

# Chapitre 3

## Organisation

Tout au long de l'année académique, la réalisation du mémoire a été impactée par la composition particulière du groupe. Trois mémorants, deux promoteurs et un client, Protos. Autant de points de vue, horaires, désirs et apports à intégrer dans ce travail. Afin de conserver un cap commun et de pouvoir concilier les objectifs de chacun, il a été nécessaire de définir, planifier et attribuer des tâches.

### 3.1 Approche de travail

Comprendre les problématiques liées au mémoire a été notre première tâche. En tant qu'étudiants en sciences informatiques, notre formation n'inclut pas de cours sur la gestion de l'eau et le contexte haïtien. L'idéal aurait été d'intégrer dans l'équipe des consultants sur les domaines méconnus et d'envisager une visite en Haïti pour rencontrer les acteurs locaux et appréhender la situation.

Pour des raisons évidentes de budget et de temps, l'option de se rendre sur place et engager plus de partenaires a immédiatement été écartée. Pour comprendre la situation et relever les défis posés par le contexte, nous avons pu compter sur l'expertise de nos promoteurs et les 700 pages de documentation qui nous ont été fournies en préambule. Cette première documentation, orientée sur le contexte, a apporté des éclaircissements sur les défis de la gestion de l'eau potable en Haïti (ci-avant présentés en section 2.1).

Tandis que les besoins non-fonctionnels de l'application étaient établis par ce contexte et les vues de Protos, le contenu fonctionnel s'est basé sur les outils actuellement en place. Nous avons vu en section 2.2 que des solutions locales ont été mises en place. En essayant de tirer le meilleur de toutes ces solutions, nous avons établi un cahier des charges (voir chapitre 4).

L'arrivée en Belgique de plusieurs acteurs haïtiens impliqués dans la gestion de l'eau potable a modifié notre approche. A partir de janvier 2019, des réunions régulières avec des utilisateurs potentiels de l'application ont été mises en place. Ce changement d'approche a permis d'orienter le développement vers un but bien plus précis qu'auparavant, nous permettant de bénéficier de nouvelles expertises et nouveaux points de vue.

## Planification

L'organisation d'un groupe, même réduit à trois personnes, est nécessaire pour orienter le travail dans un but commun. La planification de la réalisation du mémoire, qui est constitué en vaste majorité de l'application web produite, s'est effectuée sur trois plans :

**Mensuel** Au départ du mémoire, un plan général a été dressé. Au début, les inconnues (e.g., étendue du projet, fonctionnalités à implémenter) ont empêché de définir avec précision les productions et les dates butoires. Avec l'avancée du mémoire, nous avons pu estimer l'état et notre vitesse de travail. Les réunions avec Protos pour présenter nos deux plus grandes échéances, à savoir les versions alpha et beta de l'application web, ont été l'occasion de refaire le point et reformuler les objectifs.

**Hebdomadaire** Chaque semaine en alternance, nous rencontrions un de nos promoteurs. Dans la continuité de la planification mensuelle, ces réunions permettaient de présenter les productions, rapporter les tâches en cours et leur avancement, et enfin de projeter le futur du travail. Ces présentations étaient également l'occasion de rassembler les productions et d'avoir, chaque semaine, des documents ou démonstrations pouvant être comprises par des acteurs étrangers à l'équipe de développement. C'était donc autant un médium d'information que l'occasion de nous recentrer sur le travail, sans compter les précieux retours de nos promoteurs.

**Quotidien** Après une période d'expérimentation avec plusieurs outils de suivi, la gestion au quotidien des tâches à accomplir s'est ancrée sur Trello<sup>1</sup>, un tableau en ligne composé de cartes amovibles réparties dans des colonnes. Dans notre organisation, une carte représente une tâche plus ou moins atomique semblable à un *récit utilisateur*<sup>2</sup>. Nous utilisons les colonnes pour classer les cartes selon l'état de la tâche associée. Nous avons ainsi défini les colonnes : *à faire*, *à faire urgemment*, *en cours*, *problématique* et *fini*. Nous avons également utilisé des messageries instantanées (e.g. Slack<sup>3</sup>) et canaux vocaux pour organiser les tâches.

Etablir un plan et s'y tenir n'était cependant pas suffisant. Le projet a connu de nombreuses modifications dans ses objectifs et contraintes. Différents facteurs viennent expliquer ces modifications :

- La présentation de prototypes aux partenaires et aux promoteurs ont souvent amené de nouvelles idées pour l'application, ou suggéré la modification de fonctionnalités existantes.
- L'arrivée de nouveaux acteurs, principalement aux mois de février et mars. Plusieurs haïtiens ont pu apporter des éclaircissements sur la situation qui ont modifié l'application et l'orientation de son développement.
- Certaines fonctionnalités se sont révélées plus complexes à implémenter que prévu.

La figure 3.1 présente une vue simplifiée du diagramme de Gantt final sur lequel les modules implémentés (voir section 4.3) sont placés sur une ligne temporelle en dessous de laquelle les grandes phases du mémoire ont été placées, ainsi que les deux versions majeures, *alpha* et *beta*, présentées à tous les acteurs du projet.

Les phases d'*analyse* et *architecture* se sont également clôturées par des réunions de taille plus importante afin qu'un maximum de personnes puissent donner leur avis sur les artefacts (cahier des charges, ébauches d'interface graphique, etc). C'est notamment après ces réunions que les changements de planification étaient décidés pour répondre aux nouveaux besoins du client.

Le diagramme en figure 3.1 est donc la version finale. Comparé au plan initial, on constate une augmentation de la phase d'implémentation en raison de l'augmentation de la charge de travail, les modules *historique* et *finances* étaient en effet

---

1. <https://trello.com>

2. Un récit utilisateur est une phrase ou texte court décrivant une fonctionnalité à développer

3. <https://slack.com/intl/fr-be/>

imprévus. De nombreuses autres fonctionnalités ajoutées à posteriori ont allongé cette phase. Il était également prévu de passer plus de temps sur l'analyse et la conception du système. Or le cahier des charges et les ébauches n'étaient pas assez concrets pour avoir des retours précis sur l'outil. Devant ce fait et l'ampleur du projet, nous avons démarré l'implémentation plus tôt que prévu, et elle s'est également allongée sur le mois d'avril (pour un total d'environ 6 semaines supplémentaires).

Ces imprévus n'ont heureusement pas eu d'impact sur la production. Le démarrage anticipé de l'analyse en août, l'ajout d'une semaine de travail imprévue durant la session d'examen de janvier ainsi qu'une zone tampon de deux semaines prévue dans l'horaire initial ont permis d'accomplir la totalité des tâches initialement prévues, ainsi qu'une part appréciable des suggestions apportées au cours de l'année.

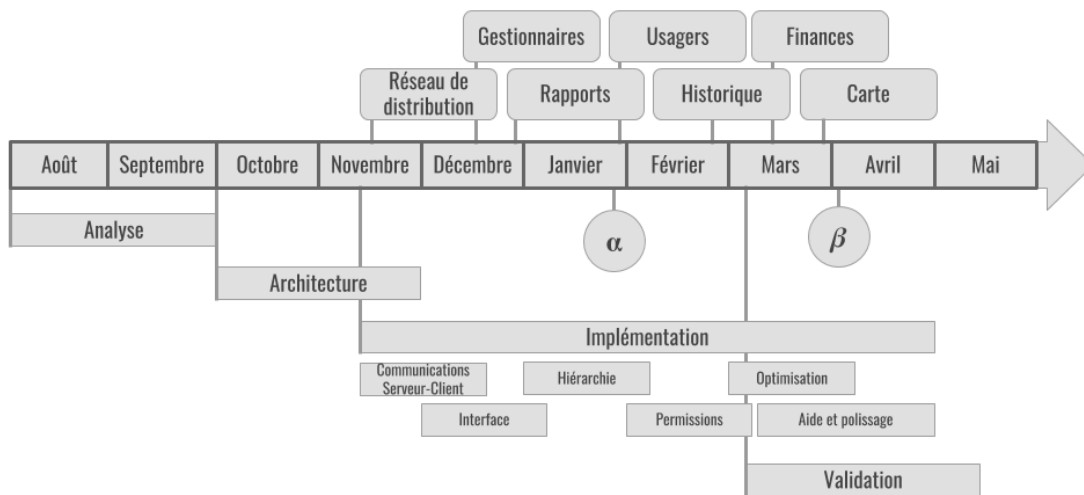


FIGURE 3.1 – Représentation simplifiée du diagramme de Gantt final

## 3.2 Méthodologie

Garantir l'avancement sans variation indésirable de tout projet se fait par le biais d'une méthodologie. Il est important que cette méthode de travail soit connue

de chacun pour garantir un rôle fixe et un suivi efficace. Cette notion de méthode est une caractéristique bien connue de la gestion de projet, le but étant de partir d'un besoin et d'apporter une solution informatisée. La méthode lie ces deux états et définit un système de communication, échéances, tâches, etc. De nombreuses méthodes déjà établies existent et sont prêtes à l'emploi par n'importe quelle équipe.

## Agile

Dans la section précédente, nous avons évoqué les changements constants dans la planification et les tâches. C'est donc tout naturellement que nous avons opté pour une *méthode agile*, en opposition aux méthodes traditionnelles dites "waterfall" qui se montrent plus rigides et réfractaires au changement.

Nous n'avons pas suivi une méthode agile fixe, préférant une version plus relaxée, adaptée aux variations de disponibilité inhérentes à notre statut d'étudiant. On peut cependant relever des grands principes inspirés de méthodes existantes.

### **Développement dirigé par les fonctionnalités** *Feature Driven Development*

en anglais, ce type de développement focalise les ressources sur la création. Tout d'abord création d'un prototype général et d'une liste de fonctionnalités. Ensuite création de celles-ci, ajoutées à l'application en impliquant le client à chaque fois qu'une fonctionnalité est créée. C'est également le client qui définit la priorité des fonctionnalités. Ce dernier point fut très utile dans notre mémoire où les acteurs haïtiens ont souhaité orienter le développement vers l'aspect financier de l'application, en opposition à Protos qui accentuait principalement le réseau de distribution.

**Itérations courtes** Une itération est une étape indépendante du développement. Chaque itération a sa propre planification et rétrospective. En méthode agile, les itérations sont courtes. Puisque nous voyions nos promoteurs toutes les semaines en alternance, nos itérations se sont naturellement calquées sur ce rythme. Nous avons évoqué l'intérêt de recentrer le développement en section 3.1. C'est également une excellente manière de détecter rapidement les risques (tant au niveau de la planification que de l'application produite) et d'agir en conséquence.

**Livraison continue** A partir de février, un acteur haïtien arrivé en Belgique a utilisé intensivement l'application. Grâce à ses retours, nous avons pu modifier l'application. Durant trois mois, nous avons déployé de nouvelles

versions de l'application toutes les deux ou trois itérations afin qu'il puisse essayer les fonctionnalités ajoutées. Cela a nécessité un peu plus de travail, pour déployer l'application et garantir l'intégrité des données entre les différentes versions. Effort récompensé par des retours fréquents de la part d'un utilisateur impliqué dont le point de vue extérieur s'est avéré extrêmement utile.

## Phases du mémoire

Si le développement (implémentation) de l'application a suivi ces principes agiles, on notera que le déroulement d'ensemble du mémoire est divisé en grandes phases. Ce principe est mis en avant dans la figure 3.1. Cette caractéristique est traditionnellement indicatrice d'un développement en cascade où les différentes étapes du développement logiciel sont délimitées en temps et responsabilités (analyse, conception, implémentation, test, déploiement).

Les raisons sont multiples. On citera notamment notre méconnaissance des contextes de la gestion de l'eau potable et d'Haïti pour lesquels il a fallu se documenter pendant une période assez longue ainsi que la quantité de documents initiaux qui ont forcé à travailler durant ces premiers mois sur la compréhension et l'ébauche d'une solution.

Les disponibilités des acteurs externes au développement (membres de Protos, Haïtiens, chercheurs externes) ont aussi été un facteur. Chacun ayant une mission déterminée, nous ne pouvions avoir de contacts trop fréquents (aussi dûs à la distance et au décalage rendant difficile l'organisation de réunions). Nous avons donc dû aussi adapter les itérations aux disponibilités, variables, de chacun. Ceci explique aussi partiellement pourquoi le développement s'est autant accéléré lorsque des acteurs haïtiens sont venus en Belgique et nous ont offert des retours fréquents, ont participé à différentes réunions. La disponibilité du client est très importante dans le développement agile afin de pouvoir évaluer le travail accompli. En l'absence de membres extérieurs, ce sont nos promoteurs qui ont parfois joué ce rôle de client.

## 3.3 Répartition des tâches

Dès le début, nous avons cherché à tirer avantage de nos centres d'intérêts et capacités différents. Nous avons eu la chance d'être une équipe complémentaire et d'avoir un mémoire intéressant au point que toutes les tâches étaient attribuées sans problème et rapidement. Dans cette section, nous discutons de cette attribution systématique et de ce qu'elle implique pour ce travail.

## Analyse

La phase d'analyse principale, d'août à octobre, avait pour but de comprendre la situation et le rôle de l'application. Afin que toute l'équipe avance vers le même objectif, il a été nécessaire que chacun soit impliqué dans l'analyse. En débutant par une recherche libre, nous avons cherché à nous documenter sur le contexte haïtien, les solutions existantes et les entreprises qui appliquent ces solutions. L'apport de plusieurs centaines de pages de documentation par Protos et nos promoteurs a nécessité une certaine organisation. Demander à chacun de nous, mémorants, de tout lire nous a semblé improductif, mais il était nécessaire que les informations circulent. Nous avons donc adopté un système de file pour les documents. Chaque nouveau document est placé dans cette file et pris en charge par un de nous trois (selon disponibilité et affinité). Au fil de la lecture, un document résumant le contenu et relevant les informations utiles pour le mémoire est produit. Le résumé terminé et mis à disposition, il est lu et discuté en groupe. Un autre objectif de ce système est de pouvoir bénéficier de documents raccourcis pour rechercher des informations, notamment dans la rédaction de ce travail.

Précisons cependant que même si ce que nous considérons être comme la phase d'analyse principale était de août à octobre, les changements d'exigences et l'apport de nouvelles informations au cours de l'année ont induit des phases d'analyse complémentaires, principalement au démarrage l'implémentation de chaque module. A nouveau, nous nous sommes tous trois impliqués dans ces phases afin que le développement aille dans un sens unique.

## Conception et implémentation

Contrairement à l'analyse, la conception et l'implémentation ont connu une très forte séparation des tâches entre nous trois, chacun ayant un champ de travail défini.

**Adrien** responsable de la gestion du projet (planification, présentations intermédiaires, communication), de l'analyse fonctionnelle et du *front-end* (le client) : interface graphique, fonctionnalités utilisateur.

**Céline** responsable du *back-end* (le serveur) : base de données, traitement des informations et calculs, gestion des performances.

**Sébastien** responsable du *service worker* (l'accès hors-ligne à l'application), des tests et renfort au *back-end* sur le module financier et la vérification des permissions.

Pour chacune de ces préoccupations, le responsable désigné s'est chargé de la conception, des schémas, de la réalisation (implémentation) et de la documentation. Une communication constante et présentation de l'avancement des tâches à chaque membre de l'équipe se sont implicitement imposées, toujours dans l'optique d'orienter le développement dans le même sens.

L'inconvénient principal de ce type d'organisation et la compartementalisation qui nuit à l'apprentissage individuel puisque tout le monde ne travaille pas avec la totalité des technologies utilisées. Cela pose un risque d'arrêt, même temporaire, du développement si un des membres se trouve dans l'incapacité d'assumer sa charge de travail. En entreprise, ce risque serait à corriger au plus vite en impliquant plus d'acteurs dans chaque branche du développement, ou en impliquant tous les acteurs dans la totalité du code (par exemple via un système de rotation).

Nous avons tout de même effectué ce choix pour la puissance de travail qu'il permet et les facilités de développement et d'organisation apportées. Chacun de nous a développé une connaissance accrue de la technologie utilisée dans sa partie, permettant une augmentation drastique de l'efficacité qui s'est particulièrement ressentie dans la seconde partie (second quadrimestre) du travail. Organisationnellement parlant, savoir directement à qui s'adresser en cas de problème ou demande de fonctionnalité a été également particulièrement pratique.

## Rédaction

La rédaction du présent texte a à nouveau rassemblé l'équipe. En tant que document résumant et justifiant le travail accompli, il était naturel d'être tous inclus dans son élaboration. Nous avons considéré les différentes sections en tant qu'entités individuelles, permettant un déroulement de la rédaction simplifié, étape par étape. A des granularités différentes (parfois allant jusqu'à établir le plan pour chaque sous-section), nous avons dressé une liste du contenu à intégrer. La section était ensuite attribuée au mémorant responsable du sujet discuté.

Une fois le premier jet établi, une relecture permettait à chacun d'apporter les modifications désirées, parfois discutées, pour obtenir un texte dans lequel chacun de nous peut s'identifier. Ce processus était répété pour chaque section et chaque jet, encouragés et orientés par les commentaires de nos promoteurs.



## Au quotidien

La planification générale avait une précision au maximum hebdomadaire. Pour le travail effectué au jour le jour, nous nous sommes entièrement reposé sur Trello avec un système de colonnes décrivant l'état et la priorité de chaque tâche comme exposé ci-avant. Autre héritage des méthodologies agiles, ces cartes sont des tâches à réaliser pour avancer le développement. Dans notre cas, les tâches étaient souvent atomiques ou ne reflétaient qu'une seule unité de développement (e.g., ajouter un champ dans un formulaire).

Un système de description de la tâche, assignation de membre(s) à une carte et étiquetage, ajouté au principe des différentes colonnes indiquant la priorité et l'état de chaque carte, a permis de gérer efficacement le travail.

Quant à l'ajout des cartes, on relève deux possibilités :

- Au début des itérations, lorsqu'on planifiait le sprint à venir.
- Durant une itération, lorsqu'un membre de l'équipe avait besoin d'une fonctionnalité qu'un autre membre devait implémenter, le premier membre ajoutait une carte en assignant le second. Par exemple, si le développement du client nécessitait l'implémentation d'une fonctionnalité du serveur, la carte était ajoutée en assignant directement la personne responsable.

La colonne de travail prioritaire s'est avérée une très bonne idée. Puisque nos fonctionnalités sont dépendantes à la fois du client et du serveur, cela permettait d'identifier clairement les besoins. Nous pouvions ainsi débiter une séance de développement à n'importe quel moment en sachant ce dont les autres avaient besoin.