



# **Web-Based Computer Hardware Purchasing and Troubleshooting Assistant Management System**

**G.P SANKALPA**  
Bit Registration No: - R180527  
Index No: - 1805274

Name of the Supervisor:  
**M.P. Sarath Wijesinghe**

**April 2022**



This dissertation is submitted in partial fulfillment  
of the requirement of the Degree of Bachelor of Information Technology of the  
University of Colombo School of Computing.



# Declaration

"I certify that this dissertation does not incorporate, without acknowledgement, any material previously submitted for a degree or diploma in any university and to the best of my knowledge and belief, it does not contain any material previously published or written by another person or myself except where due reference is made in the text. I also hereby give consent for my dissertation, if accepted, to be made available for photocopying and for interlibrary loans, and for the title and abstract to be made available to outside organizations.



Signature of Candidate: .....

Date: 29/01/2021

Name of Candidate: G.P Sankalpa

Countersigned by:



Signature of Supervisor(s)/Advisor(s): .....

Date: 29/01/2021

Name(s) of Supervisor(s)/Advisor(s): Mr. M.P.S. Wijesinghe

# **Abstract**

The computer sales and servicing industry sells and services a wide range of computers, including desktops, laptops, notebooks, palmtops, and software, as well as peripherals such as printers, scanners, and keyboards. Before being delivered to end users like homes and businesses, these products are often bought from makers and distributors in both domestic and foreign markets.

Under the present system of U-Star Digital, customers used to come into the store and communicate their demands to the technician who built their computer. Customers arrived at the store, gave over the defective item or computer to the technicians, and then waited in their restroom until the task was finished. They presently use a phone-based and WhatsApp-based order system to take online orders. There are no other options than bank transfers, and the customer must confirm the order with formal transaction documents. Due to a lack of adequate delivery information, customers are not given accurate delivery information.

The project's primary goal is to design and build an online platform for managing computer services and hardware that will assist the firm and its employees in growing their customer base and promoting their enterprise online.

For modeling, the system uses a client-server architecture and a non-object-oriented iterative software development process called Rational Unified Process. The system is designed using the Unified Modeling Language. For front-end development, PHP is utilized as the server-side programming language, along with HTML, JavaScript, and CSS. As an integrated development environment, the Microsoft Visual Studio Code is employed. The Apache web server is utilized and MySQL is used to manage the database. It may be used with a web browser on any GUI-based OS platform as this is a web-based system. This system has offered to meet the needs of the client. It will be quite beneficial in archiving their commercial objectives.

# Acknowledgements

Want to start by expressing my sincere gratitude to my supervisor, Mr. M.P. Sarath Wijeshinghe, Assistant Director - IT, Sri Lanka Export Development Board, for his unflagging support, helpful assistance, and consistent monitoring despite his busy schedule.

Mr.R.S.Kannangara, the owner of U-Star Digital, was also extremely helpful in gathering requirements and conducting user acceptance tests. I also like to thank the other U-Star Digital team members who volunteered their time to assist with domain-related information.

In order to use the knowledge from my degree programme, I'd want to thank the University of Colombo School of Computing, the Academic and Project Coordinators, and the Project Examination Board for Bachelor of Information Technology. I also want to sincerely thank all of my instructors for teaching me during my BIT degree programme.

Also owe thanks and appreciation who helped me out by offering suggestions for the user interfaces of the system and website. Finally, but certainly not least, I want to thankful to my parents for providing all of the materials needed for the project as well as for greatly inspiring and supporting me mentally to complete it.

# Table Contents

Declaration .....	ii
Abstract .....	iii
Acknowledgements.....	iv
List of Figures .....	ix
List of Tables .....	xii
List of Acronyms .....	xiii
Chapter 1 – Introduction .....	1
1.1 Chapter Introduction .....	1
1.2 Introduction to U-Star Digital .....	1
1.3 Motivation of the Project.....	1
1.4 Objectives of the Project .....	2
1.5 Scope of the Project.....	2
1.5.1 Shopping Cart Module.....	2
1.5.2 Payment Module.....	2
1.5.3 Inventory Management Module .....	3
1.5.4 Delivery Management Module .....	3
1.5.5 Troubleshooting Assistant Module.....	3
1.5.6 Invoice Generating Module .....	3
1.5.7 User Management Module .....	3
1.5.8 Report Management Module .....	3
1.6 Structure of Dissertation.....	3
Chapter 2 – Analysis .....	5
2.1 Chapter Introduction .....	5
2.2 Fact Finding Techniques .....	5
2.2.1 Interviews .....	5
2.2.2 Observation.....	6
2.2.3 Existing Documents Analyzing .....	6
2.3 Existing System.....	6
2.4 Existing System Data Flow Diagram .....	6

2.5 Requirements Analysis.....	8
2.5.1 Functional Requirements .....	8
2.5.1 Non-Functional Requirements.....	9
2.6 Similar Systems and Literature Reviews .....	10
2.6.1 Microcenter.....	10
2.6.2 Newegg .....	11
2.6.3 Comparison of the Proposed System to a Comparable System.....	11
2.7 Model of the Proposed System's Process .....	12
2.7.1 Scrum.....	12
2.7.2 Extreme Programming (XP).....	12
2.7.3 Rational Unified Process (RUP).....	12
Chapter 3 – Design.....	15
3.1 Chapter Introduction .....	15
3.2 Alternate Solution .....	15
3.2.1 Stand-alone System .....	15
3.2.2 Collection of Software.....	15
3.2.3 Justification for Choosing a Web-Based System .....	15
3.3 The Architectural Design of the System .....	16
3.4 Diagram of the Use Case for the Proposed System .....	17
3.5 Use Case Narratives for the Proposed System .....	19
3.5.1 Use Case Narratives for Login Module.....	19
3.5.2 Use Case Narratives for Insert Inventory Item.....	20
3.6 ER Diagram for the Proposed System.....	21
3.7 Relational Shecma for System.....	22
3.8 Activity Diagrams for the Proposed System .....	23
3.8.1 Activity Diagram for Login.....	23
3.8.2 Activity Diagram for Add Item .....	23
3.9 – UI Designs for the Proposed System.....	24
3.9.1 – Customer Login and Register Form UI Design.....	24
3.9.3 – Customer Checkout UI Design.....	25

3.9.3 – Add Inventory Item Form.....	26
Chapter 4 – Implementation.....	27
4.1 – Chapter Introduction .....	27
4.2 – Implementation Environment.....	27
4.2.1 – Software Environment .....	27
4.2.2 – Hardware Environment.....	28
4.2.3 – Other Software and Development Tools .....	28
4.3 – Justification for the choice of the implementation platform .....	28
4.3.1 System File Structure.....	30
4.4 – High Level Module Structure.....	30
4.5 – Major Code Segments .....	31
4.5.1 – Customer Checkout Code Segments .....	34
4.6 – Reused Existing Codes.....	35
Chapter 5 – Evaluation.....	39
5.1 – Introduction .....	39
5.2 - Description of testing approach .....	39
5.3 - Test strategy .....	39
5.3.1 - Verification and Validation.....	40
5.3.2 - Testing Process.....	40
5.5 - User Acceptance Testing .....	43
5.6 - User Evaluation.....	43
Chapter 6 – Conclusion.....	47
6.1- Introduction .....	47
6.2 - Critical evaluation of the project.....	47
6.3 - Lesson learned .....	47
6.4 - Future work.....	48
References.....	50
Appendix A – System Manual.....	51
Appendix B – Design Documentation.....	54

Appendix C – User Manual .....	66
Appendix D – Management Reports.....	75
Appendix E – Test Results.....	78
Appendix F – Client Certificate .....	85
Glossary .....	86
Index .....	87

# List of Figures

Figure 2. 1: Data Flow Diagram for the Existing System.....	7
Figure 2. 2: Interface Design of Microcenter Similar System .....	10
Figure 2. 3: Interface Design of NewEgg Similar System.....	11
Figure 3. 1 System Implementation Architectural Design.....	17
Figure 3. 2 Proposed System Use Case Diagram .....	18
Figure 3. 3: ER Diagram for Proposed System.....	21
Figure 3. 4: Activity Diagram for Login.....	23
Figure 3. 5: Activity Diagram for Add Item .....	23
Figure 3. 6: Login Form UI Design .....	24
Figure 3. 7: Order Checkout form.....	25
Figure 3. 8: Add Items form .....	26
Figure 4. 1: Hardware Environment .....	28
Figure 4. 2: File Structure .....	30
Figure 4. 3 Member Registration Code Segments .....	31
Figure 4. 4 Basic Validation Code Segment.....	32
Figure 4. 5 Advanced Validation Code Segment.....	33
Figure 4. 6 Customer Checkout Code Segment.....	34
Figure 4. 7 Reusable CSS Files.....	35
Figure 4. 8 Reusable Javascript files.....	35
Figure 4. 9 Reusable Image Upload Function .....	36
Figure 4. 10 Reusable Data Clean Function .....	37
Figure 4. 11 Reusable Main Menu Code Segment .....	37
Figure 4. 12 Reusable Sub Menu Code Segments .....	38
Figure 5. 1: User Feedback Form .....	44
Figure 5. 2: User Acceptance Test Evaluation Sample.....	45
Figure 5. 3: Summary of User Feedback Evaluation .....	46

Figure A. 1: XAMMP Server Download Page .....	52
Figure A. 2: Create Database .....	52
Figure B. 1: Use Case Diagram for Customer Registration.....	54
Figure B. 2: Use Case Diagram for Order Checkout .....	56
Figure B. 3 Use Case Diagram for Update Courier Details.....	58
Figure B. 4 Use Case Diagram for Make Payments .....	59
Figure B. 5 Use Case Diagram for Check Items.....	60
Figure B. 6: Activity Diagram for Customer Login .....	61
Figure B. 7: Activity Diagram for Order Checkout.....	62
Figure B. 8 Activity Diagram for Update Courier Details.....	63
Figure B. 9 Activity Diagram for Make Payments .....	64
Figure B. 10 Activity Diagram for Check Items.....	65
Figure C. 1: Home Page.....	66
Figure C. 2: Category Page .....	67
Figure C. 3: List Item Page .....	67
Figure C. 4: Item Page .....	68
Figure C. 5: Cart Page.....	68
Figure C. 6: Checkout Page .....	69
Figure C. 7: Invoice Page.....	70
Figure C. 8: Add Category Form .....	71
Figure C. 9: Add Brand Form .....	71
Figure C. 10: Add Models Form.....	71
Figure C. 11: Add Items Form .....	72
Figure C. 12: Add Employee .....	73
Figure C. 13: Add user modules .....	74
Figure D. 1: Report of Orders .....	75
Figure D. 2: Report of Items .....	75
Figure D. 3: Report of Order Deliveries .....	76
Figure D. 4: Report of Stock.....	76

Figure D. 5: Report of Product Categories .....	77
Figure E. 1: User Acceptance Test Result One.....	82
Figure E. 2: User Acceptance Test Result Two.....	83
Figure E. 3: User Acceptance Test Result Three .....	83
Figure E. 4: User Acceptance Test Result four.....	84

# List of Tables

Table 2. 1: Stakeholders Interview Schedule.....	5
Table 2. 2: Comparison between Proposed System and Similar Systems .....	12
Table 2. 3: Comparison between Different Process Models.....	14
Table 3. 1: Use Case Narrative for Login Module.....	19
Table 3. 2: Use Case Narrative for Inventory Item.....	20
Table 5. 1: Test Case for Customer Registration.....	42
Table 5. 2: Test Results for Customer login .....	42
Table 5. 3: Test Result for Checkout Process .....	43
Table A. 1: Hardware Requirements .....	51
Table A. 2: Software Requirements .....	51
Table B. 1: Use Case Narrative for Customer Registration.....	56
Table B. 2: Use Case Narrative for Order Checkout .....	57
Table B. 3 Use Case Narrative for Update Courier Details.....	58
Table B. 4 Use Case Narrative for Make Payments .....	59
Table B. 5 Use Case Narrative for Check Items .....	61
Table E. 1: Test Result for Customer Registration .....	79
Table E. 2: Test Result for Customer Login .....	79
Table E. 3: Test Result for Checkout Process.....	82

# List of Acronyms

**AJAX** – Asynchronous JavaScript and XML

**BIT** – Bachelor of Information Technology

**CC** – Creative Cloud

**CD** – Compact Disc

**CSS** – Cascading Style Sheets

**ER** – Entity Relationship

**FIFO** – First In First Out

**HTML** – Hypertext Markup Language

**ICT** – Information Communication Technology

**IDE** – Integrated Development Environment

**PEB** – Project Evaluation Board

**PHP** – PHP Hypertext Preprocessor

**RUP** – Rational Unified Process

**SQL** – Structured Query Language

**UAT** – User Acceptance Testing

**UCSC** – University of Colombo School of Computing

**UI** – User Interface

**UX** – User Experience

**UML** – Unified Modeling Language

**URL** – Uniform Resource Locator

**XAMMP** – Apache + Maria DB + PHP + Perl

**XP** – Extreme Programming

# **Chapter 1 – Introduction**

## **1.1 Chapter Introduction**

This chapter provides an overview of U-Star Digital and the client's planned system. This will also cover the background and rationale for starting the project, as well as the project scope and objectives, and also the dissertation format.

## **1.2 Introduction to U-Star Digital**

Many types of computers, including desktops, laptops, notebooks, palmtops, and software, as well as peripherals like printers, scanners, and keyboards, are sold and serviced in the computer sales and servicing industry. Before being supplied to end-users, such as households and businesses, these products are frequently purchased from domestic and international producers and distributors.

U-Star Digital is a renowned information and communication technology (ICT) company that has been in existence since 2010. They provide an unrivaled program and experience in ICT equipment repair, servicing, maintenance, and cleaning. They rely on its technologically cutting-edge next-generation infrastructure to produce best in class lifestyle-enhancing and customer-aware goods and services that account for customers' needs.

## **1.3 Motivation of the Project**

Customers used to come into the store and convey their needs to the technician who built their computer under the old system. Customers arrived at the shop, handed over the defective item or computer to the hardware professionals, and then waited in their restroom until the job was completed. To take online orders, they currently utilize a phone-based and WhatsApp-based order system. There are no other options than bank transfers, and the customer must send legal transaction paperwork to confirm the order. Customers are not given accurate delivery information due to a lack of appropriate delivery information.

Customers are mobile, busy, and always on the go. They want frictionless and seamless services. Developing a thorough understanding of consumer demands they are provided with a convenient product browsing experience, a simple payment system, and a delivery tracking system that allows customers to track their order as it arrives at their home.

An outstanding solution for U-Star Digital to continue their greatness in the computer sales and maintenance sector in this digital age can be added. By the same token, while pursuing a Bachelor's degree in Information Technology, this effort would substantially assist me in improving my project management, system analysis, design, and development abilities.

## 1.4 Objectives of the Project

The study's overall goal is to build and develop an online system for managing computer hardware and services that will assist the company and its employees in becoming more popular with their consumers and marketing their business through the internet.

The study's particular objectives are as follows:

- Providing a feature-rich product browsing method, as well as a simple payment and delivery mechanism, to fulfill the customer's order.
- Inventory management, which makes it easy to manipulate items for the store and receive automated alerts when stock levels are low.
- Generate a variety of reports and graphics, including sales and stock data, to assist management in making decisions.
- Using an automated question-based module to implement Online Troubleshooting, and suggest the appropriate solution for resolving the problem.

## 1.5 Scope of the Project

Web-Based Computer Hardware Purchasing and Troubleshooting Assistant management system has a broad reach that includes the following points of view.

### 1.5.1 Shopping Cart Module

This module allows users to find the entire store for items and create customer orders using a shopping cart.

### 1.5.2 Payment Module

This module provides up-to-date information on all payments made by customers, as well as advanced payment filtering options and having bank transfers, and cash on delivery options.

### **1.5.3 Inventory Management Module**

Has the ability to handle the complete inventory. Individual items and corresponding categories, brands, and models can be added, deleted, and updated.

### **1.5.4 Delivery Management Module**

This module contains all of the customer-created delivery records in the system. It displays the order tracking number as well as information about the courier company.

### **1.5.5 Troubleshooting Assistant Module**

With the help of this virtual assistant, consumers may troubleshoot their hardware failures with the most accurate videos and the description.

### **1.5.6 Invoice Generating Module**

In the system, this module generates a variety of invoices. It's compatible with the payment, computer hardware assistant, and troubleshooting modules.

### **1.5.7 User Management Module**

This module helps to manage all of the users of the system. It aids in the selection of the right individual for the right position.

### **1.5.8 Report Management Module**

This module will make it possible to generate, preview, and print a variety of reports in order to continue with the success of the business.

The warranty claim option and the work card for the repair of the computer parts were the projects that went outside of the bounds. Additionally, it creates a link to the troubleshooting form, making it simple to schedule appointments for warranty and repair inspections.

## **1.6 Structure of Dissertation**

The introduction is the first of six chapters in the dissertation. Below is a brief list of each chapter's contents.

### **Chapter 1 – Introduction**

The project's goals and scope are explained in this chapter. It also explains the present system's flaws and what motivates the creation of this system.

## **Chapter 2 – Analysis**

This chapter covers requirement collection strategies for a new system's functional and non-functional criteria, as well as analysis approaches for understanding the existing system.

## **Chapter 3 – Design**

The system's design, including database and user interface design, is covered in this chapter. To make the system's structure easier to understand, additionally this chapter includes top-level and module-level usage examples.

## **Chapter 4 – Implementation**

The structure of important code modules is covered in this chapter along with implementation technologies, hardware, and software specifications.

## **Chapter 5 – Evaluation**

This chapter describes the test cases used to evaluate the system and the results. It gives a description of the use cases and test cases.

## **Chapter 6 – Conclusion**

This chapter summarises the key takeaways and offers ideas for improving the system going forward.

A general index and a glossary of terms are also included.

# Chapter 2 – Analysis

## 2.1 Chapter Introduction

The process of identifying the underlying problem and comprehending the problem domain from a jumble of facts and figures is known as analysis. The purpose of the analysis is to provide a complete, consistent, and unambiguous image of the system. Also, what should the system supply to meet all user requirements. Requirements Engineering is another name for this procedure. The system can be designed with the help of analysis.

## 2.2 Fact Finding Techniques

To accurately describe and interpret system requirements, fact-gathering techniques are information-gathering techniques used in system analysis. The following strategies were primarily employed:

1. Interviews
2. Observation
3. Existing Documents Analyzing

### 2.2.1 Interviews

The initial technique utilized to collect and establish system requirements, as well as clarify and confirm those needs, was an interview. In order to effectively address and weigh the inputs of each interviewee, the interviewer must first comprehend the perspective of each interviewee.

The following is the interview schedule that was used (Table 2.1).

Date	Interviewee	Position	Interview Duration
23/10/2021	Mr. Samudu Kannangara	Owner	40 minutes
23/10/2021	Mr. Upali Kannangara	Stock Keeper	30 minutes
23/10/2021	Mr. Thamara Kannangara	Technician	40 minutes
25/10/2021	Mr. Susith Sewikrama	Delivery Manager	20 minutes
25/10/2021	Mr. Chamara Perera	Technician	30 minutes
27/10/2021	Miss. Kalpani Dinusha	Customer	20 minutes

Table 2. 1: Stakeholders Interview Schedule

### **2.2.2 Observation**

Observation is a fact-finding technique in which system analysts observe how individuals perform tasks and activities during site visits. This is an excellent approach to learn what end users go through in their day-to-day processes, and it gives you a lot of insight into the business process.

### **2.2.3 Existing Documents Analyzing**

Analyzing existing papers is a key strategy for acquiring requirements. When building a user-friendly system, evaluating the papers and reports of an existing system can help. Solid information and facts are usually collected by studying existing papers, which helps to corroborate and validate the requirements gathered through other methods.

## **2.3 Existing System**

From the beginning, U-Star Digital has used a semi-paper-based manual approach. During the system analysis phase, it was discovered that the current system can only perform a limited number of functions.

Customers used to come into the store and convey their needs to the technician who built their computer under the old system. Customers arrived at the shop, handed over the defective item or computer to the hardware professionals, and then waited in their restroom until the job was completed. To take online orders, they currently utilize a phone-based and WhatsApp-based order system. There are no other options than bank transfers, and the customer must send legal transaction paperwork to confirm the order. Customers are not given accurate delivery information due to a lack of appropriate delivery information.

## **2.4 Existing System Data Flow Diagram**

There were numerous downsides to the paper-based technique.

- It was discovered that data was repeating.
- Data that is unidentifiable owing to illegible handwriting.
- Data was lost because of misplacing the recording materials.
- It takes a long time.
- Manual computations are required.
- Data security is lacking.

- More manpower is required.

The following data flow diagram illustrated by existing system (Figure 2.1).

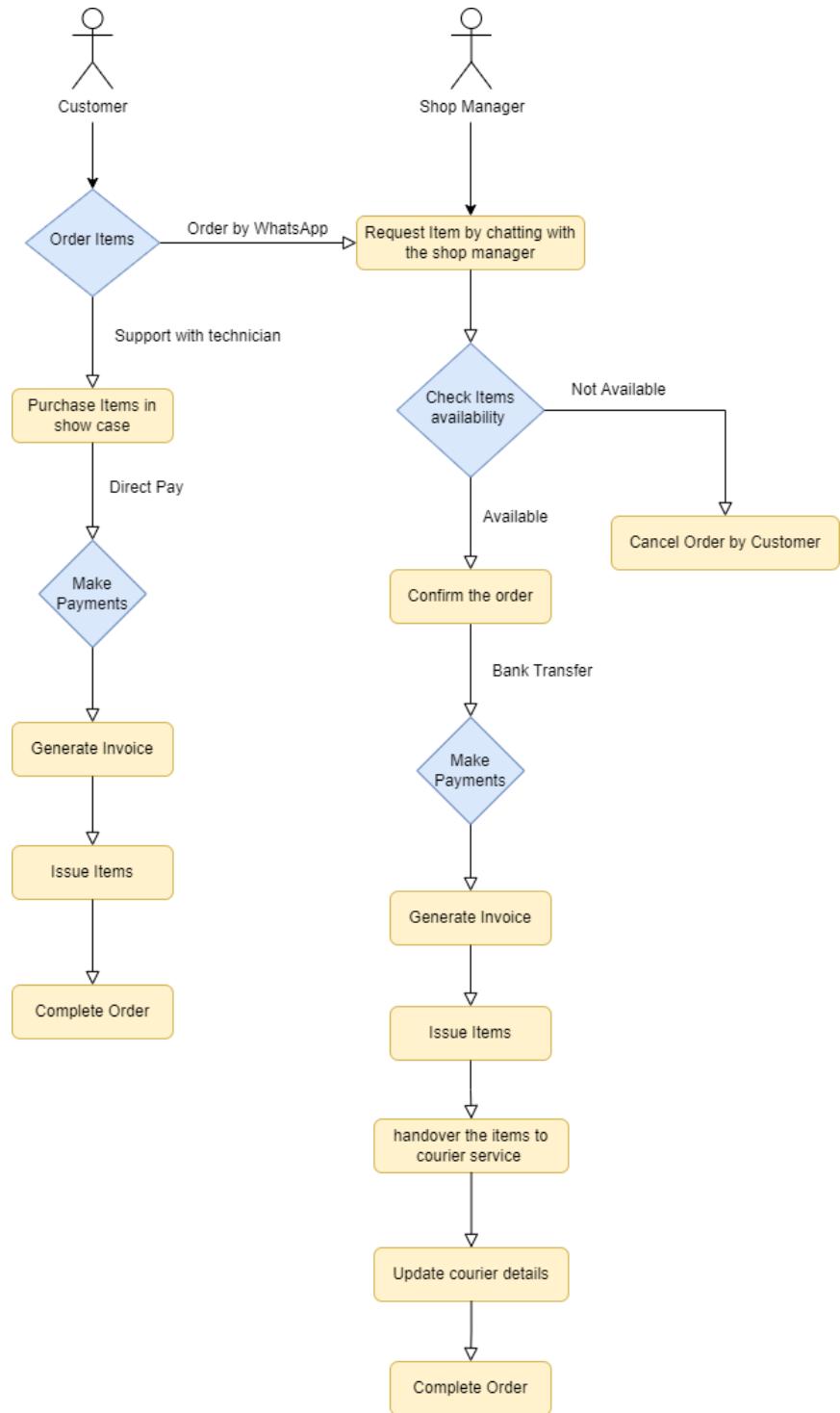


Figure 2. 1: Data Flow Diagram for the Existing System.

## 2.5 Requirements Analysis

### 2.5.1 Functional Requirements

Calculations, data insertion, manipulation, and processing, as well as data presentation, are all defined by functional requirements. The system's required features are listed below.

- **Shopping Cart**

- Customer search and select items to add to the cart.
- Customers can make the order in the selected items in the cart.
- Calculates selected items' totals and discounts.
- Orders are recorded in the Customer Dashboard Orders section and Shop Managers Orders section.
- Able to update order status.
- Notify the shop manager of a new order.

- **Payment**

- Customers can select the payment method.
- Carry the order totals to the payment method.
- Proceed to checkout with total payment.

- **Inventory Management**

- Insert, update and delete the Items, Brands, Models, and Item Specifications.
- Notify with low stock and out-of-stock Items.
- Able to create a low stock threshold.
- Able to create discounts with a selected number of date periods.

- **Delivery Management**

- Able to create courier companies.
- Customers can notify with courier company details and order tracking numbers when the order is dispatched.
- Customers can view their order details by giving the courier company a tracking URL using the tracking number.
- Customers were able to update the courier process when receiving the item.

- **Troubleshooting Assistant**
  - Customers select the type of defect.
  - Load appropriate answer form already configerd in the database.
  - Show the answers related to the customer selected answer.
- **Invoice Generation**
  - Calculate order totals and courier payment and generate an invoice for the customer.
  - Invoices can filter with the selected date range.
  - Invoices can export and have the ability to print.
- **User Management**
  - Able to manage all of the users of the system.
  - Aids to create user roles and permission.
- **Report Management**
  - This module will make it possible to generate, preview, and print a variety of reports.
  - Able to export and filter the reports within the selected time frame.

### **2.5.1 Non-Functional Requirements**

Non-functional requirements had to be met in order for a system to be valuable and successful.

- **Usability**

The system is directly connected to all of the shop's consumers. Users of the system can access necessary reports over the internet. Giving clients better service means making data easier to access and sending notifications about existing purchases.
- **Maintainability**

It's simple to correct a bug, add new features, make changes to support new operating systems, and make the software easier to maintain for others. It also improves the contentment of members.
- **Performance**

The system should be quick and finish the task in the shortest time possible. Data should be retrieved as soon as feasible, and the response time should be minimal.

- **Security**

Being that the system saves client personal data including e - mail addresses, contact information, and mailing addresses, security is the major concern for the system to shield users' sensitive data.

- **Reliability and accuracy**

The system's major features include the generation of progress reports and workout programs. As a result, the system's accuracy and dependability are critical.

## 2.6 Similar Systems and Literature Reviews

It was required to examine the existing system before designing the web-based Computer Hardware Purchasing and Troubleshooting Assistant management system. Because the current system is semi-paper-based, a literature review and related system analysis were also decided. The systems listed below were evaluated in order to have a thorough understanding of the situation.

### 2.6.1 Microcenter



*Figure 2. 2: Interface Design of Microcenter Similar System*

More computers, electronics, networking, and communication devices are available at Micro Center than from any other company more than 30,000 items are in stock. Micro Center is incredibly passionate about offering products and services for supporting technology. Since 2010, we have provided in-store pickup for online orders. [1]

## 2.6.2 Newegg

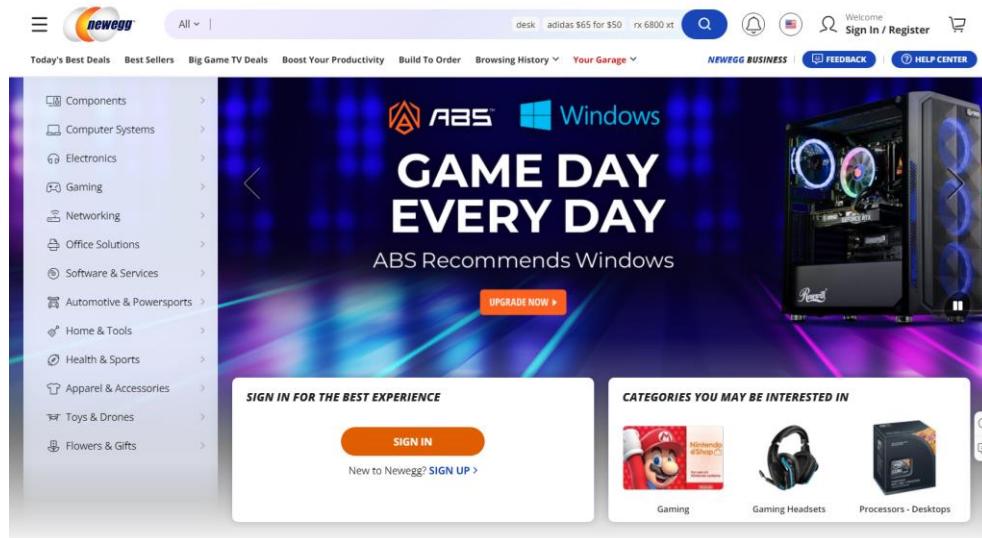


Figure 2. 3: Interface Design of NewEgg Similar System

“Today, millions of customers turn to Newegg to shop for the latest PC components, consumer electronics, smart home and gaming products. Newegg is consistently ranked as one of the best online shopping destinations, and the company regularly earns industry-leading customer service ratings.” [2]

## 2.6.3 Comparison of the Proposed System to a Comparable System

The suggested system and the related systems that were mentioned were very similar. The system's primary benefit was the ability to repair a user's computer without having to go to a computer hardware store.

When compared to the micro centre and the system, the computer hardware assistant is the more alluring feature. It enables customers to construct their computers virtually and obtain construction quote prices.

Below are some examples of similar systems' most prominent components. (Table 2.2)

Functionality	Proposed System	Microcenter	NewEgg
Shopping Cart	Yes	Yes	Yes
Payment Management	Yes	Yes	Yes
Inventory Management	Yes	Yes	Yes
Delivery Management	Yes	Yes	Yes
Troubleshooting Assistant	Yes	No	No
Computer Hardware Assistant	No	Yes	No

Mobile App	No	No	Yes
------------	----	----	-----

*Table 2. 2: Comparison between Proposed System and Similar Systems*

## 2.7 Model of the Proposed System's Process

The organisation of software development activity into phases for better design, project management, and product management is known as the software development process, sometimes known as the software development lifecycle.

Scrum, Extreme Programming (XP), RUP, and other process models exist for developing software systems.

### 2.7.1 Scrum

“The Scrum process organizes development into a sequence of sprints, each of which results in a potentially usable product with an added increment of function. The tasks for each sprint are set, in consultation with a stakeholder representative, during a sprint planning meeting and cannot be added to during the sprint. Each task is typically expressed as a user story. Each sprint is time boxed: the end date of the sprint does not change. Tasks that cannot be accomplished in time are returned by the team to the backlog for future consideration.” [3]

### 2.7.2 Extreme Programming (XP)

“Extreme Programming (XP) is an agile software development framework that aims to produce higher quality software, and higher quality of life for the development team. XP is the most specific of the agile frameworks regarding appropriate engineering practices for software development.” [4]

### 2.7.3 Rational Unified Process (RUP)

“Rational Unified Process” is what it stands for. RUP is a method for creating software that was created by IBM's Rational division. The development process is divided into four separate phases, each of which includes business modelling, analysis and design, implementation, testing, and deployment. There are four phases:

1. **Inception** - The project's concept is presented. The project's development team decides if it is worthwhile to pursue it and what resources will be required.

2. **Elaboration** - The architecture of the project and the needed resources are assessed further. The software's potential uses and development expenses are taken into account by the developers.
3. **Construction** - The project has been created and finished. The programme has been created, developed, and tested.
4. **Transition** - The world can now use the software. Final modifications or upgrades are done in response to customer feedback.

Companies can develop software products in an organised manner using the RUP development approach. It helps avoid resource waste and lowers unforeseen development expenses by providing a detailed strategy for each stage of the development process.” [5]

Based on the considerations outlined in Table 2.3 over various strategies, the Rational Unified Process (RUP) was selected as the most effective process model for the suggested system.

There were further justifications for choosing the RUP for the mythology of system development. The client's requirements are stable during requirement collection. The project scope is fixed as a result. Analyzing the requirements revealed that the project would be substantially larger than anticipated. The planning phase was the most beneficial. The milestones were the driving force. The project required the documentation procedure. The RUP contributes to maintaining the precise documentation necessary for the project's success.

RUP	Scrum	XP
A good option for a huge, long-term project.	There is no set termination date. At the end of the current iteration, the next iteration plan is determined.	Release regularly.
The planning process is driven by the end date and includes intermediate milestones.	Scrum employs a project backlog instead of a scope.	A shorter time limit
Requirements are fixed	It's ideal for rapid upgrade tasks that don't have to be completed by a certain date.	Expecting adjustments in requirements.

The scope of the project is predetermined.	Documentation is severely limited.	There is a lack of general design.
Documentation is required.		There is little to no documentation.

*Table 2. 3: Comparison between Different Process Models*

# **Chapter 3 – Design**

## **3.1 Chapter Introduction**

The overall structure of the suggested system is covered in the design chapter. The database of the system, relevant Use Case Diagrams, Entity Relationship Diagrams, and other UML Diagrams, as well as the tools, methodologies, and design procedures utilised in the design phase are all described in this chapter.

## **3.2 Alternate Solution**

Alternatives to the Web-Based Computer Hardware Purchasing and Troubleshooting Assistant management system include a mobile application, the use of a software collection, or the continued use of the current manual system.

### **3.2.1 Stand-alone System**

Software that is not bundled with another programme and can be used without an internet connection is known as stand-alone software (work offline). It is not very useful for the Web-Based Computer Hardware Purchasing and Troubleshooting Assistant management system because outdoor service management cannot be carried out without the use of the internet, it is more expensive, and installation and maintenance are more difficult than with a Web-Based System.

### **3.2.2 Collection of Software**

The applications in the following list may be helpful in managing the Computer Hardware Purchasing and Troubleshooting Assistant management system.

- Computer Hardware Assistant can help the lack of knowledge customers to select the perfect item to build their computer.
- Inventory can manage individual objects as well as their corresponding categories, brands, and models.

### **3.2.3 Justification for Choosing a Web-Based System**

The web-based system outperformed the standalone system in several ways. because the characteristics of online shopping carts were the project's declared scope. It's simple to keep track of the system's progress from any location. The system would be platform-independent. Because the database is centralized and everything is synchronized,

maintenance is simple. Allow users to access the system at any time and from any location. It also allows for the usage of a wide range of devices to access the system.

### 3.3 The Architectural Design of the System

As a Procedural Programming Design Concept, the system will be developed utilizing the Non-Object Oriented technique.

In a procedural language, a program is a list of statements, each of which instructs the computer to perform a certain task. It concentrates on the technique (function) and algorithm that are required to complete the derived computation. When a program grows in size, it is divided into functions, each with a distinct purpose. One of the fundamentals of structured programming is the division of the program into functions and modules.

Procedural-oriented programming has several characteristics.

- The emphasis is on getting things done (Functions).
- The program is divided into many functions.
- Passing parameters across functions allows them to communicate with one another.
- Global variables are shared between functions.
- The procedure calls are based on the notion.
- When it comes to program design, it takes a top-down approach.

The system implementation architectural design (Figure 3.1)

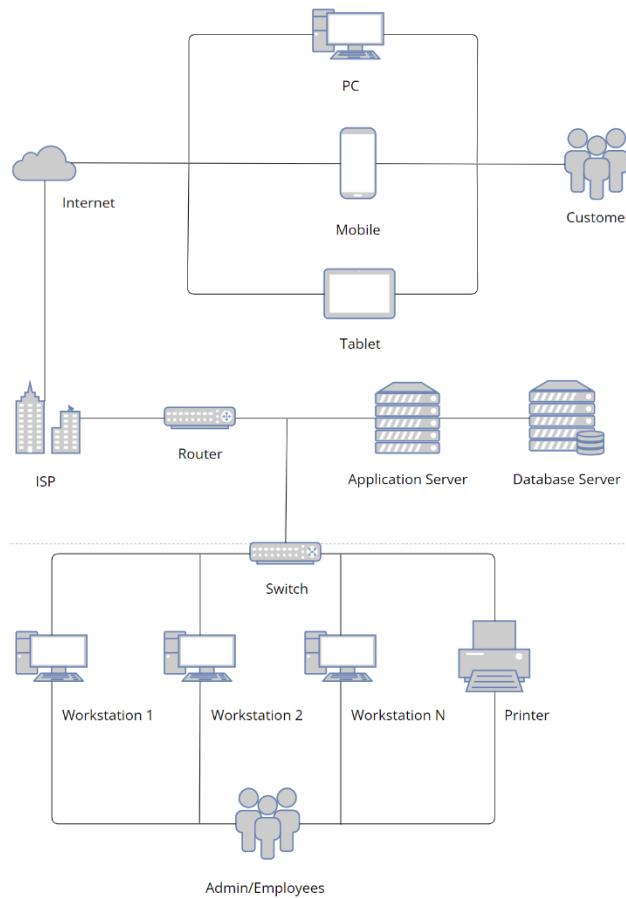


Figure 3. 1 System Implementation Architectural Design

### 3.4 Diagram of the Use Case for the Proposed System

The following is a high-level Use Case Diagram for the Proposed System.

The system's actors:

**Admin:** Create users and manage user permissions in the system.

**Shop Manager:** Manage all the orders made by the customers.

**Inventory Manager:** Manage the inventory of the store.

**Delivery Manager:** Manage all deliveries and register new courier companies

**Technician:** Match the computer hardware and manage the troubleshooting questions.

**Customer:** Make orders.



Figure 3. 2 Proposed System Use Case Diagram

### 3.5 Use Case Narratives for the Proposed System

The development of several system sections is guided by the use case narratives that follow. The most important narratives are only added in this part. There are more narratives added to section B of the appendix.

#### 3.5.1 Use Case Narratives for Login Module

The login module's whole use case narrative (Table 3.1)

<b>Use-case Number</b>	UC-04	
<b>Use-Case Name</b>	Login	
<b>Priority</b>	High	
<b>Actor</b>	Admin, Shop Manager, Inventory Manager, Delivery Manager, Technician, Customer	
<b>Description</b>	This use case explains how customers and shop employees log into the system.	
<b>Precondition</b>	All actors are properly registered to the system.	
<b>Post-condition</b>	If the use case was successful, the actor is now logged into the system. If not, the system state is unchanged.	
<b>Basic course of Action</b>	<b>User Action</b>	<b>System Response</b>
	1. The Actor is on the login page to log in to the system. 3. The actor click the "Login" button after entering his username and password.	2. Actor is encouraged by the system to input their username and password. 4. The system verifies that all the filled have been filled out and are valid. 5. The system was able to log in successfully. 6. Use case Exit
<b>Alternate course of Action</b>	4.1 The system alerts the user with the message "Invalid Username or Password" and then proceeds to step 3 of the basic course of action to input again if all fields are not filled out and do not match the username and password.	

Table 3. 1: Use Case Narrative for Login Module

### 3.5.2 Use Case Narratives for Insert Inventory Item

The insert inventory item module's whole use case narrative (Table 3.2)

<b>Use-case Number</b>	UC-08	
<b>Use-Case Name</b>	Insert Inventory Item	
<b>Priority</b>	High	
<b>Actor</b>	Inventory Manager	
<b>Description</b>	How to manage inventory items is explained in this use case.	
<b>Precondition</b>	None	
<b>Post-condition</b>	The actor can add the product to the system and configure it to sell to clients if the use case was successful.	
<b>Basic course of Action</b>	<b>User Action</b> 1. Actor clicks the “Add Item” in the Item Management section. 3. Actor fill the fields (Item Image, Item Name, Category, Brand, Model, SKU number, Insert variable specifications, reorder level, Unit price, and Sale price) 4. Actor click “Insert Item”	<b>System Response</b> 2. System prompts the form to insert details to the actor. 5. System validates the Item Name and SKU already in the system. 6. System display massage “New Item Insert”
<b>Alternate course of Action</b>	5.1. If an item is already inserted into the system display the popup message “This Item Already in the System”.	

Table 3. 2: Use Case Narrative for Inventory Item

### 3.6 ER Diagram for the Proposed System

The high-level ER Diagram below depicts the relationships between the proposed system's constituents. (Figure 3.3)

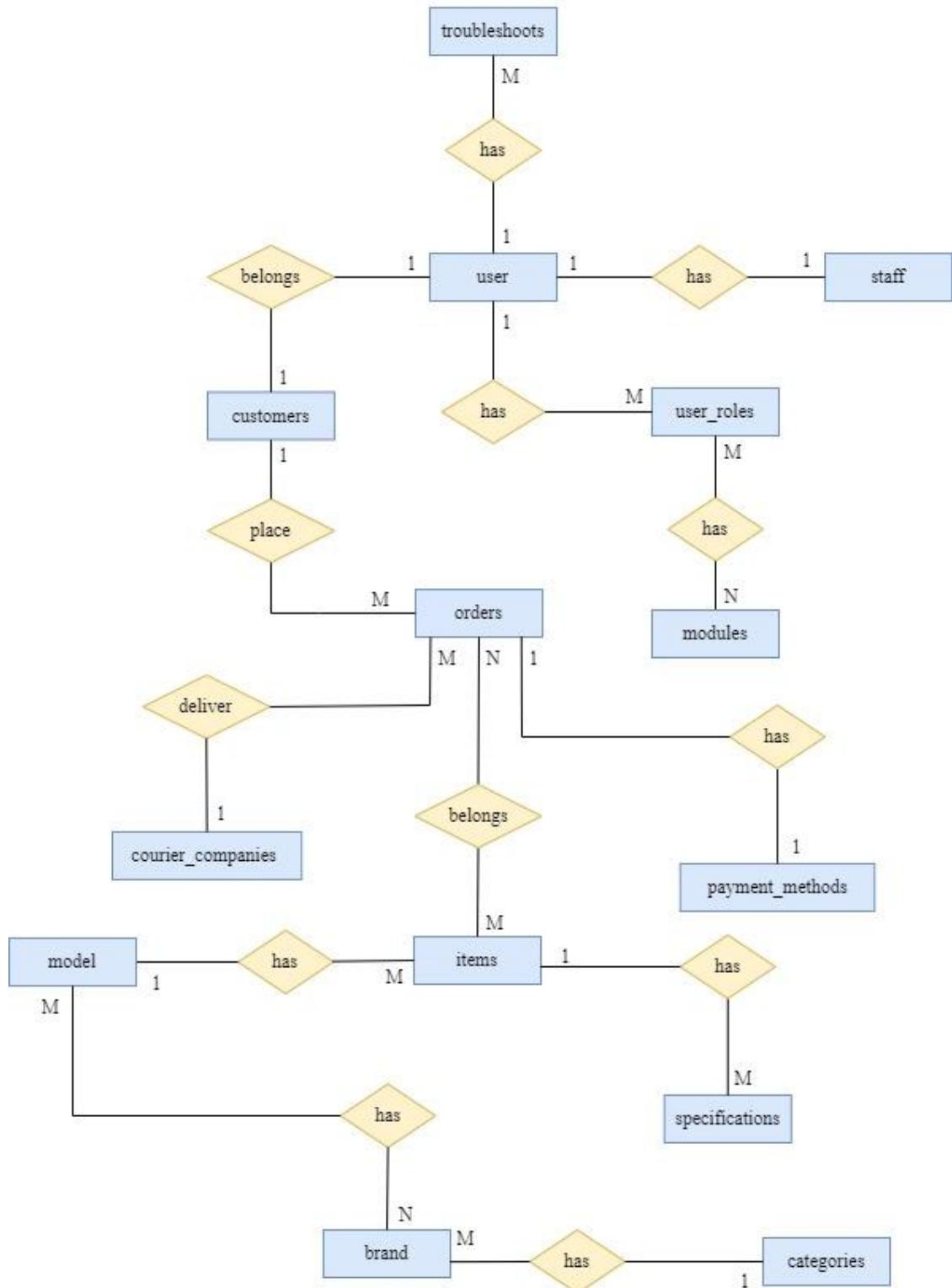


Figure 3. 3: ER Diagram for Proposed System

### 3.7 Relational Schema for System

- USERS (user\_id, user\_name, email, password, first\_name, last\_name, profile\_image, created\_date, status\*, user\_role\*, u\_notification\*)
- CUSTOMERS (cus\_id, contact\_number, postal\_code, address\_l1, address\_l2, city, province, user\_id\*)
- STAFF (staff\_id, nic, dob, contact\_number, address\_l1, address\_l2, city, postal\_code, user\_id\*, status\*)
- USER\_ROLES (user\_role\_id, role\_name, status\*)
- TROUBLESHOOTS (troub\_id, troub\_name, troub\_description, troub\_video)
- MODULES (module\_id, description, path, view, icon, status\*)
- ORDERS (order\_id, order\_number, order\_total, total\_discount, grand\_total, order\_date, order\_time, user\_id\*, delivery\_charge\*, payment\_id\*, courier\_status\*)
- PAYMENT\_METHODS (id, name, description)
- COURIERS\_COMPANIES (company\_id, company\_name, contact\_number, contact\_number\_opp, email, address\_line\_1, address\_line\_2, tracking\_url, status\*)
- ITEMS (item\_id, item\_image, item\_name, sku, reorder\_level, grn\_price, unit\_price, sale\_price, discount\_rate, item\_description, date, stock, warranty\_period, category\_id\*, brand\_id\*, model\_id\*, status\*)
- SPECIFICATIONS (spec\_id, spec, category\_id\*, status\*)
- MODELS (model\_id, model\_name, category\_id\*, status\*)
- BRANDS (brand\_id, brand\_name, status\*)
- CATEGORIES (category\_id, category\_name, category\_description, cat\_image, status)

## 3.8 Activity Diagrams for the Proposed System

### 3.8.1 Activity Diagram for Login

The activity diagram for creating workouts is shown in Figure 3.4.

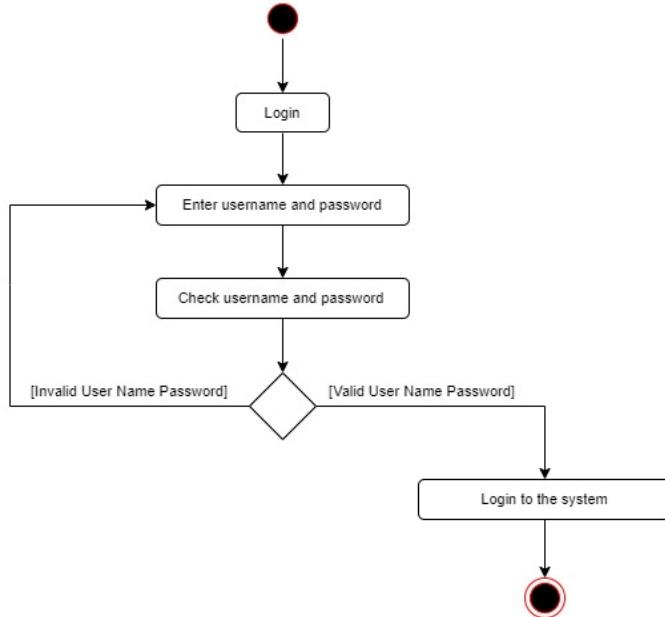


Figure 3. 4: Activity Diagram for Login

### 3.8.2 Activity Diagram for Add Item

The activity diagram for adding a new item to the inventory is shown in Figure 3.5.

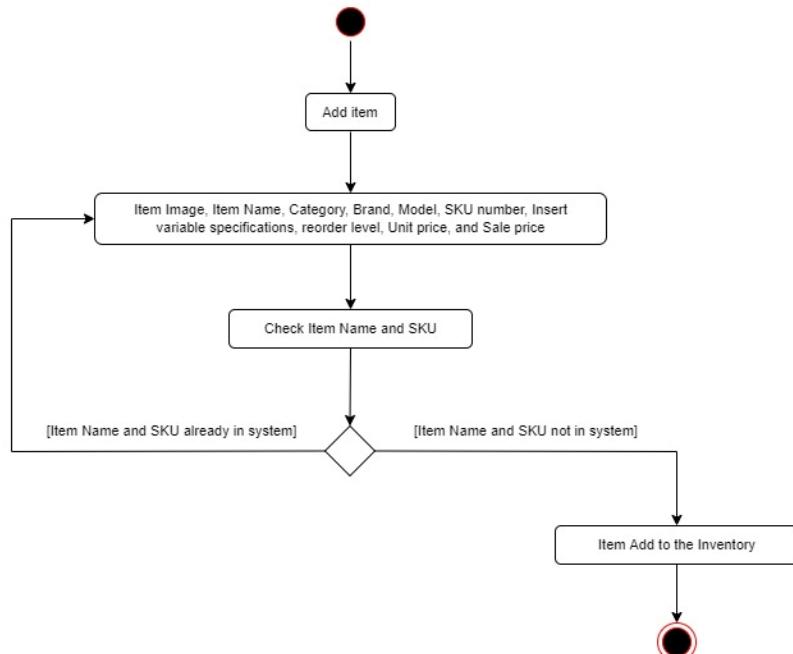


Figure 3. 5: Activity Diagram for Add Item

## 3.9 – UI Designs for the Proposed System

### 3.9.1 – Customer Login and Register Form UI Design

Customer my account login and register form (Figure 3.6)

The screenshot displays the U-Star Digital website's login and register forms side-by-side. At the top, a black header bar includes the logo, navigation links (Home, Shop, About, Services, Contact), a 'My Account' link, and a 'Cart 1' icon. Below the header, the left panel shows the 'Login' form with fields for 'Email' (nishan) and 'Password' (\*\*\*\*\*), and a 'Sign In' button. The right panel shows the 'Register' form with fields for 'First Name', 'Last Name', 'Email' (nishan), 'Password' (\*\*\*\*\*), and 'Retype password', followed by a 'Register' button. The footer contains a company bio, quick links, and social media icons.

Figure 3. 6: Login Form UI Design

### 3.9.3 – Customer Checkout UI Design

Checkout page full view of the UI design (Figure 3.7)

The screenshot displays the Order Checkout form for U-Star Digital. At the top, there is a navigation bar with links for Home, Shop, About, Services, Contact, My Account, and Cart (Cart 1). The main form is divided into two sections: 'Enter Your Billing Details' and 'Enter Your Delivery Details'. Both sections contain fields for First Name, Last Name, Phone, Email Address, Address Line 1, Address Line 2, City, Province, and Zip. A checkbox labeled 'Use same address as a delivery address' is present. Below these sections is a 'Payment Method' section with radio buttons for 'Cash On Delivery (COD)' and 'Direct Bank Transfer'. To the right is an 'Order Summary' table showing Item(s) LKR: 66,500.00, Discount LKR: (-4,000.00), Est Total LKR: 62,500.00, Delivery Charges: LKR: 0, and a final Total LKR: 62,500.00. A 'COMPLETE ORDER' button is located at the bottom of the summary table. At the very bottom of the page is a footer with the company logo, a brief description of their services, and links to Company, About, Contact, Service, and FAQ pages.

Item(s):	LKR: 66,500.00
Discount:	LKR: (-4,000.00)
Est Total:	LKR: 62,500.00
Delivery Charges:	LKR: 0
Total:	LKR: 62,500.00

Figure 3. 7: Order Checkout form

### 3.9.3 – Add Inventory Item Form

The form displays the inventory management user interface for adding products to the inventory (Figure 3.8).

The screenshot shows the 'Add Items' form in the U-Star Digital inventory management system. The left sidebar contains navigation links for Brand Management, Inventory Reports, Model Management, Category Management, Item Specifications, and Items Management. The main area has a blue header bar labeled 'Insert New Item'. The form is divided into two sections: 'Item Details' and 'Item Specifications'. The 'Item Details' section includes fields for Item Image (with a 'Choose File' button), Item Name (containing 'ASUS STRIX GAMING RADEON RX6700XT 12GB'), Category (PROCESSORS), Brand (dropdown menu showing '- Select Brand -'), Model (dropdown menu showing '- Select Model -'), SKU (Unique Code), Stock (Enter Stock Count), Reorder Level (Enter Reorder Level), GRN Price (158500), Unit Price (165800), Sale Price (163000), and Warranty Period in Days (365). The 'Item Specifications' section includes fields for Cache (Cache, Enter Cache), Total L2 Cache (Total L2 Cache, Enter Total L2 Cache), Total L3 Cache (Total L3 Cache, Enter Total L3 Cache), Processor Cores (Processor Cores, Enter Processor Cores), and Max Turbo Frequency (Max Turbo Frequency, Enter Max Turbo Frequency). At the bottom of the form is a blue 'Insert' button. Below the form is a table titled 'Available Items' with a search bar containing 'intel'. The table has columns for Item Image, Item Name (INTEL CORE i9-12900K PROCESSOR), Status (Active), View, Edit, Inactive, and Active. The 'Edit' column contains a green button with a pencil icon, the 'Inactive' column contains a red button with a crossed-out icon, and the 'Active' column contains a yellow button with a checkmark.

Item Image	Item Name	Status	View	Edit	Inactive	Active
	INTEL CORE i9-12900K PROCESSOR	Active				

Figure 3. 8: Add Items form

# **Chapter 4 – Implementation**

## **4.1 – Chapter Introduction**

In comparison to the previously described analysis and design of the system, implementation was the phase in which U-Star Digital carried out the developing and real-world testing of the Web-Based Computer Hardware Purchasing and Troubleshooting Assistant Management System. In order to maintain the system's quality and security, best practices were followed during implementation and development.

## **4.2 – Implementation Environment**

There are two primary implementation environments for a system when it comes to implementation. There are two types of implementation environments: software and hardware. To keep implementation and maintenance costs low, it was mostly done with free and open-source software and tools. The tools and procedures utilized in this phase are listed below.

### **4.2.1 – Software Environment**

The work is done on a computer that runs the Microsoft Windows operating system. As a result, a software environment that is compatible with Windows was established. The XAMPP server was utilized because it comes with Apache, MySQL, and PHP on Windows. Visual Studio Code IDE has been used as the primary design tool for development. An open-source integrated development environment (IDE) for programming languages including Java, PHP, C++, and others is called Visual Studio Code. The most recent iteration of the most well-liked HTML, CSS, and JavaScript framework for building responsive, mobile-first websites is called Bootstrap 5.

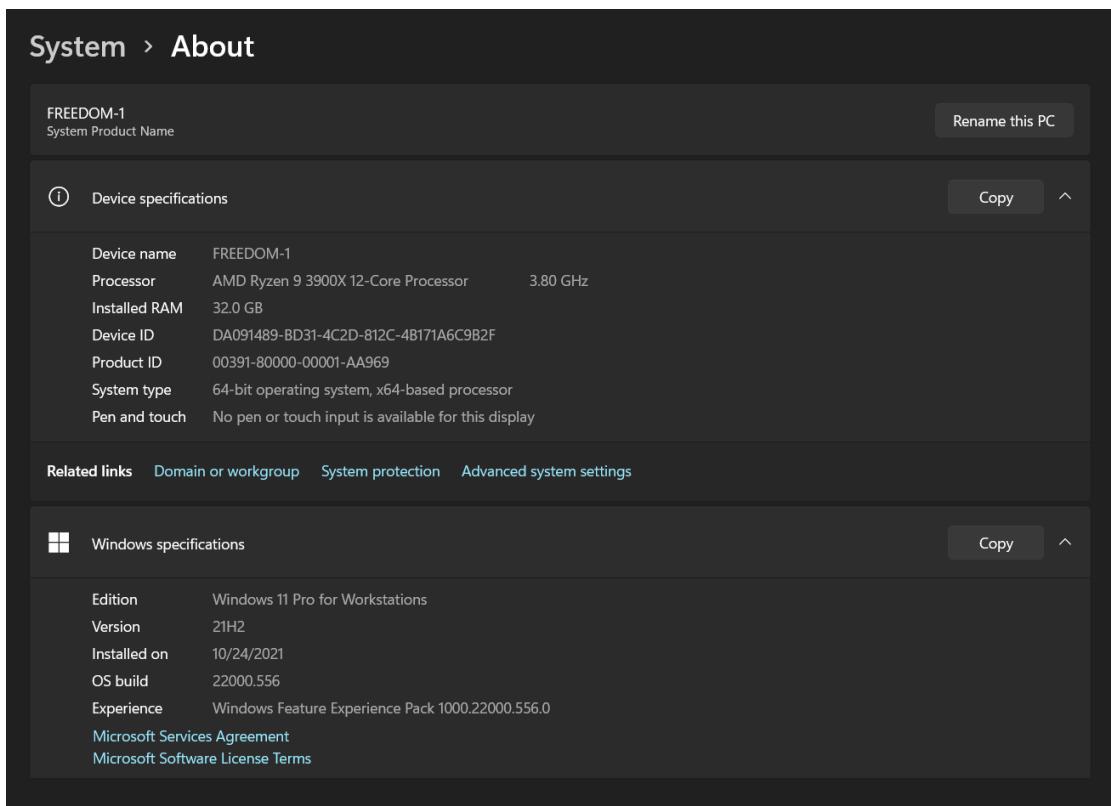
Although PHP is the primary development language, it is supported by several other languages.

- HTML – to create the web pages' content
- JavaScript – for content validation and filtering
- CSS – to make the web pages styles
- JQuery – Code libraries that can be reused
- Ajax – to make changes to the content without having to reload the page

- diagrams.net – to sketch a design diagram
- GanttPRO – to create a project timeline
- Microsoft Word – to compose a documents

#### 4.2.2 – Hardware Environment

The computer used to build this system has the properties portrayed in Figure 4.1.



*Figure 4. 1: Hardware Environment*

#### 4.2.3 – Other Software and Development Tools

Was used authoring tools such as Adobe Photoshop CC for designing graphics and web browsers such as Firefox, Google Chrome, and Internet Explorer for testing the system, in addition to software development tools.

### 4.3 – Justification for the choice of the implementation platform

- **Web Server**

The project's dedicated server, XAMPP Server, was utilized to manage the application using phpMyAdmin. The web server was handled by Apache, which is another part of its solution stack. It helps to get server side language to execute smoothly.

- **Database Server**

The software's database server was MySQL, which is also part of the XAMPP solution stack. MySQL is a relational database management system that is help to manage system database and database queries

- **PHP**

PHP is used as the entire project backend language. It supports implementing the business logic onto the project and make project more dynamic with user inputs.

- **HTML**

As a web-based project, HTML is more pertinent. It served as the front-end markup language for the project. It was useful for controlling how users may view information.

- **CSS**

CSS was used to style the system. It enhances the project's visual appeal and improves the UX and informational value of specific elements.

- **Bootstrap 5**

To make the system more responsive, Bootstrap 5 was employed. It improved the effectiveness of front-end development.

- **JavaScript**

The front-end user experience was dynamically created using Javascript. It lessened the burden on the project's web application server and assisted in making better use of the web server's resources.

- **JQuery**

In the project's front-end to back-end communication, JQuery is utilized to call ajax methods. It shortened the coding process.

- **Microsoft Word**

Documentation for the project was created using Microsoft Word.

- **Adobe Photoshop CC**

More assistance was given for modifying the photographs with Adobe Photoshop CC.

#### 4.3.1 System File Structure

The file structure of the created system is presented 4.2.

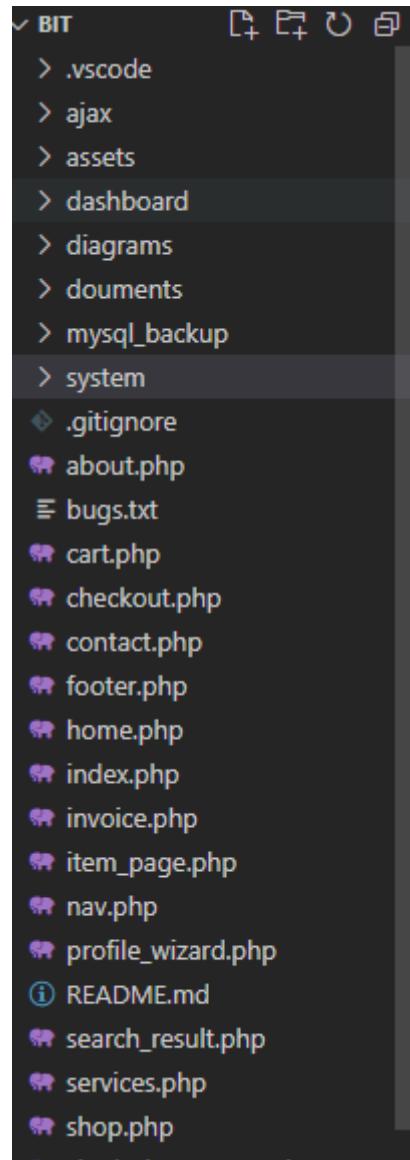


Figure 4. 2: File Structure

#### 4.4 – High Level Module Structure

Inventory Management Module – Accessed by Inventory Manager

Delivery Management Module – Accessed by Delivery Manager

Report Management Module – Accessed by Shop Manager

Troubleshooting Assistant Module – Accessed by Technician

User Management Module – Accessed by Administrator

## 4.5 – Major Code Segments

### 4.5.1 – Customer Registration Code Segments

- Member Registration Form

Customers are registered for the system using the member registration code segment in Figure 4.3. Both checkout.php and myaccount.php make use of it.

```
<form action=<?php echo htmlspecialchars($_SERVER['PHP_SELF']); ?> method="post">
    <div class="input-group mb-3">
        <input type="text" class="form-control" placeholder="First Name"
            name="reg_first_name" value=<?php echo @$reg_first_name ?>>
    </div>
    <div>
        <p style="color: red;"> <?php echo @$error['reg_first_name'] ?> </p>
    </div>
    <div class="input-group mb-3">
        <input type="text" class="form-control" placeholder="Last Name"
            name="reg_last_name" value=<?php echo @$reg_last_name ?>>
    </div>
    <div>
        <p style="color: red;"> <?php echo @$error['reg_last_name'] ?> </p>
    </div>
    <div class="input-group mb-3">
        <input type="text" class="form-control" placeholder="Username"
            name="reg_username" value=<?php echo @$reg_username ?>>
    </div>
    <div>
        <p style="color: red;"> <?php echo @$error['reg_username'] ?> </p>
    </div>
    <div class="input-group mb-3">
        <input type="email" class="form-control" placeholder="Email"
            name="reg_email" value=<?php echo @$reg_email ?>>
    </div>
    <div>
        <p style="color: red;"> <?php echo @$error['reg_email'] ?> </p>
    </div>
    <div class="input-group mb-3">
        <input type="password" class="form-control" placeholder="Password"
            name="reg_password">
    </div>
    <div>
        <p style="color: red;"> <?php echo @$error['reg_password'] ?> </p>
    </div>
    <div class="input-group mb-3">
        <input type="password" class="form-control" placeholder="Retype password"
            name="reg_con_password">
    </div>
    <div>
        <p style="color: red;"> <?php echo @$error['reg_con_password'] ?> </p>
    </div>
</div>
<div class="row">
    <div class="col-8">
        </div>
        <!-- /.col -->
        <div class="col-4" style="display: flex; flex-direction: row; justify-content: flex-end;">
            <button type="submit" class="btn btn-secondary btn-block"
                name="action" value="register">Register</button>
        </div>
        <!-- /.col -->
    </div>
</form>
```

Figure 4. 3 Member Registration Code Segments

- Basic Validation of Customer Registration Form

The basic validation of the customer registration form code segment shown in figure 4.4 is used to check user input before allowing unauthorized data to be entered into the database.

```
// register form
if ($_SERVER['REQUEST_METHOD'] == 'POST' && @$_action == 'register') {

    $reg_username = data_clean($reg_username);
    $reg_first_name = data_clean($reg_first_name);
    $reg_last_name = data_clean($reg_last_name);
    $reg_username = data_clean($reg_username);
    $reg_email = data_clean($reg_email);
    $reg_password = data_clean($reg_password);
    $reg_con_password = data_clean($reg_con_password);

    // basic validation Billing Details
    if (empty($reg_first_name)) {
        $error['reg_first_name'] = "First Name Should Not Be Empty";
    }

    if (empty($reg_last_name)) {
        $error['reg_last_name'] = "Last Name Should Not Be Empty";
    }

    if (empty($reg_username)) {
        $error['reg_username'] = "User Name Should not be empty";
    }

    if (empty($reg_email)) {
        $error['reg_email'] = "email Should Not Be Empty";
    }

    if (empty($reg_password)) {
        $error['reg_password'] = "Password not empty";
    }

    if (empty($reg_con_password)) {
        $error['reg_con_password'] = "Password not empty";
    }

    //password typo check
    if (!empty($reg_password) and $reg_con_password) {

        if ($reg_password != $reg_con_password) {
            $error['reg_con_password'] = "Password not match";
        }
    }
}
```

Figure 4. 4 Basic Validation Code Segment

- Advance Validation of Customer Registration Form

Figure 4.5 Advanced validation of customer registration form code segment is used to validate the user input and prevent entering unwanted data into the database.

```
// Advance validation
if (!preg_match("/^[a-zA-Z ]*$/", $reg_first_name)) {
    $error['reg_first_name'] = "Only Letters allowed for First Name";
}

if (!preg_match("/^[a-zA-Z ]*$/", $reg_last_name)) {
    $error['reg_last_name'] = "Only Letters allowed for Last Name";
}

if (!empty($reg_email) && @$reg_previous_email != $reg_email) {

    if (!filter_var($reg_email, FILTER_VALIDATE_EMAIL)) {

        $error['reg_email'] = "Email Address is not valid";
    } else {

        $sql_e = "SELECT * FROM users WHERE email = '$reg_email'";
        $result_e = $db->query($sql_e);
        if ($result_e->num_rows > 0) {
            $error['reg_email'] = "Email Already Exists";
        }
    }
}

if (!empty($reg_username)) {

    $sql = "SELECT * FROM users WHERE user_name = '$reg_username'";

    $result = $db->query($sql);

    if ($result->num_rows > 0) {
        $error['reg_username'] = "<b> $reg_username </b> User Already Exists";
    }
}

if (!empty($reg_password)) {
    if (strlen($reg_password) < 8) {
        $error['reg_password'] = "Password Should be at least 8 characters";
    }
}
```

*Figure 4. 5 Advanced Validation Code Segment*

#### 4.5.1 – Customer Checkout Code Segments

When customers add an item to their cart, the following customer checkout code segment is utilized for the checkout process, as shown in figure 4.6.

```

if (empty($error)) {

    $discount = $_SESSION['grand_total_sale'];
    $user_id = $_SESSION['user_id'];
    $order_total = $_SESSION['grand_total'];
    $time = date("H:i:s");
    $grand_total = $_SESSION['order_grand_total'];

    // insert order
    $sql_order = "INSERT INTO `orders` (`order_id`, `order_number`, `order_total`, `total_discount`,
        `delivery_charge`, `order_date`, `order_time`, `user_id`, `payment_id`, `grand_total`)
        VALUES (NULL, '$order_number', '$order_total', '$discount', '$d_province', '$date',
        '$time', '$user_id', '$payment_method', '$grand_total');";
    // run database query
    $query = $db->query($sql_order);

    // capture last insert ID
    $order_id = $db->insert_id;

    $_SESSION['order_id'] = $order_id;

    // change order number
    $order_number = $order_number . sprintf('%04d', $order_id);
    // order number update
    $sql = "UPDATE orders SET order_number = '$order_number' WHERE order_id = '$order_id'";
    // run database query
    $query = $db->query($sql);
}

```

```

// insert billing
$sql_billing = "INSERT INTO `billing_details` (`id`, `first_name`, `last_name`, `phone`, `email`,
    `address_line_1`, `address_line_2`, `provinces`, `city`, `zip`, `order_id`)
    VALUES (NULL, '$frist_name', '$last_name', '$phone', '$email', '$address_line_1',
    '$address_line_2', '$province', '$city', '$zip', '$order_id');";
// run database query
$query = $db->query($sql_billing);

// insert delivery
$sql_delivery = "INSERT INTO `delivery_details` (`id`, `frist_name`, `last_name`, `phone`, `email`,
    `address_line_1`, `address_line_2`, `city`, `province_id`, `zip`, `order_id`)
    VALUES (NULL, '$d_frist_name', '$d_last_name', '$d_phone', '$d_email', '$d_address_line_1',
    '$d_address_line_2', '$d_city', '$d_province', '$d_zip', '$order_id');";
// run database query
$query = $db->query($sql_delivery);

// session cart extract
foreach ($_SESSION['cart'] as $product) {

    $item_id = $product['item_id'];
    $item_price = $product['item_price'];
    $item_sale_price = $product['sales_price'];
    $grn_price = $product['grn_price'];
    $item_qty = $product['item_qty'];

    $sql = "INSERT INTO `orders_items` (`orders_items_id`, `order_id`, `item_id`, `item_qty`,
        `grn_price`, `unit_price`, `sale_price`) VALUES (NULL, '$order_id', '$item_id',
        '$item_qty', '$grn_price', '$item_price', '$item_sale_price');";
    // run database query
    $db->query($sql);
}
}

```

Figure 4. 6 Customer Checkout Code Segment

## 4.6 – Reused Existing Codes

When implementing the system, several of the well-tested re-usable components were added in order to maximize efficiency and provide more interest to the system.

- **Reusable CSS Files**

Figure 4.7 shows how the complete system's front-end styling is done using CSS codes.

```
<!-- Font Awesome -->
<link rel="stylesheet" href=<?php echo SITE_URL; ?>plugins/fontawesome-free/css/all.min.css">
<!-- Ionicons -->
<link rel="stylesheet" href="https://code.ionicframework.com/ionicons/2.0.1/css/ionicons.min.css">
<!-- Tempusdominus Bootstrap 4 -->
<link rel="stylesheet" href=<?php echo SITE_URL; ?>plugins/tempusdominus-bootstrap-4/css/tempusdominus-bootstrap-4.min.css">
<!-- iCheck -->
<link rel="stylesheet" href=<?php echo SITE_URL; ?>plugins/checkbox/icheck/icheck-bootstrap.min.css">
<!-- JQVMap -->
<link rel="stylesheet" href=<?php echo SITE_URL; ?>plugins/jqvmap/jqvmap.min.css">
<!-- DataTables -->
<link rel="stylesheet" href=<?php echo SITE_URL; ?>plugins/datatables-bs4/css/dataTables.bootstrap4.min.css">
<link rel="stylesheet" href=<?php echo SITE_URL; ?>plugins/datatables-responsive/css/responsive.bootstrap4.min.css">
<link rel="stylesheet" href=<?php echo SITE_URL; ?>plugins/datatables-buttons/css/buttons.bootstrap4.min.css">
<!-- Theme style -->
<link rel="stylesheet" href=<?php echo SITE_URL; ?>dist/css/adminlte.min.css">
<!-- overlayScrollbars -->
<link rel="stylesheet" href=<?php echo SITE_URL; ?>plugins/overlayScrollbars/css/OverlayScrollbars.min.css">
<!-- Daterangepicker -->
<link rel="stylesheet" href=<?php echo SITE_URL; ?>plugins/daterangepicker/daterangepicker.css">
<!-- summernote -->
<link rel="stylesheet" href=<?php echo SITE_URL; ?>plugins/summernote/summernote-bs4.min.css">
```

Figure 4. 7 Reusable CSS Files

- **Reusable JavaScript Files**

The reusable javascript code used to make the system's front-end dynamic is shown in Figure 4.8.

```
<!-- jQuery -->
<script src=<?php echo SITE_URL; ?>plugins/jquery/jquery.min.js"></script>
<!-- jQuery UI 1.11.4 -->
<script src=<?php echo SITE_URL; ?>plugins/jquery-ui/jquery-ui.min.js"></script>
<!-- Resolve conflict in jQuery UI tooltip with Bootstrap tooltip -->
<script>
|   $widget.bridge('uibutton', $ui.button)
</script>
<!-- Bootstrap 4 -->
<script src=<?php echo SITE_URL; ?>plugins/bootstrap/js/bootstrap.bundle.min.js"></script>
<!-- DataTables & Plugins -->
<script src=<?php echo SITE_URL; ?>plugins/datatables/jquery.dataTables.min.js"></script>
<script src=<?php echo SITE_URL; ?>plugins/datatables-bs4/js/dataTables.bootstrap4.min.js"></script>
<script src=<?php echo SITE_URL; ?>plugins/datatables-responsive/js/dataTables.responsive.min.js"></script>
<script src=<?php echo SITE_URL; ?>plugins/datatables-responsive/js/responsive.bootstrap4.min.js"></script>
<script src=<?php echo SITE_URL; ?>plugins/datatables-buttons/js/dataTables.buttons.min.js"></script>
<script src=<?php echo SITE_URL; ?>plugins/datatables-buttons/js/buttons.bootstrap4.min.js"></script>
<script src=<?php echo SITE_URL; ?>plugins/datatables-buttons/js/buttons.html5.min.js"></script>
<script src=<?php echo SITE_URL; ?>plugins/datatables-buttons/js/buttons.print.min.js"></script>
<script src=<?php echo SITE_URL; ?>plugins/datatables-buttons/js/buttons.colVis.min.js"></script>
<!-- ChartJS -->
<script src=<?php echo SITE_URL; ?>plugins/chart.js/Chart.min.js"></script>
```

Figure 4. 8 Reusable Javascript files

- **Reusable Image Upload Function**

Figure 4.9 shows how to upload images to the entire system using the reusable image upload function.

```
// Image Upload Function-----
function image_upload($image_upload = null, $target_dri = null, $previous_image = null)
{

    if (!empty($_FILES[$image_upload]['name'])) {
        $target_file = $target_dri . basename($_FILES[$image_upload]["name"]);
        $upload_ok = 1;
        $image_file_type = strtolower(pathinfo($target_file, PATHINFO_EXTENSION));
        $check = getimagesize($_FILES[$image_upload]['tmp_name']);
        if ($check !== false) {
            //Multi-purpose Internet Mail Extensions
            $upload_ok = 1;
        } else {
            $error[$image_upload] = "File is not an image.";
            $upload_ok = 0;
        }

        if (file_exists($target_file)) {
            // $error[$image_upload] = "Sorry, file already exists.";
            // $upload_ok = 0;
            unlink($target_file);
            $upload_ok = 1;
        }

        if ($_FILES[$image_upload]["size"] > 5000000000) {
            $error[$image_upload] = "Sorry, your file is too large.";
            $upload_ok = 0;
        }
    }

    if ($image_file_type != "jpg" && $image_file_type != "png" && $image_file_type != "jpeg" && $image_file_type != "gif") {
        $error[$image_upload] = "Sorry, only JPG, JPEG, PNG & GIF files are allowed.";
        $upload_ok = 0;
    }

    if ($upload_ok == 0) {
        echo "Sorry, your file was not uploaded.";
        // if everything is ok, try to upload file
    } else {
        if (move_uploaded_file($_FILES[$image_upload]["tmp_name"], $target_file)) {
            $error['photo'] = htmlspecialchars(basename($_FILES[$image_upload]["name"]));
        } else {
            $error[$image_upload] = "Sorry, there was an error uploading your file.";
        }
    } else {
        $error['photo'] = $previous_image;
    }

    return @$error;
}
```

*Figure 4.9 Reusable Image Upload Function*

- **Reusable Data Clean Function**

The reusable data clean function in figure 4.10 is used to clean user inputs to prevent input the unwanted characters.

```
function data_clean($data = null)
{
    trim($data);
    stripslashes($data);
    htmlspecialchars($data);

    return $data;
}
```

*Figure 4. 10 Reusable Data Clean Function*

- **Reusable Main Menus Code in the Dashboard**

In figure 4.11 code segment is used to dynamic the main menu for the multi-level user inputs.

```
<?php

$sql = "SELECT m.module_id, m.description, m.path, m.view, m.icon, m.status
FROM users_modules um
INNER JOIN modules m ON m.module_id = um.module_id
WHERE length(m.module_id) = '2' AND um.user_id ='" . $_SESSION['user_id'] . "' AND m.status = '1';

// database connection call
$db = db_con();

// assign the query
$result = $db->query($sql);

?>

<?php

// check the result grater than 1
if ($result->num_rows > 0) {

    // assign the result to row variable
    while ($row = $result->fetch_assoc()) {

?>
        <li class="nav-item">
            <a href="#" class="nav-link">
                <i class="nav-icon <?php echo $row['icon'] ?>"></i>
                <p>
                    <?php
                        // display module name
                        echo $row['description'];
                    ?>
                    <i class="right fas fa-angle-left"></i>
                </p>
            </a>
        </li>
    }
}
?>
```

*Figure 4. 11 Reusable Main Menu Code Segment*

- **Reusable Sub Menus Code in the Dashboard**

In figure 4.12 code segment is used to dynamic for the sub menus.

```
<ul class="nav nav-treeview">
<?php
$SQL_sub = "SELECT m.module_id, m.description, m.path, m.view, m.icon, m.status
FROM users_modules um
INNER JOIN modules m ON m.module_id = um.module_id
WHERE length(m.module_id) = '4' AND um.user_id ='" . $_SESSION['user_id'] .
"" AND substr(m.module_id, 1,2) = '" . $row['module_id'] . "' AND m.status = '1'";
// assign the query
$result_sub = $db->query($SQL_sub);
?>
<?php
if ($result_sub->num_rows > 0) {
    while ($row_sub = $result_sub->fetch_assoc()) {
        // create file path for sub menu
        $file = $row_sub['path'] . "/" . $row_sub['view'] . ".php";
?>
        <li class="nav-item">
            <a href=<?php echo SITE_URL; ?><?php echo $file; ?>" class="nav-link">
                <i class="far fa-circle nav-icon"></i>
                <p><?php echo $row_sub['description'] ?></p>
            </a>
        </li>
        <?php
    }
?>
</ul>
```

Figure 4. 12 Reusable Sub Menu Code Segments

# Chapter 5 – Evaluation

## 5.1 – Introduction

The Evaluation chapter includes descriptions of the test strategy, test outcomes, and user evaluation. To check and verify the system's performance, a wide range of data should be confirmed. During this step, the system's feature is assessed. The program is put through its paces in accordance with the test strategy.

## 5.2 - Description of testing approach

“A test approach is the test strategy implementation of a project, defines how testing would be carried out. Test approach has two techniques:

**Proactive** - An approach in which the test design process is initiated as early as possible in order to find and fix the defects before the build is created.

**Reactive** - An approach in which the testing is not started until after design and coding are completed.

There are many strategies that a project can adopt depending on the context and some of them are:

- Dynamic and heuristic approaches
- Consultative approaches
- Model-based approach that uses statistical information about failure rates.
- Approaches based on risk-based testing where the entire development takes place based on the risk
- Methodical approach, which is based on failures.
- Standard-compliant approach specified by industry-specific standards.” [6]

## 5.3 - Test strategy

“A Software Test Plan is a document describing the testing scope and activities. It is the basis for formally testing any software/product in a project. It describes the scope, approach, resources and schedule of intended test activities. It identifies amongst others test items, the features to be tested, the testing tasks, who will do each task, degree of tester independence, the test environment, the test design techniques and entry and exit

criteria to be used, and the rationale for their choice, and any risks requiring contingency planning. It is a record of the test planning process.” [7]

### 5.3.1 - Verification and Validation

“Verification and Validation is the process of investigating that a software system satisfies specifications and standards and it fulfills the required purpose. Barry Boehm described verification and validation as the following:

**Verification:** Are we building the product right?

**Validation:** Are we building the right product?

Verification is the process of checking that a software achieves its goal without any bugs. It is the process to ensure whether the product that is developed is right or not. It verifies whether the developed product fulfills the requirements that we have. Validation is the process of checking whether the software product is up to the mark or in other words product has high level requirements. It is the process of checking the validation of product i.e. it checks what we are developing is the right product. it is validation of actual and expected product.” [8]

### 5.3.2 - Testing Process

#### 1. Unit testing

Unit testing, a type of software testing, involves testing individual programme units or components. Verifying that each piece of software operates as intended is the goal. The smallest bit of testable software is called a unit.

#### 2. Integration Testing

Following unit testing, integration testing is carried out, in which many units are combined and tested as a whole. The purpose of this testing level is to highlight issues with integrated component interfaces and interactions. Depending on the definition of a unit, either white-box or black-box testing can be used.

#### 3. System Testing

System testing is a sort of testing used to determine whether an integrated system as a whole satisfies the requirements. The black box testing approach is used to test the system's functionality from end to end during system testing.

#### 4. Acceptance Testing

Before the system was put into service, this was the last phase of testing. The system was tested using data supplied by the system procurer rather than fictitious test data. This is a beta test of the product carried out by actual customers.

## 5. Regression Testing

Regression test is a testing of software testing used to make sure that present functionalities haven't been broken by a recent programme or code revision. This testing's main objective is to make that the system's current functional and non-functional features haven't been negatively impacted by the most recent updates.

### 5.4 - Proof of testing of work

The most significant component of this phase is creating test scenarios. Test cases that are well-planned should be able to validate system module functionality. System test cases must validate every type of system need. The essential components of the test cases built for this system are listed below.

- **Test case for Customer Registration**

The following major test cases ran in the system (Table 5.1)

Test No	Test Case	Expected Result	Status
1	Submit the form without filling in the mandatory fields.	Prevent submitting and show error messages	Pass
2	Use already exists username and email address	show error message	Pass
3	Invalid email insert	show error message	Pass
4	Incorrect Password length insert	show error message	Pass
6	Invalid phone number insert	show error message	Pass
7	Enter the numbers in the text fields	show error message	Pass
8	Invalid zip inserts	show error message	Pass
9	Submit the form with the correct data	Redirect to the login page	Pass

10	Without login URL browsing to the customer dashboard	Redirect to login page	Pass
----	--	------------------------	------

Table 5. 1: Test Case for Customer Registration

- **Test case for Customer Login**

The system underwent the key test cases listed below. (Table 5.2)

Test No	Test Case	Expected Result	Status
1	Enter a valid username and password	Successful login to the system	Pass
2	Enter username only	show error message	Pass
3	Enter only password	show error message	Pass
4	Enter invalid username and password	show error message	Pass

Table 5. 2: Test Results for Customer login

- **Test case for Checkout Process**

Checkout process test cases (Table 5.3).

Test No	Test Case	Expected Result	Status
1	Submit the form without filling in the mandatory fields.	Prevent submitting and show error messages	Pass
2	Enter the numbers in the text fields	show error message	Pass
3	Invalid email insert	show error message	Pass
4	Invalid phone number insert	show error message	Pass
5	Invalid zip inserts	show error message	Pass
6	Without check the delivery check box	Show error message of delivery form	Pass
7	Without selecting the payment method	Show error message	Pass
8	Change the delivery value related to the province	Show delivery charges and calculate to total	Pass

9	Submit the form with the correct data	direct to the invoice page	Pass
10	Without login URL browsing to the checkout and invoice	Redirect to the cart page	Pass

*Table 5. 3: Test Result for Checkout Process*

## 5.5 - User Acceptance Testing

“User Acceptance Testing (UAT), also known as beta or end-user testing, is defined as testing the software by the user or client to determine whether it can be accepted or not. This is the final testing performed once the functional, system, and regression testing are completed.

The main purpose of this testing is to validate the software against the business requirements. This validation is carried out by the end-users who are familiar with the business requirements.

UAT, alpha, and beta testing are different types of acceptance testing.

As the user acceptance test is the last testing that is carried out before the software goes live, obviously this is the last chance for the customer to test the software and measure if it is fit for the purpose.” [9]

## 5.6 - User Evaluation

An evaluation that includes people for whom the system has suggested users is referred to as a user evaluation because it is based on user input. Examples of user evaluation techniques include probable methods, observational methods, questionnaires, interviews, and physiological monitoring techniques. Users of various types and access levels were chosen to be part of the sample, and user evaluation surveys were given out to collect feedback.

Figure 5.1 depicts an example evaluation questionnaire for testing user satisfaction, figure 5.2 depicts a user acceptance test evaluation sample, and figure 5.3 depicts the system's user feedback results analysis diagram.

**User Feedback Form – Computer Hardware Purchasing and Troubleshooting**  
**Assistant Management System for U-Star Digital**

Role of User: .....

Test Date: .....

Mark (✓) inside the box. Please provide a more comprehensive response with your honest option.

No	Description	Exceptional	Satisfied	Bad
1	User interface design			
2	system's response time			
3	Form validation			
4	Understandability of error messages			
5	Interaction with navigation links and buttons			
6	system's responsiveness			
7	Report generating capability			
8	Recognizing the system's design			
9	Efficiency of the functionalities			
10	Overall impressions of the system			

Thank You

*Figure 5. 1: User Feedback Form*

**User Feedback Form – Computer Hardware Purchasing and Troubleshooting**  
**Assistant Management System for U-star Digital**

Role of User: ...*Customer*.....

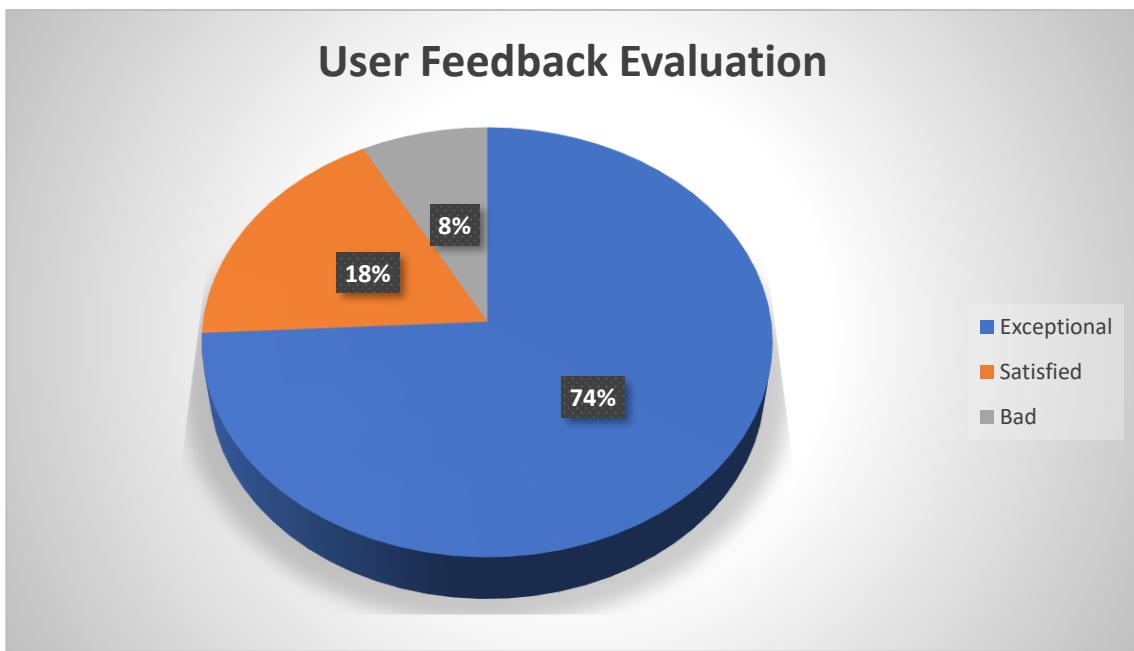
Test Date: ...*26/03/2022*.....

Mark (✓) inside the box. Please provide a more comprehensive response with your honest option.

No	Description	Exceptional	Satisfied	Bad
1	User interface design	✓		
2	system's response time	✓		
3	Form validation		✓	
4	Understandability of error messages		✓	
5	Interaction with navigation links and buttons		✓	
6	system's responsiveness	✓		
7	Report generating capability		✓	
8	Recognizing the system's design		✓	
9	Efficiency of the functionalities		✓	
10	Overall impressions of the system	✓		

Thank You

*Figure 5. 2: User Acceptance Test Evaluation Sample*



*Figure 5. 3: Summary of User Feedback Evaluation*

# **Chapter 6 – Conclusion**

## **6.1- Introduction**

This chapter highlights the lessons learned while working on the U-Star Digital project to design a Computer Hardware Purchasing and Troubleshooting Assistant Management System. It also makes suggestions for future improvements to the system's performance and usability.

## **6.2 - Critical evaluation of the project**

U-Star Digital currently does not have a system in place to link its shopping operations. To capture client information, their employees use spreadsheet programs and manual (paper papers). There is less integrity, the workload for the staff and their clients is high, inaccurate data levels occur, and all procedures are highly slow, time-consuming, and difficult to manage due to the use of paper documents and spreadsheets for data recording and data analysis.

The proposed solution was created to address those flaws while also improving their entire business process. The system was created with the most up-to-date development standards and technology in mind, as well as the requirement to expand and improve the business and the system in the future.

The client's criteria were helping the project become doable. Since the documentation's stated scope was functional and some client criteria weren't quite met, Examples include email notifications for the order process, user dashboard notifications for delivery updates, and so on. Test results indicate the system's user behaviour. The outcomes can help establish whether the system is suitable for real-time implementation.

Finally, the client provided feedback that was deemed acceptable for phase one and required further explanation of the features specified in the system as well as the use of a validation procedure to address some issues.

## **6.3 - Lesson learned**

This project was the first time I had worked on a real-world project. As a student who took on this project, this software development has provided me with a wealth of programming expertise. During the system's implementation, the programming skills learned in the previous semesters were put to use in a real-world setting.

As a student may have many queries about the front end, back end, and databases when working on this kind of project. In the pressing time period, searching the internet for the ideal solution is considerably more challenging.

Front end HTML and CSS work is manageable, but Javascript and jQuery have problems obtaining some data when used with ajax functions. Utilizing a variety of new PHP functions, further validation strategies were discovered in the back end to increase system security.

Working on a project allowed me to obtain and improve soft skills including communication, creative thinking, logical thinking, decision making, positivism, motivation, problem-solving, understandability, writing skills, self-learning, researching skills, self-confidence, and so on.

## 6.4 - Future work

The intended software is now accessible and completely functional, allowing the customer to manage and afterwards carry out the entire operation in a far better, more precise, and error-free manner. The following is the project's future scope.

- **Website Mobile Responsiveness**

The system's current state is only visually appealing for desktop computers. We intend to change the website's responsiveness so that users of mobile devices and tablets can access it more comfortably.

- **Creating a Mobile Application**

Users will be able to access and use the system more readily on their mobile devices thanks to the development of an interactive mobile application for this system.

- **Integrating the Internet payment gateway**

Customers are allowed to make payments more readily if they have access to an internet payment gateway.

- **AI Base Troubleshooting Assistant**

Implementing an artificial intelligence-based troubleshooting assistant that crawls the database and answers to provide the best response for the specified inquiry.

- **Computer Hardware Assistant**

Implementing this module assists customers who are unfamiliar with computer hardware. It takes all the details of individual items and matches them with the next most appropriate piece, and completes the assembly of an entire computer. After completing, customers have an option to check out the selected parts and make a payment to complete the assistant process.

# References

- [1] Micro Center, “About Micro Center”, <https://www.microcenter.com/>, [Online]. Available: [https://www.microcenter.com/site/content/about\\_microcenter.aspx](https://www.microcenter.com/site/content/about_microcenter.aspx) [Accessed: 18.12.2021].
- [2] newegg, “About Newegg corporate site”, <https://www.newegg.com/>, [Online]. Available: <https://www.newegg.com/corporate/about> [Accessed: 18.12.2021].
- [3] T. Alspaugh, “Software Process Models”, [www.thomasalspaugh.org](http://www.thomasalspaugh.org), [Online]. Available: <https://www.thomasalspaugh.org/pub/fnd/softwareProcess.html#scrum> [Accessed: 22.01.2022].
- [4] Agile Alliance, “What is extreme programming”, <https://www.agilealliance.org/> [Online]. Available: <https://www.agilealliance.org/glossary/xp> [Accessed: 22.01.2022].
- [5] Techterms.com, “RUP (Rational Unified Process) Definition”, <https://techterms.com/> [Online]. Available: <https://techterms.com/definition/rup> [Accessed: 22.01.2022].
- [6] “Test Approach Description”, <https://www.tutorialspoint.com/> [Online]. Available: [https://www.tutorialspoint.com/software\\_testing\\_dictionary/test\\_approach.htm](https://www.tutorialspoint.com/software_testing_dictionary/test_approach.htm) [Accessed: 18.02.2022].
- [7] Test plan, in software testing fundamentals, “Test Plan”, [Online]. Available: <https://softwaretestingfundamentals.com/test-plan> [Accessed: 19.02.2022].
- [8] “Software Engineering | Verification and Validation”, [Online]. Available: <https://www.geeksforgeeks.org/software-engineering-verification-and-validation/> [Accessed: 19.02.2022].
- [9] A Complete Guide, “User Acceptance Testing”, [www.softwaretestinghelp.com](http://www.softwaretestinghelp.com) [Online]. Available: <https://www.softwaretestinghelp.com/what-is-user-acceptance-testing-uat/> [Accessed: 20.02.2022].

# Appendix A – System Manual

The correct way to install, set up, configure, and operate the system for U-Star Digital is described in detail in this manual.

## Step 01

The following is a list of the hardware and software requirements as shown in tables A.1 and A.2.

### Hardware Requirements

Hardware	Requirements
Processor	Intel(R) Core(TM) i3-5005U CPU @ 2.00GHz
Memory	2GB or Above
Storage	25GB
Display	1366 x 768 Resolution Monitor
Internet	4 Mbps internet connection

*Table A. 1: Hardware Requirements*

### Software Requirements

Hardware	Requirements
Operating System	Microsoft Windows 7 or Above Operating System
Web Server	XAMPP release 7.3.33, 7.4.26
Web Browser	Chromium-based web browsers / Firefox
Code Editor	Microsoft Visual Studio Code / Notepad++

*Table A. 2: Software Requirements*

## Step 02

1. Downloading XAMPP Server
  - a. WAMP Server can be downloaded and installed from the following URL.  
<https://www.apachefriends.org/download.html> (Figure A.1)
  - b. Download the appropriate WAMP Server version.
  - c. When installing WAMP Server on your computer, follow the instructions on the WAMP Server installation web page.

XAMPP is an easy to install Apache distribution containing MariaDB, PHP, and Perl. Just download and start the installer. It's that easy.

Version	Checksum	Size
7.4.29 / PHP 7.4.29	<a href="#">What's Included?</a> <a href="#">md5</a> <a href="#">sha1</a>	<a href="#">Download (64 bit)</a> 159 Mb
8.0.18 / PHP 8.0.18	<a href="#">What's Included?</a> <a href="#">md5</a> <a href="#">sha1</a>	<a href="#">Download (64 bit)</a> 161 Mb
8.1.5 / PHP 8.1.5	<a href="#">What's Included?</a> <a href="#">md5</a> <a href="#">sha1</a>	<a href="#">Download (64 bit)</a> 164 Mb

Requirements Add-ons More Downloads »  
Windows XP or 2003 are not supported. You can download a compatible version of XAMPP for these platforms here.

Figure A. 1: XAMPP Server Download Page

## 2. Installing XAMPP Server

- Run the XAMPP Server execution file that you downloaded (.exe).
- Deactivate UAC
- Choose a location to install the WAMP Server and proceed to install.
- Double-click on the XAMPP icon to launch the server. The server will then be started by clicking the start button.
- Start “Apache” and MySQL modules by clicking on the “Start” button.

## Step 03

### 1. Create the Database

- Launch a web browser and enter <http://localhost/phpmyadmin>.
- If you want to set a password, log in to the database and enter a username "root" and password.
- Make a database with the specified name “db\_ustar” (Figure A.2).

Database	Collation	Action
bit	utf8mb4_general_ci	<a href="#">Check privileges</a>
db2	utf8mb4_general_ci	<a href="#">Check privileges</a>
db_ustar	utf8mb4_general_ci	<a href="#">Check privileges</a>
information_schema	utf8_general_ci	<a href="#">Check privileges</a>
mysql	utf8mb4_general_ci	<a href="#">Check privileges</a>

Figure A. 2: Create Database

2. Database File Import

- a. Choose a database that was generated and click the "import" tab
- b. Navigate to the database folder on the CD and find the "db\_ustar.sql" file.
- c. Click the Go button.

3. Store the System File

- a. Go to the directory where the XAMPP server was set up.
- b. Find the "htdocs" directory.
- c. Copy the "u\_star\_system.zip" file from the given CD.
- d. Copy the file and paste it into the htdocs folder to unzip it.

4. Launching the System

- a. Type <http://localhost/ustar/> to launch the system in your browser
- b. Type <http://localhost/ustar/system/> to login to the system in your browser.

# Appendix B – Design Documentation

The following diagrams represent a few other significant system functionalities, besides the charts for data modelling in Chapter 3 of the dissertation.

## Use Case Diagrams

A use case diagram shows how a system's components relate to one another visually. A system analysis technique for locating, delineating, and arranging system needs is the use case. Based on their respective roles, the system's actors are organised into groups.

- **Use – Case Diagram for Customer Registration**

Both Figure B.1 and Table B.1 provide the use case narrative and use case diagram for customer registration, respectively.

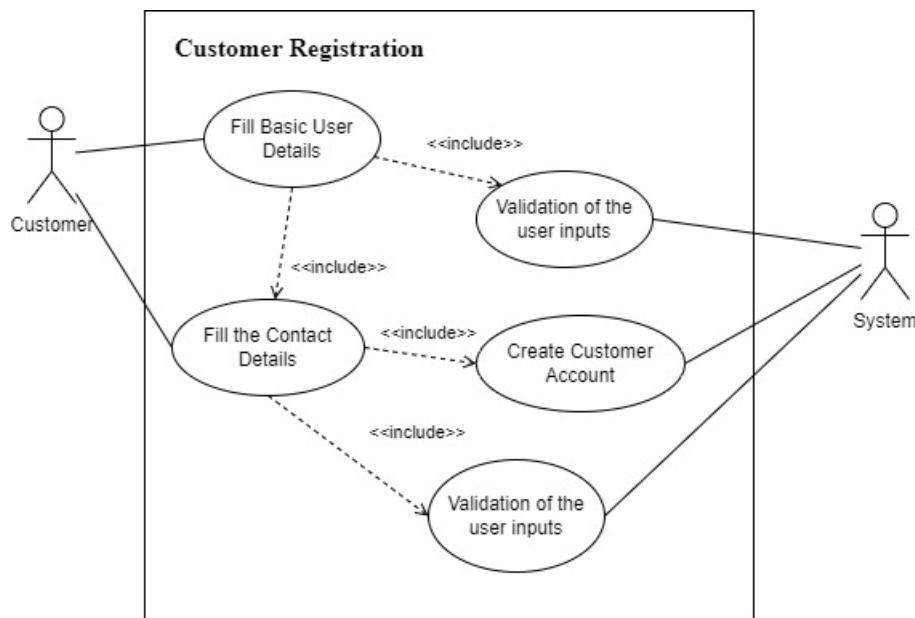


Figure B. 1: Use Case Diagram for Customer Registration

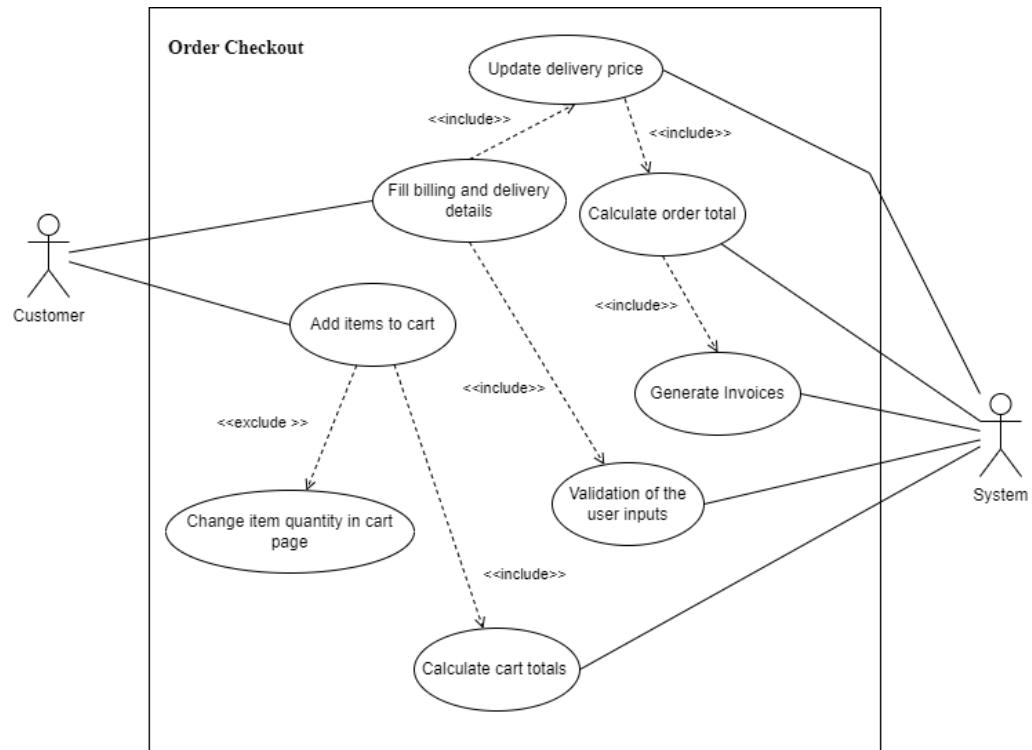
<b>Use-case Number</b>	UC-02	
<b>Use-Case Name</b>	Customer Registration	
<b>Priority</b>	High	
<b>Actor</b>	Customer	
<b>Description</b>	The procedure for enrolling users with the system is described in this use case.	
<b>Precondition</b>	None	
<b>Post-condition</b>	The actor is directed to the customer login page to log in to the system if the use case was successful.	
<b>Basic course of Action</b>	<b>User Action</b> 1. The Actor is registering as a customer on the My Account page.  3. The actor clicks the Register button after completing the form.  6. The Actor fills out the form and clicks the Done button.	<b>System Response</b> 2. The form to enter user information is promoted by the system.  4. The system checks to make sure all the fields have been correctly completed and are legitimate.  5. All fields are valid system navigates to the complete profile page.  6. The system verifies that all the filled have been filled out and are valid.  7. System navigate to the My Account page to log in to the system.  8. Use case Exit
<b>Alternate course of Action</b>	4.1 The system notifies the actor and then proceeds to step 3 of the basic course of action to request new information if all	

	<p>fields are not completed and have previously been claimed by another client.</p> <p>5.1 The system notifies the actor and then proceeds to step 3 of the fundamental course of action to request new information if all fields are not completed and have previously been claimed by another client.</p>
--	---

*Table B. 1: Use Case Narrative for Customer Registration*

- **Use – Case Diagram for Order Checkout**

Use case diagram for order checkout is given in Figure B.2 and use case narrative for order checkout is given in Table B.2.



*Figure B. 2: Use Case Diagram for Order Checkout*

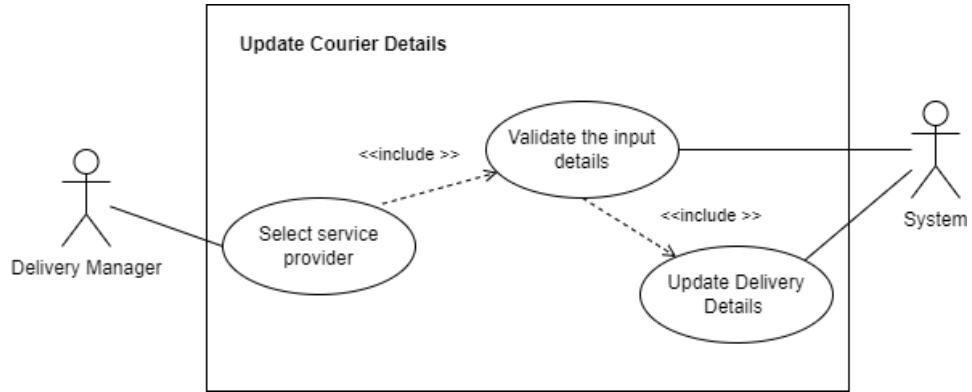
<b>Use-case Number</b>	UC-03
<b>Use-Case Name</b>	Order Checkout
<b>Priority</b>	High
<b>Actor</b>	Customer
<b>Description</b>	This use case explains how to check out the things you've chosen from your shopping cart.

<b>Precondition</b>	Need to add items to the cart	
<b>Post-condition</b>	When the process is completed successfully, the actor is directed to the invoicing page.	
<b>Basic course of Action</b>	<b>User Action</b>	<b>System Response</b>
	1. The Actor adds items to the cart. 2. Actor clicks cart icon in navigation. 5. The Actor clicks the checkout order button. 7. Actor fill the details of delivery and billing details forms and check the payment method and click the place your order button.	3. System display all items added to the cart by the actor. 4. System calculates the item total price without delivery price. 6. System prompt billing details, delivery details forms, and payment methods. 8. System validates the actors input and redirects to the order invoice page.
<b>Alternate course of Action</b>	5.1 If the Actor can change the cart item quantity, the system calculates the item total and cart total regarding to the item quantity. 5.2 If the Actor does not login to the system, the checkout page displays the login and register form 7.1 Actor needs to check the checkbox for delivery to the same billing address. 8.1 If the form is not validated properly system shows the error messages.	

Table B. 2: Use Case Narrative for Order Checkout

### • Use – Case Diagram for Update Courier Details

Use case diagram for update courier details is given in Figure B.3 and use case narrative for update courier details is given in Table B.3.



*Figure B. 3 Use Case Diagram for Update Courier Details*

Use-case Number	UC-16	
Use-Case Name	Update Courier Details	
Priority	High	
Actor	Delivery Manager	
Description	This use case describes how Shop Manager update the courier details of the individual order	
Precondition	UC-15	
Post-condition	If the use case was successful, Shop Manager can change the order status and the customer can view all the courier details.	
Basic course of Action	User Action	System Response
	1. Delivery Manager at individual order page. 3. Shop Manager Update Courier Details including Courier Company Name, Tracking Number, and Tracking Url and Click the "Update Courier Details" button.	2. The system prompts the order details including shipping details. 4. System validates the fields 5. System Display message the "Courier Details Update". 6. Use Case exists.
Alternate course of Action	4.1 System validates all fields are filled with the correct format. If not, the system displays necessary error messages.	

*Table B. 3 Use Case Narrative for Update Courier Details*

## • Use – Case Diagram for Make Payments

Use case diagram for make payments is given in Figure B.4 and use case narrative for make payments is given in Table B.4.

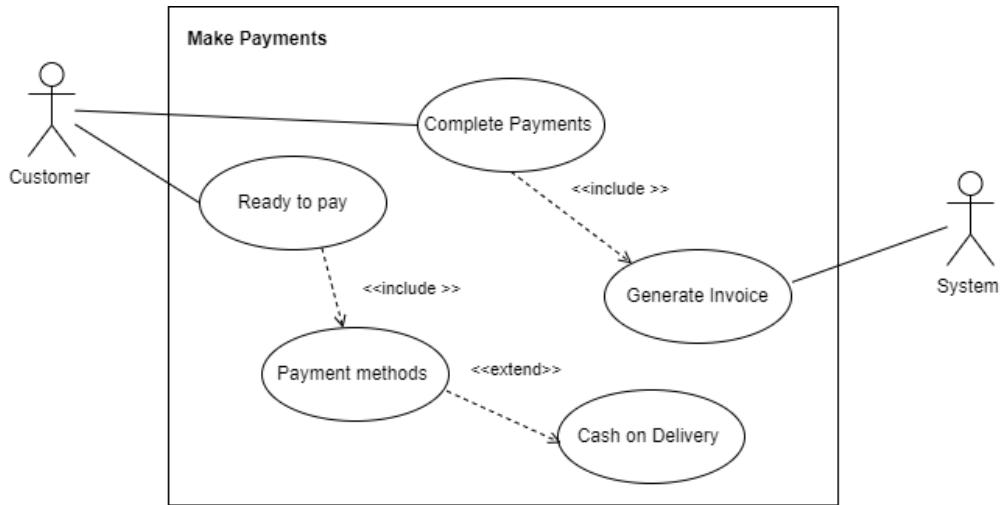


Figure B. 4 Use Case Diagram for Make Payments

Use-case Number	UC-14	
Use-Case Name	Make Payments	
Priority	High	
Actor	Customer	
Description	This use case describes payment options for the customer	
Precondition	UC-13	
Post-condition	If the use case was successful, the customer can select the payment method to place the order.	
Basic course of Action	User Action	System Response
	1. User in payment page 3. Actor selects the payment options cash on delivery and clicks “Pay Now”	2. System prompts total payment and payment options. 4. System recode the payment method and redirect to thank you page.
Alternate course of Action	4.1 System Prompt All order details and payment method on thank you page.	

Table B. 4 Use Case Narrative for Make Payments

- **Use – Case Diagram for Check Items**

Use case diagram for check items is given in Figure B.5 and use case narrative for check items is given in Table B.5.

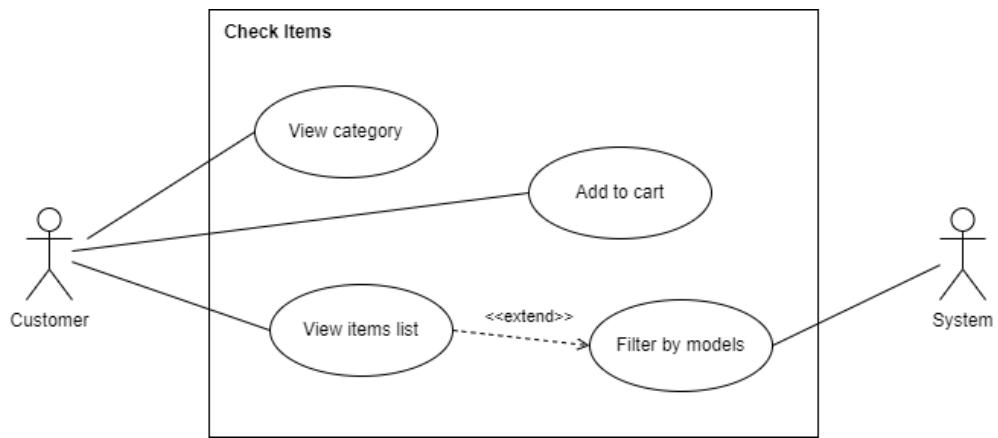


Figure B. 5 Use Case Diagram for Check Items

Use-case Number	UC-12	
Use-Case Name	Check Items	
Priority	High	
Actor	Customer	
Description	This use case describes how the customer checks the items in the system	
Precondition	UC-08	
Post-condition	If the use case was successful, the actor can check the items in the system.	
Basic course of Action	User Action	System Response
	1. User in category page. 3. User can select the category (Processors, Ram, VGA, Motherboard, Power Supply, Casing, Storage, Monitors, and Audio) and click the “View Category” Button. 5. User Click “View Item”	2. System display all available categories. 4. System navigate appropriate category page and list all available items. 6. System navigate to the individual item page and display (Item Name, Item Short Description, Price, Item Features with Details.) 7. Use Case Exist

Alternate course of Action	
----------------------------	--

Table B. 5 Use Case Narrative for Check Items

## Activity Diagrams

An key diagram in UML for describing the dynamic features of the system is the activity diagram, which is simply a flowchart to represent the flow from one activity to the next. It is possible to classify the action as a system operation.

- **Activity diagram for Customer Registration**

The activity diagram for customer login is given in Figure B.6

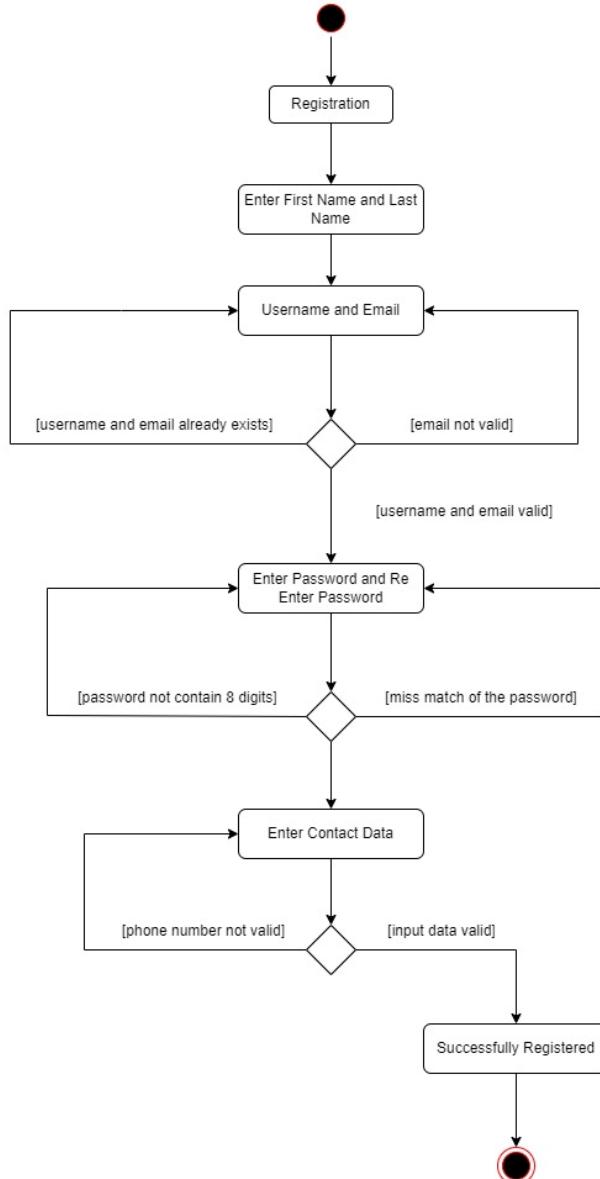


Figure B. 6: Activity Diagram for Customer Login

- **Activity diagram for Order Checkout**

The activity diagram for order checkout is given in Figure B.7

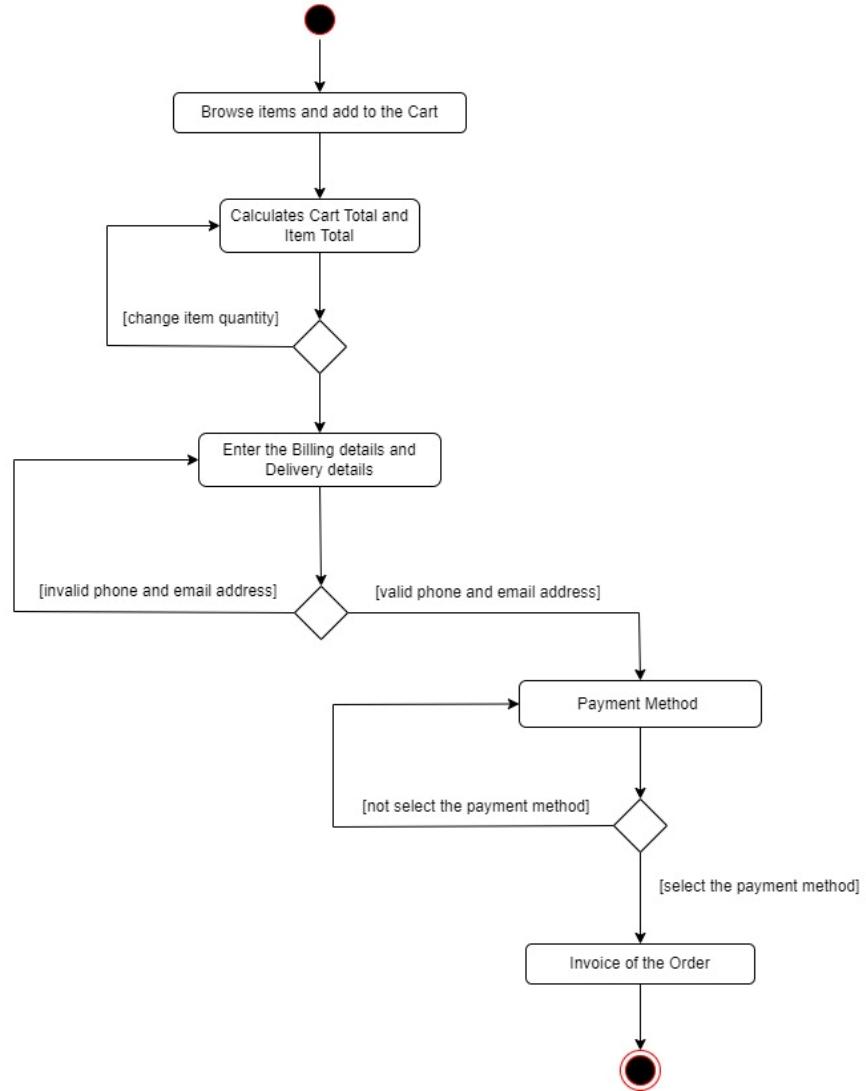


Figure B. 7: Activity Diagram for Order Checkout

- **Activity Diagram for Update Courier Details**

The activity diagram for update courier details is given in Figure B.5

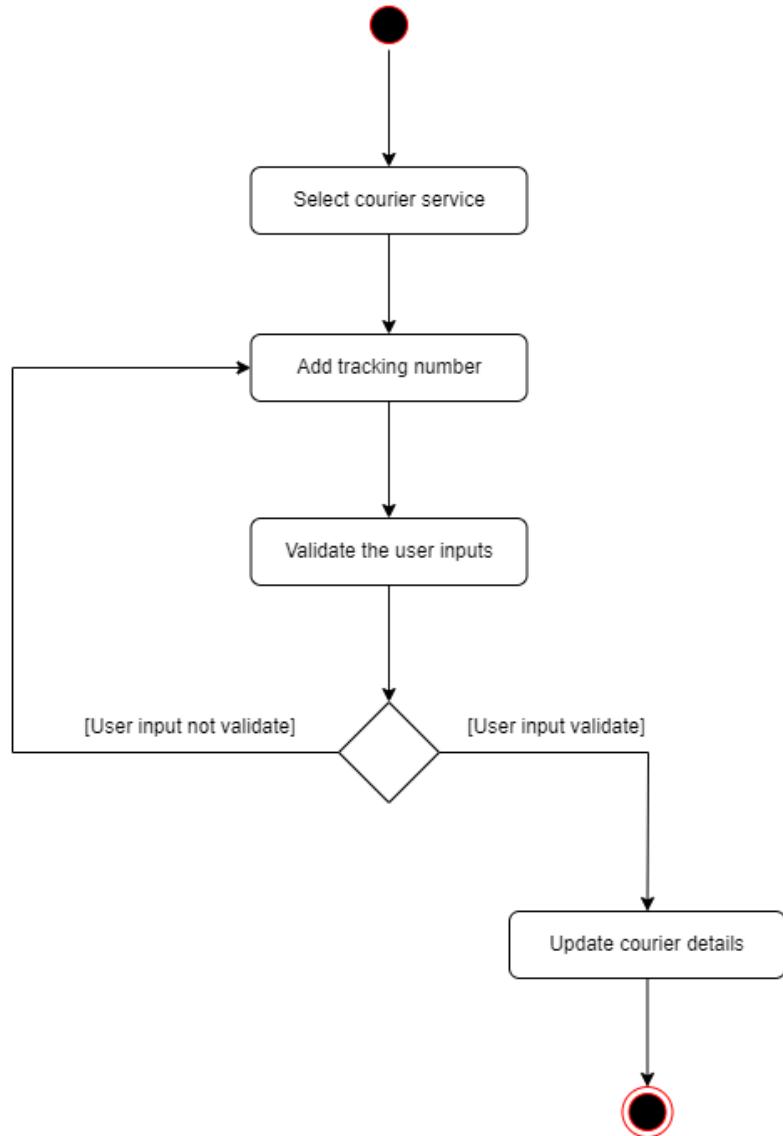


Figure B. 8 Activity Diagram for Update Courier Details

- **Activity Diagram for Make Payments**

The activity diagram for make payments is given in Figure B.9

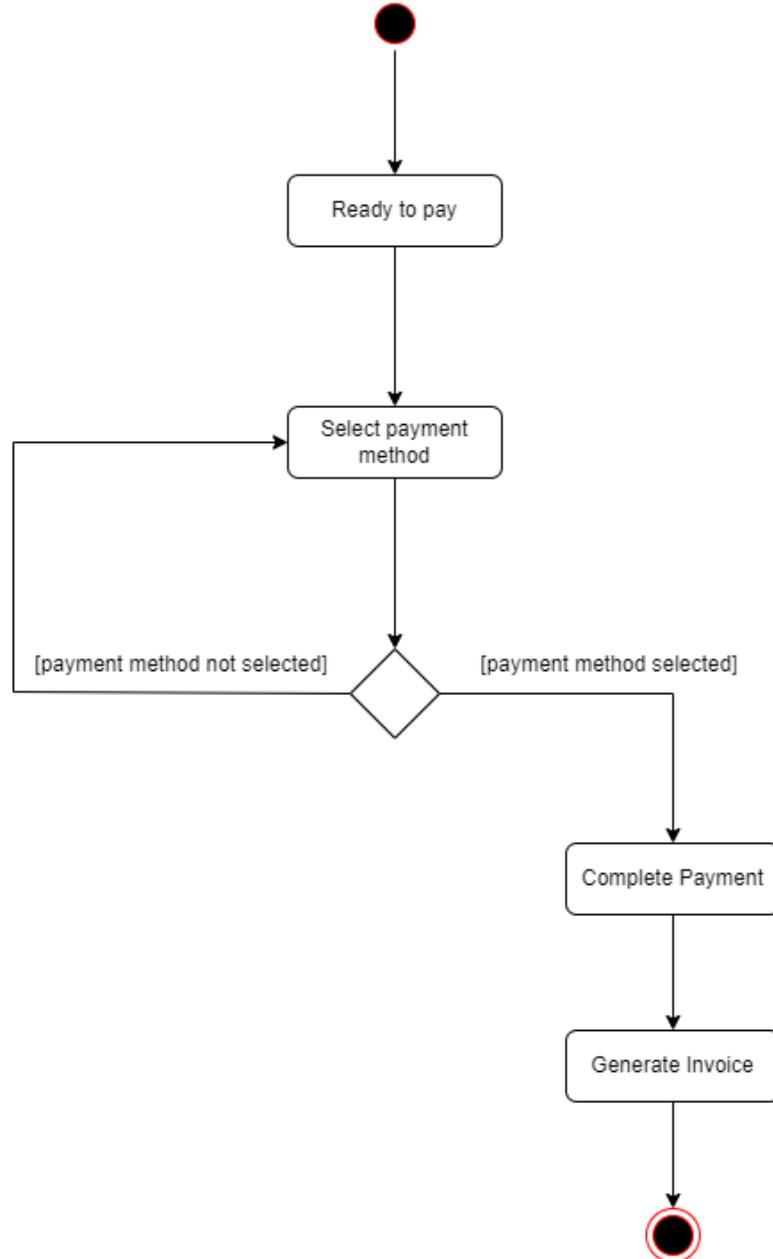


Figure B. 9 Activity Diagram for Make Payments

- **Activity Diagram for Check Items**

The activity diagram for check items is given in Figure B.10

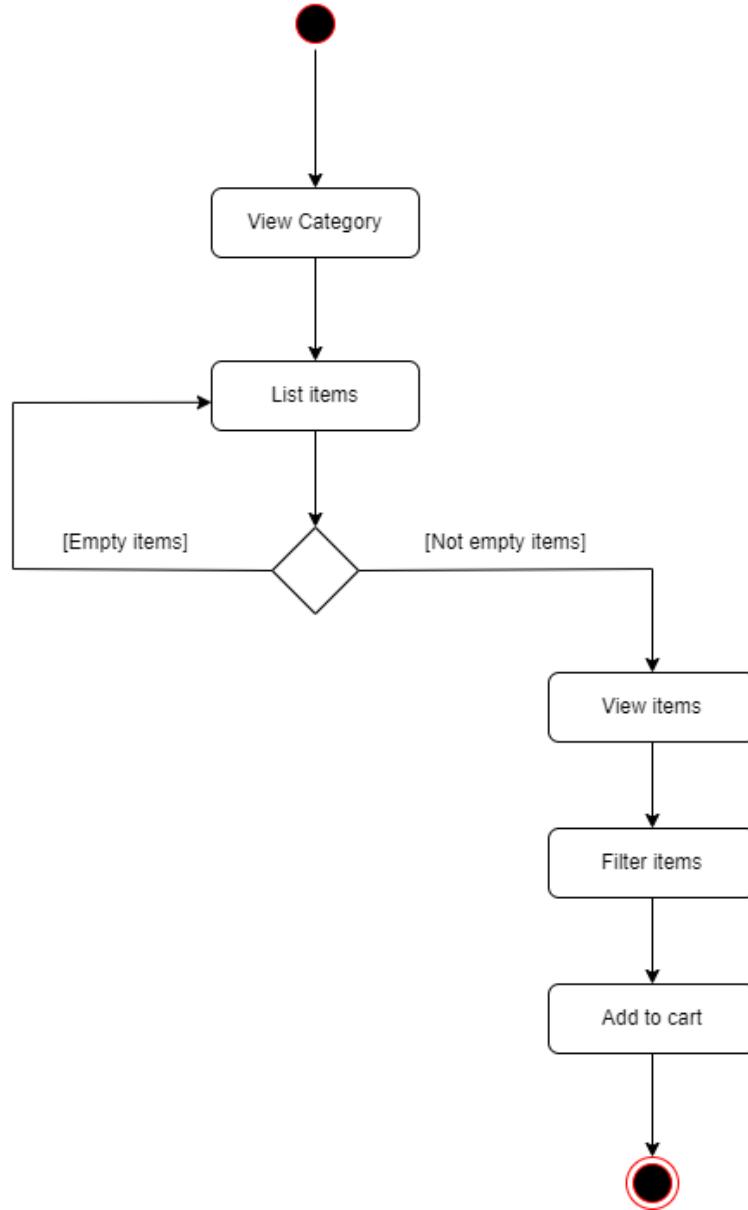


Figure B. 10 Activity Diagram for Check Items

# Appendix C – User Manual

By covering essential aspects, this manual provides the necessary instructions for Customers, Administrators, Shop Managers, Inventory Managers, Delivery Managers, and Technicians to interact with the system. This section's ultimate purpose is to make the user aware of the system's major features so that they can get the most out of it. Despite the fact that the system describes several user-level privileges, the key modules of the standard user role, Below is a brief description of various administrator setup options..

## User Manual Part 1

### User: Customer

Browse the home page of the U-Star Digital website by visiting the URL <http://localhost/ustar/>. (Figure C.1). Following the selection of the option, the consumer can proceed with the standard computer hardware shopping cart process as depicted in (Figure C.2) (Figure C.3) (Figure C.4) (Figure C.5) (Figure C.6) (Figure C.7)

The screenshot shows the homepage of the U-Star Digital website. At the top, there is a navigation bar with links for Home, Shop, About, Services, Contact, My Account, and Cart. Below the navigation bar is a search bar with the placeholder "Search by Item Name" and a "Search" button. The main content area is titled "Search Item". It displays four categories of computer hardware:

- PROCESSORS (5)**: An image of an AMD Ryzen processor chip. Description: A processor (CPU) is the logic circuitry that responds to and processes the basic instructions that drive a computer. The CPU is seen as the main and most crucial integrated circuit (IC) chip in a computer, as it is responsible for interpreting most of computers commands. [View Category](#)
- MEMORY (RAM) (0)**: An image of RAM modules on a circuit board. Description: Computer memory is the storage space in the computer, where data is to be processed and instructions required for processing are stored. The memory is divided into large number of small parts called cells. Each location or cell has a unique address, which varies from zero to memory size minus one. [View Category](#)
- GRAPHIC CARDS (4)**: An image of three graphics cards (GIGABYTE RTX 3070, GIGABYTE RX 6700 XT). Description: A graphics card is a type of display adapter or video card installed within most computing devices to display graphical data with high clarity, color, definition and overall appearance. [View Category](#)
- STORAGE (1)**: An image of a hard disk drive. Description: Disk storage (also sometimes called drive storage) is a general category of storage mechanisms where data is recorded by various electronic, magnetic, optical, or mechanical changes to a surface layer of one or more rotating disks. A disk drive is a device implementing such a storage mechanism. [View Category](#)

Figure C. 1: Home Page

Customers can start the shopping experience by selecting the category.

## Search Item



**PROCESSORS (4)**

A central processing unit (CPU), also called a central processor, main processor or just processor, is the electronic circuitry that executes instructions comprising a computer program. The CPU performs basic arithmetic, logic, controlling, and input/output (I/O) operations specified by the instructions in the program.

[View Category](#)



**MOTHERBOARD (4)**

A motherboard (also called mainboard, main circuit board, [1] or mobo) is the main printed circuit board (PCB) in general-purpose computers and other expandable systems. It holds and allows communication between many of the crucial electronic components of a system, such as the central processing unit (CPU) and memory, and provides connectors for other peripherals. Unlike a backplane, a motherboard usually contains significant sub-systems, such as the central processor, the chipsets' input/output and memory controllers, interface connectors, and other components integrated for general use.

[View Category](#)



**MEMORY (RAM) (1)**

RAM is short for "random access memory" and while it might sound mysterious, RAM is one of the most fundamental elements of computing.

[View Category](#)



**GRAPHIC CARDS (2)**

A graphics card (also called a video card, display card, graphics adapter/vga card/vga, video adapter, or display adapter) is an expansion card which

[View Category](#)

Figure C. 2: Category Page

After selecting a category customers can view the selected category product listing.



**ASUS ROG MAXIMUS Z690 EXTREME GLACIAL**

Processor Number i9-12900KF - Recommended  
Customer Price ; Total Cores 16 - Maximum Turbo Power ; Embedded Options Available No; Datasheet ;

[View Item](#)

**LKR 16,500.00**



**ASUS ROG MAXIMUS Z690 EXTREME**

Processor Number i9-12900KF - Recommended  
Customer Price ; Total Cores 16 - Maximum Turbo Power ; Embedded Options Available No; Datasheet ;

[View Item](#)

**LKR 153,000.00**

**Sale LKR 142,000.00**



**ASUS ROG MAXIMUS Z690 HERO**

Processor Number i9-12900KF - Recommended  
Customer Price ; Total Cores 16 - Maximum Turbo Power ; Embedded Options Available No; Datasheet ;

[View Item](#)

**LKR 143,000.00**

**Sale LKR 132,000.00**

Figure C. 3: List Item Page

67

Individual item page for the selected item.



**ASUS ROG MAXIMUS Z690 EXTREME**

Processor Number i9-12900KF - Recommended Customer Price ; Total Cores 16 . Maximum Turbo Power ; Embedded Options Available No; Datasheet ;

LKR 153,000.00

Sale LKR 142,000.00

Add to Cart 

 Continue Shopping

Figure C. 4: Item Page

Customers can check out the order through the cart page.

Items in Your Cart

 **ASUS ROG MAXIMUS Z690 EXTREME** Qty  Sale LKR: 142,000.00 

---

**Cart Summary**

Item(s):	LKR: 142,000.00
Discount:	LKR: (-11,000.00)
<b>Est. Total:</b>	<b>LKR: 131,000.00</b>

 CHECKOUT ORDER

Figure C. 5: Cart Page

Checkout page to enter customer billing and delivery details.

**Enter Your Billing Details**

Frist Name	Nuwan
Last Name	Samaranayake
Phone	0712487645
Email Address	nuwan@gmail.com
Address Line 1	No: 216/A, Play Ground Road
Address Line 2	Weedagama
City	Bandaragama
Province	Western
Zip	12530

**Enter Your Delivery Details**

Frist Name	Nuwan
Last Name	Samaranayake
Phone	0712487645
Email Address	nuwan@gmail.com
Address Line 1	No: 216/A, Play Ground Road
Address Line 2	Weedagama
City	Bandaragama
Province	Western
Zip	12530

Use same address as a delivery address

**Payment Method**

Cash On Delivery (COD)

**Order Summary**

Item(s):	LKR: 98,000.00
Discount:	LKR: (-4,000.00 )
Est Total:	LKR: 94,000.00
Delivery Charges:	LKR: 400.00
<b>Total:</b>	<b>LKR: 94,400.00</b>

PAY YOUR ORDER

Figure C. 6: Checkout Page

Complete order by generating the invoice.

## 📍 Invoice for your Order



Date: 2022-05-25

From  
**U-Star Digital**  
Kaluthra Road  
Bandaragama, 12530  
Phone: (804) 123-5432  
Email: info@ustardigital.com

To  
**Uchitha Perera**  
123/D, Play round Road  
Panadura  
Phone: 0712487644  
Email: uchitha@gmail.com

**Invoice No:** 202205250009  
**Account:** 968-34567

Qty	Product	Warranty (Days)	Discount	Sale Price (LKR)	Unit Price (LKR)	Subtotal (LKR)
1	ASUS STRIX GAMING Radeon RX6700XT 12GB	1080	7%	260,000.00	280,000.00	260,000.00
1	AMD RYZEN 9 5950X	1080	0%	0.00	199,000.00	199,000.00
1	Seagate IronWolf ST8000VN004 8TB 7200 RPM 256MB	1080	6%	62,500.00	66,500.00	62,500.00

Payment Method: Cash On Delivery (COD)

Please pay after the order is received. Pay for Courier agent. Don't pay more money for the courier service. If you have any problems don't hesitate to get in touch with us.

<b>Item(s):</b>	LKR: 545,500.00
<b>Discount:</b>	LKR: (-24,000.00)
<b>Est Total:</b>	LKR: 521,500.00
<b>Delivery:</b>	LKR: 400.00
<b>Total:</b>	LKR: 521,900.00

Dashboard

Figure C. 7: Invoice Page

## User Manual Part 2

### User: Inventory Manager

The inventory manager in U-Star Digital can handle the inventory items. To access the inventory manager dashboard, the inventory manager must use the [URL](http://localhost/ustar/system/inventory/).

After successful login, the inventory manager needs to add a category of the appropriate item. The inventory management process is depicted in (Figure C.8) (Figure C.9) (Figure C.10) (Figure C.11).

### Categories

Insert New Category

**Category Image**

Choose File No file chosen

**Category Name**

Enter Category Name

**Category Short Description**

Enter ...

**Insert**

This screenshot shows the 'Add Category' form. It features a blue header bar with the text 'Insert New Category'. Below this, there are three main input fields: 'Category Image' (with a file input field showing 'No file chosen'), 'Category Name' (with a text input field containing 'Enter Category Name'), and 'Category Short Description' (with a text area containing 'Enter ...'). At the bottom of the form is a blue 'Insert' button.

Figure C. 8: Add Category Form

Next need to add the appropriate brand of the item.

### Brands

Insert New Brand

**Brand Name**

Enter Brand Name

**Insert**

This screenshot shows the 'Add Brand' form. It has a blue header bar with the text 'Insert New Brand'. Below it is a single input field labeled 'Brand Name' with the placeholder 'Enter Brand Name'. At the bottom is a blue 'Insert' button.

Figure C. 9: Add Brand Form

Next need to add the appropriate Model of the item.

### Models

Insert New Models

**Category \***

- Select Category -

**Model Name**

Enter Model Name

**Insert**

This screenshot shows the 'Add Models' form. It has a blue header bar with the text 'Insert New Models'. Below it are two input fields: 'Category \*' (a dropdown menu with the placeholder '- Select Category -') and 'Model Name' (a text input field with the placeholder 'Enter Model Name'). At the bottom is a blue 'Insert' button.

Figure C. 10: Add Models Form

Next need to add appropriate item details of the item.

Add Items

Items / Add Items

Insert New Item

Item Details	Item Specifications
<b>Item Image *</b> <input type="file"/> Choose File No file chosen	<b>Graphic Engine *</b> <input type="text"/> Enter Graphic Engine
<b>Item Name *</b> <input type="text"/> ASUS STRIX GAMING RADEON RX6700XT 12GB	<b>Bus Standard *</b> <input type="text"/> Enter Bus Standard
<b>Category *</b> <input type="text"/> GRAPHIC CARDS	<b>Video Memory *</b> <input type="text"/> Enter Video Memory
<b>Brand *</b> <input type="text"/> - Select Brand -	<b>CUDA Core *</b> <input type="text"/> Enter CUDA Core
<b>Model *</b> <input type="text"/> - Select Model -	<b>Memory Speed *</b> <input type="text"/> Enter Memory Speed
<b>SKU *</b> <input type="text"/> Unique Code	<b>Memory Interface *</b> <input type="text"/> Enter Memory Interface
<b>Stock *</b> <input type="text"/> Enter Stock Count	
<b>Reorder Level *</b> <input type="text"/> Enter Reorder Level	
<b>GRN Price *</b> <input type="text"/> 158500	
<b>Unit Price *</b> <input type="text"/> 165800	
<b>Sale Price</b> <input type="text"/> 163000	
<b>Warranty Period in Days*</b> <input type="text"/> 365	
<b>Product Description</b> <input type="text"/> Additional info of the product	

**Insert**

Figure C. 11: Add Items Form

# User Manual Part 3

## User: Administrator

The Administrator in the U-Star Digital can handle the users, user modules, and user roles. To access the Administrator dashboard, the Administrator must use the <http://localhost/ustar/system/> URL. After successfully login administrator can add users to the system.

The administrative procedure is shown in (Figure C.12) and (Figure C.13).

The screenshot shows a web-based form titled 'Staff' under the heading 'Insert New Staff Member'. The form is divided into two tabs: 'Personal Details' (selected) and 'Contact Details'. The 'Personal Details' tab contains the following fields:

- Profile Image \***: A file input field showing 'No file chosen'.
- First Name**: An input field with placeholder 'Enter First Name'.
- Last Name**: An input field with placeholder 'Enter Last Name'.
- NIC**: An input field with placeholder 'Enter NIC'.
- Date of Birth**: A date input field with placeholder 'mm/dd/yyyy' and a calendar icon.
- Username**: An input field with placeholder 'Enter Username'.
- Password**: An input field with placeholder 'Password'.
- Verify Password**: An input field with placeholder 'Verify Password'.

At the bottom left of the form is a blue button labeled 'Insert' with a database icon.

Figure C. 12: Add Employee

## Modules

Insert New Module

**Main Module ID**

2 Digits

**Main Module Name**

Staff Management

**Main Module Folder Name**

Main Module folder Path Eg :- users/customers

**Main Module Icon**

Main Module Icon

 Insert

Figure C. 13: Add user modules

# Appendix D – Management Reports

The system may produce the following key reports, which are displayed below.

- **Revenue Detail Report**

Figure D.1 shows all the revenue report that the shop manager saw between May 17th and March 25th, 2022.

All Sales Item List								
Category		Brand	Model		Search Table Data			
Start Date:	05/17/2022	End Date:	05/25/2022	Export to Excel				
- Select Category -		- Select Brand -	- Select Model -		Search Data		Search	
<b>Total Purchased Price: LKR 807,100.00 Total Net Price: LKR 845,500.00 Total Revenue: LKR 38,400.00</b>								
Item Qty	Item Name	Purchased Price LKR	Net Price LKR	Revenue LKR	Category	Brand	Model	Stock Date
1	ASUS STRIX GAMING Radeon RX6700XT 12GB	250,000.00	260,000.00	10,000.00	GRAPHIC CARDS	ASUS	RTX 3080TI	2022-05-24
3	GIGABYTE B660M GAMING AC DDR4	90,000.00	96,000.00	6,000.00	MOTHERBOARDS	GIGABYTE	B450	2022-05-21
1	INTEL CORE I9-12900K PROCESSOR	158,400.00	160,500.00	2,100.00	PROCESSORS	INTEL	Core i9	2022-05-23
5	MSI GT1030 AERO ITX 2GB	128,700.00	141,500.00	12,800.00	GRAPHIC CARDS	MSI	GT 1030	2022-05-25
3	Seagate IronWolf ST8000VN004 8TB 7200 RPM 256MB	180,000.00	187,500.00	7,500.00	STORAGE	SEAGATE	IRONWOLF	2022-05-19
Total Result Count: 5								

Figure D. 1: Report of Orders

- **Revenue Summary Report**

Figure D.2 was generated to view a summary of the revenue.

All Sales Item List					
Start Date:	mm/dd/yyyy	End Date:	mm/dd/yyyy	- Select Filter -	View
<b>Total Purchased Price: LKR 1,336,720.00 Total Net Price: LKR 1,455,400.00 Total Revenue: LKR 118,680.00</b>					
Item Qty	Date	Purchased Price LKR	Net Price LKR	Revenue LKR	
6	2022-05-23	386,750.00	419,500.00	32,750.00	
6	2022-05-24	461,970.00	514,400.00	52,430.00	
3	2022-05-25	488,000.00	521,500.00	33,500.00	
Total Result Count: 3					

Figure D. 2: Report of Items

- **Delivery Report**

Figure D.3 was generated to view all delivered orders of the system.

All Delivery List					
Start Date: mm/dd/yyyy		End Date: mm/dd/yyyy		Today	Export to Excel
City	Company	Search Table Data			
- Select City -	- Select Company	Search Data		Search	
<b>Total Price: LKR 2,400.00</b>					
Order Number	Date	Customer	City	Company Name	Price LKR
202205230001	2022-05-23	Nuwan Samaranayake	Bandaragama	Domestic Express (Pvt) Ltd	600.00
202205230002	2022-05-23	Geeth Fonseka	Kiribathgoda	Prompt Xpress (Pvt) Ltd	600.00
202205240007	2022-05-24	Saman Kumara	Kirilopone	Domestic Express (Pvt) Ltd	600.00
202205240008	2022-05-24	Uchitha Perera	Panadura	Domestic Express (Pvt) Ltd	600.00

Total Result Count: 4

*Figure D. 3: Report of Order Deliveries*

- **Inventory Report**

Figure D.4 was generated to view all items in the system.

All Inventory List										
Start Date: mm/dd/yyyy		End Date: mm/dd/yyyy		Today		Export to Excel				
Item Status		Category		Brand		Model		Search Table Data		
- Select Status -		- Select Category		- Select Brand -		- Select Model -		Search Data	Search	
<b>Total Stock Count: 136 Total Stock Value: LKR 15,798,070.00</b>										
Item Id	Stock Date	Item Name	Stock	Reorder Level	SKU	Purchased Price LKR	Total Purchased Cost LKR	Mark Price LKR	Discount Price LKR	Discount Rate
1	2022-05-23	INTEL CORE I9-12900K PROCESSOR	11	4	11254488548	158,400.00	1,742,400.00	162,000.00	160,500.00	1%
<b>Warranty (Days) 1080</b>										
<b>Category</b> PROCESSORS										
<b>Brand</b> INTEL										
<b>Model</b> Core i9										
<b>Status</b> Active										
2	2022-05-14	AMD RYZEN 9 5950X	11	5	112554889	178,000.00	1,958,000.00	199,000.00	0.00	0%

*Figure D. 4: Report of Stock*

- **Customer Report**

Figure D.5 was generated to view all customers in the system.

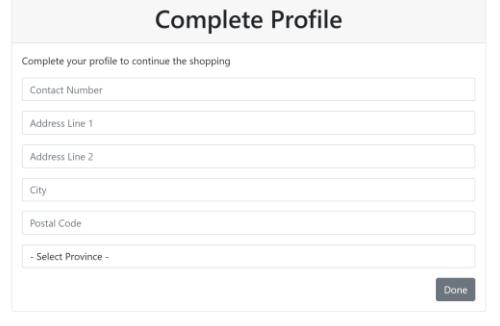
All Customer List								
User Status		City		Province		Search Table Data		
Profile Image	Register Date	Username	Name	Contact Number	Email	Address	City	Postal Code
	2022-05-23	nuwan	Nuwan Samaranayake	0712487645	nuwan@gmail.com	No: 216/A, Play Ground Road, Weedagama, Bandaragama	Bandaragama	12530
<b>Province</b> Western								
<b>Profile Status</b> Active								
	2022-05-23	geeth	Geeth Fonseka	0712487688	geeth@gmail.com	No: 266/A, Kandy Road, Kiribathgoda	Kiribathgoda	00158

Figure D. 5: Report of Product Categories

# Appendix E – Test Results

## Test Result for Customer Registration

Table E.1 shows test results of customer registration

Test No	Test Case	Expected Result	Actual Result
1	Submit the form without filling in the mandatory fields.	Prevent submitting and show error messages	
2	Use already exists username and email address	show error message	
3	Enter already exists email address	show error message	
4	Incorrect Password length insert	show error message	
6	Invalid phone number insert	show error message	
7	Enter the numbers in the text fields	show error message	
8	Invalid zip inserts	show error message	
9	Submit Correct Register Data	Navigate to the complete profile page	

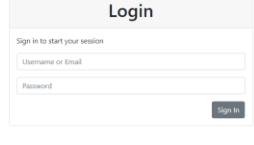
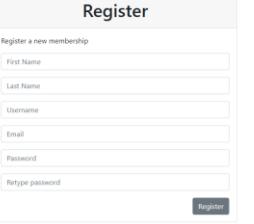
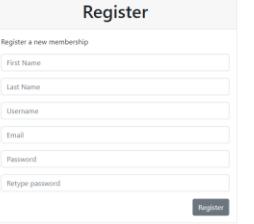
10	Submit the form with the correct data	Redirect to the login page		
11	Without login URL browsing to the customer dashboard	Redirect to login page		

Table E. 1: Test Result for Customer Registration

### Test Result for Customer Login

Table E.2 depicted the customer login test result

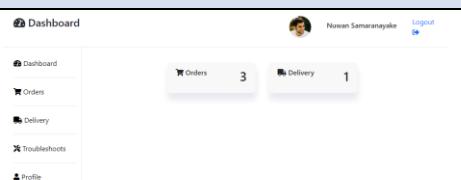
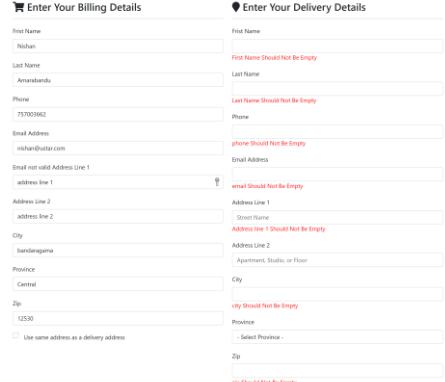
Test No	Test Case	Expected Result	Actual Result
1	Enter valid username and password	Successful login to the system	
2	Enter username only	show error message	
3	Enter only password	show error message	
4	Enter invalid username and password	show error message	

Table E. 2: Test Result for Customer Login

## Test Result for Checkout Process

Table E.3 belongs to the test result of the system order checkout process.

Test No	Test Case	Expected Result	Actual Result
1	Submit the form without filling in the mandatory fields.	Prevent submitting and show error messages	
2	Enter the numbers in the text fields	show error message	
3	Invalid phone number insert	show error message	
4	Invalid zip inserts	show error message	

5	Without check the delivery check box	Show error message of delivery form	<p><b>Enter Your Delivery Details</b></p> <p>First Name <input type="text"/> First Name Should Not Be Empty</p> <p>Last Name <input type="text"/> Last Name Should Not Be Empty</p> <p>Phone <input type="text"/> phone Should Not Be Empty</p> <p>Email Address <input type="text"/> email Should Not Be Empty</p> <p>Address Line 1 <input type="text"/> Address line 1 Should Not Be Empty</p> <p>Street Name <input type="text"/></p> <p>Address Line 2 <input type="text"/> Apartment, Studio, or Floor</p> <p>City <input type="text"/> city Should Not Be Empty</p> <p>Province <input type="text"/> - Select Province -</p> <p>Zip <input type="text"/> zip Should Not Be Empty</p>																															
6	Without selecting the payment method	Show error message	<p><b>Payment Method</b></p> <p><input type="radio"/> Cash On Delivery (COD) Select Your Payment Method</p>																															
7	Change the delivery value related to the province	Show delivery charges and calculate to total	<p><b>Order Summary</b></p> <table border="1"> <tr> <td>Item(s):</td> <td>LKR: 130,000.00</td> </tr> <tr> <td>Discount:</td> <td>LKR: 5,000</td> </tr> <tr> <td>Delivery Charges:</td> <td>LKR: 0</td> </tr> <tr> <td><b>Total:</b></td> <td><b>LKR: 125,000.00</b></td> </tr> </table> <p><b>Order Summary</b></p> <table border="1"> <tr> <td>Item(s):</td> <td>LKR: 130,000.00</td> </tr> <tr> <td>Discount:</td> <td>LKR: (-5,000.00)</td> </tr> <tr> <td>Delivery Charges:</td> <td>LKR: 400.00</td> </tr> <tr> <td><b>Total:</b></td> <td><b>LKR: 125,400.00</b></td> </tr> </table>	Item(s):	LKR: 130,000.00	Discount:	LKR: 5,000	Delivery Charges:	LKR: 0	<b>Total:</b>	<b>LKR: 125,000.00</b>	Item(s):	LKR: 130,000.00	Discount:	LKR: (-5,000.00)	Delivery Charges:	LKR: 400.00	<b>Total:</b>	<b>LKR: 125,400.00</b>															
Item(s):	LKR: 130,000.00																																	
Discount:	LKR: 5,000																																	
Delivery Charges:	LKR: 0																																	
<b>Total:</b>	<b>LKR: 125,000.00</b>																																	
Item(s):	LKR: 130,000.00																																	
Discount:	LKR: (-5,000.00)																																	
Delivery Charges:	LKR: 400.00																																	
<b>Total:</b>	<b>LKR: 125,400.00</b>																																	
8	Submit the form with the correct data	direct to the invoice page	<p><b>Invoice for your Order</b></p> <p><b>U-Star Digital</b></p> <p>From: U-Star Digital Kaluethra Road Bambaradeniya, 1230 Phone: (004) 123-5432 Email: info@ustardigital.com</p> <p>To: Nuwan Samanayake No: 216/A, Play Ground Road Viharamahadevi, 1000 Phone: 0712467645 Email: nuwan@gmail.com</p> <p>Invoice No: 202205250010 Account: 988-34567</p> <table border="1"> <thead> <tr> <th>Qty</th> <th>Product</th> <th>Warranty (Days)</th> <th>Discount</th> <th>Sale Price (LKR)</th> <th>Unit Price (LKR)</th> <th>Subtotal (LKR)</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>INTEL CORE i9-12800K PROCESSOR</td> <td>1800</td> <td>7%</td> <td>160,500.00</td> <td>162,000.00</td> <td>160,500.00</td> </tr> <tr> <td>1</td> <td>AMD RYZEN 5 3600</td> <td>1080</td> <td>0%</td> <td>0.00</td> <td>55,000.00</td> <td>55,000.00</td> </tr> </tbody> </table> <p>Payment Method: Cash On Delivery (COD)</p> <p>Please pay after the order is received. Pay to Courier agent. Don't pay more money for the courier service. If you have any problems, don't hesitate to get in touch with us.</p> <table border="1"> <tr> <td>Item(s):</td> <td>LKR: 217,000.00</td> </tr> <tr> <td>Discount:</td> <td>LKR: (-1,500.00)</td> </tr> <tr> <td>Ext Total:</td> <td>LKR: 215,500.00</td> </tr> <tr> <td>Delivery:</td> <td>LKR: 400.00</td> </tr> <tr> <td><b>Total:</b></td> <td><b>LKR: 215,900.00</b></td> </tr> </table> <p><b>Download</b></p>	Qty	Product	Warranty (Days)	Discount	Sale Price (LKR)	Unit Price (LKR)	Subtotal (LKR)	1	INTEL CORE i9-12800K PROCESSOR	1800	7%	160,500.00	162,000.00	160,500.00	1	AMD RYZEN 5 3600	1080	0%	0.00	55,000.00	55,000.00	Item(s):	LKR: 217,000.00	Discount:	LKR: (-1,500.00)	Ext Total:	LKR: 215,500.00	Delivery:	LKR: 400.00	<b>Total:</b>	<b>LKR: 215,900.00</b>
Qty	Product	Warranty (Days)	Discount	Sale Price (LKR)	Unit Price (LKR)	Subtotal (LKR)																												
1	INTEL CORE i9-12800K PROCESSOR	1800	7%	160,500.00	162,000.00	160,500.00																												
1	AMD RYZEN 5 3600	1080	0%	0.00	55,000.00	55,000.00																												
Item(s):	LKR: 217,000.00																																	
Discount:	LKR: (-1,500.00)																																	
Ext Total:	LKR: 215,500.00																																	
Delivery:	LKR: 400.00																																	
<b>Total:</b>	<b>LKR: 215,900.00</b>																																	

9	Without login URL browsing to the checkout and invoice	Redirect to the cart page	
---	--	---------------------------	--

Table E. 3: Test Result for Checkout Process

- **User Acceptance Test Results**

Figure E.1, Figure E.2, E.3, and Figure E.4 show the results of user acceptance testing.

<u><b>User Feedback Form – Computer Hardware Purchasing and Troubleshooting Assistant Management System for U-Star Digital</b></u>				
Role of User: ..... <i>Administrator</i> .....				
Test Date: ..... <i>19/03/2022</i> .....				
Mark (✓) inside the box. Please provide a more comprehensive response with your honest option.				
No	Description	Exceptional	Satisfied	Bad
1	User interface design		✓	
2	system's response time		✓	
3	Form validation		✓	
4	Understandability of error messages		✓	
5	Interaction with navigation links and buttons	✓		
6	system's responsiveness	✓		
7	Report generating capability	✓		
8	Recognizing the system's design		✓	
9	Efficiency of the functionalities		✓	
10	Overall impressions of the system		✓	

Thank You

Figure E. 1: User Acceptance Test Result One

**User Feedback Form – Computer Hardware Purchasing and Troubleshooting**  
**Assistant Management System for U-Star Digital**

Role of User: Delivery Manager  
 Test Date: .....27/03/2022.....

Mark (✓) inside the box. Please provide a more comprehensive response with your honest option.

No	Description	Exceptional	Satisfied	Bad
1	User interface design	✓		
2	system's response time	✓		
3	Form validation		✓	
4	Understandability of error messages		✓	
5	Interaction with navigation links and buttons		✓	
6	system's responsiveness		✓	
7	Report generating capability		✓	
8	Recognizing the system's design		✓	
9	Efficiency of the functionalities		✓	
10	Overall impressions of the system	✓		

Thank You

*Figure E. 2: User Acceptance Test Result Two*

**User Feedback Form – Computer Hardware Purchasing and Troubleshooting**  
**Assistant Management System for U-Star Digital**

Role of User: Shop Manager  
 Test Date: .....27/03/2022.....

Mark (✓) inside the box. Please provide a more comprehensive response with your honest option.

No	Description	Exceptional	Satisfied	Bad
1	User interface design		✓	
2	system's response time		✓	
3	Form validation	✓		
4	Understandability of error messages		✓	
5	Interaction with navigation links and buttons	✓		
6	system's responsiveness		✓	
7	Report generating capability		✓	
8	Recognizing the system's design		✓	
9	Efficiency of the functionalities	✓		
10	Overall impressions of the system		✓	

Thank You

*Figure E. 3: User Acceptance Test Result Three*

**User Feedback Form – Computer Hardware Purchasing and Troubleshooting**  
**Assistant Management System for U-Star Digital**

Role of User: Theничети.....  
 Test Date: 24/03/2021.....

Mark (✓) inside the box. Please provide a more comprehensive response with your honest option.

No	Description	Exceptional	Satisfied	Bad
1	User interface design	✓		
2	system's response time	✓		
3	Form validation		✓	
4	Understandability of error messages		✓	
5	Interaction with navigation links and buttons		✓	
6	system's responsiveness		✓	
7	Report generating capability	✓		
8	Recognizing the system's design		✓	
9	Efficiency of the functionalities	✓		
10	Overall impressions of the system		✓	

Thank You

*Figure E. 4: User Acceptance Test Result four*

# Appendix F – Client Certificate



**U-star Digital**

Kalutara Road, Bandaragama, Sri Lanka

Tel : 038-2293948 / 071-8180787

ustardigital@gmail.com | www.ustardigital.com

25<sup>th</sup> April 2022,

The Coordinator,

External Degree Center,

University of Colombo School of Computing,

Dear Sir,

**Certification letter of the acceptance of Web-Based Computer Hardware Purchasing  
and Troubleshooting Assistant Management System**

Mr. G.P Sankalpa (Index No. 1805274) designed, developed, and implemented the Web-Based Computer Hardware Purchasing and Troubleshooting Assistant Management System at our company, and I thus attest that "U-Star Digital" has approved the system.

We are pleased to report that the system met our first business requirements, and we anticipate that it will assist us in efficiently and successfully enriching our business processes. We are happy to be employing a Sri Lankan student's native software product, and we appreciate him choosing "U-Star Digital" as his customer.

Thank You.

Yours sincerely

  
Samudu Kannangara

  
U-STAR DIGITAL  
Digital Marketing Services  
Tel: 038-2293948 / 071-8180787  
Email: ustardigital@gmail.com | www.ustardigital.com

# Glossary

**AJAX** – Asynchronous JavaScript and XML is a collection of web development strategies that make use of a variety of web technologies.

**FIFO** – It is a cost flow assumption approach used in the computation of the cost of goods sold. The FIFO technique assumes that a company's oldest products have been sold first. The calculations are based on the prices paid for the oldest products.

**CSS** – Cascading Style Sheet is a style sheet language that describes how a document authored in a markup language should be presented.

**IDE** – IDE stands for Integrated Development Environment, which is a tool that allows computer programmers to develop, run, debug, and inspect their code.

**MySQL** – MySQL is an open-source relational database management system based on structured query language that is backed by Oracle.

**JavaScript** – An interpreted programming language. Use to create dynamic front-end developments.

**jQuery** – JavaScript library that is used in conjunction with HTML. It's a JavaScript library that's quick, tiny, and packed with features.

# Index

---

## A

- AI** ..... 48  
Ajax ..... 27  
**AJAX** ..... xii, 86  
Apache ..... iii, xii, 27, 28, 52  
**application** ..... 15, 28, 48  
**assistant** ..... 3, 48, 49
- 

## B

- BIT** ..... xii
- 

## C

- Cart ..... 2, 8, 11, 68  
**CD** ..... xii, 53  
Checkout 25, 34, 42, 43, 56, 57, 62, 69, 80, 82  
computers ..... iii, 1  
CSS ..... iii, xii, 27, 29, 35, 86  
customers ..... iii, 2, 11, 17, 49  
Customers .. iii, 1, 6, 8, 9, 48, 66, 67, 68
- 

## D

- database ..... iii, 4, 15, 29, 48, 52, 53, 86  
delivery ..... iii, 1, 2, 3, 6, 42, 57, 69, 81  
Delivery... 3, 5, 8, 11, 17, 19, 30, 66, 76  
Dissertation ..... 3  
documents ..... iii, 28, 47  
Documents ..... 5, 6
- 

---

## E

- ER** ..... xii, 21  
Evaluation ..... xii, 4, 39, 43, 45, 46
- 

## F

- FIFO** ..... xii, 86
- 

## H

- hardware ..... 1, 2, 3, 6, 17, 27, 49, 66  
Hardwarei, 2, 10, 11, 15, 27, 28, 44, 47, 51

- HTML ..... iii, xii, 27, 29, 86
- 

## I

- IDE** ..... xii, 27, 86  
Interviews ..... 5  
Inventory 2, 3, 8, 11, 15, 17, 19, 20, 26, 30, 66, 70
- 

## J

- JavaScript ..... iii, xii, 27, 29, 35, 86  
JQuery ..... 27, 29
- 

## M

- Microcenter ..... 10, 11  
MySQL ..... iii, 27, 29, 52, 86

---

**N**

Newegg ..... 11, 50

---

**O**

Observation ..... 5, 6

order .... iii, 1, 2, 3, 5, 6, 8, 9, 10, 27, 35,  
39, 56, 57, 62, 68, 70, 75

---

**P**

Payment..... 2, 8, 11

**PEB** ..... xii

**PHP** ..... iii, xii, 27, 29

**Proactive** ..... 39

purchased ..... 1

---

**R**

**Reactive** ..... 39

Report..... 3, 9, 30, 44, 75, 76, 77

Requirements ..... 5, 8, 9, 13, 51

**RUP** ..... xii, 12, 13, 50

---

**S**

Scrum ..... 12, 13

**Security** ..... 10

**SQL** ..... xii

student ..... 47

---

**T**

technician ..... iii, 1, 6

Techniques ..... 5

**Test** .. 39, 41, 42, 43, 44, 45, 50, 78, 79,  
80, 82, 83, 84

**troubleshooting**..... 3, 17, 48

Troubleshooting .... i, 2, 3, 9, 10, 11, 15,  
27, 30, 44, 47

---

**U**

**UAT** ..... xii, 43

**UCSC** ..... xii

**UI** ..... xii, 24, 25

**UML** ..... xii

**URL** .... xii, 8, 42, 43, 51, 70, 73, 79, 82

Use Case..... 17, 19, 20, 54, 56, 57

U-Star Digital.iii, iv, 1, 2, 6, 27, 44, 47,  
70, 73

---

**V**

**Validation**..... 32, 33, 40, 50

**Verification** ..... 40, 50

Visual Studio Code ..... iii, 27, 51

---

**W**

web-based system ..... iii

---

**X**

**XAMMP** ..... xii, 52

**XP** ..... xii, 12, 13

**IT 6105 – Project Dissertation Self-Evaluation Form**

**Index No:** 1605274

**Date:** 26/04/2022

Tick **Very Poor** if item non-existent. Tick **Very Good** if excellent with no mistakes whatsoever.

	<b>Very Poor</b>	<b>Poor</b>	<b>OK</b>	<b>Good</b>	<b>Very Good</b>
<b>Abstract (4)</b>			✓		
<b>1. Introduction (9)</b>					
Motivation			✓		
Aims & Objectives of the Project			✓		
Scope of the Project			✓		
<b>2. Analysis (16)</b>					
Review of similar systems with references				✓	
Explaining the current system using diagrams				✓	
Identifying the functional and non-functional requirements			✓		
Justification for the choice of the development life cycle				✓	
<b>3. Design (20)</b>					
Comparison of alternative design strategies			✓		
Justification for the selected design strategy			✓		
Architectural design of the system			✓		
Data modelling diagrams			✓		
User interface design				✓	
<b>4. Implementation (10)</b>					
Implementation Environment			✓		
Justification for the choice of implementation platform			✓		
Code and Module structure descriptions			✓		
Acknowledgement of any reused existing codes / APIs			✓		
<b>5. Evaluation (14)</b>					
Description of testing approach			✓		
Test plan				✓	
Proof of testing of work / Results of work			✓		
User Evaluation				✓	
<b>6. Conclusion (12)</b>					
Critical evaluation of the project			✓		
Lesson learnt				✓	
Future work				✓	
Column tick count			16	8	
X	-1	1	2	3	4
Marks			32	24	
Total for content (85) =	56				

<b>7. Appendices (5)</b>	<b>Very Poor</b>	<b>Poor</b>	<b>Ok</b>	<b>Good</b>	<b>Very Good</b>
System Manual			✓		
Design Documentation			✓		
User Manual			✓		
Management Reports			✓		
Test Cases / Results			✓		
Column tick count			5		
x	0	0.25	0.5	0.75	1
Marks			2.5		
<b>Total for Appendices (/5) =</b>	5				
Client Certificate (Yes/No)	Yes				

<b>8. Formatting (10)</b>	<b>Poor</b>	<b>Ok</b>	<b>Good</b>	
Acknowledgement		✓		
Table of contents			✓	
List of figures / List of tables			✓	
List of Acronyms			✓	
Grammar and writing style		✓		
Spellings and use of punctuations		✓		
All figures and tables referenced in the text		✓		
All references cited in text		✓		
Use reference in IEEE format		✓		
Page numbering		✓		
Column tick count	7		3	
x	0.2	0.5	1	
Marks		8.5		
<b>Total for formatting (/10) =</b>	8.5			

**TOTAL MARKS FOR DISSERTATION (/100) =**  
**(Content Marks + Appendices Marks + Formatting Marks)** 65

C. P. Sunil Kumar

Candidate signature: