Microservice

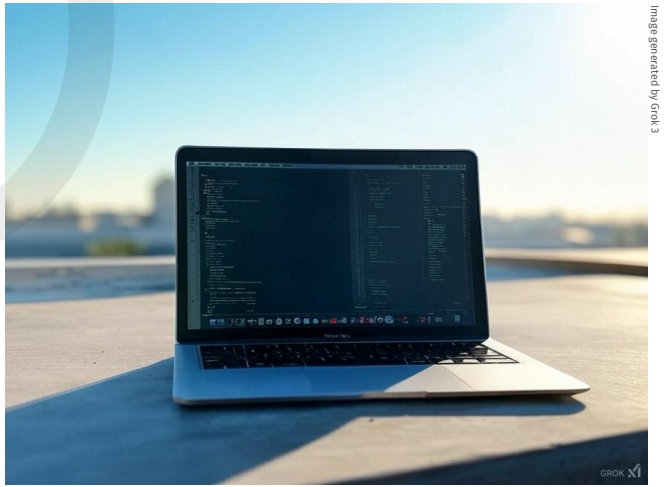


Image generated by Grok 3

Irwan Kautsar, Ph.D

(Assoc. Prof.)
Informatics Department
Faculty of Science and Technology
Universitas Muhammadiyah Sidoarjo

bit.ly/hepidad

Version 30022025.20.11

Prerequisites

1. Basic understanding of data format: **XML** and **JSON**
2. Having knowledge of **Internet Protocols**, **HTTP** is a must !
3. Already knowing how **web** works and **monolith** vs **web service**.
4. Know little **Python**, **Flask** are better. Web Service demos using Flask.
5. Grab the code at: https://github.com/hepidad/c006-microservice
   (Push ⭐ for notification update)

Web Development Stage

Microservice:

1. Next episode of **Web Service**.
2. Also, access datas through **HTTP**.
3. Also, Enable **machine to machine** access, **mobile** access. Mostly called **Web API**.
4. Also, users not directly access DB but expose data with format: **XML**, **JSON** (with selected **field**)
5. Tools for access (test the web service) : Curl, POSTMAN, bruno
6. Also, operate CRUDS data with Http Method (**GET**, **POST**, **PUT**, **DELETE**) [1]

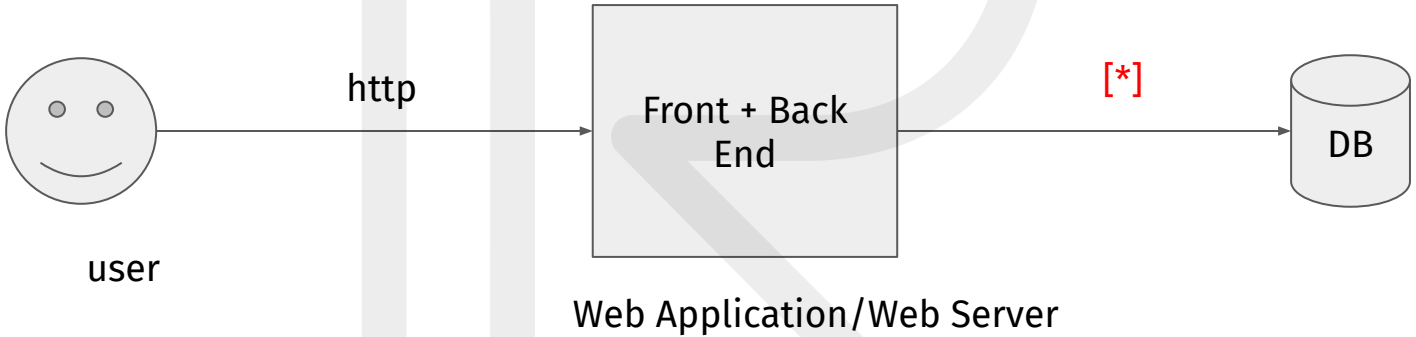
[1] Kautsar, I. A., Kubota, S. I., Musashi, Y., & Sugitani, K. (2014, December). Redefining data provider: The REST approach to solve Indonesia lecturer administrative problems. In 2014 IEEE International Conference on Teaching, Assessment and Learning for Engineering (TALE) (pp. 175-178). IEEE. https://doi.org/10.1109/TALE.2014.7062614

So, what differences with web service?

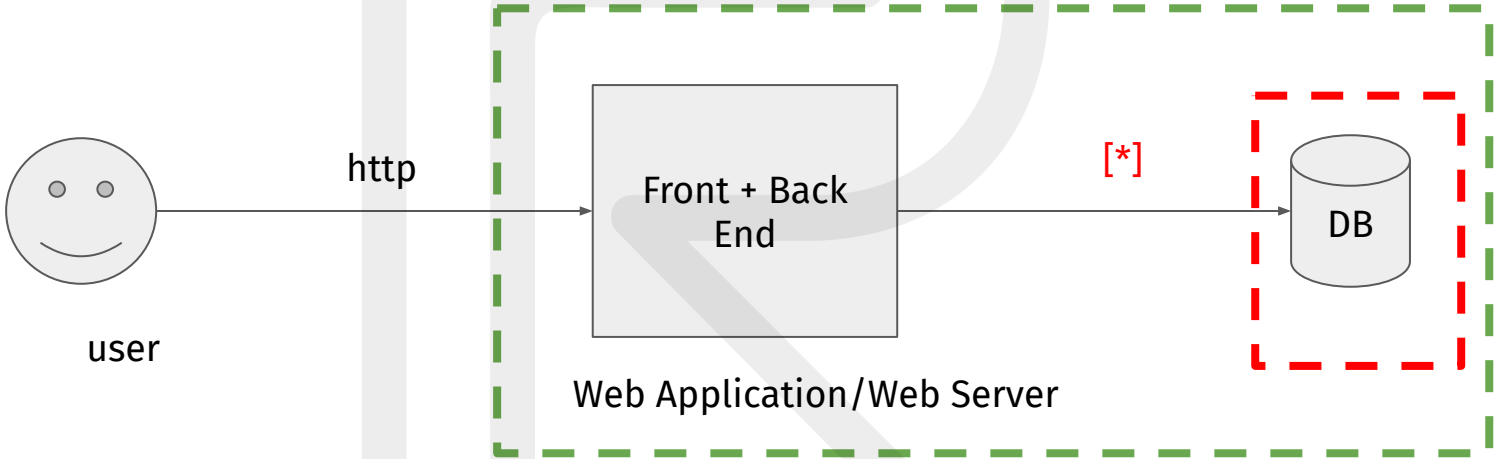
1. A program with specific features that is accessed through a **web service**. Or some web service with limited features (micro means simple).
2. Leveraging HTTP → lightweight and distribute incoming request into several connected microservices.
3. Microservice Architecture = Convert Monolith into more modular with service-based application.[1]

[1] Kautsar, I. A., Maika, M. R., Budiman, A. N., Setyawan, A. B., & Awali, J. Y. (2023, March). Microservice Based Architecture: The Development of Rapid Prototyping Supportive Tools for Project Based Learning. In *2023 IEEE World Engineering Education Conference (EDUNINE)* (pp. 1-6). IEEE. https://doi.org/10.1109/EDUNINE57531.2023.10102884

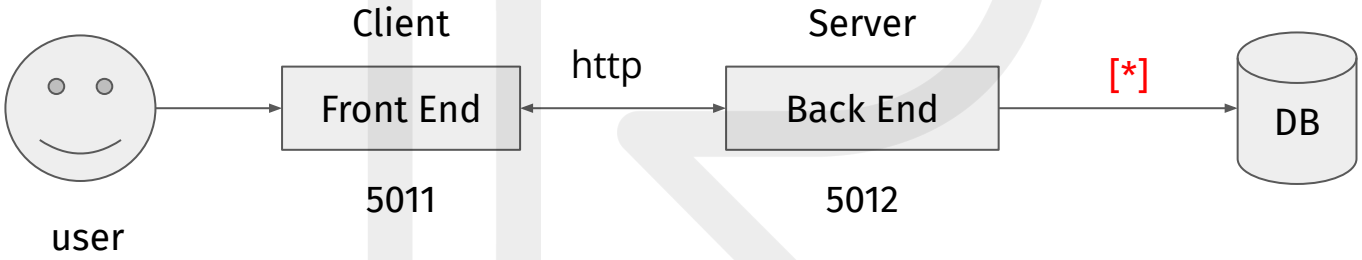
Meanwhile monolith ...



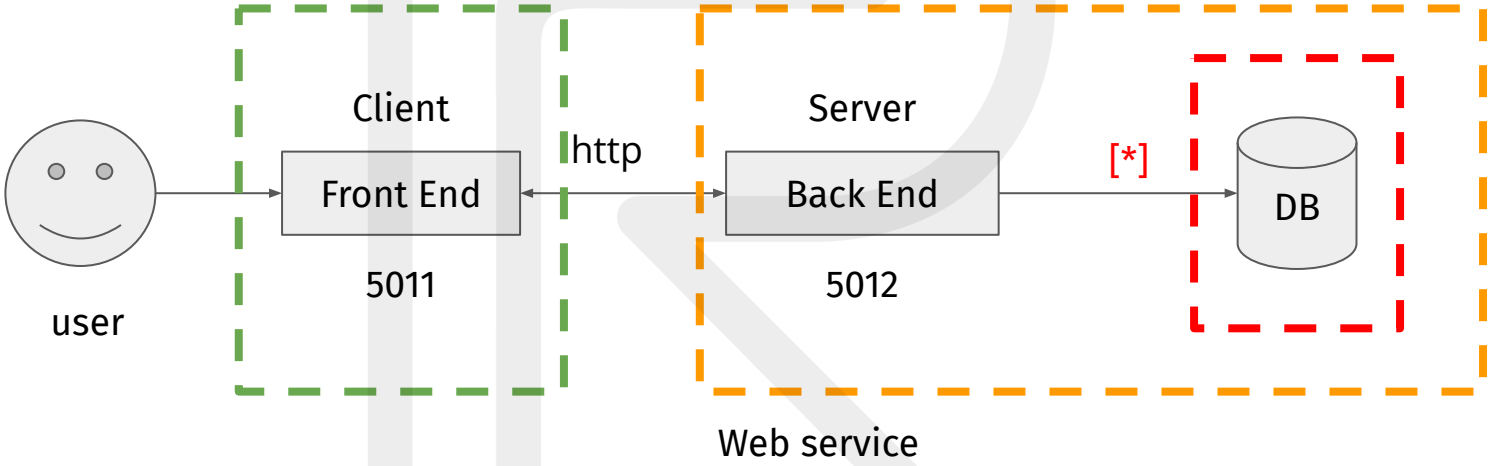
Meanwhile monolith ...



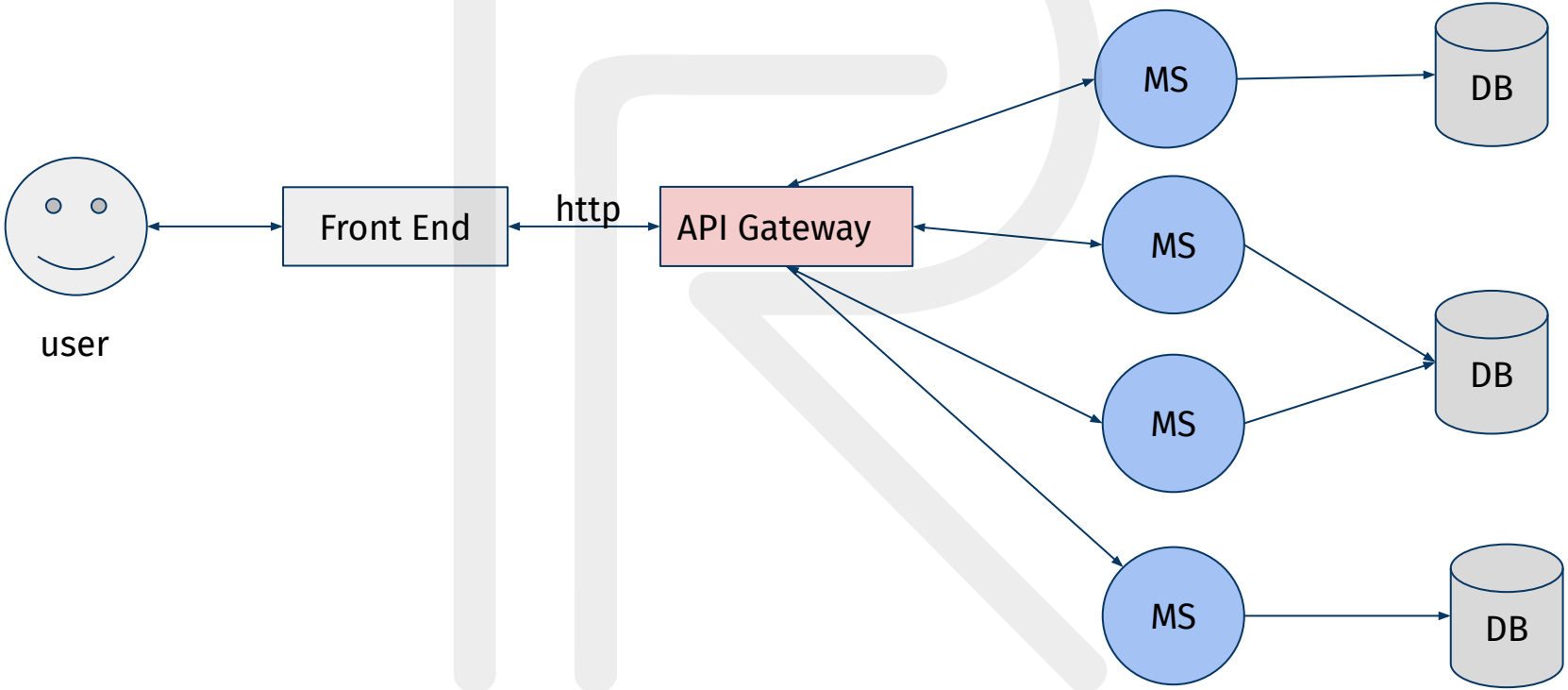
Meanwhile web service ...



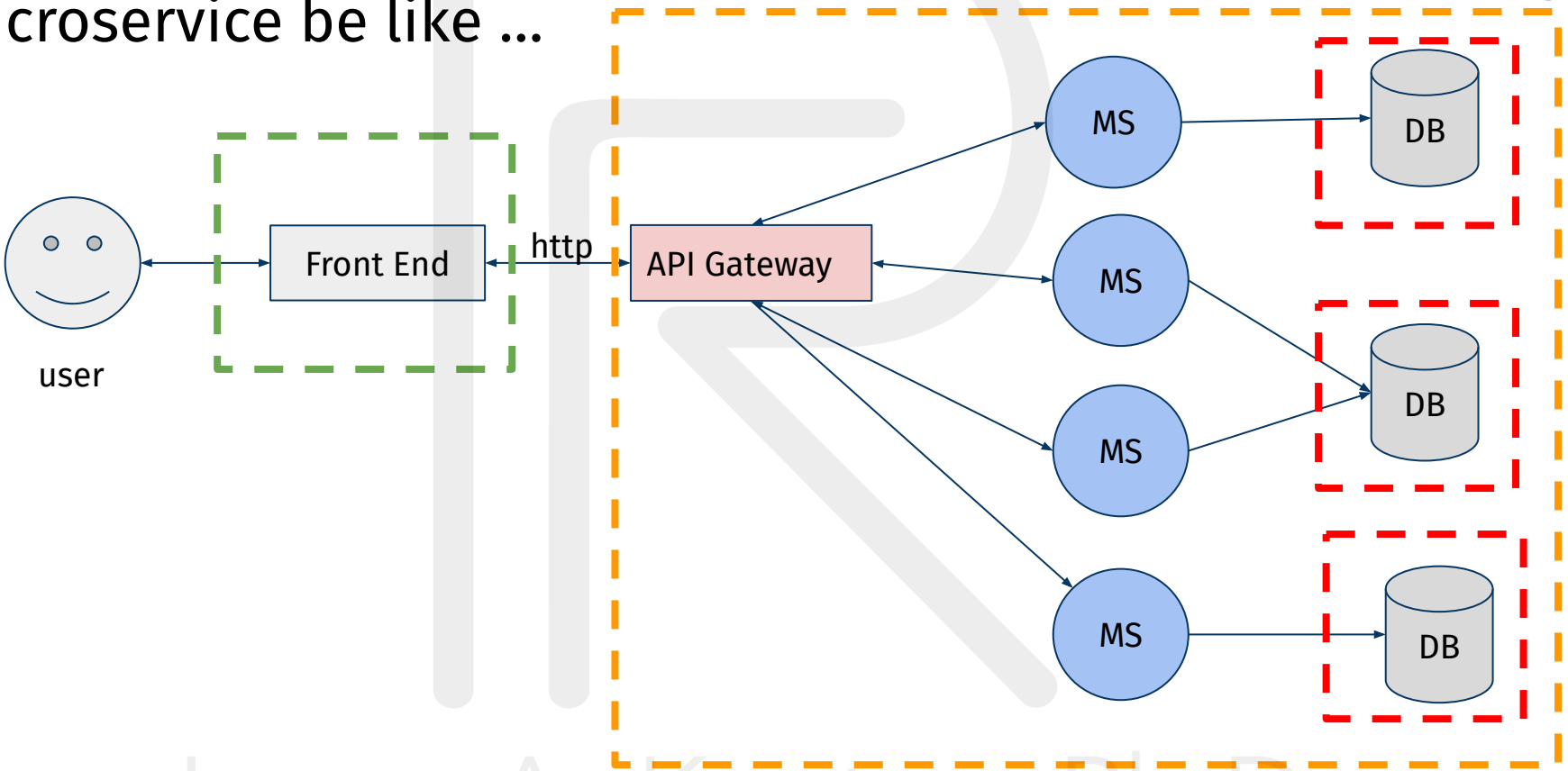
Meanwhile web service ...



Microservice be like ...

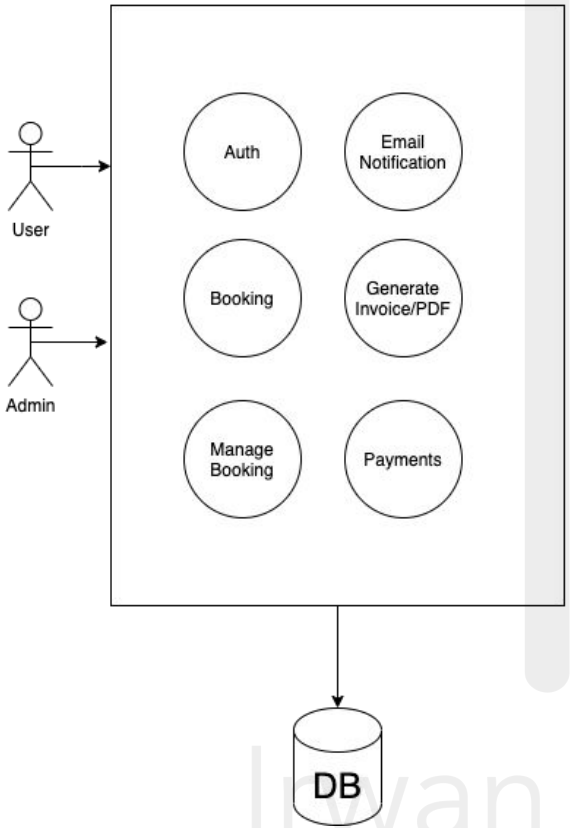


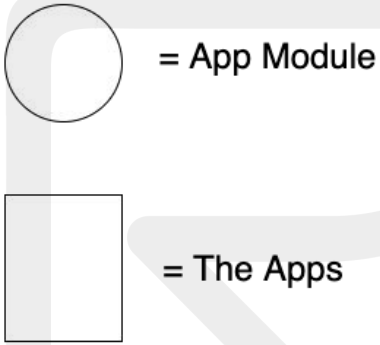
Microservice be like ...



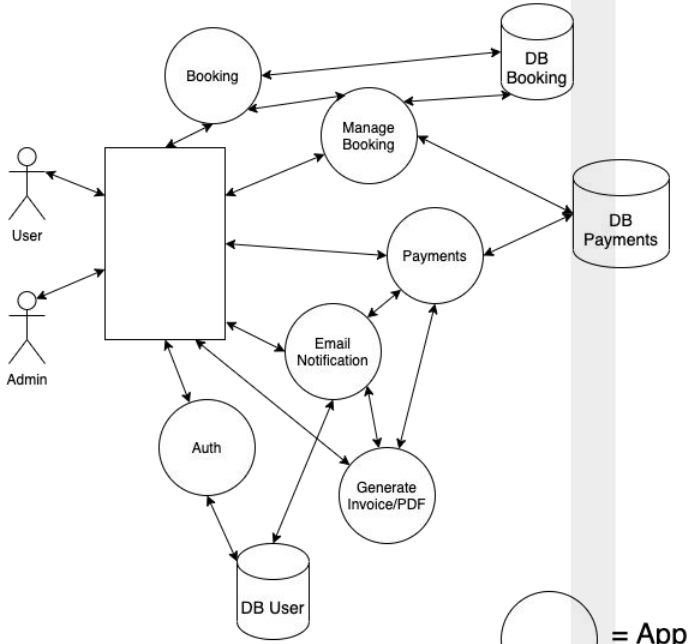
Another Example: Booking App

Meanwhile monolith ...





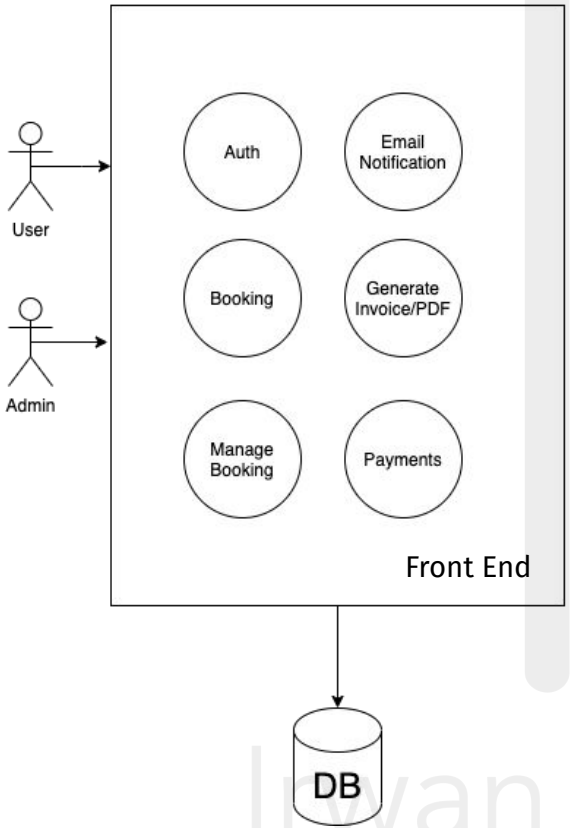
Microservice be like ...

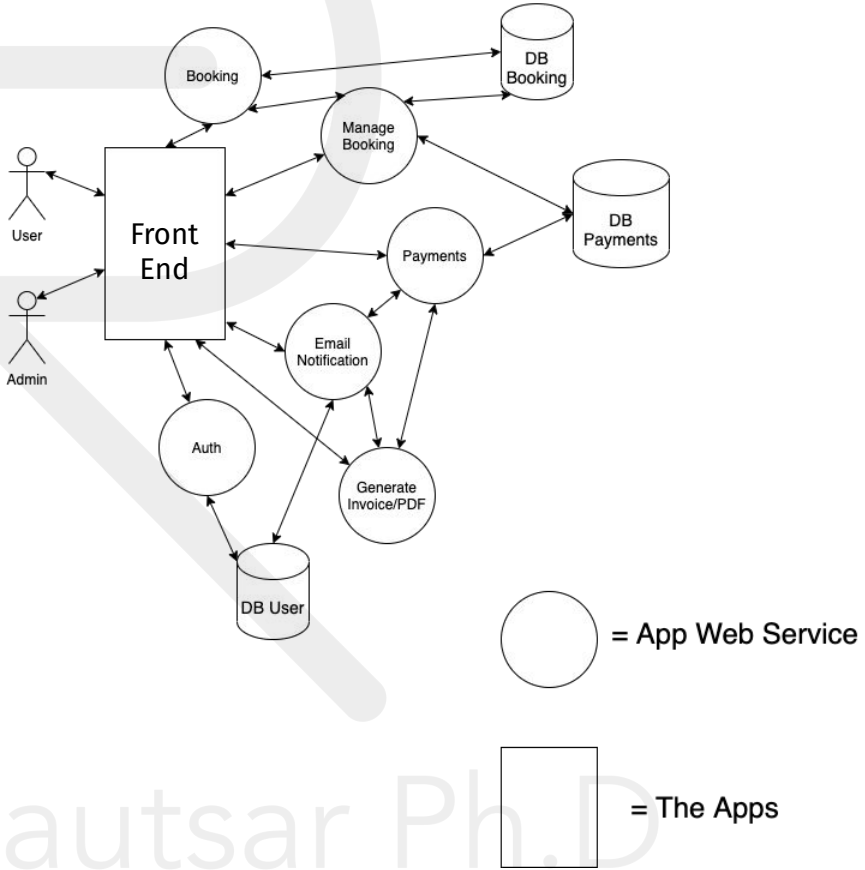


= App Web Service

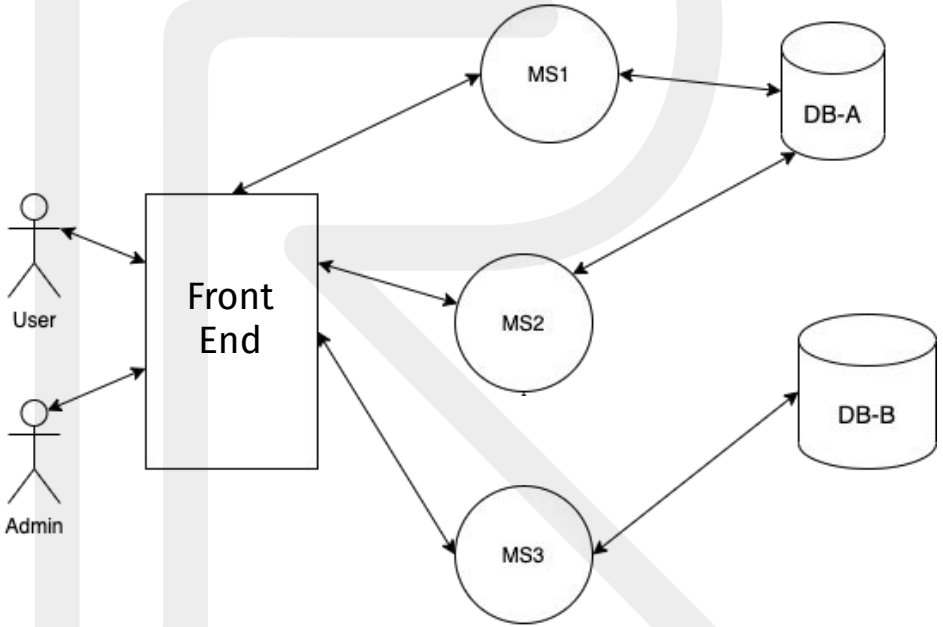
= The Apps

Monolith vs Microservice



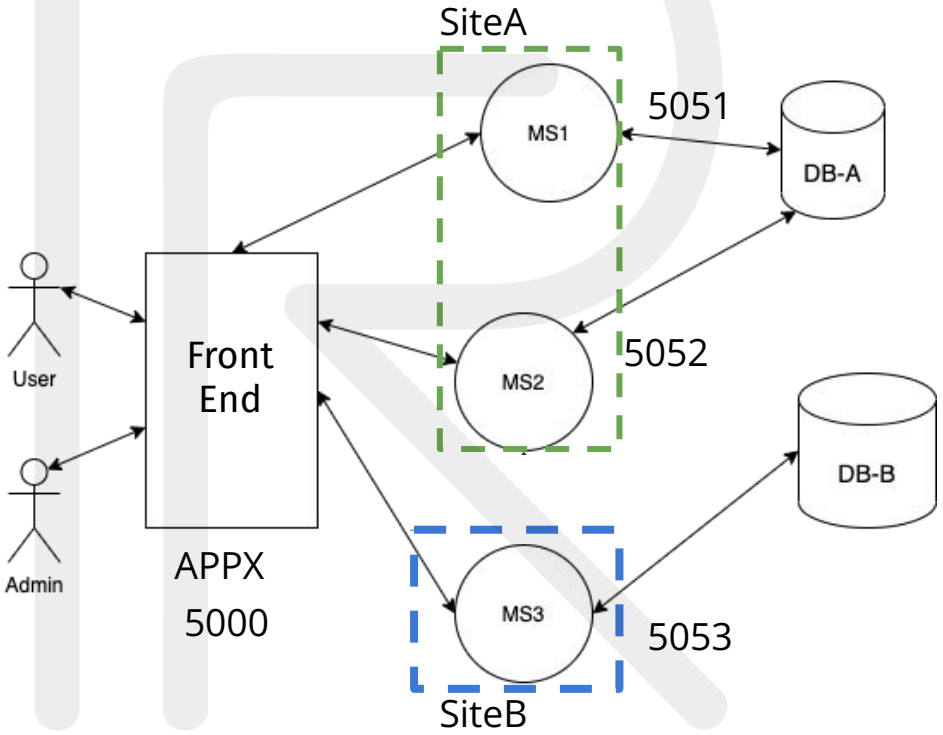


Let’s jump into code (Demos)



somedomain.com/ms1
somedomain.com/ms2
somedomain.com/ms3

Let’s jump into code (Demos)



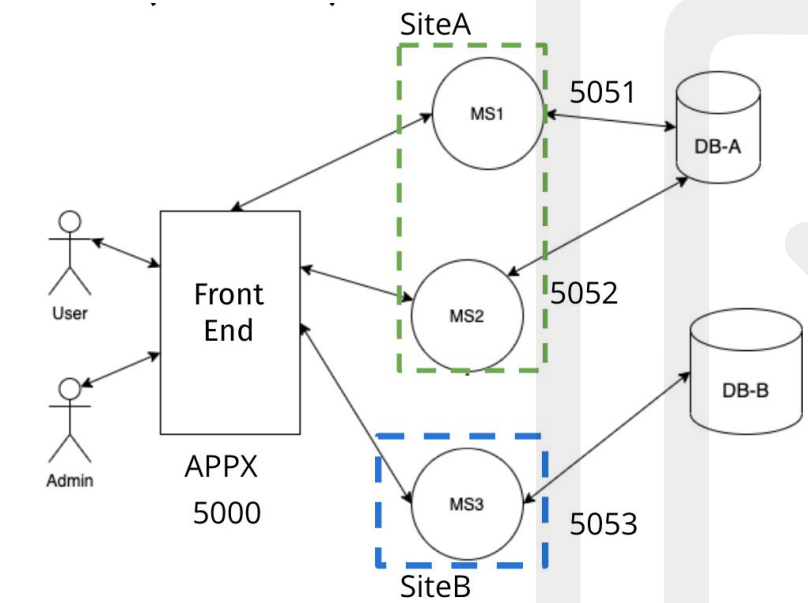
somedomain.com/ms1
somedomain.com/ms2
somedomain.com/ms3

Questions?

Assignments: PBL0901

1. Grab the code at: https://github.com/hepidad/c006-microservice
2. Push ⭐ at the repo for notification update
3. Complete **Microservice** CRUDS function (UPDATE and SEARCH) with Flash Framework

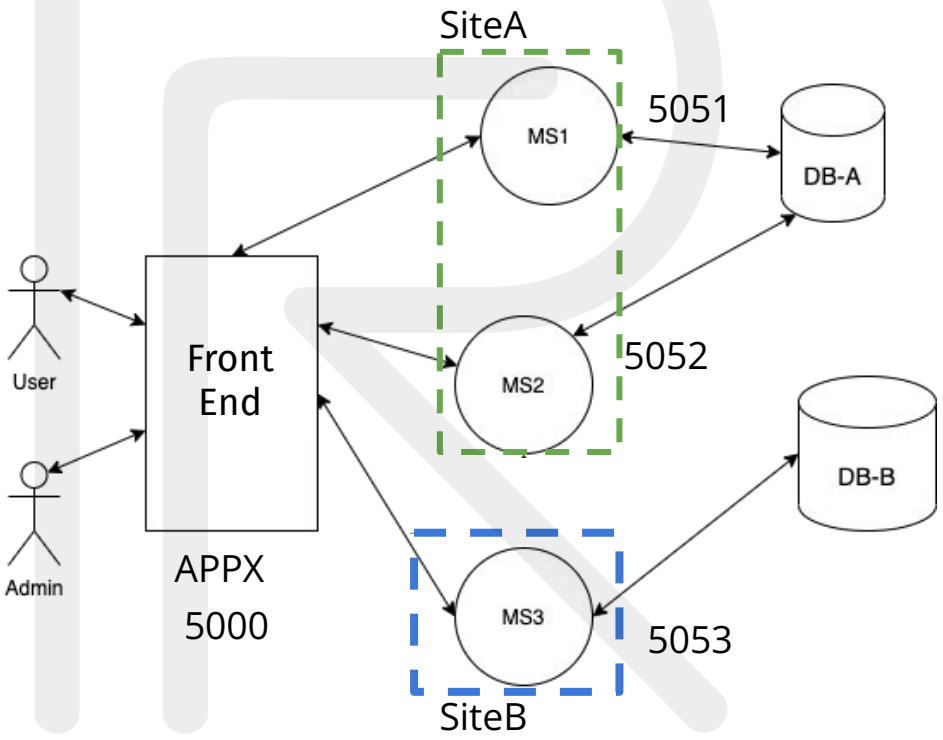
Assignments: PBL0902



somedomain.com/ms1
somedomain.com/ms2
somedomain.com/ms3

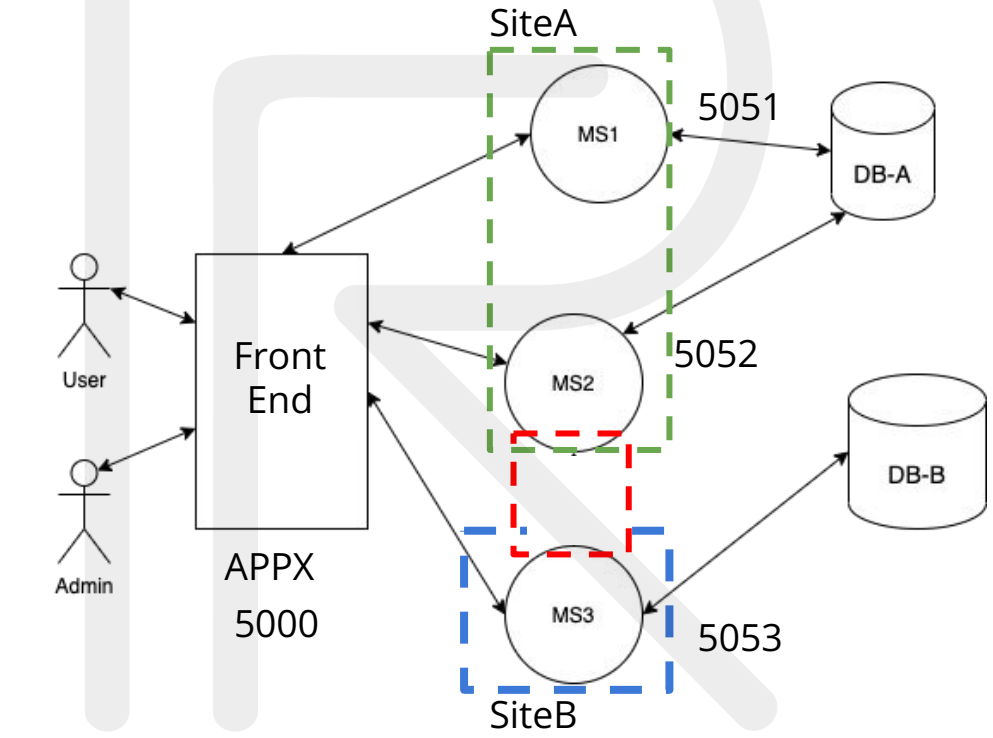
Implement CAR DATA CRUDS with **Microservice** from Chosen Frameworks (s.id/WebFrameworks) using **carsweb.db** from (https://github.com/hepidad/c006-microservice) as data source. Kindly use SQLite DB Engine. (Use Slide #16 for the architecture design)

Existing



somedomain.com/ms1
somedomain.com/ms2
somedomain.com/ms3

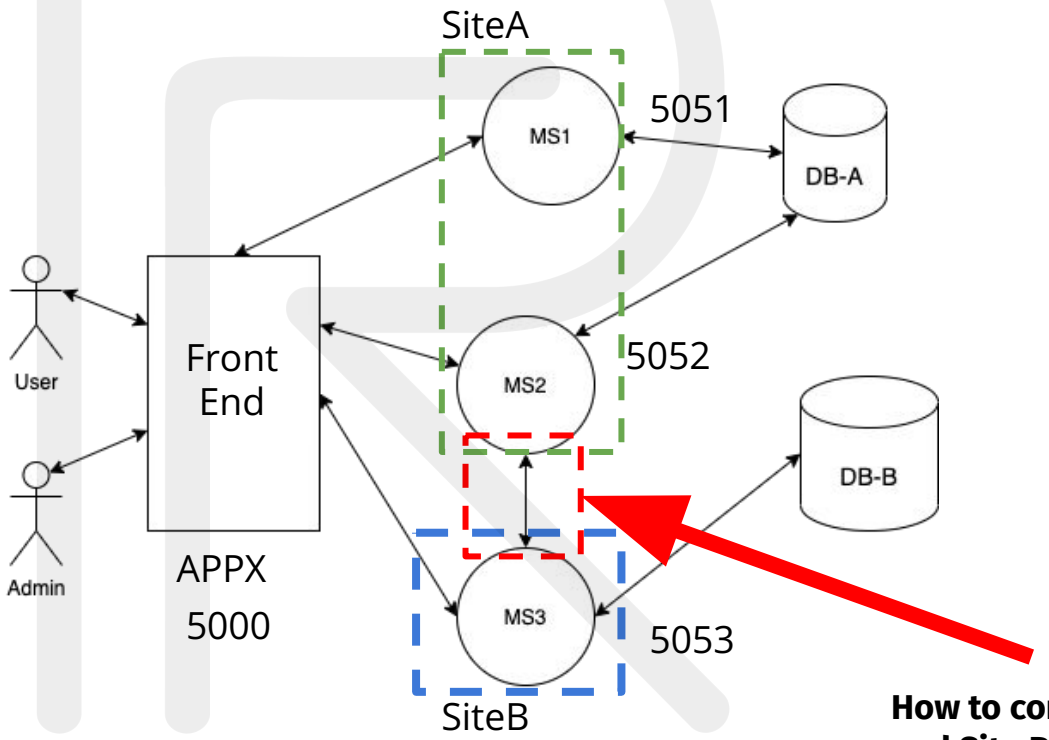
Don’t have



somedomain.com/ms1
somedomain.com/ms2
somedomain.com/ms3

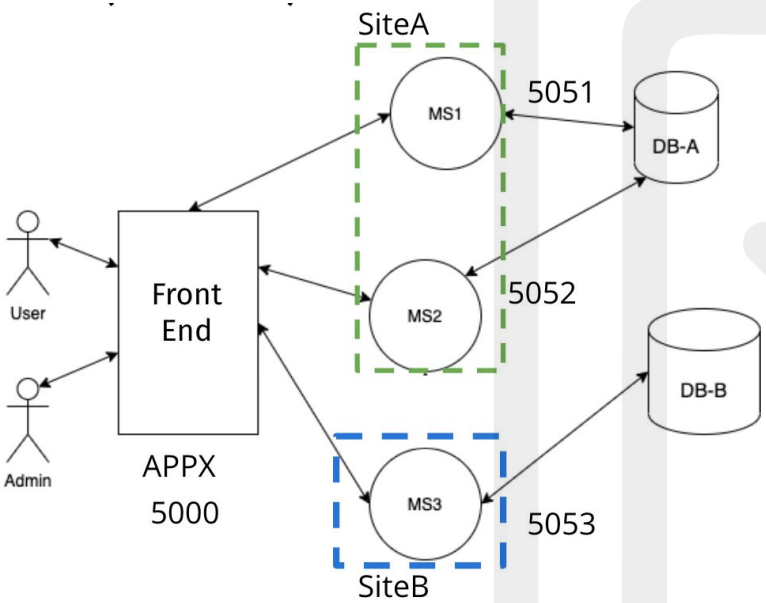
Challenge

1. Entry Via MS1 → DB-A
2. Entry Via MS2 → DB-A + DB-B
3. Entry Via MS3 → DB-B + DB-A



somedomain.com/ms1
somedomain.com/ms2
somedomain.com/ms3

Assignments: PBL0903



somedomain.com/ms1
somedomain.com/ms2
somedomain.com/ms3

Complete the challenge with chosen Framework (Use Slide #16 for the architecture design)