**Software Requirements Specification**

**for**

**QR Ticketing System**

**Version 0.1**

**Prepared by**

**Aabhas Chopra 19105115**

**Annchit Bhatnagar 19105123**

**Divyansh Gupta 19104101**

**April 16, 2023**

**Table of Contents**

[**1. Introduction 4**](#_1fob9te)

[1.1 Purpose 4](#_3znysh7)

[1.2 Document Conventions 4](#)

[1.3 Intended Audience and Reading Suggestions 4](#_6dodr1d9eaz8)

[1.4 Product Scope 4](#)

[1.5 References **5**](#)

[**2. Overall Description 6**](#)

[2.1 Product Perspective 6](#_2s8eyo1)

[2.2 Product Functions **6**](#_80g8ukqr2pbs)

[2.3 User Classes and Characteristics 7](#)

[2.4 Operating Environment 8](#)

[2.5 Design and Implementation Constraints 8](#)

[2.6 User Documentation 8](#)

[2.7 Assumptions and Dependencies **8**](#)

[**3. External Interface Requirements 9**](#)

[3.1 User Interfaces 9](#_2jxsxqh)

[3.2 Hardware Interfaces 9](#)

[3.3 Software Interfaces 9](#)

[3.4 Communications Interfaces **9**](#)

[**4. System Features 10**](#)

[4.1 User Registration and Login Module 10](#)

[4.2 Session Registration Module 11](#_xax2rbz3ghzq)

[4.3 Email Registration Module 12](#_m6angzsz0kum)

[4.4 Session Uploader Module 14](#_wsx8p17rt25c)

[4.5 QR Code Scanning Module **15**](#_lnqp2t5m931f)

[**5. Other Nonfunctional Requirements 17**](#_3whwml4)

[5.1 Performance Requirements 17](#_2bn6wsx)

[5.2 Safety Requirements 17](#)

[5.3 Security Requirements 17](#)

[5.4 Software Quality Attributes 17](#)

[5.5 Business Rules 18](#)

[5.6 Usability and Compatibility **18**](#_71htiwzauja1)

[**6. Other Requirements 19**](#)

[**Appendix A: Glossary 20**](#_sn15xmap9aqo)

[**Appendix B: Analysis Models 20**](#_muw39je2m8l5)

**Revision History**

| **Name** | **Date** | **Reason For Changes** | **Version** |
| --- | --- | --- | --- |
| Aabhas Chopra | 16-04-23 | Initial Version | 0.1 |
|  |  |  |  |
|  |  |  |  |

# 

# **Introduction**

## **Purpose**

The purpose of this document is to provide a detailed specification of the requirements for the development of a web-based application system that allows users to register for sessions and receive a unique QR code via email. This QR code will mark the user's attendance at the session.

The system will also include a session uploader module allowing the event organizers to upload details of the session and set up new sessions or events scheduled for future dates.

The system will be scalable, reliable, and secure to meet the institution's standards and maintain its trustworthiness.

## **Document Conventions**

This document uses the following conventions

| QR code | Quick Response code |
| --- | --- |
| DB | Database |
| OS | Operating System |
| ER | Entity Relation |
| DBMS | Database Management System |
| RDBMS | Relational Database Management System |

## **Intended Audience and Reading Suggestions**

This project is a QR ticketing system restricted within the bounds of college premises. This project has been implemented under the guidance of college professor Dr. Rajesh Bhatia. It is helpful for the students and the college's event-organizing committees to coordinate events and sessions on the institution's premises efficiently.

## **Product Scope**

The purpose of the QR Ticketing system is to ease the organization of in-house events and sessions of the college and to regularize the attendance system for the same. The system is based on the institution's internal database and mailing system to maintain and deliver its event and enrolled users’ details. The project will be able to sustain the traffic and the stakeholders' requirements within the institute.

## **References**

* "User Interface Design Style Guide, Version 1.2, Author: John Smith, Date: 01/01/2023, Source: Company Intranet"
* "Software Development Contract, Version 2.0, Author: ABC Company, Date: 05/01/2023, Source: Legal Department"
* "ISO 9001:2015 Quality Management System Standard, Version 2.0, Author: International Organization for Standardization, Date: 09/01/2022, Source: ISO website"
* "System Requirements Specification, Version 1.5, Author: Mary Johnson, Date: 03/01/2023, Source: Project Management Tool"
* "Use Case Diagram, Version 1.0, Author: Jane Doe, Date: 02/01/2023, Source: Requirements Analysis Report"
* "Project Charter, Version 1.1, Author: Project Manager, Date: 01/15/2023, Source: Project Management Tool"

# **Overall Description**

The system will be designed as a client-server application with a web-based user interface that users can access via a web browser. The application will be developed using modern web technologies, including HTML, CSS, JavaScript, and a server-side language like PHP or Python. The system will integrate with the email system to send users emails containing unique QR codes.

The web application users can register for the events listed on the application. Upon successful registration, they will be provided a QR code unique to them in an email which will authorize them for an entry to enter the session arena. Event organizers can host any new sessions or events with a definable scope of whoever is eligible to attend their session.

## **Product Perspective**

The system will be designed to be scalable, reliable, and secure. The system will be a client-server application, with the server hosting the application and the database. The system will be able to accommodate all types of events, and there will be no limit on the number of users who can attend the session if the organizer intends to do so.

The users register onto the web application to be able to look at and enroll for various events that have been listed. Once done, the user can choose whichever event or session they wish to attend and register for any events they deem fit. No upper limit to the number of attendees has been enforced; however, the organizer is entitled to specify if only a set number of attendees can attend a session or if only a specific batch will be eligible to attend the session. Any changes in the schedules of the events or the descriptions and details of the events will be notified to the users already enrolled for the event. The same will also be updated in the event details on the web application.

The system will house five modules

* User Management Module
* Session Registration Module
* Email Generation Module
* Session Uploader Module
* QR Code Scanning Module

## **Product Functions**

The system will consist of the following modules:

* User Management Module: It is responsible for setting up user profiles for the web application to allow for further usage during session registrations and to access the upcoming and currently underway events. Users must confirm their email address by clicking on a link sent to them via email. Users will be able to log in using their email addresses and password.
* Session Registration Module: This module assumes the responsibility of registering the individuals for different events they choose to attend. It keeps track of how many vacant slots are available in each event if any limit has been specified, and what are the timelines of each event. If some event’s registration time exhausts, it doesn’t allow further enrollments.
* Email Generation Module: It is responsible for generating the email to be shared with each attendee of the events. The email contains the details of the event the user is enrolled in and a unique QR code that grants them entry into the event's premises.
* Session Uploader Module: This module allows the event organizers to publish new event announcements and new upcoming events. In case there is some update in the schedule of the events, the organizer can change the event's details on the web application and send an update email to all attendees of the event. The module will require the event organizer to provide the following information:
* Session name
* Session date and time
* Session location
* Session description
* Session capacity
* Other relevant details
* QR Code Scanning Module: This module is intended for the event organizers to use at the time of the event to grant access to the attendees. The attendees can show their QR codes received on respective mails to the event coordinators, who will scan them using this module and automatically mark them as present for future reference.

## **User Classes and Characteristics**

The system will be designed for:

1. Regular users: Users who want to register for a session and receive a unique QR code via email.
2. Event organizers: Users who will upload session details.
3. Event coordinators: Users who will crosscheck the QRs at the time of the event

Regular Users are the individuals that will use the web application service to register for various events and to access any updates to the event schedules, if any.

Event organizers are responsible for managing current events and setting up new events. Any changes to the current events for which enrollment has already been opened need to be notified to all the enrolled users, which will take place via email. The event organizers can also define the scope of the event to specify if only a particular batch or a certain number of individuals will be eligible to attend their event.

Event Coordinators are the individuals who check the validity of the QR codes that the regular users produce in front of them at the time of entrance into the premises of the event. When the event coordinators scan a QR, it gets marked as ‘present’ for the session, and the code is no longer scannable for another attendance marking.

## **Operating Environment**

The system will be a web-based application accessible to users via their web browser. The system will be hosted on a secure server. The system will be developed using modern web technologies and compatible with all major web browsers.

The event organizers and the Regular users can perform their tasks remotely without any human intervention. The event coordinators must be physically present at the event's premises with a device compatible with modern browsers and a camera that will scan QR codes, which can be any smartphone with a secure internet connection.

## **Design and Implementation Constraints**

The QR ticketing system must conform to the institution's design conventions and programming standards. It must also integrate with the company's existing hardware and software systems. Additionally, the system must comply with any relevant security considerations and regulations that must be considered for implementation.

In case a separate server needs to be established, then constant upkeep of that new server also needs to be kept in check.

## **User Documentation**

The ticketing system will accompany user documentation in user manuals and online help. The documentation will be provided in a standard format like PDF or HTML.

A user manual will also be made available on the web application to reference its use.

## **Assumptions and Dependencies**

It is assumed that the QR ticketing system will be built using modern web development technologies and frameworks and that the necessary hardware and software infrastructure will be available for its implementation. The system depends on the availability and functionality of existing software systems, such as the user database and email system. It is also assumed that event coordinators will be trained on using the system before its implementation.

# **External Interface Requirements**

## **User Interfaces**

The ticketing system will have a web-based user interface with a responsive design to support access from different devices. The interface will include the following characteristics:

* Simple and intuitive design with clear navigation menus and buttons
* Consistent layout and design elements throughout the system
* Accessible to users with disabilities (e.g., support for screen readers)
* Support for multiple languages

## **Hardware Interfaces**

The ticketing system will interact with standard hardware components such as desktop computers, laptops, tablets, and smartphones. No specific hardware interfaces are required beyond standard input/output mechanisms.

## **Software Interfaces**

The ticketing system will interact with several software components, including

* Database Management System (DBMS) to store and retrieve data
* Operating System (OS) to provide a platform for the system to run on
* Web Server to serve the web pages to the users
* Email Service Provider to send notifications to users and agents

The data items or messages coming into and going out of the system will include:

* User authentication and authorization data
* Ticket data (e.g., creation, updates, comments)
* Email notifications (e.g., new tickets, updates, reminders)

## **Communications Interfaces**

The ticketing system will require communication functions such as

* Web browser for accessing the system
* Email protocols (e.g., SMTP) for sending and receiving email notifications
* Network protocols (e.g., HTTP, HTTPS) for transmitting data between the client and server
* Security protocols (e.g., SSL, TLS) for encrypting data transmitted over the network.

# **System Features**

## **User Registration and Login Module**

4.1.1 Description and Priority

The user management module will allow users to create an account by providing their name, email address, and password. The system will verify that the email address is valid and has not already been used to create an account. Once the account is created, the user will receive an email with a link that they need to click on to activate their account. Users will be able to log in using their email addresses and password.

4.1.2 Stimulus/Response Sequences

* User navigates to the registration page and clicks the "register" button.
* System displays a registration form where users enter their details, such as email and password.
* User submits the form by clicking on the "submit" button.
* The system verifies that the email is unique and the password is strong enough.
* If the email is already registered, the system displays an error message.
* If the password is not strong enough, the system displays an error message.
* If the email and password are valid, the system creates a new user account and redirects the user to the login page.
* The user navigates to the login page and enters their email and password.
* The system verifies the user's credentials and logs them in.
* If the email or password is incorrect, the system displays an error message.
* If the user's credentials are correct, the system redirects the user to the home page.

4.1.3 Functional Requirements

* Registration
  1. The system shall allow users to register for a new account by providing their details, such as name, email address, and password.
  2. The system shall validate the email address and ensure it is unique before creating a new account.
  3. The system shall provide a feedback message to the user if the registration is successful or unsuccessful.
* Login

1. The system shall provide a login page where users can enter their email addresses and password to access their accounts.
2. The system shall validate the user's email address and password before allowing them to log in.

* Password Management

1. The system shall allow users to reset their password in case they forget it.
2. The system shall ensure that password reset requests are authorized and only sent to the email address associated with the user account.
3. The system shall provide a feedback message to the user if the password reset is successful or unsuccessful.

## **Session Registration Module**

4.2.1 Description and Priority

The session registration module will allow users to register for a session by providing their details, including name, email address, and session information. Users will be able to register for multiple sessions. The module will verify that the user is not already registered for the session and that the session has not reached its capacity limit.

4.2.2 Stimulus/Response Sequences

* The user navigates to the sessions registration page and selects a movie they want to watch.
* The system displays the available sessions, timings, seat availability, and pricing.
* The user selects a session and chooses the number of seats they want to book.
* The system calculates the total amount based on the number of seats and the ticket price and displays it to the user.
* The user enters their personal and payment details in the registration form, such as name, email, and credit card information.
* The system validates the user's payment details and reserves the selected seats for the session.
* If the payment is successful, the system confirms the booking and displays a confirmation message to the user, along with a unique booking ID.
* The system sends a confirmation email to the user containing booking details and the booking ID.
* If the payment is unsuccessful, the system displays an error message to the user and prompts them to try again with different payment details.

4.2.3 Functional Requirements

* Session Selection
  1. The system shall display the available sessions for the selected movie along with their timings, seat availability, and pricing.
  2. The system shall allow users to select a session they wish to attend.
  3. The system shall update the seat availability status based on the user's selection.
* Seat Selection

1. The system shall allow the user to select the number of seats they want to book.
2. The system shall ensure that the user cannot book more seats than are available for the selected session.

* Payment

1. The system shall provide users a secure payment gateway to enter their payment details, such as credit card information.
2. The system shall validate the user's payment details and ensure the transaction is secure.
3. The system shall calculate the total amount based on the number of seats and the ticket price.
4. The system shall confirm the booking if the payment is successful.
5. The system shall provide the user with a unique booking ID.

* User Profile

1. The system shall allow registered users to view their past bookings and cancel a booking if necessary.
2. The system shall ensure that only the user who made the booking can cancel it.

* Accessibility

1. The system shall provide accessibility features such as keyboard navigation and screen reader support to allow users with disabilities to access the session registration functions.
2. These functional requirements ensure that the session registration for a movie ticketing system is efficient, secure, and user-friendly. By meeting these requirements, users can easily browse and book movie sessions and enjoy a hassle-free cinema experience.

## **Email Registration Module**

4.3.1 Description and Priority

The email generation module will generate an email containing a unique QR code for each user and session. The email will be sent immediately after the user registers for a session. A reminder email will also be sent before the event. The QR code will be generated based on the session and user login. The module will also verify that the email address is valid and that the email was successfully sent.

4.3.2 Stimulus/Response Sequence

* Event Organizer Logs In: The event organizer logs into the QR-based email generation module using their username and password.
* Upload Attendee List: The event organizer uploads the list of registered attendees, including their email addresses and registration details.
* Customize Email Template: The event organizer customizes the email template by including event details such as venue, date, and time.
* Generate QR Code: The module generates a unique QR code for each attendee based on their email address and registration details.
* Send Email: The module integrates with email services such as Gmail, Outlook, and others to send personalized emails to attendees, including the QR code and event details.
* Attendee Receives Email: The attendee receives the personalized email and can access their registration details through the scannable QR code.
* Attendee Receives Reminder Email: The module sends a reminder email to attendees to remind them of the event and encourage them to attend.
* Attendee Receives Confirmation Email: The module sends a confirmation email to attendees once they have successfully registered for the event.
* Attendee Receives Feedback Form: After the event, the module provides attendees with a feedback form or survey to gather feedback and improve future events.

4.3.3 Functional Requirements

* QR Code Generation: The QR-based email generation module should generate a unique QR code for each attendee based on their email address and registration details.
* Email Integration: The module should integrate with email services such as Gmail, Outlook, and others to generate and send personalized emails to attendees.
* QR Code Scanning: The QR-based email generation module should provide attendees with a scannable QR code that can be scanned by a mobile device to verify their registration and attendance.
* Email Reminder: The module should be capable of sending email reminders to attendees to remind them of the event and encourage them to attend.
* Email Confirmation: The module should send a confirmation email to attendees once they have successfully registered for the event.
* Event Feedback: The module should provide attendees with a feedback form or survey after the event to gather feedback and improve future events.

## **Session Uploader Module**

4.4.1 Description and Priority

The session uploader module will allow event organizers to upload session details. The module will require the event organizer to provide the following information:

* Session name
* Session date and time
* Session location
* Session description
* Session capacity
* Other relevant details

4.4.2 Simulated Response

* Log In: The event organizer logs into the BookMyShow app using their username and password.
* Create Event: The event organizer creates a new event by providing the event name, venue, date, time, and ticket prices.
* Customize Event Details: The event organizer customizes the event details, including the event description, images, and videos.
* Publish Event: The event organizer publishes the event on the BookMyShow platform.
* Set Ticket Availability: The event organizer sets the ticket availability and pricing for the event, including early bird discounts or VIP tickets.
* Send Event Reminders: The event organizer sends event reminders to attendees via email or push notifications.
* Manage Seating Arrangements: The event organizer manages seating arrangements, including reserved seats or assigned seating.
* Manage Refunds: The event organizer manages refunds and cancellations for attendees who cannot attend the event.
* Ensure User Authentication: The Web app provides secure authentication and authorization protocols to ensure that only authorized event organizers can access attendee data and manage events.

4.4.3 Functional Requirements

* Session Creation: The event organizer should be able to create an event by providing details such as event name, date, time, location, ticket prices, and any other relevant information
* Session Management: The event organizer should be able to manage the events they have created, including editing event details, updating ticket prices, and setting up seating arrangements.
* Ticket Management: The event organizer should be able to manage the tickets for their events, including setting up ticket types, pricing, and availability.
* Payment Management: The event organizer should be able to manage payments for their events, including processing refunds, viewing payment histories, and generating revenue reports.
* Attendee Management: The event organizer should be able to manage attendees for their events, including tracking ticket sales, issuing tickets, and sending communication to attendees.

## **QR Code Scanning Module**

4.5.1 Description and Priority

The QR code scanning module will allow the event organizer to scan the QR code of the user at the time of the event. The module will verify that the QR code is valid and has not already been used. The module will mark the user's attendance and update the session attendance record.

4.5.2 Simulated Response

* User opens the personal email and points the QR Code to the camera, and waits till the camera scans the QR code.
* Once the QR code is scanned, the system validates the QR code by checking against the event database.
* If the QR code is invalid or doesn’t match any records, it displays an error.
* When the QR code is scanned and validated, the system updates the event database in real-time to ensure accurate attendance tracking and prevent double entry.
* The app also provides immediate feedback to the user to confirm that the scan was successful.

4.5.3 Functional Requirements

* Scan QR codes: The module should be able to scan QR codes using the device's camera and interpret the data embedded in the code.
* Validate QR codes: The module should verify that the scanned QR code is valid and corresponds to an event attendee or an event-related item.
* Access control: The module should enable event organizers to control access to various areas and activities within the event based on the validity of the scanned QR code.
* Real-time syncing: The module should update the event database in real time when a QR code is scanned to ensure accurate attendance tracking and prevent double entry.
* Multi-device support: The module should be compatible with multiple devices and platforms, such as iOS and Android, to enable attendees to use their smartphones to scan QR codes.

# **Other Nonfunctional Requirements**

The system will be designed with the following non-functional requirements:

## **Performance Requirements**

* The system will handle many users and sessions.
* The system will be able to accommodate up to 2000 users simultaneously.
* The system will be designed to handle the load of generating and sending emails to many users.
* The response time for user actions should not exceed 5 seconds.

## **Safety Requirements**

* The system should prevent unauthorized access to users' personal information.
* The system should ensure that the QR code generated for each user is unique and cannot be easily replicated by others.
* The system should prevent any accidental deletion of the session details by uploaders and have a backup system to ensure that user data is not lost in case of system failure.

## **Security Requirements**

* The system should use encryption techniques to protect user data during transmission and storage.
* The system should have user identity authentication requirements, such as password protection or two-factor authentication, to ensure only authorized users can access their accounts.
* User information will be stored in a secure, encrypted database and protected with firewalls.
* User passwords will be encrypted using a one-way hash algorithm.
* The system will also use a two-factor authentication mechanism to ensure secure access to user accounts.

## **Software Quality Attributes**

* Usability: The system should have a user-friendly interface that is easy to navigate and understand.
* Maintainability: The system should be easy to maintain, update, and fix bugs in the future.
* Availability: The system should have a high level of availability to ensure that users can access it at any time.
* Reliability: The system should be reliable and accurate in generating and sending QR codes and marking attendance.

## **Business Rules**

* Only registered users can register for sessions and receive a QR code.
* Each user can only register for a session once.
* The system should prevent users from registering for already full sessions.
* Session uploaders should be able to edit or delete session details, but only if the session has not started.

## **Usability and Compatibility**

* The system will be designed to be user-friendly and easy to use. The user interface will be intuitive and easy to navigate.
* The system will be tested for usability and user satisfaction.
* The system will be designed to be compatible with a wide range of browsers and devices.
* The system will be tested for compatibility with popular browsers such as Google Chrome, Mozilla Firefox, and Microsoft Edge, as well as with popular devices such as desktop computers, laptops, tablets, and smartphones.

# **Other Requirements**

* Database requirements: The system should use a relational database management system (RDBMS) to store all data related to events, sessions, users, and bookings. The database should be designed to ensure data integrity, security, and scalability. The system should support backup and recovery procedures in case of data loss or corruption.

* Internationalization requirements: The system should support multiple languages and currencies to cater to users from different regions. All user-facing text and messages should be translatable, and the system should display dates and times in the user's local format. The system should also follow internationalization standards for formatting numbers, addresses, and other information.

* Legal requirements: The system should comply with all applicable laws and regulations, including data protection, privacy, and security. The system should obtain users' consent before collecting, storing, or processing personal information. The system should also allow users to delete their data upon request and have measures to prevent unauthorized access to user data.

* Reuse objectives: The system should be designed with modularity and extensibility to allow future enhancements and feature additions. The system should also use open standards and APIs to enable integration with third-party services and platforms. The system should follow best practices for code quality, documentation, and version control to facilitate reuse and maintenance by developers.

* Accessibility requirements: The system should be designed to be accessible to users with disabilities, including those who use screen readers, voice recognition software, and other assistive technologies. The system should follow accessibility standards and guidelines, such as the Web Content Accessibility Guidelines (WCAG), to ensure all users can access and use the system's features and content.

# **Appendix A: Glossary**

* User: A person who registers for a session and receives a unique QR code via email.
* Event organizer: A person who uploads session details.
* Session: An event that users can register for.
* QR code: A unique code that users receive via email and use to mark their attendance at the session.
* Web-based application: An application that is accessed via a web browser.
* RDBMS: It is a software system used for managing relational databases. Relational databases store data in tables, where each table consists of rows and columns. RDBMSs use a structured query language (SQL) to access the data stored in these tables.
* ER Model: It is a conceptual data model used to represent the relationships between entities in a database. The ER Model is based on the concept of entities, which are real-world objects or concepts that have attributes or properties.

# **Appendix B: Analysis Models**

Entity-Relationship Diagrams

+-------------+

| User |

+-------------+

| email (PK) |

| password |

| confirmed |

+-------------+

+--------------+

| Event |

+--------------+

| event\_id (PK)|

| name |

| description |

| date/time |

| location |

| max\_capacity |

+--------------+

+--------------+

| session |

+--------------+

| session\_id |

| start\_time |

| end\_time |

| slots\_avail |

| event\_id (FK)|

+--------------+

+------------------+

| QR\_Code |

+------------------+

| qr\_code\_id |

| user\_email (FK) |

| session\_id (FK) |

+------------------+

+--------------+

| Organizer |

+--------------+

| organizer\_id |

| name |

| contact\_info |

+--------------+

User attends → Session

Event has → Session

Organizer creates → Event

Session belongs to → Event

Session updates → Event

User receives → Email

Organizer scans → QR\_Code