**Shopping Cart System**

**POC**  
**Low Level Design (LLD)**

Date: 25/05/2022

Current Document Version: [*1.1*]

DOCUMENT APPROVAL

**Approvers of this document**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Department** | **Role** | **Signature** | **Date** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**Document Change History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Document Version #** | **Author** | **Date** | **Description** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

[1.0 Document Purpose 4](#_Toc94636300)

***1.1* *Scope*…………………………………………………………………………………………………………….…4**

[2.0 Intended Audience 4](#_Toc94636301)

[3.0 Project Background, Objective(s) 4](#_Toc94636302)

[3.1 Project Background 4](#_Toc94636303)

[3.2 Project Objective 4](#_Toc94636304)

[4.0 Design Pattern 4](#_Toc94636305)

[5.0 Solution Diagram](#_Toc94636306) 6

[5.1 Use Case Diagram 7](#_Toc94636303)

[6.0 Solution Steps 9](#_Toc94636307)

[7.0 Classes/function name 9](#_Toc94636308)

[8.0 Validations 10](#_Toc94636309)

[9.0 Data model/Tables 11](#_Toc94636310)

[10.0 HTTP Status Code 13](#_Toc94636314)

[11.0 Unit Testing 13](#_Toc94636314)

# Document Purpose

This document describes the solution architecture for Shopping Cart System

**1.1 Scope**

**In this application, we will create a fully working Online Shopping Web application with the Dotnet core Web API backend and Angular frontend. The entities that we will be working with users and merchants, and these entities will be having their access limits in the WebApp. An Admin will be having all the access to the WebApp.**

# Intended Audience

This document is intended as a reference for the following roles and stakeholders who are interested in the Online Shopping Cart technical architecture.

# Project Background, Objective(s)

## Project Background

A shopping cart system is a virtual store on the Internet where customers can login, browse the products and select products of interest. The selected items will be moved to cart. At checkout time, the items in the cart will be presented as an order. To complete the transaction the customer will be asked to fill a shipping address, and payment information

## Project Objective

The purpose of the project is to build a web application program to reduce the manual work for managing the Cart, Orders, Products, Payment. It tracks all the details about the Payment, Customer, Products.

# Design Pattern

|  |  |  |
| --- | --- | --- |
| # | Name | Description |
| 1 | API | Using HTTP requests, we will use the respective action to trigger various operations |

**Hardware and Software Requirements**

**Operating Systems**

Windows 10

**Hardware Environment**

Processor: x86 or x64

RAM: 512 MB (minimum), 1 GB (recommended)

Hard Disc: up to 3 GB of free space may be required

**Development Environment**

Microsoft Visual Studio 2019, 2022.

Visual Studio Code (Text-editor)

.NET Core 3.1 and above

Command Line (Optional. Necessary for command line deployment)

Internet Information Services (IIS) 7.0+

Microsoft SQL Server Management studio 2018, 2019

**Browser support**

Chrome Firefox Opera Edge IE

Latest Latest Latest 13 + 11 +

**Specification and other Technologies**

Angular as front end and ASP.NET Core Web API as Backend.

HTML 5

CSS 3

JavaScript

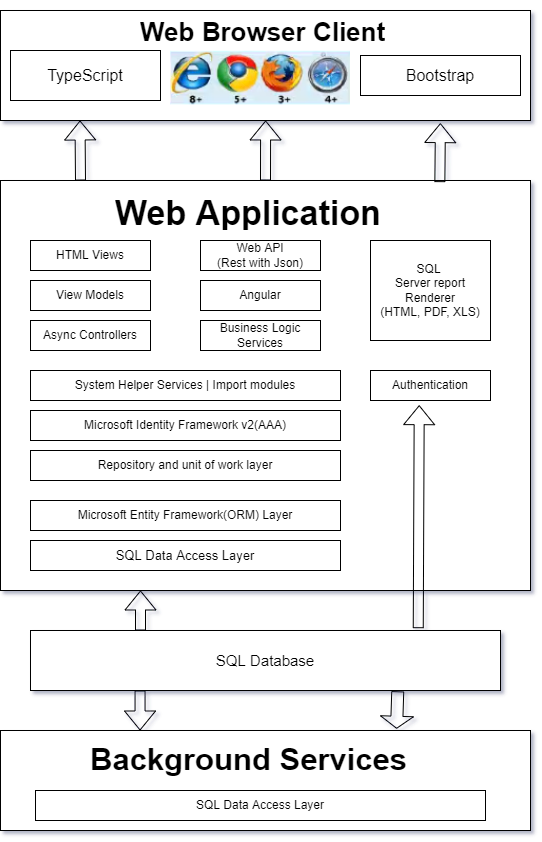
Typescript

Bootstrap

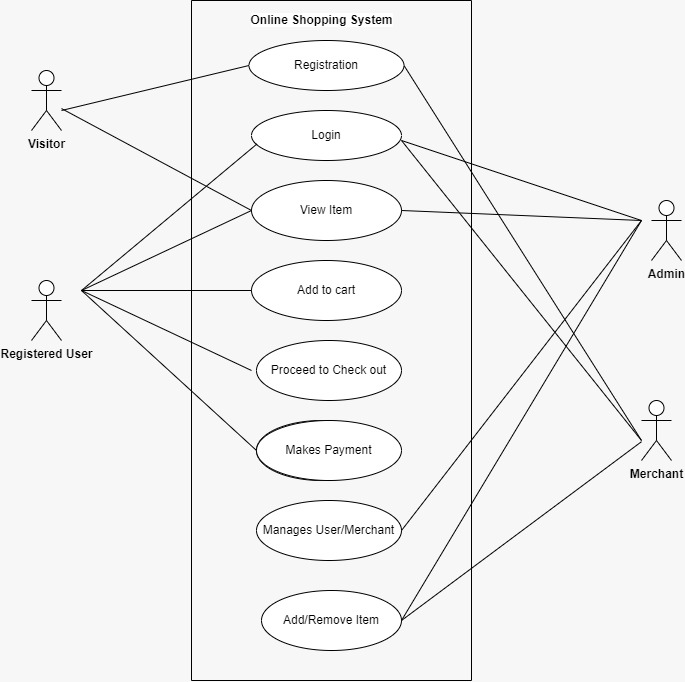
jQuery

MySQL

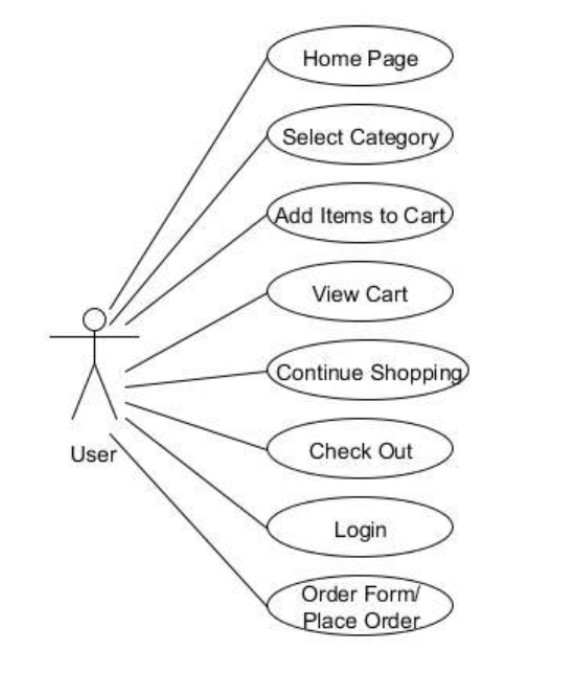
# Solution Diagram



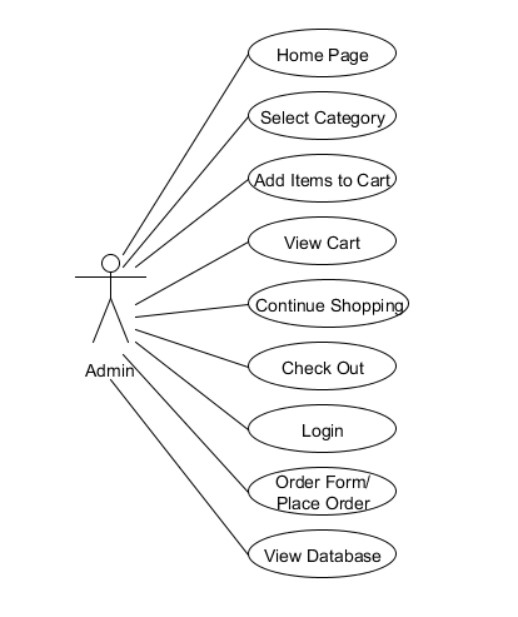
# Use Case Diagram

****

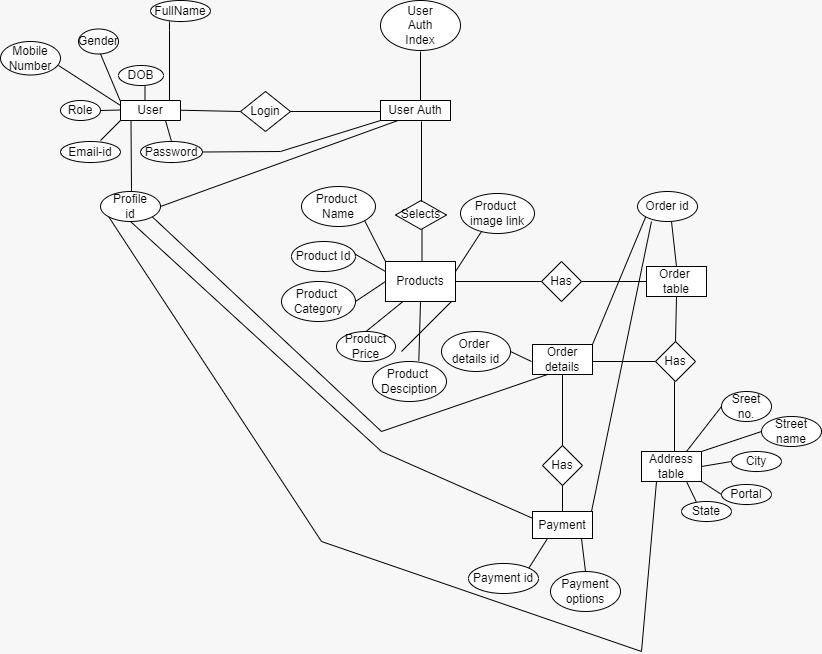
**User**



**Admin**

****

**ER Diagram**



# Solution Steps

**User Registration:**

This is for the Customer, Merchant and Admin to login/Sign Up for the WebApp

* For login, we are going to use an email id and the password.
* For Sign Up, we are going to use the details like First name, Last name, email etc. (Full details in the table section)

**Product Details:**

This is about the product details which are available in WebApp. Here some fields include like productid, Product name, Price etc. (Full details in the table section)

* This product can be seen by every user without logging in or register.
* Only Merchant can add and delete products.
* Admin can delete any product inside the WebApp. (If any inappropriate products are there)

**Cart Details:**

In this, the user selects the products and adds them to the cart which has its own id.

* The cart will display the total price of products those were added by the user.
* User will be able to remove item from the cart

**Payment Details:**

In this section, the user needs to add the proper address with the fields like flat number, street name, postal code etc. (Full details in the table section) and the payment method should be added.

* Once the address is added, this user details can be used for the future references.

# Classes/function

|  |  |  |
| --- | --- | --- |
| **#** | **Class** | **Description** |
| 1 | User.cs | Model holds the Users schema details |
| 2 | AuthenticationController.cs | Has the functionality of Register, Login, Logout it uses Dto and user to do the functionality |
| 3 | Product.cs | It Contain details about the product. |
| 4 | OrderTable.cs, OrderDetailsTable.cs | It has the functionality for the cart details and the full order detail are there is the “Order Detail Table” |
| 5 | Payment.cs | Has the Detail for the various payment method that is available |
| 6 | AddressTable.cs | It Contain detail about the address and it is connected with the user table by using the foreign key “profileID” |
| 7 | ProductController.cs | Has all the API operation for Product Table |
| 8 | OrderTableController.cs,  OrderDetailsTableController.cs | Has the API Detail for the corresponding |
| 9 | AddressController.cs | Has the Detail API for the use of Address table |
| 10 | PaymentController.cs | Has the API Details for the use of Payment table |

# 8.0 Validations

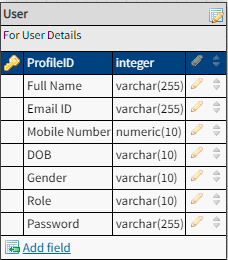
**For Registration:**

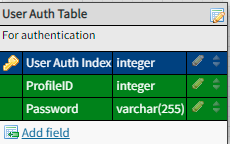
User have to do registration with valid information like Name, Email Address, Phone Number, Select User Type, Password, Confirm-password are inserted to register this System.

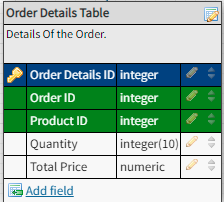
**For Login:**

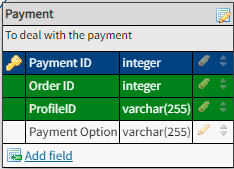
Admin, Customer and Merchant can login in this system by using Email Address and password.

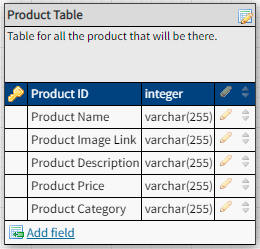
# Data model/Table

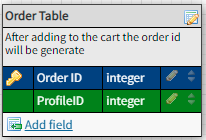


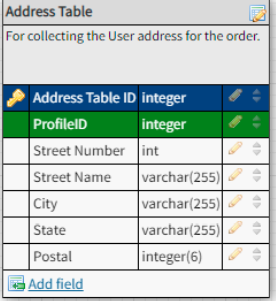












# 10.0 HTTP Status Code

201 – Customer Registered

200 - Request succeeded

400 – Inputs are invalid

404 – Customer Not found

502 – Bad gateway

# 11.0 Unit Testing

|  |  |
| --- | --- |
| Project Name | Online Shopping Cart |
| Created by |  |
| Date of Creation |  |
| Date of review |  |
|  |  |

**For Registration of customers**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test CASE ID | TEST CASE SCENARIO | TEST CASE | PRE CONDITION | TEST STEPS | TEST DATA | EXPECTED RESULT | Actual Result |
| TC\_o1 | Registration | Enter the valid data to get registered | User needs to enter all the valid details | 1) Enter  FullName: John  Email Address: customer@gmail.com  Mobile Number:0123456789  DOB:22  Select Gender  Password  Confirm Password  Profile Role  2) Enter Submit | <Valid Details> | Successful registration | Successful registration |
| TC\_o2 | Registration | Enter all the required fields to get registered | If customer misses one of the fields during registration which is marked as required in schema | 1) Enter  FullName: John  Email Address: customer@gmail.com  Mobile Number:0123456789  gender  Password  Confirm Password  Profile Role  2) Enter Submit | <DOB is missing> | You need to select dob | You need to select dob |