



**Politécnico  
de Viseu**

Escola Superior  
de Tecnologia  
e Gestão de Viseu

## **Website Estático de Adoção Animal**

Hugo Alexandre Pereira Afonso

Mateus Valente Frias Silva

Rodrigo de Almeida Martins

**Relatório do Projeto prático de Aplicações para Internet I**

**Curso Engenharia Informática**

**Janeiro de 2026**





**Politécnico  
de Viseu**

Escola Superior  
de Tecnologia  
e Gestão de Viseu



departamento de  
informática

## **Website Estático de Adoção Animal**

Hugo Alexandre Pereira Afonso (30032)

Mateus Valente Frias Silva (29989)

Rodrigo de Almeida Martins (30773)

### **Relatório do Projeto prático de Aplicações para Internet I**

Curso de Licenciatura em Engenharia Informática

Ano letivo: 2º

Trabalho efetuado sob a orientação de

Steven Abrantes

Nuno Costa

Janeiro de 2026

---

# Agradecimentos

Enquanto grupo, gostaríamos de manifestar o nosso sincero agradecimento aos professores Steven Abrantes e Nuno Costa pelo apoio, orientação e tempo que nos dedicaram ao longo deste trabalho.



# Resumo

Este relatório descreve o progresso do projeto PetBond, uma plataforma web estática voltada para incentivar a adoção responsável e oferecer suporte a animais abandonados. O objetivo principal do trabalho foi desenvolver uma solução digital que simplifique a conexão entre abrigos e possíveis adotantes, incorporando recursos práticos, como a divulgação de animais por categoria, gerenciamento de voluntariado e captação de doações.

Tecnicamente, o website foi construído seguindo os padrões modernos de desenvolvimento, utilizando HTML5 semântico e CSS3 para garantir uma estrutura acessível e um layout flexível baseado em percentagens e wrappers. A componente dinâmica foi assegurada através de JavaScript, permitindo a implementação de um slider de imagens na página principal, um menu de navegação interativo com submenus e um relógio em tempo real presente em todas as páginas. Adicionalmente, utilizou-se a biblioteca React para o desenvolvimento de componentes stateful, nomeadamente em formulários com validação de dados e persistência através do local storage do navegador.

Como resultado, obteve-se uma plataforma funcional e responsiva, que cumpre todos os requisitos pedagógicos da disciplina de Aplicações para a Internet I, oferecendo assim uma experiência de utilizador fluida e uma interface visual focada na empatia e na transparência do processo de adoção.



# Índice

1	Introdução .....	11
2	Fase de Análise .....	13
2.1	Quem é o público-alvo? Que características definem o público-alvo? .....	13
2.2	O que é que o público-alvo procura no website? Que tarefas tem que realizar no website? Quais as suas expectativas? .....	13
2.3	Que objetivos quer o grupo atingir ao implementar o website? São objetivos quantitativos ou qualitativos? .....	14
2.4	Que atitude se pretende conferir ao website? .....	14
2.5	Que sensações se pretende incutir ao utilizador? .....	14
2.6	Quais são as cores dominantes do website? .....	15
2.6.1	Dark Mode .....	15
2.6.2	Light Mode .....	15
2.7	Que funcionalidades e conteúdos gostaria de ver implementadas no site? .....	15
3	Design e Arquitetura .....	17
3.1	Arquitetura da Informação e Navegação .....	17
3.2	Escolhas de Design Visual .....	19
3.3	Adaptabilidade e Layout .....	19
4	Implementação Técnica .....	21
4.1	Componentes e Funcionalidades Críticas .....	21
4.1.1	<i>Slider</i> (Hero Section) .....	21
4.1.2	Sistema de Traduções (i18n) .....	22
4.1.3	Carrossel Aleatório com Navegação .....	22
4.1.4	Lista de Animais com Paginação .....	23
4.1.5	Mapa .....	23



4.1.6	Formulário de Contactos com React .....	23
4.1.7	<i>Dark/Light mode</i> .....	24
5	Resultados Visuais.....	25
5.1	Lista de animais com paginação .....	25
5.2	Carrossel das páginas individuais de cada animal .....	26
5.3	Mapa Interativo.....	27
5.4	Dark/Light Mode .....	28
5.5	Traduções.....	30
6	Conclusão .....	31

# Índice de Figuras

Figura 1 - Menu com paginação .....	25
Figura 2 - Lista de animais na página "Adote" .....	25
Figura 3 - Carrossel página individual de cada animal .....	26
Figura 4 - Mapa na página "Contactos" .....	27
Figura 5 - Parte da página "Adote" no lightmode .....	28
Figura 6 - Parte da página "Adote" no darkmode.....	28
Figura 7-Tradução para português      Figura 8-Demonstração da tradução para português....	30
Figura 9-Tradução para inglês      Figura 10-Demonstração da tradução para inglês .....	30
Figura 11-Tradução para espanhol      Figura 12-Demonstração da tradução para espanhol....	30



# Índice de Excertos de Código

Excerto Código 1 - Função goToSlide .....	21
Excerto Código 2 - Lógica de tradução .....	22
Excerto Código 3 - Lógica de aleatoriedade no carrossel .....	23



# 1 Introdução

Este relatório apresenta o progresso de um site voltado para a adoção de animais, desenvolvido em grupo como parte da disciplina de Aplicações para a Internet I. O principal objetivo deste projeto prático foi utilizar os conhecimentos obtidos durante o semestre, empregando as tecnologias HTML, CSS, JavaScript e React.

O assunto da adoção de animais foi selecionado com o objetivo de conscientizar sobre a relevância do bem-estar animal e de incentivar a adoção responsável, por meio de uma plataforma clara, intuitiva e visualmente atraente. O site oferece informações importantes sobre os animais e apresenta recursos interativos que aprimoram a experiência do utilizador.

Durante o desenvolvimento, levaram-se em conta fatores como o público-alvo, a facilidade de uso e a estruturação dos conteúdos. Este relatório descreve as diferentes etapas do projeto, além das escolhas técnicas e de design feitas ao longo da implementação do site.



## 2 Fase de Análise

Neste capítulo serão abordadas as perguntas propostas, e que foram realizadas antes do início do desenvolvimento, visando ter uma compreensão do tema que estávamos a explorar de forma a permitir uma melhor adaptação dos elementos a utilizar no website.

### 2.1 Quem é o público-alvo? Que características definem o público-alvo?

O público-alvo são pessoas de todas as idades que gostam de animais e que estão interessadas em adotar um cão ou gato, ser voluntário ou apoiar um centro de adoção. Inclui:

- Jovens adultos e famílias à procura de um animal de estimação;
- Pessoas que desejam apoiar causas de proteção animal;
- Voluntários e doadores que pretendem contribuir com tempo, bens ou donativos.

São utilizadores com acesso básico à internet, que valorizam simplicidade, emoção nas imagens e facilidade de contacto.

### 2.2 O que é que o público-alvo procura no website? Que tarefas tem que realizar no website? Quais as suas expectativas?

O utilizador procura:

- Ver animais disponíveis para adoção, com fotos e descrições;
- Obter informações sobre o processo de adoção;
- Entrar em contacto com o centro;
- Saber como contribuir ou ser voluntário;
- As expectativas principais são;
- Um site claro, visual e de navegação fácil;
- Imagens apelativas e informações atualizadas;
- Um formulário funcional e simples de preencher;



- 
- Um design moderno e emocionalmente envolvente, que reflita empatia e confiança.

## **2.3 Que objetivos quer o grupo atingir ao implementar o website? São objetivos quantitativos ou qualitativos?**

Objetivos qualitativos:

- Criar uma experiência positiva e emocionalmente envolvente para o utilizador;
- Sensibilizar o público para a importância da adoção responsável;
- Demonstrar domínio das tecnologias lecionadas na unidade curricular (HTML, CSS, JavaScript, React).

Objetivos quantitativos:

- Aumentar o número de formulários submetidos (contactos, pedidos de adoção, voluntariado).
- Promover maior tempo de navegação no website.

## **2.4 Que atitude se pretende conferir ao website?**

O website deve transmitir uma atitude amigável, responsável e solidária, mostrando que a adoção é um gesto de amor e compromisso. A comunicação deve ser positiva e acolhedora, evitando dramatismos, e incentivando a interação.

## **2.5 Que sensações se pretende incutir ao utilizador?**

Deseja-se que o utilizador sinta:

- Empatia e ternura pelos animais apresentados;
- Confiança no centro de adoção;
- Motivação para agir – seja adotando, doando ou partilhando o projeto;
- A experiência seja leve, emocional e esperançosa.

---

## 2.6 Quais são as cores dominantes do website?

Para garantir uma melhor experiência de navegação, o website dispõe de dois modos de visualização. Abaixo, detalham-se as cores dominantes em cada uma das versões.

### 2.6.1 Dark Mode

- Azul-Escuro: A cor de fundo, cria uma base sofisticada e que destaca as fotografias dos animais.
- Azul-Claro: Funciona como "sinalética" luminosa para guiar as ações principais do utilizador.
- Branco: Cor do texto de modo a garantir o máximo de contraste para uma leitura confortável em ambiente escuro.
- Cor-de-Tijolo: Destaca visualmente a ação de "Doar", do restante do site.

### 2.6.2 Light Mode

- Branco: A cor de fundo, maioritariamente genérica em grande parte dos websites;
- Azul-Escuro: Substitui o branco nos textos, mantendo a lógica do “*dark mode*”;
- Turquesa: Mantém a consistência da identidade visual nos ícones e botões sobre o fundo claro;
- Cor-de-tijolo: Destaca visualmente a ação de "Doar", do restante do site.

## 2.7 Que funcionalidades e conteúdos gostaria de ver implementadas no site?

Página inicial: *slider* com imagens de animais e mensagens de incentivo à adoção. Opiniões de pessoas que já adotaram animais (não foi implementado).

Página “Adote”: listagem de animais disponíveis (com nome, idade e breve descrição).

Página “Voluntarie-se”: explicação e formulário para inscrição (acabou por ficar em contactos).

Página “Contactos”: formulário de contacto com validação e armazenamento local. Horário de atendimento.

---

Menu com submenus: “Adote -> Cães/Gatos/Outros”.

Data e hora atual: visíveis no rodapé.

Design responsivo: adaptável a dispositivos móveis.

Como podemos medir o sucesso do website? Como sabemos se os objetivos foram atingidos?  
É possível medir objetivamente (quantitativamente) o sucesso do website? Como?

O sucesso do website pode ser avaliado através de métricas quantitativas e/ou qualitativas.

Tanto o número de formulários submetidos, ou seja, o número de pedidos de adoção, voluntariado e contactos, como a quantidade de visitas, revisitas e o tempo médio de navegação aquando do site é visitado, são exemplos de diferentes métricas quantitativas que nos permitirão avaliar o sucesso do nosso website.

Qualitativamente, o sucesso será avaliado com base no feedback dos utilizadores e do docente, a facilidade de navegação e o impacto visual do site, bem como no grau de envolvimento gerado pelo design e pelas imagens no site, e no cumprimento dos objetivos principais de promover a adoção responsável, sensibilizar o público e demonstrar domínio das tecnologias utilizadas.

A combinação destes indicadores permitirá verificar se o website é eficaz, apelativo e funcional, cumprindo tanto os objetivos técnicos como os comunicativos do projeto.

## 3 Design e Arquitetura

Neste capítulo serão abordadas as decisões estruturais tomadas antes de iniciar o projeto e as que chegaram à fase final de desenvolvimento, bem como algumas escolhas de design e adaptabilidade.

### 3.1 Arquitetura da Informação e Navegação

A arquitetura de informação e a navegação do projeto PetBond foram concebidas de forma integrada, assentes num modelo hierárquico simples e pouco profundo, que privilegia o acesso rápido aos conteúdos principais e respeita a regra dos “3 cliques”. A estrutura do site é clara desde a página inicial, que funciona como ponto central de entrada e apresenta a missão do projeto e os principais destaques, encaminhando o utilizador para a funcionalidade nuclear de adoção. Esta funcionalidade está organizada numa área própria dedicada aos animais, onde a navegação se ramifica de forma lógica por categorias como cães, gatos e outros animais, cada uma com listagens filtradas que conduzem a páginas de detalhe individuais, permitindo uma progressão natural entre níveis de informação sem sobrecarregar o utilizador.

Paralelamente, existem páginas institucionais como “Sobre Nós”, “Contactos” e “Voluntariado”, bem como páginas de suporte e utilitárias, como doações, perguntas frequentes e política de privacidade, acessíveis de forma transversal. Esta organização é coerente tanto na navegação visível ao utilizador como na própria estrutura de ficheiros HTML e na definição das chaves de tradução, reforçando a consistência entre o desenho conceptual e a implementação técnica.

A navegação global materializa esta arquitetura através de uma barra de navegação presente em todas as páginas, garantindo orientação constante e acesso imediato às principais secções. O menu principal utiliza uma estrutura semântica simples e integra menus *dropdown* na área de adoção, permitindo o acesso direto às subcategorias e reduzindo o número de cliques necessários. Para além da navegação entre conteúdos, a barra superior inclui controlos globais, como o seletor de idioma e a alternância entre modo claro e escuro, que afetam toda a experiência de utilização e reforçam a noção de continuidade.

Em dispositivos móveis, esta mesma arquitetura adapta-se a um padrão de menu hambúrguer, mantendo a lógica de navegação, mas reorganizando visualmente os elementos

---

para ecrãs de menor dimensão. O menu expande-se verticalmente e os *dropdowns* ajustam-se ao toque, assegurando usabilidade e coerência independentemente do dispositivo utilizado. O rodapé complementa a navegação principal ao funcionar como um mapa do site secundário, agregando links relevantes de forma temática, desde redes sociais e categorias de adoção até áreas de apoio e informação legal, permitindo acessos alternativos sem necessidade de regressar ao topo da página.

Para além da navegação global, o site integra mecanismos contextuais que reforçam a exploração do conteúdo. As listagens de animais recorrem a paginação para facilitar a navegação em grandes conjuntos de dados, enquanto botões de chamada à ação orientam o utilizador para fluxos de conversão claros, como adotar ou fazer uma doação. A navegação lateral é incentivada através de sugestões de outros animais em carrossel, promovendo a descoberta contínua, e um botão de retorno ao topo assegura um acesso rápido à navegação principal após períodos prolongados de *scroll*. Em conjunto, estes elementos demonstram uma integração consistente entre arquitetura de informação e navegação, focada na clareza, eficiência e orientação permanente do utilizador.

---

## 3.2 Escolhas de Design Visual

O design visual do PetBond privilegia uma estética moderna e minimalista, focada na usabilidade e na resposta emocional gerada pelas fotografias de alta qualidade dos animais, que assumem o protagonismo da interface. A paleta de cores foi definida para garantir contraste e acessibilidade, integrando um sistema de temas claro e escuro que se adapta dinamicamente às preferências do utilizador, enquanto a tipografia sans-serif assegura uma leitura fluida em todos os dispositivos. A estrutura visual assenta num layout responsivo organizado por cartões e grelhas, complementado por ícones intuitivos da biblioteca Bootstrap, resultando numa navegação consistente e visualmente harmoniosa que se ajusta eficazmente desde ecrãs móveis a computadores de secretária.

## 3.3 Adaptabilidade e Layout

A estratégia de adaptabilidade e layout foi desenvolvida com base nos princípios do Design Responsivo, garantindo que a interface se ajusta fluidamente a qualquer resolução, desde desktops a dispositivos móveis. A estrutura visual privilegia o uso de Flexbox e CSS Grid para o alinhamento de componentes, combinados com unidades relativas (como percentagens e rem) em detrimento de dimensões fixas, assegurando a integridade do layout.

Para otimizar a experiência em ecrãs de menores dimensões, foi implementado um Menu Hambúrguer (controlado via JavaScript) que compacta a navegação principal. Esta solução preserva o espaço útil do ecrã e mantém a interface limpa, garantindo uma usabilidade consistente independentemente do dispositivo utilizado.



## 4 Implementação Técnica

A implementação do projeto seguiu a arquitetura estática anteriormente descrita de perto, usando apenas HTML5, CSS3 e JavaScript para a interatividade no lado do cliente.

A estrutura do código foi pensada para promover a reutilização e a manutenção. Elementos comuns a todas as páginas, como a *navbar* (<header>) e o *footer* (<footer>), mantêm uma estrutura HTML consistente em todos os ficheiros. A nível de estilos, utilizou-se um ficheiro global *styles.css* para definições base, como variáveis de cores ou de tipografia, enquanto ficheiros específicos gerem o layout de páginas individuais. O JavaScript segue a mesma lógica, com um ficheiro *scripts.js* para funcionalidades globais (tema, relógio, menu mobile) e scripts dedicados para funcionalidades específicas

### 4.1 Componentes e Funcionalidades Críticas

Abaixo detalha-se a lógica de implementação dos componentes mais complexos do sistema.

#### 4.1.1 *Slider* (Hero Section)

O *slider* presente na página inicial (*index.html*), cujo código JS está em *home.js* permite a navegação automática e manual entre imagens de destaque. A lógica baseia-se na manipulação de classes CSS (*active*) num *array* de elementos.

```
// Function to change slide
function goToSlide(n) {
    // Remove active class from current
    slides[currentSlide].classList.remove('active');
    dots[currentSlide].classList.remove('active');

    // Update Index
    currentSlide = (n + slides.length) % slides.length;

    // Add active class to new
    slides[currentSlide].classList.add('active');
    dots[currentSlide].classList.add('active');
}
```

*Excerto Código 1 - Função goToSlide*



---

A parte mais crítica ou complexa desta funcionalidade está presente no Excerto Código 1, mais especificamente na parte onde se atualiza o índice, isto porque temos de garantir que não avançamos para uma posição inexistente do *slider*. Para a reprodução automática, utilizou-se o *setInterval* que invoca a função de avançar a cada 6 segundos, sendo este temporizador reiniciado (*resetTimer*) sempre que o utilizador interage manualmente com os botões de navegação.

#### 4.1.2 Sistema de Traduções (i18n)

O mecanismo de traduções (implementado em *translations.js*) foi feito sem recarregar a página, utilizando um objeto JavaScript como dicionário. A lógica passa pelo dicionário conter chaves aninhadas por idioma (*pt*, *uk* e *es*). A função *updatePageLanguage* percorre o DOM à procura de elementos com o atributo personalizado *data-i18n*. O valor deste atributo corresponde à chave no dicionário. O sistema também traduz atributos como *alt* ou *aria-label*, para garantir uma boa acessibilidade. A lógica de tradução pode ser encontrada no Excerto Código 2 abaixo.

```
elements.forEach(element => {  
  const key = element.getAttribute('data-i18n');  
  
  if (translations[lang] && translations[lang][key]) {  
    element.textContent = translations[lang][key];  
  } else {  
    console.warn(`Missing translation for: ${key} in language: ${lang}`);  
  }  
});
```

Excerto Código 2 - Lógica de tradução

A preferência do utilizador é persistida através do *localStorage*, garantindo que o idioma se mantém entre sessões ou *refreshes*.

#### 4.1.3 Carrossel Aleatório com Navegação

Na secção de animais relacionados, implementou-se um carrossel que apresenta sugestões aleatórias a cada carregamento da página (*carousel.js*). A lógica de *shuffle* passa por, antes de carregar o carrossel, o *script* recolher todos os cartões e baralhá-los utilizando o algoritmo de Fisher-Yates. Os elementos são depois reintroduzidos no DOM na nova ordem aleatória. No que toca à lógica de navegação horizontal (com o rato), esta não utiliza bibliotecas externas, mas sim a API nativa *scrollBy* com comportamento *smooth*.

---

```
for (let i = cards.length - 1; i > 0; i--) {  
  const j = Math.floor(Math.random() * (i + 1));  
  // Swap element positions  
  [cards[i], cards[j]] = [cards[j], cards[i]];  
}
```

*Excerto Código 3 - Lógica de aleatoriedade no carrossel*

No Excerto Código 3 podemos ver a parte mais complexa desta funcionalidade, sendo esta a troca dos cartões.

#### 4.1.4 Lista de Animais com Paginação

Para gerir a visualização de múltiplos animais sem sobrecarregar a página, criou-se um sistema de paginação no lado do cliente (*navigation.js*). Com base num limite de itens por página (“itemsPerPage = 6”), o script calcula o número total de páginas (*Math.ceil*). A função *showPage* determina quais os índices de animais que devem estar visíveis (“display: flex”) e quais devem ser ocultados (“display: none”), baseando-se na fórmula: “index >= start && index < end”. Sempre que a página muda, a função *scrollToTop* é invocada para melhorar a experiência do utilizador, levando o scroll para o topo da lista.

#### 4.1.5 Mapa

A implementação do mapa em *contacts.html* foi realizada através da integração de um `<iframe>`. Esta abordagem permite incorporar o Google Maps diretamente no HTML estático sem necessidade de chaves de API complexas para visualização simples, garantindo que o utilizador tem acesso à localização interativa e responsiva definida no CSS (“width: 100%”).

#### 4.1.6 Formulário de Contactos com React

Apesar de ser um site estático, a página de contactos integra a biblioteca React (via CDN) para gerir o formulário de forma mais robusta (*contacts.js*). A lógica desta parte passa pelo componente *ContactForm* utilizar *Hooks* para gerir o estado:

- *useState*: Gere os dados do formulário (*formData*), erros de validação e o estado de submissão.
- *useEffect*: Implementa uma funcionalidade de “rascunho automático”. Ao montar o componente, verifica se existe um rascunho guardado no *localStorage* (*contact\_draft*) e preenche os campos automaticamente.

---

A validação é feita em tempo real ou no envio, impedindo a submissão caso os campos obrigatórios ou o formato de email estejam incorretas.

#### **4.1.7 Dark/Light mode**

O modo escuro (*scripts.js*) é transversal a todo o site e funciona através da manipulação de classes na *tag* raiz `<html>`. A lógica de implementação do modo escuro inicia-se com a detecção da preferência do utilizador através da função *initTheme* que verifica primeiramente se existe uma configuração guardada no *localStorage* sob a chave *petbond\_theme*; na ausência desta, o sistema recorre à preferência definida no SO utilizando a query “`window.matchMedia('(prefers-color-scheme: dark)')`”. Para a alternância entre temas, a função *themeSwitch* monitoriza as alterações na caixa de seleção do menu e, ao detetar uma mudança, troca dinamicamente as classes CSS entre *dark-mode* e *light-mode*, atualizando simultaneamente o *localStorage* para garantir a persistência da escolha

# 5 Resultados Visuais

Neste capítulo serão mostradas algumas fotos do resultado do website desenvolvido, com os devidos comentários.

## 5.1 Lista de animais com paginação

O menu de navegação (Figura 1) distingue-se pelo seu design moderno, utilizando botões de cantos arredondados que se harmonizam com o tema escuro da interface. A cor ciano realça a página ativa, criando um ponto de referência visual imediato, enquanto as setas laterais tornam a interação intuitiva.

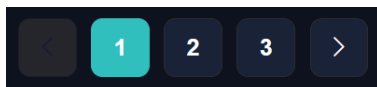


Figura 1 - Menu com paginação

A estrutura da galeria (Figura 2) foi pensada para evitar a sobrecarga visual, limitando a exibição a um máximo de seis animais por página. Esta organização fragmentada mantém o ecrã limpo e equilibrado, permitindo que o utilizador analise cada cartão com maior atenção e clareza.

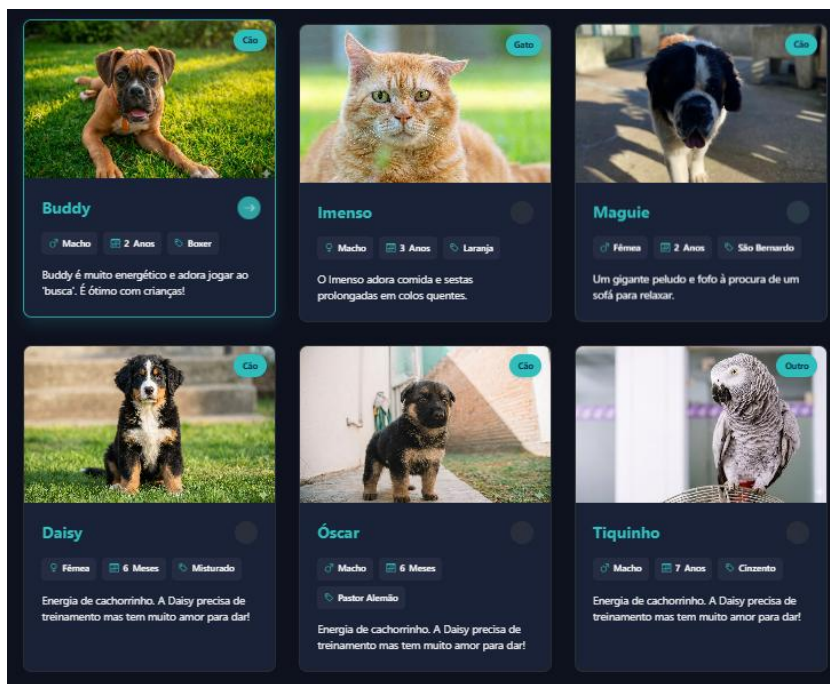


Figura 2 - Lista de animais na página "Adote"

---

Para completar a experiência, a mudança de página aciona um deslizamento suave (*scroll*) automático de volta ao topo da lista. Este detalhe funcional assegura que, ao avançar, o foco do utilizador é imediatamente direcionado para os novos animais, tornando a navegação contínua e sem esforço.

## 5.2 Carrossel das páginas individuais de cada animal

O menu de carrossel (Figura 3) apresenta um *design* moderno e coeso, adotando cartões semelhantes aos da galeria de animais, apenas mais minimalista. Existe uma hierarquia visual clara que distingue o cartão ativo ("Cuca") dos restantes: este recebe um destaque especial através de uma borda iluminada e um botão de ação preenchido com uma seta, enquanto os itens inativos possuem botões apenas contornados e menor ênfase visual, orientando o foco do utilizador.

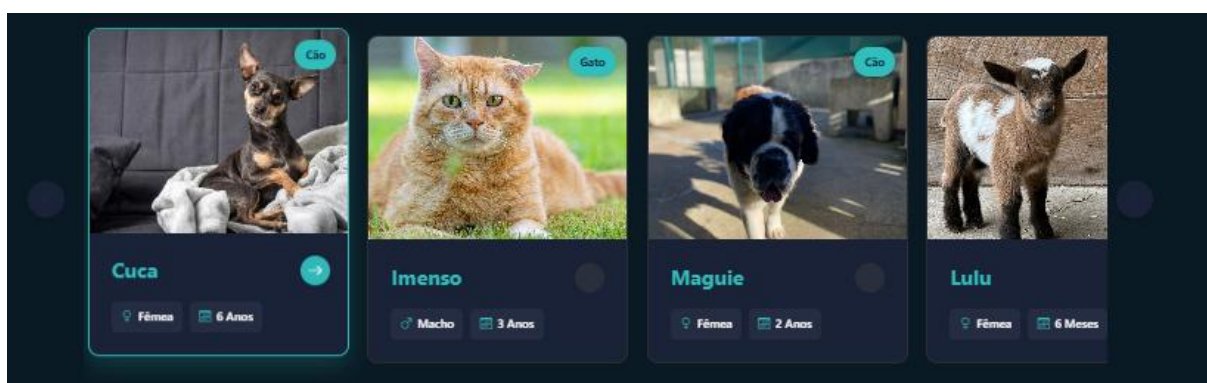


Figura 3 - Carrossel página individual de cada animal

A funcionalidade do carrossel foca-se na descoberta e na facilidade de navegação. O sistema implementa uma lógica de "sugestões dinâmicas" que, através da aleatoriedade, apresenta os animais numa ordem diferente a cada visita, mantendo o conteúdo fresco e interessante. A navegação é controlada por botões laterais (".btn-prev" e ".btn-next") que permitem deslizar os cartões horizontalmente de forma intuitiva, garantindo que o utilizador consegue explorar todas as opções disponíveis de forma fluida.

A base técnica sustenta-se numa hierarquia de três elementos CSS essenciais: um contentor relativo para posicionar os botões de navegação sobre o conteúdo, um *wrapper* ("overflow-x: auto") que funciona como janela de visualização, e um *track* com "display: flex" que alinha os cartões lado a lado. A lógica utiliza o algoritmo *Fisher-Yates shuffle* para garantir uma reordenação verdadeiramente aleatória dos elementos no DOM, enquanto o movimento de

---

*scroll* é calculado dinamicamente pelo JavaScript (“scrollBy”) com base na largura real de cada cartão (“offsetWidth”), assegurando responsividade em diferentes ecrãs.

## 5.3 Mapa Interativo

Para facilitar o contacto direto, decidimos integrar um mapa dinâmico (Figura 4) na página de contactos. Esta escolha de design visa reduzir o esforço do utilizador na procura da nossa localização, o que oferece uma ferramenta interativa diretamente no browser. A estética desta secção foi pensada para manter a sobriedade do projeto, com o uso de sombras e contornos que conferem profundidade ao layout.

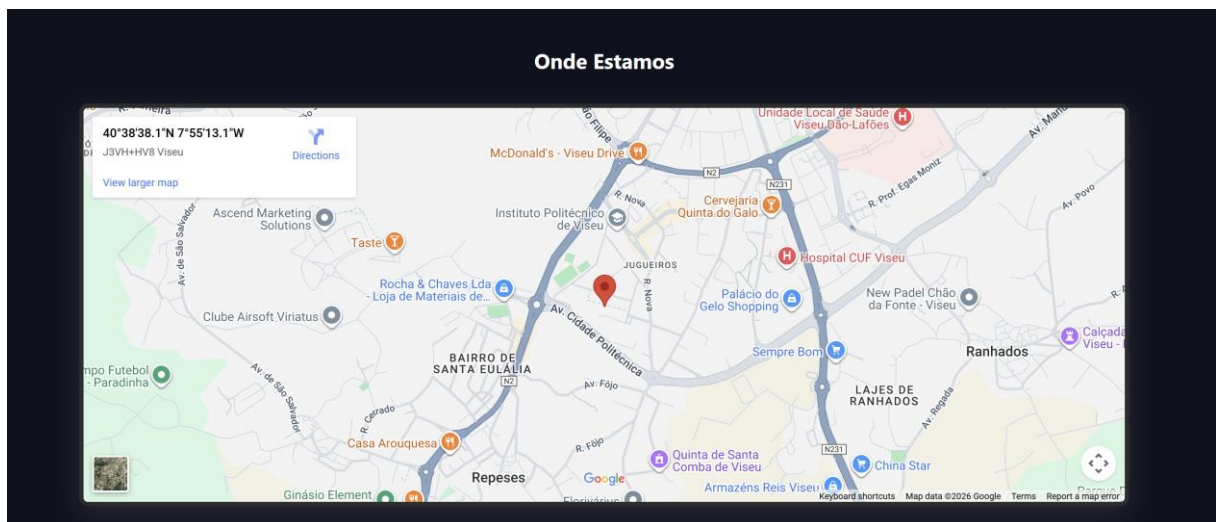


Figura 4 - Mapa na página "Contactos"

Do ponto de vista técnico, a estrutura foi montada dentro de uma *section* semântica, com o uso de um contentor flexível que garante que o mapa ocupa 100% da largura disponível, adaptando-se assim automaticamente ao tamanho do ecrã. No CSS, aplicámos uma moldura personalizada com *border-radius* de 10px e uma sombra suave (*box-shadow*), dando ao mapa um aspeto moderno que se destaca do fundo da página. Além disso, utilizámos o atributo *loading="lazy"* no código HTML para garantir que o mapa só é carregado quando o utilizador faz scroll até ele, otimizando assim a velocidade de carregamento inicial da página.



## 5.4 Dark/Light Mode

Uma das funcionalidades implementadas no PetBond é a possibilidade de alternar entre o modo claro e o modo escuro. Esta escolha de design não é apenas estética, mas também responde a uma necessidade de acessibilidade, o que permite ao utilizador adaptar a luminosidade do website ao ambiente em que se encontra, reduzindo a fadiga ocular em condições de pouca luz. O seletor de tema foi desenhado para ser discreto, mas de fácil acesso, encontrando-se integrado na barra de navegação para uma utilização intuitiva.

### Adote



Figura 5 - Parte da página "Adote" no lightmode

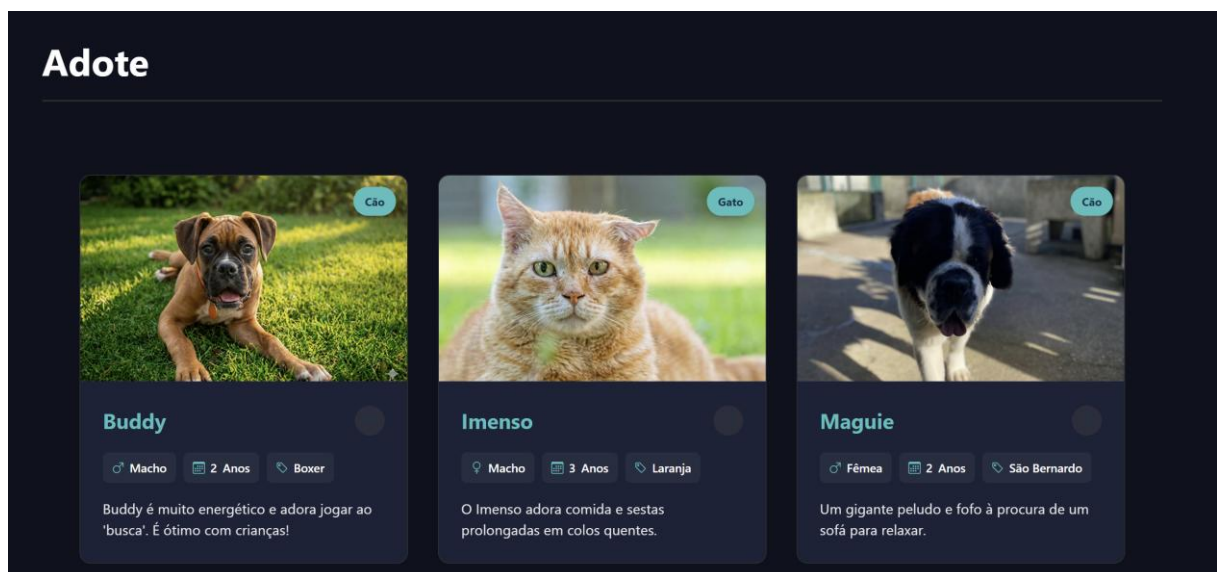


Figura 6 - Parte da página "Adote" no darkmode

---

Como é possível observar tanto na interface como no código apresentado, o sistema de troca de tema foi construído com o uso de um componente de interruptor personalizado. Utilizamos um *checkbox* escondido por trás de um *label* estilizado com CSS, o que permite manter o estado do interruptor de forma nativa e acessível através do atributo *aria-label*. No CSS, tirámos partido de variáveis personalizadas que mudam os seus valores em cascata por todo o documento, garantindo assim uma transição coerente em todas as páginas.

Para elevar a qualidade, aplicámos a propriedade “*transition: all 0.3s ease*”, que permite que a mudança de cores e o deslizar do botão ocorram de forma fluida, evitando qualquer corte visual agressivo para o utilizador. Além disso, integrámos ícones da biblioteca *Bootstrap Icons* dentro do próprio interruptor, que alteram a sua opacidade conforme o tema selecionado para fornecer um feedback visual imediato. Por fim, tivemos o cuidado de incluir um estado de “*:focus*” no CSS, tendo assim a garantia de que os utilizadores que navegam via teclado consigam identificar claramente a posição do seletor.



---

## 5.5 Traduções

Para que a PetBond pudesse chegar a mais pessoas, decidimos que o site não deveria estar limitado apenas ao português. Criámos um sistema de tradução que permite mudar o idioma de forma instantânea, tornando a plataforma muito mais inclusiva para quem não fala a nossa língua, mas quer ajudar ou adotar um animal.



Figura 7-Tradução para português



Figura 8-Demonstração da tradução para português



Figura 9-Tradução para inglês

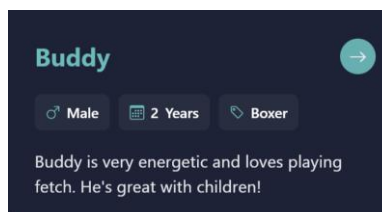


Figura 10-Demonstração da tradução para inglês



Figura 11-Tradução para espanhol



Figura 12-Demonstração da tradução para espanhol

Como podemos ver nas imagens, o utilizador tem a liberdade de escolher o idioma que lhe for mais confortável através de um menu *dropdown* simples. Esta funcionalidade é fundamental nos dias de hoje, pois permite que a nossa mensagem de apoio à causa animal ultrapasse fronteiras. Ao mudar o idioma, todo o conteúdo do site desde os menus até às descrições dos animais é atualizado no momento, sem que a página precise de ser recarregada. Além disso, o site é inteligente o suficiente para se lembrar da língua que o utilizador escolheu, ou seja se ele fechar o browser e voltar mais tarde, o site continuará a aparecer no idioma selecionado, o que torna a navegação muito mais natural e personalizada.

---

## 6 Conclusão

O desenvolvimento do projeto *PetBond* permitiu cumprir com êxito os objetivos propostos na unidade curricular de Aplicações para a Internet I, resultando numa plataforma de adoção animal funcional, responsiva e centrada na experiência do utilizador.

A principal mais-valia técnica deste projeto residiu na integração híbrida de tecnologias web fundamentais (HTML5, CSS3, JavaScript) com a biblioteca React. Esta abordagem permitiu compreender a complexidade da manipulação direta do DOM — evidente na implementação do *slider* e do sistema de internacionalização (i18n) — ao mesmo tempo que se exploraram as vantagens da reatividade na gestão de estados no formulário de contactos. A implementação de funcionalidades como o *Dark Mode* e a tradução dinâmica em tempo real reforçou a acessibilidade e a usabilidade do sistema, respondendo diretamente aos requisitos de inclusão definidos na fase de análise.

Não obstante o sucesso da implementação, o projeto, pela sua natureza estática (*Client-Side*), apresenta limitações inerentes, nomeadamente a ausência de uma base de dados robusta para a persistência definitiva dos registos de animais e utilizadores. Num contexto de evolução futura, seria fundamental a integração de uma *API Backend* para gerir dinamicamente o catálogo de adoções e autenticação de administradores.

Em suma, o *PetBond* não só solidificou as competências técnicas do grupo em desenvolvimento *Front-End*, como também promoveu o desenvolvimento de *soft skills* essenciais, como o planeamento, o trabalho em equipa e a resolução criativa de problemas, culminando num produto que junta a qualidade técnica à responsabilidade social.