

long-short
french



REPOBLIKAN'I MADAGASIKARA
FITIAVANA - TANINDRAZANA - FANDROSOANA
MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR
ET DE LA RECHERCHE SCIENTIFIQUE



ECOLE SUPERIEURE POLYTECHNIQUE D'ANTSIRANANA

Mention STIC

MÉMOIRE DE FIN D'ÉTUDES

Parcours Électronique et Informatique Industrielles

Développement de SaaS

« Groupe4SuccessManagement »

Par :

RATSIZAFY Heriniaina Midonique

Encadreurs :

M. ANDRINIRINIAIMALAZA Fanambinantsoa Philibert

M. RAMANAN'HAJA Hery Tina

Encadreur Professionnel

★ *Année Universitaire 2016 - 2017* ★
Promotion : RAITRA



REPOBLIKAN'I MADAGASIKARA
FITIAVANA - TANINDRAZANA - FANDROSOANA
MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR
ET DE LA RECHERCHE SCIENTIFIQUE



ECOLE SUPERIEURE POLYTECHNIQUE D'ANTSIRANANA

Mention STIC

MÉMOIRE DE FIN D'ÉTUDES

Parcours Électronique et Informatique Industrielles

Développement de SaaS

« Groupe4SuccessManagement »

Par :

RATSIZAFY Heriniaina Midonique

Encadreurs :

M. ANDRINIRINIAIMALAZA Fanambinantsoa Philibert

M. RAMANAN'HAJA Hery Tina

Encadreur Professionnel

Membres de jury :

Dr. RAOELIVOLOLONA Tefy Rakotobe

Président de jury

M. ANDRINIRINIAIMALAZA Fanambinantsoa Philibert

Encadreur

M. RAMANAN'HAJA Hery Tina

Encadreur

Dr. ANDRIAMIHARINJAKA Hasina

Examineur

Dr. ANDRIANA JAINA Todizara

Examineur

★ Année Universitaire 2016 - 2017 ★
Promotion : RAITRA

Projet et Mémoire de fin d'études – A.U : 2016/2017

Titre : Développement de SaaS « Group4SuccessManagement »

Contexte

Avec ou sans basculement vers le système LMD, toutefois le système d'information étudiants, des établissements d'enseignement supérieur, ne répond plus aux besoins du modèle d'activité existante. D'où l'idée de développement d'un SaaS (Software as a Service ou logiciel en tant que service) dénommé « Group4Success » qui est destiné pour : alléger les efforts de gestion de personnel, soutenir la croissance de l'établissement, stimuler la réussite des étudiants et améliorer l'efficacité administrative. Comme il s'agit d'un SaaS, l'application et infrastructure sous-jacente sont intégralement hébergées à distance et administrées par les soins de chaque établissement.

Objectif

C'est de développer et mettre œuvre le SaaS « Group4Success ».

Travaux demandés

- Études théoriques sur les systèmes et services des établissements d'enseignement supérieur (gestion de personnel, administration, scolarité, etc.),
- Étude sur le fonctionnement d'un SaaS,
- Mise en place de la partie 1 « Alléger les efforts de gestion de personnel »,
- Mise en place de la partie 2 « Soutenir la croissance de l'établissement »,
- Mise en place de la partie 3 « Stimuler la réussite des étudiants »,
- Mise en place de la partie 4 « Améliorer l'efficacité administrative »,
- Mise en œuvre du logiciel,
- Définition et mise en œuvre des services fournis selon la politique d'un SaaS,
- Test et déploiement.

Documentation

<http://www.unit4.com/fr/secteurs/education/student-management>

Encadreur(s)

- M. ANDRINIRINIAIMALAZA Fanambinantsoa Philibert,
- M. RAMANAN'HAJA Hery Tina,
- Encadreur Professionnel.

Lieu de travail

Laboratoire de Réseaux Informatiques et/ou Laboratoire d'Électronique Industrielle.

Remerciements

En préambule de ce travail, j'aimerais exprimer, par ces quelques lignes de remerciements, mes gratitudes envers tous ceux en qui, par leur présence, leur soutien, leur disponibilité et leurs conseils j'ai trouvé courage pour réaliser ce travail.

En premier lieu, j'adresse mes respectueuses considérations à mon encadreur professionnel : Dr. ANDRIAMIHARINJAKA Hasina, qui a été l'esprit pensant donnant naissance à ce sujet et qui m'a dirigée par ses précieux conseils tout en me laissant la liberté d'initiative pour la conception de la plateforme. A mes encadreurs qui sont M. ANDRINIRINIAIMALAZA Fanambinantsoa Philibert et M. RAMANAN'HAJA Hery Tina, pour avoir dirigés ce travail, pour leurs soucis et aussi pour leurs aides et leurs orientations qui m'ont permis de réaliser ce travail dans les meilleures conditions.

J'adresse mes remerciements anticipés et mes honorables considérations également aux membres de Jury qui vont évaluer et porter leur jugement à ce travail, aussi en enrichir le contenu par leurs précieuses propositions.

Je dois reconnaissance également à l'ensemble du corps enseignant de la mention STIC de leSPA (Ecole Supérieure Polytechnique d'Antsiranana).

Une mention particulière à ma famille, pour leurs soutiens et leurs attentions sans faille, dont les encouragements et l'amour.

Bref, tout ceux qui ont contribué de près ou de loin à l'accomplissement de ce travail.

RATSIZAFY Heriniaina Midonique

Table des matières

french

Remerciements	i
Table des matières	iii
Liste des figures	iv
Liste des tableaux	v
Liste des acronymes	vi
Glossaire	vii
Introduction générale	1
Chapitre 1 Études théoriques	3
1.1 Les établissements d'enseignement supérieur	4
1.2 Principes du LMD	5
1.3 Gestion des ressources humaines	12
1.4 Le logiciel en tant que service (SaaS)	15
Chapitre 2 Nature et définition des services fournis	18
2.1 Quand les ambitions deviennent réalité	19
2.2 Ambition : alléger les efforts de gestion de personnel	19
2.3 Ambition : soutenir la croissance de l'établissement	22
2.4 Ambition : stimuler la réussite des étudiants	24
2.5 Ambition : améliorer l'efficacité administrative	27
Chapitre 3 Modélisation statique et dynamique du système	30
3.1 Introduction	31
3.2 Modélisation dynamique du système	31
3.3 Modélisation de la base de données	43
3.4 Conclusion	50
Chapitre 4 Programmation et réalisation du logiciel	51
4.1 Introduction	52
4.2 Les Systèmes de Gestion de Base de Données (SGBD)	52
4.3 Les langages de programmation	53
4.4 Conception	55
Chapitre 5 Présentation et déploiement de l'application	59
5.1 Pré-requis	60

5.2	Présentation de l'application	60
5.3	Les services backend de Firebase	65
5.4	Déploiement de l'application sur Firebase	65
Conclusion générale		67
Bibliographie		I
Webographie		II
Annexes		III
A	Code TypeScript : Authentification	III
B	Upload de fichier	IV
C	Divers fonction de validation de formulaires	IV

french

Liste des figures

french

3.1	Diagramme de séquence : Authentification	36
3.2	Diagramme de séquence : Gestion des habilitations	37
3.3	Diagramme de séquence : Ajout d'utilisateur	38
3.4	Diagramme de séquence : Modification	39
3.5	Diagramme de séquence : Suppression	40
3.6	Diagramme de séquence : Demandes de congé / absence / avance	41
3.7	Diagramme de séquence : Envoi un e-mail	42
3.8	Le système de la base de données	43
3.9	Modèle Conceptuel de Données	48
3.10	Modèle Logique de Données	49
5.1	Fenêtre d'authentification	60
5.2	Fenêtre d'inscription	61
5.3	Email de réinitialisation de mot de passe	62
5.4	Email de validation de l'adresse email	62
5.5	Tableau de bord	63
5.6	Ajout d'utilisateur	64
5.7	Liste d'utilisateur	64
5.8	Page d'erreur 404	65

french

Liste des tableaux

french

1.1	Organisation des études mentions STIC, parcours	8
1.2	Organisation des études mentions Biologie, parcours	9
3.1	Dictionnaire de données	45

french

Liste des acronymes

api

bdd

crmcsscv

ecespa

godgrcgrh

html

jeejsfjsonjspjstl

lmd

mcdmldmvc

rh

saassgbdsisslasoaspa

ueuml

Glossaire

multiTenant

cloudComputinggamingOnDemandmiddleware

Introduction générale

De nouvelles dynamiques modifient rapidement l'économie de l'enseignement supérieur. L'élargissement des choix éducatifs, la globalisation et l'apparition de modèles agiles resserrent la concurrence des offres pour les étudiants tandis que les organismes d'accréditations publiques de leur côté, tiennent les institutions garantes de leur réussite. La pression de la scolarité, la réduction des budgets et les attentes plus élevées des différents acteurs poussent l'ensemble de l'écosystème à trouver une réponse adaptée à ce nouveau cadre économique.

Etudiants, corps professoral et personnel administratif attendent davantage de la révolution technologique dont ils font par ailleurs l'expérience dans leur vie pour remplacer les outils dépassés encore utilisés dans leur cadre scolaire et universitaire.

Ceci est encore plus vrai pour les systèmes d'information étudiants (Student Information Systems - SIS) qui remonte à une période bien antérieure à l'émergence des nouveaux modèles économiques de l'enseignement et des technologies disruptives d'aujourd'hui. Ces systèmes hérités perdurent uniquement grâce à des adaptations permanentes, qui rendent leur utilisation complexe et leur maintenance coûteuse.

Le Group4Success va bouleverser cet état de fait. Conçu pour répondre aux besoins évolutifs des institutions modernes, Group4Success simplifie et réduit vos coûts de support informatique, apporte à la fois à vos étudiants une utilisation mobile intuitive et vous soutient dans les initiatives stratégiques grâce toutes les données exploitables avec des possibilités d'analyses. Chaque acteur peut intervenir et agir à son niveau tout de suite et là où il se trouve. En d'autres termes, Group4Success permet d'augmenter notablement le retour sur investissement du remplacement de votre ancien système d'information. En bref, le Group4Success représente une alternative qui a tout pour changer la donne.

Ce présent travail de projet consiste à développer ledit "Group4Success" qui est une application SaaS (Software as a Service ou logiciel en tant que service) destiné pour alléger les efforts de gestion de personnel, soutenir la croissance de l'établissement, stimuler la

réussite des étudiants et améliorer l'efficacité administrative. Comme il s'agit d'un SaaS, l'application et infrastructure sous-jacente sont intégralement hébergées à distance et administrées par les soins de chaque établissement.

Par rapport à une application informatique classique, le SaaS est installé sur des serveurs distants plutôt que sur la machine de l'utilisateur. Les clients ne paient pas de licence d'utilisation pour une version, mais utilisent librement le service en ligne ou le plus généralement payent un abonnement. L'application demande une très grande protection des données, une grande rapidité et fluidité d'utilisation supportant un très grand nombre d'utilisateur ainsi qu'une très grande flexibilité avec possibilités d'extension (ajouts de modules complémentaires).

Pour s'approprier au mieux ce projet, il est essentiel, dans un premier temps de faire une étude approfondie afin de cerner les besoins techniques de l'application. En commençant par l'étude théorique sur les systèmes et services des établissements d'enseignement supérieur et l'étude sur le fonctionnement d'un SaaS.

La deuxième partie va se consacrer sur la mise en place et définition des services fournis par le Groupe4Success. Cette partie sera subdivisée en quatre dont premièrement pour "alléger les efforts de gestion de personnel", deuxièmement pour "soutenir la croissance de l'établissement", troisièmement pour "stimuler la réussite des étudiants" et enfin quatrièmement pour "améliorer l'efficacité administrative".

Dans la troisième partie, nous allons entamer à la phase d'analyse des besoins et du domaine du projet ainsi que la modélisation dynamique et statique du système. Issue des études théoriques, cette partie va décrire de manière visuelle et graphique les besoins et, les solutions fonctionnelles et techniques de notre projet.

Puis dans la quatrième partie nous verrons les détails de la programmation, la conception du logiciel ainsi que les choix des langages utilisés. Enfin la dernière partie se concentre uniquement sur la présentation, le test et le déploiement de l'application ainsi développée. Puis définir la mise en œuvre des services fournis selon la politique d'un SaaS.

Chapitre 1

Études théoriques

Sommaire

1.1 Les établissements d'enseignement supérieur	4
1.1.1 Introduction	4
1.1.2 Objectifs	4
1.1.3 La formation initiale et continue	5
1.2 Principes du LMD	5
1.2.1 Terminologie	6
1.2.2 Domaines de formation	6
1.2.3 Les mentions	7
1.2.4 Les parcours	8
1.2.5 Les crédits	9
1.2.6 L'Unité d'Enseignement (UE)	10
1.2.7 L'Elément Constitutifs (EC)	11
1.2.8 Les avantages du systeme LMD	11
1.3 Gestion des ressources humaines	12
1.3.1 Définition	12
1.3.2 Gestion des carrières	13
1.3.3 Recrutement ou gestion des postes	13
1.3.4 Gestion des candidats et leur sélection	13
1.3.5 Gestion de la correspondance	14
1.3.6 Gestion des supports	14
1.3.7 La gestion de la paie et des rémunérations	14
1.4 Le logiciel en tant que service (SaaS)	15
1.4.1 Définition	15
1.4.2 Fonctionnement	15
1.4.3 Les avantages	16
1.4.4 Les inconvénients	17

1.1 Les établissements d'enseignement supérieur

1.1.1 Introduction

L'enseignement supérieur désigne généralement l'éducation dispensée par les universités [9], avec un système plus dual de grandes écoles et d'autres institutions décernant des grades universitaires ou autres diplômes de l'enseignement supérieur.

Ces études visent à acquérir un niveau "supérieur" de compétences, généralement via une inscription ou concours d'entrée, un cursus ponctués par des examens.

Ces études se déroulent souvent autour de campus, dans un système public ou privé selon les cas, et souvent catégorisés en « sciences dures », « sciences de l'ingénieur » et « sciences humaines et sociales ».

Les types de diplômes évoluent avec le temps, mais tendent à s'homogénéiser aux niveaux européens et internationaux avec les licences, des masters, et des doctorats pour faciliter les systèmes d'équivalence ou de reconnaissance mutuelle de diplômes.

Cette formation inclut des cours et des stages, et peut également comporter des participations à la recherche scientifique (notamment au niveau du doctorat) et intégrer de la formation continue, mais accepte souvent aussi des « auditeurs libres » qui viennent simplement accroître leur culture générale ou spécialisée.

1.1.2 Objectifs

Les établissements de l'enseignement supérieur contribuent :

- au développement de la recherche, support nécessaire des formations dispensées, et à l'élévation du niveau scientifique, culturel et professionnel de la nation et des individus qui la composent ;
- à la croissance régionale et nationale dans le cadre de la planification, à l'essor économique et à la réalisation d'une politique de l'emploi prenant en compte les besoins actuels et leur évolution prévisible ;
- à la réduction des inégalités sociales ou culturelles et à la réalisation de l'égalité entre les hommes et les femmes en assurant à toutes celles et à tous ceux qui en ont la volonté et la capacité l'accès aux formes les plus élevées de la culture et de la recherche ;
- à la construction de l'espace européen de la recherche et de l'enseignement supérieur.

Les missions du service public de l'enseignement supérieur sont :

- La formation initiale et continue ;
- La recherche scientifique et technologique, la diffusion et la valorisation de ses résultats ;
- L'orientation et l'insertion professionnelle ;
- La diffusion de la culture et l'information scientifique et technique ;
- La participation à la construction de l'Espace européen de l'enseignement supérieur et de la recherche ;
- La coopération internationale.

1.1.3 La formation initiale et continue

Le service public de l'enseignement supérieur :

- offre des formations à la fois scientifiques, culturelles et professionnelles,
- accueille les étudiants et concourt à leur réussite et à leur orientation,
- dispense la formation initiale,
- participe à la formation tout au long de la vie,
- assure la formation des formateurs.

L'orientation des étudiants comporte une information sur le déroulement des études, sur les débouchés, sur les passages possibles d'une formation à une autre.

La formation continue s'adresse à toutes les personnes engagées ou non dans la vie active. Organisée pour répondre à des besoins individuels ou collectifs, elle inclut l'ouverture aux adultes des cycles d'études de formation initiale, ainsi que l'organisation de formations professionnelles ou à caractère culturel particulières.

Le service public de l'enseignement supérieur met à disposition de ses usagers des services et des ressources pédagogiques numériques.

1.2 Principes du LMD

Le LMD (Licence-Master-Doctorat) désigne un ensemble de mesures modifiant le système d'enseignement supérieur pour l'adapter aux standards européens. Elle met en place principalement une architecture basée sur trois grades : licence, master et doctorat ; une organisation des enseignements en semestres et unités d'enseignement [10]. Le système LMD devient donc un système international d'harmonisation des cursus et des diplômes, favorisant la mobilité internationale.

Le passage au système se fait progressivement et nécessite une démarche concertée entre les différents acteurs des universités (enseignants-chercheurs, étudiants, structures pédagogiques des universités et des établissements et les représentants de l'environnement socio-économique et culturel). A Madagascar, il est tant convoité par les établissements supérieurs, mais il reste un parcours encore difficile à maîtriser.

En effet, le système présente plusieurs avantages surtout pour les étudiants, à ne citer qu'il offre à celui-ci la possibilité de restructurer son parcours en cours de formation tout en ayant la possibilité de s'insérer dans le monde professionnel avec n'importe quel niveau de formation. Bien sur, les diplômes sont reconnus à l'international.

1.2.1 Terminologie

Le système LMD est fondée sur trois grades de références, mais désigne aussi trois cycles de formation et trois diplômes nationaux :

- L = Licence (Bac + 3) \Rightarrow L1, L2, L3 \Rightarrow 6 semestres : S1, S2, S3, S4, S5, S6
- M = Master (Bac + 5) \Rightarrow M1, M2 \Rightarrow 4 semestres : S7, S8, S9, S10
- D = Doctorat (Bac + 8)
- La Licence et le Master peuvent être à vocation Académique, Recherche ou Professionnelle.
- Dans le système LMD, un grade universitaire (Licence, Master, Doctorat) sanctionne la fin d'un cycle.
- Au sein des cycles Licence et Master, les formations sont organisées en semestre.

1.2.2 Domaines de formation

Les domaines constituent le cadre général de l'offre de formation de l'établissement. Ils doivent ainsi représenter des ensembles cohérent fédérant les grands champs de compétences pédagogiques et scientifiques de l'établissement [5].

Madagascar à 6 domaines de formation dont :

- SCIENCES ET TECHNOLOGIE pour la faculté des Sciences
- ARTS, LETTRES et SCIENCES HUMAINES (ALSH) pour la faculté des Lettres et Sciences Humaines (FLSH)
- SCIENCES DE LA SOCIÉTÉ pour la faculté de Droit, d'Economie, de Gestion et de Sociologie (DEGS)
- SCIENCES DE LA SANTÉ pour la faculté de Médecine

- SCIENCES DE L'INGENIEUR pour l'Ecole Supérieure Polytechnique (ESP)
- SCIENCES DE L'ÉDUCATION pour l'Ecole Normale Supérieure (ENS) et l'Ecole Normale Supérieure pour l'Enseignement Technique (ENSET)

1.2.3 Les mentions

Le domaine de formation est structuré en plusieurs Mentions. La Mention couvre un champ scientifique relativement large qui permet d'identifier le thème majeur de la formation. Elle permet de faire apparaître la finalité soit académique, recherche soit professionnelle.

Exemple 1

Domaine : SCIENCES DE L'INGÉNIEUR

Mentions (grades Licence et Master) :

- Génie Mécanique
- Hydraulique et Énergétique
- Génie Civil
- Science et Technologie de l'Information et de la Communication
- Génie Électrique

Exemple 2

Domaine : SCIENCES ET TECHNOLOGIE

Mentions (grades Licence et Master) :

- Chimie
- Physique et Application
- Biologie
- Geosysteme et Évolution
- Mathématique et Informatique

1.2.4 Les parcours

L'offre de formation est organisée sous forme de parcours diversifiés et adaptés. Un parcours de formation est un ensemble cohérent d'Unités d'Enseignements (UE) articulées entre elles de façon à offrir :

- D'une part, une progression pédagogique adaptée en fonction de l'origine de l'étudiant et de son projet personnel,

- D'autre part, des possibilités de réorientation ou de complément de formation à chacun des paliers.

Exemple 1

Domaine : SCIENCES DE L'INGÉNIEUR

Mentions (grades Licence et Master) : SCIENCES ET TECHNOLOGIE DE L'INFORMATION ET DE LA COMMUNICATION

TABLEAU 1.1 – Organisation des études mentions STIC, parcours

Semestres	Parcours	
S10, S9, S8	Électronique et Informatique Industrielle (STIC)	Télécommunication et Réseaux (TR)
S7	Tronc commun STIC	
S6, S5, S4, S3	STIC	
S2, S1	Tronc Commun Polytechnique	

Exemple 2Domaine : SCIENCES ET TECHNOLOGIEMentions (grades Licence) : BIOLOGIE

TABLEAU 1.2 – Organisation des études mentions Biologie, parcours

Semestres	Parcours				
S6, S5	Biologie des Organismes et des Ecosystèmes (BOE)	Biochimie et Biologie Moléculaire (BBM)	Entomologie Appliquée (ENTAP)	Physiologie, Pharmacologie, Cosmétologie (PPC)	Anthropologie Biologique et Evolution (ABE)
S4, S3	Tronc commun Biologie(TCB)				
S2, S1	Portail Sciences de la vie et de la terre (PSVT)				

1.2.5 Les crédits

Un crédit est une unité de mesure, une unité de compte, exprimant la valeur donnée à une Unité d'Enseignement ou à un Elément Constitutif (EC) d'une UE. Le système de crédit est applicable à toutes les activités d'enseignement y compris stages, mémoires, projets, travail personnel et aussi à toutes les formes d'enseignement (présentiel, ouvert, à distance, en ligne,...).

Au sein des cycles Licence et Master, les formations sont organisées en semestres et qu'un crédit équivaut à 10 heures de travail (travail présentiel, travail personnel) :

- Offres de formation dans le grade LICENCE :
 - Durée d'études : 06 semestres
 - Total des crédits : 180 dont 30 crédits par semestre
- Offres de formation dans le grade MASTER :
 - Durée d'études : 04 semestres
 - Total des crédits : 120 dont 30 crédits par semestre

Exemple

Pour l'Ecole Supérieure Polytechnique :

- Domaine : SCIENCES DE L'INGÉNIEUR

- Grade : MASTER
- Mention : GÉNIE ÉLECTRIQUE
- Parcours en M2 de S8 : Production, Transport et Distribution d’Energie
- Crédit : 30 (répartis dans les types d’intervention « Cours, TD, TP, Stage, etc. » des UE)

1.2.6 L’Unité d’Enseignement (UE)

L’Unité d’Enseignement (UE) est la base du dispositif LMD. En effet, toutes les études sont organisées en unités d’enseignement.

Une UE peut être une matière ou un ensemble de matières choisies pour leur cohérence dans cet ensemble.

Différentes catégories d’UE

- UE fondamentales qui sont obligatoires et doivent être suivies par tous les étudiants
- UE complémentaires qui complètent les UE fondamentales du parcours choisi par l’étudiant et doivent être prises obligatoirement afin de valider le parcours de l’étudiant.
- UE libre qui sont au choix.

Les UE peuvent être donc :

1. Obligatoires ou optionnelles (facultatives)
2. Transférables d’un parcours à l’autre
3. Capitalisables puisque toute validation d’UE est acquise quelle que soit la durée d’un parcours.

Exemple

En Faculté des Sciences :

- Domaine : SCIENCES ET TECHNOLOGIE
- Grade : LICENCE
- Mention : CHIMIE
- Parcours en L1 de S2 : Formation de base en chimie (Tronc Commun)
- Nombre de UE dans ce parcours L1/S2 : 06
 - Chimie organique physique \Rightarrow 5 crédits
 - Initiation à la physico-chimie des solutions aqueuses \Rightarrow 5 crédits
 - Chimie inorganique \Rightarrow 5 crédits

- Initiation à la géométrie et éléments d'analyse \Rightarrow 4 crédits
- Physique 1 \Rightarrow 4 crédits
- Initiation à la Science de la Vie et de la Terre \Rightarrow 7 crédits

Pour chaque UE les crédits sont répartis dans des types d'intervention « Cours, TD, TP, Stage, etc. » et le total de crédits pour cette UE est égal à 30.

1.2.7 L'Elément Constitutifs (EC)

Chaque contenu d'une UE est appelé Elément Constitutif (EC).

Chaque EC de l'UE appelé aussi matière ou composante de l'UE est évalué selon un examen semestriel.

1.2.8 Les avantages du systeme LMD

Le système LMD comporte une série d'avantages incontestables qui ont fondé son acceptation et son adoption à l'échelle mondiale. Leur perception varie selon le contexte, les réalités dans les pays qui ont déjà adopté le LMD ou qui sont en voie de le faire.

Exemple

1. Pour l'apprenant :

- Réduction de la durée d'études liée à la semestrialisation
- Possibilité de poursuivre des études dans une autre université ou dans un pays appliquant le LMD
- Possibilité d'interrompre sa formation et de la reprendre par la suite en conservant le bénéfice des crédits qu'il a validés auparavant
- Professionnalisation facilitée par les UE, stages, projets
- Diversité des parcours proposés grâce à un système de passerelles
- Choix de son parcours de formation en fonction de ses projets d'études et professionnels, etc...

2. Pour l'institution :

- Une offre de formation moderne, attractive et compétitive
- La possibilité de modifier plus facilement l'offre de formation en fonction des besoins et des valorisations de la demande (ajouter, changer le statut d'un module,...)

- Un encadrement rapproché qui implique des effectifs réduits et un contrôle continu des connaissances.
- La professionnalisation des enseignements oblige universités à être d'avantage attentives dans leur fonctionnement pédagogique aux exigences du marché du travail (profils techniques pour les étudiants) et pour les emplois à pourvoir, etc...

3. Pour la société :

- Ouverture de l'Université sur la société
- Augmentation du nombre de métiers
- Adaptation des formations aux besoins sociaux
- Meilleure participation de l'Université au développement de la société

1.3 Gestion des ressources humaines

La gestion des ressources humaines est plus que jamais un pilier de la performance de l'établissement supérieur. Les questions de gestion de compétences, de recrutement, de rémunération ou même d'implication de ses collaborateurs se positionnent au centre des préoccupations des employeurs. Pour définir simplement la notion des ressources humaines, on peut dire qu'il s'agit d'un service piloté par un Directeur des ressources humaines qui est en charge de plusieurs services. Il s'occupe par exemple de la gestion du personnel, des administrations, de la communication.

1.3.1 Définition

La Gestion des Ressources Humaines (GRH) est un sous système du management d'une entreprise ; elle est la facette humaine qui permet d'organiser le travail et de traiter les travailleurs de manière qu'ils puissent faire valoir autant que possible leur capacité à fin d'obtenir un rendement maximal pour eux-mêmes et pour leur groupe [11].

C'est l'ensemble de mesures et d'activités impliquant des ressources humaines et ayant pour objectif d'améliorer l'efficacité et la performance des individus et de l'organisation. C'est aussi un ensemble de pratiques ayant pour objectif de mobiliser et de développer les ressources humaines pour une plus grande efficacité et efficience, en soutien de la stratégie d'une organisation (association, entreprise, administration).

La GRH est une dimension de la gestion de l'entreprise, elle existe le plus souvent en tant que fonction à part entière. Même si des tendances récentes la révèlent sous un angle

de plus en plus dispersé et partagé, elle a un certain nombre de mission à accomplir et une série d'activités à organiser et à gérer, c'est ce que nous allons voir dans les sections suivantes [1].

1.3.2 Gestion des carrières

La gestion des carrières est l'ensemble du cheminement professionnel de l'individu qui va s'étendre durant la totalité de sa vie active au sein d'une organisation, elle permet de gérer de nombreux domaines :

- ✓ Le recrutement.
- ✓ La sélection.
- ✓ La formation.
- ✓ La motivation et l'implication du personnel.
- ✓ La gestion de la paie et des rémunérations.
- ✓ Les relations sociales.
- ✓ Les conditions de travail.
- ✓ L'évaluation des performances et compétences.

1.3.3 Recrutement ou gestion des postes

Une vision globale des postes :

Les caractéristiques du poste à pourvoir sont gérées : description, responsable, lieu, mission, début d'activité, type de contrat, etc...

- ✓ Le recrutement.
- ✓ Recherche multicritères sur les postes à pourvoir paramétrable.

1.3.4 Gestion des candidats et leur sélection

Vivier de candidat : dossier personne, CV, disponibilité, rémunération souhaitée, origine.

Le contenu du dossier individuel est paramétrable avec le générateur de formulaires intégré.

Impression des listes et fiches des candidats : elle est entièrement personnalisable grâce au générateur de rapports intégré.

Documents : en plus du CV saisi, il est possible d'attacher au candidat des fichiers (Word, Excel, PDF, etc...) pour stocker le CV, diplômes, certificats, etc.

1.3.5 Gestion de la correspondance

Automatisation de la correspondance : envoi d'email ou impression de courrier Word, paramétrable à partir de modèles et des statuts des candidatures.

Archivage de la correspondance dans le dossier du candidat.

Automatisation du processus administratif : Mise à jour automatique des status du candidat selon l'évolution de son dossier.

1.3.6 Gestion des supports

Gérer les supports tout naturellement

- ✓ Gestion des catalogues regroupant les cours destinés à une catégorie de personnes et pour une période donnée.
- ✓ Gestion des tableaux de bord de l'activité formation pour le management et pour les RH
- ✓ Gestion des saisies du budget de formation par service et suivi du solde, détaillé ou consolidé.

1.3.7 La gestion de la paie et des rémunérations

La gestion de la paie constitue un aspect important de l'administration des salariés. Matérialisant la relation entre l'employeur et le salarié. Le salaire, contrepartie de la prestation du travail effectuée par le salarié, constitue un coût pour l'établissement.

La fixation de la rémunération prend en compte la complexité des tâches à effectuer, mais également les conditions de travail.

1.4 Le logiciel en tant que service (SaaS)

Le SaaS (Software as a Service) est un concept assez récent qui permet aux entreprises de s'abonner à un logiciel à distance au lieu de les acquérir et de devoir les installer sur leur propre matériel informatique.

1.4.1 Définition

Le logiciel en tant que service ou Software as a Service (SaaS) [13] est un modèle d'exploitation commerciale des logiciels dans lequel ceux-ci sont installés sur des serveurs distants plutôt que sur la machine de l'utilisateur. Les clients ne paient pas de licence d'utilisation pour une version, mais utilisent librement le service en ligne ou le plus généralement payent un abonnement.

Le logiciel en tant que service (SaaS) est donc la livraison conjointe de moyens, de services et d'expertise qui permettent aux entreprises d'externaliser intégralement un aspect de leur système d'information (messagerie, sécurité,...) et de l'assimiler à un coût de fonctionnement plutôt qu'à un investissement. Le contrat de services est essentiel pour définir le niveau de qualité de service (SLA). Le logiciel en tant que service (SaaS) peut être vu comme l'équivalent commercial de l'architecture orientée service (SOA).

Les solutions logicielles en tant que service (SaaS) sont principalement développées à destination d'entreprises. Depuis quelques années le marché des SaaS est en très forte croissance. Ces logiciels présentent pour les entreprises divers avantages et inconvénients.

1.4.2 Fonctionnement

Dans ce type de service, des applications sont mises à la disposition des consommateurs. Les applications peuvent être manipulées à l'aide d'un navigateur web ou installées de façon locale sur un PC, et le consommateur n'a pas à se soucier d'effectuer des mises à jour, d'ajouter des patches de sécurité et d'assurer la disponibilité du service.

Le SaaS permet à une entreprise de ne plus installer d'applications sur ses propres serveurs mais de s'abonner à des logiciels en ligne et de payer un prix qui variera en fonction de leurs utilisateurs effectives [14].

En utilisant le SaaS, l'entreprise n'hébergera pas ses applications et ne stockera pas ses données en interne.

Il n'est donc pas nécessaire d'acquérir directement ces applications et posséder des serveurs pour les héberger. De plus, la maintenance et les mises à jour des applications seront gérées en externe par le prestataire.

Les utilisateurs de l'entreprise devront simplement disposer d'un ordinateur et des codes d'accès au service en ligne pour pouvoir travailler.

1.4.3 Les avantages

La gestion en mode SaaS permet à une entreprise de bénéficier de nombreux avantages :

- Aucun logiciel à installer sur les matériels informatique de l'entreprise,
- Pas de données stockées en interne,
- Mise à jour des applications automatiques,
- L'application peut être utilisée partout et n'importe quand : il suffit d'une simple connexion internet et d'un navigateur web.

Les deux premiers avantages cités en recourant au mode SaaS permettant en plus à l'entreprise de réaliser des économies sur ces investissements en matériel informatique.

L'utilisation de solutions logicielles en tant que service (SaaS) en entreprise permet aussi un meilleur contrôle des charges techniques. L'ensemble des solutions techniques étant délocalisées le coût devient fixe, généralement fonction du nombre de personnes utilisant la solution SaaS. Le prix par utilisateur englobe le coût des licences des logiciels, de la maintenance et de l'infrastructure. Il revient à l'entreprise utilisatrice de faire son choix entre utilisation en SaaS, d'une part, et acquisition des licences puis déploiement en interne, d'autre part.

Le mode SaaS est un service, dont le coût constitue une charge immédiatement déductible du résultat de l'entreprise, alors qu'un investissement est déduit du résultat par le biais d'amortissements étalés sur plusieurs années.

Un avantage manifeste pour les entreprises est la rapidité de déploiement lorsque le logiciel SaaS correspond exactement au besoin (et qu'il ne nécessite aucune adaptation). Les solutions SaaS étant déjà pré-existantes le temps de déploiement est extrêmement faible.

Un autre avantage pourrait être de réduire la consommation électrique en permettant la mutualisation des ressources sur des serveurs partagés par plusieurs entreprises

(architecture Multi-tenant) ainsi que l'usage d'un ordinateur à faible consommation muni d'un simple navigateur Web sans autres licences associées.

1.4.4 Les inconvénients

Le mode SaaS présente toutefois quelques inconvénients. Lors de la mise en place de solutions SaaS, les données relatives à l'entreprise cliente sont, généralement, stockées sur les serveurs du prestataire fournissant la solution. Lorsqu'il s'agit de données sensibles ou confidentielles, l'entreprise est obligée de prendre des dispositions contractuelles avec le fournisseur. Ainsi l'entreprise est dépendante du prestataire qui lui fournit le service. Que se passe-t-il si ce dernier vous fait défaut ? Dépose le bilan ? ou si un litige survient ?...

La délocalisation des serveurs de la solution SaaS permet également un accès nomade aux données de l'entreprise. Cet accès entraîne un souci de sécurité de l'information lors du départ de collaborateurs. Il est indispensable d'avoir mis en place des procédures permettant, lors d'un départ, de supprimer l'habilitation de l'ancien collaborateur à accéder aux données de l'entreprise.

Les migrations informatiques peuvent être compliquées puisqu'il faut basculer les données de la plate-forme d'un fournisseur vers celle d'un autre, avec divers problèmes associés (compatibilité, apparence pour le client, etc.). Dans le cadre du SaaS, le client se trouve lié à son fournisseur.

Chapitre 2

Nature et définition des services fournis

Sommaire

2.1	Quand les ambitions deviennent réalité	19
2.2	Ambition : alléger les efforts de gestion de personnel	19
2.2.1	Mieux connaître et contrôler les absences	19
2.2.2	Maîtriser les emplois du temps	20
2.2.3	Organiser la gestion de la paie	20
2.2.4	Group4Success apporte une réponse	20
2.2.5	Déduction des tâches-clés pour satisfaire l'ambition	21
2.3	Ambition : soutenir la croissance de l'établissement	22
2.3.1	Augmenter le nombre d'inscriptions	22
2.3.2	Répondre aux besoins d'apprenants	22
2.3.3	Améliorer la collecte de fonds	23
2.3.4	Group4Success apporte une réponse	23
2.3.5	Les tâches-clés pour satisfaire l'ambition	23
2.4	Ambition : stimuler la réussite des étudiants	24
2.4.1	Remédier efficacement au risque d'échec	24
2.4.2	Clarifier les exigences de niveau	24
2.4.3	Améliorer l'expérience étudiant	25
2.4.4	Group4Success apporte une réponse	25
2.4.5	Synthèse des tâches-clés pour satisfaire l'ambition	26
2.5	Ambition : améliorer l'efficacité administrative	27
2.5.1	Automatiser les procédures courantes	27
2.5.2	Soutenir la prise de décision sur données	27
2.5.3	Optimiser les ressources humaines et financières	27
2.5.4	Group4Success apporte une réponse	28
2.5.5	Aperçu des tâches-clés pour satisfaire l'ambition	28

2.1 Quand les ambitions deviennent réalité

Dans le secteur de l'éducation supérieur, devenu de plus en plus concurrentiel, le succès des organisations/établissements repose sur quatre ambitions fondamentales : stimuler la réussite des étudiants, soutenir la croissance de l'établissement, améliorer l'efficacité administrative et alléger les efforts de gestion de personnel.

Le succès dans chacun de ces domaines est important mais les institutions doivent exceller dans chacun d'eux. Une telle réussite dépend de solutions technologiques modernes et complètes, capables d'intégrer immédiatement, améliorations, innovations et ruptures technologiques.

La solution Grou4Success permet de répondre aux ambitions des établissements de manière spécifique et pour toute leur organisation :

- Mise à jour aisée et rapide des stratégies et procédures sans intervention technique coûteuse.
- Implémentations plus efficaces, permettant de dédier les ressources à l'excellence institutionnelle plutôt qu'à la maintenance du système.
- Expérience utilisateur accrue, combinée avec des outils de support libérant les équipes technique et administratives pour les tâches à plus forte valeur ajoutée.
- Disponible partout et tout le temps, sur PC, Mac et tablette.
- Un tableau de bord orienté rôles qui affiche les chiffres clés et une vue d'ensemble simple et directe des anomalies.

2.2 Ambition : alléger les efforts de gestion de personnel

Au sein d'une entreprise, la gestion du personnel demande la prise en charge, au quotidien, de l'ensemble des tâches administratives liées au personnel, de la paie des salariés, du développement et de tout ce qui est lié aux ressources humaines. Pour soutenir la réussite dans ces efforts, trois domaines d'action sont à privilégier.

2.2.1 Mieux connaître et contrôler les absences

Outre les congés légaux ou conventionnels, rémunérés ou non, qui sont généralement prévisibles et donc plus faciles à anticiper, un certain nombre d'événements peuvent générer une absence plus ou moins longue du salarié.

La plus délicate d'entre elles est certainement l'absence injustifiée. Il peut s'agir d'une absence sans autorisation préalable de l'employeur, ou consécutive à son refus d'autorisation, ou d'une prolongation sans autorisation d'une absence initialement autorisée.

En cas de maladie, il doit lui transmettre un certificat médical, dans un délai fixé par convention collective, règlement intérieur ou usage, et est le plus souvent de 48 heures. Il s'agit là du délai admis pour justifier l'absence, ce qui n'empêche pas le salarié à en informer l'employeur beaucoup plus rapidement.

2.2.2 Maîtriser les emplois du temps

Il apparaît aujourd'hui que le temps est devenu une variables stratégique pour les entreprises. Bien gérer le temps de travail permet de gagner en efficacité et en productivité. Le temps est donc une ressources rare et non renouvelable. Avec l'entreprise, l'expression « *le temps, c'est de l'argent* » prend alors toute son importance. Le temps doit donc être géré, contrôlé, planifié.

2.2.3 Organiser la gestion de la paie

La gestion de la paie constitue un aspect important de la gestion des personnels. Matérialisant la relation entre employeur et le salarié, le bulletin de paie est un document périodique obligatoire devant répondre à certaines règles. Le salaire, contrepartie de la prestation du travail effectuée par le salarié, constitue un coût pour l'entreprise qui doit être comptabilisé.

Les heures complémentaires sont les heures que l'employeur peut demander au salarié à temps partiel d'effectuer au-delà de la durée fixée par le contrat de travail, ils doivent être évaluées afin de calculer le montant de la rémunération de l'employé.

2.2.4 Group4Success apporte une réponse

Group4Success fournit des outils de gestion du personnel qui gère toute l'administration des ressources humaines d'une entreprise, c'est à dire que c'est lui qui s'occupe des recrutements, contrats, fiches de paie, dossiers des salariés, et autres documents règlementaires comme les cotisations sociales par exemple. Sa tâche est aussi de vérifier la bonne adaptation des salariées à leur milieu professionnel, contrôler leurs savoirs et compétences sur le terrain, s'assurer du respects des règles et des horaires.

L'automatisation des procédures d'absence accélère et simplifie grandement l'enregistrement et la gestion des absences, générant par conséquent un fort gain d'efficacité. D'un côté, les utilisateurs reçoivent des alertes proactives pour garder la trace des tâches qui leur sont assignés ; quant au système, il crée la représentation la plus claire possible des absences en rassemblant tous les types de demandes d'absences et leurs mises à jour, dans un dossier unique et complet pour chaque employé.

A chaque salarié concerné le contrôle de son emploi du temps, avec droits de modification sur son calendrier personnel qui affiche une vue d'ensemble des tâches prévues. Il peut ainsi enregistrer lui-même ses demandes de congés, d'absence et les demandes d'avance ou encore mettre à jour ses activités planifiées ou y apporter des changements.

2.2.5 Déduction des tâches-clés pour satisfaire l'ambition

Données générales

- Création des fiches employeur : nom, adresse, contact, salaire, date d'embauche, heure de travail.
- Paramètres légaux et conventionnels.
- La gestion des contrats de prévoyance.

Gestionnaire RH

- La gestion des emplois.
- La gestion de paie : journal de paie, formule de calcul, prime d'ancienneté, 13^{ème} mois et les avances.
- Traitement de masse : augmentation générales, recrutement, évaluation des entretiens d'embauches, contrat d'embauche, la gestion des requête pour approuver les demandes d'absences et congés.
- Visualisation, analyses, éditions, contrôles des personnelles et exportation des données.
- Suivi des évènement : visites médicales, affectation de matériels.

Personnelles

- Demande d'absence.
- Demande de congés.

- Demande d'avance.

2.3 Ambition : soutenir la croissance de l'établissement

De nombreuses écoles de l'enseignement supérieur ainsi que des universités font face à la dure réalité de la diminution de leurs ressources, menaçant leur existence même. La situation n'est certes pas aussi sombre pour toutes, mais ne pas anticiper un risque de perte de revenus peut mettre en péril la survie d'une institution. Pour la plupart, cela signifie qu'il faut pouvoir tout à la fois, maintenir ou augmenter le nombre d'inscriptions, offrir de nouveaux programmes flexibles à destination d'une population croissante d'apprenants "non traditionnels", ou encore lever des fonds auprès de donateurs ou d'anciens élèves. Chaque méthode apporte ses propres défis à surmonter [15].

2.3.1 Augmenter le nombre d'inscriptions

La baisse de nombre d'inscriptions d'étudiants a un impact financier sérieux sur de nombreuses institutions de l'enseignement supérieur. Pour y faire face, elles ne peuvent que se lancer dans une compétition effrénée.

Dans le même temps, les étudiants eux-mêmes sont devenus plus avertis de ces pratiques pour les recruter. Pour peser sur leur choix, toute stratégie de marketing doit s'appuyer sur des données provenant de systèmes de gestion de la relation clients performants. Ceci implique souvent l'utilisation de systèmes tiers dans l'écosystème de l'institution, induisant un surcoût.

2.3.2 Répondre aux besoins d'apprenants

L'époque d'une seule et même stratégie d'enseignement destinées à de jeunes adultes étudiant à temps plein est révolue : il s'agit maintenant de répondre aux besoins d'élèves souvent plus âgés qui suivent des cursus à temps partiel.

Les systèmes d'information destinés aux étudiants conçus avant ce nouveau phénomène d'apprentissage non-traditionnel ne sont pas prévus pour répondre à ces exigences d'organisation très spécifiques, en termes de recrutement ou de programmes. De la même façon, les applications financières ne sont plus adaptées pour gérer la rentabilité des offres d'enseignement traditionnel ou non traditionnel.

2.3.3 Améliorer la collecte de fonds

L'utilisation de procédés dépassés influe considérablement sur l'efficacité de la création et de la gestion des campagnes de collecte de fonds : augmenter le niveau des donations devient alors encore plus difficile.

De la même façon, le fait de considérer donateurs et anciens élèves séparément sans tenir compte de leur historique avec l'institution restreint les possibilités de rapprochements utiles, qui pourtant pourraient être source de message ciblés impactant.

2.3.4 Group4Success apporte une réponse

Group4 Success Management fournit les outils nécessaires pour interagir efficacement dans la durée avec chaque partie prenante : personnels, étudiants, anciens élèves, partenaires extérieurs et mécènes.

Les workflows configurables via un simple « drag & drop » permettant d'automatiser les campagnes de sensibilisation en fonction du comportement des prospects, des anciens élèves, ou des donateurs. Le même CRM (Customer Relationship Management) étant disponible pour tout utilisateur de l'institution, nous l'appelons le « CRM pour tous », qui donne une vue à 360° sur chaque individu durant sa progression en tant que candidat potentiel, étudiant, diplômé, ancien élève. Instituant ainsi une relation qui peut durer plusieurs décennies.

Group4 Success Management répond également aux besoins des apprenants non traditionnels d'aujourd'hui : sa totale flexibilité dans la gestion des inscriptions, sans contrainte de cours ou de programmes, lève la barrière de la procédure classique de recrutement trimestriel.

L'application permet aussi de faire des analyses exhaustives sur l'ensemble des données, nous pouvons déterminer la rentabilité de chaque programme ou département et prendre les décisions relatives à l'expansion des programmes d'enseignement.

2.3.5 Les tâches-clés pour satisfaire l'ambition

Des modules spécifiques sont disponibles pour gérer les tâches-clés pour satisfaire l'ambition. D'après ce qu'on a mentionné ci-dessus, la procédure se déroule comme suit :

- Améliorer la gestion des inscriptions en établissant des critères propres selon les besoins de l'établissement.

- Définir des campagnes de sensibilisation pour mieux avertir les étudiants au recrutements.
- Faire des suivis de l'institution.
- Offrir des nouveaux programme flexibles.
- Obtenir une vue globale des anciens élèves et donateurs pour tirer profit de le leurs relations avec les départements et les groupes auxquels ils appartiennent.
- Lancer des actions permettant de générer des revenus complémentaires comme la gestion des locaux, le logement ou les programmes de formation continue.

2.4 Ambition : stimuler la réussite des étudiants

Les institutions s'investissent spontanément dans la réussite de leurs étudiants et les stimulent pour réussir et en contre partie, les risques consécutifs à l'échec n'ont jamais été aussi forts. Le choix des étudiants pour une école dépend aussi du soutien que l'on va lui proposer, de la valeur et des opportunités d'employabilité de l'institution. Cette pression est également accentuées par les régulateurs, les organismes d'accréditation et les sources gouvernementales de financement, qui tiennent les institutions garantes du progrès des étudiants.

Trois domaines clés doivent être considérés pour amener l'ensemble des étudiants à la réussite attendue.

2.4.1 Remédier efficacement au risque d'échec

L'identification et l'intervention en cas de risque d'échec forment une composante fondamentale de notre stratégie de réussite étudiante. Comment identifier précisément les individus ayant la plus forte probabilité d'échec académique ?

Déterminer les catégories d'étudiants en risque d'échec est un premier pas, mais le réel tournant est de pouvoir identifier ces personnes. Il est fort probable que notre système d'information étudiant ne soit pas doté d'une telle capacité en amont.

2.4.2 Clarifier les exigences de niveau

Les étudiants sont censés achever leur cursus dans un laps de temps spécifique. Qu'il s'agisse de deux ans, quatre ans, ou de toute autre durée, ne pas en respecter le terme

entraîne des frais d'études supplémentaires et retarde le bénéfice de l'obtention du diplôme.

Il est par conséquent regrettable que tant d'étudiants fassent des choix inadaptés, faute de savoir précisément à quoi s'attendre lors du parcours retenu. Cela résulte fatalement en un allongement de la durée d'obtention des diplômes ou, bien pire, en l'abandon pur et simple des études.

2.4.3 Améliorer l'expérience étudiant

Les systèmes d'information étudiants ont eu jusqu'à présent une fonction secondaire mais les institutions souhaitent offrir toute une palette de services utiles aux étudiants et au personnel.

Le milieu étudiant (et professoral) est aujourd'hui familier des nombreuses applications commerciales d'utilisation intuitive et accessibles sur tout type de terminal. L'image de l'institution pâtit forcément d'un système d'information étudiant obsolète ou compliqué, et la productivité de ses utilisateurs s'en trouve amoindrie.

2.4.4 Group4Success apporte une réponse

La technologie est maintenant au centre de l'expérience des étudiants : ce sont des utilisateurs « clients » et ils sortent du lycée en espérant être accueillis dans des institutions équipées de solutions de dernière génération. Qu'il s'agisse d'emploi du temps, de facturation, ou de paiement des cours, ils exigent des solutions du niveau des applications qu'ils utilisent déjà dans la vie de tous les jours et qu'ils n'hésitent pas à partager avec le reste du monde sur les réseaux sociaux, en donnant leurs impressions sur le sujet. Pour ces institutions, le défi consiste donc à pouvoir offrir à leurs étudiants les applications interactives et intuitives qu'ils attendent ainsi que la puissance de travail et d'analyse dont leurs personnels ont besoin.

Afin de prévenir l'échec des étudiants, Group4 Success Management fournit des outils d'évaluation de risque permettant d'intervenir à temps auprès des étudiants concernés. En créant un profil de risque unique pour la population étudiante de l'institution, pour pouvoir interagir avec les personnes en difficulté au travers de plans individualisés tout en bénéficiant d'une collaboration inter-départements.

Group4 Success Management aide également les étudiants à rester dans la course grâce à

un outil complet d'évaluation du parcours diplômant. Cet outil permet aux étudiants de consulter les pré-requis d'obtention de semestre, année ou diplôme, leur permettant ainsi de rester motivés et bien informés pour être prêts à temps. Mieux encore, les étudiants peuvent effectuer des simulations pour évaluer leurs progrès et modifier éventuellement leur spécialité au cours de leurs études.

Le système LMD favorise l'autonomie et le travail personnel de l'étudiant. Un accompagnement numérique doit être proposé aux étudiants, comprenant notamment l'accès à des bibliothèques de ressources universitaires en ligne et la connexion à un « Environnement Numérique de Travail ». Grâce à une interface intuitive bâtie pour les utilisateurs, Group4 Success Management offre un haut niveau dans la facilité d'utilisation et la fiabilité des fonctionnalités, pour que les étudiants disposent de tout ce dont ils ont besoin sur leur propre matériel. Une grande qualité de service, toujours le « CRM pour tous » qui donne accès aux outils de la relation clients à l'ensemble des départements.

2.4.5 Synthèse des tâches-clés pour satisfaire l'ambition

Une description des tâches détermine les responsabilités et les fonctionnalités exigées afin de répondre aux besoins de l'application. Ces tâches comprennent les informations suivantes :

- La création d'un profil de risque unique, pour remédier au risque d'échec des étudiants.
- Recevoir les notifications et fournir l'outil pour l'évaluation de risque.
- Fournir un outil d'évaluation du parcours, permettant au étudiant de consulter les pré-requis d'obtention de semestre, année ou diplôme.
- Fournir un outil pour visualiser la suivi des étudiants.
- Offrir aux étudiants des applications interactives et intuitives.
- Comblent leurs attentes technophiles avec une application disponible tout le temps, partout et sur n'importe quel périphérique.
- Architecture adaptative et configurable.

2.5 Ambition : améliorer l'efficacité administrative

La pression sur les coûts éducatifs, les restrictions budgétaires et les attentes toujours plus élevées des étudiants incitent à un nouveau modèle économique pour les écoles

supérieures et les universités : il leur faut impérativement trouver de nouvelles voies dans la gestion des ressources. La gestion des coûts et l'amélioration de la productivité sont devenues de nouvelles normes de fonctionnement.

Pour permettre à l'institution d'améliorer son efficacité administrative, trois domaines requièrent plus particulièrement notre attention.

2.5.1 Automatiser les procédures courantes

Les procédures manuelles constituent une perte de temps et d'argent significative. Elles retardent le service aux étudiants, favorisent les erreurs d'enregistrement de données et peuvent générer une non-conformité importante aux règles administratives.

Les procédures opérées manuellement persistent toutefois, généralement du fait de systèmes dépassés et incapables de prendre en charge de nouveaux modes de fonctionnement, ou encore à cause de systèmes administratifs variés présentant le risque de duplication de saisie.

2.5.2 Soutenir la prise de décision sur données

Systèmes dérivés et opération verticalisées rendent les données moins exploitables : un agglomérat d'informations opaques décourage la collaboration, menace l'intégrité des données et peut rendre impossible l'analyse de mesures de performance.

Dans l'impossibilité de récupérer, tester et visualiser la masse de données que vous détenez, tout effort de collecte et de stockage est privé d'intérêt. Votre capacité d'amélioration s'en trouve fortement amoindrie, et l'impact sur votre image peut s'avérer énorme.

2.5.3 Optimiser les ressources humaines et financières

Sans système de gestion administrative unifié, vous éprouvez les pires difficultés à établir des budget proactifs et à contrôler vos dépenses, bataillant avec des sources de données varié pour comparer la rentabilité de vos programmes et départements.

Il est difficile, en outre, de quantifier le retour sur investissement du capital humain et d'identifier les meilleurs talents, surtout si vous utilisez des systèmes RH (Ressources Humaines) et paie séparés, qui complexifient encore davantage la prise en compte des évolutions réglementaires.

2.5.4 Group4Success apporte une réponse

La plateforme Group4Success apporte une technologie à la pointe de la modernité visant à libérer les intervenants afin qu'ils puissent se concentrer sur leur cœur de métier, et appliquant les principes des applications grand public que nous avons l'habitude d'utiliser quotidiennement. Le résultat en est une expérience utilisateur accrue, qui limite au maximum les nombreuses tâches fastidieuses sans valeur ajoutée et permet ainsi à l'utilisateur d'être plus présent et plus efficace là où il le choisit.

Group4Success automatise le cycle de vie de l'étudiant et élimine les dépenses liées à la maintenance de systèmes agglomérés. Les données intégrées sont en effet partagées entre les différentes fonctions, permettant un accès instantané à l'information pour toutes les parties prenantes et éliminant les doublons, notamment lors de la saisie. Il est aussi plus rapide et plus facile à déployer, vous pouvez donc immédiatement profiter de votre investissement.

Le plateforme regroupe aussi la finance, les ressources humaines, la gestion des projets et les achats, offre une efficacité et des bénéfices tangibles. Vous pouvez notamment améliorer votre transparence financière en accédant et en analysant les données de chaque département, individuellement ou sur l'ensemble de votre organisation. Le déroulement optimisé des processus et les outils intégrés améliorent notablement la productivité aidant à réduire les coûts, et augmentent vos revenus en évaluant au mieux les opportunités qui se présentent.

2.5.5 Aperçu des tâches-clés pour satisfaire l'ambition

Nous retiendrons que pour améliorer l'efficacité administratif, des tâches spécifiques sont à mettre en place :

- Automatisation des procédures courantes : certificat de scolarité, carte étudiant et bibliothèque.
- Construction des rapports détaillables par l'outil de tableau de bord permettant de zoomer jusqu'au détail et d'utiliser toutes les informations de votre données.
- Consultation, visualisation et analyse des données par campus, département, programme.
- Accès instantané à l'information pour tous les parties prenantes.

Chapitre 3

Modélisation statique et dynamique du système

Sommaire

3.1	Introduction	31
3.2	Modélisation dynamique du système	31
3.2.1	Spécification des besoins	31
3.2.2	Diagrammes de cas d'utilisation	34
3.2.3	Diagrammes de séquences	35
3.3	Modélisation de la base de données	43
3.3.1	Définitions de la base de données	43
3.3.2	Modélisation MERISE	43
3.3.3	Le dictionnaire des données	44
3.3.4	Le Modèle Conceptuel de Données (MCD)	47
3.3.5	Le Modèle Logique de Données (MLD)	47
3.4	Conclusion	50

3.1 Introduction

Un projet informatique nécessite une phase d'analyse et de modélisation. Dans la phase d'analyse, on cherche d'abord à bien comprendre et à décrire de façon précise les besoins des utilisateurs. Que souhaitent-ils faire avec le logiciel ? Quelles fonctionnalités veulent-ils ? Pour quel usage ? Comment l'action devrait-elle fonctionner ? C'est ce qu'on appelle « l'analyse des besoins ».

Nous allons dans ce chapitre faire en premier temps une modélisation orienté objet de notre système avec le langage UML (Unified Modeling Language) et ses outils : les diagrammes. Nous verrons comment ce langage contribue à la phase d'analyse des besoins et du domaines de notre projet. En deuxième temps, on va se servir de la méthode MERISE pour modéliser notre base données.

3.2 Modélisation dynamique du système

Pour ce qui est de la modélisation dynamique, on usera du formalisme UML (Unified Modeling Language) qu'on traduit par « Langage de Modélisation Unifié ». UML est un méthode de modélisation visuel constitué d'un ensemble de schémas appelés des diagrammes, qui donnent chacun une vision différente du projet à traiter. En effet, ce dernier s'impose aujourd'hui, comme le langage de modélisation objet standardisé pour la conception des logiciels.

3.2.1 Spécification des besoins

La spécification des besoins est certainement l'étape qui ne doit pas être négligée. En effet, celle-ci permet de déterminer les fonctionnalités que chaque utilisateur attend de l'application.

Les besoins des gestionnaires des ressources humaines

Pour le gestionnaire, le logiciel doit permettre :

- De gérer l'administration des employés : l'enregistrement, le suivi, le contrôle des données individuelles et la manipulation des renseignements du personnel.
- De gérer la gestion des rémunérations et de la paie : paiement des salaires, des primes.
- D'assurer la gestion des présences et des absences, des heures supplémentaires.
- D'alerter instantanément d'une nouvelle demande.

- De valider ou refuser les demandes de congés et d'avance.
- D'exporter les données.
- De créer des fiches employeur.
- D'archiver automatiquement les factures.
- De gérer les compétences et les diplômes des employés.

Les besoins des responsables administratif scolarité

Les responsables administratif scolarité veut un logiciel qui :

- Gère les formations : création des mentions, parcours, salles de classes, niveaux et matières.
- Gère les emplois du temps.
- Automatise les tâches courantes : certificat de scolarité, carte étudiant, fiche d'inscription.

Les besoins des salarié

Les salariés sont composés par trois acteur : les simples employés, les enseignants et les chef de mention.

Pour les employés, l'application doit leur permettre :

- De gérer leur agenda : ajout, modification, suppression et affichage des évènement.
- D'envoyer des demande d'absence, de congés, de remboursement ou d'avance.
- D'accéder et mettre à jour à ses informations personnelles et professionnelles (ajout de compétence, diplômes,...).
- De consulter leur emploi du temps personnel.
- D'accéder à sa fiche de paie.
- De générer un CV (Curriculum Vitæ).
- D'envoyer un message

Les chefs de mention veut une application :

- Pour organiser les études académique : définition des Unités d'Enseignement (UE) , ses Éléments Constitutifs (EC) et le système de crédit du LMD.
- Pour créer une promotion et affecter des étudiants dedans.
- Pour suivre les promotions en cours : commencer ou terminer un semestre, orientation des étudiants vers les spécialités.

- Qui permet de créer un profil de risque unique pour la population étudiante afin d'alerter les étudiants à risque d'échec.

Pour le cas des enseignants, ils attend d'une application qui :

- Permet d'affecter les notes des étudiants pour son module.
- Permet de consulter les emplois du temps.

Note : A savoir que l'acteur employé est une généralisation des deux acteurs chef de mention et enseignant, ce qui veut dire que les ces derniers ont accès à tous les taches des employés.

Les besoins des étudiants

Les attentes des étudiants du logiciel sont :

- Avoir une application haut niveau, interactives et intuitives avec facilité d'utilisation et accessible sur tout type de plateforme et enfin une architecture adaptative et configurable.
- Consulter les pré-requis d'obtention de semestre, année ou diplôme.
- Accéder à des bibliothèques de ressources universitaire en ligne.
- De communiquer entre étudiants.

Les besoins de l'administrateur client

En plus des tâches cités ci-dessus, l'administrateur client attend d'une application :

- Qui permet de gérer les utilisateur du logiciel : d'ajouter, modifier ou supprimer un utilisateur.
- Qui supporte l'envoi de message à tous les utilisateurs pendant une durée déterminée (avis de réunion, promotion, etc).
- Qui offre des notifications des éventuelles activités des personnels.
- Qui fait des suivis de l'établissement.
- Qui permet la gestion des locaux, logement, et les programmes de formation continue.
- Qui offre un tableau de bord orienté rôles qui affiche les chiffres clés et une vue d'ensemble de l'établissement.
- Qui permet de consulter et analyser les données par département et par programme.

Les besoins du prestataire de l'application

Pour le prestataire, qui est l'administrateur de l'application SaaS.

- Gestion des comptes clients et des utilisateurs.
- Gestion des abonnements : définir l'abonnement choisi par la société (le nombre d'utilisateur et les services ou modules choisis).
- Créer la base de données pour chaque nouvelle société client à partir d'une base de données de référence.
- Mise à jour et déploiement de l'application.

3.2.2 Diagrammes de cas d'utilisation

Le diagramme de cas d'utilisation est un diagramme dédié à l'expression des besoins et à la description des processus. Il est utilisé pour donner une vision globale du comportement fonctionnel du système.

Les diagrammes de cas d'utilisation qu'on montre ici représentent les relations et les interactions intervenant dans notre système.

Dans la partie gauche de chacun de ses diagramme de cas d'utilisation, sont figurés les acteurs primaires : les administrateurs, les gestionnaire RH (Ressources Humaines), les responsable de la scolarité, les enseignants et les chefs de mentions. Ceux sont les acteurs qui déclenchent les cas d'utilisation.

L'acteur secondaire, la base de données, est épargné pour ne pas encombrer les diagrammes, c'est l'acteur qui est sollicité par les cas d'utilisation.

3.2.3 Diagrammes de séquences

Elles sont la représentation graphique des interactions entre les acteurs et le système du point de vue dynamique qui nous montre le fonctionnement.

Dans un souci de simplification, on représente l'acteur principal à gauche du diagramme, et l'acteur secondaire à droite du système. Le but étant de décrire comment se déroulent les actions entre les acteurs ou objets.

La dimension verticale du diagramme représente le temps, permettant de visualiser l'enchaînement des actions dans le temps, et de spécifier la naissance et la mort d'objets.

Avant d'entrer au menu et faire l'ensemble des autres scénarios, chaque utilisateur doit impérativement s'authentifier en utilisant son login et un mot de passe pour la sécurité du système.

Le diagramme qui suit présente la phase d'authentification de chaque utilisateur.

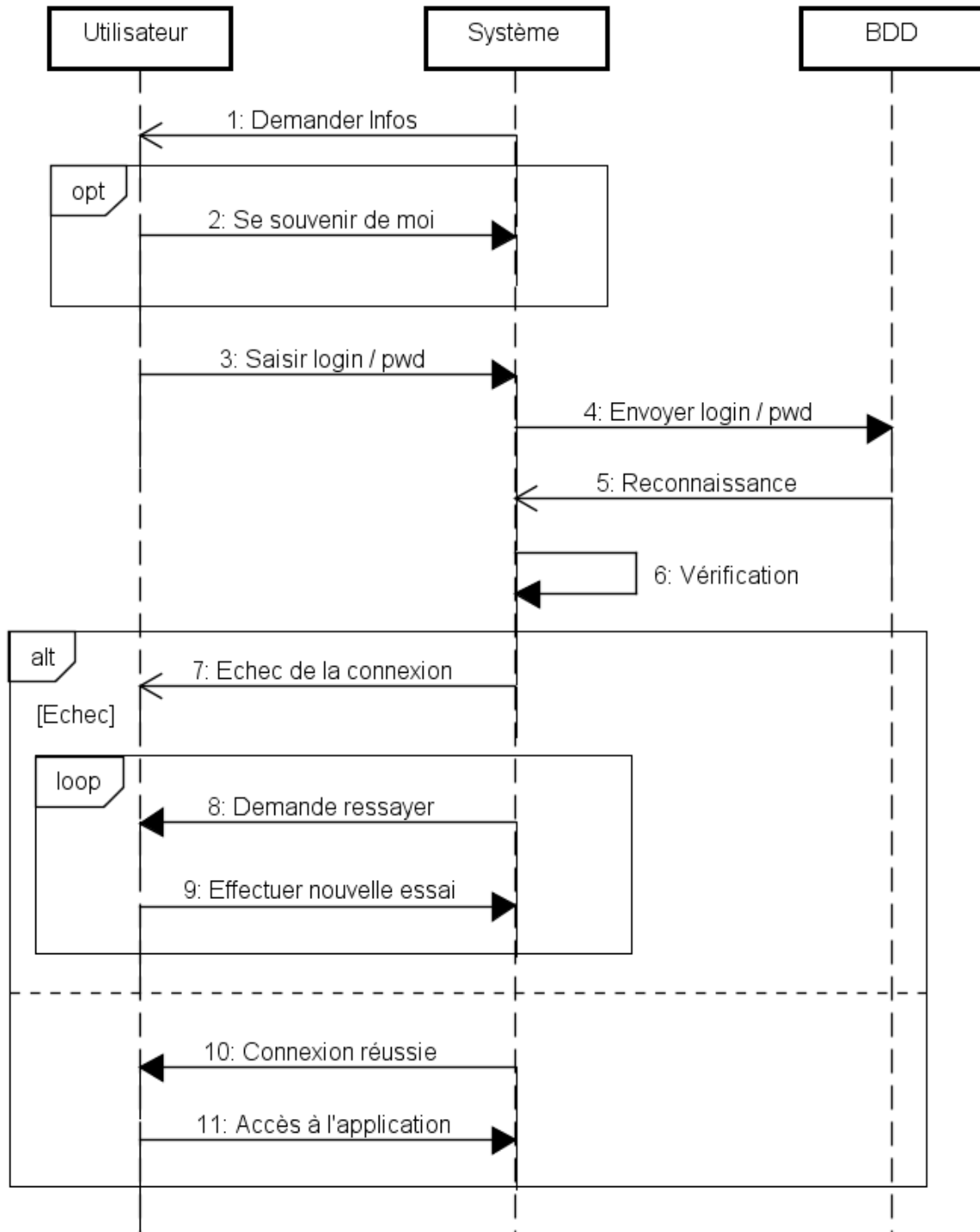


FIGURE 3.1 – Diagramme de séquence : Authentification

Cette figure décrit les scénarios possibles lors de l'authentification utilisateur.

- > L'utilisateur demande l'accès à l'application et donne le login et le mot de passe.
- > Un test doit être réalisé (existence et compatibilité du login / mot de passe).
- > Si les données sont correctes alors permettre à l'internaute d'accéder à l'application.
- > Si les données ne sont pas correctes est pas valide, l'utilisateur doit soit réessayer, soit quitter l'application.

Après l'authentification de l'utilisateur il y a tout un framework derrière, il valide l'authentification de dernier utilisateur connecter avec les droits qu'il a tout en récupérant ses derniers de la base de donnée.

Tout au long de la session de l'utilisateur, il a la possibilité d'accéder qu'aux services dont il est autorisé. Dans le cas contraire il sera redirigé vers une page d'erreur 404.

Le diagramme qui suit représente la gestion des habilitations.

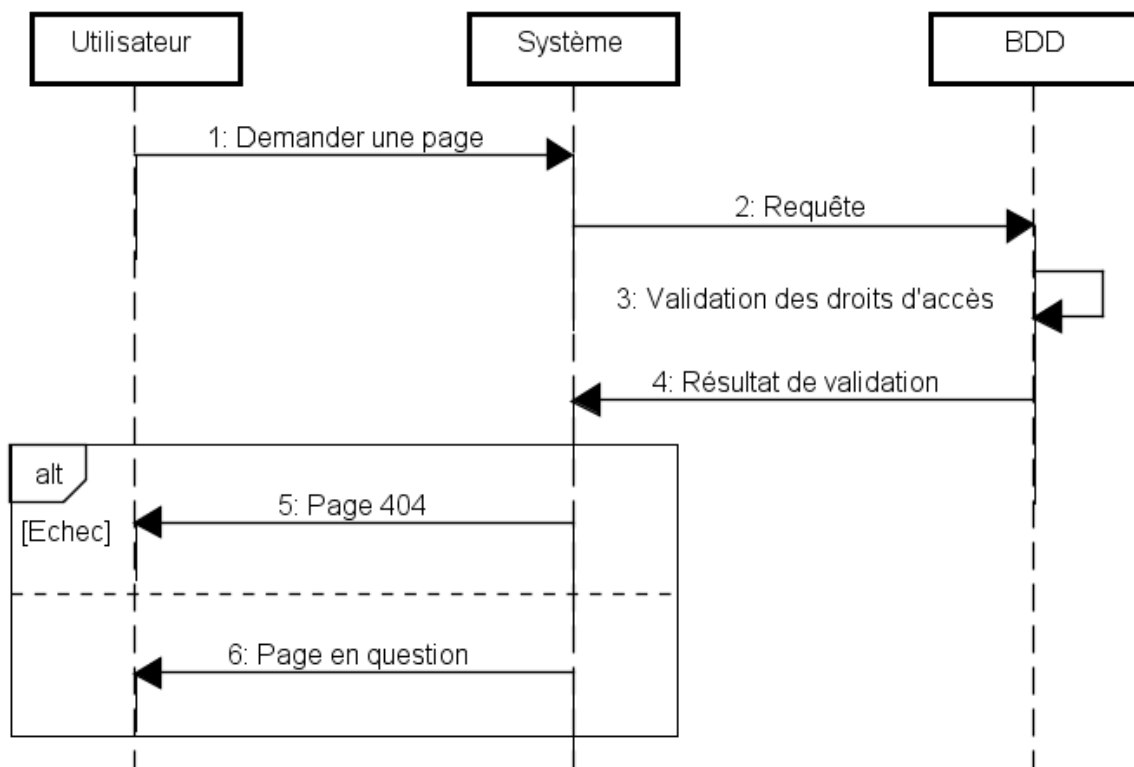


FIGURE 3.2 – Diagramme de séquence : Gestion des habilitations

Nous allons représenter ici quelques scénarios basique dont les utilisateurs pourront faire avec le système.

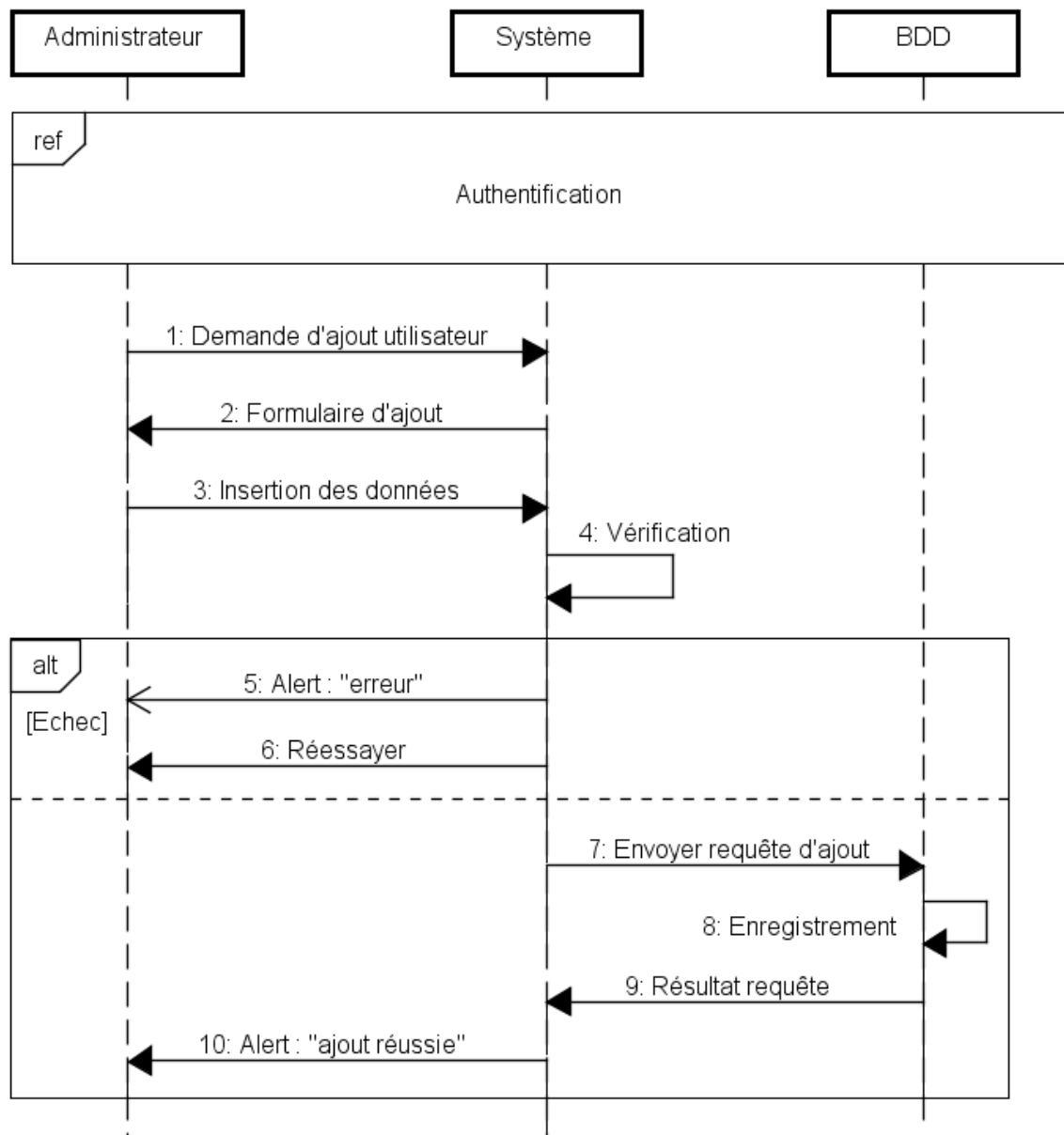


FIGURE 3.3 – Diagramme de séquence : Ajout d'utilisateur

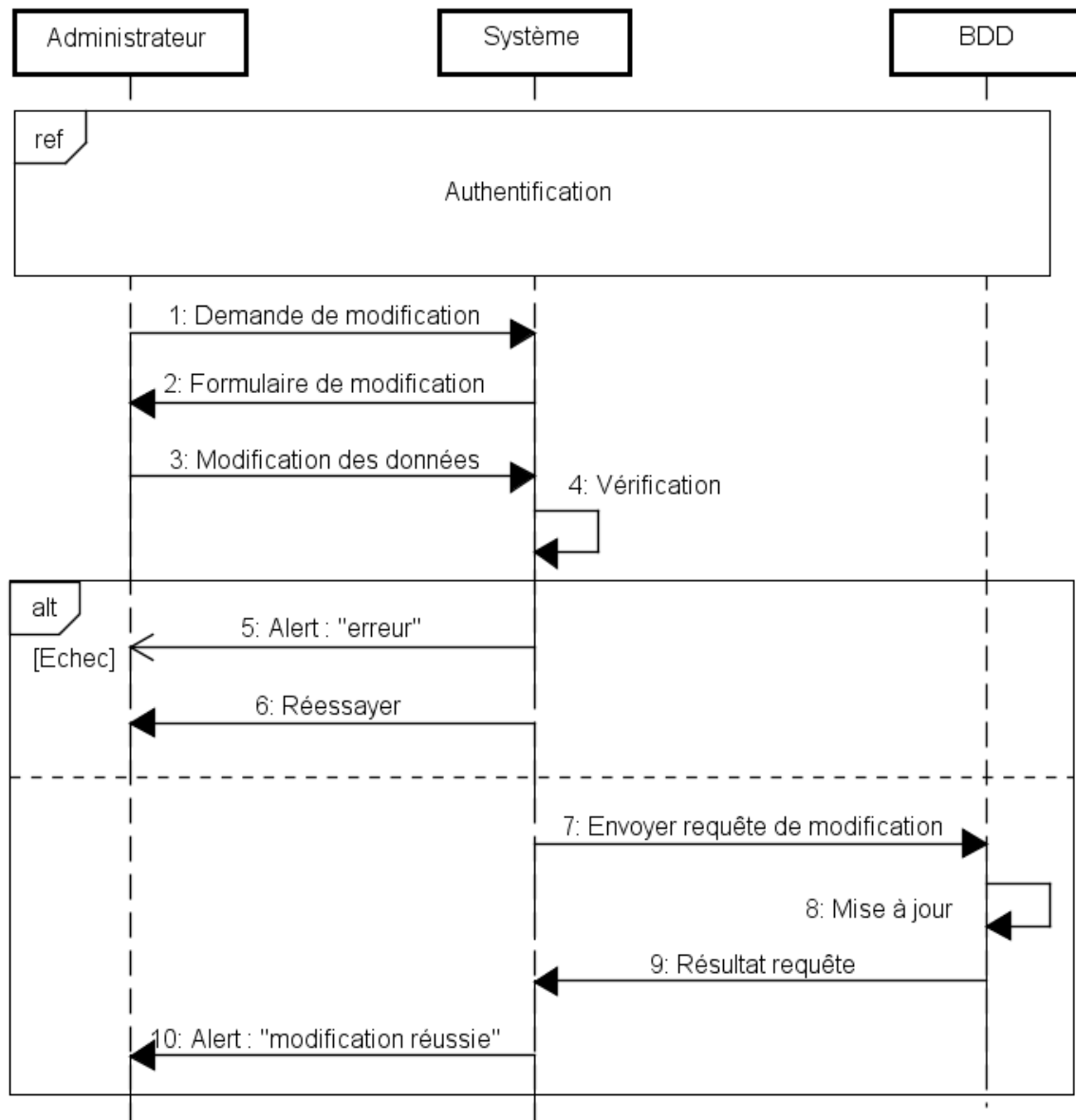


FIGURE 3.4 – Diagramme de séquence : Modification

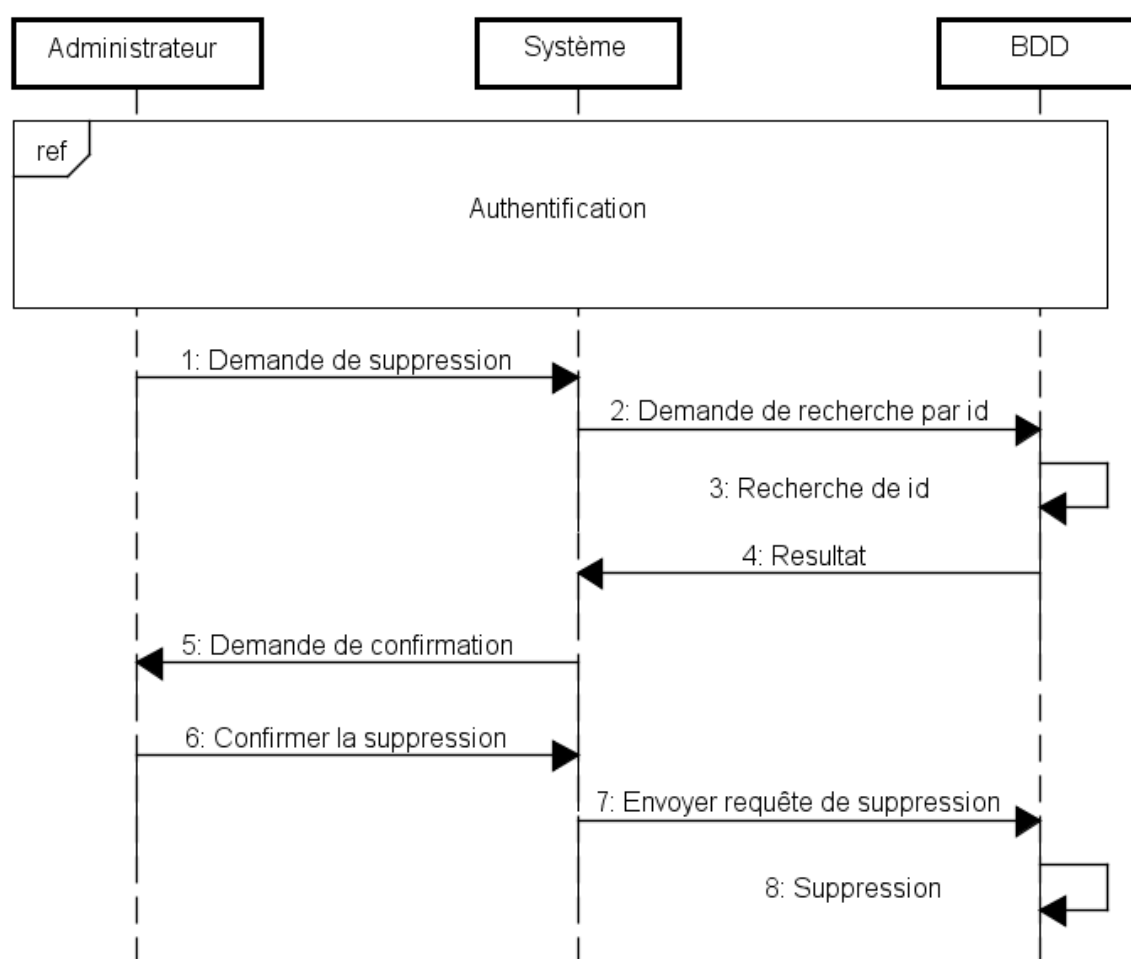


FIGURE 3.5 – Diagramme de séquence : Suppression

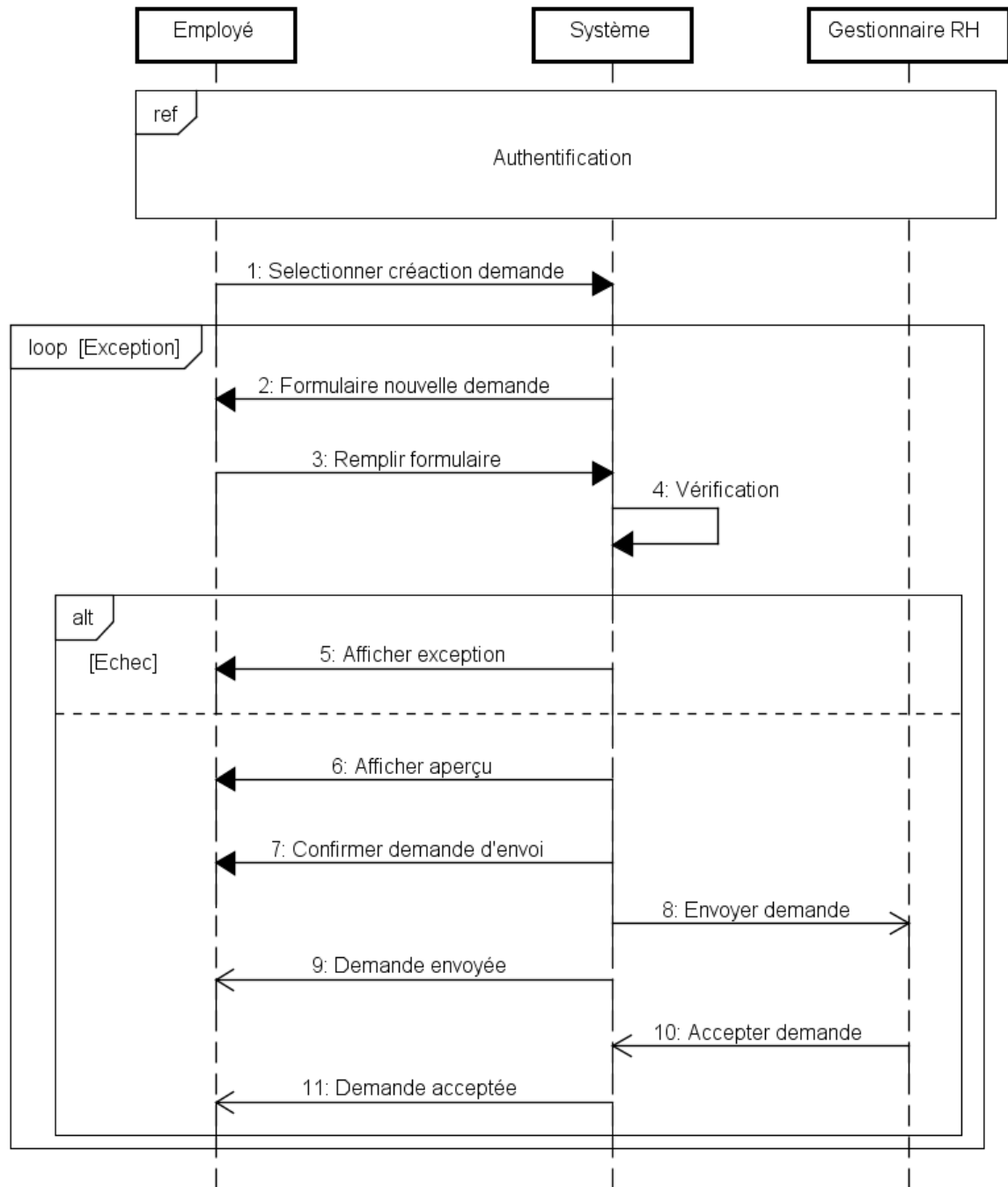


FIGURE 3.6 – Diagramme de séquence : Demandes de congé / absence / avance

En principe tous les scénarios qui restent ont presque la même logique au niveau des interactions, par conséquent nous allons présenter le scénario qui décrit l'envoi de message par mail.

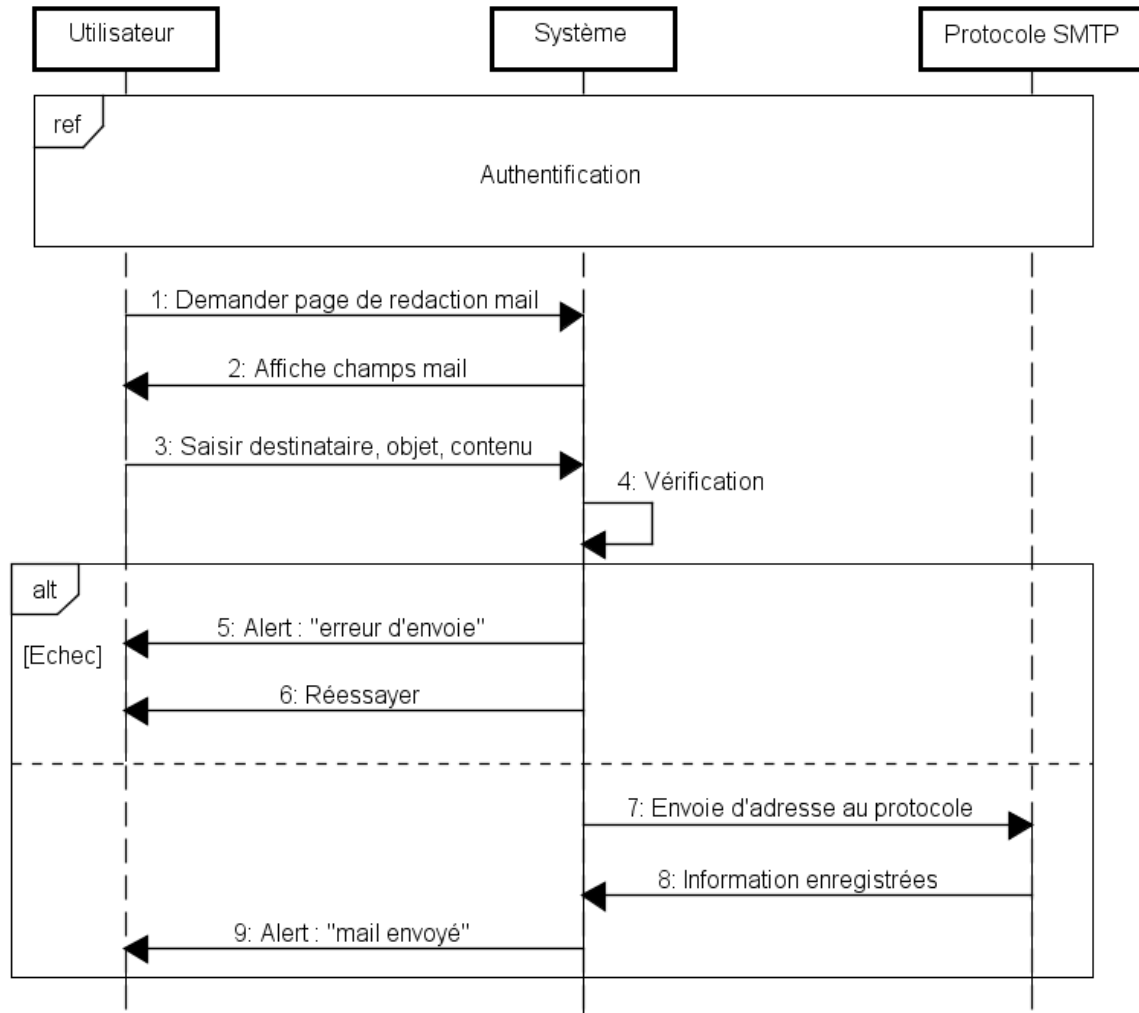


FIGURE 3.7 – Diagramme de séquence : Envoi un e-mail

Des diagrammes d'activités peuvent aussi être utilisés comme des alternatives aux diagrammes de séquence pour décrire un cas d'utilisation. Ils représentent les règles d'enchaînement des actions et décisions au sein d'une activité, sous la forme d'un déroulement d'états regroupés séquentiellement comme un organigramme. Mais ils ne représentent pas les interactions entre les éléments du système et de ses acteurs contrairement aux diagrammes d'activités.

3.3 Modélisation de la base de données

3.3.1 Définitions de la base de données

Une base de données (BDD) est un ensemble structuré de connaissance organisée à l'aide d'un formalisme. Elle représente les informations nécessaires au fonctionnement d'une entreprise [4]. Ces données doivent pouvoir être utilisées par des utilisateur différents. Ainsi, la base de données est généralement couplée à celle de réseau, afin de pouvoir mettre en commun ces informations.

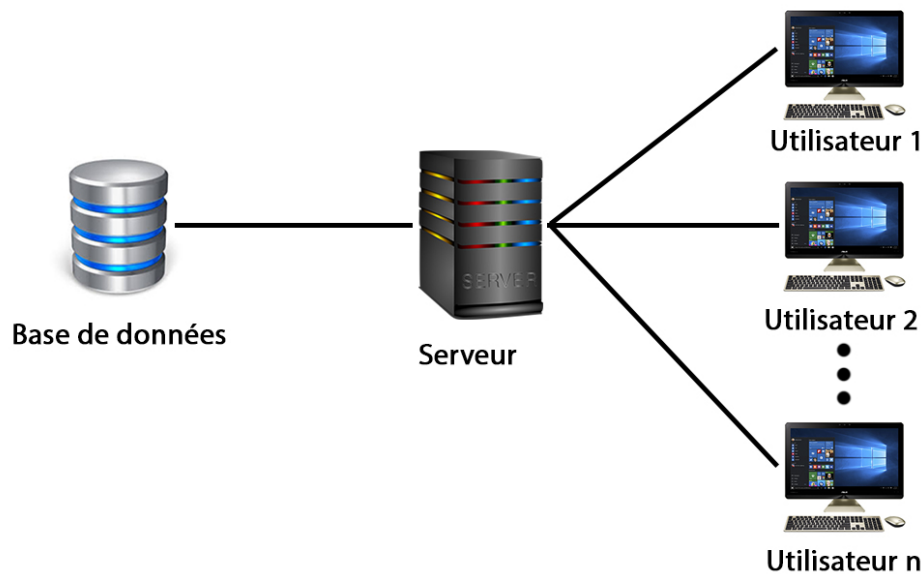


FIGURE 3.8 – Le système de la base de données

3.3.2 Modélisation MERISE

On a besoin de modéliser une base de données [4] :

- Pour définir le contenu en informations (connaissances) de la future base de données.
- Le modèle en tant que tel est un résultat scientifique car décrit et résume le "savoir" disponible sur un sujet à un instant donné.
- Pour disposer d'un outil de communication accessible aux futurs utilisateur de la base de données.

Pour ce qui est de la modélisation de la base de données, nous utiliserons la méthode de MERISE, qui est une méthode d'analyse de conception des systèmes d'information basée sur le principe de la séparation des données et des traitements [18]. Elle possède un certain nombre de modèles (ou schémas) qui sont répartis sur 3 niveaux :

- Le niveau conceptuel : contenu de la base en termes de concepts (Modèle Entité Association),
- Le niveau logique ou organisationnel : traduction du schéma conceptuel dans le paradigme du système de base de données (modèle relationnel),
- Le niveau physique : organisation et accès aux données dans la base.

Nous ne nous intéresserons ici qu'à certains schémas permettant la conception d'une base de données relationnelle.

3.3.3 Le dictionnaire des données

Le dictionnaire des données est un documents qui regroupe toutes les données que vous aurez à conserver dans votre base. Pour chaque donnée, il indique :

- Le code mnémonique : il s'agit d'un libellé désignant une donnée.
- La désignation : il s'agit d'une mention décrivant ce à qui la donnée correspond.
- La nature de donnée :
 - A ou Alphabétique : lorsque la donnée est uniquement composée de caractères alphabétique.
 - N ou Numérique : lorsque la donnée est composée uniquement de nombres (entiers ou réels).
 - AN ou Alphanumérique : lorsque la donnée peut être composée à la fois de caractères alphabétiques et numériques.
 - Date : lorsque la donnée est une date (au format AAAA-MM-JJ)
 - Booléen : Vrai ou Faux
- La taille : elle s'exprime en nombre de caractères ou de chiffres.
- Le type de donnée :
 - E : Élémentaire ;
 - Ca : Résultat d'un calcul ;
 - Co : Résultat d'une concaténation.

Voici le tableau montrant le dictionnaire de données importantes de notre base de données.

TABLEAU 3.1: Dictionnaire de données

Propriété	Contenu	Nature	Taille
id_util	Identifiant de l'utilisateur	N	10
nom_util	Nom de l'utilisateur	A	30
prenom_util	Prénom de l'utilisateur	A	60
sexe_util	Sexe de l'utilisateur	A	10
date_naissance	Date de naissance de l'utilisateur	Date	-
adresse_util	Adresse utilisateur	AN	120
etat_civil	Etat civil de l'utilisateur	AN	30
tel_util	Numéro de téléphone de l'utilisateur	N	15
email_util	Adresse email de l'utilisateur	AN	60
nation_etu	Nationalité de l'étudiant	A	30
cin_etu	Numéro carte d'identité national de l'utilisateur	N	12
photo	Numéro carte d'identité national de l'étudiant	AN	30
id_par	Identifiant d'un parent	N	10
nom_par	Nom d'un parent	A	30
prenom_par	Prenom d'un parent	A	60
adresse_par	Adresse d'un parent	AN	120
fonction_par	Fonction d'un parent	A	60
id_empl_par	Identifiant de l'employé	N	10
grade	Grade d'un enseignant	A	30
date_embauche	Date d'embauche de l'employé	Date	-
fonction_empl	Fonction de l'employé	A	60
id_dmd	Identifiant d'une demande	N	10
type_dmd	Type d'une demande	A	30
contenu_dmd	Contenu de la demande	AN	-
identifiant	Identifiant d'une authentification	AN	30

Propriété	Contenu	Nature	Taille
mdp	Mot de passe	AN	60
privilege	Droit d'accès de l'utilisateur	A	30
id_mail	Identifiant d'un mail	N	10
destinataire	Destinataire du mail	AN	60
objet	Objet du mail	AN	60
contenu	Contenu du mail	AN	-
id_admin	Identifiant de l'administrateur	N	10
nom_eta	Nom de l'établissement	AN	30
apropos_eta	Apropos de l'établissement	AN	-
id_doc	Identifiant d'un document	N	10
titr_doc	Titre du document	AN	60
description_doc	Description du document	AN	-
id_ensgn	Identifiant de l'enseignant	N	10
grade	Grade de l'enseignant	A	30
id_etu	Identifiant de l'étudiant	N	10
date_inscription	Date d'inscription de l'étudiant	Date	-
note	Note correspondante à une matière	N	10
id_matiere	Identifiant d'une matière	N	10
designation	Nom de la matière	A	60
UE	Unité d'enseignement	A	30
coeff	Coefficient de la matière	N	10
id_stage	Identifiant du stage	N	10
date_debut	Date de début du stage	Date	-
date_fin	Date de fin du stage	Date	-
id_theme	Identifiant d'un thème	N	10
nom_thme	Nom d'un thème	AN	120
nom_mention	Nom d'une mention ou département	A	120
id_parcours	Identifiant du parcours ou de la filière	N	10
nom_parcours	Nom du parcours	A	120

Propriété	Contenu	Nature	Taille
id_classe	Identifiant d'une classe	N	10
nom_classe	Nom d'une classe	AN	120
id_niveau	Identifiant d'un niveau	N	10
nom_niveau	Nom d'un niveau	AN	30

3.3.4 Le Modèle Conceptuel de Données (MCD)

Le Modèle Conceptuel des Données (MCD) a pour but d'écrire de façon formelle les données qui seront utilisées par le système d'information. Il s'agit donc d'une représentation sous forme d'un schéma des données relatives au sujet à traiter (en gros les entités, leurs attributs et les relations qu'elles entretiennent). (Voir figure 3.14).

3.3.5 Le Modèle Logique de Données (MLD)

L'étape du Modèle Logique de Données (MLD) se situe chronologiquement juste après l'étape MCD et revient à présenter les objets du MCD sous une forme compréhensible par un SGBD. Pour faire court, dans un contexte SGBD relationnel, les objets représentés sont désormais des tables (disons SQL) et les liens qui les unissent. (Voir figure 3.15).

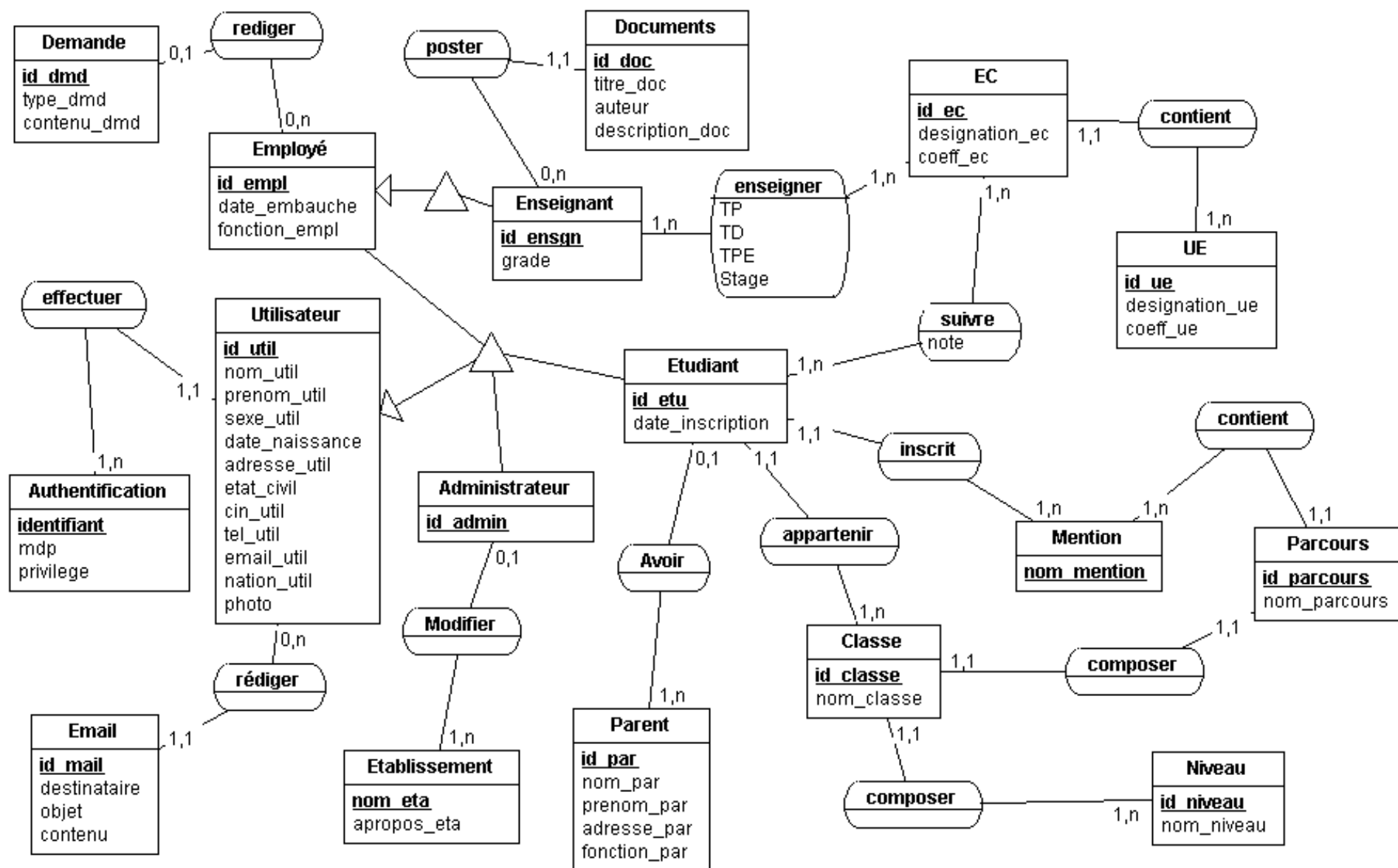
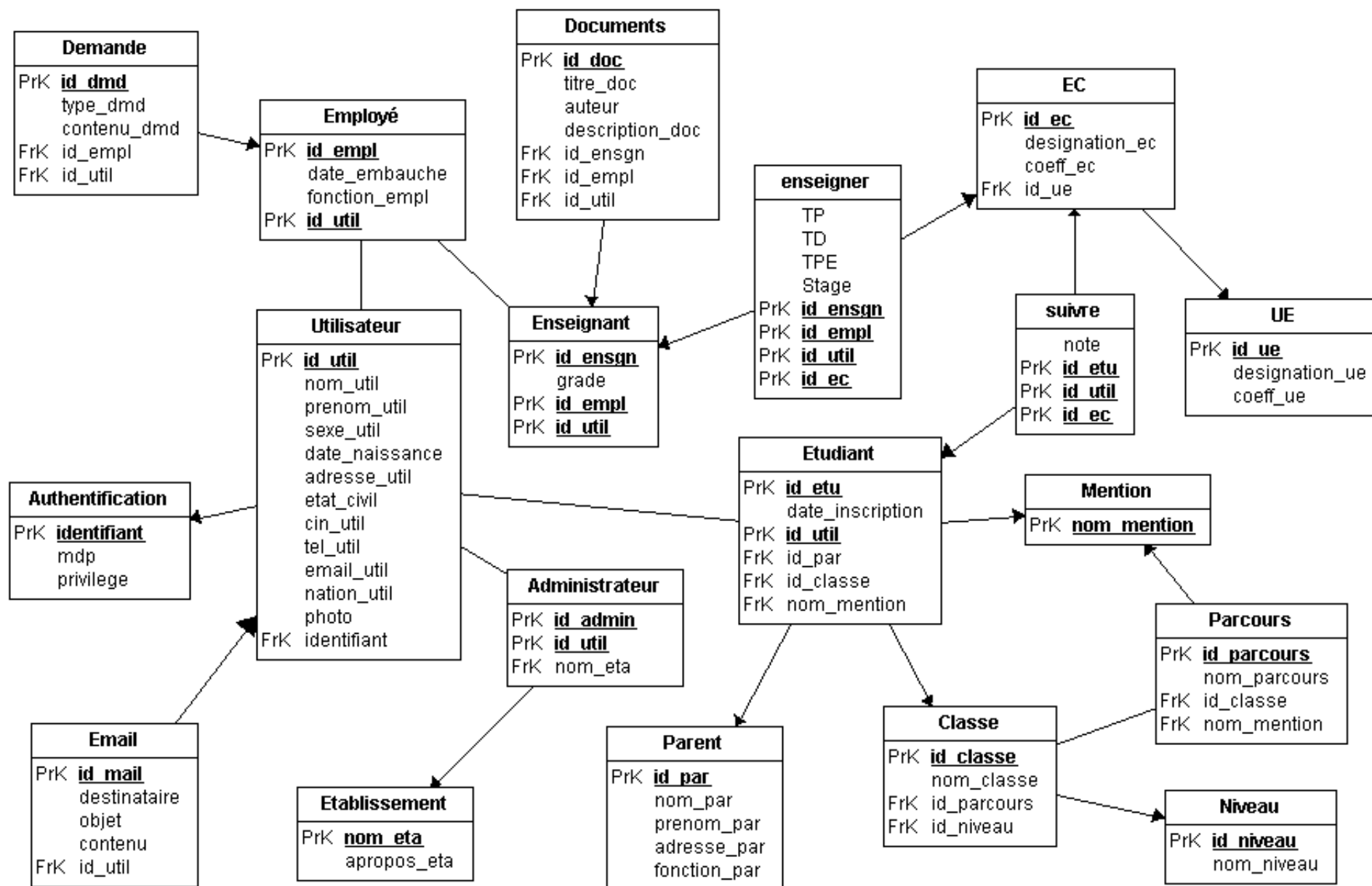


FIGURE 3.9 – Modèle Conceptuel de Données



3.4 Conclusion

Dans ce chapitre, nous avons entamé la modélisation du logiciel. La conception et la modélisation sont supposées comme les noyaux du travail. En effet, celles-ci sont garantes que le système aboutira aux résultats espérés. Nous avons choisi UML comme langage de modélisation dynamique et MERISE pour la conception de la base de données. Le chapitre suivant sera consacré à la programmation et la réalisation de l'application.

Chapitre 4

Programmation et réalisation du logiciel

Sommaire

4.1 Introduction	52
4.2 Les Systèmes de Gestion de Base de Données (SGBD)	52
4.2.1 Définition	52
4.2.2 Firebase	52
4.2.3 Pourquoi le choix de Firebase	52
4.3 Les langages de programmation	53
4.3.1 De JSF à Angular	53
4.3.2 Angular	53
4.3.3 Une navigation plus fluide pour le visiteur	54
4.3.4 Une meilleure gestion de contenu dynamique	54
4.3.5 Une plateforme extensible et modulaire	54
4.3.6 Présentation en HTML5/CSS3/JS/jQuery	55
4.4 Conception	55
4.4.1 Création du projet	55
4.4.2 La structure des dossiers	55
4.4.3 Firebase Realtime Database, une base de données en temps réel . .	57
4.4.4 Sécurisation des données	58

4.1 Introduction

Après avoir franchi l'étape de la modélisation, on peut désormais entamer la programmation et la réalisation du système. Dans cette partie, on abordera en premier lieu la description de notre Système de Gestion de Base de Données (SGBD) ainsi que les langages de programmation utilisés et par la suite on présentera les démarches de conception pour l'implémentation du système.

4.2 Les Systèmes de Gestion de Base de Données (SGBD)

4.2.1 Définition

Les SGBD sont des logiciels système destinés à stocker et à partager des informations dans une base de données, en garantissant la qualité, la pérennité et la confidentialité des informations, tout en cachant la complexité des opérations. En bref c'est un logiciel intermédiaires entre les utilisateurs et les bases de données. Une base de données est un magasin de données composé de plusieurs fichiers manipulés exclusivement par le SGBD. Ce dernier cache la complexité de manipulation des structures de la base de données en mettant à disposition une vue synthétique du contenu.

4.2.2 Firebase

Firebase est à la fois un SGBD et un ensemble de services d'hébergement pour n'importe quel type d'application (Web service, Android, iOS, Javascript, Node.js, Java, Unity, PHP, C++ ...). Il propose d'héberger en NoSQL et en temps réel des bases de données, du contenu, de l'authentification sociale (Google, Facebook, Twitter et Github), et des notifications, ou encore des services, tel que par exemple un serveur de communication temps réel [16].

4.2.3 Pourquoi le choix de Firebase

Lorsqu'on développe une application, qu'elle soit destinée au grand public ou réservée à un usage interne à l'entreprise, certaines fonctionnalités sont systématiquement requises, telles que la gestion des utilisateurs, de la connexion et des notifications. La gestion de ces fonctionnalités est fastidieuse, répétitive si le système se compose de plusieurs applications, et critiques en termes de sécurité, dans la mesure où l'on

va stocker des mots de passe. D'où l'intervention de Firebase qui va nous permettre d'externaliser cette gestion.

4.3 Les langages de programmation

La présentation des langages de programmation sont des facteurs importants pour la réussite de la conception et la réalisation d'un logiciel. En effet, chaque langage a ses propres caractéristiques qui doivent être compatibles avec les contraintes et conditions exigées par le cahier des charges. Le choix doit être fait en fonction de leur souplesse, leur adaptation aux fonctions exigées et aux ressources matérielles disponibles.

4.3.1 De JSF à Angular

Au début nous avons utiliser le langage "Multi-end" : JSF (Java Server Faces), qui est un framework MVC (Modèle Vue Controleur) Java traditionnels à base d'actions et basé sur la notion de composants. Mais en allant plus loin dans les détails, nous pouvons en déduire des principaux inconvénients qui ont empêchés la progression du développement de l'application.

Deux inconvénients majeurs JSF :

1. Grande courbe d'apprentissage. JSF est complexe, c'est juste vrai.
2. Sa nature composante. Le framwork basé sur les composants essaye de cacher la vraie nature du Web, qui vient avec une énorme quantité de complications et de désastres (comme ne pas soutenir GET dans JSF). Chaque framework basé sur des composants ajoute de l'abstraction au développement Web, et cette abstraction se traduit par un surcoût inutile et une complexité plus élevée.

En bref, JSF essaient de cacher au programmeur la vrai requête/réponse et la nature sans état du web. C'est un désavantage majeur pour JSF, certe il peut convenir à certains types d'applications (intranet, formulaires intensifs), mais pour une application web réelle, ce n'est pas une bonne solution.

4.3.2 Angular

Après avoir été bloqué avec JSF, on a recommencer le développement avec Angular, un langage fessant partie de la nouvelle vague de frameworks JavaScript portée par Google. C'est un socle technique qui se veut extensible et qui pousse vers un développement

structuré. Il s'inscrit dans un mouvement d'innovation côté front-end, dont le but est d'éviter le chargement d'une nouvelle page à chaque action demandée [17].

Angular présente la particularité d'être totalement frontend (côté client). Pour utiliser Angular dans notre application, nous allons devoir utiliser un système backend (côté serveur), dans notre cas ce sera Firebase pour gérer la connexion à la base de données. En utilisant ce langage, on voit clairement la distinction de développement entre front et back.

4.3.3 Une navigation plus fluide pour le visiteur

L'utilisation d'Angular impose un développement selon la structure MVVM (Modèle-Vue-Vue-Modèle). Ce principe offre un avantage de taille, celui de diminuer considérablement la vitesse de chargement des pages. En effet, le nombre d'accès au serveur est fortement diminué car la communication se fait majoritairement en mode asynchrone. Autrement dit, l'interface visuelle est portée côté client. En conséquence, une importante partie des requêtes supportées en arrière-plan est ainsi supprimée, ce qui permet de concevoir des applications web plus légères. Ceci explique sa parfaite adaptation pour les applications web monopage (SPA) qui ne comportent qu'une seule et unique interface.

4.3.4 Une meilleure gestion de contenu dynamique

Le framework estampillé Google étend le langage HTML traditionnel pour enrichir davantage le contenu dynamique par le biais d'un couplage bidirectionnel (two-way data-binding). Derrière ce nom barbare se cache un concept très pratique : dès qu'une vue est modifiée, la donnée est envoyée au model associé qui rafraîchit à son tour la vue. Concrètement, si un internaute remplit un champ texte, la valeur saisie peut s'afficher à un autre endroit de la page et ce sans rechargement ni soumission au préalable de l'information. Il s'agit donc d'une synchronisation entre le modèle et la vue qui permet de créer des applications plus responsives [2].

4.3.5 Une plateforme extensible et modulaire

Pour pallier à la nature statique de la solution HTML, Angular introduit la notion de directives chargée d'associer un comportement JavaScript spécifique à chaque nouvel élément de ce langage balisé. Ces composants vont permettre de rendre le code extensible et modulable. Il devient alors facile d'ajouter, de modifier ou de supprimer des directives,

ce qui fait entre autre la popularité d'Angular. Celles-ci peuvent tout-à-fait être partagées et réutilisées de projet en projet pour éviter de réinventer la roue [3].

4.3.6 Présentation en HTML5/CSS3/JS/jQuery

Pour afficher, mettre en forme et structurer les données à l'utilisateur, nous utilisons les langages HTML (Hypertext Markup Language), le CSS (Cascading Style Sheets) et éventuellement le Javascript (JS) pour l'interactivité de l'application. Ceux-ci ont une caractéristique commune importante : ils sont tous interprétés par le navigateur, directement sur la machine client.

L'utilisation de jQuery est aussi un atout, c'est une bibliothèque JavaScript libre et multi-plateforme créée pour faciliter l'écriture de scripts côté client dans le code HTML.

Cette bibliothèque va nous servir notamment aux fonctionnalités suivantes :

- Manipulation des feuilles de style en cascade (ajout/suppression des classes, d'attributs...);
- Événements;
- Effets visuels et animations;

4.4 Conception

4.4.1 Création du projet

Nous devons avant tout installer Node et npm. Ensuite pour installer angular CLI, il suffit de taper la commande suivante dans votre bash :

```
npm install -g @angular/cli
```

Pour créer maintenant notre application, il suffit d'exécuter la commande suivante :

```
ng new g4sm
```

4.4.2 La structure des dossiers

Après l'installation et la création de votre projet avec angular-cli, nous allons voir la structure des dossiers et fichiers dans l'architecture d'angular-cli.

```
// Tout ce qui va concerner les tests end to end  
|- e2e/
```

```
|----- app.e2e-spec.ts
|----- app.po.ts
|----- tsconfig.e2e.json

// les dépendances avec npm
|- node_modules/

// l'endroit où les fichiers de build seront mis
|- dist/

// Le dossier où vous allez modifier vos fichiers de code
//Là où va se trouver vos composants, services, etc..
|- src/
|----- app/
|----- app.component.css|html|spec.ts|ts
|----- app.module.ts
|----- assets/
|----- environments/
|----- environment.prod.ts|ts
|----- favicon.ico
|----- index.html
|----- main.ts
|----- polyfills.ts
|----- styles.css
|----- test.ts
|----- tsconfig.app.json
|----- tsconfig.spec.json
|----- typings.d.ts

// la configuration globale de votre application
|- .angular-cli.json // the main configuration file
|- .editorconfig      // editorconfig which is used in some VS Code setups
|- .gitignore
```

```
|- karma.conf.js
|- package.json
|- protractor.conf.js
|- README.md
|- tsconfig.json
|- tslint.json
```

Nous allons quasiment passer tout notre temps dans le dossier *src/app*. Ce dossier contient presque tous les fichiers dont nous avons besoin pour coder notre application. Les fichiers contenus dans ce dossier sont ensuite compilés dans le dossier *dist*. Nous pouvons aussi installer des dépendances avec le gestionnaire de package de node : "npm". Ces dépendances seront installées dans le dossier "node modules".

4.4.3 Firebase Realtime Database, une base de données en temps réel

La base de données avec Firebase se présente sous la forme d'un arbre "infini", composé uniquement d'objets clé/valeur. Il s'agit d'une base de données cloud où les données sont stockées sous format JSON (JavaScript Object Notation) et synchronisées en temps réel avec chaque client connecté. Par exemple, si on souhaite stocker un ensemble d'utilisateur, cela devrait plus ou moins ressembler à cela :

```
{
  "utilisateurs": {
    "id1": {
      "nom": "Midonique",
      "email": "mido@gmail.com",
    },
    "id2": { ... },
    "id3": { ... }
  }
}
```

4.4.4 Sécurisation des données

Pour faire simple : l'accès aux données présentes dans la base est régi par un ensemble de règles. Établissons un simple fichier de règles pour nos utilisateurs :

```
{  
  "rules": {  
    ".read": "auth != null",  
    ".write": "auth != null"  
  }  
}
```

C'est uniquement un utilisateur connecté pourra lire, enregistrer et modifier ces données. Il s'agit évidemment ici d'un exemple simple : nous pouvez établir des règles de sécurité bien plus complexes afin de protéger les données des utilisateurs.

Chapitre 5

Présentation et déploiement de l'application

Sommaire

5.1 Pré-requis	60
5.2 Présentation de l'application	60
5.2.1 Fenêtre d'authentification	60
5.2.2 Fenêtre d'inscription	61
5.2.3 Les email de validation et de réinitialisation de mot de passe	62
5.2.4 Le tableau de bord	63
5.2.5 Liste et ajout d'utilisateur	63
5.2.6 Page d'erreur 404	65
5.3 Les services backend de Firebase	65
5.3.1 Authentification	65
5.4 Déploiement de l'application sur Firebase	65
5.4.1 Préparer l'application pour la production	65
5.4.2 Installer Firebase-CLI	66
5.4.3 Configurer l'application Angular	66
5.4.4 Déploiement de l'application	66

5.1 Pré-requis

Pour faire tourner l'application, l'utilisateur devrait munir d'un périphérique possédant un simple navigateur web, tels un ordinateur, une tablette ou smartphone. Puis d'être abonnés à l'application et avoir les codes d'accès. Sachant que l'application est hébergée à distance, une connexion internet est indispensable. Une connexion ADSL (Asymmetric Digital Subscriber Line) de 8Mo/s sera largement suffisante pour cela.

5.2 Présentation de l'application

Nous allons faire une brève présentation de l'interface aussi bien que les fonctionnalités de l'application. Notons au tout début que c'est une application SaaS qui a pour nomination : « Group4Success ». Cette application sera utilisé par différentes établissement en utilisant le même URL.

5.2.1 Fenêtre d'authentification

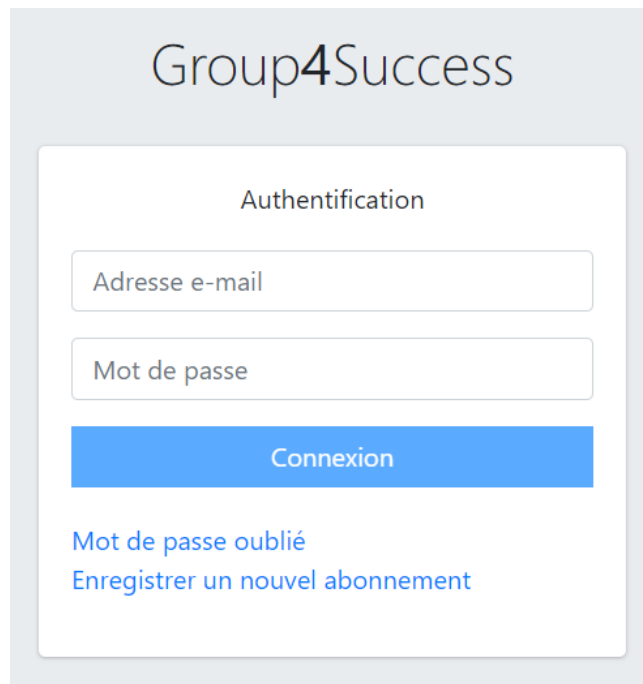
The image shows a web-based authentication interface for 'Group4Success'. The title 'Group4Success' is at the top in a large, dark font. Below it, the word 'Authentification' is centered. There are two input fields: 'Adresse e-mail' and 'Mot de passe'. Below these is a blue button labeled 'Connexion'. At the bottom, there are two links: 'Mot de passe oublié' and 'Enregistrer un nouvel abonnement'.

FIGURE 5.1 – Fenêtre d'authentification

Avant tout, l'établissement client doit impérativement être inscrit à un abonnement sur le site web de G4SM (Group4SuccessManagement) afin de recevoir un identifiant et un mot de passe pour que l'administrateur client puisse s'authentifier dans cette application.

5.2.2 Fenêtre d'inscription

Le site web de G4SM qui devrait servir de présentation et inscription à un abonnement n'est pas encore déployer en ligne. Pour cela, on peut tester l'application par l'intermédiaire du lien "Enregistrer un nouvel abonnement" accessible depuis l'écran d'authentification. Ce dernier ouvre une fenêtre d'inscription pour pouvoir créer un compte de test.

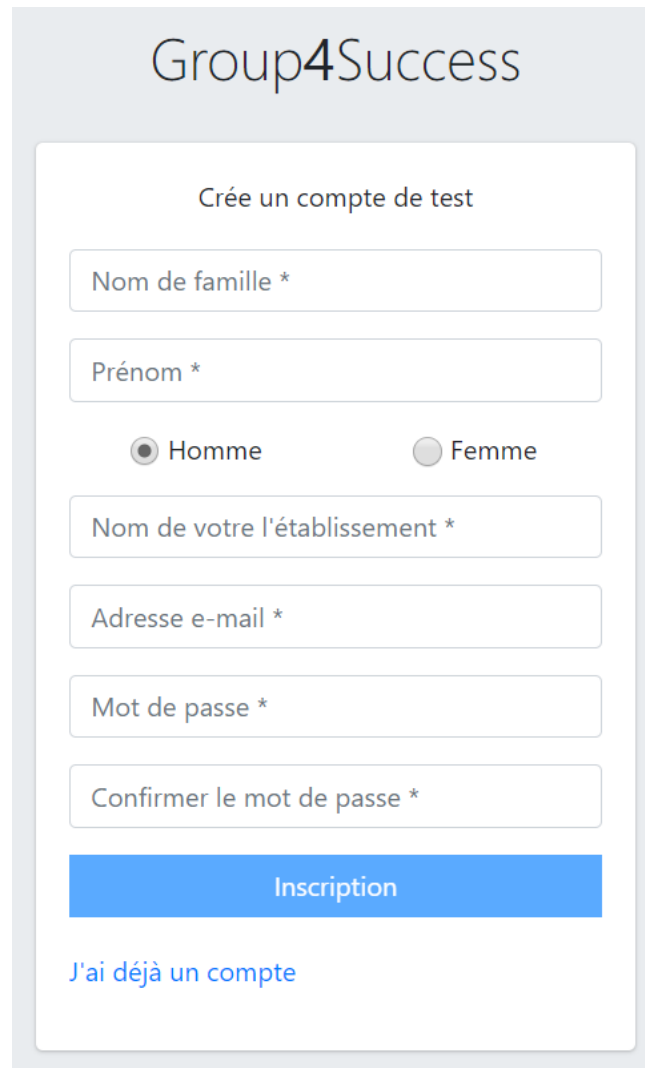
The image shows a registration form for 'Group4Success'. The form is titled 'Crée un compte de test' and is set against a light gray background. It contains several input fields: 'Nom de famille *', 'Prénom *', 'Nom de votre l'établissement *', 'Adresse e-mail *', 'Mot de passe *', and 'Confirmer le mot de passe *'. There are also two radio buttons for gender: 'Homme' (selected) and 'Femme'. A blue 'Inscription' button is at the bottom of the form, and a link 'J'ai déjà un compte' is below it.

FIGURE 5.2 – Fenêtre d'inscription

La validation des données dans les champs du formulaire est gérée par la méthode de réactive fourni par Angular. Cette méthode utilise des fonctions en JavaScript pour écouter les données saisie de l'utilisateur en temps réel.

5.2.3 Les email de validation et de réinitialisation de mot de passe

A partir de la fenêtre d'authentification on peut récupérer notre mot de passe en cas d'oubli. Pour ce faire, il faut cliqué sur le lien "Mot de passe oublié" puis de saisir l'adresse email. Après cela, le système envoi un email à ce dernier pour réinitialiser le mot de passe.



FIGURE 5.3 – Email de réinitialisation de mot de passe

Le système envoi aussi un email de validation après chaque inscription pour vérifier l'authenticité de l'adresse email.



FIGURE 5.4 – Email de validation de l'adresse email

5.2.4 Le tableau de bord

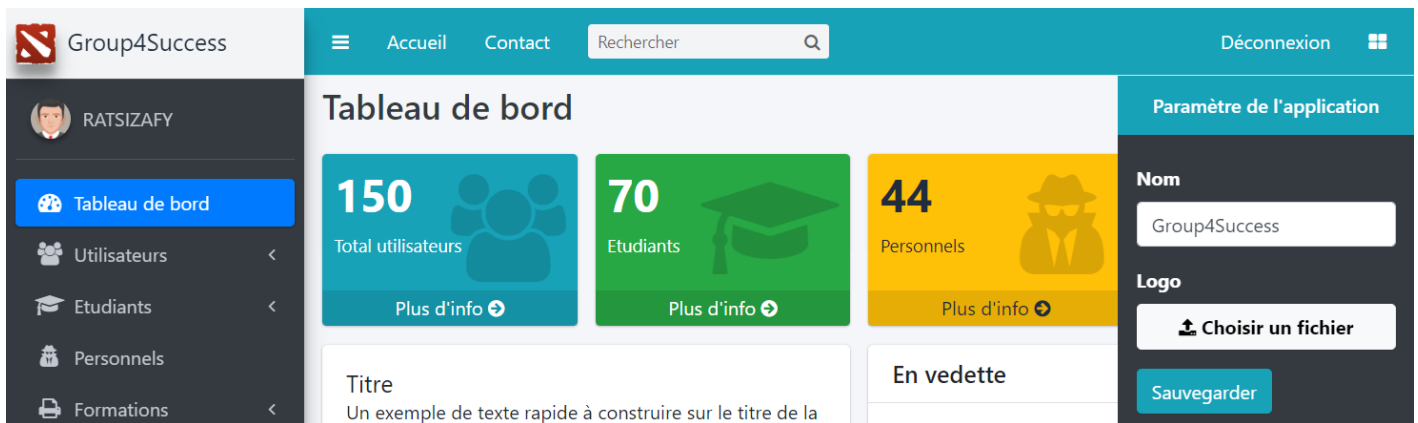


FIGURE 5.5 – Tableau de bord

Après authentification, l'utilisateur sera redirigé directement au tableau de bord orienté rôles qui affiche les chiffres clés de l'établissement et une vue d'ensemble simple et directe des anomalies. Le tableau de bord permettant de zoomer jusqu'au détail et d'utiliser toutes les informations de données de l'établissement. L'architecture du tableau de bord est adaptable et configurable, c'est-à-dire que l'on peut ajouter, supprimer ou déplacer un élément pour le façonner à notre besoin.

La paramètre de l'application, à droite n'est disponible que pour l'administrateur, elle permet de modifier les informations de l'application tels que son nom et le logo.

5.2.5 Liste et ajout d'utilisateur

Pour chaque nouveau utilisateur ajouter, un mail de vérification et réinitialisation de mot de passe sera envoyé à son adresse qu'il puisse s'authentifier à l'application. La modification ou suppression d'un compte utilisateur est accessible en passant par la liste. Cette liste est dynamique, elle offre plusieurs options d'affichage, trie par colonne ou recherche d'un élément spécifique.

Ajout d'utilisateur

Utilisateurs / Ajout

Nom *

Nom

Email *

Email *

Prénom *

Prénom

Date de naissance

jj/mm/aaaa

Rôle

Etudiant(e)

Adresse

Adresse

Situation Familiale

Téléphone

Sexe

☒ Homme ☐ Femme

Photo

Importer

Sauvegarder

FIGURE 5.6 – Ajout d'utilisateur

Liste des utilisateurs

Utilisateurs / Liste

Affiche 10 entrées

Chercher :

Nom	Prénom	Adresse E-mail	Rôle		
ANDRIANIVO	Dany	dany@gmail.com	Enseignant(e)		
HARIFERA	Finihavana	harifera@gmail.com	Etudiant(e)		
HERINJAKA	Ikotokely	kotokely@gmail.com	Gestionnaire Scolarité		
MANITRINIAINA	Jaosolo	manitriniaina@gmail.com	Gestionnaire RH		
RATSIZAFY	Midonique	midoniquebadgeek@gmail.com	Administrateur		

Affiche 1 à 6 de 6 entrées

Précédent 1 Suivant

FIGURE 5.7 – Liste d'utilisateur

5.2.6 Page d'erreur 404

Au cas où un utilisateur essaie d'accéder à un url non valide qui n'existe pas, il sera redirigé directement vers une page d'erreur 404.



FIGURE 5.8 – Page d'erreur 404

5.3 Les services backend de Firebase

5.3.1 Authentification

Firebase fournit des services de backend pour gérer la gestion des utilisateurs. Avec Firebase Authentication, les mots de passe des utilisateurs sont enregistrés de manière sécurisée dans le cloud, haché avec l'algorithme SCRYPT en 64bits (Voir figure 5.10). Ces mots de passe hachés ne sont même pas accessibles à l'administrateur de Firebase, ils sont conservés en sécurité.

5.4 Déploiement de l'application sur Firebase

Après avoir développé une application il faut pouvoir la déployer sur un serveur pour la rendre disponible par les utilisateurs via un internet. Pour cela nous allons utiliser Firebase hosting comme hébergeur.

5.4.1 Préparer l'application pour la production

A la racine de notre application Angular, on exécute la commande suivante :

```
ng build -prod
```

Celle-ci a pour effet de créer un répertoire *dist*. C'est là que se trouve l'application compilée, optimisée et prête à être déployée.

5.4.2 Installer Firebase-CLI

Firebase-CLI est nécessaire pour configurer notre application sur firebase. Pour cela, dans l'invite de commandes en tant qu'Administrateur, il faut entrer la commande suivante :

```
npm install -g firebase-tools
```

Une fois l'installation terminée, il faut se connecter avec notre compte Google :

```
firebase login
```

Cette commande associe notre PC à notre compte Firebase et nous permet alors d'accéder à nos projets.

5.4.3 Configurer l'application Angular

Nous devons initialiser Firebase Hosting avant de pouvoir déployer notre application Angular. Pour ce faire, on exécute la commande suivante :

```
firebase init
```

5.4.4 Déploiement de l'application

Après quelques configuration avec *firebase init*, on peut finalement déployer notre application avec la commande suivante :

```
firebase deploy
```

Le déploiement effectué, notre application est maintenant accessible via l'URL :

"g4sm-mad.firebaseio.com/"

Conclusion générale

Notre projet intitulé Développement de SaaS « Group4SuccessManagement » est destiné pour alléger les efforts de gestion de personnel, soutenir la croissance de l'établissement, stimuler la réussite des étudiants et améliorer l'efficacité administrative.

Contrairement à la majorité des applications classique, notre application de type SaaS fournit des services aux utilisateurs en échange d'abonnement mensuels, elle est installés sur des serveurs distants plutôt que sur la machine de l'utilisateur. En utilisant ce service l'établissement client n'hébergera pas l'application et ne stockera pas ses données en interne. Les utilisateurs devront simplement disposer d'un dispositif muni d'un navigateur web et des codes d'accès au service en ligne pour pouvoir travailler.

En ce qui concerne les étapes de notre travail, nous avons, en premier lieu, effectué une phase d'étude théorique pour déterminer les problématiques liés à la réalisation du projet pour ainsi fixer les solutions que nous devons adapter. En second lieu, nous avons défini les services fournis par le Group4Success en détaillant les solutions à faire pour satisfaire les quatre objectifs de notre application. En troisième lieu, nous avons modélisé l'application par rapport aux besoins fonctionnels et techniques de notre scénario de base. Après quoi, en quatrième partie, nous sommes passés à la réalisation de l'application en faisant soin de choisir l'outil idéal pour se faire. Dans la dernière étape, nous avons présentés l'application à travers des imprimés écrans et enfin le déploiement de celle-ci dans un serveur distant.

L'application ainsi développée fournit des services pour répondre aux besoins évolutifs des établissements d'enseignement supérieur. Elle permet aux établissements abonnés, possédant des clés d'accès à l'application, de gérer, d'une part, les gestions des employés et des étudiants. Ces gestions peuvent être, la création des fiches employeur, gestion des emplois, gestion de rémunération, traitement de masse, contrôles des personnelles et exportation des données, la gestion des formations, évaluation du parcours. D'autre part, l'outil gère l'établissement et l'administration, à savoir : la gestion des inscriptions,

suivis de l'institution, gestion des locaux, automatisation des procédures courantes, outil de tableau de bord.

Malgré quelques difficultés à mettre en œuvre tous les modules, ce travail nous a présenté une réelle occasion d'apprendre et de faire un premier pas vers le monde du SaaS, toujours en évolution. Ce travail fut très instructif, car l'utilisation d'Angular change la manière de navigation des utilisateurs. C'est plus fluide, une meilleure gestion de contenu dynamique et diminution considérable de la vitesse de chargement des pages. En effet, le nombre d'accès au serveur est fortement diminué car la communication se fait majoritairement en mode asynchrone.

À la fin de cet ouvrage, il nous est important de remarquer que des éventuelles améliorations doivent être apportés au présent projet. Dans la gestion des étudiants : nous pouvons envisager d'implanter un outil pour simulation pour auto-évaluer les progrès des étudiants. Pour l'exportation des fichiers, nous pouvons ajouter d'autres type de format autre que le PDF. Toutefois, nous estimons que les travaux demandés dans le cahier de charge ont été accompli à l'exception de la mise en œuvre de quelques modules qui suscitent d'autres études et analyses.

Bibliographie

- [1] Généralité sur la gestion des ressources humaines. *Robert* NIKIKO. IRIMAG - Licence en Administration et Management.
- [2] Deviens un ninja avec Angular. *Ninja Squad*. Learnangularjs, 2016.
- [3] Learn Angular 2 step by step. *Questpond*, 2016.
- [4] Cours de base de données. *R. Njakarison Menja*, 2013.
- [5] Les établissements d'enseignement supérieur : Structure et fonctionnement, 2017. Page 12-13. Brochures de PARFAIRE. Chapitre structure et fonctionnement. creativecommons.org/licenses/by-nc-nd/4.0/fr/.
- [6] Firebase, 2017. Gregory HOWARD - Alix DUCROS. *Build Extraordinary Apps*.
- [7] Angular 2 & 4 maîtriser le Framework Front-End de Google. ORSYS, La Grande Arche.
- [8] Firebase Realtime Database. *Landon COX*, 2017

Webographie

- [9] Études supérieures, 06 Mars 2016. Wikipédia. fr.wikipedia.org.
- [10] Réforme Licence-Master-Doctorat, Octobre 2017. Wikipédia. fr.wikipedia.org.
- [11] Gestion des ressources humaines, Septembre 2017. Wikipédia. fr.wikipedia.org.
- [12] Le cloud computing, 21 Mars 2016. Wikipédia. fr.wikipedia.org.
- [13] Le logiciel en tant que service, Juin 2016. Wikipédia. fr.wikipedia.org.
- [14] FCIC Média. Le mode SaaS, 2017. www.lecoindesentrepreneurs.fr/le-mode-saas/.
- [15] Secteur de l'éducation, 2017. Unit4. Erp. www.unit4.com/fr/secteurs/education.
- [16] Firebase, Novembre 2017. Wikipédia. fr.wikipedia.org/wiki/Firebase.
- [17] Les raisons clés de développer avec angular, 2016. Caroline PHILLIPS. www.pic.digital/blog/396-angularjs-pourquoi.
- [18] Initiation à la conception de bases de données avec merise, Février 2012. Idriss NEUMANN. ineumann.developpez.com/tutoriels/merise/initiationmerise/.
- [19] How to Use Local Storage and Session Storage In Angular 4, Janvier 2018. JsTutorials Team. www.js-tutorials.com/javascript-tutorial/use-localstorage-sessionstorage-using-webstorage-angular4/.
- [20] Angular Authentification Tutorial, 5 Décembre 2017. Ado KUKIC. auth0.com/blog/angular-2-authentication/.
- [21] Firebase Guides, 8 Mai 2018. firebase.google.com/docs/guides/.
- [22] Firebase Authentication, 8 Mai 2018. <https://firebase.google.com/docs/auth/>.
- [23] One framework Mobile and Desktop, Angular. angular.io.
- [24] Bootstrap Documentation, v4.1. Core Team. <https://getbootstrap.com/docs/4.1/>.
- [25] jQuery Tutorial, 2018. www.w3schools.com/Jquery/default.asp.
- [26] SweetAlert, Hand-crafted, by Tristan EDWARDS. sweetalert.js.org/guides/.

Annexes

A Code TypeScript : Authentication

```
onSubmit(formData) {
  this.errorMessage = "";
  if (formData.valid) {
    //console.log(formData.value);
    this.af.auth.signInWithEmailAndPassword(this.email, this.password)
      .then((success) => {
        var user = this.af.auth.currentUser;

        if (user.emailVerified) {
          this.isError = false;
          this.router.navigate(['/dashboard']);
          window.location.reload();
        }
        else {
          swal({
            title: "Compte désactivé!",
            text: "Veuillez consulter votre adresse e-mail pour
              activer votre compte",
            icon: "warning",
            buttons: ["Annuler", "Renvoyer l'email de confirmation"],
            dangerMode: false,
          })
            .then((willDelete) => {
              if (willDelete) {
                this.sendVerification();
              } else {
                this.logout();
              }
            });

          this.email = null;
          this.password = null;
        }
      })
      .catch((error) => {
        this.isError = true;
        console.log('Error Name', error['code']);
      });
  }
}
```

```

        console.log('Error Message', error['message']);
        this.getError(error);
    })
}
}

```

B Upload de fichier

```

pushUpload(upload: Upload, basePath) {
    const storageRef = firebase.storage().ref();
    const uploadTask = storageRef.child(`${basePath}/${new Date()
        .getTime()}_${upload.file.name}`)
        .put(upload.file);

    uploadTask.on(firebase.storage.TaskEvent.STATE_CHANGED,
        // trois observateur
        // 1.) changement d'etat observateur
        (snapshot) => {
            // upload en cours
            upload.progress =
                (snapshot.bytesTransferred / snapshot.totalBytes) * 100;
            console.log('Progress : ', upload.progress);
        },
        // 2.) erreur observateur
        (error) => {
            // upload failed
            console.log(error)
        },
        // 3.) success observateur
        (): any => {
            // upload success
            upload.url = uploadTask.snapshot.downloadURL;
            upload.name = upload.file.name;
            upload.createdOn = new Date();
            //this.saveFileData(upload, basePath);
        }
    )
};

```

C Divers fonction de validation de formulaires

```

export function RequiredField(c: AbstractControl) {
    if (c.value.trim() == '' || c.value.trim() == null) {
        return {'required': true}
    }
}

export function MinLenght(nbr: number) {

```

```
    return (c: AbstractControl) => {
      if (c.value.trim().length < nbr) {
        return {'minLenght': true}
      }
    }
  }

export function EmailValidation(c: AbstractControl) {
  let isValid = /^[_a-zA-Z0-9]+(\.[_a-zA-Z0-9]+)*@[a-zA-Z0-9-]+
(\.[a-zA-Z0-9-]+)*(\.[a-zA-Z]{2,4})$/ .test(c.value);
  if (!isValid) {
    return {'emailFormat': true};
  }
}

export function TelValidation(c: AbstractControl) {
  let isValid = /^032|033|034 \d{2} \d{3} \d{2}/.test(c.value);
  if (!isValid) {
    return {'telFormat': true};
  }
}

export function PasswordValidation(c: AbstractControl) {
  let isValid = /^(?=.*[a-z])(?=.*[A-Z])(?=.*[0-9])(?=.*{6,100})/
.test(c.value);
  if (!isValid) {
    return {'passwordValid': true};
  }
}
```

RÉSUMÉ

Le système d'information étudiants des établissements d'enseignement supérieur ne répond plus aux besoins du modèle d'activité existante. D'où l'idée de ce travail de mémoire de développer un SaaS (Software as a Service ou logiciel en tant que service) dénommé "Group4Success" qui est destiné pour : alléger les efforts de gestion de personnel, soutenir la croissance de l'établissement, stimuler la réussite des étudiants et améliorer l'efficacité administrative.

L'application est développée avec Angular, la nouvelle vague de frameworks JavaScript portée par Google. Elle est installée sur un serveur distant dans Firebase, plutôt que sur la machine de l'utilisateur, ainsi l'établissement client n'hébergera pas l'application et ne stockera pas ses données en interne.

En bref, le "Group4Success" est une application web de type SaaS qui va fournir des services pour répondre aux besoins évolutifs des établissements d'enseignement supérieur.

ABSTRACT

The student information system of higher education institutions no longer meets the needs of the existing business model. Hence the idea of this work of memory to develop a SaaS (Software as a Service) called "Group4Success" which is intended to : lighten the efforts of personnel management, support the growth of the establishment, stimulate student success and improve administrative efficiency.

The application is developed with Angular, the new wave of JavaScript frameworks promoted by Google. It is installed on a remote server in Firebase, rather than on the user's machine, so the client institution will not host the application and will not store its data internally.

In short, the "Group4Success" is a SaaS-based web application that will provide services to meet the evolving needs of higher education institutions.