FAST-NUCES
Department of Computer Science
BS-CS

What's Trending?

Final Year Project
Project-II CS 4092

Talha Ahmad	19F-0113
M. Talha Shehroze	19F-0171
Muhammad Farhan	19F-0254

Project Supervisor:	Mr. Rizwan ul Haq
Project Co-supervisor:	Mr. Sajid Anwer

National University of Computer and Emerging Sciences

Chiniot–Faisalabad Campus

Anti-Plagiarism Declaration

This is to declare that the above publication produced under the:

Title: What's Trending? is the sole contribution of the author(s) and no part hereof has been reproducedon **as it is** basis (cut and paste) which can be considered as **Plagiarism.** All referenced parts have been used to argue the idea and have been cited properly. I/We will be responsible and liable for any consequence if violation of this declaration is determined.

Date: ______________

Student 1

Name: Talha Ahmad

Signature: ___________________

Student 2

Name: M. Talha Shehroze

Signature: ___________________

Student 3

Name: Muhammad Farhan

Signature: ___________________

Intellectual Property Right Declaration

This is to declare that the work under the

Title: What's Trending?

Conducted in partial fulfillment of the requirements of:

o Course Project o BS FYP o MS Thesis

This is the sole property of the National University of Computer and Emerging Sciences and is protected under the intellectual property high laws and conventions. It can only be considered/used for purposes like an extension for further enhancement, product development, adoption for commercial/organizational usage, etc., with the permission of the University. The above statement applies to all students and faculty.

Date: ______________

Student 1

Name: Talha Ahmad

Signature: ___________________

Student 2

Name: M. Talha Shehroze

Signature: ___________________

Student 3

Name: Muhammad Farhan

Signature: ___________________

Supervisor (Faculty)

Name: Mr. Rizwan ul Haq

Signature: ___________________

Acknowledgment

Firstly, we would like to pay our all gratitude and thanks to Almighty ALLAH for giving us the courage to undertake this task. We would also like to thank the Department of Computer Science of FAST-NUCES, Chiniot-Faisalabad Campus, for providing us with the opportunity to be a part of this reputed organization, gain some experience and groom ourselves for future professional responsibilities.

We are very grateful to Mr Rizwan ul Haq our respectable Supervisor and Mr Sajid Anwer our respectable Co-supervisor for their continuous help, support, and time.

Table of Contents

Tabular Contents

Table of Figures

1. Introduction

What's Trending? is to educate and assist users in distinguishing between what they consume on Twitter. The web platform deals with trending keywords. Users will be able to identify organic trends from the forced or paid trends before they form an opinion. The Application will provide the user with an interface to view trending keywords, further he can manually add any keyword and check on them if the data is available for that specific word. Also, The user will also have an option to view statistical analysis on some Twitter account or on the trending keyword.

In this document, we will briefly explain the features of What's Trending? The document also explains the software requirement specifications. We will deal with all the design related issues as well as the requirements together with the scope of this project. This document will be continuously updated and on completion with be referred throughout the software development.

2. Vision Document

In this section, we discuss project vision in detail.

2.1. Problem Statement

Table 1: Problem Statement

Problem of	distinguishing right and wrong, real and fake, authentic and spam, unique, and bot tweets and thus the trends in real time.
Affects	both businesses and general public.
Impact of which	there is no product that can be used to distinguish between genuine and spam tweets, making it impossible to distinguish between genuine trends and spammy ones.

A Successful solution would be	A platform that can authenticate users, retrieve tweets from Twitter based on keywords, display results graphically, compare data, and statistically differentiate spam from legitimate tweets will be a successful solution.

2.2. Business Opportunity

People in the present period use social media, particularly Twitter, to find out what other people are thinking and talking about so they may quickly gain knowledge, comprehend, defend, and create opinions. People can learn more about what's going on in their area thanks to this.

There are some products out there that somewhat relates to what we do, but they have relatively few features. The product that is currently on the market has several restrictions. Firstly, none of them carry out our features exactly. Additionally, they do not classify and evaluate Twitter trends and rate them based on actual/spam content. Although some software offers one or two of the features, there isn't a single solution that offers all of them. This creates a significant issue for businesses and the general public who lack access to a specific platform that can be used to complete all the stated activities.

2.3. Objectives

The main Objective of What’s Trending? Is to provide people a quick and accurate overview of trends, so that they can analyse trends just at glance rather than spending time to manually analyse.

2.4. Scope

What’s Trending? Is web application. The main problem we are currently facing is that there is no platform available where people can check quality of trend, analyse trend. The impact of it is that people participate in fake/low quality trend. They consider information to be legit which was spreading through these trends. WhatsTrending? will provide an environment through which be able to analyse these trends before taking any action.

2.5. Constraints

Firstly, we assume that internet will work fine all the time to load web page. The website does not require any hardware development or procurement.

2.6. Stakeholder and User Description

As there are projects which are analysing people profiles, but they are not providing analysis of trend. The differentiating edge of this product is that application will be in real-time. The potential users of this application will be all the literate people who know the usage of web-based application and twitter.

2.6.1. Market Demographics

Our target market includes the general public, businesses, and all English-speaking Twitter users. In order to give them a better platform for unique and valid content and enable them to absorb just the real information.

Everyone is turning away from traditional sources of information and migrating toward social media, particularly Twitter.

It can be used by PTA and possibly Twitter to cut down on spam and teach users on how to make better decisions, but for this project, the intended audience is the general public.

2.6.2. Stakeholder Summary

Table 2: Stakeholder Summary

Name	Description	Responsibilities
Developer	The developer who are developing the system.	Design the flow of system development Monitor progress of development

Tester	Testing system	Removing bugs from system
Trainer	Train system on data	Check and increase accuracy of system on result
Maintainer	Provide maintenance of system	Adding, removing, or modifying feature of system.

2.6.3. User Summary

Table 3: User Summary

Name	Description	Responsibilities	Stakeholder
Actor	Person Registered	For first time user need to register on system. Otherwise, login. Can interact with application interface	Development team, User
Secondary Actor	System	Acquired data from twitter through API, Trend analysis	The end user who are getting result about trend

2.6.4. User Environment

As this is a one-year project, we are developing a web-based application which will analyse trend and will predict quality of trend. There are 3 members in our group who are working on this project.

The project has been divided into 5 different stages. In September, October we are doing Requirement Specifications, October and November are for Design and Analysis, November to February is for Implementation, February and March are for Testing, March and April are for Deployment.

2.6.5. Stakeholder Profiles

Table 4: Stakeholder Profiles

Representative	We are three members working on this project. The stakeholder of system include developer, tester, trainer, maintainer.
Description	Developer is managing requirement, planning thing according to it and writing code for application. Developer is managing requirement, planning thing according to it and writing code for application. Tester will test system and make sure that system is bug free. Trainer will train system on dataset and will make sure its accuracy of result. Maintainer will maintain the product after its development and will add or remove feature.

Type	Our group of three members are working on this project. We have divided work into small task and task was done by all of us.
Responsibilities	Develop a system which help people to analyse trend. Providing easily understandable interface to user.
Success Criteria	Firstly, system will acquire data related to trend using Twitter API, then according to option selected by user system will provide accurate result on real time data.
Involvement	All stakeholder is participating in development of system.
Deliverables	Project deliverable will be same as mentioned.
Issues	We have to face issue of identification of bot account while system training.

3. System Requirement Specification

In this section, features and requirements of the system are explained.

3.1. System Features

- User authentication
- Analyze trends
- Developer Portal
- Data visualization
- Bot account recognition in trend
- Statistical analysis of account

3.2. Functional Requirements

The functional requirements of the system are listed below.

3.2.1. User Authentication

A graphical interface will be provided to the user for login. New users will sign up for an account. Each user will be provided with a recommendation according to his added keyword.

3.2.2. Trend Quality

The Keywords provided by the user system will predict Trend quality based on factors i.e., unique account participation, unique message, analyzing trend starter accounts, checking bot accounts participation, and some other factors by using AI, and ML techniques.

3.2.3. Trend Analysis

The keyword entered by the user for statistical analysis will predict the trend timeline, growth rate, growth region, activity graph, popular tweet, and popular retweet.

3.2.4. Bot Account recognition

The username will be provided by the user; The system will identify whether the participating account was operated by a fake, bot, or authentic account.

3.2.5. Statistical analysis of Twitter account

The username will be provided by the user; the system will provide a detailed overview of that Twitter account.

3.2.6. Data Visualization

The system will provide frequency analysis and visualize data. It will include the frequency of total account participating, real account, bot account, total tweets made, unique tweets, unique accounts participating in the keyword, number of times the keyword is used, same message retweet, gender ratio, age ratio, and platform ratio.

3.2.7. Developer API Portal

System will provide a portal to developer where they provide query to system and system show result in JSON format.

3.3. Non functional requirement

The functional requirements of the system are listed below.

3.3.1. Performance

The basic functionality of What’s Trending? is its performance, how system will perform, its throughput, accuracy, response time. It’s crashing or failure chance will be very low. It should quickly respond to the user.

3.3.2. Availability

The system should be available 24/7.Every time authenticated user made request service will be provided to him.

3.3.3. Robustness

What’s Trending? should allow user to achieve their desired result.

3.3.4. Security

There will be no issue of data security. The username and password would be required for authentication. The user who has register on website would be given permission to access made request.

3.3.5. Maintainability

The software should be maintainable. This means that the proper Object oriented principle should be applied so that in future, if there is any need to update a particular module than the developer needs to modify only specific classes.

3.3.6. Capacity

What's Trending? Will be able to support multiple users at a time.

4. Use Case Diagram

A use case diagram is a graphical depiction of a user's possible interactions with a system. Use case diagrams specify how the system interacts with actors without worrying about the details of how that functionality is implemented.

Below uses case diagram show user interaction with functionalities that will be provided by what's trending.

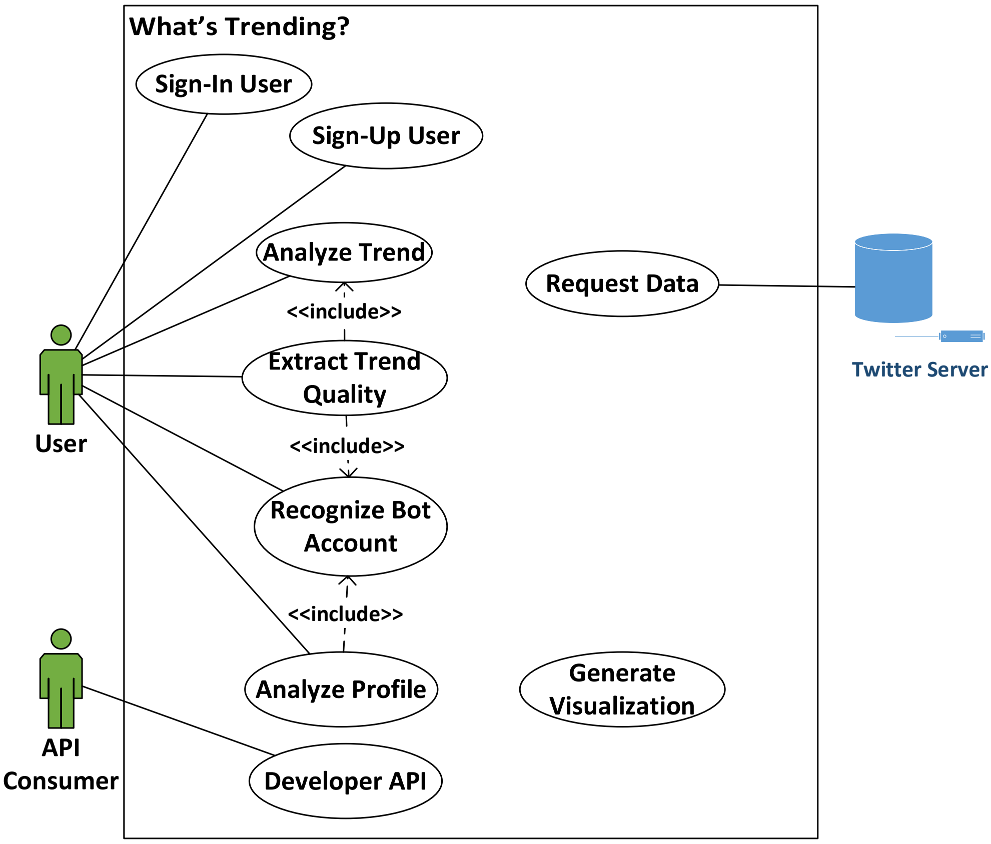


Figure 1: Use Case Diagram

5. Expanded Use Case

This description is more detailed and structured than the high-level use case description. It includes what happens to initiate the use case, actors involved, what data has to be input, use case output and alternate flow.

5.1. Extract Trend Quality

Following is brief description of Extract Trend Quality use case, that describe conditions and step that user must fulfill to get result and alternate flow if he faces problem.

Table 5: Expanded Use Case of Extract Trend Quality

Use Case Name	Extract Trend Quality
Actor	User
Description	The Keywords provided by the user, the system will predict Trend quality in terms of Authentic, Fabricated, or intermediate.
Pre-condition	The user must provide a keyword to the system, so the system can get the required data.
Post-condition	The System will provide the prediction of whether the trend is authentic or not.
Normal flow	1. The user selects the option of Trend Quality. 2. Users enter a keyword manually or from the provided keyword list. 3. The system will perform Trend Analysis. 4. The system will classify the trend. 5. The system will predict trend authenticity.
Alternate flow	5.a System is unable to perform operation due to insufficient data. Try again with another keyword with sufficient data.
Frequency of Use	Normal
Assumption	Users want to know the quality of trend of specific keyword.

5.2. Analyze Trend

Following is brief description of Analyze Trend use case, that describe conditions and step that user must fulfill to get result and alternate flow if he faces problem.

Table 6: Expanded Use Case of Analyze Trend

Use Case Name	Analyze Trend
Actor	User
Description	The Statistical method will be applied on trend to extract information. The system will show the result in graphical form.
Pre-condition	The user must provide a keyword to the system, so the system can get the required data.
Post-condition	The meaningful information on the requested trend will be shown to the user in graphical form.
Normal flow	1. Users enter a keyword manually or from the provided keyword list. 2. The user selects the option of Trend Analysis. 3. The system will apply Statistical operation on a trend. 4. The results will be shown to the user in graphical form.
Alternate flow	4.a System is unable to do trend analysis due to insufficient data. Try again with another keyword with sufficient data.
Frequency of Use	High
Assumption	The user wants an overall overview of the trend.

5.3. Recognize Bot Account

Following is brief description of Recognize Bot Account use case.

Table 7: Expanded Use Case of Recognize Bot Account

Use Case Name	Recognize Bot Account
Actor	System, User
Description	The user can request for Bot account recognition, or the System requires it during Profile Analysis.
Pre-condition	User must be logged in.
Post-condition	User/System will be notified if the account is recognized as bot or not.
Normal flow	1. System or User provides the profile's username. 2. Requests for bot account recognition. 3. The system performs analysis and displays outcome.
Alternate flow	3.a System is unable to display result because no user exists with provided username. Renter corrects username and try again. 3.b System is unable to display result due to lack of data. Try again with another username.
Special Requirement	Availability of Internet
Assumption	The User or System wants to know if an account is operated by bot or actual user.

5.4. Analyze Profile

Following is brief description of Analyze profile use case, that describe conditions and step that user must fulfill to get result and alternate flow if he faces problem.

Table 8: Expanded Use Case of Analyze Profile

Use Case Name	Analyze Profile
Actor	User
Description	The user can request for statistical analysis of any profile.
Pre-condition	User must be logged in.
Post-condition	User will get a report about the Profile’s Statistical Analysis.
Normal flow	1. The user requests for Account statistical analysis. 2. User enters the profile’s username. 3. The system performs statistical analysis and displays outcome.
Alternate flow	3.a System is unable to display result because no user exists with provided username. Renter corrects username and try again. 3.b System is unable to display result due to lack of data or account is set to private. Try again with another username.
Frequency of Use	Normal
Assumption	The user wants to know the statistical details about any user’s profile.

5.5. Generate Visualization

Following is brief description of Generate visualization use case, that describe step and condition that user should do and then system show result to user.

Table 9: Expanded Use Case of Generate Visualization

Use Case Name	Generate Visualization
Actor	User
Description	After successful statistical operation, system will show result in graphical form.
Pre-condition	System has successfully completed any desired operation.
Post-condition	The user will be provided with result in graphical form.
Normal flow	1. The user enter keyword manually or from provided keyword list. 2. The user selects any desired operation. 3. Data in graphical form is shown to user.
Alternate flow	3.a system is unable to show result to user due to insufficient data. Try again with keyword that have sufficient data.
Frequency of Use	High
Assumption	The user wants result in graphical form.

5.6. Developer API

Following is brief description of Developer API use case, that describe conditions and steps that user must fulfill to get result and alternate flow if he faces problem.

Table 10: Expanded Use Case of Developer API

Use Case Name	Developer API
Actor	API Consumer
Description	The third-party user will request the System to perform one of provided the use-case.
Pre-condition	The Request is valid. The Twitter Access keys are valid.
Post-condition	The system will respond according to request.
Normal flow	1. The Programmer will request the system. 2. The request is Valid. 3. The system will request data from Twitter API related to the keyword. 4. The system will perform operations according to request. 5. The system will respond according to request.
Alternate flow	2.a Request is invalid. Try again with correct request. 4.a system is unable to perform operation due to insufficient data. Try request with another keyword.
Frequency of Use	High
Assumption	Users want to know the quality of the trend of the related keyword.

6. Component Diagram

A component diagram breaks down the actual system under development into various high levels of functionality. A component defines its behavior in terms of provided and required interfaces.

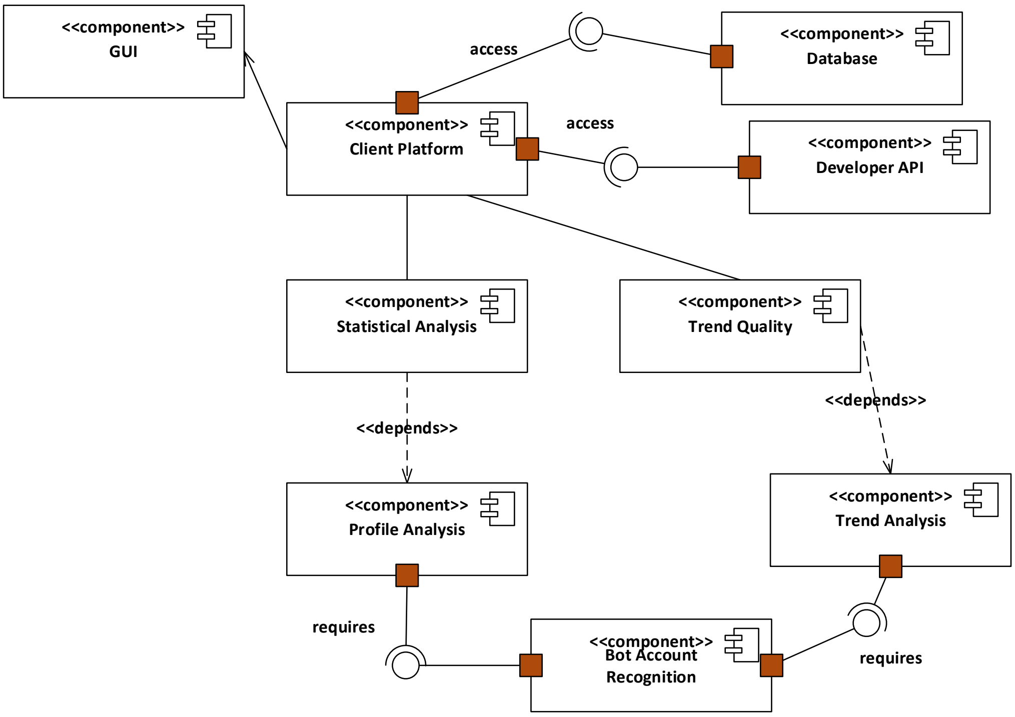


Figure 2: Component Diagram

7. Data Flow Diagram

Data-flow diagrams are system models that show a functional perspective where each transformation represents a single function or process. DFDs are used to show how data flows through a sequence of processing steps.

The figure below shows Classic DFD that is drawn for what's trending. It shows user that who will interact with system. In between user and process, there is data flow that indicate the existence of information exchange between user and system.

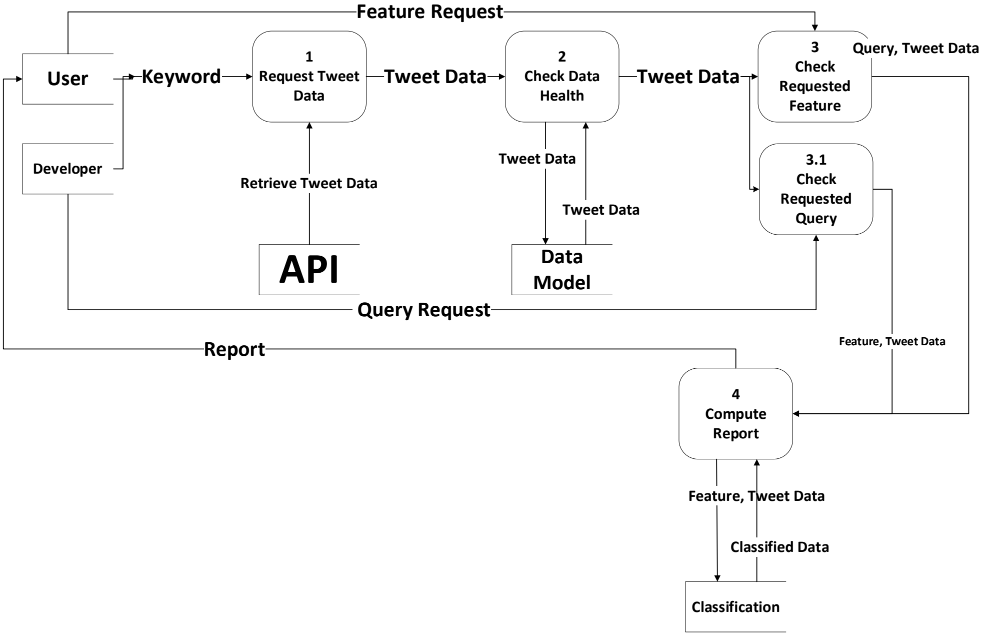


Figure 3: Classic DFD in Gane-Sarson notation

8. Package Diagram

Package diagrams are often used to illustrate the logical architecture of a system—the layers, subsystems, packages. package diagram provides a way to group elements.

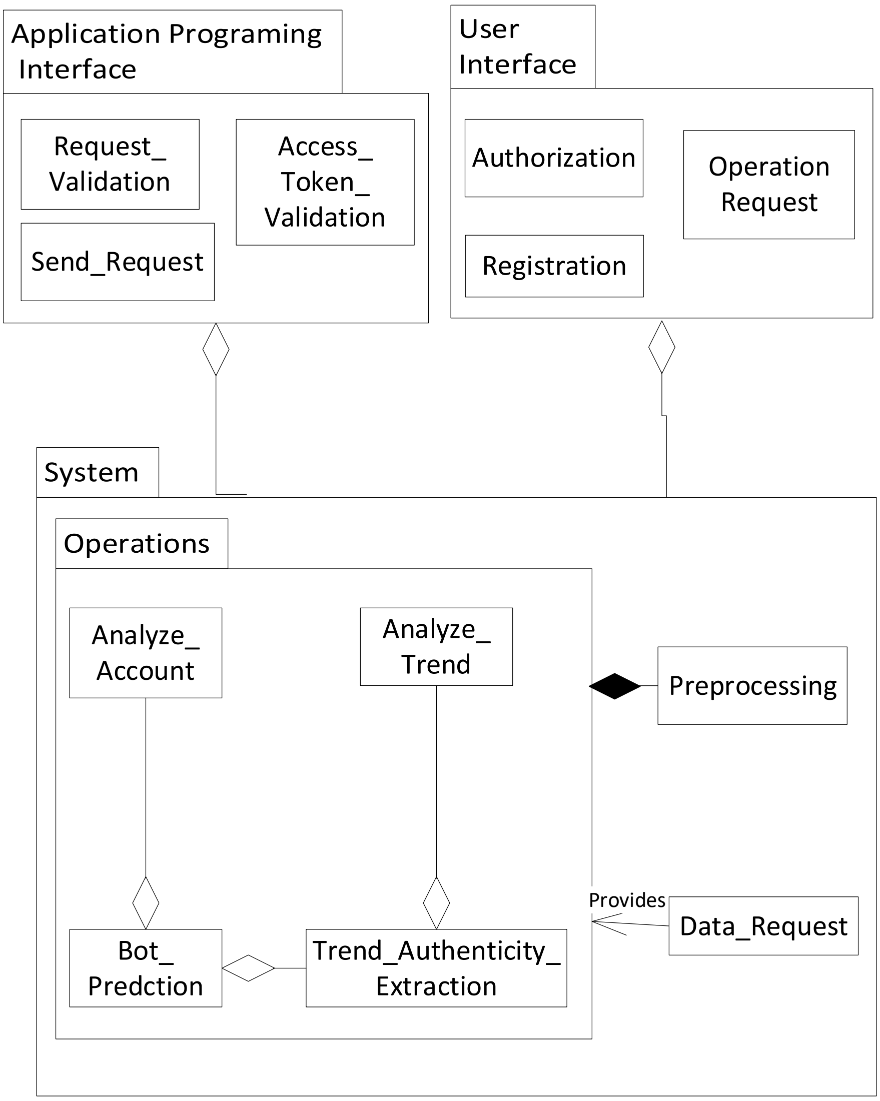


Figure 4: Package Diagram

9. State Machine Diagram

A state machine diagram shows the lifecycle of an object: what events it experiences, its transitions, and the states it is in between these events. The figure below shows state machine that is drawn for what's trending. It shows transition of state on user interaction with system. Initially system is in idle state, after user authentication state system enter dashboard state. If user select simple user, system goes into simple user state, otherwise system goes to developer portal state. In both state it waits for user interaction.

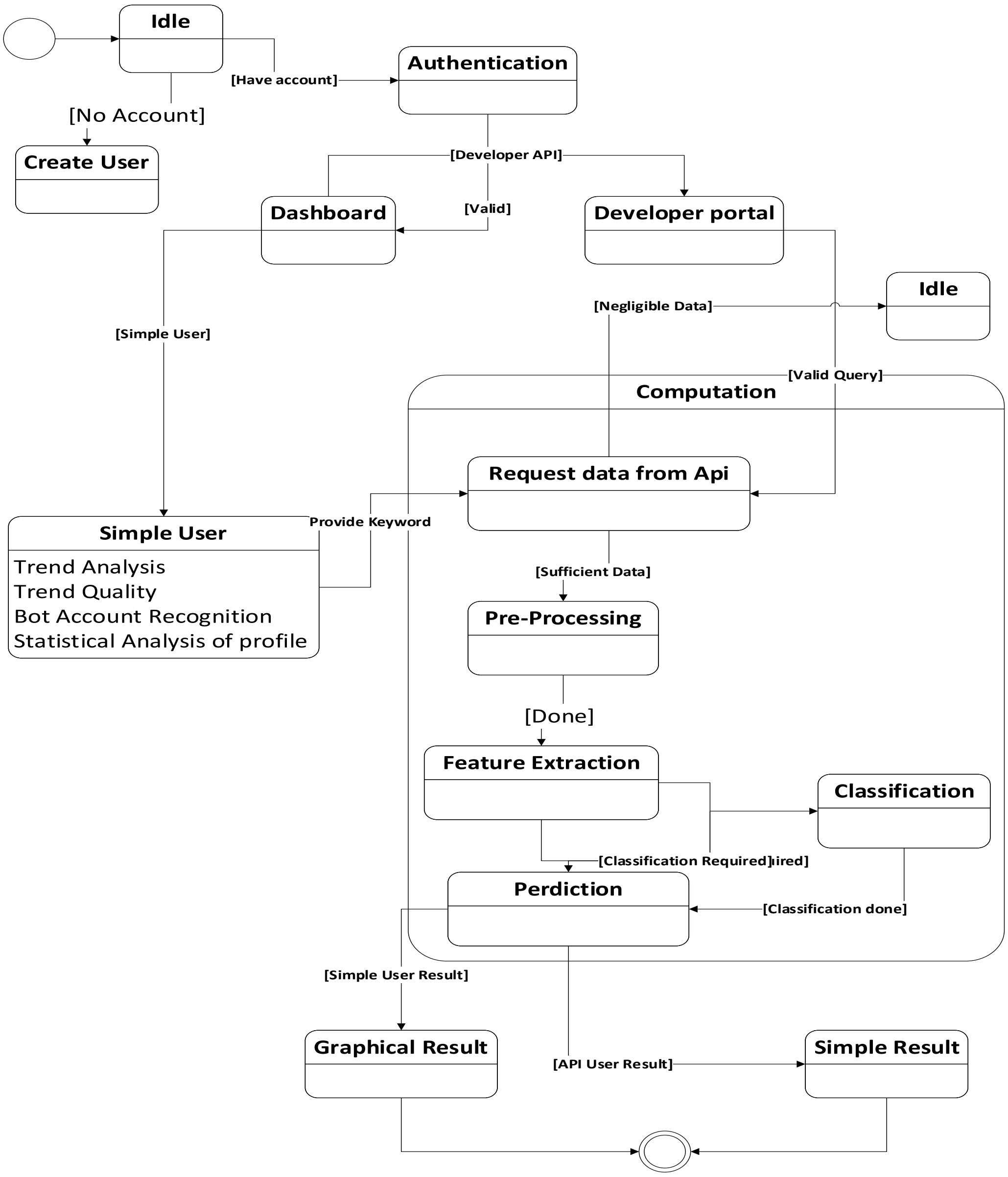


Figure 5: State Machine Diagram

10. Swim Lane Diagram

A swim lane diagram is a type of flowchart that delineates who does what in a process.

10.1. Swim Lane Diagram of User Interface

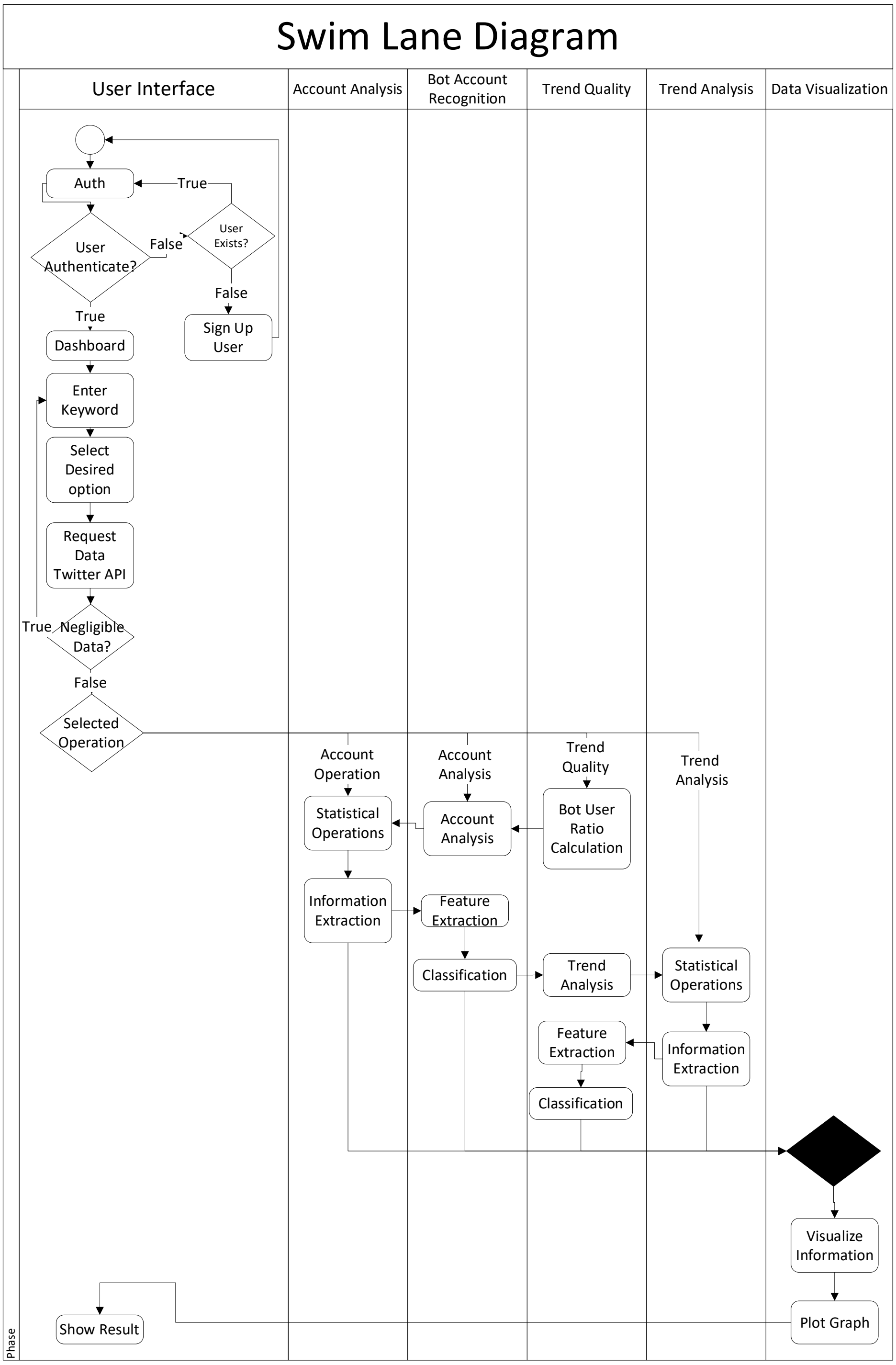


Figure 6: Swim Lane Diagram of UI Interface

10.2. Swim Lane Diagram of Developer API

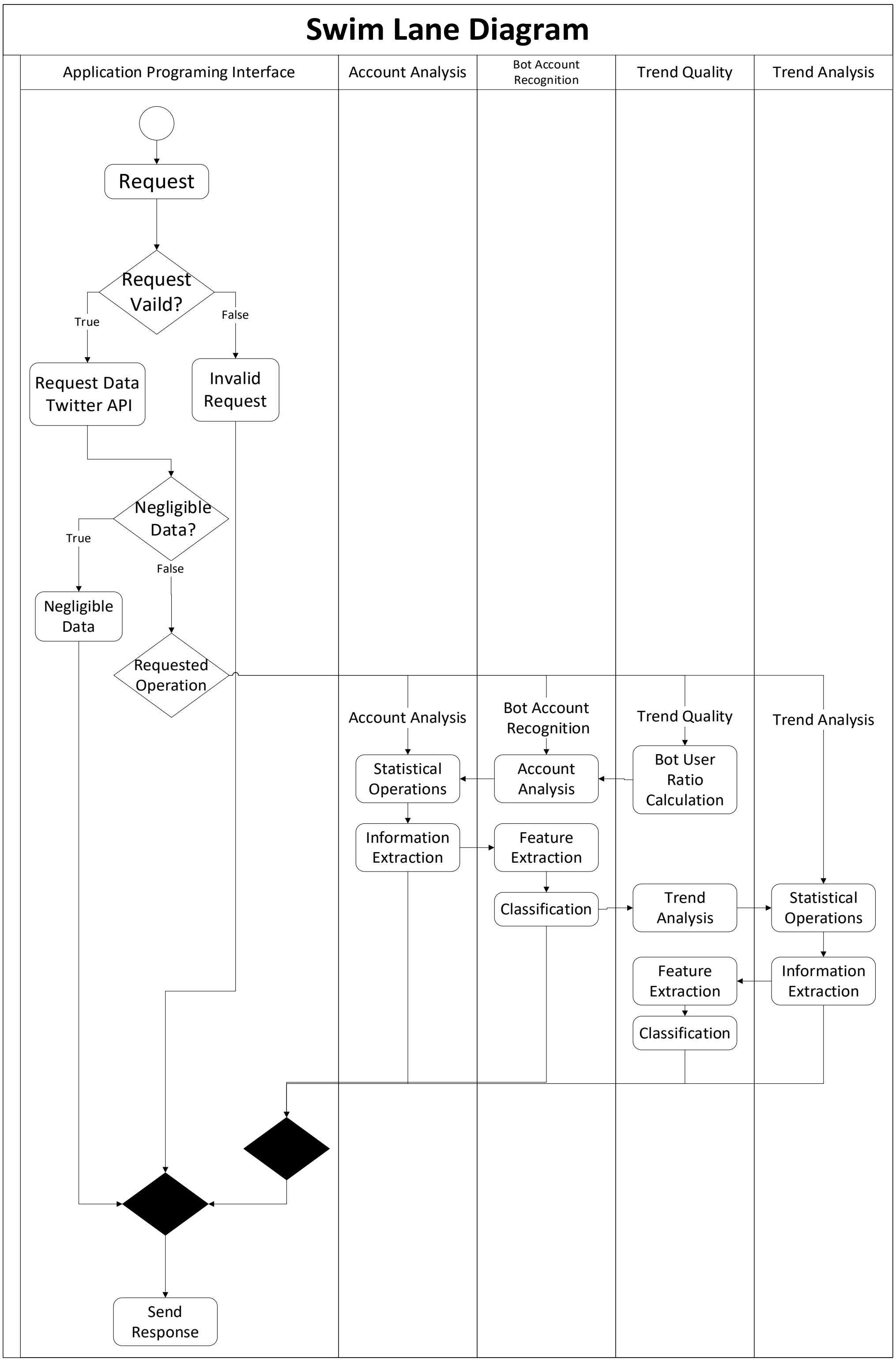


Figure 7: Swim Lane Diagram of Developer API Interface

11. System Sequence Diagram

A system sequence diagram is a picture that shows, for one particular scenario of a use case, the events that external actors generate, their order, and inter-system events.

11.1. Analyze Trend

User is an actor who will start Analyze Trend process. Then user provide keyword and system request data from Twitter API. After getting data system will perform operation on it and return result to user.

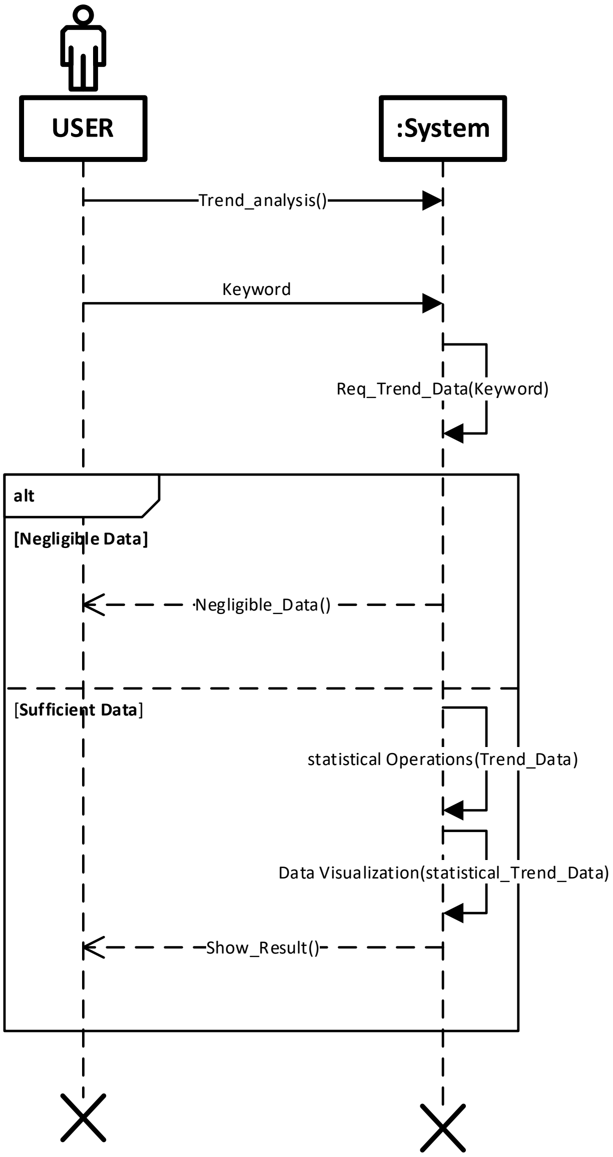


Figure 8: SSD of Analyze Trend

11.2. Extract Trend Quality

User is an actor who will start Extract Trend Quality process. Then user provide keyword and system request data from Twitter API. After getting data system will perform operation on it and return result to user.

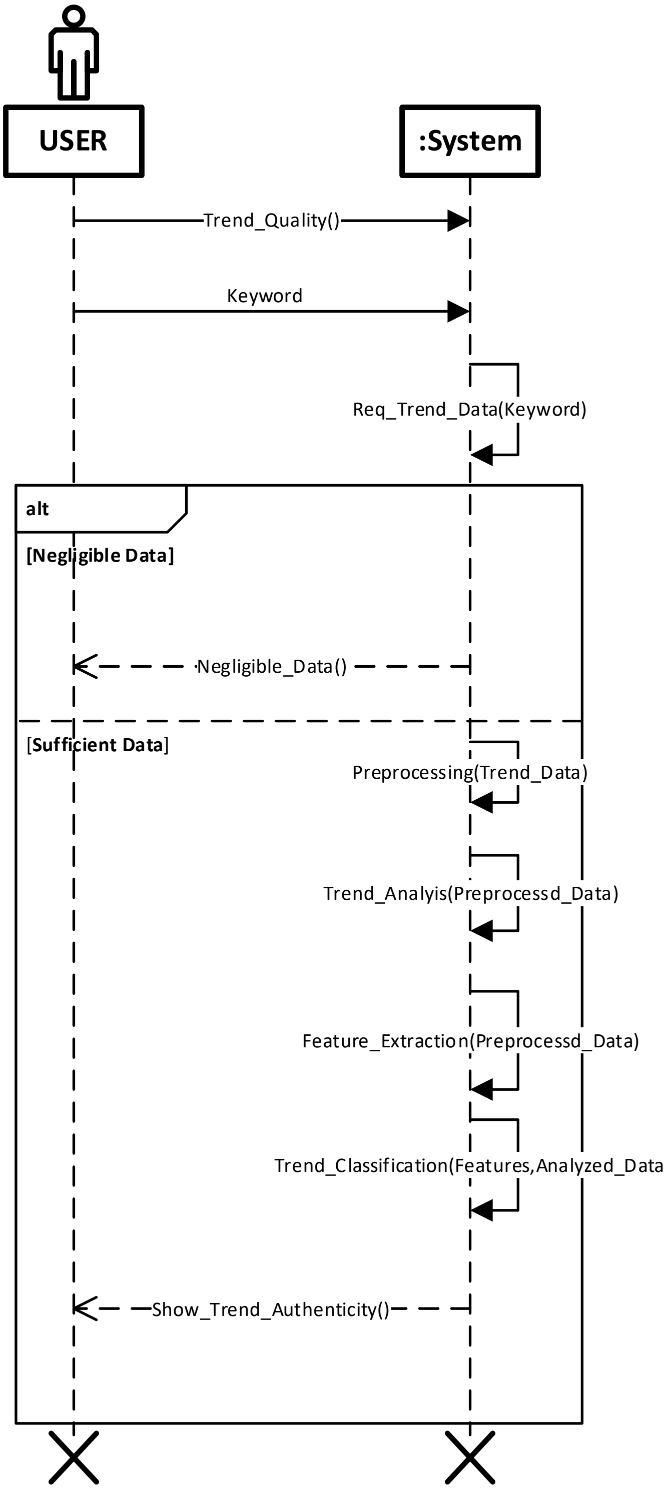

USER
:System
Trend_Quality()
Keyword
Req_Trend_Data(Keyword)
alt
[Negligible Data]
Negligible_Data()
[Sufficient Data]
Preprocessing(Trend_Data)
Trend_Analyis(Preprocessd_Data)
Feature_Extraction(Preprocessd_Data)
Trend_Classification(Features,Analyzed_Data
Show_Trend_Authenticity()

Figure 9: SSD of Extract Trend Quality

11.3. Recognize Bot Account

User is an actor who will start Recognize Bot Account process. Then user provide username and system request data from Twitter API. After getting data system will perform operation on it and return result to user.

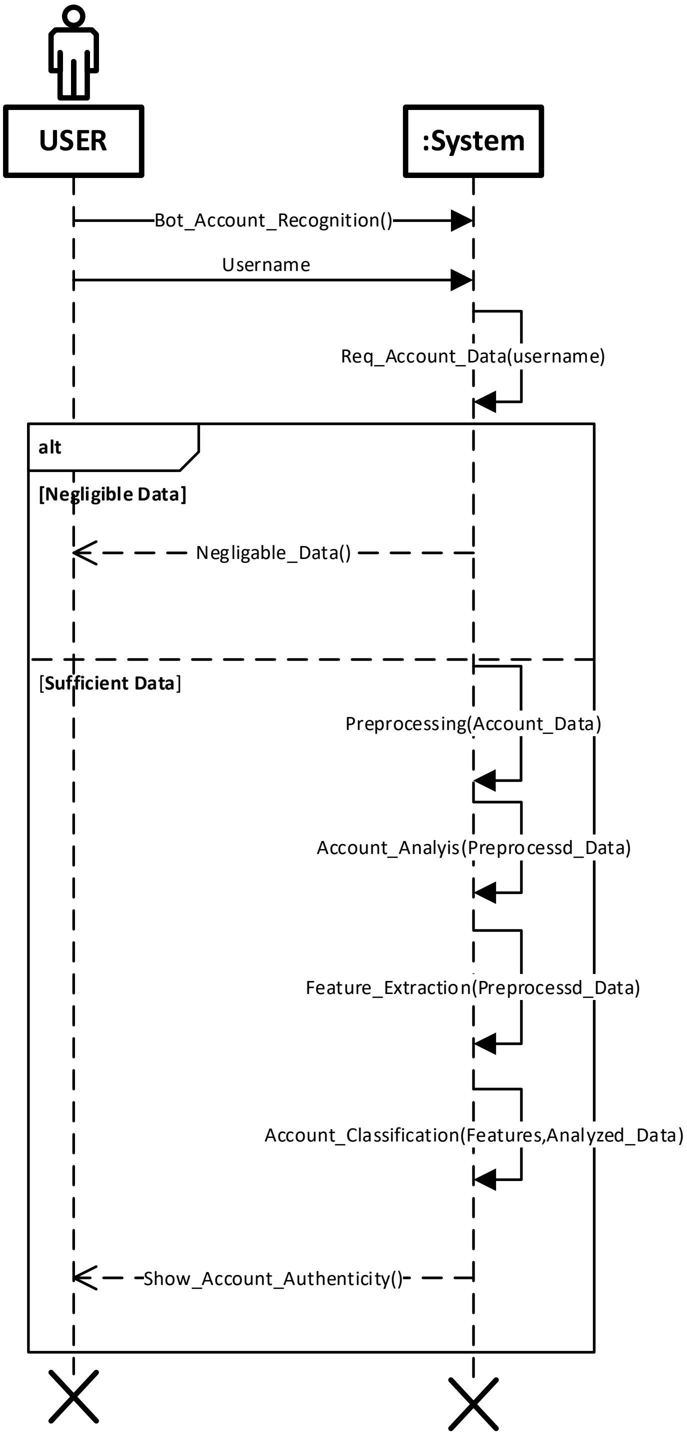

USER
:System
Bot_Account_Recognition()
Username
Req_Account_Data(username)
alt
[Negligible Data]
Negligable_Data()
[Sufficient Data]
Preprocessing(Account_Data)
Account_Analyis(Preprocessd_Data)
Feature_Extraction(Preprocessd_Data)
Account_Classification(Features,Analyzed_Data)
Show_Account_Authenticity()

Figure 10: SSD of Recognize Bot Account

11.4. Analyze Profile

User is an actor who will start Analyze Profile process. Then user enter username and system request data from Twitter API. After getting data system will perform operation on it and return result to user.

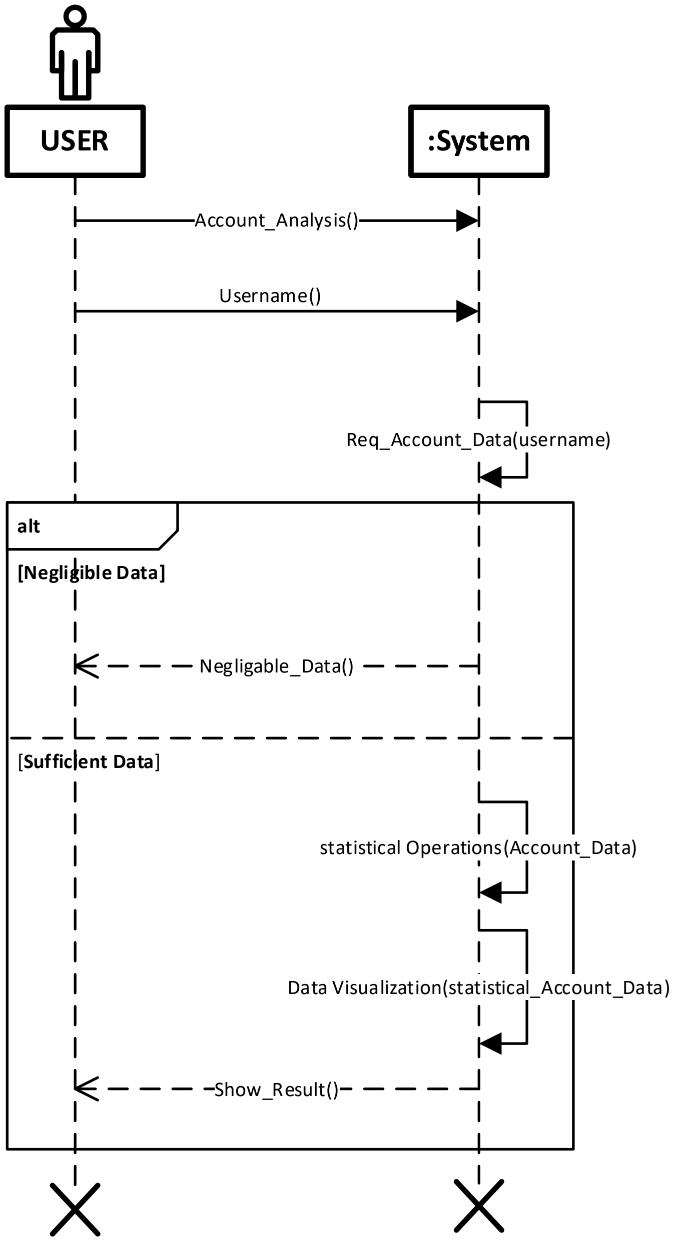

USER
:System
Account_Analysis()
Username()
Req_Account_Data(username)
alt
[Negligible Data]
Negligable_Data()
[Sufficient Data]
statistical Operations(Account_Data)
Data Visualization(statistical_Account_Data)
Show_Result()

Figure 11: SSD of Analyze Profile

11.5. Developer API

User is an actor who will start developer API process. Then user provide query and keyword and system request data from Twitter API. After getting data system will perform operation on it and return result to user.

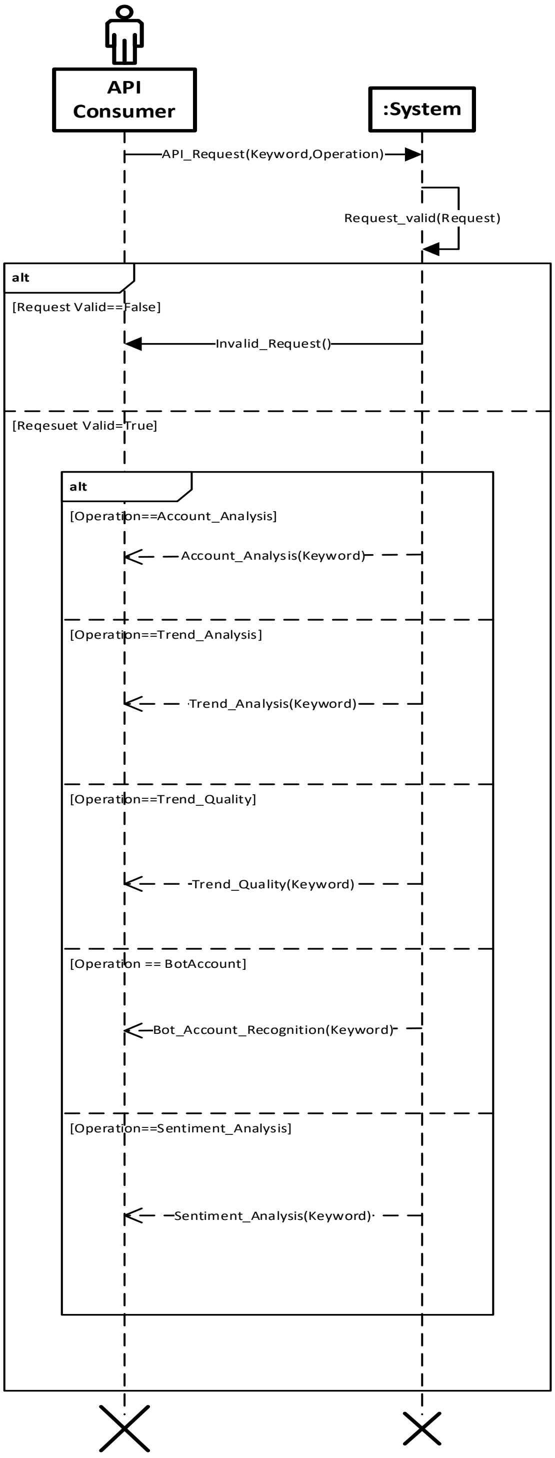

API Consumer
:System
API_Request(Keyword,Operation)
Request_valid(Request)
alt
[Request Valid==False]
Invalid_Request()
[Reqesuet Valid=True]
alt
[Operation==Account_Analysis]
Account_Analysis(Keyword)
[Operation==Trend_Analysis]
Trend_Analysis(Keyword)
[Operation==Trend_Quality]
Trend_Quality(Keyword)
[Operation == BotAccount]
Bot_Account_Recognition(Keyword)
[Operation==Sentiment_Analysis]
Sentiment_Analysis(Keyword)

Figure 12: SSD of Developer API

12. Architecture Diagram

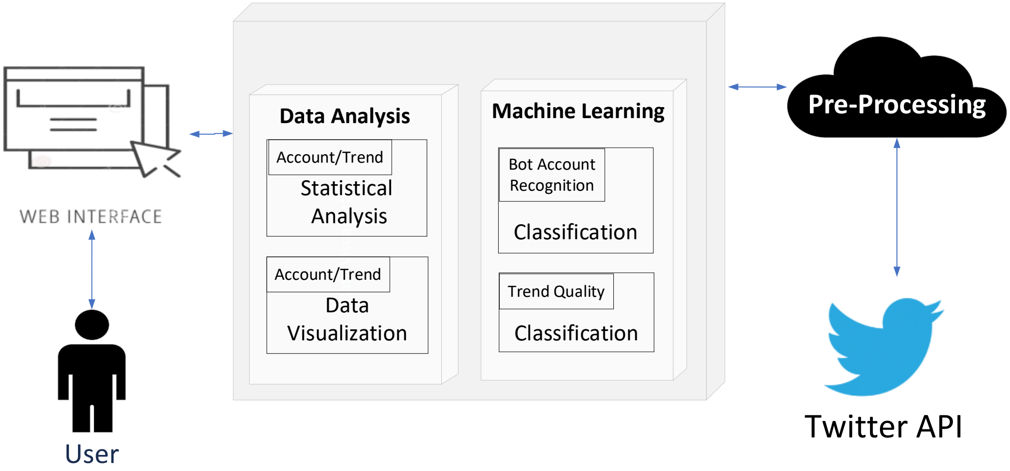


Figure 13: Architecture Diagram

13. Classification of Twitter Accounts

13.1. Data Dictionary

created_at: Date at which profile is created

default_profile: True is user havnet edited the file

default_profile_image : True if user never updated profile

description:

favorites_count

followers_count

friends_count

geo_enabled

id

lang

location

profile_background_image_url

profile_image_url

screen_name

statuses_count

verified

average_tweets_per_day

account_age_days

account_type : The label of human or bot account

bot : True is twitter account is bot

screen_name_len: length of username

bot_is_substr:

bot_in_des: bot written in description

age: the age of twitter account

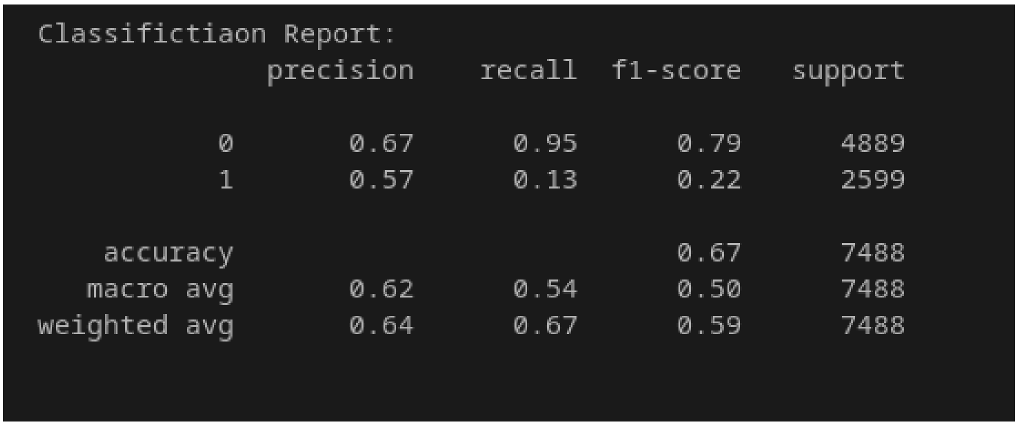
13.2. Target label

bot, Account_type

13.3. Logistic Regression

Training features: age , followers_count , friends_count , verified , favourites_count , statuses_count , default_profile , default_profile_image , screen_name_len , bot_is_substr , bot_in_des

Lable	**Coefficients**
age	0.00015701202307958\u200a36
followers_count	-8.190193801750009e-07
friends_count	1.8563807740196118e-06
verified	-7.194268150408971e-08
favourites_count	-3.9554404829103e-05
statuses_count	2.416448489328014e-06
default_profile	5.2953194742082925e-08
default_profile_image	5.2801715152072135e-09
screen_name_len	-8.974105401779116e-07
bot_is_substr	5.520716765550009e-09
bot_in_des	8.050443224551905e-09



```
Classifictiaon Report:
              precision    recall  f1-score   support

           0       0.67      0.95      0.79      4889
           1       0.57      0.13      0.22      2599

    accuracy                           0.67      7488
   macro avg       0.62      0.54      0.50      7488
weighted avg       0.64      0.67      0.59      7488
```

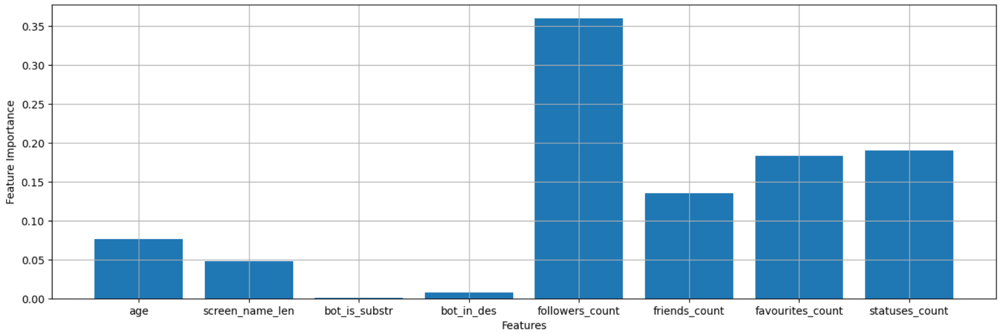
13.4. Decision Tree

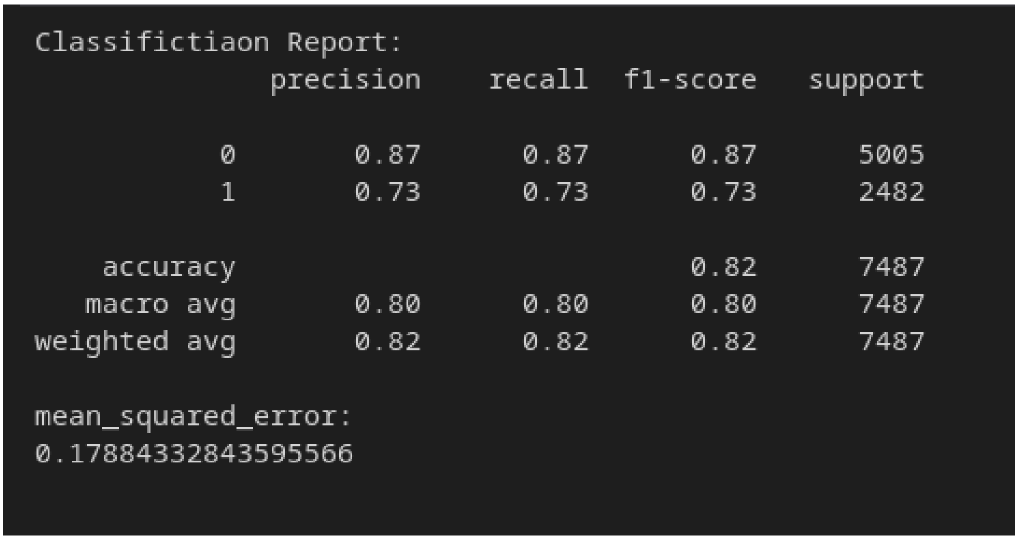
Training Features : age , screen_name_len , bot_is_substr , bot_in_des , followers_count , friends_count , favourites_count , statuses_count

Method: entropy

Note: Lower is more importunate

Label	**Entropy**
age	0.07602383217756341
screen_name_len	0.049772253705837766
bot_is_substr	0.0007148862974873882
bot_in_des	0.005718861587751301
followers_count	0.34747974775527374
friends_count	0.14265401337883268
favourites_count	0.18218861966785238
statuses_count	0.1954477854294014

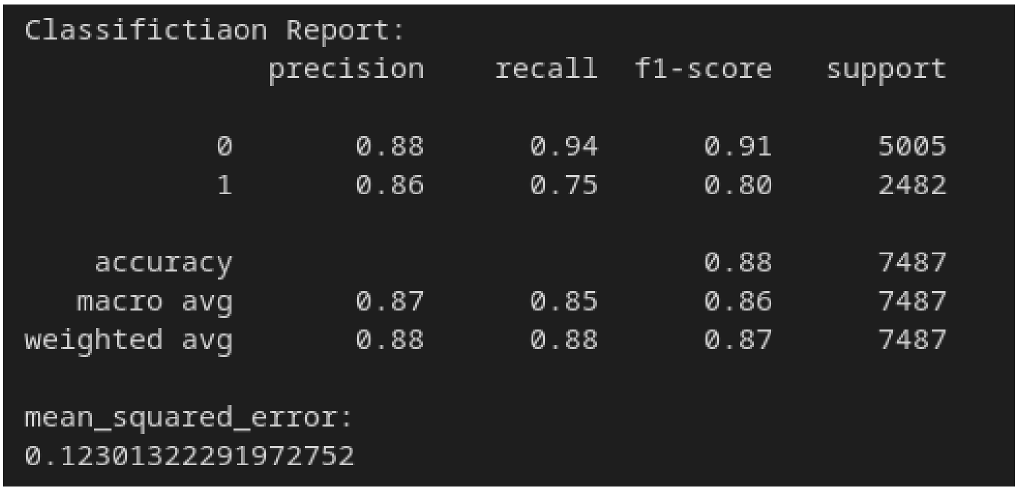




```
Classifictiaon Report:
              precision    recall  f1-score   support

           0       0.87      0.87      0.87      5005
           1       0.73      0.73      0.73      2482

    accuracy                           0.82      7487
   macro avg       0.80      0.80      0.80      7487
weighted avg       0.82      0.82      0.82      7487

mean_squared_error:
0.17884332843595566
```

13.5. Random forest

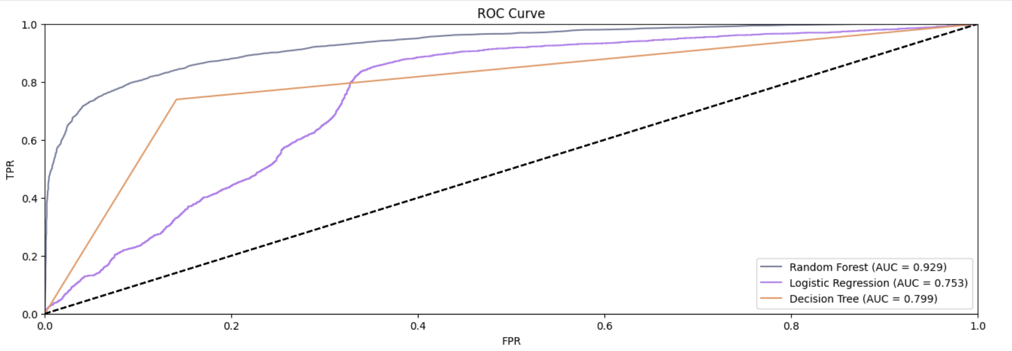
Training features: age , followers_count , friends_count , verified , favourites_count , statuses_count , default_profile , default_profile_image , screen_name_len , bot_is_substr , bot_in_des



```
Classifictiaon Report:
              precision    recall  f1-score   support

           0       0.88      0.94      0.91      5005
           1       0.86      0.75      0.80      2482

    accuracy                           0.88      7487
   macro avg       0.87      0.85      0.86      7487
weighted avg       0.88      0.88      0.87      7487

mean_squared_error:
0.12301322291972752
```

13.6. Conclusion

The random forest is most suitable classification model



features=[account_age, followers_count, friends_count, verified, favourites_count, statuses_count, default_profile, default_profile_image, screen_name_len, bot_is_substr, bot_in_des]

14. Unit Test Plan

14.1. Signup

Module:Signup				
Unit: SignupWithusernameAndEmailAndPassword(username, email, password)				
Test Case ID	Input Values	Expected Output	Actual Output	Pass/Fail Status

A001	Username= “”,
	email= “”,
	password = “”
		Alert will appear that “Please Enter All Fields”. Does not register	Alert shown. Not Signup.	Passed

A002	Username= “a”,
	email= “”,
	password = “”
		Alert will appear that “Please Enter All Fields”. Does not register	Alert shown. Not Signup.	Passed

A003	Username= "", email= "a@gmail.com", password = ""	Alert will appear that "Please Enter All Fields". Does not register	Alert shown. Not Signup.	Passed
A004	Username= "", email= "a@gmail.com", password = "1234"	Alert will appear that "Please Enter All Fields". Does not register	Alert shown. Not signup.	Passed
A005	Username= "ali", email= "a@gmail.com", password = "1234"	User will Successfully Signup.	Alert shown. Sign up Successful.	Passed

14.2. SignIn

Module: SignIn				
Unit: SignINWithEmailAndPassword(username, email, password)				
Test Case ID	Input Values	Expected Output	Actual Output	Pass/Fail Status
A006	email= "", password = ""	An Alert of error response. Does not login.	Alert shown. Not Sign in	Passed
A007	email= "a.b@gmail.com",	Alert will appear	Alert shown. Not Sign in	Passed

	password = “1234”	that “This Email is not registered.”		
A008	email= “a@gmail.com”, password = “1234”	User will Successfully login	Done Sign in	Passed
A009	email= “a@gmail.com”, password = “12345”	Alert will appear that “Password not matched.”	Alert shown. Not Sign in	Passed

14.3. Bot Account Recognition

Module: Bot Account Recognition				
Unit: TestBotAccount(username)				
Test Case ID	Input Values	Expected Output	Actual Output	Pass/Fail Status
A010	username= “”,	An Alert of error response. “Enter Username”	Alert shown. Enter Username	Passed
A011	username= “Abcde”,	An Alert of error response. “Username does not exist on twitter”	Alert shown. Enter a valid Username.	Passed

A012	username= “Ali”,	Result will Successfully Shown to user.	Result Shown to user.	Passed
A013	username= “Ahmad”,	An Alert of error response. “User Profile is private.”	Alert shown. Enter a valid Username.	Passed

14.4. Trend Analysis/ Trend Quality

Module: Trend Analysis/Trend Quality				
Unit: TestTrendAnlysis_Quality(keyword)				
Test Case ID	Input Values	Expected Output	Actual Output	Pass/Fail Status
A014	keyword= “”,	An Alert of error response. “Enter keyword”	Alert shown. Enter Keyword	Passed
A015	keyword= “adfji”,	An Alert of error response. “Not Enough data for Keyword”	Alert shown. Not enough data enter another Keyword	Passed
A016	keyword= “SuperemeCourt”,	Result will Successfully Shown to user.	Result Successfully Shown to user.	Passed