*You must send me (by email) a 3-4 pages report in which you may highlight the main goals, achievements, limitations, advantages and conclusion of the developed scenarios.*

# main goals

The main goal of this work is to explore Cypress framework{link} and create some web test scenarios with this framework which are focused on user experience and user interface.

For experimental use of Cypress I decided to use demo of open-source eCommerce platform called <https://www.opencart.com/>. Official plus demo link. In this demo website I decided to create simple test scenarios for login page, use of e-shop and use of navigation, where I found some errors in UI that are explained in next sections.

From these scenarios and also from Cypress framework I describe some of the advantages and limitations. Also, from research I will try to explain best-practices that I did not applied practically in my project.

# Achievements

In this section I will try to explain scenarios and their achievements, that I separated in 4 parts containing scenarios of Login page, Ordering item in E-shop, Navigation, Not Found Page.

### Test scenario of Login page

Testing of login page ended up very well as all asserts were successful, but from the view of user experience I think this login page can be slightly improved that I explain later.

When I started testing with Login page for the first, I had to check if I’m on correct subpage with correct heading information. Also, I considered that user maybe not have account, therefore login page should have link to register page.

Next, I tried to consider all possible error messages that alert user about incorrectly entered inputs and there I found out that website should use basic validation of empty inputs in frontend. Instead, developers chose to do all validations in server side and just with one exactly same message in different situation, that in my opinion this is lazy option, and I am considering it as not best practices.

Last after login I checked if login page redirect user to account page, which turned out successfully.

### Test scenario of Ordering item in E-Shop

### Test scenario of Navigation

### Test scenario of Not Found Page

# limitations

# advantages

Each test can be isolated in structure so you can organize test in data model object structure in folders.

Graphical user interface, text, application

Description automatically generated

We can make own custom cypress method using support folder -> index.js and commands.jsText

Description automatically generated

Login by sme mali robiť cez api poslat len request a zachytit token cez response

<https://docs.cypress.io/guides/references/configuration#cypress-json>

nieco o configuration a base url aj horny link aj dolny

https://docs.cypress.io/guides/references/best-practices#Web-Servers

Graphical user interface

Description automatically generated with medium confidence

# conclusion

**my notes:**

Text

Description automatically generated

<https://rahulshettyacademy.com/blog/index.php/2021/10/24/difference-between-cypress-and-selenium/#t-1643629689033>