Software Test Plan

Pango Pay&Go - Parking Lot Manager

1. Overview

This document presents the testing approach and validation scope for the Pango Pay&Go Parking Lot Manager.

It defines the quality goals and test coverage required to ensure the system functions correctly across critical flows such as:
user authentication, vehicle parking, input validation, cross-user consistency, and parking history tracking.

---

2. Scope

In-Scope (Functional Coverage)

This test plan covers the following features and flows of the Pango Pay&Go Parking Lot Manager

- **Vehicle Parking/Unparking**: Starting and ending a parking session using valid plate and slot inputs.
- **License Plate Field Validation**: Verifying frontend and browser-level validations for input constraints (format, length, character types).
- **Duplicate Parking Prevention**: Ensuring a vehicle cannot be parked by more than one user simultaneously.
- **History Tracking**: Checking that completed parking sessions are recorded and visible in the history table.

Out-of-Scope (Excluded Areas)

The following areas are excluded from the current testing scope:

- **API Testing**: No direct backend or API request validations.
- **Performance/Load Testing**: Stress or concurrency testing is not part of this round unless otherwise defined.
- **Mobile UI Testing**: Responsiveness and layout for mobile or tablet devices are not covered.
- **Cross-Browser Testing**: Focus is on a single browser (Chrome).

3. Test Strategy

- **Framework**: Python + PyTest + Selenium
- **Design Pattern**: Page Object Model (POM)
- **Execution**: Chrome Browser
- **Validation**: Assertions, HTML5 validation checks, text-based verifications

4. Test Environment

- Web App URL: `http://localhost:5000`
- Browser: Chrome Browser
- Python: 3.11+

5. Test Cases

TC01 - Successful Login

- **Objective**: Ensure that users can log in with valid credentials.
- **Precondition**: User account exists in the system.
- **Steps**:
  1. Open the login page.
  2. Enter a valid username and password.
  3. Click the "Login" button.
- **Expected Result**: User is redirected to the dashboard.

TC02 - Start Parking Session

- **Objective**: Validate that a user can initiate a parking session with valid input.
- **Precondition**: User is logged in.
- **Steps**:
  1. Navigate to Dashboard
  2. Enter a valid license plate (8 digits).
  3. Enter a slot
  4. Click "Start Parking".
- **Expected Result**: A success message confirms the session has started.

TC03 - Prevent Duplicate Vehicle Parking

- **Objective**: Ensure a vehicle already parked by one user cannot be parked again by another user.
- **Precondition**: Vehicle is already parked by a different user.
- **Steps**:
    1. Login as User A and start parking with plate 12121212.
    2. Logout and login as User B.
    3. Try to park the same vehicle again.
- **Expected Result**: Error alert is shown: “Vehicle already parked”.

TC04 - Validate History Entry

- **Objective**: Confirm that a completed parking session appears in the history table.
- **Precondition**: Parking session has been started and ended by the user.
- **Steps**:
    1. Start parking with a valid vehicle.
    2. End the session.
    3. Navigate to the History page.
- **Expected Result**: The vehicle appears in the history list with correct details.

TC05 - License Plate Field Validations

- **Objective**: Verify front-end and HTML5 validations for invalid license plate inputs.
- **Precondition**: User is logged in and on the Parking page.
- **Test Variants**:
    - **Empty input** → Expect HTML5 browser validation.
    - **Less than 8 digits** → App-level error: “License plate must be exactly 8 digits”.
    - **More than 8 digits** → Input should be ignored.
    - **Alphabetic characters** → Field should remain empty or ignored.
    - **Sequential digits** (e.g., 12345678) → App-level error.
    - **Identical digits** (e.g., 11111111) → App-level error.

- **Expected Result**: Each scenario triggers appropriate validation (either native HTML5 or app-level message).

6. Traceability

Requirement	**Test Case ID**
Login functionality	TC01
Vehicle parking	TC02
Prevent duplicate parking	TC03
Track parking history	TC04
Plate input validations	TC05

7. Risks & Assumptions

- Alert messages may appear on the global dashboard instead of on current page.
- Timestamps may not align with local timezone (Israel Time)
- Validation relies on both HTML5 and custom JS logic, may behave inconsistently across browsers