Software Requirements Specification

For

Online Platform for Purchasing Computers, Requesting Repairs, and Booking CCTV Installation

Prepared by

Sangita Mirashi	**Mirashisangita712@gmail.com**
Mansi Parmar	**mansiparmar1619@gmail.com**

Instructor: ***<Prof. Shyamsundar P. Magar>***

Course: **<Web Technology>**

Lab Section: ***<Web Technology >***

Teaching Assistant: ***< Prof. Shyamsundar P. Magar***

Date: **< 28-3-2025 >**

Contents

Revisions

Version	Primary Author(s)	Description of Version	Date Completed
Draft Type and Number	Full Name	Information about the revision. This table does not need to be filled in whenever a document is touched, only when the version is being upgraded.	00/00/00

1 Introduction

1.1 Document Purpose

This document specifies the software requirements for NP Solutions – "Innovate. Repair. Protect." This online platform is designed to facilitate the purchase of computers, request for repair services, and booking of CCTV installations. The document details the functional and non-functional requirements, ensuring a comprehensive understanding of the system's capabilities and constraints.

This SRS covers the entire system, including user registration, product catalog, service request management, order tracking, and an administrative panel for managing inventory and transactions. It serves as a guide for developers, testers, and stakeholders, ensuring a clear roadmap for the development and deployment of the platform.

1.2 Product Scope

The **Online Platform** is a web-based solution that allows customers to browse and purchase computers, request repair services for their devices, and book CCTV installation services. The platform integrates offline operations, such as inventory management and service scheduling, with online functionalities to provide a unified customer experience. Key benefits include:

- **Improved Customer Experience:** Customers can easily access services and products online, reducing the need for physical visits.
- **Operational Efficiency:** Streamlined processes for inventory management, service requests, and scheduling.
- **Increased Sales:** Enhanced product visibility and easy access to services.

1.3 Intended Audience and Document Overview

This document is intended for the following audiences:

- **Developers:** To understand the system requirements and design the platform accordingly.
- **Project Managers:** To oversee the development process and ensure alignment with business goals.
- **Stakeholders:** To review the system's capabilities and provide feedback.
- **Testers:** To create test cases based on the requirements outlined in this document.

The document is organized into sections that provide an overview of the product, detailed requirements, and additional information such as performance and security requirements. Readers are encouraged to start with the introduction and overall description before diving into specific requirements.

1.4 Definitions, Acronyms and Abbreviations

- **CCTV** - Closed-Circuit Television
- **CSS** - Cascading Style Sheets
- **HTML** - Hypertext Markup Language
- **HTTP** - Hypertext Transfer Protocol
- **IEEE** - Institute of Electrical and Electronics Engineers
- **MySQL** - Structured Query Language-based database
- **PHP** - Hypertext Preprocessor
- **SSL/TLS** - Secure Sockets Layer / Transport Layer Security
- **UI** - User Interface

1.5 Document Conventions

This document adheres to the IEEE 830-1998 Software Requirements Specification (SRS) standard. The following conventions are used throughout this document

1. Formatting Conventions:

- **Headings**: Bold and larger font sizes are used for major section headings.
- **Subheadings**: Italicized text is used for sub-sections.
- **Bold text**: Used for key terms and section titles.
- **Monospace font**: Used for code, database field names, and system messages.

2. Terminology and Notation:

- **"Shall"** indicates a mandatory requirement.
- **"Should"** denotes a recommended requirement but not mandatory.
- **"May"** refers to an optional feature.
- **"Will"** indicates a future planned feature.

3. Numbering Scheme:

- The document follows a hierarchical numbering format (e.g., 1, 1.1, 1.2, 2, 2.1, etc.) for easy reference.

4. Diagrams and Figures:

- All diagrams and figures will be labeled as **Figure X** and referenced within the text.

5. References and Citations:

- Any external references will be cited in the **References** section.
- Industry standards and legal requirements will be listed with appropriate document numbers.

1.6 :References and Acknowledgments

References

- IEEE 830-1998 Software Requirements Specification (SRS) Standard
- Project Synopsis Document
- Online resources for e-commerce platform development
- Database design best practices documentation

Acknowledgment

We would like to express our gratitude to our project mentor, faculty members, and peers for their valuable guidance and support. Special thanks to all stakeholders for their inputs in shaping the requirements for this system.

2 Overall Description

2.1 Product Overview

The Online Platform is a web-based application designed to facilitate the purchase of computers, request repair services, and book CCTV installations. The platform integrates with existing offline operations, such as inventory management and service scheduling, to provide a seamless customer experience. The system will be accessible via web browsers and will include an administrative panel for managing inventory, service requests, and customer data.

2.2 Product Functionality

The major functions of the system include:

- **Product Catalog:** Browse and purchase computers.
- **Repair Services:** Request repair services for devices.
- **CCTV Installation:** Book CCTV installation services.
- **User Accounts:** Register and manage user accounts.
- **Order Tracking:** Track orders and service requests.
- **Admin Panel:** Manage inventory, service requests, and customer data

2.3 Design and Implementation Constraints

1. Design Constraint

The software design must follow the COMET (Collaborative Object Modeling and Enterprise Transformation) method to ensure structured object-oriented development.

All system modeling will use UML (Unified Modeling Language) diagrams, including use case diagrams, class diagrams, and sequence diagrams for clarity in system architecture.

The UI/UX design must follow standard usability guidelines to ensure accessibility and ease of use.

2. Implementation Constraints

The system will be developed using PHP or Python for backend development and MySQL as the database.

The frontend will be implemented using HTML, CSS, and JavaScript to ensure cross-platform compatibility.

Secure communication protocols, such as SSL/TLS, must be implemented for data transmission.

The system must comply with data protection laws such as GDPR for handling user information.

3. Hardware Constraints

The application must run efficiently on a server with minimum 8GB RAM, a 4-core processor, and 100GB SSD storage.

Users should be able to access the system on devices with at least 2GB RAM and a modern web browser.

4. External Dependencies

The system will rely on third-party payment gateways for secure transactions.

Hosting services will be required to deploy the application and ensure 24/7 availability.

Assumptions and Dependencies

- The platform will rely on existing inventory and customer databases.
- The system assumes that customers have access to the internet and a compatible web browser.
- The platform depends on third-party payment gateways for processing transactions.

Specific Requirements

2.4 External Interface Requirements

2.4.1 User Interfaces

The platform will have a responsive web interface accessible via desktop and mobile browsers. Key UI components include:

- **Homepage:** Display featured products and services.
- **Product Catalog:** Browse and filter computer products.
- **Service Request Forms:** Submit repair and CCTV installation requests.
- **User Dashboard:** Manage orders, service requests, and account details.

2.4.2 Hardware Interfaces

The platform will interact with the following hardware components:

- **Servers:** Host the web application and databases.
- **Printers:** Generate invoices and service tickets.
- **CCTV Cameras:** For installation services

2.4.3 Software Interfaces

1. **User Interface (UI)**

The system will have a **web-based UI** built with **HTML, CSS, and JavaScript** for user-friendly navigation.

2. Hardware Interfaces

The system will operate on **servers supporting PHP/Python and MySQL**.

Users can access the system through **modern web browsers** on Windows, macOS, Linux, Android, and iOS devices.

3. Software Interfaces

Database (MySQL): Stores user data, product details, orders, and service requests.

Payment Gateway Integration: Secure transaction processing via **third-party APIs**.

Authentication System: Uses **JWT-based authentication** for login and session management.

Email & SMS APIs: Notification system for order confirmations, status updates, and alerts.

4. Communication Interfaces

Secure HTTP (HTTPS) Protocols for encrypted communication.

RESTful APIs for backend-to-frontend communication and external service integration.

2.5 Functional Requirements

3.2.1 F1: Computer Purchase

REQ-1: Users can browse and filter products

REQ-2: Add to cart products

3.2.2 F2: Repair Service Requests

REQ-3: Users can submit repair requests

REQ-4: System assigns a technician

REQ-5: Users can track repair status

3.2.3 F3: CCTV Installation Booking

REQ-6: Users can book an installation appointment

REQ-7: System assigns a technician

REQ-8: Users receive confirmation

2.6 Use Case Model

3.3.1 Use Case #1: Purchase a Computer (U1)

Author: **Sangita Mirashi**

Purpose: **Allow users to browse and purchase computers.**

Requirements Traceability: **F1**

Priority: **High**

Preconditions: **User must be logged in.**

Post Conditions: **Order is placed, and payment is processed.**

Actors: **Customer**

Flow of Events:

- **User browses the product catalog.**
- **User selects a computer and adds it to the cart.**
- **User proceeds to checkout and completes the payment.**

3.3.2 Use Case #2: Request Repair Service (U2)

Author: Mansi Parmar

Purpose: Allow users to request repair services for their devices.

Requirements Traceability: F2

Priority: High

Preconditions: User must be logged in.

Post Conditions: Service request is submitted.

Actors: Customer

Flow of Events:

- User navigates to the repair service page.
- User fills out the repair request form.
- User submits the request.

3 Other Non-functional Requirements

3.1 Performance Requirements

- The system shall load the homepage within 3 seconds under normal conditions.
- The system shall process payments within 5 seconds.
- System should support at least 1000 concurrent users

3.2 Safety and Security Requirements

- The system shall encrypt all sensitive customer data.
- The system shall require user authentication for accessing personal data
- Secure handling of transactions

3.3 Software Quality Attributes

4.3.1 Reliability

- The system shall have an uptime of 99.9%.
- The system shall recover from failures within 5 minutes.

4.3.2 Usability

- The system shall have an intuitive user interface with a maximum of 3 clicks to access key functionalities.

4.3.3 Maintainability:

- Easy to update and extend

4 Other Requirements

- **Scalability**: The system should be able to scale to accommodate increasing user demand.
- **Accessibility**: The platform should follow accessibility guidelines to support users with disabilities.
- **Compatibility**: The system should work across different devices and browsers.
- **Data Backup**: Regular data backups should be performed to prevent data loss.
- **Compliance**: The system should adhere to relevant legal and regulatory requirements, such as data protection laws.

Appendix A – Data Dictionary

Field Name	***Data Type***	***Description***
User_ID	*INT (PK)*	*Unique identifier for users*
Username	*VARCHAR(50)*	*User's login name*
Password	*VARCHAR(255)*	*Encrypted user password*
Email	*VARCHAR(100)*	*User's email address*
Phone	*VARCHAR(15)*	*User's contact number*
Product_ID	*INT (PK)*	*Unique identifier for products*
Product_Name	*VARCHAR(100)*	*Name of the product*
Category_ID	*INT (FK)*	*Category of the product*
Price	*DECIMAL(10,2)*	*Price of the product*
Stock_Quantity	*INT*	*Number of items available*
Order_ID	*INT (PK)*	*Unique identifier for orders*
User_ID (FK)	*INT*	*Associated user placing the order*
Order_Date	*DATETIME*	*Date and time of order placement*
Status	*VARCHAR(50)*	*Current status of the order*
Service_ID	*INT (PK)*	*Unique identifier for services (repair/CCTV)*
Service_Type	*VARCHAR(50)*	*Type of service requested*
Appointment_Date	*DATETIME*	*Scheduled date for service*
Payment_ID	*INT (PK)*	*Unique identifier for payments*
Payment_Method	*VARCHAR(50)*	*Payment type (Credit Card, UPI, etc.)*
Payment_Status	*VARCHAR(50)*	*Status of the payment (Paid/Pending)*

Appendix B - Group Log

Date	Team Member	Task Description	Status
2025-03-10	Mansi Parmar	Project idea finalization	Completed
2025-03-12	Sangita Mirashi	Initial research and planning	Completed
2025-03-14	Mansi Parmar	Requirement gathering	Completed
2025-03-16	Sangita Mirashi	SRS document preparation	In Progress
2025-03-18	Mansi Parmar	UI/UX wireframe design	In Progress
2025-03-20	Sangita Mirashi	Database schema design	Pending