

Teste Prático: Desenvolvedor Java Sênior

Objetivo: Criar uma aplicação Java utilizando Spring Boot que implemente um sistema de gerenciamento de tarefas com segurança OAuth2 e caching de tokens em Redis. A aplicação deve ser capaz de criar, ler, atualizar e deletar tarefas, além de buscar tarefas por status.

Requisitos do Projeto:

1. Backend:

- Utilizar Spring Boot para criar a API.
- Implementar endpoints RESTful para:
 - Criar uma nova tarefa.
 - Listar todas as tarefas.
 - Atualizar uma tarefa existente.
 - Deletar uma tarefa.
 - Buscar tarefas por status (concluída/não concluída).

2. Banco de Dados:

- Utilizar MongoDB como banco de dados NoSQL.

3. Programação Reativa:

- Utilizar Spring WebFlux para programação reativa.

4. Integração com AWS:

- Implementar uma funcionalidade que utilize AWS S3 para armazenar arquivos relacionados às tarefas (por exemplo, anexos).
- Utilizar AWS Lambda para processar eventos quando uma tarefa é criada.

5. Segurança:

- Implementar OAuth2 para autenticação e autorização.
- Configurar o armazenamento de tokens de acesso em um servidor de cache Redis.

6. Versionamento:

- Utilizar Git para versionamento do código.
- Criar um repositório público ou privado no GitHub e compartilhar o link.

7. Docker:

- Criar um Dockerfile para containerizar a aplicação.
- Criar um docker-compose.yml para facilitar o setup do ambiente de desenvolvimento, incluindo o MongoDB e o Redis.

8. Documentação:

- Adicionar documentação no README.md explicando como configurar e executar a aplicação, incluindo instruções para rodar os containers Docker.
- Documentar a API utilizando Swagger.

9. Testes:

- **Testes Unitários:**
 - Implementar testes unitários abrangentes para todas as funcionalidades principais da aplicação.
 - Utilizar JUnit e Mockito para os testes.
- **Testes de Integração:**
 - Implementar testes de integração para verificar a interação entre os componentes do sistema.
 - Garantir que os testes de integração cubram casos de uso comuns e cenários críticos.
 - Documentar como executar os testes no README.md.

Entregáveis:

- Código-fonte completo da aplicação versionado no GitHub.
- Envie o link do repositório GitHub com o projeto finalizado.
- Dockerfile e docker-compose.yml configurados corretamente.
- README.md com todas as instruções necessárias.
- Um arquivo de exemplo para teste das funcionalidades da AWS S3 e Lambda.

Detalhamento de Atividade Adicional:

Segurança e Cache de Tokens:

1. Configuração do OAuth2:

- Configurar o Spring Security com suporte a OAuth2.
- Implementar endpoints para autenticação e autorização de usuários.

2. Integração com Redis:

- Configurar o Redis como servidor de cache.
- Armazenar tokens de acesso gerados durante a autenticação no Redis.
- Configurar a expiração e a renovação de tokens conforme necessário.

3. Autorização de Endpoints:

- Proteger os endpoints RESTful utilizando as anotações de segurança do Spring Security.
- Garantir que apenas usuários autenticados e autorizados possam acessar os endpoints.

4. Testes de Segurança:

- Implementar testes unitários e de integração para garantir que a segurança esteja funcionando conforme o esperado.