

UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

Bellisimo

Requirement and Design Specification

Diana Obo 13134885

1 Background

1.1 Project Background

Bellissimo is a company aimed at providing an online platform for customers to browse clothing as well as food catalogues provided by the business located in Hatfield. Information about specials and promotions will be published on the online platform.

1.2 Purpose

The purpose of this document is to present the reader with a detailed description of the Bellissimo system. It will delve into the purpose and features of the system, the various interfaces of the system, the capabilities of the system, as well as the constraints under which the system must operate. The content of this document is intended for both the various stakeholders and the developers of the Bellissimo system.

1.3 Visions and Scope

The core of the system will be catalogues of items and their prices. Since Bellissimo is involved in clothing and food, the catalogues will have to ensure that these lines are well maintained. Sales and specials in each line will have to be accounted for and managed. The scope of the system is to ensure that the latest information is being provided about items and their prices.

1.4 Architecture Design of Bellissimo System

At the highest level of granularity, the Bellissimo system is based on Monolithic architecture. Second level of granularity can be visualised as to be based on model-view-controller (MVC).

1. Architectural Patterns of the Bellissimo Systems: The architectural patterns of the Bellissimo system are focused and narrowed to patterns. Therefore the following architectural patterns are identified:

- (a) Monolithic architecture
 - (b) Model-view-control architecture patterns
2. Quality Requirements of the Bellissimo Systems: should include but not limited to the following:
- (a) Performance
 - (b) Integrability
 - (c) Availability - that the latest information about the catalogues are provided
 - (d) Maintainability - ensure that the catalogues are well maintained
 - (e) Scalability
 - (f) Reliability - sales and specials in each line should be accounted for and managed
 - (g) Security - only an administrator can access the administrator interface to create, update and delete food and clothing catalogues.

1.5 Design Requirement

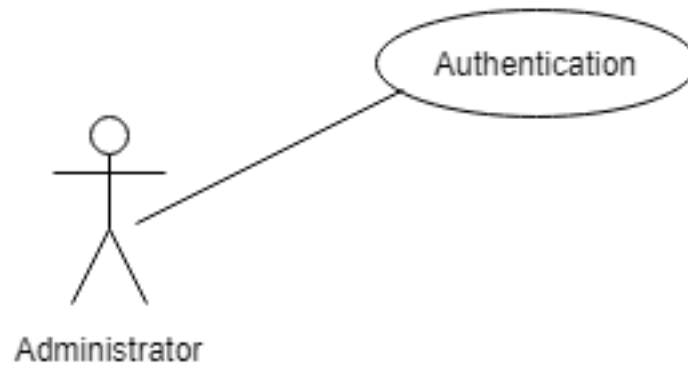
The system should be designed in such a way that there are two different interfaces that depict the clothing catalogue and the food catalogue.

There should be at least two interfaces. An admin interface to maintain the catalogues and an anonymous user interface that will allow people to interactively view the catalogues. The prices of each item in the database should be displayed along with the image of the item.

2 User Management Module

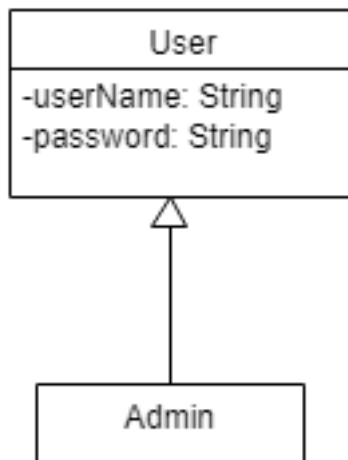
2.1 Scope

The scope of the user management module is shown in Figure 1. The user management module is responsible for maintaining information about the registered administrators of the system. Administrators can manage information about clothes and food catalogues.



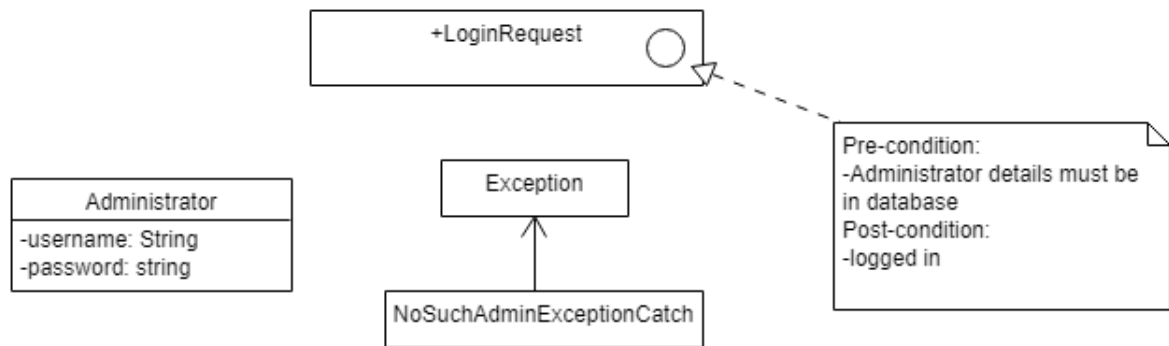
- Administrator: A default administrator account will be on the database and the administrator need to only login to access the administrator interface. The administrator of the system will receive a guideline pdf on the details of the administrator password and username and how to change the details if required.

2.2 Domain Model



2.3 Service Contracts

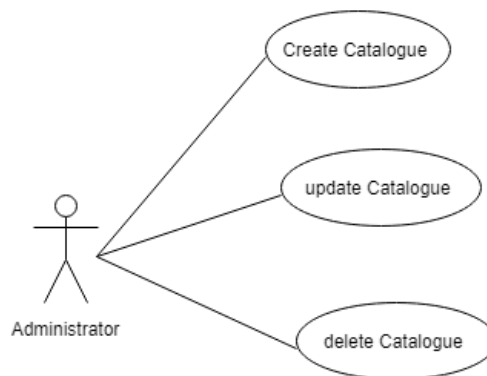
The following services should be provided
Precondition: adminName is a registered user
Exception: If adminName is not a registered user throw noSuchAdmin exception
Postcondition: No change



3 Catalogue Management Module

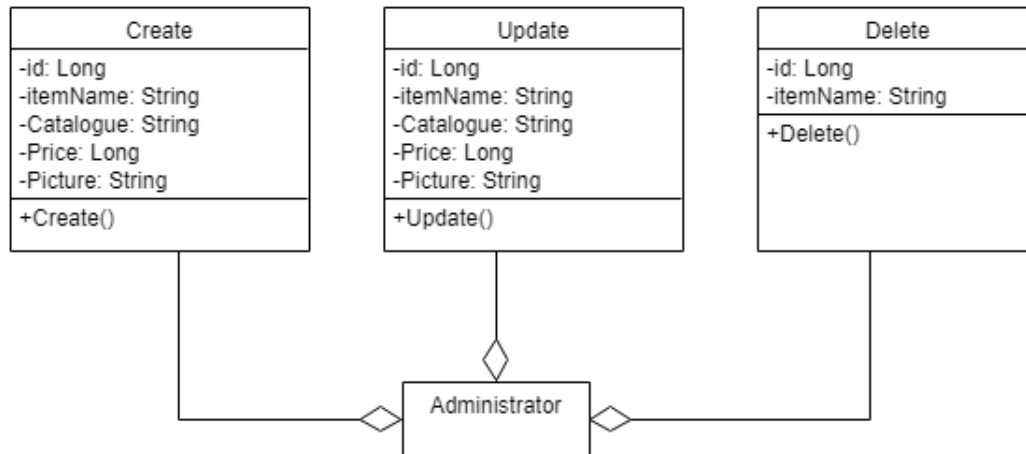
3.1 Scope

The catalogue management module will allow the administrator to add, remove and update any items in the catalogue list. The catalogue management module will also allow the administrator to add specials to the module. Specials can be applied to singular items or groups of items depending on a special group configuration.

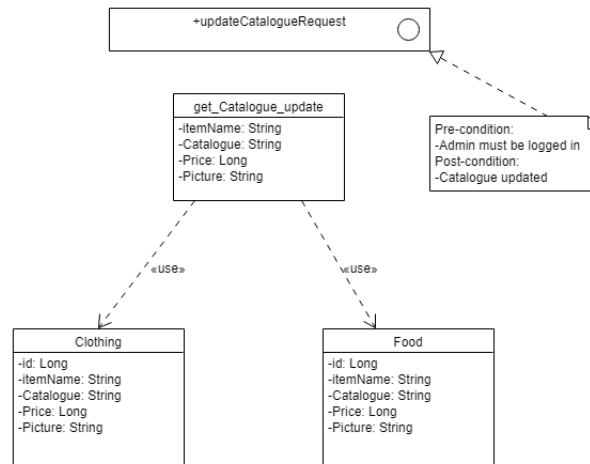
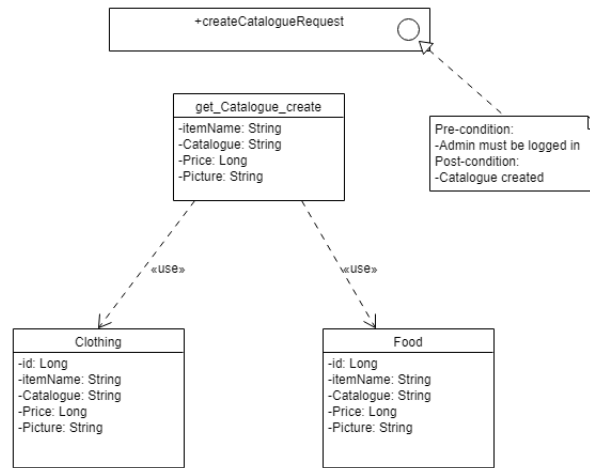


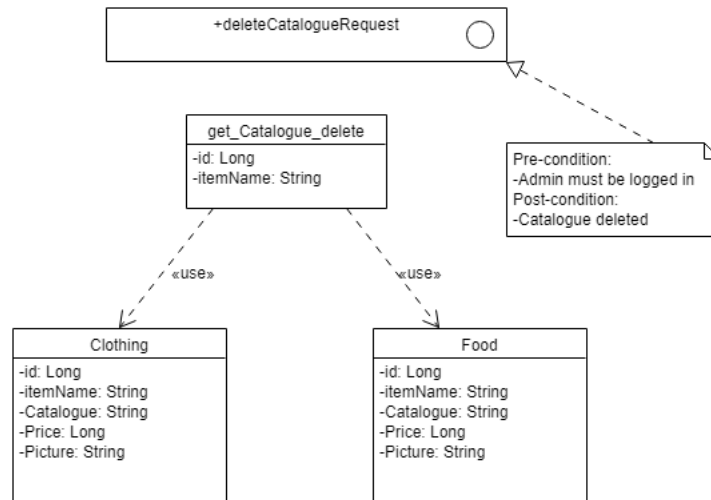
3.2 Domain Model

The domain model for the Catalogue Managemen system is simple. It is simply a description how the catalogues can be modified by either being created, being updated or being deleted.



3.3 Service Contracts

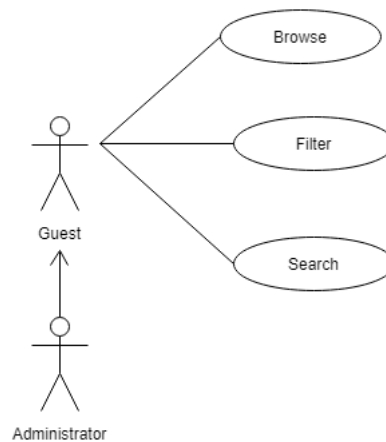




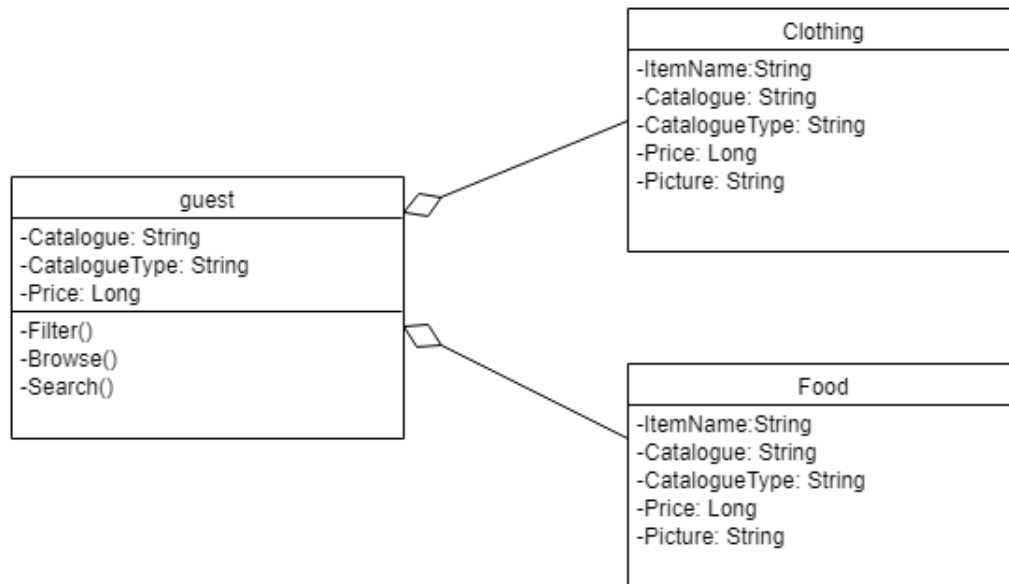
4 View Module

4.1 Scope

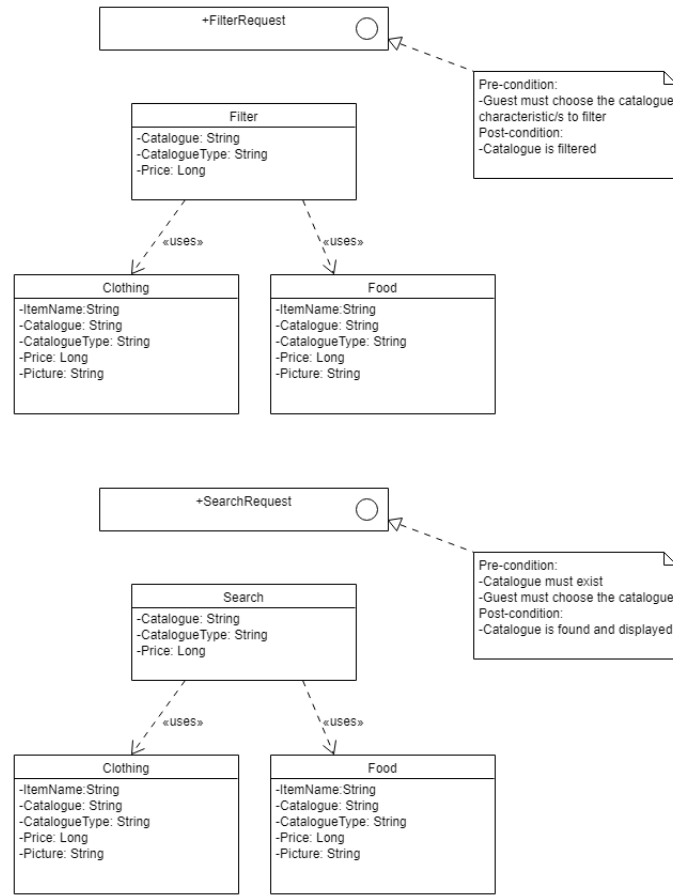
The view module will allow the guest user to browse, search and alter the catalogues.



4.2 Domain Model



4.3 Service Contracts



5 Architectural Technologies

The web application will consist of two subsystems that communicate via HTTP using REST Framework. The Java/Spring Boot application will be known as the "backend" application. The HTML5/Angular2 application will be known as the "frontend" application. The backend application is expected to communicate with the database and use Hibernate which can be imported into Maven, a dependency management tool whereas the frontend application will be hosted in the browser and NodeJS is expected to manage packages required for the application to run successfully.

5.1 Technologies

The following technologies will be used to implement the system:

- Html 5 (Html and Bootstrap CSS)
- Angular2
- NodeJS
- Spring Boot
- PostgreSQL
- Apache Maven
- Git (Github)

5.2 Monolithic Implementation

The monolithic implementation will consist of one backend spring boot application that will be deployed as a single unit. Apache Maven will be used as the dependency management tool.