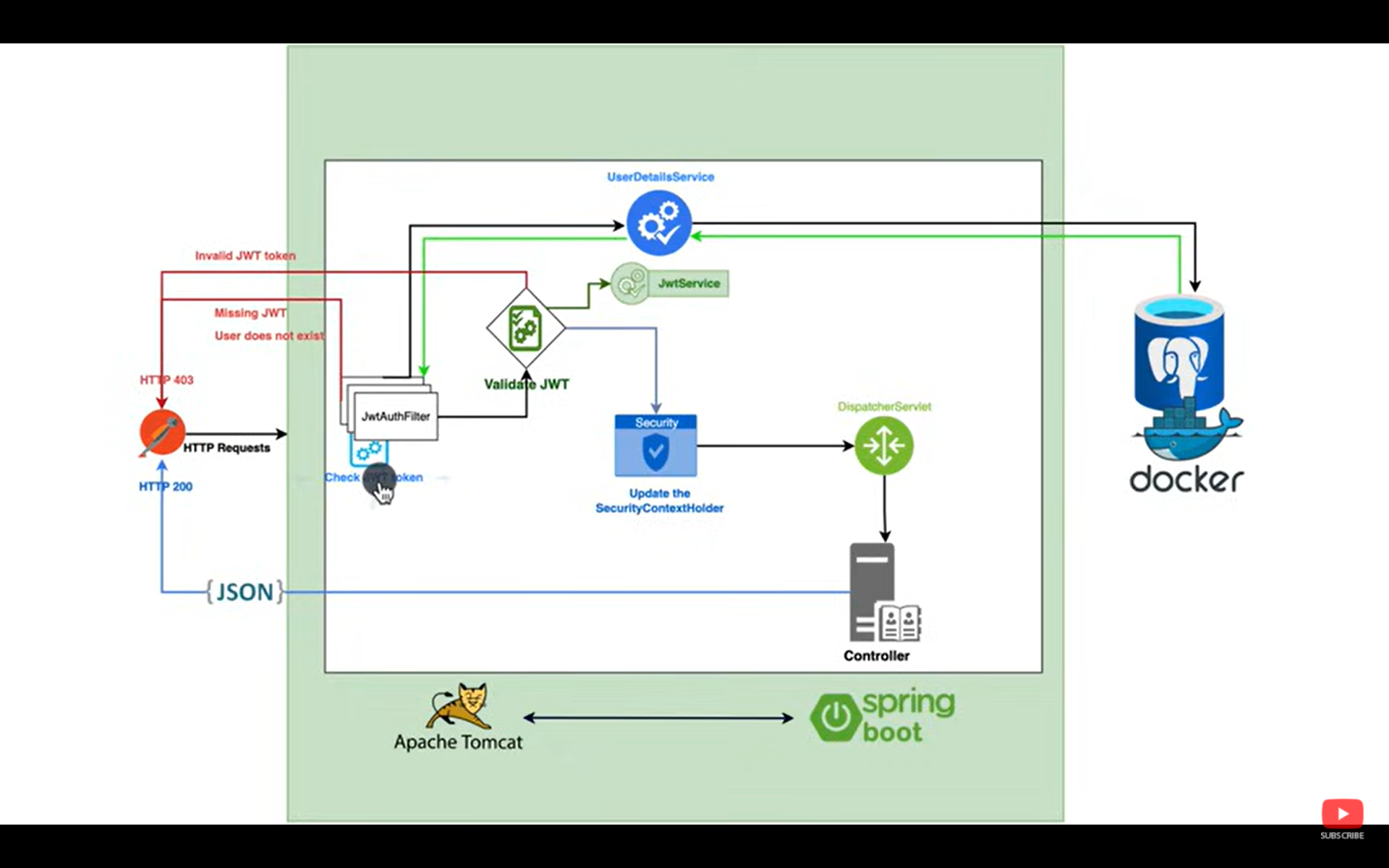
***Spring Boot 3 + Spring Security 6 - JWT Authentication and Authorisation***



First when the users sends a request..

It goes to JwtAuthFilter..

The filter gets the username and password from the token and checks with the database whether the user exists..

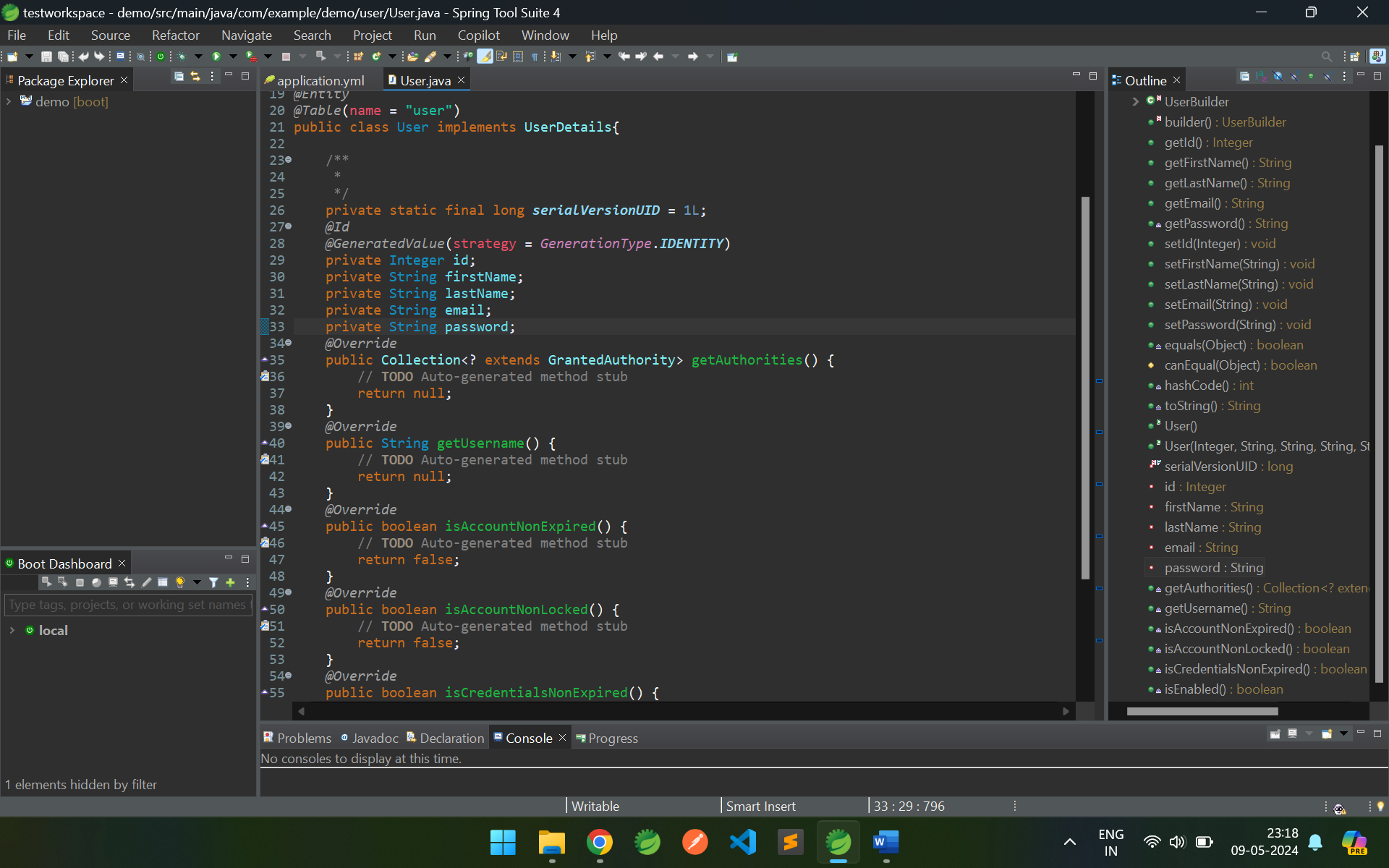
If does not exist.. then it throws 403 error.

If the user exists.. then it validates the JWT token..

If token is okay.. then it sends positive response..

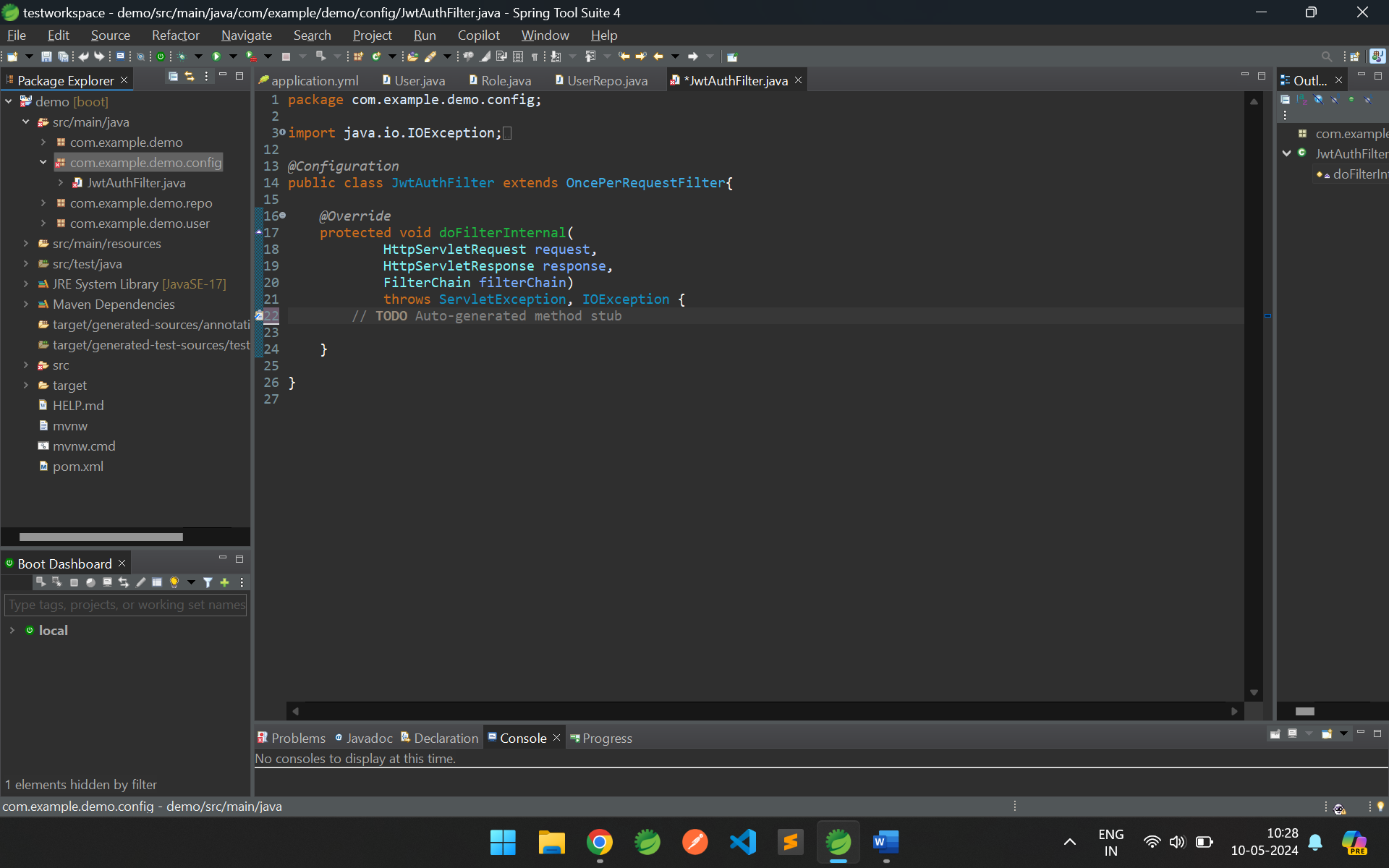
Else fi the token is wrong or token expired.. then sends 403 code.

Create a class User implementing an interface

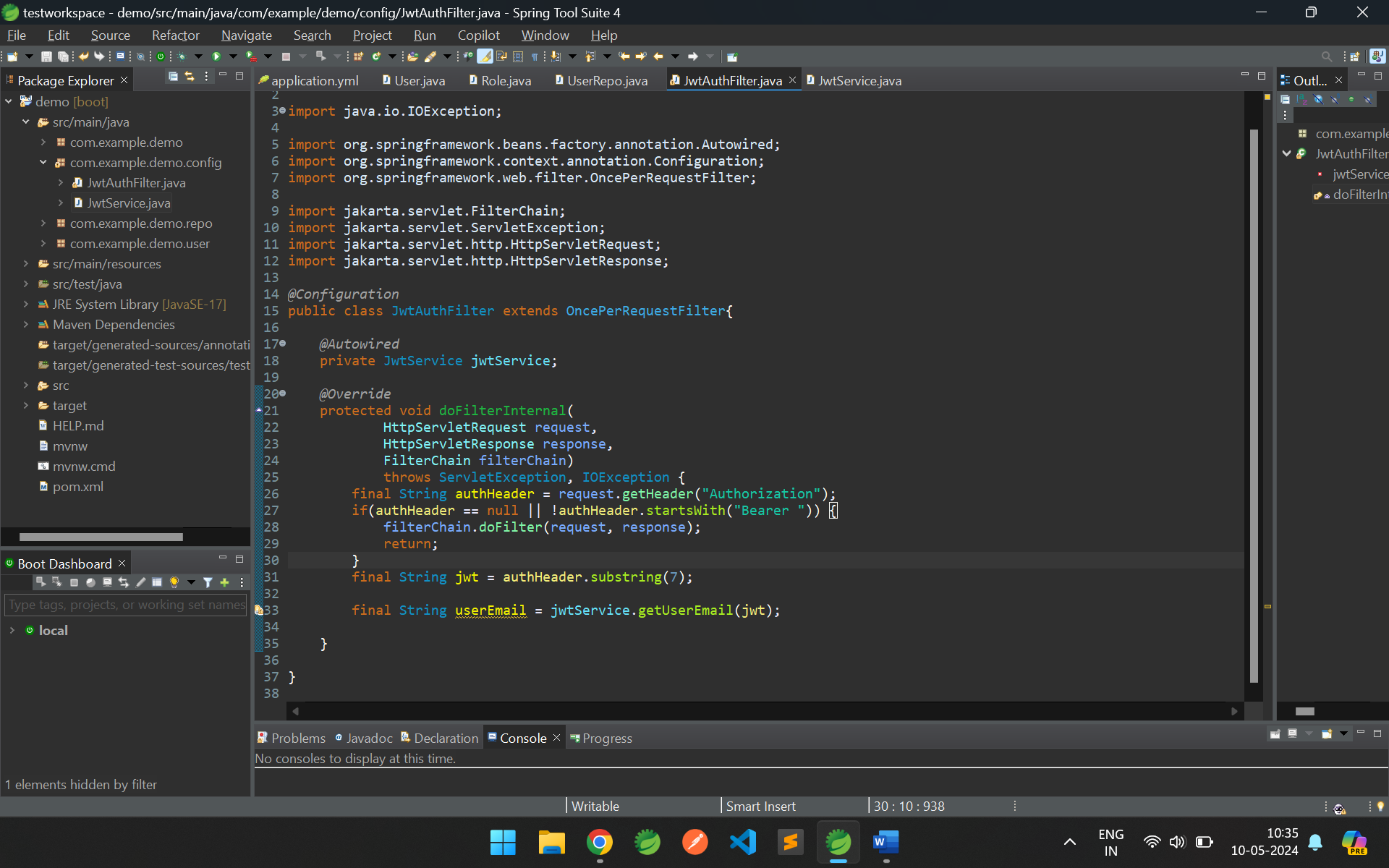


Then create a repository interface by extending JpaRepository, as usual.

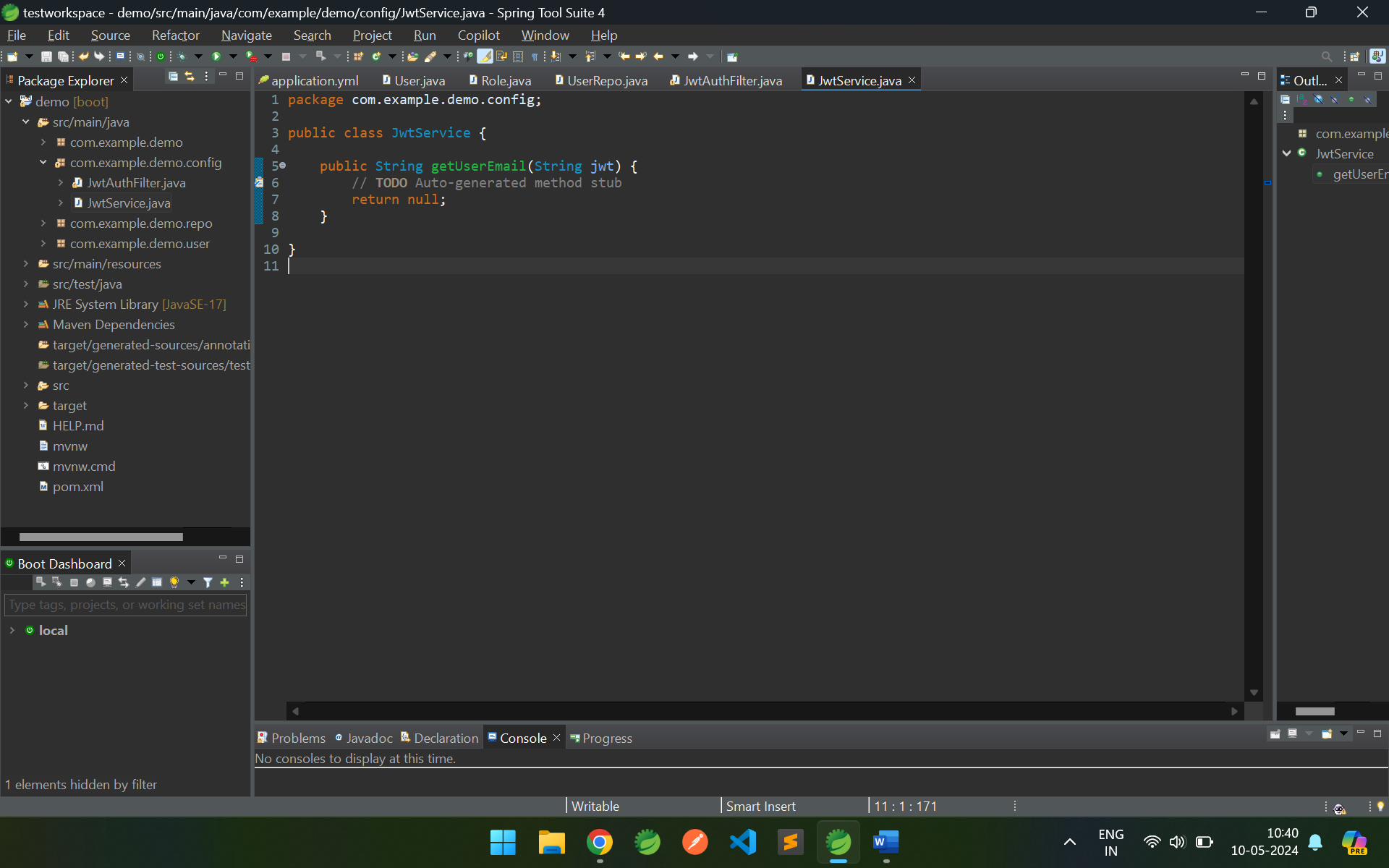
Then create a JqtFilter confifg class extending OncePerRequestFilter class.



Adding some code to extract usermail from the JWT token,



For the above we have to create JwtService class.



Now we have to add few dependencies for working with JWT stuff.

<dependency>

<groupId>io.jsonwebtoken</groupId>

<artifactId>jjwt-api</artifactId>

<version>0.11.5</version>

</dependency>

<dependency>

<groupId>io.jsonwebtoken</groupId>

<artifactId>jjwt-impl</artifactId>

<version>0.11.5</version>

</dependency>

<dependency>

<groupId>io.jsonwebtoken</groupId>

<artifactId>jjwt-jackson</artifactId>

<version>0.11.5</version>

</dependency>

Explore: <https://jwt.io/>