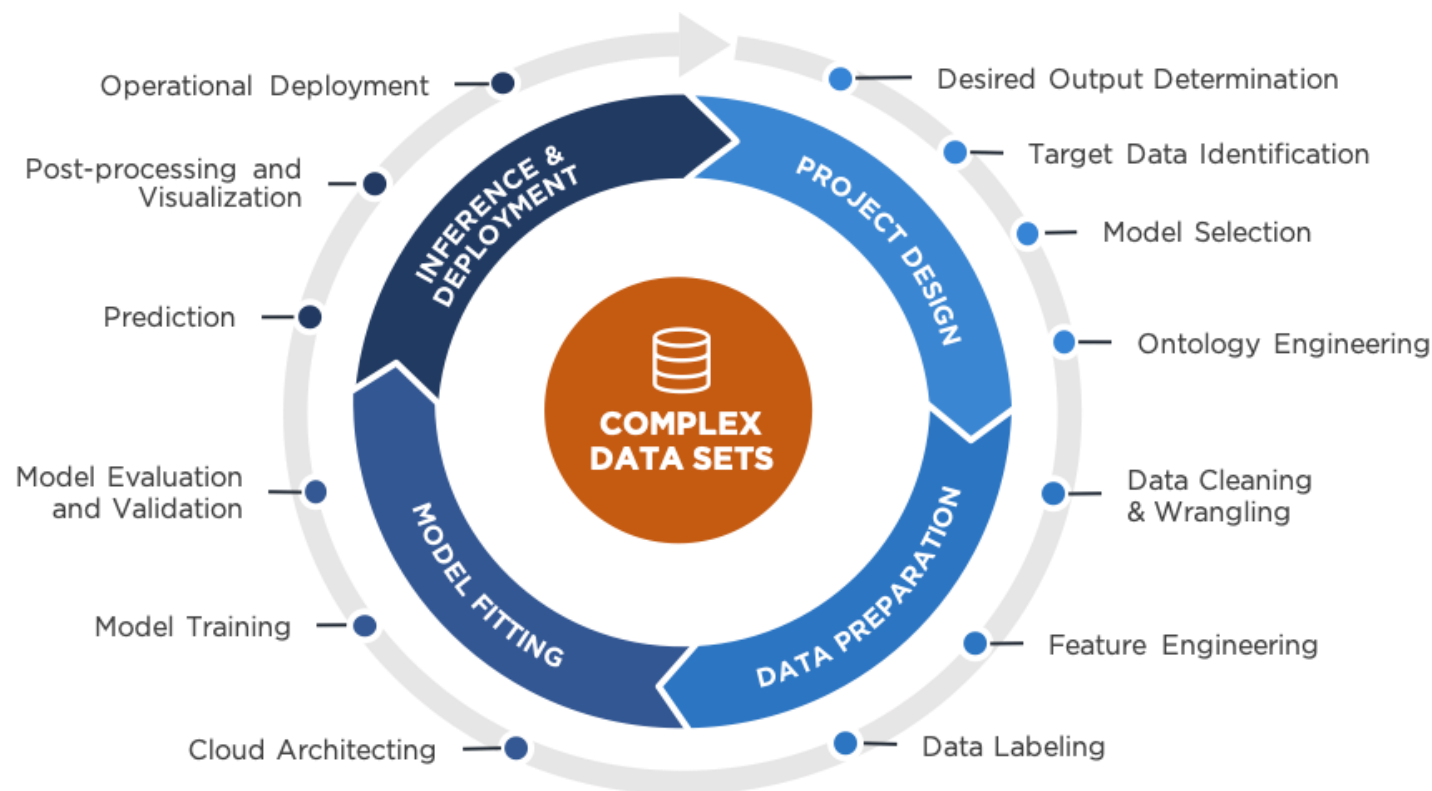


DESPLIEGUE DE MODELOS DE APRENDIZAJE AUTOMÁTICO EN API CON FASTAPI

Poner en Producción

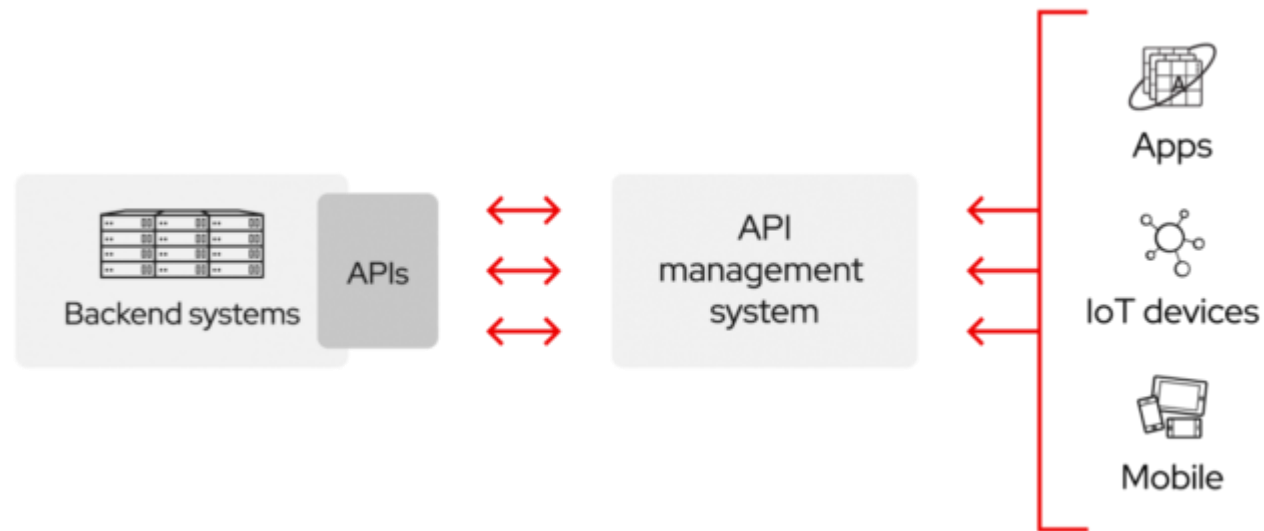
Es un **proceso** en el que un modelo se integra en un **entorno de IT** y se pone **a disposición** de otros para su uso y consumo.



Qué es una API?


API (**A**pplication **P**rogramming **I**nterface) crea un **punto de entrada** para una aplicación, a través de solicitudes **HTTP**.




API: **Abstracción** de la aplicación + simplificación de la **integración** de terceros




¿Qué es FastAPI?

Es el **marco de referencia** para crear **API robustas** y de alto rendimiento para entornos de **producción**.



FastAPI




Search


tiangolo/fastapi
0.65.3 32.6k 2.3k

FastAPI
[FastAPI](#)
Languages >
Features
FastAPI People
Python Types Intro
Tutorial - User Guide >
Advanced User Guide >
Concurrency and async / await
Deployment >
Project Generation - Template
Alternatives, Inspiration and Comparisons
History, Design and Future
External Links and Articles
Benchmarks
Help FastAPI - Get Help
Development - Contributing
Release Notes

FastAPI



FastAPI framework, high performance, easy to learn, fast to code, ready for production

Test passing
coverage 100%
pypi package v0.65.3

Documentation: <https://fastapi.tiangolo.com>

Source Code: <https://github.com/tiangolo/fastapi>

Table of contents
Sponsors
Opinions
Typer, the FastAPI of CLIs
Requirements
Installation
Example
 Create it
 Run it
 Check it
 Interactive API docs
 Alternative API docs
Example upgrade
 Interactive API docs upgrade
 Alternative API docs upgrade
Recap
Performance
Optional Dependencies
License

Características: Sintaxis simple

Similar a **Flask**. No servidor web integrado → Uvicorn (servidor ASGI)



FLASK

```
from flask import Flask

app = Flask(__name__)

@app.route("/")
def home():
    return {"Hello": "World"}

if __name__ == "__main__":
    app.run()
```

FASTAPI

```
import uvicorn
from fastapi import FastAPI

app = FastAPI()

@app.get("/")
def home():
    return {"Hello": "World"}

if __name__ == "__main__":
    uvicorn.run("hello_world_fastapi:app")
```

Características: Definir rutas



FLASK

```
from flask import request

@app.route("/", methods=["POST", "GET"])
def home():
    if request.method == "POST":
        return {"Hello": "POST"}

    return {"Hello": "GET"}
```

FASTAPI

```
@app.get("/")
def home():
    return {"Hello": "GET"}

@app.post("/")
def home_post():
    return {"Hello": "POST"}
```

Características: parámetros de ruta y de consulta



FLASK

```
@app.route("/employee/<int:id>/")
def home():
    return {"id": id}
```

```
@app.route("/employee")
def home():
    department = request.args.get("department")
    return {"department": department}
```

FASTAPI

```
@app.get("/employee/{id}")
def home(id: int):
    return {"id": id}
```

```
@app.get("/employee")
def home(department: str):
    return {"department": department}
```

Características: Solicitudes asíncronas

Carecen de solicitudes asíncronas: Flask y Django.

Programación asíncrona: el código se ejecuta por separado del hilo principal de la aplicación.

Async al controlador y el **await** como argumento antes de llamar a esa función.

FastAPI


```
app = FastAPI()
```

```
async def some_blocking_operation():
```

```
# I/O read or write
```

```
# a remote API call
```

```
# database read and write
```

```
# ...
```

```
pass
```

```
@app.get("/")
```

```
async def read_results():
```

```
results = await some_blocking_operation()
```

```
# other computations
```

```
# ...
```

```
return results
```


Características: Escritura con Pydantic

Para la **validación en tiempo de ejecución** y permite que su IDE se **complete automáticamente** y **detecte errores** relacionados con el tipo.

Ejemplo: API para publicar solicitudes de empleo y recibir decisiones aleatorias. Clase Application realiza la validación del tipo de datos: first_name de tipo string, age de tipo entero, etc

```
class Application(BaseModel):  
    first_name: str  
    last_name: str  
    age: int  
    degree: str  
    interest: Optional[str] = None
```

Características: Validación de parámetros

FastAPI permite **validar las entradas** del usuario agregando **restricciones** en los parámetros de consulta.

Validación de parámetros de consulta de cadenas. Establecemos una longitud máxima o mínima y una expresión regular en la que la consulta debe coincidir.

Validación de parámetros numéricos. Al usar los parámetros numéricos establecemos un rango como restricción usando **gt** (mayor que) y **le** (menor o igual que).

Características: manejo de errores y mensajes personalizados

Puedes definir y generar **errores personalizados** con un mensaje y un código de estado específicos para facilitar la **depuración** de errores

← → ↻ ⓘ 127.0.0.1:8000/students/12444

```
{"detail": "Student ID not found"}
```

Code	Details
------	---------

404	
-----	--

<i>Undocumented</i>	
---------------------	--

	Error: Not Found
--	------------------

	Response body
--	---------------

	<pre>{ "detail": "Student ID not found" }</pre>
--	---

Generación automática de documentos

Se generan **automáticamente** documentación y exploración de API interactivas

<http://localhost:8000/docs>

<http://localhost:8000/redoc>

FastAPI 0.1.0 OAS3
OpenAPI JSON

default

POST /applications Create Application Try it out

Parameters

Name	Description
id <small>required</small>	id
Integer (query)	

Request body required application/json

Example Value Schema

```
{
  "first_name": "string",
  "last_name": "string",
  "age": 0,
  "degree": "string",
  "interest": "string"
}
```

Responses

Code	Description	Links
200	Successful Response	No links

Media type: application/json
Content Accept header:

Example Value | Schema

Search...

Read Student Id

Download OpenAPI specification Download

FastAPI (0.1.0)

Read Student Id

PATH PARAMETERS

student_id	Integer (Student Id)
<small>required</small>	

Responses

- > 200 Successful Response
- > 422 Validation Error

GET /students/{student_id}

Response samples

200	422
<pre>{ "first_name": "string", "last_name": "string", "age": 0, "degree": "string", "interest": "string" }</pre>	

Content type: application/json

Copy Expand all Collapse all