

# Test Plan Document

## 1. Test Plan ID

TP-Automotics-001

## 2. Introduction

This document outlines the testing strategy, objectives, resources, and timeline for automating the test cases for the Automotics Web Application. The focus is on ensuring the functionality of superuser and fleet manager modules, including dashboards, reports, clients, fault codes, notifications, users, and groups.

## 3. Objectives

Ensure critical workflows (e.g., login, report upload, client management) are functional and reliable.

Validate user role-specific permissions for superusers and fleet managers.

Automate repetitive tests to improve efficiency and reduce human error.

## 4. Scope of Testing

### **In Scope:**

Dashboard: Navigation and button functionality.

Repair: Upload and manage reports, modal operations, and filters.

Client: Create, edit, delete clients, and validate email generation.

Fault Code: Manage and validate fault code solutions.

Notifications: Validate received notifications and status changes.

User Management: CRUD operations on users, permission validation.

Group Management: CRUD operations on groups.

Authentication: Login, logout, and password recovery.

### **Out of Scope:**

- Performance testing.
- Non-functional aspects like UI/UX usability testing.

## 5. Testing Approach

Framework: Cypress with Page Object Model (POM).

Test Design: Scenarios designed using user stories and acceptance criteria.

Locators: XPath and CSS selectors defined in separate JSON files.

Assertions: Use Cypress commands to validate DOM changes, API responses, and navigation.

Reports: Generate reports using Cypress dashboard or Mocha Awesome.

## 6. Test Environment

Browser Compatibility: Chrome, Firefox.

Devices: Desktop (Windows, macOS).

Environment: Staging and Production.

## 7. Test Tools

Automation Tool: Cypress.

Version Control: GitHub.

CI/CD: GitHub Actions for automated test runs.

Test Management: Jira for tracking test cases and results.

## 8. Test Cases to Automate

Superuser tests

1. Open dashboard and make sure all endpoints are working
2. Test buttons on dashboard
  - Upload report goes to /repair with upload report modal open
  - Create client goes to /client
  - Assign client goes to /repair
  - Faultcode solutions goes to /faultcode
  - Make sure see all clients goes to /client
  - Make sure see all faultcode solutions go to /faultcode

- Make sure see all reports goes to /repair

1. Repair:

- Upload 5 reports and wait for result and validate the result

1. Make sure report modal is openable (eye)

2. Check if all car data is filled in

3. Check the VIN

4. Repair modal: Change client needs to work

5. Repair modal: Change status needs to work

6. Repair modal: Test create new car

7. Repair modal: Test edit car

8. Repair modal: Test delete car

9. Repair modal: Click on comments and add new comments and reload and update comment and validate results

10. Repair modal: Reanalyze report and validate

11. Repair modal: download original report

12. Repair modal: download custom report

13. Repair modal: Open requested services

14. Requested services page: Make sure all endpoints succeed

15. Requested services page: Make sure we can create a service requested based on selection of existing service and one of a custom request and validate data is correct

16. Requested services page: Change status of opened service request

17. Requested services modal: Open service request modal and test status change

18. Requested services modal: Test comments for a service request

19. Requested services modal: Validate that the History of status is updating after changing status

20. Requested services modal: Test file upload and validate upload

21. Requested services modal: Start a subservice #1 and validate information and validate service modal is openable immediately

22. Requested services modal: Test subservice #1 comments and validate
23. Requested services modal: Test subservice #1 status change and validate
24. Requested services modal: Test subservice #1 file upload and validate
25. Requested services modal: Test subservice #1 history of status change after changing status
26. Requested services modal: Test the back button and create new subservice #2 with needs to be accepted, refuse and make sure it is not openable again
27. Requested services modal: Create new subservice #3 with needs to be accepted, accepts and make sure openable and validate information
28. Repair: You have created a couple cars, test the filter toggle to make sure it works, test and make sure all filters work and validate results
29. Repair: Test the show more button at the bottom of the page to make sure it works and validate data
30. Client: Create 5 new clients and validate information and make sure client email is created
31. Client: Make sure endpoint is getting clients
32. Client: Test create client and validate
33. Client: Test edit client and validate
34. Client: Test delete client and validate
35. Client: Test show more and validate
36. Faultcode: Make sure endpoint is getting faultcodes
37. Faultcode: Since you have created repairs, there will be faultcodes here, edit them and validate
38. Faultcode: Validate the people used this solution in the fault code solution
39. Faultcode: Test the show more
40. Faultcode: Test the delete the faultcode
41. Notifications: Validate you received notifications from the previous actions of uploading reports
42. Notifications: Read one by clicking on one and validate
43. Notifications: Click on read all and validate everything is read
44. Profile: Validate changing profile works
45. Users: Validate the endpoint is working

46. Users: Invite 3 new users one with Fleet managers and one with fleet owner and one temporary one

47. Users: Edit user by changing their name and validate

48. Users: Delete the temporary user

49. Users: Make sure the resend email works

50. Group: Make sure api works

51. Group: Create new temporary group

52. Group: Edit new temporary group

53. Group: Delete temporary group

54. Logout user

Fleet manager account test

1. Test the invite user with the previous email link

2. Forget password for the user

3. Login user

4. Make sure Repair page is accessible or not based on user permissions

5. Make sure Client page is accessible or not based on clients permissions

6. Make sure Faultcode page is accessible or not based on fault code permissions

Now continue with the previous flow while checking user permissions

7. Open dashboard and make sure all endpoints are working

8. Test buttons on dashboard

- Upload report goes to /repair with upload report modal open

- Create client goes to /client

- Assign client goes to /repair

- Faultcode solutions goes to /faultcode

- Make sure see all clients goes to /client

- Make sure see all faultcode solutions go to /faultcode

- Make sure see all reports goes to /repair

1. Repair:

- Upload 5 reports and wait for result and validate the result

1. Make sure report modal is openable (eye)

2. Check if all car data is filled in

3. Check the VIN

4. Repair modal: Change client needs to work

5. Repair modal: Change status needs to work

6. Repair modal: Test create new car

7. Repair modal: Test edit car

8. Repair modal: Test delete car

9. Repair modal: Click on comments and add new comments and reload and update comment and validate results

10. Repair modal: Reanalyze report and validate

11. Repair modal: download original report

12. Repair modal: download custom report

13. Repair modal: Open requested services

14. Requested services page: Make sure all endpoints succeed

15. Requested services page: Make sure we can create a service requested based on selection of existing service and one of a custom request and validate data is correct

16. Requested services page: Change status of opened service request

17. Requested services modal: Open service request modal and test status change

18. Requested services modal: Test comments for a service request

19. Requested services modal: Validate that the History of status is updating after changing status

20. Requested services modal: Test file upload and validate upload

21. Requested services modal: Start a subservice #1 and validate information and validate service modal is openable immediately

22. Requested services modal: Test subservice #1 comments and validate

23. Requested services modal: Test subservice #1 status change and validate
24. Requested services modal: Test subservice #1 file upload and validate
25. Requested services modal: Test subservice #1 history of status change after changing status
26. Requested services modal: Test the back button and create new subservice #2 with needs to be accepted, refuse and make sure it is not openable again
27. Requested services modal: Create new subservice #3 with needs to be accepted, accepts and make sure openable and validate information
28. Repair: You have created a couple cars, test the filter toggle to make sure it works, test and make sure all filters work and validate results
29. Repair: Test the show more button at the bottom of the page to make sure it works and validate data
30. Client: Create 5 new clients and validate information and make sure client email is created
31. Client: Make sure endpoint is getting clients
32. Client: Test create client and validate
33. Client: Test edit client and validate
34. Client: Test delete client and validate
35. Client: Test show more and validate
36. Faultcode: Make sure endpoint is getting faultcodes
37. Faultcode: Since you have created repairs, there will be faultcodes here, edit them and validate
38. Faultcode: Validate the people used this solution in the fault code solution
39. Faultcode: Test the show more
40. Faultcode: Test the delete the faultcode
41. Notifications: Validate you received notifications from the previous actions of uploading reports
42. Notifications: Read one by clicking on one and validate
43. Notifications: Click on read all and validate everything is read
44. Profile: Validate changing profile works
45. Users: Validate the endpoint is working
46. Users: Invite 3 new users one with Fleet managers and one with fleet owner and one temporary one

- 47. Users: Edit user by changing their name and validate
- 48. Users: Delete the temporary user
- 49. Users: Make sure the resend email works
- 50. Group: Make sure api works
- 51. Group: Create new temporary group
- 52. Group: Edit new temporary group
- 53. Group: Delete temporary group
- 54. Logout user

## 9. Test Deliverables

Automated test scripts.

Execution reports with logs and screenshots.

Bug reports logged in Jira.

## 10. Roles and Responsibilities

### **Test Lead:**

Prepare test plan, manage testing activities.

### **Automation Engineer:**

Develop and maintain test scripts.

### **DevOps Engineer:**

Integrate test runs with CI/CD pipeline.

## 11. Test Case Design

Example Test Case for Superuser Dashboard:

Test ID	Test Name	Steps	Expected Result
---------	-----------	-------	-----------------

TC-001	Dashboard Navigation	1. Login as superuser.	
--------	----------------------	------------------------	--

		2. Click "Upload Report".	
--	--	---------------------------	--

		3. Verify navigation to /repair.	Navigation is successful; modal opens as expected.
--	--	----------------------------------	--



## 12. Entry and Exit Criteria

### **Entry Criteria:**

Application is stable and deployed in the test environment.

All locators and test data are finalized.

### **Exit Criteria:**

All critical tests have passed.

Bugs are logged, and fixes are validated.

## 13. Reporting

Reports will be shared daily and include:

Test execution results.

Bug summary.

Automation progress updates.