projest\ibm\dataset\train_set',target_size=(64, 64),batch_
                                            color_mode='rgb',class_mode='categorical')
#performing data agumentation to test data
x_test = test_datagen.flow_from_directory(r'C:\Users\ELCOT\Downloads\projest\ibm\dataset\test_set',target_size=(64, 64),batch_si
                                            color_mode='rgb',class_mode='categorical')

Found 742 images belonging to 4 classes.
Found 198 images belonging to 4 classes.

In [6]: print(x_train.class_indices)#checking the number of classes

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

In [7]: print(x_test.class_indices)#checking the number of classes

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

In [8]: from collections import Counter as c
c(x_train .labels)

Out[8]: Counter({0: 220, 1: 156, 2: 198, 3: 168})
```

COMPILING THE MODEL:

The model is compiled using the following code.

```
In [11]: # Compiling the CNN
         # categorical_crossentropy for more than 2
         classifier.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])
```

FITTING THE MODEL:

Fitting the model with 70 epoch

SAVING THE MODEL:

Saving the Model as disaster.h5. disaster.h5 file is used to find the image

```
149/149 [==============================] - 63s 421ms/step - loss: 0.0427 - accuracy: 0.9852 - val_loss: 1.8291 - val_accurac
y: 0.7374
Epoch 66/70
149/149 [==============================] - 62s 415ms/step - loss: 0.0574 - accuracy: 0.9838 - val_loss: 1.7194 - val_accurac
y: 0.7424
Epoch 67/70
149/149 [==============================] - 62s 415ms/step - loss: 0.0434 - accuracy: 0.9879 - val_loss: 1.8268 - val_accurac
y: 0.7374
Epoch 68/70
149/149 [==============================] - 62s 415ms/step - loss: 0.0796 - accuracy: 0.9717 - val_loss: 1.7385 - val_accurac
y: 0.7323
Epoch 69/70
149/149 [==============================] - 64s 427ms/step - loss: 0.0287 - accuracy: 0.9865 - val_loss: 2.0659 - val_accurac
y: 0.7424
Epoch 70/70
149/149 [==============================] - 63s 424ms/step - loss: 0.0242 - accuracy: 0.9933 - val_loss: 3.4165 - val_accurac
```

classification files. Model.json represents that Jason stands for JavaScript object

Saving the Model

```
In [13]: # Save the model
         classifier.save('disaster.h5')

In [14]: model_json = classifier.to_json()
         with open("model-bw.json", "w") as json_file:
             json_file.write(model_json)
```

PREDICTING RESULTS:

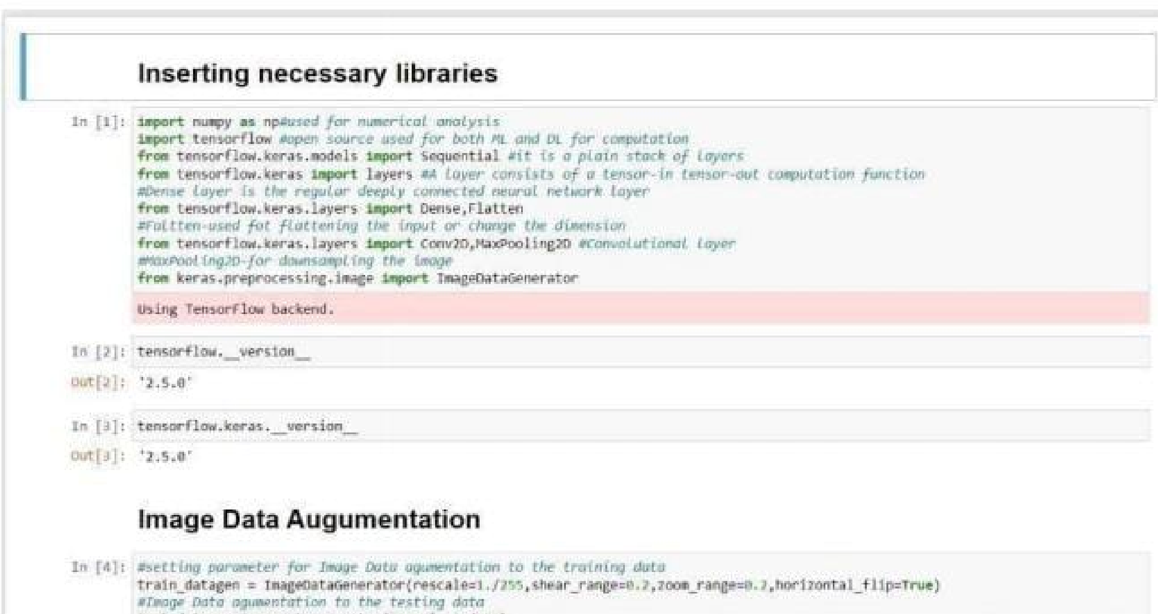
Loding model from the tensorflow keras model and loading the image then converting into array.

```
In [15]: from tensorflow.keras.models import load_model
         from keras.preprocessing import image
         model = load_model("disaster.h5") #loading the model for testing

In [ ]:

In [16]: img = image.load_img(r'C:\Users\ELCOT\Downloads\projest\ibm\dataset\test_set\Cyclone\870.jpg',grayscale=False, target_size= (64,
         x = image.img_to_array(img)#image to array\n",
         x = np.expand_dims(x,axis = 0)#changing the shape\n",
         pred = model.predict_classes(x)#predicting the classes\n",
         pred

C:\Users\ELCOT\anaconda3\lib\site-packages\tensorflow\python\keras\engine\sequential.py:455: UserWarning: `model.predict_classe
s()` is deprecated and will be removed after 2021-01-01. Please use instead:* `np.argmax(model.predict(x), axis=-1)`,   if your
model does multi-class classification   (e.g. if it uses a `softmax` last-layer activation).* `(model.predict(x) > 0.5).astype
("int32")`,   if your model does binary classification   (e.g. if it uses a `sigmoid` last-layer activation).
  warnings.warn('`model.predict_classes()` is deprecated and '

Out[16]: array([0], dtype=int64)

In [17]: index=['Cyclone','Earthquake','Flood','Wildfire']
         result=str(index[pred[0]])
         result

Out[17]: 'Cyclone'
```

7.2 FEATURE 2

DETECTION AND ANALYSIS OF DATA:

After testing and training the model,data which given in dataset are analysed and visualised effectively to detect the disaster type.



Inserting necessary libraries

```
In [1]: 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

Using TensorFlow backend.

In [2]: tensorflow.__version__
Out[2]: '2.5.0'

In [3]: tensorflow.keras.__version__
Out[3]: '2.5.0'
```

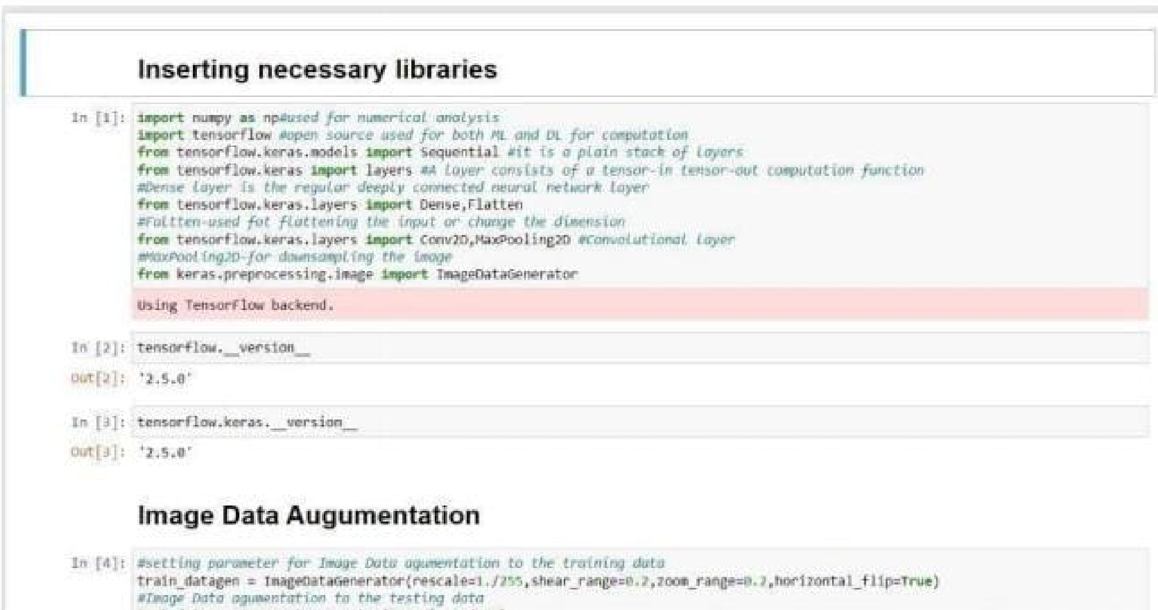
Image Data Augumentation

```
In [4]: #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
```

MODEL BUILDING:

Building a model with web application named FLASK model building process

Import the model libraries
Initializing the model
Adding hidden layer
Configure the learning process
Training and testing the model
All the above processes are testing the model.



Inserting necessary libraries

```
In [1]: 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

        Using TensorFlow backend.

In [2]: tensorflow.__version__
Out[2]: '2.5.0'

In [3]: tensorflow.keras.__version__
Out[3]: '2.5.0'
```

Image Data Augumentation

```
In [4]: #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
```

7.3 DATABASE SCHEMA

8.TESTING

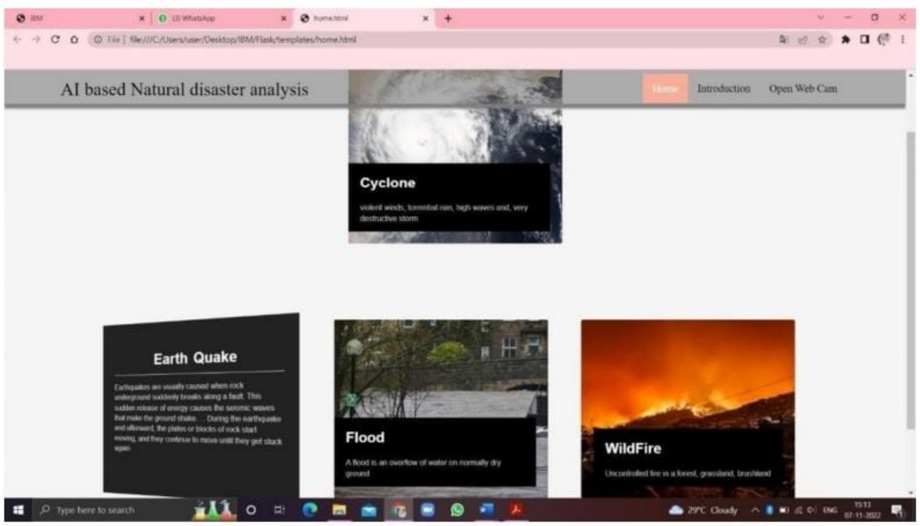
8.1 TEST CASES

COLLECTION OF DATASET:

The images of Disaster-prone areas are collected and organized into the subdirectories. The images of four types of Natural Disasters, Cyclone, Earthquake, Flood, Wildfire are collected and saved with the respective names. For more accuracy, Dataset with more images is selected and trained. The respective Dataset for the project is downloaded from the following reference link:
https://drive.google.com/file/d/11FdbTaJVrpwQmaCLV5gYYDQlfTeD0uz/view? usp=sharing

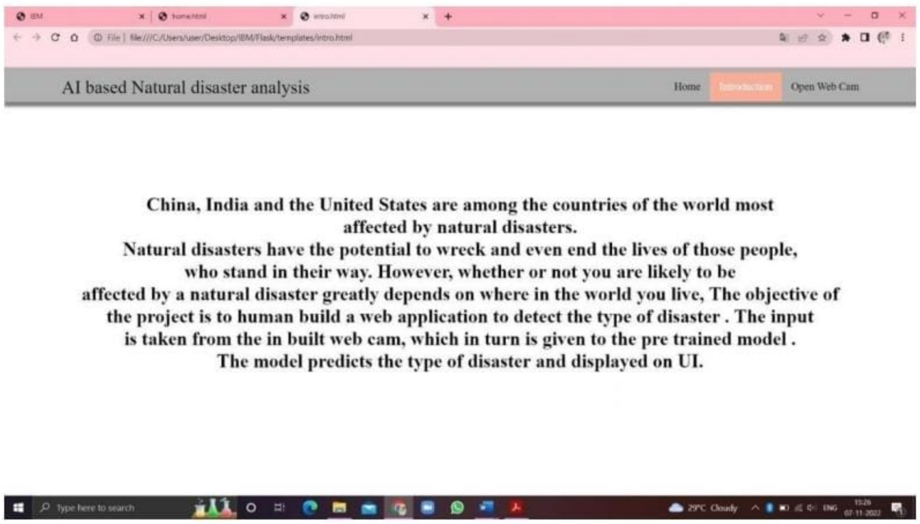
CREATION OF HOME PAGE:

Using HTML and CSS, the Home page is created. From the Home page the User can be able to know the basics of the frequently occurring Disasters. The home.html page is given below:



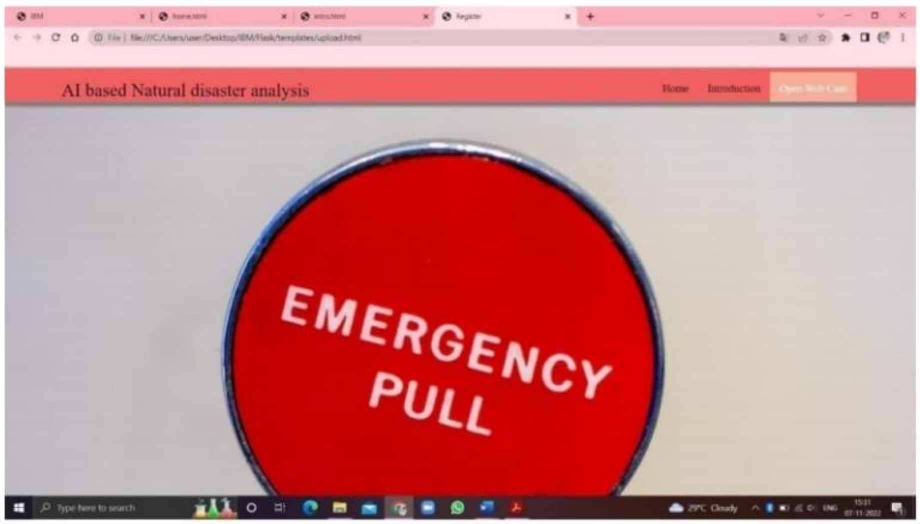
CREATION OF INTRO PAGE:

Using HTML and CSS, the intro page is created. From the intro page the user can be able to know about the project's introduction or abstract. The intro.html page is given below:



OPENING WEB CAM:

Using HTML and CSS, the upload.html page is created. Through this page User can be able to open the web cam to know about current disaster. The upload.html page is given below:



8.2 USER ACCEPTANCE TESTING

9.RESULTS

9.1 PERFORMANCE METRICS

INTEGRATE THE WEB APP WITH AI MODEL:

After creating the model ,the model should be integrated with the web app using the flask application.

```python
            output = frame.copy()
            #print("apple")
            frame = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
            frame = cv2.resize(frame, (64, 64))
            #frame = frame.astype("float32")
            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]])
            #print(result)
            #result=result.tolist()

            cv2.putText(output, "activity: {}".format(result), (10, 120), cv2.FONT_HERSHEY_PLAIN,
                        1, (0,255,255), 1)
            #playaudio("Emergency it is a disaster")
            cv2.imshow("Output", output)
            key = cv2.waitKey(1) & 0xFF

            # if the `q` key was pressed, break from the loop
            if key == ord("q"):
                break

        # release the file pointers
        print("[INFO] cleaning up...")
        vs.release()
        cv2.destroyAllWindows()
        return render_template("upload.html")


if __name__ == '__main__':
    app.run(debug=False,threaded=True)
```

10.ADVANTAGES AND DISADVANTAGES

11.CONCLUSION

Many researchers have attempted to use different deep learning methods for detection of natural disasters. However, the detection of natural disasters by using deep learning techniques still faces various issues due to noise and serious class imbalance problems. To address these problems, we proposed a multilayered deep convolutional neural network for detection and intensity classification of natural disasters. The proposed method works in two blocks—one for detection of natural disaster occurrence and the second block is used to remove imbalanced class issues. The results were calculated as average statistical values: sensitivity, 97.54%; specificity, 98.22%; accuracy rate, 99.92%; precision, 97.79%; and F1-score, 97.97% for the proposed model.

12.FUTURE SCOPE

Individual or group donations form an important aspect of disaster relief operations. Donation-based crowdfunding (DBC) tasks are often listed on crowdfunding platforms to attract donors to donate for a specific reason in a stipulated time. As the frequency and intensity of disasters has increased over time, these platforms have gained in popularity, and they need a constant and consistent flow of funds to achieve their targets. Artificial intelligence (AI) tools are often adopted by these channels to enhance their operational performance. We understand the process of adoption through uses and gratification theory, which is dominated by motivational factors, such as the utilitarian and symbolic benefits which DBC intends to achieve. The inflow of cash from multiple donors across the world, guided by AI tools, also gives rise to risks; therefore, we have used a moderating variable to better understand the operational performance of DBC. We collected empirical data through 293 responses from owners of DBC tasks in the context of disaster relief operations. We tested our hypotheses using partial least square structured equation modelling and controlled for intensity of disaster and crowdfunding task duration. Our results offer a significant extension to uses and gratification theory by understanding a positive relation between uses and gratification benefits and the adoption of AI tools for boosting operational performance. We project that, whereas the duration of a crowdfunding task plays an essential role in collecting the required funds for disaster relief operations, the intensity of the disaster does not impact the process of adopting AI tools or on their operational performance. Our study offers critical insights for understanding aspects of designing and implementing AI in DBC scenarios, which has been a grey area in understanding donors' behavior.

13.APPENDIX