User Acceptance Testing

Date	13 April 2025
Team ID	SWTID1743509015
Project Name	iMovies - Online Movie Ticket Booking System
Maximum Marks	

Project Overview:

Project Name: iMovies - Online Movie Ticket Booking System

Project Description: The iMovies project is a full-stack MERN (MongoDB, ExpressJS, ReactJS, NodeJS) application designed to provide users with a seamless movie ticket booking experience. It allows users to explore, book, and manage movie tickets online, while offering an admin panel to manage theaters, shows, and bookings.

Project Version: 1.0

Testing Period: 01/04/25 to 13/04/25

Testing Scope:

- User Registration & Login (User/Admin)
- Movie Listing and Categorization
- Show and Theater Selection
- Seat Booking and Availability Tracking
- Booking History
- Favorite Movies & Recommendations
- Admin Dashboard (Theater/Show/Movie Management)
- Role-based Access Control

User Stories or Requirements to be Tested:

- As a **user**, I want to browse movies and book tickets securely.
- As a **user**, I want to mark my favorite movies and see recommendations.
- As an **admin**, I want to manage theaters, shows, and bookings.
- As a **user/admin**, I want to login securely with hashed passwords.
- As a **user**, I want to view my booking history.

Testing Environment:

URL/Location: `http://localhost:3000` (Frontend), `http://localhost:5000` (Backend)

Credentials:

- **User**: testuser@gmail.com / Test@123
- **Admin**: admin@gmail.com / Admin@123

Test Cases:

Test Case ID	Test Scenario	Test Steps	Expected Result	Actual Result	Pass/Fail
TC-001	User Registration	1. Navigate to Register page 2. Fill form 3. Submit	User is registered and redirected to login	As expected	Pass
TC-002	User Login	1. Go to Login 2. Enter valid credentials 3. Submit	Redirected to homepage and token stored in cookies	As expected	Pass
TC-003	Browse Movies	1. Login 2. View home page	Movie cards with categories are displayed	As expected	Pass
TC-004	Movie Booking	1. Click on movie 2. Choose show 3. Select seats	Booking confirmed, stored in DB, and shown in booking	As expected	Pass

		4. Book	history		
TC-005	Add to Favorites	1. Click favorite icon on movie card	Movie added to favorites list	As expected	Pass
TC-006	View Booking History	1. Go to booking history	List of previous bookings with movie and show details shown	As expected	Pass
TC-007	Admin Login	1. Use admin credentials 2. Login	Redirected to admin dashboard	As expected	Pass
TC-008	Admin Add Theater	1. Go to Theater tab 2. Add new theater	Theater added to DB and reflected in dropdowns	As expected	Pass
TC-009	Admin Add Movie and Show	1. Fill movie details 2. Add show for selected theater	Movie and show added, shown to users	As expected	Pass
TC-010	Role-based Access	1. User trying to access admin route	Access denied or redirected	As expected	Pass

| TC-011 | Seat Availability Update | 1. Book seats

2. Refresh and view available seats | Booked seats marked as unavailable | As expected | Pass |

Tracking of Bug:

Bug ID	Bug Description	Steps to Reproduce	Severity	Status	Additional Feedback
BG-001	Favorites not updating immediately	1. Add to favorites 2. Reload page	Medium	Closed	Resolved using useEffect to re-fetch
BG-002	Booking fails if showtime is missing	1. Admin adds movie without show time 2. User tries to book	High	Closed	Validation added for show time
BG-003	Admin dashboard slow on first load	1. Login as admin 2. Navigate to dashboard	Low	In Progress	Consider using pagination/lazy loading
BG-004	Login token expires abruptly	1. Login 2. Wait 5-10 minutes 3. Perform an action	High	Closed	Increased token expiry time
BG-005	No error message on incorrect credentials	1. Login with wrong credentials	Medium	Closed	Error toast message added

Sign-off:

Tester Name: Bhaskar Rai

Date: 13/04/25

Signature: Bhaskar

Notes:

- Ensure that all test cases cover both positive and negative scenarios.
- Encourage testers to provide detailed feedback, including any suggestions for improvement.
- Bug tracking should include details such as severity, status, and steps to reproduce.
- Obtain sign-off from both the project manager and product owner before proceeding with deployment.