

Chart a Soumettre

Voici la charte de projet restructurée :

Charte de projet

Nom du projet	Gestion d'Hôtel (<code>hotelmanagementsystem</code>)
Chef de projet	Kanmegne Tabouguie Andre Yvan
Équipe	Kanmegne Tabouguie et Marc
github Repo	

But | Contexte

Que cherche-t-on à accomplir?	Développer une application de gestion d'hôtel permettant aux administrateurs de gérer les utilisateurs et les chambres, et aux clients de consulter les disponibilités des chambres.
Quelle motivation?	Valider les compétences acquises en développement, infrastructure et réseaux au cours de cette année.
Quel est le contexte du projet?	Ce projet s'inscrit dans le cadre de notre formation en développement et infrastructure, visant à mettre en pratique les compétences théoriques et pratiques acquises.

Périmètre | Critères de succès

Dans le périmètre	Comment déterminer si le projet est réussi?
- Développement d'une API pour la gestion des utilisateurs et des chambres.	- Fonctionnalité complète de l'API (CRUD pour utilisateurs et chambres).
- Développement d'une interface utilisateur pour les administrateurs et les clients.	- Interface utilisateur réactive et accessible.
- Mise en place de l'infrastructure nécessaire (serveurs web, base de données, mise en place DNS, DHCP, Gestion access)	- Déploiement réussi de l'application sur un serveur de production.

- Mise en place de la sécurité (authentification, pare-feu).

- Respect des délais.

Hors du périmètre

- **Fonctionnalités avancées non essentielles à la gestion de base des utilisateurs et des chambres(Intégration de systèmes externes(API),Fonctionnalités de gestion financière,Fonctionnalités multilingues et multi-devises)**
- Intégration avec des systèmes externes (par exemple, systèmes de paiement, CRM).

Rôles | Responsabilités

Rôles	Responsabilités
Chef de projet	- Coordination globale du projet. - Communication avec les parties prenantes. - Suivi des délais et du budget.
Développeur Backend	- Développement de l'API. - Intégration avec la base de données. - Mise en place des tests unitaires et d'intégration.
Développeur Frontend	- Développement de l'interface utilisateur. - Assurer la réactivité et l'accessibilité de l'application.
Ingénieur DevOps	- Mise en place de l'infrastructure. - Configuration des serveurs et déploiement. - Mise en place des pipelines CI/CD.
Testeur QA	- Tests fonctionnels et non fonctionnels. - Validation de la qualité du code et des fonctionnalités.

Cette charte de projet permet de définir clairement les objectifs, le périmètre, les critères de succès, et les rôles et responsabilités au sein de l'équipe pour le projet de gestion d'hôtel.

Détails supplémentaires

Planification

- **Durée : 6 jours(2 mai au 7 mai)**

- **Activités :**

- Définir les rôles et responsabilités des membres de l'équipe.
- Établir les objectifs du projet.
- Élaborer un plan de projet détaillé.
- Sélectionner les technologies à utiliser.

MVP (Minimum Viable Product)

- **Fonctionnalités implémentées :**

- **User Authentication** : Inscription, connexion et déconnexion des utilisateurs.
- **Authorization** : Protection des routes basées sur les rôles (admin ou guest).
- **Room Management** : Création, mise à jour, suppression et consultation des chambres par les administrateurs.
- **User Management** : Création, mise à jour, suppression et consultation des utilisateurs par les administrateurs.
- **Token Invalidation** : Invalidation des tokens à la déconnexion.
- **Caching** : Mise en cache des données des chambres et des utilisateurs.
- View Rooms by Users

Livrables

- UML détaillant la base de données.
- Documentation technique complète de l'architecture du projet, du code, et des configurations.
- Documentation détaillant l'objectif du projet.
- README détaillant le lancement de l'application.
- URL d'un dépôt Github.

Cette charte de projet donne une vue d'ensemble claire et détaillée de ce qui est attendu pour ce projet de gestion d'hôtel.

Phase 1: Planification avec dateline

Activités :

1. Définir les rôles et responsabilités des membres de l'équipe

- **Membre 1 (Développeur Backend & Frontend, Documentation) :**

- Responsable du développement des API backend.
- Responsable du développement des ux/ui frontend
- conception bdd
- Documentation

- **Membre 2 (Infrastructure & Documentation) :**

Configuration des serveurs et installation des serveurs (8 pts)

- Web

- Basededonnées

- Mise en place d'un DNS et d'un DHCP (4 pts)
- Mise en place d'un pare-feu (4 pts)
- Gestion des accès
- Documentation

2. Établir les objectifs du projet

- Développer une application de gestion d'hôtel avec les fonctionnalités suivantes :
 - **User Authentication** : Inscription, connexion et déconnexion des utilisateurs.
 - **Authorization** : Protection des routes en fonction des rôles des utilisateurs (admin ou guest).
 - **Room Management** : Gestion des chambres par les administrateurs (création, mise à jour, suppression, consultation).

- **User Management** : Gestion des utilisateurs par les administrateurs (création, mise à jour, suppression, consultation).
- **Token Invalidation** : Invalidation des tokens lors de la déconnexion.
- **Caching** : Mise en cache des données des chambres et des utilisateurs pour une durée spécifiée.

3. Élaborer un plan de projet détaillé

- **Semaine 1 : Planification(3 jours)(du 2 au 4 mai)**
 - Définir les rôles et responsabilités.
 - Établir les objectifs du projet.
 - Sélectionner les technologies à utiliser.
- **Semaine 1 : Conception(3 jours)(du 5 mai au 7 mai)**
 - Conception de l'architecture de l'application.
 - Création des diagrammes UML pour la base de données.
 - Rédaction de la documentation technique initiale.
- **Semaine 2-3 : Développement Backend(10 jours)(du 8 au 17 mai)**
 - Développement des API CRUD pour la gestion des utilisateurs et des chambres.
 - Implémentation de l'authentification des utilisateurs.
 - Tests unitaires des API.
- **Semaine 4 : Développement Frontend(20-26 mai)**
 - Développement de l'interface utilisateur réactive et accessible.
 - Intégration du frontend avec les API backend.
 - Tests d'interface utilisateur.
- **Semaine 5 : Infrastructure & Réseaux(27 mai - 02 juin)**
 - Configuration des serveurs (web, base de données, mail).
 - Mise en place du DNS et DHCP.

- Configuration du pare-feu et des règles de sécurité.
- **Semaine 12 : Tests & Validation**
 - Tests d'intégration.
 - Validation de l'infrastructure.
 - Tests de performance et de sécurité.
- **Semaine 13 : Documentation & Livraison**
 - Finalisation de la documentation technique et du README.
 - Préparation de la documentation utilisateur.
 - Dépôt du projet sur GitHub.
- **Semaine 14 : Présentation & Évaluation**
 - Préparation de la présentation du projet.
 - Présentation du projet aux évaluateurs.
 - Réception des feedbacks et ajustements si nécessaire.

4. Sélectionner les technologies à utiliser

- **Backend :**
 - **Node.js** avec **Express** pour la création des API RESTful.
 - **Mysql** pour la gestion de la base de données.
 - **JWT** pour l'authentification et l'autorisation.
 - **Memcached** pour la mise en cache des données.
- **Frontend :**
 - **Html, javascript** pour le développement de l'interface utilisateur.
 - **Bootstrap** pour le design de l'interface.
- **Infrastructure & Réseaux :**
 - **Nginx** pour le serveur web.
- **Outils de Documentation :**

- **Swagger** pour la documentation des API.
- **Markdown** pour la rédaction des README et de la documentation technique.
- **Draw.io** pour la création des diagrammes UML.
- **notion**
- **CI/CD :**
 - **GitHub Actions** pour l'intégration continue et le déploiement continu.

Ce plan de projet détaillé et les technologies sélectionnées permettront de structurer efficacement le développement et l'infrastructure du projet de gestion d'hôtel, tout en répondant aux exigences des compétences visées.

Phase Une Documentation(planification).