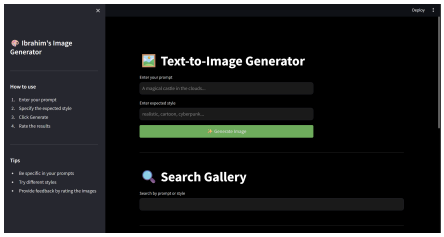
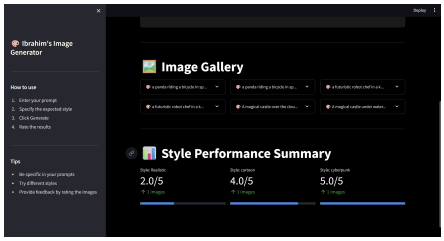
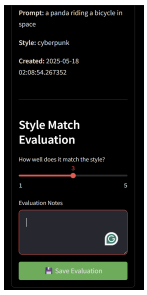
AI Image Generator

By Ibrahim Ahmed Khan

Github Link to the codebase: https://github.com/ibrahim-Cypher10/ai-image-generator.git

A powerful text-to-image generation application built with Streamlit and Stable Diffusion, featuring a modern dark theme interface and comprehensive image management system.

UI Showcase

Home & Prompt	Gallery & Summary	Evaluation	Gallery Example
			

Features

1. Image Generation

- Text-to-image conversion using Stable Diffusion
- Custom style specification
- Real-time generation progress tracking
- Automatic image saving and database storage

2. User Interface

- Dark theme with high contrast and readability
- Responsive layout with two-column design
- Progress indicators and status messages
- Intuitive sidebar with usage instructions

3. Image Gallery

- Grid layout display of generated images

- Expandable image sections
- Detailed information for each image:
    - Original prompt
    - Expected style
    - Creation timestamp
    - Style match rating
    - Evaluation notes

4. Search and Filter

- Search functionality for prompts and styles
- Real-time filtering of gallery content
- Clear search results display

5. Evaluation System

- 5-point rating scale for style matching
- Custom evaluation notes
- Performance summary by style
- Visual progress bars for ratings

🛠️ Technical Implementation

Dependencies

```
streamlit
diffusers
PIL
torch
sqlite3
datetime
```

Database Structure

```
CREATE TABLE prompts (
    id INTEGER PRIMARY KEY AUTOINCREMENT,
    prompt TEXT NOT NULL,
    image_path TEXT NOT NULL,
    expected_style TEXT,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    style_match_rating INTEGER,
    evaluation_notes TEXT
)
```

Key Components

1. **Model Loading**

- Uses Stable Diffusion v1.5
- Automatic GPU/CPU detection
- Cached model loading for performance

2. **Image Processing**
    - Automatic image saving with unique filenames
    - Safe filename generation from prompts
    - Timestamp-based versioning
3. **UI Components**
    - Custom CSS styling
    - Responsive layout
    - Progress indicators
    - Interactive elements
4. **Database Operations**
    - Automatic table creation
    - CRUD operations for prompts and images
    - Error handling and logging

Usage Guide

1. **Generating Images**
    - Enter your prompt in the text input
    - Specify the expected style
    - Click "Generate Image"
    - Wait for the generation process
    - View and evaluate the result
2. **Searching Images**
    - Use the search bar below the main content
    - Search by prompt or style
    - View filtered results in the gallery
3. **Evaluating Images**
    - Open any image in the gallery
    - Rate the style match (1-5)
    - Add evaluation notes
    - Save your evaluation

UI Customization

The application features a dark theme with the following color scheme:

- Background: #000000 (Black)

- Text: #ffffff (White)
- Secondary Text: #e0e0e0 (Light Gray)
- UI Elements: #1a1a1a (Dark Gray)
- Accent: #4CAF50 (Green)

Error Handling

The application includes comprehensive error handling for:

- Database operations
- Image generation
- File operations
- User input validation

Performance Monitoring

- Generation progress tracking
- Database operation logging
- Error logging
- Performance metrics for style matching

Future Enhancements

Potential features for future development:

1. Image comparison functionality
2. Prompt templates library
3. Advanced search filters
4. Export functionality
5. User preferences system

Notes

- The application requires a stable internet connection for model loading
- GPU is recommended for faster image generation
- Images are stored locally in the 'images' directory <<<<<<< HEAD
- Database is stored in 'prompts.db'

Example Generations

Here are some sample images generated by the app:

Prompt	Style	Example

Prompt	Style	Example
a panda riding a bicycle in space	cyberpunk	
a futuristic robot chef in a kitchen	cartoon	

Prompt	Style	Example
A magical castle over the clouds	Realistic	
a panda riding a bicycle in space	cartoon	

=======

- Database is stored in 'prompts.db'

e5d87027dc8f326ab450f49d35739bbf4fe6baec