République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université M'hamed Bougara - Boumerdès



Faculté des Sciences
Département d'Informatique

Domaine : Mathématiques Informatique

Filière : Informatique

Spécialité : Systèmes d'informatique

N° de l'Arrêté d'habilitation de la spécialité : arrêté n °872 du 26/07/2016

Mémoire de fin d'études en vue de l'obtention du
Diplôme de Licence Académique

Thème

Application web pour la gestion d'évènements et réservation

Présenté par :
Bouaffad Salim
Kherroub Fares
Loulou Ikram Lina
Rahmani Selsabil

Soutenu le ../../2026 Devant le jury composé de
Mr.REZOUG.A : Examinateur
Mme.OULD BABAALI .K : Encadreur

Remerciements

Tout d'abord, nous tenons à remercier Dieu Tout-Puissant de nous avoir accordé la santé, la patience et la force nécessaires pour mener à bien ce mémoire malgré toutes les difficultés rencontrées.

Nous souhaitons également exprimer nos sincères remerciements à Mme OULD BABAALI.K pour son accompagnement, ses précieux conseils et son soutien tout au long de ce travail. Nous remercions également M.REZOUG.A d'avoir accepté de participer en tant qu'examinateur à notre soutenance.

Nous adressons aussi nos plus sincères remerciements à nos parents, à nos familles ainsi qu'à tous nos proches qui nous ont accompagnés, aidés, soutenus et encouragés tout au long de la réalisation de ce mémoire.

Enfin, nous adressons nos remerciements à l'Université M'Hamed Bougara de Boumerdès, Faculté des Sciences, pour la formation qu'elle nous a dispensée.

Table des matières :

Introduction générale....5
Chapitre 1 :....7
Étude de l'existant....7
1.1 Introduction :....8
1.2 Problématique :....8
1.3 Solutions proposées :....8
1.4 Conclusion :....9
Chapitre 2 :....10
Conception et Analyse....10
2.1 Introduction :....11
2.2 Identification des acteurs:....11
2.3 Règles de Gestion :....11
2.4 Le diagramme général de cas d'utilisation :....13
2.5 Tableau descriptif textuel :....15
2.6 Le diagramme de classe :....16
2.7 Les diagrammes de séquence....17
2.8 Conclusion :....19
Chapitre 3 : Réalisation....20
3.1 Introduction....21
3.2 Environnements et langages de programmation....21
3.2.1 JSP (Java Server Pages) :....21
3.2.2 Java et Spring Boot :....21
3.2.3 HTML et CSS :....21
3.3 Outils de développement....22
3.3.1 IntelliJ IDEA....22
3.3.2 MySQL....22
3.3.3 Visual Studio Code:....22
3.3.4 Draw.io....23
3.4 Dictionnaire de données....23
3.5 Présentation de quelques interfaces de l'application....26
3.5.1 Page de bienvenue....26
3.5.2 Page d'accueil — Liste des événements....27
3.5.3 Interface de paiement....28
3.5.4 Interface de connexion professionnelle....28
3.5.5 Espace administrateur....29
3.5.6 Espace organisateur....30
3.6 Conclusion....30
Conclusion Générale....31
Bibliographie....32

Table des figures

Figure 1 : Diagramme général des cas d'utilisation
Figure 2 : Diagramme des cas d'utilisation pour l'utilisateur
Figure 3 : Diagramme des cas d'utilisation pour l'organisateur
Figure 4 : Diagramme des cas d'utilisation pour l'Administrateur
Figure 5 : Diagramme des classes
Figure 6 **:** Diagramme de séquence : s'authentifier
Figure 7**:** Diagramme de séquence : Organisateur_ Créer un évènement
Figure 8: Diagramme de séquence : Utilisateur réserver un évènement
Figure 9**:** Diagramme de séquence : Administrateur_consulter et valider les demandes d'évènements
Figure 10 : Page de bienvenue de l'application web
Figure 11 : Page d'accueil — Liste des événements disponibles
Figure 12 : Interface de paiement (en ligne ou sur place)
Figure 13 : Interface de connexion professionnelle
Figure 14 : Interface de l'espace administrateur
Figure 15 : Interface de l'espace organisateur

Table des tableaux

Tableau 1 : Description textuelle du cas d'utilisation : s'authentifier
Tableau 2 : Description textuelle du cas d'utilisation : Réserver un évènement
Tableau 3 : Description textuelle du cas d'utilisation : Créer un nouvel évènement
Tableau 4 : Description textuelle du cas d'utilisation : Consulter et valider les demandes de création et modification d'évènement
Tableau 5 : Compte
Tableau 6 : Administrateur
Tableau 7 : Organisateur
Tableau 8 : Utilisateur
Tableau 9 : Évènement
Tableau 10 : Salle
Tableau 11 : Réservation
Tableau 12 : Paiement
Tableau 13 : Paiement En ligne
Tableau 14 : Paiement Sur place
Tableau 15 : Paiement Création

Résumé

Dans le cadre de notre licence en Systèmes Informatique à l'Université de Boumerdès, nous avons été invités à réaliser un projet de fin d'études.

Ce projet nous a permis d'approfondir nos connaissances théoriques et techniques en concevant et en développant une application web simple de réservation d'événements.

Cette application a été conçue pour faciliter la gestion des réservations et offrir une expérience utilisateur optimale.

Son principal objectif est d'éviter les doubles réservations et les dépassements de capacité.

Introduction générale

L'organisation d'événements est devenue aujourd'hui un élément essentiel dans différents domaines tels que l'éducation, les entreprises, la culture, le divertissement, le domaine associatif, le secteur sportif, touristique et politique. Elle est en constante évolution, requiert des outils technologiques innovants pour répondre aux besoins croissants des utilisateurs et améliorer l'efficacité opérationnelle des établissements.

Dans ce contexte, ce mémoire se concentre sur la conception et la réalisation d'une application web dédiée à la gestion d'événements et de réservations. Cette application vise à simplifier le processus de réservation pour les utilisateurs, tout en offrant un système de gestion flexible et performant aux organisateurs des événements.

Les applications web représentent une solution moderne permettant de faciliter la gestion des événements, d'automatiser les processus de réservation et d'améliorer l'expérience des utilisateurs. Elles offrent une meilleure accessibilité, une centralisation des données et une gestion efficace des informations en temps réel.

La gestion des événements peut présenter des difficultés et plusieurs défis techniques concernant l'organisation des inscriptions et le risque de double réservation par un seul utilisateur dans le même événement. Cette situation engendre des problèmes dans la base de données, désorganise l'organisation de l'événement et réduit la fiabilité du système.

Pour résoudre ces problèmes, nous proposons une application web de gestion des événements et réservations. Cette plateforme centralisera la gestion des réservations, offrant une interface simple permettant aux utilisateurs de consulter les événements, effectuer des réservations et interagir facilement avec la plateforme.

Notre application web vise à offrir une plateforme centralisée permettant aux organisateurs de créer et gérer leurs événements, et aux utilisateurs de consulter les événements disponibles et d'effectuer leurs réservations facilement, tout en garantissant la fiabilité du système grâce à des mécanismes de contrôle empêchant les doubles réservations et le dépassement de capacité. Pour atteindre cet objectif, nous appliquons les points suivants :

1. Permettre aux utilisateurs de s'inscrire et de réserver une place facilement.

2. Offrir une vérification qui empêche toute double réservation.

3. Conserver un historique des réservations pour chaque utilisateur.

Dans ce contexte, notre projet consiste à concevoir et développer une application web de gestion d'événements et de réservations permettant d'assurer une organisation optimale, sécurisée et efficace.

Pour cela, nous avons divisé notre mémoire en trois chapitres principaux :

*Chapitre 1 : (Étude de l'existant)

Une introduction, problématique, solution, conclusion.

*Chapitre 2 : (Conception)

Étudier la conception de l'application où nous allons étudier les fonctionnalités ainsi que la modélisation des diagrammes UML.

*Chapitre 3 : (Réalisation)

Présenter la réalisation de l’application et expliquer ses fonctionnalités.

Nous clôturons ce mémoire par une conclusion générale résumant les points essentiels de notre travail et dégageant quelques perspectives envisagées pour notre application.

Chapitre 1 :

Étude de l'existant

1.1 Introduction :

Ce chapitre contient le cadre général du projet, une analyse critique des systèmes existants et la problématique étudiée ainsi que les solutions visées par la réalisation de l'application web de gestion d'événements et de réservations qui s'adresse aux organisateurs afin de créer et gérer leurs événements ainsi qu'aux utilisateurs de s'inscrire et de réserver une place.

1.2 Problématique :

Actuellement, il y a beaucoup d'organisations qui gèrent encore leurs événements manuellement avec des formulaires papier ou avec des systèmes très basiques, ce qui pose plusieurs problèmes

Le premier problème est le risque de double réservation par un seul utilisateur. Cela peut arriver lorsqu'un utilisateur clique plusieurs fois sur le bouton de réservation ou utilise plusieurs onglets en même temps, ce qui entraîne des incohérences dans les données et perturbe l'organisation de l'événement.

Le deuxième problème est le dépassement de capacité des événements. En effet, l'absence d'un mécanisme de contrôle du nombre de participants peut permettre à un nombre élevé d'utilisateurs de réserver pour un même événement, cette situation provoque un dépassement de capacité.

Il est donc nécessaire de développer un système de gestion d'événements et de réservations qui soit plus efficace et fonctionnel.

1.3 Solutions proposées :

Afin de résoudre les problèmes importants qui ont été discutés précédemment dans la gestion des événements et des réservations, notre projet propose les solutions suivantes :

*Vérifier que l'utilisateur n'a pas déjà effectué une réservation pour le même événement :

Cette solution consiste à vérifier si l'utilisateur a déjà effectué une réservation. Si c'est le cas et qu'il souhaite réserver à nouveau pour le même événement, sa réservation sera refusée.

*Vérifier que la capacité maximale de l'événement n'est pas atteinte :

Cette solution consiste à définir une capacité maximale pour chaque événement. Lorsqu'un événement atteint cette limite, le système bloque automatiquement toute nouvelle réservation.

1.4 Conclusion :

En conclusion, le travail effectué dans la première partie nous a présenté le cadre général de notre projet, ainsi qu'une analyse critique des systèmes existants. Ensuite, nous avons défini notre problématique et quelques solutions visées. Dans la prochaine étape, nous aborderons une analyse détaillée des besoins ainsi que la conception du système proposé.

Chapitre 2 :

Conception et Analyse

2.1 Introduction :

Dans ce chapitre nous allons présenter le processus de conception et la modélisation de notre site web. Nous utilisons des diagrammes UML (Unified Modeling Language) pour représenter clairement ces dynamiques, UML est un langage de modélisation standard utilisé dans le domaine du génie logiciel. Nous allons commencer par la définition des acteurs puis la mise en place des différents diagrammes (cas d'utilisation, classe, séquence).

Cette partie va nous permettre de bien présenter les fonctionnalités de notre site web et montrer la façon de son utilisation.

2.2 Identification des acteurs:

Dans notre système nous avons trois acteurs qui participent, sont distingués :

*__Utilisateur__ : C'est une personne qui effectue des réservations, il a la possibilité de :

-S'identifier dans le système.

-réserver une place à un événement.

-consulter ses réservations passées.

-consulter les événements.

*__Organisateurs__ : Les organisateurs peuvent gérer leurs propres évènements. Ils assurent :

-S'inscrire dans le système.

-La gestion des événements (créer, modifier, supprimer).

-Consulter les réservations de leurs événements.

*__Administrateur__ : L'administrateur est l'acteur responsable de la structure générale de l'application web. Il a tous les droits d'accès dans le système pour assurer le bon fonctionnement de l'application.

Son rôle principal est :

-Contrôler tous les évènements existants dans le système.

-Gérer les comptes des utilisateurs et des organisateurs.

2.3 Règles de Gestion :

RG01 : Inscription

-L'organisateur soumet une demande de création de compte en fournissant une adresse e-mail et un mot de passe unique.

-Le compte ne peut être activé qu'après validation explicite par l'administrateur.

RG02 : Authentification

-L'organisateur et l'administrateur s'authentifient via leur e-mail, leur mot de passe et leur rôle pour accéder à leur espace personnel.

RG03 : Identification de l'utilisateur

-L'utilisateur s'identifie via son e-mail et son numéro de téléphone. Un code PIN est envoyé par email pour confirmer son identité. Si l'utilisateur n'existe pas encore, un compte lui est créé automatiquement.

RG04 : Création d'un événement

-Seul un organisateur authentifié peut créer un événement.

- Un événement doit obligatoirement contenir : un titre, une description, une date, une capacité maximale, au moins un choix de salle et un paiement de création de 5 000 DA.

-Tout événement créé est automatiquement mis en statut « EN_ATTENTE ».

RG05 : Validation d'un événement

-Seul l'administrateur peut accepter ou refuser un événement.

- Un événement n'est visible sur la page d'accueil que s'il est au statut «ACCEPTE».

RG06 : Modification d'un événement

-Un organisateur peut modifier uniquement ses propres événements en statut « En attente »

ou « REFUSE ».

-Un événement au statut «ACCEPTE» ne peut en aucun cas être modifié, quelle que soit sa date.

-Après toute modification, l'événement repasse automatiquement au statut « EN_ATTENTE » jusqu'à validation par l'administrateur.

RG07 : Suppression d'un événement

-Un événement «ACCEPTE» dont la date est future ou en cours ne peut être supprimé par personne.

-L'administrateur peut supprimer tout autre événement.

-Un organisateur peut supprimer ses propres événements uniquement s'ils sont en statut « EN_ATTENTE », « REFUSE », ou «ACCEPTE» avec une date passée.

RG08 : Réservation

-Un utilisateur ne peut pas réserver deux fois le même événement.

-La réservation est impossible si l'événement a atteint sa capacité maximale.

RG09 : Paiement

-Chaque réservation donne lieu à un paiement, effectué soit en ligne (référence TXN-XXXXXX) soit sur place (référence REF-XXXXXX).

-Chaque paiement est tracé par une référence unique garantissant la traçabilité des transactions.

RG10 : Unicité des identifiants

-L'email et le numéro de téléphone d'un utilisateur doivent être uniques dans le système.

-L'email d'un organisateur doit être unique dans le système.

-Chaque évènement est identifié par un code unique au format EVT-AAAAMMJJ-XXX généré automatiquement.

RG11 : Gestion des comptes

-L'administrateur peut supprimer tout compte utilisateur ou organisateur.

-L'administrateur peut ajouter un organisateur directement.

RG12 : Consultation

-La consultation des événements reste accessible sans identification.

-Un utilisateur consulte uniquement ses propres réservations.

-Un organisateur consulte uniquement ses propres évènements et les réservations associées.

-L'administrateur consulte la liste globale de tous les événements et de tous les comptes.

2.4 Le diagramme général de cas d'utilisation :

Pour permettre de faire la différence entre les acteurs et le système. Nous commençons par présenter le diagramme général de cas d'utilisation. Cet outil permet de représenter les fonctions du système et les responsabilités de chaque acteur.

Ce diagramme général fournit ainsi une représentation globale et organisée de l'ensemble des services proposés par l'application.

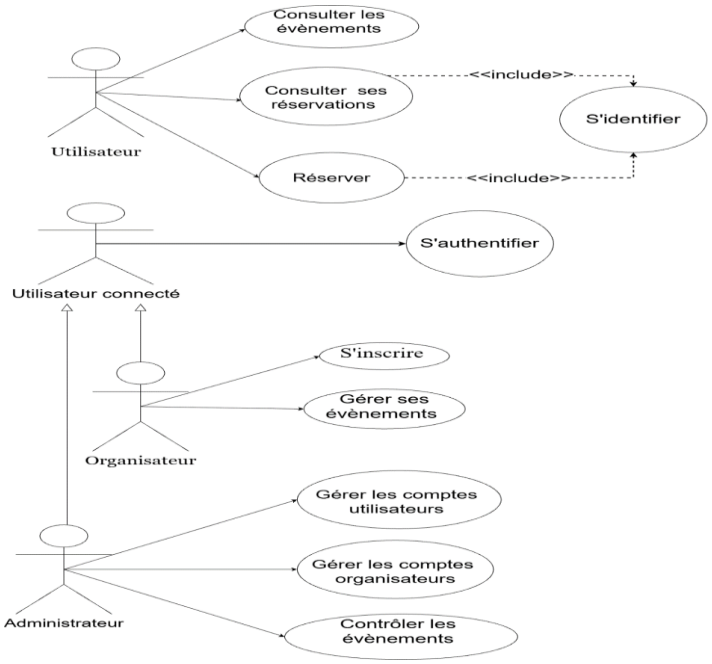


Figure 1: Diagramme général des cas d'utilisation

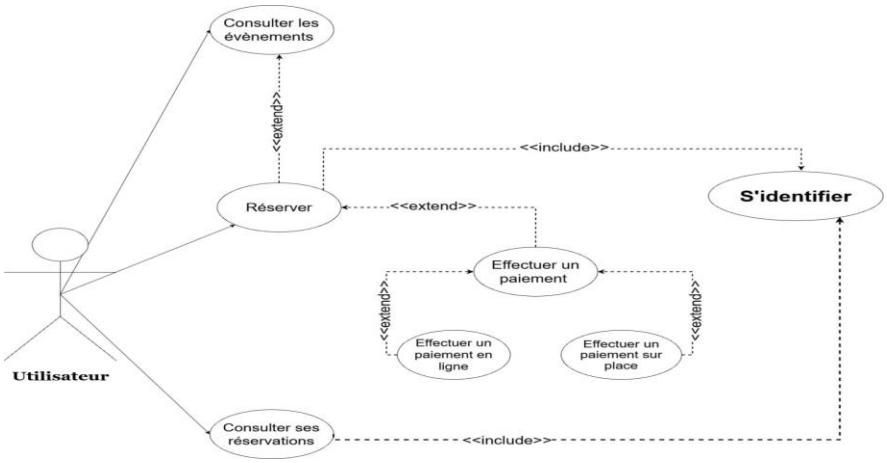


Figure 2: Diagramme des cas d'utilisation pour l'utilisateur

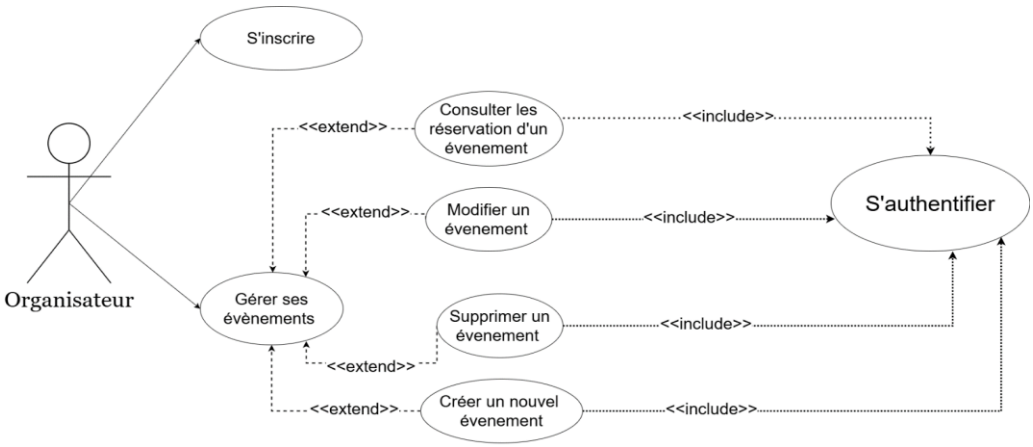


Figure3: Diagramme des cas d'utilisation pour l'organisateur



Figure4: Diagramme des cas d'utilisation pour Administrateur

2.5 Tableau descriptif textuel :

Tableau 1: Description textuelle du cas d'utilisation : s'authentifier

Élément	Description
Nom	S'authentifier
Objectif	Permettre à un acteur identifié d'accéder à son espace personnel et d'effectuer ses actions dans le système.
Acteurs	Organisateur, Administrateur
Pré conditions	L'acteur dispose d'un compte actif dans le système. Pour l'organisateur, le compte doit avoir été validé par l'administrateur.
Post conditions	Les acteurs sont redirigés vers leur espace personnel

Tableau 2 : Description textuelle du cas d'utilisation: Réserver un évènement

Élément	Description
Nom	Réserver un évènement
Objectif	Permettre à un utilisateur de réserver sa place dans un évènement avec toute simplicité.
Acteur	Utilisateur
Pré condition	Au moins un évènement est disponible
Post condition	La réservation est effectuée avec succès

Tableau 3 : Description textuelle du cas d'utilisation : Créer un nouvel évènement

Élément	Description
Nom	Créer un nouvel événement
Objectif	Permettre à un organisateur de créer un événement (titre, description, date,etc.)
Acteur	Organisateur
Pré condition	L'organisateur doit être authentifié dans le système Les informations de l'événement sont complètes (titre, date,etc.)
Post condition	L'événement est sauvegardé dans le système avec le statut EN_ATTENTE.

Tableau 4 : Description textuelle du cas d'utilisation : Consulter les demandes de création et modification d'événement

Élément	Description
Nom	Consulter et valider les demandes de création et modification d'événement
Objectif	Permettre à un administrateur d'accepter ou refuser les demandes d'événement soit un événement déjà créé ou un événement modifié.
Acteur	Administrateur
Pré condition	L'événement doit exister dans le système
Post condition	L'événement est mis à jour avec statut ACCEPTE ou REFUSE

2.6 Le diagramme des classes

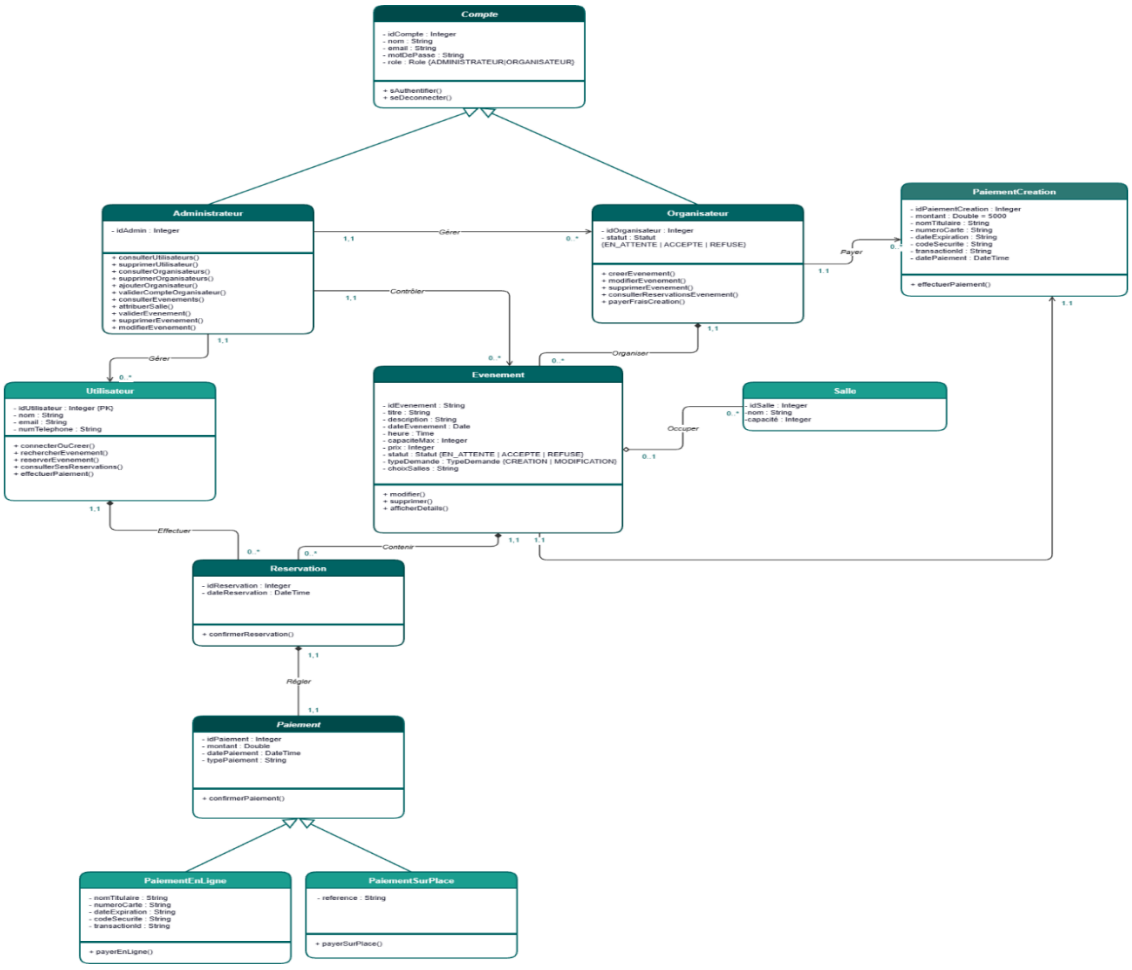


Figure 5: Diagramme des classes

2.7 Les diagrammes de séquence :

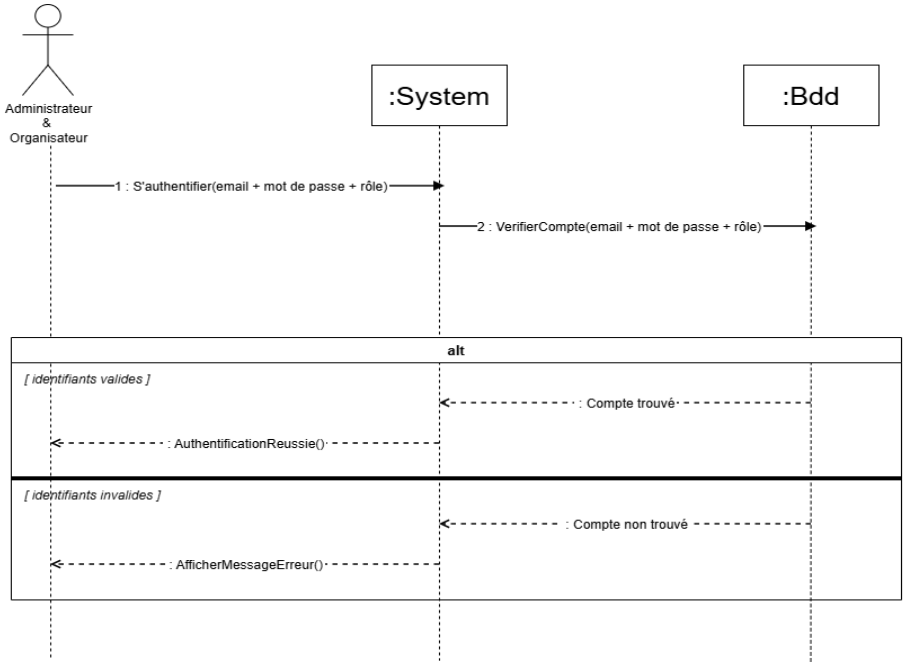


Figure 6: Diagramme de séquence : s'authentifier

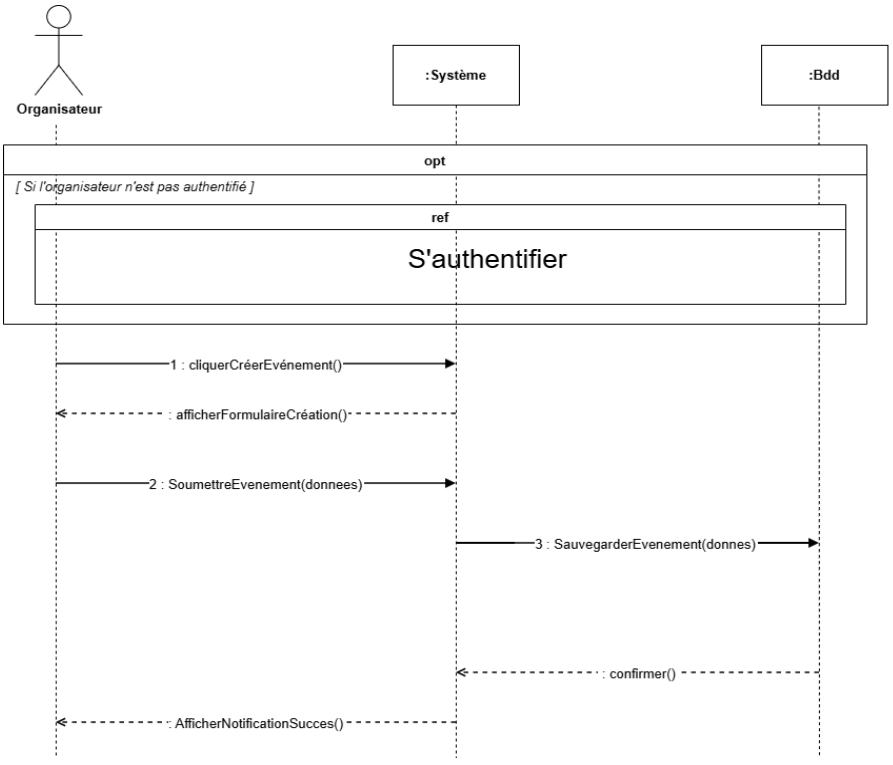


Figure 7: Diagramme de séquence : Organisateur_ Créer un évènement

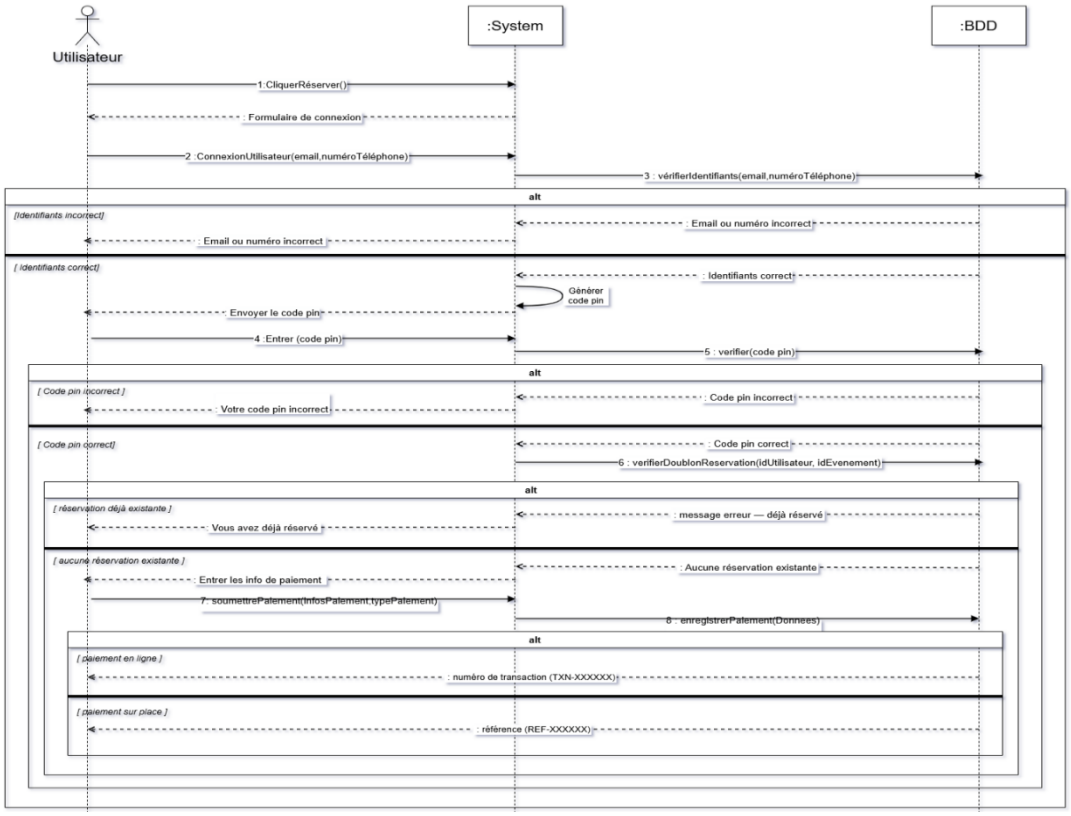


Figure 8: Diagramme de séquence : Utilisateur_Réserver un évènement

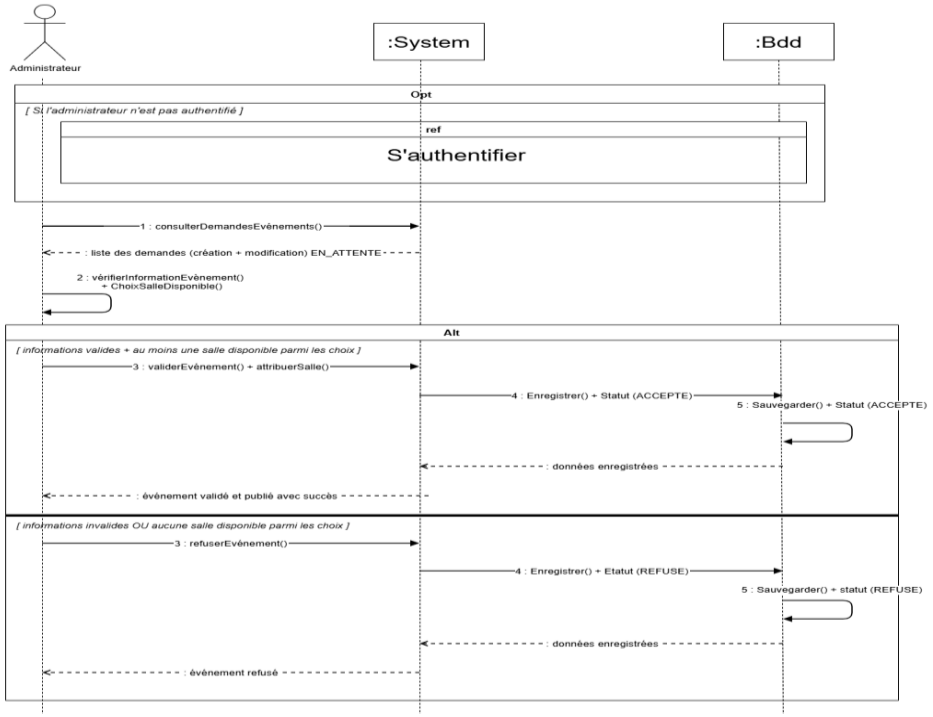


Figure 9: Diagramme de séquence :Administrateur_consulter et valider les demandes d'évènements

2.8 Conclusion :

Ce chapitre de conception et analyse a été essentiel pour jeter les bases solides de notre projet de gestion d'évènement et de réservation. Les diagrammes des cas d'utilisation, des classes et des séquences ont fourni une version détaillée de la structure et du fonctionnement du système.

D'autre part, nous avons établi un cadre précis qui guidera le développement de l'application. Ainsi, ce chapitre nous a fourni une base solide sur laquelle nous pourrons construire et développer le système de gestion d'évènement et de réservation avec confiance et clarté.

Chapitre 3 :

Réalisation

3.1 Introduction

Dans ce chapitre, nous entamons la dernière partie du projet qui est la réalisation. Nous présenterons les langages de programmation utilisés pour développer notre application web, les outils de développement, ainsi que le dictionnaire de données décrivant la structure de notre base de données. Nous allons présenter également quelques interfaces principales de notre application web sous forme de captures d'écran.

3.2 Environnements et langages de programmation

3.2.1 JSP (Java Server Pages) :

Java Server Pages (JSP) est une technologie développée par Sun Microsystems, désormais maintenue par Oracle, qui permet la création de pages web dynamiques basées sur le langage Java. Selon Wikipédia, « les JSP permettent d'intégrer du code Java dans des balises HTML, permettant ainsi la génération de contenu dynamique côté serveur ». Les fichiers JSP sont compilés par le serveur en servlets Java lors de leur première requête, ce qui garantit des performances optimales lors des accès ultérieurs. Dans notre projet, JSP est utilisé pour la couche de présentation de l'application, en combinaison avec la bibliothèque JSTL (Jakarta Standard Tag Library) pour le rendu dynamique des données. [4]

3.2.2 Java et Spring Boot :

Java est un langage de programmation orienté objet, portable et fortement typé, largement utilisé dans le développement d'applications web et d'entreprise.

Spring Boot est un framework basé sur Spring Framework qui simplifie la configuration et le déploiement d'applications Java. Il intègre un serveur Tomcat embarqué, permettant de lancer l'application sans configuration externe. Dans notre projet, Spring Boot est utilisé pour implémenter l'architecture MVC (Model-View-Controller) avec les couches Controller, Service et Repository. [5]

3.2.3 HTML et CSS :

HTML (HyperText Markup Language) est le langage de balisage standard pour la création de pages web. Il permet de structurer le contenu des pages en définissant les éléments tels que les titres, les paragraphes, les tableaux et les formulaires.

CSS (Cascading Style Sheets) est le langage de feuilles de style utilisé pour la mise en forme et le design des pages web. Il permet de définir l'apparence visuelle des éléments HTML,

notamment les couleurs, les polices, les marges et la disposition des éléments.Dans notre projet, HTML et CSS sont intégrés directement dans nos fichiers JSP pour structurer et styliser l'interface utilisateur de l'application, en assurant un rendu cohérent sur l'ensemble des pages. [6]

3.3 Outils de développement

3.3.1 IntelliJ IDEA

IntelliJ IDEA est un environnement de développement intégré (IDE) développé par JetBrains, spécialement conçu pour le développement d'applications Java. Selon Wikipédia, « IntelliJ IDEA est disponible en deux versions : une version communautaire open source sous licence Apache 2, et une version propriétaire commerciale ». Il offre des fonctionnalités avancées telles que la complétion automatique du code, le refactoring, le débogage intégré et la gestion des dépendances Maven. C'est l'IDE principal utilisé pour développer notre application Spring Boot. [7]



3.3.2 MySQL

MySQL est un système de gestion de bases de données relationnelles (SGBDR) open source, développé par Oracle Corporation. Selon Wikipédia, « MySQL est l'un des SGBD les plus utilisés au monde, notamment dans les applications web ». Il utilise le langage SQL pour la manipulation et l'interrogation des données. Dans notre projet, MySQL est utilisé pour stocker l'ensemble des données de l'application : comptes, événements, réservations et paiements. [8]



3.3.3 Visual Studio Code (outil secondaire) :

Visual Studio Code est un éditeur de code source gratuit et open source développé par Microsoft. Il permet de programmer dans plusieurs langages et propose des fonctionnalités comme l'autocomplétion, le débogage, la coloration syntaxique et l'intégration de Git. Il est disponible sur Windows, Linux et macOS. [9]



3.3.4 Draw.io

Draw.io (également connu sous le nom de diagrams.net) est une application web gratuite et open source permettant la création de diagrammes et de schémas vectoriels. Elle supporte de nombreux types de diagrammes, notamment les diagrammes UML, les organigrammes et les schémas d'architecture. Nous avons utilisé Draw.io pour concevoir l'ensemble de nos diagrammes UML (diagramme de classes, diagrammes de séquence et diagrammes de cas d'utilisation). [10]



3.4 Dictionnaire de données

Définition : Le dictionnaire de données est un document qui décrit la structure de la base de données. Il contient les tables, les champs, leurs types ainsi que les contraintes associées. Il permet de mieux comprendre l'organisation et la gestion des données dans l'application.

Tableau 5 : Compte

Champ	Type	Description	Contraintes
Id_compte	Integer	Identifiant unique du compte	Clé primaire
nom	String	Nom complet du titulaire	NOT NULL
email	String	Adresse email du compte	Unique, NOT NULL
Mot_de_passe	String	Mot de passe du compte	NOT NULL
role	Enum	Rôle du compte	ADMINISTRATEUR / ORGANISATEUR

Tableau 6 : Administrateur

Champ	Type	Description	Contraintes
Id_admin	Integer	Identifiant unique de l'administrateur	Clé primaire
Id_compte	Integer	Référence vers la table compte	Clé étrangère

Tableau 7 : Organisateur

Champ	Type	Description	Contraintes
Id_organisateur	Integer	Identifiant unique de l'organisateur	Clé primaire
Statut	Enum	Statut de l'organisateur	EN_ATTENTE / ACCEPTE / REFUSE

Id_compte	Integer	Référence vers la table compte	Clé étrangère
Id_admin	Integer	Référence vers la table administrateur	Clé étrangère

Tableau 8 : Utilisateur

Champ	Type	Description	Contraintes
Nom_utilisateur	string	Nom complet de l'utilisateur	NOT NULL
Id_utilisateur	Integer	Identifiant unique de l'utilisateur	Clé primaire
email	String	Adresse email de l'utilisateur	NOT NULL
Num_telephone	String	Numéro de téléphone	NOT NULL
id_admin	Integer	Référence vers la table administrateur	Clé étrangère

Tableau 9 : Evenement

Champ	Type	Description	Contraintes
Id_evenement	String	Identifiant unique de l'événement (EVT-YYYYMMDD-XXX)	Clé primaire
titre	String	Titre de l'événement	NOT NULL
description	String	Description de l'événement	
Date_evenement	Date	Date de l'événement	NOT NULL
heure	Time	Heure de l'événement	NOT NULL
Capacite_max	Integer	Capacité maximale de l'événement	NOT NULL
prix	Integer	Prix de l'événement en DA	NOT NULL
statut	Enum	Statut de l'événement	EN_ATTENTE / ACCEPTE / REFUSE
type_demande	Enum	Type de demande	CREATION / MODIFICATION
choix_salles	String	Choix des salles par l'organisateur	
id_salle	Integer	Référence vers la table salle	Clé étrangère
id_organisateur	Integer	Référence vers la table organisateur	Clé étrangère
id_admin	Integer	Référence vers la table administrateur	Clé étrangère

Tableau 10 : Salle

Champ	Type	Description	Contraintes
Id_salle	Integer	Identifiant unique de la salle	Clé primaire
nom	String	Nom de la salle	NOT NULL
capacite	Integer	Capacité de la salle	NOT NULL
id_admin	Integer	Référence vers la table administrateur	Clé étrangère

Tableau 11 : Réservation

Champ	Type	Description	Contraintes
Id_reservation	Integer	Identifiant unique de la réservation	Clé primaire
Date_reservation	DateTime	Date de réservation	DEFAULT CURRENT_TIMESTAMP
id_utilisateur	Integer	Référence vers la table utilisateur	Clé étrangère
id_evenement	String	Référence vers la table événement	Clé étrangère

Tableau 12 : Paiement

Champ	Type	Description	Contraintes
Id_paiement	Integer	Identifiant unique du paiement	Clé primaire
montant	Double	Montant payé	NOT NULL
Date_paiement	DateTime	Date du paiement	DEFAULT CURRENT_TIMESTAMP
Type_paiement	String	Type de paiement	EN_LIGNE / SUR_PLACE
id_reservation	Integer	Référence vers la table réservation	Clé étrangère

Tableau 13 : PaiementEnLigne

Champ	Type	Description	Contraintes
Id_paiement_en_ligne	Integer	Identifiant unique du paiement en ligne	Clé primaire
Nom_titulaire	String	Nom du titulaire de la carte	NOT NULL
Numero_carte	String	Numéro de carte bancaire	NOT NULL
Transaction_id	String	Identifiant de transaction (TXN-XXXXXX)	NOT NULL
Date_expiration	String	Date d'expiration de la carte	NOT NULL
Code_securite	String	Code de sécurité de la carte	NOT NULL
id_paiement	Integer	Référence vers la table paiement	Clé étrangère

Tableau 14 : PaiementSurPlace

Champ	Type	Description	Contraintes
Id_paiement_sur_place	Integer	Identifiant unique du paiement sur place	Clé primaire
reference	String	Référence du paiement (REF-XXXXXX)	NOT NULL
id_paiement	Integer	Référence vers la table paiement	Clé étrangère

Tableau 15 : PaiementCreation

Champ	Type	Description	Contraintes
Id_paiement_creation	Integer	Identifiant unique du paiement de création	Clé primaire
montant	Double	Montant à payer	DEFAULT 5000.00
Nom_titulaire	String	Nom complet du titulaire	NOT NULL
Numero_carte	String	Numéro de la carte	NOT NULL
Date_expiration	String	Date d'expiration de la carte	NOT NULL
Code_securite	String	Code de sécurité de la carte	NOT NULL
Date_paiement	DateTime	Date du paiement	DEFAULT CURRENT_TIMESTAMP
Transaction_id	String	Identifiant de transaction (TXN-XXXXXX)	NOT NULL
id_organisateur	Integer	Référence vers la table organisateur	Clé étrangère
Id_évènement	String	Référence vers la table événement	Clé étrangère

Note : Format de l'identifiant de l'événement

L'identifiant de l'événement est généré automatiquement selon le format suivant :

EVT-YYYYMMDD-XXX, où :

- EVT : préfixe fixe identifiant le type d'entité.
- YYYYMMDD : date de création de l'événement.
- XXX : numéro séquentiel sur trois chiffres.

Exemple : EVT-20260509-001 désigne le premier événement créé le 9 mai 2026.

3.5 Présentation de quelques interfaces de l'application

Dans cette section, nous présentons quelques captures d'écran des principales interfaces graphiques de l'application web de gestion d'événements et de réservations. De plus, nous avons conçu une interface simple et intuitive afin de faciliter le processus de réservation et d'offrir une expérience utilisateur optimale.

3.5.1 Page de bienvenue

La page de bienvenue constitue le point d'entrée de l'application. Elle présente deux options principales :

Entrer comme un utilisateur, cette option permet à l'utilisateur de visiter la plateforme et consulter les évènements disponibles ainsi que de réserver. Espace professionnel réservé aux organisateurs et à l'administrateur.

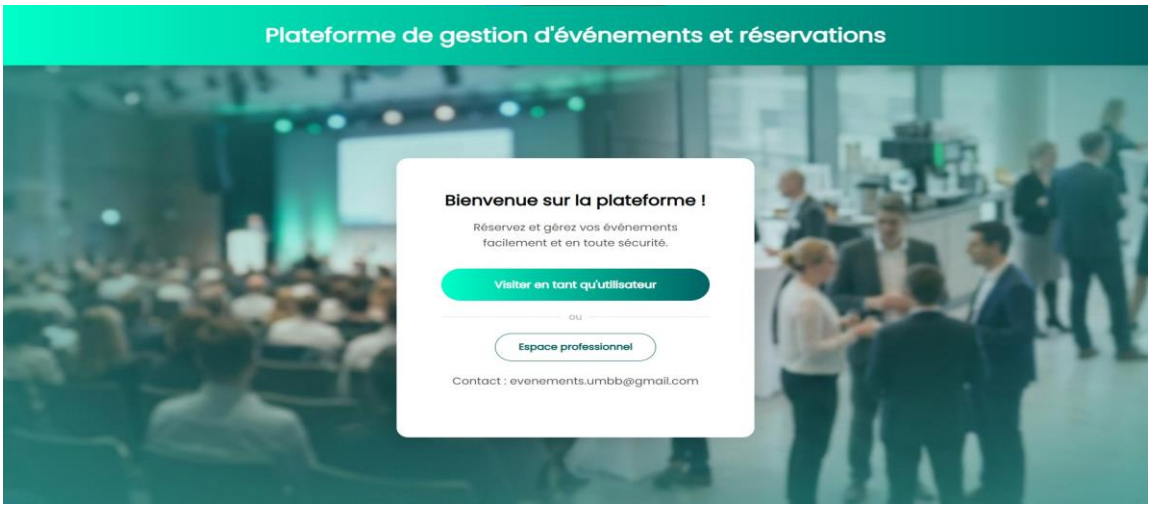


Figure 10 : Page de bienvenue de l'application web.

3.5.2 Page d'accueil — Liste des événements

La page d'accueil contient l'ensemble des événements disponibles et validés par l'administrateur. Chaque événement est présenté sous forme de carte contenant toutes les informations correspondantes. L'utilisateur peut directement cliquer sur le bouton « Réserver » pour effectuer une réservation. De même, le bouton « voir mes réservations » permet à l'utilisateur de consulter toutes ses réservations après s'être connecté.

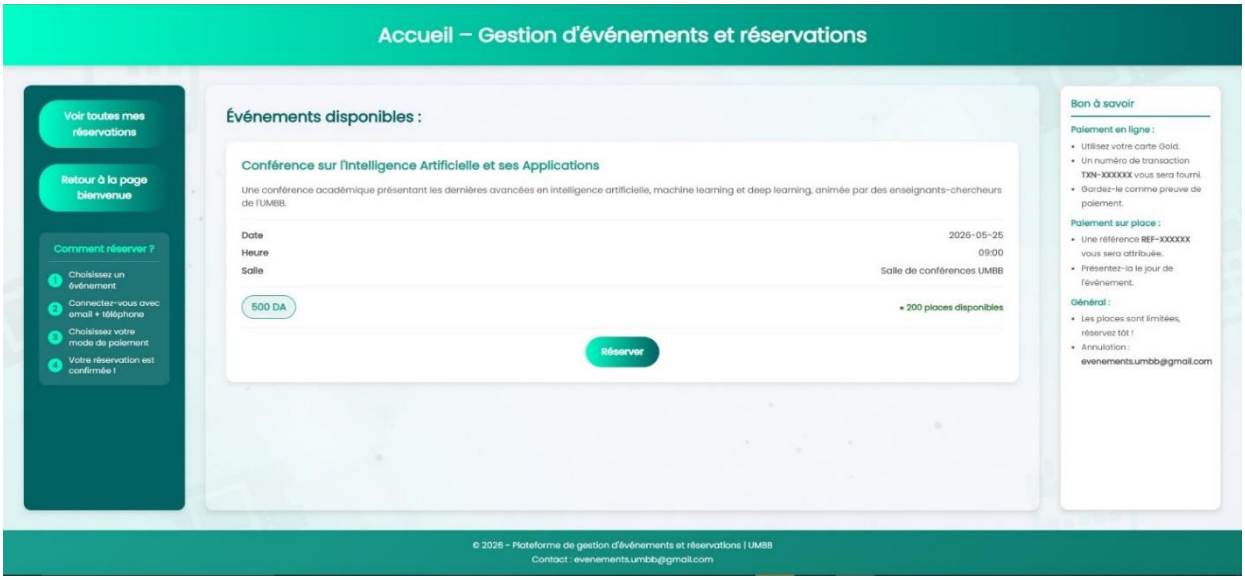


Figure 11 : Page d'accueil — Liste des événements disponibles.

3.5.3 Interface de paiement

Après avoir cliqué sur le bouton réserver, l'utilisateur est redirigé vers la page de paiement qui contient les détails de l'événement sous forme de carte récapitulative. Deux options lui sont proposées : le paiement en ligne par carte bancaire, qui génère un numéro de transaction (TXN-XXXXXX), ou le paiement sur place, qui génère une référence (REF-XXXXXX) à présenter le jour de l'événement.

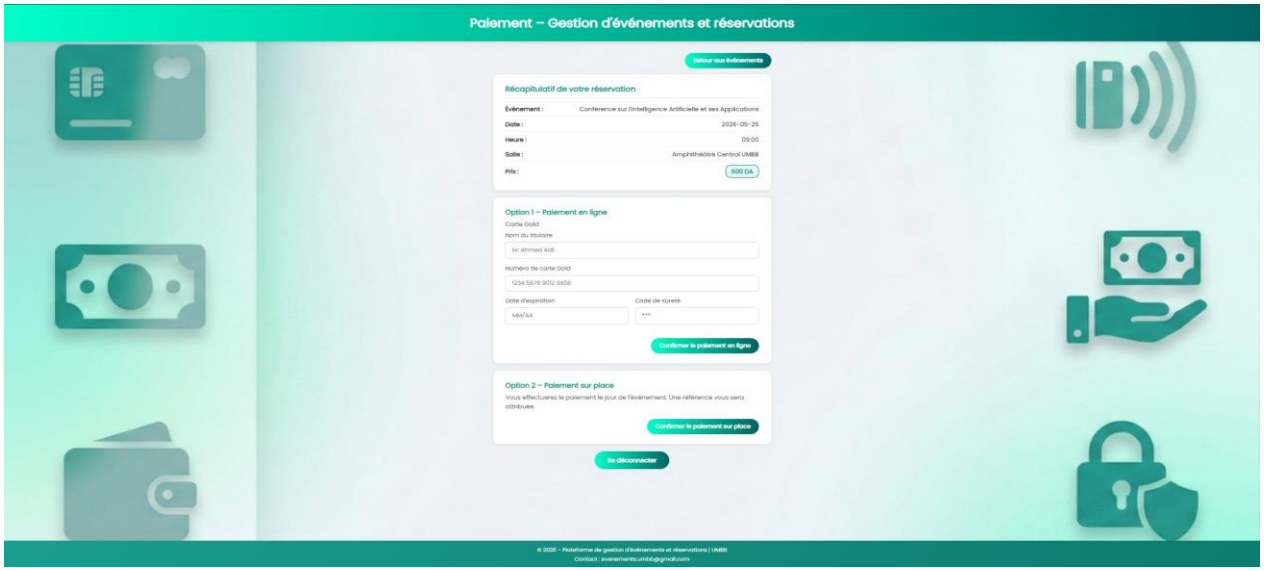


Figure 12 : Interface de paiement (en ligne ou sur place).

3.5.4 Interface de connexion professionnelle

L'interface de connexion professionnelle est réservée aux organisateurs et à l'administrateur. L'acteur doit saisir son adresse email, son mot de passe ainsi que son rôle. Le système vérifie les identifiants puis le redirige vers son espace personnel.

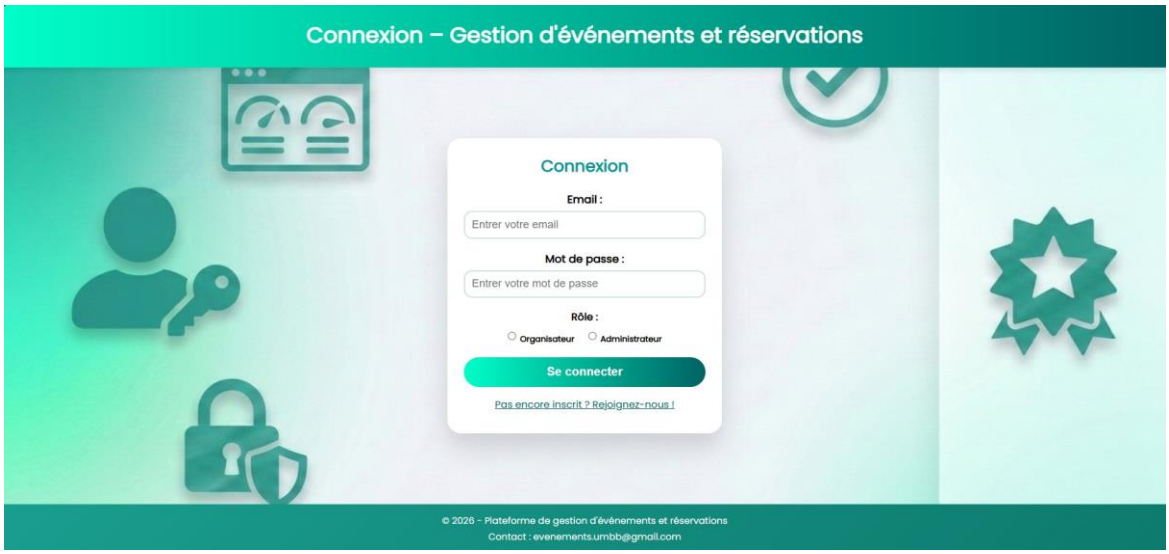


Figure 13 : Interface de connexion professionnelle.

3.5.5 Espace administrateur

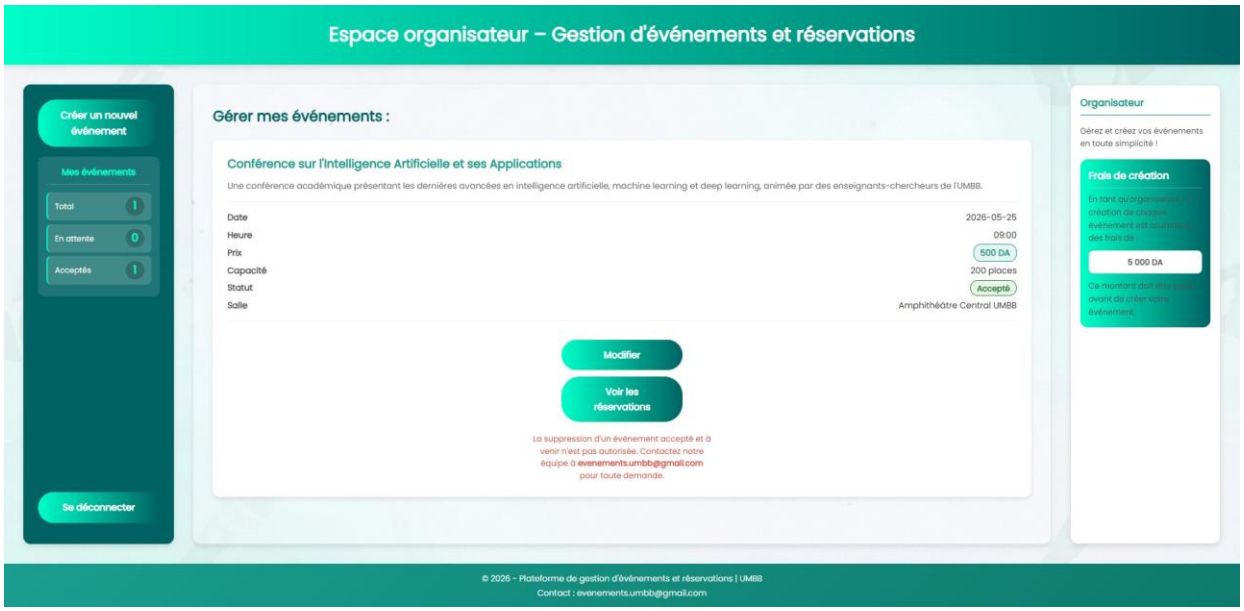
L'espace administrateur permet à l'administrateur de gérer toute la plateforme à travers les fonctionnalités suivantes : gestion des utilisateurs, gestion des organisateurs, gestion des événements.



Figure 14 : Interface de l'espace administrateur

3.5.6 Espace organisateur

L'espace organisateur permet à l'organisateur de gérer ses événements : créer un nouvel événement, modifier ou supprimer un événement existant, et consulter la liste des réservations de ses événements.



<u>Figure 15</u> : Interface de l'espace organisateur.

3.6 Conclusion

Dans ce chapitre, nous avons décrit la partie réalisation en présentant les outils de développement utilisés et détaillé la structure de notre base de données à travers un dictionnaire de données complet, ainsi que la présentation des différentes interfaces de l'application.

Conclusion Générale

Notre projet est une application web permettant aux utilisateurs de réserver des événements. Dans le premier chapitre, nous avons abordé la problématique et quelques solutions. Le deuxième chapitre a porté sur l'identification des acteurs (utilisateur, administrateur, organisateur) et sur la création de diagrammes de cas d'utilisation, des classes et de séquences. Le troisième chapitre a été consacré à la création du site web et à la présentation des langages et des outils utilisés.

Ce projet nous a permis de résoudre le problème de la double réservation par un seul utilisateur ainsi que celui du dépassement de capacité. Pour cela, nous avons répondu à ces problématiques en vérifiant si l'utilisateur avait déjà réservé pour le même évènement et en vérifiant que la capacité maximale de l'évènement n'était pas atteinte.

Nous avons suivi une démarche allant de l'étude de l'existant jusqu'à la réalisation concrète. Ce travail a enrichi nos connaissances sur les plans théorique et pratique.

À l'avenir, ce projet pourrait être amélioré en développant une version mobile adaptée aux smartphones, en intégrant un vrai système de paiement en ligne, et en ajoutant un système de sécurité avancé pour protéger les données des utilisateurs.

Bibliographie

[1] L. Khelil et Y. Khelil, “Conception et modélisation d’une application web pour la gestion de la bibliothèque de la faculté des sciences (Université M’Hamed Bougara),” Mémoire de Licence, Département d’Informatique, Université M’hamed Bougara, Boumerdès, 2024.

[2] R. Nezlioui, R. Tebbakh, et R. Y. Zamoum, “Conception et réalisation d’un système de la Gestion de Facturation et Recouvrements,” Mémoire de Licence, Département d’Informatique, Université M’hamed Bougara, Boumerdès, 2022.

[3] Z. Belabed, M. Mahfoudi, et I. A. Mahiddine, “Développement d’une Application Web d’Annotation pour l’Analyse et l’Interprétation des Images Radiologiques, Mammographiques, IRM et Echographiques,” Mémoire de Licence, Département d’Informatique, Université M’hamed Bougara, Boumerdès.

[4] J.-M. Doudoux, “Les JSP (Java Server Pages),” [En ligne]. Disponible sur : https://www.jmdoudoux.fr. [Consulté le : 14 mai 2025].

[5] SpringSource, “Spring Boot,” [En ligne]. Disponible sur : https://spring.io. [Consulté le : 14 mai 2025].

[6] W3Schools, “HTML and CSS,” [En ligne]. Disponible sur : https://www.w3schools.com. [Consulté le : 14 mai 2025].

[7] JetBrains, “IntelliJ IDEA,” [En ligne]. Disponible sur : https://www.jetbrains.com/idea. [Consulté le : 14 mai 2025].

[8] W3Schools, “SQL Tutorial,” [En ligne]. Disponible sur : https://www.w3schools.com/sql. [Consulté le : 14 mai 2025].

[9] Microsoft, “Visual Studio Code,” [En ligne]. Disponible sur : https://code.visualstudio.com. [Consulté le : 14 mai 2025].

[10] draw.io, “Diagramming Tool,” [En ligne]. Disponible sur : https://www.drawio.com. [Consulté le : 14 mai 2025].