Software Requirements Specification

for

Smart Dhaka – A Smart City Management System

July 2025

Md. Sabbir Hossain Tamim,

Tanjim Salam,

MD. Fazla Rabbi,

Md. Sabbir Hoshen Khan

Contents

Introduction

1.1 Purpose

This document outlines the software requirements for the Smart City Management System (SCMS), version 1.0. The SCMS is a web-based platform designed to streamline urban services for citizens and administrators in a smart city environment. It provides functionalities such as user registration, secure login with OTP verification, profile management, traffic violation tracking, utility bill payments, transport route management, waste collection scheduling, complaint filing, emergency service requests, event management, and educational institute information. The system aims to enhance citizen convenience, improve administrative efficiency, and promote sustainable urban management. This SRS provides a comprehensive overview of the system's features, performance expectations, constraints, and interfaces required for development, implementation, and maintenance.

1.2 Intended Audience

This SRS is designed for the following stakeholders:

- **Administrators**: Manage system operations, user accounts, and service requests.
- **Citizens (Users)**: Access services like bill payments, violation tracking, complaint filing, and event information.
- **Business Analysts (BAs)**: Define business requirements and ensure alignment with user needs.
- **Project Managers (PMs)**: Oversee the development process and ensure project milestones are met.
- **Developers**: Implement system functionalities based on specified requirements.
- **QA/QC Engineers**: Ensure the system meets quality and performance standards.
- **Testers**: Validate system functionality through test scenarios.
- **Investors**: Evaluate the project's scope and potential for funding decisions.

1.3 Intended Use

Each stakeholder group uses this SRS to understand and contribute to the SCMS development:

- **Administrators**: Understand system management tasks and deployment considerations.
- **Citizens**: Interact with the system through user-friendly interfaces for various services.
- **Business Analysts**: Translate business needs into functional requirements.
- **Project Managers**: Align development with user expectations and project scope.

- **Developers**: Use functional requirements to code system features.
- **QA/QC Engineers**: Develop test cases to ensure system reliability.
- **Testers**: Validate system behavior against specified requirements.
- **Investors**: Assess project scope and potential returns.

1.4 Product Scope

The SCMS is a centralized platform integrating urban services to enhance citizen experience and administrative efficiency. It allows citizens to manage profiles, pay utility bills, track traffic violations, view transport routes, schedule waste collection, file complaints, request emergency services, view events, and access educational institute information. Administrators manage user accounts, update service records, and ensure system integrity.

1.4.1 Purpose

The system simplifies access to city services, reduces administrative overhead, and pro- motes transparency and efficiency in urban management.

1.4.2 Benefits and Objectives

- Enhance citizen convenience through a unified platform for city services.
- Improve administrative efficiency with automated CRUD operations.
- Promote transparency by providing real-time access to service statuses.
- Support sustainable urban management through efficient resource allocation.

1.4.3 Alignment with Corporate Goals

The SCMS aligns with goals of leveraging technology for urban innovation, improving citizen satisfaction, and fostering sustainable city operations.

1.4.4 Relating to Business Strategies

By providing a digital platform for city services, the SCMS positions the city as a leader in smart urban management, encouraging citizen engagement and operational efficiency.

1.5 Risk Definition

- **User Inactivity**: Risk that citizens may not actively use the platform, impacting service adoption.
- **Administrator Workload**: Risk of high workload for administrators managing large volumes of data.
- **Security Vulnerabilities**: Risk of data breaches affecting user information and transactions.

- **System Downtime**: Risk of technical issues disrupting service access.
- **Changing Requirements**: Risk that evolving stakeholder needs may require SRS updates.

2 Overall Description

2.1 User Classes and Characteristics

The SCMS caters to two primary user classes: Citizens and Administrators.

2.1.1 User Class: Citizens

- **Characteristics**:
  - Access the system to manage profiles, pay bills, track violations, view schedules, file complaints, request emergency services, and view events and institutes.
  - Update personal information and upload profile pictures.
  - Perform actions like paying bills or fines and filing complaints.
  - Require a user-friendly, responsive interface for seamless interaction.

2.1.2 User Class: Administrators

- **Characteristics**:
  - Manage user accounts and verify registrations.
  - Perform CRUD operations on services like traffic violations, utility bills, transport routes, waste schedules, complaints, emergency requests, events, and institutes.
  - Ensure system security and compliance with policies.
  - Handle user-reported issues and maintain system integrity.

2.2 User Needs

2.2.1 Citizens

- **Profile Management**: Update personal details and upload profile pictures.
- **Service Access**: View and manage utility bills, traffic violations, transport routes, waste schedules, complaints, emergency requests, events, and institutes.
- **Payment Processing**: Securely pay bills and fines.
- **Complaint Filing**: Submit and track complaints.
- **Emergency Requests**: Request and track emergency services.
- **Mobile Responsiveness**: Access the system on various devices.

2.2.2 Administrators

- **User Management**: Verify and manage citizen accounts.
- **Service Management**: Add, update, or delete records for city services.
- **Security Management**: Implement measures to protect user data.
- **Reporting**: Access logs and summaries for auditing purposes.

2.3 Operating Environment

2.3.1 Recommended Hardware Platform

- Desktops: Intel Core i5, 8GB RAM, 256GB SSD.
- Laptops: Intel Core i3, 4GB RAM, 128GB SSD.
- Mobile Devices: iOS 13+ (iPhone 6S+), Android 9.0+.

2.3.2 Recommended Operating System and Versions

- Windows: Windows 10/11 (64-bit).
- macOS: Catalina (10.15) +.
- Linux: Ubuntu 20.04 LTS+, Fedora 33+.
- Mobile: iOS 13+, Android 9.0+.

2.3.3 Software Components and Applications

- Browsers: Google Chrome, Mozilla Firefox, Microsoft Edge, Safari (latest stable versions).
- Server: Apache or Nginx with PHP 8.2+.
- Database: MariaDB 10.4+.

2.3.4 Database Compatibility

- MariaDB 10.4+ for storing user and service data (e.g., citizens, complaints, utilitybills).

2.3.5 Interoperability

- Uses RESTful APIs for potential third-party integrations.
- Supports JSON for data interchange.

2.3.6 Network Requirements

- Stable internet connection with a minimum 5 Mbps bandwidth.
- Supports wired (Ethernet) and wireless (Wi-Fi) connections.

2.3.7 Security Considerations

- TLS encryption for secure data transmission.
- Compatibility with standard firewalls and security software.

2.4 Constraints

- **Technical**: Selection of secure frameworks for authentication and payment processing.
- **Time**: Development must meet project milestones and launch deadlines.
- **Budget**: Resource allocation for development, hosting, and maintenance.
- **Regulatory**: Compliance with data protection laws (e.g., GDPR if applicable).
- **Resource**: Availability of skilled developers proficient in PHP, HTML, CSS, and MariaDB.

2.5 Assumptions

- Citizens will actively engage with the platform for service access.
- Administrators have the skills to manage system operations.
- Users have basic proficiency with web-based applications.
- Stakeholders have access to and can comprehend the SRS.
- The system will have consistent internet connectivity for users.

3 Requirements

3.1 Functional Requirements

3.1.1 User Registration

As a citizen, I want to register with my details to access the system.
Success:

- User submits a form with full name, email, password, address, and phone number.
- System stores data in the citizens table with a unique Citizen ID and hashed password.
- User is redirected to the login page.

Failure:

- Duplicate email results in an error message.
- Invalid inputs prompt correction and resubmission.

3.1.2 User Login with OTP

As a citizen, I want to log in securely using my email, password, and OTP.
Success:

- Valid credentials trigger an OTP stored in the Database

Failure:

- Invalid credentials or OTP display an error message.
- Expired OTP prompts resending or re-entry.

3.1.3 Profile Management

As a citizen, I want to update my profile details and upload a profile picture.
Success:

- User updates full name, address, phone, or password in the citizen option.
- Profile picture (JPG, PNG, GIF, <2MB) is uploaded and linked in the Profile Picture module.
- System confirms successful update with a success message.

Failure:

- Invalid file type/size displays an error message.
- System prompts user to correct input and retry.

3.1.4 Traffic Violation Management

As a citizen, I want to view and pay my traffic violations.
As an admin, I want to manage violation records.
Success:

- Citizens view violations which they made.
- Admins add, update, or delete records in the traffic violation module.

Failure:

- Unauthorized access redirects to the login page.
- Invalid inputs in admin forms prompt error messages.

3.1.5 Utility Bill Management

As a citizen, I want to view and pay my utility bills.
As an admin, I want to manage bill records.
Success:

- Citizens can view bills from the utility bills functionality module.
- Citizens pay unpaid bills, updating Status to Paid.

- Admins add, update, or delete records in the utility bills section.

Failure:

- Unauthorized access redirects to the login page.
- Invalid inputs in admin forms prompt error messages.

3.1.6 Transport Route Management

As a citizen, I want to view transport routes.

As an admin, I want to manage route records.
Success:

- Citizens view routes from the transport table (Transport ID, Route Name, Vehicle- Type, Timing, Status).
- Admins add, update, or delete records in the transport table.

Failure:

- Unauthorized access redirects to the login page.
- Invalid inputs in admin forms prompt error messages.

3.1.7 Waste Schedule Management

As a citizen, I want to view waste collection schedules.
As an admin, I want to manage schedule records.
Success:

- Citizens view schedules from the waste management table (Waste ID, Area, Schedule, Status).
- Admins add, update, or delete records in the waste management table.

Failure:

- Unauthorized access redirects to the login page.
- Invalid inputs in admin forms prompt error messages.

3.1.8 Complaint Management

As a citizen, I want to file and track complaints.

As an admin, I want to manage complaint records.
Success:

- Citizens submit complaints with subject and description, stored in the database (Complaint ID, Citizen ID, Subject, Description, Status, Date Filed).
- Citizens view their complaints and their statuses (Pending, In Progress, Resolved).
- Admins update complaint statuses or delete records.

Failure:

- Unauthorized access redirects to the login page.
- Invalid inputs in complaint forms prompt error messages.

3.1.9 Emergency Service Requests

As a citizen, I want to request and track emergency services. **As an admin**, I want to manage emergency request records. **Success**:

- Citizens submit emergency requests with type and description, stored in the database with Request ID, Citizen ID, Type, Description, Date Requested, Status.
- Citizens view their requests and their statuses (Pending, Responded).
- Admins update request statuses or delete records.

Failure:

- Unauthorized access redirects to the login page.
- Invalid inputs in request forms prompt error messages.

3.1.10 Event Management

As a citizen, I want to view upcoming and completed events.
As an admin, I want to manage event records.
Success:

- Citizens can view events which will be held in Dhaka city.
- Admins add, update, or delete records of the events.

Failure:

- Unauthorized access redirects to the login page.
- Invalid inputs in event forms prompt error messages.

3.1.11 Educational Institute Information

As a citizen, I want to view information about educational institutes.
As an admin, I want to manage institute records.
Success:

- Citizens can view educational institutes in the Smart Dhaka City.

Failure:

- Unauthorized access redirects to the login page.
- Invalid inputs in institute forms prompt error messages.

3.2 Non-Functional Requirements

3.2.1 Performance Requirements

- Response Time: System responds to user actions within 2 seconds under normal conditions.
- Scalability: Handles 20% increase in concurrent users during peak times.

3.2.2 Security Requirements

- User Authentication: Implements secure login with OTP verification and password hashing (bcrypt).
- Data Encryption: Uses TLS for data transmission and secure storage of sensitive data.

3.2.3 Software Quality Attributes

- Usability: Intuitive interface with 90% user satisfaction in feedback surveys.
- Reliability: 99.9% uptime for continuous access.

3.2.4 Business Rules

- Only registered users with verified accounts can access services.
- Admins must approve changes to critical records (e.g., violations, bills, complaints).
- OTPs expire within a set timeframe (e.g., 5 minutes) for security.