

SMOG 20

Introduction	3
Rappel du cahier des charges	3
Objectifs	3
Spécifications	3
Environnement	3
Organisation	3
Livrable	4
Étude d'opportunité	5
Analyse fonctionnelle	5
Fonctionnalités	5
Visualisation des bâtiment du CFPT	5
Visualisation des étages d'un bâtiment du CFPT	5
Visualisation d'une salle	5
Consultation de l'horaire annuel d'une salle	5
Analyse organique	5
Architecture	5
Base de données	7
Frontend	8
Backend	8
Routes	8
Middlewares	8
CorsMiddleware	8
Controllers	8
ClassroomController	8
all	8
FloorController	8
all	8
numberfloor	9
BuildingController	9
all	9
LessonController	9
all	9
classSchedule	9
Frontend	9
Babylon.js	9
Tests	9
Apport personnel	9

Introduction

Cette documentation a pour but de détailler les étapes nécessaires à la réalisation de l'application web SMOG 20, réalisée dans le cadre du TSEM (Travaux de Semestre).

SMOG 20 est une application WEB qui permet à quiconque de visualiser les principaux bâtiments du CFPT ainsi que leurs étages et leurs salles. En effet, l'application propose une modélisation 3D des bâtiments et des étages. Pour ce qui est des salles, une photo en 360° de son intérieur est disponible. De plus, l'application permet de visualiser l'horaire annuel de chaque salle.

Rappel du cahier des charges

Objectifs

Durant environ 14 semaines, le but est de concevoir une application web qui permettra à un utilisateur quelconque de visualiser les bâtiments, étages et salles du CFPT. L'utilisateur pourra également consulter les horaires annuels de chaque classe. L'objectif principal de cette application est de faciliter la reconnaissance d'une salle parmi les différents bâtiments du CFPT.

Spécifications

L'application permettra de :

- Visualiser une modélisation 3D des différents bâtiments du CFPT
- Visualiser une modélisation 3D des différents étages d'un des bâtiments du CFPT
- Aperçu de la classe grâce à une photo en 360° degrés de celle-ci
- Consultation des horaires annuels de chaque classe.

Environnement

- Serveur Web et SGBD à choix (Laragon)
- IDE (Visual Studio Code, PHPStorm)
- Logiciel de modélisation (SketchUp, Blender)
- Outil de versionning de code (Git, avec dépôt distant sur Github)
- Logiciel de création de maquettes d'interfaces utilisateur (Sketch, Adobe XD, Proto.io, autre)
- Outil bureautique à choix pour les documents (Google Docs, MSOffice, OpenOffice)

Organisation

Élève :

- Jonathan Borel-Jaquet, jonathan.brljq@eduge.ch
- Lorenzo Bauduccio, lorenzo.bdcc@eduge.ch
- Jules Stahli, jules.sthl@eduge.ch

Enseignants :

- Francisco Garcia
- Stéphane Garchery

Livrable

Pour la fin des travaux de semestre, le 22 novembre 2020 :

- MindMap
- Cahier des Charges
- Poster
- Planning
- Rapport
- Journal de bord
- Présentation

Étude d'opportunité

Nous avons choisi d'effectuer ce projet en réponse à un des besoins que nous avons besoin lors de notre entrée au CFPT, et même encore à l'heure actuelle. Ce besoin étant la localisation exacte des différentes salles où nous avons cours. Le CFPT ayant à sa disposition différents bâtiments, il était des fois pour nous compliquer de nous repérer ainsi que de se rendre aux différentes salles. Nous avons donc choisi de créer une application permettant de régler ce problème.

Analyse de l'existant

Notre projet SMOG20 se base sur le site qui permet de consulter nos horaires SMOG15. SMOG15 étant vieillissant, une nouvelle version s'impose.

Analyse fonctionnelle

Fonctionnalités

Visualisation des bâtiments du CFPT

Cette fonctionnalité permet à un utilisateur de visualiser les modélisations 3D des différents bâtiments du CFPT, en cliquant sur l'un d'eux, l'utilisateur sera redirigé sur la modélisation 3D de son rez-de-chaussée.

Visualisation des étages d'un bâtiment du CFPT

Cette fonctionnalité permet à un utilisateur, après avoir cliqué sur un des bâtiments, de se déplacer dans les différents étages de celui-ci. L'utilisateur pourra alors cliquer sur une des salles de l'étage pour y accéder.

Visualisation d'une salle

Cette fonctionnalité permet à un utilisateur, après avoir cliqué sur une des salles d'un étage, de visualiser une photo en 360° de celle-ci.

Consultation de l'horaire annuel d'une salle

Cette fonctionnalité permet à un utilisateur, après avoir cliqué sur une salle, de consulter l'horaire annuel de celle-ci.

Interfaces

Barre de navigation

Standard



Fig. 1 - Barre de navigation standard

Smog 20 : Redirige l'utilisateur vers la page d'accueil de l'application (voir *Fig. 3 - Scène des bâtiments du CFPT*).

Bâtiments : Ouvre un menu déroulant avec tous les bâtiments du CFPT afin de s'y diriger.

Salle de classe

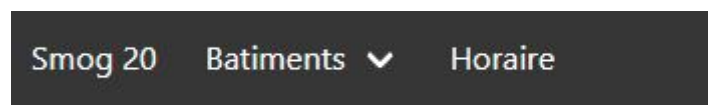


Fig. 2 - Barre de navigation dans une salle de classe

Smog 20 : Redirige l'utilisateur vers la page d'accueil de l'application (voir *Fig. 3 - Scène des bâtiments du CFPT*).

Bâtiments : Ouvre un menu déroulant avec tous les bâtiments du CFPT afin de s'y diriger.

Horaire : Affiche l'horaire de la salle de classe courante (voir *Fig. 6 - Affichage des horaires d'une salle de classe*).

Scènes

Bâtiments

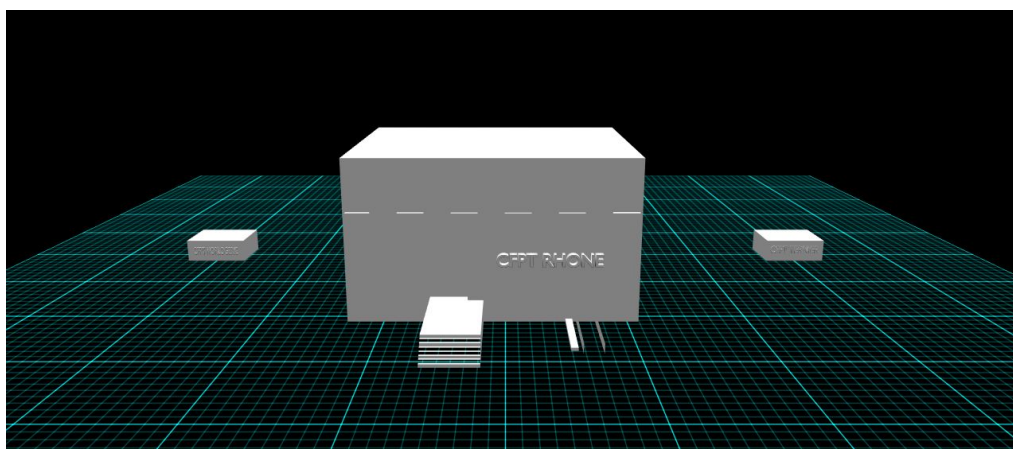


Fig. 3 - Scène des bâtiments du CFPT

Scène d'accueil de l'application, elle permet de cliquer sur un des bâtiments du CFPT afin de pouvoir y visualiser ces différents étages (voir *Fig. 4 - Scène des étages d'un bâtiment du CFPT*).

Étages

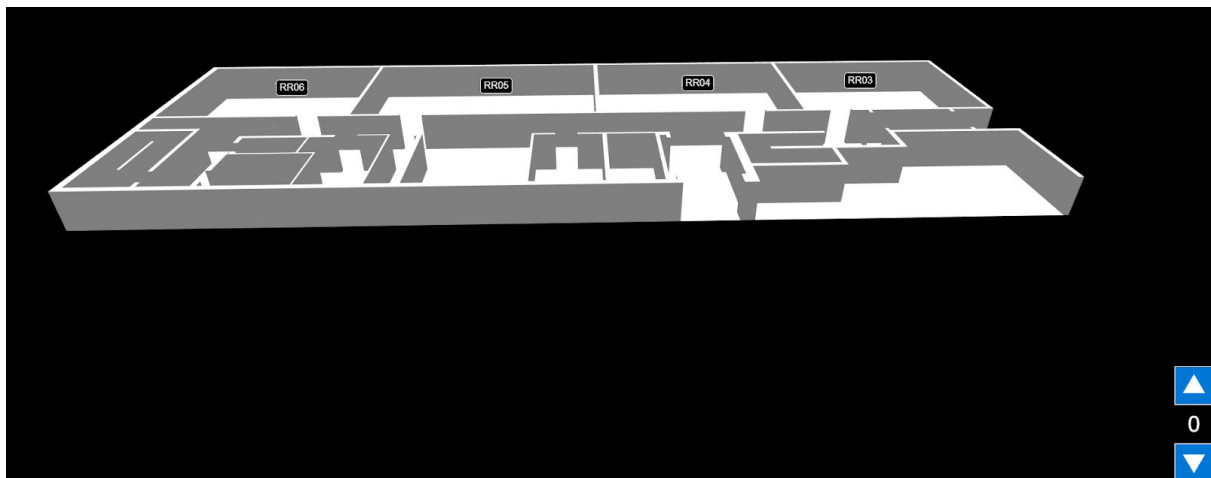


Fig. 4 - Scène des étages d'un bâtiment du CFPT

Scène des étages d'un bâtiment du CFPT, elle permet de visualiser les différents étages du bâtiment en cliquant sur les flèches directionnelles et d'accéder aux salles de classe en cliquant sur leurs noms (voir Fig. 5 - Scène de salle de classe).

Salle de classe

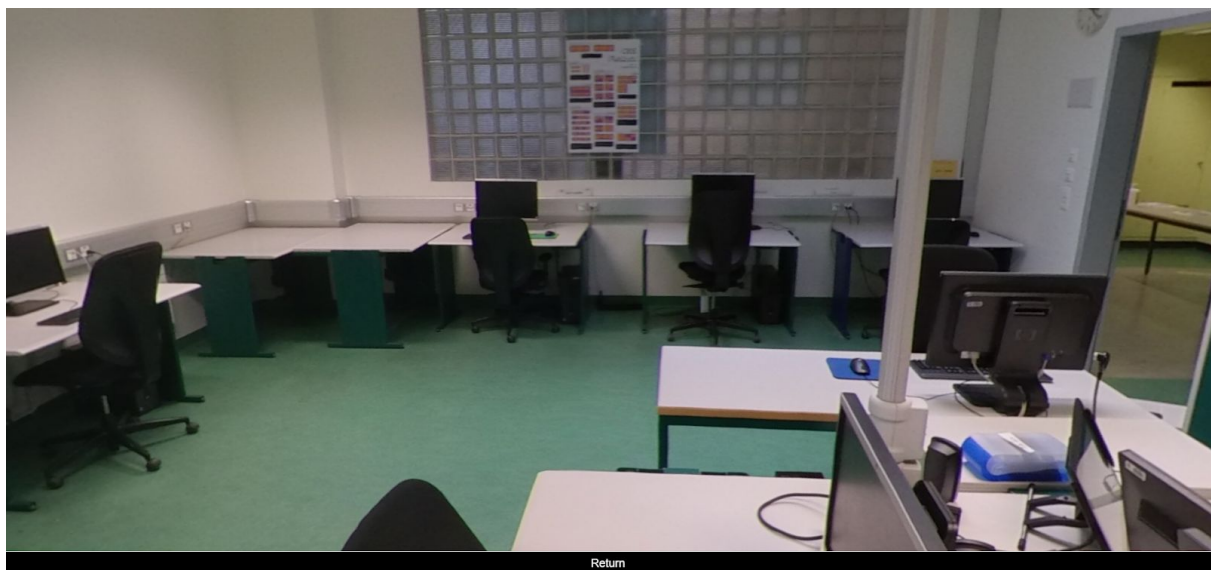


Fig. 5 - Scène de salle de classe

Scène de salle de classe, elle permet de visualiser la salle de classe en 360°.

Return : Redirige l'utilisateur vers la scène d'étage (voir Fig. 4 - Scène des étages d'un bâtiment du CFPT)

Modal

Horaire d'une salle de classe

Lundi	Mardi	Mercredi	Jeudi	Vendredi
		Laboratoire Applications		Atelier Bureautique
Module 114	Autre Appuis	Laboratoire Applications	Module (cie) 101	Atelier Bureautique
Module 114	Autre Appuis	Atelier Applications	Module (cie) 101	Atelier Bureautique
Module 114	Laboratoire Entreprise	Atelier Applications	Module (cie) 101	Atelier Bureautique
Module 114	Laboratoire Entreprise	Atelier Applications	Module (cie) 101	Atelier Bureautique
	Module 404	Laboratoire Applications	Laboratoire Applications	Laboratoire Applications
	Module 404	Laboratoire Applications	Laboratoire Applications	Laboratoire Applications
	Module 404	Atelier Applications	Atelier Applications	Atelier Applications
	Module 404	Atelier Applications	Atelier Applications	Atelier Applications
		Atelier Applications	Atelier Applications	Atelier Applications

Fig. 6 - Affichage des horaires d'une salle de classe

Cas d'utilisation

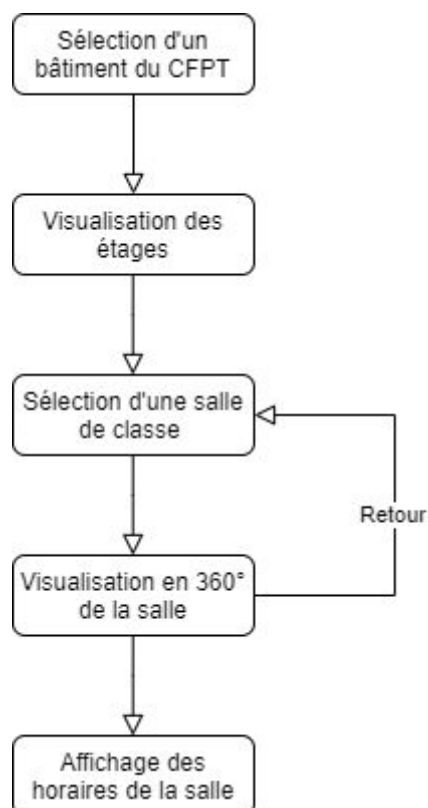


Fig. 7 - Flowchart de l'application

Analyse organique

Architecture

app

app/Console

app/Console/Console

app/Console/Console/Commands

app/Events

app/Exceptions

app/Http

app/Http/Controllers

app/Http/Middleware

app/Jobs

app/Listeners

app/Providers

bootstrap

database

database/factories

database/migrations

database/seeds

documentation

documentation/mindmap

documentation/poster

documentation/sketchup

node_modules

public

public/images

public/mesh

ressources

resources/views

routes

src

src/scss

src/ts

src/ts/librairies

src/ts/scenes

storage

storage/app

storage/framework

storage/framework/cache/data

storage/framework/views

storage/logs

tests

vendor

Base de données

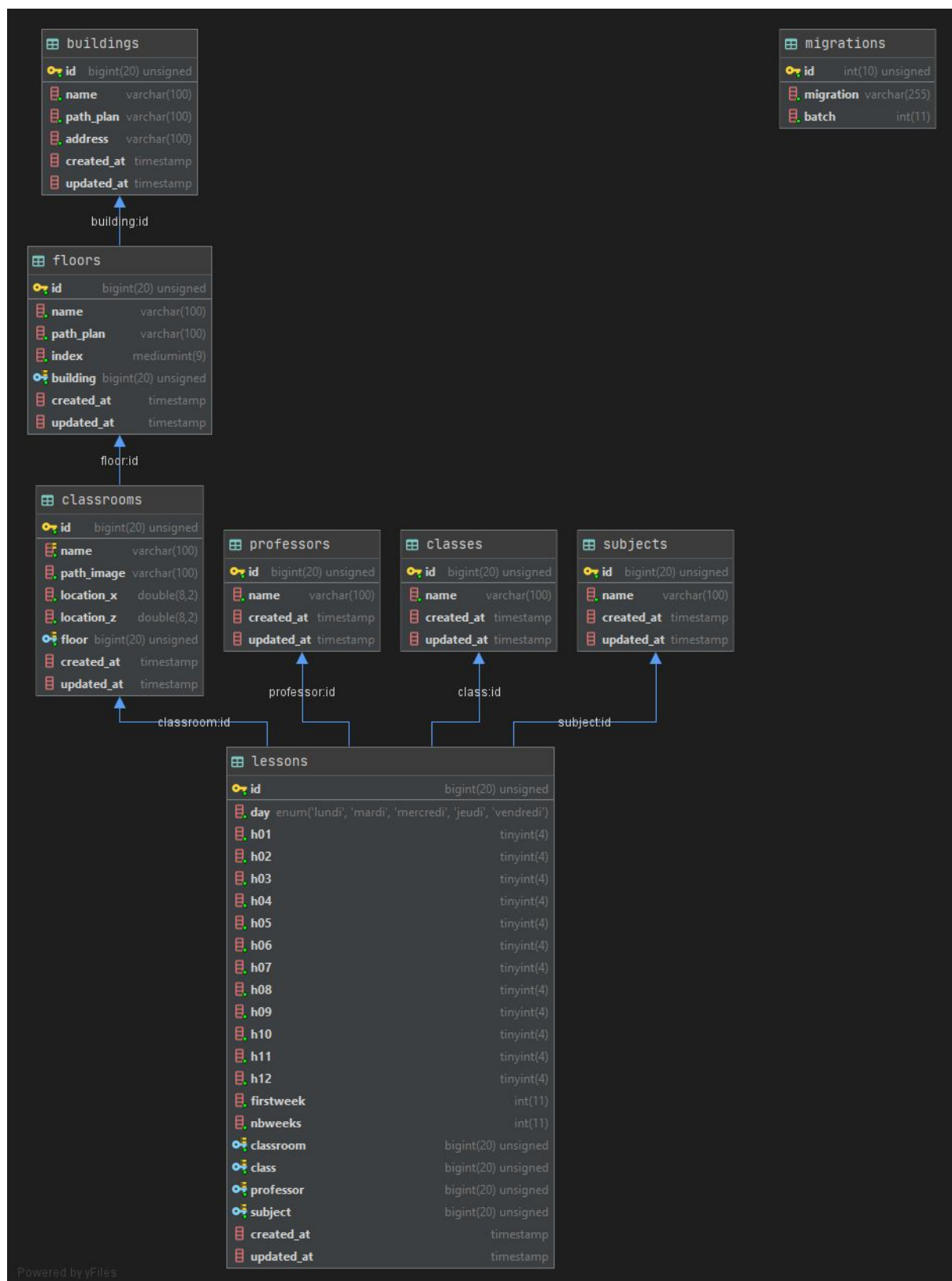


Fig. 8 - Base de données

Frontend

Librairies

Fontawesome

Bibliothèque d'icônes

<https://github.com/FortAwesome/Font-Awesome>

Bulma

Bulma est un framework css open source. Nous l'avons utilisé, car Bootstrap créait des conflits avec BabylonJs.

L'une des spécificité de Bulma est que les fichier javascript ne sont pas fournis cela nous a permis d'éviter les conflits mais nous a fait perdre un peu de temps car nous avons dû créer certaines fonctions nous-même.

<https://github.com/jgthms/bulma>

Moment.js

Momentjs est une librairie qui permet de simplifier l'utilisation des dates en javascript. Elle nous a été utile dans le calcul de la semaine courante pour afficher les horaires de la semaine.

<https://github.com/moment/moment>

WebpackJS

Webpack est un grouper de modules (module bundler) JavaScript open source . Il est principalement conçu pour JavaScript, mais il peut transformer les assets frontend tels que HTML, CSS et images si les loaders correspondants sont inclus. Webpack prend des modules avec des dépendances et génère des assets statiques représentant ces modules.

Source: Wikipedia ⇒ <https://fr.wikipedia.org/wiki/Webpack>

<https://github.com/webpack/webpack>

Axios

Axios est une librairie permettant de faire des appels ajax de façon simple et intuitive.

<https://github.com/axios/axios>

Lodash

Lodash est une bibliothèque JavaScript qui fournit des fonctions utilitaires pour les tâches de programmation courantes en utilisant le paradigme de programmation fonctionnelle .

Source: Wikipedia ⇒ <https://en.wikipedia.org/wiki/Lodash>

<https://github.com/lodash/lodash>

Ts-events

Librairie permettant une meilleure gestion des événements en Typescript.

<https://github.com/rogierschouten/ts-events>

BabylonJS

BabylonJs est une librairie javascript qui a pour but de créer un moteur graphique simple mais puissant.

Nous avons utilisé BabylonJs pour toute la partie affichage des modèles et des photos en 360.

<https://github.com/BabylonJS/Babylon.js>

Scènes

Buildings

Affiche les bâtiments du projet.

Classroom

Affiche une salle de classe en 360° ainsi que les horaires de celle-ci.

Floors

Affiche les étages d'un bâtiment.

Backend

Librairies

Lumen

Lumen est un dérivé de Laravel. Il a été conçu dans le but d'être plus léger que Laravel et de permettre de créer une API. Nous avons utilisé lumen pour gérer notre base de données grâce au système de migration. Tous les appels de notre application passent par Axios qui accède à notre API grâce au système de route de Lumen.

<https://github.com/laravel/lumen>

Php-parsecsv

Php-parsecsv est une librairie php permettant de parser des fichiers csv.

<https://github.com/parsecsv/parsecsv-for-php>

Routes

	Route	Controller	Méthode	Middlewares
GET	/	∅	∅	CorsMiddleware
POST	/api/classrooms	ClassroomController	all	CorsMiddleware
POST	/api/floors	FloorController	all	CorsMiddleware
POST	/api/buildings	BuildingController	all	CorsMiddleware
POST	/api/floorsNumber	FloorController	numberFloor	CorsMiddleware
POST	/api/schedule	LessonController	classSchedule	CorsMiddleware

Middlewares

CorsMiddleware

Autorise les requêtes provenant de machines inconnues.

Il s'applique à toutes les routes.

Controllers

ClassroomController

all

Paramètres:

floor_id	required numeric
----------	------------------

Retourne la liste des salles d'un étage.

FloorController

all

Paramètres:

building_id	required numeric
-------------	------------------

Retourne la liste des étages d'un bâtiment.

numberfloor

Paramètres:

building_id	required numeric
-------------	------------------

Retourne le nombre d'étages d'un bâtiment.

BuildingController

all

Paramètres: ∅

Retourne la liste des bâtiments.

LessonController

all

Paramètres: ∅

Retourne la liste des leçons.

classSchedule

Paramètres:

classroom_id	required numeric
nbr_week	required numeric

Retourne les horaires d'une salle.

Commandes

InsertData ⇒ php artisan insert_data

Paramètres: ∅

Récupère les données du projet contenu dans les fichiers csv et les insère dans la base de données.

Fichiers sources:

- database/buildings.csv (sans en-têtes)
- database/classes.csv (avec en-têtes)
- database/classrooms.csv (sans en-tête)
- database/floors.csv (sans en-tête)
- database/lessons.csv (avec en-tête)
- database/subjects.csv (avec en-tête)
- database/teachers.csv (avec en-tête)

Tests

Vu la nature du projet, des tests automatiques sont difficilement applicables. Nous avons donc testé notre projet manuellement.

Apport personnel

Jonathan Borel-Jaquet

Ce projet m'a permis de découvrir le moteur 3D sous forme de bibliothèque JavaScript s'appelant Babylon.js. Après approfondissement de l'outil, j'ai directement été emballé par celui-ci. Babylon.js étant lui-même codée en TypeScript, j'ai découvert une nouvelle façon d'aborder le JavaScript. Pour ce qui est du BackEnd, j'ai pu approfondir mes connaissances avec le Framework Laravel en utilisant le Micro-Framework Lumen. Je suis dans l'ensemble satisfait de la version 1.0 que nous avons produite.

Lorenzo Bauduccio

Ce projet m'a permis de retravailler en groupe, et cela, avec des technologies que je ne maîtrise pas forcément. J'ai durant ce projet eu la charge de toute la partie backend PHP. Pour le php j'ai utilisé lumen qui est un dérivé de Laravel. L'an dernier, nous avons vu en cours Laravel ce qui m'a permis de vite trouver mes marques. Travailler avec Mr. Stahli et Mr. Borel a été des fois compliqué pour ce qui est des dépendances du projet. Par exemple, la base de données nous a été fournie, mais les données ont dû être adaptées à notre base de données ce qui nous a bloqué pendant un moment. Mais travailler avec des technologies comme Babylon.js et Lumen a vraiment été intéressant pour moi et je pense que le travail accompli par notre équipe est vraiment bon.

Jules Stähli

Ce projet m'a permis de retravailler en groupe, d'utiliser certaines technologies découvertes ces dernières années et de tester différentes approches pour structurer un projet web single page sans avoir recours à un gros framework comme Angular ou Vue.

J'ai également profité de ce projet pour approfondir mes connaissances dans l'utilisation de la librairie BabylonJS particulièrement utile pour faire des rendus 3D.