

# zakia-regoug-enset-adria-test

## Problème :

On souhaite développer un POC (Proof Of Concept) sous forme d'une application web basée sur une architecture micro-service an utilisant l'écosystème Spring Boot et Spring Cloud coté backend et un Framework comme Angular ou React coté Frontend.

L'application doit permettre de gérer des porte feuilles (Wallets) de devises ainsi que les transferts entre les

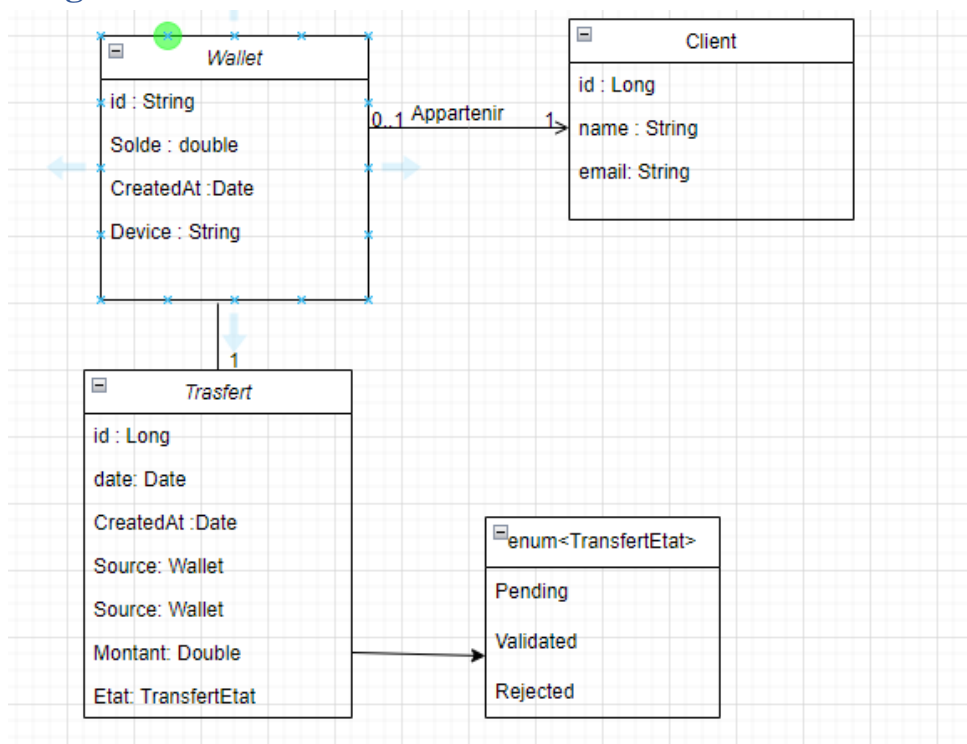
portes feuilles. Pour faire plus simple, cette application se composera de deux micro-services fonctionnels :

- **Un Micro-service** qui permet de gérer des « **Wallet** ». Chaque Wallet appartient à un client. Un client est défini par son id (numérique auto incrémenté), son nom et son email. Une Wallet est définie par son id (String de type UUID), son solde, sa date de création, sa devise.
- **Un Micro-service** qui permet de gérer des transferts entre les **Wallets**. Chaque transfert est défini par son id, sa date, Wallet source, Wallet destination, son montant, son état (PENDIND, VALIDATED, REJECTED)

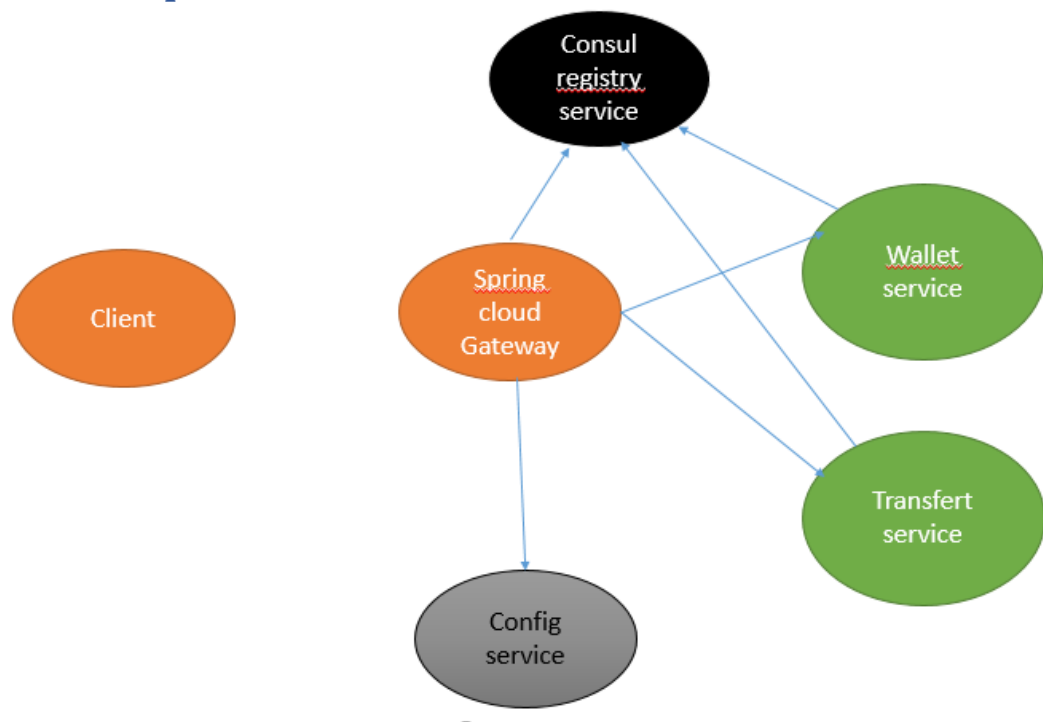
Les micro-services technique à mettre en place sont :

- Le service Gateway basé sur Spring cloud Gateway
- Le service Discovery base sur Eureka Server ou Consul Discovery (au choix)
- Le service de configuration basé sur Spring cloud config ou Consul Config (au choix)

## Diagramme de classe :



## Architecture Technique :



## Creation des microservices :

### 1- Wallet service

<b>Project</b> <input type="radio"/> Gradle - Groovy <input type="radio"/> Gradle - Kotlin <input checked="" type="radio"/> Maven	<b>Language</b> <input checked="" type="radio"/> Java <input type="radio"/> Kotlin <input type="radio"/> Groovy	<b>Dependencies</b> <a href="#">ADD DEPENDENCIES...</a> CTRL + B
<b>Spring Boot</b> <input type="radio"/> 3.2.0 (SNAPSHOT) <input type="radio"/> 3.2.0 (RC1) <input type="radio"/> 3.1.6 (SNAPSHOT) <input checked="" type="radio"/> 3.1.5 <input type="radio"/> 3.0.13 (SNAPSHOT) <input type="radio"/> 3.0.12 <input type="radio"/> 2.7.18 (SNAPSHOT) <input type="radio"/> 2.7.17		<b>Spring Web</b> <b>WEB</b> Build web, including RESTful, applications using Spring MVC. Uses Apache Tomcat as the default embedded container.
<b>Project Metadata</b> Group <input type="text" value="org.sid"/> Artifact <input type="text" value="wallet-service"/> Name <input type="text" value="wallet-service"/> Description <input type="text" value="Demo project for Spring Boot"/>		<b>Spring Data JPA</b> <b>SQL</b> Persist data in SQL stores with Java Persistence API using Spring Data and Hibernate.
		<b>H2 Database</b> <b>SQL</b> Provides a fast in-memory database that supports JDBC API and R2DBC access, with a small (2mb) footprint. Supports embedded and server modes as well as a browser based console application.
		<b>Lombok</b> <b>DEVELOPER TOOLS</b> Java annotation library which helps to reduce boilerplate code.

### 2- Transfert service

Project

☐ Gradle - Groovy
☐ Gradle - Kotlin
☒ Maven

Language

☒ Java
☐ Kotlin
☐ Groovy

Spring Boot

☐ 3.2.0 (SNAPSHOT)
☐ 3.2.0 (RC1)
☐ 3.1.6 (SNAPSHOT)
☒ 3.1.5
☐ 3.0.13 (SNAPSHOT)
☐ 3.0.12
☐ 2.7.18 (SNAPSHOT)
☐ 2.7.17

Project Metadata

Group

org.sid

Artifact

transfert-service

Name

transfert-service

Description

Demo project for Spring Boot

Package name

org.sid.transfert-service

Packaging

☒ Jar
☐ War

Dependencies

ADD DEPENDENCIES... CTRL + B

Spring Web

WEB

Build web, including RESTful, applications using Spring MVC. Uses Apache Tomcat as the default embedded container.

Spring Data JPA

SQL

Persist data in SQL stores with Java Persistence API using Spring Data and Hibernate.

H2 Database

SQL

Provides a fast in-memory database that supports JDBC API and R2DBC access, with a small (2mb) footprint. Supports embedded and server modes as well as a browser based console application.

Lombok

DEVELOPER TOOLS

Java annotation library which helps to reduce boilerplate code.

Rest Repositories

WEB

Exposing Spring Data repositories over REST via Spring Data REST.

### 3- Config service :

Project

☐ Gradle - Groovy
☐ Gradle - Kotlin
☒ Maven

Language

☒ Java
☐ Kotlin
☐ Groovy

Spring Boot

☐ 3.2.0 (SNAPSHOT)
☐ 3.2.0 (RC1)
☐ 3.1.6 (SNAPSHOT)
☒ 3.1.5
☐ 3.0.13 (SNAPSHOT)
☐ 3.0.12
☐ 2.7.18 (SNAPSHOT)
☐ 2.7.17

Project Metadata

Group

org.sid

Artifact

config-service

Name

config-service

Dependencies

ADD DEPENDENCIES... CTRL + B

Config Server

SPRING CLOUD CONFIG

Central management for configuration via Git, SVN, or HashiCorp Vault.

Spring Boot Actuator

OPS

Supports built in (or custom) endpoints that let you monitor and manage your application - such as application health, metrics, sessions, etc.

Consul Discovery

SPRING CLOUD DISCOVERY

Service discovery with Hashicorp Consul.

### 4- Gateway service

Project

☐ Gradle - Groovy
☐ Gradle - Kotlin
☒ Maven

Language

☒ Java
☐ Kotlin
☐ Groovy

Spring Boot

☐ 3.2.0 (SNAPSHOT)
☐ 3.2.0 (RC1)
☐ 3.1.6 (SNAPSHOT)
☒ 3.1.5
☐ 3.0.13 (SNAPSHOT)
☐ 3.0.12
☐ 2.7.18 (SNAPSHOT)
☐ 2.7.17

Project Metadata

Group

org.sid

Artifact

gateway-service

Name

gateway-service

Description

Demo project for Spring Boot

Package name

org.sid.gateway-service

Dependencies

ADD DEPENDENCIES... CTRL + B

Spring Boot Actuator

OPS

Supports built in (or custom) endpoints that let you monitor and manage your application - such as application health, metrics, sessions, etc.

Consul Discovery

SPRING CLOUD DISCOVERY

Service discovery with Hashicorp Consul.

Gateway

SPRING CLOUD ROUTING

Provides a simple, yet effective way to route to APIs and provide cross cutting concerns to them such as security, monitoring/metrics, and resiliency.

## Démarrer consul :

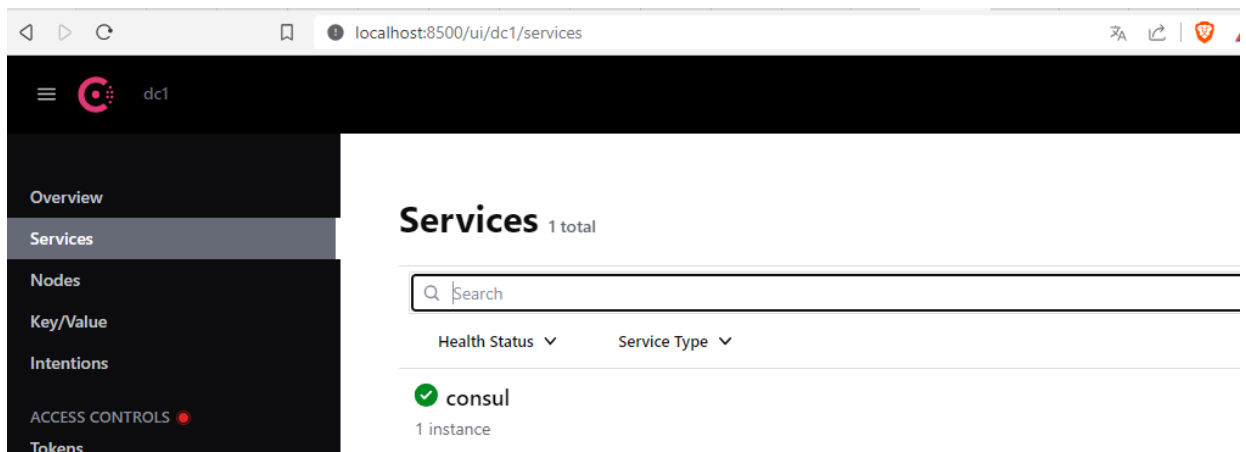
```
C:\Windows\System32\cmd.exe - consul agent -server -bootstrap-expect=1 -data-dir=consul-data -ui -bind=10.232.0.174

Passerelle par défaut. . . . . : 10.232.255.254

Carte Ethernet Connexion réseau Bluetooth :

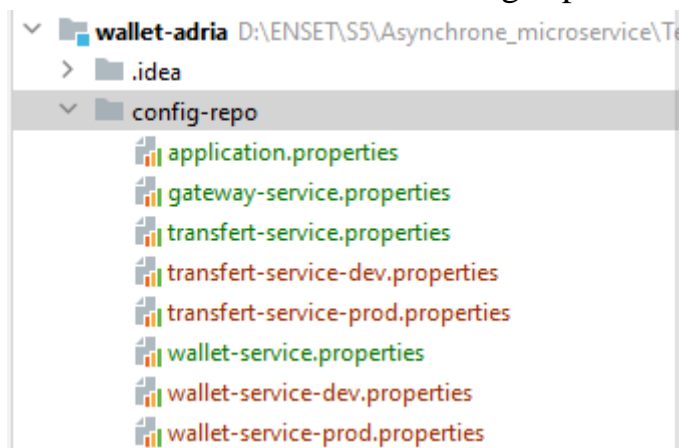
    Statut du média. . . . . : Média déconnecté
    Suffixe DNS propre à la connexion. . . :

D:\application\consul>consul agent -server -bootstrap-expect=1 -data-dir=consul-data -ui -bind=10.232.0.174
==> Starting Consul agent...
      Version: '1.16.2'
```



## Configuration :

Création de dossier externe config-repo et l'intégrer dans le service config



Le service config est bien demmarer dans le port 8888 :

```

{
  "name": "wallet-service",
  "profiles": [
    "default"
  ],
  "label": null,
  "version": "de0dd16dcedf0a0514cfba366a8848bc565d021f",
  "state": null,
  "propertySources": [
    {
      "name": "file:///D:/ENSET/S5/Asynchrone_microservice/Test_Adria_project/walle
      "source": {
        "wallet.params.x": "55",
        "wallet.params.y": "66",
        "management.endpoints.exposure.include": "*",
        "spring.datasource.url": "jdbc:h2:mem:wallet-db",
        "spring.h2.console.enabled": "true"
      }
    },
    {
      "name": "file:///D:/ENSET/S5/Asynchrone_microservice/Test_Adria_project/walle
      "source": {
        "global.params.p1": "434",
        "global.params.p2": "123"
      }
    }
  ]
}

```

## Wallet service :

localhost:8500/ui/dc1/services

dc1

- Overview
- Services**
- Nodes
- Key/Value
- Intentions
- ACCESS CONTROLS
- Tokens
- Policies
- Roles
- Auth Methods
- ORGANIZATION
- Peers

## Services 3 total

Health Status ▼
Service Type ▼

✓	consul	1 instance
✓	config-server	1 instance
✓	wallet-service	1 instance

```
localhost:8081/clients
{
  "_embedded": {
    "clients": [
      {
        "name": "Zakia",
        "email": "zakia@gmail.com",
        "_links": {
          "self": {
            "href": "http://localhost:8081/clients/1"
          },
          "client": {
            "href": "http://localhost:8081/clients/1{?projection}",
            "templated": true
          },
          "wallets": {
            "href": "http://localhost:8081/clients/1/wallets{?projection}",
            "templated": true
          }
        }
      },
      {
        "name": "Regoug",
        "email": "regoug@gmail.com",
        "_links": {
          "self": {
            "href": "http://localhost:8081/clients/2"
          },
          "client": {
            "href": "http://localhost:8081/clients/2{?projection}",
            "templated": true
          }
        }
      }
    ]
  }
}
```

localhost:8081/h2-console/login.do?jsessionid=9e2d2605421537c51

Auto commit Max rows: 1000 Auto complete Off Auto select

jdbc:h2:mem:wallet-db

- CLIENT
  - ID
  - EMAIL
  - NAME
  - Indexes
- WALLET
  - SOLDE
  - CLIENT\_ID
  - CREATED\_AT
  - DEVICE
  - ID
  - Indexes
- INFORMATION\_SCHEMA
- Users

H2 2.1.214 (2022-06-13)

Run Run Selected Auto complete Clear SQL statement:

SELECT \* FROM CLIENT

SELECT \* FROM CLIENT;

ID	EMAIL	NAME
1	zakia@gmail.com	Zakia
2	regoug@gmail.com	Regoug
3	hiba@gmail.com	hiba

(3 rows, 6 ms)

Edit

localhost:8081/h2-console/login.do?jsessionid=9e2d2605421537c5fd3ae3c6abdf8aae

Auto commit ☒ Max rows: 1000 Auto complete Off Auto select On

jdbc:h2:mem:wallet-db

- CLIENT
  - ID
  - EMAIL
  - NAME
  - Indexes
- WALLET
  - SOLDE
  - CLIENT\_ID
  - CREATED\_AT
  - DEVICE
  - ID
  - Indexes
- INFORMATION\_SCHEMA
- Users
- H2 2.1.214 (2022-06-13)

Run Run Selected Auto complete Clear SQL statement:

SELECT \* FROM WALLET;

SELECT \* FROM WALLET;

SOLDE	CLIENT_ID	CREATED_AT	DEVICE	ID
0.7711777210590173	1	2023-10-30 10:38:58.507	MAD	6a25ddaa-3a87-4ffa-8ef0-9d9cb1ce0854
0.8503080814745037	2	2023-10-30 10:38:58.545	MAD	6a058397-55e6-4fd7-a7b4-ceb496941b3a
0.31567862399967406	3	2023-10-30 10:38:58.549	MAD	dbf01f1b-872c-4560-9de8-81cff4604512

(3 rows, 2 ms)

Edit

## Gateway Service :

dc1 Help

Overview

Services

Nodes

Key/Value

Intentions

ACCESS CONTROLS

Tokens

Policies

Roles

Auth Methods

ORGANIZATION

Peers

### Services 4 total

Search

Health Status Service Type

Unhealthy to Health

- consul
  - 1 instance
- config-server
  - 1 instance
- gateway-service
  - 1 instance
- wallet-service
  - 1 instance

La gateway est bien demmarer dans le port 9999

```
{
  "_embedded": {
    "clients": [
      {
        "name": "Zakia",
        "email": "zakia@gmail.com",
        "_links": {
          "self": {
            "href": "http://DESKTOP-QDIRIP0:8081/clients/1"
          },
          "client": {
            "href": "http://DESKTOP-QDIRIP0:8081/clients/1{?projection}",
            "templated": true
          },
          "wallets": {
            "href": "http://DESKTOP-QDIRIP0:8081/clients/1/wallets{?projection}",
            "templated": true
          }
        }
      },
      {
        "name": "Regoug",
        "email": "regoug@gmail.com",
        "_links": {
          "self": {
            "href": "http://DESKTOP-QDIRIP0:8081/clients/2"
          },
          "client": {
            "href": "http://DESKTOP-QDIRIP0:8081/clients/2{?projection}",
            "templated": true
          }
        }
      }
    ]
  }
}
```