Sports Shop
Analysis and Design Document
Student: Oltean Victor
Group: 30431

Revision History

Date	Version	Description	Author
<dd/mmm/yy>	<x.x>	<details>	<name>

Table of Contents

I. Project Specification

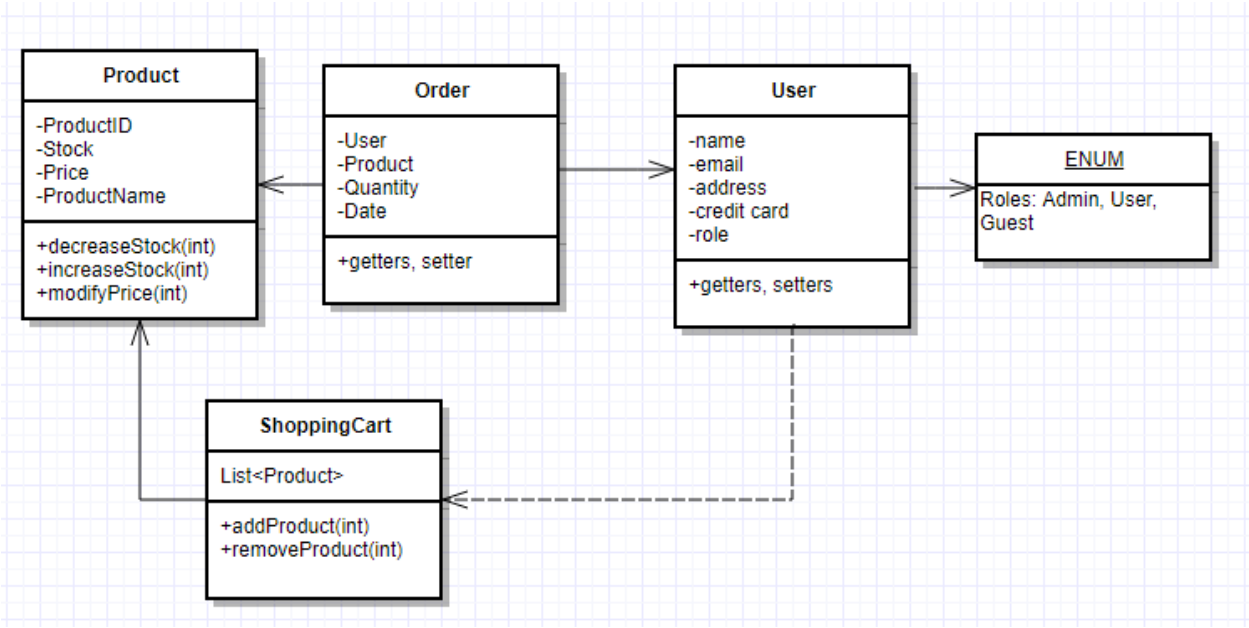
An Online Sportsshop. People can go online an order winter sport articles from the website. There is the possibility to login, in order to access your past orders and purchases, and one can also order without creating an account. If the user logs in, he can choose to save his credit card details and the shipping address, in order to make the next order easier. The user types described previously are the normal user and the guest user.

The shop administrator can add/remove products, confirm orders, increase/decrease the stock of products and cancel orders. He also has access to the phone number, email and address of the users in order to be able to place the order and announce the client that the package is in delivery.

The site's visual will be straight forward: A menu with big categories, and then a smaller menu with sub-categories. Once a sub category has been chosen, the user can choose different sorting methods to sort the products that are on the page, such as: By Price, By Discount, By Size of the article. Each user, be it a logged in one or just a guest will have a shopping cart, and the products selected are added to the cart. At the end of the shopping, the user must go to the cart to complete to transaction. A user can also add items to his favorite list, and receive an email once they are on discount.

II. Elaboration – Iteration 1.1

1. Domain Model



2. Architectural Design

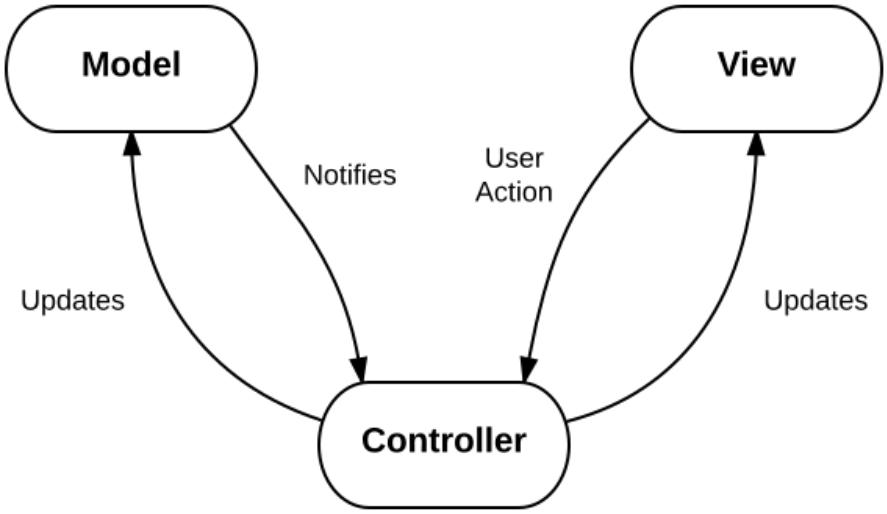
2.1 Conceptual Architecture

The system's conceptual architecture will be MVC, Model View Controller. This architecture is very used nowadays and it is what the Spring Framework in Java implements. It allows for an easy separation of concerns, letting each part of the project do it's part, communicate with other parts.

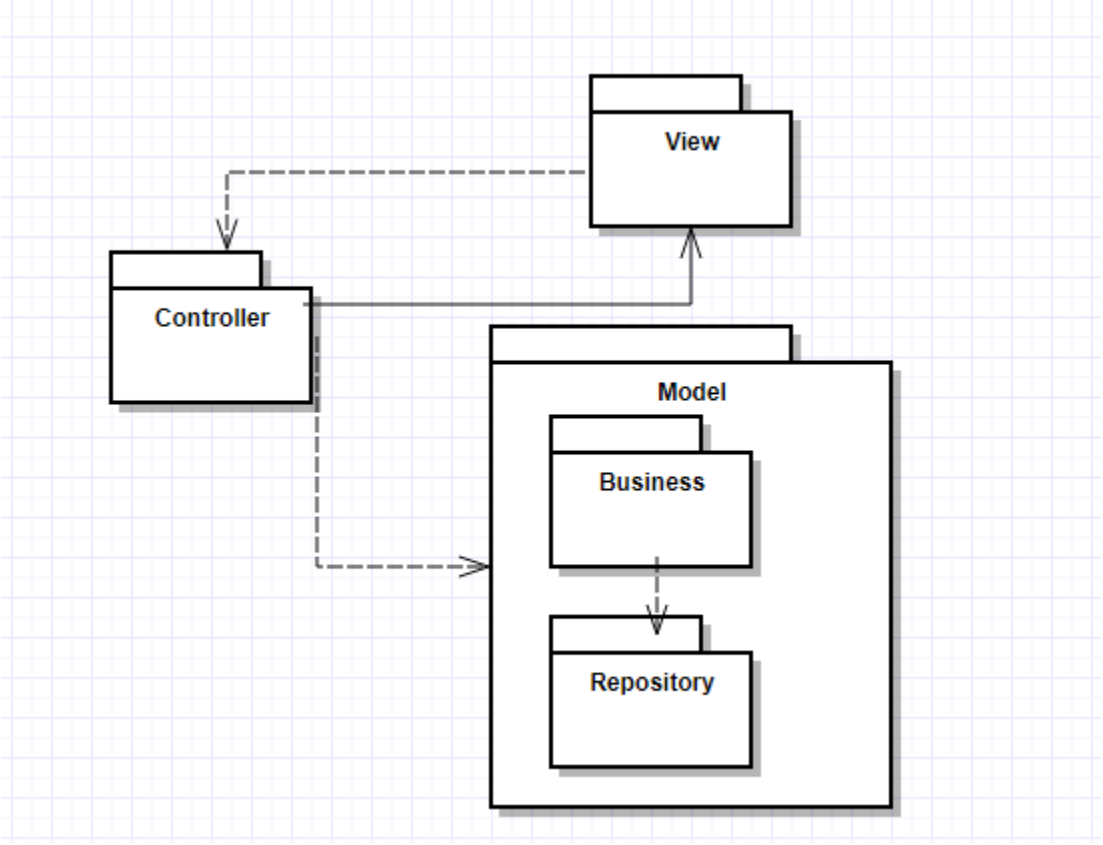
Model – Here, the model of the application is put, including the business logic and the repository. Also, the main components are represented, eg. a User class with fields username, password etc.

Controller – The controller uses the Model in order to update the View, and is being used by the View in order to update the Model. Here, everything which the application can do is put through commands. Our

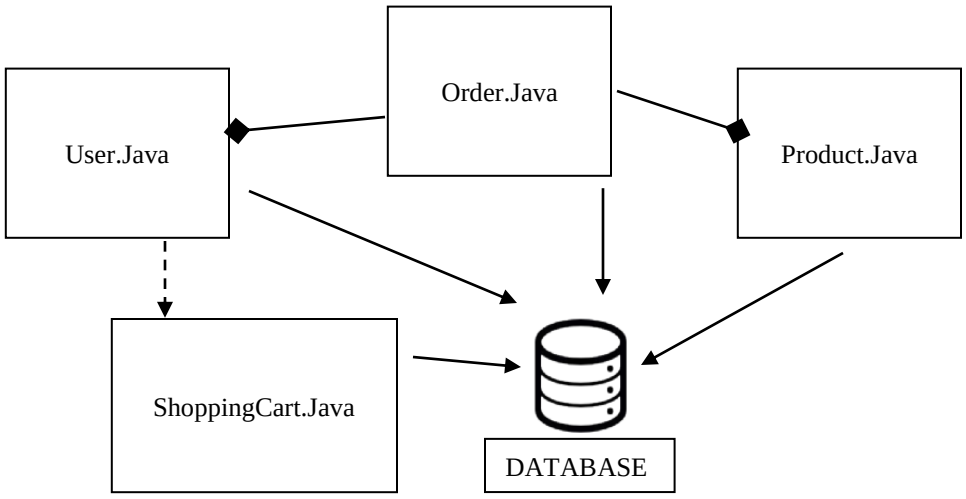
project uses a RESTful API for the controller.
View – The Front End of the application. It uses the controller in order to get the data which it is going to display.

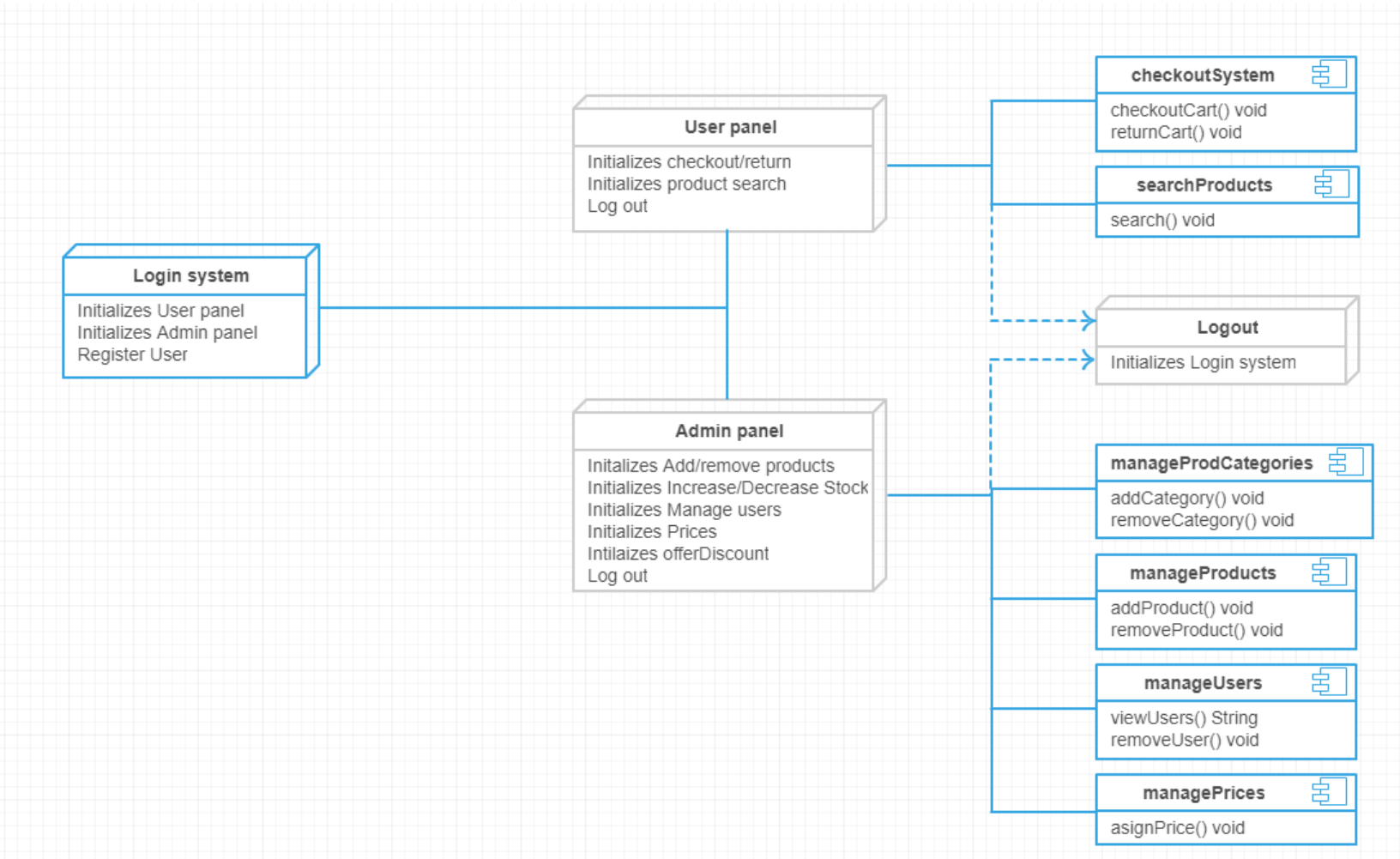


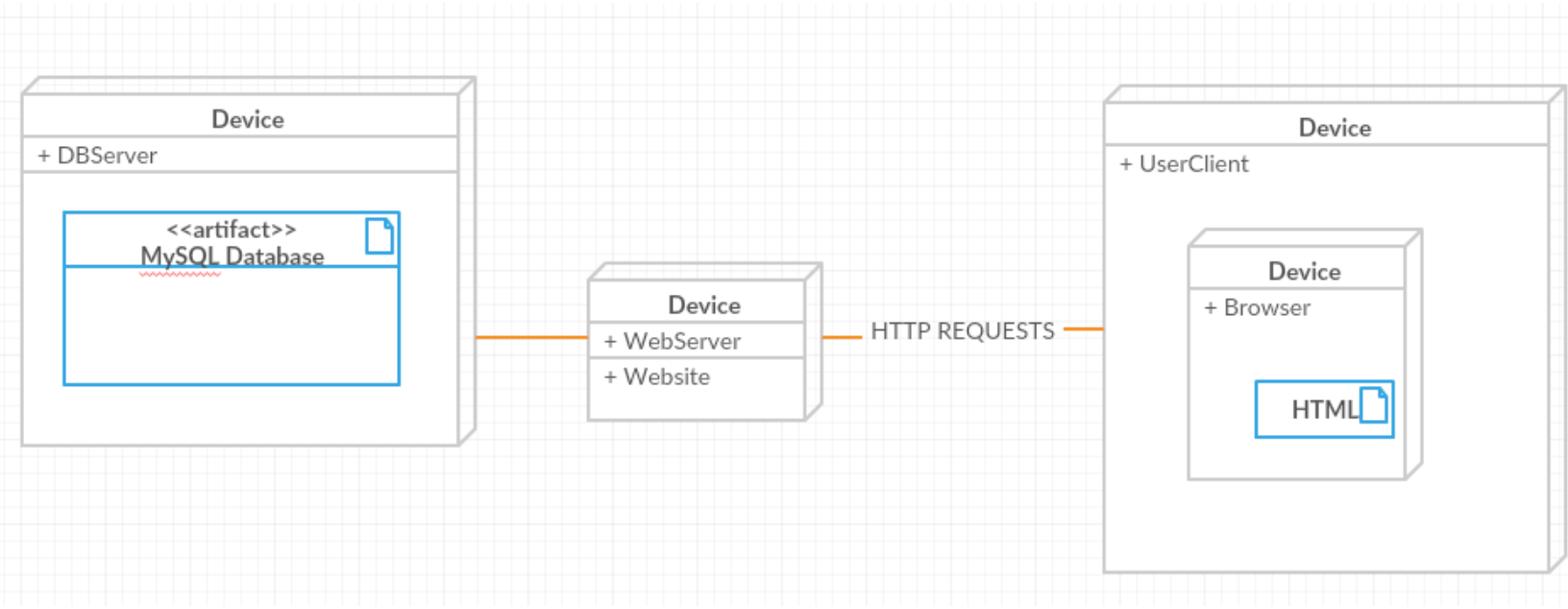
2.2 Package Design



2.3 Component and Deployment Diagrams







III. Elaboration – Iteration 1.2

1. Design Model

1.1 Dynamic Behavior

[Create the interaction diagrams (1 sequence, 1 communication diagrams) for 2 relevant scenarios]

1.2 Class Design

[Create the UML class diagram; apply GoF patterns and motivate your choice]

2. Data Model

[Create the data model for the system.]

3. Unit Testing

[Present the used testing methods and the associated test case scenarios.]

IV. Elaboration – Iteration 2

1. Architectural Design Refinement

[Refine the architectural design: conceptual architecture, package design (consider package design principles), component and deployment diagrams. Motivate the changes that have been made.]

2. Design Model Refinement

[Refine the UML class diagram by applying class design principles and GRASP; motivate your choices. Deliver the updated class diagrams.]

V. Construction and Transition

1. System Testing

[Describe how you applied integration testing and present the associated test case scenarios.]

2. Future improvements

[Present future improvements for the system]

VI. Bibliography