Página de Listagens Ativas

Página de Listagem

Começando

Baixe o código de distribuição em https://cdn.cs50.net/web/2020/spring/projects/2/commerce.zip e descompacte-o.

No seu terminal, vá para o diretório commerce.

Execute python manage.py makemigrations auctions para criar migrações para o aplicativo de leilões.

Execute python manage.py migrate para aplicar as migrações ao seu banco de dados.

Entendimento

No código de distribuição há um projeto Django chamado commerce que contém um único aplicativo chamado auctions.

Primeiro, abra auctions/urls.py, onde a configuração de URL para este aplicativo é definida. Note que já escrevemos algumas URLs para você, incluindo uma rota de índice padrão, uma rota /login, uma rota /logout e uma rota /register.

Dê uma olhada em auctions/views.py para ver as visualizações associadas a cada uma dessas rotas. A visualização index por enquanto retorna um template index.html principalmente vazio. A visualização login\_view renderiza um formulário de login quando um usuário tenta ACESSAR a página. Quando um usuário envia o formulário usando o método de solicitação POST, o usuário é autenticado, conectado e redirecionado para a página de índice. A visualização logout\_view desconecta o usuário e o redireciona para a página de índice. Finalmente, a rota de registro exibe um formulário de registro para o usuário e cria um novo usuário quando o formulário é enviado. Tudo isso é feito para você no código de distribuição, então você deve ser capaz de executar o aplicativo agora para criar alguns usuários.

Execute python manage.py runserver para iniciar o servidor web Django e visite o site em seu navegador. Clique em "Registrar" e registre-se para uma conta. Você deve ver que agora está "Logado como" sua conta de usuário, e os links no topo da página mudaram. Como o HTML mudou? Dê uma olhada em auctions/templates/auctions/layout.html para o layout HTML deste aplicativo. Note que várias partes do modelo estão envolvidas em uma verificação para se o usuário está autenticado, para que diferentes conteúdos possam ser renderizados dependendo se o usuário está conectado ou não. Você pode modificar este arquivo se desejar adicionar ou modificar algo no layout!

Finalmente, dê uma olhada em auctions/models.py. Aqui é onde você irá definir quaisquer modelos para sua aplicação web, onde cada modelo representa algum tipo de dado que você deseja armazenar em seu banco de dados. Nós começamos você com um modelo de Usuário que representa cada usuário da aplicação. Porque herda de AbstractUser, ele já terá campos para um nome de usuário, email, senha, etc., mas você pode adicionar novos campos à classe Usuário se houver informações adicionais sobre um usuário que você deseja representar. Você também precisará adicionar modelos adicionais a este arquivo para representar detalhes sobre listagens de leilão, lances, comentários e categorias de leilão. Lembre-se de que cada vez que você alterar algo em auctions/models.py, você precisará primeiro executar python manage.py makemigrations e então python manage.py migrate para migrar essas mudanças para o seu banco de dados.

Especificações

Complete a implementação do seu site de leilões. Você deve cumprir os seguintes requisitos:

Modelos: Sua aplicação deve ter pelo menos três modelos além do modelo User: um para listagens de leilões, um para lances e um para comentários feitos em listagens de leilões. Cabe a você decidir quais campos cada modelo deve ter e quais devem ser os tipos desses campos. Você pode ter modelos adicionais se desejar.

Criar Listagem: Os usuários devem ser capazes de visitar uma página para criar uma nova listagem. Eles devem poder especificar um título para a listagem, uma descrição baseada em texto e qual deve ser o lance inicial. Os usuários também devem poder opcionalmente fornecer uma URL para uma imagem para a listagem e/ou uma categoria (por exemplo, Moda, Brinquedos, Eletrônicos, Casa, etc.).

Página de Listagens Ativas: A rota padrão de seu aplicativo web deve permitir que os usuários vejam todas as listagens de leilão atualmente ativas. Para cada listagem ativa, esta página deve exibir (no mínimo) o título, descrição, preço atual e foto (se houver uma para a listagem).

Página de Listagem: Clicar em uma listagem deve levar os usuários para uma página específica daquela listagem. Naquela página, os usuários devem poder ver todos os detalhes sobre a listagem, incluindo o preço atual para a listagem.

Se o usuário estiver conectado, o usuário deve ser capaz de adicionar o item à sua "Lista de Observação". Se o item já estiver na lista de observação, o usuário deve ser capaz de removê-lo.

Se o usuário estiver conectado, o usuário deve ser capaz de fazer um lance no item. O lance deve ser pelo menos tão grande quanto o lance inicial e deve ser maior do que qualquer outro lance que tenha sido feito (se houver). Se o lance não atender a esses critérios, o usuário deve ser apresentado com um erro.

Se o usuário estiver conectado e for aquele que criou a listagem, o usuário deve ter a capacidade de "encerrar" o leilão desta página, o que torna o maior licitante o vencedor do leilão e torna a listagem não mais ativa.

Se um usuário estiver conectado em uma página de listagem fechada e o usuário tiver ganho aquele leilão, a página deve informar isso.

Usuários que estiverem conectados devem ser capazes de adicionar comentários à página de listagem. A página de listagem deve exibir todos os comentários feitos na listagem.

Lista de Observação: Usuários que estiverem conectados devem ser capazes de visitar uma página de Lista de Observação, que deve exibir todas as listagens que um usuário adicionou à sua lista de observação. Clicar em qualquer uma dessas listagens deve levar o usuário para a página daquela listagem.

Categorias: Os usuários devem ser capazes de visitar uma página que exibe uma lista de todas as categorias de listagem. Clicar no nome de qualquer categoria deve levar o usuário para uma página que exibe todas as listagens ativas nessa categoria.

Interface de Administração do Django: Através da interface de administração do Django, um administrador do site deve ser capaz de visualizar