

Plan

Week 41

- Etat de l'art afin d'effectuer des choix technologiques cohérents
- Initialisation du projet (GitFlow/Kanban)

Week 42

- Produit minimum viable avec un utilisateur capable de payer avec une carte
- Mise en place des tests de charge basiques sur la quantité de paiement pour un utilisateur, tout en se concentrant sur le fait que ces tests doivent nous servir pour construire la suite du projet.
- Mise en place de CI ainsi que Docker.

Week 43

- Ajout des comptes épargnes (avec intérêts) et de la gestion des plafonds glissants
- Ajout de plusieurs utilisateurs pour tester l'aspect stateless
- Tests de charges avec plusieurs utilisateurs afin mettre en avant les éventuelles problèmes de concurrence que cela pourrait causer.
- Mise en place de nos différents composants dans un orchestrateur de conteneurs afin d'essayer de voir comment notre système réagit à la scalabilité horizontale.

Week 44

- Tests de charge sur le déploiement de l'orchestrateur de conteneurs avec plusieurs utilisateurs effectuant diverses actions en même temps.
- Les résultats apportés par cet orchestrateur vont avoir de grosses répercussions sur notre façon de concevoir notre architecture par la suite.

Week 45

- Recherche de bottleneck au niveau CPU et mémoire des différents services et modifications de l'implémentation et/ou de l'architecture en fonction des résultats trouvés.

Week 46

- Simulation de latence réseau afin d'optimiser les échanges entre services et garantir le bon fonctionnement de la solution quelles que soient les conditions d'utilisation.

Week 47-2

Off

Éclaircissement autour des sprints :

Dans tous les cas, au fur et à mesure du développement, nous allons analyser notre solution actuelle, et essayer de voir où peuvent se situer les bottlenecks grâce à des mesures qui seront de plus en plus précises et complètes. Ces mesures nous permettront d'identifier les changements à effectuer sur notre architecture.