UNIVERSITATEA TEHNICĂ
DIN CLUJ-NAPOCA

Assignment 1 Documentation

Table of Contents:

1. Introduction

The mini-StackOverflow Web Application is a simplified version of the popular Q&A platform Stack Overflow. It allows users to ask questions, provide answers, upvote/downvote questions and answers, and calculate user scores based on their activity within the platform. The application is built using Java with Spring Boot framework for backend development and MySQL for database management.

2. Architecture

The application follows a layered architecture, consisting of the following layers:

- Presentation Layer:
  - Handled by RESTful APIs implemented using Spring MVC framework.
  - Controllers manage HTTP requests and responses, interacting with the service layer.
- Service Layer:
  - Contains business logic and handles application functionality.
  - Services communicate with repositories for data access and perform operations such as question/answer creation, score calculation, and user authentication.
- Data Access Layer:
  - Responsible for interacting with the database.
  - Uses Spring Data JPA to simplify CRUD operations.
  - Entities represent database tables, and repositories provide methods for querying data.

3. Features

Feature 1: User Interaction

- Users can register accounts, log in, and log out.
- Authenticated users can ask questions, provide answers, upvote/downvote questions and answers, and edit/delete their own questions and answers.

Feature 2: Question and Answer Management

- Users can ask questions, providing a title, description, and optional tags.
- Answers can be provided to questions, including text content and optional image attachments.
- Questions and answers can be upvoted or downvoted by users, affecting their overall score.
- Questions and answers can be edited or deleted by their respective authors.

Feature 3: Score Calculation

- User scores are calculated based on their activity within the platform.
- Scores are influenced by upvotes and downvotes received on questions and answers.

- Score calculations are performed periodically or triggered by user activity updates.

Bonus Feature 1: Score Display

- User scores are displayed alongside their username on questions and answers.
- Scores can be negative, reflecting the user's overall activity and contribution to the platform.

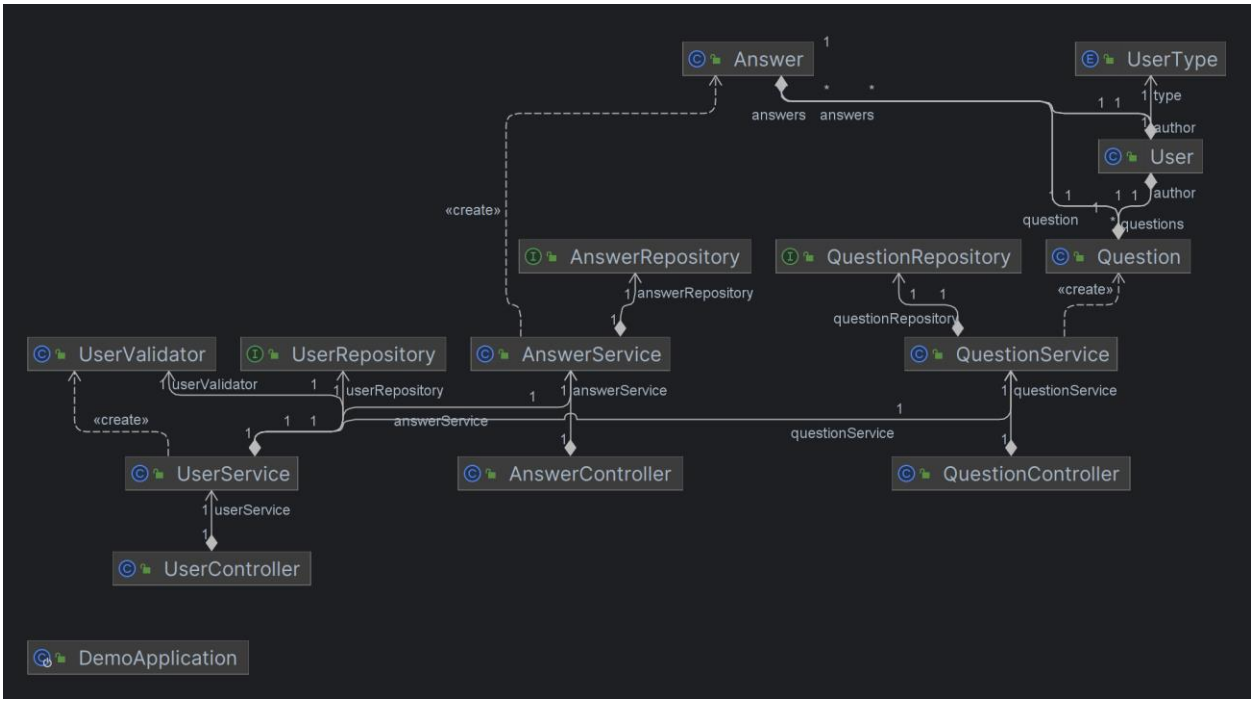
4. Data model

The application uses the following entities mapped to corresponding database tables:

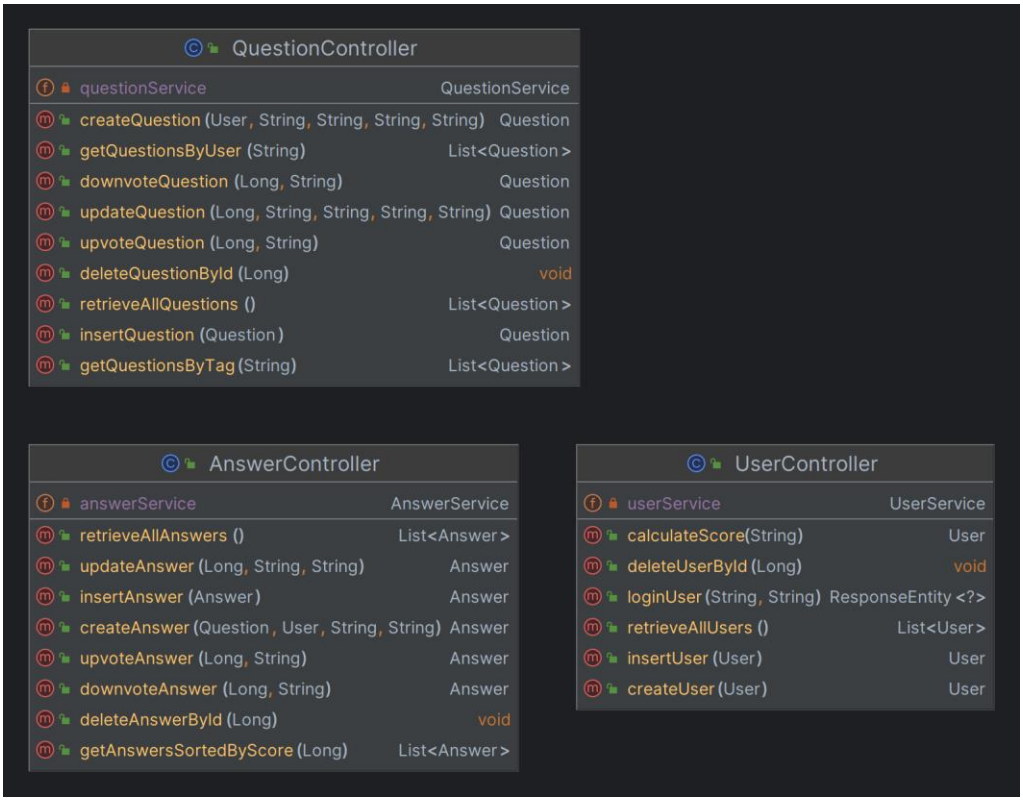
- User:
  - Represents registered users.
  - Contains fields for username, email, password (encrypted), user type, and score.
- Question:
  - Represents user-generated questions.
  - Contains fields for author, title, description, creation datetime, picture URL, tags, upvotes, and downvotes.
- Answer:
  - Represents answers provided to questions.
  - Contains fields for the parent question ID, author, title, description, creation datetime, image URL, upvotes, and downvotes.

5. Diagrams

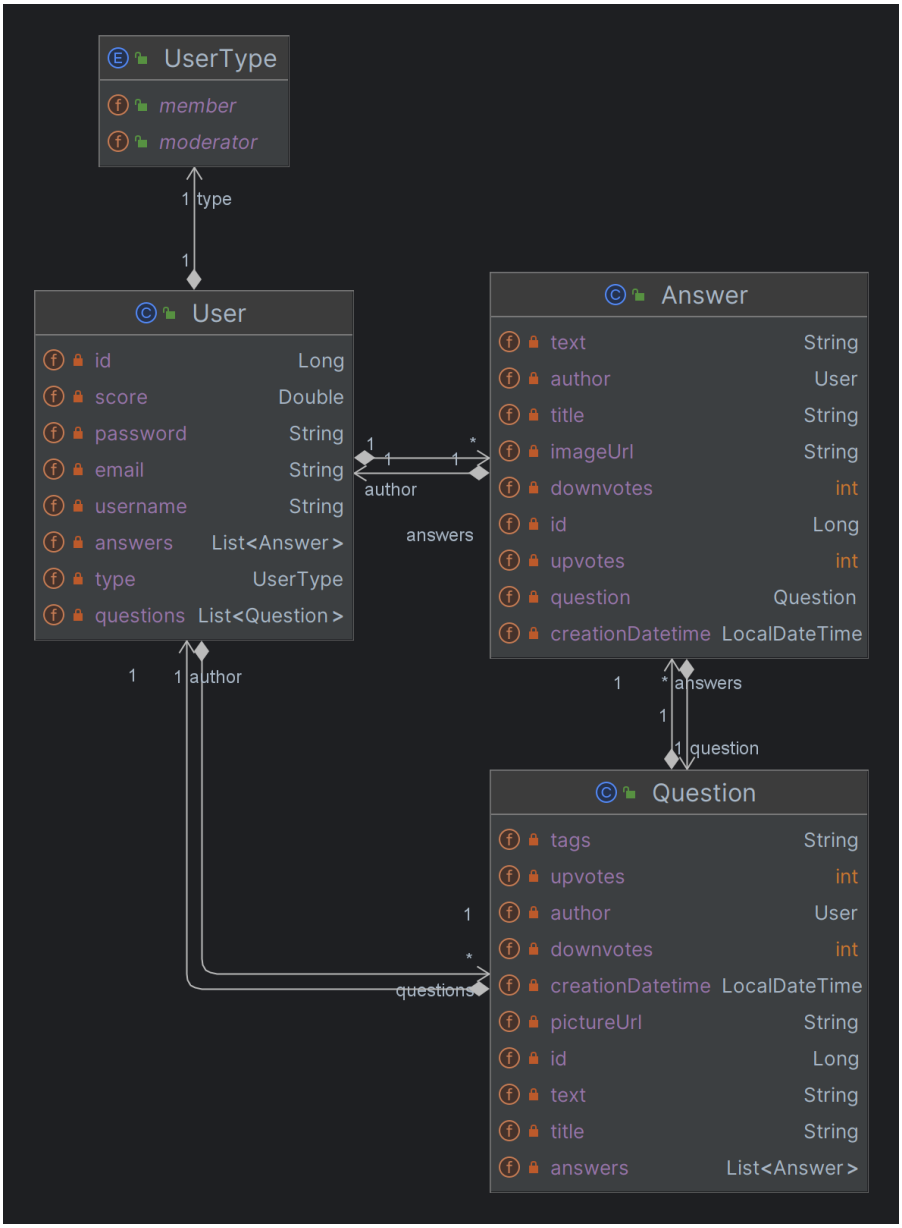
- Diagram of dependencies in the application:



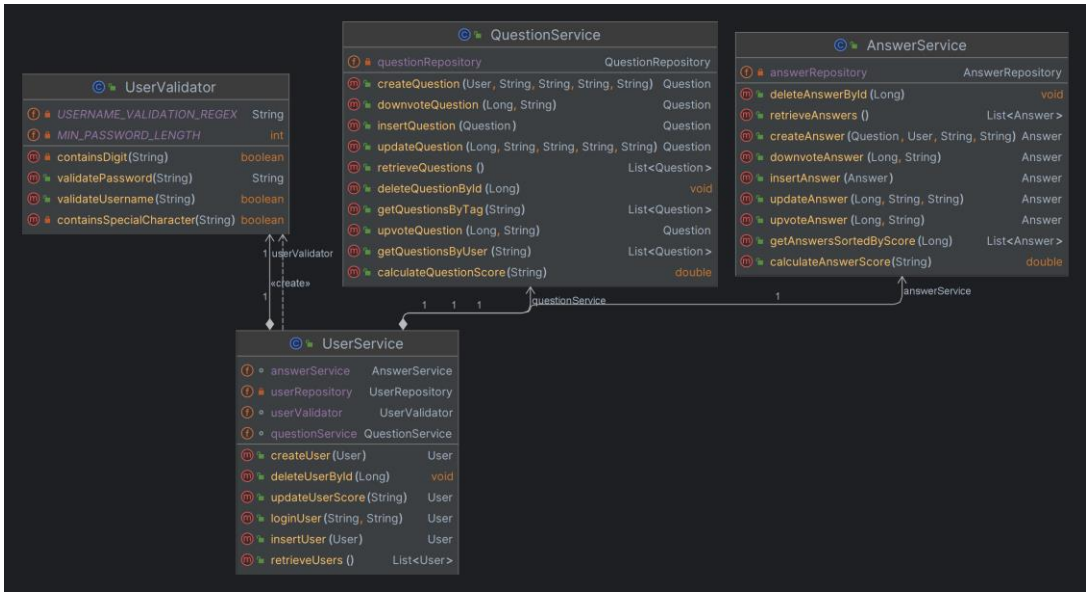
- diagram of classes belonging to the controller package:



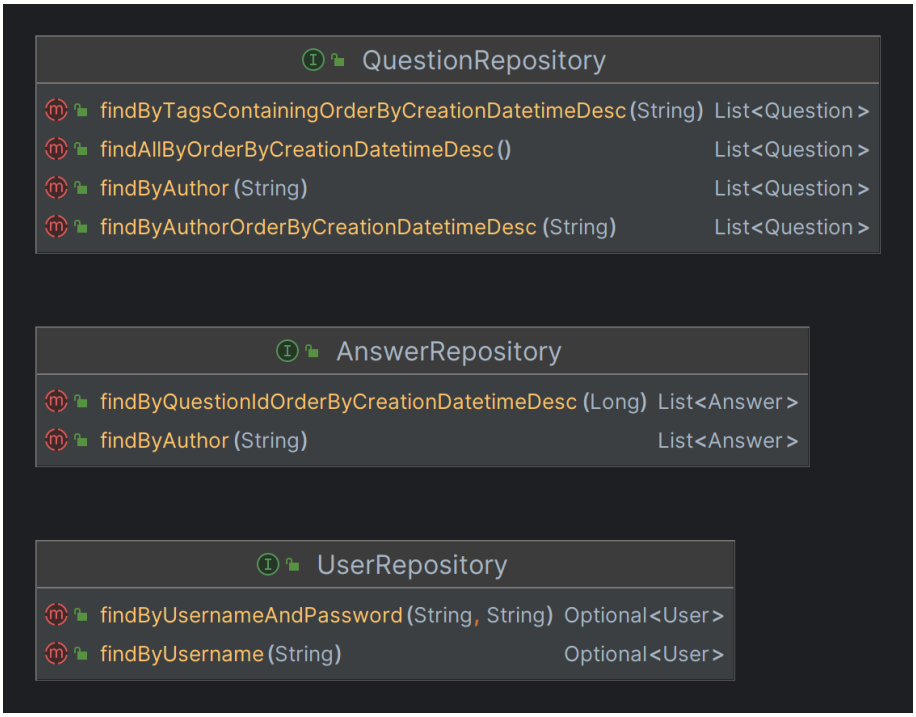
- diagram of classes belonging to the entity package:



- diagram of classes belonging to the service package:



- diagram of interfaces belonging to the repository package:



6. Bibliography

- https://www.baeldung.com
- https://www.geeksforgeeks.org
- https://www.javatpoint.com
- https://stackoverflow.com