

Ministère de l'Enseignement Supérieur Et de la Recherche Scientifique
Université de Tunis

Ecole Supérieure Privée de technologies et d'ingénierie TEKUP Tunis



Coworkingspace

Reservation intelligente pour espaces de coworking

Réalisé Par

Akrout Farah

Laabidi Mariem

Année universitaire : 2024-2025

La transformation des modes de travail a accéléré l'adoption des espaces de coworking comme solution flexible pour les professionnels indépendants, startups et entreprises. Cependant, la gestion manuelle des réservations (via cahiers physiques ou feuilles de calcul) reste une source d'inefficacité, entraînant des conflits d'horaires et une charge administrative excessive.

Dans ce contexte, notre projet intitulé "**Système Intelligent de Réservation pour Espaces de Coworking**" vise à digitaliser et automatiser la gestion des réservations de bureaux et salles de réunion. La plateforme proposée permettra aux utilisateurs de :

- Consulter les disponibilités en temps réel
- Réserver/modifier des créneaux de manière intuitive
- Bénéficier de notifications automatisées

Pour les gestionnaires d'espaces, cette solution offrira :

- Une vue centralisée des réservations
- Des outils d'analyse d'occupation
- Une réduction des erreurs humaines

Structure du rapport :

- **Contexte du projet** : Enjeux des espaces collaboratifs et analyse des besoins (utilisateurs, administrateurs, propriétaires)
- **Planification** :
 - Exigences fonctionnelles (diagramme de cas d'utilisation)
 - Architecture technique (frontend/backend)
- **Sprint 1 - Authentification** :

- Système d’inscription/connexion sécurisé
 - Gestion des rôles (user, admin)
- **Sprint 2 - Réservations :**
 - Module de réservation avec calendrier interactif
 - Prévention des conflits horaires
- **Sprint 3 - Administration :**
 - Tableau de bord de gestion
 - Reporting d’occupation
- **Clôture :**
 - Diagramme de classes complet
 - Environnement technique (React/GraphQL/Node.js)

Ce projet permettra d’acquérir des compétences en :

- Développement Fullstack (TypeScript, GraphQL)
- Gestion de projet agile
- Conception d’UI/UX pour applications professionnelles

TABLE DES MATIÈRES

Introduction générale	ii
Table des figures	vi
I Contexte du projet	1
I Présentation du projet	1
I.1 Problématiques actuelles	1
I.2 Objectifs	1
II Méthodologie	2
II.1 Sprints clés	2
II.2 Outils	2
III Technologies	2
III.1 Frontend	2
III.2 Backend	3
II Spécification des Besoins	4
I Besoins fonctionnels	4
I.1 Modules principaux	4
II Besoins non-fonctionnels	5
III Modélisation UML	5
III.1 Diagramme de Cas d'Utilisation	5

III.2	Diagramme de Classes	5
III.3	Scénarios d'utilisation	6
III.4	Stack technique	6
IV	Backlog produit	7

TABLE DES FIGURES

I.1	Architecture technique	2
II.1	Diagramme de cas d'utilisation des principaux acteurs	5
II.2	Diagramme de classes principal	6

CHAPITRE I

CONTEXTE DU PROJET

Introduction

Ce chapitre présente le contexte du système de réservation pour espaces de coworking. Nous analyserons les problématiques des solutions actuelles et justifierons nos choix technologiques.

I Présentation du projet

Notre solution vise à digitaliser la gestion des réservations dans les espaces de coworking, remplaçant les méthodes manuelles (cahiers, Excel) par une plateforme web moderne.

I.1 Problématiques actuelles

- Gestion manuelle des réservations (erreurs fréquentes)
- Conflits d'horaires non détectés
- Absence de visualisation centralisée
- Processus de réservation peu intuitif

I.2 Objectifs

- Interface utilisateur simple et intuitive
- Calendrier interactif des disponibilités
- Prévention des réservations conflictuelles
- Notifications automatiques de bord d'administration

II Méthodologie

Nous utilisons une approche Agile/Scrum avec :

II.1 Sprints clés

- Sprint 1 : Authentification et gestion des comptes
- Sprint 2 : Module de réservation
- Sprint 3 : Tableau de bord admin

II.2 Outils

- Jira pour le suivi des tâches
- Figma pour le prototypage
- Git pour le versioning

III Technologies

Notre stack technique a été choisie pour sa performance et sa modernité :

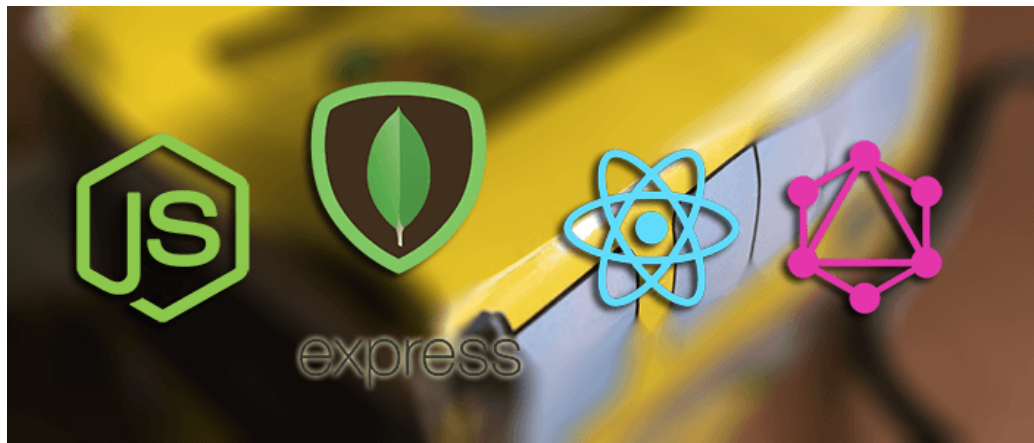


FIGURE I.1 – Architecture technique

III.1 Frontend

- React avec TypeScript

III.2 Backend

- Node.js avec Express
- GraphQL pour les APIs

Conclusion

Cette solution répondra aux besoins des utilisateurs tout en simplifiant la gestion des espaces. Les choix techniques permettent un développement efficace et évolutif.

CHAPITRE II

SPÉCIFICATION DES BESOINS

Introduction

Ce chapitre présente les exigences fonctionnelles et techniques de la plateforme de réservation d'espaces de coworking, ainsi que l'architecture globale du système.

I Besoins fonctionnels

I.1 Modules principaux

- **Gestion des espaces :**
 - Création/modification des bureaux et salles
 - Définition des capacités d'accueil
 - Visualisation des disponibilités
- **Réservations :**
 - Prise de rendez-vous en ligne
 - Annulation/modification
 - Prévention des conflits horaires
- **Utilisateurs :**
 - Inscription/connexion
 - Profils personnalisés
 - Historique des réservations

- **Administration :**
 - Gestion des membres de bord statistique
 - Configuration du système

II Besoins non-fonctionnels

- **Performance :** Temps de réponse < 1s
- **Disponibilité :** 99% uptime
- **Sécurité :** Authentification robuste
- **Responsive :** Compatible mobile

III Modélisation UML

III.1 Diagramme de Cas d'Utilisation

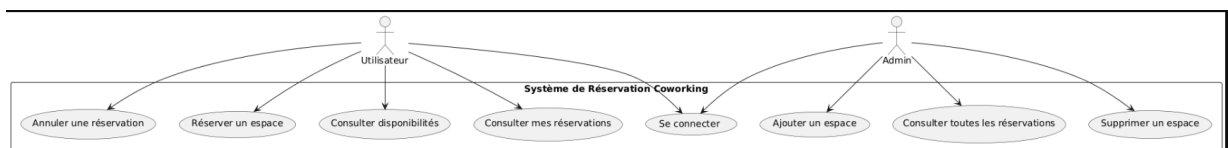


FIGURE II.1 – Diagramme de cas d'utilisation des principaux acteurs

Le diagramme de cas d'utilisation (Fig. II.1) présente les interactions entre les acteurs principaux et le système :

- **Membre :** Peut réserver/modifier des créneaux, consulter les disponibilités
- **Staff :** Gère les espaces et valide les réservations
- **Admin :** Configure le système et gère les utilisateurs

III.2 Diagramme de Classes

Le diagramme de classes (Fig. II.2) modélise les entités clés :

- **User :** Classe parente pour tous les types d'utilisateurs
- **Space :** Représente les bureaux/salles avec leurs caractéristiques
- **Reservation :** Lie utilisateurs, espaces et créneaux horaires
- **Notification :** Gère les rappels et confirmations

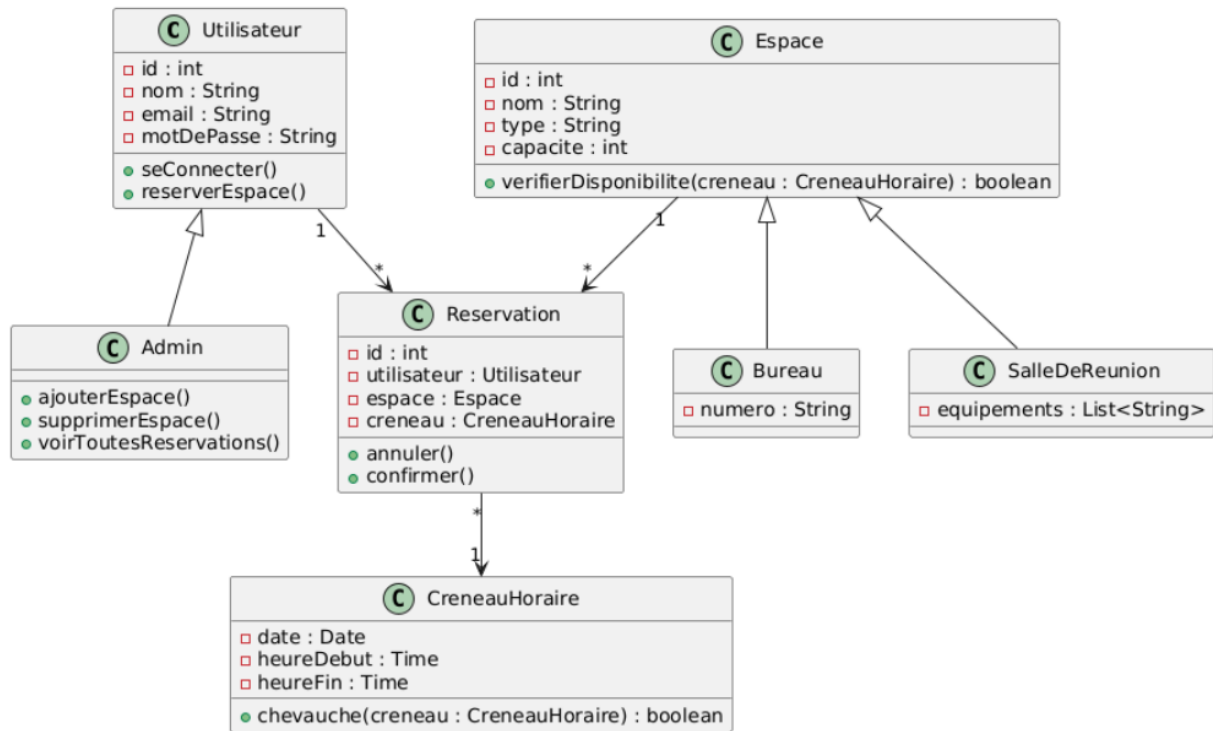


FIGURE II.2 – Diagramme de classes principal

III.3 Scénarios d'utilisation

Cas	Flux principal
Réservation	1. Consultation disponibilités → 2. Sélection espace → 3. Confirmation
Gestion espace	1. Ajout nouvel espace → 2. Définition capacités → 3. Activation
Annulation	1. Consultation réservations → 2. Sélection → 3. Confirmation annulation

TABLE II.1 – Scénarios principaux

III.4 Stack technique

- **Frontend** : React + TypeScript
- **Backend** : Node.js + GraphQL
- **Base de données** : PostgreSQL
- **Outils** : Git, Jira, Figma

IV Backlog produit

ID	Description	Priorité
US1	Inscription utilisateur	Haute
US2	Réservation d'espace	Haute
US3	Calendrier des disponibilités	Haute
US4	Notification par email	Moyenne
US5	Tableau de bord admin	Moyenne

Conclusion

Cette spécification définit les besoins essentiels pour une solution de gestion de coworking efficace, moderne et scalable.

SPRINT 1 : AUTHENTIFICATION DES UTILISATEURS ET ADMINISTRATEURS

Objectif du sprint

Ce premier sprint vise à mettre en place le système d'authentification permettant aux utilisateurs et aux administrateurs de se connecter de manière sécurisée à la plateforme de réservation de coworking. Ce module est essentiel pour garantir l'accès personnalisé aux différentes fonctionnalités selon le rôle de l'utilisateur.

Backlog du sprint

- Création des modèles **User** et **Admin** avec héritage si applicable.
- Mise en place de la base de données pour stocker les identifiants et rôles.
- Implémentation du formulaire de connexion.
- Authentification avec vérification des identifiants.
- Gestion des sessions utilisateur.
- Redirection selon le rôle (espace utilisateur ou tableau de bord admin).
- Validation côté client et serveur.
- Gestion des erreurs (identifiants incorrects, champs vides, etc.).
- Tests unitaires du module.

Livrables

- Interface de connexion fonctionnelle.

- Système d'authentification sécurisé (hashage des mots de passe).
- Redirection automatique selon le type d'utilisateur.
- Documentation du module d'authentification.

Technologies utilisées

- **Backend** : PHP / Node.js / Python (selon ton stack)
- **Base de données** : MySQL / PostgreSQL
- **Frontend** : HTML/CSS, JavaScript
- **Sécurité** : Hashage avec bcrypt ou équivalent

Résultat

Le système d'authentification a été mis en place avec succès. Les utilisateurs peuvent désormais se connecter avec leurs identifiants, et accéder à leur espace personnel selon leur rôle (utilisateur ou administrateur). La base est posée pour permettre une gestion différenciée des droits d'accès pour les futurs modules de réservation et d'administration.

Objectif du sprint

Ce deuxième sprint a pour objectif la mise en place du module de réservation, permettant aux utilisateurs de réserver des bureaux ou des salles de réunion à l'aide d'un calendrier interactif. Il inclut également la gestion des conflits d'horaires afin de garantir qu'un créneau donné ne puisse être réservé qu'une seule fois.

Backlog du sprint

- Création de l'interface de réservation avec sélection de date et heure.
- Intégration d'un calendrier interactif (ex. : FullCalendar, flatpickr).
- Développement de la logique de réservation côté serveur.
- Association des réservations aux utilisateurs et aux espaces.
- Mise en place de la vérification des conflits d'horaires :
 - Empêcher la réservation si un créneau est déjà pris.
 - Retour visuel d'erreur en cas de conflit.
- Affichage en temps réel des disponibilités.
- Sauvegarde des réservations dans la base de données.
- Tests unitaires et fonctionnels du système.

Livrables

- Module de réservation complet avec sélection horaire.
- Affichage d'un calendrier interactif avec les créneaux disponibles.

- Système de validation de disponibilité pour éviter les conflits.
- Stockage des réservations avec liaison utilisateur et espace.
- Notification de confirmation ou d'erreur.

Technologies utilisées

- **Frontend** : JavaScript, bibliothèque de calendrier (FullCalendar, flatpickr, etc.)
- **Backend** : PHP / Node.js / Python selon le stack choisi
- **Base de données** : MySQL / PostgreSQL
- **Outils de validation** : middleware ou logique métier personnalisée

Résultat

Le module de réservation a été développé avec succès. Les utilisateurs peuvent désormais réserver un espace en sélectionnant un créneau dans un calendrier interactif. Le système détecte automatiquement les conflits d'horaires et empêche toute réservation doublon. Ce sprint marque une étape importante, posant les fondations d'une gestion dynamique et fiable des espaces de coworking.

Objectif du sprint

Ce troisième sprint vise à doter les administrateurs d'un tableau de bord leur permettant de gérer les espaces de coworking, de consulter les réservations en cours et passées, et d'obtenir des statistiques d'occupation. Ce module facilite la supervision, la prise de décision et l'optimisation de l'utilisation des ressources.

Backlog du sprint

- Création d'un tableau de bord réservé aux administrateurs.
- Affichage des espaces (bureaux, salles de réunion) avec leurs statuts (libre, réservé).
- Visualisation des réservations par jour, semaine ou mois.
- Mise en place d'un système de reporting :
 - Taux d'occupation par espace.
 - Nombre total de réservations par période.
 - Heures les plus demandées.
- Possibilité de filtrer les données par date, espace ou utilisateur.
- Interface responsive et ergonomique.
- Tests de performance et de charge sur les requêtes statistiques.

Livrables

- Interface d'administration avec tableau de bord.
- Outils de visualisation des réservations.

- Rapports d’occupation en temps réel.
- Filtres interactifs pour l’analyse des données.
- Documentation d’utilisation du tableau de bord.

Technologies utilisées

- **Frontend** : HTML/CSS, JavaScript, framework (Bootstrap, Vue.js, etc.)
- **Backend** : PHP / Node.js / Python
- **Base de données** : MySQL / PostgreSQL
- **Statistiques** : Requêtes SQL avancées, graphiques avec Chart.js ou équivalent

Résultat

Le tableau de bord d’administration est désormais fonctionnel et permet une gestion complète des espaces de coworking. Les administrateurs peuvent consulter les réservations en cours et obtenir des indicateurs clés d’occupation. Ce module améliore considérablement la visibilité sur l’activité de la plateforme et pose les bases d’une gestion stratégique des ressources.

CONCLUSION

Le projet de plateforme de gestion des professionnels que nous avons réalisé constitue une solution numérique moderne et efficace, pensée pour faciliter la mise en relation entre les utilisateurs et divers experts tels que les médecins, avocats et coachs. Grâce à une architecture bien conçue et à une organisation rigoureuse par sprints, nous avons pu développer progressivement des modules clés, allant de l'authentification sécurisée à la prise de rendez-vous, en passant par la gestion des profils professionnels et des communications entre utilisateurs.

L'utilisation de la méthode agile SCRUM a été un atout majeur dans l'avancement du projet. Elle nous a permis de planifier et de suivre chaque étape du développement de manière itérative, tout en garantissant l'adaptabilité face aux retours et aux besoins fonctionnels. Chaque sprint a été l'occasion de se concentrer sur une fonctionnalité spécifique : affichage du catalogue de professionnels, filtres de recherche dynamiques, système de messagerie intégrée, ou encore génération de notifications intelligentes.

Ce projet nous a permis de mettre en œuvre une large palette de compétences techniques : conception de bases de données relationnelles, développement backend et frontend, sécurisation des accès, intégration de services tiers, et déploiement sur un environnement cloud. Sur le plan humain, il nous a appris à collaborer efficacement, à prioriser les tâches et à nous organiser selon les principes de gestion de projet agiles.

En conclusion, cette plateforme offre une base robuste et évolutive sur laquelle il sera possible d'ajouter de futures fonctionnalités, telles que la téléconsultation, les paiements en ligne, ou encore les évaluations des prestations. Ce projet marque une étape essentielle dans notre parcours de formation, alliant exigence technique, collaboration en équipe et réponse à un besoin concret du monde professionnel.