

A interface foi desenvolvida utilizando ReactJS com NextJS na linguagem TypeScript. O deploy foi realizado através do Amplify e pode ser acessado pelo link: <https://main.d1tq3my9akqawc.amplifyapp.com/>

O backend foi desenvolvido utilizando o Serverless Framework com Flask e a linguagem Python e pode ser acessado através do link: <https://1z6xhcwoca.execute-api.us-east-1.amazonaws.com/>

O backend é composto por duas rotas:

[POST] /pwd - Esta rota recebe os seguintes parâmetros:

```
{
  "use_letters": boolean,
  "use_digits": boolean,
  "use_punctuation": boolean,
  "pass_length": number,
  "pass_view_limit": number,
  "expiration_in_seconds": number,
  "sended_password": "string"
}
```

O parâmetro **sended_password** é opcional. Essa rota é responsável por gerar uma nova senha caso o parâmetro **sended_password** não seja enviado. A senha é salva no DynamoDB e é retornada ao usuário um ID que será utilizado no momento de visualização da senha. Caso o parâmetro **sended_password** seja enviado, a rota realizará a validação dos dados e depois fará o encrypt da senha utilizando de uma criptografia simétrica e salvará a senha enviada no DynamoDB.

a chave de criptografia será salva como variável de ambiente da aplicação e não ficará exposta no código.

Environment variables (2)	
The environment variables below are encrypt	
Key	Value
CRYPTO_KEY	_RYGz_
PWD_TABLE	pwd-ta

O parâmetro ***expiration_in_seconds*** será utilizado como TTL para apagar os dados no DynamoDB. Porém, como não há garantias de que esses dados serão excluídos exatamente no momento em que expirarem, a aplicação também realiza a exclusão no momento de acesso caso seja um momento oportuno.

A outra rota é:

[GET] /pwd/:pwdId onde pwdId é o id gerado pela rota citada anteriormente.

Essa rota irá recuperar os dados da senha salva pela rota anteriormente, contanto que o contador de visualizações seja maior que zero e a senha ainda não tenha expirado, então a aplicação irá fazer o decrypt dos dados e retornar para o frontend.

Exemplos de chamadas no backend:

```
curl --location
'https://1z6xhcwoca.execute-api.us-east-1.amazonaws.com/pwd' \
--header 'Content-Type: application/json' \
--data '{
    "use_letters": true,
    "use_digits": true,
    "use_punctuation": true,
    "pass_length": 20,
    "pass_view_limit": 5,
    "expiration_in_seconds": 100
}
```

```
}'
```

exemplo de resposta:

```
{  
  "pwd_id": "26a2824a-80fc-443f-a278-b72def672b9c"  
}
```

```
curl --location  
'https://1z6xhcwoca.execute-api.us-east-1.amazonaws.com/pwd/26  
a2824a-80fc-443f-a278-b72def672b9c' \  
--data ''
```

exemplo de resposta:

```
{  
  "expiration_date": "1681687635",  
  "pwd": "EEs0$`\"F_~\"~L;wRzq5^",  
  "pwd_id": "26a2824a-80fc-443f-a278-b72def672b9c",  
  "view_count": 4  
}
```

Exemplo de casos que previstos pela aplicação:



