

Aplicação: Formulário Alura

Desenvolvedora: Jhoisnáya Vitória Rodrigues de Almeida

Repositório do projeto: [Alura Form](#)

1. Introdução

Este projeto tem como objetivo apresentar uma aplicação mobile desenvolvida com o framework Flutter para o grupo Alura no processo seletivo Scuba Team Mobile. A aplicação deve possuir um formulário para receber as informações de um endereço (nome, telefone, CEP, número, logradouro, bairro, complemento, estado, cidade) e exibir as mesmas em log ou em uma tela.

Algumas validações devem ser aplicadas, como telefone contendo apenas 14 caracteres, logradouro contendo de 10 a 100 caracteres e CEP contendo apenas 8 caracteres. Os campos obrigatórios devem ser decididos pela desenvolvedora.

2. Desenvolvimento

Para começar foi escolhida a ferramenta [Figma](#) para planejamento das telas. Observou-se que a quantidade de dados de entrada tornava a tela muito grande caso todos os campos tivessem o mesmo tamanho, então foram determinadas três opções de interface:

1. Criar campos com tamanhos diferentes de modo a se encaixarem para caber na tela sem usar “scroll”;
2. Utilizar o [widget Stepper](#) para criar dois passos de cadastro numa mesma tela;
3. Criar duas telas para o cadastro.

Analisando todas as opções, viu-se que a melhor e mais simples para o usuário seria a primeira, criar campos com tamanhos diferentes que se encaixam. A escolha de cores foi feita utilizando a ferramenta [Adobe Color](#) e a paleta de cores foi definida.

Assim, a interface da aplicação foi desenhada e foi dado início ao código com o framework [Flutter](#) na linguagem [Dart](#).

O projeto possui os seguintes arquivos: main, tela de formulário e classe Address. O objeto Address foi criado para modelar um endereço.

Na tela do formulário os dados de entrada CEP e número usam o pacote [Extended Masked Text](#) para permitir que o usuário digite apenas a

quantidade correta de números em um dado formato. Também para a entrada da informação do logradouro foi limitada a entrada entre 10 e 100 utilizando o recurso de “validator” e “maxLength”, respectivamente, do [Text Form Field](#). Foi decidido que o único campo não obrigatório é o de complemento.

Por fim, as validações dos campos obrigatórios foram feitas usando também o recurso “validator” de [Text Form Field](#). Ao final do preenchimento do formulário, se o usuário acionar o botão de salvar, é verificado através da chave do [widget Form](#) criado se todos os campos estão válidos, se sim então os dados são mostrados no Log do projeto e exibidos em um modal na aplicação, se não é apresentado ao usuário uma mensagem de erro para o(s) campo(s) obrigatórios não preenchidos. Também foi adicionada a funcionalidade de resetar todos os dados do formulário acionando o botão de reset no canto superior da tela. A figura 1 apresenta o resultado final da aplicação, também há um vídeo que pode ser visto neste [link](#).

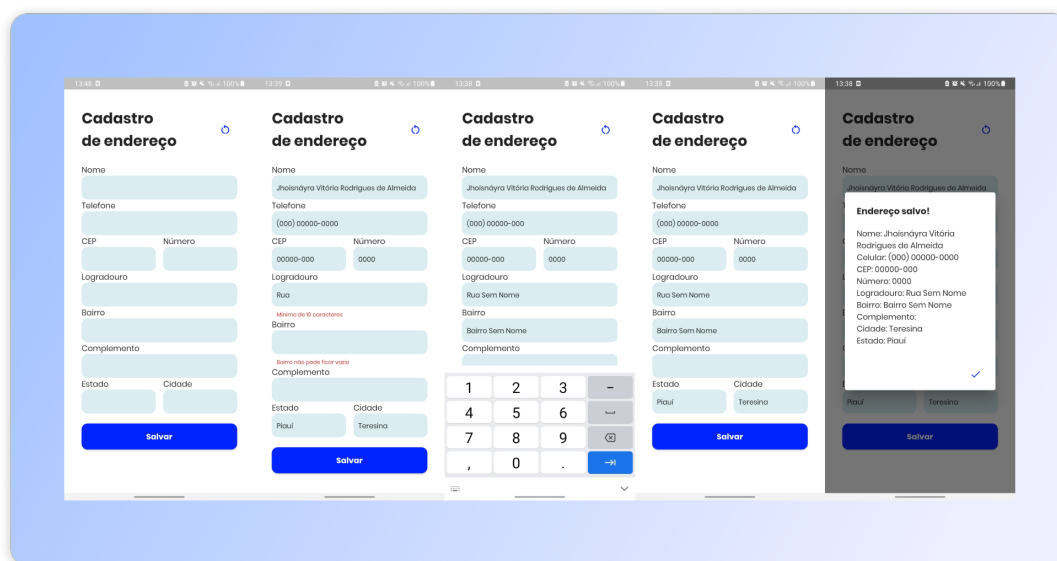


Figura 1: Screenshots do resultado da aplicação. Fonte: Autora.

3. Conclusão

Foi apresentada uma aplicação de formulário para cadastro de um endereço onde são utilizados recursos de validação. Também foram implementados recursos adicionais como modal de exibição dos dados preenchidos.

Como projeto futuro, algumas funcionalidades poderiam ser adicionadas de modo a tornar o projeto mais funcional e ter uma aplicabilidade, como salvar os dados de fato em uma base de dados, por exemplo.