

# Serveur Web

DUREL Enzo, VILLEPREUX Thibault

May 17, 2022

## Sommaire

|          |                                    |          |
|----------|------------------------------------|----------|
| <b>1</b> | <b>Introduction</b>                | <b>2</b> |
| <b>2</b> | <b>Fonctionnalités importantes</b> | <b>2</b> |
| <b>3</b> | <b>Principales classes</b>         | <b>3</b> |
| <b>4</b> | <b>Bases de données</b>            | <b>3</b> |
| <b>5</b> | <b>Technologies utilisées</b>      | <b>5</b> |
| 5.1      | Entity . . . . .                   | 5        |
| 5.2      | DAO . . . . .                      | 5        |
| 5.3      | Core . . . . .                     | 6        |
| 5.4      | GUI . . . . .                      | 6        |
| 5.5      | Gestion états client . . . . .     | 6        |
| 5.6      | Gestion de statuts . . . . .       | 6        |
| <b>6</b> | <b>Routes</b>                      | <b>6</b> |
| 6.1      | login . . . . .                    | 6        |
| 6.2      | eleves . . . . .                   | 6        |
| 6.3      | professeurs . . . . .              | 7        |
| 6.4      | gommettes . . . . .                | 8        |

## **Introduction**

Nous avons créé un serveur.

## **Fonctionnalités importantes**

Quelles sont les fonctionnalités qui vous semblent importantes ?

L'objectif est de créer un dispositif où les élèves peuvent consulter leur carnet dématérialisé. Il faut également prévoir la possibilité à un professeur de mettre des gommettes à un élève.

Plus précisément, si le professeur est connecté il peut faire plusieurs actions :

- ajouter, supprimer ou modifier un élève (son nom et son prénom)
- ajouter, supprimer ou modifier une gommette (couleur et description)
- lister, ajouter ou supprimer une gommette attribuée à un élève. Une gommette attribuée possède une date, un motif et une gommette principale qui elle-même contient une couleur (rouge, vert ou blanc) et une description générale.
- créer ou supprimer un professeur
- se connecter/déconnecter de sa session

En revanche tout le monde a accès à la liste des gommettes configurées et la liste des gommettes de chaque élève, avec possibilité de voir le professeur ayant mis la gommette.

## Principales classes

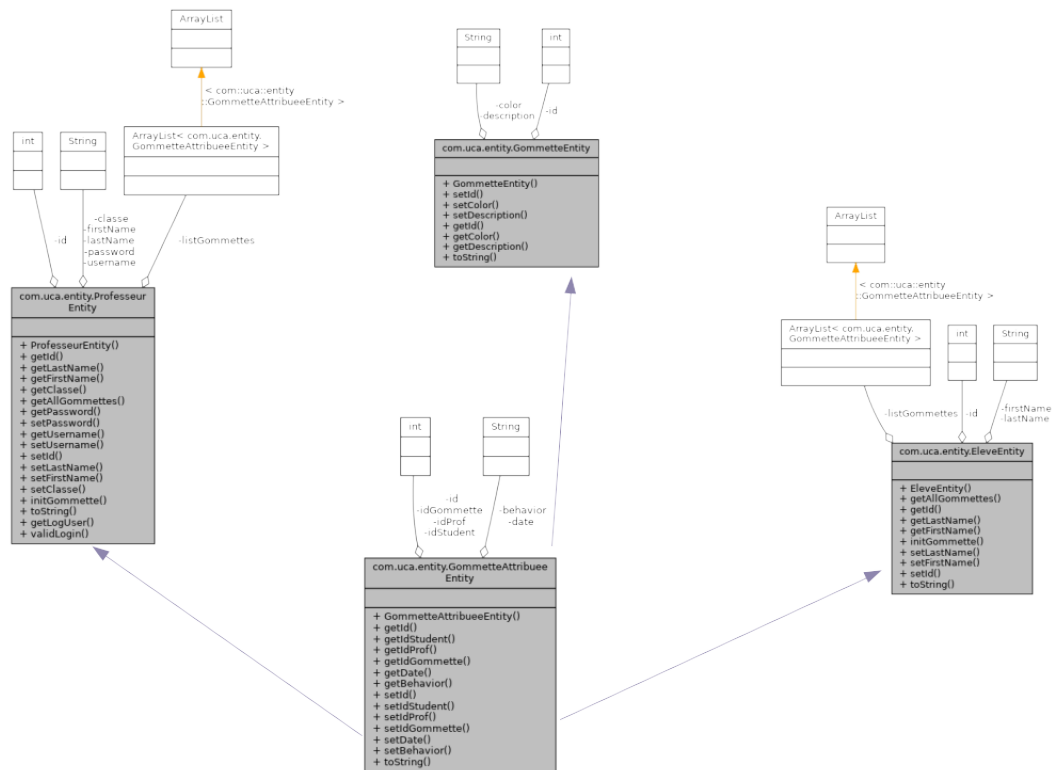


Fig. 1: UML des entités

Image disponible [ici](#) (./web-server/rapport/uml.png)

Quelles vont être les principales classes de votre applications ?

## Bases de données

Quelles vont être les tables de votre base de données ?

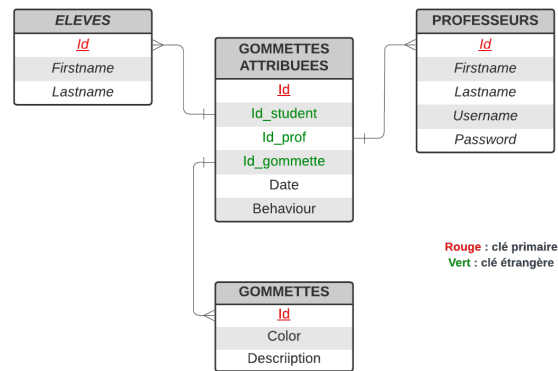


Fig. 2: UML de la Base De Données

Chaque classe principale possède une table dans la base de donnée relationnelle.

La table "PROFESSEURS" comme son nom l'indique représente la classe ProfesseurEntity, voici sa représentation avec quelques exemples :

Table 1: Table PROFESSEURS

| Id (INT, primary key)  | Firstname (VARCHAR 100) | Lastname (VARCHAR 100) |
|------------------------|-------------------------|------------------------|
| 1                      | Paul                    | Boulangier             |
| 2                      | Jules                   | Verne                  |
| 3                      | Georges                 | Lefou                  |
| Username (VARCHAR 100) | Password (VARCHAR 100)  |                        |
| paboulangier           | 1234                    |                        |
| juverne                | azerty1234              |                        |
| gelefou                | password                |                        |

La table "ELEVES" comme son nom l'indique représente la classe EleveEntity, voici sa représentation avec quelques exemples :

Table 2: Table ELEVES

| Id (INT, primary key) | Firstname ( VARCHAR 100) | Lastname (VARCHAR 100) |
|-----------------------|--------------------------|------------------------|
| 1                     | Jean                     | Valjean                |
| 2                     | Thomas                   | Ducarquoi              |
| 3                     | Timothée                 | Louis                  |

La table "GOMMETTES" comme son nom l'indique représente la classe GommetteEntity, voici sa représentation avec quelques exemples :

Table 3: Table GOMMETTES

| Id (INT, primary key) | Color (VARCHAR 50) | Description (VARCHAR 100) |
|-----------------------|--------------------|---------------------------|
| 1                     | Rouge              | Bavardage                 |
| 2                     | Vert               | Excellente note           |
| 3                     | Blanc              | Nettoie les toilettes     |

La table "GOMMETTESATTRIBUEES" comme son nom l'indique représente la classe GommetteAttribueeEntity, voici sa représentation avec quelques exemples:

Table 4: Table GOMMETTESATTRIBUEES

| Id (INT, primary key)        | Id <sub>student</sub> (INT) | Id <sub>prof</sub> (INT)               |
|------------------------------|-----------------------------|--|
| 1                            | 1                           | 1                                      |
| 2                            | 2                           | 1                                      |
| 3                            | 2                           | 1                                      |
| Id <sub>gommette</sub> (INT) | Date (VARCHAR 100)          | Behavior (VACHAR 10,000)               |
| 2                            | 2022-02-09                  | Jette de la nourriture                 |
| 1                            | 2022-06-07                  | A fait tout ces devoirs sans fautes    |
| 3                            | 2022-04-07                  | A aidé son camarade a faire ses lacets |

La date est en format VARCHAR. En effet, nous avons un formulaire HTML avec comme type une "date" afin de faciliter la saisie de date pour l'utilisateur grâce à l'apartirion d'un "calendrier dynamique". Cette date est en réalité une chaîne de caractère, ce qui nous arrange bien afin de la stockée dans la base de donnée.

## Technologies utilisées

Quelles solutions technologiques envisagez-vous ?

### 5.1 Entity

Les classes Entity correspondent aux représentation logiques de nos données. Elles possède presque uniquement des getters/setters pour faciliter le controle/emploi des instances.

### 5.2 DAO

Les classes DAO correspondent aux classes qui gère les ressources, en base de données, en fichiers texte, etc.

#### 5.2.1 SQL

Nous utilisons dans ce projet une base de données pour gérer les ressources. Elles nous semble beaucoup plus adaptés que des fichiers textes et plus durables dans le temps.

## 5.3 Core

La classe Core correspond aux classes de calcul. Elles sont appelées lorsqu'on doit utiliser un processus.

## 5.4 GUI

### 5.4.1 Freemarker

Nous utilisons un système de template pour générer la réponse graphique au client. Pour cela nous utilisons le Freemarker.

## 5.5 Gestion états client

### 5.5.1 Cookies

Nous avons décidé d'utiliser les cookies pour permettre l'authentification de l'utilisateur. Lorsque l'utilisateur se connecte correctement (mot de passe et nom d'utilisateur OK) on donne au client son id sous forme de cookies qui donnera donc à chaque requête.

### 5.5.2 Sessions

Côté serveur, nous ouvrons une session lors de la connexion du client. Ainsi nous savons si il est connecté ou non à chaque requête.

## 5.6 Gestion de statuts

## Routes

### 6.1 login

#### 6.1.1 login

GET "<http://localhost:8081/login>"

Page de connexion qui sert aussi de page principale au site.

POST "<http://localhost:8081/login>"

Méthode de vérification des identifiants du professeur, renvoie sur l'ent du professeur en cas de succes.

#### 6.1.2 logout

GET "<http://localhost:8081/logout>"

Méthode de déconnexion, redirige sur le login après la suppression de la session et du cookie de connexion.

### 6.2 eleves

GET "<http://localhost:8081/eleves>"

Affiche la liste des élèves avec leurs gommettes attribuées.

### 6.2.1 eleve

GET `"http://localhost:8081/eleves/:id"`

Affiche la page de profil de l'élève (nom, prénom, liste des gommettes attribuées).

### 6.2.2 create

GET `"http://localhost:8081/eleves/create"`

Affiche la page de création d'un élève (formulaire).

POST `"http://localhost:8081/eleves/create"`

Création de l'élève dans la bdd. Un élève est représenté par son nom et son prénom.

### 6.2.3 delete

GET `"http://localhost:8081/eleves/delete:id"`

Méthode de suppression d'un élève dont l'id dans l'uri correspond à l'id de l'élève dans la bdd. redirige vers la liste d'élèves.

### 6.2.4 modify

GET `"http://localhost:8081/eleves/modify/:id"`

Méthode de suppression d'un élève dont l'id dans l'uri correspond à l'id de l'élève dans la bdd.

POST `"http://localhost:8081/eleves/modify/:id"`

Modification de l'élève dans la bdd. L'id est spécifié dans l'uri.

## 6.3 professeurs

GET `"http://localhost:8081/professeurs"`

Affiche la liste des professeurs avec leurs gommettes attribuées.

### 6.3.1 professeur

GET `"http://localhost:8081/professeurs/:id"`

Affiche la page personnelle du professeur. Elle sert de page d'accueil et d'ent au professeur. Elle possède les interaction de base telle que création d'un élève, création d'une gommette, attribution d'une gommette à un élève.

### 6.3.2 create

GET "<http://localhost:8081/professeurs/create>"

Affiche la page de création d'un professeur. Un professeur est représenté par : nom, prénom, nom d'utilisateur, mot de passe.

POST "<http://localhost:8081/professeurs/create>"

Création du professeur dans la bdd.

### 6.3.3 delete

GET "<http://localhost:8081/professeurs/delete/:id>"

Création d'un professeur en bdd.

## 6.4 gommettes

GET "<http://localhost:8081/gommettes>"

Affiche la page de la liste des gommettes.

### 6.4.1 create

GET "<http://localhost:8081/gommettes/create>"

Affiche la page de création des gommettes.

POST "<http://localhost:8081/gommettes/create>"

Création de la gommette en bdd.

### 6.4.2 delete

GET "<http://localhost:8081/gommettes/delete:id>"

Méthode de suppression de la gommette, redirige vers la page de la liste des gommettes.

### 6.4.3 modify

GET "<http://localhost:8081/gommettes/modify/:id>"

Méthode de suppression d'une gommette dont l'id dans l'uri correspond à l'id de la gommette dans la bdd.

POST "<http://localhost:8081/eleves/modify/:id>"

Modification de la gommette dans la bdd. L'id est spécifié dans l'uri.



#### 6.4.4 attribuees

##### 1. create

GET `"http://localhost:8081/gommettes/attribuees/create/:id"`

Affiche la page d'attribution d'une gommette pour l'élève dont l'id est spécifiée dans l'uri.

POST `"http://localhost:8081/gommettes/attribuees/create/:id"`

Création de la gommette attribuee en bdd.

##### 2. delete

GET `"http://localhost:8081/gommettes/attribuees/delete/:id"`

Méthode de suppression de la gommette attribuee, l'id correspond à celui de l'élève à qui la gommette a été attribuée.

## Table des matières

|          |                                    |          |
|----------|------------------------------------|----------|
| <b>1</b> | <b>Introduction</b>                | <b>2</b> |
| <b>2</b> | <b>Fonctionnalités importantes</b> | <b>2</b> |
| <b>3</b> | <b>Principales classes</b>         | <b>3</b> |
| <b>4</b> | <b>Bases de données</b>            | <b>3</b> |
| <b>5</b> | <b>Technologies utilisées</b>      | <b>5</b> |
| 5.1      | Entity . . . . .                   | 5        |
| 5.2      | DAO . . . . .                      | 5        |
| 5.2.1    | SQL . . . . .                      | 5        |
| 5.3      | Core . . . . .                     | 6        |
| 5.4      | GUI . . . . .                      | 6        |
| 5.4.1    | Freemarker . . . . .               | 6        |
| 5.5      | Gestion états client . . . . .     | 6        |
| 5.5.1    | Cookies . . . . .                  | 6        |
| 5.5.2    | Sessions . . . . .                 | 6        |
| 5.6      | Gestion de statuts . . . . .       | 6        |
| <b>6</b> | <b>Routes</b>                      | <b>6</b> |
| 6.1      | login . . . . .                    | 6        |
| 6.1.1    | login . . . . .                    | 6        |
| 6.1.2    | logout . . . . .                   | 6        |
| 6.2      | elevés . . . . .                   | 6        |
| 6.2.1    | elevé . . . . .                    | 7        |
| 6.2.2    | create . . . . .                   | 7        |
| 6.2.3    | delete . . . . .                   | 7        |
| 6.2.4    | modify . . . . .                   | 7        |
| 6.3      | professeurs . . . . .              | 7        |
| 6.3.1    | professeur . . . . .               | 7        |
| 6.3.2    | create . . . . .                   | 8        |
| 6.3.3    | delete . . . . .                   | 8        |
| 6.4      | gommettes . . . . .                | 8        |
| 6.4.1    | create . . . . .                   | 8        |
| 6.4.2    | delete . . . . .                   | 8        |
| 6.4.3    | modify . . . . .                   | 8        |
| 6.4.4    | attribuees . . . . .               | 9        |

**List of Figures**

|   |                                     |   |
|---|-------------------------------------|---|
| 1 | UML des entités . . . . .           | 3 |
| 2 | UML de la Base De Données . . . . . | 4 |

**List of Tables**

|   |                                     |   |
|---|-------------------------------------|---|
| 1 | Table PROFESSEURS . . . . .         | 4 |
| 2 | Table ELEVES . . . . .              | 4 |
| 3 | Table GOMMETTES . . . . .           | 5 |
| 4 | Table GOMMETTESATTRIBUEES . . . . . | 5 |