Postcard Generation Assessment Report

This assessment requires you to generate a specific object of a postcard. You'll find briefs (which contain the text required to be included in the image) and a sample postcard from the client (it is the overall output) to guide you. Here's a breakdown of the postcard components you might work on:

1. Template: Design a layout using the color scheme from the client's website. Please ensure it doesn't include any placeholders.
2. Services Provided: Compile a list of the client's services, and you may add icons to symbolize each service.
3. Coupons: Either create coupons or generate a template that can be used to replicate coupons based on the briefs.
4. Contact Information: Include the client's phone number, website, and address.

Objective

The goal of this assessment was to generate a postcard image based on specific requirements, including services provided, coupons, and contact information, using a template designed according to the client's guidelines. The task involved automating the process of generating the postcard using an external API and creating a frontend interface for users to input custom data.

Steps Taken

1. Exploring Different API Options

Initially, I explored multiple APIs to automate image generation, starting with Canva's API. After experimenting with it, I found that Canva did not meet the customization needs for this assessment. This led to further research and the discovery of Renderform API, which was more suitable for generating custom postcard images based on templates.

Decision to switch from Canva to Renderform API

I concluded that Renderform API provided better flexibility for handling templates and adding custom data such as services, coupons, and contact information.

2. Creating a Base Template on Renderform.io

Next, I created a base template directly on Renderform.io. This template would serve as the structure for all future postcard images generated.

Key components included in the template:

- Header with the business's tagline.

- Section for listing services provided.

- Coupons area with customizable offers.
- Contact information, including phone numbers and website links.

3. Experimenting with the Renderform API

After setting up the template, I used Postman to test different API requests and understand how to input custom data into the Renderform API. This helped me determine the correct JSON payload structure for dynamic elements like taglines, services, and coupon details.

Learnings:

- The API accepts a JSON payload for customizing the template.
- Postman allowed me to refine the API call to handle flexible input for generating images based on client briefs.

4. Automating the API Call Using Django

To automate the postcard generation process, I chose Django as the backend framework. I integrated the Renderform API with Django, using a POST request to send the customized data and generate the image automatically. The user would be able to input custom data via a web interface, which would trigger the API call and return the generated postcard image.

Django's role:

- Managed form submission and ensured the data collected from the frontend was formatted correctly for the API.
- Handled the API call to Renderform and retrieved the generated image for display to the user.

5. Building a User-Friendly Frontend

For the frontend, I created an HTML form that allows users to input all the necessary details to generate the postcard. This included fields for:

- Business tagline.
- List of services.
- Coupon details (title, discount, conditions).
- Contact information.

The form was linked to Django, ensuring that the data entered was used to generate the postcard image dynamically. The design was kept simple and intuitive, so users could easily fill in the information without needing to know the underlying mechanics.

Frontend capabilities:

- The form validates phone number and ratings fields, ensuring valid inputs before submission.

- Submitting the form triggers the image generation process and returns the postcard image.

Summary of Achievements

- Created a reusable template for generating postcards using Renderform.io.
- Automated the process of submitting custom data to the API via a Django backend.
- Built a user-friendly HTML frontend to allow seamless data input for generating the postcard images.
- Ensured that users can easily customize the postcard content, including services, coupons, and contact details, or use default values where appropriate.

Scope of Improvement:

- Database can be added to store the response from the api call.
- Default information can be stored for the user so that the user can edit only the coupons or services that have changed which the business has to offer

Conclusion

This project successfully demonstrated the ability to create a fully automated system for generating custom postcards based on client briefs. The integration of Renderform API with Django, alongside a straightforward frontend, ensures a reusable solution that can be adapted for various client needs.