

***CitiSense: A Smart Web App for Automated Bilingual
Feedback Processing and Data Visualization***

Project Documentation Submitted to the Faculty of the
School of Computing and Information Technologies
In Partial Fulfillment of the Requirements for
Introduction to Systems and Design for IT
MNTSDEV

By

Lorenzo Emil S. Bernal
Mark Lyster C. Marabi

Gabrielle J. Cabangcala
Luis Lorenzo D. Lazaro

Table of Contents

List of Tables	v
Introduction	1
Project Context.....	1
Statement of the Problem	2
Objectives	2
Significance of the Project	3
Scope and Limitations	4
Review of Related Literature / Systems	5
Current System	9
Technical Background	9
List of Processes	9
SWOT Analysis	11
Proposed Solution.....	13
Technical Background	13
Feasibility	13
Operational Feasibility.....	13
Economic Feasibility	13
Technical Feasibility.....	14
Schedule Feasibility	14
Requirements Analysis.....	14
Project Vision	14
Prototype (Mock Flow / Wireframe).....	14
Project Lean Canvas.....	18
User Classes and Characteristics	19
Product Backlog.....	20
Product Roadmap	21
Release Plan	22
Use Case Full Description:.....	23
Fully Dressed Use Case - QuadThink	23
Conclusion	24
References	25
Appendices	27
Appendix A: Roles and Responsibilities.....	27

Appendix B: Minutes of the Meetings 28

Appendix C: Methodology..... 29

Appendix D: Project SharePoint Link..... 31

Appendix E: Use Case Full Description 31

Appendix F: Requirements Traceability Matrix..... 39

Appendix G: RACI Matrix..... 40

List of Figures

Figure 1. Process Flow of Current Sentiment Analysis of IRAD department.....	10
Figure 2. SWOT Analysis of CitiSense web application.....	11
Figure 3. Register UI of CitiSense web application.....	15
Figure 4. Login UI of CitiSense web application	15
Figure 5. Preview Data tab UI	16
Figure 6. Dashboard of the CitiSense web application	17
Figure 7. View archived visualization results of the CitiSense web application	18
Figure 8. Product Roadmap for the development of the CitiSense web application.	21
Figure 9. Use Case Diagram of the CitiSense web application.....	22
Figure 10. Delegation of Tasks for Finals Paper.....	27
Figure 11. Design Thinking Quadrant.....	29
Figure 12. User stories for product backlog	30
Figure 13. Continuation of user stories for product backlog.....	31
Figure 14. RACI Matrix of the CitiSense of the project development	40

List of Tables

Table 1. List of processes in the current system.....	9
Table 2. User Classes and Characteristics involved in the CitiSense project	20
Table 3. Product Backlog for the development of the CitiSense web application.....	21
Table 4. Product Requirements for the CitiSense web application	32
Table 5. Use case: Register	32
Table 6. Use Case: Login.....	33
Table 7. Use Case: References	34
Table 8. Use Case: View Dashboard	36
Table 9. Use Case: View past visualization.....	36
Table 10. Use Case: Manage hosted web app.....	37
Table 11. Use Case: Manage users.....	38
Table 12. Test Case Table for the CitiSense web application.....	39
Table 13. Requirements Traceability Matrix for the CitiSense web application	40

Introduction

The IRAD (*Information Resource and Analysis Division*) department of DOST-STII (*Science and Technology Information Institute*) is currently handling a time-consuming task of manually analyzing survey responses with thousands of feedback. The responses they receive come in Tagalog, English, and a mix of both, making the process complex. Due to a large amount of data coming in, the department wants a system that can handle and analyze data to quickly extract meaningful insights.

Natural language processing (NLP) is a machine learning technology that gives computers the ability to interpret, manipulate, and comprehend human language. Organizations today have large volumes of voice and text data from various communication channels like emails, forms, social media, and text messages. They employ Natural Language Processing (NLP) software to interpret this data automatically, analyze the message's intent or sentiment, and reply to human conversation in real time.

This project aims to:

- Implement a system that leverages an NLP (natural language processing) model suited for sentiment analysis to categorize incoming textual data in both English and Tagalog.
- Convert the results into visualized sentiment trends.

The system will not only reduce manual workload but also streamline the current sentiment analysis process of the IRAD department, resulting in faster, more accurate decision-making and providing actionable insights to enhance future strategies quickly.

Project Context

The project, titled *CitiSense: A SMART web application for Automated Bilingual Feedback Processing and Data Visualization*, is being developed by the QuadThink team in partnership with the Department of Science and Technology – Science and Technology Information Institute (DOST-STII), particularly its Information Resources and Analysis Division (IRAD). As a division responsible for analyzing and managing data that supports public science communication, IRAD plays a critical role in collecting feedback from stakeholders and the public. However, their current feedback processing system is limited by manual analysis, language constraints, and the lack of real-time visualization tools. The manual workflow makes it difficult to scale, slows the delivery of insight, and restricts the organization's ability to make timely, data-driven decisions.

To address these limitations, CitiSense is being developed as a bilingual web application that uses a fine-tuned Natural Language Processing (NLP) model to classify user feedback into positive, neutral, or negative sentiment in both English and Tagalog. It further enhances usability by transforming processed data into interactive visual dashboards, offering real-time insights for stakeholders and decision-makers. This solution is aligned with DOST-STII's mission to strengthen science and technology information systems and to serve as the country's leading

source of accurate, accessible science and technology content. The project supports their long-term strategic goal of improving public engagement and institutional responsiveness through modern, automated tools. By integrating seamlessly with DOST's service feedback forms, *CitiSense* ensures a more inclusive and efficient analysis process, ultimately helping the government agency act faster and more effectively on the voices of the public they serve.

Statement of the Problem

The IRAD department of the DOST-STII plays an important role in monitoring and analyzing public sentiments towards their four services, and part of their responsibility involves analyzing thousands of responses they receive quarterly. Even so, the present approach to interpreting feedback isn't adapted to the rising volume and complexity of data, causing delays in using the information to guide decisions.

To better understand the limitations of the current process, the key challenges can be outlined as follows:

- The public responses process by the IRAD department can reach up to 10,000 per quarter. Currently, both sentiment analysis and thematic sorting are performed manually, requiring lots of time and effort.
- Often, messages contain both English and Tagalog words, which adds complication to accurate sentiment analysis. The fact that meetings are only held every three months makes it more difficult for the department to gather fast insights from data.
- A manual system limits the organization's ability to handle a higher volume of analysis, react promptly to trends, and quickly provide reports for decisions.

Objectives

Main

Objective:

To develop *CitiSense*, a web application that leverages Natural Language Processing (NLP) to automate the sentiment analysis and visualization of textual feedback provided in English, Tagalog, or a combination of both, thereby improving the efficiency, accuracy, and responsiveness of the DOST-STII IRAD's feedback processing system.

Specific Objectives:

1. ~~To design and implement an NLP-driven feedback sentiment analysis system capable of accurately classifying survey responses into positive, neutral, or negative sentiments in both English and Tagalog within a six-month development timeline.~~

To design and implement an NLP-driven feedback sentiment analysis system capable of accurately classifying survey responses into positive, neutral, or negative sentiments in both English and Tagalog, with a model accuracy of at least 85% within a six-month development timeline.

2. ~~To reduce manual workload by at least 70% through the integration of automated feedback processing and data visualization, thereby accelerating the turnaround time for feedback reports and ensuring timely decision-making within DOST-STII.~~

To reduce manual workload by at least 80% through the integration of automated feedback processing and data visualization, thereby accelerating the turnaround time for feedback reports and ensuring timely decision-making within DOST-STII.

3. **To ensure seamless integration with DOST-STII's existing feedback collection workflows**, supporting scalability and future adaptation for additional languages or data sources.

Significance of the Project

This project aims to present valuable information regarding the opinions of event attendee by analyzing their feedback that is written in both English and Tagalog. The insights provided in this paper will benefit various organizations to better understand different comments and deliver necessary solutions to improve DOST-STII's services.

IRAD Department of DOST-STII. The project would benefit this department by having the system available to them as a tool to effectively analyze feedback. With the help of the system, the department can easily identify issues and enhance its ability to make data-driven decisions.

Future Project Developers. This project would allow future project developers to adopt the system and make more innovative solutions by integrating more enhanced features. The project will serve as the foundation for new systems that can be customized for different projects.

Data Analysts. The system would provide data analysts with a structured sentiment dataset that can be utilized to identify trends and measure public sentiment. Categorized feedback would be useful for data analysts to perform deeper analyses and evaluations and provide actionable insights to improve a certain service.

Other Government Agencies. Government agencies that also collect feedback from their services would also benefit from this project. These agencies could also adopt a solution to improve their service quality.

Educational Institutions. The analyzed sentiments help these institutions achieve better service outcomes and strengthen user interactions. The project can also serve as a model for these institutions to inspire innovation among AI systems.

Scope and Limitations

Scope

This project focuses on creating *CitiSense*, a web app designed to automatically analyze sentiment and visualize data from survey responses in English, Tagalog, or a mix of both languages. The system will utilize a fine-tuned Natural Language Processing (NLP) model to classify feedback into three categories: positive, neutral, and negative. Additionally, there will be an interactive dashboard that showcases real-time sentiment trends and keyword insights, helping DOST-STII's IRAD division gain a better understanding of public opinion. The app will integrate with DOST-STII's existing feedback collection form and will accept text-based feedback. The primary users of this system include data analysts, researchers, and government personnel monitoring public responses.

Limitations

The system will only address text responses and won't support audio or image feedback. While the NLP model is designed to handle both Tagalog and English inputs, using regional dialects or very informal language might impact the accuracy of sentiment classification. The model's effectiveness is dependent on the quality and variety of the training data. Furthermore, the visualization component will be limited to sentiment trends and keywords; more advanced predictive analytics or topic modeling will not be included in this initial release. Moreover, the system will not integrate real-time data from external sources outside of the existing feedback form.

Review of Related Literature / Systems

This project's Review of Related Literature was divided into multiple sections to assess other similar research and studies regarding this project. The researchers decided to categorize the basis and topics of the study into various categories that have their focus and will be tackled separately and in depth by the researchers.

Role of Sentiment Analysis

The number of comments online has become rampant, together with the fast and increasing growth of various internet applications [1]. Businesses, organizations, and researchers assess these comments and study them in order to better understand people's sentiments and opinions on different topics online. However, doing and reviewing these comments manually is time-consuming and inefficient [2]. Sentiment analysis through automated is an efficient approach to understanding and detecting different emotions in different languages and in text in order to quickly and accurately evaluate them. Sentiment Analysis is widely used in different areas such as social media, government, private institutions, and healthcare in order to make accurate analyses and support wiser decision-making.

According to [3], there are numerous amounts of research and studies that have explored different sides or aspects of sentiment analysis; however, the wide array of studies makes it hard for scholars to detect the most relevant findings of this tool. Several studies are focused on deep learning or other common tools, such as VADER, and others are focused on their SA applications in areas such as education or health [4]. On the other hand, research papers also cover multiple analysis levels, such as document-level and aspect-level sentiment detection. Regardless of these concepts, challenges are still showing, including language diversity and improving the accuracy level of all sentiment models.

Artificial Intelligence (AI) is a powerful tool that can help sentiment analysis improve and step up feedback detection. It enables machines to better understand human emotions and multiple languages for enhanced accuracy [5]. The use of Sentiment analysis through automated tools has become an essential part of daily operations, driven by the continuous advancement of AI. According to [6], future research directions imply that it can be expanded to enhanced functionalities such as combining text with images or videos, expanding language support, and developing prompt-based techniques.

Multilingual Challenges

According to [7], the meaning of multilingualism, code-mixing, or code-switching refers to people who can speak multiple languages fluently. This is also practiced in writing essays, papers, or commenting online. This scenario is widespread globally and on social media, where people integrate multilingualism into everyday life. This scenario is common in India. In India, languages such as Hindi and English are often observed and combined in one sentence. Since multilingualism has become commonly used, research in Natural Language Processing (NLP) is focused on how to manage this complexity and change in areas such as crisis response, healthcare, government services, and business industries [8].

[9] states in their research that there are many obvious challenges in processing and determining code-mixed languages. One instance of this topic is the lack of language tools for tasks such as tagging parts of speech or identifying different entities, which makes it hard to build such a reliable and accurate system. Another identified problem that [10] confirmed is filtering and recognizing code-mixed data because most online platforms don't label everything clearly. Trained people have the possibility to interpret mixed languages that depend on their language skills or understanding differently, and this causes human bias [8]. Inconsistencies in interpreting mixed languages could create problems and can cause errors in data labeling and AI models training.

Inadequate evaluation tools are also an issue regarding code-mixed language, as they judge the level of code-mixed language accuracy and correctness [11]. It could be hard to test if annotators understand mixed languages well. In addition, [12] shares that there may be instances where there are also not enough public sample datasets to utilize for researchers to test. Due to this factor, it is difficult to compare and solve code-switching or code-mixing problems. Researchers will need better tools to be able to include more languages, bigger datasets, and fairer ways to combat information bias and to evaluate how well systems work for multilingual scenarios.

User-Based Platform

A study by [13] focuses on a web platform application designed to operate and manage graduation processes in an academic institution. Students, administrators, and professors can login to the system, which is a key feature of the web platform. Each role has unique specifications, such as for students they have access to view and apply for projects, while professors can manage assignments and assign them to students. On the other hand, administrators have access to the entire system and can control it. The login-based feature was to ensure that the system adheres to the Data Privacy Act, efficient task management, and a seamless experience tailored to specific roles.

According to [13], the platform is designed and developed using these web development technologies: PHP, JavaScript, MySQL, HTML, and CSS. Data transactions and connectivity with the server are what the backend is responsible for. On the other hand, having a user-friendly interface is what the frontend aims to achieve. Generating reports is one of the strengths of this web platform as it can produce PDF documents, which are all from the central database. Users would be able to work effectively and increase productivity when using the developed platform. In addition, it also reduces human error and paperwork for office tasks, which is an advantageous technique for academic institutions.

Even though this system is focused on managing graduation processes, it can relate to the project of the developers when developing their own system and integrating sentiment analysis alongside it. It demonstrates how data visualization, login process, and role-based accounts can be applied to any system that can be used by developers. Platforms that analyze user feedback are suitable for this type of system since it deals with understanding comments and emotions, like login systems for different users, such as developers, data analysts, or general users, and can generate related data visualizations.

NLP-Driven Evaluation

In today's era where technology is rampant, businesses are keeping up with trends to be on the same boat with the emerging technologies [14]. Customer feedback is essential for businesses to improve and prosper in the future. Analyzing their sentiment is a valuable tool for their business to prosper. The tools for creating an automated system could help companies in creating an online platform and addressing customer sentiments about their products or services [15]. Customers leave their feedback after trying a new product or experiencing the provided service by the business. The feedback from customers could be positive, negative, or neutral. By analyzing reviews using a proper tool, companies and businesses can improve the quality of their products or services. Manually processing the comments takes a huge amount of time. Employees would not be productive if analyzing comments were not automated. The process is recommended to be done using Artificial Intelligence (AI) in order to process the data automatically and help make wiser decisions, benefiting the company.

Addressing this issue could be solved if companies use Natural Language Processing (NLP) and Machine Learning (ML). With the use of NLP, the system would be able to understand and analyze human language. On the other hand, [16] states that the ML's responsibility is to help the system learn from having large amounts of data to review. The main programming language in developing this system is Python. Innovating technologies allow businesses and organizations to easily process customer sentiment and respond to any of their concerns. Compared can analyze large-scale feedback without manually processing it with the help of these tools.

There are many different languages spoken all around the world, and according to [17], there are 6,500 estimated languages, which makes it very hard for computer systems to learn and understand all of them. Language contains different uses; it contains identity, emotion, culture, and diversity. Due to this problem, NLP comes in to address them. NLP is essential in learning all kinds of languages. This tool can work in a flexible manner in many areas, such as customer service, law, education, and more. With the help of this tool, work-related interactions can be easier and more productive.

Understanding Transformers and BERT

According to research by [18], Transformers is one of the powerful libraries used in NLP that can provide pre-trained models that are integrated into systems that can understand and process multilingual languages. Such models based on transformer architecture can investigate all the words in one sentence and only focus on important parts (self-attention), unlike the older models such as Recurrent Neural Networks that analyze the text step-by-step. In that way, the Transformer-based architecture possesses a more accurate and faster analysis, especially in longer sentences. Due to this, Transformers serve as a fundamental tool for serving different models for specific tasks such as translation, text summarization, and information retrieval.

On the other hand, one of the most popular models based on Transformer architecture is called BERT. The Bidirectional Encoder Representations was created by Google and is recognized for its ability to read sentences in both left-to-right and right-to-left directions. In this way, the model can better understand the meaning of words based on context. Moreover, this model was trained using a large collection of text that has been proven to perform well in huge tasks such as Masked Language Modeling (MLM) and Next Sentence Prediction (NSP). The

strength of BERT is flexibility, as it can be useful for different tasks and only requires a little bit of training [18]. In addition, the BERT model is designed for fine-tuning. This involves further training in a pre-existing model that can already grasp general language and also using custom labeled data for a more specialized downstream task.

To enhance the accessibility of Transformer-based models, the Hugging Face hub launched a platform where individuals can readily discover and apply pretrained models. This web-based model hub enables developers and researchers to leverage pre-trained models, removing the obligation to self-training. Through merely a few lines of code, users can import models like BERT or related versions for implementation in their projects, which makes NLP tools increase their accessibility and user-friendliness [19].

In addition, one useful tool originating from Hugging Face is DistilBERT. This is a more compact and faster type of BERT that was trained with fewer datasets but still performs almost as well as BERT and keeps close to 97% of BERT's effectiveness [19]. DistilBERT clearly shows how developers can make big models even when they cut down on time and with less computer effort, all while staying very accurate in understanding how people talk. Based on a study conducted by [22], the authors proved that BERT can effectively identify sentiment with high accuracy, surpassing traditional machine learning processes. Their model has achieved significant accuracy in terms of classifying multi-class sentiment categories, which validates the effectiveness of transformer-based architectures for sentiment analysis tasks. In line with this, the developers of this project aim for at least 85% classification accuracy. This target is attainable considering the information from the authors' study, similar to their multi-class sentiment classification. Moreover, the BERT model's capacity for handling huge volumes of text rapidly aligns with this project's objective of minimizing manual intervention and accelerating processing rate by at least 85%. By citing a proven high-performance NLP model and modifying it for a bilingual context, this confirms the effectiveness of the project's system accuracy and processing objectives.

In conclusion, this chapter investigates various ideas, models, and tools that help develop the project, a sentiment analysis system. Integrating role-based accounts and access, user-specific logins, and automated data processing are the key functions for structuring the project. This ensures that the web application will be able to perform efficiently and securely. All of these guidelines and insights are highly relevant and will significantly help improve the functionality of the sentiment analysis project.

Current System

Technical Background

The *IRAD (Information Resource and Analysis Division) department of DOST-STII (Science and Technology Information Institute)* currently performs sentiment analysis by reviewing large volumes of user feedback. After analysis, visualizations and insights are also generated manually.

To support this process, the IRAD department is currently utilizing tools such as:

- Google Forms: for collecting feedback from users.
- Google Sheets (Spreadsheet): where the user feedback goes, manually classifying sentiment, and generating visualizations and insights.

The current workflow is being done through desktop computers within the institution and is being used by IRAD personnel. Performing sentiment analysis is done within the intranet environment of the DOST-STII.

This manual approach is time-consuming and leads to inefficiencies in quickly identifying trends and addressing user feedback.

List of Processes

The process of sentiment analysis is currently being done by the IRAD department of DOST-STII.

Process ID	Process Name	Process Details
P001	Conduct events	IRAD conducts four different services/events that will later be evaluated.
P002	Gather User feedback	After each event/service, user feedback is collected through Google Forms
P003	Store user feedback on Google sheets.	Collected user feedback are automatically stored in Google sheets
P004	Manually analyzing of user feedback	IRAD personnel manually classifies the sentiment (positive, neutral, negative) from the user feedback collected in Google sheets.
P005	Generate visualizations and insights	Based on the manual sentiment classification, visualizations are manually created and filtered such as sentiment by gender, and service. Additionally, thematic analysis is performed by generating codes from user feedback. This is all done within Google Sheets.

Table 1. List of processes in the current system.

Table 1 lists the processes currently carried out manually by the IRAD department of DOST-STII, based on user feedback collected from events organized by DOST-STII. This

involves processes such as gathering, storing, analyzing, and visualizing user feedback primarily through Google Forms and Google Sheets.

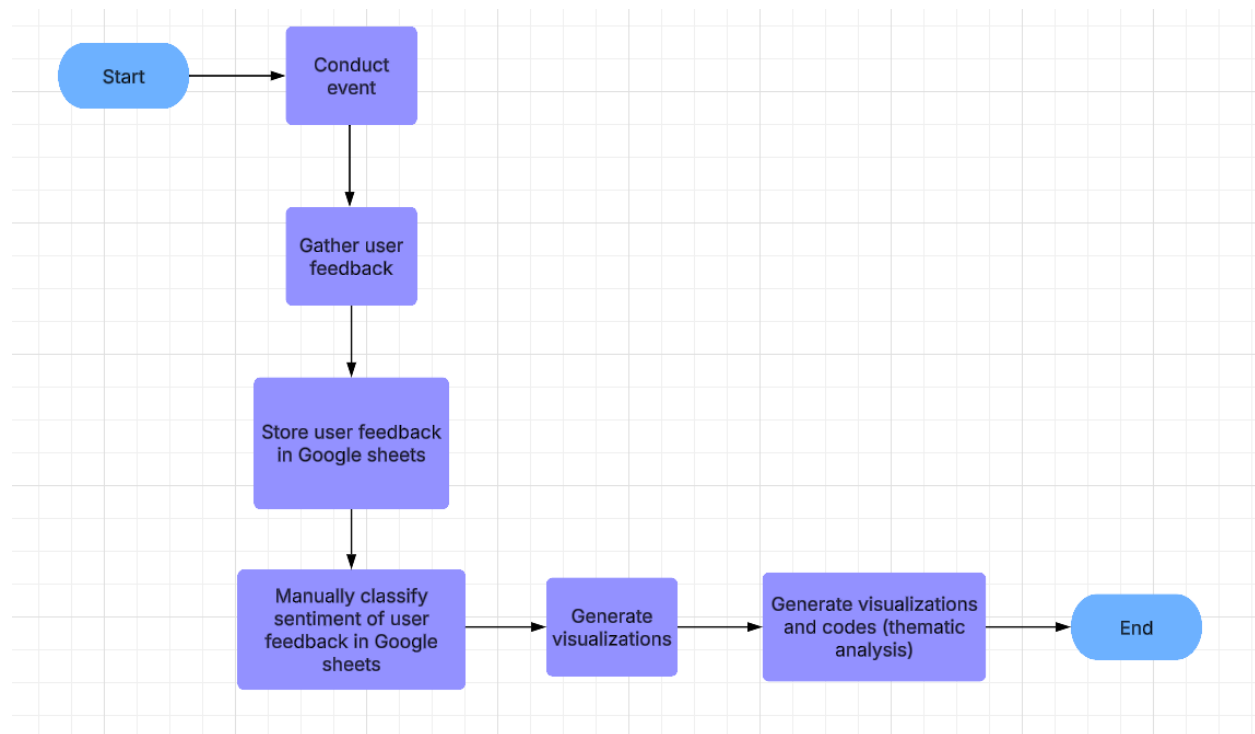


Figure 1. Process Flow of Current Sentiment Analysis of IRAD department

Figure 1 illustrates the manual workflow currently done by the IRAD department of the DOST-STII for collecting, analyzing, and visualizing user feedback manually.

These manual processes are laborious and time-consuming, which can limit the department's ability to quickly address user feedback on all services provided by the DOST-STII.

SWOT Analysis



Figure 2. SWOT Analysis of CitiSense web application

Strengths

Offline Operation. The current system runs inside the DOST-STII intranet, that is why it doesn't need any internet access. This offline operation helps keep all data secure and accessible at any time.

Established Tools. Since the collection of feedback is being done using Google Forms and Sheets, these tools are already familiar to the IRAD department, and it is easy to work with.

Low-cost tools. The current tools the IRAD department is currently using are free and don't require extra payment, which is affordable for the institution.

Data Centralization. All data collected is stored in Google Sheet, which makes it easier for the IRAD department to manage and organize the information.

Weaknesses

Manual Processing. The feedback collected is read and labeled manually, which takes a lot of time and effort from the institution.

Lacks Automation. DOST-STII has no system that automatically classifies or summarizes feedback, and this slows down the delivery of insight and delays the making of data-driven decisions.

Limited Capacity to handle volume. If there is a lot of feedback in Google Sheets, it becomes hard to keep up manually, so the current system is not suitable for handling large data quickly.

Redundant Tasks. The process of DOST-STII is repeated for each batch of feedback. This results in a waste of time and slower results.

Opportunities

Open-source Tools. Free tools such as Django can help build a better system that can be customized to fit DOST-STII's needs.

Custom Intranet Solutions. A system that is designed for the local network to be utilized fully offline. This will ensure that the security will not be compromised, and the integration will run smoothly with the current setups.

Custom Dashboards. Visual reports can be made in order to show real-time insights for the department to easily see feedback trends quickly.

Enhanced Responsiveness. Automation can help the department respond faster to issues or concerns.

Threats

Technical Limitations. Hardware may need upgrades in order to support a new system.

Adaptation Challenges. Some employees might find it hard to adjust to a new system and training will be needed for smooth work changes.

Human Error in Manual Processes. Manually handling responses can cause mistakes. If mistakes happen, this can affect the accuracy of the results.

Resource Requirements. A new system that will be utilized will need time, people, and effort to build, integrate, and maintain. Planning and support from all the teams are important in deploying and maintenance of a new system.

Proposed Solution

Technical Background

The proposed system, titled “CitiSense” is a web-based sentiment analysis system platform intended for use by the *IRAD (Information Resource and Analysis Division) department of DOST-STII (Science and Technology Information Institute)*. The system will be used within the intranet environment of the IRAD department, completely offline.

Citisense will automatically categorize user feedback into positive, neutral, or negative sentiment across all four of IRAD’s services. To achieve this, the system will utilize an NLP (Natural Language Processing) model, either pre-trained or fine-tuned, specifically for sentiment analysis on both English and Tagalog feedback. Additionally, the system will include a real-time visualization dashboard connected straight to the feedback database.

It will be developed using web technologies such as React (JavaScript framework) for serving the frontend UI, the Django (Python framework) for handling backend logic, and MySQL as the database for storing user feedback along with its corresponding sentiment classification.

Feasibility

Operational Feasibility

- Currently, the IRAD department is doing sentiment analysis work and generating charts manually, averaging over 6,000 user entries per service.
- The system will significantly streamline sentiment analysis workflow by automating sentiment analysis classification and visualization. Therefore, increasing efficiency and enabling fast delivery of actionable insights to decision-makers and stakeholders.
- As this will be the first sentiment analysis system within the IRAD department, they have expressed great interest in its implementation and will adopt the system immediately within their workflow.

Economic Feasibility

- Use of open-source software and tools for the creation of the proposed system.
- An NLP model specifically for sentiment analysis for both English and Tagalog. The NLP model will be sourced from the HuggingFace library, which is an open-source platform for offering pre-trained machine learning models. The models from HuggingFace are free and can either be used as-is or fine-tuned for better performance, without any cost.
- Development will be made possible to be free with open-source frameworks, Django for backend and React for frontend, and MySQL as the database.
 - As for the hosting of the frontend and backend servers, for frontend hosting, **Vercel** will be used with its free tier. For backend hosting, **Render** or AWS will be used.
- The anticipated benefits include a more efficient approach to performing sentiment analysis on user feedback from events, by streamlining the process. This will lead to faster

feedback-sentiment processing and visualization, and ultimately, improve the service quality of events.

Technical Feasibility

- The proposed system is technically feasible, as it utilizes widely adopted and already proven frameworks. These frameworks are well-documented; they also have a wide range of built-in features and third-party libraries that can streamline the development process.
- The front end will be developed using a JavaScript framework (React) which will provide a responsive and user-friendly interface. The backend will be developed using a Python framework (Django), which will be used for authentication, and integrating the NLP model. Lastly, the MySQL database stores user feedback and sentiment results.
- The core of the system involves a trained Natural Language Processing (NLP) model capable of processing feedback in both English and Tagalog. Additionally, visualization libraries will be used for data presentation.
- However, since ML models can yield misclassifications, a minimum classification accuracy of 70% across all positive, neutral, and negative cases is required to ensure reliability. If the threshold is not met, the chosen ML model will undergo fine-tuning using labeled feedback data from previous IRAD surveys.

Schedule Feasibility

- Following the Scrum framework, each sprint (development phase) will go on for a month. After each sprint, a sprint review will be done to showcase the work done and gather feedback from the IRAD department head. Consequently, a sprint retrospective will be done to ensure that the team can reflect, adapt, and enhance overall efficiency.
- A realistic project timeline has been set, with defined phases including model selection and fine-tuning, backend and frontend development, testing phases, and lastly, the deployment and hand-off phase.
- Potential risks that can cause delays have also been considered, such as poor NLP model accuracy in categorizing positive, negative, and neutral cases, and technical issues during deployment. Consequently, contingency plans have also been put in place, such as extensive testing before deployment, further fine-tuning of the NLP model used to enhance accuracy, and keeping previous stable builds of the system for redeployment (high availability).

Requirements Analysis

Project Vision

CitiSense is designed to give the DOST-STII IRAD department an intelligent platform to analyze feedback and turn it into visualized results. By automating the process of sorting and visualizing sentiments in English, Tagalog, and both, the system can bridge gaps and support data-informed decisions. This project envisions a transparent, and responsive government feedback process with the help of AI and real-time analytics.

Prototype (Mock Flow / Wireframe)



Register an account

The image shows a registration form with a light gray background. It contains three input fields labeled "Email", "Username", and "Password". Below the "Password" field is a link that says "Already have an account?". At the bottom is a yellow button labeled "Register".

Figure 3. Register UI of CitiSense web application



DOST-CitiSense

The image shows a login form with a light gray background. It contains two input fields labeled "Email" and "Password". Below the "Password" field is a yellow button labeled "Login". Below the "Login" button is a link that says "Forgot Password?". At the bottom is a button labeled "Sign in with Google" with the Google logo.

Figure 4. Login UI of CitiSense web application

Figure 5. Preview Data tab UI



Figure 6. Dashboard of the CitiSense web application

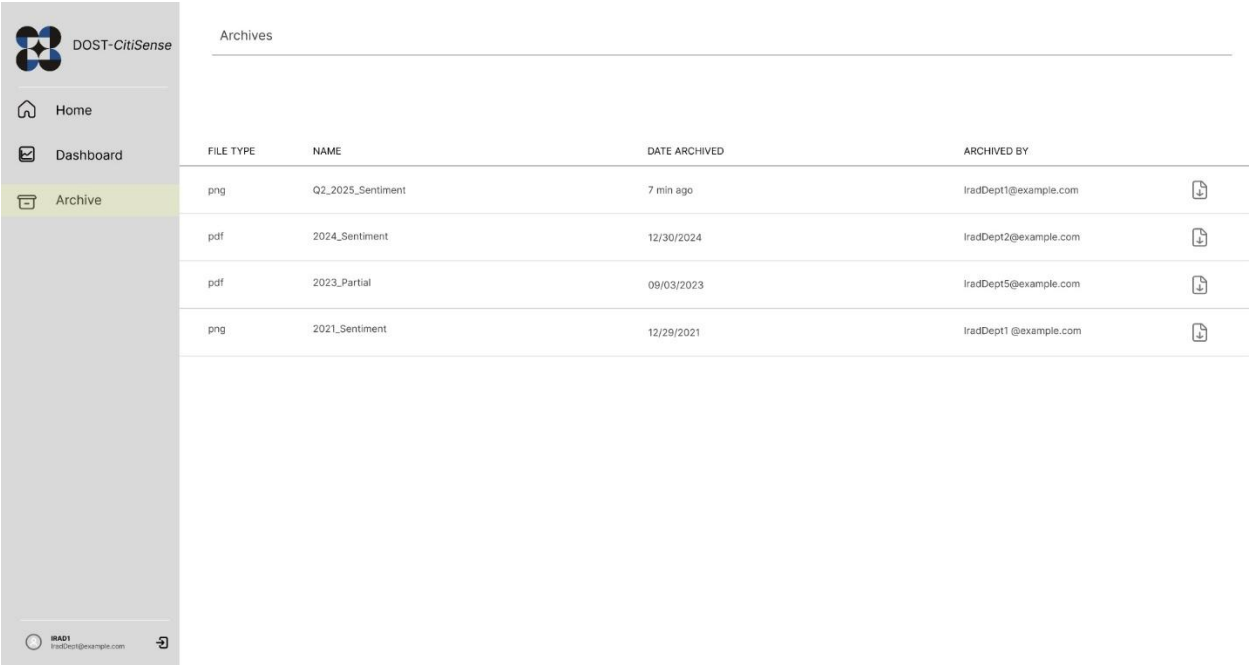


Figure 7. View archived visualization results of the CitiSense web application

Project Lean Canvas

Lean Canvas		Designed for:	Designed by:	Date:	Version:
		DOST-STII IRAD	QuadThink	06/04/25	1.0
Problem Manual analyzation of responses. Lack of real-time feedback visualization s. Different users need separate	Solution The system automatically classifies feedback into 3 categories (positive, negative, and neutral) The web application generates timely charts showing trends. System admins, data analysts, and general users	Unique Value Propo “Real-time multilingual feedback analysis with instant visualizations tailored to government agencies.	Unfair Advantage Tailored to analyze both English and Tagalog languages Designed for government feedback systems like DOST-STII	Customer Segments Educational Institutions Academic and Research Institutions Information Resources and Analysis Division (IRAD) Department of DOST-STII General User	

login accounts	have customized accounts tailored to their roles.			
Existing Alternatives Microsoft's Power BI allows organizations to connect with MS Forms to visualize responses.	Key Metrics Sentiment accuracy rate. System usage by roles (system admin, data analysts, general user). Number of event attendees processed feedback.	High-Level Concept Like Power BI, but with real-time analysis and fully automated.	Channels Internal Announcements Email Campaigns Government Partnerships and Endorsements	Early Adopters Employees of government agencies that gather feedback from event attendees. Data analysts who generate and analyze feedback. Departments that aim for data-driven solutions through feedback analysis. Institutions that seek an efficient system for understanding and processing feedback.
Cost Structure Web Application Development Machine Learning Libraries Hosting & Deployment Testing Environments			Revenue Structure Subscription Services Government funding One-time product sales	

User Classes and Characteristics

<i>Roles</i>	<i>Description</i>
<i>QuadThink Team (Developers)</i>	<i>The student development team responsible for planning, designing, building, and maintaining the web application. They implement features, fix bugs, conduct testing, and ensure the system meets user requirements. They also collaborate with stakeholders for feedback and improvements.</i>
<i>Project Adviser</i>	<i>The faculty mentor assigned to oversee the QuadThink team's development process. Provides technical and strategic guidance,</i>

	<i>reviews progress, ensures alignment with academic requirements, and helps resolve development issues.</i>
<i>Professor</i>	<i>The course instructor evaluates the team's project work. Oversees deliverables, checks compliance with the course outline, grades performance, and ensures the project meets educational standards.</i>
<i>DOST-STII Client</i>	The external stakeholder and primary client for whom the system is being developed. Provides project requirements and feedback. Validates that the solution aligns with organizational needs and may eventually adopt or deploy the system.

Table 2. User Classes and Characteristics involved in the CitiSense project

Product Backlog

CitiSense Product Backlog				
ID	As a...	I want to be able to...	So that...	Priority
1	System Administrator	Assign and update role-based access control permissions	Users can only access features relevant to their responsibilities to ensure data privacy.	Must
2	Data Analyst	Automatically categorize sentiment in both English and Tagalog user feedback	I can streamline the analysis of a large volume of user feedback from events.	Must
3	Data Analyst	Automatically generate a visualization of user sentiment on all the services provided by my department.	I can quickly interpret results based on user perception.	Must
4	Data Analyst	Automatically generate codes (thematic analysis)	I can quickly interpret patterns and generate themes	Must
5	Data Analyst	To archive and view past visualizations	I can reference previous analyses and track changes over time.	Must
6	Data Analyst	Filter through user sentiment by a specific service/s	I can generate more focused reports.	Must
7	User	Create and log in an account	I can retain my session, such as selected filters for visualizations, and continue where I left off.	Must
8	User	Edit my account details, such as email and password	I can personalize and secure my experience within the system.	Must

9	User	Login using my Google account	I can access the system more quickly and seamlessly using the same credentials I already use for Google Forms and Google Sheets in our current workflow	Should
---	------	-------------------------------	---	--------

Table 3. Product Backlog for the development of the CitiSense web application.

This is the product backlog of the CitiSense web application, a prioritized list of features and requirements that outlines what our web application should deliver. In the context of our product backlog, all features except one have priority marked as “Must”, because they contribute directly to the core functionality of the CitiSense web application. The only exception is the “Login using my Google account” prioritized as “Should” as it only enhances user convenience but is not essential to the system’s core operations.

The product backlog will serve as a guide for the development phase, breaking down functionalities into user-centered stories. Additionally, each user story has a corresponding priority level to help the development team focus on the features that have the highest priority level.

Product Roadmap

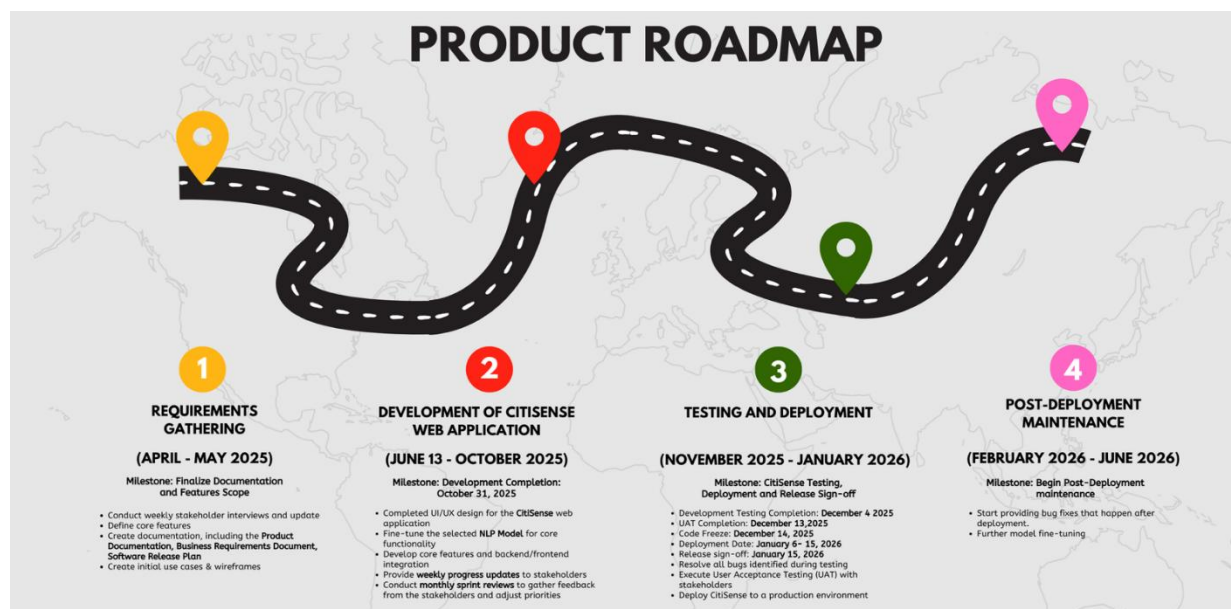


Figure 8. Product Roadmap for the development of the CitiSense web application.

The CitiSense product roadmap consists of the release dates that are essential to the development of the CitiSense web application. It helps set the pace for the development of CitiSense by providing clear development milestones. The CitiSense product roadmap not only keeps the QuadThink development aligned with the expected deliverables, but it will also help the stakeholders understand the timeline, and the expectations involved.

Release Plan

[Software Release Plan - CitiSense by QuadThink](#)

The QuadThink team is on schedule as shown in the Product Roadmap. As for the Product Backlog, it is shown in Table 3. As of now, the QuadThink team has conducted weekly interviews with the stakeholders, created initial use cases and wireframes, and has finished the documentation. The QuadThink team is preparing for the next phase which is the “Development of CitiSense web application.”

Use Case Diagram

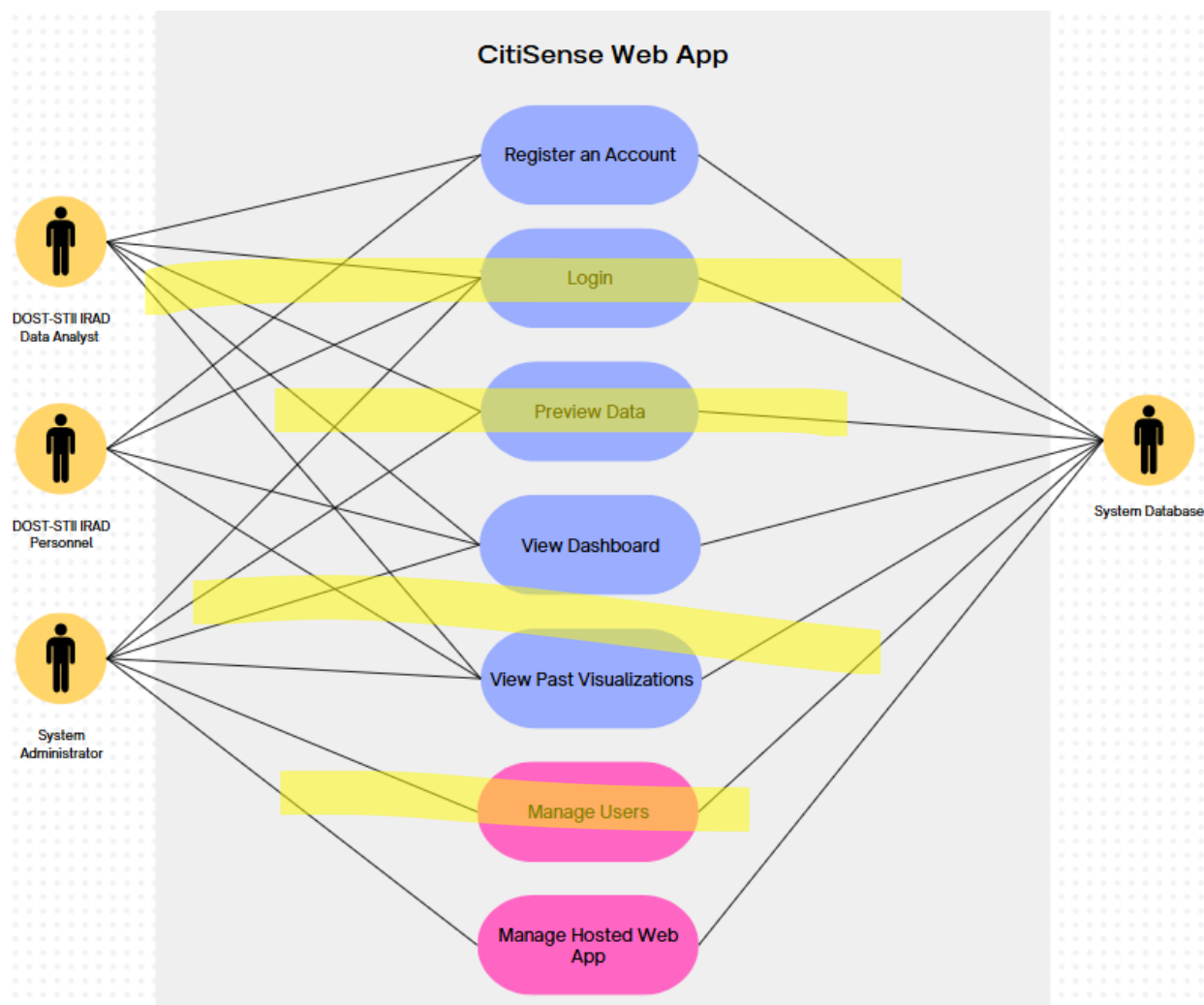
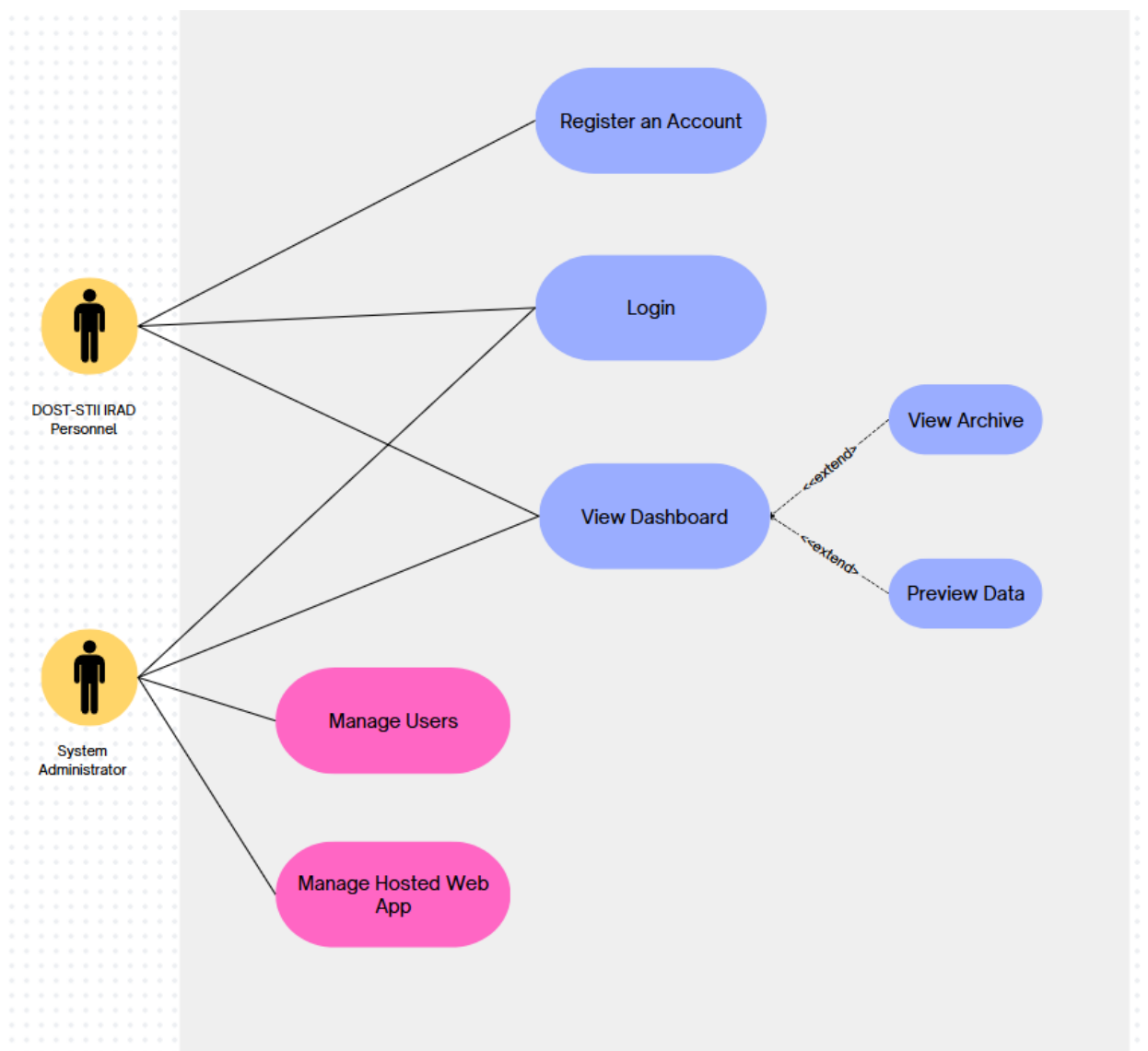


Figure 9. Use Case Diagram of the CitiSense web application



Use Case Full Description:

[Fully Dressed Use Case - QuadThink](#)

Conclusion

As of this stage in the development of *CitiSense*, significant progress has been made toward achieving the project's core objectives. The team has initially fine-tuned the bilingual NLP-based sentiment analysis model, capable of processing survey feedback in both English and Tagalog. This was accomplished through the fine-tuning of pretrained natural language processing models from the HuggingFace library, same with the dataset used to train the NLP model

Another major milestone achieved was the prototype automated sentiment classification system, which can evaluate and label survey responses as positive, negative, or neutral. Initial testing has shown somewhat good accuracy, in English, Tagalog, and code-switching language (Taglish).

On the other hand, some objectives remain unfinished. Notably, the web application interface and visualization components have not yet been developed. These are essential to ensuring that the analyzed data can be meaningfully presented to end users, such as the researchers and decision-makers at the DOST-STII IRAD department. Additional tasks also include implementing system usability features, testing larger datasets, and optimizing model performance.

For the next iteration, in the upcoming PBL subject, the focus will shift to:

- Completion of the UI/UX of the web application.
- The development of the *CitiSense* web application itself, with Django being the backend, and React as the frontend
- Fine-tuning the chosen NLP Model.
- A visualization dashboard based on the sentiment results.
- And further user testing and refinement to ensure the system meets the needs of DOST-STII IRAD department.

This next phase will move the project closer to a simple but functionable and deployable prototype, bringing the vision of *CitiSense* as a smart feedback analysis platform into full implementation.

References

- [1] P.K. Jain et al., “A systematic literature review on machine learning applications for consumer sentiment analysis using online reviews,” *Sci. Direct*, vol. 41, no. 100413, 2021. DOI: 10.1016/j.cosrev.2021.100413.
- [2] Y. Mao et al., “Sentiment analysis methods, applications, and challenges: A systematic literature review,” *Journal of King Saud University*, vol. 36, no. 4, pp. 102048–102048, 2024, DOI: 10.1016/j.jksuci.2024.102048.
- [3] G.S. Chauhan et al., “Aspect based sentiment analysis using deep learning approaches: A survey,” *Computer Science Review*, vol. 49, p. 100576, 2023, DOI: 10.1016/j.cosrev.2023.100576.
- [4] Bonta and Janardhan, “A Comprehensive Study on Lexicon Based Approaches for Sentiment Analysis,” *ResearchGate*. 2019. Available: https://www.researchgate.net/publication/333602124_A_Comprehensive_Study_on_Lexicon_Based_Approaches_for_Sentiment_Analysis
- [5] L. Gao et al., “Question-Driven Span Labeling Model for Aspect–Opinion Pair Extraction,” *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 14, pp. 12875–12883, 2021, DOI: 10.1609/aaai.v35i14.17523.
- [6] K. Bu et al., “Efficient Utilization of Pre-trained Models: A Review of Sentiment Analysis via Prompt Learning,” *Knowledge-Based Systems*, p. 111148, 2023, DOI: 10.1016/j.knosys.2023.111148.
- [7] A. Das & B. Gamback, “Code-Mixing in Social Media Text: The Last Language Identification Frontier?,” *Ntnu.no*, 2016, DOI: 1965-0906.
- [8] V. Srivastava & M. Singh, “Challenges and Considerations with Code-Mixed NLP for Multilingual Societies,” *arXiv (Cornell University)*, 2021, DOI: 10.48550/arxiv.2106.07823.
- [9] V. Kumar et al., “Enabling Code-Mixed Translation: Parallel Corpus Creation and MT Augmentation Approach,” *ACL Anthology*, pp. 131–140, 2018. Available: <https://aclanthology.org/W18-3817/>
- [10] M. Geva et al., “Are We Modeling the Task or the Annotator? An Investigation of Annotator Bias in Natural Language Understanding Datasets,” *Empirical Methods in Natural Language Processing*, 2019, DOI: 10.18653/v1/d19-1107.
- [11] S. K. Suresh et al., “CS-Sum: A Benchmark for Code-Switching Dialogue Summarization and the Limits of Large Language Models,” 2025. Available: https://www.researchgate.net/publication/391911130_CS-Sum_A_Benchmark_for_Code-Switching_Dialogue_Summarization_and_the_Limits_of_Large_Language_Models
- [12] G. Aguilar et al., “LinCE: A Centralized Benchmark for Linguistic Code-switching Evaluation,” *ACL Anthology*, pp. 1803–1813, 2020. Available: <https://aclanthology.org/2020.lrec-1.223/>

- [13] J.C. Rodriguez et al., "Frontend and Backend: The new approach to the development of a Web platform for automating the control and administration of degree processes at ITESI," *Revista de Tecnologías Computacionales*, vol. 8, no. 21, 2024, DOI: 10.35429/JOCT.2024.8.21.2.9.
- [14] F. K. Sufi and I. Khalil, "Automated Disaster Monitoring From Social Media Posts Using AI-Based Location Intelligence and Sentiment Analysis," *IEEE Transactions on Computational Social Systems*, vol. 11, no. 4, pp. 1–11, 2022, DOI: 10.1109/tcss.2022.3157142.
- [15] A. de Arriba et al., "Applying Transfer Learning to Sentiment Analysis in Social Media," *IEEE Xplore*, 2021. Available: <https://ieeexplore.ieee.org/document/9582287>
- [16] M. T. H. K. Tusar and Md. T. Islam, "A Comparative Study of Sentiment Analysis Using NLP and Different Machine Learning Techniques on US Airline Twitter Data," *IEEE Xplore*, 2021. Available: <https://ieeexplore.ieee.org/document/9641336>
- [17] A. M. Rahat et al., "Comparison of Naive Bayes and SVM Algorithm based on Sentiment Analysis Using Review Dataset," *IEEE Xplore*, 2019. Available: <https://ieeexplore.ieee.org/document/9117512?arnumber=9117512>
- [18] T. Wolf et al., "HuggingFace's Transformers: State-of-the-art Natural Language Processing," *arXiv:1910.03771 [cs]*, 2020, Available: <https://arxiv.org/abs/1910.03771>
- [19] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," *ArXiv*, 2018. Available: <https://arxiv.org/abs/1810.04805>
- [20] "Predictable pricing that scales with you," *Render*. <https://render.com/pricing>
- [21] Amazon Web Services, "What is Natural Language Processing? - NLP - AWS," *Amazon Web Services, Inc.*, 2024. <https://aws.amazon.com/what-is/nlp/>

Appendix B: Minutes of the Meetings

Minutes of the Meeting QuadThink:

https://docs.google.com/document/d/17QWkBi2pS5oeUuWN2A8IzVk7cuP_Y9MHVIfnLGiiTM4/edit?usp=sharing

Minutes of the Meeting with DOST and PBL advisor

<https://docs.google.com/document/d/1IM6fs4VZbhMvW3VQI21IGzrkiKydQhZq61mONDIUj7w/edit?usp=sharing>

Appendix C: Methodology

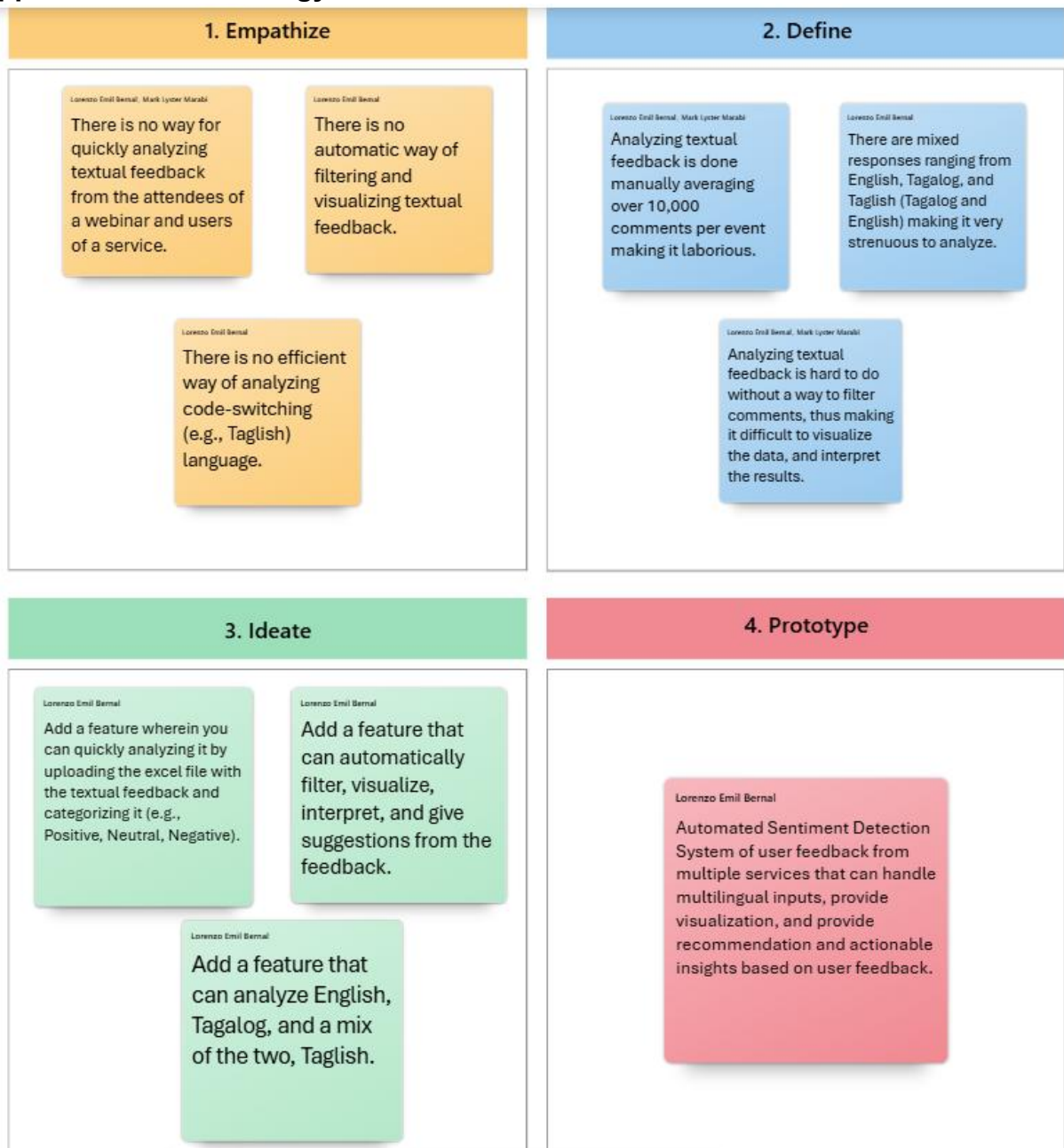


Figure 11. Design Thinking Quadrant

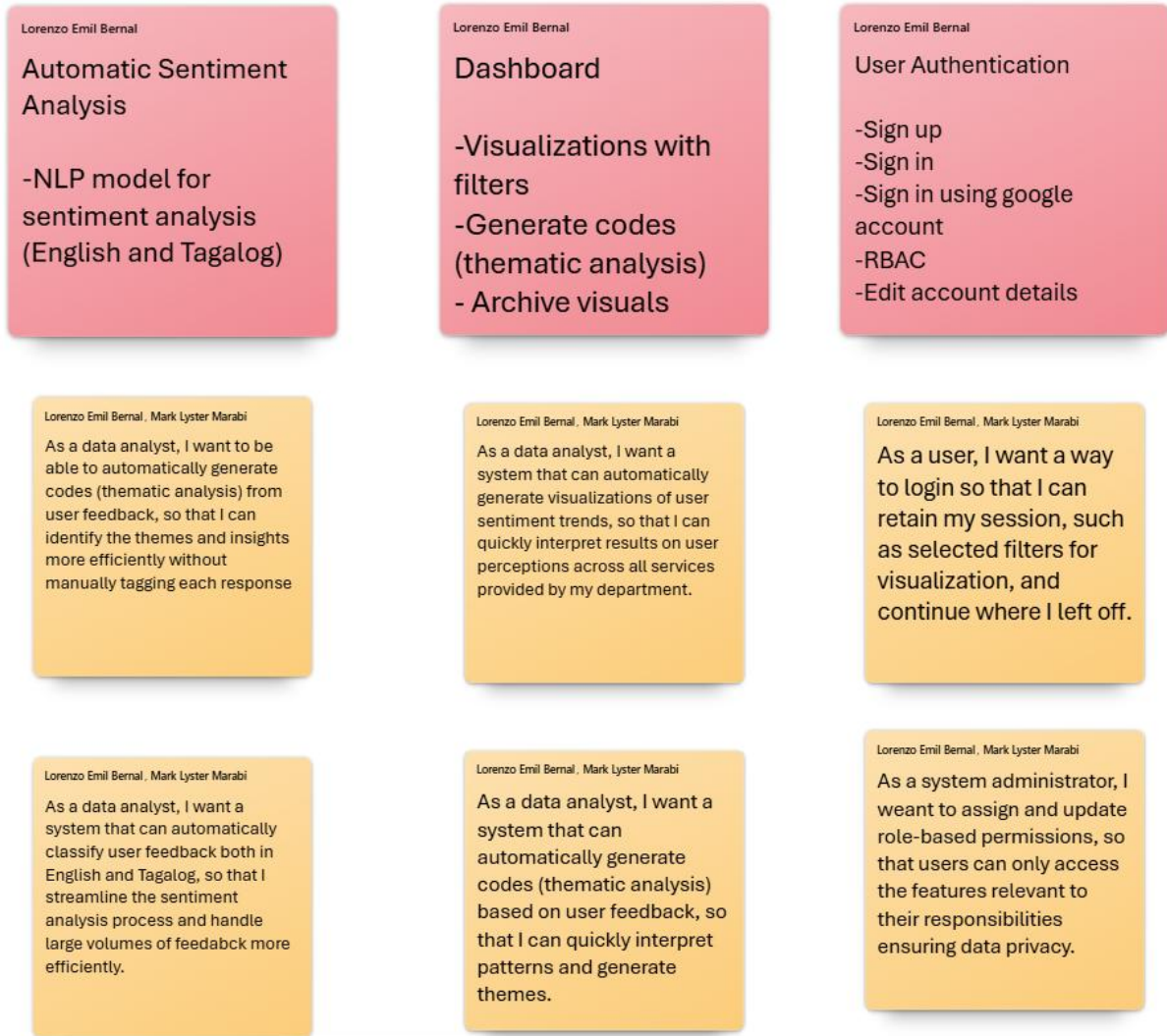


Figure 12. User stories for product backlog

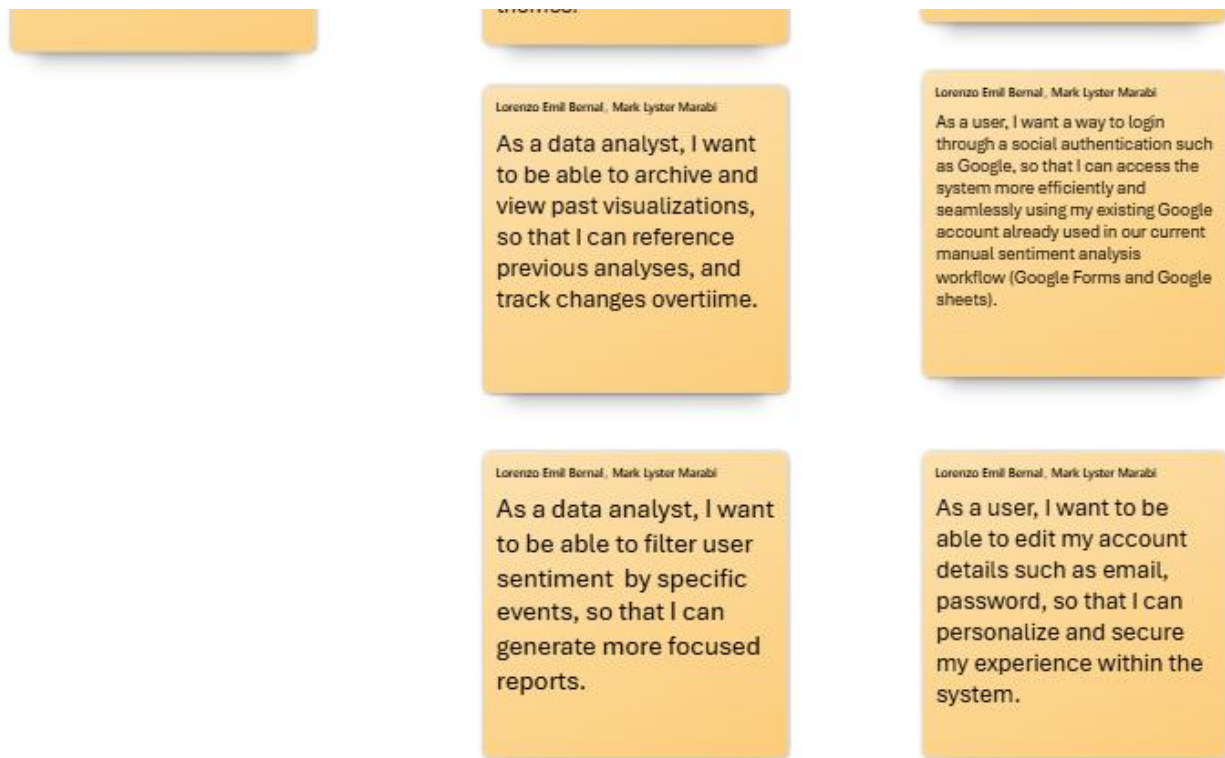


Figure 13. Continuation of user stories for product backlog

Appendix D: Project SharePoint Link

[QuadThink Sharepoint](#)

Appendix E: Use Case Full Description

Product Requirements

ID	Requirement
PR-01	DOST-STII IRAD Personnel and Data Analyst should register to access the web app
PR-02	All users should have the ability to login to the web app
PR-03	Only Data Analyst and System Administrator roles should be able to preview their data before analysis
PR-04	All users should be able to see the results displayed in the dashboard after the analysis
PR-05	All users should be able to see past analysis and visualizations
PR-06	System Administrators should be able to manage the hosted web app
PR-07	System Administrators should be able to assign or update user roles and permissions in the CitiSense web application.

ID	Requirement
PR-01	DOST-STII IRAD Personnel should register to access the web app

PR-02	All users should have the ability to login to the web app
PR-03	All users should be able to view the dashboard.
PR-04	System Administrators should be able to manage the hosted web app
PR-05	System Administrators should be able to assign or update user roles and permissions in the CitiSense web application

Table 4. Product Requirements for the CitiSense web application

Use Case Name: Register

Use case ID:	UC-01
Author:	Lorenzo Bernal, Gab Cabangcala, Luis Lazaro, Mark Marabi
Purpose:	DOST-STII IRAD Personnel and Data Analyst should register to access the web app
Requirement Traceability:	PR-01
Priority	High
Preconditions:	- Users must open the CitiSense web app
Postconditions:	- User must input valid credentials in registration - User information is verified in the system - Users should be able to login successfully
Actors:	DOST-STII IRAD personnel, DOST-STII IRAD Data Analyst, System Database
Extends:	
Flow of Actions:	Basic Flow <ul style="list-style-type: none"> - Open the CitiSense web app - Click "Register" button - Register account using valid credentials (Email, Username, and Password) - Click "Submit" - User must be redirected to the Home Page of the app - User will receive notifications of the registration being successful

Table 5. Use case: Register

Use Case Name: Login

Use case ID:	UC-02
Author:	Lorenzo Bernal, Gab Cabangcala, Luis Lazaro, Mark Marabi
Purpose:	All users should have the ability to login to the web app
Requirement Traceability:	PR-02
Priority	High

<i>Preconditions:</i>	- Users must be registered
<i>Postconditions:</i>	- User information is verified in the system - Users should be able to login successfully
<i>Actors:</i>	DOST-STII IRAD personnel, DOST-STII Data Analyst, System Administrators, System Database
<i>Extends:</i>	
<i>Flow of Actions:</i>	<p>Basic Flow</p> <ul style="list-style-type: none"> - Open the CitiSense web app - Click “Login” button - Input proper username and password - Click “Login” - User will be redirected to the Home Page of the app <p>Alternative Flow Unregistered user</p> <ul style="list-style-type: none"> - Click Login to the web app - User inputs valid credentials - The user is prompted with the message “User is not registered” - The user stays in the Login page

Table 6. Use Case: Login

Use Case Name: Preview data

<i>Use-case ID:</i>	UC-03
<i>Author:</i>	Lorenzo Bernal, Gab Cabangcala, Luis Lazaro, Mark Marabi
<i>Purpose:</i>	Only Data Analyst and System Administrators roles should be able to preview their own data before analysis
<i>Requirement Traceability:</i>	PR-03
<i>Priority:</i>	High
<i>Preconditions:</i>	 — Users must be registered — Users must be logged in — User must either have data analyst role or system administrator role — System is online
<i>Postconditions:</i>	 — Users can now be able to preview the data that will be made into a visualization chart.
<i>Actors:</i>	DOST-STII Data Analyst, System Administrators, System Database

<i>Extends:</i>	
<i>Flow of Actions:</i>	<p>Basic Flow</p> <ul style="list-style-type: none"> — Open CitiSense web app — Log in to the CitiSense web application — Click on “Preview Data” button — Preview all or choose from what service/s are to be reflected — Users will see the raw feedback data from the chosen service

Table 7. Use Case: References

Use Case Name: View dashboard

<i>Use case ID:</i>	UC-04
<i>Author:</i>	Lorenzo Bernal, Gab Cabangcala, Luis Lazaro, Mark Marabi
<i>Purpose:</i>	All users should be able to see the results displayed in the dashboard after the analysis
<i>Requirement Traceability:</i>	PR-04
<i>Priority</i>	High
<i>Preconditions:</i>	<ul style="list-style-type: none"> — Users must be registered and signed in. — System is online
<i>Postconditions:</i>	<ul style="list-style-type: none"> - Users can now see in real-time the trends and patterns of the feedback they got from event attendees.
<i>Actors:</i>	DOST-STII IRAD Personnel, DOST-STII IRAD Data Analyst System Administrators, System Database
<i>Extends:</i>	
<i>Flow of Actions:</i>	<p>Basic Flow</p> <ul style="list-style-type: none"> — Open CitiSense web app — Log in to CitiSense web application — Navigate to Dashboard — The CitiSense web app will show the visualization of sentiment analysis results from the feedback data of the chosen service/s. — Users will see the result of the analysis as it also updates at real-time pace

Use Case Name: View Dashboard

<i>Use case ID:</i>	UC-03
---------------------	-------

Author:	Lorenzo Bernal
Purpose:	All users should be able to view the dashboard.
Requirement Traceability:	PR-03
Priority:	High
Preconditions:	<ul style="list-style-type: none"> – User/s are registered – User/s are logged in – The system is online
Postconditions:	<ul style="list-style-type: none"> – Users can now see the dashboard
Actors:	DOST-STII Personnel, System Administrators, System Database
Extends:	
Flow of Actions:	<p>Basic Flow</p> <ul style="list-style-type: none"> – Open the CitiSense web app – Log in to the CitiSense web application – Navigate to Dashboard – The CitiSense web app will show the visualization of sentiment analysis results from the feedback data of the chosen service/s. – Users will see the result of the analysis as it also updates at a real-time pace <p>Alternative Flow</p> <p>View Archive</p> <ul style="list-style-type: none"> – Open the CitiSense web app – Log in to the CitiSense web application – Click on the “Archive” sub-process of the Dashboard button. – User/s will now see the past analyzed and visualized data with dates and who did the process on the information <p>Preview Data</p> <ul style="list-style-type: none"> – Open the CitiSense web app – Log in to the CitiSense web application

	<ul style="list-style-type: none"> – Click on the “Preview Data” sub-process of the Dashboard button. – Preview all or choose from the service/s to be reflected. – Users will see the raw feedback data from the chosen service
--	---

Table 8. Use Case: View Dashboard

Use Case Name: View past visualization

<i>Use case ID:</i>	UC-05
<i>Author:</i>	Lorenzo Bernal, Gab Cabangcala, Luis Lazaro, Mark Marabi
<i>Purpose:</i>	All users should be able to see past analysis and visualizations
<i>Requirement Traceability:</i>	PR-05
<i>Priority</i>	High
<i>Preconditions:</i>	<ul style="list-style-type: none"> — Users must be registered and signed in. — System is online
<i>Postconditions:</i>	<ul style="list-style-type: none"> — Users should now be able to see past analysis and visualizations
<i>Actors:</i>	DOST-STII IRAD Personnel, DOST-STII IRAD Data Analyst, System Administrators, System Database
<i>Extends:</i>	
<i>Flow of Actions:</i>	<p>Basic Flow</p> <ul style="list-style-type: none"> — Navigate to the CitiSense web app using the browser — Log in to CitiSense web application — Click the “Archive” button in the sidebar. — Browse the list of archived dashboard visuals. — Users will click on the visual they wish to view. — Users can then download the selected visual to their computer.

Table 9. Use Case: View past visualization

Use Case Name: Manage Hosted Web App

<i>Use case ID:</i>	UC-04
<i>Author:</i>	Lorenzo Bernal, Gab Cabangcala, Luis Lazaro, Mark Marabi

<i>Purpose:</i>	System Administrators should be able to manage their hosted website
<i>Requirement Traceability:</i>	PR-04
<i>Priority</i>	High
<i>Preconditions:</i>	<ul style="list-style-type: none"> - Administrators must be are logged in - Admin panel is accessible
<i>Postconditions:</i>	<ul style="list-style-type: none"> - User permissions are updated and saved in the system
<i>Actors:</i>	System Administrators, System Database
<i>Extends:</i>	
<i>Flow of Actions:</i>	<p>Basic Flow</p> <ul style="list-style-type: none"> - System Admin opens the CitiSense administrator panel. - System Administrator logs into admin panel with valid credentials - System verifies the credentials, if access granted, will be directed to admin panel - The system loads the configuration panel for the CitiSense. - The administrator can now perform the following: upload, update, and modify the CitiSense web application. - The system validates and applies changes - The administrator receives confirmation that changes have been applied successfully - The administrator verifies the changes are live as they should be reflected within the CitiSense web application.

Table 10. Use Case: Manage hosted web app

Use Case Name: Manage users

<i>Use case ID:</i>	UC-05
<i>Author:</i>	Lorenzo Bernal, Gab Cabangcala, Luis Lazaro, Mark Marabi
<i>Purpose:</i>	System Administrators should be able to assign or update user roles and permissions in the CitiSense web application.
<i>Requirement Traceability:</i>	PR-05
<i>Priority</i>	High
<i>Preconditions:</i>	<ul style="list-style-type: none"> - Administrators must be logged in. - Admin panel is accessible
<i>Postconditions:</i>	<ul style="list-style-type: none"> - User permissions are updated and saved in the system

Actors:	System Administrators, System Database
Extends:	
Flow of Actions:	<p>Basic Flow</p> <ul style="list-style-type: none"> - System Administrator opens the CitiSense administrator panel. - System Administrator logs into the admin dashboard with valid credentials. - System verifies the credentials, if access granted, will be directed to admin panel - The administrator navigates to the User Management section - The system displays a list of registered users - The administrator selects a user from the list - The system displays the selected user's current access roles and permissions - The administrator updates the user's roles (e.g., changes from "Data Analyst" to "Admin") - The administrator saves the changes - The system validates the changes and updates the database - The system will confirm the successful update

Table 11. Use Case: Manage users

Test Case Table

TC-ID	UC-ID	Test Case Name	Test Case Description
TC-01	UC-01	Verify user registration	DOST-STII-IRAD Data Analyst and Personnels should be able to register to the CitiSense web application.
TC-02	UC-02	Verify user login	All users should have the ability to login to the CitiSense web app
TC-03	UC-03	Verify that users can be able to preview their own data for analysis	All users should be able to preview the data from the database the system will generate visualizations for.
TC-04	UC-04	Verify that users can see the results being displayed in the dashboard after the analysis	All users should be able to see the results displayed in the dashboard after the analysis
TC-05	UC-05	Verify that users should be able to see past visualization and export them as an image.	All users should be able to see past visualizations and export them as an image to their computer.

TC-06	UC-06	Verify that only system administrators can be able to manage the hosted web app	Only System administrators should be able to manage their hosted web app
TC-07	UC-07	Verify that only system administrators can assign or update user roles and permission in the web system	Only System administrators should be able to assign or update user roles and permissions in the web system

TC ID	UC ID	Test Case Name	Test Case Description
TC-01	UC-01	Verify user registration	DOST-STII IRAD Personnels should be able to register to the CitiSense web application
TC-02	UC-02	Verify user login	All users should have the ability to login to the CitiSense web app
TC-03	UC-03	Verify that users can be able to view the dashboard and its sub-processes	All users should be able to preview the dashboard. Users can also opt to preview the data and view their archived data.
TC-04	UC-04	Verify that only system administrators can be able to manage the hosted web app	Only System administrators should be able to manage their hosted web app
TC-05	UC-05	Verify that only system administrators can assign or update user roles and permissions in the web system	Only System administrators should be able to assign or update user roles and permissions in the web system

Table 12. Test Case Table for the CitiSense web application

Appendix F: Requirements Traceability Matrix

Product Requirement No.	Use Case ID	Test Case ID
PR-01	UC-01	TC-01
PR-02	UC-02	TC-02
PR-03	UC-03	TC-03
PR-04	UC-04	TC-04
PR-05	UC-05	TC-05
PR-06	UC-06	TC-06
PR-07	UC-07	TC-07

Product Requirement No.	Use Case ID	Test Case ID
PR-01	UC-01	TC-01
PR-02	UC-02	TC-02
PR-03	UC-03	TC-03
PR-04	UC-04	TC-04

PR-05	UC-05	TC-05
-------	-------	-------

Table 13. Requirements Traceability Matrix for the CitiSense web application

Appendix G: RACI Matrix

RACI Matrix				
	QUADTHINK TEAM	DOST-STII CLIENT	PBL ADVISOR	PBL PROFESSOR
Documents	R / A	I	C	I
System Development	R / A	C	C / I	
Figma Presentation	R / A	C		
Use Case Diagram	R / A		C	C

Figure 14. RACI Matrix of the CitiSense of the project development

