

Instituto de Ciências Matemáticas e de Computação

Universidade de São Paulo - USP

Introdução ao Desenvolvimento Web

ENTREGA 03:

Roteiro de Testes

Docente: Dilvan Moreira

Camila Stenico dos Santos, 8530952

Henrique F. M. Freitas, 8937225

Lucas de Almeida Carotta, 8598732

Lucas Antognoni de Castro, 8936951

São Carlos

Julho de 2017

Node.js, CouchDB e Express

A aplicação na fase final utiliza um servidor node.js, conectado ao banco de dados CouchDB para realizar requisições e manipulações nos dados, assim como armazená-los.

Juntamente ao Node, foi instalado o express para facilitar a manipulação de dados, e ferramentas através do *npm*, como body-parser para tratamento de dados, nano para conectar com o couchDB e models para facilitar a busca no banco de dados.

Para iniciar o servidor basta digitar os seguintes comandos:

```
npm update
```

```
npm start
```

Cadastro

Para que o cadastro de novos usuários seja realizado deve-se clicar na opção de login e descer procurar a opção 'Clique Aqui' logo ao lado de 'Não possui cadastro?' no rodapé das informações de login de usuários. Nisso os dados necessários para o cadastro aparecerão na tela.

O cadastro é realizado através da inserção de um registro no CouchDB. Para tal o usuário preencherá os campos indicados com os respectivos valores pedidos. Após o cadastro e caso não ocorra nenhum erro de informações -- como senhas que não coincidem no campo de confirmação -- o usuário receberá uma mensagem avisando-o do ocorrido.

Os usuários são mostrados na tela de Administrador que só é possível chegar nela através do endereço http://localhost:3000/login_cadastro/adm.html, uma vez que não se chega à ela através do site por hiperlinks.

Login

O login é realizado através da verificação do usuário no sistema do couchDB comparando os dados digitados com os armazenados pelo sistema. Como o login é dado através do

e-mail e senhas cadastrados no site, caso outros dados que não os salvos no sistema sejam utilizados haverá uma mensagem de erro informando sobre o ocorrido.

Vendas

Nosso aplicativo abrange o atendimento de serviços e de venda de produtos, cada qual em suas respectivas páginas. Os pedidos são armazenados num carrinho utilizando *local storage* para permitir que todos os .html tenham acesso ao mesmo valor de variável sem que o *reload* do script apague o que foi armazenado. O usuário poderá fazer alterações de remoção e inserção no carrinho - ao finalizar seu pedido, basta utilizar o checkout.

Gerenciamento de Animais

O usuário pode cadastrar seus animais através de um formulário, o qual é enviado via *post* para o servidor, sendo salvo todos os dados. Cada animal tem o email de seu dono salvo com seus dados, para identificação do usuário. Estão presentes botões de deletar e exibir animais, mas devido a erros no servidor não retornam resultado.

Gerenciamento de Produtos

Para que o administrador gerencie os produtos, é feita uma página com formulário de cadastro com nome, descrição, preço, quantidade em estoque e total vendido. A imagem, antes carregada em formato de *FILE*, foi trocada por um *input text* devido a falta de suporte encontrada para submeter uma imagem juntamente com o formulário. Como solução, o administrador deverá submeter a imagem como URL. Os dados são submetidos para o servidor com o método POST e salvos no couchDB.

Gerenciamento de Serviços

Para que o administrador gerencie os serviços, é feita uma página com formulário de cadastro com nome do serviço, descrição e preço. A imagem, antes carregada em formato de *FILE*, foi removida por decisão de deixar a imagem de um serviço a cargo do programador responsável pelo UI. Os dados são submetidos para o servidor com o método POST e salvos no couchDB.

Telas de sucesso

Após a submissão dos formulários citados anteriormente, foi tentado realizar um redirect da página para exibir um template de sucesso, mas os comandos

```
response.sendFile(path.join(__dirname+'/forms/sucessAnimal.html'));
```

```
return redirect(/forms/sucessAnimal.html)
```

e algumas outras variações não obtiveram êxito, sempre permanecendo a página original dos formulários.

Agendar Horários

Ao abrir a página de Agendamento, o usuário poderá verificar os horários disponíveis em uma data específica para agendar um serviço para um determinado animal. Caso o horários esteja livre, o cliente deve clicar no botão “Agendar” que gerará um formulário a ser preenchido e submetido. Serviços agendados são armazenados no CouchDB que criará documentos para cada um destes. Os horários são disponibilizados em uma tabela e são recuperados do banco de dados a partir da data escolhida no calendário.

Gerenciar Horários

A gerência de horários é feita pelos administradores do sistema. As funcionalidades são as mesmas que a da seção Agendar Horários, entretanto, o administradores podem remover serviços em uma data específica ao escolher tal data no calendário, clicar no botão “Deletar”, e preencher o formulário com o nome do animal e data, cujo serviço associado irá ser removido. Assim, na ausência de erros, o documento do serviço no CouchDB será removido.