

System Design for Spheremovie

Group 3

EECS 3311

Murtaza, Karishma, Salan, Allen

# Table of Contents

CRC for all the classes.....	3 - 6
Software Architecture Diagram.....	7

**Class Name:** Movie

**Parent Class (if any):** N/A

**Subclasses (if any):** N/A

**Responsibilities:**

- Model a document representing a movie stored in the MongoDB collection of movies
- Attributes will include the movie's unique ID, its IMDb ID, its title, the release date, a link to a trailer, a poster, backdrops, and genres
- Keep a reference to associated reviews

**Collaborators:**

- Review class: it has a list of review objects that are needed to link or keep track of movie reviews.

**Class Name:** Movie Controller

**Parent Class (if any):** N/A

**Subclasses (if any):** N/A

**Responsibilities:**

- It manages and responds HTTP requests associated with the movie data
- Makes an endpoint to retrieve all the movies and a single movie by the IMDb IDs

**Collaborators:**

- Optional: Holds the movie data for when a response is not available
- Spring Framework: for annotations such as `@CrossOrigin`
- MovieService class: handles the logic necessary to access movie data
- Movie class: holds the movie details provided in the responses

**Class Name:** Movie Repository

**Parent Class (if any):** Mongo Repository

**Subclasses (if any):** N/A

**Responsibilities:**

- It is an interface that interacts with the MongoDB for the operation to create, read, update, and delete (CRUD) on the Movie class documents
- Implements method to retrieve a movie using the IMDb ID

**Collaborators:**

- Optional
- Mongo Repository
- Movie class

**Class Name:** Movie Service

**Parent Class (if any):** N/A

**Subclasses (if any):** N/A

**Responsibilities:**

- Provides service layer which interacts with the repository for managing the movie data
- Retrieves all the movies from the repository and any specific movie by the imdb id

**Collaborators:**

- Movie class
- Optional
- MovieRepository class

**Class Name:** Review

**Parent Class (if any):** N/A

**Subclasses (if any):** N/A

**Responsibilities:**

- This is a document for review which is stored in the collection of mongodb reviews
- Includes a constructor to initialize a review with the time details and content

**Collaborators:**

- LocalDateTime
- ObjectId
- Movie class

**Class Name:** Review Controller

**Parent Class (if any):** N/A

**Subclasses (if any):** N/A

**Responsibilities:**

- It manages and responds HTTP requests associated with the movie reviews
- Makes an API endpoint to create new reviews where it receives the review data and associates with a movie using the imdb id

**Collaborators:**

- Spring Framework
- Map
- ReviewService class
- Review class

**Class Name:** Review Repository

**Parent Class (if any):** MongoRepository

**Subclasses (if any):** N/A

**Responsibilities:**

- Provides access of database stored in mongodb to the Review documents
- Reviews are handled by the operation to create, read, update, and delete(CRUD)

**Collaborators:**

- MongoRespository
- ObjectId
- Review class

**Class Name:** Review Service

**Parent Class (if any):** N/A

**Subclasses (if any):** N/A

**Responsibilities:**

- New reviews are created and saved in the database
- For database operations, the mongodb tools are used
- Adding new reviews using the imdb id to the list of movie reviews

**Collaborators:**

- MongoTemplate
- Review class
- Update
- LocalDateTime
- ReviewRepository
- Criteria

- Movie

**Class Name:** Spheremovie Application

**Parent Class (if any):** N/A

**Subclasses (if any):** N/A

**Responsibilities:**

- Starts the sphere movie application
- The spring boot application is started
- Manages HTTP requests with the RestController default

**Collaborators:**

- Spring Framework
- SpringApplication
- RestController

# Software Architecture Diagram

