# A
# Project Report
# On
# "Currency Exchange Rate Prediction"

**Prepared by**
Purv Master (19DCS059)
Arya Patel (19DCS080)

**Under the guidance of**

Prof. Dipak Ramoliya

A Report Submitted to

Charotar University of Science and Technology

for Partial Fulfillment of the Requirements for the

6th Semester Software Group Project-IV (CS357)

**Submitted at**

**CHARUSAT**
CHAROTAR UNIVERSITY OF SCIENCE AND TECHNOLOGY
**Accredited with Grade A by NAAC**
**Accredited with Grade A by KCG**

**Department of Computer Science & Engineering**

**Devang Patel Institute of Advance Research & Technology**

**At: Changa, Dist: Anand – 388421**

**April 2022**

# DECLARATION BY THE CANDIDATE

I hereby declare that the project report entitled "Currency Exchange Rate Prediction" submitted by me to Devang Patel Institute of Advance Research and Technology, Changa in partial fulfilment of the requirement for the award of the degree of B.Tech in Computer Science Engineering, from to Devang Patel Institute of Advance Research and Technology, is a record of bonafide CS357 Software Group Project (project work) carried out by me under the guidance of Prof. Dipak Ramoliya. I further declare that the work carried out and documented in this project report has not been submitted anywhere else either in part or in full and it is the original work, for the award of any other degree or diploma in this institute or any other institute or university.

Purv Master(19DCS059)

Prof. Dipak Ramoliya
Devang Patel Institute of Advance Research and Technology
DEPSTAR-FTE, CHARUSAT-Changa

DEPSTAR

Department of Computer
Science & Engineering

# CERTIFICATE

This is to certify that the report entitled "**Currency Exchange Rate Prediction**" is a bonafied work carried out by **Mr. Purv Master (19DCS059)** under the guidance and supervision of **Prof. Dipak Ramoliya** for the subject CS357)-**Software Group Project-IV** (CSE) of 6th Semester of Bachelor of Technology in **DEPSTAR** at Faculty of Technology & Engineering – CHARUSAT, Gujarat.

To the best of my knowledge and belief, this work embodies the work of candidate himself, has duly been completed, and fulfills the requirement of the ordinance relating to the B.Tech. Degree of the University and is up to the standard in respect of content, presentation and language for being referred to the examiner.

Prof. Dipak Ramoliya
*Assistant Professor*
Computer Science & Engineering
DEPSTAR, Changa, Gujarat.

Prof. Parth Goel
Head of Department – Computer Science
& Engineering, DEPSTAR
CHARUSAT, Changa, Gujarat.

Dr. Amit Ganatra
Principal, DEPSTAR
Dean, FTE
CHARUSAT, Changa, Gujarat.

**Devang Patel Institute of Advance Technology And Research At: Changa, Ta. Petlad, Dist. Anand, PIN: 388 421. Gujarat**

# ABSTRACT

The main of this project is, The foreign exchange market due to its continuity, liquidity and trading volume is a perfect contender to apply machine learning methods to. Trading in this market is conducted on a continuous basis all around the globe, leading to stable numerical data and low transaction fees in comparison to other markets such as the equity, fixed-income and the derivatives market. This report summaries the implementation of four different statistical methods using Python. This paper presents different machine learning techniques like Design tree regression, to develop prediction model between five major currencies against other major currencies like Euro, Pound Sterling and US dollar, etc. Currency exchange rate prediction is a typical time series prediction problem which has been solved by time-series models, such as Autoregressive Integrated Moving Average (ARIMA), Seasonal ARIMA (SARIMA) as well as machine learning methods, such as Single Layer Perception (SLP) and Long Short-Term Memory (LSTM). The models will be trained on the basis of predicting currency through a period of two month. In this report, coding choices will be highlighted, commenting upon advantages, disadvantages and areas for further improvement.

.

# ACKNOWLEDGEMENT

I, the developer of Android application "Currency Exchange Rate Prediction", with immense pleasure and commitment would like to present the project assignment. The development of this project has given me wide opportunity to think, implement and interact with various aspects of management skills as well as the new emerging technologies.

Every work that one completes successfully stands on the constant encouragement, good will and support of the people around. I hereby avail this opportunity to express my gratitude to number of people who extended their valuable time, full support and cooperation in developing the project.

I express deep sense of gratitude towards our Head of the CSE Department, Dr. Amit Ganatra and project guides Prof. Dipak Ramoliya for the support during the whole session of study and development. It is because of them, that I was prompted to do hard work, adopting new technologies.

I would also like to thank my mentor Mr. Dev Desai and Mr. Krutik Patel for his guidelines throughout the development phase of the application. They helped me, whenever I was stuck in the Machine Learning Concepts.

Thanks,
Purv Master

# TABLE OF CONTENTS

# LIST OF FIGURES

# CHAPTER 1: INTRODUCTION

## 1.1 PROJECT OVERVIEW

Our evaluation technique involves observing the actual currency value (for the day for which we estimated) and calculating the mean squared error. We plan to achieve this by using Python as a programming language and using couple of libraries. The proposed method involves using/building and evaluating different models to predict the currency values for the available time series data. The time series data has a special property that the data is captured at regular time intervals. We consider the daily data containing the exchange rates for the target currency against the currency of other countries.

The historical data can be retrieved using this Microsoft Excel sheet by entering the standard currency abbreviation, start date, and end date. We have use Android Studio to deploy this model so we can use this application

We have used android studio for developing its application. We created API for deploying trained model of currencies. This application give option to select currencies. It also has graph option so that you can easily read data.

## 1.2 OBJECTIVE

The machine learning techniques have been applied to financial markets since the dawn of the computer age. In particular, the foreign exchange market due to its continuity, liquidity and trading volume is a perfect contender to apply machine learning methods to. Trading in this market is conducted on a continuous basis all around the globe, leading to stable numerical data and low transaction fees in comparison to other markets such as the equity, fixed-income and the derivatives market. The main objective of the project is to build a application that can predict the exchange rate for the next day or any of future day/s with the highest accuracy. This will help people to have a look ahead at currency exchange rate before making any kind of transaction.

## 1.3 SCOPE

As a foreign student, we almost every day check the currency value of our home country against the dollar value for fees and other types of usages. For a student, it is beneficial if the exchange rate is minimum.

Businessmen, Investors, Students, Travelers, many people can use this application to predict and check future price of their currency. So that it is beneficial if the exchange rate is minimum

We can produce directional Foreign Exchange forecasts that are able to yield profitable investment strategies.

## 1.4 TOOLS & TECHNOLOGY USED

- Google Colab

Colaboratory, or "Colab" for short, is a product from Google Research. Colab allows anybody to write and execute arbitrary python code through the browser, and is especially well suited to machine learning, data analysis and education. Colab notebooks allow you to combine executable code and rich text in a single document, along with images, HTML, LaTeX and more.

- Android Studio

Android Studio is the official integrated development environment for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development

- Any Android Devices

# CHAPTER 2: PROJECT MANAGEMENT

## 2.1 PROJECT PLANNING

### 2.1.1 Project Development Approach and Justification

The SPIRAL model is a risk-driven process model generator for software projects. Based on the unique risk patterns of a given project, the spiral model guides a team to adopt elements of one or more process models, such as incremental, waterfall, or evolutionary prototyping.
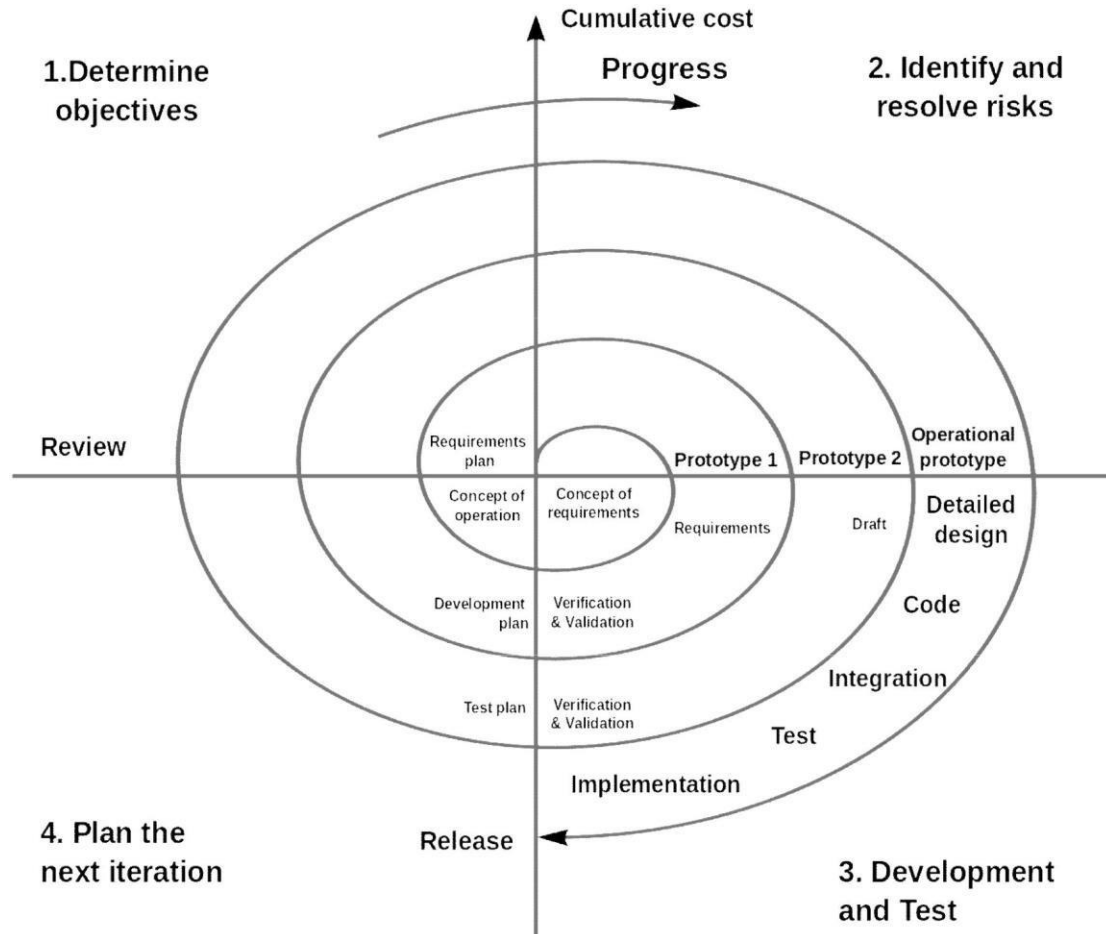
Fig 2.1 Spiral Model

1. The requirements are known in advance of implementation.
2. The requirements have no unresolved, high-risk implications, such as risks due to cost, schedule, performance, safety, security, user interfaces, organizational impacts, etc.

3. The nature of

4.  the requirements will not change very much during development orevolution.

5. The requirements are compatible with all the key system stakeholders' expectations, including users, customer, developers, maintainers, and investors.

6. The right architecture for implementing the requirements is well understood.

7. There is enough calendar time to proceed sequentially.

**Perform four basic activities in every cycle:**

This invariant identifies the four activities that must occur in each cycle of the spiral model:

1. Consider the win conditions of all success-critical stakeholders.

2. Identify and evaluate alternative approaches for satisfying the win conditions.

3. Identify and resolve risks that stem from the selected approach(es).

4. Obtain approval from all success-critical stakeholders, plus commitment to pursue the next cycle.

**Key idea:** On each iteration identify and solve the sub-problems with the highest risk.

**Advantages:**

1. Realism: the model accurately reflects the iterative nature of software development on projects with unclear requirement.

2. Flexible: incorporates the advantages of the waterfall and evolutionary methods.

3. Comprehensive model decreases risk.

4. Good project visibility.

## 2.2 PROJECT WORK SCHEDULING

### 2.2.1 Gantt Chart

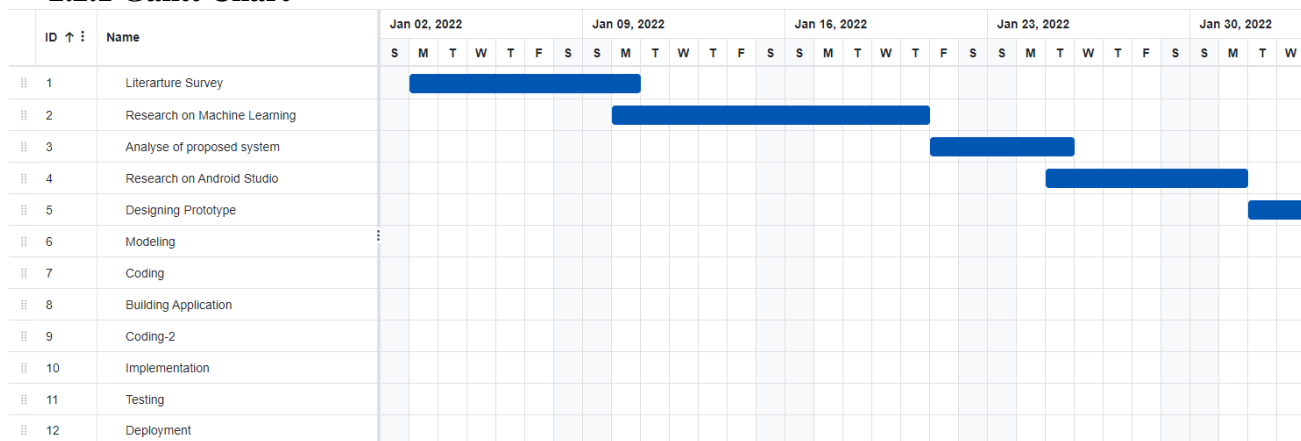| ID ↑ | Name | Jan 02, 2022 | | | | | | | Jan 09, 2022 | | | | | | | Jan 16, 2022 | | | | | | | Jan 23, 2022 | | | | | | | Jan 30, 2022 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | S | M | T | W | T | F | S | S | M | T | W | T | F | S | S | M | T | W | T | F | S | S | M | T | W | T | F | S | S | M | T | W |
| 1 | Literarture Survey | | ███ | ███ | ███ | ███ | ███ | ███ | ███ | | | | | | | | | | | | | | | | | | | | | | | | |
| 2 | Research on Machine Learning | | | | | | | | ███ | ███ | ███ | ███ | ███ | ███ | ███ | ███ | ███ | | | | | | | | | | | | | | | |
| 3 | Analyse of proposed system | | | | | | | | | | | | | | | | | | | | ███ | ███ | ███ | ███ | | | | | | | | |
| 4 | Research on Android Studio | | | | | | | | | | | | | | | | | | | | | | ███ | ███ | ███ | ███ | ███ | ███ | | | | |
| 5 | Designing Prototype | | | | | | | | | | | | | | | | | | | | | | | | | | | | ███ | ███ | ███ | |
| 6 | Modeling | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 7 | Coding | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 8 | Building Application | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 9 | Coding-2 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 10 | Implementation | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 11 | Testing | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 12 | Deployment | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

Fig 2.1 Gantt Chart

# CHAPTER 3: SYSTEM REQUIREMENTS STUDY

## 3.1 USER CHARACTERISTICS

- **End Users**

  This product is designed mainly for END USERS. So, it is feasible for the end-users to directly play the game and interact with the virtual 3D environment.

- **Administrator**

  He can add, delete or update any functionalities he wants. He has access to the core scripts and he can do optimization the way he wants. He is responsible for the design of the environment.
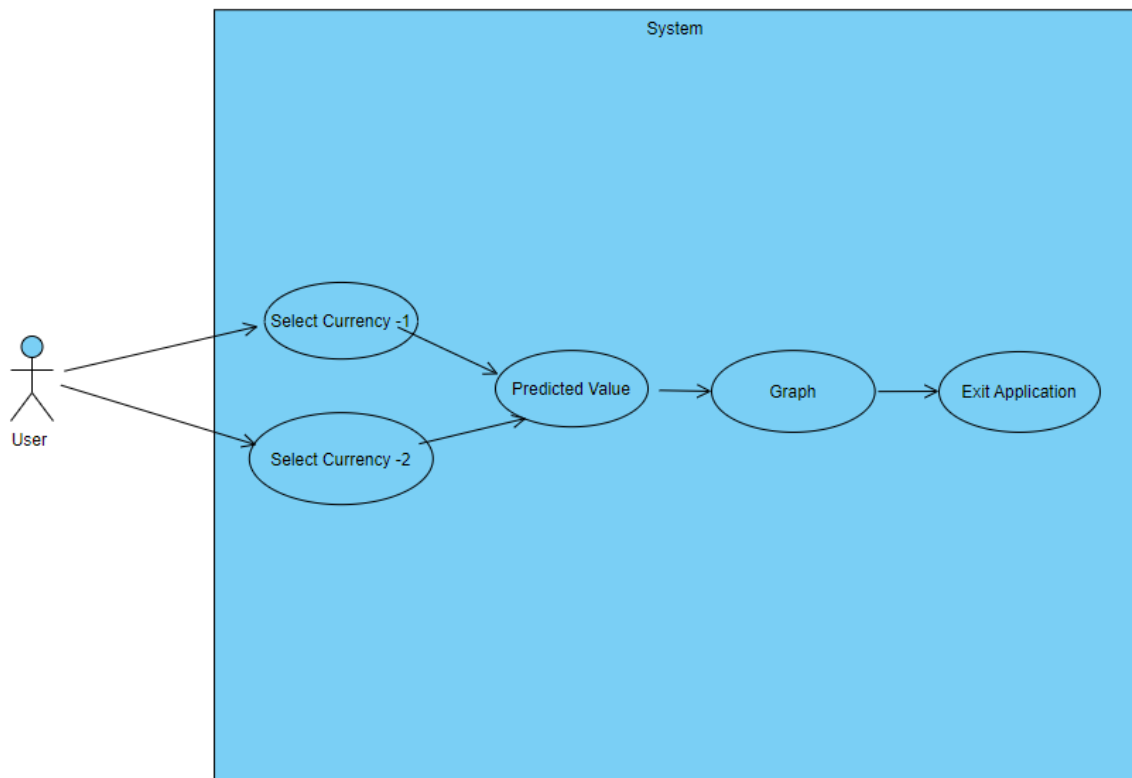
### 3.1.1 Use Case Diagram



Fig 3.1. Use Case Diagram

## 3.2 HARDWARE AND SOFTWARE REQUIREMENTS

### 3.2.1 Hardware specification

- Windows: GPU with Direct X 11 feature level 10 support

- Qualcomm Snapdragon 655 or more

- 2.5 GHz processor and 4 GB RAM

- Android Devices

### 3.2.2 Software specification:

- OS: Windows 10 or above

- Android Studio 3.0 or above

- Google Colab

- Android SDK version 5.02 or newer

- Java Development kit version 1.7 or newer

- Android Devices

## 3.3 ASSUMPTIONS AND CONSTRAINTS

### 3.3.1 Assumptions:

- Effective, efficient and transparent administration.

- Collaborative work environment.

- 24*7 access.

- Work/task prioritize

- Reusable code.

- Location independent.

- Maintenance and Testing is done.

### 3.3.2 Constraints:

- **User constraints:**

The application has been developed for Android based devices with good ram and processor. Also, the predicted currency is little bit low because of the mainly focus of the application because of time constraint, but it can be added later on.

- **Development** / **Hardware constraints**:

To train the model we have to use good GPU so that for the development and deployment we have to use good computer with better GPU.

# CHAPTER 4: SYSTEM ANALYSIS

13

## 4.1 STUDY OF PROPOSED SOLUTION

This application doesn't require any network connection. It will work as a device inhouse application. This is very useful since sometimes people need to predict data from documents while internet is not working, so this application will allow them to so

- Registering on Colab & Installing Android Studio

  Colaboratory, or "Colab" for short, is a product from Google Research. Colab allows anybody to write and execute arbitrary python code through the browser, and is especially well suited to machine learning, data analysis and education. Colab notebooks allow you to combine executable code and rich text in a single document, along with images, HTML, LaTeX and more.

  Android Studio is the official integrated development environment for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development

- Importing Datasets in Colab

  We make use of two different datasets to explore the forecasting power of two types of variables: market and fundamentals. We define a market variable as an indicator with daily to weekly frequency that has a close relationship with traded securities. We get datasets of different country from yahoo finance. We import datasets in colab.

Figure 4.1

```
from google.colab import drive
drive.mount('/content/drive')

Mounted at /content/drive
```

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
from seaborn import regression
sns.set()
plt.style.use('seaborn-whitegrid')

data = pd.read_csv("/content/drive/MyDrive/SGP/new data/INRCAD=X.csv")
data.dropna()
data.loc[~(data==0).all(axis=1)]
print(data.head())
```

```
         Date     Open     High      Low    Close  Adj Close
0  03-03-2017  0.02004  0.02009  0.02001  0.02005    0.02005
1  06-03-2017  0.02003  0.02018  0.02002  0.02003    0.02003
2  07-03-2017  0.02009  0.02015  0.02008  0.02010    0.02010
3  08-03-2017  0.02013  0.02017  0.02010  0.02013    0.02013
4  09-03-2017  0.02020  0.02029  0.02018  0.02020    0.02020
```

Figure 4.1 Dataset Import

- Training Model

For training model we have used design tree regression method to train the dataset for predicting currency. Decision tree regression observes features of an object and trains a model in the structure of a tree to predict data in the future to produce meaningful continuous output. Continuous output means that the output/result is not discrete, i.e., it is not represented just by a discrete, known set of numbers or values. By splitting data set we train our currency exchange rate prediction.

```
print(data.corr())
sns.heatmap(data.corr())
plt.show()
```

```
              Open      High       Low     Close  Adj Close
Open      1.000000  0.998912  0.998688  0.999945   0.999945
High      0.998912  1.000000  0.998582  0.998887   0.998887
Low       0.998688  0.998582  1.000000  0.998700   0.998700
Close     0.999945  0.998887  0.998700  1.000000   1.000000
Adj Close 0.999945  0.998887  0.998700  1.000000   1.000000
```
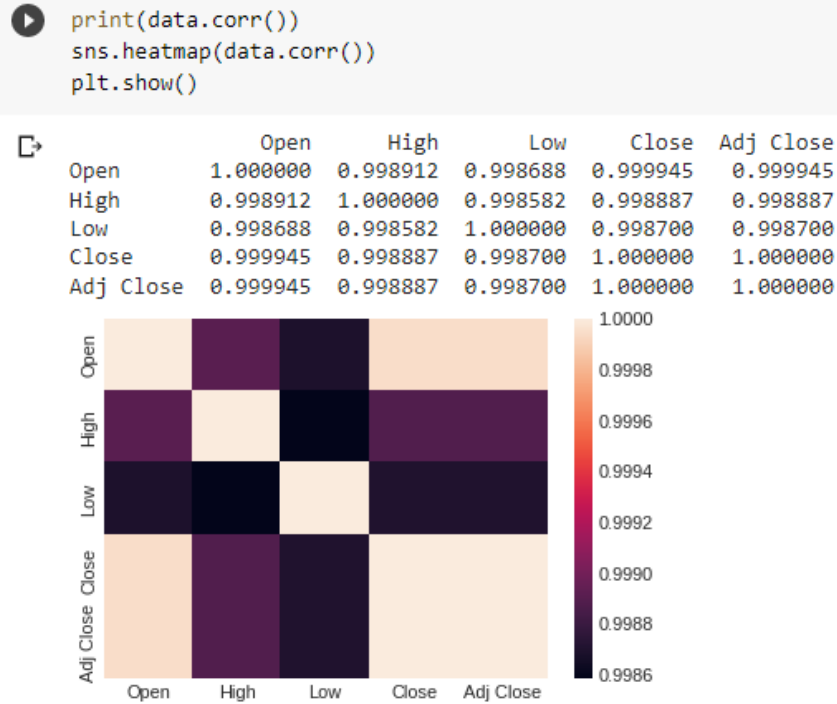
Figure 4.2 Analyze Data

- Downloading Predicted Data

  We analyze the predicted data and imported it into CSV file for application

Figure 4.3 Predicted Data

- Creating Android Application

  We created user interface with XML in android studio. With better user friendly interface. It has two options of selecting currency of which country they want to predict. It also has option to show graph of exchange rate of currency they selected.

- Creating API & Deploying Model

  We created CSV reader and stored it into array list to show in application. We have used sqlite for data base. We can stored predicted data in database. Which is also used for showing Graph of predicted currency to user. By that user can easily identify the exchange rate of currency.

  .

## 4.2 REQUIREMENTS OF PROPOSED SYSTEM

### 4.2.1 Functional Requirements

- Selecting base currency you want to predict
- Select currency you want to predict into.
- Importing Graph of predicted exchange rate

### 4.2.2 Non-Functional Requirements

- Robustness: Failures should be rare; a single failure should not bring the whole system down.
- The application should be able to run seamlessly with the user's phone. User should not experience any breakage or lagging that would affect user experience.
- processing time and ensure efficient memory management
- Extensibility: Ability to easily incorporate new functionalities to the workflow system.
- Reusability
- Minimum deviation
- Accuracy

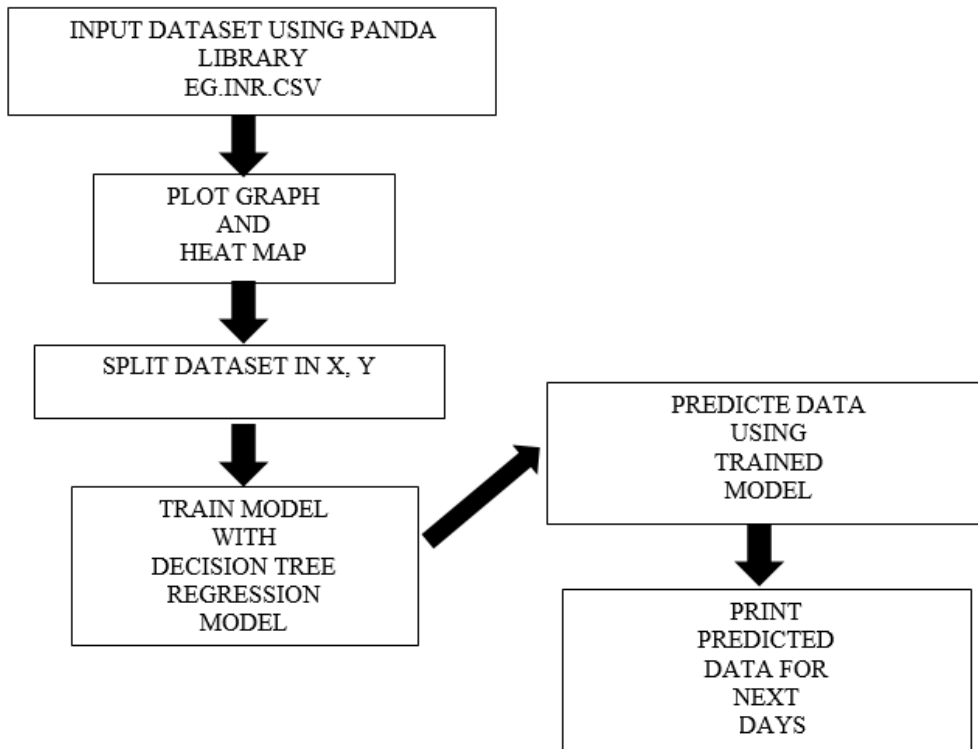## 4.3 SYSTEM DIAGRAMS

### 4.3.1 System Architecture

INPUT DATASET USING PANDA
LIBRARY
EG.INR.CSV

↓

PLOT GRAPH
AND
HEAT MAP

↓

SPLIT DATASET IN X, Y

↓

TRAIN MODEL
WITH
DECISION TREE
REGRESSION
MODEL

→

PREDICTE DATA
USING
TRAINED
MODEL

↓

PRINT
PREDICTED
DATA FOR
NEXT
DAYS

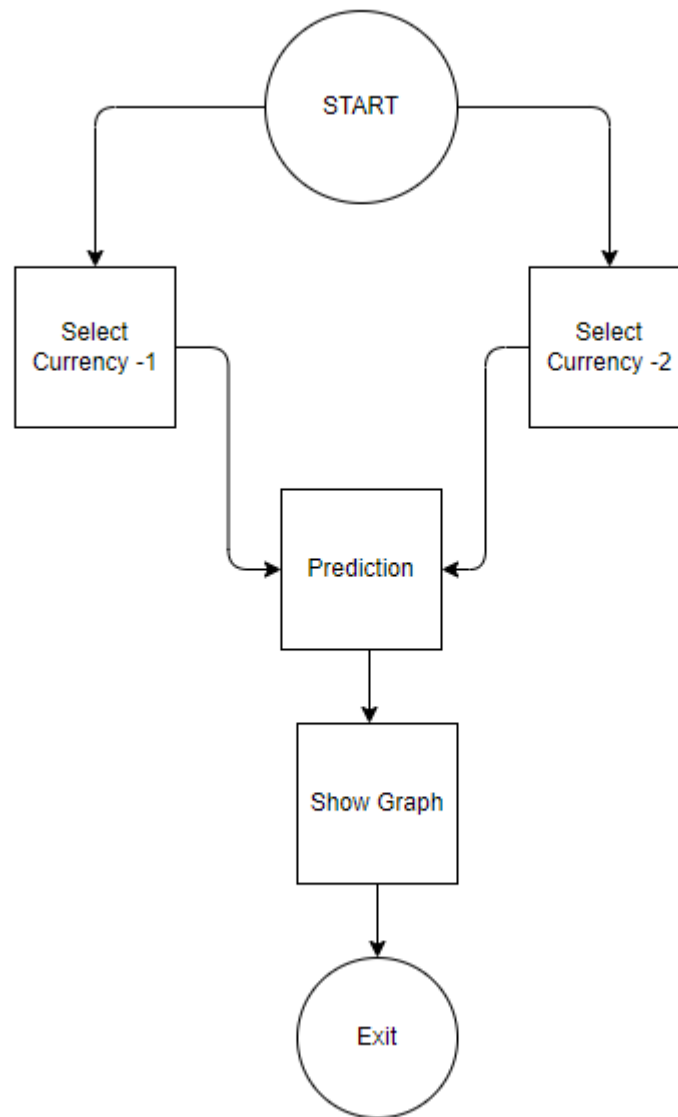Fig 4.4 System Architecture

**4.3.2 System Workflow**


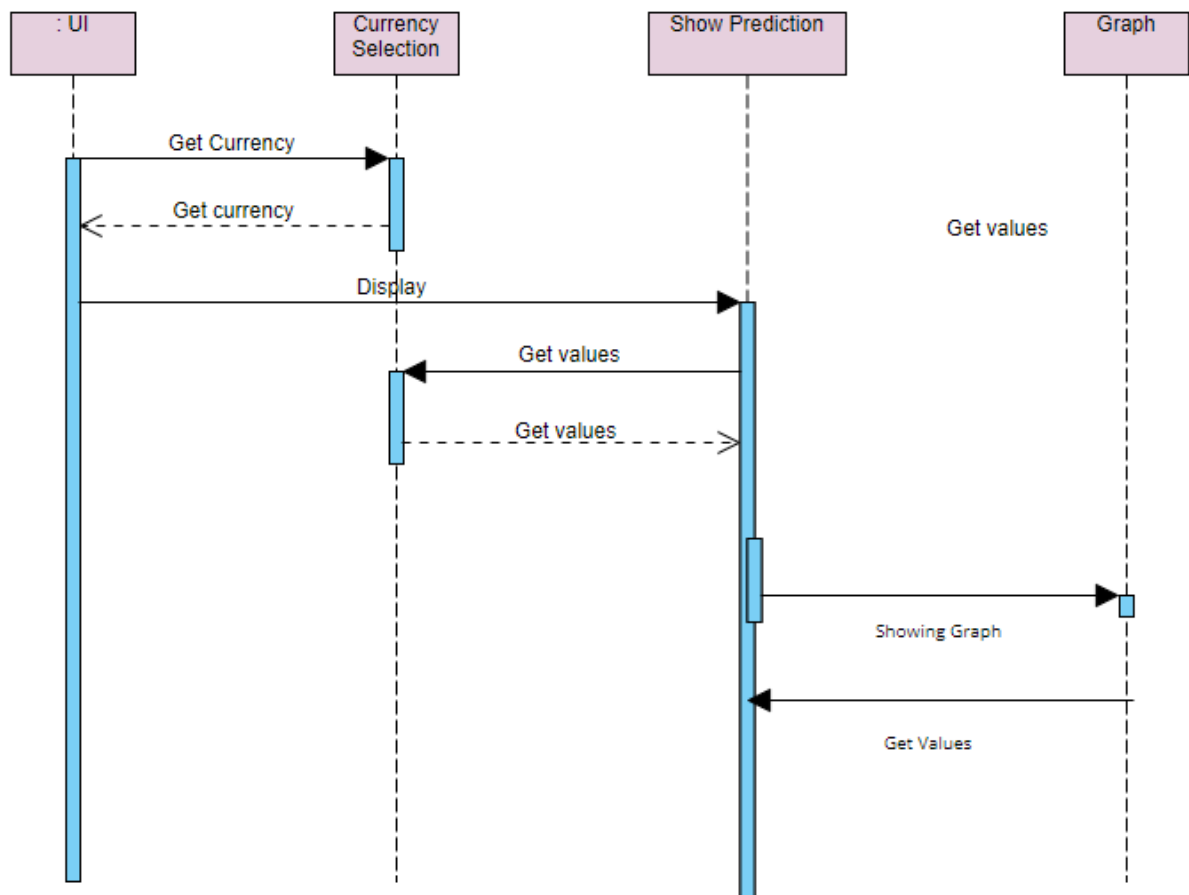
Fig 4.5 System Workflow

### 4.3.3 Sequence Diagram

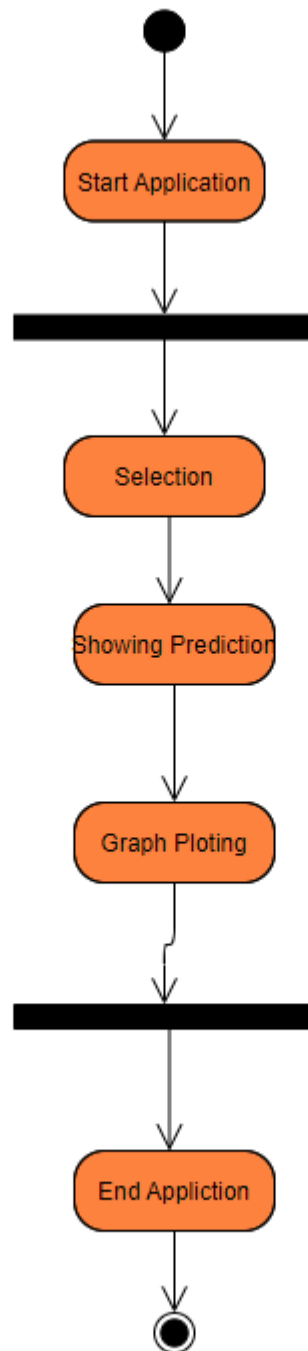

Fig 4.6 Sequence Diagram

**4.3.4 Activity Diagram**



Fig 4.7 Activity Diagram
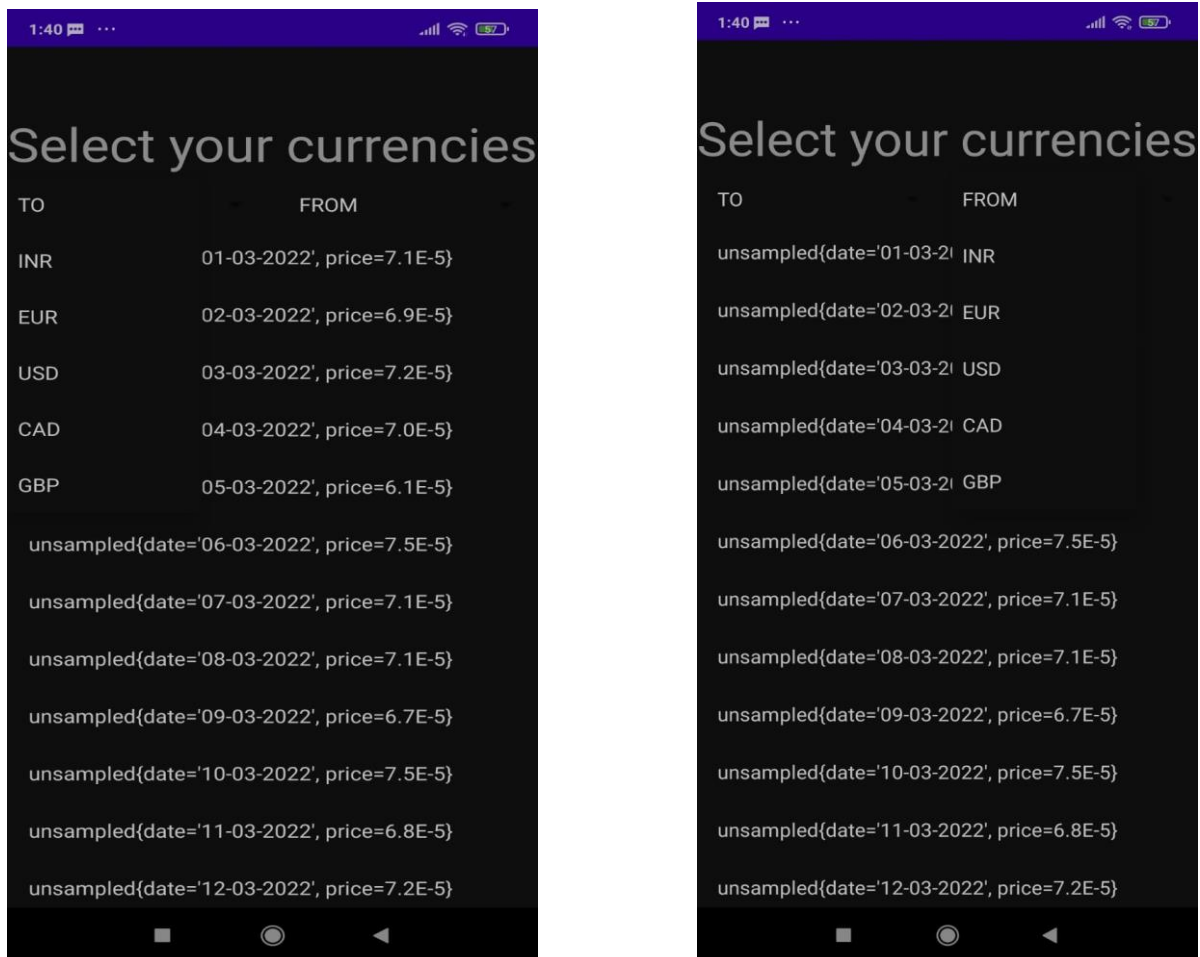
# CHAPTER 5: SYSTEM DESIGN

## 5.1 SCREEN LAYOUT



Fig 5.1 Main Screen UI

- This screen is Showing dropdown menu of currencies which you want to predict. First you have to select currencies according to your needs. It will give you different options for selection

Fig 5.2 Predicted Values

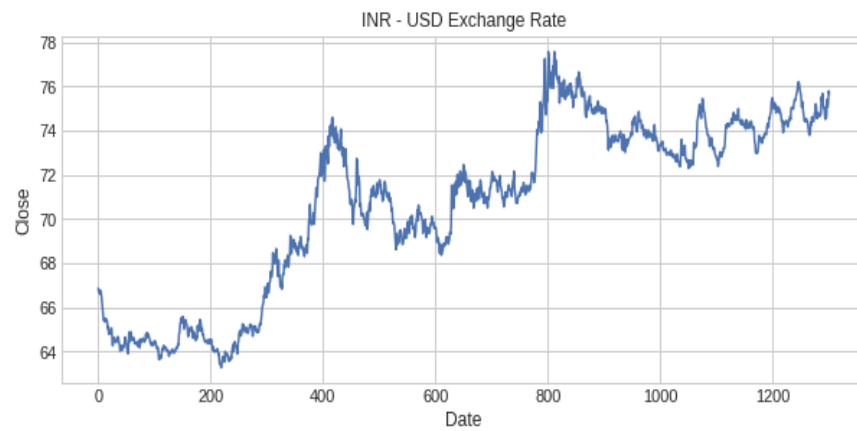- This will show predicted rates after selecting currencies. It has date wise option for exchange rate

Fig 5.3 Predicted Graph

- This graph will show you the variation of predicted values. By this you can easily analyze the data. Like it will get high or low on particular day.

# CHAPTER 6: SYSTEM IMPLEMENTATION AND TESTING

## 6.1 IMPLEMENTATION ENVIRONMENT

There can be various parameters for describing characteristics of the environment on which project implementation is done, two of the main parameters and its justification is mentioned below:

- **Single vs Multi-user**

    The software is single user, because only one user can use it at a time.

- **GUI vs Non GUI**

    The software is GUI based, it has 2D canvas window with texts and buttons.

## 6.2 CODING STANDARDS

- Use of CamelCase for naming of functions.
- Assets Management Folders are separately maintained for Prefabs, Scripts Models, Datasets, Margins etc.
- Reusable codes The scripting in the modules and functions are done in such a way that it can be REUSED in different scripts also.
- Comments are added wherever required, for easy understanding

Table 6.1 Class coding standard

| | **Part of Class/Interface Declaration Notes** | **Notes** |
|---|---|---|
| 1 | Class (static) variables | First the public class variables if any, then the protected, then package level (no access modifier), and then the private. |
| 2 | Instance variables | First public, then protected, then package level (no access modifier), and then private |
| 3 | Class (static) methods | First the public, then the protected, then package level (no access modifier), and then the privat |
| 4 | Class Methods | First public, then protected, then package level (no access modifier), and then private. |

## 6.3 TESTING METHOD

In this project, we have used the white box testing because the source code is completely developed by us and available to us. We have chosen this techniques because of following reasons:

- Easy to find out errors
- Make it efficient for use
- Optimize the code

Advantages:

- As the tester has knowledge of the source code, it becomes very easy to find out which type of data can help in testing the application effectively.
- It helps in optimizing the code.
- Extra lines of code can be removed which can bring in hidden defects.
- Due to the tester's knowledge about the code, maximum coverage is attained during test scenario writing.

## 6.4 TEST SUITES DESIGN

If each test case represents a piece of a scenario, such as the elements that simulate a completing a transaction, use a test suite. For instance, a test suite might contain four test cases, each with a separate test script:

- Test case 1: Working with Android versions
- Test case 2: Currency Prediction & Selection
- Test case 3: Prediction
- Test case 4: Exit Application

Test suites can identify gaps in a testing effort where the successful completion of one test case must occur before you begin the next test case. When you run a test suite in sequential mode, you can choose to stop the suite execution if a single test case does not pass. Stopping

the execution is useful if running a test case in a test suite depends on the success of previous test cases.

Test suites are also useful for the following types of tests –

- Build verification tests: A collection of test cases that perform a basic validation of most the functional areas in the product. The tests are executed after each product build and before the build is promoted for use by a larger audience.

- Smoke tests: A collection of test cases that ensure basic product functionality. Typically, smoke tests are the first level of testing that is performed after changes are made to the system under test.

- End-to-End integration tests: A collection of test cases that cross product boundaries and ensure that the integration points between products are exercised and validated.

- Functional verification tests: A collection of test cases that focus on a specific product function. Executing this type of test with a test suite ensures that several aspects of a specific feature are tested.

## 6.5 TEST CASES

A test case, in software engineering, is a set of conditions or variables under which a tester will determine whether an application, software system or one of its features is working as it was originally established for it to do. The mechanism for determining whether a software program or system has passed or failed such a test is known as a test oracle. In some settings, an oracle could be a requirement or use case, while in others it could be a heuristic. It may take many test cases to determine that a software program or system is considered sufficiently scrutinized to be released. Test cases are often referred to as test scripts, particularly when written - when they are usually collected into test suites.

TEST CASE ID: T01

Module Name: Working with Android versions

Pre-condition: Company's Logo and app should be loaded

Table 6.2 Test Case No. 01

| Test Case ID | Test Case (condition) | Expected Result | Actual Result | Pass/Fail |
|---|---|---|---|---|
| 1 | Company's logo should be loaded | Working fine on starting the app | Working fine on starting the app | Pass |
| 2 | Testing on different android version | Compatible with android Nougat version 7.0 or above | Compatible with android Nougat version 7.0 or above | Pass |

TEST CASE ID: T02

Module Name: Currency Selection

Pre-condition: no need for that

Table 6.3 Test Case No. 02

| Test Case ID | Test Case (condition) | Expected Result | Actual Result | Pass/Fail |
|---|---|---|---|---|
| 1 | Select Currency - 1 | Selected | Selected | Pass |
| 2 | Select Currency - 2 | Selected | Selected | Pass |

TEST CASE ID: T03

Module Name: Prediction

Pre-condition: Both Currency selection

Table 6.4 Test Case No 03

| Test Case ID | Test Case (condition) | Expected Result | Actual Result | Pass/Fail |
|---|---|---|---|---|
| 1 | Predict Currency | Rates on Different Date | Rates on Different Date | Pass |
| 2 | Show Graph | Graph Shown | Graph Shown | Pass |

TEST CASE ID: T04

Module Name: End Game | Pre-condition: Prediction should be done

Table 6.5 Test Case No 04

| Test Case ID | Test Case (condition) | Expected Result | Actual Result | Pass/Fail |
|---|---|---|---|---|
| 1 | When the Experience is over | Exit the application | Exit the application | Pass |

# CHAPTER 7: FUTURE ENHANCEMENT

- Optimization of the present code.

  Code optimization is any method of code modification to improve code quality and efficiency. A program may be optimized so that it becomes a smaller size, consumes less memory, executes more rapidly, or performs fewer input/output operations. The basic requirements optimization methods should comply with, is that an optimized program must have the same output and side effects as its non-optimized version.

- Better experience

  We want create more user friendly GUI in application. There will be better User Interface. We will add many more currencies so that other global users can also predict. We will give add to Favorite option to sort out currencies.

- Better Trained Model

  We will work on model training so that we can predict more accurate currency values. There will more accurate and precise graph of predicted currency.

# CHAPTER 8: CONCLUSION

## 8.1 SELF ANALYSIS OF PROJECT VIABILITIES

According to me, this project is completed with the primary functionalities as specified earlier, but then again there is a lot more than this which can be done. So, then it is a challenge to further develop it in to full-fledged software as it was challenge to develop up to this very stage. Due to lack of skilled knowledge, the project cannot be fully completed so far. This will help lot in in area of Machine learning & Android Studio

## 8.2 PROBLEM ENCOUNTERED AND THEIR SOLUTIONS

- **Lack of full knowledge about the technology**

   No clear ideas of how to train model better for more accurate prediction. Also we don't know about how to make better GUI in application.

- **Model Training**

   We had problem in Dataset which has null values. Because of that there was error in coding. We also face problem in accuracy of predicted values

- **Creating API & Deployment**

   First of all we don't know more about how create API and Deploy model on Android Studio. We also had problem in how import graph in application.

## 8.3 SUMMARY OF PROJECT WORK

This project has illustrated the feasibility of using methods for predicting FX rates. The models predicted the closing price with the highest mean error of 150 pips of the true price. Although impressive in-of-itself, for the application of intraday, this error would have to decrease by around a factor of ten. To achieve this more features need to be incorporated into the models. This will be achieved by first selecting more relevant features and automate the sourcing of the most current feature data using a web-scraping approach. The implementation of a distributed workload system will also be tested with the aim to improve efficiency of training. Once a larger amount of features are included, this is likely to be key to making the computations tractable. Lastly, the methods are currently just predicting the closing price, but we are of course interested in placing positions to maximize yield. These aspects of FX trading will also be included in the later code development to take one step closer to live autonomous trading.

# BIBLIOGRAPHY

1) *https://www.javatpoint.com/artificial-neural-network*

2) *https://www.youtube.com/watch?v=iZFGBF9Q9aA*

3) *https://www.geeksforgeeks.org/machine-learning/*

4) *https://www.analyticsvidhya.com/blog/2020/03/google-colab-machine-learning-deep-learning/*

5) *https://towardsdatascience.com/exchange-rate-prediction-machine-learning-with-5-regression-models-d7a3192531d*

6) *https://www.javatpoint.com/android-tutorial*

7) *https://www.analyticsvidhya.com/blog/2021/11/how-to-deploy-machine-learningml-model-on-android/*

8) *https://developer.android.com/ml*

9) *https://finance.yahoo.com/*