

# Interface Pessoa-Máquina

Licenciatura em Engenharia Informática

---

## Ficha Prática #02

---

Rafael Braga  
d13414@di.uminho.pt

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

Daniel Murta  
d6203@di.uminho.pt

(v. 2025)

## Conteúdo

<b>1</b>	<b>Objectivos</b>	<b>2</b>
<b>2</b>	<b>Protótipos de baixa fidelidade</b>	<b>2</b>
2.1	Vantagens dos protótipos de baixa fidelidade . . . . .	2
2.2	Desvantagens dos protótipos de baixa fidelidade . . . . .	2
<b>3</b>	<b>Exercícios</b>	<b>3</b>

## 1 Objectivos

1. Conceber interfaces utilizando protótipos de baixa fidelidade.

## 2 Protótipos de baixa fidelidade

Nas fases iniciais de concepção, utilizam-se protótipos de baixa fidelidade para esboçar o conteúdo e comportamento da interface, tendo em vista explorar alternativas de *design* e comunicar com os *stakeholders* (potenciais utilizadores, clientes) ou a equipa de desenvolvimento. Apesar de não servirem de base à codificação da interface (*throw away prototypes*), são de desenvolvimento rápido e simples, consistindo em desenhos (*mockups* – para esboçar o conteúdo) e mapas de navegação (para modelar o comportamento).

Os *mockups* capturam o conteúdo e estrutura das diferentes interfaces da aplicação. Os mapas de navegação são, essencialmente, máquinas de estado. Complementam os *mockups*, fornecendo uma visão global sobre comportamento da interface. Cada janela da aplicação é associada a um estado. Cada estado pode ter uma sub-máquina que representa o comportamento da janela.

Os protótipos podem ser criados diretamente em papel ou com o apoio de ferramentas como o [Figma](#).

### 2.1 Vantagens dos protótipos de baixa fidelidade

Recomenda-se começar com uma abordagem baseada em papel, uma vez que apresenta um conjunto de vantagens:

- baixo custo — não só em termos económicos, mas principalmente de tempo;
- facilidade de construção — curva de aprendizagem baixa (este aspeto é relevante pois permite a integração de utilizadores finais);
- maior liberdade de expressão — ferramentas de prototipagem inevitavelmente limitam o que pode ser feito;
- aspeto “imperfeito” encoraja contribuições e alterações;
- maior controlo sobre o nível de abstração.

### 2.2 Desvantagens dos protótipos de baixa fidelidade

Os protótipos de baixa fidelidade também têm algumas desvantagens, incluindo:

- distância para o produto final — os protótipos de baixa fidelidade não são muito parecidos com o produto final, o que pode dificultar a avaliação da experiência do utilizador.
- limitações de funcionalidade — os protótipos de baixa fidelidade geralmente não incluem funcionalidade completa, o que pode limitar o *feedback* que os *stakeholders* podem fornecer.

Para terminar, realça-se que para conceber a interface deverá já ter estudado o domínio do problema, quem são os utilizadores e quais os seus objetivos e tarefas.

### 3 Exercícios

1. Considere um futuro sistema de gestão de receitas eletrónicas. Após entrevistas com médicos, ficou a saber-se que:

- cada receita médica deve identificar o doente e o médico pelos seus códigos do SNS e pode apenas conter até três medicamentos diferentes;
- para cada medicamento deve ser indicado o princípio ativo, uma quantidade e (eventualmente) uma posologia;
- existe um tipo especial de medicamentos (psicotrópicos) que não podem ser “misturados” com medicamentos de outros tipos numa mesma receita médica;
- só podem ser receitados medicamentos constantes da base de dados de medicamentos [Infomed](#).

Foi desenvolvido o seguinte caso de uso:

---

Use case: passar receita

Pré-condição: Médico fez login e identificou o doente

Pós-condição: É emitida uma receita válida com os  
medicamentos indicados pelo médico

Fluxo normal

1. Sistema apresenta informação do doente (nome, sexo, idade e receitas anteriores)
2. Médico indica medicamentos a receitar
3. Sistema regista receita
3. Sistema emite receita

Fluxo alternativo 1 (passo 2) [médico pretende utilizar uma receita anterior]

2.1. Médico seleciona uma das receitas anteriores

2.2. Médico edita os medicamentos presentes na receita

2.3. Regressa a 3

Fluxo de exceção 2 (passo 3) [médico indicou mais de três medicamentos ou misturou medicamentos psicotrópicos e não-psicotrópicos]

1.1. Sistema avisa que combinação de medicamentos é inválida

---

Foi ainda desenvolvido o seguinte perfil de um utilizador típico da aplicação:

João Silva, 45 anos, é médico clínico geral e trabalha na USF +Braga. Ele trabalha há 20 anos na área e tem uma grande quantidade de pacientes, muitos deles com problemas crónicos ou graves. O João tem que realizar as consultas de forma rápida e eficiente, sem comprometer a qualidade do atendimento. Uma de suas preocupações é garantir que as receitas que passa são válidas e seguras, principalmente porque tem pouco tempo para atender cada doente e já no passado aconteceu misturar medicamentos psicotrópicos com não-psicotrópicos na mesma receita. Para passar as receitas recorre ao [Infomed](#) e ao registo das receitas passadas de cada doente.

Sabendo que, em cada consulta, um médico poderá querer receitar um número indefinido de medicamentos, uns psicotrópicos, outros não:

- (a) Tendo em consideração o caso de uso e o perfil de utilizador identificado acima, indique até quatro objetivos que a interface do sistema de gestão de receitas deverá satisfazer.
- (b) Desenvolva agora o protótipo de uma interface que satisfaça esses objetivos.
- (c) Discuta de que modo o sistema que propôs ajuda a acelerar a criação de receitas válidas, facilitando (ou não?) a vida ao João.

2. Considere que pretende desenvolver uma aplicação de gestão de horários para um centro de estudos à distância, para correr numa *smartTV*.

Os professores irão poder marcar aulas após indicarem o seu código de docente e PIN. Cada aula ocorre num dado dia da semana, tem uma hora de início (considere que todas começam à hora certa) e uma duração, é de uma disciplina e lecionada a uma turma, podendo ser recorrente, ou não.

Os alunos podem consultar as aulas da sua turma, quer da semana atual quer das restantes semanas.

Toda a informação necessária para o funcionamento do sistema é tratada num *backend*: os docentes, os alunos, as durações possíveis para as aulas (tipicamente 50 ou 110 minutos); as disciplinas lecionadas por cada professor (o normal é ser uma, mas podem ser mais), a data de fim do período, etc. Na prática não é aqui necessário considerar a sua edição.

Responda agora às seguintes questões:



- (a) Construa um protótipo da interface (*mockups* + navegação), sabendo que deverá ser utilizada através do comando apresentado na figura e que, por existir alguma rotatividade nos professores do centro de estudo, ela deverá ser fácil de utilizar por utilizadores inexperientes.
- (b) Considere agora que lhe foi pedido que a interface possa ser também usada sem recurso ao comando (porque se perdeu, porque ficou sem bateria). Sabendo que existem atualmente televisões com um único botão físico (capaz de reconhecer *clicks* curtos, *clicks* longos e duplos *clicks*), refaça a navegação da interface proposta para que possa ser utilizada desse modo.