# Plan d'action du ticket #1664

## L'équipe

* **Alice :** Moi-même, touche à tout avec 5 ans d'expérience
* **Bob :** Lead dev, excellent tech, 10 ans d'expérience, peu de disponibilité
* **Cindy :** UX designer avec 2 ans d'expérience
* **David :**  QA avec 3 ans d'expérience
* **Eddy :**  Alternant M2STLRES

## Plan d'action - Backlog

| ID | Nom | Description | Temps (en journée) | Epic | Priorité |
| --- | --- | --- | --- | --- | --- |
| T1E1 | Choix des technologies | Choisir les différentes technologies utilisées pour le projet | 1/2 Journée | Architecture | Haut |
| T2E1 | Création de la maquette | Créer la maquette des différentes pages. Le projet a une UI très simple | 1/2 Journée | Architecture | Moyen |
| T3E1 | Créer les environnements | Une fois les technologies choisis créer les différent environnement de développement et de préproduction | 1 Journée | Architecture | Haut |
| T4E2 | Récolte des données | Récolter les URL d'image pour populer la DB | 1/2 Journée | Data et stockage | Moyen |
| T5E3 | Fonctionnalité "Recherche" | Permet de rechercher des URL par mot-clef exact | 1 Journée | Recherche Simple et classement | Moyen |
| T6E3 | Test de la fonctionnalité "Recherche" | Test de la fonctionnalité "Recherche" | 1/2 Journée | Recherche Simple et classement | Haut |
| T7E4 | Créer le système de parser de regex | Création d'un système de parsing de regex pour pouvoir lire efficacement les regex | 3 Journées | Recherche avancée | Moyen |
| T8E4 | Fonctionnalité de "Recherche avancée" | Permet de rechercher des URL par regex en utilisant le parser créé précédemment | 2 Journées | Recherche avancée | Moyen |
| T9E4 | test de la "recherche avancée" | Test de la recherche avancée | 1 Journée | Recherche avancée | Haut |
| T10E3 | Choix des systèmes de classements proposés | Choisir le ou les systèmes de classements qui vont être proposé à l'utilisateur. | 1/2 Journée | Recherche Simple et classement | Haut |
| T11E3 | Implémentation des systèmes de classement | Implémentation du ou des systèmes de classements choisis précédemment | 1 Journée | Recherche Simple et classement | Moyen |
| T12E3 | Test du système de classement | Test du système de classement implémenté | 1 Journée | Recherche Simple et classement | Haut |
| T13E5 | Fonctionnalité de "recherche multiple" | Implémentation de la recherche multiple permettant de rechercher avec plusieurs mots | 5 Journées | Recherche multiple | Haut |
| T14E5 | Test de la fonctionnalité "recherche multiple" | Test de la fonctionnalité de la recherche multiple | 1 journée | Recherche multiple | Haut |
| T15E6 | Implémentation de l'UI | Implémentation de l'IU en suivant la maquette précédemment fixée | 1 Journées | UI | Bas |
| T16E6 | Test de l'UI | Test de l'UI | 1 Journée | UI | Bas |
| T17E7 | Écriture de la documentation de l'API | Écrire la documentation de l'API pour permettre a de futur développeurs | 2 Journées | Documentation | Moyen |
| T18E8 | Intégration | Intégrations des différents éléments | 3 Journées | Intégration | Haut |
| T19E8 | Test d'intégration | Test de l'intégration | 1 Journée | Intégration | Haut |

### Liste des EPIC

* E1 - Architecture (2 J)
* E2 - Data et stockage (1/2 J)
* E3 - Recherche Simple et classement (4 J)
* E4 - Recherche avancée (6 J)
* E5 - Recherche multiple (6 J)
* E6 - UI (2 J)
* E7 - Documentation (2 J)
* E8 - Intégration (4 J)

## Diagrammes

### EPIC

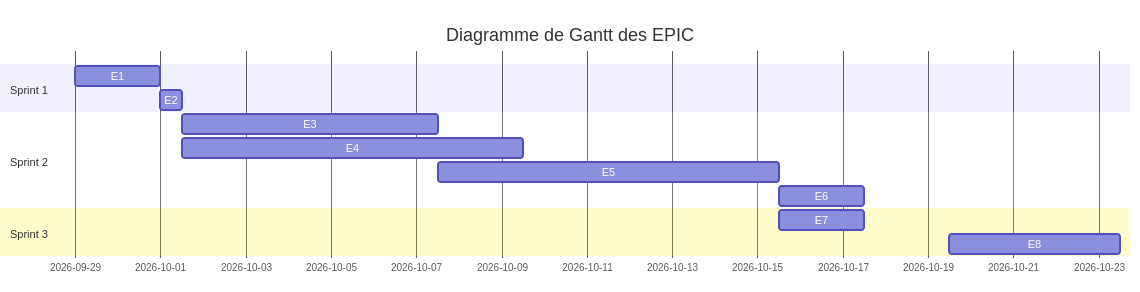


Figure :

* **Sprint 1 - Conception et préparation**

On commence par l'Epic 1 pour mettre au point l'architecture du projet et choisir les technologies qui vont être utilisées pour le projet. On poursuit en remplissant la base de donnée et créer les différents environnements.

* **Sprint 2 - Développements**

On va développer les différentes fonctionnalités du projet en essayant de les paralléliser au possible. Une fois l'ensemble terminer on passe à l'UI. On peut se permettre d'attendre la fin du développement du backend pour commencer l'UI car ce dernier est assez court à développer.

* **Sprint 3 - Intégration et documentation**

On va créer la documentation de l'API en parallèle de l'UI puis se lancer dans l'intégration de l'UI et du backend et ses tests avant la livraison finale du POC.

### RH

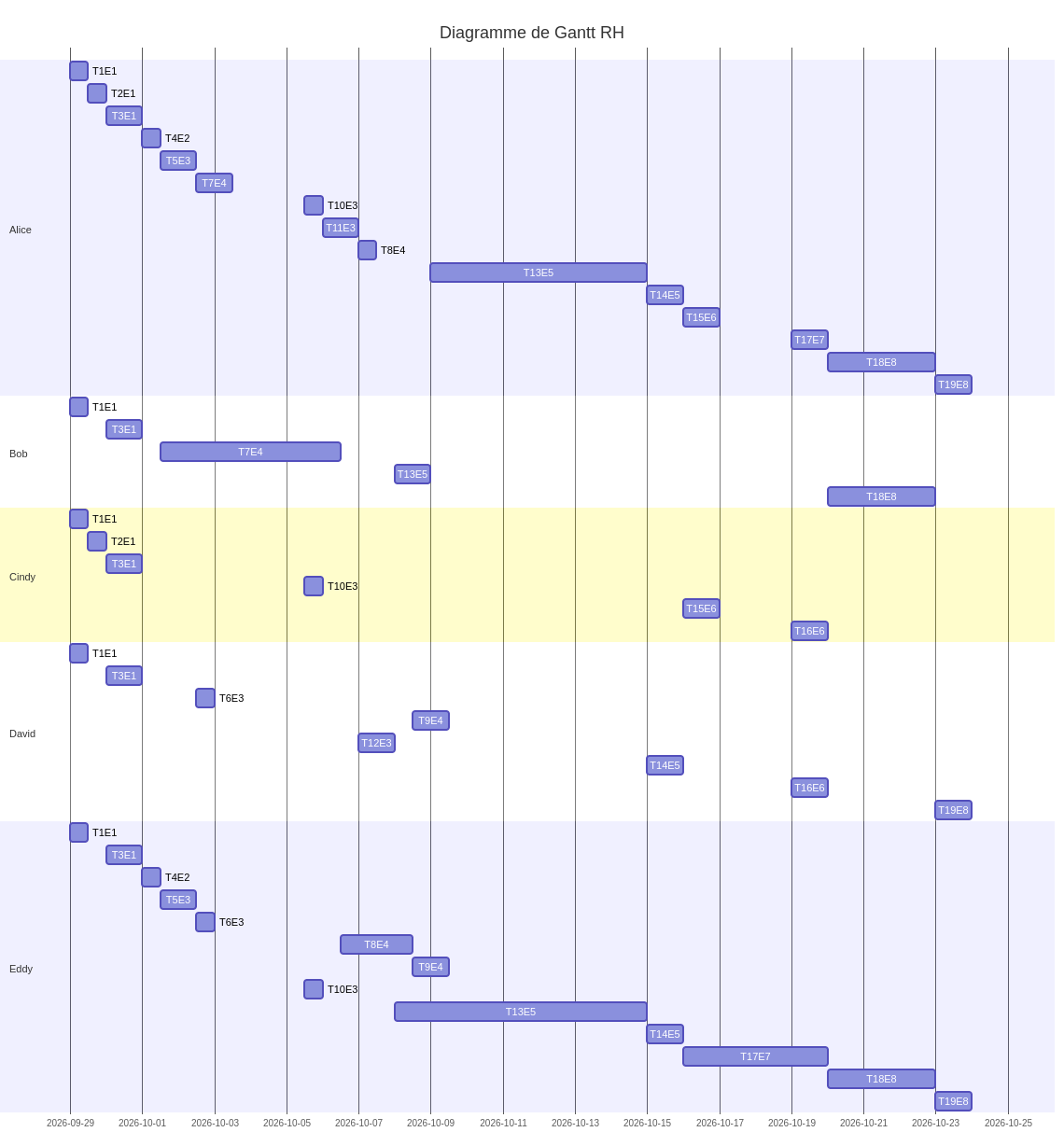


Figure :

* **Alice**

En tant que chef de projet elle intervient sur la quasi intégralité du projet et a une vision sur toute les tâches.

* **Bob**

Etant très occupé par d'autre part mais particulièrement efficace on lui fait débuter les parties compliquée du projet pour que d'autre puissent ensuite terminer.

* **Cindy**

Créer la maquette de l'UI et participe au choix du classement des réponses pour que ce dernier soit le plus pratique possible pour l'utilisateur. Elle va aussi implémenter le front end car ce dernier est suffisamment simple pour ne pas nécessité d'expert.

* **David**

Va tester toutes les fonctionnalité, souvent assisté d'Alice ou d'Eddy pour lui apprendre comment se font les tests.

* **Eddy**

Participe a la plupart du projet de façon a apprendre un maximum de chose. On lui fait rédiger la documentation pour qu'il ait une vision d'nesemble sur le projet et puisse plus simplement l'intégrer a son mémoire.

## Mail de réponse

Bonjour M.X,

Veuillez trouver ci-joint l'organisation de l'équipe qui m'est allouée pour le développement du moteur de recherche d'image.

J'ai essayé au mieux de concilier les forces de chacun et leurs disponibilités.

Nous pensons pouvoir terminer le projet pour fin Octobre.

Cordialement,  
Alice