**User Acceptance Testing**

|  |  |
| --- | --- |
| Date | 10-04-2024 |
| Team ID | SWTID1743353335 |
| Project Name | **BOOK A DOCTOR USING MERN** |
| Maximum Marks |  |

**Project Overview:**

**Project Name:** BOOK A DOCTOR USING MERN

**Project Description:**

Book a doctor is a web application built using the MERN stack (MongoDB, Express.js, React.js, Node.js). It allows patients to register, log in, search for doctors, and book appointments online. Doctors can manage their availability and view scheduled appointments. Admins oversee user management and approve doctor profiles. The app includes secure login and Stripe payment integration for confirmed bookings.

**Project Version:** "1.0.0"

**Testing Period:** 09-04-2025 to 10-04-2025

**Testing Scope:**

**List of Features and Functionalities to be Tested:**

* User Registration via Email and Gmail
* User Login and Role-based Dashboard Redirection
* Doctor Search and Appointment Booking by Patients
* Payment Integration and Confirmation via Stripe
* Doctor Availability Management
* Viewing Appointments (Doctor and Patient Dashboards)
* Admin Functions: Approving Doctors, Managing Users

**List of User Stories or Requirements to be Tested:**

* Email Registration
* Login Functionality
* Appointment Booking
* Payment Confirmation
* Doctor Availability Updates
* Doctor Appointment Views
* Admin Login
* Admin Approves Doctors
* Admin User Management

**Testing Environment:**

URL/Location: <https://localhost:5173/>

Credentials (if required):

**Doctor:**

User: [arjunmehta@gmail.com](mailto:arjunmehta@gmail.com)

Password: arjunmehta

**Patient:**

User: [user1@gmail.com](mailto:user1@gmail.com)

Password: user1

**Test Cases:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Case ID** | **Test Scenario** | **Test Steps** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| TC-001 | User Registration with Email | 1. Open Registration Page  2. Enter Name, Email, Password  3. Click Register | User is registered and redirected to confirmation message | User successfully registered | Pass |
| TC-002 | Login with Valid Credentials | 1. Open login page  2. Enter email and password  3. Click Login | User is redirected to the correct dashboard based on role | Correct dashboard shown | Pass |
| TC-003 | Book Appointment as Patient | 1. Login as Patient  2. Search for doctor  3. Select slot and click Book  4. Make Payment | Appointment is booked and payment confirmed | Booking confirmed | Pass |
| TC-004 | View Appointments as Doctor | 1. Login as Doctor  2. Navigate to "My Appointments" | List of upcoming appointments is displayed | Appointments listed | Pass |
| TC-005 | Admin Approves Doctor Registration | 1. Login as Admin  2. Go to pending doctor approvals  3. Approve a pending doctor | Doctor account status changes to active | Approval processed | Pass |

**Bug Tracking:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Bug ID | Bug Description | Steps to reproduce | Severity | Status | Additional feedback |
| BG-001 | Confirmation email not received | 1. Register as user 2. Wait for confirmation email | Medium | In Progress | Difficult in receiving emails due to SMTP misconfiguration and mail service throttling during peak requests. |
| BG-002 | Doctor dashboard not loading slots | 1. Login as doctor 2. Go to availability section 3. Click on date | High | In Progress | Possibly a database read delay |
| BG-003 | Payment API | 1. Open app on mobile 2. Click on booking  3. Click on payment | High | closed | The problem has been resolved by ensuring proper handling of asynchronous responses. |