For this project I decided to make a website where users can buy and sell products(Games).

The documentation I decided to make are all in the document folder:

* Project plan
* Design document
* Research document
* Security report
* Test plan
* Test cases
* Test report
* UX feedback report
* Code Reviews
* Group FeedPulse
* Teachers FeedPulse
* Picture of SonarQube
* docker-compose
* .gitlab-ci

The rest API, made in java spring boot, is in the Individual\_project folder.

For the API, to ensure code quality I made 121 unit tests covering 90.5% of the code and a CI/CD pipeline using SonarQube and docker.

It uses http and WebSocket to communicate with the front end, the authorization and authentication are made using JWT(json web tokens).

There are also custom exceptions to better display what went wrong. The DTO(data transfer object) pattern is applied and implemented using for example DTO Convertors(classes that convert DTOs to entities and vice versa), all the solid principles are also used.

The database used is MySQL, to connect to it ORMs(object related mapping) are used, the object related mapping also implements polymorphism, one to one, one to many and many to many joins. The database itself has indexes for the fields that are most often used and searched by. Versioning uses flyway.

The single page application, made using react, is in the myfirstreact folder.

The react front end is a lot simpler then the API it has nothing fancy, just simple JavaScript with some react elements like reactDOM, useState, UseEffect, useParams, useNaigate, Route, Link… it has some regex for validating and axios is used to connect to the API.

No Bootstrap is used, all the html and css is written manually which is one of the reasons for the simplistic look of the application, I didn’t put much work into the css.

There are tests made for the frontend, a dozen simple unit tests that use jest and one end to end test using cypress(end to end testing is amazing).