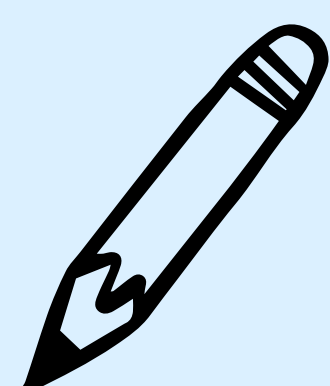


STEP-BY-STEP PLAN FOR DOT NET MAUI EXAMS

Folder structure for MVVM principle

Step 1



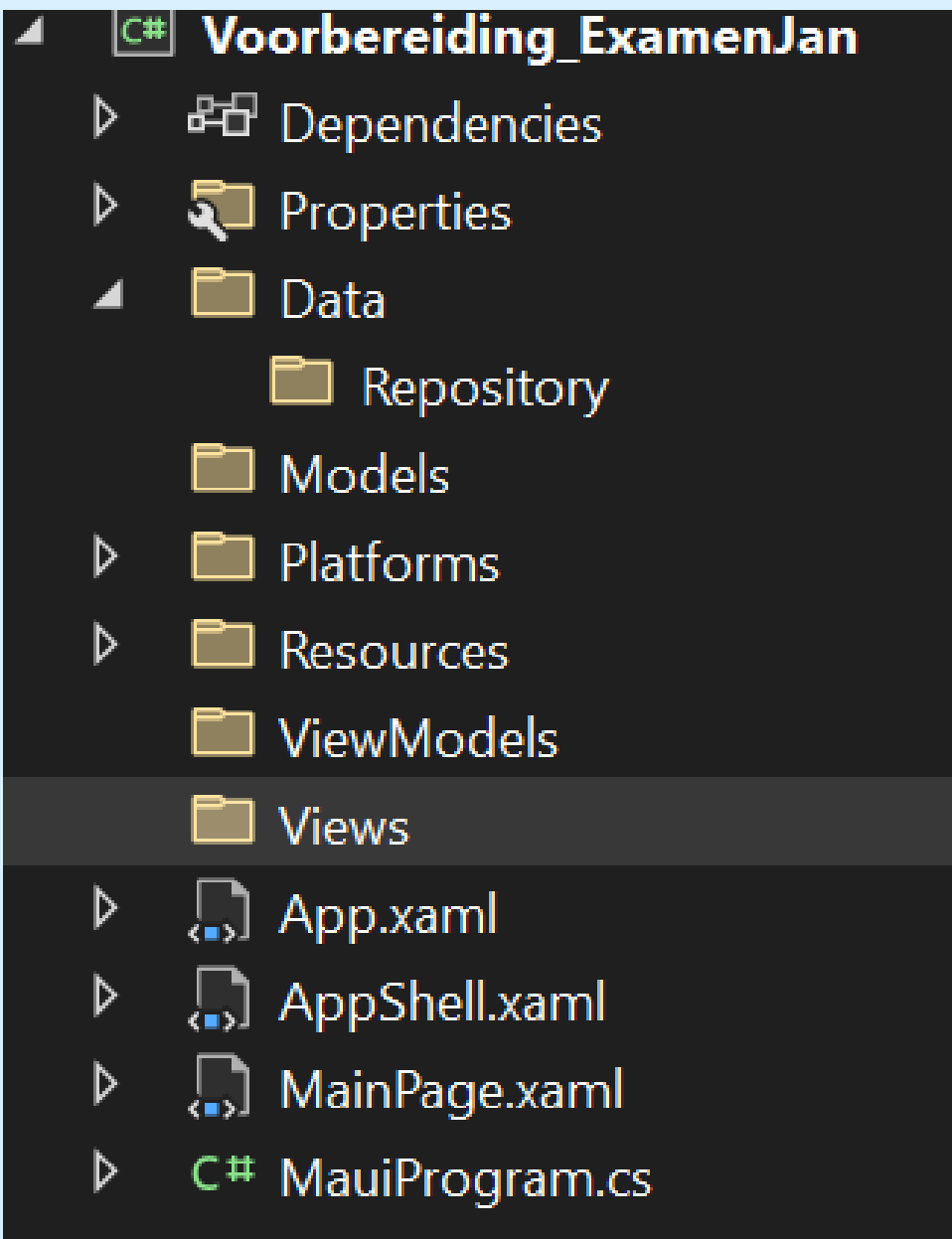
Create the necessary folders

Create the necessary folders (Data, Models, ViewModels, Views) to organize the project according to the MVVM principle.

Follow these steps to create the folder structure:

1. Create a new project in your preferred IDE.
2. In the project directory, create a folder named "Data". This folder will contain all the data-related classes and files.
3. Create a folder named "Repository" inside the "Data" folder.
4. Create a folder named "Models" inside the project directory.
5. Create a folder named "ViewModels" inside the project directory. This folder will contain all the view model classes.
6. Create a folder named "Views" inside the project directory. This folder will contain all the view classes.

By organizing your project according to the MVVM principle, you'll have a clear separation of concerns and a more maintainable codebase.



Step 2

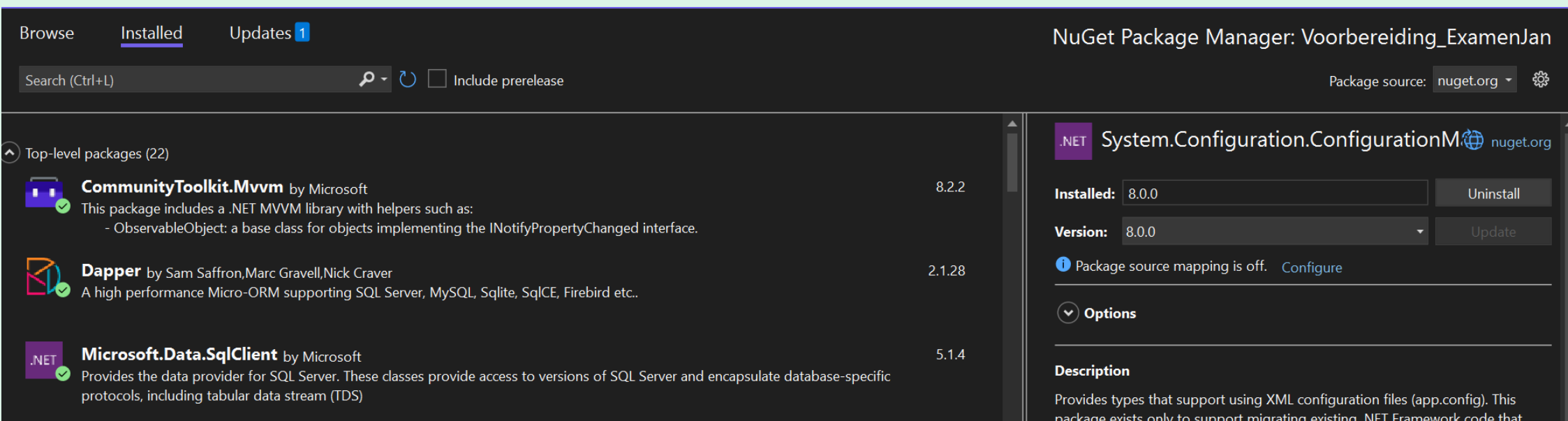


Install nugget packages

15 min

Follow these steps to install the necessary nugget packages for your Dot net Maui exams.

- Open the NuGet Package Manager in Visual Studio.
- Search for the "CommunityToolkit.Mvvm" package and install it.
- Search for the "Dapper" package and install it.
- Search for the "Microsoft.Data.SqlClient" package and install it.
- Search for the "System.Configuration.ConfigurationManager" package and install it.



Step 3

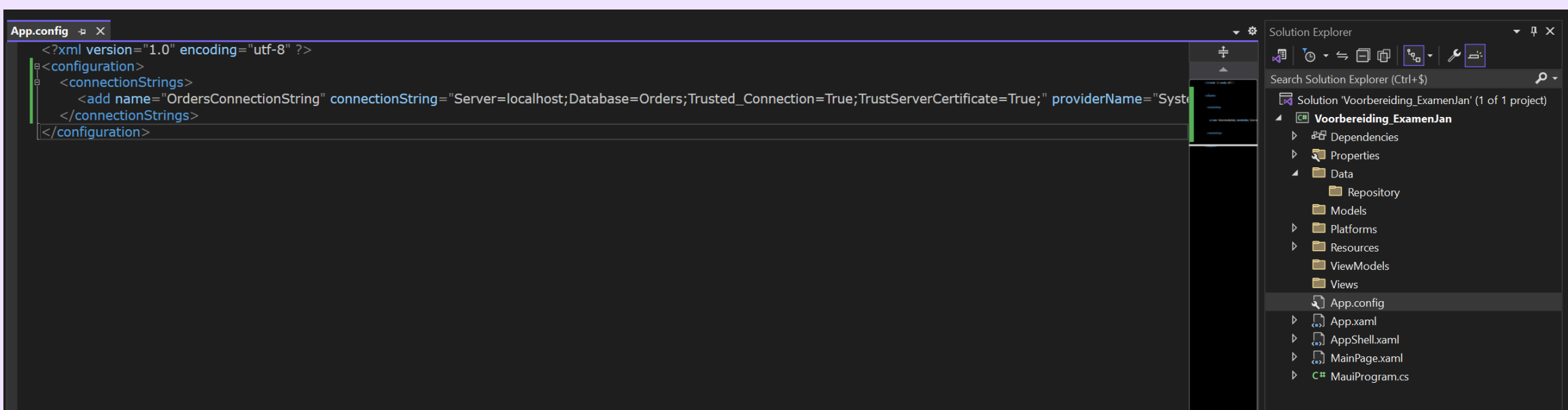


Add app.config file and insert the connection string

Follow the steps below to add the app.config file and insert the connection string:

Step 1: Create the app.config file

In the root folder of your project, create a new file named app.config.



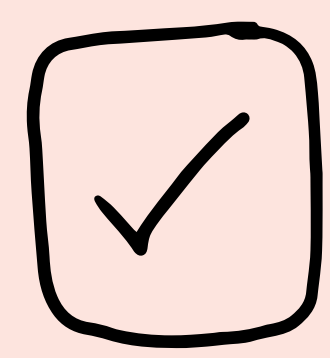
Step 2: Insert the connection string

Open the app.config file and insert the following code:

```
<configuration>
  <connectionStrings>
    <add name="MyConnectionString"
    connectionString="YOUR_CONNECTION_STRING_HERE" />
  </connectionStrings>
</configuration>
```

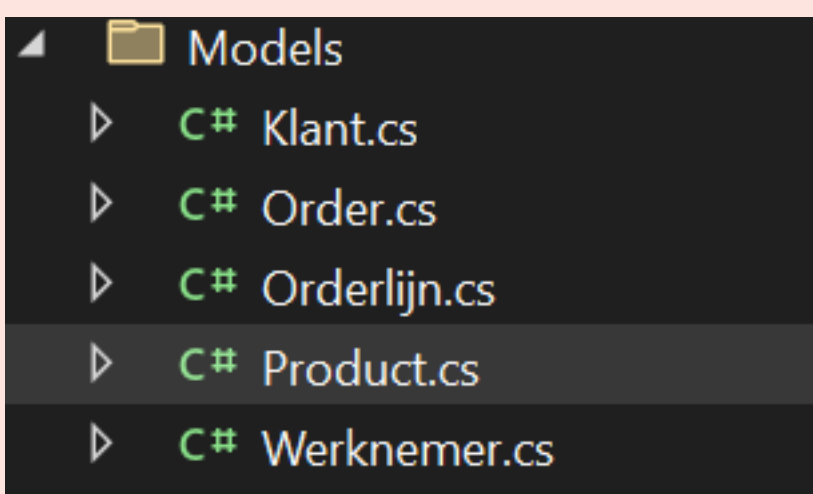
Replace YOUR_CONNECTION_STRING_HERE with your actual connection string.

Step 4

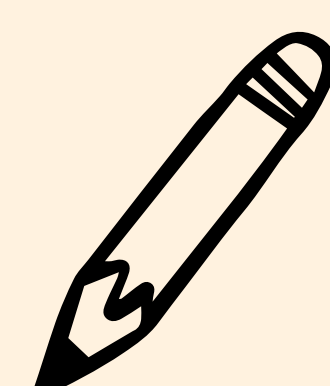


Create All the Models

Create all the necessary models for the project.



Step 5



Create the BaseRepository and DatabaseConnection files

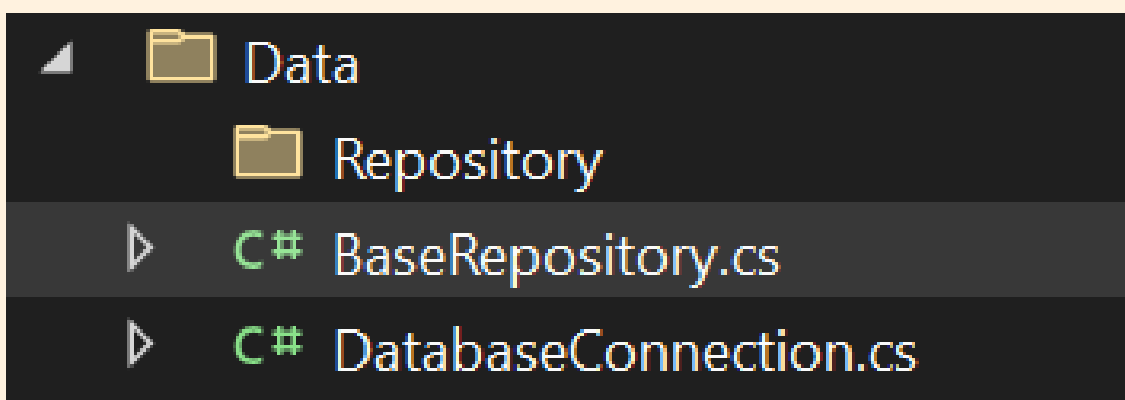
Follow the steps below to create the necessary files in the "Data" folder.

Step 1: Create the BaseRepository file

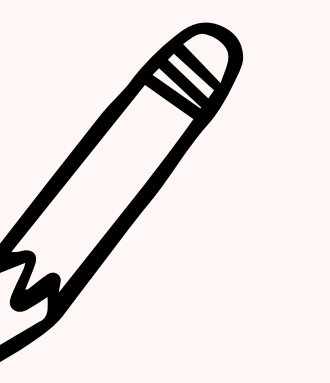
Create a new file named "BaseRepository.cs" in the "Data" folder.

Step 2: Create the DatabaseConnection file

Create another new file named "DatabaseConnection.cs" in the "Data" folder.



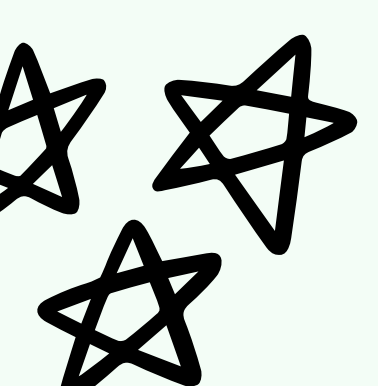
Step 6



Create Repositories and Interfaces

Implement the repositories and interfaces in the "Repository" folder.

Step 7



Create the view models

In this activity, we will create the view models for our Dot net Maui exams project.

Step 1: Create the BaseViewModel

Step 2: Create the Other ViewModels

Step 8



Create the views

Build the user interface for each view.

Make the routing in Appshell and MauiProgram.cs