

Maria Ferreira

Relatório Final

Microcredencial em Interfaces Web com Javascript

índice

[Objetivos 3](#_Toc151545337)

[Descrição dos elementos dinâmicos implementados no website 4](#_Toc151545338)

[Layout 4](#_Toc151545339)

[Formulário 4](#_Toc151545340)

[código 5](#_Toc151545341)

[declaração e inicialização de variaveis e constantes 5](#_Toc151545342)

[Métodos 6](#_Toc151545343)

[Localstore 6](#_Toc151545344)

[Criar e ler um ficheiro de imagem 7](#_Toc151545345)

[Gerar uma mensagem aleatória 7](#_Toc151545346)

[Criar e guardar um objeto do tipo card 7](#_Toc151545347)

[Estratégia de implementação geral 8](#_Toc151545348)

[Principais desafios técnicos e respetivas soluções 8](#_Toc151545349)

[Obstáculos não ultrapassados e desafios futuros 9](#_Toc151545350)



Objetivos

1. No âmbito da frequência da Microcredencial em interfaces web com javascript foi solicitado que idealizássemos e implementássemos um miniprojecto com os seguintes elementos de programação:
   1. Estruturas de decisão;
   2. Estruturas de repetição (ciclos);
   3. Arrays simples ou Arrays de objetos;
   4. Leitura de dados externos com base numa API que retorne dados em JSON;
2. E o respetivo relatório com a seguinte estrutura:
   1. Descrição dos elementos dinâmicos implementados no website;
   2. Estratégia de implementação geral;
   3. Principais desafios técnicos e respetivas soluções;
   4. Obstáculos não ultrapassados;
   5. Conclusões.

Idealizei um serviço que gerasse dinamicamente cartões personalizados, em qualquer língua, ou seja, o utilizador seleciona sua língua, seleciona uma imagem, clica no botão para gerar uma mensagem, preenche os campos emissor e destinatário, por fim, clica no botão criar cartão. Pode criar um número indeterminado de cartões e seleciona qual ou quais pretende enviar a alguém.

Comecei por pesquisar APIs gratuitas simples, que oferecessem um serviço permanente em modo continuou. Encontrei a API [*Advice Slip JSON API*](https://api.adviceslip.com/) que gera mensagens para os utilizadores e o serviço da google para tradução de páginas web.

Descrição dos elementos dinâmicos implementados no website

Serviço de tradução da Google que possibilita a escolha da língua do utilizador.



A API [***Advice Slip JSON API***](https://api.adviceslip.com/) com dois campos, um número e uma string. O número identifica a mensagem e a string o conteúdo a ser retornado.

Uma imagem com texto, captura de ecrã, Tipo de letra

Descrição gerada automaticamente

Uma imagem com felino grande, texto, Felinos grandes, mamífero

Descrição gerada automaticamenteLayout

Uma imagem com texto, mamífero, felino grande, Felinos grandes

Descrição gerada automaticamente

Formulário

A página tem três botões, o botão ***Gerar Mensagem*** que, ao clicar, mostra uma mensagem aleatória. O utilizador tem a possibilidade de gerar novas mensagens, até encontrar uma que achar mais adequada. O botão ***Criar Cartão***, tal como o nome indica, cria o objeto cartão e mostra-o na página. O botão ***Enviar Cartão***, envia o cartão personalizado para o destinatário. (esta funcionalidade ainda não está implementada).

Foi inserido dois campos do tipo texto, para o utilizador inserir o nome do destinatário e o nome do emissor.

Por último, foi colocado uma *label* para o utilizador inserir uma imagem, fazendo upload do seu computador.

código

declaração e inicialização de variaveis e constantes

Uma imagem com texto, captura de ecrã, Tipo de letra, número

Descrição gerada automaticamente

Foram declarados oito constantes para guardar as informações dos objetos do DOM (botões, caixas de texto, labels e input-file). Foi criado um objeto card com quatro elementos, nome do emissor, nome do destinatário, mensagem e imagem.

Uma imagem com texto, captura de ecrã, Tipo de letra, file

Descrição gerada automaticamente

Foi criado o array *myArrCards*, para guardar as informações do objeto do tipo card.

Métodos

Localstore

//mostrar os dados contidos no localstorage

 if(localStorage.myArrCards){

arrAux=JSON.parse(localStorage.getItem('myArrCards'));

    }

 for( let i in arrAux)

{

 cards.innerHTML +=

'<div class="card border-success mb-3" style="max-width: 18rem;">' +

'<div class=" lead card-header bg-transparent border-success"> Para: ' + arrAux[i].NomeDestina + '</div>' +

'<div class="card-body text-success ">' +

'<img src="'+ arrAux[i].imagem.src+ '" class="card-img-top" alt="...">' +

'<h5 class="card-title">Message </h5>' +

'<p class="card-text">' + arrAux[i].Mensagem+ '</p>' +

'</div>' +

'<div class="lead card-footer bg-transparent border-success"> De: ' +arrAux[i].NomeEmissor + '</div>' +

'</div>'

    }

Este código serve para mostrar na página todos os cartões que foram gerados e guardados no LocalStore. A imagem não está a funcionar, surge todos os dados menos a imagem.

Criar e ler um ficheiro de imagem

Uma imagem com texto, captura de ecrã, Tipo de letra, número

Descrição gerada automaticamente

Optei pela seleção da imagem a partir de ficheiro ao invés da seleção a partir da internet porque, o objetivo é personalizar os cartões com imagens do próprio utilizador, e a partir do seu smartphone.

Gerar uma mensagem aleatória

Uma imagem com texto, captura de ecrã, Tipo de letra, número

Descrição gerada automaticamente

Esta função serve-se da API para gerar mensagens aleatórias, ou seja, lança um trigger a partir do URL e recebe uma string como retorno. Esse retorno que é depois mostrado na página.

Criar e guardar um objeto do tipo card

Uma imagem com texto, captura de ecrã, Tipo de letra

Descrição gerada automaticamente

Esta função cria um objeto do tipo card, mostra-o na página web, e guarda no array myArrCards usando o método push(), que guarda o objeto no fim da fila.

Estratégia de implementação geral

Foram lecionadas cinco aulas, as quais compareci. Quatro para aprender os conceitos básicos de JavaScript e a última de apresentação dos miniprojectos. Tive o cuidado de visualizar os vídeos antes das lições. Durante as aulas consegui, com facilidade, compreender os assuntos abordados. A programação não se decora, compreende-se e esse objetivo foi conseguido.

Para executar o miniprojecto, comecei por criar uma página web e implementar algum código CSS, mas não gostei do resultado e fui aprender a trabalhar com o botstrap, não achei difícil, mas requer tempo para testar. Como tinha pouco tempo para executar o miniprojecto e o objetivo não era fazer algo muito extenso, optei por agarrar num template trabalhado nas aulas, estudá-lo, e usei-o para desenvolver a minha página web.

A estratégia seguida foi de tentativa-erro, ou seja, a cada inserção de um novo elemento ou alteração do código, testava no navegador. Sempre que não continha erros, fazia um commit para o github, de forma a guardar essa versão.

Principais desafios técnicos e respetivas soluções

O principal desafio foi o layout da página, como não trabalho com estas tecnologias, a última vez que trabalhei com o HTML foi há mais de 20 anos, pelo que senti muita dificuldade em manusear os elementos e o Botstrap nunca tinha trabalhado. “Perdi” que não perdi, mas consumi muito tempo a corrigir erros na parte do layout da página e a tentar que os elementos ficassem como eu idealizava, foi uma batalha que não ganhei, porque ainda assim, o layout da página web não ficou como eu queria.

Coloquei um ficheiro .css e não entendi porque é que a página não o lia. Tive de recorrer ao google para perceber como manusear ficheiros do tipo css em páginas com o Botstrap.

As APIs que inseri compreendi-as, porque são muito simples e fáceis de entender. Entendo o conceito e como se podem usar, contudo, verifiquei nas minhas pesquisas que nem todas são assim, algumas são bastante complexas.

Obstáculos não ultrapassados e desafios futuros

O layout da página como referi acima, não consegui ultrapassar, talvez por me faltar treino nessa área, mas vou continuar a trabalhar no sentido de melhorar.

Para o futuro, que espero não seja muito longínquo, tenciono terminar a app. Gerei alguns cartões personalizados, recortei e enviei por whatsapp para alguns amigos e família, que me deram alguns *inputs* para *upgrade*, nomeadamente a opção de criar cartões com diferentes cores, diferentes *borders,* incluir mais informação, ou seja, que o utilizador também possa acrescentar texto, etc. penso incluir estes melhoramentos ao longo do desenvolvimento continuo da minha página web.

# Perspectiva da visão de cima de um aperto de mãos sobre documentos empresariaisconclusão

Inscrevi-me nesta microcredencial para obter créditos para apresentar na minha escola, como formação docente. Questionei os serviços na tentativa de saber se esta formação estava acreditada para docentes da escola publica e não me souberam responder, remetendo essa resposta para mais tarde.

Uns dias depois, recebi um email dos serviços, com informação que teria de me inscrever na microcredencial até ao final do dia, inscrevi-me sem saber a resposta à questão apresentada, resposta essa, que chegou dois dias depois, e foi negativa. Pensei em enviar um email a desistir do curso, pois não servia os meus propósitos, mas não o fiz.

Estou feliz pela minha escolha, uma vez que tive a oportunidade de aprender um pouco sobre programação web e estou ciente de que, atualmente, é muito utilizada no mundo do trabalho. Estou habituada a lecionar c# e a desenvolver com os alunos aplicações desktop em c#. Após esta formação tenciono aprender mais, e levar estes conhecimentos aos meus alunos.

O objetivo traçado nesta microcredencial foi conseguido com sucesso, embora, tenha ficado aquém do que eu desejaria. Tudo requer tempo, e só se aprende a trabalhar com uma ferramenta, seja ela qual for, com muito treino.

Estou muito grata pelos ensinamentos e certa de que irei continuar a trilhar este caminho.