Logo

Description automatically generated with low confidence

**Fundamentals of**

**Software Engineering**

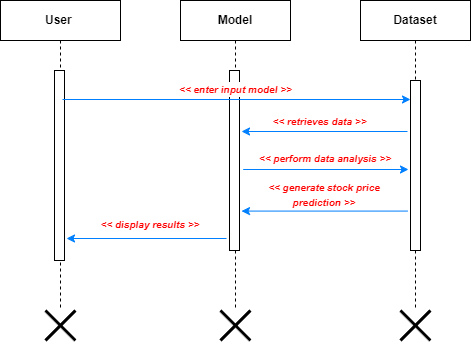
**Assignment #3**

**Iteration 2: Execution Phase (OUTLINING SPRINT)**

**Name : Eman Furrukh**

**Roll No : 21i-1726**

**Section : BSDS-U**

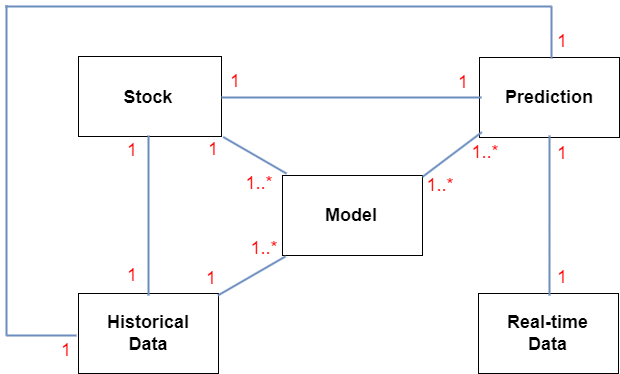
**1. Design the complete sequence diagram**

**Dataset**

**Web Page**

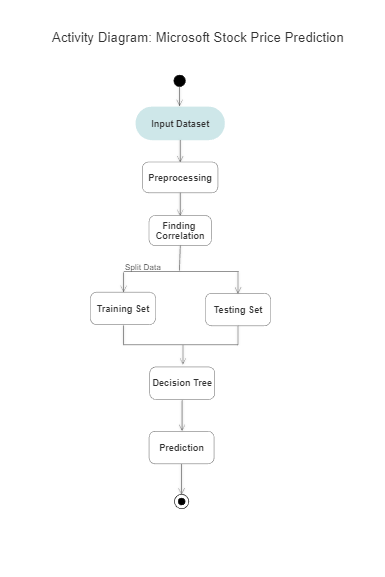
**User**

In this sequence diagram, the user enters their input data into the web application. The web application retrieves the necessary data from the database, performs data analysis, and generates a stock price prediction. The web application then displays the prediction results to the user. The arrows represent the flow of information and actions between the user, the web application, and the database.

**2. Design the complete data association diagram**

The Microsoft Stock Price Prediction system has four main data entities: Stock, Model, Historical Data, and Real-time Data. Each of these entities is represented by a box, and the relationships between them are shown by the lines connecting the boxes.

The arrows indicate the direction of the relationship between entities. For example, a Stock can have historical data associated with it, which can be used to train the predictive Model. The Model can then be used to make predictions on real-time data.

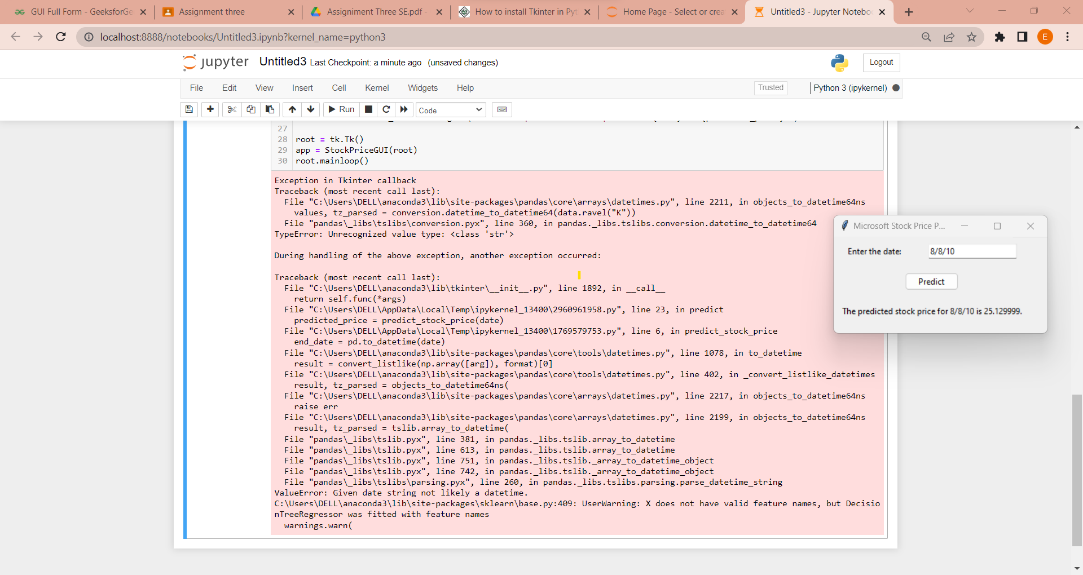
**3. Design the detail and complete activity diagram**

In this activity diagram, the User Interface collects data from the user when they input the dataset, preprocesses the data, finds the correlation between variables, trains’ and tests the model, and then uses the model to make predictions on new data. Finally, the results are displayed to the user.

**4. Build a enrich Graphical Interface of your project along with comprehensive report(first**

**level testing also a part)**

GUI -> coded in another file in this folder.

First level Testing:

Here’s a simple pseudocode to test the GUI of the project:

date = "2010-08-08"

expected\_price = 25.129999

predicted\_price = predict\_stock\_price(date)

# Compare the predicted price with the expected price

if predicted\_price == expected\_price

print("Test passed!")

else:

print("Test failed.")

In this example, we provide the function with a sample input of a date string "8/8/10". We then set the expected output to 25.12999, which is just an arbitrary value for demonstration purposes. We call the function with the sample input and store the result in predicted\_price. We then compare the predicted price with the expected price using an if-else statement. If the two values match, we print "Test passed!" to indicate that the test was successful. Otherwise, we print "Test failed." to indicate that the test was unsuccessful.