

Software Requirements Specification (SRS)

Smart QR Code Generator & Scanner Web Application

1. Introduction

1.1 Purpose

The purpose of this Software Requirements Specification (SRS) document is to comprehensively define the functional, non-functional, and system requirements for the Smart QR Code Generator & Scanner Web Application. This document will serve as a reference for the development team, project supervisor, and stakeholders to ensure clarity and accuracy throughout the software development lifecycle.

1.2 Document Conventions

- Standard IEEE SRS formatting has been followed.
- Functional requirements are labeled as FR-x.
- Non-functional requirements are labeled as NFR-x.

1.3 Intended Audience and Reading Suggestions

- Students/Developers: To understand system design and implementation requirements.
- Supervisor (Ms. Sara Farid): To review and verify project scope and completeness.
- University Evaluation Committee: To analyze project complexity and feasibility.
- End Users: For understanding how the system will perform.

1.4 Project Scope

The Smart QR Code Generator & Scanner Web Application allows users to generate customized QR codes, scan QR codes using any device camera, store scan history, and manage QR content online. It supports multiple QR formats, ensuring secure, fast, and user-friendly code processing.

1.5 Definitions, Acronyms, and Abbreviations

QR Code: Quick Response Code

UI/UX: User Interface / User Experience

API: Application Programming Interface

DBMS: Database Management System

CRUD: Create, Read, Update, Delete

2. Overall Description

2.1 Product Perspective

This is a standalone web-based system developed using modern web technologies. It interacts with:

- A web server for hosting
- A backend server for QR processing and user authentication
- A database for storing user data and scan history
- Device camera APIs for QR scanning

2.2 Product Functions

Major features of the system include:

- Generate QR Codes (URL, text, email, phone, Wi-Fi, contact info, file storage)
- Scan QR Codes through device camera or uploaded images
- Download & Share QR Codes in PNG/SVG formats
- Customization (colors, logo, styles, templates)
- User Authentication (login, registration, password reset)
- User Dashboard showing generated and scanned history
- Admin Panel for managing users & system activity

2.3 User Classes and Characteristics

- General Users: Create and scan QR codes.
- Registered Users: Save history, manage data, customize QR codes.
- Admin: Manage system operations, monitor users, handle backups.

2.4 Operating Environment

- Frontend: HTML5, CSS3, JavaScript, Bootstrap/Tailwind
- Backend: Node.js / PHP / Python

- Database: MySQL / MongoDB
- Browsers: Chrome, Firefox, Safari, Edge
- Hardware: Any device with camera

2.5 Design and Implementation Constraints

- Must comply with privacy rules regarding stored QR data.
- Limited camera access due to browser permissions.
- Internet connection required for full functionality.

2.6 Assumptions and Dependencies

- Users have a working camera for scanning.
- QR libraries function properly.
- Server hosting is available.

3. System Features

3.1 QR Code Generation (FR-1)

- Users can generate QR codes by selecting content type.
- System generates high-quality QR codes.
- Users can download QR codes in PNG or SVG.

3.2 QR Code Scanning (FR-2)

- System allows live camera scanning.
- System decodes QR from uploaded images.

3.3 QR Code Customization (FR-3)

- Users can change colors, add logos, and preview QR codes.

3.4 User Authentication (FR-4)

- Register, login, reset password, logout.

3.5 User Dashboard (FR-5)

- View history of generated and scanned QR codes.

3.6 Admin Panel (FR-6)

- Manage users, logs, backups, and system statistics.

4. External Interface Requirements

- User interfaces include dashboard, scanner screen, admin panel.
- Uses browser camera API and QR libraries.
- Communicates via HTTPS and REST APIs.

5. Non-Functional Requirements

- Performance: QR generation <1 sec.
- Security: Encrypted passwords, HTTPS.
- Usability: Responsive UI, tooltips.
- Reliability: Handles 500+ users.

6. Other Requirements

- Weekly data backup.
- Logs retained for admin.

7. Conclusion

This SRS defines complete requirements for the Smart QR Code Generator & Scanner Web Application under the supervision of Ms. Sara Farid.