TITLE OF THE PROJECT

DEEP INSIGHT - AN AI BASED CONSUMER REVIEW ANALYSIS TOOL

Submitted by

Redwan Hossain

ID: 191-16-420

Submitted to

Abdullah Bin Kashem Bhuiyan,

Lecturer, Department of CIS

Daffodil International University

Submission Date: 14 January 2023

Course code: CIS499

Department of Computing and Information System (CIS)

Daffodil International University

Dhaka, Bangladesh

APPROVAL

This Project titled "Deep Insight: An AI based consumer review analysis tool", Submitted by Redwan Hossain ID No: 191-16-420 to the Department of Computing & Information Systems, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computing & Information Systems and approved as to its style and contents. The presentation has been held on- 14-01-2023.

BOARD OF EXAMINAER

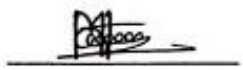
Mr. Md Sarwar Hossain Mollah — **Chairman**

Associate Professor and Head

Department of Computing & Information Systems

Faculty of Science & Information Technology

Daffodil International University



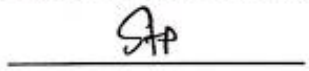
Mr. Md. Mehedi Hassan — **Internal Examiner**

Lecturer

Department of Computing & Information Systems

Faculty of Science & Information Technology

Daffodil International University



Mr. Syed Tangim Pasha — **Internal Examiner**

Lecturer

Department of Computing & Information Systems

Faculty of Science & Information Technology

Daffodil International University

Dr. Saifuddin Md. Tareeq — **External Examiner**

Professor & Chairman

Department of Computer Science and Engineering

University of Dhaka, Dhaka

DECLARATION

I hereby declare that; this project has been done by me under supervision of Abdullah Bin Kashem Bhuiyan Lecturer, department of Computing and Information System (CIS) of Daffodil International University. I am also declaring that this project or any part of there has never been submitted anywhere else for the award of any educational degree like, B.Sc., M.Sc., Diploma or other qualifications.

Supervised By

14.01.23

Abdullah Bin Kashem Bhuiyan
Former Lecturer
Department of CIS
Daffodil International University

Submitted By

Redwan hossain

Redwan Hossain
ID: 191-16-420
Department of CIS
Daffodil International University

ACKNOWLEDGEMENT

First and foremost, I'd like to thank and express my appreciation to Allah for allowing me to complete the project and documentation. Then I'd like to express my sincere gratitude to my parents and family for their support, courage, and compassion throughout my entire course. Without their unwavering support, I will be unable to complete this project. I'd like to express my heartfelt appreciation and gratitude to Abdullah Bin Kashem Bhuiya, my project supervisor. I am extremely grateful to my supervisor for helpful advice and assistance throughout the course. Finally, I'd like to thank all of my Daffodil International University teachers, as well as my friends and well-wishers, who have encouraged and supported me since the project's inception. I am eternally grateful to each and every one of them.

DEDICATION

This is my inaugural final academic project, and I would like to dedicate it to my parents who have been my inspiration and an integral part of my being. Without them, I would not be complete. They truly deserve this recognition.

EXECUTIVE SUMMARY

Deep Insight is an academic project. This project is about automating the review analysis process of an online product-based business. By uploading products database business owners can get various perspectives and comprehensive analytical reports of how the product is performing with the customers. The information provided by the services will be sufficient to assist business owners in reaching a conclusion and making business decisions. Deep Insight is a centralized solution for review analysis that generates quick, accurate, and easy-to-understand analytical reports.

Table of Contents

Table of Figures

CHAPTER 1- INTRODUCTION

1.1 Introduction

In this post-pandemic era, e-commerce has become a very popular business. Review analysis is an integral part of these product-based online businesses. It is the process of going through a product's feedback to understand how a customer feels about the product. It can help in product development, product improvement, identify product issues, improve customer experience, give a clear view of product feature prioritization and finally help guide in decision making. It is an unavoidable process if the business is customer-centric. But the review analysis can be a time-consuming and monotonous process and not to mention in large-scale businesses it requires a certain amount of budget and workforce.

Deep Insight is a centralized solution for review analysis. It fully automates the rigorous process of review analysis with a simple click of a button. This system revolutionizes the standard procedures of review analysis and reinvents it by using smart and innovative means combined with state-of-the-art technology such as artificial intelligence and machine learning.

The online Market is a convenient platform for the general customer yet a challenging business platform for business owners. This platform is expanding at a breakneck pace. An eCommerce statistic predicts that the online marketplace will grow to 6.5 trillion dollars by 2023. In this rat race, this would undoubtedly be difficult to maintain. So, business owners always need to be a step ahead of the competition to compete. But how can they do that if most of the resources, budget, and time are spent on researching how to make a perfect product? It is very difficult to know what makes a successful product. The review section is certainly helpful because it can give insight into customers' perspectives. It can provide information on the customer's takes, preferences, and dislikes which can be helpful to create a successful product. But going through all reviews can be a challenging task especially since text reviews can have deep meaning and connotations. Deep Insight excellent at handling tasks like this. It goes through all the reviews extracting deep and

insightful information and compiling them into a comprehensive analytic report. It does all of this with almost zero human intervention.

Deep Insight is both beneficial to large businesses and small businesses. In large-scale businesses, there are overwhelming amounts of reviews coming daily. They can use this system to fully automate the whole review analysis process, saving the company a significant amount of time and headaches. It will benefit small businesses by saving them money. Small businesses are usually run by a small team or even a single person. So, they do not have an expandable workforce or budget to spend on review analysis.

1.2 Document Content

The following chapters will be covered in this document in order to document the project work.

Chapter 1: Introduction
A brief introduction to the proposed system and project.

Chapter 2: Initial Phase
This section includes the preliminary research specifics for the proposed system, such as the main goals and objectives, problem area, alternative solutions, and project background.

Chapter 3: Literature Review
This chapter includes a thorough discussion of the problem domain, solutions, evaluation of existing solutions, and finally recommendations.

Chapter 4: Methodology
The importance of methodology, the various methodologies that can be used, and the preferred methodology and its implementation will be discussed in this subsection.

Chapter 5: Planning
This section discusses the different project plans, such as project plans, test plans, risk and change management, and so on.

Chapter 6: Feasibility

Here will be documented the detailed feasibility study report and cost-benefit analysis.

Chapter 7: Foundation

This chapter will include information about the problem area identification, overall requirement list, recommended technologies, and justifications.

Chapter 8: Exploration

It includes basic UML diagrams for both the old and new systems, as well as a requirement catalogue and prototype.

Chapter 9: Engineering

This chapter contains the proposed system's logical and behavioral modeling.

Chapter 10: Deployment

The development problem breakdown with development priority and coding samples are discussed here.

Chapter 11: Testing

This chapter includes a number of test plans and results.

Chapter 12: Implementation

The implementation strategy, training model, and other related issues are discussed here.

Chapter 13: Critical Appraisal and Evaluation

The assessment of the original goals that were met and those that were not met in detail.

Chapter 14: Lessons Learned

The pre-project-closing review primarily contains the project's learnings and difficulties.

Chapter 15: Conclusion

The project summary, goals, success, and experience will be documented here.

CHAPTER 2- INITIAL STUDY

2.1 Project Proposal

Introduction: Deep Insight is an AI-powered review analysis tool that aims to fully automate the review analysis process for digital marketplace businesses through natural language processing and statistical analysis. It is a very common scenario where a product gets flooded with thousands of reviews, especially in a marketplace like amazon, eBay, and many other websites. But this can be overwhelming for business owners since it will be a very rigorous task to analyze all those reviews one by one. It is possible to miss out on some crucial reviews that criticize fatal issues of a feature of the product if all the reviews are not carefully analyzed. There is also a common issue in online marketplaces where customer ratings and the sentiment in the review do not match. This is especially the case in phenomena such as “review bombing”, where the customers/bots flood the review section with negative ratings without giving a rational reason. This project solves all the issues by utilizing the power of artificial intelligence technology. It does this by generating the correct rating based on text reviews, filtering out incoherent reviews, providing reports where it is easier to detect irregular reviews which might contain crucial information regarding the product, and generating a thorough analytic report which will help the business owner understand the product and its users.

Background Study: Online marketplace is one of the popular businesses. It is still currently growing exponentially. So, the business owner will have a lot of reviews to analyze which opens up opportunities for this project. The average rate of reviews people write on Amazon is 1-2%. This means that 1-2 reviews are expected for every 100 sales (Kenji ROI,2022). This percentage may appear small, but when we consider the number of users Amazon has (310 million customers worldwide), it adds up to a lot of reviews. This is one of many examples. This project isn’t limited only to amazon reviews, it can analyze any product review as long as it has the necessary data. Thus, it is safe to assume that this project will be extremely beneficial to business owners and in high demand in the digital market industry.

Project description: Deep Insight is a web application that will help business owners thoroughly break down their product reviews using complex models. This will help them understand their customers' incentive to use their product/service, easily identify the product's strongest and weakest characteristics with a comparable score, get a better understanding of their customer's tastes and preferences, and even get suggestions based on their products' analytics. Deep Insight has services that generate both high and low-level analytic reports. The high-level reports are mostly suited for people who want a quick overview of their product. To understand low-level reports it is sometimes required to have basic statistical knowledge but it can provide more insight into the product and even hopefully answers the unanswered question about the product or the product's user. Deep Insight aims to make customer-oriented business easier by providing them with analytics reports based on the product review which will help the business owner get a deep insight into their product and the product's demographic and guide them in making crucial business decisions and devise plans.

Feasibility Study

Operational Feasibility: Deep Insight is a fully automated system that has easy-to-understand instructions. It will also have an interactive and smooth navigation system. If the user wants more control over how the review data they provide is processed, there is a section that allows the users to change the behavior of some services the system has. It is also possible for the users to integrate their language model into the system. But this is optional since the system automatically selects the best-suited model for the review.

Technical Feasibility: The core resources needed to create such a system are customer reviews, language models, and a platform for the system to interact with the customers. The customer reviews must be provided by the client. The client must provide sufficient customer reviews for the language models to conclude. The reviews have to be comprehensive and constructive. So, this will require a lot of filtering and transformation which will be done by the system. There are a lot of open-source pre-trained language

models which we can make use of with some tuning and further training. As for the platform, I decided a website will be the best option. The website will also have a database.

Economic Feasibility: Website systems are easy to deploy. Multiple websites offer free hosting servers. The system has low maintenance since everything is automated by the system. It is possible to host this system locally. I will be hosting the system locally since it is faster than the free servers.

Market research analysis:

This system can only be used for online marketplace websites. According to Google, there are more than 150 million marketplace active websites. Roughly 79% of Americans today make purchases online. 15% of Americans make online purchases every week. So, there is no shortage of platforms and demand for this system. The retailer can vastly benefit from this system since this system reduces manual work and human intervention. They can improve or change products based on the system's analysis to improve sales. Since this system provides information based on reviews it can easily identify the customer's preferences.

Foundation

Main Goals

- Review Analysis: The clients will be able to get in-depth information about the product from the customer's point of view.
- Analytics generate: Clients will be able to identify important aspects and reactions from reviews using artificial intelligence models.
- Improve the quality of the decision-making process: Based on the analytics provided by the system, clients will be able to quickly reach conclusions and make business decisions.

- Understanding product demographics: Clients will have a better understanding of the demographics of the product.
- Product assessment: Clients will know the strengths and weaknesses of their product based on the customer's review.

High-level features for achieving the goals

- Upload the File and select the section.
- Information and instructions about the services the system provides.
- Sentiment-Based Aspect Extractor
- Aspect to Score
- Similarity Cluster
- Topic Extraction
- Virtual Chat Assistant
- Visual analytics for each service

Key activities for achieving the goals

Task	Description
Requirement analysis	In this section what kind of resources needed to build the system are identified.
Planning	In this section, the information gathered from the previous section is used to formulate the system construction strategy.
Developing system	In this section, the services for the system are developed separately.
Interface design	In this section, the interface for the system is created.
Testing the system	In this section, both the interface and the backend system are tested both individually and collectively.
Documentation	In this section, the whole roadmap of building the system is recorded.

Prioritization

The functionalities are created independently in projects like this one. The development of these functionalities occurs in stages. Nevertheless, some functions are more crucial than others, and it is important to take into account any dependencies. This means that creating them requires careful planning of the sequence. Which is why I used Moscow for system development. Here is the prioritization of my system:

Must have:

- Upload and select product review.
- Services that analyze selected product reviews and output a result.

Should have:

- Option to change between product reviews.
- A suggestion-based service.

Could have:

- Information about each service.

Would have:

- Sample of available product reviews.
- Customization option for services that offer customization.

Evolutionary Development Based on the PRL (Prioritized Requirements List) the final features are given below:

- Upload File
- Select File
- Services
- Change File
- Service Info

- Product Review sample
- Customize Service

Time box for development

#	Task title	Start date	End date	Duration (Days)
1	Introductions	1/7/2022	2/7/2022	1
2	Initial study	3/7/2022	8/7/2022	6
3	Literature review	9/7/2022	15/7/2022	6
4	Methodology	16/7/2022	20/7/2022	4
5	Planning	21/7/2022	27/7/2022	6
6	Feasibility	28/7/2022	30/7/2022	2
7	Foundation	31/7/2022	4/8/2022	4
8	Exploration	5/8/2022	8/8/2022	3
9	Engineering	9/8/2022	13/8/2022	4
10	Deployment	14/8/2022	2/9/2022	18
11	Testing	3/9/2022	8/9/2022	5
12	Implementation	12/9/2022	16/9/2022	4
13	Critical Appraisal & Evolution	17/9/2022	20/9/2022	3
14	Lessons learned	21/9/2022	23/9/2022	2

15	Conclusion	24/9/2022	25/9/2022	1
Total	69 days			

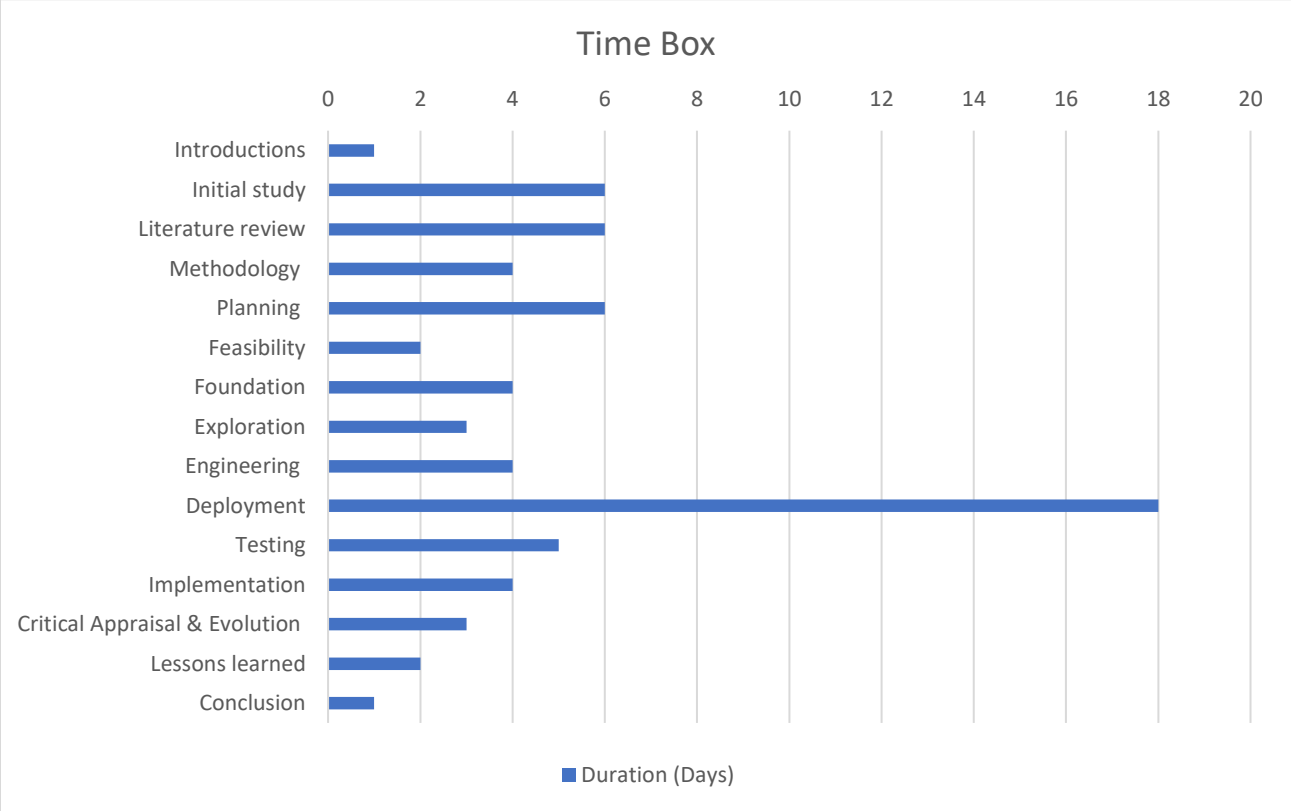


Figure 1: Gantt Chart of the Proposed Timebox Estimation

Deployment:

In this phase, the system is developed as a web application.

The UI was created using JavaScript, jQuery and bootstraps, and python's flask library for routing. SQLite was used for the database.

Most of the language models use are from hugging-face. Python was to handle model-based logic and control.

Conclusion: In current times online marketplace has a huge demand. So, there will also be a demand for this system. This system will help the business owners identify important

parts of their product that they can utilize to improve the product and hopefully boost their sales.

2.1 Background of the project

A lot of business has turned towards the online marketplace, especially after the pandemic. This market extends even to social media networks. If the business is big or if it has multiple products in the line, it will be hard for the owner to keep track of all his products.

A good product idea does not ensure it will sell well. There are multiple variables behind a product's success. Some products sell very well and some don't. Deep Insight can get this kind of information of what makes the product good and what makes it bad directly from the customer by analyzing the reviews. Some businesses manually analyze the review which is time-consuming and resource intensive. This system fully automates this step without any human intervention.

This system will be highly demanded in the current market’s state. This will be helpful to business owners.

2.2 Problem areas

It's difficult to design a product that will sell successfully. Conducting market research, review analysis, and surveys, and designing the product accordingly can raise the likelihood that a product will be successful. However, this takes a lot of time and resources. Deep Insight system solves this problem by fully automating the review analysis process. It accomplishes this by using natural language processing to transform unstructured review data into comprehensive analytical information that can be used to guide decision-making.

2.3 Possible solutions

This system can assess a significant amount of reviews automatically by doing complex calculations and producing meaningful information. The outcome will include details on key product features and how users felt about them, as determined by the reviews. This

will help the client see the product from a customer's point of view. The system will also offer details on specific evaluations and how it generated them so that the customer may if they choose, draw their conclusions. A service that compresses all the information the system has processed and offers a suggestion will be available.

CHAPTER 3- LITERATURE REVIEW

A literature review is a type of evaluation piece. A literature review is an academic document that combines new material and significant results, as well as theoretical and methodological contributions to a certain subject area. It is a secondary source that does not contain any fresh or distinctive experimental work. The four goals of a literature review are to survey, synthesize, critically evaluate, and present. (2019, rlf.org.uk) It focuses on a certain aspect of the literature review. It summarizes the information in that literature. It also analyzes the limitations of the literature and gaps in current knowledge and presents them in an organized manner.

3.1 Discussion on the problem domain

This system is focused on fully automating the research part of the company to save the client time and resources. These types of analytic services are being offered by various web services. However, there are no centralized business analysis solutions. Additionally, the data these services offer is insufficient for the business owner to make a choice; they still need to conduct additional research independently. This system offers an all-inclusive analysis solution that includes all widely used text analysis methodologies as well as its special analytical methods that will aid business owners in learning more about their customers and enhancing their production accordingly. Core problems with currently available services summarized:

- ➢ Not fully automated.
- ➢ Not centralized. The client might need to use multiple solutions from different firms to get a grasp of the situation.
- ➢ Not enough information is provided to make an important decision.
- ➢ Lack of AI-based analysis tools.

3.2 Discussion on the problem solution

Deep Insight is a centralized approach to business analysis that makes use of cutting-edge technology to give the customer precise and insightful information about their product. This system is the solution to all the problems stated above and more:

- **Informative**

  The system will use state-of-the-art language models to extract information from the review and use math and statistical calculations to summarize that data and provide clients with perceptive information on the product based on customers' opinions.

- **Remote Access**

  The system uses complex models that require expensive hardware and needs a lot of space on the hard drive. But since this system will be hosted on a server and won't run on their device so they don't have to worry about any of this. They can use this system from any web browser.

- **Privacy**

  The system will have a separate database for each client. The system will save some of the reviews that it failed to comprehend to further improve itself so it can provide more information next time.

- **Customizability**

  Some clients might want more control over how the data is processed by the system. They can provide the language model that will be used to process their data.

- **Powerful Technology**

  The system uses state-of-the-art AI modes to analyze reviews and identifies in-depth variables about the product from the customer's point of view, which is critical for making business decisions.

3.3 Comparison of the leading solutions

The primary goal of software and services is to automate or condense complex operations. This applies to the business section as well. Analytical solutions are offered by several online and offline firms. Although this system is distinct, several services operate under the same domain or have features that are comparable to those of this system.

Here are some online services that are similar to Deep Insight with their strengths and weaknesses and best features listed:

- https://monkeylearn.com/
- https://aws.amazon.com/
- https://cloud.google.com/

MonkeyLearn

MonkeyLearn is a Machine Learning platform for Text Analysis. It has a feature that is dedicated to reviewing analysis.

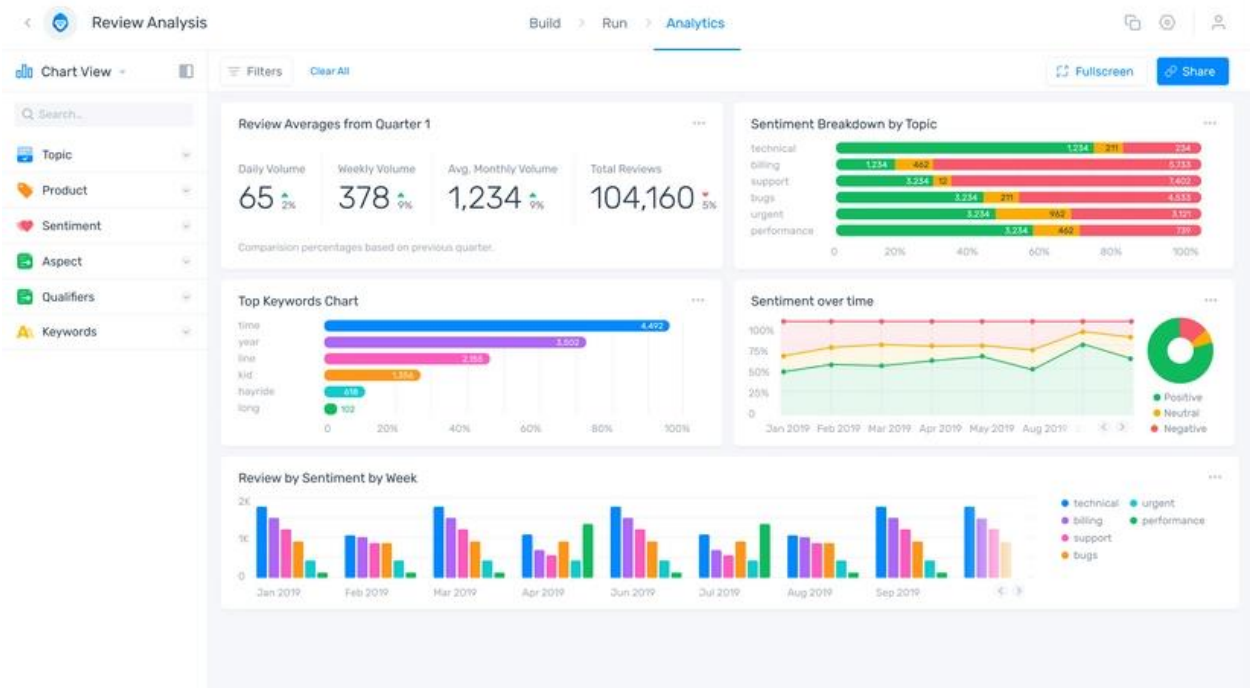


Figure 2: Overview of the MonkeyLearn

Best Features

- Beautiful Interface design
- Easy-to-use services
- Visualization reports

Limitations

- Expensive services
- Lack of review analysis-based services

Amazon Web Services

One of the largest cloud services that use state-of-the-art language models.

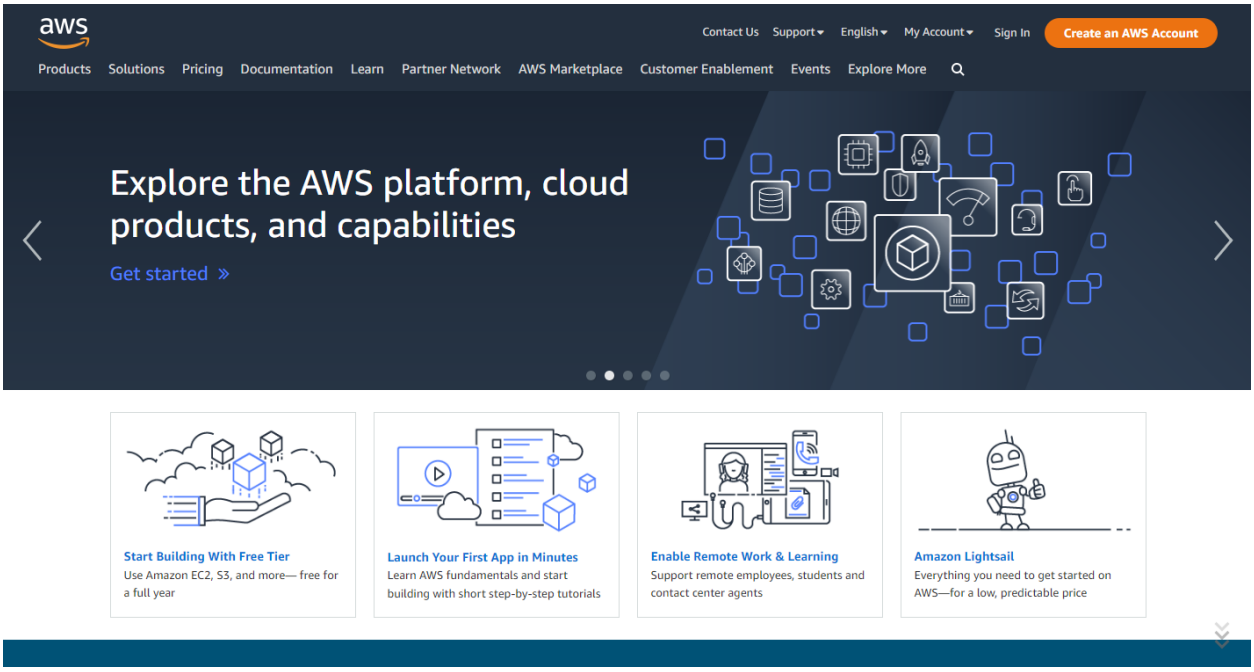


Figure 3: Overview of the Amazon Web Services

Best Features

- Powerful language model
- Secure service
- Fast servers

Limitations

- Services must be implemented by a developer
- Not centralized

Google Cloud Services

Offers Intelligent, safe, and very adaptable services.

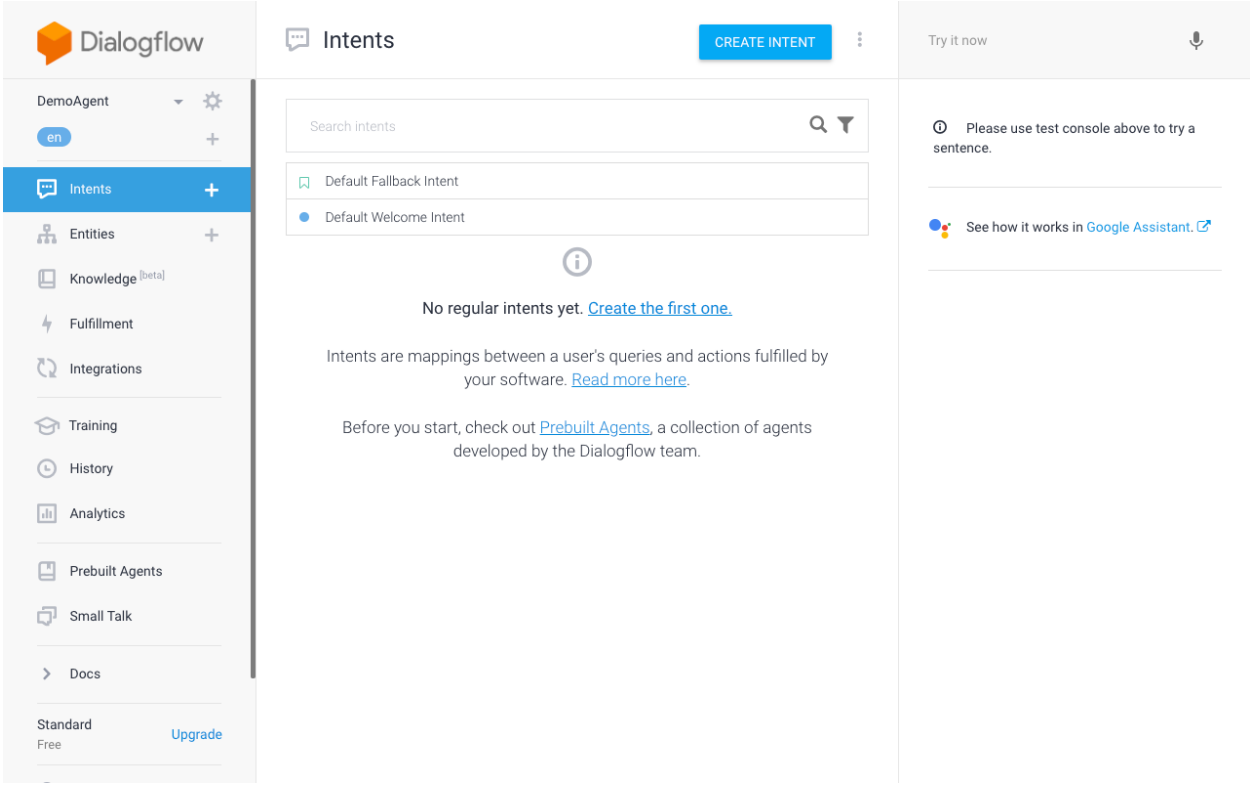


Figure 4: Overview of the Google Cloud Services

Best Features

- Powerful language model
- Easy Implementation
- Backup service

Limitations

- Lack of customization.
- Developer Focused

3.4 Recommended Approach

Following a rigorous analysis of comparison websites, the following aspects should be included in the suggested system:

- Simple and attractive interface
- Navigation bar with matching theme
- Powerful language models
- Collection of related services
- Visualized result

CHAPTER 4- METHODOLOGY

The software development methodology is a systematic approach to developing software. Each methodology has its unique life cycle, synthesis of design philosophies, and pragmatic realism that dates back to the dawn of computing.

4.1 What to use:

There are several software development methodologies available, each with its approach and discipline. Choosing the appropriate methodology for a project is critical because it benefits both the client and the development team by allowing for an accurate project timeline, increasing efficiency, and resulting in higher-quality deliverables that meet deadlines. I will list out three potential methodologies for my academic project each describing their pros and cons below:

Spiral Model

The spiral model is a risk management method that combines the iterative development process model and elements of the Waterfall model. Software engineers prefer the spiral model for large, costly, and complicated projects.

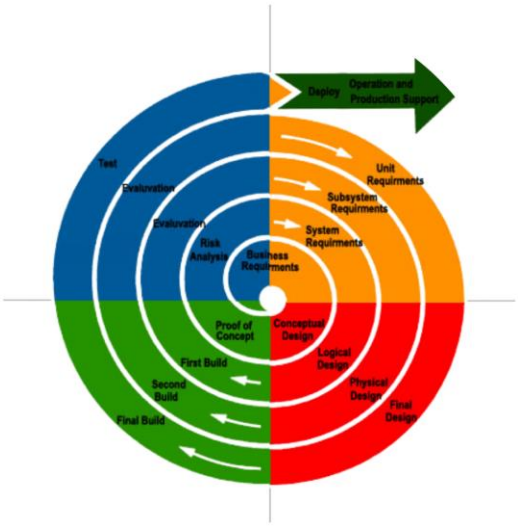


Figure 5: Spiral Methodology

Pros

- Suitable for a large and complex project
- Can handle changes in requirements
- Specializes in risk management

Cons

- Can be a complex and expensive
- Documentation can be hard as it may have many iterations
- The model has an Indefinite time frame. It is hard to determine the endpoint for this model early on.

V-Model

The V-model is divided into two phases. Design or verification phase and Test or validation phase. Both of these phases have their subsequent phases. Each subphase of the design phase is integrated with a subphase of the testing phase which creates a “V” like shape.

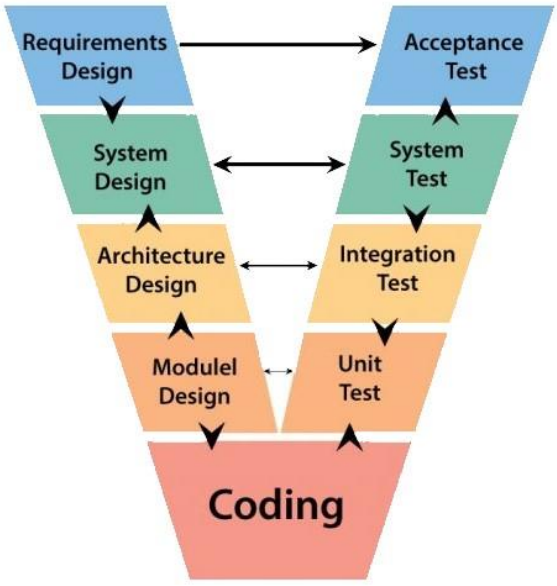


Figure 6: V-model Methodology

Pros

- Test-driven model, the defects are identified at an early stage
- Straightforward and easy-to-understand model
- Quick and time-saving model especially suited for small projects

Cons

- The model is not very flexible
- The documents might need to be constantly updated
- The model doesn't provide any prototype early on

Kanban

The word Kanban (Japanese: カンバン) roughly translates to "card you can see." This agile methodology prioritizes flexibility over following a plan and is frequently used for projects that are subject to frequent change. This was first introduced in the 1950s, Toyota pioneered and refined the use of kanban in a relay system to standardize part flow in their just-in-time (JIT) production lines.

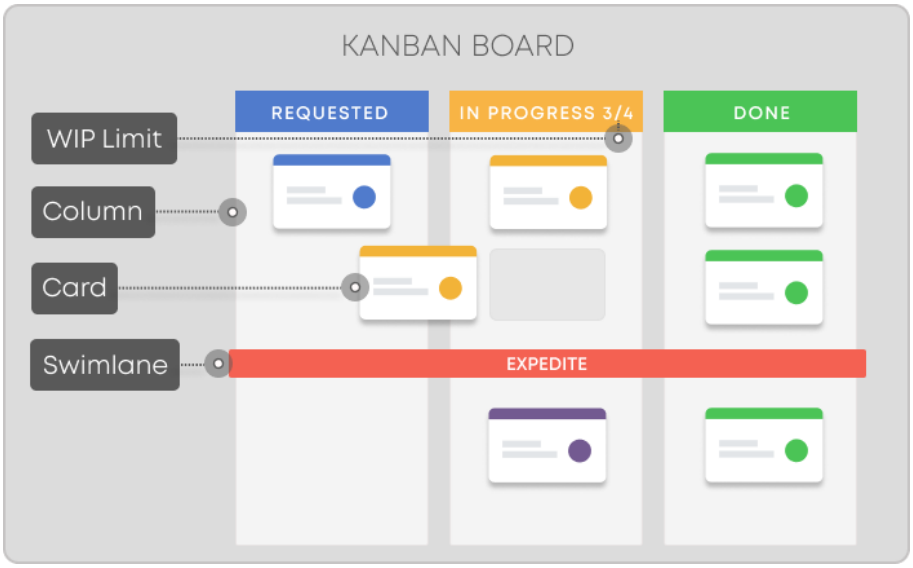


Figure 7: Kanban Methodology

Pros

- Easy to and simple implementation
- Can be used in different situations/projects
- Highly flexible

Cons

- Not suited for dynamic settings
- Doesn't provide/document the time-frame

4.2 Choosing Methodology:

After careful consideration, I have decided to utilize the Kanban methodology for project management for the proposed system. Kanban will be best suited for the project. It ensures a quality product delivery which is a must for the project because this is an analysis-focused project and accuracy is the core factor it must provide. This methodology is also popular for its way of tracking project progress with a visual board. This is also essential because this is an academic project that has to be completed within a limited time frame.

4.3 Why to use:

Adhering to a well-defined methodology allows a project to be delivered within the timeframe specified, identifies risks, provides a clear understanding of the project requirements, and ensures the delivery of a quality project.

The Kanban discipline is perfect for this project. Kanban has 6 simple principles:

- Never distribute defective goods.
- Take only that which is required.
- Produce exactly the quantity that is needed.
- Control the level of production
- Optimize the production
- Stabilize and rationalize the process.

The core reason for choosing Kanban for this project is how it uses the “pull system”. A Pull System is a technique for managing the resource flow in a system. Kanban is a continuous process. It goes through the requirements list in a prioritized order. These requirements go through all the phases in the kanban board (also known as the agile board). In kanban, the number of requirements pulled at a time is limited with the help of the WIP (Work in progress) limit. Using Kanban to limit work-in-progress encourages higher quality and better delivery performance. Limiting WIP allows the team to optimize work capacity by allowing them to pull new work only when capacity is available. WIP is crucial for the proposed system.

Deep Insight is a service-based web application. Meaning there will be multiple services, some more important than others. Attempting to build all the planned services at once might be a bad idea. Because these services are complex, the estimated completion time is unknown and since the whole project will be built by only me it will be difficult to complete the project within the proposed time. Using the pull system I will be able to complete the important services first. WIP limit will ensure high-quality services.

4.4 Sections of methodology:

Kanban for the project will have the following sections:

- **Funnel (Stage 0)**
  Ideas for the project are brainstormed in this stage
- **Analyze (Stage 1)**
  In this section, initial research, requirements analysis, and feasibility assessment regarding the ideas gathered from the previous stage are done
- **Plan (Stage 2)**
  Comprehensive sets of plans are created to successfully implement the requirements
- **Design (Stage 3)**
  The user requirements are converted into a suitable form, which aids the programmer in software development and implementation. The main goal of this stage is to create a project model that demonstrates firmness, delight, and commodity.
- **Engineering (Stage 4)**
  Requirements are pulled into this stage to make them into a working feature
- **Test (Stage-5)**
  Features are verified and validated
- **Implementation (Stage 6)**
  The features are integrated or implemented into the system
- **Completed (Stage 7)**
  Each requirement is marked as done to track and monitor overall progress

CHAPTER 5- PLANNING

5.1 Project Plan

In this chapter, the project is planned thoroughly. The project is divided into multiple segments. Each segment contains a different part of the project and every segment is done within a specific time frame.

5.1.1 Work Breakdown Structure

By using a work breakdown structure, the projects are divided into manageable tasks. With this framework, it is simpler and more effective to finish the project's duties in the allotted time. It makes a presumption regarding the working hours and tasks. Project completion may be more difficult without this framework. The suggested system has been divided into categories and subcategories with the breakdown structure in mind. Below are the charts.

#	Task title	Start date	End date	Duration (Days)
1	Introductions	1/7/2022	2/7/2022	1
2	Initial study	3/7/2022	8/7/2022	6
3	Literature review	9/7/2022	15/7/2022	6
4	Methodology	16/7/2022	20/7/2022	4
5	Planning	21/7/2022	27/7/2022	6
6	Feasibility	28/7/2022	30/7/2022	2
7	Foundation	31/7/2022	4/8/2022	4

8	Exploration	5/8/2022	8/8/2022	3
9	Engineering	9/8/2022	13/8/2022	4
10	Deployment	14/8/2022	2/9/2022	18
11	Testing	3/9/2022	8/9/2022	5
12	Implementation	12/9/2022	16/9/2022	4
13	Critical Appraisal & Evolution	17/9/2022	20/9/2022	3
14	Lessons learned	21/9/2022	23/9/2022	2
15	Conclusion	24/9/2022	25/9/2022	1
Total	69 days			

5.1.2 Resource allocation

Resource allocation is allocating and managing all resources or assets in an organized manner so that the proposed project may be finished with a proper plan. It is a key project planning activity. The assignment must be accomplished as a team, however, because this is an academic project, one individual must play many roles at different times. The primary aim of resource allocation is to keep the task schedule on track to accomplish the goals and deadlines.

#	Task title	Resource Name	Duration (Days)
1	Introductions	Analyst	1

2	Initial study	Analyst	6
3	Literature review	Analyst	6
4	Methodology	Analyst	4
5	Planning	Analyst, Developer	6
6	Feasibility	Analyst, Developer	2
7	Foundation	Analyst, Developer	4
8	Exploration	Analyst, Developer	3
9	Engineering	Analyst, Developer, Tester	4
10	Deployment	Developer, Tester	18
11	Testing	Developer, Tester, User	5
12	Implementation	Analyst, Developer, Tester, User	4
13	Critical Appraisal & Evolution	Analyst	3
14	Lessons learned	Analyst, Developer, Tester, User	2
15	Conclusion	Analyst	1

5.1.3 Time Box

Because the majority of the proposed system is built using the DSDM methodology. The actual duties are completed in an iterative approach. As a result, timeboxing is a critical component of this process. The time box is divided into tasks and periods. There are additional four-time boxes for this project, which are completed by the given resources.

Time Box	Task title	Resource Name

T-1	Introductions	Analyst
	Initial study	Analyst
	Literature review	Analyst
T-2	Methodology	Analyst
	Planning	Analyst, Developer
	Feasibility	Analyst, Developer
T-3	Foundation	Analyst, Developer
T-4	Exploration	Analyst, Developer
T-5	Engineering	Analyst, Developer, Tester
T-6	Deployment	Developer, Tester
	Testing	Developer, Tester, User
T-7	Implementation	Analyst, Developer, Tester, User
	Critical Appraisal & Evolution	Analyst
T-	Lessons learned	Analyst, Developer, Tester, User
	Conclusion	Analyst

5.1.4 Activity Network

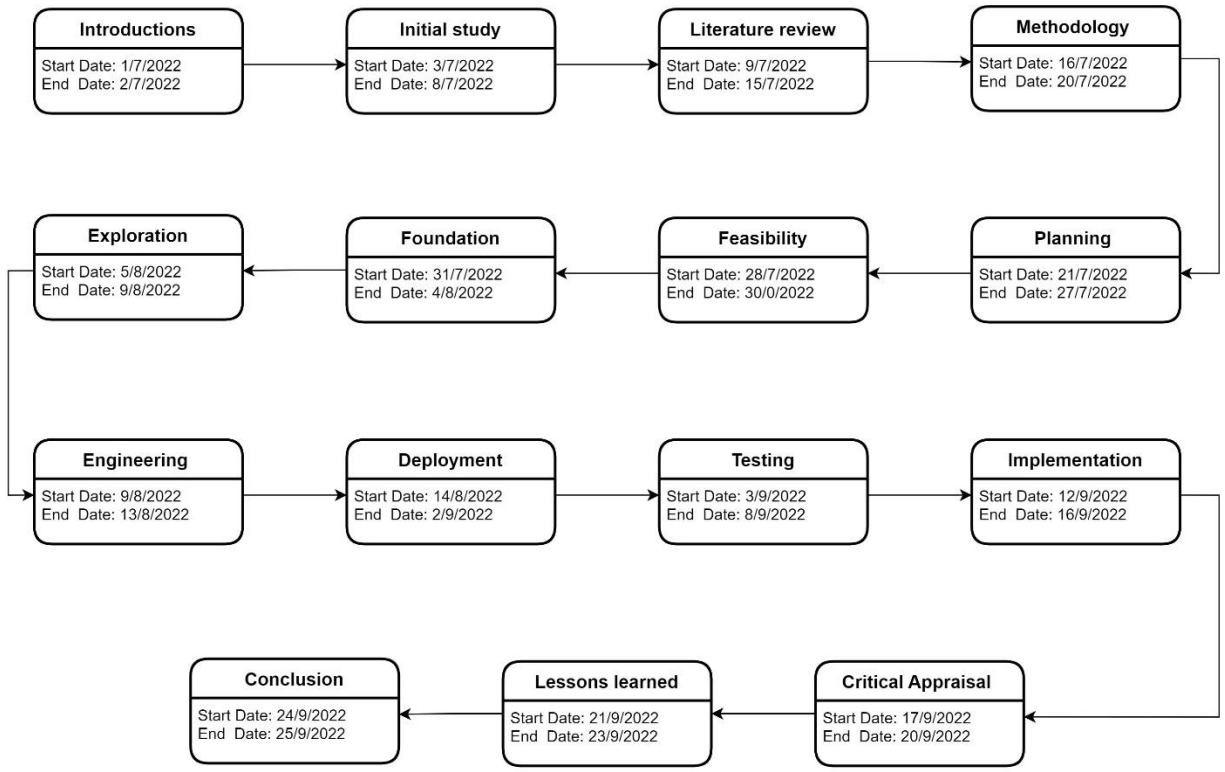


Figure 8: Activity Network of DeepInsight

5.1.4 Gnatt Chart

Gantt charts are used to illustrate project activity segmentation. It displays the period of an activity together with the start and end dates. A Gantt chart is very beneficial to developing a project. The Gantt chart for the proposed system is shown below.

Start date	End date
1/7/2022	2/7/2022
3/7/2022	8/7/2022
9/7/2022	15/7/2022
16/7/2022	20/7/2022
21/7/2022	27/7/2022
28/7/2022	30/7/2022
31/7/2022	4/8/2022
5/8/2022	8/8/2022
9/8/2022	13/8/2022
14/8/2022	2/9/2022
3/9/2022	8/9/2022
12/9/2022	16/9/2022
17/9/2022	20/9/2022
21/9/2022	23/9/2022
24/9/2022	25/9/2022

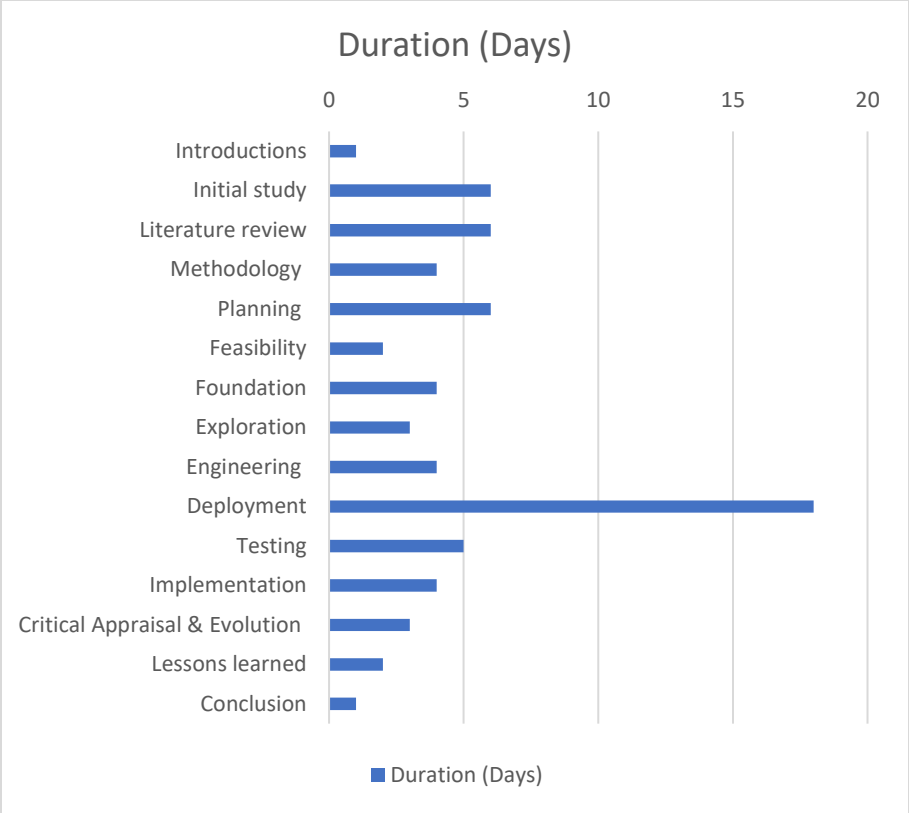


Figure 9: Ghantt Chart

5.2 Test Plan

Software testing is the process of evaluating and verifying that a software product or application does what it is supposed to do. The benefits of testing include preventing bugs, reducing development costs, and improving performance. (*What Is Software Testing and How Does It Work?* | *IBM*, n.d.)

5.2.1 Testing Against Time Boxes

Every time box will have a task name associated with the time of execution. The tasks will be completed in consecutive order. The preceding task must be completed before beginning the next. The results of the testing against the time box are as follows:

User Name: ExampleUser	**Role:** ExampleRole
Time Box ID	
Time Box Content	

Test Type	Test Steps	Expected Result	Actual Result	Comment
Unit Testing				
Integration Testing				
System Testing				
Acceptance Testing				
Security Testing				
Usability Testing				
Reliability Testing				

5.2.2 Required Testing

The two types of testing are given below:

Functional Testing:

- **Unit Testing:** This is the process of testing the smallest piece of code in a system that can be logically isolated. Unit testing validates small components of the

system individually making it easier to narrow down errors early on.

- **Integration Testing:** Refers to a type of testing where one or multiple components combined or a module is verified to see if the integrated modules work as expected. This testing is concerned with ensuring data communication between various system modules.
- **System Testing:** Testing entails testing the entire system. All modules are integrated when tested to determine whether or not the system works as expected.
- **Acceptance Testing:** This is a quality assurance process that validates whether the system meets the customer's requirements or does exactly what it intended to do.

Non-Functional Testing:

- **Security Testing:** Tests the vulnerability and security of the system. Determines whether the capabilities of defending itself against threats such as SQL injection, XSS, CSRF, etc attacks.
- **Usability Testing:** Determines whether the demographic user can easily understand and operate the system. Ensures that the system is self-explanatory and no training is required to operate the system.
- **Reliability Testing:** Reliability is defined as the probability of failure-free software operation for a specified period in a particular environment. (*What Is Reliability Testing: Definition, Method and Tools*, n.d.)

5.2.3 Test Case

The test case report will contain a test case name, class/method the test is done on, description of the test (if needed), and the comparison between expected and actual output with the steps leading to them. Format for the test case report:

Test Case Name			
Test Class			
Test Description			
Data Source	Test Steps	Expected Result	Actual Result

5.2.4 User acceptance test plan

An acceptance test will be performed to clarify whether the final product satisfies the user's criteria or if it is solving the problem it was intended to solve. This will be the final functional test before deployment. The format for this test case report is given below:

Test Case No.			
Test Type			
Test Title			
Preconditions			
User Name			
Act as			
Steps	Expected Result	Actual Result	Comments

5.3 Risk Management

Risk management is the process of identifying, analyzing, and handling the risk that occurs or might occur in the future for the proposed system throughout the system's life cycle. Risk management covers the area of project risk, technical risk, and business risk and will consider and attempt to handle the risk after the system is implemented and deployed.

For the proposed system Microsoft 365 Risk Management approach is incorporated. In this approach, the whole process of risk management will be divided into multiple phases.

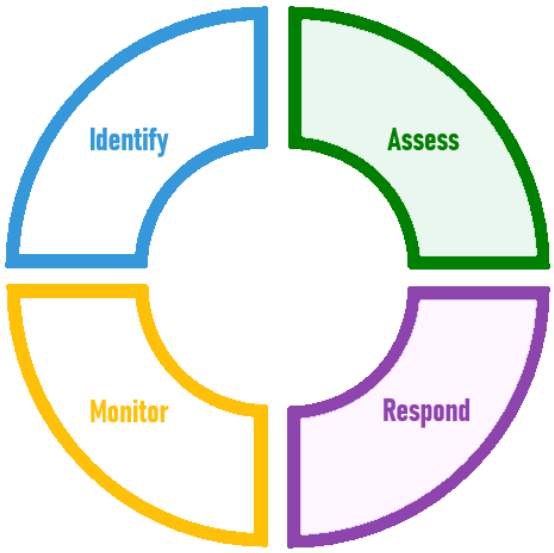


Figure 10: Microsoft 365 Risk Management

CHAPTER 6- FEASIBILITY

6.1 All Possible Types of Feasibility

Operational Feasibility: Deep Insight is a fully automated system that has easy-to-understand instructions. It will also have an interactive and smooth navigation system. If the user wants more control over how the review data they provide is processed, there is a section that allows the users to change the behavior of some services the system has. It is also possible for the users to integrate their language model into the system. But this is optional since the system automatically selects the best-suited model for the review.

Technical Feasibility: The core resources needed to create such a system are customer reviews, language models, and a platform for the system to interact with the customers. The customer reviews must be provided by the client. The client must provide sufficient customer reviews for the language models to conclude. The reviews have to be comprehensive and constructive. So, this will require a lot of filtering and transformation which will be done by the system. There are a lot of open-source pre-trained language models which we can make use of with some tuning and further training. As for the platform, I decided a website will be the best option. The website will also have a database.

The technical features that will be used in this project are listed below from this project's perspectives.

Hardware:

- AMD Ryzen™ 5 2400G with AMD Radeon™ RX Vega 11 graphics (Integrated)
- Walton UX01 800VA offline UPS
- P-Link Wifi Router

Software:

- Google Docs
- Visual Studio Code
- Microsoft Paint 3D
- Operating System: Windows 10

Database:

- SQLite

Technology:

- Client Side:
    - HTML
    - CSS
    - JavaScript
    - Bootstrap
    - JQuery
- Server Side:
    - Python

Economic Feasibility: The project will be using various web-based subscription services. There are various options available. But here are some services with costs that I would use for this project.

- Software/Service Cost:

  The entire system will be hosted in a remote server so it can be accessed from anywhere.

Equipment	**Cost Per Unit**	**Price**
Express VPN	2,000৳ per month	2,000৳

Google Services (Drive, Email, Servers)	10,000৳ per month	10,000৳
Total		12,000৳

- Hardware Cost:

Equipment	**Cost Per Unit**	**Price**
Core I 5 11th Gen	19,800৳	19,800৳
MSI B560M-Pro Motherboard	11,700৳	11,700৳
Nvidia Geforce RTX 2060 6GB	32,500৳	32,500৳
Corsair Ram 8GB x 2	4,000৳	8,000৳
Seagate 1TB HDD	5,000 ৳	5,000৳
MSI MAG 650w power supply	6,000৳	6,000৳
D-link wireless router	3,500৳	3,500৳
Others	10,000৳	10,000৳
Total	120,500৳	

6.2 Cost Benefit Analysis

Cost-benefit analysis is a way to compare the total estimated cost and the total estimated benefit generated by a project. It helps individuals or organizations determine whether a project is feasible by calculating if the project profit outweighs the project expenditures.

Total Cost

SL	Sector of Expenditure	Year 1	Year 2	Year 3	Year 4	Year 5	Total
1	Hardware Cost	120,500	-	-	-	-	120,500
2	Service & Hosting Cost	12,000	12,000	12,000	12,000	12,000	60,000
3	Staff Cost	50,000	50,000	50,000	50,000	50,000	250,000
4	Others Cost	25,000	25,000	25,000	25,000	25,000	125,000
Total Cost		207,000	87,000	87,000	87,000	87,000	555,500

Total Earning

SL	Sector of Earnings	Year 1	Year 2	Year 3	Year 4	Year 5	Total
1	Earnings from services	100,000	120,000	150,000	180,000	200,000	750,000
2	Earnings from personal ads	9,000	12,000	16,000	20,000	25,000	82,000
3	Earnings from Google AdSense	50,000	55,000	60,000	69,000	75,000	309,000
Total Earnings		159,000	187,000	226,000	269,000	300,000	1,141,000

Total Revenue

SL	Sectors	Year 1	Year 2	Year 3	Year 4	Year 5	Total
1	Total Earnings	159,000	187,000	226,000	269,000	300,000	1,141,000
2	Total Expenditure	207,000	87,000	87,000	87,000	87,000	555,500
Total Revenue		-48,000	100,000	139,000	182,000	213,000	585,500

From the cost benefit analysis done above it is clear that the organization will profit by developing this software. In the first year of developing the cost will be greater than earnings. This is because of the initial cost of hardware needed to run the system. But in the upcoming years the revenue will increase exponentially.

6.3 Is DSDM Good or Bad for this Project

DSDM stands for Dynamic Systems Development Method which is an agile project delivery framework. It is great at managing projects that have tight time constraints. It is well suited for the proposed system because it develops projects through the use of incremental prototyping in a controlled environment. Deep Insight is a service-based web application, the services could be developed in an increment procedure. This is an academic project which has a tight deadline. So DSDM will be a good methodology for developing the proposed system.

CHAPTER 7- FOUNDATION

7.1 The Problem Area Identification

To build a project, consider the user's perspective. Things like what the user prefers and how he/she interacts with the program should be considered while building the application. It is not feasible to create an application that everyone will enjoy, but it is possible to create something that will appeal to the majority of its users. Here are some techniques to gather this type of information:

7.1.1 Interview

The best way to know what the customer thinks or expects from the application is to directly ask them. This might seem simple but it is an effective way to improve the quality of the application. Interviews with a diverse group of customers can help understand the demographics more and can sometimes help detect major errors in the current solutions and help in developing a new and improved solution. Here are some predefined questions that will be asked in the interview.

For Clients:

- What problems does a company or an individual face when analyzing reviews
- How long does it take to fully analyze a product review
- It is worth investing in review analysis
- Does analyzing review help improve the product or help develop an improved version of the product
- Do you ever come across reviews where the rating and the actual review do not match?

7.1.2 Observation

Observation is a great data-gathering tool that helps us understand behavior and preferences. This technique can help the developer understand more about the user demographics. Here are some points to focus on when observing the users:

- Find out the process of review analysis
- Find out the application/software used for analyzing reviews
- Find out what kind of information is gathered from the review analysis
- Average time needed for a product's review analysis
- What kind of pieces of information are recorded in the report
- Requirements for review analysis

7.1.3 Questionnaires

Questionnaires are another excellent technique that can help gather information on specific factors of the application. Unlike the other techniques, this one focuses on very specific topics. So, this is a time-saving and efficient tool. Here are some short questions for the users:

Questions Set	
Name: Gender: Age:	
Question-1	Do check reviews of your product?
Answer	
Question-2	How much time do you spend analyzing reviews?
Answer	

Question-3	What tools do you usually use when analyzing reviews?
Answer	
Question-4	How much does analyzing a product review cost you?
Answer	
Question-5	How do you feel about automating the whole process of analyzing reviews?
Answer	
Question-6	Do you think review analysis is helpful?
Answer	

7.2 Rich Picture

A rich picture is a top-level overview of how the activities are connected in the system and how the user and actors interact with it.

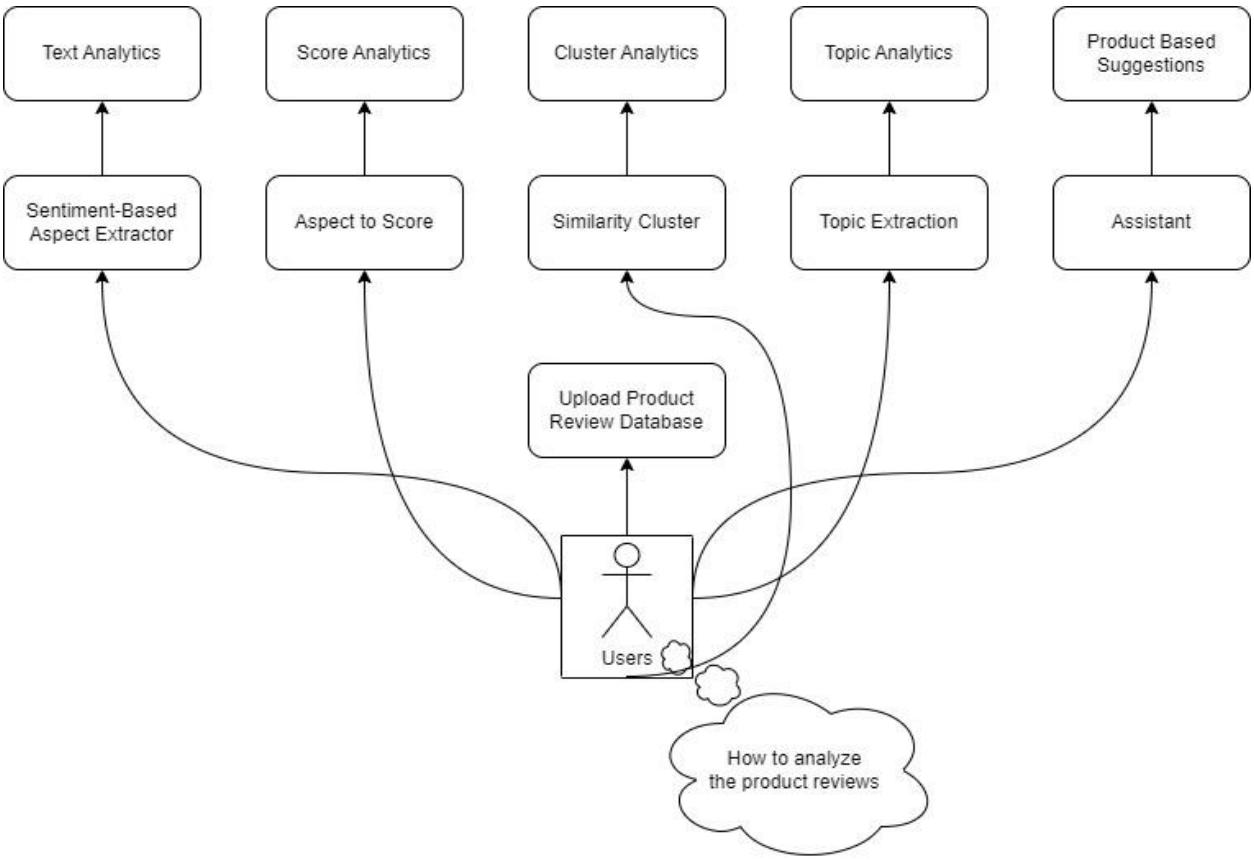


Figure 11: Rich Picture

Rich Picture Legends:

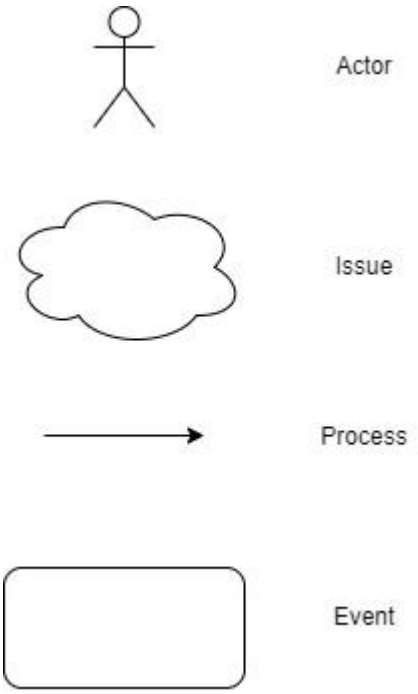


Figure 12: Rich Picture Legends

Key Actors:

There is one type actor for the proposed system:

- Users

Rich Picture Explanation:

Rich Picture is a high-level overview of a system. In the rich picture for the proposed system it is shown how a general user uses the different type of services provided by the system to solve a specific type of problem.

7.3 Specific Problem Area Identification

During the problem area identification phase, certain key issues are discovered using the information-gathering techniques above. The rich images also depict the system's business process. After all, there are several issues, the most serious of which are listed below:

- No application or automated process that specializes in review analysis
- Time required to analyze reviews for a single product
- A contradiction between review and rating in some cases
- The high-level report losses too much information or lacks information

7.4 Possible Solution

Multiple problem areas have been identified above which do indeed have solutions. Here is the solution for the problem areas:

- Since there are no web-based applications that focus mainly on review analysis this system can be a well-known solution.
- Making the process of review analysis fully automated
- Have some filters that can handle reviews with contradicting ratings.
- Have options to generate both high and low-level reports.

7.5 Overall Requirement List

There are mainly two types of requirements for a system

Functional

- File Upload
- Review analysis-focused services
- Report/Result
- Chat assistant

Non-Functional

- Easy-to-use system
- Information on the services
- High-level and low-level report generation
- A suggestion-based service

7.6 Technology to Be Implemented

This system can be implemented in two ways:

Client-server application

A client-server application is a piece of software that is installed on the user's computer. It denotes a client-related server. It transmits requests to a distant server, which is stored on that system. Here are the functionalities for client-server applications:

- The application needs to be installed on the user’s device
- The application might need to be updated if the system is updated
- The speed of the application will depend on the user’s device

Web server application

The web server program is kept on the server. The user may simply connect to the server through the internet by using a defined domain name. They can connect to the system from anywhere in the world. There is no need to install any equipment. They use a browser and the internet to access the service.

- No application is required to be installed
- Website-based application, the application can be accessed through a domain
- Requires internet connection
- No need to update anything even if the system gets updated

7.7 Justification and Recommendation

The web server application is best suited for the proposed system. The system uses heavy language models which requires a lot of processing power. This processing does

take a lot of time but can be expedited if the host device has a powerful GPU with tensor cores. Since the system focuses on product reviews, the user might upload multiple product reviews so this will require a storage system that can hold huge amounts of data. Most of the users might not have this expensive hardware. So, web server applications are recommended for easy access and a fast and stable user experience.

CHAPTER 8- EXPLORATION

8.1 Old system module

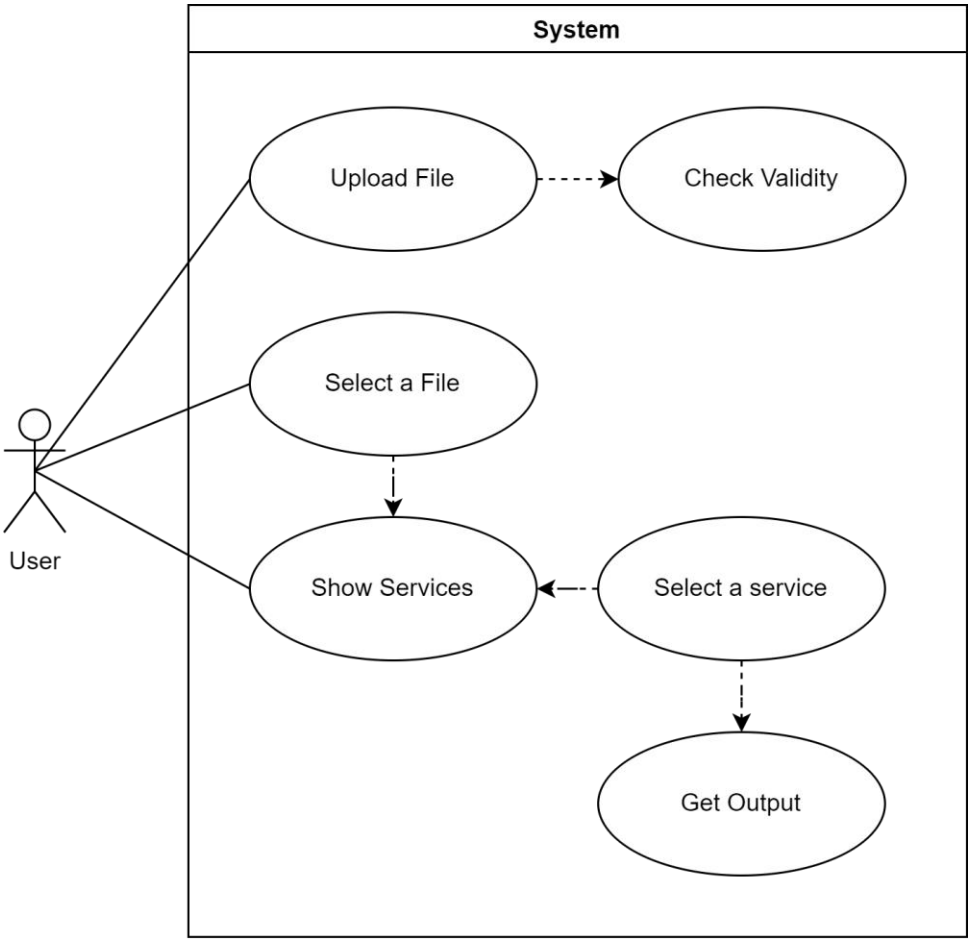


Figure 13: Old system Use Case Diagram

8.2 Activity Diagram

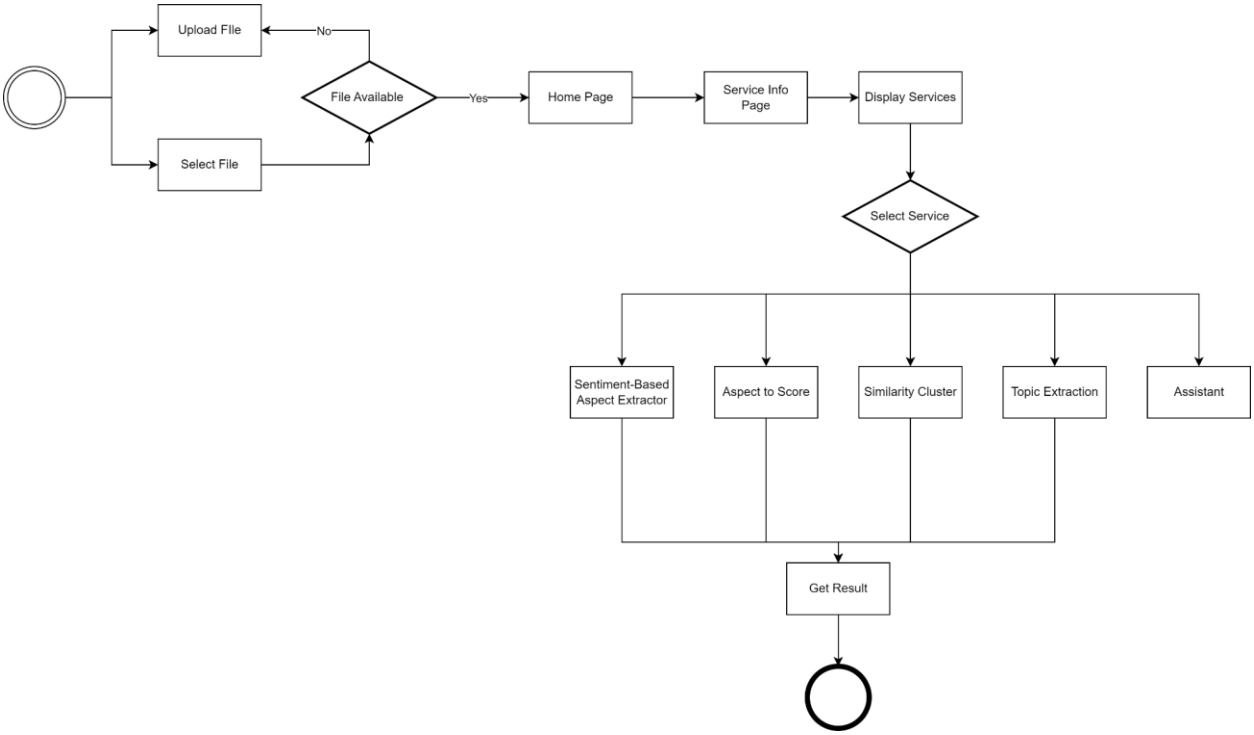


Figure 14: Activity Diagram

8.3 Full system use case diagram

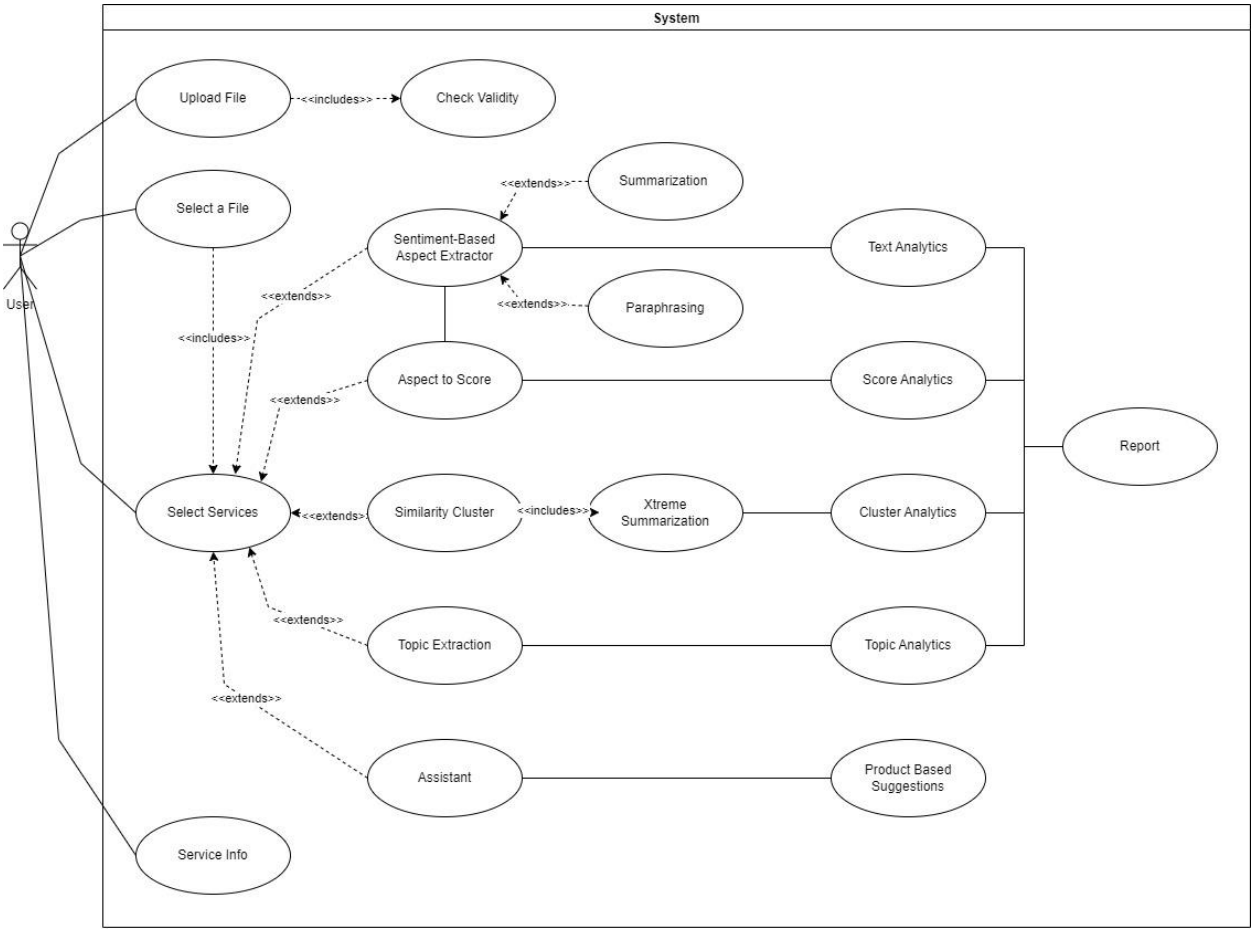


Figure 15: Use Case Diagram

8.4 Requirement Catalog

A requirement catalog is a structured and prioritized list of requirements for a system. The requirement catalog for the proposed system is given below following the standard format:

Requirement catalog for Upload and select product review.

Source	Sign off	Priority	Requirement ID

User	All users	Must	M-001
Functional requirement **Upload and select product review** To use the system, each user must upload a minimum of one review database for a product.			
Non-Functional requirement			
Description	**Target value**	**Acceptance value**	**Comments**
Database uploads	3 per day	10 per day	

Requirement catalog for Service: Sentiment-Based Aspect Extractor

Source	Sign off	Priority	Requirement ID
User	All users	Must	M-003
Functional requirement **Service: Sentiment-Based Aspect Extractor** Users will be able to a low-level text analysis report for the product review			
Non-Functional requirement			
Description	**Target value**	**Acceptance value**	**Comments**
Show reports	10 per day	500 per day	

Requirement catalog for Service: Chat Bot

Source	Sign off	Priority	Requirement ID
User	All users	Must	S-002
Functional requirement **Service: Chat Bot** Users will be able to talk to assistant and get product-related suggestions and other services			
Non-Functional requirement			
Description	**Target value**	**Acceptance value**	**Comments**
Messages	20 per day	500 per day	

8.5 Prioritized Requirement List (PRL)

Prioritized Requirement is a method for determining which candidates for requirements should be prioritized. This ensures that the most important tasks are completed first. The prioritization requirement technique used for the proposed system is Moscow rules.

Following the Moscow rules the priority list for the proposed system is given below:

- **Must have requirements:**

ID	Requirements

M-001	Upload and select product review.
M-002	File Validity Check
M-003	Service: Sentiment-Based Aspect Extractor
M-004	Service: Aspect to Score
M-005	Service: Similarity Cluster
M-006	Service: Topic Extraction

- **Should have requirements:**

ID	Requirements
S-001	Switch Product Database
S-002	Service: Chat Bot

- **Could have requirements:**

ID	Requirements
C-001	Information about each service.

- **Would have requirements:**

ID	Requirements
W-001	Demo Review

W-002	Sentiment-Based Aspect Extractor customization option

8.6 Prototype of the new system

Here is a sample of the interface design for the proposed system:

File Selection Menu

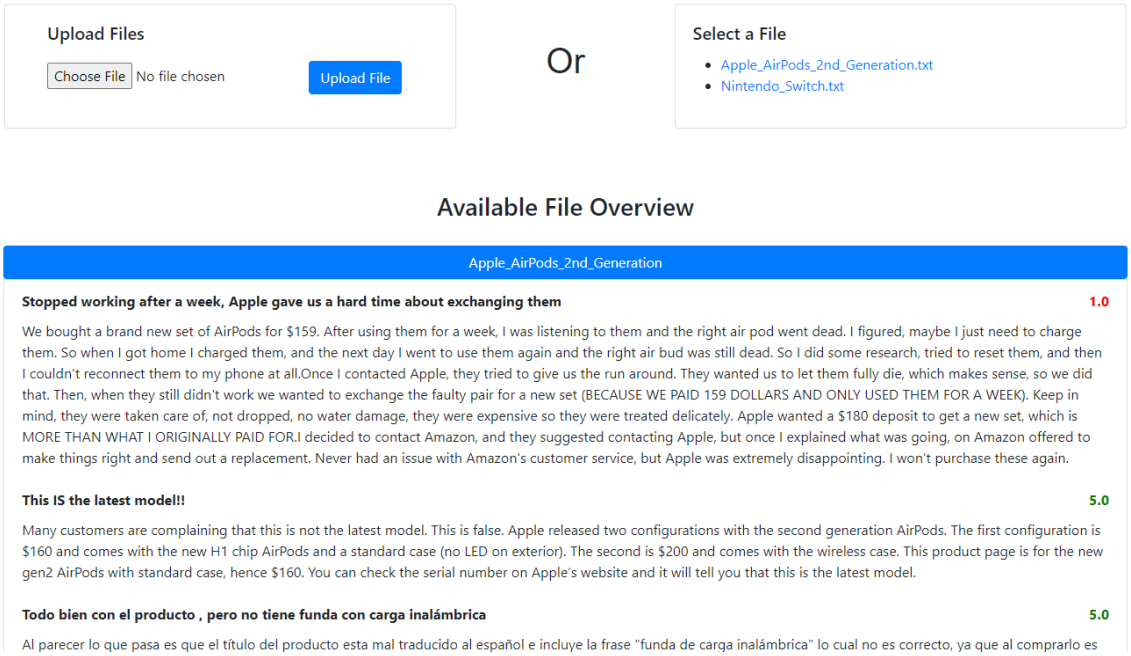


Figure 16: File Selection Menu Prototype

Service Menu

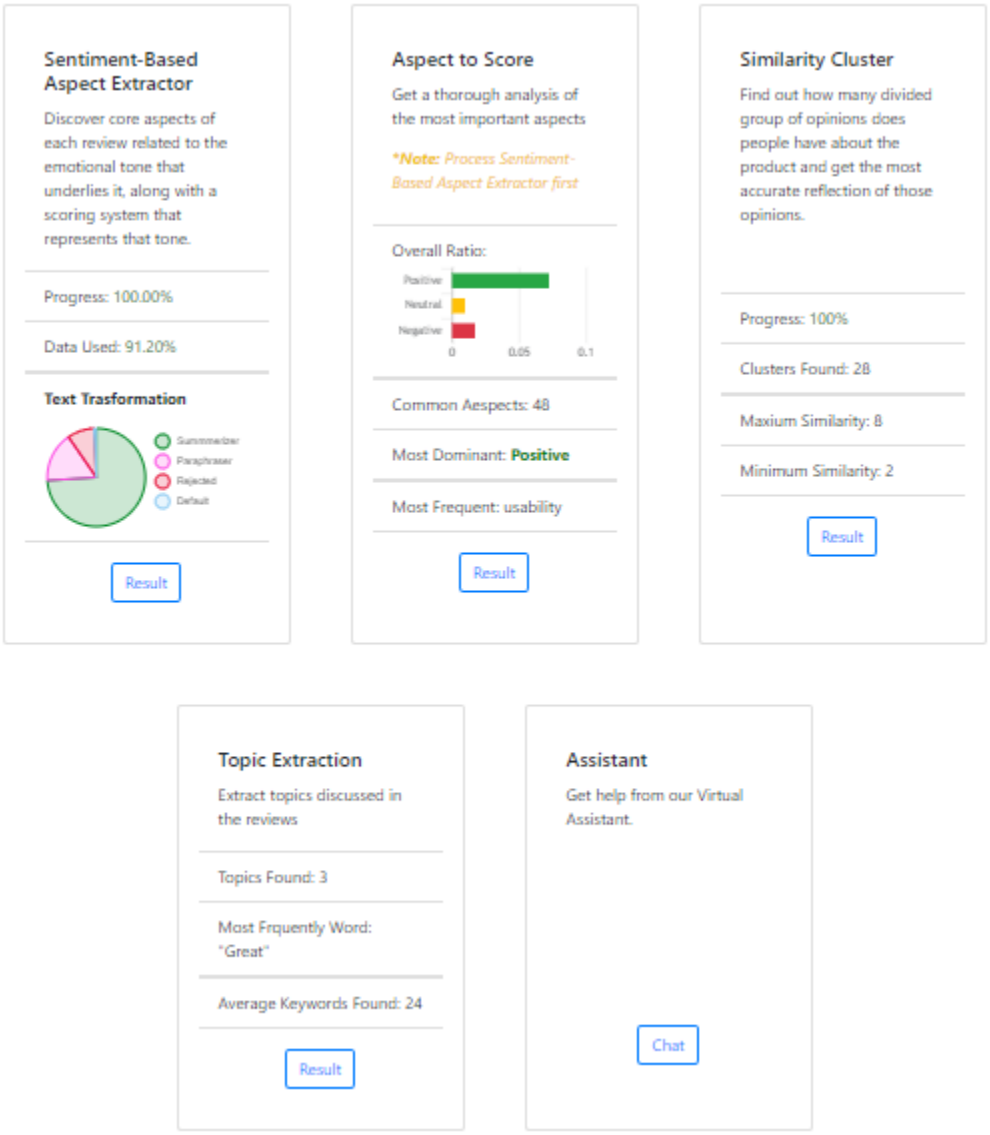


Figure 17: Service Menu Prototype

Sentiment-Based Aspect Extractor

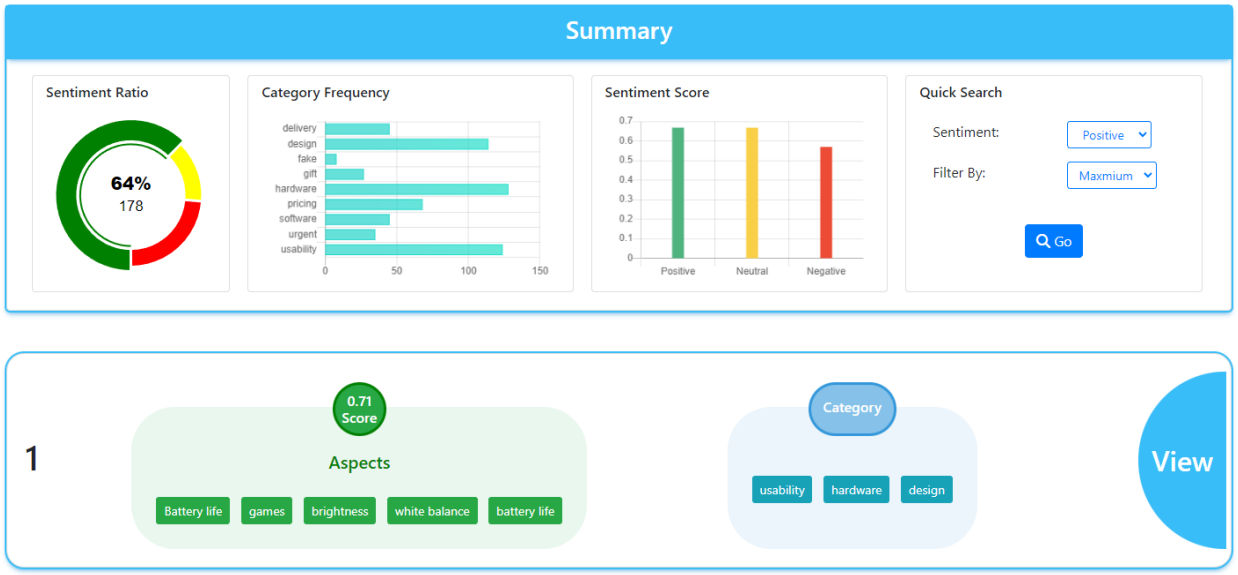


Figure 18: Sentiment-Based Aspect Extractor Page

Aspect to Score - Single aspect

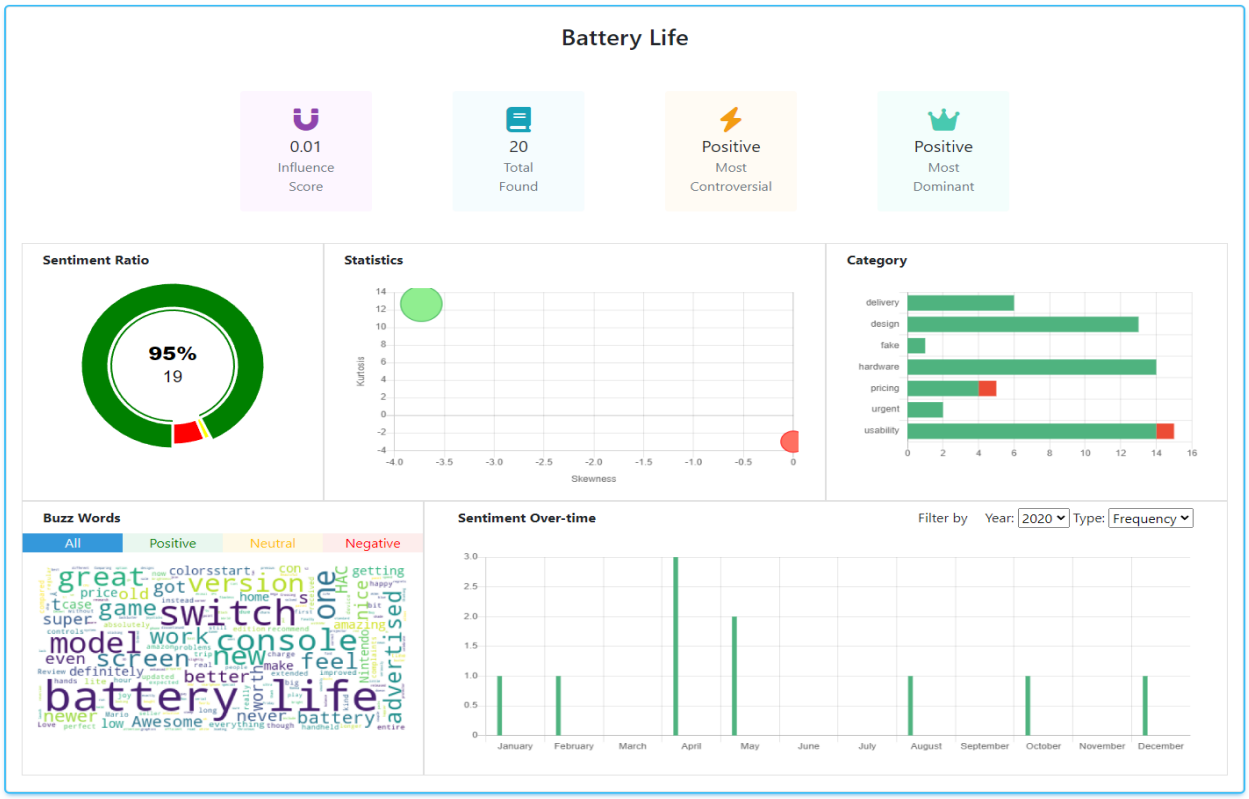


Figure 19: Aspect To Score Page

CHAPTER 9- ENGINEERING

9.1 New system module

The core modules for the proposed system are given below:

File Selection Module

Serial	User Action	Serial	System Action
1	The user clicks on Upload file	2	The system checks if the file is valid and then uploads data to the database
3	The user selects a file (After uploading)	4	The system sets the file as the current dataset and redirects to the model information page

Models Module

Serial	User Action	Serial	System Action
1	User click result on Sentiment-Based Aspect Extractor	2	The system recognizes categories for each text review, classifies aspects, and assigns a score associated with the label.
3	User clicks result on Aspect to Score	4	The system provides visual analytics for important aspects
5	User click result on Similarity Cluster	6	The system compresses each review to a single line and clusters them based on their similarity
7	User click result on Topic Extraction	8	The system performs topic modeling to extract topic keywords from the selected dataset
9	User click to chat on Assistant	10	The system connects with a chatbot

9.2 Use-case diagram

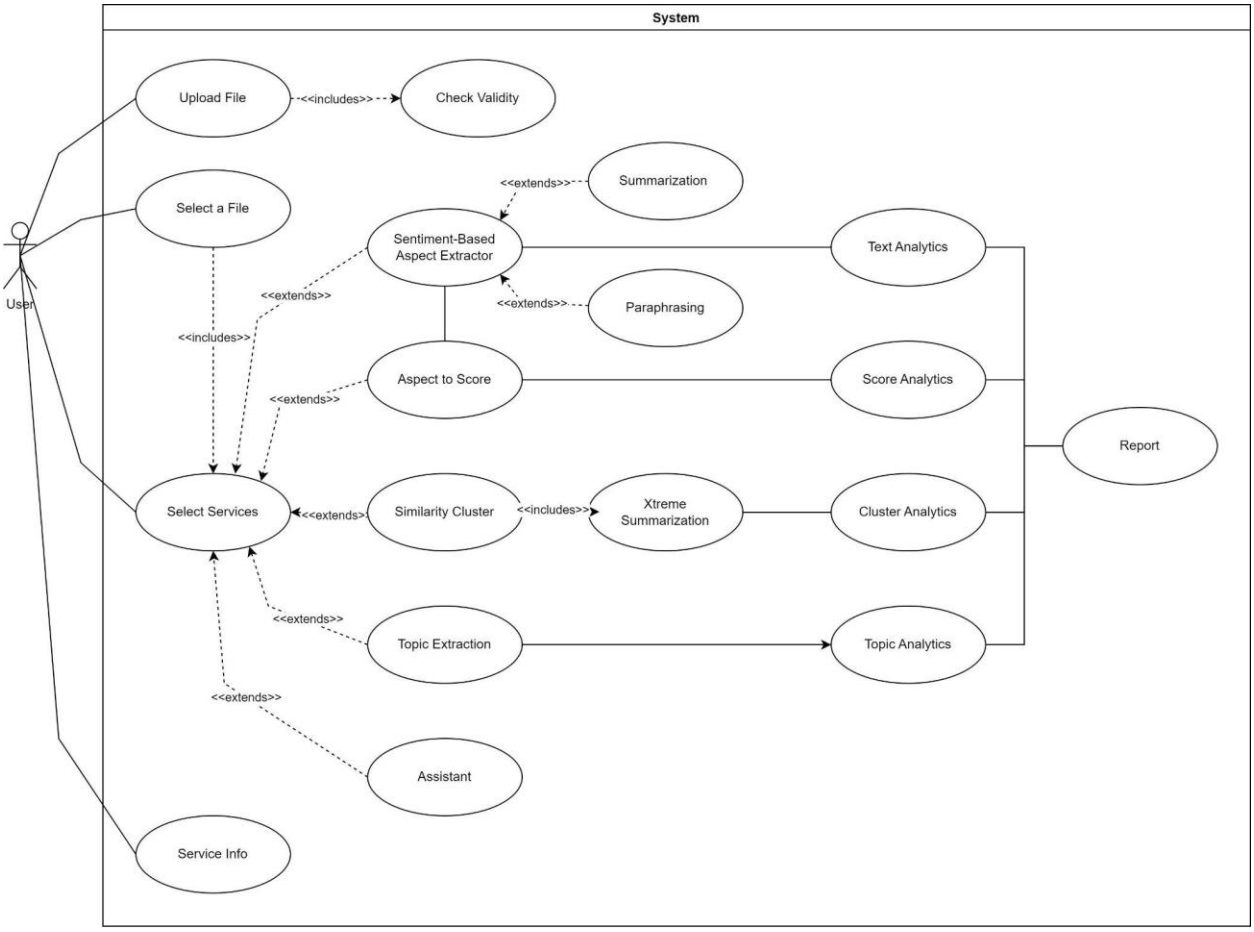


Figure 20: Use Case Diagram

9.3 ERD diagram

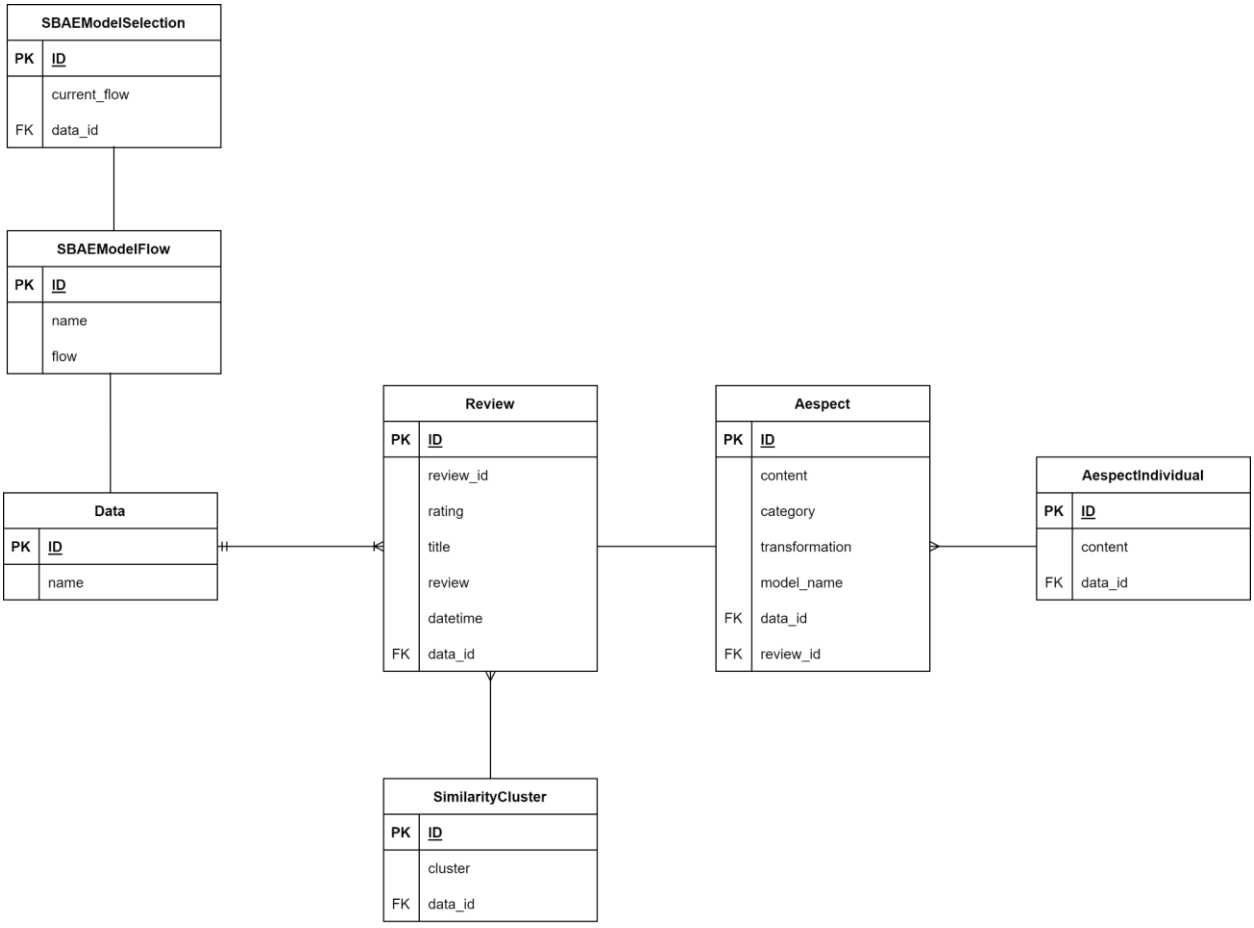


Figure 21: ERD Diagram

9.5 Sequence Diagram

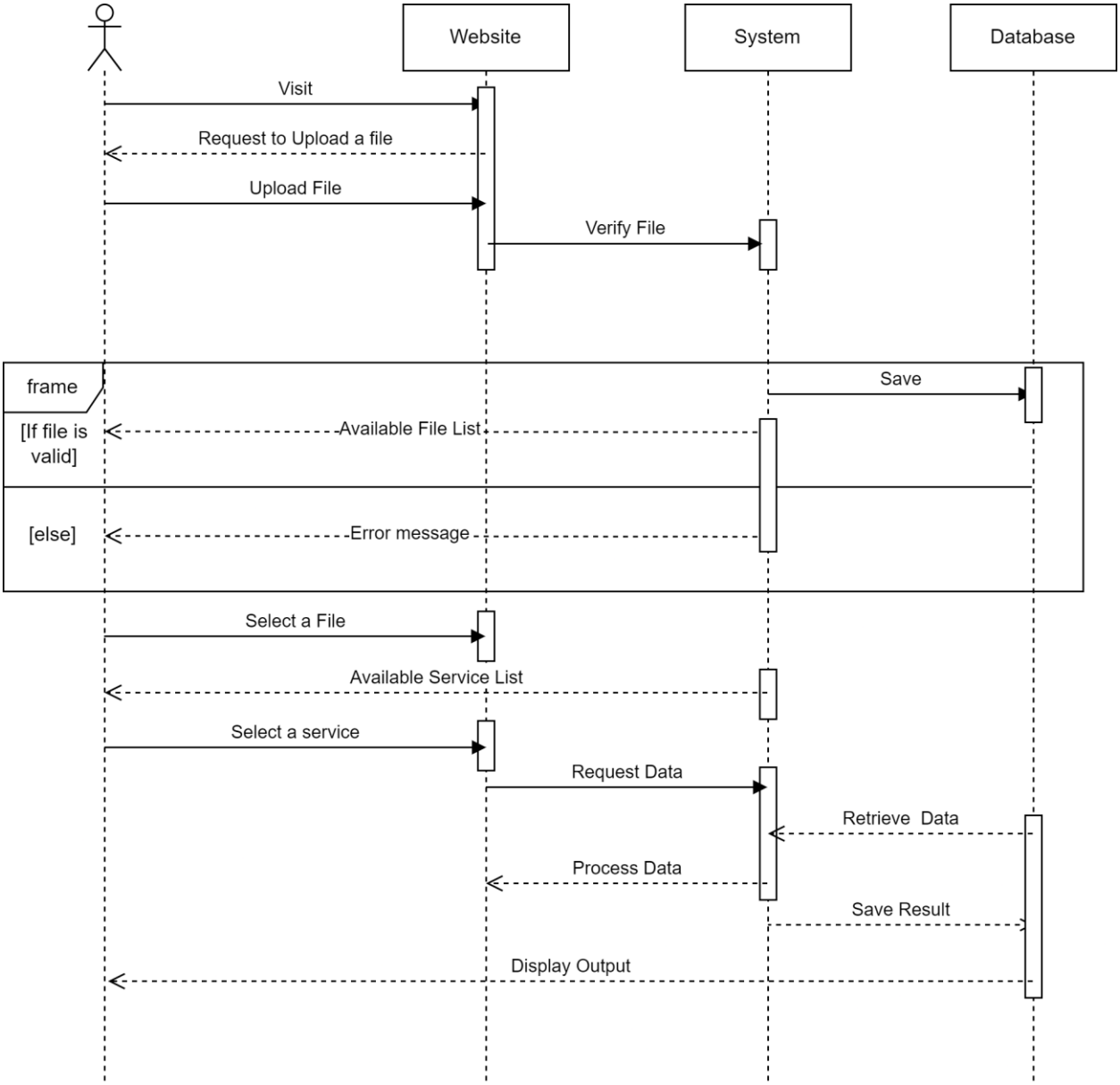


Figure 22: Sequence Diagram

9.6 Component Diagram

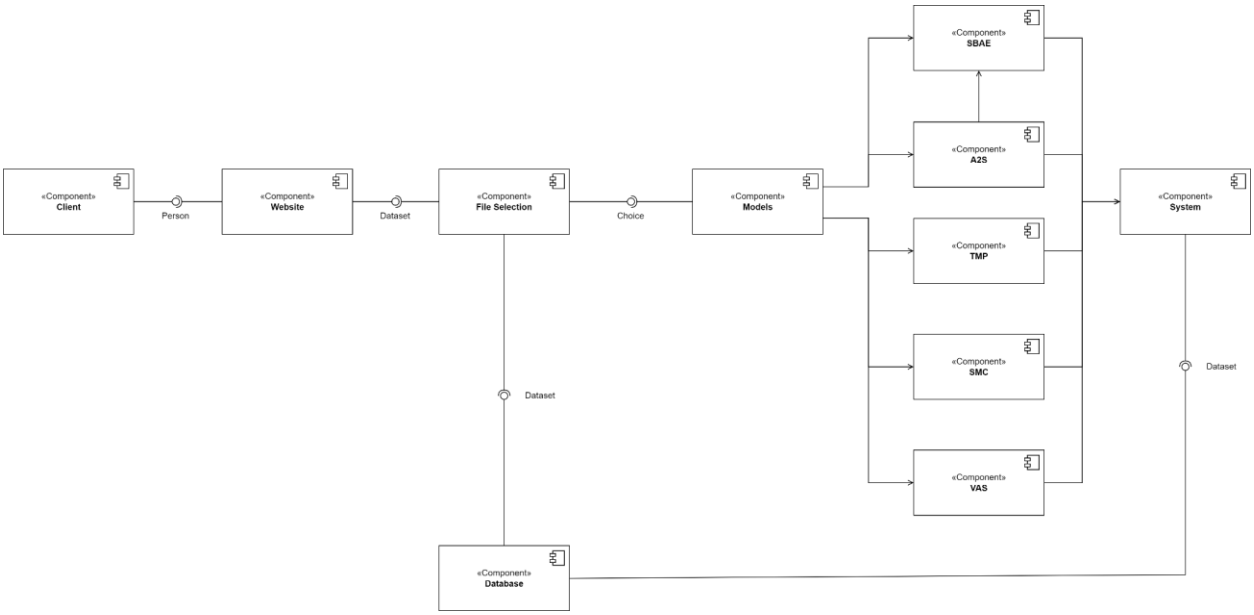


Figure 23: Component Diagram

9.7 Deployment diagram

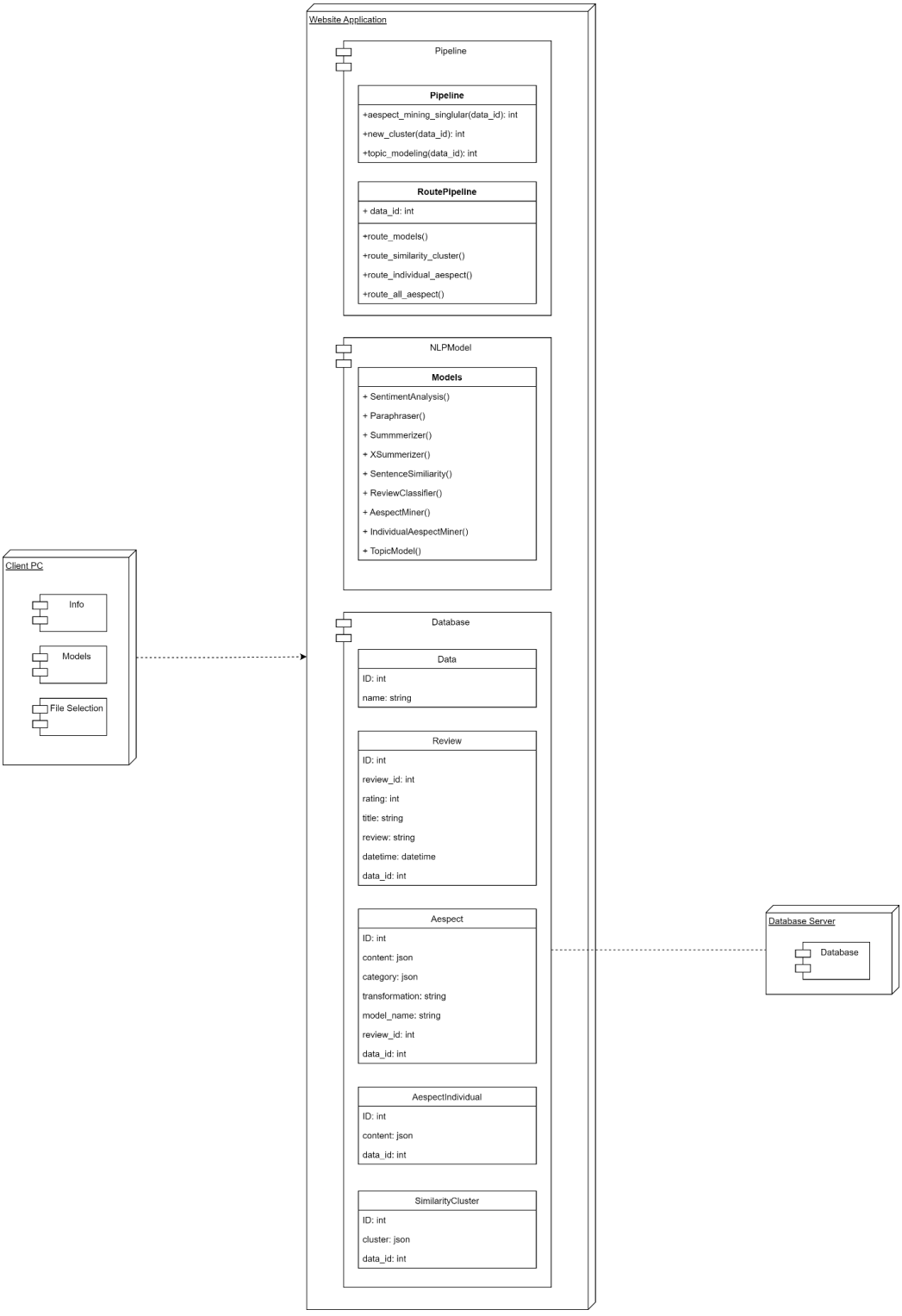


Figure 24: Deployment Diagram

9.8 New System Interface Design

Home page:

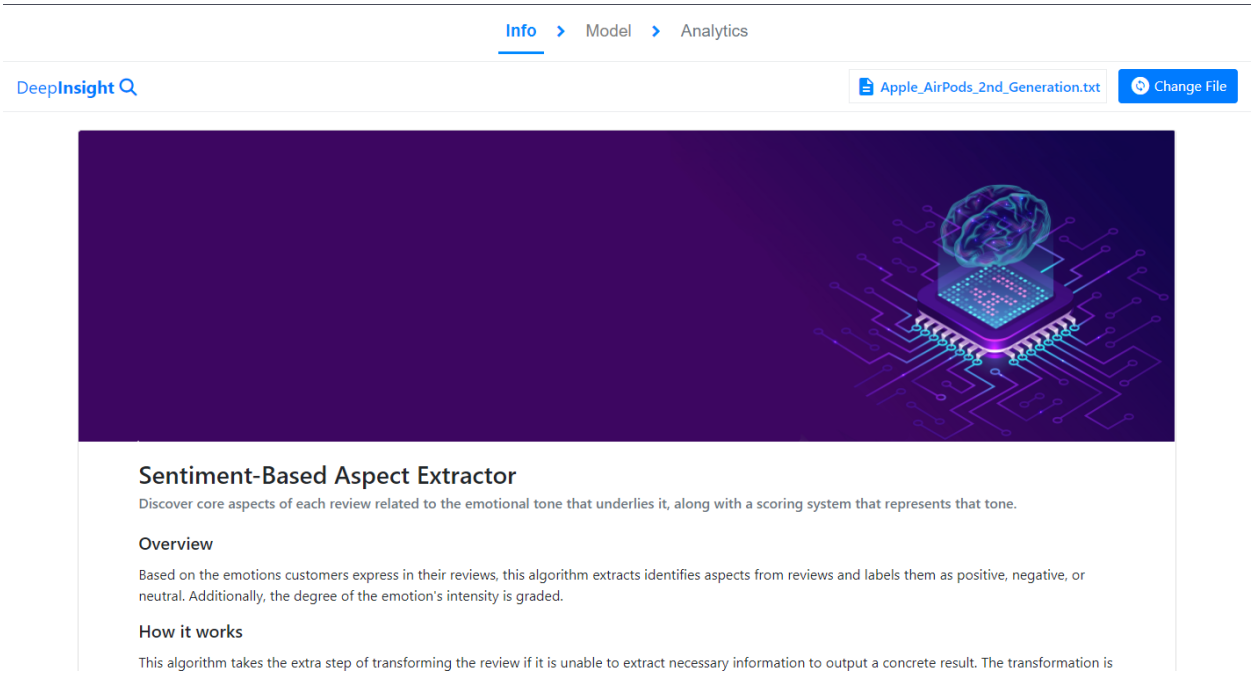


Figure 25: Home Page Design

Models/Services page:

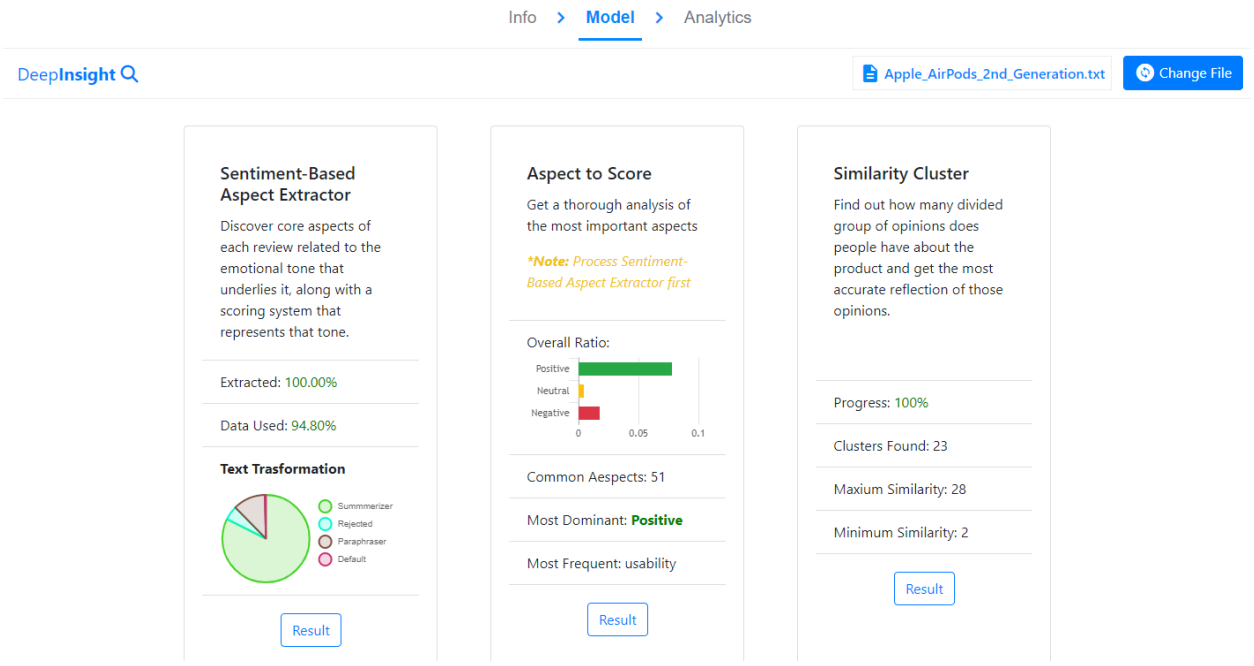


Figure 26: Models or Services Page

Sentiment-Based Aspect Extractor page:

Figure 27: Sentiment-Based Aspect Extractor Page

Aspect To Score page:

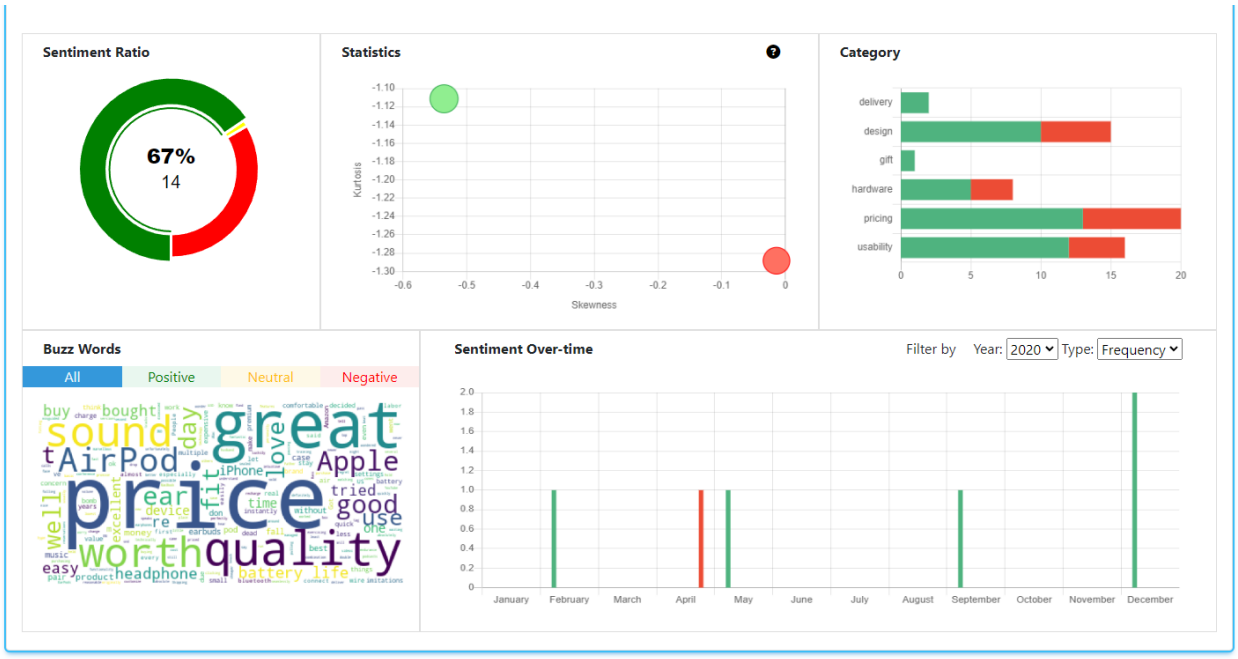
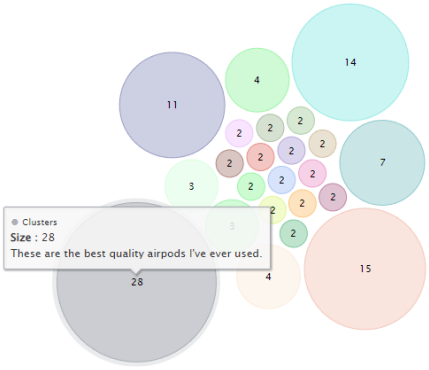


Figure 28: Aspect to Score Page

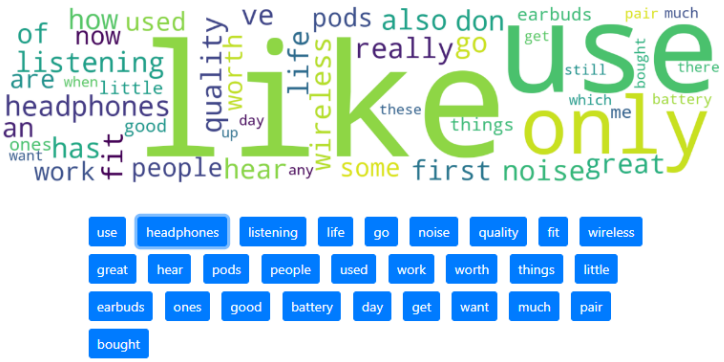
Similarity Cluster page:



Review	Similarity Found
• These are the best quality airpods I've ever used.	**28**
• These are the best earbuds we've tried so far.	**15**
• I've been using these headphones for about a month now and I've been really happy with them.	**14**

Figure 29: Similarity Cluster Page

Topic Extraction page:



LOVE - But recharge twice a day.. Oh my gosh there was a short sale and my Hubby got these for me for Mother's Day and I could not be happier!!!!Pros:* NO WIRES (until you need to charge the case itself)* These are the only in-ear headphones I can use, everything else gives me headaches after a short while.* They fit snug and very rarely fall out.* Charges last half a day of continuous listening to music* Sound clarity is very good! Callers can hear me better than from my phone by itself because of the case, and much better than my car's mic.* Recharging is very short, less than 15 minutes to fully charge inside it's case - less than an hour and the case is good to recharge the ear pieces for a 2 day period without a wall socket.* LightweightCons:* White shows EVERYTHING! Every spec of dust, every fragment of ear wax will show up from space!* The charge on the ear pieces only seems to last 5-6 hours at most, maybe only 4 with non-stop listening at a higher volume* Sometimes the ear pieces lose battery and die even while not connected to anything, if not in their case.* Sometimes one earbud seems to lose charge before the other and I still am not sure why? Stereo sound?* These do not cancel sound very well - I don't believe that is part of the design anyway, but worth mentioning that if you are trying to drown out noisy, gossiping co-workers, airplane humming, or highway tire rumbling, you'll want OVER EAR headphones. .

Figure 30: Topic Extraction Page

Assistant page:

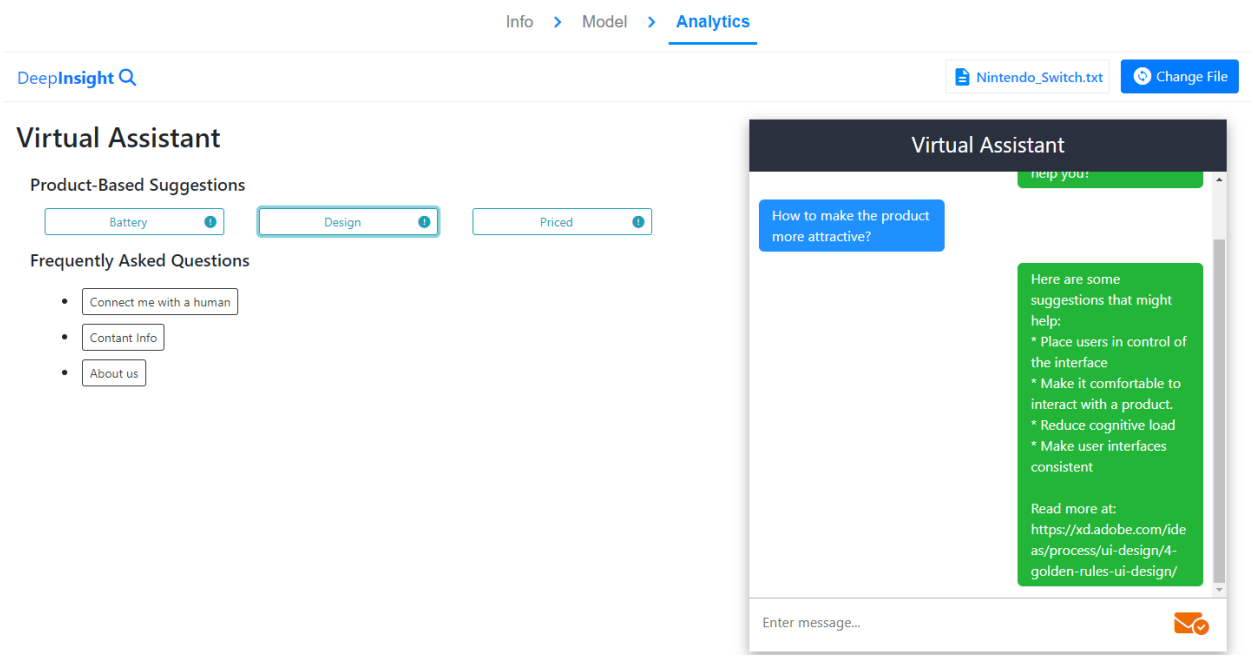


Figure 31: Assistant Page

9.9 Process Activity Flow Diagram

Here are the activity flow for some of the services of the system:

Sentiment-based Aspect Extractor Activity flow:

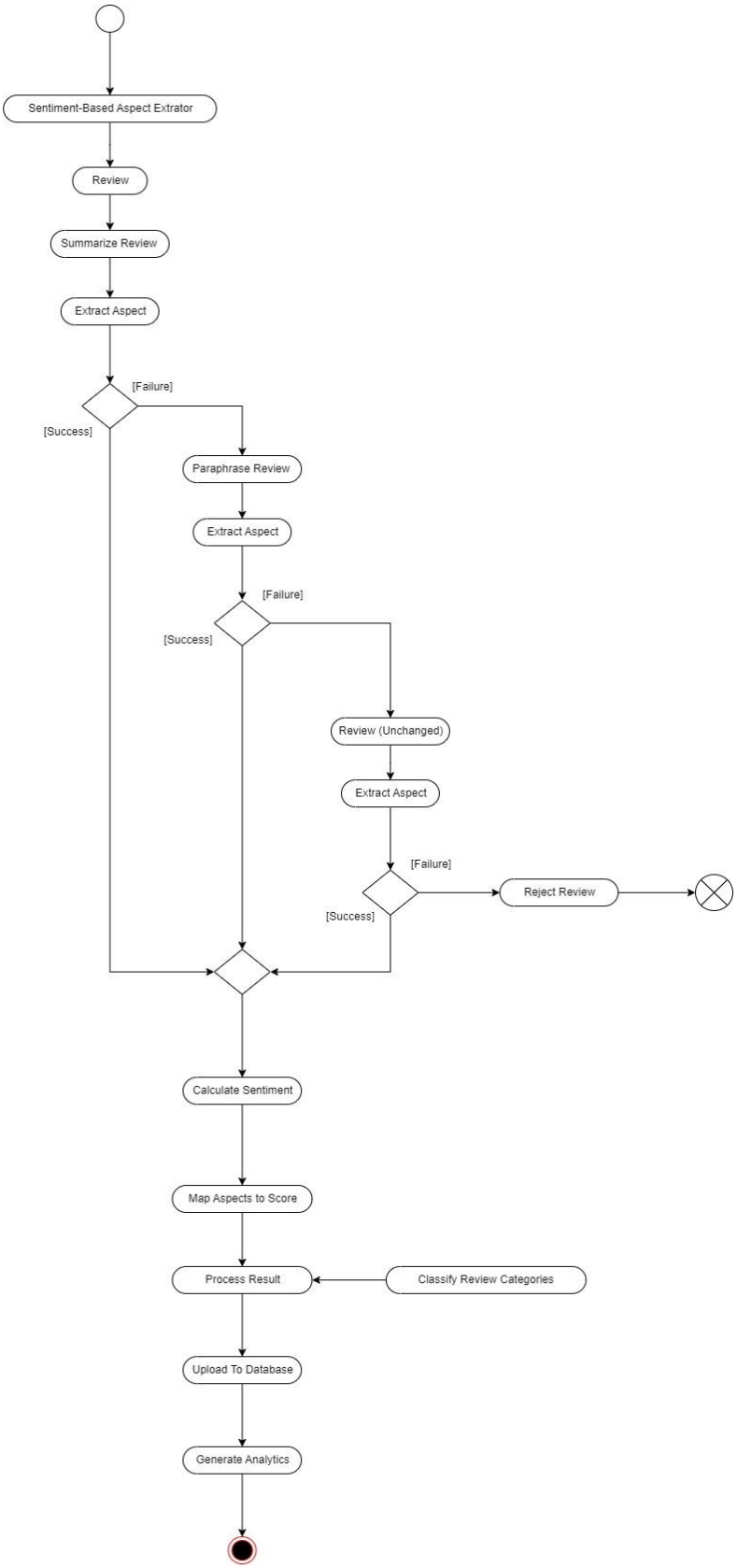


Figure 32: Activity Flow of Sentiment-Based Aspect Extractor

Aspect to score Activity flow:

Figure 33: Activity Flow of Aspect to Score

Similarity Cluster Activity flow:

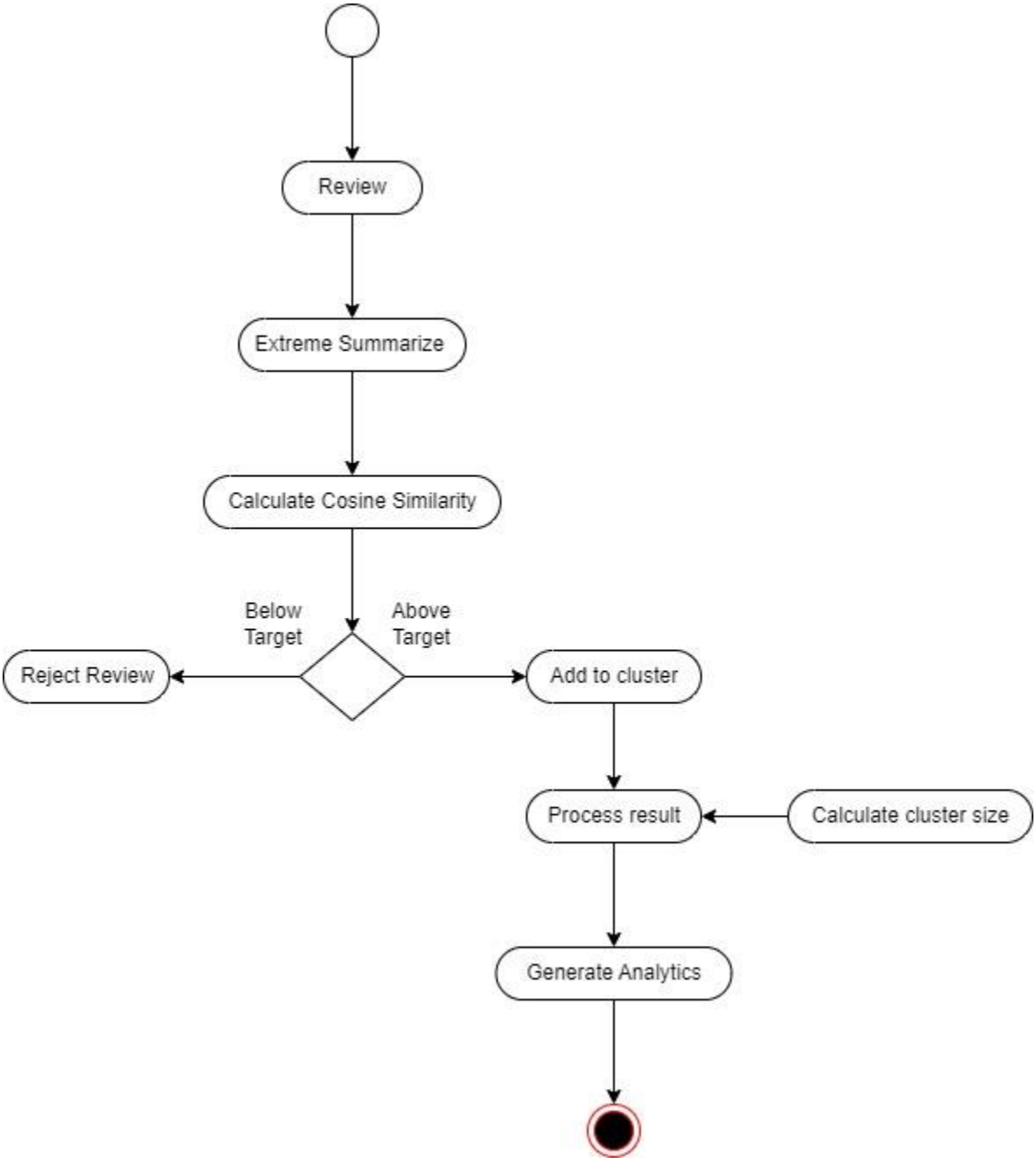


Figure 34: Activity Flow of Similarity Cluster

CHAPTER 10- DEPLOYMENT

10.1 Core Module Coding Sample

Python, HTML, CSS, JavaScript, Ajax and jQuery are used in the system's development, and some coding samples are provided below.

Routing Code Sample:

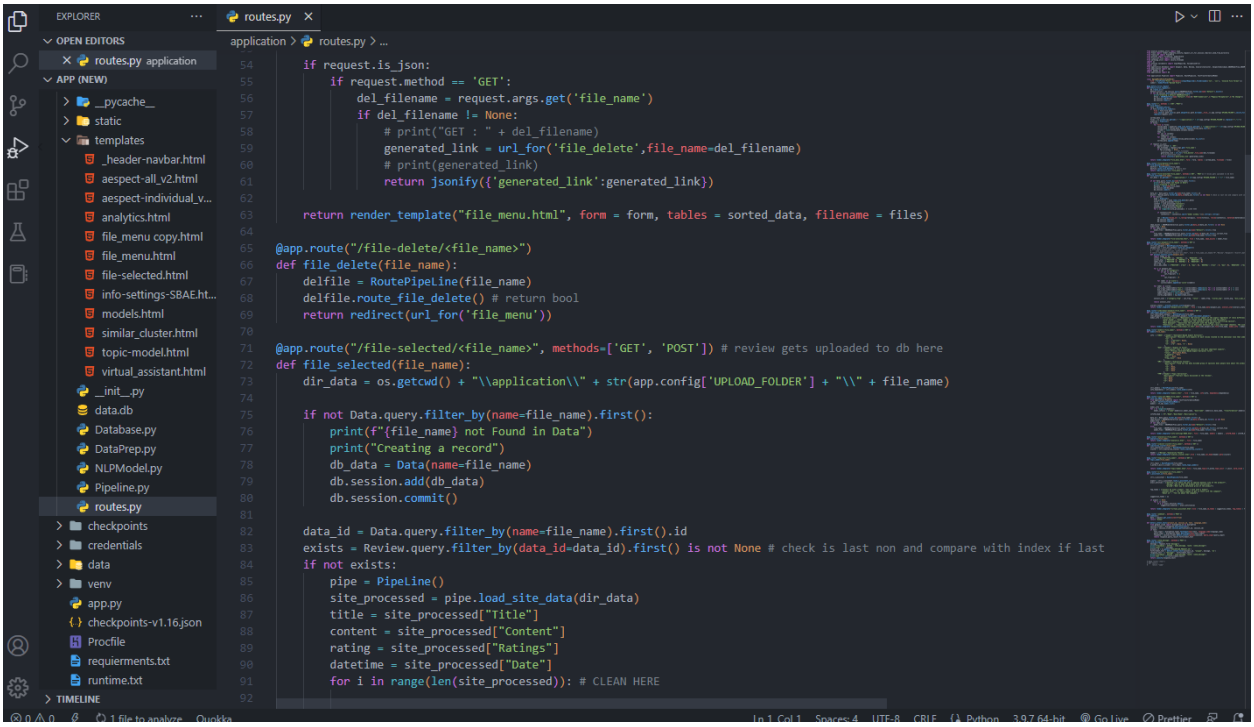


Figure 35: Python code sample for Routing

Natural Language Processing Pipeline Code Sample:

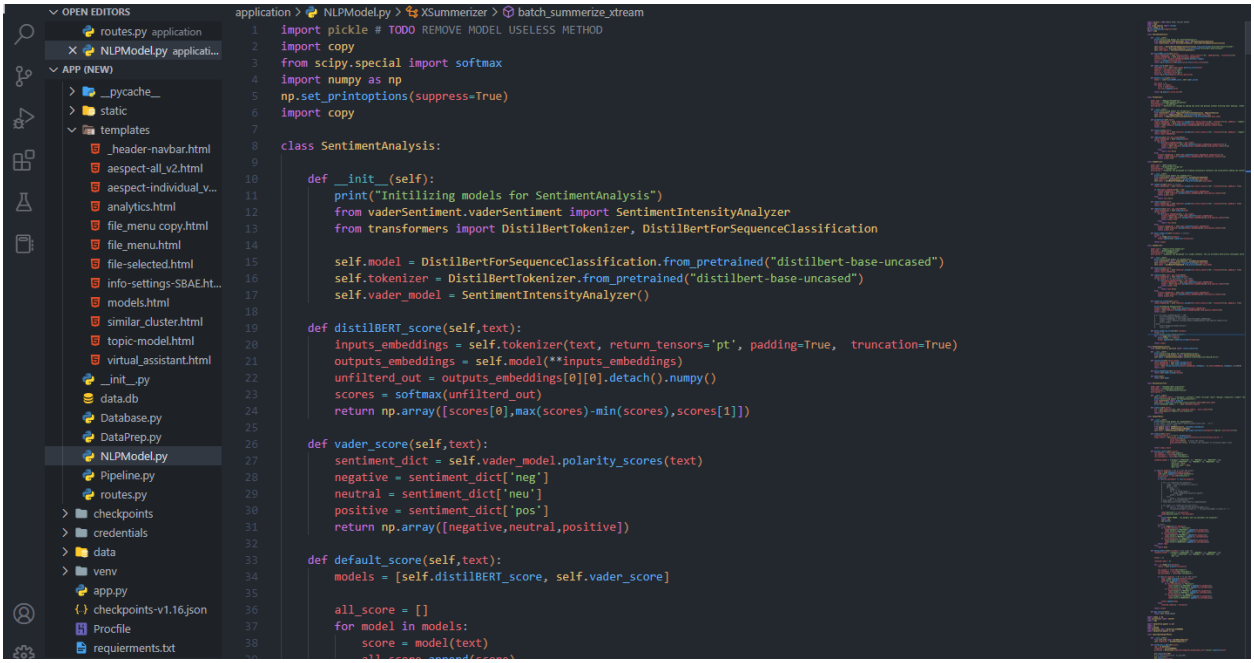


Figure 36: Python code sample for NLP pipeline

Database Handling Pipeline Code Sample:

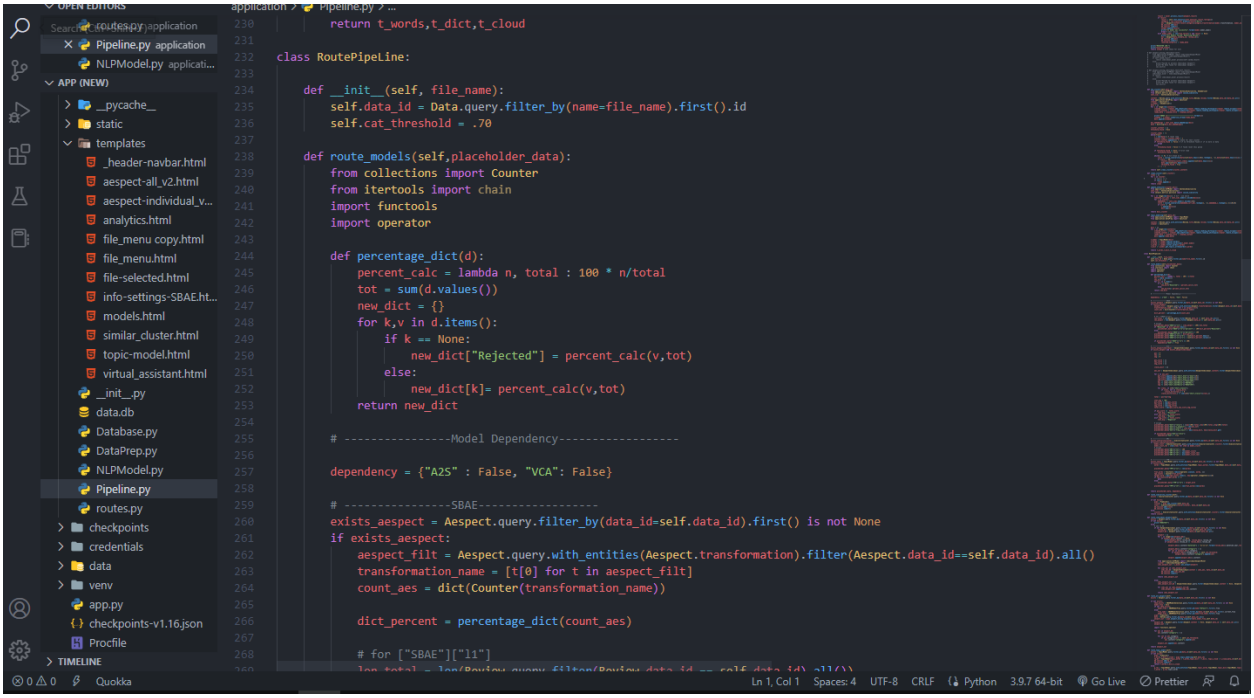


Figure 37: Python code sample for Database Handling

Database Code Sample:

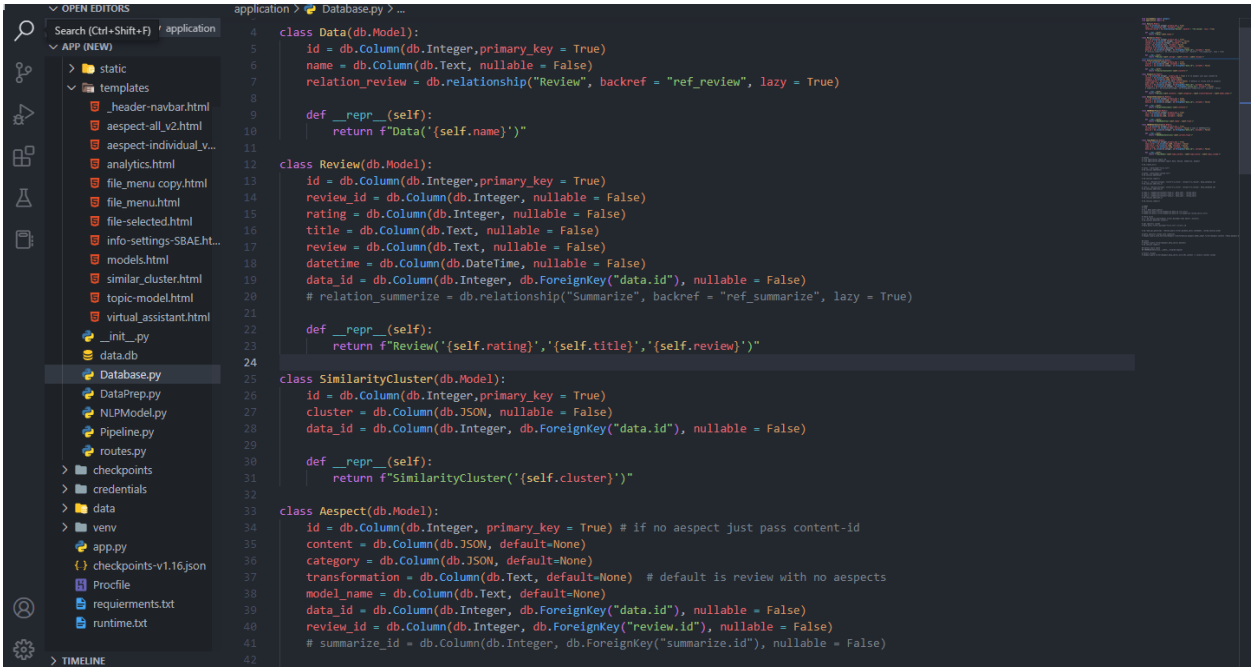


Figure 38: Python code sample for Database

Chart Generation Code Sample:

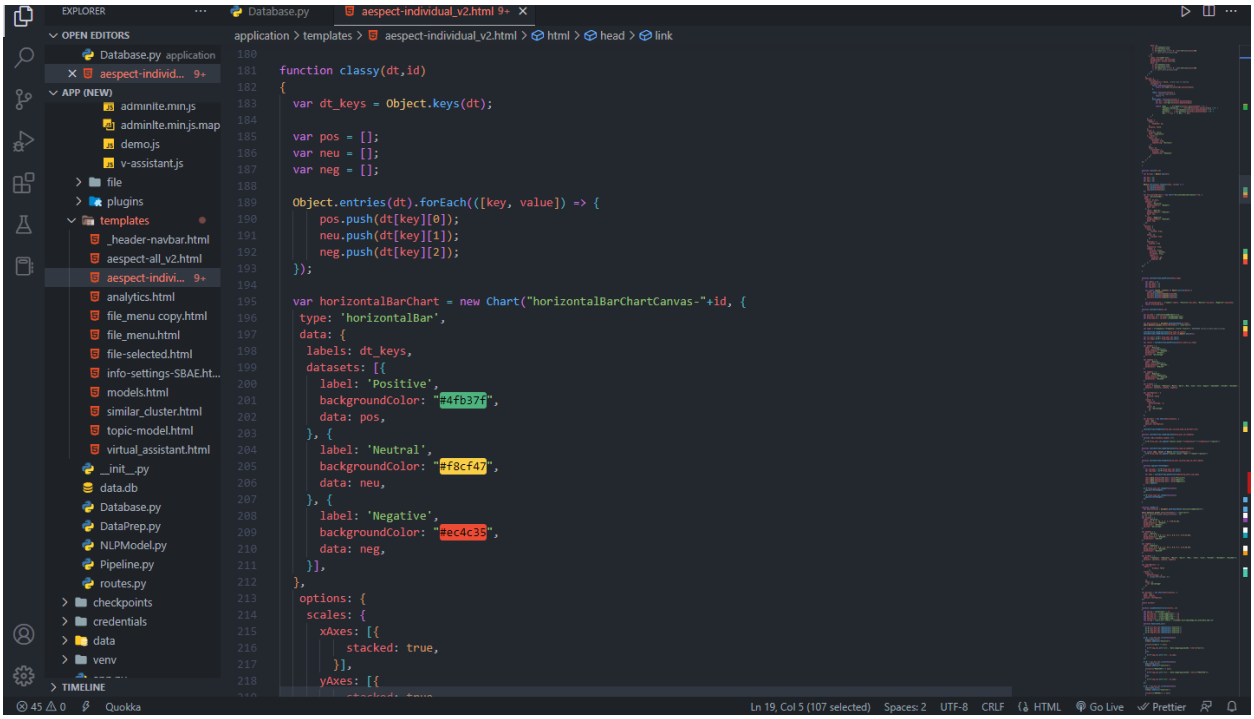


Figure 39: JavaScript code sample for chart generation

Virtual Assistant UI Event Handling Code Sample:

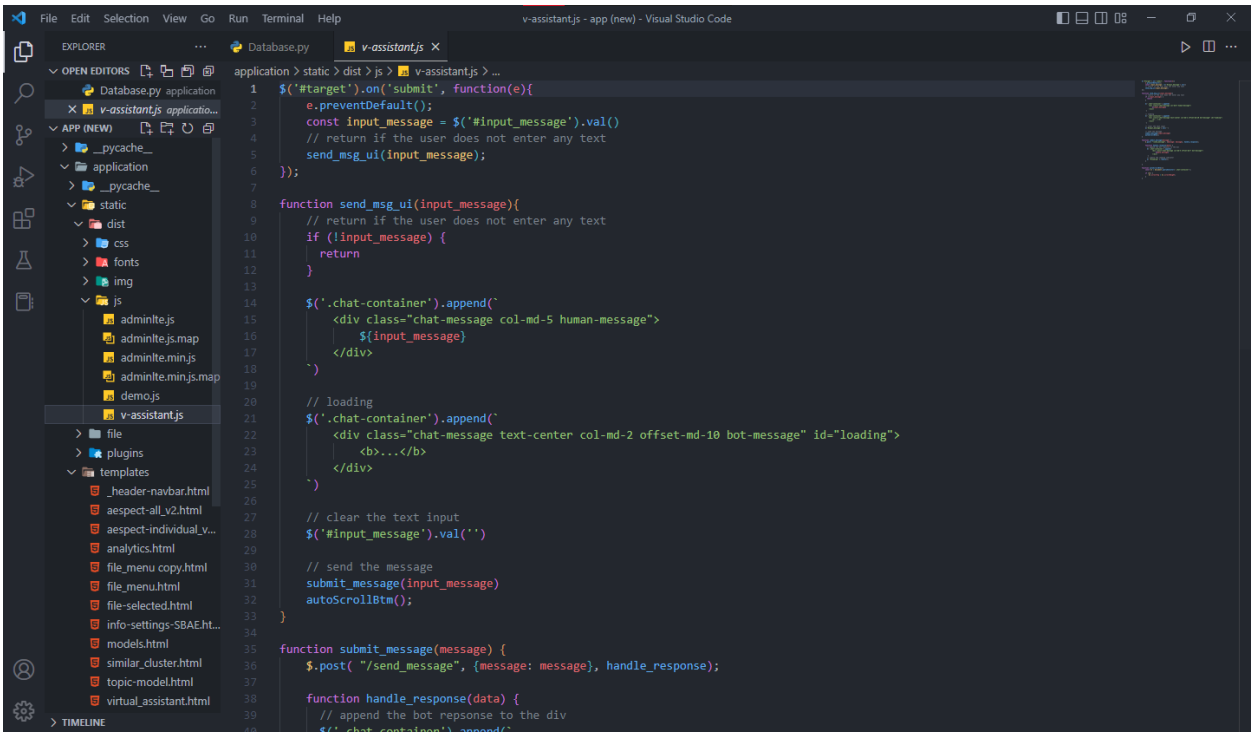


Figure 40: JavaScript code sample for Virtual Assistant Event Handling

10.2 Possible Break Down of the Problem

The general practice in software development is to break tasks down into smaller pieces before integrating them. This method simplifies the process and eliminates errors in development. The proposed system has multiple components that are responsible for completing specific tasks. So, breaking down these components will be a viable option. Here is the problem breakdown for the proposed system:

Database Creation

- Create a database for Deep Insight
- Create all the necessary tables
- Connect the table through foreign key

- Connect the database engine with what python

Interface Design

- Build the base structure for Deep Insight
- Establish proper routing to the links
- Use a light-contrast theme as the base theme
- Add graphs with the corresponding color for each label, and proper sizing

Models Processing

- Develop a pipeline connecting all the necessary procedures for each service

Database Operations

- Upload review data
- Upload processed review data
- Delete product review associated data
- Read processed data and insert them into the webpage
- Read processed data and generate graphs

CHAPTER 11- TESTING

11.1 Test Plan Acceptance

Functional Testing

Unit Testing

- File Upload Validation
- Quick Search option (in Sentiment-Based Aspect Extractor) proper navigation
- Alert the user if the uploaded file has an invalid format
- Buzzword options validation in Aspect to Score. Default image if no buzzword available in Aspect to Score
- Switch language model if the system fails to identify the aspect
- The final sentiment score should be the average of the rating score and sentiment score generated by the combined NLP models
- Database deletation check

Module Testing

- All services properly display information after processing is complete

Integration Testing

- The system picks up the correct database following the database selection menu
- Can only process Aspect to Score if Sentiment-Based Aspect Extractor progress is 100%
- After the services have been processed, the model page section should properly present an overview of each service.
- When the services have not been processed, present a default message in each overview section on the model page

Non-Functional Testing

Acceptance Testing

- Change to another database after already selecting one
- The influence score should be negative or close to 0 if the ratio is mostly negative and vice versa

Security Testing

- Properly Separated Database for each product

Accessibility Testing

- Testing with a professional Analyst
- Testing with a person who has no prior knowledge of statistics

Usability Testing

- Testing with admin

11.2 Test Case

Unit Test - test case (1)

Test Case Name	Unit test		
Test Class	FileUploadValidation		
Test Description	File Upload Validation		
Data Source	**Test Steps**	**Expected Result**	**Actual Result**
Developer	Click Submit without uploading a file	A warning requesting to upload a file first	Warning correctly displaying

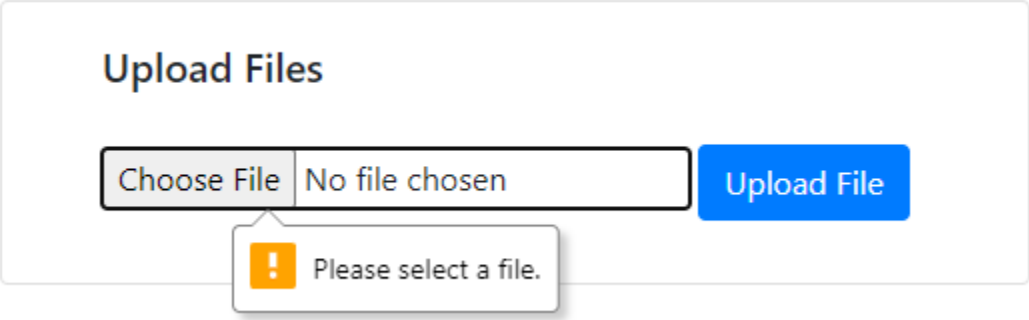


Figure 41: File Upload Validation

Unit Test - test case (2)

Test Case Name	Unit test		
Test method	search_filter()		
Test Description	Quick Search option proper navigation		
Data Source	**Test Steps**	**Expected Result**	**Actual Result**
Developer	Set Sentiment to positive and Filter to maximum when the current file is Nintendo Switch	The screen should be scrolled to the aspect card with the highest positive score value (aspect no. 146).	The screen scrolled to aspect - 146 which does have the highest positive value in the current dataset (Nintendo Switch)

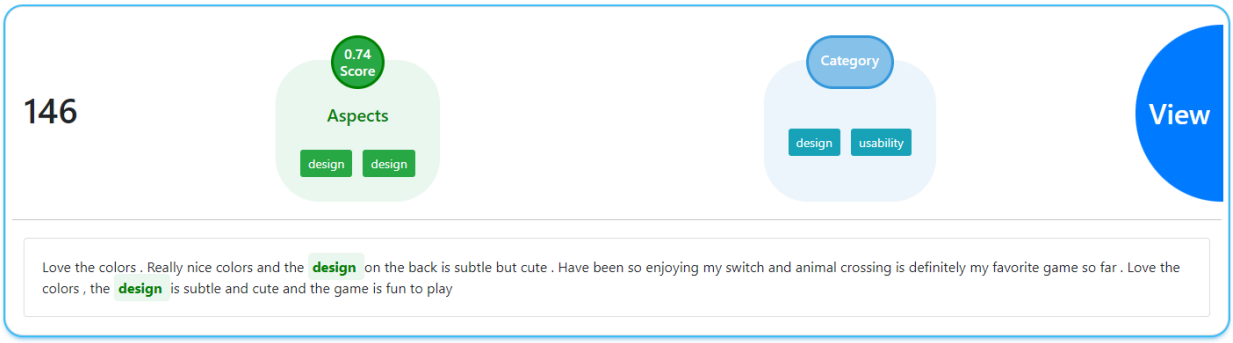


Figure 42: Search Filter validation

Unit Test - test case (3)

Test Case Name	Unit test		
Test Class	FileUploadValidation		
Test Description	File Format Validation		
Data Source	**Test Steps**	**Expected Result**	**Actual Result**
Developer	Try to upload a picture	A warning showing that the file could not be uploaded	Warning correctly displaying

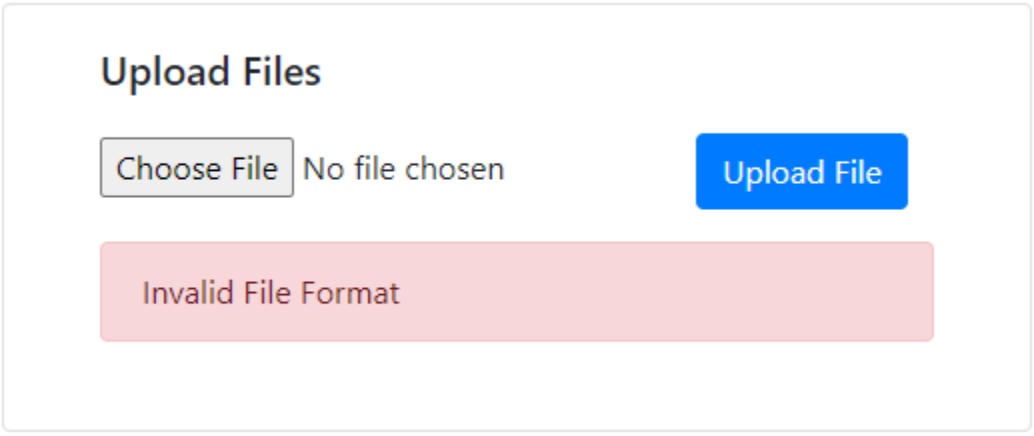


Figure 43: File Format validation

Unit Test - test case (4)

Test Case Name	Unit test		
Test method	cloudEventController		
Test Description	Buzzword options validation		
Data Source	**Test Steps**	**Expected Result**	**Actual Result**
Developer	Click a label that does not have any buzzwords	An image with the message "No Buzzwords Available" should appear	Clicked Neutral option for "Battery Life". No buzzword available image appeared.

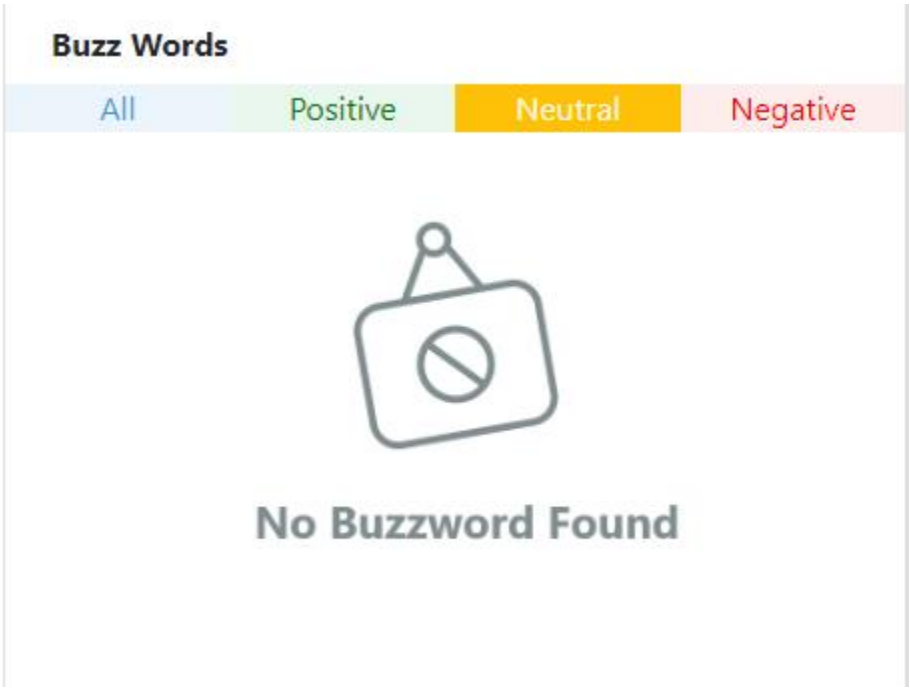


Figure 44: Buzzword option validation

Unit Test - test case (5)

Test Case Name	Unit test		
Test method	Pipeline.aespect_mining_singlular()		
Test Description	Text Transformation Validation		
Data Source	**Test Steps**	**Expected Result**	**Actual Result**
Developer	Click on the result in Sentiment-Based Aspect Extractor and check the log	The log will show a different model if the default "BART-Summarizer" failed	"Pegasus-Parapharser" was used for review no. 9

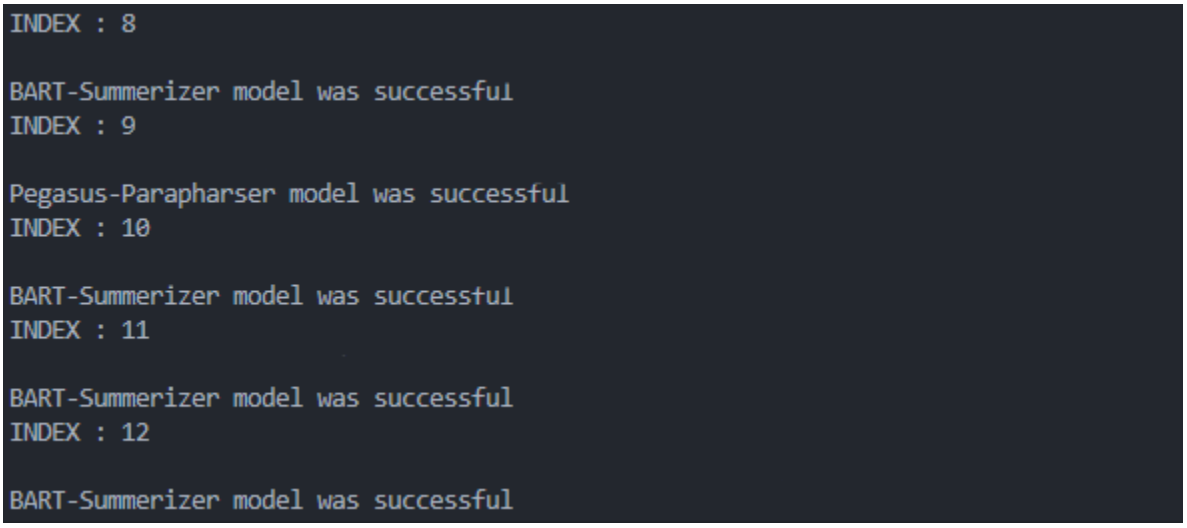


Figure 45: Text Transformation validation

Unit Test - test case (6)

Test Case Name	Unit test		
Test method	Pipeline.aespect_mining_singlular()		
Test Description	Sentiment score Validation		
Target Range	The first 10 and last 10 reviews		
Data Source	**Test Steps**	**Expected Result**	**Actual Result**
Developer	Click on result in Sentiment-Based Aspect Extractor and check the log	The “Average” value for each label should be the mean of “Original” and “NLP(Combined)”	The mean value was correct

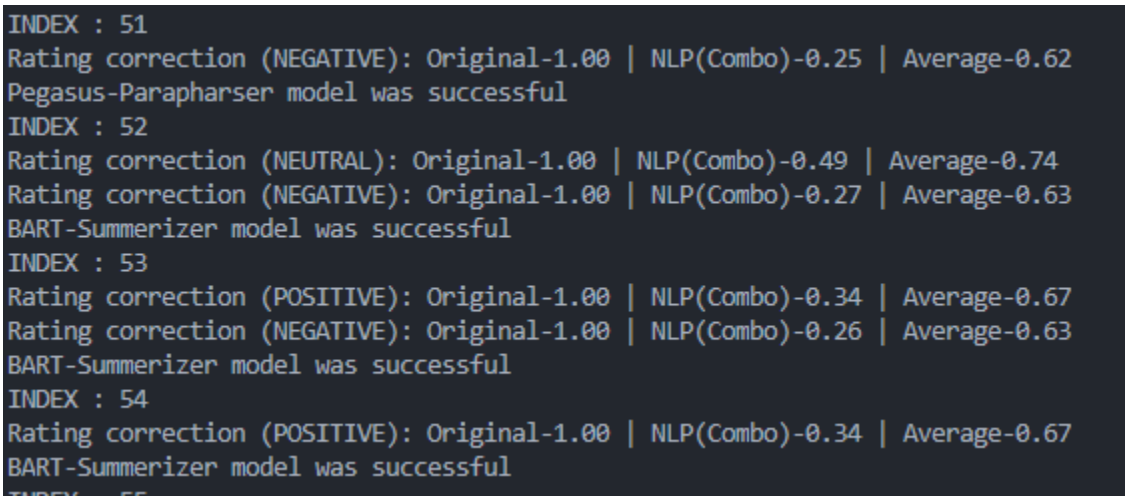


Figure 46: Sentiment Score Validation

Module Test - test case (1)

Test Case Name	Module test		
Test method	RoutePipeline.route_models()		
Test Description	All services properly display information after processing		
Data Source	**Test Steps**	**Expected Result**	**Actual Result**
Developer	Click the Result button for each service	Should be redirected to a page containing processed information	Was successfully redirected

Sentiment-Based Aspect Extractor

Figure 47: Sentiment-Based Aspect Extractor service validation

Aspect to Score

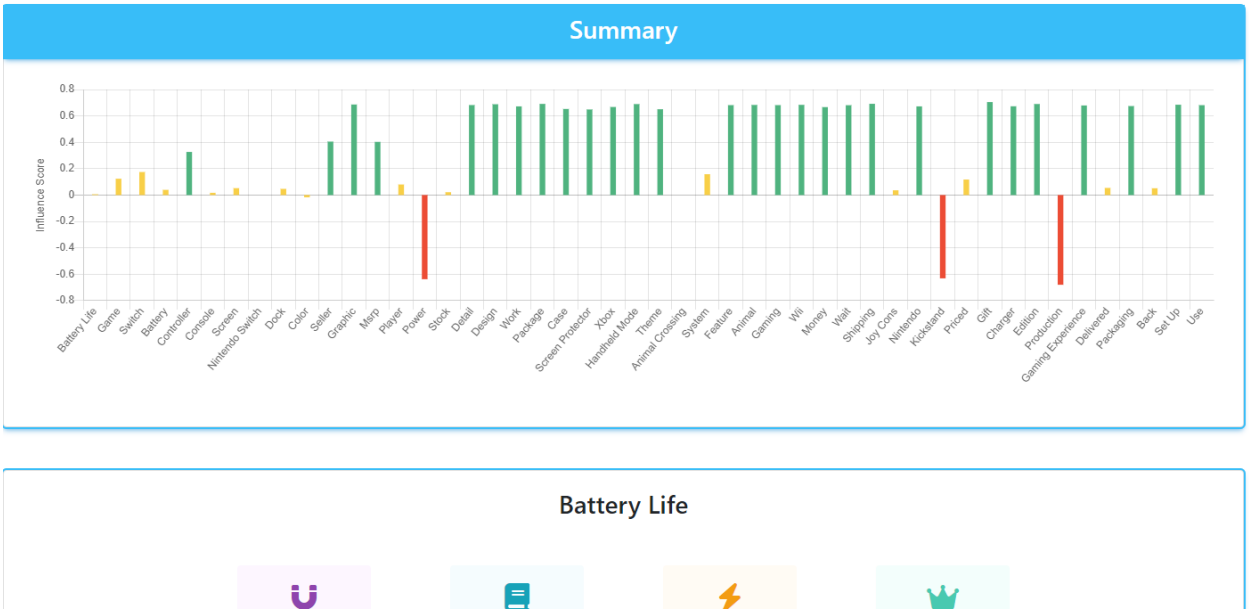


Figure 48: Aspect to score service validation

Similarity Cluster

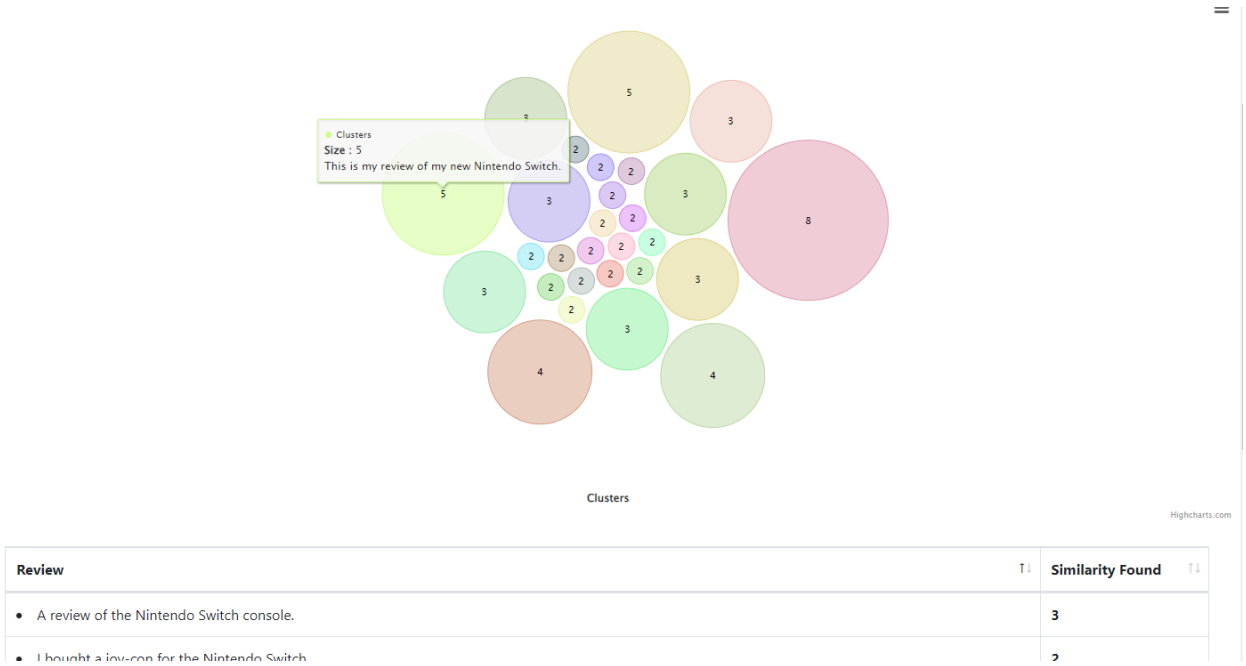


Figure 49: Similarity Cluster service validation

Topic Extraction

Figure 50: Topic Extraction service validation

Assistant

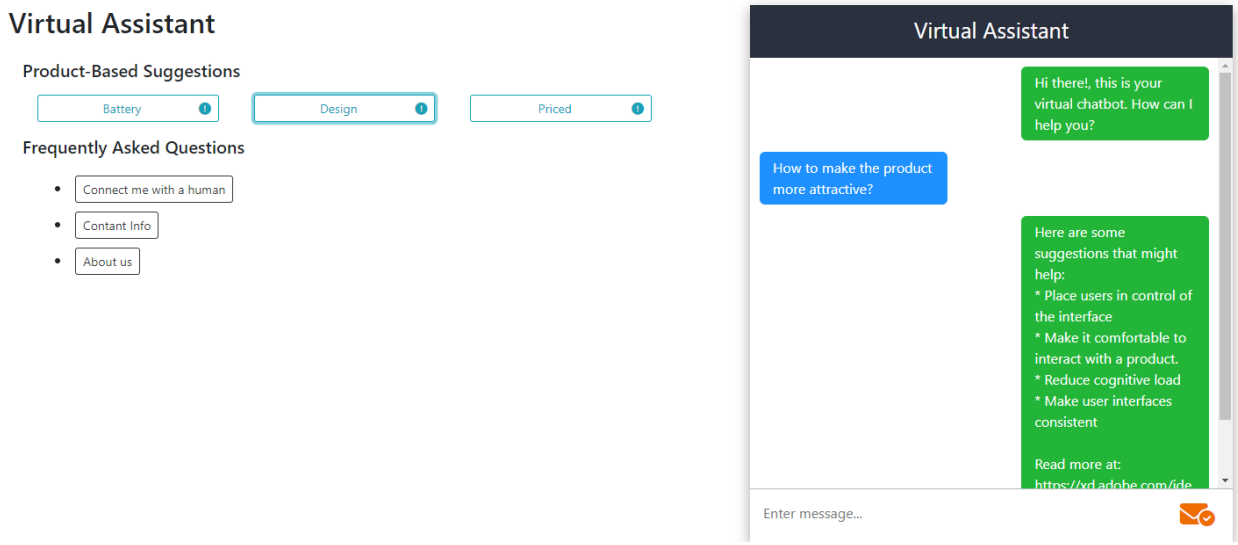


Figure 51: Assistant service validation

Integration Test - test case (1)

Test Case Name	Integration test		
Test Class	Current File		
Test Description	Correct Database selected throughout the session		
Data Source	**Test Steps**	**Expected Result**	**Actual Result**
Developer	Click the “Apple_Airpods” database on the homepage and navigate to other pages	The same name as the selected file should show on every other page	Did display the name “Apple_Airpods” on Info and Models page

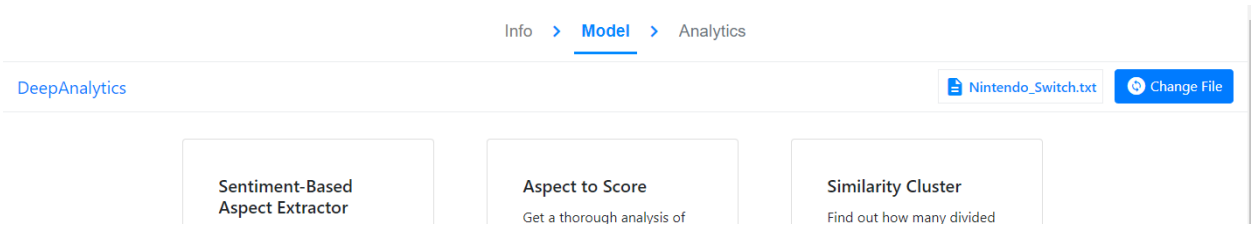


Figure 52: Database selection validation

Integration Test - test case (2)

Test Case Name	Integration test		
Test Class	RoutePipeline.route_models()		
Test Description	Services dependencies validation		
Data Source	**Test Steps**	**Expected Result**	**Actual Result**
Developer	Click on Aspect to Score result before clicking any of the other services	A pop-up should show up suggesting to process the Sentiment-Based Aspect Extractor first	Did correctly display a warning pop up

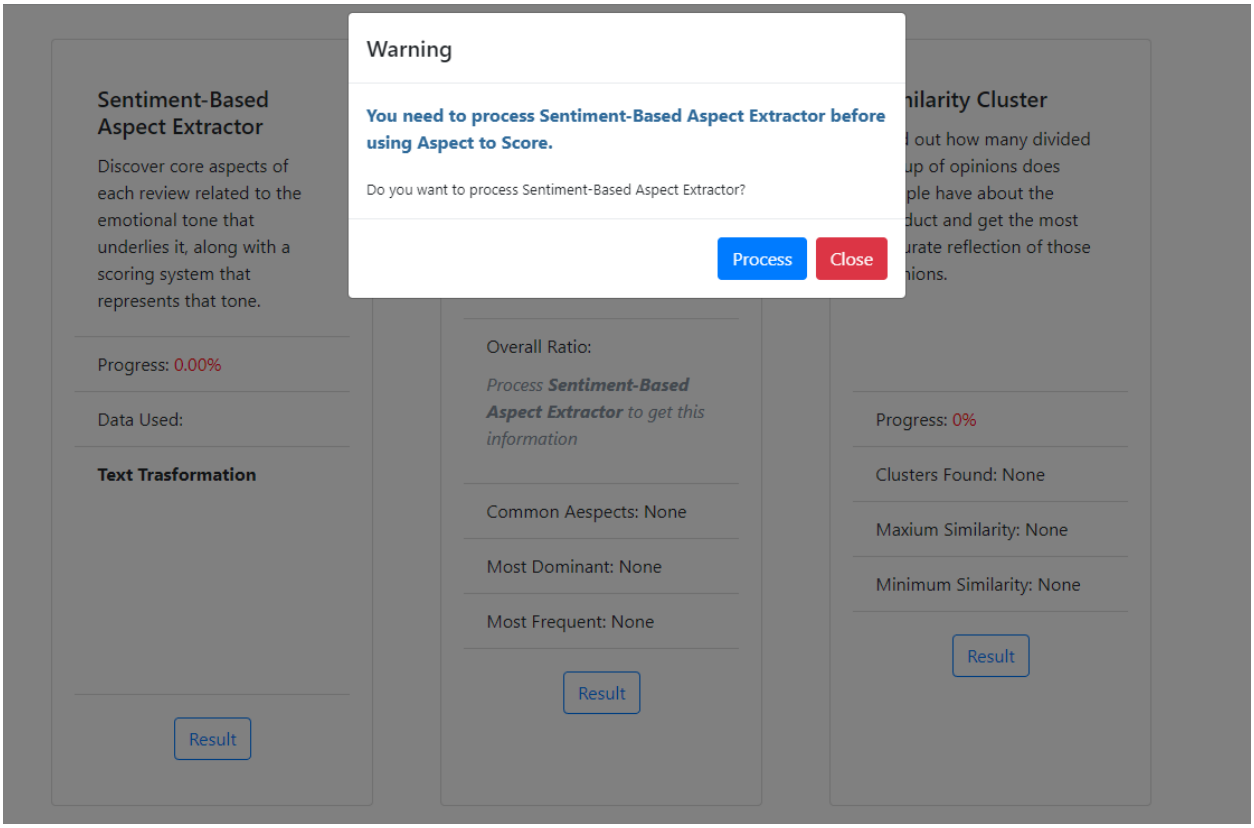


Figure 53: Service dependency validation

Integration Test - test case (3)

Test Case Name	Integration test		
Test Class	Models Info		
Test Description	Model page without processing any services		
Data Source	**Test Steps**	**Expected Result**	**Actual Result**
Developer	Goto model page and don't process any service	There should a piece of default information in the overview section for each model	Did display the default information

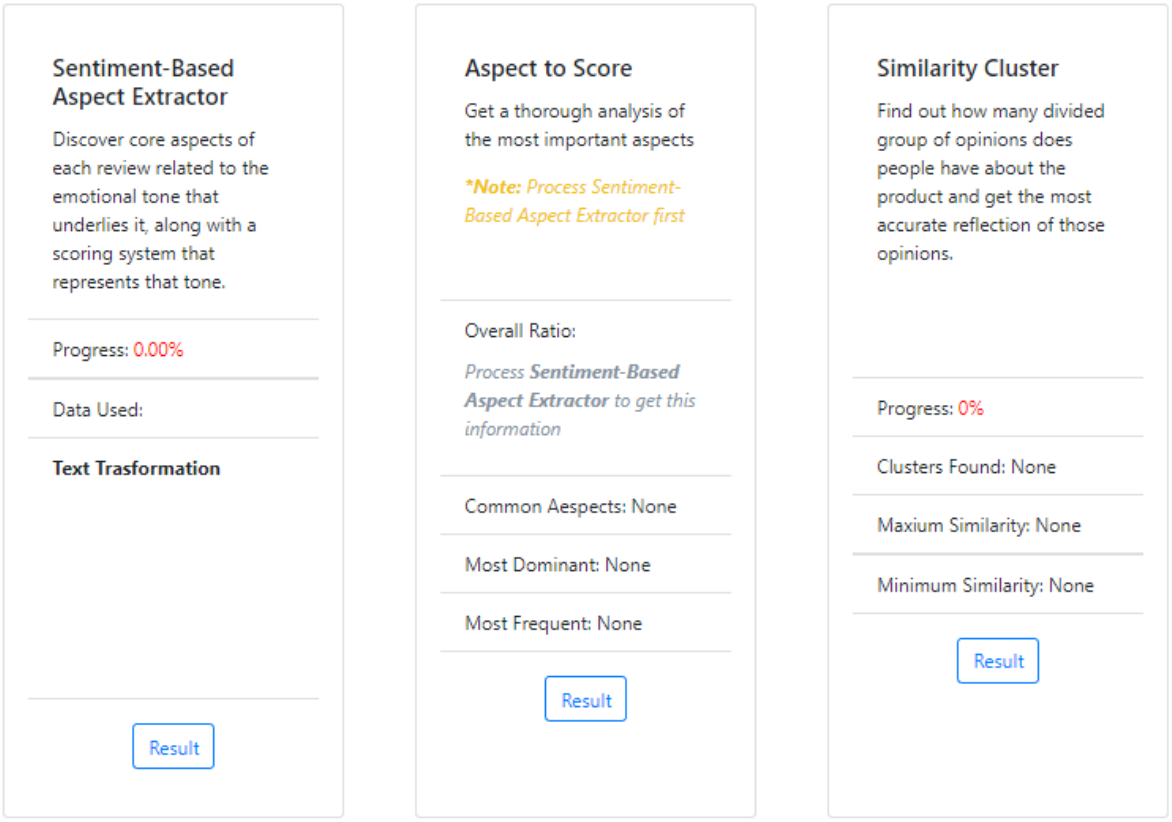


Figure 54: Unprocessed data overview info validation

Integration Test - test case (4)

Test Case Name	Integration test		
Test Class	Models Info		
Test Description	Model page with processed services		
Data Source	**Test Steps**	**Expected Result**	**Actual Result**
Developer	Goto model page and process all services and comeback	All services should display the overview information they gather after processing	Did display the correct information

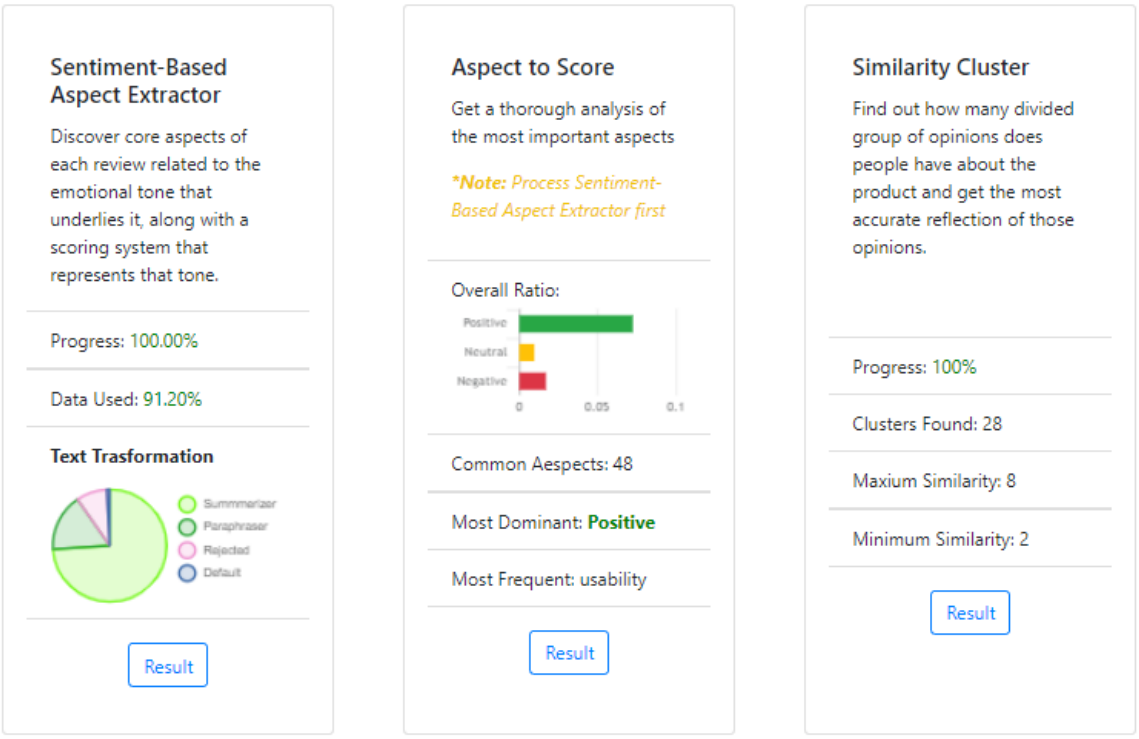


Figure 55: Processed data overview info validation

Acceptance Test - test case (1)

Test Case Name	Acceptance test		
Test Class	Change File		
Test Description	Change to another database after already selecting one		
Data Source	**Test Steps**	**Expected Result**	**Actual Result**
Developer	Select a file, get redirected to the info page then change file button and select a different file	The FIle name should change in the upper right corner	The file name correctly changed

Acceptance Test - test case (2)

Test Case Name	Acceptance test		
Test Class	Pipeline.aspect_mining_singular()		
Test Description	Influence score validation		
Data Source	**Test Steps**	**Expected Result**	**Actual Result**
Developer	Find an aspect in Aspect to score that is mostly positive but has a low score	The score should be closer to 0 if the negative influence is closer or greater than the positive	A negative score is almost equal to a positive so the influence score is 0.01

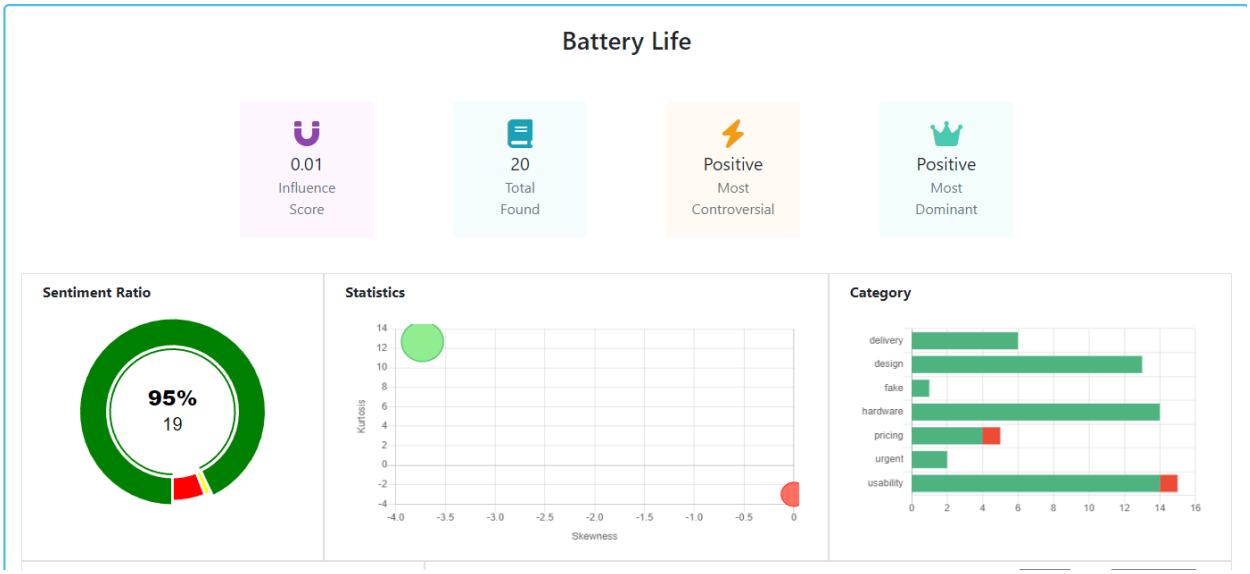


Figure 56: Influence score validation

CHAPTER 12- IMPLEMENTATION

12.1 Training

Training is the process of making the users familiar with the proposed system. This is an important step for the developer to understand if the system is too difficult for the general users to interact with. The developer must ensure that the users are easily able to interact and navigate with system. Information regarding the training session conduction for the proposed system is given below.

SL	User	Training Scope	Time Period	Comment
1	Users or business owners	Uploading database, deleting database, using all the services, Interacting with the virtual assistant	2 hr.	The procedures are being understood by the users.

12.2 Implementation Scheme

Big bang works by shutting down the existing system as well as immediately launching the new one. This scheme is much faster than the others, and it implements the new system immediately after testing is completed. It can be time consuming to transfer data from the old system, in addition to being risky even though data may be lost, crushing the new system. It is implemented on a specific website with the intention of managing only one site.

12.3 Scaling

There is no intention of scaling the system. I decided against scaling since this system is

unique and it is an academic project.

12.4 Load Balancing

Load balancing is the process of matching the system against by the impact that users have on the system. Users hit refers to the number of people are currently using the system at the time and how long the system has endured. Load balancing is the process of balancing the load with a load balancer. It distributes the load across multiple servers to keep the system running quickly.

CHAPTER 13- CRITICAL APPRAISAL AND EVALUATION

13.1 Objective that could be met

The target objectives:

- Multiple services which focus on review analysis
- High-level and low-level report
- Chat assistant

Objective-1

Success rate and others: A total of five services have been added to the system which purely focuses on generating comprehensive analytical information based on the product review. Each of the services is different and on different aspects of the product review. Users can utilize each service and successfully get precise and concise information about their product.

Objective-2

Success rate and others: Each of the services generates low-level and high-level reports. The users can successfully view low-level and high-level analytical information for their product

Objective-3

Success rate and others: A chat assistant has been added to the system as a service. The users can interact with the ai-power chat assistant and ask questions about their product.

13.2 Objectives that don’t meet/ touched

All the proposed objectives were successfully implemented. Although there was one service I could not add to the system.

The idea for this service was ad suggestions for the reviewers. The system will suggest other products that the reviewer might like based on all the analytical data. This will help the client understand more about their customers and their preferences.

The reasons why it could not be touched: To develop this service I would need a huge amount of review data and the data must include more information like customer id and customer order history. Manually parsing this kind of information will require a lot of time.

What could have been done: To build this service I need to get access to the full database of a big online marketplace. Although some old online marketplace databases can be found on the internet. But those lack in size or don't have the specific kind of information required for the service.

CHAPTER 14 - LESSON LEARNED

14.1 Pre-project Review Closing

I have completed my project proposal on Deep Insight which describes the internal processes and details about the system. This system processes product review data and generates analytical reports based on them which will aid the client in making business decisions.

14.2 What I have learned

Working on this project taught me a lot of things that will benefit me in my future profession. I've learned how to create and use complicated language models. I've also learned how to embed artificial intelligence models into a web application. I also learned how to use JavaScript libraries such as JQuery, MorrisJS, ChartJS, and HighJS to make the web application more dynamic and interactive. The most significant lesson I've learned is to operate systematically and design my architecture. The majority of the algorithms utilized in the services were written by me, which improved my efficiency and grasp of programming.

14.3 Problem I have encountered

Throughout the project development life cycle, I encountered several issues. The first challenge I faced was translating my ideas into code. Some ideas sounded simple and doable in my imagination, but putting them into action proved quite challenging. Another problem I had to overcome was selecting the appropriate language model. There are dozens of languages available on the internet, but several things have to be considered. I had to figure out how to implement the chat assistant. For the chatbot, I used Google Cloud Services. However, connecting the chatbot to my application proved problematic

due to the numerous levels of protection provided by Google Cloud. The most difficult task of all was managing my schedule.

14.4 Solutions to the problem

Every problem has a solution - is something I believe. After completing this project this belief got stronger.

When I was having trouble translating my thoughts into code, I wrote the concepts and pseudocode on note paper and built the method while looking at the notes. I tried several models and read a lot of research papers on the language model, but most of them had some type of flaw. I solved this problem by integrating various models that worked quite well. I was able to connect Google Cloud Service to my application by using a separate proxy server. I created a timebox to help me stick to my schedule.

CHAPTER 15- CONCLUSION

15.1 Project Synopsis

Deep Insight is an analytic web application that thoroughly analyzes product reviews using state-of-the-art AI models and generates high-level and low-level reports. The users can get visual analytics information about their product which will aid them in improving their product, making business decisions, and understanding the demographics of their product. Analyzing reviews is a crucial part of the business but this is a tedious and time-consuming part of the business. Online marketplace does offer some analytic reports but these reports are centered around ratings which sometimes can be deceiving and don't capture the full picture. Deep Insight offers multiple services which use AI models to analyze review text and generate an in-depth analytical report which gives a deep insight into how the customers feel about the product.

Deep Insight is developed using Python, JavaScript, JQuery, HTML, CSS, and SQL. Different diagrams and all kinds of testing are provided in the documentation.

15.2 Project Goal

The main goal of this project is to fully automate the review analysis process of a product/service-based business. Here are the goals and objectives the proposed system ought through its services:

- Automate the review analysis process
- Identify the product’s strengths and weaknesses
- Aid in making business decisions and plans
- Gain deeper insight into customer-base psychology

15.3 Project Success

The project development was successful as it fulfilled all the requirements. The users can upload their product review database and discover core aspects of each review related

to the emotional tone that underlies it along with a scoring system that represents that tone, Get a thorough analysis of the most important aspects, find out how many divided groups of opinions do people have about the product and get the most accurate reflection of those opinions, extract topics discussed in the reviews and even get help from a chatbot. This project has completed all the initial criteria it promised and even more. So this project was successful.

15.4 What I did in the documentation

I have compiled everything I have done from start to finish for this project in the documentation. This includes various diagrams, time-boxing, and testing. I have also discussed the methodology, feasibility studies, and goals the project strives to achieve in greater detail. This document describes the process, functionality, and target goals in detail, providing a thorough understanding of the project.

15.5 Project Value

Through various smart and innovative means, technology provides innovative ways of doing work. Deep Insight revolutionizes the traditional way of doing review analysis by fully automating the process using complex AI technology. This system provides a unique value for the business owner in the online market. This eliminates all of the resources required for review analysis, such as budget, time, and manpower, and allows the business to move faster, which is critical for keeping up with the competition. The challenges I encountered while developing this project helped me in gaining a better understanding of the technology used, which will benefit me in my professional career.

15.6 My Experience

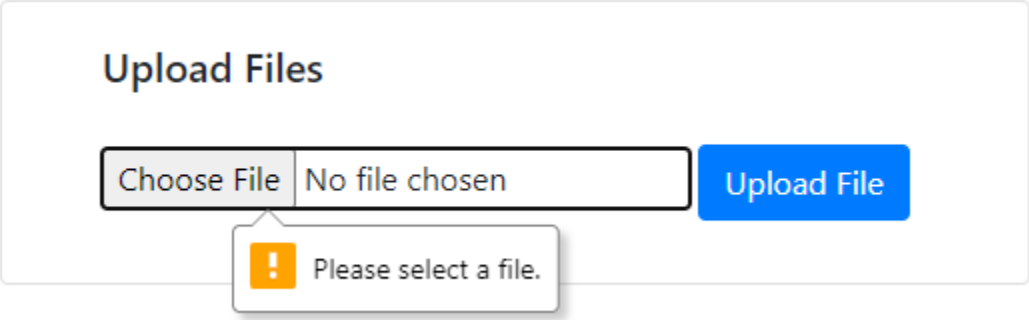
During the project development, I gained a lot of experience. I have faced multiple challenges but I have also learned how to overcome them. I had to learn about different

types of technologies to develop this project. I also now understand how to manage a large project while meeting all of the requirements and finishing it on time.

APPENDIX A

Unit Test - test case (1)

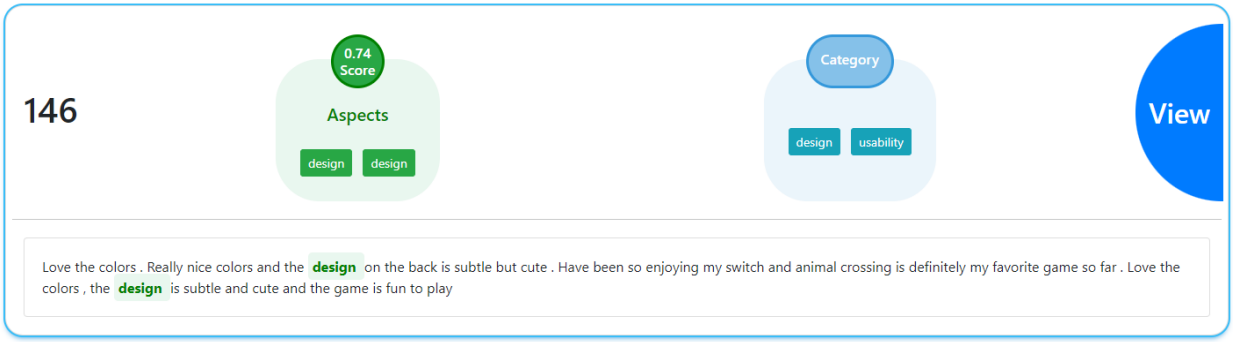
Test Case Name	Unit test		
Test Class	FileUploadValidation		
Test Description	File Upload Validation		
Data Source	**Test Steps**	**Expected Result**	**Actual Result**
Developer	Click Submit without uploading a file	A warning requesting to upload a file first	Warning correctly displaying



Unit Test - test case (2)

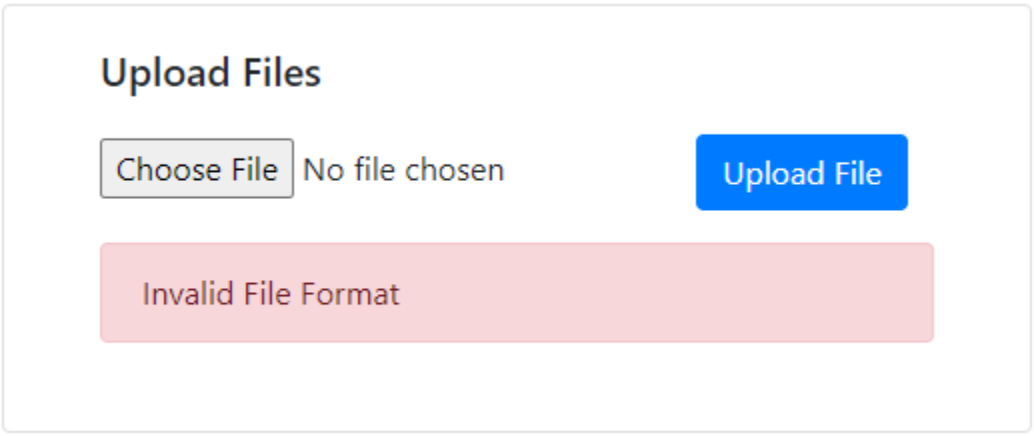
Test Case Name	Unit test		
Test method	search_filter()		
Test Description	Quick Search option proper navigation		
Data Source	**Test Steps**	**Expected Result**	**Actual Result**

Developer	Set Sentiment to positive and Filter to maximum when the current file is Nintendo Switch	The screen should be scrolled to the aspect card with the highest positive score value (aspect no. 146).	The screen scrolled to aspect - 146 which does have the highest positive value in the current dataset (Nintendo Switch)



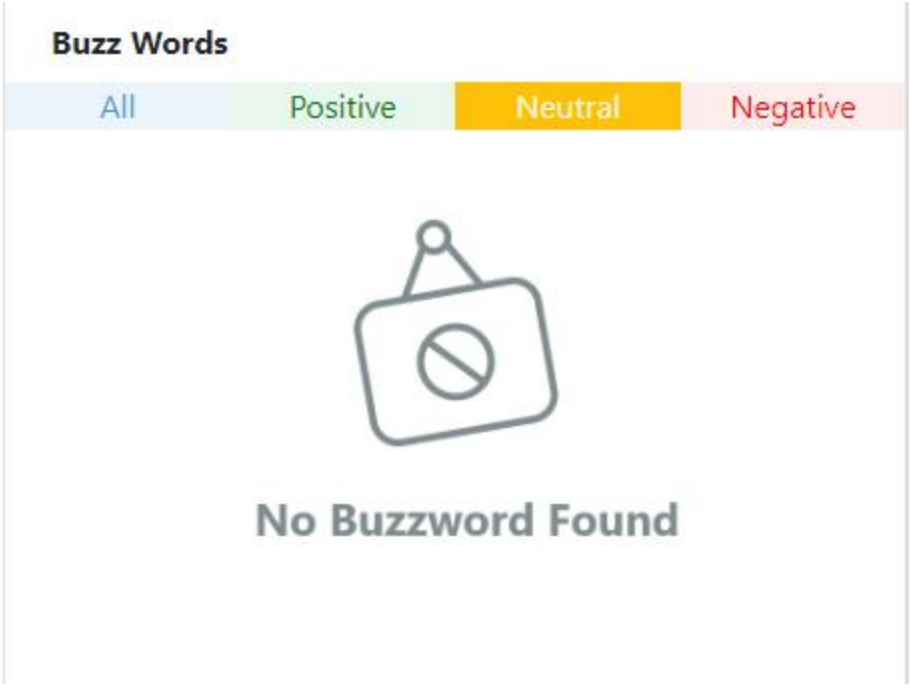
Unit Test - test case (3)

Test Case Name	Unit test		
Test Class	FileUploadValidation		
Test Description	File Format Validation		
Data Source	**Test Steps**	**Expected Result**	**Actual Result**
Developer	Try to upload a picture	A warning showing that the file could not be uploaded	Warning correctly displaying



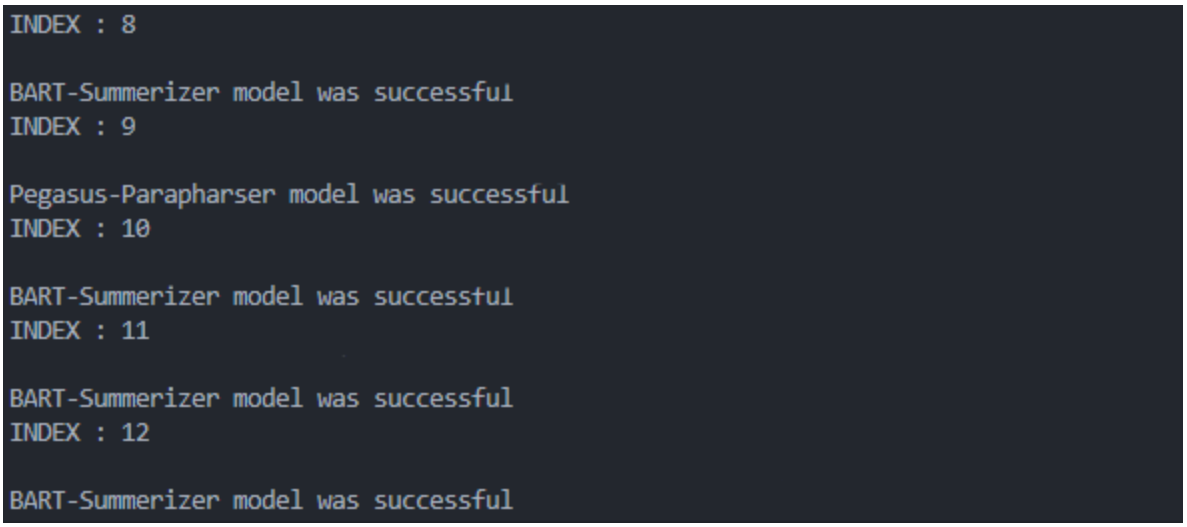
Unit Test - test case (4)

Test Case Name	Unit test		
Test method	cloudEventController		
Test Description	Buzzword options validation		
Data Source	**Test Steps**	**Expected Result**	**Actual Result**
Developer	Click a label that does not have any buzzwords	An image with the message "No Buzzwords Available" should appear	Clicked Neutral option for "Battery Life". No buzzword available image appeared.



Unit Test - test case (5)

Test Case Name	Unit test		
Test method	Pipeline.aespect_mining_singlular()		
Test Description	Text Transformation Validation		
Data Source	**Test Steps**	**Expected Result**	**Actual Result**
Developer	Click on the result in Sentiment-Based Aspect Extractor and check the log	The log will show a different model if the default "BART-Summarizer" failed	"Pegasus-Parapharser" was used for review no. 9



Unit Test - test case (6)

Test Case Name	Unit test		
Test method	Pipeline.aespect_mining_singlular()		
Test Description	Sentiment score Validation		
Target Range	The first 10 and last 10 reviews		
Data Source	**Test Steps**	**Expected Result**	**Actual Result**
Developer	Click on result in Sentiment-Based Aspect Extractor and check the log	The “Average” value for each label should be the mean of “Original” and “NLP(Combined)”	The mean value was correct

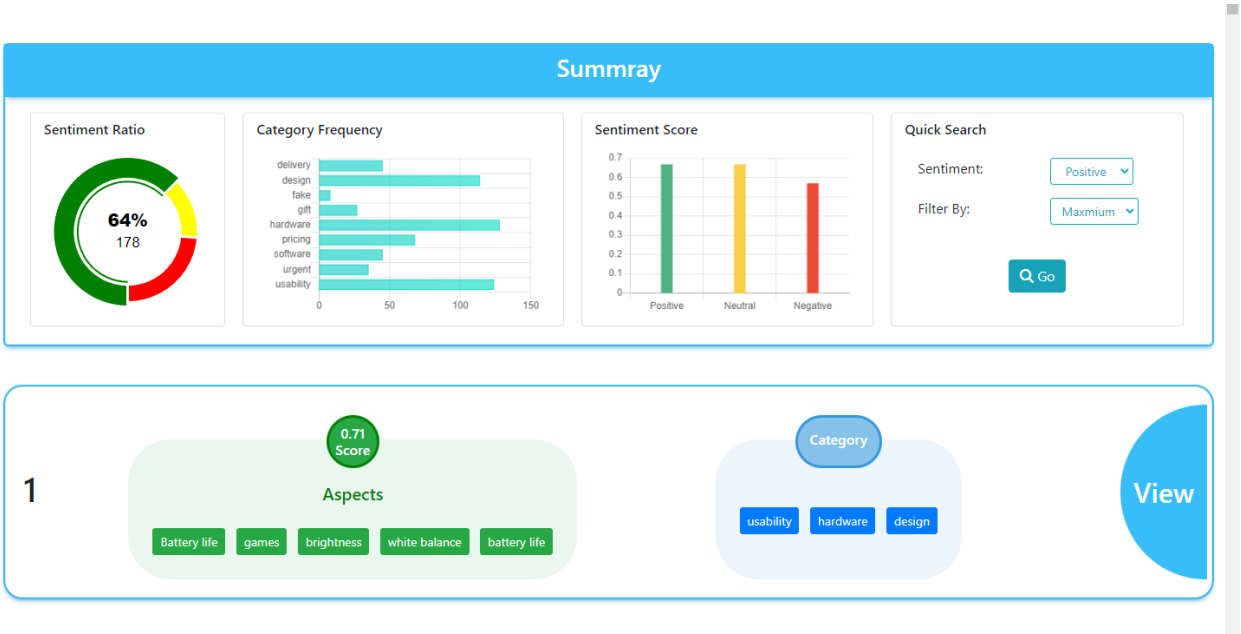
```
INDEX : 51
Rating correction (NEGATIVE): Original-1.00 | NLP(Combo)-0.25 | Average-0.62
Pegasus-Parapharser model was successful
INDEX : 52
Rating correction (NEUTRAL): Original-1.00 | NLP(Combo)-0.49 | Average-0.74
Rating correction (NEGATIVE): Original-1.00 | NLP(Combo)-0.27 | Average-0.63
BART-Summerizer model was successful
INDEX : 53
Rating correction (POSITIVE): Original-1.00 | NLP(Combo)-0.34 | Average-0.67
Rating correction (NEGATIVE): Original-1.00 | NLP(Combo)-0.26 | Average-0.63
BART-Summerizer model was successful
INDEX : 54
Rating correction (POSITIVE): Original-1.00 | NLP(Combo)-0.34 | Average-0.67
BART-Summerizer model was successful
```

Module Test - test case (1)

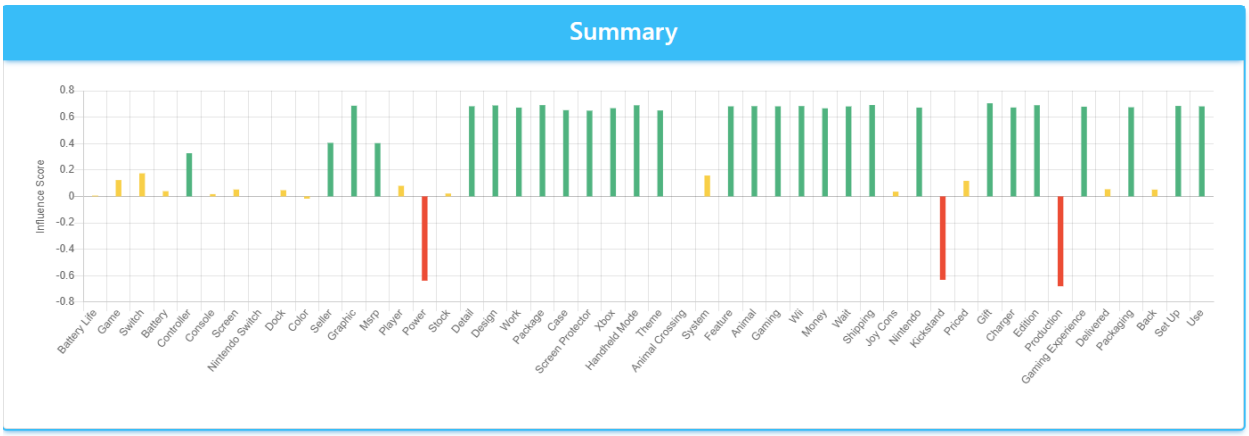
Test Case Name	Module test

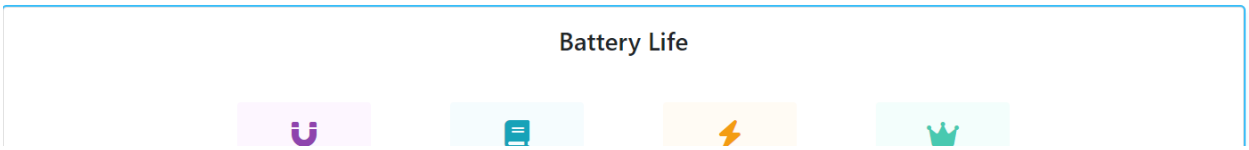
Test method	RoutePipeline.route_models()		
Test Description	All services properly display information after processing		
Data Source	**Test Steps**	**Expected Result**	**Actual Result**
Developer	Click the Result button for each service	Should be redirected to a page containing processed information	Was successfully redirected

Sentiment-Based Aspect Extractor

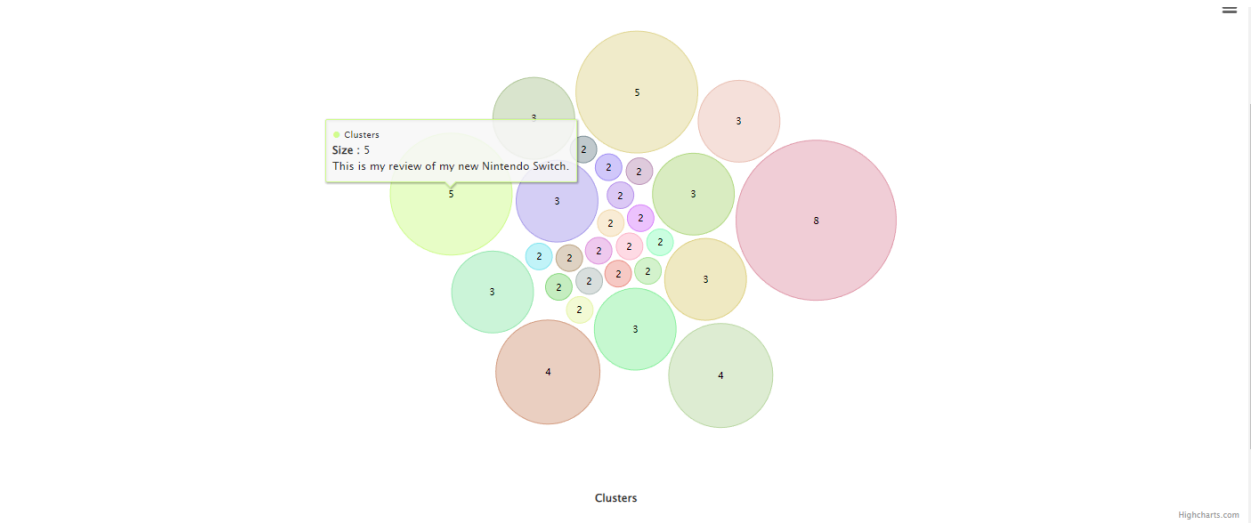


Aspect to Score





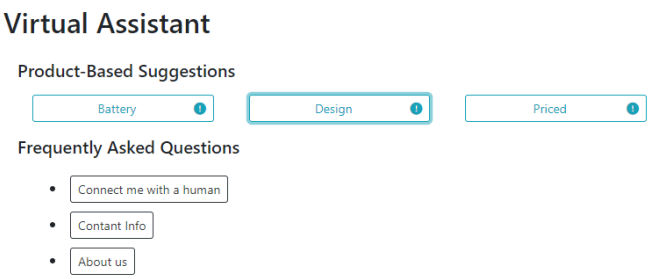
Similarity Cluster

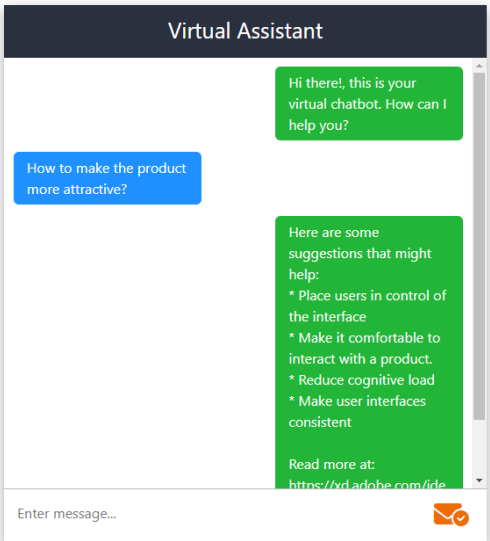


Review	Similarity Found
• A review of the Nintendo Switch console.	3
• I bought a joy-con for the Nintendo Switch	2

Topic Extraction

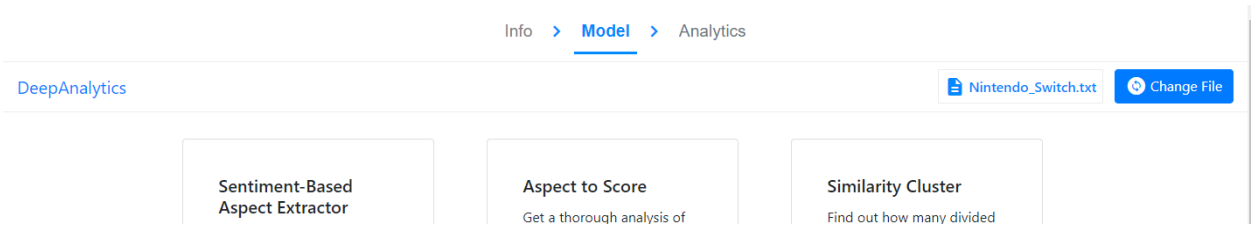
Assistant





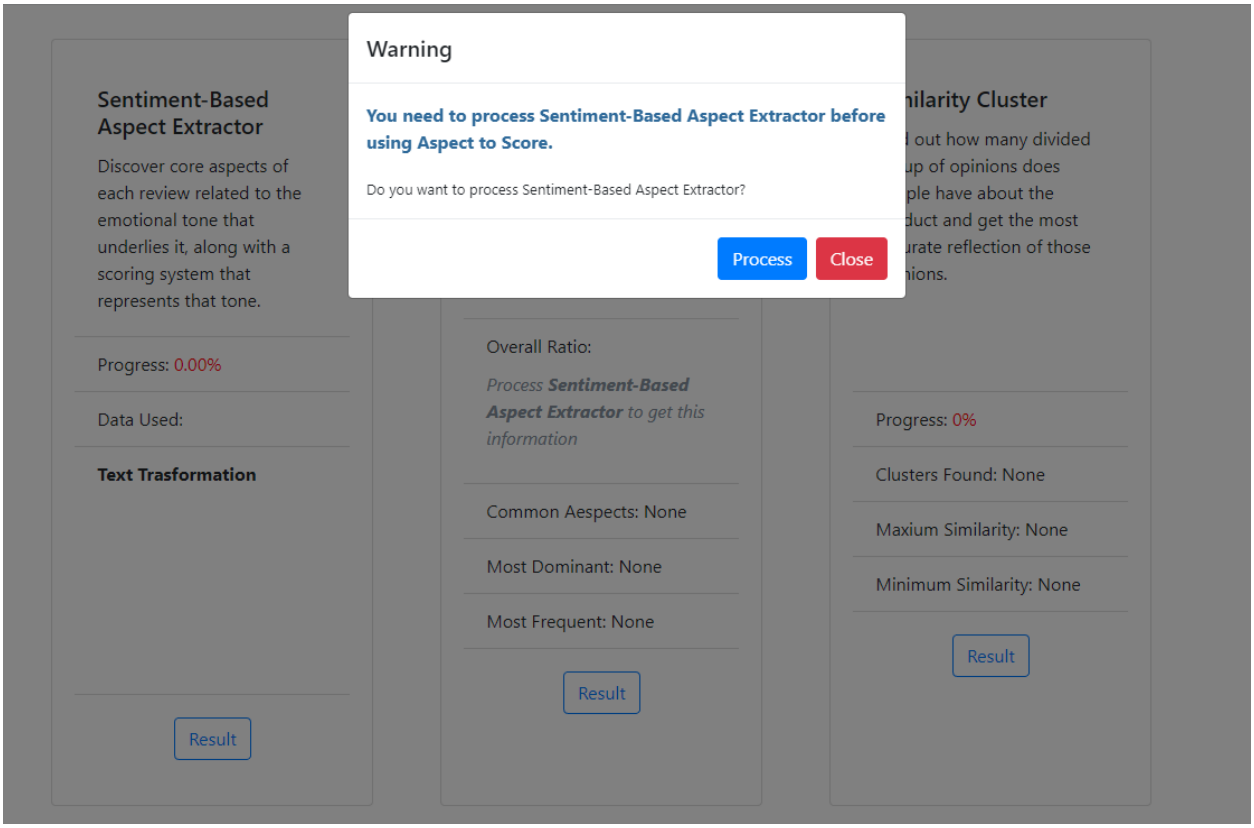
Integration Test - test case (1)

Test Case Name	Integration test		
Test Class	Current File		
Test Description	Correct Database selected throughout the session		
Data Source	**Test Steps**	**Expected Result**	**Actual Result**
Developer	Click the “Apple_Airpods” database on the homepage and navigate to other pages	The same name as the selected file should show on every other page	Did display the name “Apple_Airpods” on Info and Models page



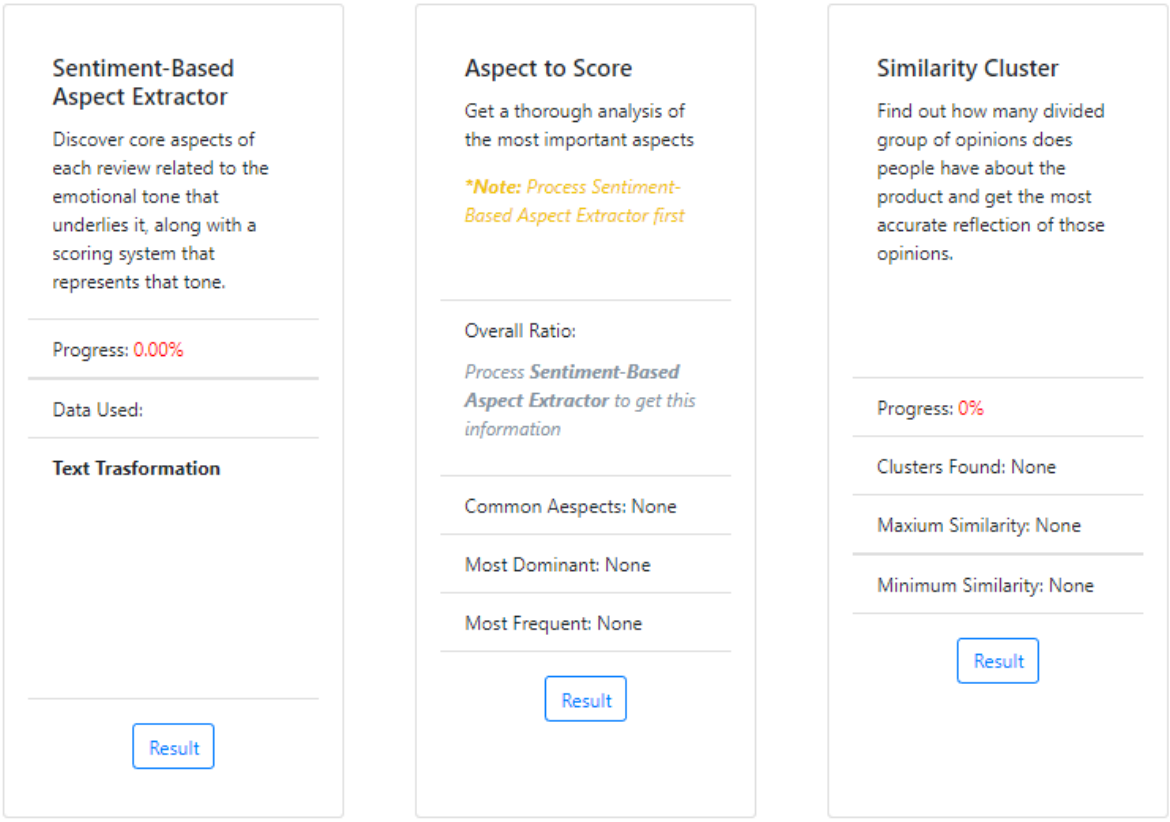
Integration Test - test case (2)

Test Case Name	Integration test		
Test Class	RoutePipeline.route_models()		
Test Description	Services dependencies validation		
Data Source	**Test Steps**	**Expected Result**	**Actual Result**
Developer	Click on Aspect to Score result before clicking any of the other services	A pop-up should show up suggesting to process the Sentiment-Based Aspect Extractor first	Did correctly display a warning pop up



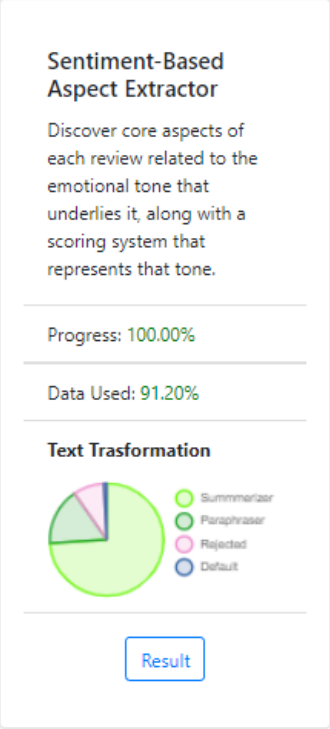
Integration Test - test case (3)

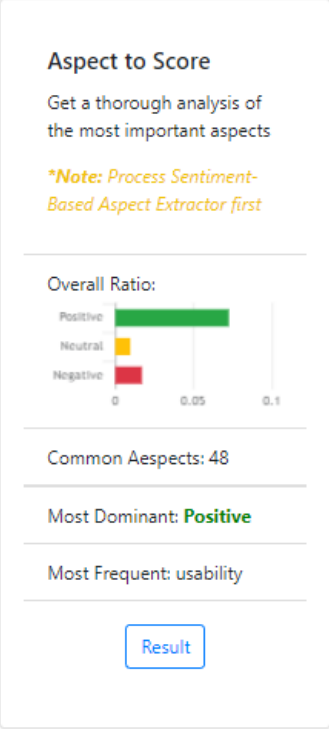
Test Case Name	Integration test		
Test Class	Models Info		
Test Description	Model page without processing any services		
Data Source	**Test Steps**	**Expected Result**	**Actual Result**
Developer	Goto model page and don't process any service	There should a piece of default information in the overview section for each model	Did display the default information

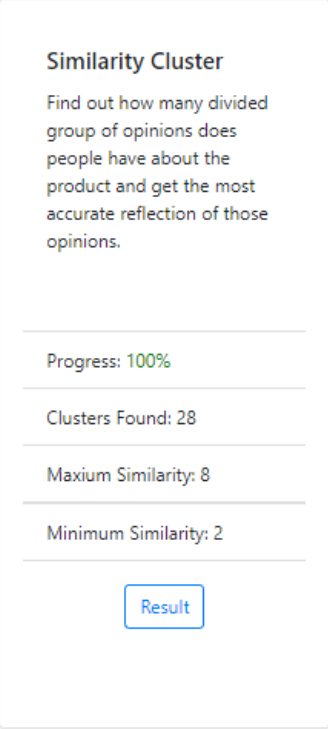


Integration Test - test case (4)

Test Case Name	Integration test		
Test Class	Models Info		
Test Description	Model page with processed services		
Data Source	**Test Steps**	**Expected Result**	**Actual Result**
Developer	Goto model page and process all services and comeback	All services should display the overview information they gather after processing	Did display the correct information





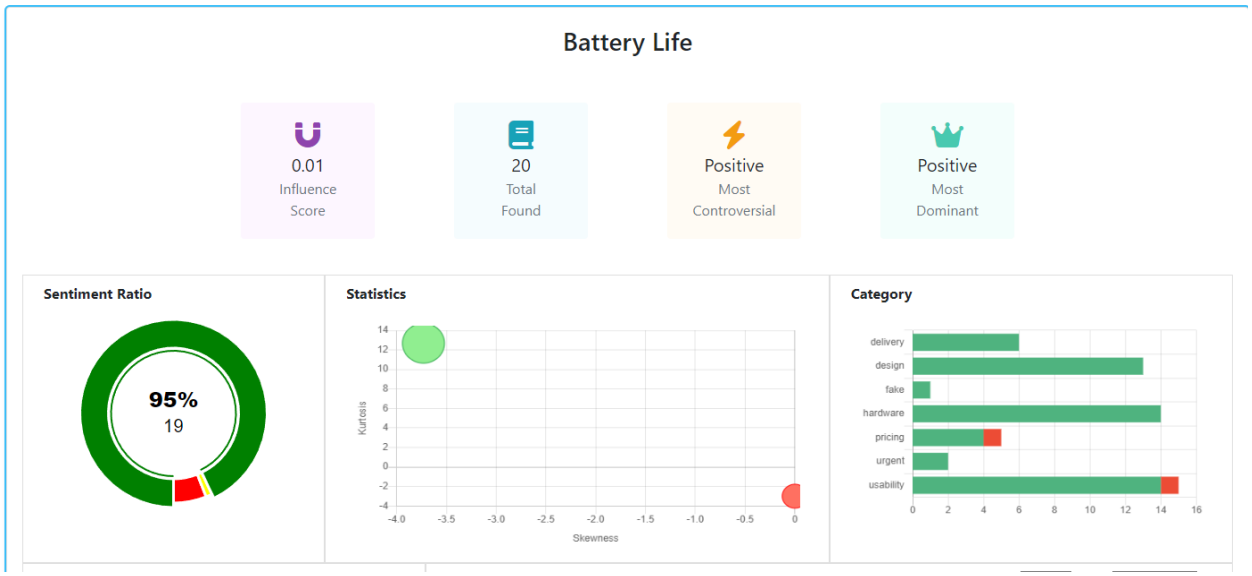


Acceptance Test - test case (1)

Test Case Name	Acceptance test		
Test Class	Change File		
Test Description	Change to another database after already selecting one		
Data Source	**Test Steps**	**Expected Result**	**Actual Result**
Developer	Select a file, get redirected to the info page then change file button and select a different file	The FIle name should change in the upper right corner	The file name correctly changed

Acceptance Test - test case (2)

Test Case Name	Acceptance test		
Test Class	Pipeline.aspect_mining_singular()		
Test Description	Influence score validation		
Data Source	**Test Steps**	**Expected Result**	**Actual Result**
Developer	Find an aspect in Aspect to score that is mostly positive but has a low score	The score should be closer to 0 if the negative influence is closer or greater than the positive	A negative score is almost equal to a positive so the influence score is 0.01



REFERENCES

[1] Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*.

[2] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). *Attention Is All You Need*.

[3] *What is Reliability Testing: Definition, Method and Tools*. (n.d.). Retrieved January 12, 2023, from https://www.softwaretestinghelp.com/reliability-testing/

[4] *What is Software Testing and How Does it Work?* | *IBM*. (n.d.). Retrieved January 12, 2023, from https://www.ibm.com/topics/software-testing

[5] Yang, H., & Li, K. (2022). *PyABSA: Open Framework for Aspect-based Sentiment Analysis*.

191-16-420

ORIGINALITY REPORT

19%	19%	2%	%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	dspace.daffodilvarsity.edu.bd:8080 Internet Source	16%
2	eprints.umm.ac.id Internet Source	1%
3	faq-facts.com Internet Source	<1%
4	www.coursehero.com Internet Source	<1%
5	www.softwaretestinghelp.com Internet Source	<1%
6	diposit.ub.edu Internet Source	<1%
7	www.halifax.ca Internet Source	<1%
8	searchsoftwarequality.techtarget.com Internet Source	<1%
9	logimetrix.co.in Internet Source	<1%

10	thrivemyway.com Internet Source	<1 %
11	melbourneinstitute.com Internet Source	<1 %
12	Jingneng Fu, Hui Zhang, Wen Luo, Xiaodong Gao. "Dynamic Programming Ring for Point Target Detection", Applied Sciences, 2022 Publication	<1 %
13	adz.atiromarket.shop Internet Source	<1 %
14	www.dau.edu Internet Source	<1 %
15	Shuntaro Takahashi, Kumiko Tanaka-Ishii. "Cross Entropy of Neural Language Models at Infinity—A New Bound of the Entropy Rate", Entropy, 2018 Publication	<1 %
16	ebin.pub Internet Source	<1 %
17	www.springerprofessional.de Internet Source	<1 %

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	On		