

Redon BRAHIMI
Gabriel SCHOENFELDER
Abdessalam KHATTRI
Kilian NAULEAU
Anas KIOUAZ
Yanis DEBBAL
Rayan NYABI

Cadrage et organisation du projet Weave

Tables des matières :

| | |
|--|-----------|
| 1. Cadrage | 3 |
| 1.1 Contexte | 3 |
| 1.2 Objectifs du projet | 3 |
| 1.3 Fonctionnalités principales prévues | 3 |
| 1.4 Organisation du projet | 4 |
| 1.5 Contraintes | 4 |
| 1.6 Parties prenantes | 4 |
| 1.7 Personas | 4 |
| 1.7.1 Jeanne | 4 |
| 1.7.2 Thomas | 5 |
| 1.7.3 Laurie | 6 |
| 1.8 Risques | 6 |
| 1.9 Pitch / Présentation du produit | 7 |
| 2. Besoins | 7 |
| 2.1 Objectif du recueil de besoins | 7 |
| 2.2 Méthodologie de recueil | 7 |
| 2.3 Besoins fonctionnels | 7 |
| 2.4 Services & Processus métier | 8 |
| 2.4.1 Processus de planification d'une demande | 8 |
| 2.4.2 Processus de gestion d'incident | 8 |
| 2.6 Synthèse des besoins prioritaires | 9 |
| 3. Réalisation | 10 |
| 3.1 Faisabilité technique | 10 |
| Base de données (SQL) | 10 |
| Backend (API REST / API sécurisée) | 10 |
| Frontend (Web et Mobile) | 10 |
| Notifications (Push / Email) | 11 |
| Sécurité des données | 11 |
| 3.2 Faisabilité opérationnelle | 12 |
| 3.3 Faisabilité légale et réglementaire | 12 |
| 3.4 Veille concurrentielle | 12 |
| 3.5 Organisation du travail | 13 |
| Répartition du temps | 13 |
| Répartition des rôles | 13 |
| 4. Conclusion | 14 |
| 5. Annexes | 15 |

1. Cadrage

1.1 Contexte

Face au vieillissement de la population et à la perte d'autonomie associée, l'organisation du soutien quotidien (courses, lien social, etc.) est complexe, les outils de coordination efficaces manquant.

Nous proposons une application web collaborative d'emploi du temps solidaire inspirée des *Voisins Solidaires*, mais avec une dimension organisationnelle structurante. Cette plateforme permettra aux proches, voisins et bénévoles de s'inscrire sur des créneaux pour les tâches d'aide, favorisant l'entraide locale et le maintien à domicile de qualité.

L'objectif est de fluidifier la coordination de l'aide et de renforcer le lien social, via une application simple, accessible et sécurisée.

Ce rapport détaille la phase de cadrage du projet, incluant la définition des besoins, l'analyse des contraintes, des risques et l'étude de faisabilité technique. Ce travail d'équipe établit les fondations d'une solution utile et pérenne.

1.2 Objectifs du projet

- Notre objectif est de concevoir une application mobile intuitive destinée à un public peu familiarisé avec les outils numériques (personnes âgées ou jeunes). Cette application vise à soutenir les personnes âgées en situation de perte d'autonomie, nécessitant une assistance régulière pour les activités de la vie quotidienne, ainsi que leurs aidants proches (famille, amis, voisins).
- L'application permettra aux personnes concernées d'améliorer l'organisation de leurs activités, de fluidifier la communication et de prévenir les malentendus ou les problèmes de coordination. Nous aspirons à rendre la vie des bénéficiaires et des aidants plus paisible, sereine et prévisible.

1.3 Fonctionnalités principales prévues

Les fonctionnalités clés de notre projet sont les suivantes :

1. **Calendrier Hebdomadaire Central** : Notre fonctionnalité principale et la plus essentielle est le calendrier hebdomadaire. Il permet aux aidants d'inscrire des activités pour la personne âgée. Les aidants peuvent ensuite s'inscrire à ces activités pour partager un moment avec leur proche et, simultanément, s'assurer de son bien-être.
2. **Messagerie Sécurisée entre Aidants** : Pour éviter les désagréments liés à l'utilisation de groupes externes (comme WhatsApp), nous intégrerons une messagerie dédiée et sécurisée entre les aidants.
3. **Journal de Bord Moral (Souvenirs)** : Ce journal a pour objectif de conserver les souvenirs liés aux activités planifiées dans le calendrier. Par exemple, une photo prise lors d'une "balade au parc" pourra y être ajoutée. Le journal de bord sera téléchargeable pour un partage facilité.

1.4 Organisation du projet

Planification et Organisation du Projet Weave

Phases du Projet :

- **Conception** : La période de conception s'étend jusqu'en décembre.

- **Développement :** La phase de codage de l'application est prévue pour les trois premières semaines de janvier, succédant immédiatement à la période de conception.

Répartition des Rôles :

- **Chef de projet :** Yanis Debbal
- **Responsable Conception :** Abdessalam Khattri
- **Responsable Anglais :** Redon Brahimi
- **Responsable Communication :** Rayan Nyabi
- **Responsable Conception :** Kilian Nauleau
- **Responsable Réalisation :** Gabriel Schoenfelder
- **Responsable Réalisation :** Anas Kiouaz

1.5 Contraintes

- - Temporelle : Retard dans le développement, charge de travail dans des délais courts.
- - Qualité : Rendre un travail soigné et de qualité et correspondant aux critères demandés.
- - Économique : Hébergement potentiellement coûteux, ressources limitées techniquement.
- - Humaine : Manque de communication dans l'équipe, fatigue car projets longs.
- - Technique : Difficulté à synchroniser les tâches en temps réel / Fuite de données personnelles / Compatibilité mobile-web

1.6 Parties prenantes

- **Personnes âgées dépendantes** : Il s'agit des individus ayant des difficultés à s'autogérer complètement et qui ont ponctuellement besoin d'aide pour des tâches essentielles comme la prise de médicaments, les déplacements (rendez-vous), ou les tâches ménagères (lessive, etc.), nécessitant ainsi un soutien extérieur.
- **Aidants (familiaux, proches)** : Ils englobent les membres de la famille, les voisins ou les amis de la personne aidée. Leur motivation est soit de fournir une assistance concrète à un proche, soit simplement de passer du temps de qualité avec celui-ci.
- **Bénévoles** : Bien que plus rares, ces individus offrent gracieusement leur temps pour aider ou passer du temps avec la personne âgée, sans contrepartie financière.

1.7 Personas

Voici différents "personas" pour rajouter du contexte à notre projet, ce sont des personnages fictifs (avatars) pour représenter les clients cibles que nous pouvons potentiellement atteindre avec notre application.

**** Les photos des personas sont fictives, venant du site <https://thispersondoesnotexist.com> ****



1.7.1 Jeanne

Profil :

- **Âge :** 78 ans
- **Situation :** Veuve, vit seule depuis le décès de son mari, elle vit dans un quartier de maisons calmes depuis 40 ans, avec quelques voisins.
- **Profession :** Retraitée, ancienne fleuriste.

Biographie :

Jeanne est une femme atteinte de la maladie de l'Alzheimer (maladie neuro-évolutive qui fait perdre la dépendance d'une personne au fil du temps). Elle vit seule depuis le décès de son mari et son état d'autonomie est entre GIR2 et GIR3, donc elle a besoin d'aide pour une majorité des soins corporels et une partie des tâches quotidiennes. Cependant, elle néglige son bien-être pensant être un fardeau pour son fils Thomas. En effet, son fils vient l'accompagner tous les jours pour les soins principaux mais Jeanne n'évoque pas son besoin social.

Frustrations :

- **La solitude :** Jeanne se sent souvent seule car elle pense être un poids pour sa famille.
- **La confusion :** Jeanne oublie souvent ses médicaments/activités, cela crée de l'anxiété.
- **La peur :** peur de tomber et de ne pas se relever car elle est seule.
- **Inquiétude :** ne pas déranger son fils Thomas.

Objectifs :

- Ne pas inquiéter Thomas.
- Avoir une assistance pour ses nécessités, se sentir moins vulnérable.
- Se sentir un peu moins seule.



1.7.2 Thomas

Profil :

- **Âge :** 26 ans
- **Situation :** Célibataire, vit seul dans un village proche à celui de sa mère Jeanne, afin d'être à proximité pour l'aider.
- **Profession :** étudiant en thèse de mathématiques.

Biographie :

Thomas est le fils unique de Jeanne. Il est très proche d'elle, il gère seul la situation depuis le diagnostic de l'Alzheimer. Il est pris entre ses études exigeantes, sa vie sociale, et il s'inquiète beaucoup pour sa mère. Il lui rend visite tous les jours après les cours et gère ses courses, ses rendez-vous médicaux et ses papiers. La charge mentale est énorme et il se sent souvent dépassé et coupable de ne pas être là plus de temps. Bien sûr, il aimeraient être aidé afin d'avoir du temps pour lui aussi.

Frustrations :

- **L'inquiétude :** Ne pas savoir ce qui se passe entre ses visites. (Exemple : "A-t-elle mangé ce midi ?", "A-t-elle pris ses pilules ?").
- **La coordination :** Communiquer les mêmes informations plusieurs fois à différentes personnes (la voisine Laurie, l'infirmière...).
- **Les fausses alertes :** S'inquiéter parce qu'elle ne répond pas au téléphone, alors qu'elle fait juste la sieste.

Objectifs :

- Ne pas s'inquiéter pour sa mère, être serein afin d'étudier tranquillement.
- Avoir un peu plus de personnes pour l'aider.
- Pouvoir réunir les informations avec les personnes aidant sa mère.



1.7.3 Laurie

Profil :

- **Âge :** 35 ans
- **Situation :** Mariée, vit avec son mari et sa fille de 7 ans, elle est dans la maison voisine de Jeanne. Elle est consciente de la situation de Jeanne et aimerait aider au minimum.
- **Profession :** professeur des écoles.

Biographie :

Laurie connaît Jeanne depuis son arrivée dans le quartier il y a 5 ans. Elle l'apprécie beaucoup de part sa bienveillance et voit les effets de la maladie progresser. Elle est de bonne volonté, mais sa vie de famille et son travail lui prennent déjà beaucoup de temps. Elle veut aider sans devenir une aidante à temps plein.

Frustrations :

- **L'incertitude :** Hésiter à prévenir Thomas par peur de le déranger au travail pour une fausse alerte.
- **Le manque d'info :** Ne pas savoir si Jeanne est là ou si Thomas l'a emmenée en rendez-vous.
- **La responsabilité :** Avoir peur de rater un signe important alors qu'elle était juste à côté.

Objectifs :

- Un moyen rapide et simple de signaler un problème ou une observation (ex: "J'ai vu Jeanne dans le jardin, tout va bien") sans devoir passer un appel téléphonique.
- Avoir accès à un planning simplifié (savoir si quelqu'un doit passer aujourd'hui).
- Des consignes claires de la part de Thomas sur ce qu'il attend d'elle.
- Être rassurée sur le fait que son aide est utile mais limitée (pas de charge médicale).

1.8 Risques

*[Voir annexes](#) pour les [tableaux de criticités](#) et [de mesures](#) pour évaluations.

Tableau d'évaluation et de gestion :

| Risque | Type | Criticité | Mesures et Gestion* |
|---|-------------------------|-----------|--|
| Retard dans le respect des deadlines IUT | Organisationnel / Temps | forte | Mesure corrective : Planification précise (Trello / GitLab), chef de projet responsable du suivi. Priorisation d'un MVP fonctionnel |
| Mauvaise communication | Humain | faible | Acceptation : Réunions régulières, utilisation de Discord / GitLab issues, rôles bien définis. |
| Absence d'un membre clé (maladie) | Humain | moyenne | Polyvalence des membres, partage des connaissances, binômes. |
| Données sensibles (médicales) mal | Sécurité / Légal | Moyenne | Respect du RGPD, pas de stockage de données réelles, chiffrement minimal |

| Risque | Type | Criticité | Mesures et Gestion* |
|---|------------------|-----------|---|
| protégées | | | |
| Difficultés tech(hébergement, db, framework) | Technique | Moyenne | Recherche précoce de solutions techniques, tests dès la phase 2 |
| Application trop complexe pour les aidants âgés | Fonctionnel / UX | Forte | Tests utilisateurs simulés, maquettes simples, ergonomie prioritaire. |

1.9 Pitch / Présentation du produit

Weave est une plateforme web d'entraide locale pensée pour faciliter l'organisation du soutien autour des personnes en perte d'autonomie.

Aujourd'hui, les familles, voisins et bénévoles veulent aider, mais manquent d'outils simples pour **se coordonner**, répartir les tâches et assurer un suivi fiable du quotidien : courses, visites, appels, accompagnements, vérification du moral, etc.

Weave apporte une solution concrète :

une application collaborative qui permet à l'entourage d'une personne dépendante de **s'inscrire sur des créneaux d'aide**, de **visualiser le planning**, de **communiquer efficacement** et de **suivre l'état moral et les besoins** du bénéficiaire.

Grâce à une interface intuitive, un système de notifications et un planning partagé, Weave garantit :

- une meilleure répartition des actions au sein de l'entourage ;
- une coordination fluide et transparente ;
- plus de sécurité et de suivi pour la personne aidée ;
- moins de charge mentale pour les aidants.

Weave ne se limite pas à organiser l'aide :

c'est un outil qui **renforce le lien social**, encourage la solidarité locale et contribue à un **maintien à domicile plus serein**, plus humain et mieux encadré.

Weave, c'est **la simplicité d'un planning partagé**, la **force d'une communauté solidaire**, et la **tranquillité d'esprit pour les familles**.

2. Besoins

2.1 Objectif du recueil de besoins

Cette étape vise à recueillir les attentes des futurs utilisateurs et à les traduire en fonctionnalités concrètes. L'application s'adresse à plusieurs types d'acteurs : les aidants familiaux, les personnes âgées bénéficiaires et les professionnels de santé impliqués dans leur suivi. L'objectif du recueil des besoins est donc de déterminer comment concevoir un outil réellement utile, simple d'usage et cohérent avec les contraintes de chacun.

2.2 Méthodologie de recueil

Le recueil des besoins repose sur plusieurs approches complémentaires. Des entretiens fictifs inspirés de situations courantes ont d'abord permis d'identifier les attentes principales : coordination des tâches, visibilité des actions et communication facilitée. Un benchmark a ensuite été réalisé pour comparer plusieurs plateformes existantes. *Voisins Solidaires* met en avant l'entraide de proximité mais reste limité dans le suivi individuel ; *Doctolib* se distingue par son interface clair et efficace, mais s'adresse avant tout aux professionnels ; *Mon Espace Santé* assure la confidentialité des données médicales, au prix d'une ergonomie complexe. Enfin, une analyse documentaire basée sur des rapports de l'*INSEE* et d'associations d'aidants a permis de confirmer la nécessité d'un outil numérique collaboratif, accessible à tous les profils.

Cette démarche a permis d'aboutir à une vision claire des besoins fonctionnels et non fonctionnels à intégrer à la solution.

2.3 Besoins fonctionnels

L'application devra permettre à chaque utilisateur de créer un compte personnel, de rejoindre un groupe d'aidants et de collaborer autour d'une personne âgée.

Un calendrier partagé constitue le cœur du système : il permet de planifier les tâches, répartir les rôles et recevoir des rappels automatiques.

Une messagerie intégrée facilitera la communication entre les membres du groupe, tandis qu'un journal de bord permettra de suivre l'état moral de la personne aidée et d'assurer un suivi global.

La sécurité des données sera primordiale, avec un accès limité aux membres autorisés et une conformité stricte au RGPD.

2.4 Services & Processus métier

La modélisation de nos processus métier ne se limite pas à une représentation technique ; elle incarne la promesse de sécurité et de fiabilité de l'application Weave. Cette section détaille le fonctionnement logique des deux piliers de notre solution : la coordination de l'aide au quotidien et la réaction face aux situations d'urgence.

2.4.1 Processus de planification d'une demande

Le cœur de l'application Weave réside dans sa capacité à transformer un besoin exprimé par la famille en une action concrète réalisée par un bénévole/aidant, sans générer de charge mentale supplémentaire pour les aidants familiaux. Le diagramme BPMN modélisant ce flux a été conçu pour gérer l'incertitude liée au bénévolat (disponibilité, oubli) tout en garantissant que chaque besoin trouve une réponse.

Lancement et Recherche Collective

Tout commence par l'Aidant Initiateur (généralement un proche) qui identifie un besoin précis, comme des courses ou un accompagnement médical. Une fois la demande qualifiée dans l'application, le processus déclenche une phase de recherche active. Contrairement à une approche séquentielle où l'on contacterait les gens un par un, la demande est diffusée simultanément à l'ensemble du groupe de volontaires. Cette phase de "consultation" est gérée de manière autonome : chaque volontaire reçoit la notification et dispose d'un temps imparti pour répondre. Pour éviter les blocages, le système intègre une gestion tacite du refus : si un volontaire ne répond pas dans le délai imparti, il est considéré comme indisponible sans avoir besoin de se justifier, ce qui préserve l'engagement sur le long terme.

Interruption Intelligente et Quota

Un point crucial de notre logique est la gestion des ressources humaines. Si une activité nécessite deux personnes, il est inutile que cinq bénévoles s'engagent. Nous avons donc modélisé un mécanisme d'interruption automatique basé sur un quota : dès que le nombre de volontaires requis (par exemple, un seul chauffeur) s'est engagé, la recherche est immédiatement stoppée pour les autres membres du groupe. Cela évite les frustrations et optimise la sollicitation de la communauté.

Sécurisation de l'Engagement (La double confirmation)

L'expérience montre qu'un engagement pris plusieurs jours à l'avance peut être oublié. Pour pallier ce risque opérationnel, le processus intègre une étape de sécurité indispensable : la confirmation de présence. Un délai avant l'activité (paramétrable, par exemple 2 heures avant), le système sollicite à nouveau le volontaire engagé. Ce dernier doit confirmer qu'il est toujours disponible. Si le volontaire confirme, l'activité est validée. En revanche, si un imprévu survient ou si le volontaire ne répond pas, le système ne laisse pas la personne âgée dans l'attente : il détecte l'absence de confirmation comme un désistement et peut, selon la configuration, relancer une recherche d'urgence ou alerter la famille. Une fois l'activité réalisée, le processus se clôture par une mise à jour de l'historique, assurant la traçabilité des interventions.

2.4.2 Processus de gestion d'incident

Le second processus, sans doute le plus critique, concerne la sécurité de la personne aidée. Il définit comment l'application réagit lorsqu'un bénévole, présent sur place, détecte une anomalie (chute, confusion, problème domestique). Notre modélisation a été pensée pour couvrir tous les scénarios, du plus bénin au plus grave, en garantissant qu'aucune alerte ne reste sans réponse.

Le Tri Initial et la Priorité Vitale

Dès la détection d'une anomalie, le processus impose un choix binaire strict au bénévole. Si la situation relève d'une urgence vitale, l'application s'efface au profit des procédures de secours classiques : le bénévole doit appeler le 15 (SAMU) immédiatement. L'application ne sert alors qu'à notifier la prise en charge. Si l'urgence n'est pas vitale, le bénévole effectue un signalement via l'application. À cet instant, il entre dans un état d'attente active : il reste auprès de la personne aidée tant qu'il n'a pas reçu le "feu vert" (un signal de libération) indiquant que la situation est prise en main par un tiers.

La Chaîne de Solidarité (Escalade Automatisée)

Une fois le signalement envoyé, la responsabilité bascule vers l'Aidant Principal. C'est ici que notre système de sécurité prend tout son sens. Nous ne pouvons pas supposer que l'aidant principal est disponible à tout moment (il peut travailler, dormir, etc.). C'est pourquoi nous avons intégré un mécanisme de temporisation (Timer). Si l'Aidant Principal traite l'alerte rapidement, il envoie un signal de résolution qui libère le bénévole sur place. Cependant, si l'Aidant Principal ne réagit pas sous un délai critique (par exemple 1 heure), le processus déclenche automatiquement une escalade. L'alerte est transférée à une personne de confiance préalablement identifiée. Le système vérifie la disponibilité de cette personne. Si elle accepte d'intervenir, elle prend le relais et libère le bénévole initial.

L'Ultime Recours

Enfin, notre modélisation couvre le cas du "pire scénario" : si l'Aidant Principal est absent et que la personne de confiance n'est pas disponible. Dans cette impasse, pour ne jamais laisser une personne vulnérable en danger potentiel, le système déclenche une procédure d'appel ultime aux services de secours ou de police pour une levée de doute. Peu importe qui résout finalement l'incident (l'Aidant, la Personne de confiance ou les Secours), l'action finale génère systématiquement un signal technique unique : "Incident Clôturé". Ce signal est capté instantanément par l'interface du bénévole sur place, l'informant qu'il a rempli sa mission et qu'il peut quitter les lieux en toute sérénité. Cette boucle fermée garantit une sécurité maximale pour le bénéficiaire et une clarté totale pour les intervenants.

2.5 Besoins non fonctionnels

Les besoins non fonctionnels se concentrent sur trois dimensions essentielles : la sécurité, l'ergonomie et la conformité réglementaire.

L'application doit d'abord garantir un haut niveau de **sécurité**, notamment en protégeant les données personnelles sensibles. Les échanges doivent être chiffrés, les accès strictement contrôlés, et l'authentification suffisamment robuste pour éviter toute intrusion non autorisée.

L'**ergonomie** représente également un enjeu majeur, l'outil étant destiné à des aidants et potentiellement à des personnes âgées. L'interface devra être simple, lisible, intuitive et adaptée à un public peu familier des outils numériques.

Enfin, l'application devra répondre aux exigences de la **conformité réglementaire**, en particulier celles du RGPD. Cela inclut l'information des utilisateurs sur le traitement des données, la gestion des consentements, ainsi que la possibilité d'exercer leurs droits (accès, rectification, suppression).

2.6 Synthèse des besoins prioritaires

| Priorité | Besoin | Type |
|----------|--|--------------------|
| Forte | Calendrier partagé et gestion des tâches | Fonctionnel |
| Forte | Notifications automatiques | Fonctionnel |
| Moyenne | Messagerie interne sécurisée | Fonctionnel |
| Moyenne | Journal de bord moral | Fonctionnel |
| Faible | Statistiques globales et personnalisation | Optionnel |

3. Réalisation

3.1 Faisabilité technique

Base de données (SQL)

SQLite n'est pas adapté ici car il ne gère pas efficacement la concurrence et les accès simultanés par plusieurs utilisateurs. Une base de données relationnelle de type **PostgreSQL (avec Supabase)** est envisagée pour sa robustesse, sa scalabilité et sa capacité à gérer des données fortement structurées.

Elle permettra notamment de stocker de manière fiable :

- les **profils utilisateurs** (aidants, seniors, familles),
- les **besoins médico-sociaux**, dont la **grille AGGIR simplifiée**,
- les **disponibilités, compétences, antécédents**,
- l'historique des **interventions, tâches et interactions**.

Critères pour la base de données :

- **Sécurité** : support natif du chiffrement et des permissions fines
- **Compatibilité** : facile à intégrer avec Node.js, FastAPI ou Symfony.
- **Apprentissage facilité** : PostgreSQL a déjà été étudié en cours → temps d'apprentissage **null**

Backend (API REST / API sécurisée)

Le backend expose une **API REST sécurisée** qui centralise l'ensemble de la logique métier de la plateforme. Il prend en charge :

- l'**authentification** et la gestion des permissions,
- le **gestion des profils** et des besoins (GIR, dépendances),
- l'**appariement** aidant ⇔ senior selon compétences et disponibilités,
- la gestion des **notifications**,
- la **communication sécurisée** avec la base PostgreSQL

Critères pour le backend :

- **Communauté et documentation** : large documentation et tutoriels.
- **Sécurité** : support JWT, OAuth2, contrôle fin des rôles et permissions.
- **Phase d'apprentissage facilitée** :
 - **Node.js + Express** : déjà étudié partiellement → **temps d'apprentissage modéré**.
 - **FastAPI** : syntaxe simple → **temps d'apprentissage modéré**.
 - **Symfony (PHP)** : complet donc long à prendre en main → **temps d'apprentissage important** (même si vue en partie en cours)

Fonctions clés gérées : authentification, gestion des profils et besoins, appariement aidant ⇔ senior, notifications, communication sécurisée avec PostgreSQL.

Frontend (Web et Mobile)

Critères pour le frontend :

- **Ergonomie et accessibilité** : interface intuitive pour seniors et aidants.
- **Réactivité et performance** : temps de chargement faible (React, Next.js, Vue.js).
- **Modularité et réutilisabilité** : composants modulaires pour faciliter la maintenance.
- **Compatibilité multi-plateforme** : responsive design, adaptabilité mobile et web.
- **Phase d'apprentissage** :
 - **JavaScript** : déjà étudié. (en partie) → temps d'apprentissage **négligeable**.
 - **React / Next.js / Vue.js** : syntaxe plus avancée → temps d'apprentissage **modéré**.
 - **PHP** (stack classique) : déjà étudié . → temps d'apprentissage **négligeable**.

Le frontend constitue le principal point de contact pour les aidants, les seniors et les familles.

- l'inscription et la gestion de profil,
- la consultation des disponibilités,
- un filtrage avancé (compétences, distance, niveau AGGIR),
- le suivi des interventions,
- un module de messagerie ou de chat sécurisé.

Deux approches technologiques sont possibles :

- **Stack classique** : HTML / CSS / PHP.
- **Stack moderne** : Next.js, React, Angular ou Vue.js.

Notifications (Push / Email)

Critères de choix :

- **Fiabilité** : les messages doivent toujours être envoyés (Firebase, SendGrid, Mailjet).
- **Rapidité** : notification instantanée pour les urgences ou messages.
- **Simplicité d'intégration** : compatible avec les frameworks choisis.
- **Phase d'apprentissage** : Firebase / SendGrid / Mailjet → temps d'apprentissage **négligeable**.

La plateforme doit être capable d'informer rapidement les utilisateurs :

- nouveaux messages,
- demandes d'aide,
- rappels d'intervention,
- suivi des tâches.

Des services spécialisés comme **Firebase Cloud Messaging**, **SendGrid** ou **Mailjet** assurent une diffusion fiable et scalable.

Sécurité des données

La plateforme traitant des **données sensibles**, incluant :

- les niveaux de dépendance (GIR),
- des besoins médicaux,
- des informations sociales confidentielles,

Les données traitées étant sensibles, la sécurité constitue un impératif majeur.
Elle repose sur :

- **Confidentialité** : chiffrement HTTPS et stockage sécurisé.
- **Anonymisation et minimisation** : limiter les données personnelles stockées.
- **authentification sécurisée** (JWT, OAuth2),
- **gestion stricte des rôles** (aidant, senior, famille, administration),
- **conformité RGPD** : consentement, droit à l'oubli, consentement explicite, traçabilité.

- protection contre les attaques courantes (XSS, CSRF, injections SQL). (en fonction du temps restant)

La faisabilité est bonne grâce à des technologies solides, mais la sécurité dépend fortement de la coordination autour du membre spécialisé réseau : yanis (parcours B).

3.2 Faisabilité opérationnelle

La faisabilité opérationnelle dépend principalement de la capacité à gérer et structurer la communauté d'utilisateurs.

- **Points Forts :**
 - **Centralisation** : agenda partagé et mutualisation des besoins médicaux facilitant la coordination.
 - **Accessibilité** : la grille AGGIR simplifiée réduit la complexité et facilite l'entrée dans le service.
- **Risques et Défis Majeurs :**
 - **Effet de réseau** : nécessité d'attirer simultanément aidants et familles (zone géographique pilote recommandée).
 - **Confiance et sécurité** : nécessité de procédures de vérification (identité, antécédents, charte).
 - **Modération** : besoin futur d'une équipe pour gérer litiges et comportements abusifs.

3.3 Faisabilité légale et réglementaire

Ce volet présente des obstacles significatifs qui doivent être traités avant tout développement.

- **Responsabilité :**
 - Que se passe-t-il en cas d'accident, de négligence, de vol ou d'abus de la part d'un bénévole ?
 - En cas d'accident ou de négligence, les limites de responsabilité doivent être clairement définies.
 - La plateforme doit se positionner comme **intermédiaire de mise en relation**, via des **CGU précises**.
- **Protection des Données (RGPD) :**
 - Le traitement de données de santé (GIR, traitements) est strictement encadré. Un DPO (Délégué à la Protection des Données) sera nécessaire.
 - Le consentement explicite pour la collecte et le partage (même partiel) de ces informations aux bénévoles est obligatoire.
- **Droit du Travail :**
 - La relation doit rester dans un cadre de bénévolat, sans rémunération ni lien de subordination, afin d'éviter toute requalification en travail dissimulé.

Le projet est faisable sous réserve de respecter strictement le cadre RGPD et d'établir des CGU solides définissant clairement les responsabilités.

3.4 Veille concurrentielle

Afin de concevoir une application pertinente, simple et adaptée aux besoins réels des utilisateurs, nous avons appuyé notre démarche sur plusieurs sources de données et références. Ces supports ont permis d'ancrer notre projet dans un contexte social concret et de mieux comprendre les pratiques existantes en matière d'entraide et de coordination des aidants.

Nous avons analysé plusieurs plateformes déjà existantes afin d'identifier leurs points forts et leurs limites :

- **Voisins Solidaires**, qui favorise l'entraide de proximité entre habitants, mais sans réelle gestion d'emploi du temps ni suivi personnalisé.
- **Doctolib**, pour son interface fluide et sa gestion efficace des créneaux et notifications.
- **Mon Espace Santé**, qui met l'accent sur la sécurité et la confidentialité des données personnelles, aspects essentiels pour notre projet. Ces comparaisons nous ont permis de définir une application combinant **simplicité d'usage, coordination des tâches et sécurité des échanges**.

- **Gitlab (Gricad)**, l'envoi d'un problème ("issue") déclenche un courriel à l'ensemble des professeurs concernés. L'un d'eux prend en charge le problème et contacte l'équipe appropriée.

3.5 Organisation du travail

Outils utilisés

Pour assurer une coordination efficace de l'équipe et un suivi rigoureux du projet, plusieurs outils ont été utilisés :

- **Discord** : pour la communication instantanée et les échanges informels entre membres.
- **Google Drive** : pour le partage et la centralisation des documents, maquettes, rapports et supports de travail.
- **GitLab / GitHub** : pour le suivi du code, la gestion des versions et le traitement des issues/problèmes.
- **GitLab Issues** : pour la planification des tâches, le suivi de l'avancement et la gestion des priorités.
- **Jira** : pour la gestion des tâches et du temps

Répartition du temps

Le projet a été organisé sur plusieurs phases principales :

1. **Conception** : jusqu'en décembre
 - Définition des besoins, étude des personas, analyse des contraintes et risques
 - Maquette et prototype initial
2. **Développement** : trois premières semaines de janvier
 - Backend : création de la base de données, API, sécurité
 - Frontend : interface web et mobile, intégration des fonctionnalités
 - Tests unitaires et validation de chaque module
3. **Coordination et suivi** : tout au long de l'année
 - Réunions hebdomadaires pour faire le point sur l'avancement
 - Suivi des tâches et réajustements via Trello/GitLab
 - Suivi de la communication et des interactions utilisateurs

Répartition des rôles

Chaque membre de l'équipe a eu des responsabilités claires pour éviter les chevauchements et assurer l'efficacité du projet :

- **Chef de projet** : Yanis Debbal - suivi global, planification et coordination.
- **Responsable Conception** : Abdessalam Khattri, Kilian Nauleau - conception des maquettes et de l'interface.
- **Responsable Réalisation** : Gabriel Schoenfelder, Anas Kiouaz - développement technique backend et frontend.
- **Responsable Communication** : Rayan Nyabi - suivi de la communication, documentation et présentation.
- **Responsable Anglais** : Redon Brahimi - relecture et adaptation en anglais.

Le tableau complet de répartition des rôles (RACI) avec les détails sur qui est Responsable, Approuve, Consulté et Informé est présenté en **Annexe 4.3**.

4. Conclusion

Le projet **Weave** répond à un besoin croissant d'organisation et d'entraide autour des personnes âgées en perte d'autonomie. Ce cadrage a permis de définir clairement les objectifs, les utilisateurs, les fonctionnalités clés et les contraintes du projet. L'analyse des besoins et des personas confirme l'importance d'une plateforme simple, sécurisée et centrée sur la coordination entre aidants.

L'étude de faisabilité montre que le projet est techniquement réalisable grâce à une architecture backend–frontend solide et des technologies adaptées. Les principaux défis concernent surtout la sécurité des données, la conformité réglementaire et la gestion de la communauté d'utilisateurs.

Ce travail fournit ainsi une base structurée pour la suite du développement. Les prochaines étapes consisteront à concevoir puis développer un prototype afin de concrétiser le produit.

5. Annexes

| | |
|--|----|
| 5.1 Contraintes et risques – tableaux | 16 |
| a) Tableau de la Criticité : | 16 |
| b) Tableau des Mesures : | 16 |
| 5.2 Diagramme BPMN | 17 |
| Processus de planification d'une demande : | 17 |
| Processus de gestion d'incident : | 17 |
| 5.3 Tableau des rôles (RACI) | 18 |
| 5.3 Diagramme de Gantt (Jira) | 19 |

5.1 Contraintes et risques – tableaux

a) Tableau de la Criticité :

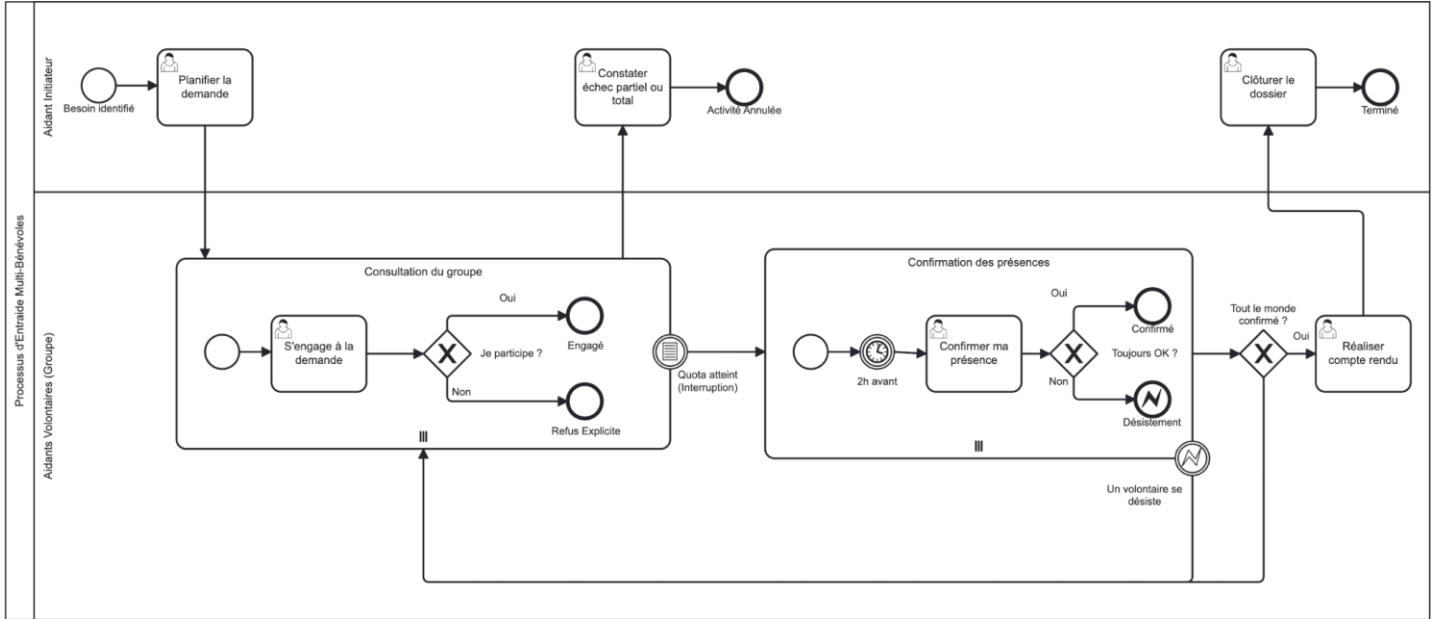
| | Probabilité d'apparition (x) | | | |
|------------|------------------------------|-----------|-----------|--------------|
| Impact (y) | | Faible 1 | Moyenne 2 | Forte 3 |
| | Faible 1 | Faible 1 | Moyenne 2 | Moyenne 3 |
| | Moyenne 2 | Moyenne 2 | Forte 4 | Forte 6 |
| | Forte 3 | Moyenne 3 | Forte 6 | Très Forte 9 |

b) Tableau des Mesures :

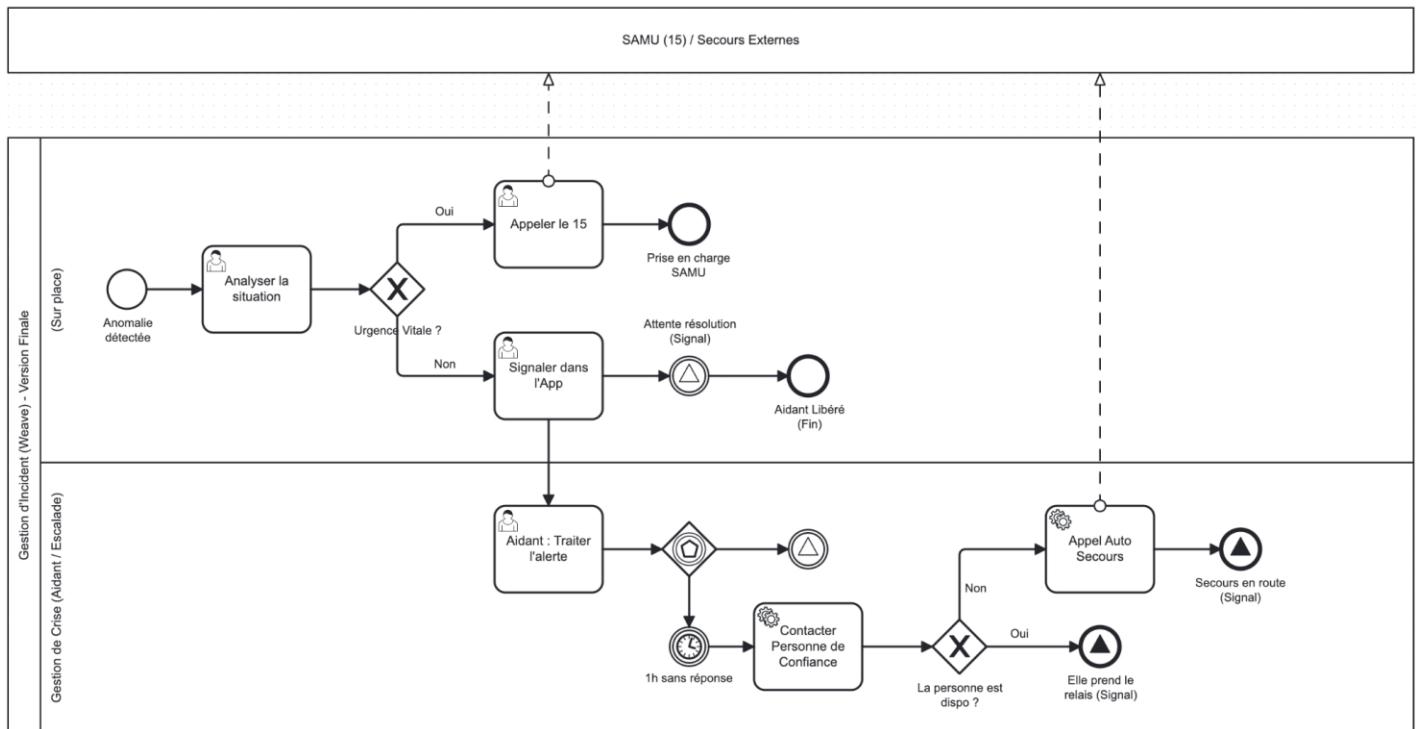
| Criticité | Mesure |
|--------------|--------------------------------|
| Faible 1 | Acceptation |
| Moyenne 2-3 | Mesure corrective |
| Forte 4-6 | Mesure corrective / préventive |
| Très Forte 9 | Mesure préventive |

5.2 Diagramme BPMN

Processus de planification d'une demande :



Processus de gestion d'incident :



5.3 Tableau des rôles (RACI)

1. Identification des rôles (RACI)

Pour assurer une organisation claire et efficace du projet, nous avons défini les rôles de chaque membre de l'équipe. Chaque rôle est associé à des responsabilités spécifiques. Nous avons utilisé la matrice RACI pour clarifier qui est Responsable, qui Approuve, qui est Consulté et qui est Informé pour chaque tâche. Cette répartition des rôles permet d'éviter les chevauchements et les confusions, assurant ainsi une meilleure gestion du projet.

| Rôle | Nom | Responsable (R) | Approuve (A) | Consulté (C) | Informé (I) |
|---------------------------|----------------------|-----------------|--------------|------------------------|------------------------------|
| Chef de Projet | Yanis Debbal | R | A | Tous les autres rôles | - |
| Responsable Communication | Rayan Nyabi | R | A | Chef de projet | Tous les membres de l'équipe |
| Responsable Réalisation | Gabriel Schoenfelder | R | A | Responsable Conception | Chef de projet |
| Responsable Réalisation | Anas Kiouaz | R | A | Responsable Conception | Chef de projet |
| Responsable Anglais | Brahimi Redon | R | A | Chef de projet | Tous les membres de l'équipe |
| Responsable Conception | Nauleau Kilian | R | A | Responsable conception | Tous les membres de l'équipe |
| Responsable Conception | Abdessalam Khattri | R | A | Chef de projet | Tous les membres de l'équipe |

- **R (Responsable)** : La personne qui effectue le travail.
- **A (Approuve)** : La personne qui valide ou approuve le travail.
- **C (Consulté)** : Les personnes dont l'avis est sollicité.

- I (Informé) : Les personnes qui sont tenues informées de l'avancement.

5.3 Diagramme de Gantt (Jira)

