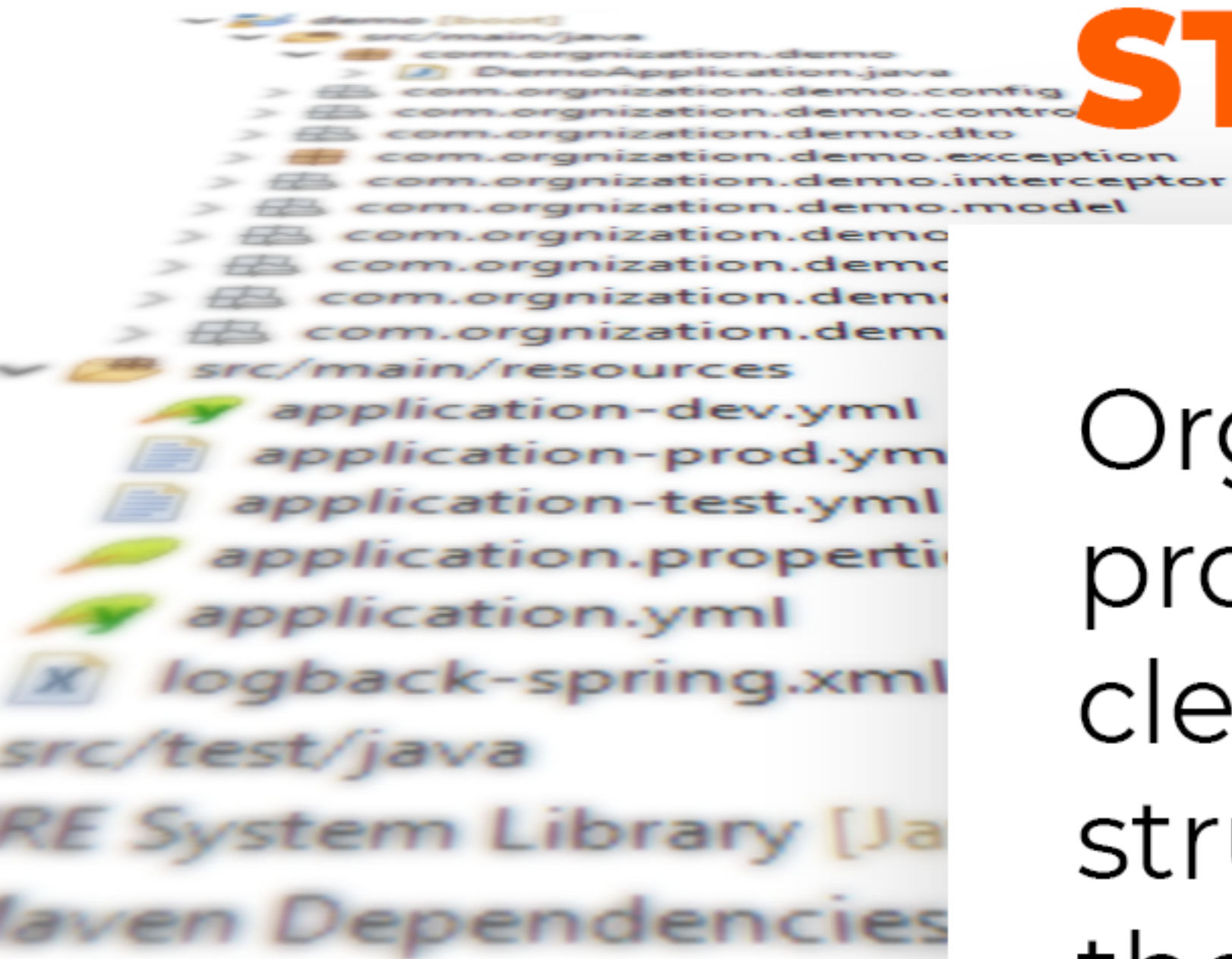




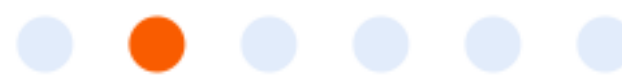
BEST PRACTICES FOR JAVA SPRING BOOT DEVELOPMENT

PROJECT STRUCTURE:

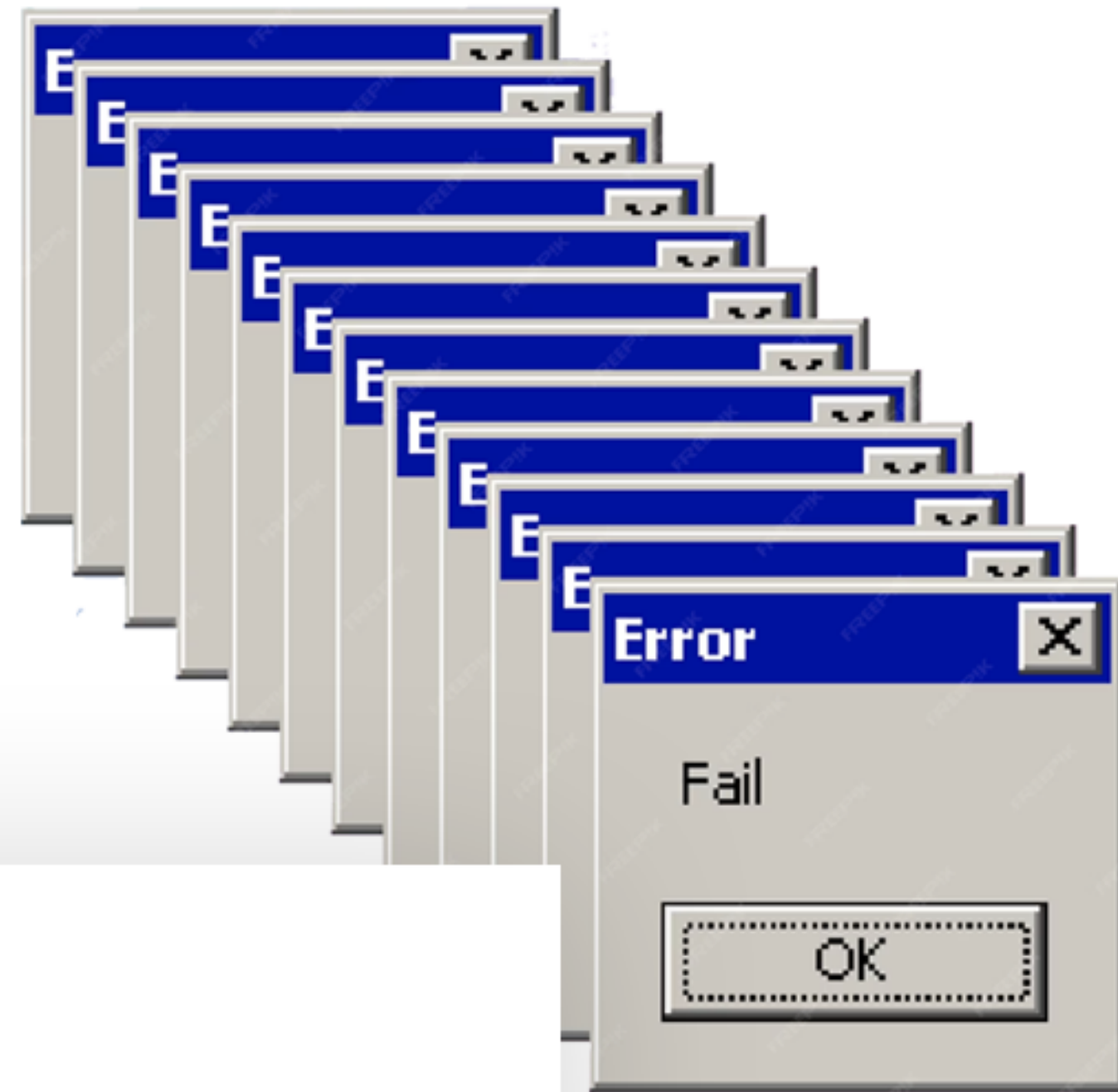


```
demo [demo]
├── src/main/java
│   ├── com.organization.demo
│   │   ├── DemoApplication.java
│   │   ├── com.organization.demo.config
│   │   ├── com.organization.demo.control
│   │   ├── com.organization.demo.dto
│   │   ├── com.organization.demo.exception
│   │   ├── com.organization.demo.interceptor
│   │   ├── com.organization.demo.model
│   │   ├── com.organization.demo
│   │   ├── com.organization.demo
│   │   ├── com.organization.demo
│   │   └── com.organization.demo
│   └── src/main/resources
│       ├── application-dev.yml
│       ├── application-prod.yml
│       ├── application-test.yml
│       ├── application.properties
│       ├── application.yml
│       └── logback-spring.xml
├── src/test/java
└── RE System Library [Java]
    └── Maven Dependencies
```

Organize your project into a clean and modular structure. Follow the standard Maven or Gradle project layout.



EXCEPTION HANDLING :



Implement custom exception handling for a clear and consistent error response. Leverage `@ControllerAdvice` for global exception handling.

TESTING:

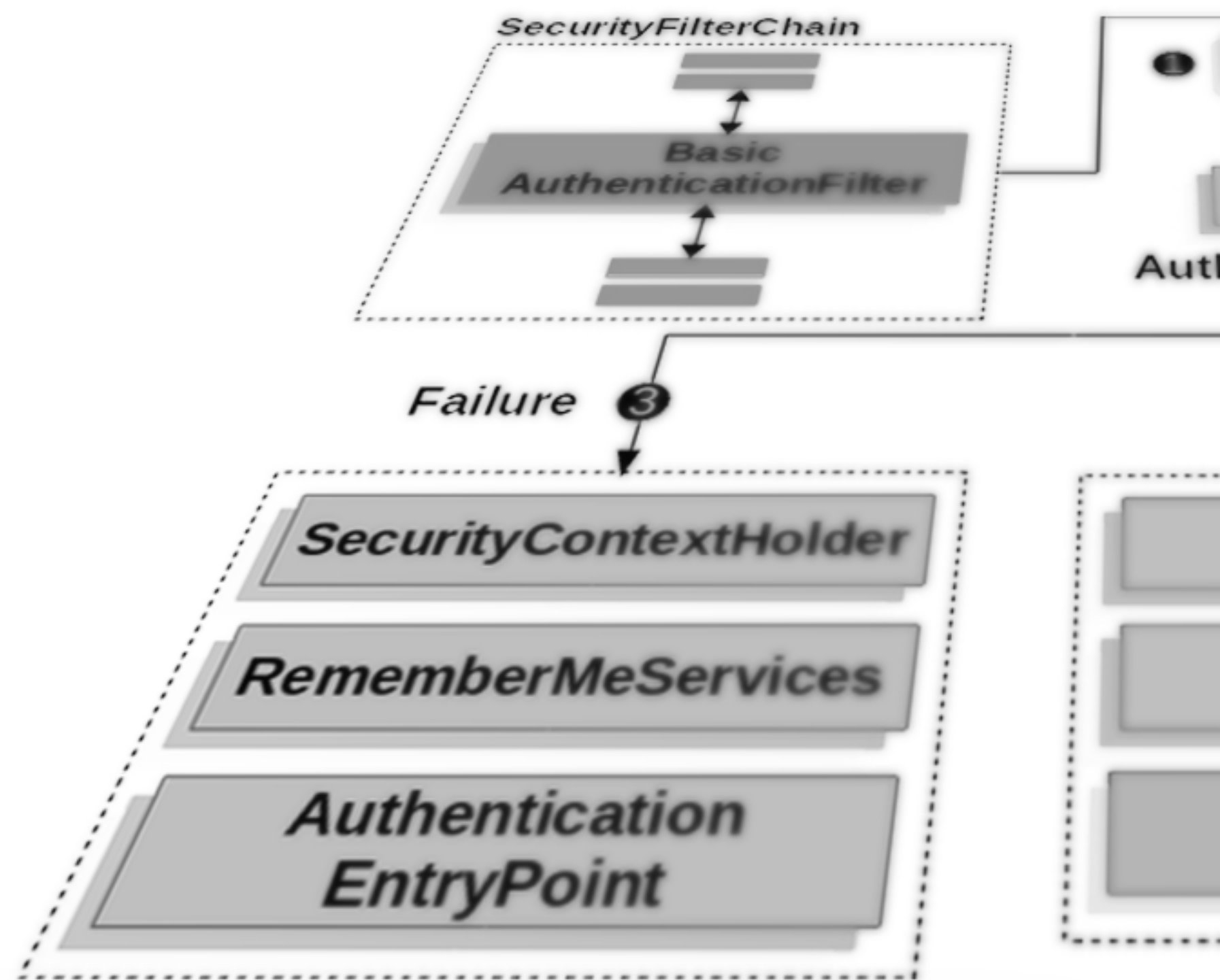


Write unit tests using JUnit and integration tests. Use tools like Mockito for mocking dependencies.

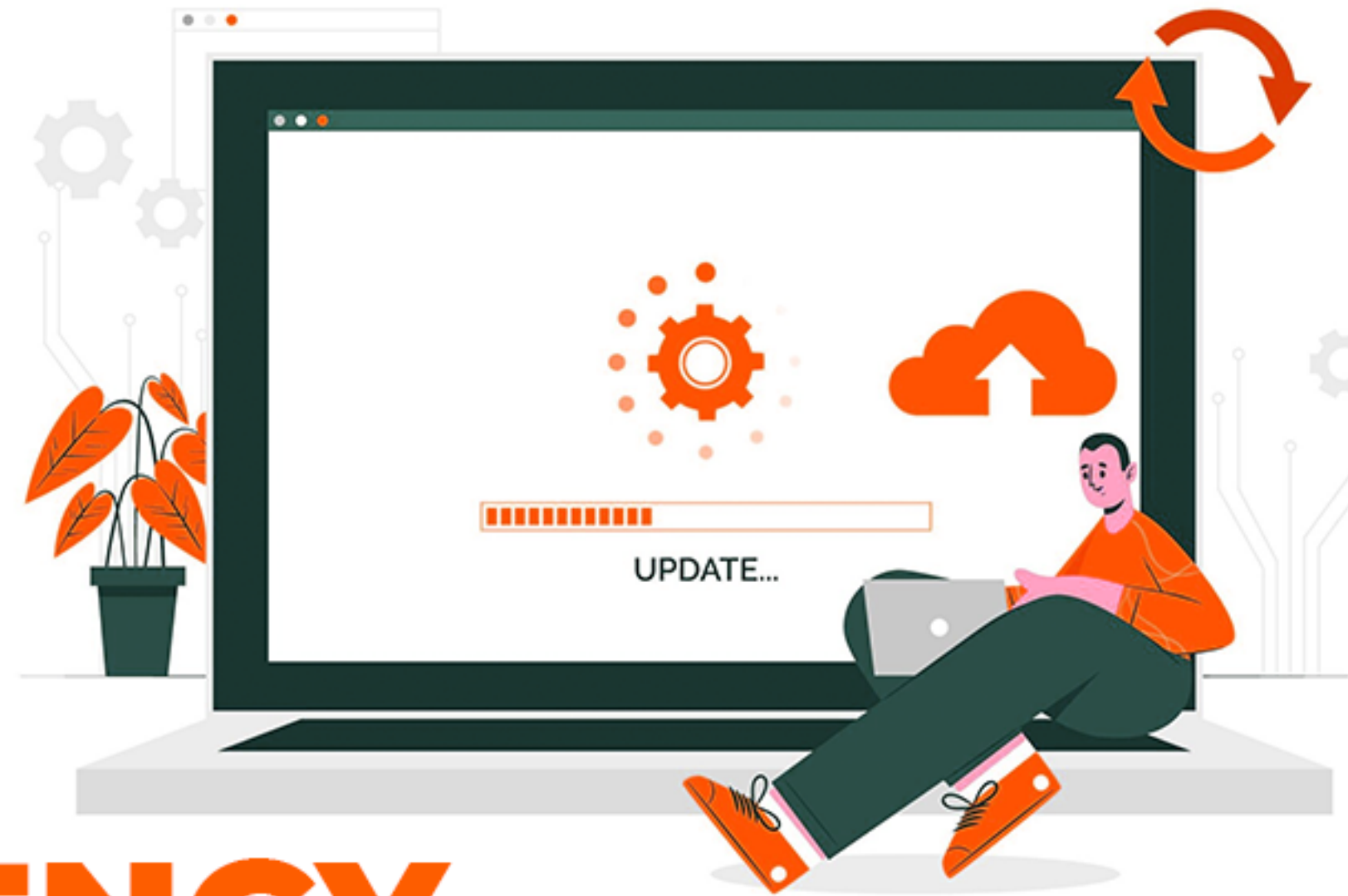




SECURITY:



Implement proper authentication and authorization using Spring Security. Keep sensitive information secure using environment-specific configuration.



DEPENDENCY MANAGEMENT:

Stay updated with the latest stable version of Spring Boot. Clearly define and manage dependencies in your build tool configuration file.