Problem Description

We are a bank and would like to modernize our Account Management system in our Retail Banking. We would like to use **Micro Service** architecture to replace the monolith application.

The system is supposed to contain the following functionalities:

- Create a Savings Account for the Customer
- Deposit money (e.g. 100$) to Account
- Withdraw money (e.g. 50$) from Account
- Read Available Balance
- List last 10 transactions of the Account

Note:

The application is only within the Account domain, while the other domains, such as Customer domain, are provided by other systems/applications within the bank.

Back-end

Requirements

You are expected to create a solution that exposes REST API's for the mentioned functionalities by using a **test driven** approach.

The REST API must be able to handle exceptions such as invalid input, as well as business validations from a common sense perspective.

The REST API must be documented with proper API specifications.

Tips

- Choose an application architecture that you are familiar with, all the necessary layers in your chosen architecture should be reflected with at least one relevant working functionality/implementation.
- Focus on the domain modelling and API design instead of making a complete application with everything.
- As the detailed business validation rules are not given, come up with some validations where you think making sense.
- Please do not spend more than 3 hours on coding this assignment, plan your time on the most important part you would like to present.

Technical requirements

- Java 8/11/17
- Junit 5
- Spring/Spring-boot
- Maven/Gradle

Hand-in

The completed code can be delivered in a publicly shared GitHub repository or via email.