# subjective test – java Front end

## **media services System (mss)**

You are required to write a Console Application as a solution for the front-end requirement of a Media services system. The back-end of the application has been implemented using My SQL Server and the front-end of the application has to be implemented using java 8. The application should be used by administrator to manage the movie related details.

This software needs to address the following requirements:

**Business Requirements:**

The project is for administrators who use the system to perform the following operations.

The Administrator module should allow administrators to:

1. View the details regarding movies and actors.
2. Search movies.
3. Save new actor details.
4. Update the actor details.
5. Delete actor details.

**User Stories:**

***US01: Managing Movie Details***

As an Administrator,

I want to view the movie details in the front-end Java application in the following format:



* The headings should be ‘Movie Name’, ‘Actor Name’, Available Language’ and ‘Movie Category’.
* The actor name should be the full name (first name and last name).
* The Movie Name and Actor Name should be in proper case.
* The details should be fetched using a stored procedure named GET\_MOVIE\_DETAILS\_WITH\_PROC which is to be created in the backend.
* The proper case for movie name and actor name should be generated using a function called PROPER\_CASE which is to be created in the backend.
* The GET\_MOVIE\_DETAILS\_WITH\_PROC should use the PROPER\_CASE function
* Proper messages should be displayed in case of errors and exceptions

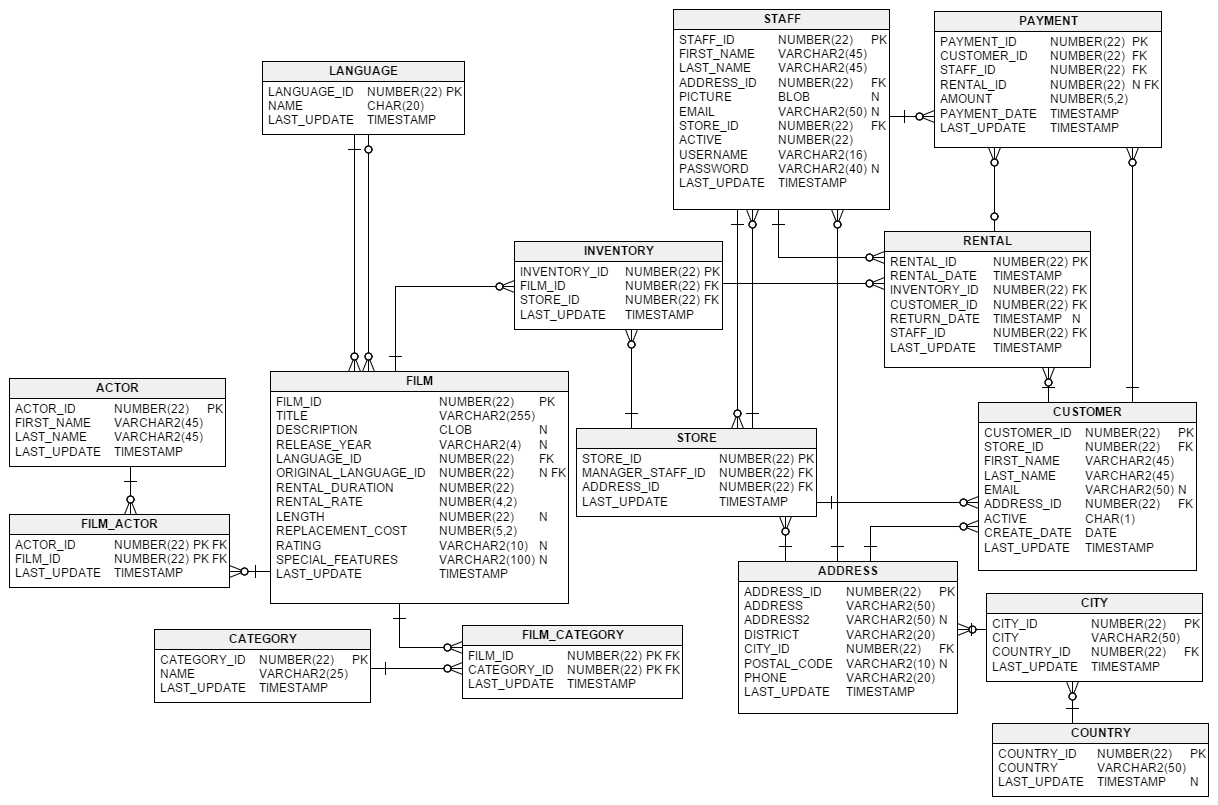
I want to search the movie details based on full of partial values for

* Actor Name,
* Language, and
* Available Language

I want to update and delete actors based on

* Actor Id and
* First Name

Given: ER Diagram



**Functional Requirements:**

1. The Application should be implemented using Java 8 and MySQL.
2. Apply singleton pattern for establishing the database connection.
3. Utilize the various features of Java 8.

**Best Practices**

1. Follow proper naming conventions and ensure code quality
2. Validate Parameter Preconditions
3. The code should have modularity.
4. Exceptions to be handled appropriately.

**Evaluation Parameters:**

The following parameters are to be met effectively in the developed solutions.

* Execution
* Completeness
* Logic Building
* Code Quality
* Best Practices (efficiency)

**Submission Requirements:**

1. Submit a single archive containing all the artefacts
2. The artefacts are Project Source Code and SQL scripts

**Project Presentation Requirements:**

1. The solutions should be presented with execution.
2. Explain the details of API used, processes involved, workflows and frameworks (if any) utilized.