

Assignment 1

Server Side Technologies – SE4

Manuel Escrig Ventura (A00180182)

GRAILS WEB APPLICATION (WEB STORE)

November 11

manuelescrig@gmail.com

Table of Contents

1. Introduction.....	3
2. Screen shots.....	5
3. Overview.	12
4. Assumptions creating the web application.....	13

Table of Figures

Figure 1 - List of domain classes	3
Figure 2 - List of domain controllers	3
Figure 3 - List of the views	4
Figure 4 - Controller: Main	5
Figure 5 - Controller: User - Action: Login.....	5
Figure 6 - Controller: User - Action: Register.....	6
Figure 7 - Controller: User - Action: Register.....	6
Figure 8 - Controller: Category - Action: Create	7
Figure 9 - Controller: Category - Action: List.....	7
Figure 10 - Category: Controller - Action: Show.....	8
Figure 11 - Category: Controller - Action: Edit.....	8
Figure 12 - Controller: Product - Action: Create	9
Figure 13 - Controller: Product - Action: List	9
Figure 14 - Controller: Product - Action: Show	10
Figure 15 - Controller: Product - Action: Edit.....	10
Figure 16 - Controller: Bag - Action: Show	11
Figure 17 - Controller: Main – Action: Index (admin)	11

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:

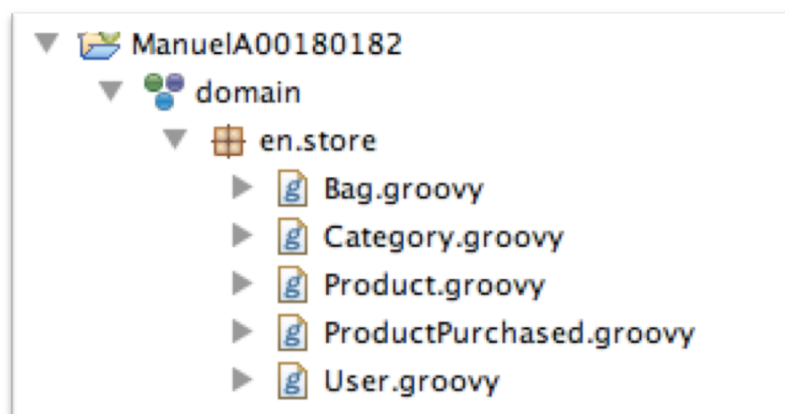


Figure 1 - List of domain classes

List of domain controllers:

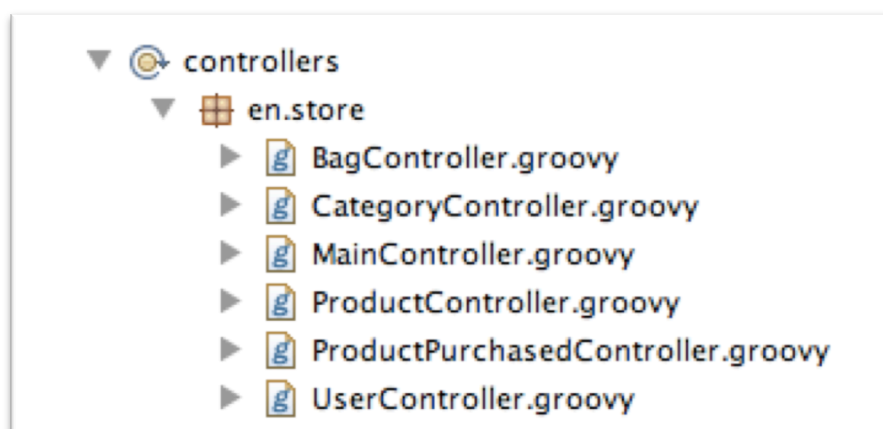


Figure 2 - List of domain controllers

List of views:

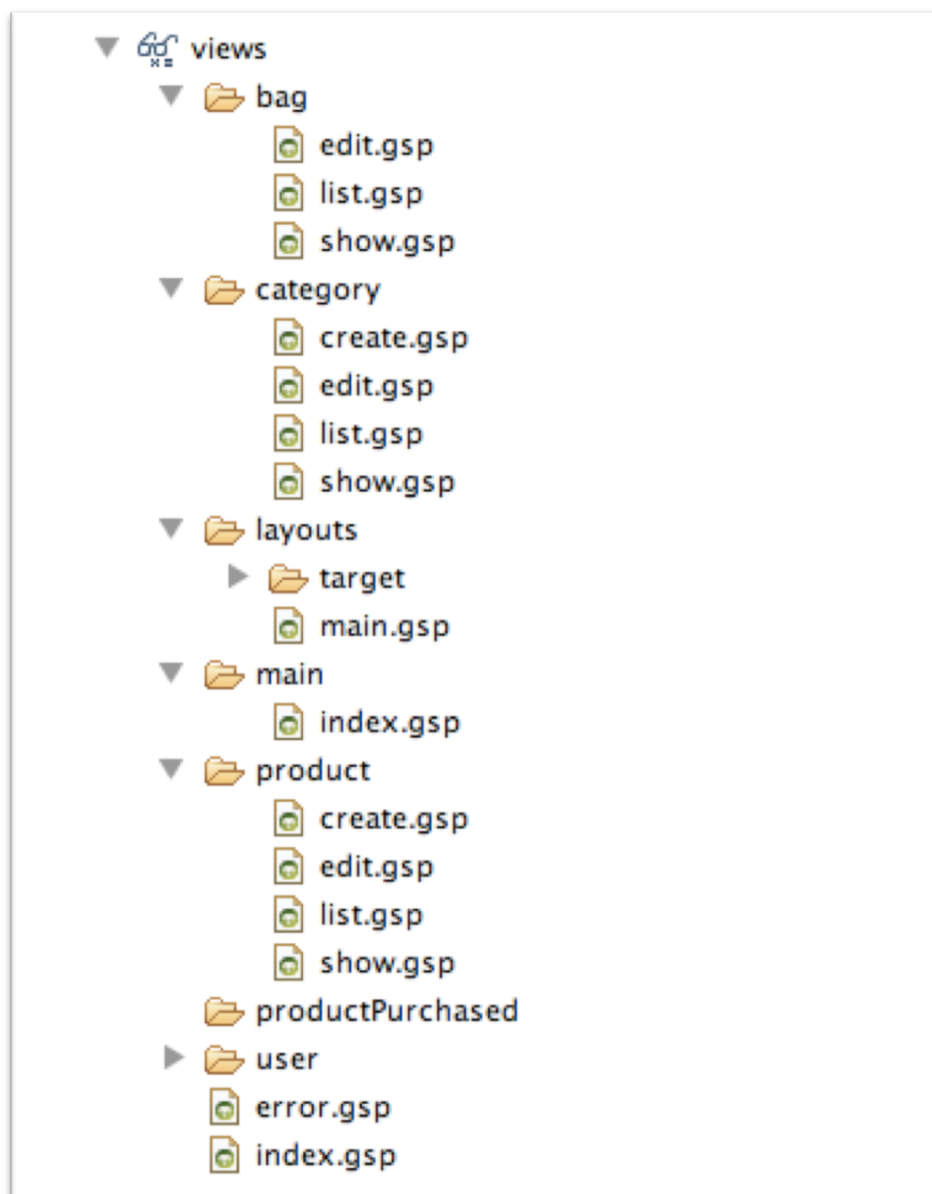


Figure 3 - List of the views

2. Screen shots.

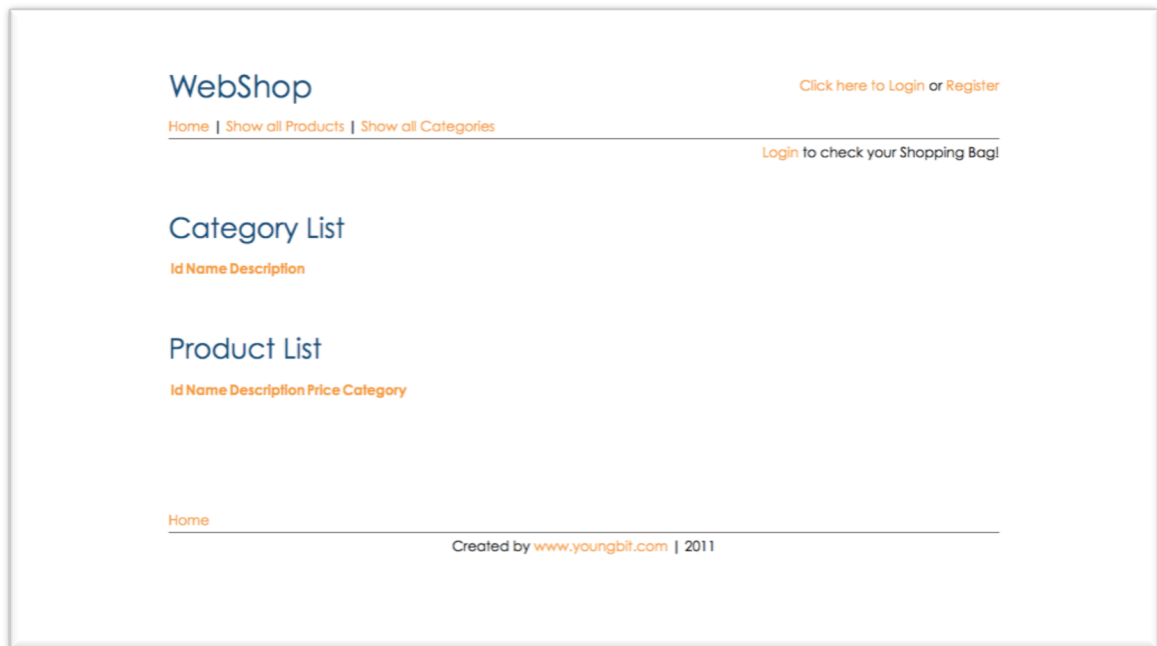


Figure 4 - Controller: Main

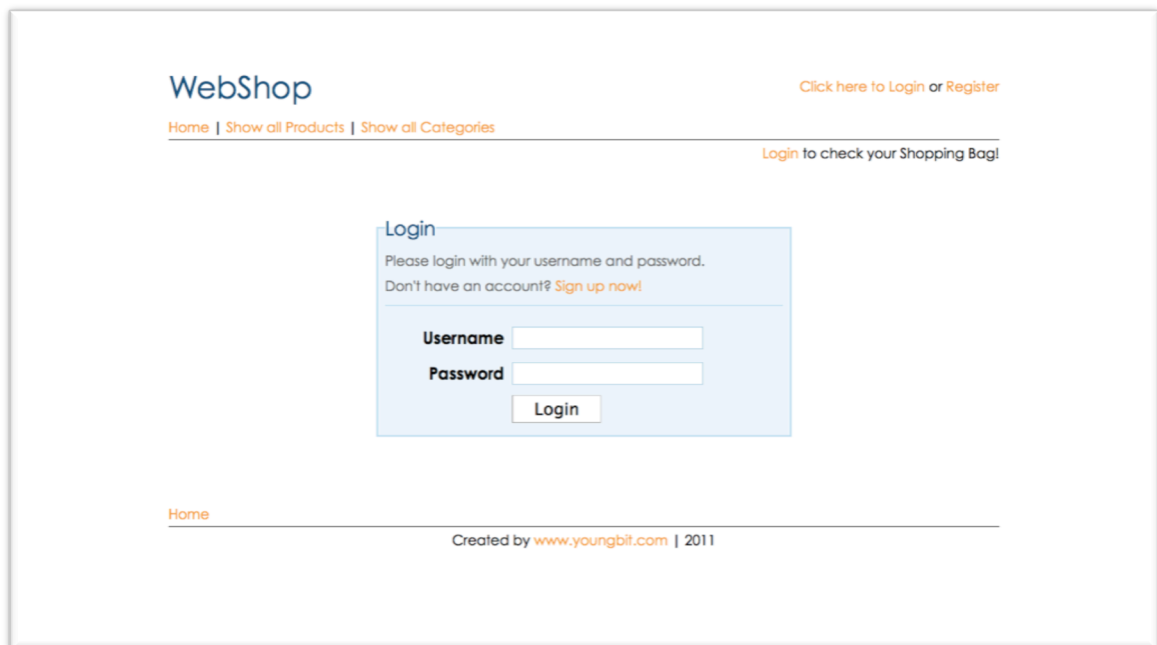


Figure 5 - Controller: User - Action: Login

The screenshot shows the 'WebShop' user registration page. At the top left is the 'WebShop' logo. To the right, there is a link 'Click here to Login or Register'. Below the logo are links for 'Home', 'Show all Products', and 'Show all Categories'. On the right side, there is a link 'Login to check your Shopping Bag!'. The main content area is titled 'User Registration' and contains the instruction 'Complete the form below to create an account!'. The form has five input fields: 'Username', 'Password', 'Confirm Password', 'First Name', and 'Last Name'. A 'Create Account' button is located at the bottom right of the form. A 'Home' link is visible at the bottom left of the page.

Figure 6 - Controller: User - Action: Register

This screenshot shows the same 'WebShop' user registration page as Figure 6, but with the form fields filled out. The 'Username' field contains the text 'admin'. The 'Password' and 'Confirm Password' fields are filled with five dots each. The 'First Name' field contains 'Manuel' and the 'Last Name' field contains 'Escrig'. The 'Create Account' button remains at the bottom right. At the bottom of the page, there is a 'Home' link on the left and a footer that reads 'Created by www.younabit.com | 2011'.

Figure 7 - Controller: User - Action: Register

WebShop Manuel Escrig | [Logout](#)

[Home](#) | [Show all Products](#) | [Show all Categories](#) | [Create Product](#) | [Create Category](#) [My Shopping Bag](#)

Create Category

Complete the form below to create a new Category!

Name	<input type="text" value="Category 1"/>
Description	<input type="text" value="Description Category 1"/>
User	<input type="text" value="admin"/>
<input type="button" value="Create Category"/>	

[Home](#)

Created by www.youngbit.com | 2011

Figure 8 - Controller: Category - Action: Create

WebShop Manuel Escrig | [Logout](#)

[Home](#) | [Show all Products](#) | [Show all Categories](#) | [Create Product](#) | [Create Category](#) [My Shopping Bag](#)

Category List

Id	Name	Description
1	Category 1	Description Category 1
2	Category 2	Description Category 2
3	Category 3	Description Category 3

[Home](#)

Created by www.youngbit.com | 2011

Figure 9 - Controller: Category - Action: List

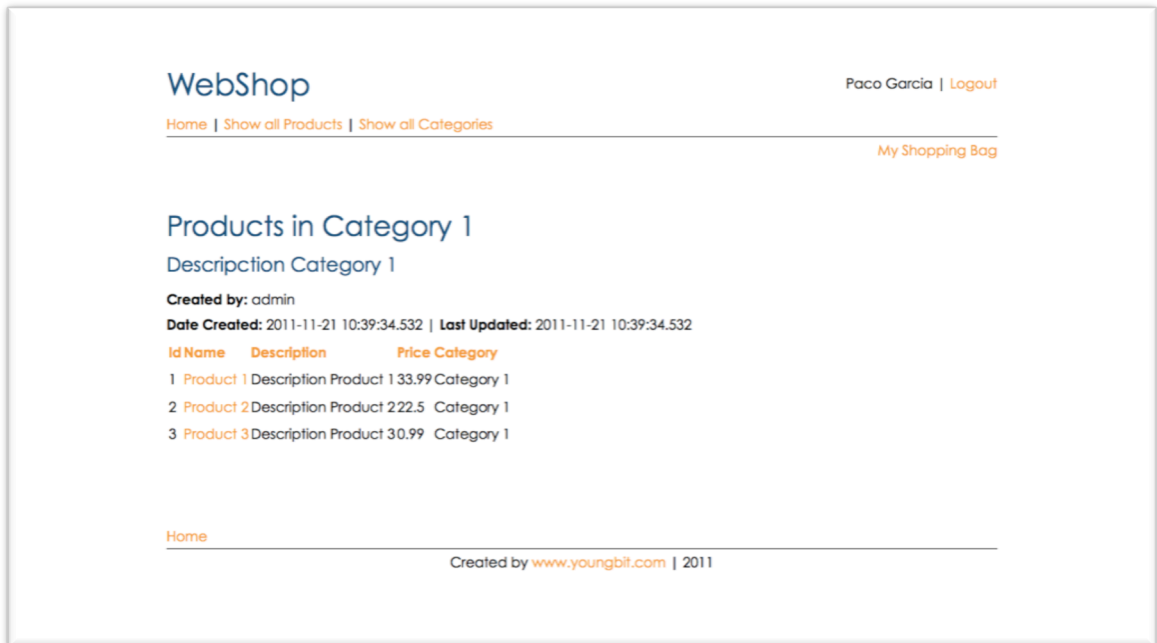


Figure 10 - Category: Controller - Action: Show

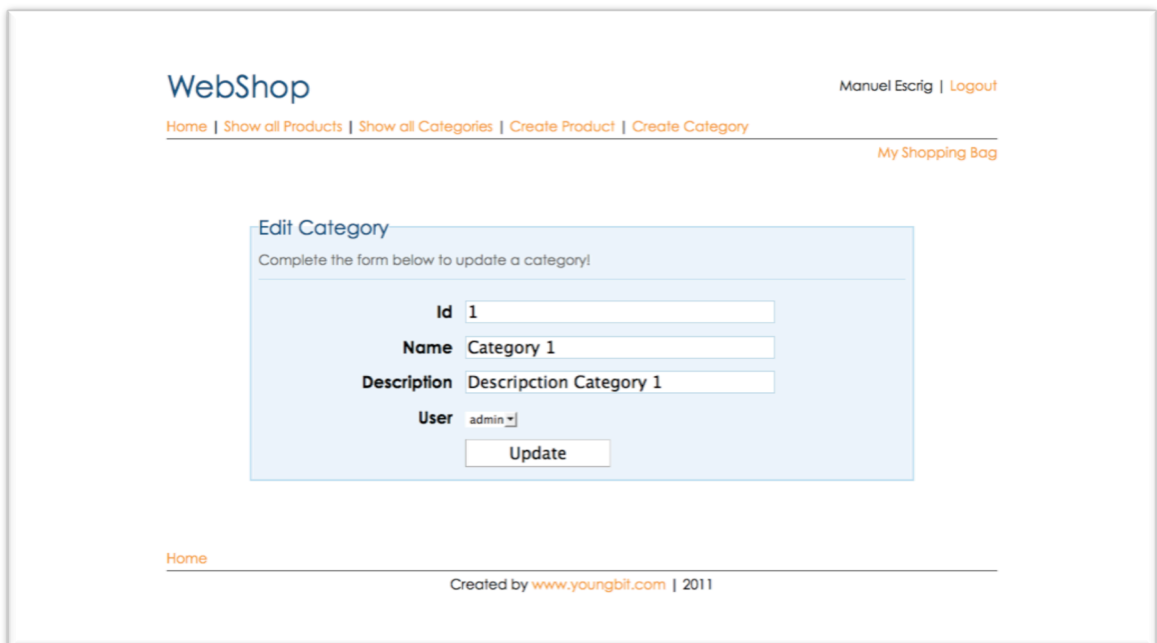


Figure 11 - Category: Controller - Action: Edit

WebShop Manuel Escrig | [Logout](#)

[Home](#) | [Show all Products](#) | [Show all Categories](#) | [Create Product](#) | [Create Category](#)

[My Shopping Bag](#)

Create Product

Complete the form below to create a product!

Name	<input type="text"/>
Description	<input type="text"/>
Price	<input type="text"/>
Stock	<input type="text"/>
Category	Category 1 ▾
User	admin ▾

Figure 12 - Controller: Product - Action: Create

WebShop Manuel Escrig | [Logout](#)

[Home](#) | [Show all Products](#) | [Show all Categories](#) | [Create Product](#) | [Create Category](#)

[My Shopping Bag](#)

Product List

Id	Name	Description	Category	User
1	Product 1	Description Product 1	Category 1	admin
2	Product 2	Description Product 2	Category 1	admin
3	Product 3	Description Product 3	Cat2	admin

[Home](#)

Created by www.youngbit.com | 2011

Figure 13 - Controller: Product - Action: List

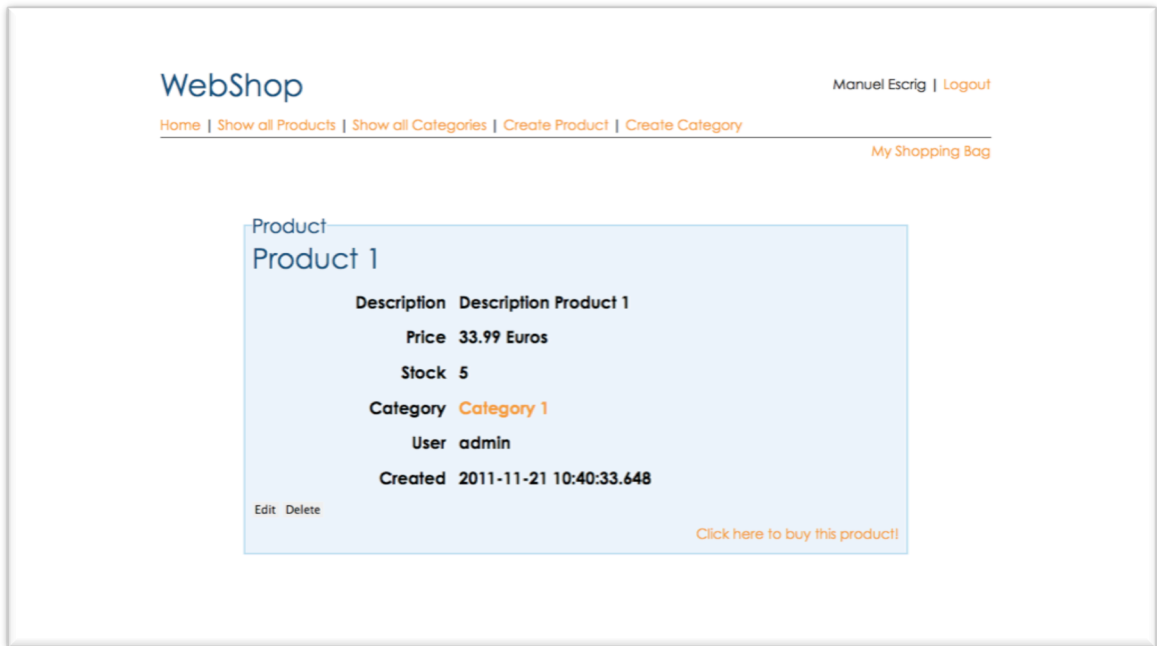


Figure 14 - Controller: Product - Action: Show

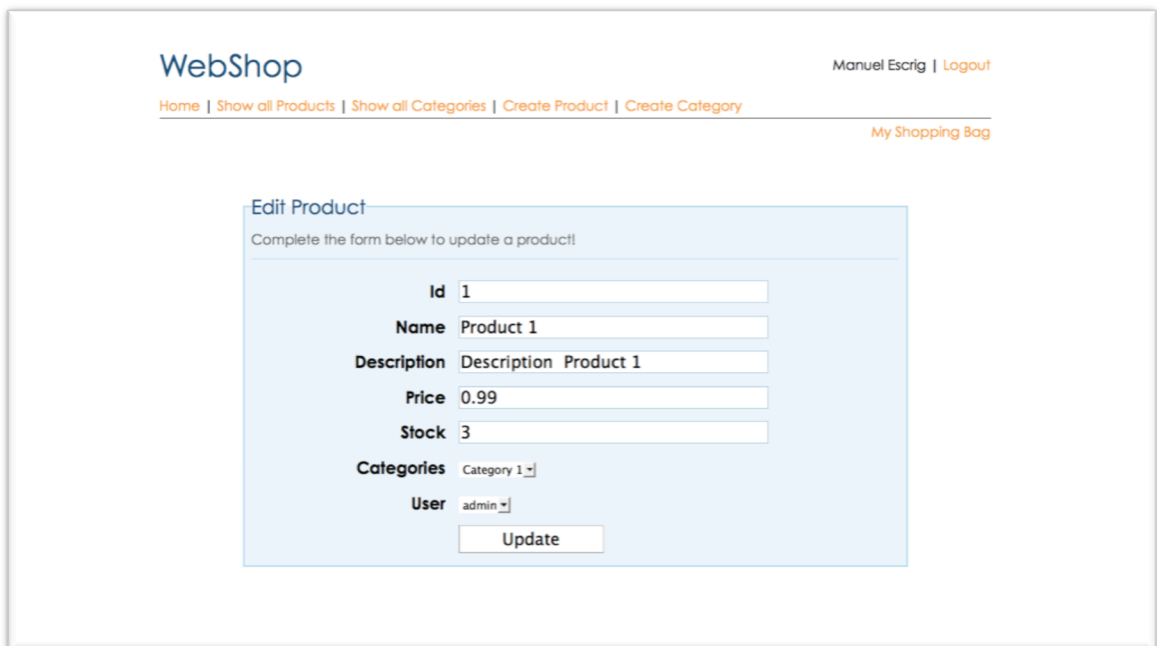


Figure 15 - Controller: Product - Action: Edit

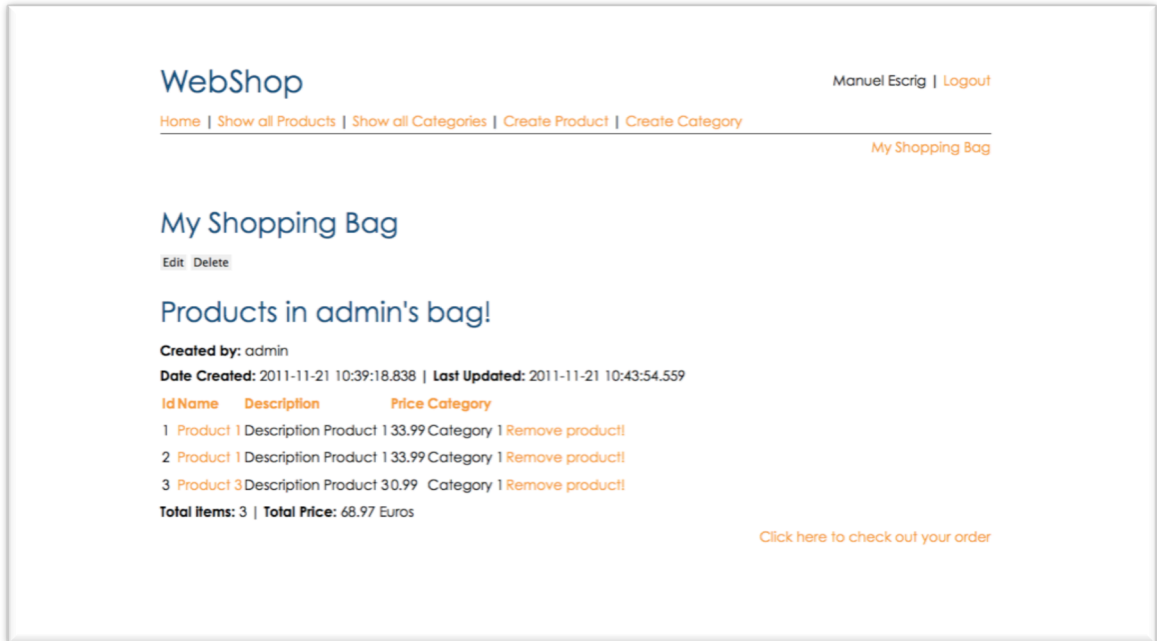


Figure 16 - Controller: Bag - Action: Show

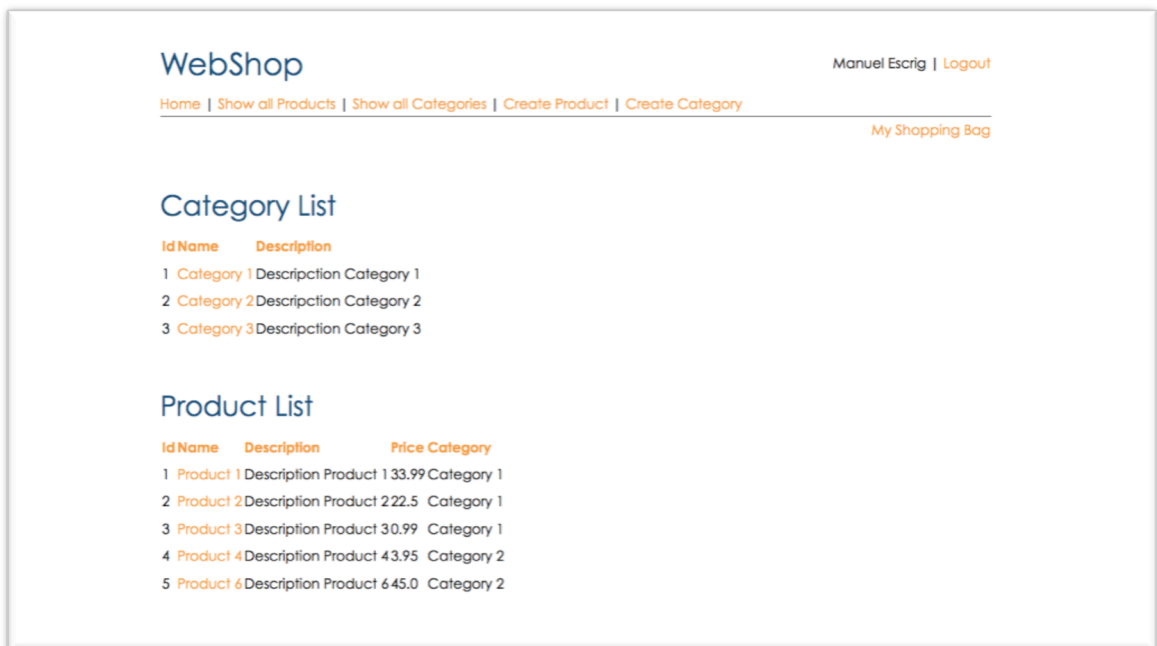


Figure 17 - Controller: Main - Action: Index (admin)

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 and 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 ProductPurchased 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 as 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.