# Solution de planification des projets

1. Calendrier des livrables

|  |  |  |  |
| --- | --- | --- | --- |
| Phase | Tâches principales | Durée | Livrables |
| Planification initiale | Réunion de lancement, élaboration du plan de projet | 1 semaine | Plan de projet détaillé |
| Analyse des besoins | Identification des besoins, rédaction des spécifications | 1 semaine | Spécifications techniques |
| Développement du prototype | Configuration, codage, intégration | 3 semaines | Prototype fonctionnel |
| Tests et validation | Tests unitaires, intégration, validation | 1 semaine | Rapport de tests |
| Rapport final | Compilation des résultats, rédaction du rapport | 1 semaine | Rapport final et recommandations |

1. Liste des taches

 **Analyse de l’existant**

* Cartographier les fonctionnalités de l’ERP Delphi.
* Identifier les points d’intégration avec SQL Server.
* Documenter les workflows actuels.

 **Recueil des besoins**

* Interviews des utilisateurs finaux et des administrateurs systèmes.
* Documentation des exigences fonctionnelles et techniques.

 **Conception préliminaire**

* Création des maquettes d’interface utilisateur.
* Développement des diagrammes d’architecture.

 **Développement du prototype**

* Construction des modules en NodeJS/Angular.
* Intégration avec SQL Server.

 **Tests et évaluation**

* Réalisation de tests unitaires et de bout en bout.
* Exécution des tests de performance.

 **Documentation**

* Compilation des résultats des tests.
* Rédaction du rapport final.

1. Liste des outils pour le suivi du projet et justification

* **Trello/JIRA** : Gestion agile des tâches et visualisation claire de l’avancement du projet via des tableaux Kanban et des backlogs.
* **GitHub** : Versionnement du code et collaboration entre les développeurs.
* **Postman** : Test des API, assurant que les intégrations fonctionnent correctement.
* **Jenkins** : Intégration continue et déploiements automatisés, réduisant les risques de bugs et améliorant la qualité du code.

#### Utilisation de Trello pour la Gestion de Projet

##### Étape 1: Création de Tableaux Kanban

* **Listes** : Créer des listes pour chaque phase du projet (À faire, En cours, Terminé).
* **Cartes** : Créer des cartes pour chaque tâche avec des descriptions détaillées, des échéances et des assignations.

##### Étape 2: Suivi des Progrès

* **Étiquettes et Checklists** : Utiliser des étiquettes pour la priorisation et des checklists pour les sous-tâches.
* **Power-Ups** : Intégrer des Power-Ups comme Calendar pour la gestion des deadlines et des intégrations avec Slack pour les notifications.

1. Indicateurs et justifications

Les indicateurs de performance (KPI) permettent de mesurer l’efficacité et la qualité du prototype. Ils doivent être choisis en fonction de leur pertinence pour le projet.

##### Exemples d’Indicateurs :

* **Temps de réponse des API**:
  + **Pourquoi ?**: Mesure la rapidité des interactions entre le prototype et la base de données, crucial pour la performance.
* **Nombre d’erreurs détectées par fonctionnalité**:
  + **Pourquoi ?**: Indique la qualité du code et l’efficacité des tests.
* **Taux de succès des tests fonctionnels**:
  + **Pourquoi ?**: Évalue la conformité du prototype aux spécifications initiales.
* **Feedback des utilisateurs finaux**:
  + **Pourquoi ?**: Permet d’identifier les améliorations à apporter en fonction des retours d’expérience réels.
*  **Temps de réponse des API** : Mesure la rapidité des interactions entre le prototype et la base de données.
*  **Nombre d’erreurs détectées par fonctionnalité** : Indique la qualité du code et l’efficacité des tests.
*  **Taux de succès des tests fonctionnels** : Évalue la conformité du prototype aux spécifications initiales.
*  **Feedback des utilisateurs finaux** : Permet d’identifier les améliorations à apporter en fonction des retours d’expérience réels.

1. Referentiel de bonne pratique

*  **Méthodologie SCRUM** : Méthodologie agile permettant une gestion flexible et itérative du projet, avec des sprints courts et des feedbacks continus.
*  **Code Review** : Garantie de la qualité du code en permettant à plusieurs développeurs de vérifier et d’améliorer le code.
*  **Tests Unitaires et de Bout en Bout** : Assurent que chaque composant fonctionne correctement de manière isolée et en interaction avec d’autres composants.
*  **Documentation Continue** : Maintient une trace claire et complète des développements, facilitant la maintenance future et la compréhension du projet.

Tableau de bord general d’avancement

+--------------------------------------------------+

| Tableau de Bord Global du Projet |

+--------------------------------------------------+

| Phases | Progression (%) |

|-------------------|------------------------------|

| Analyse | ████████████████████ 80% |

| Conception | ████████████████ 70% |

| Développement | ███████████ 50% |

| Tests | ████ 20% |

| Documentation | █ 10% |

+--------------------------------------------------+

| Livrables | Statut |

|-------------------|------------------------------|

| Maquettes UI | En cours |

| Modules Backend | En cours |

| Tests Unitaires | Non commencé |

| Rapport Final | Non commencé |

+--------------------------------------------------+

| Ressources | Allocations |

|-------------------|------------------------------|

| Dev Frontend | 40% alloué |

| Dev Backend | 60% alloué |

| Admin Systèmes | 30% alloué |

+--------------------------------------------------+

| Budget (en €) | Utilisé / Total |

|-------------------|------------------------------|

| Analyse | 8,000 / 10,000 |

| Conception | 12,000 / 15,000 |

| Développement | 25,000 / 30,000 |

| Tests | 5,000 / 8,000 |

| Documentation | 2,000 / 3,000 |

+--------------------------------------------------+

 **Avancement Global du Projet**

* **Graphique à Barres** : Pourcentage d’achèvement des phases principales (analyse, conception, développement, tests, documentation).

 **État des Livrables**

* **Tableau** : Liste des livrables avec les statuts (Non commencé, En cours, Terminé).

 **Ressources et Temps**

* **Diagramme de Gantt** : Visualisation du calendrier des tâches et des ressources allouées.

 **Budget et Dépenses**

* **Graphique Linéaire** : Suivi des dépenses par rapport au budget prévu.