



SAÉ S4.A02.1 : Web Backend

vitemespizzas

par Randy DE WANCKER et Alexandre DEUDON

vitemespizzas



Table des matières

- [Description](#)
- [Structure du projet](#)
- [Prérequis](#)
- [Cloner le dépôt](#)
- [Installation et Compilation](#)
- [Les endpoint](#)
- [Diagramme de classes](#)
- [MCD](#)
- [License](#)

Description

Ce dépôt contient la production de [Randy DE WANCKER](#) et de [Alexandre DEUDON](#) pour la SAE4.A02.1 et conciste en un projet java visant a implémenter une API rest de menu et commandes fictives de pizzas

Structure du projet

```
. # racine du projet
├── assets # contenu additionel, images, pdf, ect...
├── META-INF # contient le context.xml decrivant le processus de déploiement
├── WEB-INF # contient les sources et classes des servlets
│   ├── classes ► # Contient la même structure que src/ mais avec des fichiers compilés
│   └── src # Contient les fichiers sources des servlets
│       ├── controleurs # Classes qui traite les actions de l'utilisateur
│       ├── dao # Data Access Object
│       ├── dataset # Les jeux de données
│       └── dto # Data Transfer Objet
```

Prérequis

Ce qu'il vous faut pour pouvoir utiliser ce projet

- Le kit de développement java n°11 qui peut être installé :
 - Sous Windows grâce à cet [installateur](#)
 - Sous Linux avec la commande `sudo apt install open-jdk-11`
- Le gestionnaire de projet maven : [Télécharger maven](#)
- tomcat9 : [Télécharger tomcat9](#)

⚠ ce projet ne prend pas en charge le packaging WAR et n'est donc pas compatible avec les versions tomcat 10 et +

Cloner le dépôt

À l'aide d'un terminal, naviguez jusqu'au répertoire `webapps` de votre installation de tomcat

```
git clone https://gitlab.univ-lille.fr/alexandre.deudon.etu/vitemespizzas.git
```

Installation et Compilation

Après avoir cloné le dépôt ouvrez/déplacez un terminal dans le répertoire `vitemespizzas` et exécutez les commandes adaptée(s) selon ce que vous souhaitez faire

Installer les dépendances

```
mvn install
```

Compiler le projet

```
mvn clean compile
```

Les endpoints

Ingrédients

```
curl -i -X GET http://localhost:8080/vitemespizzas/ingredients
```

retourne la liste de tout les ingredients

```
curl -i -X GET http://localhost:8080/vitemespizzas/ingredients/1
```

retourne les caractéristique de l'ingredient numéro 1

```
curl -i -X GET http://localhost:8080/vitemespizzas/ingredients/1/name
```

retourne uniquement le nom de l'ingredient numéro 1

```
curl -i -X POST http://localhost:8080/vitemespizzas/ingredients -H "Authorization: Bearer <token>" -d '{"id":1,"name":"tomate","price":0.2}'
```

rajoute l'ingredient numéro 20 avec un prix de 0.2€

```
curl -i -X DELETE http://localhost:8080/vitemespizzas/ingredients/20 -H "Authorization: Bearer <token>"
```

supprime l'ingredient numéro 20

Pizzas

```
curl -i -X GET http://localhost:8080/vitemespizzas/pizzas
```

retourne la liste de toute les pizzas

```
curl -i -X GET http://localhost:8080/vitemespizzas/pizzas/1
```

retourne la liste de toute les caracteristiques de la pizza numéro 1

```
curl -i -X POST http://localhost:8080/vitemespizzas/pizzas -H "Authorization: Bearer <token>" -d '{"id":1,"name":"Margherita","price":10,"ingredients":1,2,3}'
```

| |
|--|
| ajoute la pizza 4 fromages avec l'id 10 et le prix de base de 7€ |
| <pre>curl -i -X DELETE http://localhost:8080/vitemespizzas/pizzas/10</pre> |
| supprime (à mon grand regret) la pizza d'id 10 (la 4 fromages) |
| <pre>curl -i -X PATCH http://localhost:8080/vitemespizzas/pizzas/1 -H "Authorization: Bearer <token>" -d '{"ty</pre> |
| modifie le type de la pizza d'id 1 |
| <pre>curl -i -X POST http://localhost:8080/vitemespizzas/pizzas/1 -H "Authorization: Bearer <token>" -d '{"ing</pre> |
| ajoute l'ingrédient "pomme de terre" à la pizza d'id 1 |
| <pre>curl -i -X DELETE http://localhost:8080/vitemespizzas/pizzas/3/6</pre> |
| retire l'ingrédient d'id 6 (les champignons) de la pizza d'id 3 (reine) |
| <pre>curl -i -X GET http://localhost:8080/vitemespizzas/pizzas/2/prixfinal</pre> |
| fournit le prix final de la pizza d'id 2 |

Commandes

| |
|--|
| <pre>curl -i -X GET http://localhost:8080/vitemespizzas/commandes</pre> |
| fournit toutes les commandes et leurs détails |
| <pre>curl -i -X GET http://localhost:8080/vitemespizzas/commandes/1</pre> |
| fournit les détails de la commande d'id 1 |
| <pre>curl -i -X POST http://localhost:8080/vitemespizzas/commandes -H "Authorization: Bearer <token>" -d '{"ic</pre> |
| ajoute une commande passée par l'utilisateur d'id 2 le 8 novembre 2022 |
| <pre>curl -i -X GET http://localhost:8080/vitemespizzas/commandes/2/prixfinal</pre> |
| fournit le prix final de commande d'id 2 |

Utilisateurs

| |
|---|
| <pre>curl -i -X GET "http://localhost:8080/vitemespizzas/users/token?login=jean&pwd=jean"</pre> |
| fournit le jeton d'authentification de jean |

Modele conceptuel de donnees

qui peut être généré par le script [basePizzas.sql](#)

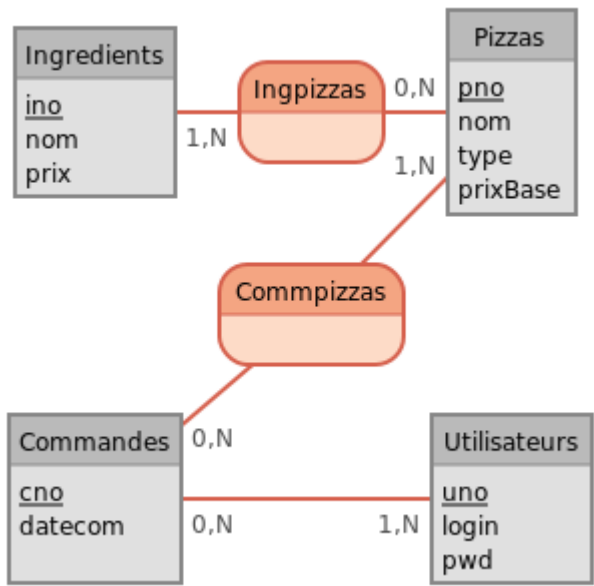
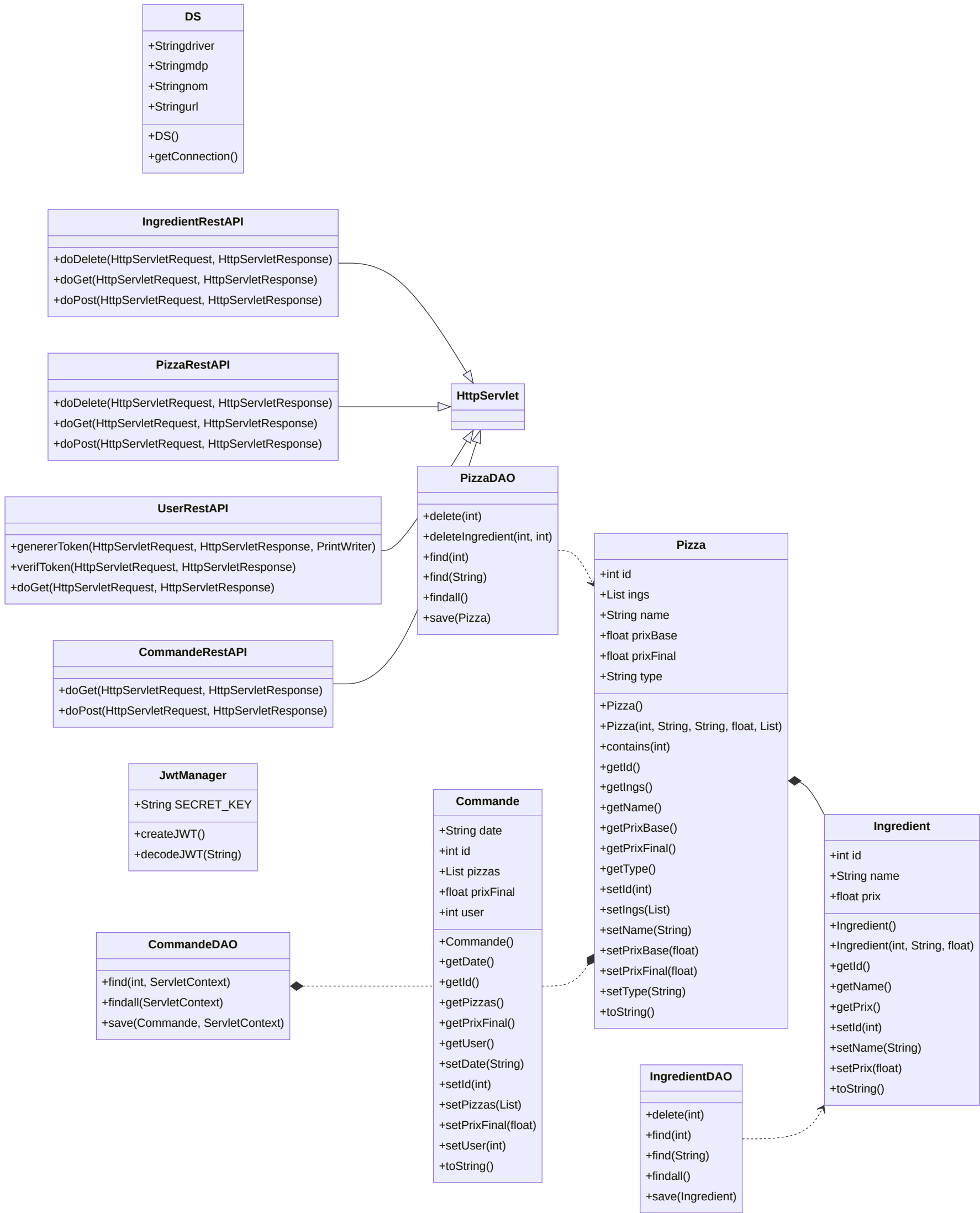


Diagramme de classes



Licence

L'ensemble des productions sur ce dépôt sont couvertes par la licence by-nc-sa 4.0.



```
1 drop table if exists users, commandes, compizzas, ingpizzas, pizzas, ingredients cascade;
2
3 create table ingredients (ino int primary key, name text, prix float);
4 insert into ingredients values (1, 'pomme de terre', 0.3),
5 (2, 'poivrons', 0.25),
6 (3, 'chorizo', 0.45),
7 (4, 'lardons', 0.4),
8 (5, 'aubergines', 0.2),
9 (6, 'champignons', 0.2),
10 (7, 'fromage', 0.5),
11 (8, 'ananas', 0.15),
12 (9, 'tomates', 0.35),
13 (10, 'olives', 0.1),
14 (11, 'oeufs', 0.2),
15 (12, 'jambon', 0.4);
16
17 create table pizzas (pno int primary key, name text, type text, prixBase float);
18 insert into pizzas values (1, 'margherita', 'tomate', 8),
19 (2, 'savoyarde', 'creme', 9),
20 (3, 'reine', 'tomate', 7);
21
22 create table ingpizzas (pno int, ino int, primary key (pno, ino));
23 insert into ingpizzas values (1, 7), (1, 6), (1, 9),
24 (2, 1), (2, 4), (2, 7),
25 (3, 5), (3, 7), (3, 12);
26
27 create table commandes (cno int primary key, uno int, datecomm date);
28 insert into commandes values (1, 1, '2022-10-12'),
29 (2, 1, '2022-10-19'),
30 (3, 2, '2022-11-03');
31
32 create table compizzas (cno int, pno int);
33 insert into compizzas values (1, 1), (1, 2),
34 (2, 2),
35 (3, 1), (3, 3), (3, 3);
36
37 create table users (uno int primary key, login text, pwd text);
38 insert into users values (1, 'jean', 'jean'), (2, 'paul', 'paul');
39
```