**Introduction**

This is a sample ecommerce web application whose primary objective is to sell templates for websites.

**Technology Stack**

Front End/UI: AngularJS

Back End/RESTful Service: ASP.NET Web-Api

Database: Microsoft SQL Server

ORM: Entity Framework 5.0

IDE: Microsoft Visual Studio 2015 Professional, SQL Server 2014 Management Studio

Libraries: AngularJS, Angular ui-router, Bootstrap

**Project Architecture:** The following projects are present in the solution:

1. ASP.NET Web Application Project (AngularJS UI):

The UI project has 4 main modules, namely: Login module, Contact Us module, About Us module and Selling module. All of these modules, except for Login have separate angular modules, controllers, services and routing config files. All these files have been referenced in the index.html and all the modules have been injected in the main angular module (in app.js).

Login module uses the main angular controller and service.

1. 2. ASP.NET Web-Api Project (RESTful Services):

The back-end Web-Api project has three controllers, namely: Contact controller, Login controller and Payment controller. Entity framework is the ORM for this project and the same uses ContactDataModel as the data model. There are four entity classes, namely: User.cs, Payment.cs, Product.cs and Contact.cs.

**Features/Content:** This web application has the following features:

1. **Authentication**: Only authorized users can browse the contents of this web application after the successful login.

NOTE: 1. No Sign-Up option is provided as it is beyond the scope of this project.

2. If the user tries to browse any pages without being logged in, an alert message is shown and the user is routed back to the login state.

2. **Contact Us:** All the authorized users can post their queries/messages to the developers/admin through the Contact US page.

3. **About Us:** All the authorized users can find the information on the team of the template selling website.

4. **Online Store:** All the authorized users can browse through the collection of products and buy the template of their choice by making payment online.

NOTE: There is no actual payment gateway installed. The card information of the user is being posted to the database upon clicking the “Make Payment” CTA.

**Preconditions, Assumptions and Recommendations:**

* This web application works only if hosted on a web server. The solution contains two projects, UI and the Backend respectively and both of them have to be hosted on the Local IIS along with a proper database. DB scripts are provided with the solution which also includes the sample values.
* Products/Templates information is currently being fetched from a local json file instead of database.
* It is recommended to use Visual Studio 2015 and SQL Server Management Studio 2014 for modifying the code.
* Connection strings may need modification in the Web.config file of back-end Web-Api project.
* The backend Web-Api project is currently hosted in the Local IIS with the following URL: <http://localhost/REST-WebApi> and the same is defined as a constant in the app.constants.js file in the AngularJS UI project. These constants must be accordingly modified if the backend Web-Api project is hosted with a different URL.
* Since, the back-end Web-Api project is hosted in the Local IIS, the Visual Studio must be opened in the administrator mode. Otherwise, Web-Api project fails to load and “**This project requires users input. Reload the project for more information**” message would be displayed in the Visual Studio.
* If you get “XMLHttpRequest cannot load <Your API URL>. Response for preflight has invalid HTTP status code 405”, then add the following code in Global.asax.cs in your WebApi layer.

protected void Application\_BeginRequest()

{

if (Request.Headers.AllKeys.Contains("Origin") && Request.HttpMethod == "OPTIONS")

{

Response.Flush();

}

}

**License:** This project is distributed under the MIT License. A copy of the same can be obtained at**:** [**https://opensource.org/licenses/MIT**](https://opensource.org/licenses/MIT)

**Developer Information:**

**Name:** Sushruta Gurumurthy Nadiger

**E-Mail:** [sushrutagn@gmail.com](mailto:sushrutagn@gmail.com)

**Git-Hub:** <https://github.com/sushrutagn>