Database Design Document - Online Movie Streaming Platform

1. Introduction

1.1 Purpose

This document provides the database design for the Online Movie Streaming Platform, including the schema, relationships, indexing strategies, and data integrity constraints.

1.2 Scope

The database will store and manage user information, movies, genres of movies, reviews, actor and director information and watchlist. It will ensure data integrity, security, and efficient retrieval for seamless user experience.

1.3 Target Audience

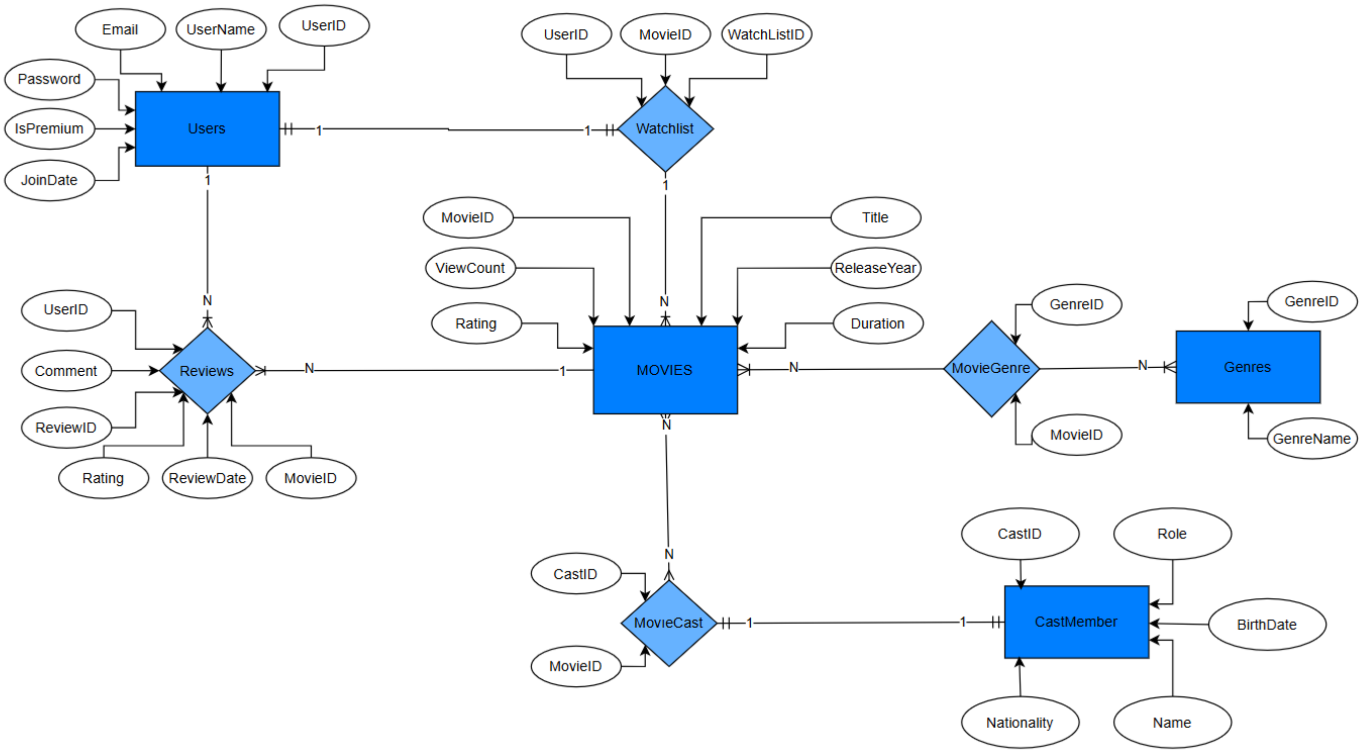
- Database administrators
- Backend developers
- System architects

2. Database Architecture

2.1 Database Management System

- **Type**: Relational Database Management System (RDBMS)
- **Preferred DBMS**: MySQL
- **Normalization**: Third Normal Form (3NF) to optimize storage and eliminate redundancy

2.2 ER Diagram



3. Database Schema

3.1 Users Table

Column	Type	Constraints	Description
UserID	INT (PK)	AUTO_INCREMENT	Unique user identifier
Username	VARCHAR(50)	UNIQUE NOT NULL	User’s name
Email	VARCHAR(100)	UNIQUE NOT NULL	User’s email
Password	VARCHAR(100)	NOT NULL	User's password
JoinDate	DATETIME	DEFAULT CURRENT_TIMESTAMP	Date the user joined the platform
IsPremium	BOOLEAN	DEFAULT FALSE	User's subscription status

3.2 Genres Table

Column	Type	Constraints	Description
GenreID	INT (PK)	AUTO_INCREMENT	Unique genre identifier
GenreName	VARCHAR(255)	UNIQUE NOT NULL	genre name

3.3 MovieGenre Table

Column	Type	Constraints	Description
GenreID	INT (FK)	REFERENCES Genres(GenreID) ON DELETE CASCADE	Unique genre identifier
MovieID	INT(FK)	REFERENCES Movie(MovieID) ON DELETE CASCADE	Unique movie identifier

3.4 Movies Table

Column	Type	Constraints	Description
MovieID	INT (PK)	AUTO_INCREMENT	Unique movie identifier
Title	VARCHAR(255)	NOT NULL	Movie Title
ReleaseYear	SMALLINT(4)	NOT NULL	Movie Release Year
Duration	SMALLINT	NOT NULL CHECK (Duration > 0)	Duration of the movie
ViewCount	INT	DEFAULT 0	Number of views of the movie
Rating	DECIMAL(3,1)	DEFAULT 0.0	Movie Rating

3.5 Reviews Table

Column	Type	Constraints	Description
ReviewID	INT (PK)	AUTO_INCREMENT	Unique review identifier
UserID	INT (FK)	REFERENCES Users(UserID) ON DELETE CASCADE	Unique user identifier
MovieID	INT (FK)	REFERENCES Movies(MovieID) ON DELETE CASCADE	Unique movie identifier
Rating	DECIMAL(3,1)	CHECK (Rating BETWEEN 0.0 AND 10.0)	Rating of the movie
Comment	TEXT	(optional)	Comment in the movie review.
ReviewDate	DATETIME	DEFAULT CURRENT_TIMESTAMP	Date of the review

3.6 CastMember Table

Column	Type	Constraints	Description
CastID	INT (PK)	AUTO_INCREMENT	Unique log identifier
Name	VARCHAR(255)	NOT NULL	cast('Actor', 'Director') name
Role	ENUM('Director', 'Actor')	NOT NULL	Role of the cast
BirthDate	DATE	(optional)	actors' birth date
Nationality	VARCHAR(100)	(optional)	nationalities of actors

3.7 Watchlist Table

Column	Type	Constraints	Description
WatchlistID	INT (PK)	AUTO_INCREMENT	Unique watchlist identifier
UserID	INT(FK)	REFERENCES Users(UserID) ON DELETE CASCADE	Unique user identifier
MovieID	INT(FK)	REFERENCES Movies(MovieID) ON DELETE CASCADE	Unique movie identifier

3.8 MovieCast Table

Column	Type	Constraints	Description
MovieID	INT (FK)	REFERENCES Movies(MovieID) ON DELETE CASCADE	Unique movie identifier
CastID	INT(FK)	REFERENCES CastMember(CastID) ON DELETE CASCADE	Unique cast identifier

4. Indexing Strategy

- **Primary Keys**: Used on `id` fields to ensure uniqueness and fast lookups.
- **Foreign Keys**: Enforce referential integrity between tables.
- **Indexes**:
  - **Index on email in Users Table (for faster login lookups).**
  - **Index on MovieID in Reviews Table (for efficient retrieval of movie reviews).**
  - **Index on GenreID in Movies Table (to quickly fetch movies by genre).**
  - **Index on ViewCount in Movies Table (for optimizing queries that list popular movies).**
  - **Index on UserID in Watchlist Table (to speed up retrieval of a user's watchlist).**

5. Security Measures

- **Data Encryption**:
  - Passwords stored using bcrypt hashing.
  - Sensitive data encrypted using AES-256.
- **Access Control**:
  - Role-based access control (RBAC) to restrict unauthorized actions.
- **Audit Logging**:
  - All user actions logged for monitoring and security.

6. Scalability & Performance Optimization

- **Read Replicas**: Database replication for load balancing.
- **Caching Mechanism**: Implement Redis or Memcached for frequent queries.
- **Partitioning Strategy**: Time-based partitioning for logs and history tables.
- **Connection Pooling**: Optimized DB connections for performance.

7. Backup & Disaster Recovery Plan

- **Database Backups**: Daily full backups, hourly incremental backups.
- **File Storage Backups**: Version-controlled backups of static assets.
- **Failover Strategy**: Multi-region deployment for high availability.
- **Recovery Time Objective (RTO)**: Less than 15 minutes.
- **Recovery Point Objective (RPO)**: Less than 5 minutes.

8. Conclusion

This database design document provides a robust foundation for the Online Movie Watching Platform. It ensures scalability, security, and efficiency while maintaining data integrity and accessibility.