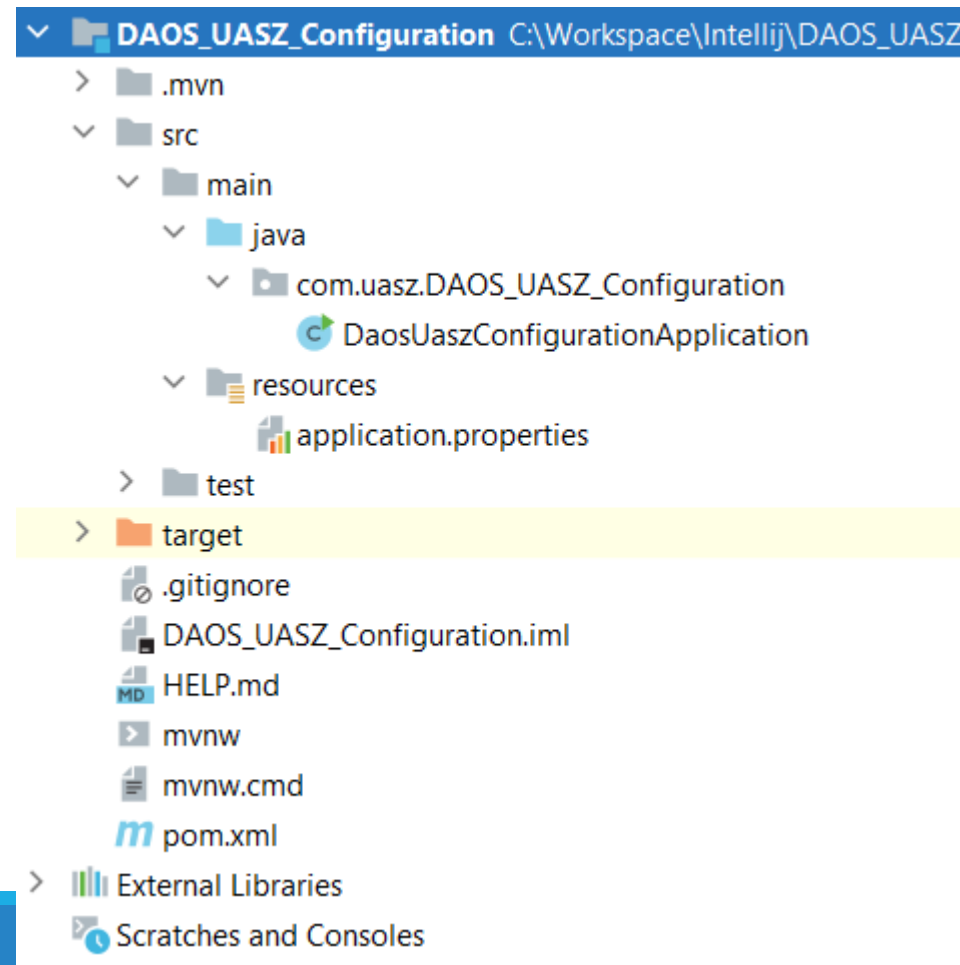


DAOS

Micro service Configuration



Pom.xml

```
<dependency>
  <groupId>org.springframework.cloud</groupId>
  <artifactId>spring-cloud-config-server</artifactId>
</dependency>

<dependencyManagement>
  <dependencies>
    <dependency>
      <groupId>org.springframework.cloud</groupId>
      <artifactId>spring-cloud-dependencies</artifactId>
      <version>${spring-cloud.version}</version>
      <type>pom</type>
      <scope>import</scope>
    </dependency>
  </dependencies>
</dependencyManagement>
```

Fichier Run

```
@EnableConfigServer
@SpringBootApplication
public class DaosUaszConfigurationApplication {

    public static void main(String[] args) {
        SpringApplication.run(DaosUaszConfigurationApplication.class, args);
    }
}
```

Application.properties

```
server.port=8888  
spring.cloud.config.server.git.uri=file://${user.home}/cloud-config  
spring.cloud.config.server.git.default-label=main
```

Creation du repository git : cloud-config

Lancer git bash :

- `cd c:`
- `cd utilisateurs/nom_user`
- `mkdir cloud-config`
- `cd cloud-config`
- `ls`
- `git init`

Creation des fichiers de configuration : cloud-config

Lancer git bash :

- cd c:
- cd utilisateurs/nom_user
- cd cloud-config
- code application.properties

Le fichier **application.properties** contient tout les parametres de configurations qui sont commun a tout les microservices

Creation des fichiers de configuration : cloud-config

Lancer git bash :

- cd c:
- cd utilisateurs/nom_user
- cd cloud-config
- code ms-service.properties

Le fichier **ms-service.properties** contient tout les parametres de configurations qui sont specifique a un microservice

maquette-service.properties

```
server.port=8081
```

```
spring.datasource.driverClassName=org.h2.Driver
```

```
spring.datasource.url=jdbc:h2:mem:maquette-db
```

```
spring.jpa.hibernate.ddl-auto=update
```

```
spring.jpa.show-sql=true
```

```
spring.h2.console.enabled=true
```

```
spring.h2.console.path=/maquette/h2
```

repartition-service.properties

```
server.port=8082
```

```
spring.datasource.driverClassName=org.h2.Driver  
spring.datasource.url=jdbc:h2:mem:repartition-db  
spring.jpa.hibernate.ddl-auto=update  
spring.jpa.show-sql=true  
spring.h2.console.enabled=true  
spring.h2.console.path=/repartition/h2
```

emploiDuTemps-service.properties

```
server.port=8083
```

```
spring.datasource.driverClassName=org.h2.Driver
```

```
spring.datasource.url=jdbc:h2:mem:emploiDuTemps-db
```

```
spring.jpa.hibernate.ddl-auto=update
```

```
spring.jpa.show-sql=true
```

```
spring.h2.console.enabled=true
```

```
spring.h2.console.path=/emploiDuTemps/h2
```

Creation des fichiers de configuration : cloud-config

Lancer git bash :

- `cd c:`
- `cd utilisateurs/nom_user`
- `cd cloud-config`
- `git status`
- `git add .`
- `git commit -m " message "`

Tester le MS Configuration

The screenshot shows a web browser interface displaying a JSON configuration file. The address bar indicates the URL is `localhost:8888/maquette-service/master`. The browser's developer tools are open, showing the JSON data in a collapsible tree view. The configuration is as follows:

```
{  "name": "maquette-service",  "profiles": {    "0": {      "label": null,      "version": "32c89280825a2fef42579f50468b5801e0da5553",      "state": null    }  },  "propertySources": {    "0": {      "name": "file://C:\\Users\\lno/cloud-config/maquette-service.properties",      "source": {        "server.port": "8081",        "spring.datasource.driverClassName": "org.h2.Driver",        "spring.datasource.url": "jdbc:h2:mem:maquette-db",        "spring.jpa.hibernate.ddl-auto": "update",        "spring.jpa.show-sql": "true",        "spring.h2.console.enabled": "true",        "spring.h2.console.path": "/maquette/h2"      }    }  } }
```

Micro Service Maquette

```
<dependency>  
  <groupId>org.springframework.cloud</groupId>  
  <artifactId>spring-cloud-starter-config</artifactId>  
  <version>4.1.0</version>  
</dependency>
```

```
spring.application.name=maquette-service  
spring.config.import=optional:configserver:http://localhost:8888/
```

Micro Service Repartition

```
<dependency>  
  <groupId>org.springframework.cloud</groupId>  
  <artifactId>spring-cloud-starter-config</artifactId>  
  <version>4.1.0</version>  
</dependency>
```

```
spring.application.name=repartition-service  
spring.config.import=optional:configserver:http://localhost:8888/
```

Micro Service EmploiDuTemps

```
<dependency>  
  <groupId>org.springframework.cloud</groupId>  
  <artifactId>spring-cloud-starter-config</artifactId>  
  <version>4.1.0</version>  
</dependency>
```

```
spring.application.name=emploiDuTemps-service  
spring.config.import=optional:configserver:http://localhost:8888/
```


Actuator

```
<dependency>  
  <groupId>org.springframework.boot</groupId>  
  <artifactId>spring-boot-starter-actuator</artifactId>  
</dependency>
```

```
management.endpoints.web.exposure.include=*
```