Software Requirements Specification (SRS)

Project: Book My Venue

Version: 1.0

Prepared by: PseudoClan Private Limited

1. Introduction

1.1 Purpose

This SRS document defines the functional and non-functional requirements for the Book My Venue platform—a web application designed to facilitate venue booking and event planning.

1.2 Scope

The system allows users to:
- Search venues by filters (date, location, price, etc.)
- Book event helpers (cooks, waiters, etc.)
- View service provider listings (photographers, makeup, decorators, etc.)
- Manage user and vendor profiles
- Estimate prices and handle bookings
- Design invitations using Canva API
- Communicate with vendors/freelancers

2. Overall Description

2.1 Product Perspective

The application is a responsive, mobile-first web platform built using:
- Frontend: React.js with Vite and Tailwind CSS
- Backend: Strapi V4 (Node.js)
- Database: PostgreSQL or MySQL (MariaDB)
- Authentication: OTP-based login with JWT
- Payments: Razorpay or Stripe
- Asset Storage: AWS S3 or equivalent

2.2 User Characteristics

Users include:
- Individuals: Planning personal or corporate events

- Vendors: Offering event-related services
- Freelancers: Providing on-demand help

3. Functional Requirements

3.1 User Authentication

OTP-based login via phone number; Social/email login with OTP verification

3.2 User Profile Management

View/edit personal details (editable fields depend on login method)

3.3 Venue Search

Filter by date, price, location, occasion, and capacity

3.4 Venue Listing

Include photos, amenities, capacity, contact info, and reviews

3.5 Event Help Booking

Book event helpers like cooks, drivers, cleaners, etc.

3.6 Booking and Cancellation

Cancellation rules defined by the platform (TBD)

3.7 Reviews and Ratings

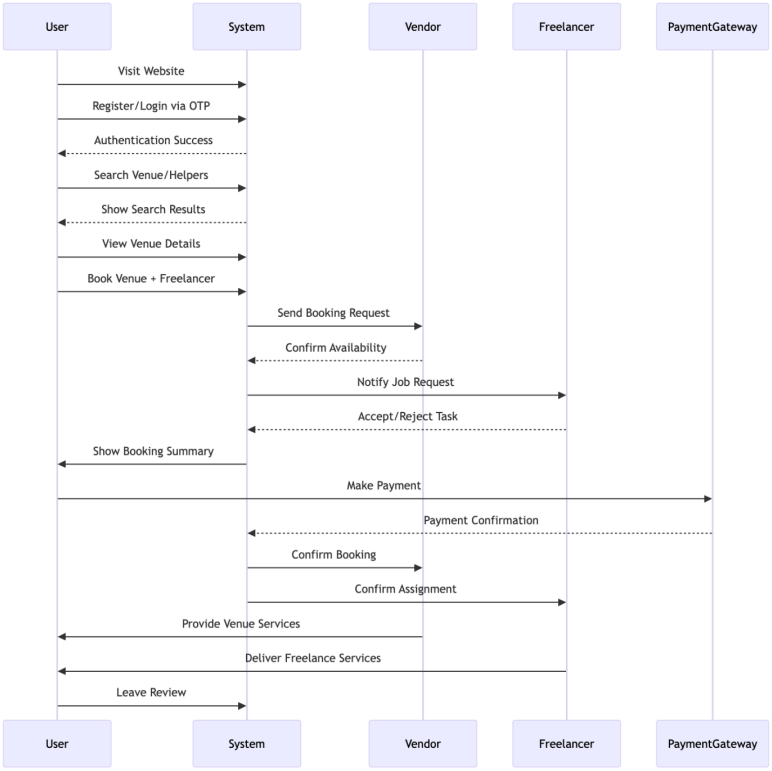
Only post-booking users can submit reviews with ratings/comments

3.8 Payment

In-app payment for individual bookings (implementation TBD)

3.9 Customer Support

Support for cancellations, refunds, and issue resolution (TBD)



4. Non-Functional Requirements

4.1 Performance

Page load time under 3 seconds; Support for 10,000+ concurrent users

4.2 Security

OTP-based authentication; Encrypted user data; Role-based access control (TBD)

4.3 Usability

Fully responsive design for mobile and desktop

4.4 Availability

99.9% uptime commitment

5. System Architecture

Frontend: React.js
Backend: Strapi V4 (Node.js)
Database: PostgreSQL / MySQL

Authentication: JWT + OTP login
Payment Gateway: Razorpay / Stripe
Asset Storage: AWS S3 or equivalent

6. API Endpoints (Sample)

POST /auth/otp-login - Phone-based OTP login

GET /venues - Get list of venues

GET /venues/:id - Venue details

GET /event-help/categories - Fetch event helper categories

POST /event-help/book - Book an event helper

POST /booking/cancel/:id - Cancel a booking

POST /reviews/venue/:venueId - Submit review for a venue

7. Modules and Features

7.1 Venue Management
- Add venues
- Upload photos/assets
- Web scraping for listings (e.g., NCR, Capital Cities)

7.2 Vendor Directory
- Categories: Photographer, Caterer, Decorator, Makeup Artist, Clothing, Jeweller, Gifts, etc.
- Package listing and service details

7.3 User Planning Tools
- Calendar integration
- Task manager
- Budget estimator
- Food menu planning
- Guest list creation
- Vendor selection

7.4 Vendor and Freelancer Portals
- Booking calendar
- Internal messaging system

7.5 Integrations
- Canva API for invitation design
- WhatsApp integration for forwarding and messaging

8. External Resources and References

Frontend:
- React: https://react.dev
- State Management: https://tanstack.com/query/latest/docs/framework/react/overview
- Tailwind CSS: https://tailwindcss.com/docs/installation/using-vite
- Vite: https://v3.vitejs.dev/guide/

Backend:
- Strapi: https://docs-v4.strapi.io/dev-docs/api/rest
- Backend API: https://api.pseudoclan.com/documentation/v1.0.0

Git Repository:
- https://github.com/vivekpseudo/book-my-venue

