

1. Go to the Spring Initializr website and add the following dependencies into our project.

- Lombok - to save boilerplate code
- Spring Web - for Web features
- Spring Security - for Spring Security - Encryption
- Spring Data JPA - for JPA Repository
- MySQL Java Driver - for connection to database
- Java Mail Sender - for SMTP mail facilities

2. We even need to add additional dependencies for JWT authentication and displaying the relative duration.

For JWT - add JJWT api , JJWT impl , JJWT Jackson

For Relative duration - we use the Github's timeago, we also have to use Kotlin's runtime libraries , as timeago is a Kotlin library.

3. Configure the application.properties - to include the database properties and SMTP server.

There will be account activation emails and comment notification emails to the users , hence we are using SMTP server to send emails.

We are making use of a fake SMTP server called MailTrap. We need to sign in to MailTrap with our account , and we get our password and username by default.

4. Database Schema Diagram

a. User -----> user_id , created , email , enabled , password , username

b. Post -----> post_id , created_date , description , post_name , url , vote_count, id, user_id

c. Vote -----> vote_id , vote_type, post_id , user_id

d. subreddit_posts -----> subreddit_id, posts_posts_id

e. subreddit -----> id, created_date , description, name , user_user_id

f. comment -----> id, created_date , text , post_id ,user_id

g. token -----> id, expiry_date , token , user_user_id

h. user_subreddits -----> user_user_id , subreddit_id

i. user_comments -----> user_user_id , comments_id

j. user_posts -----> user_user_id, posts_post_id

k. refresh_token -----> created_date, id , token

5. Creating Domain Entities

Users can create Subreddits and Posts

Other users can add comments to the Posts and Vote (Either UPVOTE , DOWNVOTE)

Lombok ---- @Data , @AllArgsConstructor , @NoArgsConstructor - takes care to generate getter , setter , constructor etc at compile time.

6. User Entity

Points to be Noted : @Data , @AllArgsConstructor , @NoArgsConstructor, @Entity , @Id, @Generated

@NotBlank ----- The annotated element must not be null and must contain at least one non-whitespace character.

@Email ----- The string has to be a well-formed email address. Exact semantics of what makes up a valid email address are left to Bean Validation providers.

@NotEmpty ----- The annotated element must not be null nor empty.

7. Post Entity

Points to be Noted : @Data , @AllArgsConstructor , @NoArgsConstructor, @Entity , @Id, @Generated, @Builder,@Nullable,@Lob

@Builder- Lombok's @Builder annotation is a useful technique to implement the builder pattern that aims to reduce the boilerplate code.

@Lob - Specifies that a persistent property or field should be persisted as a large object to a database-supported large object type.

8. Subreddit Entity

9. Vote Entity

10 Comment Entity

11. VoteType enum

12. Verification Token

13. Notification Email

14. Configure different Repositories

a. UserRepository - has one method findByUsername - returns an Optional.

- b. PostRepository - has 2 methods - findAllBySubreddit(Subreddit subreddit) , findBUser(User user).
- c. Comment Repository - has 2 methods - findByPost(Post post), findAllByUser(User user)
- d. Subreddit Repository - has 1 method , findByName(String subredditName)
- e. Vote Repository - has one method , findTopByPostAndUserOrderByVoteIdDesc(Post post,User currentUser)
- f. VerificationToken - has 1 method - findByToken(String token)