Image Generation using stable diffusion & Comfy UI

A Project Report

submitted in partial fulfillment of the requirements

of

AICTE Internship on AI: Transformative Learning
with
TechSaksham – A joint CSR initiative of Microsoft & SAP

by

Rahul Tawar,

iamrahultawar@gmail.com

Under the Guidance of

Mr. Adharsh P, Mr. Jay Rathod

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to **Mr. Adharsh P** and **Mr. Jay Rathod**, my mentors during the internship, for their invaluable guidance and support. Their expertise, patience, and encouragement have been instrumental in helping me understand and implement image generation using Stable Diffusion and Comfy UI.

Throughout the one-month internship, they provided insightful feedback, technical knowledge, and constant motivation, which greatly enhanced my learning experience. Their mentorship has not only helped me develop this project but has also strengthened my problem-solving and technical skills.

I am truly grateful for their time and effort in mentoring me, and I look forward to applying the knowledge and skills gained under their guidance in future projects.

Rahul Tawar

ABSTRACT

The **Image generation using Stable Diffusion & ComfyUI project** is designed to generate high-quality AI images from textual descriptions using Stable Diffusion and ComfyUI. This project addresses the challenge of making AI-driven image generation more accessible and efficient for users without requiring complex setups.

- Problem Statement

  Generating detailed and realistic images from text requires powerful AI models and careful parameter tuning. Many existing solutions are difficult to use, limiting accessibility for non-technical users.

- Objectives
  - Develop a web-based AI image generator using Stable Diffusion.
  - Enable real-time image generation using WebSocket communication.
  - Optimize samplers and model parameters for clearer images.
  - Provide an intuitive user interface for seamless interaction.
- Methodology

  The project integrates ComfyUI for backend processing, a Flask API for handling requests, and WebSockets for real-time updates. The frontend is built using HTML, CSS, and JavaScript, allowing users to enter prompts and receive images dynamically.

- Key results
  - Successfully implemented real-time AI image generation.
  - Improved image clarity and detail using advanced samplers and settings.
  - Optimized response time with efficient model execution.
- Conclusion

  This project demonstrates an efficient AI-powered image generation system that is easy to use and highly responsive. Future improvements will focus on expanding customization options**,** enhancing performance, and improving image quality.

TABLE OF CONTENT

Abstract ..... **I**

Chapter 1. Introduction **1**

1.1 Problem Statement 1

1.2 Motivation 1

1.3 Objectives 2

1.4. Scope of the Project 3

Chapter 2. Literature Survey **4**

2.1 Review relevant literature or previous work in this domain 4

2.2 Mention any existing models, techniques, or methodologies 4

2.3 Highlight the gaps or limitations in existing solutions 5

Chapter 3. Proposed Methodology **6**

3.1 System Design 6

3.2 Requirements Specification 7

Chapter 4. Implementation and Results **8**

4.1 Snap shots of Result 8

4.2 Github Link for Code 9

Chapter 5. Discussion and Conclusion **10**

5.1 Future Work 10

5.2 Conclusion 10

References **11**

LIST OF FIGURES

Figure No.	Figure Caption	Page No.
Figure 1	System Flow Diagram	6
Figure 2	Project Web Page	8
Figure 3	User give prompt	8
Figure 4	Generated Image shown to User	9

LIST OF TABLES

Table. No.	Table Caption	Page No.
2.1	Existing Challenges and Solution	4

CHAPTER 1

Introduction

1.1 Problem Statement :

AI-generated images have transformed creative industries, making it easier to bring ideas to life. However, many existing tools come with steep learning curves, require powerful hardware, and demand technical expertise, making them inaccessible to many users. Even when users manage to generate images, they often struggle with blurry details**,** distorted faces, and a lack of control over fine details.

Significance of the Problem -

Not everyone has access to high-end GPUs or the technical know-how to tweak AI models for better results. A solution that simplifies image generation while maintaining high quality, speed, and ease of use can open up creative possibilities for artists, designers, and everyday users alike. Making AI-powered creativity more accessible means more people can bring their visions to life—without frustration or limitations.

1.2 Motivation :

Why This Project? –

The rise of AI-generated images has unlocked incredible creative possibilities, but many existing tools remain complex, slow, and difficult to use for the average person. This project was chosen to bridge the gap between AI technology and accessibility, allowing users to generate high-quality images without needing technical expertise or expensive hardware.

Potential Applications –

Digital Art & Design – Artists and designers can create unique visuals for branding, marketing, or personal projects.

Content Creation – Writers, bloggers, and social media creators can generate custom images to enhance storytelling**.**

Game Development – Developers can use AI-generated assets for prototyping or final game visuals.

Education & Research – AI-generated images can assist in visualization, helping students and researchers understand complex concepts.

Impact –

By making AI-driven image generation more user-friendly, efficient, and widely accessible, this project empowers creatives, educators, and professionals to unlock their imagination without technical barriers. The potential to save time, reduce costs, and enhance creativity makes this a valuable tool for individuals and industries alike.

1.3 Objective :

The goal of this project is to develop a user-friendly AI image generation system that delivers high-quality results efficiently. The key objectives include:

- **Ease of Use:** Create a simple interface accessible to both technical and non-technical users.
- **High-Quality Images:** Optimize generation to reduce distortions and enhance clarity.
- **Real-Time Interaction:** Allow prompt updates without page refresh.
- **Performance Optimization:** Improve speed and efficiency for smooth operation on various devices.
- **Scalability & Flexibility:** Ensure adaptability for future enhancements and customizations.

1.4 Scope of the Project :

Scope -

This project focuses on real-time AI image generation using a user-friendly web interface. Key functionalities include:

- **Text-to-Image Generation:** Users can generate images based on prompts.
- **Real-Time Updates:** New images can be generated without refreshing the page.
- **Customization:** Users can modify prompts to refine image outputs.
- **Optimized Performance:** Ensures smooth operation with minimal processing time.

Limitations -

While the system aims to provide high-quality images, certain constraints exist:

- **Image Accuracy:** Generated images may occasionally have distortions, especially in complex details.
- **Processing Time:** High-resolution images may take longer to generate.
- **Hardware Dependency:** Performance may vary based on system specifications.
- **Limited Fine-Tuning:** Users have some control, but advanced customization requires manual adjustments.

CHAPTER 2

Literature Survey

2.1 Review relevant literature or previous work in this domain.

AI-driven image generation has evolved significantly, leveraging deep learning techniques such as Generative Adversarial Networks (GANs) and Diffusion Models.

2.2 Mention any existing models, techniques, or methodologies related to the problem.

- **DeepDream (2015) – Google**

  Early AI image transformation tool focused on enhancing image patterns. Primarily used for artistic effects rather than structured image generation.

- **StyleGAN (2018-2019) – NVIDIA**

  Introduced high-resolution image synthesis with fine-grained control. Specialized in face generation, but required extensive training data.

- **DALL·E (2021) – OpenAI**

  Utilized transformer-based architectures for text-to-image generation. Generated diverse images but struggled with fine details like hands and faces

- **MidJourney (2022-Present)**

  Focused on artistic and stylized image generation. Lacks granular user control over outputs.

- **Stable Diffusion (2022-Present)**

  Open-source diffusion model capable of high-quality text-to-image generation. Requires high computational power and detailed prompt engineering.

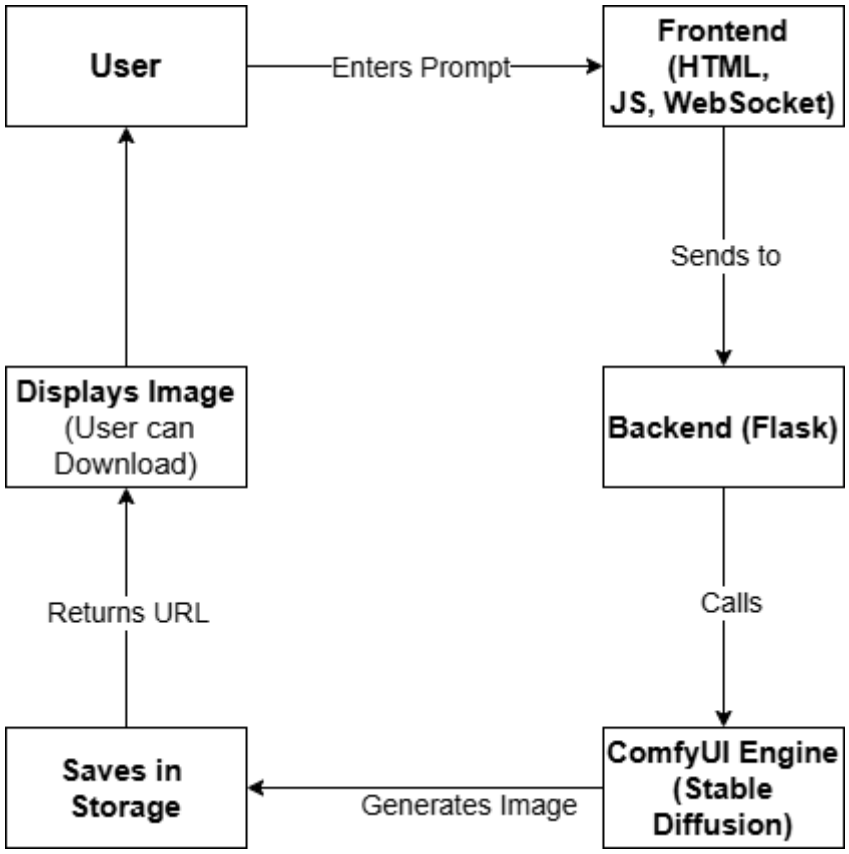
2.3 Highlight the gaps or limitations in existing solutions and how your project will address them.

Existing Challenges	Our Solution
High computational requirements restrict accessibility.	Optimize inference for faster and lighter processing.
Users need technical expertise to generate high-quality images.	Provide a simplified UI with real-time control.
Distorted faces, hands, and fine details	Use optimized samplers and enhanced diffusion techniques.
Limited real-time interaction—most tools require full page refresh.	Enable dynamic prompt updates without refresh.
Lack of precise user control over outputs.	Offer adjustable parameters for better customization.

CHAPTER 3

Proposed Methodology

3.1 System Design



The architecture consists of the following components:

1. **User Interface (Frontend)**
   - Users input a text prompt on the webpage.
   - The UI is built with HTML, CSS, and JavaScript.
   - Uses WebSockets to communicate with the backend for real-time updates.
2. **Backend Server**
   - Built using Flask.
   - Processes the user input and sends it to the AI model.
   - Manages WebSocket connections for live status updates.
3. **AI Model (ComfyUI + Stable Diffusion)**
   - Stable Diffusion generates high-quality images.

- The model runs on CUDA-enabled GPUs for faster inference.
- Uses optimized samplers (DPM++ 2M Karras) for detailed image generation.

4. **Image Processing & Storage**
   - The generated image is temporarily stored on the server.
   - A direct URL is provided for frontend display and downloads.
5. **User Output**
   - The image is displayed on the webpage.
   - Users can download the generated image.

3.2 Requirement Specification

Mention the tools and technologies required to implement the solution.

3.2.1 Hardware Requirements :

- **CPU –** Modern multi-core processor
- **RAM –** 8GB or more
- **GPU –** Nvidia GPU with at least 4GB VRAM (for GPU acceleration)

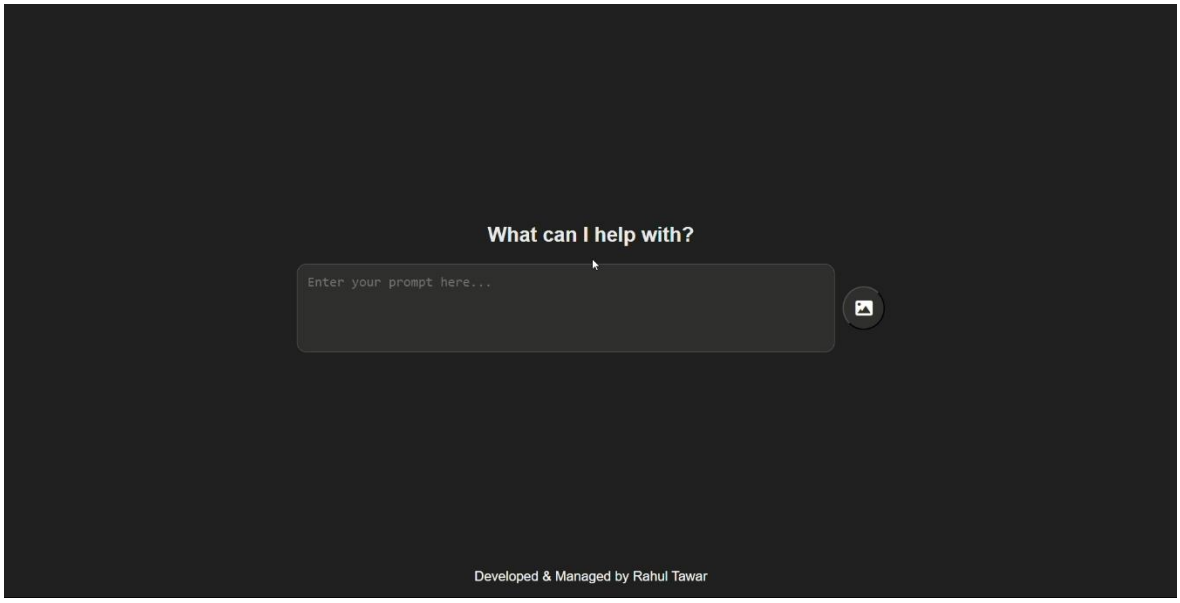
3.2.2 Software Requirements :

- Python 3.7 +
- Flask
- Requests
- WebSocket – client
- UUID
- Pillow
- ComfyUI
- Stable Diffusion

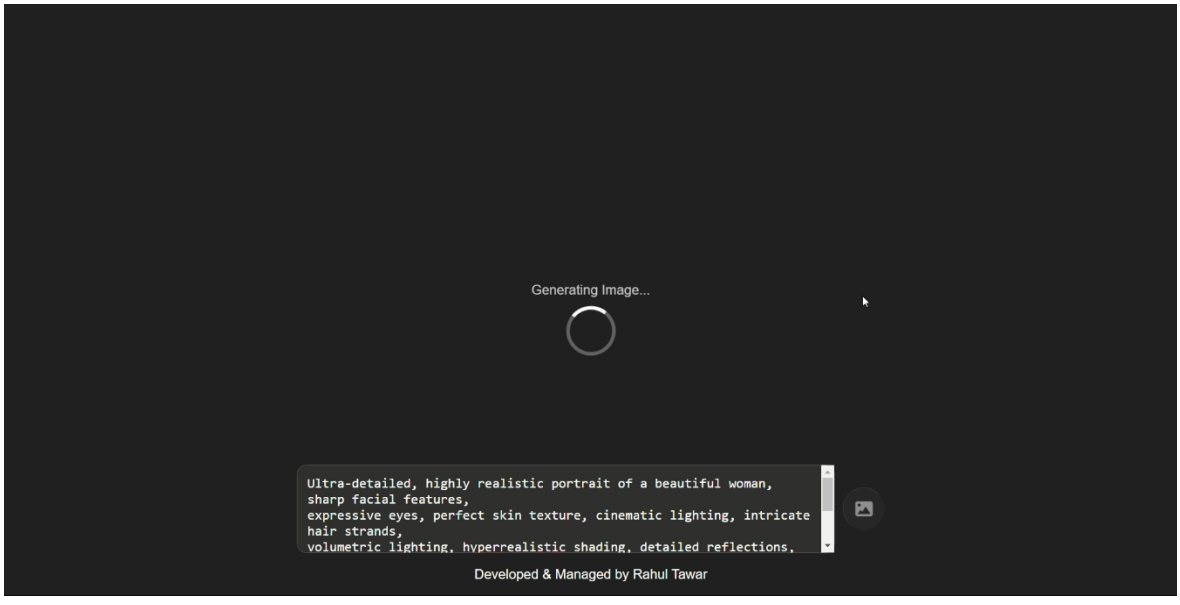
CHAPTER 4

Implementation and Result

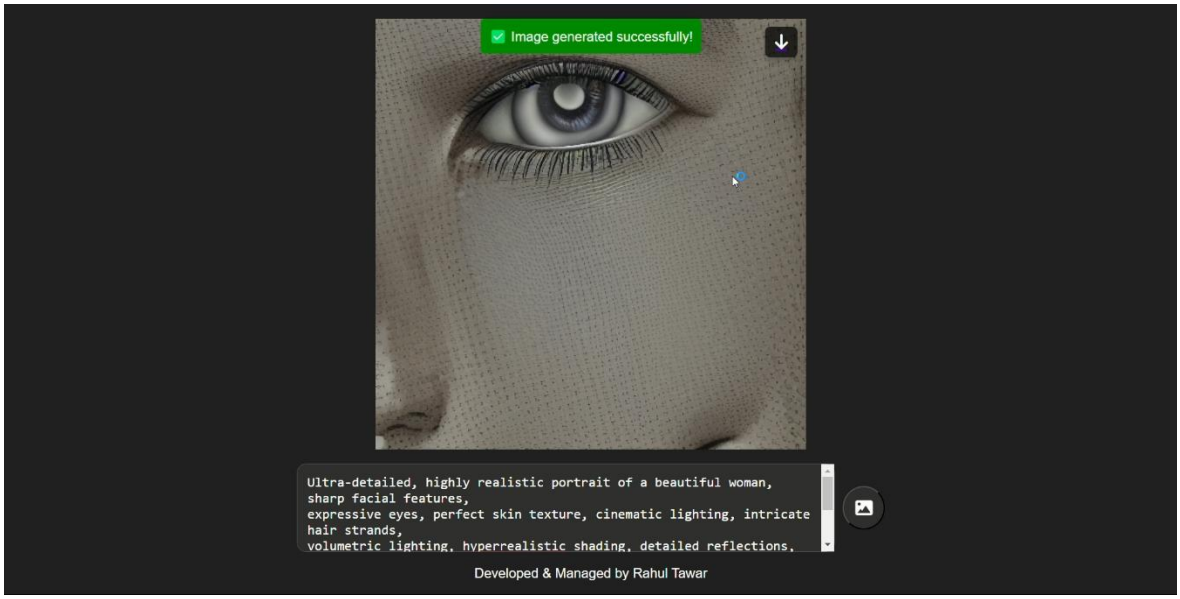
4.1 Snap Shots of Result :



Description – When user open project then this webpage will be shown to user



Description – User give positive prompt and request sent to backend. Loading will happen till image is not generated and shown to user.



Description – Based on user prompt, Image is generated and shown back to user with download functionality.

4.2GitHub Link for Code :

Github Link - https://github.com/Rahultawar/AI-Image-Generator

CHAPTER 5

Discussion and Conclusion

5.1 Future Work :

- **Enhancing Image Quality**
- **Reducing Processing Time**
- **Improved User Control**
- **Expanding Model Capabilities**
- **Better Web App & UX**
- **Scalability & Deployment**

5.2 Conclusion :

This project simplifies AI-powered image generation, making it faster, clearer, and more user-friendly. It enhances image quality, processing speed, and user control, benefiting artists, designers, and content creators.

By improving accessibility and efficiency, this project helps users harness AI for creative work with ease. Future improvements will focus on refining performance and usability to make AI-generated imagery even more practical.

REFERENCES

[1] Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.

[2] Ramesh, A., Pavlov, M., Goh, G., et al. (2021). *Zero-Shot Text-to-Image Generation*. OpenAI.

[3] Ho, J., Jain, A., & Abbeel, P. (2020). *Denoising Diffusion Probabilistic Models*. arXiv preprint arXiv:2006.11239.

[4] Kingma, D. P., & Welling, M. (2013). *Auto-Encoding Variational Bayes*. arXiv preprint arXiv:1312.6114.

[5] OpenAI. (2022). *DALL·E 2: High-resolution Image Synthesis with Deep Learning*.

[6] OpenAI Documentation - https://openai.com/research/

[7] ComfyUI GitHub - https://github.com/comfyanonymous/ComfyUI