

Programação Lógica

Programação Web em Prolog

Alexsandro Santos Soares

prof.asoares@gmail.com

2 de outubro de 2020

Bacharelado em Sistemas de Informação

Faculdade de Computação

Universidade Federal de Uberlândia

Sumário

Introdução

Usando o CURL para testar a API

Para saber mais

Referências bibliográficas

Introdução

O projeto **Bookmarks** permite ao usuário armazenar seus links prediletos juntamente com uma descrição da página.

Para começar é necessário definir a interface do serviço (**API**), informando as rotas , os métodos HTTP usados para cada uma e uma breve descrição da funcionalidade.

Além disso as rotas para as páginas do front-end também devem ser descritas juntamente com uma descrição.

Nos próximos slides a definição do site do nosso exemplo.

Método	Rota	Descrição
GET	/api/v1/bookmarks/	Retorna uma lista com todos os <i>bookmarks</i> .
GET	/api/v1/bookmarks/1	Retorna o <i>bookmark</i> com ID 1 ou erro 404 caso o <i>bookmark</i> não seja encontrado.
POST	/api/v1/bookmarks	Adiciona um novo <i>bookmark</i> . Os dados deverão ser passados no corpo da requisição no formato JSON. Um erro 400 (BAD REQUEST) deve ser retornado caso a URL não tenha sido informada.
PUT	/api/v1/bookmarks/3	Atualiza o <i>bookmark</i> de ID 3. Os dados deverão ser passados no corpo da requisição no formato JSON.
DELETE	/api/v1/bookmarks/5	Apaga o <i>bookmark</i> de Id 5.

Rota	Descrição
/	retorna a página de entrada listando todos os <i>bookmarks</i> .
/bookmark/	retorna a página para o cadastro do <i>bookmarks</i> .
/bookmark/editar/2	retorna a página para alteração do <i>bookmark</i> com ID 2 ou erro 404 caso o <i>bookmark</i> não seja encontrado.

Sempre que o usuário fizer alguma operação de alteração ou remoção de algum *bookmark*, ele será redirecionado para a página de entrada.

Usando o CURL para testar a API

Usando o CURL para testar a API

O CURL (*Client URL*) é um programa de linha de comando que nos permite testar e obter dados de uma API.

Podemos usar o CURL para enviar pedidos **GET**:

- Para obter a lista de todos os bookmarks

```
curl -v http://localhost:8000/api/v1/bookmarks/
```

- Para obter uma tupla específica, por exemplo, aquela com id 1

```
curl -v http://localhost:8000/api/v1/bookmarks/1
```

Usando o CURL para testar a API

Para enviar JSON com **POST**

- Para obter a lista de todos os bookmarks

```
curl -v -d '{"titulo":"Exemplo", "url":"https://exemplo.br/"}'  
-H 'Content-Type: application/json'  
http://localhost:8000/api/v1/bookmarks/
```

- Se o existir arquivo com o JSON, por exemplo, `tupla.json`, com o conteúdo

```
{  
  "titulo":"Exemplo",  
  "url":"https://exemplo.br/"  
}
```

Podemos enviá-lo com

```
curl -v -d @tupla.json -H 'Content-Type: application/json'  
http://localhost:8000/api/v1/bookmarks/
```

Usando o CURL para testar a API

Para enviar JSON com PUT

- Para obter a lista de todos os bookmarks

```
curl -d '{"id": "5", "titulo":"Exemplo", "url":"https://exemplo.br/"}'  
-H 'Content-Type: application/json'  
-X PUT http://localhost:8000/api/v1/bookmarks/
```

- Se o existir arquivo com o JSON, podemos enviá-lo com

```
curl -v -d @tupla.json -H 'Content-Type: application/json'  
-X PUT http://localhost:8000/api/v1/bookmarks/
```

Usando o CURL para testar a API

Para enviar um pedido com **DELETE**, por exemplo, para remover a tupla com id igual a 5:

```
curl -v -X DELETE http://localhost:8000/api/v1/bookmarks/5
```

Para saber mais

- SWI-Prolog HTTP support library. Disponível em www.swi-prolog.org/pldoc/doc/_SWI_/library/http/index.html.

Referências bibliográficas

- Anne Ogborn. *Creating Web Applications in SWI Prolog*. Disponível em <http://www.pathwayslms.com/swipltuts/html/index.html>