

# Cahier des charges : Système de réservation

## 1. Description générale du produit

Le système de réservation permet aux utilisateurs de réserver des salles et des ressources (projecteurs, ordinateurs, tableaux interactifs, etc.) pour des créneaux horaires définis. L'objectif est de faciliter la planification, éviter les conflits de réservation et centraliser l'information relative aux disponibilités. L'application peut être accessible via un navigateur web ou une application mobile.

## 2. Rôles utilisateurs

Pour simplification, le système comportera un **seul rôle utilisateur principal** :

- **Utilisateur standard** : peut consulter les disponibilités, créer, modifier ou annuler ses réservations.

Optionnel (pour extensions futures) :

- **Administrateur** : gestion des salles, ressources, utilisateurs et rapports statistiques.

## 3. Fonctionnalités essentielles

### 1. Gestion des réservations

- Créer une réservation en sélectionnant salle, ressources et créneau horaire.
- Modifier ou annuler une réservation existante.
- Visualiser l'historique de ses réservations.

### 2. Consultation des disponibilités

- Affichage des disponibilités des salles et ressources par jour, semaine ou mois.
- Indication des créneaux déjà réservés.

### 3. Gestion des ressources et salles

- Liste des salles disponibles avec leurs caractéristiques (capacité, équipements).
- Liste des ressources associées aux salles.

### 4. Notifications et alertes

- Confirmation de réservation par email ou notification.
- Alertes en cas de conflit ou de modification de réservation.

## 4. Contraintes techniques

- Application web responsive (desktop et mobile).
- Base de données relationnelle pour stocker utilisateurs, salles, ressources et réservations.

- Gestion des conflits de réservation (une ressource ou salle ne peut être réservée deux fois pour le même créneau).
- Authentification simple via email/mot de passe.
- Sauvegarde régulière des données.

## 5. Écrans obligatoires

1. **Écran de connexion / inscription**
2. **Tableau de bord / accueil** : vue globale des réservations à venir.
3. **Écran de réservation** : création, modification et suppression d'une réservation.
4. **Calendrier / planning des salles** : consultation des disponibilités.
5. **Fiche salle / ressource** : détails sur la salle ou la ressource sélectionnée.
6. **Historique des réservations** : liste des réservations passées et à venir.
7. **Paramètres utilisateur** : gestion du profil et préférences (optionnel).

## 6. Règles métier de base

- Une réservation ne peut pas se chevaucher avec une autre réservation pour la même salle ou ressource.
- La durée minimale et maximale des réservations peut être définie (ex. 30 min minimum, 4 h maximum).
- Les utilisateurs ne peuvent modifier ou annuler que leurs propres réservations.
- Certaines ressources peuvent être obligatoirement associées à certaines salles (ex. un projecteur dans une salle de réunion).
- Les créneaux horaires sont prédéfinis par l'administration ou configurables par l'utilisateur selon disponibilité.

### Liste des entités et attributs :

#### 1. Utilisateur

- **id** (identifiant unique)
- **nom**
- **email**
- **mot\_de\_passe**
- **role** (par simplification, toujours "utilisateur")

#### 2. Salle

- **id** (identifiant unique)

- **nom**
- **capacité**
- **description** (optionnel : équipements intégrés)
- **ressources\_disponibles** (liste ou relation vers Ressource)

### 3. Ressource

- **id** (identifiant unique)
- **nom**
- **type** (projecteur, ordinateur, tableau, etc.)
- **quantité disponible**
- **associée\_aux\_salles** (relation vers Salle si nécessaire)

### 4. Réservation

- **id** (identifiant unique)
- **utilisateur\_id** (relation vers Utilisateur)
- **salle\_id** (relation vers Salle)
- **ressources\_ids** (liste ou relation vers Ressource)
- **date et créneau\_horaire** (début et fin)
- **statut** (confirmée, annulée, etc.)

## Relations entre entités :

- **Utilisateur → Réservation** : un utilisateur peut faire plusieurs réservations (1-n).
- **Salle → Réservation** : une salle peut être réservée plusieurs fois dans le temps (1-n).
- **Ressource → Réservation** : une réservation peut utiliser plusieurs ressources, et une ressource peut être utilisée dans plusieurs réservations (n-n).
- **Salle → Ressource** : une salle peut avoir plusieurs ressources disponibles (1-n), et une ressource peut être associée à plusieurs salles (n-n selon complexité).

## Diagramme Mermaid :

```
classDiagram
class Salle {
+id: int
+nom: string
+capacite: int
+localisation: string
}
```

```

class Ressource {
    +id: int
    +type: string
    +description: string
    +disponibilite: bool
}

class CreneauHoraire {
    +id: int
    +date: Date
    +heureDebut: Time
    +heureFin: Time
}

class Reservation {
    +id: int
    +utilisateur: string
    +statut: string
}

Salle "1" --> "0..*" Reservation
Ressource "1" --> "0..*" Reservation
CreneauHoraire "1" --> "0..*" Reservation

```

### Diagramme de séquence (cas d'usage : créer une réservation) :

sequenceDiagram

participant U as Utilisateur

participant F as Frontend

participant B as Backend

participant DB as Base de données

U->>F: Remplit formulaire réservation

F->>B: Envoie requête POST /reservation

B->>DB: Vérifie disponibilité (Salle + Ressource + Créneau)

DB-->>B: Retourne résultat

alt Disponible

B->>DB: Insère réservation

DB-->>B: Confirmation

B-->>F: Réservation validée

F-->>U: Message succès

**else Conflit**

**B-->F: Erreur "Créneau indisponible"**

**F-->U: Message d'erreur**

**end**

**Wireframes (textuels) :**

- **Connexion** : champ email + mot de passe + bouton "Se connecter".
- **Liste des réservations** : tableau avec colonnes (Salle, Ressource, Date, Heure, Statut).
- **Formulaire CRUD** : dropdown Salle, dropdown Ressource, date picker, heure début/fin, bouton "Valider".