IMTSquare Company

AnyChange Software

Design

v3.0

Revision History

Date	Version	Type	Description	Prepared by	Checked by
21/05/2023	3.0	IFC	Preliminary draft	IMT2 Dev	IMT2 QA/QC

AnyChange
Design

1 Introduction

This document presents the software design of AnyChange. This is the project document that documents the design decisions, design artifacts and diagrams that represent theworking mechanism of the software.

2 Design Structure

Design of AnyChange follows a Model-View-Controller(MVC) architecture. Model layer represents the entities that are persistent across the application, controller layer is responsible for running the business logic of the application by using the model entities and view layer represents the user interface.

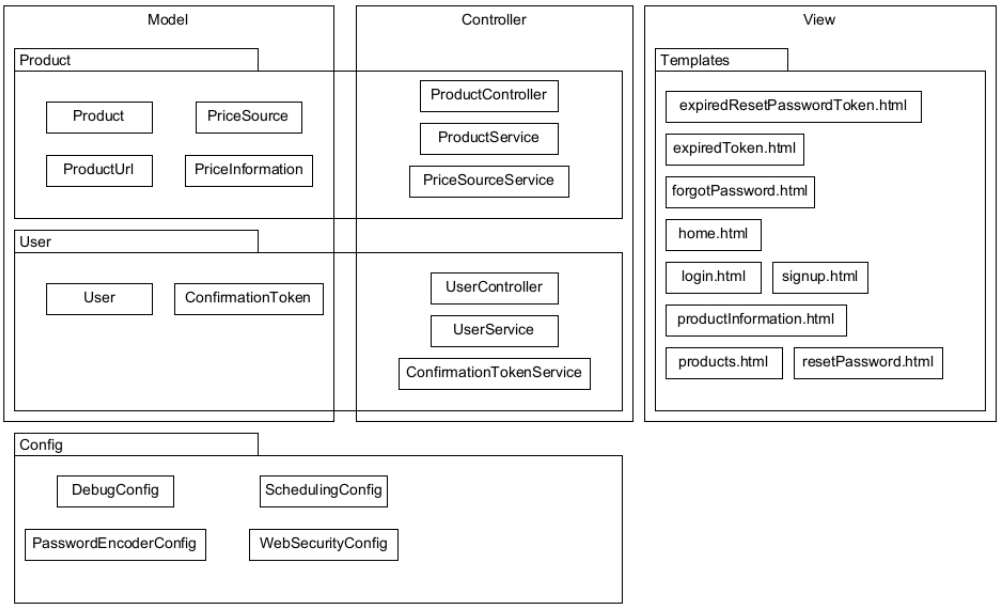

Figure 1. Package Diagram of AnyChange, highlighting Model-View-Controller structure

There are three types of users available: admin, seller and standard user. Admin user is responsible for general administration tasks. Seller users can add new products to the system and remove products they have added. Standard users can search for products which they can add to the monitoring list to be notified during price changes.

3 Subsystems

There are 4 packages that make up AnyChange.

a. User

This package is responsible for the user management logic. It enables registration, confirmation of user activities and authentication.

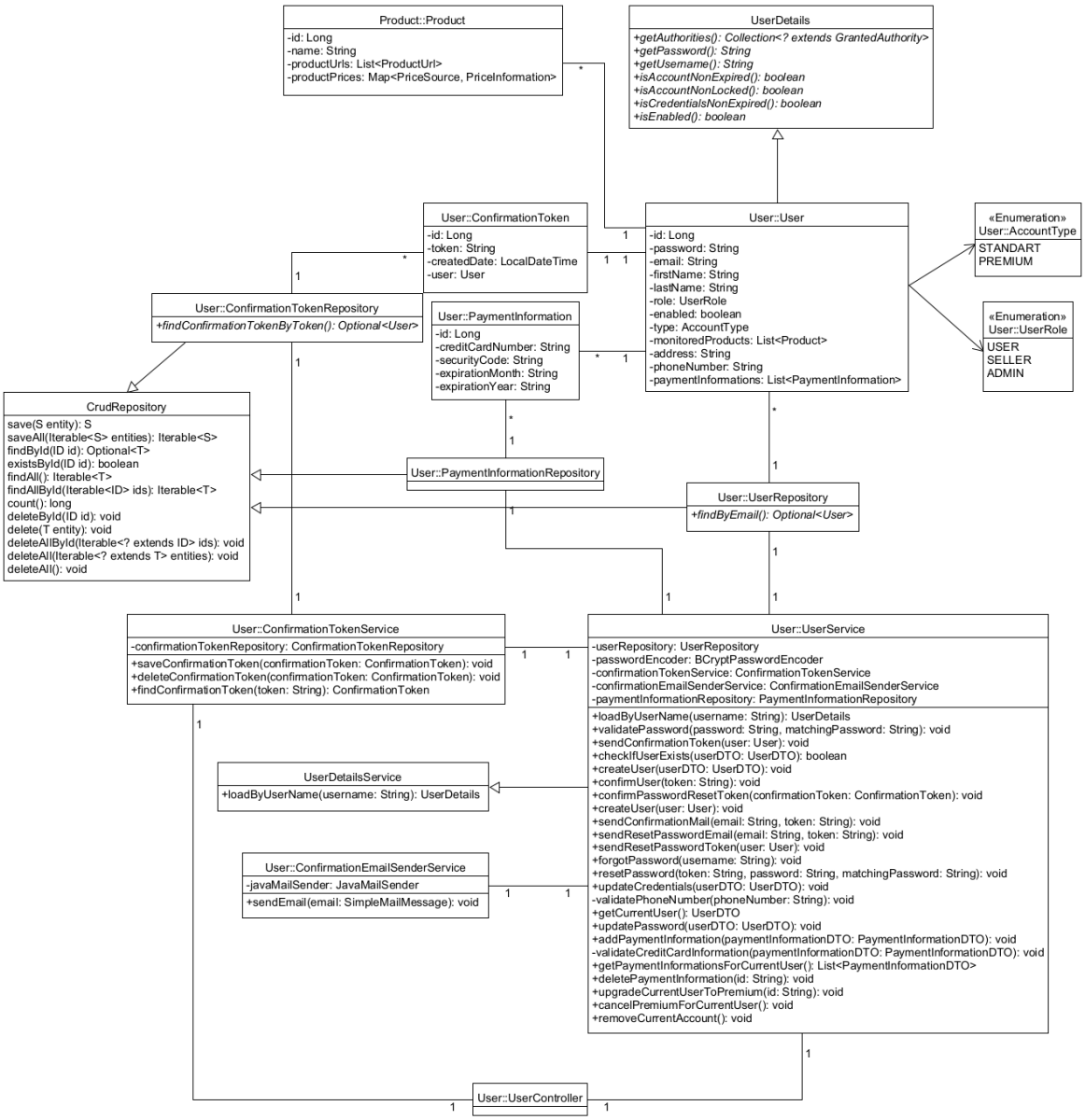

Figure 2. Class diagram of User package

b. Product

This package is responsible for the logic around dealing with products registered to the system, it serves the user product information, it updates the current product prices across registered price sources and it enables user to monitor specific products.

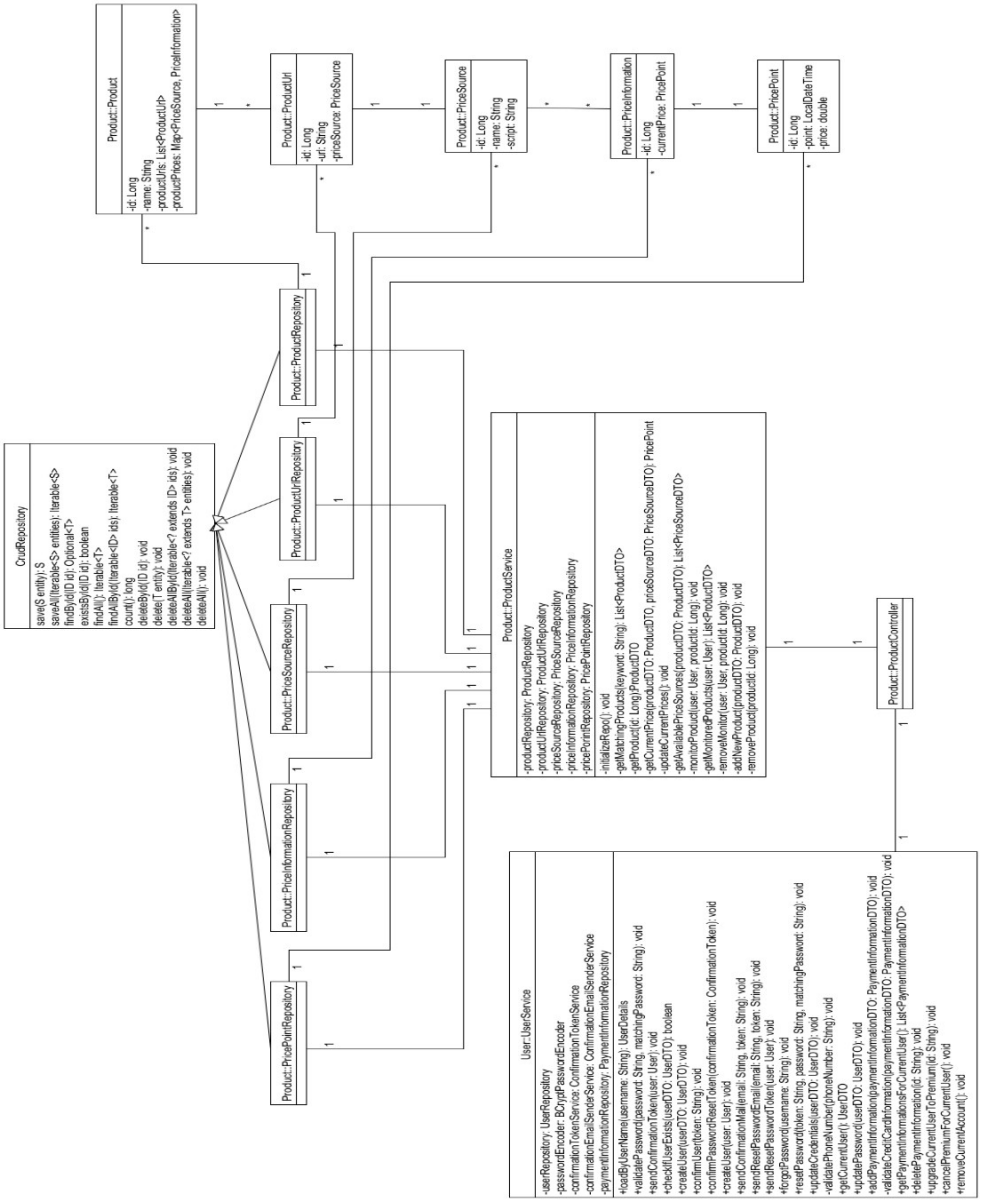

Figure 3. Class diagram of Product package

c. Config

This package is responsible for application wide configurations that are enabled by Spring-Boot framework. It works by supplying configuration beans that extends existing configuration objects.

d. Templates

This package contains Thymeleaf templates that are used for rendering user interface

4 Requirement Realizations

a. UC1-Register to System

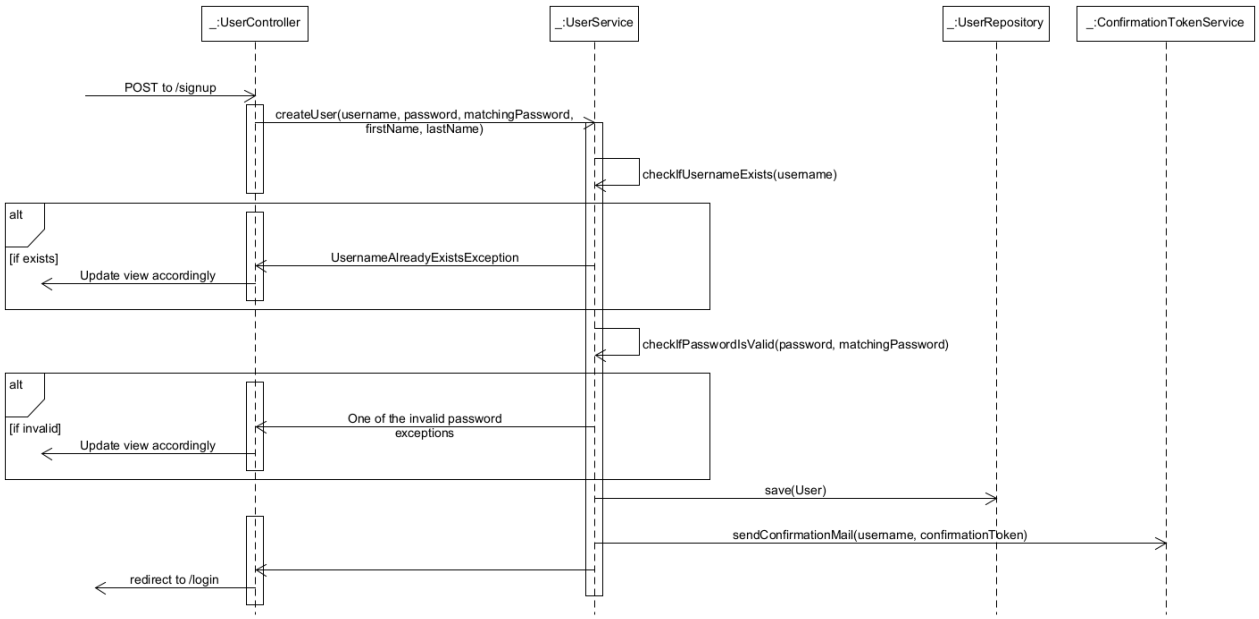

Figure 4. User Registeration sequence diagram

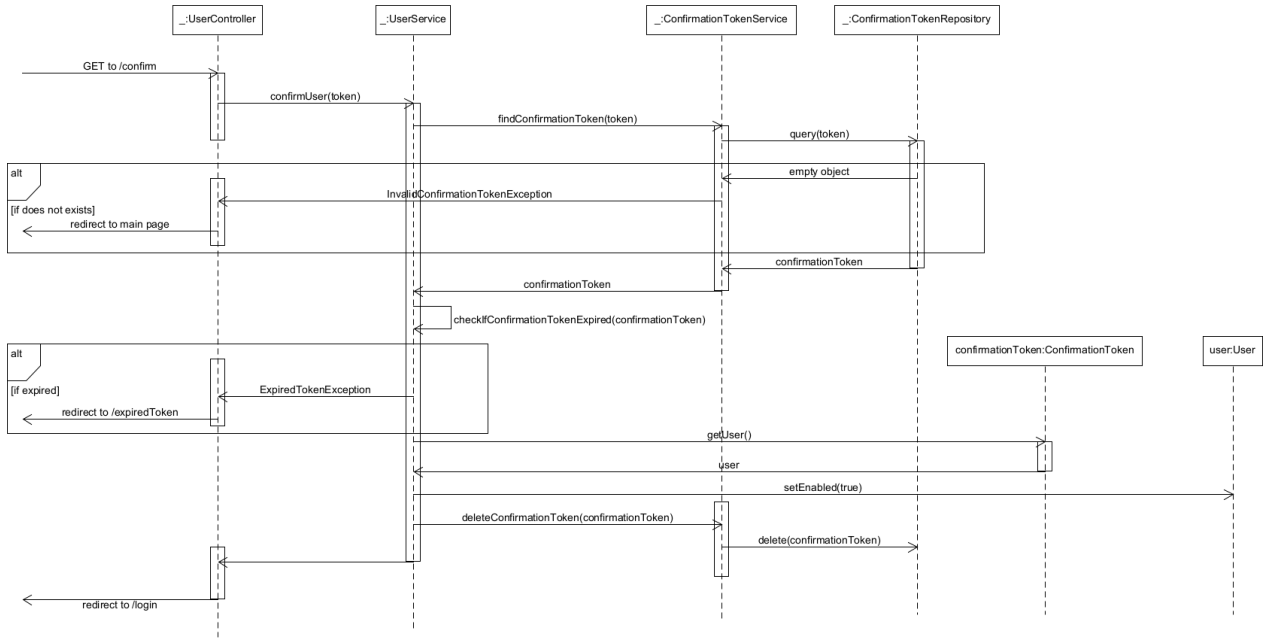

Figure 5. User Confirmation sequence diagram

b. UC6-Manage Product

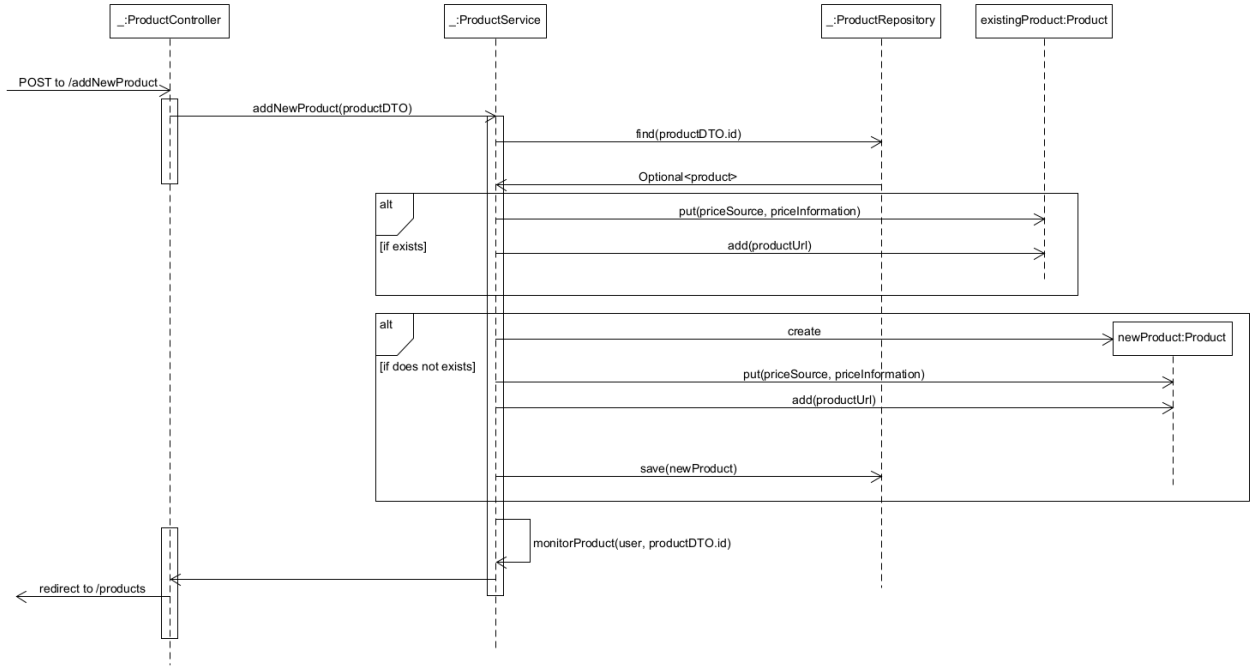

Figure 6. Add new product sequence diagram