# PHP Travels

## Master Test Plan for PHP Travels Application

**Test Plan Identifier: MTP\_PHPTravels\_2025\_v1**

1. Introduction:  
This is the main plan for testing the PHP Travels website. The goal is to check all the main parts of the site like booking Flights, Hotels, Cars, and Tours, as well as Creating an Account and Paying to make sure they all work correctly on their own.

2. Test Items: Customer Registration, agent registration, agent & customer Login, Flight Search & Booking, Hotel Search & Booking, Car Rental Search & Booking, Tour Search & Booking, Payment Processing, Profile view and update, Homepage Features (Featured Flights, Featured Hotels, Popular Tours) and Post booking actions.

3. Approach:

Modular Testing: Each major module will have a dedicated, detailed test plan and test suite.

Automation Focus: A hybrid Selenium WebDriver + TestNG + Cucumber (BDD) + Page Object Model (POM) framework will be used for automating regression tests for all modules.

Integration Testing: Specific test cycles will be dedicated to testing the flow between modules (e.g., searching for a hotel, then adding a car, and making a single payment).

4. Entry & Exit Criteria:

Entry Criteria: A stable test build is deployed to the QA environment; test data is prepared.

Exit Criteria: All test cases for a planned cycle have been executed; all critical bugs are fixed; a Test Summary Report is generated.

5. Suspension Criteria:

6. Test Deliverables:

Master Test Plan

Individual Module Test Plans

Test Cases & Automation Scripts for each module

Test Data

Defect Reports

Test Summary Report

7. Environmental Needs:  
Windows/macOS, Chrome/Firefox/Edge browsers, PHP Travels QA Environment URL.

8.Responsibilities:

Sai Bavesh: Create Testcases, Execute test cases, develop automation scripts, Defect Reports, Test Summary Report.

9.Schedule: A consolidated schedule will map out the development and testing cycles for each module to ensure resources are allocated efficiently without overlap.

10.Approvals:

Vaishali

Project Manager

## Individual Module Test Plans:

**1. Test Plan: Customer Registration**

Test Plan Identifier: TP\_PHPTravels\_Customer\_Registration\_2025\_v1

Introduction:  
This document details the testing strategy for the Customer Registration module. The objective is to validate that new customers can successfully create an account with valid data and that appropriate, clear error messages are displayed for all invalid input scenarios.

Test Items:  
My Account dropdown, Sign Up link, First Name field, Last Name field, Mobile Number field, Email Address field, Password field, Confirm Password field, Sign Up button, Success Message Alert, Error Message Alerts.

Features to be Tested:

Successful account creation with valid data.

Field-level validation for all input fields (required field checks).

Email field validation (format, uniqueness).

Mobile Number validation (format, length).

Password strength validation (complexity rules).

Password and Confirm Password matching logic.

Visibility and clarity of error messages.

Features not to be Tested:

The Login functionality (covered in the Sign In test plan).

The "Forgot Password" flow.

Email verification link functionality (only the successful registration trigger is tested).

Approach:  
Automation with Selenium WebDriver, TestNG, and Cucumber BDD. Data-driven tests will be used for multiple valid and invalid input combinations stored in an Excel file.

Item Pass/Fail Criteria:

Pass: Valid registration completes successfully; all invalid registration attempts produce the correct, specific error message.

Fail: A valid registration is blocked; an invalid registration is accepted; an incorrect or generic error message is displayed.

Suspension Criteria :

Test Deliverables:  
Test cases, Bug reports, Test summary, Test reports , Test Plan , Test Analysis Report

Environmental Needs:  
QA Environment, Chrome/Firefox/Edge browsers.

Responsibilities:

Sai Bavesh: Create Testcases, Execute test cases, develop automation scripts, Defect Reports, Test Summary Report.

Approvals:

Vaishali

Project Manager

**2. Test Plan: Agent Registration**

Test Plan Identifier: TP\_PHPTravels\_Agent\_Registration\_2025\_v1

Introduction:  
This plan outlines the testing for the Agent Registration module, which is typically a separate flow from customer sign-up. The focus is on validating the business-specific data required for agent accounts and ensuring proper validation.

Test Items:  
Agent Sign Up link, Company Name field, Email Address field, Mobile Number field, Password field, Confirm Password field, IATA/ARC field (if applicable), Address fields, Sign Up button, Success Message, Error Messages.

Features to be Tested:

Successful agent account creation with valid company and personal details.

Validation of business-specific fields (e.g., Company Name, IATA number format).

Email and mobile number validation.

Password strength and matching validation.

Error handling for invalid data.

Features not to be Tested

The customer registration flow.

Approach:  
Automation similar to customer registration, but with a separate Page Object Class and test data file specifically for agent registration fields.

Item Pass/Fail Criteria:

Pass: Valid agent registration is successful; invalid data is rejected with correct errors.

Fail: Valid agent data is rejected; invalid data is accepted.

Suspension Criteria :

Test Deliverables:  
Test cases, Bug reports, Test summary, Test reports , Test Plan , Test Analysis Report

Environmental Needs:  
QA Environment, Chrome/Firefox/Edge.

Responsibilities:

Sai Bavesh: Create Testcases, Execute test cases, develop automation scripts, Defect Reports, Test Summary Report.

Approvals:

Vaishali

Project Manager

**3. Test Plan: Sign In (Customer & Agent)**

Test Plan Identifier: TP\_PHPTravels\_SignIn\_2025\_v1

Introduction:  
This document covers the testing of the unified Sign In module, which authenticates both Customers and Agents. The goal is to verify that valid credentials grant access to the correct dashboard and that invalid credentials are handled securely.

Test Items:  
My Account dropdown, Login link, Email Address field, Password field, Login button, Remember Me checkbox, Forgot Password link, Customer Dashboard, Agent Dashboard, Error Message Alert, Logout button.

Features to be Tested:

Customer Login: Successful sign-in with valid customer credentials redirects to the Customer Dashboard.

Agent Login: Successful sign-in with valid agent credentials redirects to the Agent Dashboard/Portal.

Error Handling: Sign-in attempts with invalid email formats, incorrect passwords, and non-existent accounts produce appropriate, non-specific error messages (e.g., "Invalid login credentials").

Session: "Remember Me" functionality .

Navigation: The Forgot Password link redirects to the correct page.

Access Control: A logged-in user cannot access the login page without first logging out.

Logout: Session is terminated correctly upon logout.

Features not to be Tested:

The functionality of the Forgot Password flow itself.

The features within the Customer or Agent dashboards (only the redirection is tested).

Approach:  
Automated tests for the main sign-in scenarios. Data-driven tests will use credentials for both user types from a test data file.

Item Pass/Fail Criteria:

Pass: Valid credentials grant access to the correct dashboard; invalid credentials trigger an error; logout functions correctly.

Fail: Valid credentials are rejected; invalid credentials are accepted; user is redirected to the wrong dashboard.

Suspension Criteria :

Test Deliverables:  
Test cases, Bug reports, Test summary, Test reports , Test Plan , Test Analysis Report

Environmental Needs:  
QA Environment, Chrome/Firefox/Edge.

Responsibilities:

Sai Bavesh: Create Testcases, Execute test cases, develop automation scripts, Defect Reports, Test Summary Report.

Approvals:

Vaishali

Project Manager

**4. Test Plan: Hotel Search & Booking**

Test Plan Identifier: TP\_PHPTravels\_Hotels\_2025\_v1

Introduction:  
To validate the search, filtering, and booking flow for hotel accommodations, ensuring correct information is displayed and selected.

Test Items:  
Hotels Tab, Search Location field, Check-In/Check-Out date pickers, Guest/Room selector, Search button, Search Results, Filters (Star Rating, Amenities, Price), Hotel Details page, Room Selection, Book Now button.

Features to be Tested:

Location search and auto-suggestion.

Date range validation.

Guest/Room selection logic.

Functionality of all filters.

Accuracy of hotel information and room types displayed.

Payment processing.

Features not to be Tested:

How user/Booking details stored in the backend of website

Time taken for retrieval of content from backend.

Actual room inventory management.

Approach:  
We use a combination of automated and manual testing for the search feature. The automated test suite handles the functional regression checks to ensure core capabilities, and we also perform manual testing to verify the accuracy of the search results and to assess the visual layout and user experience.

Item Pass/Fail Criteria:

Pass: Correct hotels are shown for a location; filters narrow results appropriately; user can select a room.

Fail: No results on valid search; filters do not work; incorrect room details shown.

Suspension Criteria:

Test Deliverables:  
Test cases, Bug reports, Test summary, Test reports , Test Plan , Test Analysis Report

Environmental Needs:  
QA Environment, Chrome/Firefox/Edge.

Responsibilities:

Sai Bavesh: Create Testcases, Execute test cases, develop automation scripts, Defect Reports, Test Summary Report.

Approvals:

Vaishali

Project Manager

**5. Test Plan: Flight Search & Booking**

Test Plan Identifier: TP\_PHPTravels\_Flights\_2025\_v1

Introduction:  
To verify the end-to-end functionality of searching for flights, applying filters, selecting options, and completing the booking process through to payment initiation.

Test Items:  
Flights Tab, Trip Type selectors (One-way, Round-trip), Flying From field, Destination To field, Departure/Return Date pickers, Passenger selector, Search button, Search Results page, Filters (Stops, Airlines, Price), Book Now button, Flight Details page.

Features to be Tested:

Functionality of all trip types.

Accuracy of airport auto-suggestion.

Date validation (past dates, invalid ranges).

Passenger number validation (max limits).

Accuracy of search results and filters.

Navigation to booking page.

payment processing.

Real-time flight availability.

Features not to be Tested:

How user/Booking details stored in the backend of website

Time taken for retrieval of content from backend.

Approach:  
We use a combination of automated and manual testing for the search feature. The automated test suite handles the functional regression checks to ensure core capabilities, and we also perform manual testing to verify the accuracy of the search results and to assess the visual layout and user experience.

Item Pass/Fail Criteria:

Pass: Searches return results based on inputs; filters work correctly; user can proceed to booking.

Fail: Searches fail with valid data; filters break; Book Now button is non-functional.

Suspension Criteria :

Test Deliverables:  
Test cases, Bug reports, Test summary, Test reports , Test Plan , Test Analysis Report

Environmental Needs:  
QA Environment, Chrome/Firefox/Edge.

Responsibilities:

Sai Bavesh: Create Testcases, Execute test cases, develop automation scripts, Defect Reports, Test Summary Report.

Approvals:

Vaishali

Project Manager

**6. Test Plan: Car Rental Search & Booking**

Test Plan Identifier: TP\_PHPTravels\_Cars\_2025\_v1

Introduction:  
To verify users can search for rental cars based on location, date, and time, and successfully select a vehicle.

Test Items:  
Cars Tab, Pick-Up Location field, Drop-Off Location selector, Pick-Up/Drop-Off Date & Time pickers, Search button, Search Results, Car Options, Book Now button.

Features to be Tested:

Pick-up location search.

Logic for same vs. different drop-off location.

Date and time validation.

Display of car types and pricing.

Payment processing.

Features not to be Tested:

How user/Booking details stored in the backend of website

Time taken for retrieval of content from backend.

Insurance policy details.

Approach:  
Similar to Flight Search. Automation for main flows, manual for content verification.

Item Pass/Fail Criteria:

Pass: Cars are displayed for a valid location/date.

Fail: Search returns no results incorrectly; date/time selections are not handled.

Suspension Criteria :

Test Deliverables:  
Test cases, Bug reports, Test summary, Test reports , Test Plan , Test Analysis Report

Environmental Needs:  
QA Environment, Chrome/Firefox/Edge.

Responsibilities:

Sai Bavesh: Create Testcases, Execute test cases, develop automation scripts, Defect Reports, Test Summary Report.

10.Approvals:

Vaishali

Project Manager

**7. Test Plan: Tour Search & Booking**

Test Plan Identifier: TP\_PHPTravels\_Tours\_2025\_v1

Introduction:  
To ensure users can discover, view details, and book tours and activities.

Test Items:  
Tours Tab, Search Destination field, Date Picker, Search button, Search Results, Tour Details page, Book Now button.

Features to be Tested:

Destination search.

Tour date selection and availability.

Accuracy of tour descriptions, inclusions, and pricing.

Payment processing.

Features not to be Tested:

How user/Booking details stored in the backend of website

Time taken for retrieval of content from backend.

Approach:  
Similar to Flight Search. Automation for main flows, manual for content verification.

Item Pass/Fail Criteria:

Pass: Tours are found for a destination; details are clear.

Fail: Tours not found; details are wrong or missing.

Suspension Criteria :

Test Deliverables:  
Test cases, Bug reports, Test summary, Test reports , Test Plan , Test Analysis Report

Environmental Needs:  
QA Environment, Chrome/Firefox/Edge.

Responsibilities:

Sai Bavesh: Create Testcases, Execute test cases, develop automation scripts, Defect Reports, Test Summary Report.

Approvals:

Vaishali

Project Manager

**8. Test Plan: Homepage & Featured Content**

Test Plan Identifier: TP\_PHPTravels\_Homepage\_2025\_v1

Introduction:  
To verify the content, functionality, and navigation of the main homepage elements, including featured items, search widgets, and overall responsiveness.

Test Items:  
Homepage URL, Main Navigation, Search Widgets (Flight, Hotel, Car, Tour), "Featured Flights" section, "Featured Hotels" section, "Popular Tours" section, Footer Links.

Features to be Tested:

All navigation links lead to the correct pages.

All search widgets from the homepage function correctly.

"Featured" and "Popular" sections display content and are clickable, leading to the correct details pages.

The overall responsiveness and layout on different screen sizes (desktop, tablet, mobile).

Features not to be Tested:

The deep functionality of the pages linked to (covered in Modules 4-7).

The business logic for how items are chosen to be "featured".

Approach:  
A combination of automated smoke tests for links and search widgets, and manual testing for content accuracy and responsiveness.

Item Pass/Fail Criteria:

Pass: The homepage loads; all modules and sections are present; links and widgets work correctly across different devices.

Fail: The page does not load; sections are missing; links are broken; the layout is broken on specific screen sizes.

Suspension Criteria :

Test Deliverables:  
Test cases, Bug reports, Test summary, Test reports , Test Plan , Test Analysis Report

Environmental Needs:  
QA Environment, Chrome/Firefox/Edge.

Responsibilities:

Sai Bavesh: Create Testcases, Execute test cases, develop automation scripts, Defect Reports, Test Summary Report.

Approvals:

Vaishali

Project Manager

**9. Profile Management (View & Update – Dependent on login)**

Test Plan Identifier: TP\_PHPTravels\_Profile\_Management\_2025\_v1

Introduction:  
This document details the testing strategy for the Profile Management module. The objective is to validate that both customers and agents can successfully view their current profile information and update their personal details, with changes being saved and reflected correctly across the application.

Test Items:  
My Profile link, Profile Page, Personal Details section (First Name, Last Name, Email, Phone, Address), Profile Photo upload, Save Changes button, Success Notification, Error Messages.

Features to be Tested:

View Profile: Correct pre-population of all user details on the profile page after login.

Update Personal Details: Successful update of fields like phone number and address.

Email Update: Functionality and validation for changing email address (may require re-verification).

Read-Only Fields: Validation that certain fields (e.g., User ID) cannot be edited.

Validation: Error messages for invalid inputs during update (e.g., invalid phone number format).

Persistence: Changes saved are persistent and visible upon subsequent logins and page refreshes.

Profile Photo: Ability to upload, change, and remove a profile image (validation of file type and size).

Features not to be Tested:

Changing the user type (e.g., from customer to agent).

Updating billing information or payment methods (covered in a separate module).

The security of the password change flow (this is part of "Forgot Password" or a separate "Change Password" module).

Approach:  
Automation with Selenium WebDriver. Tests will log in, navigate to the profile page, perform updates, save, and verify the changes were applied. Data-driven tests for various input scenarios.

Item Pass/Fail Criteria:

Pass: Existing details are displayed correctly; valid updates are saved and reflected; invalid updates are blocked with appropriate errors.

Fail: Details are not pre-populated; valid updates are not saved; invalid updates are accepted; application crashes on save.

Test Deliverables:  
Test cases, Bug reports, Test summary, Test reports , Test Plan , Test Analysis Report

Environmental Needs:  
QA Environment, Chrome/Firefox/Edge.

Responsibilities:

Sai Bavesh: Create Testcases, Execute test cases, develop automation scripts, Defect Reports, Test Summary Report.

Approvals:

Vaishali

Project Manager

**10.Test Plan - Post-Booking Actions (Dependent on Module 4: Flight Booking)**

Test Plan Identifier: TP\_PHPTravels\_PostBooking\_Actions\_2025\_v1

Introduction:  
This document details the testing strategy for key user actions that are available after a flight booking has been successfully completed. This module is dependent on a confirmed booking being created from Module 4 (Flight Search, Booking & Payment). The objective is to verify that users can download their itinerary and initiate a cancellation directly from their booking confirmation or a centralized management page.

Test Items:

Download Invoice/Itinerary Button: The UI element that triggers a PDF download.

Cancel Booking Button: The UI element that triggers the cancellation flow.

Cancellation Confirmation Pop-up/Alert: The modal dialog that appears after clicking "Cancel Booking".

Pop-up Buttons: The "OK" (Confirm) and "Cancel" (Dismiss) buttons within the pop-up.

Success/Error Messages: Notifications confirming the download or cancellation action.

Features to be Tested:

Download Itinerary as PDF: Clicking the "Download Invoice" or "Download Itinerary" button successfully generates and downloads a PDF file.

File Validation: The downloaded file is a valid PDF (correct format, not corrupted).

Cancel Booking Initiation: Clicking the "Cancel Booking" button triggers a confirmation pop-up alert.

Cancellation Pop-up Content: The pop-up displays the correct text (e.g., "Are you sure you wish to cancel this booking?").

Cancellation Confirmation: Clicking "OK" on the pop-up successfully initiates the cancellation process (front-end journey).

Cancellation Dismissal: Clicking "Cancel" on the pop-up closes it and does not initiate cancellation.

Status Update: After confirming cancellation, the booking status visually updates to "Cancelled" or "Pending Cancellation" on the UI .

Features not to be Tested:

The actual payment refund processing logic.

The backend API calls to airline systems to cancel the ticket.

The deep functionality of the "My Bookings" page (filtering, sorting, etc.).

Creating the booking itself (done in Module 4).

Approach:

Dependency: Test execution requires a pre-existing, confirmed flight booking. This will be created automatically by an automated script from Module 4 before these tests run.

Automation: This will be an automated test suite using Selenium WebDriver.

PDF Download Check: Automation will verify the download prompt appears and checks for the existence of the PDF file in the designated download directory. It will not validate the content of the PDF.

Popup Handling: The script will handle the browser-native alert pop-up for cancellation.

Item Pass/Fail Criteria:

Pass: PDF is downloaded successfully and is a valid file, Cancellation pop-up appears with the correct text, Clicking ok cancels the booking, Clicking "Cancel" closes the pop-up with no change to the booking.

Fail: PDF download does not work or file is corrupted, Cancellation pop-up does not appear, UI does not respond correctly to "OK" or "Cancel" actions.

Suspension Criteria :

Environmental Needs:  
QA Environment, Chrome/Firefox/Edge.

Responsibilities:

Sai Bavesh: Create Testcases, Execute test cases, develop automation scripts, Defect Reports, Test Summary Report.

Approvals:

Vaishali

Project Manager