

**IIR**

**INGÉNIERIE  
INFORMATIQUE & RESEAUX**

**Stage d'été**

**2023/2024**

**THEME**

---

**Développement d'une Plateforme  
de Streaming Vidéo Inspirée de Netflix  
avec Déploiement Cloud**

**RÉALISÉ PAR**

**SOULAIMANE OUHMIDA**

**ENCADRÉ PAR**

**MR. CHARIF**

**SOUTENU LE**

**01/09/2024**

## Dedicace

A mes parents, dont le soutien et les encouragements indéfectibles ont été mon pilier tout au long de mes études et de ce stage. À mes sœurs Salma et Souhaila, pour leur soutien inconditionnel et leur présence réconfortante à chaque étape de ma vie. A mes amis, pour leur soutien indéfectible et leurs précieux encouragements. A M. Charif, pour sa patience, ses conseils avisés et la confiance qu'il m'a accordé. Ce rapport est dédié à tous ceux qui ont contribué, directement ou indirectement, à mon expérience de stage.

Je tiens également à remercier Saad Riadi, dont les conseils et les retours ont été déterminants pour façonner mon parcours au cours de ce stage. Leur sagesse et leurs encouragements ont eu un impact profond sur mon développement personnel et professionnel.

Enfin, j'adresse mes sincères remerciements à toute l'équipe de Concentrix pour m'avoir accueilli chaleureusement et m'avoir offert un environnement enrichissant pour apprendre et grandir.

## Remerciement

Je tiens tout d'abord à exprimer ma profonde gratitude envers Charif pour son soutien constant, ses conseils avisés et la confiance qu'il m'a accordée tout au long de mon stage. Ses précieux enseignements et son accompagnement ont été essentiels pour mon développement professionnel.

Mes remerciements vont également à l'équipe de Concentrix, particulièrement à Ezzadine, pour leur accueil chaleureux, leur collaboration et leur volonté de partager leurs connaissances. Leur expertise m'a permis d'acquérir une compréhension approfondie du domaine et de contribuer de manière significative aux projets de l'entreprise.

Je souhaite exprimer ma reconnaissance envers mes collègues, qui ont rendu mon expérience de travail enrichissante et agréable. Leur soutien et leur camaraderie ont contribué grandement à mon épanouissement professionnel au sein de l'équipe.

Enfin, je suis reconnaissant envers ma famille et mes amis pour leur soutien inconditionnel et leurs encouragements constants tout au long de cette période.

# Résumé

À une époque où les services de streaming numérique sont omniprésents, des plateformes telles que Netflix et Disney+ incarnent parfaitement la fusion entre technologie avancée et divertissement. Ces entreprises déploient des technologies de pointe pour offrir du contenu visuel de qualité à un large public. Ce projet ambitieux a pour objectif de recréer l'infrastructure complète de Netflix, comprenant une interface utilisateur attrayante, des serveurs puissants, et une architecture de base de données complexe qui soutient une expérience utilisateur fluide et immersive.

L'interface utilisateur est conçue pour capter et maintenir l'attention des utilisateurs, tandis que les systèmes backend sophistiqués gèrent la diffusion du contenu et l'administration des comptes. Chaque composant de ce projet vise à reproduire les fonctionnalités distinctives qui ont fait la renommée de Netflix. En réalisant cette réplique détaillée, le projet met en lumière non seulement des compétences techniques impressionnantes, mais il explore également les défis de l'évolutivité, de la distribution de contenu, et du design centré sur l'utilisateur dans le paysage moderne des médias.

Ce travail nous conduit à décortiquer et à recréer la magie qui a permis à Netflix de devenir un acteur incontournable du divertissement mondial. En approfondissant ces aspects, le projet aspire à offrir une compréhension approfondie des technologies et des stratégies qui sous-tendent le succès des géants du streaming.

# Abstract

In an era where digital streaming services are ubiquitous, platforms like Netflix and Disney+ epitomize the fusion of advanced technology and entertainment. These companies deploy cutting-edge technologies to deliver quality visual content to a wide audience. This ambitious project aims to recreate Netflix's entire infrastructure, including a compelling user interface, powerful servers, and a complex database architecture that supports a seamless and immersive user experience.

The user interface is designed to capture and maintain users' attention, while sophisticated backend systems handle content delivery and account administration. Each component of this project aims to replicate the distinctive features that Netflix is known for. By creating this detailed replica, the project not only showcases impressive technical skills, but also explores the challenges of scalability, content distribution, and user-centric design in the modern media landscape.

This work leads us to unpack and recreate the magic that has allowed Netflix to become a global entertainment powerhouse. By delving into these aspects, the project aims to provide an in-depth understanding of the technologies and strategies that underpin the success of streaming giants.

## Glossaire

Cloud	
API REST (Rest API)	
Agile	
Backend	
Frontend	
AWS	
API	
AWS S3	

## Liste des Figures

Figure 1: Logo de Concentrix .....	15
Figure 2: Emplacement du Concentrix dans le monde .....	16
Figure 3: Les valeurs de concentrix.....	16
Figure 4: Concentrix Maroc.....	18
Figure 5: Logo Webhelp.....	20
Figure 6: Emplacement du Webhelp dans le monde .....	21
Figure 7: Les valeurs de webhelp.....	21
Figure 8: Concentrix + Webhelp .....	22
Figure 9: Pile technologique Netflix .....	24
Figure 10: Pile technologique de ce projet .....	27
Figure 11: Logo de React JS .....	27
Figure 12: Logo de JavaScript.....	28
Figure 13: Logo de HTML .....	28
Figure 14: Logo de CSS .....	28
Figure 15: Logo de Tailwind.....	28
Figure 16: Logo de Java .....	29
Figure 17: Logo de Spring Boot.....	29
Figure 18: Logo de MySQL.....	29
Figure 19: Logo de Cassandra .....	30
Figure 20: Logo de Redis.....	30
Figure 21: Logo de AWS.....	30
Figure 22: Logo de Git .....	30
Figure 23: Logo de Jira .....	31
Figure 24 : Diagramme de cas d'utilisation .....	35
Figure 25 : Diagramme de classes .....	37
Figure 26: Diagramme de séquence " authentification " .....	38
Figure 27: Diagramme de séquence " Paiement de l'abonnement " .....	39
Figure 28: Diagramme de séquence " Navigation et lecture de vidéo " .....	40
Figure 29: Architecture du backend.....	46
Figure 30: Structure du frontend.....	48
Figure 31: Page d'accueil. ....	49
Figure 32: Page d'inscription. ....	50
Figure 33: Page de création du profil.....	51
Figure 34: Page de Connexion (Login).....	52
Figure 35: Page de connexion.....	53
Figure 36: Page de choisir le profil .....	54
Figure 37: Page d'accueil .....	54
Figure 38: Choisir ( film / serie ).....	55
Figure 39: Page (film / serie).....	56
Figure 40: Page d'un film .....	57
Figure 41: Page d'un film apres ajouter aux favoris .....	57
Figure 42: Page de favoris.....	58
Figure 43: Diagramme du cloud .....	62

## Liste des tableaux

Tableau 1: Fiche technique du Concentrix .....	15
Tableau 2: Secteurs d'activité de concentrix .....	17
Tableau 3: Fiche technique du Webhelp .....	20
Tableau 4: Les secteurs d'activité de webhelp .....	22
Tableau 5: Stack technologique frontend (netflix) .....	25
Tableau 6: Stack technologique backend (netflix) .....	25
Tableau 7: Stack technologique bases de données (netflix) .....	26
Tableau 8: Stack technologique big data (netflix) .....	26
Tableau 9: Stack technologique devops (netflix) .....	26
Tableau 10: Les besoins clés de l'utilisateur final. ....	33
Tableau 11: Les besoins fonctionnelles. ....	34
Tableau 12: Les besoins non fonctionnelles. ....	34
Tableau 13: les types de bases de données. ....	42
Tableau 14: Technologies utilisées dans backend.....	47
Tableau 15: Technologies utilisées dans frontend. ....	48
Tableau 16: Les principaux composants cloud .....	60
Tableau 17: Les services AWS Utilisés .....	61
Tableau 18: Les étapes de déploiement du Frontend (React) .....	63
Tableau 19: Les étapes de déploiement du Backend (Spring Boot) .....	63
Tableau 20: Les étapes de la configuration des Bases de Données .....	64
Tableau 21: Sécurité et Conformité .....	64



# Table des matières

Dedicace.....	2
Remerciement.....	3
Résumé .....	4
Abstract.....	5
Glossaire.....	6
Liste des Figures.....	7
Liste des tableaux .....	8
Table des matières .....	9
Introduction Générale.....	12
Contexte .....	12
Problématique .....	12
Objectifs .....	12
Méthodologie.....	12
Structure du Rapport .....	13
Chapitre 1 : Présentation de l'organisme d'accueil.....	14
Introduction .....	15
I.    Concentrix .....	15
Présentation.....	15
Les valeurs de Concentrix.....	16
Secteurs d'activité.....	17
Concentrix Maroc.....	18
II.   Webhelp .....	20
Presentaion.....	20
Les valeurs de Webhelp.....	21
Secteurs d'activité.....	22
III.  Concentrix + WebHelp .....	22
Conclusion .....	22
Chapitre 2 : Etude de projet et outils utilisés .....	23
Introduction .....	24
I.    Pile technologique Netflix .....	24
1.  Frontend .....	25
2.  Backend.....	25
3.  Bases de Données.....	26

4. Big Data.....	26
5. DevOps.....	26
II. Pile technologique de ce projet.....	27
III. Outils de développement .....	27
1. Frontend .....	27
2. Backend.....	29
3. Databases .....	29
4. Cloud .....	30
5. Gestion du Code Source.....	30
6. Outils de Collaboration.....	31
Conclusion .....	31
Chapitre 3 : Analyse et Conception .....	32
Introduction .....	33
I. Analyse des Besoins .....	33
1. Analyse des Besoins Utilisateurs .....	33
2. Analyse Fonctionnelle .....	33
3. Analyse Non Fonctionnelle .....	34
II. Modélisation UML.....	34
Diagrammes de cas d'utilisation.....	34
Diagrammes de classes.....	36
Diagrammes de séquence .....	37
Conclusion .....	40
Chapitre 4 : Conception de la Base de Données.....	41
Introduction .....	42
I. Choix de la Base de Données .....	42
II. Optimisation .....	43
Optimisation de Cassandra .....	43
Optimisation de MySQL .....	43
Optimisation de Redis .....	43
III. Collecte des données .....	43
Introduction au Web Scraping .....	43
Stockage des Données Collectées.....	44
Conclusion .....	44
Chapitre 5 : Mise en œuvre et Réalisation .....	45

Introduction .....	46
I. Développement Backend .....	46
Technologies et Outils Utilisés .....	47
Architecture Backend .....	47
Conception des Endpoints API.....	47
II. Développement Frontend .....	48
Technologies Utilisées .....	48
Développement des Fonctionnalités Principales .....	49
III. Interaction Frontend-Backend.....	53
Exemple d'Interaction Utilisateur .....	53
Récupération des Données.....	55
Mise à jour en Temps Réel .....	56
Conclusion .....	58
Chapitre 6 : Déploiement Cloud .....	59
Introduction .....	60
I. Architecture Cloud.....	60
Présentation Générale .....	60
Services AWS Utilisés.....	60
II. Processus de Déploiement.....	62
Déploiement du Frontend (React) .....	62
Déploiement du Backend (Spring Boot) .....	63
Configuration des Bases de Données .....	63
III. Sécurité et Conformité .....	64
Conclusion .....	64
Conclusion Générale et Perspectives .....	65
Annexe .....	66
Références .....	67

# Introduction Générale

Lors de mon stage d'été, j'ai entrepris de développer une application qui reproduit les principales fonctionnalités de Netflix, une plateforme de streaming vidéo de renommée mondiale. Ce projet ambitieux vise à intégrer la gestion des utilisateurs, le streaming de vidéos, et une interface utilisateur intuitive et engageante.

## Contexte

L'industrie du streaming vidéo connaît une croissance spectaculaire, portée par des plateformes comme Netflix qui redéfinissent la consommation de médias. Cette transformation est le fruit de progrès technologiques notables dans le développement web, la gestion de bases de données, et les services cloud, permettant une diffusion fluide et accessible du contenu audiovisuel.

## Problématique

La création d'un clone de Netflix pose plusieurs défis techniques majeurs, notamment la gestion efficace des flux de données, la protection des informations sensibles des utilisateurs, et l'optimisation des performances pour une expérience utilisateur fluide.

Ce projet explore ces défis et propose des solutions en s'appuyant sur des technologies modernes et une approche de développement agile.

## Objectifs

Les principaux objectifs de ce projet sont :

- Développer une interface utilisateur moderne, réactive et conviviale.
- Construire un serveur web solide, accompagné d'une API REST pour la gestion des données.
- Concevoir et implémenter une base de données optimisée pour le stockage rapide et la récupération efficace des informations vidéo.
- Utiliser des services cloud pour garantir la scalabilité et la disponibilité continue de l'application.

## Méthodologie

Le projet a été réalisé selon une approche Agile, avec des sprints hebdomadaires facilitant des itérations rapides et des ajustements basés sur les retours d'expérience. Les outils utilisés incluent Git pour le contrôle de version et Jira pour la gestion des tâches.

## Structure du Rapport

Ce rapport est organisé en plusieurs chapitres, chacun décrivant une étape spécifique du projet. Le premier chapitre introduit l'organisme d'accueil. Les chapitres suivants détaillent l'analyse et la conception du projet, la conception de la base de données, le développement frontend et backend, le déploiement cloud, les mesures de sécurité, et les tests effectués.

Le rapport conclut par une analyse des défis rencontrés et des suggestions pour les améliorations futures.

---

# CHAP1: Présentation de l'organisme d'accueil

---

## Introduction

Ce premier chapitre abordera la présentation de l'organisme d'accueil, l'entreprise Concentrix, ses domaines d'activités et ses métiers, et son rapprochement avec Webhelp.

### I. Concentrix Présentation



Figure 1: Logo de Concentrix

Concentrix Corporation est l'un des principaux fournisseurs mondiaux de solutions et de technologies d'expérience client (CX), améliorant les performances commerciales de certaines des meilleures marques au monde, dont plus de 100 clients du Fortune Global 500 et plus de 125 clients de la nouvelle économie.

Création	Concentrix a été fondée en 1983.
Forme Juridique	Société anonyme (corporation) cotée en bourse.
Slogan	We deliver extraordinary customer experiences.
Présence Mondiale	plus de 40 pays à travers le monde.
Siège Sociale	Fremont, Californie, États-Unis.
Directeur Général (CEO)	Chris Caldwell
Produits	<ul style="list-style-type: none"> <li>• Gestion de l'expérience client</li> <li>• Externalisation des processus métiers (BPO)</li> <li>• Services informatiques</li> <li>• Solutions de technologie</li> <li>• Analytique et intelligence artificielle</li> <li>• Services de gestion des interactions clients (centres de contact, support technique, etc.)</li> </ul>
Filiales	<ul style="list-style-type: none"> <li>• Concentrix Catalyst</li> <li>• TigerSpike</li> <li>• Minacs</li> </ul>
Effectif	Environ 290,000 employés à travers le monde.
Site Web	concentrix.com
Chiffre d'Affaire	Environ 5,6 milliards de dollars en 2021.

Tableau 1: Fiche technique du Concentrix

La figure suivante représente les régions de présence de Concentrix au monde et la répartition de ses consultants :

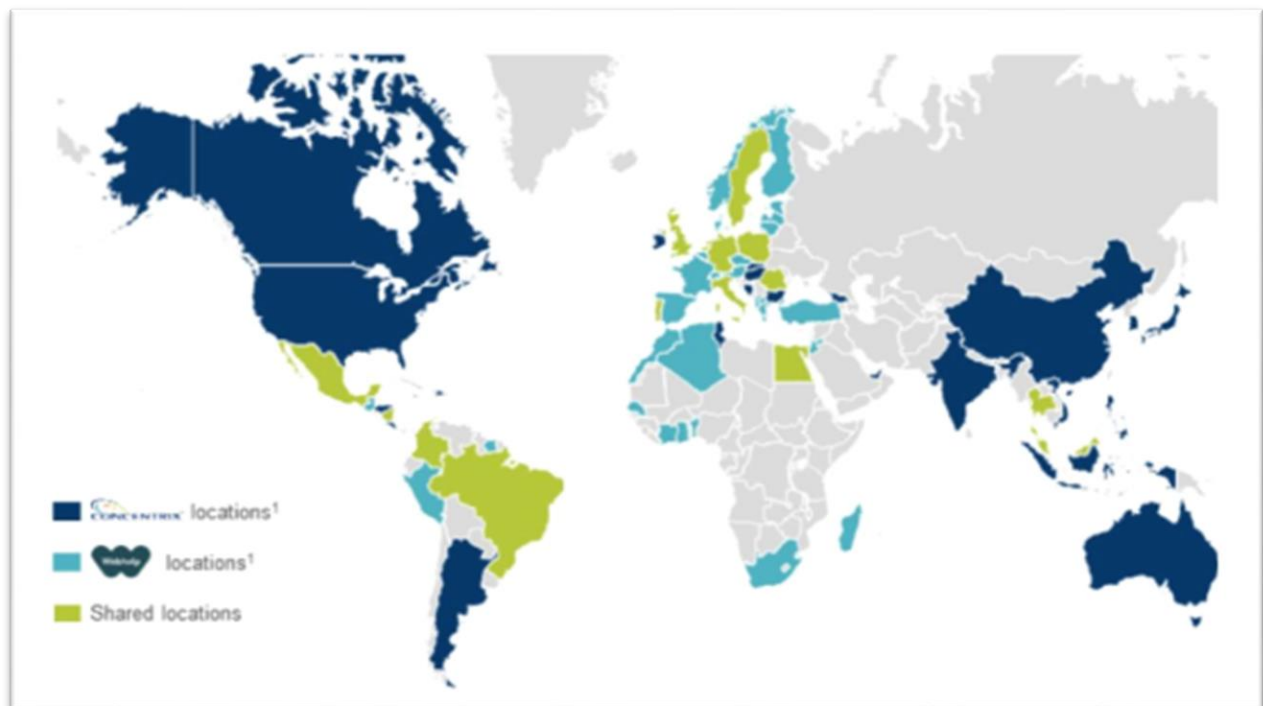


Figure 2: Emplacement du Concentrix dans le monde

## Les valeurs de Concentrix

Les valeurs et la culture de concentrix repose sur le respect d'un soutien individuel et mutuel entre collaborateurs, et d'une véritable collaboration. Les sept valeurs de Concentrix sont :



Figure 3: Les valeurs de concentrix



## Secteurs d'activité

Concentrix opère dans plusieurs secteurs d'activité, fournissant une gamme diversifiée de services pour répondre aux besoins spécifiques de chaque industrie. Voici quelques-uns des principaux secteurs d'activité de Concentrix :

Technologie	<ul style="list-style-type: none"> <li>• Services de support technique</li> <li>• Gestion des relations clients pour les entreprises technologiques</li> <li>• Solutions de service après-vente</li> </ul>
Télécommunications	<ul style="list-style-type: none"> <li>• Assistance clientèle pour les opérateurs de téléphonie mobile et les fournisseurs d'accès internet</li> <li>• Gestion des ventes et du support technique</li> </ul>
Services financiers	<ul style="list-style-type: none"> <li>• Services de gestion des comptes</li> <li>• Support client pour les banques et les compagnies d'assurance</li> <li>• Services de recouvrement et de gestion des fraudes</li> </ul>
Santé	<ul style="list-style-type: none"> <li>• Services de support aux patients</li> <li>• Gestion des réclamations d'assurance santé</li> <li>• Services de télésanté et de soins à distance</li> </ul>
Commerce de détail et e-commerce	<ul style="list-style-type: none"> <li>• Gestion des commandes et des retours</li> <li>• Support client pour les boutiques en ligne et les détaillants traditionnels</li> <li>• Solutions de marketing numérique</li> </ul>
Automobile	<ul style="list-style-type: none"> <li>• Services de support pour les constructeurs automobiles</li> <li>• Gestion des réclamations de garantie</li> <li>• Support technique pour les véhicules connectés</li> </ul>
Voyages et loisirs	<ul style="list-style-type: none"> <li>• Services de réservation et de support client pour les compagnies aériennes, les hôtels et les agences de voyages</li> <li>• Gestion des programmes de fidélité</li> </ul>
Services publics et gouvernement	<ul style="list-style-type: none"> <li>• Assistance clientèle pour les services publics</li> <li>• Solutions de gestion des citoyens pour les agences gouvernementales</li> </ul>
Éducation	<ul style="list-style-type: none"> <li>• Services de support pour les établissements d'enseignement</li> <li>• Gestion des inscriptions et des programmes éducatifs en ligne</li> </ul>
Médias et divertissement	<ul style="list-style-type: none"> <li>• Support client pour les plateformes de streaming et de médias</li> <li>• Gestion des abonnements et des services à la demande</li> </ul>

Tableau 2: Secteurs d'activité de concentrix

## Concentrix Maroc

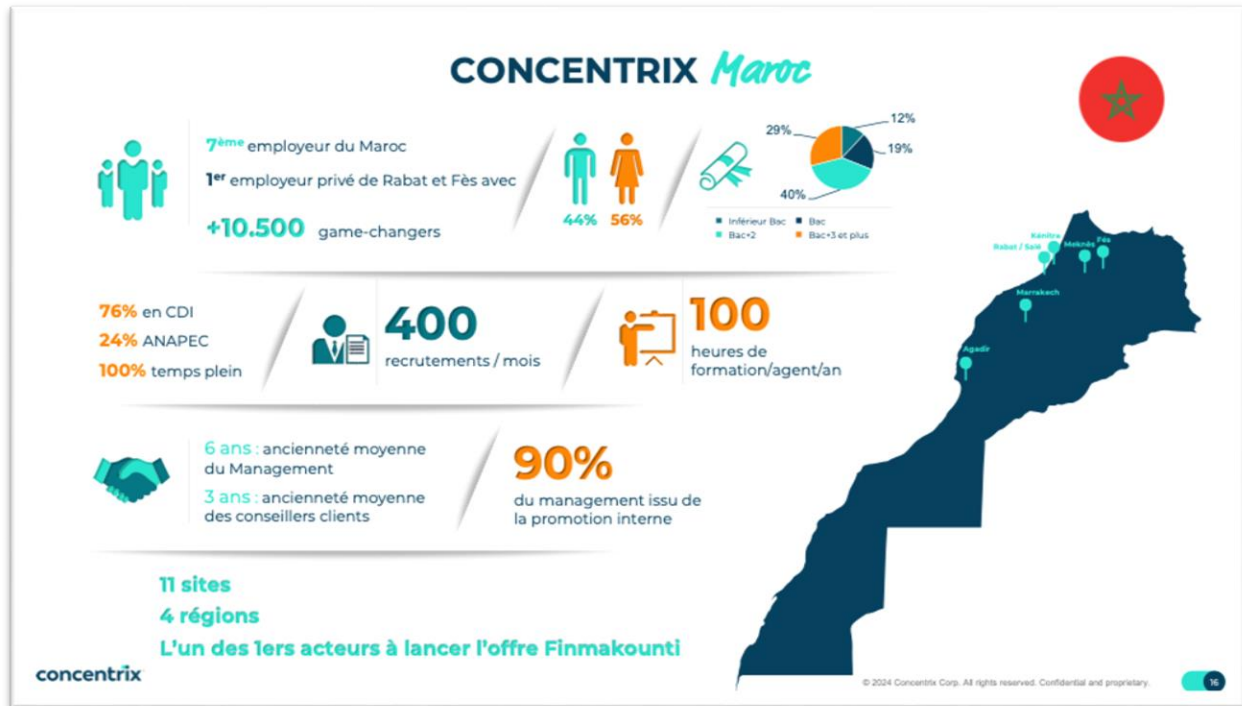


Figure 4: Concentrix Maroc

La figure suivante est la représentation schématique des liens hiérarchiques de Concentrix Maroc :



## II. Webhelp

### Présentation



Figure 5: Logo Webhelp

Webhelp est une entreprise française d'externalisation de la gestion de l'expérience client et des processus métier. Elle a été fondée en 2000 et son siège est situé à Paris en France.

Concrètement, ce groupe dirige des centres d'appels qui proposent à des entreprises des prestations de ligne directe, de télémarketing, de traitement de courriers et d'e-mails.

Création	Webhelp a été fondée en 2000.
Forme Juridique	Société par actions simplifiée (SAS).
Slogan	Making business more human.
Présence Mondiale	plus de 50 pays avec plus de 200 sites à travers le monde.
Siège Sociale	161 Rue de Courcelles, 75017 Paris, France.
Direction	<ul style="list-style-type: none"> <li>• <b>Co-fondateurs</b> : Frédéric Jousset et Olivier Duha</li> <li>• <b>CEO</b> : Olivier Duha</li> </ul>
Produits	<ul style="list-style-type: none"> <li>• Gestion de l'expérience client</li> <li>• Externalisation des processus métiers (BPO)</li> <li>• Services de centres de contact</li> <li>• Services de support technique et d'assistance</li> <li>• Solutions de transformation numérique</li> <li>• Services de modération de contenu</li> <li>• Services de gestion des réseaux sociaux</li> </ul>
Filiales	<ul style="list-style-type: none"> <li>• Webhelp Payment Services</li> <li>• Webhelp Enterprise Sales Solutions</li> <li>• Webhelp Digital Consulting</li> </ul>
Effectif	Environ 100,000 employés à travers le monde.
Site Web	webhelp.com
Chiffre d'Affaire	Environ 2,5 milliards euros en 2021.

Tableau 3: Fiche technique du Webhelp

La figure suivante représente les régions de présence de Concentrix au monde et la répartition de ses consultants :



Figure 6: Emplacement du Webhelp dans le monde

## Les valeurs de Webhelp

Webhelp, dans l'ensemble du groupe est attaché à ses 5 valeurs : la reconnaissance, l'unité, l'engagement, l'exemplarité, le Wahou.



Figure 7: Les valeurs de webhelp

## Secteurs d'activité

Webhelp opère principalement dans les secteurs suivants :

Expérience client	Fourniture de solutions pour améliorer l'interaction client à travers différents canaux comme le service client, le support technique, et la gestion des relations client.
Externalisation des processus métier (BPO)	Gestion déléguée de processus métier tels que la gestion des ressources humaines, la comptabilité, et d'autres fonctions administratives.
Conseil et technologie	Offre de conseils stratégiques et de solutions technologiques pour optimiser les opérations commerciales et améliorer l'efficacité opérationnelle.
Santé et bien-être	Services spécialisés dans le domaine de la santé, y compris la gestion des dossiers médicaux et l'assistance aux patients.

Tableau 4: Les secteurs d'activité de webhelp

## III. Concentrix + WebHelp



Figure 8: Concentrix + Webhelp

Le 25 septembre 2023, Concentrix a annoncé avoir finalisé son rapprochement avec Webhelp et que l'intégration des deux sociétés est en cours. Pendant que la société issue de la fusion finalise son nom permanent, elle opérera sous le nom commercial Concentrix + Webhelp.

## Conclusion

Dans ce chapitre, le cadre général du projet a été décrit. Après avoir présenté l'organisme d'accueil, présenté le projet, nous avons décrit par le périmètre de ce stage, la méthodologie adoptée pour atteindre la bonne conduite du projet. Le chapitre suivant sera consacré à l'étude fonctionnelle et technique du projet.

---

## CHAP2 : Etude de Projet et Outils Utilisés

---

## Introduction

Ce chapitre se concentre sur l'étude approfondie du projet et les outils techniques utilisés pour développer l'application. Le but de cette section est de fournir une vue d'ensemble des choix technologiques, des frameworks et des bibliothèques, ainsi que des raisons derrière leur sélection.

### I. Pile technologique Netflix

Netflix, utilise une infrastructure technologique avancée pour offrir une expérience utilisateur fluide et une disponibilité constante. Cette partie explore le stack technologique de Netflix, en mettant en lumière les principaux composants et technologies utilisés pour gérer le vaste écosystème de contenu, les millions d'utilisateurs et les flux de données massifs. Cette analyse comprend les aspects frontend, backend, infrastructure cloud, bases de données et sécurité.

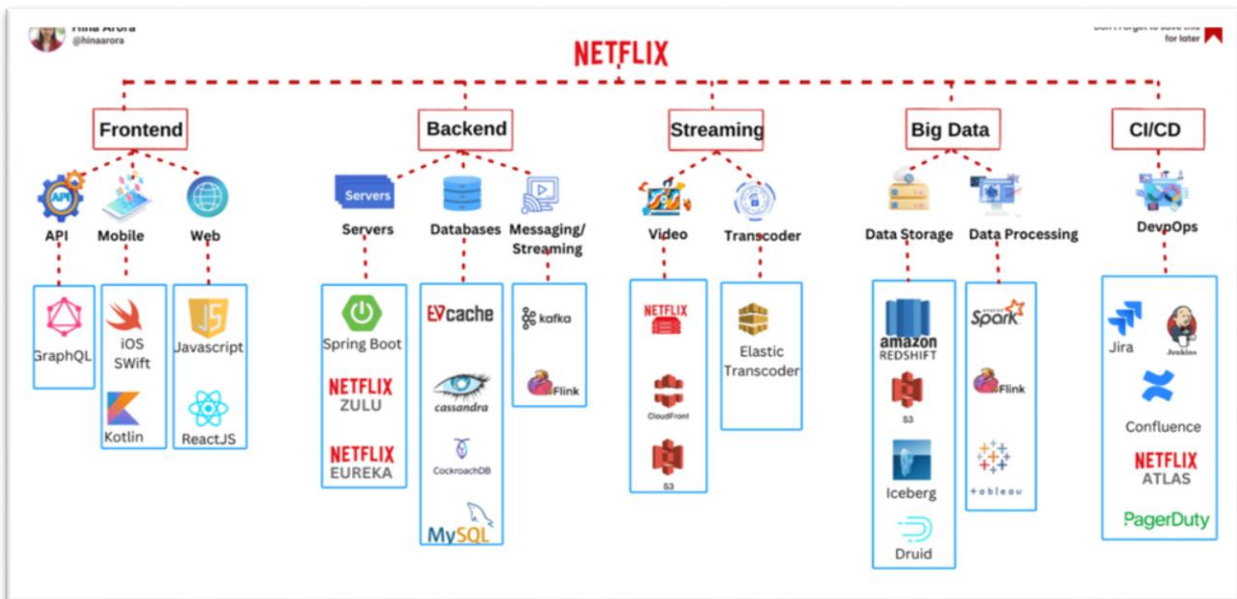


Figure 9: Pile technologique Netflix



## 1. Frontend

React JS (Web)	Framework de JavaScript Utilisé pour le développement des interfaces utilisateur web, cette technologie permet de créer des expériences utilisateur dynamiques et réactives.
Swift (Mobile)	Langage de programmation Utilisé pour le développement des applications iOS, elle permet de créer des applications performantes et sécurisées.
Kotlin (Mobile)	Langage de programmation utilisé pour le développement d'applications Android, offrant une syntaxe moderne et des fonctionnalités robustes.
GraphQL (API)	Une technologie utilisée pour la communication entre le frontend et le backend. Elle permet au frontend de demander exactement les données dont ils ont besoin.

Tableau 5: Stack technologique frontend (netflix)

## 2. Backend

Spring Boot	Framework de Java utilisé pour développer des serveurs web robustes et scalables. Spring Boot facilite le développement rapide de services web.
Apache Kafka	Plateforme de streaming utilisée pour gérer les flux de données en temps réel.
Apache Flink	Framework pour le traitement des données en streaming.

Tableau 6: Stack technologique backend (netflix)

### 3. Bases de Données

Cassandra	Base de données NoSQL utilisée pour sa scalabilité et sa capacité à gérer de grandes quantités de données.
CockroachDB & MySQL	Bases de données relationnelles (SQL) utilisées pour les transactions nécessitant une consistance stricte.
EVCache	Un système de cache distribué basé sur Memcached.

Tableau 7: Stack technologique bases de données (netflix)

### 4. Big Data

Amazon S3, Iceberg	Des services pour le stockage des données massives et les logs.
Apache Spark et Apache Flink	Des services utilisés pour le traitement des données en batch et en streaming.
Tableau	Outil de visualisation des données pour des analyses interactives.

Tableau 8: Stack technologique big data (netflix)

### 5. DevOps

Jira et Confluence	Outils utilisés pour la gestion de projet et la documentation.
Jenkins	Serveur d'intégration continue pour l'automatisation des tâches de développement et de déploiement.

Tableau 9: Stack technologique devops (netflix)

## II. Pile technologique de ce projet

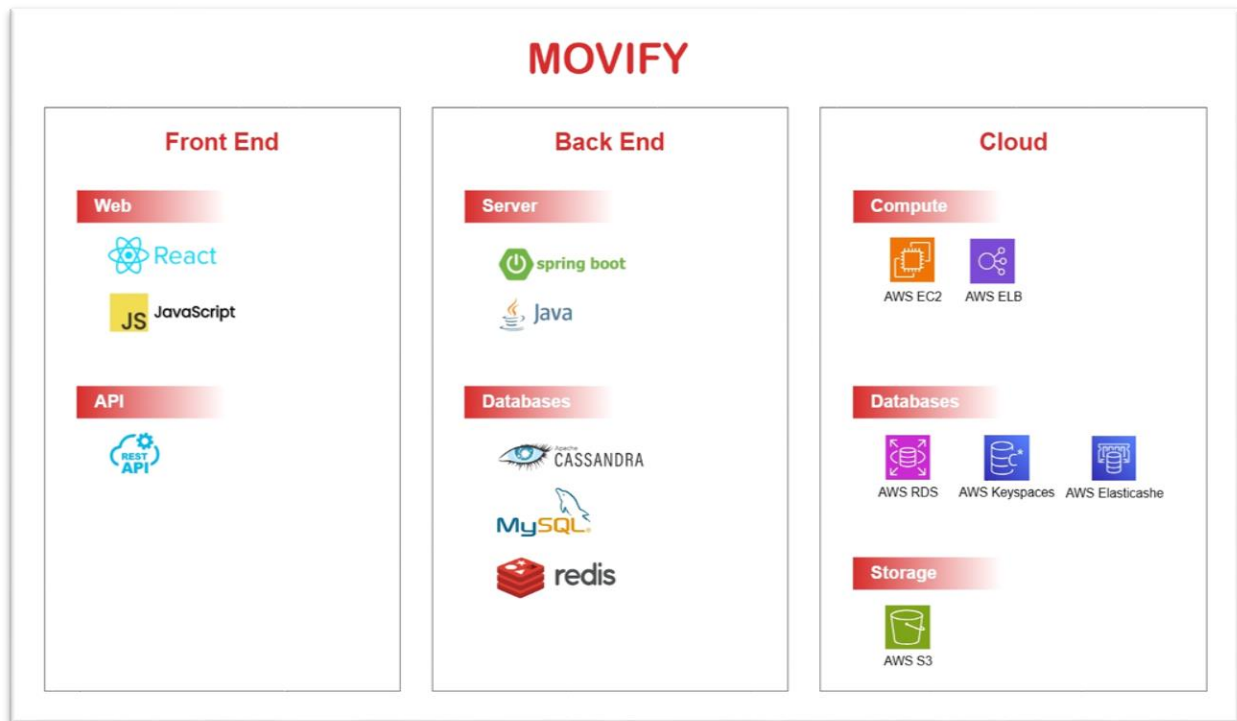


Figure 10: Pile technologique de ce projet

## III. Outils de développement

### 1. Frontend

*React JS*



Figure 11: Logo de React JS

React est une bibliothèque JavaScript open-source qui est utilisée pour construire des interfaces utilisateur spécifiquement pour des applications d'une seule page. Elle est utilisée pour gérer la couche d'affichage des applications web et mobiles. React a été créé par Jordan Walke, un ingénieur logiciel travaillant pour Facebook. React a été déployé pour la première fois sur Facebook en 2011 et sur Instagram en 2012.

## JavaScript



Figure 12: Logo de JavaScript

JavaScript est un langage de programmation qui permet d'implémenter des mécanismes complexes sur une page web. À chaque fois qu'une page web fait plus que simplement afficher du contenu statique.

## HTML



Figure 13: Logo de HTML

HTML est l'abréviation de « hypertext markup language » (langage de balisage hypertexte) et est un langage relativement simple utilisé pour créer des pages web. Comme il n'autorise pas les variables ou les fonctions, il n'est pas considéré comme un « langage de programmation », mais plutôt comme un « langage de balisage ».

## CSS



Figure 14: Logo de CSS

CSS désigne Cascading Style Sheets (pour Feuilles de style en cascade). Il s'agit d'un langage de style dont la syntaxe est extrêmement simple mais son rendement est remarquable. En effet, le CSS s'intéresse à la mise en forme du contenu intégré avec du HTML.

## Tailwind JS



Figure 15: Logo de Tailwind

Tailwind CSS est un framework permettant aux développeurs de personnaliser totalement et simplement le design de leur application ou de leur site web. Avec ce

framework CSS, il est possible de créer un design d'interface au sein même du fichier HTML.

## 2. Backend

### *Java*



Figure 16: Logo de Java

Java est un langage de programmation multiplateforme, orienté objet et largement utilisé pour coder des applications Web. Il s'agit d'un choix populaire parmi les développeurs depuis plus de deux décennies.

### *Spring Boot*



Figure 17: Logo de Spring Boot

Spring Boot est un framework de développement JAVA. C'est une déclinaison du framework classique de Spring qui permet essentiellement de réaliser des microservices (ce sont la majeure partie du temps des services web qui sont regroupés en API).

## 3. Databases

### *MySQL*



Figure 18: Logo de MySQL

MySQL est un système de gestion de bases de données relationnelles SQL open source développé et supporté par Oracle. Peut être utilisé par les développeurs et des administrateurs de bases de données pour gérer efficacement les données de leurs projets.

## Cassandra



Figure 19: Logo de Cassandra

Il s'agit d'un système de gestion de bases de données (SGBD) NoSQL open source. Cela signifie qu'il stocke les données sous forme de clé-valeur. Il stocke et manipule les données pour les restructurer.

## Redis



Figure 20: Logo de Redis

Redis est un magasin de structures de données clé-valeur rapide, open source et en mémoire. Il vous permet de stocker des paires clé-valeur sur votre RAM. L'accès à la RAM est 150 000 fois plus rapide que l'accès à un disque et 500 fois plus rapide que l'accès au SSD.

## 4. Cloud

### AWS



Figure 21: Logo de AWS

AWS (Amazon Web Services) est une plateforme de cloud computing fournie par Amazon. Offre des outils tels que la puissance de calcul, le stockage de bases de données et les services de diffusion de contenu.

## 5. Gestion du Code Source



Figure 22: Logo de Git

Git est un outil DevOps utilisé pour la gestion du code source. Il s'agit d'un système de contrôle de version gratuit et open source utilisée pour gérer efficacement des projets de petite à très grande envergure. Git est habitué à suivre les modifications du code source.

## 6. Outils de Collaboration



Figure 23: Logo de Jira

Jira est l'outil de gestion de projet agile n°1 utilisé par les équipes pour planifier, suivre, publier et prendre en charge des logiciels de classe mondiale en toute confiance. Il s'agit de la source unique de vérité pour l'ensemble de votre cycle de développement, offrant aux équipes autonomes le contexte nécessaire pour agir rapidement tout en restant connectées à l'objectif commercial global.

## Conclusion

Le choix des technologies et des outils utilisés dans ce projet a été guidé par la nécessité de créer une application robuste, évolutive et facile à maintenir. Chaque technologie a été sélectionnée en fonction de ses capacités à répondre aux besoins spécifiques du projet, tout en prenant en compte les meilleures pratiques de l'industrie.

Ce chapitre a fourni une vue d'ensemble des considérations techniques et des décisions prises pour assurer le succès du développement de l'application clone de Netflix.

---

## CHAP3 : Analyse et Conception

---



## Introduction

Ce chapitre présente l'analyse et la conception du projet. Il vise à définir les besoins des utilisateurs, concevoir l'architecture du système et spécifier les aspects fonctionnels et techniques du projet.

Cette phase est cruciale car elle permet de préparer le terrain pour le développement et la mise en œuvre réussis du système.

### I. Analyse des Besoins

#### 1. Analyse des Besoins Utilisateurs

Pour concevoir une solution efficace, il est essentiel de comprendre les besoins des utilisateurs finaux. L'utilisateur cible est un utilisateur régulier qui cherche une expérience de streaming fluide, une interface intuitive et des recommandations personnalisées.

Les besoins clés identifiés sont :

Streaming haute définition	Les utilisateurs souhaitent pouvoir regarder des vidéos en qualité optimale.
Recommandations personnalisées	Offrir des suggestions basées sur les préférences et l'historique de visionnage.
Gestion des profils	Permettre aux utilisateurs de créer et de gérer plusieurs profils avec des paramètres personnalisés.

Tableau 10: Les besoins clés de l'utilisateur final.

#### 2. Analyse Fonctionnelle

Lecture de vidéos	Fonctionnalité de streaming avec support pour les vidéos en haute définition et la lecture en continu.
-------------------	--

Système de recommandation	Algorithmes pour proposer des contenus en fonction des préférences des utilisateurs.
Gestion des profils utilisateur	Interface permettant aux utilisateurs de créer, modifier et supprimer des profils.

Tableau 11: Les besoins fonctionnelles.

Le diagramme de cas d'utilisation montre comment les différents acteurs interagissent avec ces fonctionnalités.

### 3. Analyse Non Fonctionnelle

Performance	Le système doit être capable de gérer un grand nombre d'utilisateurs simultanément avec un temps de réponse minimal.
Sécurité	Protection des données des utilisateurs contre les accès non autorisés et les fuites de données.
Scalabilité	Le système doit pouvoir évoluer pour supporter une augmentation du nombre d'utilisateurs sans dégradation des performances.

Tableau 12: Les besoins non fonctionnelles.

## II. Modélisation UML

### Diagrammes de cas d'utilisation

Les diagrammes de cas d'utilisation montrent les différentes interactions entre les utilisateurs et le système, illustrant les principales fonctionnalités telles que la recherche de contenu, la lecture de vidéos, et la gestion des profils.

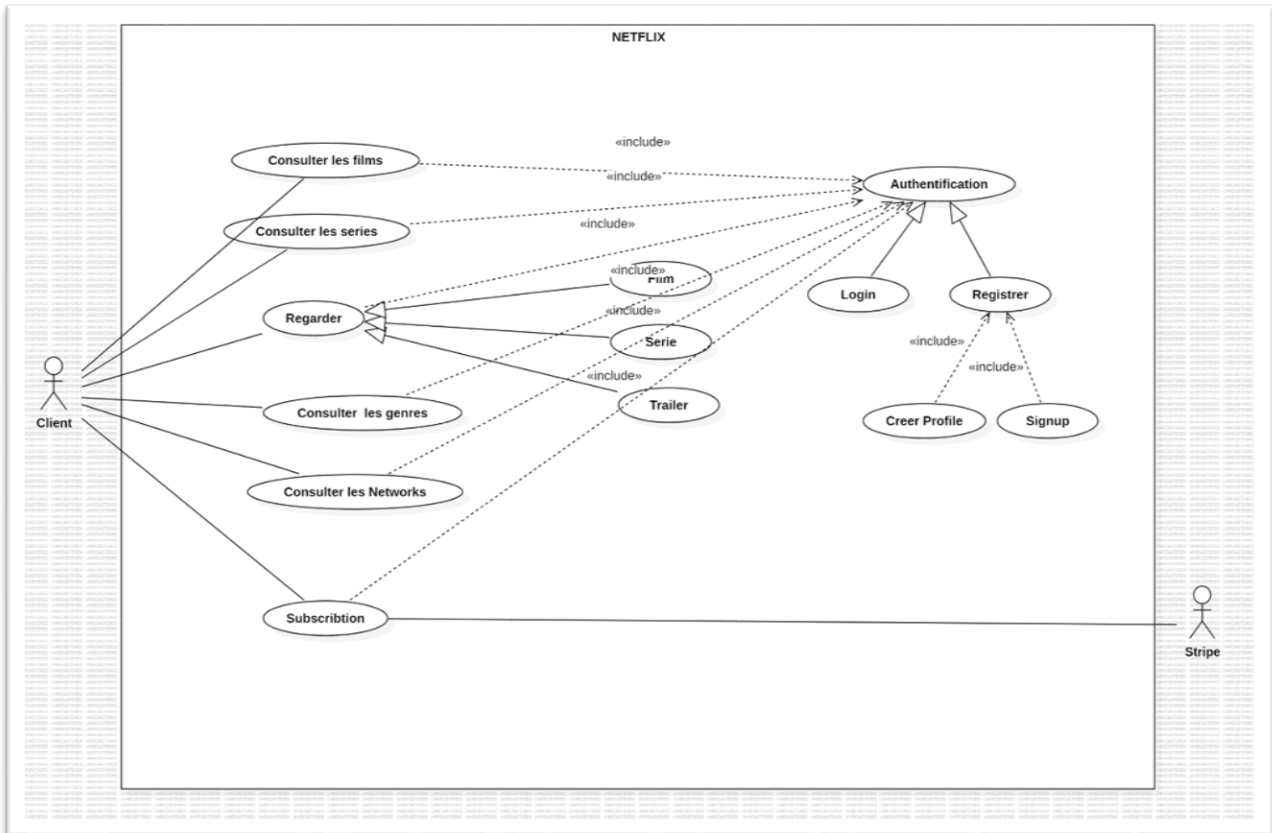


Figure 24 : Diagramme de cas d'utilisation

## Description:

### *S'authentifier (Authentication)*

Ce cas d'utilisation représente le processus par lequel un utilisateur accède à son compte sur la plateforme.

- **Register:** Inscription d'un nouvel utilisateur sur la plateforme.
- **Login:** Connexion d'un utilisateur existant.
- **Creer Profile:** Création d'un profil utilisateur, souvent nécessaire après l'inscription.

### *Consulter les films*

Ce cas d'utilisation permet à l'utilisateur de parcourir les films disponibles sur la plateforme.

### *Consulter les séries*

Permet aux utilisateurs de parcourir les séries disponibles sur la plateforme.

### *Consulter les genres*

Permet aux utilisateurs de filtrer les contenus par genre (comédie, drame, action, etc.).

### *Consulter les chaînes*

Permet aux utilisateurs de naviguer par chaînes ou collections de contenu spécifiques.

### *Regarder*

Ce cas d'utilisation inclut toutes les actions liées à la visualisation de contenu, comme regarder des bandes-annonces, des films, ou des épisodes de séries.

- **Trailer:** Regarder des bandes-annonces pour des films ou des séries.
- **Film:** Regarder des films complets.
- **Episode:** Regarder des épisodes de séries.

### *Subscribe*

Ce cas d'utilisation couvre le processus d'abonnement d'un utilisateur à la plateforme, y compris les interactions avec un service de paiement pour traiter les transactions financières.

## Diagrammes de classes

Le diagramme de classes est considéré comme le plus important de la modélisation orientée objet, il est le seul obligatoire lors d'une telle modélisation. Le diagramme de classes montre la structure interne du système. Il permet de fournir une représentation abstraite des objets du système qui vont interagir ensemble pour réaliser les cas d'utilisation. Il s'agit d'une vue statique car nous ne tenons pas compte du facteur temporel dans le comportement du système. Les principaux éléments de cette vue statique sont les classes et leurs relations. Ci-dessous le diagramme de classe qui correspond au projet :

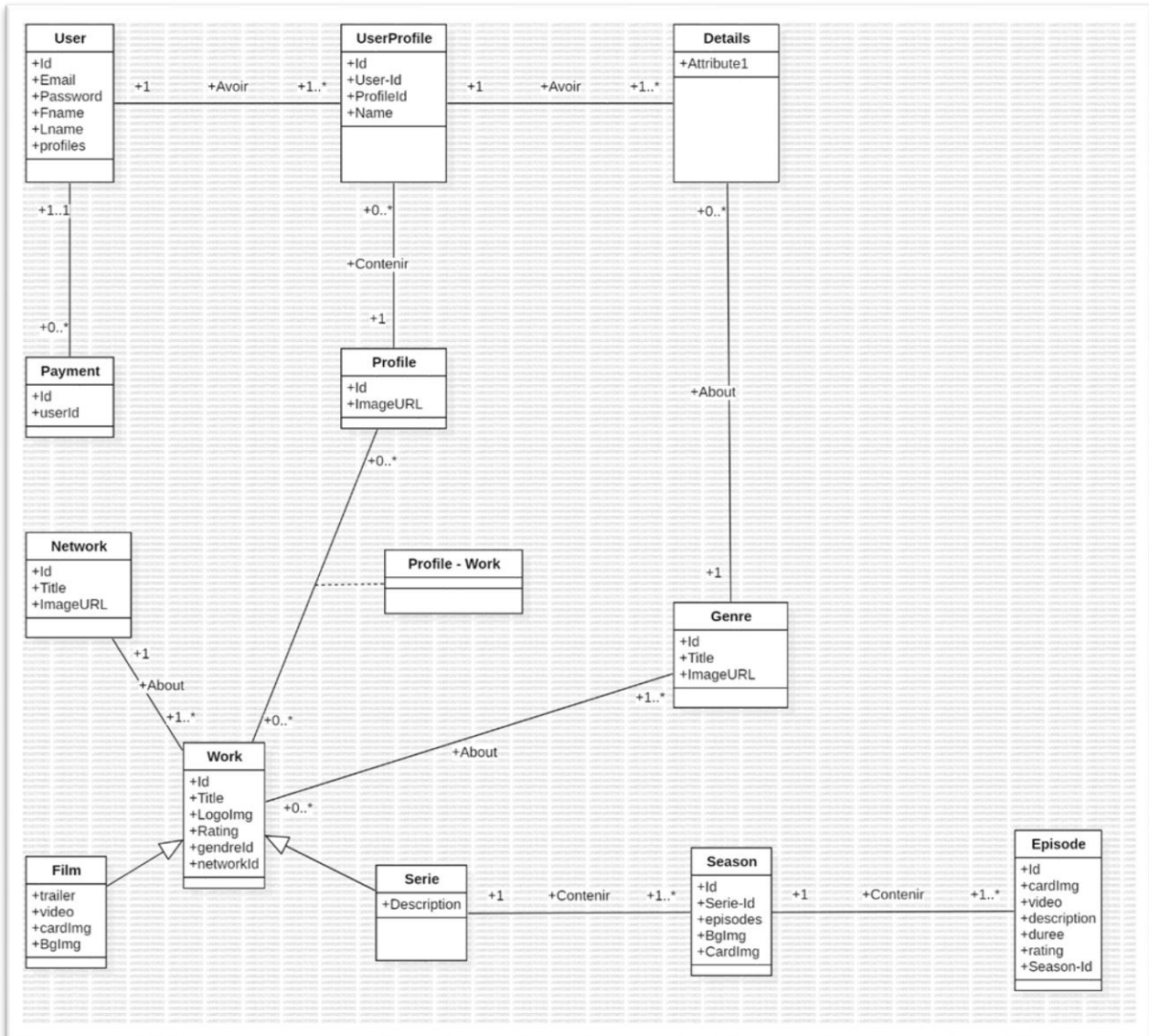


Figure 25 : Diagramme de classes

## Diagrammes de séquence

Les diagrammes de séquence illustrent les processus détaillés de certaines opérations, comme la lecture d'une vidéo ou l'ajout d'une recommandation.

## 1. Authentification des Utilisateurs

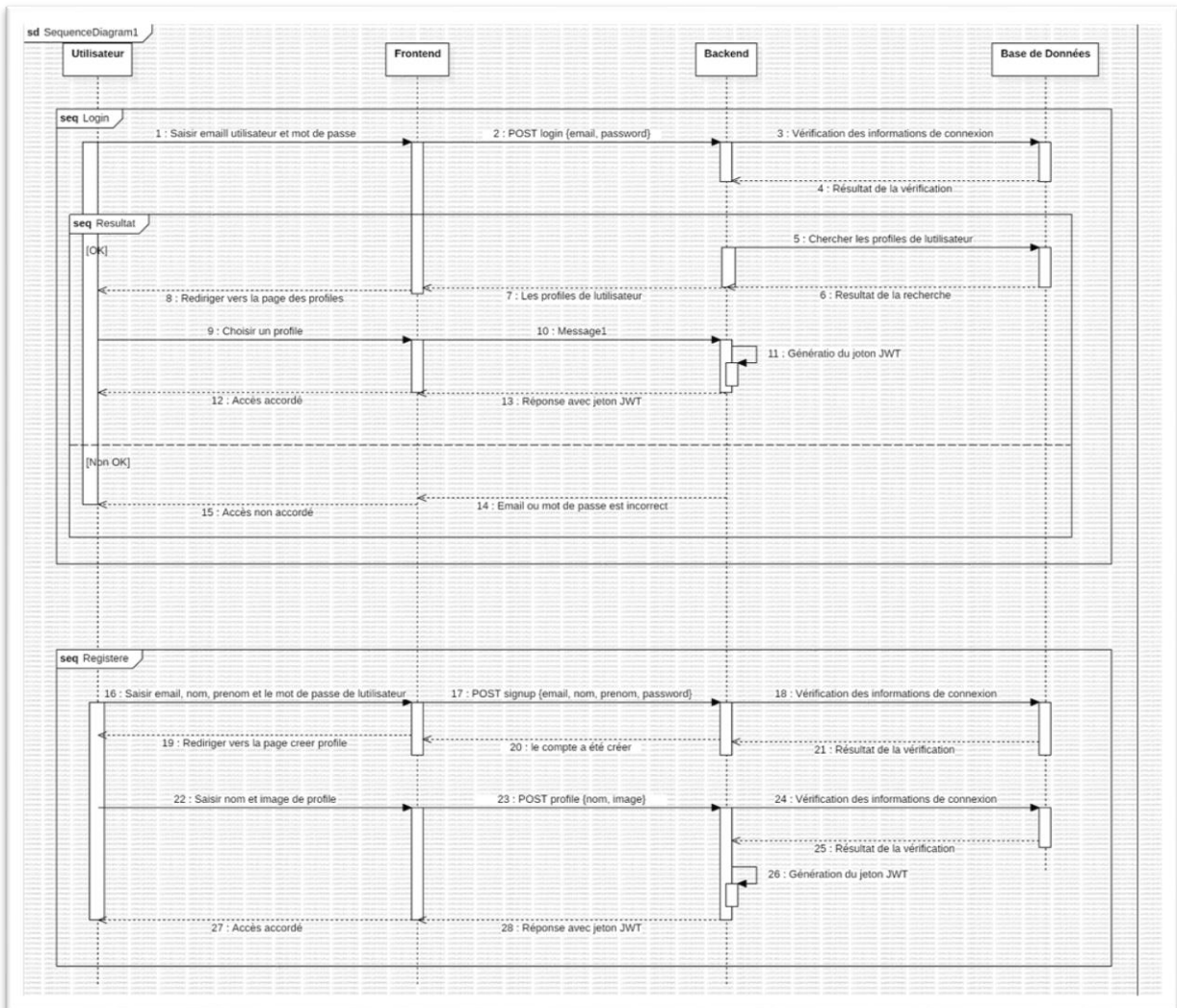


Figure 26: Diagramme de séquence " authentification "

### Description:

- L'utilisateur entre ses informations de connexion (email d'utilisateur et mot de passe) sur le frontend.
- Le frontend envoie ces informations au backend via une requête HTTP (POST).
- Le backend vérifie les informations dans la base de données.
- Si les informations sont correctes, le backend génère un jeton JWT.
- Le backend renvoie le jeton au frontend.
- Le frontend stocke le jeton pour les futures requêtes authentifiées.



## 2. le Paiement de l'Abonnement

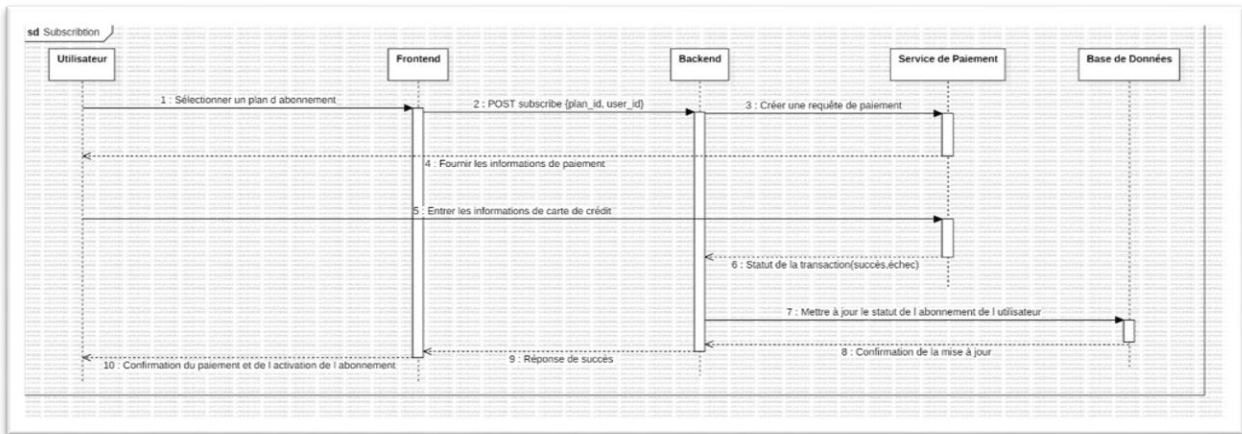


Figure 27: Diagramme de séquence " Paiement de l'abonnement "

### Description :

- L'utilisateur sélectionne un plan d'abonnement sur le frontend.
- Le frontend envoie les informations du plan d'abonnement au backend.
- Le backend prépare une requête de paiement et la transmet au service de paiement tiers.
- Le service de paiement demande à l'utilisateur de fournir les informations de carte de crédit.
- L'utilisateur fournit les informations de paiement.
- Le service de paiement traite le paiement et renvoie le statut de la transaction au backend.
- Le backend met à jour le statut de l'abonnement de l'utilisateur dans la base de données.
- La base de données confirme la mise à jour.
- Le backend envoie une confirmation de succès au frontend.
- Le frontend affiche la confirmation du paiement et de l'activation de l'abonnement à l'utilisateur.

## 3. la Navigation et la Lecture de Vidéo

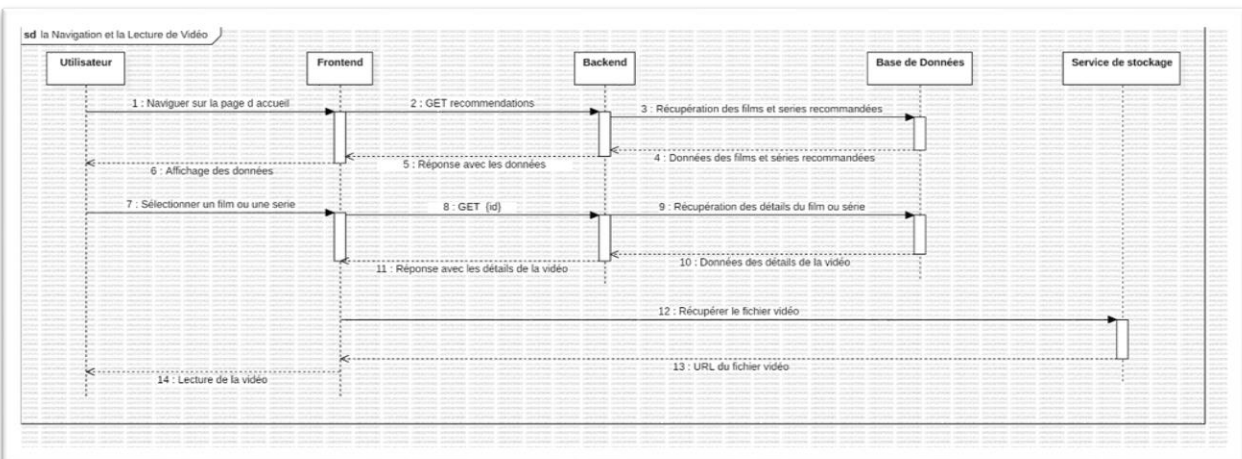


Figure 28: Diagramme de séquence " Navigation et lecture de vidéo "

### Description :

- L'utilisateur navigue sur la page d'accueil de l'application.
- Le frontend demande la liste des vidéos recommandées au backend.
- Le backend récupère les données de la base de données.
- Le backend renvoie les données au frontend.
- L'utilisateur sélectionne une vidéo à lire.
- Le frontend demande les détails de la vidéo et l'URL du fichier média au backend.
- Le backend récupère les détails de la vidéo et l'URL de stockage.
- Le backend envoie les informations au frontend.
- Le frontend charge et lit la vidéo.

### Conclusion

Le processus d'analyse et de conception a permis de définir clairement les besoins des utilisateurs, de concevoir une architecture robuste, et de spécifier les fonctionnalités et les technologies nécessaires. Les prochaines étapes du projet incluront le développement et l'implémentation de ces conceptions, en suivant le plan établi pour assurer une livraison efficace du système.



---

## CHAP4 : Conception de la Base de Données

---

## Introduction

La conception de la base de données est essentielle pour garantir la performance, la scalabilité et l'intégrité des données. Ce projet utilise une combinaison de Cassandra, MySQL et Redis pour tirer parti des points forts de chaque technologie.

### I. Choix de la Base de Données

Pour le projet de clonage de Netflix, nous avons choisi d'utiliser trois types de bases de données pour répondre aux besoins spécifiques du système :

Cassandra	Base de données NoSQL distribuée, adaptée pour des applications nécessitant une haute disponibilité et une scalabilité horizontale. Elle est particulièrement efficace pour gérer des volumes de données importants avec des exigences élevées en termes de lecture et d'écriture.
MySQL	Système de gestion de base de données relationnelle (SGBDR), idéal pour la gestion des transactions et l'intégrité des données. MySQL est utilisé pour stocker des informations critiques nécessitant une gestion transactionnelle rigoureuse.
Redis	Base de données en mémoire clé-valeur, connue pour sa rapidité et son efficacité en tant que cache. Redis est utilisé pour le stockage temporaire de données fréquemment accédées et la gestion des sessions utilisateurs.

Tableau 13: les types de bases de données.

Ces choix permettent de combiner les avantages de chaque technologie, optimisant ainsi la performance globale du système.

## II. Optimisation

### Optimisation de Cassandra

- **Partitionnement** : Les données sont partitionnées pour répartir uniformément la charge sur les différents nœuds du cluster.
- **Compaction** : Utilisation de la compaction pour fusionner les fichiers SSTable et améliorer les performances de lecture.

### Optimisation de MySQL

- **Indexation** : Création d'index sur les colonnes fréquemment utilisées dans les requêtes pour accélérer les opérations de recherche.
- **Optimisation des Requêtes** : Écriture de requêtes SQL optimisées et gestion des transactions pour éviter les blocages et améliorer la performance.

### Optimisation de Redis

- **Politique d'Éviction** : Configuration des politiques d'éviction (comme LRU - Least Recently Used) pour gérer efficacement la mémoire.
- **Options de Persistance** : Configuration des options de persistance (RDB ou AOF) pour sauvegarder les données en cas de redémarrage.

## III. Collecte des données

La collecte des données est une étape cruciale pour alimenter les bases de données du projet. Nous allons utiliser le web scraping pour extraire des données telles que des textes, des images et des vidéos à partir de sites web. Le scraping est réalisé avec Python, utilisant des bibliothèques spécialisées pour récupérer et traiter les données nécessaires.

### Introduction au Web Scraping

Le web scraping est une technique utilisée pour extraire des informations de sites web. En utilisant Python, nous pouvons automatiser la collecte de données et les stocker dans les bases de données appropriées. Les bibliothèques couramment utilisées pour le scraping en Python incluent BeautifulSoup, Scrapy, et Selenium.

## Stockage des Données Collectées

Après la collecte, les données doivent être stockées dans les bases de données appropriées :

- **Textes** : Stockés dans Cassandra selon la structure et la nécessité de la donnée.
- **Images et Vidéos** : Les fichiers sont stockés dans un service de stockage de fichiers (AWS S3).

## Conclusion

La conception des bases de données pour ce projet combine Cassandra, MySQL et Redis pour tirer parti des avantages spécifiques de chaque technologie. Cassandra est utilisée pour sa scalabilité, MySQL pour la gestion des transactions et l'intégrité des données, et Redis pour sa rapidité et son efficacité en tant que cache.

---

## CHAP5 : Mise en œuvre et Réalisation

---

## Introduction

Le développement frontend se concentre sur la création de l'interface utilisateur et l'expérience utilisateur. Pour le clonage de Netflix, nous avons utilisé des technologies modernes pour construire une interface réactive et attrayante.

### I. Développement Backend

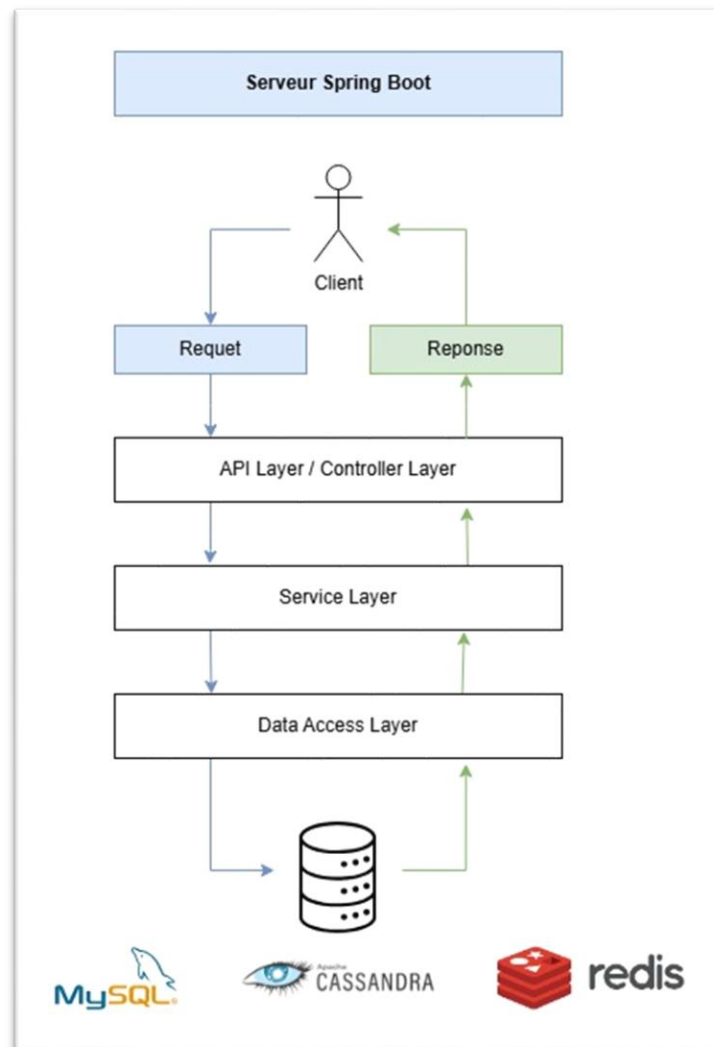


Figure 29: Architecture du backend

## Technologies et Outils Utilisés

Spring Boot	Spring Boot offre une méthode plus simple et plus rapide pour installer, configurer et exécuter des applications.
Java	Langage de programmation qui alimente les applications, les systèmes d'exploitation smartphone, les logiciels d'entreprise et de nombreux programmes connus.
JPA	JPA est utilisé pour abstraire les opérations de base de données, permettant aux développeurs d'interagir avec la base de données à l'aide d'objets Java plutôt que d'écrire directement des requêtes SQL.
Stripe	Stripe est une plateforme de traitement des paiements en ligne qui permet aux entreprises d'accepter des paiements sur Internet. Elle fournit une suite d'API et d'outils qui permettent aux entreprises de gérer les paiements, la facturation, les abonnements, etc.

Tableau 14: Technologies utilisées dans backend.

## Architecture Backend

### Conception des Endpoints API

## II. Développement Frontend

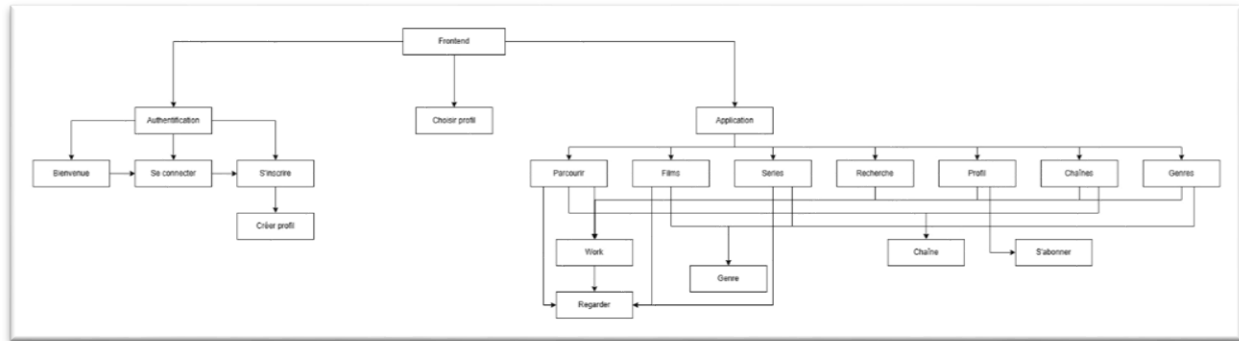


Figure 30: Structure du frontend

### Technologies Utilisées

React	Utilisé pour construire l'interface utilisateur de votre application.
Redux	Une bibliothèque de gestion d'état qui fournit un moyen prévisible de stocker et de mettre à jour l'état global de l'application. En utilisant Redux, on va stocker toutes les données dont on a besoin dans un seul « store », qui peut être partagé entre tous les composants de l'application.
React Router	Une bibliothèque permet de gérer les routes de l'application afin de créer des liens et de naviguer entre différentes pages.
Axios	Une bibliothèque JavaScript populaire, offre un ensemble complet de fonctionnalités pour effectuer des requêtes HTTP.
React icons	Une bibliothèque open-source qui offre une vaste collection d'icônes prêtes à l'emploi pour les développeurs React.

Tableau 15: Technologies utilisées dans frontend.



## Développement des Fonctionnalités Principales

### *Authentification*

Lorsque l'utilisateur visite le site, la page d'accueil apparaît si l'utilisateur n'est pas identifié.

Il peut alors choisir de se connecter à son compte, ou d'en créer un nouveau.

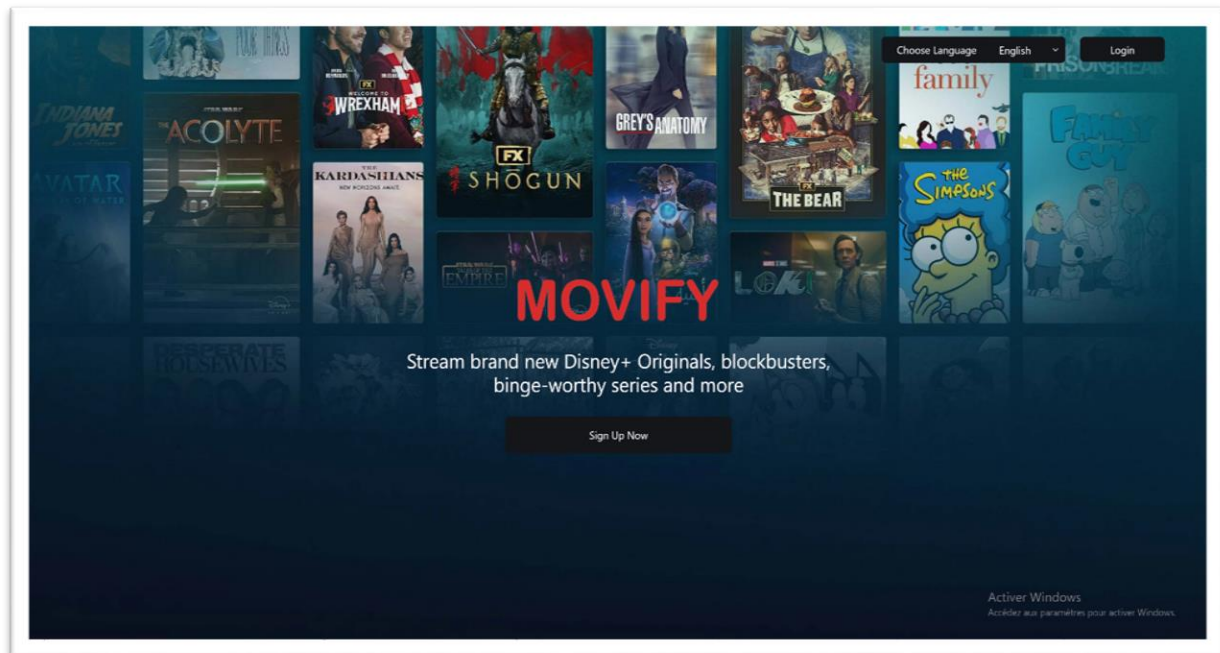


Figure 31: Page d'accueil.

### Création du compte

Le processus d'inscription et de création de profil se déroule en plusieurs étapes, chacune étant cruciale pour garantir une expérience utilisateur fluide et sécurisée.

#### 1. Formulaire d'Inscription :

L'utilisateur commence par remplir un formulaire d'inscription comportant les champs suivants :

- **Email** : Adresse email de l'utilisateur, utilisée pour l'identification et la communication.
- **Prénom** : Prénom de l'utilisateur.
- **Nom de Famille** : Nom de famille de l'utilisateur.

- **Mot de Passe** : Mot de passe choisi par l'utilisateur, qui doit respecter des critères de sécurité définis (longueur minimale, complexité, etc.).

Une fois le formulaire rempli, les données sont envoyées au serveur via une requête POST. La validation des champs est effectuée côté client avant l'envoi pour garantir que toutes les informations requises sont présentes et conformes.

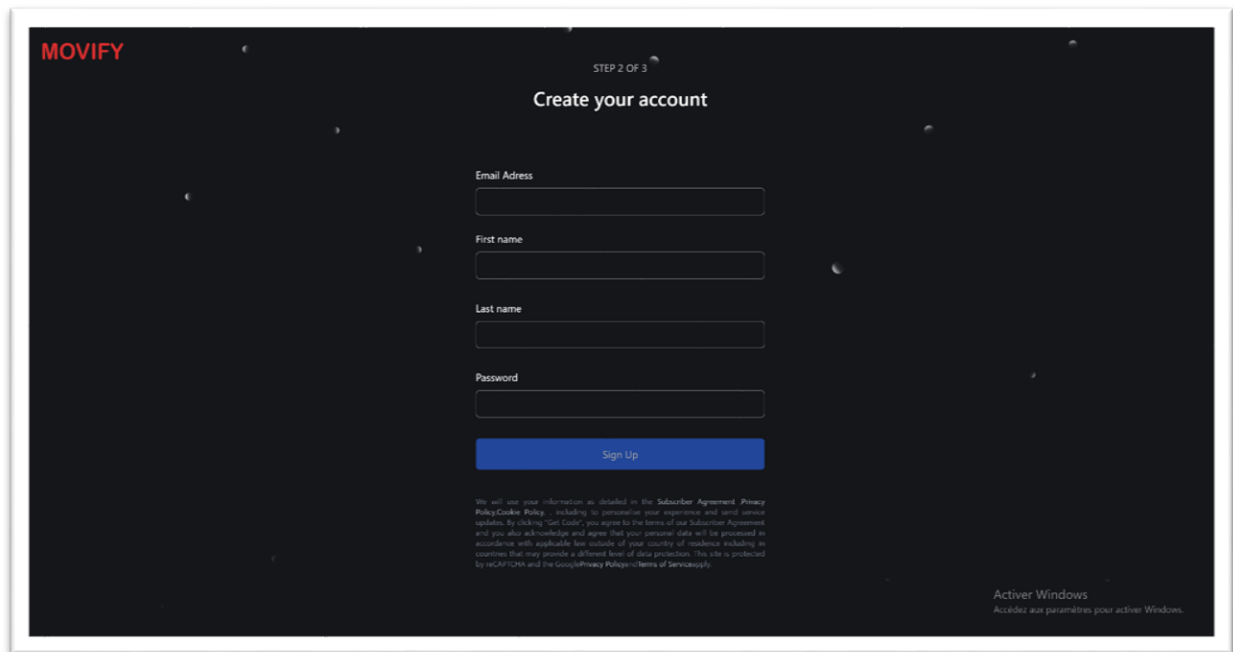


Figure 32: Page d'inscription.

## 2. Formulaire d'Inscription :

Après la création du compte, l'utilisateur est redirigé vers une page de création de profil. Cette étape permet à l'utilisateur de :

- **Choisir une Image de Profil** : L'utilisateur peut choisir parmi des images prédéfinies.
- **Compléter les Informations de Profil** : L'utilisateur remplit un formulaire pour fournir des informations supplémentaires, telles que le nom du profil.

Les données du profil, y compris l'image et le nom, sont envoyées au serveur via une requête POST.

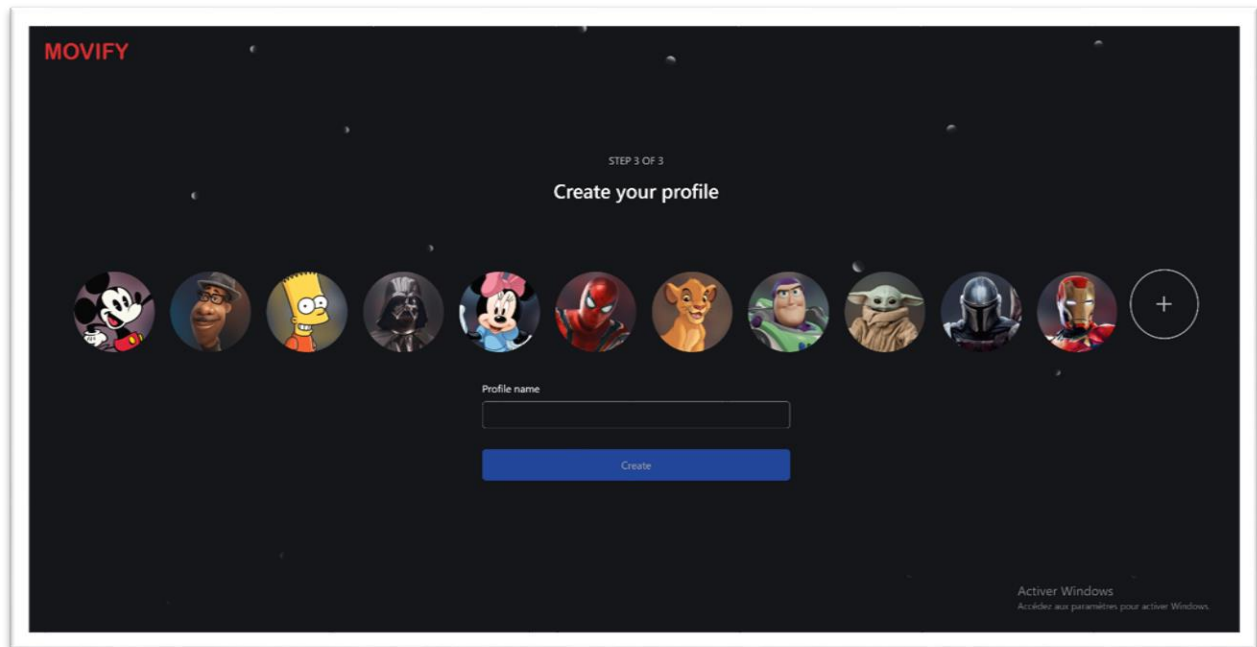


Figure 33: Page de création du profil.

## Connexion

Le processus de connexion permet à un utilisateur de se connecter à son compte en utilisant ses identifiants. Voici les étapes détaillées pour la connexion:

### 1. Formulaire de Connexion :

L'utilisateur accède à un formulaire de connexion comportant les champs suivants :

- **Email** : Adresse email associée au compte utilisateur. Ce champ est utilisé pour identifier l'utilisateur.
- **Mot de Passe** : Mot de passe correspondant à l'adresse email.

Lors de la soumission du formulaire, les données sont envoyées au serveur via une requête POST. Une validation est effectuée côté client pour vérifier que les champs ne sont pas vides avant l'envoi.

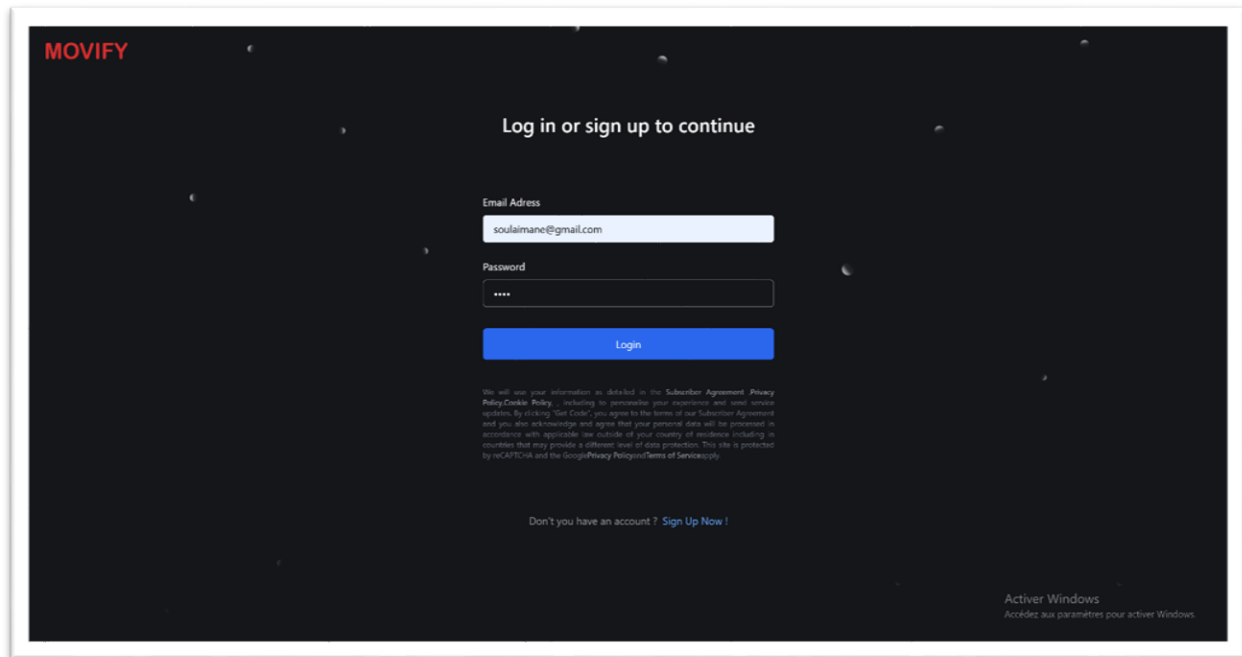


Figure 34: Page de Connexion (Login).

### *Affichage des Films et Séries*

Décrivez comment les films sont récupérés et affichés. Par exemple, l'utilisation d'un composant MovieList pour afficher une liste de films en récupérant les données depuis l'API.

### *Paielement et abonnement*

Si votre application permet aux utilisateurs de créer une liste de favoris ou un panier, détaillez comment cette fonctionnalité est implémentée et gérée.

### III. Interaction Frontend-Backend

L'interaction entre le frontend et le backend est un processus clé dans notre application. Le frontend, envoie des requêtes HTTP via Axios à l'API REST exposée par le backend.

#### Exemple d'Interaction Utilisateur

Lorsqu'un utilisateur se connecte, une requête **POST** est envoyée au backend avec les informations d'identification de l'utilisateur (l'adresse-mail et le mot de passe).

Le backend authentifie ces informations à l'aide de **Spring Security**, génère un token JWT, et le renvoie au frontend.

Ce token est ensuite stocké dans le navigateur (dans localStorage) pour des requêtes ultérieures.

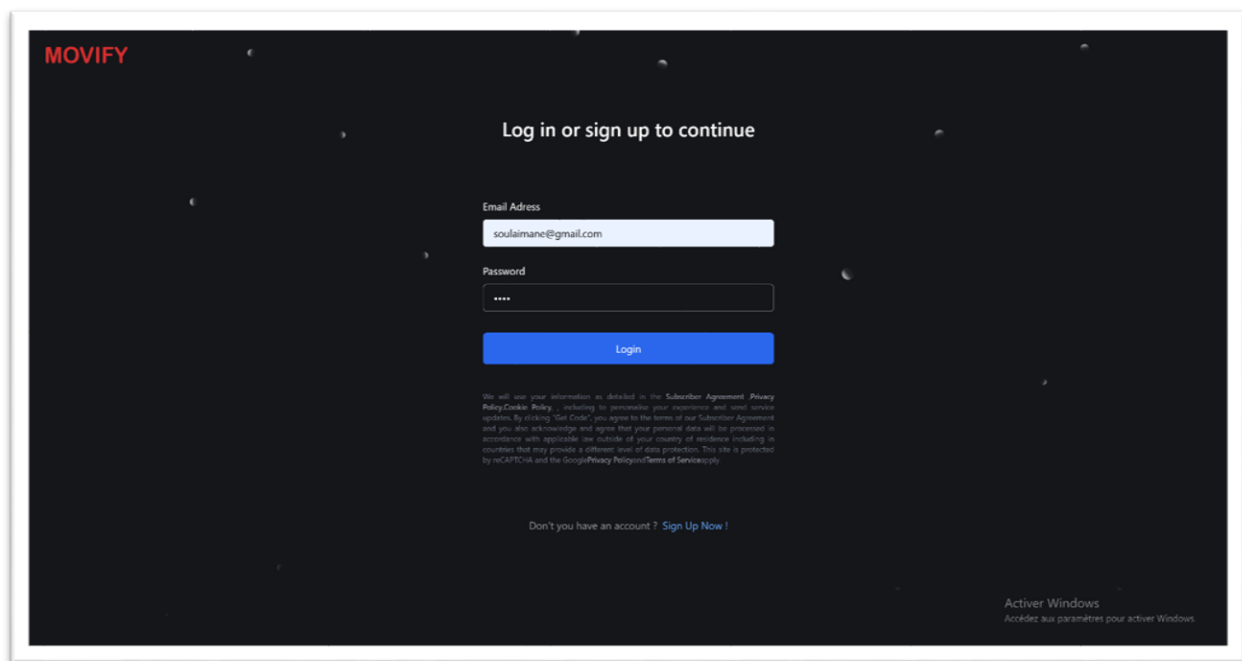


Figure 35: Page de connexion

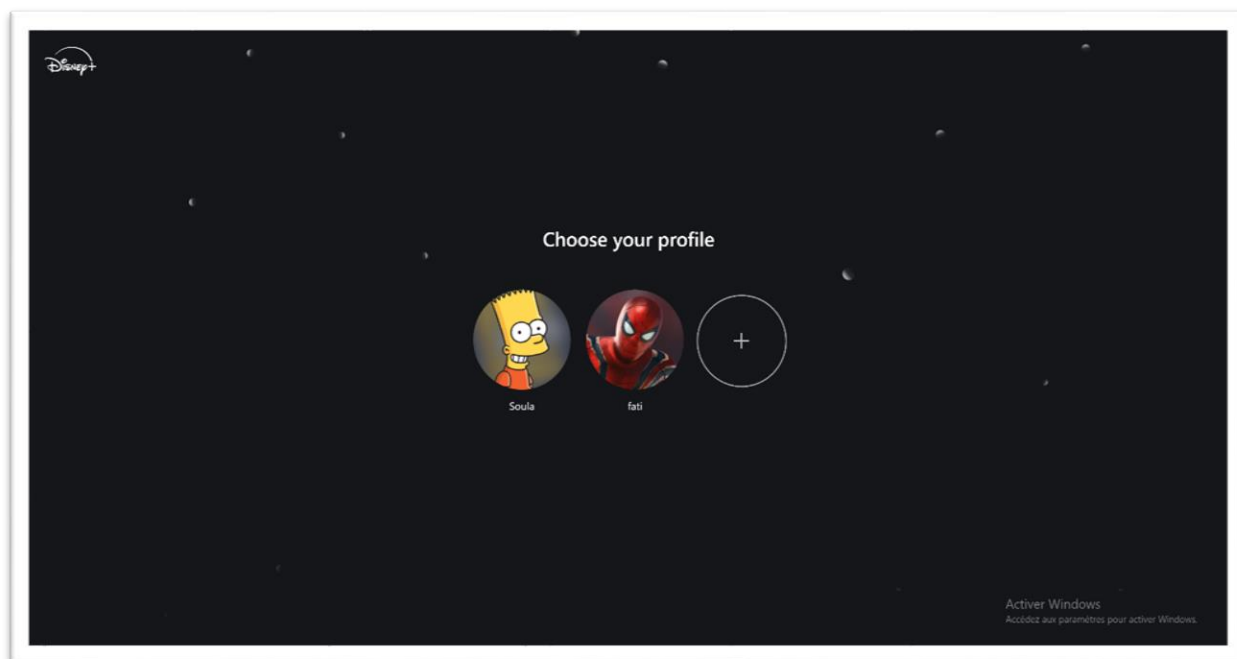


Figure 36: Page de choisir le profil

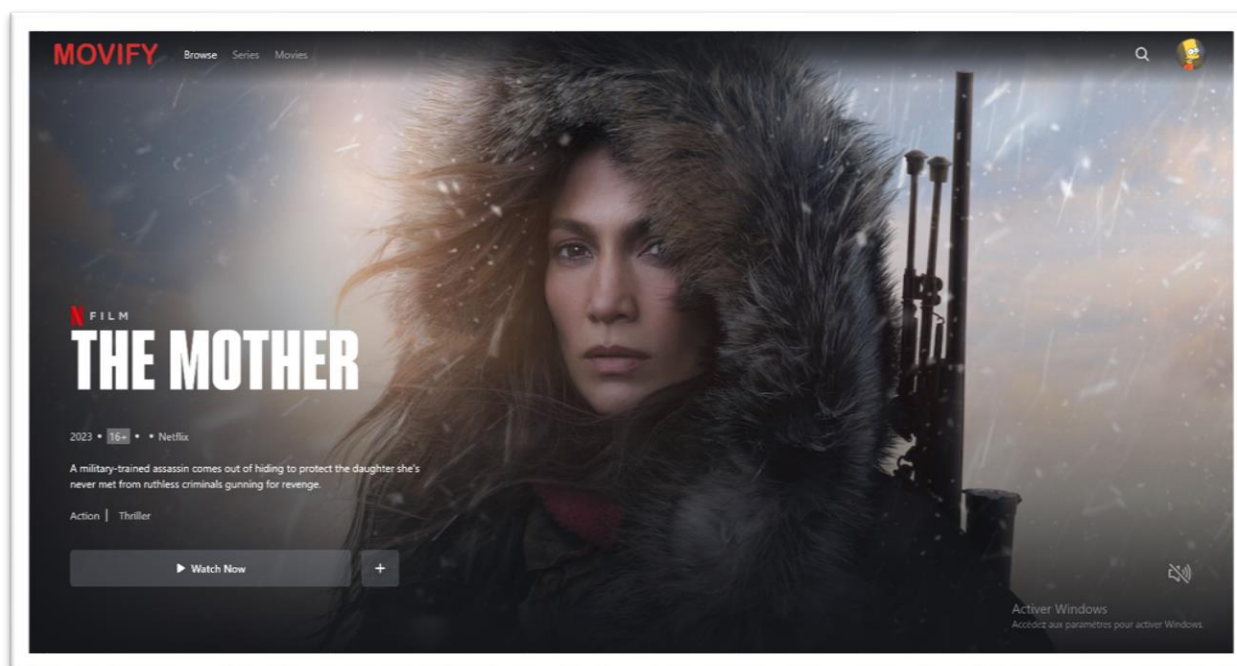


Figure 37: Page d'accueil

## Récupération des Données

Lorsqu'un utilisateur demande des informations sur un film, une requête GET est envoyée par le frontend.

Le backend interroge la base de données Cassandra, récupère les informations du film et les renvoie au frontend sous forme de JSON, où elles sont affichées dynamiquement.

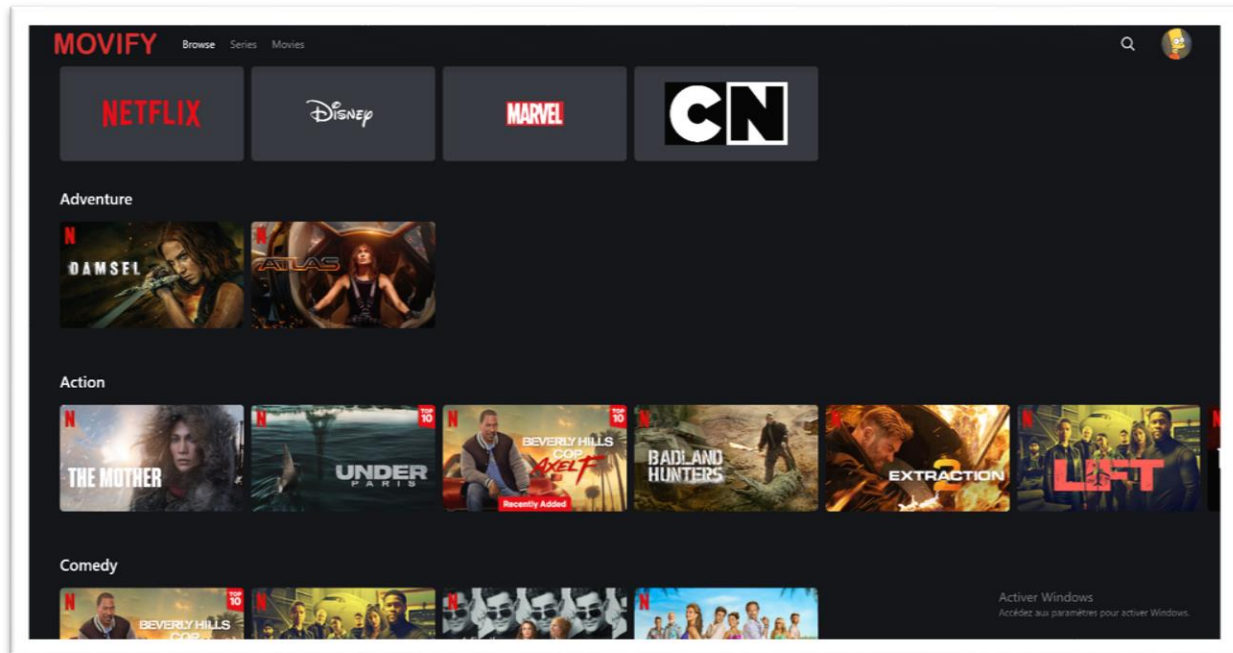


Figure 38: Choisir ( film / serie )



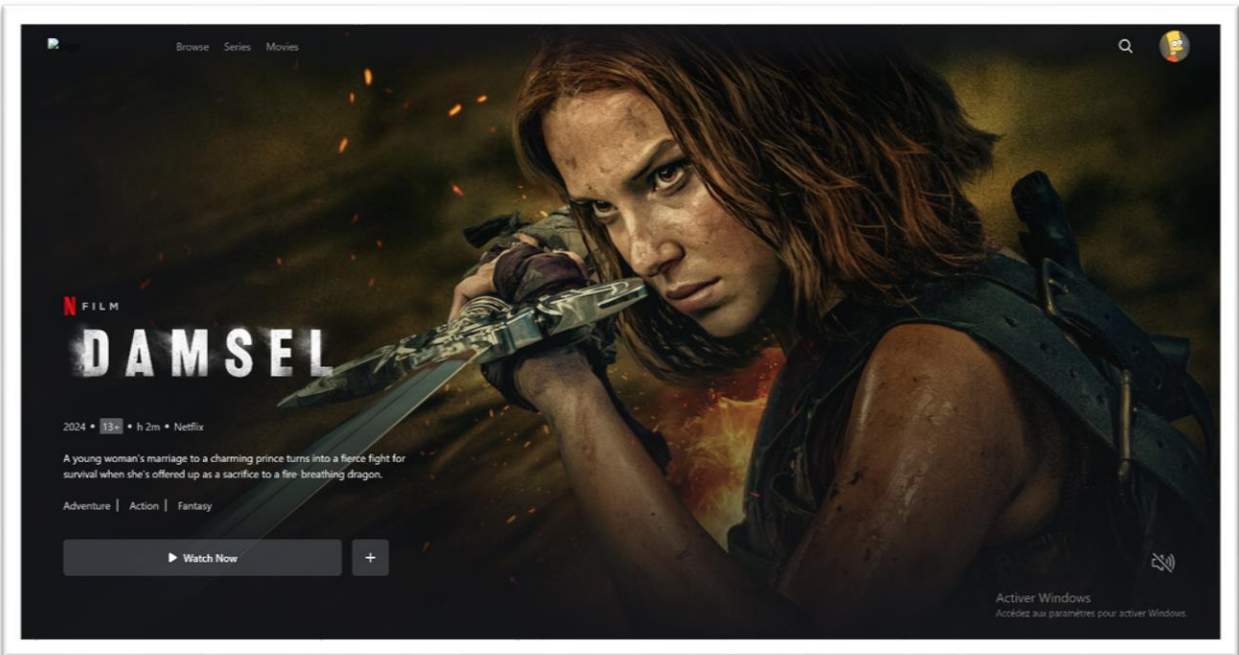


Figure 39: Page (film / serie)

## Mise à jour en Temps Réel

Lorsqu'un utilisateur ajoute un film à sa liste de favoris, une requête POST est envoyée au backend, qui met à jour la base de données.

Le frontend reflète instantanément ces changements à l'utilisateur grâce à une mise à jour de l'état de l'application.



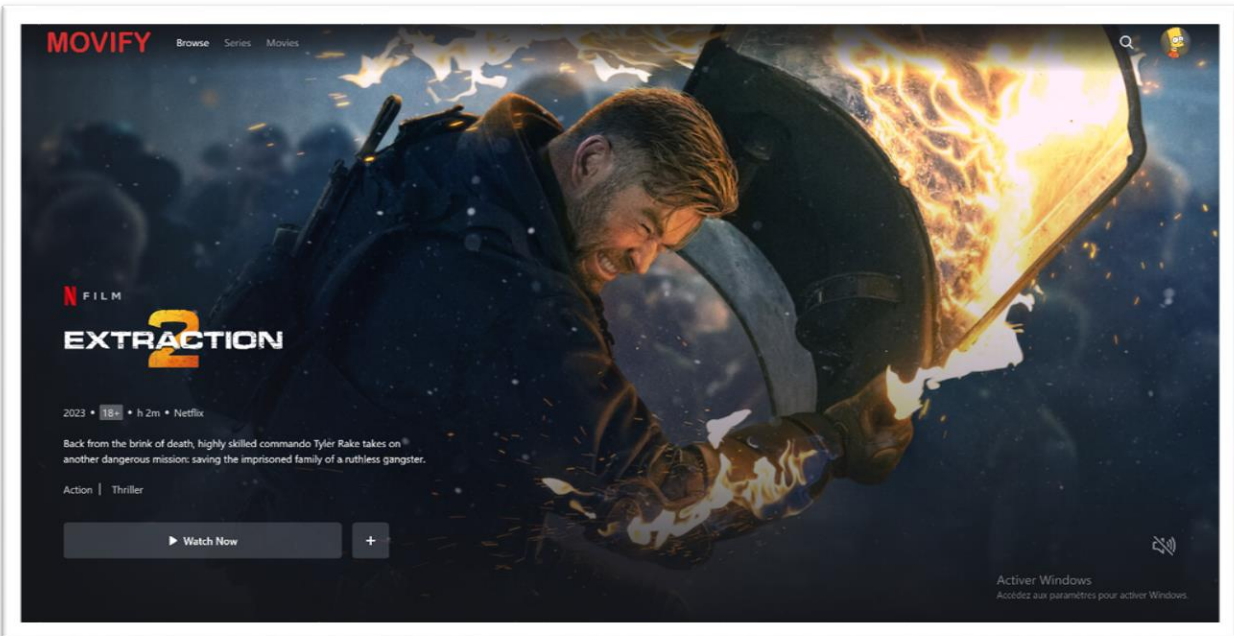


Figure 40: Page d'un film

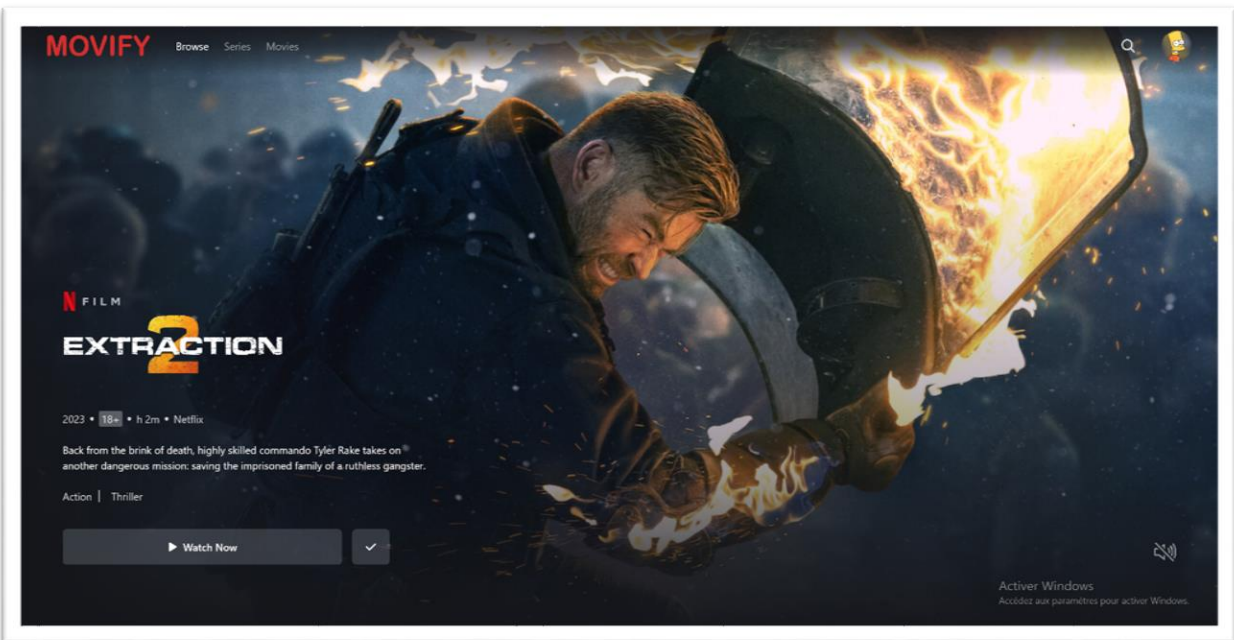


Figure 41: Page d'un film apres ajouter aux favoris

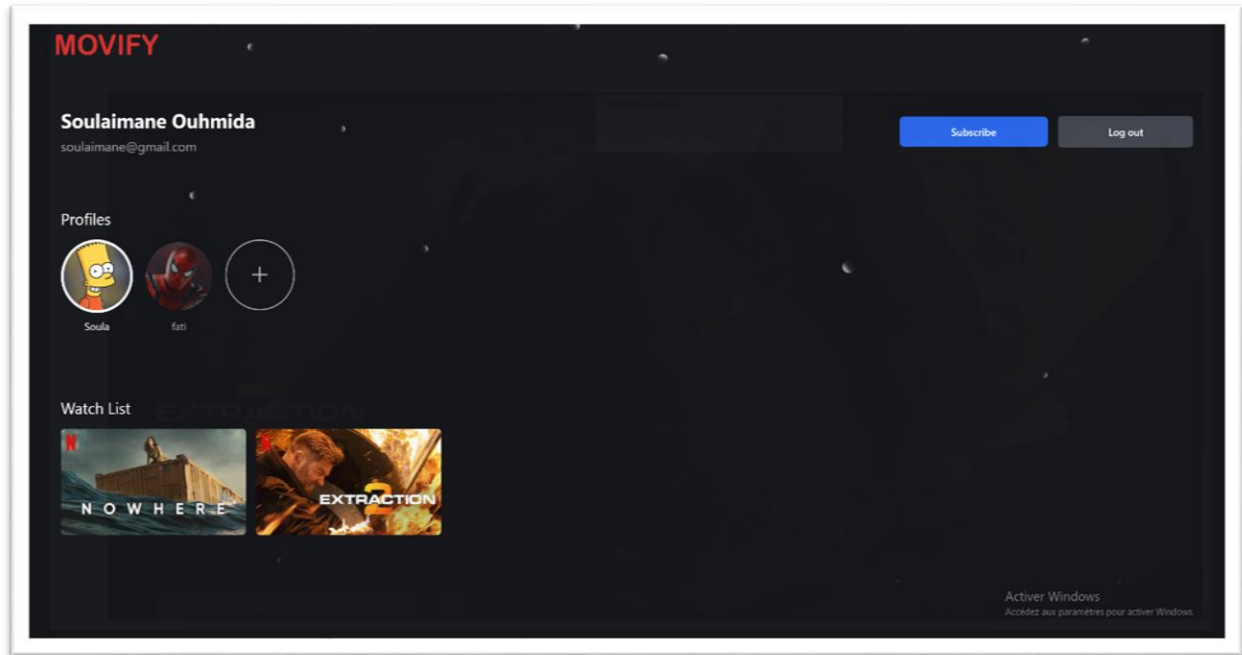


Figure 42: Page de favoris

## Conclusion

Le développement frontend et backend pour le projet de clonage de Netflix combine des technologies modernes pour offrir une interface utilisateur fluide et une gestion efficace des données. Le frontend utilise React pour créer des interfaces dynamiques, tandis que le backend avec Node.js et Express (ou Django) gère les fonctionnalités serveur et l'accès aux données.

---

## CHAP 6: Déploiement Cloud

---

## Introduction

Le déploiement sur le cloud est une étape cruciale pour garantir la disponibilité, l'évolutivité, et la résilience de l'application. Dans ce projet, nous avons choisi Amazon Web Services (AWS) pour déployer notre application en raison de sa robustesse, de sa flexibilité et de ses services complets.

Ce chapitre détaille le processus de déploiement sur AWS, les services utilisés, ainsi que les considérations en matière de sécurité et de surveillance.

## I. Architecture Cloud

### Présentation Générale

L'architecture est conçue pour offrir une haute disponibilité et une scalabilité horizontale. Les principaux composants sont :

Frontend (React)	Interface utilisateur accessible via un navigateur.
Backend (Spring Boot)	Gère la logique métier, les API REST, et les opérations de traitement.
Bases de Données	Gestion des données structurées (MySQL), semi-structurées (Cassandra), et du cache (Redis).

Tableau 16: Les principaux composants cloud

### Services AWS Utilisés

Amazon S3	Utilisé pour héberger les fichiers statiques de l'application React, comme les fichiers HTML, CSS, et JavaScript. S3 permet une distribution mondiale avec des options de sécurité avancées.
Amazon CloudFront	Service de distribution de contenu qui permet de mettre en cache les fichiers statiques à des emplacements

	géographiquement dispersés pour améliorer les temps de chargement.
Amazon EC2	Les instances EC2 sont utilisées pour héberger les applications backend développées avec Spring Boot. EC2 offre une flexibilité pour choisir les types d'instances en fonction des besoins de performance et de coût.
Amazon RDS	Utilisé pour déployer la base de données MySQL. RDS gère les tâches administratives comme les sauvegardes, le scaling et les mises à jour logicielles.
Amazon Keyspaces (for Apache Cassandra)	Service entièrement managé pour Cassandra, utilisé pour les données semi-structurées et les requêtes à haute performance. Keyspaces est compatible avec les API Cassandra, facilitant la migration et la gestion.
Amazon ElastiCache	Utilisé pour déployer Redis, un système de cache en mémoire, pour accélérer l'accès aux données fréquemment demandées et réduire la charge sur les bases de données.
Amazon Elastic Load Balancing (ELB)	Assure la répartition de la charge sur plusieurs instances EC2, augmentant la tolérance aux pannes et l'évolutivité.
Amazon Route 53	Service DNS pour gérer les noms de domaine et diriger le trafic vers les ressources AWS.
AWS IAM (Identity and Access Management)	Utilisé pour gérer les permissions et l'accès sécurisé aux ressources AWS.
Amazon CloudWatch	Pour la surveillance des ressources AWS et des applications, avec des alertes et des dashboards pour suivre les performances.

Tableau 17: Les services AWS Utilisés

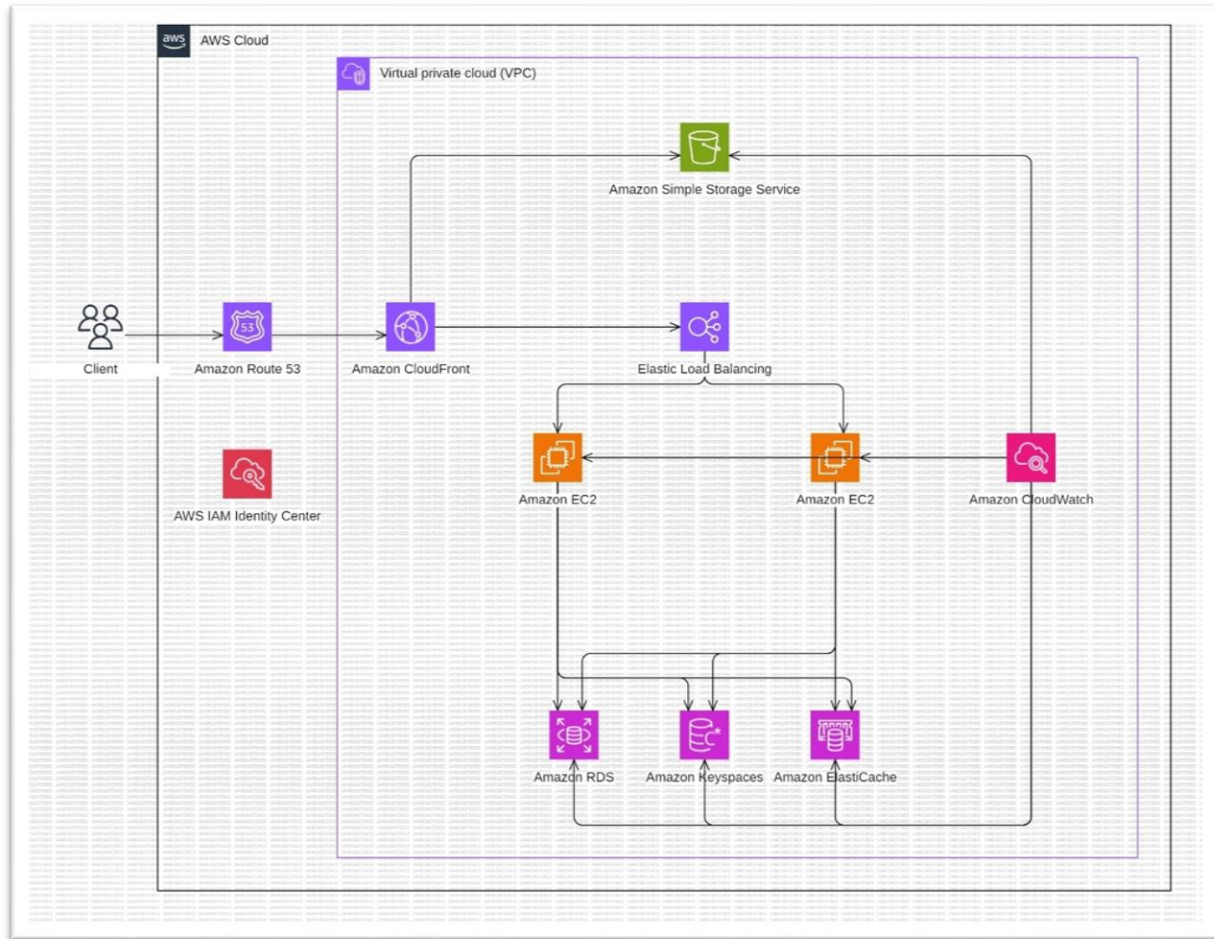


Figure 43: Diagramme du cloud

## II. Processus de Déploiement

### Déploiement du Frontend (React)

Etape	Description
Build de l'application	Utilisation de npm run build pour créer une version optimisée de l'application React.
Hébergement sur S3	Les fichiers de build sont téléchargés sur un bucket S3 configuré pour l'hébergement de sites web statiques.

Distribution via CloudFront	Configuration de CloudFront pour distribuer le contenu depuis le bucket S3, en utilisant des points de présence globaux pour une performance optimale
-----------------------------	---

Tableau 18: Les étapes de déploiement du Frontend (React)

## Déploiement du Backend (Spring Boot)

Etape	Description
Build et Packaging	Le backend est packagé en utilisant des outils comme Maven ou Gradle pour créer des artefacts (fichiers .jar ou .war).
Déploiement sur EC2	Les artefacts sont déployés sur des instances EC2 configurées avec les paramètres nécessaires pour exécuter des applications Spring Boot.
Mise en place de l'ELB	Utilisation d'Elastic Load Balancer pour répartir la charge du trafic réseau entrant entre plusieurs instances EC2, assurant ainsi l'évolutivité et la tolérance aux pannes.

Tableau 19: Les étapes de déploiement du Backend (Spring Boot)

## Configuration des Bases de Données

Etape	Description
MySQL sur RDS	Mise en place de la base de données MySQL avec Amazon RDS, incluant la configuration des paramètres de sécurité, des sauvegardes automatiques et du Multi-AZ pour la haute disponibilité.
Cassandra sur Amazon Keyspaces	Déploiement de la base de données NoSQL Cassandra sur Amazon Keyspaces, avec une attention particulière à la mise



	en place des schémas et des stratégies de partitionnement pour des performances optimales.
Redis sur ElastiCache	Déploiement de Redis pour le caching des données et la gestion des sessions. ElastiCache gère l'évolutivité, la sécurité et la surveillance.

Tableau 20: Les étapes de la configuration des Bases de Données

### III. Sécurité et Conformité

IAM et Sécurité des Instances	Configuration des rôles IAM pour restreindre l'accès aux ressources critiques, et mise en place de groupes de sécurité pour filtrer le trafic réseau vers et depuis les instances EC2.
Chiffrement	Utilisation de SSL/TLS pour sécuriser les communications entre les services et de KMS pour le chiffrement des données au repos.
Surveillance et Audit	CloudWatch est utilisé pour surveiller les accès et les activités, garantissant la conformité et facilitant les audits.

Tableau 21: Sécurité et Conformité

## Conclusion

Le déploiement sur le cloud pour notre projet de clonage de Netflix permet de bénéficier de la flexibilité, de la scalabilité et des services gérés offerts par les principaux fournisseurs de cloud. Le processus de déploiement, les stratégies de scalabilité et de gestion des performances sont essentielles pour assurer que l'application fonctionne efficacement et est capable de répondre aux besoins des utilisateurs.



## Conclusion Générale et Perspectives

Le projet de développement d'un clone de Netflix a été une expérience enrichissante et formatrice, couvrant divers aspects du développement logiciel, de la conception à la mise en œuvre. Nous avons réussi à créer une application fonctionnelle intégrant une interface utilisateur réactive, un backend robuste, une base de données optimisée, et un déploiement sur le cloud. Les défis techniques rencontrés ont été surmontés grâce à une méthodologie Agile efficace et à l'utilisation d'outils de gestion de projet modernes. Ce projet a non seulement renforcé mes compétences techniques mais m'a également permis de mieux comprendre les exigences d'un environnement de développement collaboratif et professionnel.

### Améliorations Futures:

- **Nouvelles Fonctionnalités** : Intégrer des fonctionnalités avancées comme la recommandation de contenu basée sur l'IA, le téléchargement de vidéos pour une lecture hors ligne, et des options de personnalisation d'interface.
- **Optimisation de la Performance** : Améliorer les algorithmes de streaming pour réduire la latence et optimiser la qualité vidéo en fonction de la bande passante disponible.
- **Sécurité Accrue** : Renforcer les mesures de sécurité pour protéger les données des utilisateurs et prévenir les attaques potentielles.
- **Expansion des Capacités Cloud** : Explorer l'utilisation de services cloud supplémentaires pour améliorer la scalabilité et la résilience de l'application.

Ce projet a également ouvert de nouvelles perspectives professionnelles en me préparant à des rôles de développeur full-stack et d'architecte de solutions cloud. Les compétences acquises et les défis relevés constituent une base solide pour poursuivre des projets de plus grande envergure et pour contribuer efficacement à des équipes de développement dans des environnements dynamiques.

En somme, ce projet de fin d'études a été une opportunité précieuse pour appliquer des connaissances théoriques à un projet concret et complexe, tout en acquérant une expérience pratique des meilleures pratiques en matière de développement logiciel et de gestion de projet. Il marque une étape importante dans mon parcours professionnel et constitue une base solide pour mes futures contributions dans le domaine du développement web et des technologies de l'information.

## Annexe

## Références