

**Cours de Génie Logiciel**

Enseigné par le professeur Jean Andris ADAM

**Plateforme de Gestion de Réservation de Salles**

Membres du groupe :

1. Natanael Godson Ralph SÉNATUS (CP)
2. Dénilson ETIENNE

Juin 2025

**A. Identification**

# A.1. Identification du document

|  |  |
| --- | --- |
| **Nom du document** | Conception Fonctionnelle |
| **Project:** | Plateforme de Gestion et de Réservation de Salle |
| **Version du projet:** | 1.0 |
| **Version du document:** | 1.3 |
| **Sécurité du document:** | Privé |
| **Date de création:** | 13 mai 2025 |
| **Par:** | N. Godson Ralph SÉNATUS & Denilson ÉTIENNE |

# A.2. Historique des changements

|  |  |  |
| --- | --- | --- |
| **Qui?** | **Quand?** | **Quoi?** |
| N. Godson Ralph SÉNATUS | 13 mai 2025 | Création |
| Denilson ÉTIENNE | 19 mai 2025 | Mise à jour du document |
| Denilson ÉTIENNE | 25 mai 2025 | Mise à jour du document |
| N. Godson Ralph SÉNATUS | 28 mai 2025 | Mise à jour du document |
| Denilson ÉTIENNE | 4 juin 2025 | Mise à jour du document |

# A.3. Validation

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Version | Nom | Fonction | | Date | Signature | Commentaires |
| V 1.3 | N. Godson Ralph SÉNATUS | | Chef de projet | 7 juin 2025 |  |  |

Table of Contents

[A.1. Identification du document 1](#_Toc199513590)

[A.2. Historique des changements 1](#_Toc199513591)

[A.3. Validation 1](#_Toc199513592)

[1. Objectif du Document 2](#_Toc199513593)

[2. Presentation Generale du Système 3](#_Toc199513594)

[3. Cas d’utilisation Fonctionnels (Use Cases) 3](#_Toc199513595)

[4. Diagrammes UML 4](#_Toc199513596)

[5. Règles métiers globales 5](#_Toc199513597)

[6. Structure de données Globales 5](#_Toc199513598)

[7. Contraintes Techniques à respecter 6](#_Toc199513599)

[8. Annexes 6](#_Toc199513600)

# Objectif du Document

Ce document a pour objectif de traduire les spécifications fonctionnelles en modèles concrets, structurés et compréhensibles par l’équipe de développement. Il sert de pont entre les besoins exprimés par les utilisateurs et la mise en œuvre technique du système. À travers la description détaillée des cas d’utilisation, des règles métier, des interactions entre les utilisateurs et l’application, ainsi que des structures de données à manipuler, ce document fournit un cadre clair et cohérent pour guider le travail des développeurs. Il vise ainsi à garantir que la solution développée réponde précisément aux attentes fonctionnelles tout en restant techniquement réalisable et maintenable.

# Presentation Générale du Système

Le système de réservation de salles est une plateforme web et mobile intuitive, conçue pour faciliter la gestion et la réservation de salles destinées aux différents événements ou cours. La plateforme vise à simplifier et automatiser les démarches liées à la disponibilité des salles, en éliminant les méthodes traditionnelles souvent manuelles, lentes et sujettes à des erreurs de planification. Grâce à ce système, les utilisateurs peuvent consulter en temps réel les créneaux disponibles, créer un compte sécurisé, se connecter à leur espace personnel, effectuer des réservations ou les modifier en quelques clics. Les administrateurs, quant à eux, disposent d’outils pour gérer les utilisateurs, les salles et les réservations globales. En centralisant les données et en assurant un suivi précis, la solution permet un gain de temps considérable, une meilleure organisation des ressources, et une expérience utilisateur optimisée. Parmi les fonctionnalités clés, on retrouve : la création de comptes, l’authentification, la consultation des salles disponibles, la réservation, la modification de réservation, ainsi que la gestion complète du système par un administrateur.

# Cas d’utilisation Fonctionnels (Use Cases)

* 1. Cas 1**: Créer un compte**

**Description** : Ce cas d’utilisation permet à un nouvel utilisateur de s’enregistrer sur la plateforme en fournissant des informations personnelles de base (nom, adresse e-mail, mot de passe, etc.). Une fois l’inscription validée, un compte utilisateur est créé dans le système.

**Règles métiers :**

* + L’adresse e-mail doit être unique dans la base.
  + Le mot de passe doit respecter des critères de sécurité (longueur minimale, caractères spéciaux).
  + Un email de confirmation peut être envoyé pour activer le compte.
  1. Cas #2**: Connexion Utilisateur**
  + **Description** : L’utilisateur s’authentifie à l’aide de son e-mail et de son mot de passe pour accéder à son espace personnel et effectuer des réservations.
  + **Règles métiers**:
* L’authentification est sécurisée avec un système de jeton (JWT).
* Après 3 tentatives de connexion échouées, un délai de blocage temporaire est appliqué.
* L’historique des connexions est conservé à des fins de sécurité.

* 1. Cas #3**: Consultation de la disponibilité des salles**
  + **Description** : Ce cas permet aux utilisateurs de visualiser en temps réel les plages d’horaires disponibles pour chaque salle.
  + **Règles métiers**:
* Seuls les créneaux disponibles seront affichés.
* Une salle qui est déjà réservée sur un créneau ne peut pas l’être à nouveau.

**3.4** Cas #4**: Réserver une salle**

* **Description :** L’utilisateur sélectionne une salle disponible, à une date et une heure donnée, et envoie une demande de réservation.
* **Règle métiers :**
* Une réservation ne peut être faite que sur un créneau libre.
* Les réservations passées ne peuvent plus être modifiées.

**3.5** Cas #5**: Modifier ou annuler une réservation**

* **Description :** L’utilisateur peut consulter ses réservations à venir et, si nécessaire, les modifier (changer d’horaire ou de salle) ou les annuler.
* **Règle métiers :**
* Une réservation ne peut être modifiée ou annulée que jusqu’à un certain délai avant l’heure prévue.

**3.6** Cas #6**: Gérer les utilisateurs**

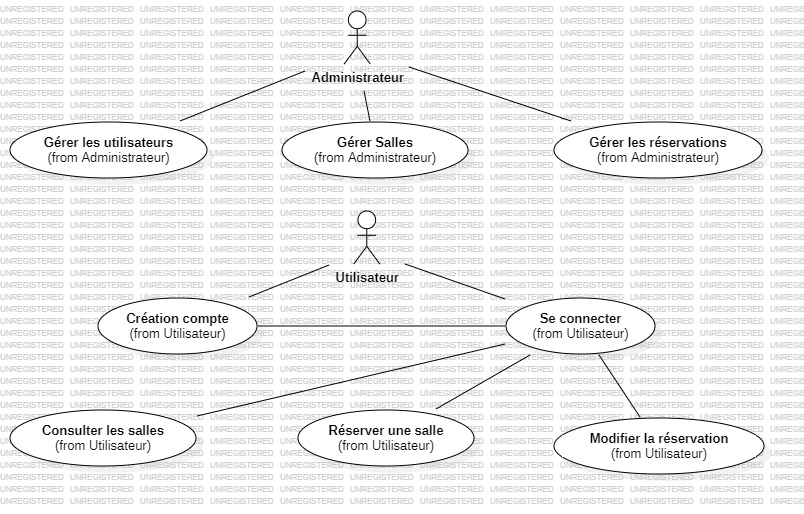
* **Description :** L’administrateur a la capacité de gérer les comptes utilisateurs : les consulter, les modifier, les supprimer.
* **Règle métiers :**
* Seuls les administrateurs authentifiés peuvent accéder à cette fonctionnalité.
* Un utilisateur supprimé ne peut plus se connecter ni faire de réservation.

**3.7** Cas #7**: Gérer les salles**

* **Description :** L’administrateur peut ajouter, modifier ou supprimer des salles dans le système, ainsi que configurer leur capacité, leur localisation, etc.
* **Règle métiers :**
* Une salle supprimée est automatiquement dissociée des futures réservations.
* Chaque salle doit avoir un identifiant unique, un nom et une capacité minimale.

# Diagrammes UML

* 1. Diagramme de **Cas d’Utilisation**

****

* 1. Diagramme de **Séquence**

On dessine un diagramme de séquence pour chaque cas d’utilisation identifié au point 3

* 1. Diagramme **d’État-Transition**

On crée un diagramme État-Transition pour chaque entité du système.

# Règles métiers globales

A ce niveau, on identifie les règles métiers que l’on doit tenir compte pour le développement et le codage pour toutes les fonctionnalités du système.

* L’adresse e-mail doit être unique dans la base.
* Le mot de passe doit respecter des critères de sécurité (longueur minimale, caractères spéciaux).
* Après 3 tentatives de connexion échouées, un délai de blocage temporaire est appliqué.
* L’historique des connexions est conservé à des fins de sécurité.
* Seuls les créneaux disponibles seront affichés.
* Une salle qui est déjà réservée sur un créneau ne peut pas l’être à nouveau.
* Une réservation ne peut être faite que sur un créneau libre.
* Les réservations passées ne peuvent plus être modifiées.
* Une réservation ne peut être modifiée ou annulée que jusqu’à un certain délai avant l’heure prévue.
* Un utilisateur supprimé ne peut plus se connecter ni faire de réservation.
* Chaque salle doit avoir un identifiant unique, un nom et une capacité minimale.

# Structure de données Globales

Dans cette partie, on identifie les structures de données qui vont permettre d’implémenter les différentes fonctionnalités du système :

* Utilisateur
  + IdUtilisateur
  + Nom
  + Prénom
  + Email
  + Mot-de-passe
  + Rôle (simple utilisateur, administrateur, gestionnaire).
* Salle
  + Idsalle
  + Nom de la salle
  + Capacité
  + Équipements de la salle
  + Disponibilité
  + Statut
* Réservation
  + IdRéservation
  + IdSalle
  + IdUtilisateur
  + Date
  + HeureDébut
  + HeureFin
* Horaire de disponibilité
  + Date
  + HeureDébut
  + HeureFin

# Contraintes Techniques à respecter

Pour assurer la fiabilité, la performance et la maintenabilité du système, plusieurs contraintes techniques doivent être rigoureusement respectées :

1. Architecture

Le système devra adopter une architecture en couches clairement définies afin de garantir une séparation des responsabilités et une meilleure évolutivité.

2. Technologies utilisées

Les technologies imposées pour le développement sont les suivantes :

* + - Backend : Node.js
    - Frontend : React.js
    - Base de données : MariaDB

3. Sécurité

La gestion des droits d’accès, l’authentification des utilisateurs et la protection contre les injections SQL doivent être assurées dès les premières phases du développement.

4. Performance

Le système devra répondre rapidement aux requêtes des utilisateurs, même en cas de forte charge. Les requêtes SQL doivent être optimisées, et les échanges client-serveur limités au strict nécessaire.

5. Compatibilité et accessibilité

L’interface utilisateur doit être compatible avec les principaux navigateurs modernes et responsive pour un accès sur différents types d’appareils (PC, tablette, mobile).

6. Sauvegarde et intégrité des données

Des mécanismes de sauvegarde régulière de la base de données doivent être prévus, ainsi qu’une gestion rigoureuse des contraintes d’intégrité.

# Annexes

Ce glossaire a pour objectif de clarifier les termes clés utilisés dans le cadre du développement du système de réservation de salle. Il permet d’unifier la compréhension entre les différents acteurs du projet en fournissant une définition précise et partagée des concepts fonctionnels.

* Réservation : Action consistant à bloquer une salle pour un créneau horaire donné, à une date précise. Une réservation peut être créée, modifiée, confirmée ou annulée selon les besoins de l’utilisateur et les règles de gestion.
* Utilisateur : Personne accédant au système via une interface d’authentification. Chaque utilisateur possède un profil et un rôle déterminant ses droits au sein de l’application.
* Disponibilité : Indique si une salle est libre ou déjà réservée à un moment donné. Elle est calculée automatiquement par le système en fonction des réservations existantes.
* Salle : Espace physique pouvant être réservé. Chaque salle est identifiée par un nom et un code, possède une capacité, et peut être équipée de matériels spécifiques.
* Calendrier de réservation : Vue interactive ou tableau permettant aux utilisateurs de consulter les disponibilités des salles sur une période donnée.
* Notification : Message automatique envoyé à un utilisateur pour l’informer d’un événement important.