Comic Book Generator Documentation

This is a documentation for the given Django-based Python code, which is a comic book generator that takes user inputs to create a captivating superhero comic book with generated images and narration.

Overview

The code uses the following third-party libraries:

- Django: A high-level Python web framework for building web applications.
- openai: The official OpenAI library for interacting with the OpenAI API.
- deepl: A Python library for interacting with the DeepL API for translating text.
- PIL (Python Imaging Library): A library for handling image processing tasks.
- requests: A library for making HTTP requests.

The code includes the following components:

1. A Django view for handling user inputs and generating comic books.
2. Functions for interacting with the OpenAI API and DeepL API.
3. Functions for processing generated images and adding narration.
4. A main function for orchestrating the entire process.

Code Walkthrough

Django Views

There are two main views in this code: HomeView and ImagesView.

HomeView

HomeView is a Django View class that inherits from LoginRequiredMixin and View. The user must be logged in to access this view. The login_url attribute is set to '/account'.

- get(): Renders the "main/home.html" template when a GET request is made.
- post(): Handles the POST request when the user submits the form with the following inputs:
- story: The user's story input.
- hero: The user's main character input.
- style: The user's chosen style for the comic book.
- prompt: The user's prompt for the image generation.
- index: The index of the current panel being generated.
- panels: The generated panels with art and narration.
- chosen: The user's chosen image for the current panel.
- The function processes the user inputs, generates the comic book panels, and renders the "main/images.html" template with the generated images and narration.

ImagesView

ImagesView is a Django View class that inherits from LoginRequiredMixin and View. The user must be logged in to access this view.

- get(): Renders the "main/images.html" template when a GET request is made.

Helper Functions

The following functions handle the interaction with the OpenAI API, DeepL API, and image processing:

- insert_narration(image_path, text): Adds narration text to the given image file.
- getImage(prompts, apiKey): Generates an image using the OpenAI API with the given prompts.
- askGPT(text): Sends a text prompt to the OpenAI API and returns the generated text.
- translateText(story, main_character): Translates the input story and main character to English using the DeepL API.
- receiveResponse(panel_response): Processes the panel response from the OpenAI API and returns a list of panels.
- main(Story, Comic_style, Main_character): The main function that orchestrates the entire process of generating the comic book.

Usage

To use this comic book generator, users must log in to access the HomeView. They can input their story, main character, and chosen style for the comic book. The generator will then create the comic book panels with generated images and narration. Users can choose the best image for each panel and view the final comic book with their chosen images and narrations.

Dependencies

To run this code, you need to install the following Python packages:

- Django
- openai
- deepl
- Pillow (PIL)
- requests