



Análise

Análise Desafio Técnico Builders

Controle de Versões			
Versão	Data	Autor	Notas de Revisão
1.0.0	19/10/2022	Ivam Galvão Filho	
1.0.1	29/10/2022	Ivam Galvão Filho	Correção de diagramas, campos e métodos.

1. Título: Desafio Técnico Builders

1.1 Finalidade

Desenvolvimento de uma aplicação SPA em Angular versão 10+ para o desafio técnico do processo de seleção da Builders.

1.2 Escopo

Desenvolva um SPA que tenha pelo menos 2 telas: uma tela de login, pedindo um usuário e senha (mockar autenticação), e outra que consuma a localização atual do usuário e exiba na interface e no input de pesquisa o endereço atual e também os dados climáticos da região. - Deve ser possível inserir manualmente o endereço a ser buscado na API. - Quem estiver deslogado deve ser redirecionado para o login. - Para fazer essa busca, pode-se usar a [API do Open Weather](#) Map.

2. Requisitos

2.1 Requisitos Funcionais

RF1 - Autenticar usuário;

RF2 - Consultar endereço;

RF3 - Consultar dados climáticos.



2.2 Requisitos Não Funcionais:

RNF1 - A aplicação deve ser um SPA com pelo menos 2 telas;

RNF2 - Mockar autenticação;

RNF3 - Utilizar uma API de dados climáticos (de preferência [API do Open Weather](#) Map);

RNF4 - Usar Angular10+ (É permitido o uso de outras bibliotecas);

RNF5 - Utilizar como repositório de versionamento o GitHub;

RNF6 - Produzir documentação;

RNF7 - Anexar Print Screen no Readme.

3. Histórias de Usuário

Cód.: US-01

Título: Autenticar usuário.

Como um: Usuário do sistema.

Eu quero: Realizar a minha autenticação na aplicação com usuário e senha com Mock.

Para que: Permitir a autenticação na aplicação.

Story Points:

Cód.: US-02

Título: Consultar localização e clima.

Como um: Usuário do sistema.

Eu quero: Consultar a minha localização atual para exibí-la na interface e campo de pesquisa e clima da região.

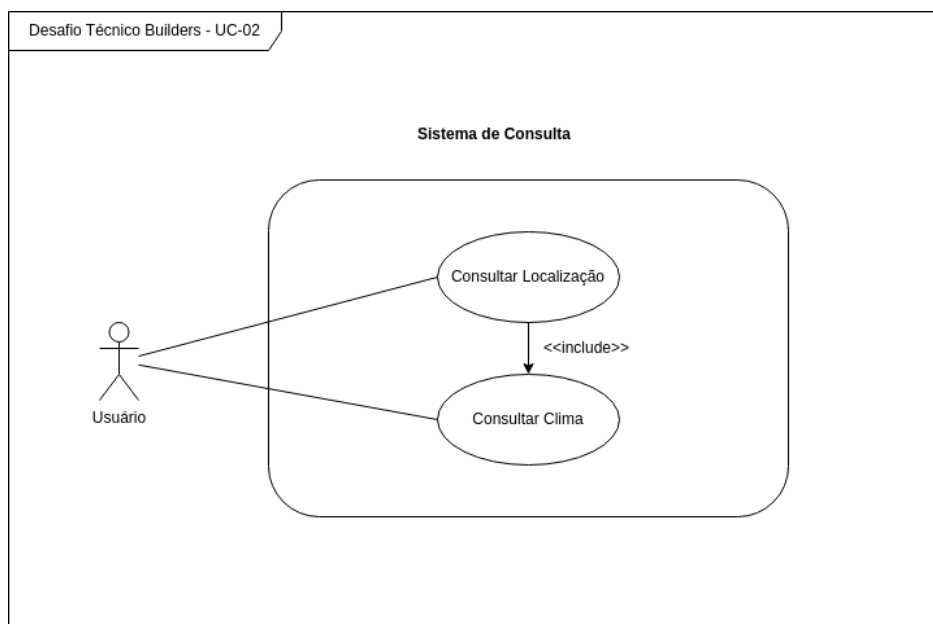
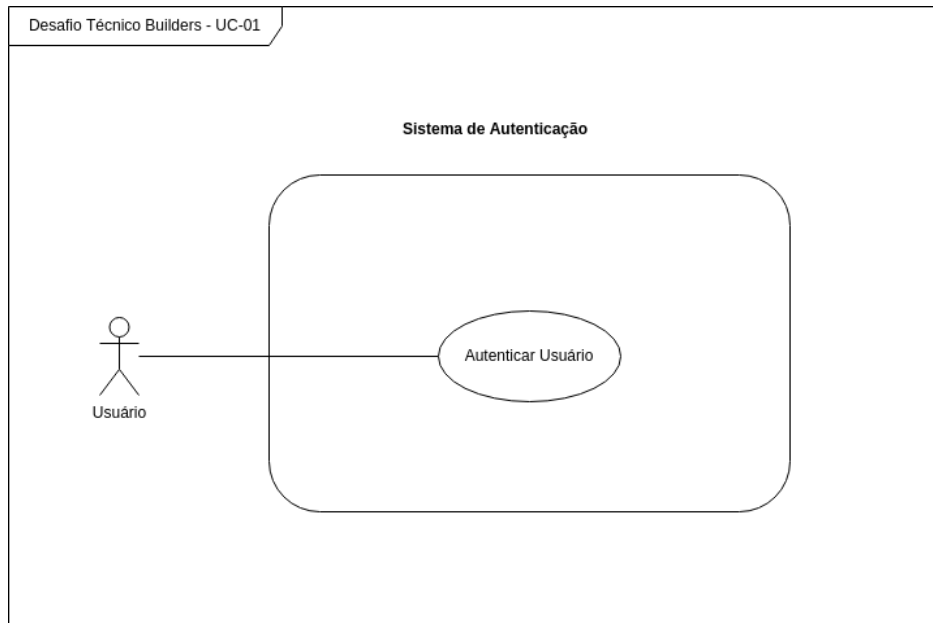
Para que: Possa consultar a localização atual e clima da região.

Story Points:



4. Casos de Uso

4.1 Diagramas de Caso de Uso



4.2 Especificação de Caso de Uso

4.2.1 Dados Gerais

Cód.: UC-01

Ator Principal: Usuário do sistema.

Atores Secundários:



Resumo: Realizar a autenticação na aplicação usando um Mock.

Pré-condições: Usuário e senha já cadastrado em mock.

Pós-condições: Autenticação de usuário me redirecionamento para tela de consulta.

4.2.2 Cenário Principal

Código	Critérios de Aceitação	Cenário	Observações
CNP-1	Deve possuir usuário cadastrado em Api fake.	Dado que: O usuário possui um cadastro de usuário e senha em Api fake. Quando: For preenchido os campos de usuário e senha e clicar no botão Entrar. Então: Será validado os campos de usuário e senha e será realizada a autenticação do usuário e redirecionamento para a tela de consulta.	
CNP-2	Deve possuir localização e dados climáticos atuais do usuário.	Dado que: O usuário foi redirecionado para a tela de consulta. Quando: a tela for renderizada. Então: Deve ser exibida a localização atual do usuário no campo de busca e interface e dados climáticos da região.	
CNP-3	Deve possuir a capacidade de pesquisar a nova localização e dados climáticos na interface e campo de pesquisa.	Dado que: O usuário queira pesquisar nova localização. Quando: O usuário digitar um endereço no campo de busca e clicar no botão pesquisar. Então: Deve ser exibida a localização e dados climáticos pesquisados no campo de busca e interface da aplicação.	



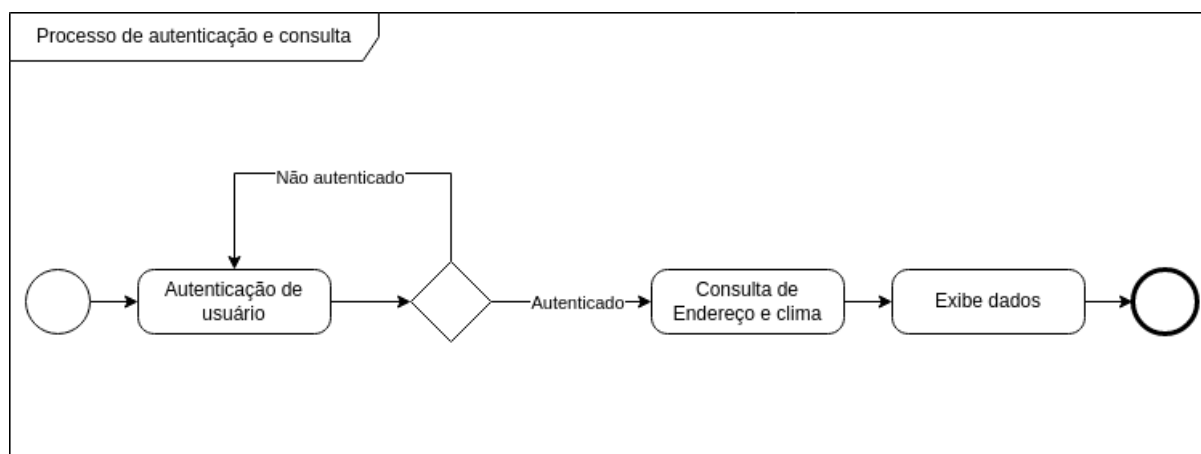
CNP-3	Deve possuir a capacidade de se deslogar da aplicação através de um botão sair com modal de deslogar e redirecionamento para a tela de login.	Dado que: O usuário queira se deslogar da aplicação. Quando: O usuário clicar no botão sair. Então: Deve ser apresentado um modal de confirmação e ao clicar em sim deve ser redirecionado para a tela de login.	
-------	---	---	--

4.2.3 Cenário Alternativo 1

Código	Critérios de Aceitação	Cenário	Observações
CNA-1	Deve possuir redirecionamento para a tela de login e mensagem de erro.	Dado que: O usuário não possui cadastro na Api Fake. Quando: Quando o usuário tentar logar na aplicação. Então: Será redirecionado para a tela de login e apresentado um alerta de erro.	

5. Fluxos

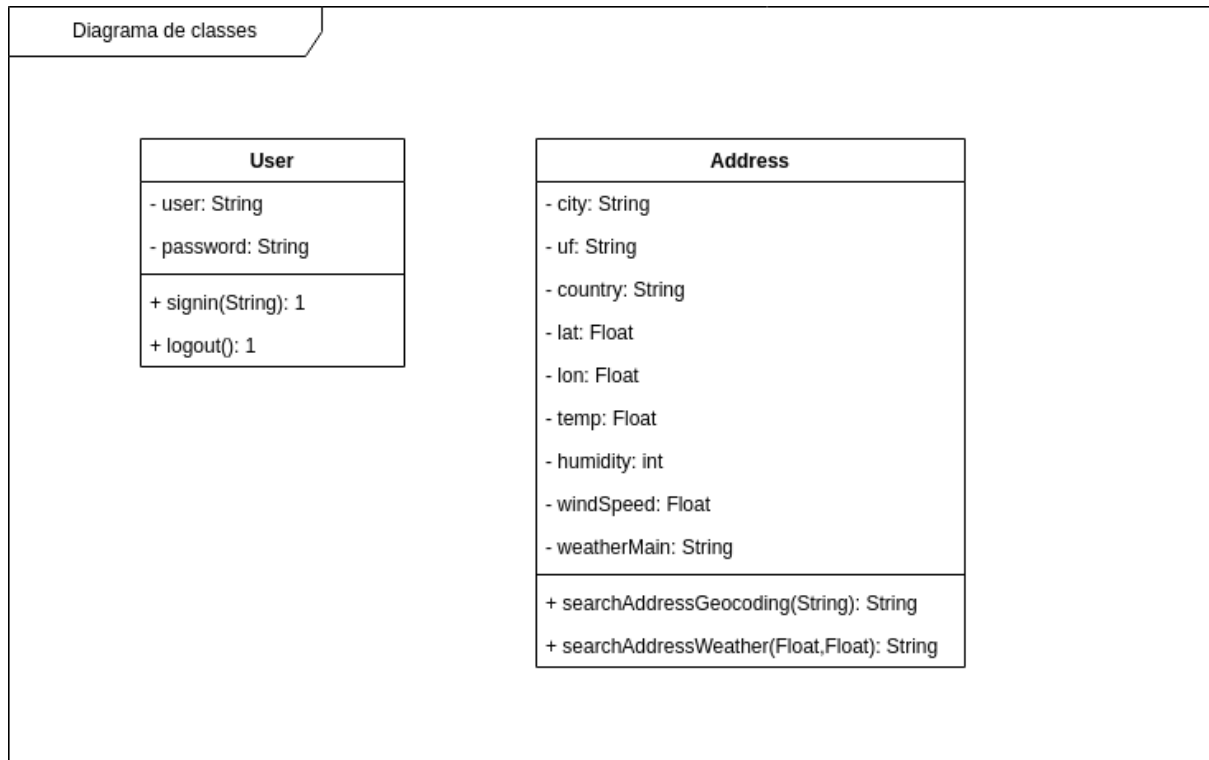
5.1 BPMN



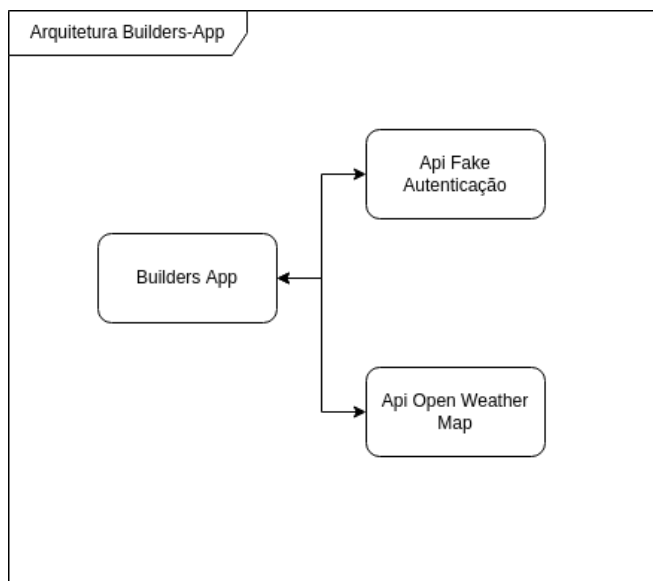


6. Classes

6.1 Diagrama de Classes



6.2 Diagrama de Arquitetura





7. Informações Técnicas

7.1 Descrição dos Campos

Nome	Descrição	Tipo de Dados	Editável	Obrigatório	Regras
user	Campo de usuário	String	Sim	Sim	
password	Campo de senha	String	Sim	Sim	
cidade	Campo de endereço	String	Sim	Sim	Deve vir preenchido com a localização atual e permitir a pesquisa manual.

7.2 Descrição dos Métodos

Método	Ação	Tipo	Restrições
signin(String)	Autenticação do usuário	int	Usar Mock para autenticação.
logout()	Desloga do sistema	int	
searchAddressGeocoding(String)	Consulta de endereço	String	Recebe a cidade como parâmetro e realiza a consulta da localização atual do usuário. Deve permitir a pesquisa manual.
searchAddressWeather(Float,Float)	Consulta de endereço e clima	String	Recebe latitude e longitude como parâmetros e realiza a consulta da localização atual do usuário. Deve permitir a pesquisa manual.



8. Entregáveis:

- Repositório no [GITHUB](<https://www.notion.so/GITHUB-4d23c6fae3f945bc82b7a3832f205d50>);
- Documentação.