Exploratory Test Plan (Test Cases)

1. *Functional Test*

Test Case: Request and Decline Booking

This test checks if the system blocks a second booking after the first one is approved. One user books and gets approved, and then another tries to book the same date. Should be rejected if availability logic works right.

1 - Log in as User A, list space

2 - Log in as User B, request space

3 - Log in as User A, accepts request from User B

4 - Log in as User C, tries to book same space

5 - The system blocks User C's request

Expected Outcome: Booking allowed for User B, space was still available. User C cannot book same space - the system disallows it as it has already been booked.

---

2. *Non Functional Test*

Test Case: Load Test – Booking Spike

Test to see how the system holds up when multiple users hit the booking feature at the same time. This test simulates a small traffic spike and checks for slowdowns, crashes, or failures under pressure.

1 - Simulate 50+ users logging in (Manual or Automated)

2 - Each user attempts to book different available spaces at the same time

3 - Monitor system response to see if it crashes or delivers any other unexpected result/s.

Expected Outcome: System handles load without crashing

---

3. *Functional Test*

Test Case: Invalid Space Listing

Tests to see how the 'List a New Space' page deals with invalid space listings, covering incomplete fields and invalid data type entries. It is expected that appropriate error messages cover these actions.

1 – Attempt to create a new listing, leaving all fields blank.

2 – Attempt to create a new listing, leaving only 'Description' blank.

3 – Attempt to create a new listing, with 'Price' input as string.

4 – Attempt to create a new listing, with dates available formatted incorrectly.

Expected Outcome: Each scenario should generate an error message prompting the user to complete the field correctly.

4. *Non Functional Test*

Test Case: Security – Invalid Sign-up Attempt

Tests to check that the system can validate email addresses and passwords.

1 – Sign up for an account using an invalid email address, e.g. example.simple.com

2 – Sign up for an account using an invalid password (according to site's validity requirements)

3 – Sign up for an account using an email address already registered

Expected Outcome: The user should receive error messages asking them to complete the fields correctly.