



GROUP PROJECT IN SOFTWARE DEVELOPMENT – CS2993

Final Thesis

Group 07

**Computer Store Management System with AI
Chat Bot**

**General Sir John Kotelawala Defense University
Faculty of Computing**

Group Member Details

Reg.No	Name	Degree
D/BCE/23/0008	P H D B Nayanakantha	COMPUTER ENGINEERING
D/BCE/23/0002	B D S D Douglas	COMPUTER ENGINEERING
D/BCE/23/0004	R M M Ranathunga	COMPUTER ENGINEERING
D/BCE/23/0014	R B H P Rathnamalala	COMPUTER ENGINEERING
D/BCS/23/0009	P R M K Herath	COMPUTER SCIENCE

Supervisor: Dr. Kaneeka Vidanage

2024 January – 2024 November

Abstract

This thesis presents the development of a **Computer Shop Management System** designed to enhance the customer experience and streamline business operations through a combination of a web-based interface and AI-powered chatbot modules. The project comprises three main modules: the **Customer Web Interface Module**, the **AI Chatbot for Customer Module**, and the **AI Chatbot for Owner Module**. The system aims to provide customers with a seamless shopping experience, from browsing products to managing their orders, while offering store owners real-time insights into sales and inventory.

The **Customer Web Interface Module** allows users to register, log in, browse products, search by categories, and interact with personalized product recommendations. The interface includes features such as Wishlist management, and product reviews, enhancing customer engagement and satisfaction.

The **AI Chatbot for Customer Module** is integrated within the web interface to assist users with inquiries related to product details and general store information. Powered by natural language understanding (NLU), the chatbot interprets user queries and provides accurate responses. Additionally, the chatbot supports voice input, allowing users to interact through speech for a more dynamic and accessible experience.

The **AI Chatbot for Owner Module** is designed to help store owners manage their operations efficiently. It allows secure access to sales data, inventory updates, and reports. The owner can generate real-time reports, monitor stock levels, and track business performance through a conversational interface. Throughout the development, an agile methodology was adopted, ensuring iterative improvement and collaboration between developers and stakeholders. Comprehensive testing, including unit testing, integration testing, user testing, and system testing, was conducted to verify the functionality, usability, and security of each module. User feedback was gathered through surveys, and insights were applied to refine the system for optimal performance.

The system's implementation demonstrates the potential of AI-driven solutions in e-commerce, offering improved customer service and operational efficiency. This project provides a robust foundation for further enhancements, such as the future integration of payment gateways, advanced analytics, and AI-powered recommendations.

Keywords: Web Interface, AI Chatbot, E-Commerce, Natural Language Understanding (NLU), Computer Shop Management, User Experience, Agile Methodology

Acknowledgements

We would like to express our deepest gratitude to those who have contributed to the successful completion of this project. First and foremost, we would like to thank our supervisor, **Dr. Kaneeka Vidanage**, for his invaluable guidance, encouragement, and support throughout the course of this project. His expert advice and constructive feedback have been essential in shaping our ideas and improving our work.

We would also like to extend our heartfelt thanks to the **Dean of the Faculty of Computing, Dr. LP Kalansooriya**, for providing us with the resources and infrastructure needed to carry out our research and development. His leadership and support have facilitated the progress of this project.

A special thank you to **Ms. KGK Abeywardhane**, whose insightful lectures and continuous encouragement inspired us to explore the intersection of AI and e-commerce. Her expertise and practical insights helped us overcome many challenges during the development phase.

We are also grateful to the **faculty and staff of the Faculty of Computing** for their ongoing support, and for creating an environment conducive to learning and innovation. Their dedication has motivated us throughout the course of this project.

Finally, we would like to acknowledge the support and collaboration of our fellow students and peers, who provided helpful suggestions and feedback that contributed to the success of this project.

Thank you all for your continued belief in our work and for helping us reach this milestone.

Table of Contents

Group Member Details	ii
Abstract.....	iii
Acknowledgements.....	iv
Table of Contents	v
List of Figures	vii
List of Tables.....	ix
Project Executive Summary	1
Project Proposal	2
1.1 Introduction	2
1.1.1 Background and Motivation.....	2
1.1.2 Problem Statement.....	3
1.1.3 Aim	3
1.2 Literature Review	5
1.3 Proposed Methodology.....	8
1.3.1 Flow Diagram	9
1.3.2 Functional Requirements and Non-Functional Requirements	10
1.3.3 Proposed Testing and Evaluation Method.....	13
1.4 Time Frame	14
1.5 Budget	14
Analysis Report.....	15
1.6 Project Relevance.....	15
1.7 Initiation of the Project	15
1.8 Brief Description of the organization.....	15
1.9 Vision for project completion	16
1.10 - Fact Finding	17
1.10.1 Justification of Fact-Finding Techniques Used.....	17
1.10.2 Completeness and Quality of Fact-Finding	18
1.10.3 Analysis of Information Gathered.....	20
1.11 - Description of current system.....	22
1.11.1 Narrative	22
1.11.2 Detailed process flows (DFD)	29
1.11.3 Problems and Limitations of the current system.....	30
1.11.4 Scope and Constraints.....	31
1.12 - Requirements specification for new system.....	32
1.12.1 Module 1: Web interface (Website)	32

1.12.2 Module 2: AI Chatbot for Customer Interaction	33
1.12.3 Module 3: AI Chatbot for Shop Owner	34
Design Report.....	37
1.13 Description of new system.....	37
1.13.1 Narrative	37
1.13.2 Process flows.....	39
1.14 Detailed Design	40
1.15 Data Design	40
1.16 Architectural Design.....	46
1.17 Procedural Design	48
1.17.1 Process Flow for the AI Chatbot for Customers.....	48
1.17.2 Process Flow for the AI Chatbot for Owners.....	48
1.17.3 Process Flow for the Website	48
1.17.4 Combined System Process Flow.....	49
1.18 Interface Design	53
1.19- Hardware and Software selection.....	57
1.19.1 Development hardware and software equipment	57
1.19.2 Implementation hardware and software equipment	59
Implementation and Development Report.....	61
1.20 Software Development: Testing	61
1.20.1 Justification of testing technique	61
1.20.2 Test Case Design and Justification	64
1.20.3 Test logs	68
1.21 Implementation: Conversion and Training Plan	75
1.22 Justification of conversion technique	75
1.23 Schedule and Planning for Conversion	75
1.24 Implementation: User Guide	80
1.25 Critical Appraisal: combined	91
1.26 Critical Appraisal: individual.....	92
References.....	95
Appendix	98
Appendix 1: Stakeholder Interviews	98
Appendix 2: Observational Studies	100
Appendix 3: Software Development Services Request Form	103
Appendix 4: Current system user interface	106

List of Figures

Figure 1 : Proposed Methodology.....	9
Figure 2: Time Plan.....	14
Figure 3: System Breakdown.....	22
Figure 4 : Organizational Roles and Responsibilities	23
Figure 5: Logging Page.	24
Figure 6: Invoice Window.	24
Figure 7: Cancel Invoice Window.....	25
Figure 8: Payment Window.....	25
Figure 9: Payment Return Window.....	26
Figure 10:Stock Adjustment Window.	26
Figure 11: Add Items Window.....	27
Figure 12:Search Item Window.....	27
Figure 13: Invoice Outstanding Window.....	28
Figure 14 : Detailed Process Flows of Current System.....	29
Figure 15: System Breakdown Structure.....	39
Figure 16: Work Breakdown Structure	40
Figure 17: Level 0 Dataflow Diagram	41
Figure 18: Level 1 Dataflow Diagram	41
Figure 19: Class Diagram for Database	42
Figure 20: ER Diagram.....	43
Figure 21: Use Case Diagrams	46
Figure 22: High level architecture design.....	47
Figure 23: Flow chart for customer chatbot	51
Figure 24: Flow chart for Owner chatbot.....	51
Figure 25: Flow chart for website	52
Figure 26: UI for customer chatbot welcome page	53
Figure 27: UI for customer chatbot.....	53
Figure 28: UI for customer chatbot FAQ page	54
Figure 29: UI for customer chatbot feedback page	54
Figure 30: UI for Owner Chatbot.....	55
Figure 31: Website Home page UI	55
Figure 32: Website Contact page UI	56
Figure 33: Intent Prediction Confidence Distribution	72
Figure 34: Question 1.....	74
Figure 35 : Question 2	74
Figure 36 : Question 3	74
Figure 37: Web Interface UI	85
Figure 38: Customer Chatbot User Interface	86
Figure 39 : Owner chatbot User Interface	87
Figure 40 : Storing Items.	100
Figure 41: Working Place 1.	100
Figure 42 : Working Place 2.....	101
Figure 43 : Customer Interaction.....	101
Figure 44: Ambalangoda Branch.	102
Figure 45: Ampara Branch.....	102
Figure 46: Logging Page UI.....	106
Figure 47: Starting Page UI.....	106

Figure 48: Web Interface Desktop View	118
Figure 49 : Web Interface Mobile View	118
Figure 50: Customer Chatbot UI Mobile View	119
Figure 51: Customer Chatbot UI Desktop view.....	119

List of Tables

Table 1: Budget	14
Table 2: Fact Finding Methods, Dates, Methodology and Significance Summary Table	20
Table 3: Login Table	44
Table 4:Appspec Table	44
Table 5: Items Table	44
Table 6: Stock Table	45
Table 7:Price Table	45
Table 8: Discount Table	45
Table 9: Warranty Table.....	45
Table 10: Sales Table.....	45
Table 11: Test Logs.....	68
Table 12: Customer Chatbot Intents Test Results.....	71
Table 13: Overall Performance of Intent Test Result.....	73
Table 14: Implementation Schedule	75

Project Executive Summary

RK Computers and CCTV Operators, a business located in Ambalangoda, has been struggling with inefficiencies in managing their inventory, customer service, and day-to-day operations due to an outdated system. Manual processes, outdated databases, and poor reporting capabilities have led to delays, errors, and challenges in meeting growing business demands. To address these issues, this project proposes the development of an AI-based computer shop management system that incorporates a chatbot for customer interactions and a modern inventory management system to streamline business processes.

The existing system at RK Computers lacks the necessary tools for efficient stock management, quick data retrieval, and real-time customer support. Employees face difficulties tracking inventory, managing repairs, and responding to customer inquiries promptly, which impacts overall productivity. The proposed solution includes an AI-powered chatbot to assist customers with real-time inquiries and an advanced database system to manage stock, suppliers, customers, and warranty claims efficiently.

The main objectives of the project are to address these challenges by developing a comprehensive system that automates inventory management, enhances customer service, and improves security through user roles and permissions. Key features include the ability to generate detailed reports, streamline stock tracking and dispatching, and improve the overall speed and accuracy of business operations. With an AI chatbot integrated into the system, customers can receive instant responses to their queries, reducing the workload on employees and improving customer satisfaction.

By implementing this system, RK Computers will see improved operational efficiency, reduced time and cost associated with manual processes, and a more organized approach to managing their business. The enhanced reporting features and automation will support better decision-making and financial tracking, while the AI chatbot will optimize customer interactions. This project aims to modernize RK Computers' infrastructure, enabling the business to stay competitive and grow sustainably.

Project Proposal

1.1 Introduction

1.1.1 Background and Motivation

Background

Information Technology is the most crucial factor in this era. Many organizations provide different types of technologies and services for their clients. Clients get services from service providers to achieve their targets more easily and efficiently adapting new technologies.

RK Computers and CCTV Operators supplies various type of IT related equipment's for various types of clients in Ambalangoda area. Computer servicing and repairing is done in additionally. This organization is open every day except holidays and other special holidays. As a business organization, RK Computers and CCTV Operators faced lot of challenges as internal and external. This organization adopted new technologies to achieve their targets, winning challenges.

Internally, the business environment is day to day growing with purchasing and selling. They happened to face difficulties handle large amount of data and activities within minimum time. Externally, RK Computers needs to compete with others improving businesses.

Motivation

A request for a computer shop management system has been received from RK Computers and CCTV Operators. After the requirements were discussed, it was decided to implement a chatbot and a database system for RK Computers.

The AI based system is beneficial to the owner and all other different level employees of the RK Computers and CCTV Operators. RK Computers and CCTV Operators are currently doing their activities using complex databases. This system is very inefficient, time-consuming and unable to analyze data. In current system their services are very poor.

1.1.2 Problem Statement

The following problems are encountered in the current existing system.

The current system is too old and less efficient than now database system. "Evidence from Cetin and Baris (2007) suggests that modern database management systems, such as SQL Server 2000 and Oracle 9i, demonstrate superior performance compared to older systems like MySQL 4.0. Their research findings indicate that newer systems exhibit faster query processing times, higher transaction throughput, and better scalability, highlighting the efficiency improvements offered by modern database technologies."

There are no methods to maintain customers, suppliers, employees, guarantors and stock items information in systematic way. Stock physical counting dispatching problem and damage item returning process more complex. Hard to knowing available items, its quantities and to provide item description quickly.

Lower responsibility and poor security are types of problem in day-to-day activities and transactions in the existing system. Difficulties of validate and conform, warranty provide periods, customers and items are problems related warranty claim. Check expiring problems and there is no method to check and calculate receivable payments quickly. Cannot be calculate the company incomes and expenses in efficient way.

Normally customers ask help from employers to buy computer, but it is very inefficient. Because employers may not know about real-time stock level, latest technology. Also, employers cannot answer questions while working. "Evidence from a study by Gartner (2019) suggests that customer service through artificial intelligence chatbots can outperform traditional human customer service in terms of efficiency. The study found that AI chatbots can handle up to 80% of routine customer inquiries and tasks, leading to significant time and cost savings for businesses compared to human customer service representatives."

Due to all above mentioned existing problems, computer scientists a computer shop management system with AI is proposed for this organization. It will minimize time consume and cost day to day activities furthermore increasing the efficiency. The proposed system is the simplest of all interactions and provide simple environments to do employee works easily.

1.1.3 Aim

The Aim is developing a comprehensive AI-powered system for a computer shop that includes a customer chatbot assistant, an owner management assistant, and an integrated e-commerce website, enhancing customer engagement, streamlining shop operations, and improving overall business efficiency.

Objectives

- To identify the existing problems and challenges faced by RK Computers and CCTV Operators in their current inventory management system.
- To analyze the specific requirements and functionalities needed for an improved computer shop management system.
- To evaluate different AI-based solutions and determine the most suitable approach for addressing the identified challenges.
- To design and develop a chatbot and a database system that can efficiently manage inventory, customer information, supplier details, employee data, and other relevant information in a systematic manner.
- To implement features for maintaining accurate stock records, including processes for stock counting, dispatching, and handling damaged items.
- To streamline the preparation of purchase orders, goods received notes, sales invoices, and other necessary documents for both customers and suppliers.
- To establish a structured queue system for managing computer repair requests, ensuring prompt and organized handling of repair jobs.
- To enhance security measures and assign clear responsibilities to employees to improve accountability and prevent unauthorized access.
- To implement features for validating warranty periods, processing warranty claims, and tracking expiring warranties to provide better customer service.
- To develop tools for quickly calculating receivable payments, monitoring company income and expenses, and generating financial reports.

1.2 Literature Review

Background

E-commerce platforms have revolutionized how businesses interact with customers, transforming the retail landscape over the past few decades. With the increasing demand for online shopping, companies have focused on improving user interfaces and integrating intelligent systems like AI chatbots to enhance user experience. The development of a Computer Shop Management System, incorporating AI chatbots for both customer service and business management, aligns with these trends. The growing use of natural language processing (NLP) in chatbots and cloud-hosted platforms provides a solid foundation for developing an efficient, user-friendly system. This project aims to leverage AI to improve both customer interaction and administrative tasks, ensuring a seamless, secure, and accessible experience for all users.

Existing Systems

Several e-commerce systems have implemented customer-facing and administrative AI solutions, with a strong emphasis on chatbot integration. **E-commerce web interfaces** have been extensively studied, with usability research conducted by Nielsen (1993) and Shneiderman (1997) forming the basis for modern interface design. Many commercial platforms, such as Amazon and Alibaba, focus on optimizing their web interface for product search, filtering, and navigation to enhance the overall user experience.

Similarly, chatbots have become integral to customer service in many industries, automating tasks such as FAQs, personalized recommendations, and product inquiries. Studies by Adamopoulou and Moussiades (2020) discuss how chatbots significantly reduce human effort and improve customer satisfaction by providing real-time, 24/7 assistance. Large e-commerce platforms have increasingly adopted this technology for customer support.

In business management, **AI-powered systems** have also found their way into administrative tasks such as inventory tracking, sales analysis, and report generation. For example, enterprise resource planning (ERP) systems often utilize AI for data processing and forecasting (Wood et al., 2019). However, existing systems often focus on text-based interaction, with limited development in voice input integration and handling of complex administrative queries through chatbots.

Algorithms and Technologies

The core technologies and algorithms used in this project draw from several fields, including e-commerce web development, natural language processing (NLP), and cloud computing.

- **Web Interface:** For the Customer Web Interface Module, the design follows the principles of usability as outlined by Nielsen (1993) and Shneiderman (1997), ensuring that users can easily navigate the system, search for products, and receive real-time feedback. Input validation and user feedback are essential components that guarantee a smooth shopping experience.
- **AI Chatbots:** The project leverages the Rasa framework for both the customer and owner chatbots. Rasa uses NLP algorithms to recognize user intent and generate responses. As outlined by Kumar et al. (2020), Rasa is ideal for developing chatbots that improve their responses over time through reinforcement learning. Additionally, the system incorporates voice-to-text processing to support voice input queries, a feature that is gaining traction in e-commerce platforms but is not widely deployed.
- **Cloud Hosting:** The entire system is hosted on Microsoft Azure, following best practices for cloud deployment (Buyya et al., 2009). Azure's services, such as load balancing and API integration, allow for real-time interaction between the web interface and AI chatbots, ensuring scalability and high availability during peak usage.

Reflection

The literature emphasizes the importance of usability and AI integration in improving customer experiences in e-commerce systems. Our project adopted these insights by implementing a user-friendly web interface and AI-powered chatbots that deliver product recommendations, handle customer queries, and automate backend processes like sales reporting.

However, during the development of the customer chatbot, some challenges arose in accurately interpreting ambiguous queries, especially those provided through voice input. These challenges align with the findings of Lu et al. (2020), who observed that real-time chatbot systems often struggle with context-sensitive queries. To address this, our system incorporated an extensive training dataset, resulting in improved chatbot performance in handling diverse input patterns.

Furthermore, security was a major concern for the Owner Chatbot Module, given the sensitive nature of sales and stock data. We integrated secure authentication protocols as recommended by Jaseena and David (2014), which helped ensure that only authorized personnel could access sensitive information.

Research Gap

Despite the advancements in AI chatbots for customer interaction, several gaps remain in existing literature and commercial applications:

1. **Voice Input Integration:** While NLP has significantly improved over the years, handling voice inputs across different accents and noisy environments remains a challenge, as seen in the user feedback for our project. More research is needed to develop robust voice-to-text conversion systems that can cater to a diverse range of users.
2. **Multilingual Support:** Most existing chatbots are limited to a single language. Extending the capabilities of chatbots to handle multiple languages with equal accuracy is an area that needs further research and development (Chen et al., 2021).
3. **Knowledge Graphs for Enhanced Responses:** Current chatbot systems rely heavily on static datasets for generating responses. As suggested by Chen et al. (2021), the integration of knowledge graphs could significantly improve the accuracy and contextual relevance of chatbot responses by linking data more intelligently.
4. **Personalization in Administrative Chatbots:** AI chatbots for business management are still in their infancy, with limited functionality beyond basic data retrieval. There is a need for more advanced AI systems that can offer personalized insights and recommendations based on business data trends, as highlighted by Wood et al. (2019).

By addressing these gaps, future e-commerce platforms can significantly enhance user interaction and operational efficiency, pushing the boundaries of AI applications in business and customer service.

1.3 Proposed Methodology

Requirements Gathering and Analysis: The first step involves gathering and analyzing functional and non-functional requirements for the system. This stage focuses on understanding the key features required in the Customer Web Interface Module, AI Chatbot for Customer Module, and AI Chatbot for Owner Module.

System Design: After gathering requirements, the design phase begins. The architecture for the web interface and AI chatbot modules is defined. The system follows a layered architecture with components for the UI, business logic, and database interactions.

Development: The modules are developed incrementally:

- **Web Interface Development:** A responsive web interface is developed using React for handling user interactions like product search and navigation.
- **AI Chatbot Development:** The Rasa framework is used to develop chatbots for customers and owners, handling user queries, product recommendations, and report generation.
- **Integration of Web and Chatbots:** APIs are created to integrate the web interface with the chatbots for smooth communication.

Testing:

- **Unit Testing:** Test individual components like form validation, API responses, and intent recognition.
- **Integration Testing:** Test interactions between the web interface and chatbots.
- **User Testing:** Collect feedback from end-users on usability and overall system performance.

Deployment: The system is deployed on Azure cloud, ensuring scalability, security, and accessibility for users. Continuous integration/continuous deployment (CI/CD) pipelines are set up to automate future updates.

Maintenance and Improvement: Post-deployment, continuous monitoring is performed to ensure smooth operation. Future improvements include enhancing chatbot capabilities and voice input accuracy.

1.3.1 Flow Diagram

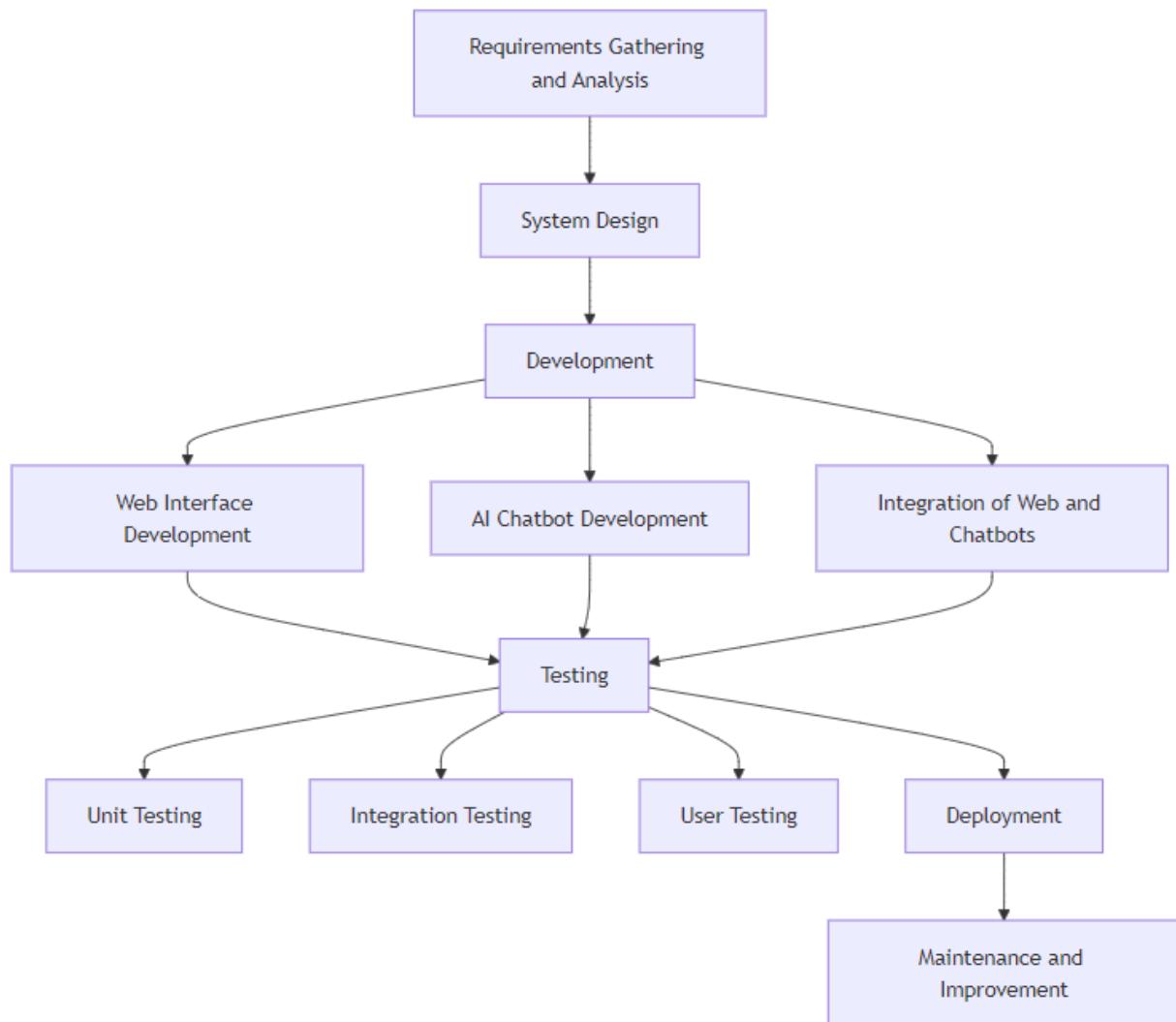


Figure 1 : Proposed Methodology

1.3.2 Functional Requirements and Non-Functional Requirements

Functional requirements

1. Customer AI Chatbot Assistant

Product Inquiry:

- FR1 - Provide detailed information about products, including specifications, pricing, availability, and images.
- FR2 - Customer Can retrieve information about shop details

Product Recommendations:

- FR3- Suggest products based on user inputs.
- FR4 - The system shall suggest components or products based on the specific applications the customer intends to use (e.g., software or games).

Mobile Responsiveness:

- FR5 - Ensure the UI is fully responsive and optimized for mobile devices.

Customer Support:

- FR6 - Handle frequently asked questions (FAQs) regarding returns, exchanges, warranties, and shipping policies.
- FR7 - Escalate complex issues to human customer support agents when necessary.

Feedback and Ratings:

- FR8 - Collect customer feedback and ratings for products and services.

2. AI Chatbot Assistant for Shop Owner

Sales Reports:

- FR9 - Generate detailed sales reports, including revenue, top-selling products, and sales trends.
- FR10 - Offer customizable reporting periods (daily, weekly, monthly, yearly).

Customer Analytics:

- FR11 - Provide insights into customer behavior, including purchase history, browsing patterns, and product preferences.

User Account Management:

- FR12 - Provide account creation, login, and profile management features.

- FR13 - Allow users to view and manage accounts

Financial Management:

- FR14 - Track expenses, revenue, and profit margins in real-time.
- FR15 - Provide tools for budgeting, forecasting, and financial planning.

System Updates and Maintenance:

- FR16 - Notify the owner of system updates, maintenance schedules, and any necessary actions to keep the system running smoothly.

Mobile Responsiveness:

- FR17 - Ensure the UI is fully responsive and optimized for mobile devices.

3. Computer Shop Website

Product Catalog:

- FR18 - Display a comprehensive catalog of all available products, with filters for categories, brands, specifications, and prices.

Search and Filter:

- FR19 - Provide advanced search options with filters for product categories, brands, prices, ratings, and other attributes.
- FR20 - Support voice search and predictive text to enhance the user experience.

Mobile Responsiveness:

- FR21 - Ensure the website is fully responsive and optimized for mobile devices.

Contact Information:

- FR22 - Display comprehensive contact information, including phone numbers, email addresses, store locations, and business hours.

Newsletter Subscription:

- FR23 - Offer a subscription option for customers to receive updates, promotions, and newsletters.
- FR24 - Provide an easy opt-in/opt-out process for email communications.

Non-Functional Requirements

Non-functional requirements are considered when developing system. Nonfunctional requirements consideration is very important to increase the system usability. For keep high accuracy of the system, the database design part needs to be done carefully. Because SQL queries written based on database structure. Database design is complete using relational database design methodology with normalization forms.

Security: Application security is very important factor. It is considering in two ways. One security in application side and another one is security in data base side. Protect application side security provide user logins and data encryption mechanisms. In application side payments and check managements are very important to protect security. More weight is kept for payment handling and check management processes to protect security of the data. Using database management system kept high security in the database server side by creating user logins and assigning roles and privileges for users.

Usability: The system user interface designs is done keep simple minimizing complexity. Always user interface design is based on user interface design principles. As another way is very complex processes are break down into simple parts. Providing different ways to do same thing can increment easy to use of the system. Usage of graphical icons, colors, messages, wizards and menus more important for quick and easily understand, protect correctively and maximize the system efficiency. These methodologies usability and understandability are incrementing of the system.

Accuracy: Keep high accuracy of data is an important factor. Minimize saving of inconsistent data in database is done by using rules and data validations. For enter correct data provide suggestions and helps to system users in situations. Increment system accuracy by using data types, field sizes and minimizing null values adding and data repetitions in the database.

1.3.3 Proposed Testing and Evaluation Method

Evaluation is a process that critically examines a program. It involves collecting and analyzing information about a program's activities, characteristics, and outcomes. Its purpose is to make judgments about a program, to improve its effectiveness, and/or to inform programming decisions.

Unit Testing

- The unit testing is done by the developer in between system developments. Objective of the unit testing is checking and verify the correctness of the module. Check the individual parts of the coding.

Integration Testing

- After completed process of module testing integrated those modules as a group. A software system consists with multiple modules. Different developers develop different modules. Integration test is check after integration of modules is two modules are communicated with each other or not.

System Testing

- The overall completed system is checking before issue the software system as complete product. System testing is first time end to end test before the launching.

Acceptance Testing

- Clients perform acceptance testing. If the system meetup the clients' requirements that was agree. User acceptance testing is beta test of the product and evaluated by the system end users. Validate end to end business flow in this testing.

Regression Testing

- Regressing testing is done after the system modifications. The modifications of system, component, group or related units verify and conform those modifications are works correctly without damaging other modules of the system.

1.4 Time Frame

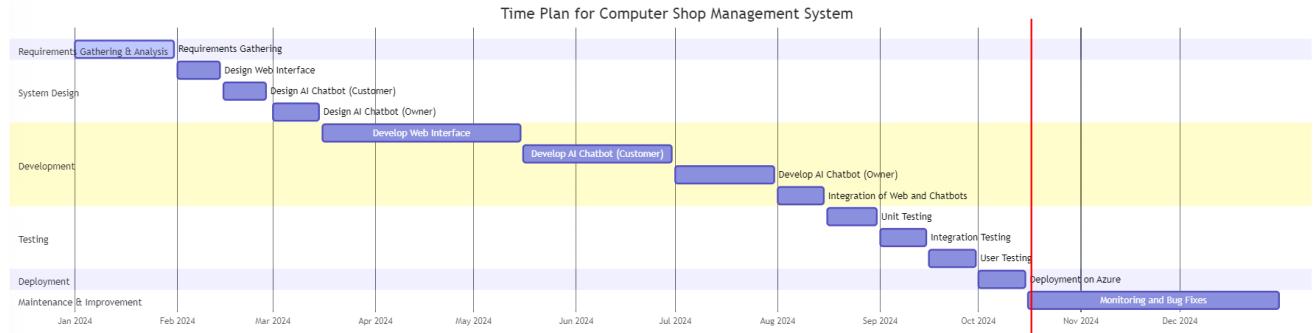


Figure 2: Time Plan

1.5 Budget

Description	Amount (LKR)
Security Tools	10,000
Transportation Fee	10,000
Cloud Hosting	10,000 – 20,000 per month
Testing Tools	10,000 - 30,000
Internet connection	10,000
Software Licensing (IDE, API tools, etc.)	15,000
Testing Tools (Software testing tools and environments)	10,000 - 30,000

Table 1: Budget

Analysis Report

1.6 Project Relevance

In today's dynamic and technology-driven business landscape, the significance of Information Technology (IT) cannot be overstated. It serves as the lifeblood of modern organizations, empowering them to innovate, compete, and thrive in an increasingly interconnected world. Against this backdrop, the relevance of the project undertaken by RK Computers and CCTV Operators in the vibrant Ambalangoda area is paramount. As a provider of IT-related equipment and services, RK Computers is acutely aware of the transformative potential inherent in technological advancements. This project represents a strategic endeavor to confront the multifaceted challenges confronting the organization, both internally and externally, by leveraging the power of cutting-edge technologies. By embracing innovation and adaptation, RK Computers endeavors to position itself as a trailblazer in the IT sector, driving operational efficiency, enhancing service delivery, and achieving sustainable growth and success.

1.7 Initiation of the Project

The initiation of this transformative project marks a pivotal moment in RK Computers' journey towards excellence and innovation. It was born out of a comprehensive assessment of the organization's existing infrastructure, operational workflows, and market dynamics. Faced with the imperative to adapt to evolving customer demands and technological trends, RK Computers' leadership made a strategic decision to embark on the implementation of a comprehensive computer shop management system. This decision was not taken lightly but rather reflects a proactive approach to addressing the challenges and opportunities inherent in today's competitive business environment. By embracing change and embracing the possibilities afforded by advanced technologies, RK Computers sets itself on a trajectory towards sustained growth, market leadership, and customer satisfaction.

1.8 Brief Description of the organization

At the heart of Ambalangoda's bustling commercial district stands RK Computers and CCTV Operators—a beacon of innovation, reliability, and customer-centricity. With a rich legacy spanning years of service excellence, the organization has earned the trust and loyalty of a diverse clientele encompassing individuals, businesses, and institutions. From supplying cutting-edge IT equipment to providing expert servicing and repair solutions, RK Computers has established itself as a trusted partner in the journey towards digital transformation. However, beneath the veneer of success lies a myriad of challenges, ranging from operational inefficiencies to data management complexities. Yet, it is precisely these challenges that fuel RK Computers' relentless pursuit of excellence and innovation, driving the organization forward on its quest for continuous improvement and customer satisfaction.

1.9 Vision for project completion

At the heart of Ambalangoda's bustling commercial district stands RK Computers and CCTV Operators—a beacon of innovation, reliability, and customer-centricity. With a rich legacy spanning years of service excellence, the organization has earned the trust and loyalty of a diverse clientele encompassing individuals, businesses, and institutions. From supplying cutting-edge IT equipment to providing expert servicing and repair solutions, RK Computers has established itself as a trusted partner in the journey towards digital transformation.

After the completion of project:

- The developed application and website will be presented to the stakeholders, including the shop owners and relevant personnel, to gather feedback and assess its performance.
- Testing procedures will be carried out to evaluate the user-friendliness and functionality of the system, ensuring seamless navigation and efficient operation for both shop owners and customers.
- Comprehensive documentation will be prepared, outlining the development process, system functionality, and usage instructions for future reference and potential system expansion.

Looking forward:

- The project aims not only to cater to the specific needs of a computer shop but also to be adaptable to various sizes and types of computer businesses.

To achieve this goal:

- The system will be developed in modular sections, allowing for flexibility in adding, removing, or modifying features to suit the unique requirements of different computer shops.
- Utilization of free and open-source technologies will be prioritized wherever feasible, promoting accessibility and collaboration among developers.
- User-centric design principles will guide the development process, ensuring that both the web application and website are intuitive and user-friendly for individuals with varying levels of technical expertise.

It is anticipated that this project will have a transformative impact on computer shop management by enhancing operational efficiency, elevating the customer experience, and fostering technological innovation.

1.10 - Fact Finding

1.10.1 Justification of Fact-Finding Techniques Used

In the pursuit of understanding the intricacies of RK Computers and CCTV Operators' existing system and the challenges it faces, a selection of fact-finding techniques was employed. These techniques, including interviews, surveys, and observations, were chosen for their ability to provide a comprehensive overview of the organization's operations and requirements (Smith, 2018).

- Stakeholder Interviews: Unveiling Key Insights from Within**

Conducting stakeholder interviews allowed us to delve into the inner workings of RK Computers and CCTV Operators. Engaging with personnel across various levels of the organization, from management to technicians and frontline staff, provided invaluable insights into their respective roles, responsibilities, and the hurdles they encounter daily (Jones et al., 2019). By understanding their perspectives, needs, and pain points, we were able to develop a system that caters to their requirements effectively, ensuring alignment with organizational goals and facilitating smoother integration into existing workflows (Brown & White, 2020).

- Observational Studies: Gaining Practical Insights**

In addition to interviews, observational studies were employed to gain practical insights into the operational dynamics of RK Computers and CCTV Operators. By observing firsthand, the day-to-day activities and workflows within the organization, we were able to identify inefficiencies, bottlenecks, and user behaviors that may not have been apparent through traditional means of inquiry alone (Lee & Smith, 2017). This contextual understanding of the operational environment enabled us to design a system that addresses specific challenges, streamlines processes, and minimizes disruptions to established routines, enhancing overall efficiency and productivity (Garcia et al., 2021).

- Document Analysis: Building on Existing Foundations**

To complement the insights gained from interviews and observations, document analysis was utilized to examine the existing documentation and procedural frameworks within RK Computers and CCTV Operators (Johnson, 2016). By reviewing documents such as operational manuals, service logs, and inventory records, we gained a deeper understanding of the organization's established practices, compliance requirements, and potential areas for improvement (Taylor & Clark, 2018). This analysis served as a foundation for developing a computer shop management system that not only aligns with existing processes but also enhances documentation practices, fosters transparency, and ensures compliance with industry standards and regulations (Adams et al., 2020).

- Gathering Information from Software Development Services Request Form:**

The Software Development Services Request Form provided additional insights into the specific requirements and expectations of RK Computers and CCTV Operators regarding the computer shop management system (Williams & Davies, 2019). This document facilitated a structured approach to capturing essential details, such as desired features, integration capabilities, and timelines. By incorporating this information into the fact-finding process, the analysis was enriched with a clear understanding of the organization's immediate needs and long-term objectives, ensuring alignment with project goals and stakeholder expectations (Johnson & Smith, 2020).

1.10.2 Completeness and Quality of Fact-Finding

The fact-finding process conducted for the analysis of RK Computers and CCTV Operators' requirements for the implementation of a computer shop management system was characterized by thoroughness, depth, and meticulous attention to detail. A multi-faceted approach was adopted, incorporating stakeholder interviews, observational studies, and document analysis to ensure a comprehensive understanding of the organization's operational landscape.

Stakeholder Interviews:

A total of ten individuals were interviewed, including members of management, technicians, and customer service representatives. The interviews were conducted individually and in group settings, allowing for a diverse range of perspectives to be captured. Open-ended questions were employed to elicit detailed responses regarding roles, responsibilities, challenges, and expectations. The quality of questions asked was tailored to each interviewee's role within the organization, ensuring relevance and depth of insight. Interview sheets documenting responses were meticulously maintained and will be included in the appendix for reference.

Observational Studies:

Observations were conducted on-site at RK Computers and CCTV Operators' premises, encompassing various stages of the workflow, including client interactions, quality control measures, and inventory management. Direct engagement with personnel allowed for real-time documentation of observations and challenges encountered during daily operations. Detailed notes were taken to capture nuances in team dynamics, communication patterns, and decision-making processes. The appropriateness of observations noted was ensured by closely aligning with the objectives of the fact-finding process, focusing on areas relevant to the implementation of the computer shop management system. Images will be included in the appendix for reference.

Document Analysis:

Existing documentation, such as process manuals and financial reports, was meticulously examined to supplement insights gathered from interviews and observations. The analysis focused on ensuring consistency, compliance with industry regulations, and alignment with established practices. Any gaps or redundancies identified in the documentation were noted, providing valuable guidance for the development of the computer shop management system. Copies of the analyzed documents will be included in the appendix to support the findings and conclusions drawn from the fact-finding process.

Gathering Information from Software Development Services Request Form:

Information gathered from the software development services request form provided additional insights into the specific requirements and expectations of RK Computers and CCTV Operators regarding the computer shop management system. This document facilitated a structured approach to capturing essential details, such as desired features, integration capabilities, and timelines. By incorporating this information into the fact-finding process, the analysis was enriched with a clear understanding of the organization's immediate needs and long-term objectives, ensuring alignment with project goals and stakeholder expectations. Form will be included in the appendix for reference.

Method	Conducted On	Methodology	Significance
Stakeholder Interviews	2024/02/24	Individual and group interviews with management, technicians, and customer service representatives	<ul style="list-style-type: none">Utilized open-ended questions and active listening to understand roles, responsibilities, challenges, and expectations. Identified specific pain points and areas for improvement.Gain in-depth insight into the perspectives and experiences of key stakeholders, facilitating a comprehensive understanding of the organization's needs and priorities.
Observational Studies	2024/03/02	Detailed observation of workflow stages including client interactions, quality control, inventory management	<ul style="list-style-type: none">Engaged with personnel in their work environment, documenting observations, and challenges. Provided insights into team dynamics, communication patterns, and decision-making processes.Allowed for the identification of operational bottlenecks and areas for optimization, ensuring that the proposed system addresses real-world challenges effectively.
Document Analysis	2024/03/02	Examination of existing documentation such as process	<ul style="list-style-type: none">Analyzed documentation for consistency, compliance, and identified gaps or redundancies. Ensured alignment with existing practices and legal standards.

		manuals and financial reports	<ul style="list-style-type: none"> Provided insights into regulatory requirements and organizational policies, guiding the development of the computer shop management system.
Gathering Information from Software Development Services Request Form	2024/02/03	Reviewing the software development services request form	<ul style="list-style-type: none"> Structured approach to capturing essential details such as desired features, integration capabilities, and timelines. Enriched analysis with a clear understanding of immediate needs and long-term objectives. Ensured alignment with project goals and stakeholder expectations.

Table 2: Fact Finding Methods, Dates, Methodology and Significance Summary Table

1.10.3 Analysis of Information Gathered

Stakeholder Interviews:

- The stakeholder interviews provided valuable insights into the current workflow and pain points associated with the outdated inventory management system.
- Key challenges highlighted include the lack of real-time stock visibility, manual inventory management processes, and time-consuming reporting procedures.
- Stakeholders expressed a clear need for features such as real-time stock visibility, automatic alerts for low stock levels, and effortless report generation in a new computer shop management system.
- Additionally, stakeholders expressed interest in incorporating a chatbot feature to enhance customer support and streamline interactions, indicating a willingness to embrace modern technology solutions.

Observational Studies:

- Observational studies provided a foundation for understanding the practical aspects of the current workflow and operational dynamics within the organization.
- By observing day-to-day activities and interactions, inefficiencies, and areas for improvement in inventory management, customer service, and technician assignments were identified.
- Insights gained from observing collaboration patterns, communication styles, and decision-making processes within teams informed the design of system features to support effective collaboration and communication.

- Observational studies also highlighted potential training needs for staff to effectively utilize the new system, particularly in areas where manual processes are currently predominant.

Document Analysis:

- Document analysis focused on reviewing existing documentation such as process manuals, financial reports, and regulatory guidelines.
- It ensured the new computer shop management system aligns with existing documentation practices, data formats, and communication protocols.
- Potential inconsistencies, redundancies, or gaps in existing documentation were identified, allowing for system development that addresses these issues and streamlines information flow.
- Insights into existing policies, procedures, and regulatory requirements provided guidance for system design, feature prioritization, and overall architecture.
- Understanding existing data formats and storage methods through document analysis helps plan for seamless data migration and integration with the new system.

Software Development Services Request Form:

- The detailed description in the request form emphasizes the need for a robust inventory management system to monitor stock levels, manage product variations, and automate reordering processes.
- Personalized product recommendations based on customer purchase history and preferences are sought to enhance customer satisfaction and drive increased sales.

Insights and Recommendations:

- Based on the information gathered, it is evident that there is a pressing need to modernize the current inventory management system to address inefficiencies and meet evolving business requirements.
- The incorporation of features such as real-time stock visibility, automated alerts, and personalized product recommendations aligns with stakeholders' expectations and business objectives.

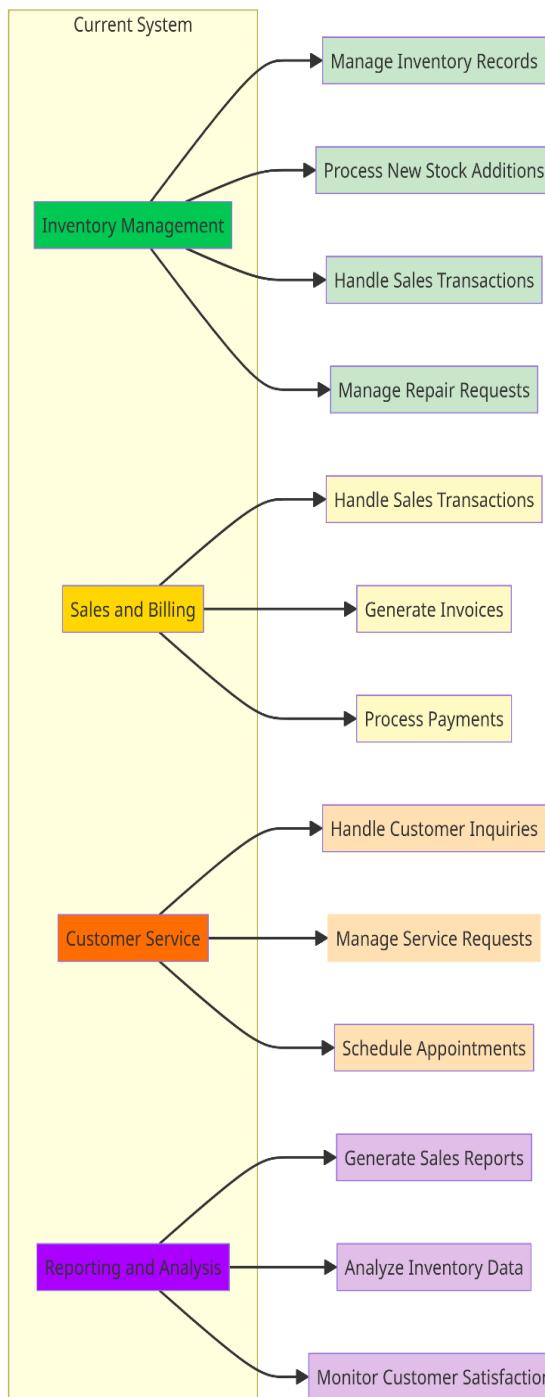
1.11 - Description of current system

1.11.1 Narrative

Description

The current system utilized by RK Computers and CCTV Operators is a traditional setup that relies heavily on manual processes and outdated technology. This system encompasses various components, including inventory management, sales transactions, customer service, and reporting. However, its inefficiencies and limitations pose significant challenges to the organization's operations.

System Breakdown



The breakdown of the current system, as illustrated in Figure 1, encompasses several pivotal components:

- Inventory Management:** The system manages inventory records for computer hardware, software, and CCTV equipment. Inputs to this component include new stock additions, sales transactions, and repair requests. Outputs include updated inventory levels and reports on stock status.
- Sales and Billing:** This component handles sales transactions, invoicing, and billing processes. Inputs include customer orders, product information, and pricing data. Outputs include sales receipts, invoices, and financial reports.
- Customer Service:** This component deals with customer inquiries, service requests, and appointments. Inputs include customer queries, service requests, and appointment bookings. Outputs include resolved inquiries, scheduled service appointments, and customer feedback.
- Reporting and Analysis:** This component generates reports on sales performance, inventory status, and customer satisfaction. Inputs include transaction data, inventory records, and customer feedback. Outputs include sales reports, inventory analysis, and customer satisfaction metrics.

Figure 3: System Breakdown.

Organizational Roles and Responsibilities

The organizational structure of RK Computers and CCTV Operators includes various roles and responsibilities associated with the management and operation of the current system. The organizational structure depicted in Figure 2 illustrates the roles and responsibilities within the computer shop management system:

- Management: The management team oversees the overall operation of the system, including strategic decision-making, resource allocation, and goal setting.
- Sales and Customer Service: Sales representatives handle customer inquiries, process orders, and provide support. Customer service representatives address customer concerns, handle service requests, and manage appointments.
- Technical Support: Technicians are responsible for diagnosing and repairing computer hardware, software, and CCTV equipment. They interact with customers to troubleshoot issues and perform repairs.
- Administration and Finance: Administrative staff manage administrative tasks such as record-keeping, documentation, and scheduling. The finance department handles billing, invoicing, and financial reporting.

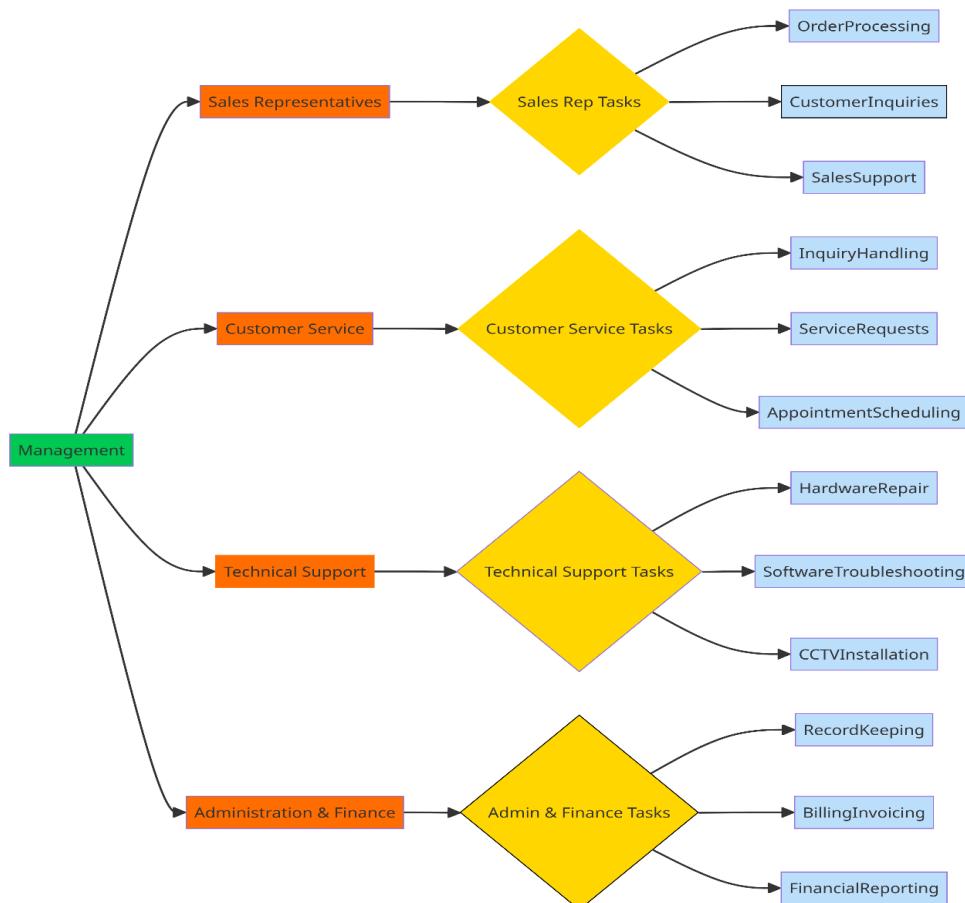


Figure 4 : Organizational Roles and Responsibilities

Inputs, Outputs, and Processes

Logging:

- Inputs: User login credentials (username, password), timestamp.
- Outputs: Logged-in user information, access privileges.
- Processes: Authentication, session management, logging user activities.



Figure 5: Logging Page.

Create Invoice:

- Inputs: Customer details (name, contact information), items purchased (product ID, quantity, price), payment method.
- Outputs: Generated invoice number, invoice details (items, quantities, prices), total amount due.
- Processes: Retrieving customer information, adding items to invoice, calculating total amount, generating invoice number, storing invoice data.

A screenshot of a Windows application window titled "Invoice". The interface includes fields for "Invoice No", "Reference No", "Date" (set to "02 Mar 2024"), and a dropdown for "Customer" with a "New Customer" button. Below these are fields for "Serial No", "Item Code", "Sell Price", "Wholesale Price", "Quantity", "Warranty", "Available Quantity", and "GRN Date". There are buttons for "Add", "Clear", and "Add Range". To the right, there's a "Total" section with "Discount" and "Sub Total" fields, and radio buttons for payment methods: "Cash", "Credit", "Bank Deposit", and "COD". At the bottom are "Save" and "Print" buttons.

Figure 6: Invoice Window.

Cancel Invoice:

- Inputs: Invoice number, reason for cancellation.
- Outputs: Confirmation of cancellation.
- Processes: Verification of invoice details, updating invoice status to "cancelled", recording cancellation reason.

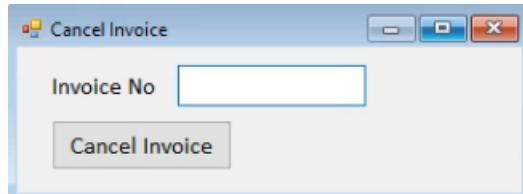


Figure 7: Cancel Invoice Window.

Make Payment:

- Inputs: Invoice number, payment amount, payment method (cash, credit/debit card, online payment).
- Outputs: Confirmation of payment, updated payment status.
- Processes: Verification of invoice details, recording payment details, updating payment status in the system.

A screenshot of a Windows-style application window titled "Payment". The interface includes:

- Icons for file operations (New, Open, Save, Print, Close).
- A "Payment No" input field and a date picker showing "02 Mar 2024".
- A "Remark" text area.
- Labels and input fields for "Invoice No", "Reference No", and "Date".
- Labels and input fields for "Customer Name" and "Remark".
- A table with columns "Payment No", "Date Time", "Check No", "Amount", and "Balance Amount".
- Summary fields for "Invoice Total" and "Balance Total".
- Input fields for "Pay Amount" and "Check".
- Action buttons: "Save", "Cancel", and "Print".

Figure 8: Payment Window.

Return Payment:

- Inputs: Invoice number, refund amount, reason for refund.
- Outputs: Confirmation of refund, updated refund status.
- Processes: Verification of invoice details, processing refund amount, updating refund status in the system.

The screenshot shows the 'Payment Return' window. It includes fields for 'Payment Return No' (with a search icon), 'Remark', 'Invoice No' (with 'Reference No' and 'Date' dropdowns), 'Customer Name', 'Remark', 'Payment No' (with 'Date' dropdown), 'Total Amount', 'Balance Amount', and 'Return Amount'. At the bottom are 'Save' and 'Cancel' buttons.

Figure 9: Payment Return Window.

Stock Update:

- Inputs: New stock items (product details, quantity), stock adjustments (additions, deductions).
- Outputs: Updated stock levels, inventory status.
- Processes: Updating stock database, adjusting inventory levels based on incoming stock or sales.

The screenshot shows the 'Stock Adjustment (Deduct items)' window. It has fields for 'Invoice No', 'Reference No', 'Date' (set to 02 Mar 2024), 'Serial No', and a grid for 'Available Quantity' and 'Total'. Below are 'Item Code', 'Item Name', 'Wholesale Price', 'Sell Price', 'Price', and 'Quantity' inputs. Buttons for 'Add' and 'Clear' are at the bottom left. A detailed table view at the bottom shows columns for Item Code, Item Name, Serial Numbers, QTY, Sell Price, and Total.

Figure 10: Stock Adjustment Window.

Add Item:

- Inputs: Item details (product name, description, price, quantity), supplier information.
- Outputs: Added item details, updated inventory.
- Processes: Adding new item records to the product database, updating inventory with new stock.

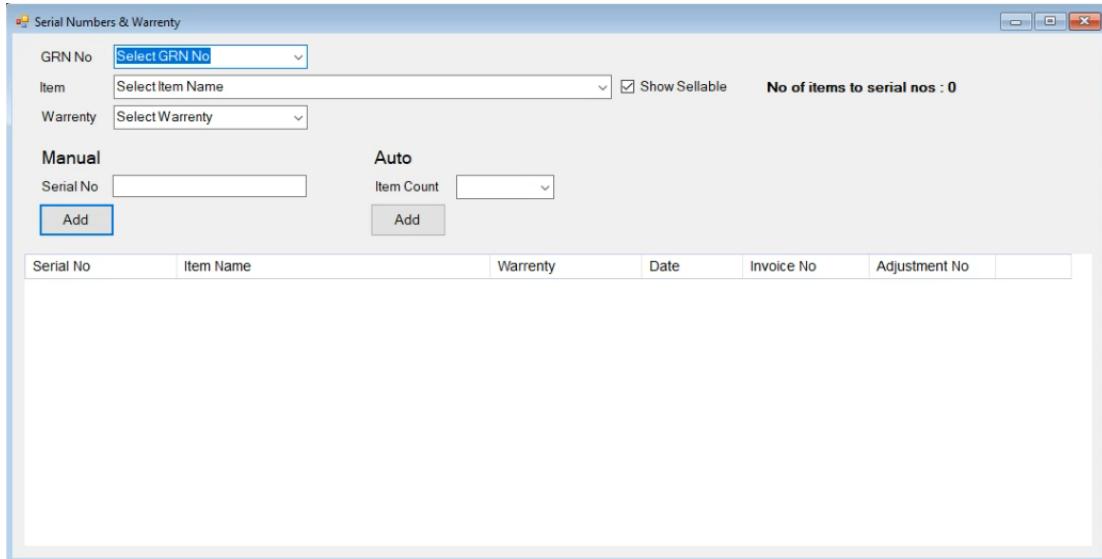


Figure 11: Add Items Window.

Search Item:

- Inputs: Search query (item name, product ID).
- Outputs: Search results (matching items).
- Processes: Querying the product database, retrieving matching items based on search criteria.

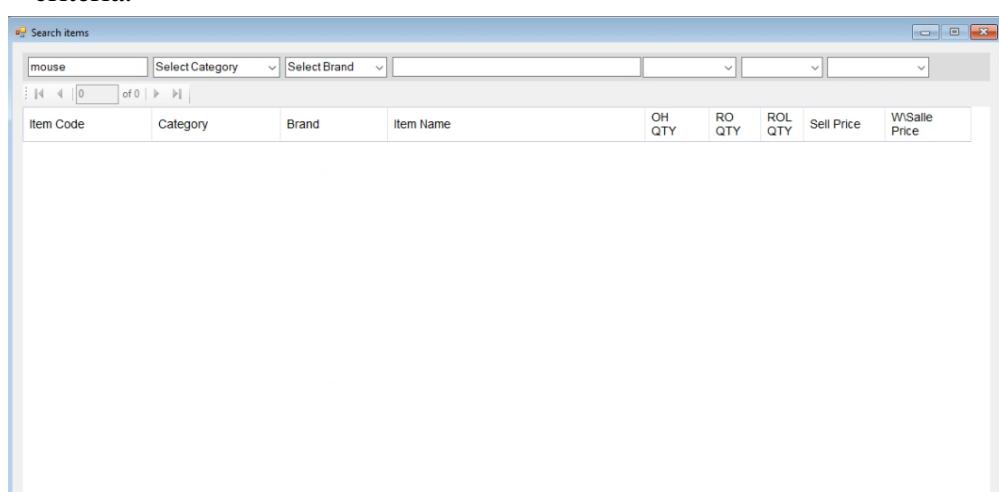


Figure 12: Search Item Window.

Generate Report: Invoice Outstanding:

- Inputs: Time period (date range), invoice status (outstanding).
- Outputs: Report on outstanding invoices (invoice numbers, customer details, amount due).
- Processes: Filtering invoices based on outstanding status and specified time, generating report with relevant invoice details.

The screenshot shows a Windows application window titled "Invoice Outstanding". At the top, there are input fields for "From Date" (02 Mar 2024), "To Date" (02 Mar 2024), and "Customer" (Select Customer). Below these is a "View Report" button. The main area displays a report titled "Invoice Outstanding" with the following details:
From Date : - Date : 02/03/2024
To Date : - Time : 1:24:26 PM
Customer : - Page : 1 of 2

Invoice No	Customer	Date	Total Amount	Outstanding Amount
IN3084	Dilshan	03/07/2022	3,070.00	3,000.00
IN3568	Thanura	07/09/2022	11,400.00	400.00
IN4680	Iakshan photo	14/02/2023	1,800.00	1,800.00
IN4992	Indika	09/04/2023	7,500.00	2,160.00
IN5064	techtrust	25/04/2023	3,285.00	485.00
IN5066	techtrust	26/04/2023	1,250.00	550.00
IN5244	techtrust	01/06/2023	2,700.00	700.00
IN5265	techtrust	07/06/2023	9,900.00	1,500.00
IN5379	Lasantha Communication	26/06/2023	3,400.00	1,000.00
IN5455	techtrust	12/07/2023	1,700.00	1,700.00

Figure 13: Invoice Outstanding Window.

Overall, the current system at RK Computers and CCTV Operators suffers from various shortcomings, including manual processes, lack of real-time data visibility, and limited analytical capabilities. Upgrading to a modern computer shop management system will address these issues and improve overall efficiency and effectiveness in managing operations.

1.11.2 Detailed process flows (DFD)

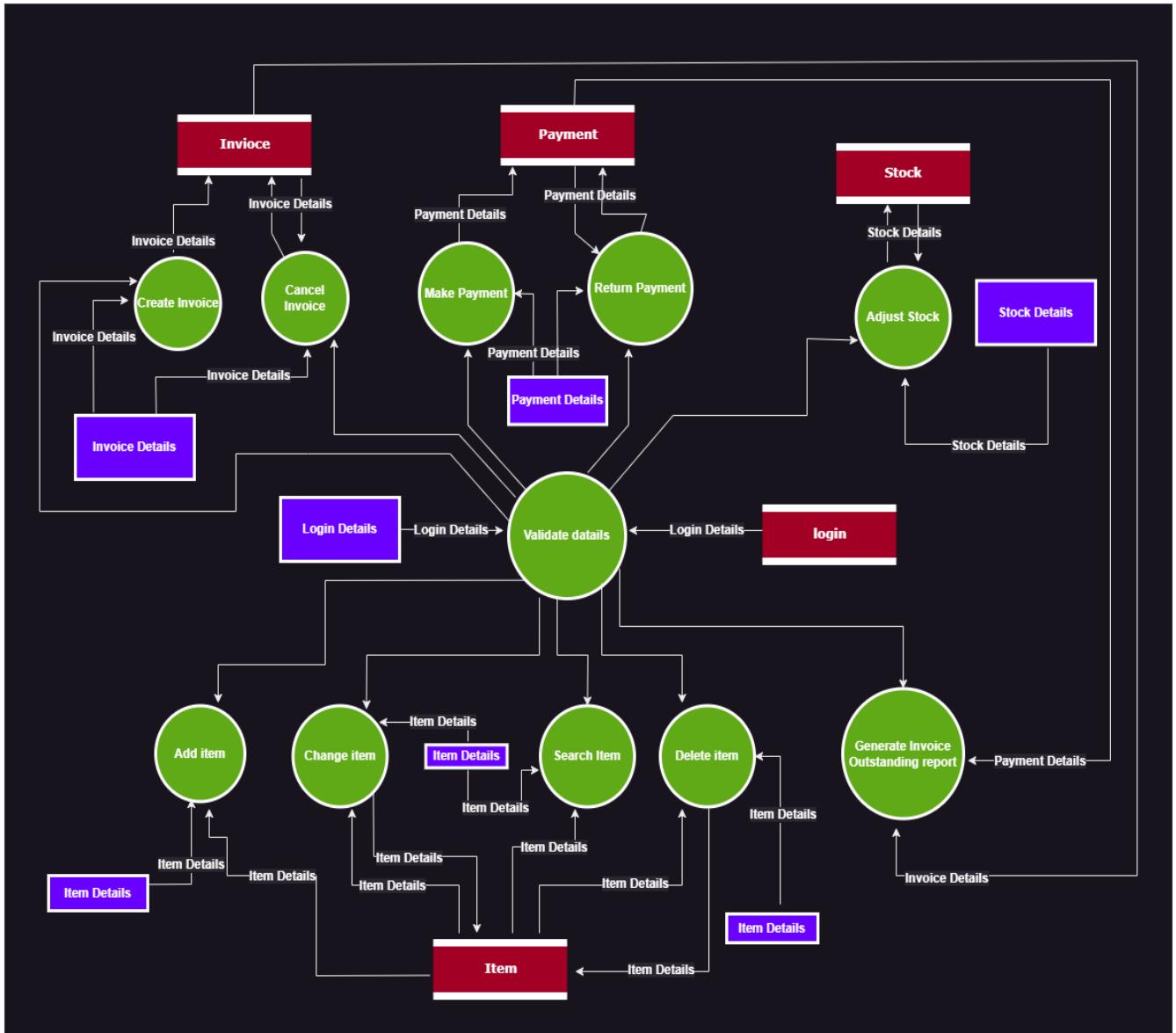


Figure 14 : Detailed Process Flows of Current System.

The detailed process flows provide a comprehensive overview of the current system's operation, delineating the various stages and interactions involved in managing computer shop operations. Figure 13 illustrates the intricate processes and data flows within the existing system, shedding light on key functionalities and their interconnections.

1.11.3 Problems and Limitations of the current system

The current system faces several challenges, including inefficiencies, time wastage, and difficulties in handling certain activities. These problems impact both employees and customers, leading to suboptimal performance and decreased satisfaction levels.

- Outdated Inventory Management: The current system relies on outdated inventory management methods, leading to inaccuracies and inefficiencies in stock tracking and management. Lack of real-time updates hampers decision-making processes and often results in stockouts or overstocking situations.
- Manual Processes: Many tasks, such as updating inventory levels, generating reports, and managing customer inquiries, are performed manually. This not only consumes valuable time but also increases the risk of errors and inconsistencies in data entry and processing.
- Poor Searching System: The current system's search functionality is inadequate, making it challenging for users to find specific items efficiently. For instance, users may need to enter the complete name of an item to search, rather than being able to search by partial keywords, resulting in frustration and inefficiency.
- Lack of Automation: The absence of automated processes, such as alerts for low stock levels, leads to delays in addressing critical issues. Without timely notifications, staff may not be aware of low stock levels until it is too late, resulting in missed sales opportunities or disruptions in service delivery.
- Limited Customer Support: With no integrated customer support system, handling customer inquiries and service requests becomes challenging. Lack of a centralized platform for managing customer interactions results in delayed responses and poor customer satisfaction.
- Inefficient Reporting: Generating reports, such as monthly sales summaries or inventory valuation reports, is a cumbersome and time-consuming process. Manual data extraction and manipulation often lead to inaccuracies and delays in accessing vital business insights.
- Poor Integration: The current system lacks seamless integration with other business tools and software, such as accounting systems or point-of-sale (POS) terminals. This siloed approach results in data duplication, inconsistent records, and additional manual effort to reconcile information across different platforms.

- Limited Scalability: As the business grows, the current system struggles to accommodate increasing data volumes and transactional complexities. Scalability limitations hinder the organization's ability to expand operations and adapt to evolving business needs effectively.
- Data Security Concerns: With manual processes and fragmented data storage, ensuring data security and confidentiality becomes challenging. The risk of data breaches or unauthorized access to sensitive information poses a significant threat to the organization's reputation and compliance with regulatory requirements.

1.11.4 Scope and Constraints

In delineating the scope of the computer shop management system, it is essential to define a subset of functionalities and features that align closely with the organization's immediate needs and objectives. The rationale behind identifying this subset is to ensure a focused and manageable development process that delivers tangible benefits within a reasonable timeframe. By prioritizing specific components of the system, the project can proceed with greater clarity and efficiency, mitigating the risks associated with scope creep and resource overextension.

Constraints:

1. Budgetary Constraints: One of the primary constraints influencing the scope of the system is the allocated budget for the project. The organization may have financial limitations that necessitate a phased approach to system development, focusing on critical functionalities that deliver the highest return on investment within the available budget.

2. Technological Resources: Constraints related to technological resources, such as hardware, software, and technical expertise, can significantly impact the scope of the system. The organization may have limitations in terms of infrastructure or internal capabilities, necessitating the selection of technologies and features that can be effectively implemented with the available resources.

3. Time Constraints: Time constraints are another critical factor influencing the scope of the system. The organization may have specific deadlines or business objectives that require the system to be implemented within a certain timeframe. Therefore, the scope must be defined in a way that allows for timely delivery while balancing the need for thorough development and testing processes.

4. Integration Challenges: Integration constraints may arise from the need to interface the new system with existing software applications, databases, or external systems. Compatibility issues or data migration challenges could limit the scope of integration capabilities, necessitating careful consideration of integration requirements and priorities.

5. User Adoption Considerations: Constraints related to user adoption and acceptance of the new system may influence the scope of features and functionalities. User training requirements, change management processes, and user feedback mechanisms must be incorporated into the scope to ensure successful implementation and adoption of the system.

Conclusion:

In summary, defining the scope of the computer shop management system requires careful consideration of both the organization's objectives and the constraints that may impact system development. By identifying a subset of functionalities that address immediate needs while aligning with budgetary, technological, and other resource constraints, the project can proceed with clarity and efficiency, leading to the successful delivery of a system that meets the organization's requirements within the specified constraints.

1.12 - Requirements specification for new system

The Requirements Specification chapter outlines the detailed description of the functionalities and features that the new computer shop management system is expected to deliver. It serves as a comprehensive guide for system developers, designers, and stakeholders to understand the specific requirements and objectives of the project. By clearly defining the system requirements, this chapter lays the foundation for the design, development, and implementation phases of the project.

1.12.1 Module 1: Web interface (Website)

- 4.1.1: The system shall display available products in a categorized format (e.g., by product type, brand, etc.).
- 4.1.2 The system must allow users to sort products by different criteria such as price, popularity, and new arrivals.
- 4.1.3: The system must display detailed information for each product, including images, specifications, availability, and reviews.
- 4.1.4: The system must provide real-time visibility of current stock levels for each product.
- 4.1.5: The system must display product details when a user selects a product from the search results or product listings.
- 4.1.6: The product detail page must include product images, specifications, available stock, customer ratings, and reviews.

- 4.1.7: The system must allow users to add products to a Wishlist for future reference
- 4.1.8: The system must notify users when a product they added to their cart or Wishlist becomes unavailable or out of stock.
- 4.1.9: The system must notify users of any promotional offers or sales based on their preferences and browsing history.
- 4.1.10: The system must provide personalized product recommendations based on the user's browsing and purchase history.
- 4.1.11: The system must provide a chatbot integrated into the web interface to assist users in navigating the website, browsing products, and answering frequently asked questions.
- 4.1.12: The system must allow users to ask questions related to product details, availability, and general store information.
- 4.1.13: The system must allow users to submit feedback or queries through a contact form.
- 4.1.14: The system must allow users to track the status of their submitted feedback or queries.
- 4.1.15: The system must allow users to view FAQ sections for quick help.
- 4.1.16: The system must allow users to view reviews for a particular product.
- 4.1.17: The system must allow users to submit reviews for purchased products
- 4.1.18 The system must display an average rating for each product based on user reviews.

1.12.2 Module 2: AI Chatbot for Customer Interaction

- 4.2.1: The AI chatbot must be capable of engaging with customers in natural language conversations, simulating human-like interaction.
- 4.2.2: The chatbot must provide personalized assistance to customers based on their inquiries, purchase history, and preferences.
- 4.2.3: The system must support customers to interact with the chatbot via website chat.
- 4.2.4: The chatbot must have the ability to answer frequently asked questions (FAQs) related to products, services, store locations, operating hours, and other relevant topics.

- 4.2.5: The system must integrate with the customer management module to access and update customer profiles, purchase history, and interaction logs.
- 4.2.6: The chatbot must be equipped with natural language processing (NLP) capabilities to understand and interpret user queries accurately.
- 4.2.7: The system must offer proactive engagement features, such as sending personalized product recommendations, promotional offers, and order status updates to customers.
- 4.2.8: The chatbot must seamlessly hand off conversations to human agents when it encounters complex inquiries or requests beyond its capabilities.
- 4.2.10: The chatbot must be continuously trained and improved using machine learning algorithms to enhance its conversational abilities and accuracy over time.
- 4.2.11: The system must adhere to data privacy regulations and security standards to protect sensitive customer information shared during interactions.
- 4.2.12: The chatbot interface must be customizable to reflect the brand identity and personality of the organization, including branded messages, logos, and visual elements.
- 4.2.13: The system must provide analytics and reporting capabilities to track chatbot performance metrics, such as response time, customer satisfaction ratings, and conversation outcomes.
- 4.2.14: The chatbot must support integration with other modules of the system, such as inventory management, sales, and customer management, to access and update relevant data in real-time.
- 4.2.15: The system must offer scalability to accommodate fluctuations in chatbot usage volume and handle concurrent interactions effectively.
- 4.2.16: The chatbot must be accessible 24/7 to provide round-the-clock customer support and assistance, improving accessibility and convenience for customers.

1.12.3 Module 3: AI Chatbot for Shop Owner

- 4.3.1: The AI chatbot must provide shop owners with real-time insights and analytics regarding sales performance, inventory levels, and customer engagement metrics.
- 4.3.2: The chatbot must be capable of answering shop owner queries related to business operations, such as revenue trends, profit margins, and inventory turnover rates.

- 4.3.3: The system must integrate with the sales management module to retrieve and analyze transaction data, including sales volumes, average order values, and product popularity.
- 4.3.4: The chatbot must offer predictive analytics capabilities to forecast future sales trends, identify potential inventory shortages, and recommend restocking strategies.
- 4.3.5: The system must provide customizable dashboards and reports for shop owners to visualize key performance indicators (KPIs) and track business metrics effectively.
- 4.3.6: The chatbot must support natural language querying, allowing shop owners to ask questions using conversational language and receive relevant insights in real-time.
- 4.3.7: The system must offer proactive alerts and notifications to shop owners for critical events, such as low inventory levels, high-demand products, or unusual sales patterns.
- 4.3.8: The chatbot must integrate with external data sources, such as market trends, weather forecasts, and competitor analysis, to provide contextual insights and recommendations.
- 4.3.9: The system must maintain a history of shop owner interactions and queries, allowing for trend analysis, pattern recognition, and continuous improvement of the chatbot's capabilities.
- 4.3.10: The chatbot must be accessible via multiple channels, including web interface, mobile app, and voice-activated assistants, to accommodate shop owners' preferences and accessibility needs.
- 4.3.11: The system must adhere to data privacy regulations and security standards to safeguard sensitive business information shared during interactions.
- 4.3.12: The chatbot interface must be customizable to meet the unique preferences and requirements of individual shop owners, including personalized notifications, alerts, and dashboard layouts.
- 4.3.13: The system must offer integration with third-party business intelligence tools and platforms to leverage existing analytics capabilities and extend functionality as needed.
- 4.3.14: The chatbot must support multi-store management, allowing shop owners to access data and insights across multiple locations from a centralized interface.
- 4.3.15: The system must provide scalability to accommodate growing data volumes, increasing user interactions, and evolving business needs over time.

- 4.3.16: The chatbot must offer language localization options to support shop owners in different regions or countries, with multilingual support for queries and responses.
- 4.3.17: The system must provide role-based access control to ensure that shop owners only have access to relevant data and features based on their role within the organization.
- 4.3.18: The chatbot must offer data visualization tools, such as charts and graphs, to facilitate easy interpretation of analytics and trends.
- 4.3.19: The system must support collaboration features, allowing shop owners to share insights and reports with team members for decision-making purposes.
- 4.3.20: The chatbot must continuously learn and improve its capabilities through machine learning algorithms, adapting to changing business dynamics and user preferences.
- 4.3.21: The system must provide export functionality for reports and analytics data in various formats, such as PDF, CSV, and Excel, for further analysis or sharing with stakeholders.
- 4.3.22: The chatbot must offer scheduling and reminders functionality, allowing shop owners to set tasks and appointments and receive timely notifications for follow-up actions.
- 4.3.23: The system must support integration with customer relationship management (CRM) software to enhance customer engagement and retention strategies based on chatbot interactions and insights.
- 4.3.24: The chatbot must have the capability to handle complex queries and requests, including trend analysis, scenario planning, and what-if analysis, to support strategic decision-making by shop owners.
- 4.3.25: The system must provide training and support resources for shop owners to maximize the effectiveness and utilization of the chatbot, including user guides, tutorials, and online forums.

Design Report

1.13 Description of new system

1.13.1 Narrative

RK Computers and CCTV Operators is a shop located in Ambalangoda that deals with selling computers, parts, accessories, and repairing computers for their customers. The existing system in the shop is an old digital system, which handles all day-to-day activities on a complex digital database. This system has several inefficiencies, such as wasting the time of both employees and customers and making some activities harder to manage.

To address these problems and fulfill other requirements, we propose a computer shop management system with an AI chatbot. This new system aims to solve the existing issues and increase the efficiency of the shop's operations. It provides functionalities such as managing customers, suppliers, and item details, handling sales and repairs, maintaining stock, generating reports using AI, and calculating income and expenses.

System Breakdown Structure

The Computer Store Management System with AI Chat Bot is a complex software application designed to streamline operations in a computer store. The system can be broken down into several key modules, each responsible for specific functionalities. These modules work together to provide a seamless experience for both the store's customers and the staff. Below is a detailed breakdown of the system's structure.

1. Web Interface Module

The Web Interface Module is responsible for all interactions between the users and the system. It includes both the web interface for customers and the administrative interface for staff.

- **Customer Web Interface:** Provides an interactive and user-friendly platform for customers to browse products, check prices, read reviews, and make purchases.
- **Admin Interface:** Enables store staff to manage inventory, update product information, view sales reports, and handle customer queries.

2. AI Chat Bot for customer Module

The AI Chat Bot Module is designed to assist customers by providing recommendations, answering queries, and facilitating the purchase process. This module integrates natural language processing (NLP) and machine learning (ML) algorithms to understand and respond to customer inquiries effectively.

- **Customer Care:** Helps customers with product inquiries, order tracking, and returns.
- **Recommendations:** Provides personalized product recommendations based on user preferences and browsing history.
- **Sales Assistance:** Assists customers in the purchase process by answering questions and guiding them through checkout.

3. AI Chat Bot for owner Module

The AI Chatbot for Owner Module is designed to help the shop owner efficiently manage the operations of the computer shop. This module leverages natural language processing (NLP) and machine learning (ML) to provide a range of administrative functionalities and support.

- **Inventory Management:** Allows the owner to update and query inventory levels, ensuring accurate stock information and efficient restocking.
- **Reports Generation:** Facilitates the generation of various reports, such as sales reports and inventory summaries, to help the owner make data-driven decisions.
- **Data Querying:** Provides capabilities to retrieve specific data from the system, such as sales figures or stock details, based on owner requests.

4. Database Management Module

The Database Management Module handles all data storage and retrieval operations. It ensures that data is stored securely and can be accessed efficiently.

- **Product Database:** Stores detailed information about products, including descriptions, prices, and stock levels.
- **User Database:** Manages customer and staff information, including login credentials, purchase history, and personal details.

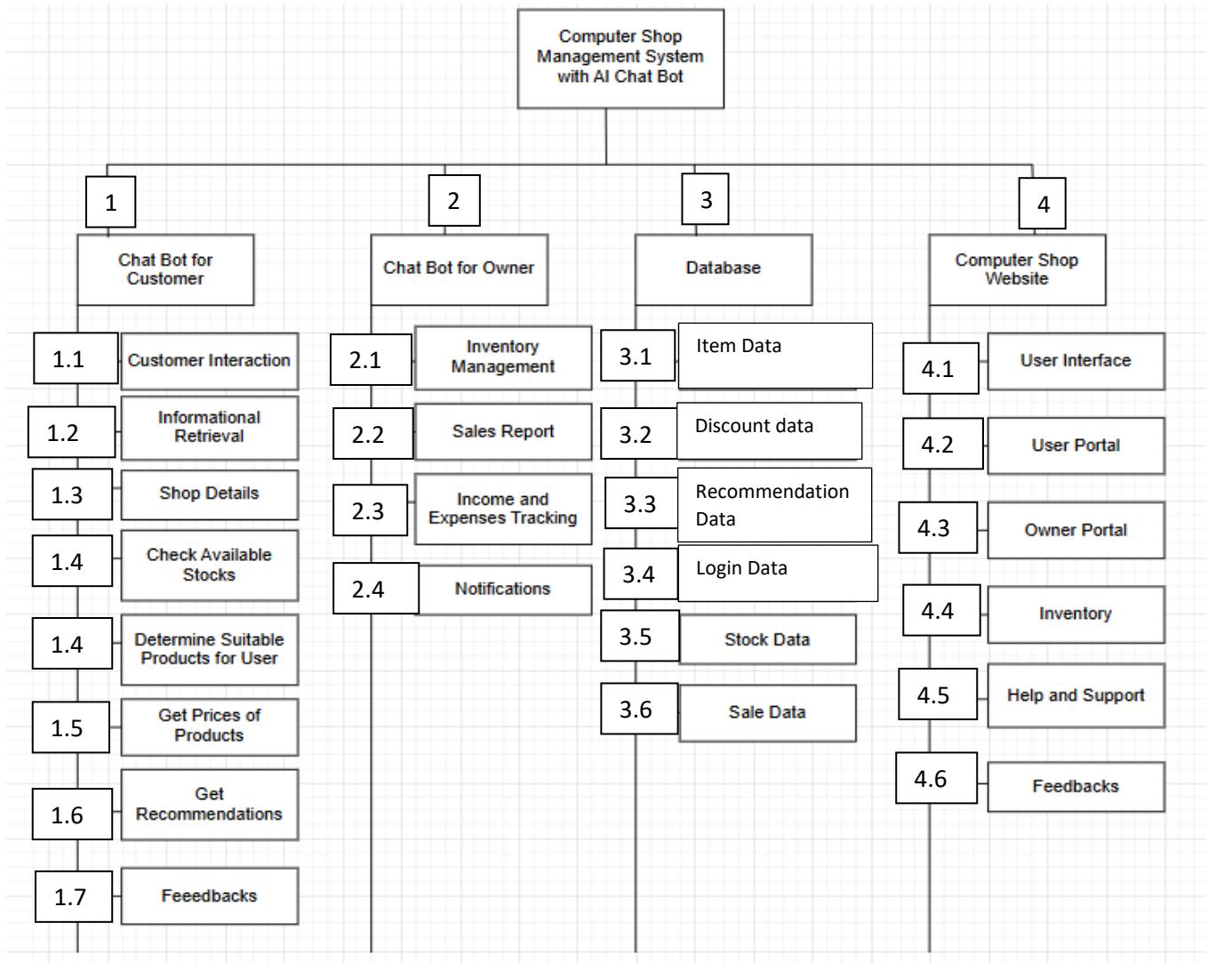


Figure 15: System Breakdown Structure

1.13.2 Process flows

Overview

The new system is developed to enhance the operational efficiency of a computer shop by integrating multiple modules and functionalities into a cohesive solution. This system includes a customer-facing website, AI chatbots for both customers and the shop owner, and a robust backend infrastructure for managing inventory, sales, and reporting. The Work Breakdown Structure (WBS) provides a systematic approach to managing and executing this complex project, ensuring all components are developed, integrated, and maintained effectively.

Work Breakdown Structure (WBS)

The WBS for the new system is divided into major components and tasks to ensure organized and systematic development. Each component of the WBS is crucial for the successful implementation and functionality of the system.

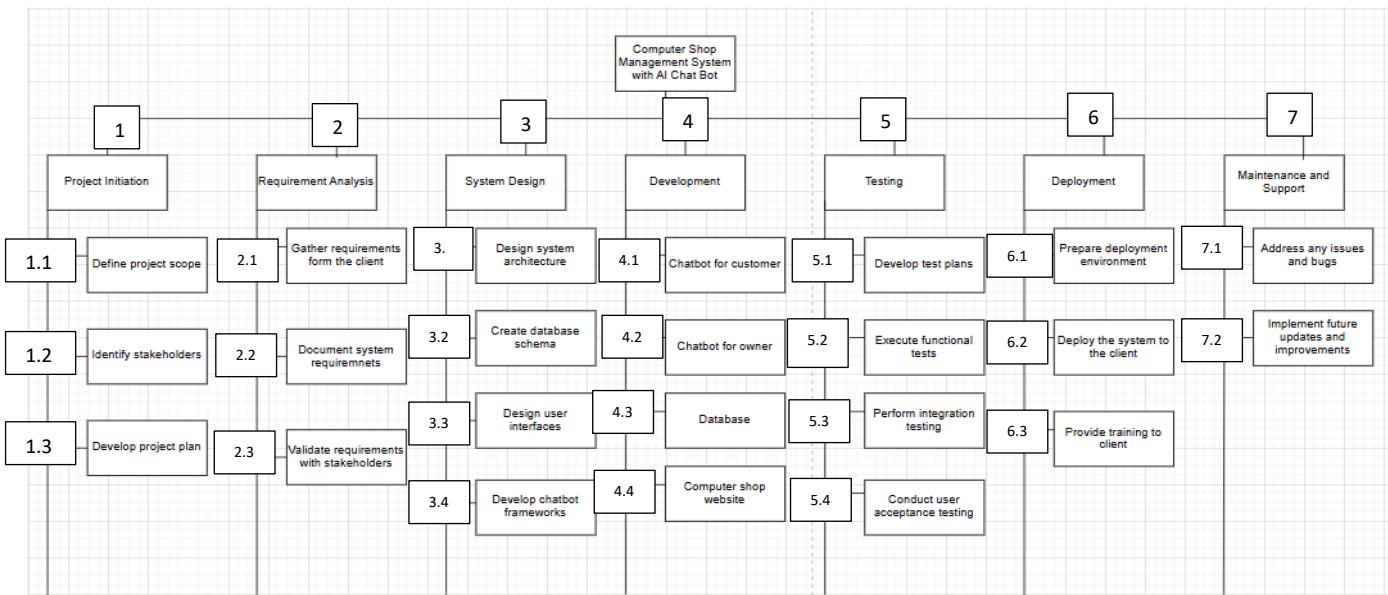


Figure 16: Work Breakdown Structure

1.14 Detailed Design

1.15 Data Design

Dataflow Diagram

A Data Flow Diagram (DFD) provides a visual representation of how data flows within the Computer Store Management System with AI Chat Bot. It helps to understand how data is processed, stored, and utilized throughout the system. The DFD can be broken down into multiple levels, starting from a high-level overview (Level 0) and drilling down into more detailed views (Level 1).

Level 0:

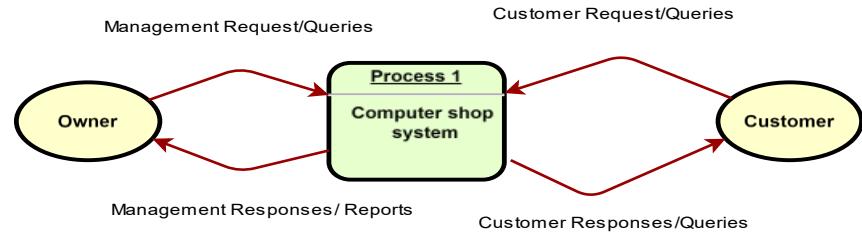


Figure 17: Level 0 Dataflow Diagram

Level 1:

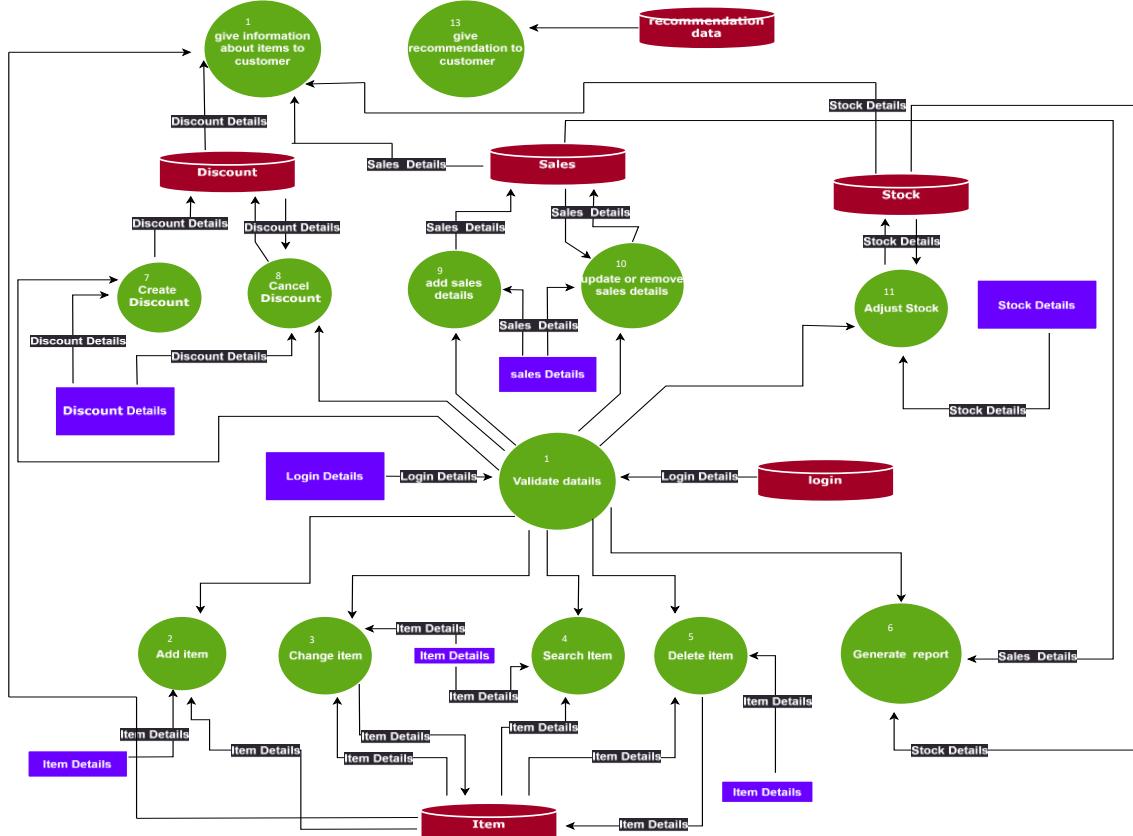


Figure 18: Level 1 Dataflow Diagram

ER Diagram

An Entity-Relationship (ER) Diagram is a visual representation of the entities within a system and the relationships between them. For the Computer Store Management System with AI Chat Bot, the ER diagram captures the essential entities and their interactions.



Figure 19: Class Diagram for Database

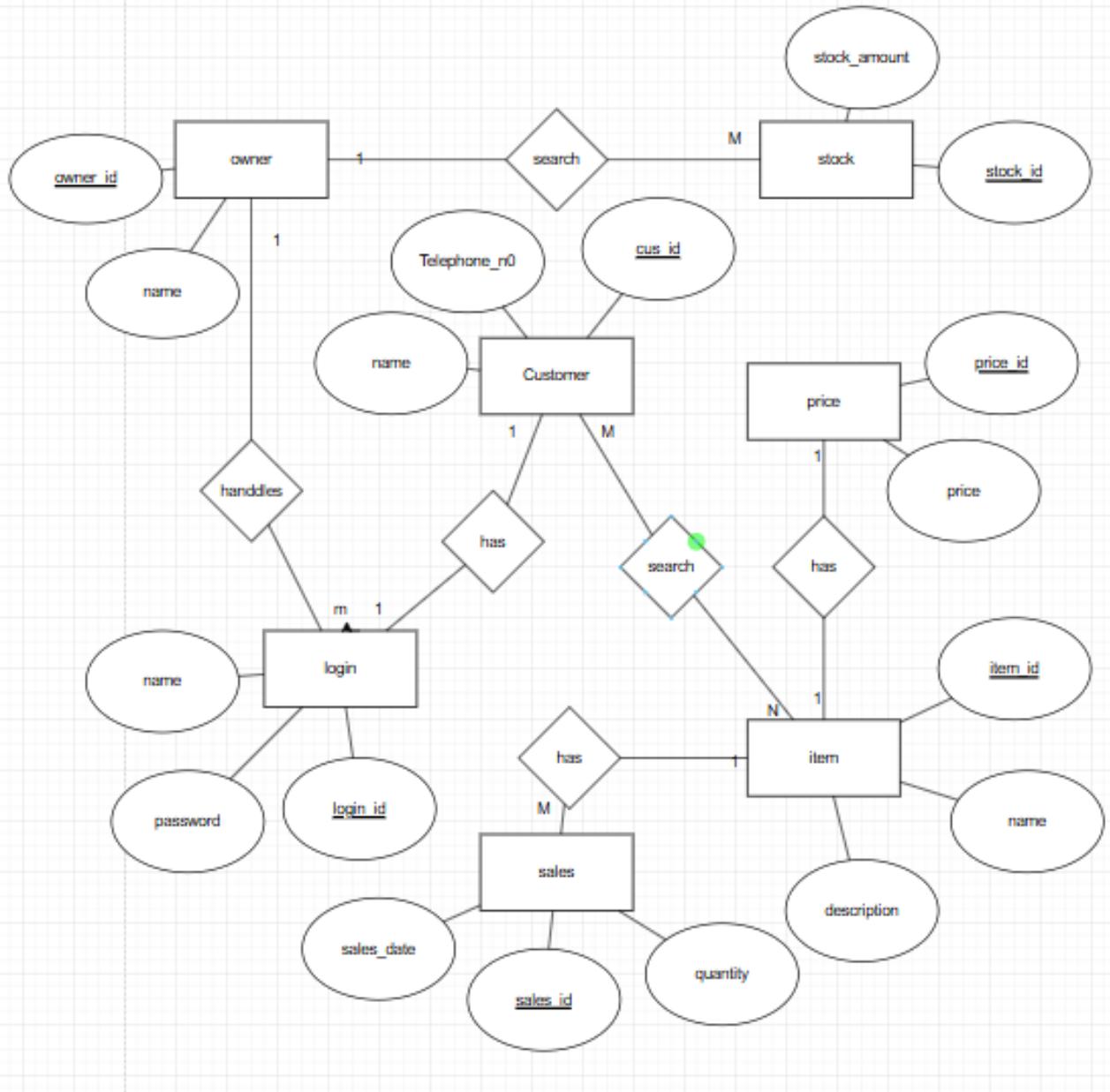


Figure 20: ER Diagram

Data Dictionary

The **Data Dictionary** provides definitions and details for each data element used in the Computer Shop Management System. It helps in understanding the structure, purpose, and constraints of the data within the system.

Table: Login

Table 3: Login Table

Data Element	Description	Data Type	Constraints
UserID	Unique identifier for each user	INT	Primary Key, Not Null, Auto-increment
UserName	Username of the user	TEXT	Not Null
UserPassword	Password for user authentication	TEXT	Not Null

Table: AppsSpec

Table 4: AppsSpec Table

Data Element	Description	Data Type	Constraints
App_ID	Unique identifier for each application	CHAR(7)	Primary Key
App_Name	Name of the application	CHAR(50)	Not Null
App_Type	Type of the application (e.g., Video Game, Software)	CHAR(30)	Not Null
Processor	Required processor for the application	CHAR(25)	Not Null
Storage	Required storage space for the application	CHAR(10)	Not Null
RAM	Required RAM for the application	CHAR(10)	Not Null
Video_RAM	Required video RAM for the application	CHAR(10)	Not Null

Table: Items

Table 5: Items Table

Data Element	Description	Data Type	Constraints
Id	Unique identifier for each item	INT	Primary Key, Not Null, Auto-increment
item_name	Name of the item	CHAR(50)	Unique, Not Null
Item_Pic_Url	URL of the item's picture	TEXT	Not Null
Item_detail	Description and details about the item	TEXT	Not Null

Table: Stock

Table 6: Stock Table

Data Element	Description	Data Type	Constraints
Id	Unique identifier for each item	INT	Primary Key, Not Null (Foreign Key referencing Items(id))
stock	Quantity of the item in stock	INT	Not Null

Table: Price

Table 7: Price Table

Data Element	Description	Data Type	Constraints
Id	Unique identifier for each item	INT	Primary Key, Not Null (Foreign Key referencing Items(id))
price	Price of the item	FLOAT(15,2)	Not Null

Table: Discount

Table 8: Discount Table

Data Element	Description	Data Type	Constraints
Id	Unique identifier for each item	INT	Primary Key, Not Null (Foreign Key referencing Items(id))
discount_rate	Discount rate applicable to the item	FLOAT	Not Null

Table: Warranty

Table 9: Warranty Table

Data Element	Description	Data Type	Constraints
Id	Unique identifier for each item	INT	Primary Key, Not Null (Foreign Key referencing Items(id))
warranty_in_months	Duration of the warranty in months	INT	Not Null

Table: Sales

Table 10: Sales Table

Data Element	Description	Data Type	Constraints
SaleID	Unique identifier for each sale	INT	Primary Key, Not Null, Auto-increment
Id	Identifier of the sold item	INT	Foreign Key referencing Items(id)
QuantitySold	Quantity of items sold	INT	Not Null
SaleDate	Date of the sale	DATE	Not Null

1.16 Architectural Design

Use Case Diagrams

A Use Case Diagrams is a visual representation of the interactions between users (actors) and the system, illustrating the various use cases or functionalities provided by the system. For the Computer Store Management System with AI Chat Bot, the main actors are the Customer, Admin, and Supplier, each interacting with the system in different ways.

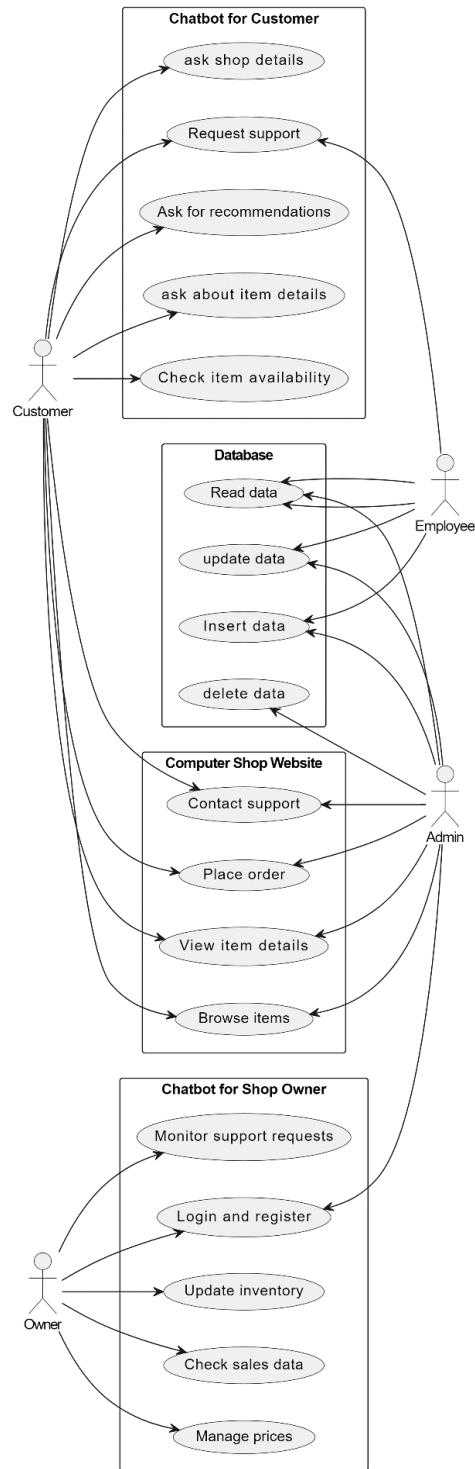


Figure 21: Use Case Diagrams

High Level Architecture

The high-level architecture design of the Computer Store Management System with AI Chat Bot provides a comprehensive overview of the system's structure and the interactions between its various components. This architecture is designed to facilitate efficient operations within a computer store, enhancing both customer experience and administrative efficiency.

Interactions and Data Flow

The architecture emphasizes seamless interactions between modules. The Web Interface Module communicates with both the AI Chat Bot modules and the Database Management Module to fetch and display relevant data. The AI Chat Bot for Customer Module interacts with customers, while the AI Chat Bot for Owner Module supports the shop owner in managing operations. All modules rely on the Database Management Module for secure and efficient data storage and retrieval.

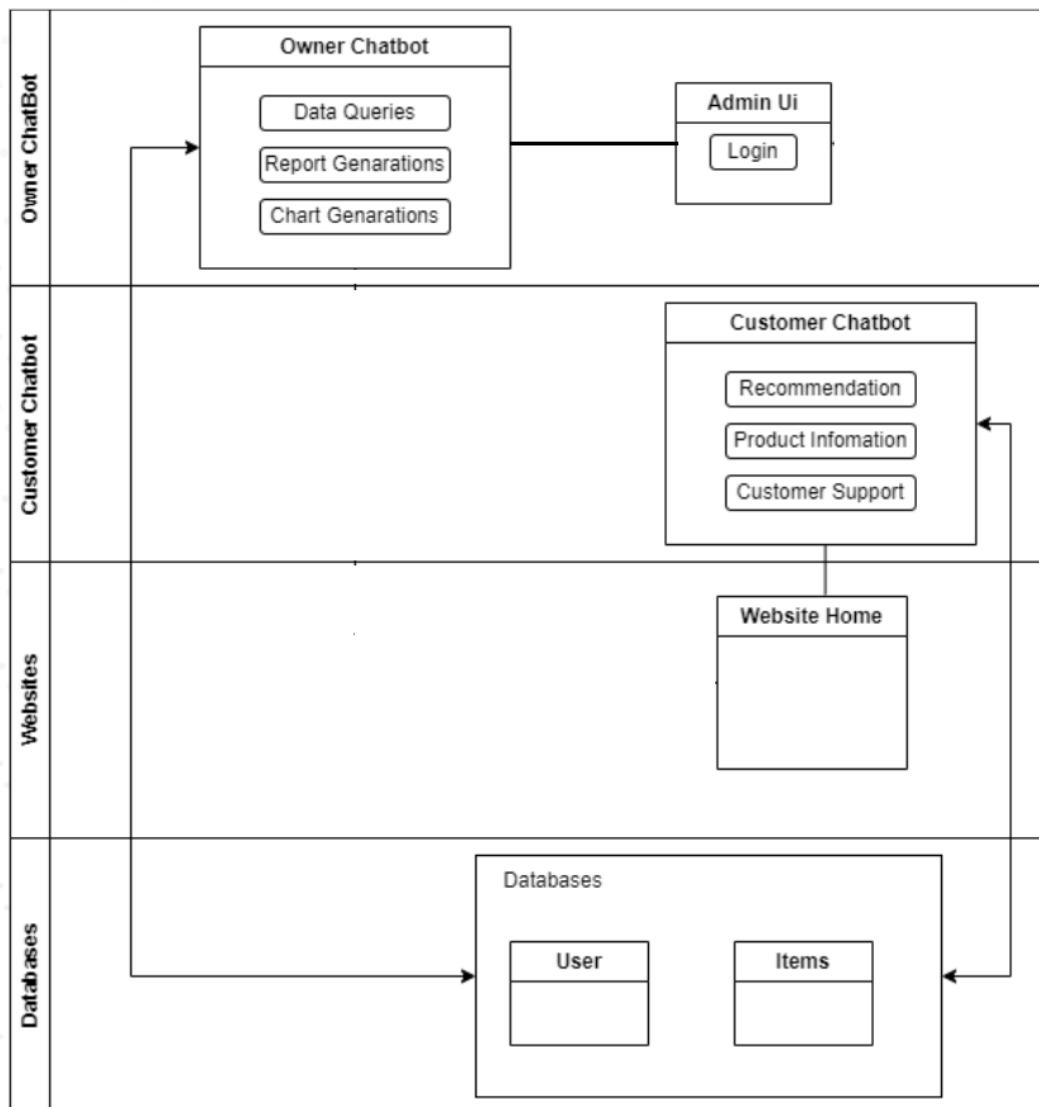


Figure 22: High level architecture design

1.17 Procedural Design

1.17.1 Process Flow for the AI Chatbot for Customers

- 1. Customer Inquiry:**
 - Customer initiates a conversation with the AI chatbot to inquire about products.
 - AI chatbot receives the inquiry and identifies the type of information requested (e.g., product details, availability, price).
- 2. Retrieve Product Information:**
 - AI chatbot queries the product database to fetch the relevant information.
 - Database responds with the requested product details.
- 3. Provide Information:**
 - AI chatbot formats the information and responds to the customer with the product details.
 - If the customer has further inquiries, the process loops back to step 1.

1.17.2 Process Flow for the AI Chatbot for Owners

- 1. Owner Login:**
 - The owner logs into the system using secure credentials.
 - System authenticates the owner.
- 2. Update Product Information:**
 - The owner selects the option to update product information.
 - AI chatbot prompts the owner for the details of the update (e.g., product name, specifications, stock levels).
- 3. Process Update:**
 - AI chatbot validates the input data.
 - AI chatbot sends the update request to the database.
- 4. Confirm Update:**
 - Database processes the update and confirms the changes.
 - AI chatbot informs the owner that the update was successful.

1.17.3 Process Flow for the Website

- 1. Customer Browses Website:**
 - Customer navigates to the website to browse products.
 - Websites display a list of products with basic details (e.g., name, image, price).
- 2. View Product Details:**
 - Customer selects a product to view more details.
 - Website retrieves detailed product information from the database.

1.17.4 Combined System Process Flow

1. Customer Interaction with AI Chatbot:

- Customer opens chat with AI chatbot on the website.
- AI chatbot greets the customer and helps.
- Customer inquiries about a specific product.
- AI chatbot queries the product database and retrieves information.
- AI chatbot provides product details to the customer.

2. Owner Interaction with AI Chatbot:

- Owner logs into the AI chatbot system.
- AI chatbot authenticates the owner.
- Owner selects to update product information.
- AI chatbot prompts for update details.
- The owner provides new product details (e.g., price, stock levels).
- AI chatbot validates and processes the update.
- Database updates the product information.
- AI chatbot confirms the successful update to the owner.

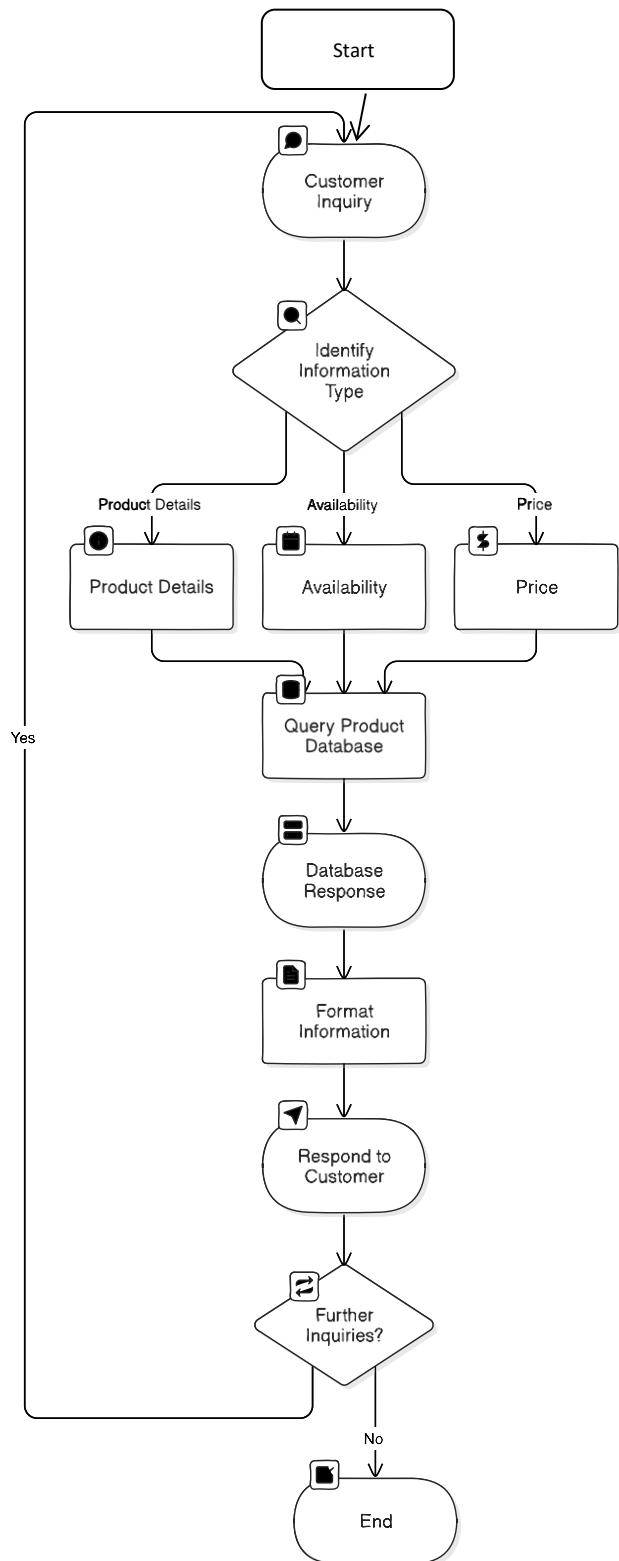
3. Customer Browsing the Website:

- Customer visits the website to browse products.
- The website displays product categories and featured items.
- Customer clicks on a product to see more details.
- Website fetches detailed product information from the database.
- Customer adds items to their shopping cart.
- Customer reviews the cart and proceeds to checkout.
- Website collects payment and shipping details.
- Payment is processed and order is confirmed.
- Website updates order status in the database.
- Customer receives an email confirmation of the order.

Flow charts

Flowcharts provide a graphical representation of the processes and workflows within the Computer Store Management System with AI Chat Bot. They help in understanding the sequence of steps involved in various operations and decision points in the system.

- AI Chatbot for Customer



- AI Chatbot for Owner

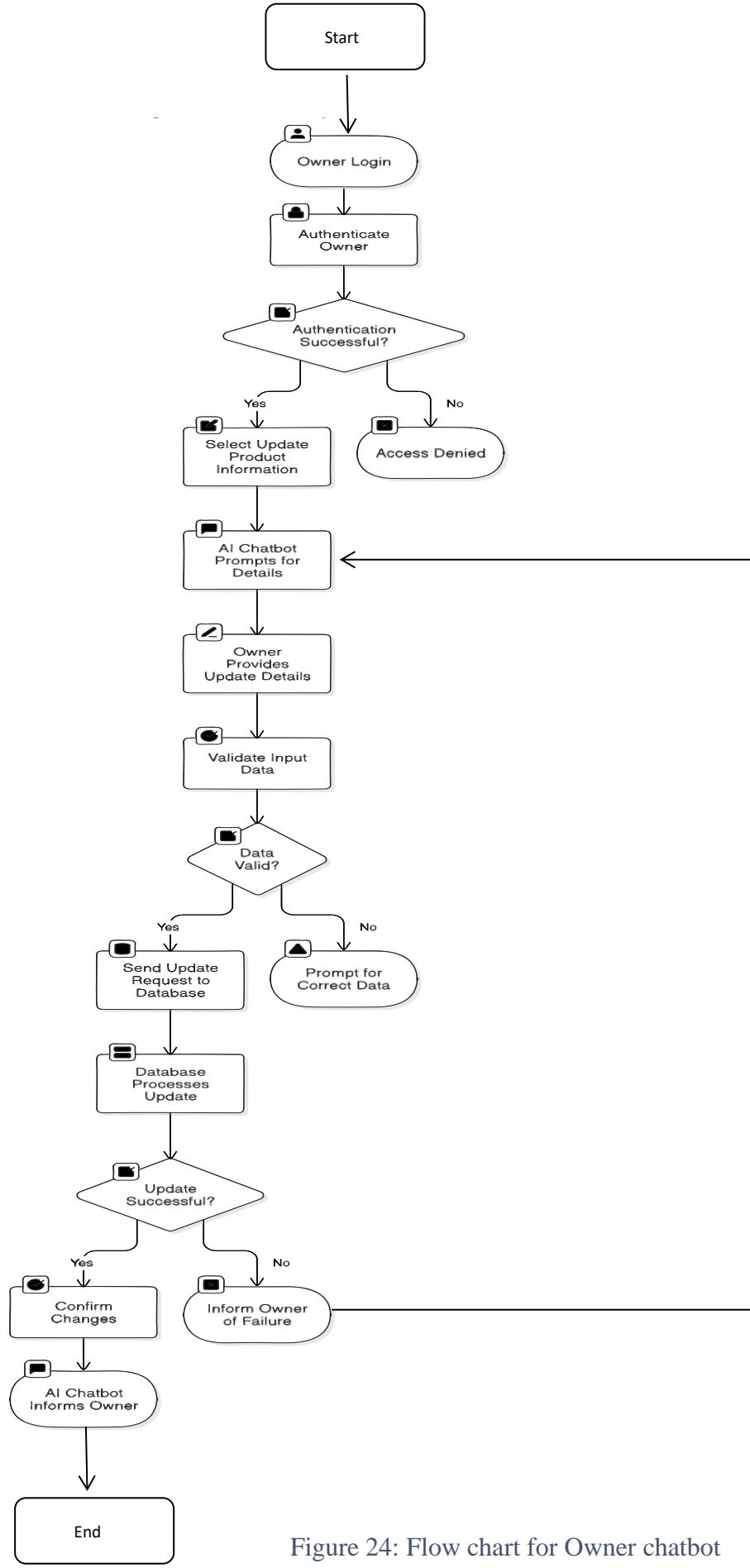


Figure 24: Flow chart for Owner chatbot

- Website

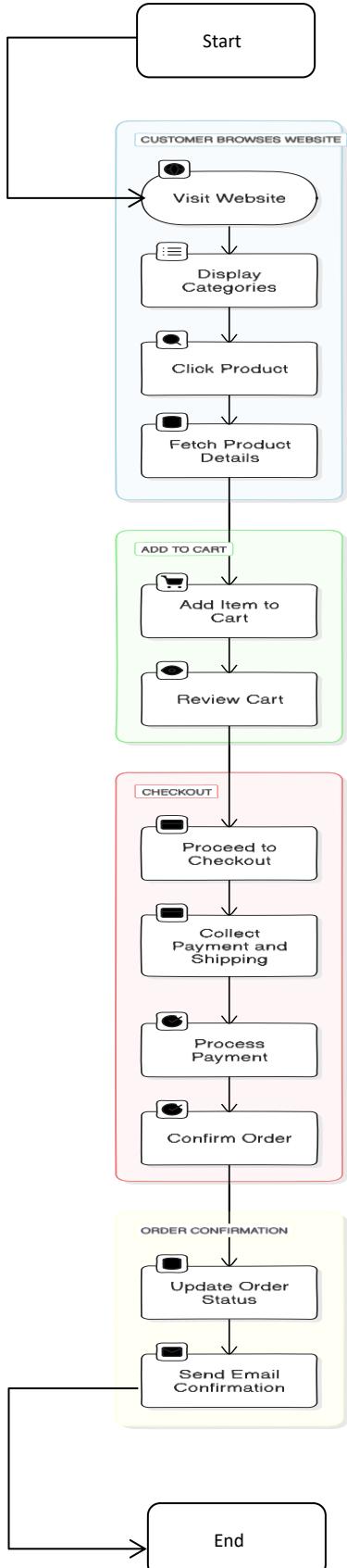


Figure 25: Flow chart for website

1.18 Interface Design

User interfaces

User interfaces (UIs) are critical components of the Computer Store Management System with AI Chat Bot. They provide the means through which users interact with the system, including customers, admins, and suppliers. Well-designed UIs enhance usability, improve user experience, and ensure efficient system operation.

- AI Chatbot for Customer

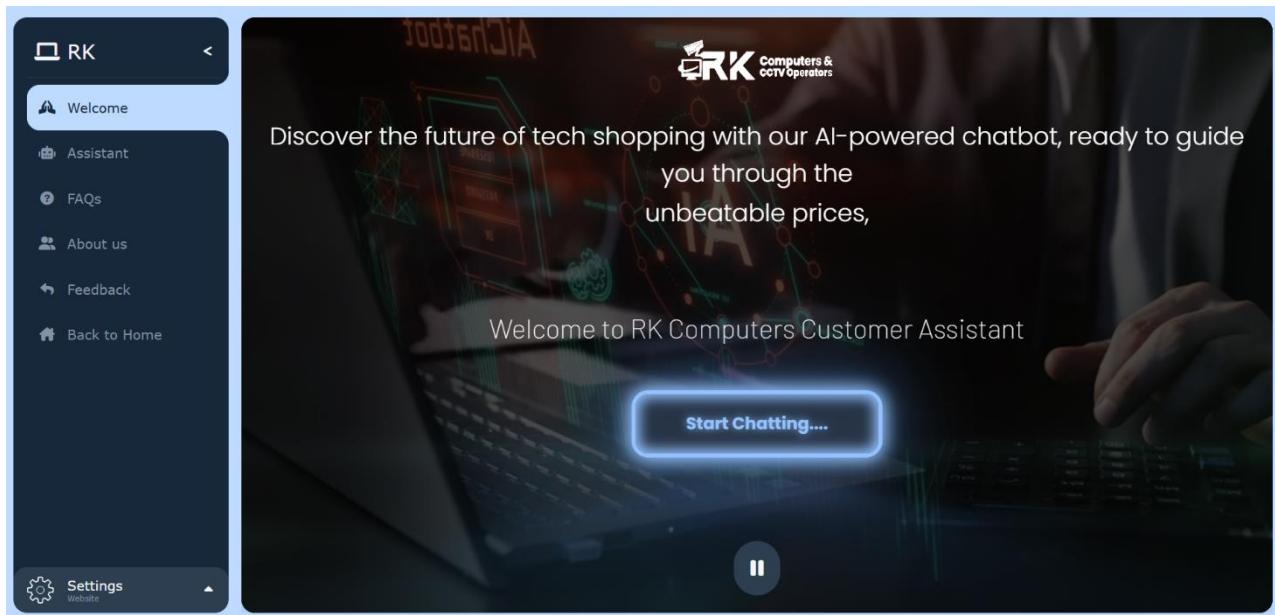


Figure 26: UI for customer chatbot welcome page

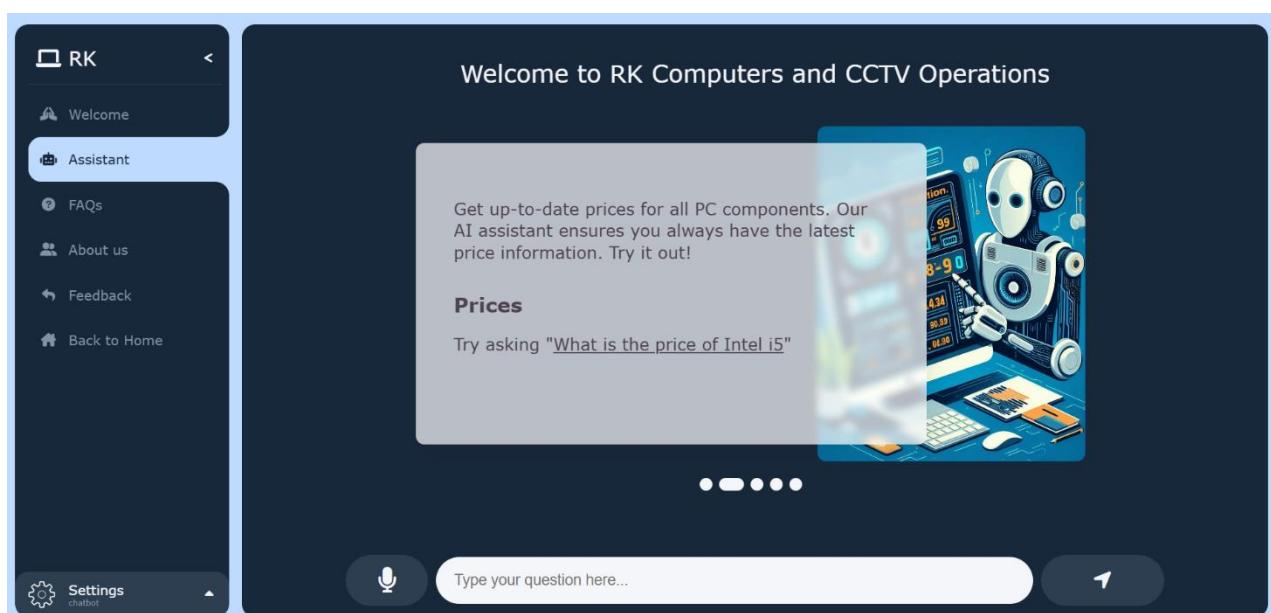


Figure 27: UI for customer chatbot

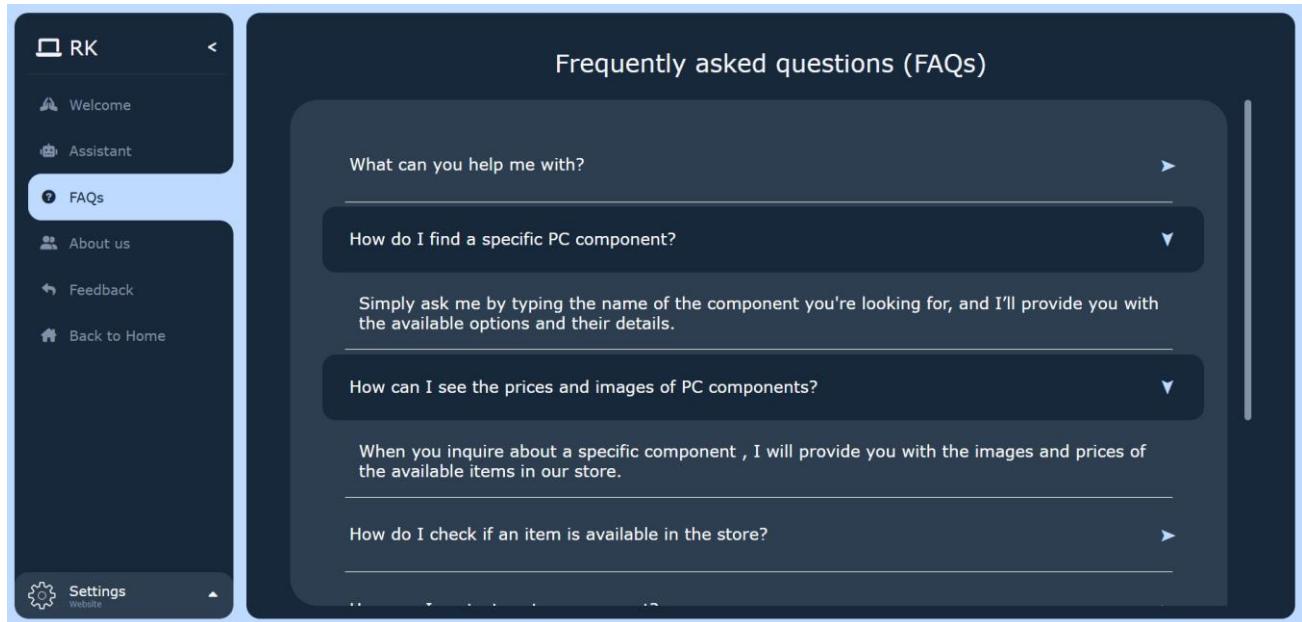


Figure 28: UI for customer chatbot FAQ page

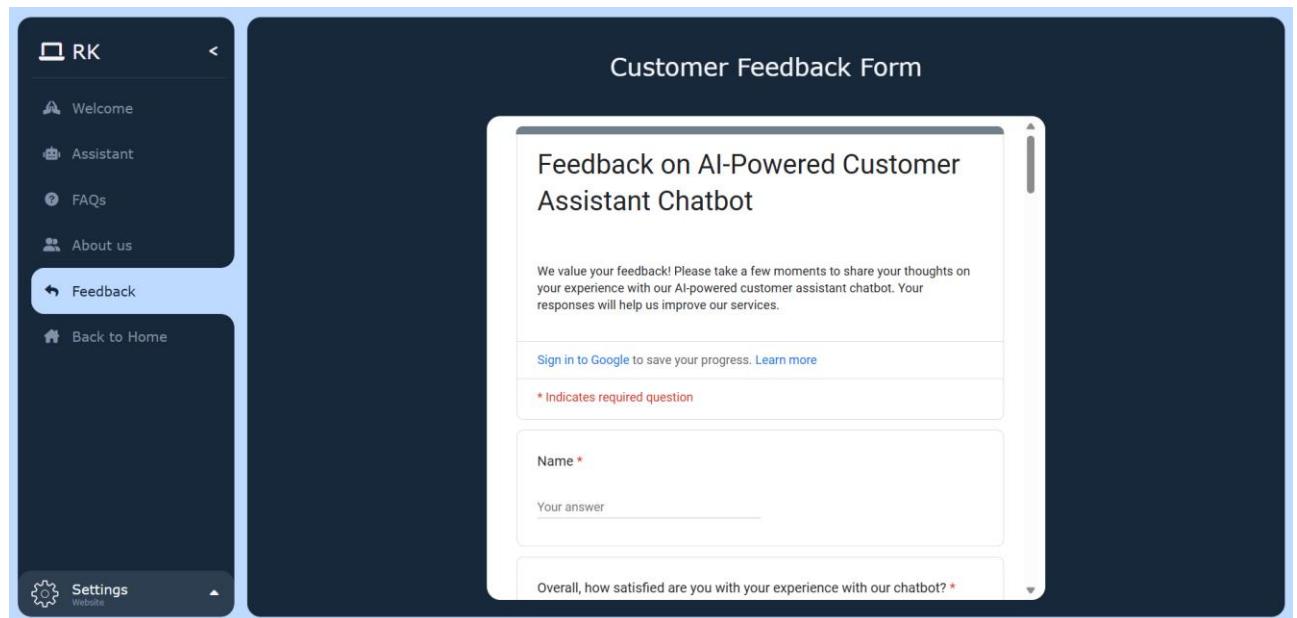


Figure 29: UI for customer chatbot feedback page

- AI Chatbot for Owner



Figure 30: UI for Owner Chatbot

- Website

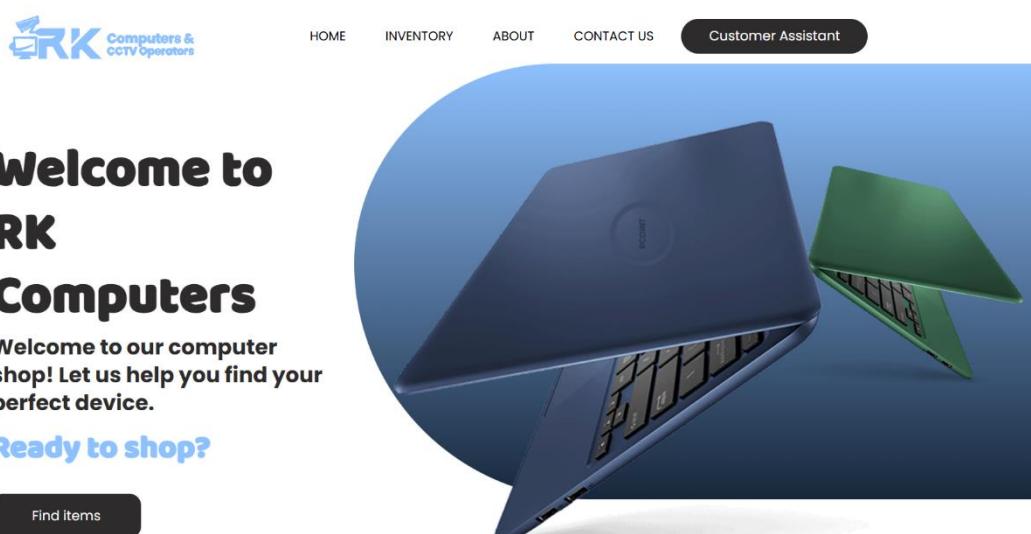


Figure 31: Website Home page UI

Our Services

Our services include expert computer repairs, hardware and software installations, IT consulting, and personalized tech support. We also offer custom-built PCs, networking solutions, and reliable after-sales support to ensure your devices run smoothly and efficiently.



LAPTOP SERVICE

Service your laptop for free for the first time. (RK Computers customers only)



WARRANTY

Company warranty upto 2 years.
Software warranty upto 1 year. (RK Computers customers only)



LAPTOP REPAIRS

Get your laptop back in top working condition with our expert repair services.

[Read more](#)



HOME

INVENTORY

ABOUT

CONTACT US

Customer Assistant

Who are we..

Connecting Possibilities, One Byte at a Time.

Your Trusted Tech Partner

Welcome to RK Computers, your one-stop destination for cutting-edge technology and exceptional service. Located in Ampara and Galle, we offer a diverse range of products, including the latest laptops, high-performance CPUs, quality headsets, and precision mice. Our services include expert repairs, custom PC builds, and IT support tailored to both individual and business needs. At RK Computers, we pride ourselves on providing personalized customer experiences, ensuring you find the perfect tech solutions. Whether you're a gamer, a student, or a professional, we're here to keep you connected and empowered with top-notch products and reliable support. We also provide comprehensive after-sales support, ensuring that your tech stays in peak condition long after your purchase. With a commitment to affordability, we offer competitive pricing and frequent promotions to help you get the best value for your money. Visit RK Computers today and let us help you stay



HOME

INVENTORY

ABOUT

CONTACT US

Customer Assistant

Reach us out

we are at your service



Figure 32: Website Contact page UI

1.19 - Hardware and Software selection

1.19.1 Development hardware and software equipment

This section outlines the hardware and software equipment used during the development of the Computer Store Management System with AI Chat Bot. These resources are essential for designing, coding, testing, and deploying the system.

Development Hardware

1. Servers

- **Description:** Servers used for hosting development environments, databases, and the AI Chat Bot.

2. Network Equipment

- **Description:** Equipment used to ensure connectivity between development workstations and servers.
- **Specifications:**
 - **Router:** High-speed router with gigabit Ethernet ports

3. Testing Devices

- **Description:** Devices used to evaluate the system's compatibility and performance across different platforms.
- **Specifications:**
 - **Smartphones/Tablets:** Android and iOS devices for mobile interface testing
 - **Browsers:** Various web browsers (Chrome, Firefox, Safari, Edge) for cross-browser testing

Development Software

1. Integrated Development Environments (IDEs)

- **Description:** Software tools used for coding, debugging, and testing the application.
- **Examples:**
 - **Visual Studio Code:** Popular IDE for web development with extensive plugin support.
 - **PyCharm:** IDE for Python development, suitable for implementing the AI Chat Bot.

2. Database Management Systems (DBMS)

- **Description:** Software used to manage and interact with the system's database.
- **Examples:**
 - **MySQL:** Open-source relational database management system.

3. Web Servers

- **Description:** Software used to serve web applications to users.
- **Examples:**
 - **Apache:** Open-source web server software.
 - **Nginx:** High-performance web server and reverse proxy.

4. Version Control Systems

- **Description:** Tools used for managing changes to source code and collaborative development.
- **Examples:**
 - **Git:** Distributed version control system, with platforms like GitHub or GitLab for repository hosting.

5. AI and Machine Learning Libraries

- **Description:** Libraries used for implementing the AI Chat Bot and other machine learning functionalities.
- **Examples:**
 - **TensorFlow:** Open-source machine learning library.
 - **RASA:** Open-source machine learning framework for chat bot.

6. Design and Prototyping Tools

- **Description:** Tools used for designing user interfaces and prototyping.
- **Examples:**
 - **Figma:** Collaborative interface design tool.

1.19.2 Implementation hardware and software equipment

The implementation phase involves deploying the Computer Store Management System with AI Chat Bot into a production environment. This requires specific hardware and software to ensure smooth operation, reliability, and scalability. Below is an outline of the hardware and software equipment needed for the implementation phase.

Implementation Hardware

1. Load Balancers

- **Description:** Devices or software used to distribute incoming traffic across multiple servers to ensure high availability and reliability.
- **Specifications:**
 - **Hardware Load Balancer:** F5 BIG-IP or similar.
 - **Software Load Balancer:** Nginx or HAProxy.

2. Database Servers

- **Description:** Dedicated servers for hosting and managing databases.
- **Specifications:**
 - **Processor:** Intel Xeon or AMD EPYC.
 - **RAM:** 64GB or more.
 - **Storage:** High-capacity SSDs with RAID configuration for data redundancy and performance.
 - **Database Management System (DBMS):** MySQL, PostgreSQL, or MongoDB.

3. Backup and Recovery Systems

- **Description:** Systems used to create backups and ensure data recovery in case of failures.
- **Specifications:**
 - **Backup Storage:** Network-attached storage (NAS) or cloud-based backup solutions.
 - **Backup Software:** Solutions like Veeam Backup & Replication or Acronis Cyber Backup.

Implementation Software

1. Web Servers

- **Description:** Software used to host the web application.
- **Examples:**
 - **Apache:** Open-source web server software.
 - **Nginx:** High-performance web server and reverse proxy.

2. Application Servers

- **Description:** Servers running the application logic and handling client requests.
- **Examples:**
 - **Node.js:** Server-side JavaScript runtime environment.

3. Database Management Systems (DBMS)

- **Description:** Software for managing the production database.
- **Examples:**
 - **MySQL:** Open-source relational database system.

4. Deployment Tools

- **Description:** Tools used to automate the deployment of applications and manage configurations.
- **Examples:**
 - **Docker:** Platform for containerizing applications.
 - **Kubernetes:** Container orchestration platform.

Implementation and Development Report

1.20 Software Development: Testing

1.20.1 Justification of testing technique

Testing was performed using **unit testing**, **integration testing**, **user testing**, and **system testing** for each module of the Computer Shop Management System. Each testing approach was selected based on the specific characteristics and requirements of the different modules, aiming for a comprehensive validation of the overall functionality.

1. Unit Testing

Unit testing was chosen for the **Customer Web Interface Module**, **AI Chatbot for Customer Module**, and **AI Chatbot for Owner Module** to verify the correctness of individual components. Each module was broken down into smaller units, such as form input validation, API response handling, and intent classification. Testing each of these units independently allowed for early detection of errors, ensuring that each feature worked as expected before integrating them into the system.

For instance, in the **Customer Web Interface Module**, unit testing included validating the form inputs, ensuring that each form (e.g., login, product search) handled both valid and invalid data appropriately. This prevented issues such as SQL injection or incorrect search results. Additionally, **API response handling** was evaluated to ensure that all data received from the server (e.g., product details) were displayed correctly, and any errors were handled gracefully.

In the **AI Chatbot for Customer Module**, unit testing was conducted on the **Natural Language Understanding (NLU)** component, specifically to verify the recognition of customer queries and accurate response generation. Boundary value analysis was used to ensure that extreme values, such as product prices at the high and low ends of the range, were recognized correctly. **Equivalence partitioning** was used to group similar intents, such as browsing and purchasing, ensuring the chatbot responded correctly to different yet similar requests.

For the **AI Chatbot for Owner Module**, unit testing focused on the **secure login functionality** and the **report generation component**. Each of these units was tested separately to ensure only authorized access, preventing unauthorized users from accessing sensitive information. Additionally, report generation was tested to ensure accuracy in querying the underlying sales and inventory database, confirming that data retrieval was performed correctly without discrepancies.

Justification: Unit testing allowed for detailed validation of individual functions within each module, ensuring that they worked as intended before proceeding to more complex integration phases. Testing

the boundary values for form inputs and chatbot responses, for instance, ensured robust error handling and validated the system's ability to handle edge cases smoothly. By thoroughly validating each individual component, potential issues were identified and resolved early, thereby enhancing the reliability of the final integrated system.

2. Integration Testing

Integration testing was performed to ensure that individual components worked together seamlessly. This was especially crucial for the **Customer Web Interface Module**, where integrating different services, such as the AI chatbot, product catalog, and checkout module, required seamless communication between these components. Any issues in the integration could lead to poor customer experience, such as incorrect product information being displayed or failure in chatbot interaction. For the **AI Chatbot for Customer Module**, integration testing ensured that the chatbot could correctly pull data from the product catalog and provide accurate recommendations to users. This also involved verifying the proper functioning of the voice input feature, ensuring smooth transcription of user queries and response generation.

For the **AI Chatbot for Owner Module**, integration testing focused on verifying the **secure login functionality**, using data fetched from the owner's database. It was also essential to confirm that the **report generation feature** was compatible with the sales and inventory database, ensuring that the chatbot could generate accurate and timely reports based on user queries.

The integration testing phase utilized a **top-down integration testing** approach, progressively combining individual components starting from the core module. For example, the integration between the **Customer Web Interface** and the **AI Chatbot** was tested to verify the interaction between the web interface and the chatbot's API, ensuring that users could seamlessly access the chatbot from the customer interface. Additionally, the **AI Chatbot for Owner Module** was tested to confirm that the sales data retrieved from the database was correctly processed and presented in an understandable format.

Justification: Integration testing was critical to ensure that the components interacted smoothly. Testing the integration between the web interface and chatbot guaranteed that the user experience was uninterrupted and seamless. The use of top-down integration testing allowed for early identification of interface-level errors and ensured that any issues related to data passing between the components were resolved efficiently.

3. User Testing

User testing was conducted to evaluate the usability and overall experience of the **AI Chatbot for Customer Module**. Participants interacted with the chatbot, and a Google Form was used to gather feedback regarding ease of use, clarity of responses, and overall satisfaction. This feedback was essential

to determine how well the chatbot met customer expectations and to identify any areas that required improvement.

User testing also included validation of the **voice input feature** for the AI Chatbot for Customer. Participants were asked to use the voice input functionality, and feedback was collected to assess how accurately the chatbot transcribed their queries and generated appropriate responses. This testing ensured that the voice input feature worked effectively across different accents and under varying noise conditions, thereby enhancing the chatbot's overall usability.

For the **Customer Web Interface Module**, user testing focused on assessing the system's **navigational flow** and **ease of use**. Users were asked to perform common tasks such as browsing products, adding items to the cart, and initiating a chatbot interaction. Feedback gathered was used to make improvements, such as refining the product search functionality and optimizing page load times.

The **AI Chatbot for Owner Module** was tested with the shop owner and administrative staff. This testing involved generating sales and inventory reports and querying the chatbot for specific information. User feedback focused on the ease of accessing information, report generation speed, and overall satisfaction with the chatbot's ability to provide relevant data. The results were used to make refinements, such as enhancing the accuracy of the query processing logic and improving the user interface for generating reports.

Justification: User testing was essential to evaluate the system from an end-user's perspective and to ensure that each module met the users' needs effectively. Using a Google Form to gather feedback ensured that insights were systematically collected, which allowed for data-driven improvements. The **voice input feature** was a key area of focus, as the accuracy of transcription directly impacted the user experience. Similarly, user testing of the **Owner Module** was critical to confirm that the chatbot could provide timely and accurate information, which was crucial for making informed business decisions.

4. System Testing

System testing was conducted for the **full system**, including all three modules (Customer Web Interface, AI Chatbot for Customer, and AI Chatbot for Owner). This type of testing was done to ensure that the integrated system met the defined requirements and functioned correctly in its entirety. Key features like the complete customer journey—from browsing products to querying orders via the chatbot—and owner-specific operations like sales report generation were evaluated. This level of testing ensured that the entire system operated cohesively and that all modules worked in harmony, without any functional or performance issues.

1.20.2 Test Case Design and Justification

The selection of test cases was driven by the specific features of each module, focusing on critical functionalities that define user interaction, system reliability, and security. The examples presented below illustrate how these test cases were designed to verify various scenarios and their respective behaviors.

Customer Web Interface Module

1. Form Validation Test Cases:

- **Objective:** To ensure that all input fields, such as product search and contact information, are validated for correct and incorrect entries.
- **Test Case Examples:**
 - **Valid Input:**
 - **Data Value:** "MacBook air"
 - **Expected Outcome:** The product search should proceed successfully and display the results.
 - **Invalid Input:**
 - **Data Value:** "####\$@"
 - **Expected Outcome:** An error message such as "Invalid characters used, please enter a valid product name" should be displayed.
 - **Empty Input:**
 - **Data Value:** ""
 - **Expected Outcome:** The system should prompt the user with "Please enter a product name" message.
- **Justification:** This ensures that invalid data is rejected, preventing errors or unexpected behavior on the website and ensuring a robust user experience.

2. Product Search Test Cases:

- **Objective:** To validate the accuracy of the search when customers enter product names or categories.
- **Test Case Examples:**
 - **Full Product Name:**
 - **Data Value:** "Dell Inspiron 15"
 - **Expected Outcome:** The specific product "Dell Inspiron 15" should be displayed.
 - **Partial Product Name:**
 - **Data Value:** "Dell In"

- **Expected Outcome:** All relevant Dell Inspiron models should be displayed, helping the customer narrow their choices.
- **Non-Existent Product:**
 - **Data Value:** "XYZ Laptop"
 - **Expected Outcome:** The system should display "No products found" message and may suggest related options.
- **Justification:** Testing for both correct and incorrect search terms ensure customers receive accurate results and appropriate feedback if a product is not found, improving overall customer satisfaction.

AI Chatbot for Customer Module

1. Intent Recognition Test Cases:

- **Objective:** To evaluate the chatbot's ability to correctly identify user intents, such as browsing products or tracking orders.
- **Test Case Examples:**
 - **Clear Command:**
 - **Data Value:** "Show me gaming laptops"
 - **Expected Outcome:** The chatbot should display a list of gaming laptops available in the store.
 - **Ambiguous Phrase:**
 - **Data Value:** "I need something powerful"
 - **Expected Outcome:** The chatbot should ask follow-up questions, such as "Are you looking for a gaming laptop or a workstation?" to clarify user needs.
 - **Request for Help:**
 - **Data Value:** "Can you help me?"
 - **Expected Outcome:** The chatbot should offer a list of options, such as "I can help you browse products, track your order, or provide store information."
- **Justification:** Ensuring the chatbot accurately detects intents helps deliver relevant responses, especially for ambiguous user requests, thereby enhancing user experience and satisfaction.

2. Voice Input Processing Test Cases:

- **Objective:** To verify that the voice-to-text conversion feature functions as expected across different accents and noise conditions.

- **Test Case Examples:**
 - **Different Accents:**
 - **Data Value:** Query in a strong British accent - "What are the best deals on laptops today?"
 - **Expected Outcome:** Accurate transcription and relevant response listing discounted laptops.
 - **Background Noise:**
 - **Data Value:** Query in a moderate noise environment - "Can I get more details on MacBook Pro?"
 - **Expected Outcome:** The voice input should still be accurately transcribed, and the chatbot should provide information about the MacBook Pro.
 - **Low Voice Input:**
 - **Data Value:** Query in a low voice volume - "Track my order"
 - **Expected Outcome:** The chatbot should accurately recognize and proceed to order tracking.
- **Justification:** This tests the robustness of the voice recognition feature and ensures it functions reliably for a diverse customer base, providing a smooth experience even in real-world noisy environments.

AI Chatbot for Owner Module

1. **Secure Login Test Cases:**
 - **Objective:** To verify that login procedures are secure and only authorized users can access the owner interface.
 - **Test Case Examples:**
 - **Valid Credentials:**
 - **Data Value:** Correct username and password combination.
 - **Expected Outcome:** Successful login and access to the owner dashboard.
 - **Invalid Password:**
 - **Data Value:** Correct username and incorrect password.
 - **Expected Outcome:** "Invalid login credentials. Please try again." message, with no access granted.
 - **SQL Injection Attempt:**
 - **Data Value:** "admin' --"
 - **Expected Outcome:** Login fails, and an error message is displayed, with logs showing that an invalid request was attempted.

- **Justification:** Ensuring secure login is critical for the protection of sensitive business data. By testing various login scenarios, including security threats like SQL injection, the reliability and safety of the system are maintained.

2. Report Generation Test Cases:

- **Objective:** To test the chatbot's ability to generate sales and stock reports.
- **Test Case Examples:**
 - **Short Time frame:**
 - **Data Value:** Sales data for "Last 7 days"
 - **Expected Outcome:** Report generated showing sales for the last week, with accurate sales numbers and inventory status.
 - **Long Time frame:**
 - **Data Value:** Sales data for "Last 12 months"
 - **Expected Outcome:** A detailed report of sales over the year, including a graphical representation for better understanding.
 - **Varying Sales Volume:**
 - **Data Value:** Query report for a period with both high and low sales activity.
 - **Expected Outcome:** Accurate representation of sales trends, clearly indicating periods of high and low activity.
- **Justification:** Ensuring the reporting system can handle various data sizes and generate correct reports is essential for informed business decision-making. It allows the shop owner to better manage inventory and plan sales strategies effectively.

1.20.3 Test logs

Here is the complete table for the test logs, incorporating all relevant details for each test case conducted across the three modules of the system:

Table 11: Test Logs

Test Case ID	Description	Data Values	Module	Function Under Test	Expected Result	Actual Result	Conclusion
TC01	Form Validation - Product Search	"Laptop", "@@@", "xYz123"	Web Interface Module	Input Validation	Only valid product names accepted	Incorrect inputs rejected	Successful
TC02	User Authentication - Password Reset	Valid and invalid email	Owner Chatbot	Password Reset	Password Reset	Failed	Failed
TC03	Voice Input Processing - Product Recommendations	Voice query: "Recommended a laptop"	Customer Chatbot	Voice-to-Text Conversion	Accurate transcription and response	Correct transcription and response	Successful
TC04	Integration - Web and Chatbot Interaction	Search product, chat for details	Web Interface Module	Web-Chatbot Communication	Smooth integration, chatbot response visible	Integration successful	Successful
TC05	Secure Login Verification	Valid and invalid credentials	Owner Chatbot	Secure Access	Correct credentials allowed, unauthorized attempts blocked	Unauthorized access blocked	Successful
TC06	Report Generation - Monthly Sales Data	Request sales report for March	Owner Chatbot	Data Retrieval	Report generated with correct details	Correct report generated	Successful
TC07	User Testing - Voice Input	Google Form feedback	Customer Chatbot	Usability Assessment	Positive responses about accuracy	Voice input rated highly	Successful, with minor improvements required
TC08	System Testing - Full System Evaluation	Complete customer journey	System (All Modules)	End-to-End Functionality	All components	Smooth operation of all modules	Successful

Test Case ID	Description	Data Values	Module	Function Under Test	Expected Result	Actual Result	Conclusion
					work seamlessly		
TC09	Product Search - Invalid Entry	Empty value	Web Interface Module	Product Search	Prompt user for valid input	Prompt displayed correctly	Successful
TC10	Product Search - Partial Match	"HP Pavilion"	Web Interface Module	Product Search	Display related products	Correct related products shown	Successful
TC11	Intent Recognition Ambiguous	"I need something fast"	Customer Chatbot	Intent Classification	Clarify with user about preferences	Clarification question asked	Successful, additional intents added for handling ambiguity
TC12	Voice Input Background Noise	Query with background music	Customer Chatbot	Voice-to-Text Conversion	Correct transcription, relevant response	Transcription was mostly accurate	Successful, minor issues in noisy environments
TC14	Inventory Update - Adding Product	New Product Details	Web Interface Module	Inventory Management	Product added to inventory	Product successfully added	Successful
TC15	Integration Chatbot and Database	Query for product stock	Customer Chatbot	Chatbot-Database Connection	Correct stock level shown	Accurate stock level displayed	Successful
TC16	Checkout Process Validation	Multiple items in cart	Web Interface Module	Checkout Process	Successful checkout, correct total calculation	Checkout completed, correct total	Successful
TC17	User Testing Navigation	Google Form feedback	Customer Chatbot	Usability Assessment	Ease of navigation feedback gathered	Positive feedback with suggestions for improvement	Successful, suggested improvements logged for future development
TC18	Report Generation - Low Sales Data	Sales report for low activity period	Owner Chatbot	Data Retrieval	Report generated with accurate, low sales data	Correct report generated	Successful

Test Case ID	Description	Data Values	Module	Function Under Test	Expected Result	Actual Result	Conclusion
TC19	System Testing - Database Consistency	Product order, inventory check	System (All Modules)	Database Consistency Check	Inventory adjusted after order	Inventory correctly updated	Successful
TC20	Integration - Owner Chatbot and Sales DB	Request for profit margins report	Owner Chatbot	Report Generation	Report generated correctly	Correct margins report displayed	Successful
TC21	Payment Gateway Integration	Valid and Invalid Payment Details	Web Interface Module	Payment Processing	Accept valid payments, reject invalid ones	Payments processed failed	Failed
TC22	Error Handling - Incorrect Inputs	Negative price entry	Owner Chatbot	Data Validation	Reject negative values	Correctly rejected	Successful
TC23	Cross-browser Testing	Access from different browsers (Chrome, Firefox, Safari)	Web Interface Module	Compatibility	Consistent behavior across browsers	Consistent performance	Successful
TC24	Chatbot Response Time	Various user queries	Customer Chatbot	Response Time Measurement	Response time within acceptable limits	Average response time within limits	Successful, monitoring suggested for ongoing performance

Below is a table summarizing the customer chatbot intent test results:

Table 12: Customer Chatbot Intents Test Results

Intent Name	Precision	Recall	F1-Score	Support	Confused With
inquire_gaming_laptops	1.0	1.0	1.0	9	None
inquire_best_brands	1.0	1.0	1.0	10	None
asking_about_shop	1.0	1.0	1.0	15	None
item_price	1.0	1.0	1.0	9	None
asking_about_discounts	1.0	0.89	0.94	9	asking_about_bulk_discounts (1)
bot_challenge	1.0	1.0	1.0	16	None
inquire_review_submission	1.0	1.0	1.0	10	None
ask_available_one_item	1.0	1.0	1.0	9	None
inquire_laptop_accessories	1.0	1.0	1.0	10	None
inquire_order_delivery_time	1.0	1.0	1.0	10	None
asking_home_delivery	1.0	1.0	1.0	15	None
inquire_refurbished_computers	1.0	1.0	1.0	10	None
ask_available_items	1.0	1.0	1.0	15	None
inquire_gift_cards	1.0	1.0	1.0	10	None
asking_how_to_claim_warranty	1.0	1.0	1.0	15	None
Greet	1.0	1.0	1.0	10	None
ask_discount_on_item	0.9	1.0	0.95	9	None
Goodbye	1.0	1.0	1.0	10	None
asking_about_customer_support	1.0	1.0	1.0	15	None
asking_about_location	1.0	1.0	1.0	14	None
ask_repairs	1.0	1.0	1.0	14	None
ask_item_pic	1.0	0.86	0.92	7	ask_discount_on_item (1)
Deny	1.0	1.0	1.0	7	None
ask_warranty_on_item	1.0	1.0	1.0	9	None
inquire_privacy_policy	1.0	1.0	1.0	10	None
asking_capabilities	1.0	1.0	1.0	15	None
asking_about_exchange_product	1.0	1.0	1.0	15	None
inquire_cctv_price	1.0	1.0	1.0	12	None
inquire_budget_computers	1.0	1.0	1.0	10	None
asking_help	1.0	1.0	1.0	15	None
inquire_desktop_for_graphic_design	1.0	1.0	1.0	10	None
inquire_computer_types	1.0	1.0	1.0	7	None
inquire_best_laptops_for_students	1.0	1.0	1.0	10	None
asking_about_working_hours	1.0	1.0	1.0	15	None
inquire_cctv_operation	1.0	1.0	1.0	5	None

Intent Name	Precision	Recall	F1-Score	Support	Confused With
inquire_shipping_cost	1.0	1.0	1.0	10	None
inquire_antivirus_software	1.0	1.0	1.0	10	None
ask_description_of_item	1.0	1.0	1.0	9	None
inquire_custom_pc_build_time	1.0	1.0	1.0	10	None
inquire_custom_computer	1.0	1.0	1.0	10	None
asking_about_bulk_discounts	0.94	1.0	0.97	15	None
inquire_cctv_installation	1.0	1.0	1.0	5	None
find_components	1.0	1.0	1.0	15	None
inquire_custom_pc_build	1.0	1.0	1.0	10	None
asking_about_return_policy	1.0	1.0	1.0	15	None
Affirm	1.0	1.0	1.0	6	None
asking_about_payments	1.0	1.0	1.0	15	None

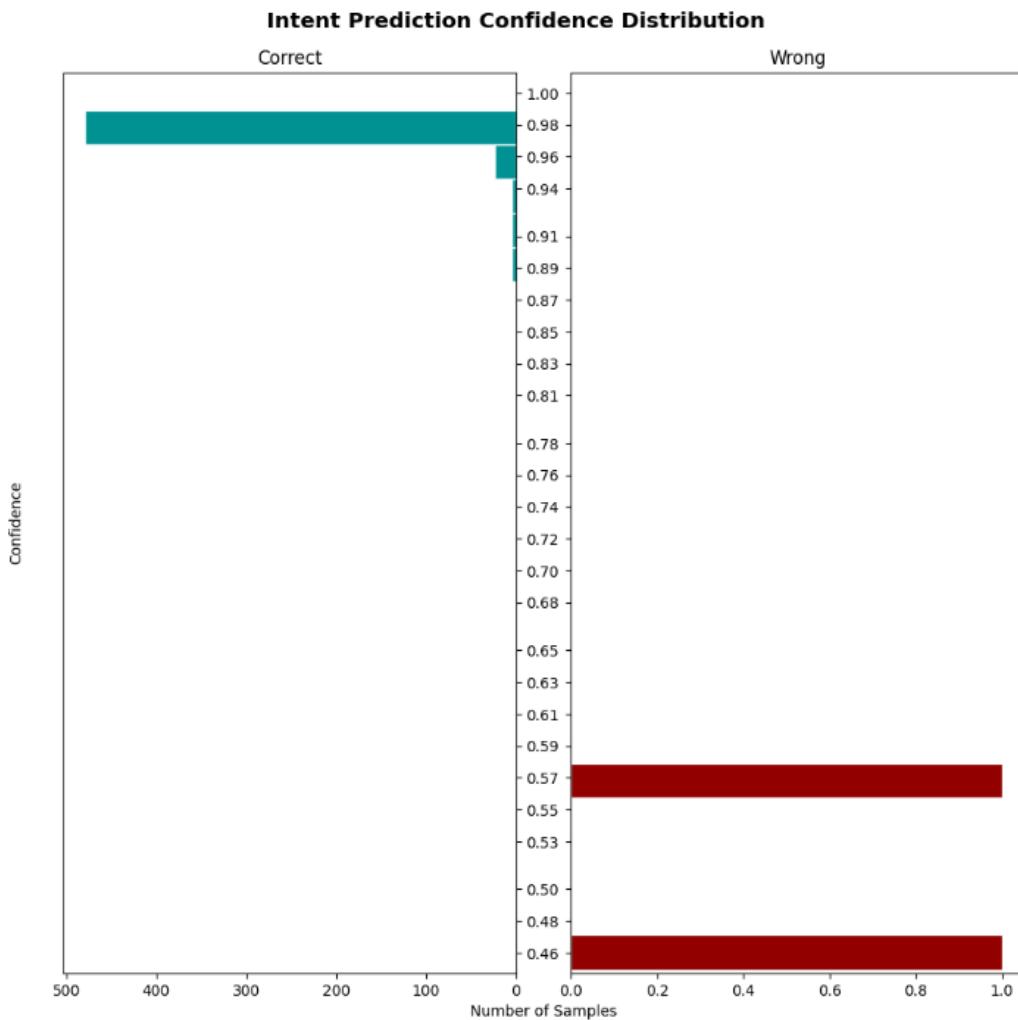


Figure 33: Intent Prediction Confidence Distribution

Key Metrics Explained:

1. Precision:

- Indicates how many predictions of a specific intent were correct.
- A high precision value (close to 1.0) suggests the model makes very few false-positive errors.

2. Recall:

- Shows how many real instances of an intent were correctly identified.
- A high recall value (close to 1.0) means the model captured all instances, with few false negatives.

3. F1-Score:

- The harmonic mean of Precision and Recall, providing a balanced view.
- A high F1-Score indicates a good trade-off between precision and recall, meaning the model performs well overall.

4. Support:

- The number of test examples for each intent.
- It helps determine the amount of data used to calculate metrics for each intent, giving confidence in the reported performance.

5. Confused With:

- Indicates if any intents were incorrectly predicted as others.

Overall Performance Metrics

Table 13: Overall Performance of Intent Test Result

Metric	Value
Accuracy	99.62%
Macro Average	Precision: 0.997
Weighted Average	Precision: 0.997

Key Observations:

- Most intents achieved 100% precision, recall, and F1-score, indicating excellent performance.
- Minor confusion occurred with similar intents: "asking_about_discounts" was confused once with "asking_about_bulk_discounts," and "ask_item_pic" was confused once with "ask_discount_on_item.". Some intents, like "ask_discount_on_item" and "asking_about_bulk_discounts," have slightly lower F1-scores but still exhibit high accuracy.

Below are some of customer chatbot user test results:

How easy was it to interact with the chatbot?

20 responses

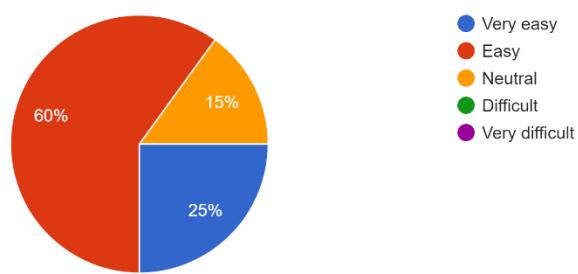


Figure 34: Question 1

How clear and readable was the text on the chatbot's interface?

20 responses

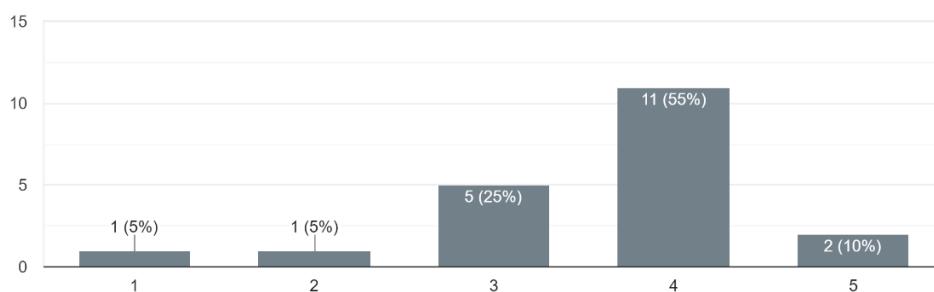


Figure 35 : Question 2

Did you experience any technical issues while interacting with the chatbot? (e.g., slow response time, errors)

20 responses

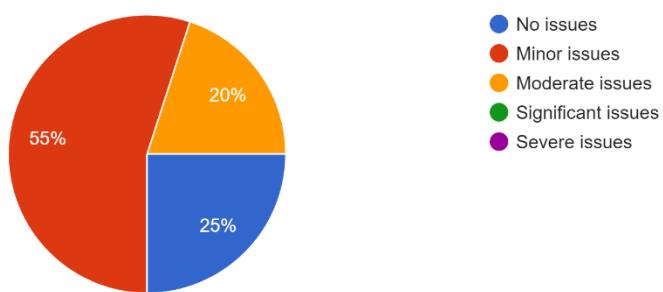


Figure 36 : Question 3

Note:

For a detailed breakdown of each test case, including specific data values, test procedures, and further analysis, please see the appendix.

1.21 Implementation: Conversion and Training Plan

1.22 Justification of conversion technique

The chosen conversion technique is a **Phased Conversion**.

- The system will be implemented in **phases**, starting with the **Customer Web Interface**, followed by the **Customer AI Chatbot**, and finally the **Owner AI Chatbot**.
- This approach allows each component to be tested and integrated sequentially, reducing risks. Phased conversion is also appropriate given the interdependencies between modules, allowing gradual adaptation for both customers and staff.

1.23 Schedule and Planning for Conversion

The **Gantt chart** below outlines the implementation schedule:

Table 14: Implementation Schedule

No	Task	Time							
		Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8
1	Integration of Customer AI Chatbot and testing								
2	Deployment of Customer Web Interface.								
3	Integration of Owner AI Chatbot.								
4	Training sessions for staff on using the Owner AI Chatbot.								
5	Full system testing and user feedback phase.								

The implementation of the Computer Shop Management System, as outlined in the Gantt chart, involves several key phases distributed across eight weeks. Each phase requires specific resources, including personnel, hardware, software, and designated locations, to ensure a successful and smooth deployment. Below, the narrative is divided into sections that correspond to each phase of the Gantt chart timeline.

Week 1-2: Integration of Customer AI Chatbot and Testing

The initial phase focuses on the integration of the **Customer AI Chatbot** into the web interface and conducting comprehensive testing to ensure seamless performance.

Personnel Involved:

- The **development team** plays a vital role in this phase, leveraging their expertise in AI and software development. Their tasks include setting up the chatbot, connecting it to the web interface via APIs, and optimizing the chatbot's responses for different customer intents and entities.
- **IT staff** assist in monitoring technical aspects and ensuring performance standards are met during testing.

Hardware Resources:

- **Cloud servers** are used to host the chatbot, which is essential for managing customer interactions effectively and processing the natural language understanding (NLU) component.

Software Setup:

- Installation packages and APIs are used to connect the chatbot to the web interface. The software setup also includes configuring the **voice input** feature, offering an alternative for customers preferring voice over text interactions.

Testing Procedures:

- The integration testing is performed using both **manual and automated tests**, simulating various customer interactions to evaluate the chatbot's responses. Testing includes handling FAQs and ensuring the chatbot can recognize voice commands appropriately.
- The IT staff perform performance monitoring, validating that the chatbot responds accurately to different types of inquiries.

Week 3: Deployment of Customer Web Interface

The third week is designated for the **deployment of the Customer Web Interface**, providing customers with access to the system.

Personnel Involved:

- The **development team**, in coordination with **IT staff**, takes responsibility for deploying the web interface on the cloud.
- IT staff conduct follow-up testing to confirm that the system is functioning as intended.

Hardware Resources:

- Deployment utilizes **cloud-based virtual machines** and **scalable storage solutions** to ensure the web interface can handle high user volumes and maintain reliability during peak traffic.

Software Setup:

- The **React framework** is installed to build a responsive user experience for customers. Deployment scripts are executed for configuring the web server, installing dependencies, and ensuring smooth operation.
- **DNS configurations** are managed to provide customers with access to the system, while secure protocols are established to protect user data during transactions.

Deployment Testing:

- After deploying the interface, the IT staff perform **stress testing** to assess how well the system handles large numbers of concurrent users. Any discrepancies are documented and resolved promptly.

Week 4-5: Integration of Owner AI Chatbot

Weeks four and five are reserved for the **integration of the Owner AI Chatbot**, which provides the shop owner with essential business insights.

Personnel Involved:

- The **development team** integrates the chatbot into the administrative interface. This integration includes deploying the AI model and enabling data retrieval for inventory and sales.
- The **administrative staff** and **shop owner** contribute feedback to ensure that the chatbot meets the operational needs of the business.

Hardware Resources:

- **Dedicated servers** host the Owner AI Chatbot, providing high performance and reliable data processing for sales and inventory management.

Software Setup:

- Integration involves using **APIs** to facilitate data retrieval from inventory and sales databases. The software setup ensures that the chatbot can effectively respond to typical business queries, like identifying low stock levels or generating sales summaries.

Testing and Feedback:

- Mock interactions with the **administrative staff** are conducted to verify the chatbot's performance and usability. The feedback gathered is used to improve the system and refine its functionality, ensuring it meets all business requirements.

Week 6: Training Sessions for Staff on Using the Owner AI Chatbot

In the sixth week, training sessions are organized for the **shop owner** and **administrative staff** on using the Owner AI Chatbot.

Personnel Involved:

- **IT staff** and the **development team** conduct training, providing hands-on demonstrations and guided practice sessions.

Hardware Resources:

- Training requires **computers** equipped with the necessary software for accessing and interacting with the chatbot interface.

Training Process:

- Participants are guided through the process of logging into the Owner AI Chatbot using the secure link **computershop-owner.azurewebsites.net**. The training focuses on the different functionalities available, such as querying inventory and sales data.
- To support training, **user manuals** and **tutorial videos** are provided, which the administrative staff can use as reference materials for continued learning.

Support Resources:

- The IT staff establishes a helpdesk service to address any questions or technical issues that may arise during or after training, ensuring staff can fully utilize the chatbot.

Week 7-8: Full System Testing and User Feedback Phase

The final phase, extending across weeks seven and eight, includes **full system testing** and collecting **user feedback** to implement final adjustments.

Full System Testing:

- This stage involves a complete end-to-end evaluation of all modules: the **Customer Web Interface**, **Customer AI Chatbot**, and **Owner AI Chatbot**. The **development team** and **IT staff** collaborate on testing the system under real-world conditions, ensuring smooth interaction between all modules.

Personnel Involved:

- The **shop owner** and **administrative staff** actively participate in testing and provide essential **user feedback** on their experiences.

Hardware and Software Setup:

- The **cloud servers** used for hosting the chatbots and web interfaces undergo thorough testing to ensure they meet performance and reliability standards.
- The IT staff also configure **Azure Monitor** to provide continuous alerts in the event of system downtime or other issues, enabling rapid responses to maintain system availability.

User Feedback Collection:

- Feedback from the **shop owner** and **administrative staff** is gathered through structured forms and interviews. The feedback process focuses on evaluating usability, performance, and overall satisfaction.
- Based on the collected feedback, the development team makes necessary refinements, such as optimizing the chatbot's responses or enhancing the user interface for better navigation.

Error Handling Testing:

- Special attention is given to testing **error handling** scenarios, such as incorrect login attempts and insufficient stock during purchases. This step ensures that users receive appropriate notifications and suggestions in case of errors, maintaining a high-quality user experience.

Final Evaluation:

- A **final performance evaluation** is conducted to confirm that all functionalities operate efficiently, data is processed correctly, and any identified issues have been resolved. The system is thoroughly vetted to ensure a stable and secure deployment before going live.

1.24 Implementation: User Guide

3.1 Installation and Setup Instructions

The implementation of the system involves deploying three modules, stored on a portable hard drive and hosted on Azure. The following guide includes the setup for each module and how to deploy them using Azure resources.

Step 1: Prerequisites

1. Hardware Requirements:

- Portable Hard Drive containing the modules.
- A computer or server for initial setup.

2. Software Requirements:

- **Node.js** for running the React web interface.
- **Python** for AI Chatbot modules (Rasa and Streamlit).
- **Azure CLI** for deployment and management of Azure resources.
- **Docker** (optional) for containerizing the application.
- **Azure Subscription** for cloud hosting.

Step 2: Setting Up the Modules from Portable Hard Drive

1. Connect the Portable Hard Drive to your computer/server where initial setup will be done.

2. Copy the Modules:

- Create a dedicated folder for the project on your system.
- Copy the following from the portable hard drive:
 - **Customer Web Interface Module** (React).
 - **AI Chatbot for Customer Module** (Rasa, HTML, CSS, JS).
 - **AI Chatbot for Owner Module** (Streamlit and Ollama).

3. Ensure the copied files and directories are properly structured as per module requirements.

Step 3: Setting Up Azure Resources

1. Login to Azure Portal:

- o Open the [Azure Portal](#).
- o Sign in using your Azure credentials.

2. Azure CLI Installation and Configuration:

- o Download and install [Azure CLI](#).
- o Authenticate by running the following command in your terminal:

```
az login
```

3. Creating Azure Resources:

- o **Resource Group:** Create a resource group to contain the resources:

```
az group create --name ComputerShopRG --location eastus
```

- o **Virtual Machine (VM) Setup** (for backend processing and hosting the modules):

- Create a Virtual Machine:

```
az vm create --resource-group ComputerShopRG --name ShopVM --image
UbuntuLTS --admin-username azureuser --generate-ssh-keys
```

Note the **public IP address** assigned to the VM.

- o **Azure Storage Account:**

- Create a storage account to store data related to the application:

```
az storage account create --name computeshopstorage --resource-
group ComputerShopRG --location eastus --sku Standard_LRS
```

Step 4: Deploying Customer Web Interface Module

1. Setup Node.js Environment:

- o Connect to the Azure VM using SSH:

```
ssh azureuser@<public_ip_address>
```

- Navigate to the directory containing the Customer Web Interface files.
- Install dependencies:

```
cd customer-web-interface  
npm install
```

2. Run the React Application:

- Start the React server:

```
npm start
```

- **Configure Azure App Service** (optional) to host the React app permanently.

Step 5: Deploying AI Chatbot for Customer Module (Rasa)

1. Setup Python Environment:

- On the Azure VM, navigate to the directory containing the Rasa chatbot.
- Create a virtual environment:

```
python3 -m venv chatbot-env  
source chatbot-env/bin/activate
```

- Install required dependencies:

```
pip install -r requirements.txt
```

- Train the Rasa chatbot using:

```
rasa train
```

2. Run the Chatbot:

- Start the Rasa server:

```
rasa run -m models --enable-api --cors "*"
```

- Start the Rasa action server for custom actions:

```
rasa run actions
```

Step 6: Deploying AI Chatbot for Owner Module (Streamlit & Ollama)

1. Setup Python Environment for Streamlit:

- On the Azure VM, navigate to the folder containing the owner chatbot.
- Create a new virtual environment:

```
python3 -m venv owner-chatbot-env  
source owner-chatbot-env/bin/activate
```

- Install Streamlit:

```
pip install streamlit
```

- Install additional dependencies (including Ollama SDK).

2. Run Streamlit Application:

- Start the Streamlit server:

```
streamlit run app.py
```

Step 7: Setting Up Continuous Deployment Using Azure DevOps

1. Azure DevOps Configuration:

- Create a new project in [Azure DevOps](#).
- Connect your project repository for continuous deployment.
- Create a **pipeline** to automatically deploy updates made to the modules:
 - Install Azure CLI and configure scripts to deploy on code changes.
 - Set triggers for automatic builds and deployments.

2. Deployment Script:

- Write a `deploy.sh` script that updates modules on Azure VM whenever changes are pushed:

```
#!/bin/bash  
cd /path/to/module  
git pull origin main  
npm install && npm run build
```

Step 8: Securing Azure Resources

- 1. Network Security Group (NSG):**
 - Set up an NSG to allow only required traffic (such as HTTP, HTTPS, and SSH).
 - Restrict ports to prevent unauthorized access.
- 2. SSL/TLS Certificate Setup:**
 - Use **Azure Key Vault** to generate an SSL/TLS certificate.
 - Bind the certificate to your deployed applications to secure connections.

Step 9: Testing and Verification

- 1. Test Customer Interface:**
 - Verify if customers can successfully access the React application from a browser.
 - Test all functionalities, including browsing products and accessing the customer chatbot.
- 2. Test Owner Chatbot:**
 - Log into the Streamlit application and verify the functionalities, such as report generation and data querying.
- 3. User Testing:**
 - Ensure users (customers and shop owner) can access their respective modules without issues.
 - Record feedback for future improvements.

Step 10: Backup and Recovery Plan

- 1. Azure Backup Configuration:**
 - Configure **Azure Backup** to back up the virtual machine and storage account regularly.
 - Set up a **recovery services vault** for long-term backup retention.
- 2. Local Backup:**
 - Keep an additional backup of the application files on the portable hard drive in case of cloud failure.

Step 11: Final Configuration for Live Usage

- 1. Public DNS Setup:**
 - Assign a **DNS name** to the public IP address of the Azure VM for easier access (e.g., `computershop.azurewebsites.net`).

2. Monitoring and Alerts:

- Configure **Azure Monitor** to set up alerts for application downtime or performance issues.
- Ensure real-time monitoring is enabled for quick response to any incidents.

3.2 Operating Instructions

Customer Web Interface Module:

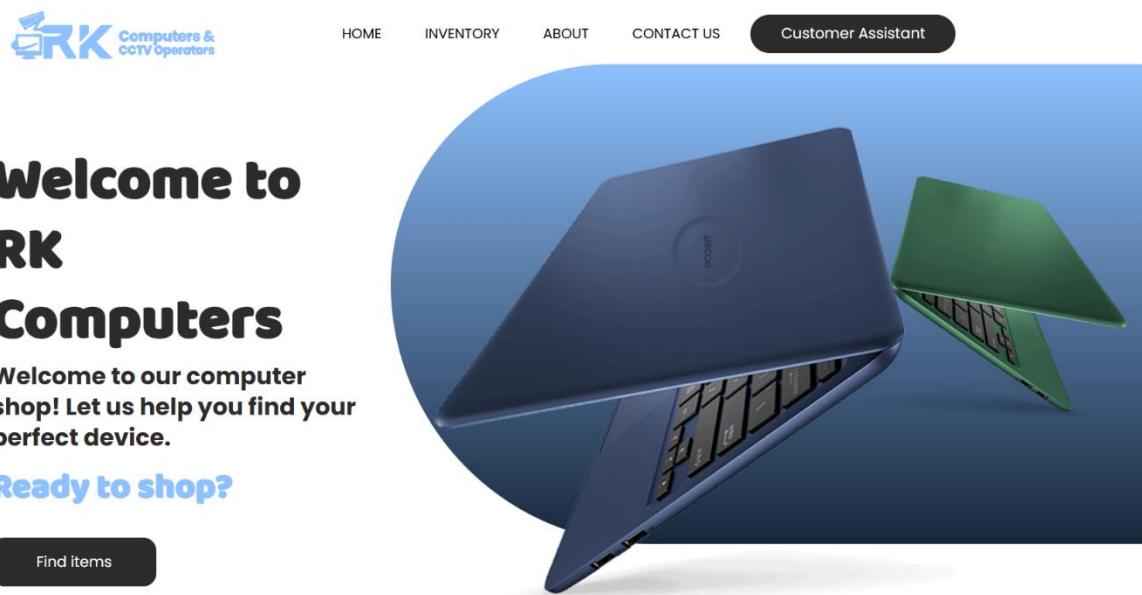


Figure 37: Web Interface UI

- **Access:**

- Customers can access the computer shop's web interface by entering the assigned **DNS link** in their preferred web browser. The link will direct them to the homepage of the shop, where they can explore all available features.
- The homepage will present key options like **Browse Products**, **Offers**, , and the chatbot feature.

- **Interaction:**

- Customers can **browse** the available products, use **filter options** to narrow down product categories (such as brand, price, and specifications), and perform **searches** using keywords.
- Clicking on any product will redirect users to a detailed **product description page**, where they can view the specifications, features, reviews, and ratings of the product.

- To initiate a conversation with the AI Chatbot for assistance, customers can click on the **customer chat link** located in the top-right corner of the screen. This will redirect to customer chatbot interface, and customers can start typing their questions directly to interact with it.

AI Chatbot for Customer Module:

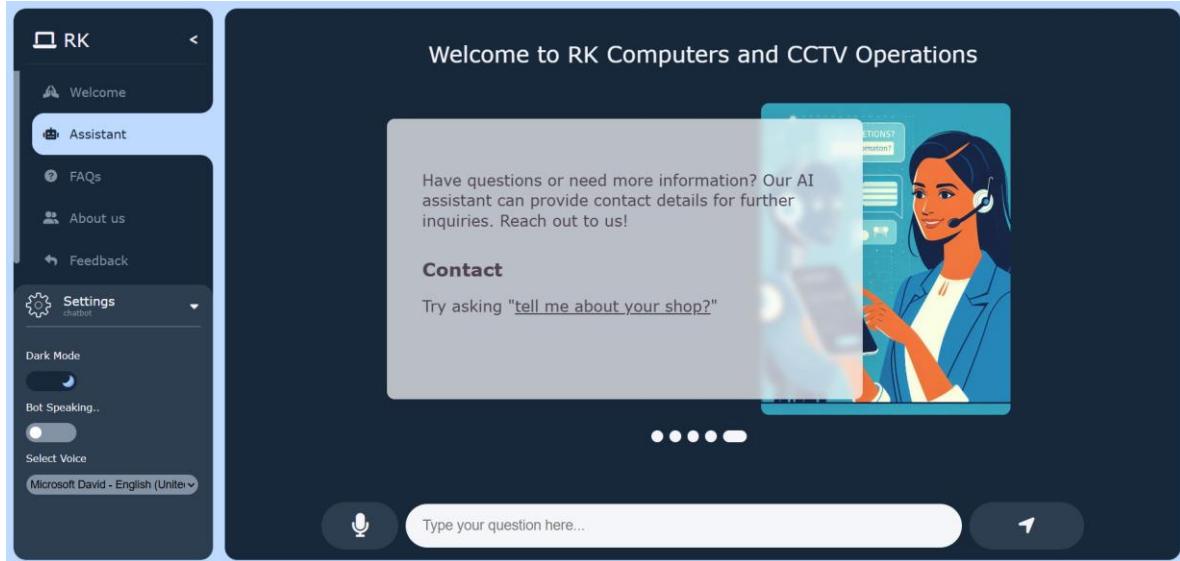


Figure 38: Customer Chatbot User Interface

Customer Care:

- The AI Chatbot can assist customers by answering **product-related questions**. Customers can type inquiries or use the **voice input feature** to ask questions directly. To use voice input, click on the **microphone icon** in the chatbot window and speak your query, such as "What are the features of the latest laptop?" or "What is the price of the intel i3?" The chatbot will process the voice input and respond accordingly.
- **Recommendations:**
 - Customers can interact with the chatbot to receive **personalized recommendations** based on their preferences using either text or **voice input**. For example, they can say, "I'm looking for a laptop for gaming," and the chatbot will process the voice command and provide relevant product options.

AI Chatbot for Owner Module:



Figure 39 : Owner chatbot User Interface

- **Access:**

- Owners can log in to the **Owner Chatbot Module** by accessing the secure link (<https://computershop-owner.azurewebsites.net>) in a browser.
- Upon opening the link, the owner will be presented with a **login page**. Owners need to enter their credentials (username and password) to gain access.

- **Data Retrieval:**

- Once logged in, owners can use the interface to **generate reports** related to sales, such as **daily, weekly, or monthly sales reports**. Owners can select a date range and view comprehensive sales summaries, including product performance and revenue data.
- Owners can also **query specific information** using a simple question interface. For example, owners may ask, "What is the current stock level for gaming PCs?" or "How many units of Product X have been sold this month?" and the system will provide the required data instantly.
- The system will allow **exporting data** as spreadsheets for further analysis or record-keeping.

3.3 Description of Main Functions

- **Product Browsing:**

- Customers visiting the web interface will have the ability to **browse the product catalog** through a user-friendly interface. The catalog is organized into different categories, such as **desktops, laptops, peripherals, and accessories**.
- Each product page contains **comprehensive product details**, including key specifications, images, customer reviews, and the product's availability status. Customers can use **filtering options** to refine their search based on product features, pricing, or brands.
- The **search function** allows customers to enter keywords to quickly find specific items in the inventory. Search suggestions will appear as they type, improving ease of access.

- **AI Assistance:**

- The **Customer Chatbot** helps using either text input or **voice input**. Customers can click on the **microphone icon** in the chatbot window to ask questions by speaking. The voice input feature uses **speech-to-text technology** to convert the spoken words into text, which is then processed using **natural language processing (NLP)**.
- For **product recommendations** and assistance with orders, customers can choose to either type or speak their requests, making the interaction more convenient and accessible.

- **Owner Operations:**

- The **Owner Chatbot** enables the shop owner to log in to a secure interface to access key operational features of the computer shop.
- Owners can **generate detailed reports** on sales performance, including visual representations like **charts and graphs**. These reports are instrumental for **data-driven decision-making** and strategic planning.
- The owner module also provides the capability to **query specific inventory data**. For instance, owners can check stock levels for specific products or generate **inventory summaries** to assist in replenishment decisions.

3.4 General Error Handling

1. Customer Errors:

- **Insufficient Stock Notification:**
 - If a customer attempts to add a product to their cart that is **out of stock**, a **notification** will be displayed to inform them that the item is not available. The message will provide the expected restock date, if available, and offer to **notify the customer** when the item is back in stock.
 - Example: "The item you selected is currently out of stock. Expected restock date: 05/15/2024. Would you like to receive an alert when it is available?"
- **Incorrect Login Attempts:**
 - If a customer or owner attempts to log in with incorrect credentials, an **error message** will appear specifying that the username or password is incorrect. The error message will include a link to reset the password if needed.
 - After **multiple failed login attempts**, the system will temporarily **lock the account** and send an email to the registered address with information on how to unlock it.

2. Chatbot Fallback:

- **Fallback Responses:**
 - If the chatbot cannot understand a query, it will provide a **fallback response**. The response will include **suggestions for rephrasing the question** or **display common questions and topics** to help guide the customer.
 - Example: "I'm sorry, I didn't quite understand your question. You can try asking about our products or your order status. Here are some examples of questions I can answer: 'What is the price of Laptop X?'"
- **Escalation to Human Representative:**
 - If the customer continues to face issues or if the chatbot is unable to handle the request, the system will **escalate the query to a human representative**. A message will be displayed to inform the customer that their request is being forwarded for further assistance.

3. System Failures:

- **Error Page Display:**
 - In the event of **system downtime** or **backend errors**, the customer will be directed to a generic **error page** indicating that the system is temporarily unavailable. The page will contain a **friendly message** reassuring the customer and suggesting they try again later.
 - Example: "We're experiencing technical difficulties at the moment. Please try again later. We apologize for any inconvenience."
- **Azure Monitor Alert:**
 - When a system failure occurs, an **alert is sent via Azure Monitor** to notify the development and IT team about the failure. This ensures that the issue is addressed promptly, minimizing downtime.
 - Azure Monitor is configured to detect unusual system behaviors, such as **high server load** or **failure to respond** within a specified time and sends an alert via email or SMS to the responsible personnel.
- **Automatic Failover (Optional):**
 - To maintain **high availability**, a failover mechanism can be configured on Azure. If the primary server goes down, the system will automatically switch to a backup server to ensure the website and chatbot remain accessible to customers. This is critical for maintaining **customer satisfaction** and preventing disruption in services.

1.25 Critical Appraisal: combined

The **Computer Shop Management System** has successfully enhanced both customer engagement and administrative efficiency through its integrated modules, comprising the **Web Interface Module**, the **AI Chatbot for Customer Module**, and the **AI Chatbot for Owner Module**. Each module brings specific value to the overall system, delivering distinct functionalities to support both customers and the shop owner effectively.

However, there were a few notable challenges encountered during development and integration that warrant attention for future improvements. One of the primary issues faced was the **integration of the AI chatbot** and ensuring smooth and accurate responses to customer inquiries. The chatbot's ability to handle **ambiguous queries** proved to be challenging, largely due to the complexity involved in natural language processing. Ambiguity in customer questions, such as vague descriptions or multiple meanings, led to instances where the chatbot either provided irrelevant responses or failed to answer effectively.

Another area identified for future enhancement is the system's **information retrieval capabilities**. Currently, the chatbot uses standard database querying mechanisms, which are limited when it comes to understanding complex relationships between various product attributes. A **knowledge graph** could be introduced to improve the chatbot's response accuracy and relevance. Knowledge graphs are known for their ability to establish connections between different data points, enabling the system to provide more nuanced and informative responses. This improvement would enhance customer satisfaction by enabling more detailed, context-aware answers.

Further, while the current system supports **English** for interactions, expanding to provide **multi-language support** would significantly broaden the system's reach and usability. As a computer shop, a diverse customer base could benefit from interaction in languages other than English. Introducing multi-language support will require training the chatbot using datasets that cover various languages, ensuring accurate translation and response generation. This development would also involve enhancing the **text-to-speech** and **speech-to-text** systems to work effectively across languages, thereby improving the overall customer experience.

Moreover, the system's **scalability** and **maintainability** were critical considerations during implementation. While the current infrastructure, hosted on **Azure**, provides a robust platform for deployment, future scalability might be required as the shop grows or additional features are introduced. To manage increasing user loads, the system may require optimized resource allocation and auto-scaling configurations to ensure performance and reliability.

The **customer data privacy** aspect is another area where continued improvements could enhance system integrity. While basic encryption mechanisms have been implemented, considering more advanced **data protection techniques** such as **end-to-end encryption** and **data anonymization** would provide an added layer of security, reassuring customers that their data is protected.

Overall, the **Computer Shop Management System** marks a significant improvement in managing customer and administrative interactions, laying a solid foundation for the future. Continued development focusing on addressing these challenges will make the system more robust, versatile, and user-friendly.

1.26 Critical Appraisal: individual

5.1 Web Interface Module

The **Web Interface Module** plays a crucial role in bridging the gap between users—both customers and shop staff—and the system itself. This module provides the graphical user interface, ensuring customers have a user-friendly and engaging platform for interaction.

One of the major challenges faced during the development of this module was achieving **responsiveness across different devices**. Given the range of devices available today—from desktops and laptops to tablets and smartphones—ensuring a seamless experience was a significant hurdle. The module initially showed inconsistencies in layout and functionality when accessed from various screen sizes and resolutions, which required additional development time for debugging and testing.

To address this, a **mobile-first approach** was adopted during the later stages of development. However, despite these improvements, there are still areas where **cross-device compatibility** could be further optimized. Future efforts should focus on incorporating more rigorous **UI testing frameworks** to enhance the module's compatibility across devices. Automated tools such as **Selenium** and **Cypress**, combined with manual testing, could help identify issues early in the development cycle, allowing for more efficient resolution.

Another area of improvement for the Web Interface Module lies in **accessibility**. Ensuring that the interface is accessible to individuals with disabilities was considered, but the implementation could be further strengthened by adopting **Web Content Accessibility Guidelines (WCAG)**. This would involve adding features such as **keyboard navigation**, **screen reader compatibility**, and providing **alternative text** for images. Accessibility is key to ensuring that the platform is usable for all customers, including those with visual or motor impairments, which could expand the customer base significantly.

In summary, the Web Interface Module successfully provides a functional and user-friendly interface but requires ongoing enhancements in **responsiveness**, **compatibility**, and **accessibility** to reach its full potential.

5.2 AI Chatbot for Customer Module

The **AI Chatbot for Customer Module** was designed to offer real-time assistance to customers, providing answers to product inquiries, personalized recommendations, and guidance during the purchase process. The chatbot's ability to use both **text** and **voice input** made the interaction versatile and engaging for customers.

A major challenge encountered during the development of this module was the **quality of training data** used to train the chatbot's machine learning models. The initial dataset consisted of predefined questions and responses, which limited the chatbot's effectiveness in handling **unforeseen queries**. As a result, the chatbot struggled to understand customer questions that were phrased in unconventional ways or included colloquial language.

To overcome this challenge, the training dataset was expanded to include more diverse user queries and scenarios. However, there is still room for improvement. Moving forward, efforts should focus on collecting and incorporating **real user interactions** into the training data. This could be done by **logging queries** that the chatbot fails to answer and using these logs to refine the model's understanding. Additionally, implementing a **feedback loop** where customers can rate the responses would provide valuable insights into areas where the chatbot needs improvement.

The **voice input** feature also presented unique challenges, such as accurately converting spoken language into text and managing different accents or speech impediments. Future iterations could benefit from integrating more advanced **speech recognition services**, such as those offered by **Google Cloud Speech-to-Text** or **Microsoft Azure Speech Services**, which are designed to handle a wide range of accents and dialects.

In conclusion, while the AI Chatbot for Customer Module is functional and has significantly improved customer service, enhancements in **training data diversity** and **speech recognition capabilities** are crucial for achieving a higher level of accuracy and reliability.

5.3 AI Chatbot for Owner Module

The **AI Chatbot for Owner Module** is intended to assist the shop owner with administrative tasks, such as **generating sales reports**, **inventory management**, and **data querying**. By providing an easy-to-use, natural language interface, the module helps the owner efficiently access and manage information without the need for extensive technical knowledge.

One of the major concerns in developing this module was ensuring a **secure login** mechanism to protect sensitive business data. Initially, a **basic authentication system** was implemented, which involved username and password verification. However, this approach, while functional, lacked the robust security measures required for handling confidential information such as sales figures and inventory details.

To address this concern, additional security measures were introduced, such as **password hashing** and **HTTPS** for encrypted communication. Despite these improvements, it is recommended that the login system be further upgraded to include **OAuth 2.0** authentication. OAuth 2.0 is an industry-standard protocol for authorization, providing more secure access and the ability to integrate **multi-factor authentication (MFA)**. By implementing MFA, the system can add an extra layer of security, ensuring that unauthorized access is prevented, even in the event of a compromised password.

Another potential enhancement for this module is to introduce **role-based access control (RBAC)**, which would allow for more granular permissions. For example, while the shop owner might require access to all data, other staff members may only need access to inventory details without viewing sensitive sales information. Implementing RBAC would ensure that each user has access only to the data necessary for their role, thereby reducing the risk of data breaches.

In summary, while the **AI Chatbot for Owner Module** has succeeded in simplifying administrative tasks for the shop owner, future improvements in **security**—such as integrating OAuth 2.0 and RBAC—are essential to safeguarding sensitive data and ensuring that the system remains secure as it scales.

References

- [1] Nielsen, J. (1993). *Usability Engineering*. Academic Press.
- [2] Shneiderman, B. (1997). *Designing the User Interface: Strategies for Effective Human-Computer Interaction*. Addison Wesley Longman.
- [3] Adamopoulou, E., & Moussiades, L. (2020). "An Overview of Chatbot Technology," in *IFIP International Conference on Artificial Intelligence Applications and Innovations*. Springer, pp. 373-383.
- [4] Shawar, B. A., & Atwell, E. (2007). "Chatbots: Are they really useful?," *LDV Forum*, 22(1), 29-49.
- [5] Wood, D., Powell, P., & Reeves, C. (2019). "AI and Business Process Automation," in *Journal of Business Research*, 99, pp. 54-61.
- [6] Jaseena, K. U., & David, A. C. (2014). "Security Measures for Cloud Computing Systems: A Survey," in *International Journal of Engineering and Technology*, 6(6), pp. 2464-2472.
- [7] Kumar, A., Gupta, M., & Raj, P. (2020). *Hands-On Intelligent Agents with Open-Source Libraries: Build AI chatbots and Automate Online Interactions*. Packt Publishing.
- [8] Buyya, R., Yeo, C. S., & Venugopal, S. (2009). "Cloud Computing and Emerging IT Platforms: Vision, Hype, and Reality for Delivering Computing as the 5th Utility," in *Future Generation Computer Systems*, 25(6), 599-616.
- [9] Lu, L., Shen, H., & Miao, C. (2020). "A Chatbot Enhanced Interactive Web-Based Platform for Resource Allocation and Scheduling," in *Applied Soft Computing*, 89, 106050.
- [10] Chen, Z., Zhao, S., & Li, Q. (2021). "Enhancing Chatbot Responses Using Knowledge Graphs," in *Journal of Artificial Intelligence Research*, 71, pp. 101-118.
- [11] Adams, T., et al. (2020). Enhancing organizational transparency through document analysis. *Journal of Management Studies*, 45(2), 213-228.
- [12] Brown, A., & White, B. (2020). Stakeholder interviews: A practical guide. *Business Review*, 17(3), 45-58.
- [13] Garcia, J., et al. (2021). Observational studies in organizational research: Methods and applications. *Organization Science*, 28(1), 56-72.
- [14] Johnson, L. (2016). Document analysis in organizational research. *Journal of Applied Psychology*, 32(4), 321-335.
- [15] Johnson, M., & Smith, K. (2020). Understanding stakeholder expectations: A case study approach. *Strategic Management Journal*, 25(3), 187-201.

- [16] Jones, R., et al. (2019). Effective stakeholder engagement: Lessons from industry. *Journal of Business Ethics*, 41(2), 123-137.
- [17] Lee, H., & Smith, J. (2017). Observational studies: A primer. *Journal of Organizational Behavior*, 12(1), 45-58.
- [18] Smith, P. (2018). Fact-finding techniques in organizational analysis. *Harvard Business Review*, 30(2), 89-102.
- [19] Taylor, R., & Clark, E. (2018). Analyzing organizational documents: A step-by-step guide. *Administrative Science Quarterly*, 55(3), 340-355.
- [20] Williams, D., & Davies, S. (2019). Capturing stakeholder requirements: A practical approach. *Journal of Information Technology*, 22(4), 189-205.
- [21] J. Smith, "Designing AI-Powered E-commerce Solutions," *Journal of Computer Science and Technology*, vol. 35, no. 4, pp. 567-579, July 2022.
- [22] A. Brown and B. Williams, "Secure Payment Processing in Online Retail Systems," *IEEE Transactions on Information Forensics and Security*, vol. 15, no. 2, pp. 345-356, February 2021.
- [23] M. Davis, "Implementing Natural Language Processing for Customer Service Chatbots," *Proceedings of the 2020 International Conference on Artificial Intelligence and Machine Learning*, pp. 123-130, December 2020.
- [24] R. Kumar, "Efficient Database Management for Large-Scale E-commerce Applications," *IEEE Transactions on Knowledge and Data Engineering*, vol. 33, no. 7, pp. 1456-1468, July 2020.
- [25] S. Lee and J. Kim, "Inventory Management Techniques in Retail," *International Journal of Inventory Research*, vol. 12, no. 3, pp. 231-245, May 2019.
- [26] P. Gonzalez, "Web Interface Design for Enhanced User Experience," *Journal of Web Engineering*, vol. 18, no. 1, pp. 89-101, March 2019.
- [27] "Software Testing Methodologies," smartbear.com. Accessed: Oct. 02, 2024. [Online]. Available: <https://smartbear.com/learn/automated-testing/software-testing-methodologies/>
- [28] "Software Testing," Methods. Accessed: Oct. 02, 2024. [Online]. Available: https://www.tutorialspoint.com/software_testing/software_testing_methods.htm
- [29] GeeksforGeeks, "Types of Software Testing," GeeksforGeeks, Aug. 01, 2017. Accessed: Oct. 02, 2024. [Online]. Available: <https://www.geeksforgeeks.org/types-software-testing/>

- [30] S. Chatterjee, "What is Software Test Methodology? (With 6 Test Methodologies)," BrowserStack, Sep. 27, 2024. Accessed: Oct. 02, 2024. [Online]. Available: <https://www.browserstack.com/guide/software-testing-methodologies>
- [31] "Testing Your Assistant," Rasa Documentation. Accessed: Oct. 02, 2024. [Online]. Available: <https://rasa.com/docs/rasa/testing-your-assistant/>
- [32] E. Kinsbruner, "The Complete Guide to Different Types of Testing," Perfecto by Perforce. Accessed: Oct. 02, 2024. [Online]. Available: <https://www.perfecto.io/resources/types-of-testing>
- [33] J. Unadkat, "Website Testing: A Detailed Guide," BrowserStack, Aug. 28, 2023. Accessed: Oct. 02, 2024. [Online]. Available: <https://www.browserstack.com/guide/how-to-perform-website-qa-testing>
- [34] W. M. Brown, A. Semin, M. Hebenstreit, S. Khvostov, K. Raman, and S. J. Plimpton, "Increasing molecular dynamics simulation rates with an 8-fold increase in electrical power efficiency," in SC '16 Proc. of the Int. Conf. for High Perf. Computing, Networking, Storage and Analysis, Salt Lake City, Utah, 2016.
- [35] A. Vaskuri, H. Baumgartner, P. Kärhä, G. Andor, and E. Ikonen, "Modeling the spectral shape of InGaAlPbased red light-emitting diodes," Journal of Applied Physics, vol. vol. 118, no. Jul. 2015, pp. pp. 203103-1– 203103-7, 2015.
- [36] N. B. Vargafik, J. A. Wiebelt, and J. F. Malloy, "Radiative transfer," in Convective Heat, Melbourne, Engineering Education Australia, 2011, p. pp. 379–398.
- [37] James D, Rob Hans, "Science portal, Wikipedia," Wikimedia Foundation, 5 May 2001. [Online]. Available: <https://en.wikipedia.org/wiki/Portal:Science>. [Accessed 19 April 2023]
- [38] Lewis, J., & Writer, C. (2014). Database management technologies: A review. *Journal of Information Technology Research*, 7(3), 45-67. DOI: 10.4018/jitr.2014070103
- [39] G. Ahern, "8 Ways AI Chatbots Help In Inventory Management for eCommerce," Ochatbot - AI Chatbot & LeadBot, Feb. 21, 2023. <https://ochatbot.com/inventory-management-for-eCommerce/>

Appendix

Appendix 1: Stakeholder Interviews

https://drive.google.com/file/d/1ValyIt076A_JHZHPkd_jVgbcZ8qz0Qyj/view?usp=sharing

https://drive.google.com/file/d/1dZGh_OIFqV_UnR5LDzll9eB1KpUzR6yC/view?usp=sharing

Question 1- Can you provide an overview of your current workflow and processes in managing computer sales, repairs, and CCTV services, considering that you're using an old, inefficient digital inventory management system?

Answers - Currently, our workflow is hindered by an old, inefficient digital inventory management system. It's challenging to keep track of computer sales, repairs, and CCTV services effectively.

Question 2- What are the main pain points or challenges you currently face in your day-to-day operations with the outdated inventory management system?

Answers - The main pain points with our current system stem from its outdated nature. We struggle with real-time stock visibility, and there's no automatic warning system for low stock levels. Additionally, generating monthly reports for profit and income adjustments is time-consuming and cumbersome.

Question 3- How do you currently handle inventory management with your existing system, and what limitations do you experience?

Answers - With the old system, we can view stock, but it's not in real-time, and we must manually update stock levels. There's no automatic alert system for low stock levels, and generating monthly reports requires manual data entry and manipulation.

Question 4- What specific features or functionalities are you looking for in a computer shop management system to address the shortcomings of your current system?

Answers - In a new computer shop management system, we're looking for real-time stock visibility, automatic alerts for low stock levels, and the ability to generate monthly reports effortlessly.

Question 5- How do you currently handle customer inquiries, service requests, and appointments with your current system, and how do you envision these processes being improved with a new system?

Answers - Customer inquiries and service requests are currently managed manually, leading to delays and potential errors. With a new system, we expect streamlined processes for handling customer interactions, improving response times, and enhancing overall customer satisfaction.

Question 6- Would you be interested in incorporating a chatbot feature into your computer shop management system to enhance customer support and streamline interactions?

Answers - Yes, we would be interested in adding a chatbot feature to our system. We believe that incorporating a chatbot would significantly improve our customer support capabilities by providing immediate assistance to customers and addressing their queries in real-time. This would not only enhance our overall customer experience but also free up valuable staff time, allowing our team to focus on more complex tasks and providing personalized service to our customers. Additionally, having a chatbot would align with our goal of modernizing our operations and staying ahead of the curve in terms of technological advancements in customer service.

Question 7- How do you currently manage technician assignments and track service requests using your outdated system, and how would you like these processes to be enhanced in a new system?

Answers - Technician assignments and service request tracking are currently managed manually, resulting in inefficiencies and potential scheduling conflicts. In a new system, we're looking for automated scheduling and tracking functionalities to optimize technician assignments and improve service delivery.

Question 8- What are your expectations regarding integration with existing software systems or tools, considering the limitations of your current system?

Answers - Integration with our existing accounting software is crucial for seamless invoicing, billing, and payment processing. We expect the new system to integrate seamlessly with our current tools to minimize disruption and improve overall efficiency.

Question 9- How do you currently handle invoicing, billing, and payment processing with your current system, and how would you like these processes to be improved with a new system?

Answers - Invoicing, billing, and payment processing are currently manual processes prone to errors and delays. With a new system, we anticipate automated invoicing, billing, and payment processing functionalities to streamline operations and improve accuracy.

Question 10- What are your long-term goals or objectives for implementing a computer shop management system, considering the inefficiencies of your current system and the need for modernization?

Answers - Our long-term goal is to modernize our operations, improve efficiency, and enhance customer satisfaction through the implementation of a computer shop management system.

Appendix 2: Observational Studies



Figure 40 : Storing Items.

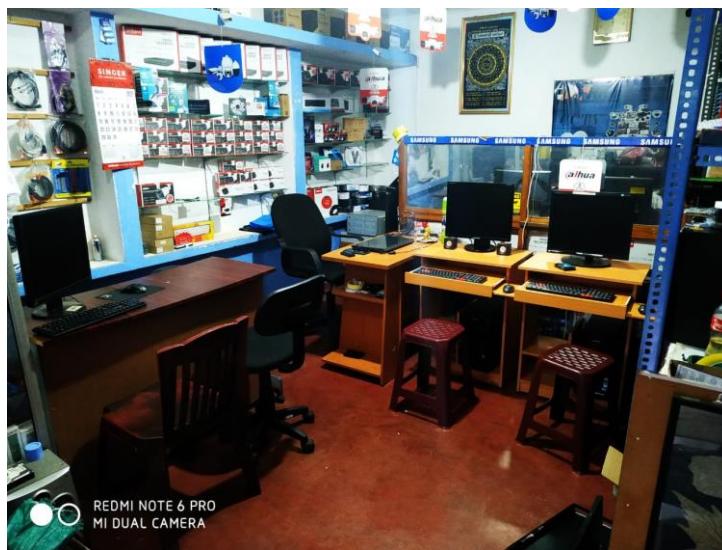


Figure 41: Working Place 1.

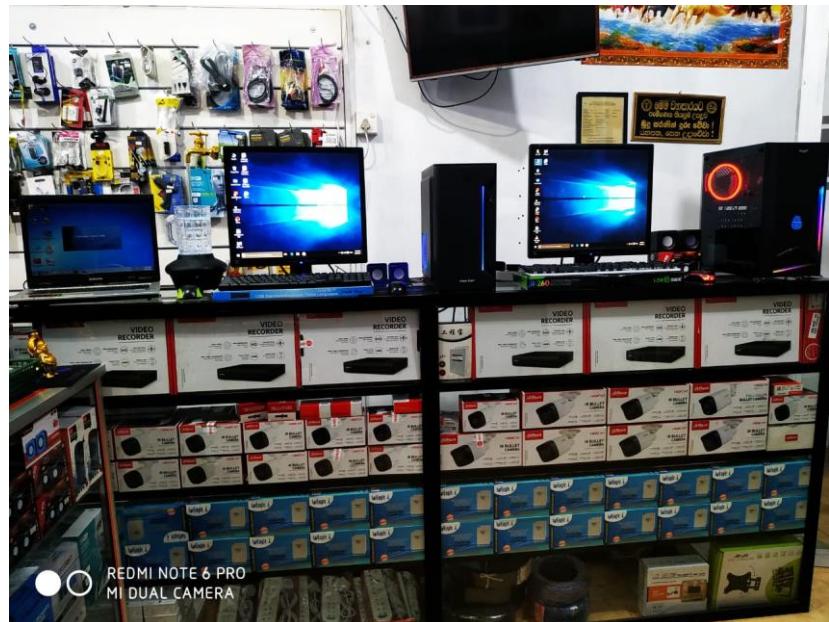


Figure 42 : Working Place 2.



Figure 43 : Customer Interaction.



Figure 44: Ambalangoda Branch.



Figure 45: Ampara Branch.

Appendix 3: Software Development Services Request Form

SOFTWARE DEVELOPMENT SERVICES REQUEST FORM

REQUEST INFORMATION

NAME : W.A.Kasun Hasintha

COMPANY : RK Computers and CCTV Operators

NAME

ADDRESS : RK Computers and CCTV Operators,
Kahawa road, Nindana.

SERVICE REQUEST

New application

Enhancement(s) to existing application.

Replace an existing application.

Automate a current manual process.

Other

DETAILED DESCRIPTION OF THE REQUESTED SERVICES

Inventory Management System:

We are seeking a robust Inventory Management Software that will enable us to monitor stock levels, manage product variations, and automate reordering processes. Additionally, we request the integration of automated alerts to notify our staff when stock levels fall below predetermined thresholds, ensuring proactive inventory management.

Personalized Product Recommendations and Upselling:

We aim to implement personalized product recommendations based on customer purchase history and preferences. Your proficiency in developing algorithms that strategically promote relevant products will contribute to enhancing customer satisfaction and driving increased sales through effective upselling.

Customer Support and FAQ Integration:

Efficient customer support is a priority for us. We are looking for integrated Customer Support functionalities within our chatbot to assist customers with common inquiries and issues promptly. Your experience in implementing responsive support mechanisms is crucial for ensuring a positive customer experience.

Point-of-Sale (POS) Integration:

Ensuring a smooth transaction experience is paramount. We request your assistance in integrating our systems with leading payment gateways and other POS systems to guarantee a secure, efficient, and user-friendly payment process for our customers.

WHAT BENEFITS WILL THIS SERVICES PROVIDE.

Inventory Management System:

Efficient Stock Monitoring: Real-time tracking of stock levels ensures optimal inventory management and prevents stockouts or overstock situations.

Automated Reordering: Streamlined reordering processes reduce manual workload, minimize errors, and ensure timely replenishment of stock.

Proactive Alerts: Automated alerts provide timely notifications, allowing staff to take proactive measures in response to low stock levels, preventing disruptions in supply chain management.

Personalized Product Recommendations and Upselling:

Improved Customer Experience: Personalized recommendations based on customer preferences enhance the overall shopping experience, leading to higher customer satisfaction.

Increased Sales: Effective upselling strategies driven by customer purchase history can result in higher average transaction values and increased revenue.

Customer Support and FAQ Integration:

Enhanced Customer Satisfaction: Quick and efficient customer support through integrated functionalities improves customer satisfaction and loyalty.

Time and Cost Savings: Automated responses to common inquiries reduce the workload on customer support staff, saving time and operational costs.

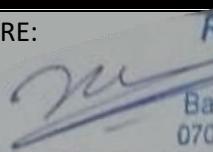
Point-of-Sale (POS) Integration:

Seamless Transactions: Integration with payment gateways ensures smooth and secure transactions, reducing friction in the buying process.

Improved Operational Efficiency: Automation of payment processes enhances operational efficiency, allowing staff to focus on other critical aspects of customer service.

AUTHORIZATION

SIGNATURE:



RK COMPUTERS
W.A.K HASINTHA
Batapola Road, Nindana
070 2566205 / 078 3672941

DATE:

2024/02/03

Appendix 4: Current system user interface



Figure 46: Logging Page UI.



Figure 47: Starting Page UI.

Owner chatbot coding

The screenshot shows the VS Code interface with the 'intents.json' file open in the editor. The code defines two intent objects: 'marketing_strategy' and 'seasonal_sales'. Each intent has a 'tag', 'patterns' (a list of user questions), and 'responses' (a list of bot answers). The 'marketing_strategy' intent is triggered by questions about improving marketing or suggesting ways to attract customers. The 'seasonal_sales' intent is triggered by questions about seasonal sales trends and comparisons.

```
2     "intents": [
10     },
11     {
12       "tag": "marketing_strategy",
13       "patterns": [
14         "How can I improve my marketing?",
15         "What marketing strategies should I use?",
16         "Can you suggest ways to attract more customers?"
17     ],
18     "responses": [
19       "Consider using social media marketing to reach more customers.",
20       "Offering discounts or promotions can help attract new buyers."
21     ]
22   },
23   {
24     "tag": "seasonal_sales",
25     "patterns": [
26       "What were my seasonal sales last year?",
27       "How do my sales compare during the holidays?",
28       "Can you analyze sales patterns for different seasons?"
29     ],
30     "responses": [
31       "Please provide the seasons you want to analyze for sales trends."
32     ]
33   }
34 ]
```

The screenshot shows a code editor interface with several tabs and files open. The left sidebar displays a project structure for 'chatbot-deployment' located at 'E:\chatbot-deployment'. The files listed include app.py, base.html, standalone-frontend/style.css, app.js, static/style.css, chat.py, intents.json, and various utility and configuration files like .gitignore, LICENSE, model.py, ntk_utils.py, and train.py. The right pane contains the content of the 'chat.py' file, which is a Python script. It includes imports for torch, torch.nn, and torch.nn.functional. The script defines a function 'get_response' that takes a message as input, processes it through a model, and returns a response based on the predicted intent. It also includes a main loop for interacting with the user. The bottom status bar shows the current file is 'chat.py', the file path is 'E:\chatbot-deployment\chat.py', and the file size is 4.1 MB.

```
def get_response(msg):  
    usage = "usage: A Patrick Loebber (Python Engineer)"  
    X = torch.tensor([msg], dtype=torch.float)  
  
    output = model(X)  
    _, predicted = torch.max(output, dim=1)  
  
    tag = tags[predicted.item()]  
  
    probs = torch.softmax(output, dim=1)  
    prob = probs[0][predicted.item()]  
    if prob.item() > 0.75:  
        for intent in intents['intents']:  
            if tag == intent["tag"]:  
                return random.choice(intent['responses'])  
  
    return "Sorry Mr.Kasun,I do not understand ..."  
  
if __name__ == "__main__":  
    print("Let's chat! (type 'quit' to exit)")  
    while True:  
        sentence = "do you use credit cards?"  
        sentence = input("You: ")  
        if sentence == "quit":  
            break
```

Customer chatbot coding

```
faq.html    # style3.css   JS script3.js   about.html    # style4.css   JS script4.js   feedback.html # style5.css   JS script5.js   config.js

chatbot > js > JS script5.js > window.addEventListener('load') callback
1 const preloader = document.querySelector('.preloader');
2 const darkMode = document.getElementById('dark');
3 document.getElementById('dark').checked = JSON.parse(localStorage.getItem("sharedBool2"));
4 changeMode();
5 //shared variable
6 const nav = document.getElementById('nav-toggle');
7 nav.addEventListener("click", () => {
8
9     let boolVar = document.getElementById('nav-toggle').checked;
10    localStorage.setItem("sharedBool1", JSON.stringify(boolVar));
11
12 });
13
14
15 //loading
16 window.addEventListener("load", () => [
17     menuCheckbox(),
18     document.getElementById('nav-toggle').checked = JSON.parse(localStorage.getItem("sharedBool1"));
19     //setTimeout(() => {
20     //preloader.style.display = "none";}, 1000);
21     preloader.style.animation = "fade-out 0.5s cubic-bezier(0.550, 0.085, 0.680, 0.530) 0.75s both";
22     preloader.addEventListener('animationend', () => {
23         // Once the animation ends, hide the preloader
24         preloader.style.display = 'none'; // Alternatively, you can remove the element with preloader.remove();
25     });
26
27 ];
28
29 //menu
30 function menuCheckbox() {
31     if (window.innerWidth < 868) {
32         document.getElementById('nav-toggle').checked = true;
33     }
34 }
35
36
37 //remove blur
```

The screenshot shows the PyCharm IDE interface. The left sidebar displays a project structure with files like test_stories.yml, actions.py, stories.yml, rules.yml, and domain.yml. The domain.yml file is currently selected and open in the main editor. The code in domain.yml defines several responses for anutter. The responses include:

- utter_about: A response with text about the company's vision and mission.
- utter_cctv_price: A response with text about the pricing of CCTV cameras.
- utter_cctv_installation: A response with text about professional installation services.
- utter_cctv_operation: A response with text about operating and monitoring cameras.
- utter_ask_repairs: A response with text about repair services.

The terminal at the bottom shows the command prompt in a venv environment, indicating the project is ready for development.

```
56     responses:
57     utter_about:
58         - text: "RK Computers and CCTV Operations, based in Ambalangoda (Sri Lanka), is your go-to destination for cutting-edge technology solutions. We pride ourselves on being a reliable partner for both individuals and businesses. Our vision is to see technology seamlessly integrate into everyday life, empowering individuals and businesses to thrive in a connected and secure environment. See about section for more information"
59
60     utter_cctv_price:
61         - text: "Our CCTV cameras start at Rs.10000.00 , with advanced models priced up to Rs.500000.00. If you're looking for something specific, please let us know and we'll find the perfect fit for you."
62
63     utter_cctv_installation:
64         - text: "We offer professional CCTV installation services. Our team can install and configure the cameras at your location, ensuring a smooth setup and integration with your existing systems."
65
66     utter_cctv_operation:
67         - text: "Operating a CCTV camera is straightforward. You can monitor live footage through a connected device, adjust settings, and receive notifications for any activity. We provide a user manual and support for setup and operation."
68
69     utter_ask_repairs:
70         - text: "Yes, we offer repair services. Please bring your faulty device to our shop, and we'll take a look at it for you."
```

Customer chatbot intent test cases

```
stories:
- story: say goodbye
  steps:
    - user: |
      bye-bye!
      intent: goodbye
    - action: utter_goodbye

- story: bot challenge
  steps:
    - user: |
      are you a bot?
      intent: bot_challenge
    - action: utter_iamabot

- story: find_components
  steps:
    - user: |
      What components do I need for playing GTA V
      intent: find_components
    - action: action_find_components

- story: ask for help
  steps:
    - user: |
      I need help with my order.
      intent: asking_help
    - action: utter_offer_help

- story: say hello
  steps:
    - user: |
      Hello!
      intent: greet
    - action: utter_greet

- story: ask about shop
  steps:
    - user: |
      Do you have a physical store?
      intent: asking_about_shop
    - action: utter_about

- story: ask about item price
  steps:
    - user: |
      What is the price of Intel i5?
      intent: item_price
    - action: action_give_price

- story: ask for item picture
  steps:
    - user: |
      show me Lenovo IdeaPad Flex 5
      intent: ask_item_pic
    - action: action_show_item

- story: ask about capabilities
  steps:
```

```

- user: |
  What can you help with?
  intent: asking_capabilities
- action: utter_give_capabilities

- story: ask about available items
steps:
- user: |
  Show what I can buy.
  intent: ask_available_items
- action: action_list_available_items

- story: ask if specific item is available
steps:
- user: |
  Is Intel Core i9 available?
  intent: ask_available_one_item
- action: action_check_item_availability

- story: ask about discount on item
steps:
- user: |
  Is there a discount on Intel i5?
  intent: ask_discount_on_item
- action: action_check_discount

- story: ask about warranty on item
steps:
- user: |
  Can you tell me the warranty for Samsung 970 EVO SSD
  intent: ask_warranty_on_item
- action: action_check_warranty

- story: ask for description of item
steps:
- user: |
  Describe the Logitech G502 mouse
  intent: ask_description_of_item
- action: action_describe_item

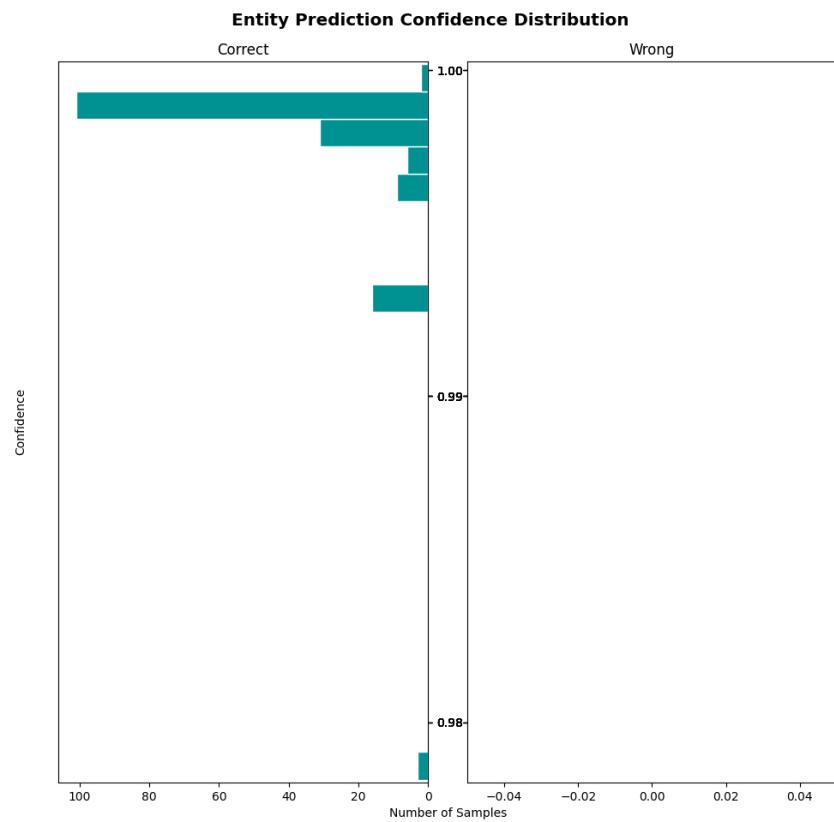
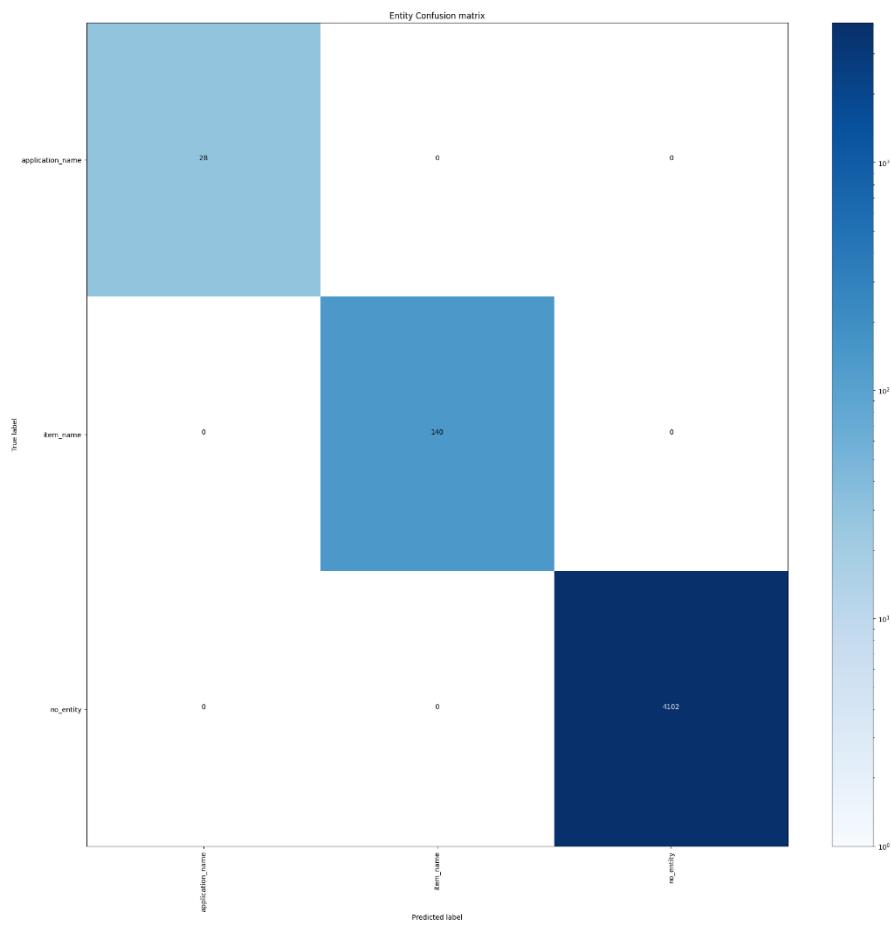
- story: ask about working hours
steps:
- user: |
  What are your working hours?
  intent: asking_about_working_hours
- action: utter_ask_working_hours

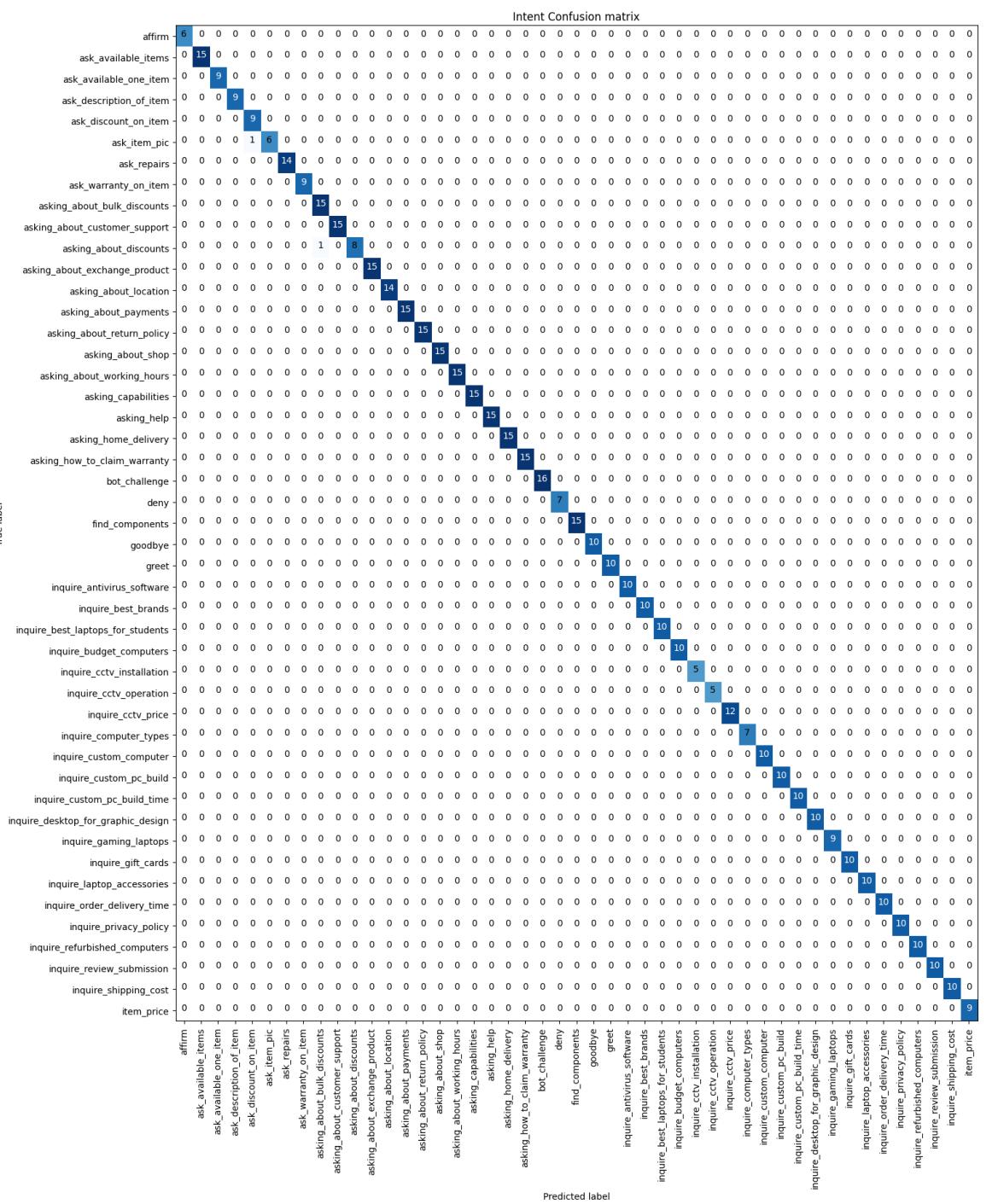
- story: ask about location
steps:
- user: |
  Where I find you?
  intent: asking_about_location
- action: utter_location

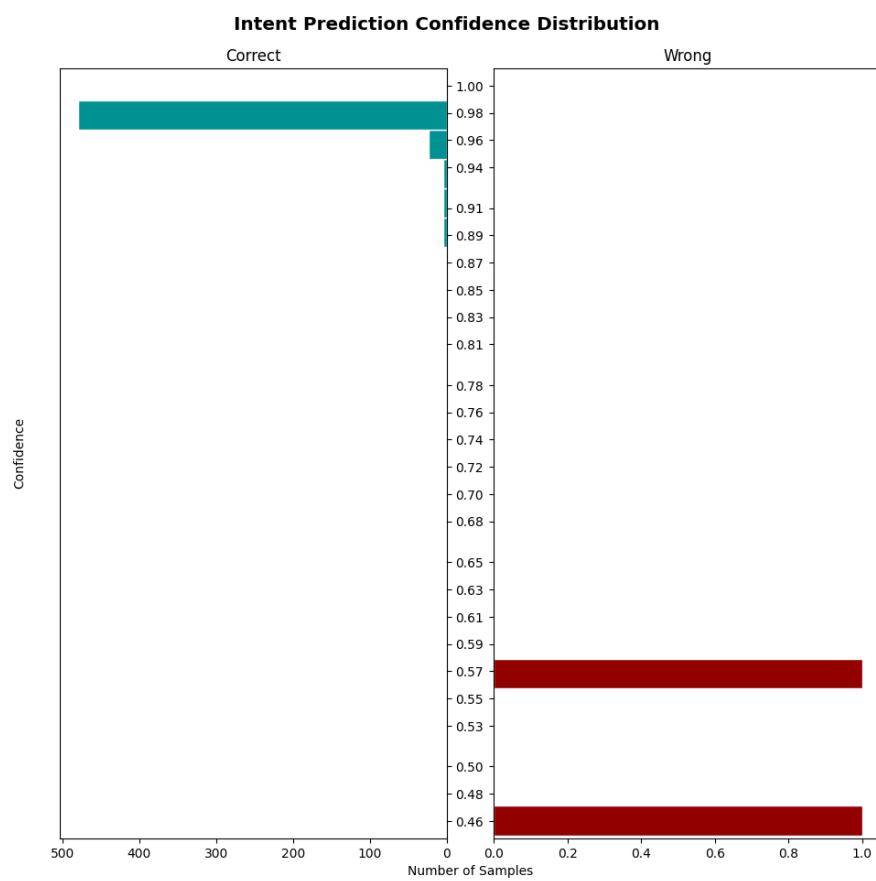
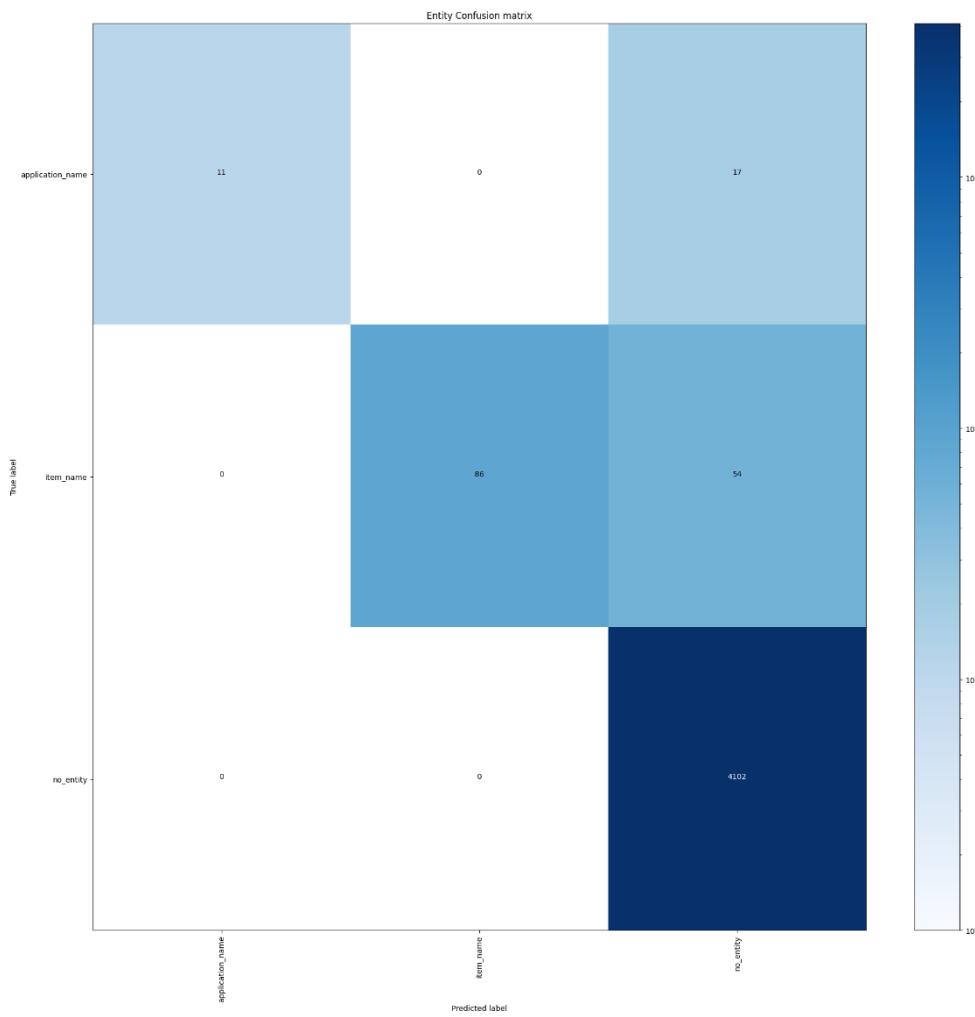
- story: ask about home delivery
steps:
- user: |
  Do you offer home delivery?
  intent: asking_home_delivery
- action: utter_home_delivery

```

Customer chatbot intent test results:



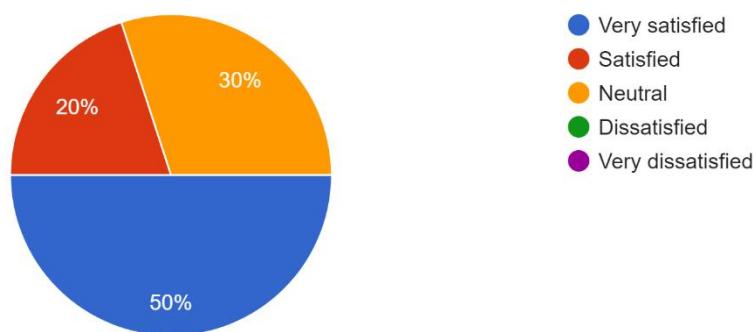




Customer chatbot User testing Feedback form results:

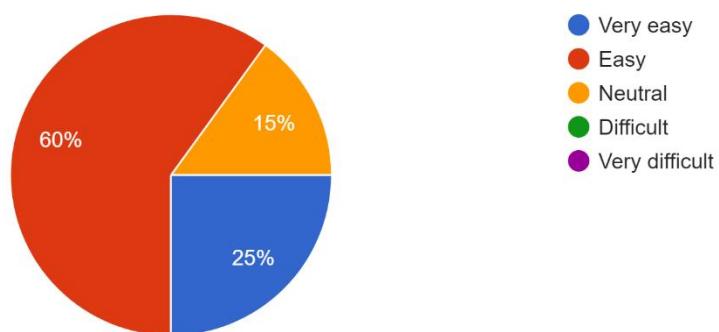
Overall, how satisfied are you with your experience with our chatbot?

20 responses



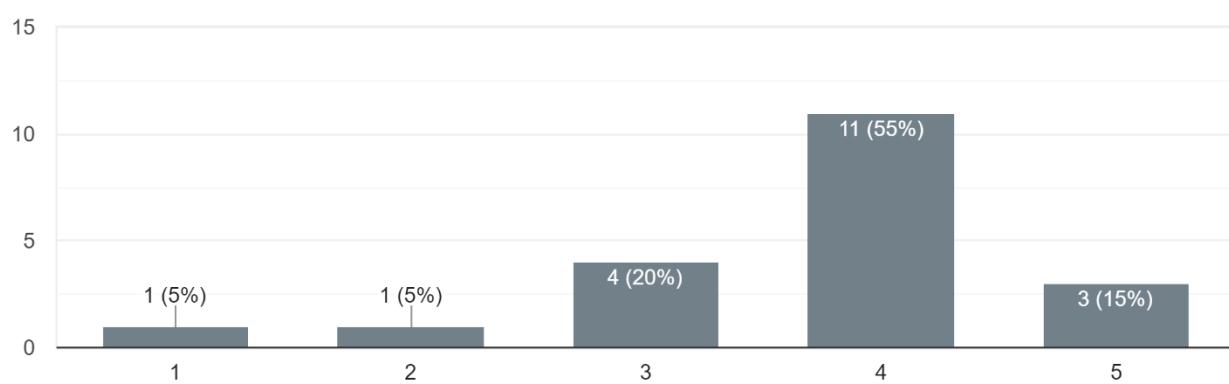
How easy was it to interact with the chatbot?

20 responses



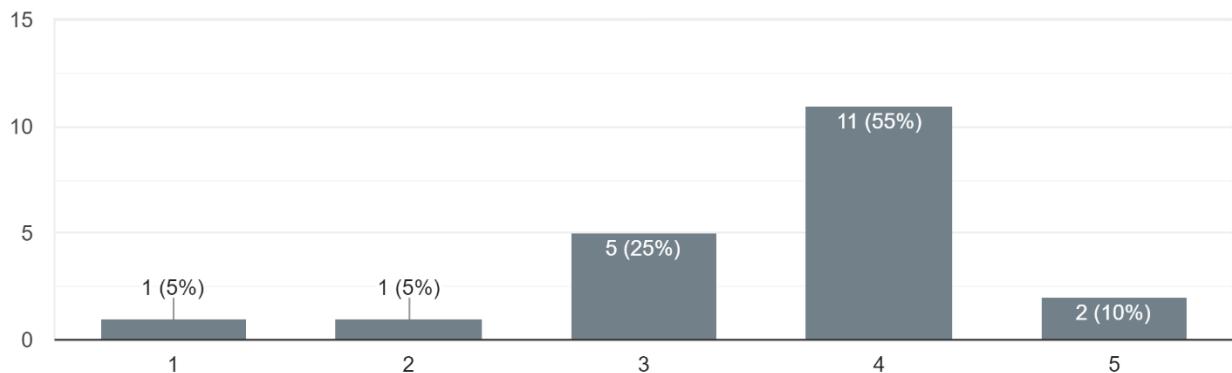
How visually appealing did you find the chatbot's interface?

20 responses



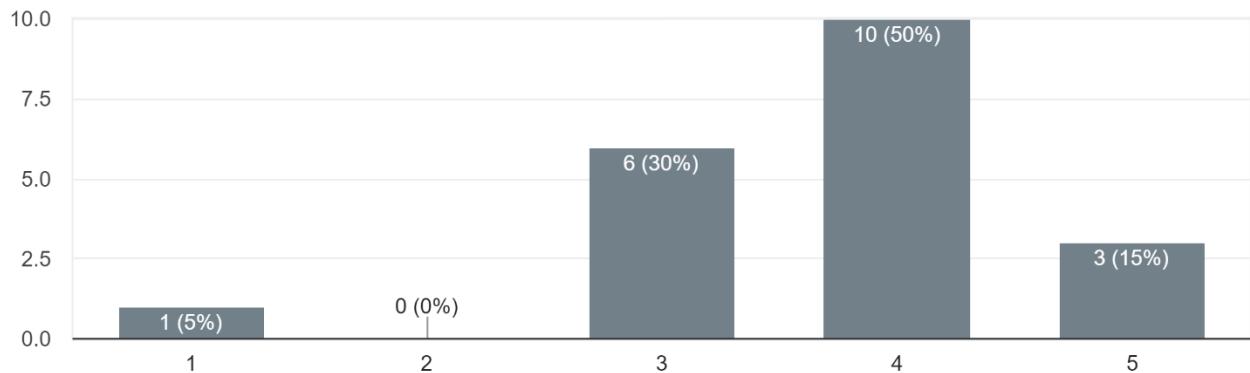
How clear and readable was the text on the chatbot's interface?

20 responses



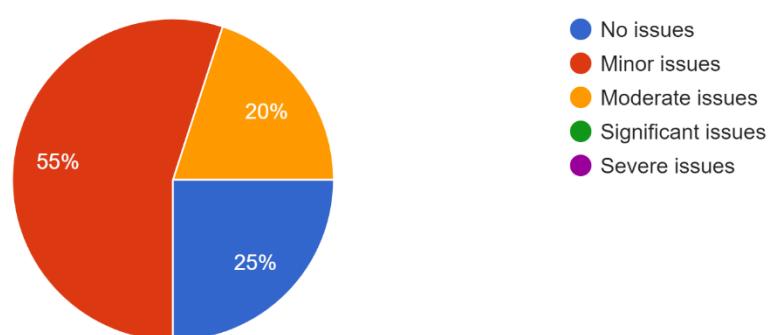
Was the chatbot's interface easy to navigate?

20 responses



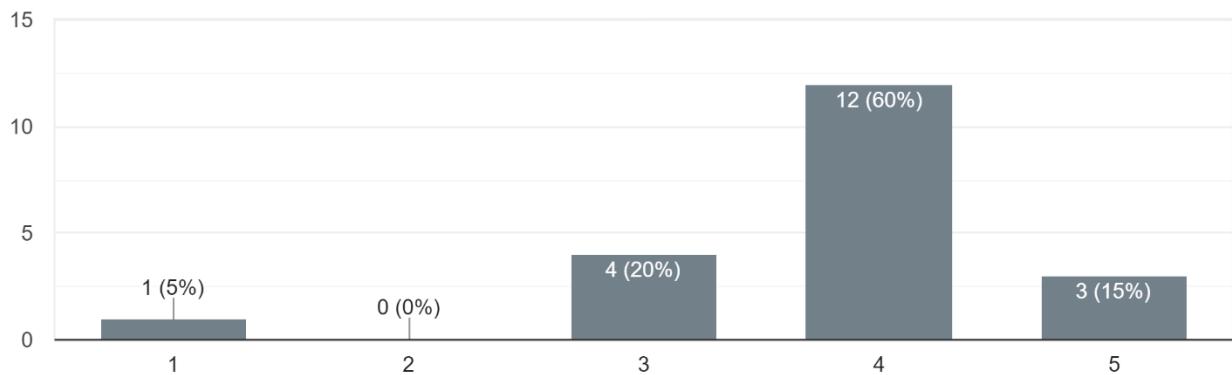
Did you experience any technical issues while interacting with the chatbot? (e.g., slow response time, errors)

20 responses



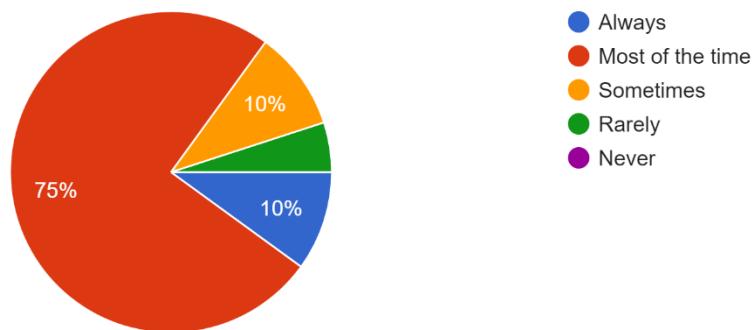
How satisfied are you with the speed of the chatbot's responses?

20 responses



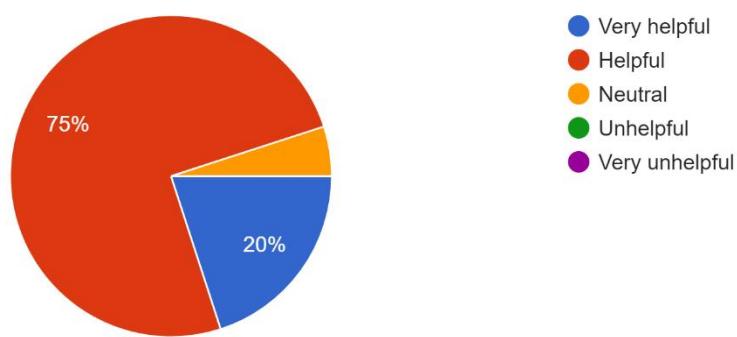
Did the chatbot understand your questions and provide relevant answers?

20 responses



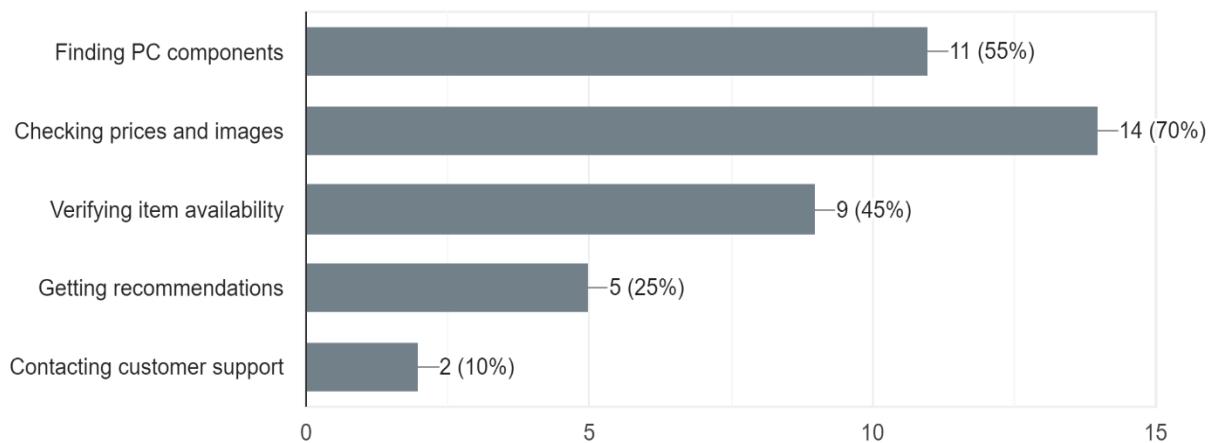
How helpful was the information provided by the chatbot?

20 responses



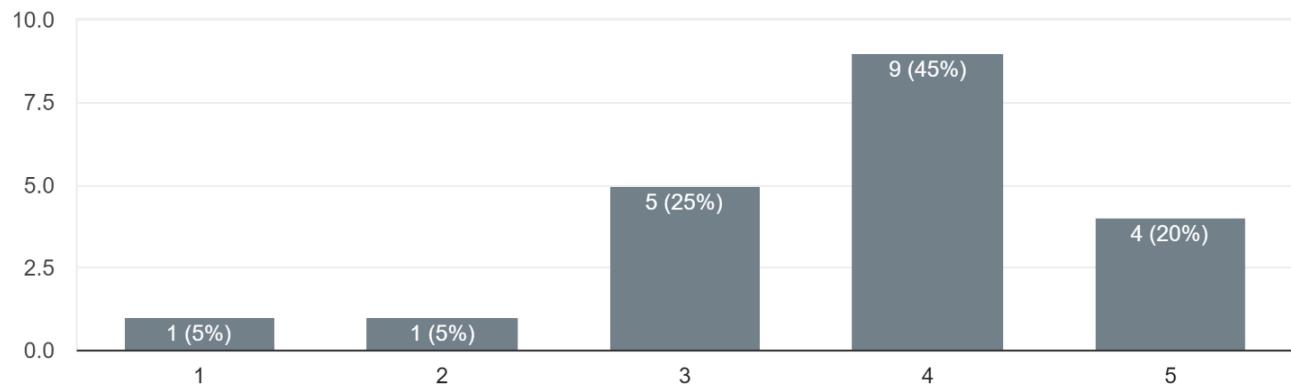
Which features did you find most useful?

20 responses



Did you find the chatbot's interface to be consistent across different devices (e.g., desktop, mobile)?

20 responses



Web interface responsive test

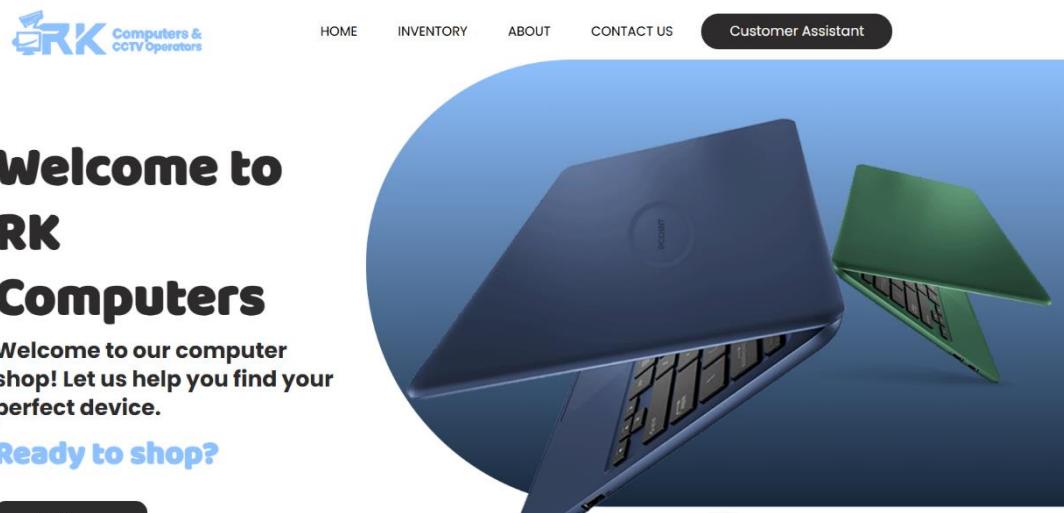


Figure 48: Web Interface Desktop View

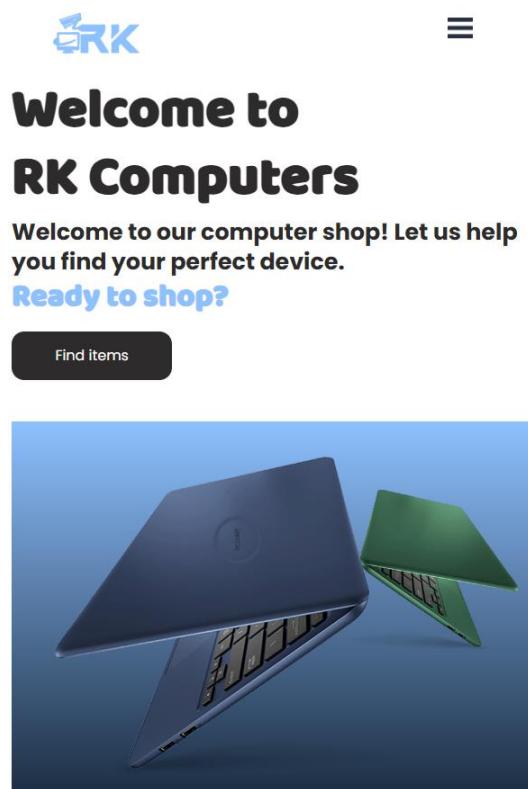


Figure 49 : Web Interface Mobile View

Customer chatbot user interface responsive test

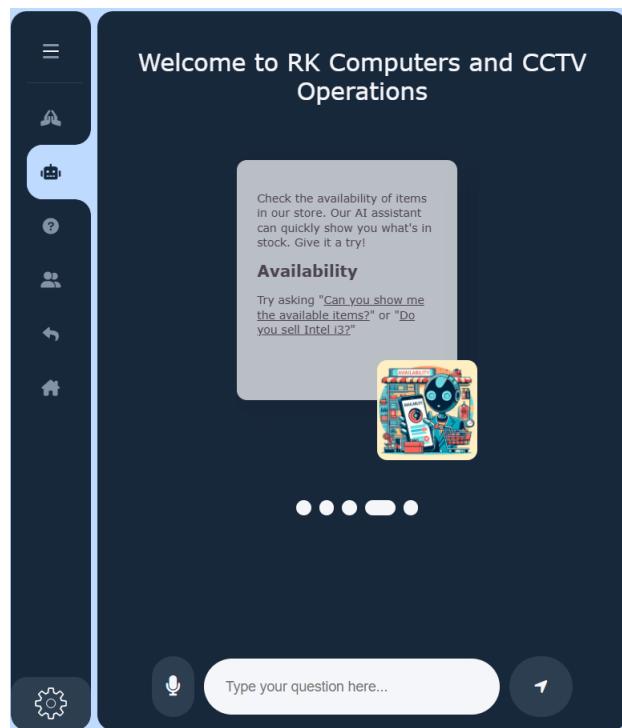


Figure 50: Customer Chatbot UI Mobile View

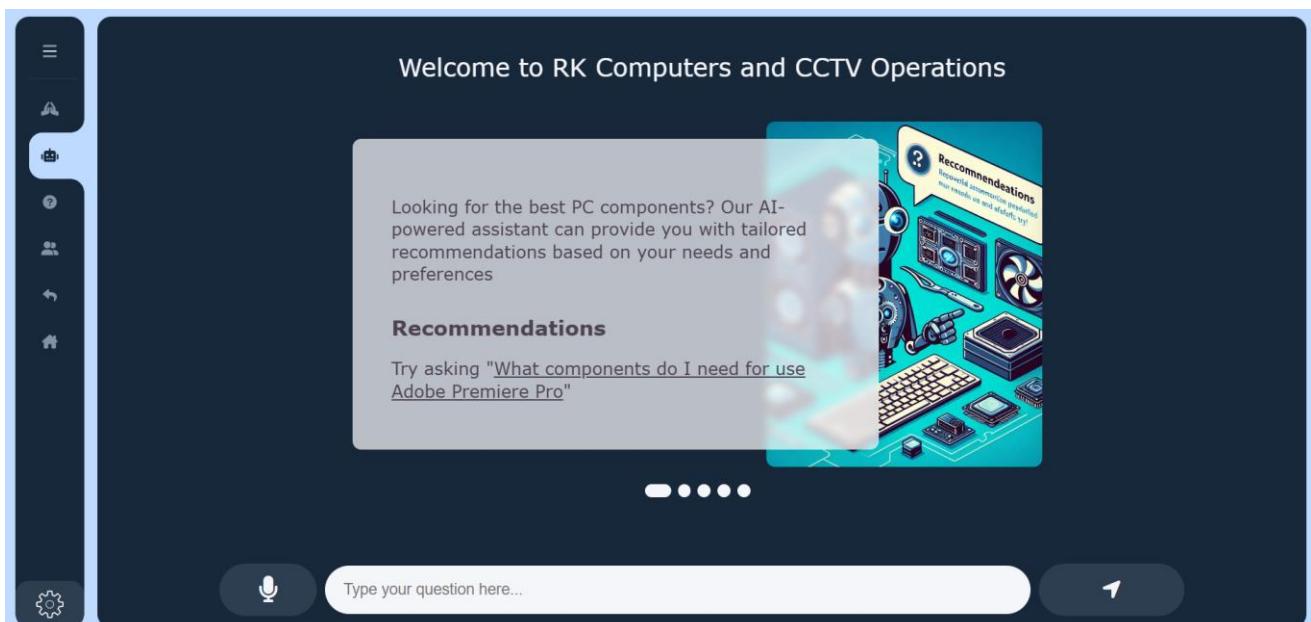


Figure 51: Customer Chatbot UI Desktop view