Test 5.3

Customize Admin Panel for Your Django Project

Task: Customize the Django admin panel for the project implemented in Test 5.1. Ensure all models are registered and configured for optimal data management.

Instructions:

1. **Admin Panel Customization**

   - Open the `admin.py` file in each Django app of your project.
   - Register each model with the Django admin interface using `admin.site.register(ModelName)`.
   - Customize the admin panel for each model to enhance usability and functionality.

2. **Enhanced Features**

   - Implement list display options (`list_display`) to show important fields of each model in the admin panel list view.
   - Configure search (`search_fields`) and filtering (`list_filter`) options to facilitate easy data retrieval.
   - Utilize inline editing (`inlines`) for related models where applicable to streamline data entry.

3. **User Experience**

   - Improve the user experience by adding custom actions (`actions`) for bulk operations on model instances.
   - Customize form fields (`fields` and `fieldsets`) to organize and present data entry forms logically.

4. **Submission**

   - Take screenshots of your customized Django admin panel for each registered model.
   - Compile these screenshots into a PDF document showcasing the admin panel configurations.
   - Submit the PDF document containing screenshots in the WhatsApp group.

Grading:

- Your submission will be graded based on the following criteria:
  - **Admin Panel Customization**: Effectiveness and creativity in enhancing the Django admin interface.
  - **User Experience**: Clarity and usability improvements demonstrated through admin panel customization.
  - **Submission Quality**: Clarity and completeness of the submitted PDF document with screenshots.

Note: This test evaluates your ability to optimize the Django admin interface for efficient data management and user interaction, reflecting real-world application development practices.