# Phase 5: Project Documentation & Submission

## 1.1 Understanding the problem :

The problem is to develop an earthquake prediction model using a Kaggle dataset. The objective is to explore and understand the key features of earthquake data, visualize the data on a world map for a global overview, split the data for training and testing, and build a neural network model to predict earthquake magnitudes based on the given features.

## 2.1 Design Thinking :

Our approach to solve this problem involves a structured design thinking process, ensuring systematic and effective analysis of Earthquake prediction.

### 3.1 Data Collection:

For this project, we use a dataset from kaggle which is recommended for our problem statement. The link of the dataset is given below .

**Dataset Link:** **https://www.kaggle.com/datasets/usgs/earthquake-database**

### 4.1 Code Implementation:

```
import numpy as np
import pandas as pd
from keras.models import Sequential
from keras.layers import Dense
import folium

# Predicting the earthquake intensity which is given by the user.
test_longitude = float(input("Enter longitude : "))       # test value : -118.0
test_latitude = float(input("Enter latitude : "))         # test value : 34.0

# Dataset recommended  by Naan Mudhalvan
data = pd.read_csv('/kaggle/input/earthquake-database/database.csv')
```

```
df = pd.DataFrame(data)
```

### 4.1.1 Normalization

As mentioned before, our model requires latitude and longitude as input and intensity , lat and long of earthquake for processing. The dataset given for us already contains latitude and longitude, so we are not modifying any fields of the dataset.

```
# Normalize longitude and latitude
df['normalized_longitude'] = (df['Longitude'] - df['Longitude'].mean()) /
df['Longitude'].std()
df['normalized_latitude'] = (df['Latitude'] - df['Latitude'].mean()) / df['Latitude'].std()

# Scale earthquake intensity between 0 and 1
df['scaled_intensity'] = (df['Magnitude'] - df['Magnitude'].min()) / (df['Magnitude'].max() -
df['Magnitude'].min())
```

### 4.1.2 Creating a Neural Network

```
# Create a neural network model for regression
model = Sequential()
model.add(Dense(units=32, activation='relu', input_dim=2))
model.add(Dense(units=1, activation='sigmoid'))

# Compile the model
model.compile(loss='mean_squared_error', optimizer='adam')

# Train the model using normalized longitude, latitude, and scaled intensity
model.fit(df[['normalized_longitude', 'normalized_latitude']], df['scaled_intensity'],
epochs=10, batch_size=32)

# Normalize the test longitude and latitude
normalized_test_longitude = (test_longitude - df['Longitude'].mean()) /
df['Longitude'].std()
normalized_test_latitude = (test_latitude - df['Latitude'].mean()) / df['Latitude'].std()

# Predict the earthquake intensity for the test location
predicted_intensity = model.predict(np.array([[normalized_test_longitude,
normalized_test_latitude]]))[0][0]
```

### 4.1.3 Visualization

To effectively display our findings we will be displaying the output of our model in a map.For visualization we will be using folium library from python in order to display our output in map.

```
# Create a base map
map = folium.Map(location=[test_latitude, test_longitude], zoom_start=12)

# Adding a circle around intensity area
folium.CircleMarker(
    location=[test_latitude, test_longitude],
    radius= predicted_intensity * 20,
    color='red',
    fill=True,
    fill_color='red',
    fill_opacity=0.6
).add_to(map)

#Saving the map as intensity_map.html
map.save('intensity_map.html')
```

### 4.1.4 Output:

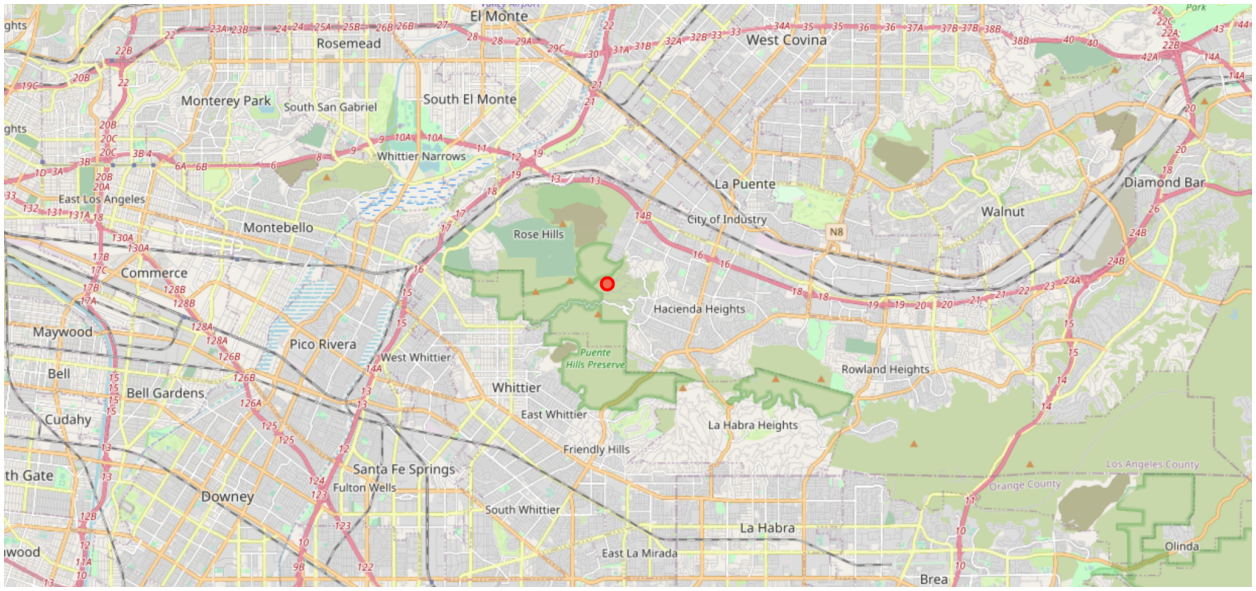
```
Enter longitude :  80.270721
Enter latitude :  13.082680

Epoch 1/10
732/732 [==============================] - 1s 1ms/step - loss: 0.0370
Epoch 2/10
732/732 [==============================] - 1s 1ms/step - loss: 0.0141
Epoch 3/10
732/732 [==============================] - 1s 1ms/step - loss: 0.0139
Epoch 4/10
732/732 [==============================] - 1s 1ms/step - loss: 0.0138
Epoch 5/10
732/732 [==============================] - 1s 1ms/step - loss: 0.0138
```

/kaggle/working

intensity_map.html

Intensity_map.html




The red dot represents the intensity of earthquakes in the given region.Wider the dot more effective the earthquake will be.