API Documentation

Introduction

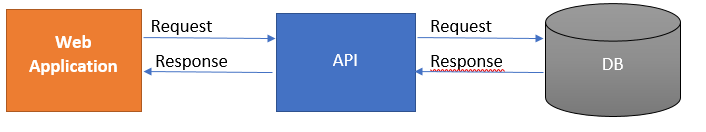
Roadmap

* What is API?
* Why is API required?
* How to create an ASP.NET Core Web API

**What is API?**

Web API as the name suggests, is an API over the web which can be accessed using HTTP protocol. It is a concept and not a technology. We can build Web API using different technologies such as Java, .NET etc.

API stands for Application Programming Interface. It is an intermediate software agent that allows two or more applications to interact with each other.

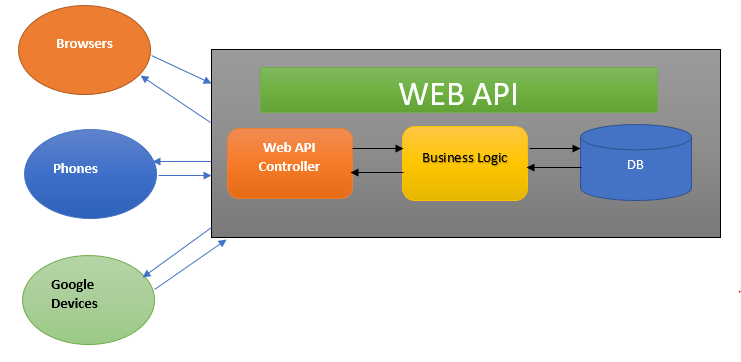


**Why is Web API required?**

The user wants to access the application from different devices like mobile, browser, Google devices, etc. In this case, Web API can be useful.

Different devices request to Web API and Web API will respond in JSON format. Most of the devices are able to understand JSON output.

Let’s see the architecture diagram below:



This diagram explains the architecture of Web API.

1. A client called api/controller – In the above diagram Browers, Phones, and Google Devices are called Web API Controllers.
2. api/Controller interact with business layer and get Data from DB.
3. The output will be returned in JSON format.

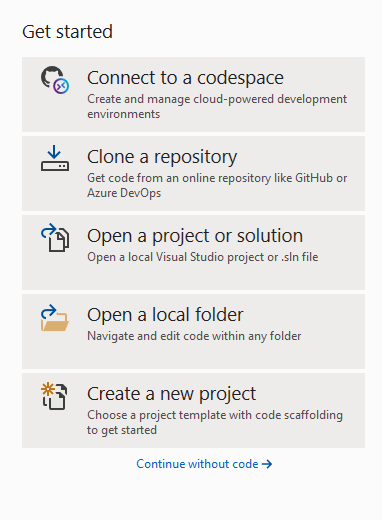
This is very basic Web API.

## How to create an Asp.Net core web API?

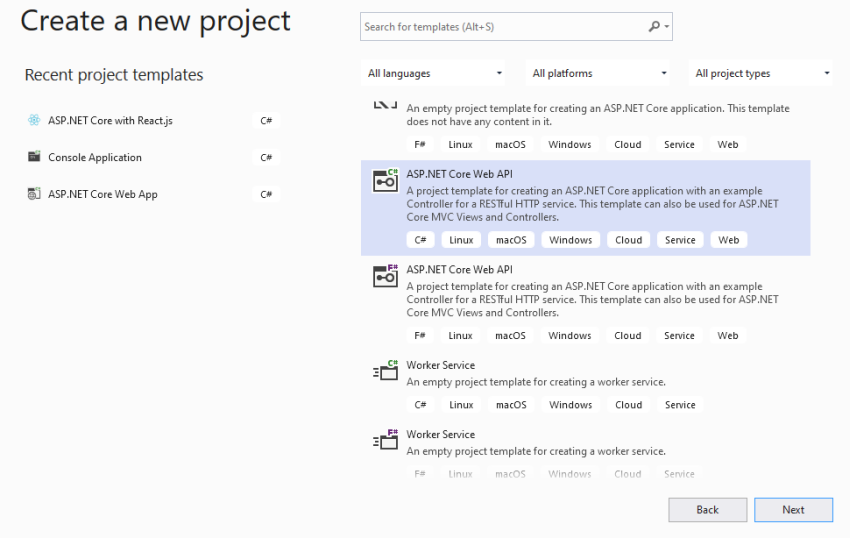
We will create our first simple Web API using Visual Studio 2019. You can download the free community version from the Microsoft official site.

Follow the below steps to create your first Web API project,

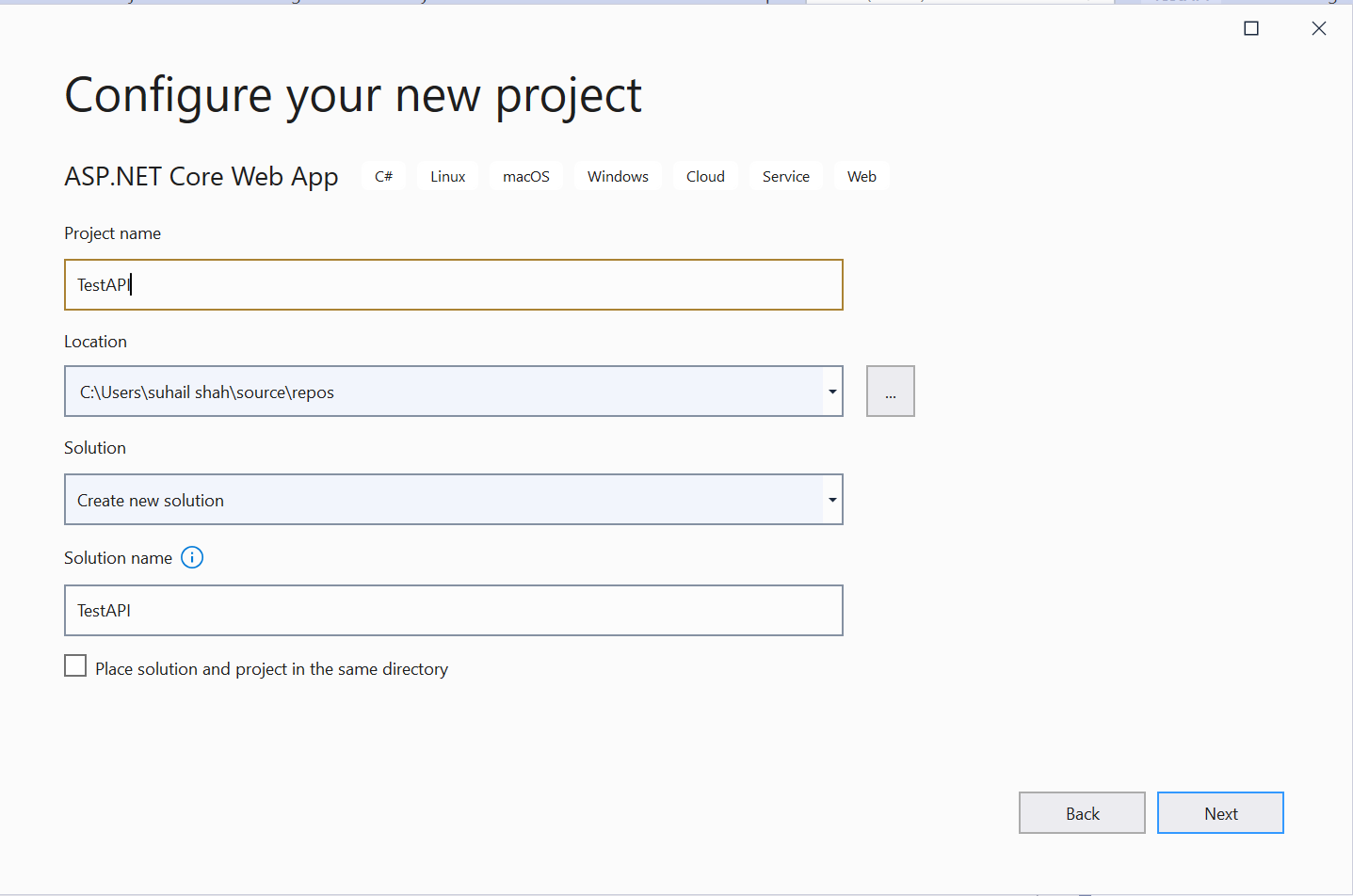
Open Visual Studio 2019 and create a new project.



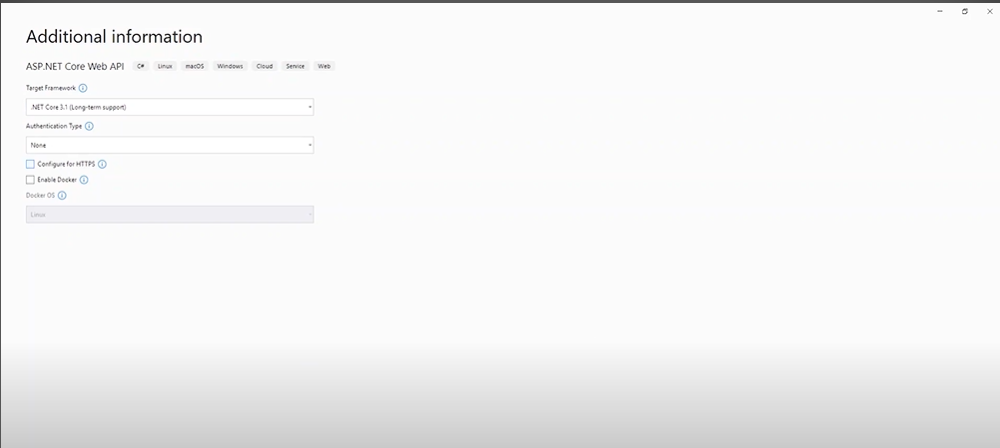
Select the “Asp.Net Core Web API” template and click on Next.



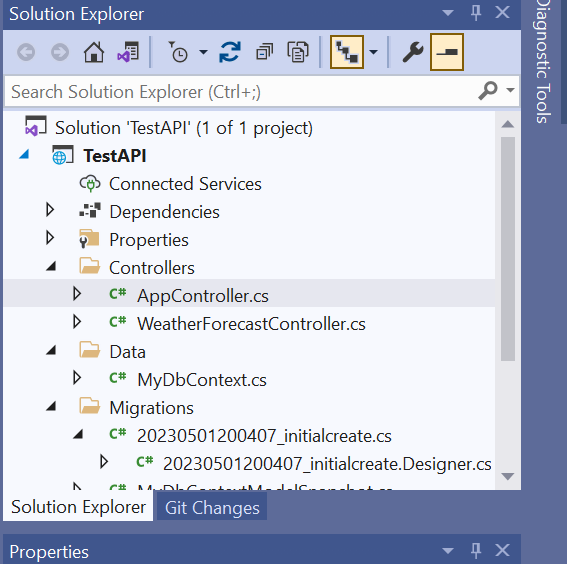
Provide Project name and location.



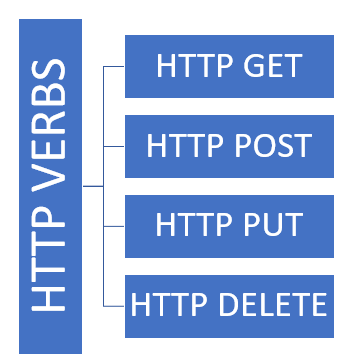
Select Target Framework and click on Create button.



TestAPI is created. See below project structure.



Web API is mostly used for CRUD (Create, Update, EDIT, DELETE) operations. It follows HTTP verbs for these operations.



* HTTP GET – Read Operation
* HTTP POST – Create Operation
* HTTP PUT – Update Operation
* HTTP DELETE – Delete Operation

First, we will create Event Model and then create AppController.

**Step 1**

Create two Folders -> Data and Model

**Step**2

Create new class Called Event in Model.

**Step 3**

Create new class MyDbContext in Data.

Note : Use own Connection String from SQL Server.

**Step 4**

Create a Controller in Controller Folder AppController

Configure Route to match the requirement.

**Step 5**

Add CRUD Operations in AppController

**Step 6**

Open Packet Manager Console and Write

* Add-migration (“initialcreate”)
* Update-database

This will create the table in the database in SQL Server with columns mentioned in the model.

**Step 7**

Start Debugging and post the URL in Postman to fetch the APIs.

