**What this isn’t:**

* This is not an Entity Framework session
* This is not an AngularJS session
* This is not a CORS session
* This is not a “Code-First” Web API session

**What this is:**

* This is a Database-First Web API session
* This is a SOLID design in Web API session
* This is a RESTful Web API session
* This is a session that discusses
  + What is the “easy” way?
  + What is the “hard” way?

**Getting Started**

* Open Visual Studio (using 2012 in this example)
  + File->New->Web Site…
    - This isn’t what we want
  + File->New->Project..
    - On the left Templates->Visual C#->Web
    - Choose **ASP.NET Web API 2 Empty Project**
    - Name it **DbWebAPI**
    - Note the “App\_Data”, “App\_Start”, “Controllers”, “Models” folders
    - Note the “App\_Start\WebApiConfig.cs” and the “Global.asax.cs” pre-setup
* Open the Nuget Console
  + Install-Package Microsoft.AspNet.WebApi.HelpPage
  + Now we have BOTH MVC and WebAPI2 installed
* Launch in Debug
  + **Failure!**
* Add this to the Global.asax.cs file

protected void Application\_Start()

{

// Add this code, if not present.

AreaRegistration.RegisterAllAreas();

// ...

}

* Does it work now?
  + No, but yes -> /help/
  + Let’s add an “index.html” to get there easier
* Let’s add our controller
  + Controllers folder -> Add controller…
    - **Web API 2 Controller with read/write actions**
      * ProductController
* Let’s visit the website again.
* FINALLY DATABASE STUFF
  + Open SQL Server Mgmt
  + Create WaffleStore Catalog
* Let’s hook up a database
  + In VS-> Server Explorer
    - WaffleStore Catalog
* Let’s be “easy”
  + Install-Package EntityFramework
  + Let’s look at the web.config changes
  + Under “Models” -> Add->New Item…
    - On the left Visual C#-> Data
    - ADO.NET Entity Data Model
      * WaffleDbModel
  + Under “Controllers”
    - Delete ProductController.cs (WHAT?!)
    - Add new Controller…
      * **Web API Controller with actions, using Entity Framework**
        + Name -> ProductController
        + Model class -> Product
        + Data context class -> WaffleStoreEntities
* Talk about DTO’s (Domain/DB Model -> Service Model-> View Model

**Script**

**Adding Help**

Install-Package Microsoft.AspNet.WebApi.HelpPage

Install-Package EntityFramework

Install-Package log4net

**References**

**Getting started with WebAPI**

[**http://www.asp.net/web-api/overview/advanced/configuring-aspnet-web-api**](http://www.asp.net/web-api/overview/advanced/configuring-aspnet-web-api)

[**http://www.asp.net/web-api/overview/getting-started-with-aspnet-web-api/creating-api-help-pages**](http://www.asp.net/web-api/overview/getting-started-with-aspnet-web-api/creating-api-help-pages)

**REST**

<http://blogs.msdn.com/b/martinkearn/archive/2015/01/05/introduction-to-rest-and-net-web-api.aspx>

<https://en.wikipedia.org/wiki/Representational_state_transfer>

**WebAPI with Entity Framework**

<http://www.asp.net/web-api/overview/data/using-web-api-with-entity-framework/part-1>

[**http://www.asp.net/web-api/overview/data/using-web-api-with-entity-framework/part-1**](http://www.asp.net/web-api/overview/data/using-web-api-with-entity-framework/part-1)

**Database First**

<https://www.asp.net/mvc/overview/getting-started/database-first-development/setting-up-database>

**Code first**

<https://www.asp.net/mvc/overview/getting-started/getting-started-with-ef-using-mvc/creating-an-entity-framework-data-model-for-an-asp-net-mvc-application>

<https://msdn.microsoft.com/en-us/data/dn456843.aspx>

**MVC / MVVM**

<https://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93controller>

<https://en.wikipedia.org/wiki/Model_View_ViewModel>

**SOLID**

<https://en.wikipedia.org/wiki/SOLID_(object-oriented_design)>

**LinqPad**

<https://www.linqpad.net/>