Software Requirements Specification (SRS)

Project Title: Warima Events — Event Organizer Booking System

Version: 1.0

Prepared By: Warima Edgar 23/06030

Date: 4/10/2025

1. Introduction

1.1 Purpose

The purpose of this document is to define the functional and non-functional requirements for the Warima Events booking system. This system allows users to plan events by submitting event details and enables administrators to manage users, events, and feedback reports.

1.2 Scope

Warima Events is a web-based platform that allows users to:

Register and log in

Book event organizing services (weddings, birthdays, corporate, etc.)

View booking confirmation

Submit feedback or reports

Admin users can:

- Access an admin dashboard
- Manage user accounts
- Manage submitted event requests
- Review user reports and feedback

1.3 Intended Audience

Developers

System testers

Project stakeholders

Hosting administrators

1.4 Definitions

Admin: A privileged user who can manage users and events.

Event Booking: Submission of an event request by a user.

Report: Feedback or complaint submitted by a user.

2. Overall Description

2.1 Product Perspective

The system is independent and locally hosted. It uses a standard LAMP stack (Linux/Apache/MySQL/PHP), with HTML/CSS/JavaScript for the frontend.

2.2 Product Functions

User authentication

Event request submission

Report feedback form

Dashboard interfaces (user and admin)

Admin event and user management

2.3 User Classes and Characteristics

User Role

Description

Regular User

Can register, log in, and book events

Admin

Can manage users, view stats, and handle event approvals

2.4 Operating Environment

OS: Windows (local development)

Browser: Chrome/Firefox/Edge

Languages: PHP 7+, HTML5, CSS3, JavaScript

Database: MySQL 5+

2.5 Constraints

Must be hosted on a local or web server with PHP/MySQL

Requires active internet connection for CDN assets

Currently no mobile app version

2.6 Assumptions and Dependencies

Users must have valid credentials to access booking features

PHP sessions are used for login handling

Admin dashboard assumes real-time connection to backend endpoints

3. System Features

3.1 User Registration and Login

New users can sign up with name, email, and password

Existing users can log in

Sessions are used to track authenticated users

3.2 Event Booking

Authenticated users can access the "Plan My Event" form

Required fields: event type, location, date, number of guests

Optional notes can be added

Submission redirects to a confirmation page

3.3 Report Submission

Users can submit reports or feedback from their dashboard

Reports include user ID, text, timestamp, IP, and user agent

3.4 Admin Dashboard

View statistics (total users, total events)

Quick links to manage users and events

Access to view and handle customer feedback

3.5 Manage Events and Users

Admin can approve/reject events

Admin can suspend users or view their profile info

4. External Interface Requirements

4.1 User Interfaces

HTML/CSS frontend with interactive JavaScript features

Responsive design for dashboard and forms

4.2 Hardware Interfaces

Client requires a device capable of running a modern browser

4.3 Software Interfaces

PHP backend with MySQL database

Communication via Fetch API (AJAX) for form submission and admin stats

4.4 Communication Interfaces

HTTP over localhost (can be extended for deployment)

5. Non-Functional Requirements

5.1 Performance

Load time for any page should be under 3 seconds on localhost

5.2 Security

Passwords are stored securely (assumed hashing in backend)

Only authenticated users can book events or submit reports

Admin dashboard is access-restricted

5.3 Usability

Simple, intuitive UI with helpful icons and tooltips

Responsive layout for desktop and mobile viewing

5.4 Reliability

Data persists in the database even on browser reloads

Sessions expire after inactivity (recommended in backend)

6. Appendix

HTML Pages: login, registration, dashboard, event_form, admin_dashboard, confirmation, manage_users, manage_events

Database Tables: users, events, reports

Technologies: PHP, HTML, CSS, JS, MySQL