Section 1+2

Using dotnet 5 ~~core~~ API (Application Programming Interface – knowing abbreviations is useful)

ASP.Net in particular

API is basically letting you communicate between different end points. The API is similar to a waiter in the restaurant. (An explanation I liked - [What is an API? (Application Programming Interface) | MuleSoft](https://www.mulesoft.com/resources/api/what-is-an-api))

Entity Framework

CLI – command line interface

(OT – NVM is Node Version Manager, allowing you to change versions of Node.js which is irrelevant to the project, but good to be aware of)

A Walking Skeleton – a model of how the program will be built. My understanding was like how a baby grows: the nerves system lets different part to communicate, while the bones (architecture of the software) are not fully fledged yet. You build each part along the way till the skeleton walked to the end and finished.

\* note: I was used to using VS for C# and now using VSC, and it’s not so bad, just need to learn what to install.

prop tab / propfull tab in VSC – shortcut for creating new properties in your C# classes.

In Windows version – ctrl + ‘.’ (ץ) is quick fix

Id/ID and UserName are conventions of Entity Framework.

A question: Neil suggests prefixing fields with underscores ‘\_’ instead of using “this.” – is it a preference known as sort of a global convention?

“API controllers provide the routine end points for where we can find our application.”

For instance – [HttpGet] is an endpoint

[ApiController] // Summary:

    //     Indicates that a type and all derived types are used to serve HTTP API responses.

    //     Controllers decorated with this attribute are configured with features and behaviour

    //     targeted at improving the developer experience for building APIs.

    //     When decorated on an assembly, all controllers in the assembly will be treated

    //     as controllers with API behaviour.

IEnumerable – allows us to use simple iteration over a collection. We can use List, but it offers a lot more than required, so keep it simple.

DB requests? Always async, it’s not a surprise, but it’s something good to fixate in the brain.

Quote from an unrelated other source: “an entity is a glorified array of components with some accessors and management functions.”