

# Rapport d'analyse informatique

M2 TSI – Projet Personnel

Hannick Abdul Kuthoos

# Table des matières

1. Introduction.....	3
1.1 Contexte .....	3
1.3 Recueil des besoins utilisateur .....	3
2. Analyse fonctionnelle .....	4
3. Etude technique .....	6
3.1 Solution : site web classique .....	7
3.2 Solution : application C++.....	8
Conclusion .....	10
Table des illustrations.....	11

# 1. Introduction

## 1.1 Contexte

Les salles de spectacles jouent un rôle vital dans la promotion de la culture, des arts et du divertissement. Pour les gestionnaires de telles salles, il est essentiel de mettre en place un système efficace de vente de billets qui non seulement simplifie le processus d'achat pour les spectateurs, mais qui permet également une gestion fluide des événements et des réservations. Ce rapport se penche sur l'analyse et le développement d'un tel système, visant à améliorer l'expérience des utilisateurs tout en optimisant les opérations internes de la salle.

Le rapport est organisé en plusieurs sections, chacune traitant d'aspects spécifiques du projet. Nous débuterons par les exigences utilisateurs, les fonctionnalités proposées, puis nous explorerons l'architecture du système, la conception de la base de données, l'interface utilisateur, et enfin, les choix de langages de développement et d'outils.

La conception d'un système de vente de billets pour une salle de spectacles présente un ensemble unique de contraintes à prendre en compte. Tout d'abord, la simplicité et la convivialité de l'interface utilisateur sont essentielles pour garantir que les spectateurs puissent naviguer facilement, choisir leurs places et finaliser leur achat sans complications. Une autre contrainte importante est la nécessité de gérer simultanément plusieurs événements et dates pour un large éventail de spectacles.

Cela exige une architecture flexible capable d'ajuster les disponibilités en temps réel, d'arrêter les ventes lorsque les spectacles sont complets, et de permettre aux gestionnaires de gérer efficacement la programmation. De plus, la gestion des ventes en période de crise, comme la pandémie de Covid-19, implique la capacité de mettre en œuvre des mesures spéciales, telles que la vente de places avec une distanciation sociale appropriée.

## 1.2 Recueil des besoins utilisateur

Les besoins des utilisateurs (clients) du système de vente de billets pour une salle de spectacles sont axés sur la simplicité, la commodité et la personnalisation de l'expérience. Les utilisateurs doivent pouvoir facilement se s'inscrire et se connecter à la plateforme de vente de billets, parcourir les spectacles disponibles, sélectionner des places selon leurs préférences, et effectuer des achats en quelques étapes simples. La possibilité de choisir des places sur un plan de la salle offre une expérience interactive et leur permet de visualiser leur emplacement. De plus, la fonction d'annulation de réservation est essentielle pour permettre aux utilisateurs de gérer leurs plans en cas de changement de circonstances.

L'administrateur du système joue un rôle essentiel dans la gestion efficace des opérations d'une salle de spectacles. Pour répondre à ses besoins, le système doit offrir des fonctionnalités de gestion avancées. L'administrateur a besoin de pouvoir ajouter et mettre à jour des informations sur les

spectacles, y compris les détails sur les dates, les horaires et la disponibilité des billets. De plus, il doit avoir la capacité de gérer les réservations en temps réel, de suivre les annulations, et d'ajuster les quotas de places disponibles pour chaque spectacle. L'accès à des rapports et à des données détaillées sur les ventes et la fréquentation est également essentiel pour aider à la prise de décision. Enfin, l'administrateur doit être en mesure de réagir rapidement à des situations exceptionnelles, telles que des restrictions de capacité en cas de pandémie, en ajustant les paramètres de vente de billets de manière appropriée.

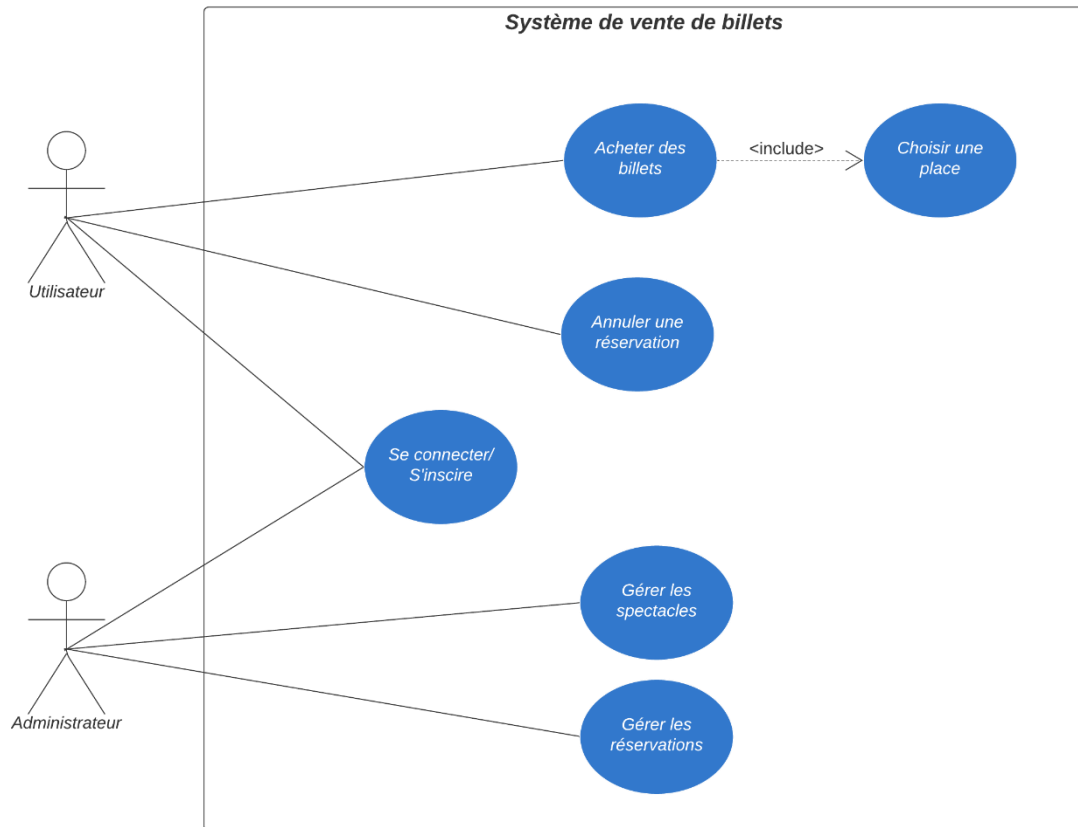


Figure 1 Diagramme de cas d'utilisation du système de vente de billets

## 2. Analyse fonctionnelle

Du côté client l'utilisateur (client) se connecte à la plateforme de vente de billets et arrive sur une page où il est confronté à plusieurs options :

- Il peut dans un premier temps choisir un spectacle pour avoir les différentes informations dessus (date, heure, prix, nom, ...).
- Il peut ensuite acheter des billets pour le spectacle choisit. Il arrive alors sur une fenêtre où il peut choisir ses places sur un plan. L'utilisateur peut ensuite confirmer son choix et il peut acheter un ou plusieurs billets en une seule fois.
- Une fois la confirmation effectuée. Il revient alors à la page précédente avec les différents choix de spectacle. Il peut ensuite aller sur une nouvelle fenêtre où il peut voir ses réservations et il peut les annuler.

Du côté client, cette fois-ci pour l'administrateur il se connecte à la plateforme de la même manière que le client. Mais il arrive directement sur une fenêtre de gestion. Il a alors deux onglets différents :

- Gestion des spectacles : Dans cet onglet l'administrateur peut gérer les différents spectacles en cours. Il peut arrêter la vente de certains spectacles et ajouter une propriété indiquant une limitation sur la vente des places pour certains spectacles en ajoutant l'option Covid-19.
- Gestion des spectateurs : Ici l'administrateur a accès aux informations liés aux réservations effectués par les utilisateurs. Il peut voir les annulations et les achats ainsi que les noms des utilisateurs il peut annuler l'achat ou l'annulation d'un billet.

Du côté client (utilisateur), une fois qu'il a sélectionné son spectacle et acheté ses billets, le processus passe à une série d'activités en coulisses qui permettent de gérer ces réservations. Pour mieux comprendre ces étapes, nous pouvons illustrer le flux de travail à l'aide d'un diagramme d'activité. Ce diagramme détaillera les actions et les décisions prises par le système et les utilisateurs tout au long du cycle de réservation, de la sélection du spectacle à la confirmation de l'achat. Voir le diagramme ci-dessous pour visualiser ces opérations de manière plus claire et précise :

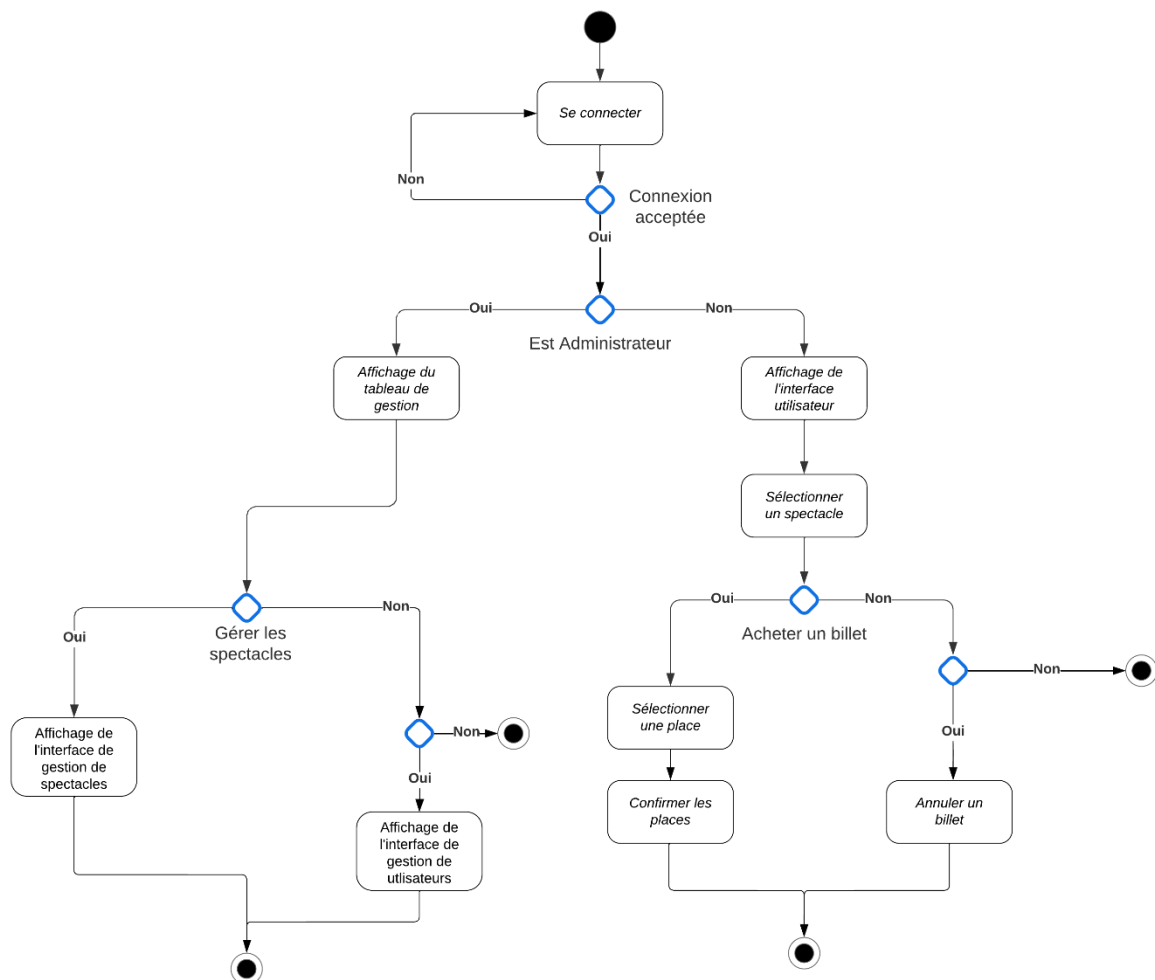


Figure 2 Diagramme d'activité du système de vente de billets

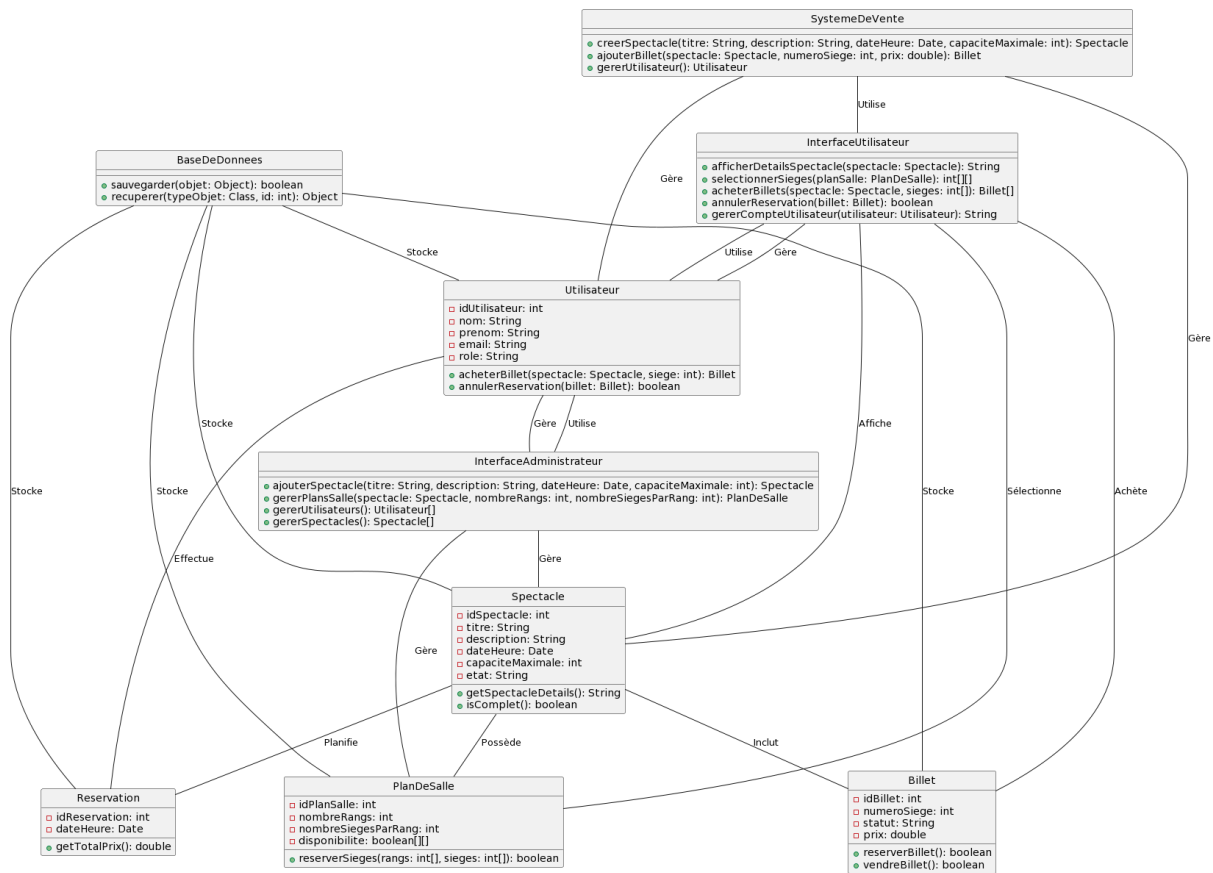


Figure 3 Diagramme de classe du système de vente de billets

L'interaction entre le client et le serveur dans ce système de vente de billets commence par l'utilisateur (représenté par la classe Utilisateur) qui utilise l'interface utilisateur pour consulter les détails d'un spectacle (Spectacle). En utilisant cette interface, l'utilisateur sélectionne des sièges sur un plan de salle (PlanDeSalle) pour acheter des billets (Billet). Le serveur, sous la forme de la classe SystemeDeVente, gère ces demandes en traitant les achats et en maintenant les données relatives aux spectacles, aux billets, et aux réservations. Si un achat est réussi, le serveur met à jour les billets et renvoie une confirmation à l'utilisateur. En cas d'échec, l'utilisateur reçoit un message d'erreur, et le système garantit que la capacité maximale n'est pas dépassée.

Le serveur fait également appel à la classe BaseDeDonnees pour stocker et récupérer les données essentielles, permettant une gestion efficace des informations relatives aux spectacles, aux billets, aux utilisateurs et aux réservations, facilitant ainsi l'ensemble du processus de vente de billets. En somme, l'interaction entre l'utilisateur (client) et le serveur (système de vente) est centrée sur les classes du diagramme, avec chacune jouant un rôle spécifique pour garantir le bon déroulement des opérations de vente de billets pour les spectacles.

### 3. Etude technique

## 3.1 Solution : site web classique

La mise en place d'un site web de vente de billets représente une solution qui semble être la plus classique et pertinente pour répondre aux besoins du projet de vente de billets pour une salle de spectacles.

- **Conception du Site Web :** La conception du site web repose sur des technologies web standard telles que HTML, CSS et JavaScript. Ces langages permettent de créer une interface utilisateur intuitive et réactive. HTML structure le contenu, CSS gère la mise en page et le style, tandis que JavaScript apporte des fonctionnalités interactives, notamment la sélection de sièges sur un plan de salle. Cette combinaison d'outils garantit une expérience utilisateur fluide et conviviale.
- **Utilisation de Frameworks :** L'utilisation de frameworks de développement web, tels que React, Angular ou Vue.js, peut simplifier la création du site web. Ces frameworks offrent des composants prêts à l'emploi, une gestion efficace de l'état de l'application et une réactivité accrue. Le choix du framework dépendra des besoins spécifiques du projet et des préférences techniques.
- **Bases de Données et Gestion des Données :** Pour stocker et gérer les informations essentielles, une base de données relationnelle, telle que PostgreSQL, peut être mise en place. PostgreSQL est open source, offre une grande flexibilité, une sécurité avancée, et est compatible avec de nombreux langages de programmation. Elle permet de stocker des données sur les spectacles, les billets disponibles, les utilisateurs et les réservations de manière efficace.
- **Sécurité et Chiffrement :** La sécurité est primordiale pour la protection des données des utilisateurs. L'application du protocole HTTPS via des certificats SSL assure le chiffrement des données transmises entre le client et le serveur. Des mesures de sécurité telles que la validation des entrées, la prévention contre les attaques par injection SQL, et l'authentification à deux facteurs renforcent la sécurité du site web.
- **Hébergement :** Le choix d'une infrastructure d'hébergement est essentiel pour garantir la disponibilité du site. Des services cloud tels qu'Amazon Web Services (AWS) ou Microsoft Azure offrent des options d'hébergement évolutives. La configuration adéquate du serveur web, tel que Nginx ou Apache, assure la distribution efficace du contenu.
- **Maintenance et Support :** Une maintenance régulière du site web est indispensable pour corriger les bugs, améliorer les performances et garantir la continuité du service. Le support technique est également nécessaire pour répondre aux questions des utilisateurs et résoudre d'éventuels problèmes.

<b>Forces (+) :</b> <ul style="list-style-type: none"> <li>• Facilité d'utilisation : Le site web offre une interface utilisateur conviviale pour les clients, simplifiant la réservation et l'achat de billets.</li> <li>• Gestion efficace des données : La base de données centralisée permet une gestion optimale des spectacles, des billets, des utilisateurs et des réservations.</li> <li>• Flexibilité technique : Le choix d'outils, tels que HTML, CSS, JavaScript, et des frameworks, offre une flexibilité pour le développement.</li> </ul>	<b>Faiblesses (-) :</b> <ul style="list-style-type: none"> <li>• Besoin de maintenance continue : Le site web nécessite une maintenance régulière pour corriger les bugs et assurer son bon fonctionnement.</li> <li>• Coût initial : La création et la maintenance du site web nécessitent des investissements initiaux.</li> </ul>
<b>Opportunités (+) :</b> <ul style="list-style-type: none"> <li>• Accès aux clients en ligne : L'application offre la possibilité d'atteindre un large public en ligne, ce qui peut augmenter les ventes de billets.</li> <li>• Données clients : Les données collectées via le site web permettent de mieux comprendre les préférences des clients et de personnaliser les offres.</li> </ul>	<b>Menaces (-) :</b> <ul style="list-style-type: none"> <li>• Concurrence en ligne : La concurrence dans le domaine de la vente de billets en ligne est prédominante.</li> <li>• Problèmes de sécurité : Les menaces de sécurité en ligne, telles que les cyberattaques, nécessitent une surveillance constante.</li> </ul>

Figure 4 Matrice SWOT de la solution web

La solution web de vente de billets tire parti de ces outils et technologies pour créer une plateforme en ligne robuste, sécurisée et conviviale. Elle permet aux utilisateurs de consulter, réserver et acheter des billets de manière efficace, tout en offrant une expérience utilisateur optimale.

## 3.2 Solution : application C++

La mise en place d'une application en C++ avec l'utilisation de Qt Creator et SQLite représente une solution technique robuste pour répondre aux besoins du projet de vente de billets pour une salle de spectacles.



- **Qt Creator** : Qt Creator est un environnement de développement intégré (IDE) spécialement conçu pour la création d'applications en C++. Il offre une large gamme de fonctionnalités, notamment un éditeur de code, un débogueur, un concepteur d'interfaces graphiques, et une intégration facile avec la bibliothèque Qt. Cette solution est pertinente pour la création d'applications de bureau performantes.
- **Bibliothèque Qt** : Qt est une bibliothèque logicielle puissante qui simplifie le développement d'applications C++. Elle offre des composants graphiques, des fonctions de gestion de données, et des API pour la création d'interfaces utilisateur interactives. En utilisant Qt, vous pouvez concevoir une interface utilisateur conviviale pour la vente de billets et gérer efficacement les opérations de réservation.
- **Base de Données SQLite** : SQLite est une base de données légère, rapide, et facile à intégrer dans des applications de bureau C++. Elle est pertinente pour stocker les données sur les spectacles, les billets disponibles, les utilisateurs et les réservations. L'intégration de SQLite dans l'application permet une gestion efficace des données, sans nécessiter l'installation de serveurs de bases de données volumineux.
- **Modèle-Vue-Contrôleur (MVC)** : L'architecture MVC est pertinente pour structurer l'application. Elle sépare la logique métier (Contrôleur), la gestion des données (Modèle), et l'interface utilisateur (Vue). Cette approche facilite la maintenance, l'extensibilité et la compréhension du code source de l'application.
- **Sécurité des Données** : L'application devra implémenter des mesures de sécurité pour protéger les données sensibles des utilisateurs. Cela inclut le chiffrement des données stockées dans la base SQLite, la validation des entrées pour prévenir les attaques, et la gestion sécurisée de l'authentification des utilisateurs.
- **Multiplateforme** : L'utilisation de Qt et C++ permet de créer des applications multiplateformes. Vous pouvez cibler plusieurs systèmes d'exploitation, notamment Windows, macOS et Linux, en utilisant un seul code source.
- **Maintenance et Support** : Une maintenance régulière de l'application est essentielle pour garantir sa stabilité et corriger les éventuels bugs. Le support technique est également nécessaire pour répondre aux questions des utilisateurs et résoudre d'éventuels problèmes.

<b>Forces (+) :</b>	<b>Faiblesses (-) :</b>
<ul style="list-style-type: none"> <li>• Haute performance : L'application de bureau C++ offre une performance élevée, ce qui garantit une expérience utilisateur fluide.</li> <li>• Contrôle des données : L'utilisation de SQLite permet un contrôle total sur les données stockées localement.</li> <li>• Multiplateforme : L'application peut être déployée sur plusieurs systèmes d'exploitation, augmentant ainsi sa portée.</li> </ul>	<ul style="list-style-type: none"> <li>• Déploiement initial : La création de l'application de bureau nécessite un développement initial plus complexe.</li> </ul>

<p><b>Opportunités (+) :</b></p> <ul style="list-style-type: none"> <li>• Marchés cibles spécifiques : L'application de bureau peut cibler des marchés spécifiques qui préfèrent les applications de bureau aux solutions web.</li> <li>• Personnalisation avancée : L'application offre des options de personnalisation avancée pour les spectacles et les réservations.</li> </ul>	<p><b>Menaces (-) :</b></p> <ul style="list-style-type: none"> <li>• Évolution des préférences des utilisateurs : Les préférences des utilisateurs évoluent, et certains peuvent préférer des solutions web plus faciles d'accès.</li> <li>• Convergence technologique : La convergence des technologies peut rendre les applications web plus performantes et accessibles.</li> </ul>
--	--

*Figure 5 Matrice SWOT de la solution C++*

La solution technique basée sur Qt Creator, SQLite et C++ offre une plateforme de bureau polyvalente et sécurisée pour la vente de billets de spectacles. Elle permet de créer une application performante avec une interface utilisateur conviviale, tout en garantissant la sécurité des données des utilisateurs et une expérience utilisateur optimale.

## Conclusion

Dans l'analyse des solutions proposées pour la vente de billets pour une salle de spectacles, il est évident que la solution web représente l'approche classique et largement utilisée pour répondre à ce besoin. Les sites web offrent une accessibilité en ligne, une convivialité pour les utilisateurs et une gestion efficace des données, ce qui en fait un choix naturel pour de nombreux projets de vente de billets.

Cependant, dans ce cas particulier, la solution en C++ avec Qt Creator et SQLite a été retenue pour des raisons pédagogiques. Cette décision découle de la volonté d'acquérir des compétences spécifiques en développement d'applications de bureau, de maîtriser les technologies C++ et Qt, et de relever le défi technique que présente le développement d'une application de bureau. La solution en C++ offre des avantages en termes de performances et de contrôle des données, mais elle est moins courante dans le domaine de la vente de billets en ligne.

# Table des illustrations

Figure 1 Diagramme de cas d'utilisation du système de vente de billets.....	4
Figure 2 Diagramme d'activité du système de vente de billets.....	5
Figure 3 Diagramme de classe du système de vente de billets.....	6
Figure 4 Matrice SWOT de la solution web.....	8
Figure 5 Matrice SWOT de la solution C++ .....	10