

## **BAB 1. HELLO WORLD API, MODEL & CONTROLLER**

### **1.1 CAPAIAN PEMBELAJARAN**

Kode CPL	: CPL08
Nama CPL	: Menjelaskan definisi dan ruang lingkup API
Kode CPMK	: 1
Nama CPMK	: Mahasiswa mampu menjelaskan definisi dan ruang lingkup API
Nama Sub-CPMK	: Menjelaskan definisi dan ruang lingkup API

### **1.2 TUJUAN PERCOBAAN**

Mahasiswa mampu menjelaskan definisi dan ruang lingkup API.

### **1.3 TINJAUAN PUSTAKA**

API adalah antarmuka yang dihadirkan program perangkat lunak ke program lain, ke manusia, dan, dalam kasus API web, ke dunia melalui internet. Desain API mengabaikan banyak hal tentang program di baliknya—model bisnis, fitur produk, dan bug yang terkadang terjadi. Meskipun API dirancang untuk bekerja dengan program lain, sebagian besar API dimaksudkan untuk dipahami dan digunakan oleh manusia yang menulis program lain tersebut.

API adalah elemen penyusun yang memungkinkan interoperabilitas untuk platform bisnis utama di web. API adalah cara identitas dibuat dan dipelihara di seluruh akun perangkat lunak cloud, mulai dari alamat email perusahaan hingga perangkat lunak desain kolaboratif hingga aplikasi web yang membantu Anda memesan pengiriman pizza. API adalah cara data prakiraan cuaca dibagikan dari sumber terpercaya seperti Layanan Cuaca Nasional ke ratusan aplikasi perangkat lunak yang berspesialisasi dalam penyajiannya. API memproses kartu kredit Anda dan memungkinkan perusahaan mengumpulkan uang Anda dengan lancar tanpa mengkhawatirkan hal-hal kecil dari teknologi keuangan serta undang-undang dan peraturan terkait.

API muncul dari kebutuhan untuk bertukar informasi dengan penyedia data yang mampu memecahkan masalah tertentu, sehingga orang-orang di perusahaan lain tidak perlu menghabiskan waktu untuk menyelesaikan sendiri masalah tersebut. Misalnya, Anda mungkin ingin menyematkan peta interaktif pada halaman web tanpa menciptakan kembali

Google Maps. Anda mungkin ingin pengguna masuk tanpa harus menemukan kembali Login Facebook. Atau, Anda mungkin ingin membuat chatbot yang sesekali berinteraksi dengan pengguna tanpa harus membangun sistem pesan real-time.

Dalam semua kasus ini, fitur dan produk tambahan dibuat menggunakan data atau interaksi dari platform khusus. API memungkinkan bisnis mengembangkan produk unik dengan cepat. Daripada melakukan inovasi baru, startup dapat membedakan penawaran produk mereka sambil memanfaatkan teknologi yang ada dan memanfaatkan ekosistem lain.

## 1.4 ALAT DAN BAHAN

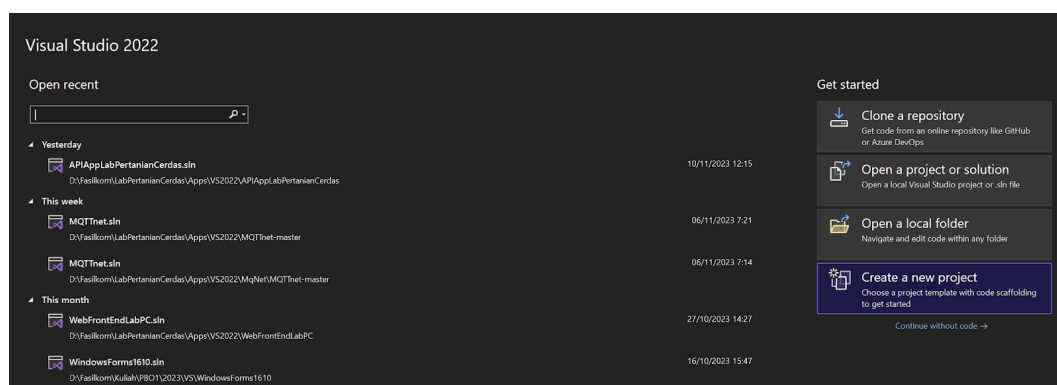
- Kebutuhan peralatan
  1. Komputer PC/Notebook
  2. Visual Studio 2022
  3. Postman
- Kebutuhan bahan praktikum
  1. -

## 1.5 PROSEDUR KERJA

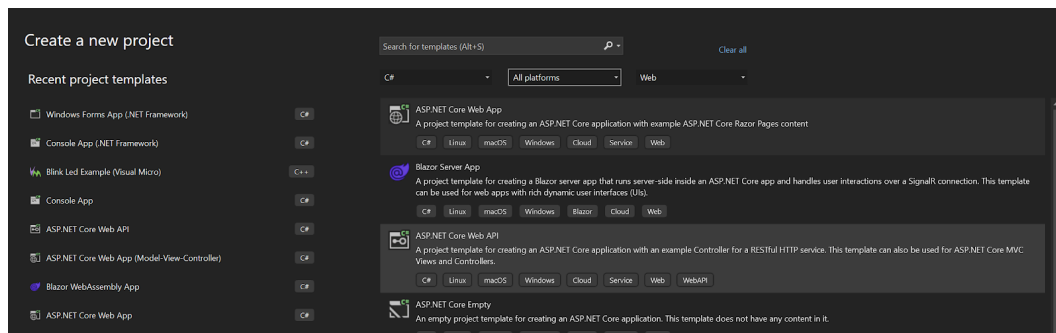
### Percobaan 1: Hello world API

Langkah-langkah percobaan:

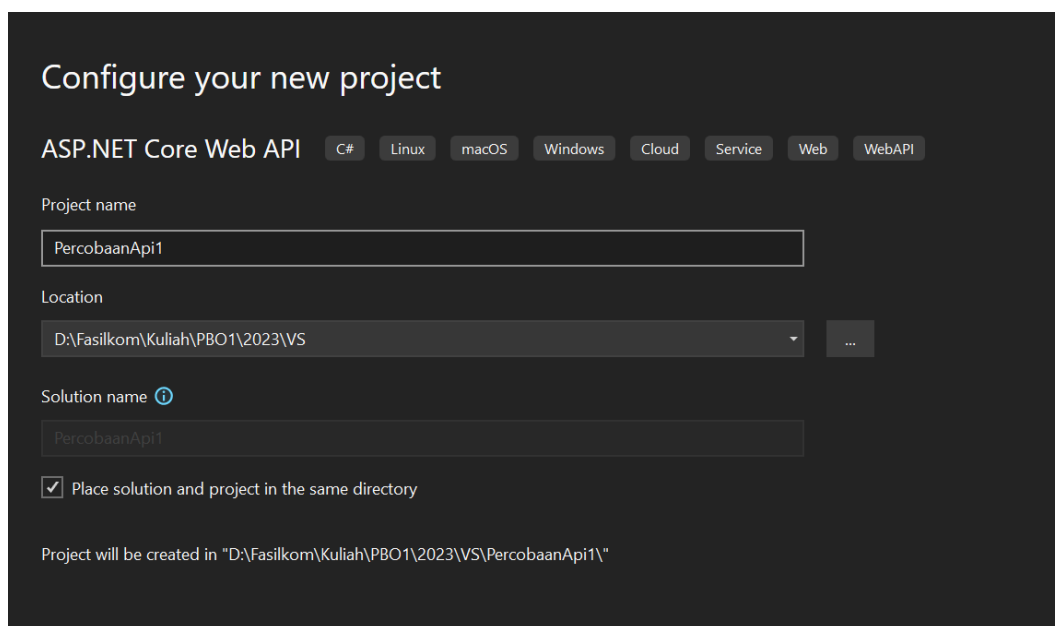
1. Buka Visual Studio 2022
2. Buatlah sebuah proyek baru



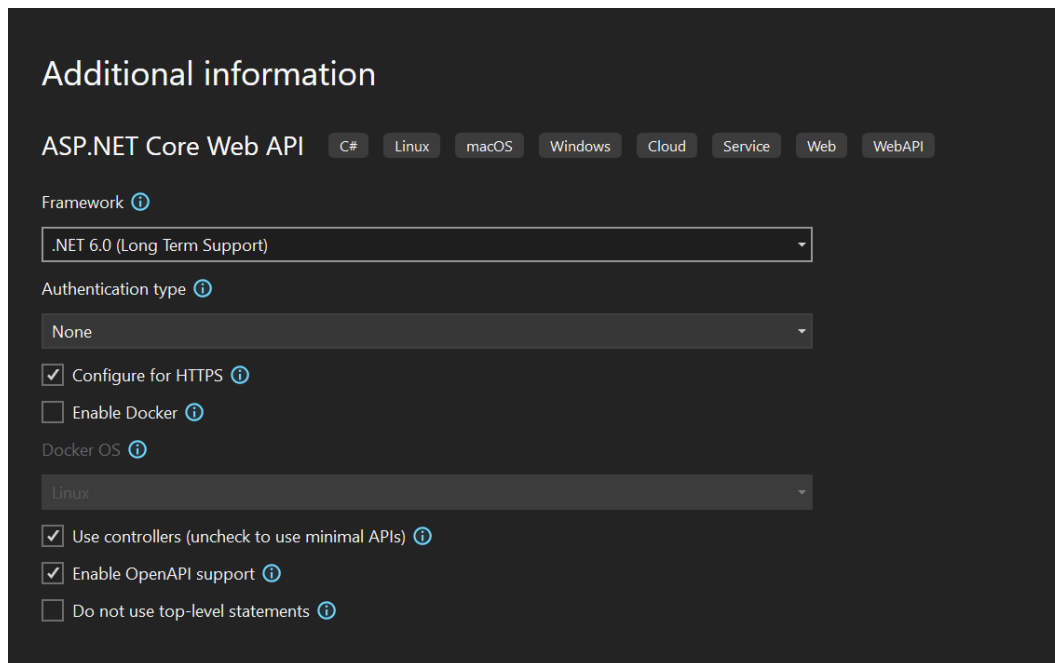
3. Pilih template ASP.NET Core Web API dan pilih Berikutnya.



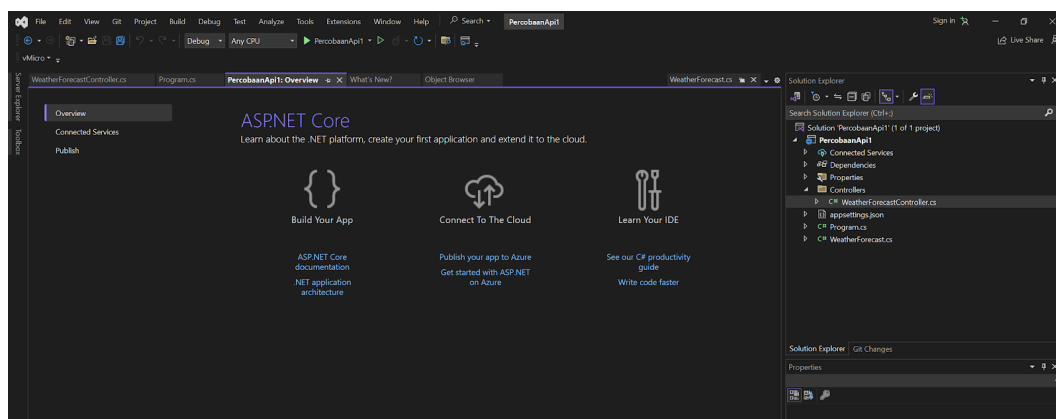
4. Simpan proyek tersebut dengan nama “PercobaanApi1”



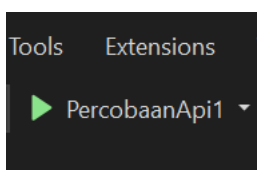
5. Pilih .NET 6.0 (Long Term Support) dan klik Create



6. Struktur proyek akan tampak seperti Gambar di bawah ini.



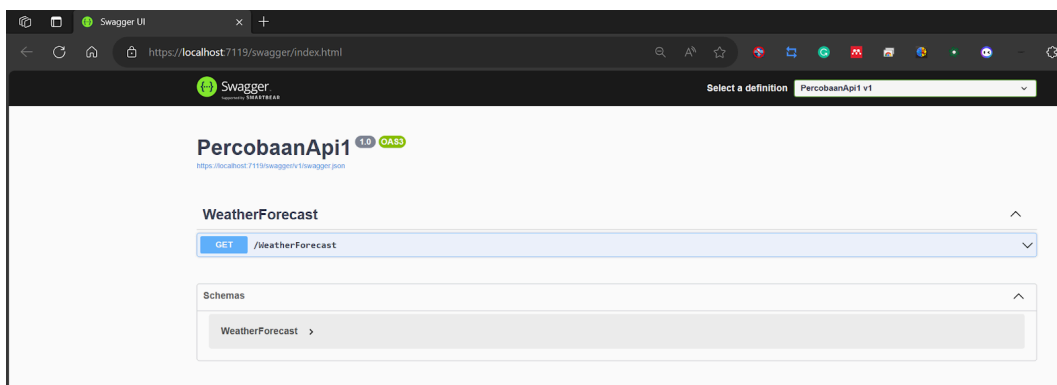
Terdapat 3 file csharp (.cs) yaitu Program.cs, WeatherForecast.cs, dan WeatherForecastController.cs. File Program.cs memuat program utama yang akan menggunakan dan membuat objek **builder** (WebApplicationBuilder) dan objek **app** (WebApplication). Perhatikan beberapa konfigurasi pada builder dan app hingga baris **app.Run()** untuk menjalankan aplikasi. File WeatherForecast.cs adalah contoh file model dan file WeatherForecastController.cs adalah contoh controller. Jalankan program dengan menekan tombol run.



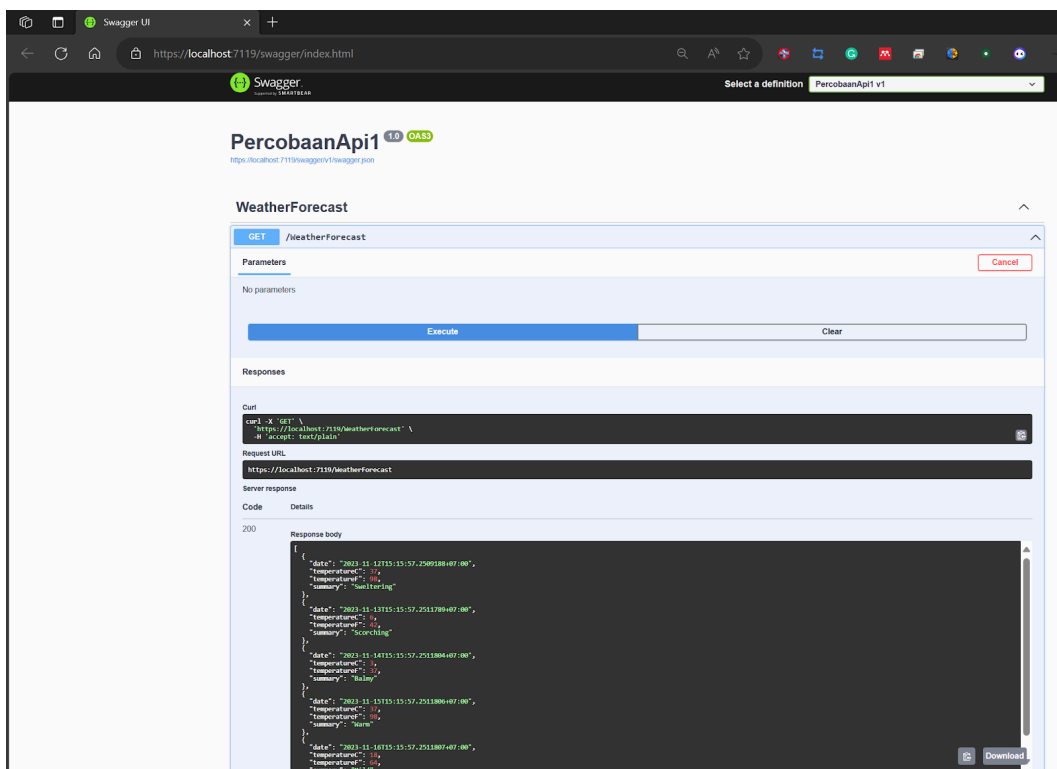
Dan akan menjalankan terminal dan web browser secara bersamaan

```
D:\Fasilkom\Kuliah\PBO1\202  X  +  v

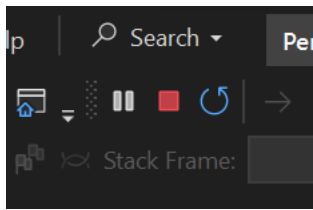
info: Microsoft.Hosting.Lifetime[14]
Now listening on: https://localhost:7119
info: Microsoft.Hosting.Lifetime[14]
Now listening on: http://localhost:5116
info: Microsoft.Hosting.Lifetime[0]
Application started. Press Ctrl+C to shut down.
info: Microsoft.Hosting.Lifetime[0]
Hosting environment: Development
info: Microsoft.Hosting.Lifetime[0]
Content root path: D:\Fasilkom\Kuliah\PBO1\2023\VS\PercobaanApi1\
|
```



Tekan tombol “GET”, kemudian “Try it out”, kemudian “Execute” maka akan menampilkan data JSON Weather Forecast sebagai berikut.



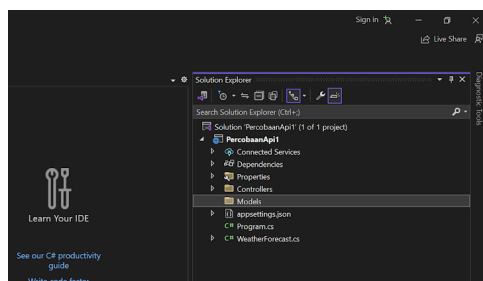
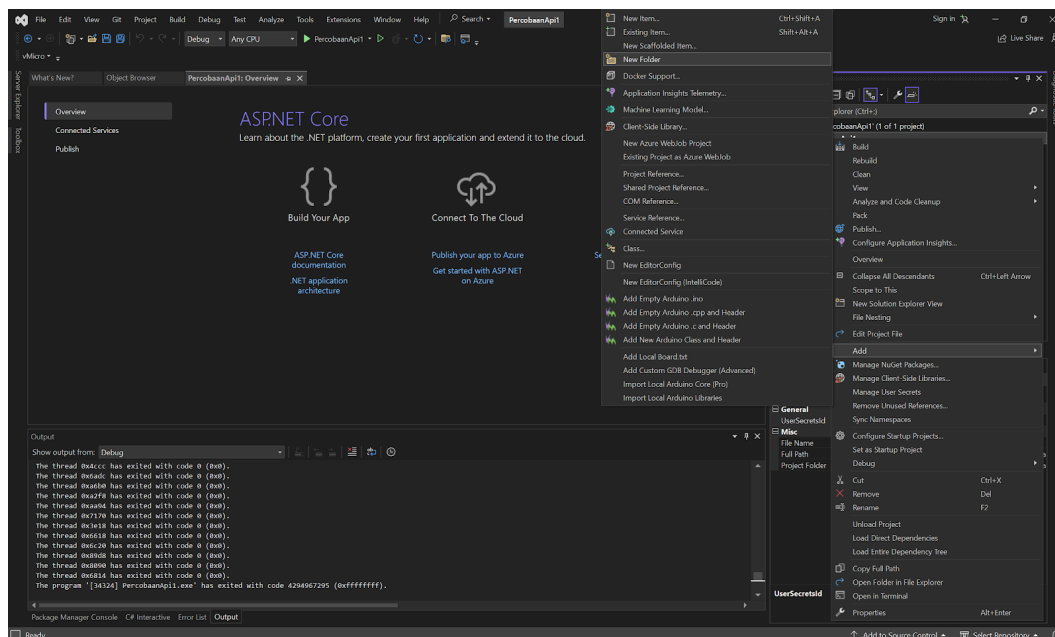
Hentikan aplikasi dengan menekan tombol stop debugging



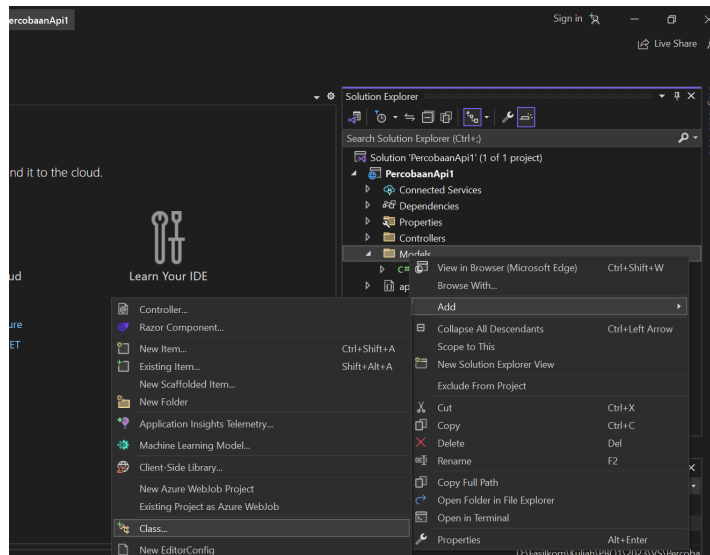
## Percobaan 2: Membuat model dan kontroler baru

### 2.1. Membuat Model

1. Tambahkan folder Models untuk menyimpan file-file model dengan klik kanan pada project, kemudian pilih “Add”, kemudian “New Folder” dan beri nama “Models”



Tambahkan kelas baru dengan klik kanan pada folder Models, pilih “Add”, lalu “Class” dan beri nama kelas baru dengan “Person.cs”.



Tuliskan kode program di bawah ini pada file “Person.cs”. Amatilah struktur dan sintak kode program.

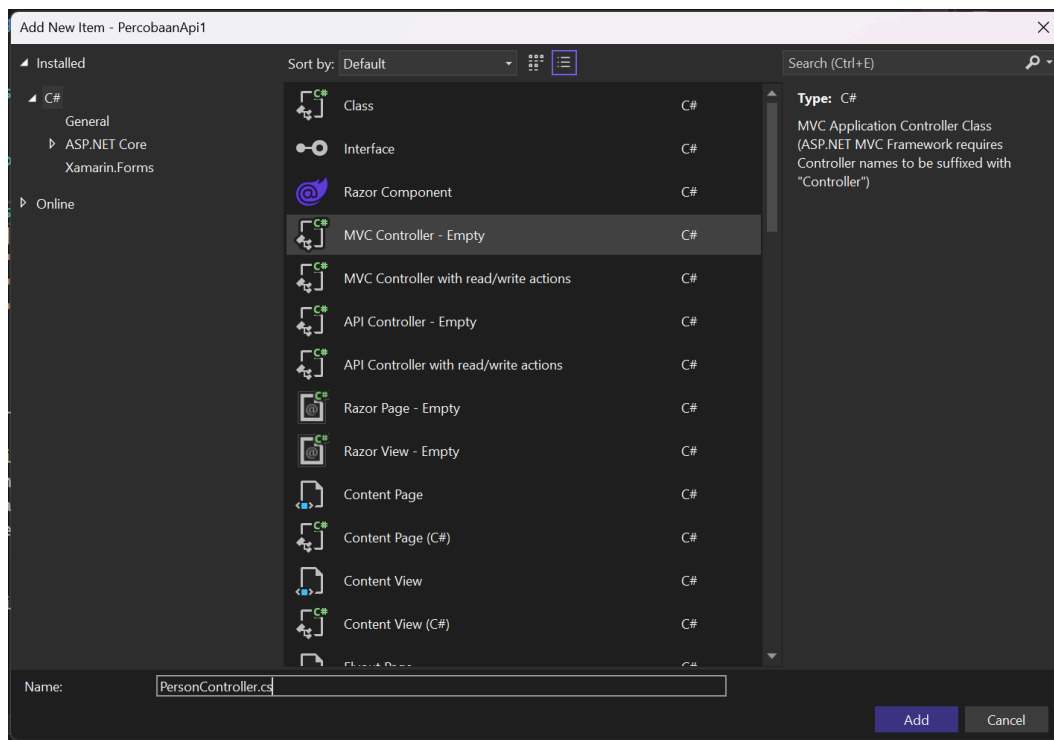
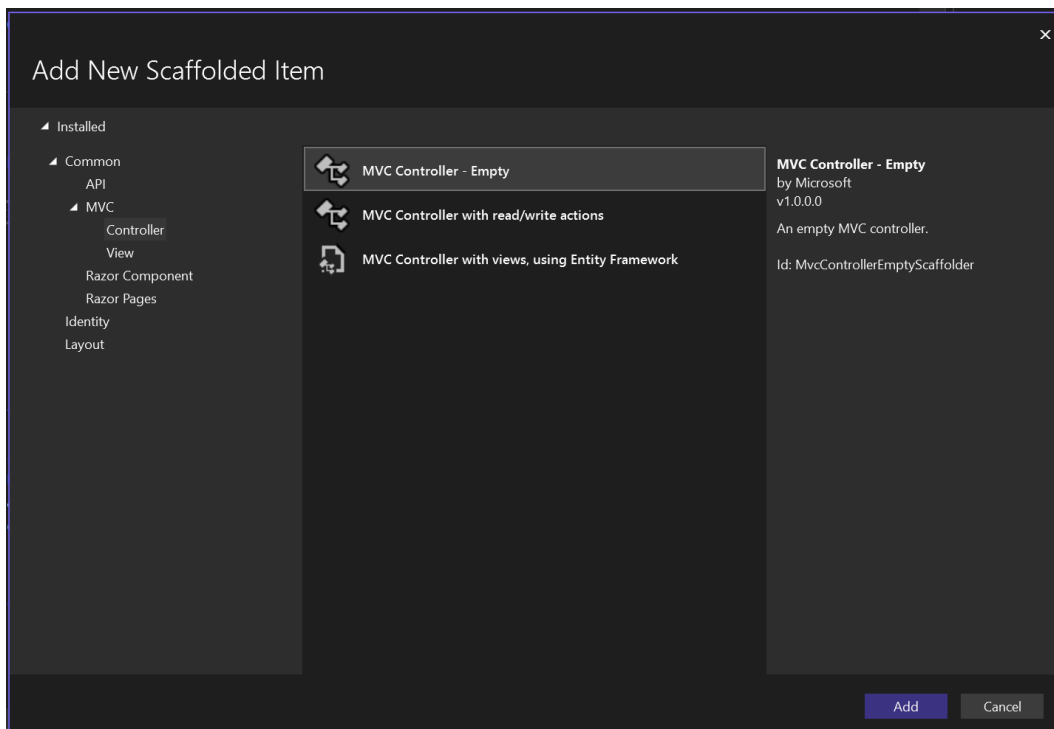
```
namespace PercobaanApi1.Models
{
    public class Person
    {
        public int id_person { get; set; }
        public string nama { get; set; }
        public string alamat { get; set; }
        public string email { get; set; }
    }
}
```

Pada direktori Models, tambahkan file “PersonContext.cs” sebagai berikut:

```
namespace PercobaanApi1.Models
{
    public class PersonContext
    {
        public List<Person> ListPerson()
        {
            List<Person> list1 = new List<Person>();
            Object[,] arrayPerson = new Object[,]{
                {1, "Budi", "Jember", "budi@gmail.com" },
                {2, "Iwan", "Banyuwangi", "iwan@gmail.com" },
                {3, "Wati", "Lumajang", "wati@gmail.com" }
            };
            for (int i = 0; i < arrayPerson.GetLength(0); i++)
            {
                list1.Add(new Person()
                {
                    id_person = int.Parse(arrayPerson[i, 0].ToString()),
                    nama = arrayPerson[i, 1].ToString(),
                    alamat = arrayPerson[i, 2].ToString(),
                    email = arrayPerson[i, 3].ToString()
                });
            }
            return list1;
        }
    }
}
```

## 2.2. Membuat Controller

Membuat kelas baru dengan nama “PersonController.cs” di folder “Controllers” dengan klik kanan pada folder Controller, pilih “Add”, kemudian pilih “Controller”, kemudian pilih “MVC controller-empty” sebagai berikut:



Kemudian tambahkan kode program di bawah ini di kelas “PersonController.cs”. Perhatikan cara membuat routing url API pada decorator method “[HttpPost(“api/person”)]”.



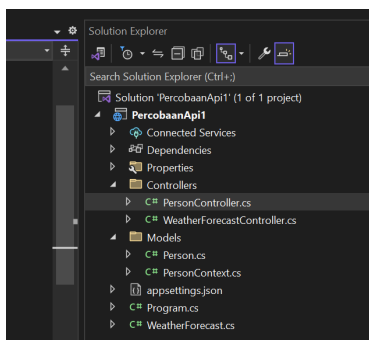
```

using Microsoft.AspNetCore.Authorization;
using Microsoft.AspNetCore.Mvc;
using PercobaanApi1.Models;

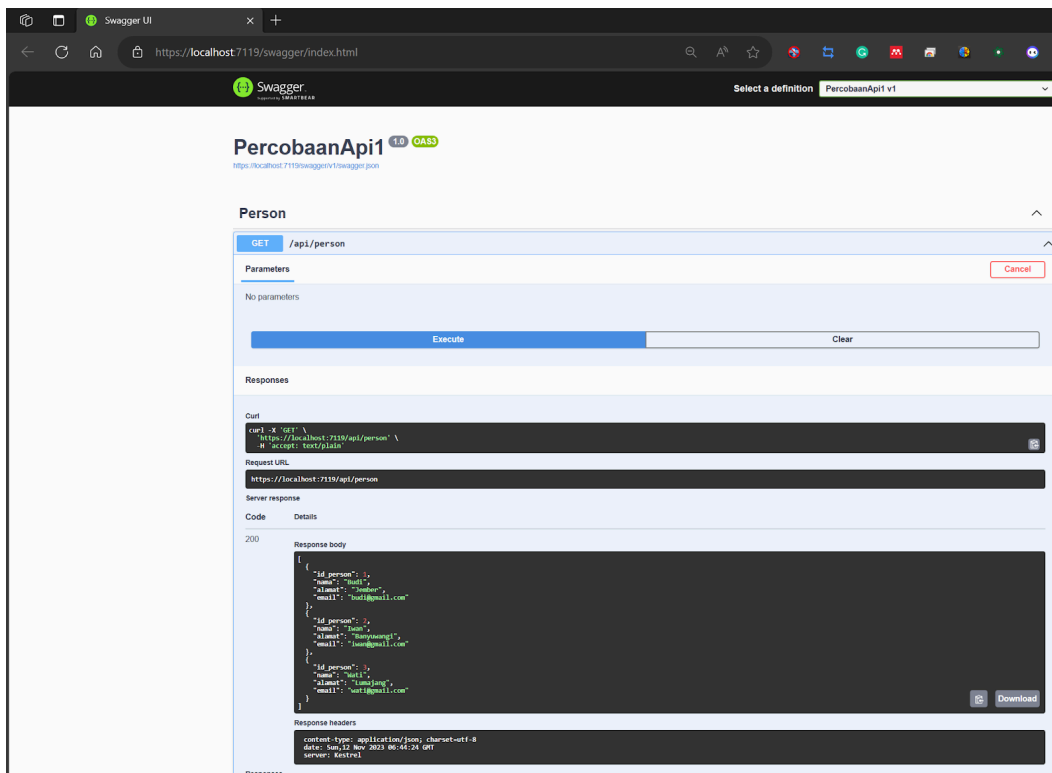
namespace PercobaanApi1.Controllers
{
    public class PersonController : Controller
    {
        public IActionResult Index()
        {
            return View();
        }
        [HttpPost("api/person")]
        public ActionResult<Person> ListPerson()
        {
            PersonContext context = new PersonContext();
            List<Person> ListPerson = context.ListPerson();
            return Ok(ListPerson);
        }
    }
}

```

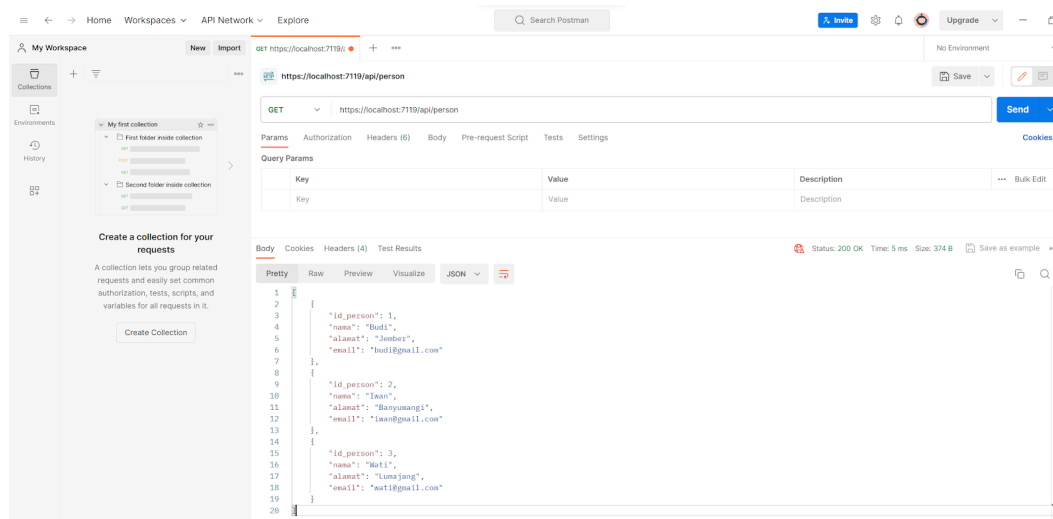
Tampilan struktur proyek akan menjadi sebagai berikut:



Jalankan aplikasi pada visual studio sehingga membuka web browser dan Swagger. Pada Swagger akan muncul API untuk Person dan ketika dipilih akan menampilkan:



Selain itu, dapat menggunakan Postman dan akan menampilkan data sebagai berikut:



## 1.6 HASIL DAN ANALISIS DATA

### Percobaan 1: Hello world API

- Hasil percobaan 1:
- Analisis:

### Percobaan 2: Membuat model dan kontroler baru

- Hasil percobaan 2:
- Analisis:

## **1.7 KESIMPULAN**

## **1.8 TUGAS MANDIRI**

Buatlah program untuk menampilkan satu data Person dengan menambahkan filter id\_person dengan menggunakan method GET. Tampilkan hasilnya pada Swagger dan Postman.

## **1.9 DAFTAR PUSTAKA**

1. Brenda Jin, Saurabh Sahni, and Amir Shevat, “Designing Web APIs, Building APIs That Developers Love”, Published by O’Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472, 2018.
2. Marinki Spasojevic, Vladimir Pecanac, “Ultimate Asp.Net Core Web Api”, -
3. <https://learn.microsoft.com/en-us/aspnet/core/tutorials/first-web-api?view=aspnetcore-7.0&tabs=visual-studio>