



Rapport de Stage

Développement d'une Application Web de Gestion des Étudiants

Réalisé par :

— KHTEIRA Habib

— BARRY Boubacar Sidy

Table des matières

1	Contexte	3
2	Introduction	4
3	Modélisation de la Base de Données	5
3.1	Explication Générale du Modèle	5
3.2	Entité principale	6
3.3	Informations avant l'inscription	7
3.4	Informations après l'inscription	8
3.4.1	Maquette	9
3.4.2	Note	13
3.5	Modifications Apportées	13
4	Réalisation	15
4.1	Introduction	15
4.2	Fonctionnement du framework Symfony	15
4.2.1	Pratique	16
4.3	Gestion du projet : GitHub	16
5	Fonctionnalités	18
5.1	Gestion des utilisateurs	18
5.2	Gestion des maquettes	19
5.2.1	Insertion des maquettes	19
5.2.2	Préparation des Fichiers XML à Partir des Données Initiales	19
5.2.3	Visualisation des maquettes	21
5.3	Gestion des étudiants	22
5.3.1	Insertion des étudiants	22
5.3.2	Visualisation des étudiants	23
5.3.3	Gestion des notes	24
5.3.4	Insertion des notes	24
5.3.5	Visualisation des notes	24
6	Conclusion	29
7	Prérequis pour utiliser le projet	30
7.1	Récupération du code source depuis GitHub	30
7.2	Installation de PHP 8.3	30
7.3	Installation de Composer	30
7.4	Installation de Symfony	31
7.5	Installation de XAMPP	31

7.6	Installation des dépendances du projet	31
8	Références	32

Chapitre 1

Contexte

Ce rapport présente le travail effectué dans le cadre du développement d'une application web dédiée à la gestion des étudiants inscrits dans les différentes filières du département informatique de l'université d'Angers. Ce projet a été initié par le département informatique afin de simplifier la gestion des étudiants et de centraliser leurs données. Actuellement, les informations des étudiants sont dispersées dans plusieurs applications de la scolarité, et certaines sont parfois supprimées, rendant la gestion complexe et inefficace.

Suite à la modélisation de la base de données centralisée regroupant toutes les informations importantes de chaque étudiant, effectuée l'année dernière, l'objectif principal du projet de cette année est d'améliorer cette base de données et de développer l'application web. Cette application permettra aux enseignants et aux responsables de filière d'accéder et de gérer de manière efficace les différentes informations des étudiants.

Dans ce rapport, nous décrivons les différentes étapes du développement de l'application. Nous présentons également les résultats obtenus jusqu'à présent ainsi que les perspectives pour l'avenir du projet.

Ce projet a été réalisé sous la direction de Touria AIT EL MEKKI et Laurent GARCIA.

Chapitre 2

Introduction

L'objectif principal de ce projet est le développement d'une application web pour la gestion des étudiants inscrits dans les différentes filières du département informatique. Cette application doit permettre aux utilisateurs de communiquer avec une base de données contenant les informations principales de chaque étudiant (son groupe, son bac, ses spécialités, son cursus externe à l'université d'Angers et ses notes). Cette communication peut être séparée en deux parties : l'insertion et la visualisation des données.

Pour l'insertion des données, l'application doit être capable de récupérer les données à partir de fichiers CSV avec des en-têtes spécifiques, de fichiers XML et des e-mails envoyés par la scolarité, au lieu d'une insertion classique par formulaires, qui n'est ni efficace ni nécessaire dans ce cas.

Pour la visualisation, l'application doit permettre de visualiser les données dans des pages séparées dédiées pour chaque type de donnée, tout en assurant la facilité d'accès à ces données pour les utilisateurs et en rendant ces données claires, que ce soit par des diagrammes ou par des tableaux bien organisés.

Pour assurer l'organisation de l'application et la sécurité des données des étudiants, cette application doit également contenir des fonctionnalités de gestion des utilisateurs. Ces fonctionnalités vont permettre aux responsables pédagogiques et administratifs de gérer les comptes des utilisateurs.

Chapitre 3

Modélisation de la Base de Données

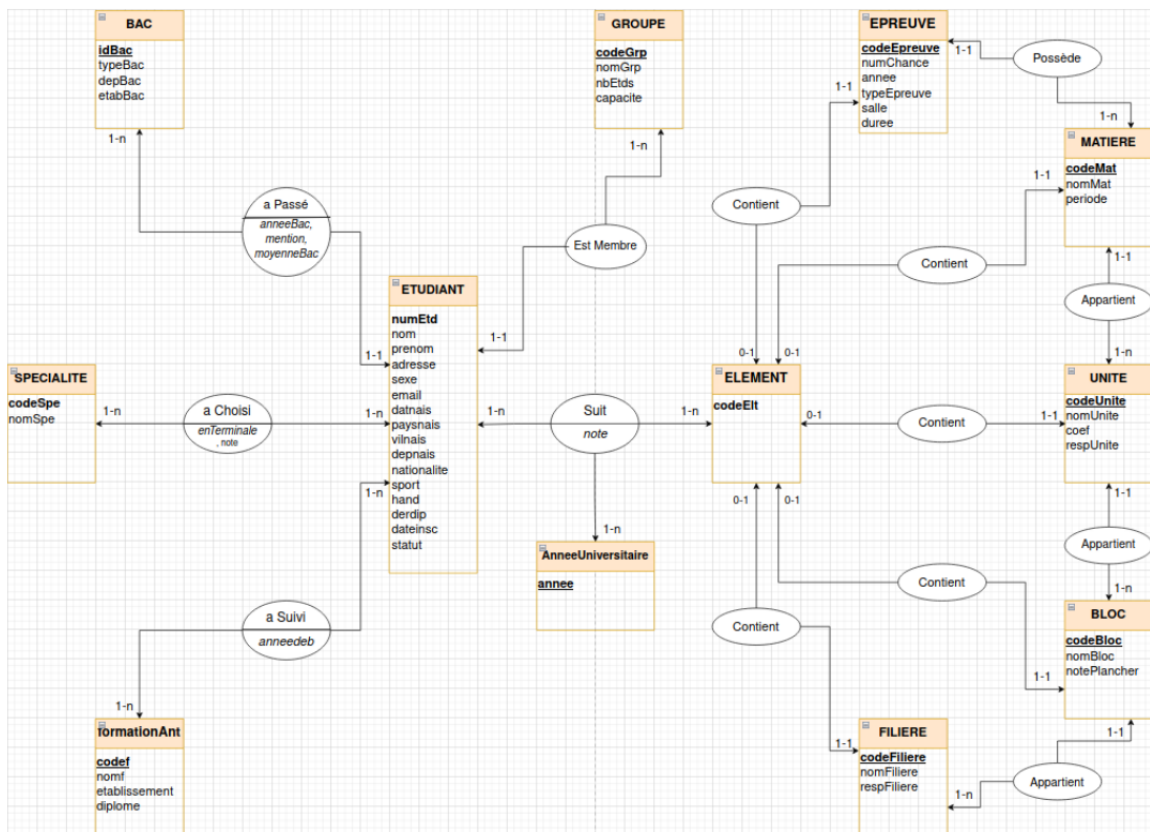


FIGURE 3.1 – Modèle complet

3.1 Explication Générale du Modèle

Pour notre base de données, nous avons repris la modélisation conçue par les étudiants de l'année précédente. Cette base de données permet de gérer efficacement les données variées et complexes liées aux parcours des étudiants. Le modèle permet de suivre les étudiants depuis leur parcours scolaire antérieur jusqu'à leur progression au sein de l'université.

3.2 Entité principale

Les données stockées dans la base de données peuvent être séparées en deux types : les informations de l'étudiant avant son inscription à l'Université d'Angers (UA) et les données après son inscription. Dans le modèle, les tables de ces données sont liées à une entité principale, **ÉTUDIANT**, représentée par un identifiant unique numEtd, qui contient les informations personnelles de l'étudiant.

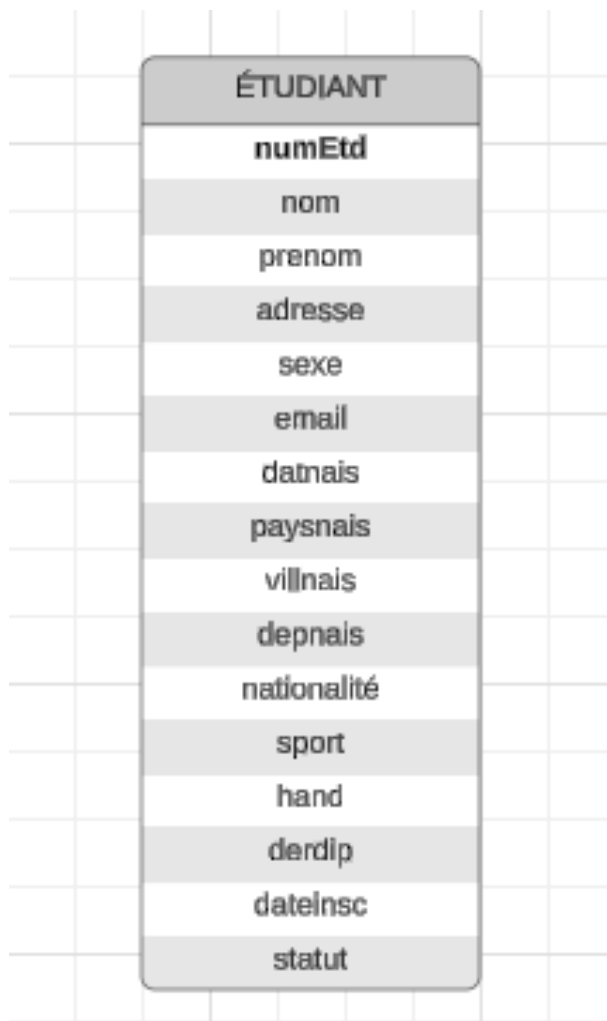


FIGURE 3.2 – Entité principale

3.3 Informations avant l'inscription

La modélisation de la base de données permet de stocker toutes les informations pédagogiques d'un étudiant avant son inscription, incluant les différentes options choisies par l'étudiant pendant ses études au lycée, les informations sur son baccalauréat, et les formations suivies après le bac.

- **BAC** : Représente les différents baccalauréats obtenus par les étudiants. La relation entre **ÉTUDIANT** et **BAC** permet de stocker l'année d'obtention, la mention et la moyenne du baccalauréat.

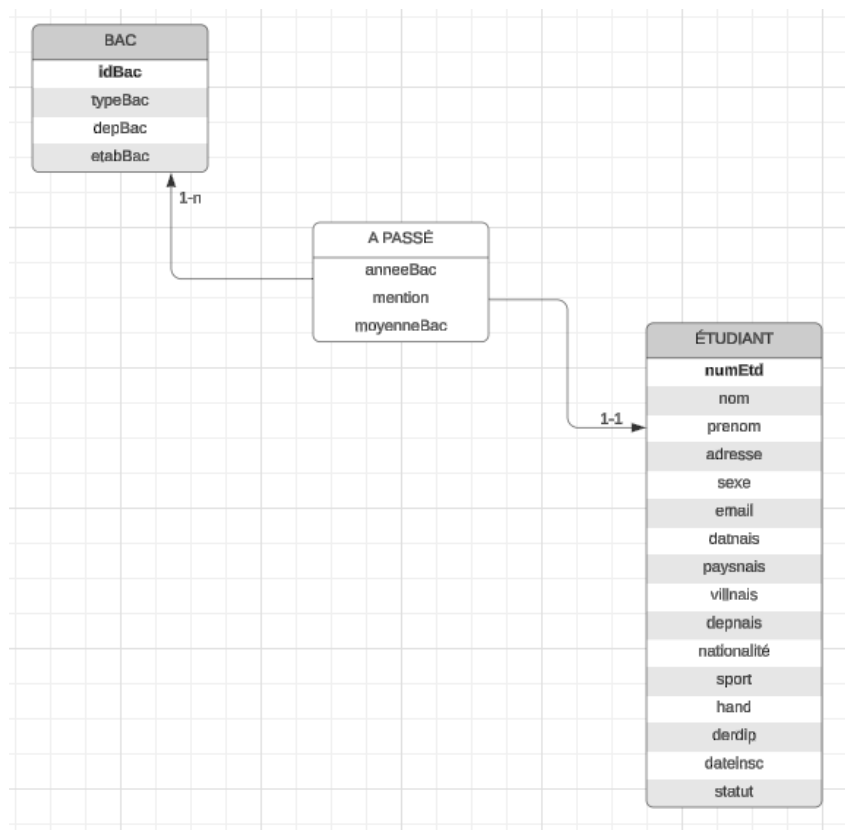


FIGURE 3.3 – BAC

- **SPECIALITE** : Représente les options choisies par les étudiants au lycée. La relation entre **ÉTUDIANT** et **SPÉCIALITÉ** permet de déterminer si une spécialité a été maintenue en terminale.

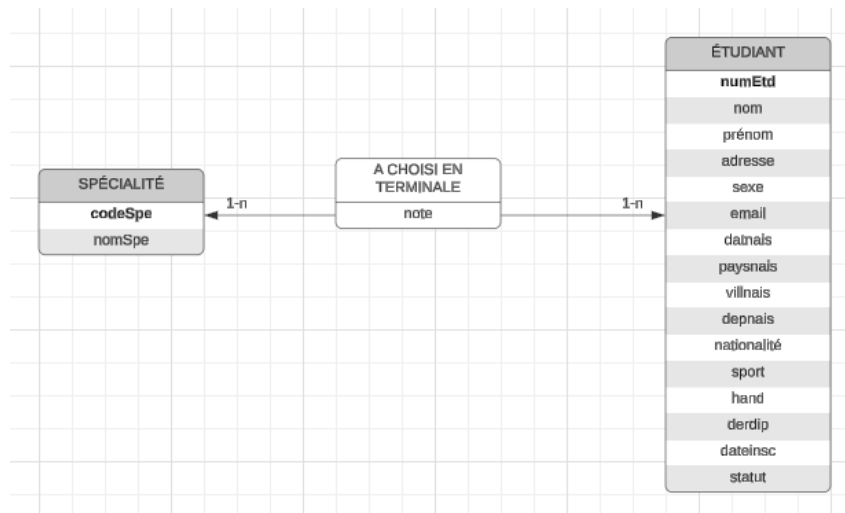


FIGURE 3.4 – Spécialité

- **FORMATIONANT** : Représente les formations suivies par étudiants parès son bac avant son inscription. La relation entre **ÉTUDIANT** et **FORMATIONANT** permet de suivre l'année de début des différentes formations antérieures.

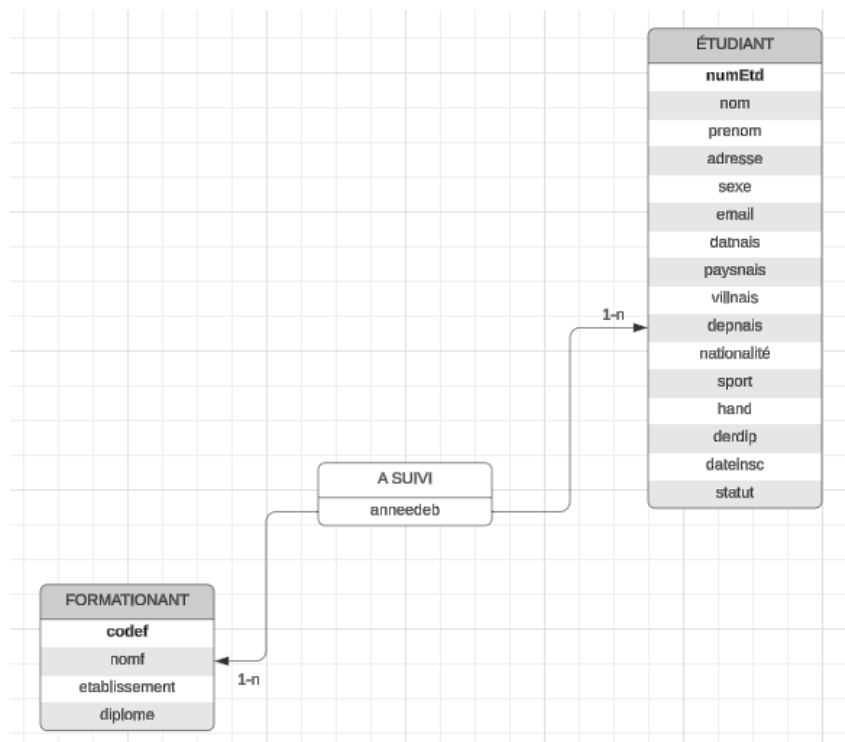


FIGURE 3.5 – FormationAnt

3.4 Informations après l'inscription

Après l'inscription, la base de données permet de stocker le groupe de l'étudiant et ses notes pour chaque année. Pour ce faire, il est nécessaire d'avoir des entités pour stocker les données des filières.

3.4.1 Maquette

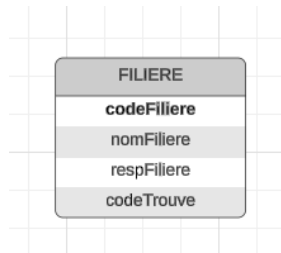
Les maquettes sont des documents descriptifs qui détaillent le contenu et l'organisation des enseignements des formations universitaire.

Licence 3 Informatique

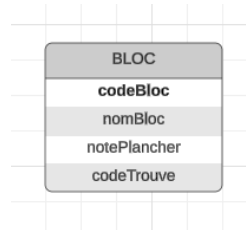
Période	Intitulés	Volumes horaires				ECTS	Coef	Chance	
		CM	TD	TP	Total			Chance 1 Assidus	Chance 2 Assidus
ANGLAIS									
Anglais 1									
P11	Anglais			8,0	8,0	2	2	CC 100% -1h20	CT 100% - 1h20
P12	Anglais			8,0	8,0				
Anglais 2									
P13	Anglais			8,0	8,0	2	2	CC 100% -1h20	CT 100% - 1h20
P14	Anglais			8,0	8,0				
3PE									
	3PE			20,0	20,0	0	0	Suivi individuel. Pas d'évaluation.	
STAGE choix 1 parmi 2									
P15	Entreprise		1,0			6	6	note	Pas de 2 nd chance
P15	Laboratoire		2,0					note	
CHOIX DE 2 UE parmi les 4 suivantes									
Note plancher 1									
P14	Synthèse d'images	6,4		17,6	24,0	2	2	TP 50% - 1h Projet 50%	TP 50% - 1h20 Projet 50% ●
P14	Initiation aux systèmes intelligents	6,9	3,4	13,7	24,0	2	2	CT 33% - 2h Projet 67%	CT 33% - 2h Projet 67% ●
P14	Développement d'interface graphique	4,0		20,0	24,0	2	2	TP 100% - 2h	TP 100% - 2h
P14	Introduction au machine learning avec Python	7,0	4,6	12,4	24,0	2	2	100% Projet	CT 67 % - 1h30 Projet 33% ●
Fondements									
Note plancher 5									
P11	Fondements de l'informatique 4	10,0	10,0		20,0	3	3	CT 100% - 2h	CT 100% - 2h
Théorie des langages 2 et compilation									
P12	Théorie des langages 2 et compilation	10,0	8,7	6,0	24,0	6	6	CC 33% - 2h	CT 100% - 2h
P13	Théorie des langages 2 et compilation	10,0	8,7	6,0	24,0			CC 67% - 2h	
Architecture et ordinateurs									
P13	Architecture des ordinateurs	10,0	5,8		15,8	6	6	CC 33% - 2h	CT 100% - 1h
P14	Architecture des ordinateurs	10,0	5,8	4,2	20,0			CC 67% - 2h	
Algorithmique et programmation									
Note plancher 5									
Programmation logique									
P13	Programmation logique	12,0		12,0	24,0	3	3	CT 67% - 1h Projet 33%	CT 67% - 1h Projet 33% ●
Programmation fonctionnelle									
P11	Programmation fonctionnelle	10,0	6,0	8,0	24,0	3	3	CT 67% - 1h Projet 33%	CT 100% - 1h ●
Algorithmique des graphes									
P14	Algorithmique des graphes	12,0	4,0	8,0	24,0	3	3	CT 67% - 2h Projet 33%	CT 67% - 2h Projet 33% ●
Programmation orientée objet 2									
P12	Programmation orientée objet 2	10,0		22,0	32,0	7	7	TP 40% - 2h	CT 100% - 2h
P13	Programmation orientée objet 2	10,0		22,0	32,0			TP 60% - 2h	
Technologie de l'informatique									
Note plancher 5									
Traitements de données									
P11	Traitement de données	6,7		13,3	20,0	3	3	CT 67% - 2h Projet 33%	CT 67% - 2h Projet 33% ●
Bases de données 3									
P11	Bases de données 3	4,0		12,0	16,0	6	6	TP 40% - 1h30	CT 100% - 1h30
P12	Bases de données 3	8,0	12,0	4,0	24,0			CC 60% - 1h30	
Développement web 3									
P11	Développement web 3	5,3		10,7	16,0	6	6	TP 40% - 1h30	TP 100% - 1h30
P12	Développement web 3	8,0		16,0	24,0			TP 60% - 1h30	
					499,9	60	59,9		

FIGURE 3.6 – Maquette

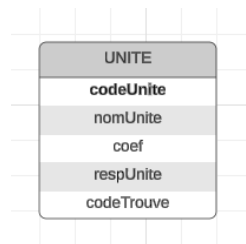
- **Entités** : Les entités de la maquette sont les suivantes : FILIERE, BLOC, UNITE, MATIERE et EPREUVE.
- **FILIERE** : Représente les filières suivies par les étudiants.



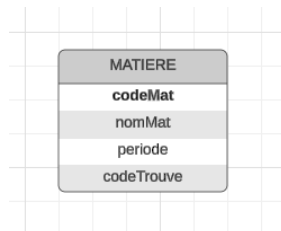
- **BLOC** : Représente un ensemble d'unités de formation. Chaque bloc possède une note plancher à dépasser pour valider l'année.



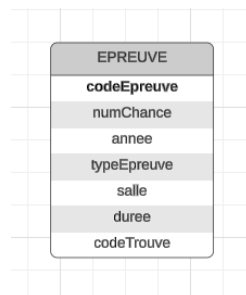
- **UNITE** : Représente les unités de formation regroupant différentes matières.



- **MATIERE** : Représente les cours suivis par les étudiants au cours d'une période.



- **EPREUVE** : Représente les évaluations des étudiants pour une matière donnée.



- **Relations** : Les relations entre les entités sont représentées de la manière suivante :
Une filière est un ensemble de blocs, un bloc est un ensemble d'unités, une unité est un ensemble de matières et une matière possède des épreuves.

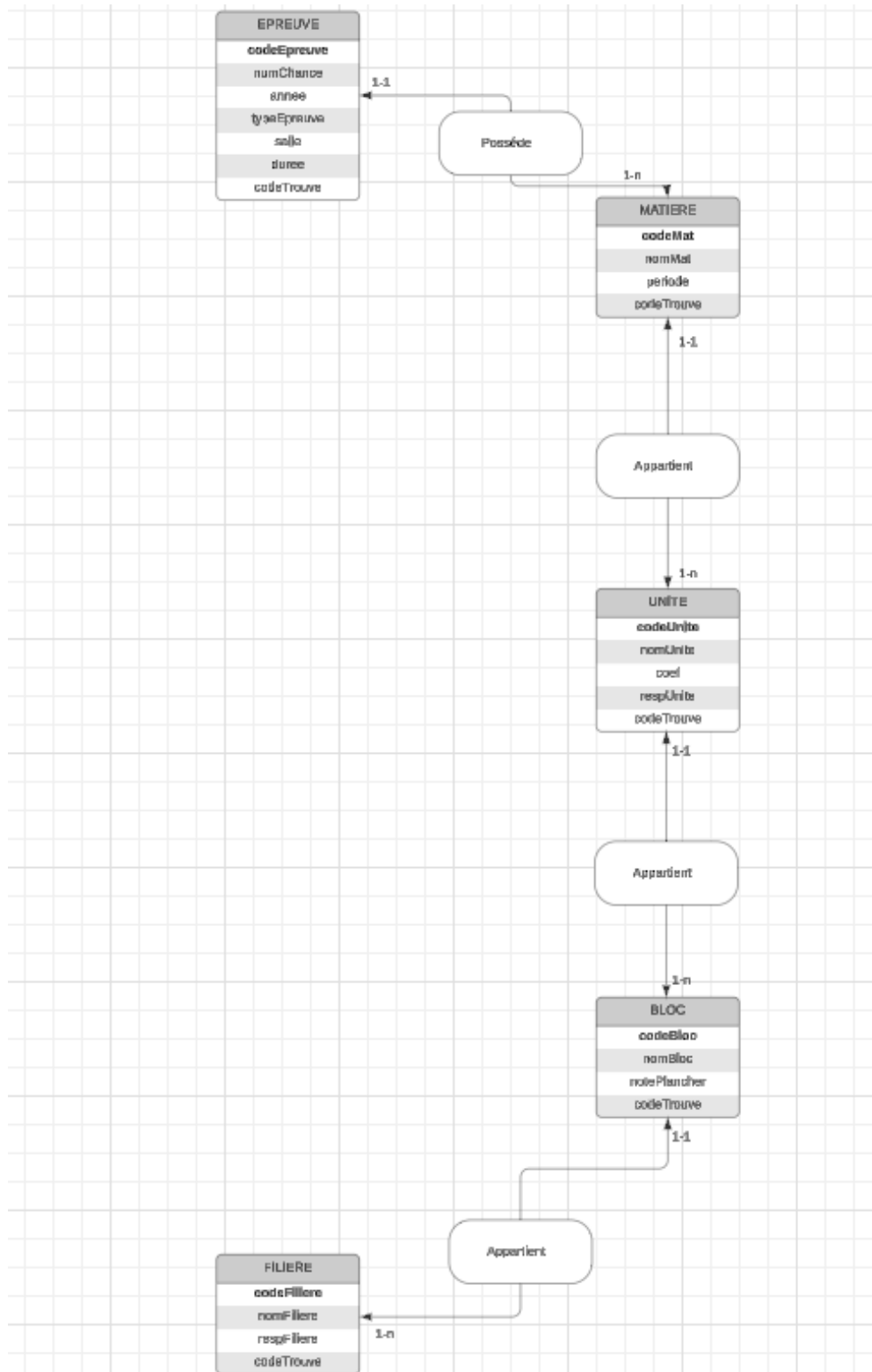


FIGURE 3.7 – Relations

- **Élément** : L'entité **ELEMENT** est une contrainte du système d'information de

l'université d'Angers utilisée pour représenter les différentes composantes académiques telles que les filières, les blocs, les unités, les matières et les épreuves.

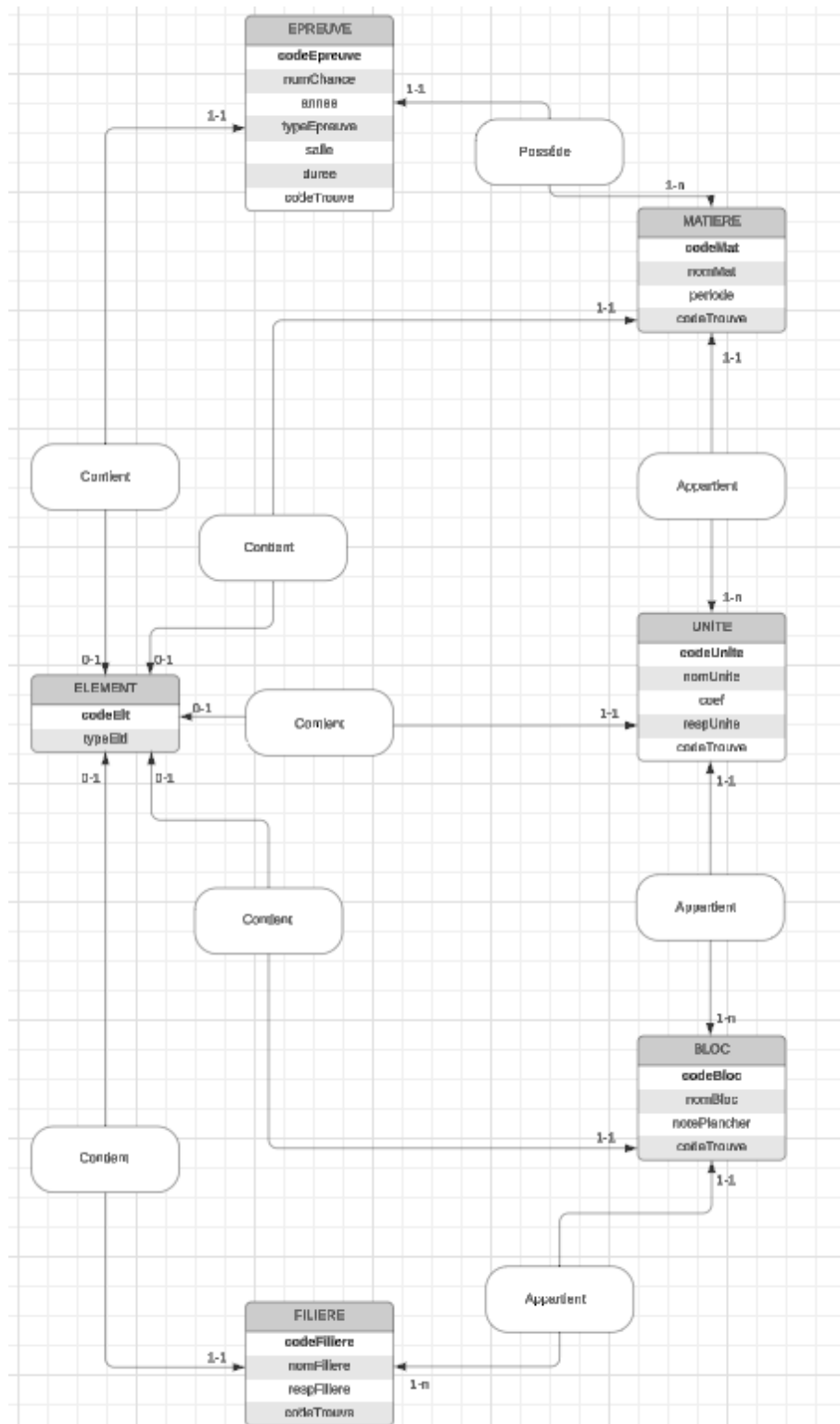


FIGURE 3.8 – ELEMENT

3.4.2 Note

Pour stocker les notes, le modèle contient une association entre les entités **ÉTUDIANT**, **AnneeUniversitaire** et **ELEMENT**. Cette association permet de stocker la note obtenue par un étudiant dans un élément avec l'année d'obtention.

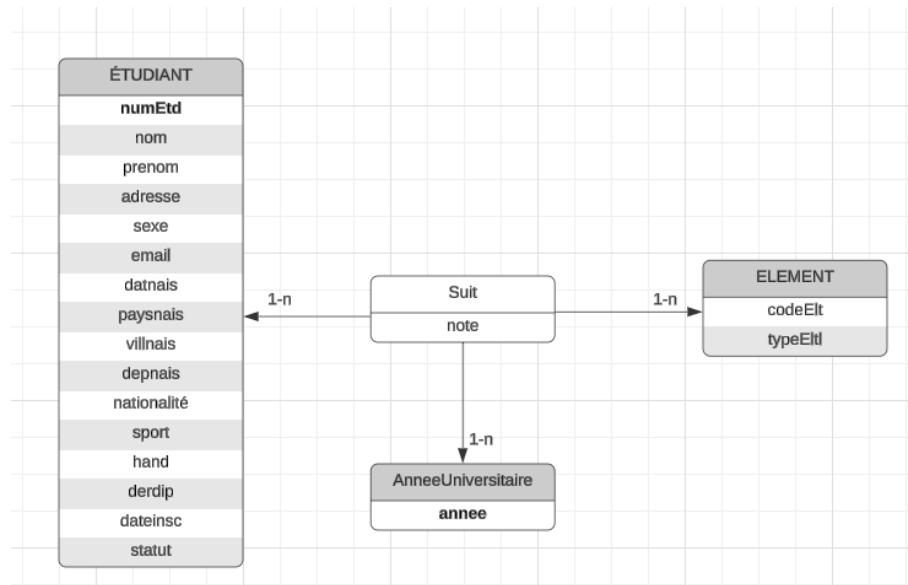


FIGURE 3.9 – Note

3.5 Modifications Apportées

Lors du développement des fonctionnalités de notre application, nous avons apporté les modifications suivantes au modèle :

- **Ajout de l'attribut typeElt à l'entité ELEMENT :**
 - Cet attribut permet de stocker le type d'élément, facilitant ainsi la récupération des notes pour un type spécifique d'élément.
- **Ajout de l'attribut codeTrouve dans les tables FILIERE, BLOC, UNITE, MATIERE, et EPREUVE :**
 - Cet attribut permet de vérifier si le code de l'un de ces éléments est bien trouvé lors de l'insertion des maquettes, garantissant l'accès aux éléments dont le code n'est pas trouvé. Dans ce cas, ces éléments seront envoyés à l'utilisateur pour qu'il puisse insérer leur code manuellement. Ainsi, le processus d'insertion des maquettes assure que chaque élément est correctement codé avant de finaliser l'intégration, évitant ainsi les erreurs et les incohérences dans les données.
- **Ajout d'une association entre les entités ETUDIANT, FILIERE, et ANNEEUNIVERSITAIRE :**
 - Cette association permet d'établir une relation directe entre une filière et ses étudiants pour une année universitaire spécifique, rendant l'accès aux étudiants d'une filière plus facile et plus rapide.

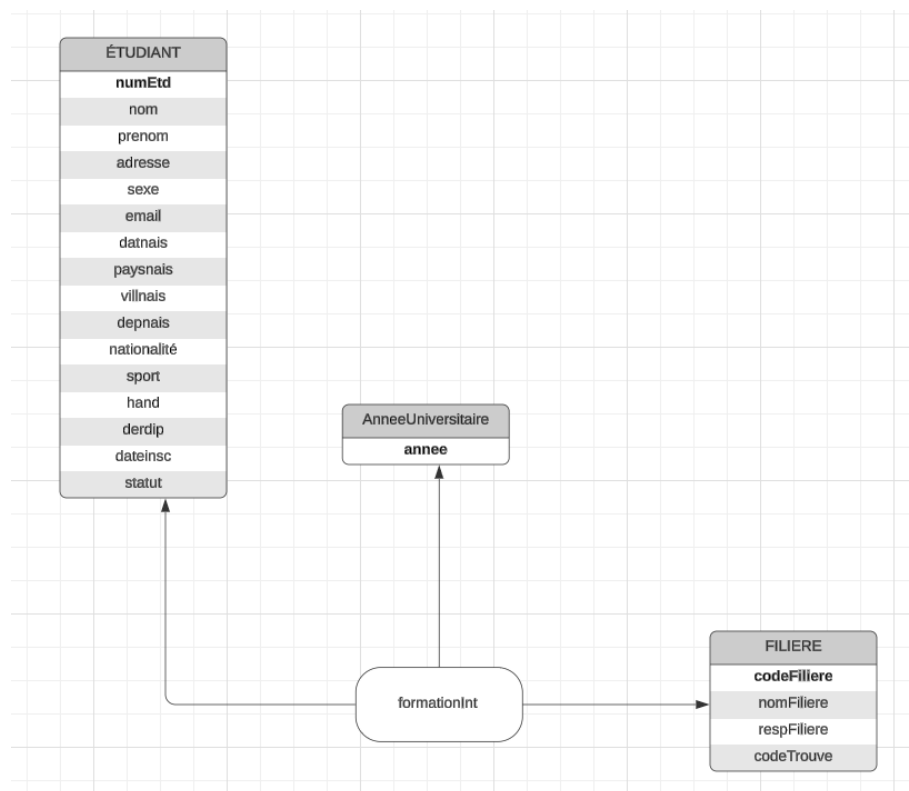


FIGURE 3.10 – FormationInt

Chapitre 4

Réalisation

4.1 Introduction

Pour mener à bien les différentes phases du développement de l'application, dont la création des tables en bases de données et l'implémentation des différentes fonctionnalités, nous avons eu recours à des outils qui nous ont aidés à atteindre nos objectifs et à améliorer notre productivité. Cet ensemble d'outils comprend des logiciels, des services en ligne et des frameworks qui nous ont permis tout au long du projet de garder un bon rythme et de résoudre efficacement les problèmes rencontrés.

Nous pouvons citer entre autres :

- Symfony comme framework pour la création des entités, les migrations, la création des tables en bases de données, etc.
- MySQL comme système de gestion de bases de données relationnelles.
- GitHub comme outil de versioning et de suivi des changements de code.

Dans la suite de ce chapitre, nous allons décrire ces outils, comment ils ont été utilisés dans notre projet, et nous discuterons de leur impact sur notre productivité et sur les résultats finaux de notre projet.

4.2 Fonctionnement du framework Symfony

Symfony est un framework PHP moderne qui facilite le développement d'applications web en suivant l'architecture MVC (Modèle-Vue-Contrôleur). Cette architecture sépare l'application en trois couches distinctes :

- **Modèle (Model)** : Gère les données et les règles de gestion. Dans Symfony, les modèles sont souvent représentés par des entités Doctrine, qui permettent de manipuler les données de la base de données comme des objets PHP.
- **Vue (View)** : Responsable de la présentation des données. Symfony utilise le moteur de template Twig pour générer des vues HTML à partir des données fournies par les contrôleurs.
- **Contrôleur (Controller)** : Traite les requêtes HTTP, exécute les actions appropriées et renvoie les réponses HTTP. Les contrôleurs orchestrent les interactions entre le modèle et la vue.

Doctrine ORM : Doctrine est l'ORM (Object-Relational Mapping) utilisé par Symfony pour gérer la persistance des données. Il permet de définir des entités PHP qui sont automatiquement mappées à des tables de base de données. Doctrine facilite les opéra-

tions CRUD (Create, Read, Update, Delete) en offrant une API intuitive pour interagir avec les données.

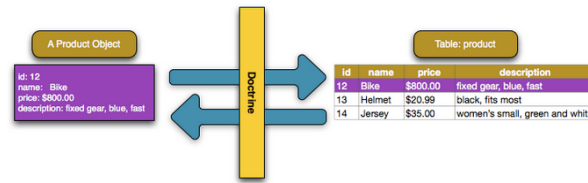


FIGURE 4.1 – Doctrine

Twig : Twig est le moteur de template de Symfony, utilisé pour générer des vues. Il offre une syntaxe claire et concise pour écrire des templates HTML. Twig permet de séparer la logique de présentation du reste du code, améliore la lisibilité et la maintenabilité des vues, et inclut des fonctionnalités avancées comme l'héritage de templates, les filtres et les extensions.

Routing : Le système de routage de Symfony associe les URL des requêtes entrantes aux actions des contrôleurs. Les routes sont définies dans des fichiers de configuration et permettent de gérer facilement les différentes URL de l'application, en spécifiant le contrôleur et la méthode à appeler pour chaque route.

Comme vous le voyez, nous avons besoin d'un environnement de développement local, d'où l'utilisation de XAMPP.

XAMPP : XAMPP est un environnement de développement local qui intègre Apache, MySQL, PHP et Perl. Dans le cadre de Symfony, XAMPP fournit le serveur web Apache pour servir les pages web et MySQL pour gérer la base de données. XAMPP facilite la configuration de l'environnement de développement en fournissant tous les outils nécessaires pour exécuter une application Symfony localement.

MySQL : MySQL est le système de gestion de base de données utilisé avec Symfony pour stocker et gérer les données de l'application. En utilisant Doctrine ORM, Symfony interagit avec MySQL pour effectuer des opérations sur les données de manière efficace et sécurisée. MySQL assure la fiabilité, l'intégrité et la performance des opérations de gestion des données.

4.2.1 Pratique

Voici quelques commandes utiles qui permettent la création de la base de données, des tables, ainsi que des relations.

Action	Commande
Création d'une entité	<code>symfony console make:entity</code>
Migration	<code>symfony console make:migration</code>
Exécuter la migration	<code>symfony console doctrine:migrations:migrate</code>

4.3 Gestion du projet : GitHub

Dans le contexte de notre projet, nous avons utilisé GitHub comme outil de contrôle de version pour suivre les changements de code et permettre la collaboration entre les

membres de l'équipe. Nous avons pu travailler sur différentes fonctionnalités du projet simultanément et fusionner nos modifications de code de manière transparente grâce aux fonctionnalités de contrôle de version de GitHub. Tout cela dans un IDE.

IDE (Environnement de développement intégré) : Pour la réalisation de notre projet, nous avons utilisé Visual Studio Code (VS Code) comme IDE. Un IDE facilite le développement en intégrant des outils tels que l'éditeur de texte avec coloration syntaxique, le débogueur et le gestionnaire de versions. VS Code, en particulier, est apprécié pour sa légèreté, son extensibilité et ses nombreuses extensions qui augmentent la productivité des développeurs en simplifiant la rédaction, le test et le débogage du code.

Chapitre 5

Fonctionnalités

Dans cette section, nous allons présenter les différentes fonctionnalités de l'application qui permettent de réaliser une gestion complète des étudiants inscrits dans les filières du département informatique. Cette application offre une interface conviviale permettant aux enseignants et aux administrateurs d'accéder aux informations essentielles et de réaliser diverses opérations.

5.1 Gestion des utilisateurs

La sécurité a été une priorité dans la conception de notre application. Un système d'authentification sécurisé garantit que seules les enseignants et les administrateurs du département informatique peuvent accéder aux fonctionnalités de l'application. Certaines fonctionnalités sont réservées uniquement aux administrateurs.

Pour ce faire, nous avons créé une table `user` dans la base de données, qui contient les attributs `nom`, `prénom`, `email`, `roles`, `password` et un attribut supplémentaire `validation` qui prend des valeurs booléennes (`true` ou `false`).

Lors de la conception de l'application, Madame AIT EL MEKKI et Monsieur Garcia ont été définis comme administrateurs. Nous avons également mis en place deux formulaires : un pour l'inscription des nouveaux utilisateurs et un autre pour l'authentification des utilisateurs existants.

Un utilisateur souhaitant s'inscrire doit remplir le formulaire d'inscription et attendre la validation par un administrateur. L'administrateur peut valider ou refuser une inscription, définir de nouveaux administrateurs ou supprimer un utilisateur existant.

Un utilisateur nouvellement inscrit aura une valeur `validation` à `false` dans la table `user`. Une fois son inscription validée par un administrateur, cette valeur sera mise à `true`, permettant à l'utilisateur de se connecter et d'accéder aux différentes ressources de l'application.

La gestion des utilisateurs n'est accessible qu'aux administrateurs. Une page dédiée leur permet de valider ou rejeter les inscriptions, de changer le rôle des utilisateurs et de supprimer des utilisateurs.



[Étudiants](#)

[Maquettes](#)

[Insérer des étudiants](#)

[Explorer](#)

Nouveaux utilisateurs en attente de validation

Prénom	Nom	user@demo.fr	refus	Confirmation
--------	-----	--------------	-------	--------------

Utilisateurs

BARRY	Boubacar	bouba@demo.fr	Supprimer	Administrateur	Modification_rôle
KHTEIRA	Habib	habib@demo.fr	Supprimer	Administrateur	Modification_rôle

FIGURE 5.1 – Page de la gestion des utilisateurs

5.2 Gestion des maquettes

La gestion des maquettes est une fonctionnalité clé de notre application, permettant aux utilisateurs d’organiser et de gérer les différents éléments de la maquette pédagogique.

5.2.1 Insertion des maquettes

L’insertion des maquettes se déroule en trois phases :

1. **Phase d’insertion des codes** : Un formulaire permet de récupérer un fichier XML contenant le nom, la nature et le code des éléments. Les données sont ensuite stockées dans la table `codes`.
2. **Phase d’insertion des informations complètes** : L’utilisateur importe un fichier XML contenant les informations complètes des différents éléments de la maquette. Le système récupère les codes des éléments à partir de la table `codes`.
3. **Phase des codes non trouvés** : Si le système ne trouve pas les codes de certains éléments, l’utilisateur est dirigé vers une page avec un formulaire pour remplir les codes manquants.

5.2.2 Préparation des Fichiers XML à Partir des Données Initiales

Pour préparer les fichiers XML à partir des données disponibles, on suivra un processus méthodique pour les codes et les informations complètes de la maquette.

- **Fichier des Codes des** : Pour récupérer les codes des éléments, on utilisera initialement un fichier PDF. En suite pour le rendre exploitable dans notre système, on le convertira en format XLSX (on peut utiliser des outils en lignes telles ilovpdf). Le fichier XLSX ainsi obtenu permettra de structurer les données sous forme tabulaire, facilitant leur traitement ultérieur. Ensuite, à l’aide d’un script Python dédié, on transformera ce fichier XLSX en format XML.

en résumé :

TIN2LET : Liste étape L2 Informatique

Libellé	Code	Nature	ECTS /Chance	Volume	Code liste
BLOC Algorithmique et programmation	TIN2B1	BLCT	16	H	TIN2L1
BLOC Fondements et théorie de l'informatique	TIN2B2	BLCT	14	H	TIN2L2
BLOC Technologie de l'informatique	TIN2B3	BLCT	16	H	TIN2L3
Transversaux et indépendants Informatique	TIN2TI	BLOC	14		TIN2LT1

en détail :

TIN2LET : Liste étape L2 Informatique

Libellé	Code	Nature	ECTS /Chance	Volume	Code liste
BLOC Algorithmique et programmation	TIN2B1	BLCT	16	H	TIN2L1
Algorithmique 3 - P6-P7	TMX2206	UE	8	H	TMX2206
1ère chance	TMX22061	SECT			
CC algorithmique 3 - P6	TMX2206C1	CC	1		
CC algorithmique 3 - P7	TMX2206C2	CC	1		
Choix décision : choix de 1 parmi 2	TMX22068	CHOI			TMX2206X
2ème chance	TMX22062	SECT			
CT algorithmique 3 - P6-P7	TMX2206E	ECR	1		
Je ne me présente pas	TMX22069	SECT			
Programmation orientée objet 1 - P8-P9-P10	TMX2504	UE	8	H	TMX2504
1ère chance	TMX25041	SECT			
CC programmation orientée objet 1 - P8	TMX2504C1	CC	1		
CC programmation orientée objet 1 - P9	TMX2504C2	CC	1		
TP programmation orientée objet 1 - P10	TMX2504T	TP	1		

FIGURE 5.2 – Exemple du fichier PDF des codes

- **Fichier des Informations Complètes de la Maquette :** Pour récupérer les informations complètes des différents éléments de la maquette pédagogique, on utilisera un fichier initial au format XLSX. Ce fichier XLSX contient toutes les données nécessaires sous forme tabulaire. Comme pour le fichier des codes, on utilisera également le même script Python pour transformer ce fichier XLSX en un fichier XML.

	A	B	C	D	E	F	G
4	CFVU du :						
5	Effectif prévisionnel : Nb d'inscrits N-1 ou estimation pour les nouvelles formations						
6	V.D.I. : code déterminé par le pôle modélisation						
7	V.E.T. : code déterminé par le pôle modélisation						
8							
9	Intitulés des blocs et des U.E. Ne doit pas dépasser 60 caractères (contrainte Apogée) - utiliser des abréviations si nécessaire	Nature voir liste dans onglet suivant	CODE CNU (sur les matières uniquement) voir liste dans l'onglet suivant	Participe aux charges O/N	Mutualisé <i>(préciser en remarque avec quelle(s) formations)</i> O/N	Nb	
10						CM	TD
11						Nombre d'Heures	Nombre d'Heures
12	SOCLE			N			
13	UE ou bloc [intitulé]			N		Total (max) par étudiant	Total (max) par étudiant
14	EC/matière			O		15,0	10,0
15	Exemple de Matière évaluée par la présence			O		15,0	
16	Exemple de Matière évaluée par un dossier			O		10,0	5,0
17	Exemple de Matière évaluée par CC + CT (coef identiques)			O		10,0	5,0
18	Exemple de Matière évaluée par CC + CT (coef différents)			O		10,0	5,0
19	UE bloc			N			
20	EC/matière			O			
21	EC/matière			O			
22	Choix de 1 EC parmi les 3 suivants : Préciser le nombre de choix à faire parmi le nombre de choix proposés			N		Total (max) par étudiant	Total (max) par étudiant
23	EC/matière			O			
24	EC/matière			O			
25	EC/matière			O			
26			Total heures :			Total (max) par étudiant	Total (max) par étudiant
27							
28							
29							
30							
31							
32							
33							
34							

FIGURE 5.3 – Exemple du fichier XLSX du maquette

5.2.3 Visualisation des maquettes

Nous avons préparé deux pages pour accéder à toutes les maquettes de la base de données ou à la maquette d'une filière spécifique. Chaque maquette est présentée dans un tableau bien organisé affichant les informations nécessaires de chaque élément.



ÉTUDIANTS
Maquettes

Insérer des étudiants

Explorer

Filière: L2 Informatique

Algorithmique et programmation

Algorithmique 3 (Coefficient: 8)

P6	Algorithmique 3	CC 50% 1h30	CT 100% 1h30
P7	Algorithmique 3	CC 50% 1h30	

Programmation orientée objet 1 (Coefficient: 8)

P8	Programmation orientée objet 1	CC 20% 1h	CT 60% 2h
P9	Programmation orientée objet 1	CC 40% 1h	
P10	Programmation orientée objet 1	TP 40% 2h	

Fondements et théorie de l'informatique

Fondements de l'informatique 2 (Coefficient: 6)

P6	Fondements de l'informatique 2	CC 30% 1h30	CT 100% 1h30
P7	Fondements de l'informatique 2	CC 70% 1h30	

Théorie des langages 1 (Coefficient: 6)

P8	Théorie des langages 1	CC 50% 1h30	CT 100% 1h30
P9	Théorie des langages 1	CC 50% 1h30	

FIGURE 5.4 – Page des maquettes

5.3 Gestion des étudiants

La gestion des étudiants est centrale dans notre application, offrant des fonctionnalités pour visualiser les étudiants existants et en ajouter de nouveaux.

5.3.1 Insertion des étudiants

Pour l'insertion des étudiants en début ou en cours de cursus, nous avons mis en place une méthode d'importation de fichier CSV. Cette méthode permet d'ajouter rapidement un grand nombre d'étudiants à partir d'une source externe, en respectant un format de fichier bien défini.

Un formulaire d'importation dédié permet de sélectionner un fichier CSV sur l'ordinateur. Une fois le fichier sélectionné, notre système lit le fichier et extrait les données des étudiants pour les insérer de la manière suivante :

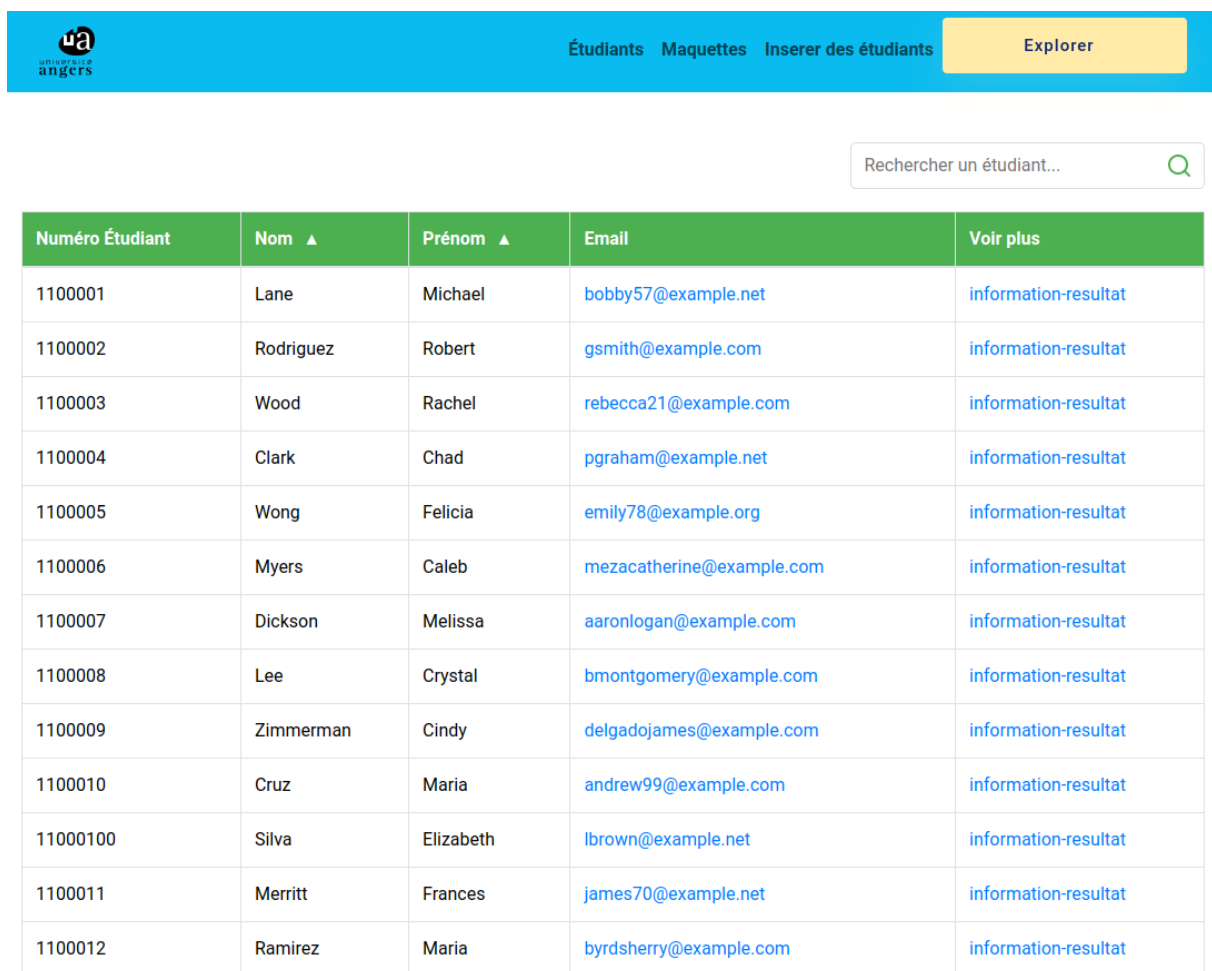
- Insertion des informations personnelles dans l'entité **ÉTUDIANT**.
- Insertion des informations sur le baccalauréat, les spécialités et les formations antérieures. Si le système détecte un baccalauréat, une spécialité ou une formation non enregistrée dans la base de données, il crée et insère l'entité correspondante.
- Création d'une relation entre l'étudiant et sa filière.

Une fois l'importation terminée, les nouveaux étudiants sont ajoutés à la base de données, prêts à être consultés et gérés.

5.3.2 Visualisation des étudiants

Pour la visualisation des étudiants, nous avons conçu une interface utilisateur conviviale et facile à naviguer. Tous les étudiants sont affichés dans un tableau avec des en-têtes clairement définis, notamment le numéro d'étudiant, le nom, le prénom et l'adresse électronique. Nous avons également ajouté un lien **info-résultat** pour chaque étudiant, ce qui permet de visualiser les informations personnelles de l'étudiant ainsi que ses résultats tout au long du cursus.

Pour faciliter la recherche d'étudiants, nous avons mis en place une barre de recherche avancée. Cette barre de recherche permet de filtrer les étudiants en temps réel, au fur et à mesure que l'utilisateur tape des caractères dans la barre. Les résultats de la recherche sont affichés dans le tableau, ce qui permet à l'utilisateur de trouver rapidement et facilement l'étudiant qu'il recherche.



Numéro Étudiant	Nom ▲	Prénom ▲	Email	Voir plus
1100001	Lane	Michael	bobby57@example.net	information-resultat
1100002	Rodriguez	Robert	gsmith@example.com	information-resultat
1100003	Wood	Rachel	rebecca21@example.com	information-resultat
1100004	Clark	Chad	pgraham@example.net	information-resultat
1100005	Wong	Felicia	emily78@example.org	information-resultat
1100006	Myers	Caleb	mezacatherine@example.com	information-resultat
1100007	Dickson	Melissa	aaronlogan@example.com	information-resultat
1100008	Lee	Crystal	bmontgomery@example.com	information-resultat
1100009	Zimmerman	Cindy	delgadojames@example.com	information-resultat
1100010	Cruz	Maria	andrew99@example.com	information-resultat
11000100	Silva	Elizabeth	lbrown@example.net	information-resultat
1100011	Merritt	Frances	james70@example.net	information-resultat
1100012	Ramirez	Maria	byrdsherry@example.com	information-resultat

FIGURE 5.5 – Page de visualisation des étudiants

5.3.3 Gestion des notes

5.3.4 Insertion des notes

Pour l'insertion des notes, nous avons prévu deux fonctionnalités pour importer les notes à partir du contenu d'un email ou d'un fichier CSV. Le système d'insertion et de calcul des notes reste le même, avec une différence au niveau des formulaires et de l'extraction des données. Les informations nécessaires, telles que le code de l'épreuve, l'année, la session, les numéros des étudiants et leurs notes, sont extraites des données reçues. Les notes des étudiants pour cette épreuve sont ensuite insérées et les notes des autres éléments liés à cette épreuve sont mises à jour en fonction de la session (1ère ou 2ème chance).

Lorsqu'il s'agit de la 1ère chance, le mécanisme suivant est appliqué : on trouve la matière associée à l'épreuve insérée et ensuite on calcule la note de cette matière à partir des notes et des pourcentages des épreuves de cette matière qui ont déjà une note insérée. La note finale de la matière est ensuite mise à jour. Ensuite, on fait la même chose pour l'unité de cette matière en calculant sa note à partir des notes et des sommes des pourcentages des épreuves de ses matières, puis on met à jour la note finale de l'unité. De même, on applique ce calcul pour le bloc et la filière, mettant à jour les notes finales en conséquence.

Pour la 2ème chance, on trouve la matière de cette épreuve puis on compare la nouvelle note avec celle de l'unité correspondante. Si la nouvelle note est plus petite, on insère simplement la note de l'épreuve sans changer les notes des autres éléments. Sinon, on remplace la note de l'unité par la note insérée et on recalculé la note du bloc et de la filière de cette unité comme pour la 1ère chance. Ce mécanisme assure que les moyennes des éléments supérieurs (matière, unité, bloc, filière) sont toujours à jour avec les notes disponibles, fournissant ainsi une évaluation continue et en temps réel des performances des étudiants.

5.3.5 Visualisation des notes

Notre application permet de visualiser les notes sur trois pages différentes :

1. **Page des étudiants de la filière** : Chaque filière possède une page affichant les étudiants inscrits et leurs moyennes.
2. **Page des informations de l'étudiant** : Cette page affiche le relevé de notes de l'étudiant pour chaque année, avec un diagramme clair des notes des unités.
3. **Page des statistiques des filières** : Des histogrammes permettent de visualiser les notes des unités de la filière.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	13/12/2022	Date:	1/		Page:	13:21:42	Heure:	Apogée	Univers	Liste de con	Demandeur:	CHEV	Florence	Chevreau / Thomas			
2	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	1234567	Nom	Prénom	Résultat					8 /	20
3	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	23456789	Nom	Prénom	Résultat					9 /	20
4	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	34567890	Nom	Prénom	Résultat					4.5 /	20
5	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	45678901	Nom	Prénom	Résultat					7 /	20
6	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	56789012	Nom	Prénom	Résultat					12 /	20
7	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	67890123	Nom	Prénom	Résultat					4 /	20
8	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	78901234	Nom	Prénom	Résultat				ABI		
9	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	89012345	Nom	Prénom	Résultat					12 /	20
10	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	90123456	Nom	Prénom	Résultat					20 /	20
11	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	12345678	Nom	Prénom	Résultat					0 /	20
12	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	23456789	Nom	Prénom	Résultat					17 /	20
13	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	34567890	Nom	Prénom	Résultat					11.5 /	20
14	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	45678901	Nom	Prénom	Résultat					1.5 /	20
15	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	56789012	Nom	Prénom	Résultat					7 /	20
16	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	67890123	Nom	Prénom	Résultat					0 /	20
17	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	78901234	Nom	Prénom	Résultat					10 /	20
18	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	89012345	Nom	Prénom	Résultat					1.5 /	20
19	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	90123456	Nom	Prénom	Résultat				ABI		
20	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	12345678	Nom	Prénom	Résultat					6 /	20
21	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	23456789	Nom	Prénom	Résultat					19 /	20
22	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	34567890	Nom	Prénom	Résultat					4.5 /	20
23	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	45678901	Nom	Prénom	Résultat					11 /	20
24	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	56789012	Nom	Prénom	Résultat				ABI		
25	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	67890123	Nom	Prénom	Résultat					16 /	20
26	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	78901234	Nom	Prénom	Résultat					17.5 /	20
27	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	89012345	Nom	Prénom	Résultat					17 /	20
28	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	90123456	Nom	Prénom	Résultat					13.5 /	20
29	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	12345678	Nom	Prénom	Résultat					5 /	20
30	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	23456789	Nom	Prénom	Résultat					9 /	20
31	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	34567890	Nom	Prénom	Résultat					10.5 /	20
32	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	45678901	Nom	Prénom	Résultat					13.5 /	20
33	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	56789012	Nom	Prénom	Résultat					3 /	20
34	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	67890123	Nom	Prénom	Résultat					10.5 /	20
35	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	78901234	Nom	Prénom	Résultat					8.5 /	20
36	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	89012345	Nom	Prénom	Résultat					6 /	20
37	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	90123456	Nom	Prénom	Résultat					16 /	20
38	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	12345678	Nom	Prénom	Résultat					13 /	20
39	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	23456789	Nom	Prénom	Résultat					15 /	20
40	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	34567890	Nom	Prénom	Résultat				ABI		
41	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	45678901	Nom	Prénom	Résultat					9 /	20
42	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	56789012	Nom	Prénom	Résultat					18 /	20
43	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	67890123	Nom	Prénom	Résultat					7 /	20
44	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	N° Etudiant	Nom	Prénom	Résultat					9 /	20
45	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	N° Etudiant	Nom	Prénom	Résultat					1.5 /	20
46	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	N° Etudiant	Nom	Prénom	Résultat					2.5 /	20
47	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	N° Etudiant	Nom	Prénom	Résultat					7.5 /	20
48	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	N° Etudiant	Nom	Prénom	Résultat					6 /	20
49	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	N° Etudiant	Nom	Prénom	Résultat					16.5 /	20
50	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	N° Etudiant	Nom	Prénom	Résultat					0 /	20
51	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	N° Etudiant	Nom	Prénom	Résultat					16 /	20
52	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	N° Etudiant	Nom	Prénom	Résultat					13 /	20
53	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	N° Etudiant	Nom	Prénom	Résultat					1 /	20
54	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	N° Etudiant	Nom	Prénom	Résultat					2.5 /	20
55	Session :	Session 1	Admission	Epreuve :	TMX1212C1	CC analyse élém	Note	N° Etudiant	Nom	Prénom	Résultat					4 /	20

FIGURE 5.6 – Exemple du format du fichier CSV des notes

 Étudiants Maquettes Insérer des étudiants Explorer					
Rechercher un étudiant...					
Année Universitaire 2023					
Numéro Étudiant	Nom ▲	Prenom ▲	Note ▲	Statut	Résultats
2200001	Saunders	Carla	8.19	Ajourné	résultat
2200002	Chaney	Jason	13.77	Admise	résultat
2200003	Bowman	Michael	5.69	Ajournée	résultat
2200004	Morgan	Stephanie	7.69	Ajournée	résultat
2200005	Hammond	Lori	8.98	Ajourné	résultat
2200006	Wells	Logan	11.63	Admise	résultat
2200007	Davis	Ellen	10.29	Admise	résultat
2200008	Spears	Kenneth	10.50	Admise	résultat
2200009	Bass	Victor	2.69	Ajourné	résultat
2200010	Lee	Rebecca	11.24	Admise	résultat
22000100	Allen	Joseph	6.64	Ajournée	résultat

FIGURE 5.7 – Page des étudiants de la filière



FIGURE 5.8 – Page des informations de l'étudiant

Statistique de la MI-Informatique (MI-I)



FIGURE 5.9 – Page des statistiques des filières

Chapitre 6

Conclusion

Ce projet nous a permis de créer une application web fonctionnelle et sécurisée pour la gestion des étudiants et des notes. Nous avons implémenté des fonctionnalités essentielles qui faciliteront les tâches des différents utilisateurs en termes de gestion des étudiants et des notes, notamment l'insertion et la visualisation d'un grand nombre d'étudiants respectivement à travers un formulaire d'importation et un tableau, la gestion des maquettes, la consultation des informations et des résultats des étudiants, ainsi que la consultation de quelques statistiques des différentes filières. Bien que nous ayons rencontré quelques défis en cours de route, nous avons réussi à surmonter les obstacles techniques et à développer des compétences précieuses en développement d'application tout au long du processus.

Nous avons également identifié plusieurs opportunités d'amélioration pour le projet. Par exemple, l'implémentation de fonctionnalités supplémentaires sur les statistiques des étudiants et l'ajout d'autres fonctionnalités permettant d'améliorer la vérification lors des différentes insertions. Nous avons pris en compte ces aspects dans la conception et l'architecture de l'application afin de faciliter leur intégration ultérieure.

En travaillant sur ce projet, nous avons consolidé nos compétences en conception de bases de données et en développement front et backend, avec la maîtrise du framework Symfony et de nombreux de ses composants, dont le composant Security pour la mise en place des systèmes d'authentification, l'ORM Doctrine, le moteur de template Twig, ainsi que des technologies comme Bootstrap et la bibliothèque ChartJS pour les graphiques. Nous avons également renforcé notre compréhension des bonnes pratiques de codage, de la structuration de la base de données et de la gestion des dépendances.

En fin de compte, ce projet a constitué une expérience d'apprentissage inestimable pour notre future carrière dans le développement informatique. Nous sommes fiers de vous offrir cet outil qui vous aidera dans le futur. Les résultats que nous avons obtenus jusqu'à présent témoignent de notre engagement et de notre compétence, et nous sommes également conscients des opportunités d'amélioration et des fonctionnalités supplémentaires que nous pourrions ajouter à l'avenir.

En conclusion, nous tenons à remercier tous ceux qui ont contribué à la réalisation de ce projet, notamment nos encadrants qui nous ont orientés et aidés à garder un bon rythme tout au long du projet. Ce projet a été une expérience enrichissante qui a renforcé nos compétences et notre passion pour le développement d'applications web.

Chapitre 7

Prérequis pour utiliser le projet

7.1 Récupération du code source depuis GitHub

Pour récupérer le code source de notre projet, rendez-vous sur le dépôt GitHub à l'adresse : <https://github.com/HabibKHTEIRA/Gestion-des-notes-des-etudiants.git>. Vous trouverez le projet dans la branche `main`. Vous pouvez ensuite récupérer le projet en copiant le lien du dépôt et en le clonant ou bien en téléchargeant le zip du projet et en le décompressant sur votre machine.

7.2 Installation de PHP 8.3

Pour télécharger PHP 8.3, rendez-vous sur le site officiel de PHP : <https://www.php.net/downloads.php> et suivez les instructions d'installation spécifiques à votre système d'exploitation. Vous pouvez également utiliser les commandes suivantes pour Linux :

```
sudo add-apt-repository ppa:ondrej/php
sudo apt install php8.3 libapache2-mod-php8.3 php8.3-mysql php8.3-common
php8.3-xml php8.3-xmlrpc php8.3-gd php8.3-imagick php8.3-dev php8.3-imap
php8.3-mbstring php8.3-openssl php8.3-zip php8.3-intl -y
php -v # Assurez-vous que la version de PHP est supérieure à 8.2
```

7.3 Installation de Composer

Vous pouvez télécharger et installer Composer à partir du site officiel : <https://getcomposer.org/> ou bien en utilisant les commandes suivantes pour Linux :

```
php -r "copy('https://getcomposer.org/installer', 'composer-setup.php');"
php -r "if (hash_file('sha384', 'composer-setup.php') ===
'dac665fdc30fdd8ec78b38b9800061b4150413ff2e3b6f88543c636f7cd84f6db9189d43a8
1e5503cda447da73c7e5b6') { echo 'Installer verified'; } else { echo 'Installer
corrupt'; unlink('composer-setup.php'); } echo PHP_EOL;"
php composer-setup.php
php -r "unlink('composer-setup.php');"
sudo mv composer.phar /usr/local/bin/composer
composer -v # Assurez-vous que la version de Composer est 2.7.2
```

7.4 Installation de Symfony

Pour installer Symfony, vous pouvez utiliser les commandes suivantes :

```
curl -sS https://get.symfony.com/cli/installer | bash
sudo mv /home/etud/.symfony5/bin/symfony /usr/local/bin/symfony
symfony check:requirements
symfony -v    # Assurez-vous que la version de Symfony est 5.8.14
```

7.5 Installation de XAMPP

Pour télécharger XAMPP, rendez-vous sur le site officiel : <https://www.apachefriends.org/download.html> Ensuite, suivez les instructions d'installation spécifiques à votre système d'exploitation.

7.6 Installation des dépendances du projet

Rendez-vous dans le répertoire du projet et lancez la commande `composer install` pour installer les dépendances du projet. Ensuite, lancez la commande `symfony serve` pour démarrer le serveur local.

Chapitre 8

Références

- *Rapport PDF de l'année passée rédigé par Moussa Deh, Adam Afif et Taise de Thes Ngangua Yabie*. Université d'Angers, 2022-2023.
- *Maquette de L3*. Disponible sur https://formations.univ-angers.fr/odf/_plugin-attachme subprogram-l1-mi-parcours-informatique/L3_INFO_2023-2024.pdf.
- *Documentation Symfony*. Disponible sur <https://symfony.com/doc/current/index.html>.
- *Documentation Twig*. Disponible sur <https://twig.symfony.com/doc/2.x/>.
- *Mise en forme Bootstrap*. Disponible sur <https://getbootstrap.com/docs/5.3/getting-started/introduction/>.
- Documents utilisés. Disponibles à la scolarité de l'UA.