RÉPUBLIQUE DU CAMEROUN
REPUBLIC OF CAMEROON
Peace – Work - Fatherland

UNIVERSITÉ DE DSCHANG
UNIVERSITY OF DSCHANG
Scholae Thesaurus Dschangensis Ibi Cordum

BP 96, Dschang (Cameroun) – Tél./Fax (237) 233 45 13 81
Website : http://www.univ-dschang.org.
E-mail : udsrectorat@univ-dschang.org

INSTITUT UNIVERSITAIRE DE TECHNOLOGIE
FOTSO VICTOR DE BANDJOUN
FOTSO VICTOR UNIVERSITY
INSTITUTE OF TECHNOLOGY

Département de Génie Informatique
Département of Computer Engineering

BP 134, Bandjoun – Tél (237) 699 31 61 30 / 670 64 23 92
Website : http://www.univ-dschang.org/iutfv/
E-mail : iutfv-bandjoun@univ-dschang.org

Département de Génie Informatique

PROJET DE FIN D'ÉTUDE

Mise en application des connaissances reçues durant les huit semestres (LMD1 - 6)

Thème : Contrôle d'accès aux activités de l'IUT-FV de Bandjoun

Effectué du 23 février au 09 juillet 2018
A l'IUT FV de l'UDS

Rédigé et présenté par :

MENKAM NGWENOM Francis Bonis **CM-UDS-14IUT0004**
MBOUKAM Alex **CM-UDS-15IUT1086**
MANFO TSIDA Alex Christian **CM-UDS-15IUT1135**
NGNEMCHIE Nathalie **CM-UDS-12IUT0774**

En vue de l'obtention de la ***Licence de technologie***

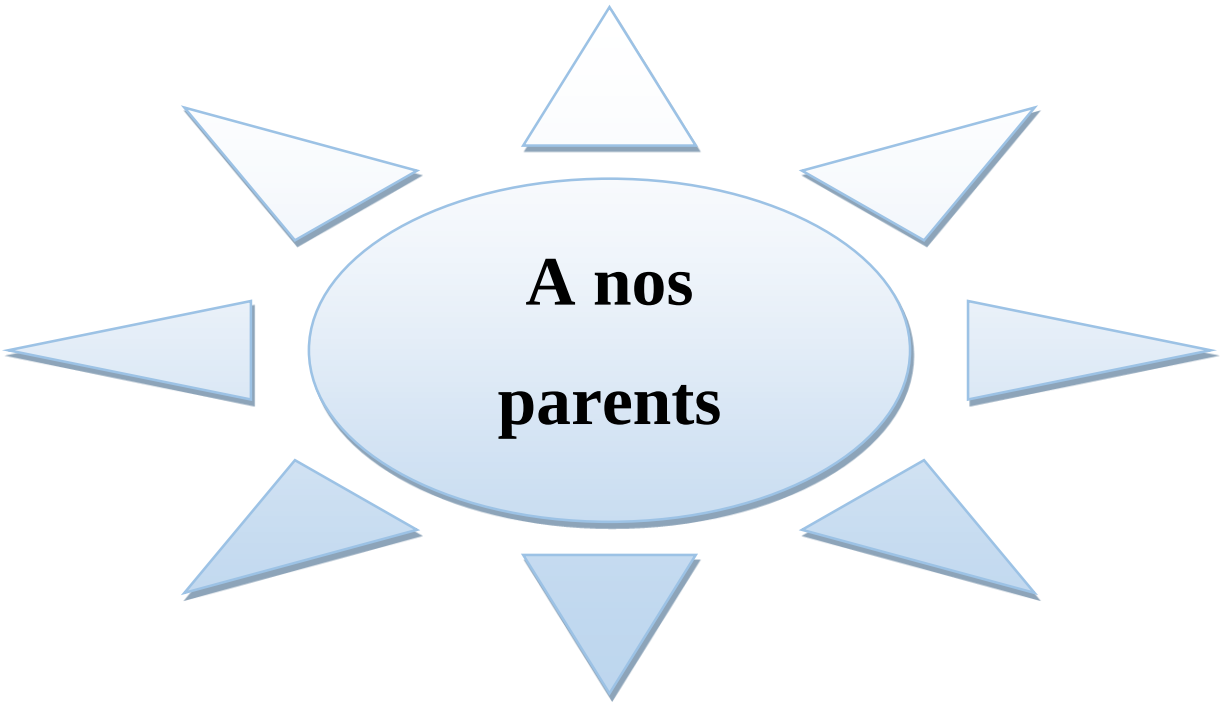
Mention : *Génie Informatique (GI)*
Parcours : *Informatique et Réseaux (IR)*
Option : *Concepteur Développeur Réseaux et Internet (CDRI)*

Sous la supervision de :

Pr NKENLIFACK Marcellin, Chef de Département Informatique de l'UDS
M. KUATE Victor, Enseignant & formateur CISCO à l'IUT-FV

Année académique 2017-2018

Dédicace



Remerciements

Ce travail n'aurait jamais pu être mené à bien sans le soutien infaillible du DIEU tout puissant, qui dans son Amour inconditionné nous a permis de rencontrer des personnes qui nous ont fait bénéficier de leur expérience et de nombreux conseils.

Nous remercions principalement :

- **S.E. Paul BIYA Barthélemy**, Président de la république du Cameroun, pour son soutient du don des ordinateur portable PBHev aux Etudiants pour qu'ils puissent faire des recherches pour leurs études ;
- **Pr. FOGUE Médard**, Directeur de l'Institut Universitaire de Technologie Fotso Victor (IUT-FV) de Bandjoun, pour nous avoir permis de suivre les cours et de faire ce projet dans sons Institut ;
- **Pr. NKENLIFACK Marcellin**, Chef de Département de Génie Informatique à l'IUTFV et aussi notre encadreur, pour sa disponibilité, ses conseils, remarques et motivations qu'il nous a donnés durant notre projet de fin d'étude ;
- **Dr. MPAME Guillène**, Responsable de niveau LIR, pour nous avoir soumis à toute sorte de projet et sous haute pression ce qui nous a permis de mieux évoluer dans ce projet ;
- **Dr. TALLA Bernard**, Enseignant à l'IUT-FV, à travers son cours d'UML, nous a fait ressortir tous les diagrammes UML relative à notre projet de fin d'étude ce qui nous a permis de bien modéliser notre projet de fin d'étude et de bien assimiler le langage UML ;
- Tous nos Enseignants du Département de Génie Informatique, pour leurs disponibilités durant notre cursus et pour les leçons apportées dans le cadre des cours dispensé ;
- Nos parents, qui ne cessent de ménager des efforts pour notre réussite à savoir : **M. et Mme MENKAM** ; *********
- Nos camarades qui nous ont aidés dans le processus de réalisation de ce projet **;**
- **********
- Tous ceux qui, de près ou de loin ont contribué à l'élaboration de ce travail et dont nous avons omis de citer les noms, qu'ils trouvent ici l'expression de notre profonde gratitude.

Liste des abréviations

Tableau 1 Liste des sigles et abréviations

Abréviations / Sigles	Définition
IUT-FV	Institut Universitaire de Technologie Fotso Victor
HTML	HyperText Markup Language
PHP	Personal Home Page
UML	Unified Modeling Language
MVC	Model View Controller
TIC	Technologie de l'Information et de la Communication
AJAX	Asynchronous JavaScript And XML
UDS	Université de Dschang
TIC	Technologie de l'information et de la Communication
SGBD	Système de Gestion des Bases de Données
2TUP	2 Track Unified Processus
OMG	Object Management Group

Résumé

Le présent travail décrit le système que nous avons mis en œuvre pour faciliter la planification, le suivi et le contrôle d'accès aux activités (tels que les examens, les travaux pratiques dans des laboratoires, et même les salles spécialisé comme le restaurant universitaire et la bibliothèque) au sein de l'UDS en générale et de l'IUT-FV en particulier. Ceci se fait en offrant au responsable (les ayants droits) une interface graphique (pages HTML) qui les permettrons de créer rapidement une activité avec tous les détails relative à cette activité par de simples clic, en plus ils pourront visualiser où même imprimer ces plannings, le système sera capable par simple lecture de la carte d'étudiant du candidat, d'autoriser ou d'interdire l'accès s'il ne remplit pas tous les conditions requises par l'activité. Cette approche avec l'intégration des TIC permettra à l'administration d'être rapide et pointu dans la gestion des activités et le management des Etudiants car la plus part des vérifications se ferons automatiquement.

Ce travail a été réalisée en suivant le processus 2TUP, la phase de conception a été modélisé par le langage UML, l'architecture utilisé est composée de trois (03) couches à savoir : la couche présentation constitué des HTML, AJAX, CSS ; la couche métier constitué des scripts PHP interprété par le serveur web Apache, pour le traitement des requêtes et en fin la couche accès aux données constitué du SGBD PostgreSQL, pour le stockage des données.

Cette application est élaborée actuellement dans sa version 1.0 et est destinée à évoluer ; de ce fait, des fonctionnalités y seront greffées au fur et à mesure.

Mots clés : Activité, contrôle d'accès.

Abstract

This work describes the system we have implemented to facilitate planning, monitoring and control of access to activities (such as exams, laboratory labs, and even specialized rooms such as the University Restaurant and the library) within the UDS in general and the IUT-FV in particular. This is done by offering the manager (the rights holders) a graphical interface (HTML pages) that will allow them to quickly create an activity with all the details related to this activity by simple click, in addition they will be able to visualize where to even print these schedules, the system will be able to simply read the student's student card, authorize or prohibit access if it does not meet all the conditions required by the activity. This approach with the integration of ICTs will allow the administration to be fast and sharp in the management of the activities and the management of the Students because the majority of the checks will be done automatically.

This work was carried out following the 2TUP process, the design phase was modeled by the UML language, the architecture used consists of three (03) layers namely: the presentation layer consisting of HTML, AJAX, CSS; the business layer consisting of PHP scripts interpreted by the Apache web server, for the processing of requests and at the end the data access layer consisting of the PostgreSQL DBMS, for data storage.

This application is currently developed in version 1.0 and is intended to evolve; as a result, features will be grafted as and when.

Keywords: Activity, access control.

Sommaire

DÉDICACE I

REMERCIEMENTS II

LISTE DES ABRÉVIATIONS III

RÉSUMÉ IV

ABSTRACT V

SOMMAIRE VI

LISTE DES TABLEAUX ET FIGURES VIII

INTRODUCTION GÉNÉRALE 1

CHAPITRE 1 PRÉSENTATION DU CADRE DE TRAVAIL ET CAHIER DE CHARGES 2

1.1 CADRE DE TRAVAIL 2
1.2 CAHIER DE CHARGES 4
1.3 PRÉSENTATION DE L'EXISTANT 5
1.4 CRITIQUES 5
1.5 PROPOSITION DE SOLUTION 6
1.6 DÉROULEMENT DU TRAVAIL ET OPÉRATIONS EFFECTUÉES 6
1.7 ETAT DE L'ART 7

CHAPITRE 2 ANALYSE, ALGORITHMES ET STRUCTURES DE DONNÉES 9

2.1 ANALYSE FONCTIONNELLE DU SYSTÈME 9
2.2 CONCEPTION TECHNIQUE, FONCTIONNELLE ET OBJET DU SYSTÈME 19

CHAPITRE 3 IMPLÉMENTATION, TECHNIQUES DE PROGRAMMATION ET RÉSULTATS 25

3.1 TECHNIQUE DE DÉVELOPPEMENT 25
3.2 DÉPLOIEMENT DE L'APPLICATION 26
3.3 RÉSULTATS CONCRETS DU DÉVELOPPEMENT 29

CHAPITRE 4 APPORT, LEÇONS, DIFFICULTÉS ET PERSPECTIVES 34

4.1 APPORTS 34
4.2 LEÇONS 34
4.3 DIFFICULTÉS 34
4.4 PERSPECTIVES ENVISAGÉES 35

CONCLUSION GÉNÉRALE 36

BIBLIOGRAPHIE **37**

ANNEXES ..... **A**

ANNEXE 1 : QUELQUES VUES DE L'UNIVERSITÉ DE DSCHANG ..... A

ANNEXE 2 : DÉPLOIEMENT DE L'APPLICATION SUR HEROKU ..... A

ANNEXE 3 : RÉSULTAT D'IMPLÉMENTATION ..... B

TABLE DES MATIÈRES ..... **A**

Liste des tableaux et figures

Tableaux :

Tableau 1 Liste des sigles et abréviations iii
Tableau 2 Liste des activités menées 6
Tableau 3 Comparaison de SGBD les plus utilisés 8
Tableau 4 Description textuelle du cas d'utilisation : « Participer à une activité » 12
Tableau 5 Description textuelle du cas d'utilisation : « ajouter activité » 13
Tableau 6 Description textuelle du cas d'utilisation : « modifier matière » 13
Tableau 7 Description textuelle du cas d'utilisation : « ajouter etudiant » 14
Tableau 8 Dictionnaire de données 20

Figures :

Figure 1 Une vue des Laboratoires de l'IUTFV 3
Figure 2 Diagramme de circulation des informations 5
Figure 3 Diagramme de cas d'utilisation de l'étudiant 10
Figure 4 Diagramme de cas d'utilisation de l'enseignant 11
Figure 5 Diagramme de cas d'utilisation de l'administrateur 12
Figure 6 Diagramme de séquence du cas d'utilisation : « crée activité » 15
Figure 7 Diagramme de séquence du cas d'utilisation du cas d'utilisation : « participer à une activité » 15
Figure 8 Diagramme d'activité du cas d'utilisation : « créer activité » 16
Figure 9 Diagramme d'activité du cas d'utilisation : « voire rapport d'activité » 17
Figure 10 Diagramme d'état-transition d'une activité 17
Figure 11 Modèle de navigation relatif à l'application web 18
Figure 12 Architecture fonctionnelle et technique du système 19
Figure 13 Diagramme de classe 23
Figure 14 Modèle physique de données 24
Figure 15 Diagramme de composants du pattern MVC 25
Figure 16 Arborescence du système 26
Figure 17 Informations système générales 27
Figure 18 Interface de Git Bash 27
Figure 19 Installation de Laravel 5.5 28
Figure 20 Migration de la base de données 28
Figure 21 Interface de gestion d'activite 29
Figure 22 Activité en cours (en temps réel) 30
Figure 23 Liste des étudiants en salle (en temps réel) 30
Figure 24 function qui affiche les activités en cours 31
Figure 25 Planning de la normale allant du 17 au 25 juin 2018 (simulation) 32
Figure 26 Requête de sélection des matières d'une activité 33
Figure 27 Rapport d'examen de normale de la période du 17 au 25 juin 2018 (simulation) 33
Figure 28 Vue panoramique de l'Uds et de l'IUT FV de Bandjoun [W-UDs-16] ..... A
Figure 29 Formulaire de connexion sur la plate forme ..... B
Figure 30 Interface de mise à jours de la bd par l'admin C
Figure 31 Formulaire de création d'une activité ..... D
Figure 32 Formulaire pour l'affichage du planning d'une activité ..... D
Figure 33 Obtenir la liste des étudiants ayant participé à une activité ..... E
Figure 34 Obtenir la liste des étudiants n'ayant pas participé à une activité ..... F

Introduction générale

Avec l'évolution de nouvelles technologies et l'avènement des TIC, les méthodes de production et de gestion actuelles sont de plus en plus complexes, car elles subissent d'importantes mutations technologiques et font en général appel à l'automatisation de façon à réduire au strict minimum l'effort physique et intellectuel de l'Homme. Le problème de contrôle d'accès s'est toujours posé dans les établissements à caractère éducatif en général et les instituts d'enseignement supérieur en particulier. C'est ainsi que dans l'optique de rendre les étudiants capables de produire des applications pouvant satisfaire les besoins des entreprises et particuliers, que chaque étudiants de licence à l'IUT-FV plus précisément dans le département GENIE INFORMATIQUE, arrivé en fin de formation, aura un projet de fin d'études qui lui permettra de mettre en pratique tous les savoirs acquis tout au long de sa formation. Ce travail d'informatisation est répartit en quatre chapitres réparti comme suit : Dans le premier, la description du projet est clairement élucidée, le second chapitre présente l'analyse fonctionnelle et technique de conception, le troisième quant à lui récapitule les techniques issues de la mise en production suivies des résultats pertinents obtenus, enfin le dernier chapitre présente les freins issus de la mise en œuvre et des perspectives d'amélioration du système.

Chapitre 1 Présentation du cadre de travail et Cahier de charges

1.1 Cadre de travail

1.1.1 Présentation de l'Université de Dschang (Uds)

L'Université de Dschang (UDS) qui fait partie des 08 universités d'Etat du Cameroun a été créée à la faveur de la réforme universitaire par le décret n° 93/026 du 19 janvier 1993. Elle compte à ce jour, sept établissements : la Faculté des Lettres et Sciences Humaines (FLSH), la Faculté des Sciences Economiques et de Gestion (FSEG), la Faculté des Sciences Juridiques et Politiques (FSJP), la Faculté des Sciences (FS), la Faculté d'Agronomie et des Sciences Agricoles (FASA), l'Institut Universitaire de Technologie Fotso Victor (IUTFV) à Bandjoun et l'Institut des Beaux-Arts à Foumban (IBAF). D'après [W-UDs-16], l'institution dispose de huit campus dans 6 des 10 régions du pays, à savoir, des antennes pédagogiques et de recherche à Bambui (Nord-Ouest), Belabo (Est), Ebolowa (Sud), Maroua (Extrême-Nord) et Yaoundé-Nkolbisson (Centre). A l'Ouest, on a le campus principal à Dschang et deux établissements à Bandjoun et à Foumban.

1.1.2 Présentation de l'IUT FV

L'Institut Universitaire de Technologie FOTSO Victor (IUTFV) de Bandjoun est l'un des sept établissements de l'Université de Dschang. L'IUTFV représente environ 3500 étudiants (en 2016), encadrés par une quarantaine d'enseignants permanents et une cinquantaine de personnels d'appui.

◆ **L'IUTFV forme des techniciens supérieurs dans divers Cursus :**

- **Diplôme Universitaire de Technologie (DUT)**, qui contient les parcours : Génie Informatique (GI), Génie Electrique (Electrotechnique et Electronique), Génie des Télécommunications et Réseaux, Maintenance Industrielle et Productique, Mécatronique Automobile, Génie Civil, Génie Thermique Energie et Environnement.
- **Brevet de Technicien Supérieur (BTS)**, qui contient les parcours : Comptabilité et Gestion des Entreprises, Electronique, Electrotechnique, Technique de Commercialisation, Secrétariat de Direction, Génie Civil.
- **Licence de Technologie (LT)**, qui contient les parcours : Informatique et Réseaux (IR), option « *Concepteur Développeur Réseaux Internet* », Génie Electrique, Ingénierie des Télécoms et Réseaux, Génie Civil, Maintenance Industrielle et Productique, Mécatronique Automobile, Gestion et Maintenance des Installation Energétiques, Géomatique.

- **Licence Professionnelle (LP)**, avec les parcours : Commerce et Marketing, (* *Banque Gestionnaire des Relations Clientèles, * Marketing Manager Opérationnel)*, Gestion Comptable et Financière, Gestion Administrative et Management des Organisations.

◆ **L'IUTFV dispose** d'un cadre de travail convivial et des facilités de recherche. La figure ci-dessous, tirée de [W-IUTFV-16], présente des échantillons de Laboratoires de l'IUTFV.

Figure 1 Une vue des Laboratoires de l'IUTFV

◆ **D'autres crénons de formation sont disponibles :**

- **La formation continue :** Programmes spécifiques de recyclage du personnel des entreprises.
- **La Formation Ouverte à Distance (FOAD)**, offerte depuis plusieurs années en LIR.
- **L'Académie Internet Cisco :** Compétences techniques et professionnelles supplémentaires en vue des certifications internationales : CCNA, Security, Internet of Evrythings, ITE, etc.

1.1.3 Présentation du Département de Génie Informatique et du Laboratoire

⇨ **Missions et Formations offertes**

- **Former des techniciens supérieurs** capables de concevoir, réaliser, commercialiser, installer, gérer et maintenir des applications et grands systèmes informatiques et réseaux *(Gestion, C/S, E-Commerce, Mobiles…)* ; Ils peuvent également réaliser/administrer des réseaux et services Internet Multidimensionnels (IMD / IOE).

Les parcours suivants sont offerts :

- DUT Génie Informatique (*4 semestres ou 2 ans), options : GL, ISR, IG*
- Lic. de Technologie Informatique et Réseaux : (2 *semestres / 1 an), option : CDRI.*
- **Equipements :** 2 Laboratoires de TP en Intranet et Serveurs de TP (+ outils de nouvelle génération) ; Des Laptots sont disponibles pour les travaux d'étudiants.

- **Recherche :** Le département anime l'équipe STIC du Laboratoire LAIA. Actuellement, plus de 50 articles sont publiés (à l'international) et plus de 10 livres sont déjà édités.

⇨ **Appui au développement :** Plusieurs projets sont réalisés pour le développement du pays.

- **Ressources humaines :** 6 Enseignants *permanents, 1 ATER et plusieurs Professionnels*.

1.2 Cahier de charges

Projet de fin d'étude

N°	Noms et Prénoms	Matricule	Cursus-Parcours-Option
1	***MENKAM NGWENOM Francis Bonis***	*CM-UDS-14IUT0004*	*Licence-IR-CDRI*
2	***MBOUKAM Alex***	*CM-UDS-15IUT1086*	*Licence-IR-CDRI*
3	***MANFO TSIDA Alex Christian***	*CM-UDS-15IUT1135*	*Licence-IR-CDRI*
4	***NGNEMCHIE Nathalie***	*CM-UDS-12IUT0774*	*Licence-IR-CDRI*

THEME: Contrôle d'accès aux infrastructures

Techniques et outils :

- Hébergeur — heroku.com
- Gestion de version — git hub
- Langage et Scripts — *HTML, Java, PHP, JavaScript, CSS*
- Serveur web — *Apache (MS4W)*
- SGBD client-serveur — *PostgreSQL 9.3*
- Langage de Modélisation Objet — *UML*
- Ingénierie système et réseaux — *TCP/IP*
- *Framework* — *Laravel 5.5*
- Système d'exploitation — *Windows 7 32bit SP1*
- *AGL* — *astah-professional*
- *IDE* — *Android Studio, PhpStorm 8.0.3*
- Sécurité données — *Bcrypt*

Contexte et présentation générale

De nos jours, avec révolution phénoménale des nouvelles technologies dans tous les domaines de la société, les Universités et Instituts ménage tous les efforts pour s'arrimer à la donne. Ainsi l'USD à après de multiple succès dans l'automatisation de la gestion de différentes procédures et ressources au sein de ses écoles, est actuellement face à un défi majeur qui est celle d'automatiser l'autorisation à l'accès à ses différentes infrastructures telles que les salles de composition, bibliothèques, bureaux, laboratoire, etc.

Pour ce qui concerne ce projet de fin d'étude, nous allons nous concentrer uniquement sur le contrôle d'accès aux salles de composition avec toutes les tâches qui seront effectué avant, pendant et après l'activité. L'intégration des TIC est primordiale pour satisfaire se besoin, ainsi que les technologies client-serveur et aspects répartis sans oublie le moyen de communication des différents module qui est l'intranet et en fin la sécurité qui est très importe. Toutes ces contrainte serons satisfaite à l'interface multiutilisateur de type Android sur tablette et Smartphone et Web sur Ordinateur.

Quelques résultats attendus

- ✓ Niveau d'accès sécurisé aux fonctionnalités de l'application ;
- ✓ Listes des étudiants autorisés à composer (par classe et matière) ;
- ✓ Listes des étudiants participent avant, pendant et après une activité ;
- ✓ Listes des étudiants autorisés à participer à l'examen par Unité d'Enseignement et par matière ;
- ✓ Interface web de planification des activités par le personnel ;
- ✓ Interface graphique d'authentification de l'étudiant ;

Fait
à Bandjoun, le 10 mars 2018

Le Directeur de Projet

1.3 Présentation de l'existant

La gestion des activités dans un établissement est une tâche très délicate et de ce fait, il sera évident de le faire avec beaucoup de précaution car la réputation de l'établissement est en jeu. Lors de nos études sur la gestion des activités dans des établissements, nous avions choisir de le faire à l'IUT-FV de Bandjoun et nous avons fait ces remarque : la planification des activités est informatiser et assurer par une application web tournant localement sous le nom de « siges ». Le contrôle d'accès en salle de composition se fait sur la base de la présentation de la CNI ou la carte d'étudiant ainsi que les reçus de paiement des droits universitaires ainsi que les frais médicaux et lorsque les épreuves sont distribuées, le surveillant fait passer une liste dans laquelle les étudiants inscriront leurs noms ainsi que leurs signatures et cette liste fera office de liste de présence. C'est avec liste qu'ils (administration) pourront les noms de ceux qui ont pris part à l'examen et aussi les absents. La figure ci-dessous présente la circulation des informations relatif à une activité de la création jusqu'à la fin de cette activité.

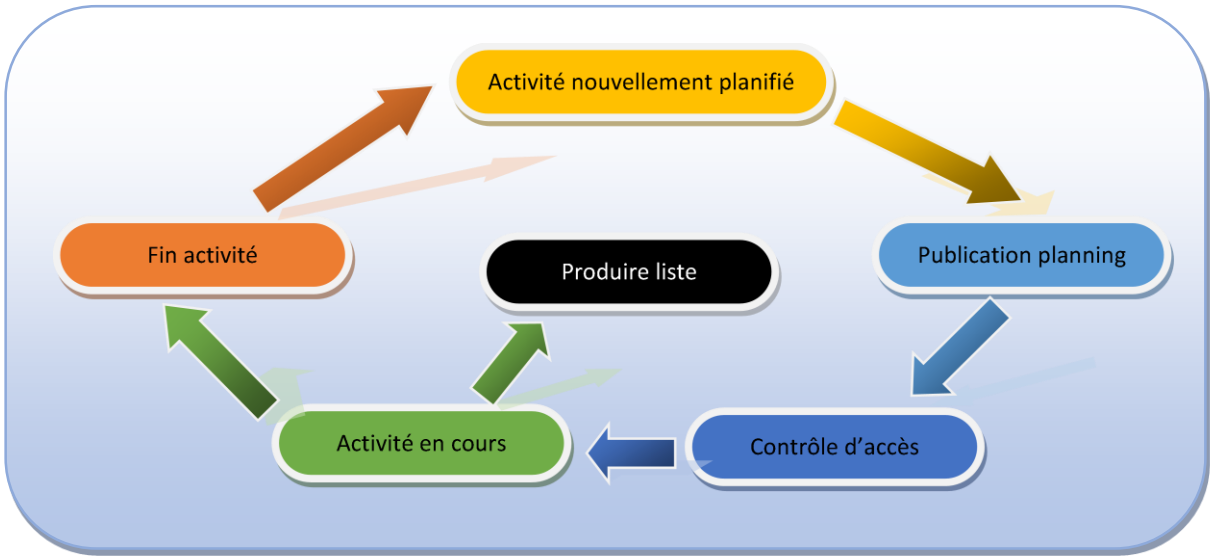


Figure 2 Diagramme de circulation des informations

1.4 Critiques

Après une étude critique sur la méthode actuelle de planification et de suivi des activités dans nos universités, nous constatons que cette méthode marche certes mais, limité par le temps car nécessité beaucoup de temps pour vérifier les informations avant de les publier et aussi, la traçabilité d'un planning par exemple mettra beaucoup de temps car les informations ne sont pas centralisé et

distribué. En ce qui concerne le suivi des activités (exemple d'un examen), l'accès dans une salle de composition se fait uniquement par simple vérification de la carte d'étudiant ou de la CNI si la photo sur la carte est identique au candidat, cet étudiant est alors autoriser de participer à l'activité ainsi toute personne pourra s'introduire dans n'importe quelle salle de composition, s'identifier et composer simplement. C'est ce qui permet à certain étudiants véreux de composer à la place des autres où même de composer une matière sans être autorisé à y participé car il est insolvable où n'est même pas inscrit à la cette matière. Voilà en fait les lacunes de l'ancienne méthode ainsi dégagé. Dans la suite, nous allons vous présentez notre approche pour remédier à ces problèmes.

1.5 Proposition de solution

L'apport de l'utilisation dans son ensemble des TCI dans la gestion des activités vient résoudre les problèmes tels la confidentialité, l'intégrité, la fiabilité, la sécurité et le gain de temps. Ainsi dans notre démarche, nous avions pris en compte tous ces qualités en proposant une application web qui servira pour la planification, le contrôle, le suivie et la génération des rapports à la fin de chaque activité et aussi une application mobile (Android) pour l'identification des étudiants afin de savoir qui est autorisé à prendre part à l'activité ou pas. De plus notre application dois permettre de savoir quels sont les étudiants qui sont en salle de composition et après l'activité l'obtention de la liste de présence et du rapport sera instantané et allègera la tâche au service de la scolarité ainsi qu'au surveillant.

1.6 Déroulement du travail et opérations effectuées

Pour réaliser ce projet, un certain nombre de tâches et d'activité ont été effectué et récapituler dans le tableau ci-dessous.

Tableau 2 Liste des activités menées

Phase	Début (date)	Fin (date)	Durée (en jours)
- Imprégnation des objectifs du projet	27-02-2018	27-02-2018	1
- Analyse préalable et définition des besoins	28-02-2018	04-03-2018	5
- Création du diagramme de classe et cas d'utilisation	05-03-2018	10-03-2018	5
- début de rédaction du rapport, Première de couverture, Remerciement, Résume, cahiers des charges	11-03-2018	14-03-2018	3
- implémentation	15-04-2018	20-05-018	25
- début de test en local et modification des erreurs	21-05-2018	28-05-2018	7
- déploiement de l'application web en ligne	29-05-2018	31-05-2018	3

- modification de l'application Android pour qu'il communique avec l'application web en ligne	01-06-2018	25-06-2018	25
- suite et fin de la rédaction du rapport	26-06-2018	08-07-2018	12
- dépôt du rapport	09-07-2018	09-07-2018	1
- Montage du PowerPoint	10-07-2018	15-07-2018	5
- Présentation du projet aux jurys	16-07-2018	16-07-2015	1

1.7 Etat de l'art

1.7.1 Choix des outils utilisés

1.7.1.1 Langage de modélisation

UML : Utilise principalement l'approche objet et facilite la modélisation des différents aspects du projet. C'est le standard de la modélisation objet adopté par l'OMG et universellement reconnu. Il est indépendant de tout langage de programmation, des processus de développement et permet la communication entre les acteurs à travers la "computation Independant Model" dans le souci de rendre le système évolutif, ce système doit être générique et évolué plus tard. Pour plus d'information sur UML bien vouloir consulter [W-UML-18] ou [PascRoq-UML-06].

1.7.1.2 Langages de programmation

- **HTML**, car il présente l'avantage d'être relativement simple à maîtriser et de produire des pages très légères et rapides à transmettre, même lorsque le débit est faible. Donc nous l'adopterons pour la structuration des informations de notre application web ;
- **Java**, pour le développement de notre interface mobile ;
- **PHP**, pour générer facilement les pages web contenant des informations provenant de la base de donnée et c'est aussi un excellent langage de script coté serveur qui sert d'intermédiaire entre le client et la base de données ;
- **JavaScript**, pour le contrôle d'information coté client et le réfléchissement ciblé des éléments sur la page ;
- **CSS**, pour le design de notre application web.

1.7.1.3 SGBD

L'exploitation des documents en ligne contenu dans [W-lessgbd-18] nous ont permis d'avoir le classement de quelques SGBD parmi lesquelles nous avions filtré ceux fréquemment jusqu'ici dans le cadre de notre formation académique et aussi les info contenu dans [Wikipedia-Kiwix] nous ont permis d'avoir plus de détail sur chacun d'eux. Ainsi on à ressortir le tableau ci-dessous.

Tableau 3 Comparaison de SGBD les plus utilisés

Rang	Nom	Développeur	Environnements	Licence	Site web
1	Oracle	Oracle Corporation	Multiplateforme	Propriétaire	www.oracle.com/
2	MySQL	MySQL AB / Oracle	Multiplateforme	GNU (propriétaire)	www.mysql.com/
4	PostgreSQL	Micheal Stonebraker	Multiplateforme	BSD	www.postgresql.org/
7	Access	Microsoft	Windows, Mac OS X	Propriétaire, EULA	office.microsoft.com/

Ainsi le SGBD que nous optons d'utiliser est PostgreSQL car c'est un outil libre disponible selon les termes d'une licence de type BSD contrairement aux trois autres qui sont propriétaire. Documentation contenue dans [Doc-PostgresSql-9.5.3]

1.7.1.4 Framework

Comme Framework (ou cadriciel en bon français), on en distingue au moins deux types à savoir les Framework **front-end** (Bootstrap de Twitter et Foundation de Zurb) et **back-end (**Symphony, Laravel et CakePhp). En revanche, le Framework que nous avons utilisé est Laravel 5.5 car propose offre une facilité d'intégration de nouvelle module et vient aussi avec des module tels que le module d'authentification, le query builder écrire des requêtes de plus Laravel utilise HTML, PHP, JavaScript et Booststrap pour le design des interfaces avec Laravel, le déploiement de l'application est très facile (Inspiration : [W-FontEnd-18]).

1.7.1.5 Dessin des diagrammes

Pour les diagrammes, les outils disponibles sont, argoUml, powAMc, astash, winDesign la liste est très longue mais pour les avoirs (ceux qui respectent les normes) bien vouloir visiter le site www.omg.com la particularité des outils est qu'il respecte tous les standards d'UML et offres les outils pour représenter chaque diagramme d'UML. Ainsi l'AGL que nous avons utilisé lors de la modélisation est astash car il est très stable et possède des modules permettant de documenté notre modélisation sous en format html et aussi les diagrammes sont très bien représentés.

Chapitre 2 Analyse, Algorithmes et Structures de données

Après avoir réunies les informations sur les différents objectifs et résultat attendus. Nous allons d'abord procéder à une délimitation du projet, qui consistera à cadrer les modules à développer par la suite. Ensuite, nous mettrons en exergue les exigences fonctionnelles par des diagrammes de cas d'utilisation tout en élucidant les scénarii de certains de ces cas d'utilisations par des diagrammes de séquences et d'activité. Enfin, nous allons représenter, décrire le système dans son aspect architectural et terminer par la représentation des classes d'objets du système ainsi que les liens existent entre eux.

2.1 Analyse fonctionnelle du système

2.1.1 Définition modulaire du projet

Notre projet est subdivisé en plusieurs modules détaillés comme suit :

- **Planification des activités :** Ce module est destiné uniquement aux personnels, en les offrants des informations réel et actualisé sous forme de formulaires pour la création des activités ;
- **Impression des plannings des activités :** ce module permet à l'administration d'imprimer le programme de passage de chaque activité dans des meilleurs délais.
- **Suivie d'activités :** Destiné aux enseignants et aussi aux étudiants, et permet d'avoir les rapports pour chaque activité en cours et aussi le rapport finale d'une activité terminé ;
- **Contrôle d'accès aux activités :** Destiné principalement aux étudiants pour les permettes de s'identifier avec leurs carte d'étudiants ;
- **Mise à jour des informations de base :** seul l'administrateur peut utiliser ce module qui qui permet de gérer (créer, modifier, supprimer) les utilisateurs, les matières, bref tous ceux dans les autres utilisateurs aurons besoin.

2.1.2 Interaction entre le système et les acteurs

2.1.2.1 Les principaux acteurs du système

Les acteurs qui doivent interagir avec le système via la réalisation d'un ou de plusieurs cas d'utilisation sont les suivants :

- **Etudiant** : C'est celui pour qui les activités sont créées et il doit s'identifier pour y participer ;

- **Enseignant** : c'est celui qui planifie les activités ;
- **Administrateur** : Celui-ci a le contrôle du système entier.

2.1.2.2 Cas d'utilisation pour l'étudiant

L'étudiant aura la possibilité de participer à une activité planifier et pour se faire, il droit d'abord s'identifier et qui parle d'identification parle de soit t'utilisation d'un lecteur d'empreinte digital à l'aide de la tablette, d'un lecteur de code bar à l'aide d'un smartphone ou bien d'un lecteur de carte magnétique à l'aide de la tablette. Ensuite il peut aussi voire son rapport d'activité après une activité (en ligne). La figure ci-dessous illustre ces cas d'utilisation ainsi que les sous cas d'utilisations.

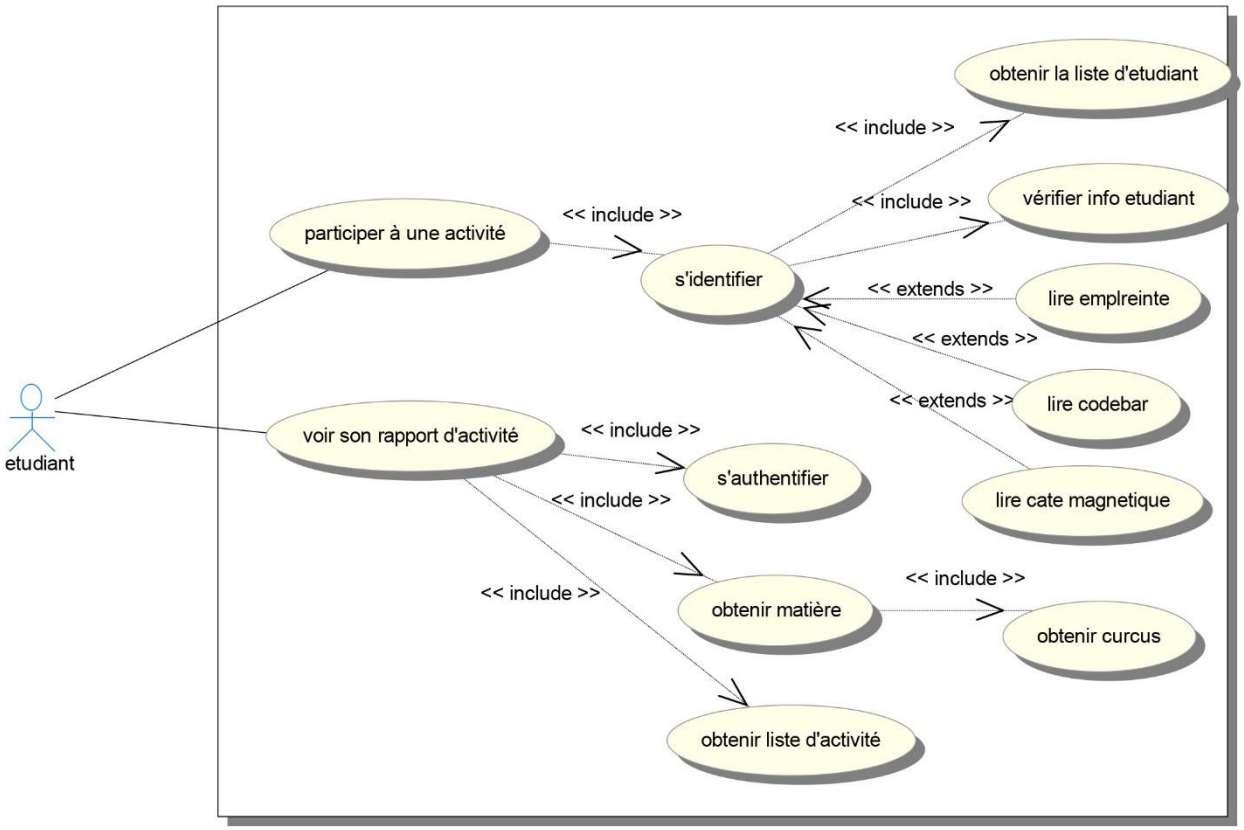


Figure 3 Diagramme de cas d'utilisation de l'étudiant

2.1.2.3 Cas d'utilisation pour l'enseignant (personnel)

L'enseignant peut afficher les activités déjà planifié et aussi peut modifier ou bien supprimer si et seulement si l'activité n'a pas encore eu lieux. De plus il peut aussi ajouter une activité et en fins peut voire les rapports des activités pendant l'activité et aussi après l'activité. La figure qui suit illustre clairement ces cas d'utilisation.

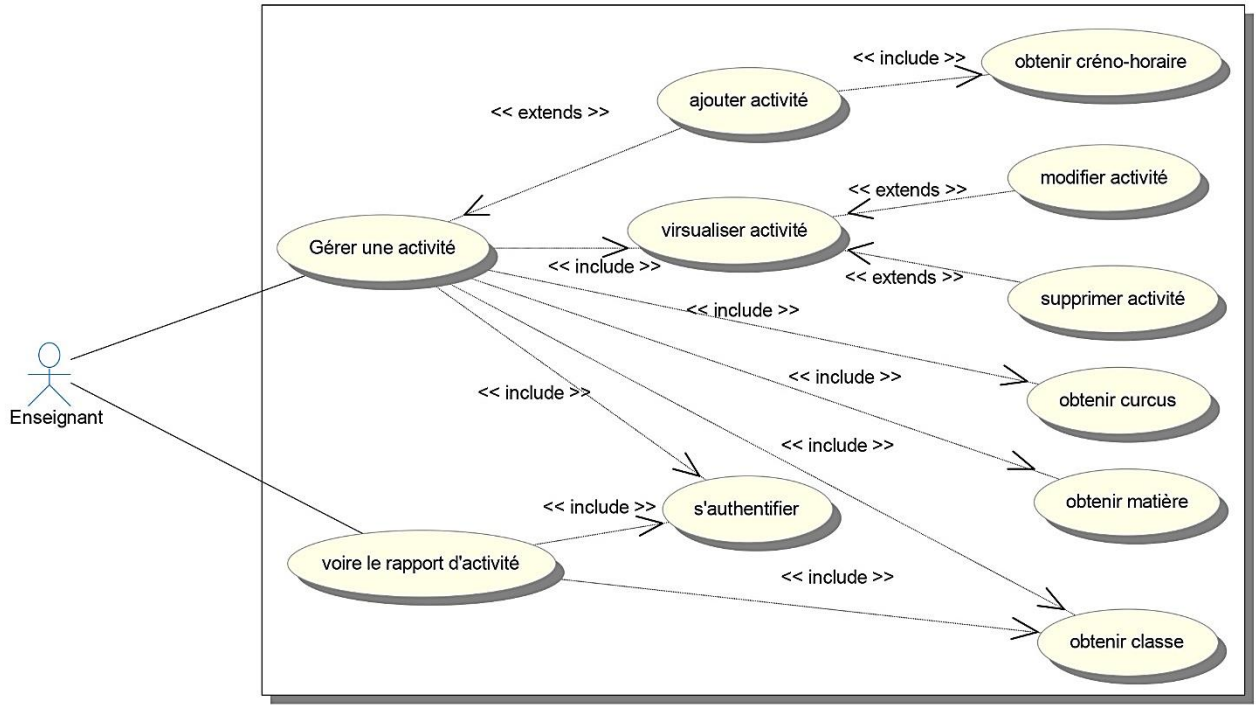


Figure 4 Diagramme de cas d'utilisation de l'enseignant

2.1.2.4 Cas d'utilisation pour l'administrateur

L'administrateur est celui qui a la lourde charge d'enregistrer les informations de base auxquelles les autres auront besoin dans le système comme par exemple l'enroulement des étudiants et enseignants, la mise à jour et l'ajout des matières, la gestion des horaires comme l'ajout de nouvelle ligne d'année académique et aussi des créneaux-horaire. Nous allons illustrer cette situations par la figure qui suivante :

Figure 5 Diagramme de cas d'utilisation de l'administrateur

2.1.2.5 Descriptions textuelle des cas d'utilisation

Les diagrammes précédents ont donnés une représentation d'un niveau d'abstraction élevé de cas d'utilisation. Ainsi, nous allons procéder aux descriptions textuelles des quelques-uns de ces cas d'utilisation.

- Participer à une activité

Tableau 4 Description textuelle du cas d'utilisation : « Participer à une activité »

Cas d'utilisation	Participer à une activité
Acteur primaire	Etudiant
Système d'information	Contrôle d'accès aux infrastructures
Intervenants	Système

Objectif		S'identifier afin de suivre l'activité
Précondition		Avoir sa CNI
Opération	**1**	S'identifier avec sa carte d'étudiant
Extension	**1. a**	Le system à bien reçu ses informations ?
	1. a.1	Si non : abandonner et retour à l'opération 1
	1. a.2	Si oui : continuer
	1. b	L'étudiant est-il sur la liste des participants ?
	1. b.1	Si nom : refuser l'accès à cet étudiant
	1. b.2	Si oui : continuer
	1. c	L'étudiant est-il inscrit en la matière ?
	1. c.1	Si nom : refuser l'accès à cet étudiant
	1. c.2	Si oui : continuer
	1. d	L'étudiant est-il solvable ?
	1. d.1	Si nom : refuser l'accès à cet étudiant
	1. d.2	Si oui : continuer
Post-condition		Accès autorisé

- Ajouter une activité

Tableau 5 Description textuelle du cas d'utilisation : « ajouter activité »

Cas d'utilisation		Ajouter activité
Acteur primaire		Enseignant
Système d'information		Contrôle d'accès aux infrastructures
Intervenants		Système
Objectif		Création d'une nouvelle activité
Précondition		S'authentifier et avoir le rôle enseignant ou admin
Opération	**1**	Remplir le formulaire et soumettre
Extension	**1. a**	Formulaire valide ?
	1. a.1	Si oui : continuer
	1. a.2	Si non : annuler l'ajout
Post-condition		Nouvelle activité créée

- Modifier une matière

Tableau 6 Description textuelle du cas d'utilisation : « modifier matière »

Cas d'utilisation	Modifier une matière
Acteur primaire	Administrateur
Système d'information	Contrôle d'accès aux infrastructures

Intervenants		Système
Objectif		Mise à jour d'une matière
Précondition		S'authentifier et avoir un rôle admin
Opération	**1**	Remplir le formulaire et soumettre
Extension	**1. a**	Formulaire valide ?
	1. a.1	Si oui : continuer
	1. a.2	Si non : annuler l'ajout
Post-condition		Matière modifié

- Ajouter un étudiant

Tableau 7 Description textuelle du cas d'utilisation : « ajouter etudiant »

Cas d'utilisation		Ajouter un étudiant
Acteur primaire		Administrateur
Système d'information		Contrôle d'accès aux infrastructures
Intervenants		Système
Objectif		Ajouter un étudiant dans la base de données
Précondition		S'authentifier et avoir un rôle admin
Opération	**1**	Remplir le formulaire et soumettre
Extension	**1. a**	Formulaire valide ?
	1. a.1	Si oui : continuer
	1. a.2	Si non : annuler l'ajout
Post-condition		L'étudiant a été ajouté dans la base de données

Dans cette partir, il était question pour nous de décrire les communications entre utilisateurs et système de quelques cas d'utilisation. Dans la suite, nous allons poursuivre cette fois ci dans la matérialisation des interactions entre objets pour mieux élucidé nos cas d'utilisation. Et pour la circonstance, UML nous offre le diagramme de séquence pour matérialiser cette chronologie des communications entre les objets du système participant et réalisant des cas d'utilisation.

2.1.3 Modélisation des interactions entre les objets

2.1.3.1 Diagramme de séquence du cas d'utilisation : « crée activité »

La création d'une activité est une action très complexe en fessant interagir plusieurs objets du système. La figue 6 présente les différentes communications chronologiques entre l'utilisateur (enseignant) et système lors de la création d'une activité.

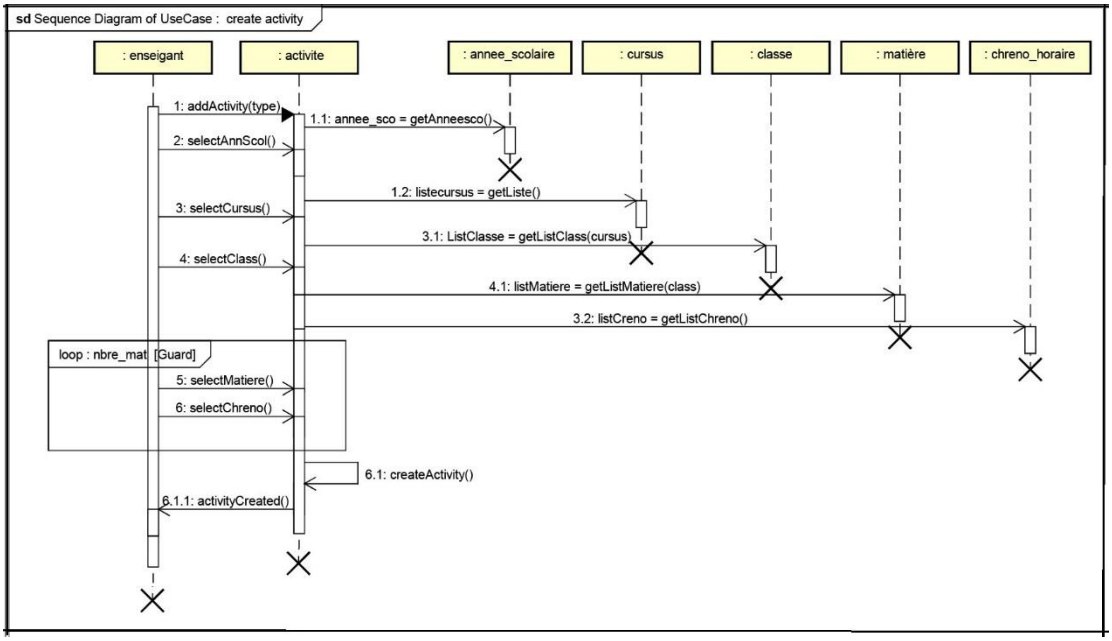


Figure 6 Diagramme de séquence du cas d'utilisation : « crée activité »

2.1.3.2 Diagramme de séquence du cas d'utilisation du cas d'utilisation : « participer à une activité »

En ce qui concerne la participation à une activité le étudiant doit se présenter devant sa salle muni de sa carte d'étudiant afin que le surveillant puisse scanner le code barre pour savoir s'il doit entrer ou pas. Et la figure 7 ci présente bien cette situation d'authentification.

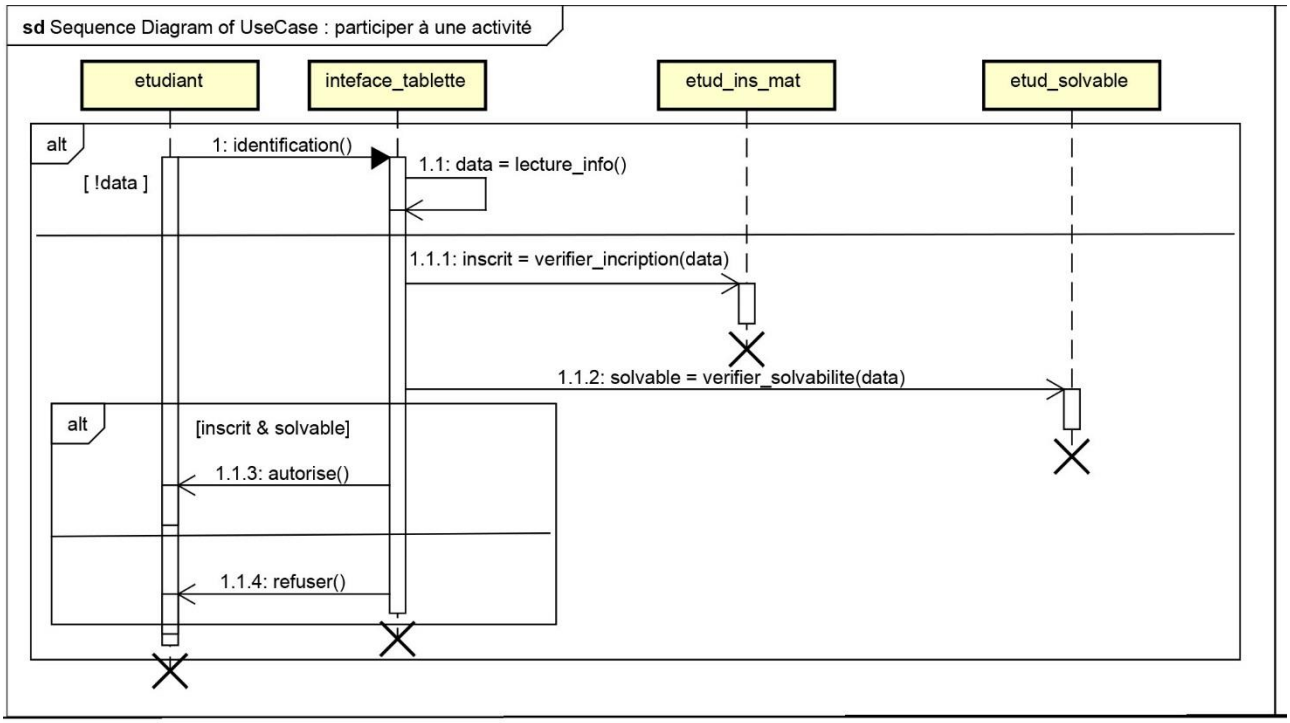


Figure 7 Diagramme de séquence du cas d'utilisation du cas d'utilisation : « participer à une activité »

2.1.4 Diagramme d'activité

2.1.4.1 Diagramme d'activité du cas d'utilisation : « créer activité »

Ici, nous allons montrer les différentes activités (étape d'UML) dans le processus de création d'activité (examen, cours ou TP) dans la figure qui suit :

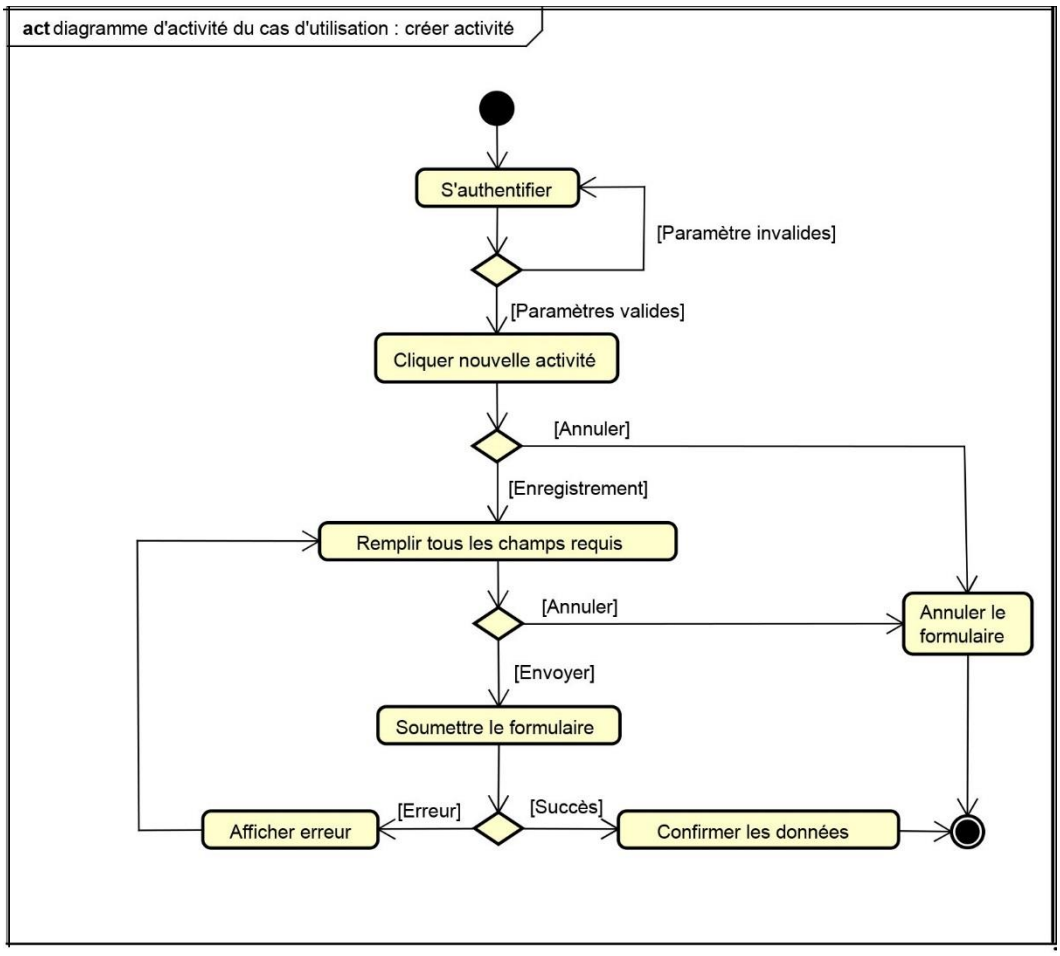


Figure 8 Diagramme d'activité du cas d'utilisation : « créer activité »

2.1.4.2 Diagramme d'activité du cas d'utilisation : « voire rapport d'activité »

L'une des fonctionnalités que l'étudiant peut effectuer dans cette application, est la visualisation de son rapport d'activité que nous illustrons au niveau de la figure 7.

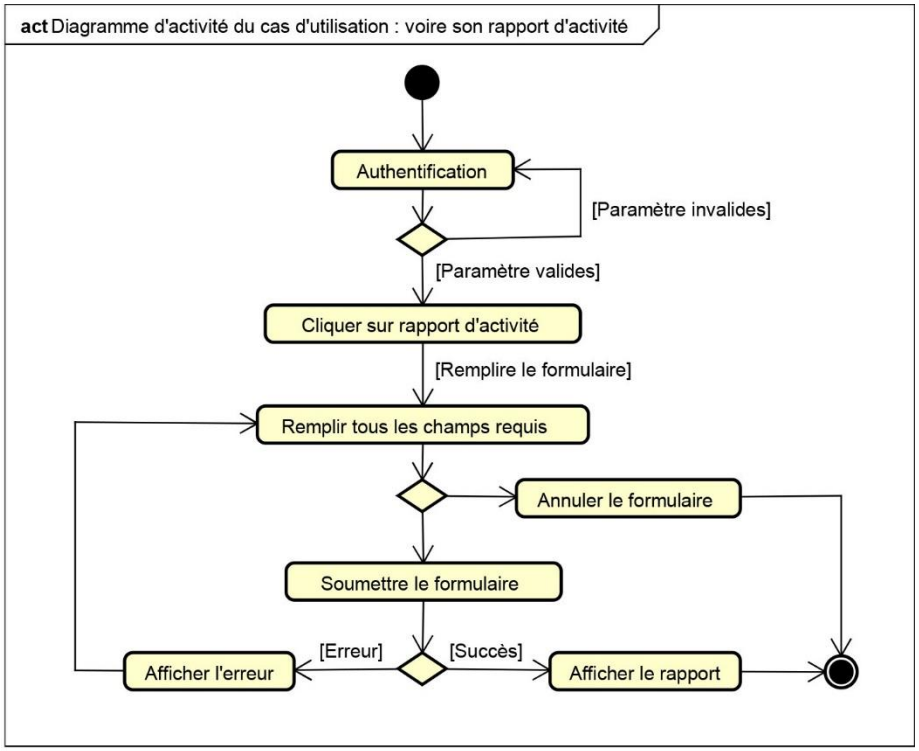


Figure 9 Diagramme d'activité du cas d'utilisation : « voire rapport d'activité »

2.1.4.3 Diagramme d'état-transition d'une activité

Lorsqu'une activité est créée, elle possède des états en fonction du temps et ses différents états sont très bien présentés dans la figure qui suit :

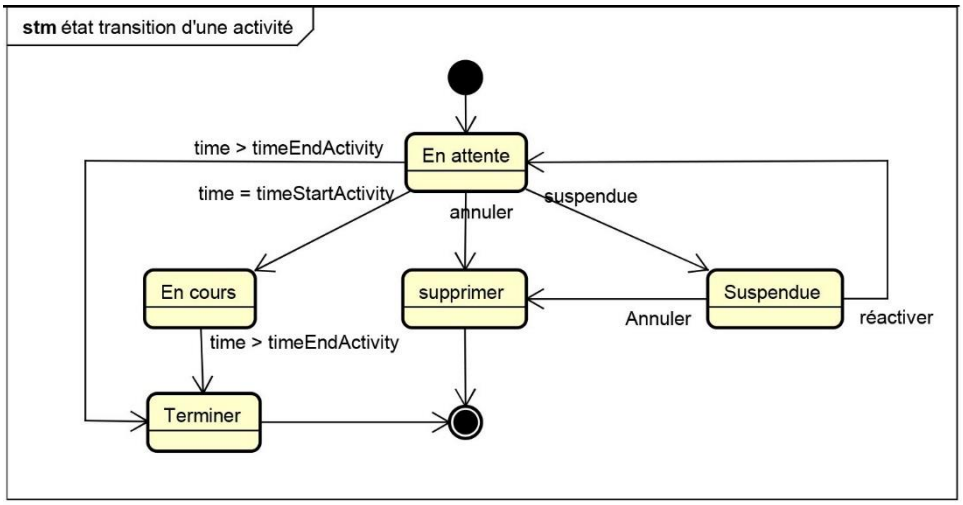


Figure 10 Diagramme d'état-transition d'une activité

2.1.5 Présentation du modèle de navigation de l'application web

La figure ci-dessous présente les différents scénarios que les utilisateurs auront à rencontrer dans cette application. Initialement, l'utilisateur est amené soit à s'authentifier s'il possède déjà un compte ou soit à créer un compte s'il n'en possède pas encore. Pour les utilisateurs possèdent un compte après l'authentification ils auront dans les menus de navigation en fonction des rôles comme présenté à la figure 12 où seule les menus les plus important sont représenté.

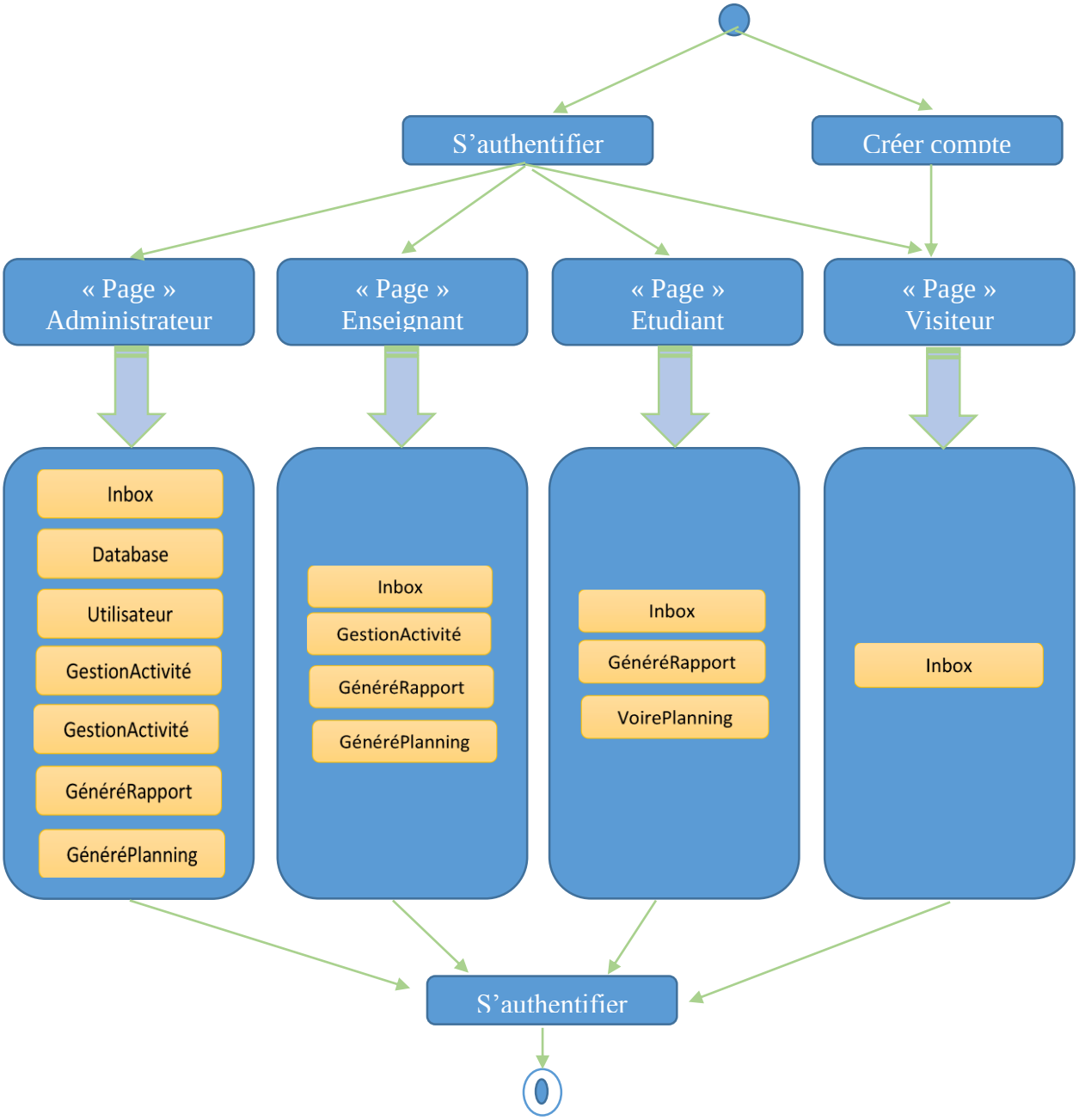


Figure 11 Modèle de navigation relatif à l'application web

2.2 Conception technique, fonctionnelle et objet du système

2.2.1 Architecture technique et fonctionnelle du système

La figure suivante présente le fonctionnement architecturel et technique du système du système :

Si par exemple nous voulons afficher la liste des étudiants d'une activité, nous allons procéder comme suit :

1. Remplir le formulaire de spécification de l'activité, qui est une suite de code HTML et CSS et le soumettre ;
2. JavaScript transmet la requête AJAX au serveur web Apache à travers le protocole http ;
3. Le serveur web examine la requête grâce au Langage de script PHP qui interroge la base de données afin d'extraire les informations ;
4. Une fois les informations récupérer de a base de données, le serveur web renvoie la réponse via le protocole HTTP.
5. JavaScript se chargera d'afficher la liste des étudiants trouvé en utilisant du code HTML qui a sont peut utiliser des scripts CSS pour l'affichage.

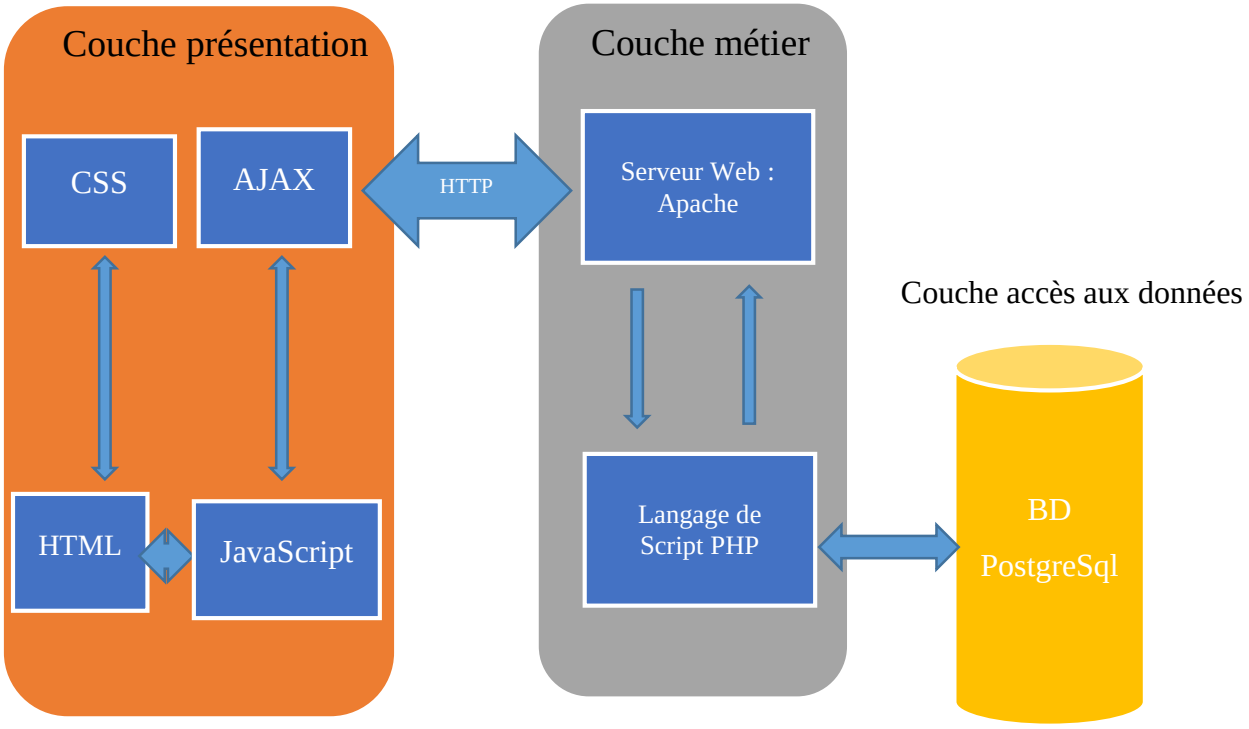


Figure 12 Architecture fonctionnelle et technique du système

2.2.2 Dictionnaire de données

Le tableau ci-dessous donne une description et type de données sur les noms d'attribut en effet ces attributs sont pour la plus part codifier afin de ne pas trop utiliser de l'espace dans la base de données.

Tableau 8 Dictionnaire de données

Classes	Attributs	Types	Descriptions
activites	id	serial	Identifiant unique d'un objet de la table activité
	id_annee	*Integer*	Clé étrangère de l'année
	id_semestre	*Integer*	Clé étrangère du semestre
	id_niveau	*Integer*	Clé étrangère du niveau
	date_debut_activite	Date	Date de démarrage de l'activité
	date_fin_activite	Date	Date de la fin de l'activité
annee_academiques	id	serial	Identifiant unique d'un objet de la table
	libelle_annee	*String*	Intitulé de l'année académique
classes	id	serial	Identifiant unique d'un objet de la table classes
	id_cursus	*Integer*	Clé étrangère du cursus
	id_departement	*Integer*	Clé étrangère du département
	id_niveau	*Integer*	Clé étrangère du niveau
	code_classe	*String*	Code de la classe
	libelle_classe	*String*	Libelle de la classe
	effectif_classe	*Integer*	Nombre de place disponible dans une classe
cours	id	serial	Identifiant unique d'un objet de la table
	id_activite	*Integer*	Clé étrangère d'une activité
	id_enseigant	*Integer*	Clé étrangère de l'enseignant
	id_matiere	*Integer*	Clé étrangère de la matière
	id_creneau	*Integer*	Clé étrangère du créneau horaire
	date_cours	Date	Date de passage de cette matière
creneaux_horaires	id	serial	Identifiant unique d'un objet de la table créneau horaire
	duree	*Integer*	Durée de ce créneau
	libelle_creneaux	*String*	Libelle du créneau
cursus_accs	id	serial	Identifiant unique d'un objet de la table cursus académique
	code	*String*	Code du cursus
	libelle	*String*	Libelle du cursus
departements	id	serial	Identifiant unique d'un objet de la table département
	code_departement	*String*	Code du département
	libelle_departement	*String*	Libelle du département
enseignants	id	serial	Identifiant unique d'un objet de la table enseignant
	matricule_enseignant	*String*	Matricule de l'enseignant
	id_user	*Integer*	Clé étrangère de l'utilisateur
	id_departement	*Integer*	Clé étrangère du département
	grade	*String*	Grade de l'enseignant
	fonction	*String*	Fonction de l'enseignant
etudiants	id	Serial	Identifiant unique d'un objet de la table étudiant
	matricule_etudiant		Matricule de l'étudiant
	id_user	*Integer*	Clé étrangère de l'utilisateur

	diplome_entre	*String*	Diplôme d'entré de l'étudiant
examens	id	Serial	Identifiant unique d'un objet de la table examen
	id_activite	*Integer*	Clé étrangère de l'activité
	id_ens_chef_dpt	*Integer*	Clé étrangère du chef de département
	id_matiere	*Integer*	Clé étrangère de la matière
	id_creneau	*Integer*	Clé étrangère du créneau horaire
	id_surveillant	*Integer*	Clé étrangère du surveillant de l'examen
	id_session	*Integer*	Clé étrangère de la session
	date_examen	Date	Date de passage de la matière
mails	id	Serial	Identifiant unique d'un objet de la table mail
	id_user_from	*Integer*	Clé étrangère de l'émetteur
	id_user_to	*Integer*	Clé étrangère du destinataire
	objet	*String*	Objet du mail
	libelle	*String*	Contenu du mail
	lue	*String*	Statut du mail
matieres	id	Serial	Identifiant unique d'un objet de la table matière
	code_matiere	*String*	Code de la matière
	libelle_matiere	*String*	Libelle de la matière
	nbr_credit	Integer	Nombre de crédit de la matière
niveaux	id	serial	Identifiant unique d'un objet de la table niveaux
	libelle_niveau	*String*	Libelle du niveau
password_resets	email	*String*	Email de l'utilisateur
	token	*String*	Clé pour la modification
roles	id	Serial	Identifiant unique d'un objet de la table rôle
	name	*String*	Nom du rôle
	description	*String*	Description du rôle
salles	id	Serial	Identifiant unique d'un objet de la table salle
	code_salle	*String*	Code de la salle
	libelle_salle	*String*	Libelle de la salle
	nbre_places	*Integer*	Nombre de place disponible dans la salle
semestres	id	Serial	Identifiant unique d'un objet de la table semestre
	libelle_semestre	*String*	Libelle du semestre
sessions	id	Serial	Identifiant unique d'un objet de la table session
	libelle_session	*String*	Libelle de la session
surveillants	id	Serial	Identifiant unique d'un objet de la table surveillant
	id_user	*Integer*	Clé étrangère de l'utilisateur
tps	id	Serial	Identifiant unique d'un objet de la table tp
	id_activite	*Integer*	Clé étrangère de l'activité
	id_enseigant	*Integer*	Clé étrangère de l'enseignant
	id_matiere	*Integer*	Clé étrangère de la matière
	id_creneau	*Integer*	Clé étrangère du créneau horaire
	date_tp	Date	Date de passage du tp
users	id	Serial	Identifiant unique d'un objet de la table utilisateur
	name	*String*	Nom de l'utilisateur
	prenom	*String*	Prénom de l'utilisateur
	sexe	*String*	Sexe de l'utilisateur
	date_nais	Date	Date de naissance de l'utilisateur
	telephone	*Integer*	Numéro(s) de téléphone de l'utilisateur

	Email	*String*	Email de l'utilisateur (login)
	password	*String*	Mot de passe de l'utilisateur (password)
	info_codebar	text	Donnée du code barre de carte d'étudiant
	info_empreinte	text	Donnée sur l'empreinte digitale de l'étudiant
	Photo	text	Photo de l'étudiant (avatar)
	remember_token	*String*	Clé de mémorisation du mot de passe
salle_activites	id	Serial	Identifiant unique d'un objet de la table salle d'une activité
	id_activite	*Integer*	Clé étrangère de l'activité
	id_salle	*Integer*	Clé étrangère de la salle
role_user	id	Serial	Identifiant unique d'un objet de la table rôle-utilisateur
	role_id	*Integer*	Clé étrangère du rôle
	user_id	*Integer*	Clé étrangère de l'utilisateur
etud_scolariser_clas	id	serial	Identifiant unique d'un objet de la table étudiant scolarisé
	id_etudiant	*Integer*	Clé étrangère de l'étudiant
	id_classe	*Integer*	Clé étrangère de la classe
	id_annee	*Integer*	Clé étrangère de l'année académique
etud_participe_tps	id	serial	Identifiant unique d'un objet de la table étudiant-tp
	id_tp	*Integer*	Clé étrangère de la table étudiant-tp
	id_etud_ins_mat	*Integer*	Clé étrangère de la matière
	statut	*Integer*	Statut de l'étudiant
etud_etudie_cours	id	serial	Identifiant unique d'un objet de la table étudiant-cours
	id_cours	*Integer*	Clé étrangère du cours
	id_etud_ins_mat	*Integer*	Clé étrangère de la table étudiant-matière
	statut	*Integer*	Statut de l'étudiant
etud_compose_examens	id	serial	Identifiant unique d'un objet de la table étudiant-examen
	id_examen	*Integer*	Clé étrangère de l'examen
	id_etud_ins_mat	*Integer*	Clé étrangère de la table étudiant-examen
	statut	*Integer*	Statut de l'étudiant
activite_conc_classes	id	serial	Identifiant unique d'un objet de la table activité-classe
	id_activite	*Integer*	Clé étrangère de l'activité
	id_classe	*Integer*	Clé étrangère de la classe
ens_chef_dpts	id	serial	Identifiant unique d'un objet de la table enseignant-département
	id_enseignant	*Integer*	Clé étrangère de l'enseignant
	id_departement	*Integer*	Clé étrangère du département
	date_debut_diriedpt	date	Date de prise de service
	date_fin_diriedpt	date	Date de passation de service
	statut	*Integer*	Statut de l'enseignant

2.2.3 Diagramme de classe

La figure 15 ci-dessous est une représentation globale des objets de notre système ainsi que les liens qui existent entre eux.

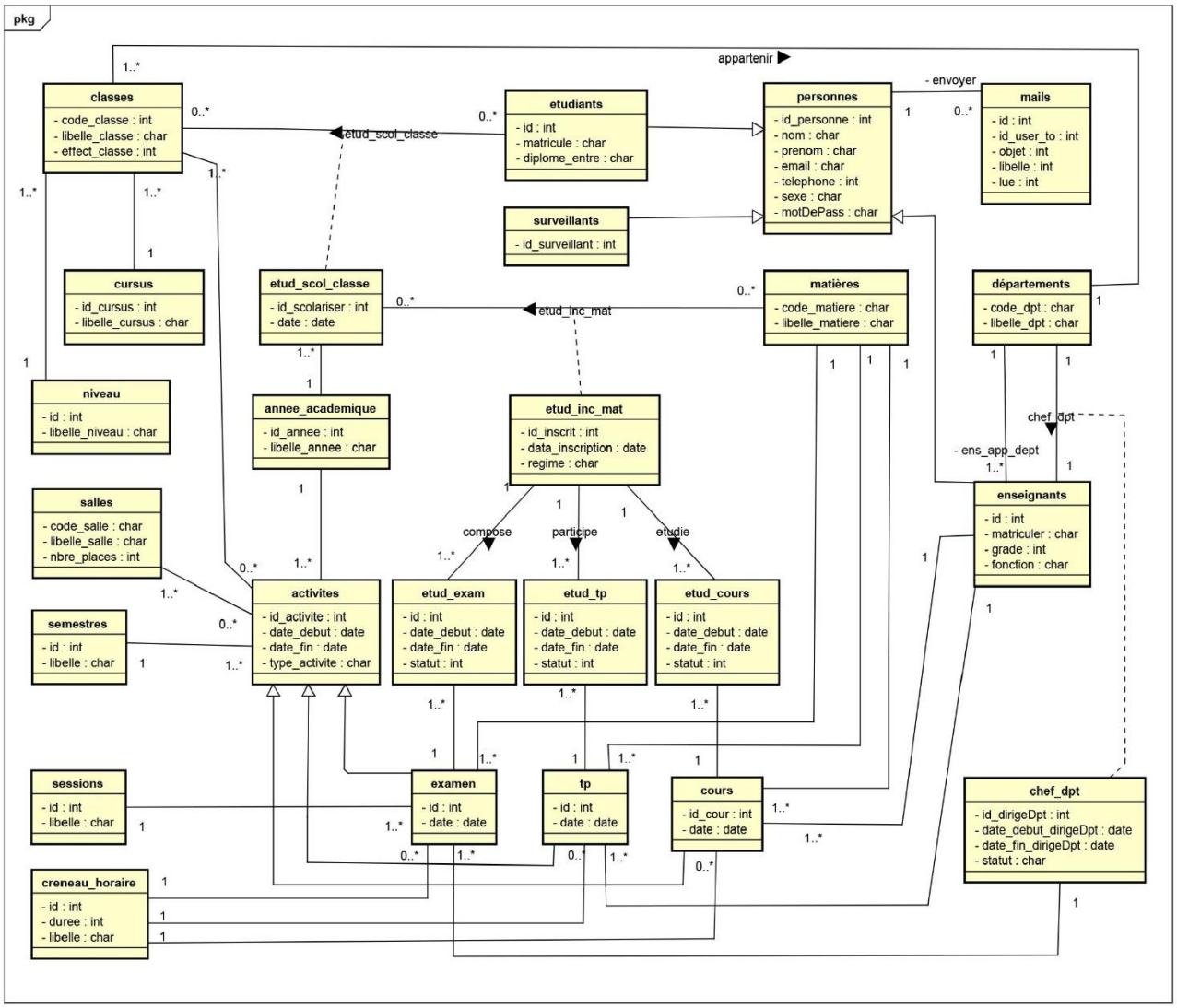


Figure 13 Diagramme de classe

2.2.4 Modèle physique de données

Après migration des clés, nous obtenons une représentation physique de nos objets et c'est avec ce modèle que nous allons créer notre base de données. La figure 16 ci-dessous représente ce modèle ainsi que les liens et clé étrangère qui existent entre les différentes tables.

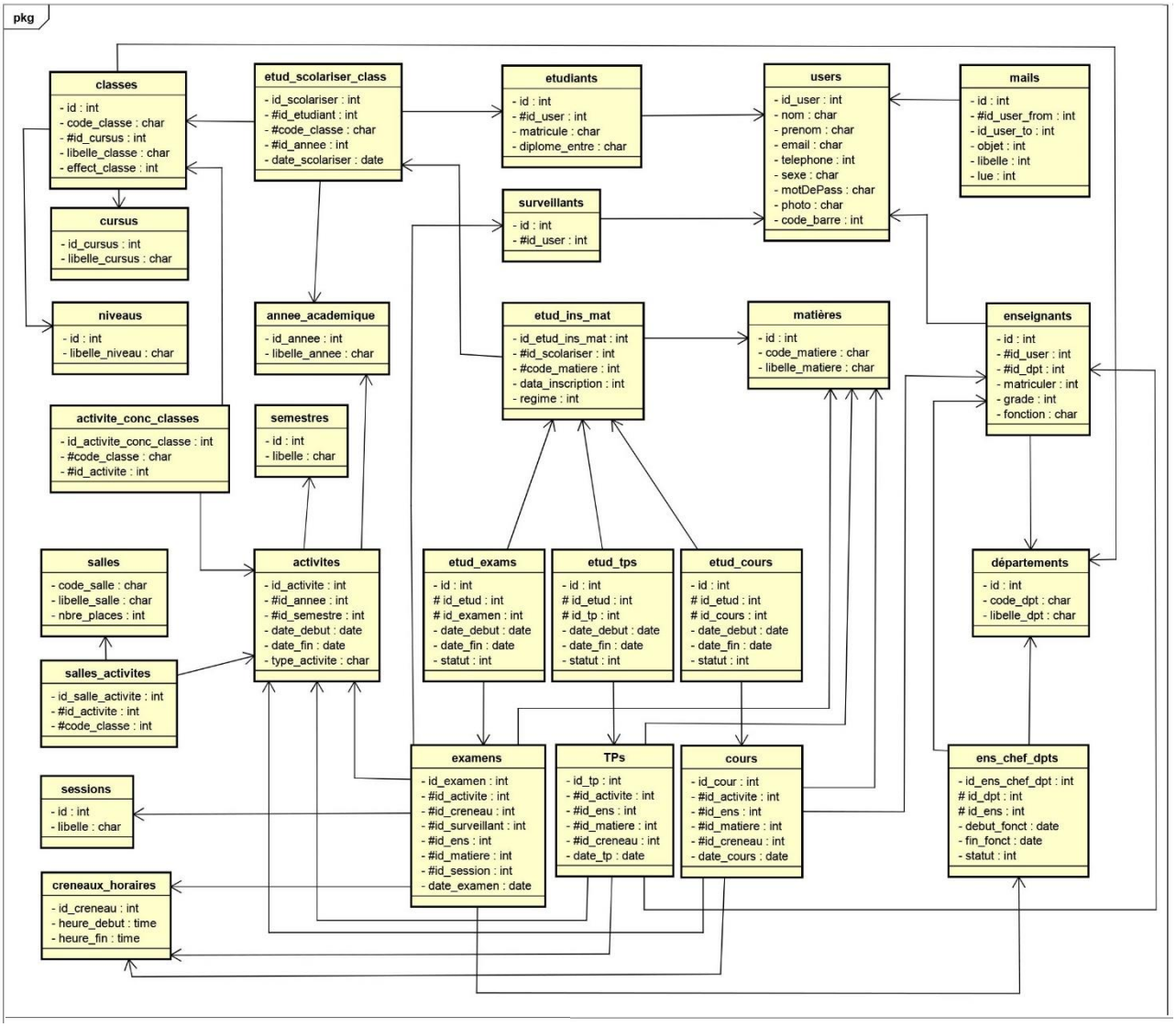


Figure 14 Modèle physique de données

Chapitre 3 Implémentation, Techniques de Programmation et Résultats

3.1 Technique de développement

3.1.1 Usage du pattern de développement MVC (Model Vue Controller)

Pour implémenter notre application nous avons adopté ce pattern (MVC) car il présente plusieurs tels : la modularité de l'application ce qui nous a permis de bien travailler en groupe, ainsi facilite le développement de l'application en suivant bien sur la logique orienté objet, de plus il rend aussi l'application facile à maintenir. Dans la figure ci-dessous nous présentons une lustration de ce pattern dans notre application en mettent les composants qui sont directement impliqués.

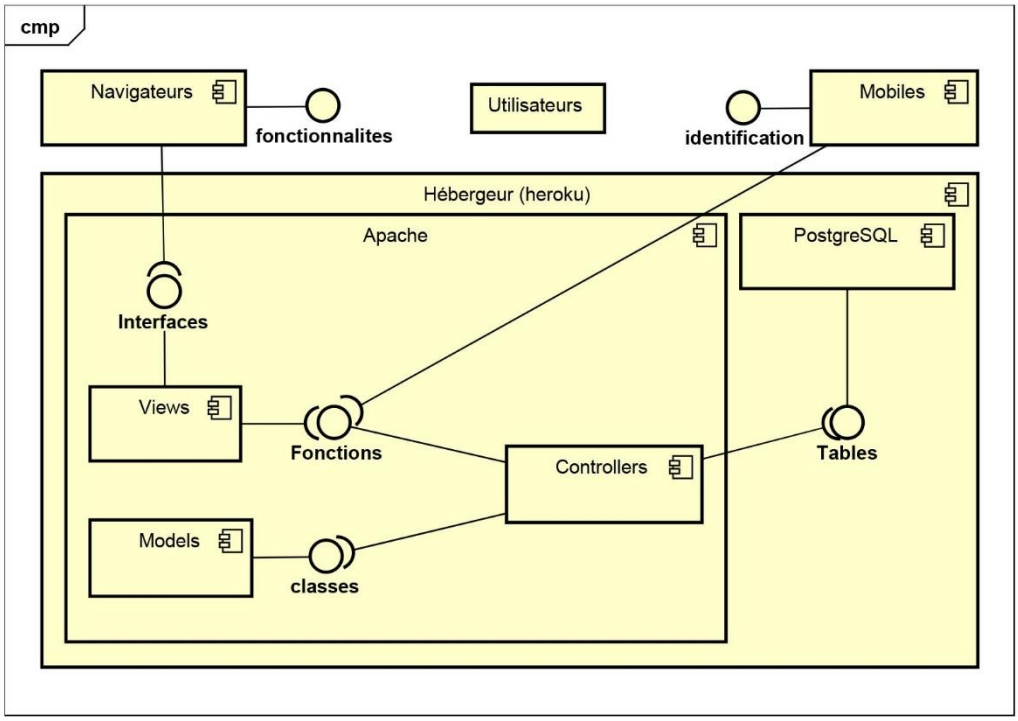


Figure 15 Diagramme de composants du pattern MVC

3.1.2 Arborescence du système

La figure 18 ci-dessous présente l'arborescence de notre système, cette architecture est propre à Laravel 5.5 et après cette figure nous allons donner les explications sur le contenu des dossiers portant un numéro dans cette arborescence.

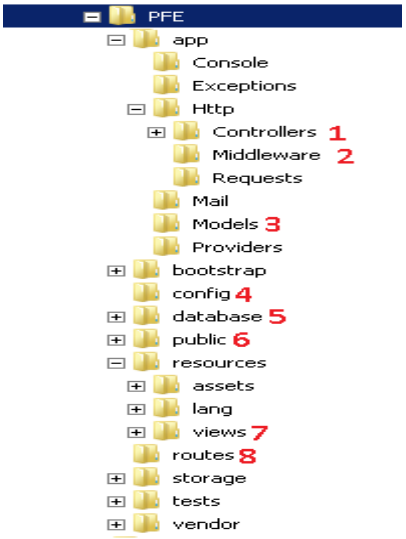


Figure 16 Arborescence du système

Le dossier portant le numéro : (source : [W-get_laravel5.5-18])

1. Contient tous les scripts PHP servant d'intermédiaire entre la vue (7) et la base de données ;
2. Contient les middlewares pour assurer la confidentialité des utilisateurs connectés en plus de middleware auth et guest de Laravel, nous avons aussi créé des middlewares pour chaque rôles des utilisateurs (administrateur, enseignant, étudiant) ;
3. Contient nos différents modèles ;
4. Contient les fichiers de configuration de l'application tels que le nom, la clé, connexion à la base de données ;
5. Contient les scripts permettent de générer les tables de la base de données ainsi que les migrations ;
6. Contient le fichier index pour le serveur et aussi les scripts (JavaScript, CSS), les images ;
7. Contient tous les fichiers de présentation (vue) ;
8. Contient la déclaration des routes que les requêtes (de type POST, GET, DELETE etc.) doivent suivre de la vue au contrôleur et vice-versa.

3.2 Déploiement de l'application

3.2.1 Environnements et outils de déploiement

Pour le déploiement de cette application, nous avons comme environnement de travail un ordinateur portable de maque HP et ayant les caractéristiques suivant :

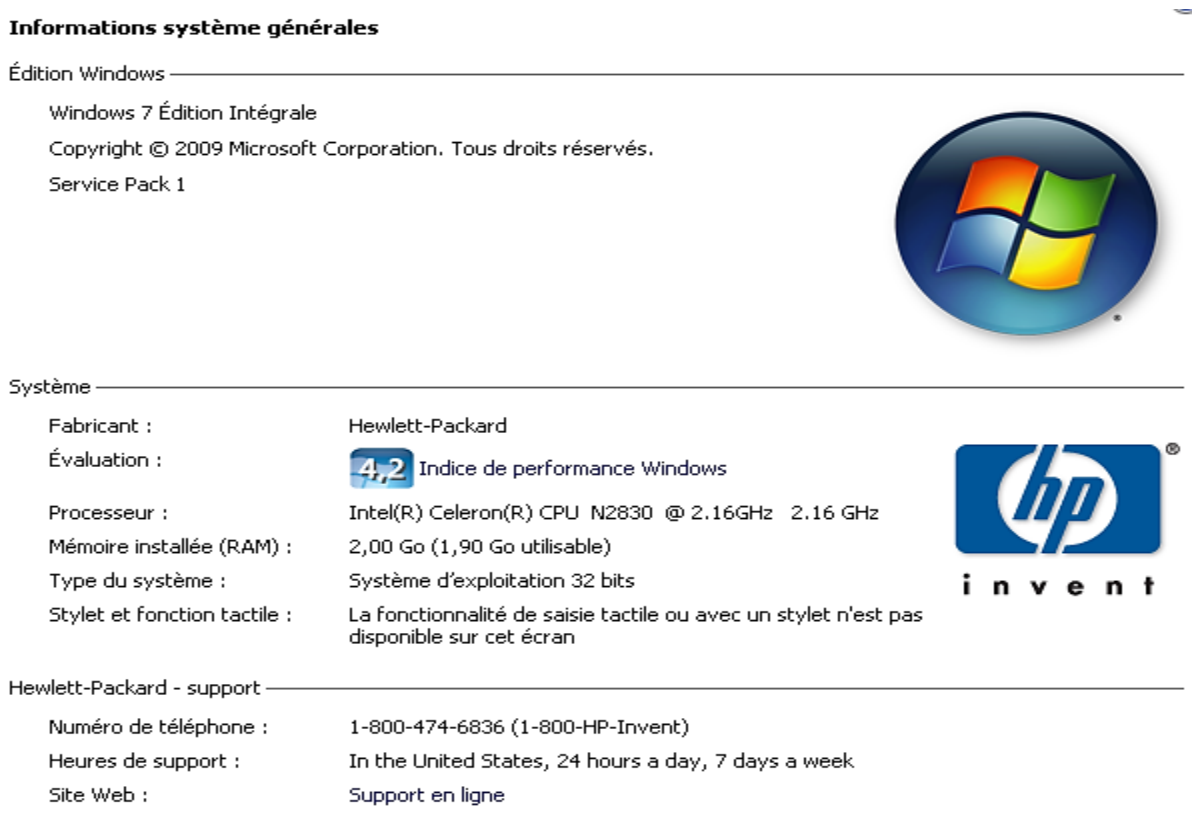


Figure 17 Informations système générales

En ce qui concerne les outils de déploiement on a :

- ✓ Télécharger et installer **composer** (comme décrit dans [W-get_composer-18] qui est un gestionnaire de dépendance PHP
- ✓ Télécharger, installer et configurer **Git Bash** (comme décrit dans [p-git_tuto]) qui est un terminal permettant d'exécuter les commandes linux sur Windows et servant aussi de liaison entre notre dépôt local avec le dépôt ce trouvant en ligne sur www.github.com

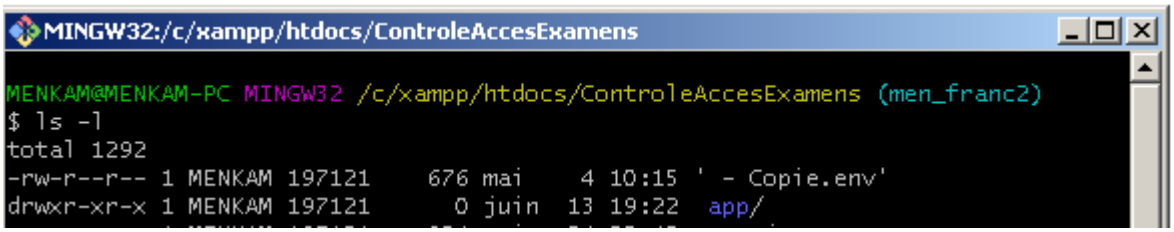


Figure 18 Interface de Git Bash

- ✓ Télécharger, installer et configurer **CLI Heroku** comme décrit dans [W-heroku-19] qui permet de communiquer avec l'hébergeur gratuit heroku.
- ✓ Télécharger et installer xampp 7.2 (version de PHP >=7.1)
- ✓ Installer l'application Laravel5.5 en ligne de commande avec composer : (source en ligne [W-heroku_cli-18])

```
MINGW32:/c/Users/MENKAM

MENKAM@MENKAM-PC MINGW32 ~
$ composer create-project --prefer-dist laravel/laravel=5.5 PFE
```

Figure 19 Installation de Laravel 5.5

3.2.2 Déploiement en local

Pour ce qui est du déploiement en local, nous allons installer un SGBD (PostgreSQL pour notre) et créer une base de données vide pourtant le même nom que celui qui se trouve dans le fichier **.env**. En suite, à l'aide de **php artisan** de Laravel nous allons effectuer la migration pour créer les tables dans l'application dans la base de données avec la commande suivante :

```
MINGW32:/c/xampp/htdocs/ControleAccesExamens

MENKAM@MENKAM-PC MINGW32 /c/xampp/htdocs/ControleAccesExamens (men_franc2)
$ php artisan migrate
*************************************
*     Application In Production!    *
*************************************

 Do you really wish to run this command? (yes/no) [no]:
 > yes

Migration table created successfully.
Migrating: 2014_10_12_000000_create_users_table
Migrated:  2014_10_12_000000_create_users_table
```

Figure 20 Migration de la base de données

3.2.3 Déploiement en ligne

Pour le déploiement en signe (sur heroku dans notre cas) les prérequis de la machine pour déployer un projet PHP sont :

- ✓ **PHP** version >= 7.0.0
- ✓ Le gestionnaire de dépendances **Composer**
- ✓ **Git**
- ✓ Créer un **compte sur Heroku**
- ✓ Installer **Heroku CLI**

La procédure de déploiement se trouve en annexe 2 (source [W-deploie-laravel-18]).

3.3 Résultats concrets du développement

3.3.1 Premier résultat : Interface de gestion d'activité

3.3.1.1 Présentation

Ce premier résultat présente une interface allant de la création à suppression d'une activité en passant par l'ajout des matières, classe et salle.

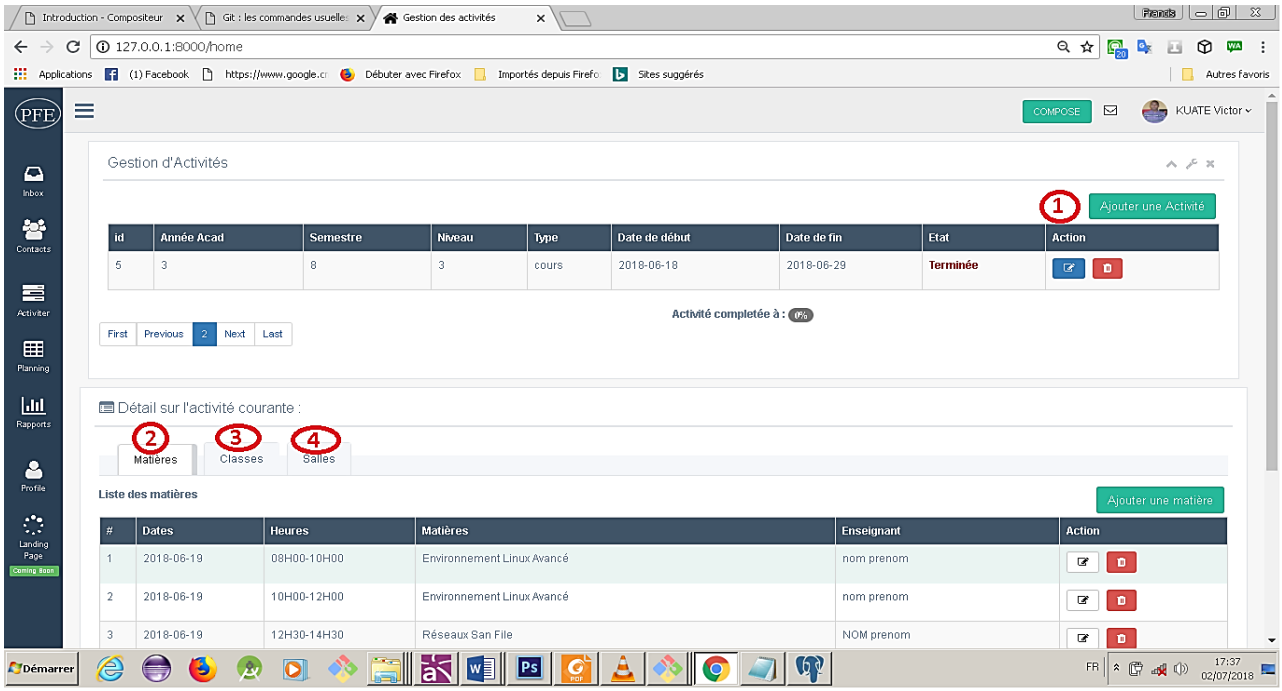


Figure 21 Interface de gestion d'activite

Sur la figure précédente le numéro :

1) Permet de créer une nouvelle activité

2) Permet d'ajouter des matières à l'activité courante

3) Permet de définir des classes à l'activité courante

4) Permet de définir des salles à l'activité courante

3.3.1.2 Algorithme d'obtention du resultat

La création d'une activité commence par La délimitation de l'activité mère à savoir :

- ✓ préciser la date de début et de fin de l'activité, pour la sélection de ces dates nous proposons dans le formulaire une plage allant de la date courante à la date de fin des activités dans l'institut et lorsque la date de début est choisie il est impossible de choisir une date de fin inférieur à la date de début. Ce qui permet de créer dans activité crédible ;

- ✓ Le type d'activité, qui peut être soit l'examen, soit le cours ou même le TP ;
- ✓ Le semestre

Après cette précision nous pouvons à présent passer soit aux numéros 2, 3 ou 4 sur la figure. Lors de l'ajout d'une matière à une activité, la date de de passage de cette activité sera proposée dans une plage délitée par la date de début et de fin l'activité. Ce permet de ne pas créer matière d'activité orpheline.

3.3.2 Deuxième résultat : Suivie en temps réel du déroulement d'une activité

3.3.2.1 Présentation

Ce résultat représente l'une des innovations dans le suivie d'une activité car cette interface de suivi (figure 24) présente pour chaque créneau horaire, l'activité qui se déroule dans cette intervalle de temps ainsi qu'un bouton dans la colonne action de chaque activité trouvé qui permet de visualiser la liste des étudiants en salle (figure 25).

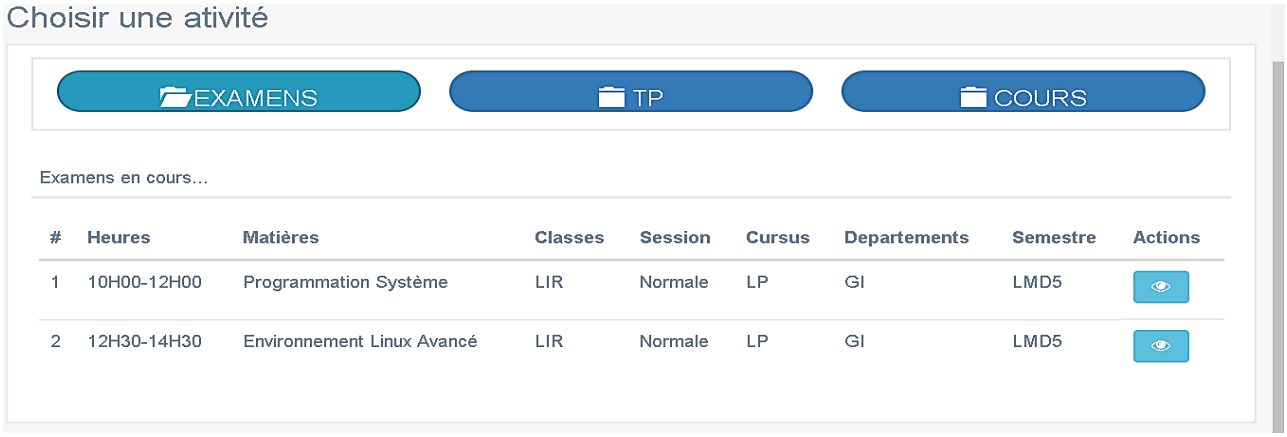


Figure 22 Activité en cours (en temps réel)

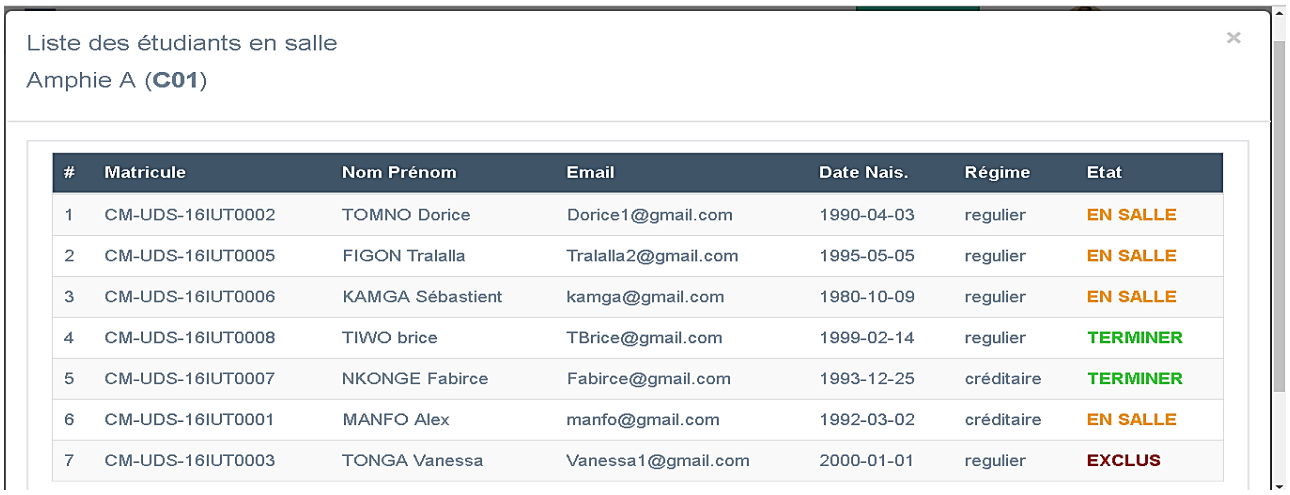


Figure 23 Liste des étudiants en salle (en temps réel)

3.3.2.2 Algorithme d'obtention du résultat

Pour avoir ce résultat, nous effectuons tous les 2000 millisecondes un requête dans la base et le résultat est immédiatement afficher à l'écran.

```
90  function afficherElement(e)
91  {
92      var classe =  $("#btn-"+e).attr("class");
93      if(classe == "btn btn-round btn-primary btn-lg form-control") {
94          intitPage();
95          //temps = 2000 milisecondes
96          if(e=="examen")
97              intervalExamen = setInterval(getListExamen,temps);
98          if(e=="cours")
99              intervalCours = setInterval(getListCours,temps);
100         if(e=="tp")
101             intervalTp = setInterval(getListTp,temps);
102
103         $("#btn-" + e).find("span").addClass("glyphicon-folder-open");
104         //$("#btn-"+e).removeClass("btn-primary");
105         $("#btn-" + e).addClass("btn-info");
106         $("#btn-" + e).addClass("active");
107         $("#content-" + e + "-enCours").slideDown();
108         $("#btn-" + e).find("span").addClass("glyphicon-folder-open");
109
110     }
111 }
```

Figure 24 function qui affiche les activités en cours

Lorsqu'une activité est terminée, elle n'apparait plus dans la liste et si on clique sur le bouton voire dans la colonne action une nouvelle fenêtre s'affichera avec les noms des étudiants en salle avec des états interprétés comme suit :

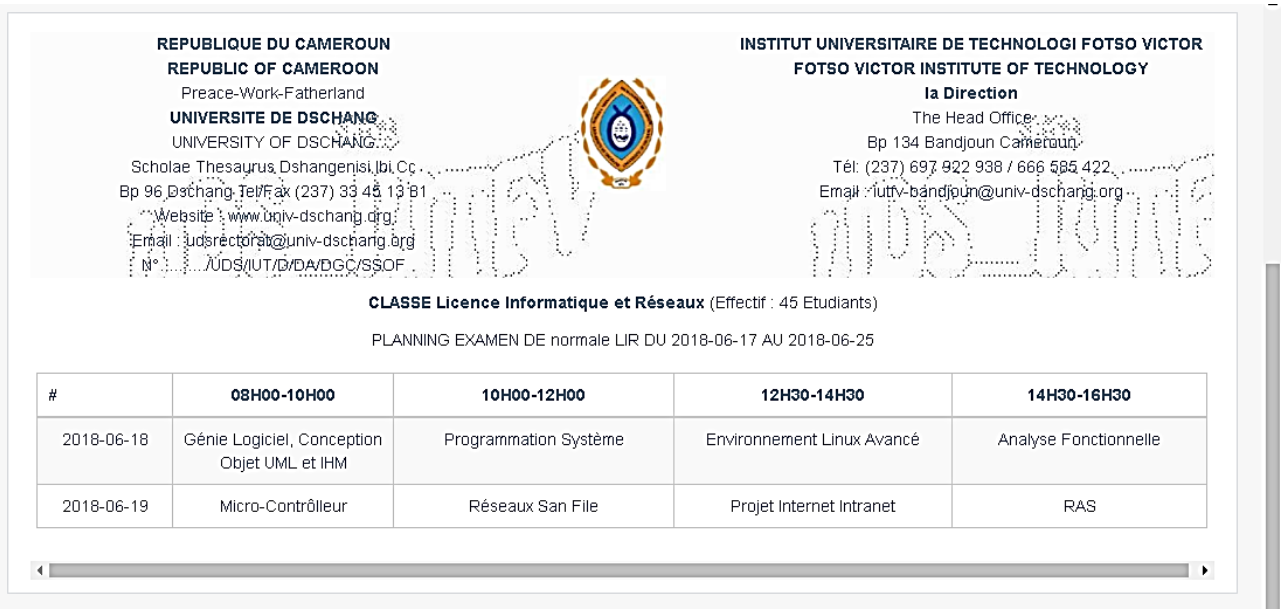
- Etat = 0 : étudiant en salle ;
- Etat = 1 : étudiant ayant terminé
- Etat = 2 : étudiant exclus de la salle

3.3.2.3 Application du résultat

Le suivie du déroulement d'une activité permet à l'administration de savoir en temps réel le nombre, le nom et le comportement de étudiant en salle et cette opération être important dans la mesure où celui en charger de suivre l'activité en cours n'a pas assez de temps pour la ronde des salles de composition où même à cause du climat défavorable (pluie où neige dans le campus).

3.3.3 Troisième résultat : Générer le planning d'une activité

3.3.3.1 Présentation



REPUBLIQUE DU CAMEROUN
REPUBLIC OF CAMEROON
Preace-Work-Fatherland
UNIVERSITE DE DSCHANG
UNIVERSITY OF DSCHANG
Scholae Thesaurus Dshangenisi Ibi Cc
Bp 96 Dschang Tel/Fax (237) 33 45 13 81
Website : www.univ-dschang.org
Email : udsrectorat@univ-dschang.org
N° ...../UDS/IUT/D/DA/DGC/SSOF

INSTITUT UNIVERSITAIRE DE TECHNOLOGI FOTSO VICTOR
FOTSO VICTOR INSTITUTE OF TECHNOLOGY
la Direction
The Head Office
Bp 134 Bandjoun Cameroun
Tél: (237) 697 922 938 / 666 585 422
Email : iutfv-bandjoun@univ-dschang.org

CLASSE Licence Informatique et Réseaux (Effectif : 45 Etudiants)

PLANNING EXAMEN DE normale LIR DU 2018-06-17 AU 2018-06-25

#	08H00-10H00	10H00-12H00	12H30-14H30	14H30-16H30
2018-06-18	Génie Logiciel, Conception Objet UML et IHM	Programmation Système	Environnement Linux Avancé	Analyse Fonctionnelle
2018-06-19	Micro-Contrôlleur	Réseaux San File	Projet Internet Intranet	RAS

Figure 25 Planning de la normale allant du 17 au 25 juin 2018 (simulation)

3.3.3.2 Algorithme d'obtention du résultat

Pour avoir ce planning, l'utilisateur doit au préalable remplir le formulaire (**annexe 3, figure 35**) et le soumettre au contrôleur qui a son tour effectue la requête comme l'indique la figure de la page suivant et retourne le résultat dans la variable Ajax pour l'affichage.

```
68          return DB::select("
69            SELECT
70              $nom_date datem,
71              creneaux_horaires.libelle_creneaux,
72              matieres.libelle_matiere,
73              activites.date_debut_activite,
74              activites.date_fin_activite,
75              activites.type_activite,
76              classes.code_classe,
77              classes.libelle_classe,
78              classes.effectif_classe
79            FROM
80              public.activites,
81              public.matieres,
82              public.creneaux_horaires,
83              public.$table,
84              public.activite_conc_classes,
85              public.classes
86            WHERE
87              $table.id_activite = activites.id AND
88              $table.id_matiere = matieres.id AND
89              $table.id_creneau = creneaux_horaires.id AND
90              activite_conc_classes.id_activite = activites.id AND
91              classes.id = activite_conc_classes.id_classe AND
92              activites.id = '$idActivite'
93            ORDER BY
94              $nom_date ASC,
95              creneaux_horaires.libelle_creneaux ASC;
96          ");
97        }
```

Figure 26 Requête de sélection des matières d'une activité

3.3.3.3 Application du résultat

Avoir le planning des activités nouvellement créée est très important car il permet aux acteurs (enseignants, surveillant, étudiants) de se préparer pour la circonstance et avoir un planning à l'avance est un atout pour le bon déroulement des activités.

3.3.4 Quatrième résultat : Générer le rapport d'une activité

3.3.4.1 Présentation

Cette interface permettant aux enseignants de générer le rapport après une activité précise.

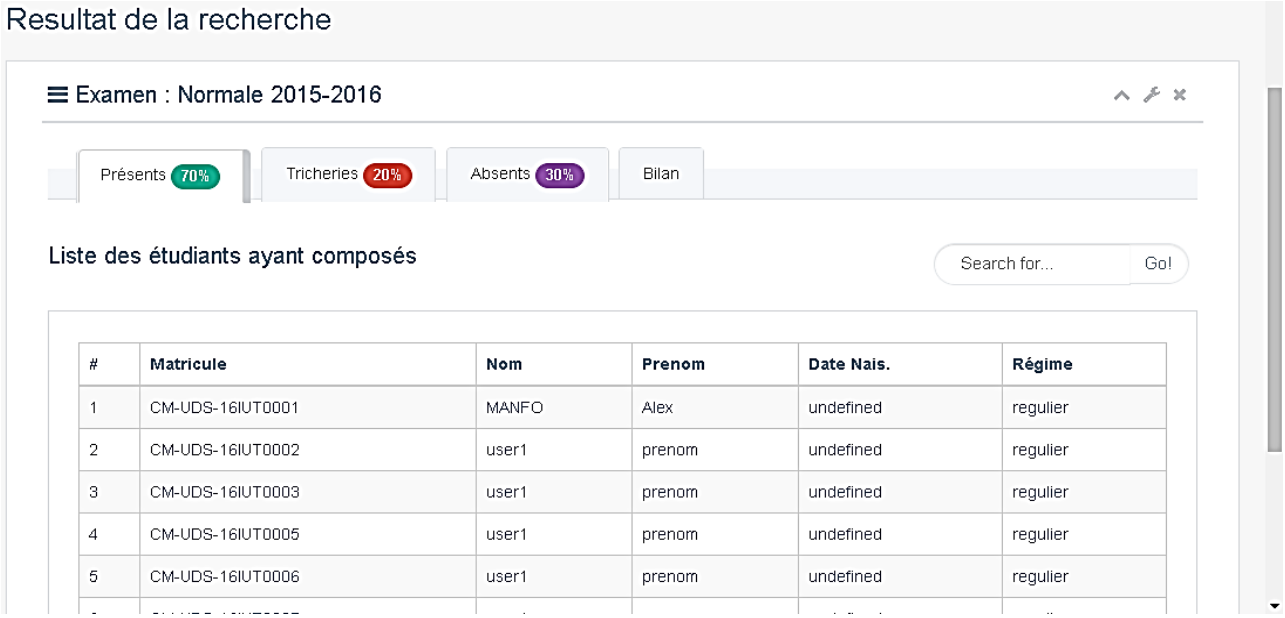


Figure 27 Rapport d'examen de normale de la période du 17 au 25 juin 2018 (simulation)

3.3.4.2 Algorithme d'obtention du résultat

Après avoir remplir le formulaire et le soumettre, le contrôleur dédié à cette opération effectue une requête SQL à la base de données ***(voir annexe 3, figure 36 et 37)*** pour récupérer la liste des étudiants et le retourner dans la variable Ajax qui a sollicité cette requête et par la suite, on prépare l'affichage en JavaScript utilisant les balises HTML que la vue se chargera d'affiche comme le présente la figure 27.

3.3.4.3 Application du résultat

Ce résultat est d'une grande importance comme la précédente car elle permet aux dirigeants d'avoir les statistiques dans les meilleur délais sur le déroulement d'une activité à tout moment et de prendre des décisions rapide en cas fraude par exemple.

Chapitre 4 Apport, Leçons, Difficultés et Perspectives

4.1 Apports

Dans le passé, les mêmes résultats était presque obtenu mais avec des heures voire même des jours de retard ce qui rend parfois la publication des résultats long où même la production des rapports pour les archives pénible car les documents et procès-verbaux ne sont pas encore disponible. Avec ce système, le traitement et la vérification des rapports d'activité se ferons à moins de 5 minutes.

4.2 Leçons

Comme leçons lors de la mise en œuvre de cette application, nous pouvons mentionner entre autre :

- L'apprentissage du Framework Laravel ;
- Le travail en utilisant les technologies comme git hub pour garder chaque version de notre application et a aussi facilité la collaboration avec surtout les autres programmeurs sur la toile ;
- Pour ce qui est du langage de programmation PHP étais l'idéal car chaque membre du groupe avais sa petite idée concernant PHP ;
- Courage d'avance même si on rencontre des obstacles
- L'utilisation des fonctionnalités du navigateur pour corriger les erreurs du code JavaScript qui ne sont pas directement affiché à l'écran mais dans la console du navigateur ;

4.3 Difficultés

Après l'analyse de cette application, nous avons eu beaucoup de problèmes comme :

- Le Framework Laravel était nouveau pour la plupart d'entre nous ce qui nous a couté en temps et en connexion internet pour apprendre ce Framework en appliquant directement sur notre application ;
- Les caractéristiques de nos machines n'étaient pas assez puissantes pour travailler avec Android Studio et qui plantait beaucoup ;
- Faible connaissant de l'équipe en ce qui concerne la programmation pour application Android
- La création de la base de données n'a pas été chose facile car très complexe ;
- Les tests concernant les modules temps réel demandais chaque fois soit de créer les activités et inscrit les étudiants dans classe et matière avant de les faire participé à l'activité soit de changer la date de la machine si on est en locale mais pour tester en ligne c'est la première solution qui sera utilisé

4.4 Perspectives envisagées

Comme perspectives envisageable dans cette application :

- ✓ Nous aimerons notifier les parties prenantes lors de la création ou modification d'une activité
- ✓ Les rapports d'activité doivent générer et envoyer automatiquement au responsable de cette activité (chef de département)
- ✓ Terminer les modules inachevés
- ✓ Pour voir créer et suivre l'activité avec l'application Android qui jusqu'à présent donne juste l'accès en salle aux étudiants.

Conclusion générale

Dans ce projet, il était question de mettre en place un système de planification et suivie d'activité ainsi que le contrôle d'accès à ses activités. Pour cela, nous avons fait une revue de la littérature sur le langage de modélisation UML, la méthode d'informatisation 2TUP. Comme outils, nous avons utilisé Laravel 5.5 comme Framework pour l'application web, PostgreSQL comme serveur de base de données, Heroku comme hébergeur, Android Studio pour l'application mobile. Ce processus et l'ensemble des outils nous ont permis de mettre en place une plateforme (application web) ayant les fonctionnalités principaux suivant : gestion des activités ; le suivie des activités en temps réel ; la visualisation des rapports d'activité ainsi que la génération du planning des activités et une application mobile (Android) pour le contrôle d'accès des étudiants en salle. Ces modules sont accessible uniquement en fonction des utilisateurs possèdent un rôle (admin, enseignant, étudiant) et qui se sont au préalable authentifié. Tout en ayant atteint les objectifs principal, nous pensons avoir ainsi réalisé le projet à 75,95%.

Bibliographie

[W-FontEnd-18] ***https://www.alticreation.com/bootstrap-foundation-frameworks-font-end***, Site Web contenant les détails sur les Framework font-end, consulté le 11 mars 2018 à 23h20.

[W-lessgbd-18] ***https://www.sgbdslam5.wordpress.com/lessgbd/***, Site Web contenant les différences entre les SGBD les plus utilisés et leurs classement en 2014, consulté le 11 mars 2018 à 00h49.

[Wikipedia-Kiwix] ***http://www.wikipedia.org/***, Kiwis (logiciel), encyclopédie en ligne et aussi en locale renferment la plus part de définitions des mots et sigle, consulté chaque jours de travail sur ce projet.

[Doc-PostgresSql-9.5.3] Documentation du logiciel PostgreSql 9.5.3, télécharger sur Internet à travers le lient suivant ***: www.doc.postgresSql.com***, également consultable dans le dossier &&/documents consulter/site web/documentation de Postgresql 9.5.3/index.html.

[W-UML-18] ***http://www.omgsysml.org/***, site web de référence en ce qui concerne UML et son évolution.

[PascRoq-UML-06] Pascal Roques « UML_2_par_la_pratique », *études d cas et exercice corrigé, 5e édition, dépôt légal août 2006, ISSN 7280.*

[p-git-tuto] Tilap « git : les commandes usuelles à connaitre », *page web enregistrer contenant le résumé sur l'installation et l'utilisation de git pour les débutants, mise à jour 20018, télécharger le 30-05-2018 à 16h55.*

[W-get_laravel5.5-18] ***https://laravel.sillo.org/cours-laravel-5-5-les-bases-installation-et-organisation/***, Site Web officiel de Laravel contenant le cours sur les bases de l'installation l'organisation d'une nouvelle application Laravel, langue de la page : française, consulté le 25 avril 2018 à 14h05.

[W-get_composer-18] ***https://getcomposer.org/doc/06-config.md***, Site Web officiel de composer contenant la documentation sur l'installation et l'utilisation de composer, langue de la page : anglaise, consulté le 25 avril 2018 à 14h05.

[W-heroku_cli-18] ***https://devcenter.heroku.com/articles/heroku-cli***, Site Web officiel de heroku contenant la documentation sur la prise en mains de heroku, consulté le 02 mai 2018 à 20h05.

[W-deploie-laravel-18] ***http://blog.occ.simplon.co/heroku/***, blog en ligne contenant le tutoriel sur le déploiement d'une application Laravel 5.5 sur heroku, langue de la page : française, consulté le 03 juin 2018 à 23h30.

Annexes

Annexe 1 : Quelques vues de l'Université de Dschang

Uds	IUTFV

Figure 28 Vue panoramique de l'Uds et de l'IUT FV de Bandjoun [W-UDs-16]

Annexe 2 : Déploiement de l'application sur heroku

1- Commençons par générer le conteneur côté heroku :
2-

Annexe 3 : Résultat d'implémentation

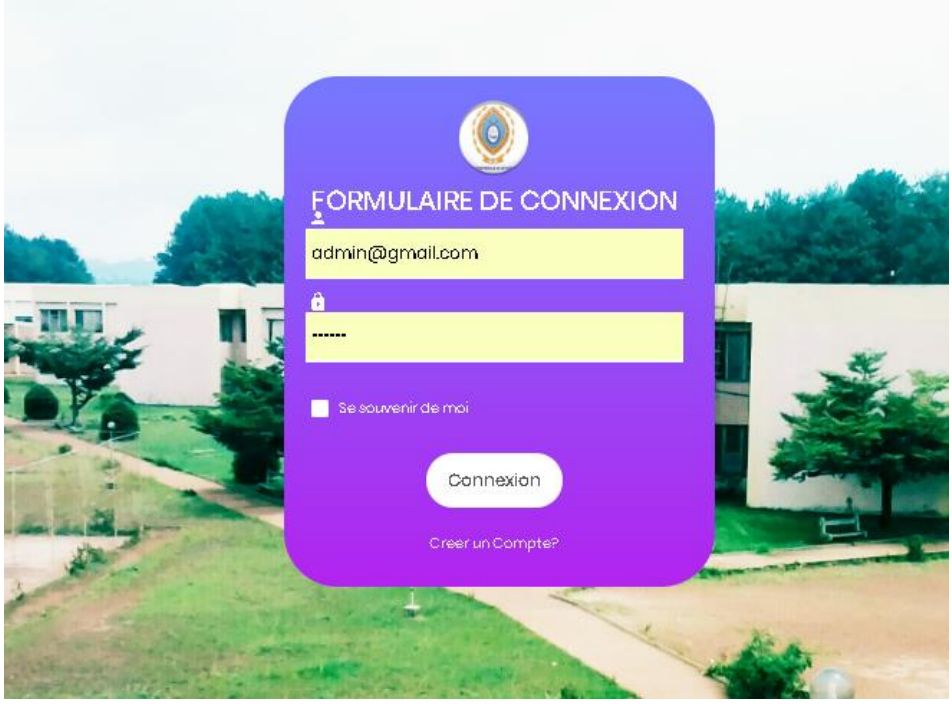


Figure 29 Formulaire de connexion sur la plate forme

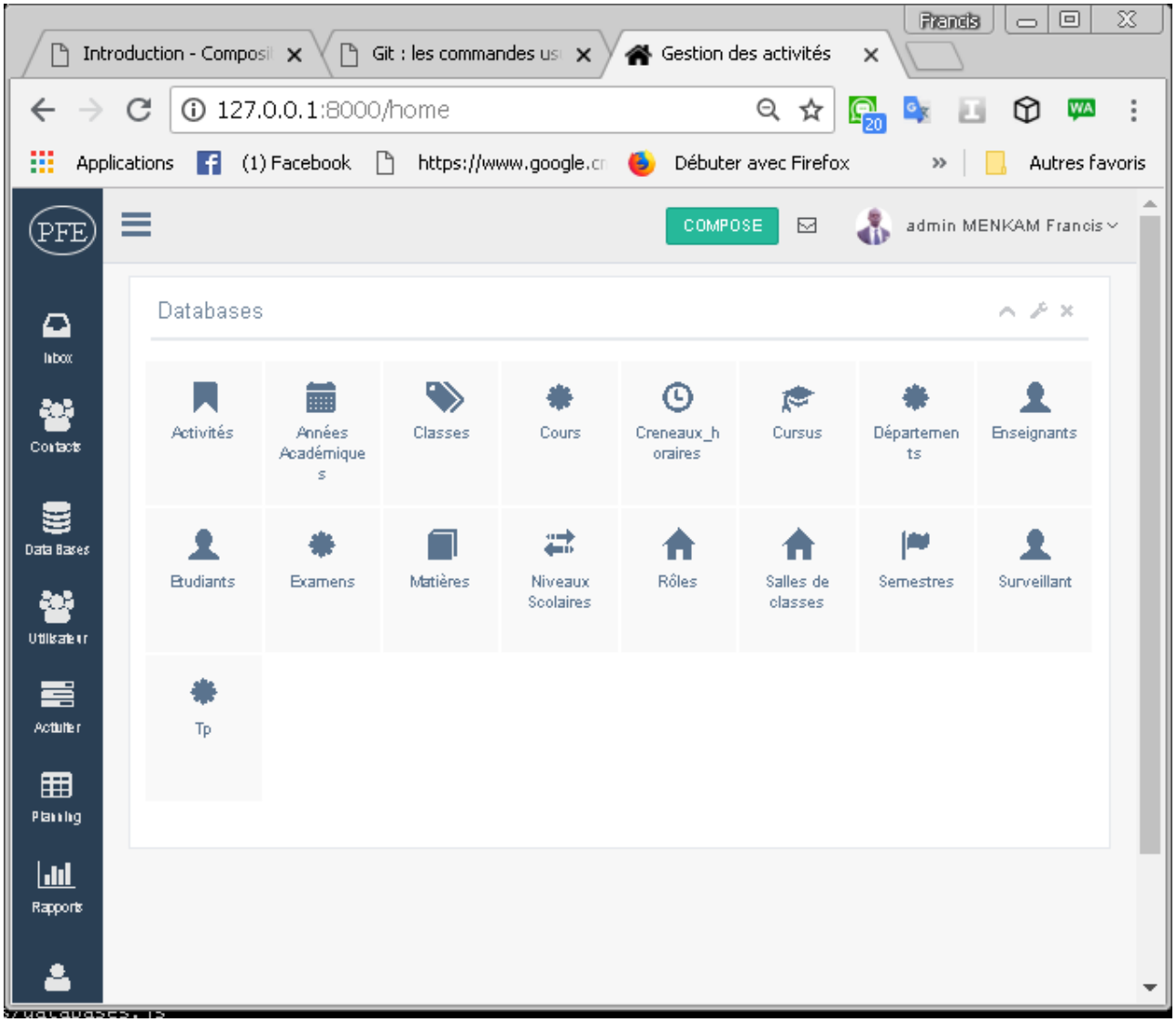


Figure 30 Interface de mise à jours de la bd par l'admin

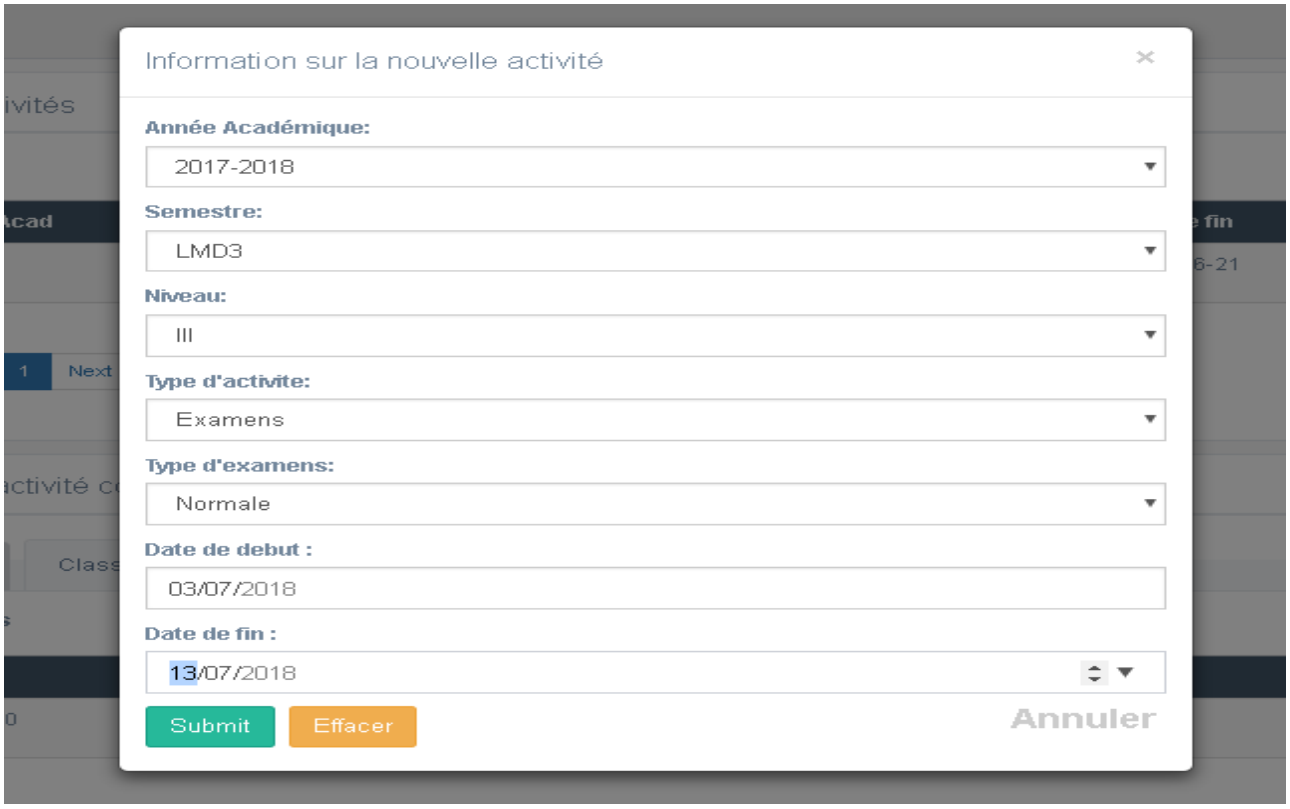


Figure 31 Formulaire de création d'une activité

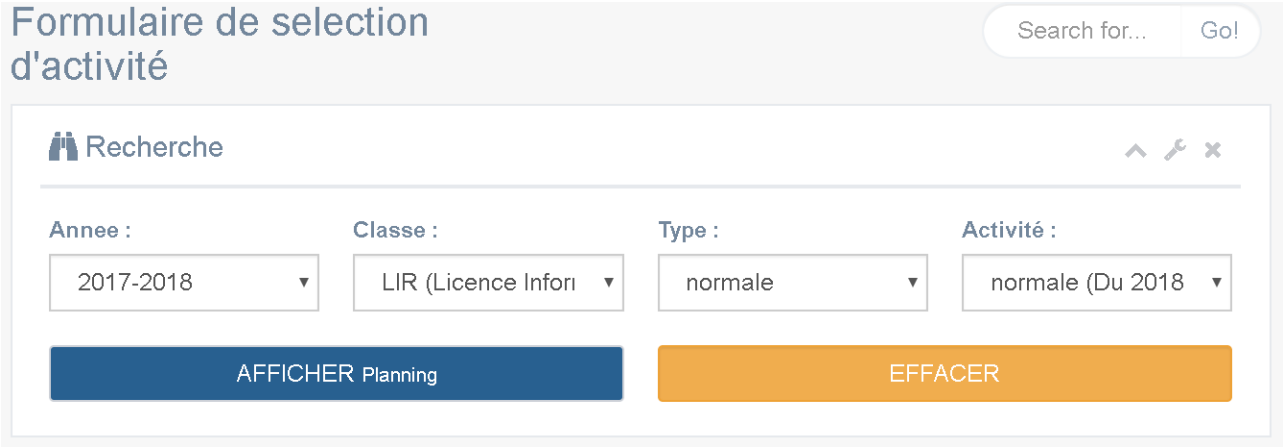


Figure 32 Formulaire pour l'affichage du planning d'une activité

```
57      public function getListePresence(Request $request)
58      {
59          $idTable = "";
60          $table2 = "";
61          $idActivite = $request->idActivite;
62          $idMatiere = $request->idMatiere;
63          $table = $request->table;
64
65          if($table=="examens"){
66              $idTable = "id_examen";
67              $table2 = "etud_compose_examens";
68          }else{
69              if($table=="cours"){
70                  $idTable = "id_cours";
71                  $table2 = "etud_etudie_cours";
72              }else{
73                  $idTable = "id_tp";
74                  $table2 = "etud_participe_tps";
75              }
76          }
77
78          return DB::select("
79              SELECT
80                etudiants.matricule_etudiant,
81                users.name,
82                users.prenom,
83                etud_ins_mats.regime
84              FROM
85                public.users,
86                public.etudiants,
87                public.etud_scolariser_clas,
88                public.etud_ins_mats
89              WHERE
90                etudiants.id_user = users.id AND
91                etud_scolariser_clas.id_etudiant = etudiants.id AND
92                etud_ins_mats.id_scolariser = etud_scolariser_clas.id AND
93                etud_ins_mats.id_matiere = '$idMatiere' AND
94                etud_ins_mats.id IN (
95                  SELECT
96                    id_etud_ins_mat
97                  FROM
98                    public.$table2,
99                    public.$table
100                 WHERE
101                   $table.id = $idTable AND
102                   id_activite = '$idActivite' AND
103                   id_matiere = '$idMatiere')
104             ORDER BY
105               users.name ASC,
106               users.prenom ASC;
107         ");
108     }
```

Figure 33 Obtenir la liste des étudiants ayant participé à une activité

menu.blade.php | RapportActiviteController.php

```
163
164     public function getListAbsent(Request $request)
165     {
166         $idTable = "";
167         $table2 = "";
168         $idActivite = $request->idActivite;
169         $idMatiere = $request->idMatiere;
170         $table = $request->table;
171
172         if($table=="examens"){
173             $idTable = "id_examen";
174             $table2 = "etud_compose_examens";
175         }else{
176             if($table=="cours"){
177                 $idTable = "id_cours";
178                 $table2 = "etud_etudie_cours";
179             }else{
180                 $idTable = "id_tp";
181                 $table2 = "etud_participe_tps";
182             }
183         }
184
185         return DB::select("
186             SELECT
187               etudiants.matricule_etudiant,
188               users.name,
189               users.prenom,
190               etud_ins_mats.regime
191             FROM
192               public.users,
193               public.etudiants,
194               public.etud_scolariser_clas,
195               public.etud_ins_mats
196             WHERE
197               etudiants.id_user = users.id AND
198               etud_scolariser_clas.id_etudiant = etudiants.id AND
199               etud_ins_mats.id_scolariser = etud_scolariser_clas.id AND
200               etud_ins_mats.id_matiere = '$idMatiere' AND
201               etud_ins_mats.id NOT IN (
202                 SELECT
203                   id_etud_ins_mat
204                 FROM
205                   public.$table2,
206                   public.$table
207                 WHERE
208                   $table.id = $idTable AND
209                   id_activite = '$idActivite' AND
210                   id_matiere = '$idMatiere')
211             ORDER BY
212               users.name ASC,
213               users.prenom ASC;
214         ");
215     }
216
```

Figure 34 Obtenir la liste des étudiants n'ayant pas participé à une activité

Table des matières

DÉDICACE I

REMERCIEMENTS II

LISTE DES ABRÉVIATIONS III

RÉSUMÉ IV

ABSTRACT V

SOMMAIRE VI

LISTE DES TABLEAUX ET FIGURES VIII

INTRODUCTION GÉNÉRALE 1

CHAPITRE 1 PRÉSENTATION DU CADRE DE TRAVAIL ET CAHIER DE CHARGES 2

1.1 CADRE DE TRAVAIL 2

1.1.1 Présentation de l'Université de Dschang (Uds) 2

1.1.2 Présentation de l'IUT FV 2

1.1.3 Présentation du Département de Génie Informatique et du Laboratoire 3

1.2 CAHIER DE CHARGES 4

1.3 PRÉSENTATION DE L'EXISTANT 5

1.4 CRITIQUES 5

1.5 PROPOSITION DE SOLUTION 6

1.6 DÉROULEMENT DU TRAVAIL ET OPÉRATIONS EFFECTUÉES 6

1.7 ETAT DE L'ART 7

1.7.1 Choix des outils utilisés 7

1.7.1.1 Langage de modélisation 7

1.7.1.2 Langages de programmation 7

1.7.1.3 SGBD 8

1.7.1.4 Framework 8

1.7.1.5 Dessin des diagrammes 8

CHAPITRE 2 ANALYSE, ALGORITHMES ET STRUCTURES DE DONNÉES 9

2.1 ANALYSE FONCTIONNELLE DU SYSTÈME 9

2.1.1 Définition modulaire du projet 9

2.1.2 Interaction entre le système et les acteurs 9

2.1.2.1 Les principaux acteurs du système 9

2.1.2.2 Cas d'utilisation pour l'étudiant 10

2.1.2.3 Cas d'utilisation pour l'enseignant (personnel) 10

2.1.2.4 Cas d'utilisation pour l'administrateur 11

2.1.2.5 Descriptions textuelle des cas d'utilisation 12

2.1.3 Modélisation des interactions entre les objets 14

2.1.3.1 Diagramme de séquence du cas d'utilisation : « crée activité » 14

2.1.3.2 Diagramme de séquence du cas d'utilisation du cas d'utilisation : « participer à une activité » 15

2.1.4 Diagramme d'activité 16

2.1.4.1 Diagramme d'activité du cas d'utilisation : « créer activité » 16

2.1.4.2 Diagramme d'activité du cas d'utilisation : « voire rapport d'activité » 16

2.1.4.3 Diagramme d'état-transition d'une activité 17

2.1.5 Présentation du modèle de navigation de l'application web 18

2.2 CONCEPTION TECHNIQUE, FONCTIONNELLE ET OBJET DU SYSTÈME 19

2.2.1 Architecture technique et fonctionnelle du système 19

2.2.2 Dictionnaire de données 20

2.2.3 Diagramme de classe 23

2.2.4 Modèle physique de données 24

CHAPITRE 3 IMPLÉMENTATION, TECHNIQUES DE PROGRAMMATION ET RÉSULTATS 25

3.1 TECHNIQUE DE DÉVELOPPEMENT 25

3.1.1 Usage du pattern de développement MVC (Model Vue Controller) 25

3.1.2 Arborescence du système 25

3.2 DÉPLOIEMENT DE L'APPLICATION 26

3.2.1 Environnements et outils de déploiement 26

3.2.2 Déploiement en local 28

3.2.3 Déploiement en ligne 28

3.3 RÉSULTATS CONCRETS DU DÉVELOPPEMENT 29

3.3.1 Premier résultat : Interface de gestion d'activité 29

3.3.1.1 Présentation 29

3.3.1.2 Algorithme d'obtention du resultat 29

3.3.2 Deuxième résultat : Suivie en temps réel du déroulement d'une activité 30

3.3.2.1 Présentation 30

3.3.2.2 Algorithme d'obtention du résultat 31

3.3.2.3 Application du résultat 31

3.3.3 Troisième résultat : Générer le planning d'une activité 32

3.3.3.1 Présentation 32

3.3.3.2 Algorithme d'obtention du résultat 32

3.3.3.3 Application du résultat 33

3.3.4 Quatrième résultat : Générer le rapport d'une activité 33

3.3.4.1 Présentation 33

3.3.4.2 Algorithme d'obtention du résultat 33

3.3.4.3 Application du résultat 33

CHAPITRE 4 APPORT, LEÇONS, DIFFICULTÉS ET PERSPECTIVES 34

4.1 APPORTS 34

4.2 LEÇONS 34

4.3 DIFFICULTÉS 34

4.4 PERSPECTIVES ENVISAGÉES 35

CONCLUSION GÉNÉRALE 36

BIBLIOGRAPHIE 37

ANNEXES ..... A

ANNEXE 1 : QUELQUES VUES DE L'UNIVERSITÉ DE DSCHANG ..... A

ANNEXE 2 : DÉPLOIEMENT DE L'APPLICATION SUR HEROKU ..... A

ANNEXE 3 : RÉSULTAT D'IMPLÉMENTATION ..... B

TABLE DES MATIÈRES ..... A