Assignment 1
Server Side Technologies – SE4

Manuel Escrig Ventura (A00180182)

GRAILS WEB APPLICTION (WEB STORE)

November 11

manuelescrig@gmail.com

Table of Contents

Table of Figures

1. Introduction

The goal of this assignment is to develop a web application using Grails. We expect to have 3-5 domain classes in the application including test cases for your domain classes.

In this case I have developed a Web Store, I choose a this application because I have in mind to finish this application with more classes and controllers and uploaded to a web server so it can be online and it can be something useful.

To develop this project I have been using STS (SpringSource Tool Suite) for Mac OSX and the version of grails is 1.3.7.
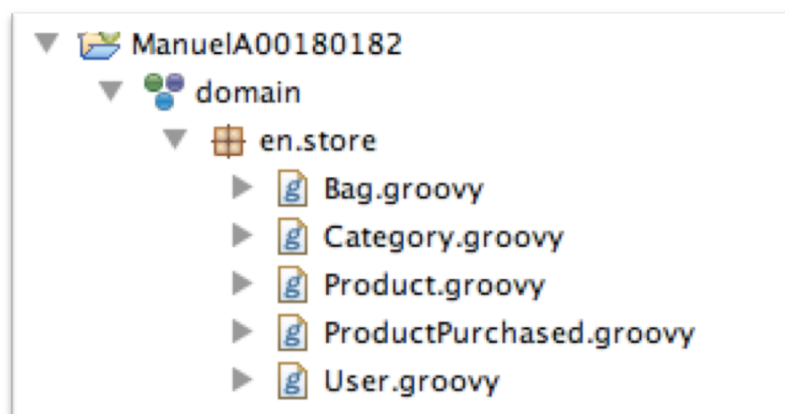
List of domain classes:

Figure 1 - List of domain classes
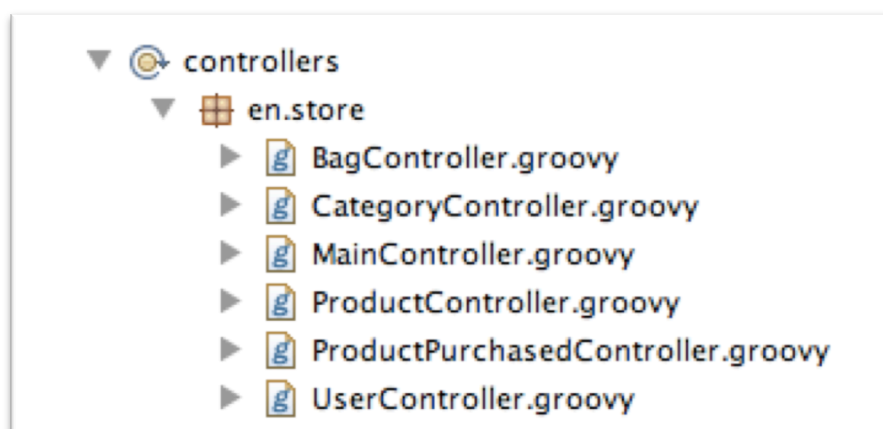
List of domain controllers:

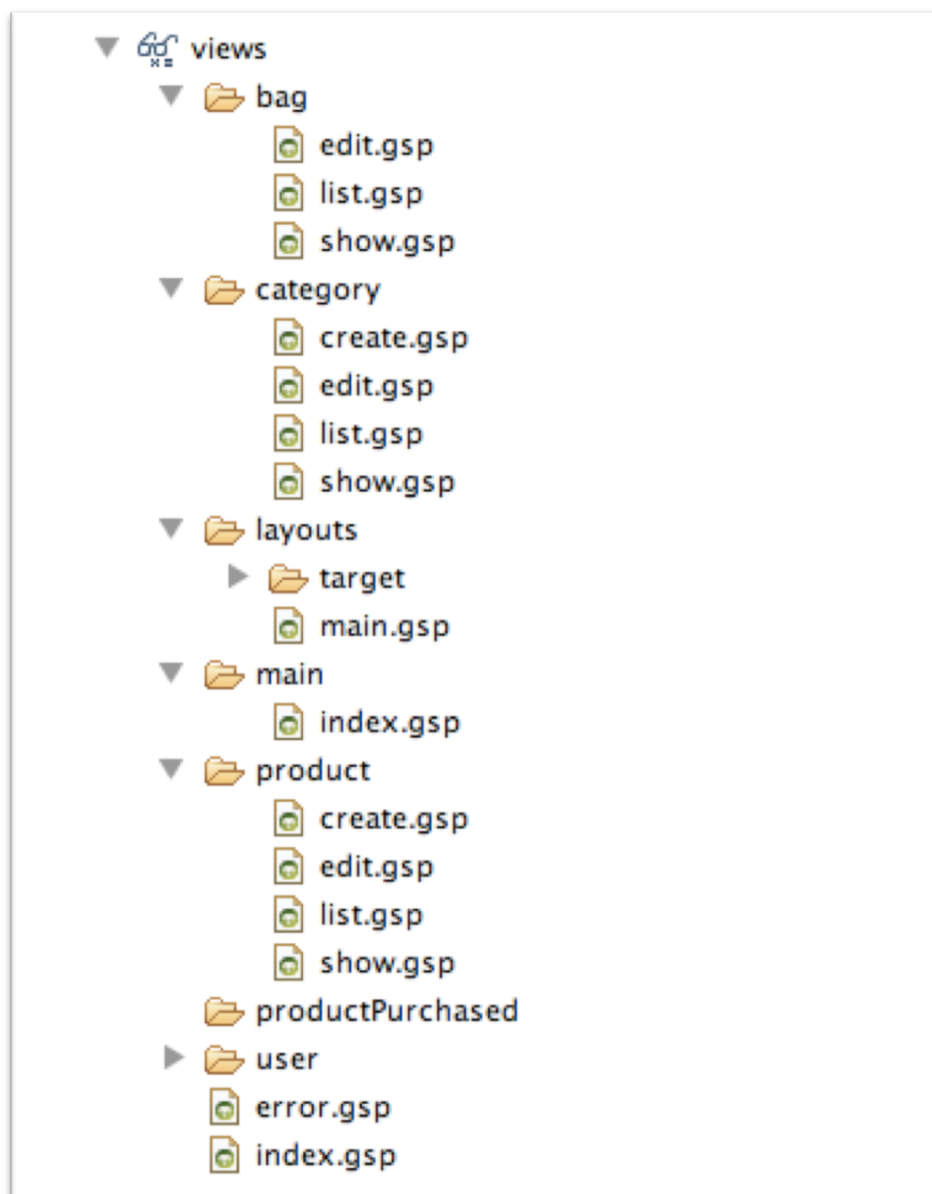**Figure 2 - List of domain controllers**

List of views:

```
▼ 🔍 views
    ▼ 📂 bag
        📄 edit.gsp
        📄 list.gsp
        📄 show.gsp
    ▼ 📂 category
        📄 create.gsp
        📄 edit.gsp
        📄 list.gsp
        📄 show.gsp
    ▼ 📂 layouts
        ▶ 📂 target
        📄 main.gsp
    ▼ 📂 main
        📄 index.gsp
    ▼ 📂 product
        📄 create.gsp
        📄 edit.gsp
        📄 list.gsp
        📄 show.gsp
    📂 productPurchased
    ▶ 📂 user
    📄 error.gsp
    📄 index.gsp
```

Figure 3 - List of the views

2. Screen shots.
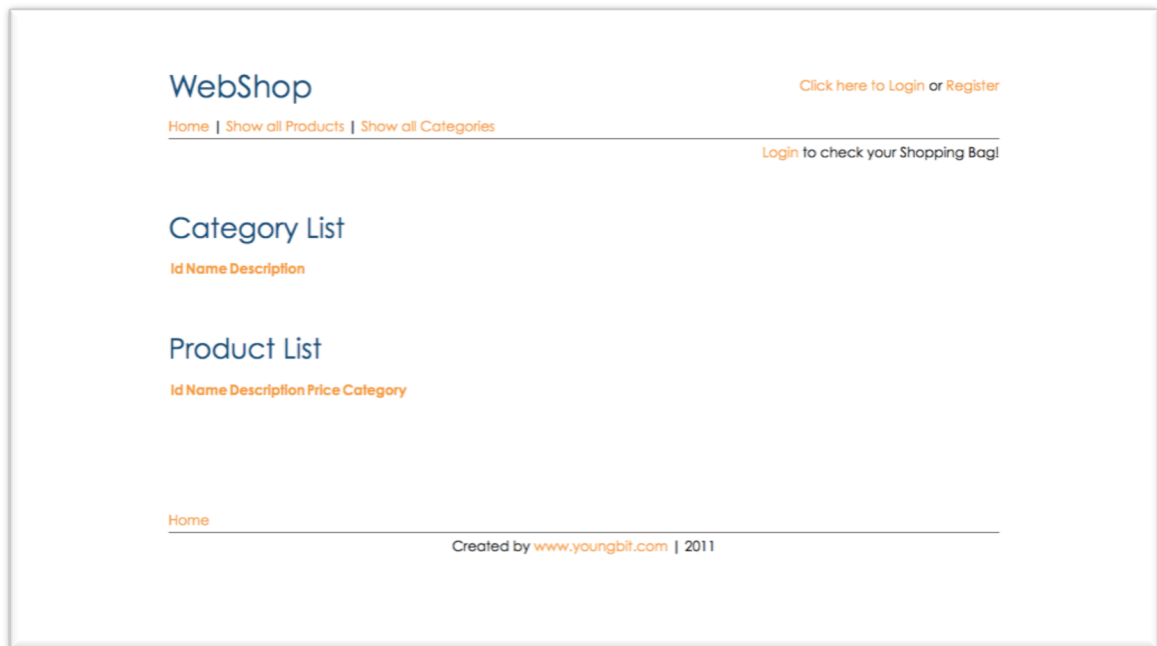
Figure 4 - Controller: Main
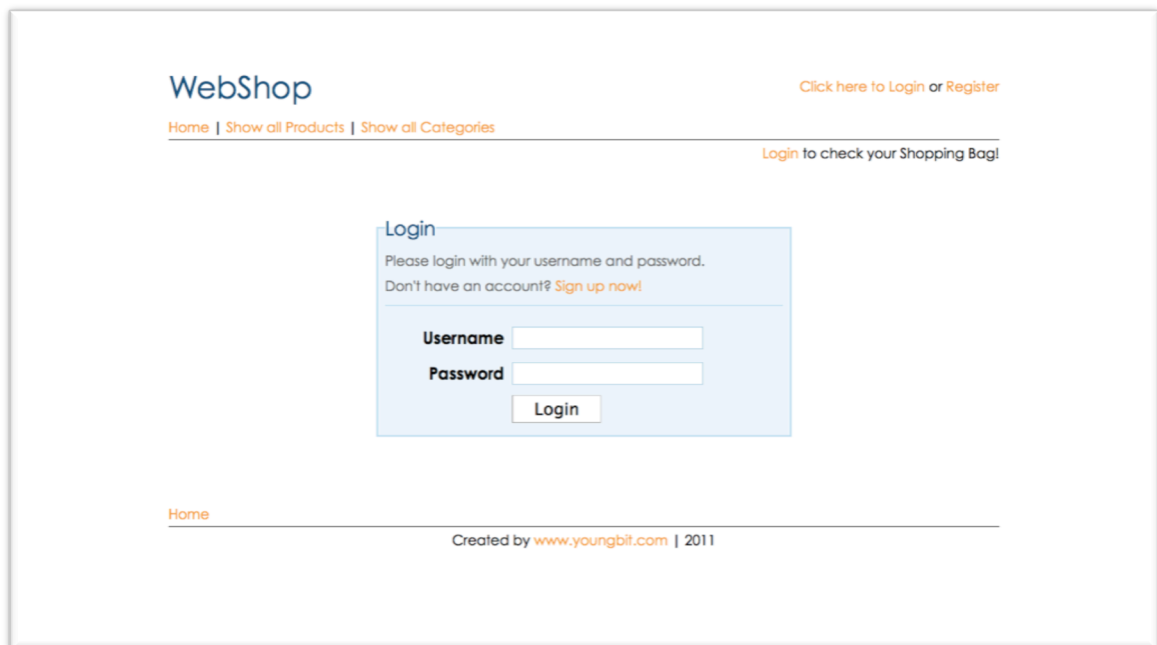
Figure 5 - Controller: User - Action: Login

Figure 6 - Controller: User - Action: Register

Figure 7 - Controller: User - Action: Register

Figure 8 - Controller: Category - Action: Create

Figure 9 - Controller: Category - Action: List
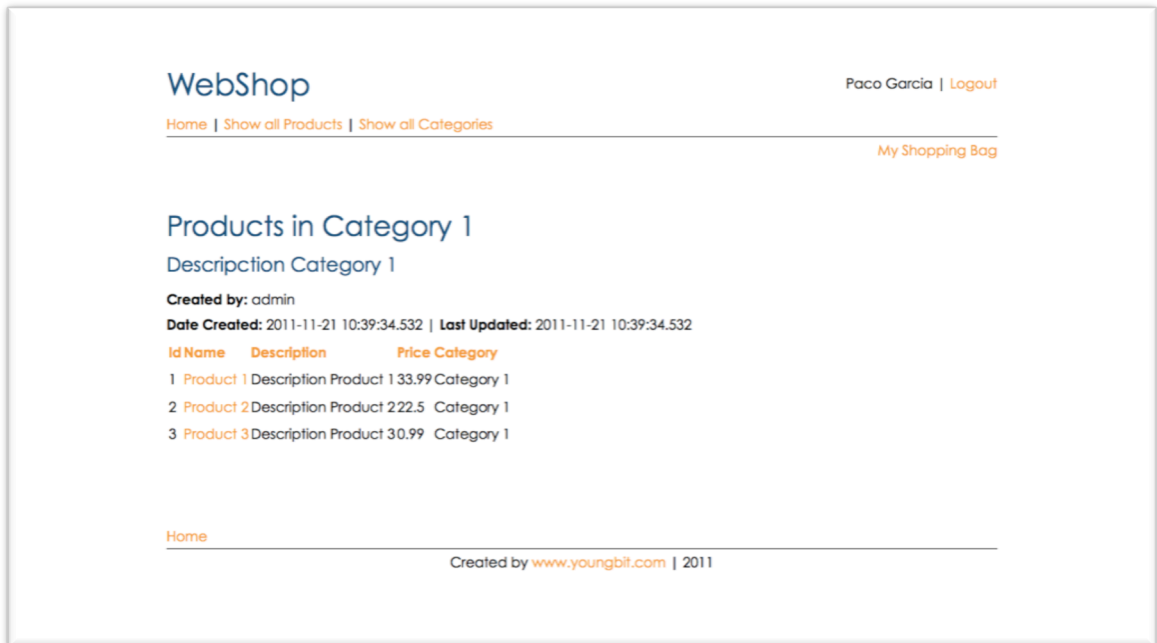
Figure 10 - Category: Controller - Action: Show
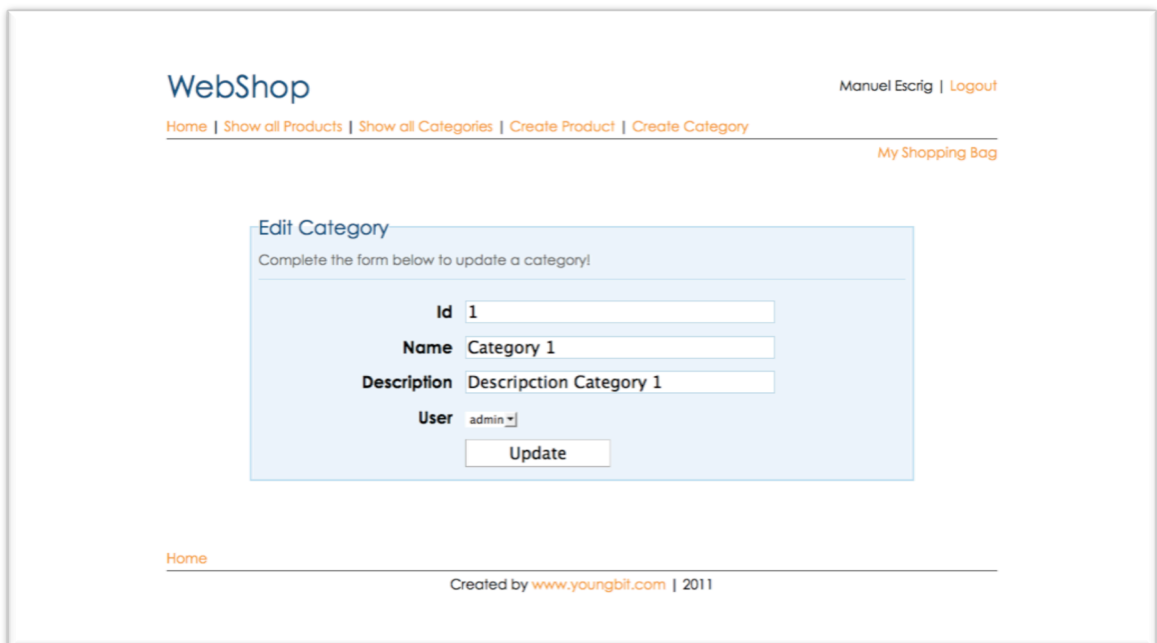
Figure 11 - Category: Controller - Action: Edit

Figure 12 - Controller: Product - Action: Create

Figure 13 - Controller: Product - Action: List

Figure 14 - Controller: Product - Action: Show

Figure 15 - Controller: Product - Action: Edit

Figure 16 - Controller: Bag - Action: Show

Figure 17 - Controller: Main – Action: Index (admin)

3. Overview.

The application has 5 domain classes:
- **User:** Used to store all the information about the users and the admin, the user that has as a username "admin" has all the control of the application and will be able to create new categories an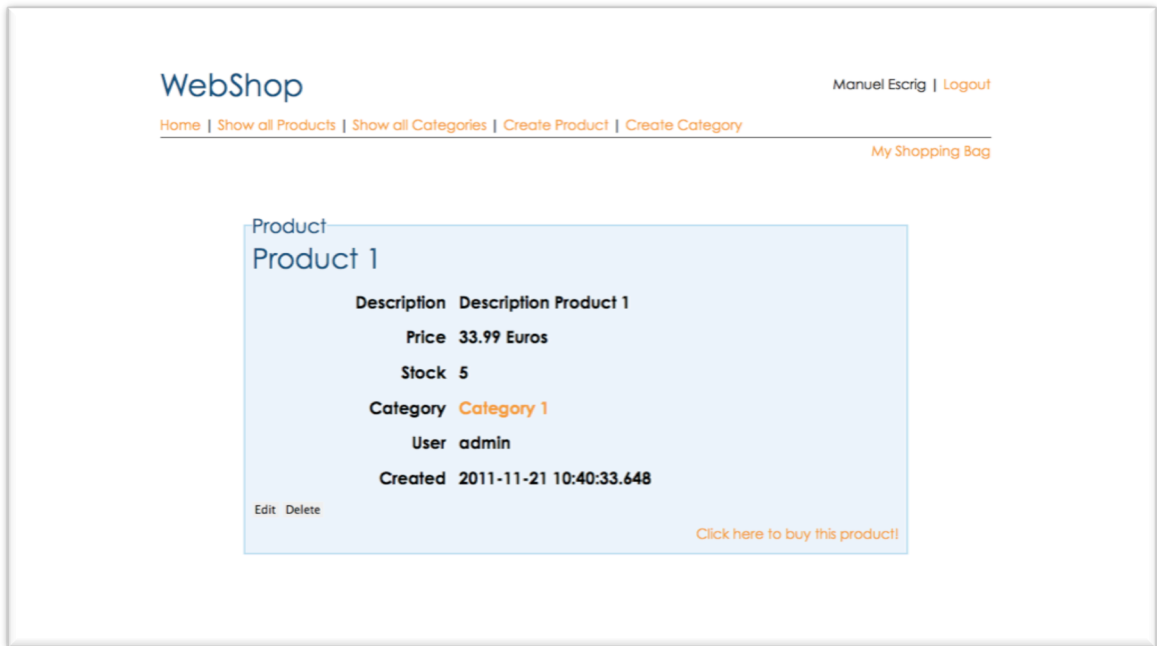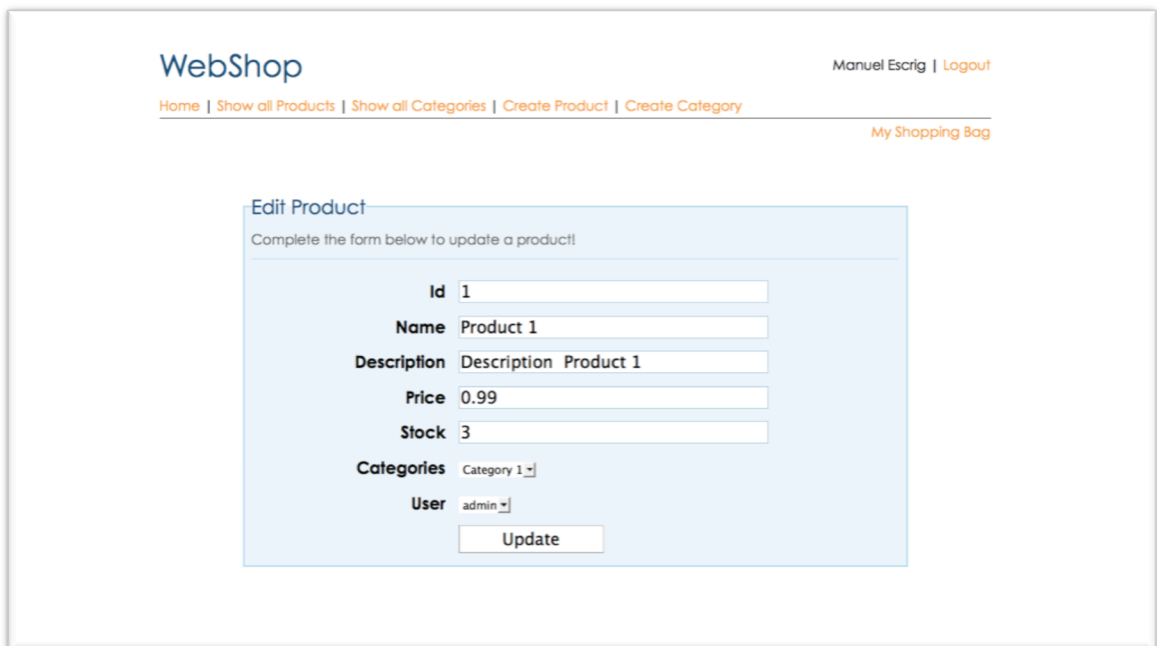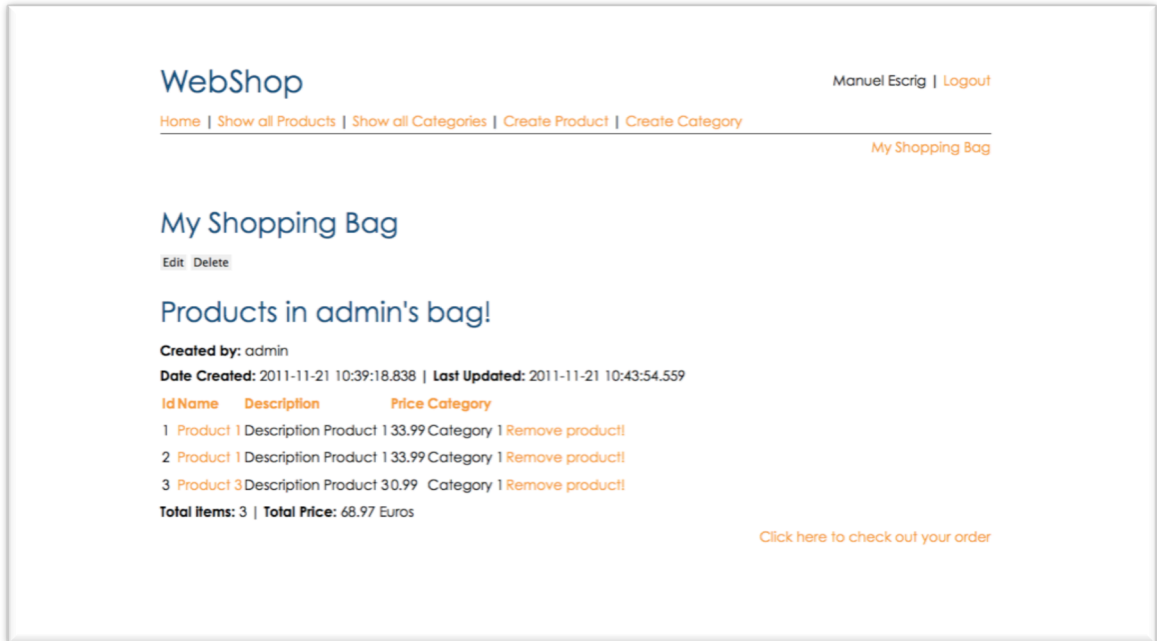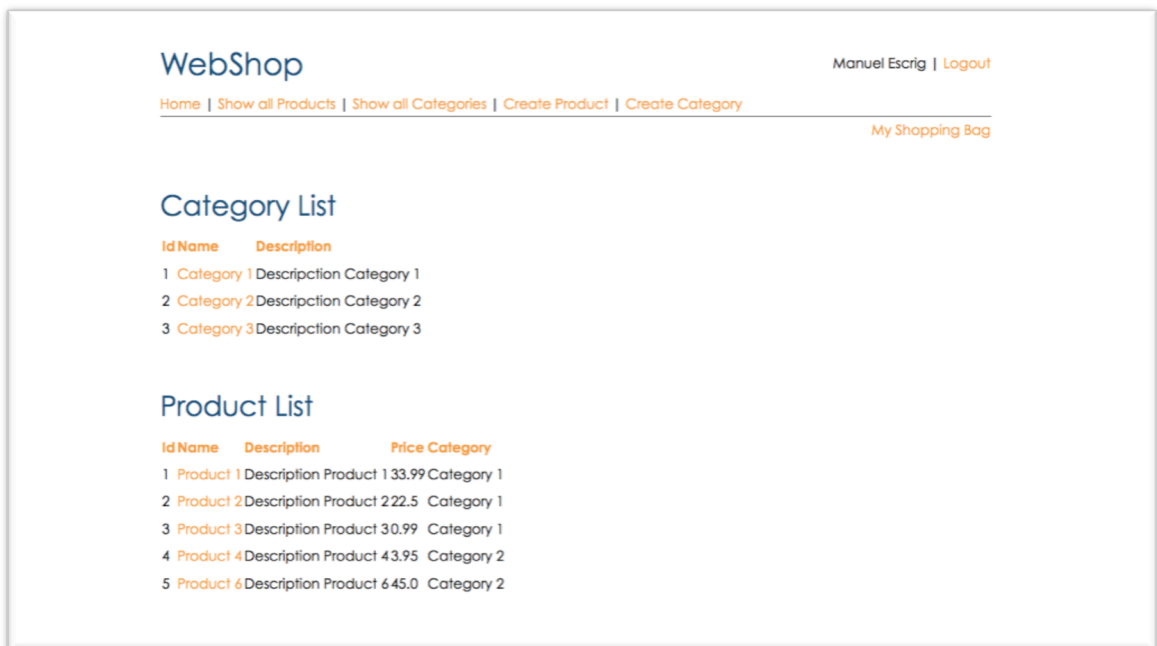d new products as well as edit them. The other users they just can buy products and see the items in the different categories
- **Product:** Has the information of the different items of the store, each product belong to one category and one user, and has different information such as name, description, price, stock…
- **Category:** Each category can have many products and is useful to see the different products of the store because you can list them throw different categories. It has a name and a description and belongs to one user.
- **ProductPurchased:** This domain class is useful when we click on "buy a product" because then we generate a new object related to the product that we want to buy, and related to the user (bag) who wants to buy it and it also stores the price (just in case it can changes) and the quantity. (I have the idea of implementing the option of when you buy one product it takes one from the stock)
- **Bag:** It has a list of the ProductPurschased that the user wants to buy, each bag belongs to one user and it store the number of products and the total of the price of the purchase.

4. Assumptions creating the web application.

I had different assumptions while I was developing this application, the main one has been the test cases, some of them I didn't have enough time to develop them well, I wanted to spend more time with the test cases but I couldn't.

Another assumption is that I would like to spend more time developing the application and I will create a new domain class called picture, because I think is important to have pictures of the products so that will be great.

Then I would like to integrate more information in those classes such us number of items that users has purchased and staff like that, so we can have statistics of the sells.

And finally spend time developing a better design of the website.