



Ecole Polytechnique de l'Université de Tours
Département Informatique
64 avenue Jean Portalis
37200 Tours, France
Tél. +33 (0)2 47 36 14 14
polytech.univ-tours.fr

Projet Recherche & Développement 2019-2020

LIFAT Manager : Un site et des services Web pour la gestion du laboratoire



Entreprise

Polytech



Tuteur entreprise

Jean-Yves RAMEL

Étudiant

Qingling MENG (DI5)

Tuteur académique

Jean-Yves RAMEL

Liste des intervenants

Entreprise

Polytech
64 avenue Jean Portalis
37200 Tours, France
polytech.univ-tours.fr



Nom	Email	Qualité
Qingling MENG	qingling.meng@etu.univ-tours.fr	Étudiant DI5
Jean-Yves RAMEL	jean-yves.ramel@univ-tours.fr	Tuteur académique, Département Informatique
Jean-Yves RAMEL	jean-yves.ramel@univ-tours.fr	Tuteur entreprise



Avertissement

Ce document a été rédigé par Qingling MENG susnommé l'auteur.

L'entreprise Polytech est représentée par Jean-Yves RAMEL susnommé le tuteur entreprise.

L'Ecole Polytechnique de l'Université de Tours est représentée par Jean-Yves RAMEL susnommé le tuteur académique.

Par l'utilisation de ce modèle de document, l'ensemble des intervenants du projet acceptent les conditions définies ci-après.

L'auteur reconnaît assumer l'entière responsabilité du contenu du document ainsi que toutes suites judiciaires qui pourraient en découler du fait du non respect des lois ou des droits d'auteur.

L'auteur atteste que les propos du document sont sincères et assume l'entière responsabilité de la véracité des propos.

L'auteur atteste ne pas s'approprier le travail d'autrui et que le document ne contient aucun plagiat.

L'auteur atteste que le document ne contient aucun propos diffamatoire ou condamnable devant la loi.

L'auteur reconnaît qu'il ne peut diffuser ce document en partie ou en intégralité sous quelque forme que ce soit sans l'accord préalable du tuteur académique et de l'entreprise.

L'auteur autorise l'école polytechnique de l'université François Rabelais de Tours à diffuser tout ou partie de ce document, sous quelque forme que ce soit, y compris après transformation en citant la source. Cette diffusion devra se faire gracieusement et être accompagnée du présent avertissement.



Pour citer ce document

Qingling MENG, *LIFAT Manager : Un site et des services Web pour la gestion du laboratoire:*, Projet Recherche & Développement, Ecole Polytechnique de l'Université de Tours, Tours, France, 2019-2020.

```
@mastersthesis{
  author={MENG, Qingling},
  title={LIFAT Manager : Un site et des services Web pour la gestion du laboratoire: },
  type={Projet Recherche \& Développement},
  school={Ecole Polytechnique de l'Université de Tours},
  address={Tours, France},
  year={2019-2020}
}
```

Table des matières

Liste des intervenants	a
Avertissement	b
Pour citer ce document	c
Table des matières	i
Table des figures	vi
1 Introduction	1
1 Acteurs, enjeux et contexte	1
2 Objectifs	1
3 Hypothèses	2
4 Bases méthodologiques.....	2
2 Description générale	3
1 Environnement du projet	3
2 Caractéristiques des utilisateurs	3
3 Fonctionnalités du système	4
4 Structure générale du système.....	4
3 Etat de l'art	6
1 Techniques standards 2D/3D	8
Graphique linéaire.....	8
Histogramme.....	8
Bar Plot.....	9
Scatterplot Matrices	9

	Camembert	9
2	Techniques orientées pixel et construction d'un modèle de données.....	10
	Segments de cercle	10
	Barres rectangulaires.....	10
	Perception-Based Classification (PBC)	11
3	Techniques matricielles et construction interactive d'un modèle des données.....	12
	CIAD (Construction Interactive d'Arbre de Décision).....	12
	UserClassifier de WEKA	12
4	Analyse et conception	14
1	Modélisation logicielle.....	14
1.1	Modele	14
1.2	Contrôleur	15
1.3	View.....	16
5	Mise en oeuvre	18
1	Introduction de la création d'une mission.....	18
2	Librairies utilisées	19
2.1	FPDF.....	19
	Introduction	19
	L'utilisation.....	19
2.2	EmailComponent	22
3	IHM de mission	23
3.1	Lister tous les missions	23
3.2	Créer une mission	24
3.3	Modifier une mission.....	27
3.4	Supprimer une mission	28
3.5	Visualiser une mission	28
3.6	Chercher une mission.....	28
3.7	Valider une mission.....	29
6	Bilan et conclusion	31
1	Bilan du semestre 9	31
	Les taches finies Pendant la S9	31
	La tache en retard.....	31
	Les taches a faire :.....	31
2	Planning du semestre 10	32
3	Bilan du semestre 10.....	32
	Les tâches finies Pendant la 10	32
	La tâche en retard.....	32

	Les tâches a faire	32
4	Bilan sur la qualité	32
5	Bilan auto-critique.....	33
Annexes		34
A Planification		35
1	Aperçu de gestion de projet	35
2	Découpage du projet en tâches.....	35
	Tâche 1 : Lecture et comprehension des documents existants liés au projet	36
	Tâche 2 : Analyse des besoins.....	36
	Tâche 3 : Rédaction le bilan de existant	36
	Tâche 4 : Rédaction le cahier de specification	37
	Tâche 5 : Etudier la fouille visuelle de données et rédaction l'état de l'art.....	37
	Tâche 6 : Rédaction le rapport	37
	Tâche 7 : Preparation soutenance mi-parcours.....	37
	Tâche 8 : Réalisation de la génération le pdf d'ordre de mission PDF1.	37
	Tâche 9 : Réalisation de la création de la mission.	37
	Tâche 10 : Réalisation d'envoi un email contenant PDF 1 au chef d'équipe.	37
	Tâche 11 : Améliorer le PDF1(contient la signature, le transport etc.)	38
	Tâche 12 : Réalisation de la génération le pdf de la validité de mission PDF2.	38
	Tâche 13 : Réalisation de la validité l'ordre de mission.	38
	Tâche 14 : Envoyer un email contenant PDF1 et PDF2 au secrétaire.	38
	Tâche 15 : Réalisation de la modification/ la recherche/ la suppression une mission.....	38
	Tâche 16 : Déployer la version actuel au serveur	38
	Tâche 17 : Ajouter une module de visualisation des statistique	39
	Tâche 18 : Tests et correction	39
	Tâche 19 : Rédaction du rapport final et le slide.....	39
B Description des interfaces externes du logiciel		40
1	Interfaces matériel/logiciel	40
2	Interfaces homme/machine.....	40
	- Un onglet «MesMissions» à ce Tool.....	40
	- A la interface connexion, ajouter une partie «Connexion par ENT/CAS»	41

	- Une option dans l'onglet «Rapport d'activité annuel» (Continue) ..	41
	- Un onglet «Publications» (Continue)	41
3	Interfaces logiciel/logiciel	42
C	Specifications fonctionnelles	44
1	Définition de la fonction 1 : Gestion de demand missions	44
	Présenter la fonction 1 :	44
	Description de la fonction 1 :	44
2	Définition de la fonction 2 : Visualisation des statistiques	46
	Présenter la fonction 2 :	46
	Description de la fonction 2 :	47
3	Définition de la fonction 3 : Connection via CAS (Continue)	49
	Présenter la fonction 3 :	49
	Description de la fonction 3 :	49
4	Définition de la fonction 4 : Verification du mail(Continue)	50
	Présenter la fonction 4 :	50
	Description de la fonction 4 :	50
5	Définition de la fonction 5 : Génération automatique de rapport d'activité annuel(Continue)	50
	Présenter la fonction :	50
	Description de la fonction 5 :	50
6	Définition de la fonction 6 : Gestion des publications (Continue)	50
	Présenter la fonction :	50
	Description de la fonction 6 :	51
D	Spécifications non fonctionnelles	52
1	Contraintes de développement et conception	52
2	Contraintes de fonctionnement et d'exploitation	52
2.1	Performances	52
2.2	Capacités	52
2.3	Contrôlabilité	52
2.4	Sécurité	52
E	Cahier du développeur	54
1	Aperçu de la diagramme de classe entier	54
2	Descriptions détaillées des classes du package « model »	55
	Classe « Mission »	55
	Classe « Transport »	55
	Classe « Lieu »	55
	Classe « Motif »	56

	Classe « Publication »	56
3	Descriptions détaillées des classes du package « contrôleur »	56
	Classe « MissionController »	56
	Classe « ExportController »	58
	Classe « PublicationController »(Continue)	59
	Classe « TransportController »	59
	Classe « LieuController »	60
	Classe « MotifController ».....	60
4	Descriptions détaillées des classes du package « view »	61
	Dans le dossier Template\Missions	61
	Dans le dossier Template\Lieux	61
	Dans le dossier Template\Transports	61
	Dans le dossier Template\Motifs	61
F	Document d'installation	62
1	Prérequis de CakePHP.....	62
	Apache / MySQL / PHP.....	62
	Extensions de PHP	62
	Git.....	63
	Composer.....	63
2	Mise en place du projet.....	63
	Télécharger le projet.....	63
	Base de donnees.....	64
	Finalisation de l'application.....	64
3	Accès à l'application	65
4	Mettre à jour en production.....	65
G	Document d'utilisation	66
H	Cahier de test	67
1	Tests unitaires de la classe« MissionsController»	68
	Bibliographie	76

Table des figures

2 Description générale

2.1	Diagramme des cas d'utilisation.....	4
2.2	Diagramme de classe.....	5

3 Etat de l'art

3.1	Fouille de données et visualisation	7
3.2	Le diagramme linéaire ci-dessus montre les fluctuations des précipitations d'une ville à une autre.....	8
3.3	L'histogramme montre la distribution des effectif selon l'âge.....	8
3.4	Le bar plot montre Pourcentage de la poste de développement senior de 3 sociétés	9
3.5	Le scatter plot matrice montre la corrélation entre deux variables continues.....	9
3.6	Le camembert montre le pourcentage de chaque option dans le total des résultats du sondage	9
3.7	Représentation segments de cercles	10
3.8	Représentation en barres rectangulaires	10
3.9	Exemple de représentation d'un ensemble de données.....	11
3.10	Représentation du processus de coupes successives	11
3.11	Représentation de l'ensemble de données segment avec CIAD.....	12
3.12	Construction interactive du modèle de données avec CIAD	12
3.13	Représentation et construction interactive du modèle de données avec UserClassifier de WEKA.....	13

4 Analyse et conception

4.1	Le diagramme de classeque les classes qui sont en attente de mise en oeuvre	14
4.2	La partie « Modele » du diagramme de classe	15

4.3	La partie « Contrôleur » du diagramme de classe.....	16
4.4	La partie « View » du diagramme de classe	16
5	Mise en oeuvre	
5.1	Le diagramme temporel	18
5.2	l'interface des missions d'un membre	24
5.3	l'interface d'ajouter d'un mission	25
5.4	l'interface d'email au chef d'équipe.....	26
5.5	l'interface de PDF ordre de mission	26
5.6	l'interface de modification de mission	27
5.7	l'interface de suppression de mission	28
5.8	l'interface de recherche d'une mission	28
5.9	l'interface de validité d'une mission	29
5.10	l'interface de email aux secrétaires	29
5.11	l'interface de PDF validité de mission PDF2	30
A	Planification	
A.1	Le diagramme de Gantt Initial.....	35
A.2	Le diagramme de Gantt Final	35
A.3	Les taches dans le diagramme de Gantt	36
B	Description des interfaces externes du logiciel	
B.1	L'interface de la gestion de «MesMissions».....	40
B.2	L'interface de ENT/CAS	41
B.3	L'interface du rapport annuel.....	41
B.4	L'interface de la gestion «Publication»	42
C	Specifications fonctionnelles	
C.1	Toutes les missions effectuées par le laboratoire en 2019.....	47
C.2	Nombre de publications publiées par LIFAT Labs en 2014 - 2019	48
C.3	Top 10 du nombre de projets associés à chaque mission en 2019.....	48
C.4	2019 pourcentage du motif de mission.....	49
C.5	Le graphe de tendance de recherche dans le domaine AI en 2010-2019	49
E	Cahier du développeur	
E.1	Le diagramme de classe entier.....	54

H Cahier de test

H.1 L'exemple de test unitaire avec PHPUnit	68
H.2 Le résultat de test unitaire avec succès.....	68
H.3 Le résultat de test unitaire échoués	68

1

Introduction

1 Acteurs, enjeux et contexte

Ce document présente le site permettant de gérer le Laboratoire d'Informatique Fondamentale et Appliquée de Tours (LIFAT). Le site peut gérer les membre , les équipes, les projets, thèses et lieux de travail de la Laboratoire. De plus, il permet aux membres du laboratoire de gérer les ordres de Mission, ainsi que de gérer les publications publiées.

Le besoin du projet a été émis par le Laboratoire d'Informatique Fondamentale et Appliquée de Tours (LIFAT). Le maitre d'ouvrage est represente par Monsieur Jean-Yves Ramel. Donc, les roles de ce projet sont :

- Client :M. Jean-Yves RAMEL(Le représentant du laboratoire LIFAT)
- MOA : M. Jean-Yves RAMEL (le professeur en departement d'informatique)
- MOE : Qingling MENG

Ce projet ayant fait le sujet du projet collectif de l'année dernière, les fonctionnalités bases du site a été mise en œuvre et le site est prêt à être utilisé. Cependant, certaines fonctions doivent encore être améliorées et la nécessité d'intégrer des contrôleurs externes connexes dans le projet, ce qui peut rendre la gestion du laboratoire plus complète.

2 Objectifs

L'objectif principal du projet est d'améliorer un système d'information pour le LIFAT sous forme d'application web. En plus des fonctions de base qui ont été mises en œuvre, LIFAT pourra également utiliser cette plate-forme directement pour gérer des missions.

Afin de protéger la sécurité du compte, lorsque l'utilisateur s'enregistre le compte, la plateforme effectue la vérification de l'email sur l'email enregistrée. Parallèlement, la plate-forme fournira une méthode de connexion via CAS, éliminant ainsi le processus d'inscription.

Afin de rendre les données plus claires, je vais également ajouter un module de visualisation de données au système. Grâce à la connaissance pertinente de l'exploration de données, les données principales sont présentées sous forme de graphiques ou de tableaux, ce qui permet aux utilisateurs de saisir plus rapidement les informations importantes du système.

En même temps, le système générera automatiquement le rapport statistique annuel du laboratoire, de sorte que le contenu du travail de chaque année puisse être vu en un coup d'œil, y compris les projets, les thèses, les missions etc.

Le système ajoute un module de gestion des publications. Uploader la publication, modifier la publication, et lister les publications etc. Parce que il y a déjà un site opérationnel a réalisé cette fonction. Donc, c'est facultatif.

3 Hypothèses

Si le temps ne suffit pas, je réaliserai principalement la gestion de mission. Assurez la sécurité et l'intégrité de cette partie.

4 Bases méthodologiques

Dans le cas du développement d'une plateforme web, j'ai décidé d'utiliser la langage PHP.

Je utilise au long de ce projet les outils suivants :

- Pour le versionning du projet : GitHub
- Pour le transfert de document avec le client : Google Drive
- Pour le gestion du déroulement du projet : Le diagramme de Gantt
- Pour la rédaction de rapport : LaTeX

2

Description générale

1 Environnement du projet

Ce projet a pour but d'améliorer le système LIFAT Manager, et intégrer le site existant ODM sous une même entité. Cette application sera réalisée avec des solutions web et hébergée sur les serveurs de l'Université de Tours.

L'environnement de développement consiste aux points suivants :

- Pour la création, la gestion et l'administration de la base de données : phpMyAdmin
- Pour le php framework de projet : cake 3.7
- Pour le système d'exploitation : Debian 8.9
- Pour l'environnement de développement : LAMP (Linux + Apache + Mysql + Php)

2 Caractéristiques des utilisateurs

On peut noter plusieurs types d'utilisateurs, et les droits de utilisateur selon leur type :

Rôle 1 Membres extérieurs :

- Ajouter/modifier son profil
- Consulter les données statistiques
- Consulter le rapport d'activité annuel

Rôle 2 Membres du laboratoire :

- Les droits du membre extérieur
- Ajouter/modifier une thèse et les taux d'encadrement
- Ajouter/modifier/supprimer un document partagé(qu'il a ajouté)
- Ajouter/modifier/supprimer un projet
- Ajouter/supprimer/modifier une mission
- Ajouter/supprimer/modifier une publication

Rôle 3 Chef de l'équipe :

- Les droits du membre du laboratoire
- Valider les missions

Rôle 4 Administrateur (chef du laboratoire, secrétaire)

- L'administrateur aura tous les droits sur le système

3 Fonctionnalités du système



Figure 2.1 – Diagramme des cas d'utilisation

Les objectifs principaux du système ce que je vais réaliser sont les suivants (dans l'ordre d'importance) :

- Gestion (ajouter / modifier / supprimer) les ordres de missions
- Gestion (ajouter / modifier / supprimer) les publications (continue)
- Génération automatique de rapport d'activité annuel (continue)
- Ajout de modules de visualisation des statistiques

4 Structure générale du système

Après avoir analysé les documents mis à mon disposition et après plusieurs échanges avec mon client, j'ai rédigé le diagramme représentant la base de données. J'ajoute une entité publication.

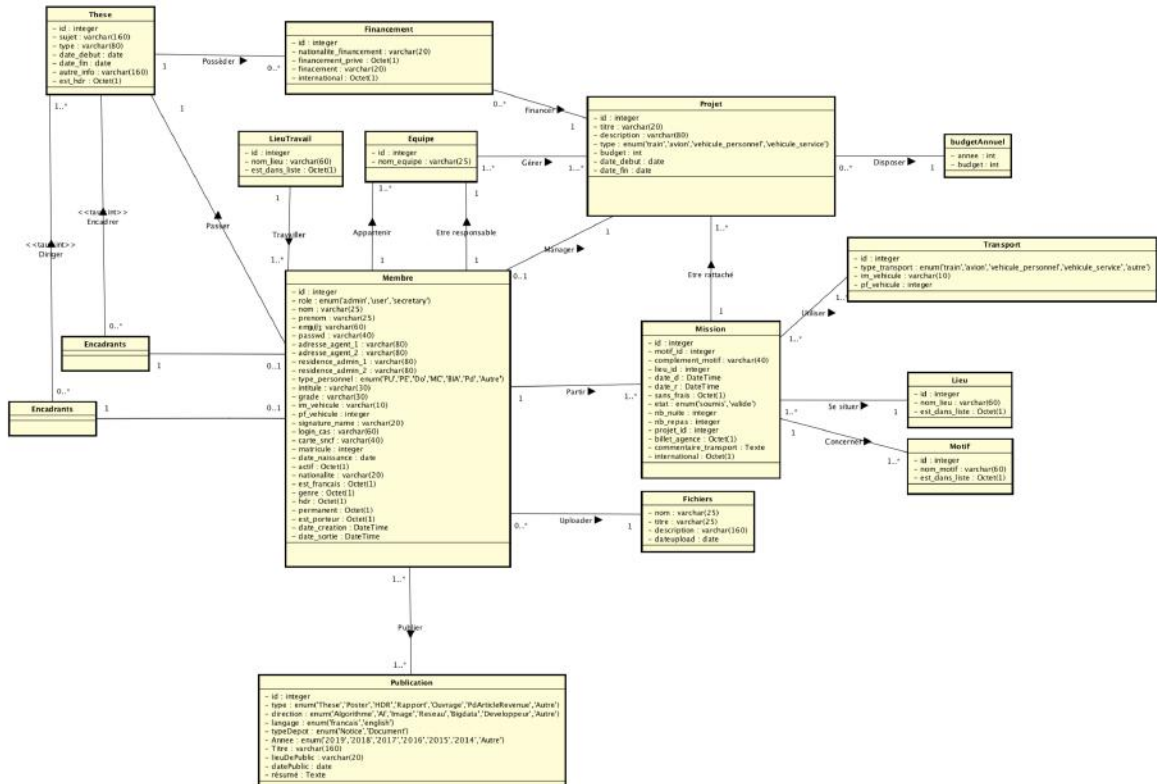


Figure 2.2 – Diagramme de classe

3

Etat de l'art

Les premiers travaux introduisant le terme de fouille visuelle de données datent de la fin des années 1990 [Keim et Kriegel, 1996], [Brunk et al., 1997], [Cox et al, 1997] et [Inselberg, 1998]. Dans un premier temps, toute l'attention a été portée vers le développement de techniques performantes et innovantes d'accès, de représentation graphique et de traitement des données. La masse de données disponible dans le monde ne cesse d'augmenter. La FVD utilise la visualisation comme canal de communication pour la découverte de corrélations dans les données. L'espace disponible sur un écran pour la représentation de ces données est limitée. Une première préoccupation concerne l'amélioration des techniques existantes de stockage, d'accès et de représentation graphique des données. Dans cet ordre d'idées, [Keim, 1996] a utilisé des techniques orientées pixel pour la représentation des données multidimensionnelles (VisDB). L'outil segments de cercle de [Ankerst et al., 1999] utilise aussi le même principe. Un autre aspect abordé dans ce domaine a conduit à une modélisation de la tâche de FVD [Ankerst, 2000]. Le modèle de tâche de FVD présenté par Ankerst possède 3 variantes (figure 1.17) suivant le mode d'utilisation des représentations graphiques. Pour chacune de ces variantes, on dénote des phases de visualisation des données, d'application de méthodes d'analyse de données avant d'aboutir à la connaissance (modèle des données). En effet, lorsqu'on se sert de la visualisation comme support en fouille de données, après la sélection des données à exploiter (première étape des 3 variantes), une alternative se présente : soit l'utilisateur sélectionne et exécute un algorithme automatique de fouille de données, soit il procède à une visualisation (exploration) de l'ensemble de données. La visualisation peut être suivie de l'application d'une méthode automatique (ou interactive) de construction du modèle des données. L'étape suivante consiste en la visualisation des résultats. On assiste enfin à une évaluation puis à une exploitation de ces résultats considérés comme des connaissances nouvelles. Plus explicitement, les variantes de la figure 1 (modèle de tâche de Ankerst) correspondent à :

- L'utilisation de la visualisation pour l'interprétation des résultats d'un algorithme automatique de fouille de données
- L'utilisation de la visualisation pour une exploration des données qui permet à l'utilisateur d'avoir une idée générale des données, suivi d'une application d'un algorithme automatique de fouille de données à l'ensemble de données et de l'utilisation de la visualisation pour une interprétation des résultats finaux.
- Dans le dernier cas, la représentation graphique sert de support aux traitements. Afin de construire le modèle des données, l'utilisateur interagit avec la représentation visuelle et procède à des traitements successifs qui lui permettent de construire un modèle des données.

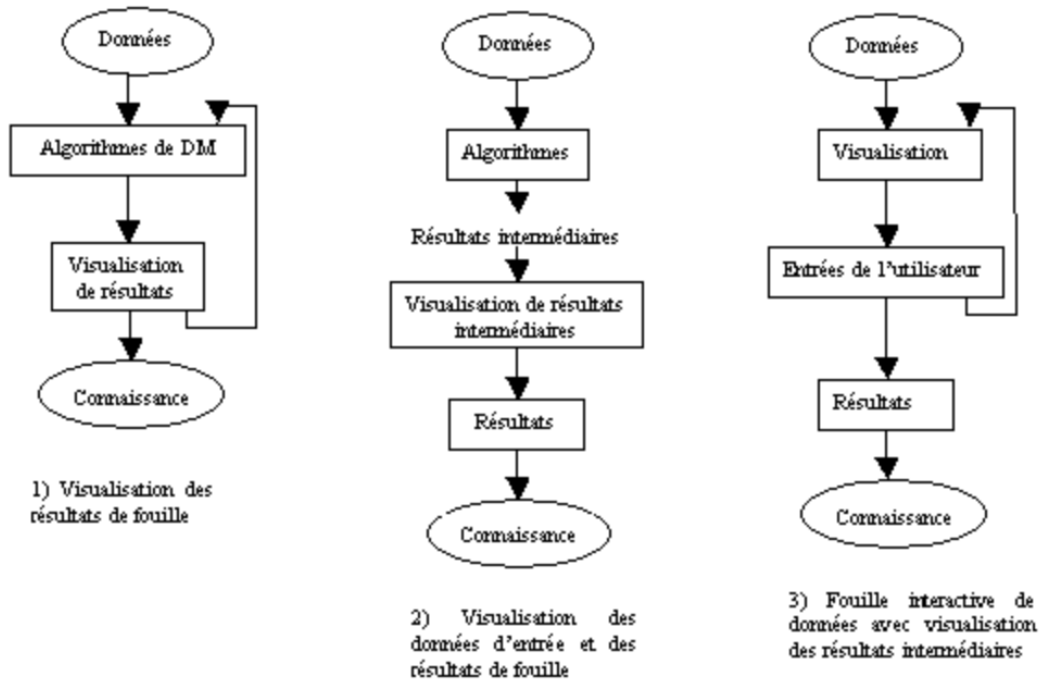


Figure 3.1 – Fouille de données et visualisation

Pour les deux premières variantes du modèle de Ankerst, la plupart de techniques de représentations graphiques utilisées ont été présentées en début de ce chapitre. Nous allons à présent dresser un état de l'art des différentes techniques de visualisation qui permettent de découvrir des corrélations dans les données de façon interactive (variante 3 du modèle de Ankerst), de construire un modèle de données et de représenter les résultats issus de la construction du modèle des données.

Le modèle de tâche en FVD montre que la construction du modèle de données peut se faire de façon automatique (deux premières variantes du modèle de Ankerst) ou alors de manière interactive (troisième variante du modèle de Ankerst). Cet état de l'art est essentiellement basé sur la construction interactive du modèle de données qui possède de nombreux avantages par rapport aux algorithmes automatiques couplés ou non aux méthodes de représentation graphique. L'exécution des algorithmes automatiques d'analyse de données nécessite une étape préalable de paramétrage, ce qui n'est pas le cas en construction interactive du modèle de données. En effet, l'algorithme automatique se comporte comme une boîte noire recevant en entrée des données et fournissant en sortie un modèle de ces données. L'utilisateur ne participe pas à la construction de ce modèle, ce qui pourrait avoir une incidence sur le degré de confiance qu'il accordera au résultat. La vision humaine peut servir à capturer des corrélations complexes dans les ensembles de données au travers de représentations graphiques. Si l'utilisateur de l'outil de fouille interactive de données est un spécialiste du domaine des données, il peut utiliser ses connaissances du domaine de données durant le processus de fouille et non seulement au moment de l'interprétation des résultats (cas des algorithmes automatiques). La confiance au modèle de données ainsi construit est élevée car l'utilisateur a participé à sa construction. Le temps de traitement avec l'algorithme interactif peut s'avérer long, surtout pour de grands ensembles de données.

Les différentes variantes du modèle de Ankerst montrent aussi que l'étape de construction du modèle de données est suivie d'une étape de post traitement au cours de laquelle l'utilisateur peut avoir recours aux techniques de visualisation. A cet effet, il existe des techniques de visualisation telles que CUBEVIS [Poulet, 2001] ou Grand tour [Asimov, 1985]. Nous nous limitons ici à l'étape de construction du modèle de données.

1 Techniques standards 2D/3D

Visualiser de statistique sous la forme de graphiques et de graphiques bien connus, ce qui est le plus courant méthode dans notre vie quotidien. 5 types le plus courantes graphiques de Data Visualisation est comme ci-dessous.

Graphique linéaire

Les graphiques linéaires peuvent afficher des données continues qui changent avec le temps (en fonction des paramètres de mise à l'échelle courants). Ils sont donc parfaits pour afficher les tendances des données à intervalles égaux.

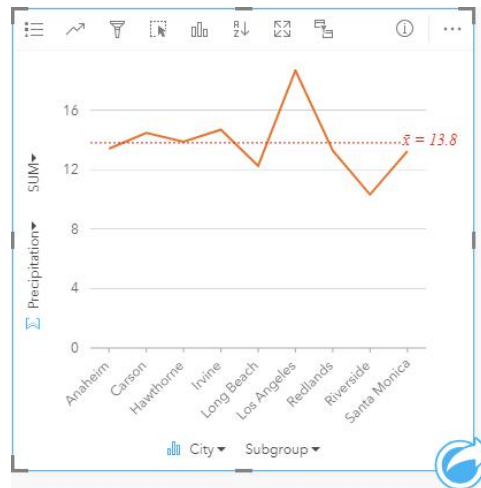


Figure 3.2 – Le diagramme linéaire ci-dessus montre les fluctuations des précipitations d'une ville à une autre

Histogramme

Un histogramme est un graphique permettant de représenter la répartition des valeurs d'une variable continue (Continuous Data). Chaque colonne de l'histogramme représente un intervalle de valeurs. La hauteur des colonnes indique le nombre d'instances dans cet intervalle. L'examen de l'histogramme permet de se faire une idée claire sur la distribution des valeurs de la feature analysée.

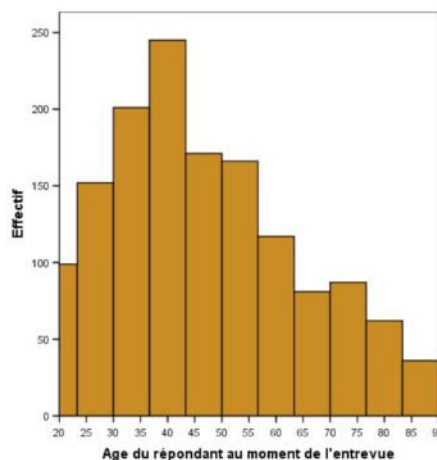


Figure 3.3 – L'histogramme montre la distribution des effectif selon l'âge

Bar Plot

Les Bar Plots sont utilisés pour visualiser des données qualitatives (Categorical Data). Chaque "Barre" d'un bar plot représente une catégorie (modalité) et la hauteur de la barre indique la taille du groupe faisant partie de cette catégorie.

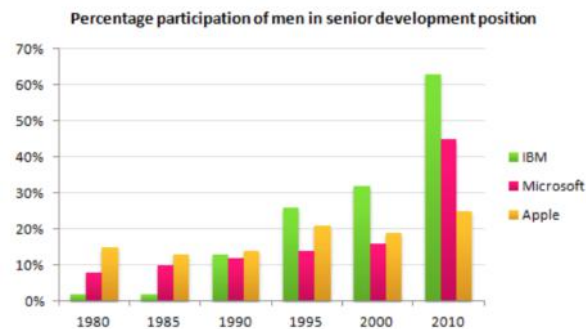


Figure 3.4 – Le bar plot montre Pourcentage de la poste de développement senior de 3 sociétés

Scatterplot Matrices

Les Scatter Plot Matrices (Diagrammes de dispersion) permettent de visualiser la corrélation entre deux variables continues. On met la première feature sur l'axe des abscisses (X) et la deuxième sur les ordonnées (Y). La dispersion des points indique la relation entre les deux features.

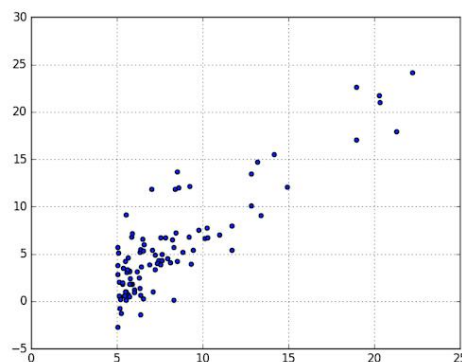


Figure 3.5 – Le scatter plot matrice montre la corrélation entre deux variables continues

Camembert

Les « camemberts » permettent de représenter des proportions et des pourcentages.

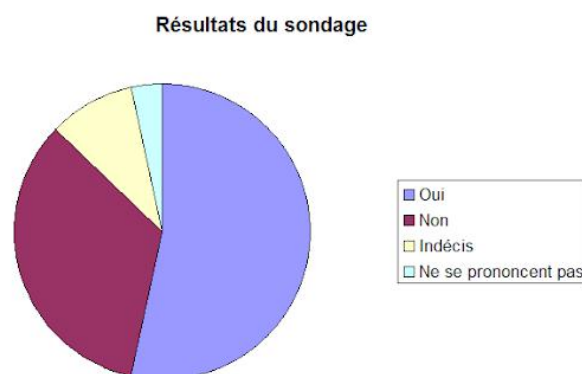


Figure 3.6 – Le camembert montre le pourcentage de chaque option dans le total des résultats du sondage

2 Techniques orientées pixel et construction d'un modèle de données

Les méthodes de visualisation qui utilisent cette technique sont : PBC [Ankerst, 1999], segments de cercle [Ankerst et Keim, 1996]) et DTViz [Han et Cercone, 2001].

Segments de cercle

Toute la base de données est représentée dans un cercle. Le cercle est divisé en segments, un segment pour chaque attribut. Dans le segment, chaque valeur d'attribut est représentée par une valeur de pixel. L'arrangement des pixels va du centre du cercle vers l'extérieur.

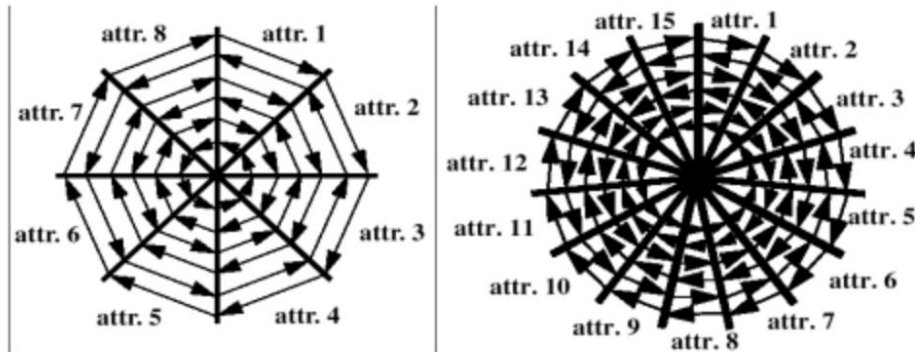


Figure 3.7 – Représentation segments de cercles

Si on opère un changement dans le parcours utilisé pour remplir les pixels dans la représentation sous forme de segments de cercle, on obtient une représentation sous forme de barres rectangulaires. Dans ces modes de visualisation, les données sont triées avant d'être affichées sous forme de pixels.

Barres rectangulaires

La figure 3.8 illustre la représentation en barres rectangulaires.

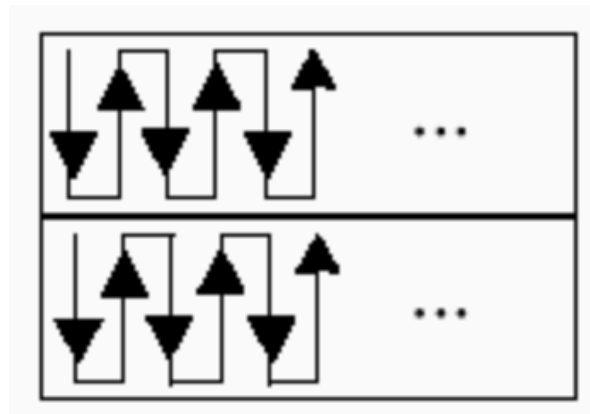


Figure 3.8 – Représentation en barres rectangulaires

Segments de cercle et barres rectangulaires permettent de visualiser des ensembles de données relativement grands. L'inconvénient de ce type d'approche tient au fait que suite au tri des données avant visualisation, l'information portant sur la distance des données est perdue. De plus, cette opération de tri constitue un coût supplémentaire dans les traitements par rapport aux techniques de représentation de type matrice de scatter plot.

Perception-Based Classification (PBC)

PBC est un algorithme interactif de construction d'arbres de décision qui utilise le principe de la pixellisation. La phase initiale de PBC permet de représenter graphiquement l'ensemble de données d'apprentissage (segments de cercle ou barres rectangulaires) et d'initialiser un arbre de décision au nœud racine, correspondant à l'ensemble de données d'apprentissage.

De façon concrète, la représentation graphique d'un ensemble de données aboutira à la figure 3.9 par exemple.

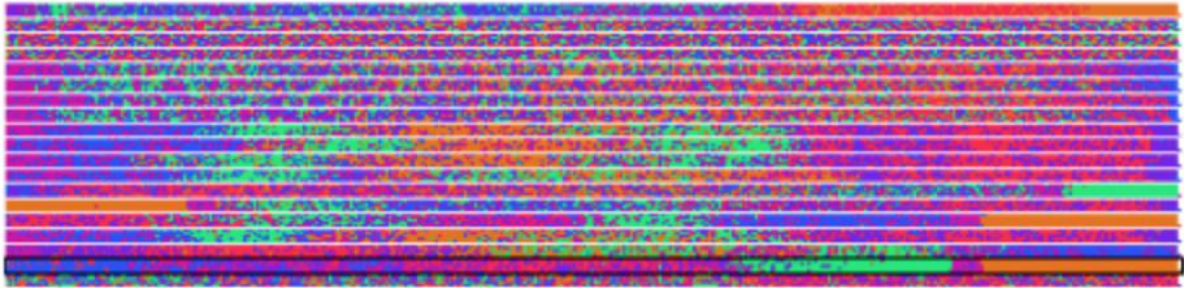


Figure 3.9 – Exemple de représentation d'un ensemble de données

La visualisation permet de sélectionner de façon interactive des données et de procéder à des coupes. A partir de la représentation de la figure 3.10, pour la construction d'un arbre de décision par exemple, on peut procéder à des coupes interactives : binaires ou n aires mono variées. L'idée ici est de concevoir de façon interactive un modèle des données. A cet effet, on utilise la stratégie suivante : recherche de la meilleure partition pure, s'il n'en existe pas : recherche de la plus grande partition dominante, s'il n'en existe pas : recherche de l'ensemble de partitions dominantes.

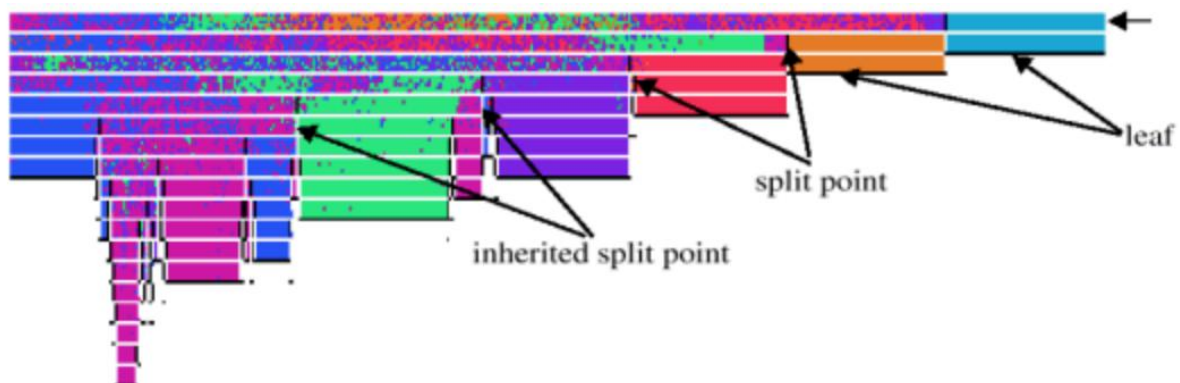


Figure 3.10 – Représentation du processus de coupes successives

L'algorithme de construction interactive d'arbres de décision (CIAD) [Poulet, 2002b] et le module UserClassifier [Ware et al, 2001] de WEKA utilisent le même principe de construction d'arbres de décision mais sont basés sur des représentations matricielles. Les sections détaillent ces deux approches.

3 Techniques matricielles et construction interactive d'un modèle des données

CIAD (Construction Interactive d'Arbre de Décision)

CIAD est un outil permettant la construction interactive d'arbres de décision. Cette technique utilise des matrices de scatter plot comme technique de visualisation et permet pour des ensembles de données avec un nombre de dimensions (n inférieur à 20) une projection de $n*(n-1)/2$ matrices. Pour $n > 20$, une représentation par défaut de l'ensemble de données est fournie avec une combinaison de 20 attributs au maximum. La première étape de traitement consiste à représenter graphiquement l'ensemble de données à traiter. La figure 3.11 représente une vue de l'ensemble de données segmentation de l'UCI [Blake et Merz, 1998] avec CIAD.

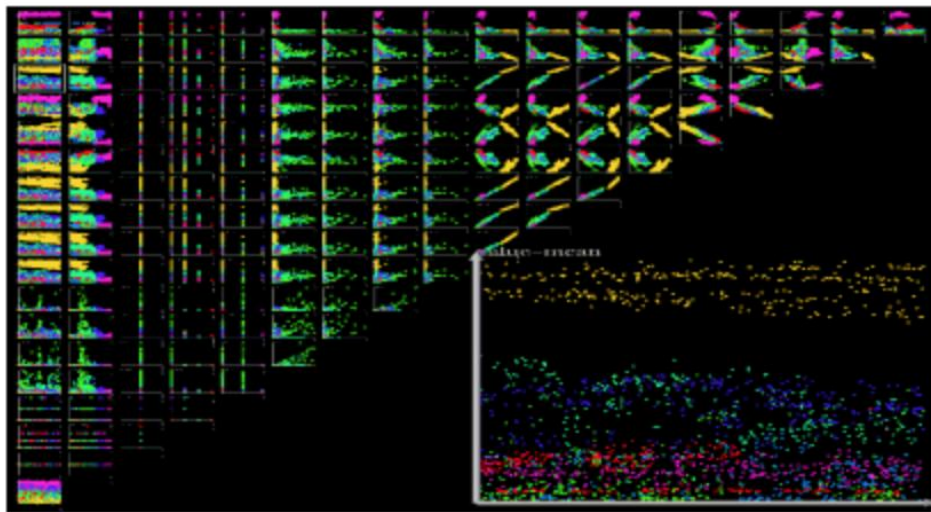


Figure 3.11 – Représentation de l'ensemble de données segment avec CIAD

Le couleur représente la classe. Les coupes effectuées pour la construction du modèle de données sont de type oblique en 2 dimensions donc sur deux variables. Ces différentes coupes sont effectuées grâce aux capacités humaines en reconnaissance de formes. Les étapes successives de ce traitement sont illustrées par la figure 3.12.

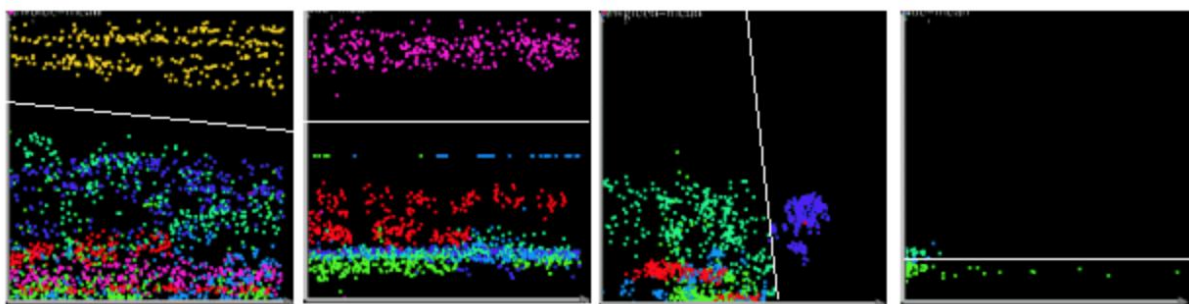


Figure 3.12 – Construction interactive du modèle de données avec CIAD

Les 4 premières coupes représentent les classes 2, 7, 6, 3. Ces classes représentent 57% des individus de l'ensemble de données. CIAD peut être exécuté en modes 100% interactif, mixte ou alors 100% automatique. Par rapport aux méthodes automatiques, CIAD permet d'obtenir une précision équivalente avec des tailles d'arbres inférieures.

UserClassifier de WEKA

Le module UserClassifier de WEKA est une implémentation de PBC qui utilise aussi des matrices 2D pour la construction interactive d'arbres de décision. UserClassifier à l'étape initiale de

présentation de données ne permet pas d'avoir une vue globale de l'ensemble de données à traiter. Une seule matrice 2D est présentée à l'écran. Pour le traitement de grands ensembles de données, la notion de contexte est perdue, il n'est pas possible de visualiser toutes les paires possibles d'attributs en même temps à l'écran. Moyennant des efforts, il est possible d'accéder à toutes les paires de combinaison possibles d'attributs une par une, ce qui n'est pas le cas avec CIAD qui ne peut aller au-delà d'un certain nombre de dimensions (limite due à l'utilisation de la représentation sous forme de matrice de matrices de scatter plot). Les coupes opérées avec le module UserClassifier sont rectangulaires, polygonales ou alors sous forme de polygones (voir la figure 3.13).

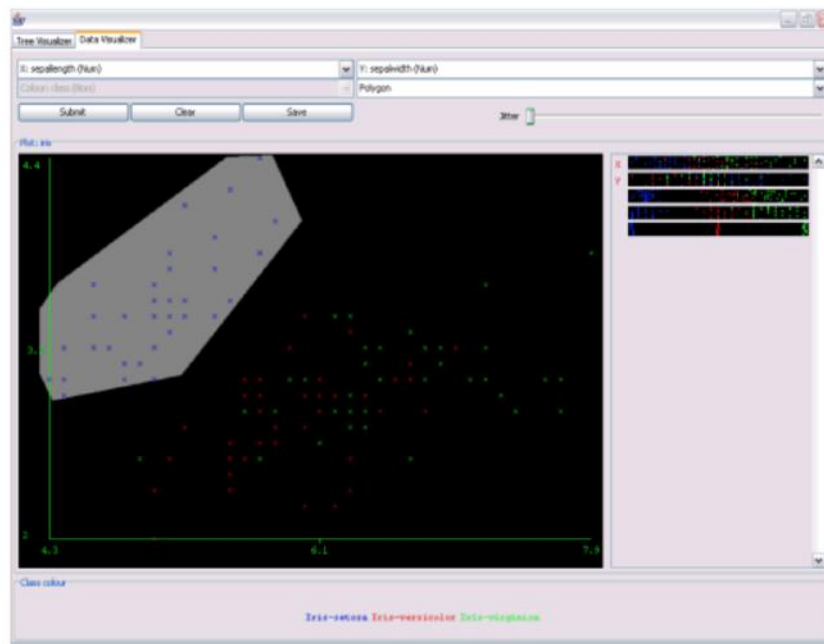


Figure 3.13 – Représentation et construction interactive du modèle de données avec UserClassifier de WEKA

UserClassifier hérite de tous les inconvénients de la représentation graphique sous forme de matrice de scatter plot, notamment, l'impossibilité de traiter des ensembles de données pourvus d'un nombre élevé d'individus. De plus, il n'existe pas de mécanisme d'aide aux utilisateurs durant la construction du modèle des données.

4

Analyse et conception

1 Modélisation logicielle

Je continue d'utiliser le pattern MVC pour le modèle de programme. Parce que les fonction base est déjà réalisé, en plus pour le diagramme plus claire et facile a comprendre. Donc ici, je juste montre le diagramme de les classe qui est en attent de mise en oeuvre : Mission, et publications. Le diagramme classe est comme ci-dessous (que les classe qui sont en attente de mise en oeuvre).

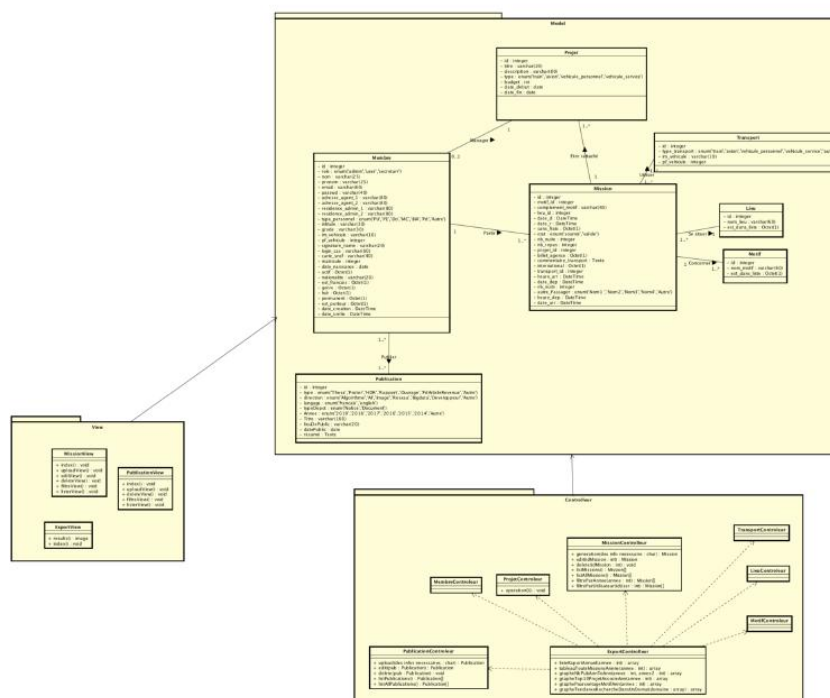


Figure 4.1 – Le diagramme de classe que les classes qui sont en attente de mise en oeuvre

1.1 Modele

La partie de modele dans le diagramme de classe est comme Figure 4.2.

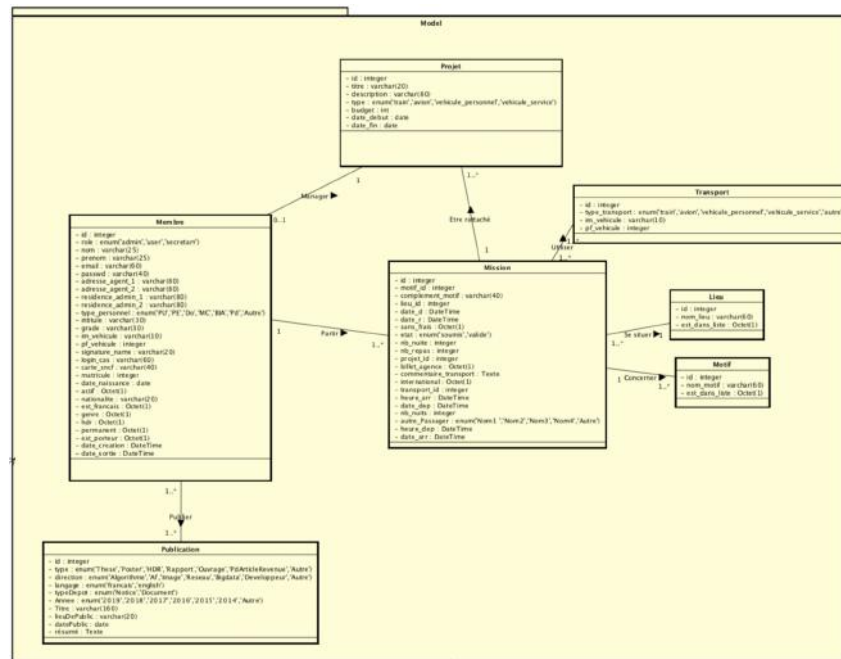


Figure 4.2 – La partie « Modele » du diagramme de classe

- Classe « Mission » : cette classe est modele de mission. Elle contient des attributs base de la mission : son identifiant et sa identifiant de motif, date début, date arrive et id de projet associé, et nombre de nuit etc.
- Classe « Transport » : cette classe est modele de transport. Elle contient d'identifiant de transport , le type de transport (véhicule, train, ou avion, ou autre), et l'immatriculation véhicule et la puissance fiscale véhicule.
- Classe « Motif » : cette classe est modele de motif. Elle contient d'identifiant de motif, le nom de motif (CONSEIL LI , ECOLE D'ETE, JURY DE THESE, REUNION D'EQUIPE' ou autre) et un indicateur pour indiquer si le motif est dans la liste. Si il n'est pas dans la liste, choisir " autre" et ajouter dans la rubrique "complément de motif".
- Classe « Lieu » : cette classe est modele de lieu. Elle contient d'identifiant de lieu, le nom de lieu et un indicateur pour indiquer si le lieu est dans la liste. Si il n'est pas dans la liste, choisir " autre" et ajouter dans la rubrique "Nouveau lieu".
- Classe «Publciatuion» :cette classe est modele de publciation. Elle contient des attributs base de la publication : son identifiant, type , direction, langage, annéeDepot, et titre etc.
- class «Membre» et «projet» est le modèle de membre et le modèle de projet, qui ont déjà réalisé. On ne repete plus.

1.2 Contrôleur

La partie de controleur dans le diagramme de classe est comme Figure 4.3.

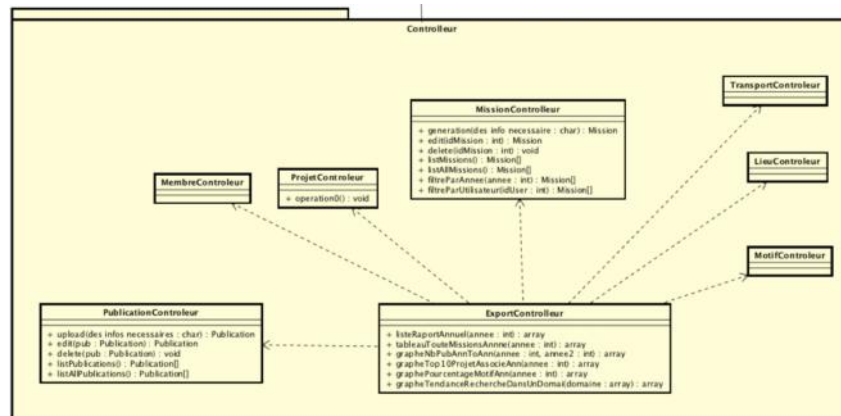


Figure 4.3 – La partie « Contrôleur » du diagramme de classe

Dans la partie « Contrôleur », il y a 6 classes : «MissionContrôleur», «ExportController», «PublicationController», «MotifController».

- Classe «MissionController» : Cette classe est pour exécuter et réaliser la gestion de mission : Ajouter, supprimer, modifier et visualiser la mission. De plus, filtrer les missions par l'année ou par utilisateur.
- Classe «ExportController» : Cette classe est pour réaliser la génération le rapport annuel ou des graphes, des tableaux .
- Classe «PublicationController» : Cette classe est pour réaliser la gestion de publication : Ajouter, supprimer, modifier et visualiser la publication.
- Classe «TransportController» : Cette classe est pour réaliser la gestion de transport : Ajouter, supprimer, modifier et visualiser le transport.
- Classe «LieuController» : Cette classe est pour réaliser la gestion de lieu : Ajouter, supprimer, modifier et visualiser le lieu.
- Classe «MotifController» : Cette classe est pour réaliser la gestion de motif : Ajouter, supprimer, modifier et visualiser le motif.

1.3 View

La partie de controleur dans le diagramme de classe est comme Figure 4.4

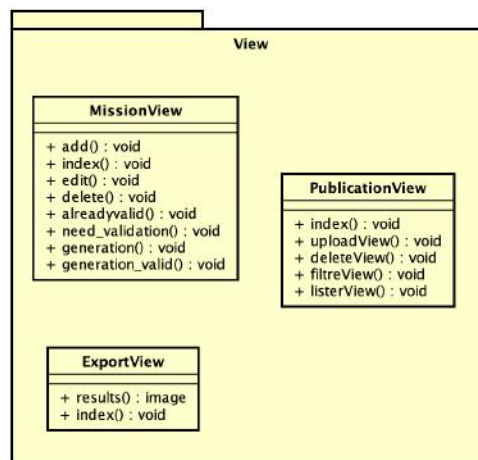


Figure 4.4 – La partie « View » du diagramme de classe

- Classe «MissionView» : Cette classe consiste à générer l'interface pour l'interaction avec l'utilisateur pour la mission.

- Classe «ExportView» : Cette classe consiste à générer l'interface pour l'interaction avec l'utilisateur pour visualiser les graphes.
- Classe «PublicationView» : Cette classe consiste à générer l'interface pour l'interaction avec l'utilisateur pour la publication.

5

Mise en oeuvre

Dans ce chapitre, nous allons vous montrer l'interface finale de mon système, et expliquer le processus spécifique et l'implémentation de chaque fonction.

1 Introduction de la création d'une mission

Il s'agit d'un processus complet de la création d'une mission à être validier. Regardez d'abord le diagramme temporel ci-dessous :

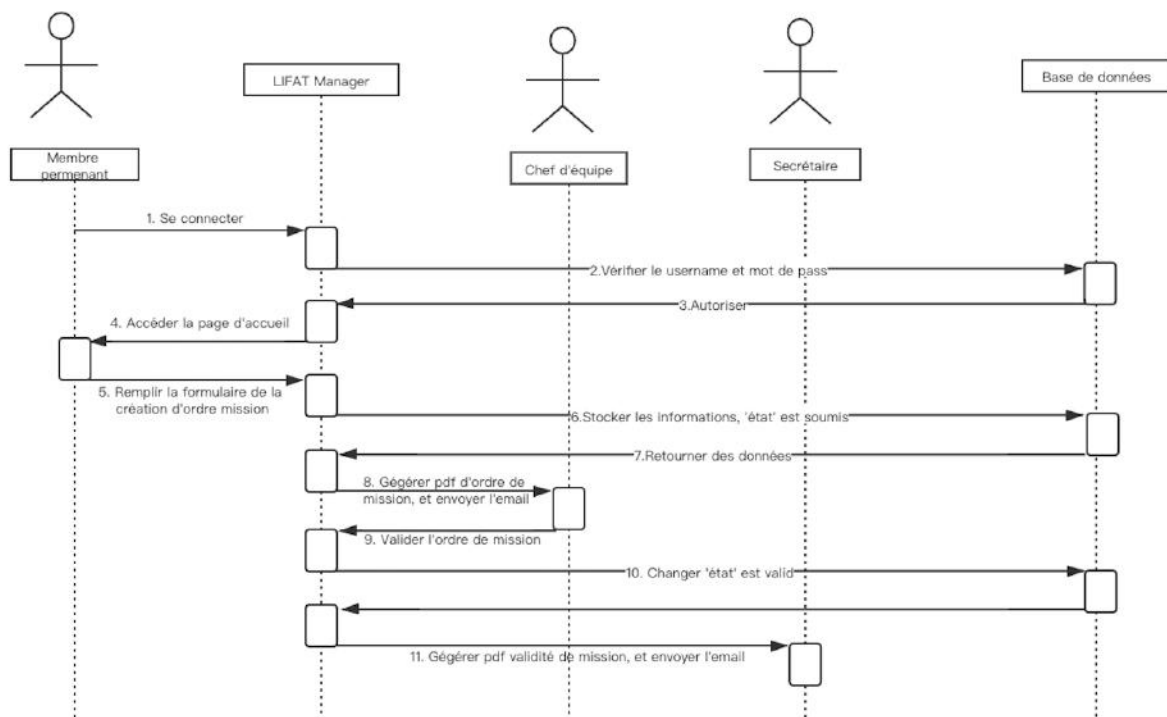


Figure 5.1 – *Le diagramme temporel*

Une fois qu'un membre permanent s'est connecté avec succès, il peut utiliser la fonction de création d'une mission.

Quand un **membre** crée une mission, d'abord le membre remplit le formulaire de la création d'ordre de mission, il doit remplir tous les champs obligatoires marqués d'un *, sinon il ne peut pas soumettre.

Le **système** stocke les informations dans la base de données et définit le champ «etat» sur "soumis". Le système utilise ces informations pour générer le pdf d'ordre de mission PDF1. Ce processus de génération de PDF sera décrit en détail dans la partie **IHM de mission**. Ce pdf contient toutes les informations relatives à la mission que l'utilisateur vient de remplir. Ensuite, le système enverra un e-mail au chef d'équipe, qui contient PDF1 et une brève introduction de l'ordre de mission.

Quand le **chef d'équipe** reçoit ce email, il a besoin de valider ce d'ordre de mission pour le rendre efficace. Le chef d'équipe se connecte au système, après passer à la page «Missions à valider», il affiche tous les missions en attendant d'être valider. Le chef d'équipe trouve la mission correspondante, et cliquer le bouton "valider".

Le **système** modifie le champ «etat» à "valid", et générer le pdf de la validité de mission PDF2. PDF2 contient les informations importantes de ce mission, et les signatures de chef d'équipe etc. (On va vous expliquer le détails dans la section partie **IHM de mission**). Ensuite, le système enverra un e-mail aux secretaire, qui contient PDF1 et PDF2 et une brève introduction de la mission.

Jusqu'à ici, la mission a été créé avec succès. Toutes les interfaces implémentées vous seront présentées dans la partie **IHM de mission**.

2 Bibliothèques utilisées

2.1 FPDF

Introduction

Lors de la génération de PDF, j'ai utilisé la classe FPDF. FPDF est une classe PHP qui permet de générer des fichiers PDF en PHP, F de FPDF signifie Free. Cette classe fournit les attributs et les méthodes nécessaires pour générer des PDF. Nous donnons ici des exemples de plusieurs méthodes couramment utilisées :

- **GetX** - renvoie la position x courante
- **GetY** - renvoie la position y courante
- **Header** - en-tête
- **Image** - imprime une image
- **SetDrawColor** - définit la couleur de tracé
- **SetFillColor** - définit la couleur de remplissage
- **SetFont** - fixe la police
- **Line** - trace une ligne
- **MultiCell** - imprime du texte avec saut de ligne
- **Output** - sauve ou envoie le document
- **Rect** - trace un rectangle
- **Ln** - saut de ligne

L'utilisation

1. La construction du modèle pdf En cours d'utilisation, nous devons hériter de la classe FPDF et appeler ces méthodes pour générer notre propre classe PDF. En d'autres termes, générez notre modèle PDF. La classe **pdfOrdre** est pour generer le pdf d'ordre de mission L'implémentation spécifique dans le code est comme suivant

```

1 class pdfOrder extends FPDF
2 {
3     function __check($x,$y,$width,$checked) {
4         if ($checked)
5             $this->rect($x,$y,$width,$width,'F');
6         else
7             $this->rect($x,$y,$width,$width);
8     }
9     function LI($sansFrais = false) {
10         $LI = 'LI';
11         $coord = 'Laboratoire informatique
12 Secrétariat Christelle Grange
13 64, avenue Jean Portalis
14 37200 Tours
15 Tél. : 02 47 36 14 42
16 Fax. : 02 47 36 14 22';
17         $this->Image(dirname(__FILE__) . '/li.jpg',10,2,20);
18         $this->SetFont('Times','B',20);
19         $this->SetFont('Times','',9);
20         $this->setXY(35,3);
21         $this->Multicell(80,4,utf8_decode($coord),0,'LT');
22     }

```

2. Transférer des données vers le modèle Ensuite, nous devons définir une classe Generator, qui est utilisée pour fournir une interface pour la fonction principale. En d'autres termes, il est utilisé pour transférer des données. Remplissez les données pour chaque champion dans le modèle. Ici, nous montrons un petit morceau de code.

```

1 class MyGenerator
2 {
3     /**
4      * Initialisation of agent part
5      */
6     public function setAgent($id, $name, $prenom, $adresseagent1,
7 $adresseagent2, $residenceadmin1, $residenceadmin2, $equipe,
8 $intitule, $grade, $personnel_type, $signature_path,
9 $chiefSignaturePath, $dateNaissance)
10    {
11        $this->_id = $id;
12        $this->_name = $name;
13        $this->_prenom = $prenom;
14        $this->_adresseagent1 = $adresseagent1;
15        $this->_adresseagent2 = $adresseagent2;
16        $this->_residenceadmin1 = $residenceadmin1;
17        $this->_residenceadmin2 = $residenceadmin2;
18        $this->_intitule = $intitule;
19        $this->_grade = $grade;
20        $this->_equipe = $equipe;
21        $this->_personnel_type = $personnel_type;
22        $this->_signature_path = $signature_path;
23        $this->_chiefSignaturePath = $chiefSignaturePath;

```



```

24         $this->_date_naissance = $dateNaissance;
25
26     }
27     /**
28      * return pdf
29      */
30     public function generate($sansFrais = false, $fileName = null)
31     {
32         // error_reporting(0);
33         ob_start();
34         $pdf = new \pdfOrder('P', 'nm', 'A4');
35         $pdf->SetAutoPageBreak(false);
36         $pdf->AddPage();
37         $pdf->LI($sansFrais);
38         $pdf->cadreAdmin($this->_matricule);
39         $pdf->agent($this->_id, $this->_name, $this->_prenom,
40             $this->_adresseagent1, $this->_adresseagent2,
41             $this->_residenceadmin1, $this->_residenceadmin2,
42             $this->_equipe, $this->_intitule, $this->_grade,
43             $this->_personnel_type, $this->_signature_path,
44             $this->_date_naissance);
45         $pdf->mission($this->_motifDeplacement, $this->_lieuM,
46             $this->_complementMotif, $this->_dateD, $this->_hDepartD,
47             $this->_heureArriveeD, $this->_dateR, $this->_hDepartR,
48             $this->_heureArriveeR, $this->_nbRepas, $this->_nbNuite);
49         $pdf->budget($this->_projet);
50         $pdf->transport($this->_transport);
51         $pdf->transportBis($this->_immatriculation, $this->_puis,
52             $this->_commentaire, $this->_reducSncf, $this->_billAgen);
53         $pdf->signatures($this->_cheifSignaturePath);
54         $pdf->justificatif();
55         if ($fileName == null) {
56             $pdf->Output("OdM.pdf", "I");
57         } else {
58             $pdf->Output($fileName, "F");
59         }
60         ob_end_flush();
61     }

```

3. Générer un pdf Enfin, définissez un objet MyGenerator dans le fichier MissionController.php et appelez la méthode generate() pour générer un pdf. Voici un petit morceau du code :

```

1     $generator = new \MyGenerator();
2
3     $generator->setMission(
4         array_column(array_column($this->Missions->find()
5             ->contain('Motifs', function (Query $q) {
6                 return $q->select(['motifs.nom_motif']);
7             })->where(['Missions.id' => $id])
8             ->all()->toList(), 'motifs'), 'nom_motif')[0],
9

```

```

10         array_column( $this->Missions->find()
11             ->select ([ 'complement_motif' ])
12             ->where ([ 'id' => $id])->all()->toArray(),
13             'complement_motif')[0],
14         $varr = array_column(array_column( $this->Missions
15             ->find()->contain( 'Lieux', function(Query $q)
16                 { return $q->select ([ 'lieus.nom_lieu' ]);} )
17             ->where ([ 'Missions.id' => $id])
18             ->all()->toList(), 'lieus'), 'nom_lieu')[0],
19         array_column( $this->Missions->find()
20             ->select ([ 'date_depart' ])->where( 'id' => $id])->all()
21             ->toArray(), 'date_depart')[0]->format( 'd/m/Y'),
22         array_column( $this->Missions->find()
23             ->select ([ 'date_depart' ])->where( 'id' => $id])->all()
24             ->toArray(), 'date_depart')[0]->format( 'H:i'),
25         array_column( $this->Missions->find()
26             ->select ([ 'date_retour' ])->where( 'id' => $id])->all()
27             ->toArray(), 'date_retour')[0]->format( 'd/m/Y'),
28         array_column( $this->Missions->find()
29             ->select ([ 'date_retour' ])->where( 'id' => $id])->all()
30             ->toArray(), 'date_retour')[0]->format( 'H:i'),
31         array_column( $this->Missions->find()
32             ->select ([ 'nb_repas' ])->where( 'id' => $id])->all()
33             ->toArray(), 'nb_repas')[0],
34         array_column( $this->Missions->find()
35             ->select ([ 'nb_nuités' ])->where( 'id' => $id])->all()
36             ->toArray(), 'nb_nuités')[0]
37     );
38     return $generator->generate($sansFrais, $fileName);

```

2.2 EmailComponent

Sous le framework cakephp3.7, il y a une classe **Email** sous **Cake\Mailer>Email** pour envoyer des emails. En cours d'utilisation, nous devons charger la classe au début du fichier :

```
use Cake\Mailer>Email;
```

Les emails sont livrés par transports. La configuration des transports permet de garder les données de configuration hors de votre code d'application et rend le déploiement plus simple car nous pouvons simplement changer les données de configuration. Un exemple de configuration de transport ressemble à ceci :

```

1
2 use Cake\Mailer\TransportFactory;
3
4 TransportFactory::setConfig( 'gmail', [
5     'host' => 'ssl://smtp.gmail.com',
6     'port' => 587,
7     'username' => 'my@exemple.com',
8     'password' => password,

```

```

9  'className' => 'Smtplib'
10 ]
11 );

```

Étant donné que les méthodes Setter d'Email retournent l'instance de la classe, nous devons définir ses propriétés avec le chaînage des méthodes. Les attributs suivants sont impliqués dans le code :

- **emailFormat** : format de l'email (html, texte ou les deux).
- **template** : si vous utilisez du contenu rendu, définissez le nom du modèle.
- **from** : e-mail ou tableau d'expéditeurs.
- **sender** : e-mail ou tableau de l'expéditeur réel.
- **to** : e-mail de destination.
- **cc** : e-mail ou tableau de copie carbone.
- **subject** : objet du message.
- **attachments** : liste des fichiers à joindre.
- **transport** : nom de configuration de transport.

Voici un morceau de code pour envoyer des e-mails comme suit

```

1  $email = new Email();
2  $email->set( 'nom', $nom_send );
3  $email->set( 'prenom', $prenom_send );
4  $email->set( 'lieu', $nomlieu );
5  $email->set( 'motif', $nommotif );
6  $email->set( 'date_depart', $date_d );
7  $email->set( 'date_retour', $date_r );
8  $email->set( 'commentaire', $commentaire_t );
9
10 return $email
11     ->template( 'submit_om' )
12     ->emailFormat( 'text' )
13     ->from( me@exemple.com )
14     ->to( $chefemail )
15     ->subject( 'Ordre de mission de ' . $prenom_send . ' à valider' )
16     ->cc( $email_send )
17     ->setTransport( 'gmail' )
18     ->attachments( [ 'ODM.pdf' => '/PRD-Projet-LIFAT/cake3_7
19                       /webroot/' . $this->Missions->id . '.pdf' ] )
20     ->send();

```

3 IHM de mission

3.1 Lister tous les missions

Dans la page index "mission", tous les missions (n'est pas en détail) d'un membre affichent sur le page. Il affiche des attributs importants de chaque mission. L'utilisateur pouvez cliquer le bouton "Nouveau mission", "Details", "Modifier", et "Supprimer" pour gérer la mission. L'interface est comme ci-dessous :

LIFAT Manager

Connecté en tant que : prenomUser2 nomUser2 (chef_equipe permanent)

[Mes Missions](#)
[Mission à Valider](#)
[Missions Validées](#)
[Administration](#)
[Export](#)
[Fichiers](#)
[Mon Profil](#)
[Deconnexion](#)

Rappel : L'ordre de mission doit parvenir au secrétariat 5 jours ouvrés avant le départ.
 Après validation par le chef d'équipe, l'OdM sera transmis au secrétariat par mail dont vous recevrez une copie.
 L'édition et la suppression ne sont possibles que pour un OdM soumis mais pas encore validé.

Recherche (Numéro et/ou État du mission) :

Filtrer

Mes missions [Nouveau mission]

Id	Motif	Lieu	Date Depart	Date Retour	Etat	Actions
382	motif10	Tours	04/04/2020 03:01	03/10/2020 03:03	soumis	Details Modifier Supprimer
567	motif10	Tours	04/06/2020 03:04	04/04/2022 03:03	soumis	Details Modifier Supprimer
568	motif10	Tours	04/06/2020 03:04	04/04/2022 03:03	soumis	Details Modifier Supprimer
569	motif10	Tours	04/06/2020 03:04	04/04/2022 03:03	soumis	Details Modifier Supprimer
565	motif2	Paris	03/04/2022 02:01	04/05/2023 04:02	soumis	Details Modifier Supprimer
570	motif4	Lille	06/03/2019 03:04	05/04/2021 03:01	soumis	Details Modifier Supprimer
571	motif3	Rennes	04/04/2019 02:02	02/03/2021 02:03	soumis	Details Modifier Supprimer
592	motif3	Tours	04/05/2020 00:03	04/04/2021 01:03	soumis	Details Modifier Supprimer
596	motif4	Orleans	03/01/2021 04:02	04/07/2021 03:02	soumis	Details Modifier Supprimer
424	motif4	Londres	06/05/2020 02:02	02/04/2021 04:03	soumis	Details Modifier Supprimer
566	motif4	Orleans	03/02/2019 01:01	04/03/2020 03:01	soumis	Details Modifier Supprimer

Figure 5.2 – l'interface des missions d'un membre

3.2 Créer une mission

Quand un membre créer une mission, le site va envoyer un email Email1 (cet email contient le document pdf d'ordre de mission - PDF1) vers chef d'équipe. Après le chef d'équipe valide ce PDF1 (mission), le site va encore envoyer un email Email2(contient le document pdf validité de mission - PDF2)vers cet membre respondant et secrétaire. Après le chef d'équipe valide le pdf1 et générer le pdf 2, la base de données est été ajouté une donnees.

L'utilisateur clique le bouton "Nouveau mission" pour créer une mission. L'interface de la création de l'ordre de mission :

Mission *
Projet associé
☐ Mission Sans Frais

Motif *
Motif
Complément du motif (Par exemple : Nom de la conférence, etc.)

Lieu *
Lieu
Nouveau lieu (si non présent parmi les propositions)

Dates *
Date et heure départ du départ
Date et heure du départ arrive
Date et heure départ du retour
Date et heure du retour arrive

Transport
☐ Train
☐ Avion
☐ Véhicule Personnel
☐ Véhicule de service
☐ Autre

Autres Passagers
Vous pouvez générer un ordre de mission identique pour d'autres utilisateurs (Utile lors d'un covoiturage, par exemple)

Admin
nomUser2
nomUser2

Aucun nouveau passager n'a été ajouté. Veuillez maintenir 'Ctrl' pour en ajouter un.

Informations complémentaires sur le transport
☒ Laisser le secrétariat commander les billets
Commentaires supplémentaires pour le secrétariat (horaires, etc.)

Si utilisation d'un véhicule
Immatriculation véhicule
Puissance fiscale véhicule
☐ Enregistrer le véhicule dans mon profil (En remplacement de l'ancien véhicule)

Nuitées et repas
Nombre de nuitées (Si le champ est laissé vide, le nombre de nuitées sera calculé automatiquement)
Nombre de repas (Si le champ est laissé vide, le nombre de repas sera calculé automatiquement)

Enregistrer la missions

Figure 5.3 – l'interface d'ajouter d'un mission

Dans le Email1 , il y a le pdf d'ordre de mission.



Figure 5.4 – l'interface d'email au chef d'équipe

Le pdf d'ordre de mission PDF1 est comme ci-dessous :

 Laboratoire informatique Secrétariat Christelle Grange 64, avenue Jean Portails 37200 Tours Tel : 02 47 36 14 42 Fax : 02 47 36 14 22		Cadre réservé à l'administration N° AGENT : 101010102 N° DEPLACEMENT : N° COMMANDE :	
ORDRE DE MISSION Laboratoire d'Informatique EA 6300			
AGENT NOM : nomUser2 PRENOM : prenomUser2 Adresse Personnelle : adresse2 Résidence administrative : residence2 Date de naissance : 07-06-1997		<input type="checkbox"/> Personnel Université <input type="checkbox"/> Titulaire/intitulé : intitulé2 <input type="checkbox"/> Contractuel/Grade : Chef d'équipe <input type="checkbox"/> Personnalité Extérieure <input checked="" type="checkbox"/> Doctorant Equipe Interne au LI : equipe1	
Je soussignée, prenomUser2 nomUser2 m'engage à souscrire une police d'assurance garantissant d'une manière illimitée ma responsabilité personnelle aux termes des articles 1382, 1383 et 1384 du Code Civil ainsi que, éventuellement, la responsabilité de l'Etat, dans les conditions définies à l'article 34 du décret num 90-437 du 28 mai 1990.			
Certifié exact par l'intéressé(e)		A Tours, le Signature	
MISSION Motif du déplacement : motif10 Lieu de la mission : Tours Nombre de repas : 364 Nombre de nuitées : 182		BUDGET D'IMPUTATION Projet : projet1	
DEPART Date : 04/04/2020 Heure de départ : 03:01		RETOUR Date : 03/10/2020 Heure d'arrivée : 03:03	
TRANSPORT Moyens de transport : Avion			
Transport : Commentaires, Réduction SNCF, Immatriculation			
Immatriculation du véhicule : Puissance fiscale du véhicule :		<input checked="" type="checkbox"/> Billet commandé par le secrétariat Réduction SNCF : carteSncf2	
Signature du responsable de l'équipe LI		Fait à Tours, le Signature du Directeur de l'EPU ou du Directeur du Laboratoire	
JUSTIFICATIF A JOINDRE : - Paiements des indemnités forfaitaires journalières : factures acquittées (hôtel et restaurants) - Prise en charge des frais de transport : billet (train-avion) - Autre dépenses (Autoroute - Taxi - Métro - RER - Bus - Parking) : Tickets, reçu, abonnements... - Si utilisation du véhicule de service (Carburant) - Attestation de non paiement pour les Agents de la Fonction Publique - Réçu du règlement des frais d'inscriptions payé directement par l'Agent - En cas de demande d'avance, joindre une lettre			

Figure 5.5 – l'interface de PDF ordre de mission

3.3 Modifier une mission

Quand un membre modifie une mission, c'est pareil comme la création du mission. L'utilisateur clique le bouton "Modifier", après sauté à la page ci-dessous :

Rappel : Vous devez remplir les options marquées qui avec *.

Mission *

Projet associé
projet1

☐ Mission Sans Frais

Motif *

Motif
motif10

Complément du motif (Par exemple : Nom de la conférence, etc.):

Lieu *

Lieu
Tours

Nouveau lieu (si non présent parmi les propositions)

Dates *

Date et heure du départ
2020 April 01

Date et heure arrivée du départ

Date et heure du retour
2020 October 09

Date et heure arrivée du retour

Transport

☐ Train

☒ Avion

☐ Véhicule Personnel

☐ Véhicule de service

☐ Autre

Autres Passagers

Vous pouvez générer un ordre de mission identique pour d'autres utilisateurs (Utile lors d'un covoiturage, par exemple)

Admin
nomUser2
nomUser2

Aucun nouveau passager n'a été ajouté. Veuillez maintenir 'Ctrl' pour en ajouter un.

Informations complémentaires sur le transport

☒ Laisser le secrétariat commander les billets

Commentaires supplémentaires pour le secrétariat (horaires, etc.):

Si utilisation d'un véhicule

Immatriculation véhicule

Puissance fiscale véhicule

☐ Enregistrer le véhicule dans mon profil (En remplacement de l'ancien véhicule)

Nuitées et repas

Nombre de nuitées (Si le champ est laissé vide, le nombre de nuitées sera calculé automatiquement)

182

Nombre de repas (Si le champ est laissé vide, le nombre de repas sera calculé automatiquement)

364

Enregistrer la missions

Figure 5.6 – l'interface de modification de mission

3.4 Supprimer une mission

Quand un membre supprime une mission, le site envoie un email (qui contient un texte indiquant que ce mission a été supprimée) à secrétaire et chef d'équipe. Cette données est supprimé dans la base de données.

L'utilisateur a juste besoin de cliquer ce bouton **“Supprimer”**.

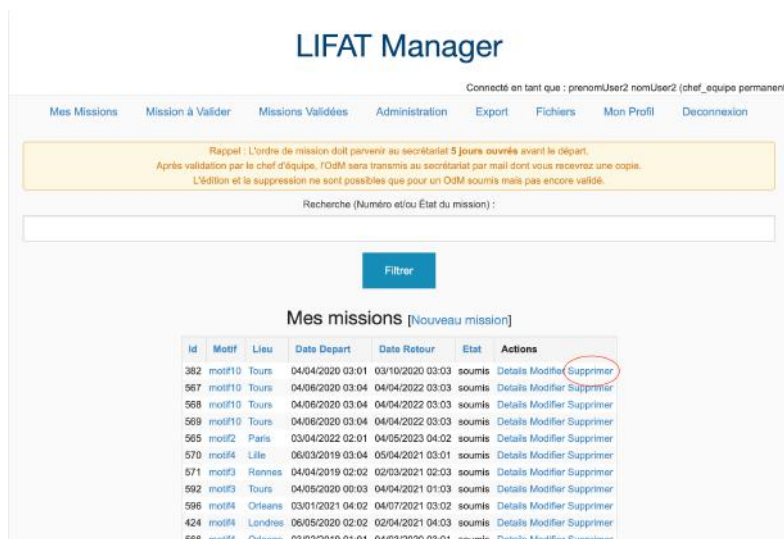


Figure 5.7 – l'interface de suppression de mission

3.5 Visualiser une mission

Quand un membre visualise une mission, le pdf d'ordre de mission(en détail) lors de la création le mission va afficher.

L'utilisateur clique le bouton **“Details”**, le pdf est pareil avec le **Figure 5.5**.

3.6 Chercher une mission

Quand un membre cherche une mission, l'utilisateur entre la mot-clé (le numéro de mission ou l'état de mission) dans le champ de recherche, après cliquez le bouton **“Filtre”**, tous les mission correspondant (n'est pas en détail) affichent sur le page.

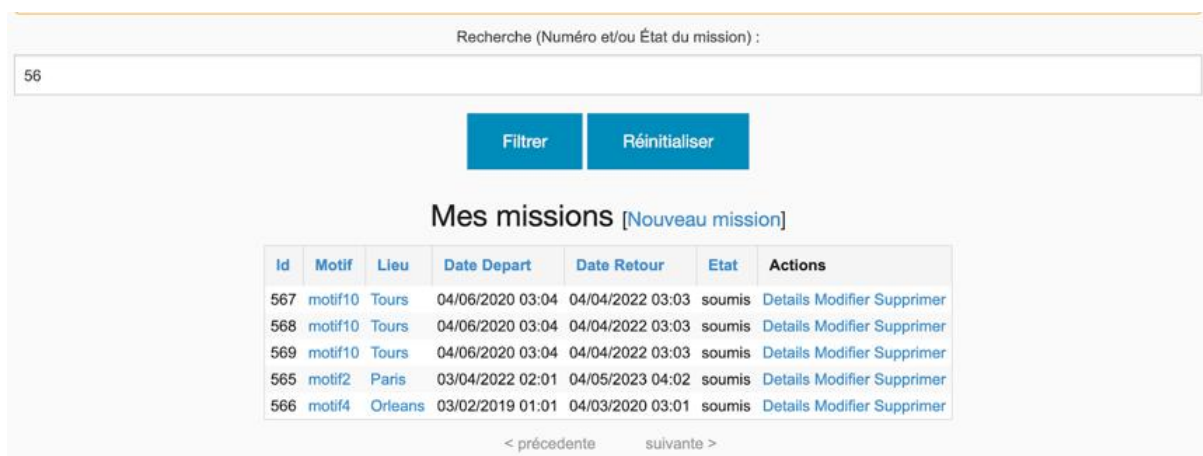


Figure 5.8 – l'interface de recherche d'une mission

3.7 Valider une mission

Quand un chef d'équipe valide une mission, le chef clique le bouton “ **Valider** ”. Ce mission va être validé.

N°	Nom	Prenom	Motif	Lieu	Date Départ	Date Retour	Etat	Actions
300	Admin	Admin			03/03/2018 03:04	03/04/2022 03:03	soumis	Edit Details Valider
301	Admin	Admin			03/03/2018 03:04	03/04/2022 03:03	soumis	Edit Details Valider
303	Admin	Admin	motif3	Bordeaux	03/03/2018 03:04	03/04/2022 03:03	soumis	Edit Details Valider
304	Admin	Admin			02/03/2016 02:01	04/04/2023 02:06	soumis	Edit Details Valider
305	Admin	Admin			02/03/2016 02:01	04/04/2023 02:06	soumis	Edit Details Valider
307	Admin	Admin			02/03/2016 02:01	04/04/2023 02:06	soumis	Edit Details Valider
309	Admin	Admin	motif4	Orleans	03/06/2017 04:04	05/03/2023 05:05	soumis	Edit Details Valider
311	Admin	Admin	motif4	Orleans	03/06/2017 04:04	05/03/2023 05:05	soumis	Edit Details Valider
312	Admin	Admin	motif4	Orleans	03/06/2017 04:04	05/03/2023 05:05	soumis	Edit Details Valider
314	Admin	Admin	motif5		06/03/2022 04:02	02/03/2023 03:02	soumis	Edit Details Valider
318	Admin	Admin	motif4	Orleans	03/04/2017 05:04	06/04/2022 02:03	soumis	Edit Details Valider
320	Admin	Admin	motif4		01/04/2016 03:03	05/08/2022 04:03	soumis	Edit Details Valider
322	Admin	Admin	motif5	Orleans	04/06/2017 05:05	04/04/2022 08:04	soumis	Edit Details Valider
328	Admin	Admin	motif7	Bordeaux	05/07/2017 08:04	03/07/2023 03:04	soumis	Edit Details Valider
331	Admin	Admin	motif7	Bordeaux	05/07/2017 08:04	03/07/2023 03:04	soumis	Edit Details Valider
334	Admin	Admin	motif7	Bordeaux	05/07/2017 08:04	03/07/2023 03:04	soumis	Edit Details Valider
337	Admin	Admin	motif7	Bordeaux	05/07/2017 08:04	03/07/2023 03:04	soumis	Edit Details Valider
345	Admin	Admin	motif3	Tours	04/06/2019 03:05	04/04/2023 01:05	soumis	Edit Details Valider
348	Admin	Admin	motif4	Orleans	03/03/2017 01:03	01/02/2023 03:02	soumis	Edit Details Valider

Figure 5.9 – l'interface de validité d'une mission

Après, le système envoyer un email aux secrétaires. Dans ce email il convient 2 pdfs : le pdf d'ordre de mission PDF1 et le pdf de validité de mission PDF2. L'email est comme ci-dessous :

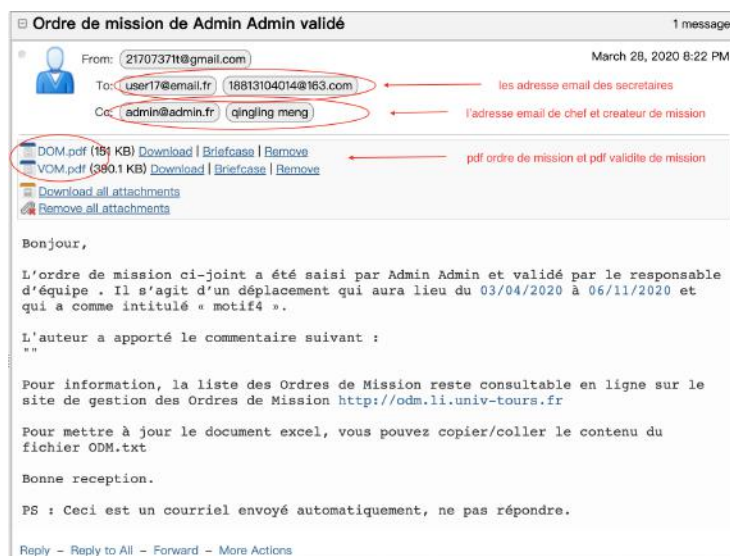




Figure 5.10 – l'interface de email aux secrétaires

Le pdf d'ordre de mission PDF1 est pareil avec le figure 4.

Le pdf de validité de mission PDF2 est comme ci-dessous :

AUTORISATION DE DEPLACEMENT
(à remplir pour chaque mission/déplacement - cf. procédure sur l'Intranet)
Document créé le 28/03/2020 à Tours

Identification de l'agent

NOM : Admin
PRENOM : Admin
Adresse mail : admin@admin.fr
Date de naissance : 11-02-2019
Unité de recherche / Composante / Service : equipe1

Motif(s) du déplacement

☐ Pédagogie ☒ Recherche ☐ Autre(précisez) :





Motif(s) : motif4
Composante, service ou unité de recherche demandant la mission :

Modalités de ALLER :

Date et heure de départ: 03/04/2020 01:02 - 03/05/2020 02:01
Trajet(s): LIFAT Paris - Orleans
Pays: France - France

Modalités de RETOUR :

Date et heure de retour : 06/11/2020 03:01 - 06/04/2022 05:02
Trajet(s): Orleans - LIFAT Paris
Pays: France - France

Pour les missions en France métropolitaine		En supplément pour les missions en France hors métropole et à l'étranger**	
<input checked="" type="checkbox"/> Utilisation du véhicule personnel* <input type="checkbox"/> Utilisation d'un véhicule de service <input type="checkbox"/> Permis de conduire en cours de validité <input checked="" type="checkbox"/> Recours au marché(train et/ou avion)		<input checked="" type="checkbox"/> Visite Médecine de prévention <input type="checkbox"/> Inscription sur le site ARIANE du MAE	
Signature de l'agent	Signature du directeur d'unité, du directeur de composante ou du chef de service	Visa du directeur des affaires juridiques	Signature du Président de l'université
			

* Joindre une copie de la carte grise ; l'utilisation du véhicule personnel doit être exceptionnelle et reste soumise à l'autorisation préalable de l'ordonnateur (horaires non adaptés, absence de transports en commun, correspondances trop nombreuses, etc ?)
** A transmettre à votre gestionnaire ou antenne financière 15 jours minimum avant le départ en mission.

Figure 5.11 – l'interface de PDF validité de mission PDF2

6

Bilan et conclusion

1 Bilan du semestre 9

Les taches finies Pendant la S9

Les taches suivantes sont deja finies :

- Lire et comprendre des documents existants liés au projet
- Analyser des besoins de client
- Ecrire le bilan de existant(Résumer des fonctionnalités déjà implementé et des fonctionnalités non implementées du projet)
- Etudier le framework cake3.7
- Etudier la fouille visuelle de données et ecrire l'état de l'art
- Ecrire le cahier de spécification
- Ecrire le rapport et Préparer soutenance mi-parcours

La tache en retard

Dans l'ensemble, ce projet se passe tres bien pendant le semestre 9. Toutes les taches sont finies en date prevue selon le diagramme de Gantt. Néanmoins, C'est la première fois que je rencontrais de Visuelle fouille donnees, donc dans le processus d'apprentissage, il y a quelques écarts. Heureusement, l'instructeur a corrigé ma direction à temps pour pouvoir terminer la tâche à la date prévue.

Les taches a faire :

- J'ai besoin d'apprendre le framework php utilisé par le projet ODM afin d'extraire du code utile du framework et de le mettre dans le projet LIFAT Manager.
- Je dois continuer à étudier le framework cake3.7 pour jeter les bases solides des travaux d'édition et de développement du semestre prochain.
- Le module de visualisation doit encore être optimisé. Des graphiques plus visuels doivent être fournis afin que les utilisateurs puissent obtenir des informations plus utiles et cachées.

2 Planning du semestre 10

Dans le diagramme de Gantt, on a déjà prévu les tâches et leurs durées pour le semestre 10. Ces tâches suivantes sont à faire pendant S10 :

- Programmer
- Déployer à la serveur d'Université Polytech
- Faire le test et correction
- Ecrire le rapport final
- Préparer de la soutenance

Les descriptions détaillées des taches sont dans l'annexes « Découpage des tâches ».

3 Bilan du semestre 10

Les tâches finies Pendant la 10

- Générer le pdf d'ordre de mission PDF1.
- Réaliser la création de la mission.
- Envoyer un email contenant PDF 1 au chef d'équipe.
- Améliorer le PDF1(convient le signature, le transport etc.)
- Générer le pdf de la validité de mission PDF2.
- Valider l'ordre de mission.
- Envoyer un email contenant PDF1 et PDF2 au secrétaire.
- Modifier une mission avant d'être validé.
- Rechercher une mission par le numéro ou état de la mission.
- Supprimer une mission.
- Ajouter une module de visualisation des statistique : Générer la liste des missions par année.
- Faire le test et correction
- Rédiger le rapport final
- Préparer le slide et vidéo pour présenter le résultat.

La tâche en retard

En combinant le temps, l'importance de la mission, la difficulté de la mission et d'autres facteurs, ainsi que l'impact de l'épidémie, j'ai finalement terminé la première partie, qui est également la partie la plus importante de ce PRD, la gestion de la mission, et généré Un tableau statistique.

Mais je garantis la qualité de la gestion de mission, et après mon déploiement sur le serveur, si l'utilisateur trouve des problèmes, je continuerai à effectuer des réparations.

Les tâches a faire

En raison de l'impact de l'épidémie, je n'ai pas été autorisé à retourner à l'école fin mars, je n'ai donc pas pu déployer le système avec succès sur le serveur de l'école pour le moment. Mais une fois l'épidémie terminée et l'école commencera à fonctionner normalement, je retournerai à l'école pour terminer cette partie de la tâche. Déployez le système sur le serveur de l'école.

4 Bilan sur la qualité

- La qualité de code : C'est la deuxième fois que je travaille sur des projets liés à PHP. Je pense avoir une certaine expérience. Je rend le code aussi simple que possible sur la base de

l'implémentation de fonctions. En même temps, les commentaires sont détaillés et suffisants. Ça rend le code plus lisible.

- La qualité de rapport : Le rapport va être contrôlé par l'encadrant.
- La qualité de programme : Je pense que la qualité du programme est bonne. Parce que j'ai effectué de nombreux tests, et en stricte conformité avec les exigences de l'encadrant, réalisé toutes les fonctions liées à la gestion de mission. Différents types d'utilisateurs ont été testés. En même temps, j'ai bien préparé les documents nécessaires tels que le cahier de spécification, le cahier de développeur, le cahier de test, le document d'utilisation. Pour garantir l'exactitude de l'interface et des fonctions.

Donc la qualité des travaux est assurée.

5 Bilan auto-critique

Je pense que ce PRD est très important pour moi. C'est la première fois que je suis seul en charge de l'ensemble du projet. Dès le début, analysez les existants, analysez les besoins des clients et rédigez un cahier de spécifications. Puis au code. Puis test et modification et amélioration continues. Bien que pour l'instant, je n'ai pas pu accomplir toutes les tâches, mais si le temps est suffisant, je dois m'efforcer de terminer la fonction. Rendez le système plus parfait.

J'ai mis beaucoup d'énergie et de temps à mon PRD, je le fais tous les mercredis, jeudis et week-ends. Bien que dans le processus de mise en œuvre, j'ai rencontré de nombreux problèmes et la tâche a été très dure, mais j'ai tout de même résolu tout. J'espère que je pourrai faire mieux que possible, mieux développer le système et le rendre plus stable. J'espère qu'il pourra être utilisé à l'avenir. Une fois l'épidémie terminée, je vais définitivement retourner à l'école et déployer le système sur le serveur afin que le système puisse être utilisé par plus de personnes.

Enfin pendant cette année, je pense que mon PRD se passe bien. Merci beaucoup pour les conseils de mon mentor tout au long du processus, guidez-moi pour préparer les documents nécessaires, mais aussi pour assurer la qualité des documents.

Annexes

A

Planification

1 Aperçu de gestion de projet

Le diagramme de Gantt Initial pour la planification de ce projet est comme Figure A.1.

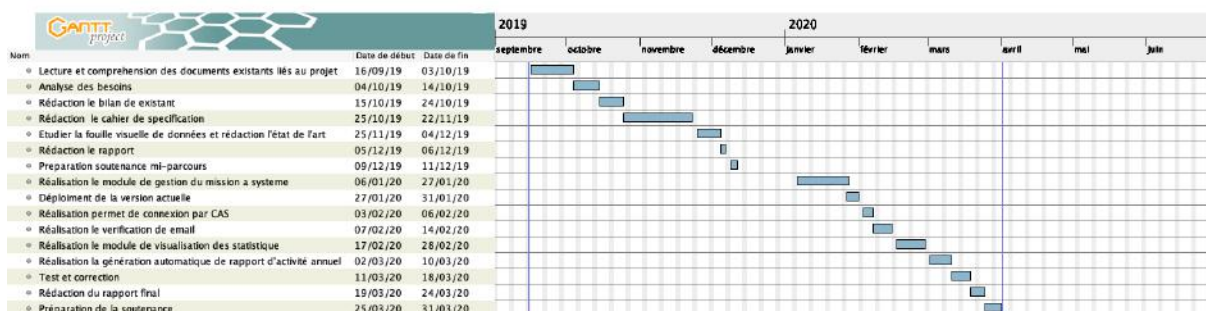


Figure A.1 – Le diagramme de Gantt Initial

Le diagramme de Gantt Final de ce projet est comme Figure A.2.

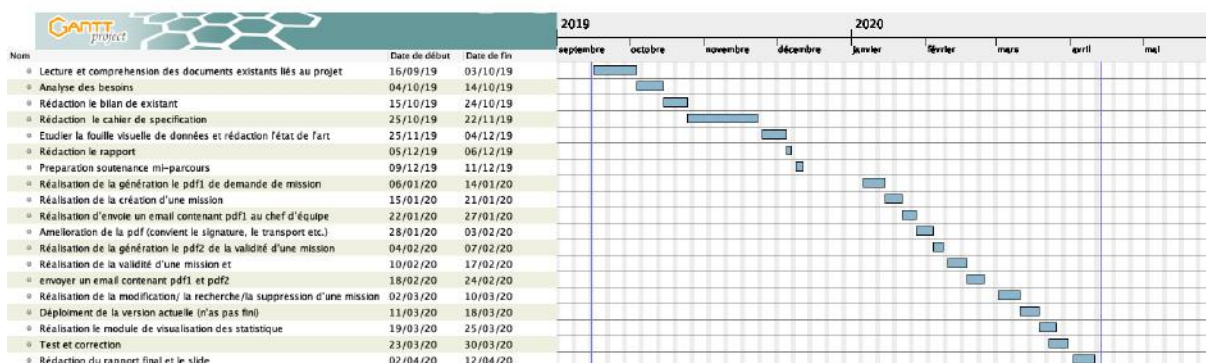
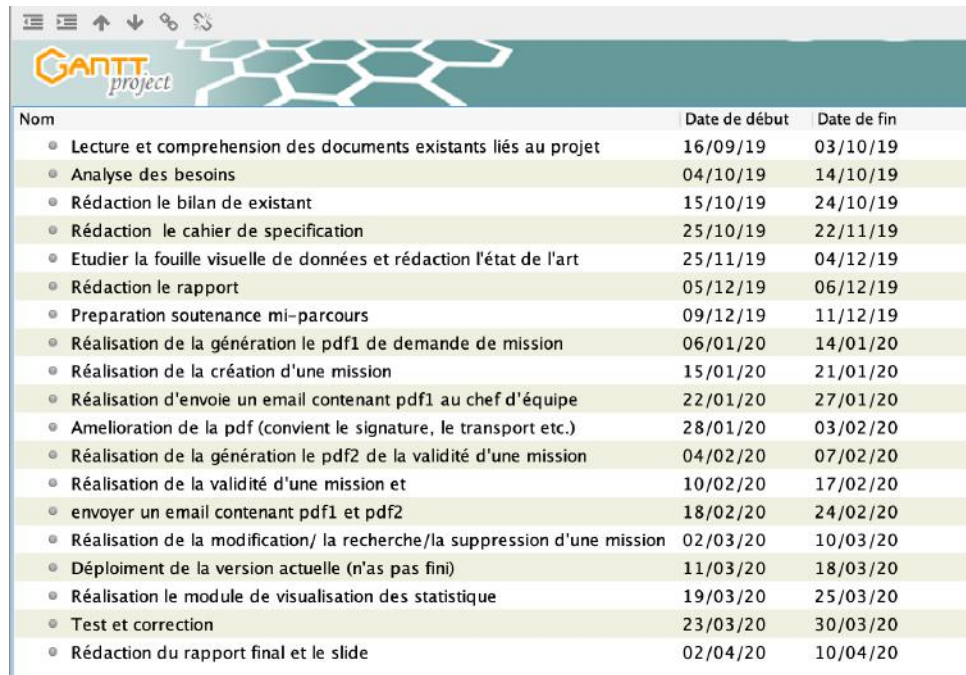


Figure A.2 – Le diagramme de Gantt Final

2 Découpage du projet en tâches

Ce qui suit est une introduction au diagramme Gantt final, la Figure A.3 est pour plus clairement afficher les tâches.



Nom	Date de début	Date de fin
• Lecture et comprehension des documents existants liés au projet	16/09/19	03/10/19
• Analyse des besoins	04/10/19	14/10/19
• Rédaction le bilan de existant	15/10/19	24/10/19
• Rédaction le cahier de specification	25/10/19	22/11/19
• Etudier la fouille visuelle de données et rédaction l'état de l'art	25/11/19	04/12/19
• Rédaction le rapport	05/12/19	06/12/19
• Preparation soutenance mi-parcours	09/12/19	11/12/19
• Réalisation de la génération le pdf1 de demande de mission	06/01/20	14/01/20
• Réalisation de la création d'une mission	15/01/20	21/01/20
• Réalisation d'envoi un email contenant pdf1 au chef d'équipe	22/01/20	27/01/20
• Amélioration de la pdf (convient le signature, le transport etc.)	28/01/20	03/02/20
• Réalisation de la génération le pdf2 de la validité d'une mission	04/02/20	07/02/20
• Réalisation de la validité d'une mission et	10/02/20	17/02/20
• envoyer un email contenant pdf1 et pdf2	18/02/20	24/02/20
• Réalisation de la modification/ la recherche/la suppression d'une mission	02/03/20	10/03/20
• Déploiement de la version actuelle (n'as pas fini)	11/03/20	18/03/20
• Réalisation le module de visualisation des statistique	19/03/20	25/03/20
• Test et correction	23/03/20	30/03/20
• Rédaction du rapport final et le slide	02/04/20	10/04/20

Figure A.3 – Les taches dans le diagramme de Gantt

Tâche 1 : Lecture et comprehension des documents existants liés au projet

- Date de début : 16/09/2019
- Date de fin : 03/10/2019
- Duree :17 jours
- Description : Mon sujet est de continuer à améliorer un système existant. Cette tache consiste a rassembler toutes les informations pertinentes sur le système qui continueront à être améliorées : documentation et code. Familiarisez-vous avec un projet existant, connaissez sa structure, quelles technologies sont utilisées et quelles fonctions ont été implémentées.

Tâche 2 : Analyse des besoins

- Date de début : 04/10/2019
- Date de fin : 14/10/2019
- Duree :11 jours
- Description : Cette tâche consiste à communiquer avec mon MOA monsieur Jean-Yves Ramel. Après une analyse approfondie et détaillée, comprenez avec précision les exigences spécifiques des utilisateurs et des projets en termes de fonctions, performances, fiabilité, etc., transformez l'expression des exigences non formelles de l'utilisateur en une définition complète des exigences et déterminez les fonctions que le système doit implémenter.

Tâche 3 : Rédaction le bilan de existant

- Date de début : 15/10/2019
- Date de fin : 24/10/2019
- Duree :10 jours
- Description : Cette tâche consiste à la rédaction le bilan de l'existant. Celle-ci permet de Résumer les fonctions qui ont été implémentées dans le système et tous les documents liés au système, y compris les documents de configuration de l'environnement, la structure de projet, etc.

Tâche 4 : Rédaction le cahier de specification

- Date de début : 25/10/2019
- Date de fin : 22/11/2019
- Duree :29 jours
- Description : Cette tache consiste à la redaction du cahier de specifications. Celle-ci permet de definir le contexte, les objectifs et leurs fonctionnalites.

Tâche 5 : Etudier la fouille visuelle de données et rédaction l'état de l'art

- Date de début : 25/11/2019
- Date de fin : 04/12/2019
- Duree :10 jours
- Description : D'abord, apprendre les technology de la fouille visuelle de données, après écrire l'état de l'art qui contient les techniques existants et courants.

Tâche 6 : Rédaction le rapport

- Date de début : 05/12/2019
- Date de fin : 08/12/2019
- Duree : 4 jours
- Description : Cette tâche consiste à rediger le rapport de S9 en integrant tous les documents écrit pendant S9.

Tâche 7 : Preparation soutenance mi-parcours

- Date de début : 09/12/2019
- Date de fin : 11/12/2019
- Duree : 2 jours
- Description : Cette tâche consiste à la préparation le soutenance mi-parcours. Il comporte de prerarer le slide et trier les idées.

Tâche 8 : Réalisation de la génération le pdf d'ordre de mission PDF1.

- Date de début : 06/01/2020
- Date de fin : 14/01/2020
- Duree : 9 jours
- Description : Cette tâche consiste à générer le pdf d'ordre de mission. Quand l'utilisateur a créé une mission, le système génère un pdf qui contient tous les informations relative à cette mission.

Tâche 9 : Réalisation de la création de la mission.

- Date de début : 15/01/2020
- Date de fin : 21/01/2020
- Duree : 7 jours
- Description : Cette tâche consiste à permettre l'utilisateur créer une mission. En d'autres termes, l'utilisateur remplit un formulaire qui contient toutes les informations relatives à la mission, puis crée la mission.

Tâche 10 : Réalisation d'envoi un email contenant PDF 1 au chef d'équipe.

- Date de début : 22/01/2020
- Date de fin : 27/01/2020
- Duree : 6 jours
- Description : Cette tâche consiste à envoyer un email au chef d'équipe. Quand un créateur de mission a rempli le tableau de la création et soumettre. Le système va envoyer un email au chef d'équipe.

Tâche 11 : Améliorer le PDF1(contient la signature, le transport etc.)

- Date de début : 28/01/2020
- Date de fin : 03/02/2020
- Duree : 7 jours
- Description : Cette tâche consiste à améliorer le PDF1, car quand il existe des champs spéciale qui à besoin de la traitement spéciale. Y compris le signature, et les informations de transport.

Tâche 12 : Réalisation de la génération le pdf de la validité de mission PDF2.

- Date de début : 04/02/2020
- Date de fin : 07/02/2020
- Duree : 4 jours
- Description : Cette tâche consiste à générer le pdf de la validite de mission PDF2. Quand le chef d'équipe a validé une mission, il va générer une pdf PDF2 qui contient les informations nécessaires sur ce mission et les signatures nécessaires.

Tâche 13 : Réalisation de la validité l'ordre de mission.

- Date de début : 10/02/2020
- Date de fin : 17/02/2020
- Duree : 8 jours
- Description : Cette tâche consiste à valider une mission. Ca permet le chef d'équipe valide une mission.

Tâche 14 : Envoyer un email contenant PDF1 et PDF2 au secrétaire.

- Date de début : 18/02/2020
- Date de fin : 24/02/2020
- Duree : 7 jours
- Description : Cette tâche consiste à envoyer un email au secrétaire. Quand le chef d'équipe valide une mission. Le système va envoyer un email au secrétaire. En même temps, le créateur de mission peut recevoir un pareil email aussi.

Tâche 15 : Réalisation de la modification/ la recherche/ la suppression une mission

- Date de début : 02/03/2020
- Date de fin : 10/03/2020
- Duree : 9 jours
- Description : Cette tâche consiste à réaliser de modifier une mission avant d'être validé. rechercher une mission par le numéro de la mission ou état de la mission. Et supprimer une mission.

Tâche 16 : Déployer la version actuel au serveur

- Date de début : 11/03/2020
- Date de fin : 18/03/2020
- Duree : 8 jours
- Description : Cette tâche consiste à déployer le système au serveur de l'école pour permettre au système d'être accessible à distance par tous les utilisateurs. Ceci est une partie très importante.

Tâche 17 : Ajouter une module de visualisation des statistique

- Date de début : 19/03/2020
- Date de fin : 25/03/2020
- Duree : 7 jours
- Description : Cette tâche consiste à ajouter une module de visualisation des statistique. Les données peuvent être affichées plus clairement et intuitivement.

Tâche 18 : Tests et correction

- Date de début : 23/03/2020
- Date de fin : 30/03/2020
- Duree : 8 jours
- Description : Cette tâche consiste à effectuer de nombreux tests pour vous assurer que les fonctions sont implémentées. Dans le même temps, la stabilité du système et la sécurité des données sont garanties.

Tâche 19 : Rédaction du rapport final et le slide

- Date de début : 02/04/2020
- Date de fin : 10/04/2020
- Duree : 9 jours
- Description : Cette tâche consiste à rédiger le rapport final, et ajouter des pièces manquantes dans S9. et préparer la soutenance final. Il comporte le slide et les résultats réalisés.

B

Description des interfaces externes du logiciel

1 Interfaces matériel/logiciel

Je continue de développer un projet existant, je donc utilise les même technologies avec le projet existant :

- Debian 9.8
- Apache 2.4.25
- CakePHP 3.7
- PHP 7.3.1

2 Interfaces homme/machine

- Un onglet «MesMissions» à ce Tool

LIFAT Manager

Connecté en tant que : prenomUser2 nomUser2 (chef_equipe permanent)

Mes Missions Mission à Valider Missions Validées Administration Export Fichiers Mon Profil Deconnexion

Rappel : L'ordre de mission doit parvenir au secrétariat 5 jours ouvrés avant le départ.
Après validation par le chef d'équipe, l'OdM sera transmis au secrétariat par mail dont vous recevrez une copie.
L'édition et la suppression ne sont possibles que pour un OdM soumis mais pas encore validé.

Recherche (Numéro et/ou État du mission) :

Filtrer

Mes missions [Nouvelle mission]

Id	Motif	Lieu	Date Depart	Date Retour	Etat	Actions
382	motif10	Tours	04/04/2020 03:01	03/10/2020 03:03	soumis	Détails Modifier Supprimer
567	motif10	Tours	04/06/2020 03:04	04/04/2022 03:03	soumis	Détails Modifier Supprimer
568	motif10	Tours	04/06/2020 03:04	04/04/2022 03:03	soumis	Détails Modifier Supprimer
569	motif10	Tours	04/06/2020 03:04	04/04/2022 03:03	soumis	Détails Modifier Supprimer
565	motif2	Paris	03/04/2022 02:01	04/05/2023 04:02	soumis	Détails Modifier Supprimer
570	motif4	Lille	06/03/2019 03:04	05/04/2021 03:01	soumis	Détails Modifier Supprimer
571	motif3	Rennes	04/04/2019 02:02	02/03/2021 02:03	soumis	Détails Modifier Supprimer
592	motif3	Tours	04/05/2020 00:03	04/04/2021 01:03	soumis	Détails Modifier Supprimer
596	motif4	Orleans	03/01/2021 04:02	04/07/2021 03:02	soumis	Détails Modifier Supprimer
424	motif4	Londres	06/05/2020 02:02	02/04/2021 04:03	soumis	Détails Modifier Supprimer
566	motif4	Orleans	03/02/2019 01:01	04/03/2020 03:01	soumis	Détails Modifier Supprimer

Figure B.1 – L'interface de la gestion de «MesMissions»

- A la interface connexion, ajouter une partie «Connexion par ENT/CAS»

Cette interface est en gardant l'esprit de l'interface ODM existante.

Connexion par ENT/CAS

Si vous possédez un identifiant ENT/CAS, vous pouvez vous connecter avec celui-ci en [cliquant ici](#).

Pour pouvoir vous connecter : votre identifiant doit avoir été associé avec votre compte de gestion des ordre de missions (Un message vous avertira, le cas échant).

Figure B.2 – L'interface de ENT/CAS

- Une option dans l'onglet «Rapport d'activité annuel» (Continue)

Cette interface est comme le figure ci-dessous. Dans le rapport annuel en terme de membre lister des projets réalisés de chaque membre cette année les thèses de recherche et les fichiers téléchargés.

Nom:	Membre1Nom	Prenom:	Prenom1	Equipe:	Equipe2	Type personnel:	PU/PE/Do	LieuTravail:	Paris
Projet réalisé	Titre			Type	Budget	Financement	Date debut	Date fin	
Mission	N°	Motif		Lieu	Date debut	Date fin	Etat	Transport	
Ficheir	Nom			Titre				Date Upload	
These	Sujet			Type		Financemnt	Date debut	Date fin	

Figure B.3 – L'interface du rapport annuel

- Un onglet «Publications» (Continue)

Cette interface est en gardant l'esprit de l'interface HAL existante. Regardez le figure B.4.

[illegible]

Figure B.4 – *L’interface de la gestion «Publication»*

3 Interfaces logiciel/logiciel

- Créer la base de donnée et le mapping avec les tables dans le base de données par le framework ORM. Il nous permet d'accéder et opérer les données des tables.

- Utilisez la classe Email de la framework cakephp3.7 pour envoyer un lien de vérification à l'adress email afin d'implémenter la fonction de vérification email.

C

Specifications fonctionnelles

1 Définition de la fonction 1 : Gestion de demand missions

Présenter la fonction 1 :

- Nom de la fonction : Gestion de demand missions
- Pour tous les membre de laboratoire, cette fonction permet d'ajouter / modifier / supprimer une mission, visualiser le pdf de la mission et tous les missions ce que vous soumettez. Pour les administrateur ou le chef de l'équipe ou le secrétaire, cette fonction permet aussi de filtrer les missions par année, et par utilisateur.
- Primordiale

Description de la fonction 1 :

- Tous les membres du laboratoire "LIFAT" ont chaque jour de nombreux programmes de mission. Tels que l'enseignement, la participation à des réunions d'équipe, des tâches de recherche, des visites de recherche. La fonctionnalité de gestion des tâches permet aux utilisateurs de soumettre leurs tâches à venir au système pour une gestion unifiée.
- Dans cette fonction, l'utilisateur entre dans le système des informations sur sa tâche, telles que "motif", projets associés, dates de tâche, etc. Cette fonctionnalité stocke les informations de tâches personnelles de l'utilisateur dans une base de données et permet à l'utilisateur de les visualiser ou de les modifier.
- Avant gérer des missions, l'utilisateur a besoin de se connecter à ce système. Cette fonction interagit à la fonction «login» et le composant «mission».
- Exception : Si l'utilisateur n'as pas entre les informations nécessaires lors de creer une mission, un message erreur apparait en haut de la page et rester la même page.

1. index()

Entrée : Aucun

Sortie : Les missions

Préconditions : Il y a le mission dans le base de données

Postconditions : Afficher la liste des missions crée par utilisateur

2. add()
Entrée : Mission
Sortie : Mission
Préconditions : La mission n'existe pas
Postconditions : La mission a bien été ajoutée dans la base de données
3. edit()
Entrée : Mission
Sortie : Mission
Préconditions : La mission existe et sélectionnée
Postconditions : La mission a bien été modifiée dans la base de données
4. delete()
Entrée : Mission
Sortie : Aucun
Préconditions : La mission existe et sélectionnée
Postconditions : Cette mission ont bien été supprimées de la base de données
5. generation()
Entrée : Id de mission, le nom de fichier
Sortie : Un fichier PDF
Préconditions : Le formulaire rempli correctement, contenant les informations de la mission.
Postconditions : PDF d'ordre de mission PDF1 est généré.
6. generationValid()
Entrée : Id de mission, le nom de fichier
Sortie : Un fichier PDF
Préconditions : Les informations de la mission est suffisant.
Postconditions : PDF de la validité de mission PDF2 est généré.
7. __calculNombreRepasNuits()
Entrée : La date début de mission, la date fin de mission.
Sortie : Un tableau qui contient le nombre de repas et le nombre de nuits.
Préconditions : La date début de mission est avant que la date de mission.
Postconditions : Il y a le nombre de repas et le nombre de nuits.
8. __sendSubmit()
Entrée : Id de mission
Sortie : Un email
Préconditions : La mission est créée avec succès.
Postconditions : Un email a été envoyé au chef d'équipe.
9. __sendPassager()
Entrée : Id de mission
Sortie : Un email
Préconditions : La mission est créée avec succès.
Postconditions : Un email a été envoyé aux passagers ensemble.

10. valid()
 Entrée : Id de mission
 Sortie : Aucun
 Préconditions : La mission est dans la liste de mission à valider.
 Postconditions : La mission a été validé.
11. __sendValidation()
 Entrée : Id de mission
 Sortie : Un email
 Préconditions : La mission a été validé avec succès.
 Postconditions : Un email a été envoyé aux secretaire.
12. needValidation()
 Entrée : Aucun
 Sortie : Les missions
 Préconditions : Il y a la mission à valider dans la base de données.
 Postconditions : Afficher la liste des missions à valider.
13. alreadyvalid()
 Entrée : Aucun
 Sortie : Les missions
 Préconditions : Il y a la mission a été validé dans la base de données.
 Postconditions : Afficher la liste des missions validé.
14. __sendConfirmModif()
 Entrée : Id de mission
 Sortie : Un email
 Préconditions : La mission existant est été modifié.
 Postconditions : Un email a été envoyé au créateur de mission.
15. listeMissionsParAnnee()
 Entrée : Année
 Sortie : Les missions
 Préconditions : Il y a le mission dans le base de données.
 Postconditions : Afficher la liste des missions de cette année entrée.

2 Définition de la fonction 2 : Visualisation des statistiques

Présenter la fonction 2 :

- Nom de la fonction : Visualisation des statistiques
- Cette fonctionnalité permet à Cette fonction permet d'afficher les informations pertinentes du laboratoire lifat (y compris projet, mission, membre) sous forme de graphiques ou de tableaux.
 Les utilisateurs peuvent connaître d'un coup d'œil le laboratoire LIFAT : en 2019, le pourcentage d'articles publiés dans différentes directions de recherche par rapport au nombre total d'articles. Ou, un graphique des tendances de la recherche du laboratoire

- LIFAT en 20xx et un graphique du nombre d'articles publiés par le laboratoire LIFAT en 20xx-20xx, et toutes les missions exécutées dans 20xx.etc
- Primordiale

Description de la fonction 2 :

- Comme le laboratoire dispose d'une grande quantité d'informations, nous devons effectuer des travaux statistiques. Cette fonction consiste à compter les informations existantes, afin que les utilisateurs puissent connaître plus clairement les informations clés liées au laboratoire sous forme de graphiques ou de tableaux. Par exemple :
 - ▷ Toutes les missions effectuées par le laboratoire en 20xx ;
 - ▷ Nombre de publications publiées par LIFAT Labs en 20xx - 20xx
 - ▷ Le top 10 du nombre de projets associés à chaque mission en 20xx (chaque mission a un projet associé)
 - ▷ Chaque motif dans 20xx en pourcentage de tous les motifs de missions (chaque mission a un motif).
- Cette fonction analysera également certaines données existantes pour obtenir des informations cachées. Par exemple, les tendances de recherche du laboratoire dans un certain domaine et les principales directions de recherche.
- Cette fonction interagit à la fonction «export». L'utilisateur a le droit d'exporter les graphiques et tableaux.

1. tableauToutesMissionsAnnnee()

Entrée : Année

Sortie : Le tableau des missions

Préconditions : L'utilisateur a choisi bien le bouton «Export Liste».

Postconditions : La liste de missions avec le titre affiche à l'écran.

N°	Motif	Lieu	départ	Arrivée
2896	CONSEIL LI	Paris	12/02/2019	15/03/2019
2897	ECOLE D'ETE	Lyon	13/04/2019	19/04/2019
3587	JURY DE THESE	Tours	25/04/2019	05/05/2019
2534	ENCADREMENT DE DOCTORANT	Tours	25/05/2019	03/06/2019
2654	REUNION EQUIPE	Orleans	05/06/2019	26/06/2019
2555	VISITEUR COLLOQUE	Paris	06/07/2019	06/07/2019
2666	TRAVAUX DE RECHERCHE	Paris	17/07/2019	15/08/2019
2634	AUTRE	Bourges	10/10/2019	21/11/2019
2634	VISITEUR COLLOQUE	Paris	22/11/2019	25/11/2019
2645	TRAVAUX DE RECHERCHE	Bourges	12/12/2019	15/12/2019

Figure C.1 – Toutes les missions effectuées par le laboratoire en 2019

2. grapheNbPubAnnToAnn() (Continue)

Entrée : Année, Année

Sortie : Le graphe

Préconditions : L'utilisateur a choisi bien le bouton «Export Graphique».

Postconditions : Le graphe pour compte le nombre de publication avec le titre affiche à l'écran.



Figure C.2 – Nombre de publications publiées par LIFAT Labs en 2014 - 2019

3. `grapheTop10ProjetAssocieAnn()` (Continue)

Entrée : Année

Sortie : Le graphe

Préconditions : L'utilisateur a choisi bien le bouton «Export Graphique».

Postconditions : Le graphe pour compte le top 10 plus importants projets associés avec le titre affiche à l'écran.



Figure C.3 – Top 10 du nombre de projets associés à chaque mission en 2019

4. `graphePourcentageMotifAnn()` (Continue)

Entrée : Année

Sortie : Le graphe

Préconditions : L'utilisateur a choisi bien le bouton «Export Graphique».

Postconditions : Le graphe pour compter chaque motif dans 20xx en pourcentage de tous les motifs de missions avec le titre affiche à l'écran.

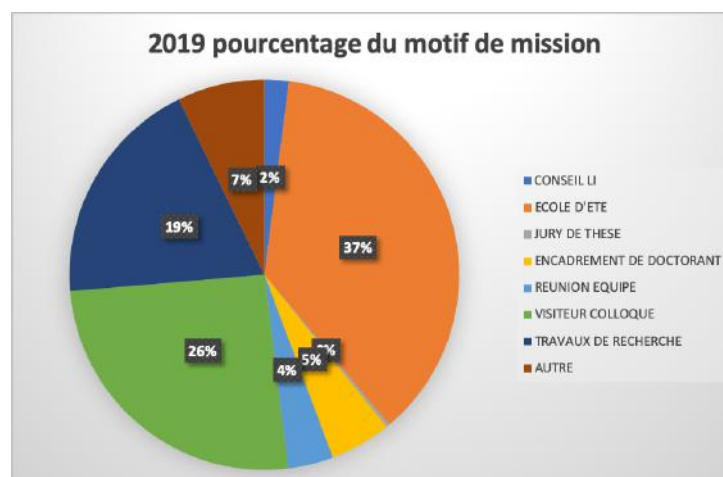


Figure C.4 – 2019 pourcentage du motif de mission

5. `grapheTendanceRechercheDansUnDomaine()` (Continue)

Entrée : Type de these (C'est une domaine)

Sortie : Le graphe

Préconditions : L'utilisateur a choisi bien le bouton «Export Graphique».

Postconditions : Le graphe pour montrer la tendance de recherche en 2010-2019 dan un certain domaine avec le titre affiche à l'écran.

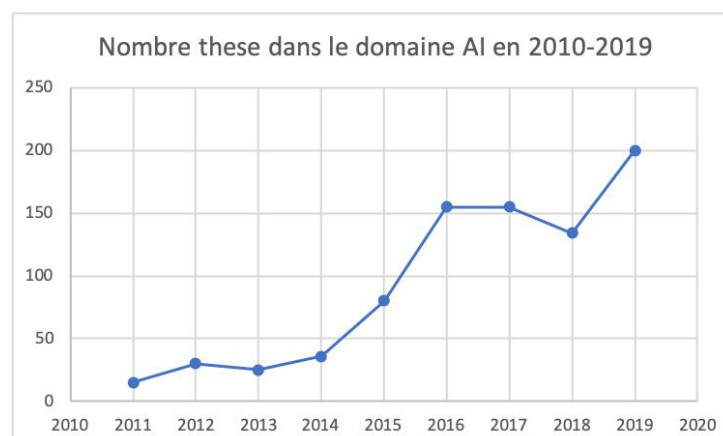


Figure C.5 – Le graphe de tendance de recherche dans le domaine AI en 2010-2019

3 Définition de la fonction 3 : Connection via CAS (Continue)

Présenter la fonction 3 :

- Nom de la fonction : Connection via CAS
- Cette fonction permet l'utilisateur se connecter à la site par le compte de l'ENT si l'utilisateur as associé le compte lors de l' inscription ou dans le profil.
- Primordiale

Description de la fonction 3 :

- Si l'utilisateur a un compte de l'ENT, c'est-à dire un compte de Université de Tours. Cette fonction permet de se connecter à ce site en utilisant le compte de Université de

Tours. Pas besoin le processus d'inscription. Si l'utilisateur sélectionne cette façon de connecter, le site va sauter directement à l'interface de connexion de Université. Après avoir entré les informations de compte de l'utilisateur, cliquez sur "Se connecter", la page reviendra automatiquement à la page d'accueil du système LIFAT. À ce stade, la fonction de connexion a été implémentée.

- Avant connecter via CAS. L'utilisateur a besoin d'avoir un compte de Université Tours. Cette fonction interagit à la fonction«login».

4 Définition de la fonction 4 : Verification du mail(Continue)

Présenter la fonction 4 :

- Nom de la fonction : Verification du mail
- Cette fonctionnalité permet à l'utilisateur d'accepter un courrier électronique d'activation de compte pour enregistrer le compte.
- Primordiale

Description de la fonction 4 :

- Lorsque l'utilisateur enregistre son compte, il entre les informations personnelles nécessaires. Cette fonction sera appelée. Cette fonction implémente l'envoi d'un courrier électronique de vérification à l'adresse électronique enregistrée. Après avoir cliqué sur le lien d'activation figurant dans le courrier électronique, l'utilisateur termine l'enregistrement.
- Avant vérifier un email. L'adresse email doit être une adresse email qui n'a pas été enregistrée sur ce site. Cette fonction interagit à la fonction«edit» (La fonction d'ajouter une membre)

5 Définition de la fonction 5 : Génération automatique de rapport d'activité annuel(Continue)

Présenter la fonction :

- Nom de la fonction : Génération automatique de rapport d'activité annuel.
- Cette fonctionnalité permet de générer le rapport d'activité annuel.
- Primordiale

Description de la fonction 5 :

- Ce rapport annuel contient toutes les données relatives au laboratoire LIFAT au cours de l'année. Le rapport est basé sur chaque membre. Répertoriez les projets que chaque membre a réalisés au cours de l'année (et toutes les informations relatives au projet, telles que le titre du projet, le budget du projet, le financement, date de début et date de fin du projet, etc.), les missions exécutées, les thèses recherchées et les fichiers uploadés, ainsi que des informations détaillées relatives à chaque élément.
- Cette fonction interagit à la fonction «export».L'utilisateur a le droit d'exporter le rapport annuel.

6 Définition de la fonction 6 : Gestion des publications (Continue)

Présenter la fonction :

- Nom de la fonction : Gestion des publications.

- Cette fonction permet d'ajouter / modifier / supprimer une publications et visualiser tous les publications de laboratoire LIFAT.
- Facultative

Description de la fonction 6 :

- Les membres du laboratoire LIFAT téléchargeront les publications dans le système. cette fonction est de gérer ces publications. L'utilisateur (appelé ici membre du laboratoire) peut télécharger le publication dans le système et le modifier ou le supprimer. Les utilisateurs de tous les sites Web ont le droit de visualiser tous les publications.
- Avant gérer des publications, l'utilisateur a besoin de se connecter à ce système. Cette fonction interagit à la fonction «login» et le composant «publication».
- Cette fonction interagit à la fonction «export».L'utilisateur a le droit d'exporter le rapport annuel.

1. upload()

Entrée : Tous attributs de la classe Publication

Sortie : Aucun

Préconditions : formulaire rempli correctement, contenant les informations de la publication(le title, les acteurs, le type etc.)

Postconditions : la publication a bien été ajouté à la base de données

2. edit()

Entrée : publication

Sortie : publication

Préconditions : publication existent et sélectionné

Postconditions : le contenu de la publication a bien été modifié

3. delete()

Entrée : publication

Sortie : aucun

Préconditions : publication existent et sélectionné

Postconditions : ce publication ont bien été supprimés de la base de données

4. listPublications()

Entrée : aucun

Sortie : publication

Préconditions : Il y a le publication dans le base de données

Postconditions : Afficher la liste des publication crée par utilisateur

5. listAllPublications()

Entrée : aucun

Sortie : publication

Préconditions : Il y a le publication dans la base de données

Postconditions : Afficher la liste des publications dans le système

D

Spécifications non fonctionnelles

1 Contraintes de développement et conception

- L'une des contraintes importantes sera le site doit marcher sur le serveur de l'Université Tours
- La deuxième contrainte est que la machine virtuelle de l'environnement de développement actuel doit avoir la même configuration qu'à celle du serveur de l'Université.

2 Contraintes de fonctionnement et d'exploitation

2.1 Performances

Les performances utilisateurs attendues sont celles d'un service web avec un temps de chargement moyen d'une page web (environ 2s maximum selon httparchive.org).

2.2 Capacités

Selon la capacité du serveur qui nous sera fourni, nous allons prendre en compte le nombre d'utilisateurs pouvant être connectés simplement sur notre application. D'après nos informations, il y aura au maximum 90 utilisateurs simultanés.

2.3 Contrôlabilité

Le framework CakePHP nous permet de suivre l'évolution du système à chaque instant grâce à des fichiers de logs.

2.4 Sécurité

Nous utiliserons un système d'identification similaire à celui du site ODM. Un utilisateur pourra se connecter de deux façons différentes :

- login = adresse universitaire + mot de passe
- en utilisant le service central authentication service (CAS) utilise pour se connecter a l'ENT de l'universite

Chaque compte aura un type precis lui permettant d'accéder a la base, ce qui restreindrait le champ d'actions possibles sur le site.

1 Aperçu de la diagramme de classe entier

La diagramme de classe entier est comme Figure E.1.

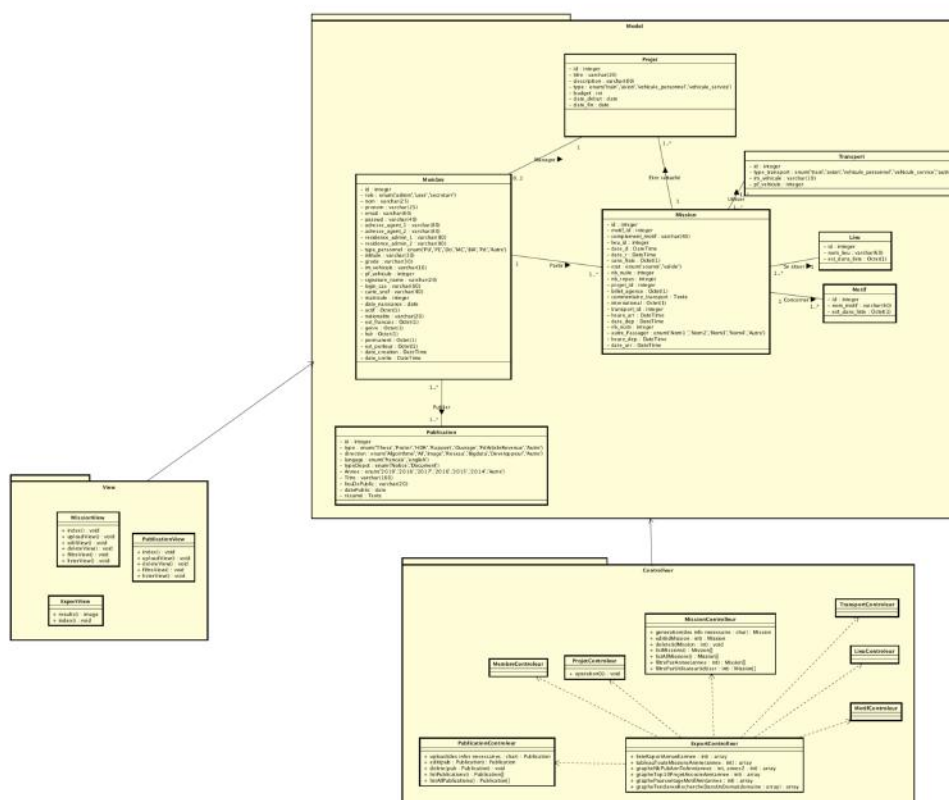


Figure E.1 – Le diagramme de classe entier

2 Descriptions détaillées des classes du package « model »

Dans le package «model», on a besoin de créer 5 classes, la classe «Mission» «Transport» «Lieu» «Motif» et «Publication». Pas le model de rapport annuel. Parce que nous n'effectuerons pas d'opérations ultérieures sur le rapport. Aucun nouvel élément de données n'est généré, nous ne choisissons donc pas de créer une entité.

Classe « Mission »

Cette classe est modèle de mission. Elle contient des attributs base de la mission.

♠ Attributs :

- id : (integer) l'identifiant de mission
- projet_id : (integer) l'identifiant de projet
- sans_frais : (octet) pour indiquer si la mission a frais
- motif_id : (integer) l'identifiant de motif
- complement_motif : (varchar) : Par exemple, le nom de conférence
- lieu_id : (integer) l'identifiant de lieu
- date_depart : (DateTime) date et heure de départ
- date_depart_arrive : (DateTime) date et heure d'arrivée de départ
- date_retour : (DateTime) date et heure de départ de retour
- date_retour_arrive : (DateTime) date et heure d'arrivée de retour
- transport_id : (integer) l'identifiant de transport
- autre_passager : enum ('Nom1', 'Nom2', 'Nom3', 'Nom4', 'Autre') Le nom des membres
- commentaire_transport : (Texte) Des compléments sur le transport pour le secrétariat, par exemple, l'horaire etc.
- etat : enum ('soumis', 'valide') : état de mission. que 2 genres soumis et valide
- nb_nuits : (integer) nombre de nuitées pour ce mission
- nb_repas : (integer) nombre de repas pour ce mission
- billet_agence : (octet) : Pour indiquer si le secrétariat commander les billets

Classe « Transport »

Cette classe est modèle de transport. Elle contient des attributs base du transport.

♠ Attributs :

- id : (integer) l'identifiant de transport
- type_transport : enum ('train', 'avion', 'vehicule_personnel', 'vehicule_service', 'autre') le type de transport.
- im_vehicule : varchar(10) : Immatriculation véhicule, c'est quel véhicule
- pf_vehicule : (integer) la puissance fiscale véhicule

Classe « Lieu »

Cette classe est modèle de lieu. Elle contient des attributs base du lieu.

♠ Attributs :

- id : (integer) l'identifiant de lieu
- nom_lieu : (enum ('Tours', 'Paris', 'Orleans', 'Caen', ..., 'autre')) nom de lieu
- est_dans_liste : (Octet) Pour indiquer si le lieu est dans la liste. Si l'utilisateur choisit le autre dans la rubrique "nom_lieu" "est_dans_liste" est "false" et écrire le lieu dans la rubrique "nouveau_lieu".
- pays : (varchar) Le pays de ce ville

Classe « Motif »

Cette classe est modèle de motif. Elle contient des attributs base du motif.

♠ Attributs :

- id : (integer) l'identifiant de motif
- nom_motif : (enum('CONSEIL LI', 'ECOLE D'ETE', 'JURY DE THESE', 'REUNION D'EQUIPE',...))

Classe « Publication »

Cette classe est modèle de publication. Elle contient des attributs base de la publication.

♠ Attributs :

- id : (integer) l'identifiant de publication
- type : enum('These', 'Poster', 'HDR', 'Rapport', 'Ouvrage', 'PdArticleRevenue',...) Le type de la publication
- direction : enum('Algorithmique', 'AI', 'Image', 'Reseau', 'Bigdata', 'Developpeur',...) Le direction de la publication
- langage : enum('français', 'english') : la language de la publication
- typeDepot : enum('Notice', 'Document') : le type de dépôt de la publication
- annee : enum('2019', '2018', '2017', '2016', '2015', '2014', 'Autre') L'année publie de la publication
- titre : (varchar) Le titre de la publication
- lieuDePublic : (varchar) Le lieu publie de la publication
- datePublic : (Date) La date publie de la publication
- résumé : (Texte) Le resume de la publication

3 Descriptions détaillées des classes du package « contrôleur »

Dans le package «Controleur», on a besoin de creer 6 classe. La classe «MissionController», «ExportController», «TransportController», «LieuController», «MotifController», «PublicationController» ■
«TransportController», «LieuController», «MotifController» que realise de gerer le transport, lieu et motif. Donc on ne répète pas.

Classe « MissionController »

Cette classe est pour exécuter et réaliser la gestion de mission : créer, supprimer, modifier et visualiser la mission. De plus, filtrer les missions par l'année ou par utilisateur.

♠ Méthodes :

- + index() : cette méthode est pour lister tous les missions de l'utilisateur actuel.
Paramètre :
 - Rien
 Valeur retournée :
 - missionListe : Mission[]
- + add() : cette methode est pour créer une mission.
Paramètre :
 - idMission : int
 Valeur retournée :
 - mission : Mission
- + edit() : cette methode est pour modifier une mission.
Paramètre :

- idMission : int
- Valeur retournée :
 - mission : Mission
- + delete() : cette methode est pour supprimer une mission.
 - Paramètre :
 - idMission : int
 - Valeur retournée :
 - rien
- + generation() : cette méthode est pour générer le pdf d'ordre de mission PDF1.
 - Paramètre :
 - idMission : int , filename : string
 - Valeur retournée :
 - fichier PDF : FPDF
- + generationValid() : cette méthode est pour générer pdf de la validité de mission PDF2.
 - Paramètre :
 - idMission : int , filename : string
 - Valeur retournée :
 - fichier PDF : FPDF
- + __calculNombreRepasNuite() : cette méthode est pour calculer le nombre de nuit et le nombre de repas d'une mission.
 - Paramètre :
 - dateDebut : Date, dateFin : Date
 - Valeur retournée :
 - array
- + __sendSubmit() : cette méthode est pour envoyer un email de soumission au chef d'équipe.
 - Paramètre :
 - idMission : int
 - Valeur retournée :
 - email : Email
- + __sendPassager() : cette méthode est pour envoyer un email de soumission à les passagers avec cette mission.
 - Paramètre :
 - idMission : int
 - Valeur retournée :
 - email : Email
- + valid() : cette méthode est pour valider un ordre de mission.
 - Paramètre :
 - idMission : int
 - Valeur retournée :
 - rien
- + __sendValidation() : cette méthode est pour envoyer un email aux secrétaires après confirmation du chef d'équipe.
 - Paramètre :
 - idMission : int
 - Valeur retournée :
 - email : Email
- + needValidation() : cette méthode est pour Lister toutes les missions en attente de validation.
 - Paramètre :
 - Rien
 - Valeur retournée :
 - missionListe : Mission[]
- + alreadyvalid() : cette méthode est pour afficher tous les mission validés.

- Paramètre :
- Rien
- Valeur retournée :
- missionListe : Mission[]
- + `__sendConfirmModif()` : cette méthode est pour envoyer la modification notifier par d'un OdM au créateur de l'OdM.
- Paramètre :
- idMission : int
- Valeur retournée :
- email : Email
- + `listeMissionsParAnnee()` : cette méthode est pour lister tous les missions par année dans un tableau.
- Paramètre :
- annee : int
- Valeur retournée :
- missionListe : Mission[]

Classe « ExportController »

Cette classe est pour réaliser la génération le rapport annuel ou des graphes, des tableaux .

♠ Méthodes :

- + `listeRaportAnnuel()` : Cette methode est pour générer le rapport d'activité annuel.(Continue)
- Paramètre :
- annee : int
- Valeur retournée :
- tableau : array
- + `tableauTouteMissionsAnnne()` : Cette methode est pour lister les missions effectuées par le laboratoire en 20xx.
- Paramètre :
- annee : int
- Valeur retournée :
- tableau : array
- + `grapheNbPubAnnToAnn()` : Cette methode est pour générer le graphe de nombre de publications publiées par LIFAT Labs en 20xx - 20xx.(Continue)
- Paramètre :
- annee : int, annee2 : int
- Valeur retournée :
- graphe : array
- + `grapheTop10ProjetAssocieAnn()` : Cette methode est pour générer le graphe de Le top 10 du nombre de projets associés à chaque mission en 20xx.(Continue)
- Paramètre :
- annee : int
- Valeur retournée :
- graphe : array
- + `graphePourcentageMotifAnn()` : cette methode est pour générer le graphe de compter que chaque motif dans 20xx en pourcentage de tous les motifs de missions.(Continue)
- Paramètre :
- annee : int
- Valeur retournée :
- graphe : array
- + `grapheTendanceRechercheDansUnDomai()` : cette methode est pour générer le graphe des tendances de recherche du laboratoire dans un certain domaine.(Continue)

Paramètre :
 - domaine : array
 Valeur retournée :
 - graphe : array

Classe « PublicationControler »(Continue)

Cette classe est pour réaliser la gestion de publicationAjouter, supprimer, modifier et visualiser la publication.

- ♠ Méthodes :
- + upload() : cette methode est pour ajouter une publciation.
 Paramètre :
 - Tous les attributs de la classe Publication
 Valeur retournée :
 - publication : Publication
 - + edit() : cette methode est pour modifier une publciation.
 Paramètre :
 - publication : Publication
 Valeur retournée :
 - publication : Publication
 - + delete() : cette methode est pour supprimer une publciation.
 Paramètre :
 - publication : Publication
 Valeur retournée :
 - rien
 - + listPublication() : cette methode est pour visualiser une publciation.
 Paramètre :
 - rien
 Valeur retournée :
 - publication : Publication
 - + listAllPublications() : cette methode est pour lister tous les publciations.
 Paramètre :
 - rien
 Valeur retournée :
 - listePublication : Publication[]

Classe « TransportController »

Cette classe est pour gérer le transport.

- ♠ Méthodes :
- + add : cette méthode est pour ajouter un transport
 Paramètre :
 - Tous les attributs de la classe Transport
 Valeur retournée :
 - transport : Transport
 - + edit : cette méthode est pour modifier un transport
 Paramètre :
 - transport : Transport
 Valeur retournée :
 - transport : Transport
 - + view : cette méthode est pour visualiser un transport
 Paramètre :

- type_transport : varchar
- Valeur retournée :
- transport : Transport
- + delete : cette méthode est pour supprimer un transport
- Paramètre :
- transport : Transport
- Valeur retournée :
- rien

Classe « LieuController »

Cette classe est pour gérer le lieu.

- ♠ Méthodes :
- + add : cette méthode est pour ajouter un lieu
- Paramètre :
- Tous les attributs de la classe Lieu
- Valeur retournée :
- lieu : Lieu
- + edit : cette méthode est pour modifier un lieu
- Paramètre :
- lieu : Lieu
- Valeur retournée :
- lieu : Lieu
- + view : cette méthode est pour visualiser un lieu
- Paramètre :
- nomLieu : varchar
- Valeur retournée :
- lieu : Lieu
- + delete : cette méthode est pour supprimer un lieu
- Paramètre :
- lieu : Lieu
- Valeur retournée :
- rien

Classe « MotifController »

Cette classe est pour gérer le motif.

- ♠ Méthodes :
- + add : cette méthode est pour ajouter un motif
- Paramètre :
- Tous les attributs de la classe Motif
- Valeur retournée :
- motif : Motif
- + edit : cette méthode est pour modifier un motif
- Paramètre :
- motif : Motif
- Valeur retournée :
- motif : Motif
- + view : cette méthode est pour visualiser un motif
- Paramètre :
- nomMotif : varchar
- Valeur retournée :

- motif : Motif
- + delete : cette méthode est pour supprimer un motif
- Paramètre :
 - motif : Motif
- Valeur retournée :
 - rien

4 Descriptions détaillées des classes du package « view »

Dans le framework cakephp3.7, tout le code de l'interface frontale est placé dans le dossier **Template**. Chaque classe a un sous-dossier, et sous le sous-dossier, il y a plusieurs fichiers frontaux.

Chaque fonctionnalité a une interface. Donc, on a plusieurs .ctp fichiers. .ctp est le suffixe de fichier de l'interface dans le framework cake3.7)

Dans le dossier Template\Missions

Il y a 7 .ctp fichiers :

- + add.ctp : l'interface de créer une mission
- + alreadyvalid.ctp : l'interface de Lister toutes les missions validées
- + edit.ctp : l'interface de modifier une mission
- + index.ctp : l'interface de lister tous les missions
- + need_validation.ctp : l'interface de Lister toutes les missions à valider
- + generation.ctp : l'interface de générer le pdf d'ordre de mission
- + generation_valid.ctp : l'interface de générer le pdf de la validité de mission

Dans le dossier Template\Lieux

Il y a 4 .ctp fichiers :

- + add.ctp : l'interface de ajouter un lieu
- + edit.ctp : l'interface de modifier un lieu
- + index.ctp : l'interface d'afficher tous les lieux
- + view.ctp : l'interface de visualiser un lieu

Dans le dossier Template\Transports

Il y a 4 .ctp fichiers :

- + add.ctp : l'interface de ajouter un transport
- + edit.ctp : l'interface de modifier un transport
- + index.ctp : l'interface d'afficher tous les transports
- + view.ctp : l'interface de visualiser un transport

Dans le dossier Template\Motifs

Il y a 4 .ctp fichiers :

- + add.ctp : l'interface de ajouter un motif
- + edit.ctp : l'interface de modifier un motif
- + index.ctp : l'interface d'afficher tous les motifs
- + view.ctp : l'interface de visualiser un motif

F

Document d'installation

Ce document regroupe toutes les informations nécessaires pour l'installation du projet sur les machines, ainsi que pour sa mise en production. Certains points présentent des spécificités pour la partie développement ou mise en production, qui sont précisées ci-dessous par les mots-clés "**Dev**" (pour les machines de développement) et "**Prod**" (pour la machine qui hébergera le site de manière à ce qu'il soit accessible par tout le monde).

1 Prérequis de CakePHP

Apache / MySQL / PHP

Ce projet a été créé avec CakePHP 3.7, Apache 2.4.25, PHP 7.3 et MariaDB 10.1.

Avant de commencer, veuillez vérifier que votre machine est à jour :

1. `sudo apt-get update`
2. `sudo apt-get upgrade`

Dev : Installer un **package AMP** (Apache + MySQL + PHP), tel que XAMPP (<https://www.apachefriends.org/fr/download.html>) ou WampServer (<http://www.wampserver.com/>) ■

Après installation, il peut y avoir un conflit de ports avec VMWare Workstation. Dans ce cas, il faut résoudre le problème en allant dans **VMWare Workstation : Edit > Preferences > Shared VMs**.

Prod : Il n'est **pas recommandé** de faire la même chose. Il faut plutôt utiliser les commandes suivantes :

1. `sudo apt-get install apache2`
2. `sudo apt-get install mysql-server`
3. `sudo mysql_secure_installation`
4. `sudo apt-get install php libapache2-mod-php`

Extensions de PHP

Plusieurs extensions de PHP sont nécessaires pour le fonctionnement de ce projet : **mbstring**, **intl**, **simplexml** et **PDO**. Celles-ci sont donc à intégrer aux machines utilisées, mais leur installation dépend de chaque environnement.

Cependant, il est possible de commencer par rechercher les lignes correspondantes dans le fichier **php.ini** situé dans le dossier d'installation de PHP, et de les décommenter en enlevant le “;” au début de la ligne. Parfois, installer les packages de PHP (par exemple : **php-intl**) via **apt** ou **apt-get** est aussi requis.

Une fois ces manipulations faites, il suffit de redémarrer le serveur (eventuellement avec la commande Linux : `sudo service apache2 restart`).

Git

Pour installer **Git**, il faut le télécharger à cette adresse pour Windows : <https://gitforwindows.org/>, ou bien d'utiliser la ligne de commande `sudo apt-get install git` sous Linux. Il est également possible de lier sa machine avec son compte Git avec les commandes :

1. `git config --global user.name "... "`
2. `git config --global user.email x@example.com`

(Bien entendu, à vous d'y mettre vos identifiants à la place de ce qui est surligné en jaune...)

Composer

Il est nécessaire de télécharger Composer soit directement sur <https://getcomposer.org/Composer-Setup.exe> pour les utilisateurs de Windows, soit en ligne de commande pour Linux :

1. `php -r "copy('https://getcomposer.org/installer', 'composer-setup.php');"`
2. `php -r "if (hash_file('sha384', 'composer-setup.php') === '48e3236262b34d30969dca3c37281b3b4bbe3221bda8db0dcd0615698a5cbe587c3f0fe57a54d8f5') echo 'Installer verified'; else echo 'Installer corrupt'; unlink('composer-setup.php'); echo PHP_EOL;"`
3. `php composer-setup.php`
4. `php -r "unlink('composer-setup.php');"`
5. `sudo mv composer.phar /usr/local/bin/composer`

2 Mise en place du projet

Télécharger le projet

Pour cloner le projet sur chaque machine, il faut tout d'abord se placer dans le dossier de **publication des sites** (par exemple : **C :\xampp\htdocs** ou encore **/var/www/html**) , ainsi que d'exécuter dans le terminal de git la commande suivante :

```
https://github.com/qinglingM/PRD-Projet-LIFAT.git
```

A la suite de cela, il faut compléter le projet en utilisant Composer via l'invite de commande :

```
cd PRD-Projet-LIFAT/cake3_7/
composer install
```

Base de donnees

Toutes les lignes de commande détaillées ci-dessous sont à exécuter depuis le **dossier “SQL” du projet**, mais il est possible d'utiliser PHPMyAdmin à la place. Pour les utilisateurs de Windows, ne pas oublier d'ajouter le **dossier “bin” de MySQL au PATH**.

1. Création et import

Les opérations suivantes sont à réaliser avec un utilisateur MySQL ayant les droits d'administrateur : `sudo mysql -u root -p` (dans cet exemple : la base de données s'appelle **projetlifat1** ; vous pouvez changer cela, mais il faudra répercuter cette modification sur un fichier décrit plus bas).

- `create database projetlifat1`
- `use projetlifat1`
- `source projetlifat1.sql`

Il est également possible, en mode **dev** uniquement, d'importer un jeu de données pour pouvoir tester le projet à l'aide de la commande : `source stub.sql`. Dans ce jeu d'essai, l'email administrateur est : “admin@admin.fr” et le mot de passe est “admin”.

2. Nouvel utilisateur MySQL

Pour créer un nouvel utilisateur (dans cet exemple : l'utilisateur s'appelle **lifat_app** et son mot de passe est **darktranquillity** ; vous pouvez changer ces informations, ces modifications seront à répercuter sur un fichier décrit plus bas), il faut suivre la démarche suivante :

- `create user 'lifat_app'@'localhost' identified by 'darktranquillity'`
- `grant all privileges on projetlifat1.* to 'lifat_app'@'localhost' ;`

3. Base de test

En environnement **Dev uniquement !**

Il est possible d'exécuter vos tests dans une autre base de données.

Pour cela, il suffit de **refaire toutes les opérations** décrites dans les deux derniers paragraphes (**sauf create user ...**), pour une autre base de données nommée **lifat_db_test** (nom modifiable avec les mêmes contraintes qu'auparavant).

Finalisation de l'application

1. Droits des dossiers

L'application génère et manipule automatiquement certains types de fichiers, mais il y a de fortes chances que le système de fichiers de votre système d'exploitation lui refuse l'accès aux dossiers du projet. En effet, PHP est souvent exécuté avec un utilisateur autre que le vôtre.

Par exemple, sous **Unix**, dans le dossier **PRD-PROJET-LIFAT/cake3_7/**, il faut exécuter la commandes suivantes :

```
chmod a+w webroot/
```

2. Fichier de configuration

Le fichier de configuration du projet se nomme “**app.php**” et est situé dans “**PRD-PROJET-LIFAT/cake3_7/config/app.php**”. Il faut donc dans un premier temps l'ouvrir, puis modifier la directive 'debug' selon l'environnement utilisé :

- (**Dev**) : 'debug' => true,
- (**Prod**) : 'debug' => false,

Il n'est pas nécessaire de modifier les paramètres d'accès à la base de données si toutes les instructions sus-citées ont bien été suivies.

3 Accès à l'application

Tout est prêt !

Il ne reste plus qu'à ouvrir un navigateur et taper l'adresse suivante : http://localhost/PRD-PROJET-LIFAT/cake3_7/. Ceci n'est valable que pour une machine sur laquelle toute l'installation aurait été mise en place.

Autrement, il faut remplacer “localhost” par l'adresse IP du serveur. Par exemple, depuis le réseau filaire de l'université, il faut utiliser l'adresse : http://10.236.5.39/PRD-PROJET-LIFAT/cake3_7/.

4 Mettre à jour en production

En environnement Prod uniquement !

En plus de la commande `git pull` (et éventuellement `composer update`) habituelle, il est recommandé de nettoyer le cache de CakePHP afin de déployer des changements à une application en production en utilisant (depuis le dossier **PRD-PROJET-LIFAT/cake3_7/**) la commande : `bin/cake schema_cache clear`. (Attention aux underscores !)

Enfin, afin d'optimiser les performances de l'application en production, la documentation de CakePHP recommande d'exécuter les commandes suivantes (toujours depuis **PRD-PROJET-LIFAT/cake3_7/**) :

- `composer dumpautoload -o`
- `rm webroot/debug_kit`
- `bin/cake plugin assets symlink`



Document d'utilisation

En raison de l'épidémie, le système n'a pas été déployé sur les serveurs de l'école. Par conséquent, pour utiliser ce système, vous devez installer le système étape par étape conformément au document d'installation : Installer un package AMP, télécharger le code source du système à partir de git et configurer la base de données. Une fois que tout est prêt, les utilisateurs n'ont qu'à entrer l'URL [http : // localhost/PRD-PROJET-LIFAT/cake3_7/](http://localhost/PRD-PROJET-LIFAT/cake3_7/) dans n'importe quel navigateur, et ils peuvent accéder et utiliser le système avec succès.

H

Cahier de test

Les tests de logiciels visent à garantir l'exactitude, l'intégrité, la sécurité et les performances du logiciel. Pour ce projet, le test est utilisé pour vérifier si chaque fonction peut fonctionner normalement.

Framework cakephp3.7 est livré avec un support de test complet intégré, CakePHP utilise PHPUnit comme framework de test de base. Donc, Le test pour ce projet est test unitaire. Le test unitaire consiste à tester chaque fonction ou chaque méthode d'une classe. Nous fournissons les données d'entrée à la méthode à tester et obtenons les résultats attendus à l'avance. Nous pouvons voir si le résultat de sortie de la méthode effectuée en utilisant les données d'entrée correspond à notre résultat attendu.

PHPUnit est la norme de facto pour les tests unitaires en PHP. Il suffit que la classe de test, il écrit la classe «TestCase» pour faire le test unitaire avec des cas de test. Cette librairie nous fournit de nombreuses méthodes d'assert, par exemple, la méthode «assertTrue ()» est utilisée pour vérifier si l'expression est vraie La méthode "assertEquals ()" est utilisée pour vérifier si deux variables sont égales. La méthode "AssertArrayEquals ()" est pour vérifier si les deux listes sont identiques, etc. Si la réponse de l'assertion est positive, le test réussit. Sinon, elle lèvera une exception. Enfin, une phrase "vendor / phpunit / phpunit /phpunit tests / TestCase / Controller / MissionsControllerTest" suffit pour démarrer le test. Une fois l'instruction exécutée, toutes les méthodes de la classe seront testées.

La **Figure H.2** montre le résultat de test unitaire avec succès.

La **Figure H.2** signifie que l'on a testé une méthodes dans cette classe, et tous les tests réussit. Si le test échoue, le résultat est comme **Figure H.3**.

Dans la **Figure H.3**, il affiche que deux méthodes sont testées, mais un test génère l'exception car le résultat de sortie n'est pas égal au résultat prévu.

```

/**
 * App\Controller\MissionsController Test Case
 */
class MissionsControllerTest extends TestCase
{
    use IntegrationTestTrait;
    /**
     * Fixtures
     *
     * @var array
     */
    public $fixtures = [
        'app.Missions',
        'app.Projets',
        'app.Lieux',
        'app.Motifs',
    ];
}

```

Figure H.1 – L'exemple de test unitaire avec PHPUnit

```

(base) localhost:cake3_7 mengqingling$ vendor/phpunit/phpunit/phpunit tests/TestCase/Controller/MissionsControllerTest
PHPUnit 6.5.14 by Sebastian Bergmann and contributors.

.
1 / 1 (100%)

Time: 581 ms, Memory: 14.00MB
OK (1 test, 1 assertion)

```

Figure H.2 – Le résultat de test unitaire avec succès

```

PROBLEMS 10 OUTPUT DEBUG CONSOLE TERMINAL
t: bash

There were 3 failures:
1) App\Test\TestCase\Controller\MissionsControllerTest::testAdd
Failed asserting that 'La date de début doit être avant la date de fin.' was in 'flash' message.
/Applications/MAMP/htdocs/PRD-Projet-LIFAT/cake3_7/vendor/cakephp/cakephp/src/TestSuite/IntegrationTestTrait.php:1108
/Applications/MAMP/htdocs/PRD-Projet-LIFAT/cake3_7/tests/TestCase/Controller/MissionsControllerTest.php:124
2) App\Test\TestCase\Controller\MissionsControllerTest::testDelete
Failed asserting that 'Suppression impossible, OdM déjà validé' was in 'flash' message.
/Applications/MAMP/htdocs/PRD-Projet-LIFAT/cake3_7/vendor/cakephp/cakephp/src/TestSuite/IntegrationTestTrait.php:1108
/Applications/MAMP/htdocs/PRD-Projet-LIFAT/cake3_7/tests/TestCase/Controller/MissionsControllerTest.php:162
3) App\Test\TestCase\Controller\MissionsControllerTest::testValid
Failed asserting that 'Mission déjà validé' was in 'flash' message.
/Applications/MAMP/htdocs/PRD-Projet-LIFAT/cake3_7/vendor/cakephp/cakephp/src/TestSuite/IntegrationTestTrait.php:1108
/Applications/MAMP/htdocs/PRD-Projet-LIFAT/cake3_7/tests/TestCase/Controller/MissionsControllerTest.php:239

```

Figure H.3 – Le résultat de test unitaire échoués

1 Tests unitaires de la classe « MissionsController »

Les tests unitaires des methodes de la classe « MissionsController » sont listes dans les tableaux suivants :

ID	IDENTIFICATION OF COMPONENT
1	Afficher toutes les missions de l'utilisateur identifié
DESCRIPTION OF THE TEST (granularity, scenario, values, actions)	
<p>Action : Tout type d'utilisateur (membre, chef d'équipe, secrétaire, administrateur)s'est connecté avec succès au système. Cliquer le bouton 'Mes Missions'.</p> <p>Cas 1 : Un utilisateur saisit correctement le username et le mot de passe. L'utilisateur n'a pas créé aucun mission.</p> <p>Cas 2 : Un utilisateur saisit correctement le username et le mot de passe. L'utilisateur a déjà créé des missions.</p>	
EXPECTED RESULTS	
<p>Cas 1 : l'utilisateur se connecte à ce système. Aucun missions affiche sur l'écran.</p> <p>Cas 2 : l'utilisateur se connecte à ce système. Tous les missions créé de l'utilisateur identifié affiche sur l'écran.</p>	
OBTAINED RESULTS	
<p>Cas 1 : l'utilisateur se connecte à ce système. Aucun missions affiche sur l'écran.</p> <p>Cas 2 : l'utilisateur se connecte à ce système. Tous les missions créé de l'utilisateur identifié affiche sur l'écran.</p>	

ID	IDENTIFICATION OF COMPONENT
2	Afficher les détails d'une mission spécifique
DESCRIPTION OF THE TEST (granularity, scenario, values, actions)	
<p>Action : Tout type d'utilisateur (membre, chef d'équipe, secrétaire, administrateur)s'est connecté avec succès au système. Cliquez sur le bouton 'Détails' derrière une mission.</p> <p>Cas 1 : Un utilisateur saisit correctement le username et le mot de passe. L'utilisateur voit une mission avec des informations incomplètes.</p> <p>Cas 2 : Un utilisateur saisit correctement le username et le mot de passe. L'utilisateur voit une mission avec des informations complètes.</p>	
EXPECTED RESULTS	
<p>Cas 1 : l'utilisateur se connecte à ce système. Il y a une exception qui indique l'erreur.</p> <p>Cas 2 : l'utilisateur se connecte à ce système. Afficher un pdf qui convient les détails de ce mission.</p>	
OBTAINED RESULTS	
<p>Cas 1 : l'utilisateur se connecte à ce système. Il y a une exception qui indique les informations sur la mission sont incomplètes, veuillez compléter.</p> <p>Cas 2 : l'utilisateur se connecte à ce système. Afficher un pdf qui convient les détails de ce mission.</p>	

ID	IDENTIFICATION OF COMPONENT
3	Ajouter une ordre de mission.
DESCRIPTION OF THE TEST (granularity, scenario, values, actions)	
Action : Tout type d'utilisateur (membre, chef d'équipe, secrétaire, administrateur)s'est connecté avec succès au système. Cliquez sur le bouton 'Nouveau mission'.	
Cas 1 : Un utilisateur saisit correctement le username et le mot de passe. L'utilisateur sélectionne la date de départ plus tard que la date de retour.	
Cas 2 : Un utilisateur saisit correctement le username et le mot de passe.,L'utilisateur ne sélectionne pas la date de départ et la date de retour.	
Cas 3 : Un utilisateur saisit correctement le username et le mot de passe.,L'utilisateur sélectionne la date de départ plus tôt que,la date de retour.	
EXPECTED RESULTS	
Cas 1 : l'utilisateur se connecte à ce système.,L'opération a échoué. Il y a une exception qui indique la date de départ doit être avant la date de retour.	
Cas 2 : l'utilisateur se connecte à ce système.,L'opération a échoué. Il y a une exception qui indique la date de départ doit être avant la date de retour.	
Cas 3 : l'utilisateur se connecte à ce système. Ajout d'un ordre de mission avec succès.	
OBTAINED RESULTS	
Cas 1 : l'utilisateur se connecte à ce système. Il y a une exception qui indique la date de départ doit être avant la date de retour.	
Cas 2 : l'utilisateur se connecte à ce système. Il y a une exception qui indique la date de départ doit être avant la date de retour.	
Cas 3 : l'utilisateur se connecte à ce système. Ajout d'un ordre de mission avec succès.	

ID	IDENTIFICATION OF COMPONENT
4	Générer le pdf d'ordre de mission
DESCRIPTION OF THE TEST (granularity, scenario, values, actions)	
Action : Quand l'utilisateur crée un ordre de mission, il va générer un pdf, d'ordre de mission qui convient tous les détails de ce mission .	
Cas 1 : L'utilisateur a entré des informations complètes sur la mission lors de la création de mission.	
Cas 2 :L'utilisateur a entré des informations incomplètes sur la mission lors de la création de mission.	
EXPECTED RESULTS	
Cas 1 : l'utilisateur se connecte à ce système.,Il y a une exception qui indique les informations sur la mission sont incomplètes, veuillez compléter.	
Cas 2 : l'utilisateur se connecte à ce système. PDF d'ordre de mission généré avec succès.	
OBTAINED RESULTS	
Cas 1 : l'utilisateur se connecte à ce système.,Il y a une exception qui indique les informations sur la mission sont incomplètes, veuillez compléter.	
Cas 2 : l'utilisateur se connecte à ce système. PDF d'ordre de mission généré avec succès.	

ID	IDENTIFICATION OF COMPONENT
5	Modifier une ordre de mission.
DESCRIPTION OF THE TEST (granularity, scenario, values, actions)	
<p>Action : Tout type d'utilisateur (membre, chef d'équipe, secrétaire, administrateur)s'est connecté avec succès au système. Cliquez sur le bouton 'Modifier' derrière une mission.</p> <p>Cas 1 : Un utilisateur saisit correctement le username et le mot de passe. Un membre veut modifier la mission d'un autre membre.</p> <p>Cas 2 : Un utilisateur saisit correctement le username et le mot de passe. l'utilisateur souhaite modifier une mission avec un état "valide".</p> <p>Cas 3 : Un utilisateur saisit correctement le username et le mot de passe. l'utilisateur souhaite modifier une mission avec un état "soumis".</p>	
EXPECTED RESULTS	
<p>Cas 1 : l'utilisateur se connecte à ce système.,L'opération a échoué. Il y a une exception qui indique permission insuffisante.</p> <p>Cas 2 : l'utilisateur se connecte à ce système.,L'opération a échoué. Il y a une exception qui indique Odm déjà validé, ne peut pas être changé.</p> <p>Cas 3 : l'utilisateur se connecte à ce système. Modifier un ordre de mission avec succès.</p>	
OBTAINED RESULTS	
<p>Cas 1 : l'utilisateur se connecte à ce système. Il y a une exception qui indique Modification impossible : permissions insuffisantes.</p> <p>Cas 2 : l'utilisateur se connecte à ce système. Il y a une exception qui indique Modification impossible, OdM déjà validé.</p> <p>Cas 3 : l'utilisateur se connecte à ce système. Modifier un ordre de mission avec succès.</p>	

ID	IDENTIFICATION OF COMPONENT
6	Générer pdf de la validité de mission
DESCRIPTION OF THE TEST (granularity, scenario, values, actions)	
<p>Action : Quand le chef d'équipe valide un ordre de mission, il va générer un pdf, de la validité de mission qui convient les informations importants de ce mission, et les signatures de chef .</p> <p>Cas 1 : Le chef d'équipe valide une mission.</p>	
EXPECTED RESULTS	
Cas 1 : PDF de la validité de mission généré avec succès.	
OBTAINED RESULTS	
Cas 1 :.PDF de la validité de mission généré avec succès.	

ID	IDENTIFICATION OF COMPONENT
7	Supprimer une ordre de mission.
DESCRIPTION OF THE TEST (granularity, scenario, values, actions)	
<p>Action : Tout type d'utilisateur (membre, chef d'équipe, secrétaire, administrateur)s'est connecté avec succès au système. Cliquez sur le bouton 'Supprimer' derrière une mission.</p> <p>Cas 1 : Un utilisateur saisit correctement le username et le mot de passe. Un membre veut supprimer la mission d'un autre membre.</p> <p>Cas 2 : Un utilisateur saisit correctement le username et le mot de passe. l'utilisateur souhaite supprimer une mission avec un état "valide".</p> <p>Cas 3 : Un utilisateur saisit correctement le username et le mot de passe. l'utilisateur souhaite supprimer une mission avec un état "soumis".</p>	
EXPECTED RESULTS	
<p>Cas 1 : l'utilisateur se connecte à ce système.,L'opération a échoué. Il y a une exception qui indique permission insuffisante.</p> <p>Cas 2 : l'utilisateur se connecte à ce système.,L'opération a échoué. Il y a une exception qui indique Odm déjà validé, ne peut pas être changé.</p> <p>Cas 3 : l'utilisateur se connecte à ce système. Supprimer un ordre de mission avec succès.</p>	
OBTAINED RESULTS	
<p>Cas 1 : l'utilisateur se connecte à ce système. Il y a une exception qui indique Suppression impossible : permissions insuffisantes.</p> <p>Cas 2 : l'utilisateur se connecte à ce système. Il y a une exception qui indique Suppression impossible, OdM déjà validé.</p> <p>Cas 3 : l'utilisateur se connecte à ce système. Supprimer un ordre de mission avec succès.</p>	

ID	IDENTIFICATION OF COMPONENT
8	Calculer le nombre de nuit et le nombre de repas d'une mission.
DESCRIPTION OF THE TEST (granularity, scenario, values, actions)	
<p>Action : Quand l'utilisateur crée un ordre de mission, utilisateur ne sait pas le nombre nuit et le nombre repas, le système auto-calcule en appeler ce fonction.</p> <p>Cas 1 : utilisateur ne sait pas le nombre nuit et le nombre repas, et la date de départ est avant la date de retour (C'est sure).</p>	
EXPECTED RESULTS	
<p>Cas 1 : le système auto-calcule le nb nuit et nb repas de ce mission et stocker dans la base de données.</p>	
OBTAINED RESULTS	
<p>Cas 1 : le nb nuit et nb repas calculés avec succès et sont stockés dans la base de données.</p>	

ID	IDENTIFICATION OF COMPONENT
9	Envoyer un email de soumission au chef d'équipe.
DESCRIPTION OF THE TEST (granularity, scenario, values, actions)	
Action : Quand l'utilisateur crée un ordre de mission, le système envoie immédiatement un email qui convient, le pdf d'ordre de mission au chef d'équipe. Cas 1 : L'utilisateur a réussi à créer un ordre de mission.	
EXPECTED RESULTS	
Cas 1 : Un email qui convient, le pdf d'ordre de mission est envoyé au chef d'équipe.	
OBTAINED RESULTS	
Cas 1 : Un email qui convient, le pdf d'ordre de mission est envoyé au chef d'équipe.	

ID	IDENTIFICATION OF COMPONENT
10	Envoyer un email de soumission à les passagers avec cette mission.
DESCRIPTION OF THE TEST (granularity, scenario, values, actions)	
Action : Quand l'utilisateur crée un ordre de mission, le système envoie immédiatement un email qui convient, le pdf d'ordre de mission à les passagers avec cette mission. Cas 1 : L'utilisateur a réussi à créer un ordre de mission et sélectionne les passagers quand créer la mission.	
EXPECTED RESULTS	
Cas 1 : Un email qui convient, le pdf d'ordre de mission est envoyé à les passagers.	
OBTAINED RESULTS	
Cas 1 : Un email qui convient, le pdf d'ordre de mission est envoyé à les passagers.	

ID	IDENTIFICATION OF COMPONENT
11	Valider un ordre de mission.
DESCRIPTION OF THE TEST (granularity, scenario, values, actions)	
Action : Chef d'équipe valide une mission. Cas 1 : Un membre voudrais valider une mission. Cas 2 : Le chef d'équipe valide une mission avec un état "valide". Cas 3 : Le chef d'équipe valide une mission avec un état "soumis".	
EXPECTED RESULTS	
Cas1 : L'opération a échoué. Il y a une exception qui indique permission insuffisante. Cas 2 : L'opération a échoué. Il y a une exception qui indique mission déjà validé. Cas 3 : L'opération success. L'ordre mission est validé.	
OBTAINED RESULTS	
Cas1 : L'opération a échoué. Il y a une exception qui indique Permission insuffisante pour valider la mission. Cas2 : L'opération a échoué. Il y a une exception qui indique Mission déjà validé. Cas 3 : L'opération success. L'ordre mission est validé.	

ID	IDENTIFICATION OF COMPONENT
12	Envoyer un email aux secrétaires après confirmation du chef d'équipe.
DESCRIPTION OF THE TEST (granularity, scenario, values, actions)	
Action : Après chef d'équipe valide une mission, le système envoie immédiatement un email qui convient, le pdf d'ordre de sion et pdf de, la validité de mission aux secrétaires.	
Cas 1 : Le chef d'équipe valide une mission.	
EXPECTED RESULTS	
Cas 1 :Un email qui convient,le pdf d'ordre de mission et pdf de,la validité de mission est envoyé aux secrétaires.	
OBTAINED RESULTS	
Cas 1 :Un email qui convient,le pdf d'ordre de mission et pdf de,la validité de mission est envoyé aux secrétaires.	

ID	IDENTIFICATION OF COMPONENT
13	Afficher tous les mission validés.
DESCRIPTION OF THE TEST (granularity, scenario, values, actions)	
Action : Le chef d'équipe se connecte à le systeme. Afficher tous les missions validés.	
Cas 1 : Le chef d'équipe entré correctement le username et le mot de passe. Le chef d'équipe n'a pas validé aucun mission.	
Cas 2 :,Le chef d'équipe entré correctement le username et le mot de passe . Le chef d'équipe a déjà validé des missions.	
EXPECTED RESULTS	
Cas 1 : Le chef d'équipe se connecte à ce système. Aucun missions affiche sur l'écran.	
Cas 2 : Le chef d'équipe se connecte à ce système. Tous les missions validé affiche sur l'écran.	
OBTAINED RESULTS	
Cas 1 : Le chef d'équipe se connecte à ce système. Aucun missions affiche sur l'écran.	
Cas 2 : Le chef d'équipe se connecte à ce système. Tous les missions validé affiche sur l'écran.	

ID	IDENTIFICATION OF COMPONENT
14	Liste toutes les missions en attente de validation.
DESCRIPTION OF THE TEST (granularity, scenario, values, actions)	
Action : Le chef d'équipe se connecte à le systeme. Afficher tous les missions en attente de validation.	
Cas 1 : Le chef d'équipe entré correctement le username et le mot de passe. Aucune mission en attente de validation.	
Cas 2 :Le chef d'équipe entré correctement le username et le mot de passe . Il y a plusieurs missions en attente de validation.	
EXPECTED RESULTS	
Cas 1 : Le chef d'équipe se connecte à ce système. Aucun missionsaffiche sur l'écran.	
Cas 2 : Le chef d'équipe se connecte à ce système. Tous les missions en attente de validation affiche sur l'écran.	
OBTAINED RESULTS	
Cas 1 : Le chef d'équipe se connecte à ce système. Aucun missions affiche sur l'écran.	
Cas 2 : Le chef d'équipe se connecte à ce système. Tous les missions en attente de validation affiche sur l'écran.	

ID	IDENTIFICATION OF COMPONENT
15	Envoyer la modification notifier par d'un OdM au créateur de l'OdM.
DESCRIPTION OF THE TEST (granularity, scenario, values, actions)	
Action : Après le chef d'équipe modifie un ordre de mission. Le système envoie un email au créateur de l'ODM.	
Cas 1 : Quand le chef d'équipe modifie un ordre de mission.	
EXPECTED RESULTS	
Cas 1 : Le créateur de l'ODM recevoir un email notifier.	
OBTAINED RESULTS	
Cas 1 : Le créateur de l'ODM recevoir un email notifier.	



Bibliographie

- (a) [Blake et Merz, 1998] Blake C., Merz C. : UCI Repository of machine learning databases, [www.ics.uci.edu/mlearn/MLRepository.html]. Irvine, University of California, Department of Information and Computer Science, 1998.
- (b) [Inselberg, 1998] Inselberg A. : Visual Data Mining with Parallel Coordinates, Computational Statistics Vol. 13(1), pp.47-63, 1998.
- (c) [Keim et Kriegel, 1994] Keim D.A., Kriegel H.-P. : VisDB : Database Exploration using Multidimensional Visualization, Computer Graphics Applications Journal, pp.40-49, 1994.
- (d) [Brunk et al., 1997] Brunk C., Kelly J. Kohavi R. : MineSet : an integrated system for data mining, International Conference on Knowledge Discovery and Data Mining (KDD'97), AAAI Press, pp 135-138, 1997.
- (e) [Cox et al, 1997] Cox K.C., Eick S.G., Wills G.J., Brachman R.J. : Visual Data Mining : Recognizing Telephone Calling Fraud, Fraud, Data Mining and Knowledge Discovery Vol. 1, pp.225-231, 1997.
- (f) [Ankerst, 2000] Ankerst M. : Visual Data Mining. PhD Thesis, Ludwig Maximilians University of Munich, 2000.
- (g) [Han et Cercone, 2001] Han J., Cercone N. : "Interactive Construction of Decision Trees" in proc. of PAKDD'2001, LNAI 2035, pp.575-580, 2001.
- (h) [Poulet, 2002b] Poulet F. : Cooperation between automatic algorithms, interactive algorithms and visualization tools for visual data mining. In Proc. of Visual Data Mining workshop, PKDD2002, pp. 67-79, 2002.

LIFAT Manager : Un site et des services Web pour la gestion du laboratoire polytech :

Qingling MENG

Encadrement : Jean-Yves RAMEL



En collaboration avec
La grosse compagnie

Objectifs

- Ajouter un module de gestion des missions
- Déployer de la version actuelle à le serveur d'Université de Tours
- Ajouter un module de visualisation des statistiques
- Se connecter via CAS
- Verifier de l'adresse mail



LABORATOIRE D'INFORMATIQUE FONDAMENTALE ET APPLIQUÉE DE TOURS

Mise en œuvre

En cours de développement, nous avons appelé la classe FPDF pour générer des PDF et la classe Email pour envoyer des e-mails. Le framework Cakephp est utilisé partout pour implémenter toutes les fonctions.



Résultats attendus

Le résultat attendu est Les utilisateurs peuvent gérer la mission dans LIFAT Manager et la déployer sur le serveur

The screenshot shows the LIFAT Manager web application. At the top, it says 'LIFAT Manager' and 'Connecté en tant que : prenomUser2 nomUser2 (chef_equipe permanent)'. Below this is a navigation bar with links: 'Mes Missions', 'Mission à Valider', 'Missions Validées', 'Administration', 'Export', 'Fichiers', 'Mon Profil', and 'Deconnexion'. A yellow warning box contains the text: 'Rappel : L'ordre de mission doit parvenir au secrétariat 5 jours ouvrés avant le départ. Après validation par le chef d'équipe, l'ODM sera transmis au secrétariat par mail dont vous recevrez une copie. L'édition et la suppression ne sont possibles que pour un ODM soumis mais pas encore validé.' Below the warning box is a search bar with the placeholder text 'Recherche (Numéro et/ou État du mission):' and a 'Filtrer' button. The main content area is titled 'Mes missions [Nouvelle mission]' and contains a table with the following data:

Id	Motif	Lieu	Date Depart	Date Retour	Etat	Actions
382	motif10	Tours	04/04/2020 03:01	03/10/2020 03:03	soumis	Détails Modifier Supprimer
567	motif10	Tours	04/06/2020 03:04	04/04/2022 03:03	soumis	Détails Modifier Supprimer
568	motif10	Tours	04/06/2020 03:04	04/04/2022 03:03	soumis	Détails Modifier Supprimer
569	motif10	Tours	04/06/2020 03:04	04/04/2022 03:03	soumis	Détails Modifier Supprimer
565	motif2	Paris	03/04/2022 02:01	04/05/2023 04:02	soumis	Détails Modifier Supprimer
570	motif4	Lille	06/03/2019 03:04	05/04/2021 03:01	soumis	Détails Modifier Supprimer
571	motif3	Rennes	04/04/2019 02:02	02/03/2021 02:03	soumis	Détails Modifier Supprimer
592	motif3	Tours	04/05/2020 00:03	04/04/2021 01:03	soumis	Détails Modifier Supprimer
596	motif4	Orléans	03/01/2021 04:02	04/07/2021 03:02	soumis	Détails Modifier Supprimer
424	motif4	Londres	06/05/2020 02:02	02/04/2021 04:03	soumis	Détails Modifier Supprimer
566	motif4	Orléans	03/02/2019 01:01	04/03/2020 03:01	soumis	Détails Modifier Supprimer

LIFAT Manager : Un site et des services Web pour la gestion du laboratoire polytech :

Qingling MENG

Encadrement : Jean-Yves RAMEL

Objectifs

- Ajouter un module de gestion des missions
- Déployer de la version actuelle à le serveur d'Université de Tours
- Ajouter un module de visualisation des statistiques
- Se connecter via CAS
- Vérifier de l'adresse mail



Mise en œuvre

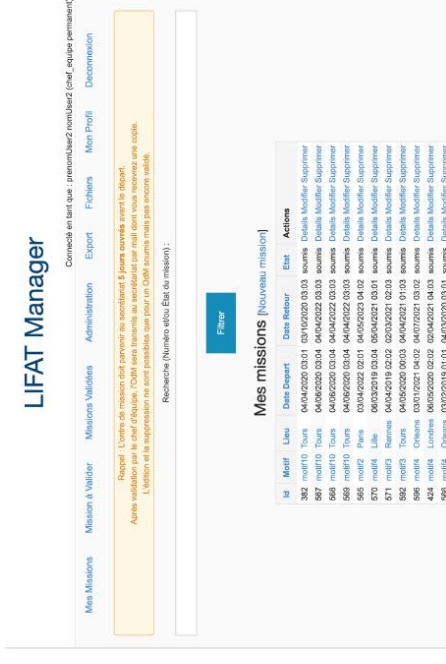
En cours de développement, nous avons appelé la classe FPDF pour générer des PDF et la classe Email pour envoyer des e-mails. Le framework Cakephp est utilisé partout pour implémenter toutes les fonctions.



En collaboration avec La grosse compagnie

Résultats attendus

Le résultat attendu est Les utilisateurs peuvent gérer la mission dans LIFAT Manager et la déployer sur le serveur



LIFAT Manager : Un site et des services Web pour la gestion du laboratoire

Résumé

Le sujet de Projet de Recherche et Développement est " LIFAT Manager : Un site et des services Web pour la gestion du laboratoire ". Ce projet vise à développer les fonctionnalités. L'objectif du projet est d'améliorer un système d'information pour le LIFAT sous forme d'application web. Permettre ce site a le droit de gérer la mission, la publication, et générer un module de fouille visuelle de données . En plus, fournir une deuxième façon de se connecter au site Web via CAS.

Mots-clés

fouille de données, fouille visuelle de données, graphique, techniques orientées pixels, techniques matricielles

Abstract

The subject of Research and Development Project is "LIFAT Manager: A Website and Web Services for Laboratory Management This project aims to develop the functionalities The objective of the project is to improve an information system for LIFAT in the form of web application Allow this site has the right to manage the mission, publication, and generate a visual data search module, and provide a second way to connect to the website via CAS.

Keywords

data mining, visual search, graphic, pixel oriented techniques, Matrix techniques

Entreprise

Polytech



Tuteur entreprise

Jean-Yves RAMEL

Étudiant

Qingling MENG (DI5)

Tuteur académique

Jean-Yves RAMEL