

Interface Pessoa-Máquina

Licenciatura em Engenharia Informática

Ficha Prática #01

Rafael Braga
d13414@di.uminho.pt

José Creissac Campos
jose.campos@di.uminho.pt

Daniel Murta
d6203@di.uminho.pt

(v. 2025)

Conteúdo

1	Objectivos	2
2	Prototipagem de interfaces	2
2.1	Ferramentas	2
2.2	Papel e lápis	2
2.3	Qual ferramenta usar	3
3	Exercícios	3

1 Objectivos

1. Praticar a utilização de ferramentas de prototipagem de interfaces.

2 Prototipagem de interfaces

A prototipagem de interfaces é uma atividade fundamental para o desenvolvimento de produtos digitais. Ela permite testar e validar ideias de forma rápida e eficiente, antes de desenvolver o código final.

2.1 Ferramentas

Existem diferentes ferramentas que podem ser utilizadas para prototipagem, cada uma com suas próprias vantagens e desvantagens.

Algumas ferramentas populares incluem:

- [Figma](#).
- [Sketch](#).
- [Balsamiq Wireframing](#).
- [Axure RP](#).

Algumas destas ferramentas, como Figma, são, por construção ferramentas *online* (executadas na *cloud*). Outras, como Balsamiq, eram originalmente ferramentas *offline*. Nos últimos anos tem-se assistido a uma migração das ferramentas para o modelo *online*, uma vez que possibilitam uma melhor comunicação dentro de equipas.

Uma ferramenta particularmente fácil de usar, e popular na comunidade *open-source*, é o [Evolus Pencil](#), uma ferramenta *offline* e gratuita.

2.2 Papel e lápis

A utilização de papel e lápis é uma opção simples e barata para prototipagem. É ideal para as fases iniciais do projeto, quando é necessário explorar diferentes conceitos e alternativas. Permite liberdade total no desenho de *mockups*, mas fornece menos apoio no que diz respeito à sua animação.

2.3 Qual ferramenta usar

A escolha da ferramenta de prototipagem ideal depende de vários fatores, incluindo:

- O estágio do projeto
- As necessidades do designer
- O orçamento disponível

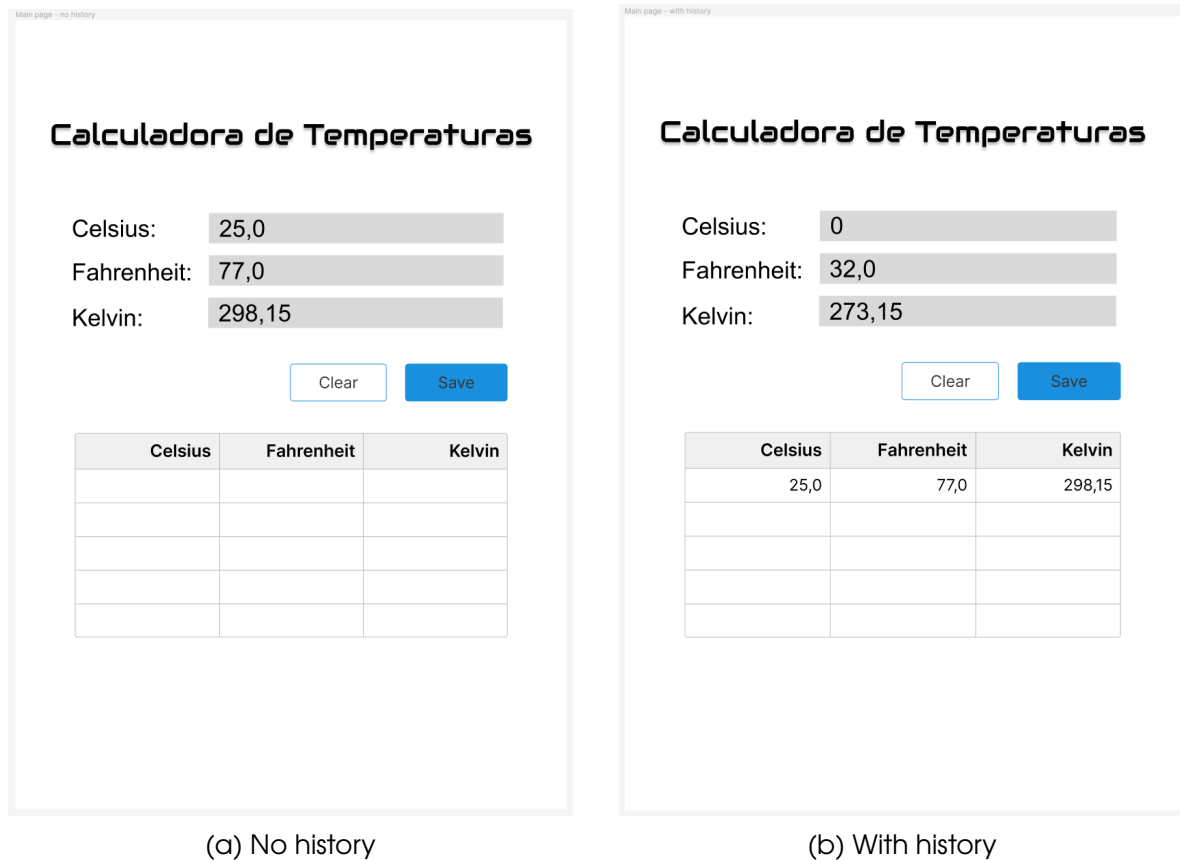
Para projetos iniciais, papel e lápis podem ser uma boa opção. Para projetos mais complexos, uma ferramenta *online* ou *offline* pode ser mais adequada. Nas aulas prática de IPM iremos utilizar Figma.

3 Exercícios

1. Caso ainda não a tenha, crie uma conta no programa [Figma for Education](#) (utilize o seu email institucional).
2. Explore o protótipo disponibilizado [aqui](#), analisando as suas *frames* e as transições entre elas. Para adicionar o erro de user name/password inválidos:
 - (a) Crie uma cópia do projecto na sua conta Figma (utilize a opção “*Duplicate to your drafts*” do menu situado junto à menção *Locked* — canto superior esquerdo).
 - (b) Na *frame* com o user name e password preenchidos, defina áreas sobre o botão para os casos de sucesso e falha do login.
 - (c) Utilize um *overlay* para apresentar a mensagem de erro em caso de falha (terá de criar uma nova *frame* para a mensagem de erro; definir a transição da área definida anteriormente para a nova *frame*; alterar a ação da transição de “*Navigate to*” para “*Open overlay*”).
3. Acrescente ao protótipo os *wireframes* apresentados na Figura 1, sabendo que a transição entre eles é descrita pela máquina de estados da Figura 2. O resultado final está disponível [aqui](#).
 - (a) Crie as *frames* utilizando o modo *Design* do Figma.

Existem *plugins* e coleções de *widgets* que pode utilizar para desenhar os protótipos:

 - Para desenhar os botões, sugere-se o *plugin* [Button Generator](#).



(a) No history

(b) With history

Figura 1: *Frames* do protótipo para o Exercício 2

- Para desenhar tabelas, sugere-se o *plugin Table Creator* (o *plugin Table Generator* é uma alternativa mais simples, mas menos flexível).
- Caso necessite de ícones, existem múltiplos *plugins* que podem ser usados, sugere-se que explore o *plugin Material Design Icons*.

(b) Utilize agora o modo *Prototype* para definir a navegação entre as *frames*.

(c) Teste o protótipo e confirme que está a funcionar corretamente.

(d) Explore agora a implementação de *scrolling* no protótipo. Vai necessitar de alterar a altura da tabela para um valor menor que a altura do seu conteúdo e ativar a opção *Clip content* em *Design* e a opção *Overflow* em *Prototype*.

4. Neste momento, o comportamento da janela de login do protótipo é ainda bastante rudimentar. Clicar no botão de login quando o user name e password não estão preenchidos (primeiro *frame*), não gera nenhuma mensagem de erro. Além disso, o preenchimento do user name e da password são feitos clicando no campo do user name, o campo da password não reage a cliques.

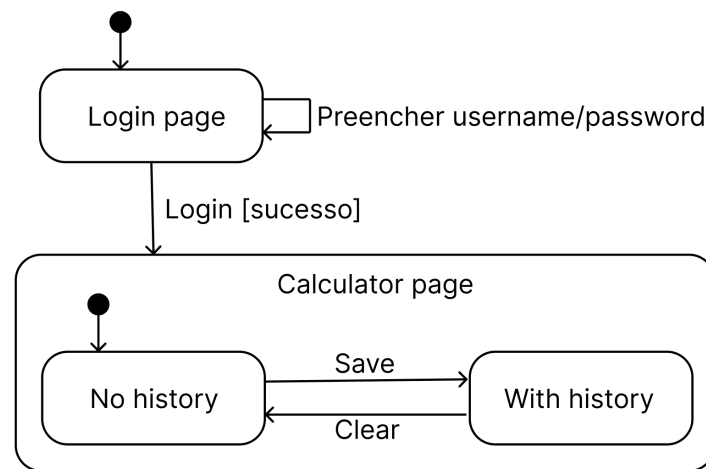


Figura 2: Mapa de navegação para o Exercício 2

Para corrigir estes problemas, é necessário distinguir entre preencher o user name e preencher a password, bem como adicionar mensagens de erro que alertem quando o user name ou a password não foram preenchidos. Uma possibilidade seria criar mais versões da *frame* de login (uma para cada combinação de preenchimento/não preenchimento do user name e da password). No entanto, para formulários com mais campos de *input*, esta abordagem tornar-se-ia rapidamente insustentável.

A solução passa pela **utilização de variáveis**. A definição e utilização de variáveis permite controlar dinamicamente o estado do formulário. É possível utilizá-las para controlar a visibilidade e o conteúdo dos elementos do protótipo, bem como para definir condições nas transições entre *frames*.

Partindo da *frame* de login que tem o user name e password preenchidos (no final poderá apagar a outra *frame* de login), realize os seguintes passos:

- defina duas variáveis booleanas para indicar que cada um dos campos user name e password está preenchido¹;
- defina a visibilidade dos elementos de texto com o user name e a password em função do valor das variáveis que adicionou;
- acrescente interações nos retângulos que representam os campos de texto do user name e da password para fazer o *toggle* do valor da variável correspondente (após este passo, clicar no campo *Username* deverá fazer aparecer/desaparecer o texto do user name e o mesmo para a password);

¹ Se estiver a utilizar a versão Web do Figma, encontra o painel de variáveis do lado direito, se estiver a utilizar a aplicação nativa, vá a View>Panels>Toggle Variables.

- (d) defina um comportamento condicional no botão de login, com as seguintes características:
- i. caso (de acordo com as variáveis) os dois campos não estejam preenchidos, devem ser apresentadas mensagens de erro (ver abaixo);
 - ii. caso os dois campos estejam preenchidos, deve transitar-se para a página principal da aplicação (não se esqueça de fazer o *reset* às variáveis).

Repita o processo, com as adaptações necessárias, para a zona de erro de login.

Quanto às mensagens de erro a mostrar, caso algum campo esteja vazio (ou os dois!), a solução mais simples será utilizar overlays de erro para cada caso (falta o user name, falta a password, faltam os dois). Uma solução mais interessante será fazer o highlight dos campos vazios: por exemplo, apresentando uma borda vermelha e uma mensagem de erro por baixo desse(s) campo(s).