# Buggy Cars Rating

## Test Approach

I would take the Exploratory testing approach to verify the functionality and security. since there’s no requirements provided, all the criteria are based on my common understanding.   
  
At the beginning, I would explore the site thoroughly. In order to plan the exploratory testing better, I would set an explore target, resource and the destination of discovery. And then I would write this down. For example, when I want to explore the login functionality. I would write it like this.

“**explore** *authentication* **with** *various username and password* **to discovery** *the function works well*”

Or like security testing example

“**explore** *editing profiles* **with** *injection attacks* **to discovery** *the security vulnerabilities*”

I would use it like a map to explore the functionality, prepare test data and execute them and record every result I got.

Once I go through all the exploratory testing, I figure out the most significant cases and write them in BDD style and create a tiny automation framework to execute them automatically.

For Example:

Feature 1: Registration:

Scenario: Register a new user (successfully)

Scenario: Register a new user but the name is used (give error msg)

Scenario: Register a new user but the confirm password doesn’t match (give error msg)

Scenario: Validate losing the require field cause the error msg

Feature 2: Login:

Scenario: Login with valid username and password

Scenario: Login with Invalid credentials (give error msg)

Feature 3: Vote:  
 Scenario: the login user can vote with a comment

Scenario: the login user can vote without a comment

Scenario: the login user cannot vote a model twice

Scenario: no-login user cannot vote

Feature 4: Update Profile

Scenario: user can successfully update the profile

Scenario: user can change the password and login with the new one

Scenario: without correct current password, the user cannot update password

Scenario: without required field, the user cannot update the profile

Feature 5: Site Navigations and view

Scenario: User can navigate to different pages(make, model, overall-rating) and go back to home

Scenario: User can sort the overall rating list

Scenario: User can use page edit to go through the overall rating list

I design an end-to-end test framework, which is based on Behavior-Driven Development (BDD), Python(pyTest, pyTest-bdd) and Selenium. Because BDD test case is written by plain English, it needs an agent to translate it into automation code. This is the way what a glue code library works. In this framework, I use Pytest-BDD as such an agent. It will find test cases from BDD features file, and matches the Given, When and Then steps with the annotations of test functions.

In order to have better maintenance, the framework follow the Page Object Model (PMO), which is a design pattern for separating the test step with the implement of UI of the page. In POM, each page should be organise as a class, in which each components and actions are defined. The advantage of doing so is that we do not need to change when the UI of the page is changed. We only need to change code in the page class. Moreover, the page class can be reuse in many different test class, which can reduce the duplicate code.

## Critical Bugs

Bug #1  
Title: user navigate to the make page, but cannot navigate back to home

Description:  
When use go to a make page, the user cannot navigate back to home page by click the home log.

Steps to reproduce:

* Click the popular make link in the home page
* Go to the make page and click the home logo, try to go back to home page

Expect Result:

* Correctly navigate to the home page

Actual Result:

* Cannot navigate to the home page
* In the console, there are some error msg

Graphical user interface, text, application

Description automatically generated

Bug #2  
Title: in some model page, there is no author info in the comment list

Description:

go to the model <https://buggy.justtestit.org/model/c0bm09bgagshpkqbsuag%7Cc0bm09bgagshpkqbsuh0>

Check the comment info, there is no author info  
  
Steps to reproduce

- Click ‘Popular Model’ image on home page

- Check the comment list



Expect Result:

* The date, author and comment can be displayed correctly

Actual Result:

* Lose the author info

Table

Description automatically generated

Bug #3

Title: password does not be encrypted when requesting login

Description:   
When user login with the credential, using the developer tools to check the request. The password does not be encrypted



Steps to reproduce

- login with valid credential

- open the developer tool (Chrome) to check the request



Expect Result:

* The password is encrypted

Actual Result:

* No any encrypted

Graphical user interface, application

Description automatically generated

## Other Identified/Potential Bugs

1. Lancia “Stratos” model page does not work
2. Lancia “Ypsilon” model image does not displayed
3. In the Page Navigation, Page 1 and 5 values are not working
4. Sorting of Rank is displayed according to mathematically but alphabetically
5. Error messages are not user friendly and have some security vulnerability:

Graphical user interface, text, application

Description automatically generated