

⇒ Bült Bitly project

We created a new project using spring initializer

↳ added some dependencies as well

- 1) Spring JPA
- 2) Spring Web
- 3) my SQL
- 4) Lombok

↳ then created some files

Main file containing @Spring Boot Application

↳ mentioning a spring boot app

Then created a new package models

↳ User.java containing

@Entity :- used to mark a class as a JPA entity which means it represents a table in database

@Data :- comes from Lombok and will generate getters(), setters(), equals(), hashCode() etc. automatically

@Table(name = "users") :- JPA annotation

used to add table name in a particular MySQL database.

@Id :- to mark a field as primary key

@GeneratedValue :- normally used with Id to automatically generate a primary key

↳ ClickEvent.java containing

@Entity , @Data , @Id , @GeneratedValue

④ ManyToOne :- current class is many and field is one

means many clickevents is mapped to one URL

④ JoinColumn:- it used current field as a foreign key to connect with other entity

↳ UrlMapping.java containing

@Entity , @Data , @Id , @GeneratedValue

④ ManyToOne (with user) :- means

many Urlmapping entity are connected with one user only

④ OneToMany (with clickevents) :- means

one url mapping entity are joined with many click events (is a list of click event type) .

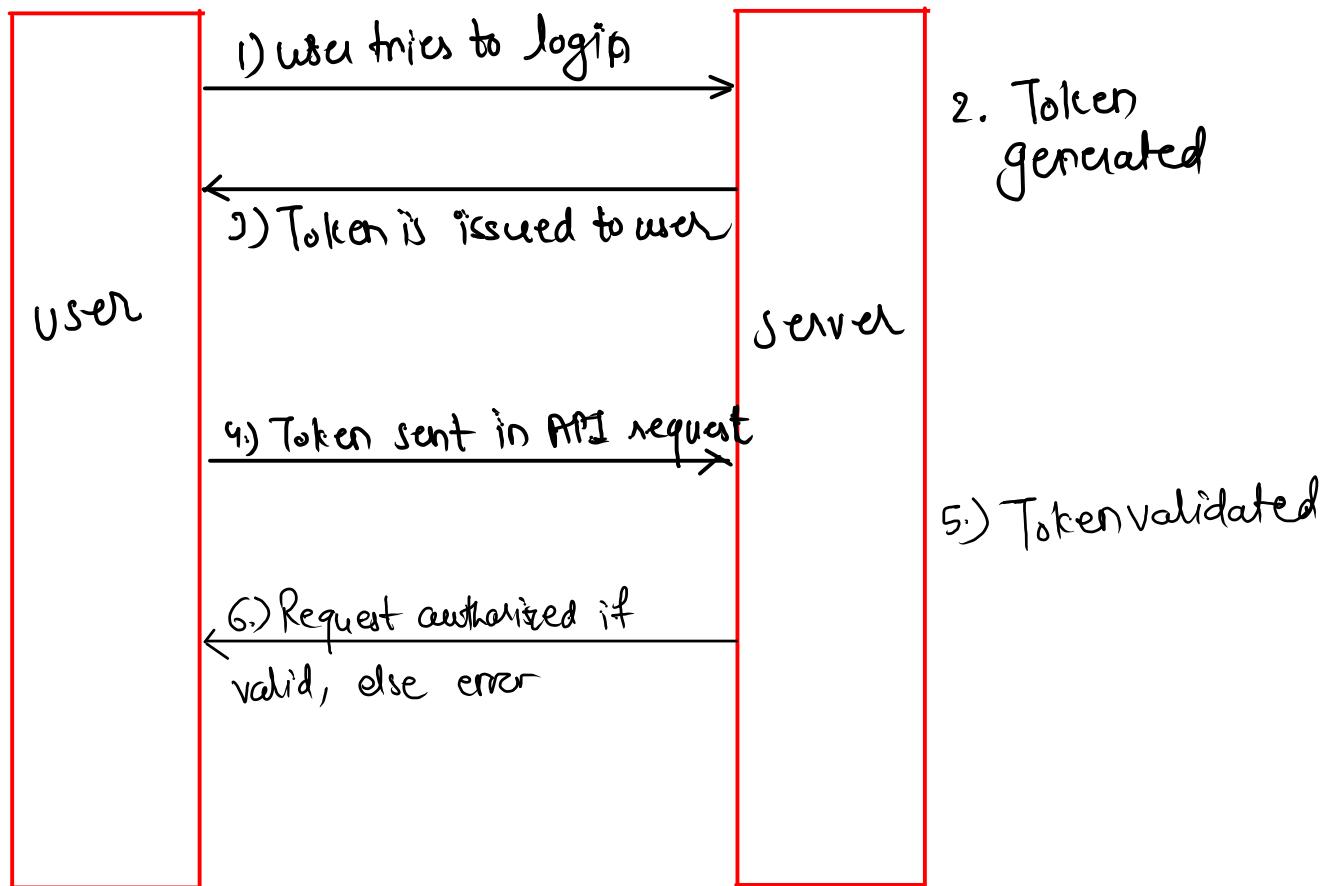
Then, we have created a sql connection with DB

used the link that we have added in applications.properties
used username and password.

and ran the application once to refresh and see that tables are now created automatically in SQL workbench magically -

⇒ JWT Tokens (Json Web Tokens)

↳ Type of authentication



Note:- Token are send using HTTP authorization header

Ex:- eg. ----- . eg JZ ----- . SfIKx -----
↑ ↑ ↑
header payload signature verification

Some files needs to be created to achieve this

JwtUtils

Jwt Authentication filter

Jwt Authentication Response

Security Config

Note:-

Spring Security provides the tools and structure to authenticate and authorize user

Spring security has its own inbuilt implimentation of user which would work, but in lot of cases custom is needed.

UserDetailsServiceImpl

UserDetailsImpl

define:-

- 1) JwtAuthenticationFilter :- Intercepts HTTP requests and extract JWT Tokens.
- 2) JwtUtils :- generate, validates tokens and provides helper methods
- 3) UserDetailsServiceImpl :- fetches user details from the DB
- 4) UserDetailsImpl :- Provides a spring security - compatible representation of the user for authentication and authorization

