ABSTRACT

The project aims to develop a sentiment analysis model using machine learning and natural language processing techniques to analyze customer reviews on the website www.cavo.ae. The model will be implemented using Python and the Django framework. The objec6ve is to create a system that can classify the sen6ment of customer reviews as positive, negative, or neutral, thereby providing valuable insights for www.cavo.ae.

Steps:

1. Data Collection: The first step involves collec6ng customer reviews from www.cavo.ae. This can be done by scraping the website or accessing their customer review database. It is important to collect a substantial number of reviews that cover various products, services, and user experiences.

2. Data Preprocessing: Once the reviews are collected, the next step is to preprocess the text data. This includes removing HTML tags, special characters, and irrelevant information. Additionally, techniques like tokeniza6on, removing stop words, and lemmatization can be applied to convert the text into a suitable format for analysis.

3. Feature Extrac6on: In this step, features need to be extracted from the preprocessed text. Similar to the previous descrip6on, techniques such as bag-of-words, TF-IDF, or word embeddings can be used to capture the relevant informa6on from the reviews.

4. Model Training: The preprocessed data with extracted features is divided into training and testing sets. We will them use deep learning models like recurrent neural networks (RNNs) or transformers can be trained on the training set. The models learn to classify the sentiment of the reviews based on the labeled data.

5. Model Integration: Once the sentiment analysis model is trained, it needs to be integrated into a Django web applica6on specifically tailored for www.cavo.ae. The Django framework provides a convenient way to develop a user-friendly interface where users can input their reviews. The web

applica6on will u6lize the trained model to predict the sentiment of the reviews and display the results to the users.

By following these steps, a sentiment analysis system using machine learning and Django can be developed specifically for www.cavo.ae. This system will enable www.cavo.ae to gain valuable insights into customer sentiment, identify areas of improvement, and make data-driven decisions to enhance their products and services

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

The aim of this project sentiment analysis for customer reviews on www.cavo.ae using machine learning and natural language processing to analyze customer reviews on the website www.cavo.ae, an e-commerce platform serving customers in the United Arab Emirates. The model will be implemented using Python and the Django framework. The objec6ve is to create a system that can classify the sen6ment of customer reviews as positive, negative, or neutral, thereby providing valuable insights for www.cavo.ae.

By automatically summarizing reviews, Cavo aims to enhance decision-making processes and improve customer satisfaction. We will collect the reviews from the website and preprocess it if the products get positive review they can can make more products if it gets more negative comments the company have to check for the drawbacks and have to solve it. The main modules covered in this web page are admin module and user module. The admin module work with provide user information and can add or delete users who are not belong to the company. The user module works with user login users are mainly the staffs in the company. The users can login to the site and can view the reviews based on their project and can work according to that.

By successfully developing a sentiment analysis review system using the machine learning and natural language processing, Cavo will be able to derive valuable insights from customer feedback, gain a deeper understanding of their products' strengths and weaknesses, and make informed decisions to enhance their offerings and customer satisfaction.

1.2 PROBLEM DEFINITION

The goal of this project is to create a system that can analyze customer reviews on www.cavo.ae and generate concise and informative summaries for each review.The company does not know about the drawbacks of their products without using a review system from the users they also dont know which of their products get more reviews .By making a sentiment analysis of the review system like this they are able to know about the quality of their products so they can make changes in their products according to the user needs. They can market good product as well.

CHAPTER 2

SYSTEM ANALYSIS

2.1 INTRODUCTION

System Study is a general term that refers to an orderly, structured process for identifying and solving problems. We call system Study process lifecycle methodology, since it relates to four significant phases in the lifecycle of all business information system. The system is studied to the minute details and analyzed. Analysis implies the process of breaking something into parts so that the whole may be understood. The definition of the system analysis includes not only the process of analysis but also that of synthesis, which implies the process of putting together to form a new whole. All activities associated with each life cycle phase must be performed, managed and documented. Hence we define system analysis as the performance, management, documentation of the activities related to the life cycle phases of a computer-based business system. In the study phase a detailed study of the project is made and clear picture of the project is in mind. In the design phase the designing of the input, output and table designs are made. Development phase is where the physical designing of the input-output screens and coding of the system is done. System implementation actually implements the system by making necessary testing

2.2 EXISTING SYSTEM

The cavo shopping site has been doing the the review system manually. It has many drawbacks we have to check all the products one by one each and every time to check whether it has a good review or not . Although it become more complex when we do it manually and also take too much time to do it. So we have introduced this system to make it more easy. There is no such existing system for this review system.

2.2.1 LIMITATIONS OF THE EXISTING SYSTEM

The 90% of the existing system is a physical system. It become more complex to the employees to manually check the review and sort it out based on these reviews. It also takes more time to evaluate all the products and their reviews manually. Manually reviewing products can only cover a limited number of items due to time constraints. This can result in an incomplete representation of the entire product market. Global markets might have products with reviews in multiple

languages, making it difficult for consumers who are not fluent in those languages to access relevant information.

2.3 PROPOSED SYSTEM

In the proposed system it provides many new feature and its avoid the drawbacks of the existing system. Its have more advantages that is very helpful as compared to the existing system. Its overcome the limitations of the existing system and include some new features. It helps to see the products and reviews at the same time and can classifies positive or negative.

he proposed sentiment analysis system empowers businesses with actionable insights derived from customer reviews. By leveraging cutting-edge NLP and machine learning techniques, the system aims to unlock the true potential of customer feedback, fostering improved products, enhanced customer experiences, and sustainable growth.

2.3.1 ADVANTAGES OF THE PROPOSED SYSTEM

• Gain valuable insights into customer opinions and emotions regarding your products.
• Quickly address negative sentiments to prevent issues from escalating and enhance customer satisfaction.
• Compare sentiment across reviews to understand how you stand out against competitors.
• Use sentiments to guide product improvements and create more customer oriented offerings.
• Monitor online reputation by identifying and addressing negative sentiments promptly.
• Adjust marketing strategies based on positive or negative sentiment trends for better engagement.
• Combine sentiment data with other sources to predict consumer behaviour and inform business decisions.

2.4 FEASIBILITY STUDY

This phase implies on the primary job of recognizing the problem. In this stage we define what the problem is and study the various inputs and outputs of the system. Recognizing the demands of the system and clearly defining the system must be the output of this phase of software development life cycle. A feasibility study determines whether the proposed solution is feasible based on the priorities of the requirements of the organization. A feasibility study culminates in a feasibility report that recommends a solution. It helps you to evaluate the cost-effectiveness of a proposed system The feasibility study is carried out to test if the proposed system is worth being implemented. Given unlimited resources and infinite time, all projects are feasible. After performing a Preliminary Investigation, gathering, and interpreting data and details concerning the project, a Feasibility Check is done which involves a series of steps to check the Technical,

Financial and Operational feasibilities

A system that passes the feasibility tests is considered a feasible system. Let us see some feasible tests in my project. The implementation ability is in terms of logistics, resource availability, cost factors and time. We did three types of feasibility study. • Technical Feasibility • Economic Feasibility • Operational Feasibility

2.4.1 TECHNICAL FEASIBILITY

It is related to the software and equipment specified in the design for implementing in this project. There are a number of technical issues, which are generally raised during the feasibility stage of the investigation. A study of function, performance and constraints gave me the ability to achieve acceptable system. The software required for this system is:

The main technical issue raised during feasibility is the existence of necessary technology and whether the proposed equipment has the capacity to hold required data. The technical guarantee of accuracy, reliability, ease and data were also investigated

2.4.2 ECONOMIC FEASIBLITY

Economic analysis is the most commonly used method for evaluating effectiveness of a system. Cost-benefit analysis is the most important assessment of economic justification of the project. Cost –benefit analysis delineates the cost for project development and weighs them against tangible and intangible benefits of a system. This type of analysis varies with the characteristics of the system to be developed, the relative site of the project, and the expected return on investment. Benefits of a new system are always determined relative to the existing mode of operation. Economic feasibility deals about the economic impact faced by the organization to implement the new system. Not only cost of hardware, software etc. is considered but also the form of reduced costs. The project, installed certainly be beneficial since there will be a reduction in manual wok, increase in speed of work. The analysis raises financial and economic questions during the preliminary investigation to estimate the following

The cost to conduct a full systems investigation. The cost of hardware and software for the class of application of the project being considered. To be judged feasible, a proposal for the specific project must pass all these tests, otherwise it is not considered as a feasible project. I gathered the details regarding the financial aspects incorporated in the system to make it cost efficient.

The cost of developing and operating This project will depend on various factors, including technology infrastructure, marketing expenses etc. Based on our analysis, we estimate that the costs for launching this will be approximately Rs 2,00000. which includes salaries for doctors, employees as well as regulatory compliance costs, marketing expenses, and other overhead expenses

2.4.3 OPERATIONAL FEASIBILITY

Operation feasibility is a measure of how people feel about the system. Operational Feasibility criteria measure the urgency of the problem or the acceptability of a solution. Operational

Feasibility is dependent upon determining human resources for the project. It refers to projecting whether the system will operate and be used once it is installed. If the ultimate users are comfortable with the present system and they see no problem with its continuance, then resistance to its operation will be zero. Behaviorally also the proposed system is feasible. A particular application may be technically and but may fail to produce the forecasted benefits, because the company is not able to get it to work. For the system, it is not necessary that the user must be a computer expert, but any computer operator given a little bit of knowledge and training can easily operate. Our Project is operationally feasible since there is no need for special training of staff member and whatever little instructing on this system is required can be done so quite easily and quickly as it is essentially This project is being developed keeping in mind the general people who one have very little knowledge of computer operation, but can easily access their required database and other related information.

2.5 SOFTWARE ENGINEERING PARADIGM

Software paradigm refers to method and steps, which are taken while designing the software programming paradigm is a subset of software design paradigm which is future for other a subset of software development paradigm. Software is considered to be a collection of executable programming code, associated libraries, and documentation. Software development paradigm is also known as software engineering, all the engineering concepts pertaining to developments software applied. It consists of the following parts as Requirement Gathering, Software design, Programming, etc. The software design paradigm is a part of software development. It includes design, maintenance, and programming.

Agile process model

The Agile process model is an iterative and incremental approach to software development that emphasizes flexibility, collaboration, and continuous improvement. It focuses on delivering working software in short iterations called sprints while accommodating changing requirements.


Agile
Development
Methodology
Brainstrom
Requirement Analysis
Design
Design Document & Prototype
Development
Iterations,Demo & Feedback
Quality Assurance
Iterations,Demo & Feedback
Deployment
Production & Technical Support

CHAPTER 3

SYSTEM DESIGN

3.1 INTRODUCTION

System design involves translating information requirements and conceptual design into technical specification and general flow of processing. After the user requirements are identified, related information is gathered to verify the problem and after evaluating the existing system, a new system is proposed. The proposed system consists of various tables and their maintenance to give accurate information for users.

For the design of Sentiment analysis for customer review system for cavo, has given for developing an efficient system, which is user friendly as well as high in performance. It has been assured that the system will have the functions and promises of the proposed system. Design phase acts as a bridge between the software requirement specification and the implementation phase, which satisfies the requirements.

The major step in design is the preparation of input forms and the design of all the major output forms in a manner acceptable to the user in all aspects. The base lies in the complete understanding of the system. The data flow diagrams explicitly specify the process flow. Table design or database design is the next major step. Extreme care has to be given here and several concepts of normalization have to be applied at many levels. Program specification comes next. Here we specify various aspects of the program and also will explain in detail the major components used in the program. The overall process flow is also explain in much detail. Various validation rules and constraints such as data format checking are applied during data input. The companies or seekers that registered to the application are validated by administrator. To ensure security strict checking of password and username had done.

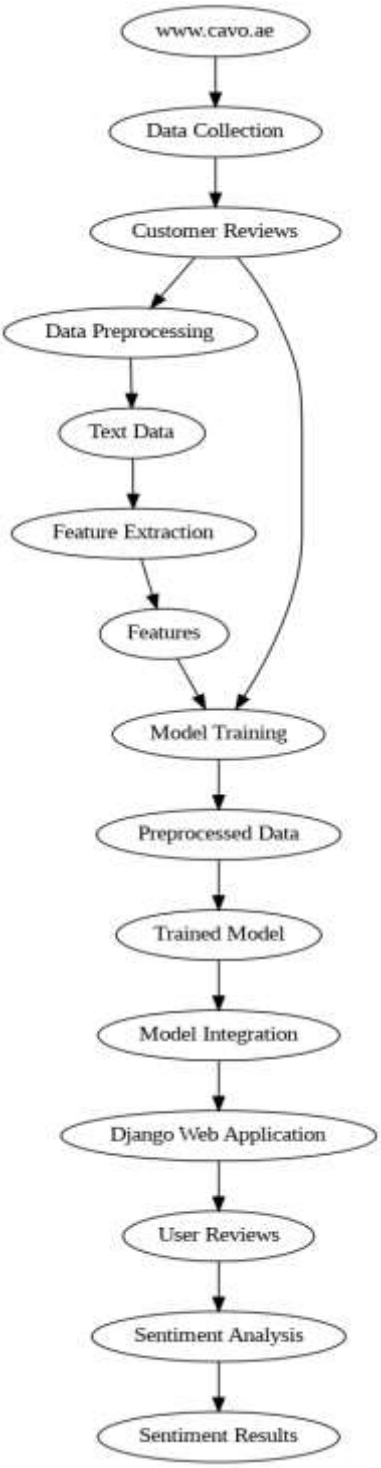
Inputs, outputs have to be designed as per predefined guidelines. Effective and meaningful navigation has to be applied. In the input design, the user-oriented inputs are converted into computer-based formats whereas in the output design, the emphasis is on producing the hard copy or soft copy of the information requested for. Code design is also of much importance. It directly refers to various codes used in the programs and their usage specification. The category to which these codes belong should also be specified.

3.2 DATABASE DESIGN

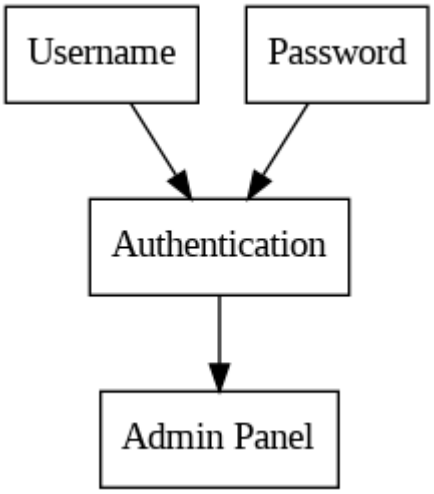
FIELDNAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
username	Char field	Not null	User name
Email address	Text field	Not null	Email address of the user
First name	Char field	Not null	First name of the user
Last name	Char field	Not null	Last name of the user

Field Name	Data Type	Constraints	Description
Document_Name	Char Field	Not Null	Name of the documenet
Documents_file	Byte	Not Null	File

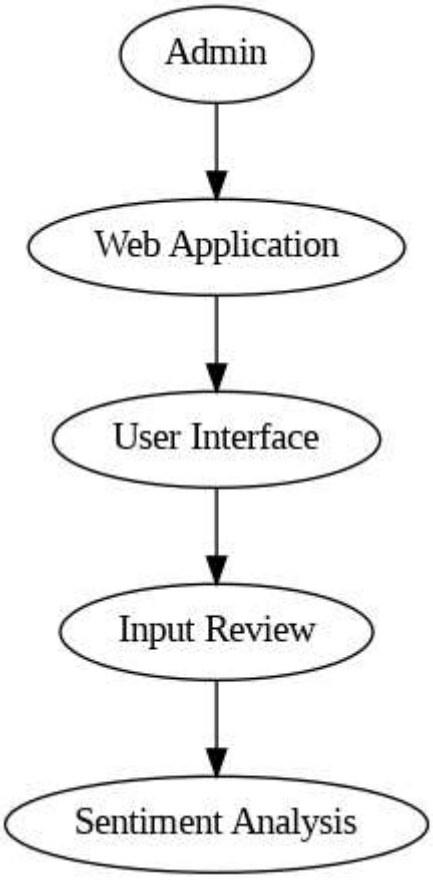
2.3 PROCESS DESIGN-DATAFLOW DIAGRAMS



Data Flow Diagram of the Web Application

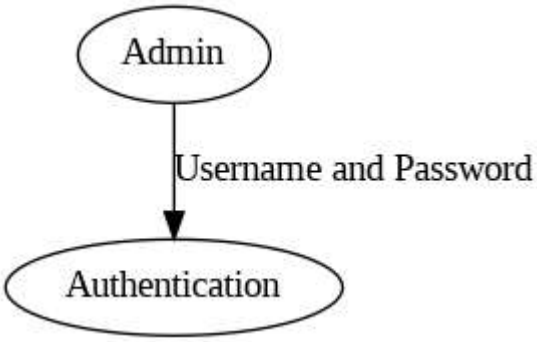


Data Flow Diagram Of the Sentiment Analysis

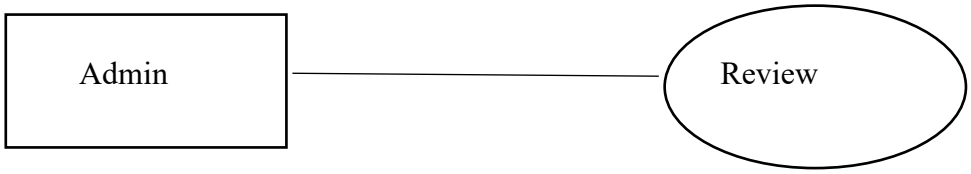


Entity Relationship Diagram of the Entire Project

Entity Relationship Diagram Of the Web Application



Entity Relationship Diagram Of the Web Application



CHAPTER 4

SYSTEM ENVIRONMENT

4.1 INTRODUCTION

The digital age has revolutionized the way consumers interact with products and services, leading to an exponential rise in online shopping platforms like Amazon.ae. To maintain a competitive edge, these platforms must continually enhance their user experience, especially in the realm of customer reviews. In the proposed system we are doing computerized review summarization. Thus we can overcome the drawbacks which caused by the manual working. The proposed system is more efficient than the existing manual mechanism.it is more comfortable and easier than the existing system

4.2 SOFTWARE REQUIREMENTS

- Platform : VS Code
- Operating System : Microsoft Windows
- Front End : HTML,CSS, JavaScript,Bootstrap
- Back End : Python
- Database : SQLlite3
- Developing Tools : Python Django,VS Code

4.3 HARDWARE REQUIREMENTS

Processor : Intel Pentium Dual Core/ above
Hard Disk Space : 512 GB

RAM : 8 GB

Display : 14.1 “Colour Monitor(LCD,CRT or LED)

4.4 TOOLS, PLATFORMS

4.4.1 FRONT END TOOL

HTML

HTML stands for Hypertext Markup Language, was invented by Tim Burners Lee. It is a simple text formatting language used to create hypertext documents. It is a platform independent language unlike most other programming languages. HTML is neutral and can be used on any platform or desktop. It is this feature of HTML that makes it popular as standard on the WWW. This versatile language allows the creation of hypertext links, also known as hyperlinks. The language used to develop web pages is called Hyper Text Markup Language(HTML). HTML is the language interpreted by a browser. HTML is specified as TAGS in an HTML document (i.e. the web page).

HTML Tags

Tags are instructions that are embedded directly into the text of the document. An HTML tag is a signal to a browser that it should do something other than just throw text up on the screen. By convention all HTML tags begin with an open angle bracket (<) and end with a close angle bracket (>).

The structure of an HTML program

Every HTML program has a rigid structure. The entire web page is enclosed within <HTML><HTML> tags. Within these tags two distinct sections are created using the <HEAD></HEAD> tags and the <BODY></BODY> tags.

CSS

Cascading Style Sheet is a style sheet language used for describing the presentation of a document written in a markup language, Although most often used to set the visual style of web page and user interfaces written in HTML and XHTML, the language can be applied to any XML document, including plain XML, SVG and XUL, and is applicable to rendering in speech, or on other media.

Along with HTML and JavaScript, CSS is a cornerstone technology used by most websites to create visually engaging webpages, user interfaces for web applications, and user interfaces for many mobile applications. CSS stands for Cascading Style Sheets. It is the language for describing the presentation of Web pages, including colors, layout, and fonts, thus making our web pages presentable to the users.

CSS Syntax

Selector {

Property 1 : value;

Property 2 : value;

Property 3 : value;

}

Selector: selects the element you want to target

Always remains the same whether we apply internal or external styling

There are few basic selectors like tags, id's, and classes

All forms this key-value pair

Keys: properties(attributes) like color, font-size, background, width, height,etc Value: values associated with these properties

BOOTSTRAP

Bootstrap is a popular open-source front-end framework developed by Twitter. It facilitates the development of responsive and mobile-first websites and web applications. Bootstrap includes a set of HTML, CSS, and JavaScript components, as well as pre-designed templates and themes, making it easier to create consistent and visually appealing user interfaces

Key features of Bootstrap include:

1.Responsive Design: Bootstrap is designed to provide a consistent and optimal user experience across various devices and screen sizes, from desktops to tablets and smartphones.

2. Grid System: The grid system in Bootstrap allows developers to create responsive layouts easily. It is based on a 12-column grid, which can be customized to accommodate different designs.

3. CSS Components: Bootstrap offers a wide range of CSS components like buttons, navigation bars, forms, tables, alerts, and much more, which can be easily customized and integrated into web pages.

4. JavaScript Plugins: Bootstrap comes with various JavaScript plugins that enhance the functionality of the components. Examples include carousels, modals, dropdowns, tooltips, and popovers.

5. Themes and Templates: Bootstrap provides a collection of professionally designed themes and templates that can be used as a starting point for website development.

6. Customizable: While Bootstrap offers pre-styled components, it is highly customizable, allowing developers to adjust styles and layouts to match the specific needs of their projects.

To get started with Bootstrap, you can either download the framework from the official website (getbootstrap.com) or include it in your project using a Content Delivery Network (CDN). Once integrated, you can use the provided CSS classes and JavaScript components to build responsive and visually appealing web pages.

JAVASCRIPT

JavaScript is a versatile programming language commonly used for creating interactive and dynamic elements on websites and web applications. When it comes to project reports, JavaScript

can be utilized to enhance the user experience, provide data visualization, and make the report more engaging and interactive. Here are some ways JavaScript can be used in project reports:

1.Data Visualization: JavaScript libraries like D3.js, Chart.js, and Highcharts can be employed to create interactive charts, graphs, and data visualizations. These visual representations make it easier for readers to understand complex data and trends, making your project report more compelling.

2.Dynamic Content: JavaScript can be used to dynamically load content into your report. This can include dynamically updating tables, lists, or sections of the report based on user interactions or changes in the underlying data.

3.Interactivity: JavaScript enables you to add interactive elements such as dropdown menus, sliders, checkboxes, and buttons. These interactive features allow readers to customize the report and explore different aspects of the project data according to their preferences.

4.Form Validation: If your project report contains web-based forms for data collection or feedback, JavaScript can be used to perform form validation on the client-side. This helps ensure that users input valid and properly formatted data, reducing errors and improving data accuracy.

5.Enhancing User Experience: JavaScript can be leveraged to create smooth and user-friendly interactions. For instance, you can add animations, tooltips, and pop-up dialogs to improve the overall experience of reading and navigating the project report.

6.Filtering and Sorting: In reports with large datasets, JavaScript can be used to implement filtering and sorting functionalities. This allows readers to focus on specific subsets of data or arrange the information according to their requirements.

7.Real-time Updates: If your project report involves real-time data, JavaScript can be used to fetch and display data from APIs or other data sources in real-time, keeping the report up-to-date at all times.

8.Print and Export Options: JavaScript can also be used to provide users with the option to print or export the project report to different formats, such as PDF or Excel, for offline access or sharing purposes.

4.4.2 BACK END

PYTHON

Python is a high-level, interpreted, and general-purpose programming language known for its simplicity, readability, and versatility. Created by Guido van Rossum in the late 1980s and first released in 1991, Python has since gained widespread popularity and has become one of the most popular programming languages in the world.

Key characteristics of Python include:

1. Readable and Expressive Syntax:

Python's syntax emphasizes readability and a clean, straightforward code structure. Its code is often described as being close to natural language, making it easy for developers to write and understand code.

2. Interpreted and Interactive:

Python is an interpreted language, which means the code is executed line-by-line by an interpreter without the need for a separate compilation step. This allows for rapid development and experimentation, making Python well-suited for interactive programming and prototyping.

3. Cross-Platform Compatibility:

Python is a cross-platform language, meaning the same Python code can run on various operating systems (e.g., Windows, macOS, Linux) without modification, making it highly portable.

4. Large Standard Library:

Python comes with a comprehensive standard library that provides a wide range of pre-built modules and functions for common tasks, making it easier to develop complex applications without having to reinvent the wheel.

5. Third-Party Libraries and Packages:

Python has a vast ecosystem of third-party libraries and packages available through the Python Package Index (PyPI). These packages provide additional functionality and tools, enabling developers to extend Python's capabilities and solve various domain-specific problems.

6. Object-Oriented Programming (OOP) Support:

Python supports object-oriented programming, allowing developers to create classes and objects, encapsulate data and behavior, and promote code reusability.

7. Dynamic Typing:

Python is dynamically typed, meaning the type of a variable is determined at runtime. This flexibility allows for more concise code but requires careful attention to type handling.

8. High-Level Data Structures:

Python includes built-in high-level data structures like lists, tuples, dictionaries, and sets, which make it easy to work with complex data and manipulate collections of elements.

9. Extensibility and Embeddability:

Python can be easily extended with modules written in other languages (e.g., C, C++, and Java), and it can also be embedded within other applications as a scripting language.

10. Community and Support:

Python has a large and active community of developers who contribute to its growth and continuously improve the language. This community support makes it easier to find solutions to problems, learn from others, and collaborate on projects.

Python is widely used in various domains, including web development, scientific computing, data analysis, machine learning, artificial intelligence, scripting, automation, and more. Its versatility and user-friendly nature make it an excellent choice for both beginners and experienced developers alike.

DEVELOPING TOOLS

PYTHON DJANGO

Python Django is a high-level web framework that allows developers to build web applications rapidly and efficiently. It follows the Model-View-Controller (MVC) architectural pattern, which is often referred to as Model-View-Template (MVT) in the context of Django. Django is open-source, maintained by the Django Software Foundation, and has a strong and active community of developers.

Key features of Django include:

1. Object-Relational Mapping (ORM): Django provides an ORM system that allows developers to interact with the database using Python objects instead of writing raw SQL queries. This simplifies database management and promotes code reusability.

2. URL Routing: Django uses a clean and flexible URL routing system, enabling developers to map URLs to specific views or functions in their code, making it easy to organize and handle different parts of the application.

3. Templates: Django comes with a built-in template engine that enables developers to define the presentation layer of their web application separately from the application's logic. This makes it easier to maintain and modify the user interface.

4. Forms: Django includes form handling functionality, which simplifies the process of creating and processing HTML forms, including data validation and security.

5. Authentication and Authorization: Django provides robust authentication and authorization mechanisms, allowing developers to manage user authentication, permissions, and access control efficiently.

6. Admin Interface: Django automatically generates an admin interface based on the models defined in the application. This interface makes it easy to manage and manipulate data in the database without having to write custom admin panels.

7. Security: Django emphasizes security best practices, such as protection against common web application vulnerabilities like cross-site scripting (XSS), cross-site request forgery (CSRF), and SQL injection.

8. Internationalization and Localization: Django has built-in support for handling multiple languages and localizing web applications, making it easier to reach a global audience.

9. Scalability: Django is designed to handle high-traffic websites and can be easily scaled to accommodate increased demands.

10. Third-party Packages: Django has a rich ecosystem of third-party packages and extensions, which can be easily integrated into your project to extend its functionality.

To get started with Django, you typically install it using pip, create a Django project, define models for your data, create views to handle HTTP requests, and design templates to display the content. Django follows the principle of "Don't Repeat Yourself" (DRY) and aims to promote clean, maintainable, and reusable code.

VS Code

Visual Studio Code (VS Code) is a lightweight yet robust source code editor developed by Microsoft. It's a cross-platform tool that provides a streamlined interface for code creation and

modification. With a user-friendly UI, it offers essential features like syntax highlighting, auto-indentation, and code folding, enhancing code readability and productivity.

One of its defining strengths is its extensibility through a vast library of extensions available in the Visual Studio Code Marketplace. These extensions enrich the editor's functionality by adding language support, debugging tools, and integration with version control systems like Git.

VS Code comes with an integrated terminal that facilitates command-line interactions without leaving the editor. Its debugging capabilities simplify the identification and resolution of code issues through breakpoint setting, variable inspection, and step-through functionality.

IntelliSense, another standout feature, offers intelligent code completion and suggestions based on context, minimizing errors and accelerating coding. Task automation and snippet support further expedite development tasks. The editor also supports collaborative coding via Live Share, allowing real-time collaboration among distributed developers.

CHAPTER 5

SYSTEM IMPLEMENTATION

5.1 INTRODUCTION

Implementation involves placing the complete and tested system software into actual work environment. Implementation is concerned with translating design specification with source code. The primary goal of implementation is to write the source code to its specification can easily be verified, and so that debugging, testing and modification can be eased. The goal can be achieved by making the source code clear and straight forward as possible. Implementation means the process of converting a new or revised system design into operational one. The three types of implementation are

- Implementation of a complete system to replace a manual system

- Implementation of a new system to replace existing one

- Implementation of a modified application to replace an existing one

IMPLEMENTATION OF THE PROJECT

Implementation is the process of personnel check out, install the required equipment and application and train user accordingly. Depending on the size of the organization and its requirements the implementation is divided into three parts

• Stage Implementation

Here system is implemented in stages. The whole system is not implemented at once. Once the user starts working with system and is familiar with it, then a stage is introduced and implemented. Also the system is usually updated, regularly until a final system is sealed

• Direct Implementation

The proposed new system is implemented directly and the user starts working on the new System. The shortcoming, if any, faced are then rectified later. Parallel Implementation

5.2 CODING

Coding is the software activity where the detailed design specification is implemented as source code. Coding is the lowest level of abstraction for the software development process. It is the last stage in decomposition of the software requirements where module specifications are translated into a programming language.

Typical tasks for Coding:

• Traceability analyses

• Source Code to Design Specification(and vice versa)

• Source Code and Source Code Document Evaluation

• Source Code Interface Analysis

5.2.1 CODING STANDARADS

In the realm of software development, adhering to established coding standards is paramount to the success of a project. Coding standards serve as a set of guidelines that govern the structure, style, and organization of code, fostering consistency, readability, and maintainability. This section introduces the importance of coding standards in our project documentation, highlighting their role in enhancing collaboration, minimizing errors, and promoting best practices.

Good software development organizations want their programmers to maintain to some well-defined and standard style of coding called coding standards. They usually make their own coding standards and guidelines depending on what suits their organization best and based on the types of software they develop. It is very important for the programmers to maintain the coding standards otherwise the code will be rejected during code review.

• Purpose of Having Coding Standards:

• A coding standard gives a uniform appearance to the codes written by different engineers.

- It improves readability, and maintainability of the code and it reduces complexity also.
- It helps in code reuse and helps to detect error easily.
- It promotes sound programming practices and increases efficiency of the programmers.
- Some of the coding standards are given below:
- Limited use of global: These rules tell about which types of data that can be declared global and the data that can't be.
- Standard headers for different modules: For better understanding and maintenance of the code, the header of different modules should follow some standard information.
- Name of the module
- Date of module creation
- Modification history
- Synopsis of the module about what the module does
- Different functions supported in the module along with their input output parameters
- Naming conventions for local variables, global variables, constants and functions: Some of the naming conventions are given below:
- Meaningful and understandable variables name help to anyone to understand.

Proper indentation is very important to increase the readability of the code. For making the code readable, programmers should use White spaces properly. Some of the spacing conventions are given below:

- There must be a space after giving a comma between two function arguments.
- Each nested block should be properly indented and spaced.
- Proper Indentation should be there at the beginning and at the end of each block in the program.
- All braces should start from a new line and the code following the end of braces also start from a new line.
- Error return values and exception handling conventions:
- All functions that encountering an error condition should either return a 0 or 1 for simplifying the debugging.
- On the other hand, Coding guidelines give some general suggestions regarding the coding style that to be followed for the betterment of understandability and readability of the code. Some of the coding guidelines are given below

- Not used a coding style that is too difficult to understand:
- Code should be easily understandable. The complex code makes maintenance and debugging difficult and expensive. Avoided using an

identifier for multiple purposes:

- Each variable should be given a descriptive and meaningful name indicating the reason behind using it. This is not possible if an identifier used for multiple purposes and thus it can lead to confusion to the reader. Moreover, it leads to more difficulty during future enhancements.
- Code should be well documented:

The code should be properly commented for understanding easily. Comments regarding the statements increase the understandability of the code. Length of functions should not be very large

5.2.2 SAMPLE CODES

ALGORITHM

```
import pandas as pd

from transformers import AutoTokenizer, AutoModelForSequenceClassification



# Load the tokenizer and model

tokenizer = AutoTokenizer.from_pretrained("MarieAngeA13/Sentiment-Analysis-BERT")

model = AutoModelForSequenceClassification.from_pretrained("MarieAngeA13/Sentiment-Analysis-BERT")



# Read the CSV file
```

```
df = pd.read_csv('updated_reviews_3000.csv')


# Define a mapping from sentiment scores to sentiment labels

sentiment_mapping = {0: 'Negative', 1: 'Neutral', 2: 'Positive'}


# Create a new column with the sentiment

df['sentiment'] = df['body'].apply(lambda x: sentiment_mapping[model(tokenizer.encode(x, return_tensors='pt'))[0].argmax().item()])


# Save the updated DataFrame to a new CSV file

df.to_csv('updated_reviews_3000_with_sentiment.csv', index=False)
```

Explanation

- We're using two special tools: pandas (for handling data) and Hugging Face Transformers (for understanding text).
- Preparing the Sentiment Analyzer: We have a pre-trained model that can tell us if text sounds negative, neutral, or positive. ,We also have a tool (tokenizer) to make the text understandable for the model.
- Reading Data: •We've collected reviews and put them in a spreadsheet (CSV file),We're going to read this spreadsheet and organize it using pandas.
- Labeling Emotions: We've defined a system to understand the emotions: 0 means "Negative", 1 means "Neutral", and 2 means "Positive".

- 5. Understanding Reviews: We're going to take each review (piece of writing) and try to figure out if it's negative, neutral, or positive., We'll use the model and the emotion labels we defined. 6. Figuring Out Emotion: , For each review, we'll follow these steps: , Change the review into a form the model understands. , Ask the model to guess the emotion. ,Find out which emotion the model guessed,We'll note this guessed emotion for each review.
- Updating the Spreadsheet: We'll add a new column to our spreadsheet called "sentiment". This column will show the guessed emotion for each review. 8. Saving Our Work: After we've guessed the emotions for all reviews and added the info to the spreadsheet, we'll save it. •We'll save it in a new spreadsheet so that we don't mess up the original one.

```
import pandas as pd

from transformers import AutoTokenizer, AutoModelForSequenceClassification



# Load the tokenizer and model

tokenizer = AutoTokenizer.from_pretrained("MarieAngeA13/Sentiment-Analysis-BERT")

model = AutoModelForSequenceClassification.from_pretrained("MarieAngeA13/Sentiment-Analysis-BERT")



# Read the CSV file

df = pd.read_csv('cavo-reviews.csv')



# Define a mapping from sentiment scores to sentiment labels

sentiment_mapping = {0: 'Negative', 1: 'Neutral', 2: 'Positive'}
```

```
# Create a new column with the sentiment
df['sentiment'] = df['body'].apply(lambda x: sentiment_mapping[model(tokenizer.encode(x, return_tensors='pt'))[0].argmax().item()])


# Function to get the sentiment distribution for a product
def get_sentiment_distribution(product_name):
    product_reviews = df[df['product_handle'] == product_name]
    sentiment_counts = product_reviews['sentiment'].value_counts().to_dict()
    average_sentiment = product_reviews['sentiment'].mode()[0]

    print(f"Product '{product_name}' reviews:")
    for sentiment, count in sentiment_counts.items():
        print(f"{count} {sentiment}")
    print(f"==========> Average: {average_sentiment}")


# Keep asking for product names until 'exit' is typed
while True:
    product_name = input("Enter the product name (or 'exit' to stop): ")
```

```
    if product_name.lower() == 'exit':
        break
    get_sentiment_distribution(product_name)


# Save the updated DataFrame to a new CSV file
df.to_csv('updated_reviews_3000_with_sentiment.csv', index=False)
#parallel computing for speed optimization
```

Explanation

- We're using two special tools: pandas (for handling data) and a smart language understanding tool called Transformers.
- Preparing the Sentiment Analyzer: • We've picked a tool that can understand if reviews are negative, neutral, or positive. • We also got a helper tool (tokenizer) to make the reviews understandable for the sentiment tool.
- Reading Review Data: • We have a list of reviews and the product they're about. • We're reading this list and organizing it neatly using pandas. 4. Emotion Labels: • We've decided to call negative reviews "Negative", neutral ones "Neutral", and positive ones "Positive". • We'll use numbers (0, 1, 2) for the computer to understand, but we'll translate them to these labels later.
- Understanding Review Emotions: • For each review, we're going to use the sentiment tool to guess if it's negative, neutral, or positive. • We'll translate the computer's guess into understandable labels using the numbers we assigned earlier.
  Adding Emotion Info: • We're going to make a new column in our review list. This column will tell us what emotions the sentiment tool guessed for each review.

- Analyzing Product Emotions: • We made a way to check the emotions of reviews for a specific product. • We'll calculate how many reviews have each emotion and what's the most common emotion for the product. 8. Asking User's Opinion: • We'll keep asking the user to type a product name. • We'll show the emotions of reviews for that product and the most common emotion. 9. Saving Our Work:

```
import pandas as pd

from transformers import AutoTokenizer, AutoModelForSequenceClassification

from multiprocessing import Pool, cpu_count


# Load the tokenizer and model

tokenizer = AutoTokenizer.from_pretrained("MarieAngeA13/Sentiment-Analysis-BERT")

model = AutoModelForSequenceClassification.from_pretrained("MarieAngeA13/Sentiment-Analysis-BERT")


# Read the CSV file

df = pd.read_csv('cavo-reviews.csv')


# Define a mapping from sentiment scores to sentiment labels

sentiment_mapping = {0: 'Negative', 1: 'Neutral', 2: 'Positive'}
```

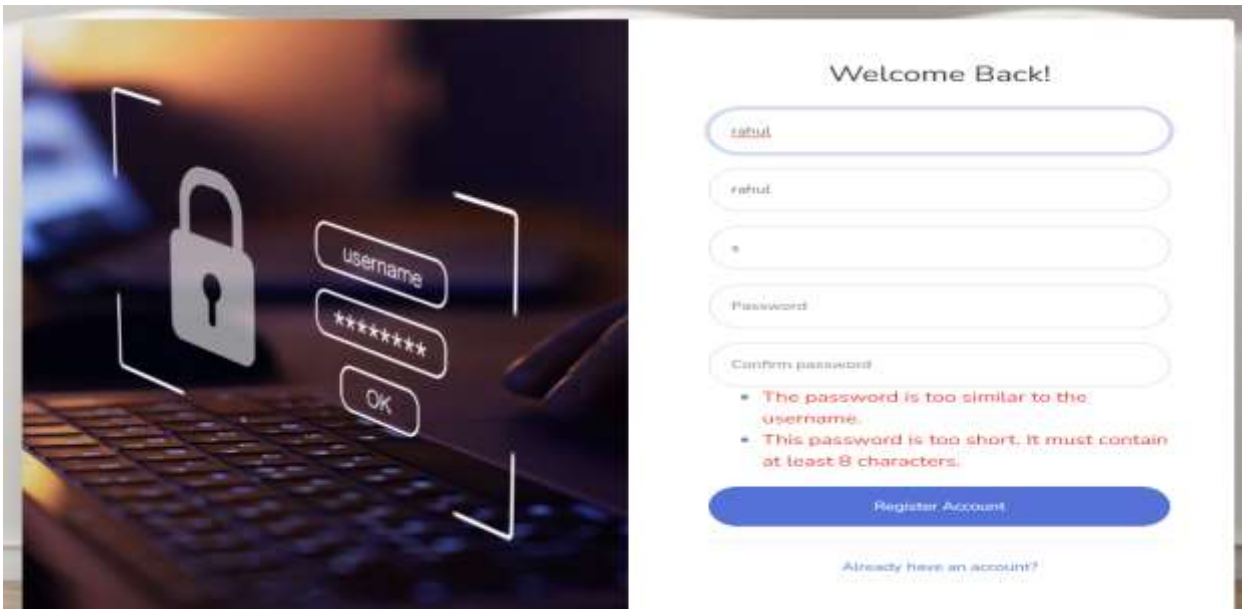
```
# Process reviews in batches
def batch_predict(sentences):
    inputs = tokenizer(sentences, padding=True, truncation=True, return_tensors='pt', max_length=128) # limit to 128 tokens
    outputs = model(**inputs)
    _, preds = outputs.logits.max(dim=1)
    return preds.tolist()

# Parallelize the processing using multi-processing
def parallel_predict(sentences, batch_size=32):
    with Pool(cpu_count()) as pool:
        chunks = [sentences[i:i+batch_size] for i in range(0, len(sentences), batch_size)]
        results = pool.map(batch_predict, chunks)
        sentiments = [item for sublist in results for item in sublist]
    return sentiments

df['sentiment'] = parallel_predict(df['body'].tolist())
df['sentiment'] = df['sentiment'].map(sentiment_mapping)
```

```
# Function to get the sentiment distribution for a product
def get_sentiment_distribution(product_name):
    product_reviews = df[df['product_handle'] == product_name]
    sentiment_counts = product_reviews['sentiment'].value_counts().to_dict()
    average_sentiment = product_reviews['sentiment'].mode()[0]

    print(f"Product '{product_name}' reviews:")
    for sentiment, count in sentiment_counts.items():
        print(f"{count} {sentiment}")
    print(f"==========> Average: {average_sentiment}")

# Keep asking for product names until 'exit' is typed
while True:
    product_name = input("Enter the product name (or 'exit' to stop): ")
    if product_name.lower() == 'exit':
        break
    get_sentiment_distribution(product_name)
```

Explanation

- Getting Ready: • We have special tools: pandas (to handle data), Transformers (to understand text), and multiprocessing (to make things faster).
- Preparing Sentiment Analysis: • We got a tool that can guess if reviews are negative, neutral, or positive. • We also have a helper tool to make reviews understandable for this sentiment tool.
- Reading Reviews: • We have a list of reviews and products. • We're reading this list to make sense of it using pandas.
- Emotion Labels: • We decided to call negative reviews "Negative," neutral ones "Neutral," and positive ones "Positive." • We'll use numbers for the computer, but we'll translate them to these labels later.
- Understanding Emotions in Batches: • We'll take groups of reviews and ask the sentiment tool to guess their emotions. • We'll do this in parts to make it faster.
- Faster Processing with Many Brains: • We'll use multiple computer cores to work on different groups of reviews at the same time. • This way, things get done much faster!
- Analyzing Emotions: • We'll use the sentiment tool on all reviews and translate the computer's guesses into words (like "Negative"). • This will help us understand the feelings in the reviews.
- Finding Emotions for Products: • We made a way to check the emotions for reviews about a specific product. • We'll count how many reviews have each emotion and find the most common emotion.
- Asking for Products: • We'll ask you for a product's name. • We'll show you how people feel about that product based on their reviews.
- Saving Our Work: • After we finish understanding the reviews and emotions, we'll save our organized information. • This way, we can use it again without starting over

5.2.3 CODE VALIDATION AND OPTIMIZATION



Signup Page

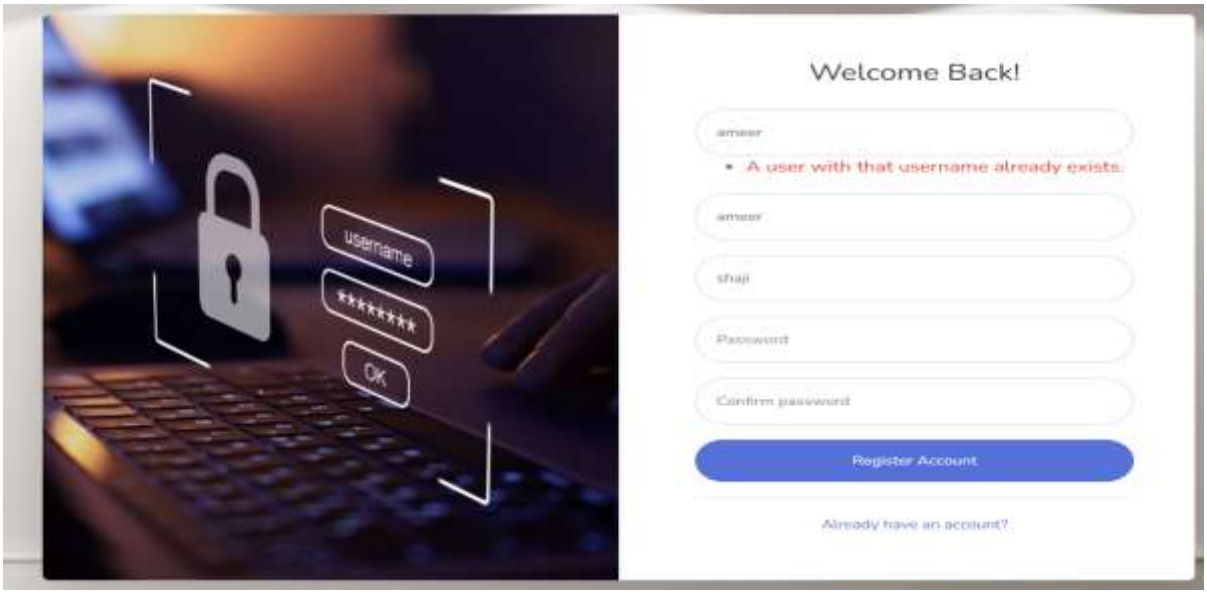


Fig: Validation

Fig: Validation in File upload Page

CHAPTER 6

SYSTEM TESTING

6.1 INTRODUCTION

The purpose of system testing is to identify and correct errors in the candidate system. Testing is an important element of the software quality assurance and represents the ultimate review of specification, design and coding. The increasing visibility of the software as a system element and the costs associated with a software failure are motivated forces for well planned, through testing. During testing the system is used experimentally to ensure that the software does not fail, i.e., it will run according to the specification and in the way the user expects. Preparation of test data plays n vital role in the system testing. Different set of test data are generated and the system under study is tested using that data. While testing using test data errors are again uncovered and corrected using different testing techniques. System testing was conducted in order to detect errors and for comparing then the final system with the requirement specification report. That is, whether the system meets requirements. During testing the software was executed with a set of test cases and the output ofthe program for the test cases was evaluated to determine if the program is performing as it was expected to. Testing presents, an interesting challenge for the software engineer attends to hold software from an abstract concept to an acceptable implementation. In testing engineer creates a series of test cases that occurs when errors are un covered uncovered. Testing is the process of Executing a program for finding errors. A good test is one that has high probability of finding an uncovered error The turn error is used to refer the difference between the actual output of the software and the correct output. Fault is a condition that causes the software to fail to perform its required function. Software reliability is defined as the required function. Testing principles

Many Of my Friends suggested a set of testing principles in 1995. These are:

1) All tests should be traceable to the customer requirements.

2) Test should be planned long before testing begins.

3) Testing should begin in small and progressed towards testing in the large.

4) Exhaustive testing is not possible.

5) For effective results, testing should be conducted by an independent third party. Attributes of a good test

A good test should have the following characteristics:

1) A good test must have a high probability of finding an error.

2) A good test is non redundant.

3) A good test should be best of breed i.e. testing should be in time and according to resource limitation.

4) A good test should be neither too simple nor too complex. Sometimes it is possible to combine a series of tests into one test.

Unit Testing

The purpose of this phase is to test the individual units of the developing software component. This phase is recursive and is to be repeated, as many as there are, levels of testing. In the Diagnostic System project, each individual form has been tested using techniques of testing namely: Client-side testing. Each individual form has been validated so that user enters only valid data at every time.

Integration Testing

Integration testing is the second level of the software testing process comes after unit testing. In this testing, units or individual components of the software are tested in a group. The focus of the integration testing level is to expose defects at the time of interaction between integrated components or units.

In This Project, The Registration and Login Modules are Integrated and test Both and The test result will show some Routing Issues, after Finding This issue it will solved.

System testing

is done when the entire system has been fully integrated. The purpose of the system testing is to test how the different modules interact with each other and whether the entire system provides the functionality that was expected

6.2 INTEGRATION TESTING

This testing level can be simply defined as integrating and then testing. i.e., here, many unit tested modules are combined into subsystems, which are then tested. Integration testing aims at whether the modules can be integrated properly. Hence, the emphasis is on testing interfaces between modules. This testing activity can be considered testing the design

USER INTERFACE TESTING

This tests the whole end to end scenario (which simulates exactly what the user would be doing). This (for the most part) goes through all the mouse clicks and keyboard presses the user goes through to get an action done. If we extend the previous example this would be invoking the same action except that this time instead of using any test hooks it would be using the keyboard shortcut and instead verify the actual displayed results on the application. User Interface testing is a process to test Real Estate user interface and to detect if application is functionally correct. USER INTERFACE testing involves carrying set of tasks and comparing the result of same with the expected output and ability to repeat same set of tasks multiple times with different data input and same level of accuracy. USER INTERFACE Testing includes how the Real Estate handles keyboard and mouse events, how different GUI components like menu bars, toolbars, dialogs, buttons, edit fields, list controls, images etc. reacts to user input and whether or not it performs in the desired manner. Implementing USER INTERFACE testing for your application early in the software development cycle speeds up development, improves quality and reduces risks towards the end of the cycle. USER INTERFACE Testing can be performed both manually with a human tester or could be performed automatically with use of a software program

TEST CASES

A test case is a document that describes an input, action or event and an expected response, to determine if a feature of an application is working correctly. A test case should contain particulars such as test case identifier, test case name, test conditions, input data requirements steps, and expected results.

Test Case

Project Title: Amazon.ae review System For Cavo

Software Tool: Python

Test objective: To check whether the entered User name and Password are valid or invalid.

Test data: User name = admin and password = admin

Step no:	**Steps**	**Data**	**Expected results**	**Actual results**
1	Enter user name and press login button	User name= admin	Should display warning message box "Please Enter Password"	Login failed
2	Enter password and press login button	Password=admin	Should display warning message box "Please Enter User name “	Login failed
3	Enter username	User= admin and	Should display warning	Login failed

	password and press login button	Password = XYZ	message box "Invalid User nameor Password"	
4	Enter User name and Password and press LOGIN Button	User= XYZ and Password = admin	Should display warning message box "Invalid User nameor Password"	Login failed
5	Enter User name and Password and press LOGIN Button	User= xyz and Password = xyz	Should display warning message box "Invalid User name or Password"	Login failed
6	Enter User name and Password and press LOGIN Button	User= “ ” and Password = “ ”	Should display warningmessage box "Please EnterUser name and Password“	Login failed
7	Enter User name and Password and press LOGIN Button	User=admin and password=admin	Should navigate to home page	Login made

CHAPTER 7

SYSTEM MAINTENANCE

7.1 INTRODUCTION

implementation is the stage of the project when the theoretical design is turned into a working system. The implementation stage is a systems project in its own right. It includes careful planning, investigation of current system and its constraints on implementation, design of methods to achieve the changeover, training of the staff in the changeover procedure and evaluation of changeover method.

The first task in implementation is planning- deciding on the methods and time-scale to be adopted. Once the planning has been completed, the major effort is to ensure that the programs in the system are working properly. At the same time concentrate on training the staff. When the staffs have been trained, the complete system, involving both computer and user can be executed effectively.

When the Manager's system is linked to terminals on remote sites, the telecommunication network and tests of the network along with the system are also included under implementation. Depending upon the nature of the system, extensive user training may be required. Programming itself is a design work. The initial parameters of the management information system should be modified as a result of programming efforts; programming provides a Reality test for the assumptions made by the analyst.

System testing check the readiness and accuracy of the system access update and retrieve data from new files. Once the program becomes available, the test data are read into the computer and processed. In this system, conventional Parallel Run was conducted to establish the efficiency of the system.

Implementation is used here to mean the process of converting a new or a revised system design into an operational one. Conversion is one aspect of Implementation. Conversion means changing from one system to another. The objective is to put the tested system into operation while holding costs, risks and personal irritation to a minimum.

Data from one or more previous periods for the whole or part of the system is run on the new system after results have been obtained from the old system and both are compared. It is performed till the completion of one system life cycle.

When the changeover has taken place there will be a need for amendment to correct or improve the new system. When the user wants to add any new records, some fields will automatically get their default values. If the user desires to change these default values he can do it.

7.2 MAINTENANCE

Maintenance activity may require the continuing involvement of a large proportion of computer department resources. For computer installations, which have already developed the basic applications for the organization, the main task may be to adapt existing system in a changing environment. Perhaps a better term to describe this activity is system evolution. All systems are dynamic and subject to constantly changing requirements. Efforts must be devoted to adapting them and design should be flexibly specified so that such changes are easily implemented. Most changes arise in two ways. As part of the normal running of the system when errors are found, users ask for improvement or external requirements change and as result of specific investigation and review of the system's performance.

Maintenance is enigma of the system development. Analysts and programmers spend far more time maintaining programs and packages than writing them. Maintenance accounts for 60-80% of the total system development cost incurred. The problems in maintenance occur largely because software is handmade product designed in adhoc fashion with a few standards. Poor documentation makes maintenance of the programs even more difficult. The more carefully is the system thought out and developed, with attention paid to external influence over reasonable lifetime, the less is the maintenance required.

7.2.1 TYPES OF MAINTENANCE

Maintenance has been classified as :

Corrective Maintenance

This means repairing the processing failures or making changes because of previously uncorrected problems or false assumptions.

Adaptive Maintenance

This means changing the program functions as and when required.

Perfective Maintenance

This means enhancing the performance or modifying the programs to respond to the user's additional or time to time changing needs.

The project work involves the complete requirements specification and the description of the project itself. The coding done using Oracle is the total coding work done in order to make the application program a useful project

CHAPTER 8

SYSTEM SECURITY MEASURES

8.1 DATABASE / DATA SECURITY

I have used own username and password. I have changed administrator password according to my password, so that no one can enter my internal parts of SQL Database.

Database security refers to the range of tools, controls, and measures designed to establish and preserve database confidentiality, integrity, and availability. This article will focus primarily on confidentiality since it's the element that's compromised in most data breaches.

Database security must address and protect the following:

• The data in the database

• Any associated applications

• The physical database server and/or the virtual database server and the underlying hardware

• The computing and/or network infrastructure used to access the database **Hide Internal Schema:**

Throughout the project the internal schema are hidden. The System provided comprehensive and multi-level security. The application allows the user to key in their User Name and their Password. It keeps data secure from unauthorised access, modification or reporting by allowing you to create as many 'users' as you wish

CREATION OF USER PROFILES AND ACCESS RIGHTS

Hierarchical User Roles:

Another level of Security will be established by setting up of various User-Roles. Every user will have their roles specified and hence the accessibility of Information will base on their User-Role and Hierarchical position. It is well customizable, but we have set our system for the following user-roles. Each these individuals have user name and password, and after an authorize Login, they get only their concern information, which is set by the Management

Application Based Security System:

Automation and web based information broadcasting System has been provided with comprehensive and multi-level security. The application allows the user to type their User Name, Password and the User-Role. It keeps data secure from unauthorized access, modification or reporting by allowing you to create as many ‘users’ as you wish. When you proceed to run automation and web based information broadcasting System**,** you will be prompted to enter your User Name and Password which is the main software part of this System.

CHAPTER 9

SYSTEM PLANNING AND SCHEDULING

9.1 INTRODUCTION

Planning is very important in every aspect of development work. Good managers carefully monitor developments at various phases. Improper planning leads to failure of the project.

Software project pan can be viewed as the following:

- within the organization: How the project is to be implemented? What are various constraints?
- What is market strategy?
- With respect to the customer: Weekly or timely meetings with the customer with presentations Status report Customer feedback is also taken and further modifications and developments are
- done. Project milestones and deliverables are also presented to the customer

9.2 STEPS INVOLVED IN PLANNING A SYSTEM

Selection of project: Includes identifying project's aims and objectives, understanding requirements and specification, methods of analysis, design and implementation, testing techniques and documentation. 1. Project milestones and deliverables

- Project estimates: including cost, time, size of code and duration
- Resource allocation: including hardware, software, previous relevant project information
- Risk management: including risk avoidance, risk detection, risk control and risk recovery
- Scheduling techniques: including work breakdown structure, activity graph, critical path method Gantt chart and Program Evaluation Review Techniques
- Quality control and standard

CHAPTER 10

SYSTEM COST ESTIMATION

10.1 INTRODUCTION

Cost estimation simply means a technique that is used to find out the cost estimates. The cost estimate is the financial spend that is done on the efforts to develop and test software in Software Engineering. Cost estimation models are some mathematical algorithms or parametric equations that are used to estimate the cost of a product or a project.

- During the planning stage, one needs to choose how many engineers are required for the project and to develop a schedule.
- In monitoring the project's progress, one needs to access whether the project is progressing according to the procedure and takes corrective action, if necessary.

10.2 LOC BASED ESTIMATION

As Lines of Code (LOC) only counts the volume of code, you can only use it to compare or estimate projects that use the same language and are coded using the same coding standards.

Features: Variations such as “source lines of code”, are used to set out a codebase. LOC is frequently used in some kinds of arguments.

LOC - Total number of lines in the file. LOC Changed - Total number of added + removed + modified lines. LOC Added - Lines added. LOC Removed - Lines deleted

.

CHAPTER 11

FURURE ENHANCEMENT AND SCOPE OF FURTHER DEVELOPMENT

11.1 INTRODUCTION

The project has a very vast scope in future. The project can be implemented on intranet in future. Project can be updated in near future as and when requirement for the same arises, as it is very flexible in terms of expansion. With the proposed software of database Space Manager ready and fully functional the client is now able to manage and hence run the entire work in a much better, accurate and error free manner

11.2 MERITS OF THE SYSTEM

• Transparency: A review system allows customers to share their honest opinions and experiences with a product, which in turn provides potential buyers with more information to make informed purchasing decisions.

• Credibility: Genuine customer reviews lend credibility to products and sellers. Positive reviews can help build trust among potential customers, while negative reviews can highlight potential issues.

• Search Ranking: Positive reviews can improve a product's visibility and search ranking on the platform, potentially increasing its sales.

11.3 LIMITATIONS OF THE SYSTEM

The limitations of the system are:

• Fake Reviews and Manipulation: The most common issue on many online review platforms, including Amazon, is the presence of fake reviews. Some sellers may create fake accounts to leave positive reviews for their products or negative reviews for competitors' products. Similarly, there could be manipulation of ratings and reviews through incentivized reviews, where customers are offered products or discounts in exchange for leaving a review.

• Changing Product Quality: Product quality can change over time due to manufacturing variations or updates. Reviews for an older version of a product might not accurately reflect the current product's quality.

11.4 FUTURE SCOPE OF THE PROJECT

Enhancements are the perquisite for development of a system. Every existing system has proposed enhancements which make it better and to be used more secure. The enhancements that have been proposed for this system are listed here. Changes can occur at anytime, anywhere. As the saying goes “Change is the only thing that is certain”. The needs can be vary day by day. Especially in business, to complete with competitors, the firm must cope up with the advancement of technologies. It is necessary to make provision for the same. The application developed can be done with the capability for easy integration with other system. Some of the futuristic scope of the system are as follows:

• Personalized Recommendations: The review system could become more sophisticated in analyzing user preferences and behaviors to provide personalized product recommendations based on their reviews and browsing history.

• Advanced Sentiment Analysis: The system could incorporate advanced sentiment analysis techniques, including natural language understanding and emotion detection, to provide more nuanced insights into customer opinions and sentiments.

• Real-time Feedback and Interaction: The review system could allow for real-time interactions between customers and sellers, fostering a more dynamic conversation and helping address customer concerns promptly.

CHAPTER 12

CONCLUSION

12.1 CONCLUSION

The website titled "**Sentiment Analysis For Customer Reviews on www.cavo.ae**" developed is designed in such a way that any further enhancement can be done with ease. Sentiment analysis for customer reviews is a valuable tool that provides insightful and actionable insights for businesses. By analyzing the sentiment expressed in customer feedback, companies can gain a deeper understanding of customer opinions, preferences, and concerns. The system has the capability for easy integration with other systems. New modules can be added to the existing system with less effort. I put as much as my effort to develop this website titled "**Sentiment Analysis For Customer Reviews on www.cavo.ae**" that is easily accessible, informative and helpful. It has been designed in such a way that it is easy to modify, can be updated efficiently and accurately. The forms are designed user friendly by providing messages and captions whenever necessary, so that user has no problem to overcome difficulties in data entry, validation, searching etc. On realizing the importance of systematic documentation all the processes are implemented using a software engineering approach. Working in a live environment enables one to appreciate the intricacies involved in the System Development Life Cycle (SDLC). As we move forward, we recognize the ongoing value of sentiment analysis in continuously improving our customer interactions and overall business performance. By leveraging these insights, we are committed to fostering lasting customer relationships, driving innovation, and maintaining a competitive edge in the ever-evolving marketplace. We have gained a lot of practical knowledge from this project, which we think, shall make us stand in a good state in the future.

Once again I would like to thank everyone who was somehow or other related with the successful completion of this project.

CHAPTER 13

ANEXURE

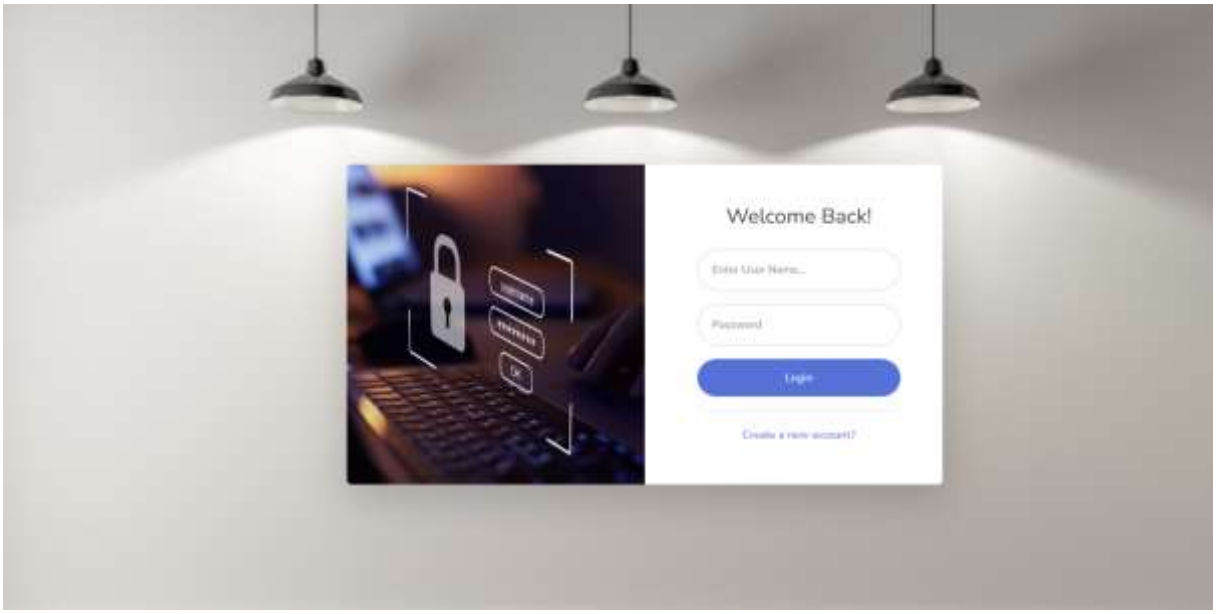


Fig 2: Login Page

Description: This is the Login page of the system. Th users can login through this form.

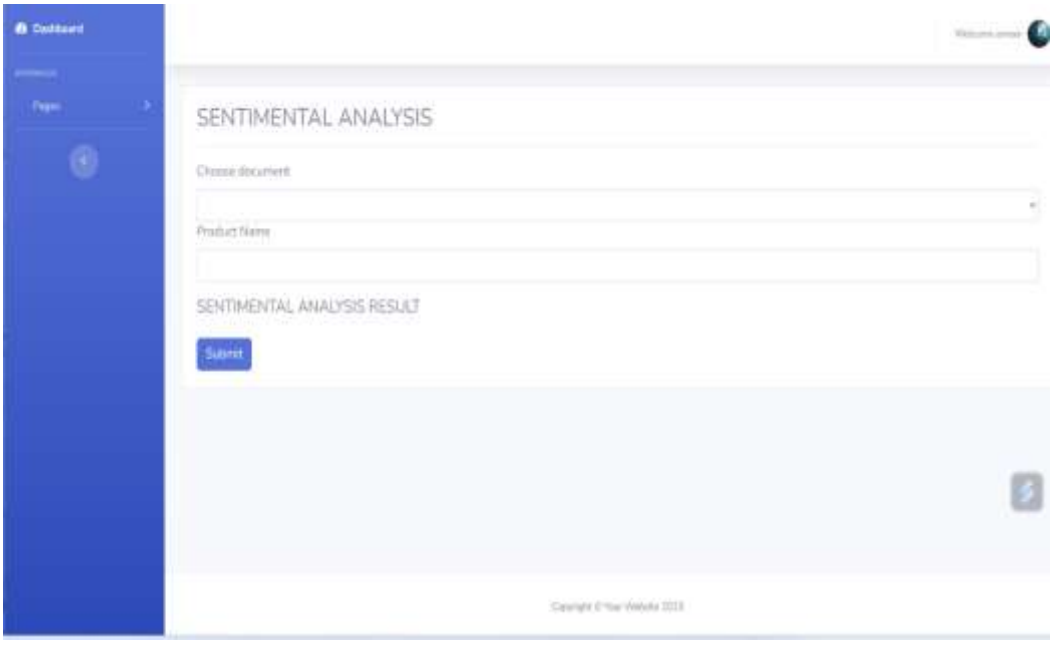


Fig 3: Sentimental Analysis form

Description: This is the sentimental analysis form. When a user using this we get the actual results of this project.

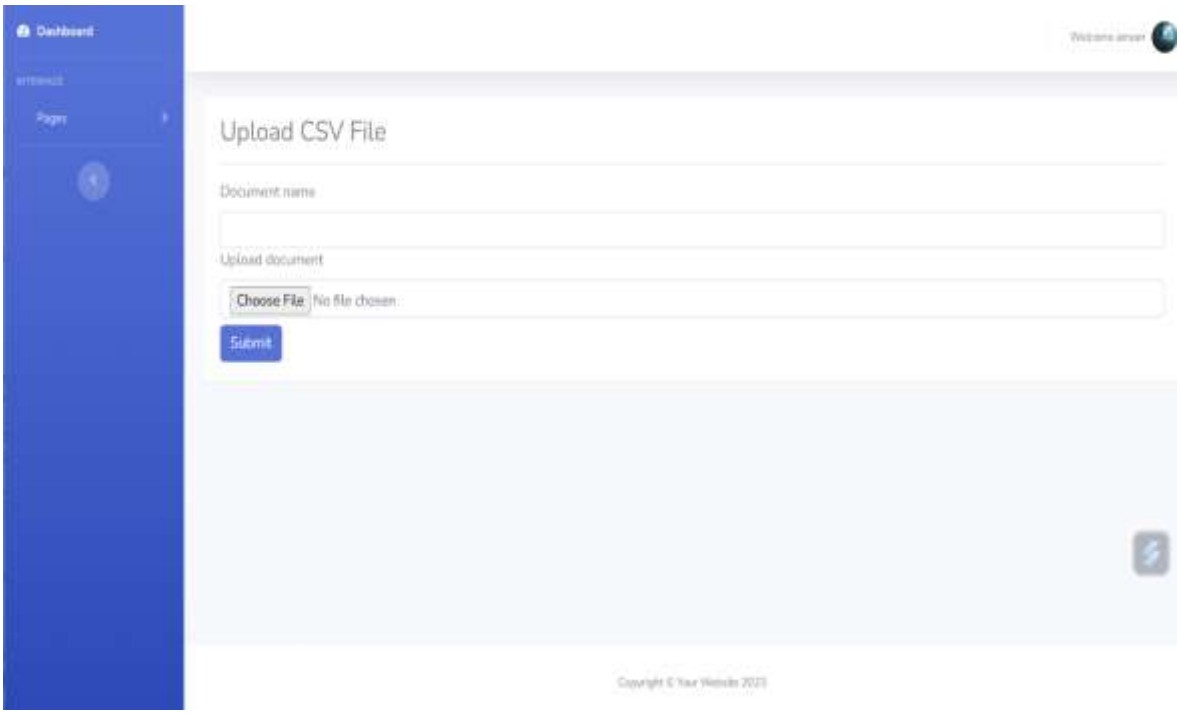


Fig 4: CSV file upload form

Description: Using by this the user can upload the document and submit.

REFERENCE

1 Introduction computing and problem solving using Python(E Balagurusamy)

2 A student's guide to python for physical modeling (Jesse M. Kinder and Philip Nelson)