Test Strategy for Motability Vehicle Application Flow
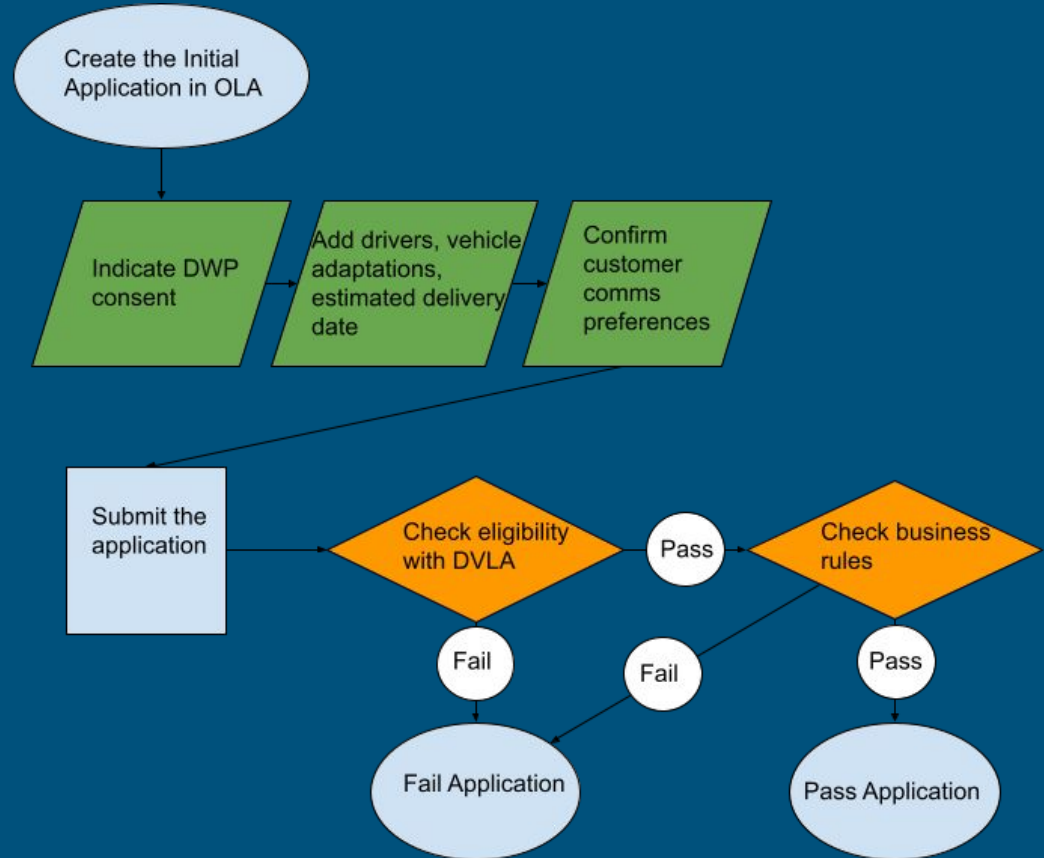
Graeme Stewart

Overview

The application process for a Motability vehicle progresses over a number of weeks, controlled through one of our main applications (called OLA).

The Dealer is acting on behalf of MO so the customers deal with Motability via that route. The dealer takes the details of the customer and enters them in OLA.

The application is saved when submitted. Once the application is submitted, we call a third party service provided by the DVLA to verify the driver(s) details and to check their eligibility. We then use a category code endorsement to set business rules.

Key Steps

- Dealer creates initial application in OLA

- Application data is input

- Application is saved and submitted

- Submission triggers eligibility checks via DVLA API

Testing Strategy Overview

Testing Types:
- Unit testing: API endpoints
- Integration Testing: OLA with external services (DVLA)
- End-to-End Testing: Full customer journey validation
- Business Rule Validation: Focused testing on eligibility logic

Eligibility Checks

Eligibility Checks include:
- DVLA Driver licence verification
- Business rule enforcement

Goal: Ensure only eligible applications proceed

Eligibility Checks are tested in all 3 types of testing: Unit, Integration and E2E

API Testing Strategy

Endpoints to Test:
- Definite endpoint for creating the initial application (/API/v2/applications) via a POST call, but could be used for other calls (for example GET to retrieve an application)
- Another endpoint could be used for DVLA checks

Test Cases:
- Valid and invalid payloads
- Missing required fields

Business Rules - What to Test

Rule	Description	Test Scenarios
DRIVER_WITH_AUTOMATIC_LICENCE_CANNOT_DRIVE_MANUAL_VEHICLE	The rule will fail if the vehicle has manual transmission and the driver has an automatic licence.	Driver with automatic licence Driver with manual licence Driver with no licence Driver with both licences Manual transmission vehicle Automatic transmission vehicle
AT_LEAST_ONE_DRIVER_WITH_FULL_LICENCE	The rule will pass if there is at least one driver with a full entitlement (i.e. not provisional), if not it will fail.	1 Driver - Full entitlement 1 Driver - not full entitlement Multiple Drivers - 1 with full entitlement Multiple Drivers - none with full entitlement Multiple Drivers - all with full entitlement
DRIVER_IS_DISQUALIFIED	The rule will fail if the driver is disqualified at the moment.	Driver is disqualified Driver is not disqualified

Business Rules - What to Test

Rule	Description	Test Scenarios
DAG_V2_DRIVER_LICENCE_CATEGORY_G HI_ENDORSEMENT	The rule will fail if the driver licence has 4 or more endorsements in the last 4 years from categories G, H, and I combined.	Driver with 0 endorsements Driver with 4 endorsements in 1 single category Driver with 4 endorsements spread across categories Driver has less than 4 endorsements spread across categories
DAG_V2_DRIVER_LICENCE_CATEGORY_G _ENDORSEMENT	The rule will fail if the driver has two or more endorsements in the last 4 years from category G and has a conviction date in the past 4 years.	Driver has 2 or more G endorsements and has a conviction date in the past 4 years Driver has 2 or more G endorsements but not a conviction date in the past 4 years Driver has no G endorsements
DAG_V2_DRIVER_LICENCE_CATEGORY_F _ENDORSEMENT	The rule will fail if the driver has any endorsements in the last 4 years from category F and has a conviction date in the past 4 years.	Driver has F endorsements and has a conviction date in the past 4 years Driver has F endorsements but not a conviction date in the past 4 years Driver has no F endorsements

Business Rules - Approach

- Create mock driver profiles with varying licence statuses
- Simulate DVLA responses using a mock server
- Validate rule engine outcomes
- Use tool such as Postman for API testing
- Automated test suites

Automation Strategy

Why Automate?

- Frequent rule updates
- Regression testing
- Faster Feedback compared to manual testing

What to Automate:

- API contract tests
- Rule validation scenarios
- End to end customer journey

Metrics for Success

Key Metrics:

- % of test coverage on eligibility rules
- Time to detect rule failures
- Number of false positives/negatives in rule enforcement
- API response times and error rates

Risks & Mitigations

Risk	Mitigation
External API downtime	Use mocks/stubs in test environments
Rule misconfiguration	Regular audits and test case reviews
Incomplete test data	Maintain a comprehensive test data set

Summary & Next Steps

Summary:

- Eligibility checks are important to application integrity
- Testing must cover API, integration, and business logic
- Automation and metrics are key to long-term success

Next Steps:

- Finalise test cases
- Set up test environments
- Begin automation of rule validations