SOFTWARE REQUIREMENTS SPECIFICATION

for

Road Repair and Tracking Software (RRTS)

Version 1.0

Prepared by:

Sumith Chandra Satkuri(22CS10067)
Rishith Kumar Kondi(22CS10037)
Nikhil Kumar Marati(22CS10042)

Team: Akatsuki

Contents

1 Introduction

1.1 Purpose

The purpose of this document is to define the requirements which the Road Repair and Tracking System (RRTS), must meet. The system is to be developed as an Online Portal for record keeping activities associated with the road repairing task of the Public Works Department of the Corporation of any large city. This document holds all the relevant information to start development.

1.2 Document Conventions

The following standards document conventions are used in our document: IEEE STD 830 -1998, IEEE Standard for Software Requirement Specification

1.3 Intended Audience and Reading Suggestions

This document is intended to assist the users when they use the software and for developers and the project managers to plan their project and implement the software required.
This Software Requirement Specification document is divided into Six subsections:
Section 1: Introduction
Section 2: Overall Description of the Software giving information about functions, user classes, operating environment, constraints and documentation.
Section 3: External Interface Requirements giving a brief introduction to user, hardware, software and communications interfaces.
Section 4: Detailed functional requirements of different features.
Section 5: List of non-functional requirements Section
6: Other requirements

1.4 Project Scope

The scope of the Road Repair and Tracking Software (RRTS) project includes the development of a comprehensive software system to automate various bookkeeping activities associated with road repair tasks of a city's Public Works Department. The software will facilitate the recording, management, checking the resource requirements and scheduling of repair requests according to the prioritization of repairs based on severity and location characteristics, and generation of reports for stakeholders. The

primary objective is to streamline and optimize the road repair process for improved efficiency and effectiveness.

1.5 References

The documents and websites referred to, are as follows:

- IEEE STD 830 -1998, IEEE Standard for Software Requirement Specification.

2 Overall Description

2.1 Product Perspective

Currently, the manual handling of repair requests, scheduling, resource allocation, and reporting is time-consuming, error-prone, and lacks efficiency. Automating these tasks will improve response times, enhance resource utilization, and provide better oversight and control over road repair activities, ultimately leading to improved infrastructure maintenance and citizen satisfaction.

This project is needed to streamline and automate the process of road repair management within a city corporation. The project addresses the challenge of coordinating repair efforts across different city suburbs, ensuring that repairs are prioritized based on severity and locality type. By providing supervisors with real-time information on repair requests, road conditions, and available resources, the system enables them to make informed decisions and optimize repair schedules. Additionally, the project facilitates data-driven decision-making at the administrative level by providing comprehensive reports and statistics on repair activities, resource utilization, and outstanding repair work. Overall, the project aims to modernize and optimize the road repair process, leading to better infrastructure management and service delivery for city residents.

2.2 Product Functions

The following functions are provided by the software

- Login by different entities- Clerk, Supervisor, Administrator.
- The Clerks can add the complaints raised by the people which will be directed to the Supervisor.
- The Supervisor can add the data form for a complaint about the required man power and resources and the priority.
- The Administrator can see the statistics of the works ongoing, pending and completed and send it to mayor upon requesting.
- The Administrator can change the available resources if any machine goes down or something else.
- Keeps track of all the information about road repairs.

2.3 User Classes and Characteristics

- **Clerks**:
  - Responsible for entering repair requests into the system received from residents either over the phone or through written complaints.
  - They are typically stationed at branch offices located in different suburbs of the city.
  - Characteristics include data entry proficiency, familiarity with the road repair process, and the ability to communicate effectively with residents.
- **Supervisors**:
  - Assigned to specific areas within the city to examine repair requests, assess road conditions, and prioritize repair work based on severity and locality type.
  - They estimate resource requirements, such as raw materials, machinery, and personnel, for each repair task.
  - Characteristics include field expertise in road maintenance, decision-making skills, and effective communication with both clerks and repair crews.
- **Administrators**:
  - Responsible for managing and updating data related to available manpower and machinery.
  - They ensure that the system reflects accurate information regarding resources that are currently available and those already committed.
  - Characteristics include proficiency in data management, organizational skills, and the ability to respond promptly to changes in resource availability.
  - Administrator also has an option to generate the work schedule for the day.

2.4 Operating Environment

The software is an online based portal and is thus platform independent and can be used in all well-known browsers.

2.5 Design and Implementation Constraints

- Limited Storage provided in the Mongo Cloud Database.

2.6 Assumptions and Dependencies

- The user must be familiar with the internet and capable to connect to the platform with a web browser.
- No accessibility features are provided on the platform assuming that the disabled users' device provide those if any such users use this platform.
- Connecting and using the platform requires:
  - working internet connection.
  - compatible browser

3 External Interface Requirements

3.1 User Interfaces

The product will consist of a web-application. The Clerks, Supervisors and administrators have different dashboards based on their functionalities and will be accessible by using their login credentials leading them to their respective user dashboards. As the number of suburbs is fixed and so is the supervisors, clerks the accounts are already made and given to them and they can change their credentials after logging in. Also Supervisors can add Clerk account if needed.

- Homescreen – Login
    - UserID
    - Password
- Clerk
    - Edit Profile
    - View the status of complaints – Add Complaint
        - Can delete “new” complaints
- Supervisor
    - Edit Profile
    - New Complaints
        - Make Data Form of needed resources – Work Schedule
        - Option to mark it completed
    - Completed Works
    - Add Clerk account
- Administrator
    - Edit Profile
    - Update the available resources
    - Statistics
    - Add Supervisor account

3.2 Hardware Interfaces

The hardware interfaces for the online portal are listed below:

- The portal can be used on any platform or PC which has a proper internet connection.
- Compatible web browser is required to avoid any problems.

3.3 Software Interfaces

The Software interfaces for the project are listed below:

- The Project will connect to React for Front-End Responsive Pages and MongoDB Cloud Database.
- The project is a web app and thus requires a working internet connection and a web browser only and is operating system independent.
- The Project uses Express JS Framework in the server side and React Javascript (React JS). All the data about complaints and works will be stored in an online MongoDB Cloud Database.

3.4 Communications Interfaces

All the communications will be done via the web-browser with the standard HTTP protocol.

4 System Features

This subsection provides details about the identified functional requirements of the software.

4.1 Register Complaint

4.1.1 Description:

- Record the complaint by the clerk in the office, which are raised by the residents, by clearly mentioning the name of the road, exact location where the problem has occurred and a brief description of the problem.

4.1.2 Stimulus/Response Sequences:

- Clerk can now see the complaint with status new in his home page and he has the option to delete the complaint until supervisor checks the complaint (As supervisor checks the place and if the complaint is false, he can discard it).

4.1.3 Functional Requirements:

- Req-1: Login: The clerk will enter their user ID and password, which are created by the officer higher up in the hierarchy (i.e., the Suburb Supervisor). The user ID will be the employee ID, and the password can be reset by the clerk after logging in.
- Req-2: Allocation of space in the server memory for the complaint.

4.2 Update Required Resources Data for the Registered Complaint

4.2.1 Description:

- The supervisor is responsible for updating the data related to registered complaints by adding attributes indicating the priority of the repair work and the estimated resources and time required. This encompasses various categories of manpower, machines, and raw materials necessary for carrying out the repair work. Additionally, the supervisor has the authority to discard the complaint if it is found to be false upon inspection of the location.

4.2.2 Stimulus/Response Sequence:

- After the Supervisor enters the data, the complaint will no longer appear on the New Complaints page. Instead, its status will be updated to 'pending,' and from next work-schedule onwards this complaint will be considered for allocation of resources.

4.3 Functional Requirements:

- Req-1: Login: The Supervisor will enter their user ID and password, which are created by the officer higher up in the hierarchy (i.e., the Corporate Administrator). The user ID will be the employee ID, and the password can be reset by the Supervisor after logging in.

- Req-2: Location Visit and Data Entry: The Supervisor must visit the location of the complaint and enter the relevant details into the system.

4.3 Update Resources

4.3.1 Description:

- Update the resources availability data at any time by the administrator.

4.3.2 Stimulus/Response Sequence:

- Upon the administrator updating the resources, the work schedule will be automatically updated to match the newly available resources.

4.3.3 Functional Requirements

- Req-1: Login: The Administrator will enter their user ID and password, which are provided to them beforehand by the system administrator. The password can be reset by the Administrator after logging in.
- Req-2: Resource Update: The Administrator should update the resources in the system whenever a machine malfunctions or due to any other unforeseen event, and then click on the 'Update Work Schedule' button.

4.4 Schedule Repair

4.4.1 Description:

- Process all the complaints from various sub-urbs for which the Supervisor has entered the data and sort the complaints based on the following criteria:
    - Priority of the work
    - Estimated time entered by Supervisor - Resources needed for the repair.

4.4.2 Stimulus/Response Sequence:

- After sorting the complaints based on the above-specified criteria, allocate resources to them. Then, Supervisors can view the ongoing works of their respective suburbs, while the Administrator can monitor all ongoing works in the city.
- The status of the complaint will be updated to “ongoing”.

4.4.3 Functional Requirements:

- Req-1: Login: The Supervisor should login to see the ongoing works in his sub-urb.
- Req-2: The Administrator is responsible for updating the work schedule daily by clicking the 'Update Work Schedule' button.

4.5 Generate Statistics

4.5.1 Description:

- Retrieve and generate statistics for completed works based on priority and the suburb. Additionally, provide statistics on the resources utilized for completed works.

4.5.2 Stimulus/Response Sequence:

- The Administrator must log in to generate statistics.

4.5.3 Functional Requirements:

- Data is retrieved from the database to obtain real-time statistics.

5 Other Nonfunctional Requirements

5.1 Performance Requirements

The user should be able to query databases quickly and the results fetched must be appropriate. This can be done by finding the right balance between performance and accuracy by using MongoDB.

5.2 Safety Requirements

The application runs on the web browser and hence harm to the user device is minimal, whereas a lot of data is to be read and written and hence the data storage and the server damage is possible but that too during heavy usage.

5.3 Software Quality Attributes

- Availability: Road Repair and Tracking Software will be available 24 hours a day, 7 days a week. Correctness: Road Repair and Tracking Software shall provide real time data about the repairs going on around the city.

- Correctness: Road Repair and Tracking Software shall provide real time data about the repairs going on around the city.

- Maintainability: Various versions of the product should be easy to maintain. For development it should be easy to add new code to the system, modify the existing code or incorporate new features to the software.

- Robustness: The software shall be robust enough to have a high degree of fault tolerance. For example, the system should not crash if the user enters an invalid input. It should display a suitable error message.

- Usability: Road Repair and Tracking Software shall provide an easy way to use graphical interface similar to existing complaint lodging portals so that users don't need to adapt to new interface features. The interface should be interactive and easily navigable. Users should be able to understand the menu and other features of the software. Any error message displayed by the software shall be polite and clear.

5.4 Business Rules

The software should not be outsourced to any third party without prior permission. Maintaining more than one account or fake account is illegal.

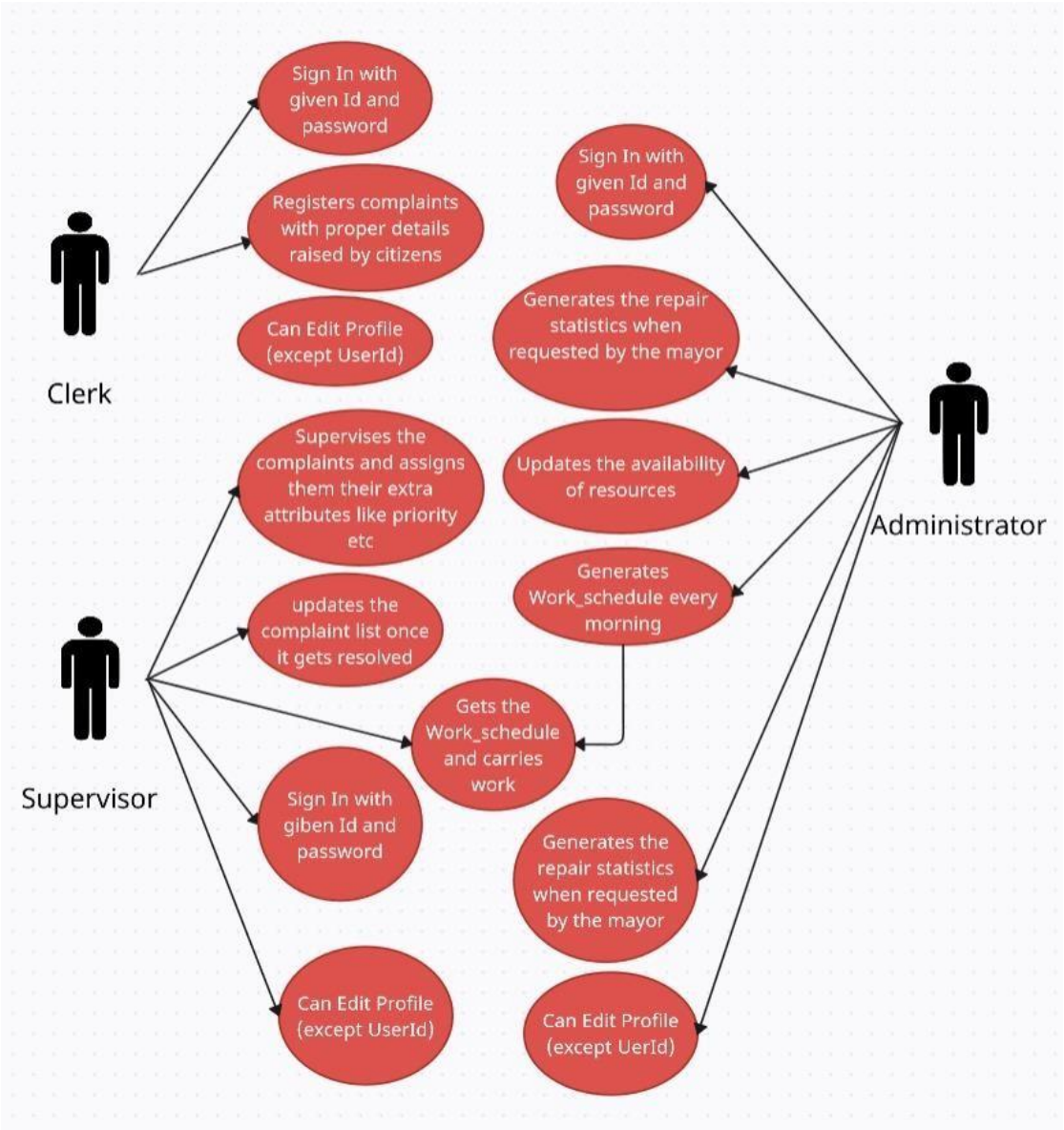
6 Other Requirements

- Securely store the data and backup server to prevent any server crashes.
- A server capable of handling heavy workload and queries.
- Licensing requirements: Applicable.

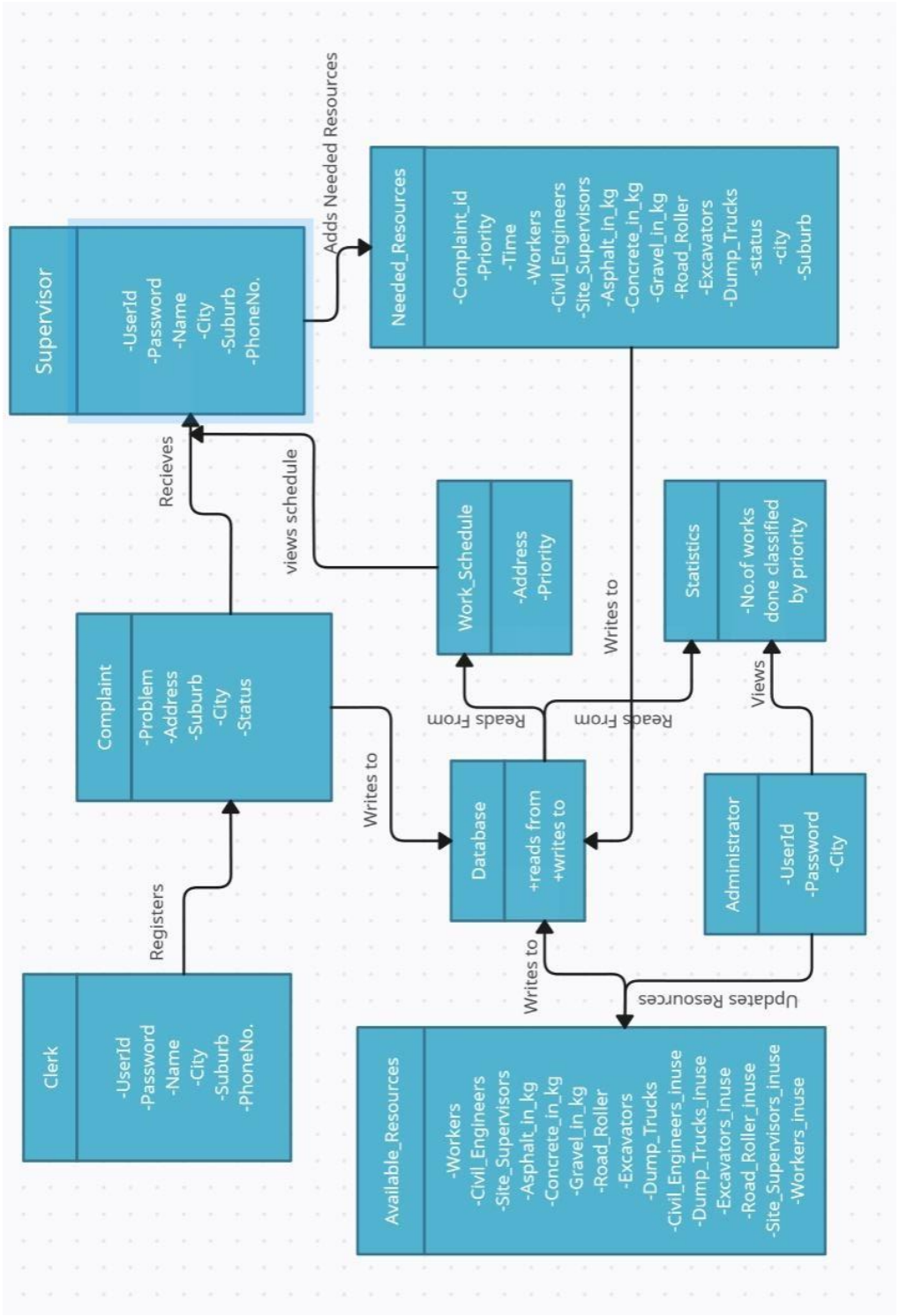
6.1 Appendix A: Glossary

- JavaScript: A Modern Multipurpose programming language.
- Express.js: A JavaScript back-end framework for web development.
- "ReactJS: Leveraging JSX (JavaScript XML) to seamlessly integrate JavaScript logic into HTML-like templates for building dynamic user interfaces."
- MongoDB: Widely used NoSQL Online Cloud Database.

6.2 Appendix B: Analysis Models



Use Case Diagram



Class Diagram