**SHOPBRIDGE**

* Created one microservice Product which is responsible to perform CRUD operations of Products.
* It is based on .NET 5 version.
* In order to run this, you need to have latest .NET5 version available. In case you don’t have the latest version, you can download from <https://dotnet.microsoft.com/download/visual-studio-sdks>
* Code is committed on: <https://github.com/deepali-95/ShopBridge>. Take the pull of main branch.
* Once opened in Visual Studio 2019, all relevant packages should get restore automatically. In case you face any issues, please run the command: dotnet restore. Link for more details: <https://docs.microsoft.com/en-us/dotnet/core/tools/dotnet-restore>
* Please run the SQL Script available in Database folder. It will create ShopBridge database and three tables: Product.ProductCategory, Product.ProductBrand, Product.ProductMaster.
* ProductBrand and ProductCategory, I have created one row from SSMS for both tables and please query these tables to get the Ids as these are foreign key to productmaster table.
* Once solution is opened in IDE, set ShopBridge.Microservices.Product.Api as startup project.

Text

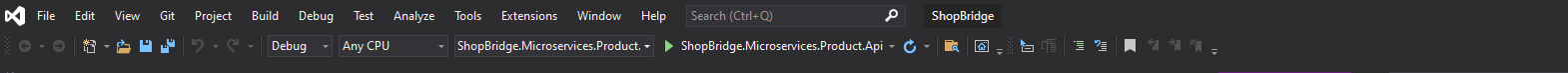
Description automatically generated

* Open the appsettings.json located under ShopBridge.Microservices.Product.Api and modify connectionstrings section with the relevant information.

Graphical user interface, text, application

Description automatically generated

* Select project ShopBridge.Microservices.Product.Api instead of IISExpress to run it. 



* The project will run on port 5001 as https. In case you need to change the port in launchSettings.json which is under Properties folder in ShopBridge.Microservices.Product.Api.
* Once launch successfully you will see swagger ui which list down all apis.  
  Graphical user interface, application

  Description automatically generated
* You can check health of the Product API by hitting: <https://localhost:5001/product/health>
* In case there is any issue or failure in establishment of database connection it will give you the below response:  
  A picture containing graphical user interface

  Description automatically generated
* In case everything is working as expected below will be the output:   
  Graphical user interface, text, application, chat or text message

  Description automatically generated
* You can test the APIs either using Swagger or any other tool like Postman. Below are the details of APIs:
  + To Add product:
    - POST
    - /api/Product
  + To update product:
    - PUT
    - /api/Product
  + To get all products:
    - GET
    - /api/Product
  + To delete specific product
    - DELETE
    - /api/Product/{id}
  + To Get specific product
    - GET
    - /api/Product/{id}
* Please go through Swagger for detailed schemas for Request and Response for above requests.