

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA
RECHERCHE SCIENTIFIQUE
UNIVERSITE MOHAMED BOUDIAF - M'SILA

**FACULTE DEMATHEMATIQUES
ET D'INFORMATIQUE**

DEPARTEMENT D'INFORMATIQUE
N° :.....



DOMAINE : INFORMATIQUE
FILIERE : SYSTEMES INFORMATION ET
GENIE LOGICIEL(SIGL)

Mémoire présenté pour l'obtention
Du diplôme de Master Académique

Par : DILMI Sarra

DELLOUM Zohra

Intitulé

DEVELOPPEMENT D'UN CARTABLE NUMERIQUE
POUR LES ETUDIANTS EN UTILISANT PYTHON ET
DJANGO

Soutenu devant le jury composé de messieurs :

CHIKOUCHE Nouredine	Université de M'sila	Président
MEZRAG Fares	Université de M'sila	Rapporteur
LAKHAL Meftah	Université de M'sila	Examineur

Année universitaire : 2022/2023

Dédicace

Je dédie ce modeste travail :

A mes chers parents :ma mère et mon père pour la patience ,leur amour leur soutien et leur encouragement.

A mes très chers frères : *Khaled ,Zakaria et Billel* et Ma belles sœurs :*Sara* et à tous leurs famille.

A mes grands parents .

A tous mes amies.

DELLOUM Zohra

Dédicace

Toutes les lettres ne sauraient trouver les mots qu'il faut. . .

Tous les mots ne sauraient exprimer la gratitude,

L'amour, le respect, la reconnaissance. . .

Aussi, c'est tout simplement que je dédie ce mémoire à :

Mes chers parents *Ammar et Samira* à qui je dois ce que je suis et ce que je serai,

Qui ont tout donné et tout sacrifié pour mon bien et ma réussite.

A mes chers frères *Mohammed, Yacine et Zino*, ma chère sœur *Imane* et sa famille, et
toute la famille *Dilmi*.

A moi-même

A toutes les personnes que j'aime en Allah.

DILMI Sarra

ملخص

في الوقت الحاضر، يمكن للطلاب الاعتماد على العديد من الموارد الرقمية والمنصات الإلكترونية وهذا لتسهيل عملية التعلم، ولكن في نفس الوقت يمكن أن يؤدي ذلك إلى القلق و الارتباك. لمعالجة هذه المسألة، فإن الفكرة هي تطوير حقيبة رقمية (تطبيق ويب) تجمع كل هذه الميزات في مكان واحد. ستتيح للطلاب هذا التطبيق إمكانية اتخاذ الملاحظات وإدارة المهام وإجراء البحوث والعثور على معلومات ذات صلة. هدف هذا المشروع هو توفير أداة عملية وسهلة الاستخدام للطلاب لتنظيم أنفسهم وتعزيز إنتاجيتهم الأكاديمية.

الكلمات المفتاحية: تطبيق ويب، محفظة رقمية، دجانغو، بايثون.

Abstract

Nowadays, students have access to numerous online resources and platforms to facilitate their learning, but this can also lead to confusion and stress. To address this issue, the idea is to develop a digital bag (a web application) that combines all these features in one place. This application will allow students to take notes, manage their assignments, conduct research, and find relevant information. The objective of this Master thesis is to provide them with a practical and user-friendly tool to organize themselves and enhance their academic productivity.

Keywords: Web application, Digital bag, Django, Python.

Résumé

De nos jours, les étudiants ont accès à de nombreuses ressources et plate-formes en ligne pour faciliter leur apprentissage, mais cela peut aussi causer de la confusion et du stress. Pour remédier à cela, l'idée est de développer un cartable numérique (une application web) regroupant toutes ces fonctionnalités en un seul endroit. Cette application permettra aux étudiants de prendre des notes, gérer leurs devoirs, effectuer des recherches et trouver des informations pertinentes. L'objectif de ce mémoire de Master est de leur fournir un outil pratique et convivial pour s'organiser et améliorer leur productivité académique.

Mots Clés Application web, Cartable numérique, Django, Python.

Table des matières

Table des figures	i
Liste des tableaux	iii
1 Notions de base sur le web	3
1.1 Introduction	3
1.2 Vue d'ensemble	3
1.3 Technologies Web	3
1.3.1 Site web	4
1.3.2 Serveur web	4
1.4 Programmation web	5
1.4.1 Programmation client-side	5
1.4.2 Programmation server-side	5
1.4.3 Application web	6
1.4.4 Fonctionnement d'une application web	6
1.5 Environnement de développement d'applications	7
1.6 Système de Gestion de Base de Données (SGBD)	7
1.7 Cartable numérique	8
1.8 Conclusion	8
2 Modélisation Conceptuelle	9
2.1 Introduction	9
2.2 Spécification des besoins	9
2.2.1 Besoins Fonctionnels	9
2.2.2 Besoins non fonctionnels	10
2.3 Le choix de méthode	10
2.3.1 Définition du processus unifié	11
2.3.2 Les caractéristiques de processus	11
2.3.3 Cycle de vie d'un processus	11

2.3.4	Les activités	12
2.4	Conclusion	12
3	Conception Détaillée	13
3.1	Introduction	13
3.2	Analyse conceptuelle	13
3.2.1	Le langage UML	13
3.2.2	Développement du langage de modélisation unifié (UML)	14
3.2.3	Les diagrammes	15
3.2.4	Diagrammes de cas d'utilisation	15
3.2.5	Diagrammes des séquences	17
3.2.6	Diagramme de classe	23
3.2.7	Le schéma relationnel	24
3.3	Conclusion	25
4	implémentation	26
4.1	Introduction	26
4.2	Outils utilisé dans le server-side	26
4.2.1	Django Framework	26
4.2.2	Python	29
4.2.3	Sqlite3	30
4.2.4	Web scraping	30
4.2.5	Bibliothèque youtube-search-python	31
4.2.6	Wikipedia API	31
4.2.7	API de dictionnaire avec l'API Google	32
4.2.8	Bibliothèque Crispy-form	32
4.3	Outils utilisé dans le client-side	33
4.3.1	HTML5	33
4.3.2	CSS3	33
4.3.3	JAVA SCRIPT	33
4.4	Visuel studio code pour le web	33
4.5	Mettre en place un environnement de développement Django	34
4.5.1	Configuration de l'environnement virtuel (virtualenv)	34
4.5.2	Démarrage du Django projet	35
4.6	Fonctionnalités du cartable numérique	38
4.6.1	Page de Login / Register.	38
4.6.2	Créer des notes	40

4.6.3	Faire des Homeworks.	41
4.6.4	Page de profil qui affiche des TO-DOs et des homeworks incomplets	43
4.6.5	Utiliser l'API dictionnaire	43
4.6.6	Rendre les résultats de recherche Youtube en utilisant l'API	44
4.6.7	Rechercher des livres via Google API	45
4.6.8	Rechercher sur Wikipedia via API	46
4.6.9	Opération commune de conversion des mesures	47
4.7	Conclusion	47
	Bibliographie	50

Table des figures

1.1	Aperçu de la technologie Web de base [1]	4
1.2	L'architecteur de serveur web [2]	5
1.3	L'architecture des applications Web [3]	6
1.4	Comment fonctionne une application web [4]	7
3.1	Diagramme de cas d'utilisation	16
3.2	Diagramme de séquence du cas d'utilisation "Register"	18
3.3	Diagramme de séquence du cas d'utilisation "Login"	19
3.4	Diagramme de séquence du cas d'utilisation "D'ajouter"	20
3.5	Diagramme de séquence du cas d'utilisation "Recherche"	21
3.6	Diagramme de séquence du cas d'utilisation "Supprission"	22
3.7	Diagramme de séquence du cas d'utilisation "Logout"	23
3.8	Diagramme de Classe	24
4.1	Model View Controller (MVC) [5]	27
4.2	Object-Relational Mapper(ORM) [6]	28
4.3	Django : Request/Response Cycle [3]	29
4.4	Langage "Python" [7]	30
4.5	API de dictionnaire.	32
4.6	Vérifier l'installation de Python.	34
4.7	Vérifier l'installation pip.	34
4.8	Installation de l'environnement virtuel.	34
4.9	Création et activation de l'environnement virtuel.	34
4.10	Istallation de Django.	35
4.11	Le contenu du dossier du projet.	36
4.12	Démarrage de serveur.	37
4.13	Migration.	38
4.14	Page de Regidter.	39
4.15	Page de Login.	39

4.16	Création des notes.	40
4.17	Listes des notes	41
4.18	Faire des homeworks.	42
4.19	Listes des homeworks.	42
4.20	Page de profil qui affiche des TO-DOs et des homeworks incomplets.	43
4.21	Utiliser l'API dictionnaire.	44
4.22	Rendre les résultats de recherche Youtube en utilisant l'API.	45
4.23	Rechercher des livres via Google API.	46
4.24	Rechercher sur Wikipedia via API.	46
4.25	Opération commune de conversion des mesures.	47

Liste des tableaux

3.1 Applications possibles des trois types de cartes utilisées	15
--	----

Introduction Générale

L'éducation joue un rôle essentiel dans notre société et, de nos jours, les étudiants ont accès à une multitude de ressources en ligne pour faciliter leur apprentissage. Cependant, cela peut également entraîner confusion et stress, car les étudiants doivent jongler entre différentes plateformes pour prendre des notes, noter leurs devoirs, effectuer des recherches et trouver des informations pertinentes.

Pour remédier à cette situation, le développement d'une application web regroupant toutes ces fonctionnalités en un seul endroit peut constituer une solution pour aider les étudiants à mieux s'organiser et à améliorer leur productivité. Cette application web permettra aux étudiants de prendre des notes, de gérer leurs devoirs, de faire des recherches sur Wikipédia, de consulter des dictionnaires et même de trouver des vidéos éducatives sur YouTube, le tout à partir d'un tableau de bord convivial.

L'objectif de ce projet est donc de créer un cartable numérique (une application web) qui offre aux étudiants une solution complète pour leur travail académique. Cette application leur permettra de se concentrer sur leur apprentissage en leur fournissant un outil efficace pour prendre des notes, suivre leurs devoirs et trouver des informations pertinentes. De plus, l'application sera conçue pour être facile à utiliser et à naviguer, afin que les étudiants puissent se concentrer sur leur travail plutôt que d'apprendre à utiliser une nouvelle plateforme.

En résumé, la création d'un cartable numérique offrira une solution pratique pour aider les étudiants à s'organiser efficacement et à améliorer leur productivité en leur offrant un accès centralisé à toutes les fonctionnalités nécessaires à leur travail académique. Notre mémoire est structuré en trois chapitres distincts :

Le premier chapitre abordera les concepts fondamentaux liés au Web.

Dans le deuxième chapitre, nous avons présenté la conception de notre application que nous avons modélisé avec le langage UML.

Dans le troisième chapitre, nous avons présenté notre application, en mentionnant les outils de programmation que nous avons utilisés, quelques interfaces graphiques y seront présentés.

Enfin, nous concluons notre travail par une conclusion générale qui récapitule les principaux résultats et apports de notre recherche. Nous abordons également quelques perspectives pour de futures améliorations ou extensions de l'application.

Chapitre 1

Notions de base sur le web

1.1 Introduction

Le présent chapitre met l'accent sur les concepts fondamentaux du web. Il explorera les différents éléments qui composent le Web, tels que les sites web, les serveurs web et les applications web. De plus, ce chapitre abordera les différents types de programmation web, tels que la programmation côté client et côté serveur, ainsi que le concept de cartable numérique et les systèmes de gestion de base de données.

1.2 Vue d'ensemble

En 1989, Tim Berners-Lee, un scientifique britannique travaillant au CERN, a inventé le World Wide Web (WWW). Il a créé le premier navigateur web, le premier serveur web et les premières pages web, qui étaient simplement des documents HTML. La première page web jamais créée était une page d'accueil qui expliquait ce qu'était le WWW et comment l'utiliser. Depuis, le WWW est devenu de plus en plus populaire et a connu une évolution incluant de nouvelles fonctionnalités telles que les moteurs de recherche, les réseaux sociaux et les applications mobiles. [8]

1.3 Technologies Web

Les technologies Web font référence aux outils et protocoles utilisés pour la communication entre différents appareils sur Internet [9]. Ils incluent des langages de balisage tels que HTML, XML et XHTML, des packages multimédias et des API Web. [10] [11]. Les technologies Web sont utilisées pour créer et développer des sites Web et des applications

Web. Certains des navigateurs Web les plus populaires incluent Netscape, Ms EDGE, Google chrome , Mozilla Firefox et Opera [12]. comme illustré dans la figure suivante :

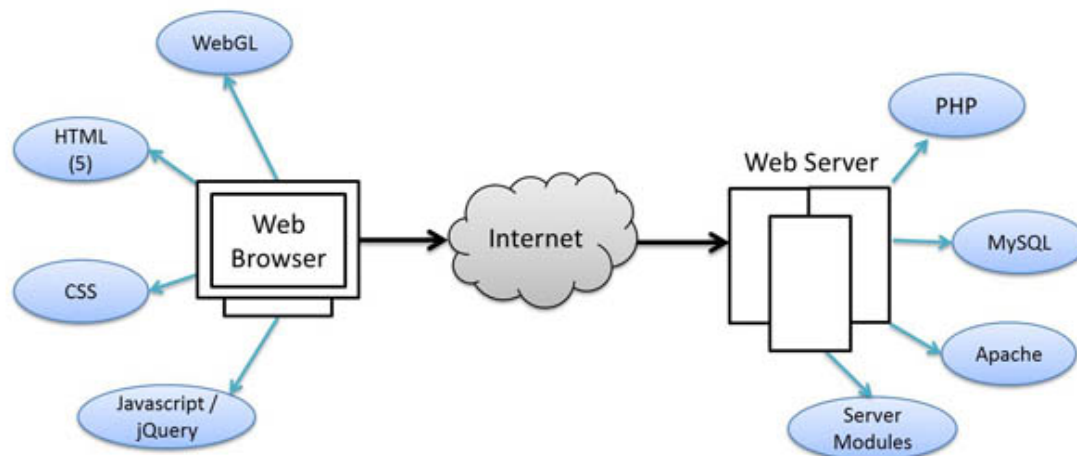


Figure 1.1: Aperçu de la technologie Web de base [1]

1.3.1 Site web

Un site Web est une collection de fichiers accessibles via une adresse Web, qui traitent d'un sujet ou d'un thème spécifique et sont gérés par une personne ou une organisation particulière. La première page du site est appelée la page d'accueil. Les sites Web sont hébergés sur des serveurs connectés au réseau Web et peuvent fournir des informations demandées par des utilisateurs du monde entier, 24 heures sur 24, 7 jours sur 7. Pour formater et présenter l'information, ainsi que pour offrir des fonctionnalités de navigation pour faciliter la navigation des utilisateurs à l'intérieur du site et sur le Web, les sites Web utilisent généralement le langage HTML [13].

1.3.2 Serveur web

Un serveur informatique est l'un des éléments clés d'un mode de communication client-serveur, dans lequel un logiciel client envoie une requête à un logiciel serveur qui répond en suivant un protocole de communication. Le terme "serveur informatique" fait également référence à l'ordinateur qui héberge ces logiciels serveurs, et qui est connecté à un réseau informatique permettant aux logiciels clients d'y accéder. Les serveurs fournissent une variété de services tels que le stockage de fichiers, le transfert de courrier électronique, l'hébergement de sites Web, etc. Il est possible pour un ordinateur ou un logiciel d'être à la fois client et serveur. La connexion client-serveur utilise des protocoles de communication,

tels que TCP/IP, qui est le protocole le plus utilisé sur Internet. [14] La figure suivante représente l'architecture des serveurs web :

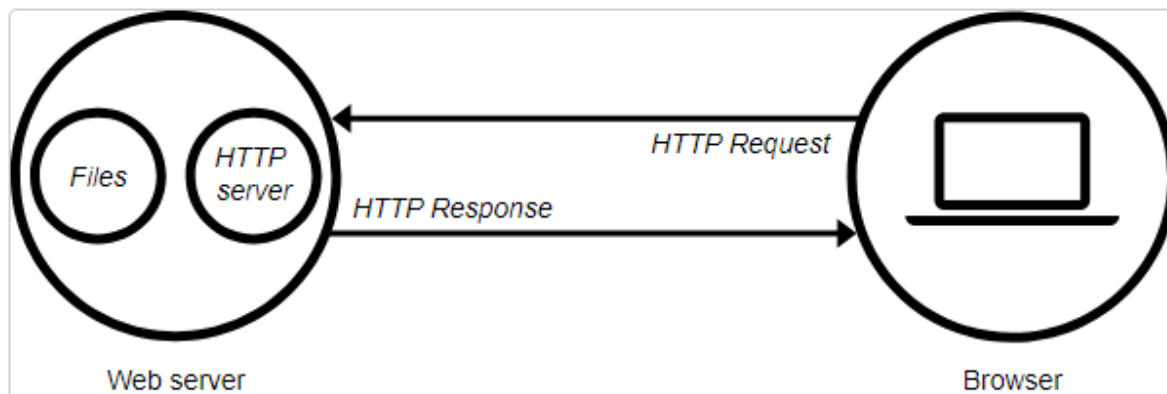


Figure 1.2: L'architecture de serveur web [2]

1.4 Programmation web

La programmation web est définie comme l'écriture, le balisage et le codage impliqués dans le développement web. Elle englobe également le contenu web, le scripting côté client et serveur, ainsi que la sécurité réseau. La programmation web diffère de la simple programmation car elle nécessite une connaissance interdisciplinaire du domaine d'application, du scripting côté client et serveur, ainsi que de la technologie des bases de données. La programmation web peut être séparée en deux catégories : la programmation côté client et la programmation côté serveur.

1.4.1 Programmation client-side

La programmation côté client, également connue sous le nom de programmation front-end, est définie comme le nom général de tous les programmes qui s'exécutent sur le client (navigateur). Elle inclut une série de langages compréhensibles par le navigateur. En termes de fonctionnalités, ces langages sont utilisés pour créer des pages web interactives et dynamiques, envoyer des requêtes au serveur et récupérer des données à partir de celui-ci. Les langages prédominants utilisés sont HTML, CSS et JavaScript.

1.4.2 Programmation server-side

La programmation côté serveur, également connue sous le nom de programmation back-end, désigne simplement le nom général de tous les types de programmes qui s'exé-

cutent sur le serveur. En termes de fonctionnalités, la programmation côté serveur se charge du traitement des entrées utilisateur, de l’affichage des pages, de la structuration des applications web et de l’interaction avec les fichiers. Des exemples de langages utilisés en programmation côté serveur sont PHP, Python, ASP.NET, Java et une multitude d’autres langages de programmation couramment utilisés.

1.4.3 Application web

En informatique, une application Web (aussi connue sous le nom de site Web dynamique ou WebApp) est un logiciel applicatif que l’on peut utiliser à l’aide d’un navigateur Web. Tout comme les sites Web, une application Web est généralement hébergée sur un serveur et on peut y accéder en manipulant des widgets via un navigateur Web, grâce à un réseau informatique tel qu’Internet, un intranet, un réseau local, etc. Les applications Web font partie de l’évolution des technologies et des pratiques du Web, souvent appelée Web 2.0. [15] Cette figure représente l’architecte des applications web :

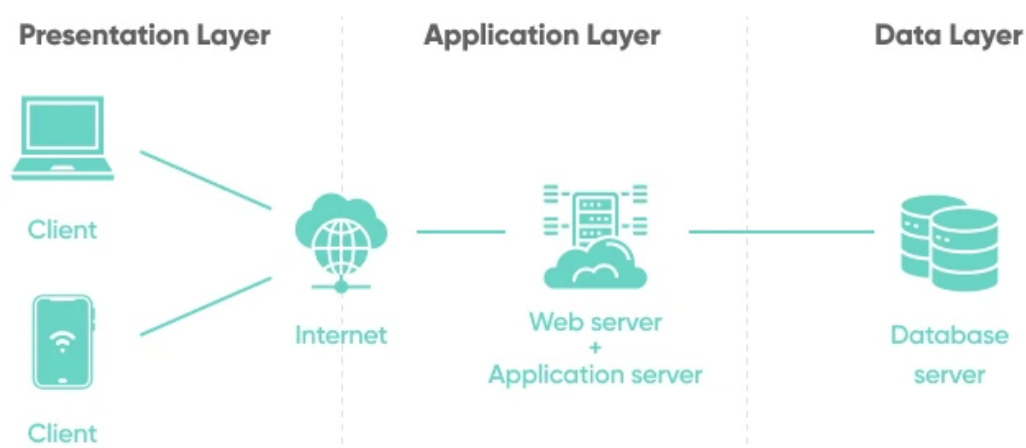


Figure 1.3: L’architecture des applications Web [3]

1.4.4 Fonctionnement d’une application web

Le fonctionnement d’une application Web peut être découpé en plusieurs étapes. Tout d’abord, l’utilisateur accède à l’application à travers un navigateur Web ou une application mobile, ce qui entraîne une demande au serveur Web via Internet. Cette étape initiale peut inclure la mise en place de mesures de sécurité, comme des pare-feux ou des courtiers de sécurité d’accès au cloud, ainsi que l’utilisation d’équilibreur de charge pour optimiser la distribution des demandes.

Ensuite, le serveur Web envoie la demande au serveur d’applications Web, qui exécute

la tâche demandée, telle que l'interrogation d'une base de données ou le traitement de données, puis génère les résultats correspondants. Le serveur d'applications Web renvoie ensuite ces résultats au serveur Web.

Enfin, le serveur Web fournit les informations demandées au client, qu'il s'agisse d'un ordinateur de bureau, d'un appareil mobile ou d'une tablette, et ces informations apparaissent sur l'écran de l'utilisateur. Ces différentes étapes sont essentielles pour garantir un fonctionnement fluide et sécurisé des applications Web [16]. Cette figure représente comme fonctionne une application web :

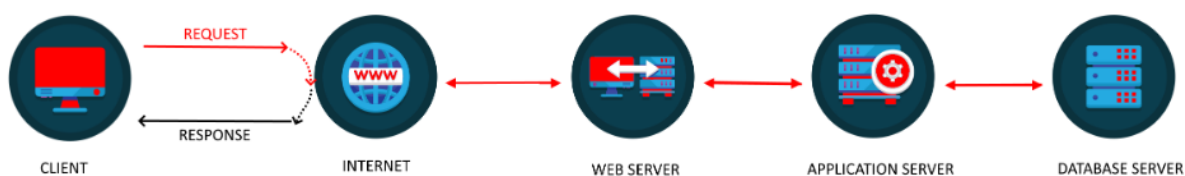


Figure 1.4: Comment fonctionne une application web [4]

1.5 Environnement de développement d'applications

Un environnement de développement d'applications (EDA) est le matériel, le logiciel et/ou les ressources informatiques nécessaires à la création de logiciels. L'EDA comprend l'infrastructure matérielle de base, telle que les serveurs, les ordinateurs et les appareils mobiles, qui hébergeront l'application. Ils sont combinés avec les ressources de génie logiciel, telles que l'environnement de développement intégré (IDE) d'un langage de programmation, ainsi que d'autres utilitaires logiciels d'évaluation des performances.

1.6 Système de Gestion de Base de Données (SGBD)

De manière générale, un SGBD est un outil qui permet de collecter, organiser et gérer des informations de manière efficace pour qu'elles puissent être facilement consultables, mises à jour et gérées. Avec un SGBD, il est possible de créer, de mettre à jour ou de supprimer des données. Les informations stockées dans une base de données peuvent être facilement consultées grâce à des requêtes qui permettent à l'utilisateur de faire des recherches. Les bases de données sont généralement stockées sur des serveurs et peuvent être migrées à tout moment. Il existe différents types de bases de données, tels que les bases de données hiérarchiques, réseau, orientées texte, etc., qui ont été développés pour répondre à des besoins spécifiques. [17]

1.7 Cartable numérique

Le cartable numérique est une plateforme en ligne qui regroupe toutes les ressources et fonctionnalités nécessaires aux étudiants pour faciliter leur apprentissage. Cette application leur permet de prendre des notes, gérer leurs devoirs, effectuer des recherches et trouver des informations pertinentes au même endroit. Ainsi, les étudiants n'ont plus besoin de perdre du temps à chercher des informations sur différentes plateformes en ligne, ce qui peut être source de confusion et de stress. Le cartable numérique a été conçu pour être pratique et convivial, dans le but d'aider les étudiants à s'organiser et à améliorer leur productivité académique. [18]

1.8 Conclusion

En effet, la conception d'une application web requiert une analyse conceptuelle minutieuse, qui nous permettra d'obtenir une vue d'ensemble du système à venir. Le chapitre suivant sera donc consacré à la conception de cette application.

Chapitre 2

Modélisation Conceptuelle

2.1 Introduction

L'étude conceptuelle est une étape cruciale dans de nombreux domaines, notamment dans la recherche, le développement de produits et la résolution de problèmes. Pour mener à bien cette étude, il est essentiel de comprendre les outils qui peuvent être utilisés pour analyser, représenter et interpréter les concepts.

Ensuite, nous aborderons les rôles spécifiques de ces outils dans l'étude conceptuelle. Nous discuterons de leur capacité à faciliter la communication, à organiser les connaissances, à identifier les lacunes conceptuelles et à soutenir la prise de décision.

Dans ce chapitre, nous explorerons les différentes définitions et les rôles des outils utilisés dans l'étude conceptuelle. Nous examinerons en détail les méthodes et les techniques qui permettent de capturer et de structurer les idées, les informations et les relations entre les concepts.

Enfin, nous mettrons en évidence les avantages et les limites de chaque outil, en soulignant les situations dans lesquelles ils sont les plus appropriés et efficaces.

2.2 Spécification des besoins

Dans le contexte de la conception, les spécifications des besoins se concentrent sur la traduction des exigences fonctionnelles et non fonctionnelles en caractéristiques techniques et en solutions concrètes.

2.2.1 Besoins Fonctionnels

Voici une liste des exigences fonctionnelles pour votre projet d'application Web :

- Gestion de projet : L'application devrait permettre aux étudiants de créer des projets, d'ajouter des devoirs, de fixer des délais et de suivre le progrès de leurs projets.
- Recherche intégrée : L'application doit avoir une fonction de recherche qui permet aux étudiants d'accéder rapidement à des ressources externes telles que Wikipédia et YouTube.
- Gestion des notes : L'application devrait permettre aux étudiants de prendre des notes sur leur recherche et de les organiser en catégories ou sections pour une consultation ultérieure.
- Tableau de contenu unique : L'application doit offrir une table de contenus unique et complète qui compile toutes les fonctionnalités pour une utilisation facile et une vue complète de l'état du projet.
- Sécurité : L'application doit garantir que les données de l'utilisateur sont sécurisées en utilisant des protocoles de sécurité avancés tels que le cryptage des données.

2.2.2 Besoins non fonctionnels

- Performances : Pour que les utilisateurs aient une expérience fluide et réactive, l'application doit être rapide et efficace.
- Fiabilité : Pour éviter la perte de données ou les interruptions inattendues, l'application doit être fiable et stable.
- Évolution : Le programme doit pouvoir s'adapter à l'utilisation croissante et au changement en réponse aux besoins changeants des utilisateurs.
- Accessibilité : le programme doit être utilisable par tous les utilisateurs, y compris ceux ayant des besoins spéciaux tels que les aveugles ou les personnes à mobilité réduite.
- Sécurité : L'application doit être sécurisée pour protéger les données de l'utilisateur contre le vol ou l'accès non autorisé.
- Utilisabilité : Le programme doit être simple à utiliser et intuitif pour permettre aux utilisateurs de le récupérer rapidement et sans difficulté.
- Maintenance : Pour assurer une utilisation optimale et un fonctionnement continu, l'application doit être simple à entretenir et à mettre à jour.

2.3 Le choix de méthode

Il existe diverses méthodes de développement logiciel qui utilisent l'UML, telles que UP, RUP, TTUP, UPagile, XP, 2TUP, etc. Dans notre cas, nous avons opté pour la

méthode UP (Unified Process) comme choix principal.

2.3.1 Définition du processus unifié

Le Processus Unifié (PU) est une approche de développement itératif et incrémentiel qui met l'accent sur l'architecture et les cas d'utilisation. Il utilise un langage de modélisation unifié et respecte le méta modèle d'ingénierie des processus système. Le PU peut être mis en œuvre pour divers systèmes logiciels, qu'ils soient simples ou complexes sur le plan technique et de gestion, et ce, dans divers domaines et cultures organisationnelles. [19]

UP répond aux préoccupations suivantes :

- Qui participe au projet ?
- Qu'est-ce qui est produit tout au long du projet, exactement ?
- Comment doit-il être achevé ?
- Quand chaque livrable est-il achevé ?

2.3.2 Les caractéristiques de processus

1. UP utilise UML pour la modélisation.
2. UP est guidé par les cas d'utilisation.
3. UP met l'accent sur l'architecture.
4. UP suit une approche itérative et incrémentale.
5. UP est basé sur des composants.

2.3.3 Cycle de vie d'un processus

Le cycle de vie d'un processus unifié (PU) est une méthode de gestion de projet utilisée dans le développement de logiciels. Il divise le développement en phases distinctes, chacune ayant ses propres objectifs et activités spécifiques. Voici les principales phases du cycle de vie d'un processus unifié :

- d'inception : le projet est planifié et mis en œuvre.
- l'élaboration : les risques majeurs sont traités en priorité, ce qui entraîne des cycles plus longs et plus étroitement surveillés.
- construction : consiste à développer toutes les fonctionnalités du projet.
- transition : se concentre sur la préparation du lancement du projet. [20]

Comme le montre cette figure :

2.3.4 Les activités

Les activités clés du processus unifié comprennent :

1. Expression des besoins : la définition des objectifs du projet .
2. Analyse : la définition des exigences .
3. Conception : la création d'un modèle de conception .
4. Implémentation : la création de code .
5. Test : la réalisation de tests et la mise en place de l'environnement de production .

2.4 Conclusion

En conclusion, ce chapitre sera une ressource précieuse pour tous ceux qui souhaitent approfondir leur compréhension des concepts clés d'un domaine donné. Les outils présentés ici nous aideront à organiser nos idées, à clarifier nos pensées et à construire des fondations solides pour notre travail futur.

Chapitre 3

Conception Détaillée

3.1 Introduction

La phase de conception constitue la première étape cruciale pour la réussite d'un projet. Elle a pour objectif de formaliser de manière claire et non ambiguë le fonctionnement futur du système afin de faciliter sa réalisation. Pour ce faire, diverses méthodes existent pour formaliser les étapes préliminaires du développement.

Dans ce chapitre, nous allons procéder à un premier découpage de notre application, définir les différentes étapes de conception, identifier les relations entre les cas d'utilisation, et élaborer certains diagrammes UML (Unified Modeling Language) tels que le diagramme de cas d'utilisation pour spécifier les besoins de notre système, le diagramme de séquence pour décrire les interactions entre les acteurs et notre système, ainsi que le diagramme de classe pour définir l'architecture statique de notre système.

3.2 Analyse conceptuelle

UML est un langage graphique conçu pour représenter un système spécifique en fonction d'une perspective spécifique .

3.2.1 Le langage UML

L'Unified Modeling Language (UML) est un ensemble de diagrammes intégrés utilisés par les développeurs informatiques pour visualiser les objets, les états et les processus dans un logiciel ou un système. Ce langage de modélisation permet de créer des modèles qui servent de base pour un projet, garantissant ainsi une architecture d'information structurée. Il aide également les développeurs à communiquer de manière compréhensible avec des

experts externes en présentant une description claire d'un système. UML est principalement utilisé dans le développement de logiciels orientés objet, mais avec les améliorations introduites dans sa version 2.0, il peut également être utilisé pour représenter des processus de gestion. [21]

3.2.2 Développement du langage de modélisation unifié (UML)

Avant l'introduction d'UML, la programmation orientée objet (POO) était déjà largement adoptée dans le développement logiciel. La POO repose sur le concept que tout est un objet, où les composants d'un programme interagissent entre eux en envoyant des messages. Chaque objet est une instance d'une classe supérieure qui définit son comportement. Les objets sont composés de données et de code, avec les données organisées en attributs et le code déterminant les méthodes.

Dans les années 1980 et 1990, de nombreuses méthodes et langages ont été développés pour représenter la POO, créant ainsi une diversité de méthodes différentes. Afin d'unifier ces langages, James Rumbaugh, Grady Booch et Ivar Jacobson ont décidé de fusionner plusieurs langages existants en un langage commun et standardisé, ce qui a abouti à la création d'UML. [21]

3.2.2.1 Les points forts

UML est un langage officiel et standardisé.

- Cela permet d'obtenir une précision accrue dans la représentation des concepts.
- C'est un gage de stabilité, garantissant la cohérence et la fiabilité des modèles.
- Il favorise l'utilisation d'outils spécifiques qui simplifient la création et la manipulation des modèles.

UML est un puissant moyen de communication.

- Il offre un cadre structuré pour l'analyse des systèmes.
- Il facilite la compréhension des représentations abstraites complexes.
- Sa polyvalence et sa flexibilité en font un langage universel adaptable à différents domaines d'application.

3.2.2.2 Les points faibles

L'adoption et la pratique de l'UML nécessitent un apprentissage et une période d'adaptation.

Il est important de noter que l'UML ne couvre pas les processus, qui sont également essentiels à la réussite d'un projet.

3.2.3 Les diagrammes

1. Définition d'un diagramme : Un diagramme UML est une représentation graphique qui traite d'une partie spécifique du modèle. Chaque classe de diagramme UML a une structure et une sémantique spécifique.
2. Les différents types de diagrammes :
Pour notre projet, nous présenterons trois diagrammes d'UML : cas d'utilisation, de séquence et de classe.

Type de diagramme	Utilité
Diagramme de cas d'utilisation	Représente divers usages.
Diagramme de séquence	Représente le moment des interactions entre les objets.
Diagramme de classes	Représente les classes.

Tableau 3.1: Applications possibles des trois types de cartes utilisées

3.2.4 Diagrammes de cas d'utilisation

Les diagrammes de cas d'utilisation permettent de représenter le comportement anticipé d'un système à l'avenir. Ces modèles sont adaptés non seulement aux systèmes logiciels, mais aussi pour prédire les procédures au sein d'une entreprise, par exemple. Un cas d'utilisation implique un acteur, qu'il s'agisse d'une personne ou d'un système, ayant un objectif spécifique. Le diagramme est généralement nommé d'après cet objectif. Les différents cas d'utilisation au sein du système répondent aux objectifs de l'acteur. [21]

Le diagramme de cas d'utilisation ci-dessous présente de manière visuelle les différentes interactions entre les acteurs et le système dans le cadre de notre application :

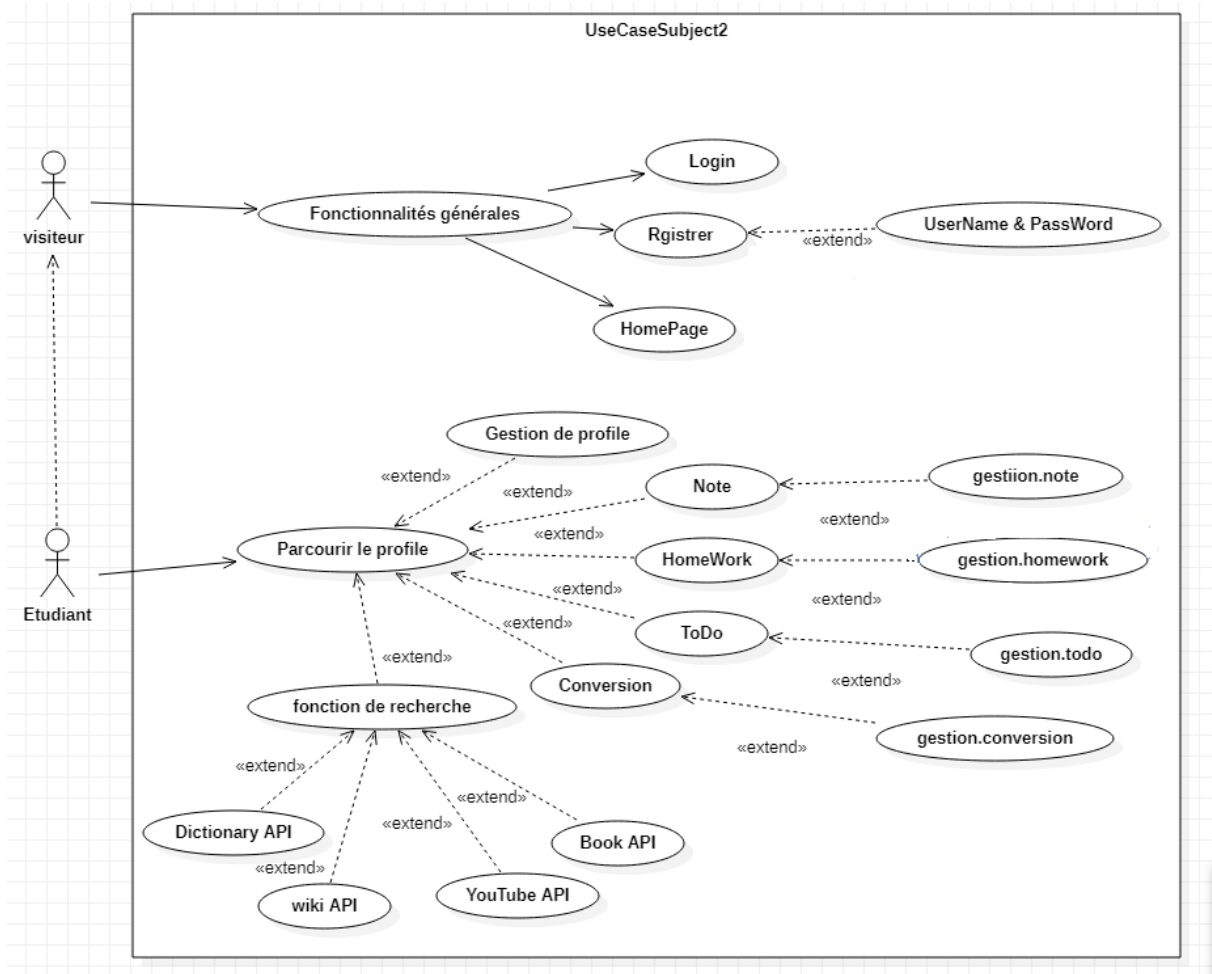


Figure 3.1: Diagramme de cas d'utilisation

3.2.4.1 identification de nos acteurs

Après avoir recensé l'ensemble des besoins , nous avons distingué les différents acteurs suivants :

- **Visiteur** : Cet utilisateur a la possibilité de parcourir la page d'accueil en général (home) ou de se connecter s'il a un compte ou de créer un compte.
- **Etudiant** : Cet utilisateur peut faire des hôtes d'applications à partir de login, ajouter, supprimer ou pré visualiser les opérations existantes(Notes,HomeWork,ToDo) ainsi que rechercher les moteurs de recherche existants(YouTube,Wikipedia,Déctionare).

3.2.4.2 Explication textuelle de cas d'utilisation

On a 3 cas d'utilisation principale :

- **Fonctions général** : Ce cas est deviser par 3 sous cas d'utilisation :Login,Register ,Home page. il permet à l'utilisateur d'accéder à la page d'accueil(home) et de voir le

contenu en général ainsi que la possibilité de se connecter ou d'ouvrir un compte dans l'application.

- **Parcourir l'application** : Ce cas est deviser par 5 sous cas d'utilisation : Gestion de Homework , il permet à l'utilisateur ou en particulier l'étudiant d'effectuer diverses fonctions de l'application de la suppression, l'ajout ou la pré-visualisation de diverses fonctionnalités existantes(ajouter,supprimer,consulter) en plus de la recherche d'informations et de contrôler le profil qui comprend les mêmes caractéristiques que Homework Todo.

- **Logout** : Ici, l'utilisateur peut se déconnecter et revenir à la page de connexion.

3.2.5 Diagrammes des séquences

Le diagramme de séquence est un type de diagramme d'interaction qui représente l'échange de messages entre les objets. Dans ce diagramme, les objets sont modélisés sous forme de lignes de vie, ce qui le rend similaire à d'autres diagrammes comportementaux tels que le diagramme d'activité. Cependant, contrairement à ces derniers, le diagramme de séquence ne vise pas à fournir une vue globale du comportement d'un système, mais plutôt à présenter en détail un scénario spécifique parmi plusieurs possibles. Il établit une chronologie et utilise une ligne en pointillés pour représenter le passage du temps.

Dans UML 2.0, les messages sont représentés de manière synchrone (flèche avec une tête pleine) ou asynchrone (flèche avec une tête ouverte). Les messages synchrones bloquent le canal de communication jusqu'à ce qu'ils reçoivent une réponse de l'objet cible. Ils définissent le comportement sous forme d'opérations synchrones. Les messages asynchrones contrôlent l'objet source appelant et peuvent être des opérations asynchrones ou des signaux (des paquets de données envoyés entre les actions). [21]

1. Diagramme séquence de "Register"

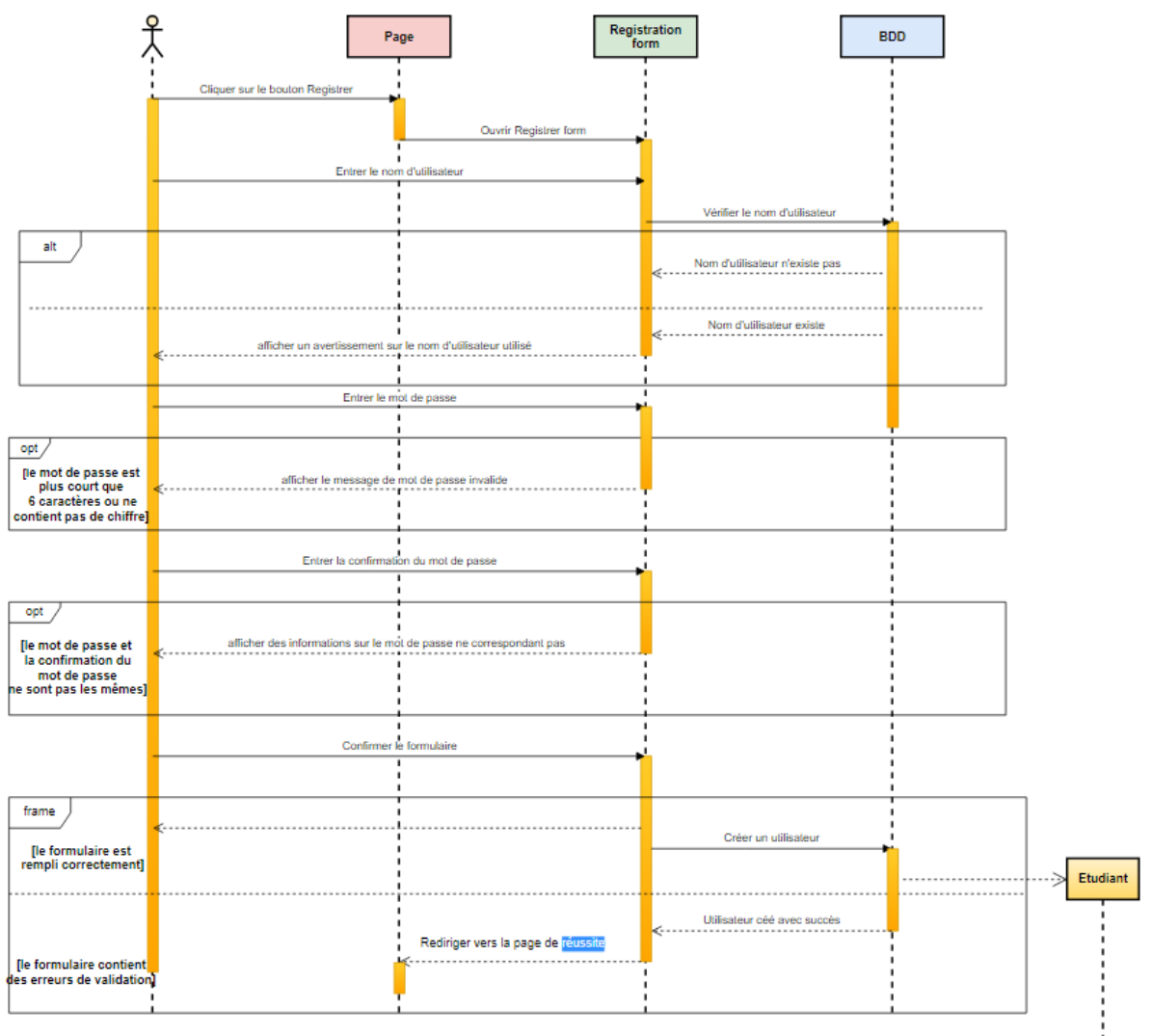


Figure 3.2: Diagramme de séquence du cas d'utilisation "Register"

Description textuelle :

L'utilisateur choisit leur situation, et il envoie leurs informations via l'interface à la logique (Register form) pour la validation. Si vous constatez des erreurs de rédaction et de forme, il enverra un message d'erreur, sinon ils sont valides, et l'inscription sera réussie.

2. Diagramme de séquence "Login"

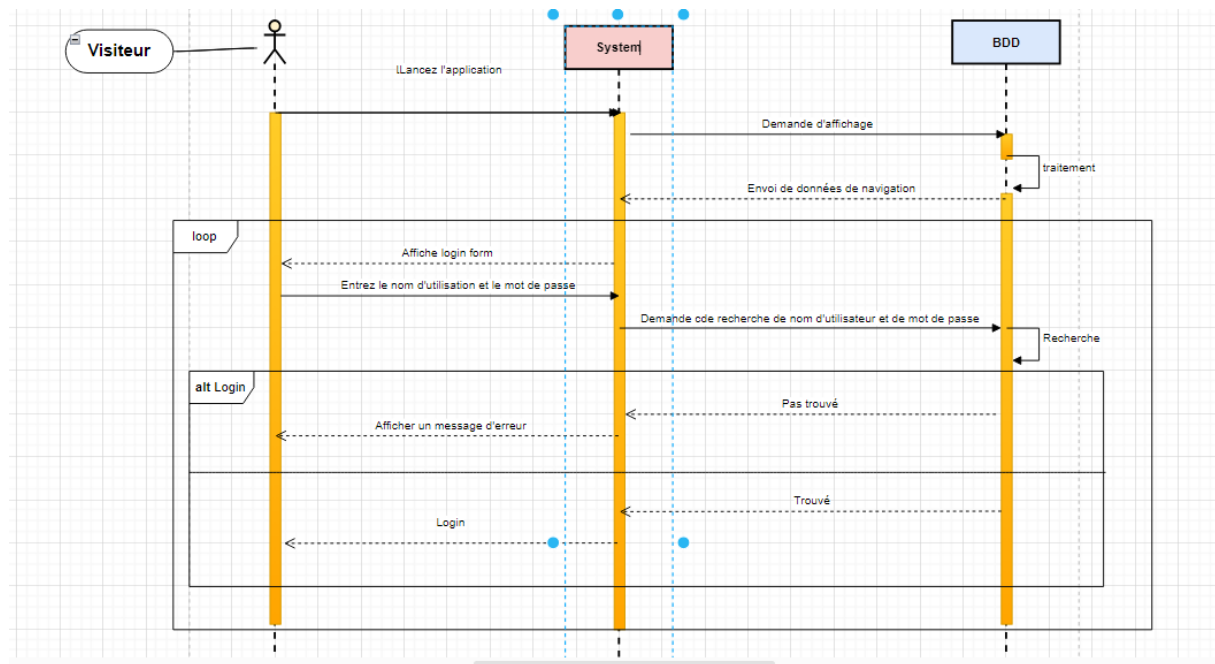


Figure 3.3: Diagramme de séquence du cas d'utilisation "Login"

Description textuelle :

L'utilisateur choisie leur situation, et il envoie leurs information via l'interface à le system pour la validation, si vous constatez des erreurs de rédaction et de forme, il enverra un message d'erreur ,sinon il s'assure que ses informations sont déjà existea partir la bdd ou bien ils sont valide ,l'inscription sera réussie si non le system envoie un message d'erreur vers l'interface.

3. Diagramme séquence "D'ajouter"

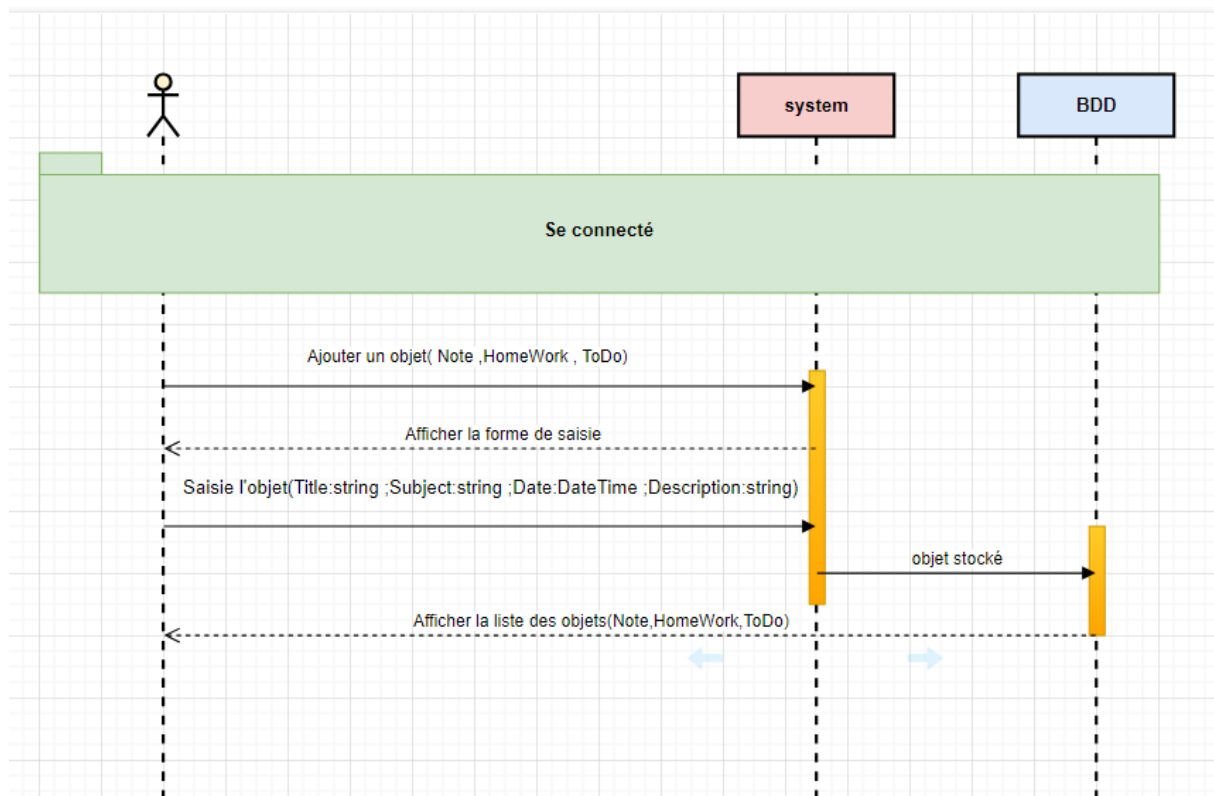


Figure 3.4: Diagramme de séquence du cas d'utilisation "D'ajouter"

Description textuelle :

Dans ce cas, l'étudiant peut ajouter ses commentaires(Notes,ToDo,HomeWork) en cliquant sur l'icône de fonctionnalité et en accédant à la page où il trouve la compétence pour saisir les information de fonctionnalité(titre, description date, is-finished) avec ces déférents types(string,date , booléen).

4. Diagramme séquence "Recherche"

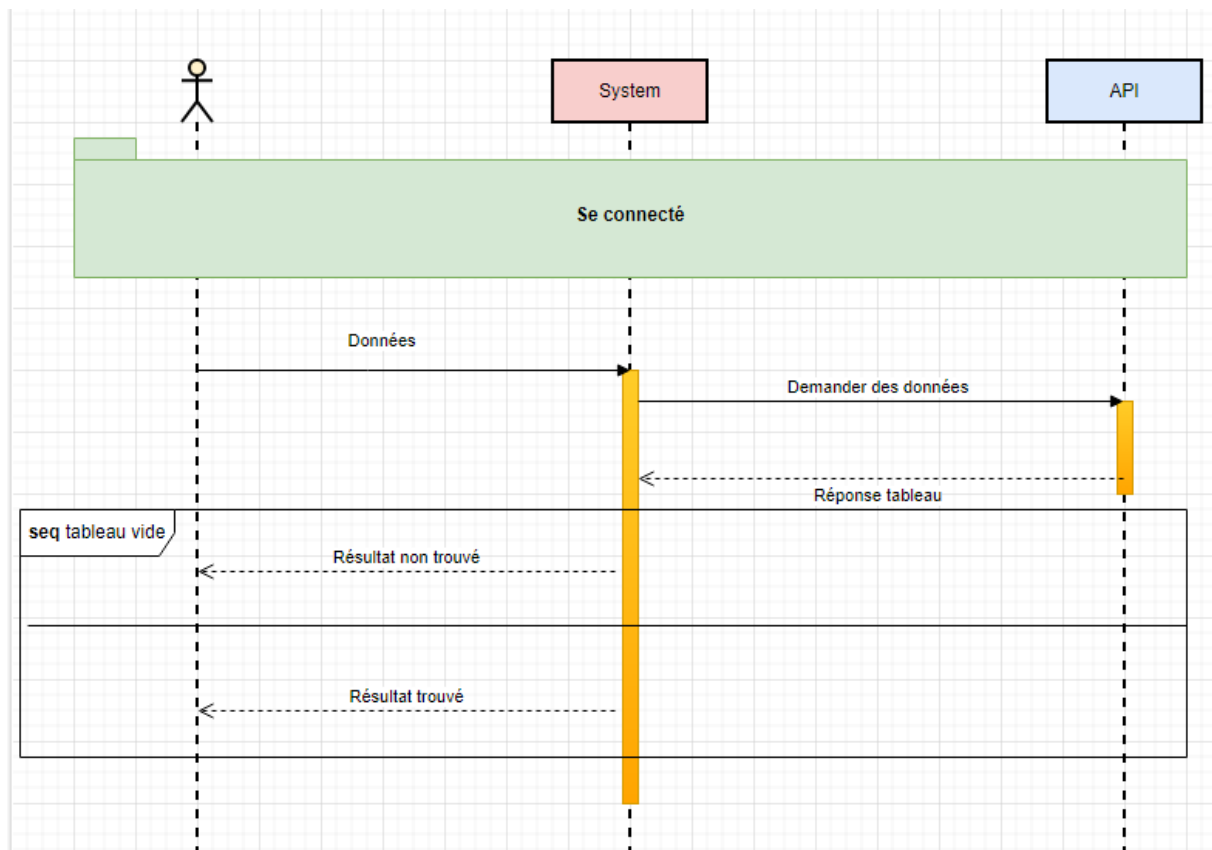


Figure 3.5: Diagramme de séquence du cas d'utilisation "Recherche"

Description textuelle :

L'utilisateur entre les informations qu'il souhaite rechercher dans l'interface de recherche, celui-ci envoie une requête vers le système pour demander des données, elle est demandée par l'API. L'API doit encore répondre au système, si les données existent il répond par une liste de données, sinon il répond par un message d'erreur.

5. Diagramme de séquence "Suppression"

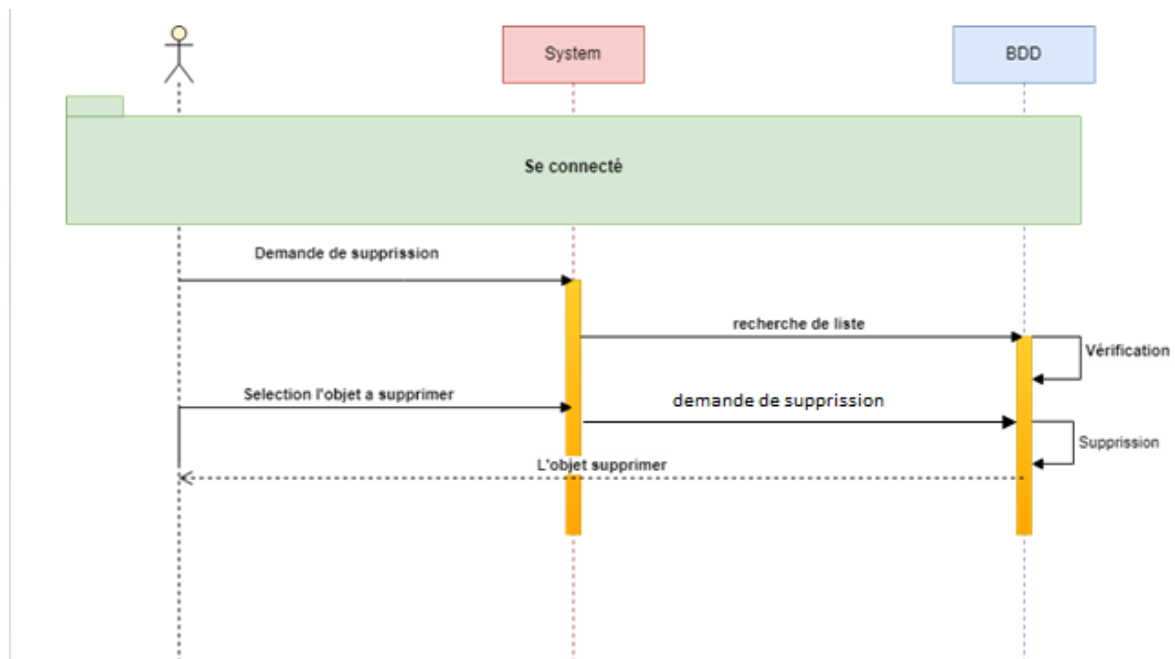


Figure 3.6: Diagramme de séquence du cas d'utilisation "Suppression"

Description textuelle :

Dans ce cas, l'étudiant peut supprimer ses commentaires (notes, todo, homework) en cliquant sur le bouton de fonctionnalité suppression. Le système recherche les données dans la base de données, puis supprime les données de l'interface.

6. Diagramme de séquence "Logout"

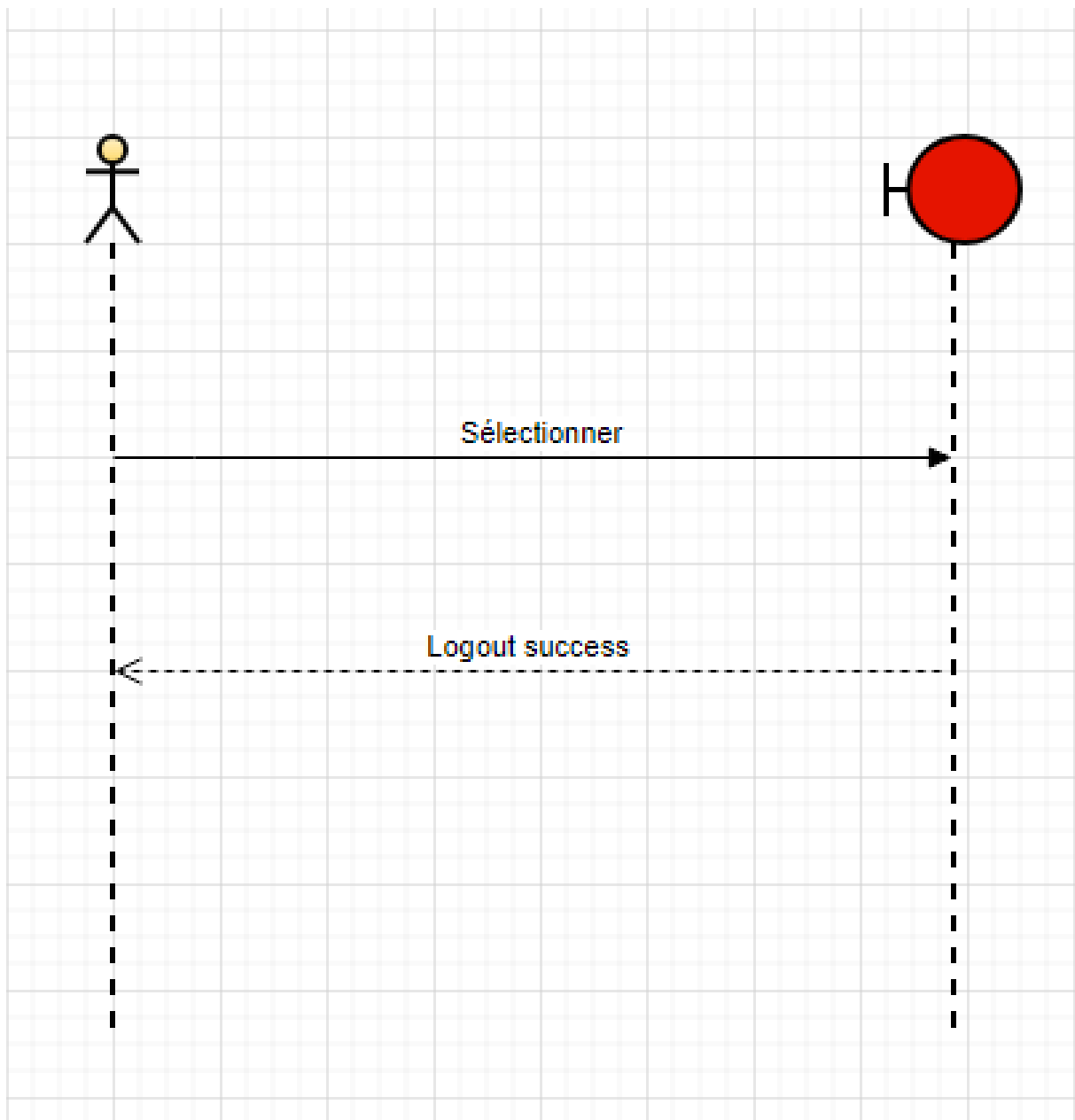


Figure 3.7: Diagramme de séquence du cas d'utilisation "Logout"

Description textuelle :

À cet étape , l'étudiant se déconnecte de l'application en appuyant sur le bouton pour envoyer un interaction au système pour succès le déconnecte.

3.2.6 Diagramme de classe

Lorsque des objets ont un comportement ou une structure commune, il est possible de les regrouper ou de les associer à une classe. La classe agit comme un élément de

simplification et de synthèse (abstraction) dans la représentation visuelle. Les classes et les objets sont connectés les uns aux autres à l'aide d'interfaces. Tous ces composants et leurs relations peuvent être représentés dans un diagramme de classes. Une classe est représentée par un rectangle dans lequel ses attributs et méthodes sont spécifiés. [21]

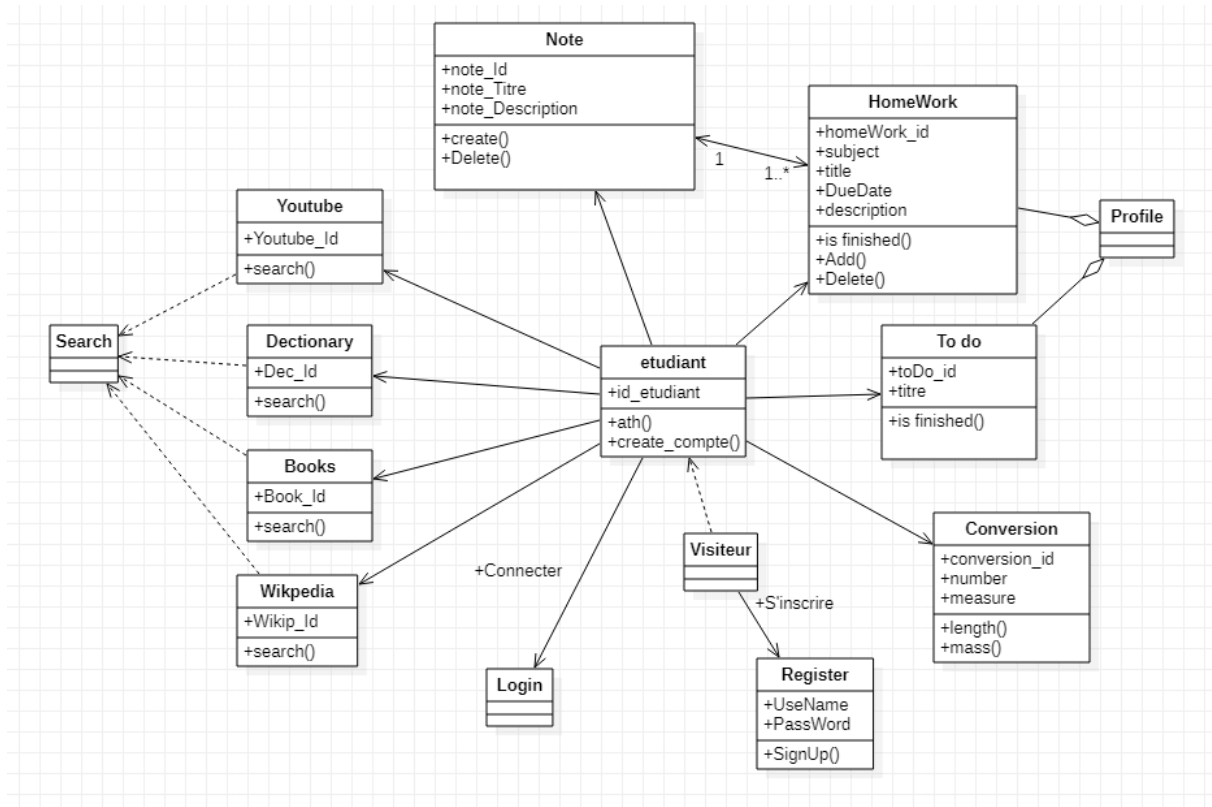


Figure 3.8: Diagramme de Classe

3.2.7 Le schéma relationnel

Nous avons suivi des règles strictes et précises pour traduire le contenu conceptuel du diagramme de classe en modèle relationnel afin de réaliser ce passage. Ces normes sont :

3.2.7.1 Règle de passage du diagramme de class au modèle relationnel

En se basant sur les règles suivantes, nous avons pu passer du diagramme de classes au modèle relationnel :

Règle 1 : Transformation des classes

Chaque classe est transformée en une table. Chaque attribut de classe se transforme en un champ de table et l'identifiant de classe devient la clé primaire.

Règle 2 : Association de plusieurs personnes à plusieurs personnes

Une association avec la forme $(?..*)$ à $(?..*)$ produit trois tables, dont la table dérivée de l'entité association aura les clés primaires des deux autres tables en tant que clé primaire, ainsi que ses propres attributs s'il s'agit d'une classe d'association.

La règle 3 : Stipule l'association d'une personne à plusieurs personnes

Une table dérivée de l'entité de la cardinalité (1..1) est créée par toute association binaire $(?..1)$ à $(?..*)$, et sa clé primaire est déposée comme clé étrangère dans la table dérivée de l'entité de la cardinalité $(?..*)$.

3.2.7.2 Le modèle relationnel

Home ($note_i d, etudiant_i d, YouTube_i d, Wikip_i d, Book_i d, Dec_i d, Profile_i d, homeWord_i d, toDo_i d, conversion_i d$).

Note ($note_i d, note_i title, note_i description$).

Search ($YouTube_i d, Wikip_i d, Book_i d, Dec_i d$).

YouTube API ($YouTube_i d$).

Wikipedia API ($Wikip_i d$).

Book API ($Book_i d$).

Dictionnaire API ($Dec_i d$).

Profile ($homeWord_i d, toDo_i d$).

HomeWork ($homeWord_i d, subject, title, DueDate, description$).

ToDo ($toDo_i d, title$).

Register ($UserName, Password$).

login ($etudiant_i d$).

Etudiant ($etudiant_i d$).

Visiteur ($etudiant_i d$).

3.3 Conclusion

Dans ce chapitre, nous avons exposé le processus de développement UP, le langage de modélisation UML, et réalisé la modélisation des différents diagrammes tels que les diagrammes de cas d'utilisation, les diagrammes de séquences et les diagrammes de classes. Cette étape de modélisation nous a permis d'obtenir une vision globale du comportement théorique des fonctionnalités offertes par notre application. Cette base théorique nous guidera lors du développement de l'application, qui sera abordé dans le chapitre suivant.

Chapitre 4

implémentation

4.1 Introduction

Ce chapitre a pour objectif de introduire au développement d'applications web et de fournir une explication détaillée des outils utilisés pour la construction de notre cartable numérique. Nous aborderons ce sujet avec un ton formel et professionnel.

4.2 Outils utilisé dans le server-side

Les outils utilisés dans ce projet pour la programmation côté serveur ou back-end comprennent le framework web, le langage de programmation serveur, la base de données, les serveurs web, etc. Ces outils sont essentiels pour assurer le bon fonctionnement du système. Ces outils sont le framework Django, le langage de programmation Python et la base de données SQLITE3. Chacun de ces outils a été choisi en fonction de ses fonctionnalités et de sa compatibilité avec les autres outils utilisés dans le projet. Une description détaillée de ces outils sera donnée dans les sections suivantes.

4.2.1 Django Framework

Django est un framework web open-source écrit en Python qui permet aux développeurs de construire des applications web robustes et évolutives rapidement et efficacement comme YouTube et Spotify. Django suit le principe du modèle-vue-contrôleur (MVC) qui sépare la logique de l'application en trois composants distincts pour faciliter le développement et la maintenance. Le modèle représente les données et leur manipulation, la vue affiche l'interface utilisateur et le contrôleur gère les entrées utilisateur et les actions en conséquence. Django fournit également un ORM (Object-Relational Mapping) qui permet

aux développeurs de travailler avec des bases de données relationnelles en utilisant des objets Python plutôt que des requêtes SQL. En outre, Django a une communauté active et de nombreux packages tiers sont disponibles pour étendre ses fonctionnalités. [22]

MVC (Model View Controller) : Le modèle MVC est une structure de développement de produit qui résout le défaut de l'approche traditionnelle consistant à écrire tout le code dans un seul fichier, en utilisant plutôt des fichiers distincts pour les différents aspects de l'application Web ou du site Web. Cette architecture est composée de trois éléments clés : le modèle, la vue et le contrôleur. [23] La figure (3.1) présente le model mvc :

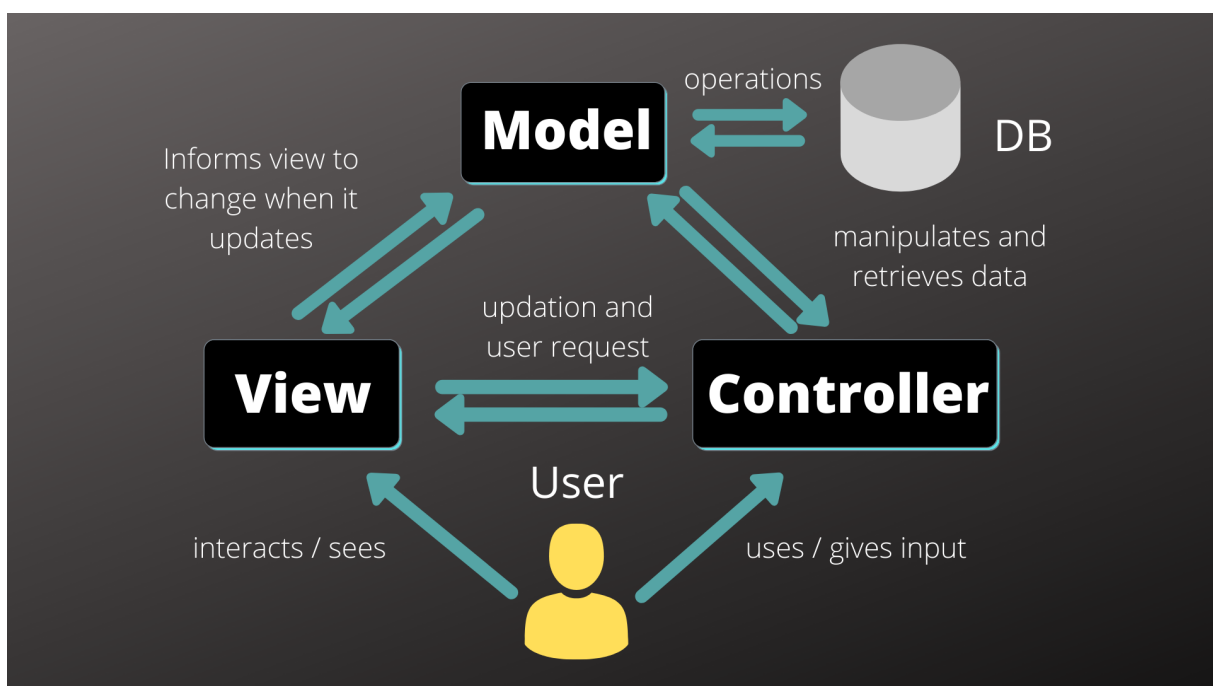


Figure 4.1: Model View Controller (MVC) [5]

L'ORM (Object-Relational Mapper) : ORM est un outil qui permet de faire la liaison entre les objets de programmation et les tables de la base de données relationnelle. Dans Django, il permet d'interagir avec la base de données de la même manière que l'on interagit avec SQL. [24]

Comme le montre dans cette figure(3.2) :

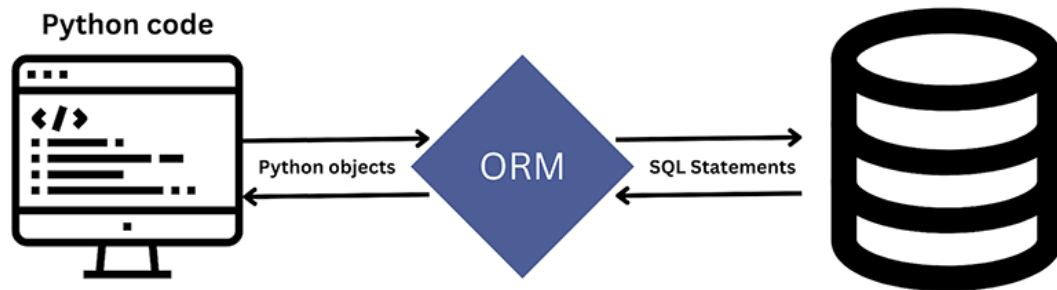


Figure 4.2: Object-Relational Mapper(ORM) [6]

4.2.1.1 Pour quoi DJANGO

Django vise à faciliter le passage des applications de la phase de conception à l'achèvement dans un temps record. Il offre également une sécurité fiable qui aide les développeurs à éviter les erreurs courantes en matière de sécurité. En outre, la capacité de Django à évoluer rapidement et de manière flexible est utilisée par de nombreux sites web populaires. [25]

4.2.1.2 DJANGO request/response processus

Django utilise des objets de requête et de réponse pour maintenir l'état tout au long du système. Lorsqu'un utilisateur demande une page, Django crée un objet `HttpRequest` contenant des informations sur la demande. Ensuite, il charge la vue correspondante et passe l'objet `HttpRequest` comme premier argument à la fonction de vue. Chaque vue est chargée de renvoyer un objet `HttpResponse`. Ce guide décrit les API pour les objets `HttpRequest` et `HttpResponse`, qui sont définis dans le module `django.http`. [26] Comme illustré dans la figure(3.3) :

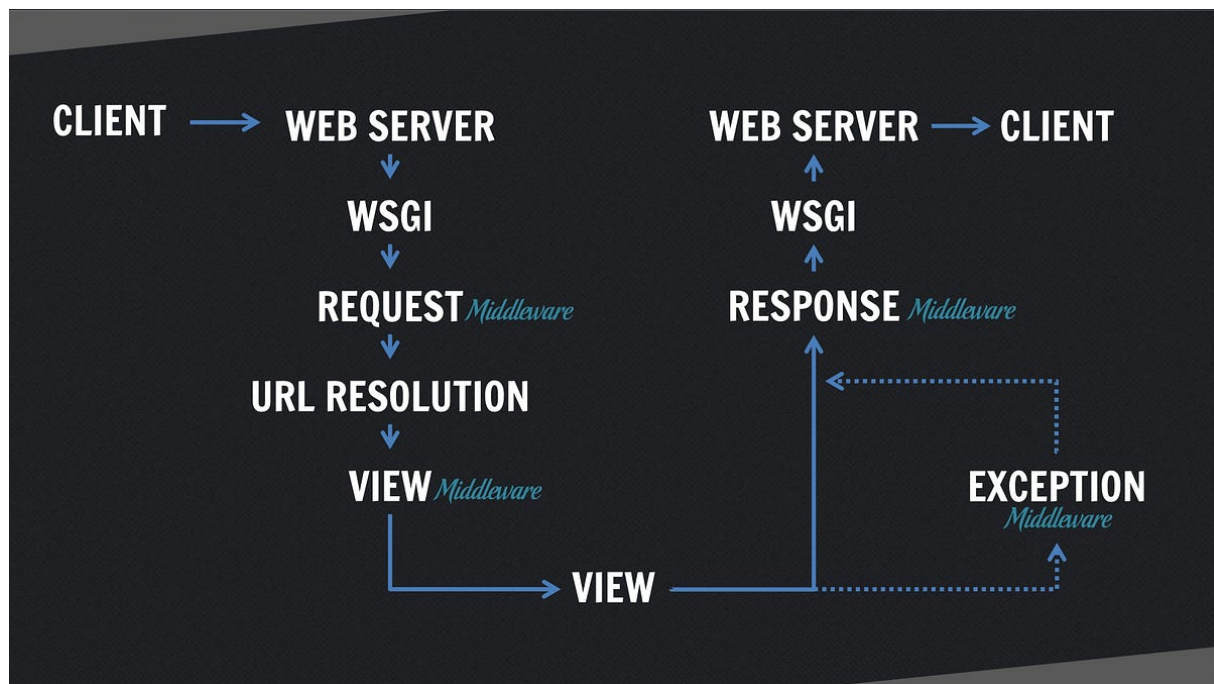


Figure 4.3: Django : Request/Response Cycle [3]

4.2.2 Python

Python, créé en 1991 par Guido van Rossum, est un langage de programmation interprété, haut niveau et orienté objet. Sa popularité croissante s'explique par sa simplicité, sa lisibilité et sa facilité d'apprentissage. Python trouve une large application dans divers domaines tels que le développement web, la science des données, l'intelligence artificielle et l'automatisation des tâches. [27]

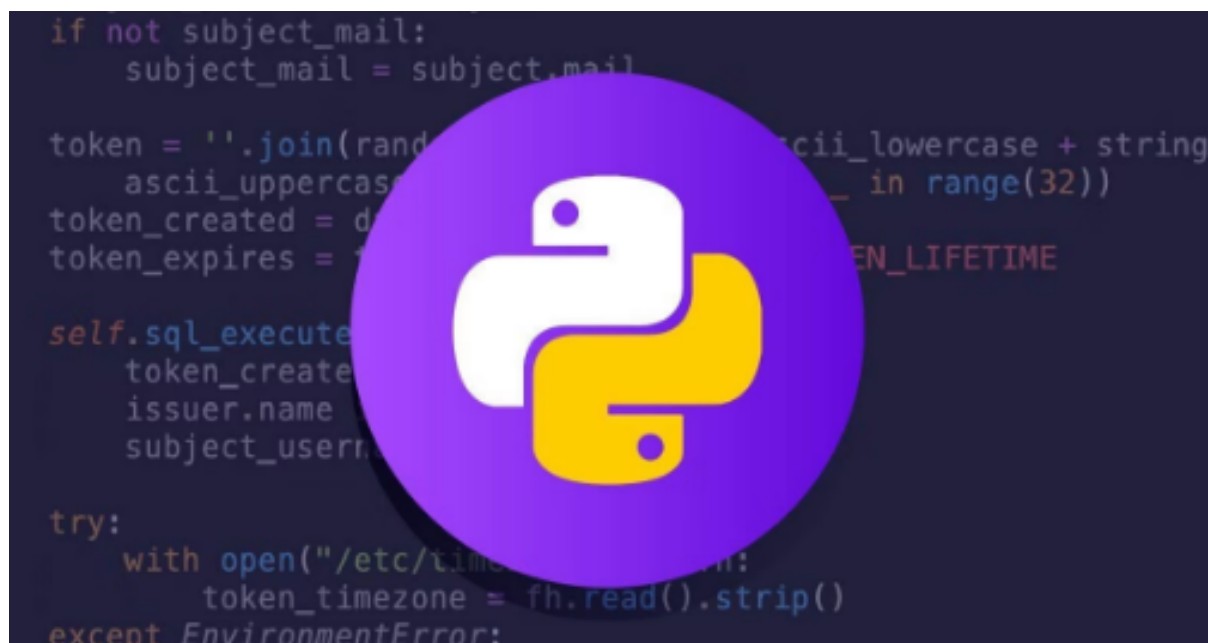


Figure 4.4: Langage "Python" [7]

4.2.3 Sqlite3

SQLite3 est un système de gestion de base de données relationnelle léger et efficace qui est largement utilisé dans le développement web. Par défaut, lorsque vous lancez votre projet de site web en utilisant les outils standards de Django (django-admin), celui-ci est configuré pour utiliser SQLite. Cette option est judicieuse car elle ne nécessite aucune configuration ou installation spécifique, ce qui en fait un choix très pratique. [28]

Pour configurer notre base de données SQLite3 dans Django, vous devez spécifier les paramètres de base dans le fichier settings.py de notre projet. Cela inclut la définition du moteur de base de données, nom, utilisateur, mot de passe, hôte et port. Une fois que nous avons configuré notre base de données SQLite3, nous pouvons commencer à construire notre application web en toute confiance en sachant que nous avons une solution de stockage de données fiable et évolutive à portée de main.

4.2.4 Web scraping

Le web scraping, également connu sous le nom d'extraction de données web, est une méthode qui consiste à collecter des données à partir de sites web en utilisant des programmes informatiques tels que des "bots" ou des "crawlers". Ces programmes parcourent les pages web, analysent leur contenu et extraient les informations pertinentes telles que du texte, des images, des liens, etc. Ces données peuvent ensuite être stockées dans une base

de données ou un fichier. Le web scraping est une technique largement utilisée dans divers domaines tels que le marketing, la recherche d'emploi, la surveillance concurrentielle, et bien d'autres. [29]

4.2.5 Bibliothèque youtube-search-python

La bibliothèque YouTube-search Python permet aux développeurs de se connecter à l'API de recherche YouTube et d'effectuer des recherches de vidéos, de chaînes et de listes de lecture, ainsi que de récupérer des métadonnées associées. [30]

Video-search est un package Python qui fournit un moyen simple de rechercher des vidéos sur YouTube en utilisant l'API de données YouTube . Pour utiliser la recherche vidéo dans projet Django, nous devons l'installer en utilisant pip. Ensuite, nous pouvons importer la classe VideoSearch du paquet et l'utiliser pour rechercher des vidéos basées sur des mots clés ou d'autres critères. Lorsque nous utilisons la bibliothèque de recherche vidéo , nous pouvons créer un formulaire avec des champs de saisie pour les mots-clés, le canal et le titre. Ensuite, vous pouvez utiliser la classe VideoSearch pour rechercher des vidéos en fonction des champs de saisie. Vous pouvez également ajouter des filtres à votre recherche, comme le nombre de résultats et l'ordre des résultats. Enfin, vous pouvez afficher les résultats de recherche sous forme de liste ou de grille à l'aide d'un modèle.

Video-search est un package Python qui fournit un moyen simple de rechercher des vidéos sur YouTube en utilisant l'API de données YouTube . Pour utiliser la recherche vidéo dans notre projet Django, nous devons l'installer en utilisant pip. Ensuite, nous pouvons importer la classe VideoSearch du paquet et l'utiliser pour rechercher des vidéos basées sur des mots clés ou d'autres critères.

4.2.6 Wikipedia API

L'API de Wikipédia est une interface de programmation qui permet aux développeurs d'interagir avec les données de l'encyclopédie en ligne. Elle offre des fonctionnalités telles que la recherche de pages, la récupération de contenu et de liens vers d'autres pages. [31]

Pour utiliser la bibliothèque Wikipedia dans votre application Django, nous pouvons commencer par installer le paquet "wikipedia" en utilisant pip. Une fois le paquet installé, vous pouvez l'importer dans vos vues ou modèles Django et utiliser les diverses fonctions fournies par le paquet pour interagir avec l'API Wikipedia. nous utilisons la fonction "wikipedia.page" pour obtenir un objet page Wikipedia pour un sujet spécifique, puis utiliser les différents attributs et méthodes de l'objet page pour récupérer des informations sur le sujet, telles que le résumé, le titre , les liens. Nous pouvons également

utiliser la fonction "wikipedia.search" pour rechercher des pages Wikipedia liées à une requête spécifique, puis utiliser la fonction "wikipedia.summary" pour obtenir un résumé de la page la plus pertinente. En utilisant la bibliothèque Wikipedia dans notre application Django, nous pouvons facilement enrichir votre contenu d'informations précieuses et améliorer l'expérience utilisateur.

4.2.7 API de dictionnaire avec l'API Google

L'API de dictionnaires Google offre une ressource précieuse aux développeurs Django souhaitant enrichir leur application avec des fonctionnalités de traduction et de recherche de mots. Cette API permet d'accéder à un large éventail de langues et fournit des définitions, des traductions et des synonymes pour les mots recherchés. [32]

Pour obtenir la définition d'un mot à l'aide de l'API, nous pouvons faire une requête GET à l'URL suivant :

```
url = "https://api.dictionaryapi.dev/api/v2/entries/en_US/"+text
```

Figure 4.5: API de dictionnaire.

Cela retournera un objet JSON contenant la langue détectée et la traduction du mot, ainsi que d'autres informations telles que la prononciation et des exemples. Vous pouvez alors extraire la définition de l'objet JSON et l'afficher dans votre modèle Django. En utilisant l'API Google Dictionary dans votre application Django, nous pouvons fournir à nos utilisateurs des définitions de mots multilingues précises et fiables.

4.2.8 Bibliothèque Crispy-form

Django-crispy-forms est une bibliothèque Python qui donne du style aux formulaires Django. Elle facilite la personnalisation des propriétés des formulaires (telles que la méthode, le bouton de soumission ou les classes CSS) côté serveur, sans nécessiter de réécriture dans le modèle. [33]

Pour utiliser Crispy Forms, nous devons l'installer en utilisant pip et l'ajouter au paramètre INSTALLED-APPS de notre projet Django. Nous pouvons ensuite créer nos formulaires en utilisant la balise crispy-forms dans nos modèles.

4.3 Outils utilisé dans le client-side

Les outils suivants ont été utilisés dans le projet pour la programmation côté client ou front-end.

4.3.1 HTML5

HTML5 is the latest version of Hypertext Markup Language, the code that describes web pages. It's actually three kinds of code : HTML, which provides the structure ; Cascading Style Sheets (CSS), which take care of presentation ; and JavaScript, which makes things happen. [34]

4.3.2 CSS3

Les feuilles de style en cascade (CSS) sont un langage de feuille de style utilisé pour décrire la présentation d'un document écrit en HTML ou XML (y compris les dialectes XML tels que SVG , MathML ou XHTML). CSS décrit comment les éléments doivent être rendus à l'écran, sur papier, dans la parole ou sur d'autres supports. [35]

4.3.3 JAVA SCRIPT

JavaScript (JS) is a lightweight, interpreted, or just-in-time compiled programming language with first-class functions. While it is most well-known as the scripting language for Web pages, many non-browser environments also use it, such as Node.js, Apache CouchDB and Adobe Acrobat. JavaScript is a prototype-based, multi-paradigm, single-threaded, dynamic language, supporting object-oriented, imperative, and declarative (e.g. functional programming) styles. [36]

4.4 Visuel studio code pour le web

L'IDE Visual Studio est un panneau de lancement créatif que vous pouvez utiliser pour modifier, déboguer et générer du code, puis publier une application. En plus de l'éditeur et du débogueur standard fournis par la plupart des IDE, Visual Studio inclut des compilateurs, des outils de complétion de code, des concepteurs graphiques et bien d'autres fonctionnalités pour améliorer le processus du développement de logiciels. [37]

4.5 Mettre en place un environnement de développement Django

L'environnement de développement correspond à une installation de Django sur votre ordinateur local que vous pouvez utiliser pour développer et tester des applications Django avant de les optimiser sur un environnement de production.

4.5.1 Configuration de l'environnement virtuel (virtualenv)

Le cadre virtuel est un environnement utilisé par Django pour exécuter une application. Il est conseillé de créer et d'exécuter une application Django dans un environnement distinct.

- Premièrement, Django nécessite Python et le pip package. Pour vérifier si ils sont installés sur le système, exécuter ces commandes à l'invite de commande :

```
python --version
```

Figure 4.6: Vérifier l'installation de Python.

```
pip --version
```

Figure 4.7: Vérifier l'installation pip.

- Python fournit un outil virtualenv pour créer un environnement Python isolé, par exécuter cette commande :

```
pip install virtualenv
```

Figure 4.8: Installation de l'environnement virtuel.

- Cela créera un environnement virtuel et un dossier nommé "env-(nom-d'env)" contenant des sous-dossiers et des fichiers similaires à ce qui suit :

```
py -m venv env_site
```

Figure 4.9: Création et activation de l'environnement virtuel.

4.5.2 Démarrage du Django projet

Pour démarrer notre application Django, nous avons suivi quelques étapes simples :

- Tout d'abord, nous avons utilisé pip pour installer Django en exécutant la commande :

```
cd env_site
```

puis

```
cd Scripts
```

puis

```
py -m pip install Django
```

Figure 4.10: Installation de Django.

- Nous avons créé un nouveau projet Django en utilisant la commande suivante :
"django-admin startproject (nom du projet)"
- Django crée un dossier avec le nom de projet sur notre ordinateur, avec un contenu comme ce qui suit :

```
Nom de projet /  
    manage.py  
Nom de projet /  
    __init__.py  
    asgi.py  
    settings.py  
    urls.py  
    wsgi.py
```

Figure 4.11: Le contenu du dossier du projet.

- Nous avons lancé le serveur de développement Django en utilisant la commande "python manage.py runserver" et nous avons testé l'application en visitant l'URL (localhost :8000) dans notre navigateur. Si nécessaire, nous pourrions déployer l'application sur un serveur de production.

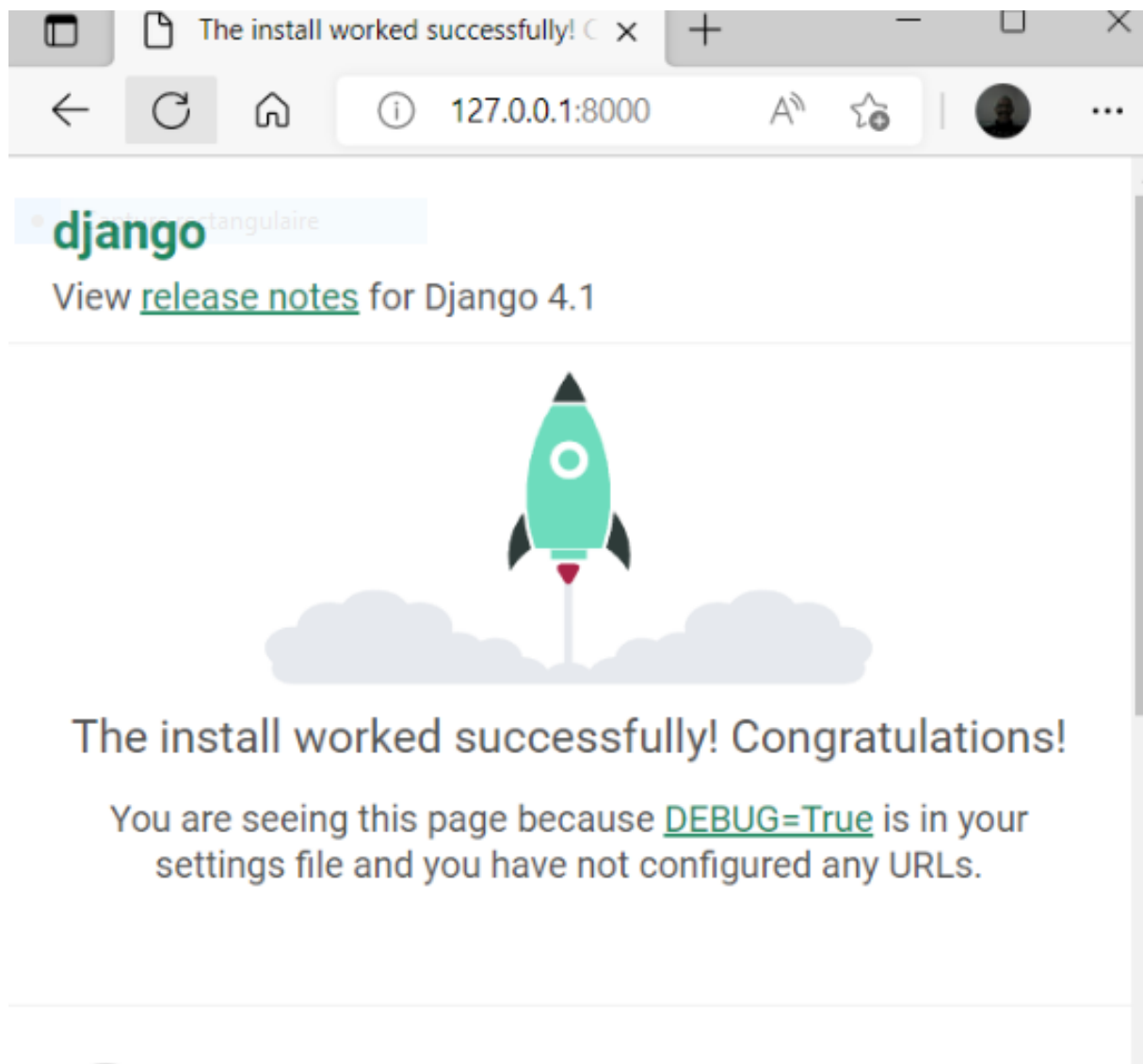


Figure 4.12: Démarrage de serveur.

- Ensuite, nous avons créé une nouvelle application Django en exécutant la commande "python manage.py (dashboard)" et nous l'avons ajoutée à la liste des applications installées dans le fichier settings.py .
- Nous avons ensuite défini les modèles de données dans le fichier models.py de l'application, exécuté les migrations pour créer les tables de base de données en utilisant la commande "python manage.py makemigrations" suivie de "python manage.py migrate", comme la figure montre :

```
C:\Users\HP-PC\Desktop\Projects>heroku run python manage.py makemigrations
Running python manage.py makemigrations on edutipster001... up, run.7706 (Free)
No changes detected

C:\Users\HP-PC\Desktop\Projects>heroku run python manage.py migrate
Running python manage.py migrate on edutipster001... up, run.1389 (Free)
Operations to perform:
  Apply all migrations: admin, auth, contenttypes, sessions, tipster
Running migrations:
  Applying contenttypes.0001_initial... OK
  Applying auth.0001_initial... OK
  Applying admin.0001_initial... OK
  Applying admin.0002_logentry_remove_auto_add... OK
  Applying contenttypes.0002_remove_content_type_name... OK
  Applying auth.0002_alter_permission_name_max_length... OK
  Applying auth.0003_alter_user_email_max_length... OK
  Applying auth.0004_alter_user_username_opts... OK
  Applying auth.0005_alter_user_last_login_null... OK
  Applying auth.0006_require_contenttypes_0002... OK
  Applying auth.0007_alter_validators_add_error_messages... OK
  Applying auth.0008_alter_user_username_max_length... OK
  Applying sessions.0001_initial... OK
  Applying tipster.0001_initial... OK

C:\Users\HP-PC\Desktop\Projects>
```

Figure 4.13: Migration.

. Nous avons également défini les vues dans le fichier `views.py` de l'application et les URL dans le fichier `urls.py` de l'application. Pour afficher les données, nous avons créé des templates HTML.

4.6 Fonctionnalités du cartable numérique

Dans cette section dédiée aux fonctionnalités du cartable numérique, nous explorerons les nombreux outils et services qui facilitent l'organisation, la collaboration et l'apprentissage des utilisateurs de notre application.

4.6.1 Page de Login / Register.

Sur la page de création de compte(Register), l'utilisateur trouve les informations nécessaires pour ouvrir un compte. En saisissant tout les informations et en respectant le formulaire requis, comme la figure suivante le montre :

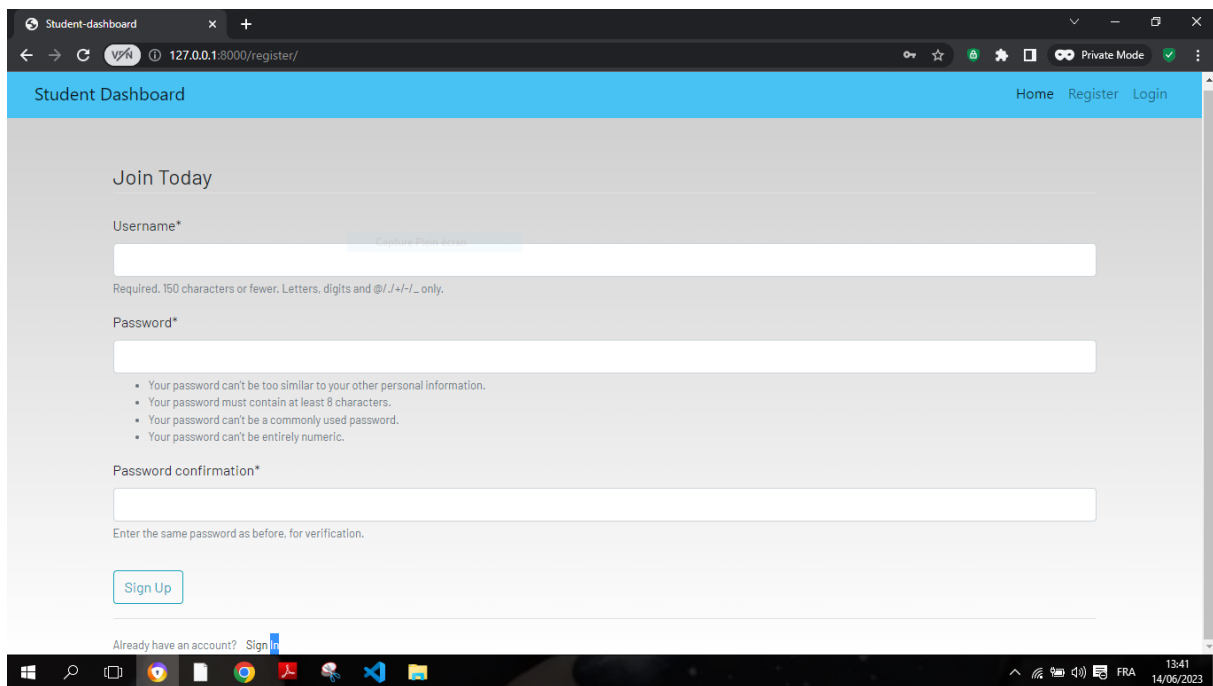


Figure 4.14: Page de Regidter.

Cette page explique comment l'utilisateur se connecte à l'application ,avec un message d'erreur apparaissant en cas d'erreur dans les informations(nom d'utilisateur, mot de passe) :

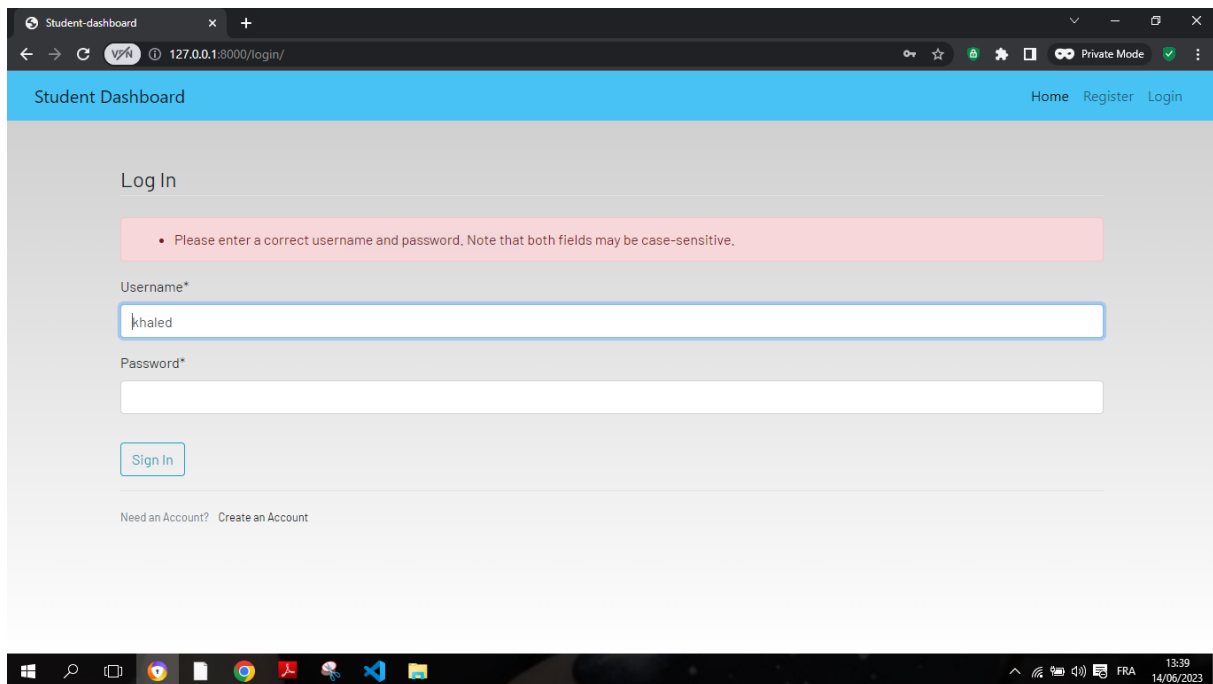


Figure 4.15: Page de Login.

4.6.2 Créer des notes

La caractéristique de la fonction note est qu'elle vous donne la possibilité d'écrire le titre de la note avec sa description. Une fois que vous appuyez sur le bouton (créer), elle est transféré aux notes spécifiques en haut de la page que l'image montre :

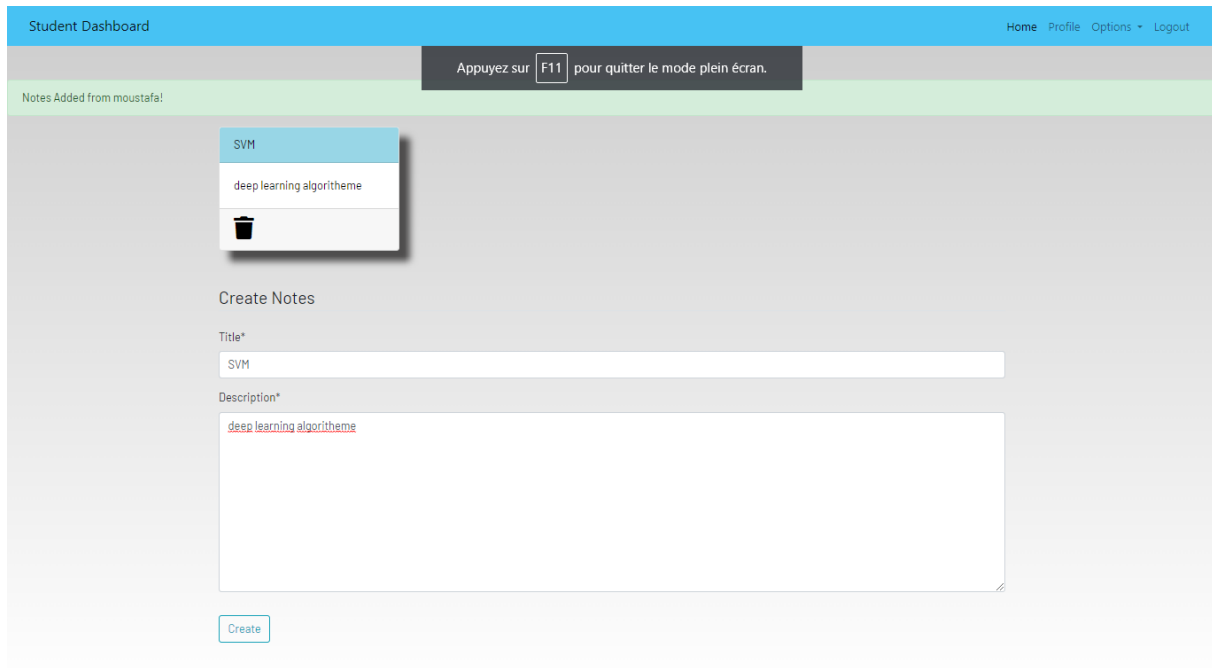


Figure 4.16: Création des notes.

Ici, nous voyons comment la liste des notes apparaît. Aussi n'oubliez pas que le nom d'utilisateur apparaît également dans une barre spéciale :

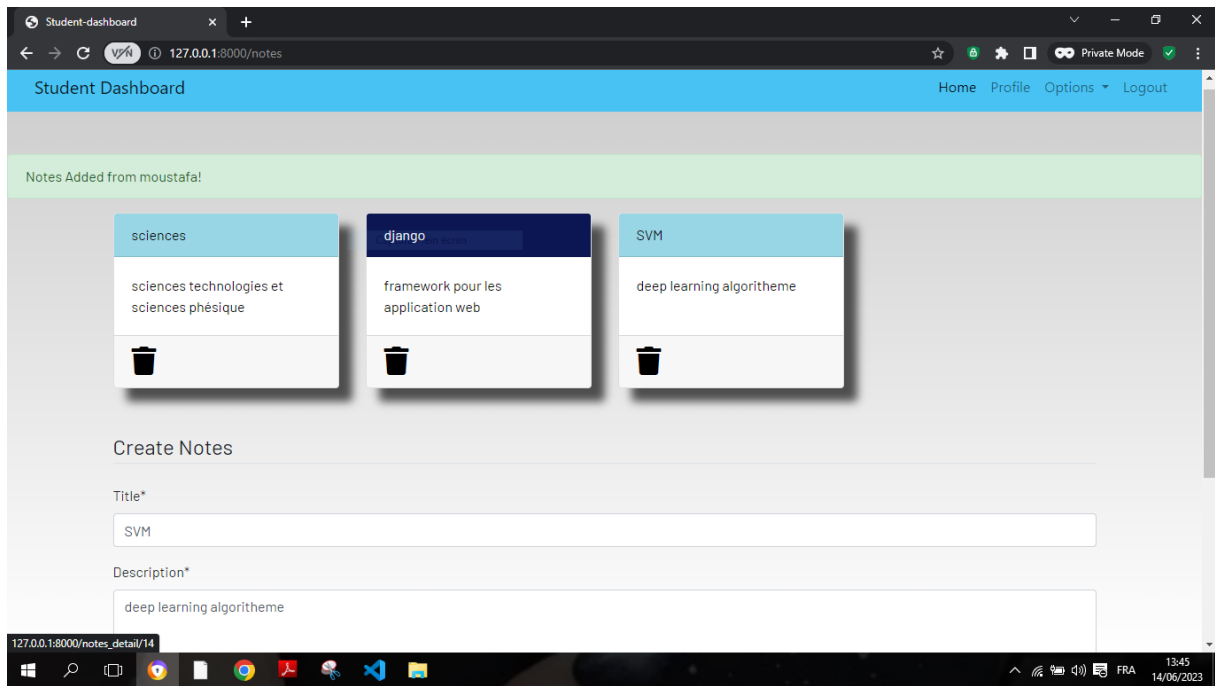


Figure 4.17: Listes des notes

4.6.3 Faire des Homeworks.

La fonction de homework contient des renseignements plus précis, qui sont le sujet, le titre et la description, ainsi que la date, et le bouton qui déterminent si la tâche est terminée ou non. Vous pouvez le voir sur cette page :

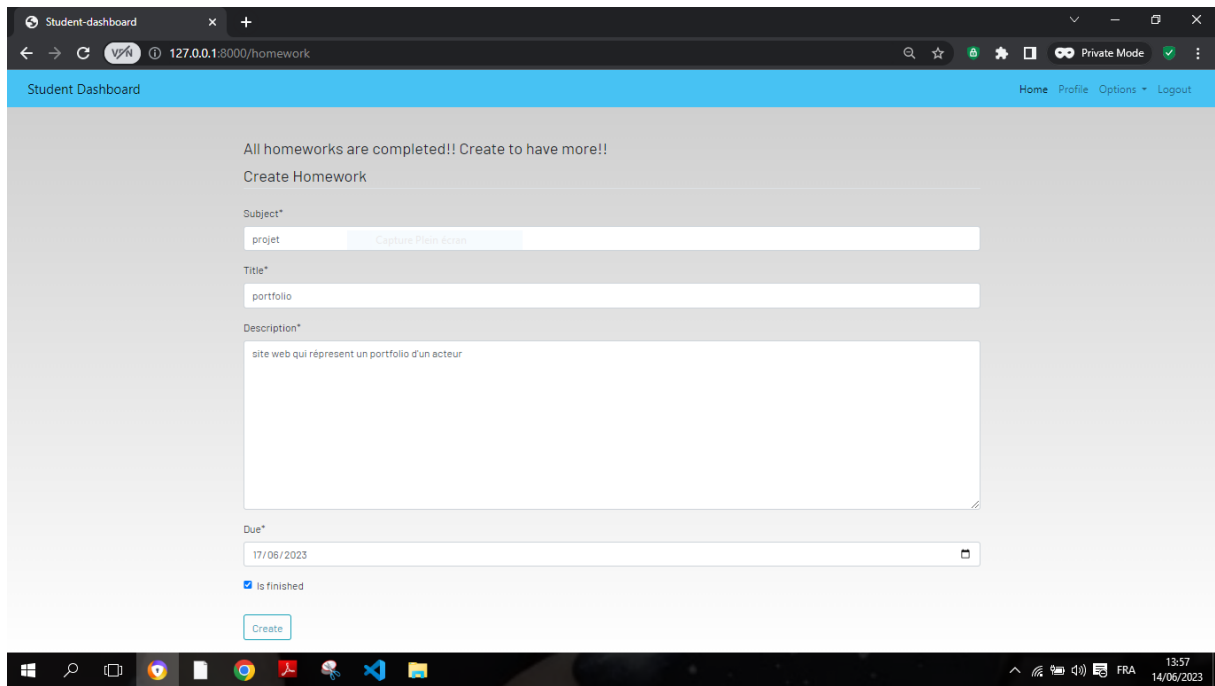


Figure 4.18: Faire des homeworks.

À la figure suivante, nous voyons comment chaque homework est ajoutée à la liste des homeworks :

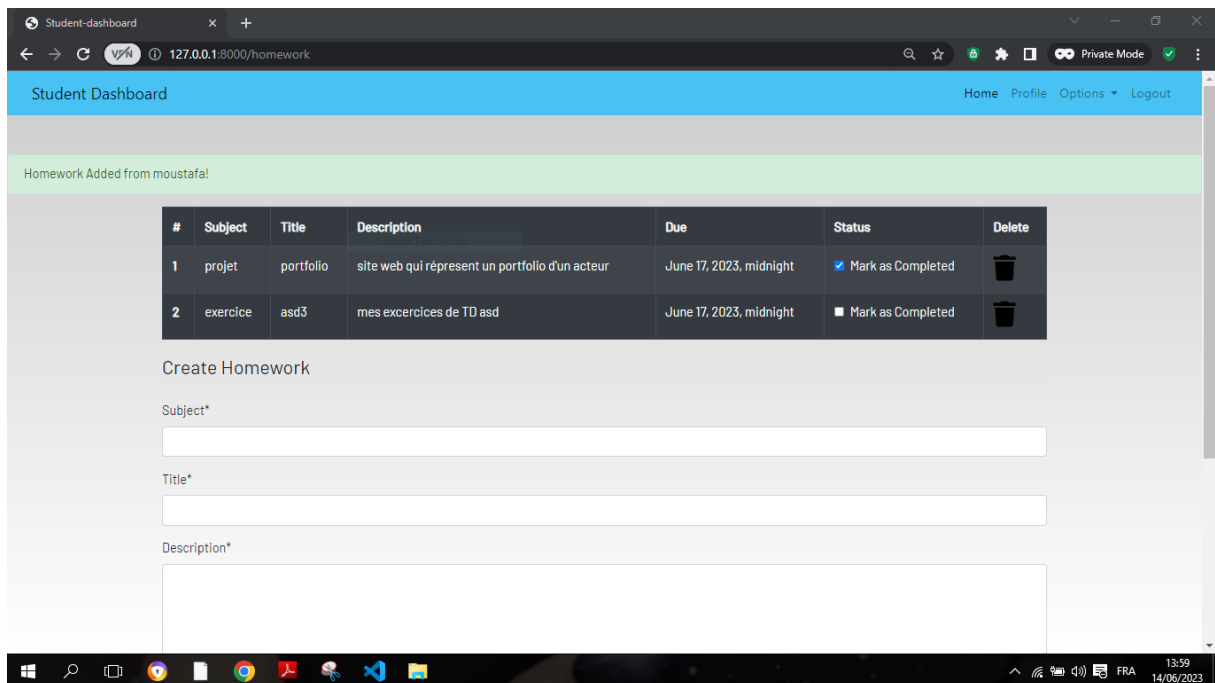


Figure 4.19: Listes des homeworks.

4.6.4 Page de profil qui affiche des TO-DOs et des homeworks incomplets

Sur cette page de profil , toutes les devoirs(homeworks) et les travaux(todos) apparaissent,avec la possibilité de supprimer et d'ajouter et de déterminer si elle est complète ou non :

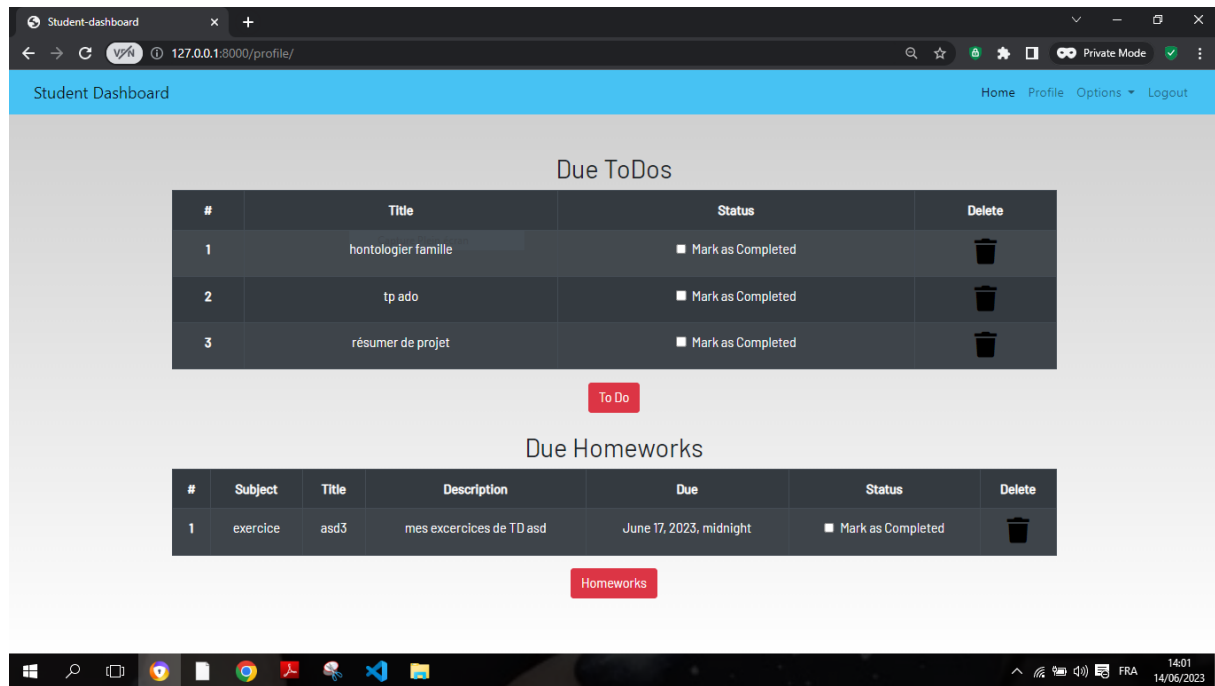


Figure 4.20: Page de profil qui affiche des TO-DOs et des homeworks incomplets.

4.6.5 Utiliser l'API dictionnaire

L'API dictionnaire permet d'accéder aux données et aux sens des mots avec des exemples et une prononciation audio.Comme illustré dans l'image ci-dessous :

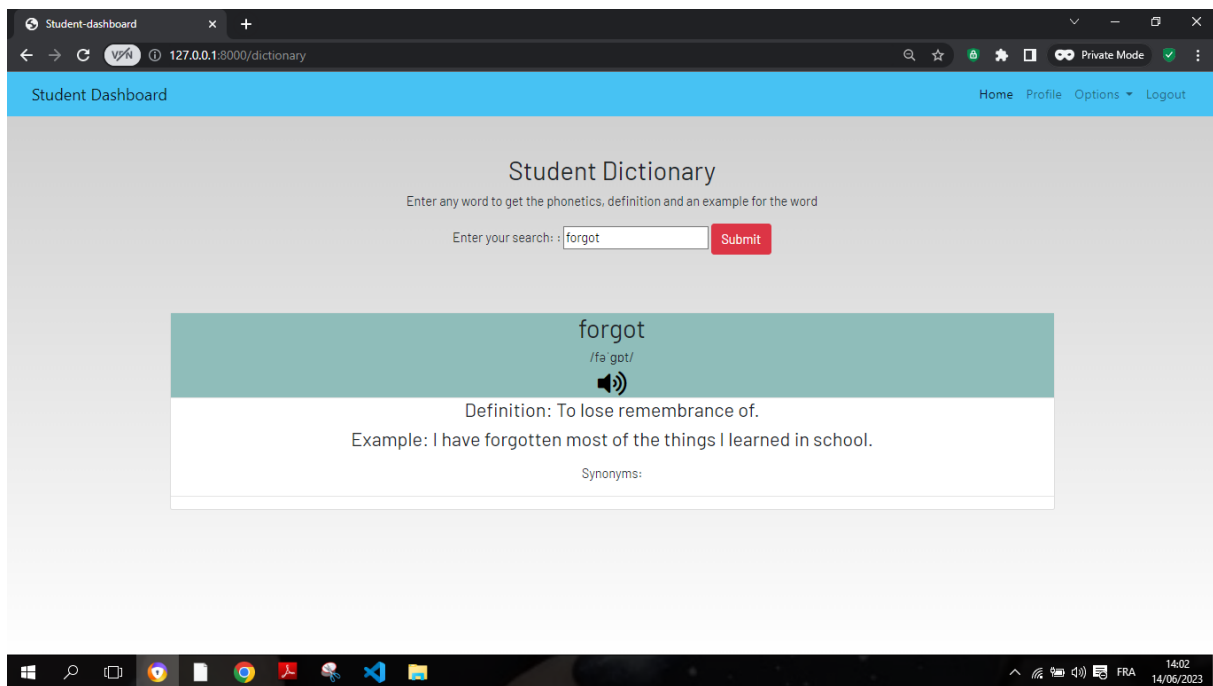


Figure 4.21: Utiliser l'API dictionnaire.

4.6.6 Rendre les résultats de recherche Youtube en utilisant l'API

Sur la page de recherche, nous pouvons voir un champ pour saisir notre recherche. Nous pouvons remplir le formulaire avec le titre, la description, la durée et la photo de la vidéo YouTube pour obtenir des résultats pertinents. Assurez-vous de rédiger une phrase claire et sans fautes d'orthographe pour obtenir les meilleurs résultats possibles. Comme nous pouvons voir ici :

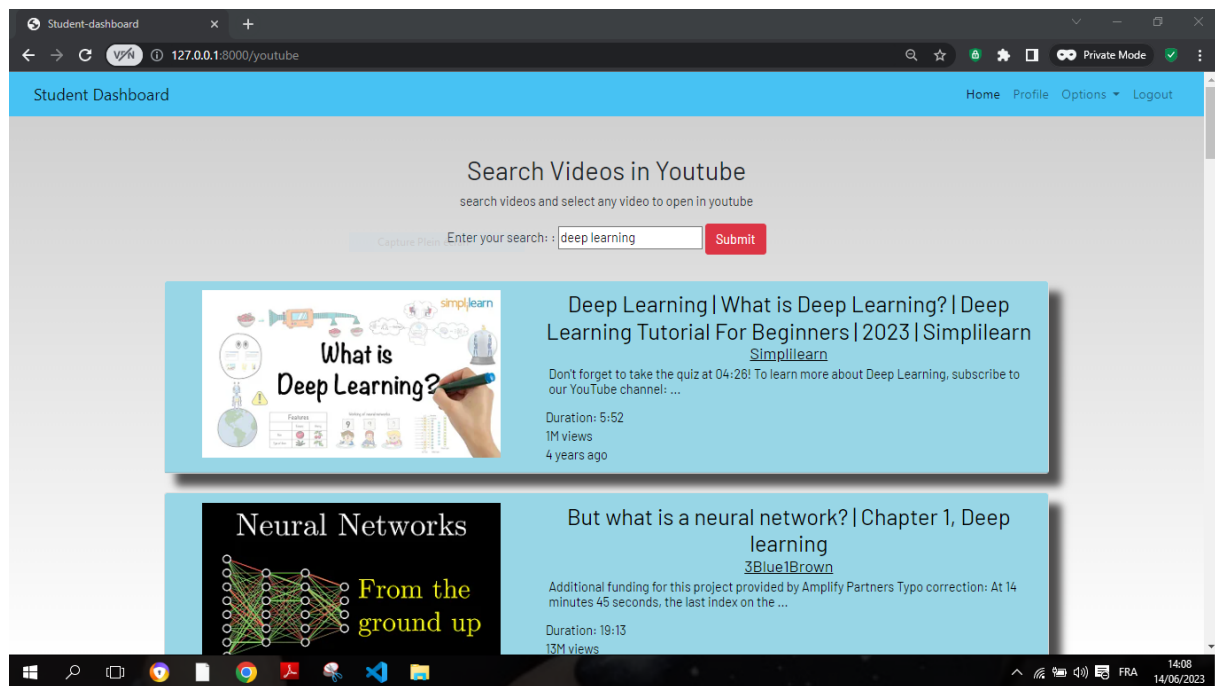


Figure 4.22: Rendre les résultats de recherche Youtube en utilisant l'API.

4.6.7 Rechercher des livres via Google API

utiliser l'API pour récupérer des informations sur les livres, telles que les titres, les auteurs, les éditeurs, les descriptions et les images de couverture. Ils peuvent également utiliser l'API pour effectuer des recherches de livres en fonction de critères tels que le titre, l'auteur ou le sujet :

4.6.9 Opération commune de conversion des mesures

La page de mesure vous permet de choisir le type de mesure à mesurer. Comme indiqué :

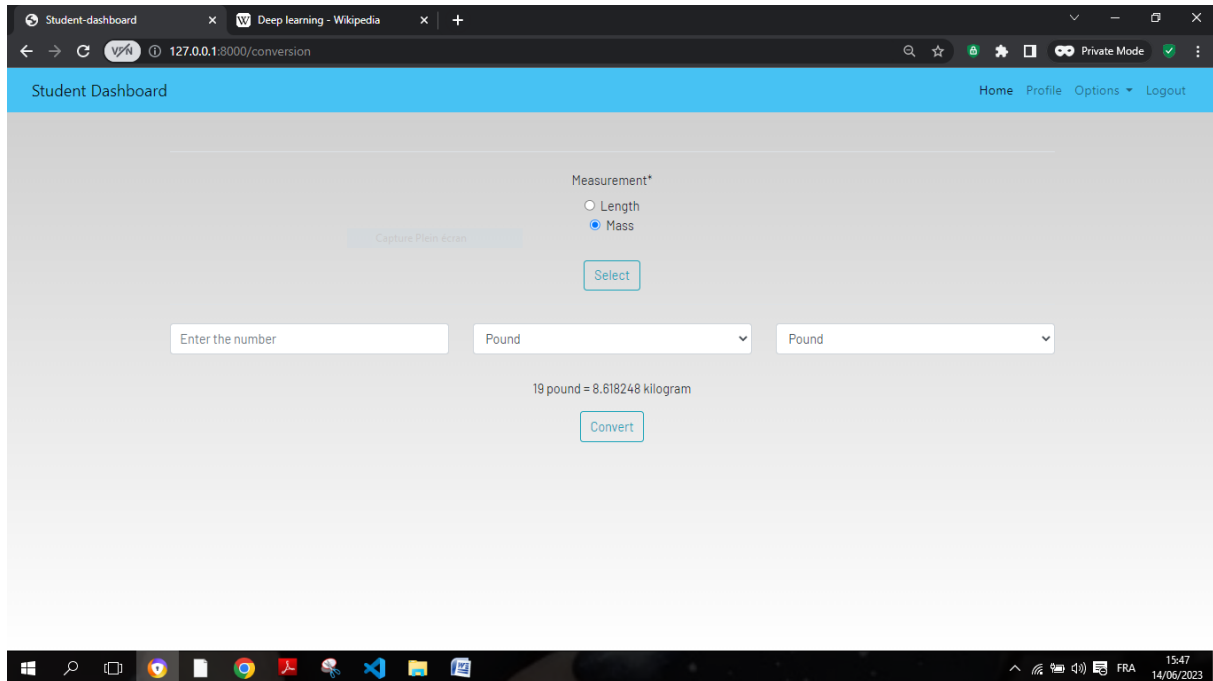


Figure 4.25: Opération commune de conversion des mesures.

4.7 Conclusion

En fin de compte, cette section a permis de présenter les différents outils utilisés pour la création de notre cartable numérique. Nous avons fourni des instructions détaillées sur la façon de configurer un environnement de développement Django. De plus, nous avons présenté les diverses bibliothèques et API que nous avons utilisées pour la création de formulaires et la récupération de données. Pour terminer ce chapitre, nous avons présenté les principales interfaces graphiques de notre cartable numérique pour bien illustrer le travail qui a été réalisé. Nous espérons que cette présentation vous a aidé à mieux comprendre les défis du développement d'applications web et vous a donné les clés pour mener à bien vos propres projets.

Conclusion Générale

En conclusion, notre projet de création d'un cartable numérique constitue une solution pratique et centralisée pour aider les étudiants à organiser efficacement leur travail académique et améliorer leur productivité. En regroupant toutes les fonctionnalités nécessaires sur une seule plateforme conviviale, nous avons offert aux étudiants un accès simplifié à la prise de notes, à la gestion des devoirs, à la recherche d'informations pertinentes sur Wikipédia, à la consultation de dictionnaires et même à la découverte de vidéos éducatives sur YouTube.

Au cours de notre recherche, nous avons abordé les concepts fondamentaux liés au Web dans le premier chapitre, ce qui nous a permis de comprendre les bases nécessaires à la création d'une application web solide. Dans le deuxième chapitre, nous avons présenté la conception de notre application en utilisant le langage de modélisation UML, assurant ainsi une planification et une structure efficaces. Enfin, dans le troisième chapitre, nous avons présenté l'application elle-même, en mettant en évidence les outils de programmation utilisés et en offrant un aperçu des interfaces graphiques développées.

Pour les perspectives d'avenir, notre application peut être améliorée et étendue de plusieurs manières. Voici quelques suggestions :

Intégration de fonctionnalités de collaboration : Permettre aux étudiants de travailler ensemble sur des projets, de partager des notes et de collaborer en temps réel.

Personnalisation de l'interface : Offrir aux utilisateurs la possibilité de personnaliser l'apparence de l'application en fonction de leurs préférences, ce qui améliorera l'expérience utilisateur.

Intégration d'autres sources de recherche : Ajouter la possibilité de rechercher des articles scientifiques, des livres en ligne et d'autres ressources académiques pour faciliter les travaux de recherche des étudiants.

Suivi des progrès et des performances : Intégrer des fonctionnalités de suivi des progrès et des performances des étudiants, permettant ainsi de mesurer et de visualiser leurs réalisations académiques.

En poursuivant le développement de notre application, en tenant compte de ces pers-

pectives d'amélioration, nous pourrons continuer à répondre aux besoins changeants des étudiants en matière

Bibliographie

- [1] Scorchsoft. Web technologies overview. <https://www.scorchsoft.com/blog/web-technologies-overview/>, 2019. Accessed on May 2, 2023.
- [2] What is a web server ? https://developer.mozilla.org/fr/docs/Learn/Common_questions/Web_mechanics/What_is_a_web_server. Accessed : <Accessed on May 2, 2023>.
- [3] Softkraft. Web application architecture : How does it work? 2021.
- [4] Pixels and Bytes. Comprendre les web apps. <https://pixelsandbytes.fr/comprendre-web-app/>, n.d. Accessed : May 9, 2023.
- [5] OpenGenus IQ. Model-view-controller (mvc) in django. *OpenGenus IQ*, 2021. Accessed : May 8, 2023.
- [6] Soshace. Optimizing database interactions in python : Sqlalchemy best practices, 2021.
- [7] Oxylabs. Python web scraping, <year>. 2023-06-14.
- [8] CERN. A short history of the web. <https://home.cern/fr/science/computing/birth-web/short-history-web>, n.d. Accessed on April 30, 2023.
- [9] GeeksforGeeks. Web technology. <https://www.geeksforgeeks.org/web-technology/>, n.d. Accessed : April 30, 2023.
- [10] TMS Outsource. Web technologies : A comprehensive guide. <https://tms-outsource.com/blog/posts/web-technologies/>, 2019. Accessed : April 30, 2023.
- [11] Mozilla Corporation. Mdn web docs. <https://developer.mozilla.org/en-US/docs/Web>, n.d. Accessed : April 30, 2023.
- [12] Fabien Gandon. Technologies for the world wide web. https://www.cs.cmu.edu/~fgandon/lecture/mass1_internet2000/techno_web/, 2000. Accessed : April 30, 2023.
- [13] Gartner. Website. <https://www.gartner.com/en/information-technology/glossary/website>, n.d. Accessed : April 30, 2023.

- [14] Mémoire vive (ram). <https://www.techno-science.net/definition/3811.html>. [Online ; accessed 1-May-2023].
- [15] Application web. <https://www.techno-science.net/glossaire-definition/Application-Web.html>. [Online ; accessed 1-May-2023].
- [16] UpGrad. Web application architecture : What is it and how does it work?, 2022.
- [17] PappleWeb. À quoi sert une base de données?, 2021.
- [18] Le-Prof.com. Cartable électronique.
- [19] Unified Process. <https://fr.theastrologypage.com/unified-process>. Accessed on May 16, 2023.
- [20] Reactive Executive. Processus unifié ou méthode agile. <https://www.reactive-executive.com/processus-unifie-ou-methode-agile/>, 2023. Accessed : juin 8, 2023.
- [21] IONOS by 11. UML - A Modeling Language for Object-Oriented Programming, n.d.
- [22] Craig Morrison. What is django? the python web framework for perfectionists, 2021.
- [23] DataFlair. Django architecture - learn how django works internally. <https://data-flair.training/blogs/django-architecture/>, Accessed 2023.
- [24] Doprax. Django tutorial for beginners part 6, n.d.
- [25] Django Software Foundation. Django, n.d.
- [26] Django Software Foundation. Http request and response objects, 2023.
- [27] Python Software Foundation. Python. <https://www.python.org/>, N/A. Accessed on : June 13, 2023.
- [28] SQLite. About SQLite. <https://www.sqlite.org/about.html>, N/A. Accessed on : Date.
- [29] IONOS. Web scraping : extraction de données sur le web. <https://www.ionos.fr/digitalguide/sites-internet/developpement-web/web-scraping-extraction-de-donnees-sur-le-web/>, N/A. Accessed on : June 13, 2023.
- [30] PyPI. YouTube-search Python Library. <https://pypi.org/project/youtube-search-python/>, N/A. Accessed on : june 13,2023.
- [31] javatpoint. Wikipedia Module in Python. https://developer.mozilla.org/fr/docs/Learn/Server-side/Django/development_environment#utiliser_django_dans_un_environnement_virtuel_python. Accessed on : June 13,2023.
- [32] Stack Overflow Community. Stack overflow - how to use to google dictionary as an api, Accessed 2023.

- [33] Merixstudio. Django crispy forms - what are they about? <https://www.merixstudio.com/blog/django-crispy-forms-what-are-they-about/>, N/A. Accessed on : June 13, 2023.
- [34] Andrew Parker. Html5 : What is it? *TechRadar*, 2012.
- [35] Mozilla Developer Network. Css, n.d.
- [36] Mozilla Developer Network. Javascript, n.d.
- [37] Microsoft. Visual studio, n.d.