



République Algérienne Démocratique et Populaire



Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Akli Mohand Oulhadj de Bouira

Faculté des Sciences et des Sciences Appliquées

Département d'Informatique

Mémoire de fin d'études

Pour l'obtention du diplôme de Master en
Informatique

Option : ISIL

Thème

Conception et réalisation d'une application de
gestion des emplois du temps au niveau de
département informatique

Encadré par

— DMOUCHE MOULOU

Réalisé par

— HOCINE KAHINA

2018/2019

Résumé

Ce travail s'intéresse à la conception et la réalisation d'un système de gestion des emplois du temps pour département informatique . L'objectif de cette application est de faciliter l'élaboration des emplois du temps au niveau des départements, de permettre la création et la modification des informations des emplois du temps.

Mots clés : application, Emploi du temps, département, ...

Abstract

This work focuses on the design and implementation of a management system of a timetable for the department. The purpose of this application is to facilitate the development of time schedules at the departmental level, allow the creation and modification of information of schedules.

Key words : application, timetable,department ...

Table des matières

Table des matières	i
Table des figures	iv
Liste des tableaux	vi
Liste des abréviations	vii
Introduction générale	1
1 Etude de l'existant	3
1.1 Introduction	3
1.2 Etude Préalable	3
1.2.1 Présentation de l'organisme d'accueil[1]	3
1.2.2 Cadre de l'étude	6
1.3 Problématique	7
1.3.1 Définition d'une planification [2] :	7
1.3.2 Définition d'un planning	7
1.3.3 Le rôle d'un emploi du temps	8
1.4 Analyse de l'existant	8
1.4.1 La pédagogie	8
1.4.2 Etude du système pédagogique	8
1.4.3 Description du problème à résoudre	11
1.5 Conclusion	14

2	Analyse et conception	15
2.1	Introduction	15
2.2	Processus de développement logiciel	15
2.2.1	Processus unifié	16
2.2.2	Modélisation avec UML :	18
2.3	Modélisation fonctionnelle de notre système :	20
2.3.1	Description des cas d'utilisations :	20
2.3.2	Description textuelle des cas d'utilisation :	20
2.4	Diagramme de cas d'utilisation de l'application :	22
2.4.1	Réalisation des diagrammes de séquences :	23
2.4.2	Diagramme de séquence du cas d'utilisation : Gestion des enseignants	24
2.4.3	Réalisation des diagrammes d'activité :	27
2.5	Modélisation structurelle de notre application	30
2.6	Modèle Relationnel de données :	33
2.7	Conclusion	34
3	Conception et implémentation	35
3.1	Introduction	35
3.2	Outils et langages de développements	35
3.2.1	XAMPP	35
3.2.2	phpMyAdmin	36
3.2.3	Langage de codage (JAVA Netbeans)	37
3.3	Implémentation de la base de données	37
3.4	Description de l'application	38
3.4.1	Fenêtre d'authentification	38
3.4.2	Menu principale	39
3.4.3	Gestion des enseignants	40
3.4.4	Gestion des modules	41
3.4.5	Gestion des locaux	41
3.4.6	Génération d'emploi de temps	42
3.5	Conclusion	43
	Conclusion générale	44

Bibliographie

45

Table des figures

1.1	Organigramme du département d'informatique	6
2.1	Les phases du processus unifié [10]	17
2.2	Les types des diagrammes UML2[10]	20
2.3	Diagramme de cas d'utilisation pour la gestion des emplois du temps.	23
2.4	Diagramme de séquence du cas d'utilisation : Authentification.	24
2.5	Diagramme de séquence du cas d'utilisation : gestion des enseignants.	25
2.6	Diagramme de séquence du cas d'utilisation : gestion des locaux.	26
2.7	Diagramme de séquence du cas d'utilisation : afficher emplois du temps par section.	27
2.8	Diagramme d'activité : Authentification.	28
2.9	Diagramme d'activité : Ajouter un enseignant.	28
2.10	Diagramme d'activité : Modifier un enseignant.	29
2.11	Diagramme d'activité : Supprimer un enseignant.	29
2.12	Dictionnaire de données épuré	30
2.13	Représentation graphique d'une classe.	31
2.14	Diagramme de classes de l'application.	32
3.1	Exemple de XAMPP	36
3.2	Base de données de notre application	38
3.3	Interface pour accéder à l'application	38
3.4	Problème d'authentification	39
3.5	Menu de l'application	39
3.6	Gestion des enseignants	40

3.7	Gestion des modules	41
3.8	Gestion des locaux	41
3.9	Génération d'emploi de temps	42
3.10	les informations de seance	43

Liste des tableaux

2.1	Description du cas d'utilisation : Authentification	21
2.2	Description du cas d'utilisation : gestion des enseignants	22

Liste des abréviations

LMD	L icence M aster D octorat
EDT	E mploi D e T emps
TD	T ravaux D irigés
TP	T ravaux P ratiques
HMin	H eure M inimale
HMax	H eure M aximale
DMin	D urée M inimale
S	S eance
R	R essource
RF	R essource F ille
UML	U nified M odelling L anguage
alt	a lternative
opt	o ptional
PHP	P ersonal H ome P age
SQL	S tructural Q uery L anguage
XAMPP	cross-plateforme A pache M ySQL P HP P erl
LAMPP	L inux A pache M ySQL P HP P erl
WAMPP	W indows A pache M ySQL P HP P erl

Introduction générale

. Au cours de notre vie professionnelle, on se trouve confrontés au problème de la planification d'horaire de travail. Dans plusieurs domaines tel que les usines, des pièces doivent être cheminées à travers plusieurs machines, la gestion de ce trafic doit répondre à certaines contraintes, tel que la maximisation de la production ou l'exploitation optimale des machines. Dans les hôpitaux les infirmiers et les médecins doivent être attribués aux postes de travail de manière à obéir à certaines règles de gestion des hôpitaux. Et cela nécessite une élaboration périodique et stratégique de plannings de travail du personnel et les raisons de cette élaboration sont variées.

L'élaboration d'un planning ou Emploi du temps, revient à résoudre un problème complexe, chaque institution (entreprises de production, hôpitaux, universités,... etc.) ayant ses propres règles et critères.

Le département d'informatique de l'université Akli Mohand oulhadj-bouira comme tous autres départements gère les emplois du temps au début de chaque semestre de chaque année.

La gestion d'emplois du temps, est une tâche d'une grande importance car, en effet elle permet de gagner beaucoup d'heures de travail pour le département, pour les enseignants et pour les étudiants, et de fournir des solutions optimales avec satisfaction d'un maximum de contraintes.

Pour cela, nous avons décidé de développer une application de gestion des emplois

du temps pour le département informatique de l'université de Bouira. En accompagnant tous les acteurs impliqués dans ce processus.

Notre rapport est composé de trois chapitres. Son organisation est comme suit :

Dans le premier chapitre, nous présentons généralement notre organisme d'accueil qui est le département d'informatique du Bouira, ses services et leur organigramme, mais avant ça dans un premier temps nous faisons un aperçu général sur l'Université Akli Mohand Oulhadj et sur Faculté des Sciences et des Sciences Appliquée qui contient notre organisme d'accueil. Dans un second temps, nous donnons une vision globale sur notre problématique et la pédagogie ainsi qu'on faisons une description du problème à résoudre .En fin , nous définons les contraintes nécessaire d'emplois du temps.

Dans le deuxième chapitre ,nous faisons la conception et la modélisation du processus de gestion des emplois du temps au niveau de départements d'informatique, en utilisant UML comme langage de modélisation et le processus unifié qu'on a suivi durant tout le processus de développement.

Dans le troisième chapitre consacré à la réalisation de notre application, dans un premier lieu nous donnons les outils de développement utilisés, tel que le langage java, MySQL et NetBeans. Dans un deuxième lieu, nous présentons quelques interfaces de notre application avec des fonctionnalités.

Enfin,nous conclurons notre travail avec une conclusion générale résumant les grands points qui ont été abordés ainsi que les perspectives que nous souhaitons accomplir dans la future.

Chapitre 1

Etude de l'existant

1.1 Introduction

L'analyse de l'existant est une étape importante dans le cycle de vie d'un système, il s'agit de connaître la situation actuelle de l'organisation pour pouvoir porter un jugement juste. Ainsi, l'analyse du système existant doit nous fournir toute l'information nécessaire, afin d'établir une bonne conception et de proposer de bonnes solutions.

Dans ce chapitre, nous allons présenter une étude générale concerné par notre étude, Dans un premier temps nous présenterons l'Université Akli Mohand Oulhadj de Bouira puis la Faculté des Sciences et des Sciences Appliquée puis département informatique. Par la suite, la problématique de la planification des horaires ainsi que le cadre d'étude. Après nous allons faire une description du problème à résoudre (Suivi pédagogique des enseignants et les contraintes).

1.2 Etude Préalable

1.2.1 Présentation de l'organisme d'accueil[1]

1) Présentation d l'Université Akli Mohand Oulhadj de Bouira :

L'Université Akli Mohand Oulhadj Bouira est nommé par le décret exécutif n12-241 du 14 Rajab 1433 correspondant au 04 Juin 2012. C'est un établissement public à caractère scientifique, culturel et professionnel doté de personnalité juridique et d'autonomie financière. Suivant le décret exécutif mentionné ci-dessus, l'université a été restructurée

par la constitution de (06) six facultés et d'Instituts présentés respectivement :

1. Faculté des sciences et de la technologie.
2. Faculté des sciences de la nature et de la vie et sciences de la terre
3. Faculté des Lettres et des Langues.
4. Faculté des sciences sociales et humaines.
5. Faculté des sciences économiques, commerciales et des sciences de gestion.
6. Faculté de droit et des sciences politiques.
7. Institut des sciences et techniques des activités physiques et sportives.

En outre de quatre sous-directions au niveau du rectorat del'université responsables de :

1. La formation supérieure des premiers et deuxièmes cycles, la formation continue et les diplômes, et la formation supérieur de graduation.
2. La formation supérieure de troisième cycles, l'habilitation universitaire, la recherche scientifique, et la formation supérieure de post-graduation.
3. Les relations extérieures, la coopération, l'animation, la communication et les manifestations scientifiques.
4. Le développement, la prospective et l'orientation

2)Présentation de Faculté des Sciences et des Sciences Appliquée

La Faculté des Sciences et des Sciences Appliquées, est une faculté jeune, créée par le Décret exécutif n 12-241 du 14 Rajab 1433 correspondant au 4 juin 2012 portant création de l'Université Akli Mohand Oulhadj de Bouira sous le nom Faculté des Sciences et de la Technologie. Son nom est modifié à Faculté des Sciences et des Sciences Appliquées par le décret exécutif n13-179 du 24 Jomada Ethania 1434 correspondant au 5 Mai 2013 modifiant et complétant le précédent texte.

La faculté des sciences et sciences appliquées est située au pôle universitaire, côté sud de la ville de Bouira.

La faculté des sciences comporte les huit départements suivants :

- Département Physique
- Département de Chimie
- Département de Mathématiques
- Département d'Informatique

- Département de Génie Civil
- Département de Génie Mécanique
- Département de Génie Electrique
- Département de Génie des Procédés

Elle est dotée de six blocs suivants :

- Bloc administratif (Doyenné)
- Bloc pédagogique
- Bloc des amphis
- Bibliothèque
- Hall des technologies
- Bloc des laboratoires

3)Présentation de département informatique :

Le département d'informatique est l'un des départements géré par la faculté des sciences sous la direction du Doyen, et par un chef de département, assisté par d'autres responsables affectés respectivement à leur propres services.

On trouve au sein de ce département deux services :

- a. **Le service d'enseignement et de la scolarité** sous la direction d'un chef adjoint attaché à la pédagogie. Ce service assure la gestion des études et toutes les tâches concernant les enseignants et les étudiants du département. Les enseignants sont classés en fonction de leur grade. On distingue :Les professeurs, les maîtres de conférences, les maîtres assistants et le jury de délibération, le promoteur et le conseil pédagogique
- b. **Le service de la recherche scientifique** sous la direction d'un chef de département adjoint attaché à la recherche scientifique. Il gère les enseignants chercheurs et les étudiants en post graduation.

Les enseignants chercheurs organisent des réunions de travail pour exposer et étudier des thèmes de recherche, suivent les nouvelles technologies et animent des conférences pour l'information des étudiants.

Les activités reliées au suivi pédagogique dans le département de l'informatique sont :

- Convoquer les enseignants.

- Etablir l'emploi du temps.
- Organiser les différents comités ou conseils.
- Organiser les examens.
- Organiser les soutenances.

Organigramme du département d'informatique

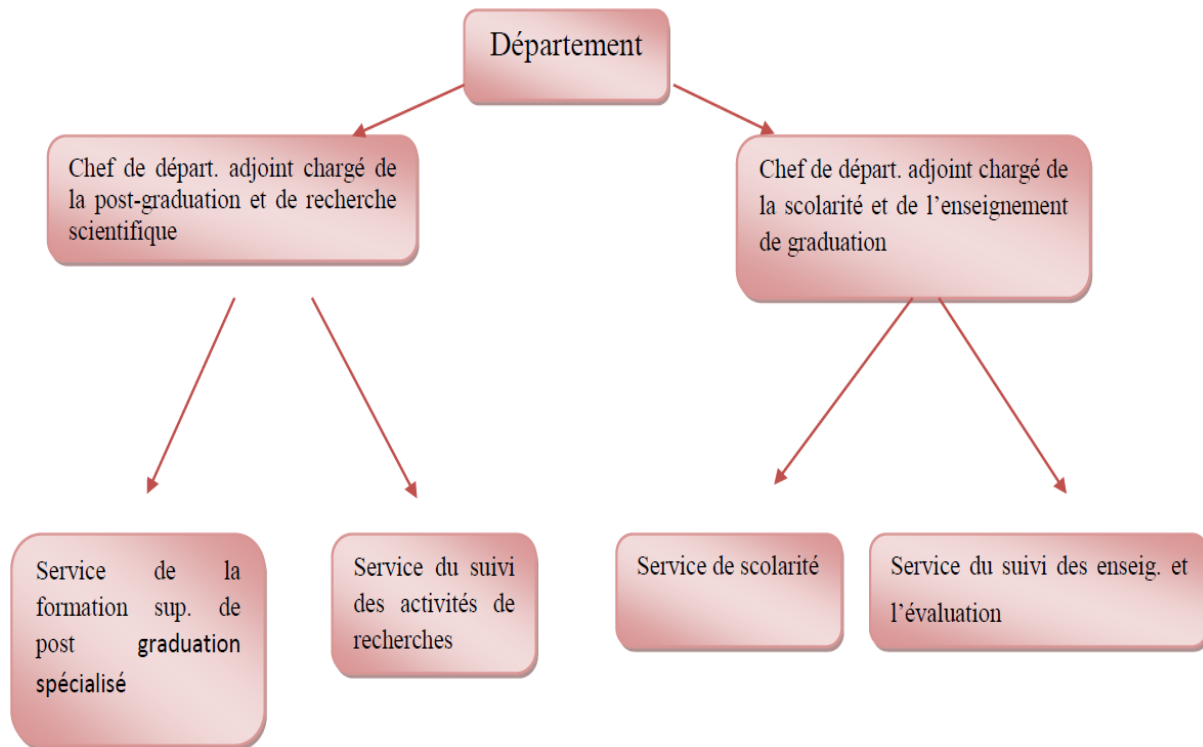


FIGURE 1.1 – Organigramme du département d'informatique

1.2.2 Cadre de l'étude

Chaque année, les responsables pédagogiques de la direction des études ont pour mission de concevoir les emplois du temps des différentes filières en essayant, au mieux, de satisfaire les contraintes « humaines » des enseignants et des étudiants, les contraintes imposées par la progression des enseignements et qui sont liées aux ressources matérielles (les salles, les équipements,..etc).

Ainsi, l'idée de mettre en place un système de gestion des emplois du temps est née, dont l'objectif est :

- La génération automatique des emplois du temps.
- Consultation et suivi des emplois du temps.

Par conséquent, notre étude concerne la direction des études et en particulier le service du suivi des enseignements et de l'évaluation.

1.3 Problématique

La problématique de notre département est la planification des emplois du temps. Pour chaque niveau d'étude, les étudiants sont rassemblés en sections et chaque section en groupes. Ces derniers ont un planning hebdomadaire, ou chaque module est assuré sous forme de séances de cours, TP ou TD dans des locaux répartis au début de chaque semestre.

1.3.1 Définition d'une planification [2] :

La planification est un instrument de gestion dont l'objectif est d'aboutir à des programmes permettant d'organiser et planifier le travail des salariés afin de rester pérenne dans l'économie globale. Ceci passe par la détermination des capacités de tout un chacun et par le recensement des activités futures et des besoins en personnel.

La planification vise à affecter les ressources humaines pour chaque intervalle de temps sur un horizon donné, de telle manière que les besoins par intervalle soient couverts et que les différentes contraintes soient satisfaites

1.3.2 Définition d'un planning

Les plannings sont des calendriers de travail, où figurent à la fois le temps, l'affectation du personnel, les jours et les horaires de travail, et les congés et repos[3]. En effet, ils apparaissent dans les cas suivants :

- Si le travail doit être assuré pendant plus d'une journée, il faut prévoir la succession de plusieurs personnes sur le même poste dans la journée. Un outil d'aide est nécessaire lorsque le nombre de postes dépasse la quinzaine, par exemple gérer les absences imprévues des salariés.
- Si le travail doit être assuré pendant plus de 35 heures par semaine, un outil automatique devient indispensable lorsque le nombre de postes dépasse la trentaine

pour gérer la succession de plusieurs personnes dans la semaine, ainsi que les absences imprévues. [4]

1.3.3 Le rôle d'un emploi du temps

Un Emploi du Temps est le dispositif qui gère de nombreux établissements ressources humaines dans la société. Pour cela, il faut avoir un système robuste qui soit face aux différentes difficultés, parmi ces problèmes :

1. L'occupation des salles.
2. La gestion des salles.
3. La gestion des séances.
4. La gestion des enseignants.
5. L'indisponibilité des enseignants.
6. La sollicitation d'un même enseignant pour diverses promos.
7. La surcharge pédagogique (un enseignant ne peut et doit pas enseigner plus de deux cours en matinée ou soirée).

1.4 Analyse de l'existant

1.4.1 La pédagogie

Pédagogie, théorie de l'enseignement, qui s'est imposée à partir du XIXe siècle comme science de l'éducation, ou didactique expérimentale, et s'interroge aujourd'hui sur les conditions de réception du savoir, sur le contenu et l'évaluation de celui-ci, sur le rôle de l'éducateur et de l'étudiant dans le processus éducatif et, plus globalement, sur les finalités de cet apprentissage, indissociable d'une norme sociale et culturelle[5]

La pédagogie est l'acte d'enseigner accompagné du discours qui l'accompagne. C'est ce qu'il faut savoir, et les compétences dont on a besoin pour maîtriser et justifier les nombreuses différences types de décisions constituant l'enseignement.[6]

1.4.2 Etude du système pédagogique

Il s'avère que ce système ne possède pas de particularité pédagogique ciblée et possède de nombreuses similarités avec le fonctionnement d'autres universités. Ainsi, l'Université

Akli Mohand Oulhadj est une université qui regroupe différentes formations du LMD. Les étudiants s'inscrivent en début de chaque année universitaire, c'est-à-dire généralement au début de septembre. Mais cela peut varier d'une formation à l'autre. Le programme pédagogique de chaque formation est connu. Ce programme précise les matières à suivre, leurs volumes horaires et quelques informations pédagogiques (répartition en cours, TD et TP). Selon les besoins pédagogiques et les conditions physiques des ressources, chaque formation est structurée en sections qui peuvent eux-mêmes être scindés en groupes .

Administrativement, les enseignants doivent assurer un nombre minimal d'heures qui est défini dans leur statut. Selon le type d'enseignement, on pondère le nombre d'heures par un coefficient. Ainsi, par exemple, une heure de cours est comptabilisée comme 1,5 heure de travaux dirigés. Lorsqu'un enseignant accepte la responsabilité d'un enseignement, il est tenu d'en respecter le volume horaire prévu par le responsable pédagogique. En cas d'absence, un enseignant doit prévoir des séances de rattrapage. Il doit donc connaître précisément la disponibilité des ressources de sa séance. Cette organisation garantit que tous les étudiants qui suivent une même formation auront eu le même volume horaire d'enseignement. Chaque formation est gérée par son responsable pédagogique qui crée les emplois du temps en concertation avec les enseignants mais aussi, moins souvent, avec les responsables des autres formations. Les enseignants peuvent assurer leurs enseignements dans différentes formations. Lors de la création des emplois du temps, les responsables pédagogiques ne tiennent généralement pas compte explicitement des salles disponibles. Dans la réalité, les responsables pédagogiques s'appuient sur l'expérience des années précédentes pour planifier les enseignements.

a) **L'activité pédagogique**

L'activité pédagogique est modélisée à l'aide de trois entités : les matières, les enseignements et les modules. D'une manière générale, les enseignants, les groupes, les matériels constituent les ressources de l'enseignement. Les matériels sont les ressources qui seront attribués aux séances de l'enseignement. Un enseignement peut être assuré simultanément par plusieurs enseignants. C'est le cas, par exemple de certains TP d'informatique. Les modules sont des ensembles d'enseignements.

b) Les ressources

Les ressources considérées sont les entités physiques nécessaires à l'élaboration des emplois du temps. Il s'agit des salles, des enseignants, des groupes, des étudiants et des matériels. Les ressources sont caractérisées par des données abstraites et des données spécifiques. Les données abstraites caractérisent chaque ressource et sont constituées d'un code, qui permet de la différencier des autres ressources, son calendrier qui précise quels sont les jours de disponibilité et d'indisponibilité et sa description. En plus de ces données abstraites, chaque ressource possède des caractéristiques spécifiques qui dépendent du type de la ressource. L'intérêt de distinguer ces deux types de caractéristiques est que l'outil peut très facilement évoluer pour prendre en compte de nouveaux types de ressources. Dans la suite nous décrivons les caractéristiques spécifiques des ressources considérées dans l'étude.

— *Les ressources de type « salle »*

Une salle est un lieu dans lequel sont assurés des enseignements. Le type d'une salle est une indication sur le type d'enseignement qu'on peut y faire.

— *Les ressources de type « enseignant »*

L'enseignant désigne une personne pouvant assurer des enseignements. Chaque enseignant est caractérisé par : son nom et son prénom, son grade, sa filière, sa spécialité. La spécialité renseigne sur les matières que peut enseigner un enseignant.

— *Les ressources de type « groupe »*

Un groupe est un ensemble d'étudiants. Il se compose d'autres groupes qui peuvent être réduits à un seul étudiant (un binôme ou monôme). Cette modélisation ne fait pas de distinction entre groupe et étudiant. En général les étudiants d'une filière sont décomposés en section et chaque section est décomposée en plusieurs groupes.

— *Les ressources de type « matériel »*

Pour qu'un enseignement puisse se dérouler, l'enseignant utilise un matériel pédagogique. Généralement ce matériel est limité à un tableau, des craies, éventuellement un rétroprojecteur et parfois un vidéo projecteur. Pour des enseignements particuliers, l'enseignant peut avoir besoin de matériel spécifique.

c) La notion du temps

- Pour modéliser le temps, les entités : date, heure, durée, créneau et calendrier sont définies. Une date désigne un instant défini par un triplet (jour, mois, année). A partir de ce triplet, on détermine la valeur qui lui est associée sur l'axe des jours. Pour avoir un grain plus fin, la notion d'heure est utilisée. Il s'agit d'un nombre entier compris entre la valeur minimale HMin et la valeur maximale HMax. Ces deux nombres correspondent à des heures par rapport à une date. Par exemple, dans notre cas, nous avons choisi HMin=8h et HMax=16h00 par défaut. Une durée est un nombre compris entre DMin et Dmax = HMax - HMin. DMin représente la plus petite unité temporelle disponible.
- Une séance correspond à une instance temporelle d'un enseignement à une date donnée, pendant un créneau précis. L'ensemble des séances d'une ressource est appelé planning de cette ressource. Il s'agit de l'ensemble des séances dans lesquelles apparaît-la ressource. Ainsi, il est possible de considérer le planning d'une salle au même titre que celui d'un enseignant. La notion de planning apparaît également au niveau des enseignements : cela permet ainsi de manipuler l'évolution dans le temps d'un enseignement particulier en vue.

d) Les filières

La notion de filière permet de limiter le nombre des ressources à considérer. En effet, le responsable d'une filière n'a besoin de connaître que le planning des ressources qu'il utilise. Cependant, certains conflits ne peuvent être détectés que lorsque les différentes filières sont rassemblées dans une base de données commune. Il s'agit surtout des conflits d'enseignants et de salles puisqu'il est fréquent qu'un enseignant intervienne dans plusieurs filières ou qu'une salle soit partagée par plusieurs filières.

1.4.3 Description du problème à résoudre

Dans un établissement éducatif, un ensemble d'étudiants groupés sous une structure hiérarchique (filières, promotions, sections, groupes,...) sont censés avoir un ensemble d'enseignements qui se répètent périodiquement, chacun de ces enseignements s'étend sur une durée de temps, dont l'unité élémentaire est la période.

Résoudre le problème de l'emploi du temps revient à affecter à chacun de ces ensei-

gnements un nombre de périodes consécutives égal à la durée qu'il exige, un local dont le type et la capacité sont convenables, et un enseignant apte à assurer le module concerné par l'enseignement de façon à prévenir les conflits sur les enseignants, sur les étudiants et sur les locaux.

Dans notre cas, le problème de l'emploi du temps étudié est celui de département informatique où les responsables pédagogiques ont besoin chaque année d'établir une nouvelle planification des différentes promotions en essayant au mieux de satisfaire les contraintes " humaines " des enseignants et des étudiants, les contraintes pédagogiques imposées par la progression des enseignements et en tenant compte des contraintes " physiques " liées aux ressources matérielles (les locaux).

Département informatique regroupe différentes formations qui ont une durée qui varie entre trois ans (licence) et cinq ans (master).

Le programme pédagogique d'emplois du temps de chaque formation est connu à priori. Ce programme précise les modules à suivre, leurs volumes horaires et quelques informations pédagogiques (répartition en cours, travaux dirigés, travaux pratiques etc...).

Selon les besoins pédagogiques et les conditions physiques des ressources, chaque formation est structurée en promotions, en sections, et en groupes. En résumé, les données du problème à résoudre sont constituées par :

- Un ensemble de créneaux horaires étalés sur une semaine de six jours, du samedi au jeudi avec un nombre six périodes. La durée d'une période est d'une heure trente minutes pour les emplois du temps
- Un ensemble de promotions ou groupes d'étudiants.
- Un ensemble de cours, TD ou TP à programmer dans la semaine.
- Un ensemble de locaux (salles, amphis).

Suivi pédagogique des enseignants

Un enseignant est chargé d'assurer des séances de cours magistral, de TD ou de TP repartis au cours du jour. Une journée pédagogique est composé de six séances d'une heure et demie chacune.

On rencontre des enseignants permanents, associés ou vacataires. Chaque enseignant possède son propre grade de professeur, maitre de conférences, maitre-assistant, ou enfin professeur-ingénieur.

Le volume horaire effectif d'un enseignant doit être au-delà de neuf heures par semaine. La charge pédagogique d'un enseignant s'étale sur trois jours au maximum dans la semaine. Pour créer un emploi du temps, on répartit les séances dans la semaine selon la situation de l'enseignant et l'occupation des salles ou d'autres contraintes[7].

Les contraintes

L'analyse sur le terrain montre que les données gérées doivent vérifier certaines contraintes pour garantir leur cohérence. De manière abstraite, les contraintes à respecter peuvent être classées en deux groupes : les contraintes physiques et les contraintes pédagogiques.

• Les contraintes physiques

Ces contraintes ne doivent pas être violées sinon cela conduirait à des situations conflictuelles. On dira qu'il y a un « conflit physique de ressource » entre deux séances s_1 et s_2 si ces deux séances ont une ressource en commun pendant une durée non nulle. Voici les contraintes physiques :

1. une ressource ne peut pas être occupée en même temps dans deux séances différentes ;
2. dès qu'une ressource R est occupée par une séance S , toutes ses ressources filles RF sont également occupées par la même séance et cela récursivement (il s'agit d'une contrainte liée à la hiérarchisation) ;
3. on ne peut pas mettre plus d'étudiants qu'il n'y a de places dans une salle
4. le volume horaire total des séances d'un enseignement ne peut pas dépasser le volume prévu ;
5. les calendriers sont respectés pour toutes les ressources.

• Les contraintes pédagogiques

Des contraintes pédagogiques ont pu être identifiées. Elles se différencient des contraintes physiques par le fait qu'elles peuvent éventuellement être violées. Typiquement ces contraintes sont utilisées pour exprimer ce que doit être un " bon " planning. Voici quelques exemples de ces contraintes :

- Un enseignant ne peut pas assurer deux enseignements en même temps.
- Pendant un créneau donné, un local ne peut pas être utilisé que pour un seul enseignement.

- Si un TD ou un TP est affecté à un groupe donné, on ne peut pas affecter un cours à la section à laquelle appartient ce groupe et vice versa.
- Un TP n'aura pas lieu dans un amphithéâtre ou dans une salle TD.
- Un TD n'aura pas lieu dans un amphithéâtre ou dans une salle TP.
- Un cours n'aura pas lieu dans une salle TP.

1.5 Conclusion

Durant l'analyse de l'existant nous avons recensé presque toutes les informations nécessaires et indispensables à notre projet à savoir la conception et la réalisation d'une application de gestion des emplois du temps au niveau de département informatique.

Ces informations tirées entre autre à partir de l'étude du système pédagogique nous aident énormément à entamer notre travail concernant le chapitre suivant sur l'analyse et la conception de notre application en utilisant le langage UML (Unified Modeling Language), et on va présenter presque tous les diagrammes nécessaires.

Analyse et conception

2.1 Introduction

Dans les projets de gestion d'emploi du temps, l'optimisation de solution est un réel problème du fait que la recherche d'une solution est complexe car il s'agit d'un problème de résolution de contraintes.

L'objectif principal de notre étude est de mettre en œuvre un logiciel qui rendra le processus de génération des emplois du temps plus efficace en termes d'implication d'utilisateurs, et qui est capable d'agir sur les changements des contraintes.

Dans ce chapitre nous avons fait la modélisation de l'application par création des diagrammes UML selon un ordre définie par le processus unifié. Au premier lieu, on vas présenter les processus de développement logiciel , puis la modélisation fonctionnelle de notre système.

2.2 Processus de développement logiciel

Un processus définit une séquence d'étapes, en parties ordonnées, qui concourent à l'obtention d'un système logiciel ou à l'évolution d'un système existant. Nous avons utilisé le processus unifié (UP) qui est présenté par la suite :

2.2.1 Processus unifié

A) Définition

Le Processus Unifié (PU ou UP en anglais pour Unified Processus) est un processus de développement logiciel, Ce dernier est une trame commune des meilleures pratiques de développement, il regroupe les activités à mener pour transformer les besoins d'un utilisateur en système logiciel.[8]

B) Caractéristiques du processus unifié

Tout processus UP répond aux caractéristiques ci-après[9]

- a) **itératif et incrémentale**, au sens que la réalisation du produit se fait en plusieurs itérations où chaque itération aboutit à livrer une partie de produit (un module) et l'ajoute aux différentes parties déjà construit (incrémentation) en gardant un produit homogène. C'est la meilleure pratique de gestion des risques d'ordre à la fois technique et fonctionnelle.
- b) **Centré sur l'architecture** Tout système complexe doit être décomposé en parties modulaires afin de garantir une maintenance et une évolution facilitée. Cette architecture (fonctionnelle, logique, matérielle, etc.) doit être modélisée en UML et pas seulement documentée en texte.
- c) **Piloté par les risques** Les risques majeurs du projet doivent être identifiés au plus tôt, mais surtout levés le plus rapidement possible. Les mesures à prendre dans ce cadre déterminent l'ordre des itérations.
- d) **Conduit par les cas d'utilisations** Le projet est mené en tenant compte des besoins et des exigences des utilisateurs. Les cas d'utilisation du futur système sont identifiés, décrits avec précision et priorité

C) Différentes phases du processus unifié

La gestion d'un tel processus est organisée suivant les quatre phases [9] :

1. **Phase d'initialisation** : Elle conduit à définir la vision du projet, sa portée et sa faisabilité afin de pouvoir décider au mieux de sa poursuite ou de son arrêt.
2. **Phase d'élaboration** : Elle poursuit trois objectifs principaux en parallèle :
 - Identifier et décrire la majeure partie des besoins des utilisateurs.

- Construire l'architecture de base du système. - Lever les risques majeurs du projet.
3. **Phase de construction** :Elle consiste surtout à concevoir et implémenter l'ensemble des éléments opérationnels (autres que ceux de l'architecture de base). C'est la phase la plus consommatrice en ressources et en efforts.
 4. **Phase de transition** :Permet de faire passer le système informatique des mains de développeurs à celles des utilisateurs finaux. Les mots-clés sont : conversion des données, formation des utilisateurs, déploiement et bêta-tests

La figure 2.1 montre les phases du processus unifié :

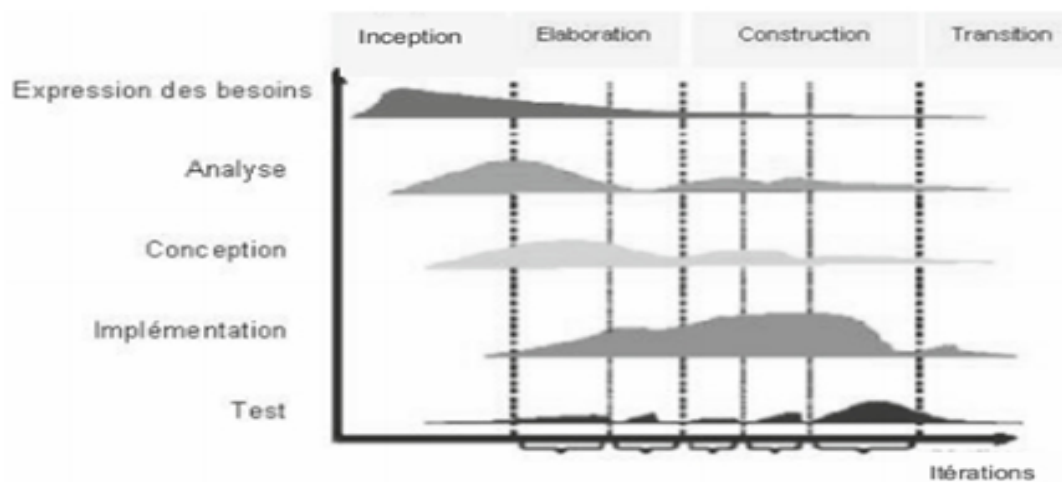


FIGURE 2.1 – Les phases du processus unifié [10]

D) Activités du processus unifié :

Ses activités de développement sont définies par cinq disciplines fondamentales qui décrivent la capture des exigences, l'analyse et la conception, l'implémentation, le test et le déploiement. Enfin, trois disciplines appelées de support complètent le tableau : gestion de projet, gestion du changement et de la configuration, ainsi que la mise à disposition d'un environnement complet de développement incluant aussi bien des outils informatiques que des documents et des guides méthodologiques [11].

2.2.2 Modélisation avec UML :

A) Définition [12] :

UML (Unified Modeling Language) ce définit comme un langage de modélisation graphique et textuel destiné à comprendre et décrire des besoins, spécifier, concevoir des solutions et communiquer des points de vue. UML unifie à la fois les notations est les concepts orientés objet. UML propose un ensemble de diagrammes

B) Points forts d'UML [13] :

- UML est un langage formel et normalisé (gains de pression, stabilité, encourage l'utilisation d'outils).
- UML est un support de communication performant :
 - Il cadre l'analyse.
 - Il facilite la compréhension abstraite.
- Son caractère polyvalent et sa souplesse en font un langage universel

C) Présentation générale des diagrammes d'UML :

UML [13] s'articule autour de treize diagrammes, chacun d'eux étant dédié à la représentation des concepts particuliers d'un système logiciel. Ces diagrammes sont repartis en deux grands groupes :

1. **Diagrammes structurels** : Ils représentent l'aspect statique d'un système (classes, objets, composants, . . .) [14]. Dans ce qui suit, nous décrivons les six diagrammes structurels d'UML.
 - **Diagramme de classes** Il représente la description statique du système où chaque classe a une partie données et une partie traitements. C'est le diagramme pivot de l'ensemble de la modélisation d'un système.
 - **Diagramme d'objets** : Permet la représentation d'instances des classes et les liens qui les relient.
 - **Diagramme de composants** : Il représente les différents constituants du logiciel au niveau de l'implémentation d'un système.
 - **Diagramme de déploiement** : Décrit l'architecture technique d'un système en se basant sur la répartition des composants dans la configuration d'explo-

tation.

- **Diagramme de paquetage** : Décrit l'ensemble du système structuré en paquetage. Chaque paquetage est un ensemble homogène d'éléments du système.
 - **Diagramme de structure composite** : Permet de décrire la structure interne d'un ensemble complexe et composé, et met aussi l'accent sur les liens entre les sous-ensembles qui collaborent.
2. **Diagrammes de comportement** : Ces diagrammes représentent la partie dynamique d'un système réagissant aux événements et permettant de produire les résultats attendus par les utilisateurs.
- **Diagramme des cas d'utilisation** : Destiné à représenter les besoins des utilisateurs par rapport au système. C'est l'un des diagrammes les plus structurants dans l'analyse d'un système.
 - **Diagramme d'état transition** : Montre les différents états des objets en réaction aux événements.
 - **Diagramme d'activités** : Donne une vision des enchainements des activités propres à une opération ou à un cas d'utilisation.
 - **Diagramme de séquences** : Permet de décrire les scénarios de chaque cas d'utilisation en mettant l'accent sur la chronologie des opérations.
 - **Diagramme de communication** : C'est une autre représentation des scénarios de cas d'utilisation qui met plus l'accent sur les objets et les messages échangés.
 - **Diagramme globale d'interaction** : Fournit une vue générale des interactions décrites dans le diagramme de séquence et des flots de contrôle décrits dans le diagramme d'activités.
 - **Diagramme de temps** : Permet de représenter les états et les interactions d'objets dans un contexte ou le temps à une forte influence sur le comportement du système à gérer.



FIGURE 2.2 – Les types des diagrammes UML2[10]

2.3 Modélisation fonctionnelle de notre système :

2.3.1 Description des cas d'utilisations :

Un cas d'utilisation est une narration qui décrit un scénario appliqué à une utilisation particulière, dans lequel les acteurs fournissent des entrées pour lesquelles le système produit une sortie observable. Le cas d'utilisation doit juste exprimer ce que doit faire un acteur, il décrit le comportement du système vu de l'extérieur [11]

Dans notre projet, le diagramme de cas d'utilisation est composé d'un seul acteur ; il s'agit de l'administrateur .Ce dernier alimente la base de données du système en introduisant les ressources nécessaire à l'élaboration des emplois du temps . Les ressources considérées sont les entités physiques suivantes : Enseignants, Locaux, Modules, Sections et groupes.

2.3.2 Description textuelle des cas d'utilisation :

La fiche de description textuelle d'un cas d'utilisation n'est pas définie par UML. Le formalisme qu'on utilisera est juste une proposition.

1. **Cas d'utilisation : authentification :** L'authentification est le point d'entrer de l'application puisque, elle présente la première fenêtre qu'il affiche pour authentifier l'administrateur.u

Cas d'utilisation	Authentification
Résumé	Vérifier les possibilités d'accès au système
Scénario nominal	[Debut] 1. Accès à l'application. 2. Saisie du login et du mot de passe. 3. Le système vérifie les données saisies. 4. Le système affiche l'interface correspondante. [Fin]
Alternative	A1 (alternative) est une étiquette qui indique que le système réaffichera le formulaire d'authentification dans le cas où les données introduites sont erronées.

TABLE 2.1 – Description du cas d'utilisation : Authentification

2. Cas d'utilisation : gestion des enseignants :

Le cas d'utilisation " gestion des enseignants " est caractérisé par les trois scénarios suivants :

- Ajout.
- Modification.
- Suppression.

Cas d'utilisation	Gestion des enseignants
Résumé	L'administrateur gère les enseignants impliqués dans la planification, ceci en ajoutant, en modifiant ou en supprimant un enseignant.
Scénario nominal	<p>[Debut]</p> <ol style="list-style-type: none"> 1. Accès à l'application. 2. Authentification. 3. Le système affiche l'interface correspondante. 4. L'administrateur demande le formulaire de gestion des enseignants. 5. Le système affiche la liste des enseignants. 6. Trois alternatives se présentent : dans le cas où l'administrateur veut ajouter un enseignant, le système affiche le formulaire d'ajout. Si l'administrateur choisit de modifier (ou de supprimer) un enseignant, il doit alors sélectionner l'enseignant en question, puis effectuer l'opération. 7. Le système affiche un message de confirmation. <p>[Fin]</p>

TABLE 2.2 – Description du cas d'utilisation : gestion des enseignants

2.4 Diagramme de cas d'utilisation de l'application :

Le diagramme de cas d'utilisation est un formalisme permettant de modéliser le fonctionnement d'un système par un découpage en fonctionnalités. Il illustre de plus la nature des interactions avec ces fonctionnalités offertes à titre de services à des acteurs externes au système. Chaque fonctionnalité est appelée un cas d'utilisation [15]

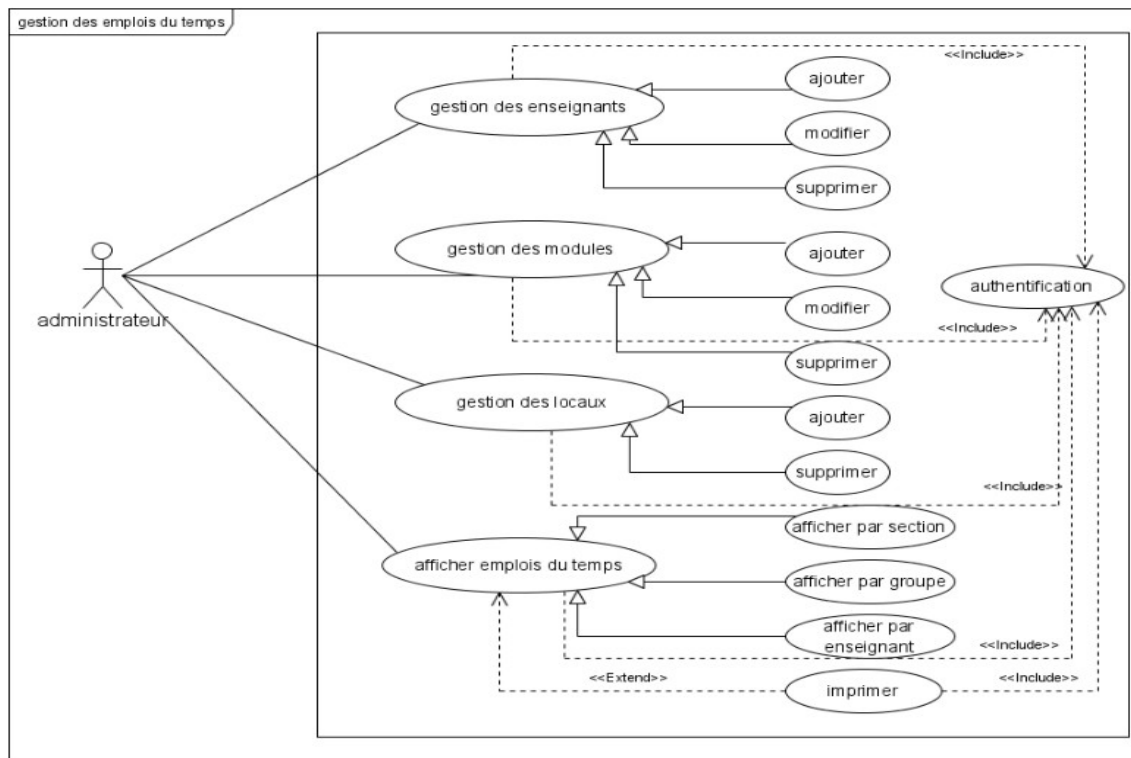


FIGURE 2.3 – Diagramme de cas d'utilisation pour la gestion des emplois du temps.

2.4.1 Réalisation des diagrammes de séquences :

L'objectif du diagramme de séquence est de représenter les interactions entre objets en indiquant la chronologie des échanges. Cette représentation peut se réaliser par cas d'utilisation en considérant les différents scénarios associés[16]

Diagramme de séquence du cas d'utilisation : Authentification

L'authentification consiste à assurer la confidentialité des données, elle se base sur la vérification des informations associées à un utilisateur (généralement un login et un mot de passe). Lors d'une authentification deux cas se présentent : les informations introduites par l'utilisateur sont incomplètes, dans ce cas un message d'erreur s'affiche, ou les informations saisies sont complètes et le système procède à leur vérification. Ceci explique l'utilisation de l'opérateur "alt".

Le même opérateur illustre les deux réactions du système, après la vérification des informations saisies par l'utilisateur, soit par l'affichage d'un message d'erreur, ou de l'interface correspondante.

L'opérateur "alt" : correspond à une instruction de test avec une ou plusieurs alternatives possibles. Il est aussi permis d'utiliser les clauses de type sinon.

L'opérateur "loop" : correspond à une instruction de boucle permettant d'exécuter une séquence d'interactions tant qu'une condition est satisfaite.

L'opérateur "ref" : permet d'appeler une séquence d'interactions décrite par ailleurs constituant ainsi une sorte de sous diagramme de séquence.

L'opérateur "opt" : (optional) correspond à une instruction de test sans alternative (sinon).

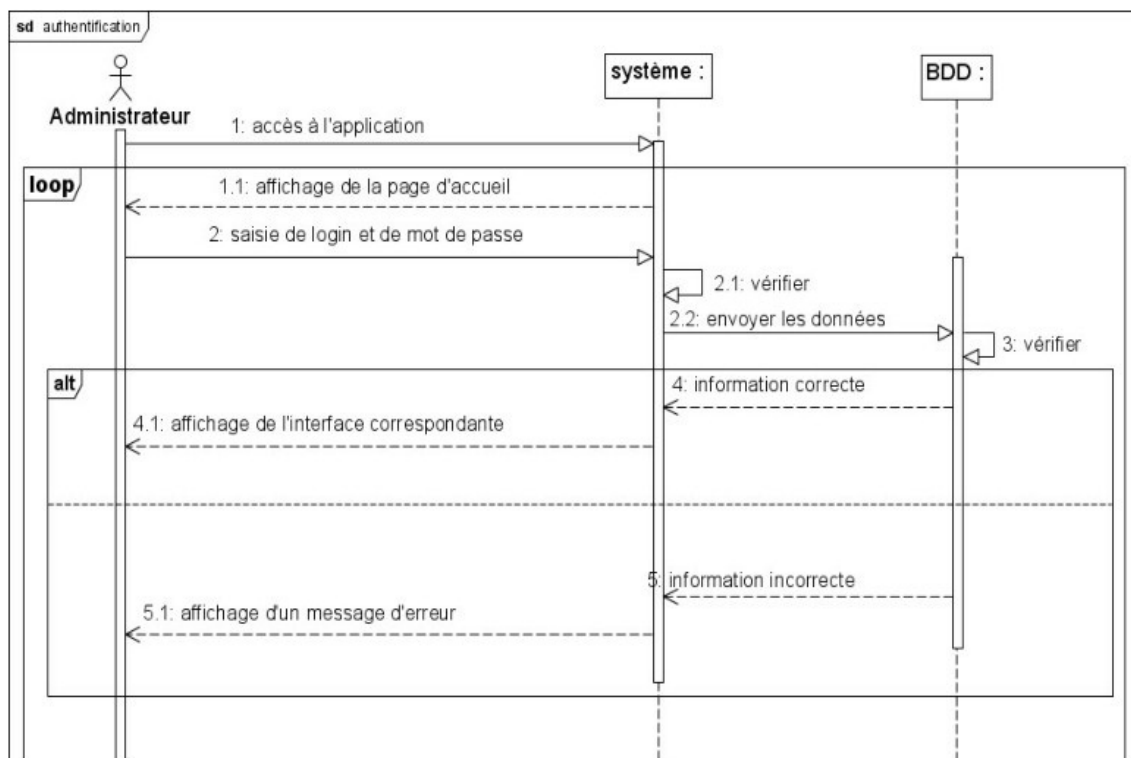


FIGURE 2.4 – Diagramme de séquence du cas d'utilisation : Authentification.

2.4.2 Diagramme de séquence du cas d'utilisation : Gestion des enseignants

Ce cas d'utilisation comporte trois scénarios "ajouter", "modifier" ou "supprimer" un enseignant.

Dans le cas d'ajout : Le système répond à la demande de l'administrateur concernant l'ajout d'un enseignant par l'affichage d'un formulaire qui sera validé après remplissage.

Dans le cas de modification : La réponse du système, pour la requête de modification, est la liste des enseignants. Une fois l'utilisateur à modifier est choisi, un formulaire s'affiche afin d'apporter les modifications souhaitées.

Dans le cas de suppression : La réponse du système, pour la requête de suppression, est la liste des enseignants. Une fois l'enseignant à supprimer est sélectionné, il sera supprimé définitivement de la base de données.

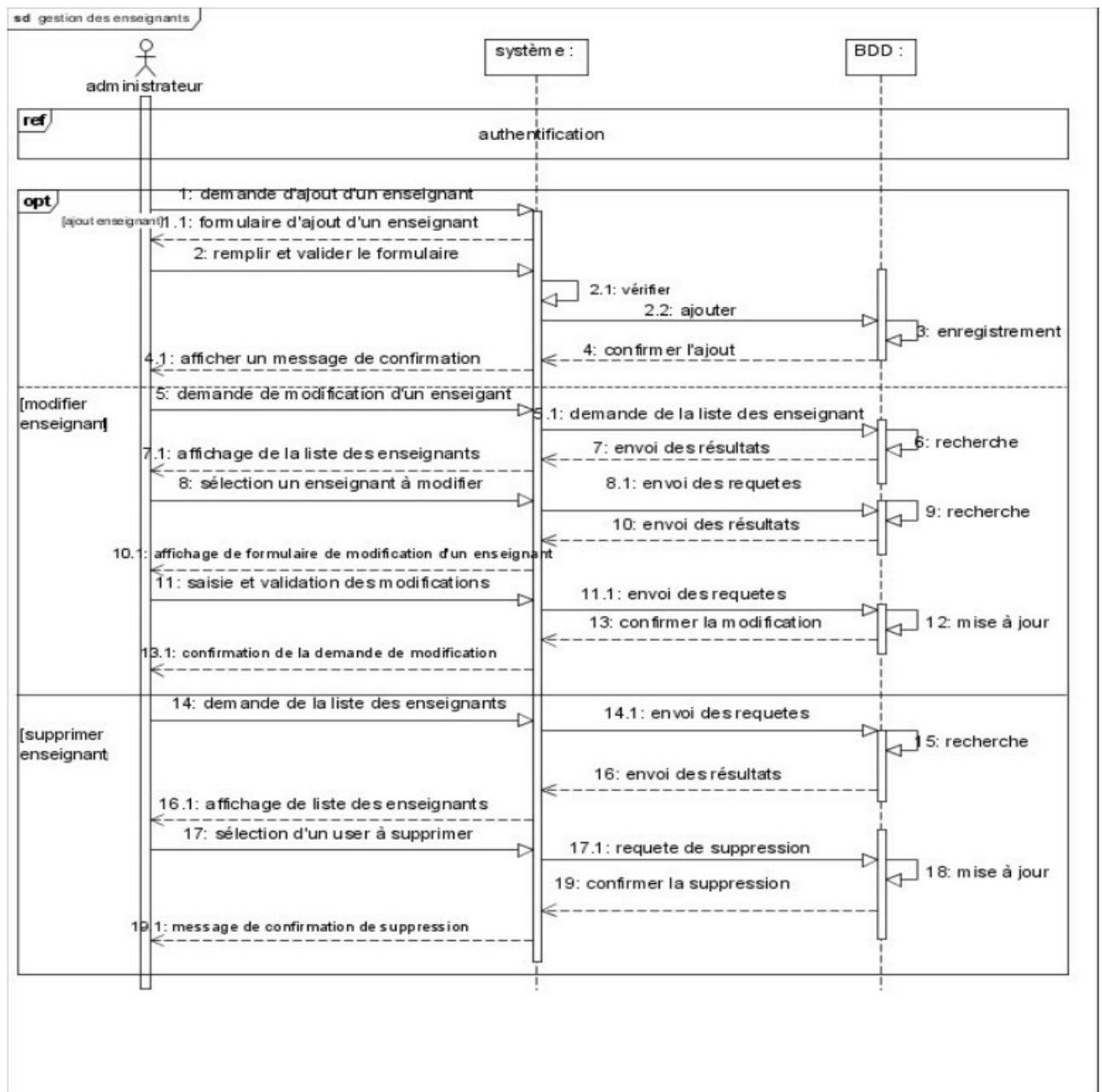


FIGURE 2.5 – Diagramme de séquence du cas d'utilisation : gestion des enseignants.

Diagramme de séquence du cas d'utilisation : Gestion des locaux

Ce cas d'utilisation comporte deux scénarios " ajouter " ou " supprimer " un local. *Dans le cas d'ajout* : Le système répond à la demande de l'administrateur concernant l'ajout d'un local par l'affichage d'un formulaire qui sera validé après remplissage. *Dans le cas de suppression* : La réponse du système, pour la requête de suppression, est la liste des locaux. Une fois un local à supprimer est sélectionné, il sera supprimé définitivement de la base de données

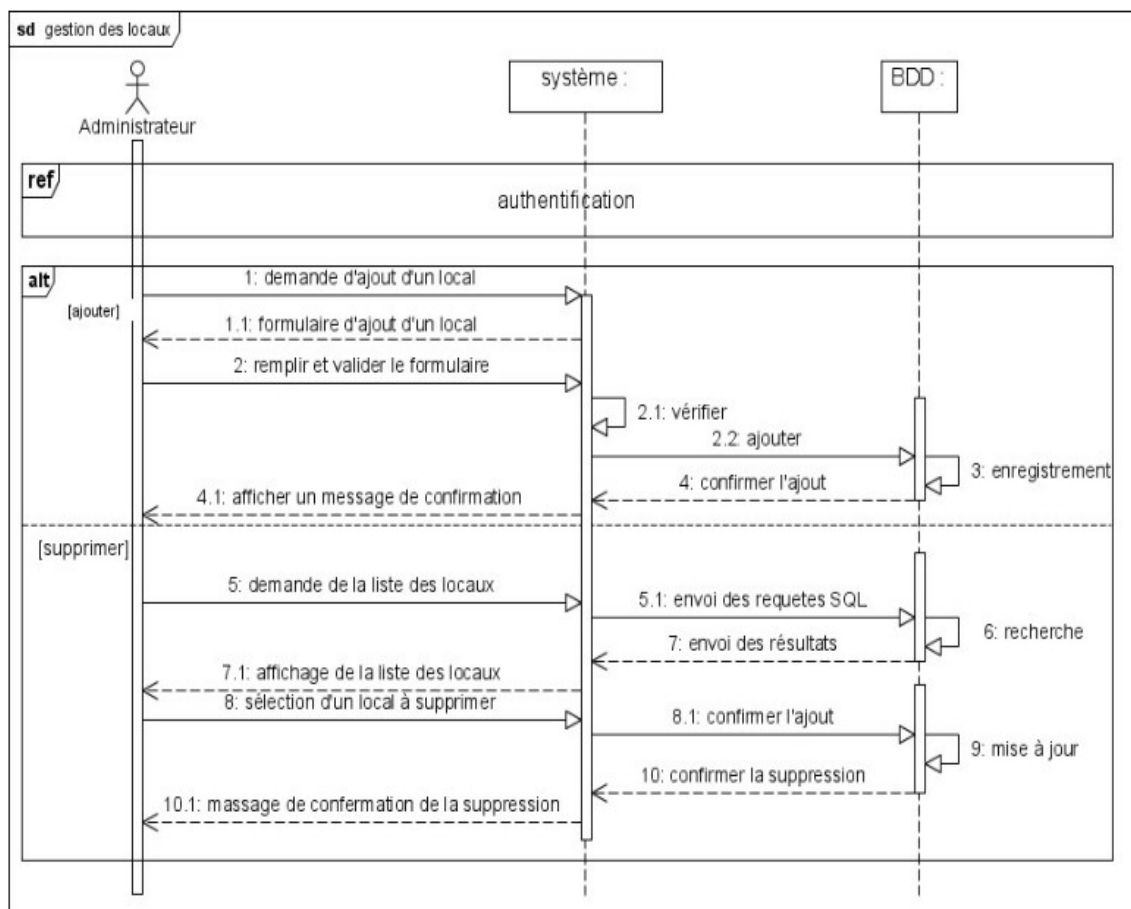


FIGURE 2.6 – Diagramme de séquence du cas d'utilisation : gestion des locaux.

Diagramme de séquence du cas d'utilisation : afficher emplois du temps par section :

Ce diagramme présente l'affichage de l'emploi du temps pour un niveau et une section donnée.

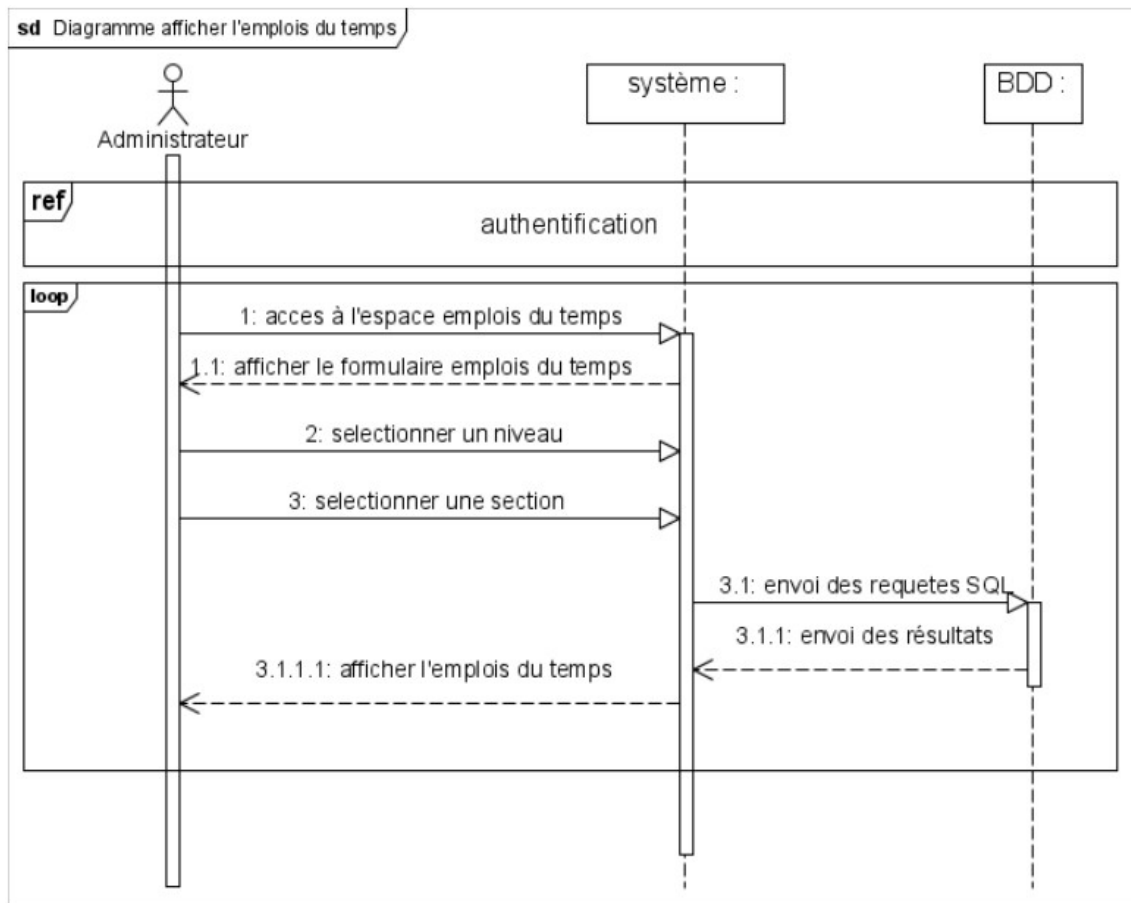


FIGURE 2.7 – Diagramme de séquence du cas d'utilisation : afficher emplois du temps par section.

2.4.3 Réalisation des diagrammes d'activité :

Le diagramme d'activité est une variante du diagramme d'état transition .Il permet de représenter graphiquement le comportement d'une méthode ou le déroulement d'un cas d'utilisation.

Les diagrammes d'activité de nos cas d'utilisation sont présentés sur les figures suivantes :

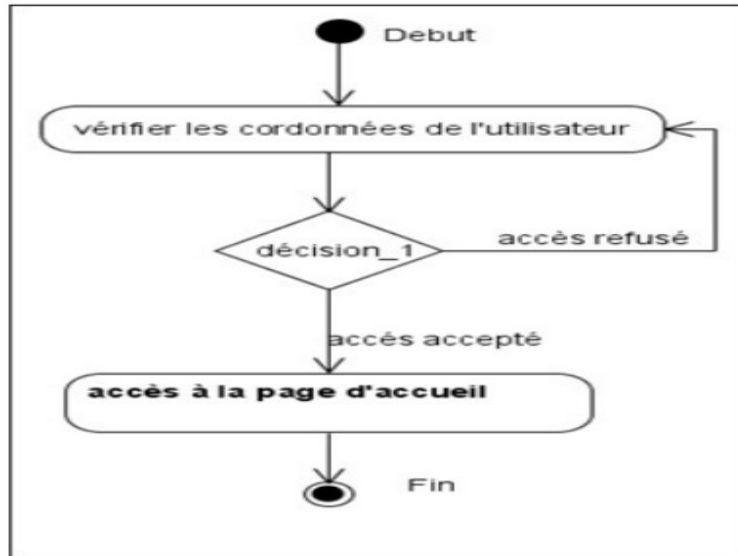


FIGURE 2.8 – Diagramme d’activité : Authentification.

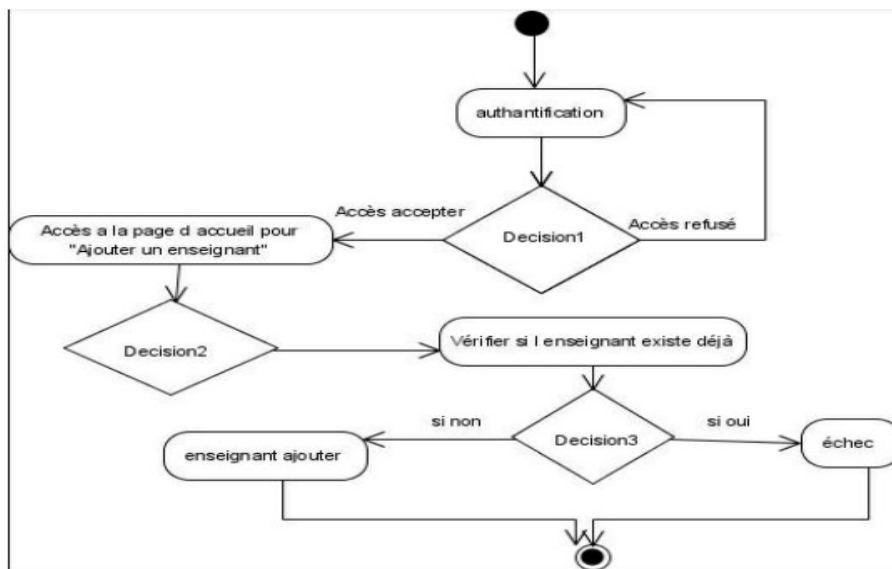


FIGURE 2.9 – Diagramme d’activité : Ajouter un enseignant.

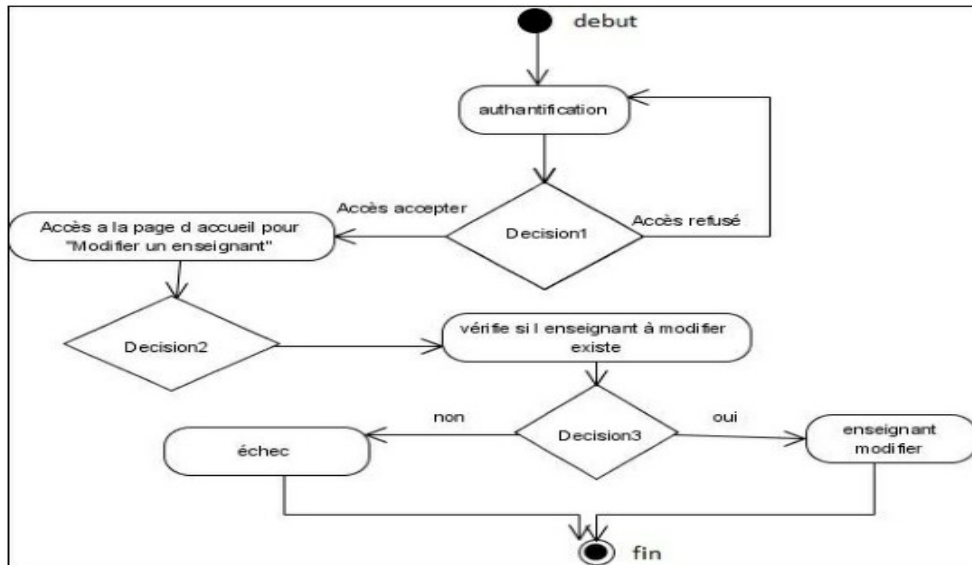


FIGURE 2.10 – Diagramme d’activité : Modifier un enseignant.

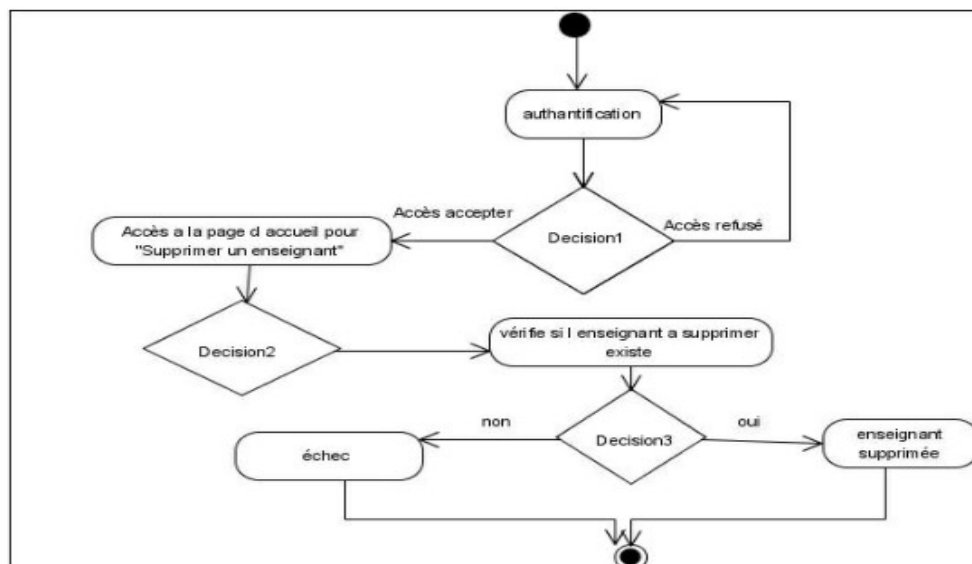


FIGURE 2.11 – Diagramme d’activité : Supprimer un enseignant.

2.5 Modélisation structurelle de notre application

- 1) **Dictionnaire de données épuré** : Le tableau suivant décrit les attributs de chacune des classes et classe d'association du diagramme de notre application

Classe	Attributs	Designation attributs	Types attributs
Enseignant	identifiant	Code de l'enseignant	VARCHAR
	nom_ens	Nom de l'enseignant	VARCHAR
	prenom_ens	Prénom de l'enseignant	VARCHAR
	grade	Grade de l'enseignant	VARCHAR
	specialite	Spécialité de l'enseignant	VARCHAR
	num_tel	Numéro de téléphone de l'enseignant	VARCHAR
	email	Email de l'enseignant	VARCHAR
	statut	Statut de l'enseignant	VARCHAR
	affiliation	Affiliation de l'enseignant	VARCHAR
module	id_module	Code du module	VARCHAR
	libelle	Nom du module	VARCHAR
	nbr_cours	Nombre de cours d'un module	INT
	nbr_td	Nombre de td d'un module	INT
	nbr_tp	Nombre de tp d'un module	INT
assurer	type_enseignement	Type enseignement	VARCHAR
	heur	Heur du créneau	VARCHAR
	jour	Jour du créneau	VARCHAR
local	id_local	Identificateur du local	VARCHAR
	num_local	Nom du local	VARCHAR
	type_local	Type du local	VARCHAR
	capacite	Nombre de place de local	INT
niveau	num_niveau	Niveau d'étude	VARCHAR
	type	Spécialité	VARCHAR
section	id_section	Identificateur de la section	VARCHAR
	nom_section	Nom de la section	VARCHAR
groupe	id_groupe	Identificateur du groupe	VARCHAR
	nom_groupe	Le numéro du groupe	VARCHAR

FIGURE 2.12 – Dictionnaire de données épuré

- 2) **Diagramme de classes** : Le diagramme de classe constitue l'un des pivots essentiels de la modélisation avec UML. En effet, ce diagramme permet de donner la représentation statique du système à développer. Cette représentation est basée sur les concepts de classe et d'association. Chaque classe se décrit par les données et les traitements dont elle est responsable. Les traitements sont matérialisés par des

opérations. Le détail des traitements n'est pas représenté directement dans le diagramme de classe, mais dans l'algorithme général et le pseudo-code correspondant[10]

3) Les concepts d'un diagramme de classe :

- **Un objet**[10] : Un Objet est un concept, une abstraction ou une chose qui a un sens dans le contexte du système à modéliser. Chaque objet a une identité et peut être distingué des autres, sans considérer a priori les valeurs de ses propriétés .
- **Une classe**[16] : Une classe représente la description abstraite d'un ensemble d'objets possédant les mêmes caractéristiques. On peut parler également de type .

Un objet est une instance d'une classe. La classe représente l'abstraction de ses objets. Au niveau de l'implémentation, c'est-à-dire au cours de l'exécution d'un programme, l'identificateur d'un objet correspond à une adresse mémoire . Ce qui suit est la représentation graphique d'une classe :

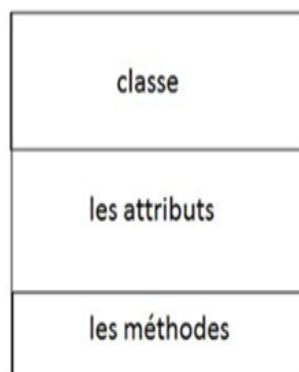


FIGURE 2.13 – Représentation graphique d'une classe.

- **Attribut** : Un attribut est une donnée élémentaire servant à caractériser les classes et les relations [15]
- **Opération** : Une opération représente un élément de comportement (un service) contenu dans une classe. Nous ajouterons plutôt les opérations en conception objet, car cela fait partie des choix d'attribution des responsabilités aux objets [11]
- **Méthode** : Une méthode est une opération programmée sur les objets d'une classe [10].
- **Classe d'association** : Il s'agit d'une association promue au rang de classe.

Elle possède de tout à la fois les caractéristiques d'une association et celle d'une classe et peut donc porter des attributs qui se valorisent pour chaque lien [16].

4) **Relations entre les classes** : En UML, il existe plusieurs types de relations entre les classes :

— **Association** : Une association se fait entre deux classes. Elle a un nom et deux extrémités qui permettent de la connecter à chacune des classes associées. Lorsqu'une association est définie entre deux classes, cela signifie que les instances de ces deux classes peuvent être reliées entre elles [17].

— **Composition** : La composition est une relation d'agrégation dans laquelle il existe une contrainte, de durée de vie entre la classe "composant" et la ou les classes "composé". Autrement dit la suppression de la classe "composé" implique la suppression de la ou des classes "composant" [10]

5) **Diagramme de classes de l'application à réaliser** : Le diagramme suivant, représente le diagramme de classes associé à l'application que nous réaliserons :

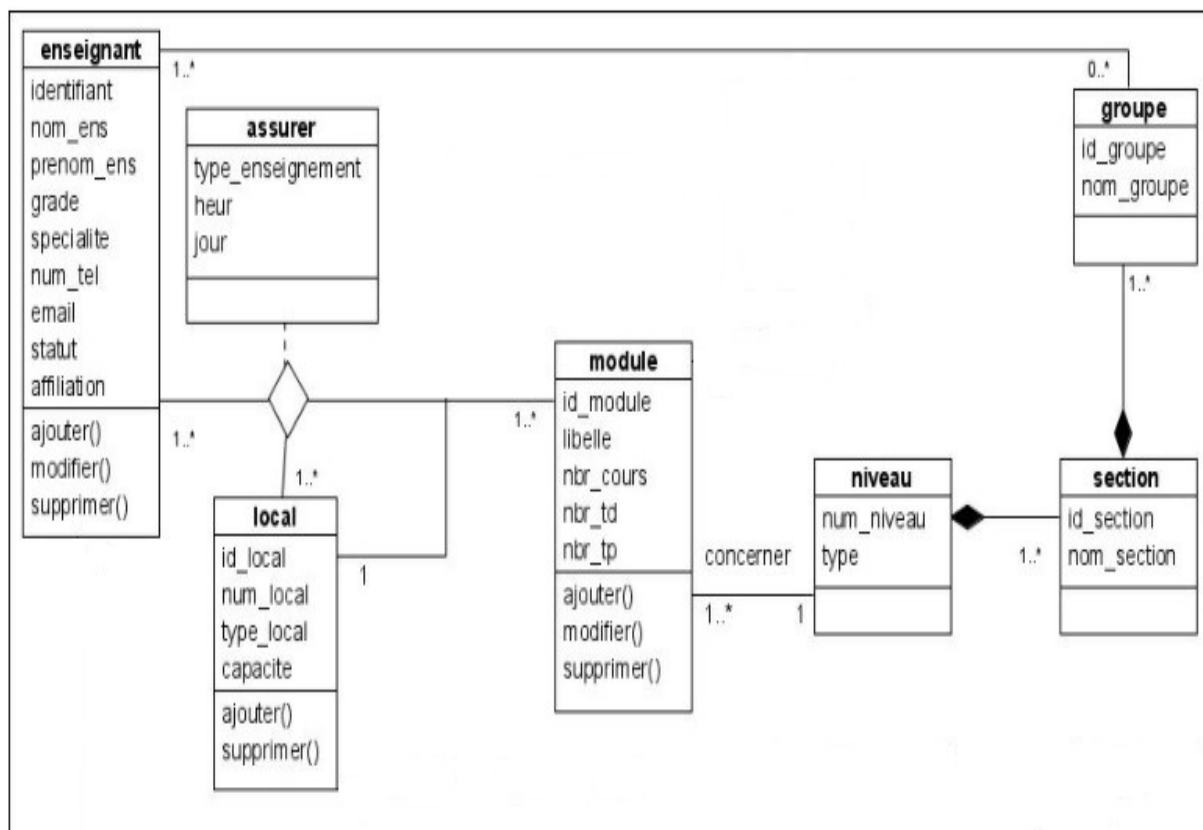


FIGURE 2.14 – Diagramme de classes de l'application.

2.6 Modèle Relationnel de données :

Le concepteur d'une base de données relationnelle doit élaborer ce qu'il est convenu d'appeler le schéma relationnel de la base de données. Cette activité consiste à définir toutes les relations normalisées de la base de données et les domaines de leurs attributs. Théoriquement cela consisterait à décrire par intention chaque relation et définir les domaines de chaque attribut de la relation[10]

Règles de dérivation du modèle relationnel à partir d'un diagramme de classes :

Pour avoir le schéma relationnel de la base de données de l'application à réaliser à partir du modèle de classes, nous utilisons les règles de passage suivantes [17]

- **Règle 1** : Transformation des classes

Chaque classe du diagramme UML devient une relation. Il faut choisir un attribut de la classe pouvant jouer le rôle de l'identifiant. Dans le cas où aucun attribut ne convient en tant qu'identifiant, il faut en ajouter un de telle sorte que la relation dispose d'une clé primaire.

Classe -association et n-aire

- **Règle 2** : Association un-à-plusieurs

Il faut ajouter un attribut de type clé étrangère dans la relation fils de l'association. L'attribut porte le nom de la clé primaire de la relation père de l'association.

- **Règle 3** : Associations plusieurs-à-plusieurs

L'association devient une relation dont la clé primaire est formée par la concaténation des identifiants des classes connectées à l'association. Les attributs de l'association doivent être ajoutés à la nouvelle relation.

L'application des règles de passage énumérées précédemment, nous permet d'avoir le modèle relationnel de la base de données de l'application à mettre en œuvre.

- Enseignant (identifiant, nom-ens, prenom-ens, grade, specialite, num-tel, email, statut, affiliation).
- Module (id-module, libelle, nbr-cours, nbr-td, nbr-tp)
- Assurer (identifiant, id-module, id-local, heure, jour, type-nseignement)
- Local (id-local, num-local, type-local, capacite).

- Niveau (num-niveau, type).
- Section (id-section, num-niveau, nom-section)
- Groupe (id-groupe, id-section, num-niveau, nom-groupe).

2.7 Conclusion

Dans ce chapitre, nous avons commencé par une présentation du processus unifié suivi durant le développement ainsi que l'UML et ses différents diagrammes. Ensuite, nous avons présenté les cas d'utilisation, les diagrammes de séquence qui leur correspondent ainsi que les diagrammes d'activité. Enfin, nous avons terminé par le diagramme de classes et le modèle relationnel de données qui nous permet d'avoir le schéma de la base de données de l'application à réaliser dans le chapitre suivant.

Conception et implémentation

3.1 Introduction

Après avoir finalisé l'étape de conception, nous passons dans ce chapitre à l'implémentation de notre application. Cette application a permis de répondre aux besoins des utilisateurs par la résolution des problèmes de la gestion de l'emploi du temps, qui est l'étape la plus importante dans ce travail.

En premier lieu, on a parlé brièvement sur l'environnement et les technologies nécessaires pour la réalisation de notre projet. On a utilisé SQL pour l'élaboration des requêtes d'interrogation de la base de données, et Java pour l'écriture du code. En suite on a enchaîné par la présentation du travail réalisé avec l'aperçu des interfaces de notre application.

3.2 Outils et langages de développements

La réalisation d'une application de gestion d'emploi du temps se base sur l'utilisation de langages. Pour cet objectif nous avons utilisé :

3.2.1 XAMPP

XAMPP est un ensemble de logiciels permettant de mettre en place un serveur Web local, un serveur FTP et un serveur de messagerie électronique. Il s'agit d'une distribution de logiciels libres (X (cross) Apache MariaDB Perl PHP) offrant une bonne souplesse d'utilisation, réputée pour son installation simple et rapide. Ainsi, il est à la portée d'un grand nombre de personnes puisqu'il ne requiert pas de connaissances particulières et

fonctionne, de plus, sur les systèmes d'exploitation les plus répandus .

- X pour cross-plateforme (LAMP pour Linux, WAMP pour Windows,...)
- A pour Apache
- M pour MySQL
- P pour PHP
- P pour Perl

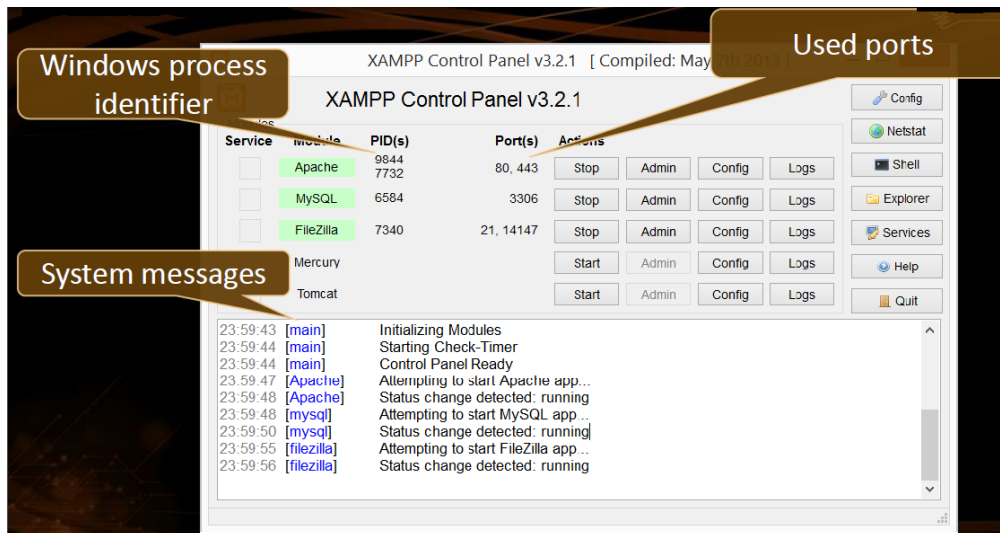


FIGURE 3.1 – Exemple de XAMPP

3.2.2 phpMyAdmin



Il s'agit de l'une des plus célèbres interfaces pour gérer une base de données MySQL sur un serveur PHP. De nombreux hébergeurs, gratuits comme payants, le proposent ce qui évite à l'utilisateur d'avoir à l'installer. Cette interface pratique permet d'exécuter, très facilement et sans grandes connaissances en bases de données, des requêtes comme les créations de table de données, insertions, mises à jour, suppressions et modifications de structure de la base de données, ainsi que l'attribution et la révocation de droits et

l'import/export. Ce système permet de sauvegarder commodément une base de données sous forme de fichier. SQL et d'y transférer ses données, même sans connaître SQL, en plus, il est très convenable pour des base de données non surchargées.

3.2.3 Langage de codage (JAVA Netbeans)



Apparu fin 1995 et développé par Sun Microsystems Java s'est très rapidement taillé une place importante dans le domaine de l'internet et des applications client-serveur. Les objectifs de java sont d'être multiplateformes et d'assurer la sécurité aussi bien pendant ledéveloppement que pendant l'utilisation d'un programme java. Java est algorithmique et orienté objet ; à ce titre il peut effectuer toutes les taches d'un tel langage (graphiques, multimédias, bases de données, environnement de développement, ... etc)

Son point fort qui le démarque des autres langages (C++, Delphi) est sa probabilité due (en théorie) à ses bibliothèques de classes indépendantes de la plate-forme, ce qui est le point essentiel de la programmation sur internet ou plusieurs machines dissemblables sont interconnectées

Dans notre projet on a implémenté notre algorithme sur la version JAVA Netbeans 8.2

3.3 Implémentation de la base de données

Notre base de données est interrogée par SQL intégré dans le serveur PHPMyadmin.Cette base regroupe les tables essentielles pour la gestion des emplois du temps, comme locaux,module, enseignants ... etc.et une autre table très nécessaire pour l'authentification et la sécurité d'application qui s'appele login.

Et voilà les tables de notre base de données créés dans la figure suivante :

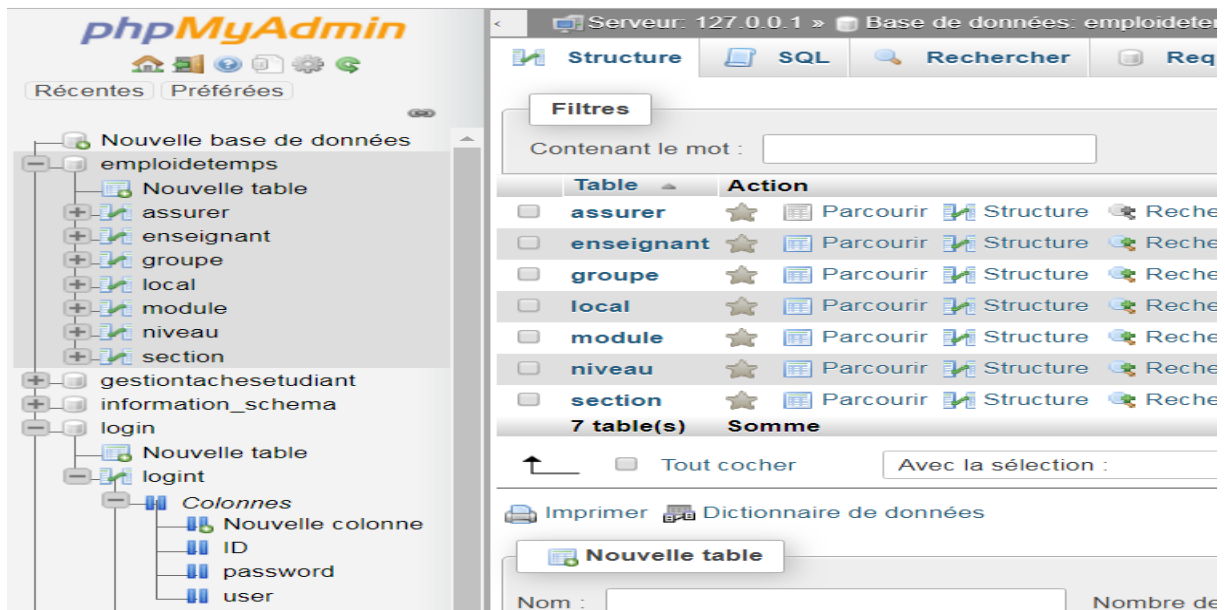


FIGURE 3.2 – Base de données de notre application

3.4 Description de l'application

3.4.1 Fenêtre d'authentification

La fenêtre principale constitue une fenêtre d'un logi mot de passe pour accéder au programme principal.



FIGURE 3.3 – Interface pour accéder à l'application

Si le mot de passe et login sont correct l'interface de menu s'affiche sinon en cas d'erreur d'authentification le système retourne un message comme suit :

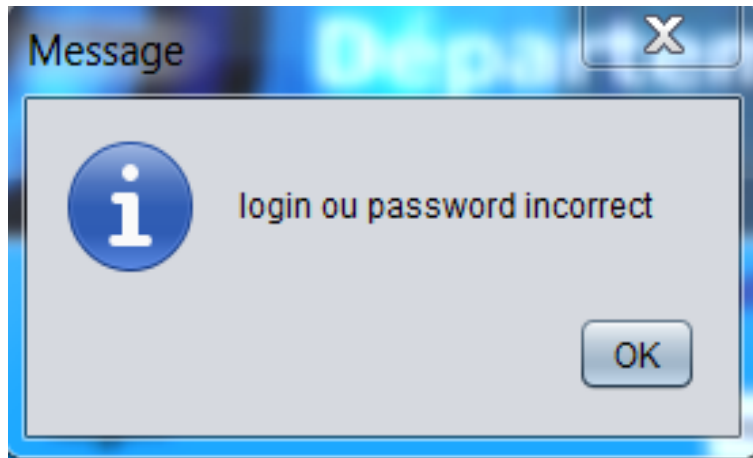


FIGURE 3.4 – Problème d'authentification

3.4.2 Menu principale

après avoir une correcte authentification , l'interface Menu est la premiere l'interface qui s'affiche et elle contient 4 choix (gestion des enseignants,des locaux,des modules ,et emploi de temps) .

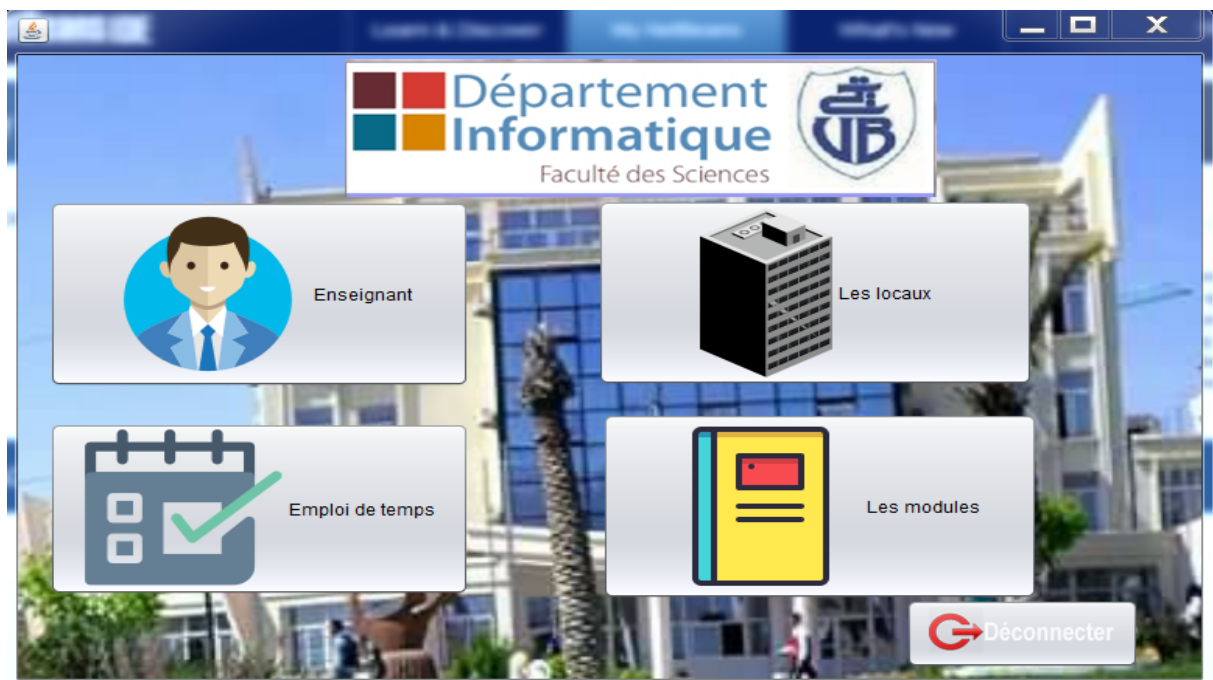


FIGURE 3.5 – Menu de l'application

3.4.3 Gestion des enseignants

Cette interface représente le formulaire de gestion d'un enseignant.

- l'administrateur peut ajouter un enseignant après avoir rempli tous les champs puis cliquer sur le bouton ajouter pour que l'enseignant soit ajouté directement à la base de données
- l'administrateur peut modifier les champs qu'il désire puis cliquer sur bouton modifier pour que l'enseignant soit modifié directement dans la base de données
- Il peut supprimer et chercher l'enseignant par son nom.

comme montre cette figure :

gestion des Enseignants

Informations:

id enseig num de tel

nom email

pre nom statut

grade affiliation

specialité

id	Nom	pre nom	grade	specialite	num_tel	email	statut	affiliation
1	ben amer	youcef	professeur	ISIL	0567435	youcef@gm...	permanant	informatique
2	benakil	aqila	professeur	reseau	07685436	benakil@gm...	permanant	informatique
3	ktab	mesouda	professeur	GSI	05467543	ktab@gmail...	permanant	informatique

FIGURE 3.6 – Gestion des enseignants

3.4.4 Gestion des modules

cette interface a des meme fonctionalité d'interface précédente mais cette fois pour les modules.

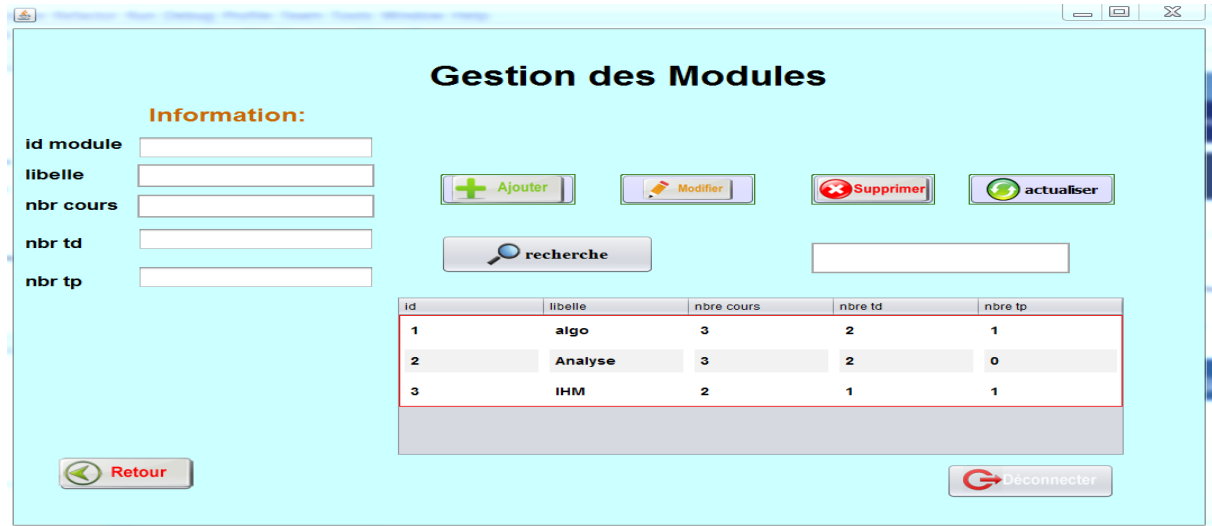


FIGURE 3.7 – Gestion des modules

3.4.5 Gestion des locaux

cette interface permet d'ajouter , supprimer et chercher les locaux mais elle permet pas de les modifier puisque les locaux sont stables

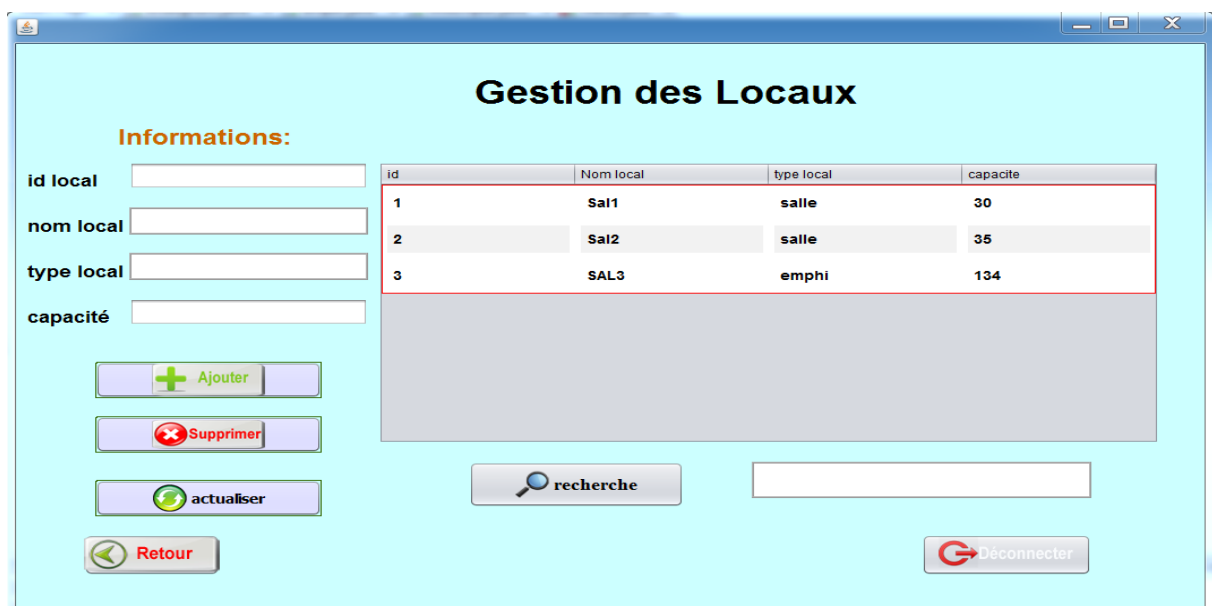


FIGURE 3.8 – Gestion des locaux

3.4.6 Génération d'emploi de temps

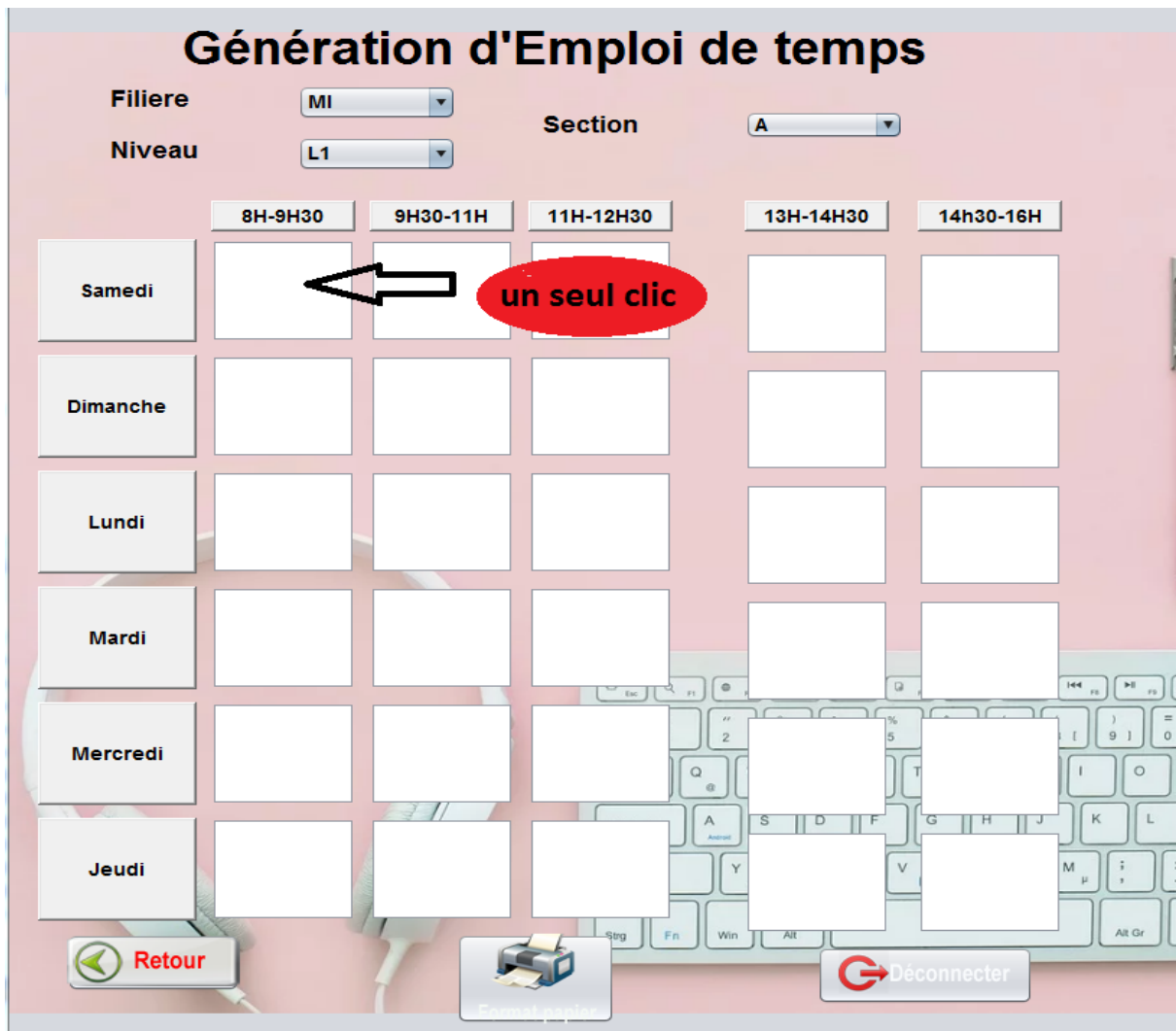
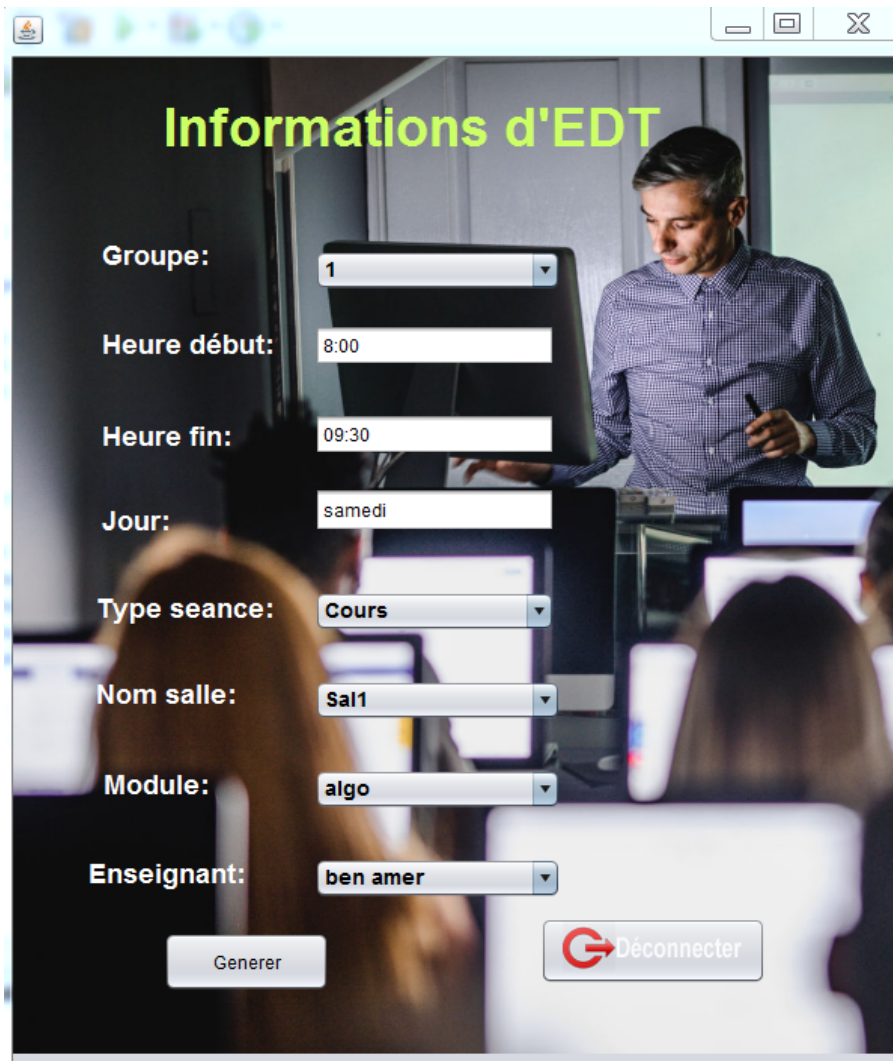


FIGURE 3.9 – Génération d'emploi de temps

On clic sur chaque cellule, il apparait l'interface suivante pour récupérer les informations de chaque cellule.

L'interface pour générer l'EDT :



Informations d'EDT

Groupe: 1

Heure début: 8:00

Heure fin: 09:30

Jour: samedi

Type seance: Cours

Nom salle: Sal1

Module: algo

Enseignant: ben amer

Generer

Déconnecter

FIGURE 3.10 – les informations de seance

Après la récupération des informations de toutes les cellules on imprime notre emploi du temps en format PDF.

3.5 Conclusion

Dans ce chapitre, nous avons défini les outils de développement que nous avons utilisés dans la réalisation de notre application, et nous avons montré les interfaces et le fonctionnement de notre application.

Conclusion générale

Durant les derniers trois mois, nous avons analysé et réalisé un projet de fin d'étude relatif à la gestion d'emploi du temps au sein du département MI de l'université Akli Mohand Oulhadj Bouira.

Dans ce projet, nous avons exploité nos efforts et nos connaissances pour pouvoir bien analyser le sujet de la génération d'emploi du temps, Malgré l'éventail de logiciels qui ont essayé de traiter ce problème et la multitude d'approches utilisées, le problème reste toujours posé, car lui-même a plusieurs facettes (selon nos recherches documentaires). Aucune modélisation standard, qui englobe toutes les variantes du problème, n'a été formulée.

Nous souhaitons et espérons avoir résolu une grande partie du problème de la génération d'emploi du temps, en désirants que nous aurons l'occasion de le développer et l'enrichir pour plus d'efficacité, selon les perspectives suivantes :

- Réalisation d'un système qui relie autres départements et facultés.
- la réalisation d'une application réaliser toutes les contraintes cité avant.
- enrichir notre application de partie des examens et des soutenances .
- Héberger le système dans le serveur web de l'Université Akli Mohand Oulhadj Bouira.

Nous espérons que dans le prochain temps , nous pouvons reprendre le travail là où nous l'avons laissé et mener à terme le développement de ce programme, et l'enrichir pour résoudre les besoins de département

Bibliographie

Bibliographie

- [1] <http://www.univ-bouira.dz>

- [2] Chan Yew Chéong, Peter, « La planification du personnel : acteurs, actions et termes multiples pour une planification opérationnelle des personnes », Thèse de doctorat, Institut IMAG, Université Joseph Fourier-Grenoble, 1 octobre 2002.

- [3] Georges Weil, Kamel Heus, Patrice François, « Gymnaste : Aide à l'élaboration des roulements infirmiers . Du traitement des absences au management participatif », Laboratoire TIMC, SILM , CHU de Grenoble , Université Joseph Fourier- Grenoble,1994

- [4] Troudi. F, Résolution du problème de l'emploi du temps : Proposition d'un algorithme évolutionnaire multi objectif, mémoire Magister en Informatique, Université Mentouri–Constantine, 2006

- [5] [Http://futurcpe.free.fr/donnees/histoire/pedagogie](http://futurcpe.free.fr/donnees/histoire/pedagogie)

- [6] <http://juliecogill.com/Chapter-2.pdf>

- [7] Fadjra. D, Noussiba. G, Conception et réalisation d'un portail web (e-université) pour le suivi pédagogique des enseignants et l'évaluation des étudiants, Université Kasdi Merbah-Ouargla, 2013

- [8] Pascal. Roque et Franck. Vallée, " ULM 2 en action, de l'analyse des besoins à la conception ", 4ème édition. EYROLLES, 2007
- [9] Pascal. Roques, "Les cahiers du programmeur UM2, modélisé une application web", 4eme édition, EYROLLES, 2007.
- [10] Joseph. Gabay et David. Gabay, "UML2 Analyse et conception", 1ère édition.DUNOD", 2008
- [11] Pascal. Roque, " ULM 2 en action, de l'analyse des besoins à la conception ", 4ème édition. EYROLLES, 2009.
- [12] [Http : //uml.free.fr](http://uml.free.fr)
- [13] [Http : // UML.developpez.com/Lp/cours/uml-free](http://UML.developpez.com/Lp/cours/uml-free)
- [14] Pascal Roques, "UML2 Modéliser une application Web", 3edition EYROLLES, 2000.
- [15] Rémy. Fanader et Hervé. Leroux, "UML principes de modélisation", DUNOD, paris", 1999.
- [16] Pascal. Roques, "UML2 par la pratique", Edition EYROLLES.Paris 5, 2006
- [17] Gilles et Roy, "UML2 modéliser une application web", EYROLLES.Paris, 4ème édition, 2008