

Master 1 GIL - Document d'informations : Bilan de sprints

Groupe 1 Agora V3-1 : Partie bibliothèque de jeux

22 mai 2024

Version	7
Date	22 mai 2024
Rédigé par	CAUCHOIS Niels MAZUY Axelle MONTAGNE Erwann VAN LIEDEKERKE Florian

Mises à jour du document

Version	Date	Modification réalisée
1	8 février 2024	Création du document Bilan sprint 1
2	22 février 2024	Bilan sprint 2
3	12 mars 2024	Bilan sprint 3
4	26 mars 2024	Bilan sprint 4
5	9 avril 2024	Bilan sprint 5
6	18 avril 2024	Bilan sprint 6
7	22 mai 2024	Bilan sprint 7 (correctif)

Table des matières

1	Introduction	4
2	Terminologie	4
3	Bilan de sprint 1	4
3.1	Avancement global	4
3.2	Communication d'équipe	4
3.3	Technologie	4
3.4	Qualité de code	4
3.5	Qualité de travail	5
4	Bilan de sprint 2	5
4.1	Avancement global	5
4.2	Communication d'équipe	5
4.3	Technologie	5
4.4	Qualité de code	5
4.5	Qualité de travail	5
5	Bilan de sprint 3	5
5.1	Avancement global	5
5.2	Communication d'équipe et qualité de travail	6
5.3	Périmètre du travail	6
6	Bilan de sprint 4	6
6.1	Avancement global	6
6.2	Communication d'équipe	6
6.3	Communication avec l'équipe plate-forme	6
7	Bilan de sprint 5	6
7.1	Avancement global	6
7.2	Communication d'équipe	7
7.3	Technologie	7
7.4	Qualité de code	7
7.5	Qualité de travail	7
8	Bilan de sprint 6	7
8.1	Avancement global	7
8.2	Communication d'équipe	8
8.3	Technologie	8
8.4	Qualité de code	8
8.5	Qualité de travail	8
9	Bilan de sprint 7 (correctif)	9
9.1	Avancement global	9
9.2	Communication d'équipe	9
9.3	Technologie	9
9.4	Qualité de code	9
9.5	Qualité de travail	9

1 Introduction

Ce document a pour but d'avoir une vision globale de l'avancement sprint par sprint du projet. Il fait notamment part de ce qui a pu ressortir lors des sprints rétrospectives.

Complété par les Scrum masters notamment, ce document sert donc notamment à l'amélioration des sprints suivants.

2 Terminologie

Scrum Master : veille à ce que tous les membres de l'équipe aient connaissance de l'état d'avancement, des objectifs et des obstacles du projet

Sprint : désigne une brève période limitée dans le temps dont une équipe Scrum a besoin pour effectuer une quantité de travail donnée

3 Bilan de sprint 1

3.1 Avancement global

Ce premier sprint a été marqué d'un bon début. L'ensemble des tâches demandées ont été réalisées, à l'exception des animations du 6 qui prend qui étaient encore à voir et terminer. Cela représente donc le seul travail du sprint 1 à terminer par rapport au sprint 2.

La documentation détaillée nous a notamment permis d'avancer efficacement en back-end, tout en apportant des points d'amélioration notamment aux entités en Base de données, et avoir une base correcte d'un jeu pour poursuivre sur le jeu suivant.

3.2 Communication d'équipe

L'ensemble des membres de l'équipe communique efficacement et la tenue de nombreuses réunions participe au bon avancement du projet.

L'ensemble des membres s'entend bien, et la bonne humeur du groupe facilite le travail et l'échange.

Néanmoins, la communication entre les équipes front-end et back-end pourrait être améliorée avec l'établissement d'un contrat préalable des fonctionnalités dont le front aurait besoin, ce qui lui éviterait de devoir modifier l'ensemble de son code une fois ces fonctionnalités développées. Un canal de discussion a donc été créé pour palier à cela.

Enfin, suite à la surdité de l'une des membres qui s'est retrouvée sans appareil, la communication s'est retrouvée plus complexe et il est donc important de mettre en place des alternatives afin que personne ne se sente isolé.

3.3 Technologie

D'un point de vue technologies, les membres sont encore quelque peu en apprentissage, mais la preuve de concept réalisée auparavant a permis d'établir de bonnes bases de connaissances mises en pratique aussitôt lors de ce sprint.

Il reste de nombreux points à voir et comprendre cela dit, comme Mercure qui pose encore problème notamment pour sa configuration, ou encore l'incompatibilité de git flow avec certains systèmes. Une réunion avec le référent technique est donc prévue pour palier à ce manquement concernant Mercure, et des commandes alternatives ont été trouvées à git flow.

3.4 Qualité de code

Une CI/CD est en cours de développement, et l'apprentissage du développement de celle-ci se fait progressivement.

Des tests commencent également à être mis en place pour assurer la bonne qualité du code.

3.5 Qualité de travail

Malgré les nombreux points positifs, certaines tendances ont pu être remarquées dans le groupe, notamment un manque de prise de responsabilité et une mauvaise gestion du temps de travail.

Il est important que chaque membre respecte le rôle qui lui est donné et que ce travail ne soit pas effectué par une autre personne par manque d'implication. De même, il est important d'apprendre à gérer son temps de travail et ne pas faire certaines tâches au dernier moment, au risque de bloquer les tâches qui en dépendent.

4 Bilan de sprint 2

4.1 Avancement global

Ce sprint était dédié au 2ème jeu, Splendor, et nous l'avons fini de manière fonctionnelle : il manquait seulement les animations qui étaient prévues pour le sprint 3. Nous avons donc rempli toutes nos tâches du sprint.

4.2 Communication d'équipe

L'ambiance est toujours aussi bonne et l'équipe s'entend bien. La mise en place du channel pour communiquer entre le front et le back a grandement amélioré les choses. Chaque partie savait parfaitement ce dont l'autre avait besoin.

Cependant, il ne faut pas hésiter à demander de l'aide lorsque l'on en a besoin, c'est important de vouloir apprendre par soi-même mais il ne faut pas pour autant mettre ses tâches en retard.

4.3 Technologie

Les technologies sont de mieux en mieux connues, chaque partie, front comme back, maîtrise ses outils et technologies et cela nous permet d'avancer rapidement.

Beaucoup d'outils de surveillance ont été mis en place pour nous assurer de la bonne santé du serveur, il est cependant bon de rappeler qu'il ne faut pas se perdre dans les configurations, il faut aussi avancer dans les tâches qui nous sont dédiées.

4.4 Qualité de code

Une CI/CD est totalement mise en place, cette dernière nous permet d'effectuer l'ensemble de nos tests ainsi que de déployer notre code sur le serveur.

Nous avons atteint 90% de couverture de tests.

4.5 Qualité de travail

Grâce à la maîtrise des technologies, les membres de l'équipe ont pris une légère avance sur leurs tâches. Il ne faut surtout pas se reposer sur cette avance mais plutôt garder le même rythme afin de prendre de l'avance sur les prochains jeux qui sont plus conséquents.

5 Bilan de sprint 3

5.1 Avancement global

Ce sprint a été marqué par le début du 3ème jeu, Glenmore, qui a été fini de manière fonctionnelle : il manquait la gestion de certaines tuiles spéciales et le visuel restait compréhensible mais sommaire.

Cette fois aussi, une bonne analyse préalable des entités et service a permis de directement se lancer dans le code de Glenmore sans être perdu.

Durant ce sprint, quelques bugs ont été découverts mais rapidement corrigés sur Splendor, cela n'a pas affecté les délais de livraison.

5.2 Communication d'équipe et qualité de travail

Cette fois ci, la communication restait bonne mais le manque de réunion en présentiel, dû aux vacances, s'est fait ressentir pouvant amener à un manque de confiance en soi. Il est important que chaque membre du groupe travaille sur cette confiance, car un manque de confiance pourrait amener à des tâches faites partiellement ou en retard.

Il est aussi important que chaque membre gagne en autonomie sur leurs tâches. En effet, l'aide à la montée en compétences est importante mais ne doit pas être au détriment de la personne qui aide. Il ne faut donc pas hésiter à se lancer et faire des petites erreurs rattrapables mais apprendre, plutôt que de se bloquer, ne rien faire et entraîner un plus gros retard.

5.3 Périmètre du travail

A contrario, il est aussi important de se limiter quant aux prises d'initiatives sur certaines tâches non prévues à l'origine dans la Spécification Technique du Besoin. Vouloir bien faire est un atout mais il ne faut pas que ce soit au détriment des tâches bien définies et programmées au long du sprint.

6 Bilan de sprint 4

6.1 Avancement global

Cette fois-ci, le 3ème jeu, Glenmore, a été totalement fini et complété, notamment pour l'ensemble des tuiles spéciales. Comme prévu, les animations de Glenmore ont également été réalisées, ainsi que des notifications permettant à l'utilisateur d'avoir des informations en temps réel et de manière agréable et fluide. Il est cependant bon de noter, que suite aux retours du client, concernant la possibilité d'inverser plateaux principal et personnel, certains développeurs, notamment front, ont dû travailler sur Glenmore un peu plus longtemps que prévu. Mais cela n'a pas affecté nos délais de livraison grâce aux tâches tampons préalablement prévues.

La majeure partie du sprint, comme annoncé sur le Plan De Développement, a été consacrée au développement du 4ème et dernier jeu, Myrmes. Cette fois encore, une analyse préalable solide a permis de se lancer directement dans le code, notamment côté back-end. Côté front-end, suite aux maquettes proposées au client et aux retours positifs de ce dernier, les développeurs ont su dans quelle direction aller.

6.2 Communication d'équipe

L'ambiance générale de l'équipe est toujours au beau fixe et la communication reste bonne. La reprise des réunions en présentiel a refacilité les échanges.

Cependant, comme il a été rappelé lors de la rétrospective de sprint, il est important que chacun comprenne sa tâche dans son ensemble et notamment dans son périmètre. Une bonne connaissance de ses tâches permet de ne pas devoir revenir compléter un code incomplet et donc de ne pas perdre de temps.

6.3 Communication avec l'équipe plate-forme

Lors de notre dernière réunion avec l'équipe plate-forme, nous avons pu nous entretenir avec eux sur certains points et notamment sur la communication. En effet, ces derniers rencontraient des points de blocage sur certains aspects. Nous leur avons rappelé de ne pas hésiter à nous solliciter, mais en revanche, de ne pas toujours demander à un membre de notre équipe en particulier mais plutôt à l'ensemble.

Suite à cela, la communication en a été améliorée et les points de blocages résolus. Il est maintenant possible de lancer un jeu depuis la plate-forme, sans devoir passer par notre boîte noire.

7 Bilan de sprint 5

7.1 Avancement global

L'ensemble des fonctionnalités annoncées lors du sprint a été réalisé par l'équipe, toutefois la gestion des objectifs de Myrmes ainsi que la phase d'atelier a été étalée sur les deux sprints afin de finaliser l'ensemble

(en front). Le client s'est vu présenté les différents rapports notamment SonarQube qui constituent un point à améliorer lors du sprint 6 d'un point de vue des issues.

Le projet touche à sa fin d'un point de vue fonctionnalités et le client est satisfait de ce que l'équipe a produit, ce qui a grandement motivé les membres de l'équipe qui en ont fait part lors de la rétrospective.

De manière générale, le team building aide à l'entente de l'équipe malgré les quelques dissensions.

7.2 Communication d'équipe

Il est toutefois à noter que l'indépendance n'est pas acquise par tous les membres de l'équipe, qui dépendent encore beaucoup de certaines personnes pour l'avancée de leur tâche (notamment par manque de confiance en soi, qui reste un point important).

De même, il est important que tous les membres apprennent à mieux gérer leur temps et ne pas faire au dernier moment leurs fonctionnalités au risque de bloquer les personnes qui dépendent d'elles.

Quelques conflits ont éclaté en dehors du projet qui ont pu avoir des répercussions sur la communication, toutefois ces effets ont été gérés.

Il est donc important que les développeurs prennent davantage confiance en eux et prennent des initiatives, sans pour autant ne plus communiquer et travailler seul dans son coin.

7.3 Technologie

D'un point de vue technologie, les développeurs ont appris à maîtriser leurs outils de développement, et certains ce sont même essayés aux technologies qu'ils connaissaient moins (des développeurs front s'essayant au back et phpunit, des développeurs back à Twig).

Un point toutefois à améliorer serait la connaissance de Jira, qui reste un point handicapant de la gestion de projet d'un point de vue génération de rapport. Les développeurs saisissent toutefois correctement leur temps sur Jira à présent et les tickets sont mis à jour plus régulièrement.

7.4 Qualité de code

La couverture de test reste assez basse d'après les rapports et de nombreux issues (même s'ils sont rapidement corrigables) restent soulevés, ce qui endommage la qualité de code.

Toutefois, les tests sont en passe d'amélioration et la couverture augmente drastiquement, de même les issues sont en cours de résolution et la factorisation aide à cela.

7.5 Qualité de travail

Les tendances remarquées ont commencé à évoluer mais reste à poursuivre, notamment sur l'indépendance et la prise de responsabilité. Ce sont des qualités qui seront acquises avec le temps et dépendamment de chacun, mais qui ne doivent pas entraver le travail de groupe.

8 Bilan de sprint 6

8.1 Avancement global

La majorité des fonctionnalités et User stories ont pu être réalisées au cours de ce sprint. Le client a pu valider les animations de Myrmes, la possibilité d'exclure un joueur, la phase ouvrière et atelier ainsi que le placement automatique des nourrices durant la phase naissance. Le système d'annulation de coups, à travers une pile, dans la phase ouvrière a également été approuvé par le client. Il ne reste donc plus à, l'équipe, que de finaliser la gestion des objectifs qui est presque finie au moment de l'écriture de ce document.

La couverture a augmenté jusqu'à 78% pour PHPUnit et 65% pour SonarQube. Le nombre d'issues a baissé et l'équipe est toujours en cours de travail dessus.

Le client a encore une fois pu montrer son enthousiasme pour le travail fourni, ce qui, avec le team building, a contribué à l'entente du groupe.

8.2 Communication d'équipe

La confiance en soi a bien progressé, notamment pour certains membres du groupe mais n'est pas assez suffisante pour le moment. Celle-ci peut amener à quelques blocages mais rapidement résolus.

La communication entre chaque membre du groupe était plus fluide et dynamique, ceci a permis à chacun de s'entraider.

8.3 Technologie

Concernant l'utilisation de Jira, les développeurs ont pris pour habitude de mettre à jour leurs tickets régulièrement et de surtout saisir leurs temps. En revanche, la génération de rapport (burndown chart) pose toujours souci, malgré les efforts d'auto-formation et de correctement saisir les temps.

D'un point de vue langages et technologies, chaque membre maîtrise sa spécialité et la technologie dans laquelle il intervient. Cela permet d'effectuer les tâches de plus en plus rapidement.

8.4 Qualité de code

La couverture de test a clairement augmenté et est toujours en voie d'amélioration. Les issues sont rapidement traitées et il ne reste que des issues rapidement corrigeables.

Comme demandé par le client et pour faciliter l'implantation de nouveaux jeux pour les futurs intervenants, la factorisation a bien débuté et nous avons pu regrouper des notions communes dans des fichiers communs à tous les jeux.

Divers outils d'analyse statique de code ont pu être mis en place comme PHP_CodeSniffer ou TwigLint (pour le front). Ces deux outils rejoignent PHPStan précédemment installé et permettent à l'équipe d'améliorer drastiquement la qualité du code.

8.5 Qualité de travail

Les anciennes remarques ont été prises en compte mais il faut faire attention, pour chaque développeur, de bien relire les règles des jeux et documents préalablement rédigés. En effet, il ne faut pas oublier certaines parties et pour cela, il faut bien réfléchir à la tâche qui nous est confiée et ce qu'elle implique.

9 Bilan de sprint 7 (correctif)

9.1 Avancement global

Toutes les fonctionnalités et User stories restantes ont été réalisées au cours du sprint. A l'issue du sprint review, le client a pu nous valider les fonctionnalités restantes, ce qui marque donc la fin du sprint mais aussi la fin du projet pour la phase de développement. L'équipe en charge des jeux aura donc réussi et sera allée au terme du projet.

9.2 Communication d'équipe

Même si la confiance en soi des membres du groupe est en légère hausse, elle n'est pas encore suffisante chez certains et peut pénaliser le groupe. Le distanciel et la charge de travail autour avec les autres projets et examens n'ont pas facilité la communication avec certains membres qui aurait pu être meilleure.

9.3 Technologie

D'un point de vue langages, chaque développeur maîtrise sa technologie et sa partie du projet ce qui fluidifie le développement.

Cependant, l'utilisation de Jira pose toujours souci, notamment du côté de la génération des diagrammes d'avancement. Nous nous sommes donc basés sur le diagramme de flux cumulés qui nous permet de voir l'évolution de l'état de nos tickets. Cela nous permet d'avoir un contrôle et un état sur notre dynamique.

9.4 Qualité de code

Les divers outils d'analyse statique ainsi que SonarQube ont permis de réduire le nombre d'issues et donc d'améliorer la qualité du code.

Du point de vue couverture de tests, nous sommes arrivés à 70% de couverture d'après PHPUnit après fusion du code avec l'équipe plateforme. Cela s'explique par notre stratégie de risk-based testing, c'est-à-dire de trier les fonctionnalités les plus à risque de créer des problèmes et de les tester en priorité dans un souci de temps.

Concernant la factorisation, elle a été réfléchie et les idées ont été consignées dans des documents réservés aux futurs intervenants. Pour notre part, il n'aurait pas été préférable de retoucher à toutes les fonctionnalités de tous les jeux seulement une semaine ou deux avant le rendu final. De plus, il est important de privilégier un code maintenable plutôt qu'un code parfaitement factorisé.

9.5 Qualité de travail

Le manque de confiance s'accompagne d'un manque d'indépendance pour certains développeurs. Il faut davantage prendre ses responsabilités et oser faire les choses sans attendre qu'on nous dise de le faire, par peur de mal faire. Cela favorisera grandement la prise de confiance personnelle.