



**ECOLE MAROCAINE DES
SCIENCES DE L'INGENIEUR**
Membre de
HONORIS UNITED UNIVERSITIES

HONORIS
UNITED UNIVERSITIES

ÉCOLE MAROCAINE DES SCIENCES DE L'INGÉNIEUR - RABAT

PROJET DE FIN D'ANNÉE
3EME ANNÉE
INGÉNIERIE INFORMATIQUE ET RÉSEAUX

Application de suivi des dépenses

Réalisé par :

Anas AIT EL QADI
Mohamed Taha IZMAR

Encadré par :

Mme.Asma OUARHIM

Dédicace

À tous ceux qui ont contribué à la concrétisation de ce projet, nous avons ensemble traversé des épreuves et trouvé des inspirations, partagé des rires et des moments de réflexion.

Ce projet de fin d'année est bien plus qu'une simple réalisation ; il est le reflet de notre dévouement, de notre créativité et de notre esprit de collaboration. Chaque membre de notre groupe a apporté une touche unique, illuminant ce projet de manière spéciale et précieuse.

Que cette dédicace témoigne de notre travail acharné, de notre solidarité et de notre passion commune. Puissions-nous nous souvenir de ce projet comme d'une preuve indélébile de ce que nous sommes capables d'accomplir ensemble.

Remerciements

Nous souhaitons exprimer notre profonde gratitude à tous ceux qui ont contribué à la réalisation de ce projet de fin d'études. Tout d'abord, nous remercions sincèrement Monsieur le Directeur de l'EMSI pour la formation de qualité qu'il nous a offerte. Grâce à son leadership et à sa vision, nous avons bénéficié d'un environnement académique exceptionnel, propice à notre épanouissement et à notre réussite. Nos sincères remerciements vont également à nos professeurs, dont l'expertise, la patience et l'engagement ont été essentiels à notre apprentissage. Leurs enseignements et leurs conseils nous ont guidés tout au long de notre parcours académique, nous permettant de développer les compétences nécessaires pour relever les défis de demain. À nos amis, merci pour votre soutien constant, vos encouragements et pour les moments de camaraderie qui ont rendu cette expérience encore plus mémorable. Votre présence à nos côtés a été une source de motivation et de réconfort. Enfin, nous souhaitons exprimer notre infinie reconnaissance à nos parents. Votre amour, votre soutien inconditionnel et vos sacrifices ont été le pilier sur lequel nous avons construit nos rêves. Sans vous, rien de tout cela n'aurait été possible. Ce projet de fin d'études est le fruit de la collaboration, du travail acharné et du soutien de chacun d'entre vous. Merci de nous avoir permis de réaliser cet accomplissement.

Résumé

Ce rapport discute du processus de développement ainsi que des principales fonctionnalités associées à une application web de gestion des transactions. L'objectif fondamental de l'application en question est de fournir un outil facile d'utilisation à de différentes personnes tout en facilitant la surveillance de leurs transactions quotidiennes, mensuelles et annuelles.

L'implémentation de cette application s'est effectuée en quatre étapes : la planification, la construction, le test et le déploiement. Planification : création d'une interface utilisateur guidant leurs interactions avec l'application. Elle se compose de la saisie des transactions, du classement automatique et de la génération des rapports détaillés.

Pour réaliser l'application, des technologies web modernes, y compris HTML, CSS, JavaScript pour le frontend et Python pour le backend, ont été utilisées. Outre, des Framework tels que Django et Bootstrap ont été utilisés pour garantir une expérience utilisateur transparente et réactive. Des tests ont été effectués à chaque étape du cycle de vie de développement pour assurer la qualité et la fiabilité de l'application. Il s'agissait de tests unitaires, d'intégration et de régression pour identifier les erreurs ou les problèmes de performance.

En fin de compte, cette application de gestion des transactions propose une solution viable et facile à utiliser aux fins de périmètre défini. DesktopBudget étant une application conviviale avec des fonctionnalités utiles, il est sûr de faciliter la prise en charge de votre périmètre de tâches de gestion des impôts ce qui est crucial.

Abstract

This report discusses the development process as well as the main functionalities associated with a transaction management web application. The basic objective of the application in question is to provide an easy-to-use tool to different people while facilitating the monitoring of their daily, monthly and annual transactions.

The implementation of this application was carried out in four stages : planning, building, testing and deployment. Planning : creating a user interface to guide their interactions with the application. It consists of transaction entry, automatic classification and generation of detailed reports.

To make the application, modern web technologies, including HTML, CSS, JavaScript for the frontend and Python for the backend, were used. Additionally, frameworks such as Django and Bootstrap were used to ensure a seamless and responsive user experience. Testing was carried out at each stage of the development lifecycle to ensure the quality and reliability of the application. This involved unit, integration and regression testing to identify errors or performance issues.

Ultimately, this transaction management application offers a viable and easy-to-use solution for scope-defined purposes. DesktopBudget being a user-friendly application with useful features, it is sure to make it easier to take care of your scope of person management tasks which is crucial.

Table des matières

Dédicace	2
Remerciements	3
Résumé	4
Abstract	5
Liste des sigles et acronymes	10
Introduction générale	11
1 Présentation du projet et étude du marché	12
1.1 Introduction	13
1.2 Critique de l'existant et solution proposée	13
1.2.1 Critique de l'existant	13
1.2.2 solution proposée	15
1.3 Spécification de besoin	15
1.3.1 Besoin fonctionnel	15
1.3.2 Besoin non fonctionnel	15
1.4 Conclusion	16
2 Etude théorique	17
2.1 Introduction	18
2.2 Langage de conception	19
2.2.1 Langage UML	19
2.2.2 Les différents types de diagrammes UML	19
2.3 Conception	20
2.3.1 Diagrammes utilisés	20
2.4 Conclusion	26
3 Etude pratique	27
3.1 Introduction	28
3.2 Outils	28
3.2.1 Django	28
3.2.2 Python	30
3.2.3 Html	31
3.2.4 css	33
3.2.5 js	35

3.2.6 staruml	36
3.2.7 postman	38
3.3 Réalisation du projet	40
3.4 Conclusion	51
Conclusion générale	52
Perspectives	53
Webographie	54

Table des figures

1.1	Tradogram	13
1.2	Tipalti	14
2.1	diagramme de cas d'utilisation	20
2.2	diagramme de classe	21
2.3	diagramme de séquence 1	22
2.4	diagramme de séquence 2	23
2.5	diagramme de séquence 3	24
2.6	diagramme de séquence 4	25
3.1	django logo	28
3.2	le fonctionnement de Django (mvt)	29
3.3	python logo	30
3.4	Html logo	31
3.5	Structure de base d'un document HTML	32
3.6	css logo	33
3.7	Structure de CSS	34
3.8	js logo	35
3.9	staruml logo	36
3.10	postman logo	38
3.11	application 1	40
3.12	application 2	41
3.13	application 3	41
3.14	application 4	42
3.15	log in	43
3.16	registre	44
3.17	dashboard	45
3.18	expenses	46
3.19	add expenses	46
3.20	edit expenses	47
3.21	incomes	47
3.22	expenses summary	48
3.23	incomes summary	48
3.24	devise	49
3.25	profile	50

Liste des tableaux

1	Tableau des sigles et acronymes	10
---	---	----

Liste des sigles et acronymes

Sigle	Description
HTML	HyperText Markup Language
CSS	Cascading Style Sheets
JS	JavaScript
VS	Visual Studio Code
UML	Unified Modeling Language

TABLE 1 – Tableau des sigles et acronymes

Introduction générale

L'importance croissante de la transparence et de la conformité, dans le monde financier, a entraîné une forte demande pour une gestion dynamique et le contrôle des transactions. Les outils de gestion des transactions sur site web sont conçus pour offrir aux clients un espace sûr et accueillant pour gérer chacune de leurs transactions quotidiennes, mensuelles et annuelles. Dans un monde de plus en plus numérique, il est indispensable de proposer une solution sûre, fiable et intuitive à tout un chacun pour la gestion avancée de ses finances personnelles.

Dans un souci d'innovation, notre projet se focalise sur la création d'une application web de gestion des transactions. Le but de ce bilan est de fournir une analyse détaillée du cycle de développement de l'application et de mettre en évidence les choix de technologies, les processus de design et les caractéristiques fonctionnelles de ceux que nous avons pu proposer pour obtenir une expérience utilisateur idéale.

Nous avons pris les mesures nécessaires pour garantir la sécurité des données financières de l'utilisateur tout en simplifiant et en rationalisant le suivi et la vérification des opérations. La solution que nous proposons repose sur une combinaison d'optimisation de l'interface utilisateur, de catégorisation intelligente de vos transactions et de génération de rapports détaillés pour vous aider à comprendre quelles sont vos habitudes de consommation et comment en prendre de meilleures.

Au cours de ce projet, nous avons utilisé notre expérience dans le développement web, le design d'interface utilisateur et la gestion de projet pour relever les défis liés à la création d'une solution de gestion de transactions globale et efficace. Ce dossier fournit une vue d'ensemble de notre méthode de conception — de l'exploration des besoins initiaux à l'intégration des fonctionnalités essentielles — en mettant en lumière les enseignements tirés et les succès obtenus tout au long de notre entreprise.

Chapitre 1

Présentation du projet et étude du marché

1.1 Introduction

Les avancées technologiques ont bouleversé la gestion financière des individus et des organisations dans leur vie de tous les jours. Dans ce cadre, notre objectif est de concevoir une application web novatrice qui se concentre sur la gestion des dépenses et des ressources. Cette application fournira aux utilisateurs une plateforme centralisée et conviviale pour gérer leurs flux financiers de manière efficace en les suivant et en les analysant.

1.2 Critique de l'existant et solution proposée

1.2.1 Critique de l'existant

À l'heure actuelle, il existe plusieurs applications de gestion des dépenses et des ressources telles que 'Tradogram' et 'Tipalti' qui sont disponibles sur le marché. Toutefois, il est fréquent que ces solutions aient des limitations majeures. Par exemple, bien que 'Tradogram' propose une gestion des achats et des dépenses, son intégration avec d'autres systèmes financiers peut être difficile. Par ailleurs, 'Tipalti' se focalise sur l'automatisation des paiements, bien qu'il puisse manquer de fonctionnalités avancées pour une gestion exhaustive des dépenses.

Les utilisateurs rencontrent souvent des difficultés avec ces applications, notamment une mauvaise gestion des flux de trésorerie, une interface utilisateur peu conviviale et des capacités d'analyse limitées. La précision et la clarté des rapports financiers peuvent également être compromise par la synchronisation de données provenant de plusieurs sources.

Tradogram :

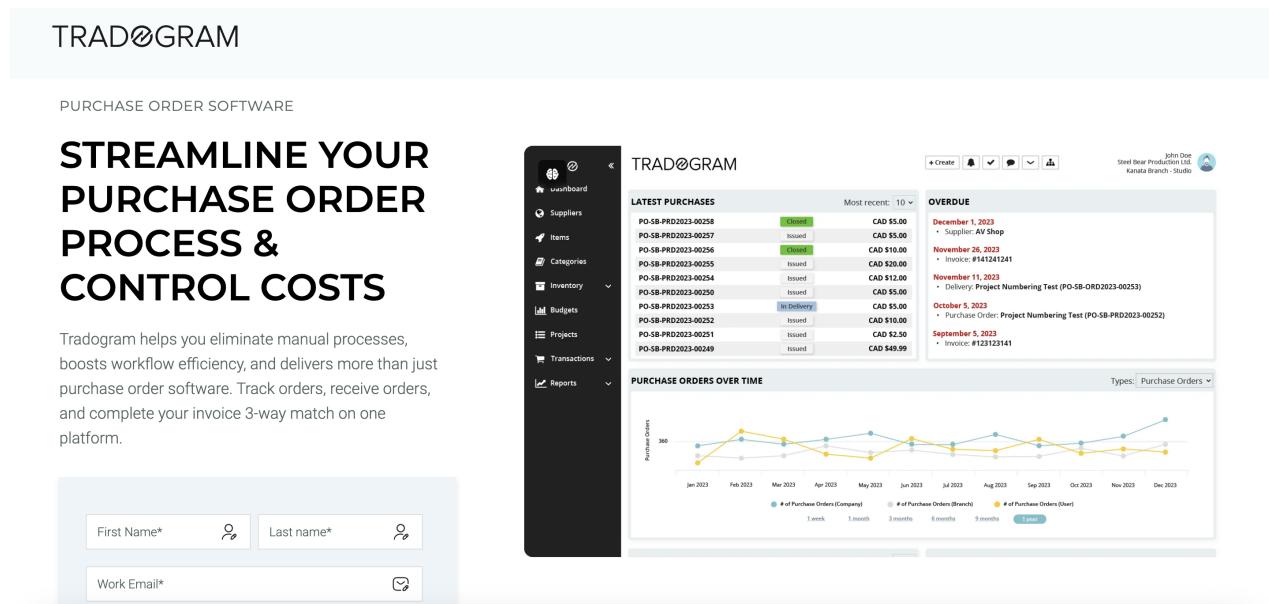


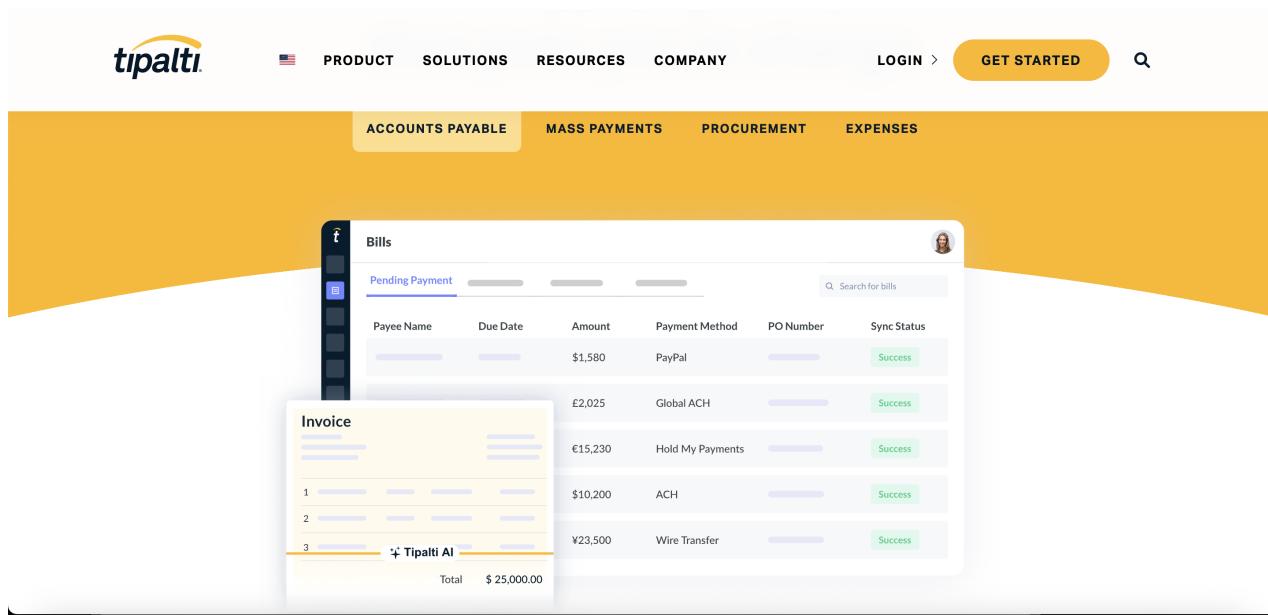
FIGURE 1.1 – Tradogram

Description : Les entreprises peuvent utiliser la plateforme de gestion des achats et des dépenses, Tradogram. Elle propose des fonctionnalités comme la gestion des validations, la création de commandes d'achat et le suivi des fournisseurs.

Points forts : L'introduction à des systèmes ERP, l'adaptation des flux d'approbation et la gestion centralisée des dépenses sont inclus.

Limitations : L'intégration avec d'autres systèmes financiers peut être complexe, et l'interface utilisateur peut parfois ne pas être très intuitive pour les utilisateurs non techniques.

Tipalti :



Payee Name	Due Date	Amount	Payment Method	PO Number	Sync Status
		\$1,580	PayPal		Success
		€2,025	Global ACH		Success
		€15,230	Hold My Payments		Success
		\$10,200	ACH		Success
		¥23,500	Wire Transfer		Success
Total		\$ 25,000.00			

FIGURE 1.2 – Tipalti

Description : Les services de Tipalti sont axés sur l'automatisation des paiements internationaux et le respect des réglementations fiscales pour les entreprises. Elle facilite le processus de paiement des fournisseurs à l'échelle mondiale.

Points forts : La mise en place de l'automatisation des processus de paiement, ainsi que le respect des réglementations pour les transactions internationales et la gestion fiscale.

Limitations : Il est possible que certaines fonctionnalités essentielles pour une gestion complète des dépenses soient manquantes, étant donné que l'accent principal est mis sur les paiements.

1.2.2 solution proposée

Afin de faire face à ces défis, notre application se démarquera par sa facilité d'utilisation et ses fonctionnalités solides. Notre offre inclut une intégration fluide avec différentes comptes bancaires et cartes de crédit, une classification astucieuse des transactions ainsi qu'une analyse poussée des habitudes de dépenses. En outre, des fonctionnalités de budget personnalisées ainsi que des notifications proactives seront intégrées afin d'aider les utilisateurs à réaliser leurs objectifs financiers.

1.3 Spécification de besoin

1.3.1 Besoin fonctionnel

Enregistrement et protection sécurisée des transactions financières.

Une catégorisation claire des dépenses et des revenus avec une représentation visuelle.

Création de rapports sur mesure et d'analyses financières prédictives.

Configuration des objectifs financiers et suivi de leur évolution.

1.3.2 Besoin non fonctionnel

Les données personnelles et financières bénéficient d'une sécurité accrue.

Obtention de performances optimales même lorsque le volume de données est élevé.

Compatible avec les principaux navigateurs web et accessible sur divers appareils.

1.4 Conclusion

Pour résumer, notre projet d'application web de gestion des dépenses et des ressources sera une réponse adaptée aux besoins grandissants des utilisateurs qui veulent améliorer leur gestion financière grâce à une solution moderne, sécurisée et conviviale. Ce chapitre a posé les fondements de notre approche et des fonctionnalités attendues, établissant ainsi les bases sur lesquelles nous allons développer et mettre en œuvre notre solution innovante.

Chapitre 2

Etude theorique

2.1 Introduction

La conception joue un rôle essentiel dans la réalisation de solutions innovantes. Que ce soit dans le domaine de l'ingénierie, de l'architecture, du design ou d'autres domaines créatifs, la conception est le processus clé qui permet de transformer une idée abstraite en une réalité tangible. Ce chapitre se concentre sur l'importance de la conception et explore les différentes étapes et approches nécessaires pour concevoir des solutions innovantes. En résumé, ce chapitre de conception nous permettra de plonger dans l'univers de l'application web de gestion des dépenses en commençant par l'élaboration du diagramme de cas d'utilisation et terminer par le diagramme de séquence .

2.2 Langage de conception

2.2.1 Langage UML

L'UML (Unified Modeling Language ou Langage de modélisation unifiée en français) est un langage graphique de modélisation informatique. Ce langage est désormais la référence en modélisation objet, ou programmation orientée objet. Cette dernière consiste à modéliser des éléments du monde réel (immeuble, ingrédients, personne, logos, organes du corps...) ou virtuel (temps, prix, compétence...) en un ensemble d'entités informatiques appelées « objet ». L'UML est constitué de diagrammes qui servent à visualiser et décrire la structure et le comportement des objets qui se trouvent dans un système. Il permet de présenter des systèmes logiciels complexes de manière plus simple et compréhensible qu'avec du code informatique. L'UML a des applications dans le développement logiciel, mais aussi dans l'industrie (pour modéliser les flux de processus par exemple), dans l'ingénierie ou le marketing.

2.2.2 Les différents types de diagrammes UML

L'UML définit 14 types de diagrammes divisés en deux catégories.

1. Les diagrammes de structure représentent les éléments du système, leurs propriétés et leurs relations entre eux :

Diagramme de classes
Diagramme d'objets
Diagramme de composants
Diagramme de structure composite
Diagramme d'ensemble
Diagramme de déploiement
Diagramme de profil

2. Les diagrammes de comportement représentent les processus et les interactions entre les objets

Diagramme de cas d'utilisation
Diagramme d'activité
Diagramme d'état-transition
Diagramme de séquence
Diagramme de communication
Diagramme de temps
Diagramme d'aperçu d'interaction

2.3 Conception

2.3.1 Diagrammes utilisés

Pour commencer, nous allons élaborer le diagramme de cas d'utilisation, qui représente les différentes interactions entre les utilisateurs et le système. Ce diagramme nous permettra d'identifier les fonctionnalités essentielles de l'application et de définir les actions principales que les utilisateurs pourront effectuer.

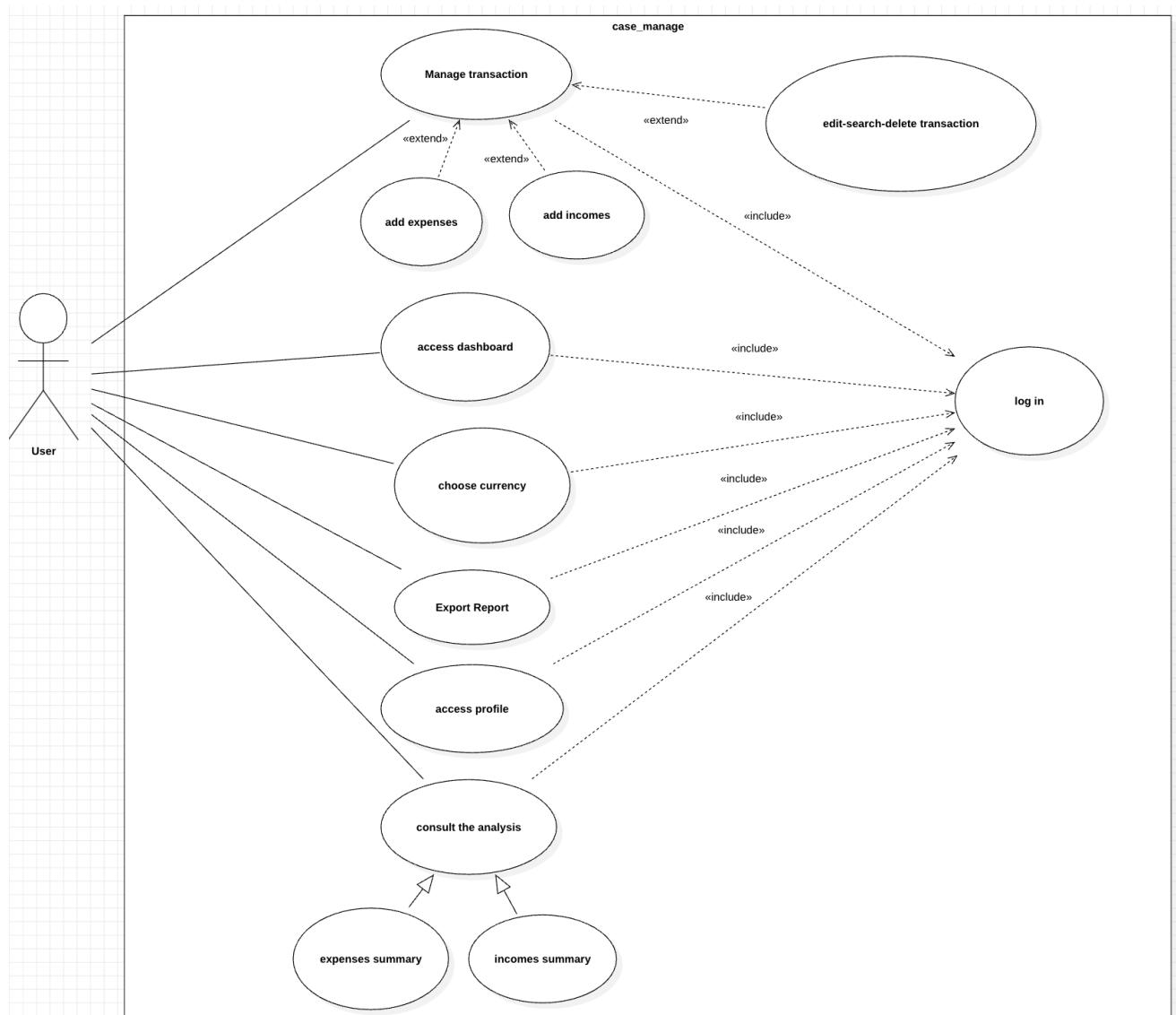


FIGURE 2.1 – diagramme de cas d'utilisation

Les principaux cas d'utilisation incluent :

Enregistrement des dépenses

Visualisation des dépenses

Consultation de l'analyse

Télécharger les rapport (excel/csv)

Après cela, nous procéderons à la création du diagramme de classes, qui illustre la structure statique du système en modélisant les classes, leurs attributs, méthodes et interrelations. Ce schéma sera utile pour que nous puissions comprendre l'interaction entre les différentes parties de l'application et concevoir une architecture cohérente et extensible.

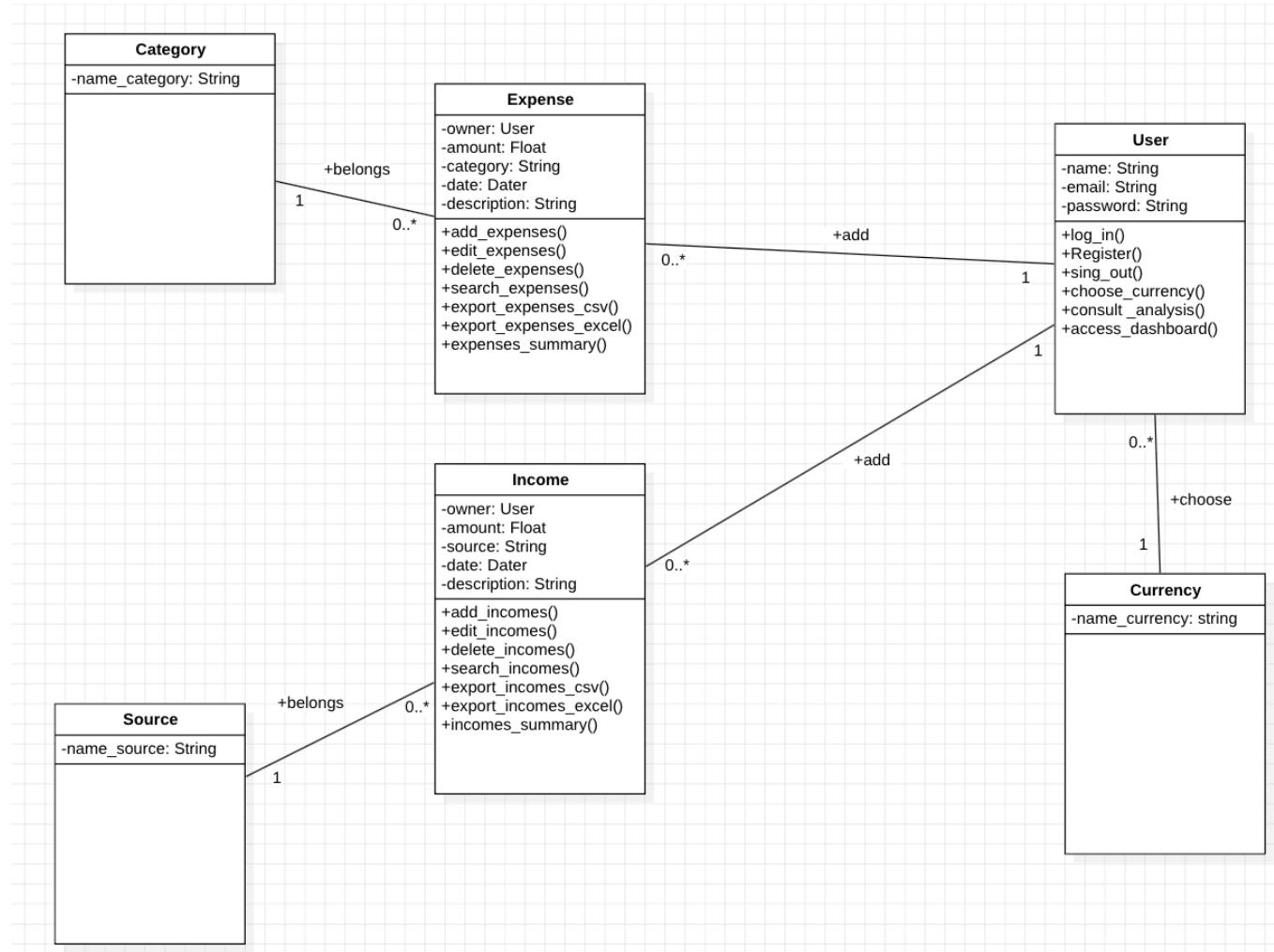


FIGURE 2.2 – diagramme de classe

Enfin, nous allons créer le diagramme de séquence pour représenter l'interaction des objets entre eux au cours du temps afin d'exécuter un cas d'utilisation précis. Ce diagramme sera utile pour voir l'ordre des messages échangés entre les objets et donner plus de détails sur l'exécution dynamique des scénarios dans le système.

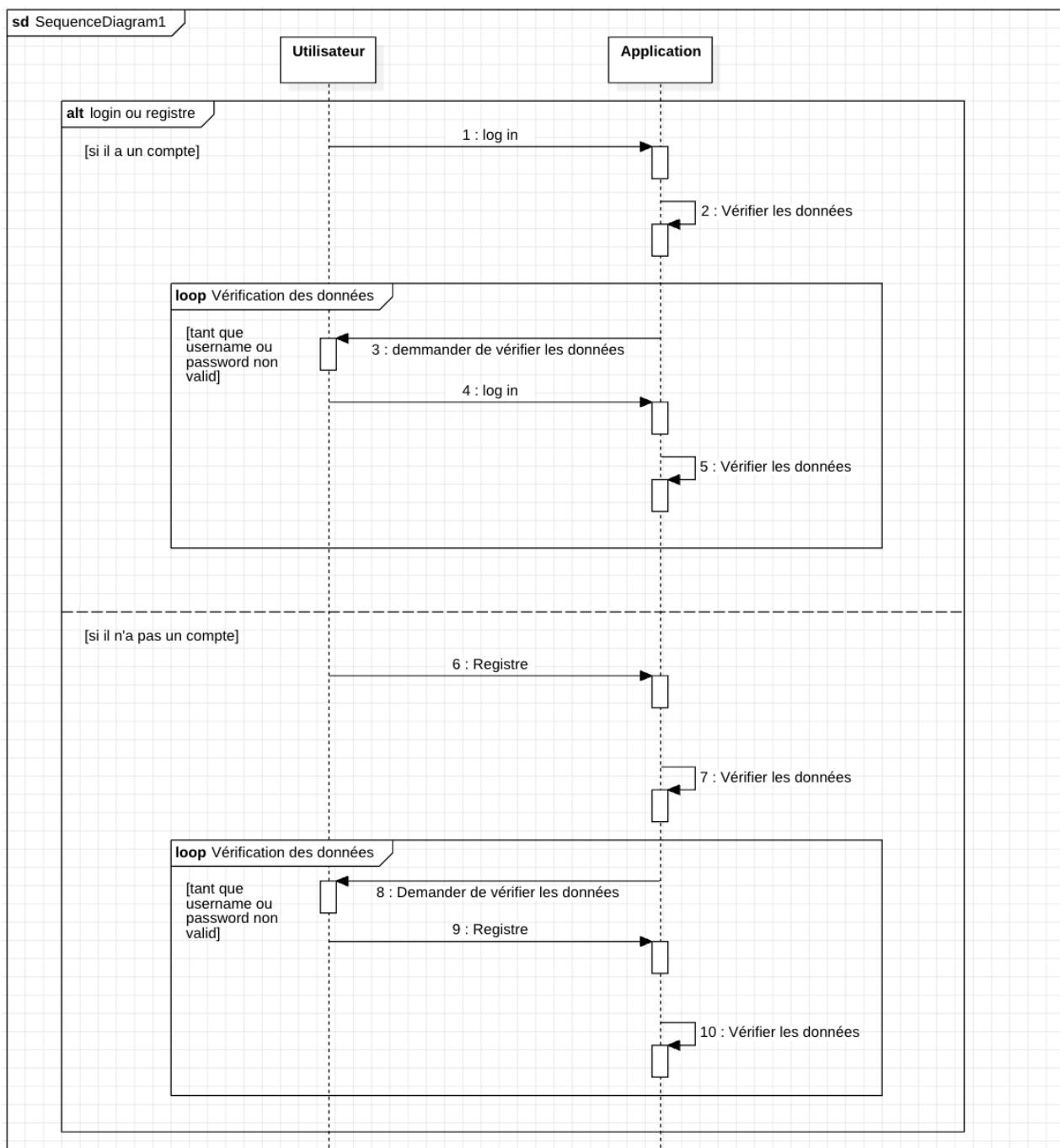


FIGURE 2.3 – diagramme de séquence 1

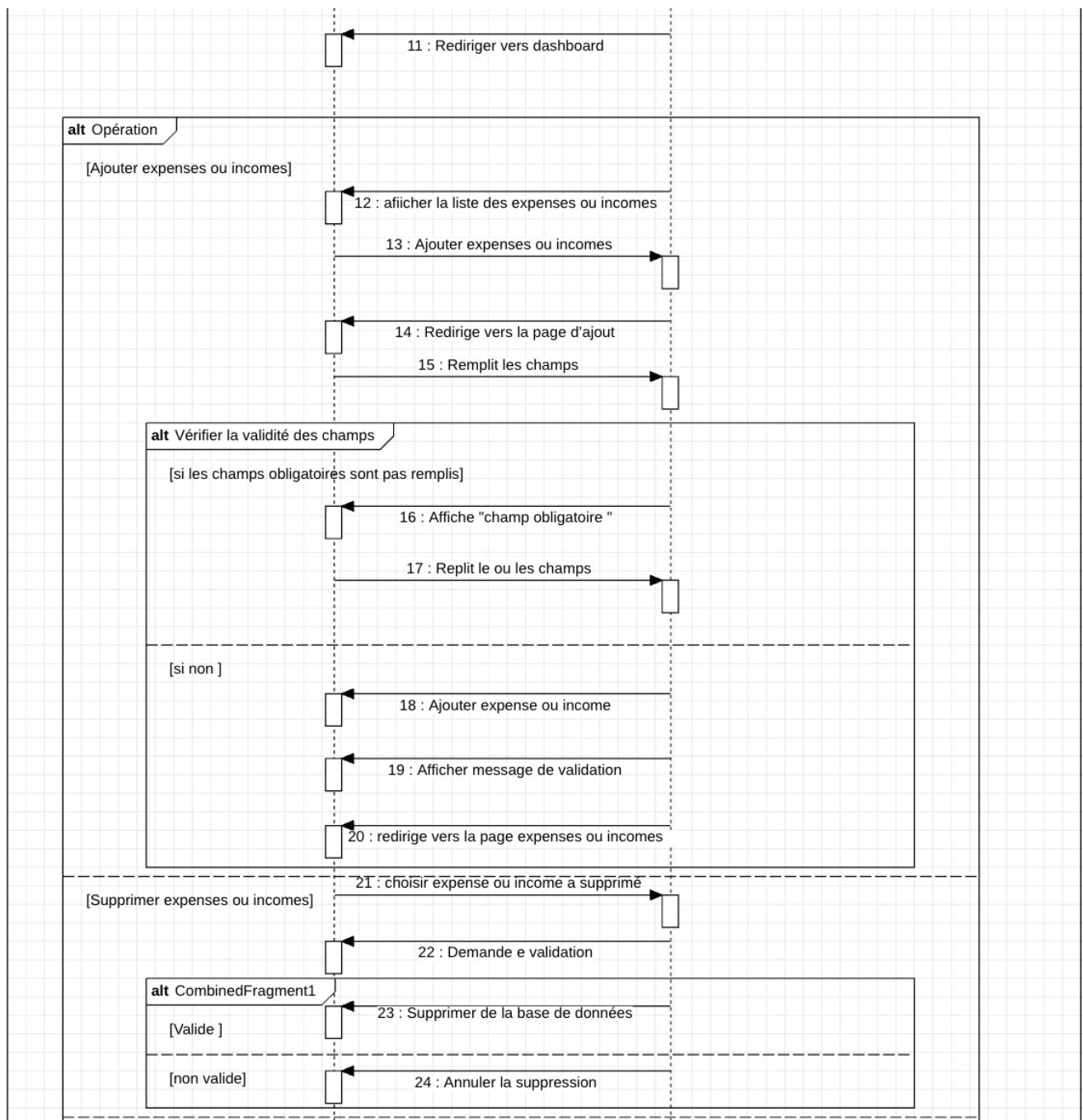


FIGURE 2.4 – diagramme de séquence 2

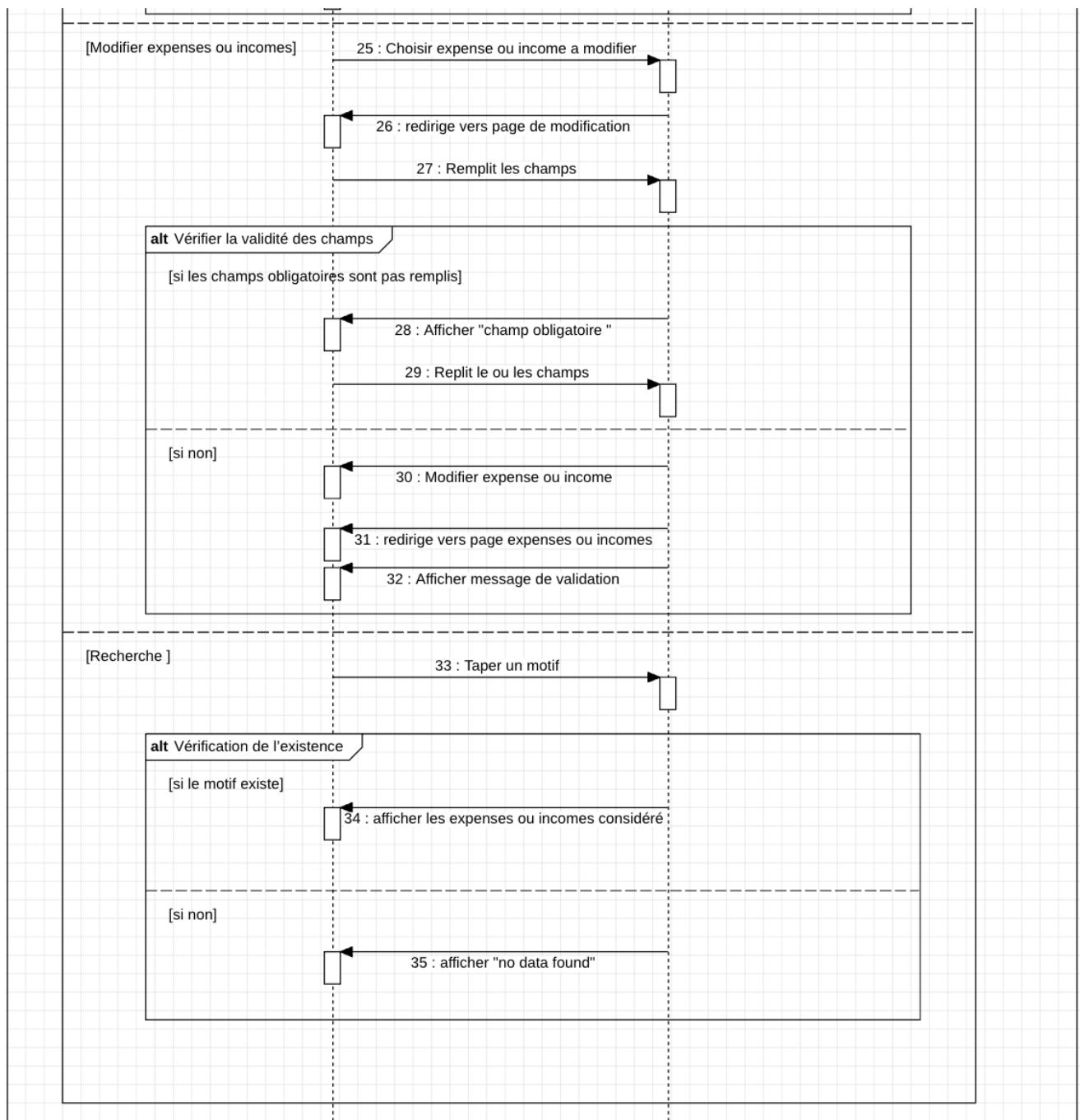


FIGURE 2.5 – diagramme de séquence 3

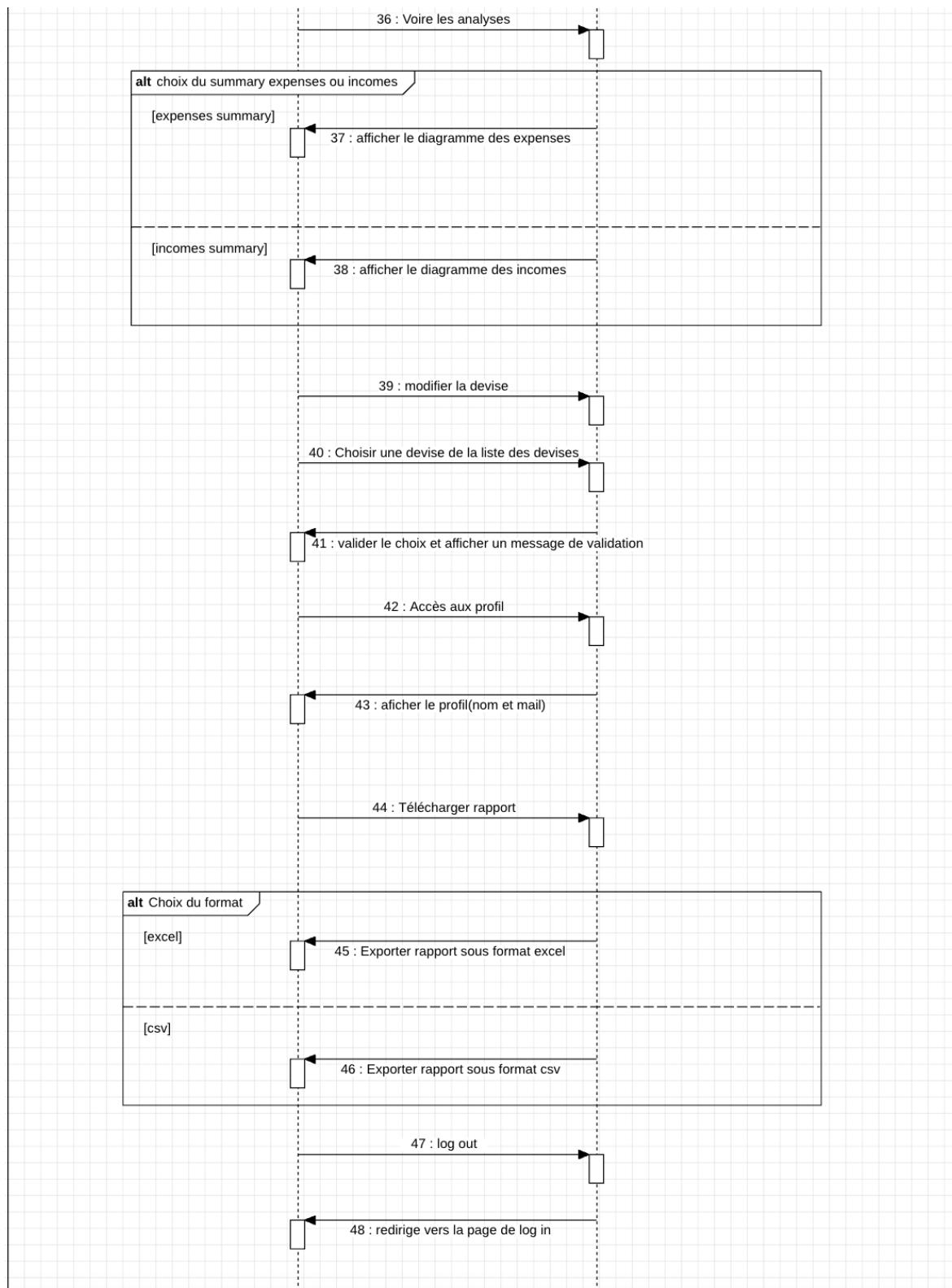


FIGURE 2.6 – diagramme de séquence 4

2.4 Conclusion

Nous avons structuré notre application de gestion des dépenses personnelles autour de compétences architecturales avec une distinction nette, et l'aide fut fournie par des diagrammes de conception pour donner le développement clair et consistant. A travers plusieurs schémas de cahier de charges, séquences de classe et autres diagrammes, nous avons pu tracer des fondations solides pour démarrer et réussir notre projet.

Chapitre 3

Etude pratique

3.1 Introduction

Idée d'amélioration pour faciliter votre suivi budgétaire, la création d'une application web dédiée à la gestion des dépenses est une solution efficace. solution, afin de rendre les processus de financement personnel automatisés et simplifiés. L'application il est possible de la développer en utilisant des technologies informatiques appropriées. Parmi les exemples figurent Django (un framework web), Python (un langage de programmation) et l'HTML. et à CSS (cascading style sheets).

3.2 Outils

3.2.1 Django



FIGURE 3.1 – django logo

Django est disponible en tant que framework Web open-source et utilise largement Python pour créer des fichiers, des paramètres et des modèles de données. Il est conçu pour relever deux défis principaux : les exigences rigoureuses des développeurs Web très expérimentés et les délais intenses d'une salle de rédaction.

Django met l'accent sur les points suivants :

Possibilité de branchement et de réutilisation des composants

Développement rapide

Le principe de non répétition

Django fonctionne selon le principe MVT, Model/View/Template, similaire au modèle MVC .

L'architecture MVT de Django est composée de trois couches :

Le Modèle :

Le Modèle définit la structure des données et les représente sous forme d'objets ou classes d'objets. C'est la couche qui interagit avec la base de données via un ORM.

La vue :

La vue contrôle ce que l'utilisateur peut consulter. Elle récupère les requêtes web et renvoie la réponse correspondant à chaque requête.

Le Template :

Le Template définit l'interface graphique que consulte l'utilisateur. Il permet de contrôler l'interaction avec l'utilisateur et gère la façon dont la réponse à une requête est retournée à celui-ci.

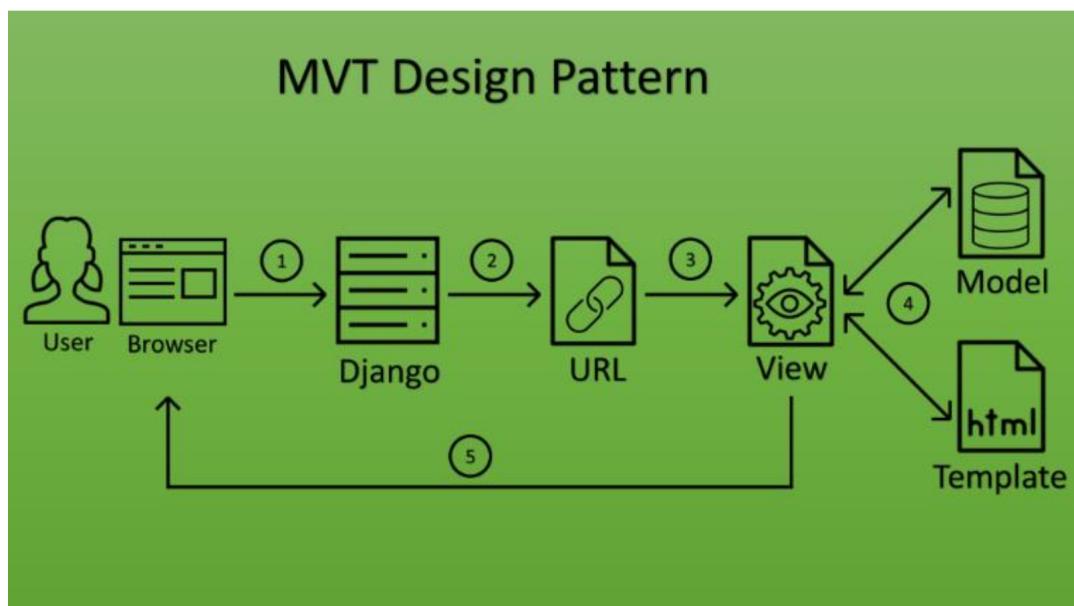


FIGURE 3.2 – le fonctionnement de Django (mvt)

Comment ça marche

Pour mieux comprendre cette architecture, voyons -à travers un exemple classique- comment ces trois couches interagissent entre elles :

Un utilisateur formule une requête HTTP pour une ressource sur un navigateur par exemple Django récupère la requête et la considère comme une URL puis cherche le schéma correspondant (URL pattern) à celle-ci dans un fichier appelé « url.py » (Un schéma d'URL fait correspondre une URL à une vue) Une fois le chemin d'URL trouvé, Django appelle la vue correspondant à l'URL et se trouvant dans le fichier « views.py »

La vue est une fonction ou une méthode qui exécute des instructions pour répondre à

la requête.

La vue cherche les modèles, dans le fichier « models.py », et les templates (fichiers HTML, fichiers statiques : CSS, JS, media ...etc) qui doivent intervenir pour retourner une réponse à l'utilisateur :

La vue peut interagir avec le modèle lorsqu'il s'agit d'interroger la base de données et de gérer les données associées à la requête de l'utilisateur La vue fait appel au template nécessaire pour construire et compléter la réponse à la requête HTTP L'utilisateur peut consulter la réponse renvoyé par la vue sur son navigateur.

3.2.2 Python



FIGURE 3.3 – python logo

Python est un langage de programmation informatique généraliste. Contrairement à HTML, CSS ou JavaScript, son usage n'est donc pas limité au développement web. Il peut être utilisé pour tout type de programmation et de développement logiciel. On s'en sert notamment pour le développement back end d'applications web ou mobile, et pour le développement de logiciels et d'applications pour PC. Il permet également d'écrire des scripts système, afin de créer des instructions pour un système informatique. Par ailleurs, Python est le langage informatique le plus populaire pour le traitement Big Data, l'exécution de calculs mathématiques ou le Machine Learning. De manière générale, il s'agit du langage de prédilection pour la Data Science.

principales caractéristiques de Python

Codage lisible.

Modules de support étendus.

Développement communautaire.

Intégration facile de services web.
Structure de données facile à utiliser.
Typé dynamiquement.
Orienté objet.
Indépendant de la plateforme.
Applications de bureau basées sur une interface graphique.

3.2.3 Html



FIGURE 3.4 – Html logo

Développé par le W3C (World Wide Web Consortium) et le WHATWG (Web Hyper-text Application Technology Working Group), le format ou langage HTML est apparu dans les années 1990. Il a progressivement subi des modifications et propose depuis 2014 une version HTML5 plus aboutie.

L'HTML est ce qui permet à un créateur de sites Web de gérer la manière dont le contenu de ses pages Web va s'afficher sur un écran, via le navigateur. Il repose sur un système de balises permettant de titrer, sous-titrer, mettre en gras, etc., du texte et d'introduire des éléments interactifs comme des images, des liens, des vidéos... L'HTML est plus facilement compris des robots de crawl des moteurs de recherche que le language JavaScript, aussi utilisé pour rendre les pages plus interactives.

Structure de base d'un document HTML

Voici un exemple

```
<!DOCTYPE html>
<html>

    <head>
        <title> Title here </title>
    </head>

    <body>
        Web page content goes here.
    </body>

</html>
```

FIGURE 3.5 – Structure de base d'un document HTML

Le rôle du HTML

est donc crucial puisqu'il va être notre langage privilégié pour indiquer aux navigateurs ce quoi est constituée chaque page et ce qu'ils doivent afficher. Grâce au HTML, on va par exemple pourvoir indiquer que tel contenu est un texte qui n'est qu'un paragraphe, que tel autre contenu est un texte qui est un titre de niveau 1 dans notre page, que tel autre contenu est une liste, un lien, etc. En plus de cela, le HTML va également nous permettre d'insérer différents médias (images, vidéos, etc.) dans nos pages web en indiquant au navigateur » à cette place- là dans ma page, je veux que s'affiche cette image « . Notez que dans ce cas précis, pour que le navigateur affiche la bonne image, on va lui fournir l'adresse de l'image dans le code HTML.

3.2.4 css



FIGURE 3.6 – css logo

Le terme CSS est l'acronyme anglais de Cascading Style Sheets qui peut se traduire par "feuilles de style en cascade". Le CSS est un langage informatique utilisé sur l'internet pour mettre en forme les fichiers HTML ou XML puisque leur options stylistiques d'un document sont quelque peu limitées. Ainsi, les feuilles de style, aussi appelé les fichiers CSS, comprennent du code qui permet de gérer le design d'une page en HTML.

À quoi sert le CSS

Le CSS permet donc de définir l'esthétique d'un site web. Sans CSS, une page web ne serait qu'une succession d'éléments noirs sur fond blanc les uns à la suite des autres.

C'est avec du code CSS que l'on définit par exemple :

la police d'un texte et son aspect (couleur, taille, etc.)

les marges et le rembourrage (padding) entourant un élément

l'apparence d'un menu

la création d'éléments géométriques

la mise en place de grilles d'images

l'ajout de bordure

l'application de différents effets ou animations simples

le changement de présentation d'un site selon l'appareil utilisé par l'utilisateur (téléphone

portable, ordinateur, tablette)

la manière dont on navigue dans un site internet

La structure d'une déclaration CSS

Vous pouvez utiliser une déclaration CSS pour déterminer quels éléments de votre document électronique doivent prendre quelles valeurs ou propriétés. Dans sa structure de base, la déclaration se compose d'un sélecteur et d'accolades. Dans ces parenthèses, vous énumérez les déclarations de propriétés, qui sont séparées par des points-virgules. Chaque directive se compose du nom de la propriété, de deux points et d'une valeur spécifique. Après la dernière déclaration de propriété et avant la parenthèse finale, il est possible de placer un autre point-virgule, mais il n'est pas obligatoire. La déclaration CSS dans le diagramme ci-dessous permet de s'assurer que le titre h1 est affiché dans la couleur bleue et la taille de police 12 :

Structure d'une déclaration CSS

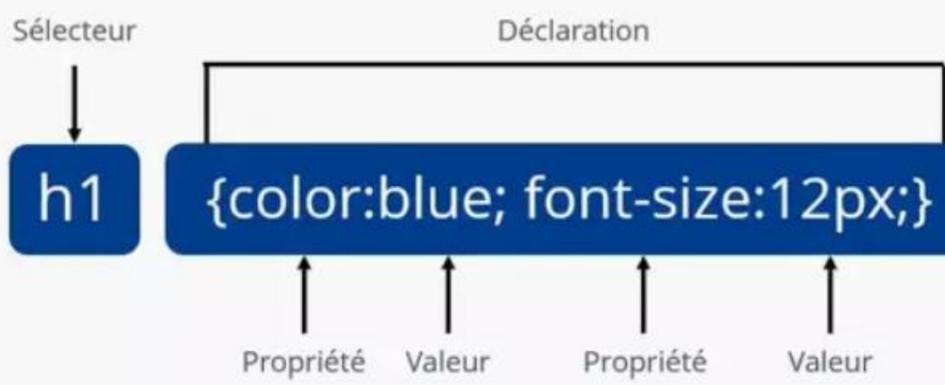


FIGURE 3.7 – Structure de CSS

3.2.5 js

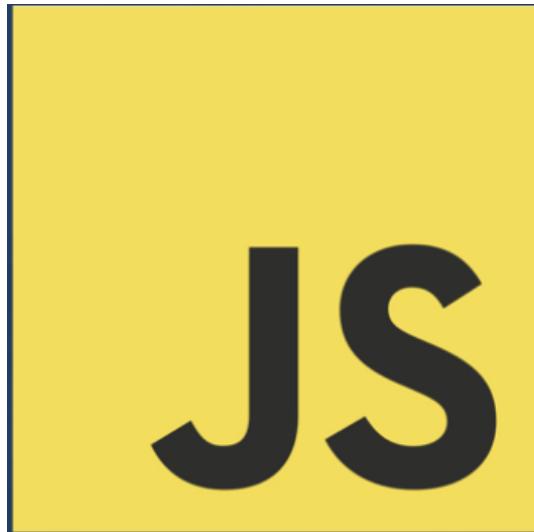


FIGURE 3.8 – js logo

Caractéristiques de JS

1. Langage de Script Côté Client : JavaScript est principalement employé côté navigateur pour manipuler le contenu HTML/CSS et interagir avec l'utilisateur, évitant ainsi de recharger la page.
2. Polyvalent et Dynamique : JavaScript permet l'utilisation de différents paradigmes de programmation, tels que la programmation orientée objet, fonctionnelle et impérative. Il convient parfaitement à la création de applications web complexes, jeux vidéo, applications mobiles et bien d'autres.
3. Syntaxe Simple : La syntaxe simple et expressive de JavaScript en fait un langage apprécié, accessible aux débutants mais proposant également des fonctionnalités avancées pour les développeurs expérimentés.
4. Modèle d'Objet Document (DOM) : En JavaScript, il est possible de manipuler le DOM qui représente les éléments HTML d'une page sous forme d'arborescence. Cela permet de faciliter la modification dynamique de l'interface utilisateur ainsi que la gestion des événements utilisateurs.
5. Large Écosystème : JavaScript est doté d'une large gamme de frameworks et de bibliothèques tels que React.js, Vue.js et AngularJS facilitent la création d'applications web modernes et réactives.
6. Exécution Rapide : JavaScript étant un langage interprété, il peut s'exécuter directement dans le navigateur. Cela offre aux utilisateurs des interactions instantanées et des mises à jour en temps réel, sans devoir recharger la page.

Utilisation de JS :

JavaScript est couramment utilisé pour améliorer l'interactivité des pages web en y ajoutant des fonctionnalités comme des animations, des formulaires dynamiques, des effets visuels et même la création de jeux ou d'applications web en temps réel. Il est également possible de développer des applications mobiles hybrides avec JavaScript. JavaScript est un élément essentiel du développement front-end moderne, se combinant souvent avec HTML et CSS pour former une base solide.

3.2.6 staruml



FIGURE 3.9 – staruml logo

StarUML offre des fonctions de modélisation orientée objet basée, avec un support complet d'UML, conforme aux normes UML 2.x :

Diagrammes d'activité

Diagrammes de séquence

Diagrammes de communication

Diagrammes chronologiques

Diagrammes de transitions d'états

Diagrammes de flux d'informations

Diagrammes de vue d'ensemble d'interactions

Diagrammes de profils

Diagrammes de classes

Diagrammes de structure composite

Diagrammes de composants

Diagrammes d'objets

Diagrammes de paquetages

Diagrammes de cas d'utilisation

StarUML dispose également de capacités de modélisation en SysML :

Diagrammes de besoins

Diagrammes de blocs

Diagrammes internes de blocs

StarUML permet également de modéliser avec des représentations graphiques plus anciennes telles que :

Organigrammes

Diagrammes Entités-Associations (ERD)

Diagrammes de flux de données (DFD)

3.2.7 postman



FIGURE 3.10 – postman logo

Les développeurs et les équipes de développement utilisent fréquemment Postman afin de faciliter et accélérer le processus de création d'API. En effet, cet outil leur offre un environnement centralisé qui permet à la fois de tester, déboguer et documenter les services web. Il s'avère être très bénéfique pour les développeurs frontend, backend et ingénieurs QA (Quality Assurance) qui se concentrent sur des projets basés sur des API.

Caractéristiques de Postman :

1. Environnement de Développement d'API : La plateforme Postman propose une interface conviviale permettant de créer, tester et déboguer des API RESTful ainsi que SOAP. Il offre aux développeurs la possibilité de créer des requêtes HTTP, d'envoyer les données, d'examiner les réponses et de vérifier le fonctionnement des API.
2. Collection d'API : Les collections Postman permettent de regrouper des requêtes, des scripts et des tests ensemble, ce qui rend la gestion et la collaboration sur des ensembles d'API complexes plus faciles. Il est facile de partager et d'importer/exporter des collections.
3. Tests Automatisés : Postman offre la possibilité d'écrire des tests automatisés en utilisant JavaScript (en utilisant la bibliothèque de tests Chai), ce qui permet de vérifier automatiquement le bon fonctionnement des API grâce à des assertions sur les réponses HTTP.
4. Environnements et Variables : Avec Postman, il est possible de définir des variables d'environnement afin de gérer les configurations spécifiques. Cela inclus par exemple les URL des serveurs pour différents environnements tels que le développement, le test et la production. De plus, l'utilisation de ces variables dans les requêtes permet une gestion simplifiée des configurations.

5. Documentation d'API : En générant automatiquement des vues conviviales basées sur les collections, Postman facilite la création de documentation pour les API en incluant des exemples de requêtes, des descriptions et des détails sur les paramètres.
6. Intégration Continue : Il est possible d'intégrer Postman aux pipelines CI/CD (Intégration Continue/Déploiement Continue) afin de rendre automatiques les tests d'API et la validation des modifications apportées aux services web pendant le processus de développement logiciel.

3.3 Réalisation du projet

Cette application contient 3 espaces principaux :

-Page d'Accueil,

-Page de Connexion,

-L'interface de l'application (dashboard,page des 'expense',page des 'incomes',page de profile,page pour changer la devise et les pages des statistiques).

Page d'Accueil

Notre page d'accueil a une disposition amicale et générale qui montre les caractéristiques primordiales ainsi que les avantages de notre plateforme pour la gestion des coûts personnels. Pour aider les utilisateurs à bien gérer leurs finances, il y a des informations très cruciales et des astuces pratiques. Il y a deux boutons bien visibles sur la page permettant aux utilisateurs de se connecter ou de créer un compte afin qu'ils puissent commencer à utiliser l'application.

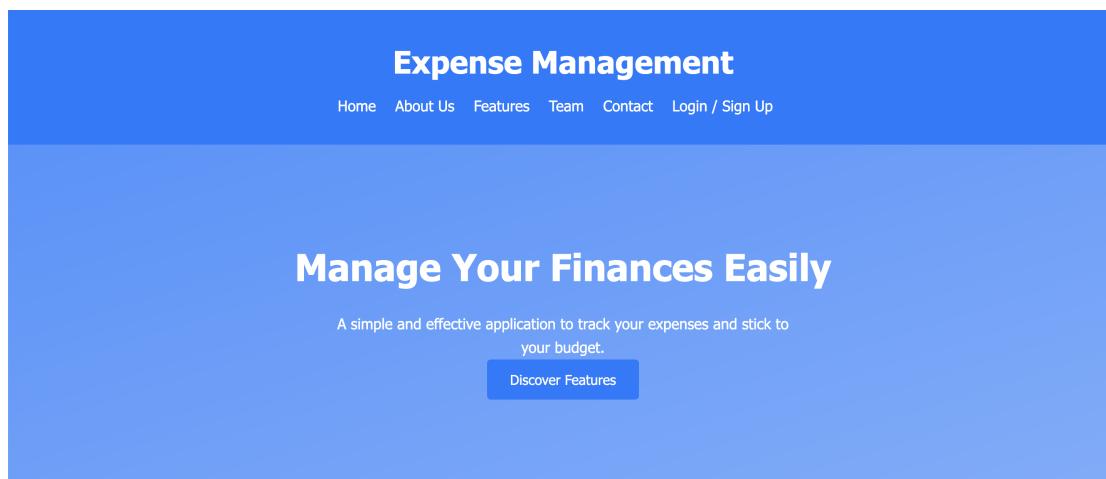


FIGURE 3.11 – application 1

About Us

At Expense Management, our mission is to help you manage your finances better by providing simple and effective tools to track your expenses and stay on budget.

Features

Expense Tracking

Record and categorize your expenses to get an overview of your finances.

Budget Management

Create budgets for different categories and track your progress.

FIGURE 3.12 – application 2

Our Team



Anas AIT EL QADI

Founder & CEO



Mohamed Taha IZMAR

Founder & Chief Financial Officer

FIGURE 3.13 – application 3

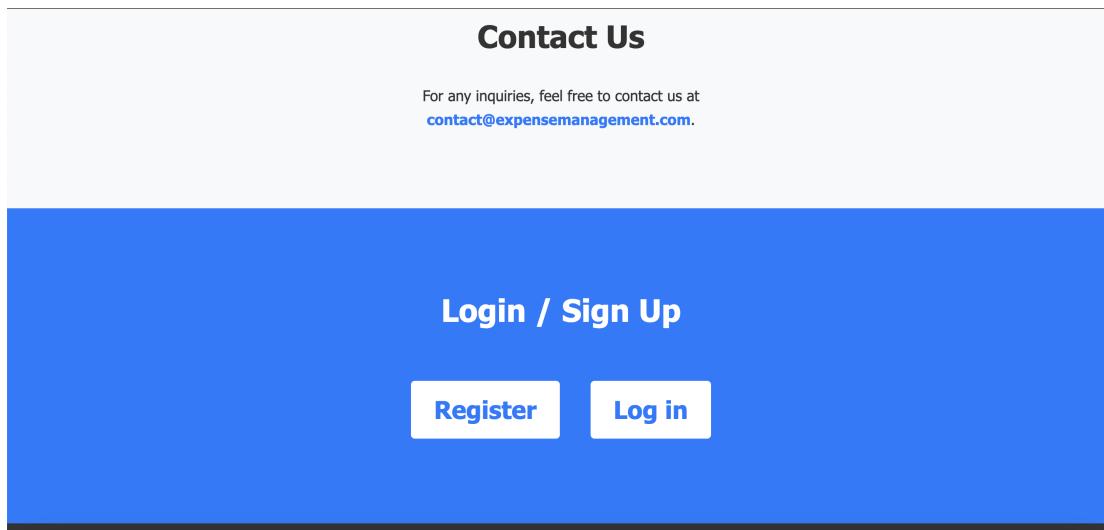


FIGURE 3.14 – application 4

Page de Connexion :

La page de connexion est conçue pour être à la fois simple et sécurisée afin que les utilisateurs puissent saisir leur nom d'utilisateur et leur mot de passe pour accéder à leur compte. Une interface simplifiée garantit une expérience utilisateur fluide avec des champs de saisie clairs.

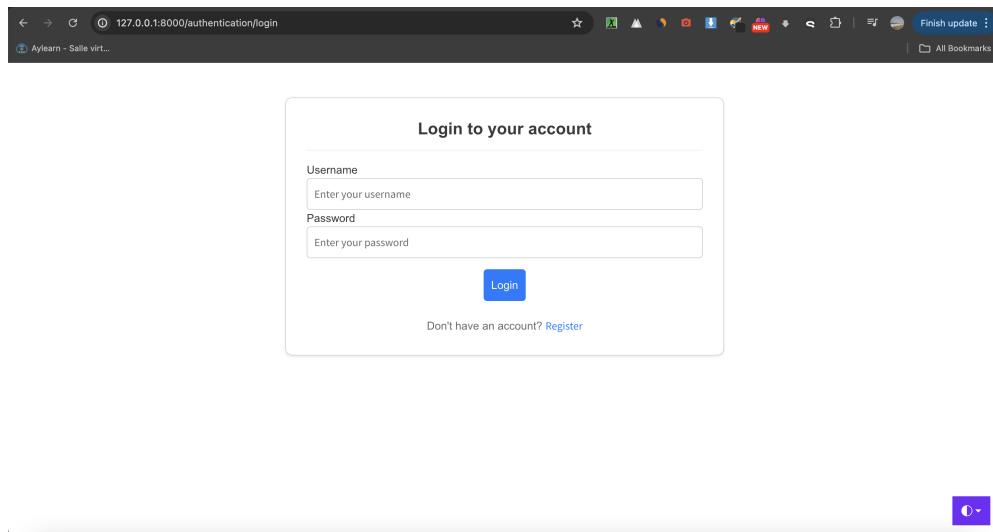


FIGURE 3.15 – log in

Page de Création de Compte :

Les nouveaux utilisateurs peuvent se créer un compte sur la page de création de compte en inscrivant simplement leur nom d'utilisateur, leur adresse email et leur mot de passe. Conçue pour être intuitive, cette page guide les utilisateurs tout au long du processus d'inscription grâce à des instructions claires et à une validation en ligne des données qu'ils ont fournies.

Register for a Free Account

Username

Email

Password

SHOW

Already registered? [Log in](#)

FIGURE 3.16 – registre

Dashboard :

La Page Principale offre donc un graphe général de toutes les ressources et toutes les charges des six mois en cours , et aussi offre le total des dépenses , des ressources et le budget .

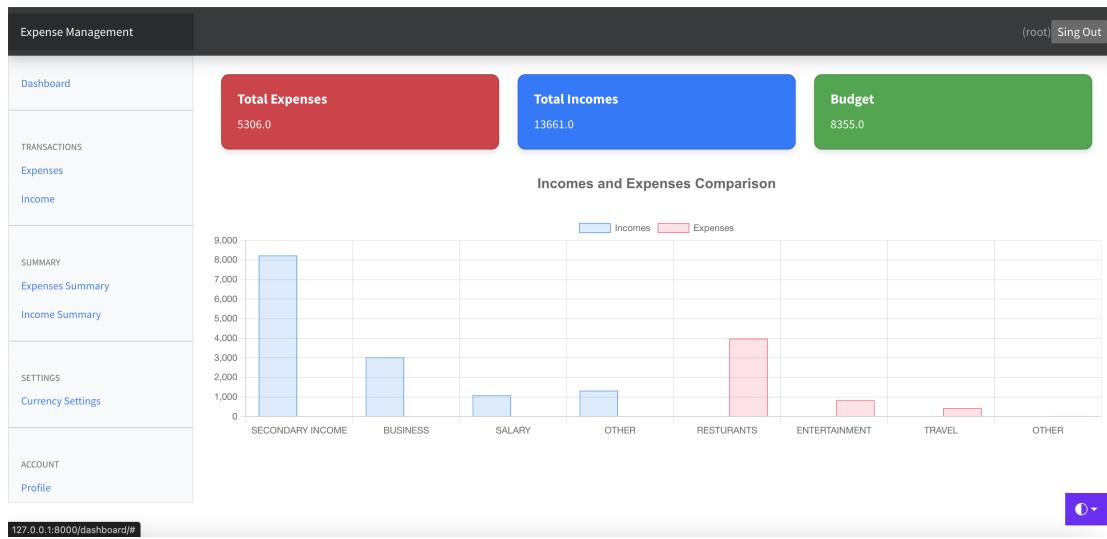
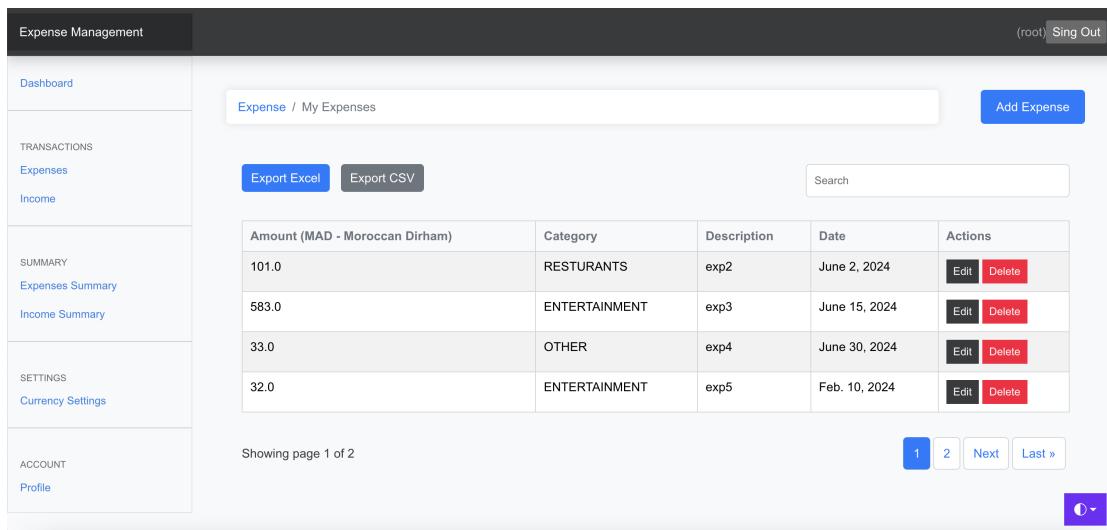


FIGURE 3.17 – dashboard

Page d'ajout , suppression , modification et affichage des ressources et des dépenses :

Durant ces pages , l'utilisateur peut ajouter , visualiser , modifier , supprimer et aussi la recherche par la date ou n'importe quel donnée , en plus des fonctionnalités de l'exportation des rapports sous form de CSV ou EXCEL.



The screenshot shows the 'Expense Management' application interface. On the left, there is a sidebar with the following navigation options:

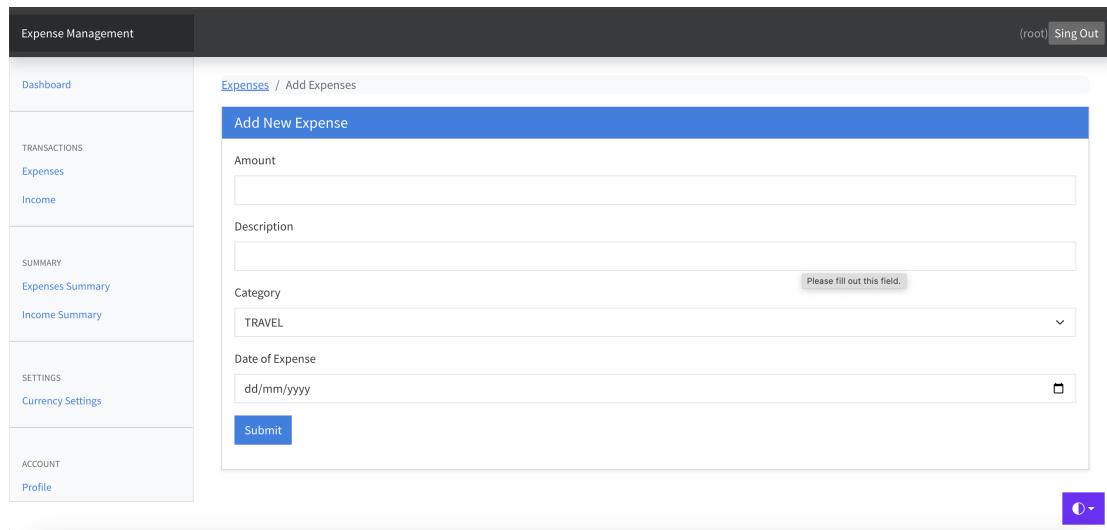
- Dashboard**
- TRANSACTIONS**
 - Expenses
 - Income
- SUMMARY**
 - Expenses Summary
 - Income Summary
- SETTINGS**
 - Currency Settings
- ACCOUNT**
 - Profile

The main content area displays a table of expenses:

Amount (MAD - Moroccan Dirham)	Category	Description	Date	Actions
101.0	RESTURANTS	exp2	June 2, 2024	Edit Delete
583.0	ENTERTAINMENT	exp3	June 15, 2024	Edit Delete
33.0	OTHER	exp4	June 30, 2024	Edit Delete
32.0	ENTERTAINMENT	exp5	Feb. 10, 2024	Edit Delete

Below the table, there is a search bar and pagination controls (1, 2, Next, Last).

FIGURE 3.18 – expenses



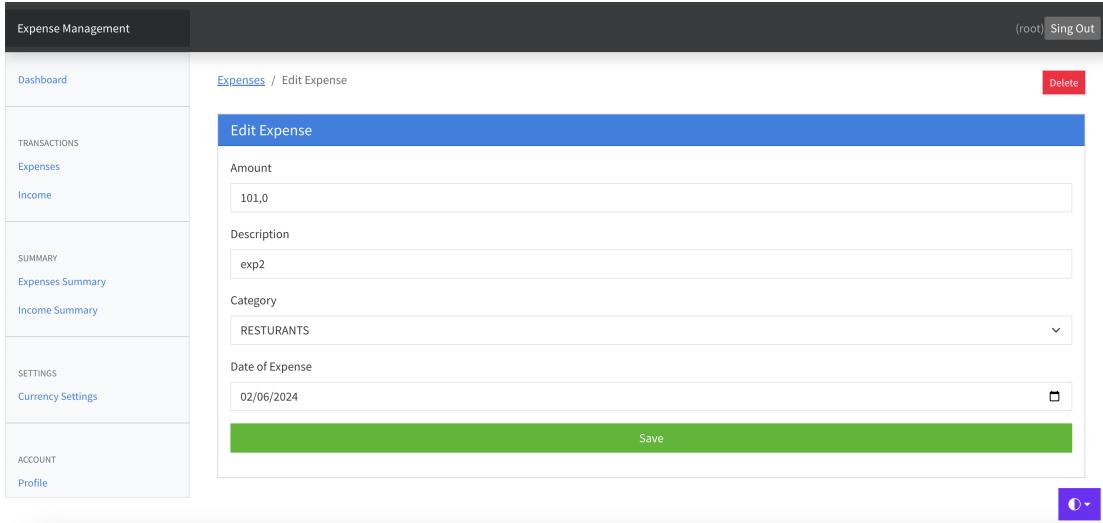
The screenshot shows the 'Expense Management' application interface. On the left, there is a sidebar with the following navigation options:

- Dashboard**
- TRANSACTIONS**
 - Expenses
 - Income
- SUMMARY**
 - Expenses Summary
 - Income Summary
- SETTINGS**
 - Currency Settings
- ACCOUNT**
 - Profile

The main content area displays a form titled 'Add New Expense':

Amount	<input type="text"/>
Description	<input type="text"/>
Category	<input type="text"/> Please fill out this field.
Date of Expense	<input type="text"/> dd/mm/yyyy
Submit	

FIGURE 3.19 – add expenses



Expense Management

(root) Sing Out

Dashboard

TRANSACTIONS

Expenses

Income

SUMMARY

Expenses Summary

Income Summary

SETTINGS

Currency Settings

ACCOUNT

Profile

Expenses / Edit Expense

Delete

Edit Expense

Amount
101,0

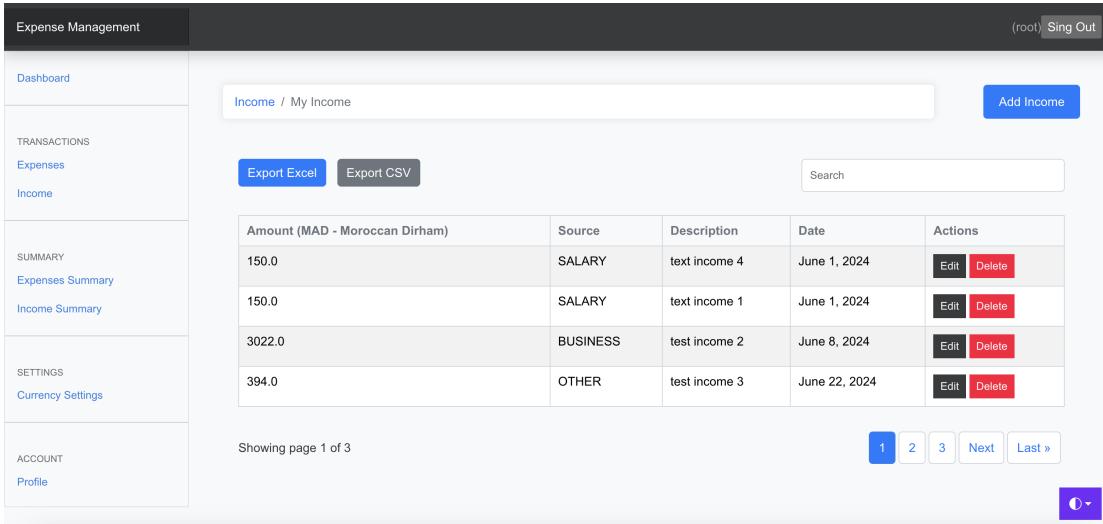
Description
exp2

Category
RESTAURANTS

Date of Expense
02/06/2024

Save

FIGURE 3.20 – edit expenses



Expense Management

(root) Sing Out

Dashboard

TRANSACTIONS

Expenses

Income

SUMMARY

Expenses Summary

Income Summary

SETTINGS

Currency Settings

ACCOUNT

Profile

Income / My Income

Add Income

Export Excel Export CSV Search

Amount (MAD - Moroccan Dirham)	Source	Description	Date	Actions
150.0	SALARY	text income 4	June 1, 2024	Edit Delete
150.0	SALARY	text income 1	June 1, 2024	Edit Delete
3022.0	BUSINESS	test income 2	June 8, 2024	Edit Delete
394.0	OTHER	test income 3	June 22, 2024	Edit Delete

Showing page 1 of 3

1 2 3 Next Last »

FIGURE 3.21 – incomes

Pages des analyses :

Les pages d'analyse fournissent des visualisations détaillées des dépenses et des revenus. L'analyse des dépenses et des revenus est présentée sous forme de graphiques , offrant une compréhension claire et visuelle des différentes catégories des dépenses ou des ressources.

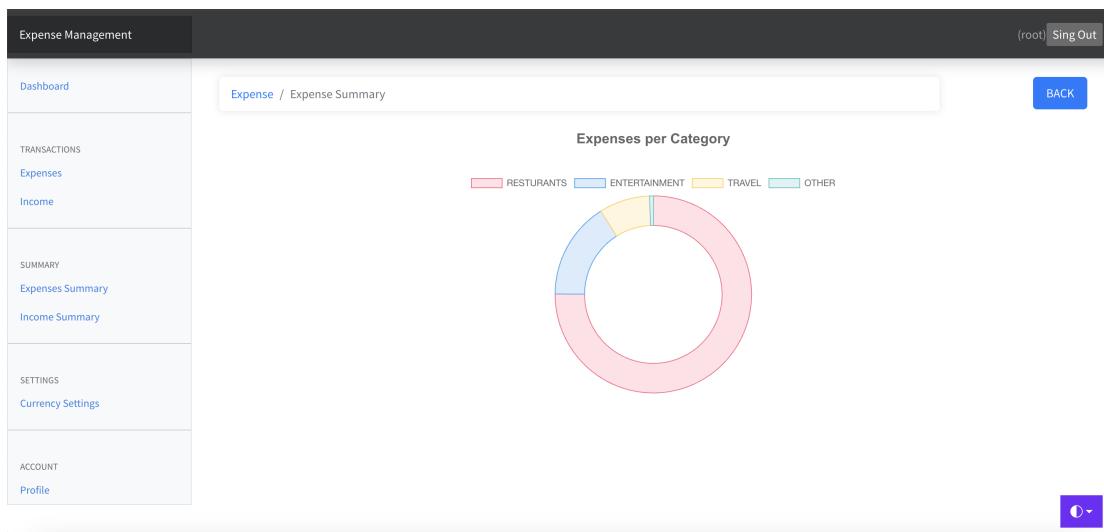


FIGURE 3.22 – expenses summary

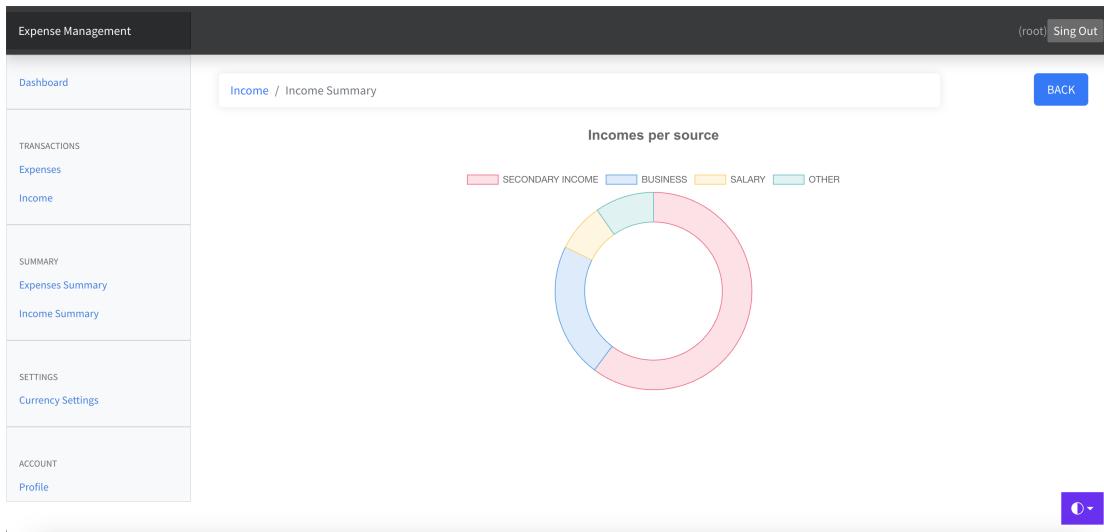


FIGURE 3.23 – incomes summary

Page de devise :

C'est cette page de changent de la devise qui offre à l'utilisateur la possibilitésde changer la devise selon leur pays.

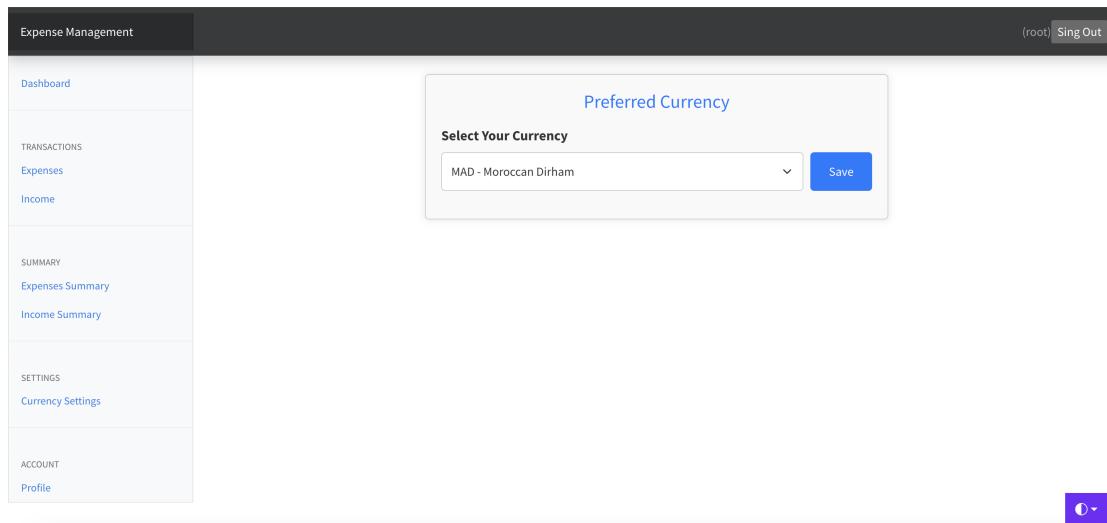


FIGURE 3.24 – devise

Page de profile :

cette page offre à l'utilisateur la possibilité de visualiser leur profil qui comporte le nom de l'utilisateur et le mail

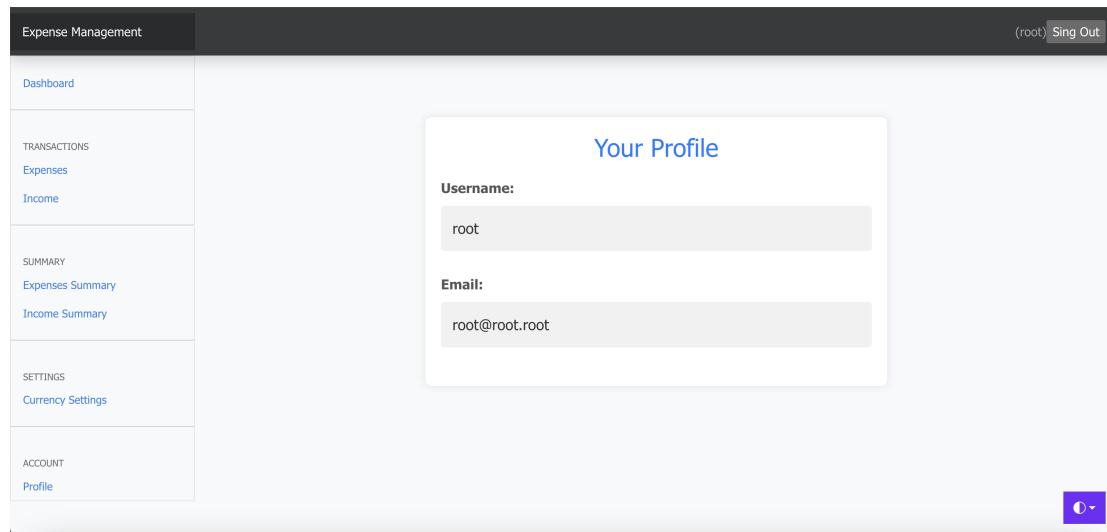


FIGURE 3.25 – profile

3.4 Conclusion

Pour conclure cette section consacrée à l'étude pratique de notre projet, nous avons minutieusement analysé les outils de développement utilisés et la séquence des interfaces. Nos éléments sont essentiels afin de garantir la solidité technique et l'ergonomie de notre application pour gérer les dépenses personnelles. Le choix méticuleux des outils et la planification minutieuse des interfaces représentent deux étapes cruciales pour réaliser notre vision.

Conclusion générale

En conclusion, notre application web dédiée à la gestion des dépenses et revenus constitue une avancée majeure dans le domaine de la gestion financière individuelle. Nous avons créé une plateforme facile à utiliser et sécurisée qui permet aux utilisateurs de suivre et gérer leurs finances au quotidien, chaque mois et chaque année avec efficacité.

En utilisant des technologies modernes et des approches de conception avancées, nous avons développé des fonctionnalités importantes telles que la classification intelligente des transactions et la création de rapports détaillés. Cela permet d'améliorer notre compréhension et l'optimisation des habitudes financières de nos utilisateurs.

Ce projet a prouvé combien il est crucial d'avoir une interface utilisateur facile à utiliser et une infrastructure technique solide pour assurer la meilleure expérience possible aux utilisateurs. Le développement nous a confrontés à des défis qui ont contribué à améliorer nos compétences en matière de développement web, de design UX/UI et de gestion de projet. Ces obstacles ont renforcé notre aptitude à proposer des solutions innovantes et fiables dans le domaine financier personnel. Nous croyons fermement que notre application sera d'une valeur inestimable pour ses utilisateurs, capable de leur fournir les moyens indispensables à une gestion financière plus transparente, contrôlée et optimisée.

Ce projet représente une double victoire, tant sur le plan technique que dans la promotion de meilleures pratiques financières individuelles. Notre solution bénéficiera des leçons apprises et des succès obtenus au cours de ce parcours enrichissant, qui constitueront une base solide pour les développements et améliorations à venir.

Perspectives

Afin de faire évoluer notre application de manière constante, nous avons identifié plusieurs initiatives stratégiques visant à améliorer son utilité et l'expérience utilisateur.

Tout d'abord, nous souhaitons internationaliser l'application en incluant des langues additionnelles. Non seulement cela permettra d'élargir l'accès à l'application pour un public plus large, mais également de satisfaire aux exigences particulières de nos utilisateurs internationaux. Les traductions incluront tant l'interface utilisateur que tous les documents de support et d'aide.

Par la suite, on compte réaliser une amélioration majeure du design. L'application sera complètement revue sur le plan visuel afin de lui donner un aspect plus moderne et d'améliorer son ergonomie. Le but est de concevoir une interface plus facile à utiliser pour tous les utilisateurs, qui soit intuitive, cohérente et agréable du point de vue esthétique.

En outre, nous mettrons à disposition une fonctionnalité de mode nuit qui permettra aux utilisateurs de passer en un thème sombre. Beaucoup d'utilisateurs préfèrent cette fonctionnalité car elle réduit la fatigue des yeux et améliore l'expérience dans des conditions de faible luminosité.

De plus, il est primordial de développer une version mobile de l'application. Cette version a été conçue spécifiquement pour les smartphones et les tablettes afin d'offrir une expérience utilisateur optimale, avec une interface adaptée aux écrans de petite taille qui garantit une utilisation fluide. Le but est de rendre toutes les fonctionnalités de l'application facilement accessibles et efficaces pour les utilisateurs, peu importe leur emplacement.

Nous nous engageons à proposer une application de haute qualité, qui évolue avec les besoins des utilisateurs. Nous croyons fermement que ces améliorations contribueront à accroître la satisfaction de nos utilisateurs et à répondre aux demandes croissantes du marché.

Webographie

1. Pour UML : <https://staruml.io/user-guide>
2. Pour Python et Django : <https://www.python.org/downloads/>
3. Pour l'éditeur de texte VS code : <https://code.visualstudio.com/download>
4. Pour postman : <https://learning.postman.com/docs/getting-started/introduction/>
5. Pour w3schools (guide por HTML/CSS/JS) : <https://www.w3schools.com/>
6. Pour tradogram : <https://www.tradogram.com/>
7. Pour tipalti : <https://tipalti.com/>