

20.06.2022					
Fourchette de temps		Total de temps	Travail accomplis	Sources	Appréciations
09:00	10:00	01:00	Solance de début de TPI. Signature du cahier des charges et briefing sur ce qui devra être fait dans celui-ci. J'ai posé des questions sur ce qui devra être fait au niveau des tests et au niveau de ce qui est requis au niveau du Frontend ainsi que quelques autres informations au niveau de la présentation du rapport du TPI et de ce qui devra être précisé dedans.	N/A	Tout les points requis pour le TPI ont été assimilés et compris.
10:00	12:00	02:00	Création du planning sur gantit via le site web <i>TeamGantit</i> . J'ai posé quelques questions au niveau de la réalisation du Gantit au chef de projet. Particulièrement au niveau de ce qui devait être précisé dedans. Comme par exemple les rendez-vous avec les experts, ou les absences. J'ai dû également changer la répartition du temps pour laisser la place aux activités sus-mentionnées.	https://prod.teamgantit.com/	Mis à part quelques incompréhensions de ce qui devait être fait dans le planning, il n'y a eu aucun soucis pour pouvoir répartir les tâches que je dois faire sur 40h de travail. Le planning est désormais fini excepté si des modifications supplémentaires sont jugées nécessaires.
13:30	14:00	00:30	Création de ce Google Sheet pour créer mon journal de travail. Création de la structure qui va être utilisée pour chaque jours dans le JDT.	https://docs.google.com/	Aucun problème pour la création du Google Sheet.
14:00	15:00	01:00	Recherche pour les différentes technologies qui vont être utilisées durant le projet. La sélection étant faite voici mes choix et ceux qui ont été imposés sur le cahier des charges: Frontend - React.js comme librairie pour faire mon site web, Tailwind CSS pour le style du site. Backend - C#.NET avec Swagger. Base de donnée - MySQL, et comme outil d'administration phpmyadmin afin que la base puisse être facilement accessible pour une démo du projet. bien que je vais utiliser Datalogrip de Jetrains pour la gestion de la base durant le développement. Devops/Environnement - Docker, toutes les parties sus-mentionnées seront mis dans des containers docker.	https://reactjs.org/ https://tailwindcss.com/ https://dotnet.microsoft.com/en-us/ https://swagger.io/ https://www.mysql.com/ https://www.phpmyadmin.net/ https://www.jetbrains.com/datalogrip/ https://www.docker.com/	La majorité des technologies & langages présentés ici ont été déjà été utilisées tout au long de mon apprentissage, et pourront ainsi faciliter le travail que j'aurai à faire durant ce mini-TPI. Je n'ai pas eu de difficulté pour la sélection de celles-ci, au contraire.
15:00	16:00	01:00	Recherches pour le déploiement d'un backend C#.NET avec Docker. N'ayant pas encore pratiqué ceci auparavant. Ça a été faisable et je ne pense pas, encore du moins, que ça va retarder le temps marqué sur le planning.	https://github.com/microsoft/dotnet-framework-docker/blob/main/samples/README.md https://hub.docker.com/_/microsoft-dotnet-framework https://docs.docker.com/samples/dotnetcore https://stackoverflow.com/questions/11244653/what-month-and-day-parameters-describe-an-unrepresentable-datetime-exception	Rien à vraiment rajouter, j'ai quand même quelques inquiétudes vu que je n'ai pas encore fait ça avant de tomber dans des problème qui vont me retarder, même si ça n'a pas l'air d'être trop compliqué.
16:00	17:00	01:00	Création et préparation du repository, et de l'environnement de développement sur Github. Création d'une orga pour héberger un repository documentation et un repository avec le code.	https://github.com/Mini-TPI-Jaav/Ex https://github.com/Mini-TPI-Jaav/Ex/SmallTPI https://github.com/Mini-TPI-Jaav/Ex/SmallTPI/tree/feature/database	J'ai décidé, enfin de prendre de l'avance, de déjà créer les repository et l'organisation pour la suite. Je pense que ceci me permettra de prendre un peu d'avance sur le planning initial.
17:00	18:00	01:00	Création du Kanban où j'ai pu transposer toutes les tâches à faire qui étaient dans le planning dans le Github Project de l'organisation cité ci-dessus. J'ai également mis des storypoints sur chaque issues afin de suivre une méthode de projet en Scrum. Il y a un total de 126 Storypoints dans le projet et la vélocité sera évaluée chaque jour après avoir fini les issues de la journée.	https://github.com/orgs/Mini-TPI-Jaav/Ex/projects/1/views/2	Il y a encore bien sûr encore des choses à rajouter dedans. Mais il s'agit de choses spécifique qu'il faudra que je remplisse en cours de route.
18:00	19:00	01:00	Remplissage du Rapport et du Journal de travail en fin de journée + Envoi du mail avec les documents respectifs et mise à jour du repository avec les documents cités	N/A	Aucun problème pour la réalisation du Kanban. Rien à commenter.
TOTAL DE TEMPS				Bilan de la journée	
08:30				Pour la première journée de ce mini-TPI tout s'est bien passé et ai même pris pas mal d'avance sur la conception du planning ce qui m'a permis de pouvoir avancer sur d'autres parties du projet qui vont grandement me faire gagner de l'avance durant les prochains jours du mini-TPI.	

21.06.2022					
Fourchette de temps		Total de temps	Travail accomplis	Sources	Appréciations
09:00	10:00	01:00	Résolutions de 3 issues dans mon projet. Création d'un fichier Docker compose avec lequel je déploie un conteneur MySQL et un conteneur phpmyadmin. J'ai également fait en sorte qu'au démarrage le conteneur MySQL charge le fichier dump qui a été fourni pour ce TPI. J'ai également dû fixer le dump en question car celui-ci faisait planter mon conteneur dû à la première ligne du fichier, celle-ci fait un drop de database sans vérifier si la base de données existe au préalable, il a donc été facile de fixer ça avec une condition "IF EXISTS"	https://github.com/orgs/Mini-TPI-Jaav/Ex/projects/1/views/2 https://github.com/Mini-TPI-Jaav/Ex/SmallTPI/tree/feature/database https://github.com/conservius/kata-manga/blob/master/import/data/KataManga_structure_and_data.sql	Ayant déjà fait ceci durant de précédent projets je n'ai eu aucune réelle difficulté à effectuer cette tâche. Même avec la problématique du drop database, j'ai réussi très rapidement à trouver le problème et le fixer.
10:00	10:15	00:15	Remplissage du journal de travail pour la tâche citée précédemment	N/A	J'ai pris 1h d'avance sur le planning sur cette partie.
10:15	11:00	00:45	J'ai accomplis le setup initial de mon projet en C#, en y ajoutant un Swagger intégré. J'ai également ajouté un .gitignore pour éviter de commit des fichier binaires.	N/A	Rien à commenter.
11:00	11:30	00:30	Absence pour aller faire un test covid.	N/A	Rien à commenter.
13:00	16:30	03:30	Création de modèles pour mon backend C#. La raison que j'ai pris autant de temps dessus est parce que je n'ai pas voulu bêtement créer les modèles et les commits, mais j'ai voulu les tester en faisant un contexte et un contrôleur de test. Malheureusement dû à peut être un manque d'expérience concernant la liaison entre un serveur MySQL et un backend C#, je suis tombé dans plein de problèmes. Typiquement avec les formats des dates. MySQL accepte les dates zéro. C'est à dire qu'une date à l'an 0 ou le mois 0 ou encore le jour 0 peut être crée sans problèmes. Cependant avec le système de typages de C# ceci est incompatible, et j'ai donc dû faire beaucoup de recherches pour résoudre ce problème. J'ai également appelé mon chef de projet (Nicolas Borboen) afin de m'aider à résoudre cette problématique. Le problème a enfin été fixé et je suis capable de faire un simple GET des mangas de ma base de donnée.	https://stackoverflow.com/questions/2934844/unable-to-convert-mysql-date-time-value-to-system-datetime https://stackoverflow.com/questions/11244653/what-month-and-day-parameters-describe-an-unrepresentable-datetime-exception https://github.com/Mini-TPI-Jaav/Ex/SmallTPI/tree/feature/backend https://github.com/Mini-TPI-Jaav/Ex/SmallTPI/tree/feature/backend/kata-manga-backend/kata-manga-backend/Models	J'ai malheureusement perdu beaucoup de temps sur cette tâche là, ce qui va potentiellement m'handicapper durant la suite du projet. Je pense cependant que je peux essayer de rattraper cela assez vite si rien d'autre me retarde comme ceci.
16:30	19:00	02:30	Création de contrôleurs pour mon projet. Malheureusement ceci n'est pas encore fini et est encore qu'à 50% fonctionnel. J'ai perdu beaucoup trop de temps sur la tâche d'avant. Pour l'instant je suis capable de récupérer toute les données qui sont sur chaque table et bientôt également de pouvoir en rajouter après avoir réglé un problème d'auto-incrément sur le dump de la base MySQL. J'ai également crée le context nécessaire pour pouvoir faire les requêtes dans les contrôleurs bien que celui-ci n'est pas complet à 100%.	https://github.com/Mini-TPI-Jaav/Ex/SmallTPI/tree/feature/backend/kata-manga-backend/kata-manga-backend/Controllers	J'ai également ici perdu pas mal de temps par rapport à ce qui était prévu, j'ai quelques inquiétudes au niveau du Backend et vais peut-être devoir à un moment commencer à faire le Frontend afin d'avancer le projet si la conception du Backend prend trop de temps.
19:00	20:00	01:00	Remplissage du Rapport et du Journal de travail en fin de journée + Envoi du mail avec les documents respectifs et mise à jour du repository avec les documents cités	N/A	Je suis un peu déçu d'être tombé dans autant de problèmes techniques avec le dump de la base MySQL (Tel que le problème d'auto-incrément et de date à 0) et avec des hésitations dû à un manque de pratique en développement Backend par rapport au Frontend. Rien à commenter.
TOTAL DE TEMPS				Bilan de la journée	
09:30				Aujourd'hui fut malheureusement remplis de pleins de retards et d'échecs. Je pense que j'aurais dû demander de l'aide plus tôt pour les problèmes qui sont arrivés durant la création des modèles. Ce qui m'aurait permis de perdre beaucoup moins de temps et ensuite me concentrer sur la partie des contrôleurs. Me permettant de finir plus ou moins les tâches comme prévues dans le planning. Retrospective - A la fin de ce mini sprint j'arrive à une vélocité de 32 Storypoints pour un total restant (nombres de storypoints mis à jour) de 120 Storypoints. Me donnant une fin estimée du projet dans 3.75 jours de travail. Me montrant qu'un retard estimé de 0.75 jour s'est effectivement crée durant cette journée.	