Web API with .NET Core

Introduction to C# and Web API Project Structure.

January 23, 2020

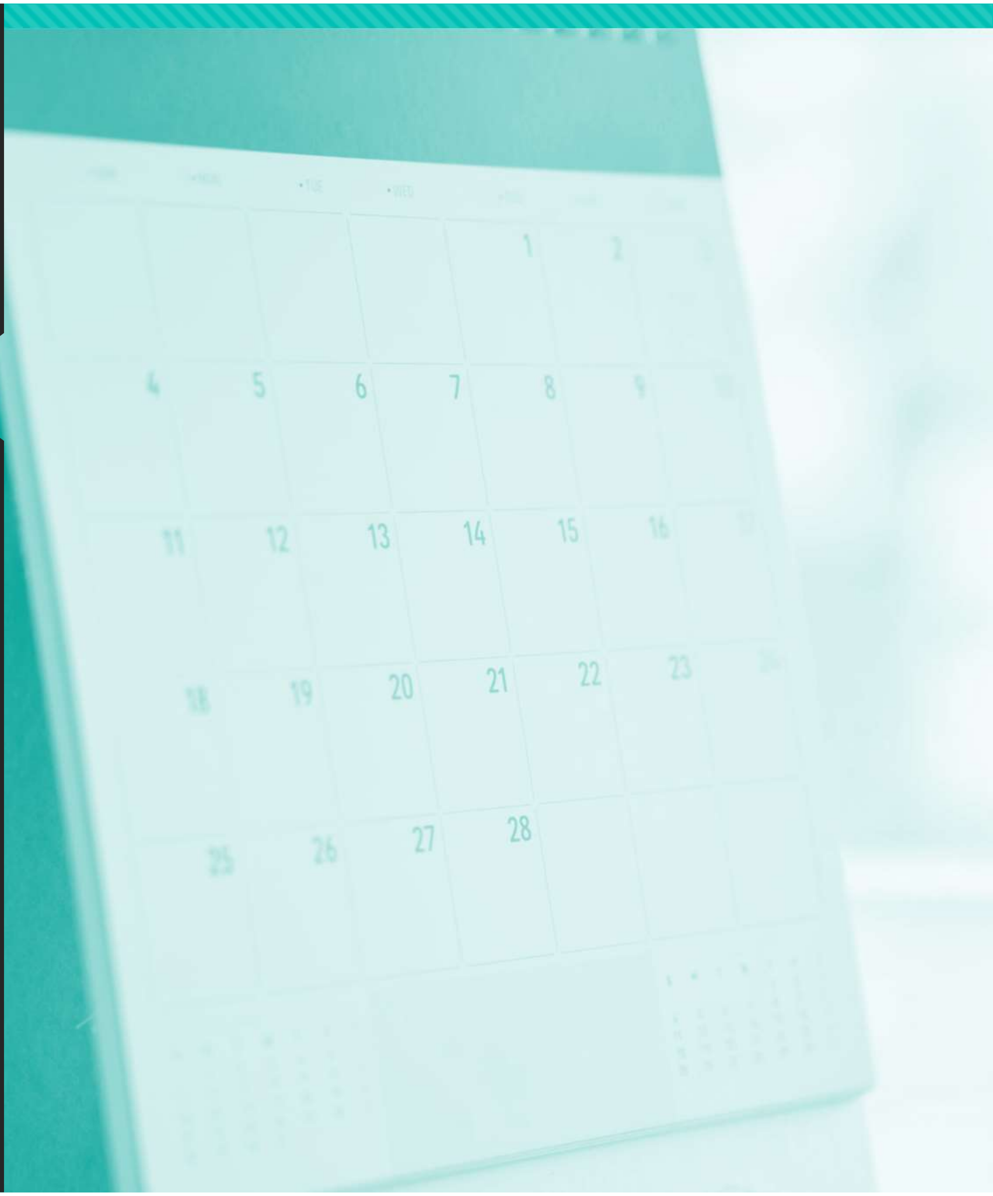
New room?

This room (EOS)

Different Room (Maybe a BLL class)

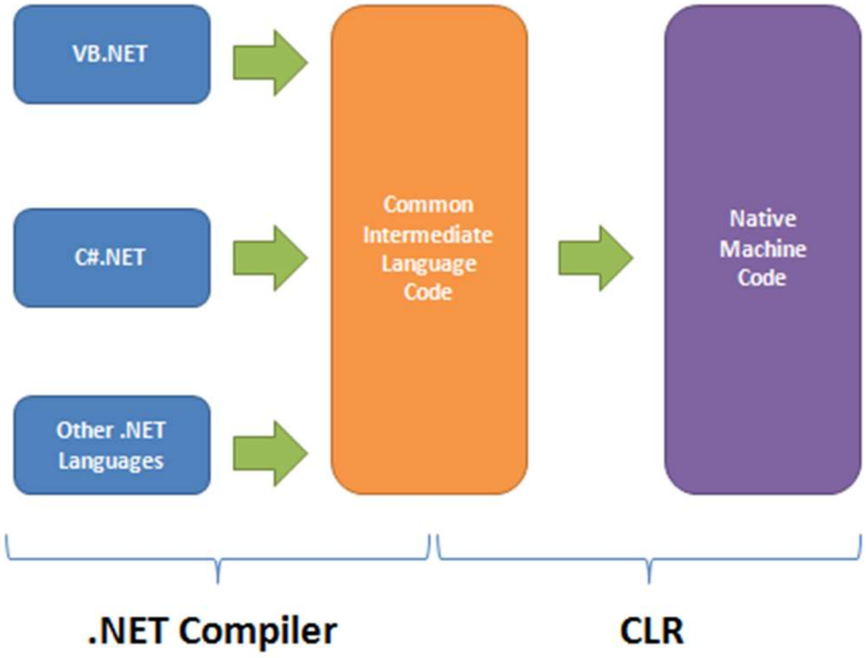
Upcoming Events

- **Lunch at Initech Global (Feb . 18, 19, or 20)** – What day works best?
- Other Events:
  - **HashCode Feb. 20**
    - Dealine: Feb. 17
  - **CalvinHacks Feb. 21-22**
    - Application is Open
  - **DataFest March 20-22**
    - Deadline: Feb 14



Intro to C#

What’s the deal with .NET?

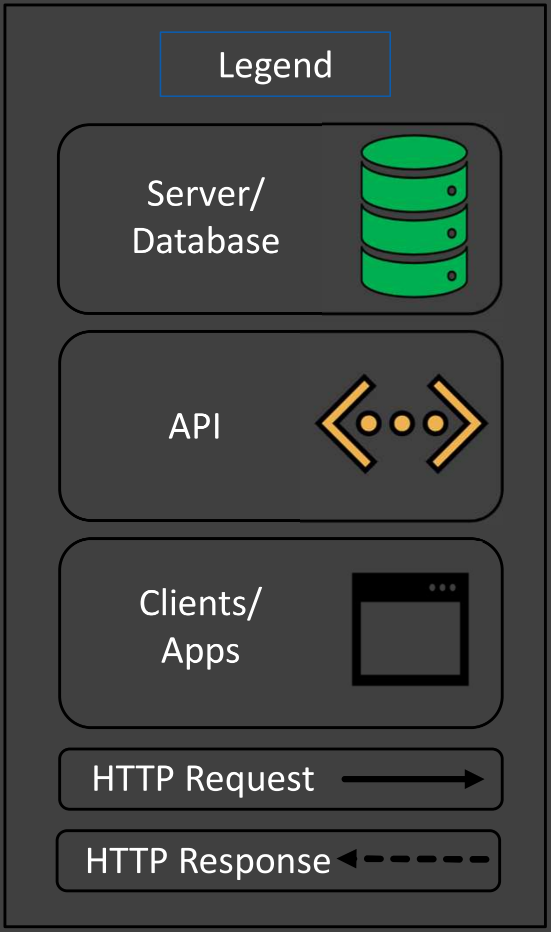


Lets cover a few things

- Dotnet run and build
- Console.Write()
- Namespaces – Basically a Java package
- IFs, Loops, so on
- Strings
  - == Works
- Lists
- Dictionaries
- Classes
  - Member Variables
  - Inheritance
- PascalCase for member variables and functions
- camelCase for local variables

Review

What is a Web API?



Legend

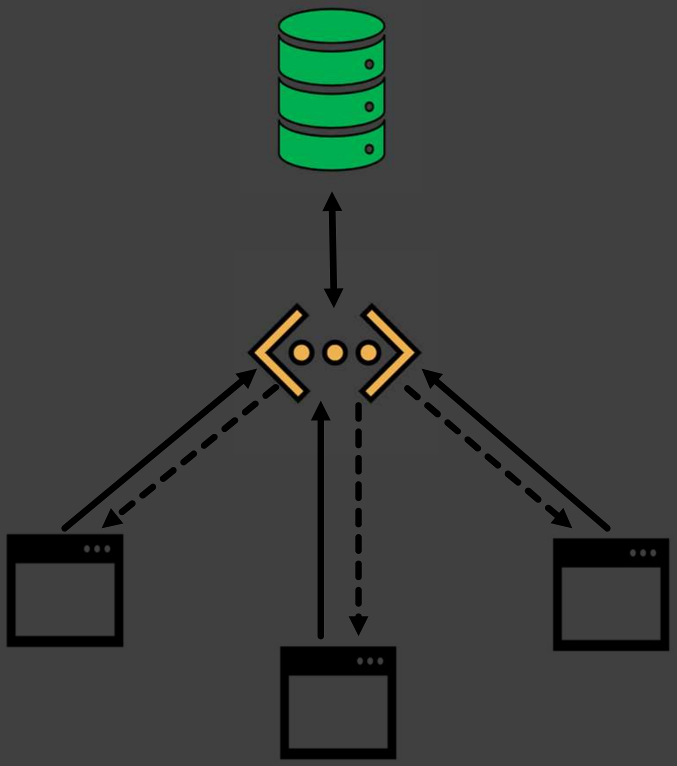
Server/ Database

API

Clients/ Apps

HTTP Request

HTTP Response



Why use a Web API?

- **Centralization** of data access.
- Any kind of app on any platform can call the API.
- The model is loosely coupled.

Lets get started!

Do Step 1

Notable Files

Program.cs

Entry point

Startup.cs

Where services are configured

We will mess around with this a little

Time to continue

Step 2 - 4

Making API Endpoints for HTTP Requests

Controllers

Creates the API endpoints in URL Format

RESTful HTTP Requests

- **GET** – Retrieve Data
- **POST** – Create Data
- **PUT** - Update Data
- **DELETE** – Delete Data

Models

Blueprint for HTTP Response Body

- Looks at the class, and converts the data into JSON Format

Lets make an Endpoint!

GET Request

Intended for getting data

Routing

Next Week

- Hooking us up to a database (We can’t do the others without one)
- POST, PUT, DELETE