Software Requirements Specification
for
RFID Ticketing System

Peraverse: Crowd Management Software

Version 2.0

Prepared by: RFID and Ticketing System Development Team

Department of Computer Engineering,
Faculty of Engineering,
University of Peradeniya

Date: May 30, 2025

Table Of Contents

1. Introduction

This Software Requirements Specification (SRS) document outlines the requirements for the **RFID Ticketing System**, a critical component of the crowd management software (**Peraverse**) developed for the 75th anniversary exhibition of the Faculty of Engineering at the University of Peradeniya, Sri Lanka. The RFID Ticketing System leverages Ultra-High Frequency (UHF) RFID technology to enable seamless visitor entry, access control, and behavior-driven analytics for an estimated 10,000 concurrent attendees over a three-day event.

1.1. Purpose

The purpose of this SRS is to specify the functional and non-functional requirements for the RFID Ticketing System, which includes developing RFID tracking hardware modules, integrating these modules with a MySQL database, and creating the backend and frontend for a visitor registration portal. The system aims to provide secure, efficient ticketing and access control, prevent ticket fraud, and collect data for analytics (e.g., booth visit durations).

This document ensures that the development team, consisting of second-year undergraduate students, delivers a robust solution within the project's budget and timeline (3 months, targeting early September 2025), the system complies with Sri Lanka's Personal Data Protection Act (2022).

1.2. Document Conventions

This SRS adheres to the IEEE Software Requirements Specification template with the following conventions:

- **Shall:** Desirable but non-mandatory feature.
- **Should**: Mandatory requirement.
- **Will**: Assumed or planned system behavior.
- Acronyms (e.g., RFID: Radio Frequency Identification) are defined on first use.
- Requirements are numbered (e.g., FR-T1 for Functional Requirement, Ticketing 1).
- Technical terms are explained for non-technical stakeholders.

1.3. Intended Audience and Reading Suggestions

This document is intended for all stakeholders involved in the design, development, deployment and usage of the Peraverse. The primary audiences include:

- **Development Team**: Second-year students. Focus on Section 3 (Requirements) and Section 4 (System Features).
- **Project Supervisors**: Faculty advisors. Review Section 1 (Introduction) and Section 2 (Scope).
- **Event Organizers**: Faculty and volunteers. Read Section 2 (Scope) and Section 3.1 (Functional Requirements).
- **Administrators** and **Sponsors**: Stakeholders. Focus on Section 1 (Introduction) and Section 5 (Constraints).
- **Client/Stakeholders**: Review system features to ensure alignment with business needs.

- **Testers**: Responsible for validating system functionality. Focus on **Section 4 (System Features)** for functional requirements and **Section 6.1 (Testing)** for testing protocols and procedures.

Start with Section 1 (Introduction) and Section 2 (Scope) for context. Technical readers should prioritize numbered requirements; non-technical readers should focus on narrative descriptions.

Reader's Need	Recommended Section	Purpose
Unfamiliar with the project	Section 1: Introduction	Provides background and context for the system
Looking for a high-level system overview	Section 2: Overall Description	Summarizes what the system will do
Interested in detailed system functionality	Section 4: System Features	Defines all functional requirements in detail
Focused on user interfaces and integration points	Section 3: External Interface Requirements	Describes UI, hardware, software, and communication interfaces
Concerned with nonfunctional aspects (e.g., performance, security)	Section 5: Nonfunctional Requirements	Covers quality attributes and system constraints

1.4. Product Scope

The RFID Ticketing System is a subsystem of the Peraverse crowd management software, designed to manage visitor entry and tracking for the 75th anniversary exhibition. It uses 12,000 UHF RFID wristbands and 10–20 RFID readers to enable:

- Fast ticket validation and access control with QR code backup.
- Minimal data collection (first name/initials, mobile number, age range, gender, ticket type).
- Registration methods
  - Pre-registration (online) and on-site registration (mobile/kiosk) with RFID/QR wristbands.
  - On-Site Registration (Mobile/Kiosk)
- Ticket Distribution
  - **Pre-registered visitors** show their QR codes at the gate to receive RFID wristbands.
  - **On-site visitors** are issued wristbands immediately after registration.
  - Each wristband contains a **unique identifier** (RFID or QR), securely mapped to the visitor’s data.
- Group tracking to identify lost individuals.
  - In cases where groups (such as school students) arrive together, the system supports **group registration under a single group identifier** (e.g., the teacher-in-charge’s registration).

        - Although each individual within the group has a unique ID, the system **recognizes the set of unique IDs as a single group entity**. This enables efficient admission without registering each person separately.

- Seamless ticket validation at entry gates and restricted zones (e.g., VIP areas).
- Fraud prevention through unique RFID/QR IDs.Analytics on attendee behavior (e.g., entry/exit times, booth visits).
- Integration with a registration portal for assigning RFID tags to visitors.
- **Multilingual interface support** to ensure accessibility for all visitors, including international attendees, reinforcing inclusivity and ease of use.

The system shall support 10,000 concurrent users with a maximum 3-second response time per ticket scan, comply with Sri Lanka's Personal Data Protection Act (2022), and operate within a budget.

2. Overall Description

This section describes the RFID Ticketing System's context, components, and constraints within the Peraverse software.

2.1. Product Perspective

The RFID Ticketing System is the entry point for Peraverse, replacing manual ticketing with passive RFID wristbands and QR code backups. It provides modular APIs for integration with tracking, analytics, and engagement systems. Components include:

1. **RFID Hardware**: 10–20 UHF RFID readers and 12,000 passive wristbands.
2. **Registration Portal**: React (frontend) and Node.js (backend) for pre-registration, on-site registration, batch management, and RFID/QR assignment.
3. **Database**: MySQL for ticket, attendee, batch, group, and lifecycle data.
4. **Interfaces**: Connects to university payment gateway and parent system's backend.

It operates in a 50,000 sq. ft. indoor/outdoor venue with Wi-Fi (802.11ac) and 4G backup.

2.2. Product Functions

1. **Ticket Validation**: Validates wristbands at 10–20 gates with ≤ 1% error rate.
2. **Access Control**: Restricts zone access based on RFID permissions.
3. **Visitor Registration**: Collects name, optional contact, organization, and group type.
4. **Batch Registration**:
    - School students (100 per batch) under a teacher with proximity alarms.
    - Friends (15 per batch) with location sharing on maps.
    - (25 per batch) with lead person tracking and child proximity alarms.
5. **RFID Lifecycle Management**: Assigns, deactivates, and recycles tags.
6. **Analytics Logging**: Tracks entry/exit, booth visits, and group data.
7. **Fraud Prevention**: Uses unique RFID IDs. All functions support a 3-second scan response time and 10,000 concurrent users.

2.3. User Classes and Characteristics

1. **Visitors**: Visitors are individuals attending the event. They interact with the system primarily through the portal and RFID wristbands. Subtypes vary by group behavior and monitoring needs.
    - **General Public**:
        - Description: Individuals or small groups with diverse backgrounds.
        - Characteristics: Vary in technical proficiency.
        - System usage: Register, scan wristbands, access event zones.
    - **School Students**:
        - Description: Students attending in batches under teacher supervision.
        - Characteristics: Typically minors; arrive in groups of less than 100 per teacher.
        - System usage: Proximity tracking; batch registration.

- **Friends**:
  - Description: Informal groups of up to 10 people.
  - Characteristics: Social use; interested in sharing locations..
  - System usage: Share real-time location via map interface.
- **Families**:
  - Description: Groups of up to 15 with one adult lead.
  - Characteristics: Includes children requiring extra safety.
  - System usage: Child tracking, proximity alarms, lead batch controls.
- **Group Types**:
  - Values: School student, Teacher, University Student, Company Representative, General Public.
  - Usage: Used to define batch permissions and monitoring scope.

2. **Event Staff**:
   - Description: On-site personnel handling visitors and system logistics.
   - Characteristics: Moderately tech-savvy, 1-hour training.
   - Usage: Register visitors / manage batches, recycle tags, view real-time visitor activity.
3. **System Developers (Administrators)**:
   - Description: Second-year students responsible for implementing and maintaining the system.
   - Characteristics: Proficient in hardware, software, and systems integration.
   - Usage: Configure and maintain hardware, database, portal; monitor lifecycle events and system logs.

2.4. Operating Environment

- **Venue**: 50,000 sq. ft. mixed indoor/outdoor area located at the University of Peradeniya. The environment includes open zones, halls, and outdoor paths, requiring robust connectivity and weather-resistant hardware.
- **Hardware**:
  - RFID Readers: 10–20 UHF RFID readers, IP65-rated (weather-resistant), 10-meter range.
  - Tags: IP65-rated (weather-resistant), 10-meter range.
  - Servers: Local servers deployed on-site for data processing, redundancy, and offline support.
- **Network**: Wi-Fi (100 Mbps, WPA3 encryption, 80% coverage approx.), 4G backup (up to 20 Mbps).
- **Client Devices**:
  - Visitor Portal: Accessible via mobile phones, tablets, and desktop browsers - Google Chrome, Mozilla Firefox, Apple Safari (latest stable versions).
  - Staff Interfaces: Accessed via tablets and desktop systems (devices connected via Wi-Fi or LAN depending on station).

- **Environmental Factors**: Average conditions(32°C, 80% humidity); rated hardware devices for outdoor use and resistant to heat, humidity, and dust (IP65).
- **Operational Duration**: Intended to operate over a 3-day event, each day lasting 12 hours with offline ticketing support.

2.5. Design and Implementation Constraints

- **Budget**: LKR 500,000–1,000,000, using cost-effective RFID and free-tier cloud services (e.g., AWS).
- **Timeline**: 6–9 months, targeting early 2026.
- **Technology**: Flutter (frontend for parent app integration), React/Node.js (portal), MySQL (database).
- **Hardware**: Limited to 20 readers, 12,000 wristbands.
- **Compliance**: Sri Lanka's Personal Data Protection Act (2022) with TLS 1.3, AES-256.
- **Skills**: Second-year students, limiting algorithm complexity.

2.6. User Documentation

- **Visitor Guide**:
  - 1-page PDF (languages: Sinhala, Tamil, English) on wristband scanning instructions, registration process overview, and batch and group feature usage(location sharing, child tracking).
  - Provided via QR code at the venue and online during registration.
- **Staff Manual**:
  - 5-page PDF on registration steps, batch management, tag recycling procedures, and viewing live visitor analytics (active / excited users).
  - Used during 1-hour pre-event training sessions.
- **Technical Documentation**:
  - 10-page PDF on hardware setup and configuration, database schema and maintenance procedures, visitor portal backend / frontend deployment and troubleshooting and log access hosted on GitHub.
  - Audience: System developers and maintainers.
  - Access: GitHub repo link shared with internal team and future maintainers.

2.7. Assumptions and Dependencies

- **Assumptions**:
  - **RFID Adoption**: More than 95% of attendees use RFID wristbands; Up to 5% may use paper tickets.
  - **Data Consent**: Visitors provide explicit consent and basic information (name, group, organization) during registration.
  - **Batch Leadership Devices**: Teachers and lead persons are assumed to carry smartphones for receiving alerts and managing batch-related features.

    - **Network Availability**: Wi-Fi coverage is available in at least 80% of the venue. A 4G mobile network fallback is assumed to support staff devices and batch leaders' smartphones in areas with limited Wi-Fi coverage.
- **Dependencies**:
    - **RFID Hardware Procurement**: Availability of UHF RFID readers and passive tags within the project's budget constraints.
    - **University Payment Gateway**: Integration with the UOP's payment gateway for online ticketing and registration.
    - **Parent System Components**:
        - Backend services for user management.
        - Database for visitor and batch records.
        - Interactive map and analytics module for zone tracking and batch movement visualization.

3. External Interface Requirements

3.1. User Interfaces

- **Registration Portal (Web)**:
  - Shall use React, accessible via Chrome, Firefox, Safari on desktops/tablets.
  - Shall support registration, ticket purchase, group categorization, and batch management.
  - Shall provide Sinhala, Tamil, English, and WCAG 2.1 accessibility.
  - Shall process registrations within 5 seconds.
- **Staff Interface (Ticketing View)**:
  - Shall integrate with the parent system's React/Node.js dashboard.
  - Shall display active/excited visitors, batch alerts, and tag status.
  - Shall require role-based access.
- **Mobile App Integration**:
  - Shall interface with the parent system's Flutter app for batch location sharing and alerts.
  - Shall send proximity alarms to teachers/leads via push notifications or SMS.
  - Shall send proximity alarms to teachers, Organizing committee members and security personnel.

3.2. Hardware Interfaces

- **UHF RFID Readers**:
  - Shall include 10–20 readers (IP65, 10m range) at entry/exit gates and zones.
  - Shall process 100 scans/minute/reader with ≤ 1% error rate.
  - Shall support proximity detection for batch tracking.
  - Shall connect via Wi-Fi (802.11ac) or Ethernet.
- **RFID Wristbands**:
  - Shall include 12,000 UHF wristbands with unique IDs.
  - Shall support read/write for ticketing, batch assignment, and status updates.Wristbands shall be **color-coded** based on visitor group type (e.g., blue for school students, green for university students, red for public). This visual cue will assist staff in quickly identifying group categories at a glance.
- **Local Servers**:
  - Shall provide backup storage for ticketing, batch, and tag data.
  - Shall support MySQL with RAID.

3.3. Software Interfaces

- **University Payment Gateway**:
  - Shall integrate for ticket purchase (TLS encryption).
  - Shall validate tickets within 3 seconds.

- **MySQL Database**:
    - Shall store visitor, ticket, batch, and tag data (see Appendix F.2).
    - Shall support relationships (e.g., visitor-to-batch, tag-to-visitor).
    - Shall handle 10,000 concurrent queries with ≤ 5-second latency.
- **Parent System Backend**:
    - Shall provide scan, batch, and tag data for analytics and maps.
    - Shall use REST APIs for data exchange.

3.4. Communications Interfaces

- **Wi-Fi (802.11ac)**: Shall provide 100 Mbps, WPA3, for 10,000 devices.
- **4G Backup**: Shall provide 20 Mbps for RFID readers.
- **HTTPS/ TLS**: Shall use TLS 1.3 for secure data.
- **SMS Gateway**: Shall send batch proximity alerts within 10 seconds.

4. System Features

4.1. RFID Ticket Validation (FR-T1)

Purpose: To validate RFID wristbands for entry and access control.

Functionality:

- Shall validate 12,000 wristbands at 10–20 gates with ≤ 1% error rate.
- Shall check permissions for restricted zones.
- Shall log entry/exit times for analytics.

Requirements:

- Shall process scans within 3 seconds.
- Shall integrate with the payment gateway.

4.2. Shall support offline scanning with post-event sync.Fraud Prevention (FR-T2)

Purpose: To prevent ticket duplication and unauthorized access.

Functionality:

- Shall use unique RFID IDs to detect duplicates.
- Shall log scan attempts for audit.
- Shall alert organizers of fraud via dashboard.
- Shall prevent duplicate active RFID tag assignments.

Requirements:

- Shall ensure ≤ 1% false positives.
- Shall encrypt logs (AES-256).

4.3. Visitor Registration and Group Categorization (FR-T3)

Purpose: To register visitors, categorize by group, and assign RFID tags.

Functionality:

- Shall collect name, optional contact, organization, and group type (School Student, Teacher, University Student, Company Representative, General Public).
- Shall support batch registration for school students, friends, and families.
- Shall assign unique RFID tags linked to visitor records.
- Shall integrate with payment gateway for ticket purchases.

Requirements:

- Shall use React (frontend) and Node.js (backend).
- Shall support Sinhala, Tamil, English, and WCAG 2.1.
- Shall process registrations within 5 seconds.

4.4. Database Integration (FR-T4)

Purpose: To manage visitor, batch, and tag data for analytics.

Functionality:

- Shall store visitor profiles, tickets, batches, tags, and logs in MySQL (see Appendix F.2).
- Shall support relationships (e.g., visitor-to-batch, tag-to-visitor).
- Shall provide data for analytics and maps.

Requirements:

- Shall handle 10,000 concurrent queries with ≤ 5-second latency.
- Shall use AES-256 encryption.
- Shall support ACID transactions.

4.5. Batch Registration: School Students (FR-T5)

Purpose: To register school students under a teacher with proximity-based safety.

Functionality:

- Shall register up to 100 students per batch under a teacher's RFID tag.
- Shall track students within a 10-meter radius of the teacher.
- Shall trigger an alarm (push notification/SMS) to the teacher if a student exceeds the radius.
- Shall log proximity violations for organizer review.

Requirements:

- Shall process proximity checks every 5 seconds.
- Shall deliver alarms within 10 seconds.
- Shall anonymize student data.

4.6. Batch Registration: Friends (FR-T6)

Purpose: To enable groups of friends to share locations on maps.

Functionality:

- Shall register up to 15 friends per batch with linked RFID tags.
- Shall display batch members' locations on the parent system's map.
- Shall update locations every 5 seconds via RFID scans.

Requirements:

- Shall integrate with the parent system's Flutter app.
- Shall ensure ≤ 5-second map refresh.
- Shall anonymize location data.

4.7. Batch Registration: Families (FR-T7)

Purpose: To allow family leads to track members and ensure child safety.

Functionality:

- Shall register up to 25 family members under a lead person's RFID tag.
- Shall display family members' locations on the parent system's map.

- Shall track children (under 16) within a 10-meter radius of the lead.
- Shall trigger an alarm (push notification/SMS) to the lead if a child exceeds the radius.

Requirements:

- Shall process proximity checks every 5 seconds.
- Shall deliver alarms within 10 seconds.
- Shall support age-based tagging.
- Shall anonymize data.

4.8. RFID Tag Lifecycle Management (FR-T8)

Purpose: To manage RFID tag assignment, deactivation, and recycling.

Functionality:

- Shall assign RFID tags to visitors at registration.
- Shall mark visitors as “checked-out” and deactivate tracking upon exit detection.
- Shall make RFID tags available for reuse within 5 minutes of exit.
- Shall log entry/exit times and visit locations for analytics.

Requirements:

- Shall process exit detection within 3 seconds.
- Shall release system resources for inactive visitors.
- Shall ensure tags are not reassigned while active.

4.9. Staff Visitor Status View (FR-T9)

Purpose: To allow staff to monitor active and excited visitors.

Functionality:

- Shall display current active and excited visitors with group and batch details.
- Shall show entry/exit times and visited locations.
- Shall support filtering by group type or batch.

Requirements:

- Shall integrate with the parent system’s dashboard.
- Shall update within 5 seconds.
- Shall require role-based access.

4.10. Out-of-Zone Visitor Alarm Notification (FR-T10)

Purpose: To alert security staff and OC members when a visitor exits designated exhibition zones, using RFID tracking, in order to protect faculty property and ensure visitor safety.

Functionality:

- Shall continuously monitor visitor locations using RFID tags.
- Shall detect and flag any visitor who crosses out of predefined exhibition boundaries.

- Shall send real-time alarm notifications to nearby security staff and OC members.
- Shall include visitor ID, group/batch, last known location, and time of zone breach.
- Shall allow configuration of alert zones and staff response radius.

Requirements:

- Shall integrate with the RFID tracking system and staff location database.
- Shall deliver alerts within 3 seconds of zone breach detection.
- Shall support SMS, email, and dashboard alerts based on user role and proximity.
- Shall require role-based access and support custom notification preferences.
- Shall log all alarms, locations, and responses for audit and reporting.

5. Other Nonfunctional Requirements

5.1. Performance Requirements

- Shall support up to 10,000 concurrent users with a scan response time ≤ 3 seconds.
- Shall process ≥ 100 RFID scans per minute with error rate ≤ 1% .
- Shall handle 1,000+ concurrent active visitor records without performance degradation.
- Shall process batch proximity module every 5 seconds.
- Shall auto-release RFID resources within 5 minutes of visitor exit.
- Shall maintain ≥ 99% uptime during the 3-day event.

5.2. Security Requirements

- Shall comply with Sri Lanka's Personal Data Protection Act (2022).
- Shall encrypt data in transit (TLS 1.3) and at rest secured using AES-256 encryption.
- Shall anonymize visitor and batch data for analytics.
- Shall enforce Role-Based Access Control (RBAC) for backend operations and administrative dashboard.
- Shall implement audit logging of all RFID scans, tag assignments, user operations (e.g., registration edits), and system alerts. Logs shall include timestamps and device IDs, be encrypted, and retained for a minimum of 30 days post-event.

5.3. Software Quality Attributes

- **Usability**: Shall require no training for visitors; 1-hour training for staff.
- **Reliability**: Shall achieve ≥ 99% uptime and maintain an accuracy of an error rate ≤ 1%.
- **Scalability**: Shall support 10,000 users and be reusable for future events.
- **Maintainability**: Shall be modular, well-documented, and structured codebase for ease of future updates.

6. Other Requirements

6.1. Testing Requirements:

- The system shall undergo:
  - Stress testing with 10,000+ simulated scans, batch operations, and tag lifecycles.
  - RFID accuracy testing, including proximity detection and tag recycling in a mock environment.
  - Validation of offline ticketing and batch / group functionality under limited connectivity.

6.2. Deployment Requirements:

- The system shall be fully deployed 1 week before the event.
- The development team shall include on-site support by the development team.
- Upon visitor exit scan, the system shall optionally display or print a QR code linking to a visitor feedback form, particularly targeting school and university groups for educational evaluation.

7. Appendices

Visitor Groups

School Student

Teacher

University Student

Company Representative

General Public

Sample Database Schema

Table: visitors

Field	Type	Description
id	UUID	Unique record ID
name	String	Visitor's full name
group_type	Enum	Group classification
organization	String	Affiliation or company/school
rfid_tag	String	RFID UID assigned
status	Enum	`active`, `exited`
visited_locations	Enum	Code numbers of exhibit locations on faculty
entry_time	Timestamp	Time of check-in
exit_time	Timestamp	Time of check-out

Table: rfid_tags

Field	Type	Description
tag_id	String	Unique RFID tag ID
assigned_to	UUID	Visitor ID (nullable)
status	Enum	`available`, `in_use`

Glossary

- RFID: Radio Frequency Identification, using radio waves to track wristbands.
- UHF RFID: Ultra-High Frequency RFID (860–960 MHz, 10m range).
- MySQL: Open-source database for visitor, batch, and tag data.
- React/Node.js: Frameworks for registration portal.
- TLS 1.3: Protocol for secure data transmission.
- AES-256: Encryption standard for stored data.
- Batch Registration: Grouping attendees for tracking and safety.