Test Plan

Crossyn Project

Authors	:	**Dobri Trifonov, Joran van de Moosdijk, Robert Enuta, Stanislav Petkov, Tobias Halomoan**

Contents

1.Context

1. Introduction

We have been asked by Crossyn to create a system that checks data they collect and make it visible to the clients of the company. The data is collected by Crossyn and sent to us to be tested and refined. We create trips out of the data received and assure the trips are complete and enriched.

2. Objectives

Creating a well-rounded and rich application, while using agile, means continuous changes and updates are made. Changes that can lead to oversights in code or even bugs, for this reason extensively testing the product is important.

Testing will be done in multiple ways to ensure the quality of the code and the product as a whole.

3. Scope

While the project is in development, both back-end and front-end need to be tested before deployment, but also the connectivity between the front-end and back-end and between back-end and database.

2.Testing strategy

1. Overview

For testing of the methods and objects found inside the java application, unit tests will be implemented alongside new code, but this will be also tested in GitHub, by the CI/CD pipelines that have been set up, and a visualisation of the quality of the code will be available in SonarQube.

//The React functionality will be tested using Cypress.

The test will mostly focus on testing the algorithms in the backend and the logic used to perform tasks. Unit test will cover all the code that focuses on changing the data received.
//The testing done in the front-end will focus mostly on making sure the html elements work and display properly.

Testing will be done on a high level, focused on each individual element rather than only on the totality of the system. Having a large number of detailed tests will make discovering and fixing issues more efficient.

2. Testing

1. Unit Tests

The unit tests will be written in parallel with the java code. Every class will have its own unit tests, because of these finding errors and fixing them will be much faster.

The testing will happen in IntelliJ Idea, every time the updates are pushed to the git repository.

2. Integration Tests

Integration tests are ran automatically bit GitHub every time a new version is uploaded to the repository. These tests will build the project, run the unit tests and, using SonarQube, will reveal what flaws the code contains.
We will also make use of Docker to further speed up the testing process.

3. User Acceptance Tests

User acceptance tests confirm that the project is ready to be rolled out and used. During acceptance tests the users compare the final product with the initial requirements.

Since this set of tests requires users to use the system, a random group of people will be selected for this test. A test script will be created to guide users through the tasks they need to perform. The tests will be recorded, and the data will be used for further improvements.

For conducting these tests, we will use a computer running Windows OS. No additional tools or resources are required at this time, but if any necessities arise, we will discuss about what tools are needed and the results will be talked about with the client.

3. Schedule

Unit tests and CI/CD tests take place whenever a new version of the app is pushed to the GitHub repository, these tests don't take much time to conduct and are vital to keeping the code in check, for these reasons they will be conducted as often as possible.

User acceptance tests will take place after a version stable enough for users becomes available.

3. Hardware Requirements

Different tests will require different equipment for through completion.

Continuous integration tests are temporary ran through the development computers, and they will be set up to run through a docker.

User acceptance tests will be done on a computer running Windows 10/11, since that is the target operating system and the one used for development. If the need arises, other systems will be tested such as machines running Linux or MacOS

4. User Acceptance

ID	**Scenario**	**Acceptance criteria**	**Requirements**	**Testing plan**
1	As a user I can login using my registered information so I can access the application.	User can log into the website.	-User is connected to the website. -User has an account.	-Open main page and navigate to the login button. -Login page opens and a valid account is inputted.
2	As a personal user I can register as a driver of a vehicle so I can acquire personalized data for myself.	User can create an account and can select or register a vehicle.	-User has an email that can be used for registration.	-Open the registration page. -Input valid information. -Select a vehicle or register one.
3	As a vehicle owner I want to see mainly my vehicle trip data/analytics so I can be aware of my driving routine.	User can view their trips.	-User has an account and vehicle. -User has driving data.	-Open the website and go to the trip page.
4	As a fleet owner I want to see an overview of the trips for my collection of cars so I can easily know the state of my fleet.	Fleet owner can see a collection of vehicles.	-User has a valid fleet owner account and vehicles registered.	-Open the vehicle page.
5	As a fleet owner, I want to be able to register my vehicles on the website.	Fleet owner can register vehicles.	-Fleet owner has a valid account.	-Fleet owner opens the vehicle page. -Fleet owner clicks the create vehicle button and inputs the vehicle data.
6	As a fleet owner I want to see an overview of the trips for my collection of cars so I can easily know the state of my fleet.	Fleet owner can see their cars and the trips assigned to them.	-Fleet owner registered at least one vehicle.	-Tester logs in. -Tester clicks on a product. -Tester clicks on a user account.

			-Vehicles registered have at driving data	
7	As a user I want to view a graphical representation for each of my trip so I can easily understand my trips.	User can see a representation of their trip on a map.	-User has a vehicle registered and at least a trip travelled.	-Visit the trip page. -Select the trip to be visualised.
8	As a vehicle owner I want to see mainly my vehicle trip data/analytics so I can be aware of my driving routine.	User can see details about their trips.	User has at least a trip registered.	-Visit the trips page.
9	As a user I want to view additional information of my trips (weather, distance, time elapsed) so I can react accordingly with the provided information.	User can see the weather and other details from their trips.	-User has a trip registered.	-Visit the trip page.
10	As a user I can sort my trip using multiple filters so I can easily access my trips.	User can see a list of trips sorted by different categories.	-User has multiple trips registered.	-Visit the trip page. -Select the data on which the list will be sorted.