

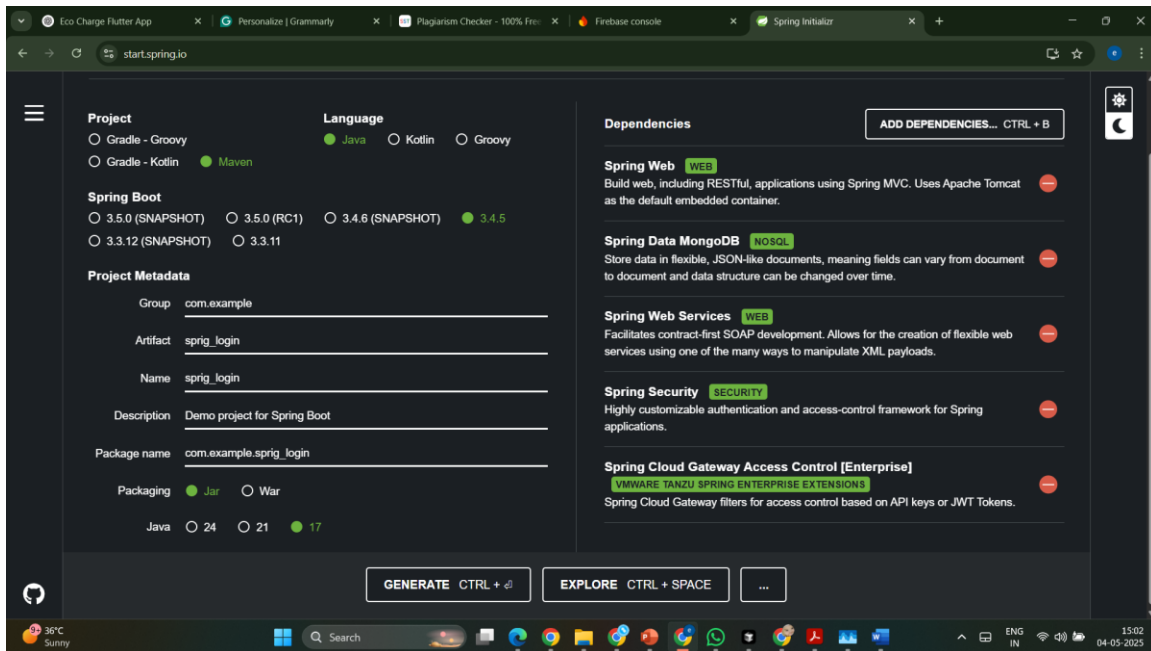
Step-by-Step Guide to Create Login and Signup Modules in Spring Boot with MongoDB

1. Project Setup

Link: <https://start.spring.io/>

Create a Spring Boot project using Spring Initializr or IntelliJ Idea Paid Version with dependencies:

- Spring Web
- Spring Security
- Spring Data MongoDB
- jjwt (for JWT token authentication).

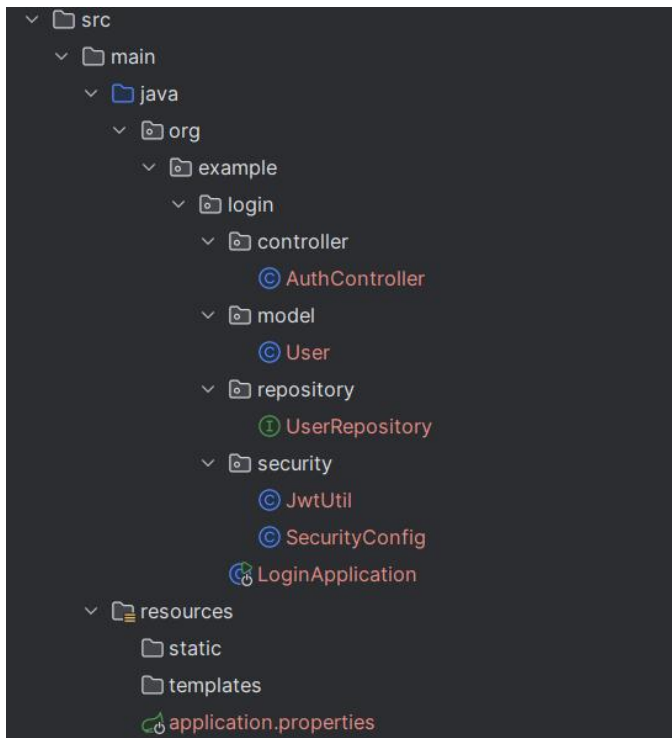


2. Package Structure

Organize your code with the following structure:

- config
- controller
- dto
- model
- repository

- service
- security



3. Add Dependencies in pom.xml

Ensure `pom.xml` contains:

- spring-boot-starter-web
- spring-boot-starter-security
- spring-boot-starter-data-mongodb
- jjwt-api, jjwt-impl, jjwt-jackson
- spring-boot-starter-validation

4. Configure application.properties

Add MongoDB connection details:

...

server.port=8080

MongoDB Connection

spring.data.mongodb.uri=mongodb+srv://shyamsundar41550:shyam123@cluster0.ketn4.

mongodb.net/?retryWrites=true&w=majority&appName=Cluster0

spring.data.mongodb.database=login_db

5. Create Repository

Create `UserRepository` extending `MongoRepository<User, String>`.

Include methods like this,

```
Optional<User> findByUsername(String username);
boolean existsByUsername(String username);
}
```

6. Configure Password Encoder

Create a `@Bean` in `SecurityConfig` to return `PasswordEncoder` using `BCryptPasswordEncoder`.

7. Implement JWT Utilities

- Generate JWT token on login.
- Validate token on protected requests.
- Create `JwtUtil` and `JwtFilter` classes.

8. Create Controllers and save it in MongoDB

Create `AuthController`:

- `@PostMapping("/api/auth/signup")` for registration and save the user
- `@PostMapping("/api/auth/login")` for login and authenticate with mongodb
- Return token on success

09. Configure Spring Security

- Allow `/api/auth/**` URLs to be accessed publicly
- Use `JwtAuthenticationFilter` for secured requests
- Disable CSRF, use stateless session management

10. Test the Application

Using Postman:

- POST to `/api/auth/signup` with name, email, password
- POST to `/api/auth/login` with email, password
- Use returned JWT in `Authorization: Bearer <token>` header to access secure APIs

Output from springBoot Tomcat Server :

```
l, topologyVersion=TopologyVersion{processId=681209a4303995d6989a197e, counter=3}, lastWriteDate=Sun May 04 15:05:30 IST 2025, lastUpdateTimeNanos=106329922097600}
2025-05-04T15:05:32.093+05:30 INFO 23972 --- [ngodb.net:27017] org.mongodb.driver.cluster : Discovered replica set primary cluster0-shard-00-01.ketn4.mongodb.net:270
17 with max election id 7fffffff00000000000000015 and max set version 1
2025-05-04T15:05:32.133+05:30 INFO 23972 --- [ngodb.net:27017] org.mongodb.driver.cluster : Monitor thread successfully connected to server with description ServerDe
scription{address=cluster0-shard-00-00.ketn4.mongodb.net:27017, type=REPLICA_SET_SECONDARY, cryptd=false, state=CONNECTED, ok=true, minWireVersion=0, maxWireVersion=25, maxDocument
Size=16777216, logicalSessionTimeoutMinutes=30, roundTripTimeNanos=325393400, minRoundTripTimeNanos=0, setName='atlas-cth37i-shard-0', canonicalAddress=cluster0-shard-00-00.ketn4.m
ongodb.net:27017, hosts=[cluster0-shard-00-02.ketn4.mongodb.net:27017, cluster0-shard-00-01.ketn4.mongodb.net:27017, cluster0-shard-00-00.ketn4.mongodb.net:27017], passives=[], arb
iters=[], primary='cluster0-shard-00-01.ketn4.mongodb.net:27017', tagSet=TagSet{[Tag{name='availabilityZone', value='aps1-az1'}, Tag{name='diskState', value='READY'}, Tag{name='nod
eType', value='ELECTABLE'}, Tag{name='provider', value='AWS'}, Tag{name='region', value='AP_SOUTH_1'}, Tag{name='workloadType', value='OPERATIONAL'}]}, electionId=null, setVersion=
1, topologyVersion=TopologyVersion{processId=681206f7ba124d107941b8b1, counter=4}, lastWriteDate=Sun May 04 15:05:30 IST 2025, lastUpdateTimeNanos=106329972998000}
2025-05-04T15:05:32.603+05:30 INFO 23972 --- [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port 8080 (http) with context path '/'
2025-05-04T15:05:32.615+05:30 INFO 23972 --- [main] org.example.Login.LoginApplication : Started LoginApplication in 4.263 seconds (process running for 4.781)
```

