

ORBIT

(Optimal Route Booking Intergalactic Travel)

Compte Rendu d'Activité 1

Niveau 1 : Présentation générale	2
Méthode de travail	2
Avancement général	2
Maquette Figma	2
Niveau 2 : État des modules	3
Module L3/L3bis - Développement web (Thomas FRESSIN)	3
Module L4 - Développement efficace (Laurent BULTEAU)	3
Module L2 - Développement Java/JDBC (Michel CHILOWICZ)	3
Module L1 - Algorithmique sur les graphes (Didier CAUCAL)	3
Niveau 3 : Analyse détaillée	4
Progrès et Difficultés Actuelles	4
Plan d'Action Prioritaire	4
Amélioration de l'Organisation	5
Vision à Long Terme	5
Annexes	6
Annexe 1 : Diagramme de Gantt	6
Annexe 2 : Figma en image (capture d'écran)	7

Niveau 1 : Présentation générale

Méthode de travail

Nous travaillons en méthode Scrum, utilisant YouTrack pour le suivi des tickets de développement, avec un diagramme de Gantt comme support pour planifier et organiser le projet. Ces outils nous permettent de visualiser les étapes clés et d'assurer un suivi rigoureux.

Voir [Annexe 1](#) (Diagramme de Gantt).

Avancement général

Responsable	Tâches principales	Avancement
Julien	Carte dynamique, formulaire de recherche avancé, scripts backend	70%
Abdelrahim	Front-end (interface graphique, login, navigation)	70%
Alexis	Graphes, algo C, mise en relation avec JDBC	50%

Points forts :

- Style visuel et interface utilisateur presque finalisés.
- Algorithmes de graphe fiables prêts à être intégrés.

Prochaines étapes critiques :

- Intégration de la carte dynamique et du formulaire avancé dans l'application.
- Finalisation du développement Java (JDBC) pour la gestion des graphes et la liaison avec la base de données.

Maquette Figma

Voir [Annexe 2](#) (Maquette Figma en image) ou en y accédant par ce lien :

<https://www.figma.com/design/BNL3KpfxPlodWwheyR56pb/Maquette-site?node-id=0-1&t=n4in2a6PpfutuobR-1>

Niveau 2 : État des modules

Module L3/L3bis - Développement web (Thomas FRESSIN)

Avancement réalisé :

- Développement de la carte dynamique avec les planètes.
- Création d'un formulaire de recherche avancé des trajets.
- Scripts backend pour les calculs de distances et interactions avec les données des planètes.

Travaux restants :

- Intégration de la carte et du formulaire à l'application principale (frontend d'Abdel).

Module L4 - Développement efficace (Laurent BULTEAU)

Avancement réalisé :

- Algorithmes de manipulation de graphes développés en C (Dijkstra, A*).
- Structures fiables pour l'exploration des plus courts chemins.

Travaux restants :

- Liaison avec le backend Java via des fichiers JSON pour la compatibilité avec JDBC.

Module L2 - Développement Java/JDBC (Michel CHILOWICZ)

Avancement réalisé :

- Début d'apprentissage et tests statiques avec JDBC.

Travaux restants :

- Mise en relation des données stockées en base avec les graphes en C.
- Implémentation de l'export JSON pour connecter les graphes et l'interface.

Module L1 - Algorithmique sur les graphes (Didier CAUCAL)

Avancement réalisé :

- Tous les algorithmes théoriques pour la gestion des graphes ont été implémentés.

Travaux restants :

- Mise en relation entre les données de la base et les structures algorithmiques.

Niveau 3 : Analyse détaillée

Le projet avance bien dans l'ensemble, mais certains problèmes techniques ralentissent la progression. Ces blocages concernent principalement l'interconnexion entre les différentes parties (Java, C, et base de données), le manque de maîtrise sur certains outils, et des contraintes liées à une petite équipe. Pour avancer, il faudra renforcer les compétences techniques, prioriser les tâches, et intégrer progressivement les modules pour éviter de rester bloqués.

Progrès et Difficultés Actuelles

Le module **JDBC** est essentiel puisqu'il gère les échanges entre Java et la base de données. Cependant, nous avons encore du mal avec les transactions complexes et les relations entre tables, ce qui complique son utilisation.

Un autre problème concerne la liaison entre les **algorithmes de graphes**, le C, Java et la base de données. Ces algorithmes fonctionnent bien seuls, mais ils n'utilisent pas encore les données réelles de la base, ce qui freine les tests en conditions réelles.

Enfin, avec seulement trois membres dans l'équipe, chacun se concentre sur une technologie précise. Si cela permet de se spécialiser, il en est aussi question d'un problème : nous sommes dépendants du travail des autres pour avancer, et nous avons peu de marge pour travailler en parallèle. Même avec des outils comme YouTrack et la méthode Scrum appliquée, les délais sont restreints.

Plan d'Action Prioritaire

Pour sortir de cette situation, il faut d'abord que nous montons en compétences sur JDBC, et cela passera par des exercices pratiques.

En parallèle, nous devons aussi collaborer pour intégrer la carte dynamique et le formulaire avancé au frontend. L'objectif est de synchroniser les données en temps réel entre l'utilisateur et la base. Une fois cette partie intégrée, nous pourrons enfin tester une version complète du système pour recueillir les premiers retours.

Les algorithmes de graphes, eux, doivent commencer à utiliser des données réelles issues de la base. Pour cela, il faudra automatiser l'export des données via JDBC, puis les faire passer dans les modules en C. Une fois ces étapes franchies, des tests avec des scénarios réalistes pourront valider leur fonctionnement ou mettre en lumière des ajustements nécessaires.

Amélioration de l'Organisation

Pour éviter d'être bloqués à cause des dépendances, nous devons documenter clairement comment les modules communiquent entre eux, en particulier les API. Des réunions plus régulières (type Scrum) peuvent aider à ajuster les priorités et détecter les problèmes avant qu'ils ne deviennent critiques. L'objectif est que tout soit clair et accessible, pour que chaque membre puisse avancer même en cas d'absence ou de difficulté de l'un d'entre nous.

La communication entre membres doit aussi être renforcée. Même dans une petite équipe, des points réguliers pour partager les avancées et échanger sur les problèmes rencontrés permettront de mieux répartir les efforts et d'éviter une surcharge inutile pour certains.

Vision à Long Terme

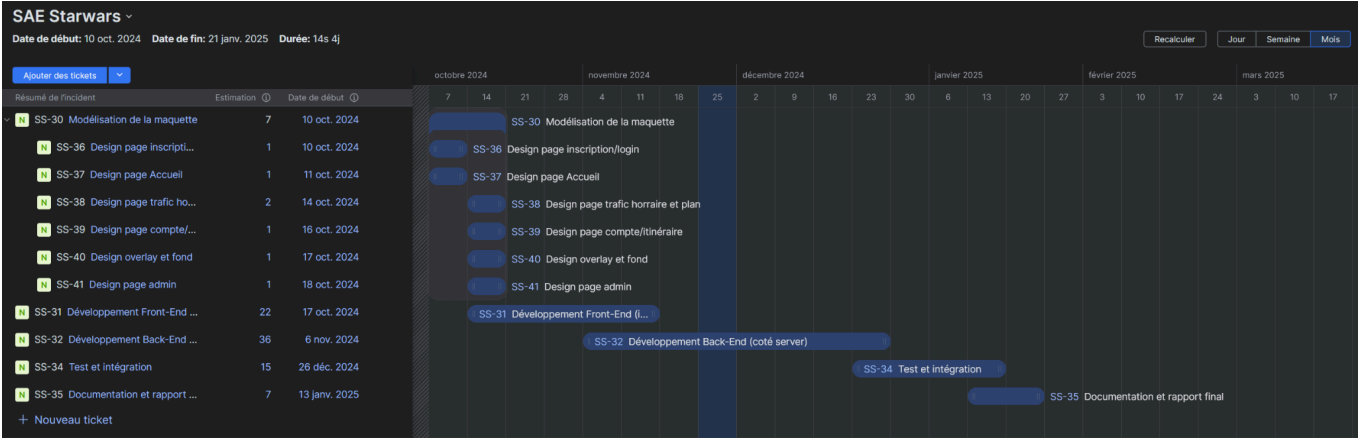
Avant d'envisager des optimisations ou des nouvelles fonctionnalités, il faut s'assurer que tout fonctionne correctement et que les bases techniques sont solides. Une fois les modules bien reliés et testés, nous pourrons enrichir l'application avec des outils ou interactions plus poussés pour améliorer l'expérience utilisateur.

Au-delà du projet lui-même, cette expérience est aussi une chance pour chacun d'apprendre. Gagner en compétences sur des technologies comme JDBC ou sur la communication entre Java et C sera utile pour d'autres projets à venir. Si l'on combine une bonne organisation, de l'entraide et une approche méthodique, nous pourrons non seulement réussir ce projet, mais aussi mieux gérer les prochains.

En résumé, nous devons garder une approche simple : résoudre les blocages techniques, avancer étape par étape et rester alignés sur les priorités. Avec ça, le projet est sur la bonne voie pour atteindre ses objectifs.

Annexes

Annexe 1 : Diagramme de Gantt



Annexe 2 : Figma en image (capture d'écran)

