



Licence MIASHS troisième année

Rapport de projet informatique

KEYPER : Application web de gestion de foyer : "Get things done, together"

Projet réalisé du 10 octobre 2025 au 10 décembre 2025

Membres du groupe

Saba Camilia 43003141
Salekha FALL 43013188
Youcef HAMISSI 44008745
Jorys KANCEL 45005523

Lien GitHub du projet: <https://github.com/krat0s-web/KeyPer>

Table des matières

1	Introduction	3
2	Présentation de l'organisme	3
2.1	Présentation générale	3
2.2	Organisation de l'équipe	4
2.3	Compétences et objectifs	5
3	Présentation du projet	6
3.1	Analyse de l'existant	6
3.2	Critique de l'existant	6
3.3	Problématique	6
3.4	Solution proposée	7
3.5	Comparaison des approches	7
3.6	Choix de la méthodologie	8
3.7	Le choix de la méthode	8
4	Mise en place du projet avec SCRUM	9
4.1	Définition de la méthode Scrum	9
4.2	Les rôles dans l'équipe	9
4.3	Artefacts Scrum	10
4.4	Événements Scrum	10
4.5	Gestion des outils	10
4.6	Planification des sprints	11
5	Modélisation du système	12
5.1	Diagramme des cas d'utilisation	12
5.2	Modèle des données	14
5.3	Maquettes des interfaces	15
6	Conclusion	16

1 Introduction



FIGURE 1 – Logo KEYPER

Keyper est une solution innovante de gestion de foyer collaboratif, conçue pour simplifier et optimiser l'organisation quotidienne des familles et des colocataires. Ce cahier des charges a pour objectif de définir précisément les spécifications techniques et fonctionnelles de l'application, ainsi que les modalités de sa mise en œuvre. Dans un contexte où la gestion du foyer devient de plus en plus complexe, Keyper se positionne comme un outil complet permettant de centraliser les tâches ménagères, les courses, les dépenses, les notes, la gestion des repas et la communication entre les membres d'un même foyer. Ce document servira de référence tout au long du développement du projet, en établissant clairement les attentes, les contraintes et les objectifs à atteindre. Il s'adresse à la fois aux équipes techniques en charge du développement et aux parties prenantes du projet, garantissant ainsi une vision commune et partagée de l'application à réaliser.

Keyper est une application web innovante conçue pour simplifier et optimiser la gestion des foyers modernes. Dans un monde où l'organisation domestique devient de plus en plus complexe, Keyper apporte une solution complète et intuitive pour les familles et les colocataires. Cette plateforme collaborative permet de centraliser la gestion des tâches ménagères, des courses, de la communication et de l'organisation du foyer en un seul endroit. Face à la multiplication des applications spécialisées, Keyper se distingue par son approche globale et personnalisable, offrant une expérience utilisateur fluide et adaptée aux besoins spécifiques de chaque foyer.

2 Présentation de l'organisme

2.1 Présentation générale

Dans le cadre de notre 5e semestre de Licence MIAGE, nous avons décidé de nous lancer dans le développement de Keyper, une application de gestion de foyer collaborative. Ce projet nous permet de mettre en pratique les compétences acquises au cours de notre formation tout en relevant des défis techniques concrets. Notre motivation commune pour ce projet repose sur notre intérêt pour les technologies web modernes et notre désir de créer un outil utile au quotidien, répondant directement aux besoins soulevés par les utilisateurs, en particulier pendant la période Covid.

2.2 Organisation de l'équipe

Notre équipe est structurée de manière à tirer parti des compétences spécifiques de chaque membre tout en favorisant l'apprentissage mutuel. Notre organisation nous permet de couvrir tous les aspects du développement tout en maintenant une communication fluide. Nous nous réunissons régulièrement pour des points d'avancement et nous échangeons régulièrement nos fichiers, ainsi que des inspirations, liens, captures d'écran et enregistrements d'écran depuis un groupe de discussion WhatsApp, un répertoire github privé, un fichier Figma et un Google docs partagé.

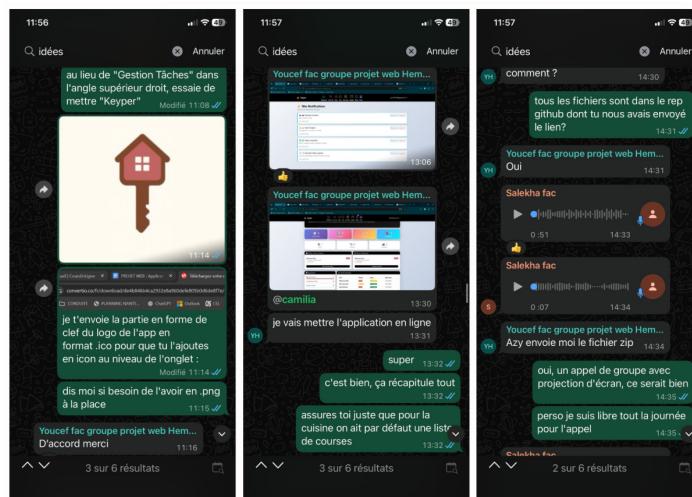


FIGURE 2 – Exemples d'échanges d'idées dans notre groupe de discussion WhatsApp



FIGURE 3 – Extrait du Google docs utilisé dans le cadre de ce projet

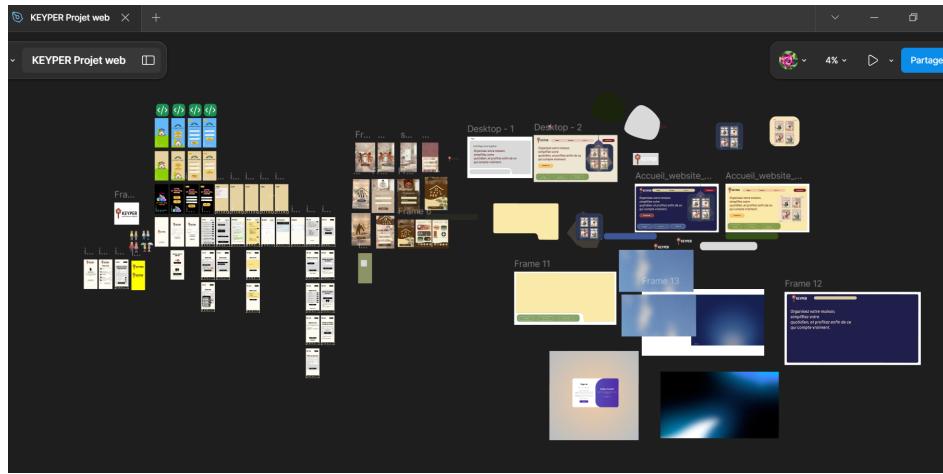


FIGURE 4 – Fichier partagé dans Figma utilisé pour mettre en place des prototypes pour KEYPER

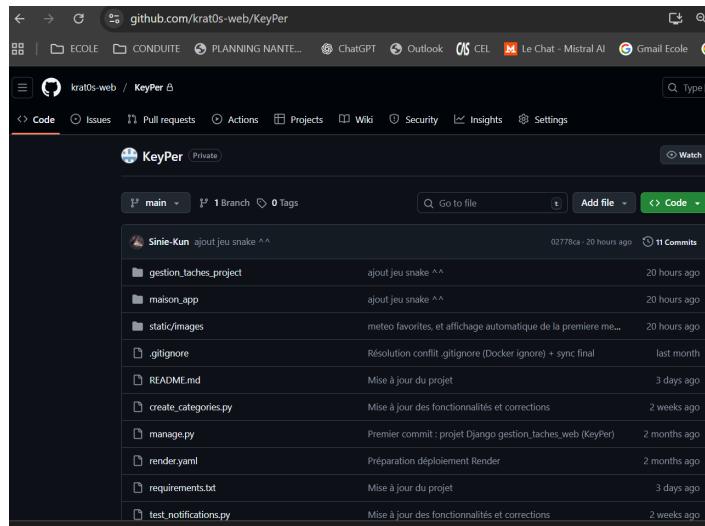


FIGURE 5 – Github utilisé par notre équipe pour échanger des fichiers

2.3 Compétences et objectifs

À travers ce projet, nous visons à approfondir nos compétences en développement web full-stack, en gestion de projet agile et en travail d'équipe. Nos principaux objectifs sont de maîtriser des technologies modernes comme Django, d'approfondir nos connaissances en gestion de bases de données ainsi que celles en développement logiciel, tout en maintenant une bonne organisation et communication au sein du groupe.

3 Présentation du projet

3.1 Analyse de l'existant

Notre analyse du marché des applications de gestion de foyer et des outils de planification et communication disponibles sur nos téléphones a révélé une offre existante fragmentée entre plusieurs solutions spécialisées (liste de courses, gestion des tâches, messagerie) mais peu d'outils offrant une approche intégrée. Les solutions actuelles présentent des lacunes en termes de personnalisation et de flexibilité pour s'adapter aux différents types de foyers. Les applications comme "Tody" (<https://play.google.com/store/apps/details?id=com.looploop.tody&hl=fr>) offrent des fonctionnalités intéressantes mais restent limitées dans leur approche globale de la gestion domestique.

3.2 Critique de l'existant

Les solutions actuelles présentent plusieurs limites notables. D'une part, elles sont souvent trop rigides dans leur structure, ne permettant pas de s'adapter aux spécificités de chaque foyer, qu'il s'agisse d'une famille ou d'un groupe d'amis vivant en colocation etc. D'autre part, la plupart des applications existantes se concentrent sur un seul aspect de la gestion domestique, les tâches ménagères ou les dépenses par exemple, sans offrir de lien fluide entre les différentes fonctionnalités. Les problèmes récurrents identifiés incluent : une mauvaise gestion des rôles et permissions, une absence de personnalisation des espaces, et des interfaces souvent trop complexes pour une utilisation quotidienne. Cela nous a donc amenés à concevoir Keyper comme une plateforme web intuitive.

3.3 Problématique

Comment concevoir une application de gestion de foyer qui réponde efficacement aux besoins variés des utilisateurs, tout en restant simple d'utilisation ? La principale difficulté réside dans la création d'une interface suffisamment intuitive pour être adoptée par tous les membres du foyer, tout en offrant des fonctionnalités assez complètes pour couvrir l'ensemble de leurs besoins en matière d'organisation et de communication. Cette problématique est d'autant plus cruciale dans un contexte post-Covid 19 où la gestion partagée de l'espace de vie est devenue une préoccupation majeure (<https://sante.gouv.fr/archives/archives-famille-enfance/les-familles-a-1-epreuve-de-la-crise-sanitaire/article/les-familles-a-1-epreuve-de-la-crise-sanitaire>).



FIGURE 6 – Problématique : famille mal organisée et qui communique mal

3.4 Solution proposée

Keyper propose une approche innovante avec une architecture en plusieurs parties organisées autour de 4 besoins principaux : la gestion des tâches, la communication, l'organisation de l'espace de vie et le suivi des dépenses. Notre solution se distingue par sa capacité à s'adapter à différents types de foyers (colocations, familles, couples etc.) grâce à un système de personnalisation. L'interface de Keyper assure une expérience utilisateur cohérente et intuitive sur tous les appareils. La possibilité de créer plusieurs foyers permet aux utilisateurs de gérer différents espaces de vie (domicile, résidence étudiante, etc.) depuis un même compte.

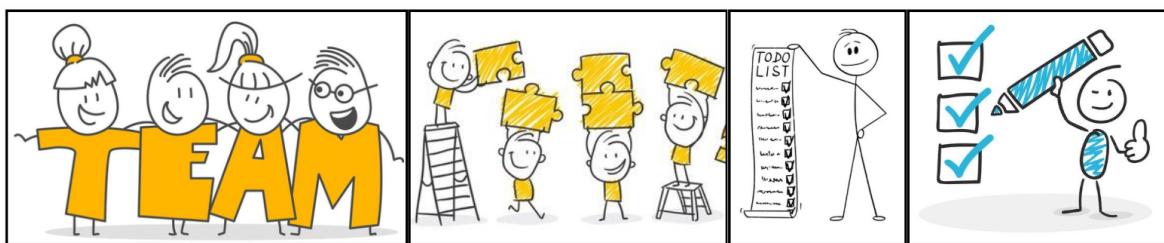


FIGURE 7 – Solution : Famille bien organisée qui communique bien grâce à Keyper

3.5 Comparaison des approches

Notre analyse montre que Keyper se positionne comme une solution plus complète que les alternatives existantes. Contrairement à "Tody" qui se concentre uniquement sur les tâches ménagères, ou "Splitwise" (<https://www.splitwise.com/>) qui ne gère que les dépenses partagées, notre application web intègre l'ensemble des fonctionnalités nécessaires à la gestion d'un foyer.

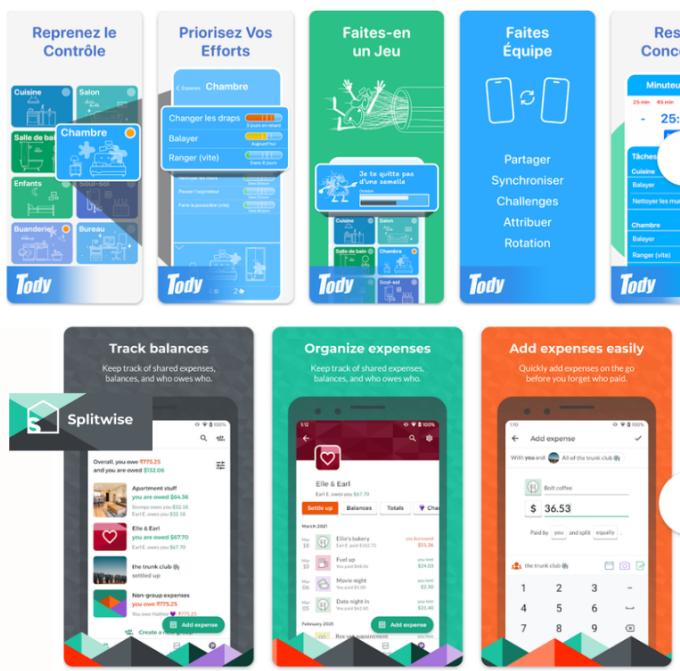


FIGURE 8 – Extrait des applications Tody et Splitwise

3.6 Choix de la méthodologie

Pour mener à bien ce projet, nous avons opté pour une approche Agile avec la méthodologie Scrum. Ce choix s'est imposé naturellement compte tenu de la complexité du projet et de notre configuration d'équipe. La nature itérative de Scrum nous permet d'ajuster régulièrement nos priorités en fonction des retours des membres de l'équipe et des contraintes techniques rencontrées. Cette approche est particulièrement adaptée à notre contexte étudiant car elle offre une grande flexibilité et permet d'intégrer rapidement les apprentissages tout au long du développement. L'un des outils de communication utilisé par notre groupe est un Google doc partagé, un document dont le contenu est le fruit de la rédaction collaborative des membres du groupes, cet outil est très utile pour mener des "Brainstormings" (chacun y liste ses idées avec quelques explications puis les membres du groupe discutent ensemble de la faisabilité et de l'utilité de chaque élément).

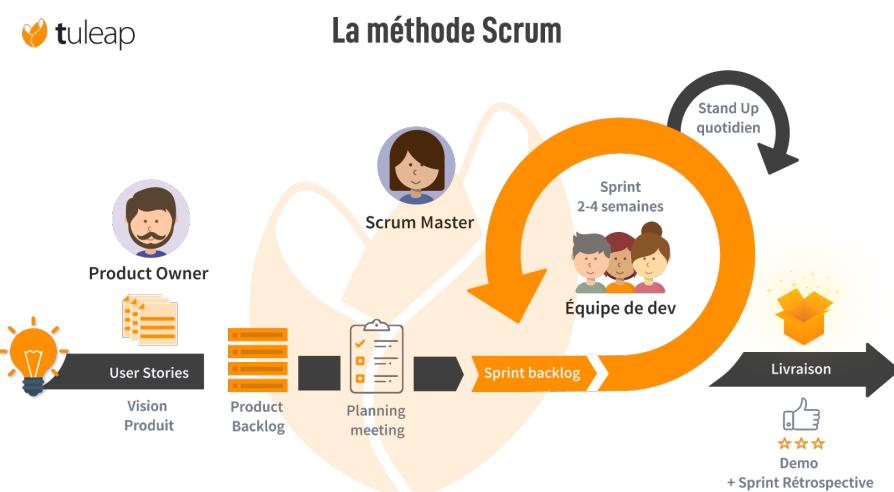


FIGURE 9 – Méthodologie Scrum

3.7 Le choix de la méthode

La méthodologie Scrum est la mieux adaptée au caractère polyvalent et communautatif des membres du groupe, contrairement au cycle en V, trop rigide, ou à la méthode Waterfall, trop linéaire. Scrum offre l'avantage de pouvoir s'adapter aux changements et d'intégrer rapidement les retours de chaque membre, pour une meilleure gestion du projet. Notre implémentation de Scrum comprend des sprints d'environ 2 semaines, avec des revues et commentaires régulières. Cette approche nous permet de maintenir un rythme de développement soutenu tout en assurant la qualité du produit final. L'utilisation d'outils comme Notion, WhatsApp et Google docs pour le suivi des tâches, et GitHub pour le versionnage du code complète parfaitement cette approche.

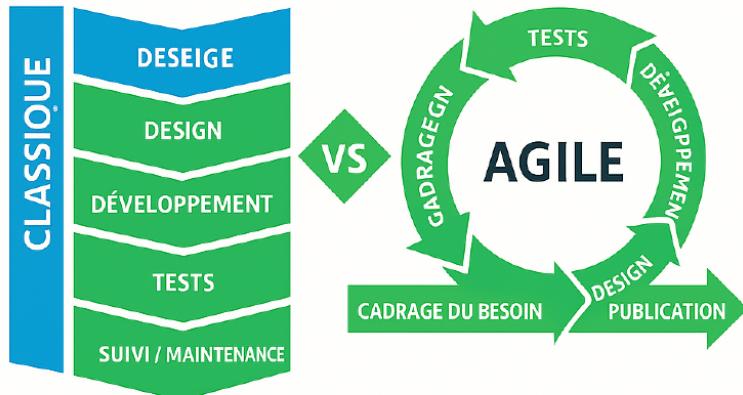


FIGURE 10 – Comparaison entre méthodes classique et agile

4 Mise en place du projet avec SCRUM

4.1 Définition de la méthode Scrum

Pour mener à bien le développement de Keyper, nous avons choisi d'adopter la méthodologie Scrum, une approche agile fondée sur 3 principes : la transparence, l'inspection et l'adaptation. Ces piliers garantissent une organisation rigoureuse tout en laissant la place à la flexibilité nécessaire pour un projet étudiant impliquant plusieurs membres avec des disponibilités variées.

La transparence assure que les informations essentielles au suivi du projet (état d'avancement, obstacles, priorités etc.) sont accessibles et comprises par l'ensemble de l'équipe. L'inspection permet de vérifier régulièrement l'évolution du produit et d'identifier rapidement tout écart par rapport aux objectifs initiaux. Enfin, l'adaptation garantit la capacité de l'équipe à ajuster ses priorités, son organisation ou ses solutions techniques en fonction des imprévus, retours et nouveaux besoins identifiés.

Scrum repose également sur trois rôles clés : le Product Owner, le Scrum Master et l'équipe de Développement. Leur coopération structurée permet de maximiser l'efficacité et d'assurer la progression continue du projet.

4.2 Les rôles dans l'équipe

La répartition des rôles a été définie en tenant compte des compétences, de l'expérience et des préférences de chacun :

- Product Owner (Salekha Fall) : responsable de la vision du produit, de la priorisation du Product Backlog et de la bonne compréhension des besoins utilisateurs. Elle veille à ce que chaque fonctionnalité développée apporte une valeur réelle.
- Scrum Master (Camilia Saba) : garante de l'application du cadre Scrum. Elle facilite les échanges, résout les blocages, encourage la communication et veille à maintenir un environnement de travail collaboratif et productif.
- Développeurs (Jorys Kancel et Youcef Hamissi) : chargés de l'implémentation des fonctionnalités, de l'intégration technique, des tests et des livraisons successives. Leur complémentarité entre compétences frontend et backend constitue un atout essentiel pour le développement de Keyper.

Cette organisation permet une répartition claire des responsabilités et assure une progression régulière malgré les contraintes de disponibilité de chacun.

4.3 Artefacts Scrum

3 artefacts principaux structurent notre organisation :

- Product Backlog : document stocké sur Google Docs et accompagné de maquettes sur Figma. Il regroupe toutes les User Stories fonctionnelles et techniques classées par priorité. Il évolue en continu selon les nouvelles idées, retours internes et les contraintes de faisabilité.
- Sprint Backlog : géré sous forme de tableau Kanban dans notre groupe WhatsApp, avec des colonnes “À faire”, “En cours” et “Terminé”. Ce format très accessible permet des mises à jour instantanées et une vision claire de l'avancement.
- Incrément : ensemble des fonctionnalités terminées et validées à la fin de chaque sprint. Les incréments sont versionnés sur GitHub avec des commits explicites, permettant un suivi précis de l'évolution du code source.

Cette structure garantit une cohérence permanente entre conception, développement et documentation.

4.4 Événements Scrum

Le déroulement du projet est rythmé par quatre événements Scrum essentiels :

Daily Stand-up Effectuée quotidiennement sur WhatsApp, cette courte synchronisation permet de partager les avancées, de signaler les difficultés et de maintenir une bonne coordination malgré des emplois du temps différents.

Sprint Planning À chaque début de sprint, l'équipe sélectionne les User Stories prioritaires, discute des dépendances, estime la charge de travail et fixe un objectif clair. Cette étape garantit une vision commune de ce qui doit être accompli.

Revue de Sprint À la fin de chaque sprint, les développeurs présentent les nouvelles fonctionnalités implémentées. L'équipe échange ensuite sur les améliorations possibles et ajuste le Product Backlog en conséquence.

Rétrospective de Sprint Cette réunion permet d'identifier les points forts du sprint, les difficultés rencontrées et les pistes d'amélioration. Elle contribue à renforcer la cohésion de l'équipe et à optimiser notre organisation au fil du temps.

4.5 Gestion des outils

L'organisation du projet repose sur un ensemble d'outils complémentaires :

- WhatsApp : communication rapide et tableau Kanban simplifié.
- Google Drive : centralisation des documents de travail (brainstormings, textes, comptes rendus, spécifications).
- Figma : création de maquettes et prototypes interactifs facilitant la conception UX.
- GitHub : gestion du code source, versionnement, branches et documentation technique.

Cette combinaison d'outils, simple mais efficace, s'adapte parfaitement à nos contraintes étudiantes et encourage une collaboration fluide.

Sprint	Fonctionnalité	User Story / Tâches Principales	Priorité (Points)	Durée
1	Infrastructure	Mise en place de la base de données, Configuration git, Environnements de développement.	5	5 jours
1	Fondations Backend	Configurer l'environnement Django, Modèles de base (Utilisateur, Foyer, Rôles), Système d'authentification.	8	4 jours
1	Fondations Frontend	Structure du projet React, Pages d'authentification, Layout principal, Thème clair/sombre.	7	5 jours
2	Gestion des Tâches (Cœur)	Création/Édition/Suppression de tâches, Attribution aux membres, Priorités et statuts, Filtres et recherches.	10	5 jours
2	Tableau de Bord	Vue d'ensemble des tâches, Statistiques personnelles, Calendrier des échéances (Affichage des données).	6	5 jours
2	Notifications	Système de notifications en temps réel, Notifications par email.	4	4 jours
3	Chat et Messages	Messagerie instantanée, Notifications de lecture, Gestion des pièces jointes.	8	4 jours
3	Gestion des Dépenses	Suivi des dépenses partagées, Rappels de paiement, Historique des transactions.	7	5 jours
3	Améliorations Foyer/Profil	Profils utilisateurs, Paramètres du foyer, Export de données.	5	5 jours
4	Gamification	Système de points, Défis hebdomadaires, Tableau des scores (Jeu non inclus dans cette tâche).	6	5 jours
4	Optimisations	Tests automatisés, Optimisation des performances (Vitesse, Requêtes), Gestion des erreurs.	8	4 jours
4	Déploiement Final	Documentation utilisateur, Tests utilisateurs finaux, Déploiement en production.	6	5 jours

FIGURE 11 – KeyPer : Organisation des Sprints

4.6 Planification des sprints

Nos sprints ont une durée moyenne de deux semaines, un rythme cohérent au regard de notre disponibilité et du volume de travail. Chaque sprint suit la structure suivante :

- sélection des User Stories pertinentes ;
- découpage en tâches techniques ;
- estimation de la charge de travail ;
- suivi quotidien via WhatsApp ;
- livraison sur GitHub à mesure des complétions ;
- présentation de l'incrément en fin de sprint.

5 Modélisation du système

5.1 Diagramme des cas d'utilisation

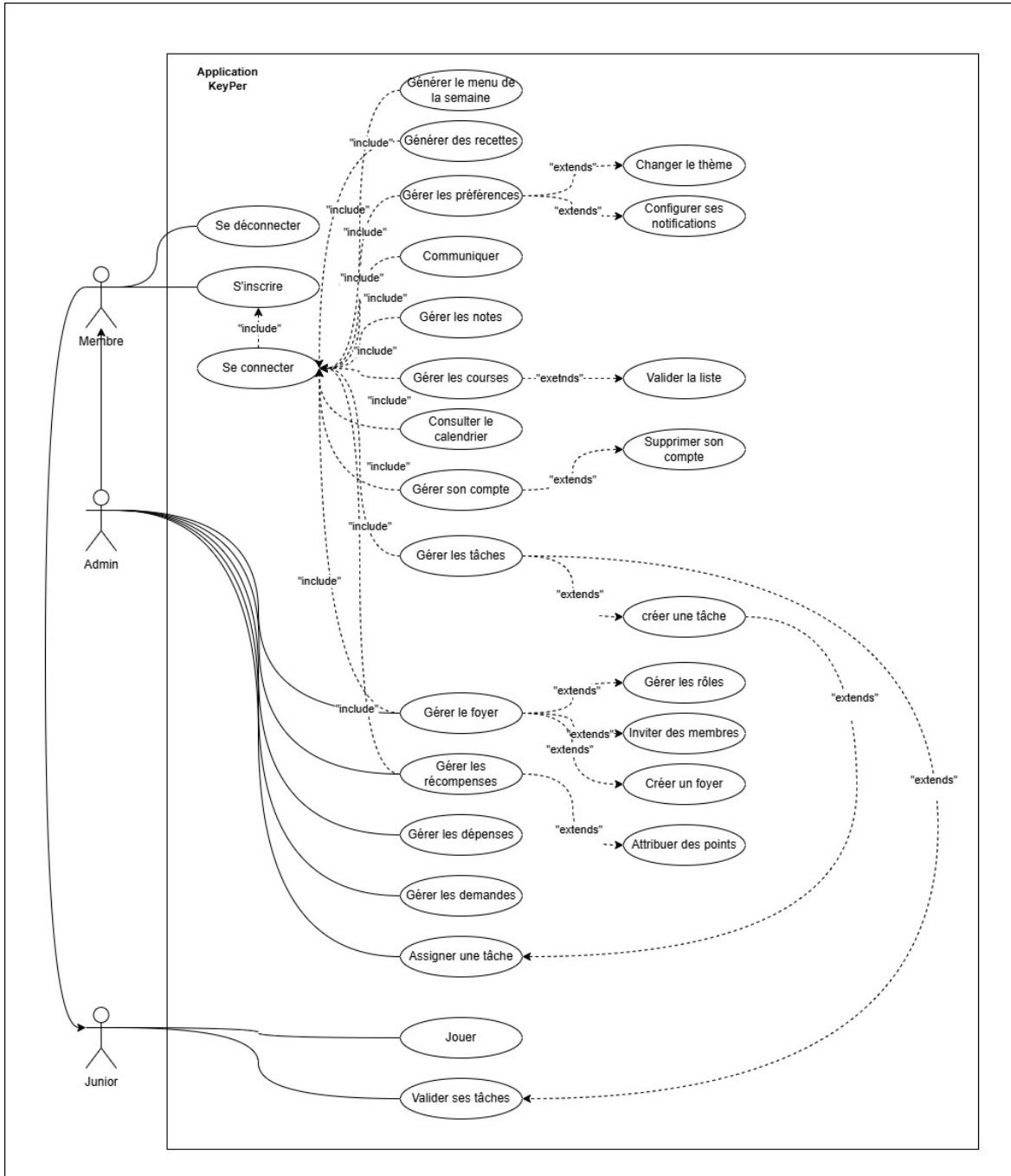


FIGURE 12 – Diagramme USE CASE

L'application Keyper s'articule autour de quatre rôles distincts, chacun disposant de droits d'accès spécifiques. L'administrateur du foyer dispose des droits les plus étendus, incluant la gestion des membres, la configuration des paramètres du foyer et la validation des inscriptions. Les utilisateurs standards bénéficient d'un accès complet aux fonctionnalités de base telles que la gestion des tâches, la communication et la consultation des informations partagées. Le rôle junior, conçu pour les utilisateurs plus

jeunes, offre un accès restreint aux fonctionnalités sensibles, limitant notamment les actions liées à la gestion financière et aux paramètres avancés.

Enfin, l'observateur dispose d'un accès en lecture seule, lui permettant de consulter les informations sans pouvoir les modifier. La gestion des tâches constitue le cœur fonctionnel de l'application. Le système permet la création, l'assignation et le suivi des tâches domestiques, avec des fonctionnalités avancées telles que la définition de priorités, l'ajout de pièces jointes et la mise en place de rappels. Chaque tâche peut être associée à une pièce spécifique du foyer et à un ou plusieurs membres responsables. Le suivi en temps réel de l'état d'avancement et les notifications automatiques facilitent la coordination entre les différents membres du foyer.

La gestion financière intégrée offre une vue d'ensemble des dépenses du foyer. Les utilisateurs peuvent enregistrer leurs dépenses, les catégoriser et les visualiser sous forme de graphiques évolutifs. Le système prend en charge la création de budgets personnalisés par catégorie et génère des rapports périodiques. La répartition équitable des charges entre les membres est simplifiée grâce à des outils de calcul automatique et des rappels pour les paiements récurrents. La communication au sein du foyer est facilitée par plusieurs canaux intégrés. Un système de messagerie instantanée permet les échanges directs entre membres, tandis qu'un tableau d'affichage centralise les annonces importantes. Les commentaires sur les tâches et les notifications en temps réel assurent une coordination efficace entre les différents utilisateurs. Chaque membre peut personnaliser ses préférences de notification pour recevoir les alertes selon ses préférences.

La sécurité des données et la confidentialité sont garanties par un système d'authentification robuste et une gestion fine des autorisations. Les administrateurs peuvent définir précisément les droits d'accès pour chaque rôle, assurant ainsi que les utilisateurs ne voient et ne modifient que les informations qui les concernent. La conformité aux réglementations en vigueur, notamment le RGPD, est assurée par des fonctionnalités de gestion des données personnelles et de suppression des comptes. Cette structure complète et détaillée couvre l'ensemble des interactions possibles au sein de l'application, offrant ainsi une vision claire des fonctionnalités disponibles pour chaque type d'utilisateur.

5.2 Modèle des données

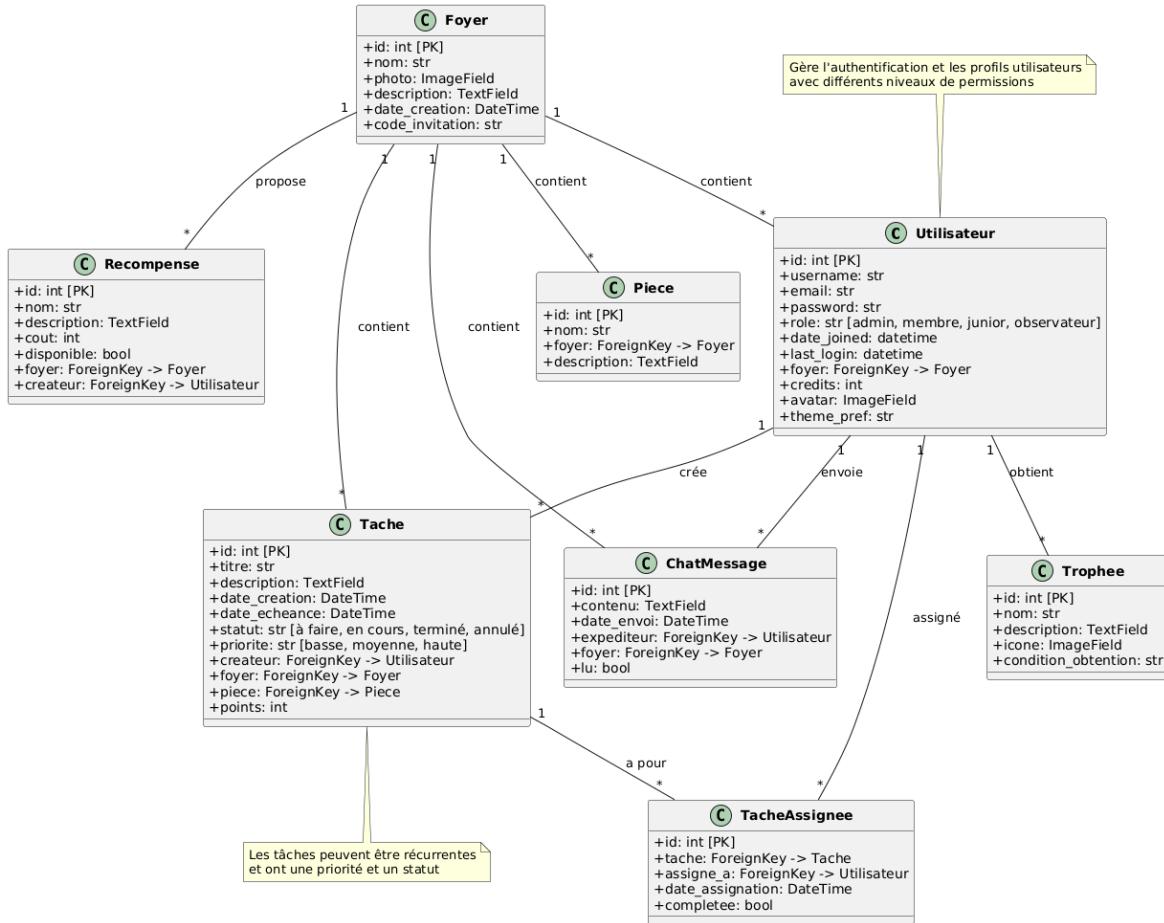


FIGURE 13 – Diagramme de classes de Keyper

Le modèle de données de Keyper repose sur une structure relationnelle robuste, conçue pour refléter précisément les interactions entre les utilisateurs et leur environnement domestique. La base de données est organisée autour de plusieurs entités principales qui interagissent pour offrir une expérience utilisateur fluide et cohérente. Les utilisateurs, au cœur du système, sont liés aux foyers par une relation plusieurs-à-plusieurs, permettant à une même personne de faire partie de plusieurs espaces domestiques avec des rôles différents. Chaque foyer contient ses propres tâches, dépenses et messages, créant ainsi des espaces de travail distincts et sécurisés. Le système gère finement les droits d'accès en fonction des rôles (administrateur, utilisateur, junior, observateur), garantissant que chaque membre ne voit et ne modifie que les informations qui le concernent.

Les tâches, élément central de l'application, sont conçues pour être flexibles et complètes. Chacune peut être assignée à un ou plusieurs utilisateurs, classée par priorité et associée à des dates butoirs. Le suivi en temps réel de leur avancement permet une coordination efficace entre les membres du foyer. Les commentaires et pièces jointes associés à chaque tâche facilitent la communication et le partage d'informations. La gestion financière s'appuie sur un modèle de dépenses détaillé, permettant d'enregistrer chaque transaction avec sa catégorie, son montant et sa preuve. Le système calcule automatiquement les répartitions entre les membres et génère des rapports personna-

lisés. Cette approche offre une transparence complète sur les finances du foyer tout en simplifiant la gestion budgétaire.

La communication entre les membres est assurée par un système de messagerie intégré, complété par des notifications en temps réel. Chaque message est lié à un foyer et peut faire l'objet de réponses, créant ainsi des fils de discussion organisés. Les notifications informent les utilisateurs des nouvelles activités les concernant, favorisant une réactivité optimale. La sécurité des données est une priorité absolue. Le chiffrement des informations sensibles, la journalisation des actions importantes et les sauvegardes régulières garantissent l'intégrité et la confidentialité des données. Le modèle respecte les normes de sécurité les plus strictes, notamment en matière de protection des données personnelles.

L'architecture de la base de données a été optimisée pour des performances optimales, avec une indexation stratégique des champs fréquemment interrogés. La structure modulaire permet des évolutions futures sans remettre en cause l'intégrité des données existantes, assurant ainsi la pérennité de la solution. Cette conception réfléchie et robuste fait de Keyper une application fiable et évolutive, capable de s'adapter aux besoins spécifiques de chaque foyer tout en maintenant une expérience utilisateur intuitive et réactive.

5.3 Maquettes des interfaces

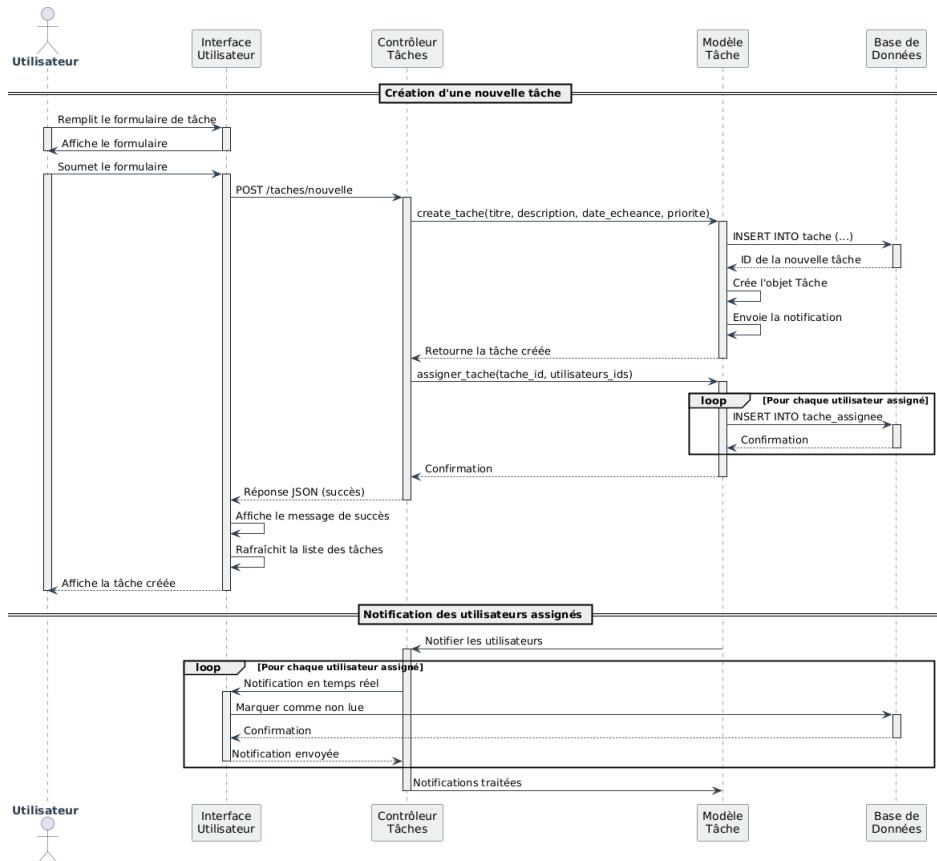


FIGURE 14 – Diagramme de séquence

L'interface de Keyper est conçue pour offrir une expérience utilisateur fluide et intuitive, tout en s'adaptant parfaitement aux différents appareils. La navigation prin-

cipale s'articule autour d'un menu latéral fixe, donnant un accès rapide aux différentes sections de l'application. Le tableau de bord principal offre une vue d'ensemble des tâches en cours, des dépenses récentes et des activités du foyer, avec des indicateurs visuels pour les priorités et les échéances. La gestion des tâches se fait via une interface épurée, permettant de créer, modifier et suivre l'avancement des tâches en quelques clics. Les utilisateurs peuvent filtrer les tâches par statut, priorité ou pièce, et bénéficient d'un système de glisser-déposer pour les réorganiser facilement. Les notifications en temps réel informent des mises à jour importantes, comme une tâche assignée ou un commentaire ajouté.

La messagerie intégrée offre une communication fluide entre les membres du foyer, avec la possibilité d'ajouter des pièces jointes et de réagir aux messages. Les conversations sont organisées par sujets, facilitant le suivi des discussions. Les notifications discrètes en haut de l'écran informent des nouveaux messages sans perturber l'expérience utilisateur. La gestion financière est présentée sous forme de tableaux de bord interactifs, avec des graphiques évolutifs pour visualiser les dépenses par catégorie et par période.

Les utilisateurs peuvent ajouter des dépenses, avec une reconnaissance automatique des catégories et la possibilité d'ajouter des justificatifs photos. Les rappels de paiement et les alertes de budget dépassé aident à garder le contrôle des finances du foyer. Le profil utilisateur est personnalisable, avec la possibilité d'ajouter une photo, de définir des préférences de notification et de gérer les paramètres de confidentialité. L'application propose également un mode sombre pour un confort visuel optimal en toutes circonstances.

6 Conclusion

Le développement de Keyper représente pour nous l'opportunité d'appliquer nos connaissances théoriques à un projet concret, une application web complète. Notre approche basée sur Scrum nous a permis de gérer efficacement la complexité du projet tout en maintenant une grande flexibilité. Les défis techniques rencontrés, notamment dans la gestion des rôles utilisateurs et la synchronisation des données en temps réel, ont été surmontés grâce à une collaboration bien organisée et une communication continue au sein des membres du groupe. Ce projet nous a non seulement permis de développer nos compétences techniques en développement web, mais aussi d'acquérir une expérience précieuse en gestion de projet et en travail d'équipe.