Flask: es un microframework minimalista escrito en python el cual nos permite crear páginas web dinámicas de manera muy sencilla y con una curva de aprendizaje muy baja.

Framework: Es un conjunto de herramientas que nos dan un esquema de trabajo junto con una serie de funciones y utilidades que nos facilitan la construcción de páginas web dinámicas.

¿por qué micro? → únicamente te da las bases de lo que necesitas para empezar a programar, sin la necesidad de configurar muchos archivos ni módulos para empezar a programar. Osea cuando instalamos Flask tenemos las herramientas necesarias para crear una aplicación web funcional, pero si llegáramos a necesitar una nueva funcionalidad hay una gran cantidad de extensiones que podemos instalar para aumentar la funcionalidad.

¿Por qué es minimalista?--> porque no necesita ser dependiente de otras cosas para hacer cosas básicas

Flask nos da la libertad de elegir qué librerías o módulos usar que nos ajuste a lo que necesitamos, osea no se casa con ninguna en particular.

¿Por qué tiene una curva de aprendizaje muy baja? → no tenes que tener muchos conocimientos para empezar a programar cosas y vas integrando conocimiento de a poco para ir sumando funcionalidades. Osea no tenes que aprender muchas cosas de golpe para ir creando aplicaciones en Flask

PORQUE USAR FLASK

* Incluye un servidor web de desarrollo: no necesitamos una infraestructura con un servidor web aparte, para crear nuestras aplicaciones. Esto nos permite de manera sencilla correr un servidor web para ir viendo los resultados que vamos obteniendo al programar. (jinja 2)
* Incluye un depurador: si tenemos un error podemos depurar ese error para poder ver el estado de las variables y poder así ver dónde hemos cometido errores.
* Incluye un soporte integrado para pruebas unitarias: podemos ir haciendo pruebas para ir probando nuestra aplicación.
* Es compatible con WSGI: (es el estándar o protocolo que utilizan los servidores web para servir las páginas web en python) por lo que es fácil poner nuestra aplicación en producción.
* Buen manejo de Rutas: cuando trabajamos con aplicaciones tenemos el controlador que reciben las peticiones de los clientes y tenemos que determinar que ruta esta accediendo el cliente para entonces ejecutar el código necesario, Flask tiene un buen manejo de estas rutas.
* Soporta el uso de cookies
* Se pueden usar sesiones
* Podemos usar extensiones para ORM’s (no tiene ORM propia)
* Mucha documentación disponible

EXTENSIONES DE FLASK (estas son algunas)

* Flask-Script: me permite tener un comando de la linea de comandos para manejar nuestra aplicación.
* Flask-Bootstrap: como framework de hojas de estilo para nuestras páginas.
* Flask-WTF: sirve para generar los formularios HTML con clases y objetos.
* Flask-Sqlalchemy: puede generar el modelo de datos a partir con esta extensión.
* Flask-login: para que la app maneje usuarios y contraseñas autentificarse en la aplicación.

DIFERENCIAS CON DJANGO

* Flask nos permite integrar solo lo que realmente necesitemos, Django es más robusto el cual ya tiene pre-cargado varias librerías, paquetes y settings que hay que configurar. Con Flask se arranca desde cero.
* Django es muy seguro, viene con muchos ficheros de seguridad integrados como encriptación de bases de datos, web top games, etc. todo esto viene por defecto.
* Django tiene una ORM muy sencilla de utilizar.
* Django también tiene mucha documentación y recursos disponibles para no perderse, y una gran comunidad respaldandolo.
* Desventaja de Django: es un framework para servidores monolíticos (hay un único servidor que se encarga de todo o principal y los demas servicios estan basados en este servidor) ya que fue creado antes de que esta arquitectura de microservicios comenzara a ser popular.
* Desventaja de Django: es un poco más lento que otros frameworks (aunque suele ser suficiente)
* Ventaja de Flask: es muy sencillo y ligero, lo cual lo hace mas rapido al estar ejecutando menos cantidad de código y muy pequeño el paquete Flask
* Ventaja de Flask: es muy flexible, ya que si no le queres instalar nada extra a Flask se puede dejar asi y va a ser una app muy rápida y sencilla y conforme vas ocupando mas cosas las vas integrando al proyecto.
* Ventaja de Flask: posee una gran cantidad de librerías para poder integrar, muy diferente de django el cual solo tiene un camino y solo se puede seguir ese camino.
* Flask no tiene una ORM propia pero podes elegir entre varias cual podes integrar a Flask (SqlAlchemy, PonyORM o cualquier otra ORM de python)
* Desventaja de Flask: la documentación esta muy dispersa, por el hecho de que cada uno utiliza flask de manera diferente y puede resultar difícil encontrar por ejemplo como integrar tal librería con el framework. Esto es por el hecho de ser tan flexible.
* Ventaja de Django: mas facil encontrar la documentación, ya que siempre utiliza la misma librería y la misma arquitectura. Por ello la mayoría de tutoriales y documentación disponible va a servir para la aplicación que estes creando.
* El mercado esta dominado por Django, por lo que hay mayor demanda de desarrolladores de Django que de Flask

REQUISITOS PARA APRENDER FLASK

Mediano conocimiento de Python

Mínimos conocimientos de HTML