UNIVERSITY OF WESTMINSTER

INFORMATICS INSTITUTE OF TECHNOLOGY

INFORMATICS INSTITUTE OF TECHNOLOGY

In collaboration with

UNIVERSITY OF WESTMINSTER

Sentiment Analysis for Product Review Star Rating Prediction using Generative Pre-trained transformer – GPT-03

by

Group SE-82

Department of Computing

5COSC021C - Software Development Group Project

Module Leader: Mr. Banuka Athuraliya

Supervised by: Mr. John Sriskandarajah.

UOW ID	IIT ID	Student Name
w1904573	20211507	R.Abhinav
w1866986	20200909	D.D.L.S.N.Jayawardhane
w1866983	20200655	H. H. Welikannage
w1898904	20211209	W.S.R.Nirmal Wijenayaka

Declaration Page

We hereby declare that this is the result of Sentiment Analysis for Product Review Star Rating Prediction using Generative Pre-trained transformer – GPT-03 implementation under the supervision of Mr. John Sriskandarajah.

Abstract

When it comes to buying a product, online customers cannot know about the real product until it arrives at the doorstep. To get a better understanding of the products in online, reviews and star rating is used.

The implementation done here is to generate the star rating using machine learning for the reviews other customers writes. This can prevent the mismatch in review and the star rating and use can get the idea of the product without reading the review and looking at the accurate star rating.

Acknowledgement

First and foremost, we would like to thank Mr. Banuka Athuraliya, the Software Development Group project module leader and Mr. John Sriskandarajah, our mentor, for taking their time out of their busy schedules to guide us. We also want to thank all the lecturers who guided us, as well as those who helped with this report.

Finally, we would like to appreciate the efforts of our team members, and we want to thank all our family and friends who helped us in any way.

Table of Contents

DECLARATION PAGE ..... **I**

ABSTRACT ..... **I**

ACKNOWLEDGEMENT ..... **I**

TABLE OF CONTENTS ..... **II**

TABLE OF FIGURES ..... **IV**

LIST OF TABLES ..... **IV**

CHAPTER 1: IMPLEMENTATION **1**

1.1 Chapter Overview 1

1.2 Overview of the Prototype 1

1.3 TECHNOLOGY SELECTION **2**

1.4 IMPLEMENTATION OF THE DATA SCIENCE COMPONENT **3**

1.5 Implementation of the backend component 4

1.6 Implementation of the front-end component 5

1.6.1 Home Page *5*

1.6.2 Product buy page. *6*

1.6.3 Comment page *7*

1.7 GIT Repository 8

1.8 Deployments/CI-CD Pipeline 10

1.9 Chapter Summary 11

CHAPTER 2: TESTING **12**

2.1 Chapter Introduction 12

2.2 Testing Criteria 12

2.3 Testing functional requirements 13

2.4 Testing non-functional requirements 16

2.5 Unit testing 17

2.6 Performance testing 19

2.7 Usability testing 20

2.8 Compatibility testing 20

2.9 Chapter Summary 20

CHAPTER 3: EVALUATION **21**

3.1 Chapter Overview 21
3.2 Evaluation methods 21
3.3 Quantitative evaluation 22
3.4 Qualitative evaluation 22
3.5 Self evaluation 22
3.6 Chapter Summary 23
CHAPTER 4: CONCLUSION 24
4.1Chapter Overview 24
4.2 Achievements of aims and objectives. 24
4.3 Limitations of the research 24
4.4 Future enhancements 25
4.5Concluding remarks 26
APPENDIX I
Work break down I

Table of Figures

Figure 1: Product Table 4
Figure 2:Comment Table 4
Figure 3: Mani Page of The Web Application 1 5
Figure 4: Mani Page of The Web Application 2 6
Figure 5: Product Buy Page of The Web Application. 6
Figure 6: Comment Page Of The Web Application. 7
Figure 7: Git Repository Screen Shot. 8
Figure 8: GitHub Desktop Screenshot. 9
Figure 9: Continues Integration Screenshot 1. 10
Figure 10: Continues Integration Screenshot 2. 10
Figure 11: Unite Testing Figure 1. 17
Figure 12: Unite Testing Figure 2. 17
Figure 13: Unite Testing Figure 3. 18
Figure 14: Comment Page Of The Web Application 19
Figure 15: Product Buy Page Of The Web Application. 19

List of Tables

Table 1: Functional Requirement Test Table 15
Table 2: Non-functional Requirements Test Table. 16
Table 3: Work Breakdown Table. ii

Chapter 1: Implementation

1.1 Chapter Overview

This report outlines the implementation of a prototype for determining a product's overall star rating based on customer reviews. The prototype utilizes OpenAI's GPT-3 language model for sentiment analysis and Python for preprocessing the text data. The preprocessing step involves removing special characters and numbers from the customer comments and performing text data cleaning. The sentiment analysis process involves sending the preprocessed data to the OpenAI API, which returns the count of words with positive and negative connotations. The backend component uses the returned data to generate the overall sentiment rank of the comments and calculate the overall star rating for the product. The implementation of the data science and backend components have ensured the efficiency and accuracy of the sentiment analysis process, resulting in a robust solution for determining product ratings based on customer feedback.

1.2 Overview of the Prototype

The prototype's objective is to offer a tool for creating a product's overall star rating based on user reviews. The comments are first preprocessed to get rid of extra spaces at the beginning and end of each phrase, special characters, and numerals. The language model of OpenAI is then applied to the preprocessed comments, analyzing the phrase to determine the number of terms with positive and negative meanings. The total of these numbers is used to calculate the remark's overall score on a scale of 1 to 5, where 5 represents a highly positive opinion and 1 represents a strongly unfavorable comment. The validity of the prototype's output is dependent on the accuracy of OpenAI's language model in recognizing sentiment in text. In recent years, sentiment analysis has been widely researched and improved upon in the natural language processing field. Numerous studies have compared the performance of various sentiment analysis models on large datasets, with some models showing high accuracy levels. By aggregating the sentiment scores of multiple comments, the prototype is able to generate an overall star rating for the product. This information can be useful for businesses to gauge customer satisfaction and make improvements where necessary. Additionally, by providing an objective analysis of customer feedback, the prototype

can assist in reducing subjectivity in the decision-making process. In conclusion, the prototype provides a tool for generating an overall star rating for a product based on customer comments, using a combination of text preprocessing and sentiment analysis. Further research and improvement of the language model used in the prototype may lead to even more accurate results.

1.3 Technology Selection

Objective:

This report's goal is to describe the technological choices made for the creation of a product's overall star rating, which is based on user reviews.

Technologies and tools:

1. Text preprocessing: Extra spaces, special characters, and digits will be removed from the comments during preprocessing. Python string manipulation methods will be used for this.
2. Sentiment Analysis: The preprocessed comments' sentiment will be examined using OpenAI's language model. The amount of phrases with good and negative connotations will be determined by the model, and this data will be used to establish the comment's total score.
3. Python: Python will be used as the prototype's main programming language. It will be used to prepare texts and build sentiment analysis algorithms.
4. OpenAI: For sentiment analysis, OpenAI's language model will be employed. The model was chosen for this project since it has been trained on a sizable dataset and has demonstrated great accuracy levels in sentiment analysis.

Conclusion:

The selected tools and technologies are expected to provide a robust and efficient solution for generating an overall star rating for a product based on customer comments. Python and OpenAI's language model have been chosen due to their popularity and suitability for text processing and sentiment analysis tasks. The combination of these tools is expected to provide an accurate and objective analysis of customer feedback, which will be useful for businesses to gauge customer satisfaction and make improvements where necessary.

1.4 Implementation of the data science component

The implementation of the data science component involved several key technologies to perform sentiment analysis on product reviews. The main focus was to use algorithmic techniques to identify the overall star ratings of a product, based on customer comments.

The following tools and libraries were used in the implementation:

OpenAI's GPT-3 language model to perform sentiment analysis on customer comments.

Python programming language and its popular libraries such as Re for pre-processing the data.

Re library was used to remove special characters and numbers from the customer comments, and to perform cleaning and processing of the text data.

The code is organized into several functions, each of which is responsible for a specific task. The pre-processing function is used to remove unwanted characters and numbers from the customer comments, and to perform cleaning and processing of the text data. The sentiment analysis process is performed by sending the pre-processed data to OpenAI's GPT-3 language model, which returns the count of words with positive and negative connotations.

Finally, the calculate function is used to calculate the overall sentiment score of the customer comments by determining the ratio of positive words to the total number of words in the comments. The calculated sentiment score is then rounded to one decimal place to obtain the final rating of the product.

In conclusion, the data science component of this project has been implemented successfully and is capable of performing sentiment analysis on customer comments to determine the overall star rating of a product. The use of OpenAI's GPT-3 language model and the Python programming language has ensured the efficiency and accuracy of the sentiment analysis process.

1.5 Implementation of the backend component

The database used for the implementation is SQL and to interact with the database PHP is used. The PHP components are added after the front-end static web pages are created. For the PHP component msqli procedure method is used. To run the database and the PHP file Xampp is used throughout the implementation. For the database, there are two tables called product and comments.

The products table contains the detail of the product and the product star rating as “psRate”.

pId	pName	pImg	pPrice	pDescript	psRate
2	iPhone 14 Plus - 128GB - Blue	i14p.jpg	449999.00	The iPhone 14 Plus is an excellent choice if you'r...	3.70

Figure 1: Product Table

The comment table contains the comment in “comm” column and the comment star rating in “csRate” column.

cId	comm	csRate	pId
1	Good product	5.0	2
2	Bigger display with better colors	5.0	2
3	Bad mic But Other components are good	3.3	2
17	I was nervous about the condition and especially a...	4.0	2
23	So this is my first iphone! I have been an android...	5.0	2
41	How is everybody feeling about...	1.9	2
40	I feel like the battery i...	1.7	2

Figure 2:Comment Table

1.6 Implementation of the front-end component

For the front-end component, the team selects HTML, CSS, and JavaScript for the web application. First team members who work in the front end created the design of the page without PHP and created using HTML, CSS, and JavaScript as a static web page. Frameworks have not been used for this implementation. The JavaScript component is used for the navigation bar and to show the star rating of the product and comment.

1.6.1 Home Page

Home page contains the products. for the prototype one product is used. For each product their star rating is shown.

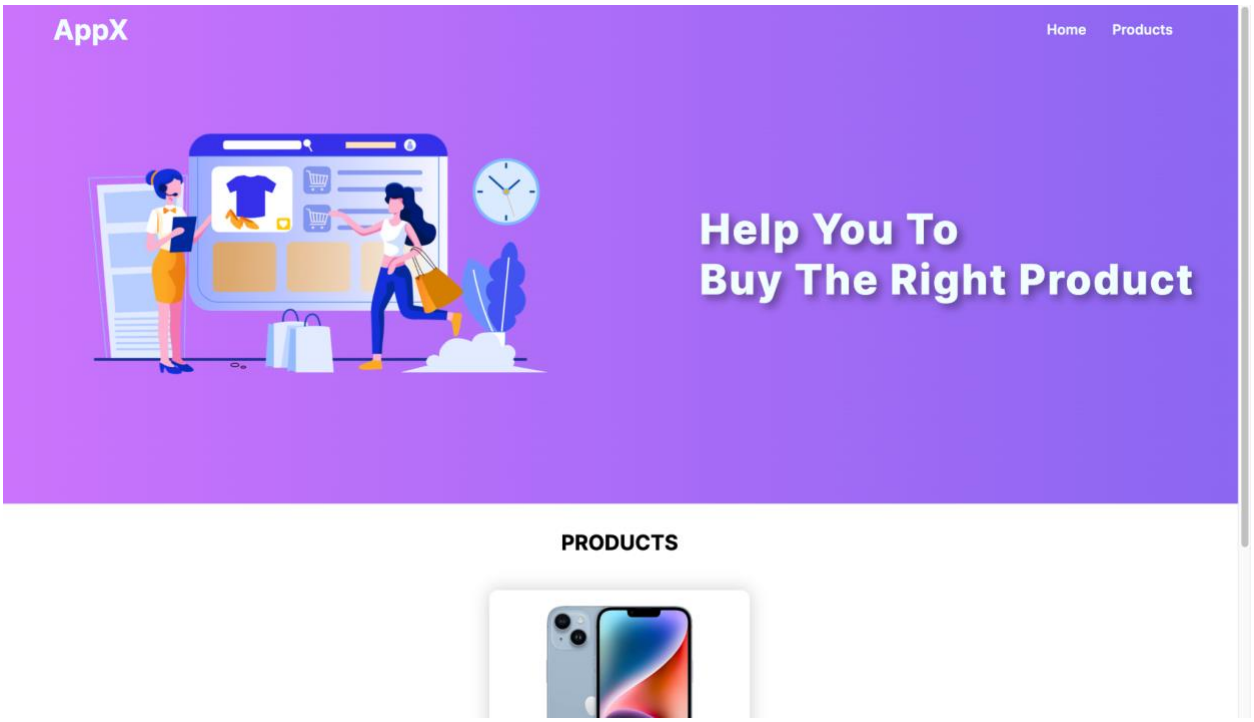


Figure 3: Mani Page of The Web Application 1

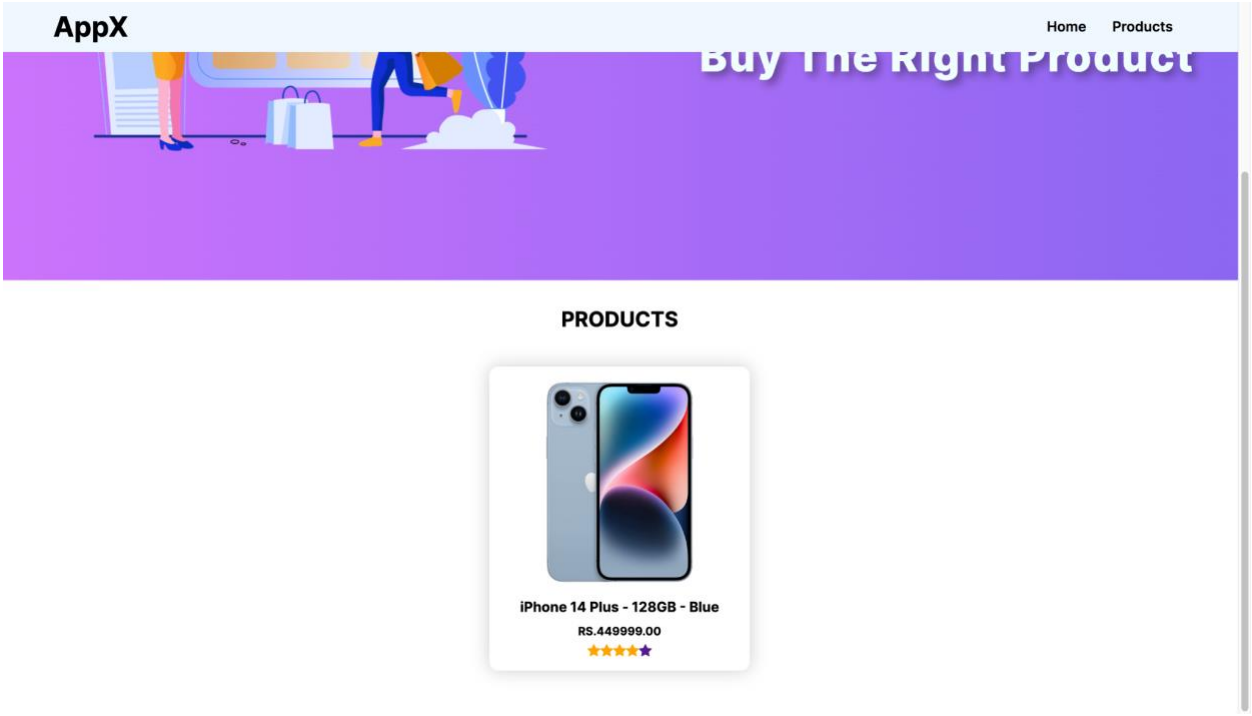


Figure 4: Mani Page of The Web Application 2

1.6.2 Product buy page.

Product buy page contains the product details, product star rating, write comment button and the comments written for the product and star rating for each product.

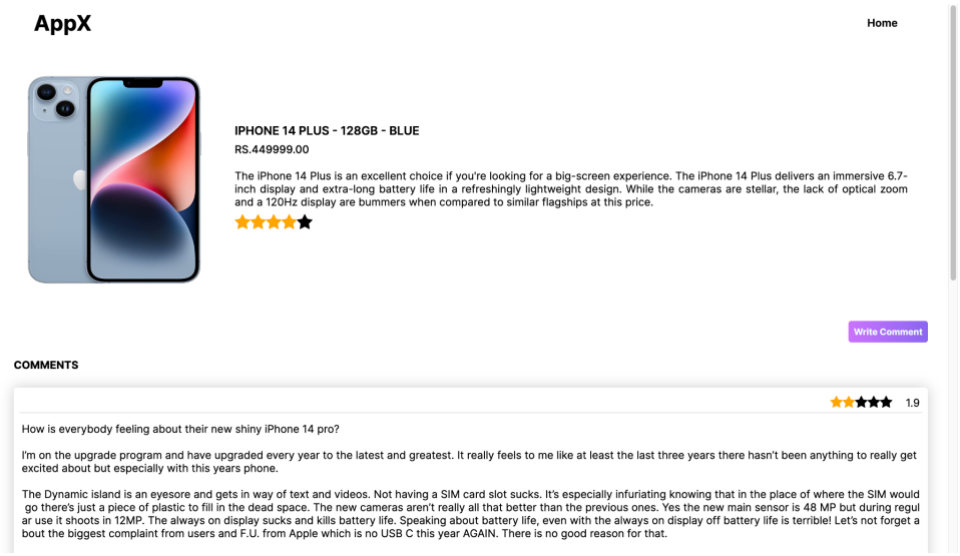


Figure 5: Product Buy Page of The Web Application.

1.6.3 Comment page

When the comment button is clicked the user will be directed to the comment page. Here user can write comments.

Figure 6: Comment Page Of The Web Application.

1.7 GIT Repository

For the project version control system git is used and the team used the GitHub desktop application for the version control. First, the repository is created, and Folders were created for each main component for example Mani, comment, product, and python backend. Then team members will work on each folder and push it into the main branch.

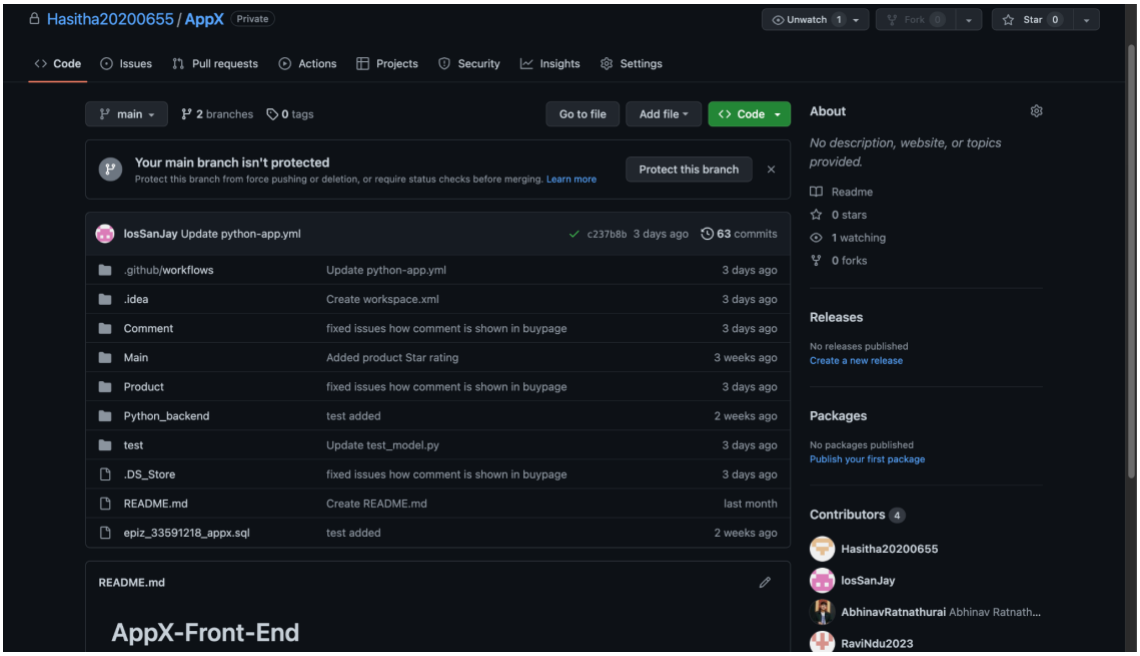


Figure 7: Git Repository Screen Shot.

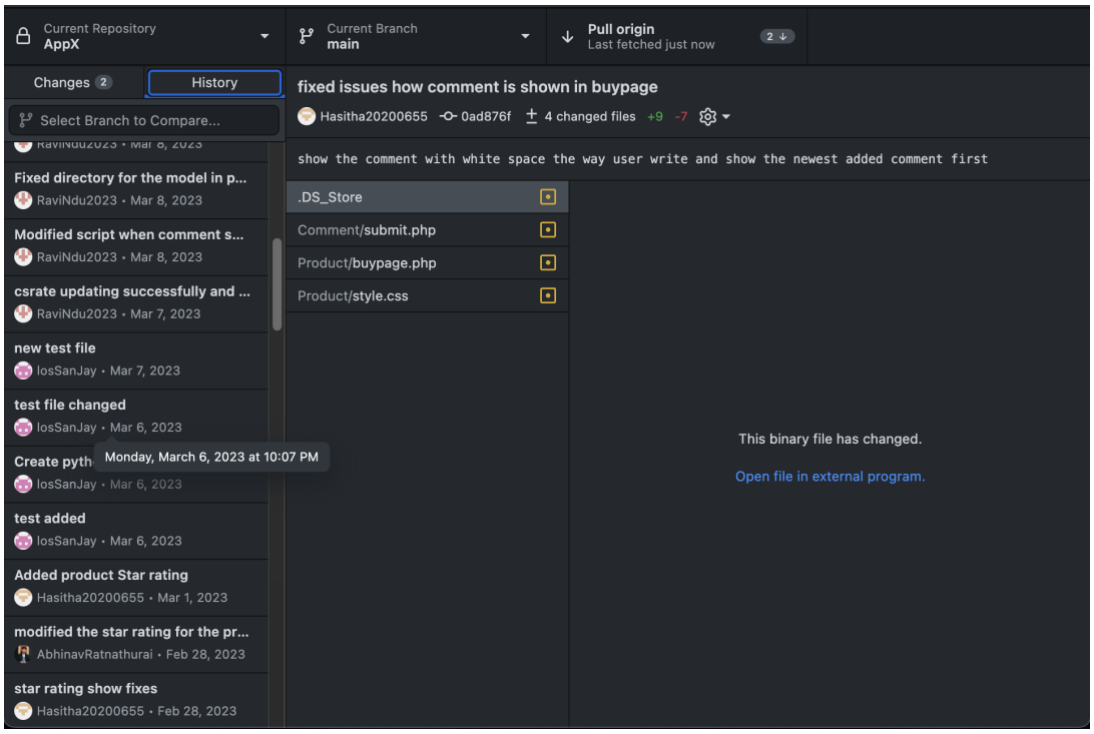


Figure 8: GitHub Desktop Screenshot.

1.8 Deployments/CI-CD Pipeline

In order to release a new version of software, a continuous integration and continuous deployment (CI/CD) pipeline is necessary.

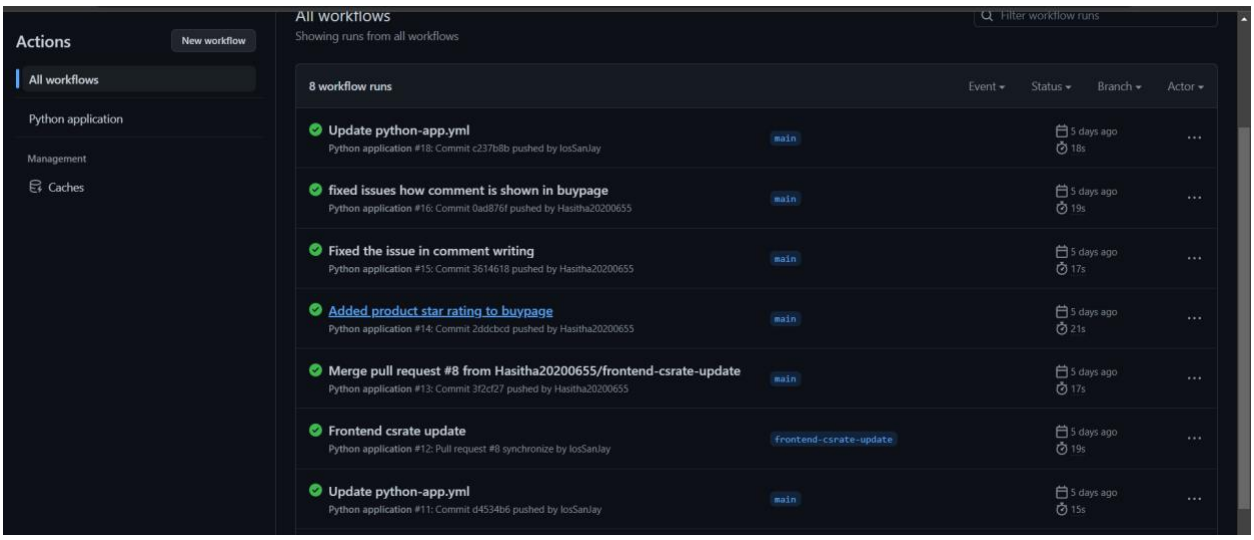


Figure 9: Continues Integration Screenshot 1.

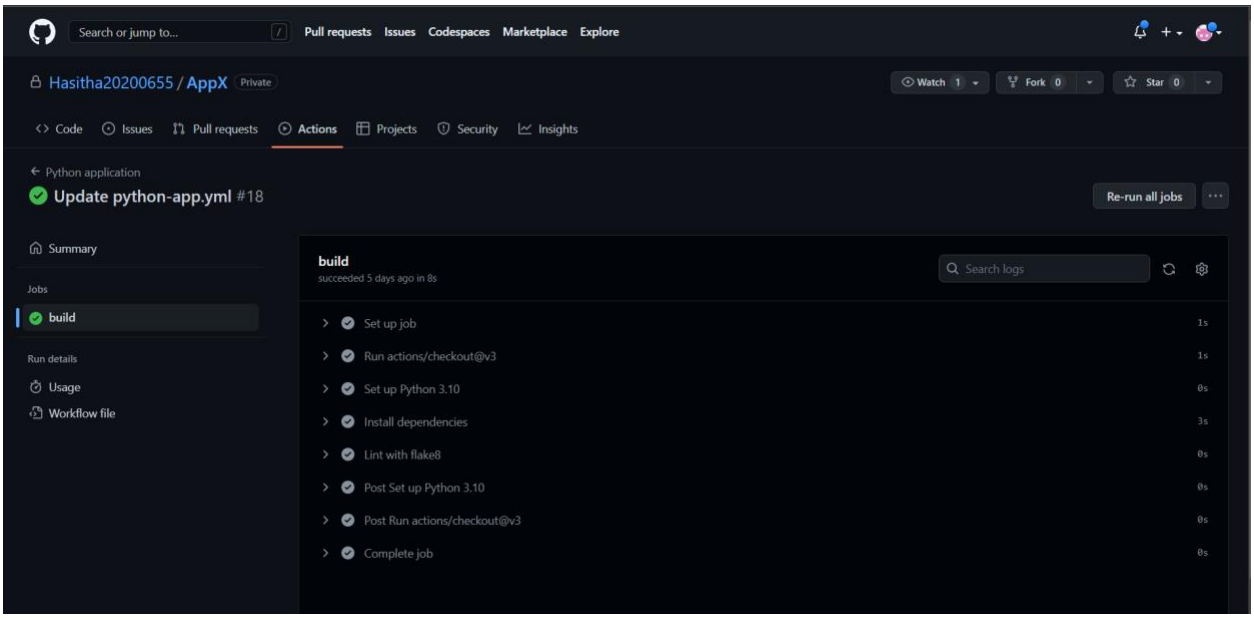


Figure 10: Continues Integration Screenshot 2.

1.9 Chapter Summary

In this chapter, there are a few goals to describe. The starting point is at the top of the paragraph. In that topic, describe the main idea about the prototype and describe the operational side of the prototype. In the second point (technology selection), explain the technologies the team used to build these prototypes. The implementation of the data science component explains how to generate the star rating and what technologies to use for that. In the implementation of the front-end component, show the front end and explain the method that helped build the front end. At last, the GIT repository And the Deployments/CI-CD Pipeline responds.

Chapter 2: Testing

2.1 Chapter Introduction

All projects must include testing, but software development is especially dependent on it. It entails the procedure of confirming and validating the performance, quality, and functionality of a system or product to make sure that it complies with the requirements and operates as intended.

Testing needs to be done at every stage of the project lifecycle, from gathering requirements to deployment. To guarantee that testing is complete and efficient, it is crucial to organize testing activities and allot enough time and resources.
The cost and time needed to remedy faults and errors can be reduced with proper testing because it helps to find them early in the development process. The effectiveness, dependability, and compliance with quality requirements of the system or application are also ensured by testing.

2.2 Testing Criteria

In this Sentiment Analysis for Product Review Star Rating has a set of both functional and non-functional requirements. The testing is done to make sure the system functions as it should. It concentrates on both functional and non-functional requirements for the system. generated test cases for this. In order to evaluate and confirm that the system meets its objective, numerous of the conditions that were previously listed were used when developing the test cases.

2.3 Testing functional requirements

Feature Tested	Test Case Description	Expected Outcome	Actual Outcome	Status
FR4	**Load the page**	**The system should be able to make a new login page for each new user.**	**The system should be able to make a new login page for each new user.**	**Pass**
FR5	**Choose the product.**	**The user should be able to choose a product. the system should be able to able to show product price, available content, measurement, and other details about the product**	**The user should be able to choose a product. the system should be able to able to show product price, available content, measurement, and other details about the product**	**Pass**

FR6	**Review the product**	**The user should be able to review the products**	**The user should be able to review the products**	**Pass**
FR7	**Analyze the reviews given by customers**	**By using reviews, the system should be able to identify positive reviews and negative reviews. The system should identify positive and negative words**	**By using reviews, the system should be able to identify positive reviews and negative reviews. The system should identify positive and negative words**	**Pass**
FR8	**Calculate the star rating of the review**	**Calculate the star rating using the sentiment of the review**	**Calculate the star rating using the sentiment of the review**	**Pass**

FR9	**show the star point**	**The system should be able to show the star rating made by the user by the comment**	**The system should be able to show the star rating**	**Pass**
FR10	**Final product overall star point**	**The system should be able to show the final star point count of the product**	**The system should be able to show the final star point count of the product**	**Pass**

Table 1: Functional Requirement Test Table

2.4 Testing non-functional requirements

Feature Tested	Test Case Description	Expected Outcome	Actual Outcome	Status
NFR1	**Accuracy**	**The system should be accurate when analyzing the sentiment of customer reviews.**	**The system should be accurate when analyzing the sentiment of customer reviews.**	**Pass**
NFR2	**Performance**	**Website should work well without lag or performance issues**	**Website should work well without lag or performance issues**	**Pass**
NFR3	**Reliability**	**The system should be reliable to get a star rating of products using users' reviews.**	**The system should be reliable to get a star rating of products using users' reviews.**	**Pass**
NFR4	**Usability**	**The website user experience should be simple and user-friendly**	**The website user experience should be simple and user-friendly**	**Pass**

Table 2: Non-functional Requirements Test Table.

2.5 Unit testing

Unit tests were done using python unit testing. Checked whether those test cases were passed or not.

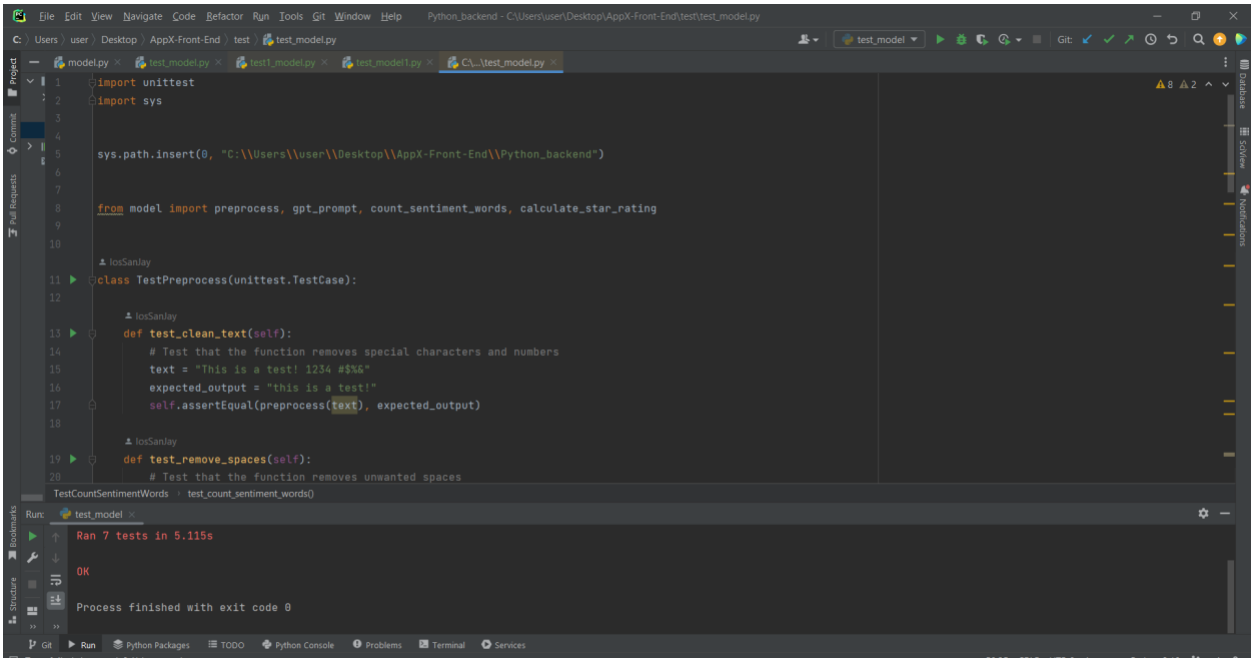


Figure 11: Unite Testing Figure 1.

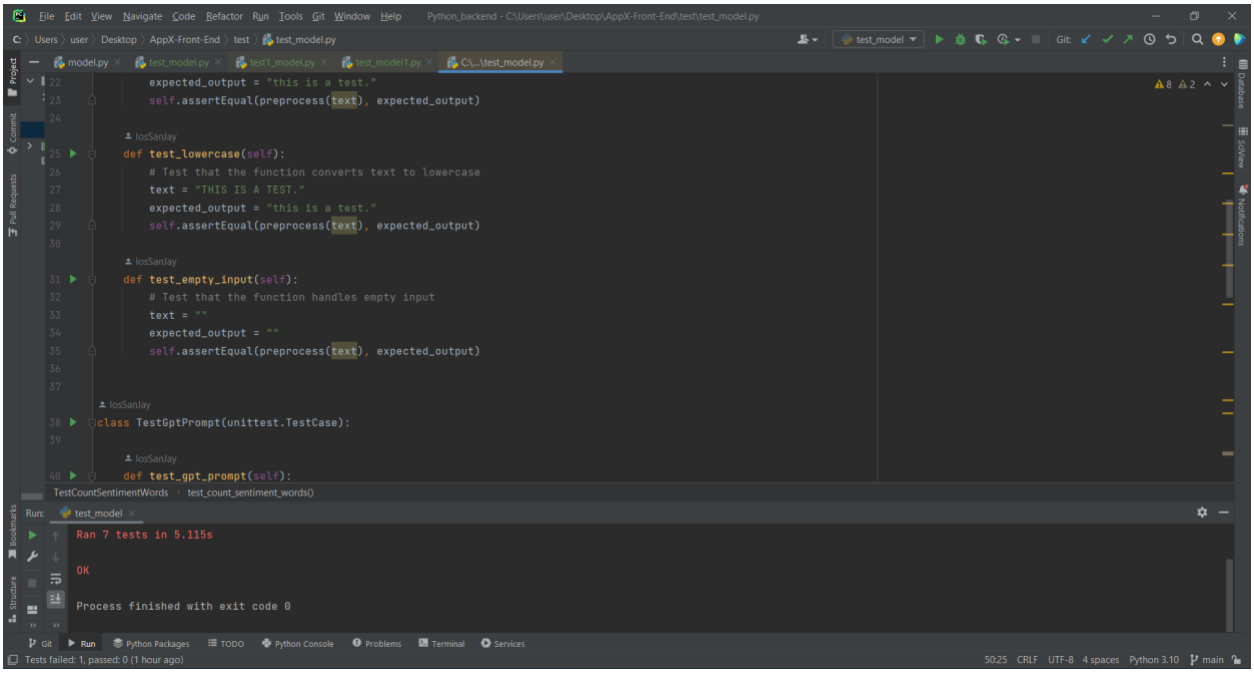


Figure 12: Unite Testing Figure 2.

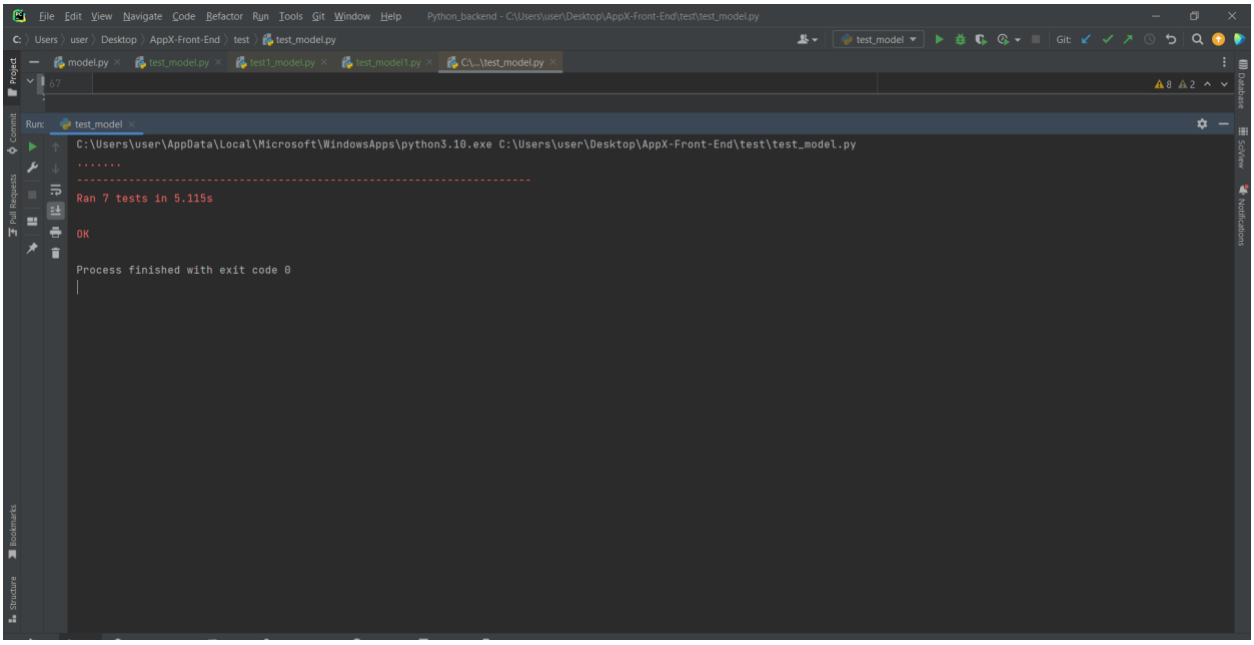


Figure 13: Unite Testing Figure 3.

2.6 Performance testing

This software is used for demonstration purposes only. The user can add any positive, negative comment there. After submitting the comment at the product page, the star rating was showing.

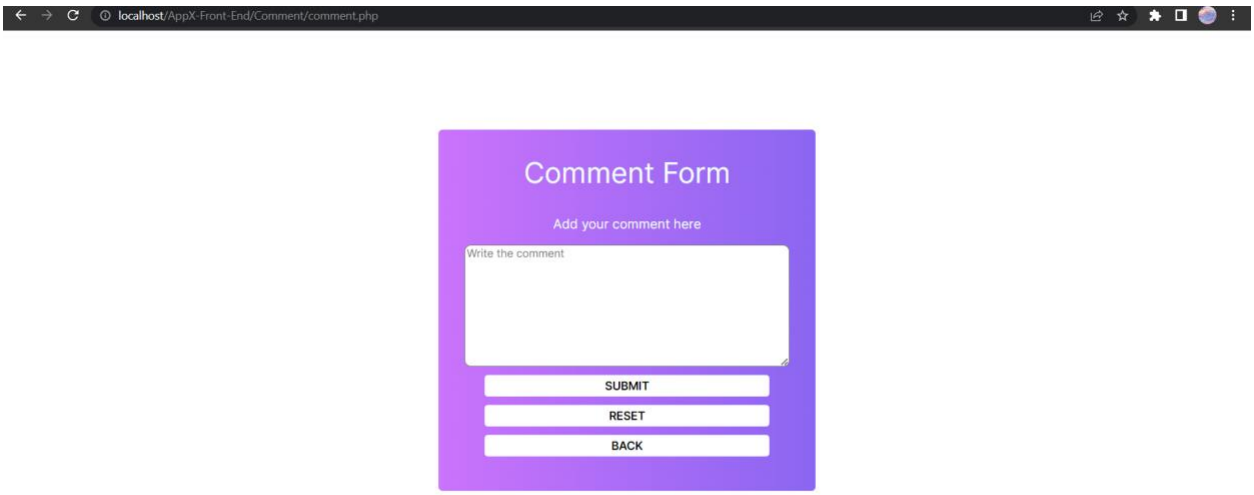


Figure 14: Comment Page Of The Web Application

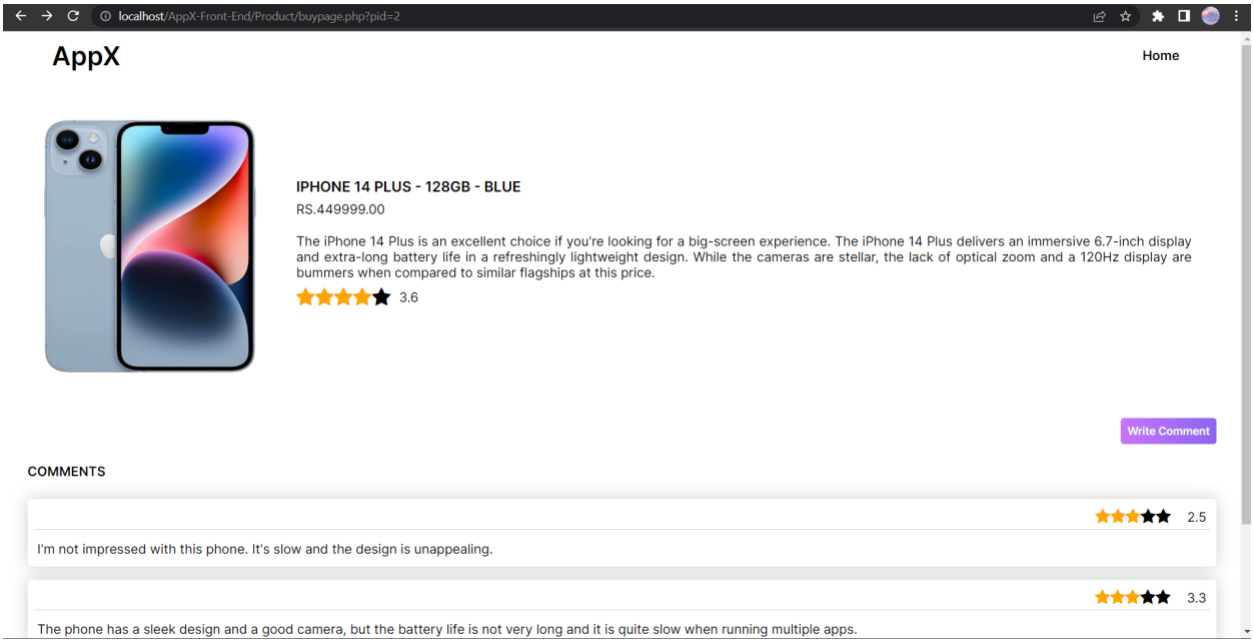


Figure 15: Product Buy Page Of The Web Application.

2.7 Usability testing

This test was conducted to check the UI / UX of this application. A small group of end-users is used to expose usability flaws in software. Usability testing focuses just on the application's ease of use, flexibility in handling controls, and ability to meet its goals. The user interface of this software application was user friendly responsiveness interface. This application includes an ‘add comment’ option. Users can add a comment there and submit.

2.8 Compatibility testing

Compatibility testing, a subset of software testing, examines a software program's ability to function on a range of hardware, network configurations, and mobile devices. It is a non-functional testing technique. This application is tested in windows. It works properly.

2.9 Chapter Summary

This chapter concentrated on the project's testing phase. The testing information for both the functional and non-functional requirements, as well as the numerous testing assessments of implementation, were covered in the chapter.

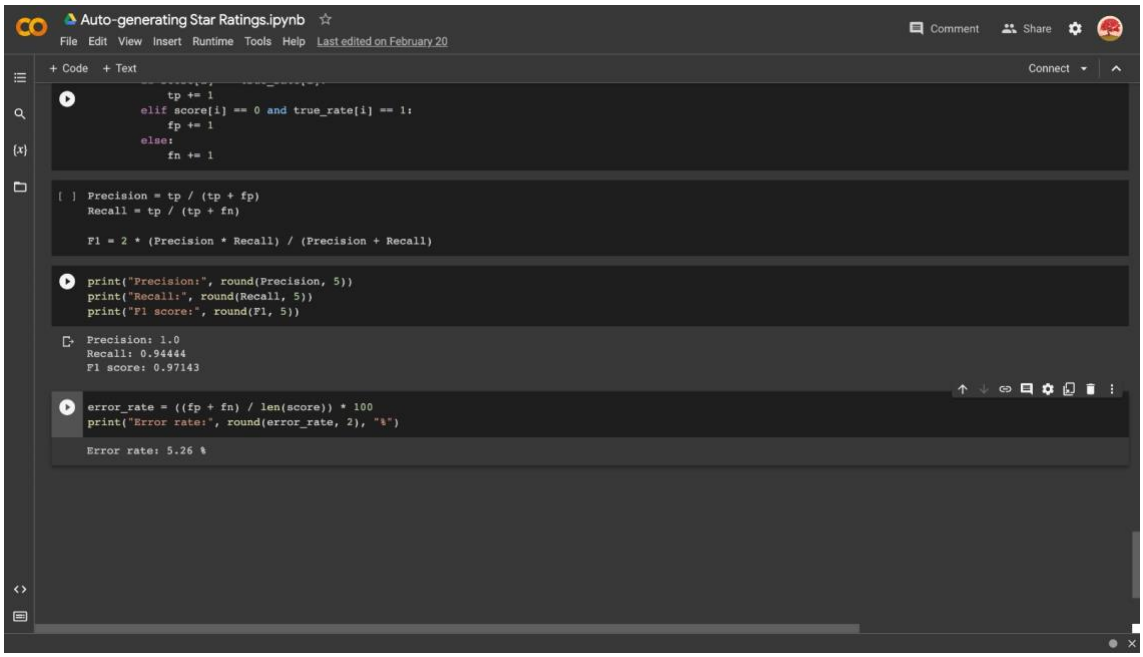
Chapter 3: Evaluation

3.1 Chapter Overview

This chapter covers the Evaluation of the project. In the chapter evaluation methodologies are used to test the process and the prototype as well as the feedback of the end users, domain experts and the industry experts.

3.2 Evaluation methods

In this project use Confusion matrix evaluation method only to the model. A table called a confusion matrix is used to assess how well a machine learning model is performing. It allows for the calculation of performance indicators like accuracy, precision, recall, and F1-score by displaying the anticipated and actual labels of a classification task. The F1-score is a weighted average of precision and recall, which evaluates the proportion of true positives among anticipated positives, the proportion of true positives among actual positives, and overall accuracy. Studying these data can assist find areas where the model needs to be tweaked and modified to perform better. According to the project's model evaluation, a precision of 1.0, recall of 0.94, and F1 score of 0.97 is received together with an error rate of 5.26%.



3.3 Quantitative evaluation

Quantitative assessment is a data quantification and evaluation technique that employs numerical measurements or statistical methodologies. It involves collecting and analyzing numerical data to draw conclusions about a particular event or procedure. Quantitative assessment was used twice in this undertaking. Surveys are one type of quantifiable assessment technique. To build this prototype, the team must gather data, which is accomplished through the use of surveys. The primary sentiment analysis system makes use of dates to analyze and produce star points. The result of statistical analysis is to collect and evaluate data, find patterns and trends, and make predictions based on the data. It is one of the quantitative assessment techniques.

3.4 Qualitative evaluation

“Star rating help users make informed decisions about which products or services to choose. By reading the ratings and reviews, users can get an idea of the quality of a product before making a purchase.”

-Sandaru Bandara (End User)-

3.5 Self evaluation

Abhinav

Team leader, build the backend model of the project. The main star generating part is done by the leader and also the connection of the model with the database.

Loshini

Creating Front end comments adding part, all the testing and CI pipeline build the testing and CI pipeline.

Hasitha

Main page design and creating the page using HTML and CSS. Then converting all the pages to dynamic pages using PHP and creating MySQL database. Connecting all the pages with the database.

Nirmal

Automation of the python script to work automatically when the system runs, and also creating a product buy page.

3.6 Chapter Summary

The project evaluation explained with evaluation methods, Quantitative evaluation, Qualitative evaluation, self-evaluation. In the qualitative evaluation it contains the feedback of end users, domain experts and industry experts. In the self-evaluation each member of the group is done individually and contains the feedback of each member.

Chapter 4: Conclusion

4.1Chapter Overview

This chapter covers the achievements of the aims and objectives of the prototype and the limitations of the research already conducted. Also, future enhancements will be explained in this chapter.

4.2 Achievements of aims and objectives.

By research, create and implement the product to calculate the star rating using GPT-3 for online products created. Team was able to complete the project on time with some feature reduction.
The project started with identifying problems more about this covered in the introduction chapter of the SRS. Then checked the research done in the problem domain and the competitors. All details about literature review are covered in SRS chapter 2. The implementation team went with the agile methodology. The requirements for the project are discussed in detail in the SRS. For the design, high level architecture, class diagram, sequence diagram, process flow charts are created in the SRS with the UI for the web application. Testing for the prototype is covered in the chapter 2 of this document with unit testing, performance testing, usability testing.

4.3 Limitations of the research

- Dependence on OpenAI API: The research relies on the OpenAI API for sentiment analysis, which may not always be accurate and may result in false predictions.
- Limited data sample: The research uses a limited number of data samples for evaluation, which may not represent the entire population of comments.
- Human bias: The research may still have human bias, as the pre-processing step and the calculation of the sentence rank may be influenced by the personal opinions of the researcher.
- Generalizability: The findings of this research may not be generalizable to other domains or other types of comments, as the sentiment analysis may be context specific.

- Limited evaluation metrics: The research uses only precision, recall, and F1 score to evaluate the performance of the sentiment analysis, which may not provide a comprehensive assessment of the performance.
- Lack of tuning of hyperparameters: The research does not perform hyperparameter tuning, which could potentially result in suboptimal performance.
- Lack of further analysis: The research does not perform further analysis to better understand the factors that contribute to the sentiment analysis performance.

4.4 Future enhancements

Future enhancements for this research could include:

- Improving the accuracy of the sentiment analysis process by training the language model on a larger dataset or using a combination of different models for sentiment analysis.
- Incorporating additional sources of data, such as social media posts, to provide a more comprehensive analysis of customer feedback.
- Developing a more sophisticated algorithm for aggregating sentiment scores from multiple comments, to generate a more accurate overall star rating for the product.
- Enhancing the user interface to make the prototype more user-friendly and accessible for non-technical users.
- Expanding the application to cover a wider range of products, including services and intangible products, to provide more comprehensive customer feedback analysis.
- Providing more detailed and actionable insights to businesses, such as areas for improvement and customer preferences, by incorporating natural language processing techniques.
- Evaluating the performance of the prototype on a larger dataset, to validate its scalability and generalizability to other domains.
- In conclusion, there is significant room for improvement in this research, and future enhancements will help to provide more accurate and comprehensive customer feedback analysis.

4.5Concluding remarks

In this prototype, make a perfect five-star rating for the review. So this will help to provide a good star rating for the product and will help people get a good idea about the product.

Appendix

Work Breakdown Table.

Task	Group Members			
	Abhinav	Nirmal	Hasitha	Loshani
Chapter 1 Implementation				
1.1 Chapter Overview	✓			
1.2 Overview of the prototype	✓			
1.3 Technology selections	✓			
1.4 Implementation of the data science component	✓			
1.5 Implementation of the backend component			✓	
1.6 Implementation of the front end component			✓	
1.7 GIT Repository			✓	
1.8 Deployments/CI-CD Pipeline				✓
1.9 Chapter Summary		✓		
Chapter 2 Testing				
2.1 Chapter Introduction				✓
2.2 Testing Criteria				✓
2.3 Testing functional requirements				✓
2.4 Testing non-functional requirements				✓
2.5 Unit testing				✓
2.6 Performance testing				✓
2.7 Usability testing				✓
2.8 Compatibility testing				✓
2.9 Chapter Summary				✓

Task	Group Members			
	Abhinav	Nirmal	Hasitha	Loshani
Chapter 3: Evaluation				
3.1 Chapter Overview			✓	
3.2 Evaluation methods		✓		
3.3 Quantitative evaluation		✓		
3.4 Qualitative evaluation			✓	
3.5 Self evaluation		✓		
3.6 Chapter Summary			✓	
Chapter 4: Conclusion				
4.1 Chapter Overview		✓		
4.2 Achievements of aims and objectives			✓	
4.3 Limitations of the research	✓			
4.4 Future enhancements	✓			
4.5 Extra work				
4.7 Concluding remarks		✓		

Table 3: Work Breakdown Table.