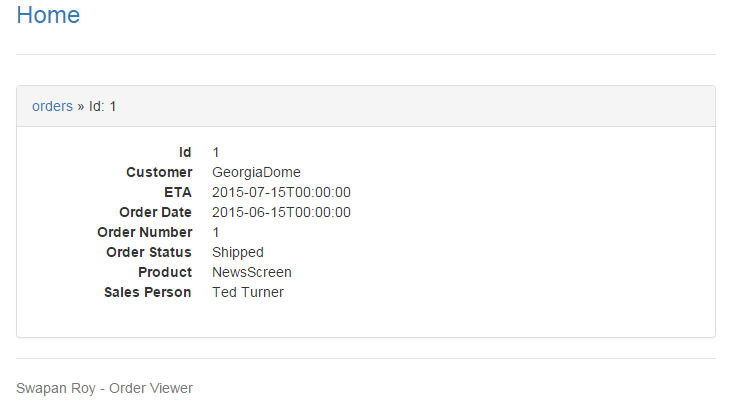
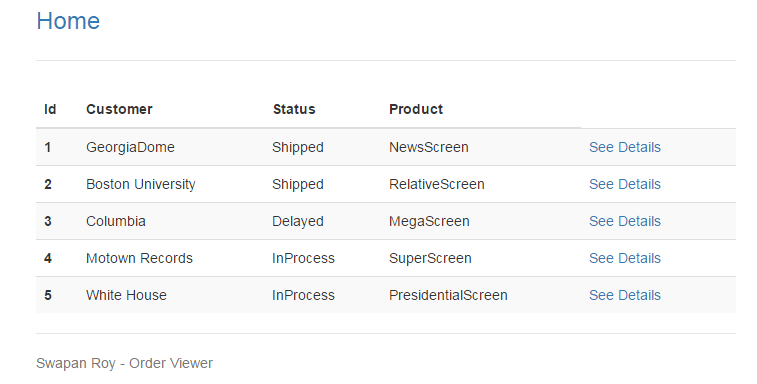
# Order Reviewer

This package contains a simple Web API application implementation example showing two views - Orders and Order Details.

The Orders view displays a list of orders. Each item in the list contains a button, which when clicked displays the Order Details view for the item.



The Order Details page provides a breadcrumb on the top of the list, which can be used to switch back to the List item.

## About the design

1. This was developed on Visual Studio 2012.
2. Three exernal packages are used – Nunit, Moq, and NewtonSoft.json.
3. The project is a MVC 4 project with Web API template. A test project was also added.
4. The MVC controller OrdersContoller was added to contain all the action methods for the REST verbs related to Order.
5. In the Model area, the Order.cs contains the data model for Order, and the OrderRepository .cs that provides the data access services to the OrdersController through the IOrderRepository interface.
6. The OrdersController class contains two constructors, one to use the actual IOrderRepository interface for normal use, and the other to use the mock IOrder Repository interface for testing.
7. Since this is a simple example application, no sql database was used. The data is preloaded in json format from the App\_data /Order.json file. Read and write private functions are provided in the OrderRepository.cs to get and save the data from the file.
8. The REST verbs Get and Get/id only are used at present for this application. But the code can be easily extended to support the other verbs as well. Both returns json data to the client containing the list of orders and the order details for specific order ID.
9. The OrderReviewTest project contains a OrdersControllerTest.cs file. It contains a Moq test mehod to test the Get/Id verb implementation of the controller. This is provided just as an example and can be extended to test other functions in the application as well.
10. The client side code included in this project should be seen as something for exercising the server side code that is being submitted for the review. It therefore uses some existing pieces of code that has been modified for the purpose.   
    As a quick overview, it consists of the index.html containing the html templates used to render the two views, and script/main.js containing the application code. The main.js is supported by the script/vendor.js, and plugin.js. The vendor.js contains jQuery and Director library, and the plugin.js contains a custom version of the Bootstrap library. The CSS is in the styles/main.css.  
      
    Note – when testing using the Visual Studio debugger, it may be necessary to change the localhost port number at the top of the main.js.

Swapan Roy

8/19/20015