Chapter 6: Results and Discussions

Our study focused on analyzing the inventory of diamond stocks using a specially designed system. The system has been tested on a comprehensive dataset consisting of numerous files that have been collected and analyzed. While our current implementation has shown promising results, it is important to acknowledge the potential for future failures. It is possible that the system may encounter difficulties in processing certain types of files, particularly when handling miscellaneous file formats. However, it is worth noting that the system's failure would typically be limited to specific components rather than a complete breakdown.

The cornerstone of our analysis lies in the functionality provided by the 'Common Converter' code. This code plays a pivotal role in the analysis process, as the output file generated by the 'Common Converter' serves as the input for subsequent analysis. Through our rigorous development efforts, we have ensured that the 'Common Converter' code functions reliably and consistently.

In the analysis section, we have successfully processed both supplier and user files. These files serve as the basis for creating dictionaries that capture specific requirements. By amalgamating these dictionaries, we enable effective communication of requirements between suppliers and users. This facilitates smoother collaboration and improves overall efficiency in the inventory management process. The processed master file can be seen in Figure 6.1.

Furthermore, our system boasts a user-friendly frontend interface that can be triggered through API calls or bucket triggers. This frontend system has been thoroughly tested and performs exceptionally well. Its versatility is evident in its ability to handle multiple file uploads and facilitate easy downloads. Our code has been designed to be universally compatible, making it accessible to other teams seeking to convert their files and streamline their workflow. Snapshots of Frontent are portrayed in Figure [6.2-6.3].

In conclusion, our diamond stock inventory analysis system has demonstrated its effectiveness in processing and analyzing a vast array of files. While potential future challenges may arise, particularly with miscellaneous file formats, our 'Common Converter' code has been robustly developed to ensure

reliable conversion. The analysis section has successfully utilized supplier and user files to generate comprehensive dictionaries, promoting effective communication between stakeholders. The user-friendly front-end system adds an additional layer of convenience, allowing for seamless file handling. Overall, our system has been well-received and widely adopted by other teams seeking to enhance their inventory management processes.

clarity	carat	color	shape	fluoresce	raprate	cut	polish	symmetry	table	length	width	depth	price per	discount
vs2	1.08	h	round	m	9100	ex	ex	ex	57	6.59	6.58	4.07	5425.42	40.38
vs1	1.01	i	round	f	7800	ex	ex	ex	58	6.42	6.37	4	5580.12	28.46
vs2	1.01	i	round	m	7400	ex	ex	ex	59	6.42	6.37	3.99	5072.7	31.45
vs2	1.01	i	round	n	7400	ex	ex	ex	58	6.44	6.39	3.96	5882.26	20.51
si1	1.01	i	round	n	6700	ex	ex	ex	57	6.41	6.37	4.01	5059.84	24.48
vs1	1.2	d	round	n	13700	ex	ex	ex	56	6.81	6.75	4.27	10480.5	23.5
vs1	1.2	f	round	m	12100	ex	ex	ex	58	6.79	6.74	4.25	7453.6	38.4
vs2	1.2	f	round	m	10700	ex	ex	ex	58	6.83	6.79	4.19	7441.85	30.45
si1	1.2	f	round	n	8500	ex	ex	ex	56	6.77	6.72	4.26	6755.8	20.52
si1	1.26	g	round	m	8100	ex	ex	ex	56	6.87	6.84	4.33	6036.12	25.48
vs2	1.2	h	round	n	9100	ex	ex	ex	57	6.79	6.74	4.23	7322.77	19.53

Figure 6.1: Screenshot of Master File Output

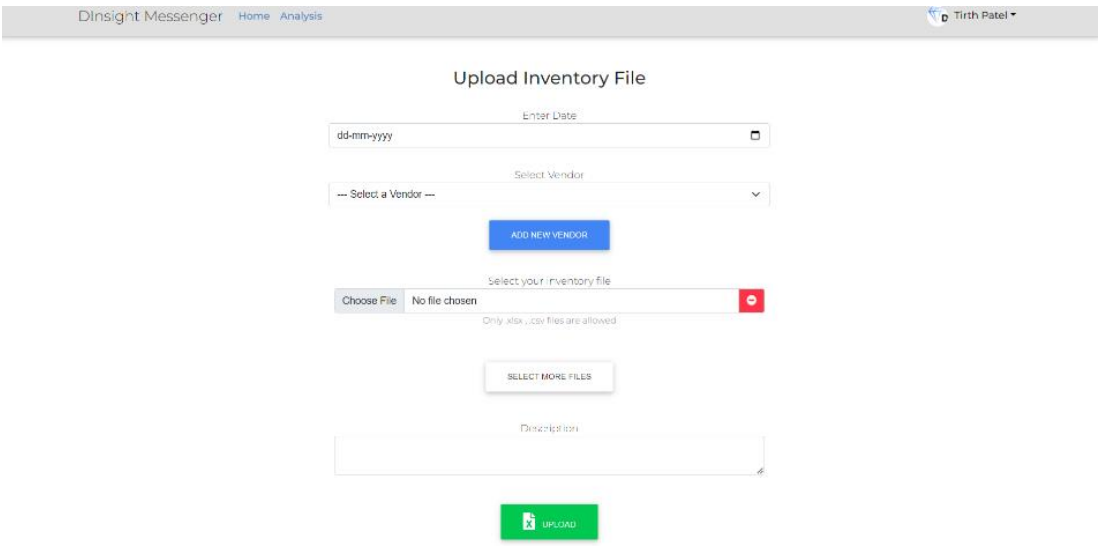

Figure 6.2: Snapshot of Frontend Home Page

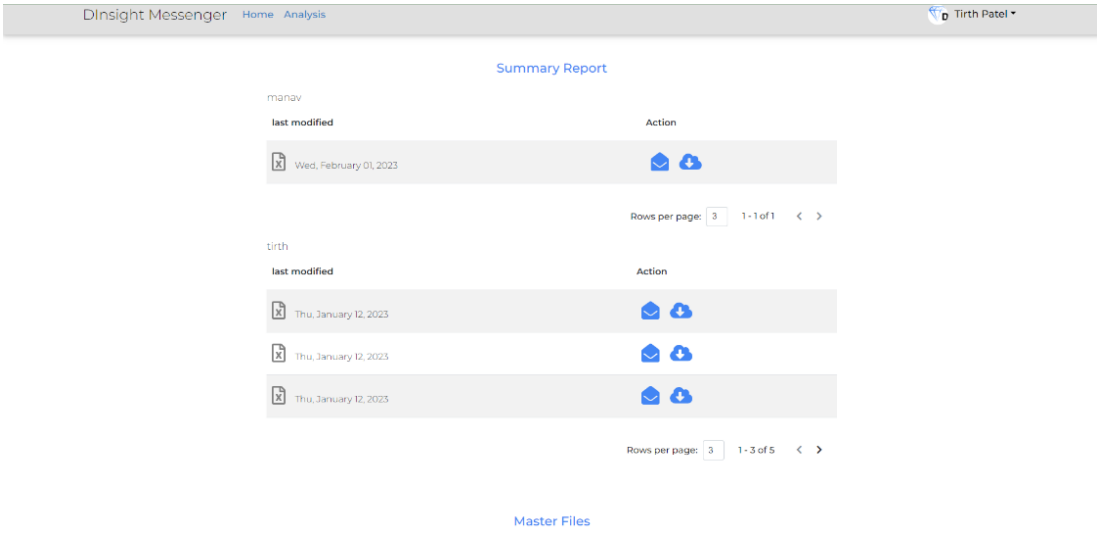

Figure 6.3: Snapshot of Analysis Page

Chapter 7: Conclusion and Future Work

7.1 Conclusion

During this internship, our primary objective was to convert files into a standard format. Throughout the course of the internship, we acquired knowledge of various tools and technologies such as cloud function (GCP). Initially, our code was unorganized, but we later recognized the importance of writing organized code using proper classes and functions. We also realized the significance of unit testing. However, we faced challenges, including bridging communication gaps due to time differences with our supervisor in New York, encountering merge conflicts during code merging, and difficulties in debugging the code. Eventually, we successfully developed a robust statistical model to extract target columns, modularized the code, performed unit testing to improve the code's reliability, and deployed the code in various ways to enable easy usage for our team and others. Overall, this internship provided valuable experience that will undoubtedly benefit me in my future endeavors.

7.2 Future work

Moving forward, our primary objective within the analysis component of our project is to facilitate comparison between various supplier files across different days, allowing for the monitoring of products sold and added to the inventory. Moreover, we anticipate that this analysis could be equally beneficial for individual users, who could leverage our tools to evaluate their own files across multiple dates, tracking price fluctuations, new item additions and removals from their inventory.

Through the performance of these analyses, we expect to provide users with valuable insights into the trends and movements of the market. Additionally, we have developed a categorization scheme, based on the dictionary keys used in our analyses, which will enable us to more efficiently organize and classify the data.

Furthermore, our analysis will extend beyond simple counting, exploring additional dimensions of the data to identify patterns and trends related to item age, and other pertinent factors. Ultimately, the output

of this analysis will be a comprehensive price chart, providing users with the insights necessary to make informed market decisions.

Our plan is to centralize all team codes into a single account, providing us with streamlined file management capabilities and seamless access to their GCP buckets. This approach will eliminate the need for sharing files through emails, which can be cumbersome and restricted by size limitations. Instead, we can directly access and manage the files within the shared account, eliminating the need for dividing and sending files through multiple emails. This centralized approach will enhance collaboration and efficiency within the team, allowing for easier access and sharing of code files.

Chapter 8: Additional Project: Voice Intelligence

System incorporates the Alexa interface on the frontend and Java on the backend to create a voice intelligence application. The primary purpose of this system is to engage children aged 7-10 in a quiz-like activity, where Alexa asks questions and maintains their scores based on correct answers shown in Figure 8.3. By utilizing Alexa as the interface, I capitalize on the increasing familiarity and convenience that children have with voice-activated devices, eliminating the need for manual typing and providing a more interactive experience.

To initiate the voice intelligence functionality, users invoke Alexa skills by issuing a specific speech command, such as "call voice intelligence." Subsequently, Alexa retrieves data from the AWS RDS database and prompts users to choose their preferred set of questions and set of modules are shown in Figure 8.2. The integration between AWS RDS and the Alexa skill kit enables seamless communication, as triggering Alexa prompts the lambda function to connect with the AWS database. Process Diagram is shown in Figure 8.1

My implementation involves deploying a jar file containing the necessary code on the lambda function, specifying its primary handler name. Additionally, I have developed an admin panel with login and register pages shown in Figure 8.4, ensuring security measures are in place. Access to the database is restricted to the admin, who can leverage the frontend panel to add more questions and skills. The admin panel is configured using the Spring framework, with Spring Boot as a dependency, ensuring smooth and efficient management of the system.

The underlying motivation for creating this functionality lies in addressing the concern of children being exposed to uncensored and unhealthy content. By providing an engaging and enjoyable voice intelligence experience, I aimed to provide a safe and educational alternative, where children can interact with Alexa and learn through interactive quizzes.

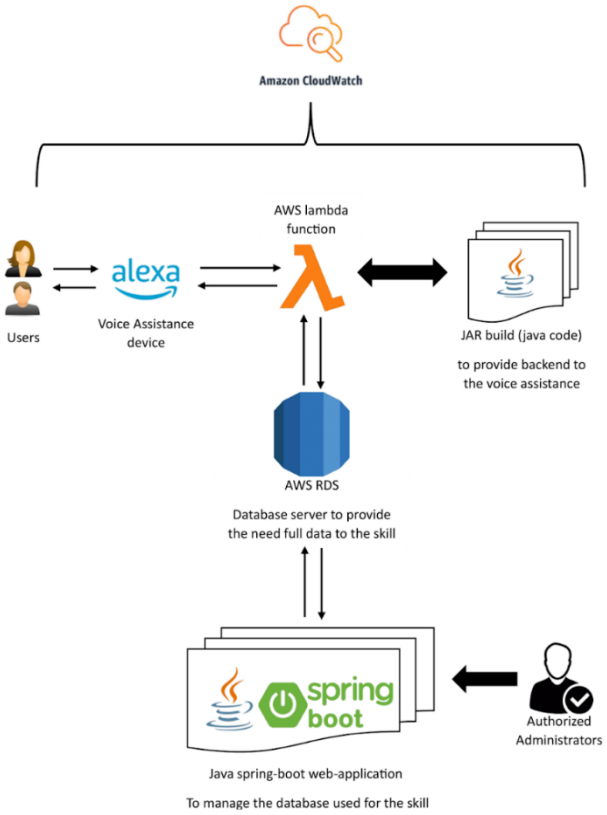

Figure 8.1: Process Diagram

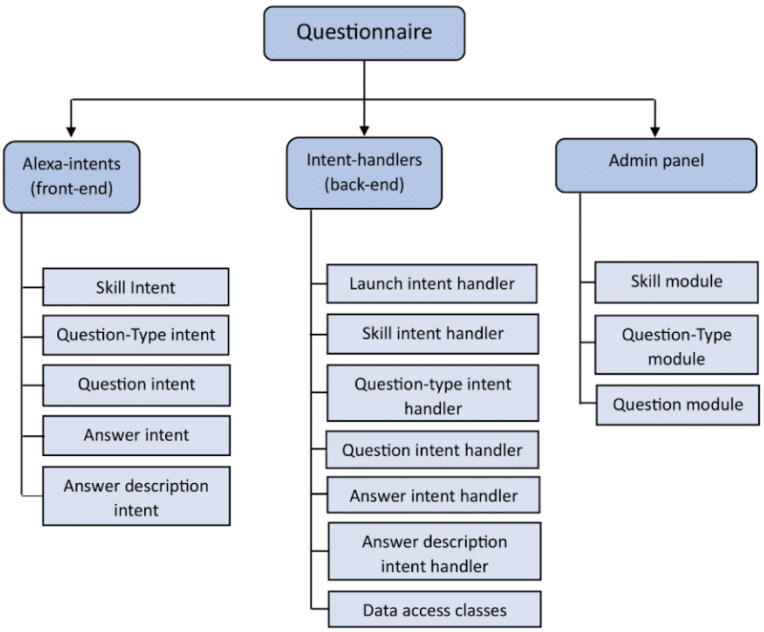

Figure 8.2: Questionnaire Modules

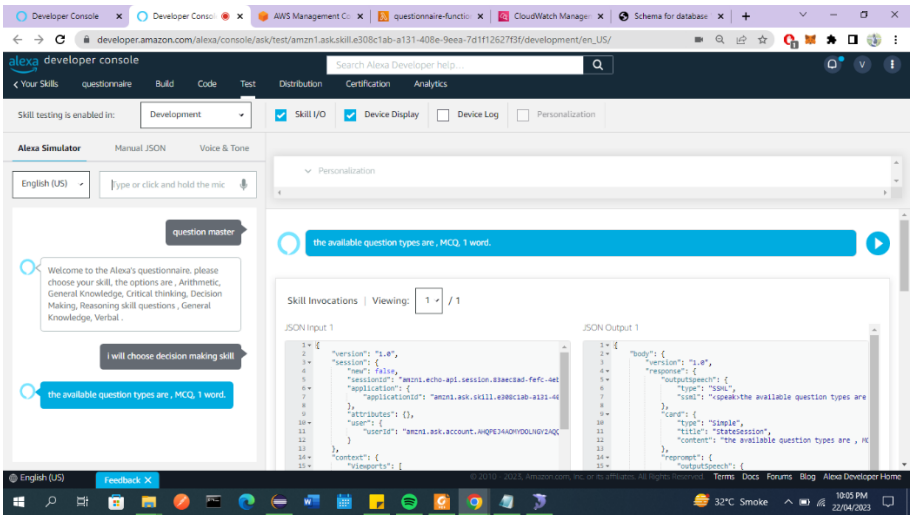

Figure 8.3: Snapshot of Alexa (Frontend)

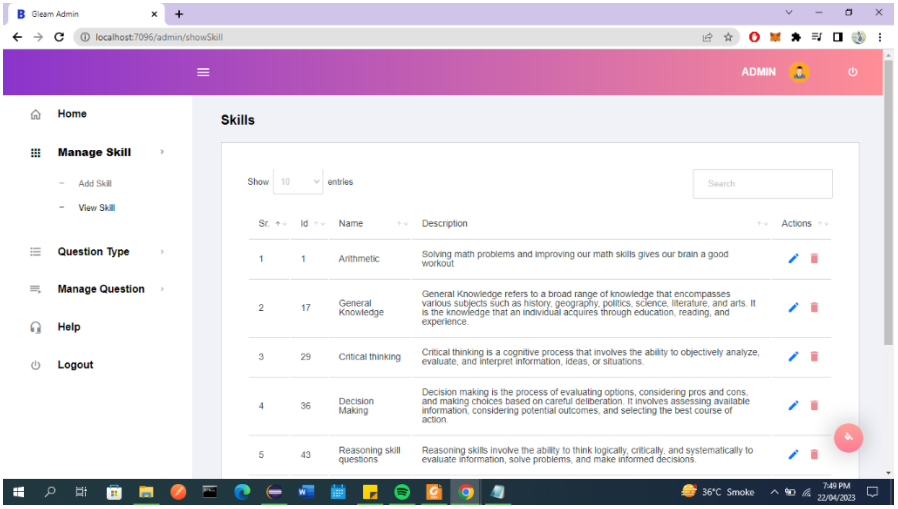

Figure 8.4: Admin Panel

In the development of our Voice Intelligence system using Alexa, we have employed a design pattern that utilizes POJO classes and Java objects as the main components of our logic code. This design pattern enhances the structure and modularity of the code, allowing for easier maintenance and future feature enhancements. While the current version of the project is fully functional, there are plans to further enhance its capabilities through the implementation of Natural Language Processing (NLP) configuration.

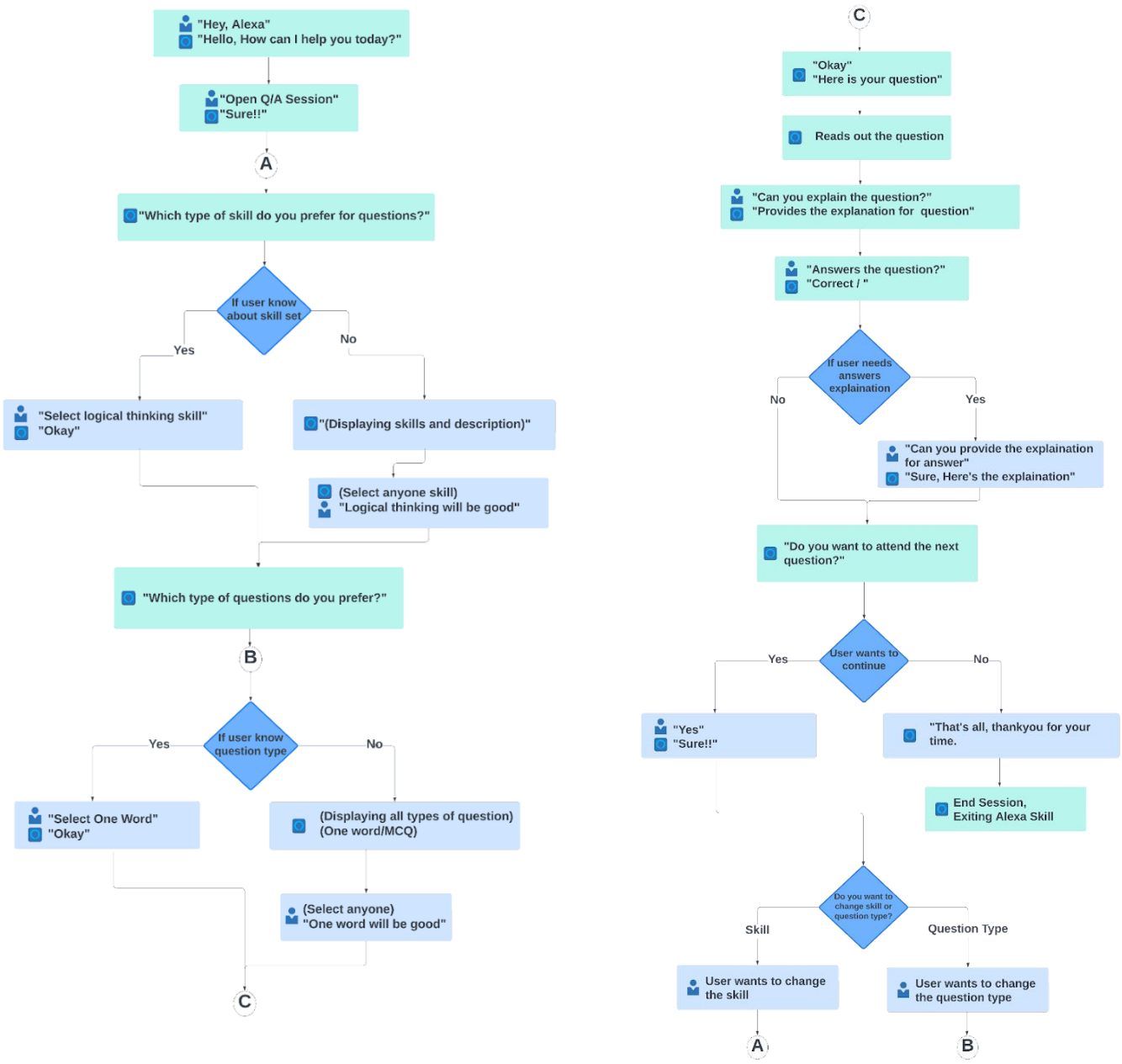

Figure 8.5: Work Flow of System

By collaborating with educators and understanding the specific needs of school-going children, I will tailor the system accordingly. Recognizing its potential as an invaluable educational tool, I aim to deploy our Voice Intelligence system as a product for schools, aligning with the curriculum and addressing the needs of teachers and students. Through continuous refinement and consultation, our goal is to deliver an impactful solution that supports students' educational journey.