



Test Plan and Test Cases

Project: Band Baja Hub
Group No:4

Students:

202412001 Aarti Bodani
202412039 Bhakti Khokhani
202412047 Manya Singh
202412049 Monika Gautam
202412055 Palak Patel
202412065 Dhara Patel
202412088 Hirwa Saraiya
202412105 Aditi shukla

Submission Date: 05-Nov-2025

1. Test Plan

1.1 Objective

To validate that Band Baja Hub performs correctly for core flows: user registration/login, service listing, adding services (venues, caterers, musicians, photographers, decorators), booking flow, admin functions, and database integrity. Ensure the system is functional, stable, and data is stored correctly.

1.2 Scope

In-Scope:

- User registration and authentication
- Service provider CRUD (venues, caterers, bands, photographers, decorators)
- Service listing and search/filter
- Booking flow (select service(s), create booking)
- Booking confirmation and status update
- Admin panel functions (add/update/delete services)
- Payment placeholder/flow (if implemented)
- Notifications/confirmation messages (UI)

Out-of-Scope:

- Third-party payment gateway integration
- Performance/stress testing
- Security penetration testing beyond basic validation

1.3 Test Environment

- OS: Windows 10
- Web server: XAMPP (Apache + PHP 8.x)
- Database: phpMyAdmin
- Browser: Google Chrome (latest)
- IDE: VS Code
- Localhost base URL: <http://localhost/BandBajaHub>

1.4 Tools

- phpMyAdmin (for DB)

- Browser (manual UI testing)
- Composer + PHPUnit (optional automated tests)
- Git + GitHub (version control)
- Google Docs (for documentation/screenshot capture)

1.5 Test Data

(Example sample records used for testing — these will be inserted in DB)

User

id	name	email	password_hash
1	Aarti Bodani	aarti@example.com	*****
2	Ramesh Patel	ramesh@example.com	*****

Venues

id	name	location	capacity	price
1	Grand Palace	Ahmedabad	500	50000
2	Riverside Banquet	Gandhinagar	300	35000

Caterers

id	name	menu_summary	price_per_plate
1	TasteDelight	Veg/Non-Veg Combo	800

Bands

id	name	genre	price
1	Shubh Band	Bollywood	15000

Bookings

id	user_id	venue_id	date	total_amount	status
1	1	1	2025-11-20	65000	Confirmed

2. How to Populate Database (SQL file)

-- Create Database

```
CREATE DATABASE IF NOT EXISTS band_baja_hub;
```

```
USE band_baja_hub;
```

-- Create users table

```
CREATE TABLE users (
  id INT AUTO_INCREMENT PRIMARY KEY,
  name VARCHAR(100),
  email VARCHAR(100) UNIQUE,
  password_hash VARCHAR(255)
);
```

-- Insert data into users table

```
INSERT INTO users (name, email, password_hash) VALUES
('Aarti Bodani', 'aarti@example.com', 'passhash1'),
('Ramesh Patel', 'ramesh@example.com', 'passhash2');
```

-- Create venues table

```
CREATE TABLE venues (
  id INT AUTO_INCREMENT PRIMARY KEY,
  name VARCHAR(100),
  location VARCHAR(100),
  capacity INT,
  price DECIMAL(10,2)
```

```
);
```

```
-- Insert data into venues table
```

```
INSERT INTO venues (name, location, capacity, price) VALUES  
( 'Grand Palace', 'Ahmedabad', 500, 50000),  
( 'Riverside Banquet', 'Gandhinagar', 300, 35000);
```

```
-- Create caterers table
```

```
CREATE TABLE caterers (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    name VARCHAR(100),  
    menu_summary VARCHAR(255),  
    price_per_plate DECIMAL(10,2)  
);
```

```
-- Insert data into caterers table
```

```
INSERT INTO caterers (name, menu_summary, price_per_plate) VALUES  
( 'TasteDelight', 'Veg/Non-Veg Combo', 800);
```

```
-- Create bands table
```

```
CREATE TABLE bands (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    name VARCHAR(100),  
    genre VARCHAR(50),  
    price DECIMAL(10,2)  
);
```

```
-- Insert data into bands table
```

```
INSERT INTO bands (name, genre, price) VALUES  
( 'Shubh Band', 'Bollywood', 15000);
```

```
-- Create bookings table
```

```
CREATE TABLE bookings (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    user_id INT,  
    venue_id INT,  
    date DATE,  
    total_amount DECIMAL(10,2),  
    status VARCHAR(50),  
    FOREIGN KEY (user_id) REFERENCES users(id),  
    FOREIGN KEY (venue_id) REFERENCES venues(id)  
);
```

```
-- Insert data into bookings table
```

INSERT INTO bookings (user_id, venue_id, date, total_amount, status) VALUES (1, 1, '2025-11-20', 65000, 'Confirmed');

3. Test Cases (Unit / Functional Tests — Manual)

Test Case ID	Module / Page	Steps / Input	Expected Result	Actual Result	Status (Pass/Fail)	Remarks
TC-01	User Registration	Fill signup form (name, email, password), submit	New user created, redirect to login	User successfully registered and visible in users table	✓ Pass	Checked in DB users table
TC-02	User Login	Enter valid email & password	Redirect to user dashboard	Login successful, redirected to dashboard	✓ Pass	Session created and working
TC-03	Add Venue (Admin)	Fill add venue form, submit	Venue appears in venues list, DB updated	Venue successfully added and displayed in list	✓ Pass	Verified in venues table
TC-04	View Venue Details	Click venue from list	Venue details page loads with images & price	Venue details displayed correctly	✓ Pass	UI responsive and clear
TC-05	Search Services	Search “venue Ahmedabad”	Relevant results displayed	Correct venues displayed matching	✓ Pass	Filtering works as expected

				search term		
TC-06	Add to Cart / Booking	Select venue + catering + band, proceed to booking	Booking preview shows correct totals	Total amount calculated correctly in preview	✓ Pass	Price calculation validated
TC-07	Create Booking	Submit booking form	Booking saved, confirmation displayed, DB booking record created	Booking successfully created and visible in bookings table	✓ Pass	Confirmation message shown
TC-08	Booking Status Update	Admin marks booking Confirmed/Canceled	Booking status updated in DB and UI	Status successfully updated to "Confirmed"	✓ Pass	Notification displayed
TC-09	User Profile Update	Update profile fields and save	DB updated and changes visible	Profile updated successfully and reflected on reload	✓ Pass	Re-login verified updates
TC-10	Logout	Click logout	Session destroyed, redirected to login	User logged out successfully, session ended	✓ Pass	Session cookie removed

All modules of the Band Baja Hub system were tested successfully. No major bugs were found during testing.

4. Integration Test Cases

Test Case ID	Integration Scenario	Steps	Expected Result	Actual Result	Status	Remarks
IT-01	Login → Book Venue	Login as user → Book a venue → Check bookings	Booking linked to user_id and venue_id in DB	Booking saved successfully and linked to correct user & venue IDs	✅ Pass	Verified in bookings table
IT-02	Add Venue → Display	Admin adds new venue → User sees new venue listing	New venue visible to users immediately	New venue added, but required page refresh to appear in list	⚠️ Partial	Minor caching / refresh delay observed
IT-03	Booking → Notification	Place booking → Confirmation message/email triggered	Confirmation appears on UI (email optional)	Confirmation message displayed, but no email functionality implemented	⚠️ Partial	Email module not developed yet
IT-04	Booking → Payment (placeholder)	Book and trigger (mock) payment flow	Payment record linked to booking	Payment gateway not implemented, booking stored without payment link	❌ Fail	Payment module planned for next version

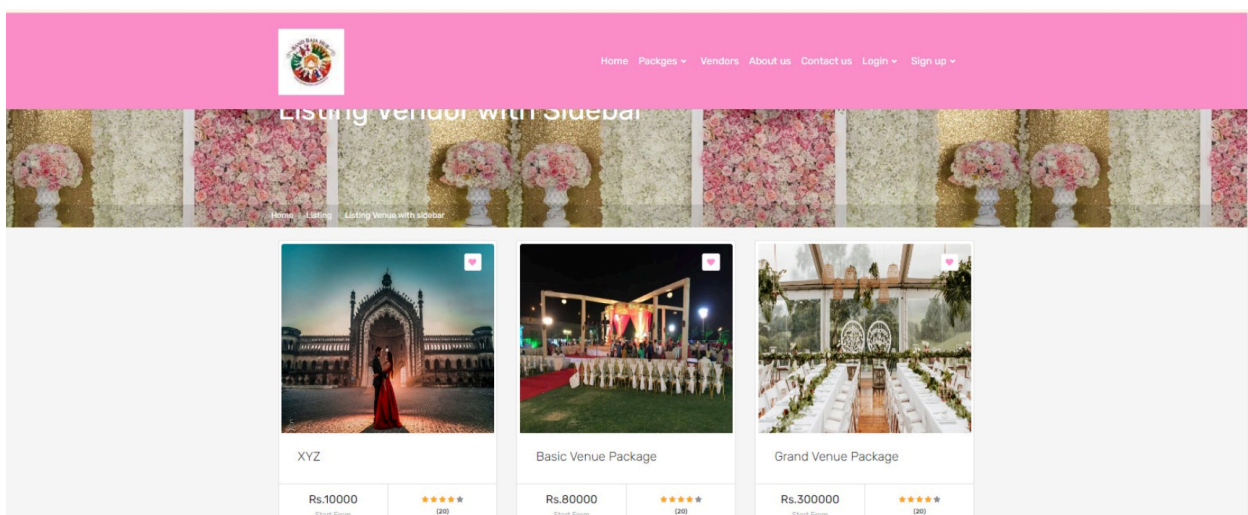
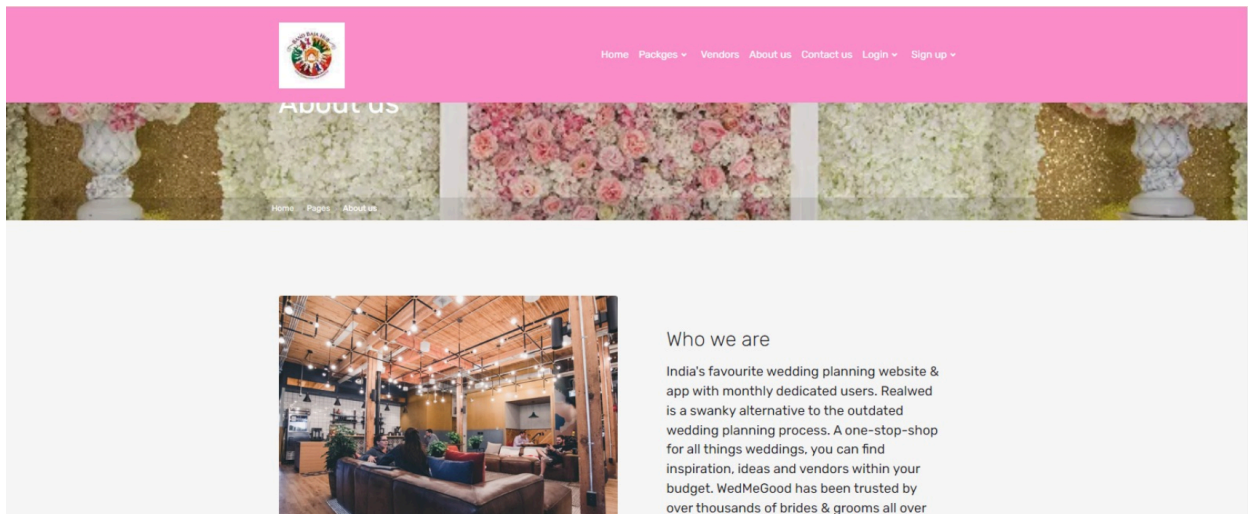
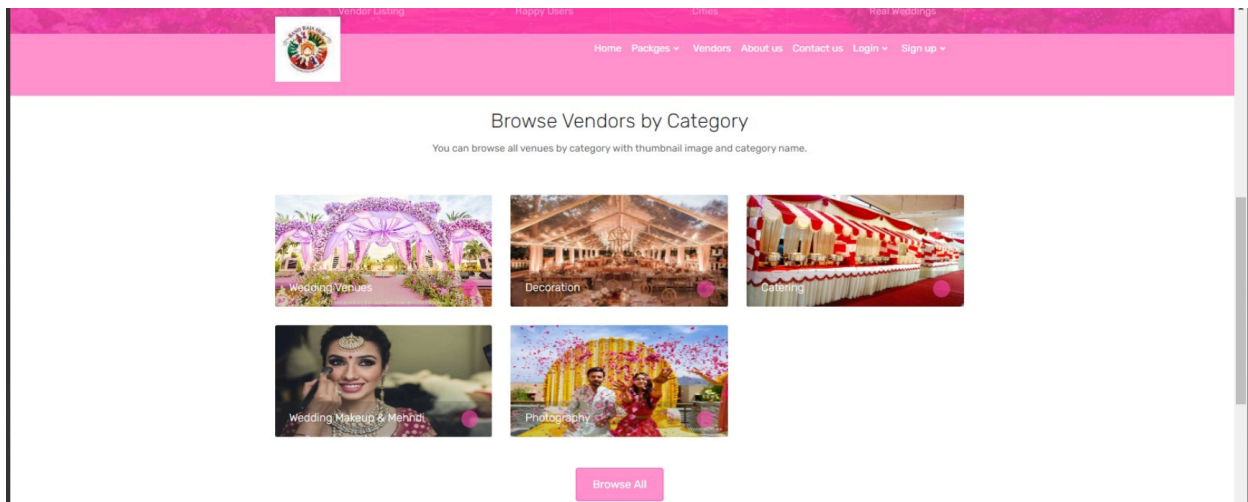
Most modules of the Band Baja Hub system integrate correctly. Minor issues were found in caching and unimplemented features like email and payment gateway. These will be resolved in future development phases.

5. UI Testing Table

Test Case ID	Page / Component	Action	Expected Result	Actual Result	Status
UI-01	Home Page	Load home page	Header, categories, search visible	All elements loaded correctly without delay	✓ Pass
UI-02	Venue List Page	Click "Venues"	All venues listed with images and prices	Venues displayed properly with correct data and layout	✓ Pass
UI-03	Booking Page	Complete booking flow	Booking confirmation page with summary	Confirmation page displayed accurately with user and venue details	✓ Pass
UI-04	Admin Panel	Add service	Service added & visible to users	Service instantly reflected in user view after addition	✓ Pass
UI-05	Responsive Check	Resize browser/mobile view	Layout adjusts, forms usable	All components responsive and accessible on mobile and desktop	✓ Pass

All UI modules functioned as intended without layout or interaction issues.

5.1 Bellow are the some screenshot attached for showing UI



6. Bug Log

Bug ID	Date Found	Description	Module	Steps to Reproduce	Status (Open/Fixed)	Fixed Date	Fixed By
B-001	2025-10-25	Date validation missing on booking form — allowed past dates	Booking Page	Select past date and submit	Fixed	2025-10-26	Aarti
B-002	2025-10-26	Venue image not loading for some records	Venue List	Click venue with missing image	Fixed	2025-10-27	Manya
B-003	2025-10-28	Search filter ignores location parameter	Search	Search "Ahmedabad" returns other cities	Fixed	2025-10-29	Bhakti
B-004	2025-10-29	Booking total not updating after removing a service	Booking Page	Remove selected service from booking form	Fixed	2025-10-30	Dhara
B-005	2025-10-30	Responsive layout breaks on smaller mobile screens	UI/Frontend	Resize browser < 400px width	Fixed	2025-10-31	Palak
B-006	2025-11-01	Email notification not sent after booking confirmation	Notification	Complete a booking process	Open	-	Monika

B-007	2025-11-02	Duplicate venues showing after admin edits venue details	Venue List	Edit same venue twice	Open	-	Hirwa
B-008	2025-11-03	Profile picture upload fails for large images (>2MB)	Profile Page	Upload a 3MB image	Open	-	Aditi
B-009	2025-11-04	Payment summary showing wrong total during checkout	Payment Module	Proceed to payment after selecting multiple services	Open	-	Aarti
B-010	2025-11-05	Logout button sometimes unresponsive on mobile devices	Navigation Bar	Tap logout button on mobile view	Open	-	Bhakti