



Université Catholique de Lille
Faculté de Gestion, Économie & Sciences
École du Numérique



Projet de Conception de Logiciel

Planif

Sprint 5 - Optimization

(04 Décembre 2024 - 17 Décembre 2024)

Présenté par:

Akil Wael
Aliligali Amir
Hebbinckuys Hugo
Kilito Yazid

Présenté à:

Mousin Lucien

Décembre 2024

1. Tâches

1.1. Analyse des Goulots d'Étranglement

- Responsable: Tous les membres de l'équipe
- Durée: 2 jours
- Description: Identifier les étapes du processus qui ralentissent l'exécution du bot, telles que le traitement vidéo ou l'extraction des clips. Les tests de performance précédents seront utilisés pour déterminer les zones de faible performance. Des outils de profiling seront utilisés pour évaluer l'impact des différentes étapes sur le temps d'exécution et la consommation de ressources.

1.2. Optimisation du Traitement Vidéo

- Responsable: Akil Wael
- Durée: 3 jours
- Description: Améliorer la vitesse de conversion des vidéos au format TikTok et optimiser le rendu des transitions vidéo. Cela inclut l'amélioration des algorithmes de compression, l'optimisation des bibliothèques utilisées, et la gestion de la mémoire pendant le traitement vidéo pour réduire les coûts en ressources.

1.3. Optimisation de l'Extraction des Clips

- Responsable: Kilito Yazid
- Durée: 3 jours
- Description: Améliorer la récupération des clips via l'API Twitch et le scraping. Cela implique la mise en cache des résultats et l'optimisation des filtres de recherche (par durée, vues, date), ainsi que l'amélioration de l'intégration des paramètres pour minimiser le nombre de requêtes API.

1.4. Optimisation de l'Intégration avec TikTok

- Responsable: Aliligali Amir
- Durée: 2 jours
- Description: Optimiser la vitesse de téléchargement des vidéos sur TikTok et garantir une meilleure gestion des erreurs d'intégration. Cela inclut la gestion des API rate limits et la réduction des risques d'échec des uploads grâce à des mécanismes de reprise automatique en cas d'échec.

1.5. Réduction de la Consommation des Ressources

- Responsable: Hebbinckuys Hugo
- Durée: 2 jours

- Description: Optimiser la consommation des ressources, notamment la RAM et le CPU, pendant l'exécution du bot. Les solutions envisagées incluent l'utilisation de threads ou de processus légers pour gérer les tâches de manière asynchrone et réduire le temps d'attente pendant les opérations.

1.6. Tests d'Optimisation

- Responsables: Tous les membres de l'équipe
- Durée: 1 jour
- Description: Tester la version optimisée du bot pour vérifier les améliorations de performance. Comparer les résultats des tests avant et après optimisation en termes de vitesse d'exécution, consommation de mémoire et fluidité générale du processus.

1.7. Mise à jour de la Documentation Technique

- Responsable: Tous les membres de l'équipe
- Durée: 1 jour
- Description: Mettre à jour la documentation pour refléter les optimisations réalisées. Cela inclut la mise à jour des diagrammes de flux, des descriptions des fonctionnalités modifiées et des explications sur les améliorations de performance.

2. Suivi et Communication

- Réunions régulières sur Discord avec partage d'écran pour suivre l'avancement et résoudre les problèmes.
- Utiliser GitHub pour le contrôle de version. S'assurer que les modifications sont revues et testées avant d'être fusionnées dans la branche principale.
- Documenter tous les problèmes rencontrés avec l'API Twitch ou le scraping web et les prioriser pour leur résolution.