Online Gradebook

Analysis and Design Document

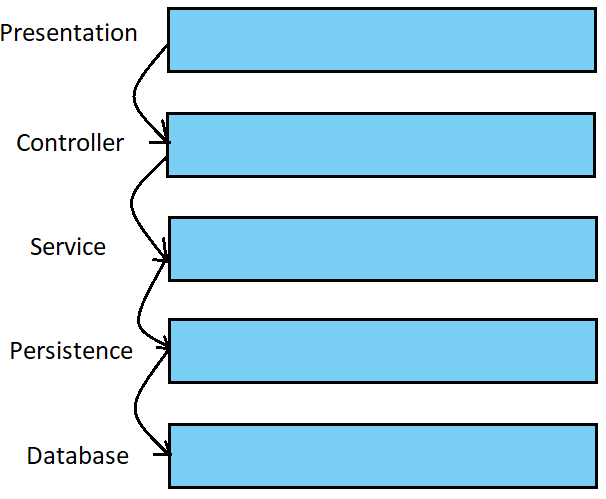
# Elaboration – Iteration 1.1

# Domain Model

# Architectural Design

## Conceptual Architecture

I decided to decompose my project into 5 (vertical) layers: view, controller, service, persistence and database. The view is represented by the browser webpage, and the controller “listens” to the user input and sends the required operation to the service layer (business logic). There a check is done whether the requested data is valid (mark is between 1 and 10, there is such person in the database, etc.) and then through the persistence layer the needed data is extracted from the database by using the adequate operation (CRUD – create, read, update, delete).



## Package Design

## Component and Deployment Diagrams

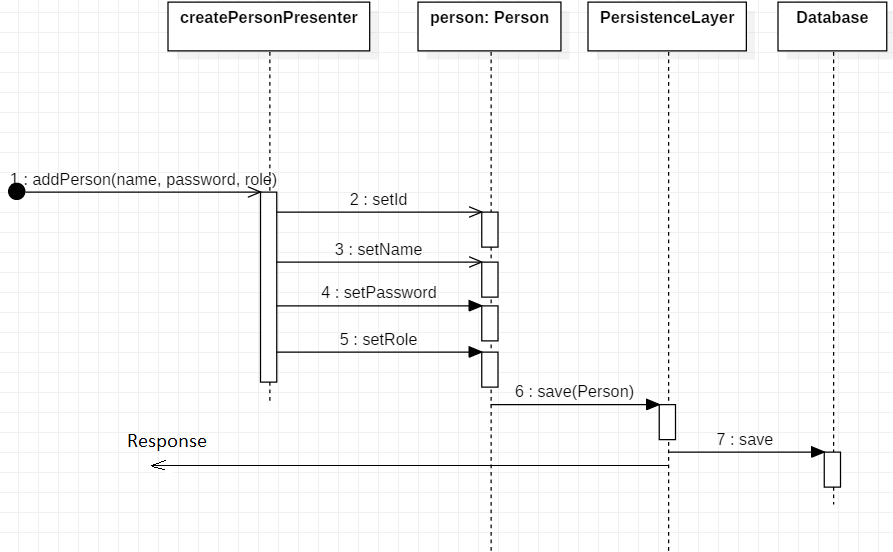
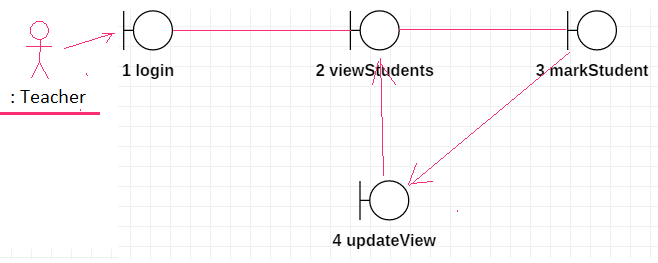
# 

# Elaboration – Iteration 1.2

# Design Model

## Dynamic Behavior

## 

******

## Class Design

# Data Model

# Test Strategy

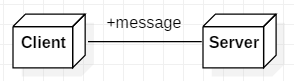
My project consists of 2 parts: Client side and server side, both parts will be tested as follows:

1. Firstly I will test the server side using Junit tests (all the methods from the business logic)

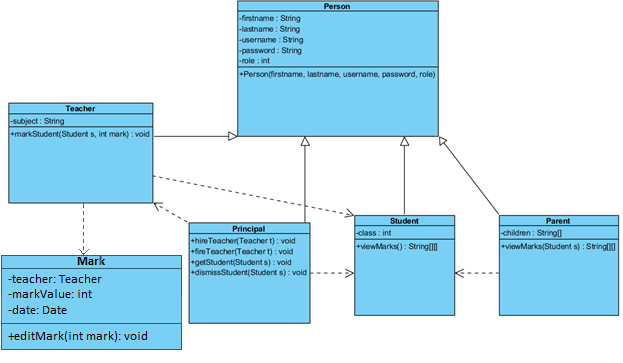
2. Then when I put together the front-end and the back-end I will use Postman tests. After the 2 components are linked together, the whole application will be tested.

# Elaboration – Iteration 2

# Architectural Design Refinement



# Design Model Refinement

**

Compared to the previous design, I added the “Mark” class

# Construction and Transition

# System Testing

As I mentioned above, most part of the testing will be focused on the methods from the business logic followed by an integration test where the components are put together. Small unit tests will be applied also on the methods from the data access layer to ensure the methods used in the business logic work correctly. So if a method gives an incorrect result, then for sure the problem is in the method from the business logic.

# Future improvements

In general, no program is final. They can be improved all the time. This program is not an exception either. The list of possible improvements is the following:

* Enable the possibility of creating / changing the principal so that all the schools could use this program
* Enable comment section (teachers could write a comment for each mark)
* Viewing personal info of each student (age, grade(class which is part of), … )
* Mailing between users
* Many others