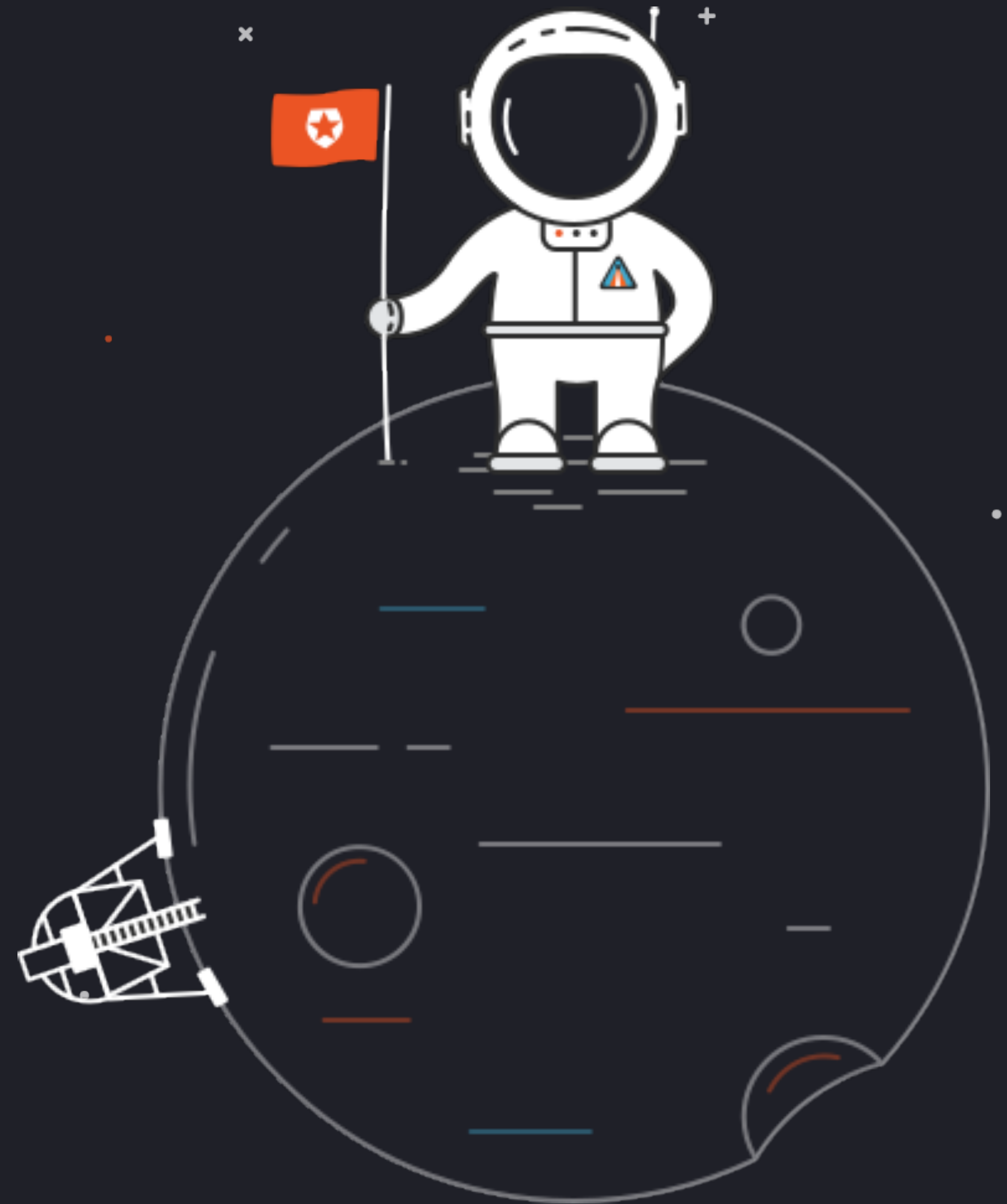


# Serverless in Deep





Mercedes Wyss  
@itrjwyss



**Community Leader**  
Devs+502 & JDuchess Chapter Guatemala

**Ex-JUG Member**  
Guatemala Java Users Group (GuateJUG)

**Chief Technology Officer (CTO) at Produactivity**  
Full Stack Developer

**Auth0 Ambassador &  
Oracle Developer Champion**



**ORACLE®**  
Developer  
Champion



- **About Serverless**
- Function as a Service - FaaS
- Serverless Architecture
- Benefits and Drawbacks
- Design Patterns and Use Cases
- Demos
- FNProject

“**Serverless architectures** refer to applications that significantly **depend** on third-party services (known as Backend as a Services - **BaaS**) or on custom code that’s **run** in **ephemeral containers** (Function as a Service - **FaaS**)”

MartinFowler.com





# Backend as a Service

- Applications that significantly or fully depend on 3rd party applications / services (“in the cloud”) to manage server-side logic and state.
- Cloud accessible databases (Parse, Firebase)
- Authentication Services (Oracle Identity Cloud Service, Auth0, Amazon Cognito)

# Functions as a Service

- Run in stateless compute containers that are event-triggered
- Ephemeral
- Fully managed by a 3rd party
- AWS Lambda, Google Cloud Functions, Firebase Functions, Azure Functions, FNPProject



# Key Characteristics of a Serverless Application

- No server management
- Flexible scaling
- High availability
- Never pay for idle  
(Integrated Development Environment)



Servers are  
fully-abstracted



Scaling is event-driven  
*not resource-driven*



Pay only  
for what you use





# What is not Serverless?

- Platform as a Service (PaaS)
- Containers
- #NoOps
- Stored Procedures as a Service

## Pre-Cloud B.Y.O. Servers



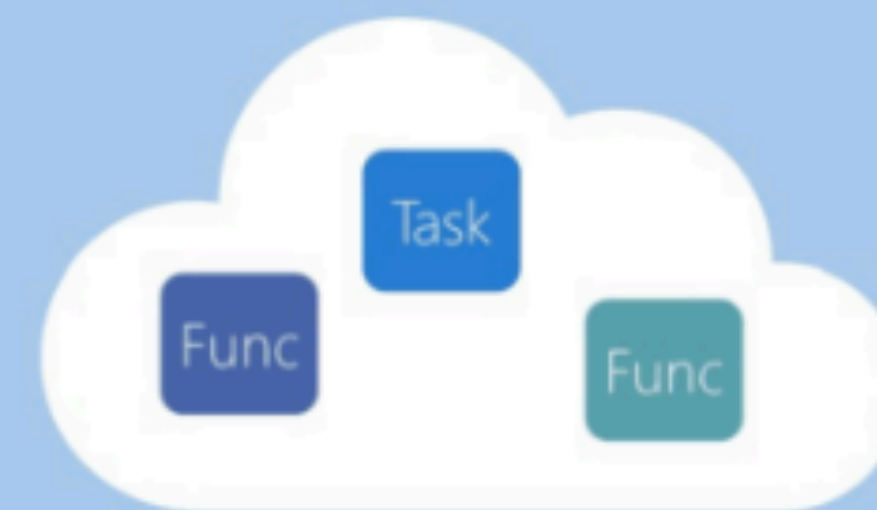
## IaaS

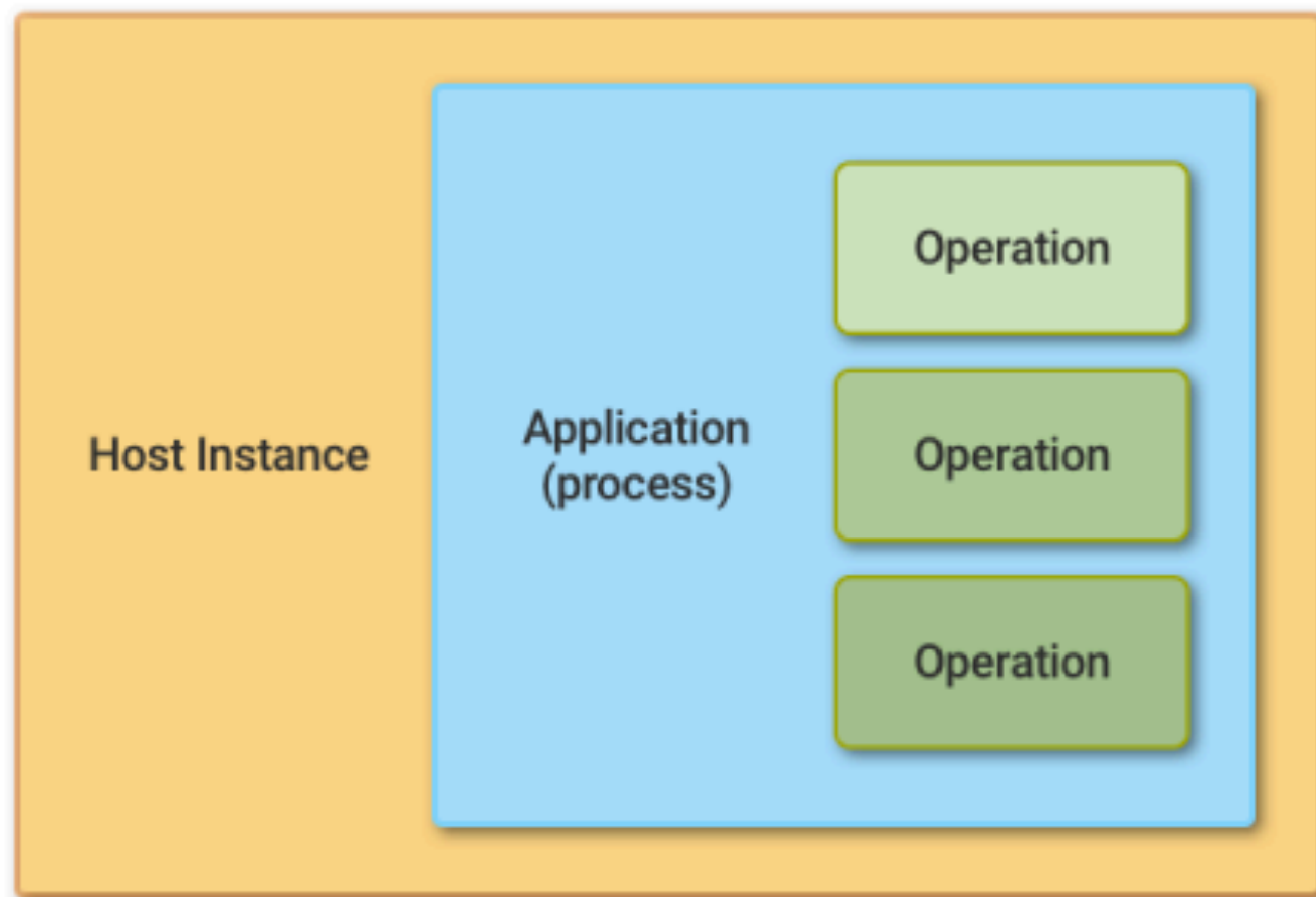


## PaaS



## "Serverless"





**ORACLE®**  
Developer  
Champion



- About Serverless
- **Function as a Service - FaaS**
- Serverless Architecture
- Benefits and Drawbacks
- Design Patterns and Use Cases
- Demos
- FNProject



# Function as a Service

- Serverless computing via Serverless architectures.
- Deploy an individual “function”, action, or piece of business logic.
- Event-driven processing part of the serverless architecture.



# Principles of FaaS

- Complete abstraction of servers away from the developer.
- Billing based on consumption and executions, not server instance sizes.
- Services that are event-driven and instantaneously scalable.

# FaaS in Terms of a Cloud Platform

- Run code without provisioning or managing servers.
- We can run code for virtually any type of application or backend service.
- Zero administration. Just upload the code, and we will run.
- And scale.
- Code with high availability, automatically trigger from other services.
- Can call it directly from any web or mobile app.



# FaaS State

- Are Stateless.
- Provide pure functional transformations of their input.

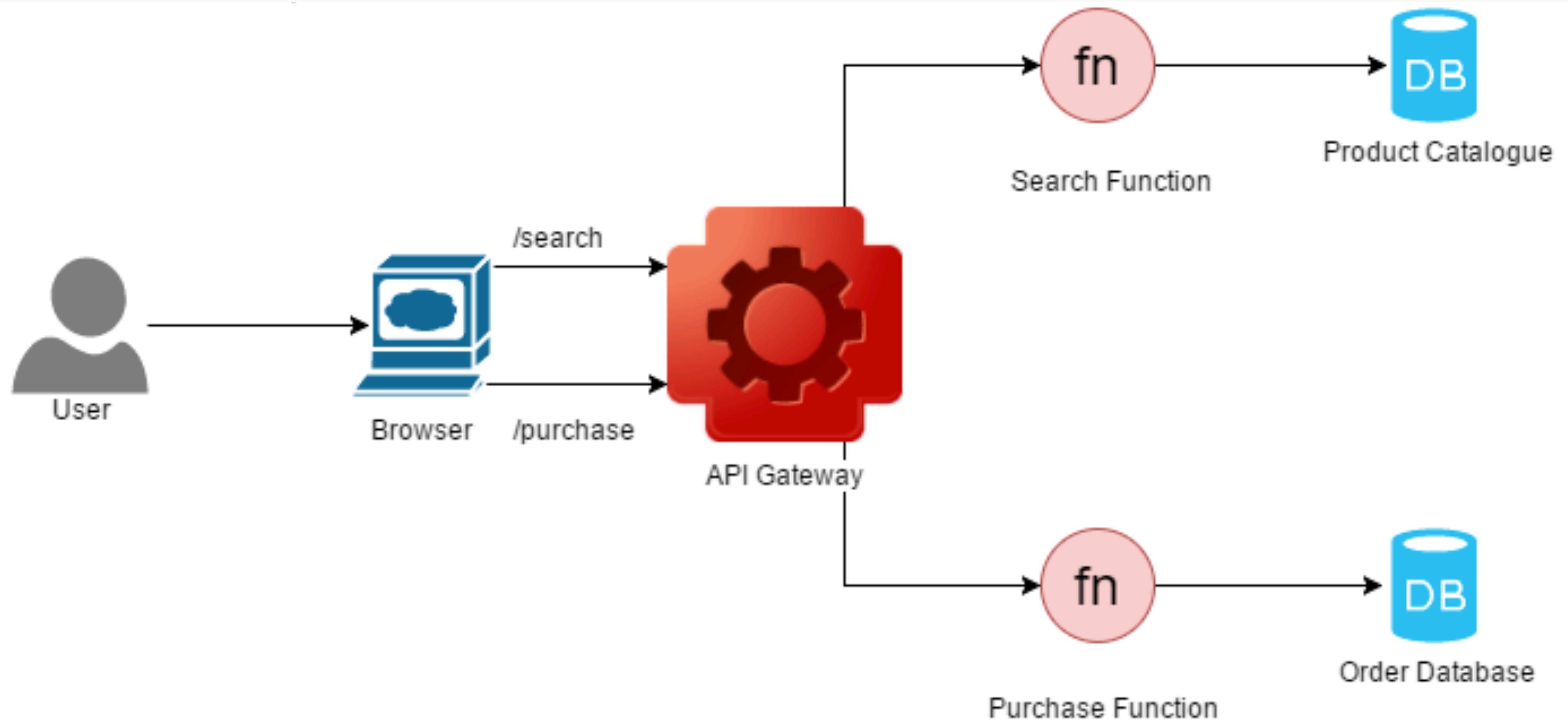
# FaaS Some Restrictions

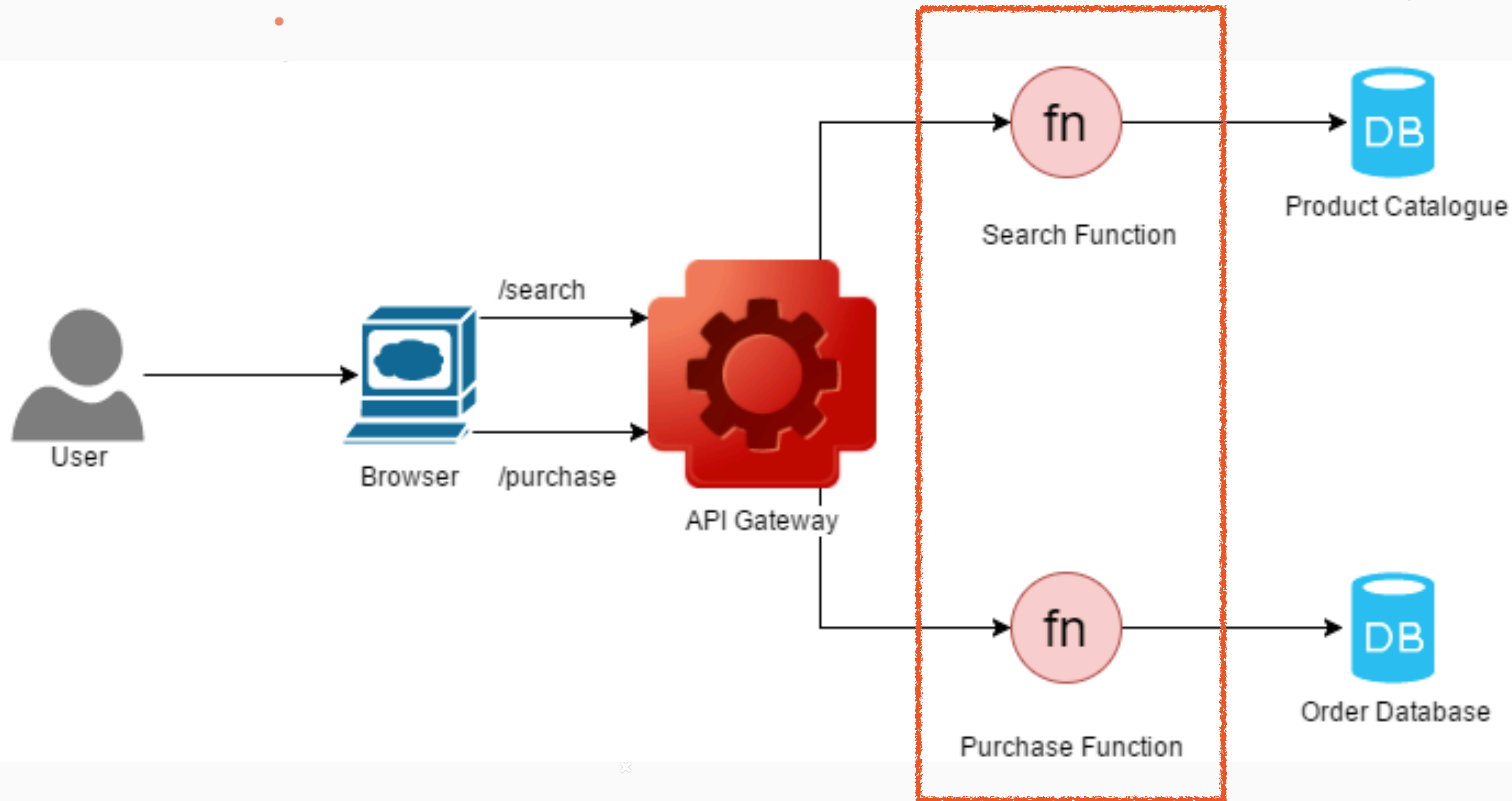
- FaaS functions are typically limited in how long each invocation is allowed to run.
- Which programming languages can be used.
- All the architecture need to be in the same cloud.

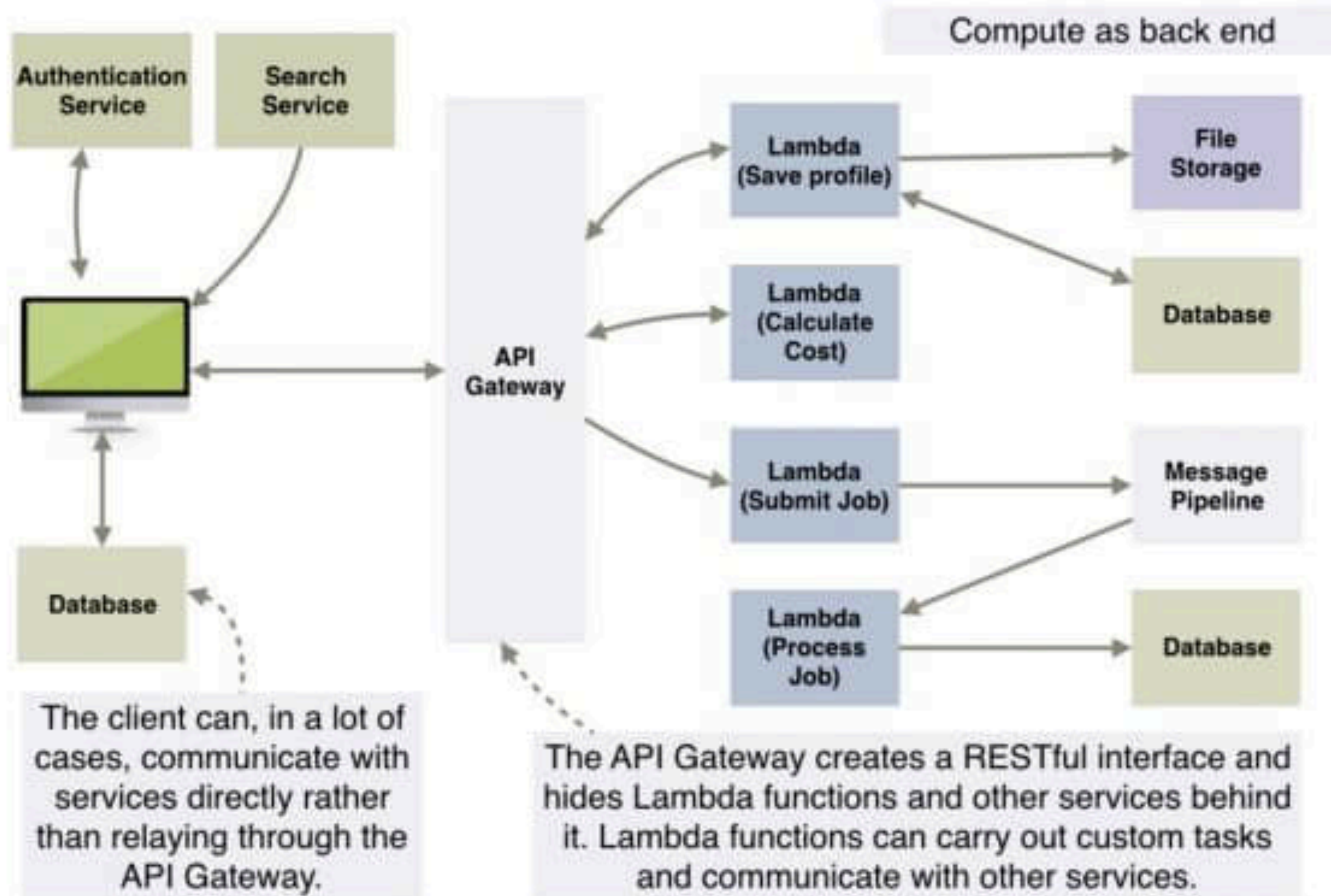




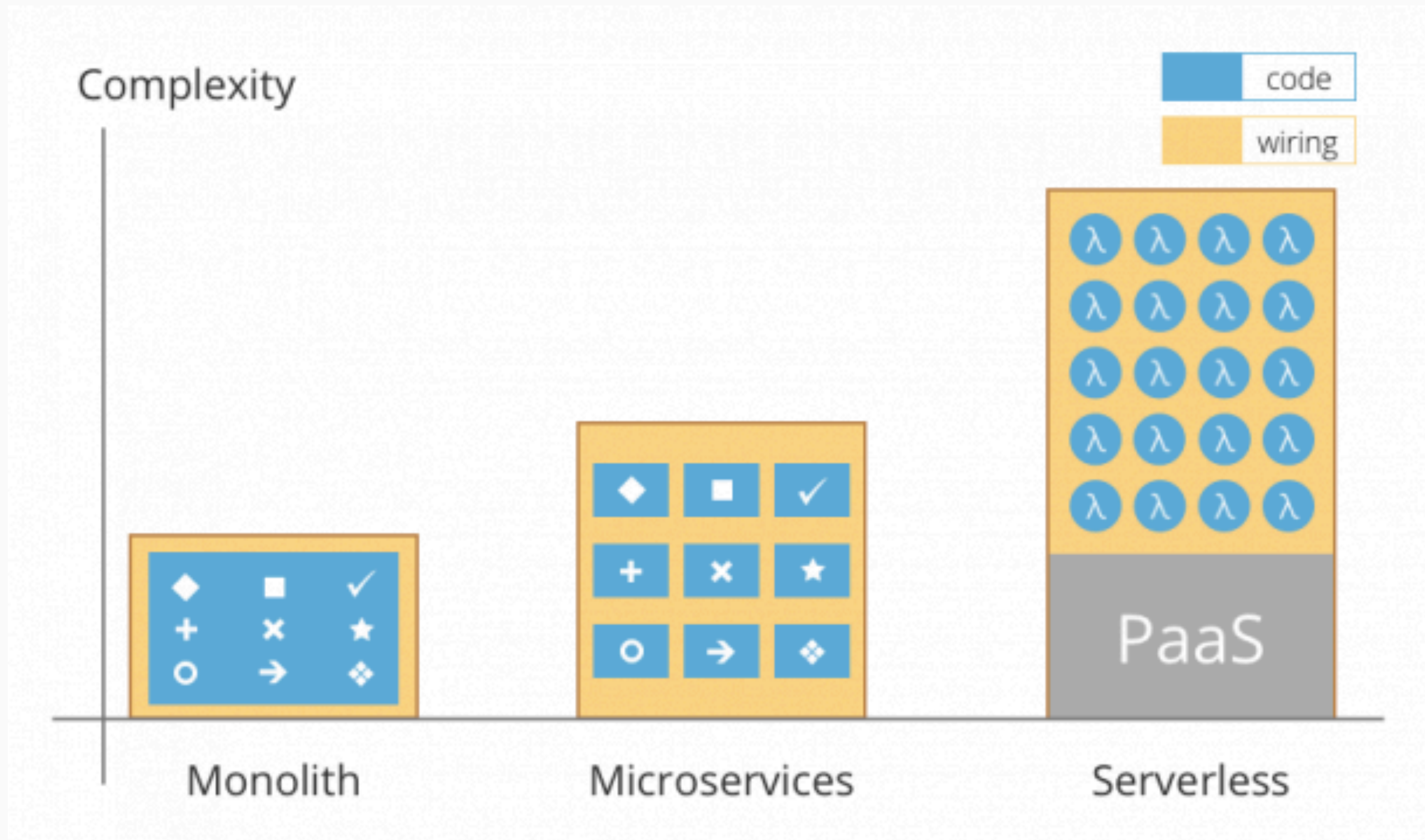
- About Serverless
- Function as a Service - FaaS
- **Serverless Architecture**
- Benefits and Drawbacks
- Design Patterns and Use Cases
- Demos
- FNProject













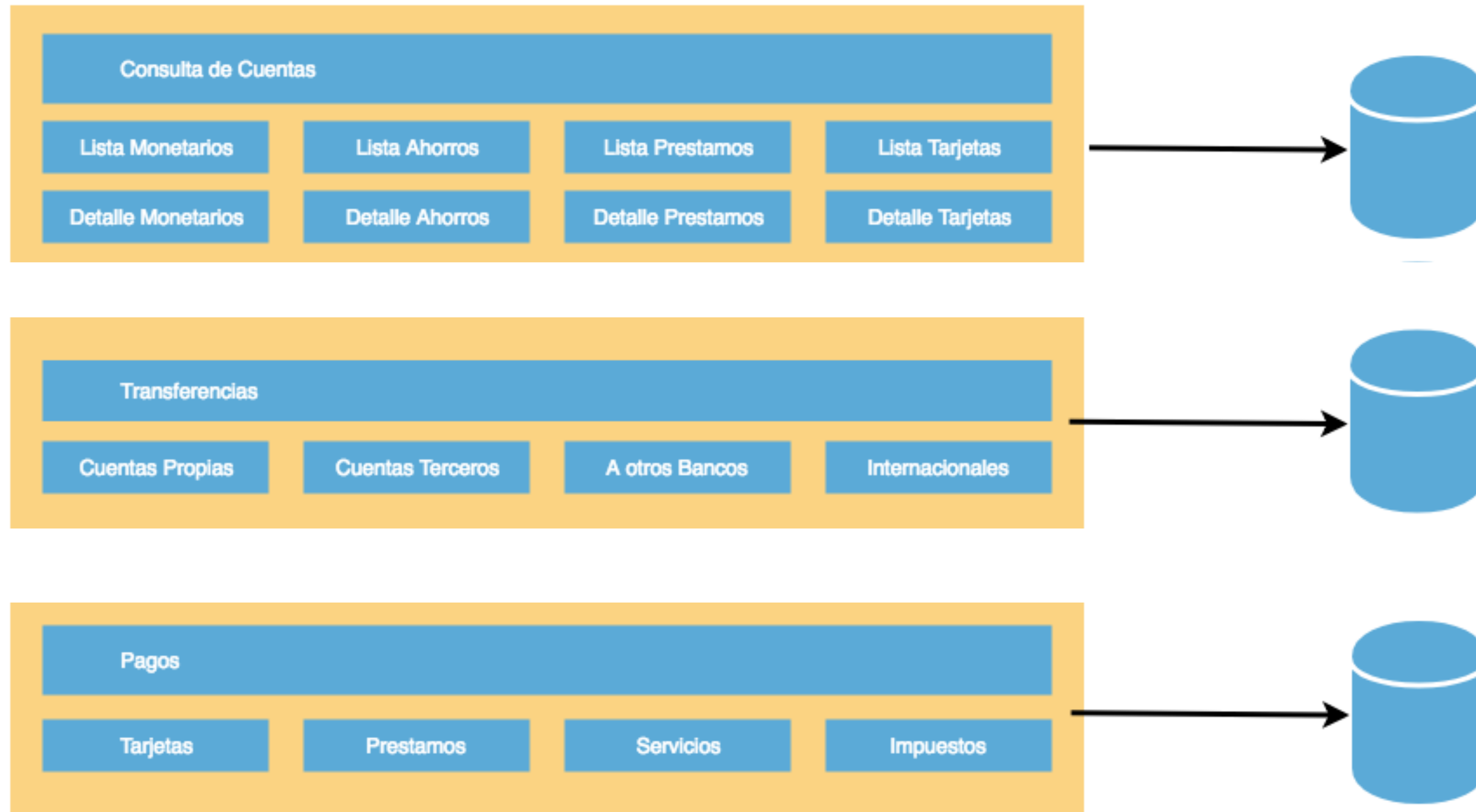
# Implementación Bancaria

*Este caso es solo para ejemplificar la diferencia entre Monolitos, Arquitectura de Microservicios y Serverless. No es una sugerencia de implementación.*

# Monolito



# Microservicios



**ORACLE®**  
Developer  
Champion

# FaaS

Lista Monetarios

Lista Ahorros

Detalle Monetarios



- About Serverless
- Function as a Service - FaaS
- Serverless Architecture
- **Benefits and Drawbacks**
- Design Patterns and Use Cases
- Demos
- FNProject



# Benefits

- Time-to-market Improvement
- Reduced Operational Cost
- Infrastructure Cost Reduction  
(FaaS scaling cost)
- BaaS - reduced development cost
- ✕ • Easier Operational Management



# Drawbacks

- Problems due to Third-party API system
- Lack of operational tools
- Architectural complexity
- Monitoring Challenges
- Implementation drawbacks

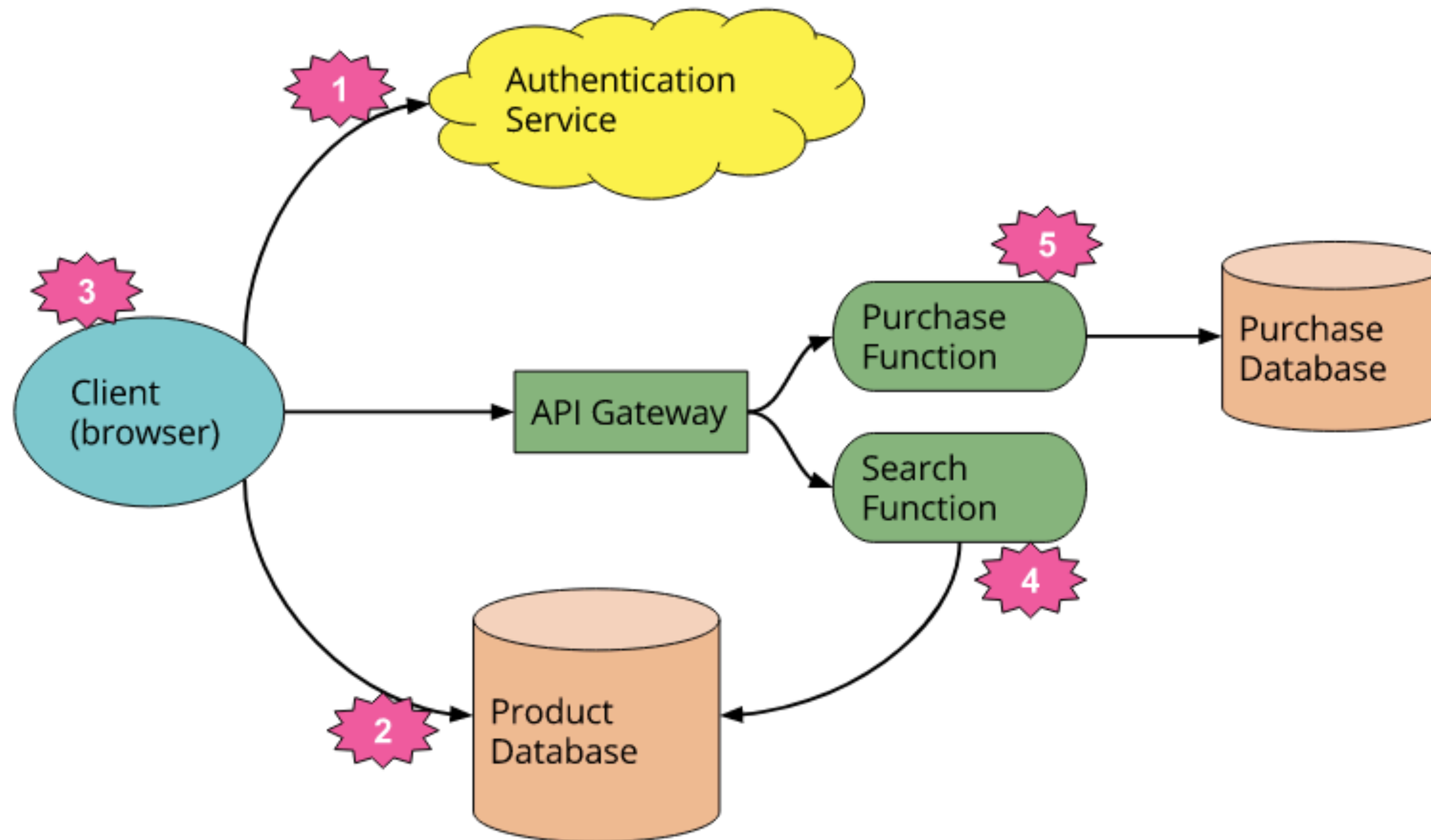
- About Serverless
- Function as a Service - FaaS
- Serverless Architecture
- Benefits and Drawbacks
- **Design Patterns and Use Cases**
- Demos
- FNProject



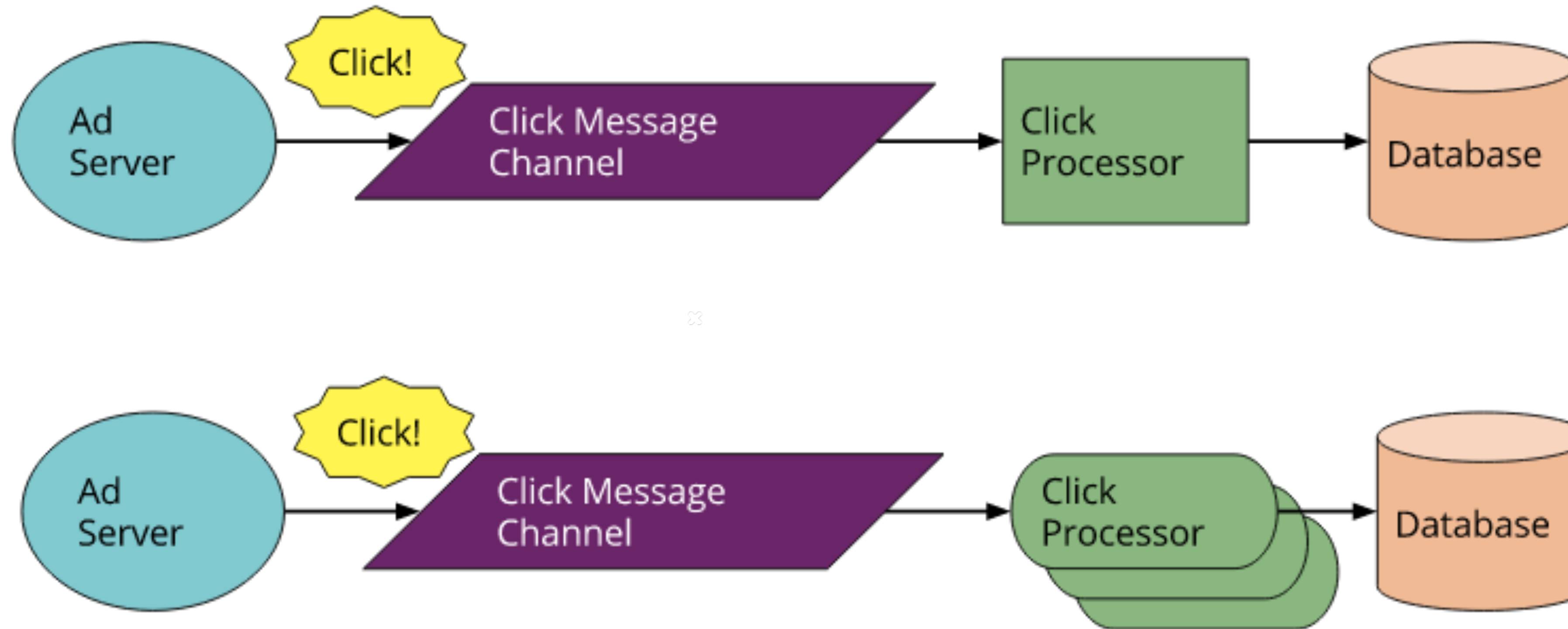
# Example 1: UI-driven applications



# Example 1: UI-driven applications



# Example 2: Message-driven applications



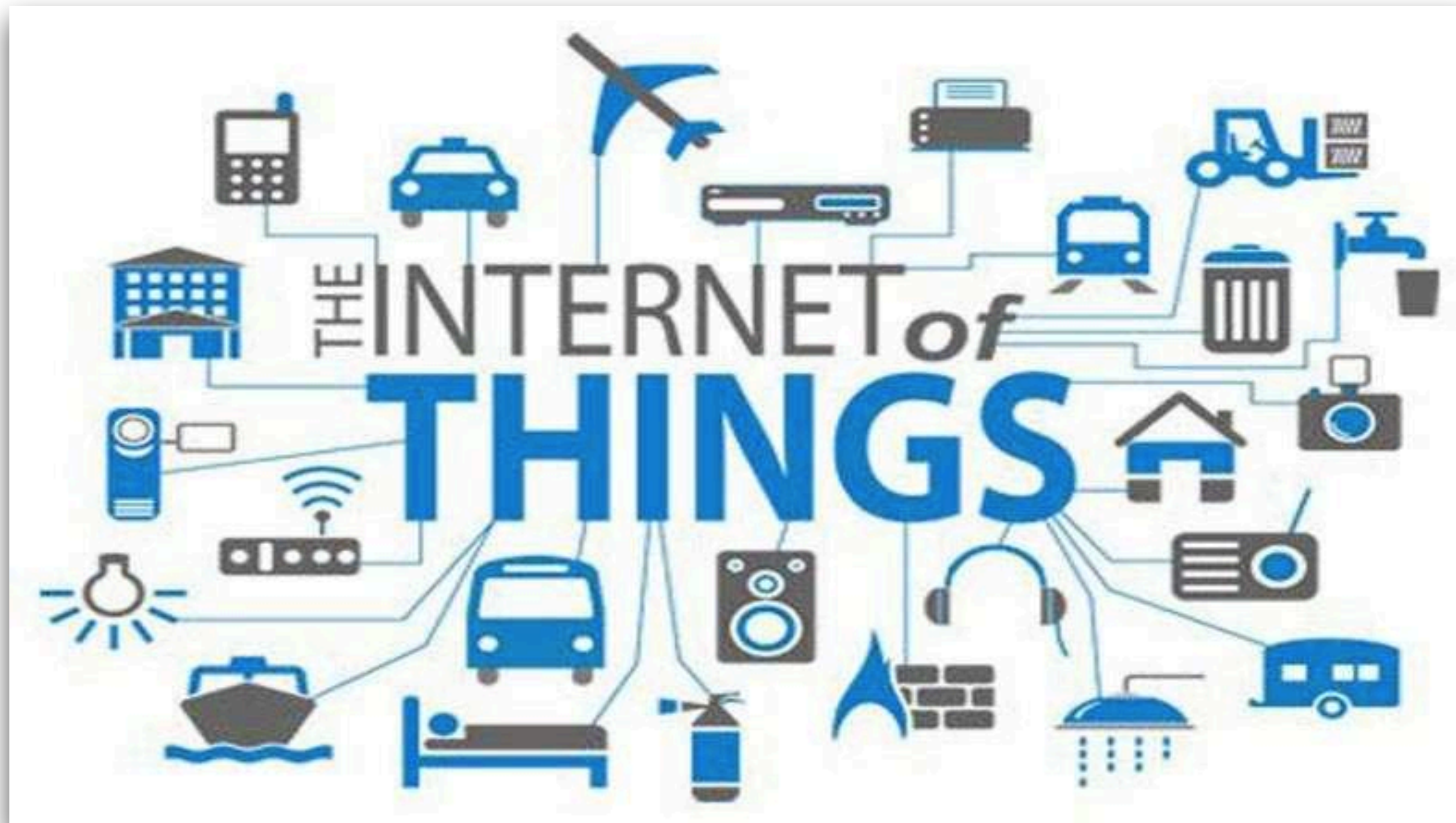


# Example 3: Data Processing

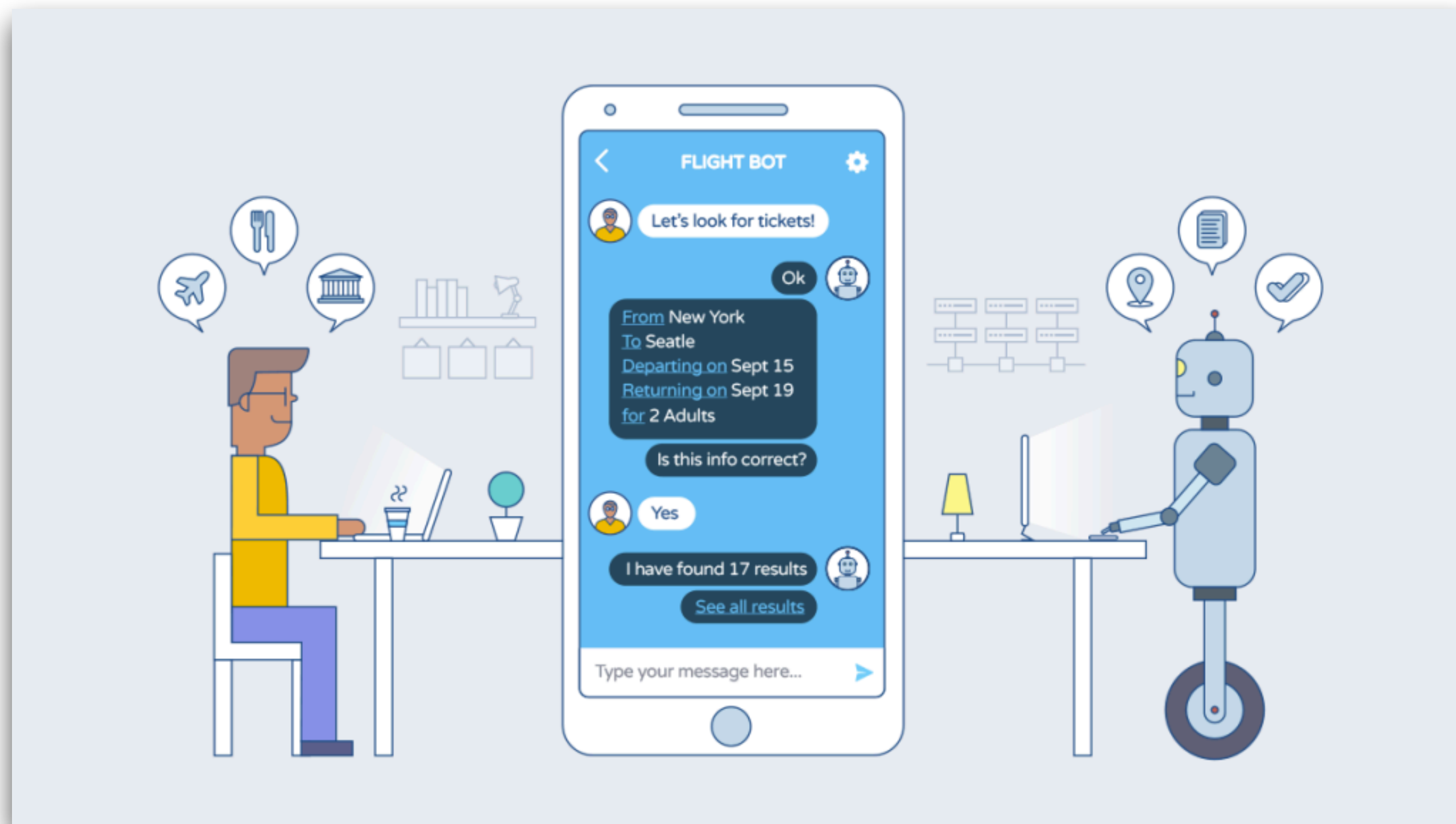




# Example 4: Internet of Things



# Example 5: Chatbots





**ORACLE®**  
Developer  
Champion



- About Serverless
- Function as a Service - FaaS
- Serverless Architecture
- Benefits and Drawbacks
- Design Patterns and Use Cases
- **Demos**
- FNProject



# Serverless Function



webtask



MySQL®



# Look at the video DBCreation.mp4

Identity Domain mmwyss - North America

**Guided Journey**  
Explore what you can do with Oracle Cloud services

**Create Instance**  
Provision a new service in minutes

**Account Management**  
Administer and manage your account and orders

**Customize Dashboard**  
Specify which services appear on the dashboard

## Cloud Services

0 Important Notifications

Promotion

4,945.42 of 5,000 USD  
Remaining (306 days left)



You currently have no services shown

Services with instances are automatically shown. Click on Create Instance to add an instance to a service. Otherwise, click on Customize Dashboard to view the list of all services you have access to, and to update shown services.



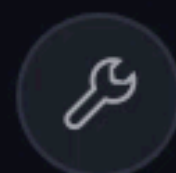
# Look at the video ReadDB.mov



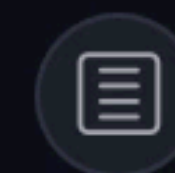
There is no open Webtask

⌘ P to Quick Search in your webtasks  
or you can [Create a new one](#)


# Look at the video WriteDB.mov



addRecipe >



```
2
3  /**
4   * @param context {WebtaskContext}
5   */
6  module.exports = function(context, callback) {
7    connection = mysql.createConnection({
8      host      : context.secrets.HOST,
9      user      : context.secrets.USER,
10     password   : context.secrets.PASSWORD,
11     database   : context.secrets.DB
12   });
13
14   connection.connect();
15
16   connection.query(
17     'SELECT * FROM recipes ORDER BY recipe_id LIMIT 100',
18     callback);
19  };
```

Powered by  Extend



0.43 Kb out of 100 kb

<https://wt-5284972561f7f3b5d409c4fc3df66c71-0.sandbox.auth0-extend.com/addRecipe>



In 7, col 3

- About Serverless
- Function as a Service - FaaS
- Serverless Architecture
- Benefits and Drawbacks
- Design Patterns and Use Cases
- Demos
- **FNProject**



# FNProject

- Multi Cloud
- Developer Experience
- Container Native  
(Docker, Kubernetes)
- Vision and Deep

# FNProject

- Fn Load Balancer
- Fn Server (FaaS)
- Fn FDK's
- Fn Flow





<https://github.com/itrjwyss/Journey18>

<https://www.facebook.com/itrjwyss>

**@itrjwyss**

