

Entrevista Bluelab - Parte 2

Implementar um serviço REST e chat em Node.js:

Pode utilizar qualquer framework para desenvolvimento. De preferência utilizar Express ou NestJS, que são os utilizados por nós dentro da Bluelab.

- Rota 1: Recebe os parâmetros nome, sobrenome, telefone e cpf. Essa informação deve ser guardada para que posteriormente seja utilizada pela rota 2. Retorna um Boolean "success" e uma possível mensagem de erro, caso success = false.

Devem ser validados se o telefone informado é válido (formato) e se o cpf informado é válido (validador de cpf).

Ex.: Entrada -> {

```
    "nome": "João",  
    "sobrenome": "Carlos",  
    "telefone": "(11)968552211",  
    "cpf": "88522633910"
```

```
}
```

Saída -> {

```
    "success": false,  
    "msg": "CPF inválido."
```

```
}
```

- Rota 2: Deve ser possível a busca de um usuário. Como o exemplo, buscar pelo parâmetro cpf com o retorno "success=true" e as informações referentes àquele cpf caso já tenham sido capturadas pela primeira rota, caso contrário deve retornar uma mensagem de erro.

Ex.: Entrada -> {

```
    "cpf": "88522633910"
```

```
}
```

Saída -> {

```
    "success": false,  
    "msg": "Informações de CPF não armazenadas."
```

```
}
```

- Rota 3: Deve realizar a atualização de um usuário em específico. Em caso de erro seguir o padrão das anteriores;
- Rota 4: Deve realizar a remoção de um usuário em específico. Em caso de erro seguir o padrão das anteriores;;
- Rota 5: Listagem de todos os usuários cadastrados. Em caso de erro seguir o padrão das anteriores.

Por fim, um usuário poderá conversar com outro via chat ou com todos os usuários online na aplicação. Então deve desenvolver um serviço de chat utilizando Sockets.

Implementar uma aplicação que consome o serviço REST em ReactJS ou Next.js.

A aplicação deve consumir todas as rotas do serviço junto com o chat e possuir uma interface amigável para o usuário que está utilizando a aplicação.

Outras funcionalidades

Essa funcionalidade será tratada como um diferencial não eliminatório.

- Autenticação.

Desenvolvimento

Para o desenvolvimento é esperado a utilização de boas práticas e, de preferência, utilizar o método TDD (Test Driven Development).

Entregáveis

- Código fonte (repositório do GitHub);
- Testes unitários;
- Documentação.