FUNDAMENTOS DE FASTAPI

Requisitos necesarios para trabajar:

1. Visual Studio Code
2. Python 3.10+
3. Pip: Instalador de librerías de Python

Python es un lenguaje que tiene su potencia en que se comunica con todo.

Su gran incremento de demanda se debe a Hacking y Big Data.

Tenemos múltiples ramas dentro de Python:

* Hacking
* Big Data: Gráficos y análisis de datos
* Jupyter: Big data
* Programas sencillos
* Django: Aplicaciones Web
* FastApi: Creación de Apis con lenguaje Python

A todos estos elementos se les llama Frameworks. Un Framework es una arquitectura basada en un lenguaje y que genera una aplicación.

Un Api es un servicio que permite representar datos y también nos permite interactuar con dichos datos, ya sea para modificar, insertar o recuperar datos.

Un servicio Api es tan popular porque representa datos en su mínima expresión, es decir, devuelve texto.

El texto que devuelve es un estándar, es decir, un texto representativo con un estándar que puede ser tanto JSON como XML.

JSON es una representación textual de un objeto (más popular)

XML es una representación textual con etiquetas de un objeto. (España)

Un servicio Api está formado por diversas características:

* Controller: Son los routers que permiten acceder a unos determinados Endpoints.
* EndPoints: Son los accesos URLS para las peticiones a cada elemento de un Controller.
* Models: Son las clases que puedo recibir/enviar entre las peticiones a cada Controller.
* Responses: Las respuestas pueden ser simples o pueden ser algo más complejas, depende de nuestro servicio.
* Services: Son los elementos de dónde recuperamos los datos a utilizar en el Api y sus controladores.

Además de estas características un servicio puede tener una serie de métodos dentro de los controladores:

* GET: Este método se utiliza cuando recuperamos datos de cualquier Api
* PUT: Este método se utiliza cuando queremos modificar algún elemento dentro del Api.
* DELETE: Este método se utiliza cuando queremos eliminar algo del Api.
* POST: Este método tiene dos vertientes:
  + Insertar información en el Api.
  + Enviar objetos y hacer con ellos lo que deseemos, por ejemplo, en la seguridad de los Apis se utiliza información POST.

<https://apicruddepartamentoscore.azurewebsites.net/index.html>

Departamentos es nuestro **Controller**

GET: Api/Departamentos es un **EndPoint**

Lo que estamos visualizando es un Api con documentación **Swagger**

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

Configuración de nuestro entorno de trabajo.

En algún momento crearemos un Fast Api en la carpeta en la que tengo el documento de Word.

El primer paso para utilizar Apis es consumir dichos Apis.

Lo bueno de un Api es que se puede consumir con cualquier lenguaje:

* JavaScript, JQuery, React, Angular, Java, C#, Python

Lo primero que vamos a realizar es crear una carpeta y consumir el Api de departamentos con un GET.