MovieDL

MovieDLproject.com

Grant Crawford
@Zosimov735

Description

MovieDL is a fully featured ecommerce application allowing for filtering, sorting, selection and purchase of more than 15,000 movies. MovieDL allows users to securely register and login with freedom to customize their information. With easy to use admin features site managers can control details of movie products and view sitewide order history.

Features

- REST API serving proprietary movie collection (Sourced from TMDB API)
- Product browse with random recommendations across site
- Search from any page
- Filter movies by genre
- Sort products alphabetically, by price, or by movie release date
- Custom shopping cart system with simultaneous client and server persistence via REST endpoints, with present-cart prioritization
- User checkout with Stripe API payment UI for credit card payments (test mode only)
- User order history tables with sort, pagination
- User registration, login and editable profile page
- Custom logging for Spring Security filters
- Admin portal/dashboard - edit product details, delete products, add new products, view all orders
- Chat GPT API integration - AI chatbot accessible from any page, answers user movie questions
- EmailJS API integration via Contact Us page
- Fully responsive styling, custom built UI components
- Unit testing for Rest API
- Hosted via AWS: RDS, EC2, S3, CloudFront, Route 53 - www.moviedlproject.com

Planning - User Stories

These are just a few of the more than 50 user stories completed.

b999: Order history - create and connect to DB
in list Sprint 6 Completed

Members
JJ +

Notifications
👁 Watch

📝 **Description** [Edit]

(Medium)

As a user I would like to be able to see the purchases I have made from MovieDL. These will be returned from an API endpoint pulling from the database.

F1509b Cart user flow for LOGGED IN USER
in list Sprint 6 Completed

Members
👤 +

Labels
🟦 +

Notifications
👁 Watch

📝 **Description** [Edit]

(XXXL)

As a user I would like any previous cart to be replaced with my newest cart when I log in if I have placed items in the cart before log in if I do not have any items in my cart before logging in I would like the last cart I had to still be present.

- On log in, DB cart is represented in state locally, replaces existing localStorage cart.

B106: Genre Filtering Feature
in list Sprint 3 Completed 👁

Members
GC +

Notifications
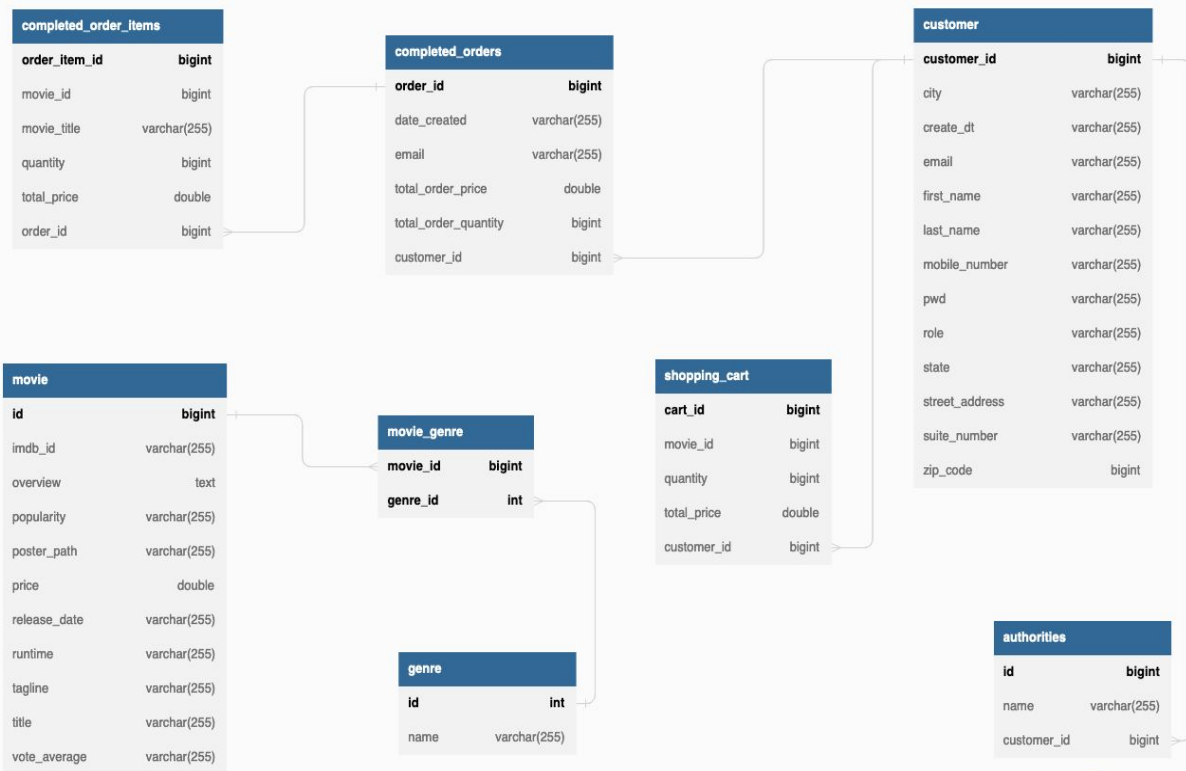👁 Watching ✓

📝 **Description** [Edit]

(MEDIUM)

As a user, I would like to be able to filter the results of searches by specific genre[s]. I would like the filter to apply to all movies before entering a search term, and I would like it to apply to search terms if entered.

To be complete this feature would need to be able to return based on one or more genres.

Planning - Database

Tables:

- authorities
- completed_order_item
- completed_orders
- customer
- genre
- movie
- movie_genre
- shopping_cart

completed_order_items

order_item_id	bigint
movie_id	bigint
movie_title	varchar(255)
quantity	bigint
total_price	double
order_id	bigint

completed_orders

order_id	bigint
date_created	varchar(255)
email	varchar(255)
total_order_price	double
total_order_quantity	bigint
customer_id	bigint

customer

customer_id	bigint
city	varchar(255)
create_dt	varchar(255)
email	varchar(255)
first_name	varchar(255)
last_name	varchar(255)
mobile_number	varchar(255)
pwd	varchar(255)
role	varchar(255)
state	varchar(255)
street_address	varchar(255)
suite_number	varchar(255)
zip_code	bigint

movie

id	bigint
imdb_id	varchar(255)
overview	text
popularity	varchar(255)
poster_path	varchar(255)
price	double
release_date	varchar(255)
runtime	varchar(255)
tagline	varchar(255)
title	varchar(255)
vote_average	varchar(255)

movie_genre

movie_id	bigint
genre_id	int

shopping_cart

cart_id	bigint
movie_id	bigint
quantity	bigint
total_price	double
customer_id	bigint

genre

id	int
name	varchar(255)

authorities

id	bigint
name	varchar(255)
customer_id	bigint

dbdiagram.io

Technology Stack

Front End

- ReactJS
- React Router DOM
- Zustand (State management)
- Styling via Bulma, MUI, FontAwesome Icons

Back End

- Spring Boot
- Spring Security w/ JWT
- Hibernate & JPA
- MySQL

Libraries & APIs

- TMDB - The Movie Database API
- Stripe
- JWT-Decode
- EmailJS
- Toastify

- OpenAI (GPT)
- React Select

Demo: MovieDLproject.com

What I Learned

- ReactJs
- RestAPI in Spring Boot
- Spring Security
- Styling in Bulma & MUI
- API integration
- Unit Testing
- Deployment + hosting with AWS
- State management for client side
- JWT authentication for server side
- Expanded JPA capability

What's Next

- Guest checkout
- Expanded admin
- Expanded filter, sort, search capability
- Movie reviews
- Favorites
- Lost Password recovery