Criação e deployment de uma aplicação de autenticação - Planeamento

Sistemas de Computação na Cloud

Grupo 6

Carlos Bruno Machado Martins – 18836

João Ricardo Pinto Azevedo - 18845

Professor

Miguel Lopes

Ano letivo 2022/2023

Mestrado em Engenharia Informática

Escola Superior de Tecnologia

Instituto Politécnico do Cávado e do Ave

Índice

Arquitetura global

O objetivo é desenvolver um micro serviço de autenticação que sirva para suporte para a gestão de acessos e permissões de outros micro serviços.

De modo a que este seja um serviço de autenticação robusto devem ser garantidas certas condições tais como:

A autenticação deve expirar, isto é um utilizador não deve ter a sua sessão a ativa de forma permanente. Para isto iremos utilizar a biblioteca JSON Web Token.

Implementaremos também um mecanismo de proteção contra ataques de *brute-force* e, para isso, temos definidas duas abordagens que necessitam de alguma ponderação. Uma delas é a utilização do reCAPTCHA que é um serviço disponibilizado pela Google e azoutra passa pela implementação um mecanismo que bloqueie as tentativas de login durante um certo tempo após um número de tentativas falhadas.

Além disso, também deve ser assegurada o armazenamento das credenciais de forma segura e, para tal, será utilizada a biblioteca bcrypt que, basicamente, encripta qualquer tipo de dados.

Relativamente à estrutura de logs será composta por tabelas que contém toda a atividade relativa ao serviço de autenticação.

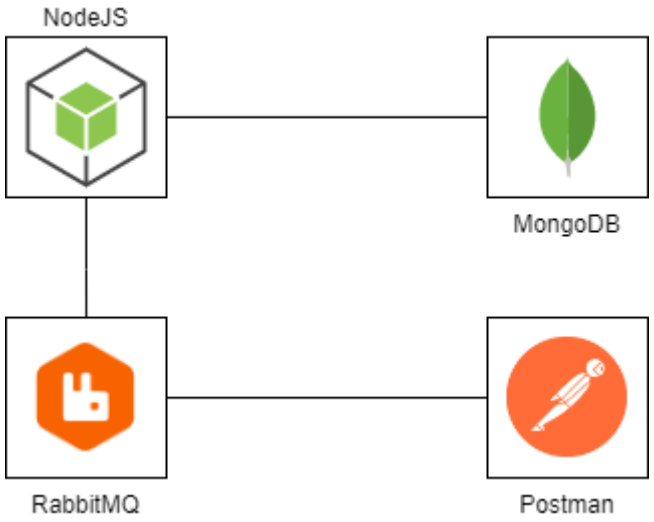

Figura 1 - Arquitetura do micro serviço

Na figura 1 é possível identificar a arquitetura global do micro serviço.

Tecnologias a utilizar

O micro serviço é composto por uma base de dados MongoDB que é do tipo não relacional que guarda dados em forma de documento. Na figura 2 encontra-se a estrutura da base de dados.

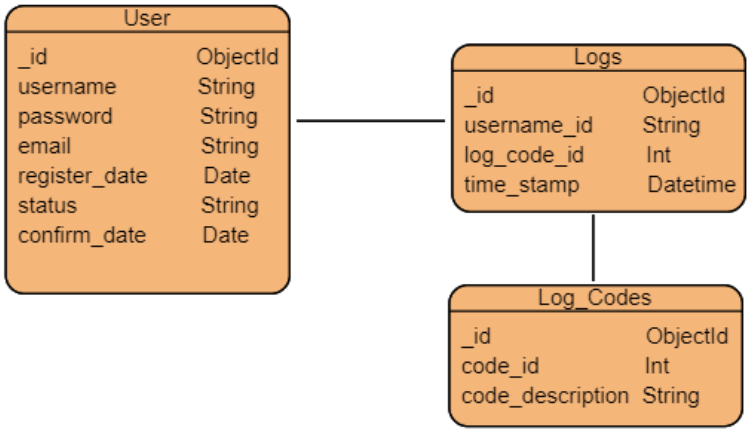

Figura 2 - Estrutura da base de dados

A base de dados é composta por uma tabela *User* que guarda todo o tipo de informações relativamente ao utilizador e por duas tabelas de *logs*. A tabela *Log Codes* guarda todos os tipos de log possíveis e o respetivo código. Já a tabela *Logs* apresenta a atividade de todos os utilizadores.

O servidor vai ser desenvolvido utilizando NodeJS tirando proveito de algumas bibliotecas mencionadas anteriormente e, numa fase inicial, será utilizado Postman como cliente.

Por fim, o RabbitMQ será utilizado de modo a manter uma lista com a ordem de todos os pedidos realizados.

Endpoints necessários

Numa primeira abordagem os endpoints a desenvolver serão os seguintes:

- Login
- Logout
- Registar utilizador
- Recuperar palavra-passe
- Validar a conta (email)