Dashboard for analyzing the content in social media (Twitter)

A project report submitted in partial fulfillment of the
requirement for the award of the degree of

BACHELOR IN COMPUTER SCIENCE

BY

Ahmed Sahloli (201700159)

Naif Albarqi(201711902)

Project Supervisor

Dr. Essa Alhazmi

COLLEGE OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY

JAZAN UNIVERSITY, JAZAN (KSA)

2020-2021 (Second Semester)

APPROVAL FOR BINDING OF GRADUATION PROJECT REPORT
GRADUATION PROJECT (495 COMP- 3)

GROUP (CS) : 2776

SESSION 2020-2021 (Spring Semester)

To be filled in by student

Student name(s) : **Ahmed Sahloli | Naif Albarqi**

University ID(s). : **201700159 | 201711902**

Project title : **Dashboard for analyzing the content in social media (Twitter)**

To be filled in by supervisor*

This is to certify that the project submitted and presented by the above mentioned student(s) is

□ **COMPLETE & ACCEPTED** □ **INCOMPLETE** □ **NOT ACCEPTED**

and the draft of graduation project report has been corrected from all content flaws, typing errors and language mistakes.

Supervisor name : Dr. Essa Alhazmi

Signature :

Date :

***NOTE**: The student may proceed with Ring binding ONLY after this section has been completed. The student shall submit three **copies** of Ring bound draft report for Graduation Project to the panel of examiners on the day of final presentation.

For Departmental Office use only

Date received :

Signatures availability :

Acknowledgement

All this became possible thanks to God Almighty. we Very indebted to a some people who helped guide us and Ongoing support throughout the project implementation period, which is invaluable Without their precious help and support for us, we are grateful to our supervisor Graduation Project Dr. Issa Al-Hazmi, we appreciate the time you spent on making the great effort for us, and we assure you that we will meet your good expectations and that you will not feel later that your efforts have been lost in the wrong place. Thank you very much for your support.

Thanks, gratitude, and deep and sincere gratitude from us to our parents and everyone.
Who helped us one day support us moving forward with our studies and Constant drive over the past years. and sincerely we thank Dr.Ismail Alaqta. Words and phrases do not meet you right. Thank you very much for all you have given us in completing our academic studies Professional life.

Thank You All

Ahmed Sahloli Naif Albarqi

Abstract

The content in social media have an impact on our society in many aspects including the spreading of abusive content ,fake news , misinformation.

In this project, we seek to build a dashboard that enables the user to track and analyze what is in trends at the current time, as well as analyze contents.

The dashboard will provide the user with detailed information. It will also provide high-level interactive charts and geographical maps.

Keywords : Social media (Twitter), Dashboard, Natural language processing, Python.

Table of content

Chapter 1 | INTRODUCTION

Chapter 2 | INFORMATION GATHERING

Chapter 3 | SYSTEM ANALYSIS

Chapter 4 | SYSTEM DESIGN

List of Figure

List of Table

CHAPTER 1

Introduction

1.1 Introduction:

Twitter is one of the most popular social media platforms in Saudi Arabia, millions of tweets are published daily, and now the Twitter platform is used to express feelings, raise social issues, and many things that have an effective role in society.

In this project, we seek to build a dashboard that enables the user to track and analyze what is in trends at the current time, as well as analyze contents.
The dashboard will provide the user with detailed information. It will also provide high-level interactive charts and geographical maps.

We will also create a model using deep learning where the process of tracking offensive content is carried out.

The dashboard is equipped to work with artificial intelligence / deep learning models,
AI/DL models helps in tracking the spreading of abusive content ,fake news , misinformation.

1.2 Problem Definition :

- The content in social media have an impact on our society in many aspects including the spreading of abusive content ,fake news , misinformation.
- The need of monitoring and tracking of these contents is very a priority for many sectors.
- Providing a dashboard for analyzing the contents and flexible to add features such as pre-trained machine learning models is our objective.
- Analyzing the contents and understand their spreading are also very challenging and needs effective tools to monitor them.

1.3 Solution :

The project contains the following activities:

- User interface is provided.
- The user can know the trend at any time.
- Users can search for and locate the user, their contents.
- User can search for trend contents.

1.4 Goals and Objectives :

- Building an interactive dashboard , where the user can search for specific content, such as (hashtag, user account, content) and review results about it.
- Graphical representation of the analyzed data.

1.5 Project Scope :

- Artificial Intelligence, Natural Language Processing, Data Science, Software Engineering.

- Applying pre-trained model to analyze the data on the Twitter platform**.**

1.6 Hardware and Software Tools :

- **Required Hardware:**
- 4 GB RAM minimum, 8 GB RAM recommended
- 20 GB hard disk space + at least 1 GB for caches
- 1024×768 minimum screen resolution
- **Software :**
- Python 2.7, or Python 3.5 or newer
- IDEs and Code Editors for Python
- Web Browser

1.7 Work Breakdown Structure & Gantt Chart :

1.7.1 Work Breakdown Structure

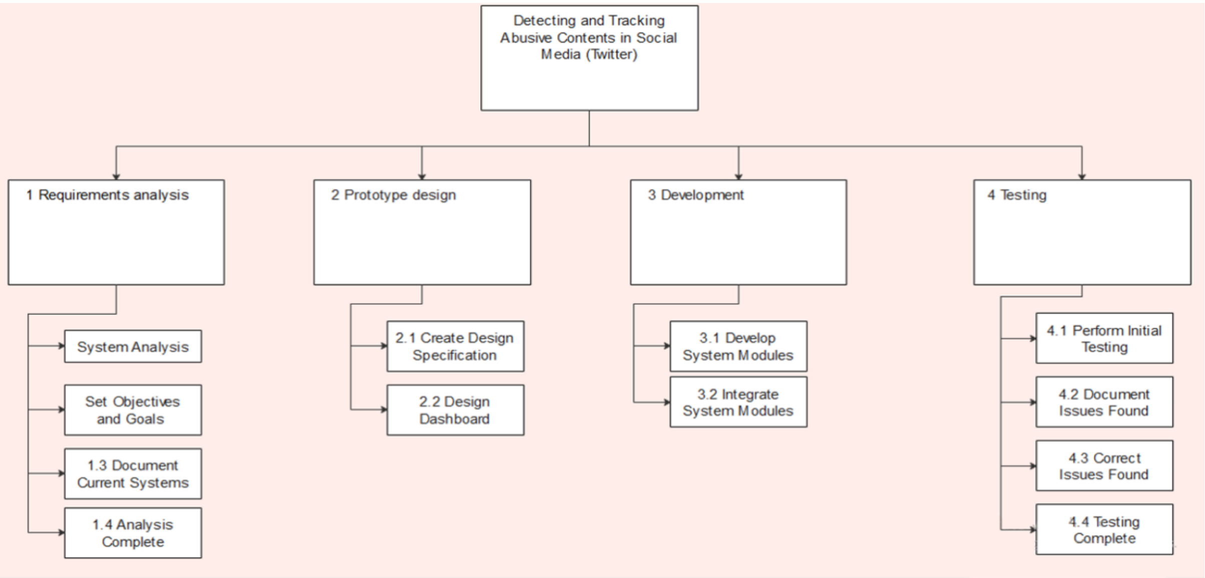
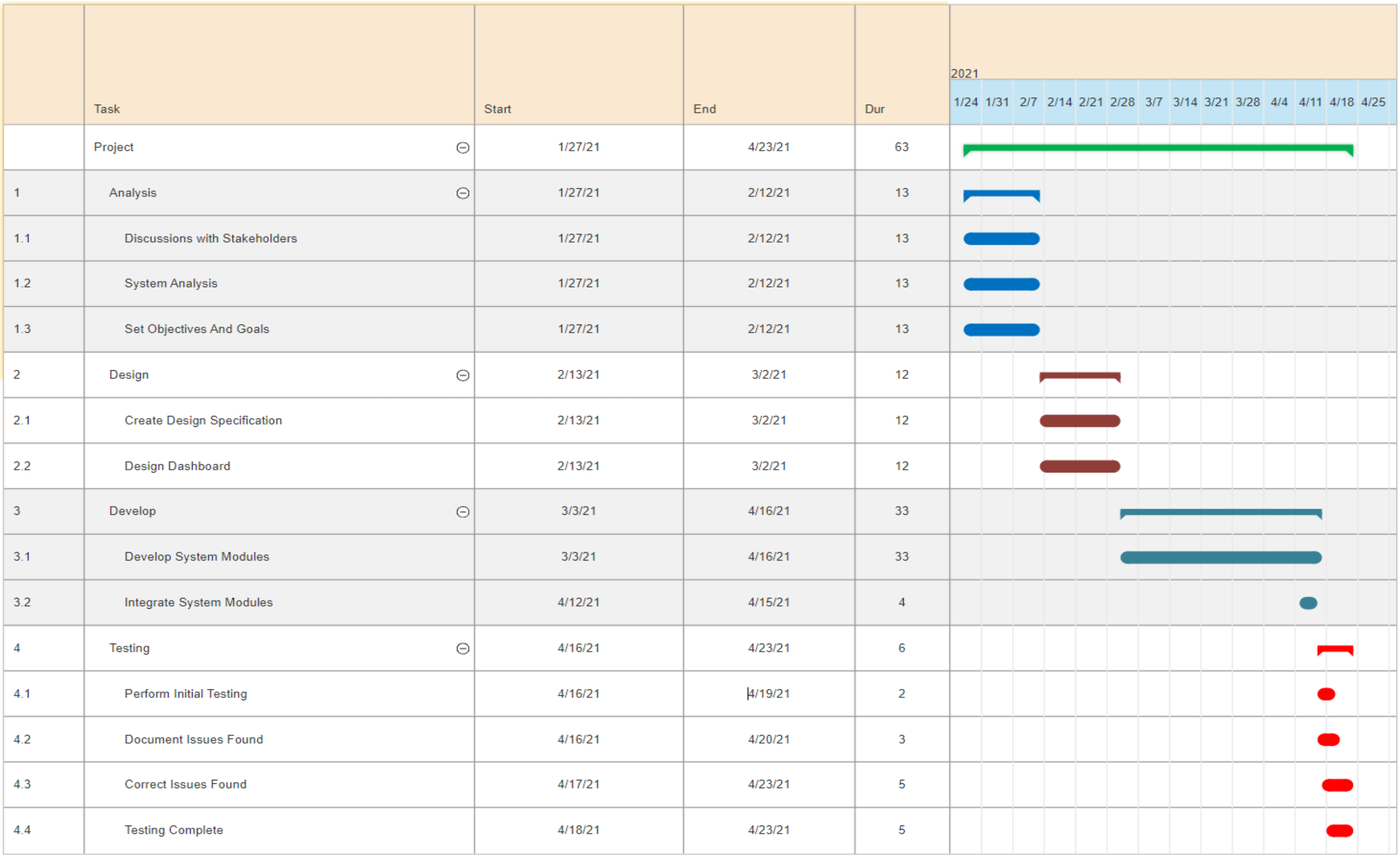

Detecting and Tracking Abusive Contents in Social Media (Twitter)
1 Requirements analysis
System Analysis
Set Objectives and Goals
1.3 Document Current Systems
1.4 Analysis Complete
2 Prototype design
2.1 Create Design Specification
2.2 Design Dashboard
3 Development
3.1 Develop System Modules
3.2 Integrate System Modules
4 Testing
4.1 Perform Initial Testing
4.2 Document Issues Found
4.3 Correct Issues Found
4.4 Testing Complete

Figure 1.7.1: Work Breakdown Structure

1.7.2 Gantt Charts



	Task	Start	End	Dur
	Project ⊖	1/27/21	4/23/21	63
1	Analysis ⊖	1/27/21	2/12/21	13
1.1	Discussions with Stakeholders	1/27/21	2/12/21	13
1.2	System Analysis	1/27/21	2/12/21	13
1.3	Set Objectives And Goals	1/27/21	2/12/21	13
2	Design ⊖	2/13/21	3/2/21	12
2.1	Create Design Specification	2/13/21	3/2/21	12
2.2	Design Dashboard	2/13/21	3/2/21	12
3	Develop ⊖	3/3/21	4/16/21	33
3.1	Develop System Modules	3/3/21	4/16/21	33
3.2	Integrate System Modules	4/12/21	4/15/21	4
4	Testing ⊖	4/16/21	4/23/21	6
4.1	Perform Initial Testing	4/16/21	4/19/21	2
4.2	Document Issues Found	4/16/21	4/20/21	3
4.3	Correct Issues Found	4/17/21	4/23/21	5
4.4	Testing Complete	4/18/21	4/23/21	5

Figure 1.7.2 Gantt Charts

1.8 Delegating Tasks

1.8.1 Tasks of team members:

Team Members	Ahmed Sahloli	Naif Albarqi
The Mission	System Developer	Systems Analyst
Tasks completed	- Define web application requirements. - Information Gathering. - Design System. - Development and implementation. - Cleaning and processing and analyzing data. - Data Visualization. - System Testing. - System Demonstration.	- use case Diagram. - Detailed use case Description. - Sequence Diagram. - Conceptual Diagram. - System Architecture. - Data Flow Diagram.

Table 1.8: Tasks of team members

1.8.2 Tasks completion rate:

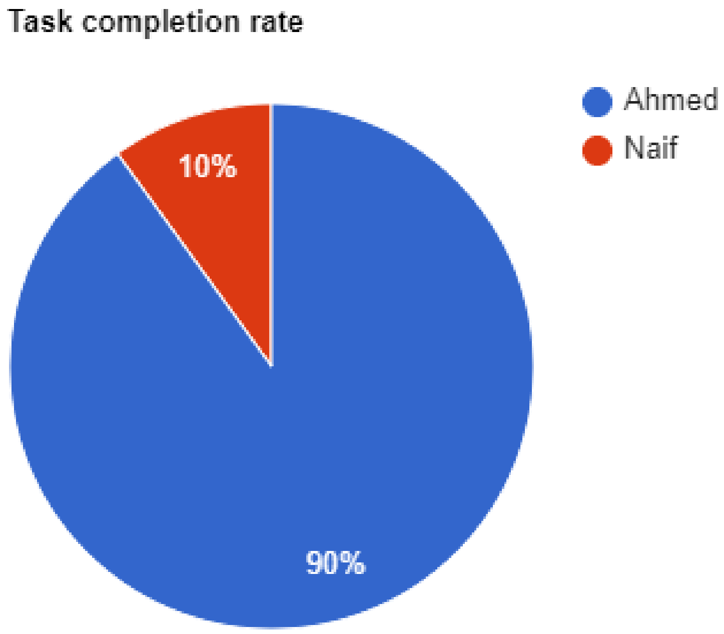


Figure 1.8.2: Task's completion rate

CHAPTER 2

Information Gathering

2.1 Information Gathering Techniques :

In our project, we have relied on several techniques, the most important of which are:

- **Brainstorming**

  The concept of brainstorming includes defining a simple or a complex problem for which solution are generated different ideas that are later accepted or rejected. The participants in the process of brainstorming present and reveal new ideas as potential solutions for a specific problem. In this process, brainstorming helps the stimulation of the human mind towards creative problem-solving.[1]

- **Searching for similar Dashboards**

  We searched for similar control panels to find out what services they offer, as well as what are the main pros and cons.

  Through this chapter, we will review most prominent current Dashboards.

2.1.1 Literature Review :

Recently, researchers have shown a growing interest in creating dashboards that contribute to content analysis through social media, but most of these panels did not focus on Middle Eastern content or rather did not focus on providing sufficient support for the Arabic language, in this chapter we will show famous dashboards and we will list their pros and cons.

2.1.2 Related Application :

- **Socialbakers**

  Socialbakers is a trusted social media marketing partner to thousands of enterprise brands and SMBs, including over 100 companies on the list of Fortune Global 500. Leveraging the largest social media data-set in the industry and machine learning, Socialbakers' marketing suite of solutions helps brands ensure their investment in social media is delivering measurable business outcomes.[2]

- **Datapine**

  datapine was founded in 2012 with the idea to facilitate the complex process of data analytics through technical innovation. After many years of extensive development to perfect our application, we are proud to count thousands of users in more than 25 countries around the globe. In the meantime we have gathered an outstanding team of data scientists, developers, technological leaders and academics who all work for the same goal: to help people learn from their data. We empower businesses of all sizes, technical and non-technical users alike, to explore, visualize and communicate their information.[3]

- **Dashboardom**

  A collection of dashboards and apps, made by Elias Dabbas. For data exploration, learning, fun, prototyping, and sharing. Mostly made with Plotly's Dash (Python).[4]

- **Lucidya**

  Lucidya is an award-winning, AI-powered social listening tool geared towards Arabic brands.[5]

2.1.3 Comparison table of the related applications

Name of App / Features	Socialbakers	Datapine	Dashboardom	Lucidya	Our dashboard
Arabic Language support	✗	✗	✓	✓	✓
Free	✗	✗	✓	✗	✓
Ease of use	Easy	Easy	Very Easy	Easy	Very Easy

Table 2.1.3 : Comparison related applications

2.2 Conclusion and Outcomes :

The most prominent thing that we will present in this project is to create a dashboard that supports the Arabic language. We will also create many charts that contribute to a more understanding of the content. The most important thing about our panel is its ease of use and detailed analysis, as the ease of use distinguishes us from the rest of the dashboards.

CHAPTER 3

System Analysis

3.1 Development Methodology :

In our project we used Incremental Model

What is Incremental Model: is a process of software development where requirements are broken down into multiple standalone modules of software development cycle.[6]

Incremental development: is done in steps from analysis design, implementation, testing/verification, maintenance.[6]

Characteristics of an Incremental module includes:

- System development is broken down into multiple standalone modules.
- Partial systems are successively built to produce a final total system.
- Highest priority requirement is tackled first.[6]

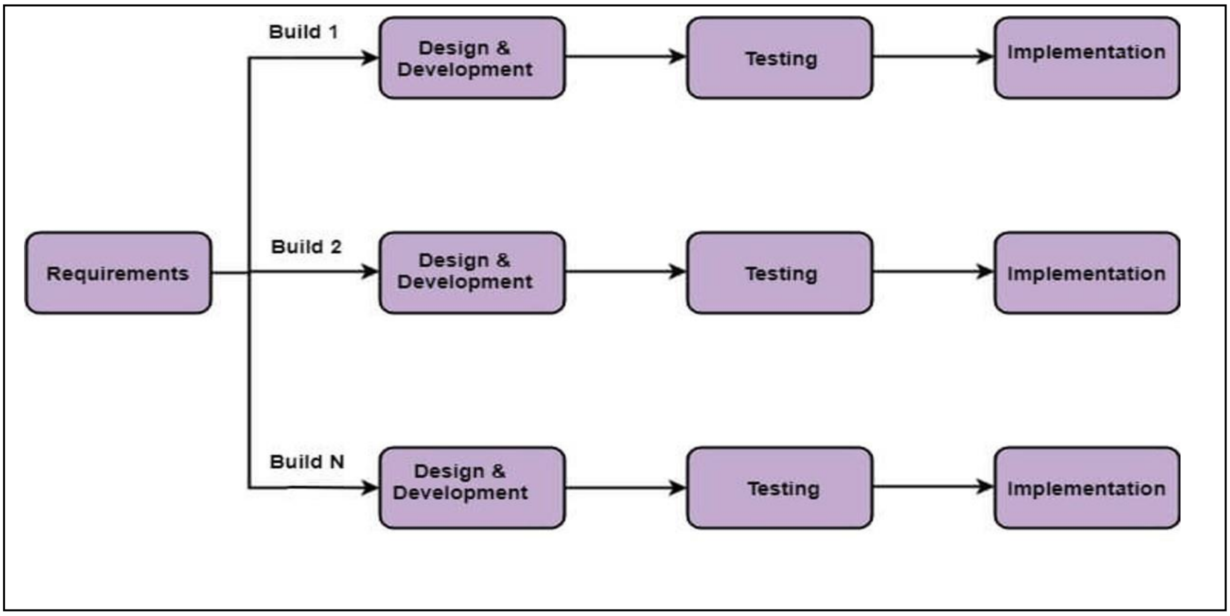


Figure 3.1 Incremental Model

3.2 User Characteristics :

- **Search for a user :**

The user can search for a user and see the profile of user .

- **Determine the location of the user :**

The user can know the location of the user through the profile.

- **View the top 50 trends today :**

It will show the user the top five trends today in form of a table.

- **View the contents of the trend :**

The user can search for the trend and see how many tweets and how rank this trend.

3.3 User and System Requirements :

1- User requirements: describe what the user does with the system, such as what activities that users must be able to perform.

2- System requirements: are the building blocks developers use to build the system. These are the traditional "shall" statements that describe what the system "shall do".

3.3.1 Functional Requirements :

Function Requirement of the system	
Requirement ID	Requirement Description
1-D	The user should ability to search
2-D	The user should ability to interactive
3-D	The user should ability to view
4-D	The user should ability to Export
5-D	The user should ability to Share

Table 3.3.1 Function Requirement of the system

3.3.2 Non-Functional Requirements :

1. **Performance Requirements**

Performance Requirement of the system	
Requirement ID	Requirement Description
1-P	The system must respond in the least amount of time possible.
2-P	The system must be able to deal with errors encountered during execution.
3-P	The system must be integrated with other components.

Table 3.3.2: Performance Requirement of the system

2. **Safety& Security and Privacy Requirements:**

We did not use a database in this system, so there are no Safety requirements to take backup periodically, also there are no requirements related to security and privacy because it does not have a database to save user data or authorized users.

3.4 System analysis Models :

is a description of how data should be used to meet the requirements given by the end user. Data modeling helps to understand the information requirements. One of the important functions of data modeling is that it helps to understand the information requirements.

3.4.1 Use Case Diagram :

Use case diagram It is the primary form of system / program requirements for a new, undeveloped program. Use cases define the expected behavior (what), not the exact way to achieve it (how). One-time specific use cases can be referenced to both textual and visual representation (i.e. use case diagram). The basic concept of use case modeling is that it helps us design a system from an end-user perspective. It is an effective technique for communicating system behavior according to user conditions by externally identifying all visual system behavior.[7]

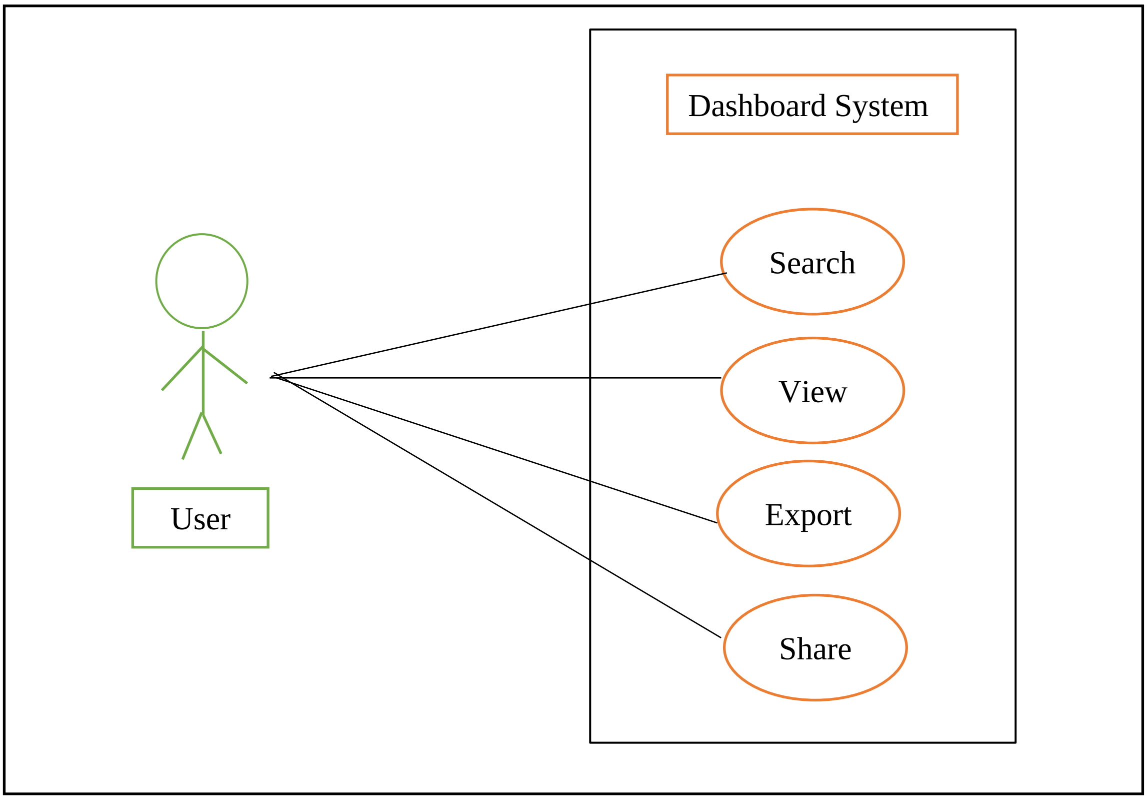


Figure 3.4.1 Use case diagram

3.4.2 Detailed use case Description:

- **Search**
  User has ability to search for data like tweets ,trends, whatever on twitter.

- **View**
  User has ability to visualize the contents.

- **Export**
  User can Export the results in a report form.

- **Share**
  User can share the results out the system by a URL.

3.4.3 Sequence Diagram :

The sequence diagram simply depicts the interaction between the objects in a sequential order, that is, the order in which these interactions take place. We can also use the terms event diagrams or event scenarios to refer to a sequence diagram. Sequence diagrams describe how and in what order of objects in the system function. These diagrams are used extensively by business and software developers to document and understand the requirements of new and existing systems.[8]

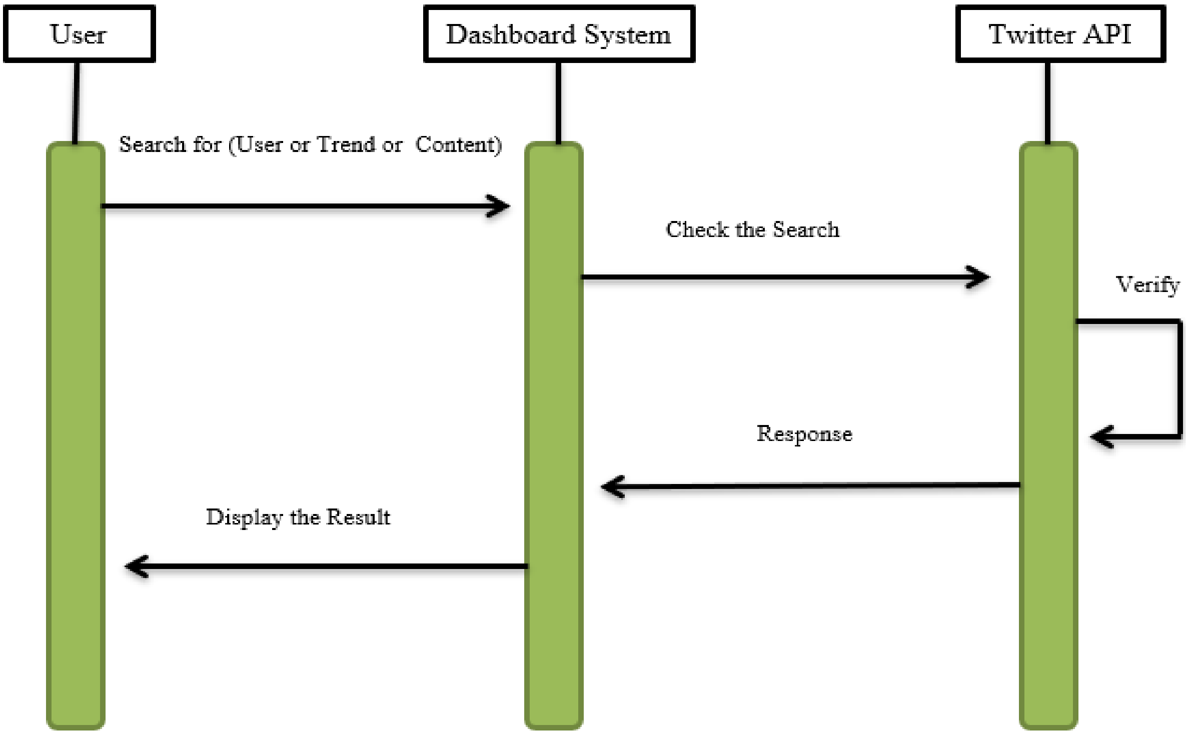


Figure 3.4.3: Sequence Diagram

3.4.4 Conceptual Diagram:

Conceptual diagrams are concise and visually-stimulating illustrations that use symbols or drawings to depict the important features, processes and management challenges in a particular environment, such as coastal waterways. This is accomplished using the most current knowledge or understanding of that particular environment and is presented in a way that is easy to understand.[9]

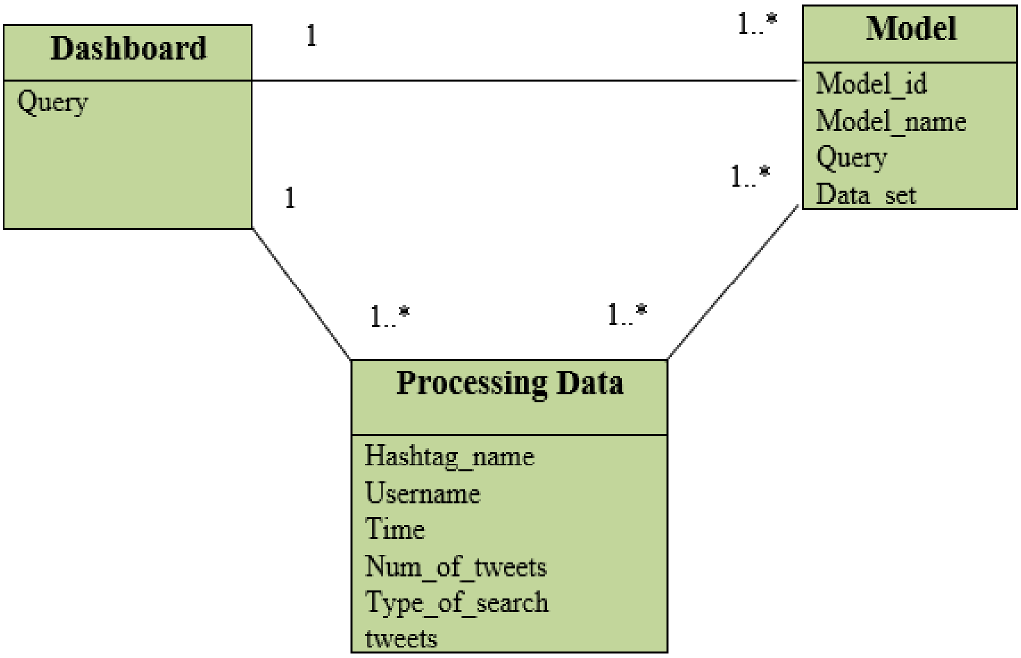


Figure 3.4.4: Conceptual Diagram

CHAPTER 4

System Design

4.1 System Architecture :

A system architecture is the conceptual model that defines the structure, behavior, and more views of a system. An architecture description is a formal description and representation of a system, organized in a way that supports reasoning about the structures and behaviors of the system.[10]

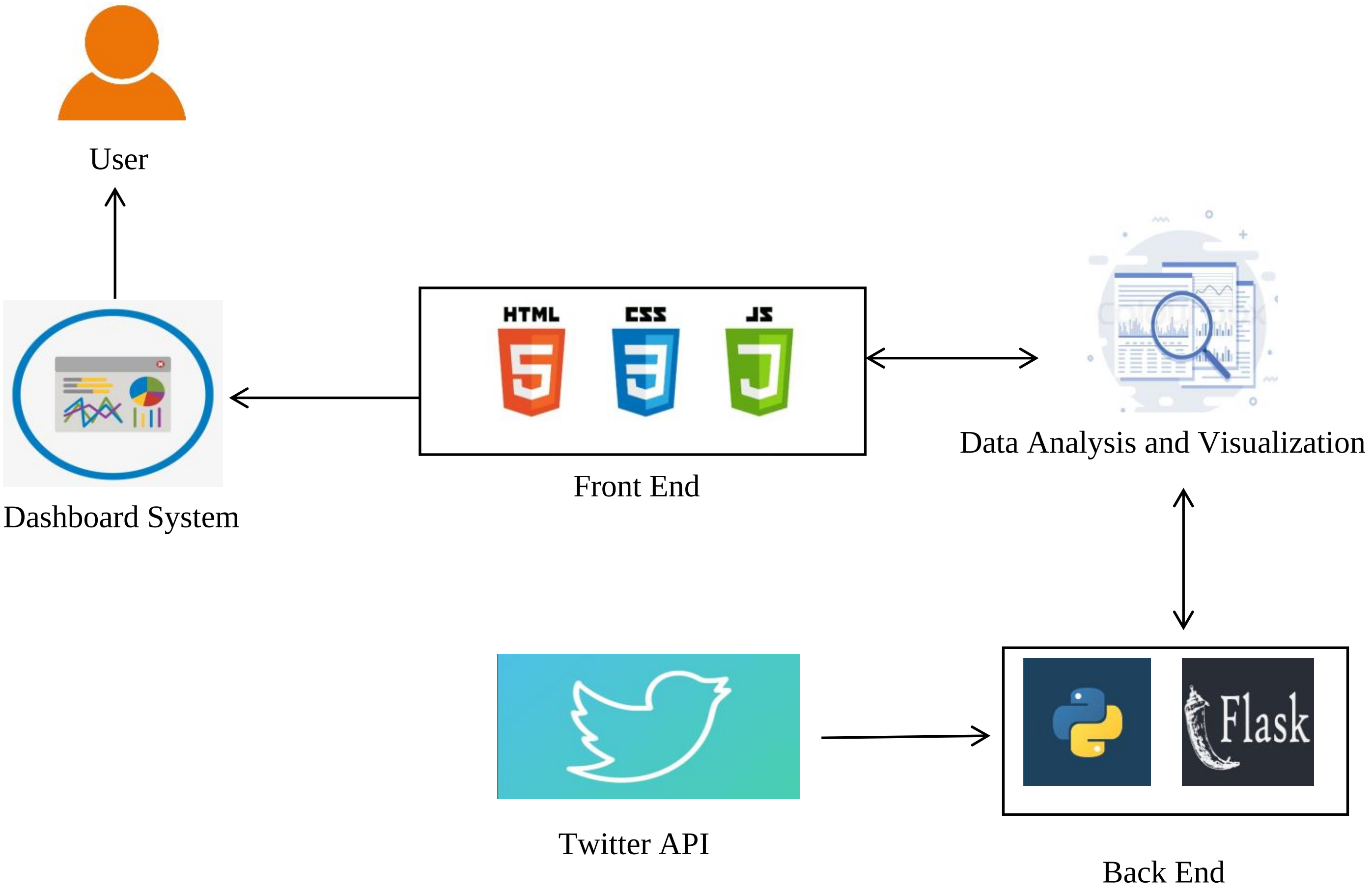


Figure 4.1 : System Architecture

4.2 Data Flow Diagram :

Data flow diagrams are used to graphically represent the flow of data in a business information system. DFD describes the processes that are involved in a system to transfer data from the input to the file storage and reports generation. Data flow diagrams can be divided into logical and physical. The logical data flow diagram describes flow of data through a system to perform certain functionality of a business. The physical data flow diagram describes the implementation of the logical data flow.

DFD graphically representing the functions, or processes, which capture, manipulate, store, and distribute data between a system and its environment and between components of a system. The visual representation makes it a good communication tool between User and System designer. Structure of DFD allows starting from a broad overview and expand it to a hierarchy of detailed diagrams.[11]

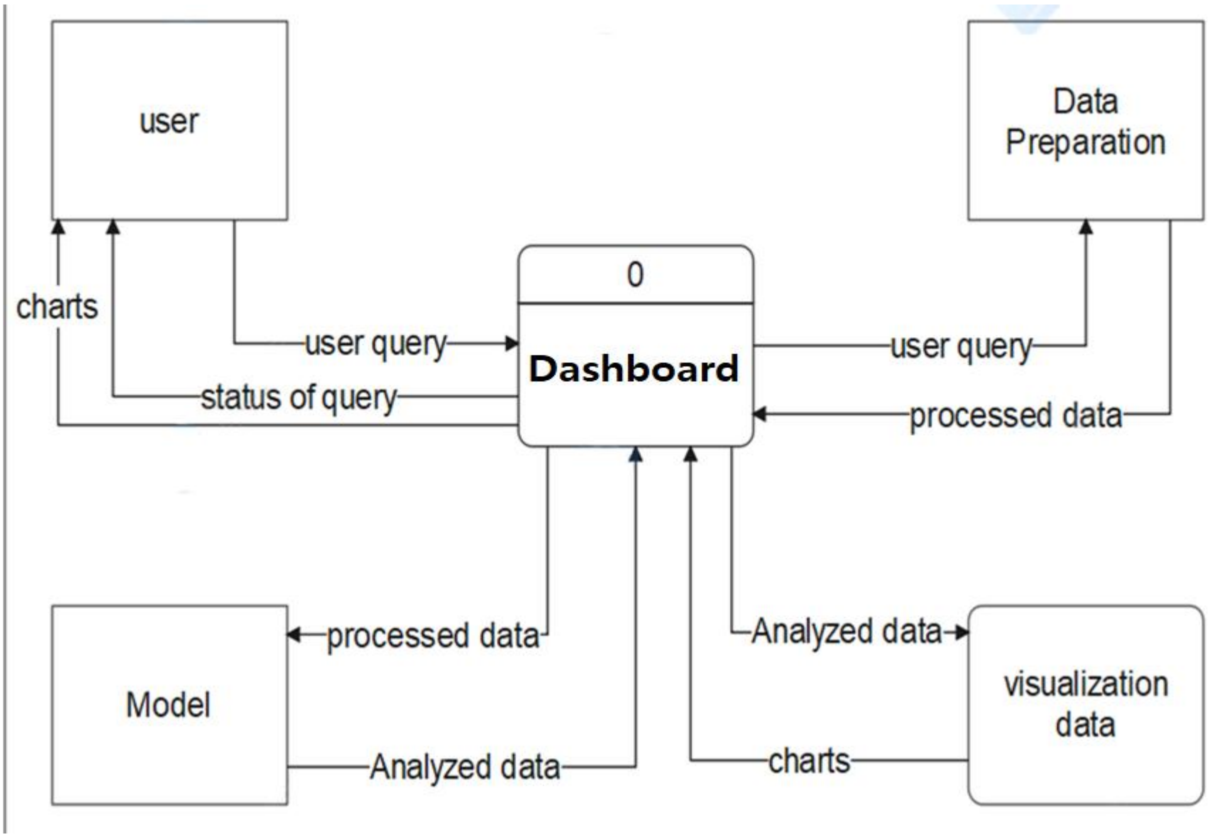


Figure 4.2 Data Flow Diagram

4.3 Class Diagram :

The UML Class diagram is a graphical notation used to construct and visualize object-oriented systems. A class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system.[12]

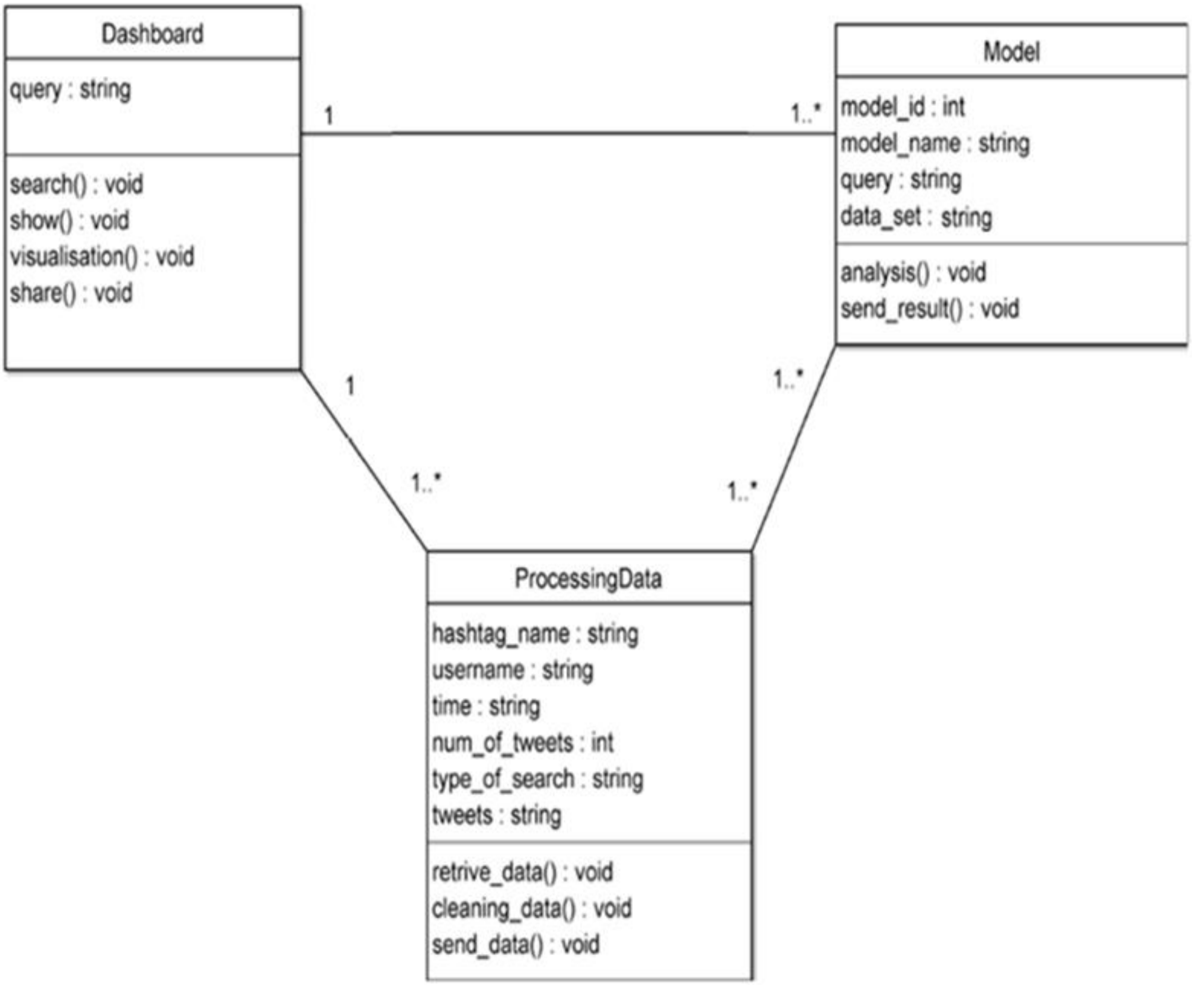


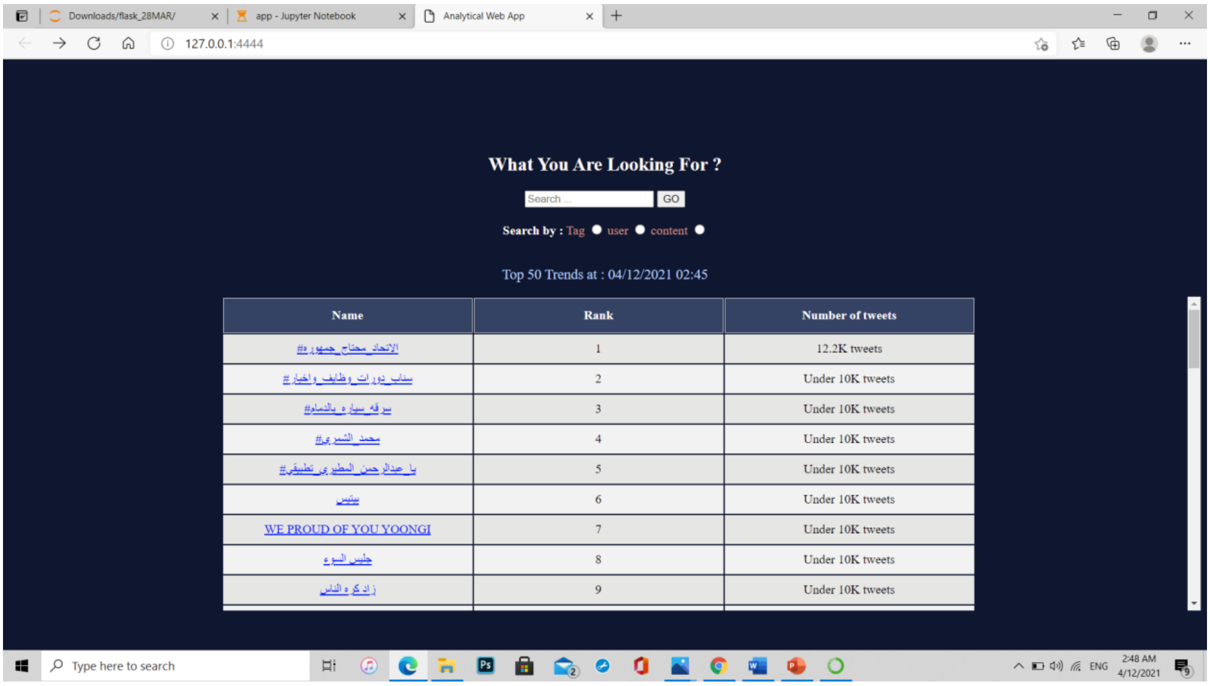
Figure 4.3 Class Diagrams

4.4 Database Design :

This project is not using a database.

4.5 User Interface Prototype :

user interface -UI design is the process designers use to build interfaces in software or computerized devices, focusing on looks or style. Designers aim to create interfaces which users find easy to use and pleasurable. UI design refers to graphical user interfaces and other forms, voice-controlled interfaces.



What You Are Looking For ?

Search ... GO

Search by : Tag ● user ● content ●

Top 50 Trends at : 04/12/2021 02:45

Name	Rank	Number of tweets
#الاتحاد_محتاج_جمهوره	1	12.2K tweets
#سناب_دورات_وظايف_واخبار	2	Under 10K tweets
#سرقه_سياره_بالدمام	3	Under 10K tweets
#محمد_الشمري	4	Under 10K tweets
#يا_عبدالرحمن_المطيري_تطبيقي	5	Under 10K tweets
بيتيس	6	Under 10K tweets
WE PROUD OF YOU YOONGI	7	Under 10K tweets
جليس السوء	8	Under 10K tweets
زاد كره الناس	9	Under 10K tweets

CHAPTER 5

System Implementation

5.1 System Implementation

Before we touch upon the implementation stage, we would like at the outset to list some things that will explain the course of establishing the project. We will also see some things that forced us to change the techniques used in the implementation of the project, and why did we do that?

In the planning and design stage, we searched for similar projects, and we searched for the mechanism of implementation and what techniques are used, and we found some have used a Python framework called Dash, Dash is a Python framework for building analytic web applications.

We explored some open-source dashboards implemented with Dash, we saw many of them that were cool and interesting, and the thing that impressed us most was the high-level interactive charts. We created the home page, linked it to the datagrams, and created the interactive charts, after we dug deeper, we found that Dash is not suitable for our project! Dash is a great framework, and as we mentioned earlier it was designed to create building web analytic applications, but we thought we wanted to create a project that is more than just an analytical dashboard of limited development, we wanted to create a web application that was more scalable, faster, and more flexible, so we looked for a better alternative option and found Flask. We also found that we can create high-level interactive diagrams like the one that Dash uses, so we found that we can do this using the PLOTLY.JS library.

5.1.1 Struggle to get Twitter API !

Lately it has become very difficult to get the Twitter API.

Since we have been trying to obtain the API since last term, but unfortunately every request that we send we receive rejection, we requested help from our supervisor, he provided us with help and explained to us some things that help in sending a clear request, we also sent a request hoping that this time we will get API but unfortunately, we got rejected!

Our supervisor provided his API, we appreciate his efforts in helping us.

5.2- Implementation Process

5.2.1 Front End

The front end of a website is the part that users interact with. Everything that you see when you are navigating around the Internet, from fonts and colors to dropdown menus and sliders, is a combo of HTML, CSS, and JavaScript being controlled by your computer's browser. [13]

we use HTML for the basic structure of webpages , CSS to designing and styling of the structure or html elements, JavaScript to making the webpages interactive. Plotly.js to create high-level interactive charts.

5.2.2 Back End

The back end refers to parts of a computer application or a program's code that allow it to operate and that cannot be accessed by a user. Most data and operating syntax are stored and accessed in the back end of a computer system. Typically, the code is comprised of one or more programming languages. The back end is also called the data access layer of software or hardware and includes any functionality that needs to be accessed and navigated to by digital means.[14]

For Back-End we used Python(Flask).And we do not use database because our web application is originally based on Twitter API, we take the data, process it, and then display it.

5.2.3 Receiving and processing data

To receive data like tweets ,trends, whatever on twitter, we use twitter API, also we use Mapbox API to get custom online maps.

After receiving data, we process it for analysis and visualization.

5.2.4 Data Visualization

After we have cleaned the data, we define the elements that we are going to Visualize, and at this phase we will use plotly.js library, which provides interactive charts.

Also, we use Mapbox API to provide a geo maps.

We have chosen Mapbox for the following reasons:

- Free API up to 50,000 monthly loads [15].
- Customization of maps is easy.
- Easy to integrate.

5.2.5 Detecting abusive contents

Unfortunately, after the decision to submit the study was issued, the time became very short and we were not able to complete this stage, as it is a stage that requires a lot of time and effort, for example: If we want to build a AI model to track the offending content, it is necessary to provide a huge number of data to train the model on it in order to achieve Acceptable or almost high accuracy in classifying content.

5.3 Techniques and technologies used:

5.3.1 Techniques and technologies:

The Techniques Used	Description
Twitter	A social media platform where people communicate with one another using 280-character tweets, images, videos, and hashtags. [16]
PYTHON	Python is a programming language that lets you work quickly and integrate systems more effectively.
FLASK	Flask is a web framework, it's a Python module that lets you develop web applications easily. It's has a small and easy-to-extend core: it's a microframework that doesn't

	include an ORM (Object Relational Manager) or such features.[17]
WEB SCRAPING	Web scraping refers to the extraction of data from a website. This information is collected and then exported into a format that is more useful for the user. Be it a spreadsheet or an API.[18]
API	An API is a set of definitions and protocols for building and integrating application software. API stands for application programming interface. APIs let your product or service communicate with other products and services without having to know how they're implemented. [19]
PYCHARM	PyCharm is a dedicated Python Integrated Development Environment (IDE) providing a wide range of essential tools for Python developers, tightly integrated to create a convenient environment for productive Python, web, and data science development.[20]
HTML	HTML stands for Hypertext Markup Language. It allows the user to create and structure sections, paragraphs, headings, links, and blockquotes for web pages and applications.[21]
CSS	CSS stands for Cascading Style Sheets. It is a style sheet language which is used to describe the look and formatting of a document written in markup language.[22]
JS	JavaScript is a dynamic computer programming language. It is lightweight and most commonly used as a part of web pages, whose implementations allow client-side script to interact with the user and make dynamic pages. It is an interpreted programming language with object-oriented capabilities.[23]
SLACK	Slack is a workplace communication tool, "a single place for messaging, tools and files." This means Slack is an instant messaging system with lots of add-ins for other workplace tools. The add-ins aren't necessary to use Slack, though, because the main functionality is all about

	talking to other people. There are two methods of chat in Slack: *channels* (group chat), and *direct message* or *DM* (person-to-person chat). Let's take a quick look at the user interface. [24]
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. More technically, Colab is a hosted Jupyter notebook service that requires no setup to use, while providing free access to computing resources including GPUs.[25]
Plotly.js	Plotly.js is a standalone Javascript data visualization library, and it also powers the Python and R modules named plotly in those respective ecosystems (referred to as Plotly.py and Plotly.R). Plotly.js can be used to produce dozens of chart types and visualizations, including statistical charts, 3D graphs, scientific charts, SVG and tile maps, financial charts and more.[26]
Mapbox	Mapbox is an American provider of custom online maps for websites and applications such as Foursquare, Lonely Planet, Facebook, the Financial Times, The Weather Channel and Snapchat.Since 2010, it has rapidly expanded the niche of custom maps, as a response to the limited choice offered by map providers such as Google Maps. Mapbox is the creator of, or a significant contributor to, some open source mapping libraries and applications, including the Mapbox GL-JS JavaScript library, the MBTiles specification, the TileMill cartography IDE, the Leaflet JavaScript library, and the CartoCSS map styling language and parser.[27]

Table: 5.3.1 Techniques and technologies

5.3.2 Library Used:

Library Used	Description
bs4	For Web scraping
pandas	for data analysis. **Pandas** allows importing data from various file formats such as comma-separated values, JSON, SQL, Microsoft Excel. **Pandas** allows various data manipulation operations such as merging, reshaping, selecting, as well as data cleaning, and data wrangling features.[28]
re	For Regular expression operations
urllib3	Helpers for retrying requests and dealing with HTTP redirects.[29]
warnings	to alert the user of some condition in a program.[30]
tweepy	Handel twitter API
plotly	For charts
wordcloud	For wordcloud figure
arabic_reshaper	To Reconstruct Arabic sentences to be used in applications that don't support Arabic script.[31]
bidi.algorithm	to write a reordered version of the string.
json	For JSON Files
requests	For making HTTP requests in Python.

Table: 5.3.2 Library Used

5.4 Sample Code:

5.4.1 app.py

```
from flask import Flask, render_template, request,redirect,url_for
from bs4 import BeautifulSoup
import pandas as pd
import re
import urllib3
import warnings
import tweepy
from pandas.io.json import json_normalize
from wordcloud import WordCloud
from arabic_reshaper import arabic_reshaper
from bidi.algorithm import get_display
import json
import requests
import os


consumer_key = "XXXXXXXXUkD8an1I8WEutE8t9"
consumer_secret = "XXXXXXXXYOVWa3tyhsHmqh4k9aEd6na6cVBvZWEbF1imsRv41L"
access_token = "XXXXXXXX0184682496-mn01gKYyMHMNBFaEyBJDlPEFV5TV6PN"
access_token_secret = "XXXXXXXXDvgXZ2eRFNLBEhICrRT1oczzJr349LUx3bDPc"

auth = tweepy.OAuthHandler(consumer_key, consumer_secret)

auth.set_access_token(access_token, access_token_secret)


api = tweepy.API(auth, wait_on_rate_limit=True,
wait_on_rate_limit_notify=True)
app = Flask(__name__)


def draw(location):
path='https://api.mapbox.com/geocoding/v5/mapbox.places/'+location+'.json
?limit=1&access_token=pk.eyJ1IjoibGVlZHoiLCJhIjoiY2tteXZ5OHR1MDgxazJ1dDdq
dDRmNzAzcyJ9.P5pBFmP_Fv1-jeHa8Wpt6g&LANG=UTF-8'

    getdata = requests.get(path)
    with open("map.json", 'wb') as f:
        f.write(getdata.content)
        f.close()
    with open('map.json', encoding="utf-8") as d:
        dictData = json.load(d)
        a = [dictData['features']]
        a = str(a)
        findex = a.find("center")
        findex = findex + 10
        lindex = a.find("geometry")
        lindex = lindex - 4
        print(a[findex:lindex])
        x = a[findex:lindex].split(",")

    site_lat = x[1]
    site_lon = x[0]

    return site_lat, site_lon
```

```
@app.route('/submit',methods=['POST','GET'])
def submit():

    if request.method == 'POST' and request.form['radio'] == 'user':
        search =request.form['search']
        return redirect(url_for('user', search=search))

    else:
        search =request.form['search']
        return redirect(url_for('content',search=search))

    return render_template('submit.html',search=search,rad=radio)


def removeWeirdChars(text):
    weridPatterns = re.compile("["
                               u"\U0001F600-\U0001F64F"
                               u"\U0001F300-\U0001F5FF"
                               u"\U0001F680-\U0001F6FF"
                               u"\U0001F1E0-\U0001F1FF"
                               u"\U00002702-\U000027B0"
                               u"\U000024C2-\U0001F251"
                               u"\U0001f926-\U0001f937"
                               u'\U00010000-\U0010ffff'
                               u"\u200d"
                               u"\u2640-\u2642"
                               u"\u2600-\u2B55"
                               u"\u23cf"
                               u"\u23e9"
                               u"\u231a"
                               u"\u3030"
                               u"\ufe0f"
                               u"\u2069"
                               u"\u2066"
                               u"\u200c"
                               u"\u2068"
                               u"\u2067"
                               u"RT"
                               "]+", flags=re.UNICODE)
    return weridPatterns.sub(r'', text)
@app.route('/content/<search>/')
def content(search):



    max_tweets = 30
    searched_tweets = [status for status in tweepy.Cursor(api.search,
q=search).items(max_tweets)]
    searched_tweets = [i._json for i in searched_tweets]



    df = json_normalize(searched_tweets)

    df = pd.DataFrame(df, columns=['created_at', 'text','source'])
    of = pd.DataFrame(df, columns=['text'])
    df.to_csv("content.csv", index=True)
    df = pd.read_csv('content.csv', encoding='utf8', header=0)
    contentlist = list(df.values)
#rt=retweets, mn=mentions, a=android,p=iphone , u= unknown
    rt=0
    mn=0
    for i in contentlist:
        if 'RT' in i[2]:
            rt=rt+1

    for i in contentlist:

        if '@' in i[2]:
            mn = mn + 1
    a =0
    p=0
    u=0
    for i in contentlist:
        if 'iPhone' in i[3]:
            p=p+1
        elif 'Android' in i[3]:
            a=a+1
        else:
            u=u+1


    text = removeWeirdChars(str(of['text']))
    text = [''.join(sentence) for sentence in text]
    text = ''.join(text)
    reshaped_text = arabic_reshaper.reshape(text)
    arabic_text = get_display(reshaped_text)

    wordcloud = WordCloud(font_path='NotoNaskhArabic-Regular.ttf',
width=700, height=300,
                          background_color="white").generate(arabic_text)

    os.remove("static/images/pic.jpg")

    wordcloud.to_file("static/images/pic.jpg")

    return
render_template('content.html',content=contentlist,q=search,rt=rt,mn=mn,a
=a,p=p,u=u)




@app.route('/user/<search>/')
def user(search):
    try:
        user = api.get_user(search)
    except tweepy.error.TweepError:

       return render_template('error.html')
    dam=[]


    for follower in user.followers():
        if not follower.location:
            follower.location = 'Unknown'
               dam.append([follower.name,follower.screen_name,
                           follower.created_at,follower.statuses_count,
                           follower.followers_count,
                           follower.friends_count,follower.location])
    if not user.location:
        user.location= 'unknown'
    else:
        a = draw(user.location)



    return render_template('user.html',search=search,
                           tweets=user.statuses_count,
                           user_name = user.name,
                           user_desc = user.description,
                           user_location = user.location,
                           followers=user.followers_count,
                           following=user.friends_count,
                           crated=user.created_at,dam=dam,
                           a=a[0], b=a[1])


@app.route('/tag/<search>/<rank>/<num>/')
def tag(search,rank,num):
    return render_template('tag.html',search=search,rank=rank,num=num)




@app.route('/')
def Stocksa():

    warnings.filterwarnings('ignore')
    http = urllib3.PoolManager()
    data = []

    url = 'https://getdaytrends.com/saudi-arabia'
    resp = http.request('GET', url)
    html_content = resp.data.decode('utf-8')
    soup = BeautifulSoup(html_content, "lxml")
    rank = 0

    x = soup.body.findAll('table')
    total = []
    td = []
    for i in x[0].findAll('td'):
        if i.div:
            total.append(
                str(i.div).replace('<div class="desc"><span class="small
text-muted">', '').replace('</span></div>',''))

        if i.title:
            td.append(re.search(r'\<title\>(.*?)\:',
i.title.decode()).group(1))
    for i in x[1].findAll('td'):
        if i.div:
            total.append(
                str(i.div).replace('<div class="desc"><span class="small
text-muted">', '').replace('</span></div>',''))
        if i.title:
            td.append(re.search(r'\<title\>(.*?)\:',
i.title.decode()).group(1))
```

```
    for j in range(len(td)):
        rank = rank + 1
        numbers=re.findall('[0-9.]+', total[j])

        numbers=str(numbers).replace("['","").replace("']","")

        data.append([rank, td[j], total[j],numbers])

    df = pd.DataFrame(data, columns=['rank', 'trend', 'total','numbers'])

    df.to_csv("TRENDS1.csv", index=False)
    df = pd.read_csv('TRENDS1.csv', encoding='utf8', header=0)

    stocklist = list(df.values)
    trend = 'https://twitter.com/search?q=%23'

    return render_template('index.html', stocklist=stocklist,
trend=trend)




if __name__ == '__main__':
    app.run(debug=True)
```

5.4.2 index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Analytical Web App</title>
     <link href="../static/css/index.css" rel="stylesheet">
        <script src="https://cdn.plot.ly/plotly-latest.min.js"></script>

</head>
<body>

<br><br><br><br><br>

        <h2> What You Are Looking For ?</h2>
        <form   action="/submit" class="form3" method="post">
        <input type="search" name="search" placeholder="Search ..."
required>
        <input type="submit" value="GO">
            <br><br>
            <b>Search by :</b>
            <label>user</label>
        <input type="radio" name="radio" value="user"required >
            <label>content</label>
        <input type="radio" name="radio" value="content" >
        </form>

<br>
<p>Top 50 Trends at : <span id='datetime'></span></p>
<div class="divt">
     <table class="center">
              <thead>
                <tr>
                  <th>Name</th>
                  <th>Rank</th>
                    <th>Number of tweets</th>
                </tr>
              </thead>
              <tbody>
                {% for value in stocklist%}
                <tr>
                {% set i =value[1] %}
                   {% if i[0]=='#' %}
                    {% set i =i[1:] %}
                {% endif %}
                    <td><a href="/tag/{{i}}/{{value[0]}}/{{
value[2]}}">{{ value[1] }}</a></td>
                    <td>{{ value[0]}}</td>
                    <td>{{ value[2]}}</td>
                </tr>
                {% endfor %}
              </tbody>
     </table>
</div>
<br><br>
<div id="test" width="1900" height="1000px">
    <script>

        var data = [{
            x: ['{{stocklist[0][1]}}', '{{stocklist[1][1]}}',
'{{stocklist[2][1]}}', '{{stocklist[3][1]}}', '{{stocklist[4][1]}}'],
            y: [ {{stocklist[0][3]}},  {{stocklist[1][3]}},
{{stocklist[2][3]}},  {{stocklist[3][3]}}, {{stocklist[4][3]}}],
           type: 'bar',
            title: "wwww"
        }];

        var layout = {

 title:{
        text: "Top 5 Trends",
          horizontalAlign: "center",
      },
            height:500,width: 1100,paper_bgcolor: '#0E162E',font: {size:
18,color:'white'},margin: {
          l: 200,
          r: 20,
          b: 100,
          t: 50,
          paddingBottom: 20
        },};
        var config = {responsive: true};
        TESTER = document.getElementById('test');
        Plotly.newPlot(TESTER, data, layout, config);

    </script>
</div>
<br><br>
</body>
<script>
var dt = new Date();
document.getElementById("datetime").innerHTML =
(("0"+(dt.getMonth()+1)).slice(-2)) +"/"+ (("0"+dt.getDate()).slice(-2))
+"/"+ (dt.getFullYear()) +" "+ (("0"+dt.getHours()).slice(-2)) +":"+
(("0"+dt.getMinutes()).slice(-2));
</script>
</html>
```

5.4.3 user.html

```
<html>
    <head>
        <meta charset="UTF-8">
        <link type="text/css" href="{{
url_for('static',filename='css/user.css') }}" rel="stylesheet"><link
href="https://api.mapbox.com/mapbox-gl-js/v2.2.0/mapbox-gl.css"
rel="stylesheet">
    <script src="https://api.mapbox.com/mapbox-gl-js/v2.2.0/mapbox-
gl.js"></script>
    <script src="https://kit.fontawesome.com/237a5e667c.js"
crossorigin="anonymous"></script>
     <title>Analytical web App</title>
    </head>
    <body>
        <td>
            <h1>You are Search for : </h1>

            <div id='d1'>
                <p>User</p><p id="v1">{{ search }}</p>
                <i class="fab fa-twitter"></i>
            </div>

            <div id='d2'>
                <p>Name</p><p id="v2">{{ user_name}}</p>
                <i class="far fa-user"></i>
            </div>

            <div id='d3'>
                <p>Location </p><p id="v3">{{user_location}}</p>
                <i class="fas fa-map-marker-alt"></i>
            </div>

            <div id='d4'>
                <p>Descraption </p><p id="v4">{{ user_desc }}</p>
                <i class="far fa-address-card"></i>
            </div>

            <div id='d5'>
                <p>Followers </p><p id="v5">{{followers}}</p>
                <i class="fas fa-user-friends"></i></div>

            <div id='d6'>
                <p>Following</p><p id="v6">{{ following}}</p>
                <i class="fas fa-users"></i>
            </div>

            <div id='d7'>
                <p>Tweets</p><p id="v7">{{ tweets}}</p>
                <i class="fas fa-user-edit"></i>
            </div>

            <div id='d8'>
                <p>Crated at :</p><p id="v8">{{ crated}}</p>
                <i class="far fa-calendar-alt"></i>
            </div>

        <br><br>
        <h1>Recent 20 Followers</h1>
            <div class="divt">
```

```
             <table class="center">
                      <thead>
                        <tr>
                          <th>Name</th>
                          <th>User</th>
                            <th>Crated at:</th>
                            <th>Tweets</th>
                            <th>Followers</th>
                            <th>following</th>
                            <th>location</th>
                        </tr>
                      </thead>
                      <tbody>
                        {% for value in dam%}
                         <tr>
                            <td>{{ value[0]}}</td>
                            <td>{{ value[1]}}</td>
                            <td>{{ value[2]}}</td>
                            <td>{{ value[3]}}</td>
                            <td>{{ value[4]}}</td>
                            <td>{{ value[5]}}</td>
                            <td>{{ value[6]}}</td>
                         </tr>
                        {% endfor %}
                      </tbody>
             </table>
            </div>


    <h1>User Location</h1>
    <div id="map"></div>

        <script>
            mapboxgl.accessToken =
'pk.eyJ1IjoibGVlZHoiLCJhIjoiY2tteXZ5OHR1MDgxazJ1dDdqdDRmNzAzcyJ9.P5pBFmP_
Fv1-jeHa8Wpt6g';
        var map = new mapboxgl.Map({
        container: 'map',
        style: 'mapbox://styles/mapbox/dark-v10',
        center: [44,24],
        {#//styles/mapbox/streets-v11#}
        zoom: 3
        });

        var marker = new mapboxgl.Marker({
            size:10,
        color: "#324263",
        draggable: true
        }).setLngLat([{{ b }}, {{a}}])
        .addTo(map);

        map.addControl(
        new MapboxGeocoder({
        accessToken: mapboxgl.accessToken,
        mapboxgl: mapboxgl
        })
        );
        </script>
    </body>
</html>
```

5.4.4 tag.html

```
<html>
    <head>
<link type="text/css" href="{{ url_for('static',filename='css/tag.css')
}}" rel="stylesheet">

<script src="https://kit.fontawesome.com/237a5e667c.js"
crossorigin="anonymous"></script>

    </head>
<body>
    <div id='d4'>
  <a href="/"><h2>Back</h2></a>
        </div>

    <h1>You are Search for : </h1>

    <div id=d1>
        <p>Rank</p>
        <p id="v1">{{ rank }}</p>
        <i class="fas fa-sort-amount-up-alt"></i>
        </div>

    <div id='d2'>
        <p>Tag</p>
        <p id="v2">{{ search}}</p>
        <i class="fas fa-hashtag"></i>
        </div>

    <div id='d3'>
        <p>Tweets </p>
        <p id="v3">{{num}}</p>
       <i class="fas fa-sort-numeric-up-alt"></i>
        </div>


    </body>
</html>
```

5.4.5 content.html

```
<!DOCTYPE html>
<html lang="en">
<head>
        <link type="text/css" href="{{
url_for('static',filename='css/content.css') }}" rel="stylesheet">

    <script src="https://cdn.plot.ly/plotly-latest.min.js"></script>
    <meta charset="UTF-8">
    <title>Title</title>

</head>
<body>
<br>
<br>
<h2>You are search for : {{ q }}</h2>
<div class="divt">
     <table class="center">
              <thead>
                <tr>
                    <th>Num</th>
                  <th>Created at :</th>
                  <th>Tweet</th>
                    <th>Device</th>
                </tr>
              </thead>
              <tbody>

                {% for value in content%}
                <tr>


                    <td>{{ value[0]+1}}</td>
                  <td>{{ value[1][:20]}}</td>
                    {% if 'RT'in value[2] %}
                    <td id="tt">{{ value[2]}}</td>

                    {% else %}
                <td>{{ value[2]}}</td>
                    {% endif %}
                    <td>{{value[3][72:-4]}}</td>
                </tr>
                {% endfor %}
              </tbody>
<tr>
    <td></td>
    <td></td>
    <td>Numbers of Retweets ={{ rt }}</td>
    <td></td>
</tr>
     </table></div>

<br><br>

<div id="test" width="1900" height="1000px">
    <script>

        var data = [{

            x: ['Mentions', 'Retweets'],
            y: [{{mn}}, {{rt}} ],
           type: 'bar',
            title: "wwww"
        }];

        var layout = {

 title:{
        text: "Numbers of Retweets and Mentions ",
          horizontalAlign: "center",
      },

            height:500,width: 1100,paper_bgcolor: '#0E162E',font: {size: 18,color:'white'},margin: {
          l: 200,
          r: 20,
          b: 100,
          t: 50,
          paddingBottom: 20
        },};
        var config = {responsive: true};
        TESTER = document.getElementById('test');
        Plotly.newPlot(TESTER, data, layout, config);

    </script>
</div>

<br><br>
<h2>WordCloud</h2>
<div id="a">
<img src="{{ url_for('static',filename='images/pic.jpg') }}"/>
</div>
<br><br>
<h2>Pie chart for Devices</h2>
<div id="test1" width="1900" height="1000px">

    <script>
var data = [{
  values: [{{a}}, {{p}}, {{u}}],
  labels: ['Android', 'Iphone','Unknown'],
  type: 'pie'
}];

var layout = {
  height: 400,
  width: 700,
    paper_bgcolor: '#0E162E',
    font: {size: 18,color:'white'},margin: {
          l: 200,
          r: 20,
          b: 100,
          t: 50,
          paddingBottom: 20
        },};

Plotly.newPlot('test1', data, layout);
    </script></div>

</body>
</html>
```

5.4.6 error.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <link type="text/css" href="{{
url_for('static',filename='css/error.css')}}" rel="stylesheet">
    <meta charset="UTF-8">
    <title>Error</title>
</head>
<body>
<h1>404</h1>


<p>The user you are looking for is not exist ! , Try to Enter The
username in English.</p>

<a href="/"> back</a>

</body>
</html>
```

5.4.7 index.css

```
 td, th {
  border: 1px solid #ddd;
  padding: 8px;
  width: 300px;
  text-align: center;
}
.divt{
    text-align: center;
    style=width:320px;
    height:400px;
    overflow:auto;
}

p{
    text-align: center;
    color: #b6d1fc;
    font-size: large;
}
title{
    text-align: center;
    }

.center{
    margin-left: auto;
    margin-right: auto;
    text-align: center;
    cellspacing=0px;
    cellpadding=1px;
    border=1px;
    width=300px;
}

tr:nth-child(even){background-color: #f2f2f2;}

tbody{
background-color: #e6e6e4;
}
tr:hover {background-color: #ddd;}
th {
  padding-top: 12px;
  padding-bottom: 12px;
  text-align: center;
  background-color: #324263;
  color: white;
}

body{
background-color:#0E162E ;
}
b{
    color:white;
}

h2{
    text-align: center;
    color: snow;
}

.form3{
```

```
text-align: center;

}
label{
    color: #de9890;
}
```

5.4.8 user.css

```
body
{
    background-color: #081226;
}

.divt{
    text-align: center;
     style=width:320px;
    height:400px;
    overflow:auto;
}
#map { margin-left:23%;width: 50%;height: 50% }

.center {
  margin-left: auto;
  margin-right: auto;
    text-align: center;
     cellspacing=0px;
     cellpadding=1px;
     border=1px;
     width=300px;
}

tr:nth-child(even){background-color: #f2f2f2;}

tbody{
background-color: #e6e6e4;
}
tr:hover {background-color: #ddd;}
th {
  padding-top: 12px;
  padding-bottom: 12px;
  text-align: center;
  background-color: #324263;
  color: white;
}


h1{
    margin-top: 100;
    text-align: center;
    color:white;
}

#d1 p{
text-align: center;
padding-top: 8;
}
#d2 p{
    text-align: center;
```

```
        margin-top: 1;
    padding-top: 8;
}
#d3 p{
    text-align: center;

    padding-top: 8;
}
#d4 p{
    text-align: center;

    padding-top: 8;
}
#d5 p{
    text-align: center;

    padding-top: 8;
}
#d1 .fab{
   margin-left: 200;
    margin-top: -10;
    font-size: 30;
    margin-top: -35;
}
#d2 .far{
   margin-left: 200;
   font-size: 30;
    margin-top: -50;
}

#d3 .fas{
   margin-left: 205;
margin-top: -42;
   font-size: 30;
}
#d4 .far{
   margin-left: 205;
   font-size: 30;
    margin-top: -77;
}


#d5 .fas{
   margin-left: 200;
margin-top: -40;
   font-size: 28;
}
        #d6 .fas{
   margin-left: 205;
   font-size: 32;
    margin-top: -40;
}
            #d7 .fas{
   margin-left: 205;
   font-size: 32;
    margin-top: -44;
}
    #d8 .far{
   margin-left: 205;
   font-size: 32;
    margin-top: -40;
```

```
}
#d1{
    background-color: #324263;
    color: red;
    width: 250;
    height: 80;
    margin-left: 150;
    margin-top: 80;
    color:white;

  border-radius: 12px;
}
#d2{

    background-color: #324263;
    width: 250;
    height: 80;
    margin-left: 420;
    margin-top: -79;
     border-radius: 12px;
    color:white;
}
#d3{
    background-color: #324263;
    width: 250;
    height: 80;
    margin-left: 690;
    margin-top: -95;
     border-radius: 12px;
    color:white;
}

#d4{
    background-color: #324263;
   width: 250;
    height: 80;
    margin-left: 960;
    margin-top: -95;
     border-radius: 12px;
    color:white;
}


 #d5{
    background-color: #324263;
    width: 250;
    height: 80;
    margin-left: 150;
    border-radius: 12px;
    color:white;
}

    #d6{

    background-color: #324263;
    width: 250;
    height: 80;
    margin-left: 420;
    margin-top: -93;
    border-radius: 12px;
    color:white;
```

```
}
    #d7{
    background-color: #324263;
    width: 250;
    height: 80;
    margin-left: 690;
    margin-top: -93;
     border-radius: 12px;
    color:white;
}

#d8{
    background-color: #324263;
    width: 250;
    height: 80;
    margin-left: 960;
    margin-top: -93;
    border-radius: 12px;
    color:white;
}



    p{
        text-align: center;

      font-size: 15;
    }




    #v1{
        margin-top: -10;
        margin-right: 30;

    }
    #v2{
        position:relative;
        top:-5px;

    }
        #v3{
        margin-top: -10;

    }
        #v4{

            text-align:left;

                font-size: 13
    }
        #v5{
        margin-top: -10;
            margin-right: 30;

    }
```

5.4.9 tag.css

```css
body
{
    background-color: #081226;
}

h1{
    margin-top: 100;
    text-align: center;
    color:white;
}

#d1 p{
text-align: center;
padding-top: 8;
}
#d2 p{
    text-align: center;
        margin-top: 1;
    padding-top: 8;
}
#d3 p{
    text-align: center;

    padding-top: 8;
}
#d4 p{
    text-align: center;

    padding-top: 8;
}
#d5 p{
    text-align: center;

    padding-top: 8;
}
#d1 .fas{
   margin-left: 220;

    font-size: 40;
    margin-top: -65;
}
#d2 .fas{
   margin-left: 225;
   font-size: 35;
    margin-top: -65;
}

#d3 .fas{
   margin-left: 235;
margin-top: -65;
   font-size: 40;
}

#d1{
    background-color: #324263;
    color: red;
    width: 280;
    height: 80;
    margin-left: 250;
    margin-top: 80;
    color:white;

  border-radius: 12px;
}
#d2{

    background-color: #324263;
    width: 280;
    height: 80;
    margin-left: 555;
    margin-top: -79;
     border-radius: 12px;
    color:white;
}
#d3{
    background-color: #324263;
    width: 280;
    height: 80;
    margin-left: 860;
    margin-top: -95;
     border-radius: 12px;
    color:white;
}

#d4{
    background-color: #324263;
    width: 100;
    height: 30;
    margin-left: 86;
    border-radius: 12px;
    color:white;
}


p{
    text-align: center;
    font-size: 20;
}


#v1{
    margin-top: -20;
    font-size: 25;
}

#v2{
    position:relative;
    top:-20px;
    font-size: 17;
}
#v3{
    margin-top: -20;
    font-size: 25;
}
#d4 h2{
    color: white;
        text-align: center;
}
```

5.4.10 content.css

```css
 td, th {
  border: 1px solid #ddd;
  padding: 8px;
width: 300px;
     text-align: center;

}
.divt{
    text-align: center;
     style=width:320px;
    height:400px;
    overflow:auto;
}

p{
    text-align: center;
    color: #b6d1fc;
    font-size: large;
}
.center {
  margin-left: auto;
  margin-right: auto;
    text-align: center;
     cellspacing=0px;
     cellpadding=1px;
     border=1px;
     width=300px;
}

tr:nth-child(even){background-color: #f2f2f2;}

tbody{
background-color: #e6e6e4;
}
tr:hover {background-color: #ddd;}
th {
  padding-top: 12px;
  padding-bottom: 12px;
  text-align: center;
  background-color: #324263;
  color: white;
}

body{
background-color:#0E162E ;

}
b{
    color:white;
}

h2{
    text-align: center;
    color: snow;
}

.form3{
text-align: center;
}
label{
    color: #de9890;
}
        #tt{
            background-color: lightskyblue;
        }
 #a{
     margin-left: 300px;
 }
        #test1{
            margin-left: 300px;
        }
```

5.4.11 error.css

```
        h1{
            font-size: 200px;
            text-align: center;
            color: white;
        }
        p{
font-family: 'Josefin Sans', sans-serif;
            color: white;
            margin-left: 30%;
            margin-top: -10%;
        }
        a {
             margin-left: 46%;
  font-family: 'Josefin Sans', sans-serif;
  font-size: 14px;
  text-decoration: none;
  text-transform: uppercase;
  background: transparent;
  color: #c9c9c9;
  border: 2px solid #c9c9c9;
  display: inline-block;
  padding: 10px 25px;
  font-weight: 700;
  -webkit-transition: 0.2s all;
  transition: 0.2s all;
}

 a:hover {
  color: white;
  border-color: #324263;
}
body{
  background-color:#081226;
      }
```

CHAPTER 6

System Testing

6.1 Unit Testing

Test Case No.	**1**
Test Data	**HTML search field in the index page.**
Possibilities of defect	- **No data**: Check system response when no data is submitted. - **Invalid data**: Check system response when *Invalid* test data is submitted. - **Illegal data format**: Check system response when test data is in an invalid format.
How was the defect resolved?	**Name of Defect:** **No data** **Patching Steps:** **By make the search element as required** **Snippet code:** `<input type="search" name="search" placeholder="Search ..." required>`
	Name of Defect: **Invalid data & Illegal data format** **Patching Steps:** **Redirect the user to the errors page.**

Table: 6.1.1 Unit Testing

Test Case No.	**2**
Test Data	**HTML Radio button in the index page.**
Possibilities of defect	• **No data**: Check system response when no data is submitted.
How was the defect resolved?	**Patching Steps:** **By make the radio button element as required.** **Snippet code:** `<input type="radio" name="radio" value="user" required>`

Table: 6.1.2 Unit Testing

Test Case No.	**3**
Test Data	**Word cloud function**
Possibilities of defect	• **Data contains bad characters like:** ➢ **Emojis.** ➢ **Flags.** ➢ **Symbols**
How was the defect resolved?	**Patching Steps:** **By cleaning Data**

Table: 6.1.3 Unit Testing

Test Case No.	4
Test Data	➢ Data Visualization function.
Possibilities of defect	➢ Inability to visualize data.
How was the defect resolved?	Patching Steps: ➢ Cleaning Data before passing to the function. ➢ Ensure the correctness of the data elements that will be sent to the function.

Table: 6.1.4 Unit Testing

Test Case No.	5
Test Data	➢ Twitter API
Possibilities of defect	➢ rate limit exceeded
How was the defect resolved?	Patching Steps: ➢ Set a limit for user searching (30 Tweets).

Table: 6.1.5 Unit Testing

Test Case No.	6
Test Data	➢ Mapbox API
Possibilities of defect	➢ The user has a fake location , for Example: - on the moon - in the space
How was the defect resolved?	Patching Steps: ➢ If the location is not found, return "Unknown".

Table: 6.1.6 Unit Testing

6.2 Integration testing

Test Case No.	1
Test Data	➢ Twitter API
description	➢ Test Web Application with Twitter API Ensure that the integration process between components is correct and error free.
Status (Pass/Fail)	Pass

Table: 6.2.1 Integration testing

Test Case No.	2
Test Data	➢ MapBox API
description	➢ Test Web Application with MapBox API Ensure that the integration process between components is correct and error free.
Status (Pass/Fail)	Pass

Table: 6.2.2 Integration testing

Test Case No.	3
Test Data	➢ Data Visualization Libraries & Functions
description	➢ Test Web Application with Data Visualization Libraries and Functions. Ensure that the integration process between components is correct and error free.
Status (Pass/Fail)	Pass

Table: 6.2.3 Integration testing

Test Case No.	4
Test Data	➢ Web Application files.
description	➢ Test Web Application files like: Integrate HTML with CSS and JS files. Ensure that the integration process between components is correct and error free.
Status (Pass/Fail)	Pass

Table: 6.2.4 Integration testing

6.3 Performance testing

We have not deployed the system to an external server yet, so if we perform a performance test in the current situation, the result will be inaccurate.

6.4 User acceptance testing

The system was not deployed on external server , so we could not gather many users, just 8.

Each user used the system and then filled out the test form. In the end, I gathered and summarized the views on the system.

	Strongly Disagree	Disagree	Natural	Agree	Strongly Agree
1. The system is characterized by ease of use?				1	7
2. The system supports the Arabic language very efficiently?					8
3. Was the interactive charts very helpful?				1	7
4. The system provides knowledge about what is currently trading?					8
5. The system was able to analyse the content that you searched for?					8
6. I have not encountered any errors while using the system?					8
7. Feel completely satisfied with the system?					8

Table: 6.4 User acceptance testing

6.4.1 Conclusion

As you can see in the form above, the system impressed the users.

Users had some suggestions, the most important of which was:

- Connect the system with AI models.
- Improved user interface.

6.5 Test cases

Test Case No.	**1**
Description	**Check system response when valid data is submitted.**
Test Data	**Valid Data**
testing environment	**Web Application**
Expected Result	**Success**
Actual Result	**Success**
Status (pass/fail)	**pass**

Table: 6.5.1 Test Case

Test Case No.	**2**
Description	**Check system response when no data is submitted.**
Test Data	**No data**
testing environment	**Web Application**
Expected Result	**Will show alert that told user he must inter data**
Actual Result	**Same as expected**
Status (pass/fail)	**pass**

Table: 6.5.2 Test Case

Test Case No.	**3**
Description	**Check system response when Invalid data or Illegal data format is submitted.**
Test Data	**Invalid data & Illegal data format**
testing environment	**Web Application**
Expected Result	**Will redirect user to errors page.**
Actual Result	**Same as expected**
Status (pass/fail)	**pass**

Table: 6.5.3 Test Case

CHAPTER 7

System Demonstration

7.1 System Screen Flow

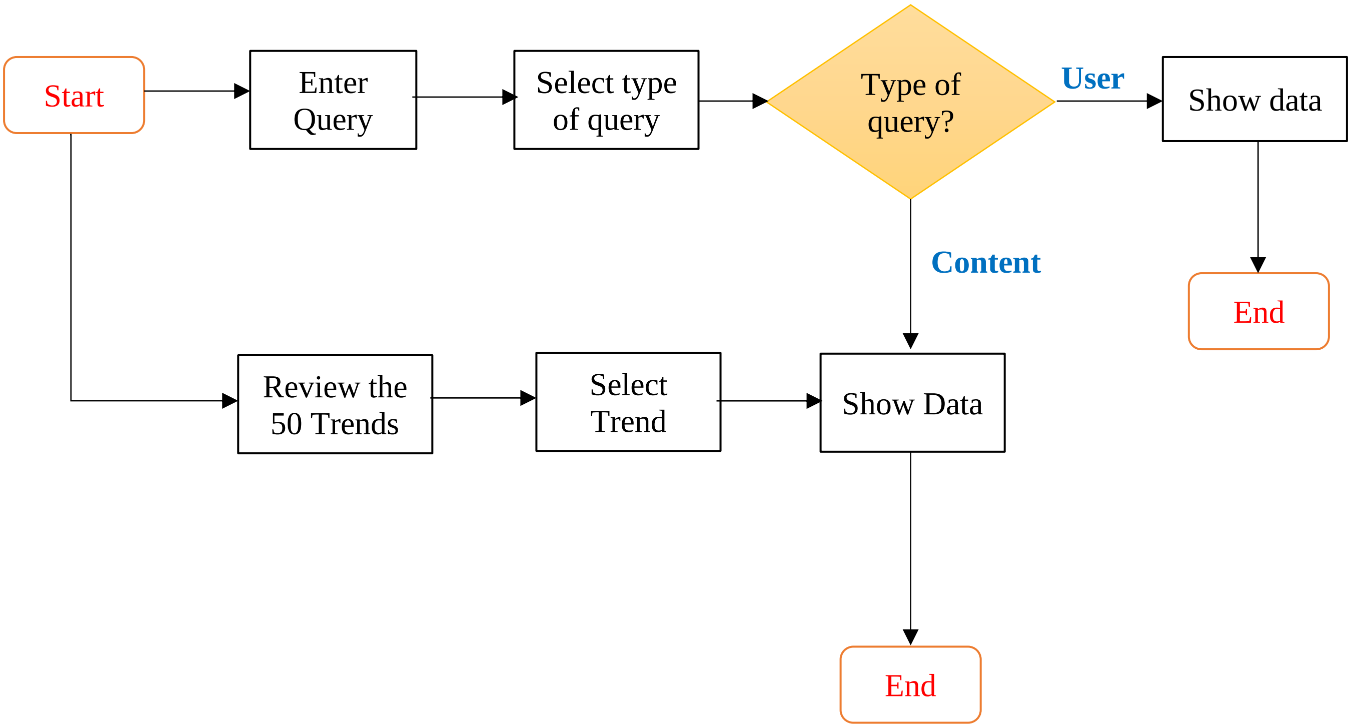
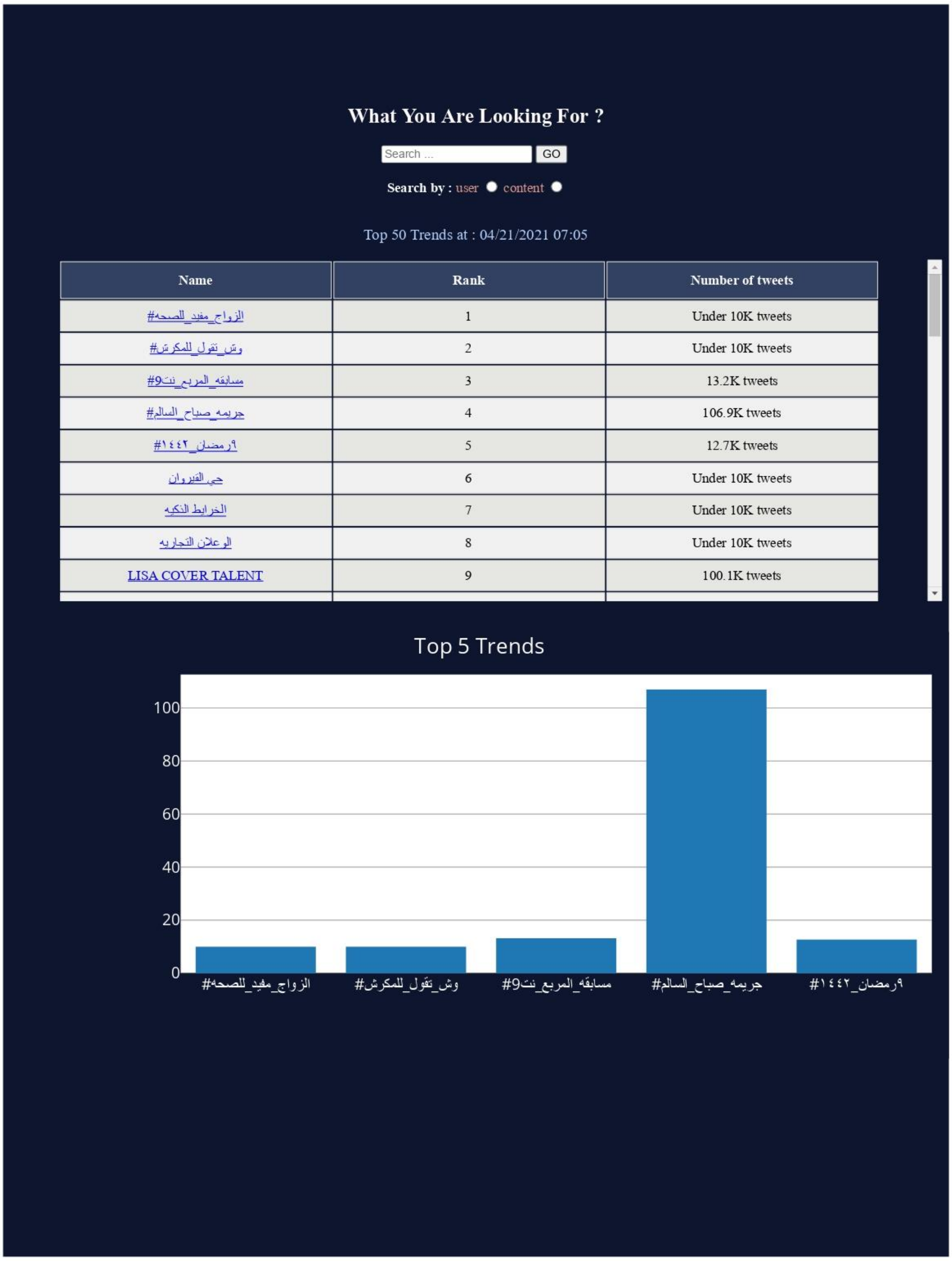


Figure 7.1: System screens flow

7.2 System Screens snapshots

7.2.1 index.html



What You Are Looking For ?

Search ... GO

Search by : user ○ content ○

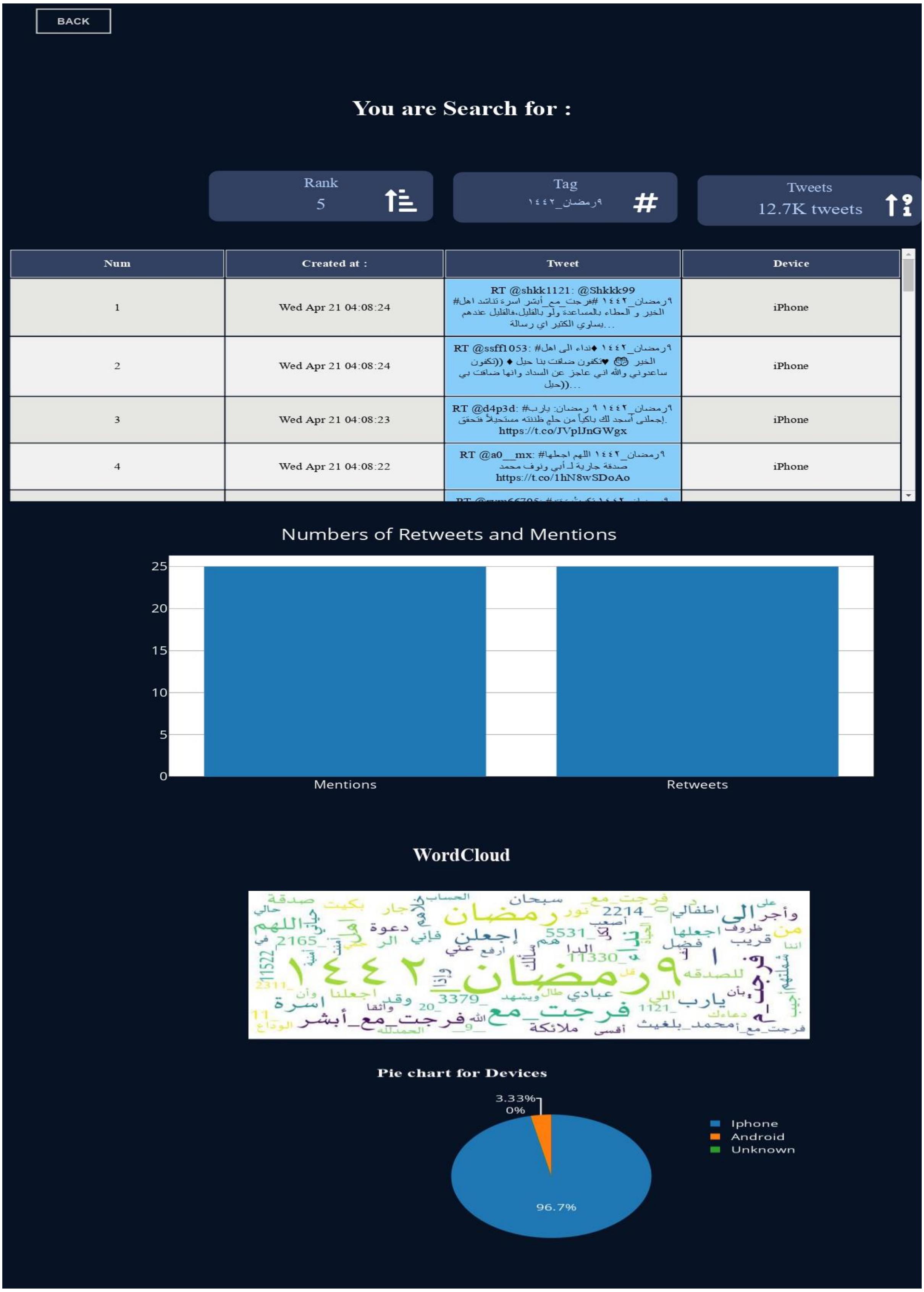
Top 50 Trends at : 04/21/2021 07:05

Name	Rank	Number of tweets
#الزواج_مفيد_للصحه	1	Under 10K tweets
#وش_تقول_للمكرش	2	Under 10K tweets
#مسابقه_المربع_نت9	3	13.2K tweets
#جريمه_صباح_السالم	4	106.9K tweets
#٩رمضان_١٤٤٢	5	12.7K tweets
حي_القيروان	6	Under 10K tweets
الخرايط الذكيه	7	Under 10K tweets
الإعلان التجاريه	8	Under 10K tweets
LISA COVER TALENT	9	100.1K tweets

Top 5 Trends

Figure: 7.2.1 index.html

7.2.2 tag.html



You are Search for :

Rank 5

Tag ٩رمضان_١٤٤٢

Tweets 12.7K tweets

Num	Created at :	Tweet	Device
1	Wed Apr 21 04:08:24	RT @shkk1121: @Shkkk99 #٩رمضان_١٤٤٢ #فرجت_مع_أبشر اسرة تناشد اهل الخير و العطاء بالمساعدة ولو بالقليل،فالقليل عندهم يساوي الكثير اي رسالة...	iPhone
2	Wed Apr 21 04:08:24	RT @ssff1053: #٩رمضان_١٤٤٢ ◆نداء الى اهل الخير 🥺 ♥تكفون ضاقت بنا حيل ◆ ((تكفون ساعدوني والله اني عاجز عن السداد وانها ضاقت بي حيل))...	iPhone
3	Wed Apr 21 04:08:23	RT @d4p3d: #٩رمضان_١٤٤٢ ٩ رمضان: يارب إجعلني أسجد لك باكياً من حلمٍ ظننته مستحيلاً فتحقق. https://t.co/JVplJnGWgx	iPhone
4	Wed Apr 21 04:08:22	RT @a0__mx: #٩رمضان_١٤٤٢ اللهم اجعلها صدقة جارية لـ أبي ونوف محمد https://t.co/1hN8wSDoAo	iPhone

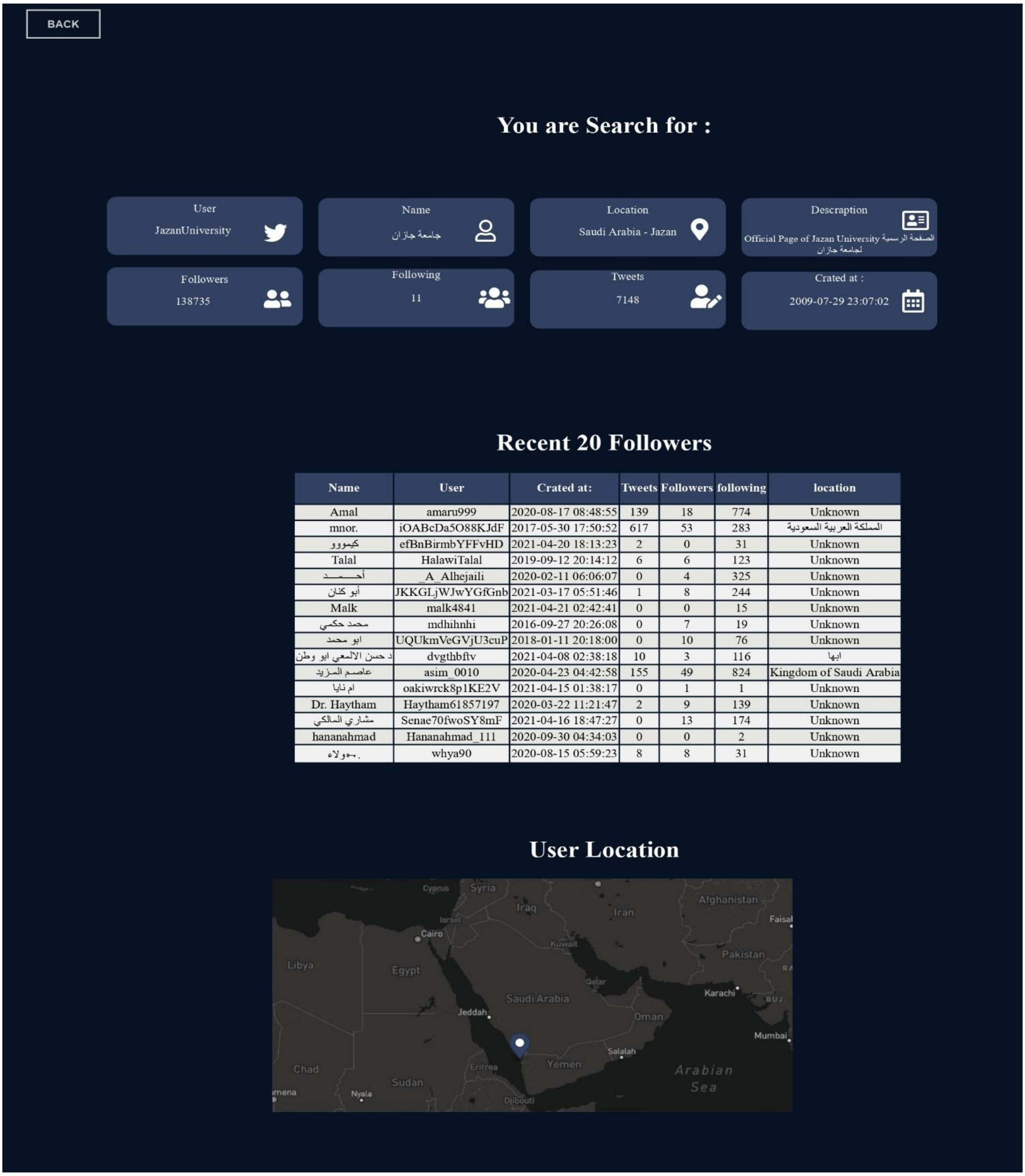
Numbers of Retweets and Mentions

WordCloud

Pie chart for Devices

Figure: 7.2.2 tag.html

7.2.3 user.html



You are Search for :

User
JazanUniversity

Name
جامعة جازان

Location
Saudi Arabia - Jazan

Descraption
Official Page of Jazan University الصفحة الرسمية لجامعة جازان

Followers
138735

Following
11

Tweets
7148

Crated at :
2009-07-29 23:07:02

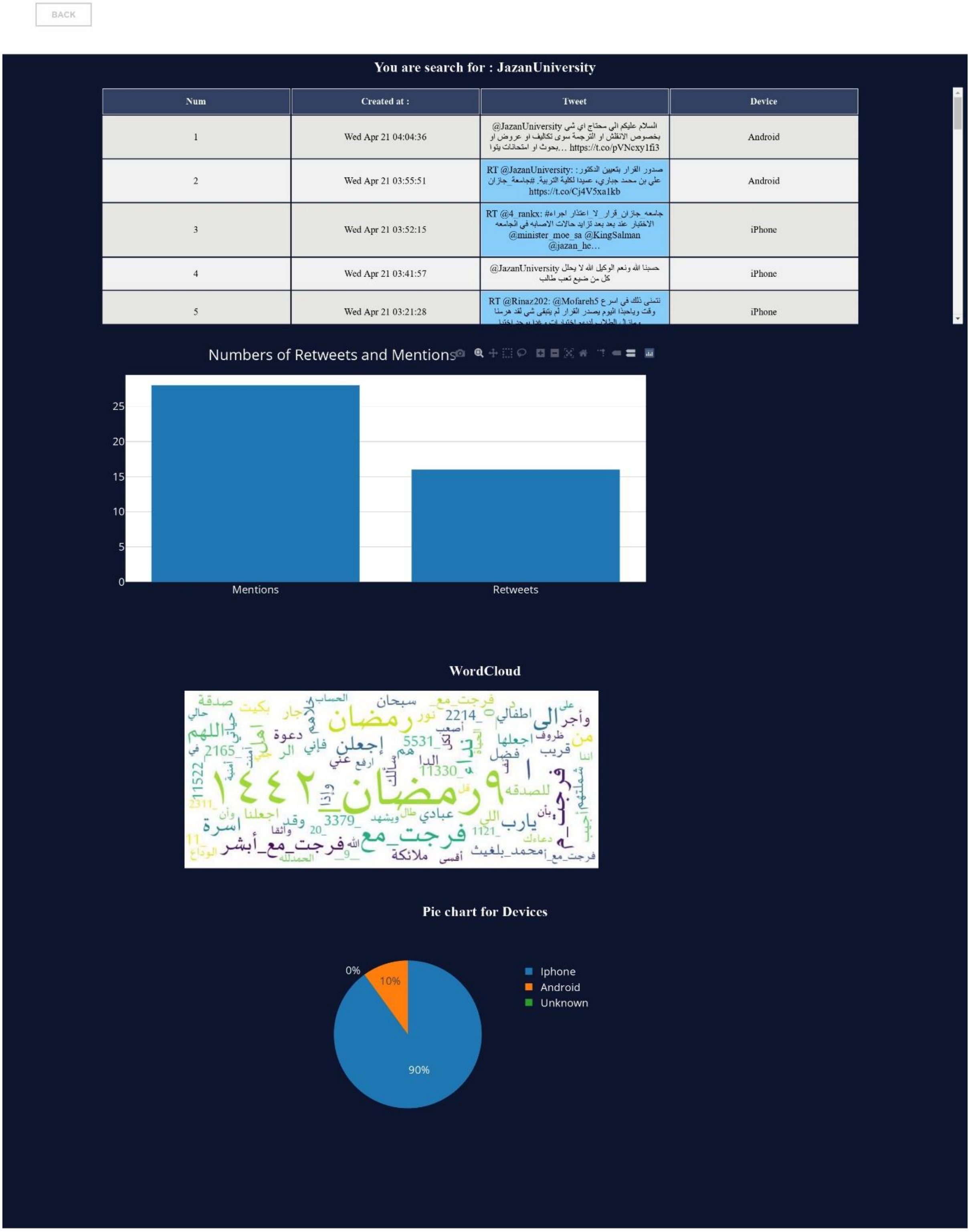
Recent 20 Followers

Name	User	Crated at:	Tweets	Followers	following	location
Amal	amaru999	2020-08-17 08:48:55	139	18	774	Unknown
mnor.	iOABcDa5O88KJdF	2017-05-30 17:50:52	617	53	283	المملكة العربية السعودية
كيموود	efBnBirmbYFFvHD	2021-04-20 18:13:23	2	0	31	Unknown
Talal	HalawiTalal	2019-09-12 20:14:12	6	6	123	Unknown
أحــــمــــد	_A_Alhejaili	2020-02-11 06:06:07	0	4	325	Unknown
أبو كنان	JKKGLjWJwYGfGnb	2021-03-17 05:51:46	1	8	244	Unknown
Malk	malk4841	2021-04-21 02:42:41	0	0	15	Unknown
محمد حكمي	mdhihnhi	2016-09-27 20:26:08	0	7	19	Unknown
ابو محمد	UQUkmVeGVjU3cuP	2018-01-11 20:18:00	0	10	76	Unknown
د حسن الالمعي ابو وطن	dvgthbftv	2021-04-08 02:38:18	10	3	116	ابها
عاصـم المـزيد	asim_0010	2020-04-23 04:42:58	155	49	824	Kingdom of Saudi Arabia
ام نايا	oakiwrck8p1KE2V	2021-04-15 01:38:17	0	1	1	Unknown
Dr. Haytham	Haytham61857197	2020-03-22 11:21:47	2	9	139	Unknown
مشاري المالكي	Senae70fwoSY8mF	2021-04-16 18:47:27	0	13	174	Unknown
hananahmad	Hananahmad_111	2020-09-30 04:34:03	0	0	2	Unknown
ـهولاء.	whya90	2020-08-15 05:59:23	8	8	31	Unknown

User Location

Figure: 7.2.3 user.html

7.2.3 content.html



BACK

You are search for : JazanUniversity

Num	Created at :	Tweet	Device
1	Wed Apr 21 04:04:36	السلام عليكم الي محتاج اي شي @JazanUniversity بخصوص الانفلش او الترجمة سوى تكاليف او عروض او بحوث او امتحانات يتوا… https://t.co/pVNcxy1fi3	Android
2	Wed Apr 21 03:55:51	RT @JazanUniversity: : صدور القرار بتعيين الدكتور علي بن محمد جباري، عميدا لكلية التربية. #جامعة_جازان https://t.co/Cj4V5xa1kb	Android
3	Wed Apr 21 03:52:15	RT @4_rankx: #جامعه_جازان_قرار_لا_اعتذار اجراء الاختبار عند بعد بعد تزايد حالات الاصابه في الجامعه @minister_moe_sa @KingSalman @jazan_he…	iPhone
4	Wed Apr 21 03:41:57	حسبنا الله ونعم الوكيل الله لا يحلل @JazanUniversity كل من ضيع تعب طالب	iPhone
5	Wed Apr 21 03:21:28	RT @Rinaz202: @Mofareh5 نتمنى ذلك في اسرع وقت وياحبذا اليوم يصدر القرار لم يتبقى شي لقد هرمنا	iPhone

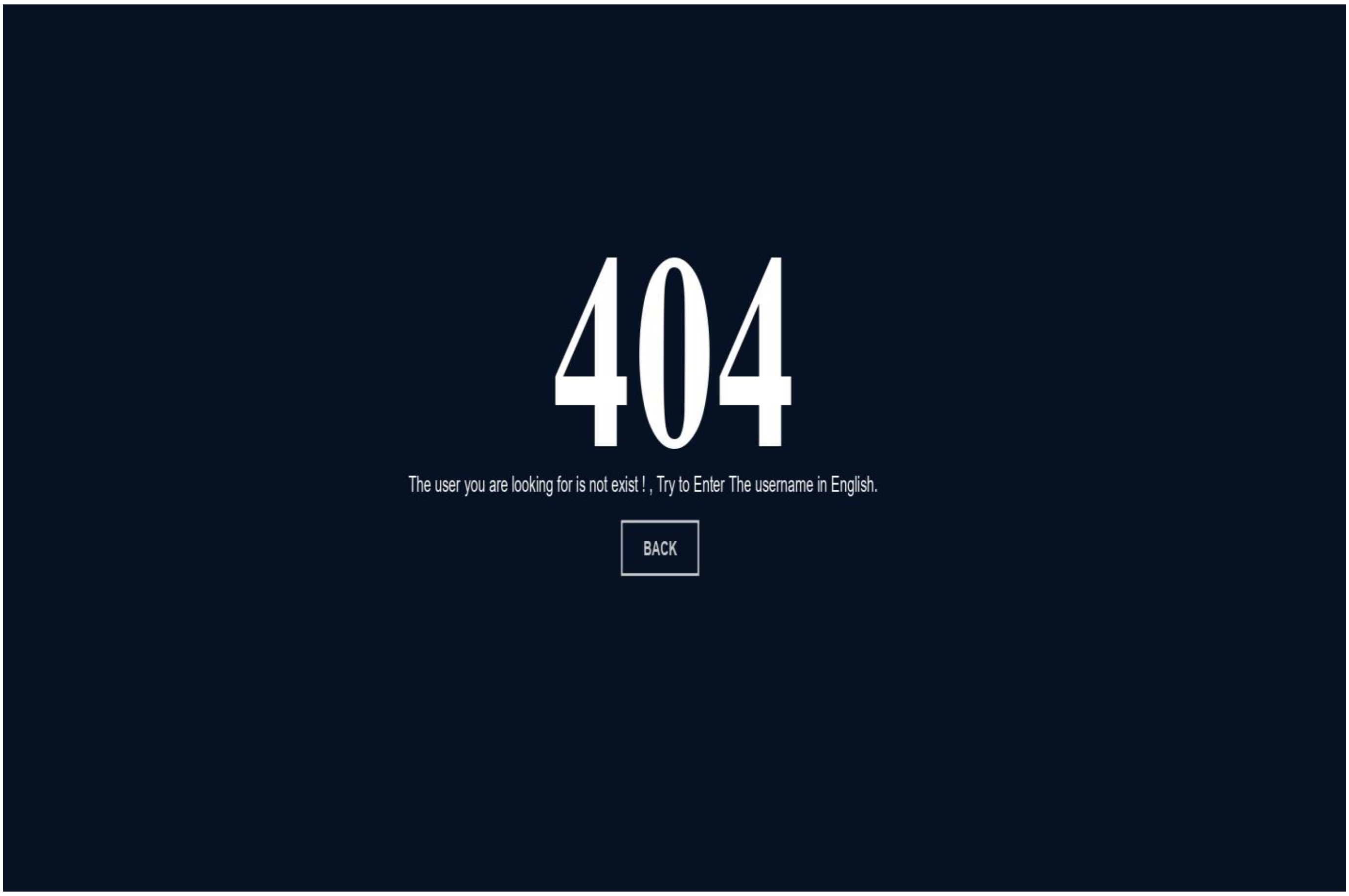
Numbers of Retweets and Mentions

WordCloud

Pie chart for Devices

Figure: 7.2.4 content.html

7.2.4 error.html



404

The user you are looking for is not exist ! , Try to Enter The username in English.

Figure: 7.2.5 error.html

CHAPTER 8

Conclusion

8.1 Summary

We have developed an analytical web application that is characterized by its full support for the Arabic language, and it also makes it easier for users to see the details of the latest trends at the real time, as it provides interactive charts for the user and also provides geographical maps, with this application the user will be able to know the most prominent words Which is circulating in the trends, as it will be known in the form of a word cloud, and it will also be able to know the number of tweets also mentions and the type of devices that have been tweeted from as well as a geographical map of the location of the users if they were allowing that.

8.2 Impact of the project on society

The user will be able to search for content and users on the Twitter platform, in more detail and easily, and he will see charts that provide many analyzes, which helps in understanding more easily.

8.3 Limitations and Future Work

8.3.1 Limitations:

The biggest challenge we faced was to complete the project on time, as a full month of project time was deducted because of the decision issued to submit the study.

Twitter API has a rate limit of 900 requests/15-minutes. This hinders the process of analyzing all trends content.

Combining words with symbols or numbers, and when we perform the process of cleaning the data, these symbols will be deleted and thus the words become meaningless.

8.3.2 Future Work:

For future work we recommend that there be more than one artificial intelligence model to classify the content, whether in the classification of offensive content or the classification of feelings.

8.4 Lessons Learned:

1- We learned how to be full stack developer , we learned the techniques and programming languages use in the front-end and Back-End.

- Front-End like : HTML, CSS, JavaScript.
- Back-End like : Python , flask, APIs.

2- many libraries in python such as : pandas , tweetpy, requests and many, as we mentioned in chapter 5.

3- we learned how to visualize data.

4- How clean and processing and analysis data .

5- we learned how to be testing applications and handle errors.

6- Managing projects.

7- We developed the skill of working in a team, and using popular tools using in this field such as Slack.

References

1- https://www.imindq.com/uses/brainstorming

2- https://www.socialbakers.com/company

3- https://www.datapine.com/about-us

4- https://www.dashboardom.com/

5- https://lucidya.com/2020/03/19/10-social-media-analytics-tools-you-should-be-using-for-your-brand/

6- https://www.guru99.com/what-is-incremental-model-in-sdlc-advantages-disadvantages.html

7- https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-use-case-diagram/

8- https://www.geeksforgeeks.org/unified-modeling-language-uml-sequence-diagrams/

9- https://ozcoasts.org.au/conceptual-diagrams/

10- https://softwareengineering.stackexchange.com/questions/343643/software-architecture-vs-system-architecture-vs-class-diagrams

11- https://www.visual-paradigm.com/guide/data-flow-diagram/what-is-data-flow-diagram/

12- https://www.visual-paradigm.com/guide/uml-unified-modeling-language/uml-class-diagram-tutorial/

13- https://www.udacity.com/blog/2014/12/front-end-vs-back-end-vs-full-stack-web-developers.html

14- https://whatis.techtarget.com/definition/front-end

15- https://www.mapbox.com/pricing/

16- https://www.igi-global.com/dictionary/i-found-myself-retweeting/30754

17- https://pythonbasics.org/what-is-flask-python/

18- https://www.parsehub.com/blog/what-is-web-scraping/

19- https://www.redhat.com/en/topics/api/what-are-application-programming-interfaces

20- https://www.jetbrains.com/help/pycharm/quick-start-guide.html

21- https://www.hostinger.com/tutorials/what-is-html

22- https://www.javatpoint.com/what-is-css

23- tutorialspoint.com/javascript/javascript_overview.htm

24- https://www.howtogeek.com/428046/what-is-slack-and-why-do-people-love-it/

25- https://research.google.com/colaboratory/faq.html

26- https://github.com/plotly/plotly.js/

27- https://air.inc/blog/logos/mapbox-Logo-slack-theme

28- https://www.geeksforgeeks.org/time-series-plot-or-line-plot-with-pandas/

29- https://pypi.org/project/urllib3/

30- https://docs.python.org/3/library/warnings.html

31- https://docs.rs/arabic_reshaper/0.4.0/arabic_reshaper/
32-

Appendices

Keywords

AI: Artificial Intelligence.

DL: Deep Learning.

NLP: Natural language processing.