# **Technical Architecture:**

# 

# 

# **Components & Technologies:**

## 1. User Interface

### **Description**

User interacts with the application for the prediction of Any Natural disaster which will happen in future minutes.

### **Technology**

HTML, CSS, JavaScript, Django, Python.

## 2. Feature Engineering Pipeline

### **Description**

Algorithms can't make sense of raw data. We have to select, transform, combine, and otherwise prepare our data so the algorithm can find useful patterns.

### **Technology**

Image processing, pattern extraction, etc.

## 3. Model Training kit

### **Description**

It learns patterns from the data. Then they use these patterns to perform particular tasks

### **Technology**

Multiclass Classification Model, Regression Model, etc.

## 4. Prediction unit

### **Description**

This function is used to predict outcomes from the new trained data to perform new tasks and solve new problems.

### **Technology**

Decision trees, Regression, Neural networks.

## 5. Evaluation system

### **Description**

It monitors that how Algorithm performs on data as well as during training

### **Technology**

Chi-Square, Confusion Matrix, etc.

## 

## 6. Interactive services

### **Description**

To interact with our model and give it problems to solve. Usually this takes the form of an API. a user interface or a command-line interface

### **Technology**

Application programming interface, etc.

## 7. Data collection unit

### **Description**

Data is only useful if it's accessible, so it needs to be stored ideally in a consistent structure and conveniently in one place.

### **Technology**

IBM Cloud, SQL Server.

## 8. Data generation system

### **Description**

Every machine learning application lives off data. That data has to come from somewhere. Usually, it's generated by one of your core business functions

### **Technology**

Synthetic data generation.

## 9. Database management system

### **Description**

An organized collection of data stored in a database, so that it can be easily accessed and managed.

### **Technology**

MySQL, DynamoDB etc.

## 10. IBM Cloud services

### **Description**

Processed data stored in cloud service which can be access by the admin anywhere over the internet.

### **Technology**

IBM Cloud etc.

# Application Characteristics:

## 1. Open-Source Frameworks

### **Description**

An open source framework is a template for software development that is designed by a social network of software developers. These frameworks are free for public use and provide the foundation for building a software application.

### **Technology**

Keras, tensor flow.

## 2. Authentication

### **Description**

This keeps our models secure and makes sure only those who have permission can use them.

### **Technology**

Encryption and Decryption (OTP).

## 3. Application interface

### **Description**

User uses mobile application and web application to interact with model

### **Technology**

Android and Web Development (PhoneGap, ReactNative, and NativeScript).

## 

## 4. Availability (both Online and Offline work)

### **Description**

It includes both online and offline work. A good internet connection is needed for online work to explore the software perfectly. Offline work includes the saved data to explore for later time.

### **Technology**

Caching, backend server.

## 

## 5. Regular Updates

### **Description**

The truly excellent software product needs a continuous process of improvements and updates. Maintain your server and make sure that your content is always up-to-date. Regularly update an app and enrich it with new features.

### **Technology**

* Waterfall Approach
* Incremental Approach
* Spiral Approach

## 

## 6. Personalization

### **Description**

Software has features like flexible fonts, backgrounds settings, colour themes, etc. which make a software interface look good and functional.

### **Technology**

* HubSpot
* Proof