Movie Website Project

December 2nd, 2022

Lucas Colegrove

TABLE OF CONTENTS

Introduction - 3

Related Works - 3

System Architectural Design - 4

ER Diagram - 5

Relational table: movie - 6

Detailed Description of Components - 6

Movie Information Viewing Servlets - 6

Movie Servlet Pictures - 7

Random Movie Servlet - 11

Conclusion - 12

Introduction

The movie website project, from now on referred to as MovieQuest or its URL 54.163.37.17/FinalProject/final.html provides an easy-to-use website for movie lovers to look up information on movies of various genres and decide what to watch. At the home page users are allowed to view various movie lists based off genre. Upon clicking a specific movie or the random button they are sent to a corresponding movie page that has various pieces of information about each specific movie. The implementation of this project was designed as a three-tier web application that separates the client presentation, server-side logic using tomcat, and a MySQL database data system for storing movie information. The presentation layer is static content delivered to the user as a front-end interface interpreted by whatever web browser the client is using. When the client interacts with the website, server-side logic handles the request and pulls information from the MySQL database. This data is returned to the presentation layer where it is dynamically presented on the page. The MySQL database provides storage for all the movie information that is requested via the client side.

Related Works

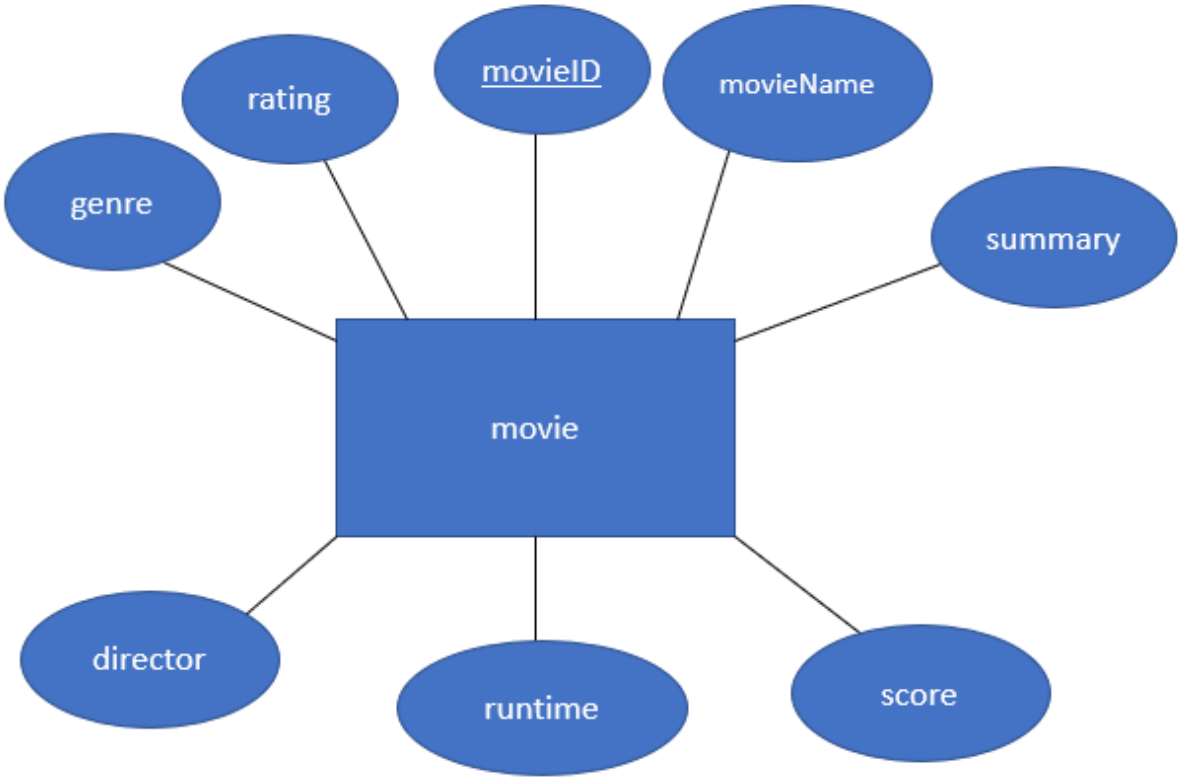
The functionality of MovieQuest draws inspiration from the popular movie database website IMDB. IMDB has useful information on 1000s of movies and is the go-to source for many movie lovers when researching their favorite films. MovieQuest is not as fleshed out as this industry giant but hits the similar goal of displaying information about movies. Unlike IMDB, the random feature of MovieQuest was inspired by the indecisiveness people face when choosing what to watch. By giving the choice to chance my website hopes to give people a unique alternative to their regular ritual of picking what to watch.

System Architectural Design

The system architecture is composed of the general three-tier web application design described in the introduction. The application was deployed across 2 EC2 instances, one contains the presentation layer and the other contains the logic and data layers. The first instance runs Apache Tomcat server containing the Java servlets and MySQL for the database storing MovieQuest data. The second instance runs Apache web server and hosts static web content. The URL provided in the introduction points to the instance running Apache Web Server. The web server connects to the Tomcat server using mod_jk so that requests are routed to Tomcat server worker. Each Servlet is assigned a specific path after the /FinalProject/ part of the URL depending on the requested action. Java servlets use the Java Database Connectivity library to connect with the MovieQuest MySQL database. The front-end HTML and CSS did not use any special framework. All information pulled by the servlets was done by getting the database ID of each specific movie and selecting all its information in a MySQL SELECT statement. This information was then concatenated into a string in the format of an HTML document to be represented by the browser. Using two EC2 instances to host client-side presentation and server-side logic required cross-origin resource sharing. MySQL was a good option for the data system as the relational database management system proved to be easy to use and manipulate. The design of the entity-relationship model for the data needed by the MovieQuest application is shown below.

E-R diagram

The ER diagram displays how the information is organized in the database. Each movie entity has several attributes that represent what a client may want to know about the movie. Each movie entity has 1 of each attribute. For example, a movie can't have more than one movieName.



Relational table: movie

The movie table stores the data for the movies displayed on the MovieQuest home page. The primary key is the movieID which is used by the servlets to gather the rest of the movie information from the MySQL database.

movieID	name	genre	rating	summary	runtime	director	score
varchar(20)	varchar(50)	varchar(20)	varchar(20)	varchar(300)	varchar(3)	varchar(50)	varchar(4)

Detailed Description of Components

The following is a description of each component of the MovieQuest application as a servlet via Tomcat. Screenshots are provided to see how the front end represents the servlet response. All of the servlets forward requests to the doGet method. The Java Database Connectivity Library is utilized for all the communication between the database and the servlets.

Movie Information Viewing Servlets

The following screenshots are for each servlet that handles a request when a user clicks a movie info button. Each movie's button is tied to a separate servlet to display the information about that movie that is stored in the database. The servlet takes the ID of the movie in the database and pulls all the information necessary to fill in the page. The servlet contains HTML being stored as a string which is interpreted by the client side to render the page. By doing this I hope in the future to make unique pages for the movies with aesthetic choices that match each particular movie. (see pages 7-11 for servlet pictures)

Not secure | 54.163.37.17/FinalProject/DjangoServlet

MovieQuest

Back

Django Unchained

- Genre: Action
- Rating: R
- Summary: With the help of a German bounty-hunter, a freed slave sets out to rescue his wife from a brutal plantation-owner in Mississippi.
- Runtime: 165 minutes
- Director: Quentin Tarantino
- Score: 8.4/10 ★

Not secure | 54.163.37.17/FinalProject/AvengersServlet

MovieQuest

Back

The Avengers

- Genre: Action
- Rating: PG-13
- Summary: Earths mightiest heroes must come together and learn to fight as a team if they are going to stop the mischievous Loki and his alien army from enslaving humanity.
- Runtime: 143 minutes
- Director: Joss Whedon
- Score: 8.0/10 ★

Not secure | 54.163.37.17/FinalProject/StarWarServlet

MovieQuest

Back

Star Wars: Episode IX - The Rise of Skywalker

- Genre: Action
- Rating: PG-13
- Summary: In the riveting conclusion of the landmark Skywalker saga, new legends will be born-and the final battle for freedom is yet to come.
- Runtime: 141 minutes
- Director: J.J. Abrams
- Score: 6.5/10 ★

Not secure | 54.163.37.17/FinalProject/Servlet1917

MovieQuest

Back

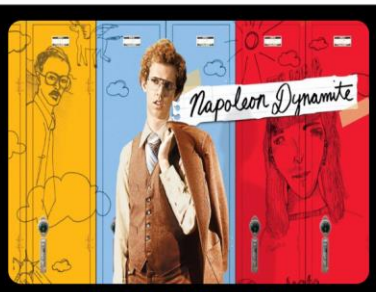
1917

- Genre: Action
- Rating: R
- Summary: April 6th, 1917. As an infantry battalion assembles to wage war deep in enemy territory, two soldiers are assigned to race against time and deliver a message that will stop 1,600 men from walking straight into a deadly trap.
- Runtime: 119 minutes
- Director: Sam Mendes
- Score: 8.2/10 ★

Not secure | 54.163.37.17/FinalProject/NapoleonServlet

MovieQuest

Back



Napoleon Dynamite

- Genre: Comedy
- Rating: PG
- Summary: A listless and alienated teenager decides to help his new friend win the class presidency in their small western high school, while he must deal with his bizarre family life back home.
- Runtime: 156 minutes
- Director: Jared Hess
- Score: 6.9/10 ★

Not secure | 54.163.37.17/FinalProject/HangoverServlet

MovieQuest

Back

The Hangover

- Genre: Comedy
- Rating: R
- Summary: Three buddies wake up from a bachelor party in Las Vegas, with no memory of the previous night and the bachelor missing. They make their way around the city in order to find their friend before his wedding.
- Runtime: 100 minutes
- Director: Todd Phillips
- Score: 7.7/10 ★

Not secure | 54.163.37.17/FinalProject/AnchormanServlet

MovieQuest

Back

Anchorman: The Legend of Ron Burgundy

- Genre: Comedy
- Rating: R
- Summary: Ron Burgundy is San Diegos top-rated newsman in the male-dominated broadcasting of the 1970s, but thats all about to change for Ron and his cronies when an ambitious woman is hired as a new anchor.
- Runtime: 94 minutes
- Director: Adam Mckay
- Score: 7.1/10 ★

Not secure | 54.163.37.17/FinalProject/ItServlet

MovieQuest

Back

It

- Genre: Horror
- Rating: R
- Summary: In the summer of 1989, a group of bullied kids band together to destroy a shape-shifting monster, which disguises itself as a clown and preys on the children of Derry, their small Maine town.
- Runtime: 135 minutes
- Director: Andy Muschietti
- Score: 7.3/10 ★

Not secure | 54.163.37.17/FinalProject/ConjuringServlet

MovieQuest

Back

The Conjuring

- Genre: Horror
- Rating: R
- Summary: Paranormal investigators Ed and Lorraine Warren work to help a family terrorized by a dark presence in their farmhouse.
- Runtime: 112 minutes
- Director: James Wan
- Score: 7.5/10 ★

Not secure | 54.163.37.17/FinalProject/UsServlet

MovieQuest

Back

Us

- Genre: Horror
- Rating: R
- Summary: A familys serene beach vacation turns to chaos when their doppelgängers appear and begin to terrorize them.
- Runtime: 116 minutes
- Director: Jordan Peele
- Score: 6.8/10 ★

Not secure | 54.163.37.17/FinalProject/HereditaryServlet

MovieQuest

Back

Hereditary

- Genre: Horror
- Rating: R
- Summary: A grieving family is haunted by tragic and disturbing occurrences.
- Runtime: 127 minutes
- Director: Ari Aster
- Score: 7.3/10 ★

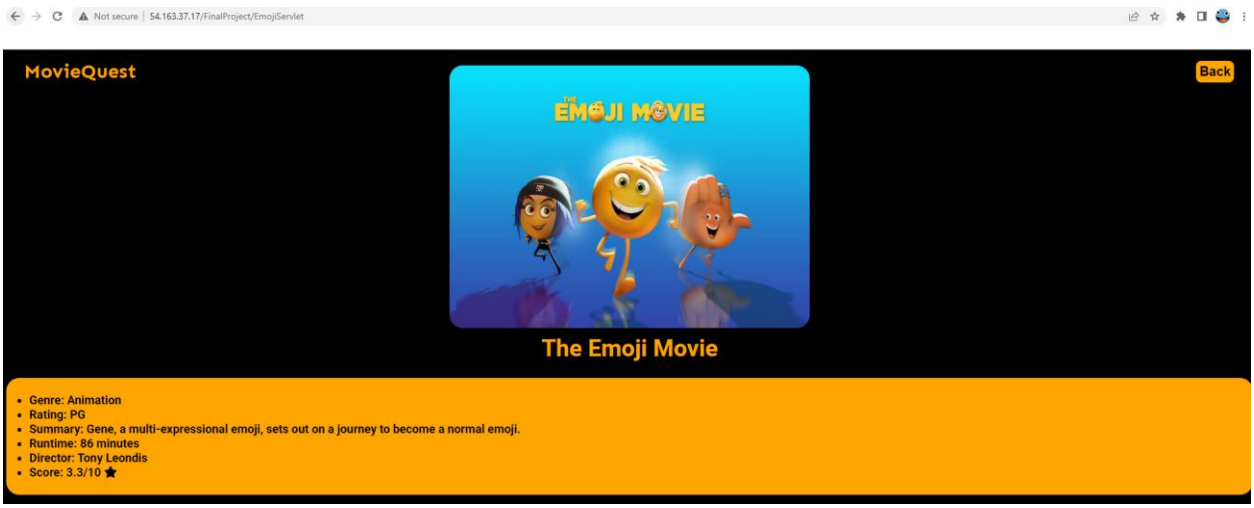
Not secure | 54.163.37.17/FinalProject/BatmanServlet

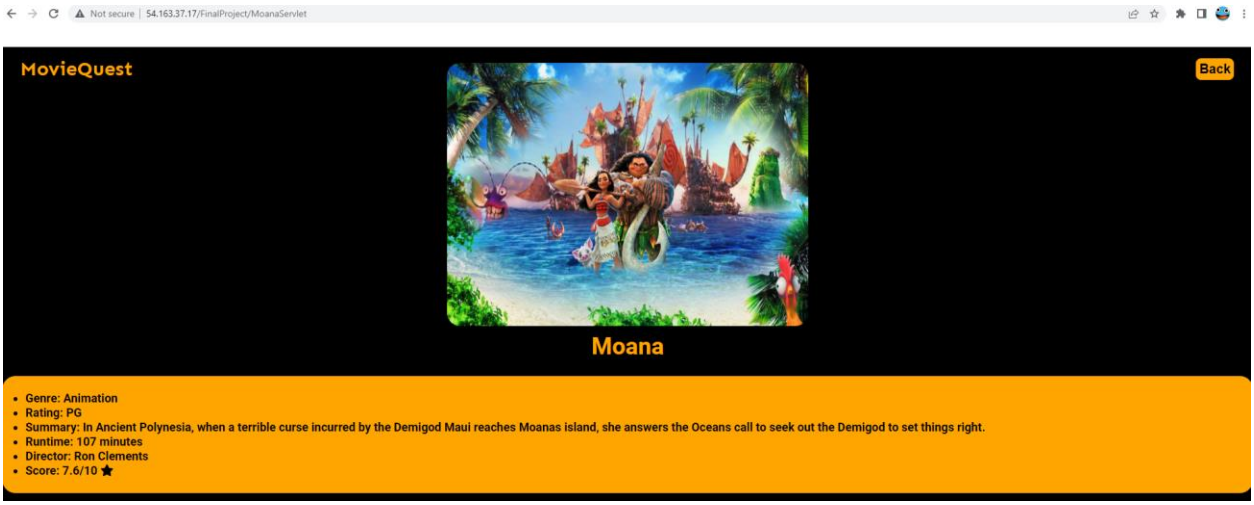
MovieQuest

Back

The Lego Batman Movie

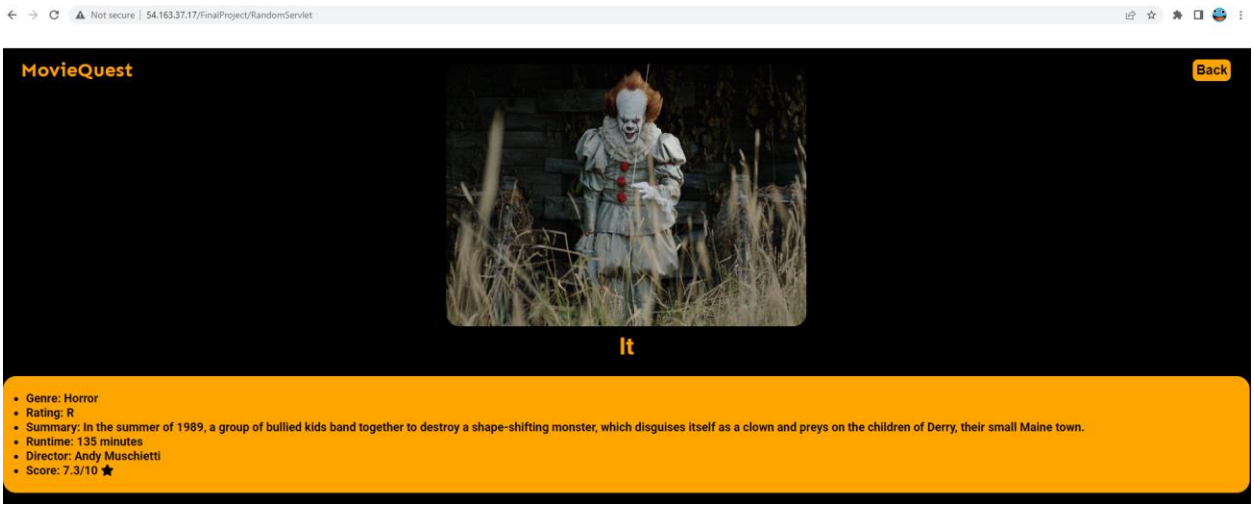
- Genre: Animation
- Rating: PG
- Summary: A cooler-than-ever Bruce Wayne must deal with the usual suspects as they plan to rule Gotham City, while discovering that he has accidentally adopted a teenage orphan who wishes to become his sidekick.
- Runtime: 104 minutes
- Director: Chris Mckay
- Score: 7.3/10 ★





Random Movie Servlet

The following screenshot is for the servlet that handles the request when a user clicks the random button. The servlet takes the ID of a random movie in the database and pulls all the information necessary to fill in the page. The servlet contains HTML being stored as a string which is interpreted by the client side to render the page. The purpose of the random choice is to make deciding what to watch a fun game for spontaneous users. (see page 12 for servlet picture)



Conclusion

The final MovieQuest demo delivers a functional three-tier web app hosted at 54.163.37.17/FinalProject/final.html and provides an easy-to-use webpage for looking up information on popular movies. The web app has no observable errors but in the future I would like to add search bar functionality and functionality to the arrows to move through a row of movies on the home page (final.html). Overall, the web app runs responsively and the layout is sleek. The MySQL database the movie information is stored on is efficient and can be updated at any time to include information on more movies.