**Test report**

*Report by Jordan Radushev*

In the following report I will be explain how my application is tested using different methods to prove it works the way it should.

1. **Automated tests – CI**

As my project consists of two frontend applications and 3 microservices as backend, I set the following tests as an automated test as a part of the CI of my project:

* Repository tests – backend, to prove that the requests to the database are executed correctly.
* Controller tests- backend, to prove that every controller act as it should, based on standard MVC tests – on input corresponds output or an error with an appropriate message.

1. **Backend security tests**

I use Zuul proxy as an entry point to my backend. It filters the requests and forwards the request to the selected microservice if the conditions to do that are met, for example you need a valid JWT in the header to pass or you need to have a ROLE\_ADMIN as a role in the database to proceed. These test are made with postman and they are not automated, since it needs real work environment to debug it.

1. **UI Tests**

* As an UI test, I make a set of instructions “how to use it”. The main reason for choosing this strategy of testing is lack of time to implement more effective strategy. About this strategy – it is easy to be implemented, it acts as guide to my applications’ UIs.
* On both applications I made a validation on most of the inputs, so the user cannot type inappropriate strings.
* In terms of UX, tried to implement some basic principles of UX, so the user cannot do actions such as sending an empty order, or sending an order without and address attached to it.
* End to End testing belongs to the non-functionality testing. I order all the features implemented in appropriate sequence an test it real time if every single feature works