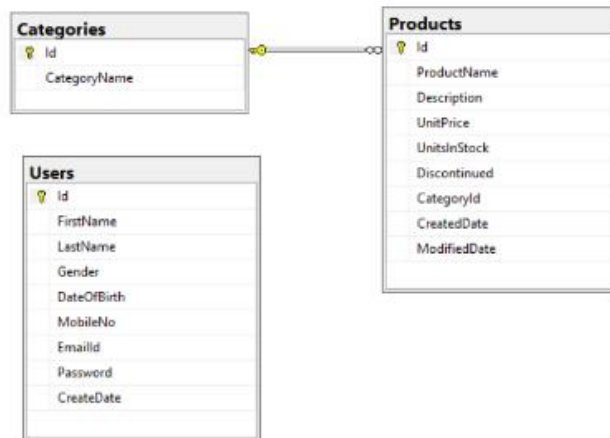


Grocery App

Technologies to be used

- ADO.NET
- WebAPI
- Angular

Database Design



a) Table Name: Users

This table contains the details of all the users

Field Name	Data Type	Description
Id	Int	Identity should start with 1 and increment by 1, Primary Key
FirstName	Varchar (50)	User FirstName, Mandatory
LastName	Varchar (50)	User LastName, Mandatory
Gender	Char (6)	Male or Female or Others Mandatory
DateOfBirth	Datetime	Should be greater than 18 years, Mandatory
MobileNo	Char (10)	Unique, Mandatory
EmailId	Varchar (150)	Unique, Mandatory
CreateDate	Datetime	Default today's Datetime, Mandatory

b) Table Name: Categories

This table contains the details of the category

Field Name	Data Type	Description
Id	Int	Identity should start with 101 and increment by 1, Primary Key
CategoryName	Varchar (100)	Name of the category, mandatory

c) Table Name: Products

This table contains the details of the products

Field Name	Data Type	Description
Id	Int	Identity, Primary Key
ProductName	Varchar (100)	Mandatory
Description	Varchar(250)	Product Description
UnitPrice	Money	Should be greater than or equal to 1, mandatory
UnitsInStock	Int	Should be greater than or equal to 1, mandatory
Discontinued	Bit	Whether the product is available or not, default should be 1, mandatory
CategoryId	Int	Should be an existing ID in 'Categories' table
CreatedDate	Datetime	Default today's Datetime, Mandatory
ModifiedDate	Datetime	When a product is updated, populate this field

ADO.NET (Data Access Layer)

DAL

- RegisterUser()
- LoginUser()
- AddProduct()
- AddCategory()
- EditProduct(int id)
- GetProduct(int id)
- GetAllProducts()
- SearchProduct(int categoryId)
- SearchProduct(string productName, [decimal price])

Note: As per the requirements, you can add more methods

ASP.NET Web API 2

The reference to the Data Access Layer (DLL) should be added to ASP.NET Web API 2.

Add the necessary Properties and Classes in the Web API

ANGULAR

User stories

1. HomePage

- should have a navigation bar using the navbar component
- should have header using header component
- Display the category list in the grid using a separate component

2. AboutPage

- should have a navigation bar using the navbar component
- static information

3. ContactPage

- should have a navigation bar using the navbar component
- static information

4. ProductsPage

- should have a navigation bar using the navbar component
- should have a subcategory component which will read catId from URL fetch
 - o Subcategories from API and display in the sidebar
- should have productList component which will read catId from URL fetch
 - o Products under specified categories in a grid
- The user can able to click product which will navigate to the productDetail component

5. ProductDetail

- should have a navigation bar using the navbar component
- Should be able to read productId from URL and fetch individual product
 - o from API and display detailed information
- should have to add to cart button

6. RegisterPage

- should have a navigation bar using the navbar component
- Should able to register the user by making a post request
- Validation should be applied
- New users can be registered by providing the following information
 - o firstName (required)
 - o lastName(required)
 - o email (required, unique, valid email address)
 - o mobile (required)
 - o password (required, minlength(6))
- should display a response message after successfull or failed request

6. LoginPage

- should have a navigation bar using the navbar component
- Should able to log in the user by making a post request
- Validation should be applied
- The user can login by providing the following information
 - o email (required, unique, valid email address)
 - o password (required, minlength(6))
- should display a response message after a successful or failed request
- should save tokens in local storage

7. Auth Guard service

- Should protect route using can activate
- should able to redirect to login if try to access protected pages