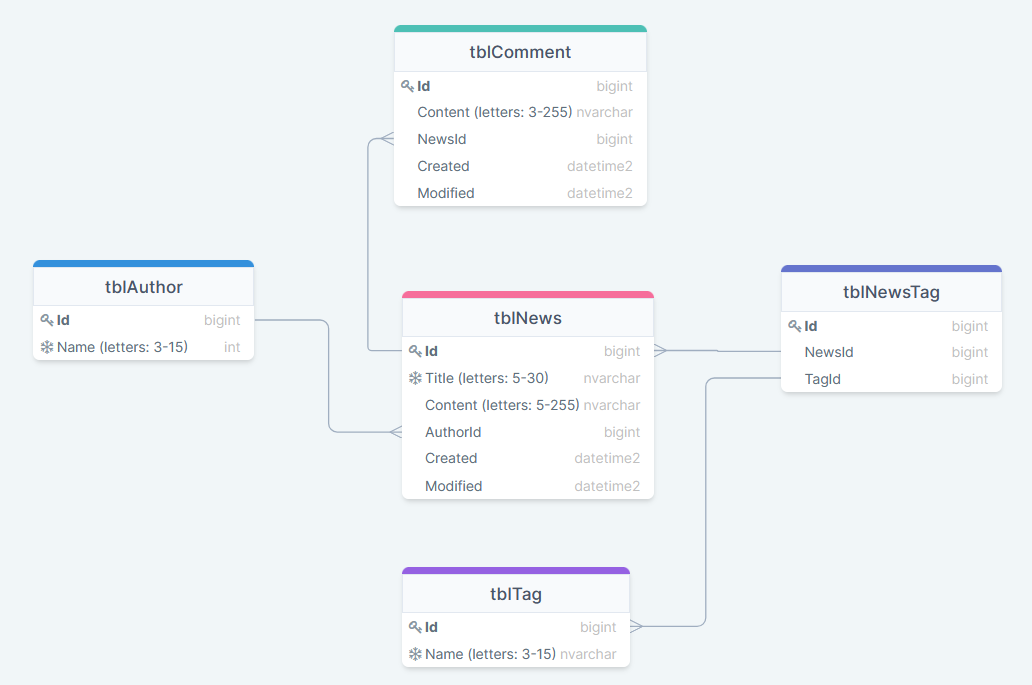
**News Management**

**Business requirements:**

1. Develop web service for News Management system with the following entities:



* – unique value
* All *Name, Title* and *Content* fields are required

2. The system should expose REST APIs to perform the following operations:

* CUD operations ([what\_is\_CRUD](https://en.wikipedia.org/wiki/Create,_read,_update_and_delete)) for News and Comment. If new tags and authors are passed during creation/modification – they should be created in the DB
* Get News:
  + get all
  + by Id
  + search (all params are optional and can be used in conjunction):
    - by tag names and tag ids (many tags)
    - by author name (one author)
    - by part of Title
    - by part of Content
  + sort by Created, Modified Asc/Desc. Default: Created Desc
* Get Comments:
  + by News Id (URL example: /news/{newsId}/comments)
  + by Id
  + sort by Created, Modified Asc/Desc. Default: Created Desc
* CRD operations for Tag, and Author
* Get Tags and Authors:
  + get all
  + by Id
  + by part Name
  + by News Id (URL example:

/news/{newsId}/tags *- should return tags collection*

/news/{newsId}/authors *- should return 1 author*)

* Get authors with amount of written news. Sort by news amount Desc.
* Optional: load Content of News through separate operation

3. Request URL, params and body should be validated.

4. Support HATEOAS on REST endpoints.

5. Use Swagger to document RESTful API.

**General requirements:**

1. Code should be clean and should not contain any “developer-purpose” constructions.

2. App should be designed and written with respect to OOD and SOLID principles.

3. Clear layered structure should be used with responsibilities of each application layer defined.

4. All business logic should be written in the service layer: mapping model to DTO and vice versa, validation, etc.

5. Place queries in constants (if they exist).

6. JSON should be used as a format for client-server communication messages. Optional: support XML.

**Application requirements:**

1. Gradle, latest version. Multi-module project.

2. Application packages root: com.mjc.school.

3. Hibernate should be used as a JPA implementation for data access.

4. Spring Framework & Spring Boot, the latest version.

5. Java Code Convention is mandatory.

**Demo:**

1. Application should be deployed before demo.

2. Demonstrate API using Postman tool. Prepare for demo Postman collection with APIs.

3. Candidate should be able to answer theoretical and practical questions during demo session.