

SOFTWARE REQUIREMENT SPECIFICATION

For

Road Repair and Tracking System

Prepared by: **Archisman Chakraborty(224CS1004)**

Submitted to: **Dr. Judhistir Mahapatro**

Date: **05th September, 2024**

Contents

Contents	2
1 Introduction.....	4
1.1 Purpose	4
1.2 Scope	4
1.3 Definitions, Acronyms, and Abbreviations	4
1.4 References	5
1.5 Overview	5
1.6 Overview of Developer Responsibilities.....	5
2 General Description.....	5
2.1 Product Perspective	5
2.2 Product Functions Overview	6
2.3 User Classes and Characteristics	6
2.4 Operating Environment	6
2.5 Design and Implementation Constraints	7
2.6 Assumptions and Dependencies	7
3 System Features	7
3.1 Feature 1: Complaint Management	7
3.1.1 Introduction.....	7
3.1.2 Inputs.....	7
3.1.3 Processing	7
3.1.4 Outputs	8
3.2 Feature 2: Supervisor Dashboard	8
3.2.1 Introduction.....	8
3.2.2 Inputs.....	8
3.2.3 Processing	8
3.2.4 Outputs	8
3.3 Feature 3: Resource Estimation.....	8
3.3.1 Introduction.....	8
3.3.2 Inputs.....	8
3.3.3 Processing	8
3.3.4 Outputs	9
3.4 Feature 4: Automated Scheduling	9
3.4.1 Introduction.....	9
3.4.2 Inputs.....	9
3.4.3 Processing	9
3.4.4 Outputs	9
3.5 Feature 5: Resource Management	9
3.5.1 Introduction.....	9
3.5.2 Inputs.....	9
3.5.3 Processing	10
3.5.4 Outputs	10
3.6 Feature 6: Reporting and Analytics	10
3.6.1 Introduction.....	10
3.6.2 Inputs.....	10

3.6.3	Processing	10
3.6.4	Outputs	10
3.7	Feature 7: Administrator Control Panel.....	10
3.7.1	Introduction.....	10
3.7.2	Inputs.....	11
3.7.3	Processing	11
3.7.4	Outputs	11
3.8	Feature 8: User Authentication and Security.....	11
3.8.1	Introduction.....	11
3.8.2	Inputs.....	11
3.8.3	Processing	11
3.8.4	Outputs	11
4	<i>External Interface Requirements</i>	12
4.1	User Interfaces.....	12
4.2	Hardware Interfaces	12
4.3	Software Interfaces.....	12
5	<i>System Non-Functional Requirements</i>	13
5.1	Performance Requirements	13
5.2	Security Requirements	13
5.3	Reliability Requirements.....	13
5.4	Availability Requirements.....	13
5.5	Maintainability Requirements	13
5.6	Portability Requirements.....	13
5.7	Usability Requirements	13
6	<i>Other Non-Functional Requirements</i>	14
6.1	Regulatory and Compliance Requirements.....	14
6.2	Data Integrity Requirements	14
7	<i>Conclusion</i>	14

1 Introduction

1.1 Purpose

The purpose of this Software Requirements Specification (SRS) document is to outline the requirements for the Road Repair and Tracking System (RRTS), developed to automate the road repair processes for the Public Works Department of a large city's Corporation. RRTS aims to streamline complaint management, prioritize repairs based on severity, schedule resources efficiently, and provide real-time updates to stakeholders, enhancing overall operational efficiency and service delivery. This document serves as a guide for developers, project managers, and stakeholders, ensuring alignment on system functionality, design, and validation criteria.

1.2 Scope

The Road Repair and Tracking System (RRTS) aims to automate road repair processes for the city's Public Works Department, covering:

- **Complaint Management:** Handle entry, tracking, and reporting of road repair complaints from residents.
- **Supervisor Dashboard:** Enable supervisors to assess road conditions, prioritize repairs, and estimate resource needs.
- **Automated Scheduling:** Schedule repairs dynamically based on priority and resource availability.
- **Resource Management:** Manage the availability and allocation of manpower, machinery, and materials.
- **Reporting and Analytics:** Generate reports on repair statistics, outstanding tasks, and resource utilization.
- **Administrator Control Panel:** Allow administrators to manage configurations, resources, and user access.
- **User Authentication and Security:** Ensure secure access with authentication and role-based permissions.

This system will enhance road repair efficiency, resource optimization, and service delivery to the public.

1.3 Definitions, Acronyms, and Abbreviations

- **RRTS:** Road Repair and Tracking System
- **PWD:** Public Works Department
- **UI:** User Interface
- **API:** Application Programming Interface
- **DBMS:** Database Management System
- **ERP:** Enterprise Resource Planning
- **SRS:** Software Requirements Specification
- **CRUD:** Create, Read, Update, Delete
- **GPS:** Global Positioning System
- **MFA:** Multifactor Authentication

1.4 References

This section lists all the documents, standards, and sources that were consulted or used during the development of the SRS for the RRTS. It ensures that the SRS is based on sound research and existing guidelines.

1. West Bengal Government PwD, Roads: <https://pwd.wb.gov.in/roads>
2. IEEE Standard 830-1998: IEEE Recommended Practice for Software Requirements Specifications.
3. Dr. Judhistir Mahapatro – Personal guidance and insights provided on the understanding of software engineering concepts and development of the Road Repair and Tracking System SRS.

1.5 Overview

This document outlines the software requirements for the RRTS, detailing the system's functionalities, external interfaces, non-functional requirements, and design constraints. It provides the foundation for the design, development, and deployment of the system, ensuring all stakeholders have a clear understanding of the project's objectives and requirements.

1.6 Overview of Developer Responsibilities

The developer will be responsible for:

- **Development:** Designing and creating the software according to the outlined requirements and specifications. This includes coding, testing, and ensuring that all functionalities meet the needs of the agency.
- **Installation:** Setting up the software on the agency's infrastructure, including any necessary configuration and integration with existing systems.
- **Training:** Providing comprehensive training for agency staff and users to ensure effective adoption and use of the software.
- **Maintenance:** Offering ongoing support and updates to address any issues, implement improvements, and ensure the software remains functional and relevant over time.

By automating key aspects of the agency's operations, this project aims to significantly improve efficiency, reduce errors, and enhance overall service quality, thereby leading to better customer satisfaction and financial performance.

2 General Description

This section provides a high-level overview of the Road Repair and Tracking System (RRTS). It outlines the product's relationship with existing systems, describes its main functionalities, identifies the intended users and their characteristics, and defines the operating environment. Additionally, it discusses the design constraints, assumptions, and dependencies that impact the system's development and operation.

2.1 Product Perspective

RRTS is an independent, stand-alone system that automates road repair and maintenance processes within the city's Public Works Department. It integrates with the existing database and ERP systems

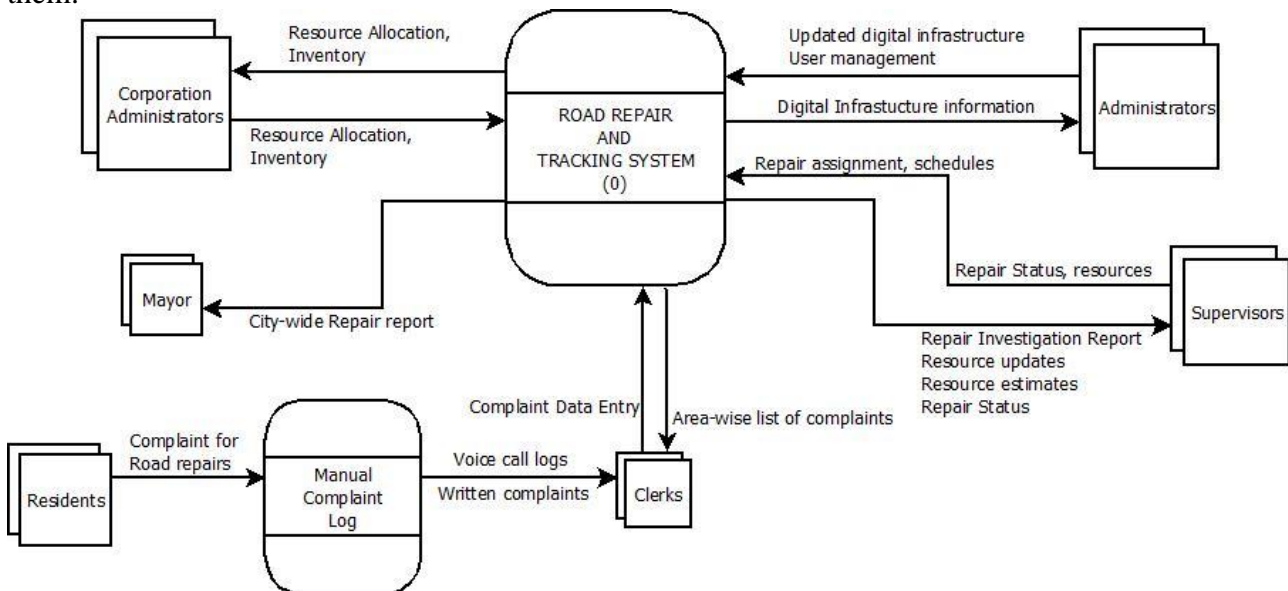
for resource management but operates primarily as a dedicated solution for managing road repair workflows and tracking.

2.2 Product Functions Overview

The primary functions of the RRTS include:

- Managing and tracking road repair complaints from residents.
- Enabling supervisors to prioritize repairs and allocate resources.
- Automating the scheduling of repair tasks.
- Managing resources such as personnel, machinery, and materials.
- Generating reports on repair statistics, resource utilization, and pending tasks.
- Providing administrative control for system configuration and resource management.
- Ensuring secure user access and role-based permissions.

The level 0 data flow diagram that follows provides this same high-level overview of the system which might include the several actors acting on it and the several functions associated with each of them.



2.3 User Classes and Characteristics

- **Residents:** Users who file road repair complaints via phone or in writing.
- **Clerks:** Staff who input complaints into the system.
- **Supervisors:** City officials responsible for assessing complaints, prioritizing repairs, and managing on-site repair activities.
- **City Corporation Administrators:** Manage system configurations, update resource data, and oversee the overall system.
- **Mayor and City Officials:** Use the system for reviewing reports and making high-level decisions.
- **Technical Support Staff:** Responsible for maintaining and troubleshooting the system.

2.4 Operating Environment

- **Hardware:** Standard desktop computers with internet access, printers for generating reports.
- **Software:** The system will run on a web-based platform compatible with major web browsers (Chrome, Firefox, Edge).

- **Operating System:** Compatible with Windows, Linux, and MacOS.
- **Database:** A relational DBMS like MySQL or PostgreSQL.
- **Network:** High-speed internet connection for real-time updates and data synchronization.

2.5 Design and Implementation Constraints

- The system must integrate with the existing city corporation's ERP and DBMS.
- Data privacy and security must comply with city IT security policies.
- The system should handle peak loads during high complaint influx periods.
- Must be developed within a budget of Rs. 2,00,000 and completed within 400 hours.

2.6 Assumptions and Dependencies

- Assumes consistent internet connectivity for real-time data processing.
- Dependent on timely updates from supervisors and administrators for accurate scheduling.
- Assumes availability of necessary resources (manpower, materials, machinery) as estimated by the system.

3 System Features

This section details the primary features of the RRTS, including the key functionalities that will be provided to meet user needs. Each feature is described along with its priority, purpose, and associated functional requirements. The aim is to provide a clear understanding of what the system will do and how it will be used by different user classes.

3.1 Feature 1: Complaint Management

3.1.1 Introduction

This feature allows residents to file road repair complaints, which are then entered into the system by clerks. The system tracks and manages these complaints, assigning them to the appropriate supervisor. This feature is of high priority as it initiates the entire repair process.

3.1.2 Inputs

1. Call logs of all the complaints registered with the branch office
2. Logs of all the written complaints filed at the branch office

3.1.3 Processing

- The system shall allow clerks to enter new complaints with details such as location, type of damage, and urgency.
- The system shall automatically assign complaints to supervisors based on area and workload.
- The system shall track the status of each complaint and provide updates to residents.
- The system shall generate area-wise complaint reports for supervisors daily.

3.1.4 Outputs

1. Updated complaint list
2. Status report of complaint list

3.2 *Feature 2: Supervisor Dashboard*

3.2.1 Introduction

The Supervisor Dashboard provides supervisors with tools to review and prioritize complaints, assess road conditions, and manage repair tasks. This feature is critical as it directly influences the effectiveness and timeliness of repairs.

3.2.2 Inputs

1. Complaint Details
2. Supervisor Inputs
3. Status update

3.2.3 Processing

- The system shall provide a dashboard for supervisors to view assigned complaints.
- The system shall allow supervisors to assess the severity of each complaint and set repair priorities.
- The system shall enable supervisors to estimate required resources for each repair task.
- The system shall track the progress of ongoing repairs and update the complaint status accordingly.

3.2.4 Outputs

1. Priority list of complaints assigned
2. Updated Repair Schedules

3.3 *Feature 3: Resource Estimation*

3.3.1 Introduction

This feature assists supervisors in estimating the materials, machinery, and manpower needed for each repair task. It is essential for accurate planning and resource allocation.

3.3.2 Inputs

1. Repair Details
 1. Locality type
 2. Urgency
2. Supervisor's Estimates

3.3.3 Processing

- The system shall allow supervisors to input estimates for raw materials, machinery, and personnel required for each repair.
- The system shall automatically update the resource inventory based on supervisor inputs.
- The system shall alert administrators when estimated resources exceed available inventory.

3.3.4 Outputs

1. Resource details
 1. Personnel
 2. Machinery
 3. Raw material Inventory
2. Resource Allocation
 1. Complaint details
 2. Resources that can be allocated
 3. Resources pending

3.4 *Feature 4: Automated Scheduling*

3.4.1 Introduction

Automated Scheduling prioritizes and schedules repair tasks based on complaint severity, resource availability, and locality type. It is a key feature for ensuring timely and efficient repairs.

3.4.2 Inputs

1. Repair details
 1. Type
 2. Locality
 3. Urgency
2. Resources
 1. Availability
 2. Current allocations

3.4.3 Processing

- The system shall schedule repair tasks automatically based on priority and resource availability.
- The system shall allow administrators to override schedules if necessary.
- The system shall update schedules in real-time when resource availability changes.
- The system shall notify supervisors of scheduled repairs and any changes to the schedule.

3.4.4 Outputs

1. Scheduled repair tasks
2. Updated inventory
 1. Resource details
 2. Resource allocation

3.5 *Feature 5: Resource Management*

3.5.1 Introduction

Resource Management ensures that the system accurately tracks the availability and status of all resources. It is vital for maintaining an up-to-date view of resource allocations and availability.

3.5.2 Inputs

1. Resource updates from UI

2. Resource updates from system

3.5.3 Processing

- The system shall track the status of all resources, including manpower, machinery, and materials.
- The system shall allow city corporation administrators to update the status of resources, such as marking machinery as out of service.
- The system shall reschedule repair tasks if a critical resource becomes unavailable.
- The system shall provide reports on resource utilization to help optimize resource allocation.

3.5.4 Outputs

1. Updated resource details
 1. Current inventory
 2. Allocation details
 3. Additional Requirements
2. Request for additional resource allocation based on requirements
3. Overview or Usage Reports for Resources

3.6 *Feature 6: Reporting and Analytics*

3.6.1 Introduction

Reporting and Analytics provide city officials with insights into repair operations, resource utilization, and outstanding tasks. This feature is important for informed decision-making and transparency.

3.6.2 Inputs

1. Report Generation request

3.6.3 Processing

- The system shall generate reports on the number and types of repairs carried out over a specified period.
- The system shall provide reports on outstanding repair tasks and their status.
- The system shall offer analytics on resource utilization, helping identify inefficiencies.
- The system shall allow administrators to generate custom reports based on specific criteria.

3.6.4 Outputs

1. Generated reports in various formats

3.7 *Feature 7: Administrator Control Panel*

3.7.1 Introduction

The Administrator Control Panel allows administrators to manage system configurations, user roles, and resource data. It is crucial for maintaining system operations and security.

3.7.2 Inputs

1. System configuration changes
2. User Management
 1. Role based access control
 2. Files and resources
 3. Module management

3.7.3 Processing

- The system shall provide a control panel for administrators to manage user accounts and roles.
- The system shall allow administrators to configure system settings and update resource data.
- The system shall enable administrators to monitor system performance and usage.
- The system shall log all administrative actions for auditing purposes.

3.7.4 Outputs

1. Updated system settings
2. User Management Records
 1. Users and roles
 2. Files and resources usage
 3. Modules in use

3.8 *Feature 8: User Authentication and Security*

3.8.1 Introduction

This feature ensures secure access to the system, protecting sensitive data and maintaining user accountability. It is essential for compliance with security policies and protecting city infrastructure.

3.8.2 Inputs

1. User inputs
 1. Username
 2. Password
 3. MFA token

3.8.3 Processing

- The system shall require users to authenticate via username and password and MFA tokens
- The system shall implement role-based access control, restricting access to sensitive features based on user roles.
- The system shall enforce password policies, including complexity requirements and expiration.
- The system shall log all user access and actions for security monitoring.

3.8.4 Outputs

1. User logs
 1. Time
 2. Actions
2. Outlier detection (secondary)

4 External Interface Requirements

This section describes the external interfaces that the RRTS will interact with, including user interfaces, hardware, software, and communication interfaces. It specifies how the system will connect and communicate with users, other software systems, and devices, ensuring seamless integration and interaction.

4.1 User Interfaces

- The system shall provide a web-based user interface accessible via major browsers (Chrome, Firefox, Edge).
- The interface shall be responsive, supporting both desktop and mobile devices.
- The system shall provide intuitive navigation, ensuring ease of use for all user classes.
- The UI shall support
 - Login page for all users
 - Username-password
 - MFA token
 - Data entry forms
 - For clerks – entering complaint data
 - For supervisors – reporting on complaints registered
 - For city corporation administrator – updating resources
 - Dashboards
 - For clerks – area-wise list of complaints entered
 - For supervisors – list of complaints assigned, repairs assigned, their status
 - For city corporation administrators – resource management
 - For mayor – overview of whole city's infrastructure
 - Report generation tools
 - For clerks – area-wise list of complaints registered
 - For supervisors – complaint investigation report, complaint and repair assignment
 - For city corporation administrators – resource allocation and utilization report
 - For mayor – overall report on repair works, resources city-wide

4.2 Hardware Interfaces

- The system shall interface with standard desktop computers, mobile phones and printers for generating reports.
- The system shall support integration with GPS devices for location-based tracking of complaints.

4.3 Software Interfaces

- The system shall integrate with the city's existing ERP system for resource management.
 - If ERP was done manually, it would be upgraded to digital mode
 - If ERP was done using rudimentary digital system, will be upgraded to online digital system with API integrations for communications
- The system shall interface with a relational DBMS (MySQL, PostgreSQL) for data storage.

- The system shall provide APIs for potential integration with third-party applications.

5 System Non-Functional Requirements

This section outlines the non-functional requirements that define the system's overall qualities and constraints. These requirements include performance, security, reliability, availability, maintainability, portability, and usability specifications. They help establish the standards that the system must meet to function effectively and efficiently.

5.1 Performance Requirements

- The system shall handle up to 1,000 simultaneous user sessions without performance degradation.
- The system shall process complaint entries within 2 seconds.
- The system shall generate reports within 10 seconds.

5.2 Security Requirements

- The system shall implement data encryption for all stored and transmitted data.
- The system shall include role-based access controls to protect sensitive data.
- The system shall enforce two-factor authentication for administrator access.

5.3 Reliability Requirements

- The system shall have a 98% uptime during operational hours.
- The system shall include failover mechanisms to minimize downtime during maintenance.

5.4 Availability Requirements

The system shall be available 24/7 with scheduled maintenance windows communicated in advance.

5.5 Maintainability Requirements

- The system shall be modular to allow updates and maintenance with minimal impact on operations.
- The system shall provide clear logs and error messages to assist in troubleshooting.

5.6 Portability Requirements

- The system shall be compatible with Windows, Linux, and MacOS environments.
- The system shall be deployable on cloud-based platforms as well as on-premises servers.

5.7 Usability Requirements

- The system shall be user-friendly, requiring minimal training for new users.
- The system shall provide help documentation and tooltips for key functions.

6 Other Non-Functional Requirements

This section addresses additional non-functional requirements that are crucial for the system's operation but do not fall under typical categories. It includes regulatory and compliance requirements as well as data integrity standards that the system must adhere to, ensuring that it meets legal, ethical, and operational norms.

6.1 Regulatory and Compliance Requirements

- The system shall comply with local data privacy regulations and IT security standards.
- The system shall adhere to accessibility standards to support users with disabilities.

6.2 Data Integrity Requirements

- The system shall ensure data accuracy and consistency across all modules.
- The system shall validate all user inputs to prevent data corruption.

7 Conclusion

The Road Repair and Tracking System (RRTS) outlined in this SRS aims to streamline the road maintenance process for the city, enhancing efficiency, transparency, and responsiveness to citizen complaints. By automating complaint management, resource estimation, scheduling, and reporting, the RRTS will significantly reduce manual workload and improve decision-making for the Public Works Department of the Corporation.

This document captures the functional and non-functional requirements necessary to build a robust system that addresses the critical needs of the city's road maintenance operations. Main features like *complaint tracking*, *supervisor dashboards*, *automated scheduling*, and *detailed analytics* will empower city officials to manage road repairs more effectively.

Moving forward, the focus will shift toward the design and implementation phases, where the detailed specifications outlined here will be translated into a working system.