Film Catalog

A Django Movie Database

A Final Project for the Python Development Program

By: Emanuel Panait

October 20, 2025

Project Overview

What is Film Catalog?

A dynamic, full-stack web application designed to manage and display a collection of movies with a user-friendly interface for browsing and searching, supported by a secure backend for content administration.

Core Goal

To create a seamless experience for film enthusiasts to discover, search, and manage movie information with an intuitive interface and robust backend security.

Built with Django

Developed from the ground up using the **Django framework** in Python, leveraging its robust features for rapid development and clean, pragmatic design.

Key Feature - Browsing and Filtering

Dynamic Movie Display

The homepage populates with all movies from the database, displayed in a clean, card-based layout for easy browsing.

The Shawshank
Drama

Inception
Sci-Fi

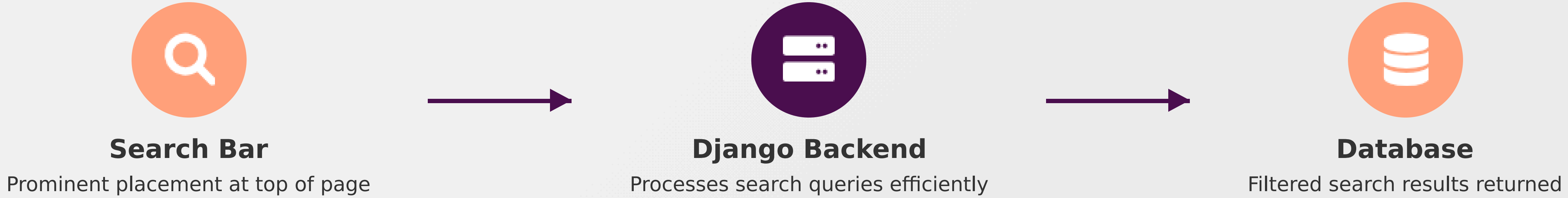
Die Hard
Action

Intuitive Genre Filtering

Instantly sort the catalog by clicking on genre buttons. Try clicking **Drama**, **Action**, or **Sci-Fi** to see the dynamic filtering in action.

All | Drama | Action | Sci-Fi | Comedy | Horror

Key Feature - Powerful Search



Search Bar

Prominent placement at top of page

Django Backend

Processes search queries efficiently

Database

Filtered search results returned

Find Movies Instantly

A prominent search bar at the top of the page allows users to search the entire database for a specific title with lightning speed.

Backend Integration

The search function sends a query to the Django backend, which filters the database and returns only the matching results.

Key Advantage: This integrated approach ensures that finding a particular film is both quick and highly efficient, enhancing the overall user experience.

Admin Panel - Secure Content Management

Secure Authentication

The application implements a stringent role-based access control system to safeguard the movie database, ensuring that only authorized administrators can access content management functions.

Role-Based Access Controls

Public View

- Regular visitors remain unaware of admin panel
- Content management controls hidden from public
- Standard browsing functionality available

Admin View

- Authorized admins gain access to all controls
- "Add Movie", "Update Movie", and "Delete Movie" buttons become visible
- Full control over movie database integrity

Security Benefits

This layered security approach is fundamental to preserving the integrity and security of the Film Catalog's data, ensuring that the movie database remains accurate and protected from unauthorized changes.

Admin Features - Full CRUD Operations

Administrators have full control over the movie database through these CRUD operations:

Create

Add new movie entries to the database through a user-friendly form.

Read

View all movies on the main page.

Update

Modify existing movie details using the Update function on each card.

Delete

Permanently remove movies using the Delete function.

These operations are only accessible to authenticated administrators.

Supporting Pages

Enhancing the application's professional presentation and user engagement

About Me Page

A dedicated section providing information about the developer, Emanuel Panait, offering context about the project's origin and the developer's background.

✓ Humanizes the application

Contact Page

Designed to facilitate communication, this page offers a straightforward method for users or visitors to get in touch, completing the professional feel of the application.

✓ Enhances user engagement

Professional Completion

These supporting pages contribute to the application's completeness, providing essential human and contact elements that professionalize the overall presentation and user experience.

User Experience - Smooth Transitions

Modern Web Design

The application was conceptualized with contemporary web design principles, aiming for an aesthetic and functional interface that feels fresh and professional.

Fluid Interface

Smooth transitions are implemented throughout the application, creating a seamless flow between different states and enhancing the overall user experience.

User Satisfaction

The result is a more intuitive, enjoyable experience that keeps users engaged and coming back for more.

Seamless Navigation

Transitions between page loads and during filter actions minimize jarring screen refreshes, creating a more cohesive browsing experience.

Interactive Elements

Genre filtering and search functions update the display without full page reloads, providing instant feedback and maintaining context.

Transition Effect Demo

Hover over the colored bar to see transition effect

Technology Stack

Backend

Python with the **Django Framework**

- High-level web framework for rapid development
- Clean, pragmatic design principles

Frontend

HTML, CSS (Tailwind), and JavaScript

- Structure with HTML
- Styling with CSS (utility-first)

Database

PostgreSQL or SQLite with **Django ORM**

- Relational database system
- Object-Relational Mapper for Python

Architecture

Model-View-Template (MVT) Design Pattern

- Separates data logic (Model)
- Handles presentation logic (View)

Conclusion

The Film Catalog project successfully demonstrates a full-stack web application with key features for both users and administrators.

Dynamic Filtering & Search

Implemented advanced features allowing users to efficiently browse and locate films through genre filtering and a powerful search bar.

Secure Admin Authentication

Established a secure login and logout system for administrators, ensuring controlled access to content management functionalities.

Full CRUD Capabilities

Provided administrators with complete Create, Read, Update, and Delete operations, enabling comprehensive management of the movie database.

Responsive User Interface

Designed an intuitive and visually appealing interface with smooth transitions, enhancing the overall user experience.

A successful demonstration of Django's capabilities for full-stack web application development.

Thank You & Questions

Thank you!

for your attention

Questions?

Emanuel Panait
Python Development Program