

CPRG 214 – Lab 3

20 points

Complete on individual basis.

In this lab you will create a simple ASP.NET MVC application that retrieves data from Products table in Halloween database. You do not need to provide any DML operations (Insert, Update, or Delete), just retrieve the data and display in a list (do not show LongDescription nor ImageFile in the list). There is a Details button by each products that leads to another view showing all data of the selected product, including long description and image.

Styling of the page is totally up to you; however, the page should be responsive (use Bootstrap).

Detailed Requirements

- Start with ASP.NET Web Application with MVC template. You do NOT need to change the views for Index, About, and Contact Actions of the home controller.
- In the Models folder define two classes: Product.cs and ProductDB.cs. ProductDB should have GetProducts() method that retrieves all products' data and returns a list of products. Alternatively, you may use Entity Framework to generate class Product from the database and retrieve products' data. (In that case ProductDB class is not needed.)
- All Bootstrap and CSS styles are located in the Content folder.
- Modify HomeController by adding Products action. Retrieve data from Products table using method from ProductDB class and pass the list of products to the view. The view needs to be created with Template: List and Model class: Product. LongDescription and ImageFile should not be included in the list display. Remove code for automatically generated buttons for Add New, Edit and Delete (they will not be needed), but leave the Details button.
- When user clicks on the Details button by a product, another page comes up that shows the all data of the selected product, including image of the products with alt text being the file name. To create a view for that page, use Template: Details and Model class: Product.

Marking Scheme

Marking Criterion	Out Of
Application structure complies with ASP.NET MVC template and contains folders Controllers, Models, and Views. All views that belong to HomeController are located in the Home sub-folder of Views folder. Classes Product.cs and ProductDB (if present) reside in Models folder.	5
Application retrieves data of all products and displays them in a list form as required	7
Details page shows all data of the selected product, including the image	5
Application is responsive and reasonably styled (does not have to be fancy). Any Bootstrap and CSS files are placed in Content folder.	
Total:	20

Submit

Zip the project folder and submit to the appropriate Dropbox folder.