

CYCLE DE FORMATION

Analyste en Génie Logiciel et Réseaux

Titre RNCP- Niveau 6

PROJET DE FIN D'ETUDES

Sujet :

Amélioration de l'application GPCI du centre IFIDE SUP'FORMATION

SOUTENU par : ROBLE GUIRE Mane

ENTREPRISE : IFIDE SUP'FORMATION

Sous le tutorat de : M. DIEMER Michel

Période du 09/05/2022 au 13/07/2022



REMERCIEMENTS

Je tiens à adresser mes remerciements les plus distingués à Monsieur MOSIN sans qui je n'aurais pas eu cette occasion de faire un stage.

Ensuite, j'aimerais adresser un remerciement particulier à mon tuteur de stage Monsieur DIEMER pour sa présence et son aide précieuse.

A Madame KUGEL Carole, pour ses efforts et son aide précieuse dans la recherche de stage.

A tous mes professeurs qui ont enrichi à la fois mes savoir-faire et -être et qui ont pris part à ma formation tant dans le milieu académique que dans le milieu professionnel.

Et enfin à l'administration du Centre IFIDE SUP'FORMATION pour leur aide au quotidien ; je voudrais que ce rapport de stage soit un témoignage de reconnaissance.

RESUME

Ce rapport présente le projet de fin d'études, un projet réalisé lors d'un stage d'une période de deux mois effectués au sein du centre IFIDE SUP'FORMATION. Durant cette période de stage, j'ai travaillé sur un projet qui consiste à étudier, enlever les bugs et ajouter de nouvelles fonctionnalités sur l'application interne GPCI.

Cette application permet à l'administration du centre de gérer le planning des cours et aux formateurs de renseigner leurs indisponibilités afin de pouvoir mettre en place le planning pour les cours, de semaine en semaine.

Les formateurs sont des professionnels en exercice et sont libres d'accepter des missions même en cours d'année scolaire. Ainsi, entre un et deux mois en avance, ils sont invités à renseigner leurs indisponibilités, et sur cette base le planificateur peut s'assurer de la disponibilité des formateurs pour les cours.

L'application, réalisé par d'anciens étudiants du centre, est déjà fonctionnelle mais a besoin de certaines améliorations.

Ce rapport comporte une documentation de l'application, ses interfaces et une analyse du code de l'application. Et les améliorations qu'on a pu apporter à l'application.

ABSTRACT

This report presents the final project, a two-month internship at the IFIDE SUP'FORMATION center. During this internship period, I worked on a project that consists of studying, removing bugs and adding new features on the internal GPCI app.

This app allows the administration of IFIDE SUP'FORMATION to manage the schedule of courses of the center and the trainers (teachers) to inform their unavailability in order to be able to set up the schedule for the courses of the week.

The application is already functional but, needs some improvements.

This report includes a documentation of the app, interfaces and analysis of the app's software code. And the improvements we were able to make to the app.

Table des matières

REMERCIEMENTS	2
RESUME.....	3
ABSTRACT	3
INTRODUCTION.....	5
1 Présentation de l'entreprise	6
1.1 Organigramme de l'IFIDE SUP'FORMATION Eckbolsheim	7
1.2 Chiffre d'affaires	8
1.3 Clients et fournisseurs	8
1.4 Concurrence.....	9
1.5 Management	10
2 Etude de l'existant	11
2.1.1 Installation et configuration de la machine de développement	11
2.1.2 Serveur Samba.....	15
2.1.3 PuTTY	18
2.1.4 Github.....	19
2.1.5 Visual Studio Code.....	20
3 GPCI.....	21
2. Backend (base de données)	22
3.1.1 Créer un user	22
3.1.2 Les routes	22
3.1.3 Les requêtes	23
3.1.4 Services et controllers.....	23
3.1.1 Accueil.....	24
3.1.2 Administration.....	25
3.1.3 Planification.....	26
3.1.4 L'enseignement	28
4 Améliorations réalisées	30
4.1.1 Gestion des salles	30
4.1.2 Correction sur menu Administration	33
CONCLUSION (Version française).....	34
CONCLUSION (English version).....	35
REFERENCES.....	36
LISTE DES FIGURES	37
ANNEXES	38

INTRODUCTION

Actuellement en stage de développement web au centre IFIDE.

Mon poste consiste à améliorer et apporter des nouvelles fonctionnalités à l'application GPCI.

L'application GPCI (Gestion Planning Centre de formation IFIDE), est une application de gestion de planning du centre IFIDE. Elle permet aux formateurs du centre IFIDE SUP'FORMATION de renseigner leurs indisponibilités.

Les cours et la réalisation de petits projets au cours de l'année m'ont permis quant à eux de me familiariser avec certains outils que j'ai utilisés pour la réalisation du projet. L'assistance au quotidien de mon tuteur m'a grandement aidée à développer un vocabulaire technique professionnel ainsi il m'a fait découvrir des outils très intéressants.

Les objectifs du projet sont :

- Créer un environnement de développement, donc avec les bonnes versions de PHP, d'Angular JS, créer la base de données...
- Analyse de la structure existante du projet, côté backend (base de données) et frontend (interface utilisateur). Être capable de créer une nouvelle interface en respectant la structure du code.

Comment va-t-on réaliser ce projet en respectant ses objectifs et répondre aux attentes du client ?

Dans un premier temps, on va créer un environnement de développement adapté. Ensuite, on va étudier et bien cerner l'application en analysant son interface utilisateur (frontend) et la partie base de données (backend). Et enfin, on va apporter les améliorations possibles qu'on puisse faire.

1 Présentation de l'entreprise

Créé à Metz en 1990 par Monsieur Norbert GUIRKINGER, le centre IFIDE SUP'FORMATION était essentiellement basé sur la formation en informatique et bureautique de salariés.

L'ouverture d'agences à Eckbolsheim et Nancy a permis à l'IFIDE SUP'FORMATION de se positionner comme un acteur majeur de la formation dans le Grand-Est avec un panel de clients allant des entreprises régionales aux petites entreprises locales et associatives.

Le centre IFIDE SUP'FORMATION d'Eckbolsheim, ouvert en tant que SARL avec un capital social de 7 622,45 € le 1er mars 1998 est actuellement dirigé par Monsieur Didier MOSIN. Il est le dernier à proposer encore des formations en informatique et bureautique, les deux autres centres ne proposant plus que des formations diplômantes et ayant abandonné le nom « IFIDE » n'utilisant plus que « SUP'FORMATION ».

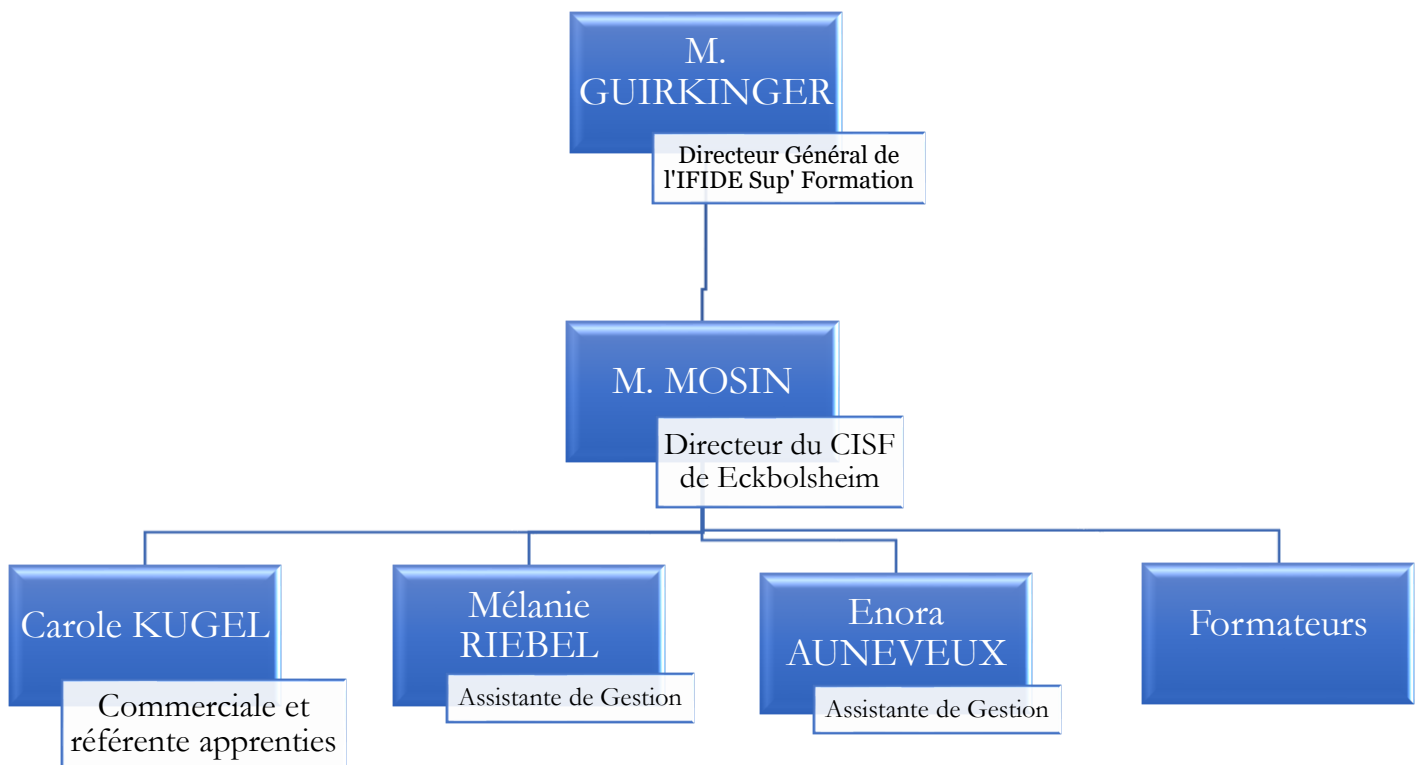
Aujourd'hui le centre IFIDE SUP'FORMATION d'Eckbolsheim est à la fois un centre de formation en informatique et bureautique, un Centre de Formation en Alternance (CFA) et un établissement d'enseignement privé de cursus scolaire postbac.

L'IFIDE propose des formations postbac du Bac +2 (Brevet Technicien Supérieur) au Bac +3 (Bachelor) dans les domaines de l'informatique, du commercial et de l'administratif.

Les formations postbac sont depuis 2020 certifiées QUALIOPi en formation en initial ou alternance, à différents BTS avec **des résultats qui nous placent régulièrement en tête au niveau académique.**

1.1 Organigramme de l'IFIDE SUP'FORMATION Eckbolsheim

L'équipe de l'IFIDE SUP'FORMATION d'Eckbolsheim est composée de Monsieur Didier MOSIN, le directeur du Centre, Madame Carole KUGEL qui est commerciale et référente des



élèves, Madame Mélanie RIEBEL Assistante Ressources Humaines, des formateurs et Enora AUNEVEUX Assistante de Gestion.

1.2 Chiffre d'affaires

Le chiffre d'affaires a évolué durant ces cinq dernières années. En effet, le chiffre d'affaires est passé de 280 000 € en 2017 à 400 000 € en 2021.

Cette hausse est due à la hausse du nombre de clients, les formations sont un succès malgré la période du Covid-19 car les formations ont su être adaptées en visio-conférence et cela est de même pour les formations postbac, plusieurs classes se sont ajoutées.

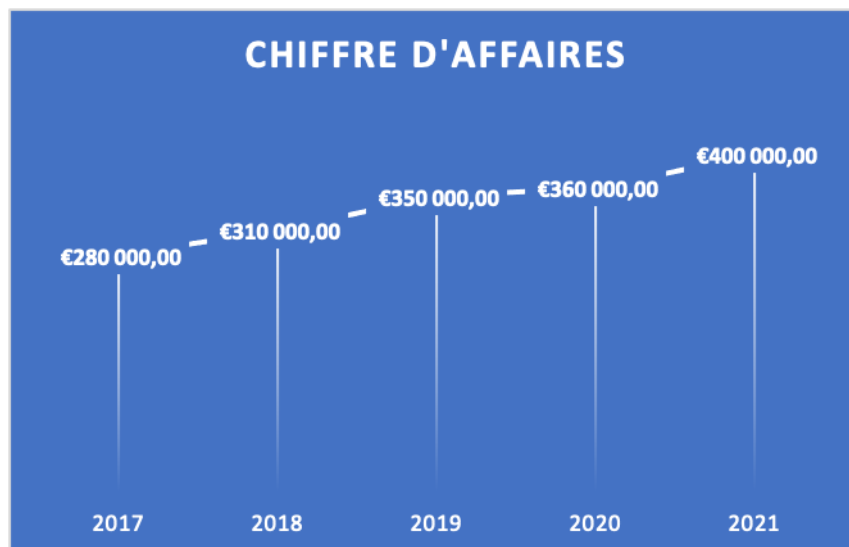


Figure 1 : Chiffre d'affaires

1.3 Clients et fournisseurs

Nos clients sont répartis en 2 catégories :

- Les clients des formations bureautique et informatique

Ces clients sont principalement des organisations, des entreprises ou associations qui veulent améliorer les compétences de leurs salariés ou des particuliers qui souhaitent tout simplement se performer sur un logiciel ou domaine précis. Ils viennent en général de Strasbourg et ses alentours.

- Les clients des formations postbac :

Ces clients sont des particuliers qui souhaitent acquérir un diplôme d'étude supérieur au bac. Ils peuvent suivre leur formation en initial, en formation professionnelle à temps partiel et avoir un travail à coté ou en alternance. Nos étudiants viennent de partout en France et de l'étranger.

L'IFIDE SUP'FORMATION n'a pas de fournisseurs spécifiques mise à part le matériel administratif (fournit par Bureau Vallée) et les services d'entretien.

1.4 Concurrence

Les concurrents directs sont ceux qui proposent le même type de service, d'une part les centres de formations privés proposant des diplômes de type BTS (Brevet Technicien Supérieur) et Bachelor (ESTUDIA, PIGIER, ESGM, IESA, CESI, etc...) et d'une autre part les centres de formations publics proposant des formations diplômantes (lycées et CFA). Étant donné que les étudiants viennent de toutes la France et de l'étranger, cette concurrence est nationale à travers toutes les écoles, lycées et CFA soit, environ 3 500 établissements.

Les concurrents de formations bureautiques sont de manière générale des centres de formation proposant les mêmes formations que le centre IFIDE Sup' Formation (Epitech, 3WA Academy, EPSI, etc...). La localité de ses concurrents se situe sur toute la région du Grand-Est, ils sont environ une vingtaine.

Par ailleurs ses concurrents indirects, sont ceux qui proposent des formations différentes mais pouvant se substituer aux nôtres, comme les centres de formations privés et publics proposant des diplômes similaires, dont notamment l'Université de Strasbourg. En termes de formation bureautique, nos concurrents indirects sont principalement les sites de formation en ligne, comme les plateformes de e-learning tel que le CNFDI par exemple.

1.5 Management

Chaque site à son indépendance. Le management est organisé autour de Monsieur Didier MOSIN, directeur du site d'Eckbolsheim.

Le style de management est le management participatif, en effet Monsieur MOSIN laisse ses salariés s'organiser eux-mêmes sur leurs tâches et missions de travail tout en les accompagnants.

Ce style de management permet aux salariés d'être impliqué dans le développement et la prise de décision de l'entreprise. Cependant le management participatif a pour risque de créer une désorganisation si les salariés ne fournissent pas le travail nécessaire et que le manager n'est pas assez formé.

Monsieur MOSIN adopte également un management paternaliste. Ici, le directeur fait preuve d'autorité dans les décisions mais aussi d'une grande bienveillance envers ses salariés, notamment pour ce qui concerne leurs conditions de travail. Il est attentif aux difficultés et besoins des employés y compris en dehors du temps de travail. Ce qui crée une véritable complicité au sein de l'équipe.

2 Etude de l'existant

Il y a une application à l'IFIDE qui s'appelle GPCI, qui permet aux formateurs de renseigner leurs indisponibilités.

Cette application est complémentaire avec le fichier Excel et avec le système de planning de l'IFIDE ; elle permet de visualiser les conflits et de les résoudre.

Cette application fonctionne avec d'anciennes versions de PHP et Angular JS et M. MOSIN exige l'utilisation de ces anciennes versions tout simplement parce que cela fonctionne.

1. Environnement de développement

L'application GPCI ayant été créée en 2015, il m'a fallu adapter les versions des langages utilisés. Les outils et les procédures utilisés pour créer un environnement de développement de projet fonctionnelle seront décrites dans cette partie.

2.1.1 Installation et configuration de la machine de développement

- **Installation**

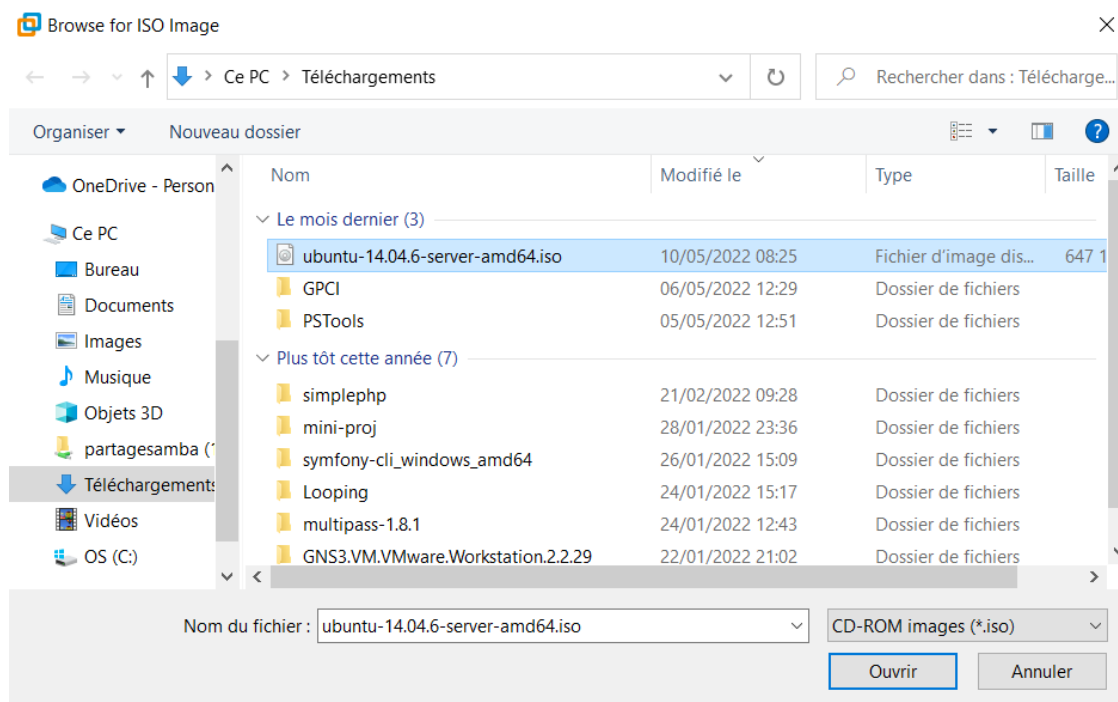


Ci-dessous les étapes d'installation de la machine virtuelle Ubuntu version 14.04.

Etape 1 : début de l'installation avec VMWare

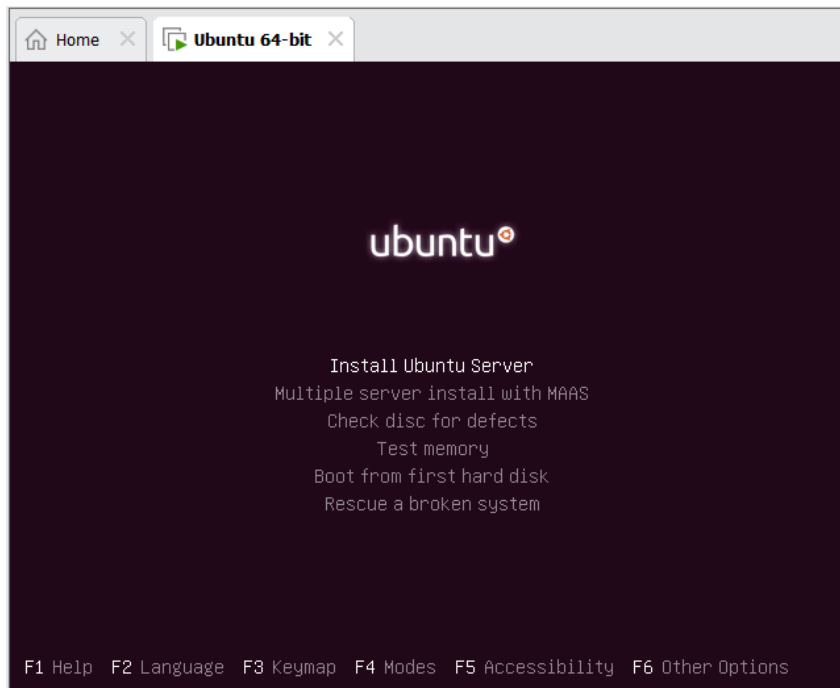


Etape 2 : Insérez un disque pour l'installation de Ubuntu 14.04 et redémarrez votre ordinateur.



Etape 3 : Ensuite, appuyer sur la touche Entrée pour démarrer l'installation. Et puis vous allez sélectionner votre pays pour le fuseau horaire, sélectionnez la région etc...

Etape 4 : L'installation est terminée. Le système redémarre.



Etape 5 : Le système démarre et affiche l'invite de connexion comme ci-dessous. Connectez-vous avec un utilisateur que vous avez configuré lors de l'installation

```
Ubuntu 14.04.6 LTS GPCI tty1
GPCI login:
```

- **Serveur web (apache2)**

Apache2 est un serveur HTTP, c'est à dire un module chargé de recevoir et de renvoyer des données selon le protocole HTTP. En d'autres termes, le navigateur internet de l'internaute envoie des requêtes à apache2 et en retour apache2 renvoie les données des pages à afficher du site internet que l'internaute est en train de consulter.

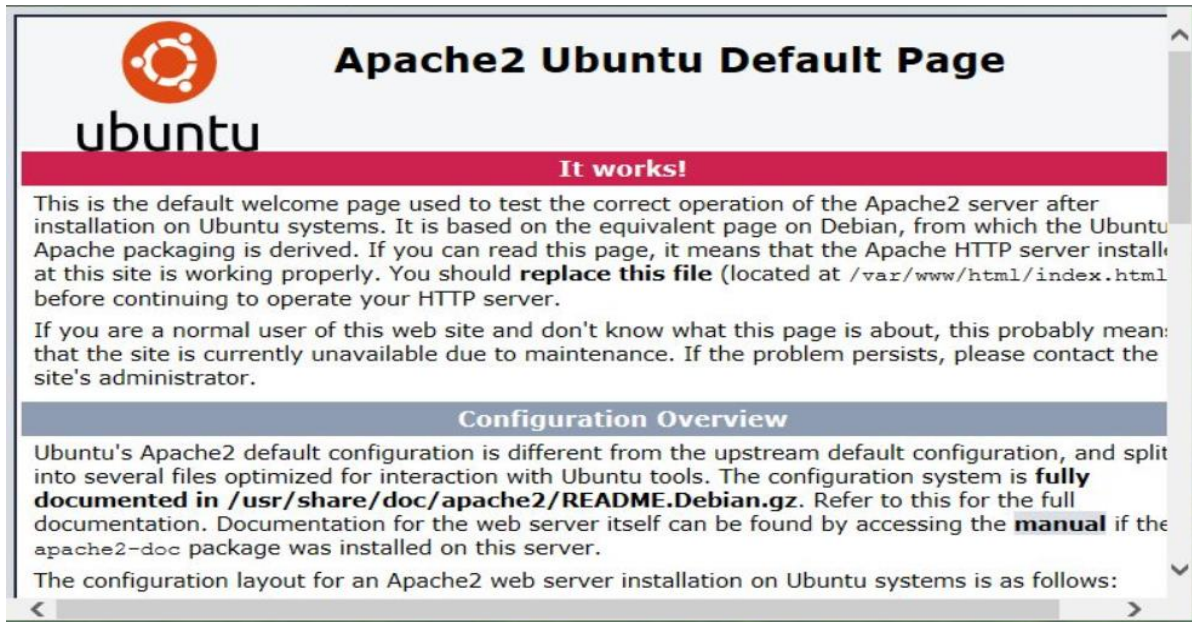
Cette commande sert à installer le serveur apache en précisant la version qu'on veut.

```
mane@GPCI:~$ apt-get -y install install apache2
```

Cette commande permet de redémarrer le serveur.

```
mane@GPCI:~$ /etc/init.d/apache2 restart
```

Accès à "http://(nom d'hôte ou adresse IP de votre serveur) /" avec le navigateur web. Il est OK si la page suivante est affichée. (Page par défaut)



- **Base de données**

- **MySQL**

MySQL est un serveur de base de données distribué sous licence libre GNU (General Public License). Il est, la plupart du temps, intégré dans la suite de logiciels LAMP qui comprend un système d'exploitation (Linux), un serveur web (Apache) et un langage de script (PHP).

Cette commande sert à installer MySQL en précisant la version qu'on veut.

```
mane@GPCI:~$ apt-get -y install mysql-server-5.6
```

- **PhpMyAdmin**

PhpMyAdmin est une interface permettant d'exécuter facilement des requêtes SQL comme par exemple le CRUD (create, read, delete, update). Il est également possible d'y exporter ou importer une base de données.

Cette commande sert à installer PhpMyAdmin.

```
mane@GPCI:~$ apt-get -y install phpmyadmin
```

Accès à "http://(votre nom d'hôte ou adresse IP)/phpmyadmin/" et connexion avec mon utilisateur dans MySQL.



The image shows the phpMyAdmin login page. At the top is the phpMyAdmin logo. Below it, the text "Bienvenue dans phpMyAdmin" is displayed. There are two main sections: "Langue - Language" with a dropdown menu set to "Français - French", and "Connexion" with fields for "Utilisateur :" (containing "mane") and "Mot de passe :". An "Exécuter" button is at the bottom right.

2.1.2 Serveur Samba



Samba est un progiciel qui donne aux administrateurs réseau flexibilité et liberté en termes d'installation, de configuration et de choix de systèmes et d'équipements. En raison de tout ce qu'il offre, Samba a gagné en popularité, et continue de le faire, chaque année depuis sa sortie en 1992. Il permet principalement le partage de dossiers dans un réseau local.

Le **partage de dossiers et d'imprimantes dans un réseau local** est une fonctionnalité des systèmes d'exploitation modernes permettant d'accéder à des ressources d'un ordinateur (dossiers de données et imprimantes) à partir d'un autre ordinateur situé dans un même réseau local (réseau domestique ou d'entreprise).

J'ai utilisé ce logiciel pour pouvoir travailler de mon poste (ordinateur) et avoir accès aux dossiers sur ma machine virtuelle. Ci-dessous est décrit les étapes d'installation et de sa mise en fonction.

Premièrement on installe le logiciel avec cette ligne de commande.

```
mane@GPCI:~$ apt-get -y install samba
```

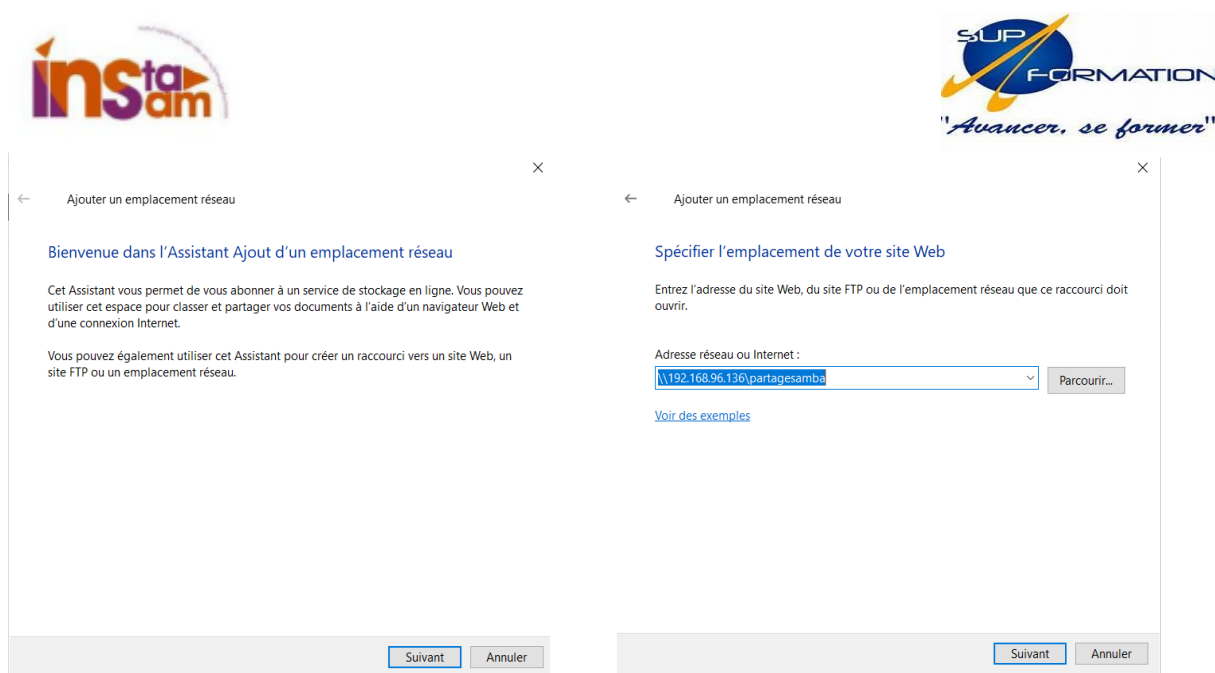
Une fois installé, on va commencer par la configuration du logiciel en accédant d'abord à son fichier de configuration (smb.conf).

```
mane@GPCI:/etc/samba$ nano smb.conf
```

On rajoute ces lignes au fichier smb.conf. Les paramètres ajoutés sont le nom (Partage Samba) que le dossier partagé va porter, un commentaire « Partage de mon dossier via samba », le chemin où notre est situé « /var/www/html » etc...

```
[PartageSamba]
comment = Partage de mon dossier via samba
path = /var/www/html
writable = yes
browseable = yes
guest ok = no
guest only = no
create mode = 0777
directory mode = 0777
share modes = yes
```

Pour accéder à mon dossier de mon ordinateur, je me rends dans l'explorateur des fichiers et j'ajoute un emplacement réseau en saisissant l'IP de ma machine virtuelle et le nom du dossier partagé.



Je saisis mon identifiant et le mot de passe créer pour accéder à mon dossier.

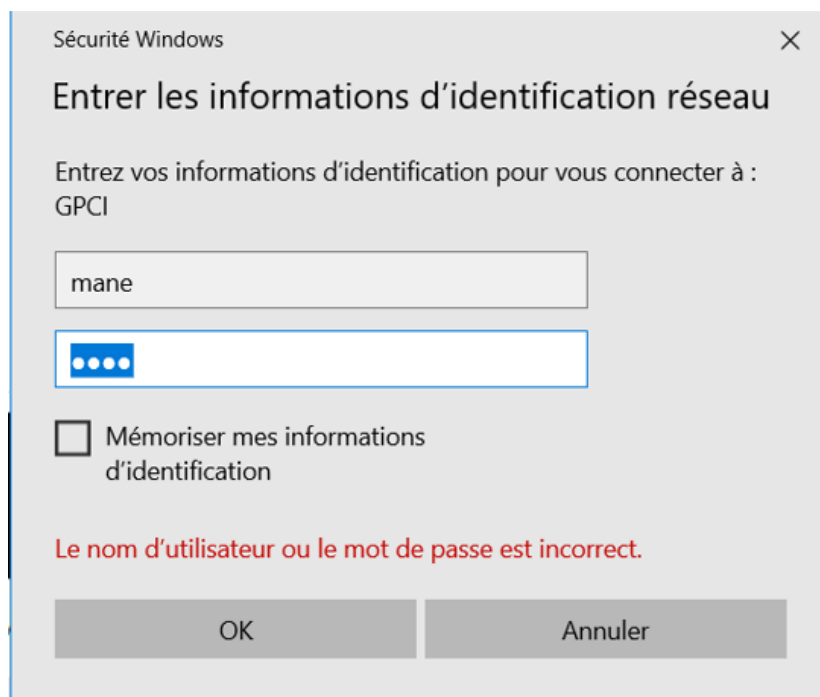
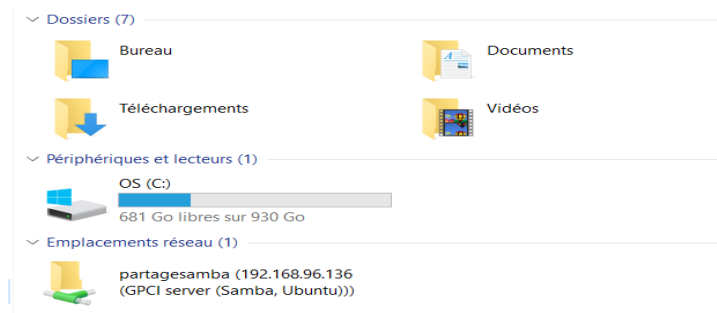


Figure 2 : Connexion à GPCI

Après identification on peut désormais accéder et modifier le contenu du dossier partagé.

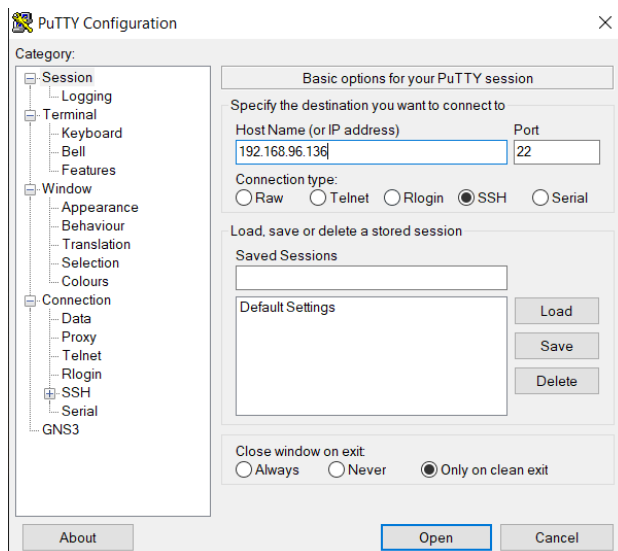


2.1.3 PuTTY



PuTTY est un client SSH et telnet, développé à l'origine par Simon Tatham pour la plateforme Windows. **PuTTY** est un logiciel open source disponible avec le code source et développé et pris en charge par un groupe de bénévoles. **PuTTY** est un émulateur de terminal doublé d'un client pour les protocoles SSH, Telnet, rlogin, et TCP brut. Il permet également des connexions directes par liaison série RS-232.

J'utilise **PuTTY** pour accéder à ma machine virtuelle via l'adresse IP.



Là, on est bien dans notre machine virtuelle. Je peux exécuter toutes les commandes que je souhaite et travailler avec ma machine virtuelle.



2.1.4 Github



J'ai utilisé GitHub pour collaborer avec mon tuteur. GitHub est un service web d'hébergement et de gestion de développement de logiciels, utilisant le logiciel de gestion de versions Git. Ce site est développé en Ruby on Rails et Erlang par Chris Wanstrath, PJ Hyett et Tom Preston-Werner.

Vous pouvez utiliser les produits GitHub pour améliorer votre processus de gestion logicielle et collaborer avec d'autres personnes.

2.1.5 Visual Studio Code



Visual Studio Code est un éditeur de code extensible développé par Microsoft pour Windows, Linux et macOS. Ce logiciel m'a servi d'éditeur de texte, je l'ai utilisé pour coder.

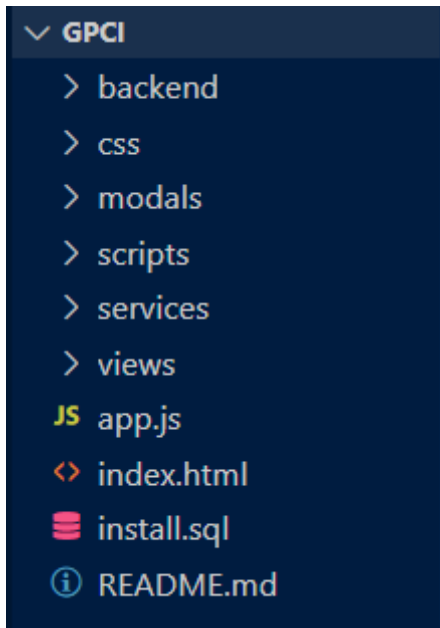


Figure 3 : Arborescence du code de l'application GPCI.

3 GPCI

1. Slim Framework



L'application GPCI est principalement développée avec la Framework SLIM.

- C'est quoi

Slim est un micro Frameworks PHP qui vous aide à écrire rapidement des applications web simples mais puissantes. C'est un outil idéal pour créer des API qui consomment, réutilisent ou publient des données. C'est également un excellent outil pour le prototypage rapide. Vous pouvez même construire des applications web complètes avec des interfaces utilisateur. Plus important encore, Slim est super rapide et a très peu de code.

- Comment ça marche

Tout d'abord, vous avez besoin d'un serveur web comme Apache. Vous devez configurer votre serveur Web pour qu'il envoie toutes les requêtes appropriées à un fichier PHP « front-controller ». Vous instanciez et exécutez votre application Slim dans ce fichier PHP.

Une application Slim contient des routes qui répondent à des requêtes HTTP spécifiques. Chaque route appelle un callback et renvoie une réponse HTTP. Pour commencer, vous devez d'abord instancier et configurer l'application Slim. Ensuite, vous définissez vos routes d'application. Enfin, vous lancez l'application Slim.

- Requête et réponse

Lorsque vous construisez une application Slim, vous travaillez souvent directement avec les objets requête et réponse. Ces objets représentent la requête HTTP réelle reçue par le serveur web et la réponse HTTP éventuelle retournée au client.

Chaque route d'application Slim reçoit les objets requête et réponse actuels comme arguments pour sa routine de callback. Ces objets implémentent les interfaces PSR-7. L'application Slim peut inspecter ou manipuler ces objets si nécessaire. En fin de compte, chaque route Slim doit retourner un objet PSR-7 réponse.

2. Backend (base de données)

3.1.1 Créer un user

D'abord, on crée un utilisateur avec tous les rôles (administrateur, planificateur et enseignant) ce qui va me permettre de voir l'interface de tous les utilisateurs quoiqu'elle soit leurs rôles.



Et puis, une fois l'utilisateur créé maintenant on peut utiliser l'application.

id	login	password	firstName	lastName	email
1	admin	6f2108a3551e39e4b75bedf4d3ed734a7c2aac44	Admin	ADMIN	admin@localhost

3.1.2 Les routes

Dans l'exemple ci-dessous on est dans le fichier **planificateur.php**, la méthode que cette route va utiliser est un « get ». La routine de callback de cette fonction a 3 paramètres.

- Le premier argument qui représente la requête HTTP courante.
- Le deuxième argument qui représente la réponse HTTP actuelle.
- Le troisième argument qui contient un tableau de valeurs pour les espaces réservés nommés de la route courante.

```

171 $app->get('/plan/matiere', $authenticateWithRole('planificateur'), function () use ($app) {
172     $matiere = Matieres::with('user')->get();
173     $app->response->headers->set('Content-Type', 'application/json');
174     $app->response->setBody($matiere->toJson());
175 });

```

3.1.3 Les requêtes

Ces objets représentent la requête HTTP réelle reçue par le serveur web et la réponse HTTP éventuelle retournée au client.

```

114 class Matieres extends Model
115 {
116     public $timestamps = false;
117
118     public function user()
119     {
120         return $this->belongsToMany('Users', 'users_matieres', 'id_Matieres', 'id_Users');
121     }
122     public function cours()
123     {
124         return $this->hasMany('Cours', 'id_Matieres');
125     }
126 }

```

3.1.4 Services et controllers

Un controller est une fonction PHP que vous créez qui lit les informations de la requête objet et crée et renvoie une réponse objet. La réponse peut être une page HTML, JSON, XML, un téléchargement de fichier, une redirection, une erreur 404 ou autre. Le contrôleur exécute la logique arbitraire dont votre application a besoin pour restituer le contenu d'une page.

[Voir le code en annexe page](#)

Les services sont utilisés pour le rendu des modèles, envoyer des e-mails, interroger la base de données et tout autre "travail" que vous pouvez y penser.

[Voir le code en annexe page](#)

Entre les routes et les requêtes il y'a les services et les contrôleurs. Pour qu'un évènement se passe le code sera exécuté sous cette ordre

ROUTE → SERVICE → CONTROLLER → REQUÊTE

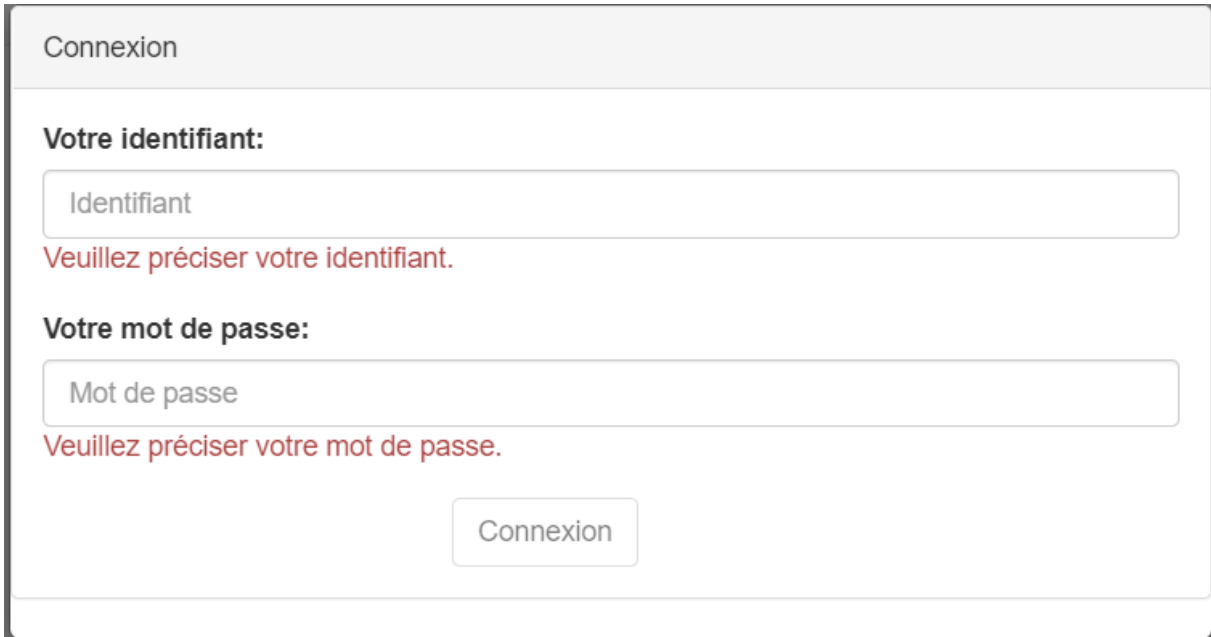
L'application se construit ensuite par inclusion des composantes dans le fichier App.js.

[Voir le code en annexe page](#)

3. Frontend (interface utilisateur)

3.1.1 Accueil

Tout d'abord il faut se connecter pour accéder à l'application.



The screenshot shows a login form with the following elements:

- Connexion** (Title)
- Votre identifiant:** (Label)
- (Input field)
- Veuillez préciser votre identifiant.* (Error message)
- Votre mot de passe:** (Label)
- (Input field)
- Veuillez préciser votre mot de passe.* (Error message)
- (Submit button)

Figure 4 : Connexion

Ensuite on la barre de navigation en haut avec les

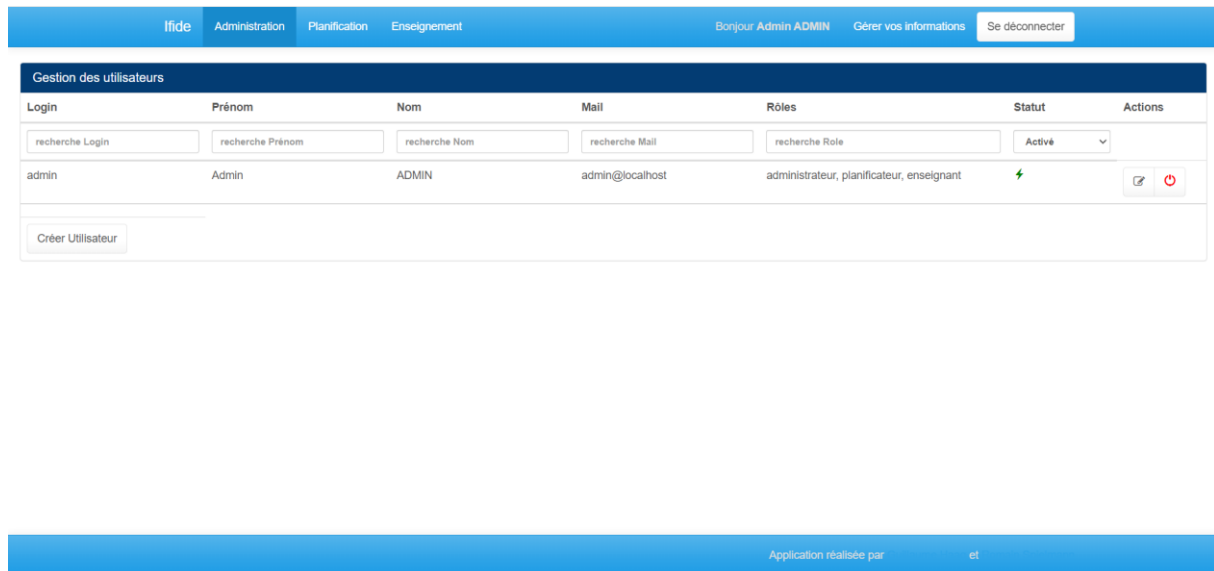


Bienvenue sur le planificateur des cours de l'lfide!

Figure 5 : Accueil

3.1.2 Administration

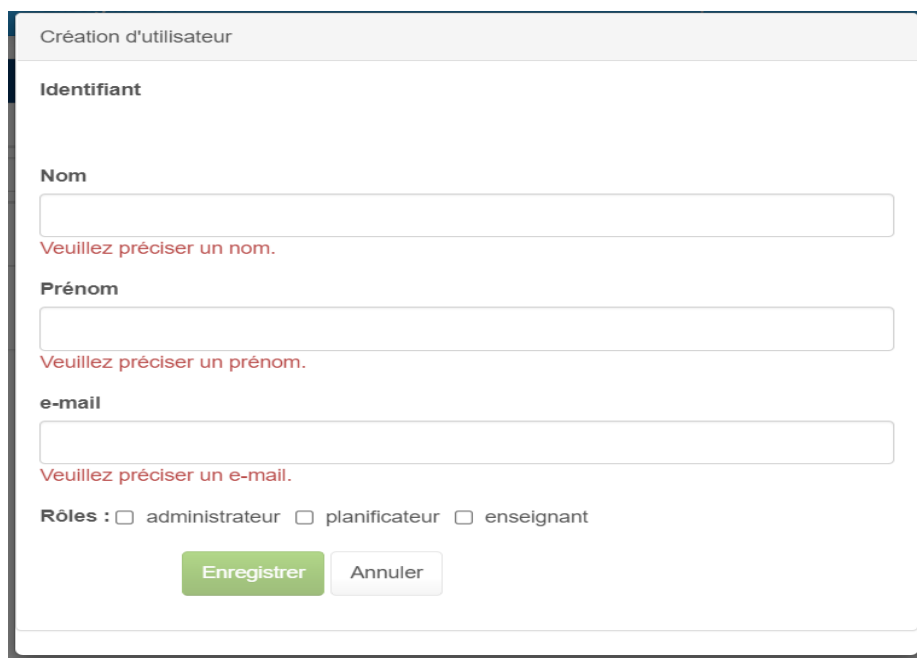
Après identification on a accède à la page d'accueil. Ayant tous les rôles, on va commencer par l'administration et ce que l'administrateur à accès sur l'interface de l'application.



The screenshot shows the 'Gestion des utilisateurs' (User Management) page. At the top, there's a navigation bar with tabs: 'Ifide', 'Administration', 'Planification', and 'Enseignement'. The 'Administration' tab is active. To the right of the tabs, it says 'Bonjour Admin ADMIN' and 'Gérer vos informations', with a 'Se déconnecter' button. Below the navigation bar, the page title is 'Gestion des utilisateurs'. There's a table with columns: 'Login', 'Prénom', 'Nom', 'Mail', 'Rôles', 'Statut', and 'Actions'. The table contains one user: 'admin' with first name 'Admin', last name 'ADMIN', email 'admin@localhost', roles 'administrateur, planificateur, enseignant', and status 'Actif'. There are search filters for each column and a 'Créer Utilisateur' button at the bottom left.

Figure 5 : Accueil administration

Dans la partie administration, on fait la gestion des utilisateurs ; voir, gérer tous les utilisateurs et ajouter les nouveaux utilisateurs.



The screenshot shows the 'Création d'utilisateur' (Create User) form. It has a title bar 'Création d'utilisateur'. Below it, there's a section 'Identifiant'. The form has three input fields: 'Nom', 'Prénom', and 'e-mail'. Each field has a red error message below it: 'Veuillez préciser un nom.', 'Veuillez préciser un prénom.', and 'Veuillez préciser un e-mail.' respectively. Below the input fields, there's a section 'Rôles' with three checkboxes: 'administrateur', 'planificateur', and 'enseignant'. At the bottom, there are two buttons: 'Enregistrer' (green) and 'Annuler' (grey).

Figure 7 : Création nouveau utilisateur

3.1.3 Planification

Ci-dessous on a le menu du planificateur, on va les parcourir tous pour découvrir leurs interfaces.

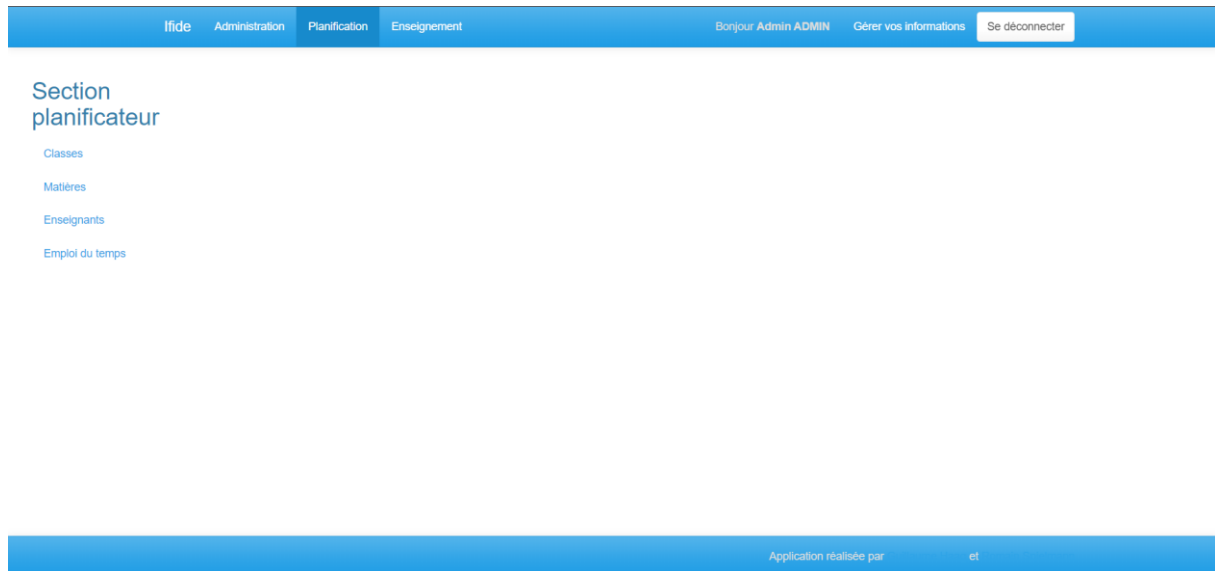


Figure 8 : Accueil planification

Le planificateur est responsable de la gestion des classes, des matières et des enseignants pour éviter un encombrement dans le planning.

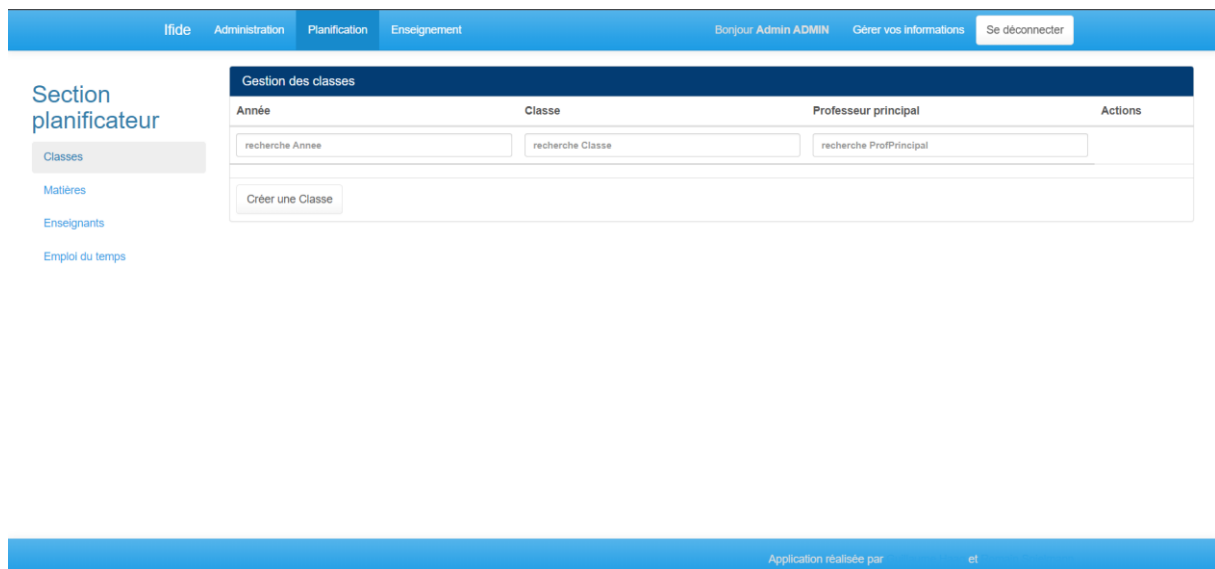


Figure 9 : Planification des classes

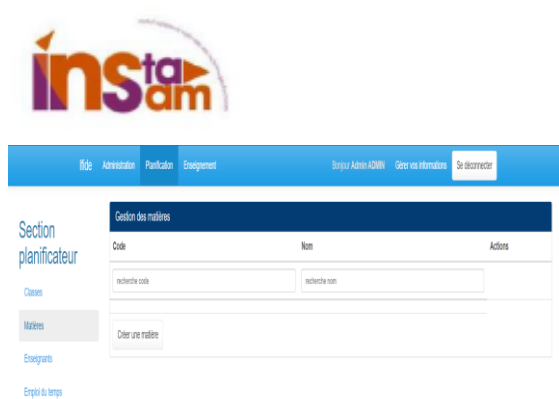


Figure 10 : Planification matières



Figure 11 : Planification enseignants

Quand le planificateur ajoute les, les classes et les enseignants. Cette partie consiste à les placer dans le planning.

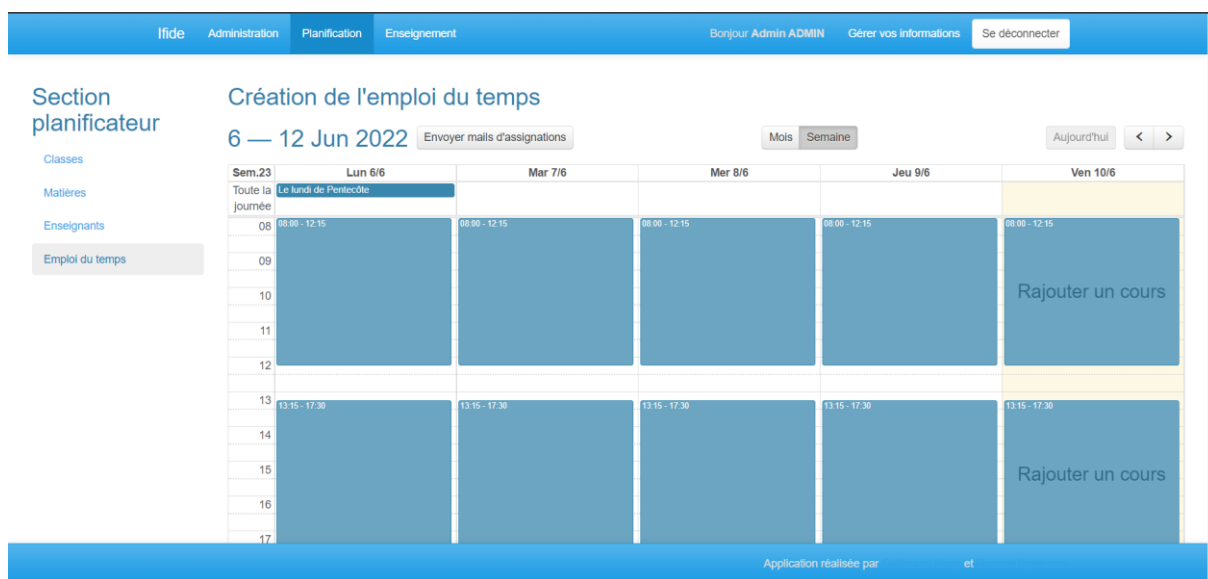
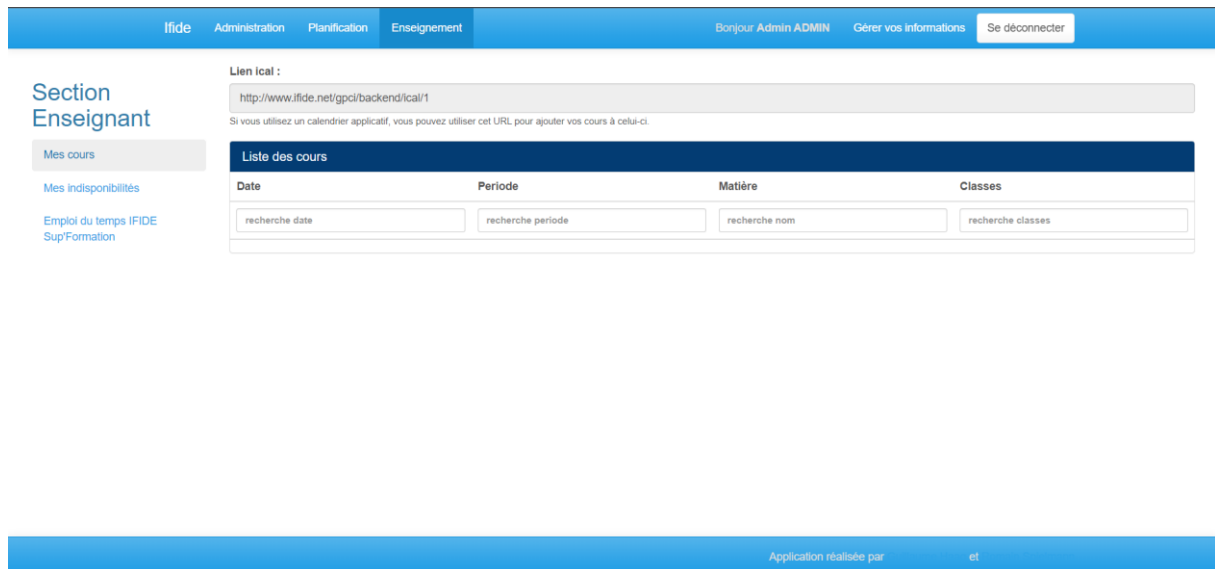


Figure 12 : Emploi du temps

3.1.4 L'enseignement

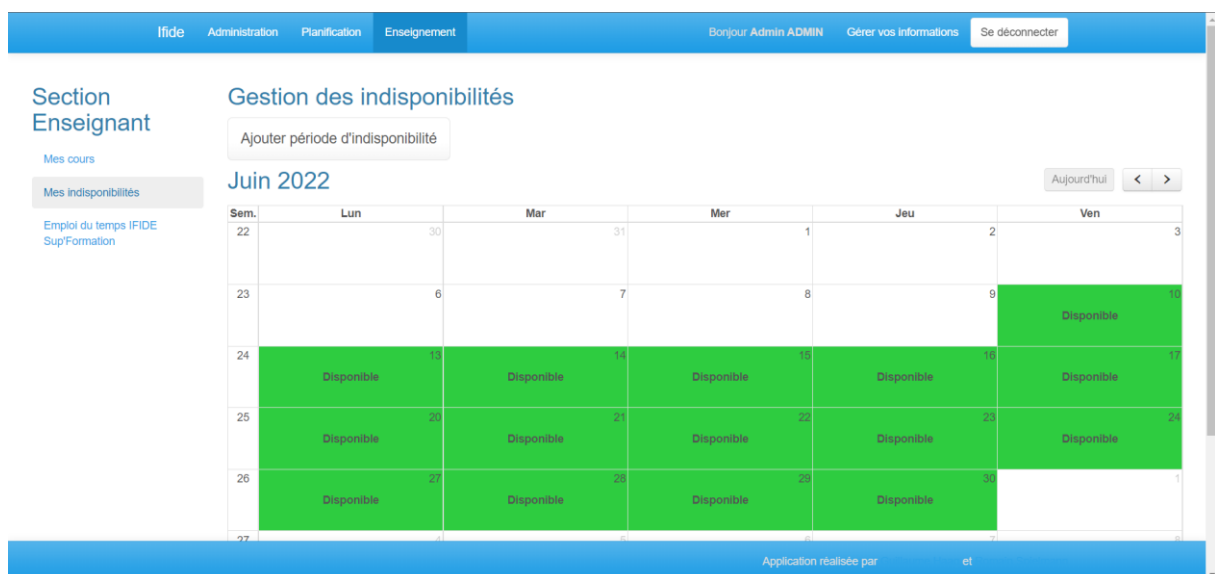
Dans l'interface des utilisateurs enseignants, ils ont accès aux cours déjà programmés, ils informent leurs indisponibilités et peuvent voir le planning des tous les classes.

Chaque enseignant aura sur cette sa liste des cours programmés.



The screenshot shows the 'Section Enseignant' interface. The top navigation bar includes 'Ifide', 'Administration', 'Planification', and 'Enseignement'. The 'Enseignement' section is active, showing 'Bonjour Admin ADMIN', 'Gérer vos informations', and 'Se déconnecter'. On the left, there's a sidebar with 'Section Enseignant' and links for 'Mes cours', 'Mes indisponibilités', and 'Emploi du temps IFIDE SupFormation'. The main content area is titled 'Liste des cours' and features a search bar with the URL 'http://www.ifide.net/gpci/backend/ical/1'. Below the search bar, there's a table with columns: 'Date', 'Période', 'Matière', and 'Classes'. Each column has a corresponding search input field labeled 'recherche date', 'recherche période', 'recherche nom', and 'recherche classes'.

Figure 13 : Accueil enseignement



The screenshot shows the 'Gestion des indisponibilités' interface. The top navigation bar is the same as in Figure 13. The sidebar on the left is also the same. The main content area is titled 'Gestion des indisponibilités' and features a button 'Ajouter période d'indisponibilité'. Below this, there's a calendar for 'Juin 2022'. The calendar has columns for days of the week (Lun, Mar, Mer, Jeu, Ven) and rows for weeks (Sem. 22, 23, 24, 25, 26, 27). The calendar cells are colored green and labeled 'Disponible'. The bottom of the interface shows 'Application réalisée par' and 'et'.

Figure 14 : Indisponibilités enseignants

Un enseignant peut aussi voir tous les cours programmés dans le centre.

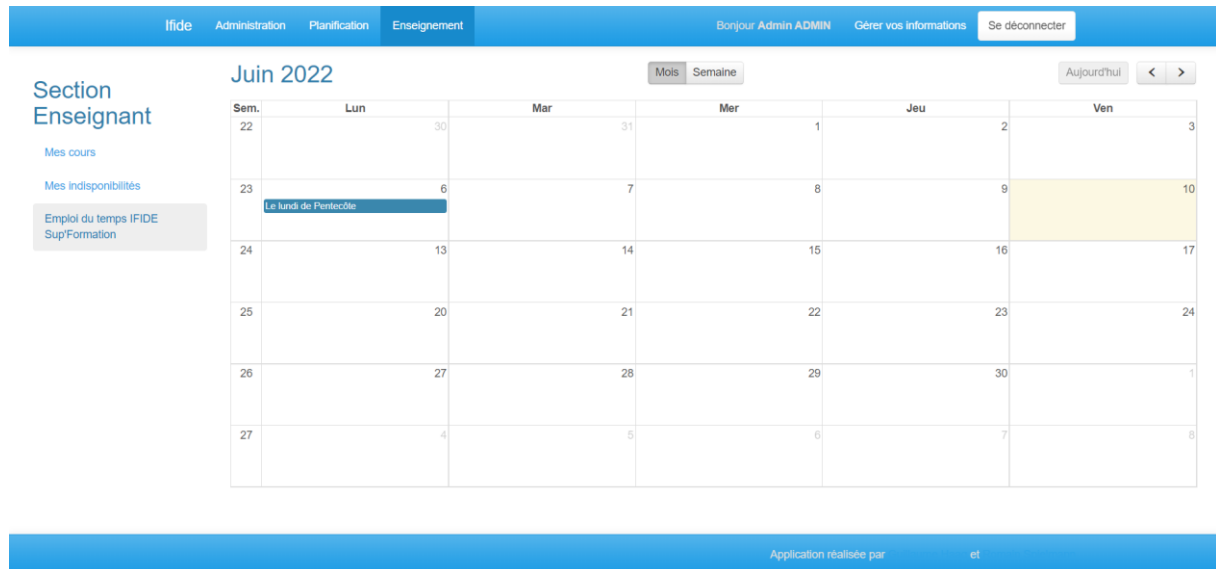


Figure 15 : Emploi du temps IFIDE Sup' Formation

4 Améliorations réalisées

L'application est opérationnelle mais plusieurs améliorations sont possibles. En particulier, dans l'interface où un formateur indique ses indisponibilités, le formateur ne voit pas les cours qui sont planifiés.

Une autre amélioration suggérée par M. MOSIN est la gestion des salles : pouvoir renseigner que tel cours est renseigné dans telle salle.

Dans cette partie on va montrer les améliorations qu'on a pu faire.

4.1.1 Gestion des salles

La gestion des salles est la principale amélioration faite sur l'application. Pour réaliser cette tâche, je suis partie du côté client vers la base de données (côté backend).

Etape 1 : créer une tables salles dans la base de données

id_Salles	nomSalle
s1	Salle1
s2	Salle2
s3	Salle3
s4	Salle4
s5	Salle5
s6	Salle6
s7	Salle7

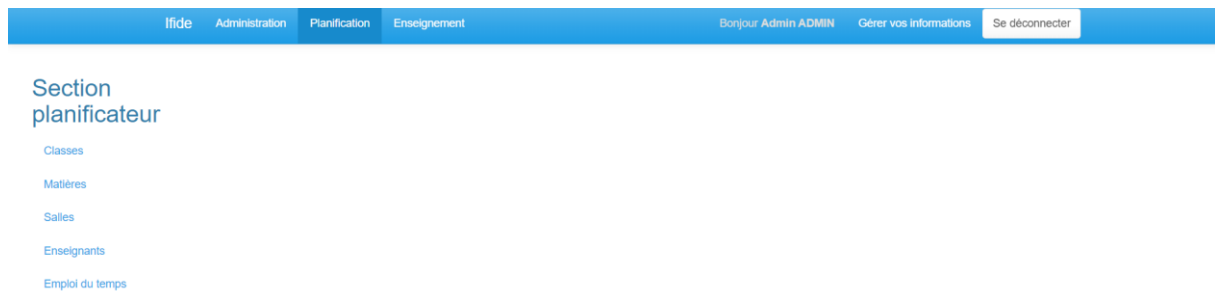
Etape 2 : ajouter une menue salle dans le planificateur (navigation.html)

```

14      <li>
15      |   <a ui-sref-active="active" ui-sref="planification.salles">Salles</a>
16      |
17      |   </li>
18      |
19      |   <li>

```

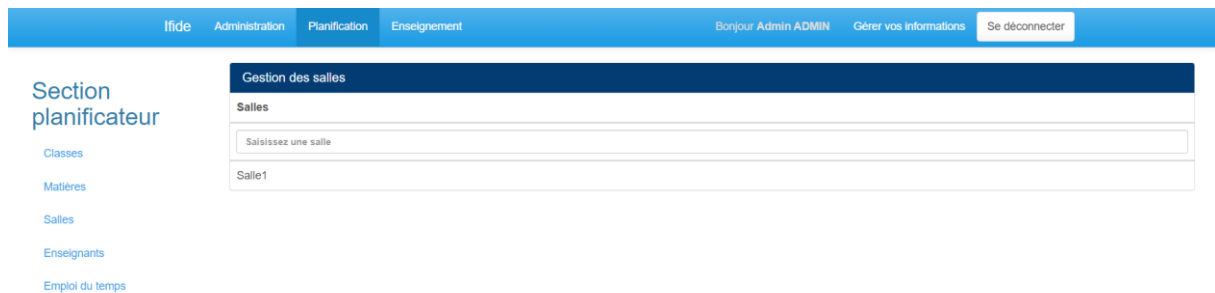
Résultat : on a une menu « Salles » ajouter au-dessous de « Matières »



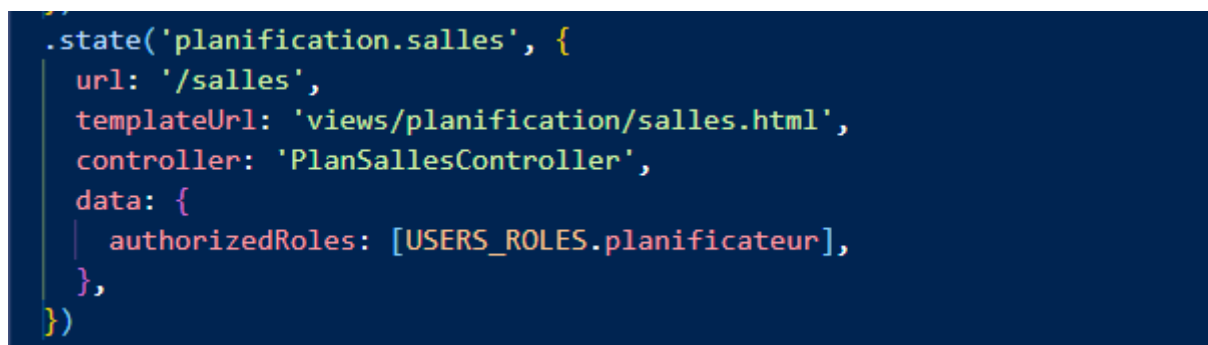
Etape 3 : construire un formulaire pour la gestion des salles

Voir le code en annexe p.38

Résultat :



Etape 4 : déclaration d'une route (url)



Etape 5 : créer un contrôleur

Voir le code en annexe p.39

Etape 6 : créer un service

Voir le code en annexe p.40

Etape 7 : Ajouter les salles dans le calendrier.

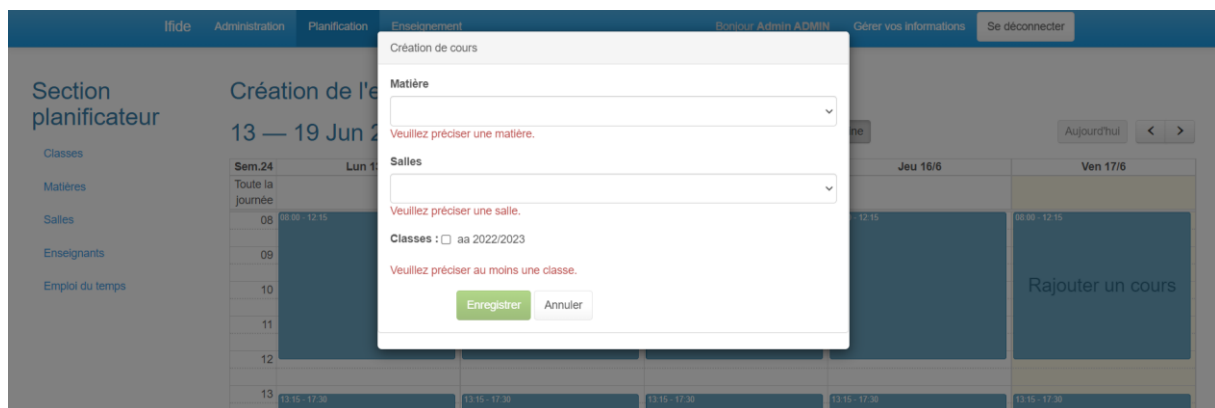
Pour faire cela, j'ai d'abord rajouté ce formulaire dans le fichier coursDetails.html

```

21 <div class="form-group">
22   <label>Salles</label>
23   <select
24     class="form-control"
25     name="nom"
26     ng-model="cours.nomSalle"
27     ng-options="salle.id as salle.nom for salle in salles"
28     required
29   ></select>
30   <span class="text-danger" ng-show="form.nom.$error.required"
31     >Veuillez préciser une salle.</span>
32   >
33 </div>

```

Résultat : un cours est bien associé à une salle.



Etape 8 : J'ai créé un objet « Class Salles », Cet objet représente la requête HTTP réelle reçue par le serveur web et la réponse HTTP éventuelle retournée au client.

```

/**
 * \class      Salles model.php "Backend/model.php
 * \brief      corresponding to the "Salles"
 * \details    corresponding to the "Salles". A "classe" is a classroom
 */
class Salles extends Model
{
    public $timestamps = false;
}

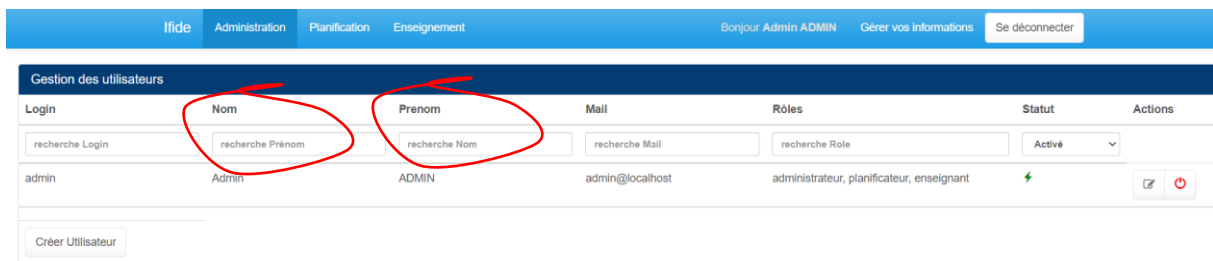
```




Etape 9 : Enfin, j'ai créé une route qui va répondre aux requêtes HTTP envoyés par l'objet Salles (Class Salles extends modal). Cette route appelle un callback et renvoie une réponse HTTP qui va retourner au client.

Voir le code en annexe p.42

4.1.2 Correction sur menu Administration

La première imperfection que j'ai constaté sur l'application était que le nom et le prénom étaient échangés dans les champs de saisie.



Login	Nom	Prenom	Mail	Rôles	Statut	Actions
admin	Admin	ADMIN	admin@localhost	administrateur, planificateur, enseignant	Actif	 

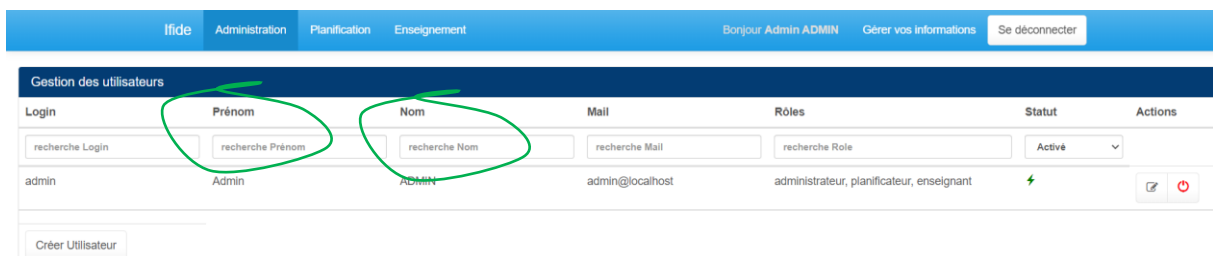
Pour remédier à ce problème, j'ai cherché dans la vue le fichier qui correspondait à cette interface et y apporter les changements que je voulais.



```

13      <tr>
14      <th st-sort="login">Login</th>
15      <th st-sort="firstName">Prénom</th>
16      <th st-sort="lastName">Nom</th>
17      <th st-sort="email">Mail</th>
18      <th st-sort="roles.role">Rôles</th>
19      <th st-sort="enabled">Statut</th>
20      <th>Actions</th>
21      </tr>

```

Après la modification du code je rafraîchis le navigateur web, on remarque que le changement a bien été accompli.



Login	Prénom	Nom	Mail	Rôles	Statut	Actions
admin	Admin	ADMIN	admin@localhost	administrateur, planificateur, enseignant	Actif	 

CONCLUSION (Version française)

L'application existe depuis 2015, elle est déjà fonctionnelle mais, il y avait besoin de l'optimiser. Les versions des langages et des serveurs utilisés pour développer l'application sont anciennes et les versions que j'utilise actuellement sur mon ordinateur sont différentes.

Alors j'ai créé un environnement de développement convenable au fonctionnement de l'application avec les bonnes versions.

L'environnement de développement installé, j'ai pu étudier l'application et j'ai fait une documentation sur les interfaces et code existants. La rédaction de la documentation m'a permis de bien comprendre sa structure de développement, ce qui m'a permis de reproduire le code pour rajouter d'autres interface et des fonctionnalités.

Pour l'instant, la fonctionnalité majeure ajouter à l'application est la gestion des salles. Je suis en train de travailler sur d'autres fonctionnalités pour améliorer l'application comme créer une interface qui permet et voir sur le planning les indisponibilités dans l'enseignement.

Ce stage m'a aidé à approfondir mes connaissances dans ce domaine et aussi découvrir plein d'outils et des méthodes sur le développement web.

CONCLUSION (English version)

The GPCI application exists since 2015, it is already functional but, there was need to optimize it. The versions of the languages and servers used to develop the application are old and the versions I currently use on my computer are different. So, I created a development environment suitable for the operation of the application with the good old versions.

Once I installed the development environment, I was able to study the application and I did a documentation on the existing graphical user interfaces and code. Writing the documentation allowed me to understand its development structure, which allowed me to reproduce the code and add other interfaces and other functionalities.

For now, the major feature I added to the app is room management. I am working on other features to improve the application such as creating an interface that allows teachers and trainers to see their unavailability on their schedules.

This internship helped me to deepen my knowledge in this field and discover many tools and methods on web development.

REFERENCES

https://www.server-world.info/en/note?os=Ubuntu_14.04

<https://docs.github.com/en/get-started>

<https://fr.wikipedia.org/wiki/GitHub>

https://fr.wikipedia.org/wiki/Visual_Studio_Code

<https://putty.org/>

<https://www.samba.org/samba/>

<https://www.slimframework.com/>

<https://www.journaldunet.fr/web-tech/dictionnaire-du-webmastering/1203595-mysql-my-structured-query-language-definition/>

<http://glossaire.infowebmaster.fr/css/>

LISTE DES FIGURES

Figure 1 : Chiffre d'affaires

Figure 2 : Connexion à GPCI

Figure 3 : Arborescence du code de l'application GPCI.

Figure 4 : Connexion

Figure 5 : Accueil administration

Figure 6 : Création nouveau utilisateur

Figure 7 : Accueil planification

Figure 8 : Planification des classes

Figure 9 : Planification matières

Figure 10 : Planification enseignants

Figure 11 : Emploi du temps

Figure 12 : Chiffre d'affaires

Figure 13 : Accueil enseignement

Figure 14 : Indisponibilités enseignants

Figure 15 : Emploi du temps IFIDE Sup' Formation

ANNEXES

Annexe 1 : code menu planification

```
views > planification > navigation.html > ...
1  <div class="container-fluid">
2    <div class="row">
3      <div class="col-md-2">
4        <h2>Section planificateur</h2>
5        <ul class="nav nav-pills nav-stacked">
6          <li>
7            <a ui-sref-active="active" ui-sref="planification.classes">Classes</a>
8          </li>
9          <li>
10             <a ui-sref-active="active" ui-sref="planification.matières"
11             >Matières</a>
12           >
13         </li>
14         <li>
15           <a ui-sref-active="active" ui-sref="planification.salles">Salles</a>
16         </li>
17         <li>
18           <a ui-sref-active="active" ui-sref="planification.enseignants"
19           >Enseignants</a>
20         >
21       </li>
22       <li>
23         <a ui-sref-active="active" ui-sref="planification.calendar({id:-1})"
24         >Emploi du temps</a>
25       >
26     </li>
27   </ul>
28 </div>
29 <div class="col-md-10" ui-view></div>
30 </div>
31 </div>
32
```

Annexe 2 : code pour créer un controller 'PlanSallesService'

```
scripts > controllers > planification > salles > JS salles.js
1  webApp.controller(
2    'PlanSallesController',
3    function ($scope, $uibModal, sallesService) {
4      $scope.salles = []
5
6      sallesService.getList().then(function (data) {
7        $scope.salles = data
8        $scope.sallesView = [].concat($scope.salles)
9      })
10
11     $scope.open = function (salle) {
12       //Il faut mettre l'idList dans $scope pour qu'il soit accessible dans resolve:
13       $scope.salle = salle
14
15       var modalDetails = $uibModal.open({
16         animation: true,
17         templateUrl: 'modals/sallesDetails.html',
18         controller: 'SalleDetails',
19         size: 'md',
20         resolve: {
21           salle: function () {
22             return $scope.salle
23           },
24         },
25       })
26
27       modalDetails.result.then(function (salleP) {
28         if (salleP.toDelete) {
29           $scope.remove(salleP)
30         } else {
31           sallesService.save(salleP).then(function () {
32             updateTable()
33           })
34         }
35       })
36     }
37
38     $scope.remove = function (salle) {
39       sallesService.remove(salle).then(function () {
40         updateTable()
41       })
42     }
43
44     function updateTable() {
45       sallesService.updatelist().then(function () {
46         sallesService.getList().then(function (salles) {
47           $scope.salles = salles
48           $scope.sallesView = [].concat($scope.salles)
49         })
50       })
51     }
52   }
53 )
54
```

Annexe 3 : code pour créer un service 'sallesService'

```
scripts > services > planification > JS salles.js > ...
1  webApp.factory('sallesService', function ($q, notifService, Restangular) {
2      var list = []
3
4      function updateList() {
5          return $q(function (resolve, reject) {
6              //TO DO Lancer toastr chargement
7              Restangular.all('plan/salle')
8                  .getList()
9                  .then(
10                     function (data) {
11                         list = [].concat(data)
12                         //TO DO SUCCESS TOASTR
13                         resolve()
14                     },
15                     function () {
16                         //TO DO ERROR TOASTR
17                         reject()
18                     }
19                 )
20          })
21      }
22
23      function getList() {
24          return $q(function (resolve, reject) {
25              if (list) {
26                  resolve(list)
27              } else {
28                  updateList().then(function () {
29                      resolve(list)
30                  })
31              }
32          })
33      }
34
35      function getOne(id) {
36          return $q(function (resolve, reject) {
37              //TO DO Lancer toastr chargement
38              Restangular.one('plan/salle', id)
39                  .get()
40                  .then(
41                     function (data) {
42                         //TO DO SUCCESS TOASTR
43                         resolve(data)
44                     },

```



```

45         function () {
46             //TO DO ERROR TOASTR
47             reject()
48         }
49     )
50 })
51 }
52
53 function getNew() {
54     return Restangular.one('plan/salle')
55 }
56
57 function save(salle) {
58     return $q(function (resolve, reject) {
59         notifService.saving()
60         salle.save().then(
61             function () {
62                 notifService.saved()
63                 resolve()
64             },
65             function (response) {
66                 notifService.error(response.data.message)
67                 reject()
68             }
69         )
70     })
71 }
72
73 function remove(salle) {
74     return $q(function (resolve, reject) {
75         notifService.deleting()
76         salle.remove().then(
77             function () {
78                 notifService.deleted()
79                 resolve()
80             },
81             function (response) {
82                 notifService.error(response.data.message)
83                 reject()
84             }
85         )
86     })
87 }

```

Annexe 4 : code pour créer une route 'Salles'

```

349 $app->get('/plan/salle', $authenticateWithRole('planificateur'), function () use ($app) {
350     $salles = Salles::all();
351     $app->response->headers->set('Content-Type', 'application/json');
352     $app->response->setBody($salles->toJson());
353 });
354
355 $app->get('/plan/salles/:id', $authenticateWithRole('planificateur'), function ($id) use ($app) {
356     $salles = Salles::where('id', $id)->first();
357     $app->response->headers->set('Content-Type', 'application/json');
358     $app->response->setBody($salles->toJson());
359 });
360
361 $app->post('/plan/salle/', $authenticateWithRole('planificateur'), function () use ($app) {
362     try {
363         $json = $app->request->getBody();
364         $data = json_decode($json, true);
365
366         $salles = new Salles();
367         $salles->classe = $data['classe'];
368         $salles->cour = $data['cour'];
369         $salles->nomSalle = $data['salle'];
370         $salles->save();
371     } catch (Exception $e) {
372         $app->response->headers->set('Content-Type', 'application/json');
373         $app->response->setStatus(400);
374         $app->response->setBody($e);
375     }
376 });
377
378 $app->put('/plan/salle/:id', $authenticateWithRole('planificateur'), function ($id) use ($app) {
379     try {
380         $salles = Salles::where('id', $id)->firstOrFail();
381         $json = $app->request->getBody();
382         $data = json_decode($json, true);
383
384         $salles->classe = $data['classe'];
385         $salles->cour = $data['cour'];
386         $salles->nomSalle = $data['salle'];
387         $salles->save();
388     } catch (Exception $e) {
389         $app->response->headers->set('Content-Type', 'application/json');
390         $app->response->setStatus(400);
391         $app->response->setBody($e);
392     }
393 });
394
395 $app->delete('/plan/salle/:id', $authenticateWithRole('planificateur'), function ($id) use ($app) {
396     try {
397         $cour = Salles::where('id', $id)->has('cour')->count();
398         if ($cour == 0) {
399             Salles::find($id)->delete();
400         } else {
401             $app->response->headers->set('Content-Type', 'application/json');
402             $app->response->setBody(json_encode(array("message" => "Cette salle est déjà associé à un cour!")));
403             $app->response->setStatus(400);
404         }
405     } catch (Exception $e) {
406         $app->response->headers->set('Content-Type', 'application/json');
407         $app->response->setStatus(400);
408         $app->response->setBody($e);
409     }
410 });
411

```

FICHE DE CONFIDENTIALITE

Titre du projet de fin d'études :

AMELIORATION DE L'APPLICATION GPCI (Gestion Planning Centre Formation IFIDE)

Nom & Prénom de l'apprenant :

ROBLE GUIRE Mane

Nom de l'entreprise d'accueil :

IFIDE SUP'FORMATION

☒ Projet de fin d'études non confidentiel☐ Projet de fin d'études à confidentialité limitée.

Le projet de fin d'études ne peut être lu que par des membres de l'école (intervenants et apprenants). Un

exemplaire est classé dans le dossier pédagogique de l'étudiant cité ci-dessus et un autre exemplaire est conservé à disposition d'autres futurs apprenants.

☐ Projet de fin d'études confidentiel.

Le projet de fin d'études est rendu à son propriétaire après présentation devant les membres du jury

chargé de son évaluation. Néanmoins, un exemplaire non communicable sera classé dans le dossier pédagogique de l'alternant cité ci-dessus.

☐ Clause de confidentialité propre à l'entreprise.

Fait à ECKBOLSHEIM, le 15 / 06 / 2022

Pour « l'entreprise »
(Nom et signature du
représentant)

DIEMER Michel



L'apprenant
(Nom et signature)

ROBLE GUIRE Mane



Pour l'école
(Nom et signature du
représentant)

Patrick GOURY



IFIDE SUP'FORMATION
2 Rue Ettore Bugatti - 67201 STRASBOURG

Tél. 03 88 10 83 84 - Fax 03 88 10 95 96
SARL au capital de 7 622,45 E
RCS Strasbourg 418 099 941 00075

FORMULAIRE D'ENGAGEMENT ANTI-FRAUDE

LA FRAUDE OU TENTATIVE DE FRAUDE CONSTITUE UNE INFRACTION QUI ENTRAINE UNE ELIMINATION IMMEDIATE ET DURANT 5 ANS A L'EXAMEN DE LA CERTIFICATION.

Constituent une fraude :

- La fraude à l'inscription : Consiste à un acte malhonnête fait dans l'intention de tromper en contrevenant au règlement afin de faciliter son admission à présenter la Certification Professionnelle « Responsable Marketing Opérationnel »
- La fraude aux évaluations :
 - La substitution de personne lors d'un examen ou d'un travail
 - L'exécution par une autre personne d'un travail
 - L'utilisation totale ou partielle d'un texte d'autrui (y compris image, tableau ou tout autre support) : plagiat
 - L'obtention par vol, manœuvre ou corruption ou par tout autre moyen illicite, de textes, tableaux, images
 - La possession ou l'utilisation du support d'un autre étudiant
 - La présentation à un examen d'un travail dont le contenu a été obtenu, en totalité ou en partie, par achat ou échange.
 - Vol de sujets, copies ou évaluations
 - Possession de documents ou appareils (téléphone, calculatrice...) non autorisés

Précisions concernant le plagiat : Il consiste à reproduire un texte, une partie d'un texte, toute production littéraire ou graphique, ou à paraphraser un texte sans indiquer quel en est l'auteur. Le plagiat enfreint les règles de la déontologie de l'enseignement supérieur et il **constitue une fraude** dans les travaux donnant lieu à notation. Le plagiat constitue également une atteinte au droit d'auteur et à la propriété intellectuelle, susceptible d'être assimilé à un délit de contrefaçon.

Lorsque l'auteur d'un travail universitaire éprouve le besoin de s'appuyer sur un autre texte, il doit le faire en respectant les règles suivantes :

- lorsqu'un extrait, même court, est cité exactement, il doit être placé entre guillemets (ou en retrait et en caractères légèrement plus petits si le texte fait plus de quelques lignes) et la référence (nom de l'auteur et source) doit être indiquée ; l'extrait cité doit être court ;
- lorsque le texte ou un passage du texte est paraphrasé ou résumé, la référence (nom de l'auteur et source) doit être donnée.

Ces obligations s'appliquent de la même manière en cas de textes originellement publiés sur internet et de traductions (originales ou non) ; elles concernent aussi les illustrations, tableaux et graphiques.

MODELE DE FORMULAIRE ANTI-FRAUDE

Je soussigné (nom, prénom) : ROBLE GUIRE Mane

apprenant(e) au sein de l'établissement : IFIDE Sup' Formation

déclare :

- avoir pris connaissance du formulaire d'engagement anti-fraude
- avoir rédigé mon PROJET DE FIN D'ÉTUDES dans l'intention d'en faire un travail personnel,
- m'engager à indiquer toutes les références des textes sur lesquels je m'appuierai

Fait à STRASBOURG, le 17/06/2022

Signature ROBLE GUIRE Mane