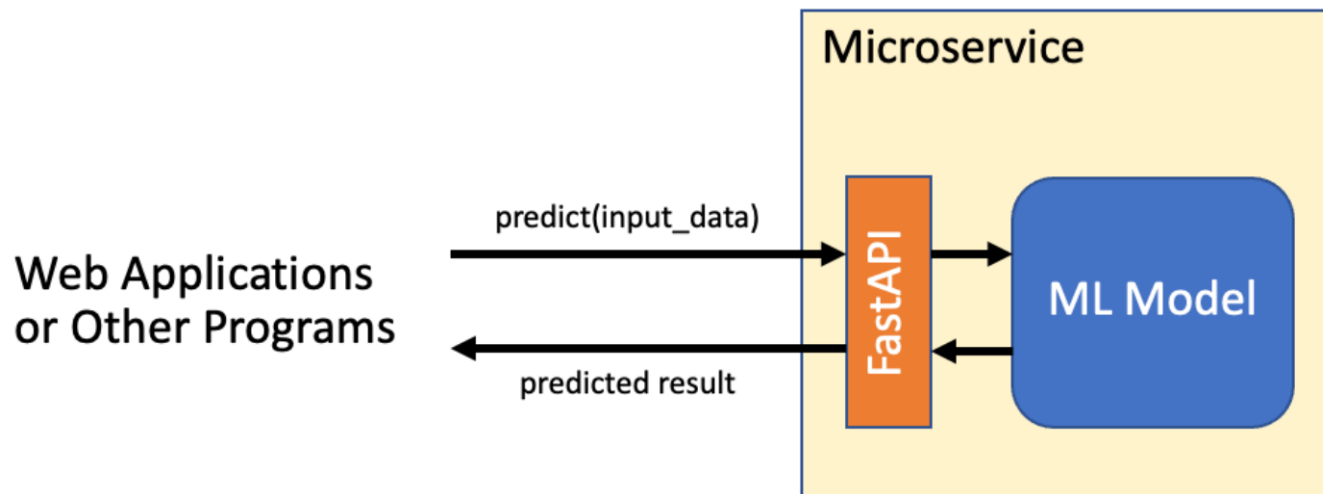# FastAPI

Based on Prof.Natawut's slide
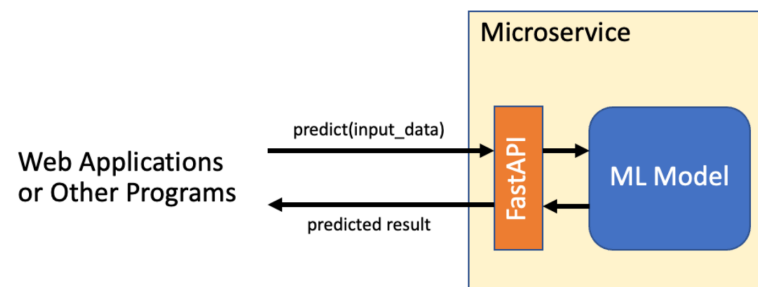
Provisioning ML Model : Synchronous
(interactive; expected to get results back)

- Encapsulating your model as a microservice is a popular approach to deploy machine learning model

- The model becomes a black box and can be invoked using REST protocol

- Using a microframework such as FastAPI can simplify microservice development
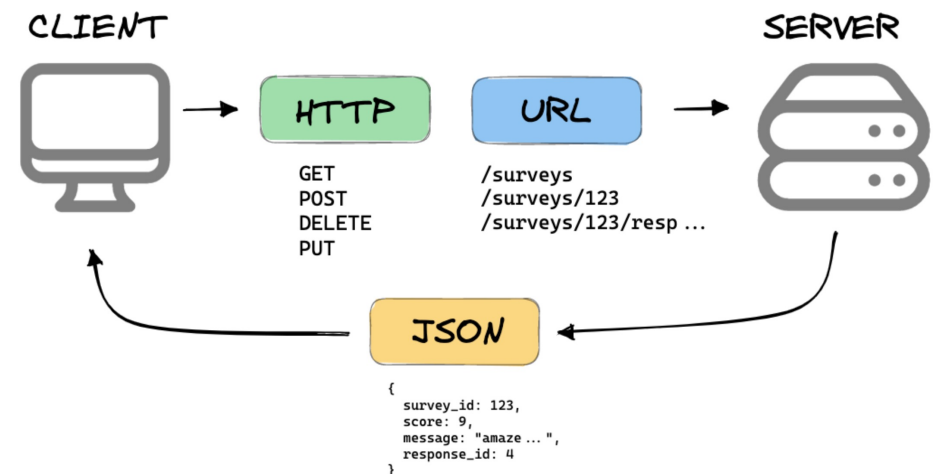
Microframework

- Minimalistic web application frameworks designed for Microservices and API development
- Facilitate receiving an HTTP request, routing the HTTP request to the appropriate controller, dispatching the controller, and returning an HTTP response
- Designed for building the APIs for another service or application; therefore, lacking most of the functionality expected in full-stack framework

REST - Heart and Soul of Microservices

- REpresentational State Transfer
- Lightweight and simple
  - Only simple XML or JSON on HTTP
  - Utilize HTTP methods instead of using "envelope" style
  - URI as method identification
  - Emulate CRUD operations
    - Create, Read, Update, Delete
    - **GET** → **Read**
    - **POST** → **Create**
    - **PUT** → **Update**
    - **DELETE** → **Delete**

WHAT IS A REST API?

CLIENT → HTTP → URL → SERVER

HTTP:
GET
POST
DELETE
PUT

URL:
/surveys
/surveys/123
/surveys/123/resp ...

JSON:
{
  survey_id: 123,
  score: 9,
  message: "amaze ... ",
  response_id: 4
}

REST over HTTP – GET method
(Postman, Python)

Request

```
GET /music/artists/beatles/recordings HTTP/1.1
Host: media.example.com
Accept: application/xml
```

Method

Resource

Response

```
HTTP/1.1 200 OK
Date: Tue, 08 May 2007 16:41:58 GMT
Server: Apache/1.3.6
Content-Type: application/xml; charset=UTF-8

<?xml version="1.0"?>
<recordings xmlns="…">
   <recording>…</recording>
   …
</recordings>
```

State transfer

Representation

FastAPI

- Web framework for developing RESTful APIs in Python based on Pydantic and starlette
- Provide several facilities for REST development
  - Data validation / serialization / deserialization
  - Auto-generate OpenAPI documents
  - Asynchronous programming
  - Dependency Injection
  - Support GraphQL for complex data model
  - Run with Uvicorn and Unicorn (simulated web server)

Installation

- FastAPI & uvicorn
- pip install fastapi uvicorn

- Postman in your laptop

- Ngrok (web tunnel) → for Colab
- https://ngrok.com/

Lab

- 1. Get
- 2. Get with path param
- 3. Post
- 4. Put