

# Pandora

<b>Description du Projet.</b>	<b>2</b>
Pitch du projet.	2
Problématique.	2
Valeur Ajoutée.	2
Cible.	3
<b>Présentation technique.</b>	<b>3</b>
Architecture	3
Technologies utilisées	3
Problématiques techniques	4
<b>L'Équipe et son Organisation.</b>	<b>4</b>
Membres de l'Équipe et Compétences	4
Organisation du Projet :	5
- Découpage du Projet :	5
- Répartition des Tâches :	5
Compétences à développer.	5
- Formation Complémentaire :	5

# Description du Projet.

## Pitch du projet

Pandora est une application innovante de recommandations de films et séries TV personnalisées. En utilisant le machine learning et l'intelligence artificielle, Pandora analyse les préférences de chaque utilisateur pour leur proposer des suggestions adaptées à leurs goûts et intérêts.

## Problématique

Les utilisateurs sont souvent submergés par le vaste catalogue de films et séries disponibles sur différentes plateformes de streaming, rendant difficile la découverte de nouveaux contenus qui correspondent à leurs goûts spécifiques.

## Valeur Ajoutée

Contrairement à d'autres services de recommandations qui se basent uniquement sur les tendances générales ou les évaluations des utilisateurs, Pandora utilise des algorithmes avancés de machine learning pour offrir des suggestions ultra-personnalisées. Cela garantit que chaque utilisateur reçoit des recommandations qui correspondent précisément à ses préférences uniques.

## Cible

Pandora s'adresse à tous les cinéphiles et amateurs de séries TV qui utilisent des services de streaming comme Netflix, Amazon Prime Video, et Disney+. Notre cible

principale est constituée d'utilisateurs âgés de 18 à 45 ans, technophiles, et avides de découvrir de nouveaux contenus.

## Présentation technique.

### Architecture

- **Frontend** : Application mobile développée en React Native pour garantir une expérience utilisateur fluide et intuitive sur iOS et Android.
- **Backend** : API RESTful développée en Node.js avec Express.js pour la gestion des données et des utilisateurs.
- **Base de données** : MongoDB pour sa flexibilité et sa capacité à gérer des données non structurées et volumineuses.
- **Machine Learning** : Modèles de recommandation développés avec TensorFlow pour l'analyse des préférences et la génération des recommandations.

### Technologies utilisées

- **Frontend** : React Native, Redux (pour la gestion de l'état), et Axios (pour les requêtes API).
- **Backend** : Node.js, Express.js, MongoDB, et Mongoose (pour la gestion des schémas MongoDB).
- **Machine Learning** : TensorFlow pour le développement des modèles de recommandation.
- **Autres Outils** : Git (pour le contrôle de version), Jenkins (pour l'intégration continue), et Docker (pour la containerisation des applications).

## Problématiques techniques

- **Gestion des Données Massives** : Utilisation de MongoDB pour sa capacité à gérer efficacement de grandes quantités de données non structurées.
- **Personnalisation des Recommandations** : Utilisation de TensorFlow pour développer des algorithmes de machine learning capables de fournir des recommandations personnalisées précises.
- **Intégration avec des API de Streaming** : Intégration des API de services de streaming comme Netflix et Amazon Prime pour récupérer des données sur les films et séries.

## L'Équipe et son Organisation.

### Membres de l'Équipe et Compétences

- **london\_j (Chef de Projet et Développeur Full Stack)** : Expertise en développement frontend et backend, gestion de projet.
- **bouzia\_b (Développeuse Frontend)** : Spécialisé en React Native et UX/UI design.
- **elyazi\_i (Développeur Backend)** : Expertise en Node.js, Express.js, et MongoDB.
- **aubry\_c (Data Scientist)** : Spécialiste en machine learning et TensorFlow.
- **coskun\_h (DevOps)** : Expertise en intégration continue, déploiement, et gestion des environnements Docker.

### Organisation du Projet

- Découpage du Projet :
  - **Run 1** : Analyse des besoins et définition des spécifications.

- **Run 2** : Développement du frontend et du backend.
  - **Run 3** : Développement des modèles de machine learning.
  - **Run 4** : Intégration et tests.
  - **Run 5** : Déploiement et maintenance.
- 
- Répartition des Tâches :
    - **london\_j** : Coordination de l'équipe, développement backend.
    - **bouzia\_b** : Développement de l'interface utilisateur, expérience utilisateur.
    - **elyazi\_i** : Développement de l'API backend, gestion de la base de données.
    - **aubry\_c** : Développement et entraînement des modèles de machine learning.
    - **coskun\_h** : Mise en place de l'intégration continue, déploiement des applications.

## Compétences à développer

- Formation Complémentaire :
  - **london\_j** : Formation avancée en gestion de projet agile.
  - **bouzia\_b** : Formation en design d'interfaces utilisateur pour mobile.
  - **elyazi\_i** : Formation en optimisation de bases de données NoSQL.
  - **aubry\_c** : Formation en déploiement de modèles de machine learning en production.
  - **coskun\_h** : Formation en sécurité des applications web et mobiles.