|  |  |
| --- | --- |
| **Project:** Moffat Bay Lodge – Group 3 |  |
| **Course:** CSD460 – Capstone in Software Development |  |
| **Description:** Web Site for the Marina Bay Lodge |  |
| **Date:** 09/04/2024 |  |

To complete the test case plan, fill out the information for Project, Course, Description and Date in the header below.

.

**TABLE OF CONTENTS**

[Landing Page Layout and Navigation Functionality 2](#_Toc1586041572)

[Link Functionality 2](#_Toc1240220851)

[Login Success 3](#_Toc1817378862)

[Login Error Handling 4](#_Toc370477294)

[Registration Success 4](#_Toc1219144704)

[Registration Error Handling 5](#_Toc1947499850)

[Reservation Form Display and Field Validation 6](#_Toc813616698)

[Reservation Process with Correct Data 7](#_Toc771400751)

[Invalid Room Selection Handling 8](#_Toc1661515035)

[Form Validation 9](#_Toc2084535007)

[Successful Login Redirects to My Reservations 10](#_Toc1132937585)

[Anonymous User Cannot Access “My Reservations” or “Make a Reservation” 11](#_Toc1482457202)

[User Logout of Session 11](#_Toc891891166)

For each test, the developer should: provide a test description, a test objective, the developer name and date tested. For each step, fill out the actions to be taken and describe the expected results, and indicate yes or no for each step depending upon the result. The peer tester should provide their name, date tested, indicate yes or no for each step depending upon the result, and a screenshot (thumbnail) of the result.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 1 | **Landing Page Layout and Navigation Functionality** | | | |
|  | **Test Objective:** Verify that the landing page displays all key elements correctly and the navigation bar functions as expected | **Developer:** Brandon  **Date tested:** <yyyy/mm/dd> | **Peer tester:**  **Date tested:** <yyyy/mm/dd> | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail**  **+ Screenshot** |
| 1 | Open the landing page in a browser | The page loads with the correct background image | <yes/no> | <yes/no> |
| 2 | Verify that the logo is centered in the navigation bar | Logo is centered and visible between the menu items | <yes/no> | <yes/no> |
| 3 | Click on the "Home" menu link | Page reloads or stays on the current page (no errors) | <yes/no> | <yes/no> |
| **Comments** | The landing page should visually resemble the provided mockup. Any misalignment or missing images should be noted. The background image and font settings must be checked for accuracy. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 2 | **Link Functionality** | | | |
|  | **Test Objective:** Verify that all the links in the navigation bar and buttons function correctly and lead to the appropriate pages or placeholders. | **Developer:** Brandon  **Date tested:** <yyyy/mm/dd> | **Peer tester:**  **Date tested:** <yyyy/mm/dd> | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | Click on the "Home" link in the navigation bar | Reloads the current landing page without errors | <yes/no> | <yes/no> |
| 2 | |  | | --- | | Click on the "About Us" link in the navigation bar |  |  | | --- | |  | | Leads to the About Us page or a placeholder page | <yes/no> | <yes/no> |
| 3 | Click on the "Attractions" link in the navigation bar | Leads to the Attractions page or a placeholder page | <yes/no> | <yes/no> |
| 4 | Click on the "Lodging" link in the navigation bar | Leads to the Lodging page or a placeholder page | <yes/no> | <yes/no> |
| 5 | Click on the "Make a Reservation" link in the navigation bar | Leads to the Make a Reservation page or a placeholder page | <yes/no> | <yes/no> |
| 6 | Click on the "My Reservations" link | Leads to the My Reservations page or a placeholder page | <yes/no> | <yes/no> |
| 7 | Click on the "Login/Register" button | Leads to the Login/Register page or a placeholder page | <yes/no> | <yes/no> |
| 8 | |  | | --- | | Click on the "Learn More" button in the hero section |  |  | | --- | |  | | Leads to the specified page or a placeholder page | <yes/no> | <yes/no> |
| **Comments** | All links should either function correctly or lead to placeholders if the pages are not yet created. If any links do not work or lead to 404 errors, they should be checked for typos or broken paths. Links should open without issues, and the URLs should match the expected destinations. | | | |

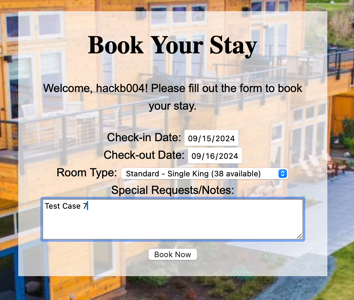
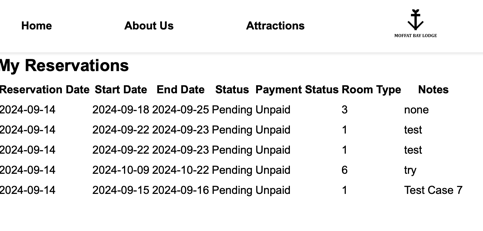
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 3 | **Login Success** | | | |
|  | **Test Objective:** Connects to MySQL database. Allows and verifies user input. Properly redirect user to reservation page after successful login. | **Developer:** Alexis  **Date tested:** <yyyy/mm/dd> | **Peer tester:**  **Date tested:** <yyyy/mm/dd> | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | User input correct username | Proceed to type in password | YES | <yes/no> |
| 2 | User input correct password | Proceed to click login button | YES | <yes/no> |
| 3 | Click “Log in” button | Login successful and user is redirected to registration page. | YES | <yes/no> |
| **Comments** | User should be able to type in the corresponding text boxes. In the case that the input is all correct, user is then redirected to the reservations page. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 4 | **Login Error Handling** | | | |
|  | **Test Objective:** Connects to MySQL database. Allows and verifies user input. Redirects user to refreshed login page with a note explaining the issue. | **Developer:** Alexis  **Date tested:** <yyyy/mm/dd> | **Peer tester:**  **Date tested:** <yyyy/mm/dd> | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | User input incorrect username but correct password, click “login” button | Page refreshes with “user not found” | YES | <yes/no> |
| 2 | User input correct username but incorrect password, click “login” button | Page refreshes with “invalid password” | YES | <yes/no> |
| 3 | Click “Log in” button without input | Popup bubble “Please fill out this field” | YES | <yes/no> |
| **Comments** | User should be able to type in the corresponding text boxes. There are many cases in which unsuccessful login may occur, including: no user input, wrong username and correct password, correct username and wrong password. The page will prompt if there are blank text boxes that must be filled before proceeding to the next page. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 5 | **Registration Success** | | | |
|  | **Test Objective:** Connects to MySQL database. Allows and verifies user input. Properly inserts user input into database. | **Developer:** Alexis  **Date tested:** <yyyy/mm/dd> | **Peer tester:**  **Date tested:** <yyyy/mm/dd> | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | User input user info such as name, email, phone, address, city, state, zip code, country | Proceed to username creation | YES | <yes/no> |
| 2 | User input username at least 6 characters long | Proceed to create password | YES | <yes/no> |
| 3 | User input password at least 8 characters in length with at least 1 numerical value. | Proceed to click register | YES | <yes/no> |
| 4 | User clicks “Register” button | User information successfully stored into database, and user is prompted to Login page. | YES | <yes/no> |
| **Comments** | The user should be able to type in their information smoothly and click “Register Now” to submit this information into the database. Backend code should effectively enter this information into the database and the user should be redirected to the Login page and able to login with their created username and password. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 6 | **Registration Error Handling** | | | |
|  | **Test Objective:** Connects to MySQL database. Allows and verifies user input. Prompts messages wherever necessary for incorrect user input. | **Developer:** Alexis  **Date tested:** <yyyy/mm/dd> | **Peer tester:**  **Date tested:** <yyyy/mm/dd> | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | User input blank | Prompts message bubble “Please fill out this field” | YES | <yes/no> |
| 2 | User input correct Name, Email, Address, State, City, Zip, and Country but input phone number less than 10 digits long or non-numerical values. | User is prompted to match requested format “Phone number must be exactly 10 digits” | YES |  |
| 3 | User input correct information: Name, Email, Phone, Address, City, State, Zip Code, Country | Proceed to create username | YES | <yes/no> |
| 4 | User input username less than 6 characters in length and password 8 characters long with at least 1 number | User is prompted to match requested format of at least 6 characters long. | YES | <yes/no> |
| 5 | User input username at least 6 characters in length and password less than 8 characters long | User is prompted that the password should be at least 8 characters long and include at least 1 numerical value. | YES | <yes/no> |
| 6 | User input username at least 6 characters long and password 8 characters long but no number | User is prompted that the password should be at least 8 characters long and include at least 1 numerical value. | YES | <yes/no> |
| **Comments** | As there is a high margin of error when it comes to user input, there should be plenty of measures to account for these errors. Users should not be able to proceed to the next step (registering their account) without having filled out all the required information and should therefore be prompted when any information is missing. Furthermore, to keep the database consistent, there are restrictions on phone number entry (10 digits, no special characters). Lastly, there are minimum requirements for username and password for basic security. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 7 | **Reservation Form Display and Field Validation** | | | |
|  | **Test Objective:** Verify that the reservation form fields are displayed correctly and require user input before submission. | **Developer: Brandon**  **Date tested:** 09/14/2024 | **Peer tester:**  **Date tested:** <yyyy/mm/dd> | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | |  | | --- | | Open the Make a Reservation page in a browser | | The page loads with the correct banner and reservation form | YES | <yes/no> |
| 2 | Leave all form fields empty and click "Book Now" | A validation message prompts the user to fill out the fields | YES |  |
| 3 | Enter valid check-in, check-out dates, and room selection, and submit | Form is submitted without validation errors | YES | <yes/no> |
| **Comments** | The reservation form should be fully visible with all necessary fields. Validation must work properly, preventing submission when required fields are missing. | | | |



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 8 | **Reservation Process with Correct Data** | | | |
|  | **Test Objective:** Verify that a reservation is successfully submitted when all the required data is entered correctly. | **Developer: Brandon**  **Date tested:** 09/14/2024 | **Peer tester:**  **Date tested:** <yyyy/mm/dd> | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | |  | | --- | | Enter valid check-in, check-out dates, select a room, and click "Book Now" | | Reservation is successfully submitted and stored in the database | YES | <yes/no> |
| 2 | Verify the data in the Reservations table in the database | The new reservation record appears with the correct details | YES |  |
| 3 | The user is redirected to my\_reservations.php | Redirection happens without errors | YES | <yes/no> |
| **Comments** | The reservation form should be fully visible with all necessary fields. Validation must work properly, preventing submission when required fields are missing. | | | |

A screenshot of a hotel

Description automatically generatedA screenshot of a computer

Description automatically generatedA screenshot of a phone

Description automatically generated

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 9 | **Invalid Room Selection Handling** | | | |
|  | **Test Objective:** Verify that the system handles cases where a user selects a room that is no longer available (e.g., 0 available rooms). | **Developer: Brandon**  **Date tested:** 09/14/2024 | **Peer tester:**  **Date tested:** <yyyy/mm/dd> | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | |  | | --- | | Select a room with 0 available rooms | | A validation message appears informing the user of the issue | YES | <yes/no> |
| 2 | Verify that the user cannot proceed with the reservation | The form is not submitted, and the reservation is not recorded | YES |  |
| **Comments** | The reservation form should be fully visible with all necessary fields. Validation must work properly, preventing submission when required fields are missing. | | | |

A screenshot of a hotel

Description automatically generated

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 10 | **Form Validation** | | | |
|  | **Test Objective:** Verify that the form on the **room\_reservation.php** page validates all fields correctly without needing to submit data to the database. This test will focus on client-side form validation. | **Developer: Brandon**  **Date tested:** 09/14/2024 | **Peer tester:**  **Date tested:** <yyyy/mm/dd> | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | |  | | --- | | Leave the **check-in** and **check-out** fields empty and try to submit | | Form does not submit, and validation error appears for the required fields | YES | <yes/no> |
| 2 | Enter a check-out date earlier than the check-in date and try to submit | Form does not submit, and a validation error appears for the date fields | YES |  |
| 3 | Enter valid dates but leave the **room type** unselected and try to submit | Form does not submit, and validation error appears for the room type field | NO – picks first option |  |
| **Comments** | This test ensures that the **room\_reservation.php** form has proper client-side validation, preventing submission when required fields are empty, invalid dates are entered, or the room type is not selected. No database interaction is involved, and errors should appear directly on the form. | | | |

A screenshot of a hotel

Description automatically generatedA white background with black text

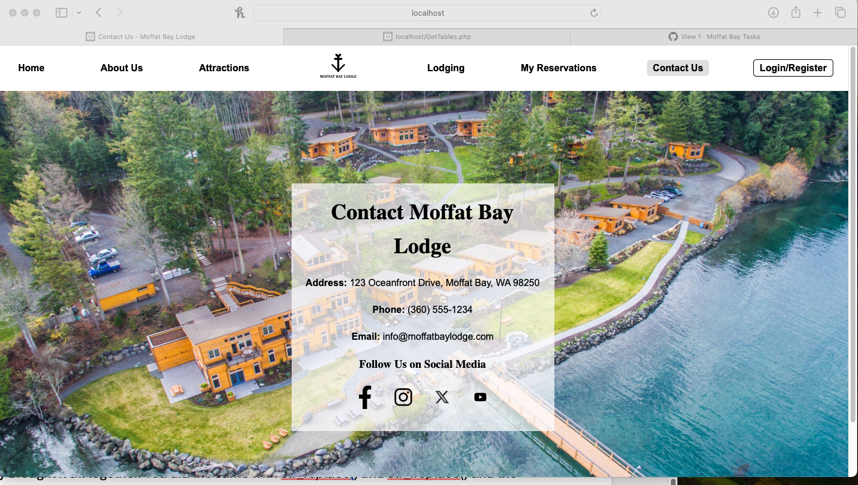
Description automatically generated

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 10 | **Successful Login Redirects to My Reservations** | | | |
|  | **Test Objective:** Verify that the system can validate a user’s login session and properly redirect them to the reservation page. The page should properly display any reservations they have made. | **Developer: Alexis Yang**  **Date tested:** 09/15/2024 | **Peer tester:**  **Date tested:** <yyyy/mm/dd> | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | User logs into their account using the correct username and password. | User is redirected to the My Reservation page upon successful login. | YES | <yes/no> |
| 2 | User has not made any reservations. | The page shows that there are no reservations found. | YES |  |
| 3 | User has made a reservation already and wishes to view it. | The page properly shows a table with their reservation details. | YES |  |
| **Comments** | The My Reservations page should only open when a user is logged in. When the user is logged in, they are properly redirected to the page to view the details of their reservation(s), otherwise they are shown that there are no reservations. | | | |

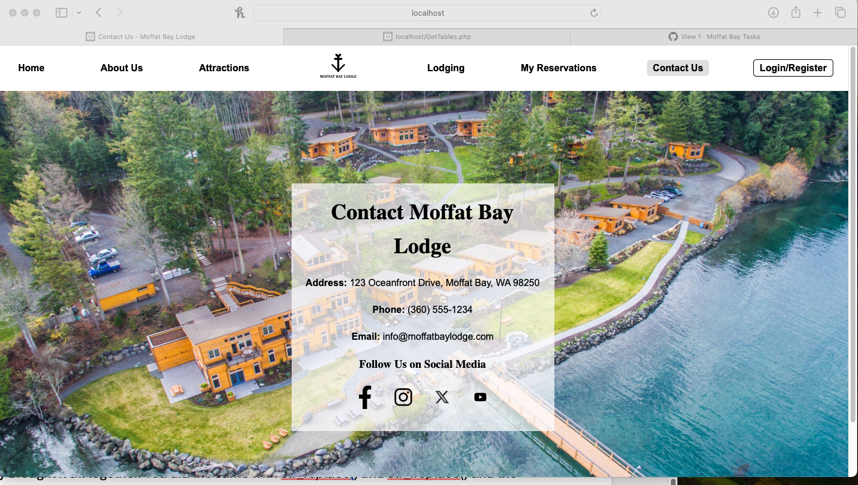
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 11 | **Anonymous User Cannot Access “My Reservations” or “Make a Reservation”** | | | |
|  | **Test Objective:** my\_reservations.php should only be accessed by registered users. In the case of anonymous users, they are redirected to the login page and have the option to register. | **Developer: Alexis Yang**  **Date tested:** 09/15/2024 | **Peer tester:**  **Date tested:** <yyyy/mm/dd> | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | Anonymous user clicks on “My Reservations” link in the nav bar. | User is redirected to the Login page. | YES | <yes/no> |
| 2 | User has not made any reservations. | The page shows that there are no reservations found. | YES |  |
| 3 | User has made a reservation already and wishes to view it. | The page properly shows a table with their reservation details. | YES |  |
| **Comments** | The My Reservations page should only open when a user is logged in. When the user is logged in, they are properly redirected to the page to view the details of their reservation(s), otherwise they are shown that there are no reservations. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 12 | **User Logout of Session** | | | |
|  | **Test Objective:** Users should be able to log out of their session properly to preserve the integrity of the website. | **Developer: Alexis Yang**  **Date tested:** 09/15/2024 | **Peer tester:**  **Date tested:** <yyyy/mm/dd> | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | Logged in user clicks Logout button that can be seen on the two dynamic pages “Make a Reservation” and “My Reservations.” | Backend process, system is redirected to logout.php, ending the session. Frontend, user is redirected to the login page. | YES | <yes/no> |
| **Comments** | Users should be allowed to log out of their session to prevent outsiders from making changes to their account without their knowledge and preserving the website’s integrity. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 13 | **Contact Us Page Information Display** | | | |
|  | **Test Objective:** Verify that the contact information is displayed correctly on the Contact Us page. | **Developer: Brandon Hackett**  **Date tested:** 09/22/2024 | **Peer tester:**  **Date tested:** <yyyy/mm/dd> | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | Open the Contact Us page in a browser | The page loads with the correct contact information (address, phone, email) | YES | <yes/no> |
| 2 | Check the social media icons for Facebook, Instagram, X (Twitter), and YouTube | Icons are correctly displayed for each platform | YES | <yes/no> |
| 3 | Click on each social media icon | The link opens the correct platform (if the link is available) | TBD | <yes/no> |
| **Comments** | Ensure that all contact details are correct and the social media links work as expected. | | | |



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 14 | **Contact Us Page Layout** | | | |
|  | **Test Objective:** Verify that the layout and design elements of the Contact Us page match the specifications. | **Developer: Brandon Hackett**  **Date tested:** 09/22/2024 | **Peer tester:**  **Date tested:** <yyyy/mm/dd> | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | Open the Contact Us page in a browser | The page layout matches the specified design, with the hero section at the top and social media icons below | <yes/no> | <yes/no> |
| 2 | Ensure the hero section contains the title and contact details in a readable format | The title, address, phone number, and email are properly aligned and easy to read | <yes/no> | <yes/no> |
|  |  |  |  |  |
| **Comments** | Verify the layout matches the expected design and that text and icons are properly aligned and formatted. | | | |



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 15 | **Reservation Lookup with Invalid Data** | | | |
|  | **Test Objective:** Verify that the reservation lookup feature correctly handles invalid data (e.g., non-existent reservation ID or incorrect email). | **Developer: Eclevia**  **Date tested:** <yyyy/mm/dd> | **Peer tester: Brandon**  **Date tested:** 2024/09/22 | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | Enter an invalid reservation ID or email in the form and submit | The page displays a "Reservation not found" message | <yes/no> | YES |
| 2 | Check that no data is displayed when using incorrect inputs | No reservation details are shown when invalid data is entered | <yes/no> | YES |
| **Comments** | Verify the layout matches the expected design and that text and icons are properly aligned and formatted. | | | |

A screenshot of a reservation form

Description automatically generated

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test 16 | **Reservation Lookup with Valid Data** | | | |
|  | **Test Objective:** Verify that the reservation lookup feature retrieves and displays the correct reservation information for valid inputs. | **Developer: Eclevia**  **Date tested:** <yyyy/mm/dd> | **Peer tester: Brandon**  **Date tested:** 2024/09/22 | |
| **Step** | **Action** | **Expected results:** | **Developer pass/fail** | **Tester pass/fail** |
| 1 | Enter a valid reservation ID or email in the form and submit | The correct reservation details are displayed, including name, email, room type, and dates | <yes/no> | YES |
| 2 | Verify the displayed reservation details are accurate based on the database | All details (e.g., reservation ID, check-in/check-out dates) match the stored data | <yes/no> | YES |
| **Comments** | Verify the layout matches the expected design and that text and icons are properly aligned and formatted. | | | |

**A screenshot of a registration form

Description automatically generated**

