

## DEDICACE



Avec beaucoup de joie et  
d'affection, nous dédions ce  
travail à nos familles  
respectives.

## REMERCIEMENTS

C'est une tâche très agréable, mais bien délicate, de présenter nos remerciements à tous ceux qui nous ont aidé dans la réalisation de ce travail. En effet, vous êtes innombrable, vous qui avez sans cesse contribué à notre formation jusqu'ici et nous vous sommes très reconnaissant.

Nous voulons tout d'abord exprimer notre gratitude envers DIEU, pour Son Amour, Sa protection, la Sagesse, la Sante, le don de vie et le privilège de marcher jour après jour sous Sa grâce et Sa protection.

Nous ne saurons continuer cette rédaction sans remercier le Directeur de l'**Institut Universitaire de Technologies FOTSO Victor**, le Prof **TAMO TATIETSE Thomas** pour son aptitude et sa détermination à résoudre nos problèmes administratifs durant cette année académique, le Prof **TAYOU Clémentin**, chef du département de Génie Informatique de ladite institut, pour sa disponibilité et son encadrement durant ces trois années de formation.

Qu'il nous soit permis aussi, d'exprimer ici notre reconnaissance à l'endroit du **Dr FOTSING TALLA Bernard** et **M KONGNE Éric** pour leur encadrement de plus en plus croissant, leur dynamisme pour le traitement et l'examinassions de ce travail.

Nous tenons à exprimer nos affectueuses reconnaissances et profondes gratitudes à chacune de nos familles pour le soutien qu'elles nous ont accordées de bout en bout et tout au long de notre parcours.

## RESUME

Ce document présente de façon succincte le projet de fin d'étude (PFE) que nous avons réalisé tout au long de cette année académique. Ce projet a été effectué en vue de l'obtention d'une **Licence de Technologie**, option **Concepteur - Développeur, Réseaux et Internet (CDRI)**. Ce projet, qui vise à faciliter l'élaboration des emplois de temps au sein des écoles supérieures au Cameroun et partout dans le monde, est réalisé en deux versions : une version **web**, qui est principalement destinée aux paramétrages et à l'administration de l'application et une version **mobile**. Ce système, permet aux responsables de niveaux de mieux élaborer les emplois de temps des salles dont ils sont responsables. Il permet aux étudiants et enseignants de consulter aisément leur emploi de temps hebdomadaire et en toute quiétude. Il permet aussi aux enseignants de notifier leurs disponibilités de la semaine. Pour mieux assurer ces fonctionnalités, nous avons décomposé notre application en plusieurs modules à savoir : Emploie de temps, Validation, Publication, Confirmation et Statistique. Pour avoir rapidement un prototype de notre application, nous avons utilisé le cycle de vie en **spirale** avec la méthode **RUP<sup>1</sup>**. Nous avons utilisé le langage de modélisation UML pour modéliser les différents diagrammes durant la phase de conception ; Au cours de la phase l'implémentation, nous avons utilisé le langage de programmation Python avec son Framework Django pour le back-end ; et pour le front-end nous avons utilisé le **HTML**, le **CSS** et **JavaScript** ; enfin pour l'application mobile, l'IDE **Android studio**. L'architecture utilisée pour la partie web est le **MVT** (Model Vue Template) qui est une architecture en couche. Pour la partie mobile, l'architecture utilisée est le **client-serveur**. La réalisation de ce projet nous a permis de concilier la théorie apprise depuis ces trois années avec la pratique. Ainsi nous nous sommes familiarisés avec les nouvelles technologies de développement. Cette application élaborée est actuellement à la version 1.0 et cette version est assez stable et déjà utilisable. Néanmoins, la version la plus complète reste à venir.

**Mots Clés :** Emplois de Temps étudiant, Emplois de Temps professeur, Statistique, Matière, Ressource Critique

---

<sup>1</sup> RUP : Rational Unified Process

## LISTES DES FIGURES

Figure 1: Diagramme de cas d'utilisation .....	11
Figure 2: Diagramme de classe .....	14
Figure 3: DS: paramétrer le système .....	16
Figure 4: DS: réserver une double heure .....	17
Figure 5: DS: créer un nouveau planning.....	17
Figure 6: DS: faire des statistiques.....	18
Figure 7: Avantages de la méthode RUP .....	20
Figure 8: Cycle de vie en spirale .....	21
Figure 9: Architecture MVT.....	28
Figure 10: Architecture client-serveur.....	29
Figure 11: Emploi de temps vide.....	32
Figure 12: Recherche de la classe "GL" .....	33
<i>Figure 13: Interface de réservation des plages horaires .....</i>	33
Figure 14: Message affiché après réservation des plages horaires.....	34
Figure 15: Message affiché après réservation des plages horaires.....	34
Figure 16: Interface de consultation d'emploi de temps de l'étudiant .....	35
Figure 17: Authentification .....	35
Figure 18: Interface d'accueil      Figure 19: Interface d'accueil avec une notification .....	36
Figure 20:Dashbord: programme de la journee en cours      Figure 21: Dashboard (suite)....	37
Figure 22: Choix de la journée      Figure 23: choix de la semaine .....	38
Figure 24: Profile.....	38
Figure 25: Option du profil.....	38
Figure 26:Service POSTGRESQL .....	40
Figure 27: création de la base de données .....	40
Figure 28: déploiement d'Apache Tomcat .....	41
Figure 29:oploading du programme serveur .....	41
Figure 30: Choix de la matiere a programmer.....	XLIX
Figure 31: emploi de temps .....	XLIX
Figure 32: emplois en attente de plublication.....	XLIX

## SOMMAIRE

DEDICACE .....	i
REMERCIEMENTS .....	ii
RESUME .....	iii
LISTES DES FIGURES .....	iv
SOMMAIRE .....	1
INTRODUCTION GENERALE .....	3
CHAPITRE I: Description du projet.....	4
Introduction .....	4
I. Présentation de l'arrêté su projet .....	4
II. Position du problème.....	5
III. Etat de l'art .....	6
Conclusion.....	7
CHAPITRE II: Analyse Fonctionnelle et Techniques de Conception du Système .....	8
Introduction .....	8
I. Besoins fonctionnels et non-fonctionnels du système.....	8
II. Analyse Fonctionnelle.....	9
1. Le diagramme de cas d'utilisation .....	10
2. Le diagramme de classe .....	13
3. Le diagramme de séquence .....	15
4. Le diagramme d'activité.....	18
III. Technique de conception du système .....	18
IV. Présentation de l'équipe et planification .....	24
1. Présentation de l'équipe .....	24
2. Planifications.....	24
Conclusion.....	26

CHAPITRE III: Mise en œuvre : implémentation et résultats obtenus .....	27
INTRODUCTION .....	27
I.    Technique de développement.....	27
II.   Architecture et déploiement des applications.....	29
1.   Architecture générale du système.....	29
2.   Déploiement des applications.....	30
III.  Résultats concrets du développement .....	31
1.   Sur le côté web .....	31
2.   Sur le côté mobile.....	35
IV.   Hébergement du programme serveur de l'application .....	39
V.   Procédures d'hébergement .....	40
1.   Hébergement de la base de données.....	40
2.   Hébergement du programme serveur de l'application mobile .....	41
VI.   Evaluation des coûts .....	42
Conclusion.....	43
CHAPITRE IV: Limites du travail, propositions et suggestions.....	44
Introduction .....	44
I.    Apports, frein et limites du travail.....	44
1.   Apports du projet.....	44
2.   Freins et limites du travail.....	44
II.   Propositions et perspectives .....	45
Conclusion.....	45
CONCLUSION GENERALE .....	46
BIBLIOGRAPHIE ET WEBOGRAPHIE .....	XLVII
ANNEXE.....	XLIX
TABLE DES MATIERES.....	Erreur ! Signet non défini.

## INTRODUCTION GENERALE

Le présent travail s'inscrit dans le cadre de la réalisation d'un projet de fin d'étude en vue de l'obtention d'une Licence de Technologie, diplôme couronnant la fin du cycle de Licence à l'IUT-FV de Bandjoun. Il a été réalisé dans la période allant de **janvier à Juillet 2021**. Le projet proposé porte sur le Développement d'un système de gestion des emplois de temps dans les écoles supérieures au Cameroun et particulièrement à l'Institut Universitaire de Technologies FOTSO Victor (**IUT-FV**) de Bandjoun. En travaillant sur ce thème, notre souci est de mettre à la disposition des chefs d'établissements un système plus fiable d'élaboration des emplois de temps et facile à utiliser. Durant cette période de développement du système, nous avons bénéficié de l'encadrement et du suivi de Plusieurs personnes. Premièrement, ce travail était soumis aux corrections et appréciations du **Dr FOSTING TALLA Bernard**, enseignant à l'IUT-FV de Bandjoun, notre Encadreur Académique. Deuxièmement, nous avons bénéficié des orientations technologiques de **M. KONGNE Éric**, Responsable de la Cellule Informatique dudit établissement.

Le présent rapport intègre quatre chapitres. Le premier, intitulé « **Description du projet** », positionne le problème, donne l'arrêté du projet et présente l'état de l'art. Le second, intitulé « **Analyse fonctionnelle et techniques de conception** », présente les différentes phases et angles de conception du projet. Le troisième, intitule « **Mise en œuvre : Implémentation et résultats obtenu** », donne un aperçu des techniques de développement utilisées et les résultats obtenus. En fin, le dernier est intitulé « **Propositions / suggestions** ». Il présente les limites et perspectives de notre projet.

# CHAPITRE I: Description du projet

## Introduction

Dans ce chapitre introductif de ce document, nous nous attellerons à fournir toutes les informations que vous avez besoin, en tant lecteur, pour mieux comprendre les raisons d'être de ce thème. Pour cela, nous commencerons par présenter les circonstances qui ont conduits au choix de ce thème, nous poursuivrons cette rédaction une partie qui sera destinée à circonscrire le problème concret résolu, enfin, nous aurons un aperçu de ce qui a déjà été fait par nos prédecesseurs dans ce domaine.

### I. Présentation de l'arrêté su projet

Le commanditaire du projet, le Prof **TSAFACK NANFOSSO Roger**, recteur de l'université de DSCHANG, a exprimé le besoin d'une gestion plus optimale des emplois de temps au sein de ses campus, particulièrement celui de l'IUT-FV de Bandjoun. Plusieurs fois, il a remarqué des imperfections dans la gestion des plannings qui étaient élaborés au sein de cette établissement. Il désire désormais éviter toute incohérence au sein de ces derniers. Pour se faire, il a spécifié les besoins suivant :

- Développer un système qui permettra d'élaborer plus efficacement les emplois de temps au sein de cet établissement en permettant la communication entre Responsable ;
- Développer un système qui permettra un accès facile et rapide aux emplois de temps disponibles ;

- Permettre aux enseignants de réserver une plage horaire et/ou une ressource critique ;
- Développer un système qui permettra d'évaluer la progression des cours (par extension de ses fonctionnalités);
- Développer un système qui permettra aux parents de connaître les emplois de temps hebdomadaire de leurs enfants ;

## II. Position du problème

Dans les établissements primaires, secondaires et dans certaines grandes universités du Cameroun, les cours se font grâce à des emplois de temps **statiques**. C'est-à-dire le programme des cours reste le même au moins pendant tout un trimestre ou un semestre. Mais dans les universités cela est difficile à mettre en place à cause de la disponibilité des enseignants et des ressources critiques comme : les salles de classes, les vidéo projecteurs...

Au sein de chaque université ou grande école, nous avons plusieurs départements supervisés par un chef de département et qui regroupe plusieurs filières. Une filière est formée de plusieurs niveaux d'étude; chaque niveau est géré par un responsable de niveau. C'est à ce dernier qu'incombe la responsabilité d'élaborer les emplois de temps du niveau de la filière dont il est responsable. Cet emploi de temps sera validé par le chef de département avant d'être effectif et publié. Un enseignant peut intervenir dans plusieurs départements et filières différents.

Le problème qui se pose actuellement est que les responsables de niveau n'ont aucun moyen facile et rapide de communiquer entre eux et avec les enseignants. En effet le responsable de niveau n'a aucune connaissance du planning hebdomadaire d'un enseignant, a moins de contacter celui-ci s'il est joignable, au cas contraire, il ne peut pas savoir si un enseignant présent sur le campus a déjà été programmé dans une salle ou pas. Ce qui conduit au fait de programmer un même enseignant dans deux salles différentes aux mêmes heures d'un même jour. En conclusion, une salle est sacrifiée au profit d'une autre. Etant donné aussi qu'un enseignant à un volume horaire qu'il est tenu de remplir chaque année, toute heure supplémentaire qu'il fait au-dessus de ce volume horaire est comptabilisée et payé par l'établissement dans lequel il a effectué ces heures supplémentaires. Ainsi donc si le responsable de l'établissement vient à découvrir qu'il y'a des fois où un enseignant était programmé dans des salles différentes aux mêmes heures d'un même jour, ce dernier pourrait être considéré de fraudeur ; car le directeur pourra se dire « *Il est à la recherche des heures*

**supplémentaires** », pourtant c'était une erreur commise par les différents responsables de niveau. Pour donc pallier à ces problèmes, justifier les enseignants et assurer une meilleure planification des cours, nous avons donc décidé de mettre sur pieds un **site Web et d'une application mobile aux fonctionnalités adaptées permettra à l'IUTFV de Bandjoun d'assurer une cohérence entre les plannings qu'il produit chaque semaine.**

### III. Etat de l'art

**L'état de l'art**, autrement dit **analyse de l'existant** est une nécessité pour quiconque veut développer un système rivalisant et respectant un certain standard. L'état de l'art est un panorama synthétique et organisé des travaux qui ont déjà été réalisés. En d'autres termes, il renseigne sur les technologies existantes créées pour proposer des solutions à un type de problèmes particulier et fait le point sur les solutions déjà proposées. C'est dans cette optique que nous avons menés nos recherches qui ont abouties à des conclusions pertinentes au sujet de notre projet. En effet, jusqu'à nos jours, l'IUT-FV ne dispose pas encore d'un système lui permettant de gérer de façon efficace les emplois de temps au sein du campus. Les emplois de temps continus d'être élaborer de manière archaïque comme partout ailleurs. De plus il en ressort que **“emploidetemps.cm”** ne sera pas le seul ni le premier site web qui propose une plateforme de gestion des emplois de temps. Il existe déjà au moment où nous produisons cette application, plusieurs sites et applications qui proposent les mêmes services que nous voulons offrir. Nous pouvons citer entre autre :

- **EDT :** Qui est une application d'élaboration d'emplois de temps pour les collèges et lycées de France ;
- **UnDeuxTemps :** Qui est une application conçue pour aider les nouveaux chefs d'établissement à élaborer sans ambiguïté les différents emplois de temps qu'ils doivent faire ;

Toutes ces applications ont néanmoins un défaut : ils n'ont pris en compte que des réalités de vie en France. Dans le contexte et avec les réalités de vie au Cameroun et dans certains pays d'Afrique, ces applications deviennent obsolètes. D'où le besoin de développer une application qui réponde au mieux à nos attentes et s'adapte aux réalités de notre vie quotidienne.

## Conclusion

Tout au long de ce chapitre introductif, nous nous sommes investis pour faire comprendre les raisons d'être de ce projet. A présent, si vous avez lu ce document de bout en bout, vous avez les détails de la problématique, de l'origine du problème, ce qui a déjà été fait dans ce domaine pour tenter de solutionner le problème, les limites de ces solutions et la solution que nous proposons. Le chapitre qui suit portera sur l'analyse profonde de cette solution que nous apportons.

# CHAPITRE II: Analyse Fonctionnelle et Techniques de Conception du Système

## Introduction

Dans ce chapitre, nous allons nous attarder sur les détails de conception du système. Il est principalement destiné à montrer comment, en tant que spécialistes du domaine, nous avons agencés nos idées pour pouvoir atteindre nos objectifs. Donc ce chapitre vient apporter des détails explicatifs du fonctionnement du système sous tous les angles.

## I. Besoins fonctionnels et non-fonctionnels du système

Pour effectuer une analyse objective et conséquente du fonctionnement d'un système, il est impératif de connaître les besoins ou spécifications attendus au terme du développement. Ainsi donc, à la fin de ce projet, le produit logiciel qui sera mis en circulation devra remplir les spécifications suivantes :

- Sur le plan fonctionnel, il devra permettre de :
  - Paramétriser le système : <sup>2</sup>CRUD sur les départements, les filières, les modules, les matières, les ressources critiques...
  - Toujours dans le paramétrage du système : Gérer les différents profils ;
  - Permettre à un Responsable de Niveau de gérer un emploi de temps ;
  - Permettre à un enseignant de Solliciter une plage horaire

---

<sup>2</sup> CRUD : Create, Read, Update, Delete

- Permettre à un enseignant de réserver une ressource critique ;
  - Faire les statistiques de l'évolution des cours ;
  - Permettre à un parent de consulter l'emploi de temps de tous ses enfants
  - Permettre à un étudiant de créer un nouveau parent
  - Permettre à un Chef de Département de publier les emplois de temps en attente
  - Permettre à un délégué de la classe de confirmer l'effectivité d'un cours ;
  - Permettre à un enseignant de consulter son propre emploi de temps de la semaine en cour
  - Empêcher la programmation d'un enseignant dans plus d'une salle ou plus d'une matière à la même heure
- Sur le plan non-fonctionnel, il devra être :
- Sécurisé ;
  - Facile à utiliser ;
  - Web et mobile ;
  - Compatible avec plusieurs navigateurs ;
  - Ergonomique ;

## II. Analyse Fonctionnelle

L'analyse fonctionnelle du système est cette partie du travail réservée aux détails explicatifs du fonctionnement du système sous tous les angles, donc, de la modélisation.

La modélisation consiste à créer une représentation virtuelle de la réalité de façon à faire ressortir les points auxquels on s'intéresse. Dans notre projet, nous utilisons le langage de modélisation unifiée, de l'anglais **Unified Modeling Language** (UML), qui est un langage de modélisation graphique à base de pictogrammes conçu pour fournir une méthode normalisée pour visualiser la conception d'un système. Il est couramment utilisé en développement logiciel et en conception orientée objet. UML permet de spécifier, visualiser, modifier et construire les documents nécessaires au bon développement d'un logiciel orienté objet et offre un standard de modélisation international pour représenter l'architecture logicielle.

Ainsi pour mieux appréhender le fonctionnement de notre système, nous avons élaboré plusieurs diagrammes, chacun montrant le système sous un angle différent. Notamment, nous avons : le diagramme cas d'utilisations (CU), le diagramme de classe, le diagramme de séquence système (pour quatre cas d'utilisations), le diagramme d'activité (pour trois cas d'utilisations) et le diagramme de déploiement.

## 1. Le diagramme de cas d'utilisation

Le diagramme de cas d'utilisation est un diagramme UML qui a pour rôle de recueillir, d'analyser, d'organiser les besoins, et de recenser les grandes fonctionnalités d'un système. Il scinde les fonctionnalités du système en unités cohérentes : **les cas d'utilisation**, ayant un sens pour les acteurs. Les cas d'utilisation (représentés par une ellipse), permettent d'exprimer le besoin des utilisateurs du système. Les cas d'utilisations sont réalisés par des acteurs dits primaire (placés à droite système) ou secondaire (placés à gauche du système). Un acteur n'est primaire ou secondaire que par rapport à un cas d'utilisation.

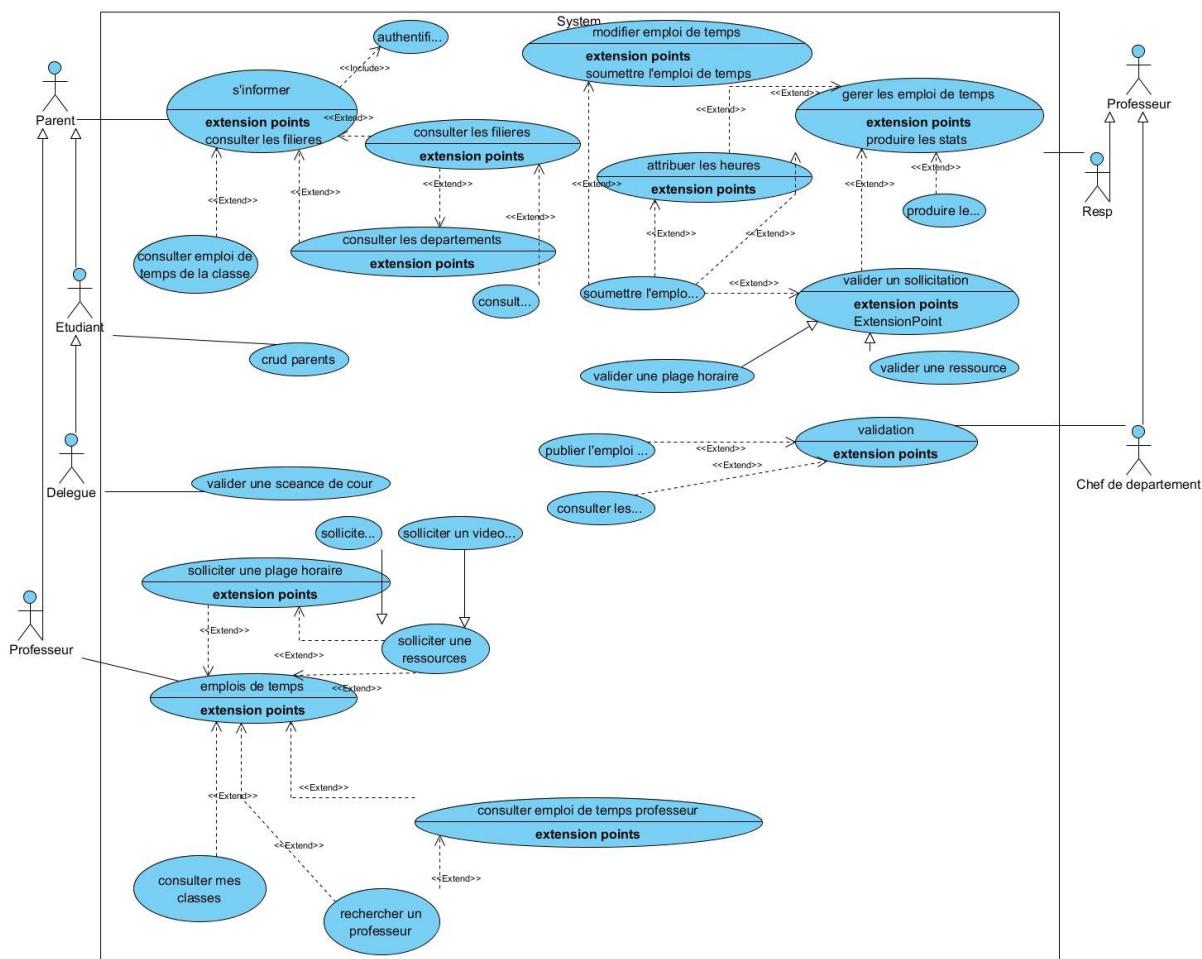


Figure 1: Diagramme de cas d'utilisation

## **Descriptions textuelle des cas d'utilisations**

➤ **Nom :** Consulter mon emploi de temps

- + **Acteur :** Etudiants ;
- + **Objectif :** Permettre à un étudiant de voir son programme de la semaine ;
- + **Précondition :** L'étudiant doit être inscrit dans la base de données, posséder un téléphone Android, une connexion internet, être connecté au site ou avoir la version mobile de notre application dans son téléphone ;
- + **Scenario :** L'étudiant clique sur le lien « **consulter mon emploi de temps** » et le système va chercher et afficher pour lui l'emploi de temps de la semaine s'il est déjà disponible ;
- + **Post-condition :** L'étudiant est désormais informé de son programme hebdomadaire ;

➤ **Nom :** Consulter l'emploi de temps d'une salle ;

- + **Acteurs :** Etudiant, Enseignant ;
- + **Objectif :** Permet à l'étudiant ou l'enseignant de voir l'emploi de temps publié d'une salle (au compte de la semaine en cours) ;
- + **Précondition :** Succès d'authentification ;
- + **Scenario :** L'intéressé sélectionne le département ensuite le niveau puis l'option (de la filière) / la salle dont il désire voir le programme de la semaine, puis clique sur le lien « **consulter l'emploi de temps** » et le système va chercher et afficher pour lui l'emploi de temps de la semaine s'il est déjà disponible ;
- + **Post-condition :** L'emploi de la salle/niveau choisie est affichée ;

➤ **Nom :** Consulter l'emploi d'un collègue ;

- + **Acteur :** Enseignant, responsable de niveau ;
- + **Objectif :** Permet à un enseignant de voir le planning hebdomadaire de son collègue ; Permet au responsable de niveau de voir la disponibilité notifiées d'un enseignant ;
- + **Précondition :** Succès d'authentification ;
- + **Scenario nominal:** L'enseignant saisie le nom complet de son collègue et une recherche sur le nom est effectuée, ensuite si ce dernier avait déjà notifié sa disponibilité de la semaine, celle-ci est affichée ;

- **Scenario alternatif :** Au cas où ce collègue n'a pas encore notifié sa disponibilité de la semaine, un message d'indisponibilité de planning est donc envoyé ;
  - **Post-condition :** Le planning hebdomadaire du collègue est affiché ;
- **Nom :** Valider une tranche horaire (cour effectif) ;
- **Acteur :** Délégué ;
  - **Précondition :** Succès d'authentification, succès de consultation de la liste des matières du jour;
  - **Scenario :** Le délégué coche la/les cases correspondantes, indiquant l'effectivité du cours ;
  - **Post condition :** Le champ « *CourEffectif* » de la table « *doubleHeure* » passe à « *True* » ;
- **Nom :** Réserver une plage horaire ;
- **Acteur :** Enseignant ;
  - **Objectif :** Donner la possibilité à un enseignant de solliciter une plage horaire pour son cours ;
  - **Précondition :** l'enseignant doit être connecté au site ou avoir la version mobile de l'application sur son portable et être déjà enregistré dans la base de données du système;
  - **Scenario :** l'enseignant sélectionne une salle, un jour ouvrable de la semaine et une ou plusieurs tranches horaires et sollicite dispenser son cours ce jour et à ces heures dans cette salle.
  - **Post-condition :** Les modifications appropriées sont apportées au planning hebdomadaire de l'enseignant ;
- **Nom :** Valider emploi de temps
- **Objectif :** Valider une proposition d'emploi de temps et le rendre consultable par les étudiants ;
  - **Acteur :** Chef de département ;
  - **Précondition :** Un emploi de temps doit avoir été élaboré et proposé ;
  - **Scenario :** le chef de département regarde et donne son approbation pour la publication du dit emploi de temps ;
  - **Post-condition :** l'emploi de temps est validé et mis à la disposition des utilisateurs du système (étudiants et enseignants) ;

➤ **Nom : Paramétrer le système**

- + **Acteur** : responsable de la cellule informatique ;
- + **Objectif** : rendre le système accessible et utilisable par les ayant droit ;
- + **Précondition** : le responsable de la cellule doit être connecté au site ;
- + **Scenarii** : il procède à l'ajout des salles, l'inscription des étudiants dans le système, l'ajout des enseignants, des ressources critiques... Il pourra dans le même temps effectuer des opérations telles que : supprimer, modifier les informations d'un utilisateur ou d'une ressource critique. Plus précisément, il est question ici d'attribuer les noms de rôles aux utilisateurs.
- + **Post-condition** : selon l'action menée, un utilisateur peut avoir été ajouté, modifié ou supprimé du système. De même qu'une ressource critique.

➤ **Nom : Publier un emploi de temps** ;

- + **Acteurs** : Responsable de niveau, Chef de département ;
- + **Objectif** : Mettre un emploi à la portée de tous ;
- + **Précondition** : L'emploi doit avoir été validé par le chef de département ;
- + **Scenario** : l'acteur qui réalise ce C.U clique sur le bouton « *publier* » ;
- + **Post-condition** : L'emploi est désormais consultable par les étudiants ;

➤ **Nom : Produire des statistiques** ;

- + **Acteurs** : Chef de département, responsable de la cellule informatique ;
- + **Objectif** : Donner les statistiques sur l'état d'avancement d'une matière, d'un module ou des cours d'un niveau ;
- + **Scenario** : L'acteur sélectionne la matière, le module ou la classe dont il souhaite produire une statistique sur l'état d'avancement des cours dispensés et clique sur le bouton « *voir les statistiques* » ;
- + **Post condition** : Une courbe ou une estimation en pourcentage du niveau d'avancement des cours lui est affichée ;

## 2. Le diagramme de classe

Le diagramme de classes est considéré comme l'un des diagrammes les plus importants de la modélisation orientée objet. Il fait partie des diagrammes statique d'UML, car il fait abstraction des aspects temporels et dynamique. Il montre la structure interne du système. Il permet de fournir une représentation abstraite des objets du système qui vont interagir ensemble pour réaliser les cas d'utilisations.

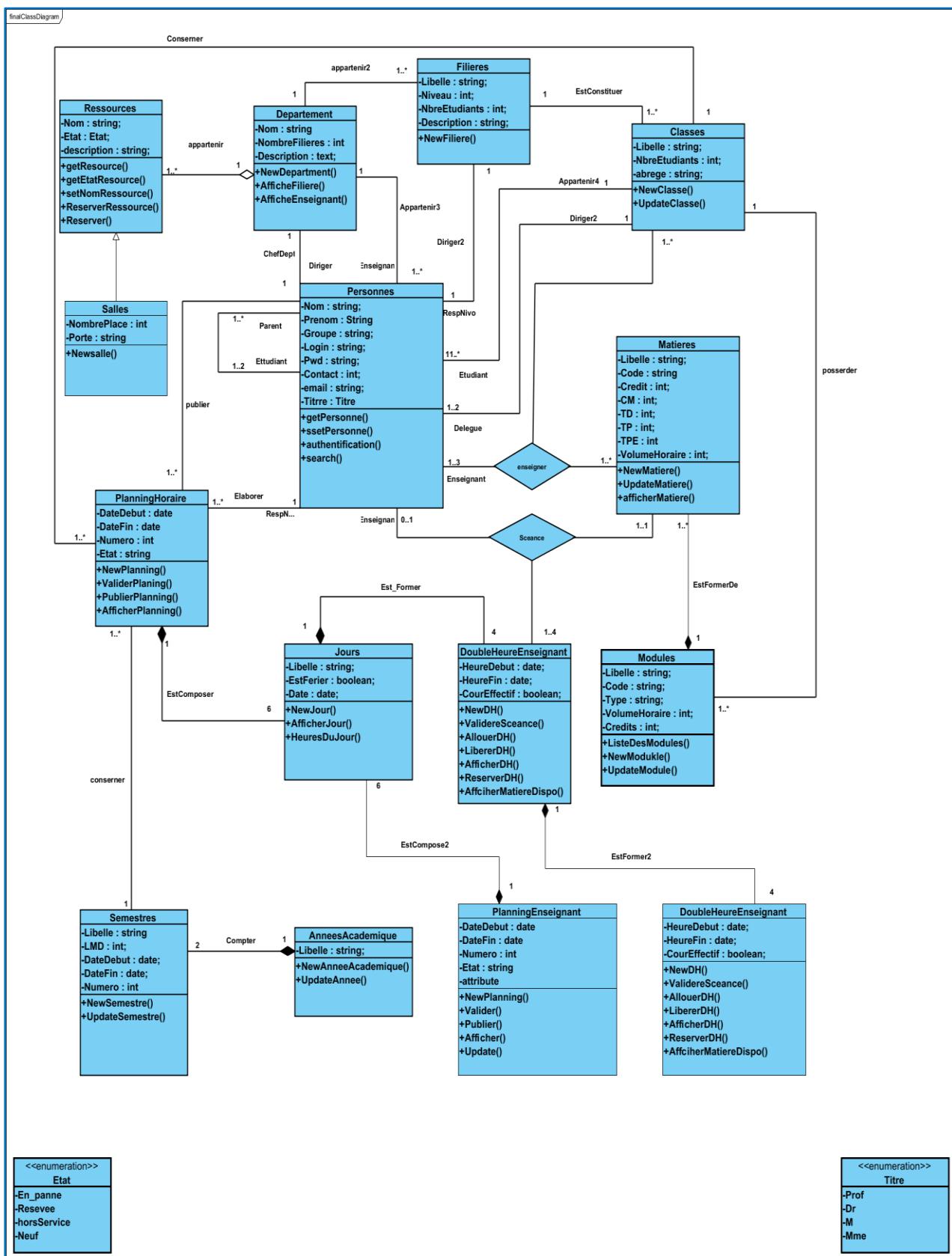


Figure 2: Diagramme de classe

### 3. Le diagramme de séquence

Le diagramme de séquence est une solution populaire de modélisation dynamique en UML. Il se concentre plus précisément sur les lignes de vie, les processus et les objets qui vivent simultanément ainsi que les messages qu'ils échangent entre eux. Il décrit comment et dans quel ordre les objets fonctionnent ensemble. Il est utilisé pour :

- Représenter les détails d'un cas d'utilisation ;
- Voir comment les objets et les composants interagissent entre eux pour effectuer un processus ;
- Modéliser le déroulement logique d'une procédure, d'une fonction ou une opération complexe ;
- Schématiser et comprendre le fonctionnement détaillé d'un scenario existant ou à venir ;

Pour notre projet, nous avons sélectionné quelques cas d'utilisations les plus pertinents et complexe pour réaliser leur diagramme de séquence afin d'enlever toute ambiguïté de compréhension dans l'esprit de tout un chacun.

### a) Cas : paramétrer le système

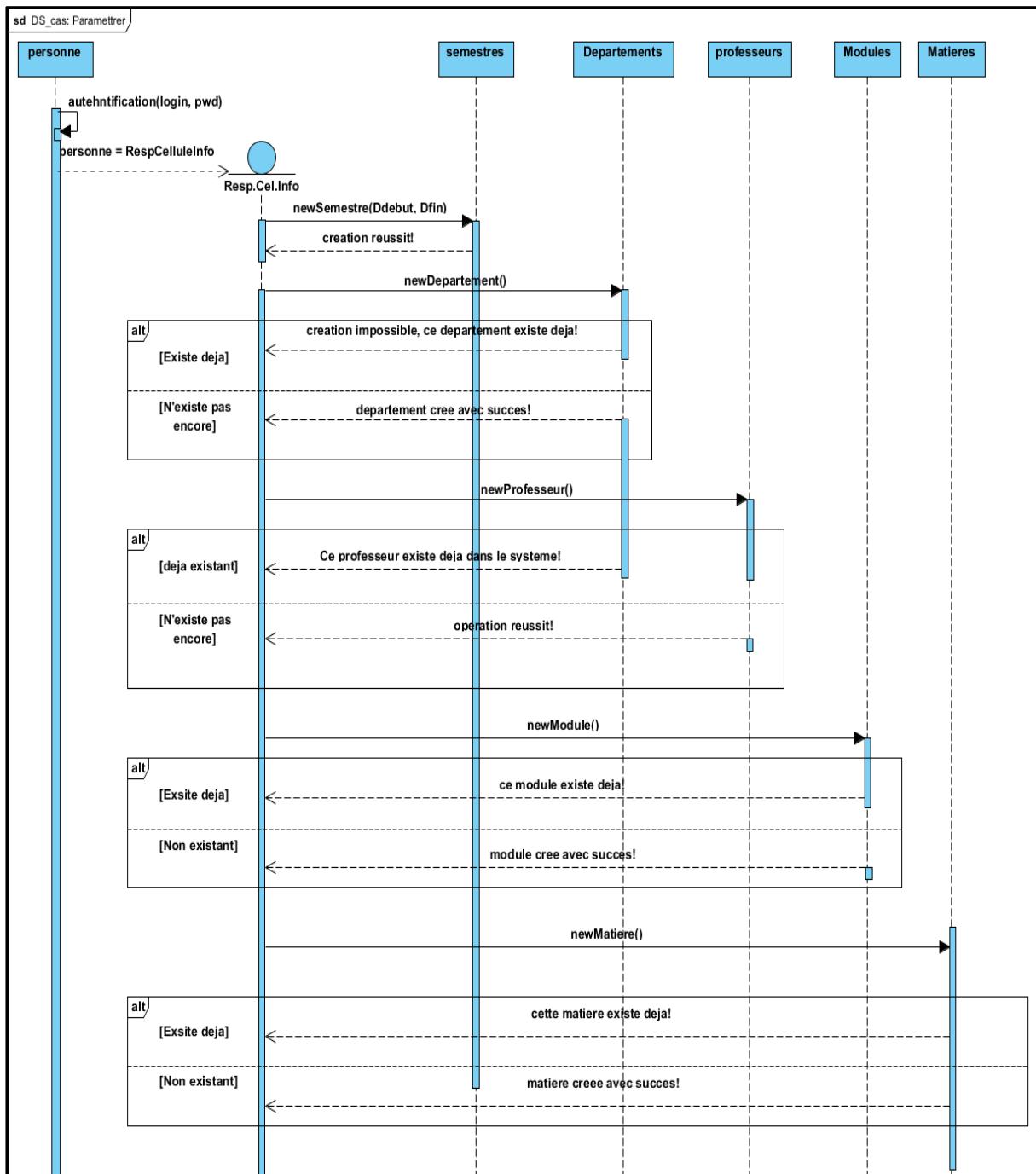


Figure 3: DS: paramétrer le système

### b) Réserver une double-heure

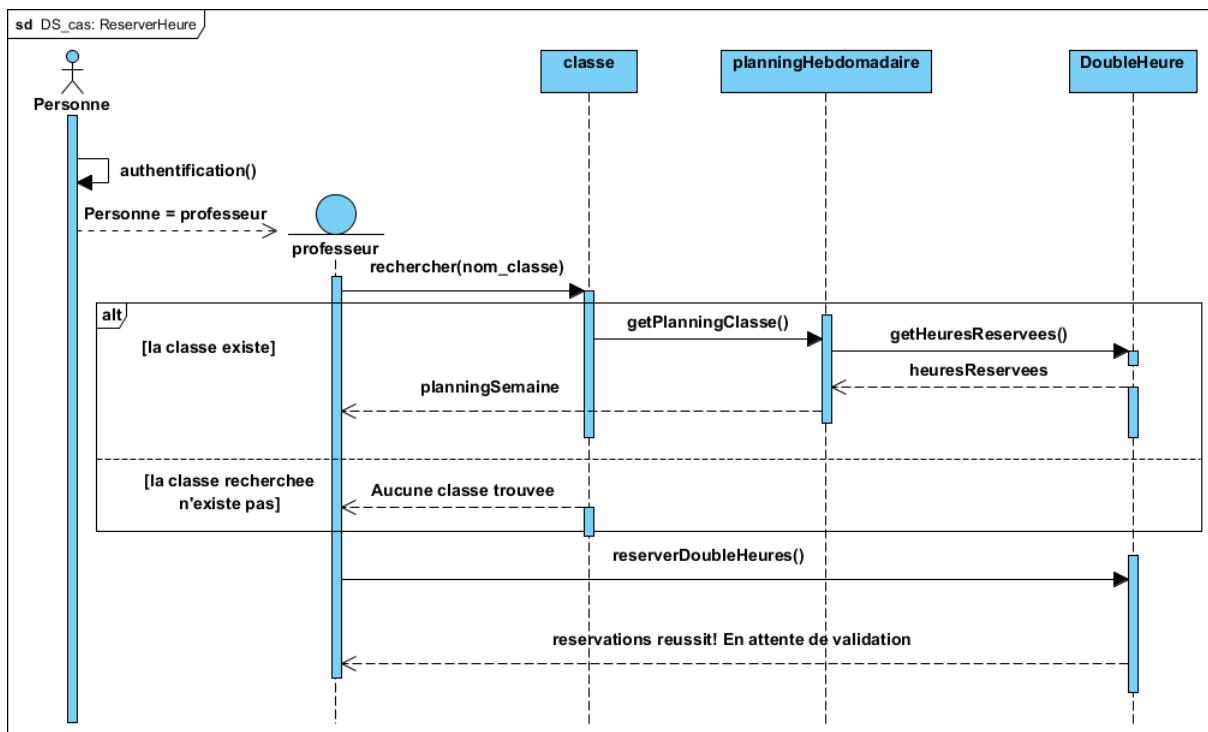


Figure 4: DS: résERVER une double heure

### c) Cas : Elaborer un planning

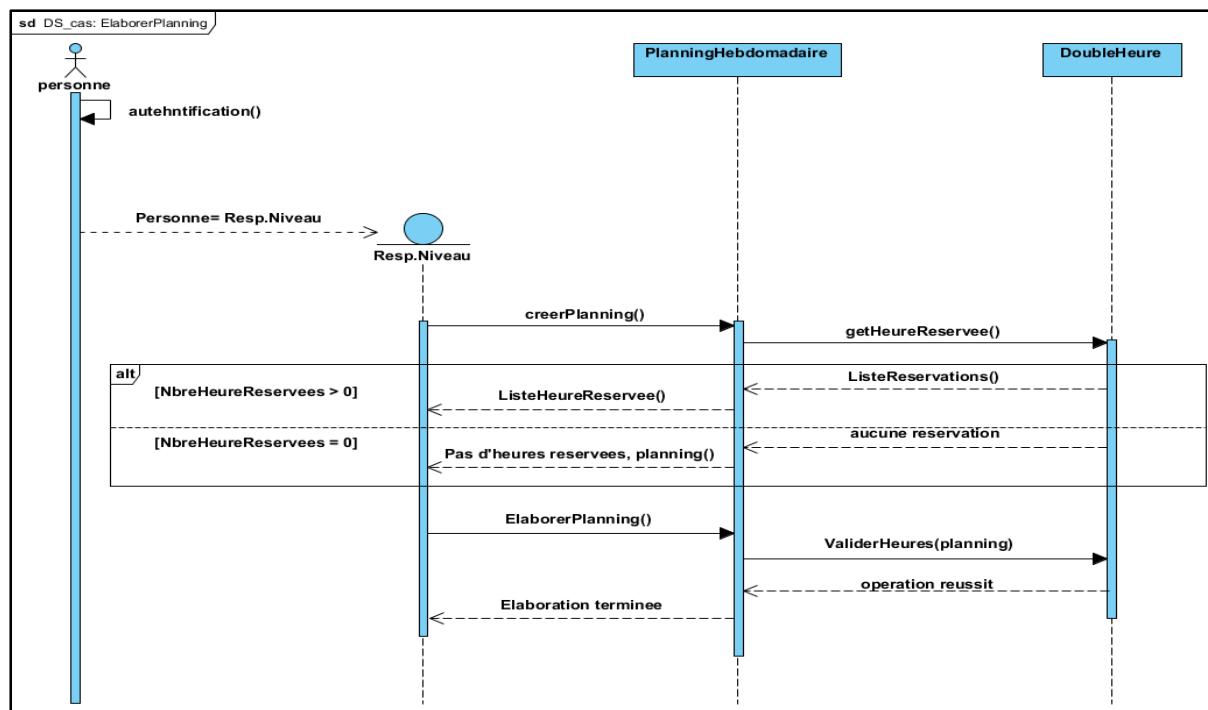


Figure 5: DS: créer un nouveau planning

#### d) Cas : faire des statistiques

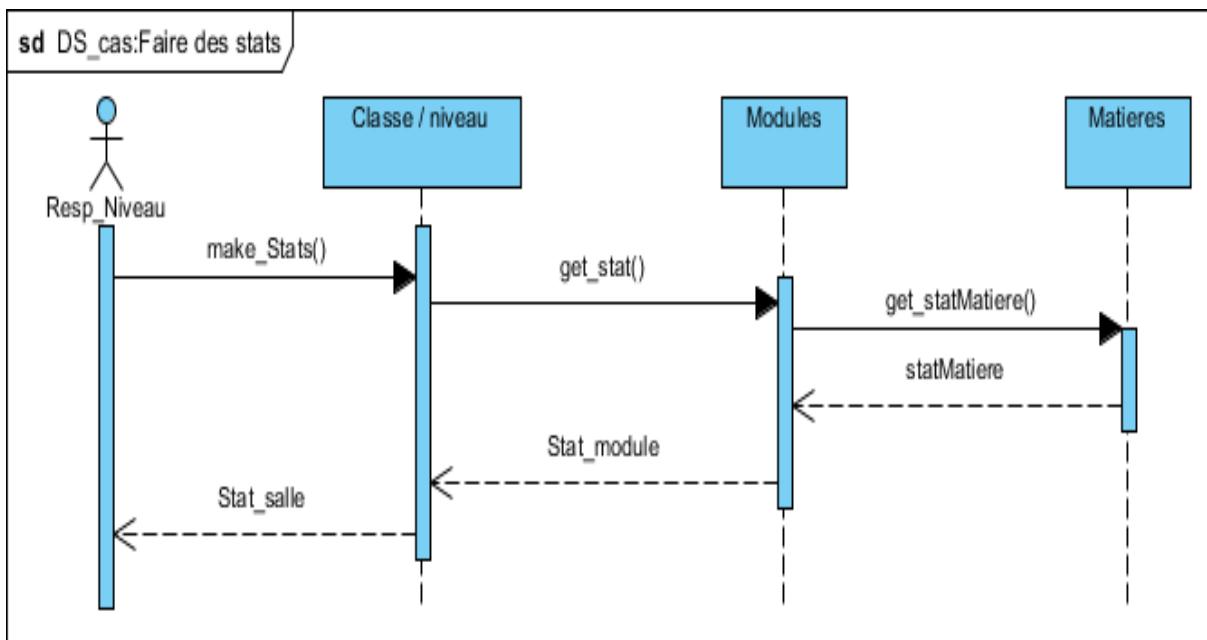


Figure 6: DS: faire des statistiques

## 4. Le diagramme d'activité

Une action est une composante élémentaire de l'activité. Une activité est un comportement décrit par un séquencement organisé d'unités dont les éléments simples sont les actions. Un diagramme d'activité est une formalisation graphique des actions qui sont réalisées dans un cas d'utilisation. Il met l'accent sur le flot de contrôle d'une activité à l'autre. Il est bien adapté pour la description des cas d'utilisation.

## III. Technique de conception du système

La réalisation de ce projet a été rendu possible grâce à plusieurs techniques et outils informatiques à savoir :

### a) Méthode et cycle de vie utilisées

Pour développer notre système, nous avons fait le choix d'utiliser la méthode de développement <sup>3</sup>RUP. C'est une méthode générique, itérative et incrémentale de prise en charge du cycle de vie d'un logiciel. Il est piloté par des risques et centré sur l'architecture (fonctionnelle, matérielle et logique). Par rapport aux autres méthodes, RUP présente plusieurs avantages :

---

<sup>3</sup> Rational Unified Process

	Description	Points forts	Points faibles
<b>Cascade</b>	<ul style="list-style-type: none"> <li>Propose de dérouler les phases du projet de manière séquentielle ;</li> <li>Cite pour des raisons historiques</li> </ul>	<ul style="list-style-type: none"> <li>Distingue clairement les phases du projet ;</li> </ul>	<ul style="list-style-type: none"> <li>Non itératif ;</li> <li>Ne propose pas de modèle de documents ;</li> </ul>
<b>RUP (Rational Unified Process)</b>	<ul style="list-style-type: none"> <li>Promu par rational ;</li> <li>RUP est à la fois une méthodologie et un outil prêt à l'emploi ;</li> <li>Cible des projets de plus de 10 personnes</li> </ul>	<ul style="list-style-type: none"> <li>Itératif ;</li> <li>Spécifie le dialogue entre les différents intervenants du projet : les livrables, les plannings, les prototypes, ...</li> <li>Propose des modèles de documents et des canevas pour des projets type ;</li> </ul>	<ul style="list-style-type: none"> <li>Couteux à personnaliser ;</li> <li>Axe processus, au détriment du développement : peu de place pour le code et la technologie ;</li> </ul>
<b>XP : eXtreme programming</b>	<ul style="list-style-type: none"> <li>Ensemble de « best practices » de développement ;</li> <li>Cible des projets de moins de 10 personnes</li> </ul>	<ul style="list-style-type: none"> <li>Itératif ;</li> <li>Simple à mettre en œuvre ;</li> <li>Fait une large place aux aspects techniques ;</li> <li>Innovant</li> </ul>	<ul style="list-style-type: none"> <li>Ne couvre pas les phases en amont et en aval du développement ;</li> <li>Assez flou dans sa mise en œuvre</li> </ul>

Figure 7: Avantages de la méthode RUP

Le cycle de vie d'un logiciel, désigne toutes les étapes du développement d'un logiciel, de sa conception à sa disparition. Il permet de détecter les erreurs au plus tôt et ainsi de maîtriser la qualité du logiciel, les délais de sa réalisation et les coûts associés. Il existe plusieurs modèles de cycles de vie. Notamment : Le cycle de vie en V, le cycle en cascade, le cycle en W, le cycle en spirale, ... Pour notre projet, nous avons choisi d'adopter le cycle de vie en spirale. C'est une

succession de cycles comprenant 6 phases : analyse du risque (1), développement d'un prototype (2), tests du prototype (3), détermination des besoins (4), validation des besoins (5), planification du prochain cycle (6).

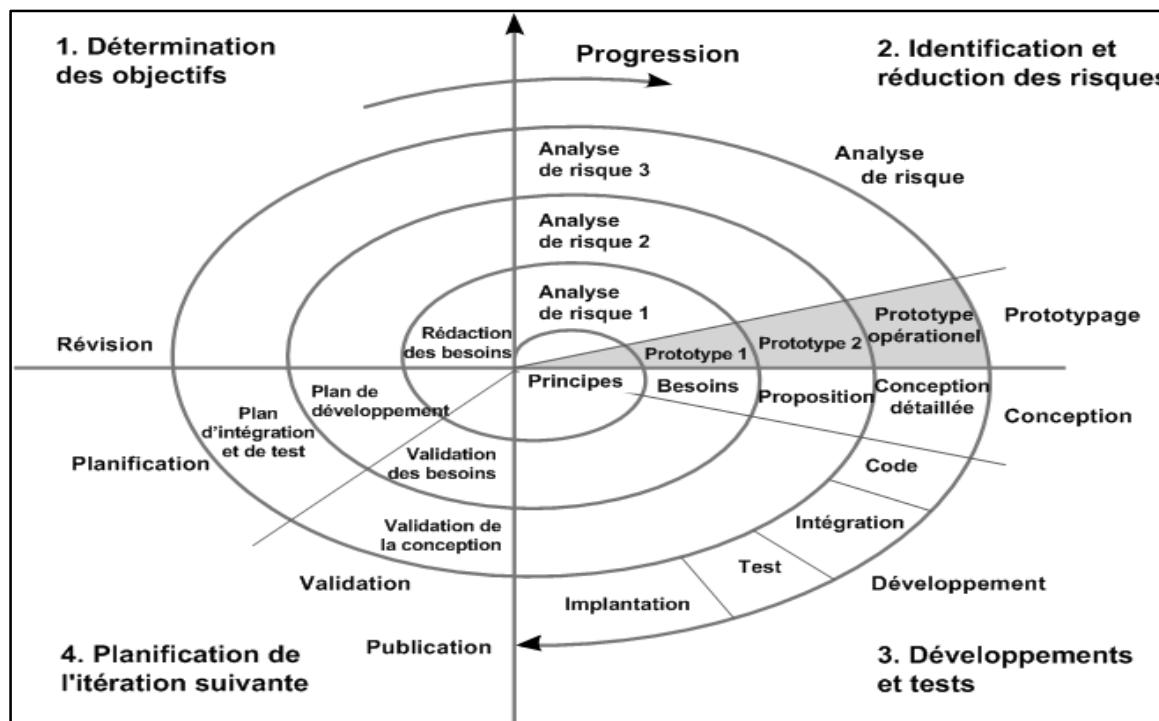


Figure 8: Cycle de vie en spirale

### b) Systèmes d'exploitation utilisés

Plusieurs systèmes d'exploitations ont été utilisés dans la réalisation de ce projet. Nous avons notamment :



Windows 10 est la version du système d'exploitation WINDOWS que nous avons utilisé, notamment pour la conception et la rédaction des différents comptes rendus de l'état d'avancement de notre projet



Pour des raisons de compatibilité (et surtout de rapidité), nous avons utilisé la distribution KALI de la famille LINUX pour l'implémentation de notre projet

### c) Logiciels et langages de programmation utilisés



Pour la modélisation de notre projet, nous avons essentiellement utilisé le logiciel **Visual Paradigm** ;



Le langage de programmation utilisé pour le l'implémentation de notre projet est **PYTHON** ;



Django est le Framework python que nous avons utilisés pour créer notre application ;



Le design de notre travail a été fait avec la version 3 du langage de style nommé **CSS**



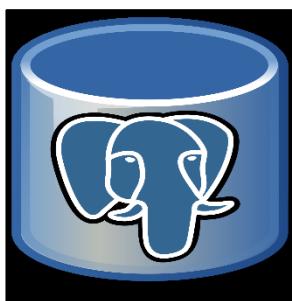
Les formulaires d'insertion de données ont été réalisés avec la version 5 du langage de balisage appelé HTML



La version mobile de notre projet a été développée avec l'émulateur ANDROID STUDIO



La partie serveur de projet a été programmée avec l'IDE Eclipse



Le système de gestion de bases de données utilisé pour ce projet est PostgreSQL

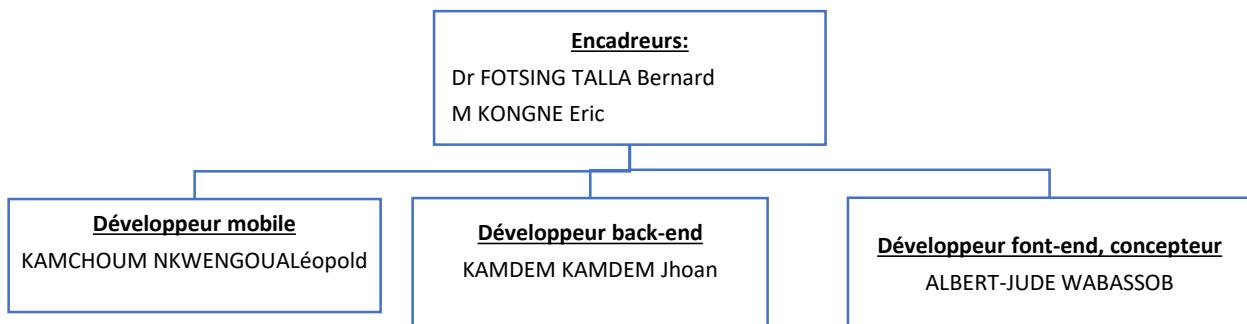


Nous avons utilisé Apache Tomcat comme serveur local lors du développement de ce projet

## IV. Présentation de l'équipe et planification

### 1. Présentation de l'équipe

L'équipe responsable de la réalisation de ce projet est organisée comme suit :



### 2. Planifications

La planification est un processus qui consiste à fixer (pour un groupe ou une entité), après étude et réflexion, les objectifs à atteindre et les étapes de réalisation de ceux-ci. Dans un projet informatique, la planification du projet est une étape qui consiste à déterminer la liste des tâches à réaliser, fixer les délais de livraison et assigner chacune d'elles à un responsable. Pour notre projet, la planification des tâches est comme suit :

Tableau 1: Planification des tâches

Project/étapes	Date de début	Date de fin	Durée en jours	Progression	Responsable
<b>Phase d'initialisation</b>	<b>1-avri</b>	<b>5-avri.</b>	<b>4</b>	<b>100%</b>	
cahier de charge	1-avri	3-avri.	3	100%	Albert
planification	4-avri	5-avri	1	100%	Jhoan
<b>Phase de lancement</b>	<b>5-avri</b>	<b>20-avri</b>	<b>15</b>	<b>83.33%</b>	
spécification du projet	5-avri	8-avril	2	100%	Léopold
maquettes	8-avri	15-avri	7	50%	Jhoan
Debut de la rédaction des contenus	15-avri	20-avri	5	100%	Albert

<b>Phase de conception</b>	<b>20-avri</b>	<b>01 juin</b>	<b>41</b>	<b>94.66%</b>	
Diagramme de cas d'utilisations	20-avri	26-avri	6	100%	Jhoan
diagramme de classe	26-avri	1-mai	5	100%	Jhoan
Diagramme de séquence	1-mai	5-mai	4	100%	Albert
Diagramme d'activité	5-mai	10-mai	5	60%	Jhoan
Diagramme de déploiement	13-mai	17-mai	5	100%	Albert
Rédaction des contenus : phase 2	17 mai	01 juin			
<b>Phase de production</b>	<b>01 juin</b>	<b>16-Juillet</b>	<b>46</b>	<b>89%</b>	
Développement back office	1-juin	14 juin	14	90%	Jhoan
Développement web font office	14 juin	04 Juillet	20	90%	Albert
Rédaction des contenus : phase 3	01 Juillet	12 Juillet	12	98%	Albert
Développement mobile font office	04 Juillet	13 Juillet	09	80%	Léopold
intégration	13 Juillet	16 Juillet	3	88%	Jhoan
<b>Phase d'exploitation</b>	<b>17 Juillet</b>	<b>20 juillet</b>	<b>3</b>	<b>0%</b>	
mise en ligne	17- Juillet	20 Juillet	3	0%	Jhoan

## Conclusion

Dans ce chapitre, il était question pour nous d'effectuer une conception détaillée du système. Il nous a donc permis de nous faire comprendre par les autres spécialistes du domaine. Il était garni de diagrammes explicatifs. Chacun de ces diagrammes était fait dans le souci de mieux matérialiser ou exprimer ce qui, jusque-là paraissait très abstrait. Pour la suite, le chapitre suivant portera sur l'implémentation de la solution proprement dite et sonnera un aperçu des résultats obtenus à l'issu de cela.

# CHAPITRE III: Mise en œuvre : implémentation et résultats obtenus

## INTRODUCTION

Ce chapitre est destiné à présenter les techniques de développement adoptées et les résultats concrets obtenus à la fin du projet. Il vient donc permettre même aux non informaticiens de comprendre réellement ce que nous nous efforçons à expliquer et faire comprendre à nos lecteurs et clients depuis le début de ce document.

### I. Technique de développement

La raison d'être de cette partie est de pouvoir donner un détail explicatif sur les différentes techniques et approches utilisées au cours du développement de notre projet.

Pour développer le côté web de notre système, nous avons particulièrement utilisé la technique de répartition des fichiers en trois principaux pôles. Notamment, l'architecture **MVT**.

Le **MVT** signifie littéralement **Modèle – Vues – Template**. C'est une technique de développement en python, intégré dans le Framework Django.

- **Le Modèle :** C'est un fichier d'extension « **.py** », assurant l'interaction avec la base de données. Sa mission est de chercher dans une base de donnée les items / occurrences correspondant à une requête <sup>4</sup>**SQL** donnée et de renvoyer une réponse facilement exploitable par le reste du programme. Les modèles s'appuient sur un traducteur

---

<sup>4</sup> SQL : Structured Query Language;

automatique appelle : **ORM (Object Relational Mapping)**. Un **ORM** traduit les résultats d'une requête SQL en objets Python avec lesquels on peut facilement interagir. De même, il permet d'écrire une requête SQL en Python.

- **La Vue :** Elle joue un rôle central dans le projet. Sa responsabilité est de recevoir une requête HTTP et d'y répondre de manière intelligible via un navigateur. Elle réalise également toutes les actions nécessaires pour répondre à la requête :
  - Si une interaction avec la base de données est requise, la vue appelle un modèle et récupère les objets renvoyés par ce dernier ;
  - Si un gabarit (Template) est nécessaire, la vue l'appelle.
- **Le Template :** C'est un fichier HTML qui peut recevoir des objets Python et qui est lié à une Vue. Dans un projet utilisant l'architecture MVT, les Template sont placé dans le dossier « Template » du projet.

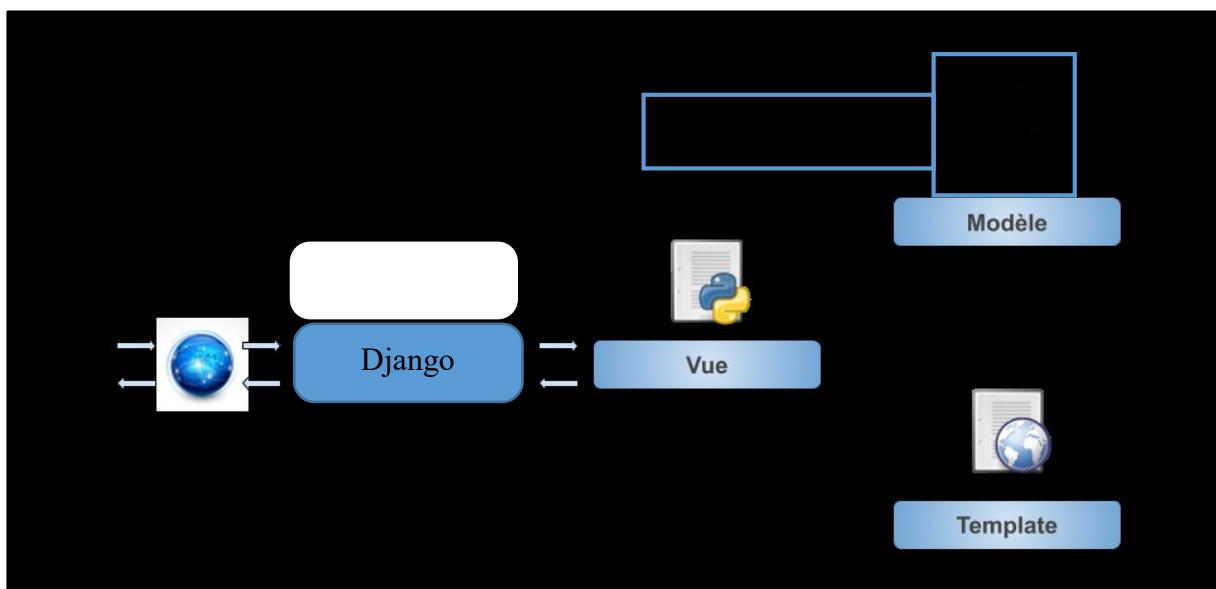


Figure 9: Architecture MVT

Pour de cote mobile, nous avons utilisés l'architecture Client-serveur. Il définit un mode de communication entre plusieurs ordinateurs d'un réseau. Le client (le plus souvent un navigateur) émet une requête vers le serveur grâce à son adresse IP et le port, qui désigne un service particulier du serveur. Le serveur reçoit la demande et répond à l'aide de l'adresse de la machine cliente et son port.

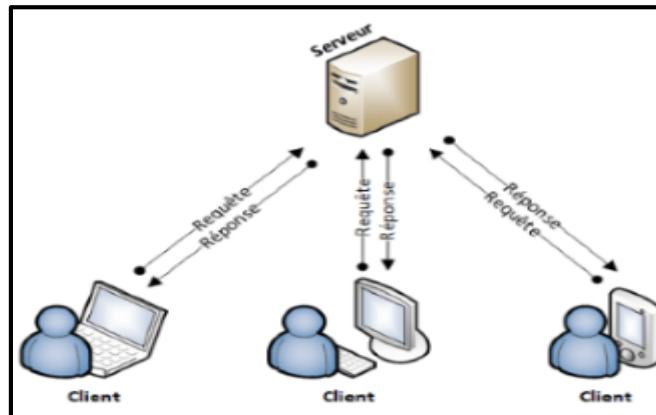
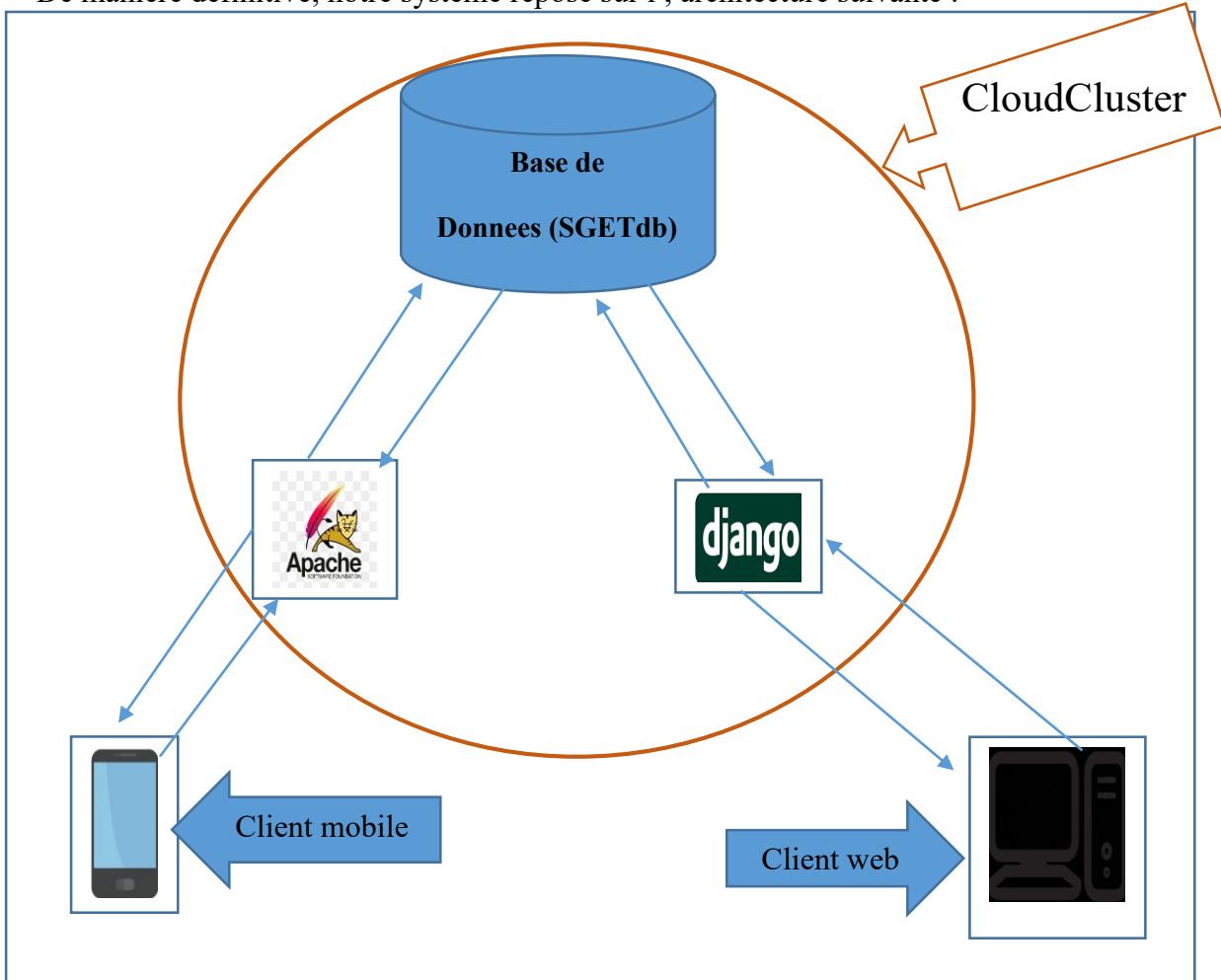


Figure 10: Architecture client-serveur

## II. Architecture et déploiement des applications

### 1. Architecture générale du système

De manière définitive, notre système repose sur 1 ; architecture suivante :



## 2. Déploiement des applications

De manière générale, le déploiement est le fait d'introduire une nouvelle solution technique et des services dans une organisation. Un déploiement réussi repose sur l'installation des ressources adéquates, une surveillance et une évaluation / contrôle ou surveillance continue de ces ressources.

Pour déployer et jouir des fonctionnalités de nos applications, plusieurs ressources sont requises :

- Pour déployer la version web, il faudra :
  - Disposer d'un <sup>5</sup>PC portable ou fixe avec au minimum : **4Go de RAM**, un processeur de **2.5GHz** de fréquence ;
  - Sur cette machine, télécharger et installer au moins la version **7.0 de python** sur son site officiel : [www.python.com](http://www.python.com) ;
  - Télécharger et installer la version 13.0 (de préférence) du <sup>6</sup>SGBD **PostgreSQL** ;
  - Télécharger et installer la version au moins la version **3.2.3** du Framework **Django** ;
  - Télécharger et installer les packages suivants sur leurs sites officiels :

Tableau 2: [Packages requis pour le développement de l'application](#)

<u>Packages</u>	<u>Versions (minimale)</u>
<b>Asgiref</b>	3.3.4
<b>Django</b>	3.2
<b>django-debug-toolbar</b>	3.2.1
<b>pip</b>	21.0.1
<b>psycopg2 binary</b>	2.8.6
<b>pytz</b>	2021.1
<b>setuptools</b>	56.0.0
<b>django-filter</b>	2.4.0

<sup>5</sup> PC: Personal Computer;

<sup>6</sup> SGBD: Système de Gestion de Bases de Données ;

<b>djangorestframework</b>	3.12.4
<b>importlib-metadata</b>	4.0.1
<b>Markdown</b>	3.3.4
<b>Sqlparse</b>	0.4.1
<b>Wheel</b>	0.36.2
<b>typing-extensions</b>	3.7.4.3
<b>zipp</b>	3.4.1
<b>Android Studio</b>	4.1.3
<b>eclipse</b>	4.19.0
<b>postgresql</b>	13.0
<b>websocket</b>	1.2.2
<b>picasso</b>	2.5.2
<b>Apache Tomcat</b>	9.0

➤

- Pour déployer la version mobile, il faudra :
  - Disposer d'un téléphone mobile sur lequel est installé au moins la **version 8.0** d'Android ;
  - Ce périphérique doit être capable de se connecter à Internet ;

### III. Résultats concrets du développement

A l'issue de notre travail, nous avons obtenus plusieurs résultats, couronnant ainsi l'ensemble du laborieux travail abattu tout au long de l'année académique.

#### 1. Sur le côté web

- Tout commence par le paramétrage de l'application. Ici nous avons dans la figure ci-dessous la page d'administration du site. Avant toute utilisation du système l'administrateur devra créer dans l'ordre les groupes d'Utilisateurs, les Professeurs, les Responsables de Niveau, les Chefs de Départements, les Départements, les Filières, les Options (Classe), les Etudiants, les Ressources Critiques et enfin l'Année Académique.

- Ensuite le responsable de niveau crée tous les plannings du semestre et va les modifier au fur et à mesure de l'évolution du temps. Lorsque les plannings sont créer ils sont dans l'état ‘**non valider**’ ensuite ils seront dans l'état ‘**valider**’ et enfin ‘**publier**’.

GESTION PLANNING

Dashboard

Profile

Emplois De Temps

IUT-FV

Validation

Publication

Confirmation

Statistiques

Contactez Nous

Initialisation

Search...

SEA IUT-FV NOTIFICATIONS DPT

## Creation d'un nouveau semestre

Numero :

Lmd :

Date debut :

Date fin :

Année :

- Emploi de temps vide, généré à la création du semestre

GESTION PLANNING

E-mail: [udrectorat@univ-dschang.org](mailto:udrectorat@univ-dschang.org)

E-mail: [iuty-bandjoun@univ-dschang.org](mailto:iuty-bandjoun@univ-dschang.org)

Dashboard

Profile

Emplois De Temps

IUT-FV

Validation

Publication

Confirmation

Statistiques

Contactez Nous

Initialisation

emploi de temps du 28 juin 2021 au 3 juillet 2021

Annee Academique : 2020/2021 Classe : genie logiciel Semestre : S4/lmd6 Niveau : 2 Enseignant Responsable: noulamo thierry Semaine No : 15

	7H30 - 9H30	9H30 - 11H30	11H30 - 12H00	12H00 - 14H00	14H00 - 16H00
<b>Lundi</b>	---	---	Pause	---	---
<b>Mardi</b>	---	---	Pause	---	---
<b>Mercredi</b>	---	---	Pause	---	---
<b>Jeudi</b>	---	---	Pause	---	---
<b>Vendredi</b>	---	---	Pause	---	---
<b>Samedi</b>	---	---	Pause	---	---

planning précédent      planning de la semaine en cours      planning suivant

MATERIAL DASHBOARD DJANGO      © Creative Tim - coded by AppSeed

Figure 11: Emploi de temps vide

- Pour l'établissement du planning, le système donne la possibilité à un enseignant de rechercher les salles dans lesquelles il dispense ses cours. En cliquant juste sur Emploi De Temps.



Figure 12: Recherche de la classe "GL"

- Après la recherche, l'enseignant est capable de voir le planning hebdomadaire de la salle  
Mais aussi de notifier ses disponibilités de la semaine

The screenshot shows the 'Emplois De Temps' section of the SGET application. The sidebar includes: GESTION PLANNING, Dashboard, Profile, Emplois De Temps, IUT-FV, Validation, Publication, Confirmation, Statistiques, Contactez Nous, and Initialisation. The main content displays a weekly schedule from June 21 to June 26, 2021. The schedule grid shows time slots from 7H30 to 16H00 for each day of the week. Some cells in the grid contain text like 'base de donnee : Dr. thierry noulamo', 'Cryptographie', or 'Analyse fonctionnelle'. The header of the schedule indicates it is for the academic year 2020/2021, class 'genie logiciel', semester 'S4/lmd6', level '2', teacher responsible 'noulamo thierry', and week 'Semaine No : 14'. The university logo of the University of Dschang is visible in the top right corner.

Figure 13: Interface de réservation des plages horaires

- Apres soumission du planning, un message est affiché à l'enseignant, lui notifiant que ses réservations ont bien été prises en compte et qu'elles sont en attente de validation ;

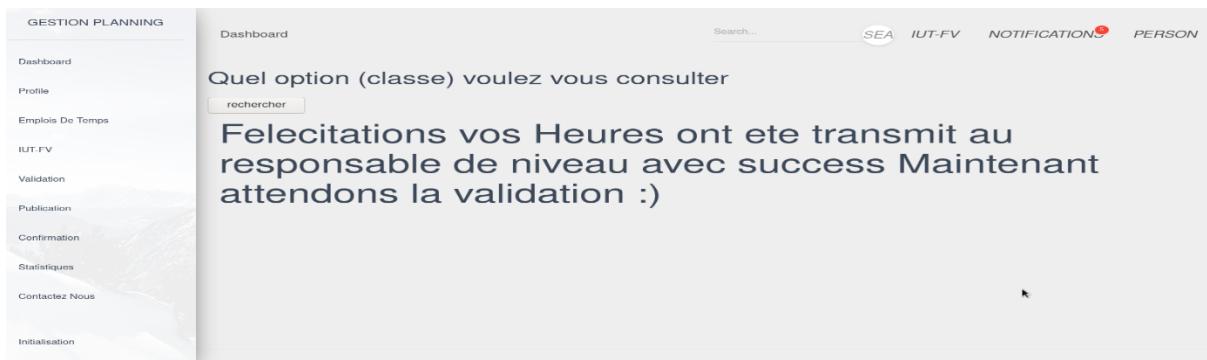


Figure 14: Message affiché après réservation des plages horaires

- Une fois l'heure sollicitée, les autres professeurs peuvent voir l'heure qui a déjà été sollicité et soit sollicitée la même heure soit choisir une autre plage. Aussi on peut naviguer entre les emplois de temps de la semaine en cour, passée et suivante.

The screenshot shows a weekly time-off schedule from July 5 to July 10, 2021. The grid has days of the week (Lundi, Mardi, Mercredi, Jeudi, Vendredi, Samedi) on the Y-axis and time slots (7H30 - 9H30, 9H30 - 11H30, 11H30 - 12H00, 12H00 - 14H00, 14H00 - 16H00) on the X-axis. Cells contain buttons like "...deja solliciter .." or "...choisir la matière...". A cursor is hovering over the Saturday slot. The sidebar includes links for "GESTION PLANNING", "Dashboard", "Profile", etc. The footer contains copyright information: "KAMDEM KAMDEM JHOAN | ALBERT-JUDE WABASSOB | KAMCHOUM NKWENGOUA LÉOPOLD" and "© Gestion Planning - coded by Groupe 16".

Figure 15: Message affiché après réservation des plages horaires

L'un des principaux acteurs de notre système est l'étudiant. Pour ce fait, nous avons prévus une interface qui lui permettra de visualiser plus aisément un emploi de temps de son choix.

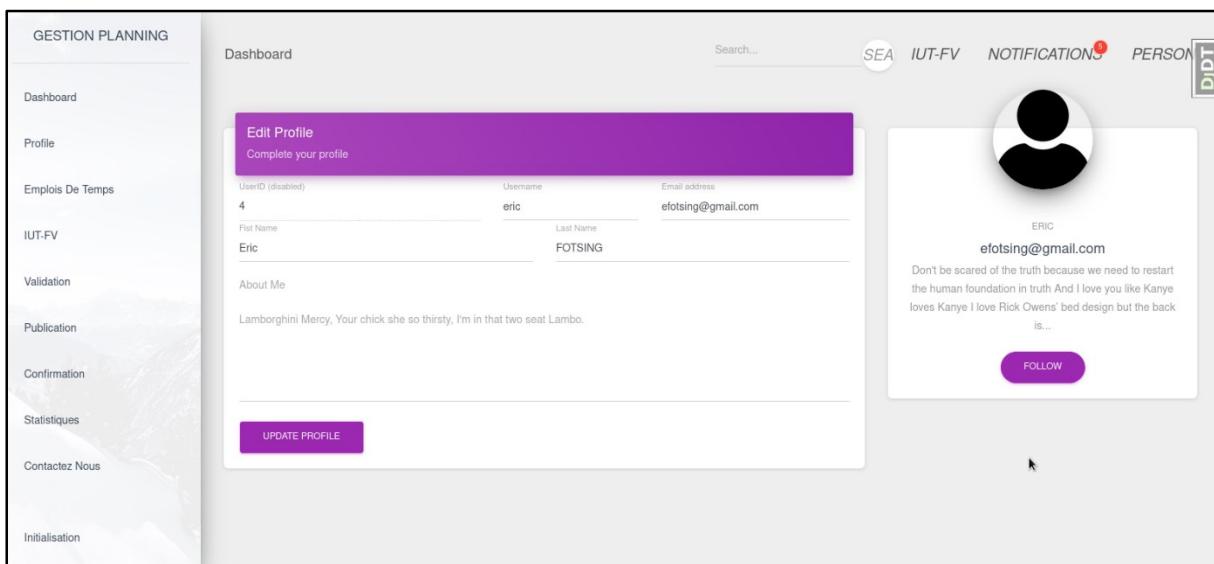


Figure 16: Interface de consultation d'emploi de temps de l'étudiant

## 2. Sur le côté mobile

Pour utiliser les fonctionnalités qu'offre notre application mobile, il faut posséder l'APK sur son téléphone Android. Il est disponible sur sigesonline (site web de l'IUT-FV).

Comment l'utiliser ?

- Il faut s'authentifier

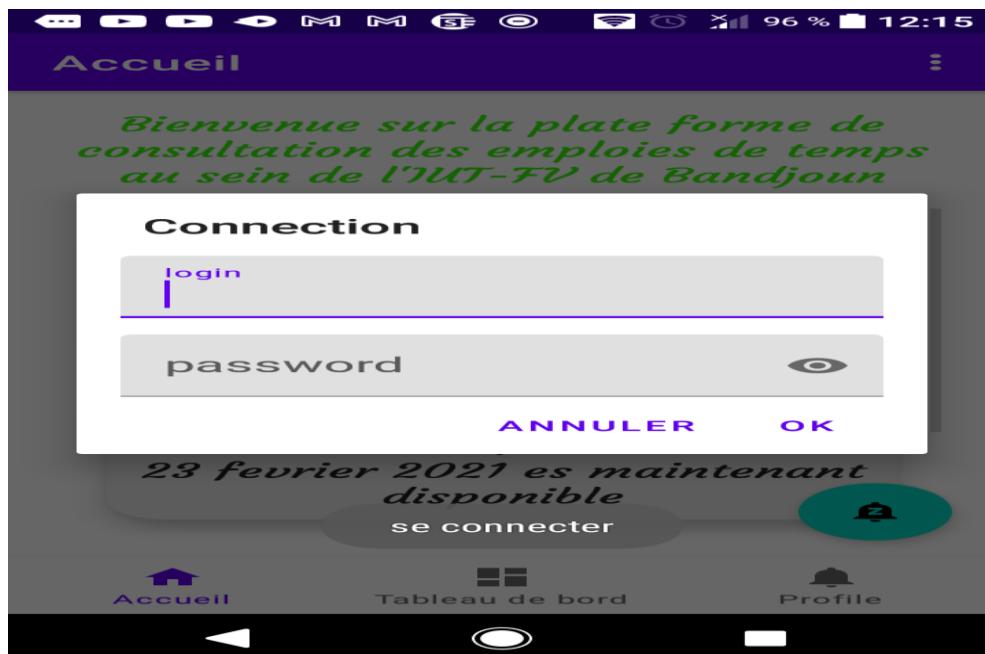


Figure 17: Authentification

Les paramètres d'authentification sont ceux fournis par sigesonline après l'admission au concours.

- **Accueil :** Apres authentification, l'utilisateur a accès à la page d'accueil et des notifications



Figure 18: Interface d'accueil

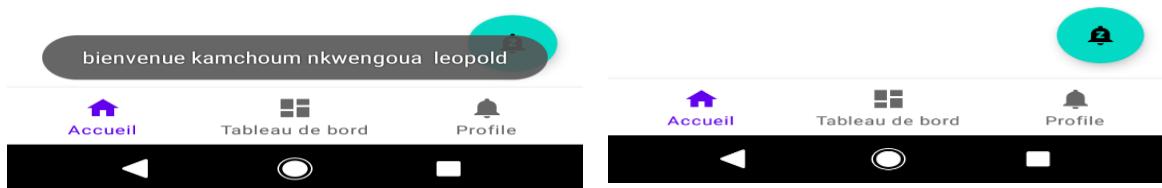


Figure 19: Interface d'accueil avec une notification

- **Le Dashboard :** Une fois que l'utilisateur arrive à l'interface d'accueil, il a la possibilité de se rendre sur le Dashboard. C'est l'interface principale de l'application. Il affiche par défaut le programme de la journée en cours.



Figure 20: Dashboard: programme de la journée en cours

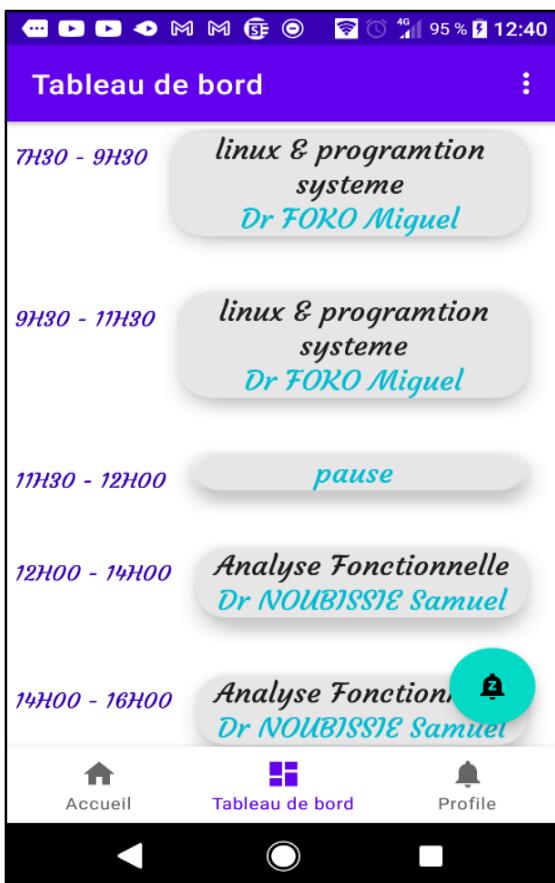


Figure 21: Dashboard (suite)

## Paramétrages sur le Dashboard

L'utilisateur à la possibilité de choisir une journée et une semaine de son choix et le programme lui affichera le programme de la journée de cette semaine s'il est déjà disponible.



Figure 22: Choix de la journée



Figure 23: choix de la semaine

- Le profil :** L'utilisateur a la possibilité d'avoir des informations sur son profil

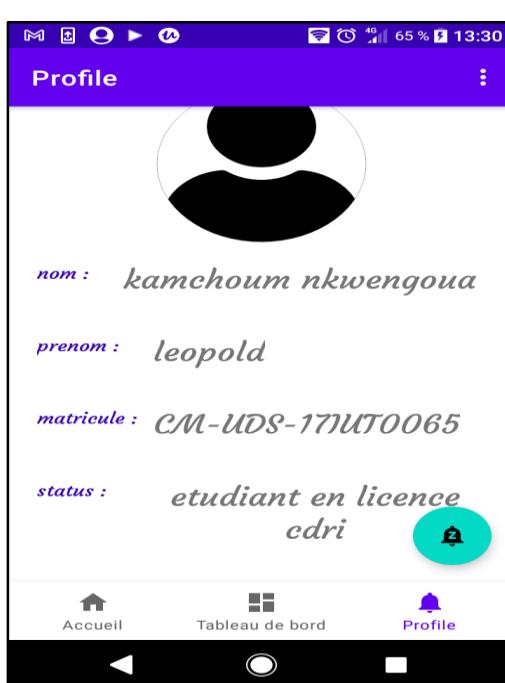


Figure 24: Profile

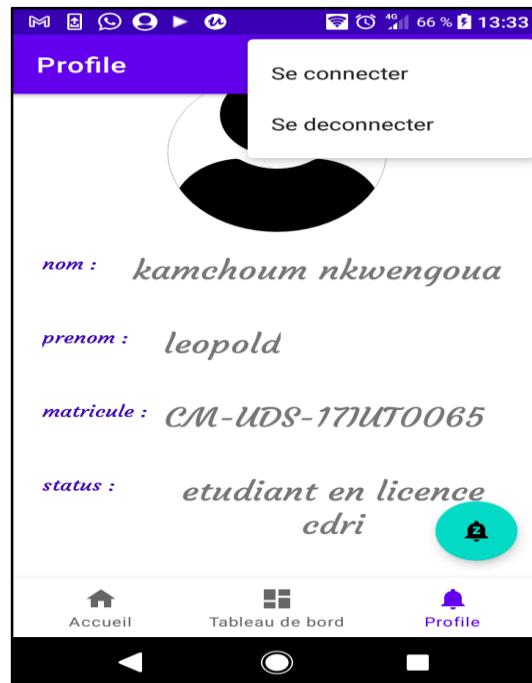
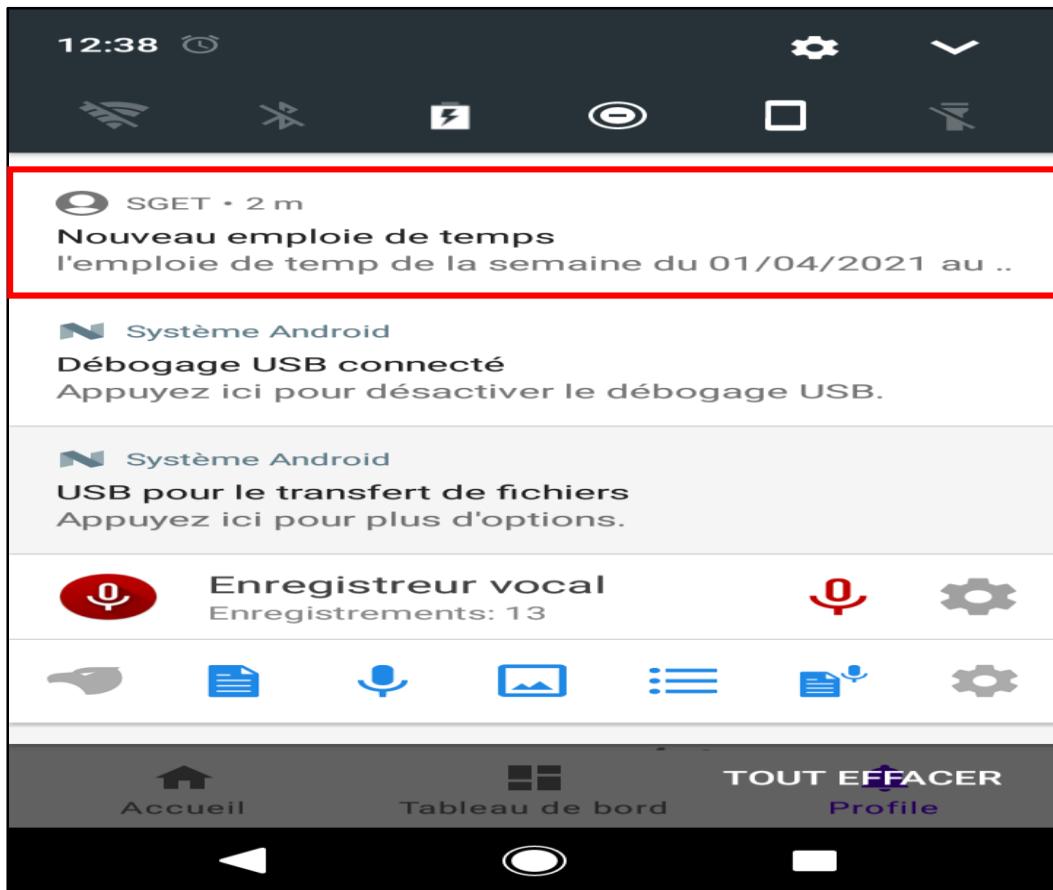


Figure 25: Option du profil

L'application est munie d'un système de notifications. En cas de publication d'un nouveau emploi de temps, une notification est envoyée à tous les utilisateurs connectés.



## IV. Hébergement du programme serveur de l'application

L'hébergement est le fait de mettre sur des serveurs sécurisés distant un ensemble de programmes offrant des services particuliers auxquels tout individu nécessiteux pourra facilement avoir accès. Pour notre application, nous avons hébergé plusieurs programmes à savoir : la base de données, le programme serveur web et le programme serveur mobile. Tous ces programmes ont été hébergés sur le serveur distant **CLOUDCLUSTER** (payant).

## V. Procédures d'hébergement

### 1. Hébergement de la base de données

Les étapes d'hébergement de la base de données (BD) que nous avons suivis sont les suivantes :

- Création d'un compte sur la plateforme CloudCluster : pour cela, il faut une adresse e-mail valide. Pour notre cas nous avons l'adresse suivante : kamchoumleopold@gmail.com ;
- Choisir les services que l'on souhaite consommer : Pour ce projet, nous avons choisi les services PostgreSQL.

The screenshot shows the 'mes applications' section of the CloudClusters interface. It lists two services:

- JavaClusters-35100 (v11)**: Actif, créé le 17/06/2021, rester 7 jour(s), cycle de facturation Mensuel, configuration (Express) 2 cœurs / 2 Go de RAM / 60,00 Go de SSD. Buttons: Faire en sorte, Renouveler.
- PostgreSQL-35095(v12.7)**: Actif, créé le 17/06/2021, rester 7 jour(s), cycle de facturation Mensuel, configuration (Express) 2 cœurs / 2 Go de RAM / 60,00 Go de SSD. Buttons: Faire en sorte, Renouveler.

Figure 26: Service POSTGRESQL

- Créer une base de données sur la plateforme : Pour notre cas, nous avons créée une BD nommée **sgetdb**.

The screenshot shows the 'Créer une base de données' dialog box. The 'Nom' field is highlighted with a red box and contains the value 'sgetdb'. Other fields include 'Créer une base de données', 'Jeu de caractères' (utf8 selected), and 'Utilisateurs' (empty). Buttons: Fermer, Sauvegarder.

Figure 27: création de la base de données

- Importer la base de données créée en local dans celle qu'on vient de créer ;

## 2.Hébergement du programme serveur de l'application mobile

Pour héberger ce programme il faut :

- Déployer le serveur Apache Tomcat sur la plateforme ;

	Nom	Statut	Actions
	JavaClusters-35100 (v11)	actif	<a href="#">Faire en sorte</a> <a href="#">Renouveler</a>
	PostgreSQL-35095(v12.7)	actif	<a href="#">Faire en sorte</a> <a href="#">Renouveler</a>

Figure 28: déploiement d'Apache Tomcat

- Exporter le projet réalisé en local avec l'extension .war et le charger sur la plateforme ;

Nom	Taille	Dernière mise à jour
manager	—	4 months ago
host-manager	—	4 months ago
examples	—	4 months ago
docs	—	4 months ago
User.war	56.9 MB	9 hours ago
ServerWebsocketPfe.war	921.11 KB	a few seconds ago

Figure 29:uploading du programme serveur

## VI. Evaluation des coûts

L'évaluation des couts est la partie du projet qui consiste à confronter réellement les implications financières. Elle permet aux développeurs de voir si le projet mérite d'être lancé et aux recruteurs de savoir s'il est à mesure de se procurer une telle solution pour ses structures. Le cout de notre projet est élaboré suivant trois angles : temporel, matériel et humain.

- Sur le plan temporel, nous estimons la durée de développement de cette solution à **12 semaines** de travail, soit trois mois ;
- Sur le plan matériel, il faut trois laptots ayant au moins les caractéristiques suivantes : **8Go de RAM**, un processeur **core i5 8<sup>e</sup> génération**, un disque dur de **500Go**. La location des machines répondant à ces caractéristiques coute **500.000FCFA** pour les trois mois. Il faut également un routeur wifi qui servira de point d'accès Internet pour des éventuelles recherches. Le prix d'achat s'élève à **80.000FCFA**. Et pour les trois mois de travail, le cout de la connexion internet s'élève à **135.000FCFA**, soit un cout mensuel de **45.000FCFA**. A la fin de l'implémentation, nous assurons l'accessibilité. Pour cela, le projet est hébergé sur un serveur distant à raison de **100\$/an**, soit **50.000FCFA**.
- Sur le plan humain, les ressources requises sont : trois développeurs qualifiés, tous titulaire d'une Licence en Informatique. Le salaire d'un développeur titulaire d'une licence est estimé à **1500FCFA/ heure**. Une journée de travail va de 08h00 à 18h00, soit **10h** de travail par jour pour un développeur et **60h** par semaine. Ainsi le salaire hebdomadaire d'un développeur s'élève à : **1.500x60 = 90.000FCFA**. Soit un salaire mensuel de **360.000FCFA**.

*Tableau 3: Estimation des couts*

Numéros	Désignations	Nombres	P.U	Total
1	Semaines	12		12
2	PC (location)	03(mois)		500.000
3	Connexion internet	03 (mois)	45000FCFA/mois	215.000
4	Hébergement	1an	200\$	100.000

<b>5</b>	Développeurs	03	360.000	1.080.000
<b>6</b>	Transport		50.000	60.000
<b>7</b>	Visual paradigm		50.000	150.000
<b>8</b>	Pycharm		50.000	200.000
<b>9</b>	<b>TOTAL</b>			<b>2.305.000FCFA</b>

## Conclusion

En guise de conclusion, nous pouvons dire qu'à présent, la solution salutaire que nous avions en vue de proposer depuis le début de ce document est à maintenant comprise par tous. Chacun a pu voir un aperçu de la concrétisation de cette solution. Nous en sommes fières. Mais il est à noter que nous ne sommes pas parvenus à ce niveau tout d'un coup, sans obstacles. Nous avons préféré vous préserver de tous ces détails tant vous n'aviez pas cernés la pertinence du projet. Maintenant, il est tant que vous le sachiez. Ce sera l'objet du chapitre qui suit.

# CHAPITRE IV:    **Limites du travail, propositions et suggestions**

## Introduction

Dans ce chapitre, nous allons nous appesantir sur les limites de notre solution ainsi que sur les différents freins et obstacles que nous avons rencontrés au cours de l'implémentation de cette solution ainsi que quelques perspectives de la solution.

### I. Apports, frein et limites du travail

#### 1. Apports du projet

Ce projet vient apporter un atout considérable pour nous et pour la communauté étudiante. Pour nous, la réalisation de ce projet nous a permis d'apprendre et approfondir nos connaissances du langage de programmation Python, le Framework Django. Il vient s'ajouter à notre palmarès de projets déjà réalisés et augmenter notre maîtrise du développement web et mobile. Pour la communauté étudiante, il permettra désormais aux parties prenantes de mieux effectuer leurs tâches.

#### 2. Freins et limites du travail

Sans doute, nous avons effectué un laborieux travail. Nous avons fourni une solution fiable aux problèmes de gestion des emplois de temps que rencontraient plusieurs établissements d'enseignement supérieur. Mais il est à noter que tout cela n'a pas été fait sans difficultés. Parmi les obstacles qui se placés sur notre chemin, nous pouvons notamment citer :

- La méconnaissance du langage de programmation : Notre projet nécessitait d'être codé en Python, et aucun de nous n'avait le niveau requiert dans ce langage de programmation. Raison pour laquelle nous avons pratiquement passés deux mois à apprendre et nous familiariser avec ce langage ;

- L'incompatibilité des <sup>7</sup>SGF de Windows avec ceux utilisés par Django nous ont contraint de migrer vers la distribution KALI du <sup>8</sup>SE Linux ;

A cote de ces obstacles, il y a des limites qui, jusque-là, nous n'avons pas encore pu franchir. Au nombres de celles-ci, nous pouvons citer :

- Notre solution n'est pas encore en mesure de prendre en compte les cours effectués en tronc commun ;
- Elle n'implémente pas encore la contrainte établie sur les ressources critiques comme les salles de classe ;

## II. Propositions et perspectives

En guise de propositions, nous dirons que, pour tous ceux qui voudrons se lancer et réaliser un pareil projet, il leur faudra s'assurer qu'ils ont une bonne de LINUX, qu'ils savent déjà bien programmer en python, ceci afin de gagner efficacement en temps. Pour que l'application soit plus rapide, il faut qu'elle soit exécutée sur des ordinateurs plus récents et ayant un processeur plus rapide. Pour la version mobile, il est plu apparier de l'exécuter sur un téléphone ayant au moins la version 10 d'Android.

En perspective, les versions suivantes de nos applications prendront en comptes toutes les limites suscitées. La gestion académique sera un nouveau module que nous augmenterons a nos applications.

## Conclusion

Dans ce chapitre, il était question pour nous de présenter le bilan, les apports, les limites et les suggestions. Afin de finaliser notre rapport, nous allons passer à la conclusion générale du rapport.

---

<sup>7</sup> SGF: Système de Gestion de Fichiers

<sup>8</sup> SE: Système exploitation

## CONCLUSION GENERALE

Dans une dimension d'appréciation finale, nous pouvons dire sans risque de se tromper que la réalisation d'un projet de fin d'étude permet effectivement de fixer les théories apprises des différentes matières au programme. La réalisation de bout en bout de ce projet nous a permis de témoigner cette assertion. Ainsi, notre travail s'est articulé autour de quatre chapitres. Le premier portait sur la **description du projet** et là nous avons fait une présentation de l'arrêté du projet, situé le problème et donnés l'état de l'art. Le deuxième chapitre portait sur **l'analyse fonctionnelle et technique de conception du système**. Là nous avons effectués une analyse fonctionnelle de notre projet. Celle-ci consistait en la réalisation de différents diagrammes conceptuels ; Ensuite, nous avons présentés les techniques de conception du système, c'est à dire la méthode, les systèmes d'exploitation et les logiciels utilisés. Le troisième portait sur la **mise en œuvre** de notre solution et là nous avons présenté la technique de développement utilisées pour développer notre système; les résultats obtenus après notre implémentation. Le dernier chapitre, quant à lui était axé sur **les limites du travail, les perspectives et les suggestions**. Il ressort de ce travail que la réalisation de ce système de gestion des emplois de temps nous a permis de monter en connaissance en développement web et mobile, de nous confronter aux difficultés réelles du monde des développeurs et de nous armer pour le monde du travail.

## BIBLIOGRAPHIE ET WEBOGRAPHIE

[1] Site de téléchargement de PostgreSQL : <https://www.postgresql.org/download/> (consultation le 23 mars 2021)

[2] Site de téléchargement de python: <https://www.python.org/downloads/mac-osx/> (consultation le 23 mars 2021)

[3] Site de téléchargement de Pycharm : <https://www.jetbrains.com/fr-fr/pycharm/download/download-thanks.html?platform=windows&code=PCC> (consultation le 23 mars 2021)

[4] Openclassrom : Découvres le Framework Django :  
<https://openclassrooms.com/fr/courses/4425076-decouvrez-le-framework-django/>  
(consultation le 29 mars 2021)

[5] Cours « SI-GL-RUP-FOTSING » du Dr FOTSING TALLA Bernard

[6] Cours « Cours\_GenieLog-UML-IHM\_Nkenlifack\_fev13\_IUTFV » du Pr NKELEFACK Julius Marcelin

[7] Cour « Cours-UML2\_OCL\_LaurentAUDIBERT » pris sur le site : <https://laurent-audibert.developpez.com/Cours-UML/> (consultation le 23 avril 2021)

[8] Site officiel de UnDeuxTemps : <https://www.axess.fr/conception-gestion-emploi-du-temps>  
(consultation le 23 février 2021)

[9] Site officiel de EDT : <https://www.index-education.com/fr/logiciel-emploi-du-temps.php>  
(consultation le 23 février 2021)

[10] Hébergement sur le Cloud : <https://clients.cloudclusters.io/> (consultation le 19 mai 2021)

[11] Programmer une notification avec Android Studio :  
<https://www.simplifiedcoding.net/andabarre de oid-push-notification-tutorial/> (consultation le 23 mars 2021)

[12] Programmer les CardView pour l'affichage des notifications :  
<https://www.journaldev.com/1https://www.journaldev.com/13358/android-google-maps->

[current-location-night-mode0024/android-recyclerview-android-cardview-example-tutorial](#)  
(consultation le 23 février 2021)

- [13] Programmer le design du bouton de notification :  
<https://guides.codepath.com/android/Design-Support-Library>  
<https://www.youtube.com/watch?v=SAAybxsckrg&feature=youtu.be>  
<https://www.learn2crack.com/category/androiddev> (consultation le 20 février 2006)
- [15] Communication client-serveur par le protocole WebSocket :  
<https://www.pubnub.com/blog/java-websocket-programming-with-android-and-spring-boot> (consultation le 17 janvier 2021)
- [16] Programmation de l'affichage des images de profils par défaut :  
<https://www.flaticon.com/> (consultation le 12 avril 2021)  
[https://www.youtube.com/results?search\\_query=picasso+android+studio+2020](https://www.youtube.com/results?search_query=picasso+android+studio+2020)  
(consultation le 20 février 2021)  
<https://www.youtube.com/c/CodinginFlow/playlists> (consultation le 10 février 2021)
- [17] Programmation de la connexion entre le programme serveur distant et la base de données : <https://www.codejava.net/> (consultation le 20 avril 2021)
- [18] Comment installer PostgreSql sur Debian : <https://computingforgeeks.com/how-to-install-pgadmin4-on-debian/> (consultation le 20 février 2021)
- [19] Afficher une image prise depuis un serveur distant sur le mobile d'un client :  
<https://square.github.io/picasso/> (consultation le 23 mars 2021)

# ANNEXE

The screenshot shows the GESTION PLANNING software interface. At the top, there are three boxes: one for the University of Dschang, one for the Institute of Technology, and one for the Department of Computer Engineering. Below these is a central header: "emplois de temps du 21 juin 2021 au 26 juin 2021". Underneath is a sub-header: "Annee Academique : 2020/2021 Classe : genie logiciel Semestre : S4/lmd6 Niveau : 2 Enseignant Responsable: noulamo thierry Semaine No : 14". The main content is a 6x5 grid representing the weekly schedule from Monday to Saturday, from 7H30 to 16H00. The grid contains course names like "base de donnee", "Cryptographie", and "Analyse fonctionnelle". Some cells are highlighted in purple.

	7H30 - 9H30	9H30 - 11H30	11H30 - 12H00	12H00 - 14H00	14H00 - 16H00
Lundi	base de donnee : Dr. thierry noulamo --choisir la matière--	base de donnee : Dr. thierry noulamo Cryptographie	Pause	base de donnee : Dr. thierry noulamo Cryptographie	base de donnee : Dr. thierry noulamo Cryptographie
Mardi	Cryptographie	Cryptographie	Pause	Cryptographie	Cryptographie
Mercredi	Analyse fonctionnelle	Cryptographie	Pause	Choisir la matière	Choisir la matière
Jeudi	--choisir la matière--	--choisir la matière--	Pause	--choisir la matière--	--choisir la matière--
Vendredi	--choisir la matière--	--choisir la matière--	Pause	--choisir la matière--	--choisir la matière--
Samedi	--choisir la matière--	--choisir la matière--	Pause	--choisir la matière--	--choisir la matière--

Figure 30: Choix de la matière a programmer

The screenshot shows the GESTION PLANNING software interface. At the top, there are three boxes: one for the University of Dschang, one for the Institute of Technology, and one for the Department of Computer Engineering. Below these is a central header: "emplois de temps du 21 juin 2021 au 26 juin 2021". Underneath is a sub-header: "Annee Academique : 2020/2021 Classe : genie logiciel Semestre : S4/lmd6 Niveau : 2 Enseignant Responsable: noulamo thierry Semaine No : 14". The main content is a 6x5 grid representing the weekly schedule from Monday to Saturday, from 7H30 to 16H00. The grid contains course names like "base de donnee", "Cryptographie", and "Analyse fonctionnelle". Some cells are highlighted in purple.

	7H30 - 9H30	9H30 - 11H30	11H30 - 12H00	12H00 - 14H00	14H00 - 16H00
Lundi	base de donnee : Dr. thierry noulamo	base de donnee : Dr. thierry noulamo	Pause	base de donnee : Dr. thierry noulamo	base de donnee : Dr. thierry noulamo
Mardi	--choisir la matière--	--choisir la matière--	Pause	--choisir la matière--	--choisir la matière--
Mercredi	--choisir la matière--	--choisir la matière--	Pause	--choisir la matière--	--choisir la matière--
Jeudi	--choisir la matière--	--choisir la matière--	Pause	--choisir la matière--	--choisir la matière--
Vendredi	--choisir la matière--	--choisir la matière--	Pause	--choisir la matière--	--choisir la matière--
Samedi	--choisir la matière--	--choisir la matière--	Pause	--choisir la matière--	--choisir la matière--

Figure 31: emploi de temps

The screenshot shows the GESTION PLANNING software interface. On the left is a sidebar with navigation links: GESTION PLANNING, Dashboard, Profile, Emplois De Temps, IUT-FV, Validation, Publication, Confirmation, Statistiques, Contactez Nous, and Initialisation. The main area has a header: "Bienvenue Prof. layou clementine chef de département de Genie Informatique". Below it is a title: "liste des planning en attente de publication". There is a "publier" button at the bottom of the list area.

Figure 32: emplois en attente de publication