Sprint Retrospective, Iteration #3

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| User Story # | Task # | Task Assigned To | Estimated Effort per Task  *(in hours)* | Actual Effort per Task *(in hours)* | Done  *(yes / no)* | Notes |
| *User creates an account. Then he creates a house so his housemates can join.* | Create User Controller and Repository | Fabian | 1.5 | 2 | Yes |  |
| Create House Controller and Repository | Fabian | 1.5 | 2 | Yes |  |
| Create Request Controller and Repository | Ina | 0.5 | 1 | Yes |  |
| Create User  Entity | Ina && Fabian | 0.5 | 1 | Yes |  |
| Create House  Entity | Ina && Fabian | 0.5 | 1 | Yes |  |
| Create Request Entity | Ina && Fabian | 0.5 | 1 | Yes |  |
| Create tests for User Entity | Ina | 0.5 | 0.5 | Yes |  |
| Create tests for House Entity | Ina | 0.5 | 0.5 | Yes |  |
| Create tests for Request Entity | Ina | 0.5 | 0.5 | Yes |  |
| Create tests for Request Controller | Ina | 0.5 | 1 | Yes |  |
| Create tests for House Controller | Fabian | 0.5 | 0.5 | Yes |  |
| Create tests for User Controller | Fabian | 0.5 | 0.5 | Yes |  |
| User registration and authentication | Atanas | 2 | 3 | Yes | I started working on the authentication microservice. I had used Spring security before, so this was not that challenging, but connecting everything and microservice communication turned out way harder than expected. |
|  | Research JWT tokens and their importance in the microservice architecture | Atanas | 1 | 2 | Yes | I had little experience with JWT tokens. I had only tried to break authentication before, which gave me a unique perspective on the security issues related to JWT.  I still had to properly research how to use the tokens from a developer’s perspective and how to implement the communication between the microservices. |
| *User adds products to the fridge. Then, after there are products in the fridge, users are able to make use of products, by adding transactions in order to keep track of how the products are used* | Create Product Entity | Kendra | 0.5 | 0.5 | Yes |  |
| Create Transaction Repository | Stoyan | 1 | 1 | Yes | From the past week I already have some standard how to create the Transaction repository and I developed it this week. |
| Create Transaction Entity | Stoyan | 1 | 1 | Yes | The transaction repository was created. and tested manually. |
| Create Product Controller and Repository | Kendra | 1.5 | 2 | Yes | The Product controller was successfully created only with some basic methods. We will later discuss and add other more complicated methods to support the requirements needed. |
| Create Transaction Controller | Stoyan | 1.5 | 1.5 | Yes | I created the transaction controller with some basic methods that I believe will be useful. |
| Create tests for Product entity | Kendra and Oskar | 1 | 1.5 | Yes | Some basic tests were implemented in for the Product entity class. I believe that this class is complete and won’t need many more testing. |
| Create tests for Transaction entity | Oskar | 1 | 0.5 | Yes |  |
| Create tests for Product controller | Kendra and Oskar | 1 | 1.5 | Yes | We have tested nearly all the methods implemented in the controller. However, there is still room for improvement, especially given the fact that we still need to add methods to the controller and therefore we will need to keep the tests up to date with the changes. |
| Create tests for Transaction Controller | Stoyan | 0.5 | 0.5 | Yes | I have created basic tests for the transaction controller. |
|  | Implement a gateway service that serves as an entry point to the application’s REST API | Atanas | 1.5 | 3.5+ | No | I had to research a lot about the way a gateway is supposed to work in a microservice project architecture. I found some great java libraries that were linked by Spring, so I decided it would be great to use them in our project.  I also found a load balancer service that was connected to the gateway, so I started researching it as well, because it would be helpful if we get a lot of users, because it would be easier to scale in the future. |
| Initialize database create sql schemas | Oskar | 2 | 2 | Yes |  |
|  |  |  |  |  |  |

Project: Software Engineering Methods - Student House Food Management

Group: 51

Main Problems Encountered

**Problem 1**

Description: We had problems in setting up the multiple microservice in one project.

Reaction: Further document on the process of working with different microservices in one project.

**Problem 2**

Description: We had problems regarding gradle and how to compile all microservices, because they are in different folders, but should be connected by a single build.gradle file. We also had problems with the gitlab pipeline and how to make it run scripts for all projects.

Reaction: We researched gradle and gitlab, and found how to link the microservices from the settings.gradle.

**Problem 3**

Description: We realized that we will have to refactor our microservices so that they have the same structure.

Reaction: We talked with each other and we decided that we will follow a particular structure so that it will be easier to merge and connect the microservices later.

Adjustments for the next Sprint Plan

Work on setting up the microservices correctly and create a proper communication between them.