**CHAPTER-1**

**INTRODUCTION**

EARTHQUAKE hypocenter localization is essential in the field of seismology and plays a critical role in a variety of seismological applications such as tomography, source characterization, and hazard assessment. This underscores the importance of developing robust earthquake monitoring systems for accurately determining the event origin times and hypocenter locations. In addition, the rapid and reliable characterization of ongoing earthquakes is a crucial, yet challenging, task for developing seismic hazard mitigation tools like earthquake early warning (EEW) systems. While classical methods have been widely adopted to design EEW systems, challenges remain to pinpoint hypocenter locations in real-time largely due to limited information in the early stage of earthquakes. Among various key aspects of EEW, timeliness is a crucial consideration and additional efforts are required to further improve the hypocenter location estimates with minimum data from 1) the first few seconds after the P-wave arrival and 2) the first few seismograph stations that are triggered by the ground shaking.

The localization problem can be resolved using a sequence of detected waves (arrival times) and locations of seismograph stations that are triggered by ground shaking. Among various network architectures, the recurrent neural network (RNN) is capable of precisely extracting information from a sequence of input data, which is ideal for handling a group of seismic stations that are triggered sequentially following the propagation paths of seismic waves. This method has been investigated to improve the performance of real-time earthquake detection and classification of source characteristics. Other machine learning based strategies have also been proposed for earthquake monitoring. Comparisons between traditional machine learning methods, including the nearest neighbor, decision tree, and the support vector machine, have also been made for the earthquake detection problem. However, a common issue in the aforementioned machine learning based frameworks is that the selection of input features often requires expert knowledge, which may affect the accuracy of these methods. Convolution neural networks-based clustering methods have been used to regionalize earthquake epicenters or predict their precise hypocenter locations. In the latter case, three-component waveforms from multiple stations are exploited to train the model for swarm event localization.

In this study, we propose a RF-based method to locate earthquakes using the differential P-wave arrival times and station locations . The proposed algorithm only relies on P wave arrival times detected at the first few stations. Its prompt response to earthquake first arrivals is critical for rapidly disseminating EEW alerts. Our strategy implicitly considers the influence of the velocity structures by incorporating the source-station locations into the RF model. We evaluate the proposed algorithm using an extensive seismic catalog from Japan. Our test results show that the RF model is capable of determining the locations of earthquakes accurately with minimal information, which sheds new light on developing efficient machine learning.

**CHAPTER-2**

**LITERATURE SURVEY**

**2.1 RESEARCH ON EARTHQUAKE WARNING SYSTEM**

In the past twenty years, EEWSs have been implemented in different regions of the world and are considered a useful tool to reduce seismic risk ([Satriano et al., 2011b](https://www.frontiersin.org/articles/10.3389/feart.2021.685751/full" \l "B43)). EEWSs were developed with different approaches, methodologies, and combining new experiences. At present, many countries have operational or prototype EEWSs. [Allen et al. (2009b)](https://www.frontiersin.org/articles/10.3389/feart.2021.685751/full#B3) described the status of EEW in the world and the principal operating systems at that time. Other examples include EEWS in Japan ([Odaka et al., 2003](https://www.frontiersin.org/articles/10.3389/feart.2021.685751/full" \l "B36)), Taiwan ([Wu and Teng, 2002](https://www.frontiersin.org/articles/10.3389/feart.2021.685751/full#B47); [Hsiao et al., 2009](https://www.frontiersin.org/articles/10.3389/feart.2021.685751/full#B27)), Mexico ([Suarez et al., 2009](https://www.frontiersin.org/articles/10.3389/feart.2021.685751/full#B45)), Turkey ([Erdik et al., 2003](https://www.frontiersin.org/articles/10.3389/feart.2021.685751/full" \l "B23); [Alcik et al., 2009](https://www.frontiersin.org/articles/10.3389/feart.2021.685751/full" \l "B1)), and Romania ([Böse et al., 2007](https://www.frontiersin.org/articles/10.3389/feart.2021.685751/full" \l "B8)). The principal active systems are based on the software ElarmS ([Allen and Kanamori, 2003](https://www.frontiersin.org/articles/10.3389/feart.2021.685751/full#B4); [Allen et al., 2009a](https://www.frontiersin.org/articles/10.3389/feart.2021.685751/full#B2)) and ShakeAlert ([Kohler et al., 2020](https://www.frontiersin.org/articles/10.3389/feart.2021.685751/full#B31)) in California, on Virtual Seismologist in California and Switzerland ([Cua et al., 2009](https://www.frontiersin.org/articles/10.3389/feart.2021.685751/full" \l "B18)), in Europe ([Clinton et al., 2016](https://www.frontiersin.org/articles/10.3389/feart.2021.685751/full#B14)), and in particular PRESTo in Italy ([Iannaccone et al., 2010](https://www.frontiersin.org/articles/10.3389/feart.2021.685751/full" \l "B28); [Satriano et al., 2011a](https://www.frontiersin.org/articles/10.3389/feart.2021.685751/full" \l "B42)).

Major developments have led to two main types of systems: a regional alert system and an on-site system ([Satriano et al., 2011b](https://www.frontiersin.org/articles/10.3389/feart.2021.685751/full" \l "B43); [Zollo et al., 2014](https://www.frontiersin.org/articles/10.3389/feart.2021.685751/full" \l "B49)). The regional system, based on the use of a regional network that records seismic events, aims to detect, locate, and determine the magnitude of an event starting from the analysis of a few seconds of the first arrivals of the P waves recorded at the stations ([Picozzi, 2012](https://www.frontiersin.org/articles/10.3389/feart.2021.685751/full" \l "B37)). The on-site system consists of a single sensor or more sensors near or inside the target structure to be alerted. In this system the P-wave recordings to the sensor are used to predict the peak ground motion at the site ([Colombelli et al., 2015](https://www.frontiersin.org/articles/10.3389/feart.2021.685751/full" \l "B16)). This approach could be considered useful for sites located within the BZ of a regional EEW system, allowing for a useful warning before the arrival of strong shaking waves. [Caruso et al. (2017)](https://www.frontiersin.org/articles/10.3389/feart.2021.685751/full#B9) proposed a P-wave-based EEW approach called on-site alert level (SAVE). Many studies combined the two EEW approaches ([Zollo et al., 2010](https://www.frontiersin.org/articles/10.3389/feart.2021.685751/full" \l "B48); [Colombelli et al., 2012a](https://www.frontiersin.org/articles/10.3389/feart.2021.685751/full" \l "B15)); these systems combine local parameters and predicted ground motions at a regional scale to provide reliable and rapid estimates of the seismic source and the expected damage zone ([Colombelli et al., 2015](https://www.frontiersin.org/articles/10.3389/feart.2021.685751/full" \l "B16)). “point-source” (simply the source as a concentrated volume) or “finite fault” (a more sophisticated and realistic characterization of the source, considering the entire fault area). Most studies have used the “point-source” demonstrating the reliability of this approach for the magnitude estimation of small to moderate events. However, it has been shown that this approach is not always accurate for strong earthquakes (magnitude> 6.5–7), due to the saturation of the P-wave parameters. Several authors (for example, [Colombelli et al., 2012b](https://www.frontiersin.org/articles/10.3389/feart.2021.685751/full" \l "B17)) estimated the magnitude over time windows longer than the recorded P-wave and/or the S-wave signal to obtain more accurate final values. These magnitude calculations are reliable at the cost of requiring more data and time ([Velazquez et al., 2020](https://www.frontiersin.org/articles/10.3389/feart.2021.685751/full#B46)). In our study, the selected earthquakes have a moderate magnitude (≤ 6.5) and were considered as point sources. Potentially, an EEWS can produce and transmit alert messages to different end-users to allow them to adopt several types of safety measures in a few seconds. The main benefits of an EEWS include public warning, first responder mobilization, and safety of health care and utility infrastructures, transit systems, and workplaces ([Allen and Melgar, 2019](https://www.frontiersin.org/articles/10.3389/feart.2021.685751/full#B5)). Whereas in most cases evacuation of buildings is unrealistic, due to the short time available to act; a portion of the affected population can receive the alert and take safety measures in certain types of structures and infrastructures ([Iervolino et al., 2008](https://www.frontiersin.org/articles/10.3389/feart.2021.685751/full" \l "B29)).Receiving an alert message increases personal situational awareness and yields a more rapid response, especially in well-trained people who can take precautionary and protective actions like “Drop-Cover-Hold on”, suspending delicate medical procedures, or slowing down a train ride. In shaking areas, a time of 10 s allows people to protect themselves and prepare for evacuation ([Fujinawa and Noda, 2013](https://www.frontiersin.org/articles/10.3389/feart.2021.685751/full" \l "B25)). A time interval of 5–7 s could be enough to trigger automatic mitigation actions ([Cauzzi et al., 2016](https://www.frontiersin.org/articles/10.3389/feart.2021.685751/full" \l "B11)) at power plants, energy sector grids, and utilities infrastructures to prevent explosions, combustions, loss of water, flooding, fatal collisions, and elevator interruptions. Social studies have demonstrated that receiving “point-source” (simply the source as a concentrated volume) or “finite fault” (a more sophisticated and realistic characterization of the source, considering the entire fault area). Most studies have used the “point-source” demonstrating the reliability of this approach for the magnitude estimation of small to moderate events. However, it has been shown that this approach is not always accurate for strong earthquakes (magnitude> 6.5–7), due to the saturation of the P-wave parameters. Several authors (for example, [Colombelli et al., 2012b](https://www.frontiersin.org/articles/10.3389/feart.2021.685751/full" \l "B17)) estimated the magnitude over time windows longer than the recorded P-wave and/or the S-wave signal to obtain more accurate final values. These magnitude calculations are reliable at the cost of requiring more data and time ([Velazquez et al., 2020](https://www.frontiersin.org/articles/10.3389/feart.2021.685751/full#B46)). In our study, the selected earthquakes have a moderate magnitude (≤ 6.5) and were considered as point sources.Potentially, an EEWS can produce and transmit alert messages to different end-users to allow them to adopt several types of safety measures in a few seconds. The main benefits of an EEWS include public warning, first responder mobilization, and safety of health care and utility infrastructures, transit systems, and workplaces ([Allen and Melgar, 2019](https://www.frontiersin.org/articles/10.3389/feart.2021.685751/full#B5)). Whereas in most cases evacuation of buildings is unrealistic, due to the short time available to act; a portion of the affected population can receive the alert and take safety measures in certain types of structures and infrastructures ([Iervolino et al., 2008](https://www.frontiersin.org/articles/10.3389/feart.2021.685751/full" \l "B29)).Receiving an alert message increases personal situational awareness and yields a more rapid response, especially in well-trained people who can take precautionary and protective actions like “Drop-Cover-Hold on”, suspending delicate medical procedures, or slowing down a train ride. In shaking areas, a time of 10 s allows people to protect themselves and prepare for evacuation ([Fujinawa and Noda, 2013](https://www.frontiersin.org/articles/10.3389/feart.2021.685751/full" \l "B25)). A time interval of 5–7 s could be enough to trigger automatic mitigation actions ([Cauzzi et al., 2016](https://www.frontiersin.org/articles/10.3389/feart.2021.685751/full" \l "B11)) at power plants, energy sector grids, and utilities infrastructures to prevent explosions, combustions, loss of water, flooding, fatal collisions, and elevator interruptions. Social studies have demonstrated that receiving The approaches for regional EEW can be classified as EEW systems provide rapidly earthquake source information before the damaging S wave arrival [22]. The EEW system is currently being developed in a number of countries, including Indonesia. The EEW was developed by the Agency for Meteorology Climatology and Geophysics (BMKG). When this magnitude estimation study was constructed, the test phase of the EEW, namely Indonesia Earthquake Early Warning System, was conducted by BMKG (InaEEWS).

Indonesian Earthquake Early Warning System (InaEEWS) would be in 3 existing systems:

1. Earthquake sensor monitoring system which detects earthquake signals,
2. Automatic Processing System that processes data appropriately, and
3. Dissemination system for the dissemination of early warning earthquake information to relevant stakeholders and the public.

Earthquake early warning systems (EEW) within the Earthquake Sensor Monitoring System are crucial to a fast and precise determination of earthquake scale. The magnitude scale and trembling intensity scale determinations of the initial P wave are two main aspects of early early warning [23].

In every EEW analysis, researchers investigated whether an approximation of the high magnitude (Mw ≥ 6.5) of the earthquake and its location could be achieved using the fastest reaching P waves only. First, the concept of an early warning earthquake (EEW) suggested by J.D. Cooper in November 1868, the notion of an early warning earthquake for San Francisco, California after a M = 7 earthquake on Hayward's fault. If the detectors were triggered by an earthquake, an electrical pulse will be transmitted to San Francisco by telegraph. This warning would ring a large bell in the City Hall to alert people that an earthquake had happened [22].

On the basis of strong-motion data in Taiwan [23]–[25], Southern California [26] and Japan [27], the scaling relationship between magnitude and characteristic time following P wave was assessed. Taiwan, California and Japan have taken together data into account, see Kanamori [24], [28].

Numerous machine learning and prediction experiments on earthquake events had been performed by researchers, leading to various findings on the issues under consideration. Kundu et al., [16] did calculation of the approximate moment magnitude using the Artificial Neural Network system. Determination of the moment magnitude estimate (Mw) is done by taking seismic wave data on the Z (vertical) 1 station 3 components using ANN back propagation algorithm. The ANN method was trained with 62 earthquake events with a magnitude range of 6 - 9.1 Mw. The amplitude spectrum of the seismogram was used as input for the training with a P wave window 3 seconds from onset.

Another related study was the research of classification deep neural network, Li et al., see [9], the issue of wave discrimination in EEW has arisen from a mixture of generative adversarial networks (GANs) and Random Forests. The study was trained the GAN first so that it could identify and emit practical synthetic P waves. Then use the trained critic as an automated waveform function extractor and combine it with the Random Forest Classifier to be a P wave discriminator.

Wang et al., see [17] suggested a new method focused on deep learning for P-wave magnitude prediction (EEWNet), it was used regression type of deep learning and take time series data as input instead of function parameters. Only unfiltered vertical part accelerograms were included. Moderate number of data sets (10,000s of records) was used on EEWNet, However, it produces outstanding performance in the forecast of magnitude relative to the methods using the τlog, τc and Pd parameters.

Convolution neural networks-based clustering methods have been used to regionalize earthquake epicenters or predict their precise hypocenter locations. In the latter case, three-component waveforms from multiple stations are exploited to train the model for swarm event localization.

In this analysis, earthquake signals were obtained from the data archive of the Indonesian Meteorological, Climatological and Geophysical Agency (BMKG) for the period 2015-2020. BMKG supports the collection, data storage and dissemination of seismological data. Generally, when we get raw numbers, there are both earthquakes and plain recordings of some noise. Extracting the earthquake from the recording is important. There could be numerous earthquakes or aftershocks within 24 hours. Many of these earthquakes must be defined separately. https://geof.bmkg.go.id/webdc3/is a software interface for monitoring seismic activities and requesting an earthquake data. It allows the user to select the type of event and/ or the earthquake stations. We used data set that consists of 1825 recorded raw seismogram by single station seismogram, from 1825 signals we only use 604 with seismic moment magnitude type Mwp, Mwp is magnitude dependent on an approximation of moment measured from the integral of the displacement of the P wave recorded on broadband sensors [29]. The hypocentral distance between 5km and 653km. For P-wave arrival, each seismogram was marked. At station BLJI, time windows of 4 s lengths are cut beginning 1 s before the P-phase onset, and the data frequency sampling was 20Hz.

Machine learning in earthquake studies is developed in 5 main stages [7], [15], [30]–[32], namely:

* Seismic data collecting and partitioning that used for training and testing,
* Seismic data preprocessing for formatting and erasing / recovering seismic data and cleaning the data,
* The training model uses a numerical optimization algorithm to perfect the seismic variables,
* Evaluate the model using test results in terms of prediction accuracy, and
* Generate fresh data using the ML algorithm for prediction.

Unlike previous earthquake magnitude estimation models suggested in the literature, a comparison of the prediction model between classification and regression type is proposed in these report. The Random Forest, also known as bagging [35], is a group of decision trees that are learned by data bootstrapping and performance aggregation. Random Forests improve tree diversity by adding additional randomness, such as sub-sampling, as the decision tree is grown [36]. Decision trees are supervised tools for machine learning and have been used for the purposes of classification and regression [14]. By using ensemble learning methodologies, the efficiency of decision trees can be improved. These features make Random Forests among the most efficient algorithms available today for machine learning. Instead of the deep neural network itself, using the Random Forest as a classifier leads to a much higher classification. Simple decision trees typically train or memorize the data and random forests are one way of eliminating this attribute, thus reducing the variance of the final predictive model. This requires the estimation of multiple uncorrelated decision trees educated on the sampled dataset [35]. On the basis of tests on datasets, the choice of the number of decision trees is determined. In our case, on the basis of experimentation, 500 numbers of trees were selected for ensemble construction. In the Random forest model, for this analysis, we first use SMOTE to get the synthetic data. The mathematical model of the first raw data and the catalog is based on the imbalanced quantity. Data imbalance exists because, relative to other classes, the number of objects is a greater data class. The data class with more objects is considered the main class, while others are called minor classes. The effect of the use of unbalanced model effects data obtained in order to generate very large models. Irrespective of the data imbalance, algorithm processing tends to be overwhelmed by the main class and ignored by the minor class. The SMOTE approach was suggested by [37] methods for working with data are not matched by values that vary from the over- sampling method proposed above. When the principled oversampling procedure reproduces randomized observations, the SMOTE method increases the number of minor class data to be equal to the main class by producing artificial data. Models with SMOTE are more reliable since the resultant AUC is higher than the model without SMOTE. AUC was an output assessment method for classification problems to assess the threshold of the model. Artificial or syntetic data is based on the closest k-neighbor (k-nearest neighbor). Amount of closest k-neighbors determined to take into account the convenience of inside getting it out. The numerical generation of artificially scaled data is separate from the categorical generation. Numeric data is determined by its proximity to the Euclidean distance, while the categorical data is simpler with the mode value [38].

**CHAPTER-3**

**PROBLEM IDENTIFICATION**

**3.1 EXISTINGSYSTEM**

Earthquake early warning (EEW) systems are required to report earthquake locations and magnitudes as quickly as possible before the damaging S wave arrival to mitigate seismic hazards. Deep learning techniques provide potential for extracting earthquake source information from full seismic waveforms instead of seismic phase picks.

In Existing, developed a novel deep learning EEW system that utilizes fully convolutional networks to simultaneously detect earthquakes and estimate their source parameters from continuous seismic waveform streams. The system determines earthquake location and magnitude as soon as very few stations receive earthquake signals and evolutionarily improves the solutions by receiving continuous data. apply the system to the 2016 M 6.0 Central Apennines, Italy Earthquake and its first-week aftershocks. Earthquake locations and magnitudes can be reliably determined as early as 4 s after the earliest P phase, with mean error ranges of 8.5–4.7 km and 0.33–0.27, respectively.

**3.1.1 DISADVANTAGES OF EXISTING SYSTEM**

* An existing system method is not investigated to improve the performance of real-time earthquake detection and classification of source characteristics.
* Convolution neural networks-based clustering methods have not been used to regionalize earthquake epicenters or predict their precise hypocenter locations.

**3.2 PROPOSED SYSTEM**

The system proposes a RF-based method to locate earthquakes using the differential P-wave arrival times and station locations. The proposed algorithm only relies on Pwave arrival times detected at the first few stations. Its prompt response to earthquake first arrivals is critical for rapidly disseminating EEW alerts. Our strategy implicitly considers the influence of the velocity structures by incorporating the source-station locations into the RF model.

The proposed system evaluates the proposed algorithm using an extensive seismic catalog from Japan. Our test results show that the RF model is capable of determining the locations of earthquakes accurately with minimal information, which sheds new light on developing efficient machine learning.

**3.2.1 ADVANTAGES OF EXISTING SYSTEM**

* The number of stations is a critical factor that determines the data availability and prediction accuracy. The proposed RF model takes the arrival times of P waves recorded at multiple stations as the input, hence a more stringent requirement of simultaneous recording at an increased number of stations lowers the availability of qualified events.
* The localization problem can be resolved using a sequence of detected waves (arrival times) and locations of seismograph stations that are triggered by ground shaking. Among various network architectures, the recurrent neural network (RNN) is capable of precisely extracting information from a sequence of input data, which is ideal for handling a group of seismic stations that are triggered sequentially following the propagation paths of seismic waves.

**3.3 FEASIBILITY STUDY**

An important outcome of preliminary investigation is the determination that the system request is feasible. This is possible only if it is feasible within limited resource and time. The different feasibilities that have to be analyzed are

* **Operational Feasibility**
* **Economic Feasibility**
* **Technical Feasibility**

**3.3.1 Operational Feasibility**

Operational Feasibility deals with the study of prospects of the system to be developed. This system operationally eliminates all the tensions of the Admin and helps him in effectively tracking the project progress. This kind of automation will surely reduce the time and energy, which previously consumed in manual work. Based on the study, the system is proved to be operationally feasible.

**3.3.2 Economic Feasibility**

Economic Feasibility or Cost-benefit is an assessment of the economic justification for a computer based project. As hardware was installed from the beginning & for lots of purposes thus the cost on project of hardware is low. Since the system is a network based, any number of employees connected to the LAN within that organization can use this tool from at anytime. The Virtual Private Network is to be developed using the existing resources of the organization. So the project is economically feasible.

**3.3.3 Technical Feasibility**

According to Roger S. Pressman, Technical Feasibility is the assessment of the technical resources of the organization. The organization needs IBM compatible machines with a graphical web browser connected to the Internet and Intranet. The system is developed for platform Independent environment. Python, JavaScript, HTML, SQL server and WebLogic Server are used to develop the system. The technical feasibility has been carried out. The system is technically feasible for development and can be developed with the existing facility.

**3.4 HARDWARE AND SOFTWARE REQUIREMENTS**

**Software requirements**

* Operating System - Windows 10
* Server - XAMPP
* Front End - HTML, CSS, JS
* Back End - Python
* Data base - MYSQL
* IDE - Pycharm
* Framework - Django

**Hardware requirements**

* Processor- Intel (R) Core (TM) i3-4200U
* CPU - 1.6GHz
* RAM:4 GB
* Hard Disk: 500 GB.

**CHAPTER-4**

**SYSTEM DESIGN**

System design is therefore the process of defining and developing systems to satisfy specified requirements of the user. The basic study of system design is the understanding of component parts and their subsequent interaction with one another.

**4.1 UML DIAGRAMS**

The Unified Modelling Language (UML) is a standard language for specifying, visualizing, constructing, and documenting the artifacts of software systems, as well as for business modeling and other non-software systems. The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems. The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects. Using the UML helps project teams communicate, explore potential designs, and validate the architectural design of the software.

As the strategic value of software increases for many companies, the industry looks for techniques to automate the production of software and to improve quality and reduce cost and time-to-market. These techniques include component technology, visual programming, patterns and frameworks. Businesses also seek techniques to manage the complexity of systems as they increase in scope and scale. In particular, they recognize the need to solve recurring architectural problems, such as physical distribution, concurrency, replication, security, load balancing and fault tolerance. Additionally, the development for the World Wide Web, while making some things simpler, has exacerbated these architectural problems. The Unified Modeling Language(UML) was designed to respond to these needs. Simply, Systems design refers to the process of defining the architecture, components, modules, interfaces, and data for a system to satisfy specified requirements which can be done easily through UML diagrams.

**Object-Oriented Concepts**

UML can be described as the successor of object-oriented (OO) analysis and design.

An object contains both data and methods that control the data. The data represents the state of the object. A class describes an object and they also form a hierarchy to model the real-world system. The hierarchy is represented as inheritance and the classes can also be associated in different ways as per the requirement.

Objects are the real-world entities that exist around us and the basic concepts such as abstraction, encapsulation, inheritance, and polymorphism all can be represented using UML.

UML is powerful enough to represent all the concepts that exist in object-oriented analysis and design. UML diagrams are representation of object-oriented concepts only. Thus, before learning UML, it becomes important to understand OO concept in detail.

Following are some fundamental concepts of the object-oriented world −

* **Objects** − Objects represent an entity and the basic building block.
* **Class** − Class is the blue print of an object.
* **Abstraction** − Abstraction represents the behavior of an real world entity.
* **Encapsulation** − Encapsulation is the mechanism of binding the data together and hiding them from the outside world.
* **Inheritance** − Inheritance is the mechanism of making new classes from existing ones.
* **Polymorphism** − It defines the mechanism to exists in different forms.

**OO Analysis and Design**

OO can be defined as an investigation and to be more specific, it is the investigation of objects. Design means collaboration of identified objects.

Thus, it is important to understand the OO analysis and design concepts. The most important purpose of OO analysis is to identify objects of a system to be designed. This analysis is also done for an existing system. Now an efficient analysis is only possible when we are able to start thinking in a way where objects can be identified. After identifying the objects, their relationships are identified and finally the design is produced.

The purpose of OO analysis and design can described as −

* Identifying the objects of a system.
* Identifying their relationships.
* Making a design, which can be converted to executables using OO languages.

There are three basic steps where the OO concepts are applied and implemented. The steps can be defined as

OO Analysis → OO Design → OO implementation using OO languages

The above three points can be described in detail as −

* During OO analysis, the most important purpose is to identify objects and describe them in a proper way. If these objects are identified efficiently, then the next job of design is easy. The objects should be identified with responsibilities. Responsibilities are the functions performed by the object. Each and every object has some type of responsibilities to be performed. When these responsibilities are collaborated, the purpose of the system is fulfilled.
* The second phase is OO design. During this phase, emphasis is placed on the requirements and their fulfilment. In this stage, the objects are collaborated according to their intended association. After the association is complete, the design is also complete.
* The third phase is OO implementation. In this phase, the design is implemented using OO languages such as Java, C++, etc.

**Role of UML in OO Design**

UML is a modeling language used to model software and non-software systems. Although UML is used for non-software systems, the emphasis is on modeling OO software applications. Most of the UML diagrams discussed so far are used to model different aspects such as static, dynamic, etc. Now whatever be the aspect, the artifacts are nothing but objects.

Hence, the relation between OO design and UML is very important to understand. The OO design is transformed into UML diagrams according to the requirement.

In this project ,basic UML diagrams have been explained

1. Use Case Diagram
2. Class Diagram
3. Sequence Diagram
4. Collaboration Diagram
5. Activity Diagram
6. Deployment Diagram

**4.1.1 CLASS DIAGRAM**

UML class diagrams model static class relationships that represent the fundamental architecture of the system. Note that these diagrams describe the relationships between classes, not those between specific objects instantiated from those classes. Thus the diagram applies to all the objects in the system.

A class diagram consists of the following features:

* **Classes:** These titled boxes represent the classes in the system and contain information about the name of the class, fields, methods and access specifies. Abstract roles of the Class in the system can also be indicated
* **Interfaces:** These titled boxes represent interfaces in the system and contain information about the name of the interface and its methods. Relationship Lines that model the relationships between classes and interfaces in the system.
* **Dependency:** A dotted line with an open arrowhead that shows one entity depends on the behavior of another entity. Typical usages are to represent that one class instantiates another or that it uses the other as an input parameter
* **Aggregation:** Represented by an association line with a hollow diamond at the tail end. An aggregation models the notion that one object uses another object without "owning" it and thus is not responsible for its creation or destruction.
* **Inheritance:** A solid line with a solid arrowhead that points from a sub-class to a super class or from a sub-interface to its super-interface.
* **Implementation:** A dotted line with a solid arrowhead that points from a class to the interface that it implement
* **Composition:** Represented by an association line with a solid diamond at the tail end. A composition models the notion of one object "owning" another and thus being responsible for the creation and destruction of another object.

****

**Fig 4.1.1 Class Diagram**

**4.1.2 USECASE DIAGRAM**

A use case diagram in the Unified Modelling Language (UML) is a type of behavioural diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms.

A use case is a methodology used in system analysis to identify, clarify, and organize system requirements. The use case is made up of a set of possible sequences of interactions between systems and users in a particular environment and related to a particular goal. It consists of a group of elements (for example, classes and interfaces) that can be used together in a way that will have an effect larger than the sum of the separate elements combined. The use case should contain all system activities that have significance to the users. A use case can be thought of as a collection of possible scenarios related to a particular goal, indeed, the use case and goal are sometimes considered to be synonymous.The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

**Parts of Use Case Diagram**

**System boundary boxes (optional)**

A rectangle is drawn around the use cases, called the system boundary box, to indicate the scope of system. Anything within the box represents functionality that is in scope and anything outside the box is not **Relationships.**

**Include**

In one form of interaction, a given use case may include another. "Include is a Directed Relationship between two use cases, implying that the behavior of the included use case is inserted into the behavior of the including use case”.

The first use case often depends on the outcome of the included use case. This is useful for extracting truly common behaviours from multiple use cases into a single description. The notation is a dashed arrow from the including to the included use case, with the label "«include»". This usage resembles a macro expansion where the included use case behavior is placed inline in the base use case behavior. There are no parameters or return values. To specify the location in a flow of events in which the base use case includes the behavior of another, you simply write include followed by the name of use case you want to include, as in the following flow for track order.

**Extend**

In another form of interaction, a given use case (the extension) may extend another. This relationship indicates that the behavior of the extension use case may be inserted in the extended use case under some conditions. The notation is a dashed arrow from the extension to the extended use case, with the label "«extend»". The notes or constraints may be associated with this relationship to illustrate the conditions under which this behavior will be executed. Modelers use the «extend» relationship to indicate use cases that are "optional" to the base use case. Depending on the modeler’s approach "optional" may mean "potentially not executed with the base use case" or it may mean "not required to achieve the base use case goal".

**4.1.2.1 Use Case Diagram for Remote User**

Remote Users

**Fig 4.1.2.1 Use Case Diagram for Remote User**

**4.1.2.2 Use case Diagram for Service Provider**

Service Provider

**Fig 4.1.2.2 Use case Diagram for Service Provider**

**4.1.3 SEQUENCE DIAGRAM**

A sequence diagram in Unified Modelling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. A Sequence diagram depicts the sequence of actions that occur in a system. The invocation of methods in each object, and the order in which the invocation occurs is captured in a Sequence diagram. This makes the Sequence diagram a very useful tool to easily represent the dynamic behavior of a system.

**Elements of sequence diagram**

The sequence diagram is an element that is used primarily to showcase the interaction that occurs between multiple objects. This interaction will be shown over certain period of time. Because of this, the first symbol that is used is one that symbolizes the object.

**Lifeline**

A lifeline will generally be generated, and it is a dashed line that sits vertically, and the top will be in the form of a rectangle. This rectangle is used to indicate both the instance and the class. If the lifeline must be used to denote an object, it will be underlined.

**Messages**

To showcase an interaction, messages will be used. These messages will come in the form of horizontal arrows, and the messages should be written on top of the arrows. If the arrow has a full head, and it’s solid, it will be called a synchronous call. If the solid arrow has a stick head, it will be an asynchronous call. Stick heads with dash arrows are used to represent return messages.

**Objects**

Objects will also be given the ability to call methods upon themselves, and they can add net activation boxes. Because of this, they can communicate with others to show multiple levels of processing. Whenever an object is eradicated or erased from memory, the "X" will be drawn at the lifeline's top, and the dash line will not be drawn beneath it.

This will often occur as a result of a message. If a message is sent from the outside of the diagram, it can be used to define a message that comes from a circle that is filled in. Within a UML based model, a Super step is a collection of steps which result from outside stimuli.

**Steps to Create Sequence Diagram**

* + Set the stage for the interaction by identifying which objects play a role in interaction.
  + Set the lifetime for each object.



**Fig 4.1.3 Sequence Diagram**

**4.1.4 ACTIVITY DIAGRAM**

diagram is another important diagram in UML to describe dynamic aspects of the system. Activity diagram is basically a flow chart to represent the flow form one activity to another activity. The activity can be described as an operation of the system. So the control flow is drawn from one operation to another. This flow can be sequential, branched or concurrent. Activity diagrams deals with all type of flow control by using different elements like fork, join etc.

**How to draw Activity Diagram?**

Activity diagrams are mainly used as a flow chart consists of activities performed by the system. But activity diagram are not exactly a flow chart as they have some additional capabilities. These additional capabilities include branching, parallel flow, swim lane etc. Before drawing an activity diagram we must have a clear understanding about the elements used in activity diagram. The main element of an activity diagram is the activity itself. An activity is a function performed by the system. After identifying the activities we need to understand how they are associated with constraints and conditions. So before drawing an activity diagram we should identify the following elements.

* Activities
* Association
* Conditions
* Constraints

The following are the basic notational elements that can be used to make up a diagram:

**Initial state**

An initial state represents a default vertex that is the source for a single transition to the default state of a composite state. There can be at most one initial vertex in a region. The outgoing transition from the initial vertex may have a behavior, but not a trigger or guard. It is represented by Filled circle, pointing to the initial state.

**Final state**

A special kind of state signifying that the enclosing region is completed. If the enclosing region is directly contained in a state machine and all other regions in the state machine also are completed, then it means that the entire state machine is completed. It is represented by Hollow circle containing a smaller filled circle, indicating the final state.

**Rounded rectangle**

It denotes a state. Top of the rectangle contains a name of the state. Can contain a horizontal line in the middle, below which the activities that are done in that state are indicated.

**Arrow**

It denotes transition. The name of the event (if any) causing this transition labels the arrow body.

**Steps To Construct Activity Diagram**

* Identify the preconditions of the workflow
* Collect the abstractions that are involved in the operations

**4.1.4.1 Activity diagram for Remote User**

Register & Login

Check

Valid

Invalid

Predict earthquake early warning type

View Your profile

Log out

**Fig 4.1.4.1 Activity diagram For Remote User**

**4.1.4.2 Activity Diagram for Service Provider**

Login

Check

Valid

Invalid

Train & Test Data Sets

View Trained & tested Accuracy in Bar chart

View Trained & Tested Accuracy results

View Prediction of earthquake early type warning

View Earthquake early warning type ratio

Download predicted data sets

Log out

View all Remote Users

View earthquake early warning type ratio results

**Fig 4.1.4.2 Activity Diagram for Service Provider**

**4.1.5 DEPLOYMENT DIAGRAM**

Deployment diagram represents the deployment view of a system .It is related to the Component diagram. Because the components are deployed using the deployment diagrams. A deployment diagram consists of nodes. Nodes are nothing but physical Hardware’s used to deploy the Applications.



**Fig 4.1.5 Deployment Diagram**

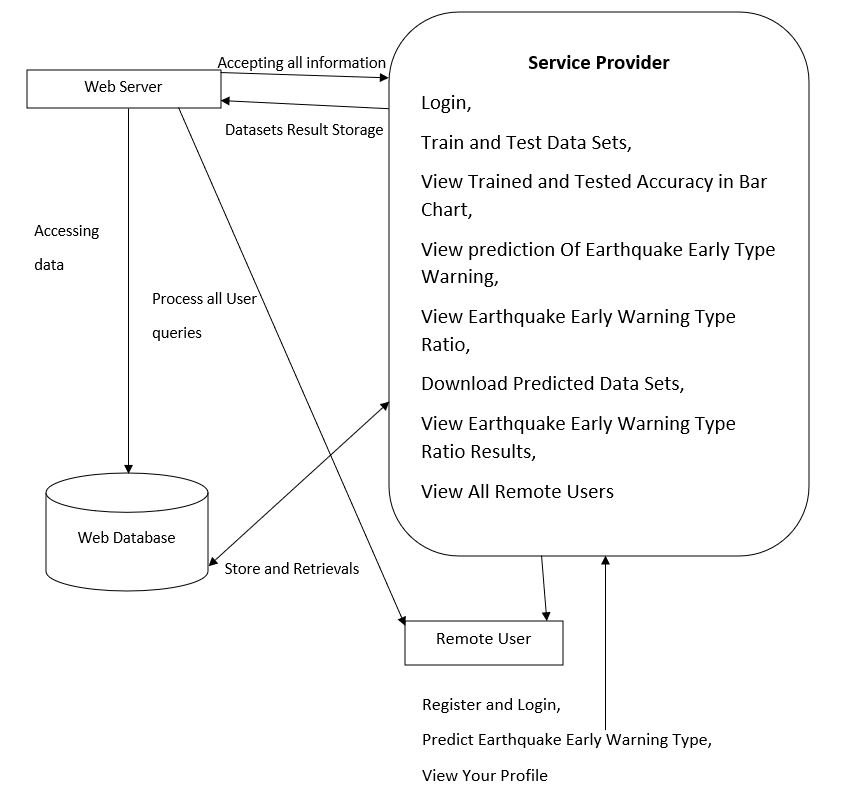
**4.1.6 COLLABORATION DIAGRAM**

A collaboration diagram, also known as a communication diagram, is an illustration of the relationships and interactions among software [objects](https://searchapparchitecture.techtarget.com/definition/object) in the Unified Modeling Language ([UML](https://searchsoftwarequality.techtarget.com/definition/Unified-Modeling-Language)). These diagrams can be used to portray the dynamic behavior of a particular [use case](https://searchsoftwarequality.techtarget.com/definition/use-case) and define the role of each object.



**Fig 4.1.6 Collaboration Diagram**

**4.1.7 ARCHITECTURE**

****

**Fig 4.1.7 Architecture**

**CHAPTER-5**

**SYSTEM IMPLEMENTATION**

System implementation is a set of procedures performed to complete the design (as necessary) contained in the approved systems design document and to test, install, and begin to use the new or revised Information System.

**5.1 IMPLEMENTATION PROCESS**

1. In this research work with data with attributes are observable and then all of them are floating data. And there’s a decision class/class variable. This data was collected from UCI machine learning repository.
2. In this research 70% data use for train model and 30% data use for testing purpose.
3. Random Forest is used as Classifier .
4. In the classification report we were able to find out the desired result
5. In this analysis the result depends on some part of this research. However, which algorithm gives the best true positive, false positive, true negative, and false negative are the best algorithms in this analysis.

**5.2 MODULES**

**Service Provider**

In this module, the Service Provider has to login by using valid user name and password. After login successful he can do some operations such as Login, Train & Test Data Sets, View Trained and Tested Accuracy in Bar Chart, View Trained and Tested Accuracy Results, View Prediction Of Earthquake Early Type Warning, View Earthquake Early Warning Type Ratio, Download Predicted Data Sets, View Earthquake Early Warning Type Ratio Results, View All Remote Users.

**Remote User**

In this module, there are n numbers of users are present. User should register before doing any operations. Once user registers, their details will be stored to the database. After registration successful, he has to login by using authorized user name and password. Once Login is successful user will do some operations like Register And Login, Predict Earthquake early Warning Type, View Your Profile.

**CHAPTER-6**

**TESTING**

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

**TYPES OF TESTS**

**Unit testing**

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

**Integration testing**

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

**Functional testing**

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centred on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive

Processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

**System Testing**

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

**White Box Testing**

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

**Black Box Testing**

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

**6.1 UNIT TESTING:**

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

**6.1.1 Test strategy and approach**

Field testing will be performed manually and functional tests will be written in detail.

**6.1.2 Test objectives**

* All field entries must work properly.
* Pages must be activated from the identified link.
* The entry screen, messages and responses must not be delayed.

**6.1.3 Features to be tested**

* Verify that the entries are of the correct format
* No duplicate entries should be allowed
* All links should take the user to the correct page.

**6.2 INTEGRATION TESTING**

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

**Test Results:** All the test cases mentioned above passed successfully. No defects encountered.

**6.3 SYSTEM TESTING**

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

**Test Results:** All the test cases mentioned above passed successfully. No defects encountered.

**6.4 ACCEPTANCE TESTING**

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

**Test Results:** All the test cases mentioned above passed successfully. No defects encountered.

**6.5 TEST CASES**

**6.5.1 Test Case-1 :** Service Provider Home Page

**Expected Output :** Service Provider Home Page is successful and Service Provider main menu is displayed

**Actual Output :** When Service Provider username and Password is entered into login page. On Successful credentials, Service Provider home page is displayed and main menu is displayed in Service Provider home page

**6.5.2 Test Case-2 :** User Registration

**Expected Output :** User Registration is successful

**Actual Output :** User Registration is failed because in registration page , Instead on giving the email format, the other format is entered. As the email format does not matches. It returns user registration is not successful

**CHAPTER-7**

**SAMPLE SOURCE CODE**

**Register.html**

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

{% load static %}

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<title>Register</title>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<style type="text/css">

<!--

.style5 {

font-size: 24px;

color: #FF0000;

}

.style12 {font-weight: bold}

.style13 {font-size: 24px; color: #FF0000; font-weight: bold; }

.style19 {color: #FFFF00; font-weight: bold; }

.style22 {color: #FF0000; font-weight: bold; }

.style23 {

color: #FFFFFF;

font-weight: bold;

}

.style24 {color: #FFFF00}

-->

</style>

</head>

<body>

<div class="main">

<div class="header">

<div class="header\_resize">

<div class="menu\_nav">

<p>&nbsp;</p>

</div>

<div class="mainbar">

<h1 align="center"><a href="index.html"><span class="content style5">Machine learning for fast and reliable source-location estimation in earthquake early warning</span></a></h1>

<div class="tab-content tab-space">

<div class="tab-pane active" id="preview-alerts">

<link href="https://fonts.googleapis.com/css?family=Open+Sans:300,400,600,700" rel="stylesheet" />

<script src="https://kit.fontawesome.com/42d5adcbca.js" crossorigin="anonymous"></script><link href="https://unpkg.com/soft-ui-design-system@1.0.1/assets/css/soft-design-system.min.css" rel="stylesheet" /><div class="container py-5">

<div class="row">

<div class="alert alert-primary text-white font-weight-bold" role="alert">

<p align="center"><span class="active"><span class="style12"><a href="{% url 'index' %}">Home| </a><a href="{% url 'login' %}">Remote User </a>|<a href="{% url 'serviceproviderlogin' %}"> Service Provider </a></span></span></p>

</div>

<div>

</div>

<img src="{% static 'Banner.jpg'%}" width="1297" height="327" alt="" class="fl" /> </div>

<div class="clr"></div>

<div class="slider">

</div>

<div class="clr"></div>

</div>

</div>

<div class="content">

<div class="content\_resize">

<div class="mainbar">

<div class="article">

<h2 align="center" class="style13"> Earthquake Early Warning (EEW) system;Machine learning; Earthquake Location..</h2>

<div class="img">

<div align="center">

<div class="container-fluid">

<div>

<div class="row">

<div >

<form role="form" method="POST" >

{% csrf\_token %}

<fieldset>

<p>{% load static %}

<img src="{% static '/Register.jpg' %}" alt="My image"><br>

<span class="style1 style22">REGISTER YOUR DETAILS HERE !!!</span> </p>

<table width="823" border="0" align="center">

<tr>

<td width="161" bgcolor="#FF0000"><span class="style19">Enter Username </span></td>

<td width="174"><span class="form-group">

<input type="text" name="username" id="username" placeholder="User Name" required="required" />

</span></td>

<td width="202" bgcolor="#FF0000"><span class="style19">Enter Password </span></td>

<td width="268"><span class="form-group">

<input type="password" name="password" id="password" placeholder="Password" required="required" />

</span></td>

</tr>

<tr>

<td height="47" bgcolor="#FF0000"><span class="style19">Enter EMail Id </span></td>

<td><span class="form-group">

<input type="email" name="email" id="email" placeholder="Enter Email " required="required" />

</span></td>

<td bgcolor="#FF0000"><span class="style19">Enter Address </span></td>

<td><span class="form-group">

<textarea name="address" placeholder="Enter Address "></textarea>

</span></td>

</tr>

<tr>

<td height="37" bgcolor="#FF0000"><span class="style19">Enter Gender </span></td>

<td><select name="gender">

<option>----Select Gender ----</option>

<option>Male</option>

<option>Female</option>

</select> </td>

<td bgcolor="#FF0000"><span class="style19">Enter Mobile Number </span></td>

<td><span class="form-group">

<input type="number" name="phoneno" id="phoneno" placeholder="Enter Mobile Number" required="required" />

</span></td>

</tr>

<tr>

<td bgcolor="#FF0000"><span class="style19">Enter Country Name </span></td>

<td><span class="form-group">

<input type="text" name="country" id="country" placeholder="Enter Country Name" required="required" />

</span></td>

<td bgcolor="#FF0000"><span class="style19">Enter State Name </span></td>

<td><span class="form-group">

<input type="text" name="state" id="state" placeholder="Enter State Name" required="required" />

</span></td>

</tr>

<tr>

<td bgcolor="#FF0000"><span class="style19">Enter City Name </span></td>

<td><span class="form-group">

<input type="text" name="city" id="city" placeholder="Enter City Name" required="required" />

</span></td>

<td bgcolor="#FF0000"><span class="style24"></span></td>

<td><input type="submit" class="btn btn-lg btn-primary" name="submit" value="Register" /></td>

</tr>

</table>

</fieldset>

<div><br>

<button class="btn btn-lg ">

</form>

</div>

</div>

<table border="0" align="left" >

<tr><td width="246" bgcolor="#FF0000"><div align="center" class="style23"><span class="style6">Registered Status</span> :: {{object}}</div></td>

</table>

<div class="mainbar">

<div class="tab-content tab-space">

<div class="tab-pane active" id="preview-alerts">

<link href="https://fonts.googleapis.com/css?family=Open+Sans:300,400,600,700" rel="stylesheet" />

<script src="https://kit.fontawesome.com/42d5adcbca.js" crossorigin="anonymous"></script><link href="https://unpkg.com/soft-ui-design-system@1.0.1/assets/css/soft-design-system.min.css" rel="stylesheet" /><div class="container py-5">

<div class="row">

<div class="alert alert-primary text-white font-weight-bold" role="alert">

<p align="center"><span class="active"><span class="style12"><a href="{% url 'login' %}">Home| </a><a href="{% url 'login' %}">Remote User </a>|<a href="{% url 'serviceproviderlogin' %}"> Service Provider </a></span></span></p>

</div>

<div>

</div>

<div class="col-md-2">

<!-------null------>

</div>

<div class="col-md-5"> </div>

</div>

</div>

</div>

</div>

</div>

<div class="clr"></div>

</div>

</div>

<div class="sidebar">

<div class="searchform"></div>

<div class="clr"></div>

</div>

<div class="clr"></div>

</div>

</div>

<div class="fbg"></div>

<div class="footer"></div>

</div>

<div align=center></div>

</body>

</html>

**Login.html**

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

{% load static %}

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<title>Login</title>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<link rel="stylesheet" type="text/css" href="{% static 'style.css'%} " />

<link rel="stylesheet" type="text/css" href="{% static 'coin-slider.css'%}" />

<script type="text/javascript" src="{% static 'cufon-yui.js'%} "></script>

<script type="text/javascript" src="{% static 'cufon-aller.js'%}"></script>

<script type="text/javascript" src="{% static 'jquery-1.4.2.min.js'%}"></script>

<script type="text/javascript" src="{% static 'script.js'%}"></script>

<script type="text/javascript" src="{% static 'coin-slider.min.js'%}"></script>

<style type="text/css">

<!--

.style5 {

font-size: 24px;

color: #FF0000;

}

.style12 {font-weight: bold}

.style13 {font-size: 24px; color: #FF0000; font-weight: bold; }

.style16 {color: #FF0000}

.style20 {color: #FF0000; font-weight: bold; }

-->

</style>

</head>

<body>

<div class="main">

<div class="header">

<div class="header\_resize">

<div class="menu\_nav">

<p>&nbsp;</p>

</div>

<div class="mainbar">

<h1 align="center"><a href="index.html"><span class="content style5">Machine learning for fast and reliable source-location estimation in earthquake early warning</span></a></h1>

<div class="tab-content tab-space">

<div class="tab-pane active" id="preview-alerts">

<link href="https://fonts.googleapis.com/css?family=Open+Sans:300,400,600,700" rel="stylesheet" />

<script src="https://kit.fontawesome.com/42d5adcbca.js" crossorigin="anonymous"></script><link href="https://unpkg.com/soft-ui-design-system@1.0.1/assets/css/soft-design-system.min.css" rel="stylesheet" /><div class="container py-5">

<div class="row">

<div class="alert alert-primary text-white font-weight-bold" role="alert">

<p align="center"><span class="active"><span class="style12"><a href="{% url 'index' %}">Home| </a><a href="{% url 'login' %}">Remote User </a>|<a href="{% url 'serviceproviderlogin' %}"> Service Provider </a></span></span></p>

</div>

<div>

</div>

<img src="{% static 'Banner.jpg'%}" width="1297" height="355" alt="" class="fl" /> </div>

<div class="clr"></div>

<div class="slider">

</div>

<div class="clr"></div>

</div>

</div>

<div class="content">

<div class="content\_resize">

<div class="mainbar">

<div class="article">

<h2 align="center" class="style13"> Earthquake Early Warning (EEW) system;Machine learning; Earthquake Location..</h2>

<div class="img">

<div align="center"><form method="POST" role="form">

{% csrf\_token %}

<fieldset>

{% load static %}

<img src="{% static '/login.jpg' %}" alt="My image">

<p class="style5"> Login Using Your Account: </p>

<div class="form-group">

<input type="text" name="username" placeholder="User Name" required>

<br />

<br />

</div>

<div class="form-group">

<input type="password" name="password" placeholder="Password" required>

</div>

<div>

<p>

<input type="submit" name="submit1" class="btn btn-md" value="Login">

<br />

</p>

</div>

<div>

<button>

<span class="style16"><strong>Are You New User !!</strong></span><span class="style20">! </span><a href="{% url 'Register1' %}">REGISTER</a></button>

</div>

</fieldset>

</form></div>

</div>

<div class="mainbar">

<div class="tab-content tab-space">

<div class="tab-pane active" id="preview-alerts">

<link href="https://fonts.googleapis.com/css?family=Open+Sans:300,400,600,700" rel="stylesheet" />

<script src="https://kit.fontawesome.com/42d5adcbca.js" crossorigin="anonymous"></script><link href="https://unpkg.com/soft-ui-design-system@1.0.1/assets/css/soft-design-system.min.css" rel="stylesheet" /><div class="container py-5">

<div class="row">

<div class="alert alert-primary text-white font-weight-bold" role="alert">

<p align="center"><span class="active"><span class="style12"><a href="{% url 'login' %}">Home| </a><a href="{% url 'login' %}">Remote User </a>|<a href="{% url 'serviceproviderlogin' %}"> Service Provider </a></span></span></p>

</div>

<div>

</div>

<div class="post\_content"></div>

<div class="clr"></div>

</div>

</div>

<div class="sidebar">

<div class="searchform"></div>

<div class="clr"></div>

</div>

<div class="clr"></div>

</div>

</div>

<div class="fbg"></div>

<div class="footer"></div>

</div>

<div align=center></div>

</body>

</html>

**Header.html**

<!DOCTYPE html>

{% load static %}

<html lang="en">

<body>

<head>

<meta charset="UTF-8">

<title>Remote User</title>

<link href="https://fonts.googleapis.com/css?family=Russo+One" rel="stylesheet">

<style>

body{background: url("{% static 'bg.jpg' %}");

background-size:cover;

font-family: 'Russo One', sans-serif;

background-color: #000000;

}

h1{

color:white;

}

.topnav {

overflow: hidden;

background-color: #812;

}

.topnav a {

float: left;

color: #FFFFFF;

text-align: center;

padding: 14px 16px;

text-decoration: none;

font-size: 17px;

}

.topnav a:hover {

background-color: #ddd;

color: black;

}

.topnav a.active {

background-color: #8e4fd1;

color: white;

}

.style1 {color: #FF0000}

</style>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8"></head>

<body>

<h1 align="center" class="style1">Machine learning for fast and reliable source-location estimation in earthquake early warning</h1>

<div class="tab-content tab-space">

<div class="tab-pane active" id="preview-alerts">

<link href="https://fonts.googleapis.com/css?family=Open+Sans:300,500,500,200" rel="stylesheet" />

<script src="https://kit.fontawesome.com/42d5adcbca.js" crossorigin="anonymous"></script>

<link href="https://unpkg.com/soft-ui-design-system@1.0.1/assets/css/soft-design-system.min.css" rel="stylesheet" /><div class="container py-5">

<div class="alert alert-success text-white font-weight-bold" role="alert">

<a href="{% url 'Predict\_Earthquake\_Early\_Warning\_Prediction' %}">PREDICT EARTHQUAKE EARLY WARNING TYPE</a>||

<a href="{% url 'ViewYourProfile' %}">VIEW YOUR PROFILE</a>||

<a href="{% url 'index' %}">LOGOUT</a>

</div>

</div>

</div>

<div class="mainholder">

{% block userblock %}

{% endblock %}

</div>

</body>

</html>

**Predict\_EarthQuake.html**

{% extends 'RUser/Header.html' %}

{% block userblock %}

<link rel="icon" href="images/icon.png" type="image/x-icon" />

<link href="https://fonts.googleapis.com/css?family=Lobster" rel="stylesheet">

<link href="https://fonts.googleapis.com/css?family=Righteous" rel="stylesheet">

<link href="https://fonts.googleapis.com/css?family=Fredoka+One" rel="stylesheet">

<style>

body {background-color:#000000;}

.container-fluid {padding:50px;}

.container{background-color:white;padding:50px; }

#title{font-family: 'Fredoka One', cursive;

}

.text-uppercase{

font-family: 'Righteous', cursive;

}

.tweettext{

border: 2px solid yellowgreen;

width: 904px;

height: 202px;

overflow: scroll;

background-color:;

}

.style1 {

color: #FF0000;

font-weight: bold;

}

.style4 {color: #FFFF00; font-weight: bold; }

.style7 {color: #FFFFFF}

.style9 {color: #FFFFFF; font-weight: bold; }

.style10 {font-size: 18px}

.style12 {

font-size: 24px;

color: #FFFFFF;

font-weight: bold;

}

</style>

<body>

<div class="container-fluid">

<div class="container">

<div class="row">

<div class="col-md-5">

<form role="form" method="POST" >

{% csrf\_token %}

<fieldset>

<p class="text-uppercase pull-center style1">PREDICTION OF EARTHQUAKE EARLY WARNING !!! </p>

<hr>

{% csrf\_token %}

<table width="989" align="center">

<tr>

<td height="44" colspan="4" bgcolor="#FF0000"><div align="center" class="style12">Enter Datasets Details Here !!! </div></td>

</tr>

<tr>

<td width="289" height="44" bgcolor="#FF0000"><div align="center" class="style4">Enter ewtime</div></td>

<td width="176"><input type="text" name="ewtime"></td>

<td width="364" bgcolor="#FF0000"><div align="center"><span class="style4">Enter latitude</span></div></td>

<td width="273"><input type="text" name="latitude"></td>

</tr>

<tr>

<td height="44" bgcolor="#FF0000"><div align="center" class="style4">Enter longitude</div></td>

<td><input type="text" name="longitude"></td>

<td bgcolor="#FF0000"><div align="center"><span class="style4">Enter depth</span></div></td>

<td><input type="text" name="depth"></td>

</tr>

<tr>

<td height="44" bgcolor="#FF0000"><div align="center" class="style4">Enter mag</div></td>

<td><input type="text" name="mag"></td>

<td bgcolor="#FF0000"><div align="center"><span class="style4">Enter magType</span></div></td>

<td><input type="text" name="magType"></td>

</tr>

<tr>

<td height="44" bgcolor="#FF0000"><div align="center" class="style4">Enter nst</div></td>

<td><input type="text" name="nst"></td>

<td bgcolor="#FF0000"><div align="center"><span class="style4">Enter gap</span></div></td>

<td><input type="text" name="gap"></td>

</tr>

<tr>

<td height="44" bgcolor="#FF0000"><div align="center" class="style4">Enter dmin </div></td>

<td><input type="text" name="dmin"></td>

<td bgcolor="#FF0000"><div align="center"><span class="style4">Enter rms</span></div></td>

<td><input type="text" name="rms"></td>

</tr>

<tr>

<td height="44" bgcolor="#FF0000"><div align="center" class="style4">

<div align="center">Enter net</div>

</div></td>

<td><input type="text" name="net"></td>

<td bgcolor="#FF0000"><div align="center"><span class="style4">Enter idn</span></div></td>

<td><input type="text" name="idn"></td>

</tr>

<tr>

<td height="44" bgcolor="#FF0000"><div align="center" class="style4">

<div align="center">Enter updated Time </div>

</div></td>

<td><input type="text" name="updated"></td>

<td bgcolor="#FF0000"><div align="center"><span class="style4">Enter place</span></div></td>

<td><textarea name="place" cols="25" rows="3"></textarea></td>

</tr>

<tr>

<td height="44" bgcolor="#FF0000"><div align="center" class="style4">Enter horizontalError</div></td>

<td><input type="text" name="horizontalError"></td>

<td bgcolor="#FF0000"><div align="center"><span class="style4">Enter depthError</span></div></td>

<td><input type="text" name="depthError"></td>

</tr>

<tr>

<td height="44" bgcolor="#FF0000"><div align="center" class="style4"><span class="style2">Enter </span>magError</div></td>

<td><input type="text" name="magError"></td>

<td bgcolor="#FF0000"><div align="center"><span class="style4">Enter magNst</span></div></td>

<td><input type="text" name="magNst"></td>

</tr>

<tr>

<td height="44" bgcolor="#FFFFFF">&nbsp;</td>

<td><input name="submit" type="submit" class="style1" value="Predict"></td>

<td>&nbsp;</td>

<td>&nbsp;</td>

</tr>

</table>

</fieldset>

</form>

<form role="form" method="POST" >

{% csrf\_token %}

<fieldset>

<hr>

<div>

<table width="769" height="61" border="0" align="center" >

<tr><td width="425" bgcolor="#FF0000"><div align="center" class="style10"><span class="text-uppercase pull-center style1 style7">Prediction Of Earthquake Early Warning</span> <span class="style9">::</span> </div></td>

<td width="265" bgcolor="#FFFFFF" style="color:red; font-size:20px; font-family:fantasy" ><div align="center">{{objs}}</div></td></tr>

</table>

</div>

</fieldset>

</form>

</div>

<div class="col-md-2">

<!-------null------>

</div>

</div>

</div>

</div>

{% endblock %}

<tr>

**Views.py**

from django.db.models import Count

from django.db.models import Q

from django.shortcuts import render, redirect, get\_object\_or\_404

import pandas as pd

from sklearn.feature\_extraction.text import CountVectorizer

from sklearn.metrics import accuracy\_score, confusion\_matrix, classification\_report

from sklearn.metrics import accuracy\_score

from sklearn.tree import DecisionTreeClassifier

from sklearn.ensemble import VotingClassifier

# Create your views here.

from Remote\_User.models import ClientRegister\_Model,earthquake\_early\_warning\_prediction,detection\_accuracy

def login(request):

if request.method == "POST" and 'submit1' in request.POST:

username = request.POST.get('username')

password = request.POST.get('password')

try:

enter = ClientRegister\_Model.objects.get(username=username,password=password)

request.session["userid"] = enter.id

return redirect('ViewYourProfile')

except:

pass

return render(request,'RUser/login.html')

def index(request):

return render(request, 'RUser/index.html')

def Add\_DataSet\_Details(request):

return render(request, 'RUser/Add\_DataSet\_Details.html', {"excel\_data": ''})

def Register1(request):

if request.method == "POST":

username = request.POST.get('username')

email = request.POST.get('email')

password = request.POST.get('password')

phoneno = request.POST.get('phoneno')

country = request.POST.get('country')

state = request.POST.get('state')

city = request.POST.get('city')

address = request.POST.get('address')

gender = request.POST.get('gender')

ClientRegister\_Model.objects.create(username=username, email=email, password=password, phoneno=phoneno,

country=country, state=state, city=city,address=address,gender=gender)

obj = "Registered Successfully"

return render(request, 'RUser/Register1.html',{'object':obj})

else:

return render(request,'RUser/Register1.html')

def ViewYourProfile(request):

userid = request.session['userid']

obj = ClientRegister\_Model.objects.get(id= userid)

return render(request,'RUser/ViewYourProfile.html',{'object':obj})

def Predict\_Earthquake\_Early\_Warning\_Prediction(request):

if request.method == "POST":

if request.method == "POST":

ewtime= request.POST.get('ewtime')

latitude= request.POST.get('latitude')

longitude= request.POST.get('longitude')

depth= request.POST.get('depth')

mag= request.POST.get('mag')

magType= request.POST.get('magType')

nst= request.POST.get('nst')

gap= request.POST.get('gap')

dmin= request.POST.get('dmin')

rms= request.POST.get('rms')

net= request.POST.get('net')

idn= request.POST.get('idn')

updated1= request.POST.get('updated')

place= request.POST.get('place')

horizontalError= request.POST.get('horizontalError')

depthError= request.POST.get('depthError')

magError= request.POST.get('magError')

magNst= request.POST.get('magNst')

df = pd.read\_csv('Earthquake\_Warning\_Datasets.csv')

def apply\_response(Label):

if (Label == "earthquake warning"):

return 0 # earthquake

elif (Label == "explosion"):

return 1 # explosion

df['Results'] = df['Label'].apply(apply\_response)

cv = CountVectorizer()

X = df['place']

y = df['Results']

print("place")

print(X)

print("Results")

print(y)

X = cv.fit\_transform(X)

models = []

from sklearn.model\_selection import train\_test\_split

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.20)

X\_train.shape, X\_test.shape, y\_train.shape

print("Naive Bayes")

from sklearn.naive\_bayes import MultinomialNB

NB = MultinomialNB()

NB.fit(X\_train, y\_train)

predict\_nb = NB.predict(X\_test)

naivebayes = accuracy\_score(y\_test, predict\_nb) \* 100

print("ACCURACY")

print(naivebayes)

print("CLASSIFICATION REPORT")

print(classification\_report(y\_test, predict\_nb))

print("CONFUSION MATRIX")

print(confusion\_matrix(y\_test, predict\_nb))

models.append(('naive\_bayes', NB))

# SVM Model

print("SVM")

from sklearn import svm

lin\_clf = svm.LinearSVC()

lin\_clf.fit(X\_train, y\_train)

predict\_svm = lin\_clf.predict(X\_test)

svm\_acc = accuracy\_score(y\_test, predict\_svm) \* 100

print("ACCURACY")

print(svm\_acc)

print("CLASSIFICATION REPORT")

print(classification\_report(y\_test, predict\_svm))

print("CONFUSION MATRIX")

print(confusion\_matrix(y\_test, predict\_svm))

models.append(('svm', lin\_clf))

print("Logistic Regression")

from sklearn.linear\_model import LogisticRegression

reg = LogisticRegression(random\_state=0, solver='lbfgs').fit(X\_train, y\_train)

y\_pred = reg.predict(X\_test)

print("ACCURACY")

print(accuracy\_score(y\_test, y\_pred) \* 100)

print("CLASSIFICATION REPORT")

print(classification\_report(y\_test, y\_pred))

print("CONFUSION MATRIX")

print(confusion\_matrix(y\_test, y\_pred))

models.append(('logistic', reg))

print("Decision Tree Classifier")

dtc = DecisionTreeClassifier()

dtc.fit(X\_train, y\_train)

dtcpredict = dtc.predict(X\_test)

print("ACCURACY")

print(accuracy\_score(y\_test, dtcpredict) \* 100)

print("CLASSIFICATION REPORT")

print(classification\_report(y\_test, dtcpredict))

print("CONFUSION MATRIX")

print(confusion\_matrix(y\_test, dtcpredict))

models.append(('DecisionTreeClassifier', dtc))

print("Random Forest Classifier")

from sklearn.ensemble import RandomForestClassifier

rf\_clf = RandomForestClassifier()

rf\_clf.fit(X\_train, y\_train)

rfpredict = rf\_clf.predict(X\_test)

print("ACCURACY")

print(accuracy\_score(y\_test, rfpredict) \* 100)

print("CLASSIFICATION REPORT")

print(classification\_report(y\_test, rfpredict))

print("CONFUSION MATRIX")

print(confusion\_matrix(y\_test, rfpredict))

models.append(('RandomForestClassifier', rf\_clf))

classifier = VotingClassifier(models)

classifier.fit(X\_train, y\_train)

y\_pred = classifier.predict(X\_test)

place1 = [place]

vector1 = cv.transform(place1).toarray()

predict\_text = classifier.predict(vector1)

pred = str(predict\_text).replace("[", "")

pred1 = pred.replace("]", "")

prediction = int(pred1)

if (prediction == 0):

val = 'Earthquake Warning'

elif (prediction == 1):

val = 'Explosion Warning'

print(val)

print(pred1)

earthquake\_early\_warning\_prediction.objects.create(

ewtime=ewtime,

latitude=latitude,

longitude=longitude,

depth=depth,

mag=mag,

magType=magType,

nst=nst,

gap=gap,

dmin=dmin,

rms=rms,

net=net,

idn=idn,

updated=updated1,

place=place,

horizontalError=horizontalError,

depthError=depthError,

magError=magError,

magNst=magNst,

Prediction=val

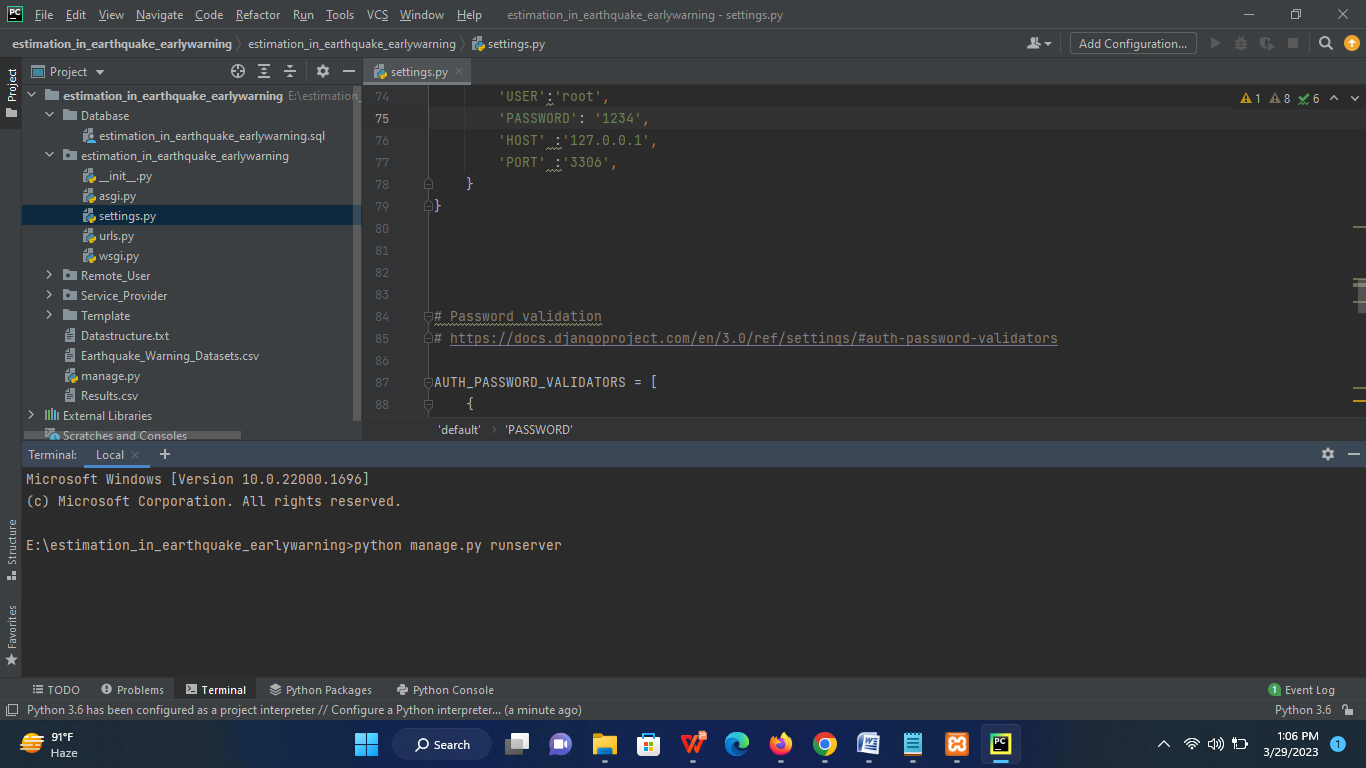
)

return render(request, 'RUser/Predict\_Earthquake\_Early\_Warning\_Prediction.html',{'objs': val})

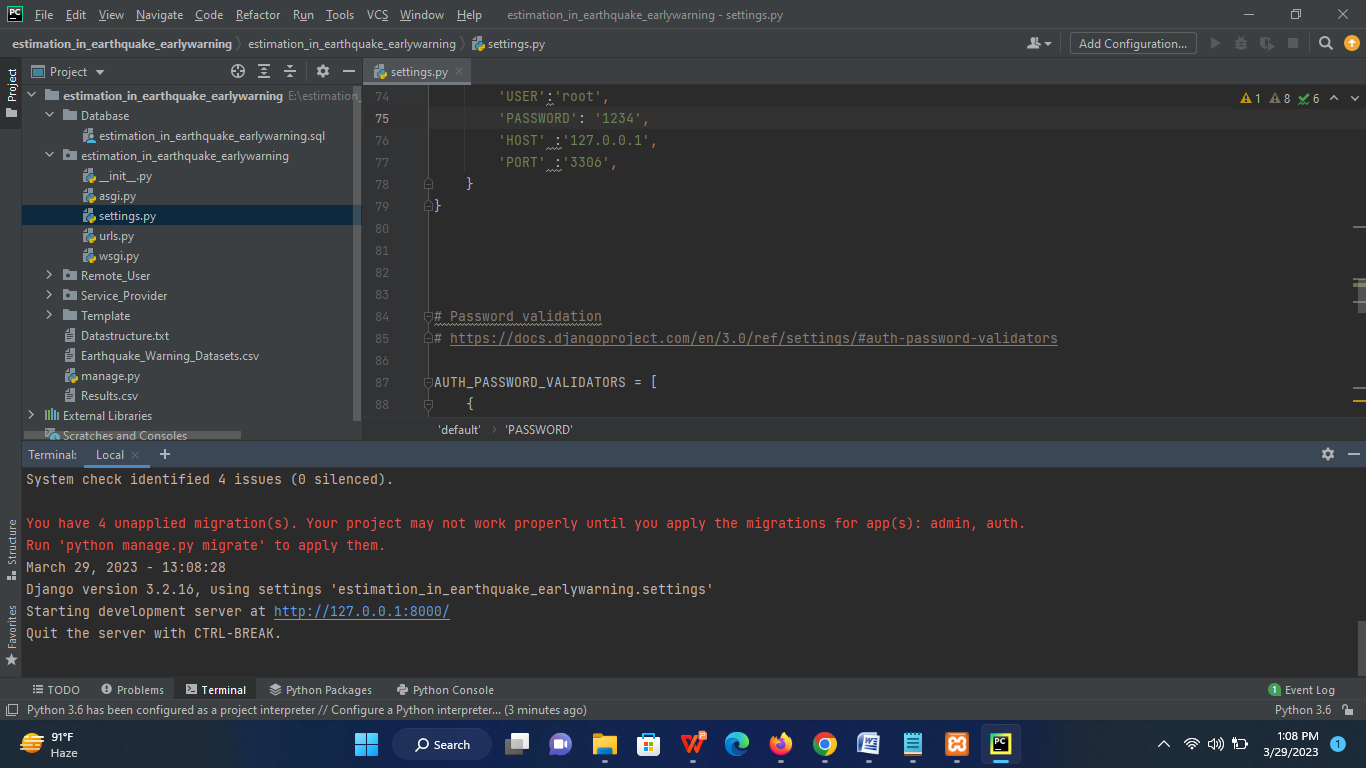
return render(request, 'RUser/Predict\_Earthquake\_Early\_Warning\_Prediction.html')

**CHAPTER-8**

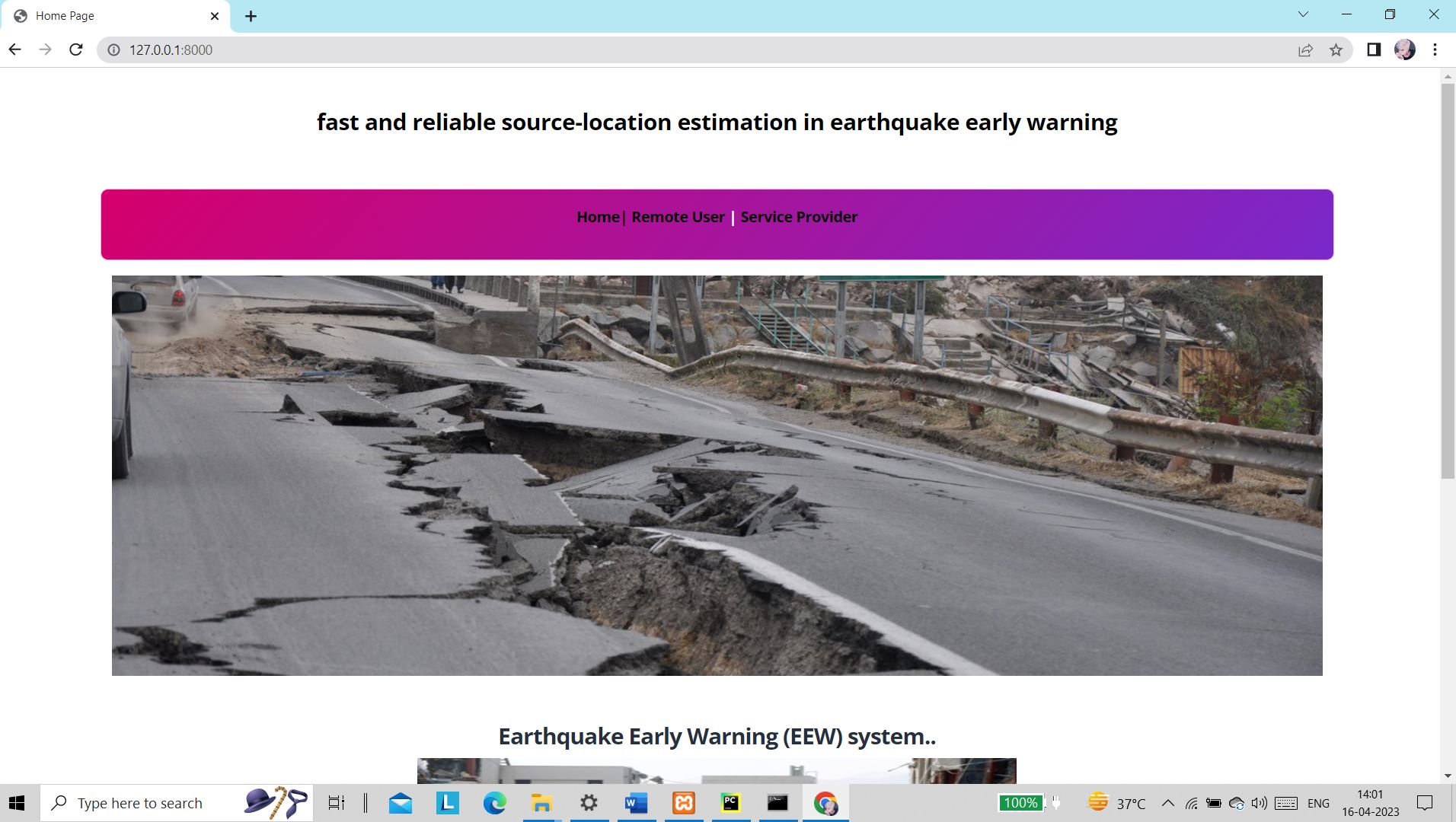
**OUTPUT**

****

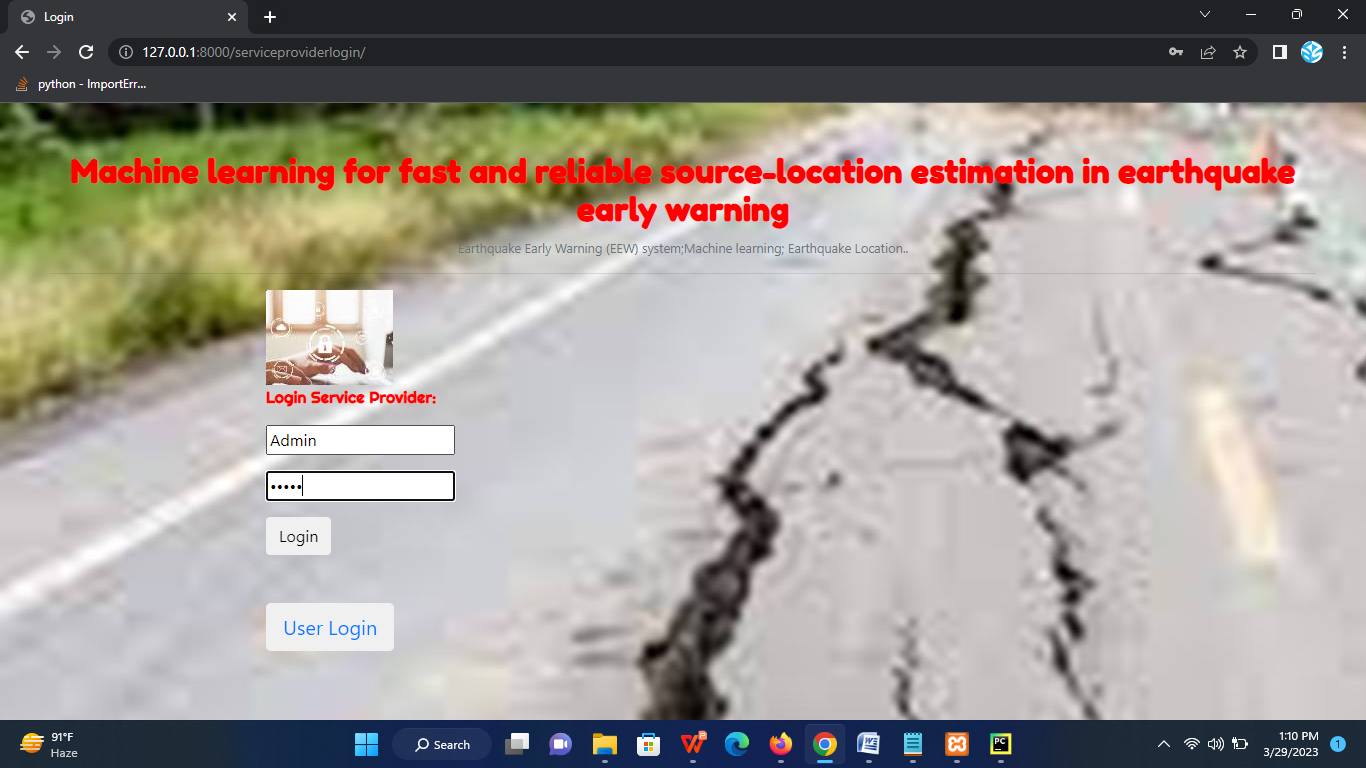
**Fig 8.1 Running Command on console**

****

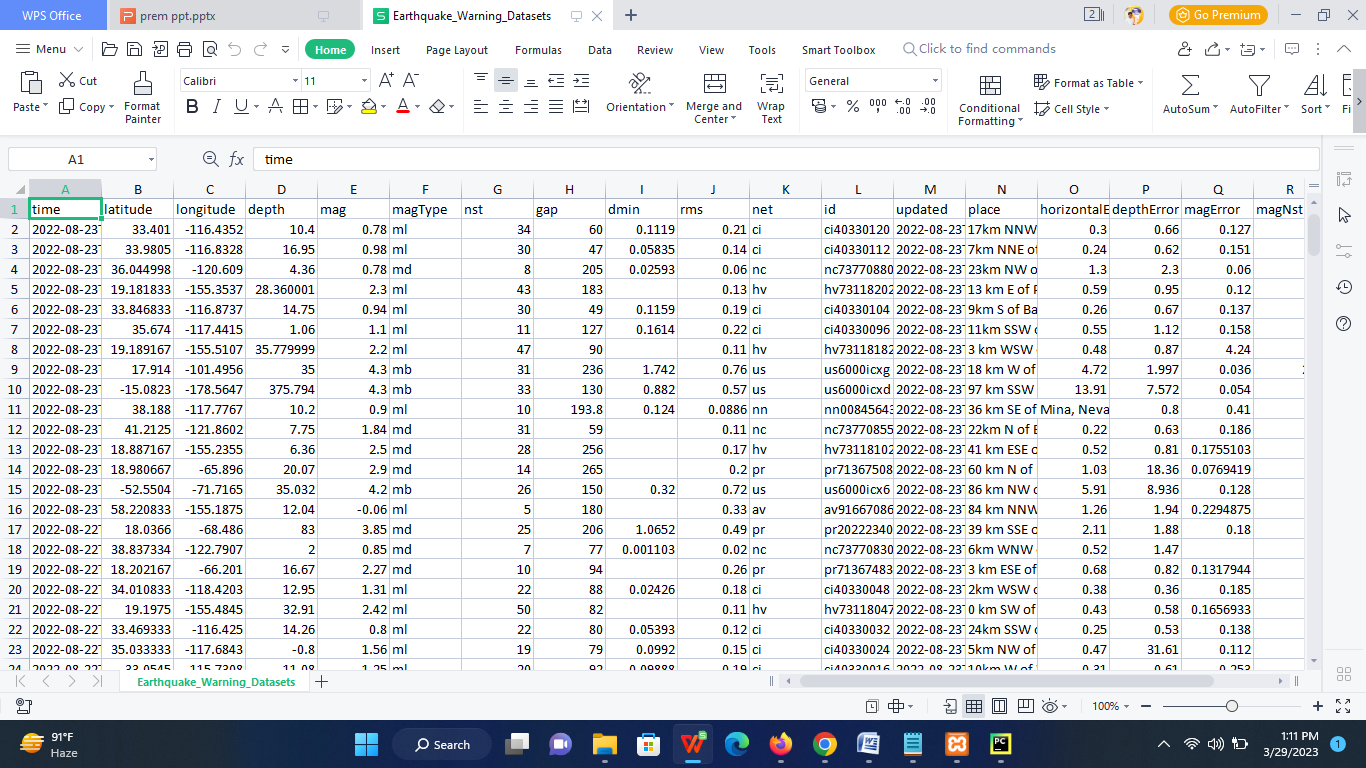
**Fig 8.2 Generating Link**

****

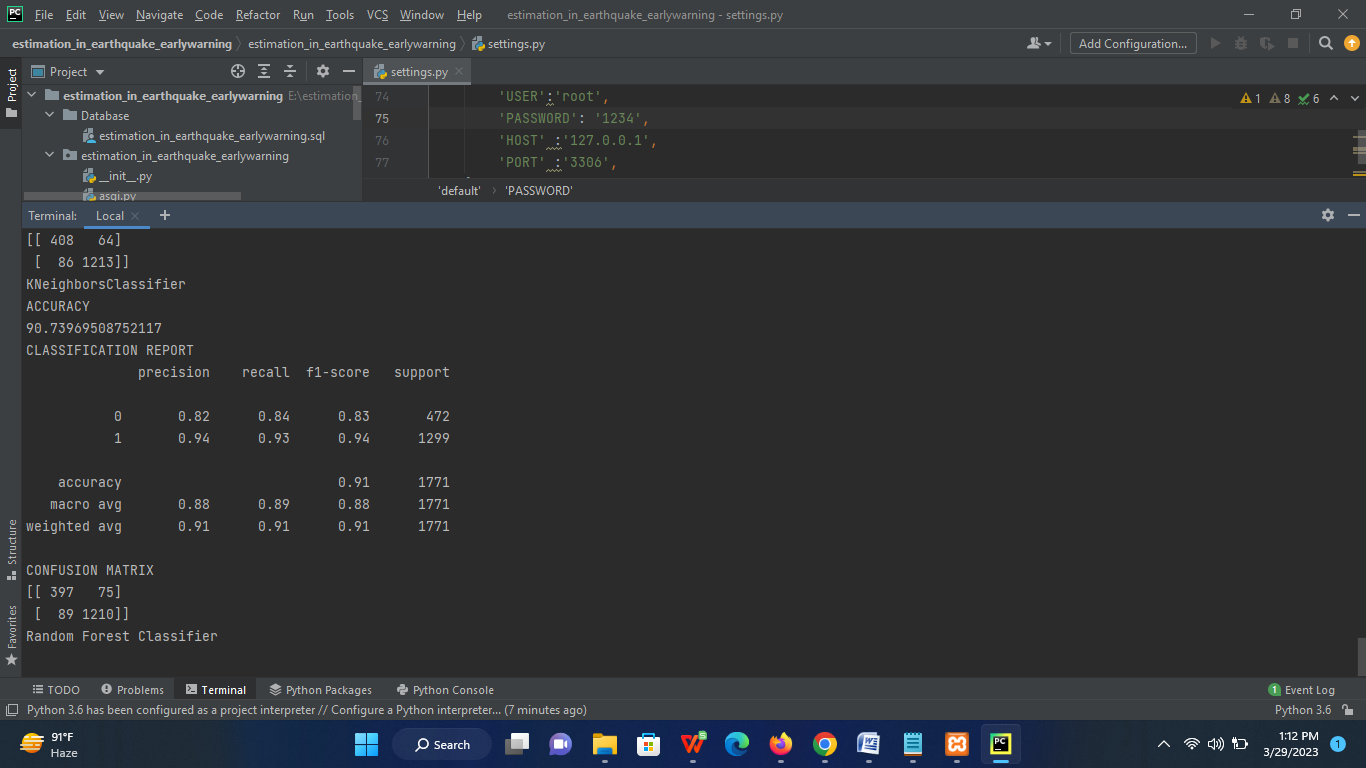
**Fig 8.3 Home Page**

****

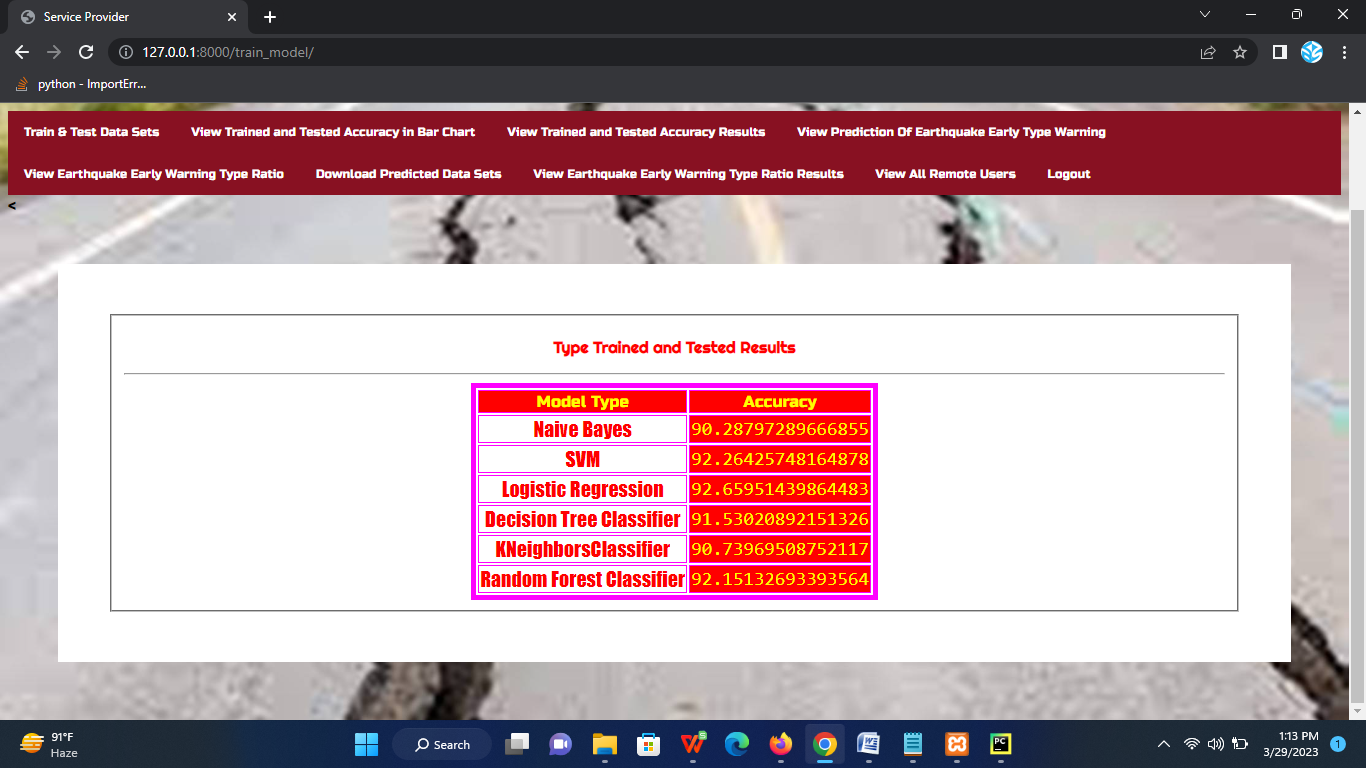
**Fig 8.4 Service Provider Login**

****

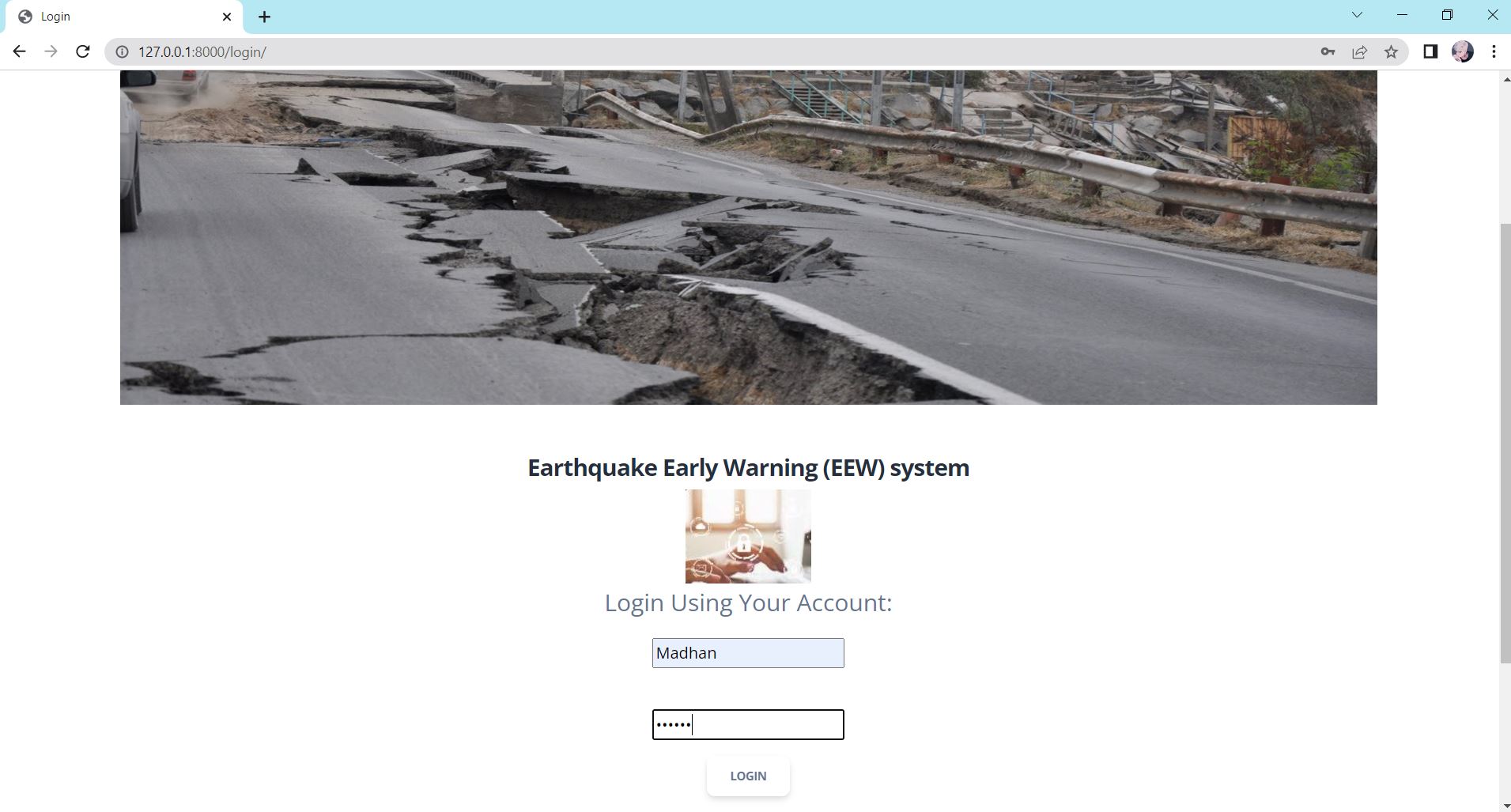
**Fig 8.5 Earth Quake Dataset**

****

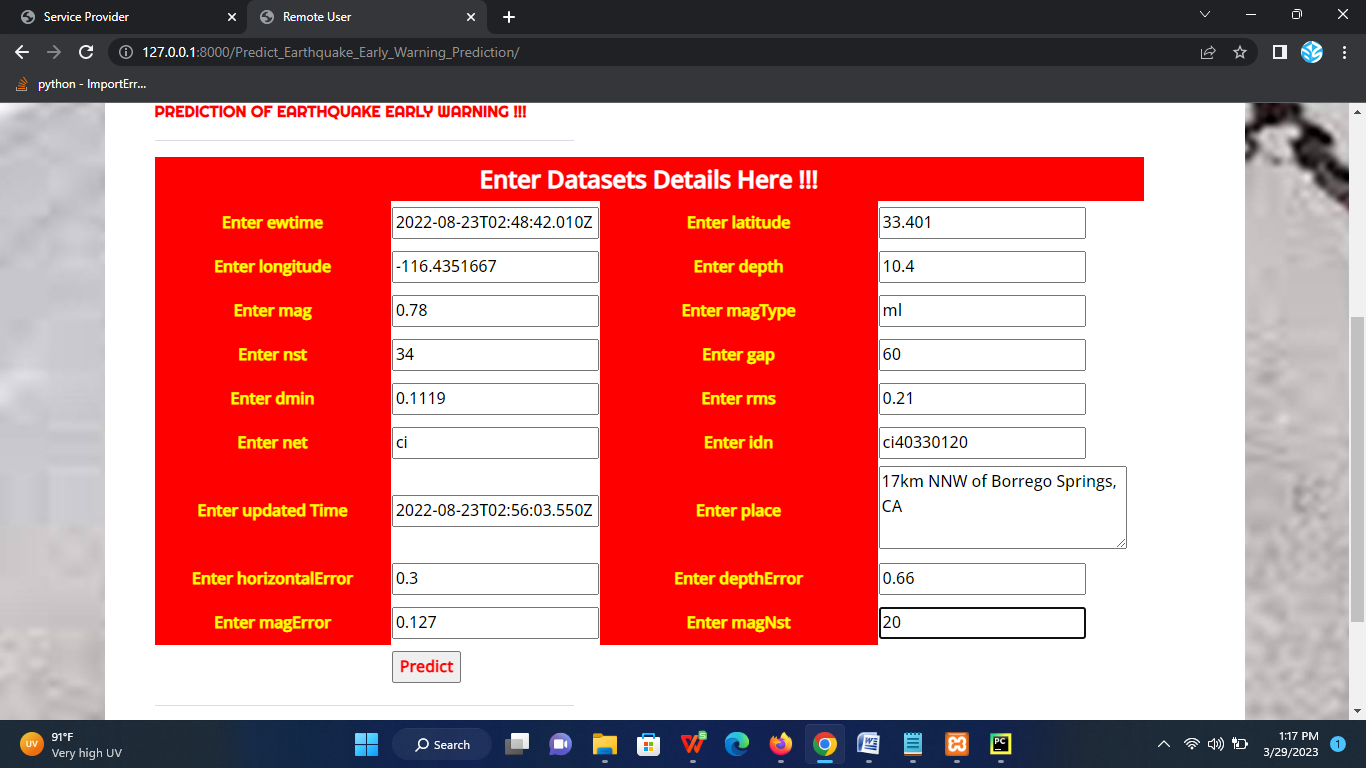
**Fig 8.6 Training with ML Algorithms**

****

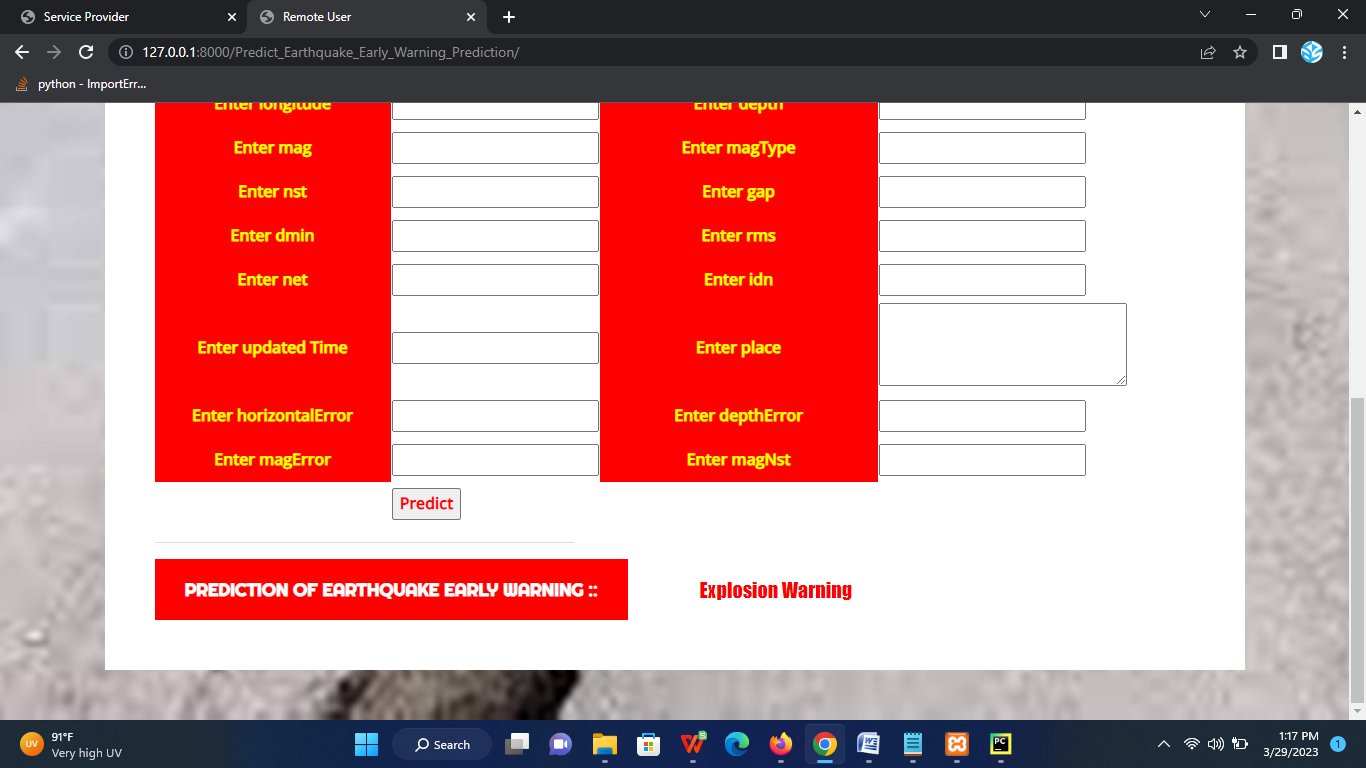
**Fig 8.7 Trained and Tested results**

****

**Fig 8.8 User Login**

****

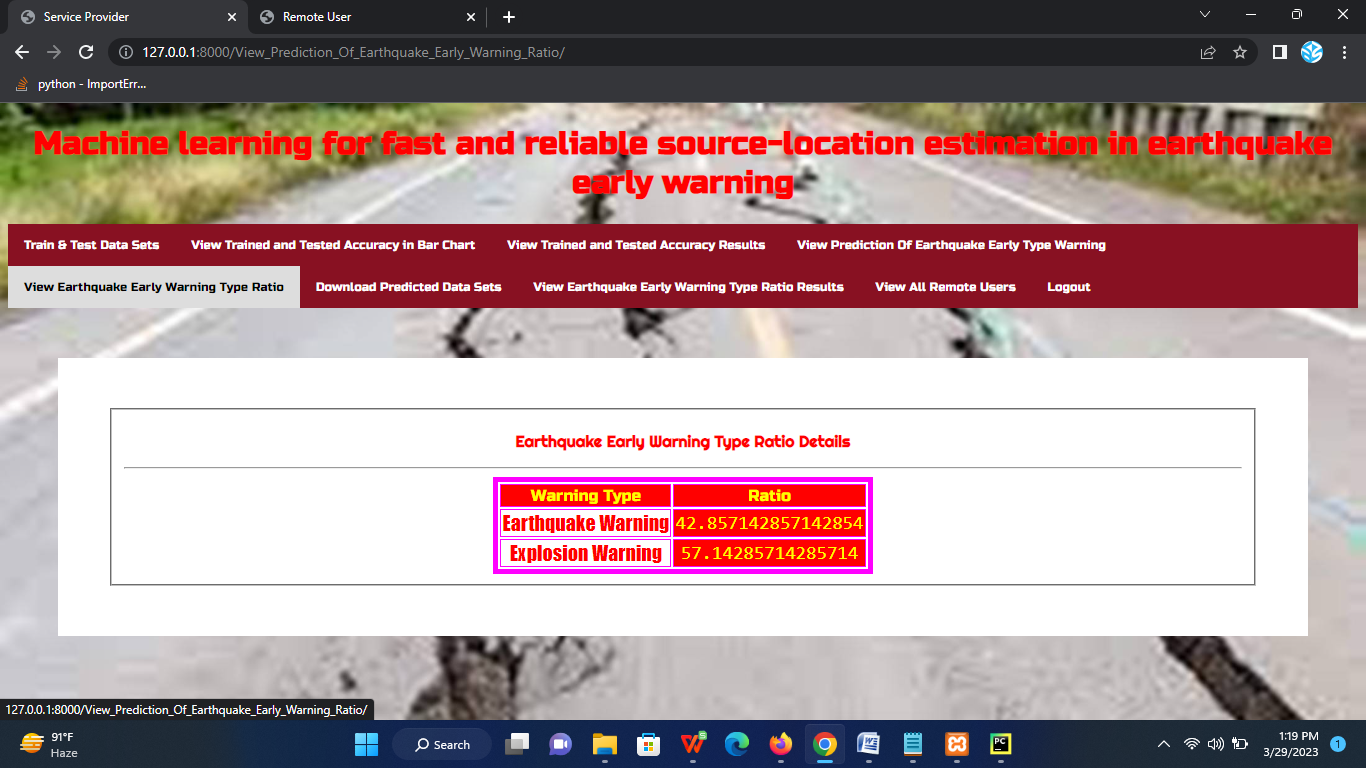
**Fig 8.9 Enter Values For Prediction**

****

**Fig 8.10 Prediction Result**

****

**Fig 8.11 Earthquake Early warning Prediction Type Details**

****

**Fig 8.12 Earthquake Early warning Type Ratio Details**

**CHAPTER-09**

**CONCLUSION**

We use the P-wave arrival time differences and the location of the seismic stations to locate the earthquake in a real-time way. Random forest (RF) has been proposed to perform this regression problem, where the difference latitude and longitude between the earthquake and the seismic stations are considered as the RF output. The Japanese seismic area is used as a case of study, which demonstrates very successful performance and indicates its immediate applicability. We extract all the events having at least five P-wave arrival times from nearby seismic stations. Then, we split the extracted events into training and testing datasets to construct a machine learning model. In addition, the proposed method has the ability to use only three seismic stations and 10% of the available dataset for training, still with encouraging performance, indicating the flexibility of the proposed algorithm in real-time earthquake monitoring in more challenging areas.

**CHAPTER 10**

**FUTURE WORK**

Despite the sparse distribution of many networks around the world, which makes the ML method difficult to train an effective model, one can use numerous synthetic datasets to compensate for the shortage of ray paths in a target area due to insufficient catalog and station distribution. So In future we try to implement on deep learning approaches.

**CHAPTER-12**

**REFERENCES**

[1] Q. Kong, R. M. Allen, L. Schreier, and Y.-W. Kwon, “Myshake: A smartphone seismic network for earthquake early warning and beyond,” Science advances, vol. 2, no. 2, p. e1501055, 2016.

[2] T.-L. Chin, K.-Y. Chen, D.-Y. Chen, and D.-E. Lin, “Intelligent real-time earthquake detection by recurrent neural networks,” IEEE Transactions on Geoscience and Remote Sensing, vol. 58, no. 8, pp. 5440–5449, 2020.

[3] T.-L. Chin, C.-Y. Huang, S.-H. Shen, Y.-C. Tsai, Y. H. Hu, and Y.-M.Wu, “Learn to detect: Improving the accuracy of earthquake detection,” IEEE Transactions on Geoscience and Remote Sensing, vol. 57, no. 11, pp. 8867–8878, 2019.

[4] O. M. Saad, A. G. Hafez, and M. S. Soliman, “Deep learning approach for earthquake parameters classification in earthquake early warning system,” IEEE Geoscience and Remote Sensing Letters, pp. 1–5, 2020.

[5] X. Zhang, J. Zhang, C. Yuan, S. Liu, Z. Chen, and W. Li, “Locating induced earthquakes with a network of seismic stations in oklahoma via a deep learning method,” Scientific reports, vol. 10, no. 1, pp. 1–12, 2020.

[6] L. Breiman, “Random forests,” Machine learning, vol. 45, no. 1, pp. 5–32, 2001.

[7] S. M. Mousavi, W. L. Ellsworth, W. Zhu, L. Y. Chuang, and G. C. Beroza, “Earthquake transformeran attentive deep-learning model for simultaneous earthquake detection and phase picking,” Nature Communications, vol. 11, no. 1, pp. 1–12, 2020.