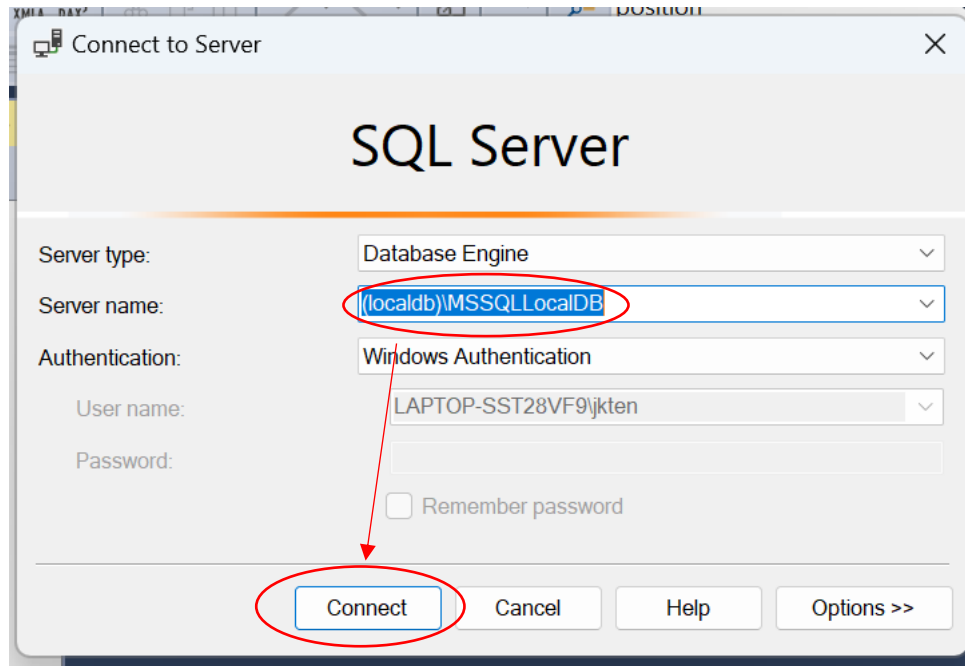


## **Mise en place de l'application**

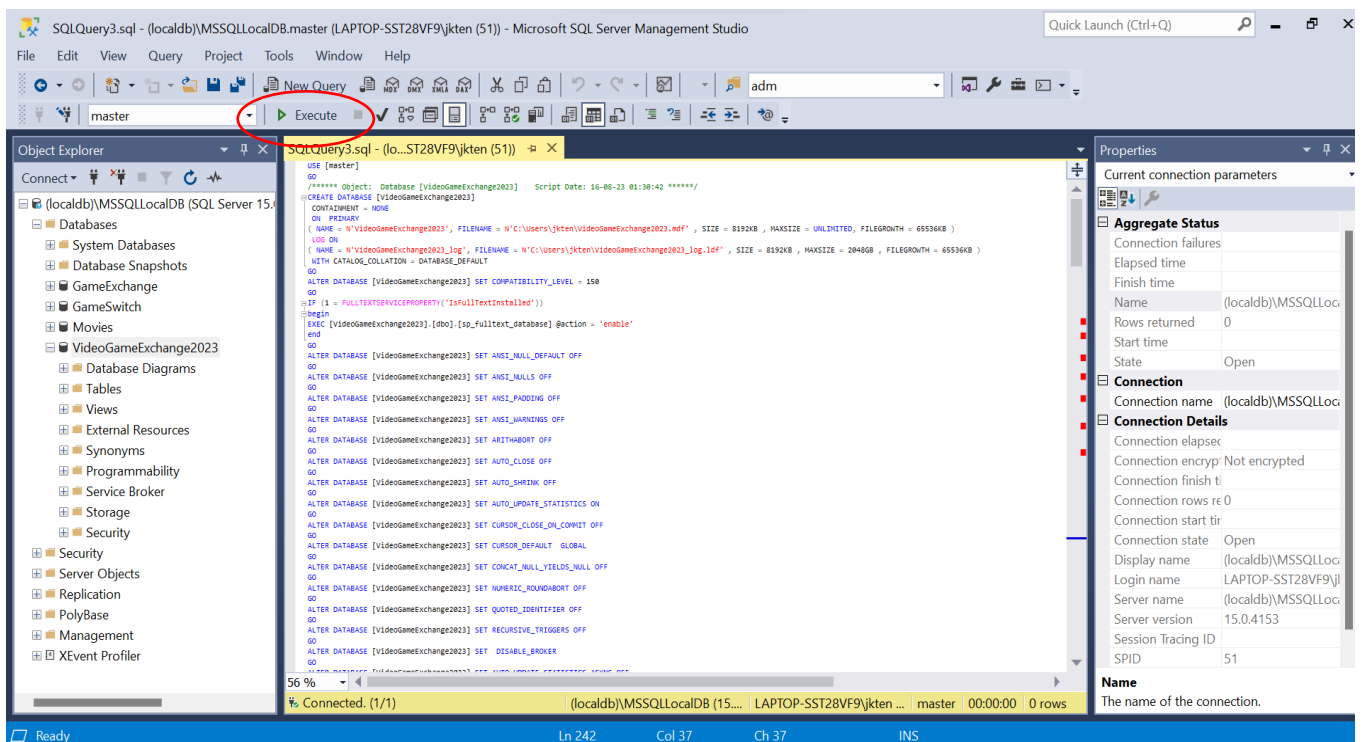
1. Ouvrir MSSQL,
2. Se connecter en local,  
Pour se connecter en local, il faut entrer une chaîne de connexion qui est généralement celle-ci :  
(localdb)\MSSQLLocalDB



Cliquer sur Connect, on est normalement connecté en local.

3. Vérifier qu'il n'y a aucune base de données nommée VideoGameExchange2023, s'il y en a une, la supprimer.
4. Exécuter le script de création de la Base de données, des tables et d'insertion des données :  
Le script se trouve à la racine lors du dézippage du projet : SQLQuery3

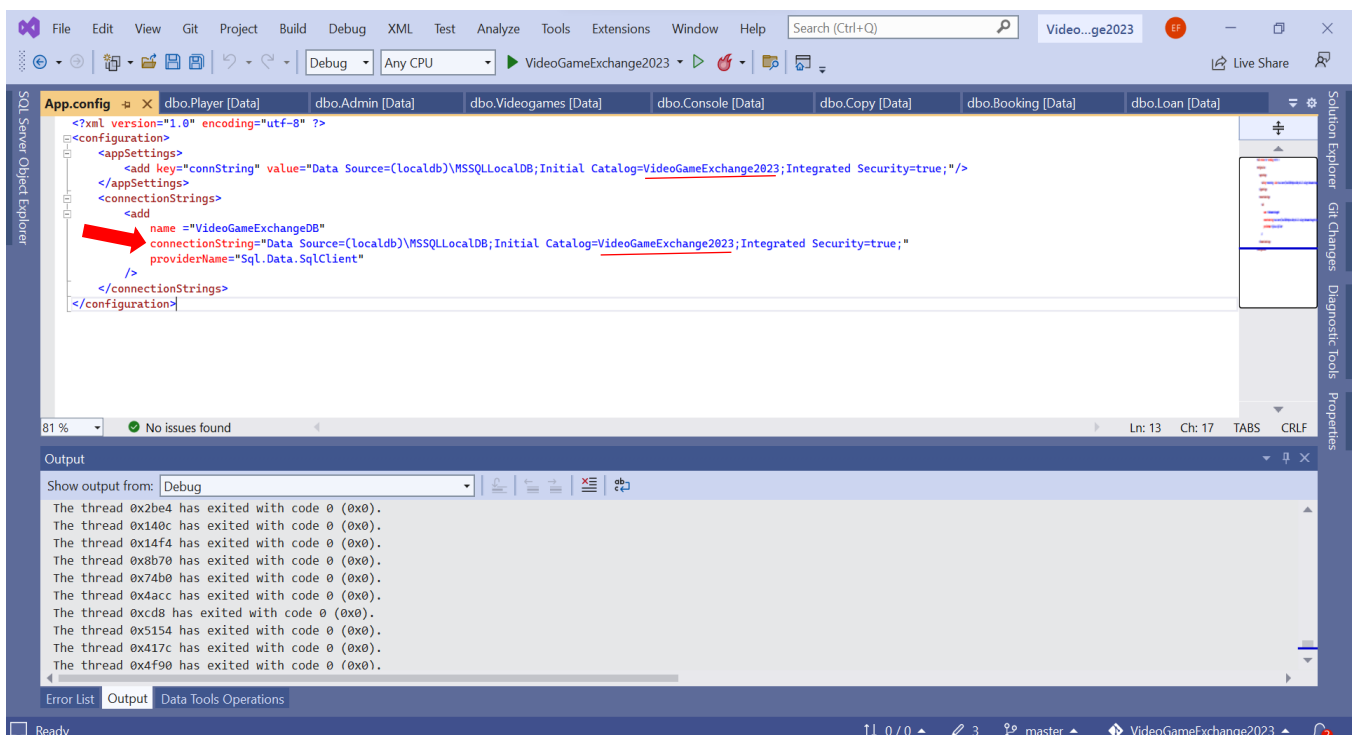
Aller dans fichier dans le menu, ouvrir, fichier, choisir le fichier là où il est sur l'ordinateur. Le fichier va s'ouvrir, cliquer ensuite sur exécuter pour exécuter le script.



La base de données devrait normalement être créée et les données insérées.

Si la chaîne de connexion à la base de données est différente, il faut la changer.

La chaîne de connexion se trouve dans le fichier App.config, il faut juste changer la chaîne de caractères qui se trouve dans connectionString.



Une fois la mise en place de la base de données effectuée, on peut lancer l'application.

Pour lancer l'application il faut soit ouvrir la solution donnée dans le zip, soit la cloner depuis github.

Lorsque vous lancez l'application, vous êtes redirigé vers une page de connexion. Cette page est accessible tant aux administrateurs qu'aux joueurs.

Plusieurs administrateurs et joueurs ont déjà été enregistrés dans le but de tester différentes fonctionnalités :

| dbo.Admin [Data] App.config |          |  |
|-----------------------------|----------|--|
| Max Rows: 1000              |          |  |
| username                    | password |  |
| adm                         | 1234     |  |
| admin                       | admin    |  |

| dbo.Player [Data] App.c |          |  |
|-------------------------|----------|--|
| Max Rows: 1000          |          |  |
| username                | password |  |
| Anton                   | Anton    |  |
| Buba                    | Buba     |  |
| Enny                    | Frans    |  |
| Lola                    | Lola     |  |
| Marie                   | Marie    |  |
| Mimi                    | Mimi     |  |

MainWindow

CLOSE

User Name

Password

Login

SIGN UP / new player

Trois options sont possibles :

- Cliquer sur signup pour aller à la page d'inscription et s'inscrire
- Entrer son username et son mot de passe afin de se connecter en tant qu'administrateur
- Entrer son username et son mot de passe afin de se connecter en tant que joueur.

Pour l'inscription : **Le pseudonyme doit être unique, et la combinaison du nom d'utilisateur et du mot de passe doit également être unique.**

SignupWindow

Sign UP

Please complete the form below :

Name (user name) :

Pseudo :

Password :

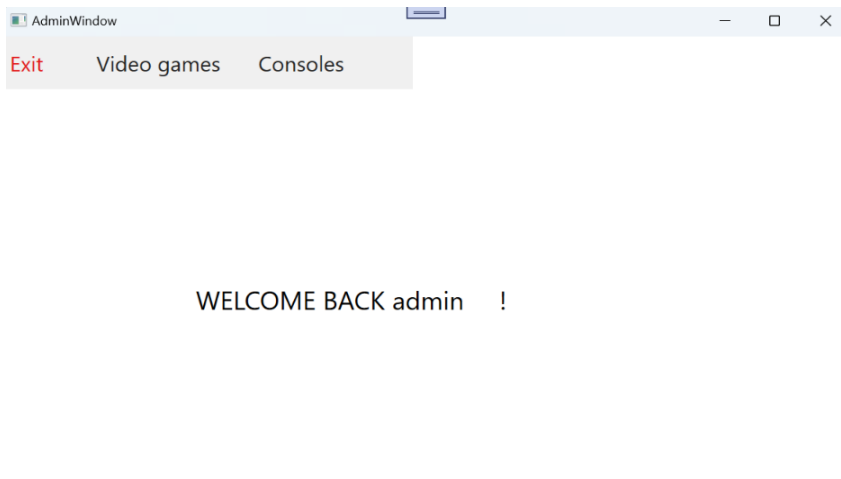
Date of Birth :

Select a date 15

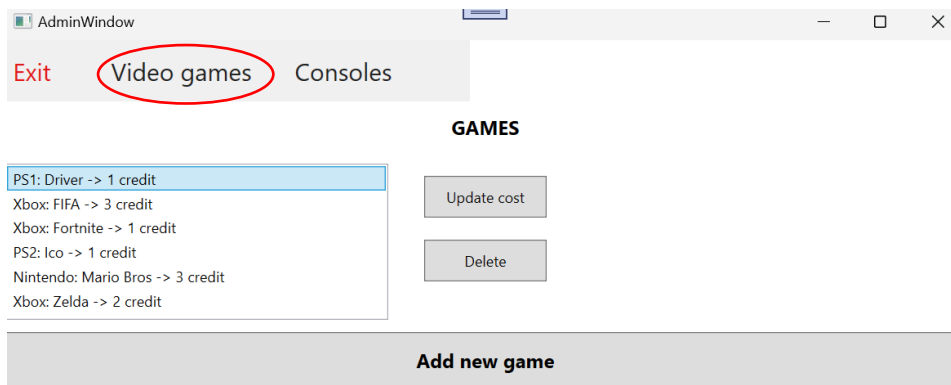
SUBMIT

CANCEL

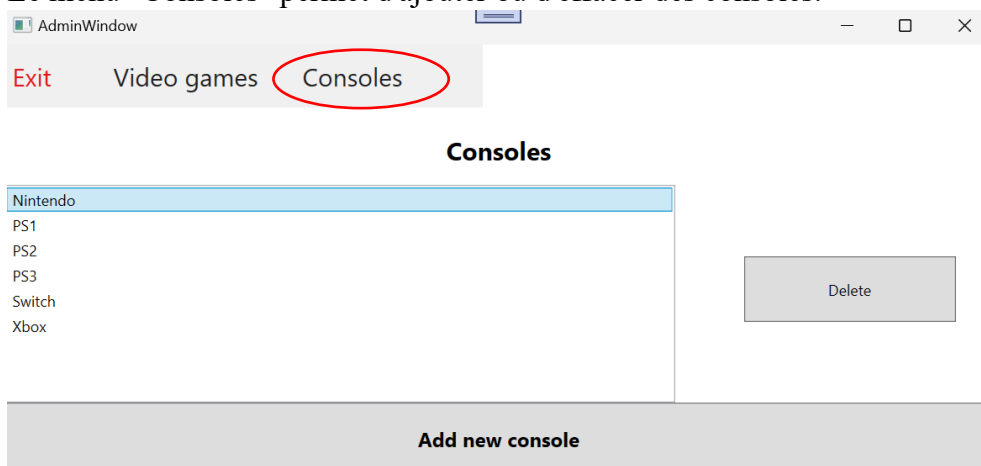
Page initiale de l'administrateur après s'être connecté (logged in) :



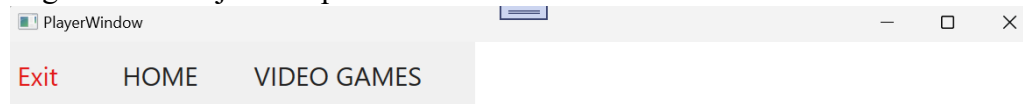
Le menu "Video games" permet d'ajouter, de modifier le crédit et d'effacer un jeu vidéo.



Le menu "Consoles" permet d'ajouter ou d'effacer des consoles.

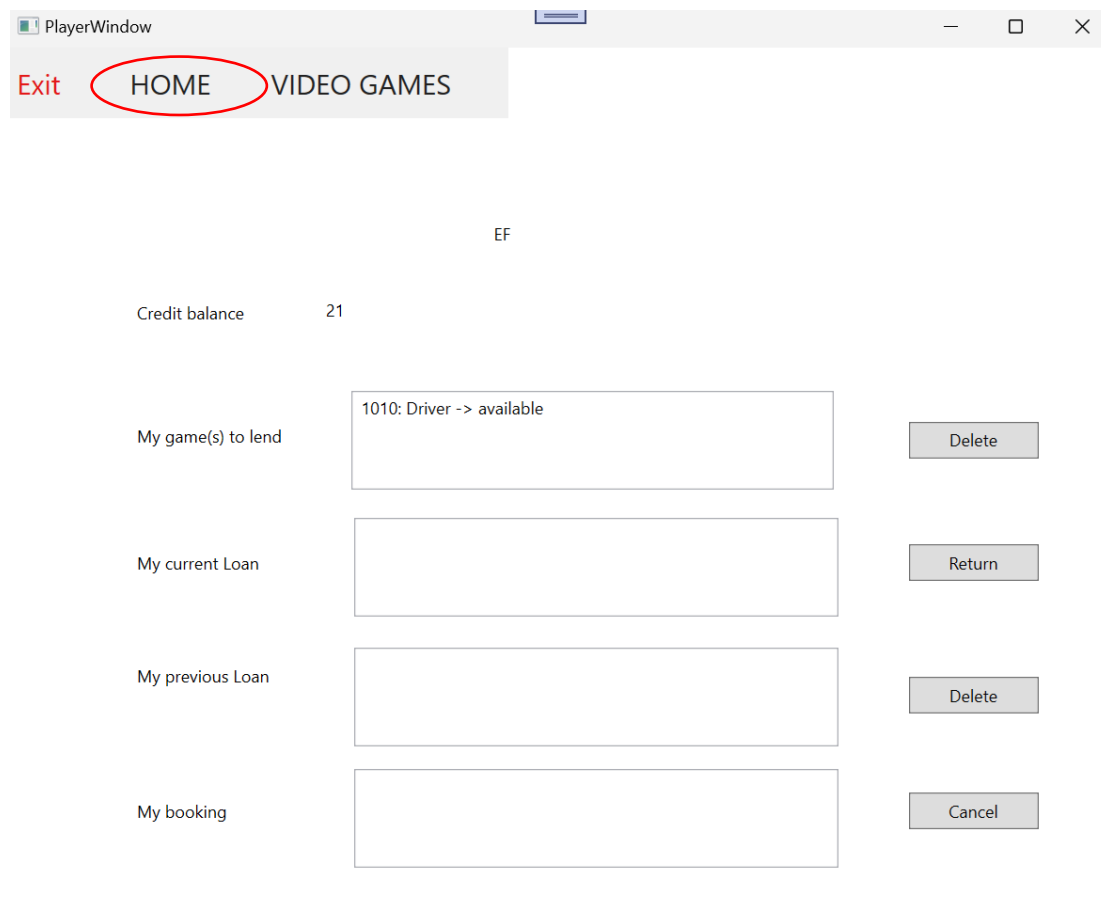


Page initiale du joueur après s'être connecté :

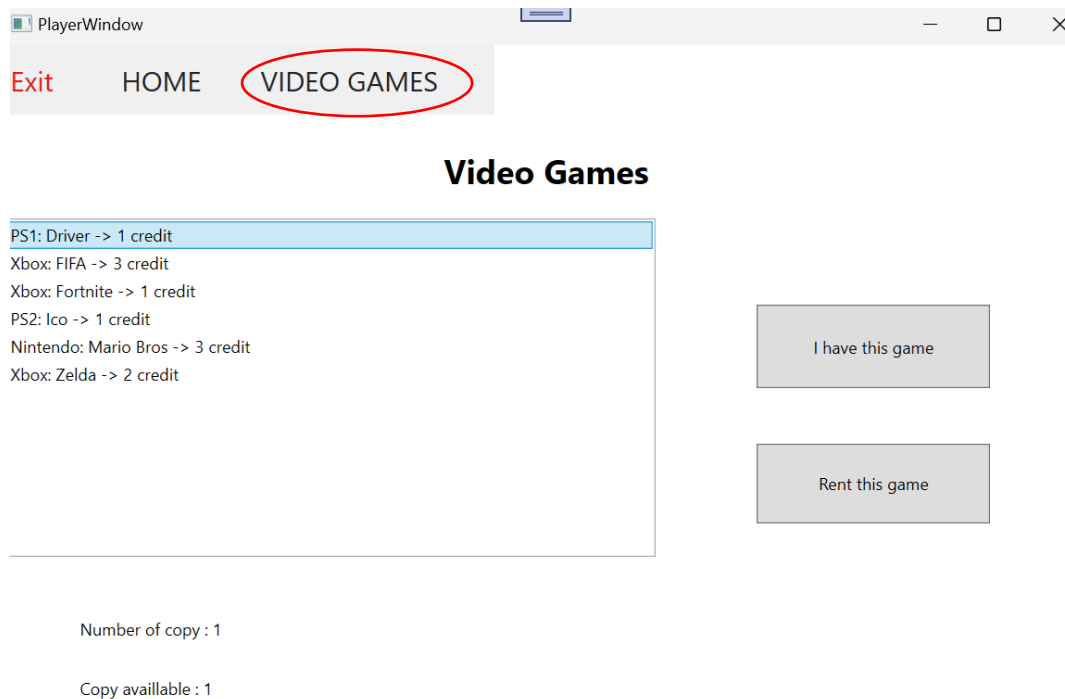


Welcome, Enny (EF)! You have 21 credit(s).

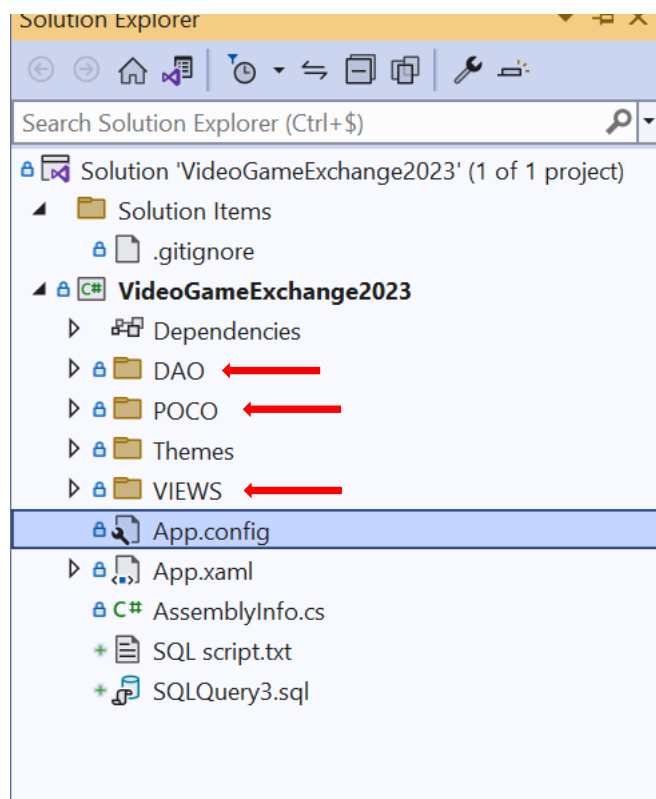
Menu Home, permetre de consulter plusieurs information de jouer connectée.



Menu "VIDEO GAMES" : permet de consulter la liste des jeux vidéo, de mettre à disposition, de louer et de réserver (booking):

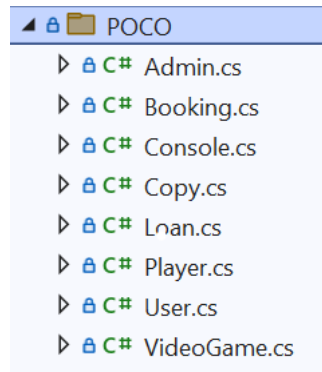


## Architecture

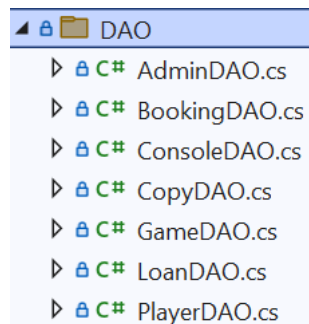


Ce projet comprend trois dossiers importants : POCO, DAO et VIEWS.

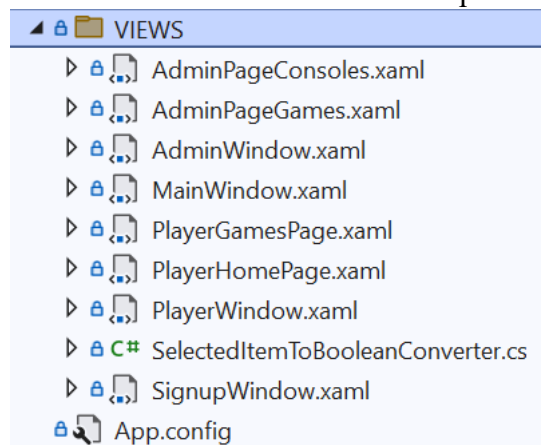
Les classes dans le dossier POCO contiennent la logique métier (Business Logic):



Le dossier DAO sert à lier les classes de logique avec la base de données:



Et le dossier VIEWS est dédié à la partie interface utilisateur (UI).



## **Implementation de C#**

Dans le développement de ce projet, j'ai mis en œuvre plusieurs notions de C# :

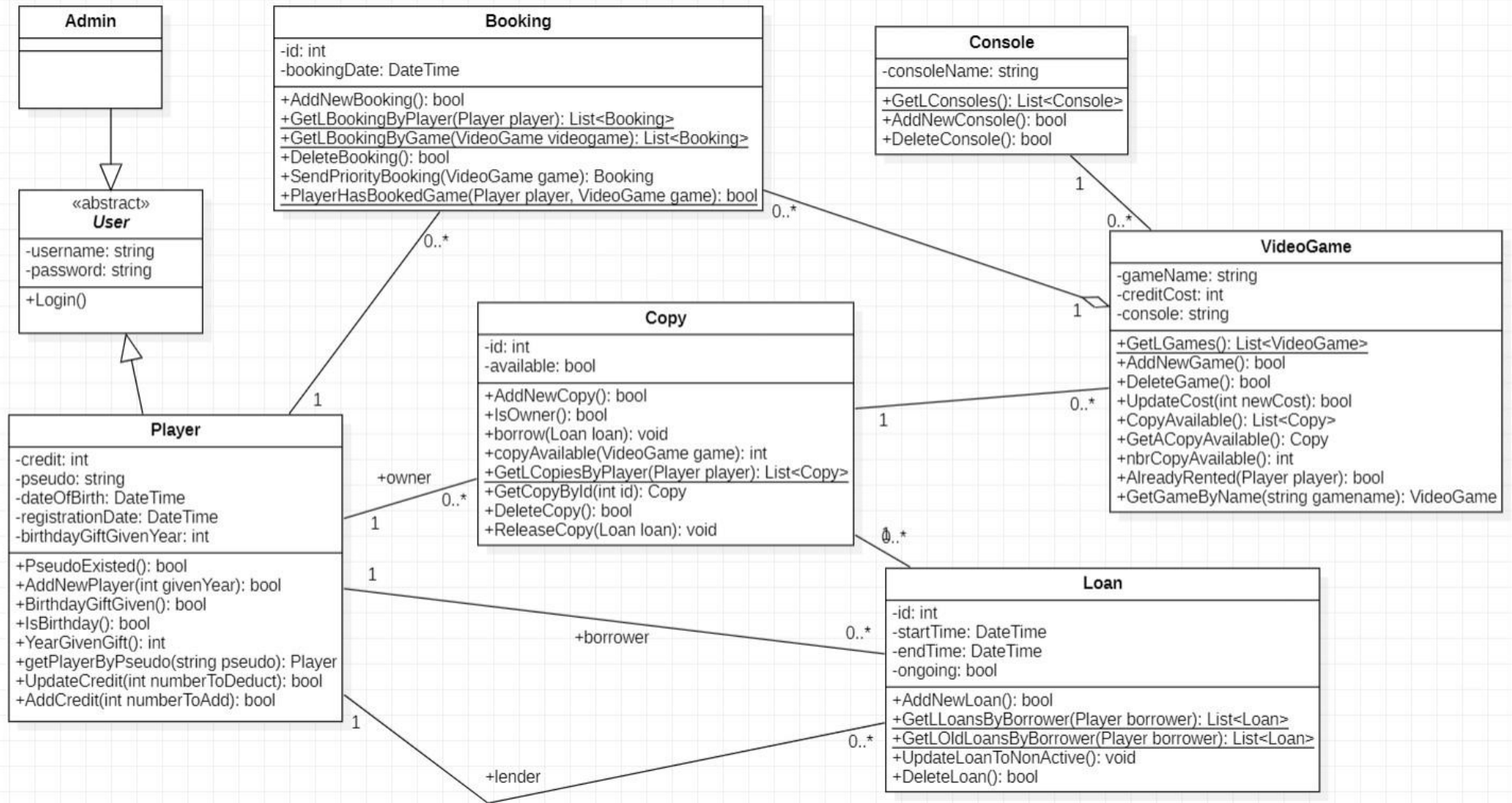
- Encapsulation : Elle est appliquée dans toutes les classes logiques (POCO).
- L'utilisation de la classe abstraite : J'ai utilisé une classe abstraite pour la classe User en tant que parent, avec les classes Player et Admin qui en héritent.
- Constructor Overloading : J'ai utilisé cette technique en fonction des besoins pour créer de nouveaux objets avec différentes configurations.
- Static : J'ai utilisé le mot-clé "static" pour accéder à certaines propriétés d'une classe sans avoir besoin de créer une instance de cette classe.
- Lambda Expression : J'ai utilisé des expressions lambda pour rendre le code un peu plus élégant.
- Opération Conditionnelle (ternaire) : J'ai utilisé cette opération pour simplifier et raccourcir le code.

Quelques particularités dans ce projet :

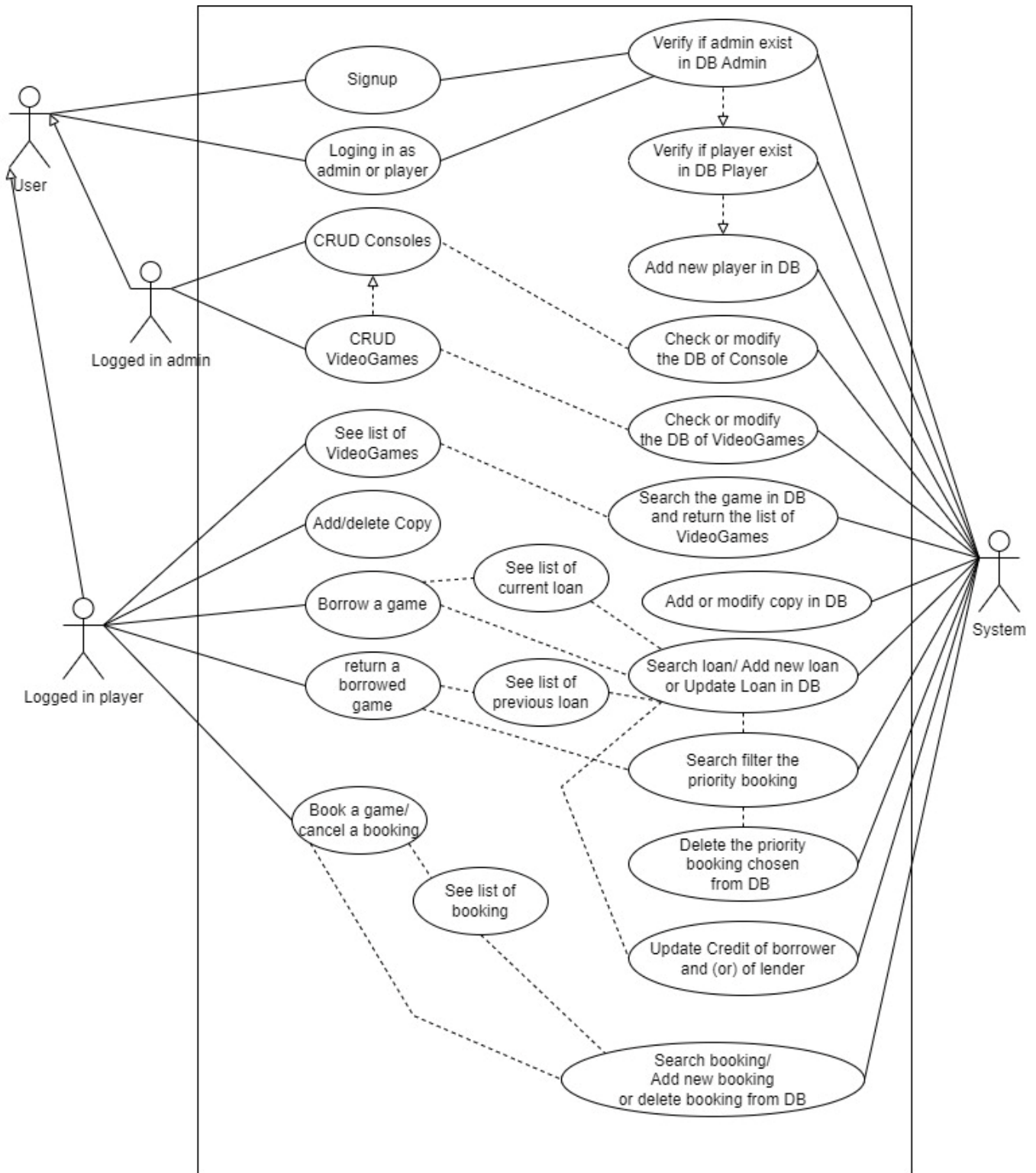
- Inscription – Birthday gift : Les joueurs s'inscrivant le jour de leur anniversaire ou avant bénéficieront d'un crédit bonus s'ils se connectent le jour de leur anniversaire ou après.
- Copie de jeu rendue par un joueur : Lorsqu'un joueur rend une copie de jeu, le système vérifie s'il y a des réservations pour ce jeu. Si des réservations existent, le système trie une priorité pour permettre à un joueur en tête de liste de louer automatiquement cette copie ,



## Diagramme des classes



## Diagramme des cas d'utilisation



## Description du cas d'utilisation :

### I. Scénarios nomininaux

#### 1. Utilisateur : Personne non enregistrée (user / Future Player)

Précondition : la méthode associée ainsi que la table **Player** dans la base de données sont opérationnelles.

- 1.1. L'utilisateur cliquer sur bouton Sign up.
- 1.2. Le formulaire d'inscription s'affiche.
- 1.3. L'utilisateur remplit le formulaire et clique sur le bouton **Submit**.
- 1.4. Une message de succès s'affiche pour indiquer que l'inscription a été réalisée avec succès, et un bouton **Go to Signin Page** s'affiche.
- 1.5. Le nouvel player appuie sur bouton **Go to Signin Page**.
- 1.6. Le Login page s'ouvre, permettant au nouvel player de se connecter.

#### 2. Utilisateur : Administrateur (admin)

Précondition :

- Toutes les méthodes associées à la connexion (login) en tant qu'administrateur sont implémentées, ainsi que la table Admin dans la base de données. Au moins un administrateur doit déjà être enregistré dans cette table.
  - Toutes les méthodes liées à CRUD pour de Video-Games et les Consoles sont mises en œuvre, ainsi que les tables Console et VideoGame dans la base de données sont prêt.
- 2.1. Sur la page Login, l'administrateur saisit son nom d'utilisateur et son mot de passe, qui sont stockés dans la table Admin de la base de données.
  - 2.2. L'administrateur clique sur le bouton **Login**.
  - 2.3. Une page d'administration s'ouvre, affichant un message de bienvenue et trois menus : « Exit », « Video games », et « Consoles ».
  - 2.4. L'administrateur clique sur le menu **Consoles**.
    - 2.4.1. Une page contenant une liste de Consoles s'affiche si il en existe déjà, accompagnée d'un bouton **Add new console**.
    - 2.4.2. L'administrateur clique sur l'une de consoles de la liste, ce qui fait apparaître un bouton **Delete**.
    - 2.4.3. L'administrateur clique sur le bouton **Delete**.
    - 2.4.4. Un message-box s'affiche pour demander confirmation de la suppression de la console sélectionnée, offrant les options « yes » ou « no ».
    - 2.4.5. L'administrateur clique sur le bouton « yes ».
    - 2.4.6. La console choisi est supprimée de la base de donnée et disparaît de la liste de consoles.
    - 2.4.7. Enclenchant le processus, l'administrateur clique sur le bouton **Add new console**.
    - 2.4.8. Cela fait apparaître un petit formulaire nécessitant d'entrer le nom de la console, accompagné des options **Add** et **Cancel**.
    - 2.4.9. L'administrateur cliquer sur bouton **Cancel**.
    - 2.4.10. La page est alors rétablie a son affichage initial (étape 2.4.1).
    - 2.4.11. Les étapes 2.4.7 et 2.4.8 sont réitérées.
    - 2.4.12. L'administrateur entre le nom de la nouvelle console et cette fois-ci valide en cliquant sur le bouton **Add**.
    - 2.4.13. Cela entraîne un retour à la page initiale 2.4.1 où la liste des consoles est actualisée pour inclure la nouvelle console qui vient d'être ajoutée. En même temps, un message-box apparaît, indiquant le succès de l'ajout, avec un bouton **OK**.

2.5. L'administrateur sélectionne le menu **Video games**.

2.5.1. Une page affichant une liste de Video games apparaît si il en existe déjà, accompagnée d'un bouton **Add new game**.

2.5.2. En cliquant sur l'un de video game de la liste, l'administrateur fait apparaître les boutons **Update cost** et **Delete**.

2.5.3. L'administrateur clique sur le bouton **Update cost**.

2.5.4. Cela déclenche l'affichage d'un petit formulaire demandant d'entrer le nouveau crédit, accompagné des options **Update** et **Cancel**.

2.5.5. L'administrateur choisit de cliquer sur bouton **Cancel**.

2.5.6. Cette action le ramène à la configuration initial de la page (étape 2.5.1.)

2.5.7. Les étapes 2.5.2, 2.5.3, et 2.5.4 sont répétées.

2.5.8. L'administrateur entre la nouvelle valeur du crédit et confirme en cliquant sur le bouton **Update**.

2.5.9. Un Message-box apparaît, indiquant le succès de la modification du crédit, avec un bouton **OK**.

2.5.10. L'administrateur clique sur le bouton OK de Message-box.

2.5.11. La page est alors restaurée a son affichage initial (comme dans l'étape 2.5.1) avec la nouvelle valeur de crédit appliquée au Video game qui avait été sélectionné.

2.5.12. L'administrateur clique sur le bouton **Add new game**.

2.5.13. Cela déclenche l'affichage d'un petit formulaire, accompagné des options **Add** et **Cancel**.

2.5.14. L'administrateur cliquer sur bouton **Cancel**.

2.5.15. La page est alors rétablie a son affichage initial (étape 2.5.1).

2.5.16. Les étapes 2.5.12 et 2.5.13 sont réitérées.

2.5.17. L'administrateur remplit le formulaire avec toutes les informations de nouveau Video game.

2.5.18. L'administrateur clique sur le bouton **Add**.

2.5.19. La page est retournée à son affichage initial (comme à l'étape 2.5.1) avec le nouveau jeu vidéo ajouté à la liste de jeux vidéo. En même temps, une fenêtre contextuelle (Message-Box) apparaît pour indiquer le succès de l'ajout du nouveau jeu vidéo, avec un bouton « OK » permettant de fermer la fenêtre contextuelle.

2.6. L'administrateur clique sur Exit.

2.7. La page d'administrateur c'est fermer

2.8. La page de Login s'ouvre

3. Utilisateur : Player

4. Précondition :

- Toutes les méthodes liées à la connexion (login) en tant que Player sont implémentées, ainsi que la table Player dans la base de données.
- L'utilisateur est un Player enregistrée et reconnu par le système.
- Les Objets de video games ont été préalablement ajoutés par un administrateur.
- Toutes les méthodes liées à la CRUD pour les Copies, les prêts (Loans), et les réservations (Booking) sont mises en œuvre. De plus, les tables associées dans la base de données sont prêtes.

4.1. Sur la page Login, le jouer (Player) saisit son nom d'utilisateur et son mot de passe, qui sont stockés dans la table Player de la base de données.

4.2. Ensuite, le joueur clique sur le bouton **Login**.

4.3. Une fois connecté avec succès, une page dédiée au joueur (Player) s'ouvre. Cette page affiche un message de bienvenue et trois menus : « Exit », « Home », et « Video Games ».

4.4. Le joueur sélectionne le menu **Video games**.

- 4.4.1. Une liste de différents jeux vidéo s'affiche.
- 4.4.2. Lorsque l'utilisateur clique sur l'un des jeux vidéo de la liste, la page affiche les informations suivantes pour le jeu sélectionné:

- Le nombre total de copies disponibles pour ce jeu.
- Le nombre de copies actuellement disponibles pour la location.

En fonction de la disponibilité des copies, les boutons sont affichés comme suit :

- Si personne ne possède actuellement de copie de ce jeu vidéo, seul le bouton «I have this game» apparaît.
  - Si des copies du jeu sont disponibles à la location, les boutons «I have this game» et «Rent this game» apparaissent.
  - Si aucune copie n'est disponible à la location, les boutons «I have this game» et «Book this game» apparaissent à la place du bouton «Rent this game».
- 4.4.3. Le joueur clique sur le bouton «I have this game»
- 4.4.4. Une fenêtre contextuelle (Message-box) apparaît, confirmant le succès de l'ajout de la copie, avec un bouton « Ok ». Un nouvel objet de copie est alors enregistré dans la base de données.
- 4.4.5. En cliquant sur le bouton « Ok » de la fenêtre contextuelle, la page revient à son affichage initial, comme dans l'étape 3.4.1.
- 4.4.6. Les étapes 3.4.2 réitérées.
- 4.4.7. Le joueur clique sur le bouton « Rent this game »
- 4.4.8. Une fenêtre contextuelle (Message-box) apparaît, confirmant le succès de l'ajout de l'emprunt (Loan), avec un bouton « Ok ». Un nouvel objet de copie est ainsi enregistré dans la base de données.
- 4.4.9. En cliquant sur le bouton « Ok » de la fenêtre contextuelle, la page revient à son affichage initial, comme dans l'étape 3.4.1.
- 4.4.10. Les étapes 3.4.2 réitérées.
- 4.4.11. Le joueur clique sur le bouton « Book this game »
- 4.4.12. Une fenêtre contextuelle (Message-box) apparaît, confirmant le succès de l'ajout de la réservation (Booking), avec un bouton « Ok ». Un nouvel objet de réservation est ainsi enregistré dans la base de données.

4.5. Le joueur sélectionne le menu "Home".

4.5.1. Une page s'affiche avec les éléments suivants :

- Son pseudonyme.
  - Le total de crédits qu'il possède.
  - Une liste déroulante des copies de jeux qu'il détient actuellement, avec un bouton "Delete" pour effacer.
  - Une liste déroulante des jeux qu'il est en train d'emprunter, accompagnée d'un bouton "Return".
  - Une liste déroulante des jeux qu'il a empruntés dans le passé (et déjà retournés), avec un bouton "Delete" pour effacer.
  - Une liste déroulante de ses réservations, avec un bouton "Cancel".
- 4.5.2. Le joueur sélectionne une copie qu'il possède dans la liste déroulante.
- 4.5.3. Ensuite, il clique sur le bouton "Delete".
- 4.5.4. Une fenêtre contextuelle (message-box) s'affiche pour demander confirmation de la suppression de la copie sélectionnée, offrant les options « yes » ou « no ».
- 4.5.5. Le joueur choisit l'option « yes ».

- 4.5.6. La copie effacée ne sera plus visible dans la liste déroulante des copies de jeux qu'il détient actuellement.
- 4.5.7. Le joueur sélectionne un jeu qu'il est en train d'emprunter dans la liste déroulante.
- 4.5.8. Ensuite, il clique sur le bouton "Return".
- 4.5.9. Une fenêtre contextuelle (message-box) s'affiche pour demander confirmation du retour du jeu sélectionné, offrant les options « yes » ou « no ».
- 4.5.10. Le joueur choisit l'option « yes ».
- 4.5.11. Le jeu retourné ne sera plus visible dans la liste déroulante des jeux qu'il emprunte actuellement, mais apparaîtra dans la liste déroulante des jeux qu'il a empruntés dans le passé (et déjà retournés).
- 4.5.12. Le joueur sélectionne un jeu qu'il est emprunter dans le passé dans la liste déroulante.
- 4.5.13. Ensuite, il clique sur le bouton "Delete".
- 4.5.14. Une fenêtre contextuelle (message-box) s'affiche pour demander confirmation de effacer du jeu sélectionné, offrant les options « yes » ou « no ».
- 4.5.15. Le joueur choisit l'option « yes ».
- 4.5.16. Le jeu effacée ne sera plus visible dans la liste déroulante des jeux qu'il emprunte actuellement, mais apparaîtra dans la liste déroulante des jeux qu'il a empruntés dans le passé.
- 4.5.17. Le joueur sélectionne un jeu qu'il est en train de réserver dans la liste déroulante « booking ».
- 4.5.18. Ensuite, il clique sur le bouton "Cancel".
- 4.5.19. Une fenêtre contextuelle (message-box) s'affiche pour demander confirmation de l'annulation de la réservation de ce jeu sélectionné, offrant les options « yes » ou « no ».
- 4.5.20. Le joueur choisit l'option « yes ».
- 4.5.21. Le jeu ne sera plus visible dans la liste déroulante des jeux qu'il réserve actuellement.
- 4.6. Le joueur clique sur le bouton "Exit".
- 4.7. La page du joueur se ferme.
- 4.8. La page de connexion (Login) s'ouvre.

## II. Scénario alternatif :

- 1.3. L'utilisateur remplit le formulaire Sign-up/ inscription avec des données (pseudo, ou une combinaison de nom d'utilisateur et de mot de passe) qui existent déjà, puis clique sur le bouton "Submit".
- 1.4. Un message d'erreur apparaît pour signaler la duplication et prévenir de la création de doublons.  
⇒
- 1.3. L'utilisateur ne remplit pas le formulaire d'inscription, puis clique sur le bouton "Submit".
- 1.4. Un message d'erreur apparaît pour demander de remplir tous les champs.  
⇒
- 2.1. Sur la page Login, l'administrateur saisit son nom d'utilisateur et son mot de passe, qui ne correspondent pas à ceux présents dans la table Admin de la base de données.
- 2.2. L'administrateur clique sur le bouton **Login**.
- 2.3. Un message d'erreur apparaît pour signaler l'erreur de saisie.  
⇒
- 2.1. Sur la page Login, l'administrateur ne saisit pas son nom d'utilisateur et/ou son mot de passe.
- 2.2. L'administrateur clique sur le bouton **Login**.
- 2.3. Un message d'erreur apparaît pour demander de saisir à la fois son nom d'utilisateur et son mot de passe.  
⇒
- 2.4.2. L'administrateur clique sur l'une des consoles de la liste (qui est liée à un ou plusieurs jeux vidéo), ce qui fait apparaître un bouton "Delete".
- 2.4.3. L'administrateur clique sur le bouton **Delete**.

2.4.4. Un message d'erreur s'affiche pour indiquer la contrainte qu'il n'est pas possible d'effacer cette console.

⇒

2.4.12. L'administrateur entre le nom d'une nouvelle console qui existe déjà et valide en cliquant sur le bouton **Add**.

2.4.13. Un message d'erreur apparaît pour signaler la duplication et prévenir de la création d'un doublon.

⇒

2.5.8. L'administrateur entre une nouvelle valeur de crédit qui est inférieure ou égale à 0, ou supérieure à 5, puis confirme en cliquant sur le bouton **Update**.

2.5.9. Un message d'erreur indique qu'il faut entrer un crédit entre 1 et 5.

⇒

2.5.17. L'administrateur ne remplit pas toutes les informations de nouveau Video game.

2.5.18. Ensuite, l'administrateur clique sur le bouton **Add**.

2.5.19. Un message d'erreur apparaît pour indiquer qu'il faut compléter tous les champs.

⇒

3.1. Sur la page Login, le joueur saisit son nom d'utilisateur et son mot de passe, qui ne correspondent pas à ceux présents dans la table Admin de la base de données.

3.2. Ensuite, le joueur clique sur le bouton **Login**.

3.3. Un message d'erreur apparaît pour signaler l'erreur de saisie.

⇒

3.1. Sur la page Login, le joueur ne saisit pas son nom d'utilisateur et/ou son mot de passe.

3.2. Ensuite, le joueur clique sur le bouton **Login**.

3.3. Un message d'erreur apparaît pour demander de saisir à la fois son nom d'utilisateur et son mot de passe.

⇒

3.4.3. Le joueur clique sur le bouton "I have this game" alors qu'il est déjà propriétaire de ce jeu.

3.4.4. Une fenêtre contextuelle (Message-box) apparaît pour indiquer que le joueur est déjà propriétaire de ce jeu vidéo.

⇒

3.4.7. Le joueur clique sur le bouton « Rent this game » alors qu'il est en train de louer ce jeu.

3.4.8. Une fenêtre contextuelle (Message-box) apparaît pour indiquer que le joueur ne peut pas louer le jeu qu'il est en train de louer.

⇒

3.4.11. Le joueur clique sur le bouton « Book this game » alors qu'il est en train de louer ce jeu ou en train de le réserver.

3.4.12. Une fenêtre contextuelle (Message-box) apparaît pour indiquer que le joueur ne peut pas réserver le jeu qu'il est en train de louer ou de réserver.

⇒

3.5.2. Le joueur sélectionne une copie qu'il possède, mais ce jeu est en train d'être loué par un autre joueur.

3.5.3. Ensuite, il clique sur le bouton "Delete".

3.5.4. Une fenêtre contextuelle (Message-box) apparaît pour indiquer que ce n'est pas possible d'effacer un jeu qui est en train d'être loué.