

Student Feedback Sentiment

Analysis

A Minor Project Report
Submitted To



Chhattisgarh Swami Vivekanand Technical University
Bhilai, India

For
The Award of Degree
of
Bachelor of Technology

zn
Computer Science & Engineering
By

Adhireddy Praveen Kumar

Roll No-303302218102

En. No. - BF4794

Semester 7th (CSE)

Devwrat Singh

Roll N0-303302218010

En. No. – BF4702

Semester 7th (CSE)

Anirudh Gopawar

Roll No-303302218116

En. No. - BF4808

Semester 7th (CSE)

Under the Guidance of

**Dr. Chaitali Biswas
Dutta**

Assistant Professor

Department of Computer Science & Engineering
S.S.I.P.M.T, Raipur



Department of Computer Science & Engineering
Shri Shankaracharya Institute of Professional
Management & Technology
Raipur (C.G.)

Session: 2021 - 2022

DECLARATION BY THE CANDIDATE

We the undersigned solemnly declare that the Major project report entitled "**Student Feedback Sentiment Analysis** " is based our own work carried out during the course of our study under the supervision of **Assistant Prof. Dr. Chaitali Biswas Dutta**

We assert that the statements made and conclusions drawn are an outcome of the project work. We further declare that to the best of our knowledge and belief that the report does not contain any part of any work which has been submitted for the award of any other degree/diploma/certificate in this University/Deemed university of India or any other country.

Adhireddy Praveeen Kumar
Roll No.-303302218102
En. No. - BF4794
Semester 7th (CSE)

Devwrat Singh
Roll No.-303302218010
En. No. - BF4702
Semester 7th (CSE)

**Dr. Chaitali Biswas
Dutta**

Professor, Dept of C.S.E.
S.S.I.P.M.T
Raipur (C.G.)

Anirudh Gopawar
Roll No.-303302218116
En. No. - BF4752
Semester 7th (CSE)

CERTIFICATE BY THE SUPERVISOR

This is to certify that the Major project report entitled "*Student Feedback Sentiment Analysis*" is a record of project work carried out under my guidance and supervision for the fulfillment of the award of degree of Bachelor of Technology in the faculty of Computer Science & Engineering of Chhattisgarh Swami Vivekananda Technical University, Bhilai (C.G.) India.

To the best of my knowledge and belief the report

- i) Embodies the work of the candidate himself
- ii) Has duly been completed
- iii) Fulfills the partial requirement of the ordinance relating to the B.Tech. degree of the University
- iv) Is up to the desired standard both in respect of contents and language for being referred to the examiners.

(Signature of the Supervisor)

Assistant Prof. Dr. Chaitali Biswas Dutta

Professor, Dept of C.S.E.
S.S.I.P.M.T, Raipur (C.G.)

Forwarded to Chhattisgarh Swami Vivekananda Technical University

Bhilai

(Signature of HOD)

Dept. of Computer Science & Engineering
S.S.I.P.M.T
Raipur, C.G

(Signature of the Principal)

Dr. Alok Kumar Jain

S.S.I.P.M.T
Raipur, C.G

CERTIFICATE BY THE EXAMINERS

The project report entitled " Student Feedback Sentiment Analysis "has been examined by the undersigned as a part of the examination of Bachelor of Technology in the faculty of Computer Science & Engineering of Chhattisgarh Swami Vivekanand Technical University, Bhilai.

Internal Examiner

Date:

External Examiner

Date:

ACKNOWLEDGEMENT

Working for this project has been a great experience for us. There were moments of anxiety, when we could not solve a problem for the several days. But we have enjoyed every bit of process and are thankful to all people associated with us during this period we convey our sincere thanks to our project guide **Assistant Prof. Dr. Chaitali Biswas Dutta** for providing me all sorts of facilities. His support and guidance helped us to carry out the project. We owe a great dept. of his gratitude for his constant advice, support, cooperation & encouragement throughout the project we would also like to express our deep gratitude to respected **Dr.JP Patra** (Head of Department) for his ever helping and support. We also pay special thanks for his helpful solution and comments enriched by his experience, which improved our ideas for betterment of the project. We would also like to express our deep gratitude to respected **Dr. Alok Kumar Jain** (Principal) and college management for providing an educational ambience. It will be our pleasure to acknowledge, utmost cooperation and valuable suggestions from time to time given by our staff members of our department, to whom we owe our entire computer knowledge and also we would like to thank all those persons who have directly or indirectly helped us by providing books and computer peripherals and other necessary amenities which helped us in the development of this project which would otherwise have not been possible.

Adhireddy Praveen Kumar
Roll No.-303302218102
En. No. - BF4794
Semester 7th (CSE)

Devwrat Singh
Roll N0-303302218010
En. No. - BF4702
Semester 7th (CSE)

Anirudh Gopawar
Roll No-303302218116
En. No. - BF4808
Semester 7th (CSE)

Abstract

Student Feedback Sentiment Analysis is a tool that helps admin or university to check student feedback for the university and faculty. This tool primary purpose is to identify the problem in the university by the student feedback. This tool performs sentiment analysis to find out the data insights from the student feedback. Student can give their feedback directly on this tool.

This tool will give the quick and accurate result to the admin by performing analysis on feedback data. So that university can figure out the real problem in the system and fix it as soon as possible.

This tool can be used in other business like, to check customer review in restaurants and hotels, only thing need to be change is dataset. To find out customer wants and expectation, this tool can be used to perform sentiment analysis on customer feedback data for the hotel and restaurants and their food.

Technologies Used

- Django
- Python
- HTML
- CSS
- JavaScript
- MySql
- Machine Learning

Table Of Contents

ABSTRACT 1

LIST OF FIGURES v

CHAPTER 1

INTRODUCTION.....5

1.1 Project Plan.....6

1.1.1 About the project.....6

1.1.2 Purpose and scope.....7

1.2 Front End.....9

1.2.1 Introduction Of HTML.....9

1.2.2 Introduction of CSS.....13

1.2.3 Introduction of JavaScript.....16

1.3 Back End.....19

1.3.1 Python.....19

1.3.2 Django.....23

CHAPETR 2

SOFTWARE DEVELOPMENT LIFE CYCLE.....26

2.1 Requirement Analysis Phase.....27

2.1.1 System Requirement Specification.....28

2.1.2 Hardware and Software Requirement.....	29
2.1.3 Functional Requirement	29
2.1.4 Non-Functional Requirement	30
2.1.5 Feasibility Study.....	32
2.1.6 Technical Feasibility	33
2.1.7 Economic Feasibility.....	34
2.1.8 Behavioural Feasibility	35

CHAPETR 3

DIAGRAMS.....	36
1. Data Flow Diagram.....	37
2. E-R Diagram.....	39
3. Component Diagram.....	40
4. Use Case Diagram.....	41

CHAPTER 4

CODING.....	42
4.1 PYTHON Code.....	43
4.2 HTML Code.....	46

CHAPTER 5

TESTING.....58

5.1 Unit Testing.....59

5.2 Integration Testing.....60

5.3 System Testing.....61

5.4 Functional Testing.....62

CHAPETR 6

SNAPSHOTS.....60

CHAPETR 7

FUTURE ENHANCEMENT.....66

REFERENCES.....67

CHAPTER 1

INTRODUCTION

Introduction

1.1 PROJECT PLAN

1.1.1 ABOUT THE PROJECT

This tool is a continuation of the feedback data sentiment analysis tool which is used to find the customer or user sentiments in the feedback.

It is developed for the university to perform analysis on the student feedback so that admin can understand the real problem in the system and students requirements. This tool gives full insights of the feedback data using Machine Learning Algorithm.

The main feature of this tool is-

1. It allows student to give the feedback.
2. It allows university admin to perform analysis on the student feedback dataset
3. Perform analysis and give insights from the data and represent in charts and tables.

Different libraries are used to perform sentiment analysis and data visualization, some of them are -

1. sklearn
2. NLTK
3. Pandas
4. Matplotlib
5. TextBlob

This application represents end result in word cloud, confusion matrix, classification report and pie charts.

This is highly scalable and can be used in different scenarios like to perform analysis hotel and restaurant feedback. It was developed to run on Window, Mac OS X and Linux.

1.1.2 PURPOSE AND SCOPE

It is crucial to understand the patterns generated by student feedback data to effectively improve the performance of the institution. There is also a requirement to automate the student feedback system in order to handle a large amount of data and analyse them effectively.

To achieve this proposed system is created for educational data mining. Educational data mining refers to practices, mechanisms and researches invented for automatically extracting meaning from large repositories of data generated by or related to student's learning activities in educational settings.

There are many techniques like machine learning and sentiment analysis which can be used to accomplish the task of educational data mining.

Sentiment analysis techniques can be applied on the student feedback to extract the opinions and subjectivity from the student feedback.

We have proposed a feedback analysis system based on sentiment analysis and text summarization techniques which

generates a feedback. We have applied various machine learning techniques for classifying student feedback. In order to ensure the correctness of the feedback summary, we chose the technique yielding the maximum classification accuracy, and then with the help of a text summarization technique, we get the more accurate feedback summary.

➤ **Future Scope**

Sentiment analysis is a uniquely powerful tool for businesses that are looking to measure attitudes, feelings and emotions regarding their brand. To date, the majority of sentiment analysis projects have been conducted almost exclusively by companies and brands through the use of social media data, survey responses and other hubs of user-generated content. By investigating and analyzing customer sentiments, these brands are able to get an inside look at consumer behaviors and, ultimately, better serve their audiences with the products, services and experiences they offer.

The future of sentiment analysis is going to continue to dig deeper, far past the surface of the number of likes, comments and shares, and aim to reach, and truly understand, the significance of social media interactions and what they tell us about the consumers behind the screens. This forecast also predicts broader applications for sentiment analysis – brands will continue to leverage this tool, but so will individuals in the public eye, governments, nonprofits, education centers and many other organizations.

1.2 Front End

1.2.1 Introduction of HTML

HTML is an acronym which stands for **Hyper Text Markup Language** which is used for creating web pages and web applications. Let's see what is meant by Hypertext Markup Language, and Web page.

Hyper Text:

HyperText simply means "Text within Text." A text has a link within it, is a hypertext. Whenever you click on a link which brings you to a new webpage, you have clicked on a hypertext. HyperText is a way to link two or more web pages (HTML documents) with each other.

Markup language:

A markup language is a computer language that is used to apply layout and formatting conventions to a text document. Markup language makes text more interactive and dynamic. It can turn text into images, tables, links, etc.

Web Page:

A web page is a document which is commonly written in HTML and translated by a web browser. A web page can be identified by entering an URL. A Web page can be of the static

or dynamic type. **With the help of HTML only, we can create static web pages.**

Hence, HTML is a markup language which is used for creating attractive web pages with the help of styling, and which looks in a nice format on a web browser. An HTML document is made of many HTML tags and each HTML tag contains different content.

Brief History of HTML:

In the late 1980's , a physicist, Tim Berners-Lee who was a contractor at CERN, proposed a system for CERN researchers. In 1989, he wrote a memo proposing an internet based hypertext system.

Tim Berners-Lee is known as the father of HTML. The first available description of HTML was a document called "HTML Tags" proposed by Tim in late 1991. The latest version of HTML is HTML5, which we will learn later in this tutorial.

HTML Versions:

Since the time HTML was invented there are lots of HTML versions in market, the brief introduction about the HTML version is given below:

HTML 1.0: The first version of HTML was 1.0, which was the barebones version of HTML language, and it was released in 1991.

HTML 2.0: This was the next version which was released in 1995, and it was standard language version for website design. HTML 2.0 was able to support extra features such as form-based file upload, form elements such as text box, option button, etc.

HTML 3.2: HTML 3.2 version was published by W3C in early 1997. This version was capable of creating tables and providing support for extra options for form elements. It can also support a web page with complex mathematical equations. It became an official standard for any browser till January 1997. Today it is practically supported by most of the browsers.

HTML 4.01: HTML 4.01 version was released on December 1999, and it is a very stable version of HTML language. This version is the current official standard, and it provides added support for stylesheets (CSS) and scripting ability for various multimedia elements.

HTML5 : HTML5 is the newest version of HyperText Markup language. The first draft of this version was announced in January 2008. There are two major organizations one is W3C (World Wide Web Consortium), and another one is WHATWG (Web Hypertext Application Technology Working

Group) which are involved in the development of HTML 5 version, and still, it is under development.

Features of HTML

- 1) It is a very **easy and simple language**. It can be easily understood and modified.
- 2) It is very easy to make an **effective presentation** with HTML because it has a lot of formatting tags.
- 3) It is a **markup language**, so it provides a flexible way to design web pages along with the text.
- 4) It facilitates programmers to add a **link** on the web pages (by html anchor tag), so it enhances the interest of browsing of the user.
- 5) It is **platform-independent** because it can be displayed on any platform like Windows, Linux, and Macintosh, etc.
- 6) It facilitates the programmer to add **Graphics, Videos, and Sound** to the web pages which makes it more attractive and interactive.
- 7) HTML is a case-insensitive language, which means we can use tags either in lower-case or upper-case.

1.2.2 Introduction of 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. It provides an additional feature to HTML. It is generally used with HTML to change the style of web pages and user interfaces. It can also be used with any kind of XML documents including plain XML, SVG and XUL.

CSS is used along with HTML and JavaScript in most websites to create user interfaces for web applications and user interfaces for many mobile applications.

➤ What does CSS do

You can add new looks to your old HTML documents.

You can completely change the look of your website with only a few changes in CSS code.

➤ Why use CSS

These are the three major benefits of CSS:

1) Solves a big problem

Before CSS, tags like font, color, background style, element alignments, border and size had to be repeated on every web page. This was a very long process. For example: If you are developing a large website where fonts and color information

are added on every single page, it will become a long and expensive process. CSS was created to solve this problem. It was a W3C recommendation.

2) Saves a lot of time

CSS style definitions are saved in external CSS files so it is possible to change the entire website by changing just one file.

3) Provide more attributes

CSS provides more detailed attributes than plain HTML to define the look and feel of the website.

➤ Advantages of CSS

CSS saves time – You can write CSS once and then reuse same sheet in multiple HTML pages. You can define a style for each HTML element and apply it to as many Web pages as you want.

Pages load faster – If you are using CSS, you do not need to write HTML tag attributes every time. Just write one CSS rule of a tag and apply it to all the occurrences of that tag. So less code means faster download times.

Easy maintenance – To make a global change, simply change the style, and all elements in all the web pages will be updated automatically.

Superior styles to HTML – CSS has a much wider array of attributes than HTML, so you can give a far better look to your HTML page in comparison to HTML attributes.

Multiple Device Compatibility – Style sheets allow content to be optimized for more than one type of device. By using the same HTML document, different versions of a website can be presented for handheld devices such as PDAs and cell phones or for printing.

Global web standards – Now HTML attributes are being deprecated and it is being recommended to use CSS. So its a good idea to start using CSS in all the HTML pages to make them compatible to future browsers.

➤ **CSS Versions**

Cascading Style Sheets level 1 (CSS1) came out of W3C as a recommendation in December 1996. This version describes the CSS language as well as a simple visual formatting model for all the HTML tags.

CSS2 became a W3C recommendation in May 1998 and builds on CSS1. This version adds support for media-specific style sheets e.g. printers and aural devices, downloadable fonts, element positioning and tables.

1.2.3 Introduction of JavaScript

JavaScript is a light-weight object-oriented programming language which is used by several websites for scripting the webpages. It is an interpreted, full-fledged programming language that enables dynamic interactivity on websites when applied to an HTML document. It was introduced in the year 1995 for adding programs to the webpages in the Netscape Navigator browser. Since then, it has been adopted by all other graphical web browsers. With JavaScript, users can build modern web applications to interact directly without reloading the page every time. The traditional website uses js to provide several forms of interactivity and simplicity.

Although, JavaScript has no connectivity with Java programming language. The name was suggested and provided in the times when Java was gaining popularity in the market. In addition to web browsers, databases such as CouchDB and MongoDB uses JavaScript as their scripting and query language.

➤ Features of JavaScript

There are following features of JavaScript:

- All popular web browsers support JavaScript as they provide built-in execution environments.
- JavaScript follows the syntax and structure of the C programming language. Thus, it is a structured programming language.

- JavaScript is a weakly typed language, where certain types are implicitly cast (depending on the operation).
- JavaScript is an object-oriented programming language that uses prototypes rather than using classes for inheritance.
- It is a light-weighted and interpreted language.
- It is a case-sensitive language.
- JavaScript is supportable in several operating systems including, Windows, macOS, etc.
- It provides good control to the users over the web browsers.

➤ **History of JavaScript**

In 1993, **Mosaic**, the first popular web browser, came into existence. In the **year 1994**, **Netscape** was founded by **Marc Andreessen**. He realized that the web needed to become more dynamic. Thus, a 'glue language' was believed to be provided to HTML to make web designing easy for designers and part-time programmers. Consequently, in 1995, the company recruited **Brendan Eich** intending to implement and embed Scheme programming language to the browser. But, before Brendan could start, the company merged with **Sun Microsystems** for adding Java into its Navigator so that it could compete with Microsoft over the web technologies and platforms. Now, two languages were there: Java and the scripting language. Further, Netscape decided to give a similar

name to the scripting language as Java's. It led to 'Javascript'. Finally, in May 1995, Marc Andreessen coined the first code of Javascript named '**Mocha**'. Later, the marketing team replaced the name with '**LiveScript**'. But, due to trademark reasons and certain other reasons, in December 1995, the language was finally renamed to 'JavaScript'. From then, JavaScript came into existence.

➤ **Application of JavaScript**

- JavaScript is used to create interactive websites. It is mainly used for:
- Client-side validation,
- Dynamic drop-down menus,
- Displaying date and time,
- Displaying pop-up windows and dialog boxes (like an alert dialog box, confirm dialog box and prompt dialog box),
- Displaying clocks etc.

1.3. Back End

1.3.1 Introduction of Python

Python tutorial provides basic and advanced concepts of Python. Our Python tutorial is designed for beginners and professionals.

Python is a simple, general purpose, high level, and object-oriented programming language.

Python is an interpreted scripting language also. *Guido Van Rossum* is known as the founder of Python programming.

➤ What is Python

Python is a general purpose, dynamic, high-level, and interpreted programming language. It supports Object Oriented programming approach to develop applications. It is simple and easy to learn and provides lots of high-level data structures.

Python is *easy to learn* yet powerful and versatile scripting language, which makes it attractive for Application Development.

Python's syntax and *dynamic typing* with its interpreted nature make it an ideal language for scripting and rapid application development.

Python supports *multiple programming pattern*, including object-oriented, imperative, and functional or procedural programming styles.

Python is not intended to work in a particular area, such as web programming. That is why it is known as *multipurpose* programming language because it can be used with web, enterprise, 3D CAD, etc.

We don't need to use data types to declare variable because it is *dynamically typed* so we can write `a=10` to assign an integer value in an integer variable.

Python makes the development and debugging *fast* because there is no compilation step included in Python development, and edit-test-debug cycle is very fast.

➤ **Python 2 vs. Python 3**

In most of the programming languages, whenever a new version releases, it supports the features and syntax of the existing version of the language, therefore, it is easier for the projects to switch in the newer version. However, in the case of Python, the two versions Python 2 and Python 3 are very much different from each other.

A list of differences between Python 2 and Python 3 are given below:

Python 2 uses **print** as a statement and used as `print "something"` to print some string on the console. On the other hand, Python 3 uses **print** as a function and used as `print("something")` to print something on the console.

Python 2 uses the function `raw_input()` to accept the user's input. It returns the string representing the value, which is typed by the user. To convert it into the integer, we need to use the `int()` function in Python. On the other hand, Python 3 uses `input()` function which automatically interpreted the type of input entered by the user. However, we can cast this value to any type by using primitive functions (`int()`, `str()`, etc.).

In Python 2, the implicit string type is ASCII, whereas, in Python 3, the implicit string type is Unicode.

Python 3 doesn't contain the `xrange()` function of Python 2. The `xrange()` is the variant of `range()` function which returns a `xrange` object that works similar to Java iterator. The `range()` returns a list for example the function `range(0,3)` contains 0, 1, 2.

There is also a small change made in Exception handling in Python 3. It defines a keyword **as** which is necessary to be used. We will discuss it in Exception handling section of Python programming tutorial.

➤ **Python History**

Python was invented by **Guido van Rossum** in 1991 at CWI in Netherland. The idea of Python programming language has

taken from the ABC programming language or we can say that ABC is a predecessor of Python language.

There is also a fact behind the choosing name Python. Guido van Rossum was a fan of the popular BBC comedy show of that time, "**Monty Python's Flying Circus**". So he decided to pick the name **Python** for his newly created programming language.

Python has the vast community across the world and releases its version within the short period.

➤ **Where is Python used?**

Python is a general-purpose, popular programming language and it is used in almost every technical field. The various areas of Python use are given below.

- Data Science
- Data Mining
- Desktop Applications
- Console-based Applications
- Mobile Applications
- Software Development
- Artificial Intelligence
- Web Applications
- Enterprise Applications
- 3D CAD Applications
- Machine Learning

1.3.2 Django

Django is a web application framework written in Python programming language. It is based on MVT (Model View Template) design pattern. The Django is very demanding due to its rapid development feature. It takes less time to build application after collecting client requirement.

This framework uses a famous tag line:

The web framework for perfectionists with deadlines.

By using Django, we can build web applications in very less time. Django is designed in such a manner that it handles much of configure things automatically, so we can focus on application development only.

➤ History

Django was design and developed by Lawrence journal world in 2003 and publicly released under BSD license in July 2005. Currently, DSF (Django Software Foundation) maintains its development and release cycle.

Django was released on 21, July 2005. Its current stable version is 2.0.3 which was released on 6 March, 2018.

➤ Popularity

Django is widely accepted and used by various well-known sites such as:

- Instagram

- Mozilla
- Disqus
- Pinterest
- Bitbucket
- The Washington Times

➤ **Features of Django**

- Rapid Development
- Secure
- Scalable
- Fully loaded
- Versatile
- Open Source
- Vast and Supported Community

➤ **Rapid Development**

Django was designed with the intention to make a framework which takes less time to build web application. The project implementation phase is a very time taken but Django creates it rapidly.

➤ **Secure**

Django takes security seriously and helps developers to avoid many common security mistakes, such as SQL injection, cross-site scripting, cross-site request forgery etc. Its user authentication system provides a secure way to manage user accounts and passwords.

➤ **Scalable**

Django is scalable in nature and has ability to quickly and flexibly switch from small to large scale application project.

➤ **Fully loaded**

Django includes various helping task modules and libraries which can be used to handle common Web development tasks. Django takes care of user authentication, content administration, site maps, RSS feeds etc.

➤ **Versatile**

Django is versatile in nature which allows it to build applications for different-different domains. Now a days, Companies are using Django to build various types of applications like: content management systems, social networks sites or scientific computing platforms etc.

➤ **Open Source**

Django is an open source web application framework. It is publicly available without cost. It can be downloaded with source code from the public repository. Open source reduces the total cost of the application development.

CHAPTER 2

SOFTWARE DEVELOPMENT LIFE CYCLE

2.1 REQUIREMENT ANALYSIS PHASE

The Requirements Analysis Phase begins when the previous phase objectives have been achieved. Documentation related to user requirements from the Concept Development Phase and the Planning Phase shall be used as the basis for further user needs analysis and the development of detailed requirements. Multiple-release projects require only one iteration of the Requirements Analysis Phase, which should involve requirements definition for all planned releases.

The objective of this phase is to define in more detail the system inputs, processes, outputs and interfaces. At the end of this phase the system's processes will be defined at the functional level, meaning the functions to be performed will be known, but not necessarily how they will be performed. Unless specifically constrained by the Project Charter, Requirements Analysis should not consider the computer programs, files and data streams. Requirements Analysis will identify and consider the risks related to how the technology will be integrated into the standard operating procedures. Requirements Analysis will collect the functional and system requirements of the business process, the user requirements and the operational requirements (e.g., when operational what is necessary to keep the system up and running).

2.1.1 System Requirement Specification

Software requirements specification establishes the basis for an agreement between customers and contractors or suppliers on how the software product should function (in a market-driven project, these roles may be played by the marketing and development divisions). Software requirements specification is a rigorous assessment of requirements before the more specific system design stages, and its goal is to reduce later redesign. It should also provide a realistic basis for estimating product costs, risks, and schedules. Used appropriately, software requirements specifications can help prevent software project failure.

The software requirements specification document lists sufficient and necessary requirements for the project development. To derive the requirements, the developer needs to have clear and thorough understanding of the products under development. This is achieved through detailed and continuous communications with the project team and customer throughout the software development process.

Purpose

The purpose of this document is to give a detailed description of the requirements for the “Amazing Lunch

Indicator” (ALI) software. It will illustrate the purpose and complete declaration for the development of system. It will also explain system constraints, interface and interactions with other external applications. This document is primarily intended to be proposed to a customer for its approval and a reference for developing the first version of the system for the development team.

2.1.2 Hardware and Software Requirement

1. Hardware Specification

- RAM 4 GB
- GPU

2. Software Requirements:

- Python
- PyCharm
- Browser to Test

2.1.3 Functional Requirement

In Software engineering and systems engineering, a functional requirement defines a function of a system or its

component. A function is described as a set of inputs, the behaviour, and outputs. Functional requirements may be calculations, technical details, data manipulation and processing and other specific functionality that define what a system is supposed to accomplish. Behavioural requirements describing all the cases where the system uses the functional requirements are captured in use cases. Functional requirements are supported by non-functional requirements which impose constraints on the design or implementation.

As defined in requirements engineering, functional requirements specify particular results of a system. This should be contrasted with non-functional requirements which specify overall characteristics such as cost and reliability. Functional requirements drive the application architecture of a system, while non-functional requirements drive the technical architecture of a system.

2.1.4 Non-Functional Requirement

In systems engineering and requirements engineering, a non-functional requirement is

a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviours. They are contrasted with functional

requirements that define specific behaviour or functions.

The plan for implementing functional requirements is detailed in the system design. The plan for implementing non-functional requirements is detailed in the system architecture, because they are usually Architecturally Significant Requirements.

Broadly, functional requirements define what a system is supposed to do and non-functional requirements define how a system is supposed to be. Functional requirements are usually in the form of, an individual action or part of the system, perhaps explicitly in the sense of a mathematical function, a black box description input, output, process and control functional model or IPO Model. In contrast, non-functional requirements are in the form of, an overall property of the system as a whole or of a particular aspect and not a specific function. The system's overall properties commonly mark the difference between whether the development project has succeeded or failed.

Non-functional requirements are often called "quality attributes" of a system. Other terms for non-functional

requirements are "qualities", "quality goals", "quality of service requirements", "constraints" and "non-behavioural requirements". Informally these are sometimes called the "ilities", from attributes like stability and portability. Qualities—that is non-functional requirements—can be divided into two main categories:

1. Execution qualities, such as safety, security and usability, which are observable during operation.
2. Evolution qualities, such as testability, maintainability, extensibility and scalability, which are embodied in the static structure of the system.

2.1.5 Feasibility Study

Feasibility study is made to see if the project on completion will serve the purpose of the organization for the amount of work, effort and the time that spend on it. Feasibility study lets the developer foresee the future of the project and the usefulness.

A feasibility study of a system proposal is according to its

workability, which is the impact on the organization, ability to meet their user needs and effective use of resources. Thus when a new application is proposed it normally goes through a feasibility study before it is approved for development.

Following are its feature

2.1.6 TECHNICAL FEASIBILITY

The system must be evaluated from the technical point of view first. The assessment of this feasibility must be based on an outline design of the system requirement in the terms of input, output, programs and procedures.

Having identified an outline system, the investigation must go on to suggest the type of equipment, required method developing the system, of running the system once it has been designed.

Technical issues raised during the investigation are:

- ☐ Does the existing technology sufficient for the suggested one?

- Can the system expand if developed?

2.1.7 ECONOMIC FEASIBILITY

The developing system must be justified by cost and benefit. Criteria to ensure that effort is concentrated on project, which will give best, return at the earliest.

One of the factors, which affect the development of a new system, is the cost it would require.

The following are some of the important financial questions asked during preliminary investigation:

- The costs conduct a full system investigation.
- The cost of the hardware and software.
- The benefits in the form of reduced costs or fewer costly errors.

Since the system is developed as part of project work, there is no manual cost to spend for the proposed system. Also all the resources are already available, it give an indication of the system is economically possible for development

2.1.8 BEHAVIORAL FEASIBILITY

This includes the following questions:

- Is there sufficient support for the users?
- Will the proposed system cause harm?

The project would be beneficial because it satisfies the objectives when developed and installed. All behavioural aspects are considered carefully and conclude that the project is behaviourally feasible.

CHAPTER 3

DIAGRAMS

Implementation or Architecture Diagrams

1. Data Flow Diagram (DFD)

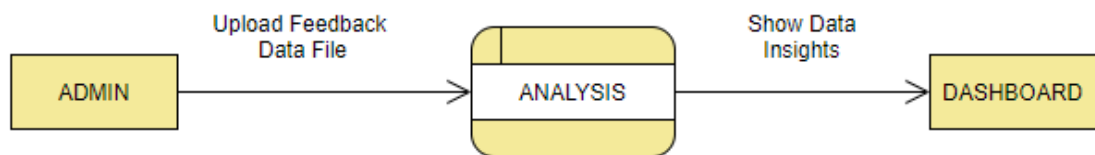
A data flow diagram is a graphical representation that depicts information flow and the transforms that are applied as data move from input to output. The basic form of a data flow diagram, also known as a data flow graph or a bubble chart, The data flow diagram may be used to represent a system or software at any level of abstraction. As information moves through software, it is modified by a series of transformations.

A data flow diagram is a graphical representation that depicts information flow and the transforms that are applied as data move from input to output. The basic form of a data flow diagram, also known as a data flow graph or a bubble chart. DFD is an abstract description of the system. The data flow diagram may be used to represent a system or software at any level of abstraction. DFDs may be partitioned into levels that represent increasing information flow and functional detail. Therefore, the DFD provides a mechanism for functional modeling as well as information flow modeling.

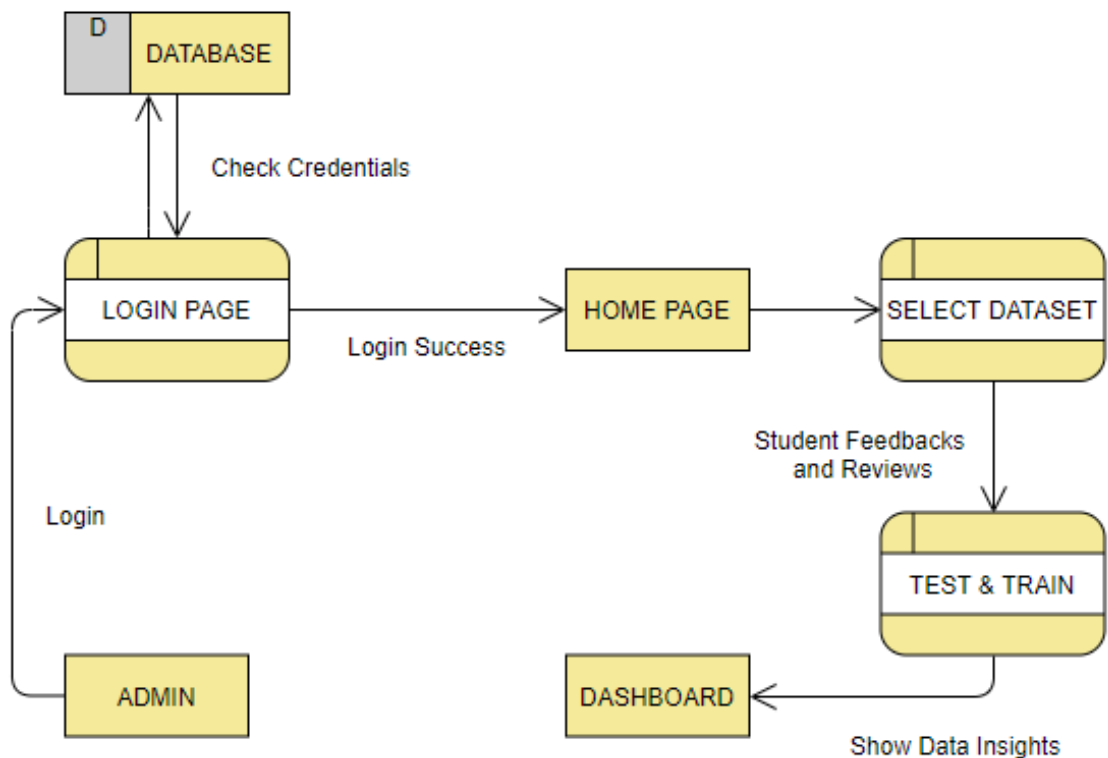
DFDs are very useful in understanding a system and can be effectively used during analysis. DFDs can be

hierarchically organized, which helps in progressively partitioning and analyzing large systems. Such DFDs are called leveled DFDs.

Level-0 DFD



Level-1 DFD

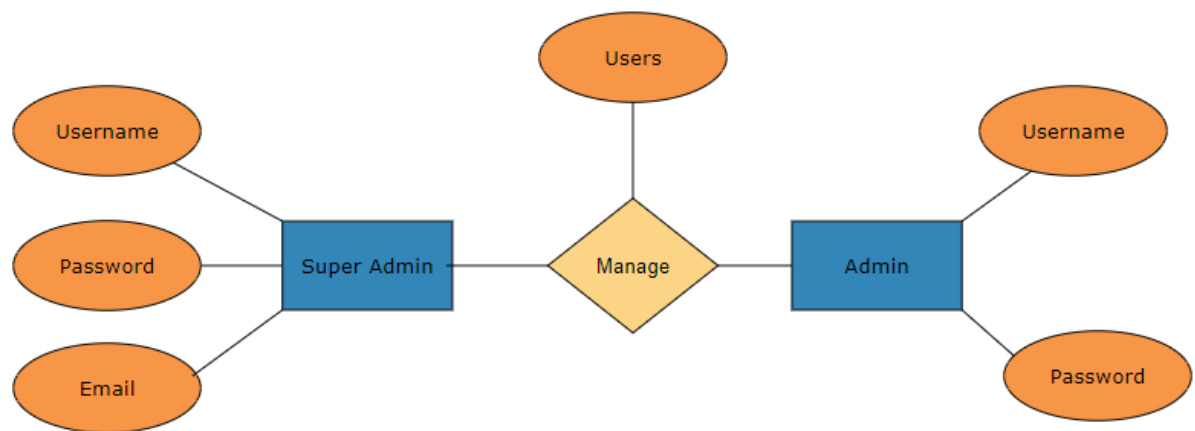


2. ER DIAGRAM

An entity–relationship model (ER model) describes interrelated things of interest in a specific domain of knowledge. A basic ER model is composed of entity types (which classify the things of interest) and specifies relationships that can exist between instances of those entity types.

In software engineering, an ER model is commonly formed to represent things that a business needs to remember in order to perform business processes. Consequently, the ER model becomes an abstract data model, that defines a data or information structure which can be implemented in a database, typically a relational database.

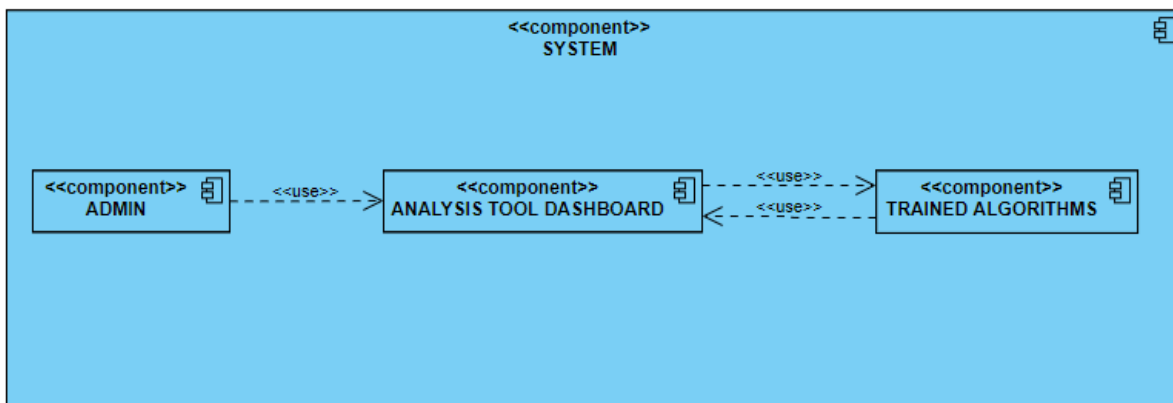
Entity–relationship modeling was developed for database design by Peter Chen and published in a 1976 paper. However, variants of the idea existed previously. Some ER models show super and subtype entities connected by generalization-specialization relationships, and an ER model can be used also in the specification of domain-specific ontologies



3. Component Diagram

A component diagram, also known as a UML component diagram, describes the organization and wiring of the physical components in a system. Component diagrams are often drawn to help model implementation details and double-check that every aspect of the system's required functions is covered by planned development.

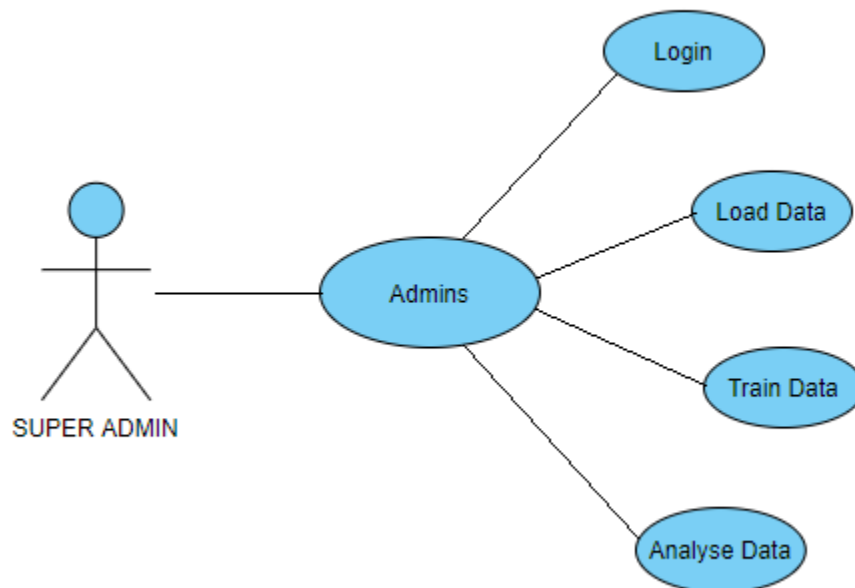
A component diagram allows verification that a system's required functionality is acceptable. These diagrams are also used as a communication tool between the developer and stakeholders of the system. Programmers and developers use the diagrams to formalize a roadmap for the implementation, allowing for better decision-making about task assignment or needed skill improvements. System administrators can use component diagrams to plan ahead, using the view of the logical software components and their relationships on the system.



4. Use Case Diagram

To model a system, the most important aspect is to capture the dynamic behavior. Dynamic behavior means the behavior of the system when it is running/operating.

Only static behavior is not sufficient to model a system rather dynamic behavior is more important than static behavior. In UML, there are five diagrams available to model the dynamic nature and use case diagram is one of them. Now as we have to discuss that the use case diagram is dynamic in nature, there should be some internal or external factors for making the interaction. These internal and external agents are known as actors. Use case diagrams consists of actors, use cases and their relationships. The diagram is used to model the system/subsystem of an application. A single use case diagram captures a particular functionality of a system. Hence to model the entire system, a number of use case diagrams are used.



CHAPTER 4

CODING

1. Python Code for Sentiment Analysis (Controller)

```
from django.contrib.auth import authenticate, login
from django.shortcuts import render, redirect
from sklearn.feature_extraction.text import CountVectorizer
from nltk.tokenize import RegexpTokenizer
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import MultinomialNB
from sklearn.metrics import classification_report, confusion_matrix
import pandas as pd

from textblob import TextBlob

import seaborn as sn
from wordcloud import WordCloud

import matplotlib.pyplot as plt
from io import StringIO
import numpy as np
import pickle

from openpyxl import Workbook
import openpyxl

def index(request):
    if request.method == 'POST':

        if 'shorts' in request.POST:
            return redirect('updateshorts')

        username = request.POST.get('username')
        password = request.POST.get('password')

        user = authenticate(request, username=username, password=password)

        if user is not None:
            login(request, user)
            return redirect('home')
        else:
            print("wrong credentials.")

    return render(request=request,
                  template_name='index.html')

def home(request):

    if request.method == 'POST':

        path = request.POST.get('path')
        data = pd.read_excel(path)

        enum_list = list(zip(data['feedback'], data['points']))
```



```

        pickle_out = open("dict.pickle", "wb")
        pickledata = {'enum_list':enum_list,
                      'path':path}
        pickle.dump(pickledata, pickle_out)
        pickle_out.close()

        return redirect('ShowTableData')

    return render(request=request,
                  template_name='home.html')

def dashboard(request):
    return render(request=request,
                  template_name='dashboard.html')

def ShowTableData(request):
    if request.method == 'POST':

        pickle_in = open("dict.pickle", "rb")
        pickledata = pickle.load(pickle_in)
        path = pickledata.get('path')
        data = pd.read_excel(path)

        token = RegexpTokenizer(r'[a-zA-Z0-9]+')
        cv = CountVectorizer(lowercase=True, stop_words='english',
ngram_range=(1, 1), tokenizer=token.tokenize)
        text_counts = cv.fit_transform(data['feedback'])

        X_train, X_test, y_train, y_test = train_test_split(text_counts,
data['points'], test_size=0.3, random_state=1)

        clf = MultinomialNB().fit(X_train, y_train)

        predicted = clf.predict(X_test)

        confusionMatrix = confusion_matrix(y_test, predicted)
        classificationReport = classification_report(y_test,
predicted,output_dict=True)

        negativeReview = classificationReport['-1']
        neutralReview = classificationReport['0']
        positiveReview = classificationReport['1']

        accuracy = classificationReport['accuracy']
        macro_avg = classificationReport['macro avg']
        weighted_avg = classificationReport['weighted avg']

    def return_graph(matrix):

        df_cm = pd.DataFrame(matrix, index=[i for i in "ABC"],
                              columns=[i for i in "ABC"])

        fig = plt.figure(figsize=(5, 5))

        sn.heatmap(df_cm, annot=True)
        plt.xlabel("Predicted")
        plt.ylabel('Actual')

```

```

        imgdata = StringIO()
        fig.savefig(imgdata, format='svg')
        imgdata.seek(0)

    return imgdata.getvalue()

def show_wordcloud(data, title=None):
    wordcloud = WordCloud(
        background_color='white',
        max_words=1000,
        max_font_size=72,
        scale=5,
        random_state=42
    ).generate(str(data))

    cloud_fig = plt.figure(figsize=(5, 5))
    plt.axis('off')
    plt.imshow(wordcloud)

    imagedata2 = StringIO()
    cloud_fig.savefig(imagedata2, format='svg')
    imagedata2.seek(0)

    return imagedata2.getvalue()

def show_piechart(data):
    positive_review = 0
    negative_review = 0
    neutral_review = 0

    for i in data['points']:
        if i == 1:
            positive_review += 1

        if i == -1:
            negative_review += 1

        if i == 0:
            neutral_review += 1

    y = np.array([positive_review, negative_review, neutral_review])
    mylabels = ["Positive Reviews", "Negative Reviews", "Neutral
Reviews"]
    myexplode = [0.1, 0, 0]

    pie_fig = plt.figure(figsize=(5, 5))
    plt.pie(y, labels=mylabels, autopct='%1.1f%%', explode=myexplode,
shadow=True)

    imagedata3 = StringIO()
    pie_fig.savefig(imagedata3, format='svg')
    imagedata3.seek(0)

    return imagedata3.getvalue()

```

```

        context = {'graph': return_graph(confusionMatrix),
                    'wordcloud': show_wordcloud(data['feedback']),
                    'piechart': show_piechart(data),
                    'classificationReport': classificationReport,
                    'negativeReviewPrecision': negativeReview["precision"],
                    'negativeReviewRecall': negativeReview["recall"],
                    'negativeReviewF1Score': negativeReview["f1-score"],
                    'negativeReviewSupport': negativeReview["support"],

                    'neutralReviewPrecision': neutralReview["precision"],
                    'neutralReviewRecall':
"{:.2f}".format(neutralReview["recall"]),
                    'neutralReviewF1Score': neutralReview["f1-score"],
                    'neutralReviewSupport': neutralReview["support"],

                    'positiveReviewPrecision':
"{:.2f}".format(positiveReview["precision"]),
                    'positiveReviewRecall': positiveReview["recall"],
                    'positiveReviewF1Score':
"{:.2f}".format(positiveReview["f1-score"]),
                    'positiveReviewSupport': positiveReview["support"],
                    'accuracy': "{:.2f}".format(accuracy),

                    'macro_avg_precision': "{:.2f}".format(macro_avg["precision"]),
                    'macro_avg_recall': "{:.2f}".format(macro_avg["recall"]),
                    'macro_avg_F1Score': "{:.2f}".format(macro_avg["f1-
score"]),
                    'macro_avg_Support': macro_avg["support"],

                    'weighted_avg_precision': "{:.2f}".format(weighted_avg["precision"]),
                    'weighted_avg_recall':
"{:.2f}".format(weighted_avg["recall"]),
                    'weighted_avg_F1Score': "{:.2f}".format(weighted_avg["f1-
score"]),
                    'weighted_avg_Support': weighted_avg["support"]}]

    return render(request=request,
                  template_name='dashboard.html',
                  context=context)

    pickle_in = open("dict.pickle", "rb")
    pickledata = pickle.load(pickle_in)
    enum_list = pickledata.get('enum_list')
    print(pickledata)
    return render(request=request,
                  template_name='showtabledata.html',
                  context={'enum_list': enum_list,
                          'enum_list_len': len(enum_list)})

def studentfeedback(request):
    if request.method == 'POST':
        feedback_data = request.POST.get('studentfeedback')
        data = TextBlob(feedback_data)
        points = data.sentiment.polarity

```

```

    if points > 0.5:
        print("Positive ",1)
        final_points=1
    elif points<0.5 and points>0:
        print("Average ",0)
        final_points=0
    else:
        print("Negative ",-1)
        final_points=-1

    print(feedback_data)

    #

    path = "studentfeedback.xlsx"
    wb_obj = openpyxl.load_workbook(path)
    sheet_obj = wb_obj.active

    max_row = sheet_obj.max_row

    data = []
    for i in range(1, max_row + 1):
        cell_obj1 = sheet_obj.cell(row=i, column=1)
        cell_obj2 = sheet_obj.cell(row=i, column=2)

        data.append([cell_obj1.value, cell_obj2.value])

    print(data)

    book = Workbook()
    sheet = book.active

    data.append([final_points, str(feedback_data)])

    for row in data:
        sheet.append(row)

    book.save('studentfeedback.xlsx')

    return redirect('index')
return render(request=request,
              template_name='studentfeedback.html')

```

2. HTML Index Page Code

```

<!doctype html>
<html lang="en">
<head>
<meta charset="utf-8">

```

```

    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.0/dist/css/bootstrap.min.
css" rel="stylesheet" integrity="sha384-
KYZXEAg3QhqlMpG8r+8fhAXLRk2vvoC2f3B09zVXn8CA5QIVfZOJ3BCsw2P0p/We"
crossorigin="anonymous">

    <title>Home</title>
    {% load static %}
</head>

<nav class="navbar navbar-dark bg-dark justify-content-center">
    <div class="container">
        <a class="navbar-brand" href="">
            
            Sentiment Analyser
        </a>
    </div>
</nav>

    {% load static %}
    <body background="{% static 'myapp/back.jpg' %}" style="background-size:
cover;background-repeat: no-repeat;background-attachment: fixed;">
        <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.0/dist/js/bootstrap.bundle
.min.js" integrity="sha384-
U1DAWAZnBHeqE1lVSCgzq+c9gqGAJn5c/t99JyeKa9xxaYpSvHU5awsuZVVFIhvj"
crossorigin="anonymous"></script>

        <section class="vh-100 gradient-custom">
            <div class="container py-4 h-100">
                <div class="row d-flex justify-content-center align-items-center
h-100">
                    <div class="col-12 col-md-8 col-lg-6 col-xl-5">
                        <div class="card bg-dark text-white" style="border-radius:
1rem;">
                            <div class="card-body p-3 text-center">
                                <div class="mb-md-5 mt-md-4 pb-5">
                                    <h2 class="fw-bold mb-1 text-uppercase">Login</h2>
                                    <p class="text-white-50 mb-5">Please enter your login
and password!</p>
                                    <form name="loginform" method="POST">
                                        {% csrf_token %}
                                        <div class="form-outline form-white mb-4">
                                            <input type="text" name="username" class="form-control
form-control-lg" placeholder="username" required/>
                                        </div>
                                        <div class="form-outline form-white mb-4">
                                            <input type="password" name="password" class="form-
control form-control-lg" placeholder="Password" required/>
                                        </div>

```

```

        <button class="btn btn-outline-light btn-lg px-5"
type="submit">Login</button>

        </div>
    </div>
</div>
</div>
</div>
</div>
</div>
</section>

</body>
</html>

```

3. HTML Home Page Code

```

<!doctype html>
<html lang="en">
    <head>
        <meta charset="utf-8">
        <meta name="viewport" content="width=device-width, initial-scale=1">
        <link
href="https://cdn.jsdelivrivr.net/npm/bootstrap@5.1.0/dist/css/bootstrap.min.
css" rel="stylesheet" integrity="sha384-
KyZXEAg3QhqLMpG8r+8fhAXLRk2vvoC2f3B09zVXn8CA5QIVfZ0J3BCsw2P0p/We"
crossorigin="anonymous">

        <title>Home</title>
        {% load static %}
    </head>

    <nav class="navbar navbar-dark bg-dark justify-content-center">
        <div class="container">
            <a class="navbar-brand" href="">
                
                Sentiment Analyser
            </a>
        </div>
    </nav>
    {% load static %}
    <body background="{% static 'myapp/back.jpg' %}" style="background-size:
cover;background-repeat: no-repeat;background-attachment: fixed;">
        <script
src="https://cdn.jsdelivrivr.net/npm/bootstrap@5.1.0/dist/js/bootstrap.bundle
.min.js" integrity="sha384-
U1DAWAznBHeqEILVSCgzq+c9gqGAJn5c/t99JyeKa9xxaYpSvHU5awsuZVVFIhvj"
crossorigin="anonymous"></script>

        <section class="vh-100 gradient-custom">
            <div class="container py-5 h-90">
                <div class="row d-flex justify-content-center align-items-center
h-100">
                    <div class="col-12 col-md-8 col-lg-6 col-xl-8">

```

```

        <div class="card bg-dark text-white" style="border-radius:
1rem;">
            <div class="card-body p-3 text-center">
                <div class="mb-md-5 mt-md-4 pb-4">
                    
                    <h2 class="mb-2">Upload Student Feedback File
here</h2><br>
                    <form name="loginform" method="POST">
                        {% csrf_token %}
                        <div class="form-outline form-white mb-4">
                            <input type="text" name="path" class="form-control
form-control-lg" placeholder="Enter feedback file path.." required/>
                        </div>
                        <button class="btn btn-outline-light btn-lg px-5"
type="submit">Submit</button>
                    </div>
                </div>
            </div>
        </div>
    </div>
</section>

</body>
</html>

```

4. HTML Table Page Code

```

<!doctype html>
<html lang="en">
    <head>
        <meta charset="utf-8">
        <meta name="viewport" content="width=device-width, initial-scale=1">
        <link
href="https://cdn.jsdelivrivr.net/npm/bootstrap@5.1.0/dist/css/bootstrap.min.
css" rel="stylesheet" integrity="sha384-
KyZXEAg3QhqLMpG8r+8fhAXLRk2vvoC2f3B09zVXn8CA5QIVfZ0J3BCsw2P0p/We"
crossorigin="anonymous">

        <title>Dashboard</title>
        {% load static %}
    </head>
    <nav class="navbar navbar-dark bg-dark justify-content-center">
        <div class="container">
            <a class="navbar-brand" href="">
                
                Sentiment Analyser

```

```

    </a>
  </div>
  </nav>
  {% load static %}
  <body>
    <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.0/dist/js/bootstrap.bundle
.min.js" integrity="sha384-
U1DAWAznBHeqEILVSCgzq+c9gqGAJn5c/t99JyeKa9xxaYpSvHU5awsuZVVFIhvj"
crossorigin="anonymous"></script>

    <div style="text-align:center;margin-top:50px;">
      <form method="POST">
        {% csrf_token %}
        <br><br>
        <button class="btn btn-dark btn-lg px-5" type="submit">Test &
Train</button>
      </form>
    </div>
    <div style="padding:50px;">

      <p class="navbar-brand">
        
        
        
        
        {{enum_list_len}}
      </p>

      <table class="table table-dark table-striped">
      <thead>
        <tr>
          <th scope="col">Feedback</th>
          <th scope="col">Point</th>
        </tr>
      </thead>
      <tbody>

        {% for feedback,points in enum_list %}
        <tr>
          <td>{{ feedback }}</td>
          <td>{{ points }}</td>

        </tr>
        {% endfor %}

      </tbody>
    </table>
  </div>

```



```
</body>
</html>
```

5. HTML Dashboard Design Code

```
<!doctype html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.0/dist/css/bootstrap.min.
css" rel="stylesheet" integrity="sha384-
KyZXEAg3QhqLMpG8r+8fhAXLRk2vvoC2f3B09zVXn8CA5QIVfZ0J3BCsw2P0p/We"
crossorigin="anonymous">

    <title>Dashboard</title>
  </head>
  {% load static %}
  <body>
    <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.0/dist/js/bootstrap.bundle
.min.js" integrity="sha384-
U1DAWAZnBHeqE1lVSCgzq+c9gqGAJn5c/t99JyeKa9xxaYpSvHU5awsuZVVFIhvj"
crossorigin="anonymous"></script>

    <nav class="navbar navbar-dark bg-dark justify-content-center">
      <div class="container">
        <a class="navbar-brand" href="">
          
          Sentiment Analyser
        </a>
      </div>
    </nav>

    <div style="text-align:center;margin-top:50px;"></div><br>
    <div content="column">
      <p style="text-align:center;font-size:30px;">
        Data Insights
        <p>
      </p>
    </div>
    <div class="row row-cols-1 row-cols-md-2 g-4 text-center"
style="padding:50px;">
      <div class="col">
        <div class="card shadow-lg p-3 mb-5 bg-white rounded">
          <div class="card-body">
            <h2 class="card-title">Confusion Matrix</h2>
            <p class="card-text">{{ graph|safe }}</p>
          </div>
        </div>
      </div>
```

```

</div>
<div class="col">
  <div class="card shadow-lg p-3 mb-5 bg-white rounded">
    <div class="card-body">
      <h3 class="card-title">Most Common Words Used in Feedback</h3>
      <p class="card-text">{{ wordcloud|safe }}</p>
    </div>
  </div>
</div>
<div class="col">
  <div class="card shadow-lg p-3 mb-5 bg-white rounded">
    <div class="card-body">
      <h2 class="card-title">Feedback Percentage</h2>
      <p class="card-text">{{ piechart|safe }}</p>
    </div>
  </div>
</div>
<div class="col">
  <div class="card shadow-lg p-3 mb-5 bg-white rounded">
    <div class="card-body">
      <h2 class="card-title">Classification Report</h2>
      <p class="card-text">
        <br><br><br>

        <table class="table table-dark table-striped">
          <thead>
            <tr>
              <th scope="col"></th>
              <th scope="col">Precision</th>
              <th scope="col">Recall</th>
              <th scope="col">F1-Score</th>
              <th scope="col">Support</th>
            </tr>
          </thead>
          <tbody>
            <tr>
              <th scope="row">-1</th>
              <td>{{negativeReviewPrecision}}</td>
              <td>{{negativeReviewRecall}}</td>
              <td>{{negativeReviewF1Score}}</td>
              <td>{{negativeReviewSupport}}</td>
            </tr>
            <tr>
              <th scope="row">0</th>
              <td>{{neutralReviewPrecision}}</td>
              <td>{{neutralReviewRecall}}</td>
              <td>{{neutralReviewF1Score}}</td>
              <td>{{neutralReviewSupport}}</td>
            </tr>
            <tr>
              <th scope="row">1</th>
              <td>{{positiveReviewPrecision}}</td>
              <td>{{positiveReviewRecall}}</td>
              <td>{{positiveReviewF1Score}}</td>
              <td>{{positiveReviewSupport}}</td>
            </tr>
          </tbody>
        </table>
      </p>
    </div>
  </div>
</div>

```

```

        <th scope="row">Accuracy</th>
        <td></td>
        <td></td>
        <td>{{accuracy}}</td>
        <td></td>
    </tr>
    <tr>
        <th scope="row">Macro Avg</th>
        <td>{{macro_avg_precision}}</td>
        <td>{{macro_avg_recall}}</td>
        <td>{{macro_avg_F1Score}}</td>
        <td>{{macro_avg_Support}}</td>
    </tr>
    <tr>
        <th scope="row">Weighted Avg</th>
        <td>{{weighted_avg_precision}}</td>
        <td>{{weighted_avg_recall}}</td>
        <td>{{weighted_avg_F1Score}}</td>
        <td>{{weighted_avg_Support}}</td>
    </tr>
</tbody>
</table>

    <br><br><br>

</p>
</div>
</div>
</div>
</div>

</body>
</html>

```

6. HTML Student Feedback Page Code

```

7. <!doctype html>
    <html lang="en">
        <head>
            <meta charset="utf-8">
            <meta name="viewport" content="width=device-width, initial-
scale=1">
            <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.0/dist/css/bootstrap.m
in.css" rel="stylesheet" integrity="sha384-
KyZXEAg3QhqLMpG8r+8fhAXLRk2vvoC2f3B09zVXn8CA5QIVfZ0J3BCsw2P0p/We"
crossorigin="anonymous">

            <title>Student Feedback</title>
            <script type = "text/javascript">
                function msgprint() {

                    alert("Thanks for Your Feedback");

                }

```

```

</script>
{% load static %}
</head>

{% load static %}
<body background="{% static 'myapp/back.jpg' %}" style="background-
size: cover;background-repeat: no-repeat;background-attachment:
fixed;">
    <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.0/dist/js/bootstrap.bun
dle.min.js" integrity="sha384-
U1DAWAznBHeqEILVSCgzq+c9gqGAJn5c/t99JyeKa9xxaYpSvHU5awsuZVVFIhvj"
crossorigin="anonymous"></script>

<br>
    <section class="vh-100 gradient-custom">
        <div class="container py-4 h-100">
            <div class="row d-flex justify-content-center align-items-
center h-100">
                <div class="col-12 col-md-8 col-lg-6 col-xl-7">
                    <div class="card bg-dark text-white" style="border-radius:
1rem;">
                        <div class="card-body p-3 text-center">
                            <div class="mb-md-5 mt-md-4 pb-5">
                                <h2 class="fw-bold mb-1 text-uppercase">Give your
feedback here</h2>
                                <p class="text-white-50 mb-5">Your valuable feedback
will help us to improve !</p>
                                <form name="loginform" method="POST">
                                    {% csrf_token %}
                                    <div class="form-outline form-white mb-4">
                                        <textarea class="form-control" placeholder="Leave a
comment here" id="feedbackinput" name="studentfeedback" style="height:
100px" required></textarea>
                                    </div>
                                    <button class="btn btn-outline-light btn-lg px-5"
type="submit" onclick = "msgprint()" >Submit</button>
                                </form>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </body>
</html>

```


CHAPTER 5

TESTING

Testing Phase

Testing refers to test the software so it is also called software testing. **Software testing** is an investigation conducted to provide stakeholders with information about the quality of the software product or service under test.^[1] Software testing can also provide an objective, independent view of the software to allow the business to appreciate and understand the risks of software implementation. Test techniques include the process of executing a program or application with the intent of finding software bugs (errors or other defects), and verifying that the software product is fit for use.

Software testing involves the execution of a software component or system component to evaluate one or more properties of interest. In general, these properties indicate the extent to which the component or system under test-

- meets the requirements that guided its design and development,
- responds correctly to all kinds of inputs,
- performs its functions within an acceptable time
- is sufficiently usable
- can be installed and run in its intended environments, and
- achieves the general result its stakeholders desire.

5.1 Unit Testing

In computer programming, unit testing is a software testing method by which individual units of source code, sets of one or more computer program modules together with associated control data, usage procedures, and operating procedures, are tested to determine whether they are fit for use.

Parameterized unit tests (PUTs) are tests that take parameters. Unlike traditional unit tests, which are usually closed methods, PUTs take any set of parameters. PUTs have been supported by Testing, JUnit and various .NET test frameworks.

Suitable parameters for the unit tests may be supplied manually or in some cases are automatically generated by the test framework. Testing tools like Quick Check exist to generate test inputs for PUTs.

The goal of unit testing is to isolate each part of the program and show that the individual parts are correct. A unit test provides a strict, written contract that the piece of code must satisfy. As a result, it affords several benefits.

In our tool we tested each and every unit (or module) and it was successfully executed.

We perform sentiment analysis by our tool and its accuracy is around 80% and which is up to the expectation.

5.2 Integration Testing

Integration testing (sometimes called integration and testing, abbreviated I&T) is the phase in software testing in which individual software modules are combined and tested as a group. It occurs after unit testing and before validation testing.

Integration testing takes as its input modules that have been unit tested, groups them in larger aggregates, applies tests defined in an integration test plan to those aggregates, and delivers as its output the integrated system ready for system testing.

Some different types of integration testing are big-bang, mixed (sandwich), risky- hardest, top-down, and bottom-up. Other Integration Patterns are: collaboration integration, backbone integration, layer integration, client-server integration, distributed services integration and high-frequency integration.

In the big-bang approach, most of the developed modules are coupled together to form a complete software system or major part of the system and then used for integration testing. This method is very effective for saving time in the integration testing process.

However, if the test cases and their results are not recorded properly, the entire integration process will be more complicated and may prevent the testing team from achieving the goal of integration testing.

In Sentiment Analyser we tested all components by merge together and it passed all our criteria.

5.3 System Testing

System testing of software or hardware is testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements. System testing falls within the scope of black-box testing, and as such, should require no knowledge of the inner design of the code or logic.

As a rule, system testing takes, as its input, all of the "integrated" software components that have passed integration testing and also the software system itself integrated with any applicable hardware system.

The purpose of integration testing is to detect any inconsistencies between the software units that are integrated together or between any of the assemblages and the hardware. System testing is a more limited type of testing; it seeks to detect defects both within the "inter-assemblages" and also within the system as a whole.

System testing is performed on the entire system in the context of a Functional Requirement Specification and/or a System Requirement Specification (SRS). System testing tests not only the design, but also the behaviour and even the believed expectations of the customer. It is also intended to test up to and beyond the bounds defined in the software/hardware requirements specification.

After performing system testing with our Sentiment analysis tool we find that it follow all requirements and working as per requirement, it takes the input as student

feedback to process data and give the proper insights of feedback data.

5.4 Functional Testing

Functional testing is a quality assurance (QA) process and a type of black-box testing that bases its test cases on the specifications of the software component under test.

Functions are tested by feeding them input and examining the output, and internal program structure is rarely considered (unlike white-box testing). Functional testing is conducted to evaluate the compliance of a system or component with specified functional requirements. Functional testing usually describes what the system does.

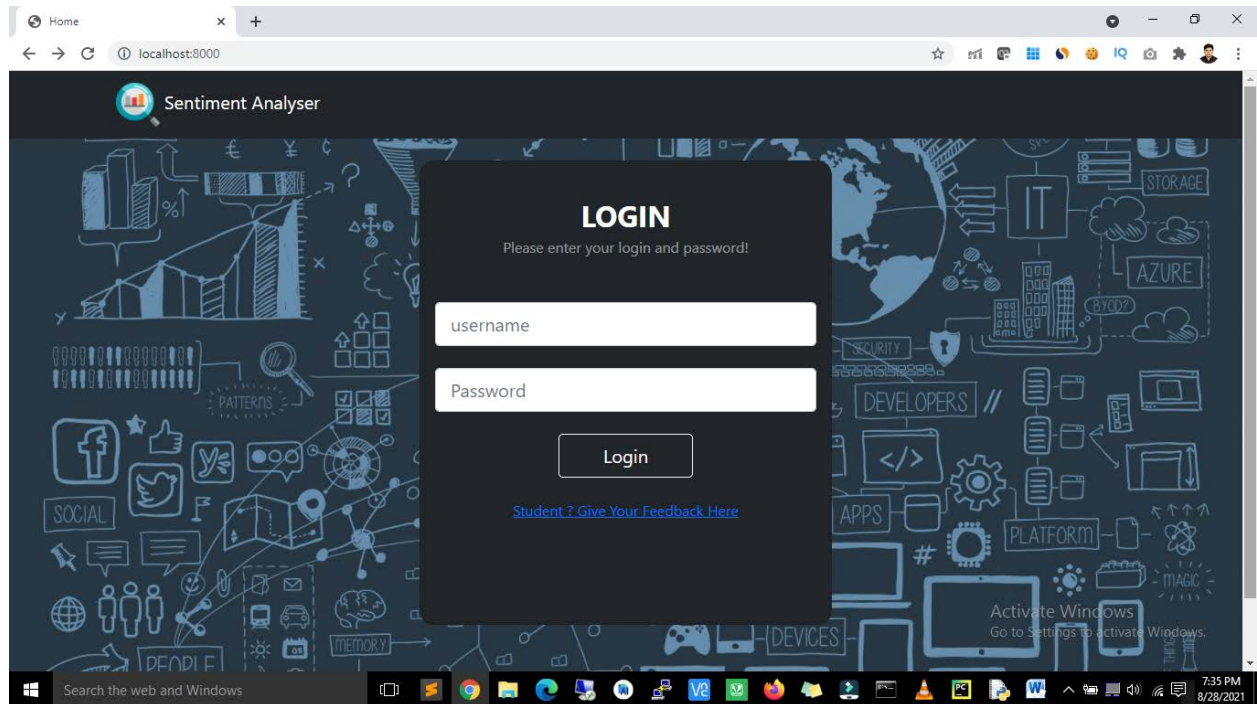
All the functionality of Student feedback sentiment analysis tool are working properly.

It take input properly and give the insights to the admin.

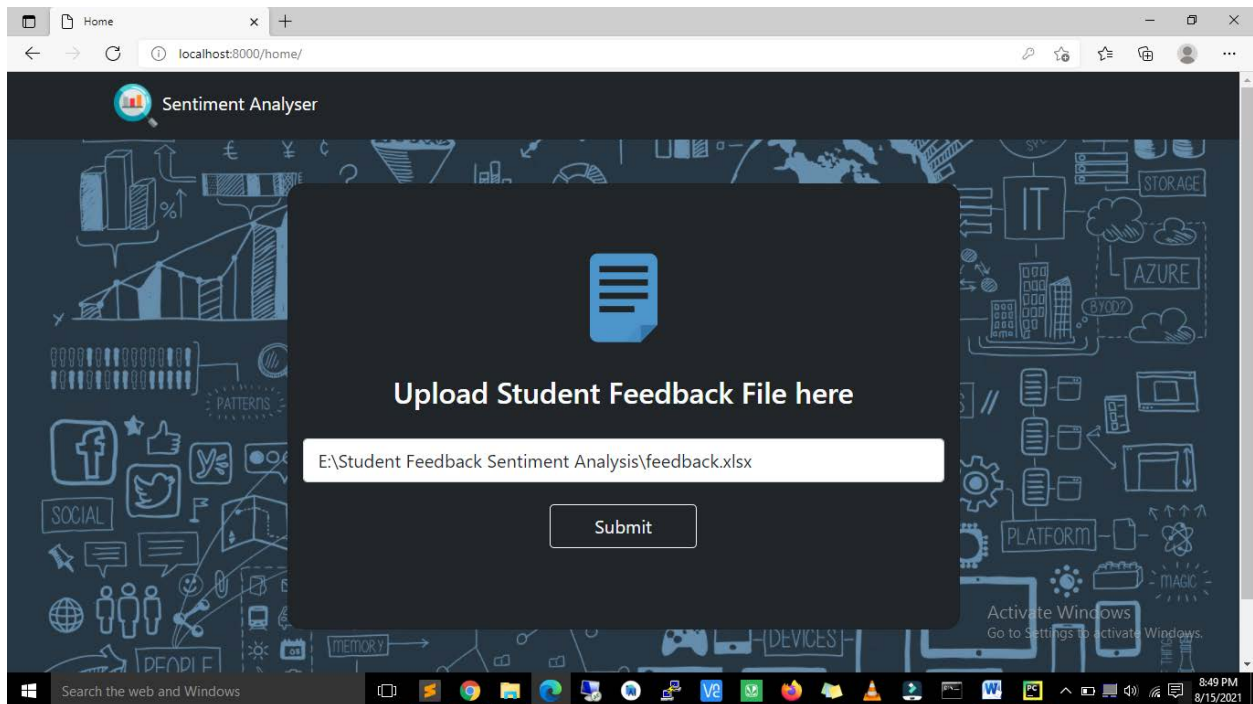
CHAPTER 6

SNAPSHOTS

1. Index Page



2. Home Page



3. Feedback Data Table Page

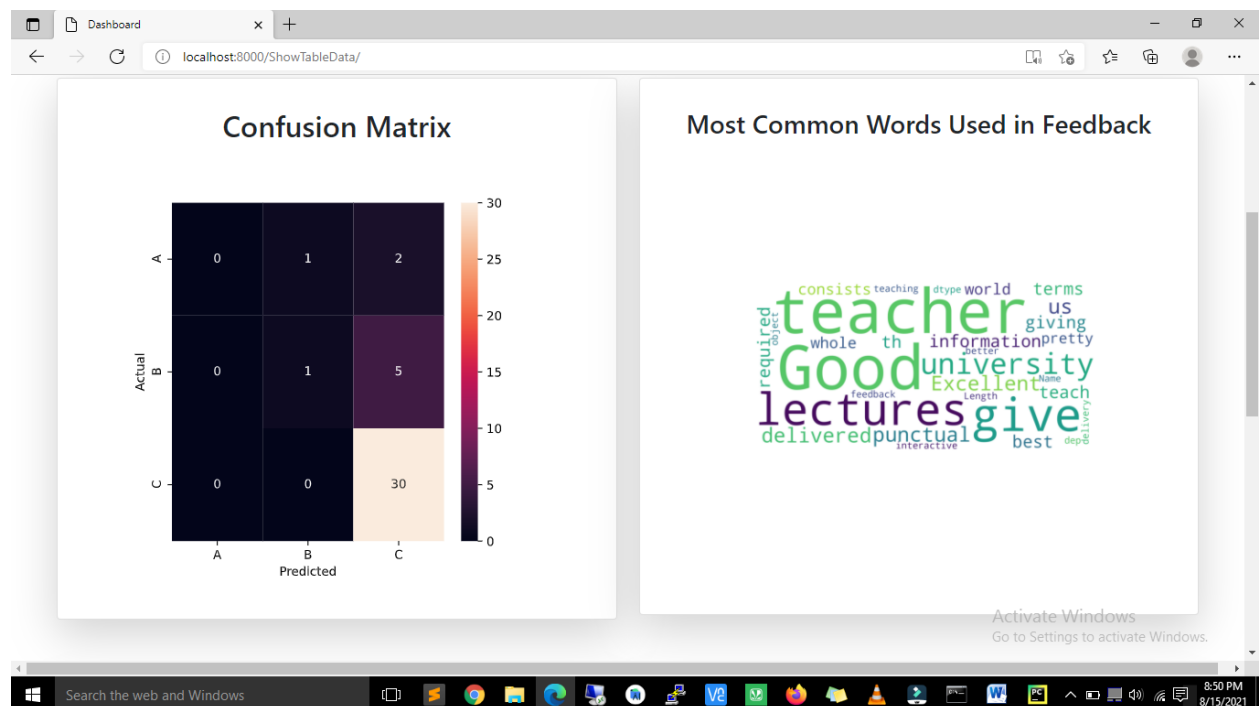
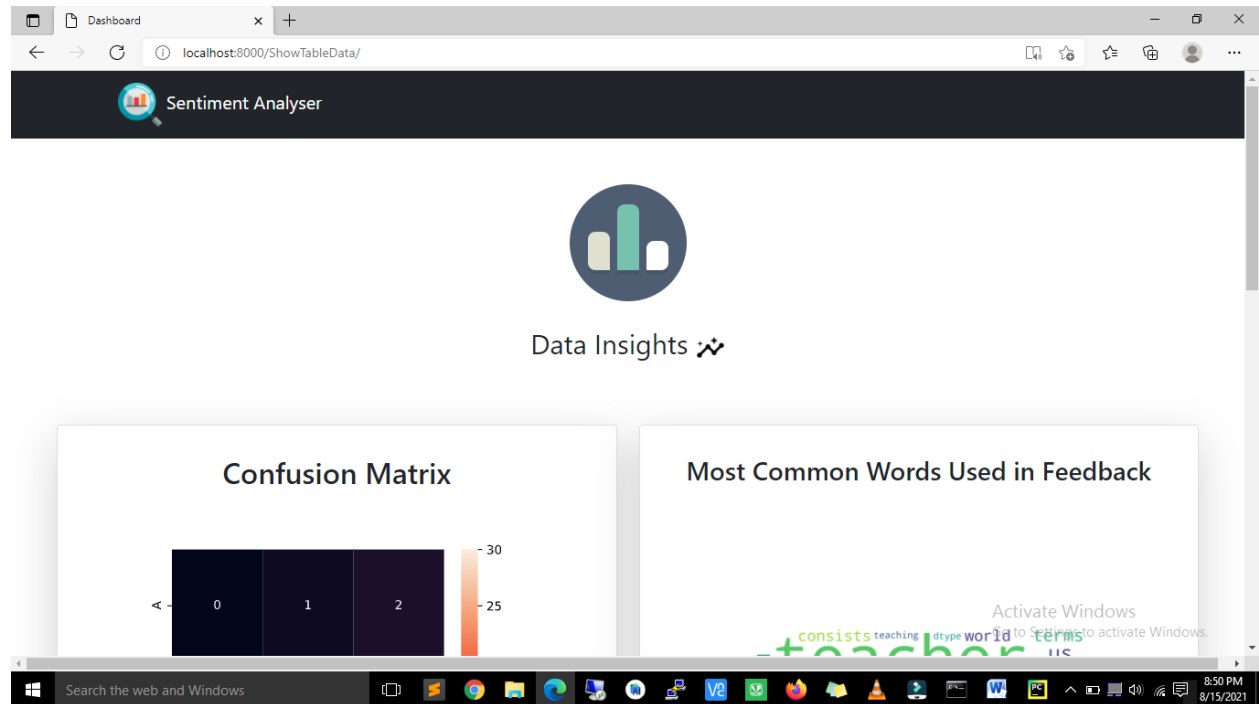
The screenshot shows a web browser window with the address bar displaying 'localhost:8000/ShowTableData/'. The page title is 'Sentiment Analyser'. Below the title is a large icon of a person standing next to a gear and a document. Below the icon is a button labeled 'Test & Train'. Below the button is a row of five sentiment icons (happy, neutral, sad, angry, and a black square) followed by the number '129'. Below this is a table with two columns: 'Feedback' and 'Point'. The table contains five rows of feedback data. The first row has a feedback text 'teacher are punctual but they should also give us the some practical knowledge other than theoretical' and a point of 0. The second row has 'Good' and a point of 1. The third row has 'Excellent lectures are delivered by teachers and all teachers are very punctual.' and a point of 1. The fourth row has 'Good' and a point of 1. The fifth row has 'teachers give us all the information required to improve the performance' and a point of 1. The table is partially obscured by a Windows taskbar at the bottom, which shows the time as 8:49 PM on 8/15/2021.

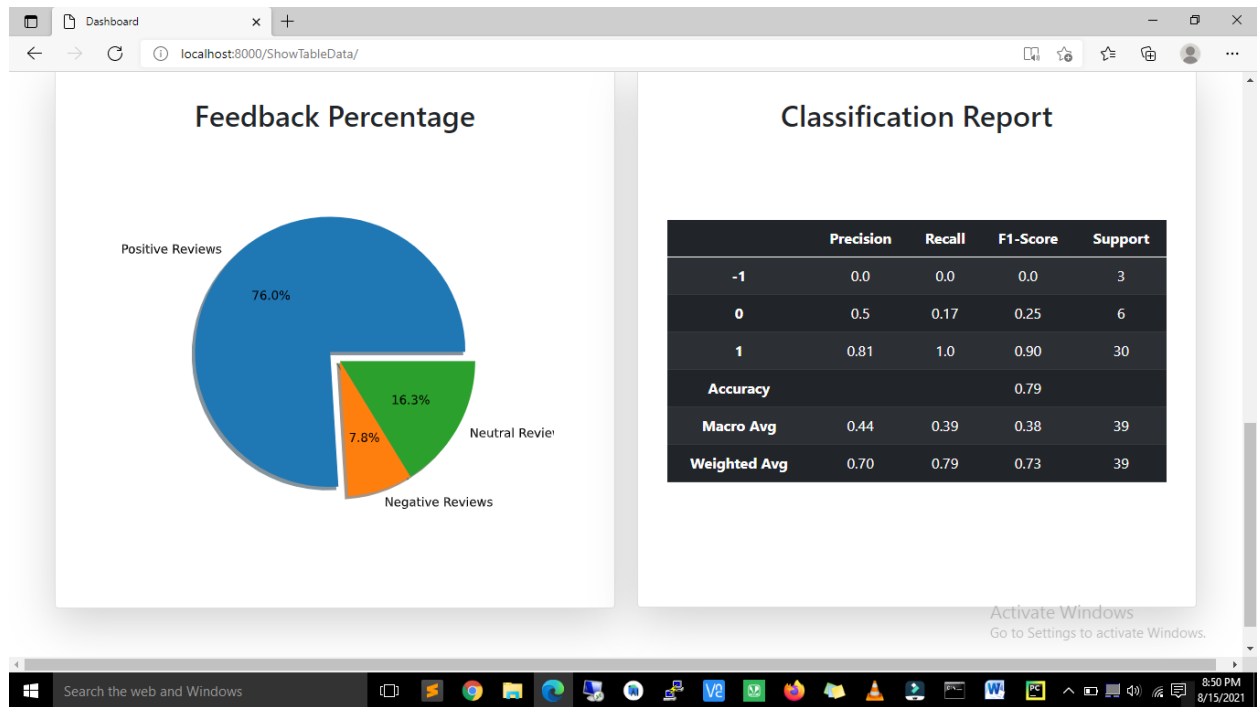
Feedback	Point
teacher are punctual but they should also give us the some practical knowledge other than theoretical	0
Good	1
Excellent lectures are delivered by teachers and all teachers are very punctual.	1
Good	1
teachers give us all the information required to improve the performance	1

The screenshot shows the same web browser window as the previous one, but with more feedback data displayed in the table. The table now contains 15 rows of feedback data. The first row is the same as in the previous screenshot. The second row is 'Good' with a point of 1. The third row is 'Excellent lectures are delivered by teachers and all teachers are very punctual.' with a point of 1. The fourth row is 'Good' with a point of 1. The fifth row is 'teachers give us all the information required to improve the performance.' with a point of 1. The sixth row is 'Yes' with a point of 1. The seventh row is 'good and punctual' with a point of 1. The eighth row is 'It is good' with a point of 1. The ninth row is 'Good' with a point of 1. The tenth row is 'Good' with a point of 1. The eleventh row is 'It's good when compare with my Btech lectures and teaching activities.' with a point of 1. The twelfth row is 'Good' with a point of 1. The thirteenth row is 'way of teaching is good' with a point of 1. The fourteenth row is 'University teaching here is very much dependent upon slides,though it is a easy way of teaching but still leaving some concepts unclear..otherwise interaction with the' with a point of 0. The fifteenth row is 'faculty is good Lecture delivery sometimes is not upto the mark and punctuality is fine' with a point of 0. The table is partially obscured by a Windows taskbar at the bottom, which shows the time as 8:50 PM on 8/15/2021.

Feedback	Point
teacher are punctual but they should also give us the some practical knowledge other than theoretical	0
Good	1
Excellent lectures are delivered by teachers and all teachers are very punctual.	1
Good	1
teachers give us all the information required to improve the performance.	1
Yes	1
good and punctual	1
It is good	1
Good	1
Good	1
It's good when compare with my Btech lectures and teaching activities.	1
Good	1
way of teaching is good	1
University teaching here is very much dependent upon slides,though it is a easy way of teaching but still leaving some concepts unclear..otherwise interaction with the	0
faculty is good Lecture delivery sometimes is not upto the mark and punctuality is fine	0

4. Dashboard Page





5. Student Feedback Page

Student Feedback

localhost:8000/studentfeedback/

GIVE YOUR FEEDBACK HERE

Your valuable feedback will help us to improve !

Leave a comment here

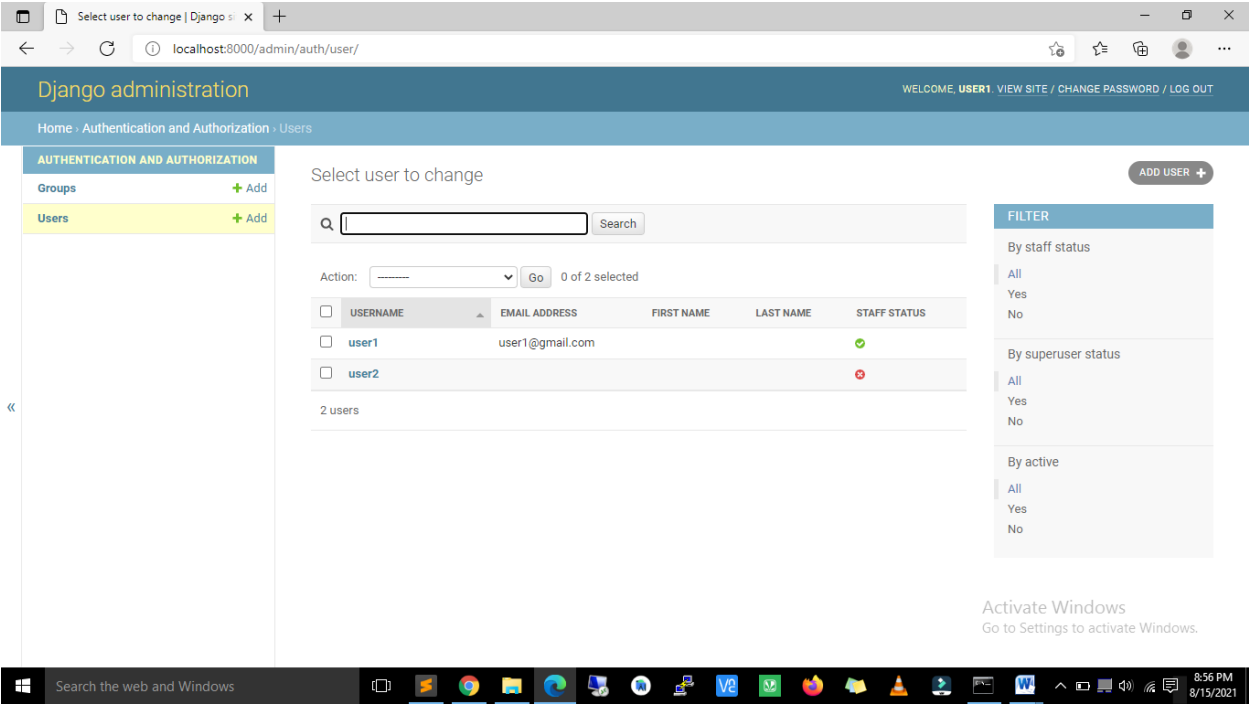
Submit

Activate Windows
Go to Settings to activate Windows.

Search the web and Windows

7:35 PM
8/28/2021

6. Admin Panel



CHAPTER 7

FUTURE ENHANCEMENT

Future Enhancement

In Future we will add some more features to our student feedback sentiment analysis tool like-

1. We can allow students to create their account on this tool directly to check their details.
2. To show more insights of feedback data, we can add more metrics in this dashboard.
3. We can add emotion sentiment analysis feature to this tool to get more information from the student feedback.
4. We can enhance this tool accuracy by adding large feedback dataset.

REFERENCES

BOOKS:

- Ian Somerville ‘Software engineering’
- Rajeev mall ‘Software engineering’

ONLINE REFERENCE:

1. www.google.co.in
2. <https://stackoverflow.com/>
3. <https://docs.djangoproject.com/en/3.2/>
4. <https://www.w3schools.com/>