

Trabalho Prático 2 - Portal de álbuns geolocalizadas com JSONServer e API de mapas



Modelo



Publicado



Editar





Nesse trabalho, vamos dar vida ao layout desenvolvido no trabalho anterior, apresentando informações reais sobre os álbuns, obtidas dinamicamente a partir de um servidor JSON Server que oferece uma API e servirá como back end da aplicação.

Neste trabalho, pode-se evoluir o layout do Trabalho Prático 1 ou começar um projeto de layout do zero. Essa decisão fica a cargo de cada um. Você deverá montar um layout sobre os álbuns e apresentar funcionalidades que apresentamos neste enunciado.



Para montar sua aplicação utilizaremos o JSONServer como repositório para os dados e a API do Mapbox para exibição de mapas com os álbuns geolocalizados. Na sequência, explicamos como o uso destas ferramentas deve ser feito.

IMPORTANTE: O trabalho deve ter foco exclusivamente na construção de um álbum de fotos com informações geolocalizadas e os dados devem ser persistidos em um servidor JSONServer. NÃO SERÃO ACEITOS TRABALHOS COM OUTRA TEMÁTICA E COM OUTRAS API.


JSONServer

O [JSONServer](https://github.com/typicode/json-server)  (<https://github.com/typicode/json-server>) oferece uma API baseada [no padrão RESTful](https://pucminas.instructure.com/courses/148288/modules/items/3058918) (<https://pucminas.instructure.com/courses/148288/modules/items/3058918>) para acesso e manipulação de dados dispostos em um arquivo JSON no servidor. [Um exemplo funcional do JSONServer](https://replit.com/@rommelpuc/JSONServer)  (<https://replit.com/@rommelpuc/JSONServer>) está acessível no ambiente do Replit e você pode utilizar como base para o seu trabalho. Para isso, basta clonar (procedimento de FORK no Replit) e alterar o arquivo db.json, colocando os seus dados.



INFO

- Exemplo de uso: <https://replit.com/@rommelpuc/JSONServer>  (<https://replit.com/@rommelpuc/JSONServer>) (REPLIT)
- Referência da API do JSON Server: <https://github.com/typicode/json-server>  (<https://github.com/typicode/json-server>)

Mapas com Mapbox

A API do [Mapbox](https://www.mapbox.com/)  (<https://www.mapbox.com/>) permite a criação de mapas em sites web, bem como a utilização de uma série de outras funcionalidades associadas como navegação e pesquisa de endereços, para citar algumas. Será necessário criar uma conta no ambiente do Mapbox e criar uma access token a ser incluída na sua aplicação.

INFO

- Exemplo de uso: <https://replit.com/@rommelpuc/Mapbox>  (<https://replit.com/@rommelpuc/Mapbox>) (REPLIT)
- Referência da API: <https://docs.mapbox.com/mapbox-gl-js/api/>  (<https://docs.mapbox.com/mapbox-gl-js/api/>)

Estruturas de Dados

Você deve estruturar os dados para a montagem da sua aplicação. Utilize o projeto de exemplo do JSONServer, faça uma cópia para o seu trabalho. No arquivo **db.json**, você deve criar as suas estruturas de dados.

Para a montagem do seu trabalho, deve haver, pelo menos, três estruturas distintas: (1) álbuns, (2) fotos e (3) destaques. As informações básicas e OBRIGATÓRIAS de cada uma das estruturas mencionadas são apresentadas a seguir

ESTRUTURA ÁLBUNS

- Identificador do álbum
- Título do álbum
- Descrição do álbum
- Localização geográfica do álbum (longitude e latitude)
- Data do álbum
- URL da imagem de capa do álbum

ESTRUTURA FOTOS

- Identificador da foto
- URL da imagem
- Descrição da imagem
- Identificador do álbum ao qual a foto faz parte

ESTRUTURA DESTAQUES

- Identificador do destaque
- Identificado do álbum em destaque
- Texto para destaque (pode ser a própria descrição do álbum ou um texto alternativo)

Detalhamento do escopo

O trabalho deve contemplar as seguintes telas e suas funcionalidades descritas em seguida.

Tela principal

- O cabeçalho traz o nome, o logotipo do site e um menu de opções
- A página deve apresentar um carrossel com fotos registradas na estrutura **DESTAQUES** e que fazem parte dos diversos álbuns registrados na aplicação

ORIENTAÇÃO: use [componente do bootstrap](#)

(<https://getbootstrap.com/docs/5.0/components/carousel/#with-captions>)

- A página deve apresentar uma grade com todos os álbuns registrados na estrutura **ÁLBUNS**. Deve ser apresentada as seguinte informações associada a cada um dos álbuns exibidos na página: uma imagem, o título do álbum, um breve descritivo (20 caracteres) e um link para mais detalhes desse álbum.
- A página deve apresentar um mapa com marcadores sinalizando a localização geográfica de cada um dos álbuns registrados na estrutura **ÁLBUNS**.

ORIENTAÇÃO: utilize o [exemplo do Mapbox](#) ➞ (<https://replit.com/@rommelpuc/Mapbox>) para montar o seu mapa

- Rodapé com informações sobre responsável pelo site (aluno/aluna)

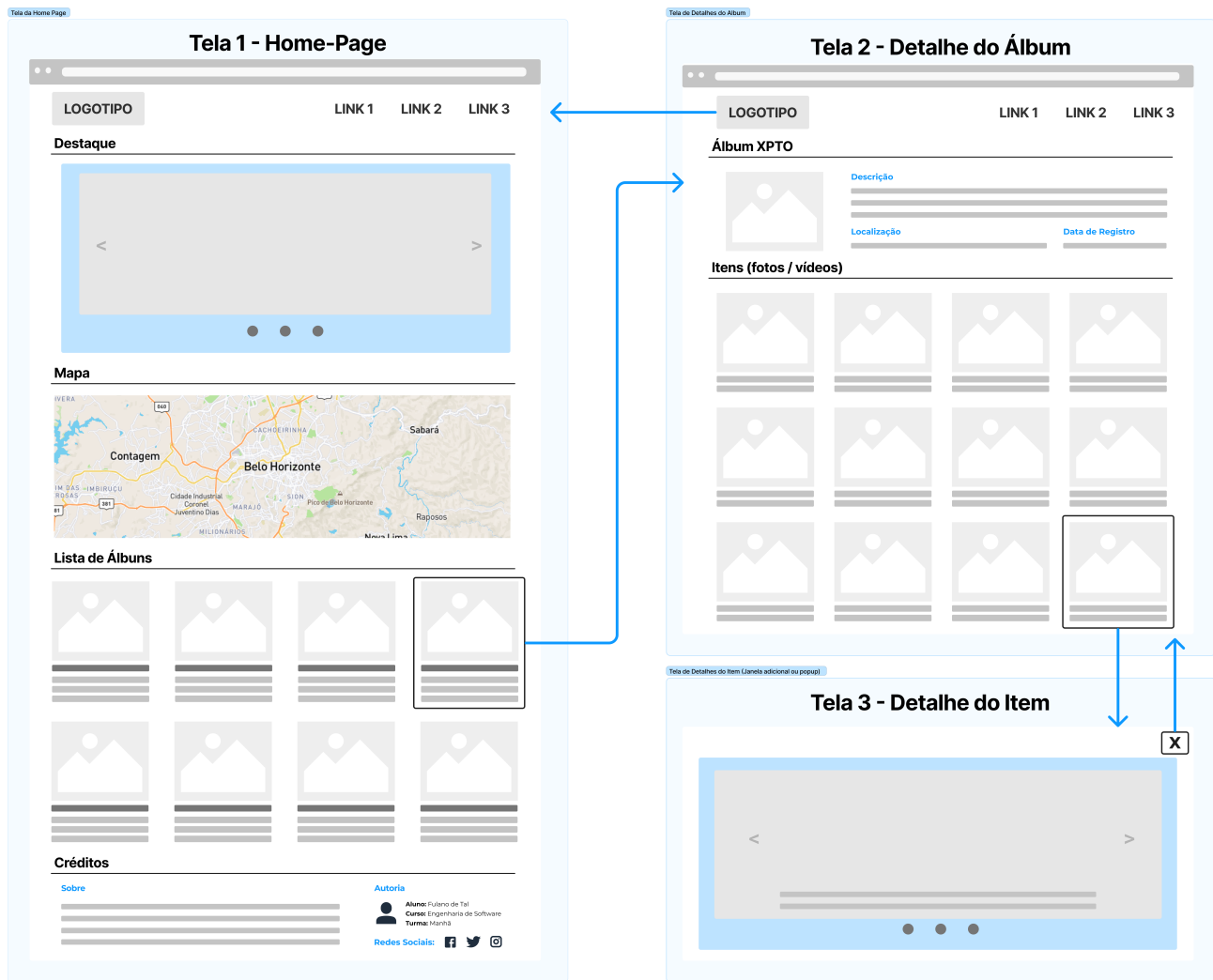
Tela de detalhes do **ÁLBUM**

- Informações gerais do **ÁLBUM** que apresente: foto de capa do album, título, descritivo, geolocalização (endereço ou latitude/longitude) e data de registro
- Deve ser possível marcar o álbum como um destaque para ser exibido na home-page registrando essa informação na estrutura **DESTAQUES** (via requisição de POST na API JSONServer)
- Grade com fotos incluídas no **ÁLBUM** e apresentadas de forma reduzida (thumbnail) com descrição. Os dados devem ser obtidos da estrutura **FOTOS**.

Tela/popup/modal de zoom da FOTO

- A funcionalidade (página independente, popup ou modal) deve apresentar as fotos do álbum na forma de um carrossel, oferecendo botão para exibir a foto anterior e a próxima foto
- A página de detalhes pode ser apresentada de forma independente da tela do álbum ou como um popup (no caso de popup, colocar botão para fechar ou fechar com tecla ESC)
- A página deve apresentar a descrição da foto de forma visível

O fluxo de telas que deve ser implementado na aplicação entregue e os wireframes com os detalhes do escopo são apresentados na imagem a seguir.



Processo de Avaliação

Cada pessoa deve, **OBRIGATORIAMENTE**, enviar um arquivo compactado no formato ZIP com os arquivos necessários para a exibição do site. O aluno poderá também, em caráter **ADICIONAL** e **OPCIONAL**, enviar um link para o site publicado em algum servidor na Internet. A entrega do arquivo ZIP é obrigatória.

A avaliação do trabalho envolverá todos os alunos da disciplina no esquema de avaliação por pares. **Nesse processo, você deve avaliar 3 projetos de outros alunos e isso conta pontos adicionais na disciplina além da pontuação desse trabalho.**

IMPORTANTE: Os trabalhos devem ter identidade visual própria e código fonte original. Cada aluno deve se identificar na Home-Page e fazer o seu trabalho de forma independente. Será realizada uma verificação de originalidade do código fonte e os trabalhos que tiverem partes significativas de código copiado dos demais colegas serão penalizados na nota.

Para avaliar os trabalhos e atribuir pontos, siga os critérios apresentados abaixo. Seja honesto na avaliação. A nota final será a média das avaliações dos 3 outros colegas, sendo cancelada pelos professores envolvidos.

Os critérios detalhados podem ser visualizados [neste link](https://bit.ly/4126iDu) ➡ [_\(https://bit.ly/4126iDu\)_](https://bit.ly/4126iDu), ou na própria RUBRICA desta tarefa.

Você deve utilizar estes critérios para avaliar o trabalho dos colegas e deve ser honesto. Aplicando exatamente os critérios listados para cada requisitos.

Alunos que não fizeram a avaliação dos trabalhos dos colegas corretamente, seguindo os critérios, perderão 5 pontos.

Assim como foi feito no TP1, vamos divulgar, na próxima semana um conjunto de instruções para o processo de avaliação por pares. O foco desta semana é fazer o trabalho seguindo os requisitos detalhados na rubrica.

Pontos 25

Enviando um URL de site ou um upload de arquivo

Tipos de arquivo zip

Entrega	Para	Disponível a partir de	Até
3 dez	Todo mundo	1 nov em 0:00	3 dez em 23:59

Rubrica - TP2 NEW (2023-2)

Você já avaliou alunos com esta rubrica. Qualquer grande alteração poderá afetar os resultados da avaliação.

Critérios	Avaliações					Pts
<p>CRITÉRIO 1 - Formato e responsividade</p> <p>A página deve seguir o wireframe proposto, contendo o header com a logo e menus, e o rodapé contendo a identificação da pessoa responsável pelo trabalho (nome e código de pessoa).</p> <p>R1.1: Layout totalmente funcional no desktop</p> <p>O site apresenta adaptação e adequação de conteúdos para a versão DESKTOP. Componentes se adaptam ao tamanho, imagem segue o padrão do grid, conteúdo ajusta a resolução com uso correto de media queries.</p> <p>R1.2: Layout totalmente funcional no mobile</p> <p>O site apresenta adaptação e adequação de conteúdos para a versão MOBILE. Componentes se adaptam ao tamanho, imagem segue o padrão do grid, conteúdo ajusta a resolução com uso correto de media queries.</p> <p>R1.3: Respiro no design aplicado corretamente</p> <p>O site aplica corretamente respiros (espaçamento correto entre os componentes) no design, tanto no desktop quanto no mobile.</p>	<p>3 pts</p> <p>3 requisitos</p> <p>atendidos</p>	<p>2 pts</p> <p>2 requisitos</p> <p>atendidos</p>	<p>1 pts</p> <p>1 requisito</p> <p>atendido</p>	<p>0 pts</p> <p>Nenhum</p> <p>requisito</p>	3 pts	
<p>CRITÉRIO 2: Utilização correta da API</p> <p>API JSON Server disponível contendo pelo menos 3 endpoints (álbuns, fotos e destaques), com pelo menos 2 itens de destaque, pelo menos 8 álbuns, e pelo menos 8 fotos em cada álbum, conforme critérios abaixo:</p> <p>API deve ser disponibilizada de forma online, hospedada no replit, por exemplo, e deve ser do próprio autor, NÃO PODE utilizar APIs de terceiros. SE NÃO ATENDER ESTE REQUISITO A NOTA DESTE CRITÉRIO DEVE SER ZERADA.</p> <p>R2.1: Home - DESTAQUES</p> <p>Dados exibidos na seção destaques (carousel da homepage) são recuperados dinamicamente, todos obtidos a partir da API JSON Server.</p>	<p>6 pts</p> <p>4</p> <p>requisitos</p> <p>atendidos</p>	<p>4,5 pts</p> <p>3</p> <p>requisitos</p> <p>atendidos</p>	<p>3 pts</p> <p>2</p> <p>requisitos</p> <p>atendidos</p>	<p>1,5 pts</p> <p>1</p> <p>requisito</p> <p>atendido</p>	<p>0 pts</p> <p>Nenhum</p> <p>requisito</p>	6 pts

Critérios	Avaliações					Pts
<p>Existem pelo menos 2 itens de destaque cadastrados na API.</p> <p>R2.2: Home - Mapa Dados exibidos no mapa da homepage (Localização, título e descrição do popup de cada álbum) são recuperados dinamicamente, todos obtidos a partir da API JSON Server. Existem pelo menos 4 álbuns com informações de localização (latitude e longitude) cadastrados na API.</p> <p>R2.3: Home - ÁLBUNS Dados exibidos na lista de álbuns da homepage (capa, título e descrição) são recuperados dinamicamente, todos obtidos a partir da API JSON Server. Existem pelo menos 8 álbuns cadastrados corretamente na API.</p> <p>R2.4: Detalhes do ÁLBUM Dados exibidos na página de detalhes do álbum (capa, título, descrição, localização e fotos) são recuperados dinamicamente, todos obtidos a partir da API JSON Server. Todos os álbuns possuem todos os dados cadastrados (título, descrição, localização e data) e cada álbum possui pelo menos 8 fotos e/ou vídeos cadastrados.</p>						
<p>CRITÉRIO 3: Home - Componente de Destaques</p> <p>Apresentar o componente de álbuns em destaque, conforme critérios abaixo:</p> <p>R3.1: Conteúdo de destaque Todos e apenas os álbuns marcados como destaque são exibidos na galeria de destaques. Se um novo álbum é adicionado ou removido de destaques, a galeria reflete corretamente a nova lista de destaques, exibindo apenas as fotos da capa dos álbuns marcados como destaque.</p> <p>R3.2: Exibe corretamente os destaques no carousel Tanto no desktop quanto no mobile a galeria de destaques possui o layout correto sem quebra e funciona corretamente passando as fotos</p>	<p>2 pts 2 requisitos atendidos</p>	<p>1 pts 1 requisito atendido</p>		<p>0 pts nenhum requisito</p>		<p>2 pts</p>

Critérios	Avaliações				Pts
<p>CRITÉRIO 4: Home - Componente Mapa</p> <p>Apresentar o componente do mapa com os marcadores, conforme critérios abaixo:</p> <p>R4.1: Exibir corretamente mapa com todos os marcadores visíveis O mapa é exibido corretamente com todos os marcadores dos álbuns visíveis</p> <p>R4.2: Utiliza corretamente a API MapBox O mapa é montado com a utilização da api Mapbox utilizando uma token válida</p> <p>R4.3: Estilo dos marcadores Ao clicar em um marcador exibe corretamente um popup com os dados do álbum (ex: nome, descrição) e um link que ao clicar, permite abrir a página de detalhes do álbum (detalhes.html) com o identificador do álbum sendo passado via URL (query string).</p>	<p>4 pts 3 requisitos atendidos</p>	<p>2 pts 2 requisitos atendidos</p>	<p>1 pts 1 requisito atendido</p>	<p>0 pts nenhum requisitos atendido</p>	<p>4 pts</p>
<p>CRITÉRIO 5: Home - Listagem de Álbuns</p> <p>Apresentar a listagem de álbuns na home page, de forma responsiva, conforme critérios abaixo:</p> <p>R5.1: Montar corretamente os cards Cada item da seção de álbuns deve apresentar a imagem representativa (capa do álbum), o título destacado em uma linha e a descrição do álbum com 2 a 4 linhas.</p> <p>R5.2: Link para detalhes do álbum Ao clicar no card do álbum na listagem, o usuário deve ser direcionado para uma página de detalhes do álbum (exemplo: detalhes.html), com o identificador do álbum sendo passado via URL (query string).</p> <p>R5.3: Layout e responsividade dos cards no container Os cards são exibidos corretamente, de forma responsiva, com espaçamento, cards do mesmo tamanho, mantendo os cards bem distribuídos em todos os tamanhos de tela.</p>	<p>4 pts 3 requisitos atendidos</p>	<p>3 pts 2 requisitos atendidos</p>	<p>1,5 pts 1 requisito atendido</p>	<p>0 pts nenhum requisito atendido</p>	<p>4 pts</p>

Critérios	Avaliações				Pts
<p>CRITÉRIO 6: Detalhes do Álbum</p> <p>Ao clicar em um álbum na home, o site deve exibir os detalhes do álbum, conforme critérios abaixo:</p> <p>R6.1: Exibir corretamente os dados do álbum</p> <p>Página de detalhes que obtém o identificador (id) do álbum por meio de parâmetro via URL (query string), carrega os dados do referido álbum (estrutura ÁLBUNS) e suas fotos (estrutura FOTOS) via API JSONSever e exibe as informações detalhadas do álbum numa primeira parte.</p> <p>R6.2: Exibir corretamente as fotos do álbum</p> <p>Exibe as fotos do álbum em um grid contendo a foto (em tamanho reduzido, formato de card) e uma breve descrição (até 2 linhas) de acordo com dados recuperados da API JSON Server.</p> <p>R6.3: Opção para marcar o álbum como destaque</p> <p>Existe uma opção para marcar/desmarcar o álbum como destaque via requisição POST/DELETE para a API JSONServer, na estrutura de DESTAQUES. Ao selecionar esta opção, a foto de capa do álbum passa a ser exibida na galeria de destaques da página inicial.</p>	<p>4 pts</p> <p>3 requisitos</p> <p>atendidos</p>	<p>2 pts</p> <p>2 requisitos</p> <p>atendidos</p>	<p>1 pts</p> <p>1 requisito</p> <p>atendido</p>	<p>0 pts</p> <p>nenhum</p> <p>requisito</p>	<p>4 pts</p>
<p>CRITÉRIO 7: Galeria de fotos do álbum</p> <p>Na página de detalhes do álbum, ao clicar em uma foto, ela deve ser exibida ampliada, em formato de galeria (carousel), sendo possível na própria galeria navegar pelas fotos (próximas e anteriores), conforme critérios abaixo:</p> <p>R7.1: Exibir as fotos do álbum ampliadas</p> <p>A partir da página de detalhes do álbum, ao clicar em uma foto, deve permitir a exibição ampliada (zoom) da foto em página independente ou um popup com informações adicionais sobre a foto.</p> <p>R7.2: Permite navegação pelas fotos do álbum ampliadas</p> <p>Após clicar em uma foto e exibir</p>	<p>2 pts</p> <p>2 requisitos</p> <p>atendidos</p>	<p>1 pts</p> <p>1 requisito</p> <p>atendido</p>	<p>0 pts</p> <p>nenhum</p> <p>requisitos</p>	<p>2 pts</p>	

Critérios	Avaliações			Pts
ampliada, deve permitir passar as fotos (próximas e anteriores) exibindo todas as fotos do álbum, mantendo o formato ampliado. Deve permitir também o fechamento do popup ou da página retornando para a lista de fotos do álbum.				
Total de pontos: 25				