

BLAAJJPAIN

ANALYSE DE LA RÉPARTITION DU TEMPS DE TRAVAIL



Auteurs :

Benoît Schopfer
Loyse Krug
Adrien Allemand
Antoine Rochat
Jérémy Châtillon
James Smith

Client :

René Rentsch

Mai 2018

Table des matières

Planification initiale du projet.....	1
Comparaison entre la planification et la réalisation	3
Conclusion	6

Planification initiale du projet

La planification initiale du projet prévoyait 537 heures de travail, réparties équitablement sur l'ensemble des 6 développeurs du groupe, selon le planning suivant :

	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15	S16	Total [h]
Adrien Allemand	3.5	6.5	8	7	6	5.5	6.5	6.5	8	6	6	7	6	5.5	1.5		89.5
Antoine Rochat	3.5	9	8	7	6	6	5.5	7	7	5	6	7	6	5.5	1.5		90
Benoît Schopfer	3.5	10	8	7	6	6	6	5	8	5	6	7.5	6	5.5	1.5		91
James Smith	3.5	5.5	8	7	6	6	6	8	8	5	6	6.5	6	5.5	1.5		88.5
Jérémy Chatillio	4	7.5	8	7	6	5	7	6	8	5	6	6	6	5.5	1.5		88.5
Loyse Krug	3.5	8.5	8	7	6	5.5	6	5.5	8	5	6	7.5	6	5.5	1.5		89.5
Total [h]	21.5	47	48	42	36	34	37	38	47	31	36	41.5	36	33	9	0	537

Tableau 1 - Répartition du travail par développeurs¹

Cette planification initiale a été réalisée durant la première semaine et s'est avérée relativement éloignée du résultat final.

Nous avons rapidement compris que l'ordre de réalisation prévu des tâches était erroné, principalement avec la planification de la réalisation du *Undo/Redo* en semaine 9. En effet, nous avons réalisé que l'intégralité des outils allait devoir implémenter les fonctionnalités requises par l'historique pour que ce dernier fonctionne. Nous avons donc procédé à une redistribution des priorités et une modification de l'ordre de réalisation des tâches.

Face à ce problème, notre erreur a été de ne pas modifier la planification directement sur le fichier de base. Nous avons commencé par mettre la priorité sur l'historique en pensant qu'un simple décalage ne demandait pas de modification de l'ordre des autres tâches. Cependant, les répercussions sur la suite du projet se sont rapidement multipliées et, en parallèle avec le groupe qui développait l'historique, les autres membres du groupe se sont réparti "à la volée" les tâches qu'il restait à faire. Résultat, nous nous sommes passablement éloignés du planning initial et ce dès le début de notre projet. De plus, d'autres erreurs ont été faites dans la planification initiale. Par exemple, nous n'avons pas pris en compte la semaine de vacances de Pâques et l'avons comptée comme une semaine de travail normal, ce qui fût évidemment pas le cas...

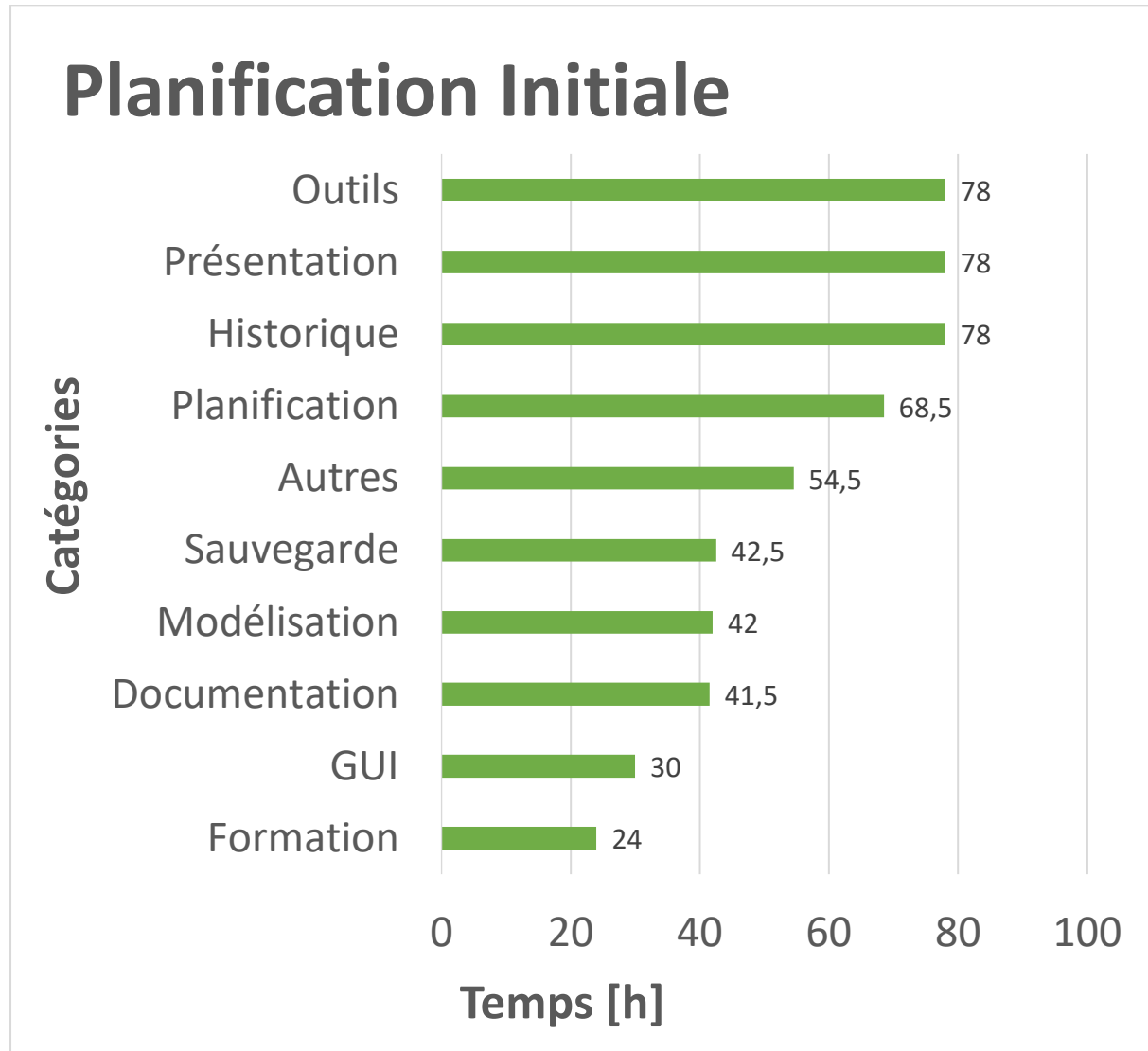
Pour finir nous nous sommes presque basés sur une méthode *AGILE* avec une répartition des fonctionnalités à implémenter à la volée où chacun prenait la fonctionnalité suivante à implémenter lorsqu'il avait fini celle sur laquelle il travaillait.

Bien que toutes les fonctionnalités aient pu être traitées avec cette méthode, elle pose le problème majeur qu'il nous a été impossible de réaliser un fichier Gantt a posteriori pour comparer avec la planification initiale l'ordre de réalisation des tâches.

¹ Le détail des tâches prévues pour chaque semaine par développeurs est disponible dans le fichier annexe *Répartition_heures.xlsx*.

Afin de proposer une analyse de planification sensée, nous avons choisis de répartir les tâches en catégories et de nous concentrer sur la différence entre les temps prévus initialement pour la réalisation de chacune des catégories et le temps effectivement utilisé.

Voici la répartition des heures de travail planifiées au début du projet et réparties en catégories :



Graphique 1 - Planification initiale des heures de travail, réparties par catégorie de tâches.

La répartition en catégories s'est faite selon les critères suivants :

- **Outils** : regroupe tous les outils de dessin ou d'édition de calques.
- **Présentation** : regroupe le temps passé à la réalisation de la présentation finale du projet.
- **Historique** : regroupe le temps passé aux fonctionnalités de gestion de l'*Undo/Redo*.
- **Planification** : regroupe la réalisation de la planification du travail, du cahier des charge, des séances de réunion, des discussions sur le projet et de la critique du travail effectué.
- **Sauvegarde** : regroupe le temps passé à la sauvegarde et au chargement du projet au format .blaajj.
- **Modélisation** : regroupe la planification du code, la création du schémas UML, du modèle de conception, la refactorisation du code et la modification de conception à posteriori.

- **Documentation** : regroupe le temps de rédaction du rapport, de la *JavaDoc* et des commentaires.
- **GUI** : regroupe *JavaFX*, l'interface graphique, et la liaison de l'interface et du backend.
- **Formation** : regroupe la recherche et l'auto-formation sur les APIs et modèles conceptuels.
- **Autres** : regroupe le temps prévu en réserve pour finaliser le projet ainsi que diverses petites tâches.

Comparaison entre la planification et la réalisation

En analysant la planification on peut remarquer que nous avons estimé le gros du temps de travail entre le développement des **Outils**, la mise en place de l'**Historique** et la réalisation de la **Présentation** avec 78h par catégorie.

Par opposition nous pensions pouvoir réaliser rapidement la **GUI** et ne devoir investir que peu d'heures dans la **Formation** avec respectivement 30 et 24 heures.

A noter aussi que nous avons prévu une réserve de 36h en semaine 11 pour la finalisation du projet en vue du rendu final.

Il est intéressant de mettre en perspective cette planification initiale avec la réalisation du projet en regardant les différences entre les deux.

Le graphique de la page suivante met en évidence des écarts importants pour plusieurs catégories notamment pour la réalisation des **Outils** qui a demandé un total de 187h soit 109h de plus que les 78h prévu initialement.

Cela est dû à plusieurs choses : la planification se base sur un modèle MVC qui devait distinguer le Model (les calques) du Controller (les outils), or il s'est rapidement avéré difficile de séparer les deux ce qui a compliqué la mise en place des outils, notamment leurs factorisations. Conséquence de quoi ceux-ci ont été réalisés en exploitant les fonctions offertes par *JavaFX* pour la modification des canevas qui contiennent "l'image" à modifier. Cela a rendu leur développement fastidieux car en dépit du temps important investi dans la formation, nous avons rencontré de multiples petits bugs qui demandaient des heures de recherche pour trouver une solution qui s'est avérée courte et plutôt simple, ce qui nous a parfois frustré par l'impression d'avancer lentement. Par exemple le trait du *Pencil* faisait des petits piques au lieu d'une belle ligne droite car il manquait une simple ligne de code pourtant cette ligne ne figurait quasiment nulle part dans la documentation de *JavaFX* et il a fallu plus de 8h de recherches réparties sur 3 développeurs pour en venir à bout.

Ce sont principalement les difficultés à utiliser *JavaFX* qui ont fait exploser le temps passé à développer les outils.

La **Modélisation** a elle-aussi pris beaucoup plus de temps que prévu car nous avons dû régulièrement refactoriser le code pour l'adapter à tel ou tel fonctionnalité que nous avons mis en place. A nouveau, cela est principalement dû à la mauvaise connaissance de *JavaFX* dont la découverte des différentes fonctionnalités qu'il offre est venue au fur et à mesure de l'avancée du projet, ceci malgré la formation initiale que nous avions prévue. Nous avons parfois été forcé de modifier tel ou tel partie du projet en révisant sa modélisation à la suite de la découverte d'une manière de faire plus appropriée ou plus souple que ce qui était déjà en place.

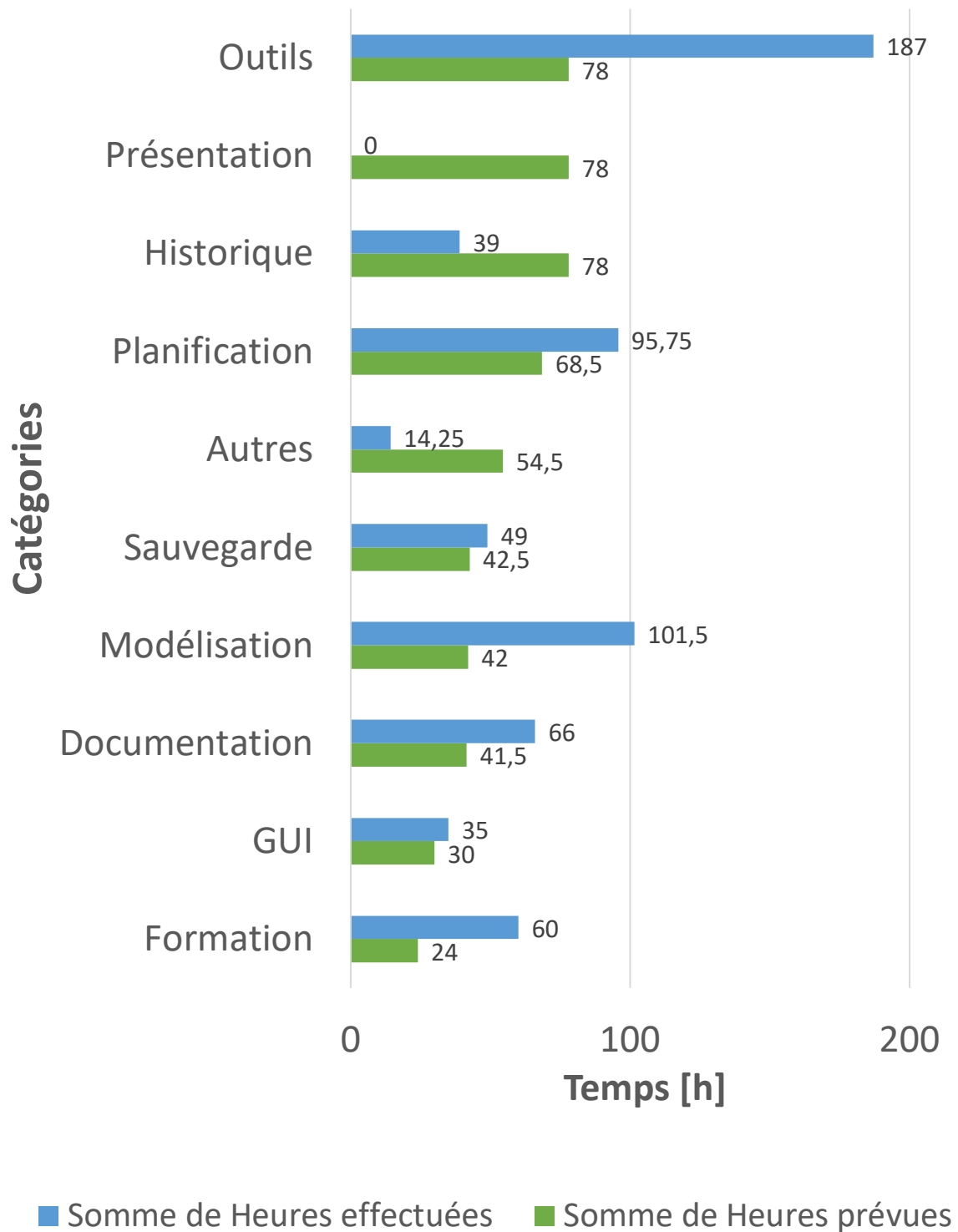
Comme mentionné précédemment, la **Formation** a elle-aussi pris plus de temps que prévu car nous avons régulièrement dû approfondir nos connaissances sur *JavaFX* ou en matière de Sériation.

La mise en place de **l'Historique** a été la seule catégorie qui a réellement pris moins de temps que planifié. En effet la mise en place d'une structure d'enregistrement des modifications sous forme de sauvegardes "undoables" implémentant une variante du model conceptuel *commande* a été efficace. A noter cependant que l'implémentation des classes de sauvegardes propre aux outils a été traitée dans la catégorie susmentionnée, ce qui explique en partie l'explosion des heures pour les **Outils**.

Pour la **Présentation**, au moment où ce rapport est écrit, aucun membre du groupe n'a consacré des heures dessus, cela explique que les heures dans la réalisation soit de 0. Cette tâche sera effectuée dans les prochaines semaines.

Le reste du travail a pris à peu près le temps estimé, il est sans surprise que, de manière générale, le développement d'un tel projet prenne plus de temps que prévu.

Planification vs Réalisation



Graphique 2 - Planification initiale vs temps de réalisation

Conclusion

La planification initiale du projet a été un des points les plus difficile de tout le projet. Réaliser une planification aussi détaillée que celle qui nous était demandée relevait de l'impossible car nous n'avions aucune idée du temps que prendrait chaque tâche. Nous ne connaissions même pas encore les modèles que nous allions utiliser pour implémenter des fonctionnalités telles que l'historique ou la sauvegarde du projet d'où la difficulté à estimer la charge de travail qu'elles représenteraient.

Paradoxalement, nous avons étudié, en parallèle de la réalisation du projet, des méthodes de travail dans le cadre du cours de Génie Logiciel qui nous auraient été fort utile de connaître avant la réalisation d'un tel projet.

Si un tel projet était à refaire nous emploierions certainement une méthode plus flexible et tolérantes face aux changements de direction.