**Review Report: Web Application Considerations for Pet Heaven Society**

1. Design Decisions

The Pet Heaven Society web application is designed to fulfill the organization’s need by integrating the following core functions.

* Information Pages: The homepage clearly communicates the society purpose, facilities and pet-related information.
* Pet Listings: A dynamically updated page lists all available pets for adoption, including their details and images.
* Pet Release Form: Pet owners who wish to surrender their pets can submit an online form which will be emailed to the admin Gmail account.
* Adoption Request Form: Interested adopters can fill out a form specifying the pet they wish to adopt which will be emailed to the email admin Gmail account.
* Email integration: Submitted forms are automatically sent to the society’s administrator for processing.

1. Screen Layout

The layout is structured for ease of navigation and accessibility.

* Homepage: Features a hero section with a welcoming message, followed by sections on the society’s mission and facilities.
* Pet Listings page: A grid layout with pet images, names and brief description. Clicking a pet opens a pop-up dialog which allows the user to indicate that they are interested to adopt the pet.
* Pet Release page: A structured form requiring pet details, medical history and owner’s information.
* Contact Page: Includes an inquiry form and direct contact details for the society.

1. Styles

* A clean and minimalistic design is adopted, ensuring a visually appealing and user-friendly interface.
* Tailwind CSS is used for efficient styling and responsiveness.
* A soft color palette with pastel tones enhances readability and conveys a welcoming atmosphere.
* Cards and models are used for pet details to improve user interaction.
* Consistent typography with readable fonts ensures accessibility.

1. Backend and database

This application implements a REST API using python and the user, pet details are stored in a JSON file.

* REST API is implemented using Flask and python
* Flask\_cors is used to enable communication between the REACT application and backend API.
* SMTPLIB python library is used to send email from the backend.
* Sender email and password are hardcoded in the main.py file. An application password is generated to use the email.

Conclusion

The Pet Heaven Society web application is designed to be user-friendly, informative and functional. While it lacks backend interaction, it successfully achieves it goal through intuitive UI/UX design, effective from submissions and clear communication of the society’s purpose. Future improvements could include better formatting for the emails sent to the admin and membership handling.

Deployed react project link: <https://merry-bombolone-a44e3a.netlify.app/AboutUs>

* Backend cannot be deployed because the deployment service (Render) is always failing for my API deployment.

A screen shot of a black background

AI-generated content may be incorrect.

* Please run (npm install) for react project to include all the necessary node modules and use (npm run dev) to run the project.
* Please run (pip3 install requirements.txt) for backend python api and use (python main.py) to start the API.