

## Teste NodeJS

Nesse teste você deve desenvolver uma aplicação back-end utilizando NodeJS. Sua aplicação pode ser escrita utilizando Javascript ou TypeScript (preferência por TypeScript).

### O que deve ser desenvolvido?

Você deve criar uma API REST com rotas para criação, atualização, deleção e recuperação de um usuário. Haverá dois tipos de usuário, o usuário administrador e o usuário padrão. Apenas um usuário do tipo administrador pode apagar usuários que não sejam o seu. A API não deve permitir a criação de usuários administradores. Um usuário administrador deve ser criado ao subir o sistema, caso esse não exista.

Deve se guardar as seguintes informações de um usuário:

- Nome e sobrenome
- Foto de perfil
- Data de nascimento
- Data de criação do usuário
- Data de atualização
- E-mail
- Senha (devem ser criptografadas)

Data de nascimento e senha não devem ser retornados ao recuperar um usuário pela API REST. No lugar da data de nascimento deve ser retornado a idade do usuário.

Deve haver um end-point para que o usuário administrador liste todos os usuários. Este não pode atualizar outro usuário, mas pode deletá-lo.

Usuário devem poder realizar login e logout. Apenas usuários logados devem poder atualizar, apagar, recuperar e listar usuários (listagem exclusiva para usuários administradores).

### Infraestrutura e testes

Sua aplicação pode utilizar qualquer sistema gerenciador de banco de dados. Você deve testar sua aplicação para garantir o bom funcionamento da mesma. Deve se poder ser capaz de rodar sua aplicação apenas rodando o seguinte comando do docker-compose:

***docker-compose up***

Após a execução do comando acima seus end-points devem estar disponíveis para uso.

### Pontos adicionais

- Realize o cache onde for possível
- Mande um e-mail ao usuário cadastrado apenas para dar boas-vindas ao mesmo
- Se divirta 😊

## Considerações finais

Não se esqueça de adicionar um README ao seu repositório do GitHub com as instruções de como rodar sua aplicação e os end-points disponíveis.