# General Info

Student Name: Marco Teran Gutmanowitz

Course: ARP400

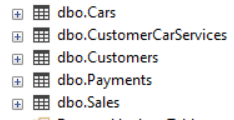
Lab: MAUI

Target Grade: B

# Database

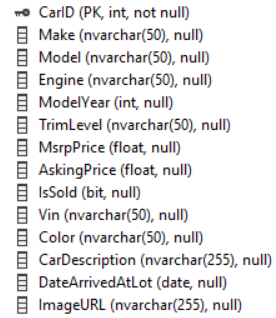


## Tables

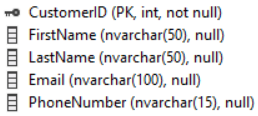
****

**NOTE: Only Cars and Customers table are used for the Lab. You can ignore the other tables.**

**Dbo.Cars**

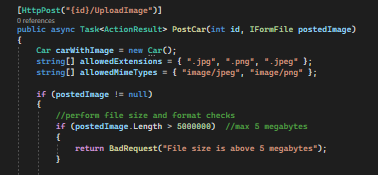
****

**Dbo.Customers**

****

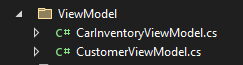
## API

* Made in ASP .NET Core Web API project
* Scaffolded controllers
* Nothing fancy except in CarsController -> PostCar(int carId, IFormFIle postedImage) method. See below image



# MAUI APP

* Following MVVM pattern
* Using local ViewModels instead of Global ViewModel

****

# Screenshots from MAUI app

## For grade D

“For this grade you will create an interface where you show data from one table in your database. You shall have the possibility to perform CRUD operations for the table”

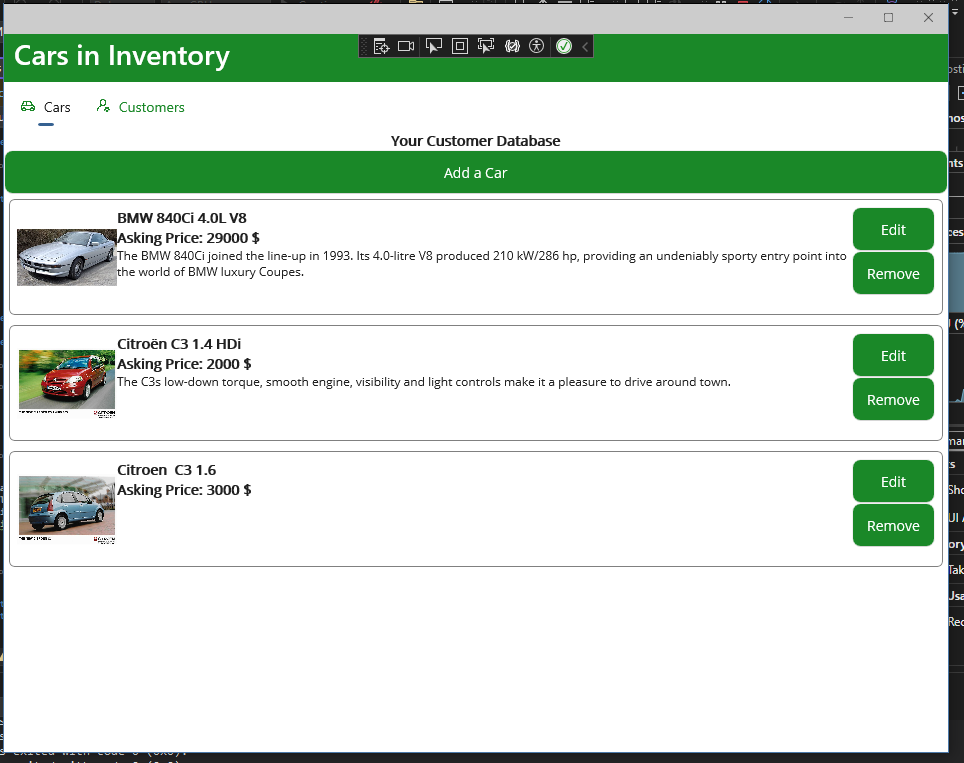


Figure – Read

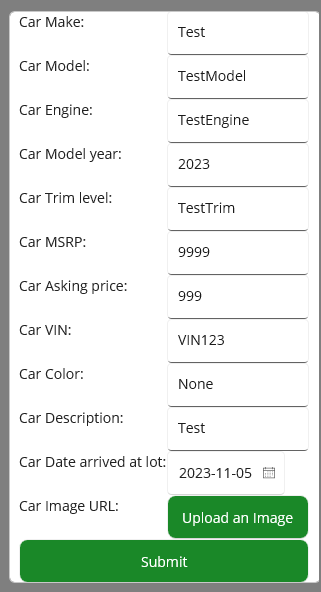


Figure – Create (1)

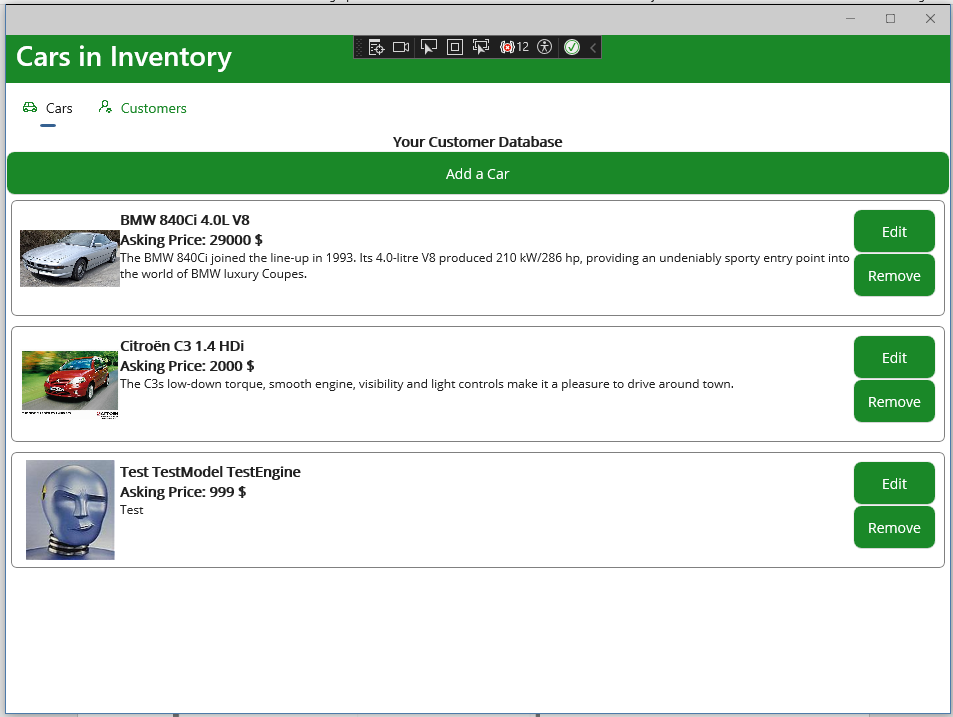


Figure - Create (2)

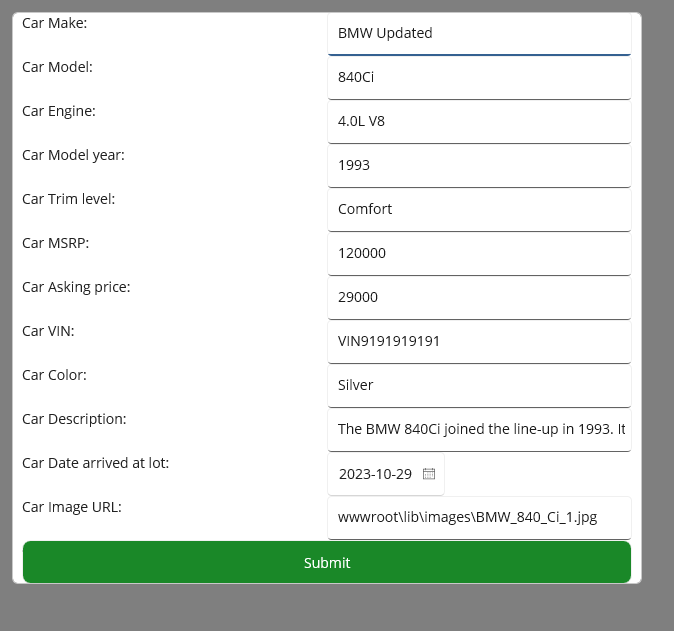


Figure – Update (1)

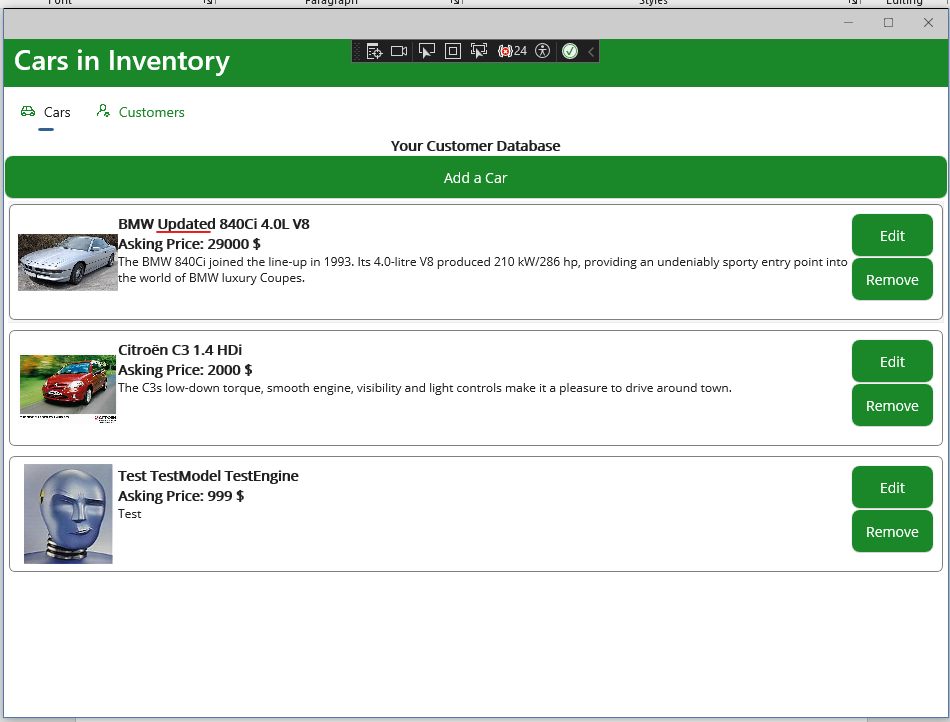


Figure - Update (2)



Figure - Delete (1)

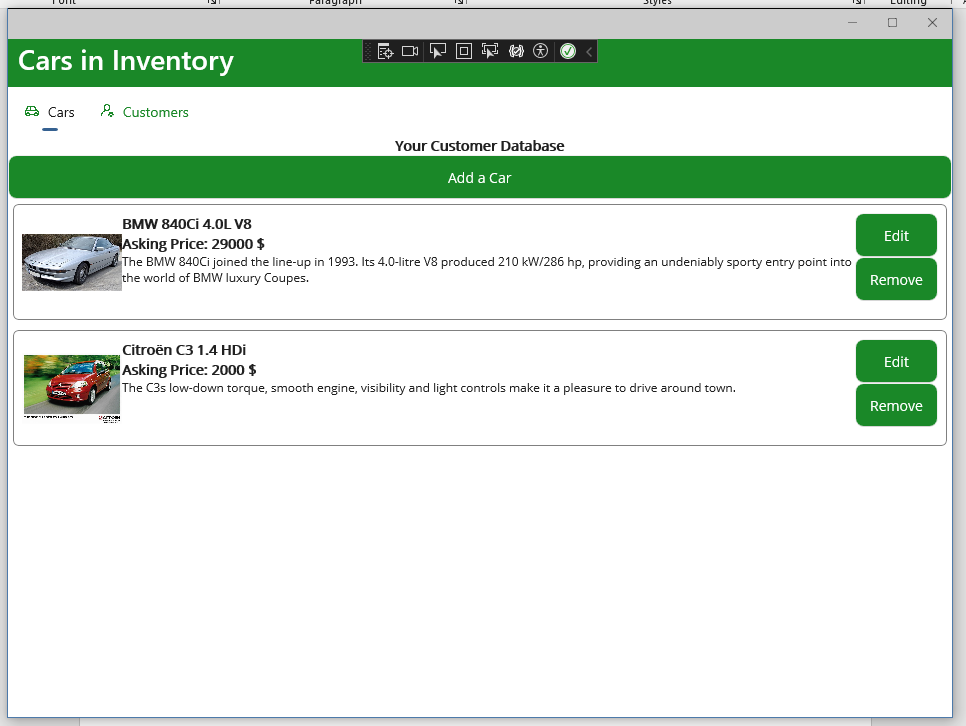


Figure - Delete (2)

## For grade C

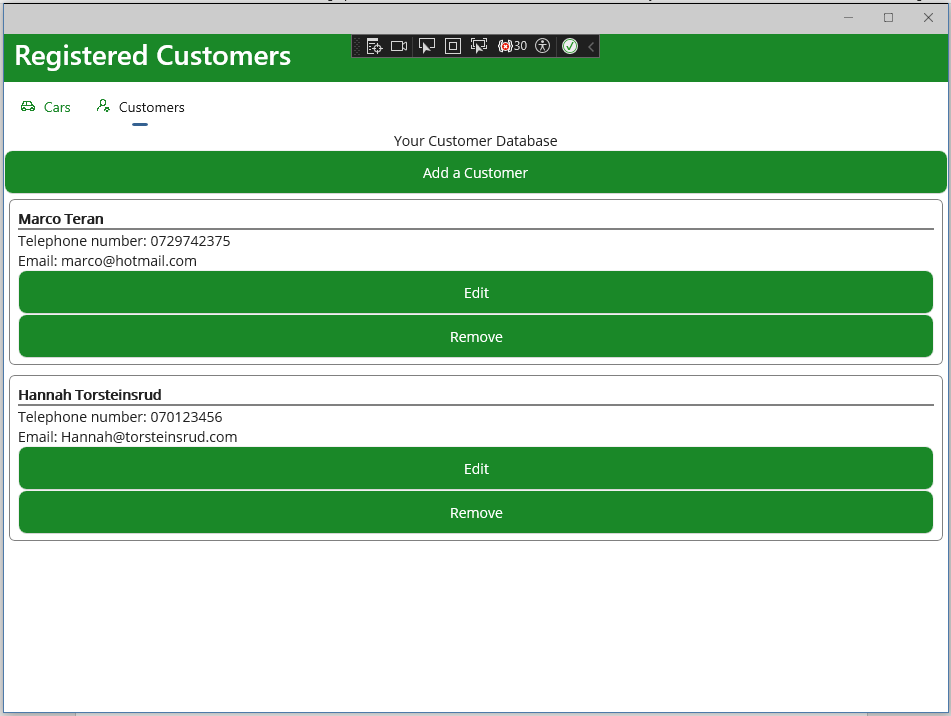


Figure – Read

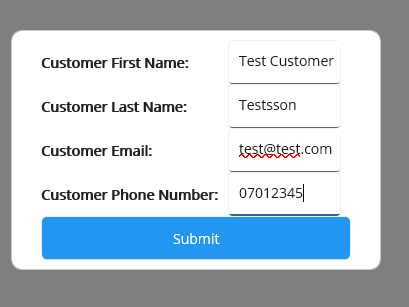


Figure - Create (1)

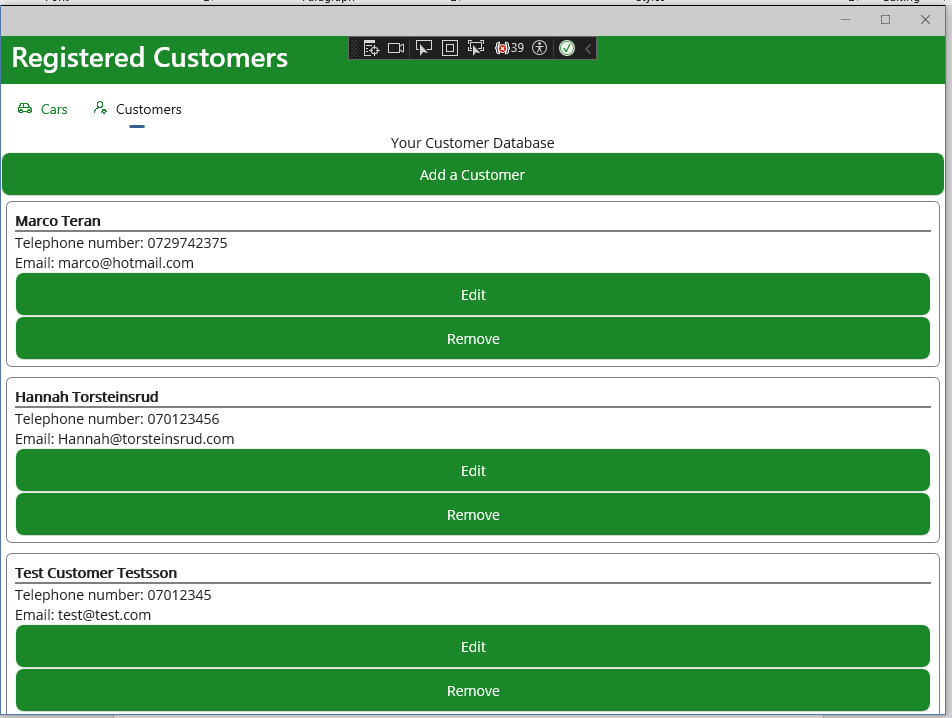


Figure - Create (2)

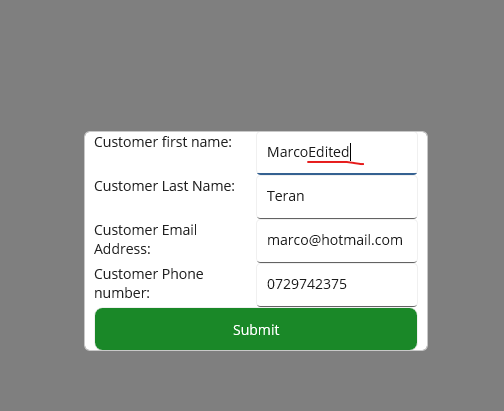


Figure - Update (1)

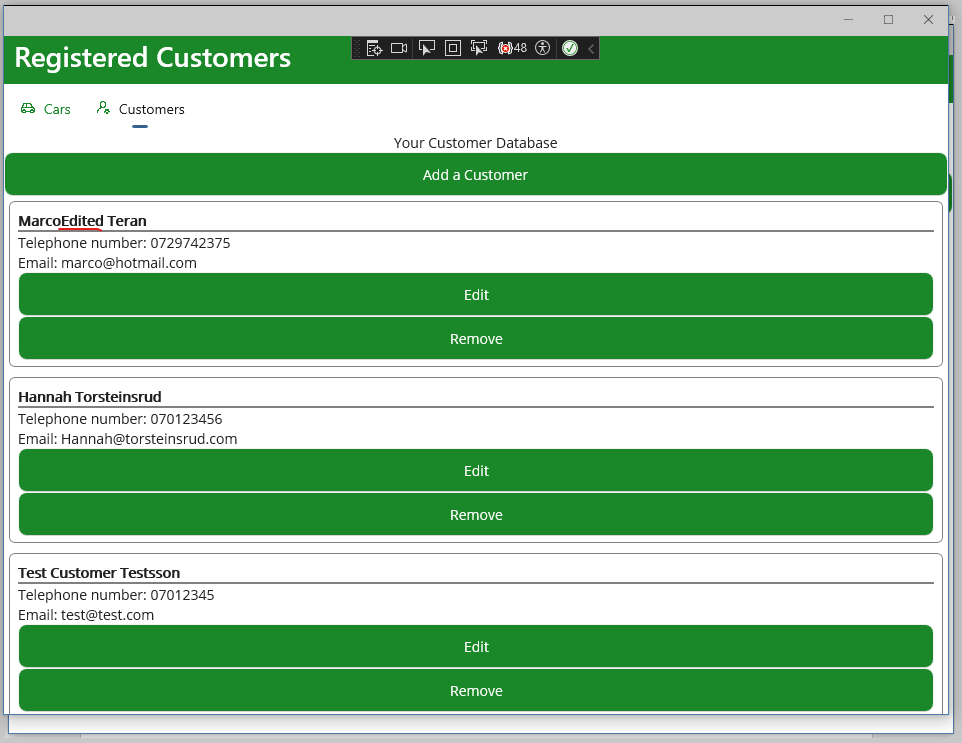


Figure - Update (2)

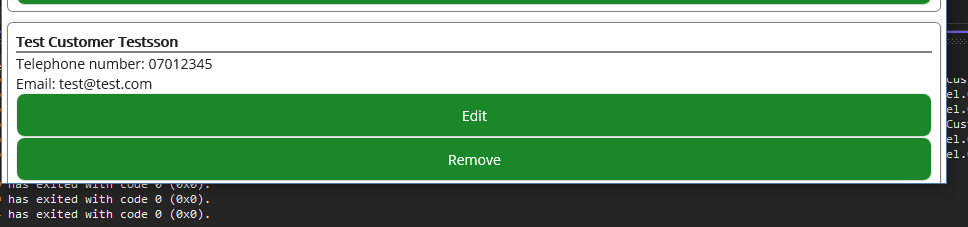


Figure – Remove (1)

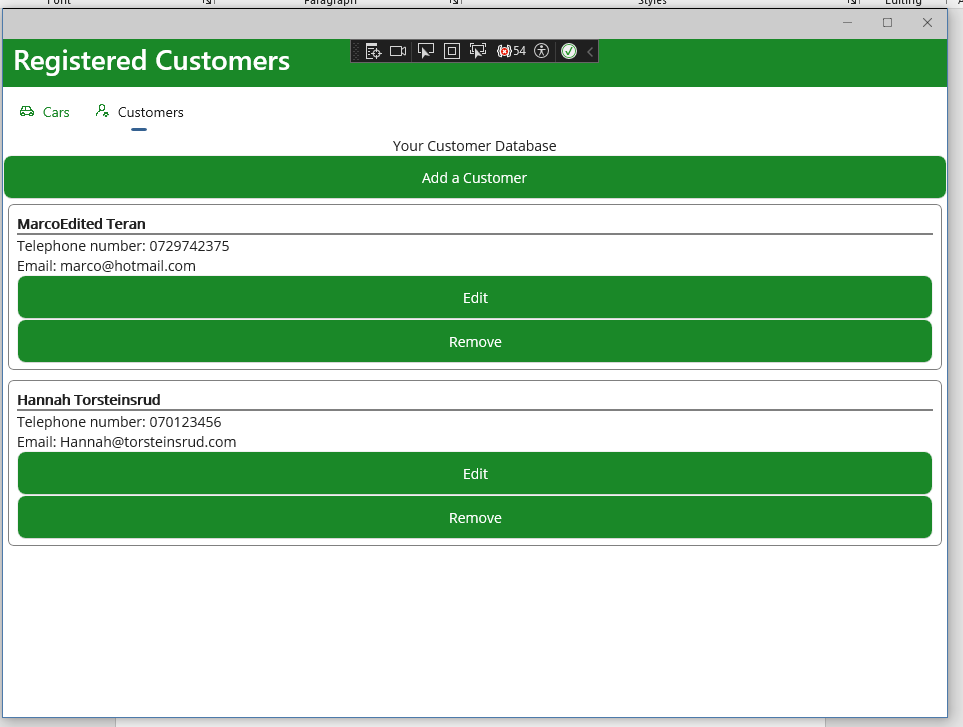


Figure - Remove (2)

**MVVM pattern**

****

Figure - Implementing INotifyPropertyChanged

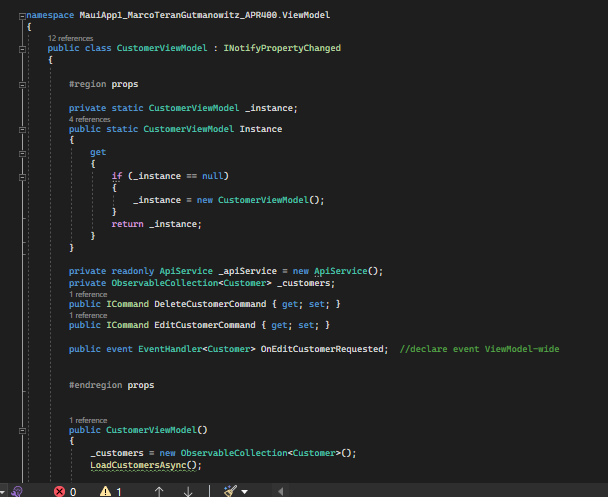


Figure Using singleton instance for ViewModel, utilizing ObservableCollection, Icommand and events in ViewModel

Please see code for more info on MVVM.

## For Grade B

As you can see from previous pictures, It Is possible to upload images for the Cars table. The images are displayed in the CarsPage.

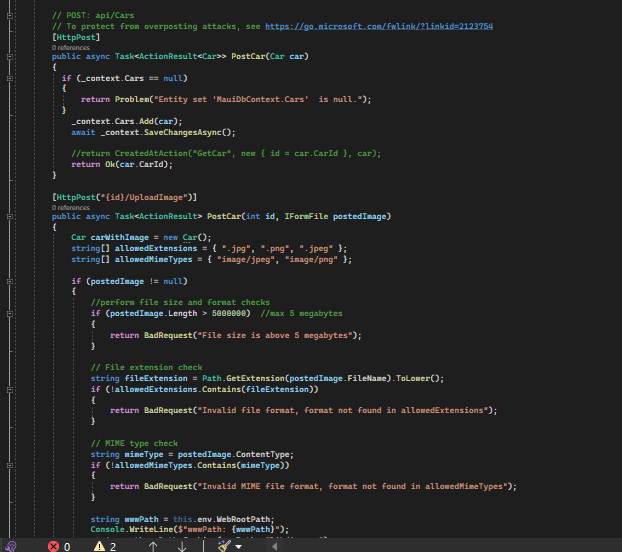


Figure - Code for the upload image method on API side

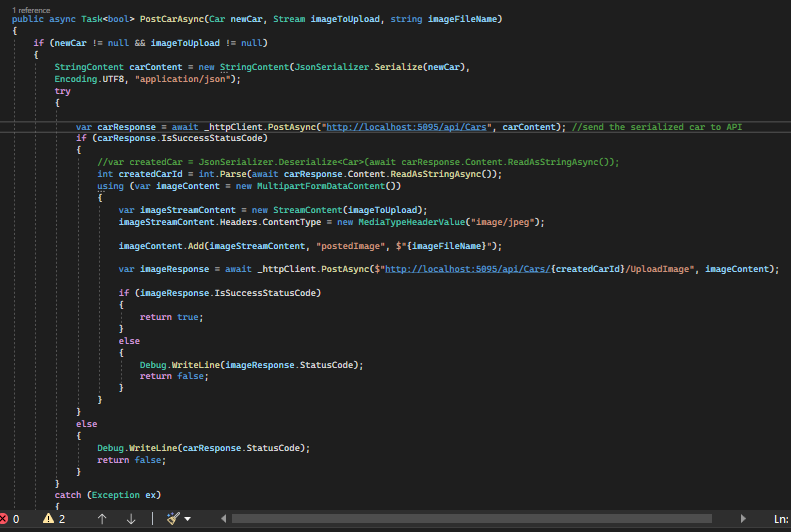


Figure - Method on MAUI app side (Services/ApiService.cs)

As you can see, the method on the MAUI side first POSTs the Car object through the default scaffolded controller method, and if the response is Success, then the image will be posted as well.

**Tabs Layout**

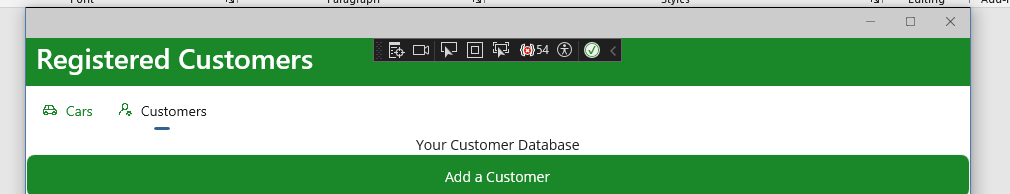
****

Figure - Tabs layout with Icons

The custom splash screen is not implemented due to Jan Olofs announcement on Canvas that it was not necessary.

**Design**

As you can see, I have tried to make the design fitting for an application that can be used on multiple platforms. There are not any functions that would work on one platform but not another et cetera.

**App Icon**

****