Design Specification: imagRBG

https://github.com/maraljr/Maral_Project2.git

1. General Description of the Project

imagRBG is a Python-based application that generates customizable color palettes from user-provided images. Aimed at designers, artists, and developers, the application extracts dominant colors from images to inspire and inform creative projects. By analyzing an image, imagRBG identifies the most prevalent colors and presents them with detailed information, enhancing the user's ability to incorporate these colors into their work.

Key Features:

- **Dominant Color Extraction:** Identifies the most significant colors in an image.
- **Customizable Output:** Allows users to specify the number of colors in the palette.
- **Detailed Color Information:** Provides color names, HEX codes, RGB values, and CMYK values.
- **Post-Processing Options:** Enables adjustments to brightness and saturation.
- **Ordered Color Presentation:** Displays colors in meaningful sequences (e.g., dominance, hue).
- **Palette Saving and Exporting:** Offers options to download palettes and save them for future use.

External Libraries and Packages:

- **Pillow (PIL):** Image loading and basic processing.
- **NumPy:** Efficient numerical computations.
- **scikit-learn:** K-Means clustering for color quantization.
- **Webcolors:** Mapping RGB values to official color names.
- **Streamlit (Version 2):** Interactive web-based GUI development.

User Interface:

- **Graphical User Interface (GUI):** Developed using Streamlit for intuitive interaction.
- **Command-Line Interface (CLI):** Minimal operation support for non-GUI environments.

Future Enhancements:

- **Remote Control/API Integration:** Potential for integration with other applications.
- **Image Capture:** Incorporating webcam or mobile device image capture.

2. Task Vignettes (User Activity Flow)

Vignette 1: Generating a Basic Color Palette

Narrative:

Emma, a graphic designer, wants to create a color scheme inspired by a photograph. She opens imagRBG and selects her image using the "Upload Image" button. She inputs "5" for the number of dominant colors and clicks "Generate Palette." The application displays the top five colors with their HEX and RGB values. Emma downloads the palette as an image file for her design software.

Technical Details:

- **User Inputs:**
  - Image file via upload.
  - Number of colors (integer).
- **Processing Steps:**
  - Load image with Pillow.
  - Resize for efficiency.
  - Extract colors using K-Means clustering.
  - Convert RGB to HEX codes.
- **Outputs:**
  - Color swatches displayed in order of dominance.
  - Color codes presented alongside swatches.
  - Option to download the palette image.

Vignette 2: Customizing Color Information and Adjustments

Narrative:

Carlos, a web developer, needs CMYK values and lighter shades. He accesses "Advanced Options" in imagRBG, enables CMYK display, and adjusts brightness by +10%. The updated palette shows adjusted colors with HEX, RGB, and CMYK codes. Carlos copies the codes into his style guide.

Technical Details:

- **User Inputs:**
  - Selection of CMYK display.
  - Brightness adjustment slider.
- **Processing Steps:**
  - Adjust RGB values for brightness.
  - Convert adjusted RGB to CMYK.
  - Update display with new values.

- **Outputs:**
    - Adjusted color swatches.
    - Comprehensive color information.

---

3. Technical Flow

Overall Data Flow Description

Stages:

1. **User Interaction Blocks:**
    - **Image Input Block:** User uploads or captures an image.
    - **Parameter Input Block:** User specifies settings (number of colors, adjustments).
2. **Processing Blocks:**
    - **Image Preprocessing Block:** Resizes image, converts color space.
    - **Color Extraction Block:** Identifies dominant colors using K-Means clustering.
    - **Color Adjustment Block:** Applies brightness/saturation changes.
    - **Color Mapping Block:** Converts RGB values to HEX, CMYK, assigns color names.
3. **Output Blocks:**
    - **Display Block:** Shows color palette and information.
    - **Save/Export Block:** Allows downloading palette and data.

User Interaction Points

- **Inputs:**
    - Upload/select images.
    - Enter parameters (number of colors, adjustments).
- **Outputs:**
    - Display generated palettes.
    - Download files.