# Movie Advisor

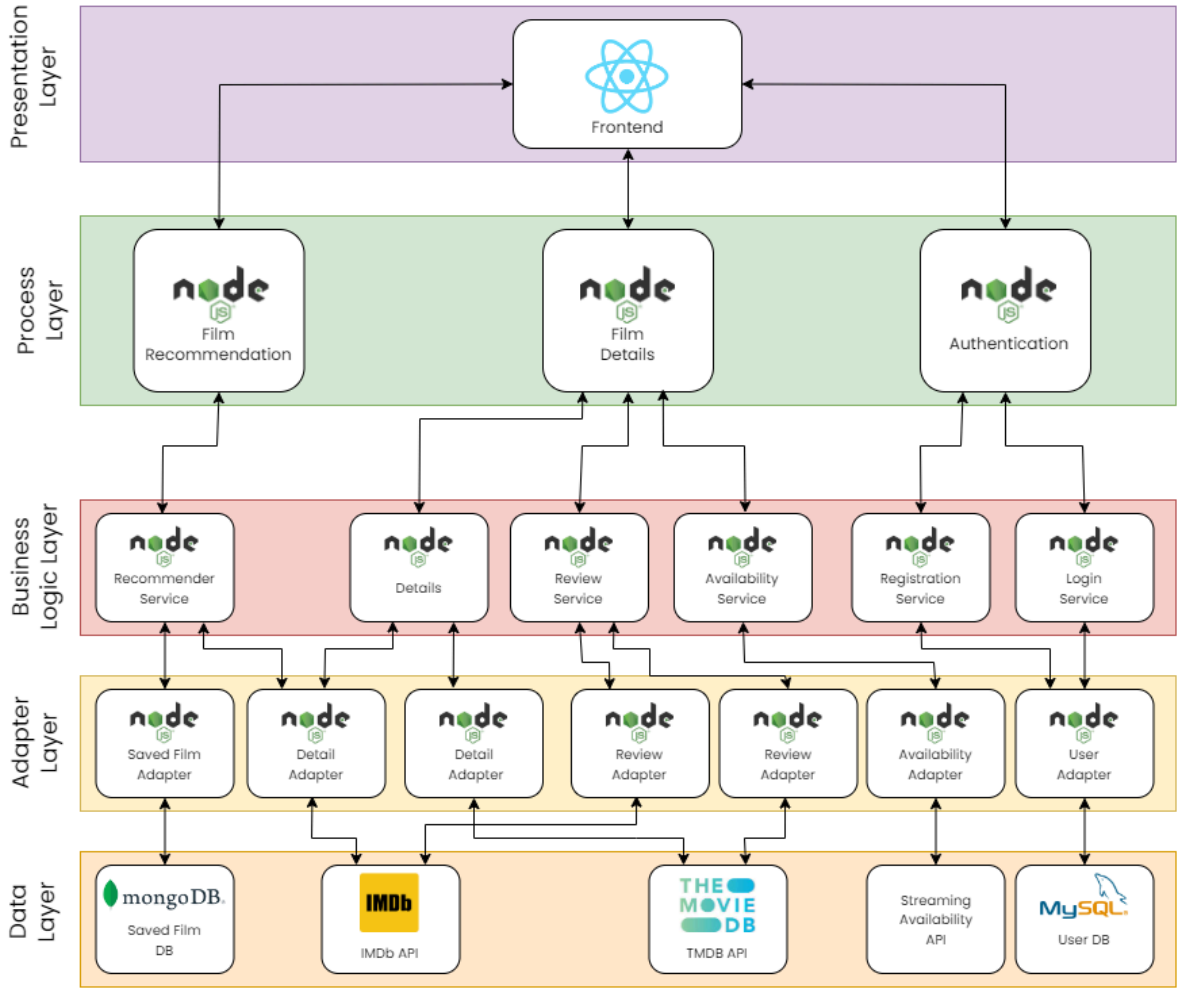
**Service Design and Engineering 2023/24**

Murtas Cristian 248025, Wang Marco 249368

## Introduction

The objective of this project was to provide users with a comprehensive overview of a film, by aggregating information from various sources. Recognizing that different platforms may contain different data about the same movie. Additionally, to mitigate bias, we offered a rating score based on multiple review system sources and averaged the result. Lastly, our platform also offers film recommendations based on users preferences, allowing them to maintain a history of the recommended films for future viewing.

# Architecture



## Presentation Layer

Through the Presentation Layer we provide a user-friendly way of interacting with the application,  we used technologies such as React, Html, Scss to build a responsive and powerful web application, furthermore we divided the application into components to ensure reusability and efficiency. All the user interactions are transformed into requests headed to the process centric services.

## Process Layer

The process-centric layer acts as a gateway to serve all the requests coming from the Presentation Layer. Each service coordinates services from the business layer to provide and fulfill complex user requests. The modules are designed to be as reusable and composable as possible.

- ### Authentication

  The user authentication process is crucial, since it allows us to keep track of the user information such as the history of recommended films. We provide a classic login option with email and password.
  This module also focuses on user registration and enables the monitoring of user sessions through the utilization of JWT tokens.

- ### Film Details Service

  This component is essential for gathering comprehensive information about a film. Encompassing details such as the plot, cast members, production company, reviews, and many others, as well as additional information like its availability on different streaming platforms. Giving the user a complete overview about the film and its different aspects.

- ### Film Recommendation Service

  This component offers users a personalized film discovery experience by leveraging their preferences to suggest similar movies based on a movie that the user enjoyed. Additionally, it provides history of all recommended films, ensuring users can conveniently access and watch them at a later time.

## Business Layer

On this layer we find the logic of all the functionalities offered by the application. For the  Authentication process, this layer ensures that the information coming from the user is validated. Furthermore, it encompasses the logic behind the JWT. For the detail and review process, it manages the logic by gathering and processing all the information coming from the respective adapters, eliminating redundant information and averaging the rating score from different sources.

## Adapter Layer

This layer contains the adapters that standardize data concerning various aspects of films, reviews, and user details. These adapters enable abstraction, allowing the upper layers to efficiently process the data without being worried about the specific format of the  source.
An example could be the various review adapters that source data from different origins, all of which are standardized to adhere to the same format.

## Data Layer

This layer is responsible for providing all data related requests. In particular, it is concerned with persistence and retrieval tasks, communicating with external APIs. For our project we relied on different data sources, such as:

- A MySQL database used within the Authentication module, to store user credential information.
- A MongoDB database utilized by the Film Recommendation process to save the history of recommended films for a specific user.
- IMDB api , TMDB api  are our source of information relative to the film, they provide information such as title, cast, reviews and many more.
- Streaming availability API this api allows us to search for the availability of a film on different platforms.

## Additional Information

The deployment process was carried out using Docker and orchestrated using Docker-compose, perfectly aligning with the service oriented architecture. The services are developed using Nodejs and the Express framework, while the UI has been implemented with JS and React.

## Useful Links

- The source code of the application is available on our GitHub Repository