experts club

SpringSecurity: Testando endpoints com autenticação Basic Auth com Spring Security Test , Junit e Mockito

04-01-2021

Agenda

experts club

- Quem sou eu
- Objetivo
- Do que o Spring Security cuida
- Conceitos essenciais de segurança de aplicações
- Módulos do Spring Security
- Apresentação do projeto já desenvolvido tecnologias utilizadas
- Escrita dos testes unitários

Sobre mim e a minha relação com o código

experts club

- Dev backend na Ame Digital
- Microsoft mvp em developer technologies
- Community Organizer
- Book co-author
- speaker
- content creator @kamila_code
- 4 anos de xp
- https://app.rocketseat.com.br/me/kamila-de-fatima-santos-oliveira-1566497781

O que vamos desenvolver hoje?

experts club

Vamos escrever os testes unitários de todas as operações CRUD do projeto, validando cenários de credenciais válidas, usuário mockado, sem credenciais e credenciais inválidas.

Ferramentas utilizadas

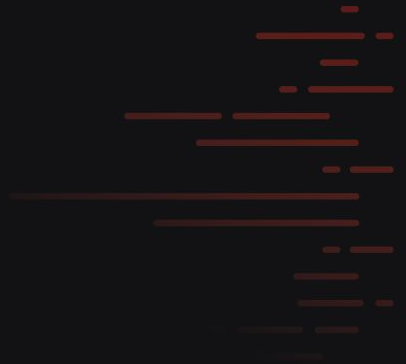
experts club

Java 11+

Maven

Spring (Web, Security e Data)

Junit, Spring Security Test e Mockito



Do que o Spring security cuida?

O Spring Security entra na parte de segurança de aplicações, podendo controlar se um usuário tem acesso a aplicação e exatamente ao que ele terá acesso.

Conceitos essenciais de segurança de aplicações

experts club

Identificação-> representação de um usuário no sistema (geralmente um usuário e senha)

Autenticação -> valida se os usuários e senhas informados correspondem aos que existem na base

Conceitos essenciais de segurança de aplicações

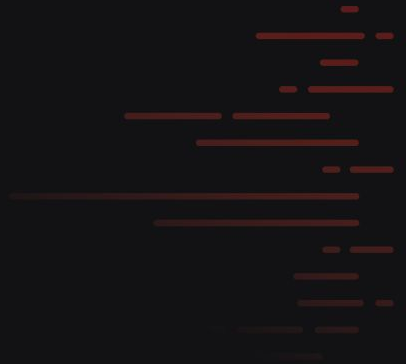
experts club

Autorização -> valida se o usuário e senha que são válidos tem permissão para executar determinada ação que está tentando executar.

Módulos do Spring Security

Core -> módulos prinicipais do Spring security

Web -> segurança para aplicações Web



Módulos do Spring Security

experts club

Config -> para fazer configurações em classes Java e arquivos XML

LDAP -> para integração com LDAP identity providers

OAuth 2.0 Core -> parte central e obrigatória para utilização do Oauth authorization e do OpenId

Módulos do Spring Security

OAuth 2.0 Client -> client para utilização do Oauth authorization e do OpenId

OAuth 2.0 Jose-> provê suporte para o JOSE (Javascript Object Signing and Encryption)

Módulos do Spring Security

OAuth 2.0 Resource Server -> para servidores de autenticação OAuth

Spring Security Test-> para testes em endpoints que usam autenticação

CSRF

Esse tipo de ataque envia solicitações desautorizadas de um usuário no qual o website confia.

Basic Auth

experts club

Usa usuário e senha para se autenticar (aba Basic Auth no Postman ou Header Authorization Basic(com o usuário e senha codificados em base 64)

Testes unitários

Os testes unitários tem por objetivo testar pequenas unidades funcionais de um código

Testando cada unidade de modo separado, podemos dar a devida atenção a cada parte do código e identificar mais facilmente erros.

Testes unitários

Quando possuem uma boa descrição/títulos , também são uma ótima fonte de conhecimento de um sistema.

Teste de integração

Vão testar como as partes do sistema atuam em conjunto, no nosso caso vamos testar a autenticação via controller -> que chama a service -> que chama o repository - que chega ao banco de dados.

O que vamos usar para escrever os testes

experts club

Junit: um dos frameworks mais utilizados para escrever testes em Java

Spring Security Test: Módulo do Spring Security para escrita de testes

Mockito : para criar mocks de classes e objetos

Cenários de teste que iremos cobrir /POST

experts club

Criação de estudante com usuário válido - sucesso

Criação de estudante com usuário invalido - falha - 401

Criação de estudante com senha invalida - falha - 401

Criação de estudante com mock user - sucesso

Criação de estudante sem credencial - falha -401

Cenários de teste que iremos cobrir /GET

experts club

Esse endpoint não usa autenticação então só vamos validar um cenário:

Listagem de todos os estudantes sem autenticação - sucesso

Cenários de teste que iremos cobrir /get/id

experts club

Buscar estudante pelo id com usuário válido - sucesso

Buscar estudante pelo id com usuário invalido - falha - 401

Buscar estudante pelo id com senha invalida - falha - 401

Buscar estudante pelo id com mock user - sucesso

Buscar estudante pelo id sem credencial - falha -401

Cenários de teste que iremos cobrir /put

experts club

Atualizar estudante pelo id com usuário válido - sucesso

Atualizar estudante pelo id com usuário invalido - falha - 401

Atualizar estudante pelo id com senha invalida - falha - 401

Atualizar estudante pelo id com mock user - sucesso

Atualizar estudante pelo id sem credencial - falha -401

Cenários de teste que iremos cobrir /delete

experts club

Excluir estudante pelo id com usuário válido - sucesso

Excluir estudante pelo id com usuário invalido - falha - 401

Excluir estudante pelo id com senha invalida - falha - 401

Excluir estudante pelo id com mock user - sucesso

Excluir estudante pelo id sem credencial - falha -401

experts club

Obrigada !

Kamila Santos
https://youtube.com/Kamilacode
https://github.com/Kamilahsantos
https://www.linkedin.com/in/kamila-santos-oliveira/
https://www.instagram.com/kamila_code/