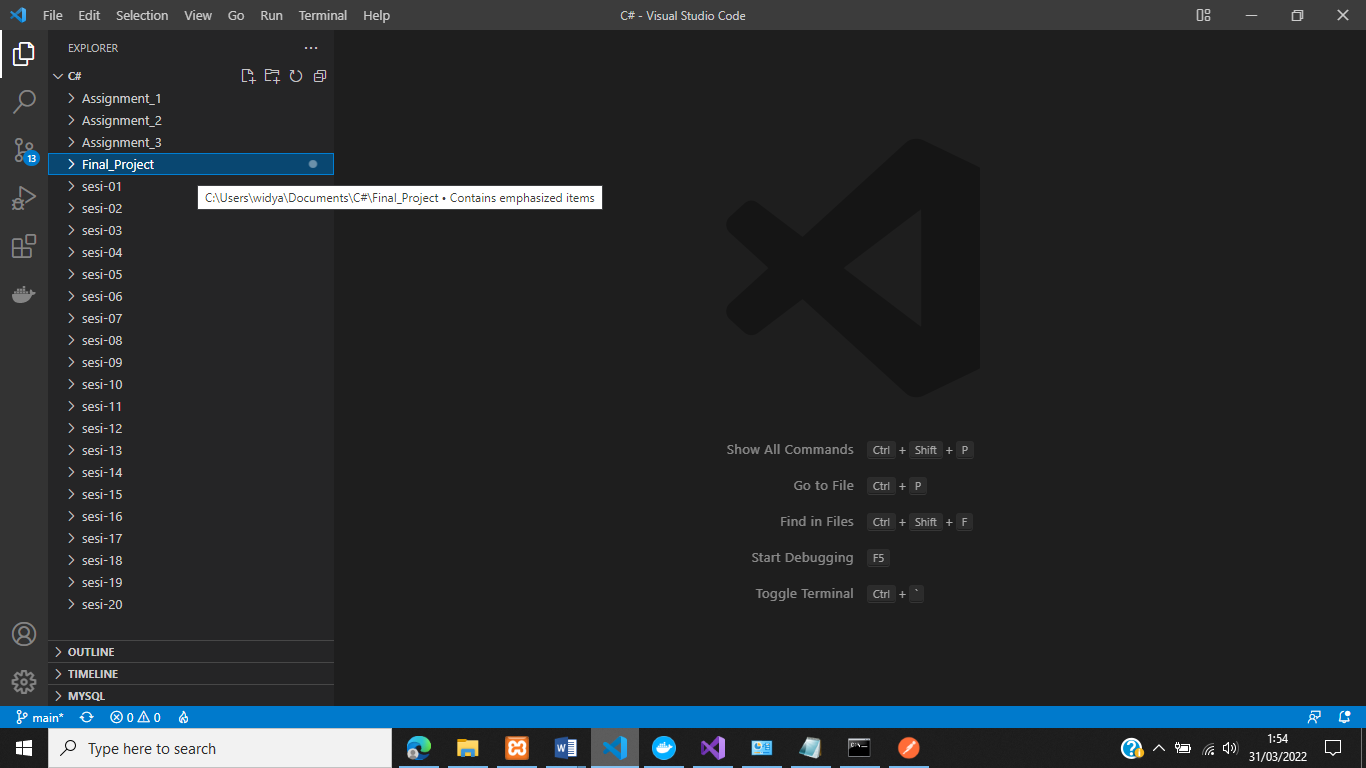
**Nama : Dini Tri Widianingsih**

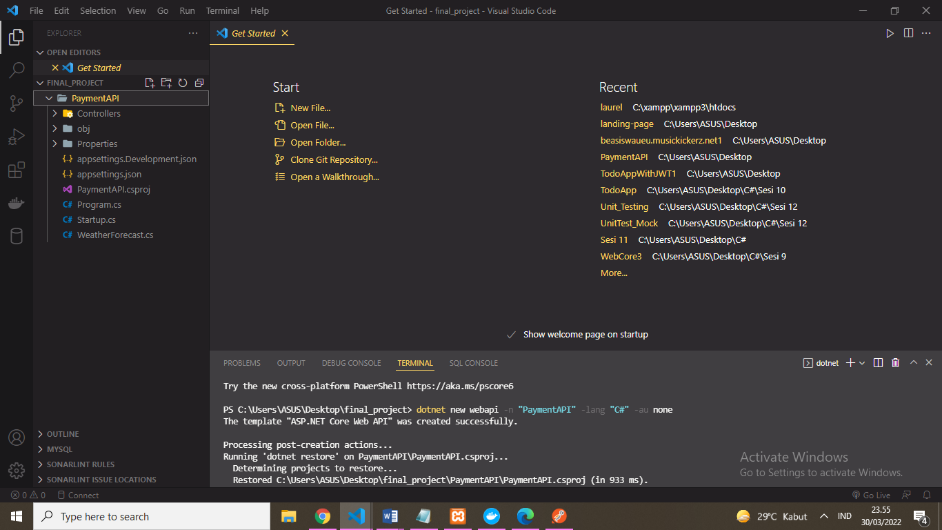
**No. Peserta : FSDO003ONL007**

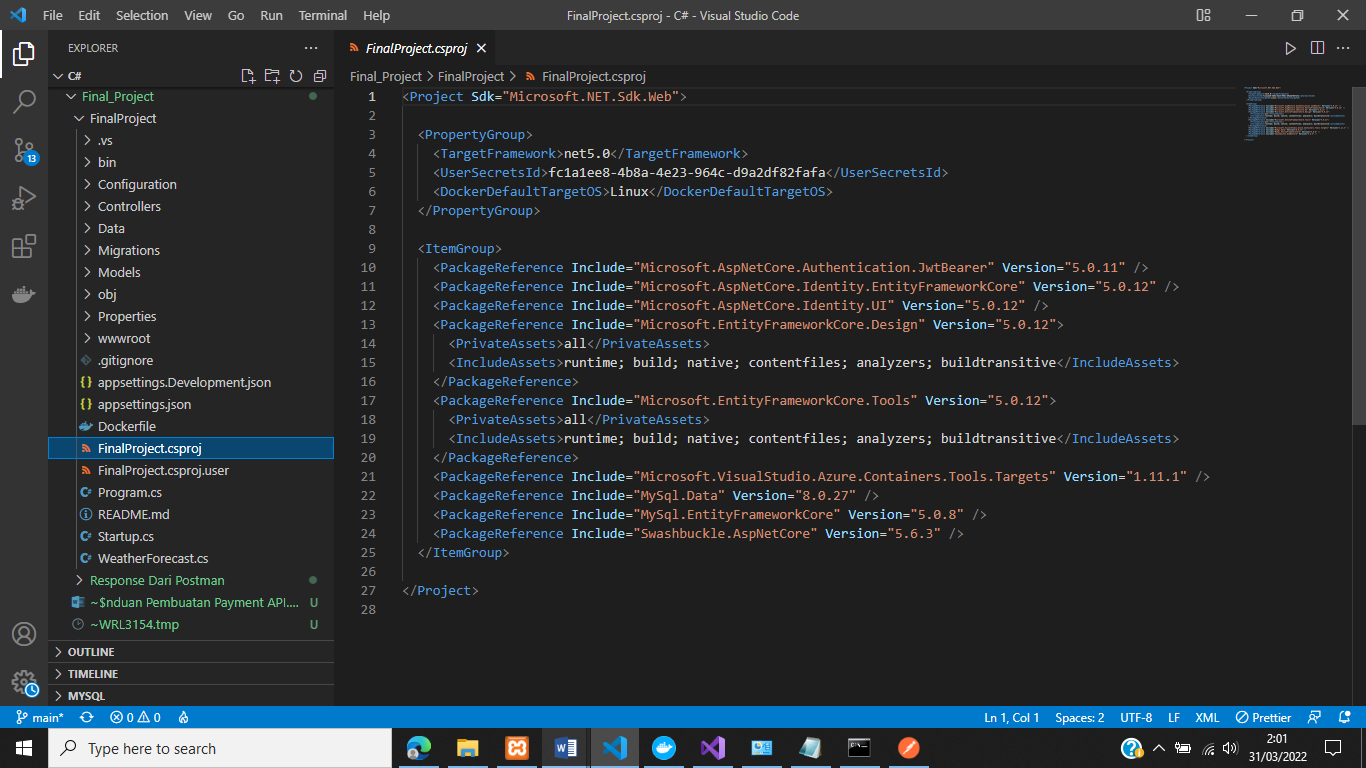
**PANDUAN PEMBUATAN FINAL PROJECT**

**PEMBUATAN**

* Buka Visual Studio Code > Create folder Final\_Poject

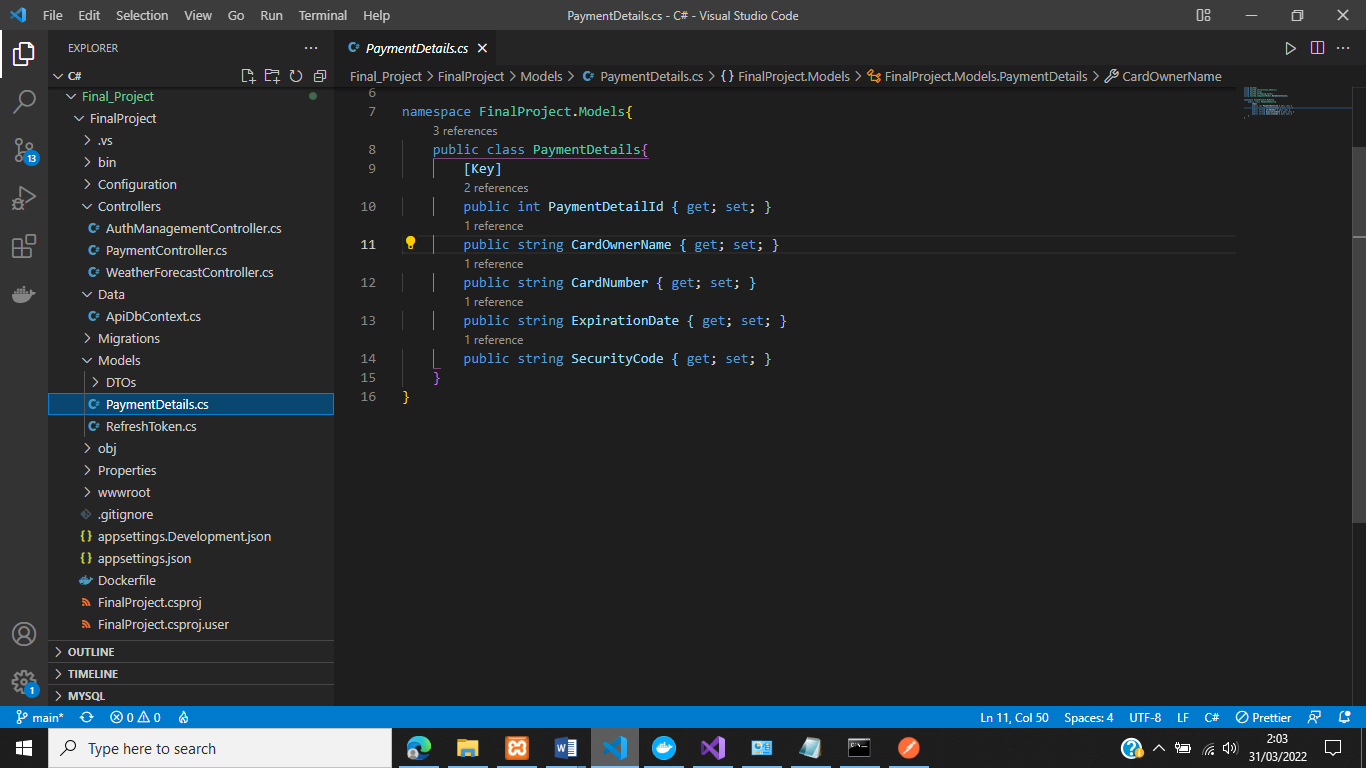


* Inisiasi project di dalam folder tersebut > ***dotnet new webapi -n "Final\_Project" -lang "C#" -au none*** lalu akan mucul folder – folder baru
* Install package yang dibutuhkan
* dotnet add package NuGet.CommandLine.XPlat --version 5.11.0
* dotnet add package Microsoft.EntityFrameworkCore.Tools --version 5.0.11
* dotnet add package Microsoft.EntityFrameworkCore.Design --version 5.0.11
* dotnet add package Microsoft.AspNetCore.Authentication.JwtBearer --version 5.0.11
* dotnet add package Microsoft.AspNetCore.Identity.EntityFrameworkCore --version 5.0.11
* dotnet add package Microsoft.AspNetCore.Identity.UI --version 5.0.11
* dotnet add package Microsoft.Mysql.Data --version 8.0.28
* dotnet add package Microsoft.Mysql.EntityFrameworkCore --version 5.0.10



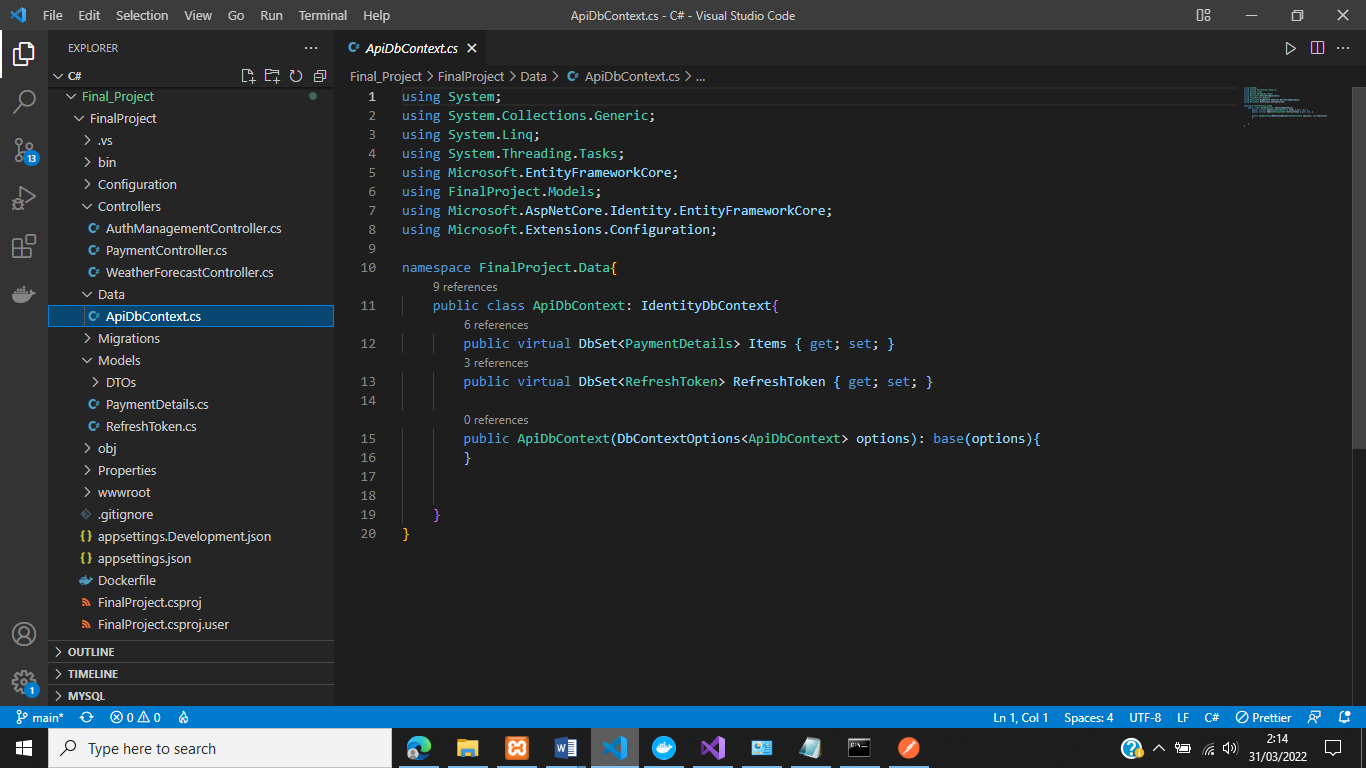
* Buat folder Models, lalu tambahkan file PaymentDetails.cs di dalam folder tersebut

*Location : Final\_Project/Models/PaymentDetails.cs*



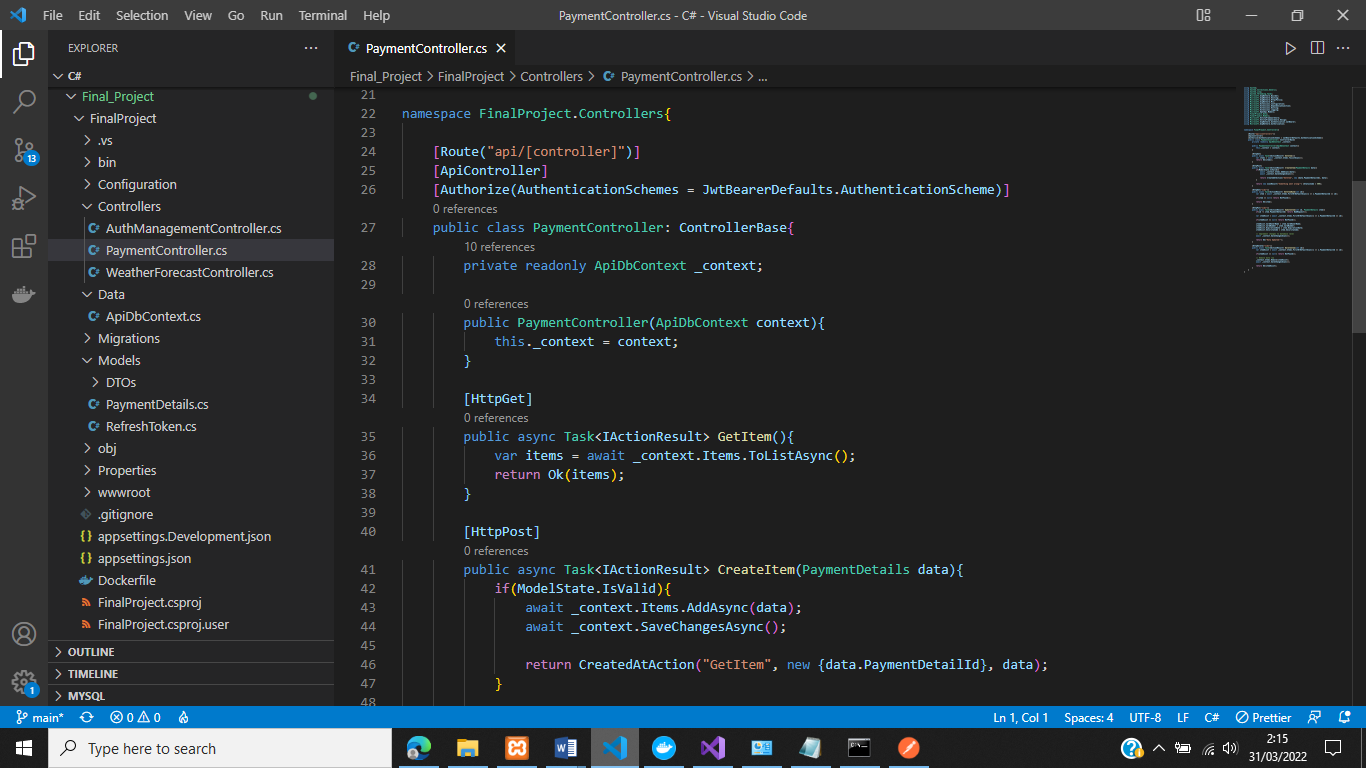
* Buat folder Data, lalu tambahkan file ApiDBContext

*Location : Final\_Project/Data/ApiDbContext.cs*



* Buat folder Controllers, lalu lanjutkan build PaymentDetailController untuk method post, get, put, delete

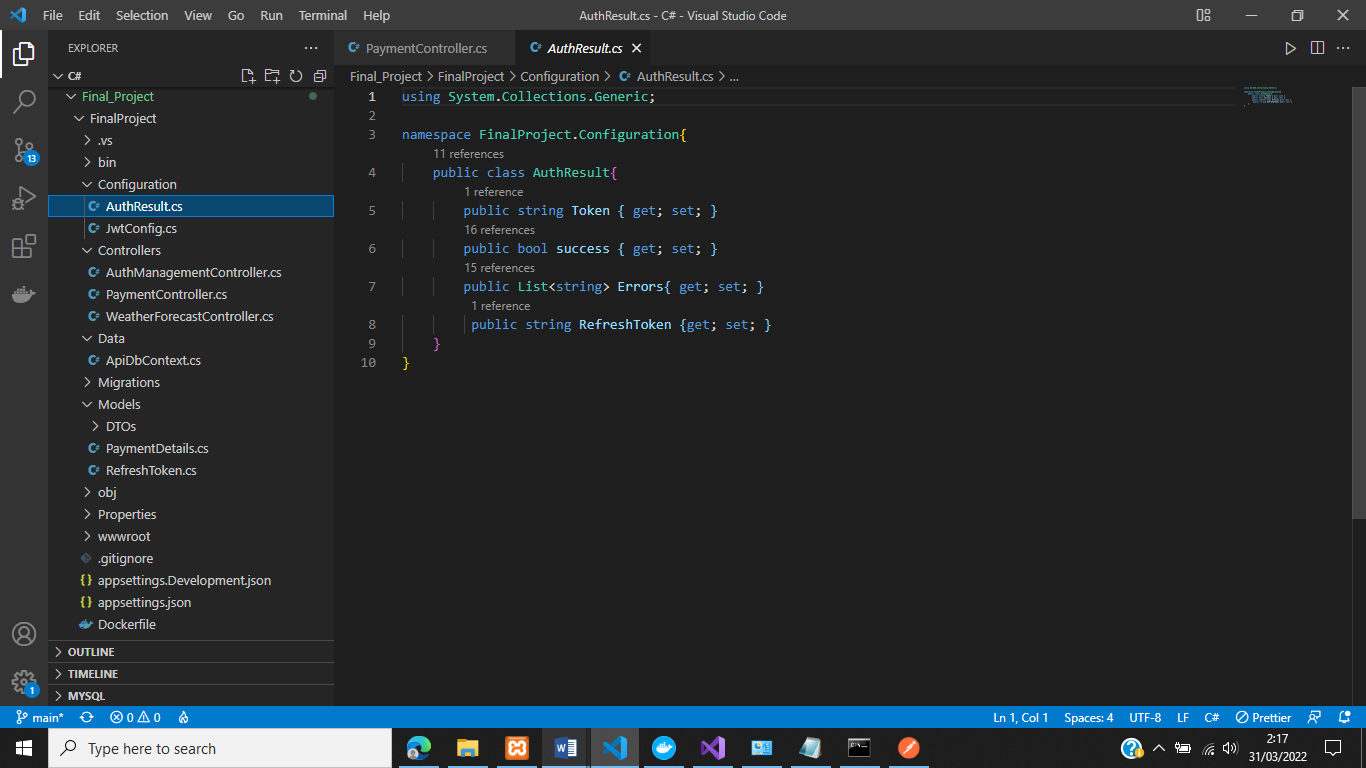
*Location : Final\_Project/Controllers/PaymentController.cs*



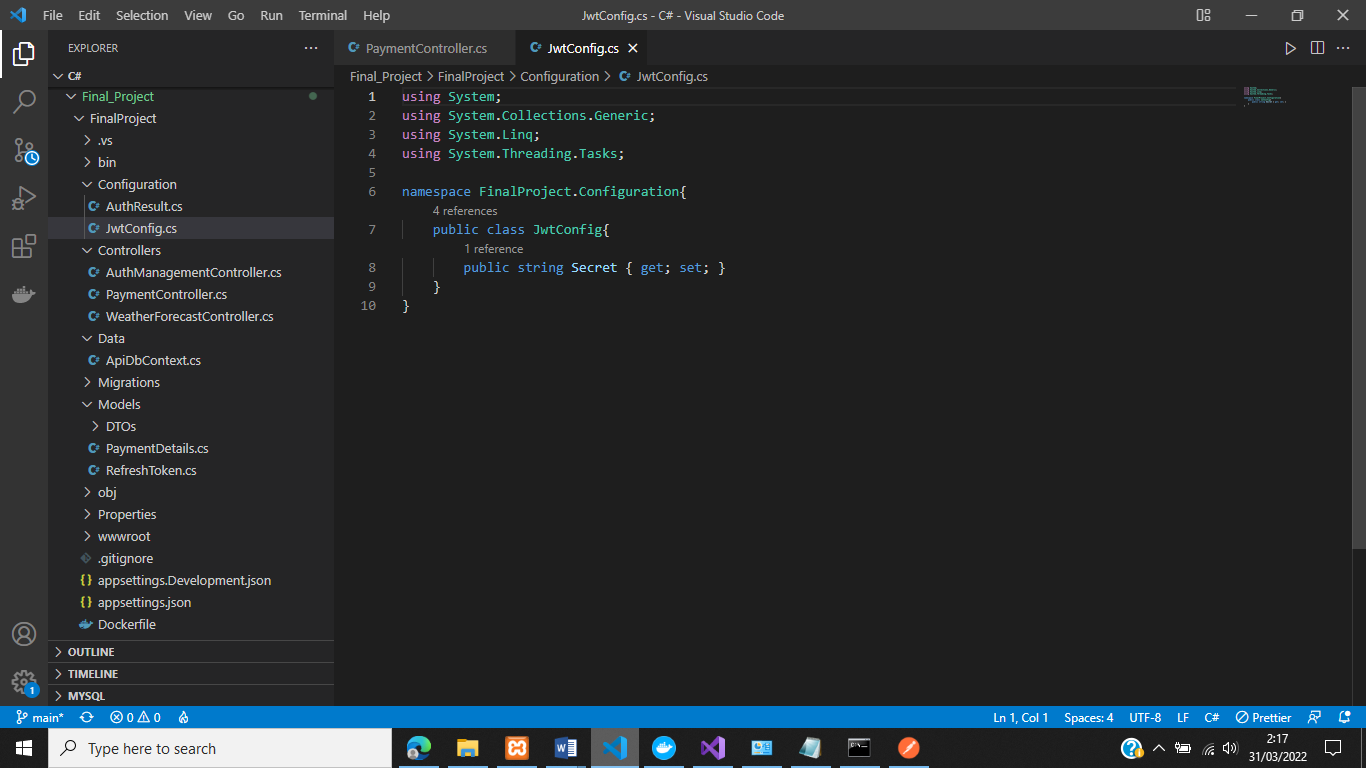
**Penambahan JWT Token dan Refresh Token**

* Buat folder Configuration, lalu tambahkan file AuthResult.cs dan JwtConfig.cs:

*Location : Final\_Project/Configuration/AuthResult.cs*

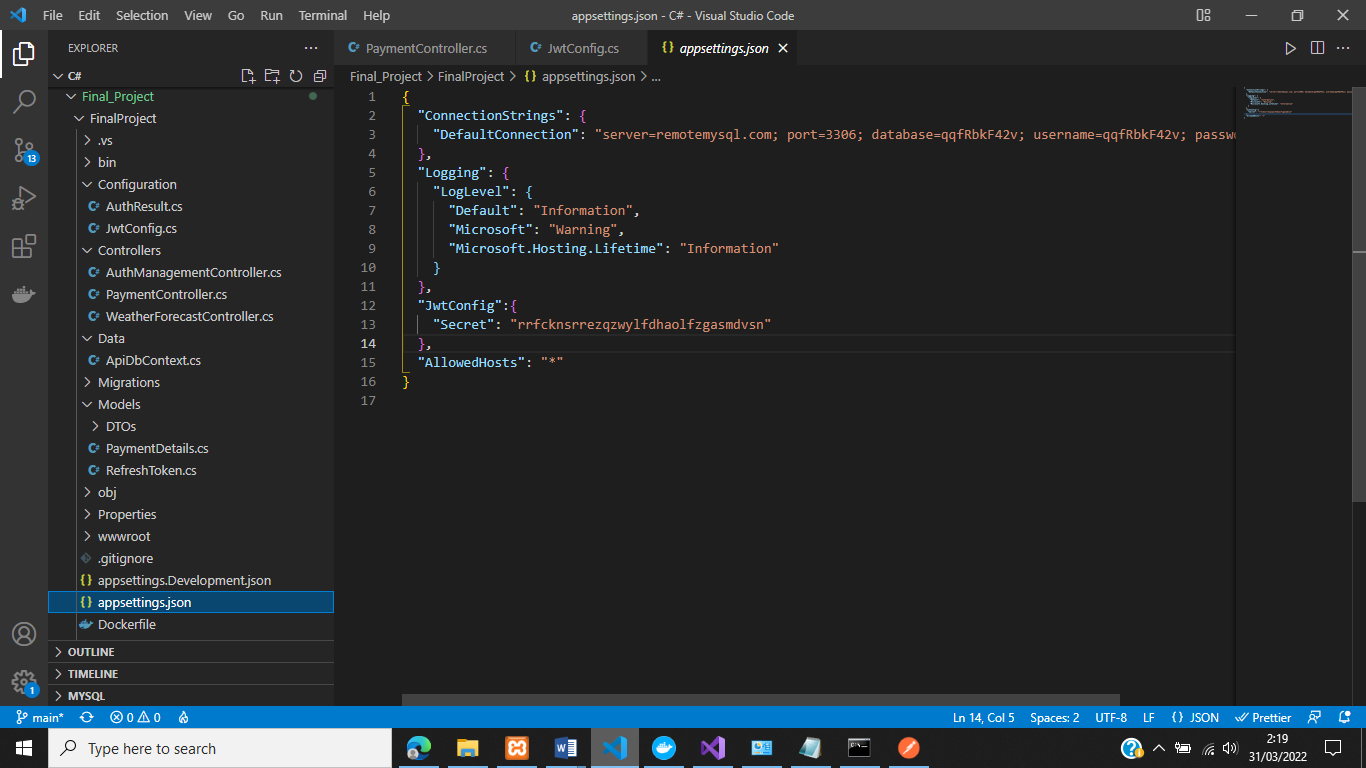


*Location : Final\_Project/Configuration/JwtConfig.cs*



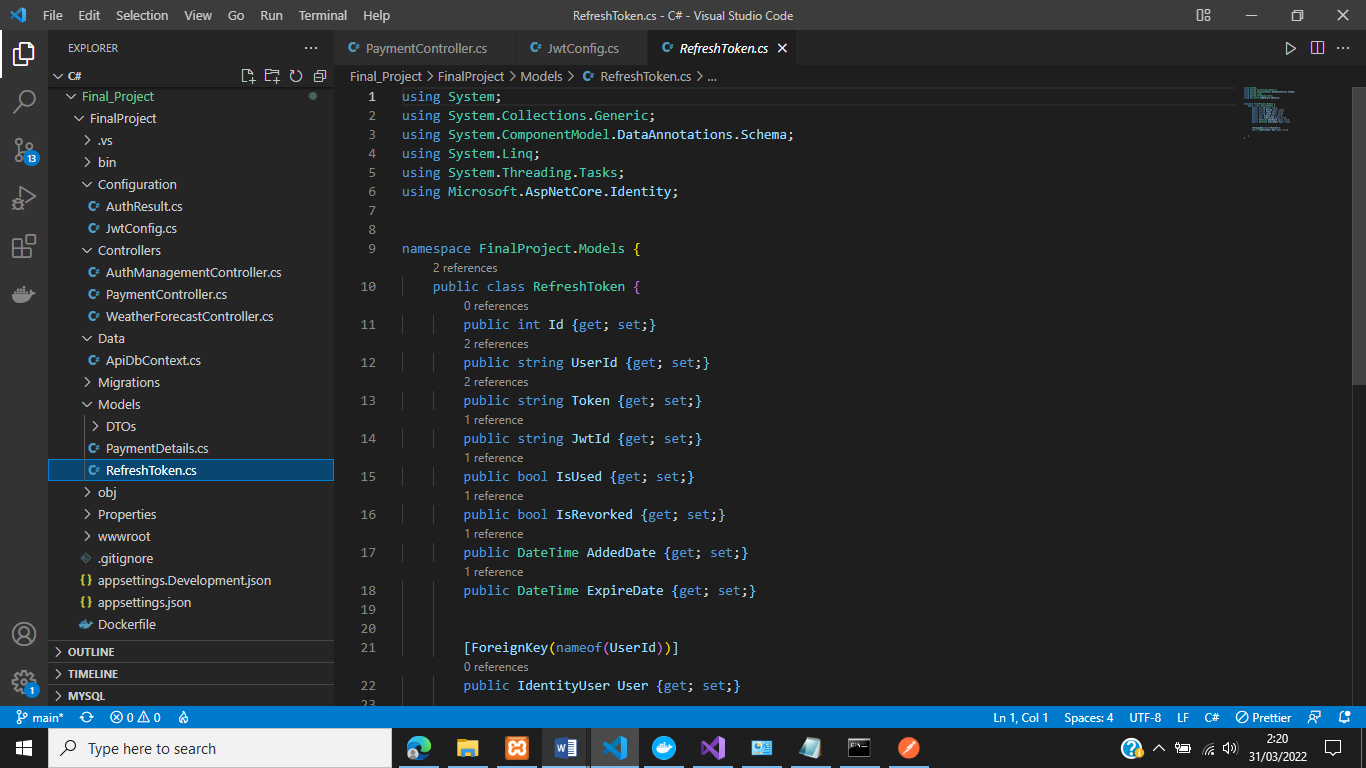
* Pada file appsettings.json tambahkan JwtConfig.Secrets

*Location : Final\_Project/appsettings.json*



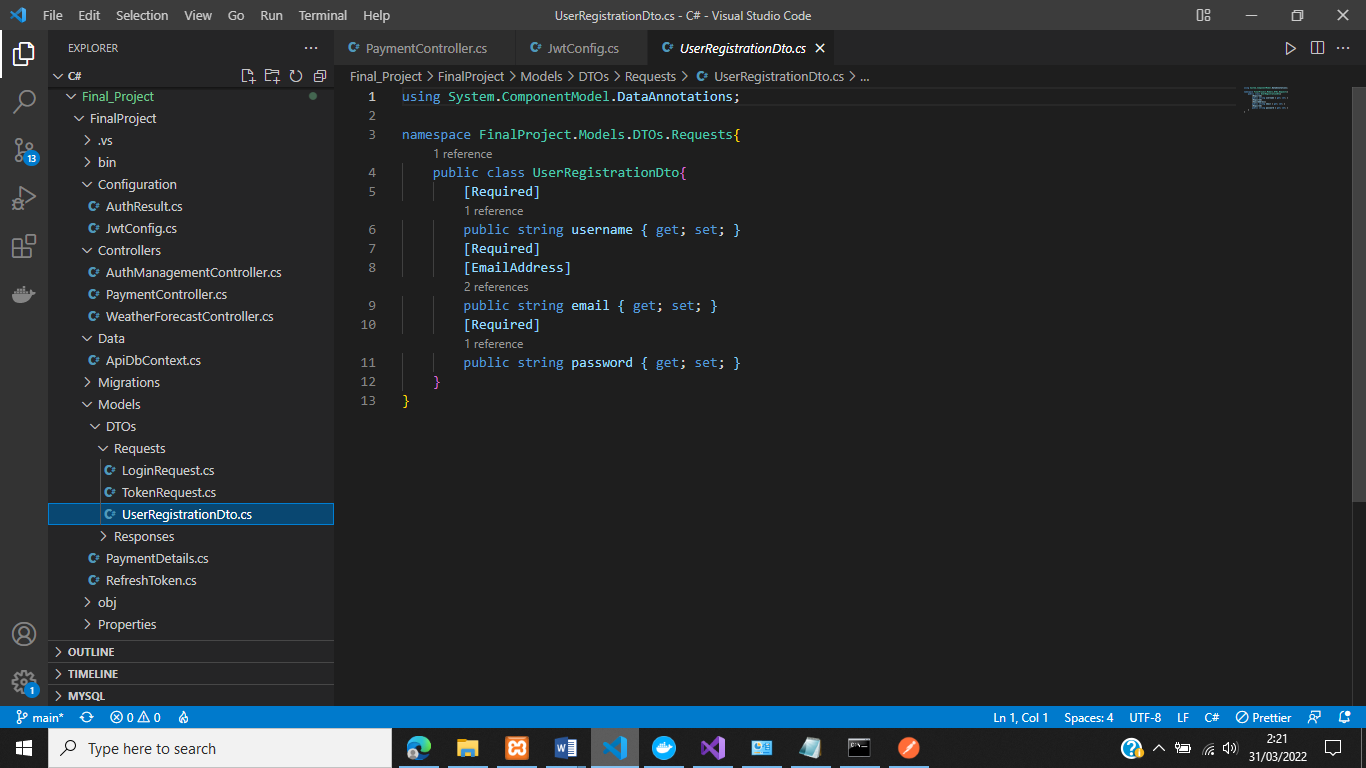
* Buat file baru RefreshToken.cs yang terdapat pada folder Models

*Location : Final\_Project/Models/RefreshToken.cs*

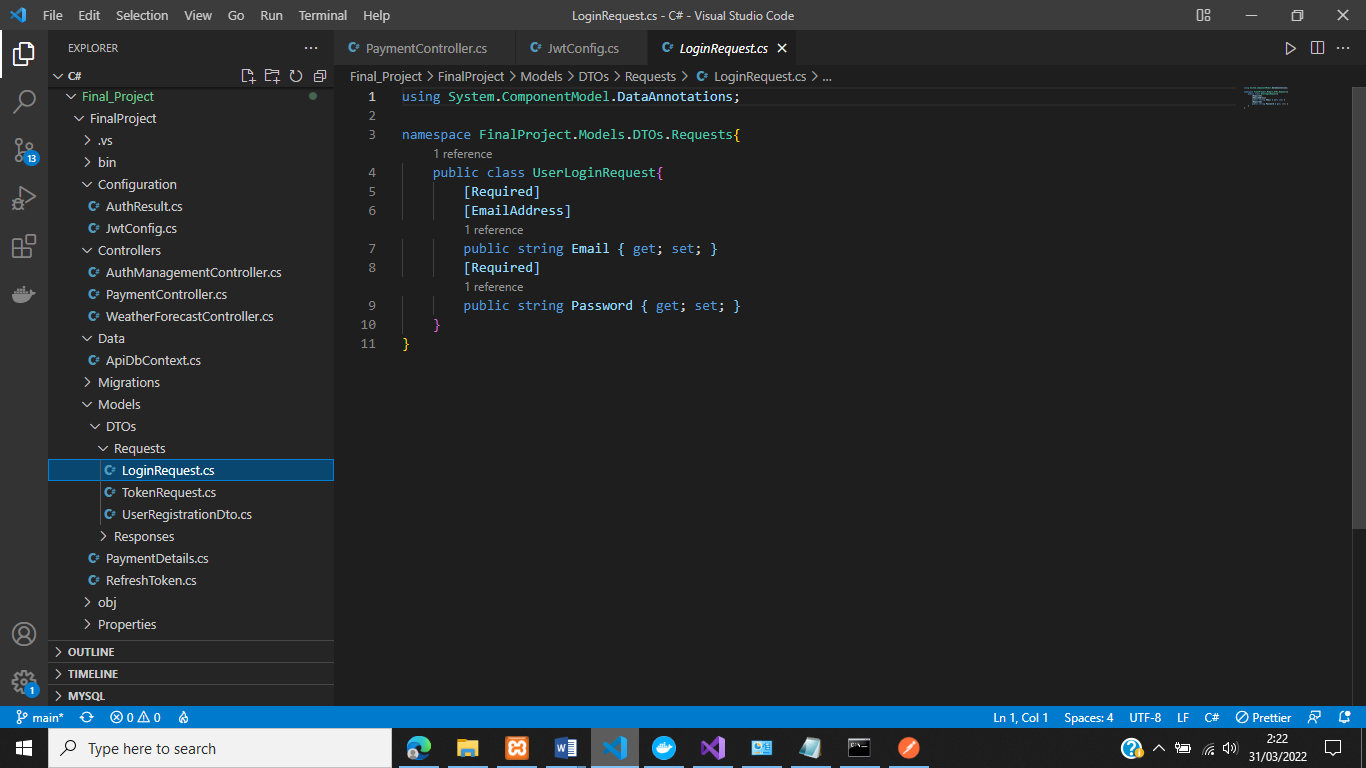


* Buat folder baru bernama DTOs yang di dalamnya terdapat juga folder Request & Response
* **Folder Request**

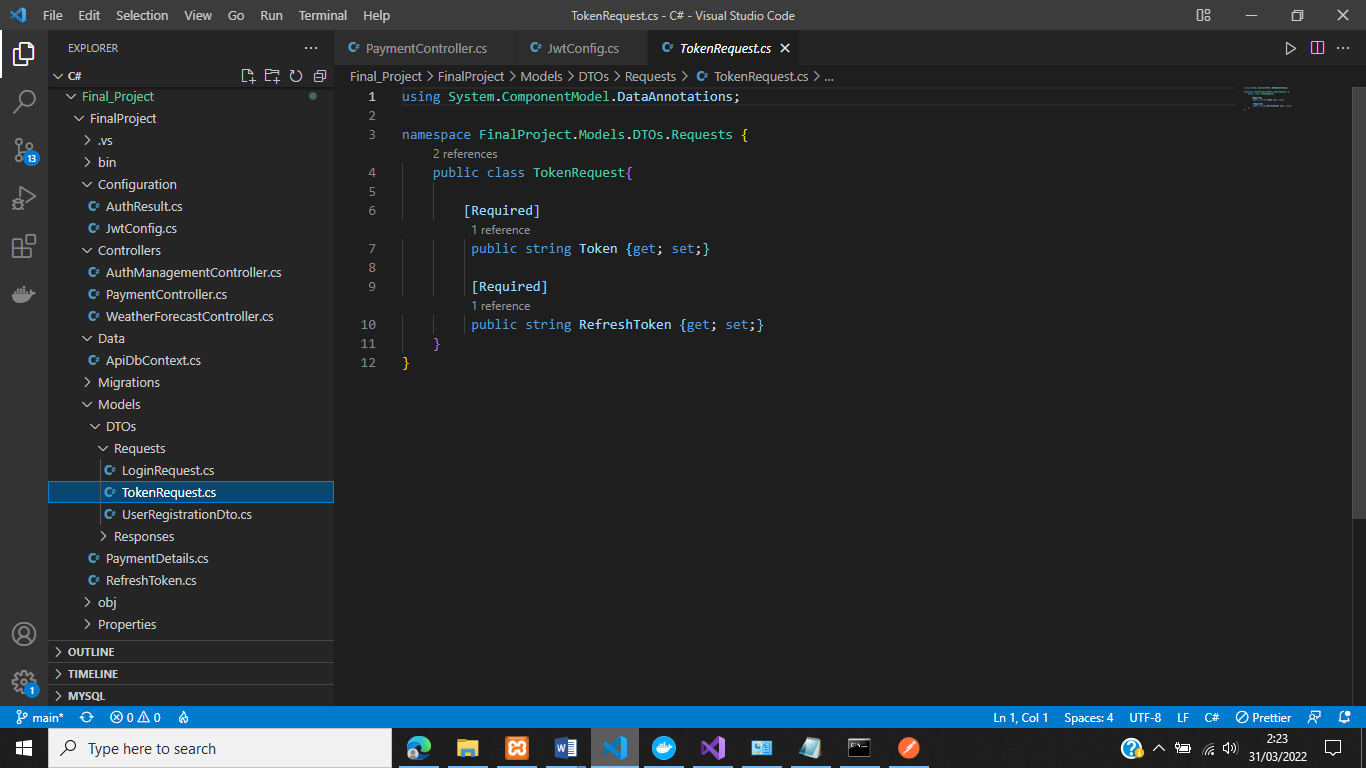
*Location : Final\_Project/Models/DTOs/Requests/UserRegistrationDto.cs*



*Location : Final\_Project/Models/DTOs/Requests/LoginRequest.cs*

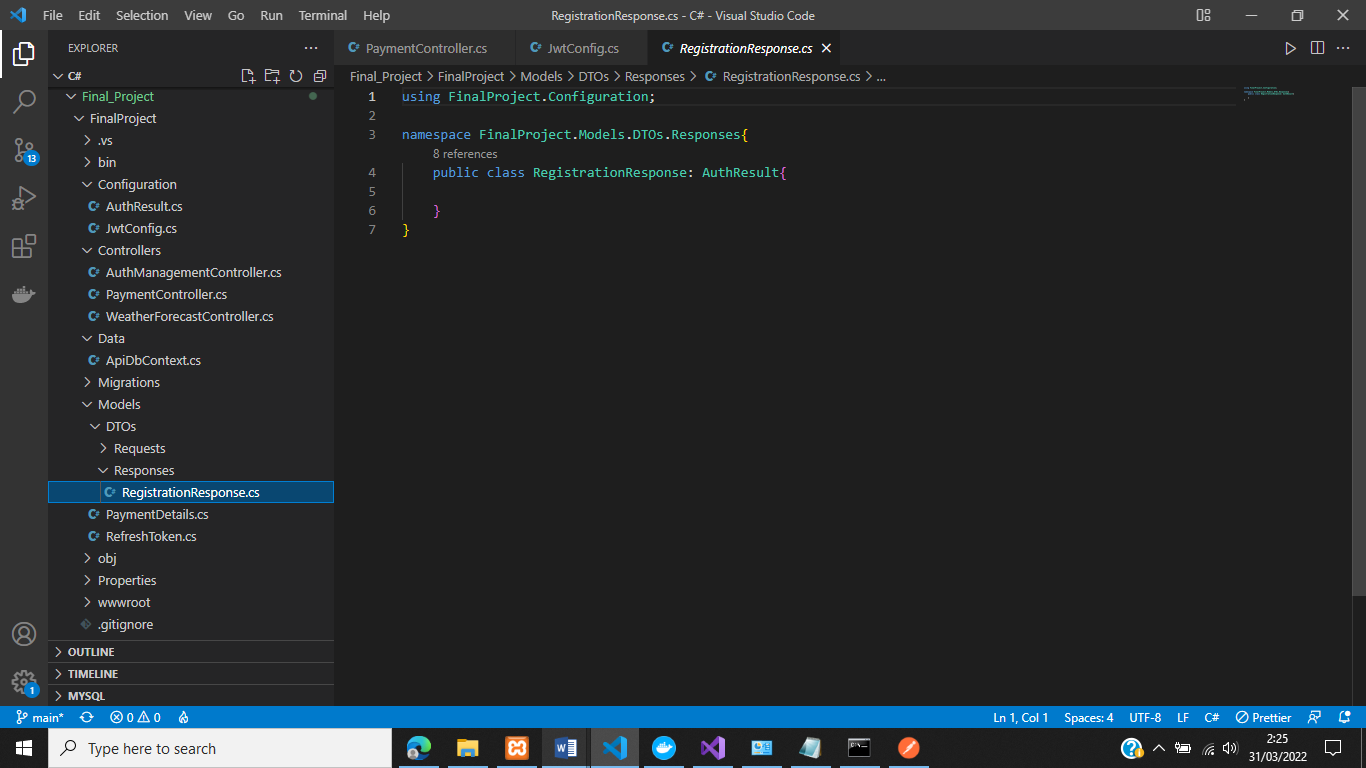


*Location : Final\_Project/Models/DTOs/Requests/TokenRequest.cs*



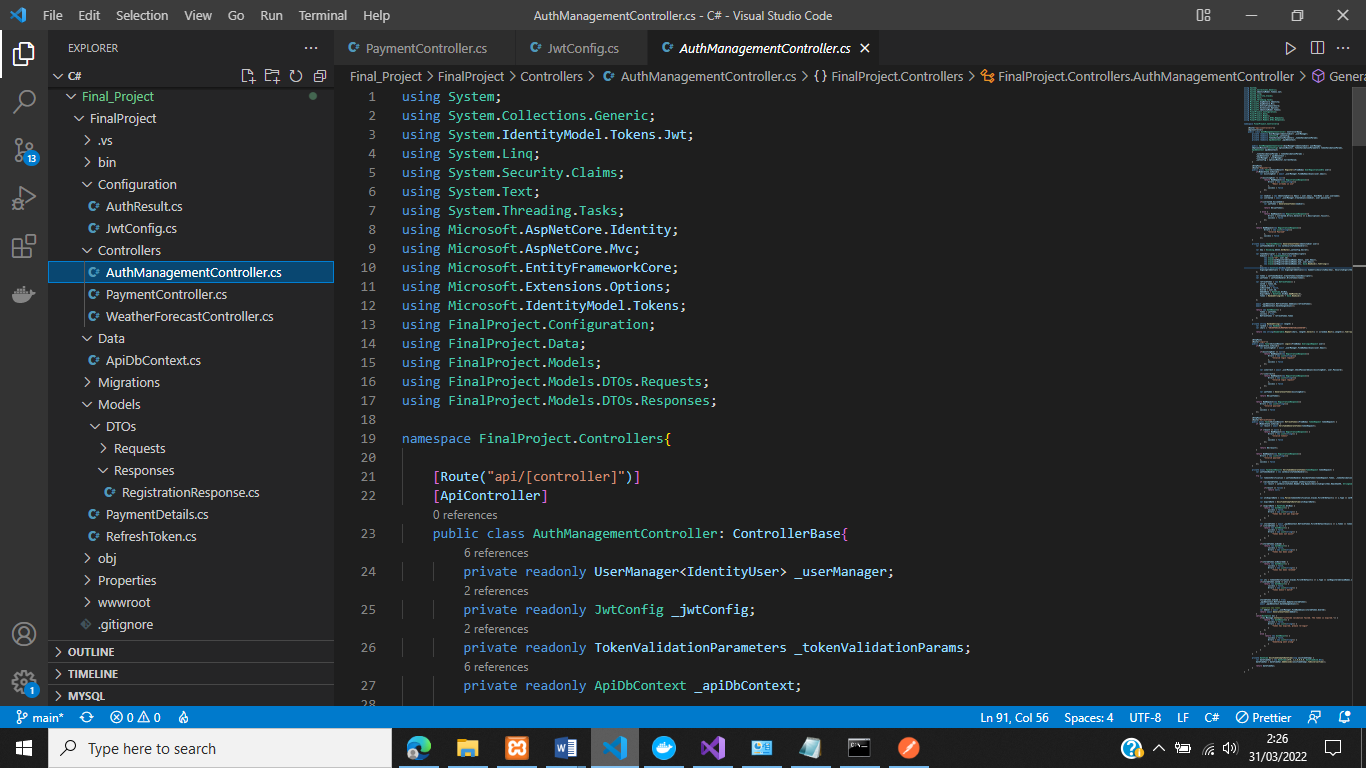
* **Folder Response**

*Location : Final\_Project/Models/DTOs/Response/RegistrationResponse.cs*



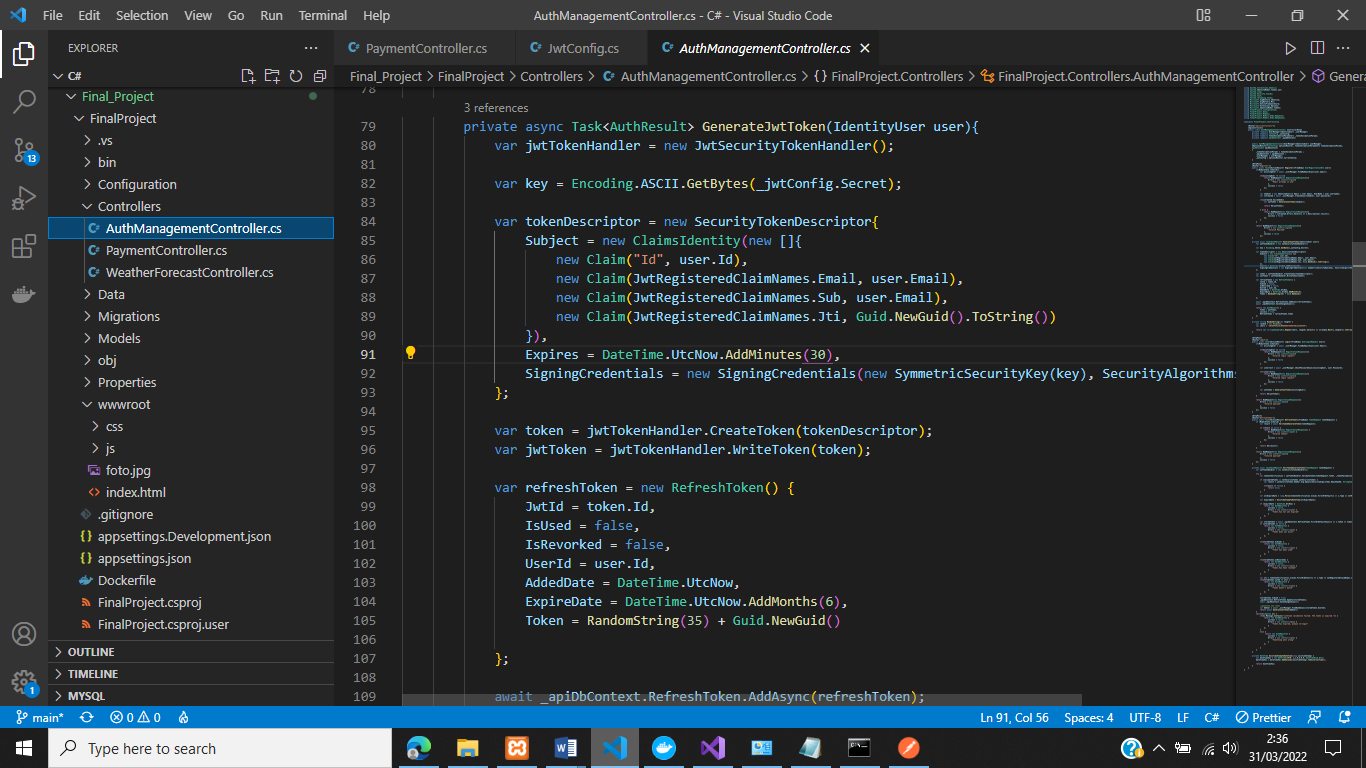
* Tambahkan file AuthManagement pada folder Controllers

*Location : Final\_Project/Controllers/AuthManagementController.cs*



* JWT dengan batas waktu token 30 menit

Set waktu expired selama 30 menit pada fungsi GenerateJwtToken di AuthManagementController.cs



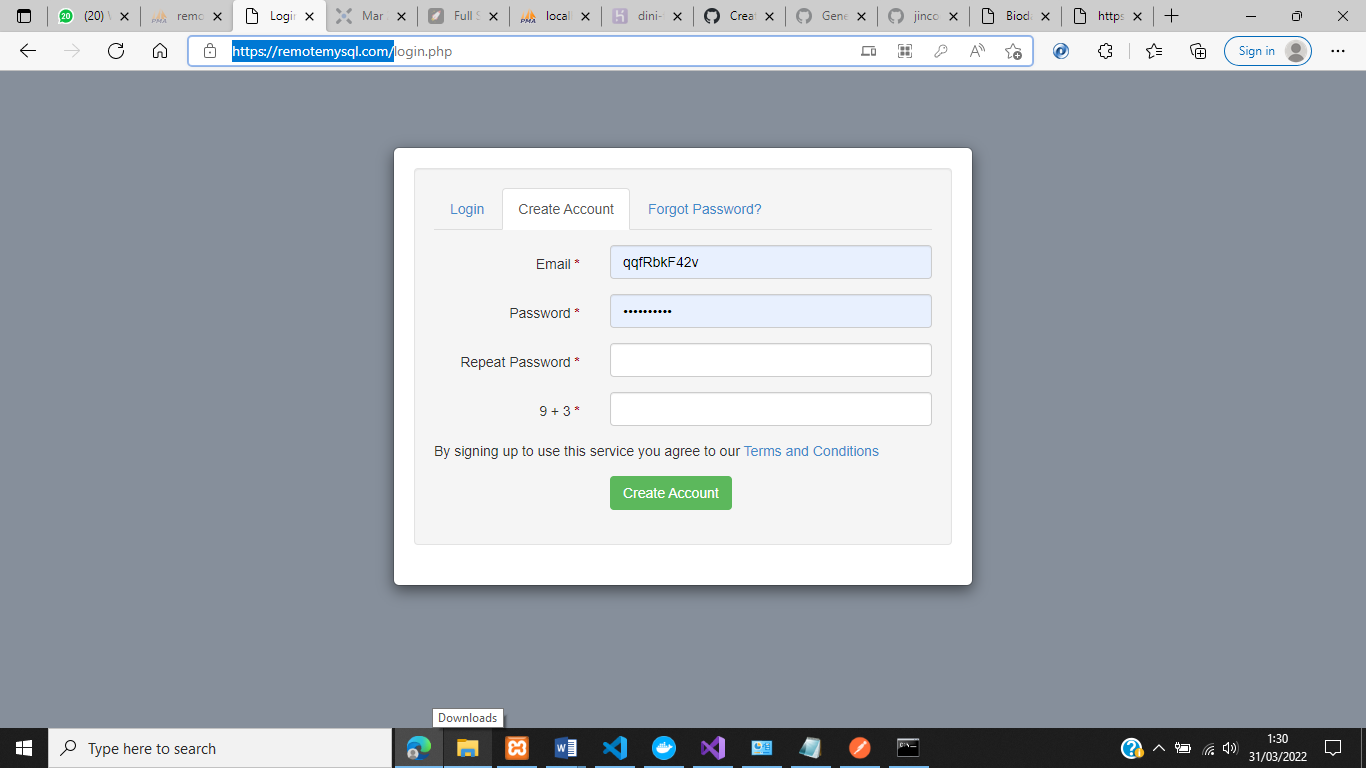
* Pembuatan Landing page

Pada project Final\_Project buat file “wwwroot” yang berisi file index.html sebagai landing page.



**DATABASE**

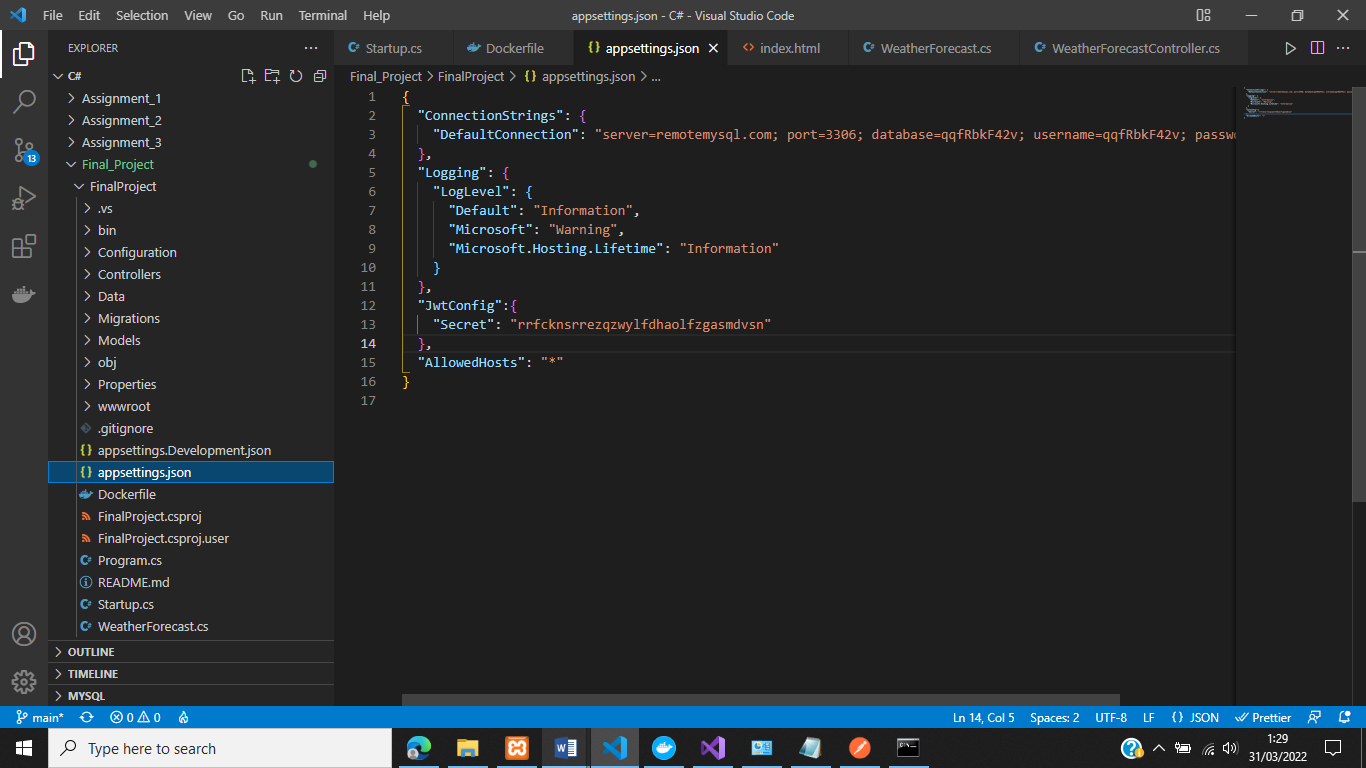
* Buat Akun di <https://remotemysql.com/>



* Setelah masuk dashboard, klik link : <https://remotemysql.com/databases.php?action=new> untuk membuat database

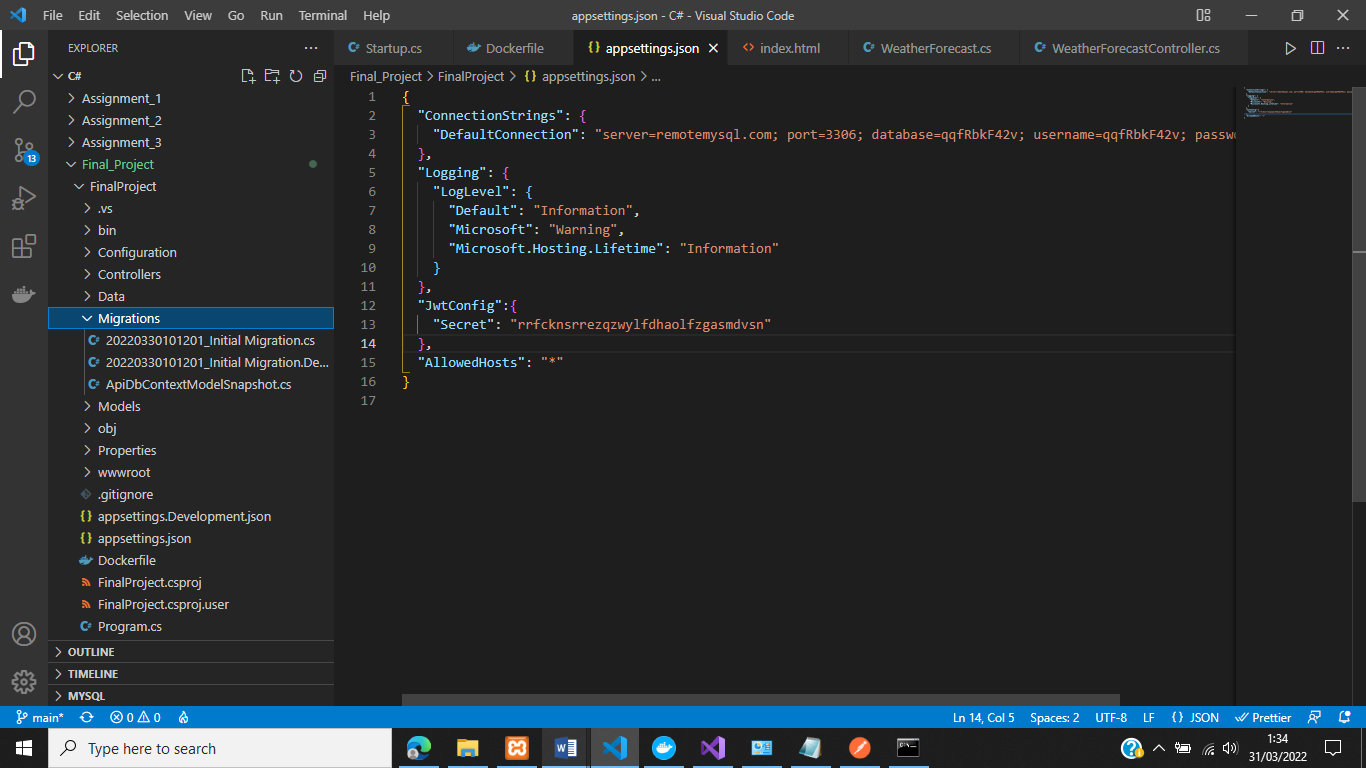
((disini kita akan menerima username, password, dll yang kemudian akan disesuaikan pada file appsettings.json))

* Sesuaikan database didalam appjsettings.json



* Lakukan Migrasi ***dotnet ef migrations add "Initial Migrations"***

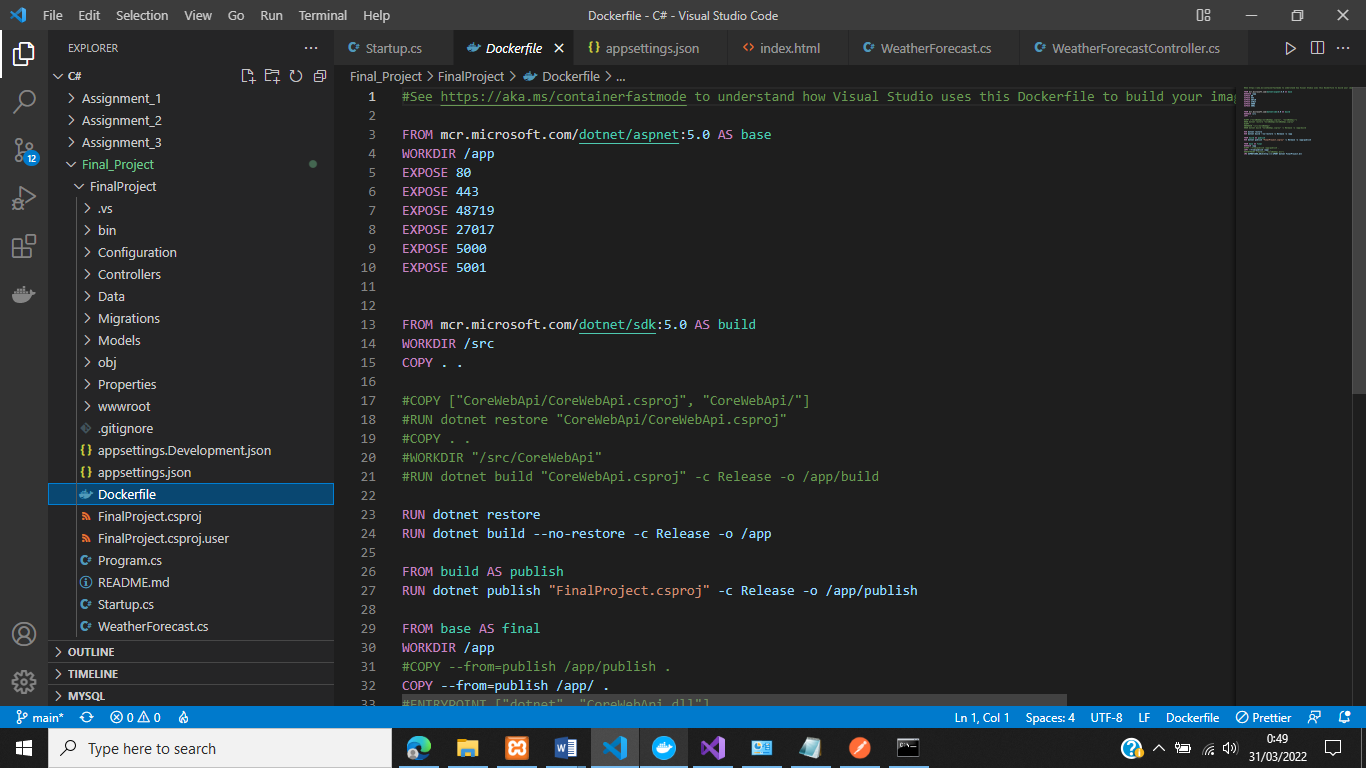
Maka akan muncul File Migrations di dalam Folder Final\_Project



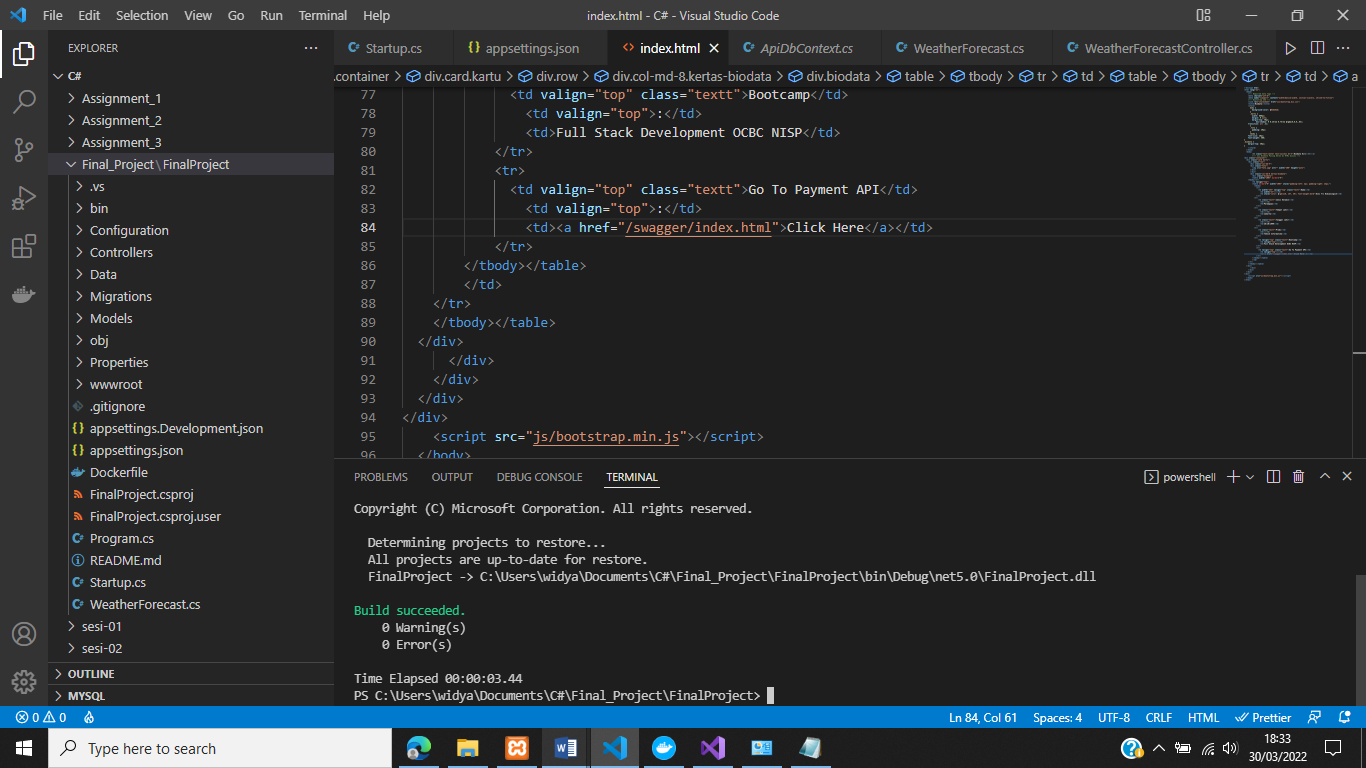
* Lakukan Database Update.

**DOCKER**

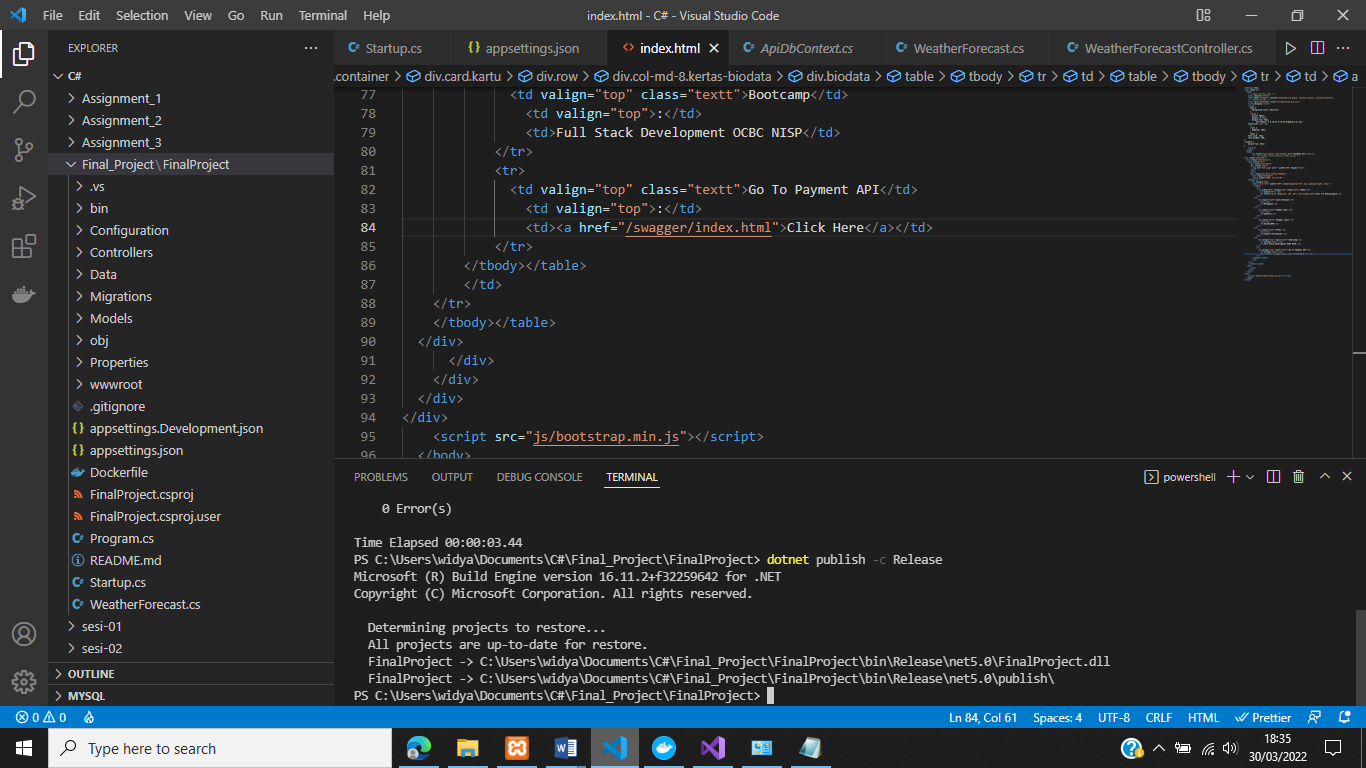
* Buat file Dockerfile di dalam folder Final\_Project



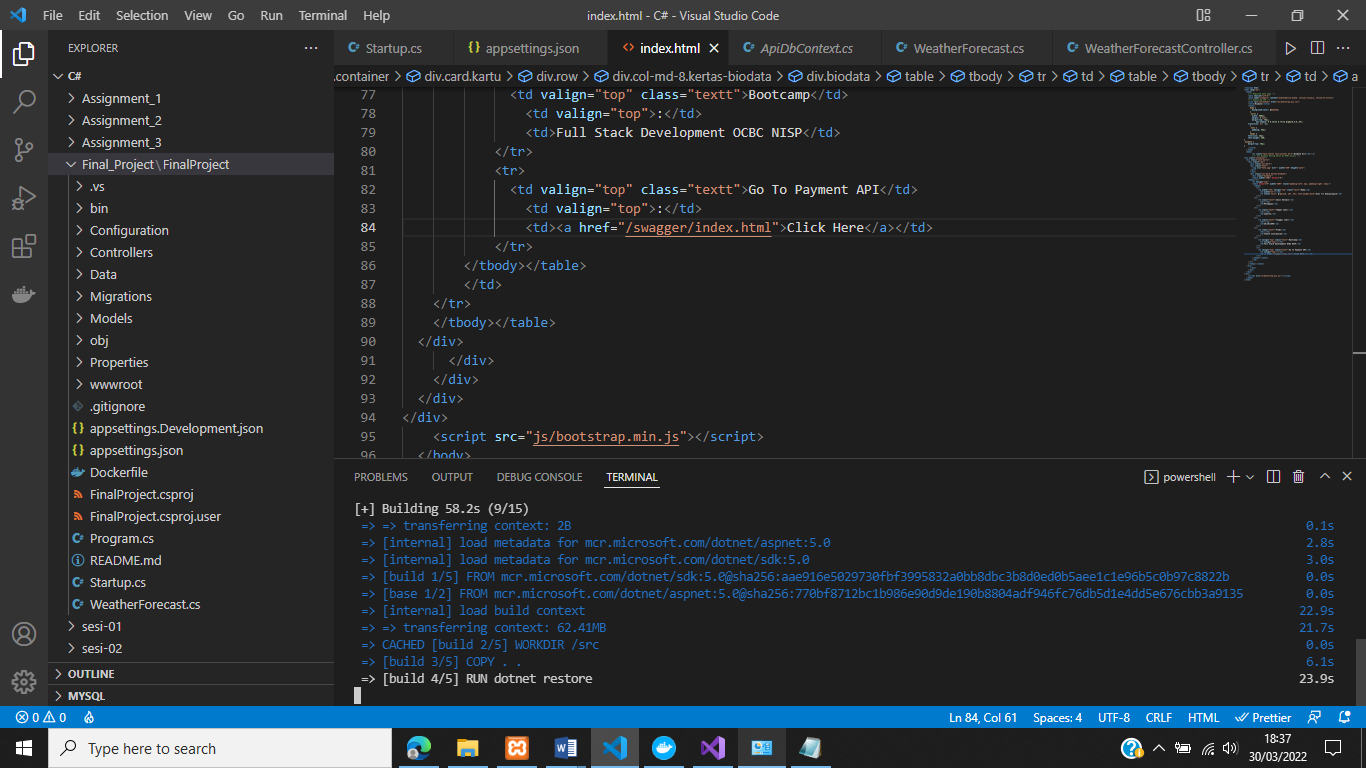
* Lakukan build dengan perintah dotnet build pada terminal

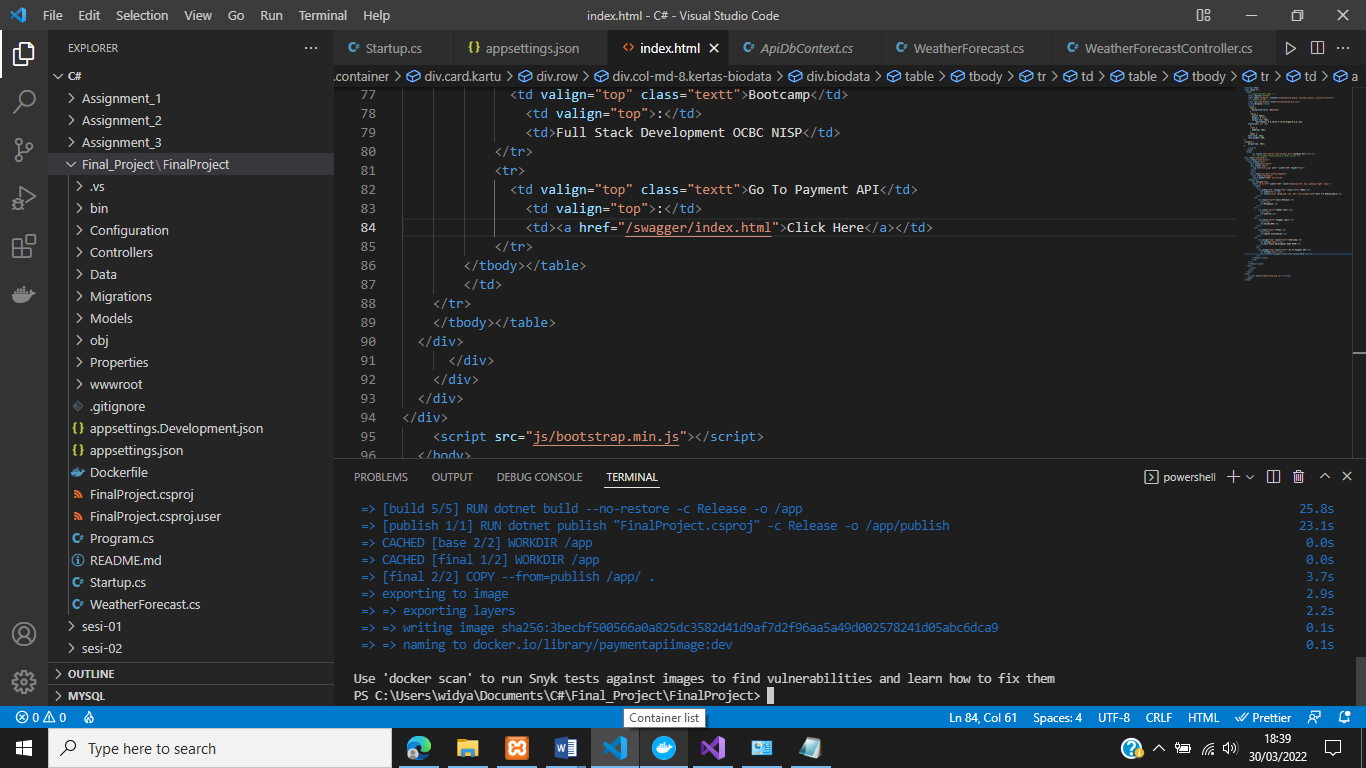


* Lakukan Release dengan perintah ***dotnet publish –c release***

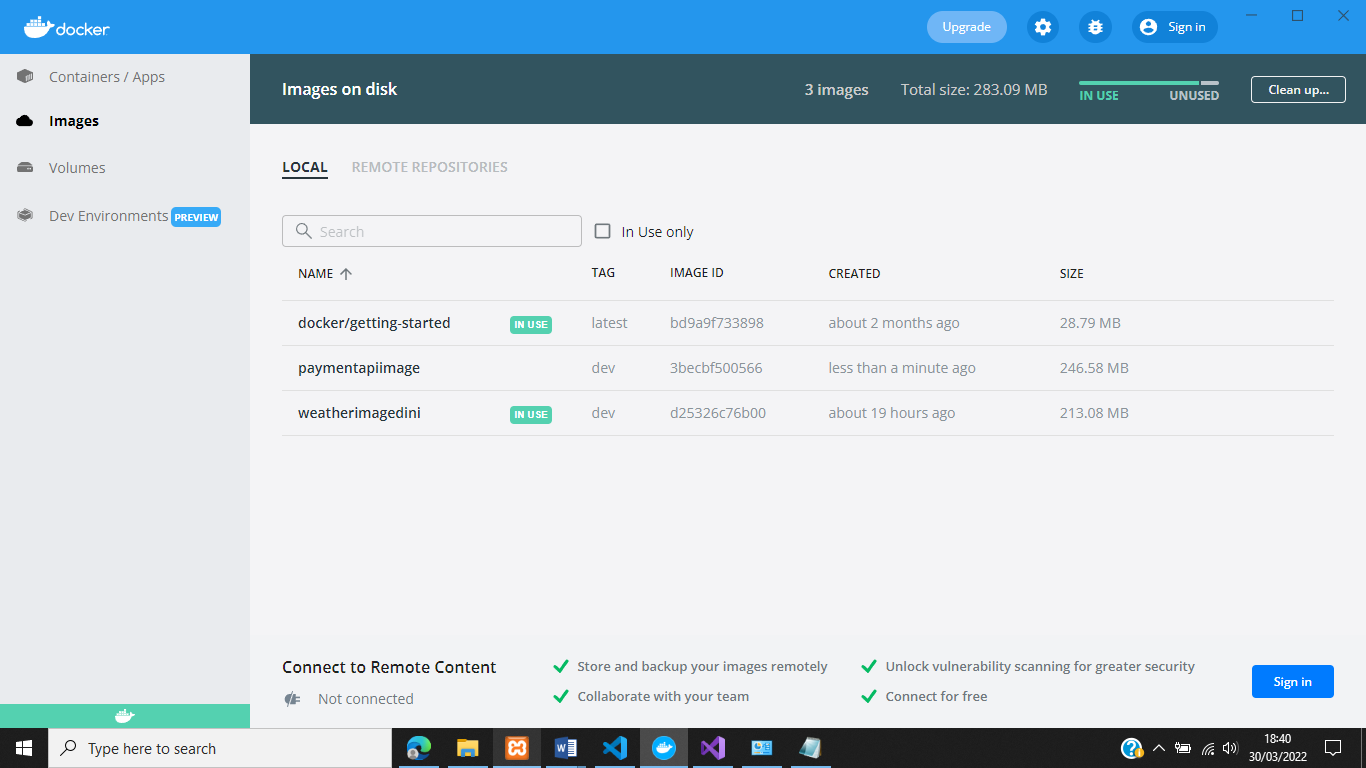


* Pembuatan docker image dengan syntax ***docker build –t paymentapiimage:dev .***

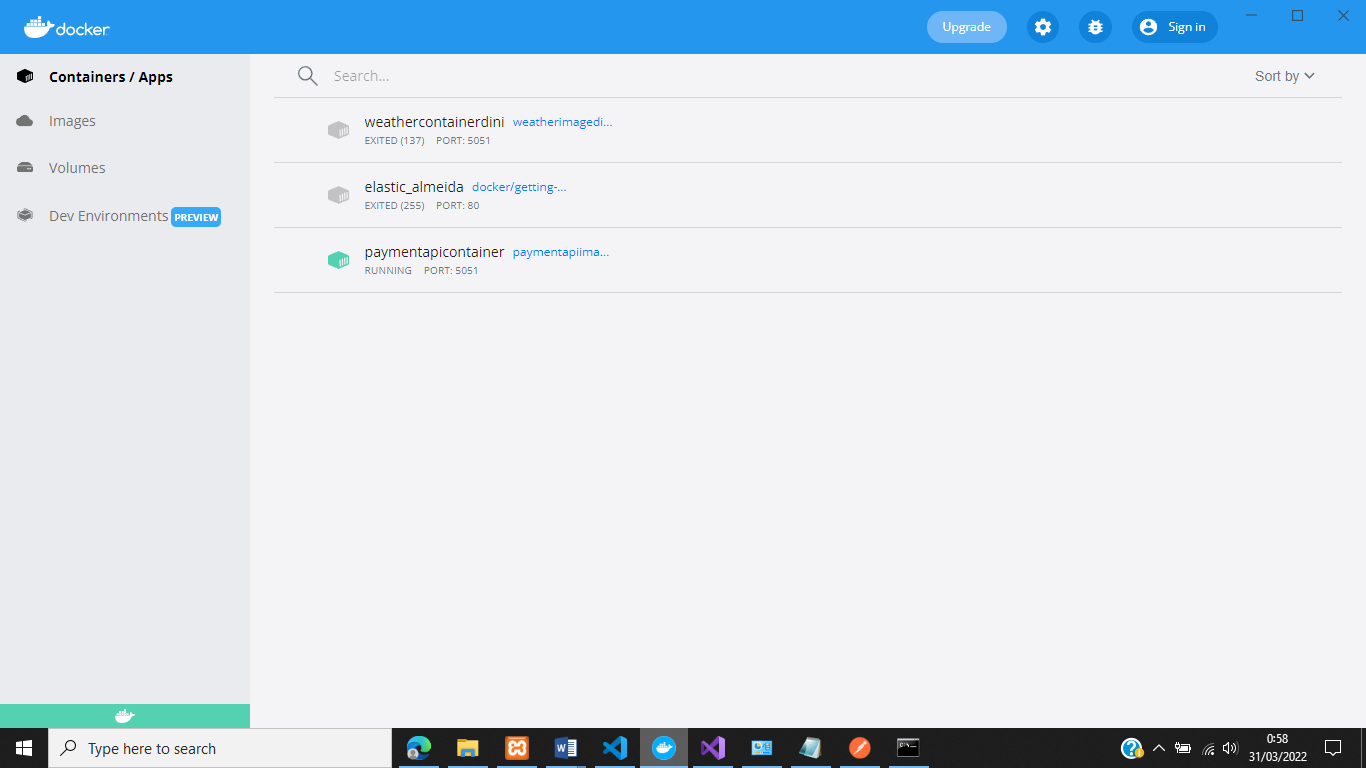




((Apabila berhasil, akan muncul paymentapiimage pada Docker))

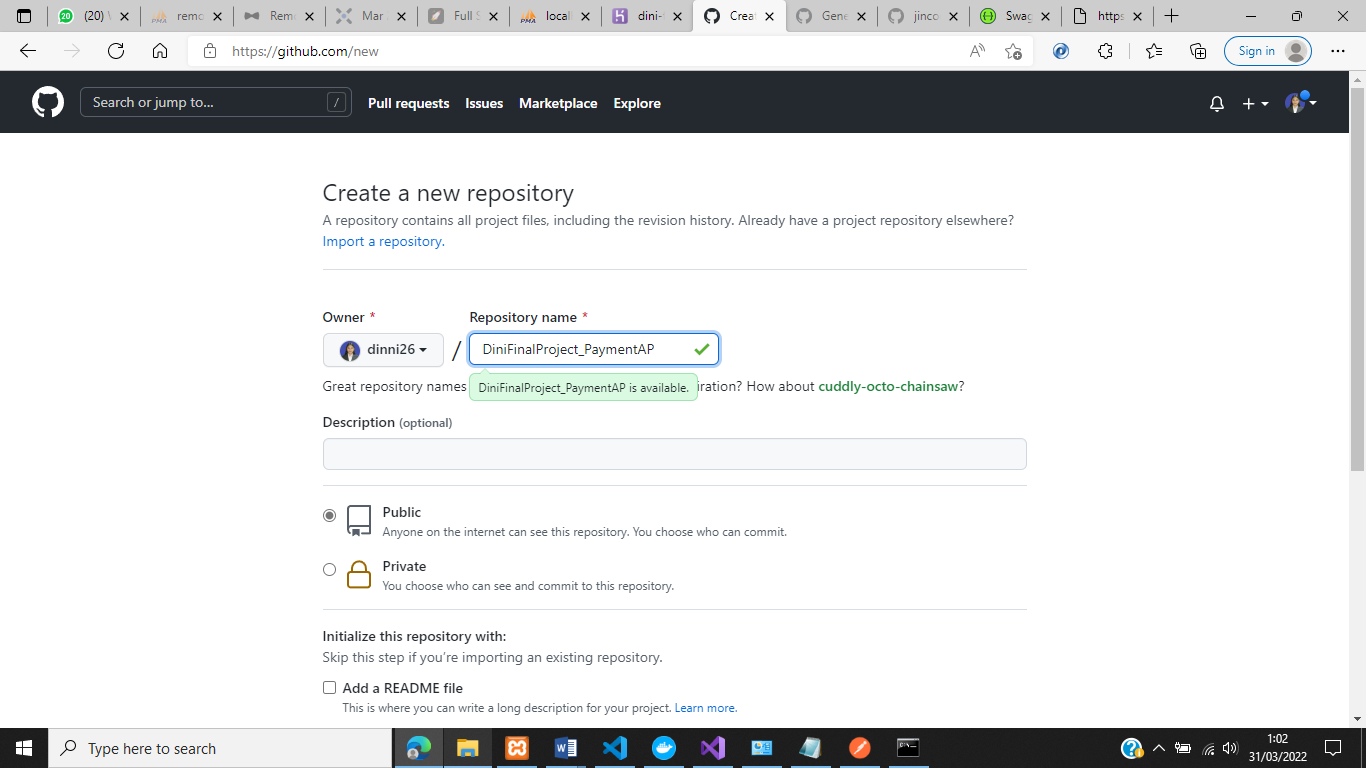


* Membuat container dengan menjalankan perintah ***docker run –d –p 5051:80 –name paymentapicontainer paymentapiimage:dev*** (aplikasi akan berjalan pada port 5051)

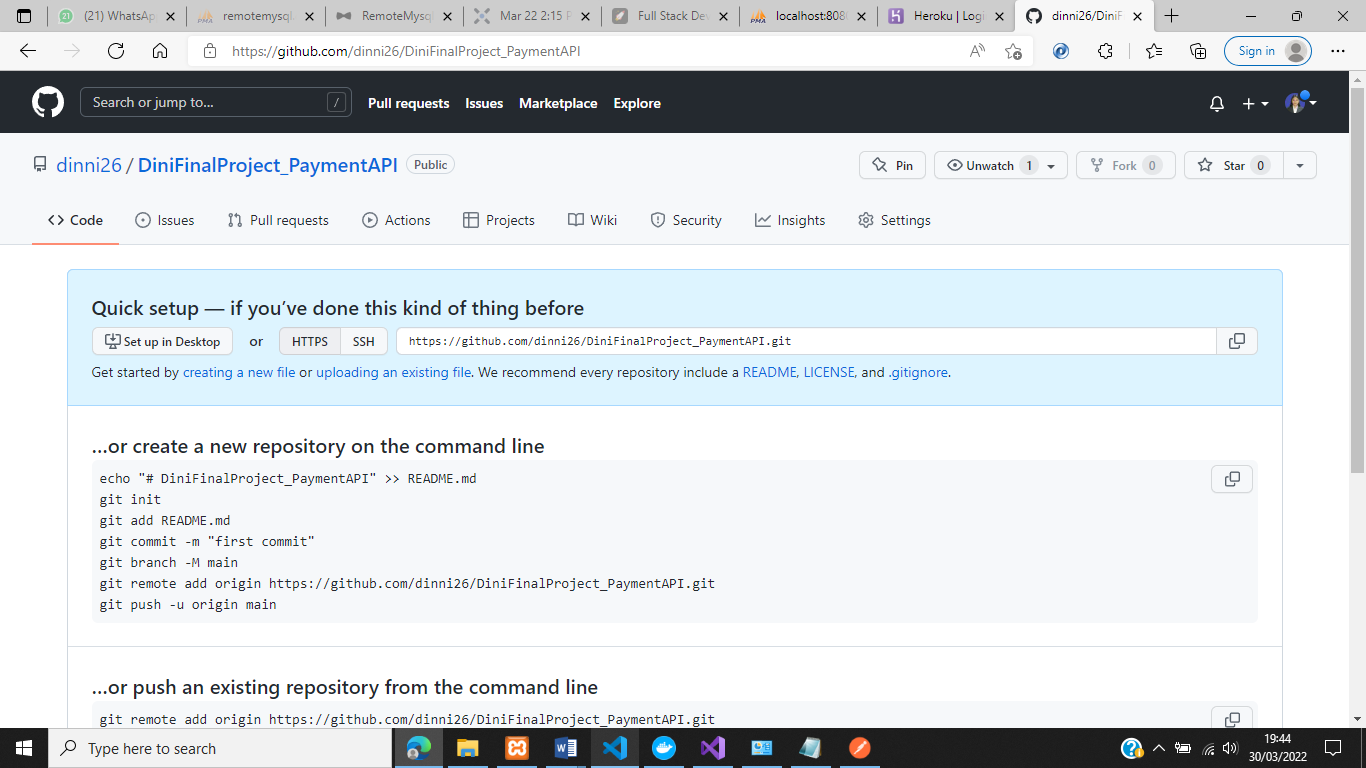


**PEMBUATAN REPO BARU DI GITHUB**

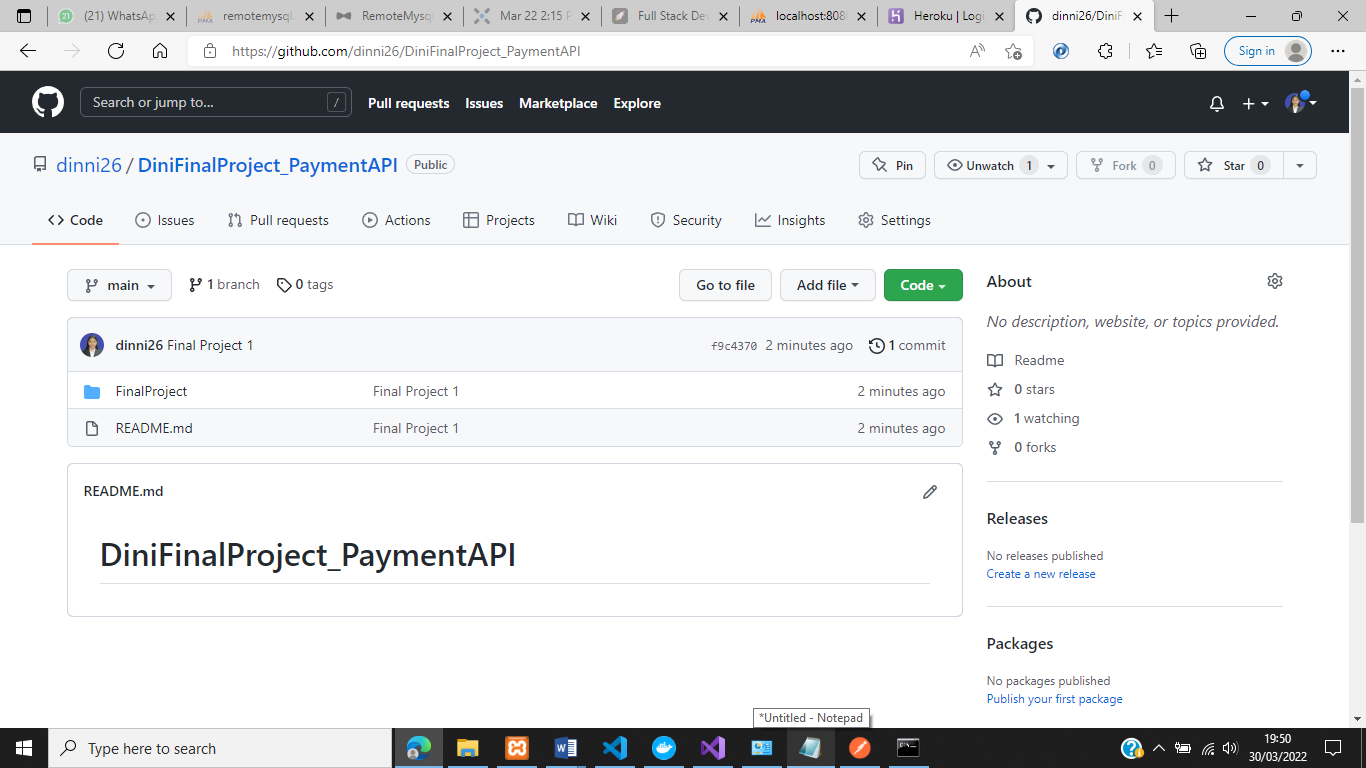
* Create new repo lalu beri nama repo yang baru dibuat



* Setelah terbuat, akan muncul code yang akan kita jalankan di cmd

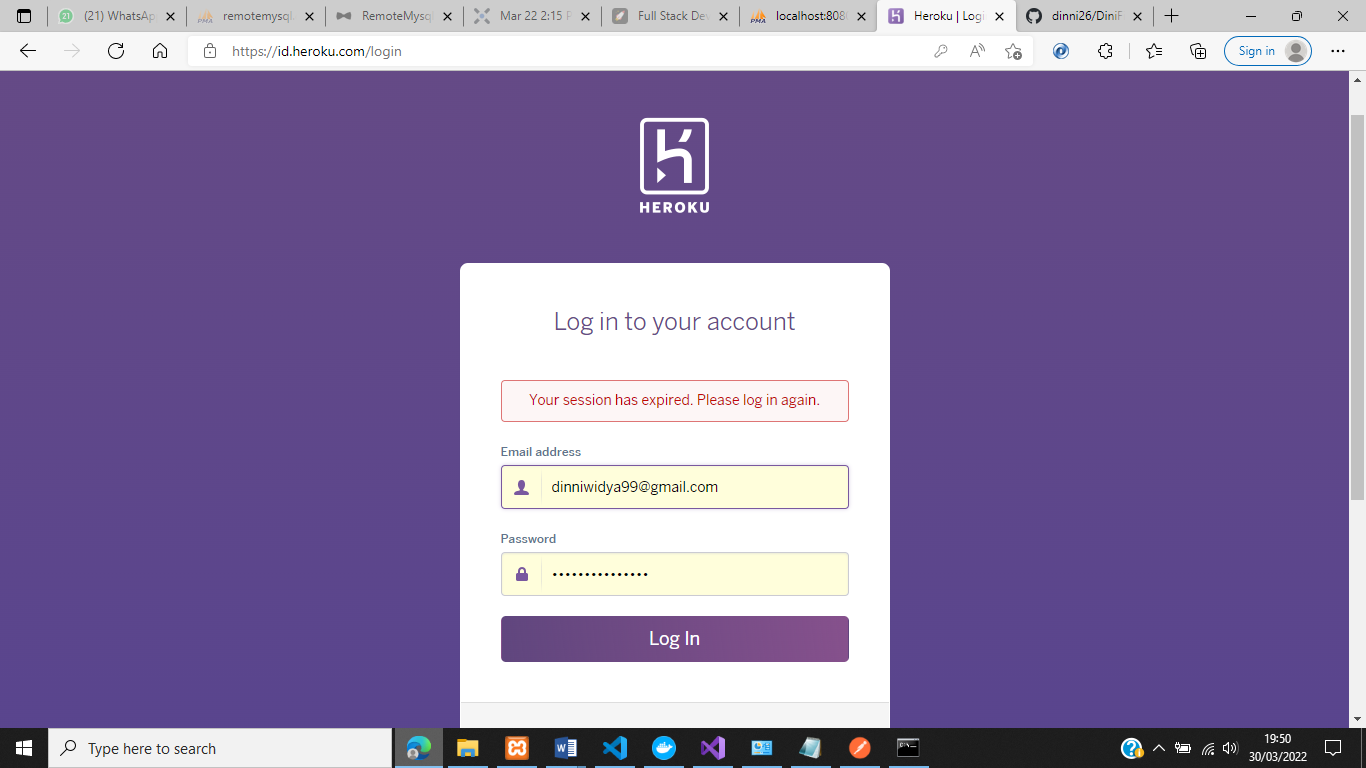


* Setelah dijalankan melalui cmd, makan repository baru akan tampil folder FinalProject yang sudah dibuat

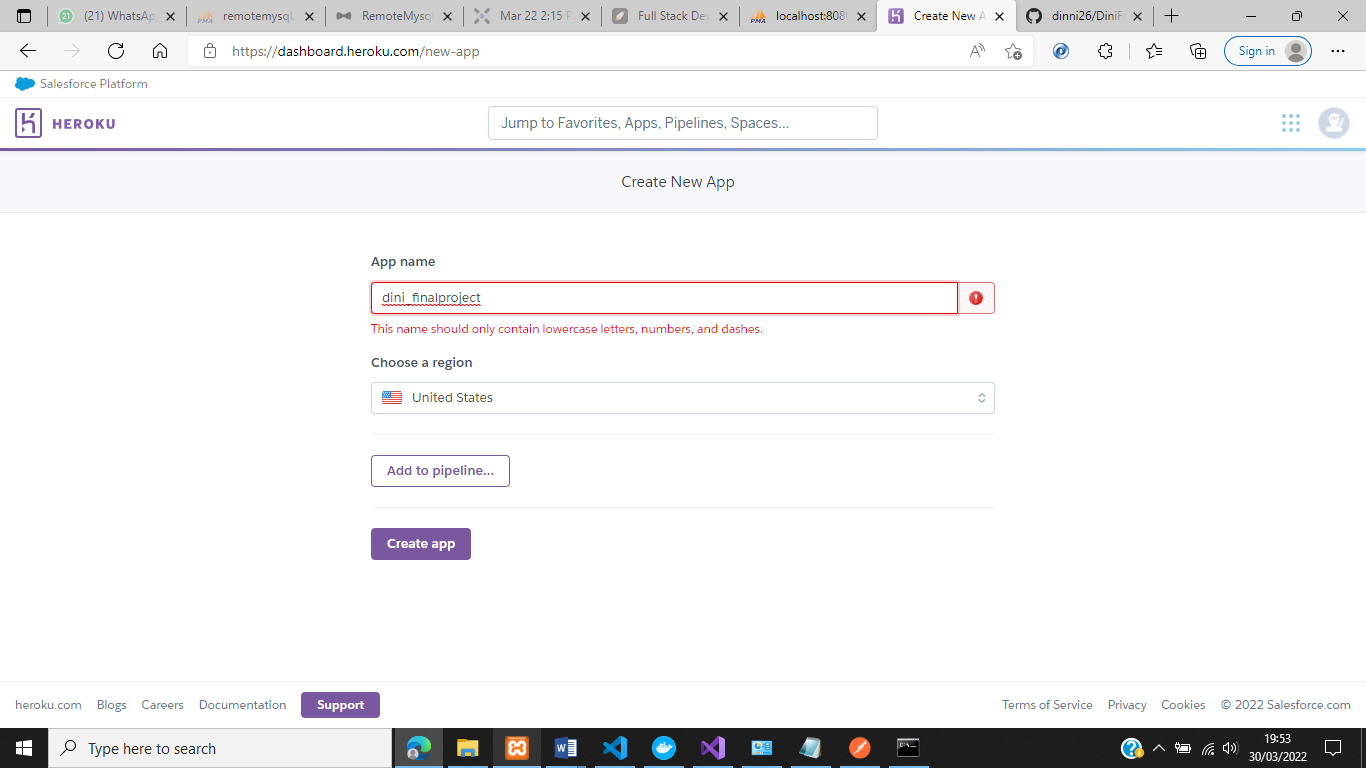


**HEROKU**

* Login Akun Heroku



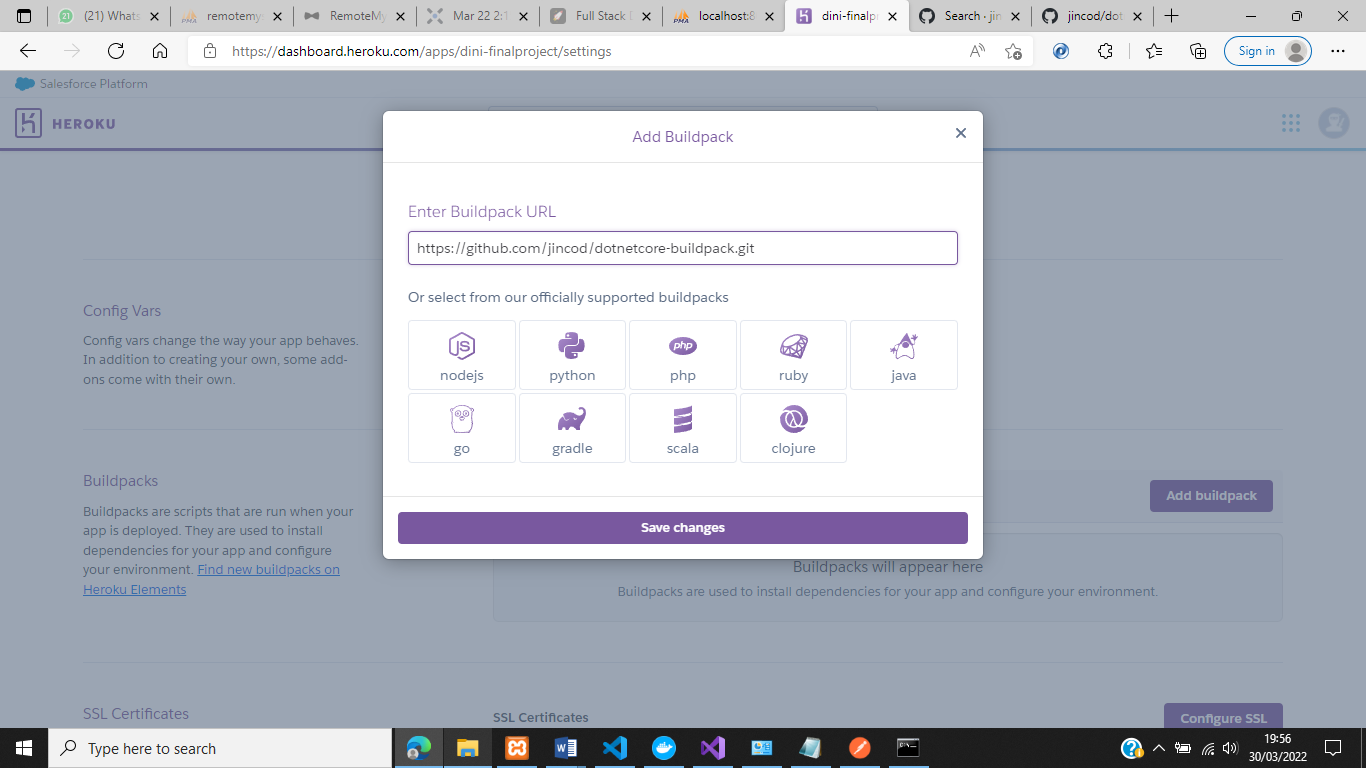
* Create New App lalu masukan nama app yang akan dibuat lalu create app.



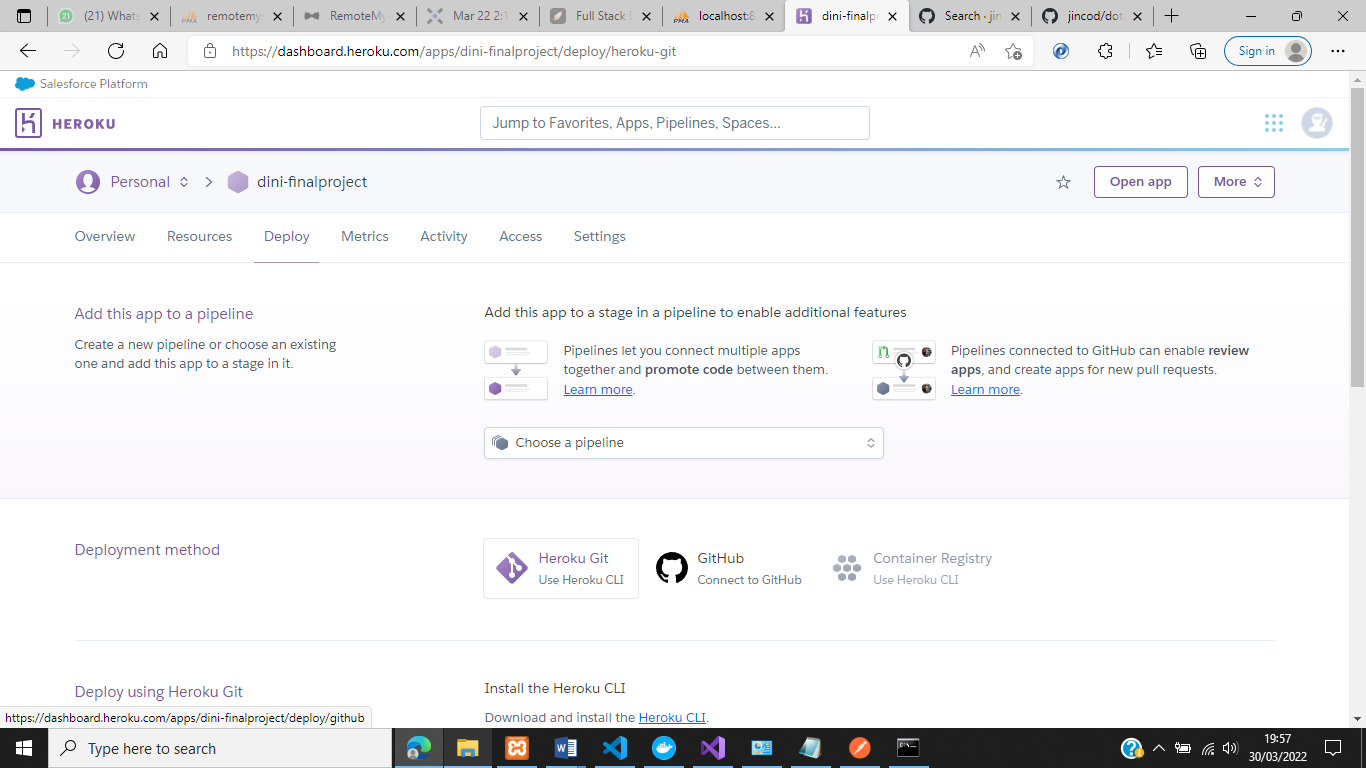
* Setelah Create App > Settings > Add Buildpack > Copy Link Jincod dari github



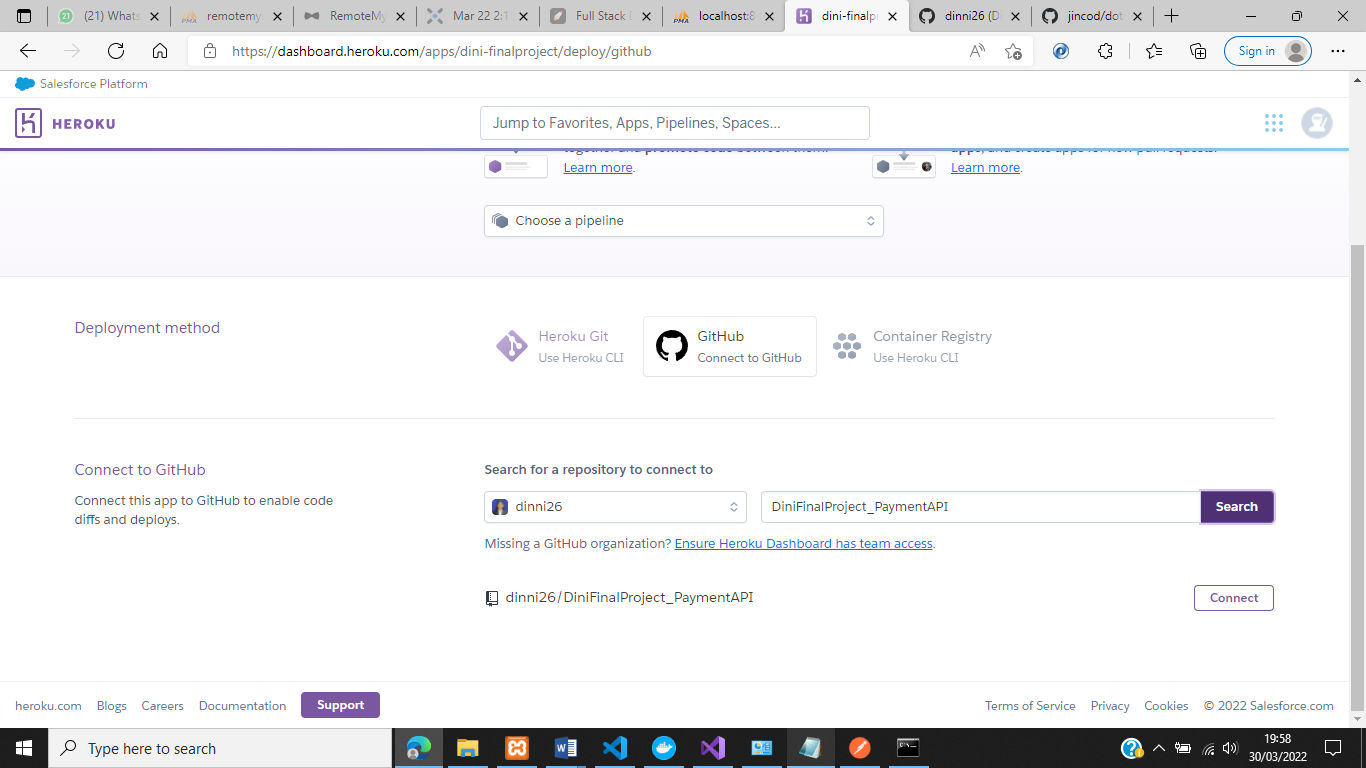
* Paste link Jincod Kedalam Buildpack URL



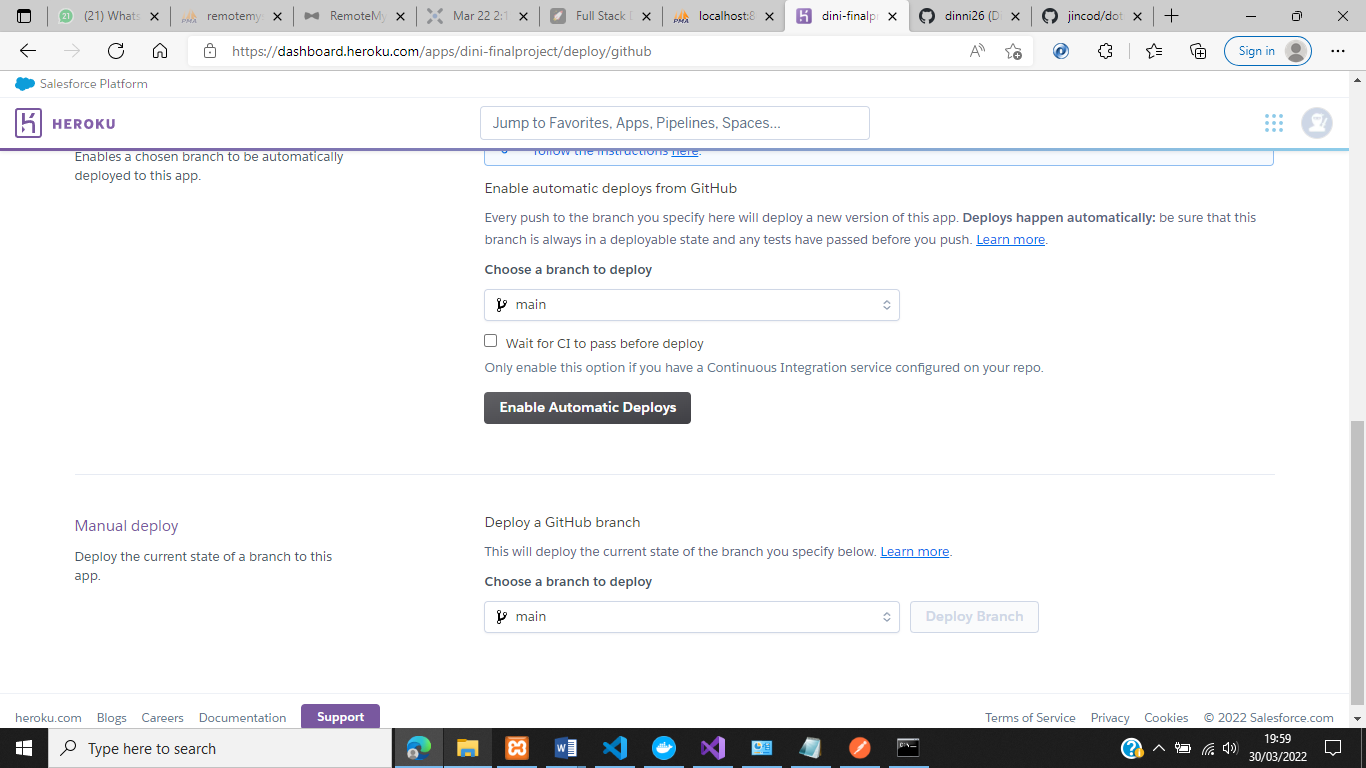
* Lalu Ke Deploy > Connect to Github



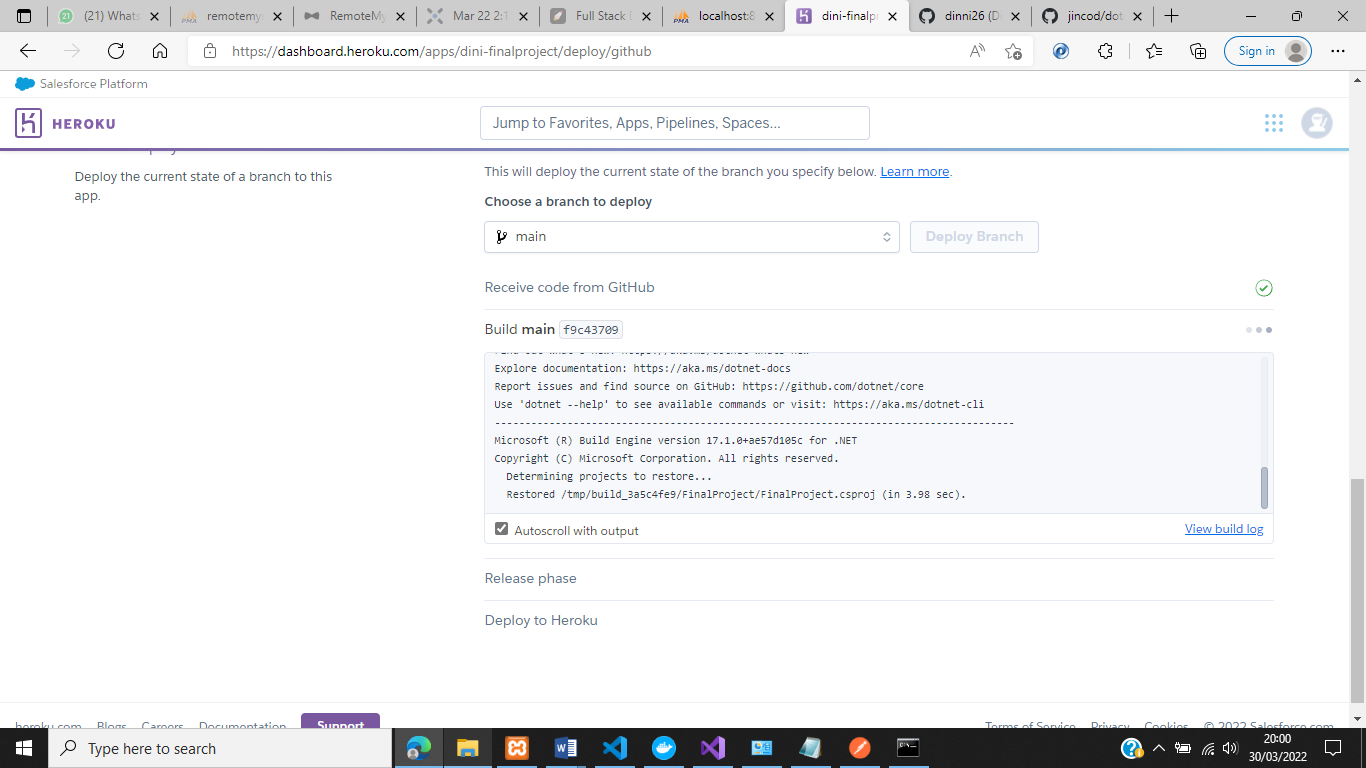
* Masukan Repo Github lalu Connect



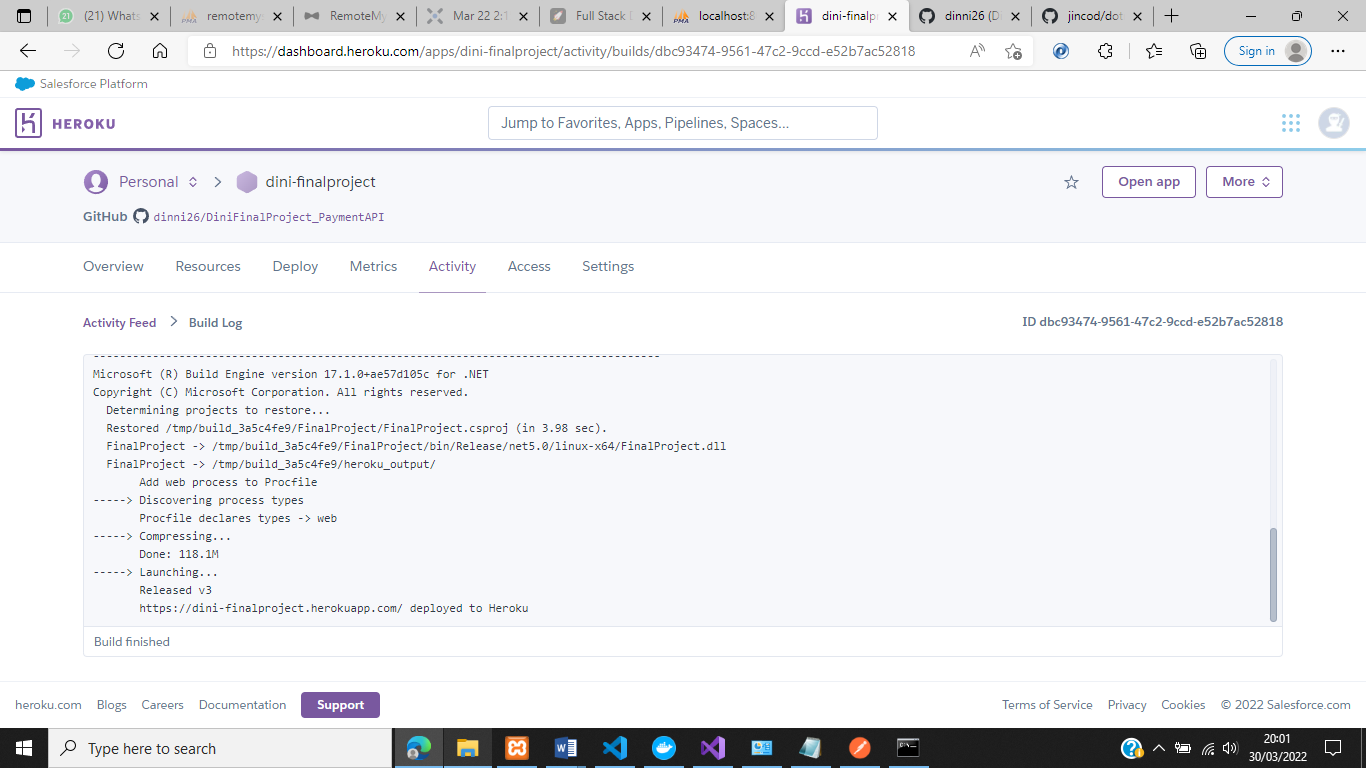
* Setelah itu scroll ke bawah lalu klik “Deploy Branch”



* Tunggu sampai proses selesai



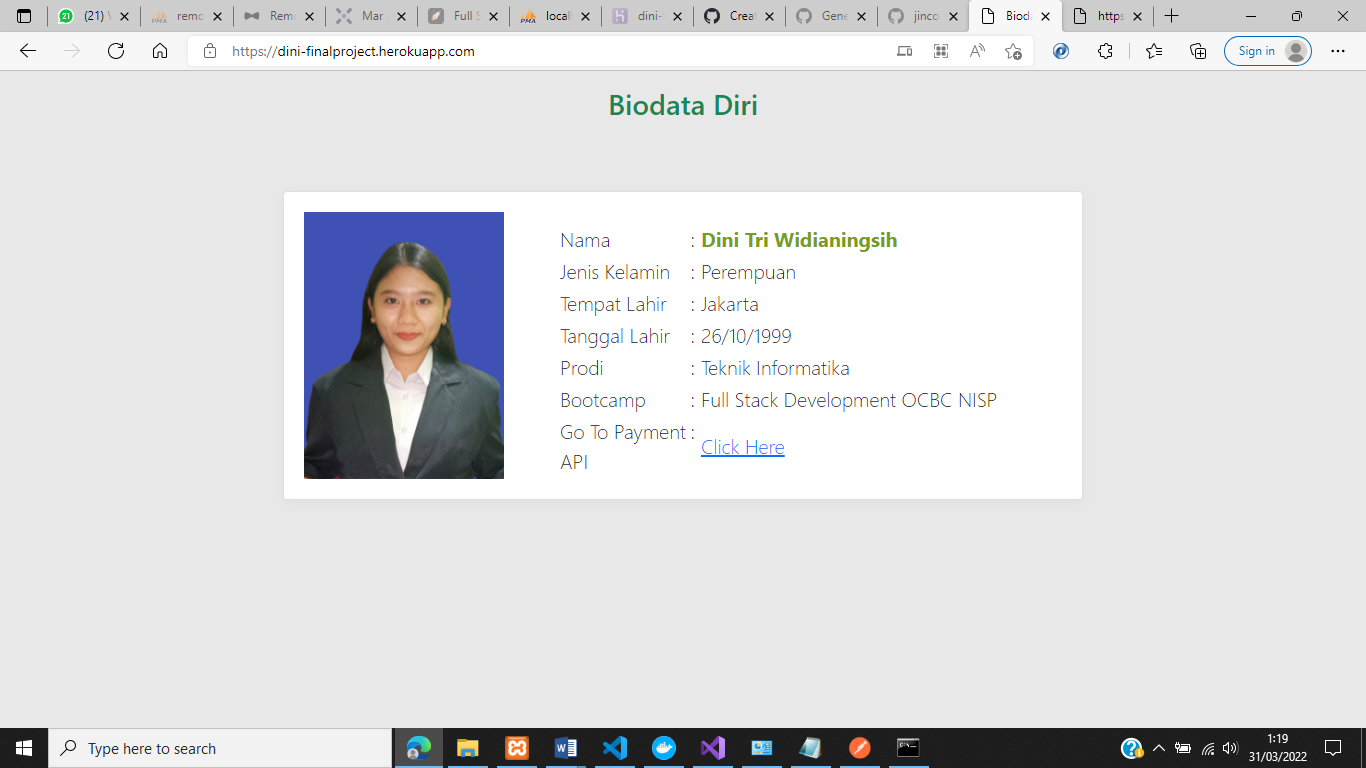
* Setelah selesai maka akan ada link heroku yang nantinya akan kita jalankan untuk penggunaan ProjectAkhir



* Salin Link > Browser untuk menjalankannya.

Link : [Biodata (dini-finalproject.herokuapp.com)](https://dini-finalproject.herokuapp.com/)

***Tampilan Awal***



***((Lanjut ke panduan penggunaan aplikasi))***