

Cahier des Charges - Application de Bureau "ReservationSalles"

Table des matières

1. Présentation du projet.....	1
2. Objectifs	2
3. Fonctionnalités principales	2
3.1. Interface utilisateur	2
3.2. Gestion des réservations.....	3
3.3. Gestion des salles (Administration).....	3
3.4. Gestion des utilisateurs (Administration).....	4
3.5. Planning unifié.....	4
3.6. Autres fonctionnalités.....	4
4. Contenu (Données gérées)	4
5. Contraintes techniques.....	5
6. Maintenance et évolutivité.....	6
7. Validation du projet.....	6

1. Présentation du projet

- **Nom du projet:** ReservationSalles
- **Objectif principal:** Développer une application de bureau (client lourd) robuste et fonctionnelle en utilisant WPF (.NET 8) pour la gestion de réservations de salles. L'application doit permettre l'authentification des utilisateurs, la consultation et l'ajout de réservations, ainsi qu'une interface d'administration dédiée à la gestion complète des utilisateurs, des salles et des réservations existantes. Elle doit également offrir une vue de planning consolidée.

2. Objectifs

- **Application de bureau fonctionnelle:** Fournir une interface utilisateur claire et intuitive développée avec WPF pour Windows.
- **Gestion centralisée:** Permettre la gestion efficace des salles, des utilisateurs (avec différents rôles) et des réservations via des interfaces dédiées.
- **Contrôle d'accès:** Implémenter un système d'authentification et d'autorisation basé sur les rôles (Utilisateur standard et Administrateur).
- **Visualisation du planning:** Offrir une vue calendaire (hebdomadaire et mensuelle) des réservations avec des options de navigation, de filtrage, d'export et d'impression.
- **Persistance des données:** Utiliser une base de données locale (SQLite) pour stocker les informations de manière fiable, gérée via Entity Framework Core.

3. Fonctionnalités principales

3.1. Interface utilisateur

- **Fenêtre de connexion (LoginWindow):** Permet aux utilisateurs de s'authentifier avec un nom d'utilisateur et un mot de passe. Inclut une option pour afficher/masquer le mot de passe.
- **Fenêtre principale (MainWindow):** Interface affichée après connexion pour les utilisateurs standards.
 - Affichage de la liste des salles.
 - Affichage de la liste des réservations existantes.
 - Formulaire permettant d'ajouter une nouvelle réservation (sélection salle, date, heures début/fin, informations participant/sujet).
 - Bouton d'accès à l'administration (visible/actif uniquement pour les administrateurs).
 - Bouton d'accès au planning unifié.
 - Bouton de déconnexion.
- **Fenêtre d'administration (AdminWindow):** Interface dédiée aux administrateurs.
 - Gestion des utilisateurs (liste, ajout, suppression, réinitialisation de mot de passe).

- Gestion des salles (liste, ajout, suppression).
- Gestion des réservations (liste, modification, suppression).
- **Fenêtre d'édition de réservation (EditReservationWindow):** Fenêtre modale permettant de modifier les détails d'une réservation existante (accessible depuis AdminWindow ou UnifiedPlanningWindow).
- **Fenêtre du Planning Unifié (UnifiedPlanningWindow):**
 - Affichage du planning en mode hebdomadaire ou mensuel.
 - Navigation entre les semaines/mois.
 - Filtrage des réservations affichées (par salle, utilisateur, objet).
 - Représentation visuelle des réservations sur la grille, avec couleurs distinctes par salle.
 - Info-bulle détaillée au survol d'une réservation.
 - Modification d'une réservation par double-clic.
 - Export des réservations visibles au format CSV.
 - Fonction d'impression de la vue actuelle du planning.

3.2. Gestion des réservations

- **Ajout:** Possible depuis MainWindow. Vérification automatique de la disponibilité du créneau horaire avant l'ajout. Les horaires sont sélectionnables par tranches de 30 minutes.
- **Modification:** Possible depuis AdminWindow ou par double-clic dans UnifiedPlanningWindow. La modification via EditReservationWindow permet de changer les horaires, le participant et l'objet (pas la salle). La disponibilité du nouveau créneau est vérifiée.
- **Suppression:** Possible uniquement par les administrateurs depuis AdminWindow ou UnifiedPlanningWindow.

3.3. Gestion des salles (Administration)

- **Ajout:** Via AdminWindow. Définition du nom et de la capacité (nombre entier positif). Vérification de l'unicité du nom (insensible à la casse).
- **Suppression:** Via AdminWindow. Entraîne la suppression automatique des réservations associées (via la configuration EF Core Cascade).

3.4. Gestion des utilisateurs (Administration)

- **Ajout:** Via AdminWindow. Définition du nom d'utilisateur, du mot de passe initial et du rôle (Admin ou User). Vérification de l'unicité du nom d'utilisateur (insensible à la casse). Validation de la complexité du mot de passe avant ajout.
- **Suppression:** Via AdminWindow. L'utilisateur "admin" principal ne peut pas être supprimé.
- **Réinitialisation du mot de passe:** Via AdminWindow. Attribution d'un mot de passe temporaire (défini dans le code, ex: "Password123!").

3.5. Planning unifié

- **Vues:** Hebdomadaire (grille horaire de 8h à 19h) et Mensuelle (calendrier).
- **Navigation:** Boutons Précédent/Suivant pour changer de semaine ou de mois.
- **Filtrage:** Sélection du critère (Tous, Salle, Utilisateur, Objet) et saisie de texte pour affiner l'affichage.
- **Export:** Génération d'un fichier CSV contenant les réservations actuellement visibles dans le planning.
- **Impression:** Impression de la vue actuelle de la grille/calendrier.

3.6. Autres fonctionnalités

- **Authentification & Autorisation:** Système basé sur les rôles (UserRole.Admin, UserRole.User). L'accès aux fonctionnalités d'administration est réservé au rôle Admin.
- **Persistence:** Utilisation d'EF Core et SQLite pour la sauvegarde des données. La base de données est créée/migrée au premier lancement.
- **Logging:** Enregistrement des événements importants (connexion, erreurs, opérations CRUD) dans un fichier application.log.
- **Sécurité Mots de Passe:** Hashage SHA256 des mots de passe.

4. Contenu (Données gérées)

- **Utilisateurs (User):**
 - Id (Clé primaire, entier auto-incrémenté)
 - Username (Texte, unique)
 - Password (Texte, hash SHA256)

- Role (Entier, représentant l'enum UserRole: 0=Admin, 1=User)
- **Salles (Room):**
 - Id (Clé primaire, entier auto-incrémenté)
 - Name (Texte, unique)
 - Capacity (Entier)
- **Réservations (Reservation):**
 - Id (Clé primaire, entier auto-incrémenté)
 - RoomId (Clé étrangère vers Rooms.Id)
 - StartTime (Date et Heure)
 - EndTime (Date et Heure)
 - AttendeeFirstName (Texte)
 - AttendeeLastName (Texte)
 - MeetingSubject (Texte)

5. Contraintes techniques

- **Plateforme:** Application de bureau Windows.
- **Framework:** .NET 8
- **Interface Utilisateur:** WPF (Windows Presentation Foundation)
- **Langage:** C#
- **Accès aux données:** Entity Framework Core 9.x
- **Base de données:** SQLite (fichier local dans %LOCALAPPDATA%\ReservationSallesApp)
- **Architecture:** MVVM (Model-View-ViewModel)
- **Dépendances clés:** Microsoft.EntityFrameworkCore, Microsoft.EntityFrameworkCore.Sqlite, Microsoft.EntityFrameworkCore.Tools.
- **Système d'exploitation cible:** Windows 7 ou supérieur (spécifié par net8.0-windows).

6. Maintenance et évolutivité

- **Maintenance corrective:** Le logging applicatif (application.log) facilite le suivi des erreurs. Les dépendances (framework .NET, EF Core) devront être maintenues à jour pour les correctifs de sécurité.
- **Maintenance évolutive:**
 - L'architecture MVVM facilite l'ajout de nouvelles fonctionnalités ou la modification des existantes.
 - EF Core Migrations permet de gérer les évolutions du schéma de la base de données.
 - **Améliorations possibles:** Implémentation du salage des mots de passe, optimisation des requêtes LINQ potentiellement lentes, stratégie de validation plus centralisée, amélioration de la gestion des erreurs, ajout de fonctionnalités (ex: notifications, réservations récurrentes, gestion plus fine des droits).

7. Validation du projet

- **Contexte:** Application développée comme un projet spécifique (client lourd). La validation fonctionnelle repose sur l'exécution de l'application et la vérification manuelle des fonctionnalités décrites.