

République Algérienne Démocratique et Populaire
Ministère de l'enseignement supérieur et de la recherche scientifique



Université 20 Août -1955-Skikda

Faculté des sciences- Département d'informatique

Mémoire de fin d'études pour l'obtention du diplôme de Master

Académique en informatique.

Option : SI

THEME

Mise à jour MVC Pour le site web du
département d'informatique

Réalisé par les étudiants :

BOUANENE RIM

DADI SISTA WIAM

Dirigé par :

Mr. LAROUM TOUFIK

Remerciement

*Louange à Dieu qui nous a éclairés sur le chemin de la science et de la connaissance, qui nous a aidés à accomplir notre devoir et qui nous a aidés à accomplir cette œuvre. Nous adressons nos sincères remerciements et notre appréciation à tous ceux qui nous ont aidés, de près ou de loin, à mener à bien ce travail, et nous remercions notre éminent professeur, le Dr « **Laroum Toufik** », pour avoir aimablement supervisé cette recherche et pour tous les conseils et efforts qu'il nous a fournis tout au long de la réalisation de cette recherche. À tous les amis et collègues qui nous ont aidés à préparer cette recherche, nous demandons à Dieu de bien les récompenser pour nous et de faire leur travail dans la balance de leurs bonnes actions.*

Dédicaces

Louange à Dieu, pour l'amour, la gratitude et la gratitude pour le début et la conclusion de celui qui a dit : « Je suis pour elle, il l'a eu » Le voyage n'a pas été court et ne devrait pas l'être, le rêve n'était pas proche et la route n'était pas pleine de facilités, mais je l'ai fait et je l'ai reçu, et me voici aujourd'hui debout devant vous et mon cœur est rempli d'un mélange de sentiments contradictoires, de joie d'accomplissement et de tristesse de sentiments de séparation mélangés aux souvenirs d'un long voyage éducatif plein de défis et de réalisations À ceux qui ont planté en moi une morale noble,

*L'amour de la science et le respect des autres Au symbole du don et du sacrifice Mon idéal, à celui dont je porte fièrement le nom « **mon cher père, que Dieu le protège et prolonge sa vie** ».*

*À celui qui m'a porté et félicité pour sa faiblesse, au symbole de l'amour et du baume guérisseur, au sens de la tendresse et du dévouement, à ceux dont les prières ont été le secret de mon succès, à la bien-aimée très précieuse « **ma chère mère, que Dieu prolonge sa vie** ».*

*À ceux à qui je me confie dans les moments d'adversité, à ceux qui sont plus musclés pour eux, à ceux qui ont des intentions sincères et des cœurs bons, et je vois de l'optimisme dans leurs yeux Mes frères : <<**Khaled SEIF EDDINE, Hamza, Mousaab** >>.et ma nièce **NAYA siline***

*À ceux que j'ai parcourus avec eux sur les chemins de la vie douce et amère, À ceux qui étaient avec moi sur le chemin du succès et de la bonté « Mes amis : **Nor EL Hoda, Randa** ».*

RYM

Dédicaces

Je dédie ce modeste travail à :

*A mes parents. Aucun hommage ne pourrait être à la hauteur
De l'amour Dont ils ne cessent de me combler. Que dieu leur procure
Bonne santé et longue vie.*

*A celui que j'aime beaucoup et qui m'a soutenue tout au long de ce
projet mon fiancé **HAMZA**.*

*Et bien sûr A mes frères **SARA, ACHRAF, AMIRA, BESSEMALA**
et **RIHAB**.*

Sans oublier la femme de mon frère.

Les maris de mes sœurs et les enfants de ma sœur

Bellsem et mohamed tamime

*A toute ma famille, et mes amis, Et à tous ceux qui ont contribué de
près ou de loin pour que ce projet soit possible, je vous dis merci.*

WIAM

ملخص

على مر السنين، كان التقدم في تكنولوجيا المعلومات والاتصالات الجديدة (NICT) ملحوظا. ومن أروع مظاهر هذا التطور بلا شك ظهور الإنترنت. ومن خلال هذه الشبكة، تم تطوير عدد كبير من البرامج، بما في ذلك تطبيقات الويب التي تعمل على تبسيط العمل الإداري اليومي للمؤسسات والشركات، بما في ذلك في مجال التعليم والتدريب العالي والبحث العلمي.

وفي إطار تحسين وتحديث موقع قسم علوم الحاسوب بكلية العلوم بجامعة سكيكدة، تقدم هذه التطورات التكنولوجية حلا فعالة. يساعد هذا التطبيق على تبسيط العمليات الإدارية وتقليل السفر الجسدي وتحسين التواصل بين المعلمين والطلاب. وبالتالي فهو يوفر منصة عملية وفعالة لاستشارة الإعلانات وإدارة الطعون واقتراح موضوعات مشروع نهاية الدراسة. استعملنا في هذا المشروع إطار العمل Laravel من أجل تطوير نسخة MCV لموقع قسم الاعلام الآلي.

Résumé

Au fil des années, les progrès dans les Nouvelles Technologies de l'Information et de la Communication (NTIC) ont été remarquables. L'une des manifestations les plus spectaculaires De cette évolution est indéniablement l'avènement d'Internet. À travers ce réseau, une pléthore de programmes a été développée, notamment des applications web qui simplifient le travail administratif quotidien des institutions et des entreprises, y compris dans le domaine de l'éducation, de la formation supérieure et de la recherche scientifique.

Dans le cadre de l'amélioration et de la mise à jour du site du Département d'Informatique de la Faculté des Sciences de l'Université de Skikda, ces avancées technologiques offrent des solutions efficaces.

Cette application permet de rationaliser les processus administratifs, de réduire les déplacements physiques et d'améliorer la communication entre enseignants et étudiants. Elle offre ainsi une plateforme pratique et efficace pour la consultation des annonces, la gestion des recours et la proposition des thèmes de

projets de fin d'études. Nous avons utilisé Laravel pour développer une version MVC du site web du département d'informatique.

Abstract

Over the years, progress in New Information and Communication Technologies (NICT) has been remarkable. One of the most spectacular manifestations of this evolution is undoubtedly the advent of the Internet. Through this network, a plethora of programs have been developed, including web applications that simplify the daily administrative work of institutions and companies, including in the field of education, higher training and scientific research.

As part of the improvement and updating of the site of the Department of Computer Science of the Faculty of Sciences of the University of Skikda, these technological advances offer effective solutions. This application helps streamline administrative processes, reduce physical travel and improve communication between teachers and students. It thus offers a practical and effective platform for consulting announcements, managing appeals and proposing end-of-study project themes. We used Laravel Framework to develop a MVC version of the web site of computer sciences department.

Table des matières

Introduction Générale.....	1
Chapitre 01 : les application web et les application mobile	3
1.Introduction	4
2. Généralité sur le web	4
2.1.La définition de web	4
2.2. L'historique du web	4
2.3.page web	5
2.4.site web	5
2.5. les application web	8
3.larchitecture client-serveur	9
3.1.la fonctionnement d'un système client/serveur	10
3.2. les types des Architectures client /serveur	11
3.3. les caractéristique d'un serveur	14
3.4.les caractéristique d'un client.....	14
4. les application mobile	14
4.1. les systèmes d'exploitation mobiles.....	15
4.2..les types des application mobile	18
4.3. les avantage et les inconvénients des application mobile	20
5. conclusion	21
Chapitre 02 : Laravel.....	22
1.Introduction.....	23
2.Qu' est –ce que laravel.....	23
2.1. pourquoi utiliser laravel ?	23
2.2. Avantage de laravel.....	23

2.3. Qu'est-ce qu'un Framework ?	24
2.4.modèle –vue-contrôleur mvc	24
2.5. Tout ce dont vous avez besoin pour utilise laravel Blade.....	27
3.comment installer Laravel ?.....	28
3.1.installez Xampp	28
3.2.installez composer.....	29
3.3. installez laravel à l'aide de composer	30
4. structure du projet	32
4.1. Routage et contrôleurs	33
5. Eloquent Laravel.....	34
5.1. Qu'est-ce que laravel Eloquent ?	34
5.2. configuration d'un ORM éloquent.....	35
5. 3.Que sont les relation éloquentes ?.....	35
5. 4.Avantage des relations éloquentes	36
5.5.base de données.....	36
6. Laravel Middleware ?	39
6. 1.Qu'est-ce qu'un middleware ?.....	39
6.2. Importance du middleware dans le Framework Laravel.....	40
6. 3.Explication du middleware dans le développement web	40
7. Notifications Laravel	41
7.1. Qu'est-ce qu'un Notifications ?	41
7. 2.canaux de notification de laravel	41
8. Laravel Breeze	41
8.1. Qu'est-ce que laravel Breeze ?	41
8.2. pourquoi choisir Laravel Breeze	42
8.3.Installer Laravel Breeze	42
9. conclusion	46

Chapitre 03 : Analyse et conception	47
1.Introduction	48
2. la méthode UML	48
2.1. Définition	48
2.2.pourquoi la méthode UML.....	48
2.3. principe UML	49
2.4 Définition d'un Diagramme	50
2.5. Vues UML	50
2. 6. Pourquoi modéliser	51
2.7. Les avantages et les inconvénients d'UML	52
3. la conception de l'application	53
3.1. Diagramme de cas d'utilisation	53
3. 2.Diagramme de séquence	57
4. conclusion	62
Chapitre 04 : Implémentation	63
1. Introduction.....	64
2. outils de développement	64
2.1. la machine de développement.....	64
2. 2.langages de programmation	64
2. 3. Les outils.....	66
3.Quelques exemples d'interface	67
4. conclusion	76
.Conclusion Générale	78
Références Bibliographiques	80

Table des figures

N Figure	Figure	page
Figure1.1	connections entre client et serveur, cas d'une page statique	06
Figure1.2	Connections entre client et serveur, cas d'une page dynamique	07
Figure1.3	L'architecture client/serveur	10
Figure1.4	Schéma de fonctionnement d'un système client/serveur	11
Figure1.5	architecture à 2 niveaux	12
Figure1.6	architecture à 3 niveaux	13
Figure1.7	architecture multi niveaux	14
Figure1.8	L'icône représentant le système Android	15
Figure1.9	L'icône représentant le système i OS	16
Figure1.10	L'icône représentant le système BlackBerry OS	16
Figure1.11	L'icône représentant le système Windows mobiles	17
Figure1.12	Exemplaire d'un téléphone sous Symbian OS	18
Figure1.13	Les types de l'application mobile	19
Figure 2.1	Approche de développement MVC	25
Figure 2.2	installation de XAMPP	29
Figure 2.3	vérifiez l'installation de composer	30
Figure 2.4	Page d'accueil de Laravel	31
Figure 2.5	Architecture des dossiers de Laravel	32
Figure 2.6	Exemple les route	34
Figure 2.7	Le formulaire de création de base de données	37
Figure 2.8	Exemple basique de Code de la table posts	38
Figure 2.9	Exécution des migrations	39
Figure 2.10	Tables créées	39
Figure 2.11	Exemple basique code de la table users	43
Figure 2.12	les routes de laravel breeze	44
Figure 2.13	l'écran initial après installation Breeze	44
Figure 2.14	l'interface de connexion et d'inscription	45
Figure 2.15	L'interface de page du tableau de bord	46
Figure 3.1	Diagramme de cas d'utilisation du l'étudiant	54

Figure 3.2	Diagramme de cas d'utilisation de l'enseignant	55
Figure 3.3	Diagramme de cas utilisation d'administration	56
Figure 3.4	Diagramme de Séquence d'authentification	57
Figure 3.5	Diagramme de Séquence de faire recours	58
Figure 3.6	Diagramme de Séquence de faire annonce	59
Figure 3.7	Diagramme de Séquence Ajouter PFE	60
Figure 4.1	Page d'accueil enseignant	68
Figure 4.2	page d'accueil étudiants	68
Figure 4.3	Page de connexion enseignant	69
Figure 4.4	Page inscription étudiants	69
Figure 4.5	Page de connexion étudiants	70
Figure 4.6	Page d'affichage enseignant	70
Figure 4.7	.Page mes d'affichage	71
Figure 4.8	Page fiche d'encadrement	71
Figure 4.9	Page mes projets	72
Figure 4.10	Page d'affichage étudiant	72
Figure 4.11	Page choisir les affichages	73
Figure 4.12	Page de notification par email	73

Introduction Général

Introduction Générale

Le monde actuel est témoin d'une avancée technologique sans précédent dans divers secteurs, notamment les domaines scientifiques et économiques. Cette progression rapide est en grande partie attribuable à l'informatique, une science dédiée aux techniques de traitement automatique de l'information.

Avant l'avènement de l'ordinateur, les informations étaient enregistrées manuellement sur des supports papier, ce qui entraînait de nombreux problèmes tels que la perte de temps considérable dans la recherche d'informations, la dégradation des documents et des défis en matière de sécurité des données.

Aujourd'hui, l'ordinateur demeure l'outil le plus fiable pour le traitement et la sauvegarde de l'information. Cette invention a permis l'informatisation des systèmes de données des entreprises, facilitant ainsi l'exploitation et l'analyse de diverses ressources grâce à des logiciels performants. L'internet, une architecture interconnectant différents ordinateurs, joue également un rôle crucial dans cette transformation.

Grâce à ce réseau mondial, il est désormais possible de communiquer instantanément avec presque n'importe quel endroit de la planète. Le monde entier s'invite chez vous à travers l'écran de votre ordinateur, permettant de nouvelles formes de relation et d'interaction. Les sites Web sont au cœur de cette interaction, fournissant une plateforme pour l'échange d'informations et la communication globale.

Ce mémoire se focalise sur la conception et la réalisation d'un site web et d'une application mobile pour le département informatique de l'université de Skikda. Ces plateformes technologiques jouent un rôle vital dans la communication et la gestion de l'information, promouvant l'image de l'université et fournissant des informations essentielles sur les programmes d'études, les services aux étudiants, les événements universitaires.

L'application que nous avons développée offre aux enseignants un moyen efficace de consulter et de publier des annonces, de répondre aux recours soumis par les étudiants, et de proposer des thèmes de projets de fin d'études, le tout sans avoir à se déplacer physiquement au département. Cela permet non seulement de gagner un temps précieux mais aussi d'améliorer l'efficacité des processus.

De leur côté, les étudiants peuvent consulter les annonces et soumettre des recours via l'application.

Pour atteindre notre objectif, notre travail est structuré comme suit :

Chapitre 1 : Notions de base du développement d'applications web et mobile

Ce chapitre introduira les concepts essentiels au développement d'applications modernes, offrant aux lecteurs une base solide en la matière.

Chapitre 2 : Création d'un nouveau projet avec le Framework Laravel

Ici, nous décrirons en détail le processus de création d'un projet en utilisant Laravel, un Framework populaire pour le développement web.

Chapitre 3 : Phase d'analyse et de conception de l'application

Ce chapitre abordera la méthodologie adoptée, les besoins identifiés et les fonctionnalités spécifiques de notre application. Nous mettrons également en lumière les choix d'architecture logicielle et de conception de bases de données.

Chapitre 4 : Outils et interfaces développés Enfin, nous présenterons les différents outils et interfaces que nous avons créés, illustrant concrètement comment notre application répond aux besoins initialement définis.

En somme, ce mémoire vise à démontrer comment une application bien conçue peut améliorer la communication et la gestion des informations dans un contexte universitaire, tout en simplifiant les processus pour les enseignants et les étudiants.

Chapitre 01 : Les application web et les applications mobile

1.Introduction

Au cours des dernières années, Internet s'est imposé comme l'outil de communication prédominant à l'échelle mondiale, connectant des millions d'utilisateurs bien au-delà des médias traditionnels. Ce chapitre se propose d'aborder divers concepts liés au web, tels que les définitions essentielles, la classification des sites web, le fonctionnement de l'architecture client-serveur, ainsi que l'importance stratégique de posséder une présence en ligne. De plus, nous aborderons de manière générale les applications mobiles, en examinant les différents systèmes d'exploitation et les catégories d'applications qui les composent.

2. Généralité sur le web

2.1. La définition de web

En informatique, une application web (aussi appelée web application, de l'anglais) est une application manipulable directement en ligne grâce à un navigateur web et qui ne nécessite donc pas d'installation sur les machines clientes, contrairement aux applications mobiles. De la même manière que les sites web, une application web est généralement installée sur un serveur et se manipule en actionnant des widgets à l'aide d'un navigateur web, via un réseau informatique (Internet, intranet, réseau local, etc.). La messagerie web, les systèmes de gestion de contenu, les wikis et les blogs représentent des exemples d'applications web.

Les moteurs de recherches, les logiciels de commerce électronique, les jeux en ligne, les logiciels de forum, les agrégateurs peuvent être sous forme d'application web. Des appareils réseau tels que les routeurs sont parfois équipés d'une application web dans leur micro logiciel. Les applications web font partie de l'évolution des usages et de la technologie du Web appelée Web 2.0. [1]

2.2. L'historique du Web

Le Word Wide Web (www) a été mis en place par Tim-Berners Lee qui est considéré comme le père fondateur du web. Au milieu des années 1990, Internet fait son apparition au grand public en version 1.0 via des pages statiques codés en HTML. Il s'agit de sites non interactifs principalement destinés (la recherche d'informations : encyclopédies, etc. Au début des années 2000, le web a évolué et il est devenu dynamique. Il s'agit de sa version 1.5. Il est maintenant possible de consulter du contenu dynamique en ligne, via des bases de données :

boutique en ligne, etc. (A noter que c'est aussi l'avènement des start-up qui surfent sur la vague du web dynamique, accessible à tous).

En 2004, Dale Dougherty utilise le terme « Web 2.0 » qui sera vite repris par Tim O'Reilly, spécialiste du World Wide Web ; l'apparition du web 2.0 prend une véritable dimension sociologique, pour Joshua Porter, le web 2.0 « c'est le partage de l'information, fondé sur des bases de données ouvertes qui permettent à d'autres utilisateurs de les employer.

Depuis quelques mois, on entend de plus en plus parler du Web 3.0, c'est-à-dire un web encore plus humain, encore plus intelligent. [2]

2.3. Page web

Une page web est une ressource du world wide web conçue pour être consultée par des visiteurs à l'aide d'un navigateur web. Techniquement, une page web est souvent constituée d'un document en HTML et d'images. [2]

2.4. Site web

Un site web est un ensemble de pages écrites dans un langage normalisé de manière à être compris par tous les navigateurs. L'organisme qui décrit et décide des évolutions de ce langage s'appelle le W3C, et préconise l'utilisation du HTML pour la création de pages web. [2]

2.4.1. Le site web statique

Un site statique est composé uniquement de pages html créées une par une par un webmestre. Dans ce cadre, chaque modification de page demande la réédition de cette dernière et une nouvelle mise en place sur le serveur. L'ordinateur qui se connecte au serveur, demande une page. Celle-ci lui est directement servie (elle est stockée toute prête sur le serveur). [3]



Figure 1.1:connections entre client et serveur, cas d'une page statique.

❖ Les avantages

- Rapide et facile à mettre en place, même par quelqu'un qui ne possède pas beaucoup d'expérience (par un éditeur Web par exemple : FrontPage, Dreamweaver...).
- Cache convivial, une copie peut être démontré à de nombreuses personnes.

Ne nécessite pas de ressources particulières (langages de scripts dynamiques, fonction mail...), un serveur ordinaire fait l'affaire.

- Un coût réduit de conception et hébergement.
- Adéquat pour un site personnel, un site vitrine d'une petite compagnie pour montrer des infos ou produits à changement peu fréquent.

❖ Les Inconvénients

- Difficile à maintenir quand un site est de grande taille.
- Difficile de garder la cohérence à jour.
- Offre peu de personnalisation aux visiteurs (tous de vrait être du côté client).

- Une mise à jour stagnée (difficile à la suivre si on a besoin plus fréquemment).
- Peu pratique pour un utilisateur (s'il veut envoyer des commentaires, messages n'y a pas de Formulaire).

2.4.2. Le site web dynamique

Le site Web dynamique est un site Web dont les pages sont générées dynamiquement à la demande. Le contenu est obtenu (par exemple) en combinant l'utilisation d'un langage de scripts ou de programmation et une base de données. Il s'agit souvent de PHP pour le langage et MySQL pour la base de données. [3]



Figure 1.2 : Connexions entre client et serveur, cas d'une page dynamique.

❖ Les Avantages

- Un site beaucoup plus fonctionnel (réception de données, commentaires, messages, ...par l'utilisateur vers le serveur distant ou insertion dans la base de données. Ainsi qu'un nombre d'autres applications, en résumer l'implémentation de modules complémentaires (impression instantanée de facture selon nos commandes, suivie de commande, paiement en ligne...).
- Mise à jour beaucoup plus facile par l'administrateur ou propriétaire du site grâce au CMS (Système de gestion de contenu).

- Nouveau contenu brings people back to the site and helps in the search engines
- Peu fonctionner comme un système permettant les collègues de collaborer et travailler ensemble sur un même projet
- Conviviale, une meilleure expérience aux visiteurs qui peuvent personnaliser certaines rubriques/parties du site (apparence, ajout de contenu, avoir une zone membre privée...)
- Affichage dynamique de produits (du nouveau à l'ancien, insertion d'un moteur de recherche multicritères)
- Tous les sites réussis du monde sont dynamiques
- La bonne solution pour un site volumineux nécessitant une mise à jour régulier.

❖ **Les Inconvénients**

- Plus lent / Plus cher à développé
- Un hébergement un peu plus coûteux

2 .5. Les applications web

En informatique, une application web (aussi appelée web application, de l'anglais) est une application manipulable directement en ligne grâce à un navigateur web et qui ne nécessite donc pas d'installation sur les machines clientes, contrairement aux applications mobiles. De la même manière que les sites web, une application web est généralement installée sur un serveur et se manipule en actionnant des widgets à l'aide d'un navigateur web, via un réseau informatique (Internet, intranet, réseau local, etc.). La messagerie web, les systèmes de gestion de contenu, les wikis et les blogs représentent des exemples d'applications web. Les moteurs de recherches, les logiciels de commerce électronique, les jeux en ligne, les logiciels de forum, les agrégateurs peuvent être sous forme d'application web. Des appareils réseau tels que les routeurs sont parfois équipés d'une application web dans leur micro logiciel. Les applications web font partie de l'évolution des usages et de la technologie du Web appelée Web 2.0. **[1]**

2 .5.1. La différence entre Application Web ou site Web

Un [site web](#) est ce que l'on retrouve lorsqu'on entre une adresse et qu'on atterrit sur une page. Le site correspond à cette page et toute page reliée qui est gérée par la même entité (souvent sous le même nom de domaine). Le rôle principal d'un site web est de fournir et présenter de l'information aux visiteurs. Un blogue, un site de nouvelles ou un site d'information sur un produit ou une compagnie sont de bons exemples de sites web.

Une [application web](#) est tout site web qui permet à ses utilisateurs d'accomplir des tâches spécifiques. Une application gère donc généralement des utilisateurs et toutes sortes de données selon les requis spécifiques au projet. [4]

2 .5.2. Les avantages du développement d'application Web

L'application web est le meilleur outil pour atteindre vos objectifs. Contrairement aux autres médias, le Web est accessible de partout à travers le monde, il y a de nombreux avantages au développement d'application web parmi eux, vous trouverez

- C'est moins cher
- Un accès plus rapide
- Fonctionne sur tous les systèmes d'exploitation
- Sécurité des contenus
- Accessible de partout
- Travail en simultané dans « Cloud » [5]

3. L'architecture client-serveur

De nombreuses applications fonctionnent selon un environnement client/serveur, cela signifie que des machines clientes (des machines faisant partie du réseau) contactent un serveur, une machine généralement très puissante en termes de capacités d'entrée-sortie, qui leur fournit des services. Ces services sont des programmes fournissant des données telles que l'heure, des fichiers, une connexion, etc.

Les services sont exploités par des programmes, appelés programmes clients, s'exécutant sur les machines clientes. On parle ainsi de client lorsque l'on désigne un programme tournant

sur une machine cliente, capable de traiter des informations qu'il récupère auprès d'un serveur (dans le cas du client FTP il s'agit de fichiers, tandis que pour le client de messagerie il s'agit de courrier électronique). [6]

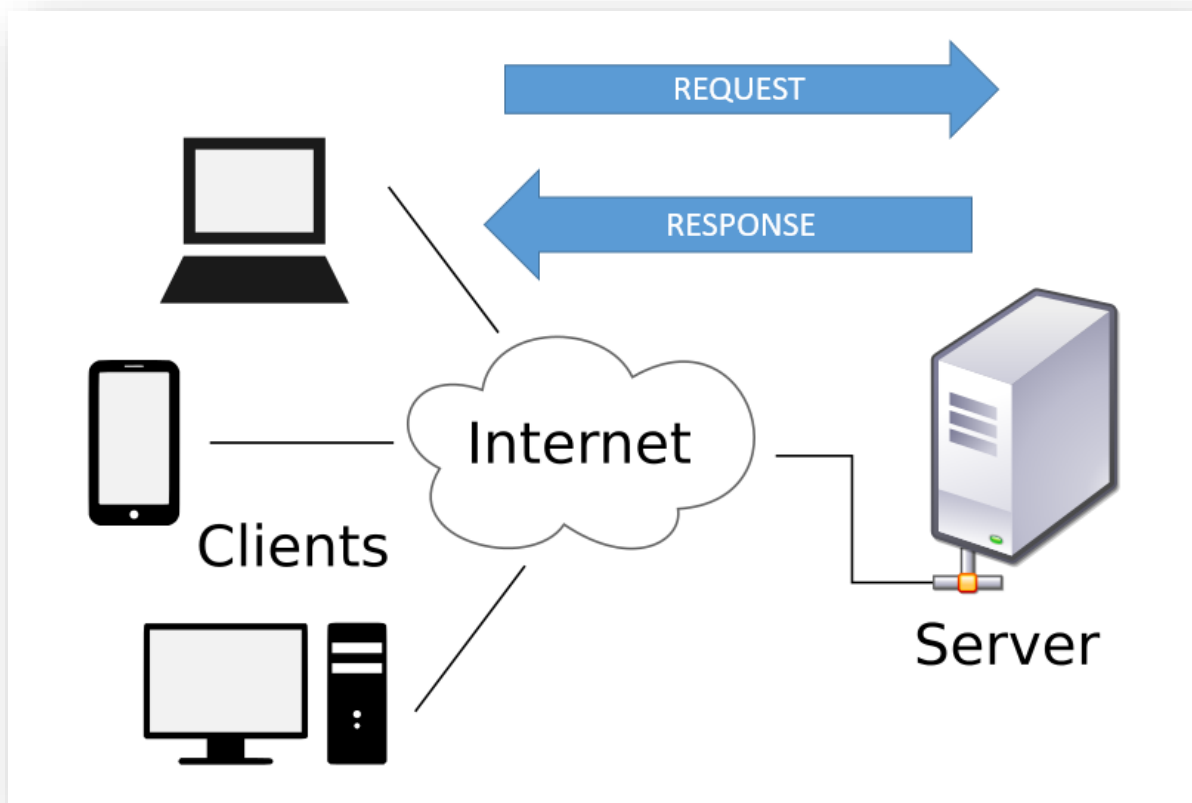


Figure 1.3 : L'architecture client/serveur.

3.1. La Fonctionnement d'un système client/serveur

Depuis quelques mois, on entend de plus en plus parler du Web 3.0, c'est-à-dire un web encore plus humain, encore plus intelligent. N système client/serveur fonctionne selon le schéma suivant :

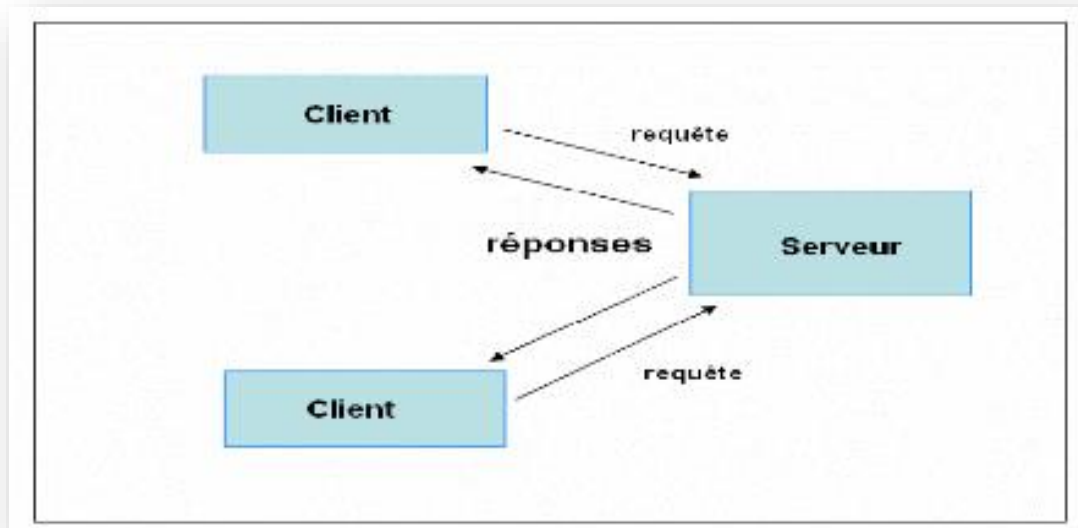


Figure 1.4: Schéma de fonctionnement d'un système client/serveur.

- Le client émet une requête vers le serveur grâce à son adresse IP et le port, qui désigne un service particulier du serveur.
- Le serveur reçoit la demande et répond à l'aide de l'adresse de la machine cliente et son port. [6]

3.2. les types des Architectures client/serveur

3.2.1. L'architecture client/serveur à 2 niveaux

Aussi appelée architecture 2-tier, tier (signifiant étage en anglais) caractérise les systèmes clients/serveurs dans lesquels le client demande une ressource et le serveur la lui fournit directement. Cela signifie que le serveur ne fait pas appel à une autre application afin de fournir le service. [7]

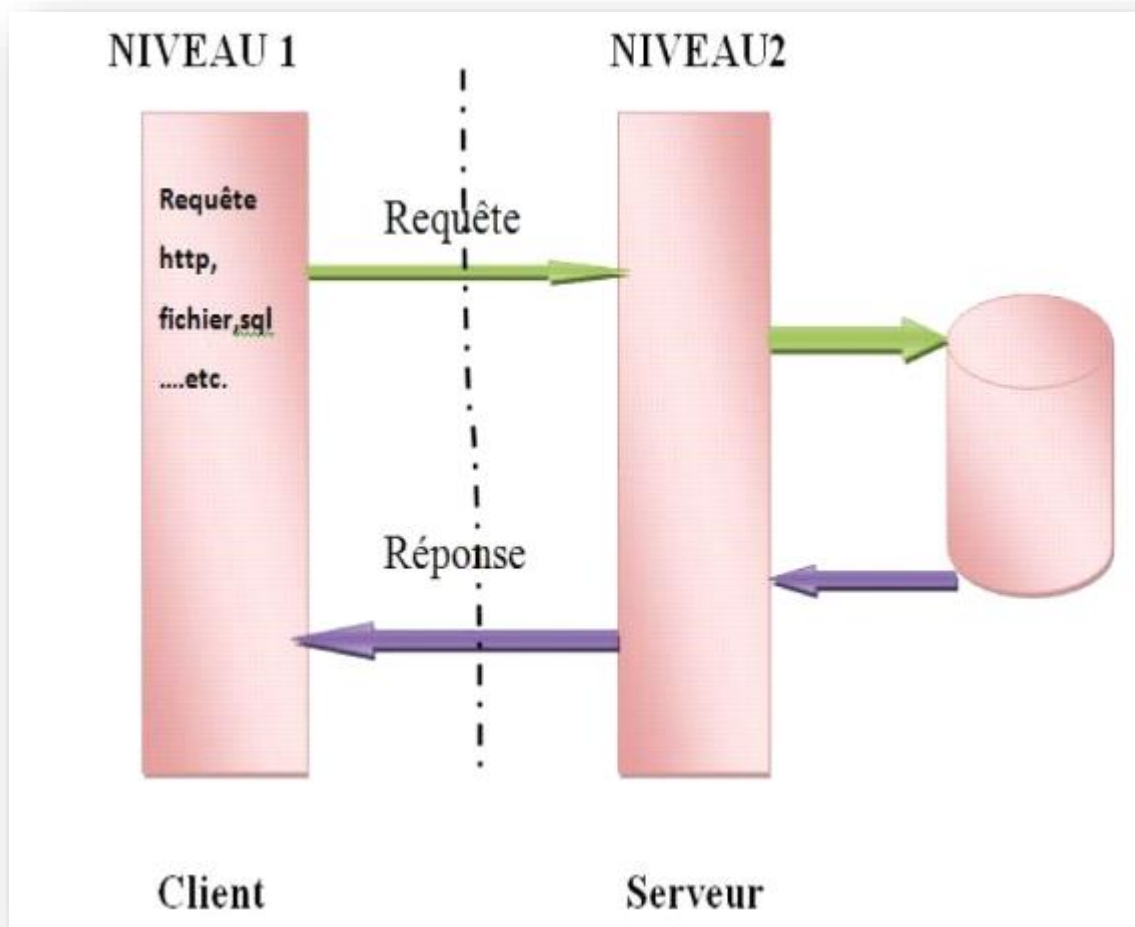


Figure 1.5 : architecture à 2 niveaux

3.2.2. L'architecture client/serveur à 3 niveaux

Dans l'architecture à 3 niveaux (appelées architecture 3-tier), il existe un niveau intermédiaire, c'est-à-dire que l'on a généralement une architecture partagée entre un serveur, le client et un autre serveur secondaire.

- Le client : le demandeur de ressources.
- Le serveur : chargé de fournir la ressource mais faisant appel à un autre serveur.
- Le serveur secondaire : généralement un serveur de base de données, fournissant un service au premier serveur. [7]

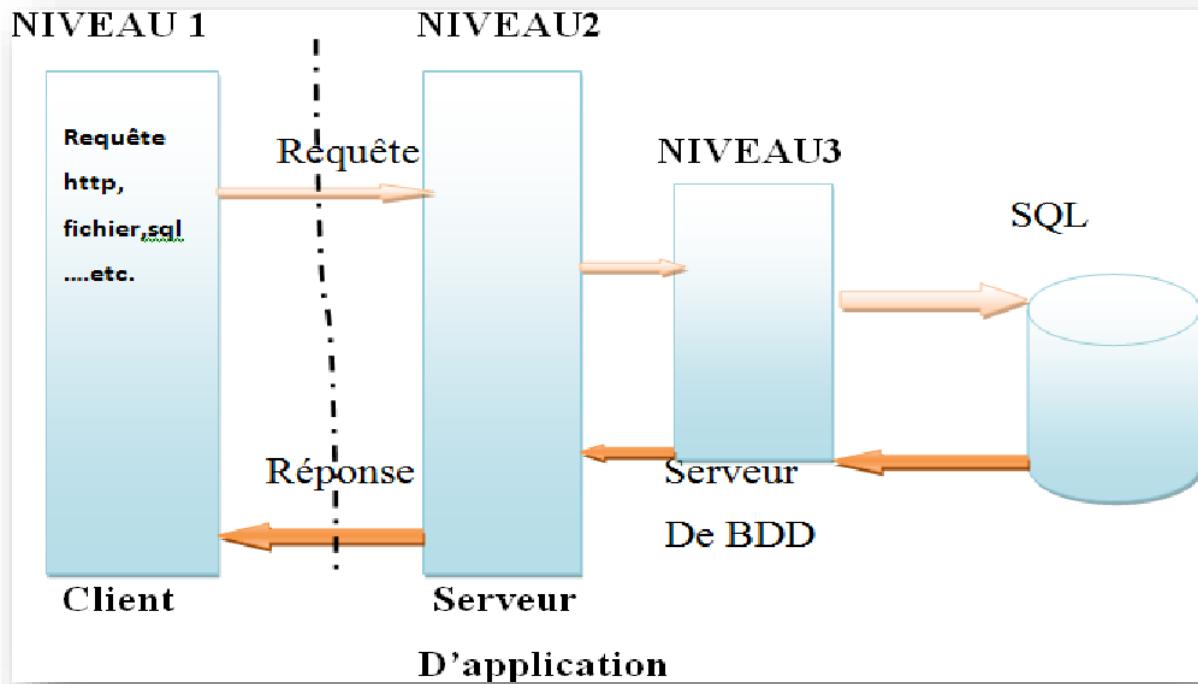


Figure 1.6 : architecture à 3 niveaux.

3.2.3. L'architecture multi-niveaux

Dans l'architecture à 3 niveaux, chaque serveur (niveaux 1 et 2) effectue une tâche (un service) spécialisée. Ainsi, un serveur peut utiliser les services d'un ou plusieurs autres serveurs afin de fournir son propre service. Par conséquent, l'architecture à trois niveaux est potentiellement une architecture à N niveaux. [7]

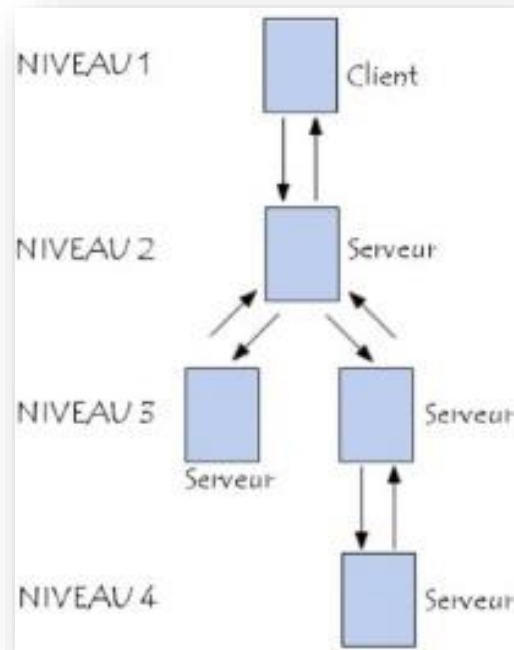


Figure 1.7 : architecture multi niveaux

3.5. Les caractéristique d'un serveur

- Il est initialement passif (ou esclave, en attente d'une requête).
- Il est à l'écoute, prêt à répondre aux requêtes envoyées par des clients.
- Dès qu'une requête lui parvient, il la traite et envoie une réponse. [8]

3.6. Les caractéristique d'un client

- Il est actif le premier (ou maître).
- Il envoie des requêtes au serveur.
- Il attend et reçoit les réponses du serveur. [8]

4. Les applications mobile

Une application mobile est un logiciel applicatif transportable et autonome, développé pour être installé sur un appareil électronique mobile. Elle est identifiée par un ou plusieurs programmes téléchargeables de façon gratuite ou payante depuis un magasin d'applications "Application Store", permettant d'accéder à un contenu homogène et exécutable à partir du

système d'exploitation du Smartphone. Les applications mobiles permettent en général un accès plus pratique, rapide et efficace à des sites en version mobile ou web.

4.1. Les systèmes d'exploitation mobiles

Tout comme un ordinateur dispose d'un système d'exploitation, les téléphones mobiles se composent d'une plate-forme qui contrôle toutes ses fonctionnalités. Ceci est connu comme un système d'exploitation mobile. Généralement connu sous le nom d'OS mobile, il s'agit d'un système d'exploitation qui exploite un appareil mobile (smartphone, tablette, etc.).

Le marché des téléphones mobile est aujourd'hui dominé par cinq grandes entreprises de technologie Smartphone qui sont Apple, RIM, Google, Microsoft et Nokia qui développent respectivement les systèmes d'exploitation iOS, BlackBerry OS, Android, Windows Phone et Symbian OS. On distingue 5 principaux systèmes qui sont : [9]

4.1.1. Android OS (Operating System)

C'est un OS fondé sur un noyau Linux, conçu par Google et l'Alliance. Celui-ci équipe aujourd'hui des appareils très variés : Smartphones, tablettes, montres digitales, autoradios Android est gratuit pour les constructeurs d'appareils souhaitant l'utiliser et partiellement open source. Il a intégré plusieurs services de Google pour accéder rapidement aux services d'internet comme Gmail, YouTube, Google Talk, Google Calen dar et Google Maps. [10]

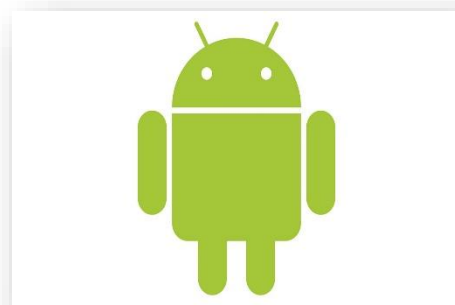


Figure 1.8 : L'icône représentant le système Android.

4.1.2. iPhone (iOS)

Est le système d'exploitation mobile développé par Apple pour l'iPhone. Reconnu pour sa fluidité et son intuitivité ; C'est le système le plus abouti à ce jour iOS se caractérise par une interface peu chargée, qui permet l'exécution de tâches rapidement pour les utilisateurs réguliers, mais qui laisse peu de place à la personnalisation. IOS est dérivé de Mac OS X et donc d'Unix. [10]



Figure 1.9 : L'icône représentant le système i OS.

4.1.3. BlackBerry OS

Est un Système du fabricant canadien RIM (Research In Motion), Il fut le premier à proposer la notification instantanée d'emails, en mode push. Il optimise également l'utilisation mobile en compressant les pages web, ainsi que les pièces jointes des mails. Le mail est donc le point fort des BlackBerry qui a fait son succès auprès des cadres et dirigeants. [10]



Figure 1.10 : L'icône représentant le système BlackBerry OS.

4.1.4. Windows Mobile

Anciennement appelé Windows Phone, est le système d'exploitation conçu par Microsoft en 2010. Il est présent sur les smartphones et tablettes tactiles de Windows. Proposant une interface simple et épurée ainsi que de multiples possibilités de personnalisation, il permet d'arriver rapidement à l'exécution d'une tâche ce qui a particulièrement séduit les utilisateurs. Il est basé sur un noyau Windows CE. [10]



Figure 1.11 : L'icône représentant le système Windows mobiles.

4.1.5. Symbia OS

Il est créé par Nokia en 1998 en compagnie de Panosanic, Psiron, Ericsson et Motorola. Nokia fut ensuite le principal utilisateur de Symbian pendant de nombreuses années pour équiper ses téléphones mobiles et Smartphones et racheta tous les droits du consortium Symbian Ltd en 2008. Il

Offre une plateforme flexible, néanmoins depuis quelques années, la part de marché de Symbian diminue à cause de concurrence avec d'autres plateformes. [10]



Figure 1.12: Exemple d'un téléphone sous Symbian OS.

4.2. Les types des applications mobile

Il existe trois familles d'applications mobiles : les applications natives ou « Natives Apps », les applications Web ou « Web Apps » et, comme évoqué précédemment, les applications hybrides ou « Hybrid Apps ». [5]



Figure1 .13 : Les types de l'application mobile.

4.2.1. Les applications natives

Une application native est une application développée spécifiquement pour une seule plateforme, grâce aux outils conçus pour celle-ci. Elle est développée avec un langage spécifique à son système d'exploitation et est distribuée uniquement par l'intermédiaire de son store (App Store IOS, Play Store pour Android, ...). [9]

4.2.2. Les applications web (HTML)

En opposition à une application native, une Web Apps est une application mobile développée avec les outils de développement web actuel : HTML5, CSS3 et JavaScript.

C'est une application qui une fois développée est accessible et exécutable sur tous les smartphones via leur navigateur web. L'avantage de ces applications, c'est le gain de temps et d'argent réalisé grâce à leur développement unique et leur déploiement multiplateformes. [9]

4.2.3. Les applications hybrides

Les applications hybrides sont des applications qui combinent les éléments d'une Web Apps et les éléments d'une application native. Elles reposent essentiellement sur la solution Cordova/Phone Gap, cette solution sert de passerelle entre le langage web et le natif. Cette solution permet d'utiliser un seul et même outil pour le développement et les langages issus du développement Web pour tous les mobiles (IOS, Android et Windows Phone). [9]

4.3. Les avantages et les inconvénients des applications mobiles

4.3.1. Les avantages

- Une parfaite ergonomie est assurée pour les applications mobiles en comparaison aux sites mobiles cela encourage les utilisateurs à demeurer fidèles aux applis. En effet, le développement d'application mobile tient compte la taille du smartphone, le temps de chargement et autres paramètres.
- Les applications mobiles favorisent l'intégration des options de téléphone et ainsi, l'expérience utilisateur devient plus développée.
- Pas besoin d'avoir accès à l'internet pour que l'application fonctionne.
- Facile à trouver sur les stores par rapport aux sites mobiles, les applications mobiles ont connu ainsi un usage plus répandu auprès des jeunes surtout qu'elles notifient sur les événements en cours. [11]

4.3.2. Les inconvénients

- La soumission aux normes et règles édictées par les sociétés des plateformes mobiles à savoir Apple, Google, Windows et autres.
- Un investissement lourd pour le développement d'une application mobile adaptée à chaque système d'exploitation mobile contrairement au cout qu'exige le développement d'un site mobile.

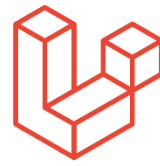
- Lors de toute mise à jour d'application mobile, le mobinaute se trouve dans l'obligation de la faire à travers le store alors que le site mobile se met à jour d'une manière automatique.

[11]

5. Conclusion

Dans ce chapitre, nous avons vu quelque notion sur les technologies web tel que les application web, et son importance dans la facilitée de la communication avec les individus et les programmes, et nous aussi aborder l'ensemble des applications mobiles.

Chapitre 02 : Laravel



1.Introduction

Dans cette section, je vous invite à explorer les fondamentaux de Laravel : où se le procurer, comment l'installer, et quelles configurations sont nécessaires pour le faire fonctionner. Je vais également expliquer son organisation et sa philosophie : comment les requêtes sont orientées par des routes vers des contrôleurs pour traitement, et comment une réponse adéquate est ensuite retournée.

2. Qu'est -ce que laravel ?

C'est un Framework PHP open source qui suit la logique du design pattern MVC et qui propose des outils pour construire une application web. Le créateur de Laravel, Taylor Otwell, a regroupé les bibliothèques les plus performantes pour effectuer chaque fonctionnalité nécessaire à la création d'un site web. Laravel comprend notamment des fonctionnalités exclusives qui sont propres à lui comme le moteur de Template Blade (il s'occupe de séparer tout le code de présentation, tout ce qui est HTML et CSS, donc la VUE du modèle MVC, et le code applicatif c'est à dire le code PHP). [12]

2.1. Pourquoi utiliser Laravel ?

Après avoir répondu à la question « Qu'est-ce que Laravel ? », la prochaine question est « Pourquoi Laravel ? ». Laravel fournit un large éventail d'outils robustes qui facilitent et accélèrent le processus de développement d'applications web. Les bases de code des applications finales sont bien structurées et faciles à maintenir. [13]

2.2. Avantage de Laravel

- ✓ Un Framework fait gagner du temps et donne l'assurance de disposer de composants bien codés et fiables.
- ✓ Laravel est un Framework novateur, complet, qui utilise les possibilités les plus récentes de PHP et qui est impeccablement codé et organisé.
- ✓ Laravel adopte le patron MVC mais ne l'impose pas, il est totalement orienté objet.
- ✓ La documentation de laravel est complète et précise. [14]

2.3. Qu'est-ce qu'un Framework ?

Un Framework est un environnement de travail modulaire constitué de bibliothèques et de conventions, facilitant le développement rapide d'applications. Il offre suffisamment de composants logiciels et impose une rigueur nécessaire pour créer une application complète et facile à maintenir. Ces composants sont conçus pour fonctionner ensemble de manière interactive.

Un Framework propose un ensemble de fonctionnalités simplifiant la création de tout ou partie d'un système logiciel, ainsi qu'un guide architectural qui segmente le domaine visé en modules. Il est généralement implémenté en utilisant un langage orienté objets, bien que cela ne soit pas une obligation. Un Framework objet fournit une structure architecturale en divisant le domaine en classes, en définissant leurs responsabilités et les interactions entre elles. Certaines de ces classes peuvent être abstraites.

2.4. Modèle-Vue-Contrôleur MVC

2.4.1. Qu'est-ce que MVC ?

Le modèle Model-View-Controller a été introduit par Trygve Reenskaug en 1979 comme une solution aux problèmes des utilisateurs qui contrôlent de grandes quantités de données.

MVC est un modèle destiné à répondre aux besoins des applications interactives en séparant les problématiques liées aux différents composants au sein de leurs architectures respectives. Il définit trois couches principales : [9]

✓ La couche Vue

C'est la partie de votre code qui s'occupera de la présentation des données à l'utilisateur, elle retourne une vue des données venant du modèle, en d'autres termes c'est elle qui est responsable de produire les interfaces de présentation de votre application à partir des informations qu'elle dispose (page HTML par exemple). Cependant, elle n'est pas seulement limitée au HTML ou à la représentation en texte des données, elle peut aussi être utilisée pour offrir une grande variété de formats en fonction de vos besoins.

✓ La couche Contrôleur

C'est la couche chargée de router les informations, elle va décider qui va récupérer l'information et la traiter. Elle gère les requêtes des utilisateurs et retourne une réponse avec l'aide de la couche Modèle et Vue.

✓ La couche Modèle

C'est la partie de votre code qui exécute la logique métier de votre application. Ceci signifie qu'elle est responsable de récupérer les données, de les convertir selon les concepts de la logique de votre application tels que le traitement, la validation, l'association et tout d'autre tâche concernant la manipulation des données. Elle est également responsable de l'interaction avec la base de données, elle sait en quelque sorte comment se connecter à une base de données et d'exécuter les requêtes (CREATE, READ, UPDATE, DELETE) sur une base de données. [15]

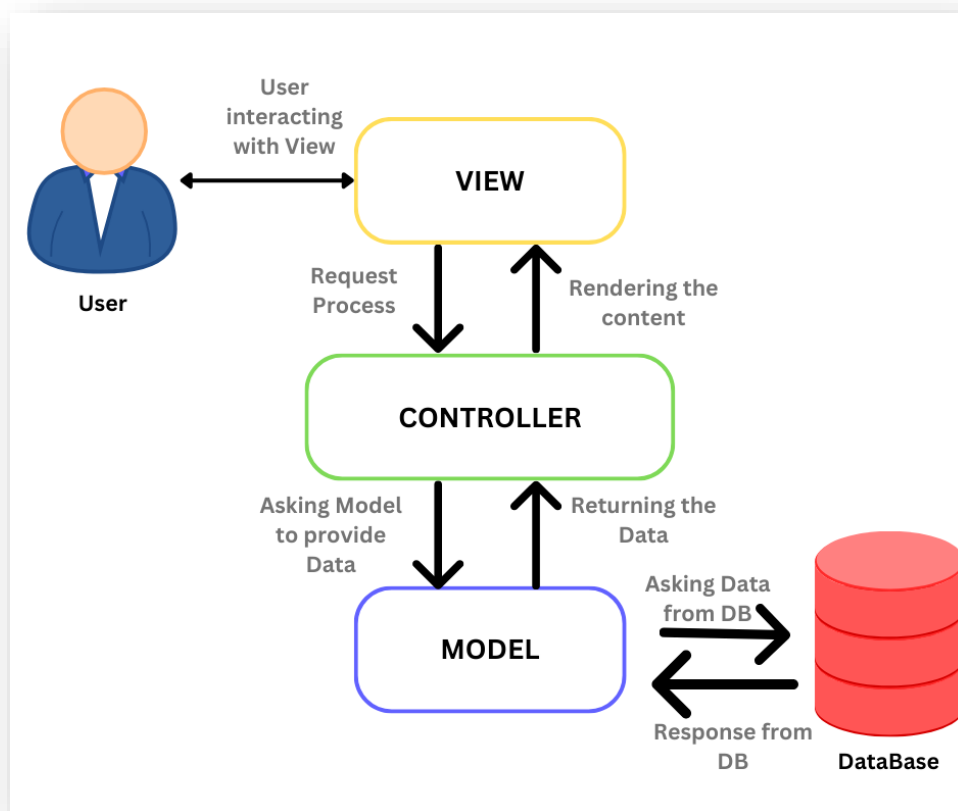


Figure 2.1 : Approche de développement MVC.

2.4.2. Principe de l'architecture MVC

1. Le visiteur envoie sa requête HTTP, la transmet au serveur d'application qui la transmet directement à la partie de notre code qu'on appelle le Contrôleur.

2. Le contrôleur fait le routage de l'information en décidant qui va récupérer l'information et ensuite la traiter, il appelle en effet le Modèle qui contient les informations structurées.

3. Le modèle va effectuer des calculs ou traitement sur ces informations et les renvoie au Contrôleur.

4. Le contrôleur va demander à la vue de générer une vue (Page Web).

5. La vue génère une page web selon la demande du contrôleur et la lui transmet.

6. Le contrôleur reçoit la page web qui sera envoyée au visiteur comme résultat de sa requête. [15]

2.4.3. Avantages de l'utilisation du Framework MVC

• Organisation structurée des projets d'applications web à grande échelle :

Le modèle MVC (Modèle-Vue-Contrôleur) divise l'application en trois composants principaux, ce qui permet de mieux organiser le code. Cette séparation des préoccupations facilite la gestion des projets complexes et améliore la lisibilité et la maintenabilité du code.

• Simplification des modifications :

Grâce à la structure claire du MVC, il est plus facile d'apporter des modifications à l'application. Chaque composant (Modèle, Vue, Contrôleur) a une responsabilité définie, ce qui permet aux développeurs de localiser et de modifier rapidement la partie pertinente du code sans avoir à comprendre l'ensemble du système.

• Isolation des modifications :

Dans un système MVC, les modifications apportées à une partie du code (par exemple, la logique métier dans le Modèle) n'affectent pas les autres parties (comme l'interface

utilisateur dans la Vue). Cette isolation réduit les risques d'introduire des erreurs lors des mises à jour et facilite le développement collaboratif où plusieurs développeurs peuvent travailler simultanément sur différentes parties du projet.

- **Accélération du processus de développement :**

La structure modulaire du MVC permet une réutilisation plus efficace du code et une meilleure collaboration entre les équipes. Par exemple, les designers peuvent se concentrer sur les Vues tandis que les développeurs backend travaillent sur les Modèles et les Contrôleurs. Cela permet de paralléliser les efforts et d'accélérer le cycle de développement, de la conception initiale au déploiement final.

En résumé, l'utilisation du Framework MVC offre une organisation claire et efficace pour les projets web à grande échelle, simplifie les modifications, isole les changements pour minimiser les risques et accélère le processus de développement grâce à une meilleure répartition des tâches et une réutilisation du code plus facile.

2.5. Tout ce dont vous avez besoin pour utiliser Laravel Blade

Laravel Blade est le moteur de modèles par défaut du Framework Laravel. Il vous permet d'utiliser des variables, des boucles, des instructions conditionnelles et d'autres fonctionnalités PHP directement dans votre code HTML. Pour créer des fichiers Blade, il suffit de définir des vues blade en créant des fichiers avec l'extension `blade.php` dans le répertoire `resources/views` de votre projet Laravel, puis de structurer vos pages dynamiques préférées dans ces fichiers. [16]

2.5.1. Pourquoi devriez-vous utiliser Blade ?

L'un des principaux avantages de Blade est son organisation modulaire du code. Blade vous aide à organiser votre code en modules réutilisables que vous pouvez facilement ajouter, supprimer ou mettre à jour sans affecter le reste de votre application.

L'encapsulation du code est un autre avantage de Blade. Blade aide à encapsuler les fonctions, ce qui rend les tests, le débogage et la maintenance du code plus faciles à gérer. Cette approche est avantageuse pour les applications de grande envergure, car les applications non organisées peuvent rapidement devenir difficiles à gérer.

Le moteur de création de modèles de Blade est performant et c'est le Framework PHP le plus rapide . Le moteur compile toutes vos vues Blade en code PHP simple, puis les met en cache jusqu'à ce que vous les modifiez. Cette approche favorise un rendu plus rapide et une meilleure performance globale. [16]

3.comment Installer Laravel ?

Pour installer Laravel suivez les étapes détaillées ci-dessous :

3.1. Installez XAMPP

3.1.1. Definition Xampp

XAMPP est un ensemble de logiciels permettant de mettre en place un serveur Web local, un serveur FTP et un serveur de messagerie électronique. Il s'agit d'une distribution de logiciels libres (**X** (cross) **A**pache **M**aria **D**B **P**erl **P**HP) offrant une bonne souplesse d'utilisation, réputée pour son installation simple et rapide. Ainsi, il est à la portée d'un grand nombre de personnes puisqu'il ne requiert pas de connaissances particulières et fonctionne, de plus, sur les systèmes d'exploitation les plus répandus.

XAMPP permet de configurer un serveur de test local avant la mise en œuvre d'un site Web. [17]

3.1.2. Comment installer Xampp ?

Pour installer XAMPP, suivez ces étapes :

- ✓ Rendez-vous sur la page de téléchargement de XAMPP et téléchargez l'installateur correspondant à votre version de Windows.

- ✓ Exécutez l'installateur et suivez les instructions à l'écran. Pendant l'installation, vous pouvez choisir les composants à installer. Assurez-vous de sélectionner PHP et MySQL.
- ✓ Une fois l'installation terminée, ouvrez le panneau de contrôle XAMPP et démarrez les services Apache et MySQL.

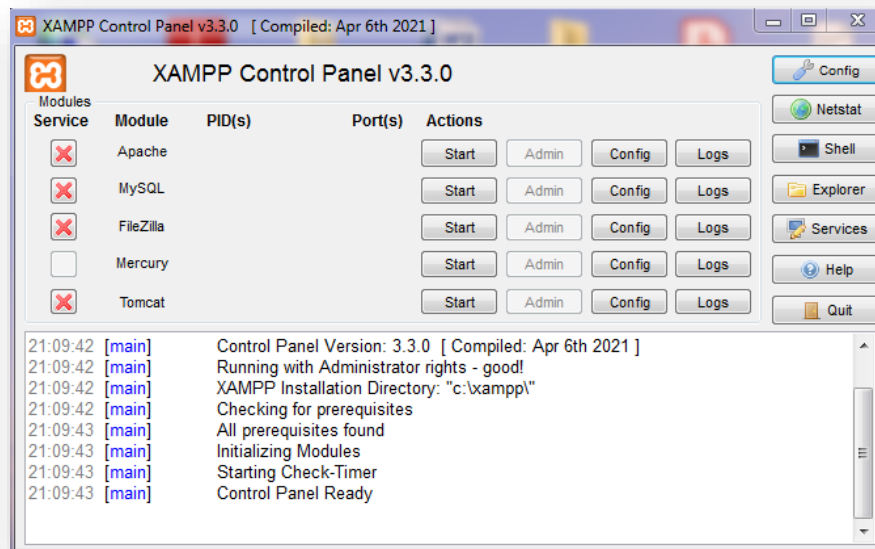


Figure 2.2 : installation de XAMPP.

3.2. Installez Composer

3.2.1. définition composer

Composer est un logiciel gestionnaire de dépendances libre écrit en PHP. Il permet à ses utilisateurs de déclarer et d'installer les bibliothèques dont le projet principal a besoin. Le développement a débuté en avril 2011 et a donné lieu à une première version sortie le 1er mars 2012. Développé au début par Nils Adermann et Jordi Boggiano (qui continuent encore aujourd'hui à le maintenir), le projet est maintenant disponible sur la plateforme GitHub. Il est ainsi développé par toute une communauté.

Le logiciel Composer est à l'initiative d'un portage en PHP du logiciel Libazyp satsolver d'Open Suse. [18]

3.2.2. Comment installer composer ?

❖ Installation :

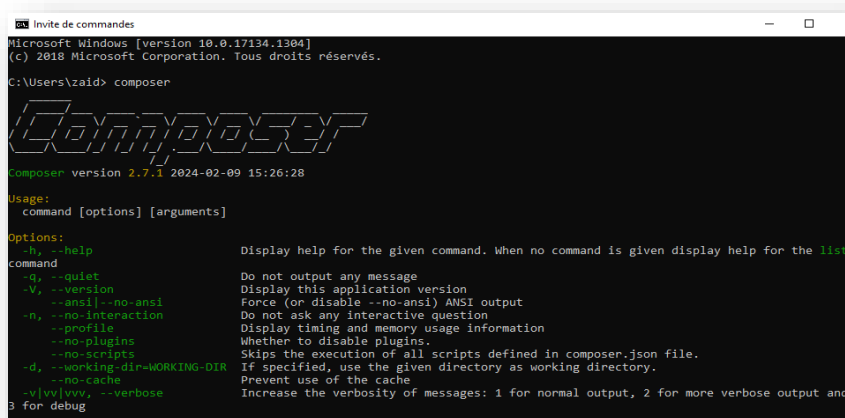
Pour installer Composer, suivez ces étapes :

Laravel utilise Composer comme gestionnaire de dépendances. Pour l'installer sur votre ordinateur :

- ✓ Pour Windows, téléchargez l'installateur depuis la page officielle (<https://getcomposer.org/download/>).
- ✓ Exécutez l'installateur, qui se chargera de tout configurer proprement et d'ajouter Composer à la variable d'environnement PATH, le rendant ainsi utilisable depuis n'importe quel emplacement.
- ✓ Lors de l'installation, l'installateur vous demandera de spécifier l'emplacement de php.exe car Composer est un fichier PHP et nécessite PHP pour s'exécuter.

3.2.3. Vérifiez l'installation de Composer

Pour vérifier que Composer a été installé correctement, ouvrez l'invite de commande.



```
Microsoft Windows [version 10.0.17134.1304]
(c) 2018 Microsoft Corporation. Tous droits réservés.

C:\Users\zaid> composer

Composer

composer version 2.7.1 2024-02-09 15:26:28

Usage:
  command [options] [arguments]

Options:
  -h, --help                Display help for the given command. When no command is given display help for the list
  -q, --quiet                Do not output any message
  -V, --version              Display this application version
  --ansi|--no-ansi          Force (or disable --no-ansi) ANSI output
  -n, --no-interaction       Do not ask any interactive question
  --profile                  Display timing and memory usage information
  --no-plugins               Whether to disable plugins.
  --no-scripts               Skips the execution of all scripts defined in composer.json file.
  -d, --working-dir=WORKING-DIR If specified, use the given directory as working directory.
  --no-cache                 Prevent use of the cache
  -v|vv|vvv, --verbose      Increase the verbosity of messages: 1 for normal output, 2 for more verbose output and
                             3 for debug
```

Figure 2.3 : vérifiez l'installation de composer.

3.3. Installez Laravel à l'aide de Composer

Pour installer Laravel à l'aide de composer, suivez ces étapes :

Ouvrez l'invite de commande et exécutez la commande suivante

Composer create-project laravel/laravel app-name

Cette commande téléchargera automatiquement tous les fichiers nécessaires à la création d'un nouveau projet Laravel.

3.3.1. Démarrez le serveur

Démarrez le serveur en naviguant dans le répertoire de votre projet et en exécutant la commande suivante :

```
Cd app-name
```

```
Php artisan serve
```

3.3.2. Exécutez le projet dans votre navigateur

Pour vérifier si l'installation a réussi. Ouvrez Google chrome ou Firefox, et rendez-vous à l'adresse suivante : <https://localhost:8000>

Si tout a bien fonctionné, vous devriez voir le logo de Laravel en plein centre de la page.

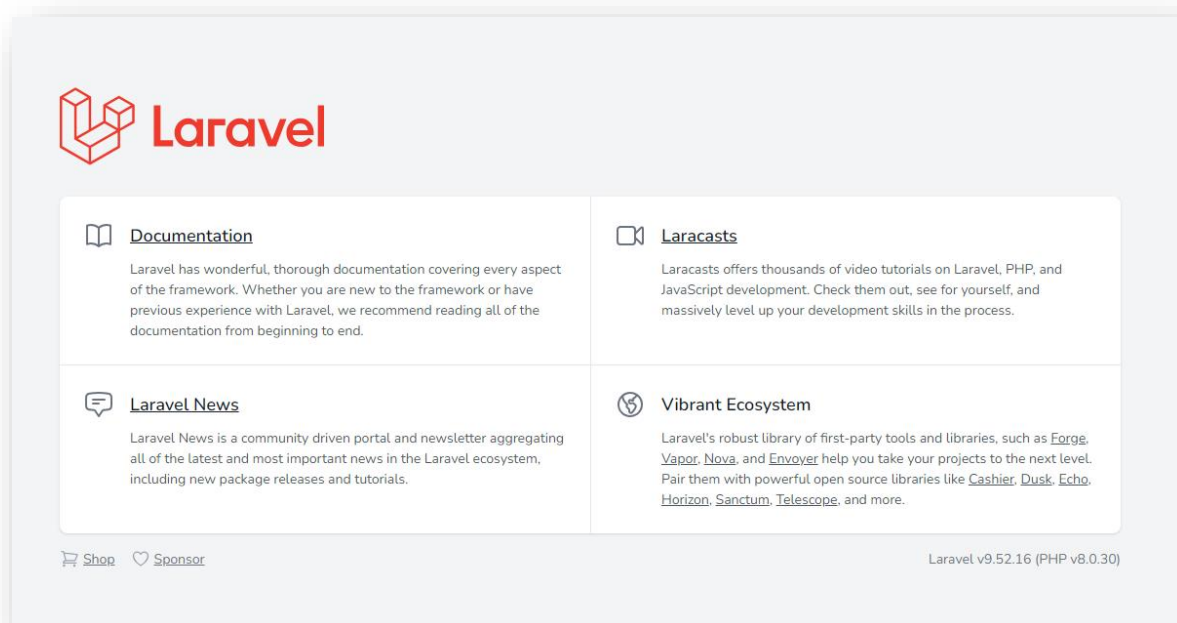


Figure 2.4 : Page d'accueil de Laravel.

Pour finalement se retrouver avec l'architecture suivante.

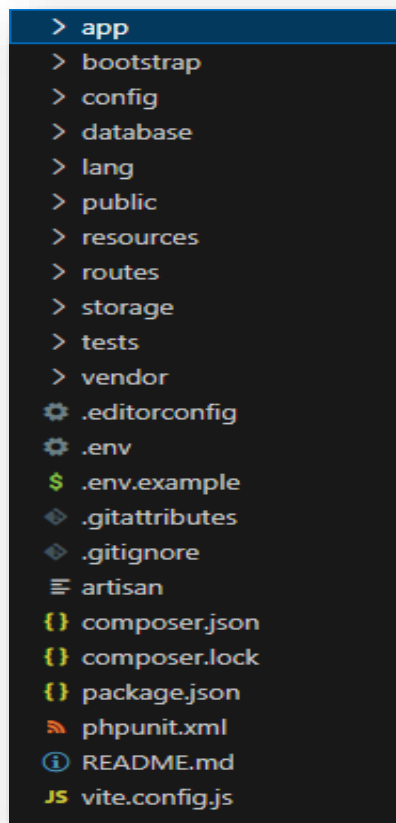


Figure 2.5 : Architecture des dossiers de Laravel.

4. Structure du projet

Une application Laravel contient plusieurs dossiers à sa racine. Connaître leur contenu permet de mieux appréhender le futur développement et d'améliorer la compréhension du fonctionnement du Framework.

A la racine de notre projet, on a un dossier App qui va contenir le code de notre application. Ce dossier sera aussi la racine du Namespace \App. A l'intérieur de ce dossier-là, on trouvera différents dossiers qui correspondent à des classes préconçues par Laravel, mais on aura l'occasion de revenir dessus lorsque l'on va commencer à écrire nos premières lignes de code. Pour l'instant, il faut juste savoir que c'est dans ce dossier là que vous allez écrire la plupart de votre code.

Le dossier bootstrap n'est pas forcément un dossier que vous allez toucher, mais il contient le code qui va permettre de faire démarrer une application Laravel.

Le dossier config qui lui va être un dossier important qui va contenir la configuration des différents éléments du Framework. Par exemple, le fichier "mail. PHP" permettra de configurer le système de mailing, le fichier "cache. PHP" permettra de configurer le cache et on a comme ça un fichier par type de configuration que l'on peut faire. A l'intérieur de ces fichiers la configuration est donnée sous forme de tableau associatif.

Le dossier data base contiendra les différents éléments concernant la base de données. A l'intérieur, on a trois dossiers, que l'on évoquera plus tard.

Le dossier public servira de racine pour notre application. Lorsqu'on créera un serveur web, c'est ce dossier qui sera utilisé comme racine pour faire fonctionner notre application. Tout ce qui est dans le dossier "public" sera, comme son nom l'indique, publiquement accessible et on retrouve à l'intérieur le fichier "index. PHP" qui sera le véritable point d'entrée de Laravel.

Le **ressources** contient des ressources CSS et JavaScript qui serviront au front-end, mais il va aussi contenir un dossier **view** qui permettra de spécifier les fichiers qui vont permettre de générer les pages HTML.

Le dossier **routes** va contenir les chemins de notre application. C'est ce qui permet de dire "lorsque je vais sur la page d'accueil, voilà ce que je dois faire". Donc, on a différentes catégories de routes en fonction de ce que l'on peut faire. Les routes pour la partie API, pour la partie application web, pour la console, et les charnels pour les web sockets.

Enfin, le dossier **Storage** va contenir les fichiers générés par l'application (cache et téléversement utilisateur). [19]

4.1. Routage et contrôleurs

La fonction essentielle de tout Framework d'application web est de répondre aux requêtes des utilisateurs et de fournir des réponses, généralement via HTTP. Ainsi, la définition des itinéraires d'une application est le premier et le plus important aspect à aborder lors de l'apprentissage d'un Framework web. Sans itinéraires, il est impossible d'interagir avec l'utilisateur final.

Dans cette section, nous examinerons les routes dans Laravel et expliquerons comment les définir, les lier au code qu'elles doivent exécuter, et utiliser les outils de routage de Laravel pour répondre à divers besoins de routage.

4.1.1. Définition des routes

Les routes dans laravel sont définies dans le dossier 'routes' les principaux fichiers sont :

- 'Web. Php' : pour les routes accessibles via le navigateur.
- 'api. Php' : pour les routes destinées aux API.

```
routes > web.php
10 |
11 | Here is where you can register web routes for your application. These
12 | routes are loaded by the RouteServiceProvider and all of them will
13 | be assigned to the "web" middleware group. Make something great!
14 |
15 | */
16 |
17 | Route::get('/', function () {
18 |     return view('welcome');
19 | });
```

Figure 2.6 : Exemple les route.

4.1. 2. Definition des Contrôleurs

Au lieu de définir toute votre logique de gestion des requêtes comme des closures dans vos fichiers de routage, vous pouvez organiser ce comportement à l'aide de classes "Controller". Les contrôleurs peuvent regrouper la logique de gestion des requêtes associées dans une seule classe. Par exemple, une classe User Controller peut gérer toutes les demandes entrantes liées aux utilisateurs, y compris l'affichage, la création, la mise à jour et la suppression d'utilisateurs. Par défaut, les contrôleurs sont stockés dans le répertoire app/Http/Controller.

Pour créer un contrôleur, taper ligne de commande suivante

```
php artisan make:controller IndexController
```

5. Eloquent Laravel

5.1. Qu'est-ce que Laravel Eloquent ?

Le Framework PHP Laravel est fourni avec Eloquent Object Relational Mapper (ORM), qui fournit un moyen extrêmement simple de communiquer avec une base de données. Comme les développeurs doivent créer des sites Web complexes et d'autres applications, ils préfèrent un temps de développement plus court et sans tracas. Laravel contribue à accélérer le développement et apporte une solution adéquate à la plupart des problèmes rencontrés. Les différentes exigences commerciales sont satisfaites grâce à un développement plus rapide, ainsi qu'à un code bien organisé, réutilisable, maintenable et évolutif. Il fonctionne avec des applications Web personnalisées car il peut gérer plusieurs bases de données et effectuer des opérations de base de données courantes. [20]

5.2. Configuration d'un ORM éloquent

Avant de pouvoir commencer à utiliser Eloquent ORM, vous devez configurer votre connexion à la base de données dans les fichiers de configuration de Laravel. Laravel prend en charge plusieurs connexions à la base de données et vous pouvez les configurer dans le `config/database.php` fichier. Une fois votre connexion à la base de données configurée, vous pouvez commencer à créer des modèles Eloquent pour représenter vos tables de base de données, suggérez une des principales. [21]

5.3. Que sont les relations éloquentes ?

Les relations éloquentes font référence aux relations entre les modèles dans Eloquent, l'ORM (Object-Relational Mapping) intégré au Framework Laravel. Eloquent facilite la définition et la gestion des relations entre les tables de la base de données à travers des modèles, permettant aux développeurs de manipuler les données de manière intuitive et orientée objet.

Types de relations éloquentes

- **Relation un-à-un (One-to-One) :**

Cette relation signifie qu'un enregistrement dans une table est associé à un seul enregistrement dans une autre table.

- **Relation un-à-plusieurs (One-to-Many) :**

Cette relation signifie qu'un enregistrement dans une table peut être associé à plusieurs enregistrements dans une autre table

- **Relation plusieurs-à-plusieurs (Many-to-Many) :**

Cette relation signifie que plusieurs enregistrements dans une table peuvent être associés à plusieurs enregistrements dans une autre table. Cette relation nécessite une table pivot pour gérer les associations.

- **Relation un-à-plusieurs à travers (Has Many Through) :**

Cette relation permet de définir une relation un-à-plusieurs à travers un autre modèle.

- **Relation polymorphe (Polymorphic Relations) :**

Les relations polymorphes permettent un modèle de relation dynamique où un modèle peut appartenir à plusieurs autres modèles en utilisant une seule association.

5.4. Avantages des relations éloquentes

- ✓ **Simplicité et clarté** : Les relations éloquentes permettent de définir les relations entre les modèles de manière simple et compréhensible, rendant le code plus lisible.
- ✓ **Requêtes efficaces** : Eloquent optimise les requêtes SQL en fonction des relations définies, améliorant ainsi les performances de l'application.
- ✓ **Manipulation aisée des données** : Grâce aux relations éloquentes, il est facile de récupérer, modifier et associer des données entre différentes tables sans écrire de SQL complexe.

En résumé, les relations éloquentes dans Laravel permettent de gérer les associations entre les modèles de manière intuitive et efficace, en utilisant des conventions orientées objet qui simplifient le développement d'applications complexes.

5.5. base de données

Une base de données, également appelée BDD ou database en anglais, est un système qui regroupe un ensemble d'informations organisées de manière à être facilement

accessibles, gérables et mises à jour par ses utilisateurs. Les données y sont structurées en lignes, colonnes et tableaux, et sont indexées pour faciliter les recherches. Les entreprises s'en servent pour stocker, gérer et récupérer des informations.

5.5.1. Les migrations

Une migration permet de créer et de mettre à jour un schéma de base de données. Autrement dit, vous pouvez créer des tables, des colonnes dans ces tables, en supprimer, créer des index... Tout ce qui concerne la maintenance de vos tables peut être pris en charge par cet outil. Vous avez ainsi un suivi de vos modifications. [22]

5.5.2. Créer une base de données

Pour créer une nouvelle base de données pour votre application :

1. Démarrez les serveurs Apache et MySQL dans le panneau de contrôle XAMPP et visitez <http://localhost/phpmyadmin> dans votre navigateur.

2. Cliquez sur Nouveau dans la colonne latérale de gauche. Vous devriez voir ce qui suit :



Figure :2.7. Le formulaire de création de base de données.

1. Ajoutez un nom de base de données et cliquez sur **Créer**.

2. Modifiez le fichier `env` de votre application à la racine de votre application Laravel. Il contient toutes les variables d'environnement utilisées par l'application. Localisez les variables préfixées par `DB_` et modifiez-les avec les informations d'identification de votre base de données : [23]

```
DB_CONNECTION=MySQL
```

```
DB_HOST=127.0.0.1
```

```
DB_PORT=3306
DB_DATABASE=nom DB
DB_USERNAME=root
DB_PASSWORD=
```

5.5.2 Créer une table

Les lignes de données de votre application sont stockées dans des tables. Pour cette application, vous n'avez besoin que d'une seule table, créée à l'aide des migrations Laravel.

1. Pour créer une table et générer un fichier de migration à l'aide de l'interface en ligne de commande de Laravel, Artisan, exécutez :

```
php artisan make:migration create_posts_table
```

La commande ci-dessus crée un nouveau fichier, `app/Models/Post.php`. Dans `database/migrations`.

2. Ouvrez le fichier pour définir les colonnes que vous voulez dans votre table de base de données dans la fonction `up` :

```
public function up ()
{
    Schema::create ('posts', function (Blueprint $table) {
        $table->id ();
        $table->string ('title');
        $table->text ('body');
        $table->timestamps ();
    });
}
```

Figure .2.8 : Exemple basique de Code de la table posts.

Ce code définit le contenu de la table `posts`. Elle comporte quatre colonnes : `id`, `title`, `body`, et `timestamps`.

3. Exécutez les fichiers de migration dans le dossier `database/migrations` pour créer des tables dans la base de données :

```
php artisan migrate
```

Le résultat ressemble à ceci :

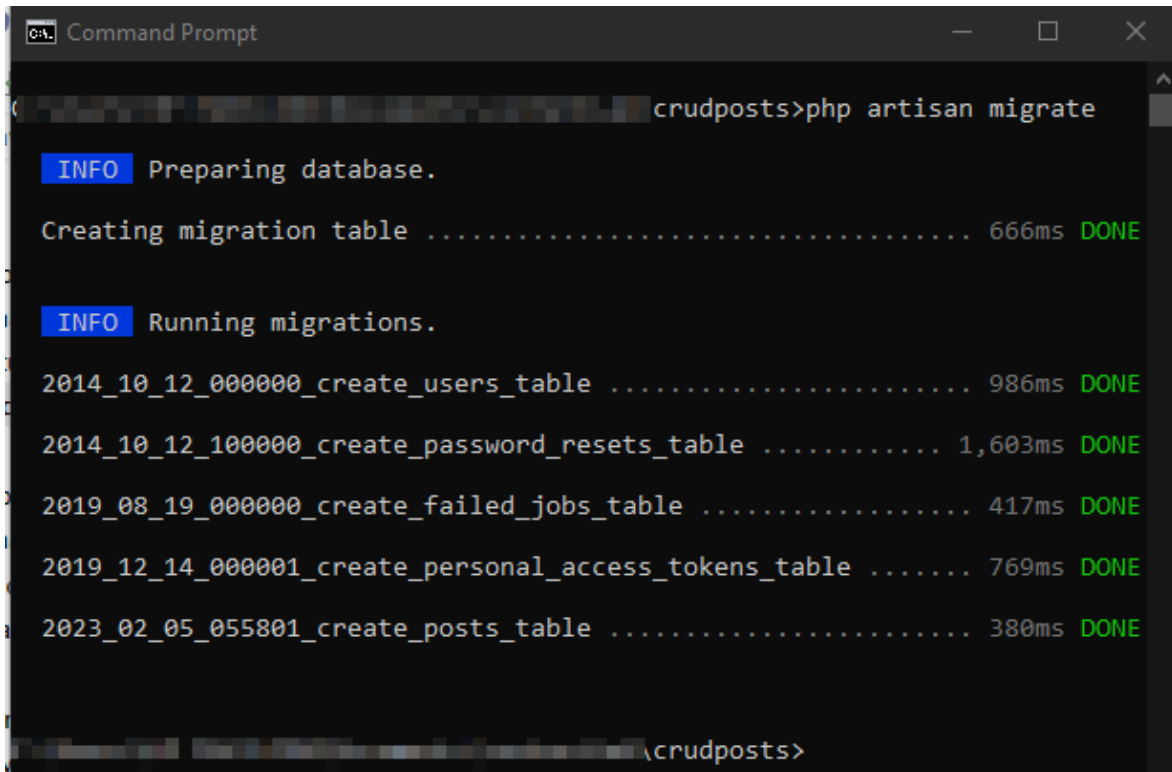


Figure :2.9. Exécution des migrations.

4. Accédez à la base de données que vous avez créée précédemment pour confirmer que vous avez créé les tables : **[23]**

Table	Action	Rows	Type	Collation	Size	Overhead
<input type="checkbox"/> failed_jobs	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	32.0 K1B	-
<input type="checkbox"/> migrations	★ Browse Structure Search Insert Empty Drop	5	InnoDB	utf8mb4_unicode_ci	16.0 K1B	-
<input type="checkbox"/> password_resets	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	16.0 K1B	-
<input type="checkbox"/> personal_access_tokens	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	48.0 K1B	-
<input type="checkbox"/> posts	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	16.0 K1B	-
<input type="checkbox"/> users	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	32.0 K1B	-
6 tables	Sum	5	InnoDB	utf8mb4_general_ci	160.0 K1B	0 B

Figure :2.10. Tables créées.

6. Laravel Middleware

6.1. Qu'est-ce qu'un middleware ?

Le middleware utilise un mécanisme pour filtrer les requêtes HTTP entrant dans l'application. Par exemple, dans Laravel, il existe un middleware par défaut qui authentifie les utilisateurs. Ce middleware vérifie l'authentification de l'utilisateur : si l'utilisateur n'est pas authentifié, il est redirigé automatiquement vers l'écran de connexion. En revanche, si l'utilisateur est authentifié, la requête est autorisée à poursuivre son chemin dans l'application. En plus de l'authentification, il est possible de créer des middlewares supplémentaires pour accomplir diverses tâches. Laravel inclut plusieurs middlewares intégrés, tous situés dans le répertoire `app/Http/Middleware`. Habituellement, un développeur peut modifier ces middlewares existants ou en créer de nouveaux pour répondre aux besoins spécifiques de l'application.

6.2. Importance du middleware dans le Framework Laravel

Dans Laravel, le middleware joue un rôle important dans la configuration du flux de requêtes HTTP entrantes et de réponses sortantes. La couche middleware de Laravel se situe entre la requête entrante et les gestionnaires de routage de l'application, offrant aux développeurs un mécanisme robuste pour injecter une logique personnalisée à différentes étapes du cycle de vie de la requête.

L'un des principaux atouts du middleware Laravel réside dans sa flexibilité et sa polyvalence. Qu'il s'agisse de mettre en œuvre des mécanismes d'authentification et d'autorisation, de gérer la protection contre la falsification de requêtes intersites (CSRF), de journaliser les requêtes ou d'effectuer des opérations personnalisées, le middleware permet aux développeurs d'adapter le processus de traitement des requêtes en fonction des exigences spécifiques de leurs applications. [24]

6.3. Explication du middleware dans le développement Web

Le middleware, dans son essence, fait référence à des composants logiciels situés entre différentes couches d'une application, facilitant la communication et l'échange de données. Dans le développement Web, le middleware intercepte les requêtes et les réponses HTTP, permettant aux développeurs d'effectuer diverses tâches telles que l'authentification,

l'autorisation, la journalisation, etc., avant d'atteindre le gestionnaire ou contrôleur de route prévu.

Le middleware fonctionne sur le principe du modèle de conception « chaîne de responsabilité », dans lequel chaque composant middleware de la chaîne a la capacité de modifier la demande, de la transmettre au middleware suivant ou d'arrêter complètement la demande. Cette approche modulaire favorise non seulement la réutilisabilité du code, mais permet également aux développeurs de compartimenter les préoccupations, rendant ainsi les applications plus maintenables et extensibles. [24]

7. Notifications Laravel

7.1. Qu'est –ce qu'un Notifications ?

Les notifications peuvent être considérées comme des messages brefs et directs envoyés aux utilisateurs pour les informer d'informations et d'événements importants ou pour provoquer une réponse dans l'application. Idéalement, ils tiennent les utilisateurs informés et augmentent leur engagement. Laravel prend en charge l'envoi de notifications via une variété de canaux. Par défaut, il inclut les canaux de messagerie , Slack , de base de données , de diffusion et Vonage . En utilisant la commande Laravel Artisan, vous pouvez créer rapidement des notifications. Vous pouvez également personnaliser les détails de la notification. [25]

7.2. Canaux de notification de Laravel

Laravel vous permet de choisir parmi différents canaux de notification pour envoyer des notifications dans votre application. Vous pouvez utiliser plusieurs canaux.

Mail : Ces notifications sont envoyées par e-mail aux utilisateurs.

SMS : Les utilisateurs reçoivent ces notifications SMS sur leurs téléphones mobiles.

Base de données : Ces notifications sont stockées dans la base de données et vous pouvez les afficher à l'utilisateur avec une interface utilisateur personnalisée. [25]

8. Laravel Breeze

8.1. Qu'est-ce que laravel Breeze ?

Laravel Breeze est un package qui simplifie le processus d'ajout de fonctionnalités d'authentification au site web ou à l'application web. Il offre un moyen rationalisé de gérer efficacement les processus d'enregistrement, de connexion et de déconnexion des utilisateurs.

Les services de développement professionnels de Laravel implémentent Breeze pour gagner du temps et des efforts. Ils exploitent ses vues et routes d'authentification prédéfinies. Ce package garantit des pratiques d'authentification sécurisées telles que le hachage de mot de passe et la gestion de session. Et vous n'avez pas besoin d'effectuer une configuration manuelle.

Laravel Breeze possède également une conception cohérente et s'intègre assez efficacement au reste de l'écosystème. Les développeurs peuvent ainsi se concentrer sur la

Création de leurs applications plutôt que de se soucier des flux de travail d'authentification. [26]

8.2. Pourquoi choisir Laravel Breeze ?

Il y a plusieurs raisons pour lesquelles vous devriez envisager d'utiliser Laravel Breeze pour l'authentification dans vos projets Laravel :

Simplicité : Breeze offre un moyen simple et intuitif d'intégrer la fonctionnalité d'authentification dans votre application. Il élimine le besoin d'une installation et d'une configuration complexes, ce qui vous permet d'être rapidement opérationnel.

Flexibilité : Breeze propose plusieurs options frontend pour répondre à vos préférences. Vous pouvez choisir entre Blade, Livewire ou React/Vue pour la pile frontale de votre application. Cette flexibilité vous permet de travailler avec la technologie avec laquelle vous êtes le plus à l'aise.

Personnalisation : Breeze vous permet de personnaliser les vues d'authentification, les itinéraires et les contrôleurs en fonction de vos besoins spécifiques. Vous avez un contrôle

total sur l'apparence et le comportement des fonctionnalités d'authentification, garantissant une intégration transparente avec le reste de votre application. [27]

8.3. Installer Laravel Breeze [28]

Avant d'installer laravel/breeze, nous devons nous assurer d'avoir importé la table « `users` » ou migré `database/migrations/..._create_users_table.php` dans la base de données en exécutant la commande `php artisan migrate`. Breeze utilise la table `users`.

```
Schema:: create ('users', function (Blueprint $table) {
    $table->id ();
    $table->string('name');
    $table->string('email')->unique ();
    $table->timestamp('email_verified_at')->nullable ();
    $table->string('password');
    $table->remember Token ();
    $table->timestamps ();
});
```

Figure :2.11 : Exemple basique code de la table users.

Pour installer laravel/breeze dans un projet Laravel, on exécute la commande composer suivante :

```
Composer require laravel/breeze
```

Une fois laravel/breeze installé avec composer, on l'implémente dans l'application en exécutant la commande artisan suivante :

```
Php artisan breeze: install
```

Cette commande installe les contrôleurs, les vues, les routes et d'autres ressources de Breeze pour l'authentification des utilisateurs.

Les routes

Breeze insère la route nommée « Dashboard » (tableau de bord) puis inclut le fichier `/routes/auth.php` où sont définis ses routes dans le fichier `/routes/web.php`

```
// Les routes de laravel/breeze

Route::get ('/dashboard', function () {
    return view('dashboard');
})->middleware(['auth'])->name('dashboard');
require __DIR__.'/auth.php';
```

Figure :2.12 : les routes de laravel breeze.

Pour finalement installer puis compiler les assets CSS et JavaScript, on exécute les commandes npm suivantes :

```
npm install
npm run dev
```

Ceci est l'écran initial après une installation réussie et terminée de Laravel Breeze.

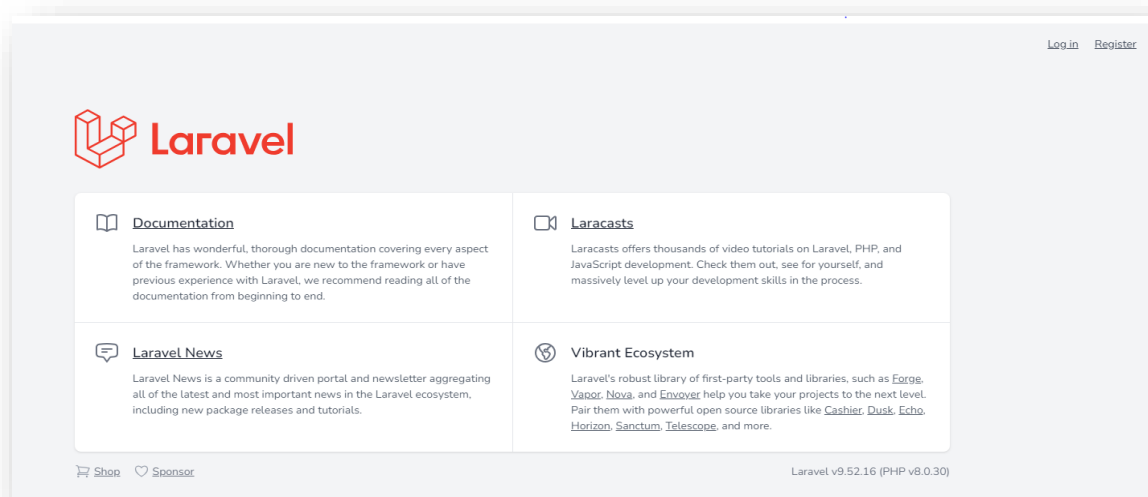


Figure : 2.13 : l'écran initial après installation Breeze.

En utilisant Laravel Breeze, vous obtiendrez une page de connexion, d'enregistrement et de réinitialisation du mot de passe, sans avoir à créer de pages et leur logique.

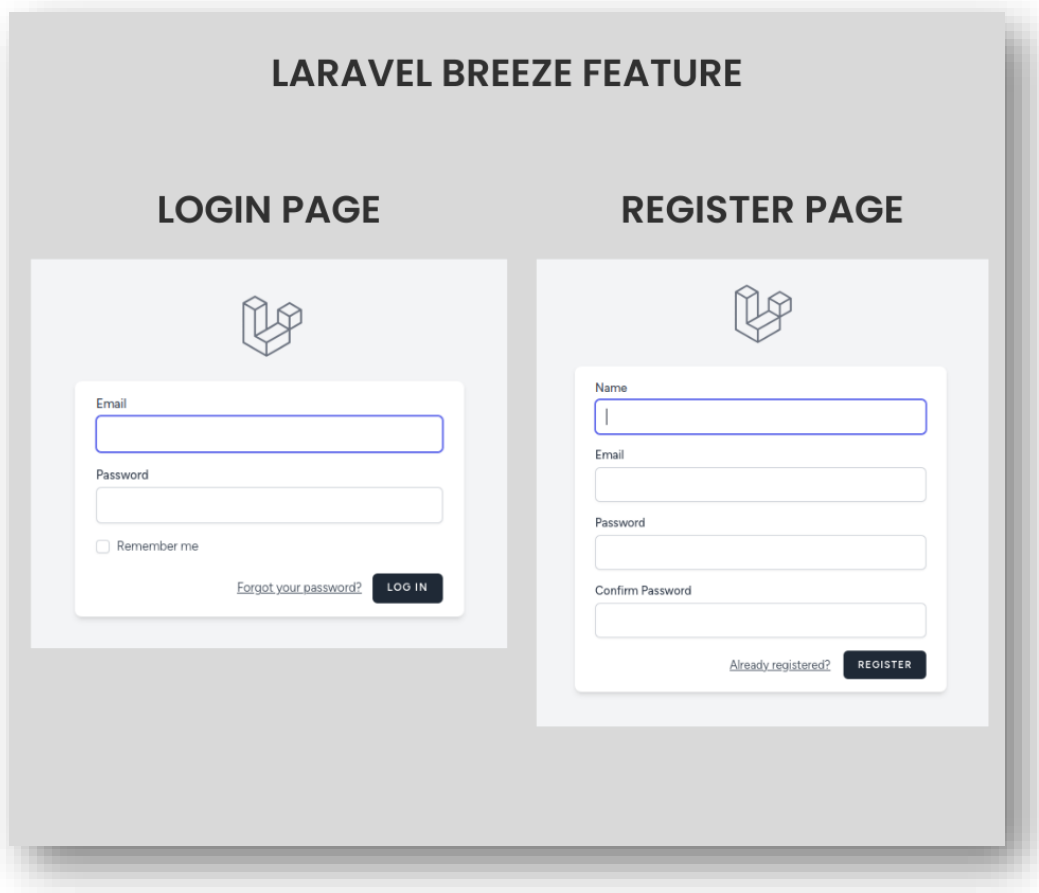


Figure : 2.14 : l'interface de connexion et d'inscription.

Voici l'affichage de la page du tableau de bord une fois que vous vous êtes Inscrit et Connecté avec succès.

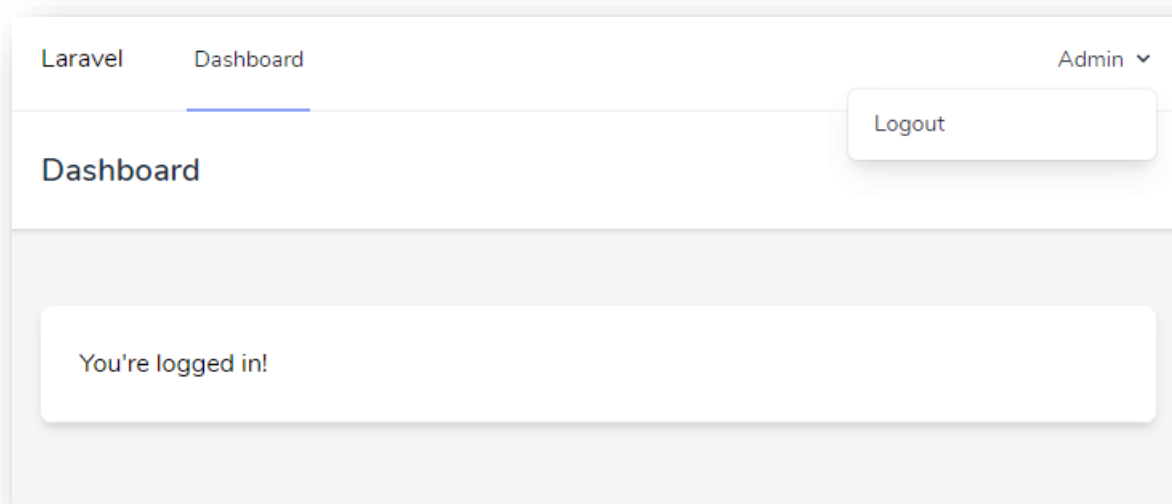


Figure. 2.15 : L'interface de page du tableau de bord.

9. Conclusion

Pour conclure, dans cette partie nous avons vu comment créer un nouveau projet avec le Framework Laravel. Cet outil utilise le système MVC (Modèle, Vue, Controller) Il utilise également le système de migrations afin de pouvoir utiliser une base de données.

Chapitre03 :Analyse et Conception

1.Introduction

Dans ce chapitre, nous débuterons par une présentation de notre projet, puis nous passerons à la phase d'Analyse, qui mettra en lumière les différents intervenants de l'application ainsi que leurs besoins. Ensuite, nous aborderons la phase de Conception, laquelle, en s'appuyant sur l'analyse précédente, fournira la modélisation des objectifs à atteindre. Ainsi, notre approche se fondera sur le langage UML, ce qui nous permettra de représenter de manière précise les aspects fonctionnels, statiques et dynamiques de notre projet à travers la série de diagrammes qu'il propose.

2. La méthode UML

2. 1. Définition

UML (Unified Modeling Language) permet de présenter et de manipuler les concepts objet, et de faire une démarche d'analyse qui permet de concevoir une solution de manière itérative grâce aux diagrammes, et d'exprimer visuellement une solution objet. Il se caractérise comme un langage de modélisation graphique et textuel qui est une étape importante du cycle de développement des systèmes utilisé ainsi pour visualiser, comprendre et définir des besoins, spécifier et construire les documents nécessaires au bon développement d'un logiciel orienté objet, esquisser des architectures logicielles, concevoir des solutions et communiquer des points de vue. Ces modèles doivent être proche de la réalité. [29]

2.2. Pourquoi la méthode UML ?

Après l'étude de quelques méthodes. On a trouvé que « UML » est la meilleure. Elle offre

Plusieurs avantages parmi nous citons :

- UML n'est pas une méthode ou un processus !
- Si l'on parle de méthode d'objet pour UML, c'est par abus du langage !
- UML a été pensé pour permettre de modéliser les activités de l'entreprise.
- UML est support de communication (un langage universel).
- Sa notation graphique permet d'exprimer visuellement une solution objet.

- L'aspect formel de sa notation limite les ambiguïtés et l'incompréhension.
- Son aspect visuel facilite la comparaison et l'évaluation de solution.
 - Penser objet avec UML, pour concevoir objet.
- Pour penser et concevoir objet, il faut savoir prendre de la hauteur, jonglé avec des concepts abstraits indépendante du langage d'implémentation et des contraintes purement technique.
- UML comble une lacune importante des technologies. Il permet d'exprimer et d'élaborer des modèles objet, indépendamment de tous les langages de programmation.
 - UML : le chemin vers l'unification des processus.
- Guidée par les besoins des utilisateurs de système.
- Centre sur l'architecture logicielle.
- Interactive et incrémentation. [30]

2.3. Principe UML

Les 3 experts, Grady Booch, James Rumbaugh et Ivar Jacobson, ont focalisé leur attention sur les deux aspects : modélisation et formalisation afin de concevoir un langage de modélisation standard et universel utilisé notamment pour le développement informatique en langage objet.

UML 2 est une évolution majeure du langage. UML 2.5, la dernière spécification, est supportée par l'OMG Object Management Group. La modélisation et la formalisation à l'aide d'un vocabulaire standardisé, et de surcroît orienté objet, confèrent à la méthode tout son intérêt.

En effet, la formalisation et la modélisation facilitent la définition du problème à traiter et la compréhension par l'ensemble des principales parties prenantes après, il est vrai, un court apprentissage.

Une fois le modèle bien défini, il est plus aisé de s'y référer lors du développement afin de s'assurer de la conformité de ce dernier. Un outil précieux qui explique à lui seul l'essor de la démarche UML. [31]

2.4. Définition d'un Diagramme

Un diagramme UML est une représentation graphique, qui s'intéresse à un aspect précis du modèle. C'est une perspective du modèle, pas "le modèle". Chaque type de diagramme UML possède une structure (les types des éléments de modélisation qui le composent sont prédéfinis).

Un type de diagramme UML véhicule une sémantique précise (un type de diagramme offre toujours la même vue d'un système). Combinés, les différents types de diagrammes UML offrent une vue complète des aspects statiques et dynamiques d'un système. Par extension et abus de langage, un diagramme UML est aussi un modèle (un diagramme modélise un aspect du modèle global). [32]

2.5. Vues UML

2.5.1. Les vues statiques :

Diagrammes de cas d'utilisation : décrit les interactions entre le futur système et ses utilisateurs.

Diagrammes de classes : est généralement considéré comme le plus important dans un développement orienté objet exprime la structure statique du système en termes de classes et de relations entre ces classes.

Diagrammes d'objets : représente des instances d'objets qui existent à un moment donné dans le système.

Diagramme de packages : Il montre l'organisation logique du modèle et les relations entre packages.

Diagramme de structure composite : Il montre l'organisation interne d'un élément statique complexe.

Diagrammes de composants : décrivent les composants et leurs dépendances dans l'environnement de réalisation.

Diagrammes de déploiement : montrent la disposition physique des différents matériels (les nœuds) sur lesquels le logiciel sera déployé. [33]

2.5.2. Les vues dynamiques :

Diagrammes de collaboration : permet de mettre en évidence les interactions entre les différents objets du système.

Diagramme de vue d'ensemble des interactions : Il fusionne les diagrammes d'activité et de séquence pour combiner des fragments d'interaction avec des décisions et des flots.
Diagramme de temps : Il fusionne les diagrammes d'états et de séquence pour montrer l'évolution de l'état d'un objet au cours du temps.

Diagrammes de séquence : montre un enchaînement d'opérations entre des objets du système

Diagrammes d'états-transitions : Ils ont pour rôle de représenter les traitements (opérations) qui vont gérer le domaine étudié. Ils définissent l'enchaînement des états de classe et font donc apparaître l'ordonnancement des travaux.

Diagrammes d'activités : il montre l'enchaînement des activités qui concourent au processus.

[33]

2.6. Pourquoi modéliser

Un modèle est une abstraction d'objets de la réalité. C'est donc une simplification du monde réel. La problématique consiste alors à trouver le bon niveau d'abstraction et à exprimer les concepts du monde réel dans un langage assez abstrait mais aussi précis qu'un langage de programmation pour que ce langage soit interprétable par un programme informatique. Le code source d'un système logiciel est un modèle du système. Ce code source ne fournit cependant qu'un seul niveau d'abstraction, celui de la mise en œuvre sur une infrastructure matérielle particulière, compréhensible par une partie seulement des intervenants du projet informatique, les développeurs. Le processus d'informatisation consiste à définir des étapes pour aller d'un cahier des charges rédigé en langage naturel à une mise en œuvre dans un code source particulier. Le modèle du système dans les premières phases de ce processus est nécessairement une simplification du système réel. Le processus de modélisation vise à mieux cerner les limites du système à réaliser. Ensuite, les modèles sont raffinés de plus en plus pour aboutir au code. [34]

2.7. Les avantages et les inconvénients d'UML

2.7.1. Les points forts d'UML

- UML est un langage formel et normalisé.
- Gain de précision.
- Gage de stabilité.
- Encourage l'utilisation d'outils.
- UML et un support de communication performant.
- Il cadre l'analyse.
- Il facilite la compréhension des représentations abstraites complexes.
- Son caractère polyvalent et sa souplesse en font un langage universel. [30]

2.7.2. Les points faibles d'UML

- La mise en pratique d'UML nécessite un apprentissage et passe par une période d'adaptation :
- Même si l'Espéranto est une utopie, la nécessité de s'accorder sur des modes d'expression communs est vitale en informatique. UML n'est pas à l'origine des concepts objets, mais il constitue une étape majeure, car il unifie les différentes approches et donne une définition plus formelle.
- Le processus (non couvert par UML) est une autre clé de la réussite d'un projet Or, l'intégration d'UML dans un processus n'est pas triviale et améliorer un processus est une tâche complexe et longue.
- Les auteurs d'UML sont tout à fait conscients de l'importance du processus, mais l'acceptabilité industrielle sont tout de la modélisation objet passe d'abord par la disponibilité d'un langage d'analyse objet performant et standard. [30]

3.La conception de l'application

La conception est l'étape qui suit l'analyse. Elle consiste à modéliser et à détailler tous les éléments de modélisations issus de la phase d'analyse. Au cours de cette étape, nous illustrons les diagrammes de séquences pour certains cas d'utilisations.

3.1. Diagramme de cas d'utilisation

Ce diagramme est constitué d'acteurs et de cas d'utilisations et il illustre les relations entre ces Éléments. Un acteur est une entité extérieure qui déclenche l'un des cas d'utilisation.

Le diagramme de cas d'utilisation englobe les fonctionnalités offertes par le système et décrit Son comportement, en identifiant les actions et les interactions entre les acteurs. Ce diagramme Est utilisé dans la phase d'analyse afin de définir les besoins des utilisateurs. [35]

✓ Éléments de base

Acteur : Les acteurs, qu'ils soient des personnes ou des systèmes, sont représentés par des bonshommes en fil de fer.

Système : Le système concerné par le cas d'utilisation est représenté par un rectangle.

Use Case : Le cas d'utilisation est représenté par une ellipse contenant généralement une courte phrase décrivant le processus.

4.3.1. Diagramme de cas d'utilisation d'étudiant

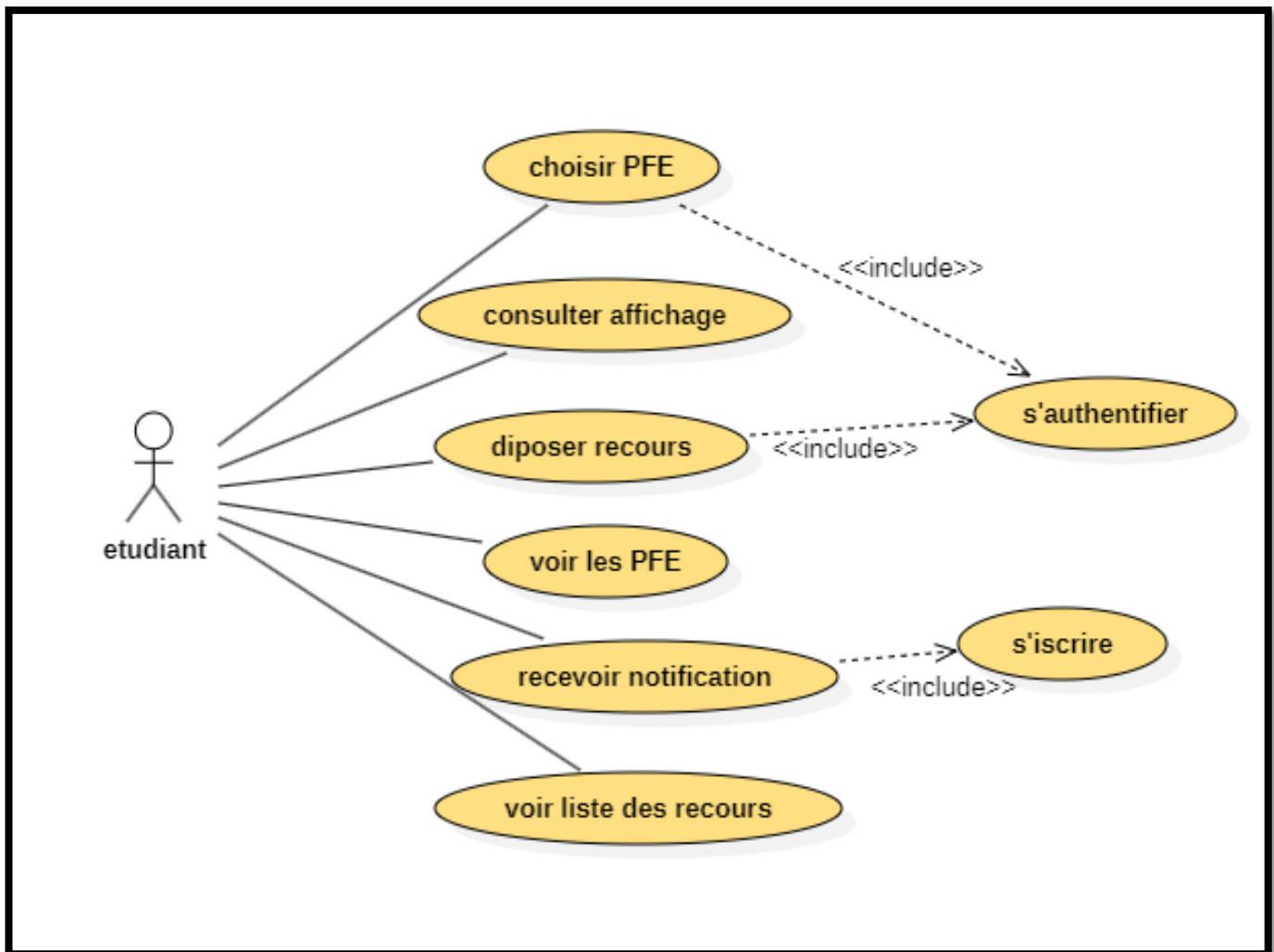


Figure 3 .1. Diagramme de cas d'utilisation du l'étudiant.

4.3.2. Diagramme de cas d'utilisation d'enseignant

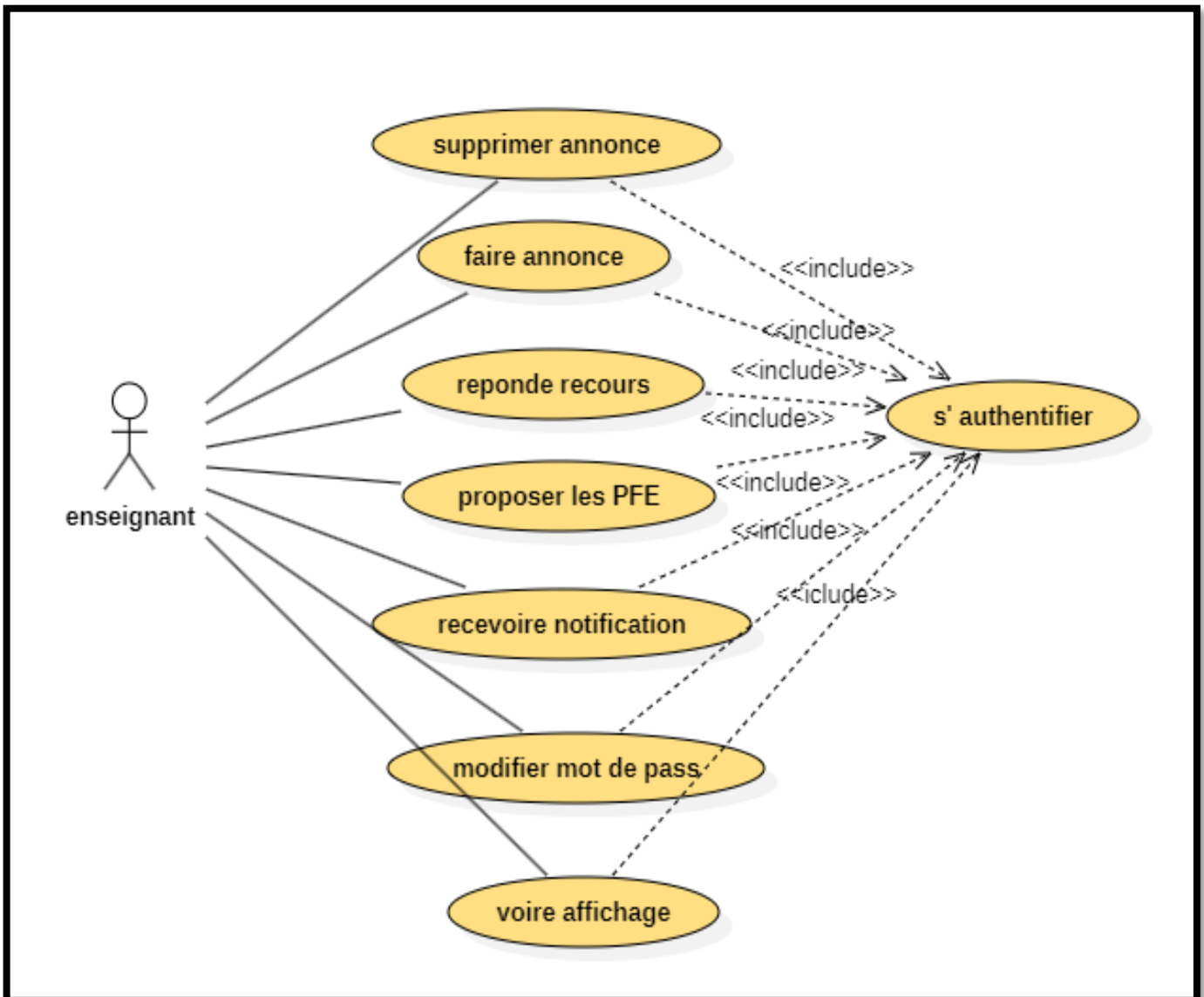


Figura :3 .2. Diagramme de cas d'utilisation de l'enseignant.

4.3.3. Diagramme de cas d'utilisation d'administration

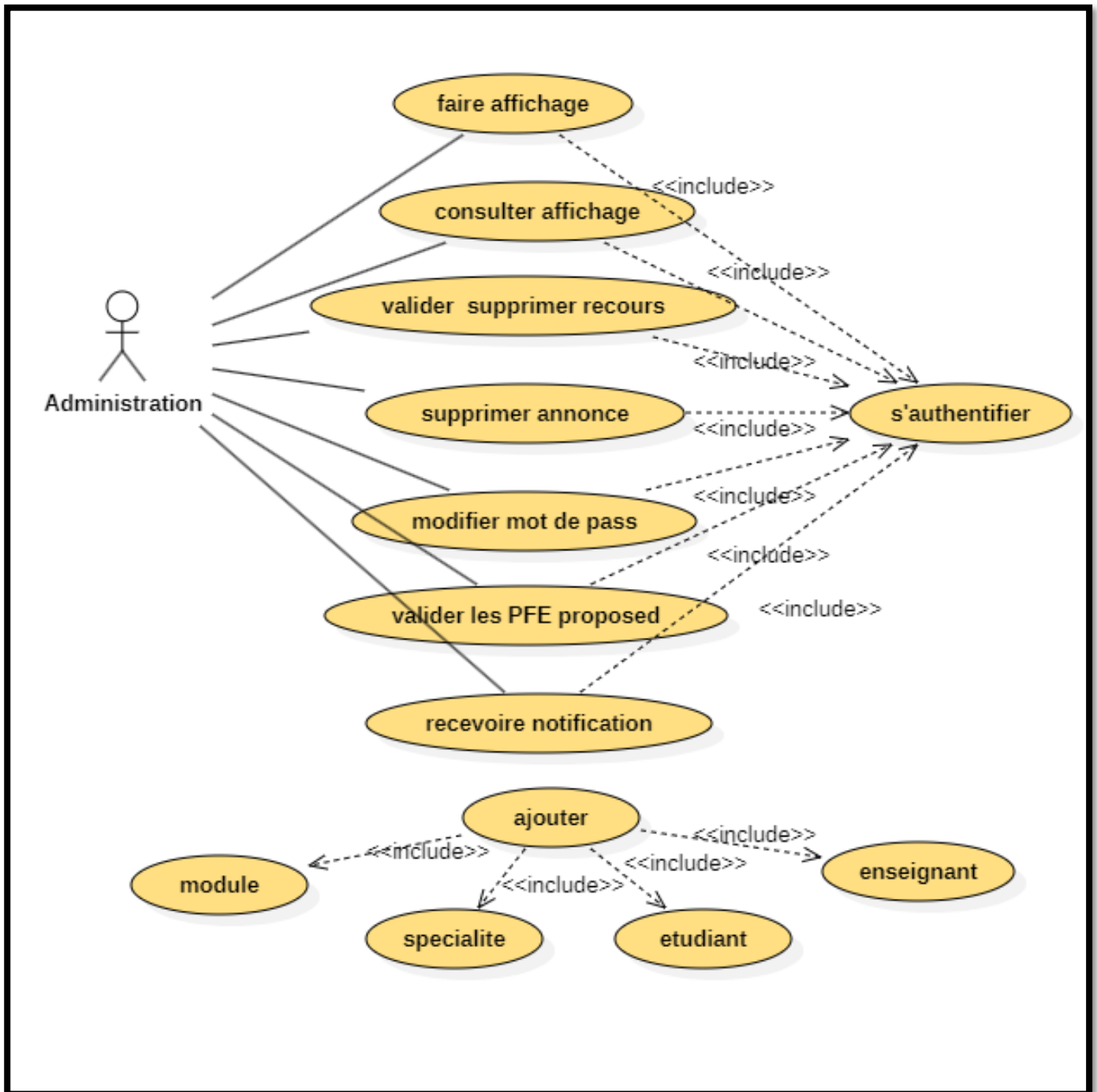


Figure 3.3. Diagramme de cas utilisation d'administration.

3.2. Diagramme de séquence

Il se compose des objets et des flèches. Ces éléments sont organisés selon deux axes

Perpendiculaires : le premier axe exprime le temps, et le deuxième axe représente les instances.

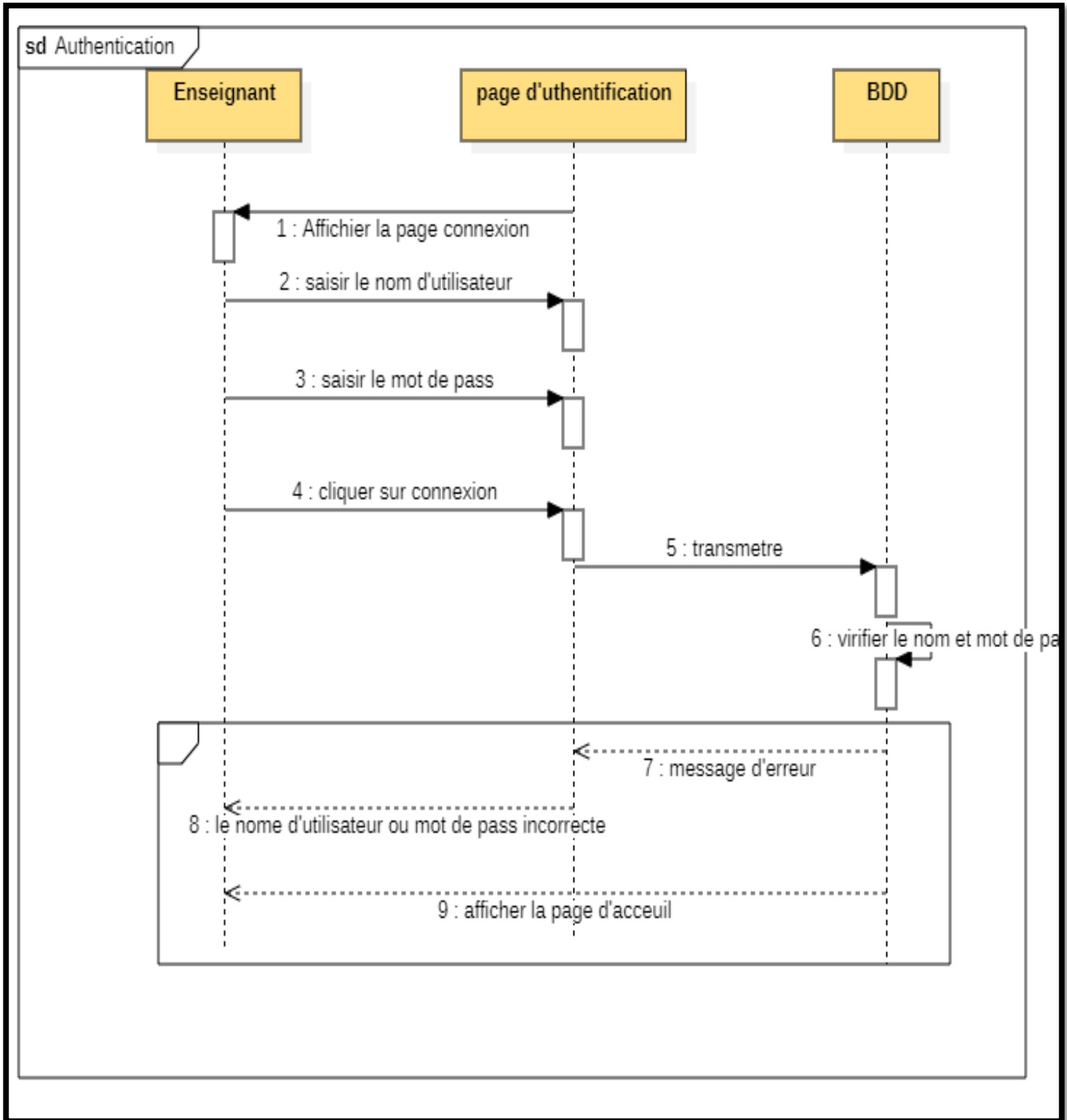
Le diagramme de séquence présente le déroulement d'un cas d'utilisation sous forme d'un

Scénario entre les acteurs et leurs interactions séquentielles par le changement des messages entre les objets, en respectant l'ordre chronologique (en fonction de temps). [35]

✓ Élément de base

- Le diagramme a deux dimensions :
 - Dimension verticale : temps
 - Dimension horizontale : objets
- L'ordre d'envoi d'un message est déterminé par sa position sur l'axe vertical du diagramme le temps s'écoule "de haut en bas" de cet axe.
- La disposition des objets sur l'axe horizontal n'a pas de conséquence pour la sémantique du diagramme. [5]

3.2.1. Diagramme de séquence authentification :



3.2.2. Diagramme de séquence faire recours

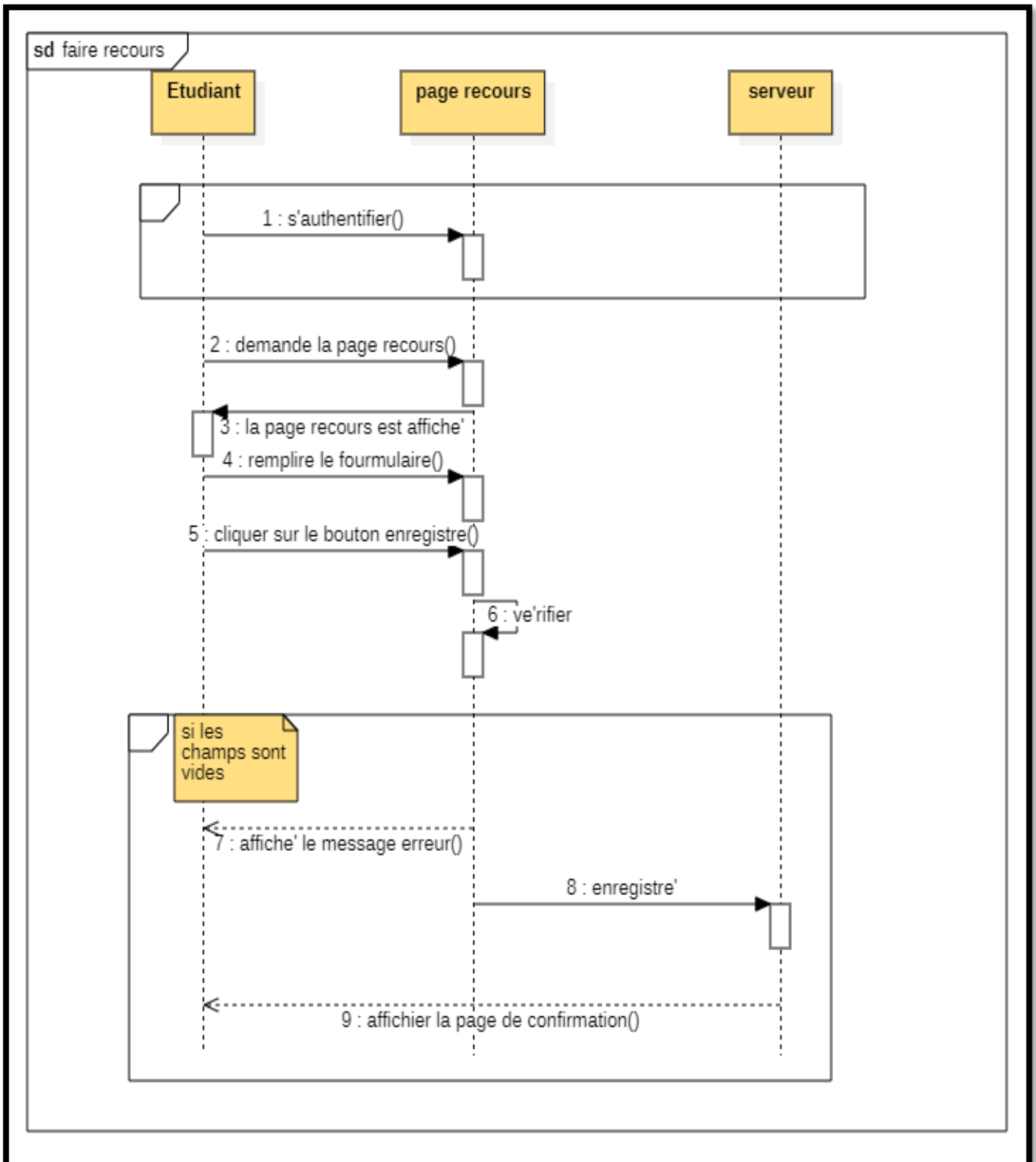


Figure 3 .5. Diagramme de Séquence de faire recours.

3.2.3. Diagramme de séquence faire annonce

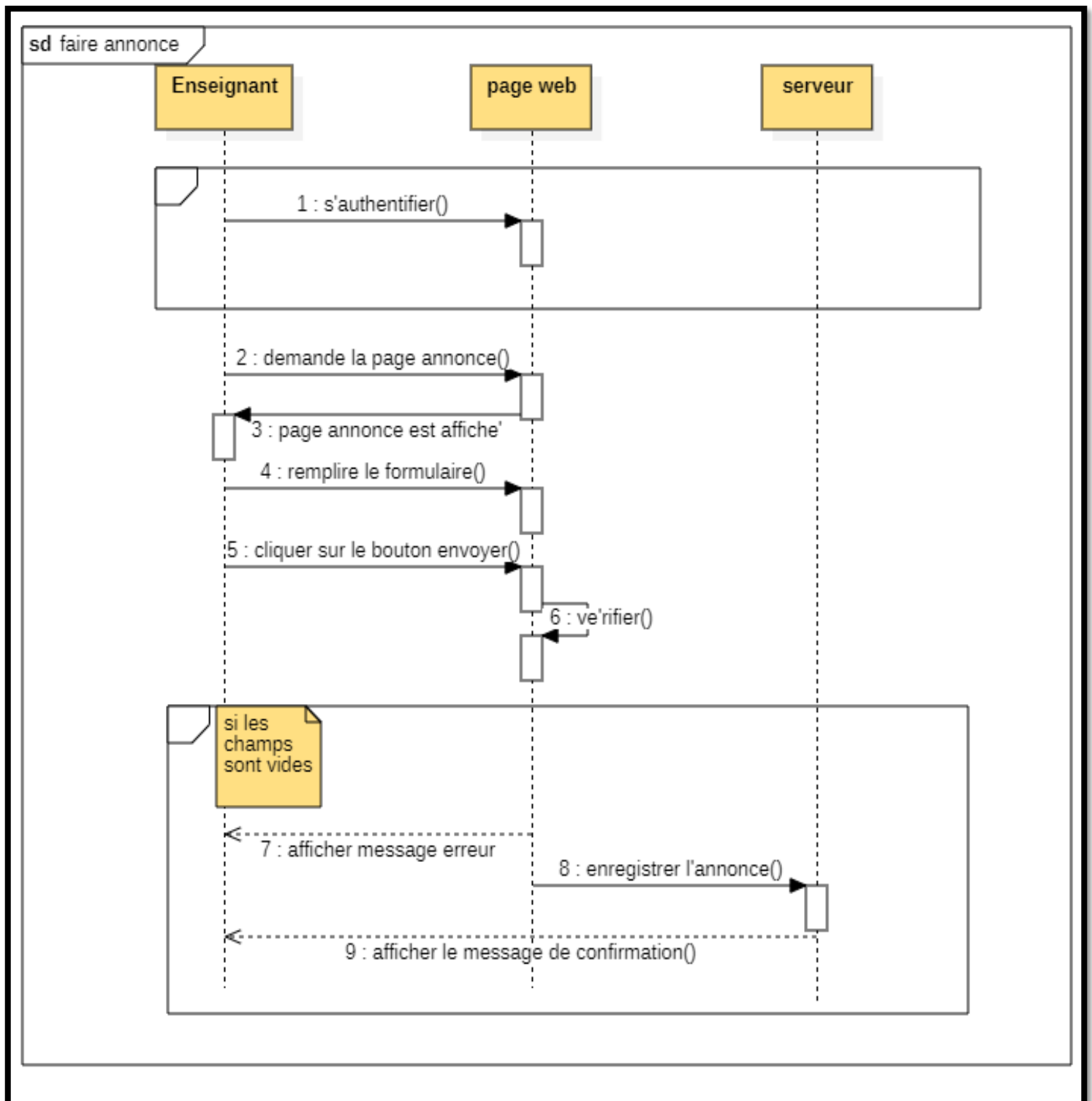


Figure 3 .6. Diagramme de Séquence de faire annonce.

3.2.4. Diagramme de séquence Ajouter PFE

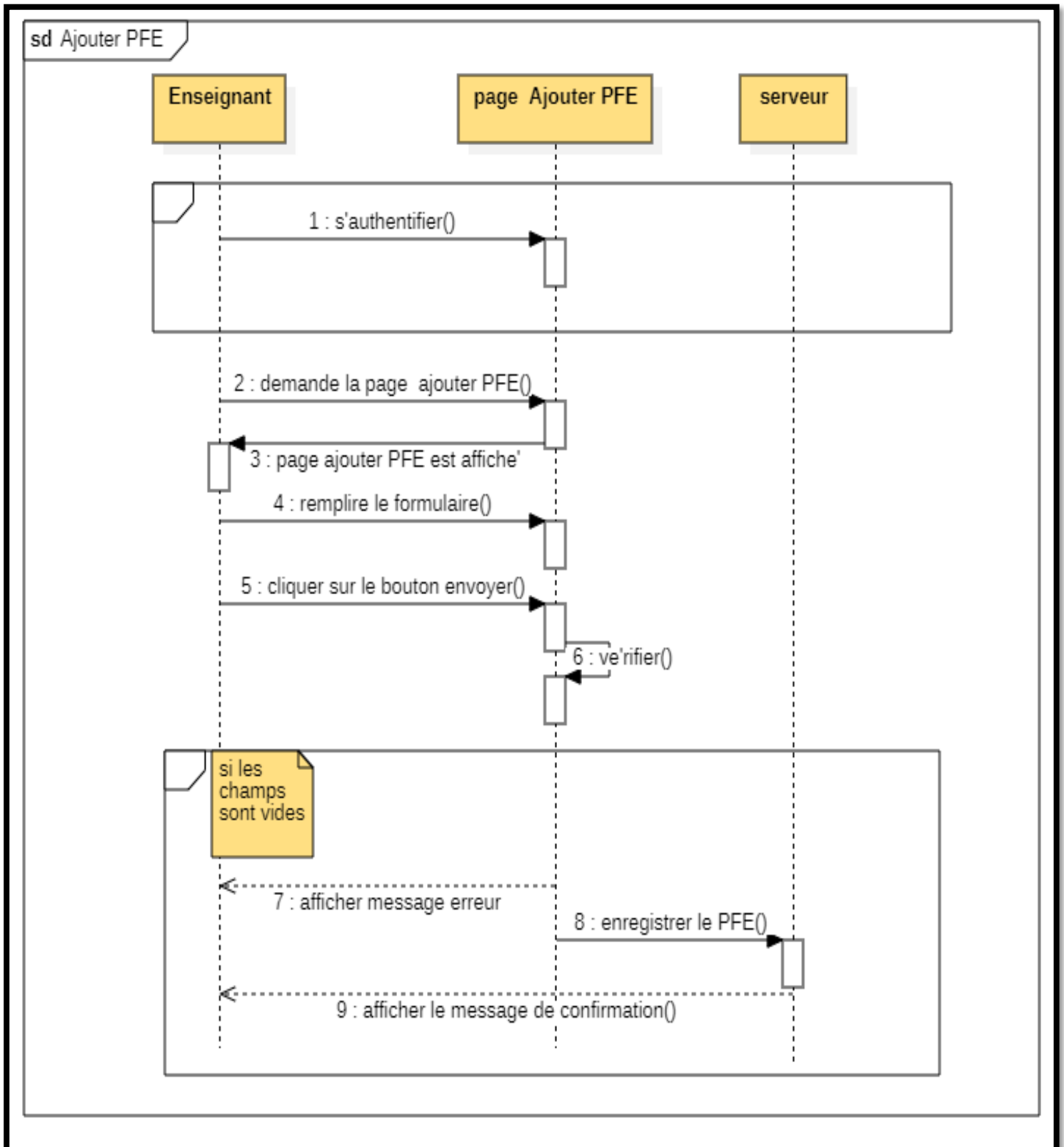


Figure 3.7. Diagramme de Séquence Ajouter PFE.

Conclusion

Dans ce chapitre, nous avons modélisé notre application avec le langage UML en sélectionnant les principaux diagrammes structurels et dynamiques, tels que le diagramme de classes, le diagramme de cas d'utilisation et le diagramme de séquence.

Chapitre 04 : Implémentation

1.Introduction

Après avoir analysé les besoins et défini la méthodologie de conception, nous allons programmer l'essentiel de notre projet dans un environnement approprié. Dans ce chapitre, nous présenterons l'environnement de développement choisi ainsi que les outils nécessaires à la réalisation de l'application. Nous montrerons également quelques interfaces utilisateur correspondant aux pages Web développées.

2. Outils de développements

2.1. La machine de développement :

Le logiciel a été développé sur un micro-ordinateur disposant des caractéristiques suivantes :

Système d'exploitation : Windows 10.

Processeur : Intel(R) Core (TM) i5 -7400U CPU @ 3.00GHz 2.40 GHz

Mémoire (RAM) :12288MB.

2.2. Langages de programmation



✓ **Node.js.**

Node JS est une plateforme logicielle libre en JavaScript. Il fonctionne comme un environnement d'exécution JavaScripts. Pourtant, Node Js n'est ni un Framework, ce n'est pas non un langage de programmation, et il n'a pas non plus de relation avec .NET, PHP, ou JAVA. En réalité Node JS est un outil pour exécuter JavaScript en dehors du navigateur.

Il est très apprécié des développeurs, car c'est une technologie flexible et non bloquante, c'est-à-dire qu'il permet de concevoir des applications en réseaux performantes (telles qu'un

serveur web, des messageries instantanées, une API ou un job CRON), et cela en étant non-bloquant : les tâches sont donc réalisées à la suite sans attendre les résultats des tâches précédentes. [36]



✓ NPM.

npm est le gestionnaire de paquets par défaut pour l'environnement d'exécution JavaScript Node.js.

npm se compose d'un client en ligne de commande, également appelé npm, et d'une base de données en ligne de paquets publics et privés payants, appelée le registre npm. Le registre est accessible via le client, et les paquets disponibles peuvent être parcourus et recherchés via le site Web de npm. Le gestionnaire de paquets et le registre sont gérés par npm, Inc.

Depuis la version 0.6.3 de Node.js, npm fait partie de l'environnement et est donc automatiquement installé par défaut. Npm fonctionne avec un terminal et gère les dépendances pour une application. Il permet également d'installer des applications Node.js disponibles sur le dépôt npm. [37]



✓ MYSQL

MySQL est un système de gestion de base de données relationnel, un langage de requêtes vers les bases de données exploitant le modèle relationnel et utilise le langage SQL comme langage de requête. [5]

2.3. Les outils



✓ Visual studio code

Est un éditeur de code développé par Microsoft. Il est l'un de ces premiers produits open source et gratuit, et surtout disponible sur les systèmes d'exploitation Windows, Linux et Mac. Vs code est développé avec le Framework Electron et conçu principalement pour développer des projets avec JavaScript, Node.js ou encore Type Script. [32]

✓ Word 2016

Word est le logiciel phare de la suite Bureautique Microsoft Office. C'est l'un des logiciels les plus utilisés dans le monde et permet de rédiger des lettres, CV, rapports et tous types de documents texte. Word permet de mettre en page vos documents, de générer des sommaires, de numéroter automatiquement des pages, corriger la grammaire et l'orthographe, de créer des schémas, de faire du publipostage...

Word est plutôt bien conçu et facile à prendre en main, il vous permettra de rédiger lettres et courriers, ou encore de mettre en forme votre CV. Vous pourrez ensuite choisir un jeu de typographies et couleurs pour personnaliser votre document. Il est possible de générer un sommaire automatiquement pour les gros documents (pratique si vous écrivez un mémoire). Depuis la version 2007, il est possible d'intégrer des petits schémas, graphiques, que l'on appelle Smart Art. Vous pouvez également insérer des tableaux ou graphiques réalisés avec Excel. [38]



✓ STAR UML

StarUML est un outil spécialisé dans la modélisation UML dans le domaine du développement d'applications. Riche en fonctionnalités, il a une ergonomie forte permettant de créer des diagrammes de haute qualité et de les exporter en photos, mais aussi de générer du code à partir d'un diagramme de classe. [12]

2.4. Les Framework



BOOTSTRAP

Bootstrap est un Framework développé par l'équipe de Twitter. Proposé en open source sous licence MIT, ce Framework utilise les langages HTML, CSS et JavaScript pour fournir aux développeurs des outils facilitant la création de sites web. Conçu pour développer des sites avec un design responsif qui s'adapte à tous types d'écrans, notamment les smartphones, Bootstrap offre des styles prédéfinis pour les typographies, les boutons, les interfaces de navigation, et bien plus encore. Ce type de Framework est appelé "Framework front-end."

3. Quelques exemples d'interface

3.1. Page d'accueil du site des enseignant



Figure 4.1 : Page d'accueil enseignant.

3.2. Page d'accueil du site étudiants

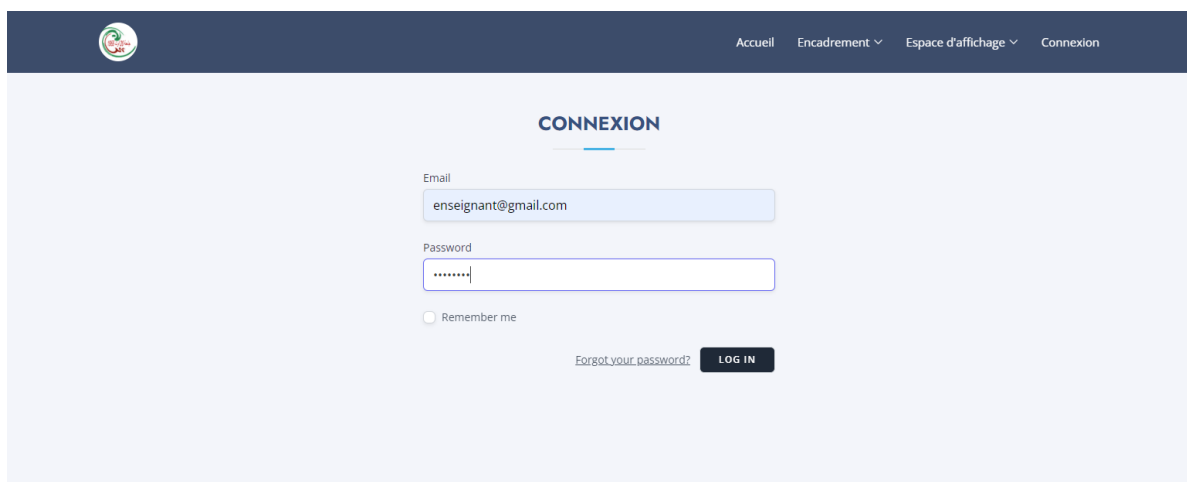


Figure 4.2 : page d'accueil étudiants.

3.3. Page de connexion

L'authentification des étudiants et des enseignants pour accéder au site du département avec un email personnel et le mot de passe.

3.3.1. Page de connexion enseignants



CONNEXION

Email
enseignant@gmail.com

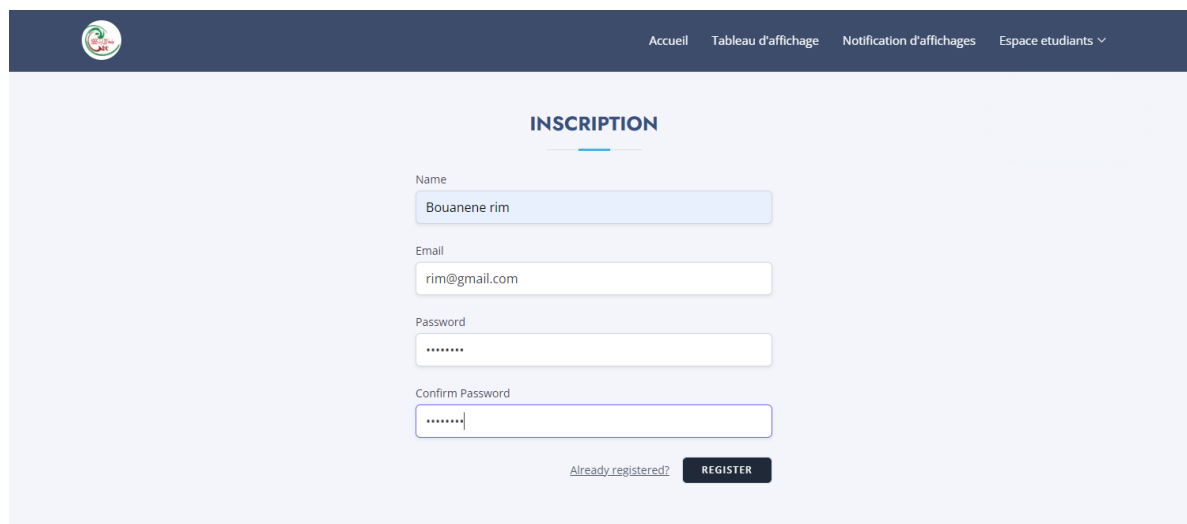
Password
.....

Remember me

[Forgot your password?](#) **LOG IN**

Figure :4.3. Page de connexion enseignant.

3.3.2. Page de Inscription étudiants



INSCRIPTION

Name
Bouanene rim

Email
rim@gmail.com

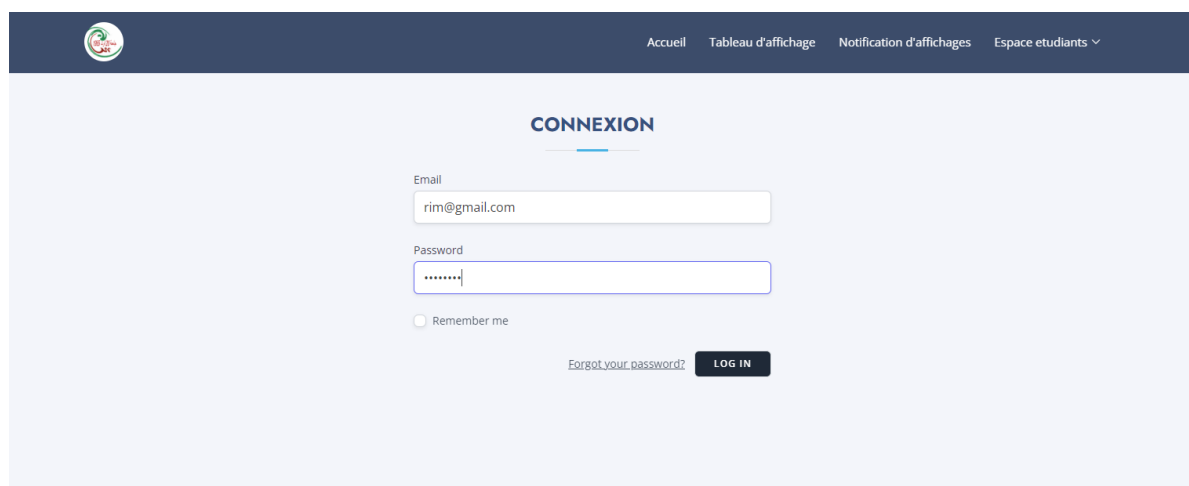
Password
.....

Confirm Password
.....

[Already registered?](#) **REGISTER**

Figure :4.4. Page inscription étudiants.

3.3.3. Page connexion étudiants

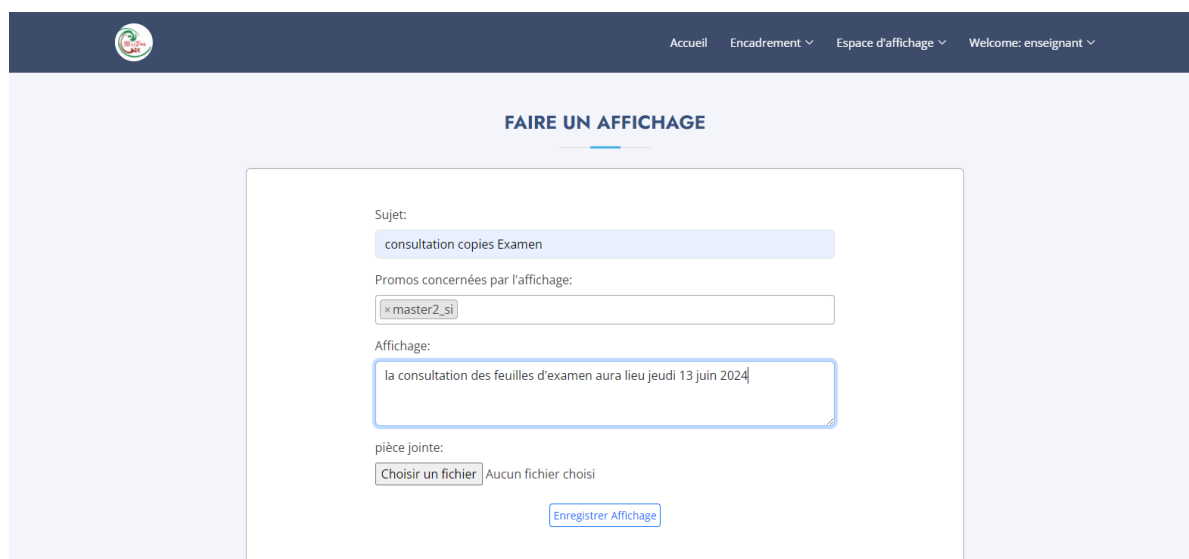


The screenshot shows a web interface for student login. At the top, there is a dark blue navigation bar with a logo on the left and menu items: 'Accueil', 'Tableau d'affichage', 'Notification d'affichages', and 'Espace etudiants' with a dropdown arrow. The main content area is light blue and features the heading 'CONNEXION' in bold. Below the heading, there are two input fields: 'Email' containing 'rim@gmail.com' and 'Password' with masked characters. A 'Remember me' checkbox is present below the password field. At the bottom of the form, there is a link 'Forgot your password?' and a dark blue 'LOG IN' button.

Figure :4.5. Page de connexion étudiants.

3.4. Page affichage enseignant

Créer un affichage



The screenshot displays the 'FAIRE UN AFFICHAGE' (Create Announcement) page for a teacher. The top navigation bar is dark blue with a logo and menu items: 'Accueil', 'Encadrement' with a dropdown arrow, 'Espace d'affichage' with a dropdown arrow, and 'Welcome: enseignant' with a dropdown arrow. The main content area is light blue and contains the heading 'FAIRE UN AFFICHAGE'. The form includes several fields: 'Sujet:' with the value 'consultation copies Examen'; 'Promos concernées par l'affichage:' with a dropdown menu showing 'master2_si'; 'Affichage:' with the text 'la consultation des feuilles d'examen aura lieu jeudi 13 juin 2024'; and 'pièce jointe:' with a file selection button 'Choisir un fichier' and the text 'Aucun fichier choisi'. At the bottom of the form, there is a blue 'Enregistrer Affichage' button.

Figure :4.6. Page d’affichage enseignant.

3.5. Page mes daffichage (enseignante)

TABLEAU D’AFFICHAGES (3 / TOTAL DE 3)

toutes les promo

enseignant
le : 14/06/2024
Affichage pour: [licenc2_sci](#)

Examen de rattrapage

Les étudiants concernés par l’examen de rattrapage doivent se présenter demain à 10.30 salle 22

enseignant
le : 10/06/2024
Affichage pour: [master2_sci](#)

consultation copies Examen

la consultation des feuilles d’examen aura lieu jeudi 13 juin 2024

Figure :4.7. Page mes d’affichage.

3.6. page Fiche d’encadrement

AJOUTER FICHE D’ENCADREMENT

Titre du projet:

Etudiants:

Promo du projet

Observation:

Figure :4.8. Page fiche d’encadrement.

3.7. page mes projets (enseignant)



Figure :4.9 : Page mes projets.

3.8. Tableau d'affichage étudiante

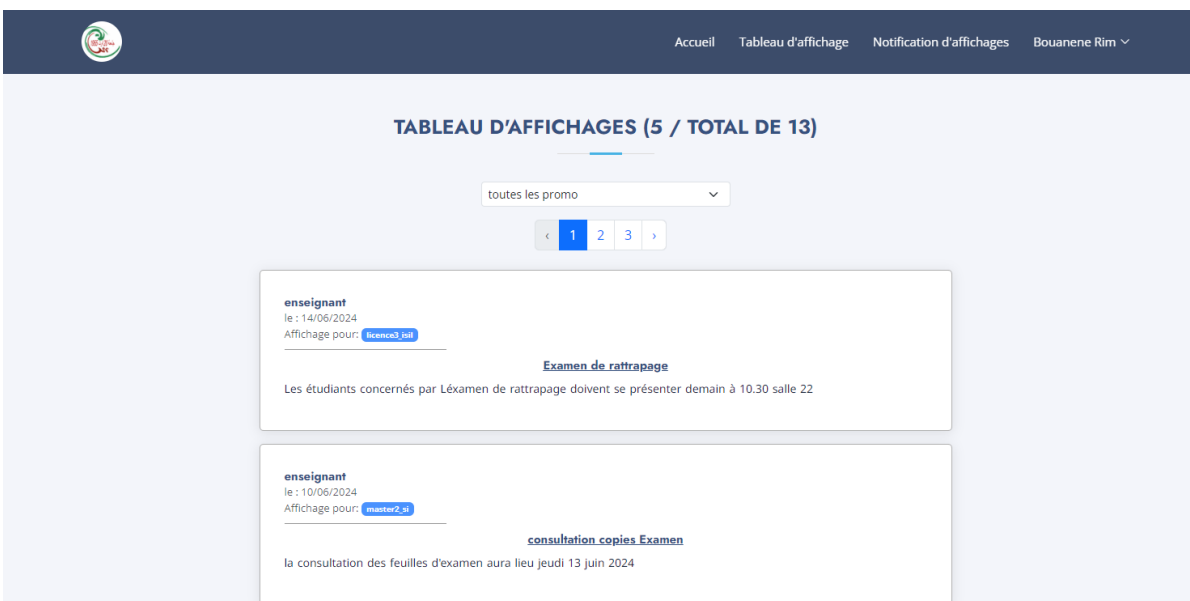


Figure :4.10. Page d'affichage étudiant.

3.9. page parametres de notification d'affichage

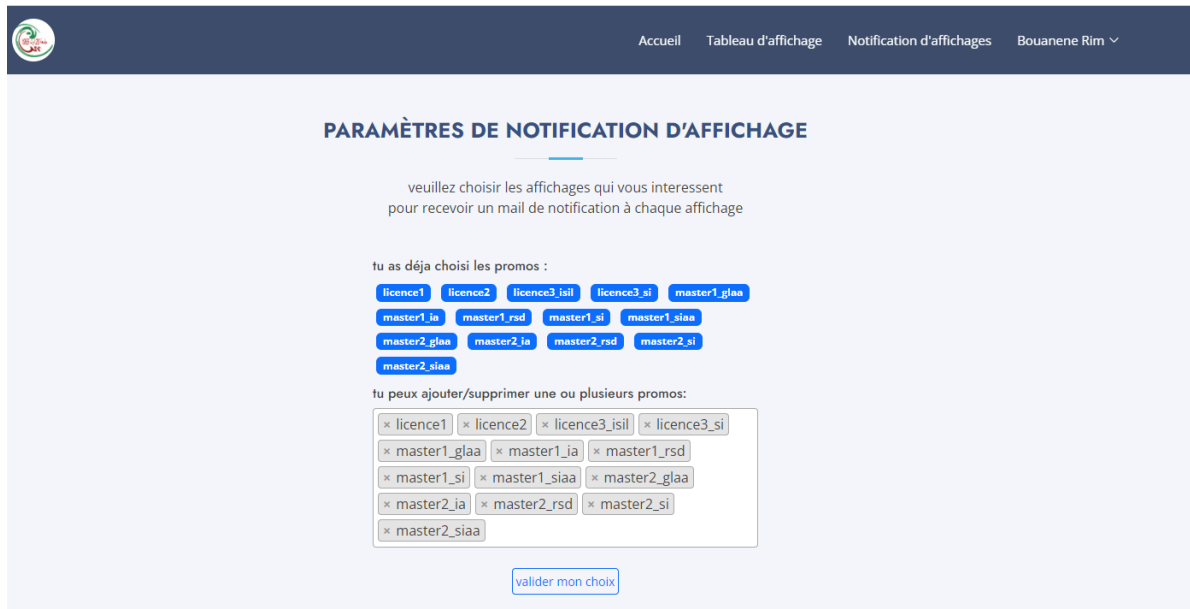


Figure :4.11. Page choisir les affichages.

3.10. Page d'affichage sur email

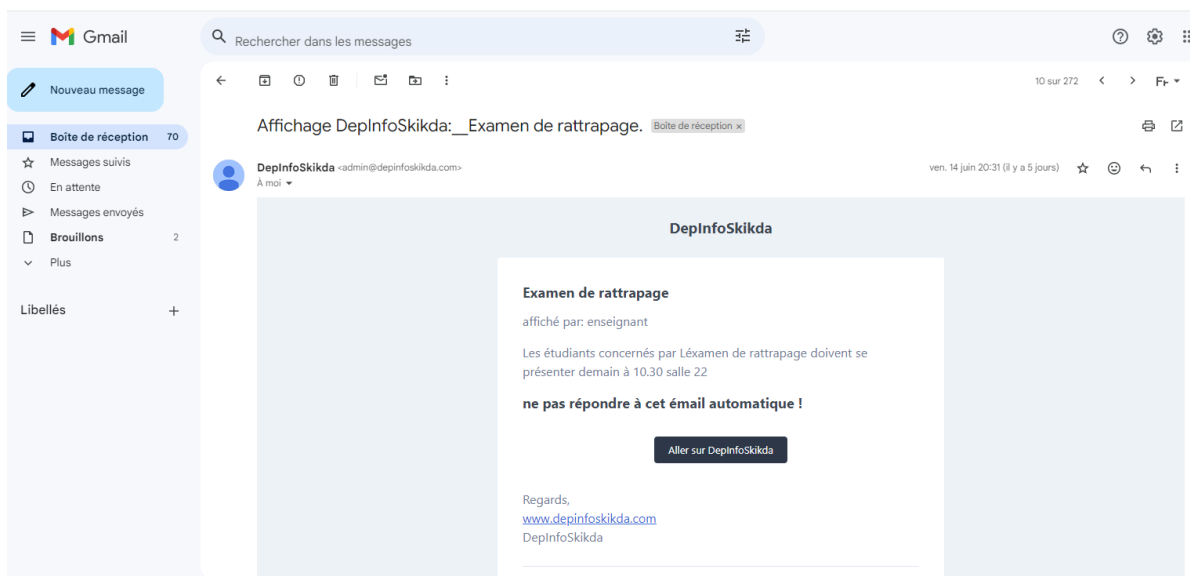


Figure : 4.12: Page de notification par email.

4. Conclusion

Ce chapitre récapitule l'ensemble des travaux accomplis dans le cadre de ce projet et présente les résultats des tests réalisés. Nous avons détaillé les techniques et les logiciels utilisés. Pour développer notre application et conclu en présentant quelques interfaces de notre application, mettant en lumière les principales fonctionnalités de notre système.

Conclusion Générale

Conclusion générale

Notre objectif était de concevoir et développer une application web MVC pour le département informatique de l'université de Skikda. Cette nouvelle version a pour but de corriger certains problèmes techniques et limites dans la version actuelle.

Cette application vise à simplifier les processus administratifs au sein du département, à réduire les déplacements physiques et à améliorer la communication entre enseignants et étudiants.

Pour assurer une analyse et une conception claires de notre application, nous avons utilisé le langage de modélisation UML.

L'application web a été développée avec le Framework PHP Laravel, ainsi qu'avec plusieurs autres outils comme Node.js, Bootstrap, XAMPP, Composer, et le gestionnaire de bases de données MySQL. Nous avons également utilisé des requêtes **Eloquent** de laravel pour manipuler les données et avons développé l'application dans l'environnement de programmation Visual Studio.

Ce projet a été une expérience très enrichissante, nous apportant de nombreux bénéfices. Il nous a permis de renforcer nos connaissances et compétences, notamment en programmation.

Références Bibliographiques

- [1] : KHAREZE Naima, Mémoire : Conception et Réalisation d'une Application web pour la Gestion des Rendez-vous Médicaux, UNIVERSITE ABDELHAMID IBN BADIS – MOSTAGANEM.
- [2] : OUCHAREF Sofiane, Mémoire : Conception et réalisation d'une application Web pour La gestion des archives (plans)Cas : Entreprise d'ELECTRO-INDUSTRIE Azazga, UNIVERSITE MOULOUD MAMMERI DE TIZI-OUZOU.
- [3] : Bahloul Imed Eddine, Khaled Faris, Mémoire : Conception d'un site Web Dynamique pour la gestion pédagogiques, Université de Guelma.
- [4] : <http://codegenome.com/blog/posts/la-difference-entre-un-site-web-et-une-application-Web>.
- [5] : Kerbouche Rayene, Marsa Louiza, Mémoire : Conception et réalisation d'une application web et mobile pour Ecole Normale de l'Enseignement Technologique Skikda-ENSET, Université 20 Août 1955 – Skikda.
- [6] : <https://facsc.univ-annaba.dz/wp-content/uploads/2018/04/Chapitre-3-Architecture-Client- Serveur.pdf>.
- [7] : Mr. IBAZIZENE Aziz, Mémoire : Conception et réalisation d'un site web pour le laboratoire LAMPA, UNIVERSITE MOULOUD MAMMERI, TIZI-OUZOU.
- [8] : Moukli Aghiles, Mansouri Sofiane, Mémoire : Conception et réalisation d'une application Client Serveur pour le service personnel Cas : NAFTAL Oued-Aissi Tizi-Ouzou, Université Mouloud Mammeri de Tizi Ouzou.
- [9] : BENREJDAL Souhila, Mémoire : Conception et réalisation d'une application Mobile CRM, UNIVERSITE MOULOUD MAMMERI TIZI-OUZOU (UMMTO).
- [10] : Mlle BENYOUCEF Lynda. Mlle BOUAMARA Nedjma, mémoire : Conception et Réalisation d'une application mobile multiplateforme pour la gestion des ventes, Université Mouloud Mammeri de Tizi-Ouzou.
- [11] : <https://www.taktilcommunication.com/blog/applications-mobile/site-mobile-vs-application-mobile-avantages-et-inconvenients.html>.

- [12] : DAOUI Rabéa, mémoire : Conception et Réalisation d'une Application Web pour la gestion du Laboratoire d'Informatique Médicale (LIMED) de l'Université de Bejaia, Université A/Mira de Bejaia.
- [13] : <https://kinsta.com/fr/base-de-connaissances/qu-est-ce-que-laravel/>
- [14] : Ait Ziane Meriem, mémoire : Le contrôle d'accès pour la protection des données personnelles médicales informatisées, Université SAAD DAHLEB BLIDA.
- [15] : KHALOUF OURDIA, STOUTAH DYHIA : Conception et réalisation d'une application web jee pour la gestion de centre médecine de travail. CAS : ENIEM, UNIVERSITE MOULOUD MAMMERI DE TIZI OUZOU.
- [16] : <https://kinsta.com/fr/blog/laravel-blade/>
- [17] : <https://fr.wikipedia.org/wiki/XAMPP>
- [18] : [https://fr.wikipedia.org/wiki/Composer_\(logiciel\)](https://fr.wikipedia.org/wiki/Composer_(logiciel))
- [19] : <https://grafikart.fr/tutoriels/structure-laravel-2113>
- [20] : <https://stackify.com/laravel-eloquent-tutorial/>
- [21] : <https://manoj-shu100.medium.com/eloquent-orm-in-laravel-simplifying-database-interactions-b0269942f190>
- [22] : <https://laravel.sillo.org/cours-laravel-10-les-donnees-migrations-et-modeles/#:~:text=et%20les%20mod%C3%A8les,-,Les%20migrations,en%20charge%20par%20cet%20outil.>
- [23] : <https://kinsta.com/fr/blog/crud-laravel/>
- [24] : <https://dev.to/mikevarenek/laravel-middlewares-3nlo>
- [25] : <https://www.honeybadger.io/blog/php-laravel-notifications/>
- [26] : <https://wpwebinfotech.com/blog/laravel-breeze/>
- [27] : <https://medium.com/@thoyibh07/laravel-breeze-the-ultimate-guide-to-authentication-cb7e8a7e3277>
- [28] : <https://www.akilischool.com/cours/laravel-lauthentification-avec-breeze>

[29] Mr. BELLAKHDAR Abdelouahab Mr. ZAIDI Nabil, mémoire : Gestion De Patient De L'hôpital Ophtalmologie de l'amitié Algero-Cubain, Université Kasdi Merbah Ouargla.

[30] : Lezghed Amir, mémoire : La Génération Automatique des Ontologies à partir des Diagrammes de classes UML, Université de 8 Mai 1945 – Guelma.

[31] : <https://www.piloter.org/projet/methode/uml.htm>

[32] : Mr. BOUKHEMIS Salih, Mémoire : Amélioration d'une application Web-Mobile pour la gestion des mémoires de fin d'études et l'affichage dans le département d'informatique, Université 20 Août -1955-Skikda.

[33] : Bentaleb Wahid Ferhat Hmida Salim, Mémoire : Système Distribué Pour La Gestion De Production, UNIVERSITE KASDI MERBAH OUARGLA.

[34] : <https://www-inf.telecom-sudparis.eu/cours/CSC4002/EnLigne/Cours/CoursUML/3.2.html>

[35] : ZAHOUI Anissa Amel, Mémoire : DEVELOPPEMENT D'UNE CHAINED'OUTILS EN FONCTION DU NOUVEAU STANDARD FONDATIONNEL UML (FUML), UNIVERSITE BADJI MOKHTAR.

[36] <https://www.sooyoos.com/publication/node-js/>

[37] : <https://fr.wikipedia.org/wiki/Npm/>

[38] : <https://cours-informatique-gratuit.fr/dictionnaire/microsoft-office-word/>