CHAPTER 5

IMPLEMENTATION

Implementation is the process of converting a new system design into an operationalone. It is the key stage in achieving a successful new system. It must therefore be carefully planned and controlled.

5.1 Steps for Implementation of python Application

5.1.1 Creating a New Project

To create a new project:

1. Select File > New > Project.

2. Click on selected project Project > New File.

3. Select the contents for the project:

> a. Enter a Project Name. This will be the name of the folder where your project is created.
>
> b. Under Contents, select Create new project in workspace. Select your project workspace location.
>
> c. Under Target, select an Android target to be used as the project's Build Target. The Build Target specifies which Android platform you'd like your application built against.
>
> d. Under Properties, fill this all necessary fields.
>
> > • Enter an Application name. This is the human-readable title for your application- the name that will appear on the Android device.
> >
> > • Enter a Package name. This is the package namespace (following the same rules as for packages in the Java programming language) where all your source code will reside.
> >
> > • Select Create Activity (optional, of course, but common) and enter a name for your main Activity class.

5.1.2 **Procedure to Run the Application**

- **Run the Application**

    To run (or debug) your application, select Run > Run (or Run > Debug) from the Pycharm main menu. The ADT plug in will automatically create a default launch configuration for the project. When you choose to run or debug your application, Eclipse will perform the following:

    1. Compile the project (if there have been changes since the last build).

    2. Create a default launch configuration (if one does not already exist for the project).

    3. Install and start the application on an emulator or device.

5.2 Implementation Issues

The implementation phase of software development is concerned with translating design specifications into source code. The primary goal of implementation is to write source code and internal documentation so that conformance of the code to its specifications can be easily verified and so that debugging testing and modification are eased. This goal can be achieved by making the source code as clear and straightforward as possible. Simplicity clarity and elegance are the hallmarks of good programs and these characteristics have been implemented in each program module.

The goals of implementation are as follows.

- Minimize the memory required.
- Maximize output readability.
- Maximize source text readability.
- Minimize the number of source statements.
- Minimize development time.

5.2 **Tools need to be installed**

- Selenium
    - o Selenium is installed by following command

      sudo pip install selenium

- Pandas
    - o Pandas is installed by following command

      sudo pip install pandas

- nltk(Natural Language Toolkit)
    - o nltk is installed by following command

      sudo pip install nltk

5.4 Modules Implemented

5.4.1 Web Data Extraction

- Tools used : selenium
- In this module the investors' reviews about stocks are extracted from web through selenium tool.
- The company data is acquired by developers and stored in csv file.
- Application uses these url to extract investors comments about the stocks from web.
- Selenium creates session for each url of company.
- Project is specific to the following website.

  https://mmb.moneycontrol.com/forum-topics/stocks/company_name.html

	A	B
1	Symbol	Url
2	ACC	https://mmb.moneycontrol.com/forum-topics/stocks/acc-865.html
3	APOLLO	https://mmb.moneycontrol.com/forum-topics/stocks/apollo-tyres-9282.html
4	AXISBANK	https://mmb.moneycontrol.com/forum-topics/stocks/axis-bank-3142.html
5	BANKBARODA	https://mmb.moneycontrol.com/forum-topics/stocks/bankbaroda-261.html
6	BBL	https://mmb.moneycontrol.com/forum-topics/stocks/rbl-bank-510311.html
7	CESC	https://mmb.moneycontrol.com/forum-topics/stocks/cesc-48181.html
8	EXCELCROP	https://mmb.moneycontrol.com/forum-topics/stocks/excel-crop-care-119095.html
9	HCLTECH	https://mmb.moneycontrol.com/forum-topics/stocks/hcl-tech-682.html
10	HDFC	https://mmb.moneycontrol.com/forum-topics/stocks/hdfc-bank-4962.html
11	HEXAWARE	https://mmb.moneycontrol.com/forum-topics/stocks/hexaware-tech-2745.html
12	HFCL	https://mmb.moneycontrol.com/forum-topics/stocks/hfcl-2961.html
13	HIL	https://mmb.moneycontrol.com/forum-topics/stocks/hil-37825.html
14	HILTON	https://mmb.moneycontrol.com/forum-topics/stocks/hilton-metal-246743.html
15	ICICIBANK	https://mmb.moneycontrol.com/forum-topics/stocks/icici-bank-6422.html
16	IDEA	https://mmb.moneycontrol.com/forum-topics/stocks/idea-cellular-246717.html
17	ITI	https://mmb.moneycontrol.com/forum-topics/stocks/iti-12542.html
18	JBMA	https://mmb.moneycontrol.com/forum-topics/stocks/jbm-auto-243555.html
19	JETAIRWAYS	https://mmb.moneycontrol.com/forum-topics/stocks/jet-airways-219558.html
20	JKTYRE	https://mmb.moneycontrol.com/forum-topics/stocks/jk-tyre--ind-8382.html

Figure 5.1 Url list of companies

Pseudo code for selenium for data extraction

browser = webdriver.Chrome('/home/raghu/Pictures/chromedriver')

```
for j in range(len(df)):
        print("Browser Opened")
        browser.get(df.loc[j, "Url"])
        print (df.loc[j, "Symbol"],"Company Forum loaded on browser")
        time.sleep(random.randint(5, 10))
        print("Scroll to y=2000 axis down")
        browser.execute_script("window.scrollTo(0, 2000)")
        time.sleep(random.randint(5, 10))
        print("Testing stop")
        alldata = []
        allData = browser.find_elements_by_class_name("rht_content")
        print(allData)
```

5.3.2 Data Pre-processing

- Tools used : nltk

- The extracted selenium data need to be processed so that we only have todays investors review.

Pseudo code for processing todays data

```
for i in allData:

        comment = i.find_element_by_class_name("txt16gry")

        #print(comment)

        date = i.find_element_by_class_name("link13gry")

        #print(date)

        try:

                c = comment.find_element_by_tag_name('a')

                comments.append(c.text)

                dates.append(date.text)

        except:

                print ("No Recommendation.")

        continue
```

5.3   Application uses predefined keywords to identify the stock

lists = [

'Agent', 'Ask/Offer', 'Assets', 'At the money', 'Bear Marke',

'Beta', 'Bid', 'Blue Chip Stock', 'Board Lot', 'Bonds:',

'Book', 'Broker/Brokerage Firm', 'Bull Market',

'Business Day', 'Call Option', 'Close Price', 'Commodities',

'Convertible Securities', 'Debentures', 'Defensive Stock',

'Delta', 'Derivatives', 'Diversification', 'Dividend', 'Equity',

'Bear Market', 'Beta', 'Blue Chip Stocks', 'Bourse'

]

- These keyword are processed by nltk tool kit to produce lemmas.

```
wordnet.lemmas("sell")
[Lemma('sell.n.01.sell'),
 Lemma('sell.v.01.sell'),
 Lemma('sell.v.02.sell'),
 Lemma('sell.v.03.sell'),
 Lemma('deal.v.06.sell'),
 Lemma('sell.v.05.sell'),
 Lemma('sell.v.06.sell'),
 Lemma('sell.v.07.sell'),
 Lemma('betray.v.02.sell')]
```

Figure 5.2 Lemmas of keyword 'sell'

5.3.3 Sentimental Analysis

- Sentences are selected only if atleast one of the predefined keyword is present.
- Sentiment   scores   is   calculated   for   selected   sentences   athrough SentimentIntensityAnalyzer.
- Scores are called polarity scores.

```
>>> sid.polarity_scores("Good profit")
{'neg': 0.0, 'neu': 0.0, 'pos': 1.0, 'compound': 0.7003}
```

Figure 5.3 Polarity score example

- For each keyword the polarity scores are calculated.

- Rating is given based on
- If sentence is positive then rating of that keyword is incremented
- If not then it is a negative sentence hence ratings of that keyword decremented.
- Ratings of all keyword calculated.

5.3.4 Classifier

- Now the system selects all positive/negative/neutral keyword and their ratings are added.

- If a company has more positive rating then its stock can be brought.

- If a company has more negative rating then stock can be sold.

- Otherwise hold the stocks.

5.3.5 Complete Methodology

  i.   Store the company Url in csv file.
  ii.   Connect to the website by selenium tool.
  iii. Scrap the webpages having unstructured data through selenium tool to get semi structured data.
  iv.   Parse these semi structured data to get Data relevent to the Application.
  v.   Classify each comments as positive, negative or neutral.
  vi.   Apply Sentiment Analysis on refined data to get score which determines which is best choice(buy,sell,hold) for stocks.
  vii. Provide Result in CSV file.