



# Automated Text-to-Image Generator

SYEDA WARISHAH BUKHARI (1980151)  
ADEEN ASAD (1980104)

Supervised by: MR. SYED MAHDI ABBAS SHAH

In partial fulfillment of requirement for the degree  
Bachelor of Science (Software Engineering)  
Shaheed Zulfiqar Ali Bhutto Institute of Science and Technology  
Islamabad, Pakistan

Fall 2022

# Revision History

Compiled By	Checked By	Date	Reason for Change	Version
Warishah & Adeen	Mr. Zain Arshad	1st Oct 2022	Initial Version	1.0
Warishah & Adeen	Mr. Zain Arshad	5th Oct 2022	Introduction Revised	1.2
Warishah & Adeen	Mr. Syed Mahdi Abbas Shah	10th Oct 2023	Revised 5.1 & 8	2.0
Warishah & Adeen	Mr. Syed Mahdi Abbas Shah	17th Oct 2023	Revised 2,7 & References	2.5

# Automated Text-to-Image Generator

*By*

*Syeda Warishah Bukhari and Adeen Asad*

## CERTIFICATE

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE  
REQUIREMENTS FOR THE DEGREE OF BACHELOR OF SCIENCE IN  
SOFTWARE ENGINEERING (BSSE).

It is to certify that the above students thesis has been completed to my satisfaction and, to my belief, its standard is appropriate for submission for Evaluation. I have also conducted plagiarism test of this thesis using HEC prescribed software and found similarity index "write plagiarism percentage" at that is within the permissible limit set by the HEC for the BSSE degree thesis. I have also found the thesis in a format recognized by the SZABIST for the BSSE thesis.

---

(Supervisor)

Mr. Syed Mahdi Abbas Shah

---

(Program Manager BSSE)

Mr. Babar Jehangir

Date: \_\_\_\_\_

Department of Computer Science, Shaheed Zulfikar Ali Bhutto Institute of  
Science and Technology, Islamabad Campus

Fall 2022

# Project Overview

As a software engineer, keeping in mind the scope of Artificial Intelligence (AI) technology is increasing rapidly, we have chosen a system for our project that caters to the needs of those who love nature photography, creative art, etc. Our proposed application “Automated Text-to-Image Generator” is a new AI-based system that can create realistic images and creative art from a text description in natural language, that shall provide users ease to create realistic images or creative artworks from the written description. Automated Text-to-Image Generator will be an online website for generating illustrations from written descriptions. The program uses artificial intelligence (AI) to create realistic artwork in high-quality images from a text description written in natural language with an attractive interface.

Automated Text-to-Image Generator is a method used for generating images related to given textual descriptions. It has a significant influence on many research areas as well as a diverse set of applications (e.g., photo-searching, photo-editing, art generation, computer-aided design, image reconstruction, captioning, and portrait drawing). The most challenging task is to consistently produce realistic images according to given conditions. Existing algorithms for Automated Text-to-Image Generator create pictures that do not properly match the text. We considered this issue in our study and built a ML and deep learning-based architecture for semantically consistent image generation: Natural Language Processing (NLP) and diffusion Method. It successfully bridges the advancements in text and picture modeling, converting visual notions from words to pixels. The proposed model was trained on the OpenAi dataset, and its performance was evaluated using an inception score. The experimental results demonstrate that our model is capable of generating more realistic photos from given captions. In the future, we aim to train the proposed model on multiple datasets.

# Dedication

Firstly, we dedicate our project to the creator Allah Almighty and dedicate to whom the world owes its existence Muhammad (Peace Be Upon Him) and dedicate this to our beloved parents, our extremely dedicated and generous teachers and supportive friends, their prayers always pave the way to success for us.

# Acknowledgement

The requirements for the degree of Bachelor of Software Engineering.

---

Syeda Warishah Bukhari (1980151)

---

Adeen Asad (1980104)

# Contents

<b>Revision History</b>	<b>2</b>
<b>Project Overview</b>	<b>ii</b>
<b>Dedication</b>	<b>iii</b>
<b>Acknowledgements</b>	
<b>List of Figures</b>	<b>iii</b>
<b>List of Tables</b>	<b>iv</b>
<b>1 Introduction</b>	<b>v</b>
1.1 Product Purpose . . . . .	1
1.2 Product Scope . . . . .	1
1.2.1 Existing System Description . . . . .	2
1.2.2 Future System Usage Analysis . . . . .	8
1.3 Objectives . . . . .	9
1.3.1 Users/Stakeholders . . . . .	9
1.4 Problem Statement/Limitations . . . . .	9
1.5 Proposed Solution . . . . .	10
1.6 Intended Market of Product . . . . .	11
1.7 Intended Users of Product . . . . .	11
1.8 Software Process Model . . . . .	12
1.8.1 Process Model Introduction . . . . .	12
1.8.2 Justification . . . . .	13
1.8.3 Steps of Process Model . . . . .	13
<b>2 SOFTWARE REQUIREMENTS SPECIFICATION</b>	<b>15</b>
2.1 Introduction . . . . .	15
2.1.1 Document Scope . . . . .	15
2.1.2 Audience . . . . .	16
2.2 Functional Requirements . . . . .	16
2.2.1 Functional Requirements . . . . .	16
2.3 Non-Functional Requirements . . . . .	17
2.3.1 Software Quality Attributes . . . . .	18
2.3.2 Other Non-Functional Requirements (e.g.Legal, Platform, etc.) . . . . .	18
2.4 Requirement Gathering Techniques Used . . . . .	18
2.4.1 Analyzing Existing Literature . . . . .	18
2.4.2 Brainstorming . . . . .	19
2.4.3 Questionnaire . . . . .	19
2.5 Time Frame . . . . .	19
<b>3 SOFTWARE PROJECT PLAN</b>	<b>20</b>
3.1 Deliverables of the Project . . . . .	20
3.1.1 SRS Document: . . . . .	20
3.1.2 Design Document: . . . . .	20

3.1.3	Database: . . . . .	20
3.1.4	Website Frontend: . . . . .	20
3.1.5	Database Connection: . . . . .	20
3.1.6	Website Backend: . . . . .	20
3.1.7	Image Generator: . . . . .	21
3.1.8	User Manual: . . . . .	21
3.2	Software Project Management Plan . . . . .	21
3.2.1	Project Planning . . . . .	21
3.2.2	Project Scheduling . . . . .	24
3.3	Managerial Process . . . . .	27
3.3.1	Management Objectives and Priorities . . . . .	27
3.3.2	Assumptions and Constraints . . . . .	27
3.4	Project Risk Management . . . . .	28
3.4.1	Risk Management Plan . . . . .	28
3.4.2	Risk Management Activities . . . . .	29
<b>4</b>	<b>FUNCTIONAL ANALYSIS AND MODELING</b>	<b>31</b>
4.1	Use Case Modeling . . . . .	31
4.1.1	User Stories . . . . .	31
4.1.2	Individual Actor Use Cases . . . . .	32
4.1.3	Expanded Use Cases . . . . .	33
4.1.4	Use Case Diagrams . . . . .	34
4.2	Functional Modeling . . . . .	37
4.2.1	Entity Relationship Diagram . . . . .	37
4.2.2	Data Flow Diagram . . . . .	37
<b>5</b>	<b>SYSTEM DESIGN</b>	<b>41</b>
5.1	Structure Diagrams . . . . .	41
5.1.1	Class Diagram . . . . .	41
5.1.2	Deployment Diagrams . . . . .	41
5.2	Behavioral Diagrams . . . . .	42
5.2.1	Activity Diagrams . . . . .	43
5.2.2	Communication Diagrams . . . . .	44
5.2.3	Sequence Diagrams . . . . .	44
5.2.4	View Database . . . . .	47
<b>6</b>	<b>SYSTEM INTERFACE AND PHYSICAL DESIGN</b>	<b>50</b>
6.0.1	System User Interfaces . . . . .	50
6.0.2	Main Module Interface . . . . .	55
6.0.3	Systems Working Interface . . . . .	57
6.0.4	Speech To Text Interface . . . . .	59



# List of Figures

1.1	Workflow of Agile SDLC Model . . . . .	12
3.1	Gantt Chart . . . . .	24
3.2	Work Breakdown Structure . . . . .	25
3.3	Critical path . . . . .	26
4.1	Full System Use Case Diagram . . . . .	36
4.2	Entity Relationship Diagram . . . . .	37
4.3	Data Flow Diagram Level 0 . . . . .	38
4.4	Data Flow Diagram Level 1 . . . . .	39
4.5	Data Flow Diagram Level 2 . . . . .	40
5.1	Class Diagram . . . . .	41
5.2	Deployment Diagram . . . . .	42
5.3	Activity Diagram . . . . .	43
5.4	Communication Diagram . . . . .	44
5.5	Login Sequence Diagram . . . . .	45
5.6	Audio to Text Sequence Diagram . . . . .	46
5.7	MongoDB Database . . . . .	47
5.8	User Database . . . . .	47
5.9	User Database . . . . .	48
5.10	Image Collection Database . . . . .	48
5.11	Image Collection Database . . . . .	49
6.1	Home Page . . . . .	50
6.2	Home Page . . . . .	50
6.3	About Us . . . . .	51
6.4	About Us . . . . .	51
6.5	Sign-up . . . . .	52
6.6	Sign-up . . . . .	52
6.7	Sign-up . . . . .	53
6.8	Sign-in . . . . .	53
6.9	Sign-in . . . . .	54
6.10	Generator . . . . .	55
6.11	Generator . . . . .	55
6.12	My Collection . . . . .	56
6.13	My Collection . . . . .	56
6.14	Generator . . . . .	57
6.15	Generator . . . . .	57
6.16	My Collection . . . . .	58
6.17	Generator . . . . .	58
6.18	My Collection . . . . .	59
6.19	Mic Testing . . . . .	59
6.20	Save in Collection . . . . .	60
6.21	My Collection . . . . .	60

# List of Tables

1.1	Abbreviations . . . . .	2
1.2	Applications Comparison . . . . .	8
2.1	Functional Requirements . . . . .	17
2.2	Time Frame . . . . .	19
3.1	Milestones Plan . . . . .	21
3.2	Documentation Plan . . . . .	22
3.3	Documentation Plan . . . . .	22
3.4	Quality Plan . . . . .	23
3.5	Documentation Plan . . . . .	29
4.1	Register User . . . . .	32
4.2	Login User . . . . .	32
4.3	Image Generator . . . . .	32
4.4	My Collection . . . . .	33
4.5	Expanded 01 Register User . . . . .	33
4.6	Expanded 02 Login User . . . . .	33
4.7	Expanded 03 Image Generator . . . . .	34
4.8	Expanded 04 My Collection . . . . .	34

# Chapter 1

## Introduction

When people listen to or read a narrative, they quickly create pictures in their minds to visualize the content. Many cognitive functions, such as memorization, reasoning ability, and thinking, rely on visual mental imaging or “seeing with the mind’s eye”. Developing a technology that recognizes the connection between vision and words and can produce pictures that represent the meaning of written descriptions is a big step toward user intellectual ability. Image-processing techniques and applications of computer vision (CV) have grown immensely in recent years from advances made possible by artificial intelligence and deep learning success. One of these growing fields is text-to-image generation as well as, Multi-modal artificial intelligence which is an emerging field of Artificial Intelligence (AI) that deals with enabling AI to process and relate multi-modal data (an emerging field that enables the AI/ML model to learn from and process multiple modes and types of data (i.e image, text, audio, video) rather than just single type of data.

Automated Text to Image Generator will be an online website for generating illustrations from written descriptions. The program uses artificial intelligence (AI) to create realistic artwork in high-quality images from a text description written in natural language with an attractive interface. It will allow people to type in a descriptive prompt of up to 250 words, for example, ”an astronaut riding a horse in an impressionist style” and AI magic will create the image. As well as, there are several ways to use Automated Text to Image Generator. The first thing you’ll see once you have created an account on our system is that you can type in. This is the prompt field. You’ll type in a description of the image which they want to create in up to 250 words. It will take some practice. Click ‘generate’ and Automated Text to Image Generator will create four image options based on your prompt. If you get an error message, just try again after a moment. Users can complete signing up and creating an account on our website. You will be asked to enter your email address and a security code and to create an eight-digit password then the user will receive an email with a link that you’ll have to click through to verify and you’ll also receive a text code by SMS that you’ll have to enter to confirm your identity. Click ‘continue’ to accept the terms and conditions, and you’ll be ready to get started using the Automated Text to Image Generator.

Automated Text to Image Generator will launch as a paid website. So, it will start using a credit-based model. Users will get 50 free credits that they can put towards generating an image (image generations return four 1024 X 1024-pixel images for the cost of one credit). After that, the user will get 15 free Automated Text to Image Generator credits each month. To get more, users will have to buy them at a price of Not Decided Yet! credits. We will also limit the ability of Automated Text to Image Generator to create violent, hateful, or adult images. By removing the most explicit content from the training data, we minimized the Automated Text to Image Generator’s exposure to these concepts. We will also use advanced techniques to prevent photo-realistic generations of real individuals’ faces, including those of public figures. Automated Text to Image Generator will learn the relationship between images and the text used to describe them.

It uses a process called “diffusion,” which starts with a pattern of random dots and gradually alters that pattern towards an image when it recognizes specific aspects of that image.

Automated Text to Image Generator provides user-friendly and attractive online making realistic images from a text description through a responsive website for laptops, PC, and mobile device users in a responsive layout.

## 1.1 Product Purpose

The purpose of this application is to provide an interface where the user can generate digital images from natural language descriptions and can expand images beyond what’s in the original canvas, creating expansive new compositions as well as, making realistic edits to existing images from a natural language caption called ”prompts”. Automated Text-to-Image Generator uses a technique called “diffusion”, which chooses a pattern of randomly placed dots and modifies the pattern to resemble a picture as it picks up on details of that image.

It builds on that system by offering more realistic, accurate image generation with quadruple resolution. Automated Text-to-Image Generator can consistently create different types and styles of images, even when the input (your description) changes. It can be generalized to many types of image creation by a text description.

With Automated Text-to-Image Generator, you can create amazing art within a few seconds. You can create images from your text description. And can write not only unique content with the help of an AI but also create unique new images based on your suggestions. These images are so unique that you won’t even need stock images anymore.

As we know, The future holds a wide array of endless opportunities for creatives and their clients alike. Through programs such as Automated Text-to-Image Generator, the limits of our imagination have been once again extended to unimaginable new heights.

## 1.2 Product Scope

The scope of our proposed system is to provide people ease to maintain a convenient and easy-to-use application for users, trying to create amazing images by a text description. Our system will overcome the limitations of purchasing very expensive artistic art and also overcome the limitations of the DALL.E 2 application. Automatic Text-to-Image Generator will provide the online editor to create interesting and your favorite art images at a reasonable price.

Automated Text-to-Image Generator is a new AI system that automatically creates and edits images simply by following the natural language instructions you give it. Automated Text-to-Image Generator is a text-based AI image generator based on machine learning. It’s a generative tool, which means users can generate art from scratch by using text prompts.

First, We can now express things that were hard to express before. As we explained earlier, we can now generate images that were difficult to express even for humans with only text as an input. Automated Text-to-Image Generator has made it possible for AI

to enter the realm of “creation”. Second, it can be a way to test whether AI can produce more than human data or just use the data that is provided. We can use this model to determine whether AI can do more than humans have taught AI could do. Finally, we will be able to see how much AI can understand the human world. We’ll see if the human world that Automated Text-to-Image Generator sees is in the realm of our imagination, or if it’s out of it.

It’s also important to see if this model will be used for good or bad, ethically or unethically. And looking at it will inevitably be a valuable tool for social consensus on safe and beneficial AI development. Open AI claims that the Automated Text-to-Image Generator model will help develop safe and beneficial AI.

Automated Text-to-Image Generator products hope that this tool could help them surpass human creativity. Automated Text-to-Image Generator comprises two OpenAI technologies; GPT-3 and CLIP. GPT-3 is a powerful engine that has the ability to understand the human text and respond meaningfully. CLIP is an advanced computer vision system that understands imagery and can describe the contents. In Automated Text-to-Image Generator has reversed the CLIP process by having GPT-3 supply a description of an image, then CLIP interprets the text and creates a visual image. The results are truly amazing.

There is a table of all conventions are abbreviations used in the document with their description as shown below:

Table 1.1: Abbreviations

Serial #	Abbreviation	Definition
1	AI	Artificial Intelligence
2	ML	Machine Learning
3	CV	Computer Vision
4	NLP	Natural Language Processing
5	CLIP	Contrastive Language–Image Pre-training
6	GPT-3	Generative Pre-trained Transformer 3

### 1.2.1 Existing System Description

The existing systems that are related to the proposed system ”Automated Text-to-Image Generator” with their some features and limitations. We have selected 3 applications. So, let’s have a look at the following applications as stated here:

**DALL·E 2** [1] DALL-E (stylized as DALL·E) and DALL-E 2 are machine learning models developed by OpenAI to generate digital images from natural language descriptions. DALL-E was revealed by OpenAI in a blog post in January 2021 and uses a version of GPT-3 modified to generate images. In April 2022, OpenAI announced DALL-E 2, a successor designed to generate more realistic images at higher resolutions that ”can combine concepts, attributes, and styles”.

DALL·E 2 can expand images beyond what’s in the original canvas, creating expansive

new compositions. DALL·E 2 can make realistic edits to existing images from a natural language caption. It can add and remove elements while taking shadows, reflections, and textures into account. of it inspired by the original.

The four features of DALL·E 2 are listed below:

- More Clarity Between Visuals and Texts
- More Realistic and High-Resolution Images
- Editing and Retouching Made Simpler
- Produce Multiple Iterations of an Image

The limitations of the DALL·E 2 are listed below:

- If the data labeling is incorrect, it can produce false results, just like someone who learned the wrong word.
- When it receives a text that it has not learned before, it will try to produce similar results to what it saw during training, but the results may be too different.
- Prompts it can't work out, lack of common sense.

**Jasper** [2] Dave Rogenmoser is co-founder and CEO of Jasper, an Austin-based company using artificial intelligence to create online content such as advertisements and blog posts. Jasper AI is a copy-writing tool that uses the power of artificial intelligence and machine learning to automatically produce persuasive, enticing copy for your blog posts, landing pages, social media, ads, marketing emails, and much more allowing you to write 5x more content in a fraction of the time. Jasper Art allows you to generate images from text. It uses artificial intelligence to interpret the text and create an image that represents it.

The main features of Jasper are listed below:

- Jasper uses AI to generate images from text prompts.
- Creates unique images in a variety of styles.
- Has the option to draw photo-realistic images.
- Perfect for making marketing or blog images.
- Works in your web browser (no software download required).
- Can potentially replace using some online design tools.
- Allows unlimited use for a very small monthly fee.

The limitations of the Jasper are listed below:

- Sometimes the system does not give you the correct result.
- Sometimes the system gets too slow that your result takes too much time to be displayed.
- Jasper Art subscription is not free. It's too much expensive if you're a single user.
- Jasper art introduces a system with two modes one is a simple mode and another is the premium mode. Simple mode has only basic features that they provide to users and in premium mode, the subscription charges are too much high that every person cannot afford them.

- Images are only created in a square 1:1 ratio. It would be nice if it created images in a widescreen format too.
- You need to watch Jasper's writing like a hawk: From the input to output, you need to guide it along every step as it optimizes for control.
- You need to practice with the tool a bit: Jasper will NOT give you exactly what you want.
- No carry forward for unused credits: It should be like the Audible model, where you can save up the credit later on.
- Limited rewriting functions: The summary and paraphrasing function can only handle a limited word count (30 words) and is not very good compared with other rewriting tools.
- Jasper has only trained with 10per of web content from up to 2014. This means you should not rely on Jasper to be accurate with its facts. You should always check its fact.

**Nightcafe** [3] NightCafe NFT Creator was founded by Angus Russell in November 2019. It is an AI NFT Generator. It creates digital artworks to sell as NFTs using the power of Artificial Intelligence. It uses a neural network to interpret the text and generate an image that represents it. It is also available as a mobile application in the app store (for androids).

The main features of Nightcafe are listed below:

- Create creative images from a text
- Image editing
- Blockchain support
- Content import/export
- Bulk creation of artwork
- Unique tools that can blend and mix multiple styles to create an NFT art
- It can manage NFT artwork on multiple devices like laptops, tablets, phones, and desktops

The limitations of the Nightcafe are listed below:

- You have to buy or earn credits to keep creating art which would be too much costly
- Sometimes it gives the wrong result (Software Failure)

**Craiyon** [4] Craiyon, formerly Dall-e mini, is a text-to-image generator created by Boris Dayma originally for a coding competition. Craiyon, is an alternative version of DALL-E mini, by the same developers. It is an AI model that can draw images from any text prompt that you enter.

The main features of Craiyon are listed below:

- Craiyon is free and there are no limits.
- Add art style descriptions.

- Add locations or actions.
- Add composition descriptions.
- Add mood descriptions.
- Combine multiple objects.
- Ask for screenshots.

The limitations of the Craiyon are listed below:

- Faces and people in general are not generated properly.
- Animals are usually unrealistic.
- It is hard to predict where the model excels or falls short. . . Good prompt engineering will lead to the best results.
- The model has only been trained with English descriptions and will not perform as well in other languages.

**Dream by Wombo** [5] Wombo Dream is a text-to-image AI art generator that produces original “artworks” depending on a text prompt. Wombo, a Canadian firm that first gained notoriety for its AI-enabled lipsyncing video app of the same name, has released Dream by Wombo.

The main features of Dream by Wombo are listed below:

- It has 42 art-style options like flora, meme, realistic, HDR, etc.
- Putting input images is optional, as your input image will be used as a reference for the final output artwork.
- There are six prompt text options: sunset clips, never-ending flowers, groovy time, etc.
- Apart from the given prompt options, you can use any of your customizable prompts to create unique art out of that prompt.
- On the home page, there is a sun and moon option to change the background color to black and white, respectively, to get a prominent vision of the page.
- It never repeats its design.

The limitations of the Dream by Wombo are listed below:

- It would abandon the creation process halfway through. Fortunately, this only happened a few times and restarting the app seems to stop it from happening again for awhile.
- It wouldn’t be able to properly process a face and keep it from looking odd though.

**starryai** [6] starryai is an AI art generator app. You simply enter a text prompt and our AI transforms your words into works of art. AI Art generation is usually a laborious process that requires technical expertise, we make that process simple and intuitive. starryai is available for free on iOS and Android.

The main features of starryai are listed below:



- User and beginner friendly.
- Different AI algorithms and Models to choose.
- Community features.
- The App is available for iOS and Android.

The limitations of the starryai are listed below:

- It can take more than 5 minutes to generate an image.

**Deep Dream Generator [7]** DeepDream is a computer vision program created by Google Engineer Alexander Mordvintsev. The Deep Dream Generator is a computer vision platform that allows users to input photos into the program and transform them through an artificial intelligence algorithm.

The main features of Deep Dream Generator are listed below:

- The ability to generate visually striking and unique images by layering and manipulating the features learned by a pre-trained neural network.
- The ability to generate new images from scratch by providing the generator with random noise as input.
- The ability to apply the DeepDream algorithm to any pre-trained neural network, allowing for a wide range of possible image generations.

The limitations of the Deep Dream Generator are listed below:

- The generated images can appear abstract and difficult to interpret, making it hard to use in a practical application.
- The algorithm can be computationally intensive, requiring significant processing power and time to generate a single image.
- The generated images are often highly stylized and do not always resemble real-world objects or scenes.
- There is no control over what features the network is highlighting and how they are being manipulated.
- It is based on a pre-trained model, and if it is not trained on the specific dataset that is needed, the output may not be as expected.

**Deep AI [8]** Kevin Baragona is the founder of Deep AI. This is an AI Image Generator. It creates an image from scratch from a text description. Text-to-image uses AI to understand your words and convert them to a unique image each time.

The main features of Deep AI are listed below:

- The ability to generate realistic images from text descriptions.
- The ability to handle a wide range of text inputs, including captions, headlines, and paragraphs.
- The ability to produce a diverse range of images, including objects, scenes, and people.

The limitations of the Deep AI are listed below:

- The quality of the generated images may vary depending on the input text, and the images may not always be of high quality.
- The algorithm may not always be able to accurately interpret or depict the meaning of the text input, resulting in images that do not match the intended description.
- The algorithm may struggle with more complex or abstract concepts, such as emotions or abstract ideas.
- The algorithm is based on deep learning models and requires large amounts of computational power and data to train.
- The generated images may not be completely original as it is based on pre-trained model and the output may be similar to the training images.

**Fotor** [9] Fotor owns by Chengdu Hengtu Technology Co., Ltd. Fotor's AI art generator uses AI image generation technology to turn photos into art in one click, you can discover your creative side and easily convert your ordinary photos into amazing oil paintings, watercolors, cartoons, sketches, and more.

The main features of are listed below:

- 10 free image generations per day.
- 2 image conversion models: text-to-image and image-to-image.
- Fast image generation mode.
- Ability to produce great images from text.
- 9 different conversion styles to choose from.

The limitations of the are listed below:

- It is not reliable.
- It's very costly.

Table 1.2: Applications Comparison

Features	Applications									
	DALL.E 2 [1]	Jasper [2]	NightCafe [3]	Cariyon [4]	Dream by Wombo [5]	Starryai [6]	Deep Dream [7]	Deep AI [8]	Fotor [9]	Proposed System (ATIG)
Natural Language Processing (NLP) capability	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
High-quality image generation	✓	✗	✗	✓	✓	✗	✗	✗	✗	✓
Produce multiple iterations of an image	✓	✗	✗	✓	✓	✗	✗	✗	✗	✓
Flexibility (to handle a wide range of text inputs)	✗	✗	✓	✓	✓	✗	✗	✗	✓	✓
Description limit is up-to 250 characters	✓	✗	✗	✓	✗	✓	✗	✗	✗	✓
Scalability (generate images quickly and efficiently)	✗	✗	✗	✗	✗	✗	✗	✓	✓	✓
Originality (generate completely original images)	✓	✗	✗	✓	✗	✗	✗	✗	✓	✓
Transparency (provide an explanation of generated image)	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓
Error handling (handle errors and provide meaningful feedback to the user)	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓

## 1.2.2 Future System Usage Analysis

As this is a Web-based project, the end-user must require a laptop/PC to use the application and also should have an internet connection in order to get services from the platform. It is now becoming a basic need for people to have a PC/laptop in order to perform business activities searching jobs, online study, and use software tools for software engineers. On other hand, due to the evolution in the internet service provider industry like 3G, 4G, and 5G services, it is very easy for anyone to get connected to the internet and get services online. It is already been observed that after the occurrence of COVID-19, the speed of digitization of businesses is at on peak. These factors can help us to understand and relate to this application usage in the future.

So, in terms of our proposed application "Automated Text-to-Image Generator", Images that once required a team of artists to produce can now be created in a matter of seconds, with only one artist needing to correct them. Additionally, since they are pro-

duced on highly powerful servers, photographs have a cost, especially when many users come to create their own. It will be extremely expensive for new players to enter the market, which can deter some businesses.

As a result, we'll be anxiously awaiting the development of AI photos. In the past year, the market has changed so drastically that it seems fresh developments are being made daily. But with AI-based picture alteration now showing up on some of our smartphones, a lot could change in the next year or two.

## 1.3 Objectives

To provide the best experience for those who love creativity, and photography and to make artistic art possible through high-quality image generation from a text description is our aim. From the business point of view, our major goal is to earn money by learning or exploring something new through the development of AI-based systems particularly Web-based software development.

To provide users to make digital or artistic art without having any expert skills in creating any art at reasonable pricing with the ease of an easy-to-use application is or another important goal. In order to achieve the goals, we have made some specific objectives are:

- To launched as a Web-based system and run on a windows-based platform.
- To be a paid website.
- To develop Web-based application software this is suitable for our proposed AI-based system.
- To be the application have a database server supporting a lot of images generated by the text description.

### 1.3.1 Users/Stakeholders

The stakeholders of our proposed software system are:

- End-users: The end-users are the people that will ultimately use our system.
- Customers: The customers are the people who will pay to use system tools.
- System Admin: The system administrator stakeholder will be responsible for administering and managing or maintaining the website.

## 1.4 Problem Statement/Limitations

In terms of creativity, Automated Text-to-Image Generator is an AI image generator that can enable people to express themselves visually in ways they previously couldn't. This field has gained a lot of attention in recent years due to its potential applications in various areas such as computer vision, art and design, and gaming. Automated Text-to-Image Generator is an AI that generates images from a text description but it can be used in other creative fields. Consider storyboarding a short film with a lot of visual effects. Your out-of-this-world description of ideas will allow Automated Text to Image

Generator to spark your imagination. Writers, concept artists, and fabricators can come up with new creature ideas or worlds so they can spend more time developing their craft.

The main goal of our project is to create images that are consistent with the textual descriptions provided. This can be done by training a deep neural network to understand the meaning of the text and then generate an image that captures the key objects, attributes, and relationships described in the text. The development of such a system requires a dataset of text-image pairs, where each pair contains a textual description and a corresponding image. This dataset is used to train the AI model to understand the relationship between text and images.

Once trained, the AI model can then generate new images based on new text descriptions. This can be useful in various applications such as creating images for books, games, and other media where images are needed but not yet created. Relevant existing systems do not provide these features on one platform. Multiple websites provide just only to edit images from image to image. Our website provides a unique image by your imagination, just from a natural language text description.

Through our web-based application, people will allow writing a text description of up to 250 characters and the AI magic will generate amazing images within a few seconds. Overall, text-to-image generation is a challenging task, but with the advances in deep learning and computer vision, it has become a promising area of research.

## 1.5 Proposed Solution

We introduced Automated Text-to-Image Generator which resolves the drawbacks or limitations of DALL.E2, Jasper Art, and Nightcafe AI. Automated text-to-image generator generates images within a few seconds. Automated text-to-image generator provides the best quality image by using diffusion models.

We provide a web-based application for artists, photographers, and those who love creativity. People can create amazing images from their own imagination world by just writing a text description in Natural Language.

### 1.5.1 Administrator Module:

The modules of our proposed system "Automated Text-to-Image Generator" are as stated here:

- View Home Page
  - Logo
  - About us
  - Help
  - Sign up/Login
- Sign-up Form
  - Create your account
  - Enter your email
  - Create a Strong Password
  - Retype a Password

- Continue with Google account
  - Continue with Facebook account
  - Continue with Instagram account
  - Verify your Phone Number
  - Welcome to the World of Imagination!
- Login Form
  - Enter your email
  - Enter password
  - Continue with Google account
  - Continue with Facebook account
  - Continue with Instagram account
  - Verify your Phone Number
  - Welcome to the World of Imagination!
- Services provided by Automated Text-to-Image Generator
  - Generator
  - My Collection
- Logout
  - Shutdown the system

## 1.6 Intended Market of Product

Currently, we're only aiming to provide the best platform for creating amazing art images with a natural language text description.

The scope of our proposed system is to provide people ease to maintain a convenient and easy-to-use application for users, trying to create amazing images by a text description. Our system will overcome the limitations of purchasing very expensive artistic art and also overcome the limitations of the DALL.E 2 application. Automatic Text to Image Generator will provide the online editor to create interesting and your favorite art images at a low price in Pakistan.

## 1.7 Intended Users of Product

Automated Text-to-Image Generator is for people who need content and need it quickly. In the world of online commerce, content is king. You need content that attracts attention to sell products or promote brands. And good images are key to beating the competition. There are excellent websites that provide good-quality stock images. But the online marketplace is so competitive. And other content creators use the same sites to source their images. That means you lose originality. And that could affect your SEO score and Google ranking.

Automated Text-to-Image Generator gives you unique images to use for original content. You can generate AI art from any keywords, meaning you have something original. Plus, you can tailor each generated image to match your content. And because there are so many variables, your rivals will unlikely search for the same thing. You don't have to rely on the same old stock photos.

Our proposed "Automated Text-to-Image Generator" AI web-based system has a target market as follows:

- Artist
- Graphic Designers
- Photographers
- Passionate

## 1.8 Software Process Model

As a software developer of the project, We would like to use Agile. The Agile SDLC Model will be the most suitable model to select for our project because Agile is derived from agility and flexibility to adapt to new changes easily and quickly.

### 1.8.1 Process Model Introduction

The Agile SDLC model refers to a software development approach based on iterative development. Each iteration is considered as a short time "frame" and involves a team working through a full software development life cycle including planning, requirements analysis, design, coding, and testing before a working product is demonstrated to the end user.

The phases of the Agile model are as follows:

- Requirements Gathering
- Design the Requirements
- Construction/Iteration
- Testing/Quality Assurance
- Deployment
- Feedback

The workflow process of agile model is shown in the below figure 1.1:

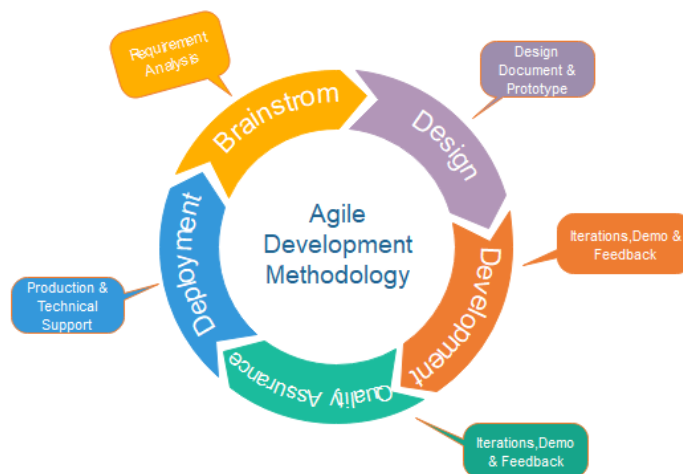


Figure 1.1: Workflow of Agile SDLC Model

In this SDLC model, we would like to follow the methodology of Agile Scrum. Agile Scrum is one of the latest software development approaches or methodologies which helps teamwork together in order to achieve quality results within a quick span of time. It's easily adaptable, easy to manage the risk, quality improvement, one-to-one and team attractions between the stakeholders in order to share details, Analyze the current position, and get and set future steps and directions for the project. Every iteration will consist of one to two-week sprints and we will build the most important features first and get the deliverable product. This methodology effectively works for smaller teams and repeated relationships with clients.

### 1.8.2 Justification

Scrum is currently running as the latest and widely used development model around the globe due to its speedy development and deliverable releases as well as the collective and collaborative approach of the whole stakeholder team to achieve a unique goal. In this project, we are assuming two group members as the Development team, the supervisor as the Project manager, and the team of teachers who is responsible for Project work assessment as Product owners. We have successfully taken approval for the project, and also have decided on the project scope and the deliverable of different phases so now we are looking forward to the requirement specification in order to proceed with development according to scrum phases.

Compare scrum methodology to other process models scrum allow products to be built faster. With the sprint-based model, the time is estimated (less than 15 days) to perform a set of tasks to develop. The scrum team receives feedback from stakeholders after each sprint. If there are any problems or improvements, to have more useful iterations, the scrum methodology can easily and rapidly modify product targets during future sprints. While using waterfall methodology is a linear process model, where requirements are gathered at the beginning of the project, and then a sequential project plan is developed to fulfill those requirements. There is no interference from the customer after the start of development on the waterfall model and if there is a problem raised after the third or fourth phase then you have to begin working on it from scratch. That's why we use the scrum methodology.

### 1.8.3 Steps of Process Model

The 5 common phases of scrum methodology are as follows:

1. Product Backlog Creation
2. Sprint planning and creating a backlog
3. Working on sprint
4. Testing and Product Demonstration
5. Retrospective and the next sprint planning

**Step 1. Product Backlog Creation:** This is the very first phase of spiral development, in which we gather the requirement and analyze the system. We also made SRS also in this phase.



**Step 2. Sprint planning and creating backlog:** In this phase, we identify the risks in our project and make alternative solutions also determine what will be our sprint duration. We are planning it in the short sprint which will allow us to release the working version of a product more frequently. As a result, customer feedback will be received more often, and all the possible bugs and errors will be revealed in time.

**Step 3. Working on sprint:** This is the core development phase, in which we start coding and developing the product according to requirements that we gathered in the product backlog so after actual user stories for the current phase are chosen, the development process begins. To track the current working process, a task board is commonly used.

**Step 4. Testing and Product Demonstration:** In all of our sprints, the output will be potentially a working product increment that can be able to present to the Product Owner. After each iteration, the development team creates a new version of a software product with increased value. During the Sprint Review, which is the end part of every sprint, the overall results can be demonstrated and analyzed. On the basis of all this info, the stakeholders can take a decision about further project changes and plan the next sprint.

**Step 5. Retrospective and the next sprint planning:** Our main aim for Retrospective will be to discuss the results and determine the main points or actions that need to be taken for the next step. An important thing is that at this stage it is the processes of work and interaction that are discussed in order to improve the output of the project with the team as a whole.

# Chapter 2

## SOFTWARE REQUIREMENTS SPECIFICATION

The Software Requirements Specification of our project contains a detailed, definition of our system functions. This Software Requirements Specification is defining the requirements of our system. because If the requirements are done well, the software design of our project flows logically and smoothly.

### 2.1 Introduction

The Software Requirement Specifications will serve as a composed and documented understanding of the needs and functionalities of the application. It'll be utilized to analyze all the requirements accumulated from different stakeholders. This document is aimed to serve as input to the development team and serve as a basis for system design. It will define the product scope and it will help us to keep the system on the right path by clearly identifying the deviated path of the system. Developers will be able to track their work progress and will understand what they have to develop by using SRS. This document provides a complete description of the proposed application "Automated Text-to-Image Generator". Automated Text-to-Image Generator is a web-based application. To create amazing art images from written text in natural language. This document includes the functions, operation, purpose, and modules of the proposed system. The document is designed both for the stakeholder and developers.

#### 2.1.1 Document Scope

This document will include all sets of requirements including all functional and non-functional requirements as well as constraints related to the product and project on which the system will be developed. The number of tools and techniques and also the methodologies and the process which will be adopted in order to gather the requirement are stated in this document.

This document also will include the constraints on the system and the environments where the system will be operational. It will also consist of effecting quality factors which are compulsory to be covered in order to achieve a successful product.

Basically, the purpose of this document is to gather all needs and requirements for the product and the project in order to proceed with the analysis phase which will lead us to design architecture and design for the proposed system "Automated Text-to-Image Generator". It will also help to ensure all stakeholders view to be on the same page regarding the features and functionalities of the product.

### 2.1.2 Audience

The intended audience of this document is the people who want to get aware of the proposed application of Automated Text-to-Image Generator that how people can create amazing art images by a written description in natural language. Automated Text-to-Image Generator is for those audiences who need content and need it quickly. In the world of online commerce, everyone wants content that attracts attention to sell products or promote brands and good images are key to beating the competition.

This document is written for the following individuals are:

- Developers who can write the code of the system according to the needs and requirements.
- Designers who can build architecture and design the system according to their requirements.
- Project Managers who can keep the project on the right path.
- Integrator who cannot make any mistake while the integration of the system.
- Quality assurance engineers who can review the quality specifications, cross-check them with requirements and provide meaningful feedback.
- Product Owners who can have total information that what will be the features and functionalities of the product.

## 2.2 Functional Requirements

Functional requirements are the product features or functions that developers must implement to enable users to accomplish their tasks. So, it's important to make them clear both for the development team and the stakeholders. Generally, functional requirements describe system behavior under specific conditions. Some important functional requirements of our system "Automated Text-to-Image Generator" are stated here:

### 2.2.1 Functional Requirements

The functional requirements of the proposed system "Automated Text-to-Image Generator" are as shown in the below table 2.1:

Table 2.1: Functional Requirements

FR No.	Functional Requirements	Description
FR1	NLP Capability	System has the ability to understand and interpret the meaning of the text input and generate an image that accurately represents the description through Natural Language Processing (NLP).
FR2	High-quality Image	User can generate high-quality, realistic images that are indistinguishable from real-world photographs.
FR3	Multiple Iterations	The system will produce multiple iterations of the generated image.
FR4	Flexibility	The system will handle a wide range of text inputs, including captions, headlines, and paragraphs, and generate a diverse range of images, including objects, scenes, and people.
FR5	Description Limit	Our system has the description limit is up-to 250 characters.
FR6	Scalability	User can generate images quickly and efficiently, even with large amounts of text input.
FR7	Originality	Our system can generate completely original images and not just variations of pre-existing images.
FR8	Transparency	The system will provide an explanation of how the image was generated based on the text input.
FR9	Error Handling	Our system has the ability to handle errors and provide meaningful feedback to the user when the input text is not understood or does not result in a good image.

## 2.3 Non-Functional Requirements

Nonfunctional Requirements (NFRs) define system attributes such as security, reliability, performance, maintainability, scalability, and usability. They serve as constraints or restrictions on the design of the system across the different backlogs. The non-functional requirements of our proposed system "Automated Text-to-Image Generator" are stated here:

- Usability
- Security
- Maintainability
- Reliability
- Availability
- Performance

### 2.3.1 Software Quality Attributes

Some basic factors which are directly related to the success and enhancement of our software quality are as follows:

**1- Usability** The system will be usable enough that it will be user-friendly, easy to learn, and will have easy navigation. **Justification:** Usability and Learnability are closely linked with each other. Improving one of these will result in the improvement of the other. We also have an assumption that our users will be literate enough to use the system but we also have to provide easy IDE in order to deal with illiterate users. The application design and operation are easy to use for both the user and admin.

**2- Security** The system will be secure to the extent that it will prevent unauthorized access and will prevent information loss and also will be secure enough. **Justification:** The account of every user will be secured by a password which will be set after verification through email or number by text thus no one else will be able to access the account of any user except that specific user. As this is an Android application so all the data will be stored at the host so there is not a high risk of data loss.

**3- Maintainability** The system will be able to add new features and modify features according to new technology and trend. **Justification:** The system will add new features and update things from time to time hence maintainability will be achieved by the system "Automated Text-to-Image Generator".

### 2.3.2 Other Non-Functional Requirements (e.g.Legal, Platform, etc.)

The proposed application "Automated Text-to-Image Generator" is an online web-based AI application for Artists, Editors, Photographers, etc. And as we launch this website in Pakistan. So, the proposed application should register with the government of Pakistan to make the application legally independent.

## 2.4 Requirement Gathering Techniques Used

The process of gathering requirements includes identifying the required project-related requirements. To build solutions in the form of products, services, and software. We applied different Requirement Elicitation Techniques after observing the problem nature and need of the phase as follows:

### 2.4.1 Analyzing Existing Literature

This technique was very useful for us because having set that other existing systems have some limitations but it is also a fact that a good amount of work is already has been done in the field of Artificial Intelligence (AI) as our proposed system of "Automated Text-to-Image Generator". So, studying available litterateur in this domain also helped us in cross-checking the requirements which are already gathered.

### 2.4.2 Brainstorming

We conducted brainstorming sessions with our fellows and teachers that helped us gather more information and requirements for this system. It helped us generate as many ideas as possible. After gathering raw information we reshaped and combined these ideas. It increases the creativity of the SDLC team, and the best of each individual is used to address challenges. We can benefit from brainstorming as a critical requirement elicitation method.

### 2.4.3 Questionnaire

Questionnaires, allow us to collect information from many people in a relatively short amount of time. When using questionnaires, the questions should be oriented and organized by a feature or project target. We planned to conduct a questionnaire technique online from the users of existing applications. It helped us to learn from the limitations of existing systems and to overcome these drawbacks from our application as well.

## 2.5 Time Frame

The time duration of our proposed system "Automated Text-to-Image Generator" is shown below in table 2.2:

Table 2.2: Time Frame

Sr. No.	Phase	Duration
1	Inception	4 Days
2	Elicitation	3 Days
3	Elaboration	7 Days
4	Negotiation	2 Days
5	Specification	8 days
6	Validation	3 Days
7	Requirement Management	6 Days
8	SRS Document	6 Days

# Chapter 3

## SOFTWARE PROJECT PLAN

### 3.1 Deliverables of the Project

The deliverables of the project is that the Image captioning has recently gathered a lot attention specifically in the natural language domain. There is a pressing need for context based natural language description of images domain. There is a pressing need for context based natural language description of images however this may seem a bit far fetched but recent development in fields like neural networks computer vision and natural language processing has paved a way for accurately describing images i.e. representing their visually grounded meaning we are leveraging state of the art techniques like convolutional neural network can recurrent neural network run and appropriate datasets of images and their huma perceived description to achieve the same we demonstrate that our alignment model produces results in retrieval experiments on datasets such as flicker.

#### 3.1.1 SRS Document:

This This document includes information that what features of the system are actually user requirements and all the strategies that are followed in requirement specification and requirement gathering.

#### 3.1.2 Design Document:

This document includes information that what features of the system are actually user Interface and all the strategies that are followed in making the system user-friendly and an attractive Graphical user interface.

#### 3.1.3 Database:

This store all data functionality which using in front-end and backend.

#### 3.1.4 Website Frontend:

Interactive interface provide users friendly frontend web page design through this users easy interact with website.

#### 3.1.5 Database Connection:

Database connection make connection between front end server and backend server.

#### 3.1.6 Website Backend:

Design backend of the webside through python.

### 3.1.7 Image Generator:

Text to image generation through artificial intelligence.

### 3.1.8 User Manual:

This is a detailed guide for end-user that will help users to do a task without any extra effort. These are the instruction that guide the user that how to write correct text which create image.

## 3.2 Software Project Management Plan

A software project plan is a collection of documents that outline the tasks and timeline of your software development. Software project plans typically include projected start and end dates, launch plans, requirements, configurations, installation procedures, databases, and training standards.

### 3.2.1 Project Planning

Project planning of our project is important at the early stage of project development because it will determine the scope of our project. Project planning is necessary for the completion of our project otherwise our project may lead to the scope creep, or stage of failure. To complete our project within the given schedule, we create several project plans that will keep track of resources, tools, techniques, used in our project and that will help us to achieve the final objectives of our project.

**3.2.1.1 Milestones Plan** Milestones are simply used to help break up the project and improve progress monitoring. Table 3.1 shows all milestones of our project, total duration, start date and end date of the project's milestone plan.

Table 3.1: Milestones Plan

Sr. No.	Milestone	Duration	Start Date	End Date
1	User Sign in page.	2 Weeks	24th October, 2022	8th November, 2022
2	Home page	3 Weeks	7th November, 2022	21st November, 2022
3	Store text through NLP in the system	6 Weeks	17th November, 2022	17th December, 2022
4	Convert text to image form	6 Weeks	28th January, 2023	21st April, 2022
5	Testing the system	4 Weeks	22nd April, 2022	22nd June, 2022



**3.2.1.2 Documentation Plan** The Documentation Plan of our project is shown in the below table 3.2:

Table 3.2: Documentation Plan

Sr. No.	Document	Distribution	Duration
1	Project Proposal	Warishah	2 Weeks
2	Chapter 1, 2	Adeen, Warishah	4 Weeks
3	Chapter 3, 4	Adeen, Warishah	4 Weeks
4	Chapter 5, 6	Warishah, Adeen	3 Weeks
5	Chapter 7, 8	Warishah, Adeen	4 Weeks
6	User Manual	Adeen	2 Weeks

**3.2.1.3 Resources Plan** A resource plan identifies, organizes, and lists the resources required to complete a project. Table 3.3 shows all the tasks that are needed to complete in the project.

Table 3.3: Documentation Plan

Sr. No.	Resource	Task	Duration
1	Warishah, Adeen	Requirement Gathering	3 Weeks
2	Warishah, Adeen	Develop SRS Document	3 Weeks
3	Warishah, Adeen	Develop Design Document	2 Weeks
4	Warishah, Adeen	Develop Project Plan	2 Weeks
5	Warishah, Adeen	Build Prototype	1 Weeks
6	Warishah, Adeen	GUI	2 Weeks
7	Warishah, Adeen	Final Product	2 Weeks

**3.2.1.4 Quality Plan** The Quality Plan of our project is shown in the below table 3.4:

Table 3.4: Quality Plan

Sr. No.	Quality Attributes	Test Start Date	Test End Date	Description
1	Performance	21st September, 2022	28th September, 2022	While considering the performance of the system, it will be testing how much time it will take to respond to every command of the user.
2	Usability	2nd October, 2022	13th October, 2022	In this test, we will test how many user interfaces are interactive and all properties of human-computer interaction are applied.
3	Maintainability	21st October, 2022	8th November, 2022	This test contains a test that will check the maintainability of the system. We are configuring our system so that later every change will be easy.
4	Reliability	5th November, 2022	22nd November, 2022	This test contains a test that will check the reliability of the system. We are configuring our system that later every change will be easy.

### 3.2.2 Project Scheduling

Project scheduling is a project management process that consists of creating and managing a schedule to organize the tasks, deliverables, and milestones of a project on a timeline.

**3.2.2.1 Gantt Chart** A Gantt chart is one of the most popular and useful ways of showing activities (tasks or events) displayed against time. In fig 3.1 all the tasks of our project is shown and displayed against the time period which we plan to complete our task in that time. So the below figure shows our tasks and time period of completion of our project's tasks.

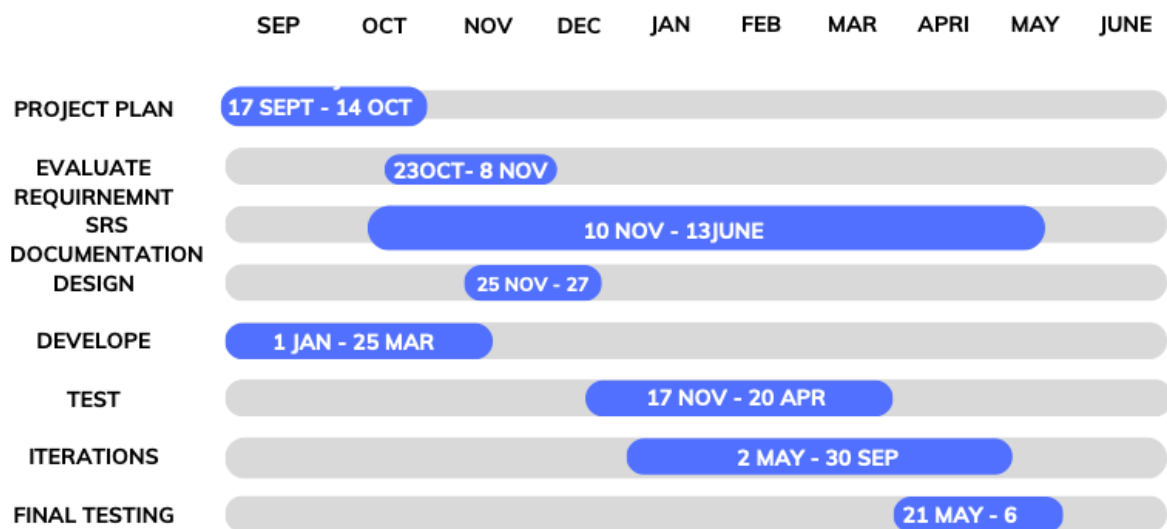


Figure 3.1: Gantt Chart

**3.2.2.2 Work Breakdown Structure** A work breakdown structure visually organizes project deliverables into different levels based on dependencies. It's essentially your project plan in a visual form, with your project objective at the top, then dependencies and sub-dependencies below. Fig 3.3 shows all the dependencies and sub-dependencies of our project. Work breakdown structures defines below figure 3.2:

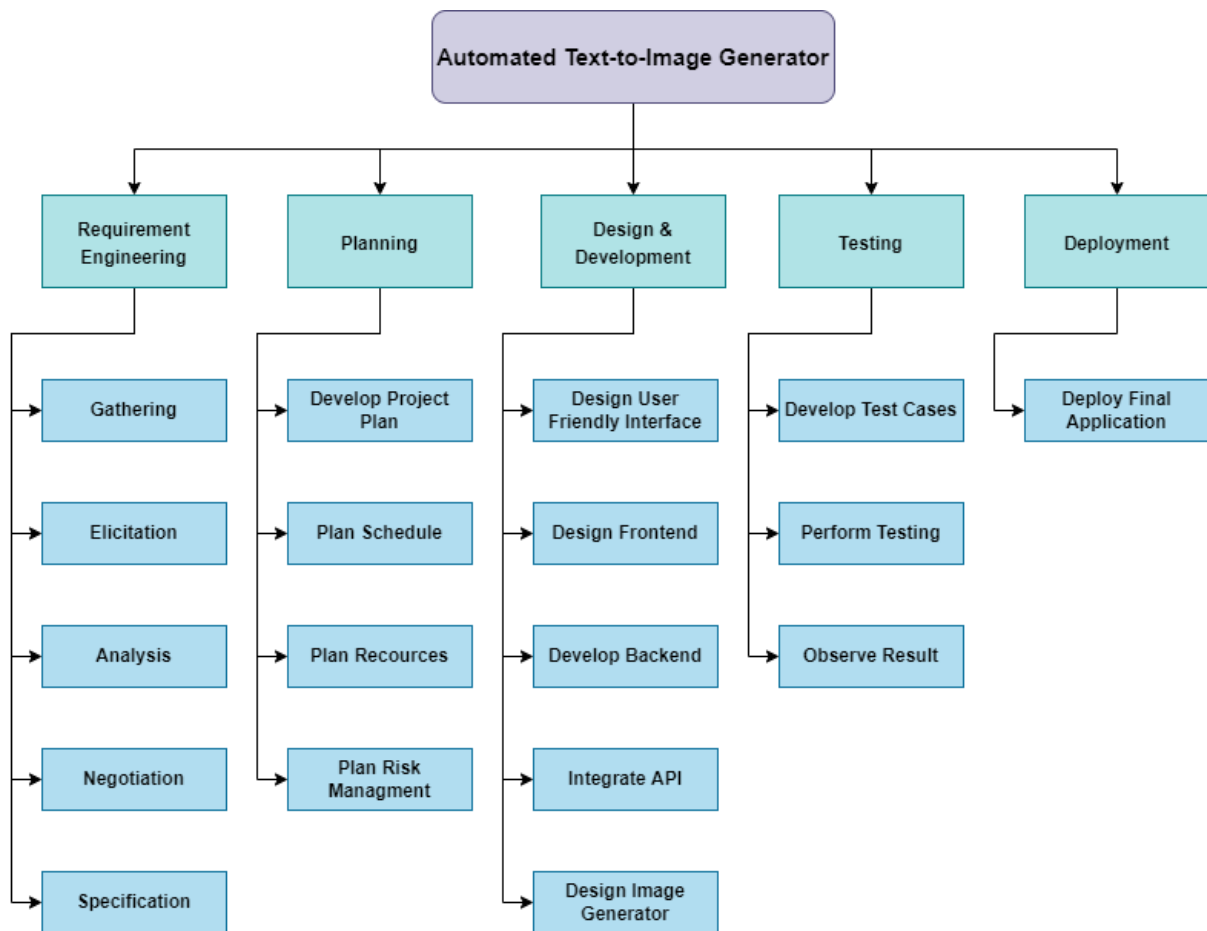


Figure 3.2: Work Breakdown Structure

**3.2.2.3 Critical Path Method** The critical path method is a technique that allows you to identify tasks that are necessary for project completion. The critical path in project management is the longest sequence of activities that must be finished on time to complete the entire project. Fig 3.2 shows the critical paths of our project that must be completed on time so our project will be completed on time.

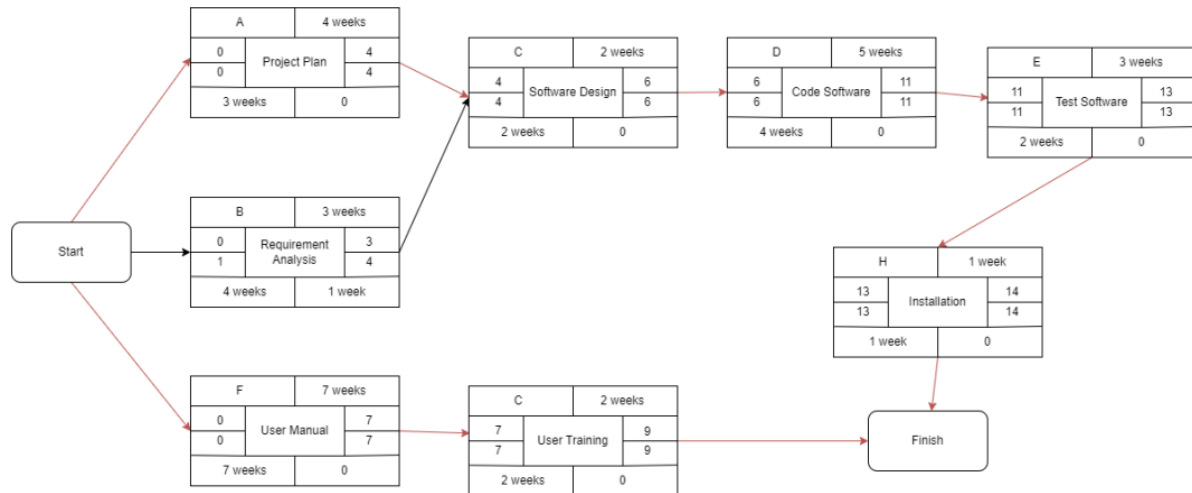


Figure 3.3: Critical path

## 3.3 Managerial Process

Management process is a process of setting goals, planning and/or controlling the organizing, and leading the execution of any type of activity, such as a project or a process.

### 3.3.1 Management Objectives and Priorities

In automated image generator our main goal is provide high quality images to the content creators and bloggers.

#### 3.3.1.1 Objectives:

- The successful Implementation of all the tasks and sub-tasks of the project.
- Effective communication between all the team members and stakeholders.
- Achievement of project goals within given constraints.
- Deliver a project on time and that is accordingly to the requirements and needs of the customer.

#### 3.3.1.2 Priorities:

- Make sure all the quality objectives and met throughout the project.
- The project deliverable must be completed as planned.
- Focus on the main features of the system.
- The project must be delivered on time within the allocated budget.
- The project should not fall behind or exceeds the scope of the project.
- Track all milestones, deliverables, and change requests which occur during the project.

### 3.3.2 Assumptions and Constraints

**3.3.2.1 Assumptions:** Following are a few assumptions that have been made from the view of project management:

- The scope of the project will not change throughout the project.
- We have all the resources required to develop and deliver the project.
- Team members have all the necessary skills and knowledge to develop the application.
- Two team members are enough to develop, test and deploy the system.
- We have all the tools needed for the project to be produced and completed on time.

**3.3.2.2 Constraints:** Every project has to manage four basic constraints: scope, schedule, budget, and quality. The success of a project depends on the skills and knowledge of the project manager to take into consideration all these constraints and develop the plans and processes to keep them in balance.

## 3.4 Project Risk Management

Risk can refer to the loss we can expect, if a risk occurs in the project” automated image generator “then it can impact our project’s scope and its schedule. Therefore, to identify and eliminate the risk in our project, we follow the Risk Management process in order to identify, and check the risk probability and its impact, and will try to mitigate it. If we don’t follow the risk management process, then it can negatively impact our project overall objectives. Our team’s project managers are responsible for managing the risk management process over the lifetime of a given project. In automate image generator there is a risk of loss of user data or personal data so by avoid that risk we testing the system many times and check is our customer data is safe or not.

### 3.4.1 Risk Management Plan

Risk is the probability of something uncertain and unusual happening that affects the schedule, quality, and performance of the software being developed. Project risk management is the process of identifying the potential risk that may occur during the project and planning ways to minimize, avoid and mitigate risks so that the project remains on track and meet its desired goal. Risk is something that could impact or in other words affect the scope, schedule, or budget of the project Deciding on the approach to and planning of risk management.

**3.4.1.1 Risk Identification:** Identifying likely risks that might affect the project as well as them features.

**3.4.1.2 Qualitative Risk Analysis:** Performing a qualitative analysis of risks and creating conditions to prioritize their effects on project objectives.

**3.4.1.3 Quantitative Risk Analysis:** Measuring the probability and consequences of risks and estimating their implications for project objectives.

**3.4.1.4 Risk Response Planning:** Developing procedures and techniques to enhance opportunities and reduce threats from risks to the project’s objectives

**3.4.1.5 Risk Monitoring and Control:** Monitoring residual risks, identifying new risks, executing risk reduction plans, and evaluating their effectiveness throughout the project life cycle. Organizations will often combine several of these processes into one process with the main purpose of identifying all the risks and developing responses for those with the greatest consequences to the project objectives as quickly and efficiently as possible. Planning for risk early on in the project and putting mitigation measures in place could save a project from premature termination (Darnall, 2001). Project managers are required to educate and reassure their teams (and clients) about the risk processes and help them understand that risk management is an essential element of any project area. The PMIS could be an enabler in this regard.

**3.4.1.6 Purpose:** The purpose of a risk management plan in the context of a text-to-image generator system is to identify, assess, and prioritize potential risks that may impact the development, implementation, and operation of the system. The goal is to minimize the negative consequences of these risks and to ensure the success and sustainability of the project. The risk management plan should include strategies to mitigate or avoid identified risks, contingency plans for managing potential impacts, and methods for monitoring and reviewing the effectiveness of the risk management process.

**3.4.1.7 Roles and Responsibilities:** The role and responsibilities of our software project team are describe in the below table 3.5:

Table 3.5: Documentation Plan

Name	Title	Project Roles	Responsibilities
Syeda Warishah Bukhari	Group Leader	Frontend and backend development, project management, implementation of all the functionalities of our project.	Front side website design, user interactive design, database connection, linked with backend, model implementation.
Adeen Asad	Group Member	Requirement analysis unit testing, error debugging, documentation, requirement gathering	SRS document develop test cases Develop user manual.

### 3.4.2 Risk Management Activities

Risk management activities are performed to manage any potential risks that may occur.

**Risk 1:** Completing the project within the deadline **Consequences:** If the project is not completed on time it will impact the acceptance and success of the project. **Strategy:** We have planned each and every phase of the project. We build network diagrams and found a critical path so that we complete the project before the deadline.

**Risk 2:** Change in requirements **Consequences:** Too many changes may lead to the project losing its scope. **Strategy:** Mitigation- In the development of the system we are making code more and more flexible so that it will be easy for anyone to understand and change.

**Risk 3:** Product Quality **Consequences:** If the system does not provide the required quality, It will not be acceptable and lead to project failure. **Strategy:** Monitoring While monitoring and enabling the verification and validation from the SRS document, we can meet the system's quality

**3.4.2.1 Risk Identification** Risk identification is the process in which possible risks may occur and affect the application. To deal with such risk, risk identification beforehand is essential. to identify the risk similar applications are reviewed, and gather documents related to our system, and brainstorming is done. Once the possible risks are identified better solutions are proposed to mitigate the risk.



**3.4.2.2 Risk Analysis** Risk Analysis is the process to analyze and identify the issue that could impact our system. This process helps to avoid the risk and to find a better solution. Risk analysis is done by following these steps.

- Risk identification
- Explain the uncertainty level
- Estimate the impact of risk
- Create a Risk Analysis model
- Analyze the risk result

**3.4.2.3 Risk Response Planning** Risk response planning is the process of developing plans to reduce risks. After analyzing the risk and its consequences, we plan some strategies to overcome the risk when they occur.

**3.4.2.4 Rating Risk Likelihood and Impact** When we identify all risks then we measure what is the probability of occurrence of the specific risks. In this step, we find what the chances of risks are and what will have an impact on the project. We can only measure the known risks and take mitigation techniques to handle them. The probability of occurrence of risk no 2 is high if this risk occurred then the project will result in failure or delay.

**3.4.2.5 Risk Monitoring and Control** Risk monitoring and control is the process of identifying analyzing and planning for the risk that was not identified before and discovered during execution and managing identified risk. Throughout the process we have to track identified risks, reveal new risks implement risk response plans and measure the risk response plans effectiveness. Hence constant monitoring and control is the key to the success of the project. The inputs to risk monitoring and control are:

**3.4.2.6 Risk Assessment** Software risk assessment is the process of identifying, analyzing, and prioritizing all the possible risks(predictable and known) that may occur. As our project is a mediumscale project so there is a great chance that many risks will influence it. Each project risk requires an assessment strategy to avoid failure or loss of the project. Risk assessment is an important step in the project life cycle to prepare the team for the potential that can occur within any software project.

# Chapter 4

## FUNCTIONAL ANALYSIS AND MODELING

### 4.1 Use Case Modeling

This section contains Use Case Modeling of our project "Automated Text-to-Image Generator". It will show a view of the system from the user perspective, thus describing what our system does without describing how our system does it.

#### 4.1.1 User Stories

The general explanation of our system features from the perspective of the end user are as stated under here:

##### 1. Authentication:

- As a user, I want to login into the system using my email address and password.
- As a system administration, I want to register new users by entering their personal information.
- As I want to create an account through the system.

##### 2. Explore about the application:

- As a user, I want to know about the application by downloading the research material.
- As a user, I want to see the details through About us.

##### 3. Generate Images:

- As a user, I want to create realistic images by writing text description in natural language.
- As a user, I want to create different variations of an image from image.
- As a user, I want to generate image in high quality.
- As a user, I want to generate multiple iterations of an image.

##### 4. Flexibility:

- As a user, I want to get a wide range of input text prompt.

**5. Scalability:**

- As a user, I want to generate images quickly and efficiently.
- As a user, I want to generate completely original images.
- As a user, I want to get some information related to the generated images.

**6. Collection:**

- As a user, I want to save my generated images as my collection.

**4.1.2 Individual Actor Use Cases**

Table 4.1: Register User

<b>Use Case No</b>	<b>01</b>
<b>Use Case</b>	Register User
<b>Actor</b>	End User
<b>Type</b>	Primary
<b>Description</b>	The end-user who wants to use our website application, first they will register themselves. After this they will receive verification email. They have to verify themselves to enter the image generator's page of the application.

Table 4.2: Login User

<b>Use Case No</b>	<b>02</b>
<b>Use Case</b>	Login
<b>Actor</b>	End User
<b>Type</b>	Primary
<b>Description</b>	The end-user who wants to use our web-based application, secondly they will login themselves by entering their email and password. After this they entered into the image generator's page of the application.

Table 4.3: Image Generator

<b>Use Case No</b>	<b>03</b>
<b>Use Case</b>	Image Generator
<b>Actor</b>	End User
<b>Type</b>	Primary
<b>Description</b>	The end-user who want to create images. So, after login themselves they can use image generator of our web-site and can create more realistic images by writing text description in natural language.

Table 4.4: My Collection

<b>Use Case No</b>	<b>04</b>
<b>Use Case</b>	My Collection
<b>Actor</b>	End User
<b>Type</b>	Secondary
<b>Description</b>	The end-user can save their generated images in to my collection page by simply click the button of "Add to Collection".

### 4.1.3 Expanded Use Cases

Table 4.5: Expanded 01 Register User

<b>Actor Action</b>	<b>System Response</b>
1. User open the application.	2. The application shows a login page.
3. If the user is not registered, he will click on the register button.	4. System will show the registration form.
5. Users input all the required data in the given fields and click on the sign-up button.	6. The system will send user verification email.
7. User will verify himself through email notification.	8. System will notify users and show a message "Successfully registered".

#### UC No.1 Extension:

- System generates an error message if the user leaves any field blank.
- System generates an error message if user ignore the requirements for a strong password.
- User input data in all blank fields.
- System will show the message of "Successfully Register".

Table 4.6: Expanded 02 Login User

<b>Actor Action</b>	<b>System Response</b>
1. This use case begins when user opens the application.	2. System shows login page to User.
3. User input his credentials i.e. Email and password.	4. System validates that the credentials are correct or not.
	5. Email and password are correct.
	6. System directs the user to image generator page.

#### UC No.2 Extension:

- User enters incorrect credentials i.e. email or password.
- System generates an error message stating email or password is incorrect.
- System prompts user to enter correct credentials.
- User enters the correct email and password.
- The system will display the image generator page successfully.

Table 4.7: Expanded 03 Image Generator

Actor Action	System Response
1. User enters a text prompt.	2. System generates the 4 images to the relevant text description.
3. User clicks on the button of add collection to save the generated images.	4. System saves the generated images into the page of my collection.

**UC No.3 Extension:**

- System will not generate any image if the text limit exceeded from 250 characters.
- User enter the text with the limit of 250 characters.
- System generates 4 images in the output.

Table 4.8: Expanded 04 My Collection

Actor Action	System Response
1. User see their image creations in the collection page.	2. System shows the collection history of the user.
3. User deletes their saved image collections.	4. System removes the deleted images from the collection page.

**UC No.4 Extension:**

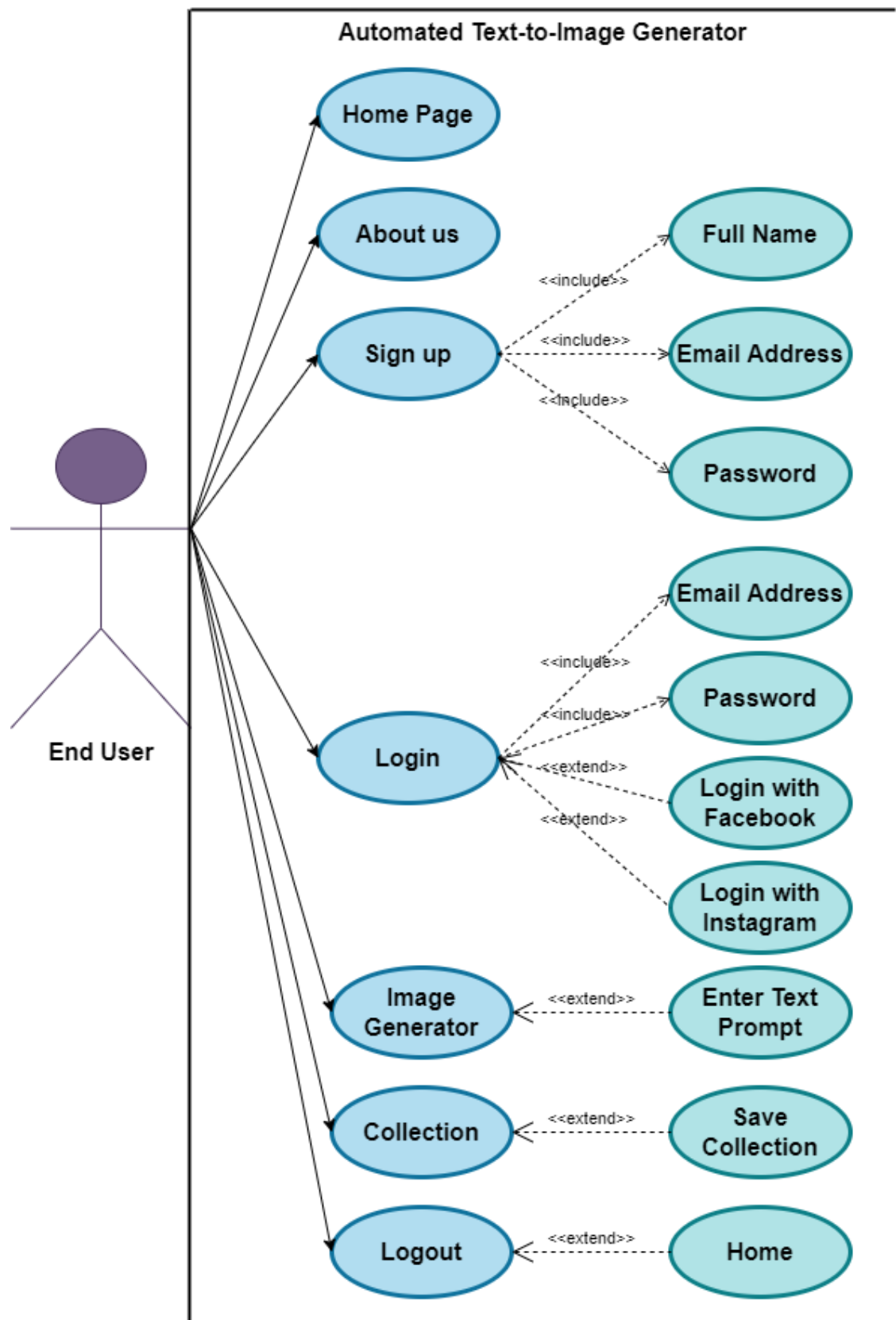
- System will show all the image collection history of the user.
- System will permanently removes the deleted images.

#### 4.1.4 Use Case Diagrams

Our system has only one use case diagram which represents the overall functionality of the system "Automated Text-to-Image Generator" are as given below:

**4.1.4.1 Use Case Description (Full System):**

- User can visit home page without login/sign-up into the system.
- User can also view the details about the system without login/sign-up.
- To use our application, The user should sign-up with their credential details by the filling the registration form of the user.
- User can go to the image generator page by login with their email and password or login with Facebook/Instagram button to create realistic art images by the magic of AI.
- User can save their generated images in My Collection by click the button of add collection.
- User can see their saved image collection in the My Collection page.
- User can remove their saved images from the My Collection page.
- User can shutting down the system by using the logout button.



## 4.2 Functional Modeling

Functional Modelling of our project will give the process perspective of the object oriented analysis model and an overview of what our system is supposed to do. It also defines the function of the internal processes in our system with the aid of Data Flow Diagrams (DFDs).

### 4.2.1 Entity Relationship Diagram

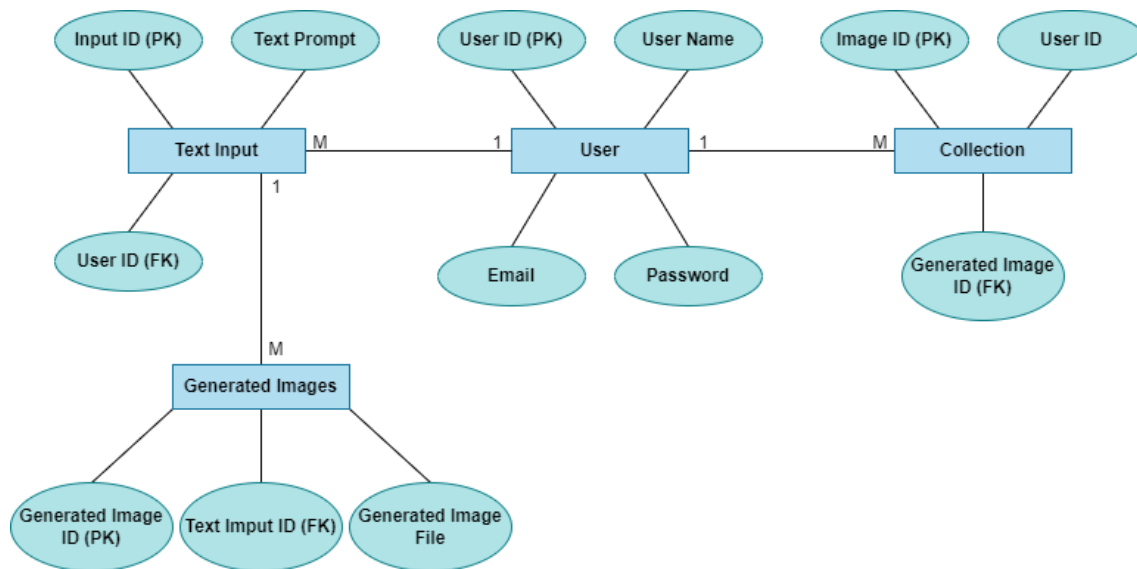


Figure 4.2: Entity Relationship Diagram

### 4.2.2 Data Flow Diagram

The Data Flow Diagram of our system elaborates the flow of data so that our project's functionalities can be made better without technical solutions or without implementing the actual system.

#### Data Flow Diagram Description:

**(End-User):** User can login and sign-up into the application. User can create realistic art images by written text description in natural language.



## 4.2.2.1 DFD Level 0

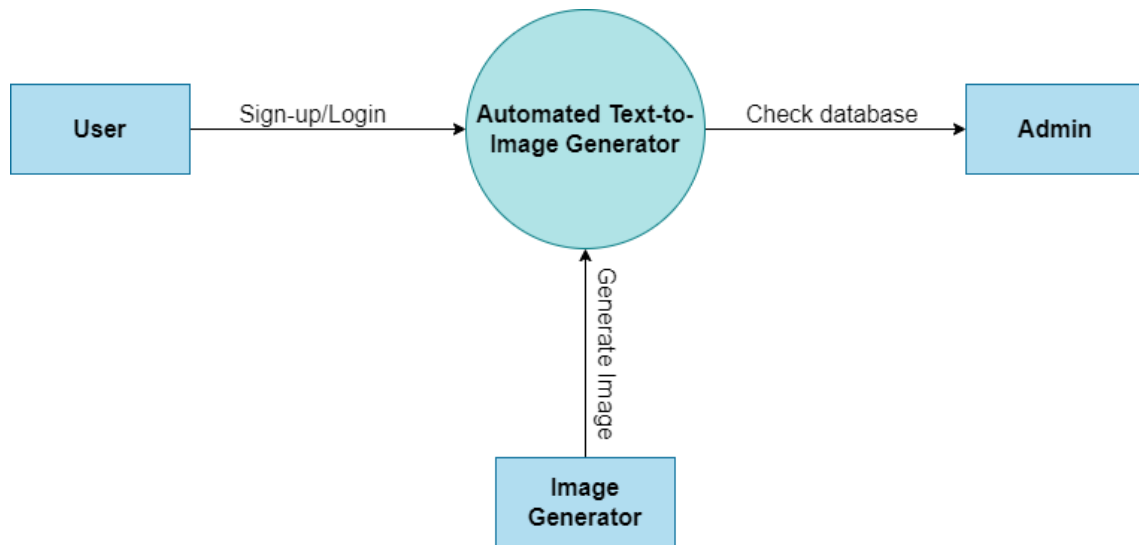


Figure 4.3: Data Flow Diagram Level 0

## 4.2.2.2 DFD Level 1

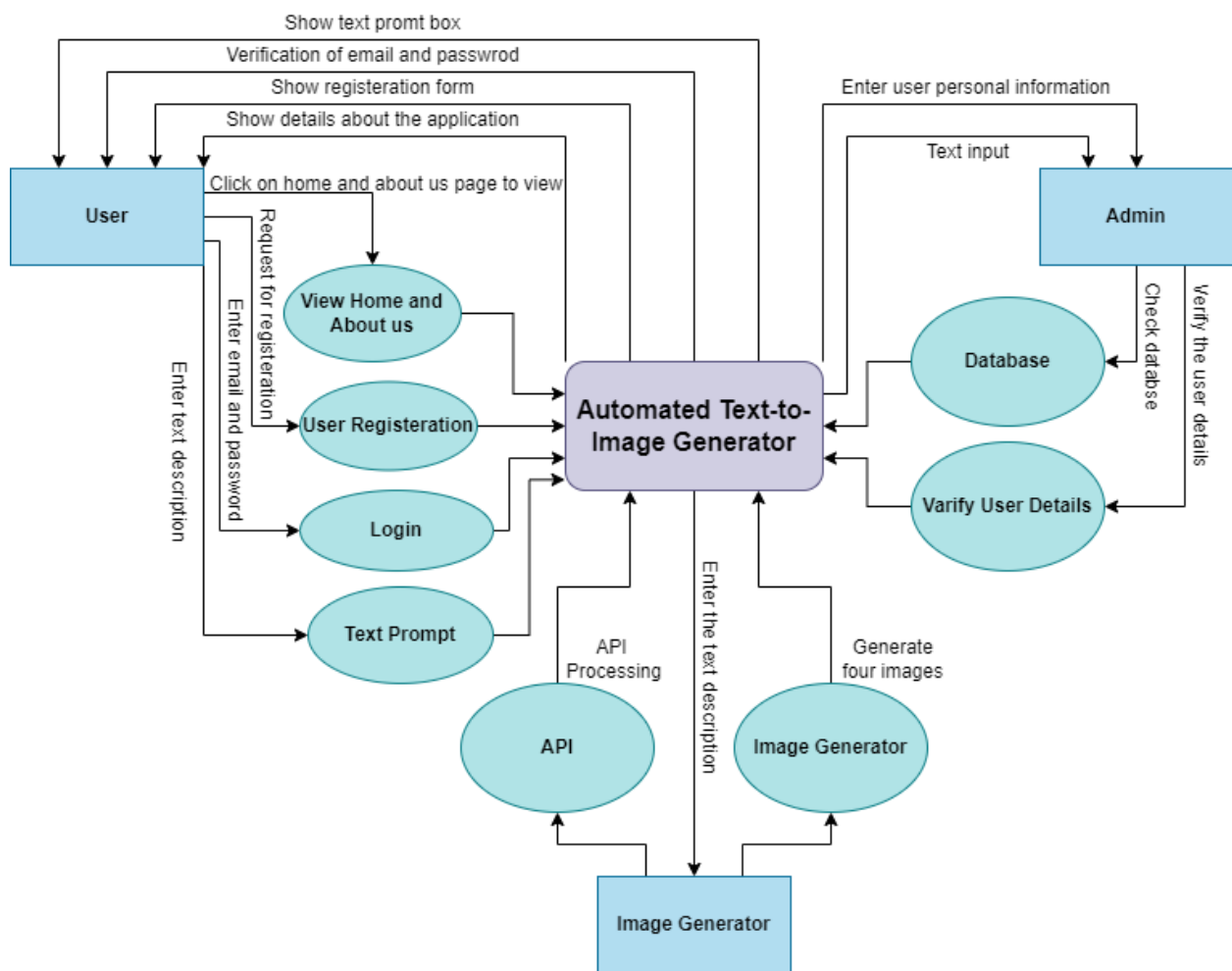


Figure 4.4: Data Flow Diagram Level 1

## 4.2.2.3 DFD Level 2

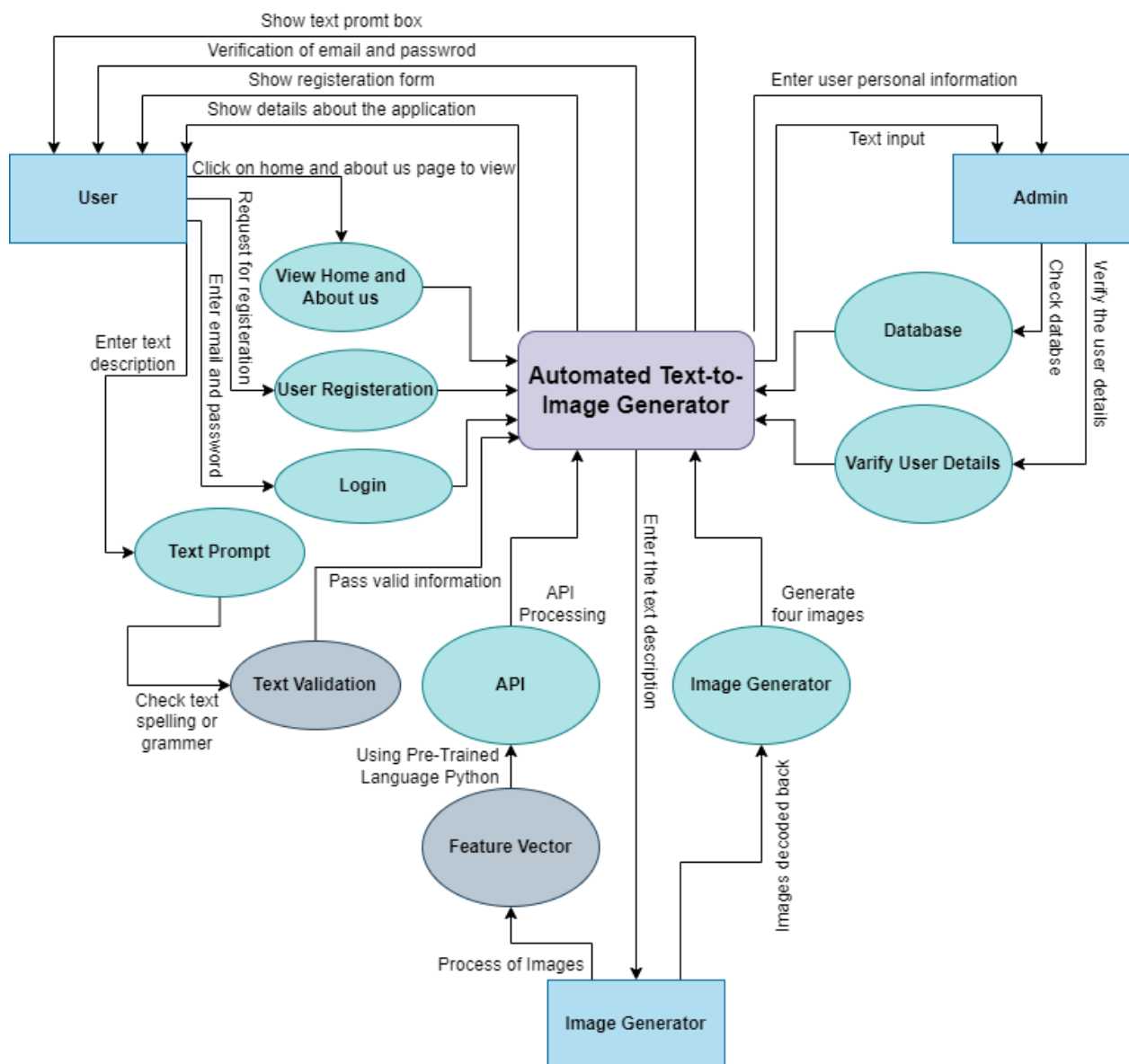


Figure 4.5: Data Flow Diagram Level 2

# Chapter 5

## SYSTEM DESIGN

### 5.1 Structure Diagrams

The structure diagrams of our project are basically a conceptual modelling of our project that describes the various components that go into our system.

#### 5.1.1 Class Diagram

The class diagrams of our project "ATIG" application is representing the complete subsystem's blueprints. Which is useful for modelling the system's constituent parts, showing the relationships among them, and describing the functions and services that each perform. The characteristics (attributes) and functions of each class are described in our project's class diagram. Figure 5.1 represents the class diagram of our system.

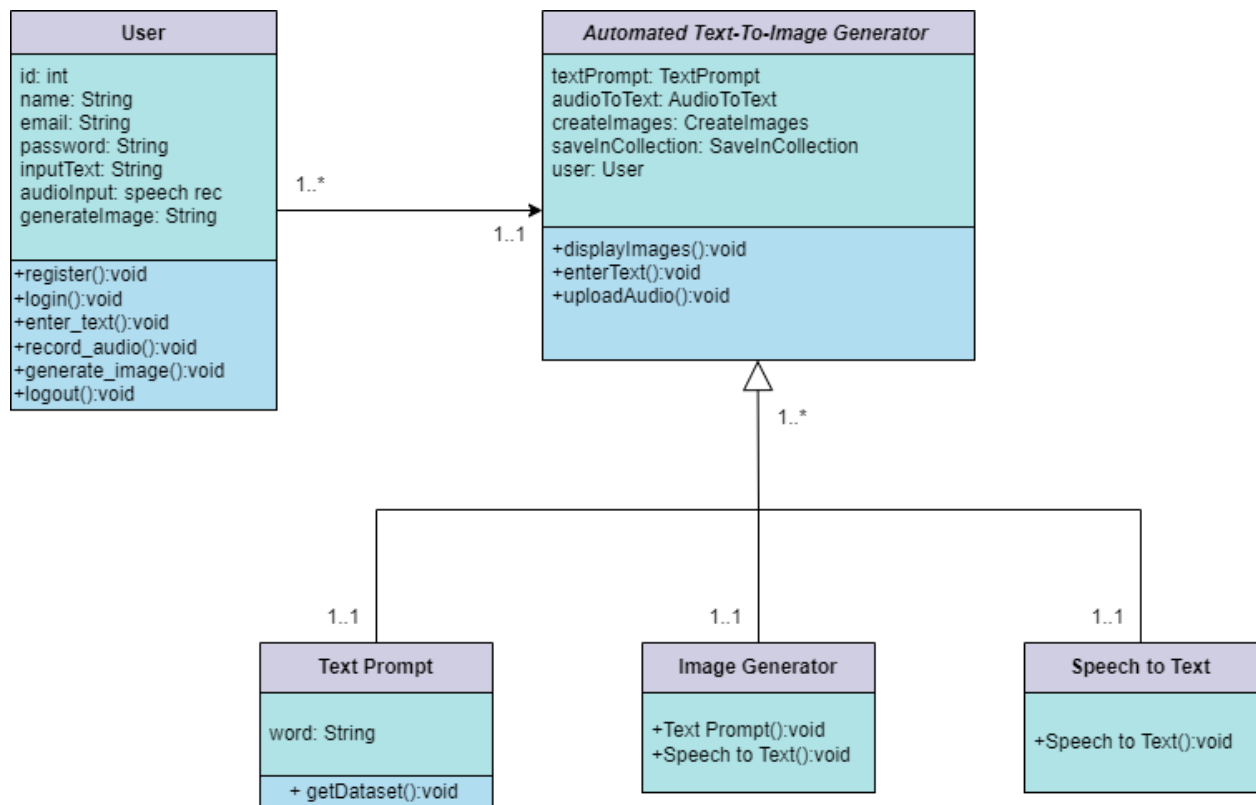


Figure 5.1: Class Diagram

#### 5.1.2 Deployment Diagrams

The "Deployment diagram" of our system "ATIG" application is representing the structure of our system's physical parts, also it shows where our system's software components are installed. This diagram is representing the relationships between our system's hardware and software components as well as the physical distribution of the processing

are shown in the below mention figure 5.2.

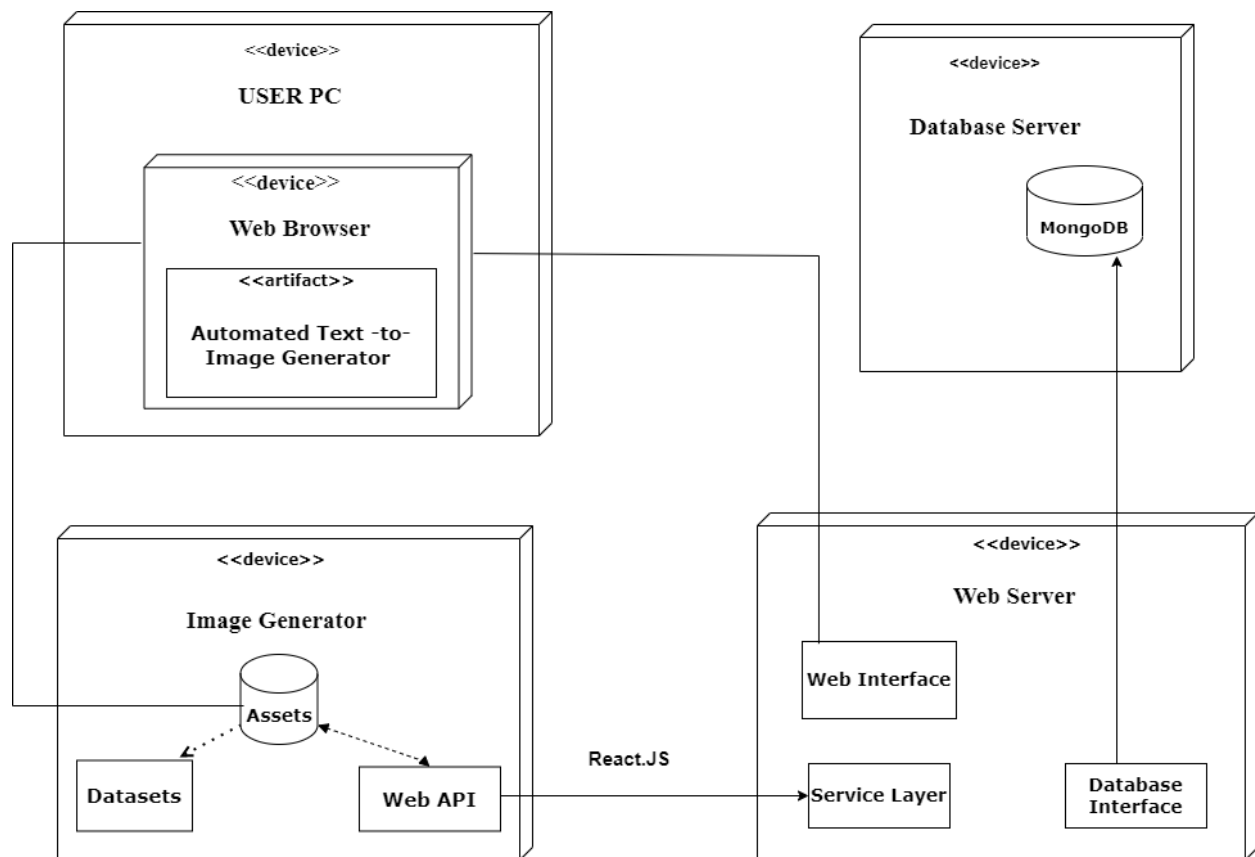


Figure 5.2: Deployment Diagram

## 5.2 Behavioral Diagrams

The dynamic elements of our system "Automated Text-to-Image Generator (ATIG) AI, web-based application" are visualised and specified using below mentioned behavioural diagrams.

### 5.2.1 Activity Diagrams

An activity diagram of our proposed system "ATIG" visually displays a series of actions and the flow of control in our proposed system. End Users, system components, or software can all perform these tasks. The Figure 5.3 is representing the "Activity Diagram" of our system:

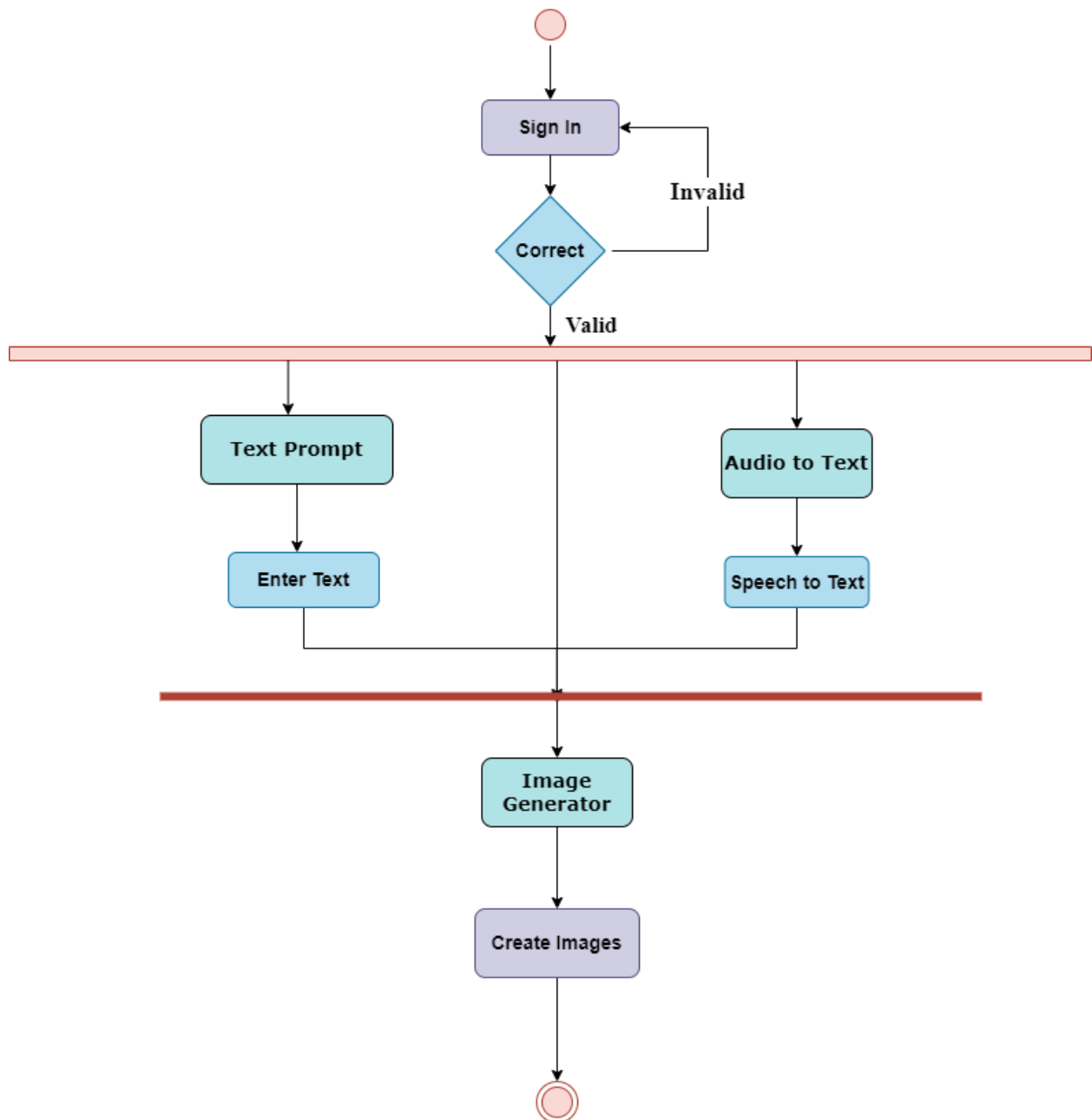


Figure 5.3: Activity Diagram

## 5.2.2 Communication Diagrams

The communication diagrams of our project depicts the interactions between different elements of the application at run-time. Communication Diagrams helps us to visualise interactions between objects in our system. The Figure 5.4 represents the "Communication Diagram" of our system:

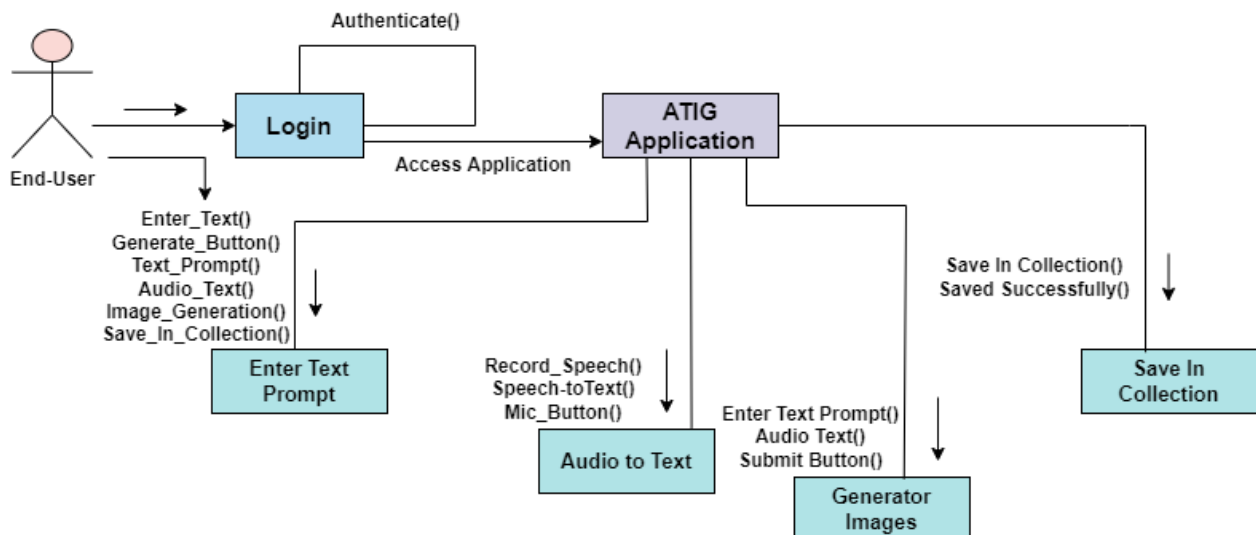


Figure 5.4: Communication Diagram

## 5.2.3 Sequence Diagrams

The Sequence diagrams of our project "ATIG" illustrates the interactions between group of objects and the order in which they occur in our system.

**5.2.3.1 User Login Sequence Diagram** The Figure 5.5 represents the "Login Sequence Diagram" of our system which is visualizing the login sequence by the end user:

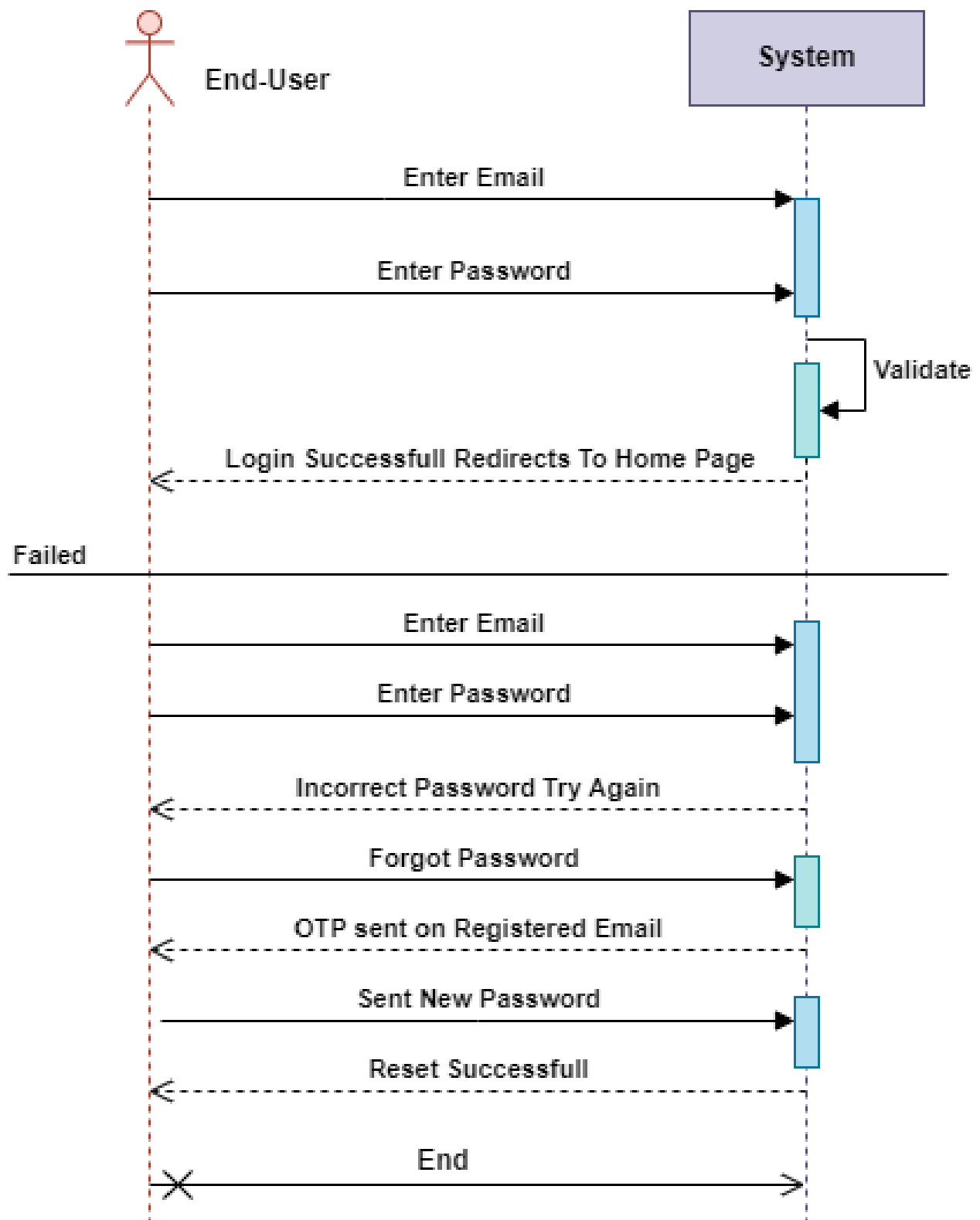


Figure 5.5: Login Sequence Diagram



**5.2.3.2 Speech to Text Sequence Diagram** The Figure 5.6 represents the "Speech to Text Sequence Diagram" of our system which Transcribes the audio into whatever language the audio is in and Translate and transcribe the audio into English:

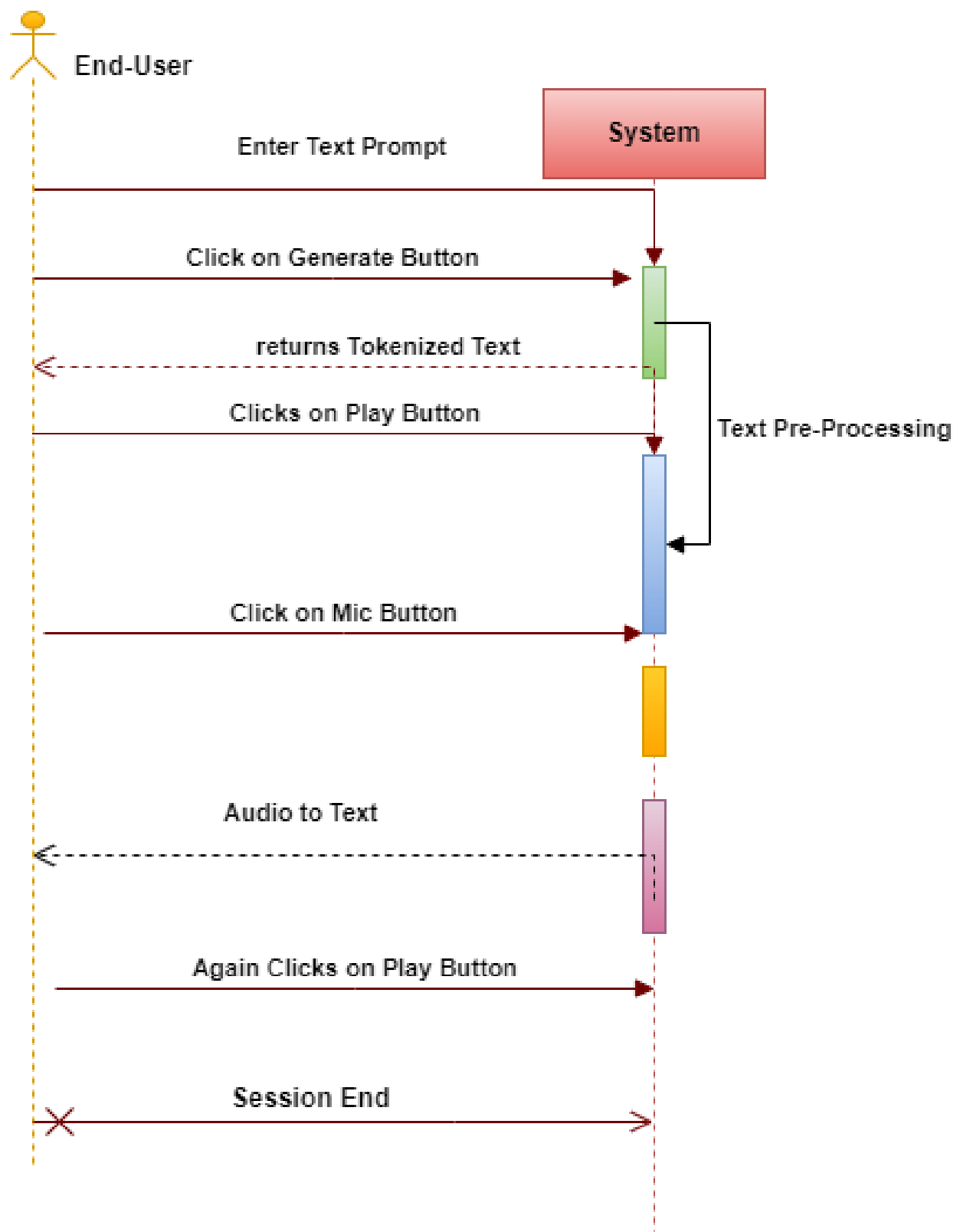


Figure 5.6: Audio to Text Sequence Diagram

## 5.2.4 View Database

**5.2.4.1 MongoDB User Database** The "MongoDB User Database" view of our "ATIG" application is represented in figure 5.14 which is showing the end-user's login data:

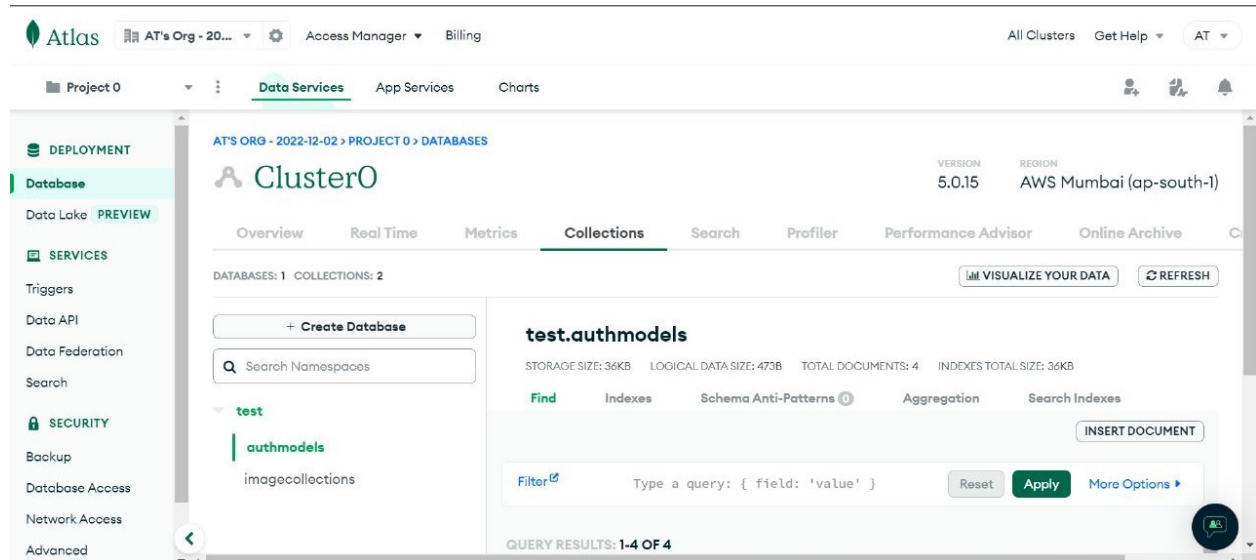


Figure 5.7: MongoDB Database

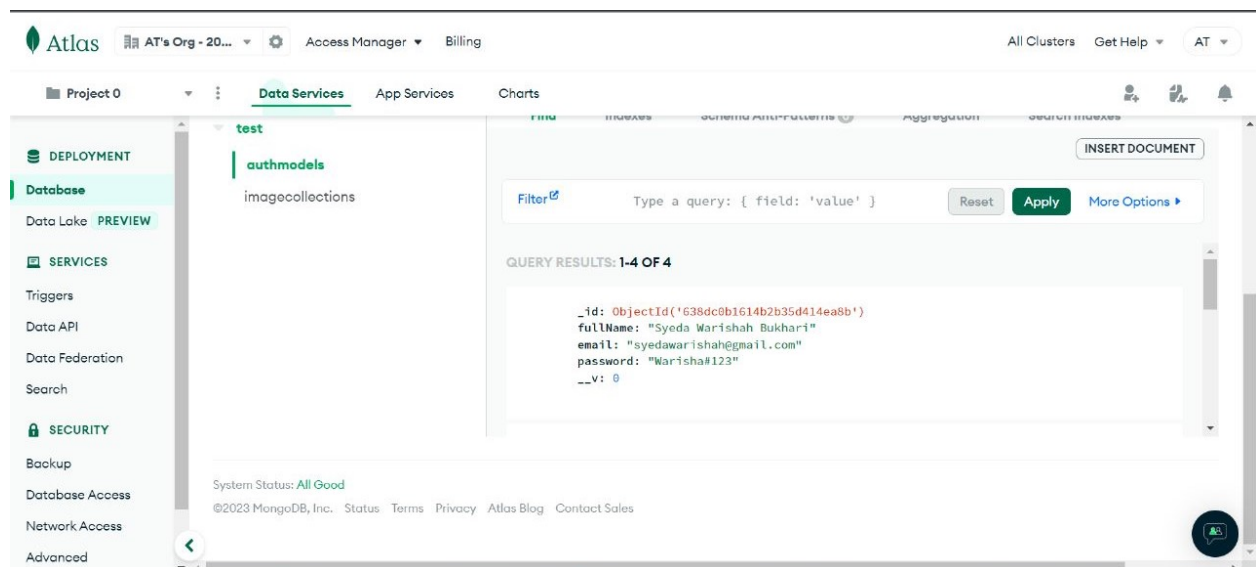


Figure 5.8: User Database

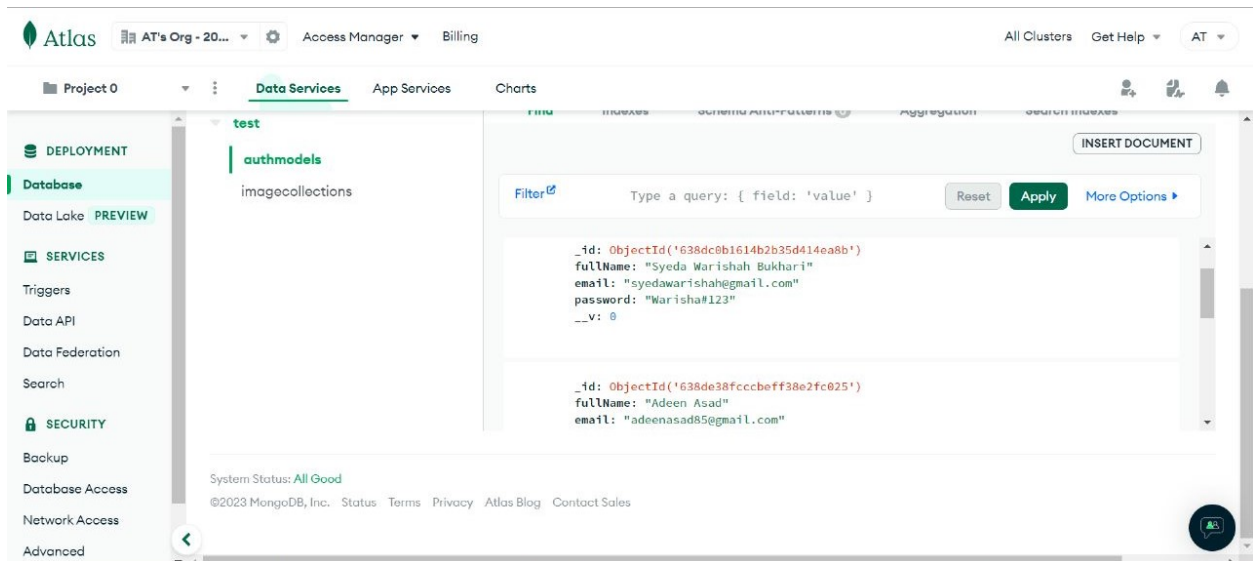


Figure 5.9: User Database

**5.2.4.2 MongoDB Image Collection Database** The "MongoDB Image Collection Database" view of our "ATIG" application is represented in figure 5.14 which is showing the end-user's saved collection:

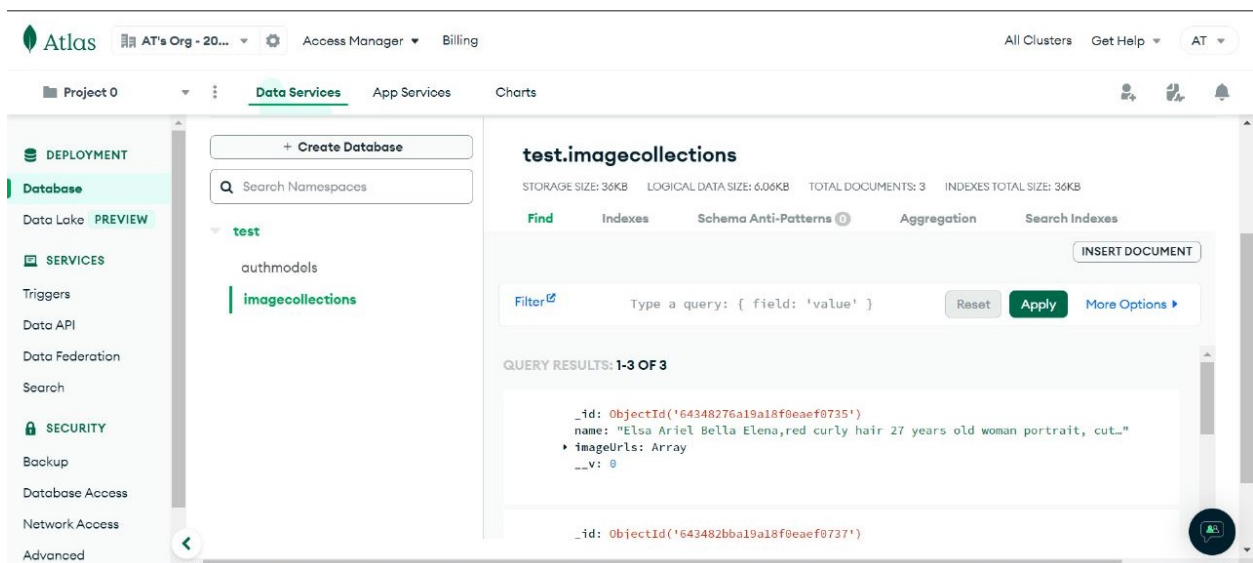


Figure 5.10: Image Collection Database

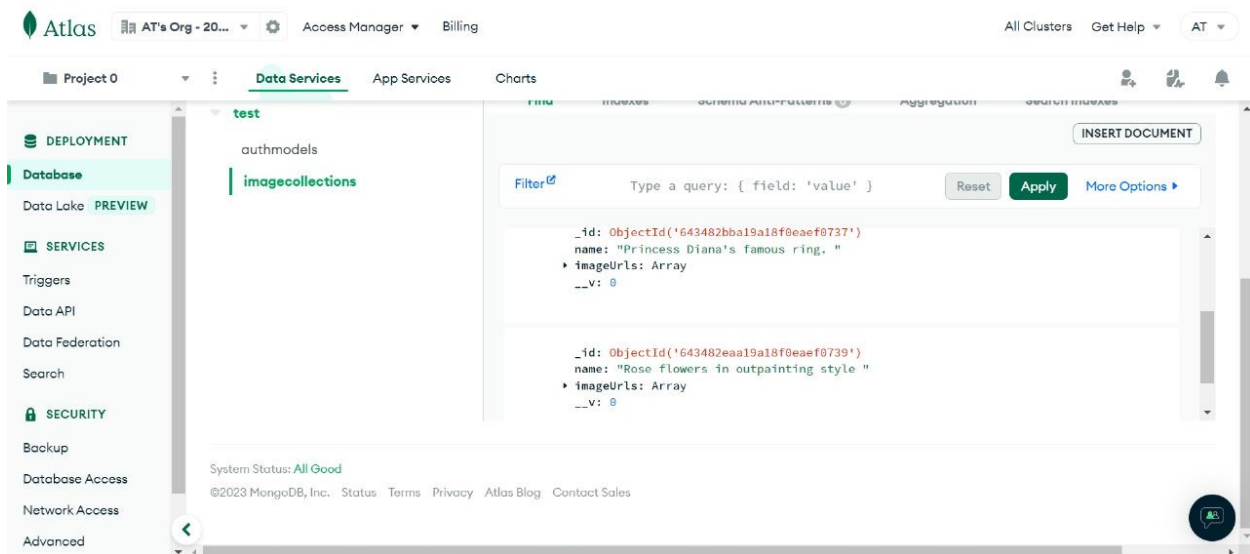


Figure 5.11: Image Collection Database

# Chapter 6

## SYSTEM INTERFACE AND PHYSICAL DESIGN

### 6.0.1 System User Interfaces

**6.0.1.1 Home Page Interface** The Figure 6.1 is representing the "Home" screen of our Website and Figure 6.2 shows the footer side of the home screen.

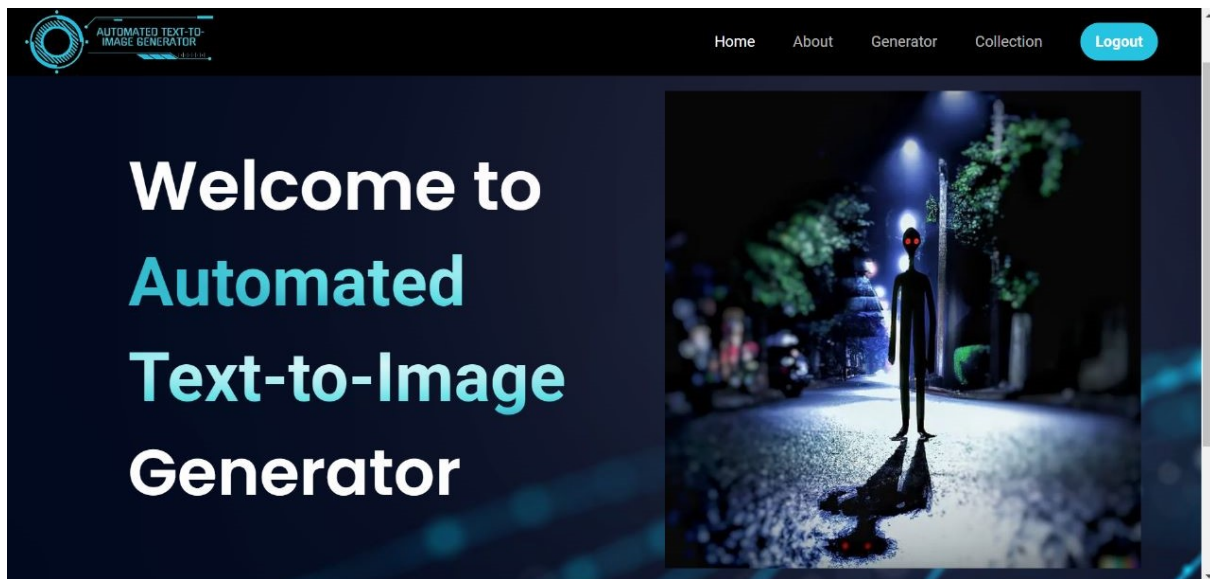


Figure 6.1: Home Page

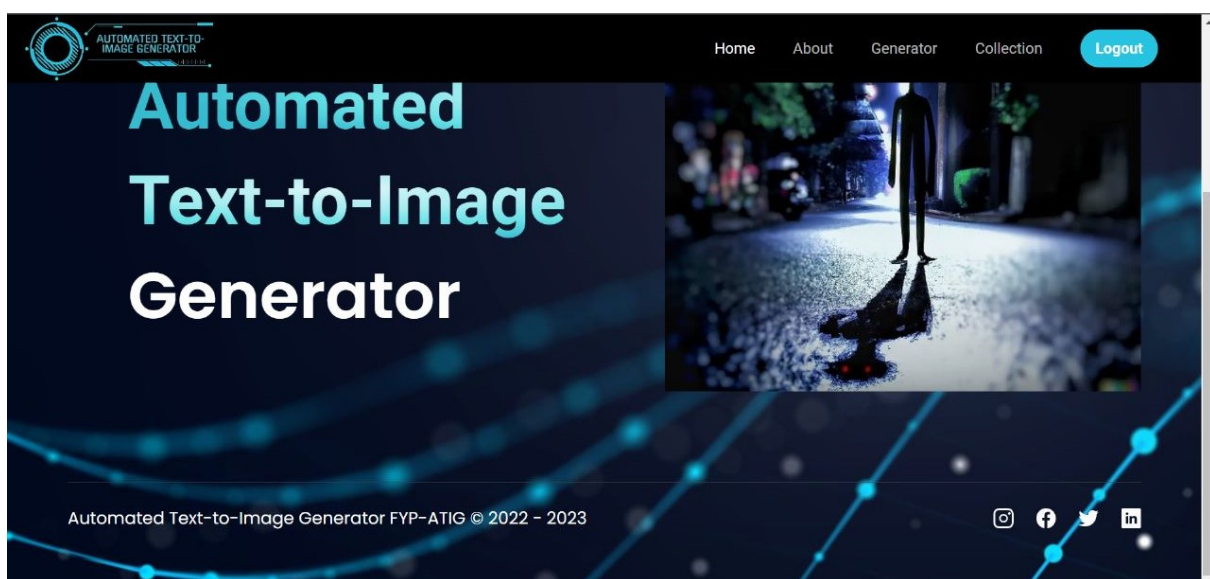


Figure 6.2: Home Page



**6.0.1.2 About Us Interface** The Figure 6.3 is representing the "About" screen of our Website and Figure 6.4 shows the footer side of the about screen.

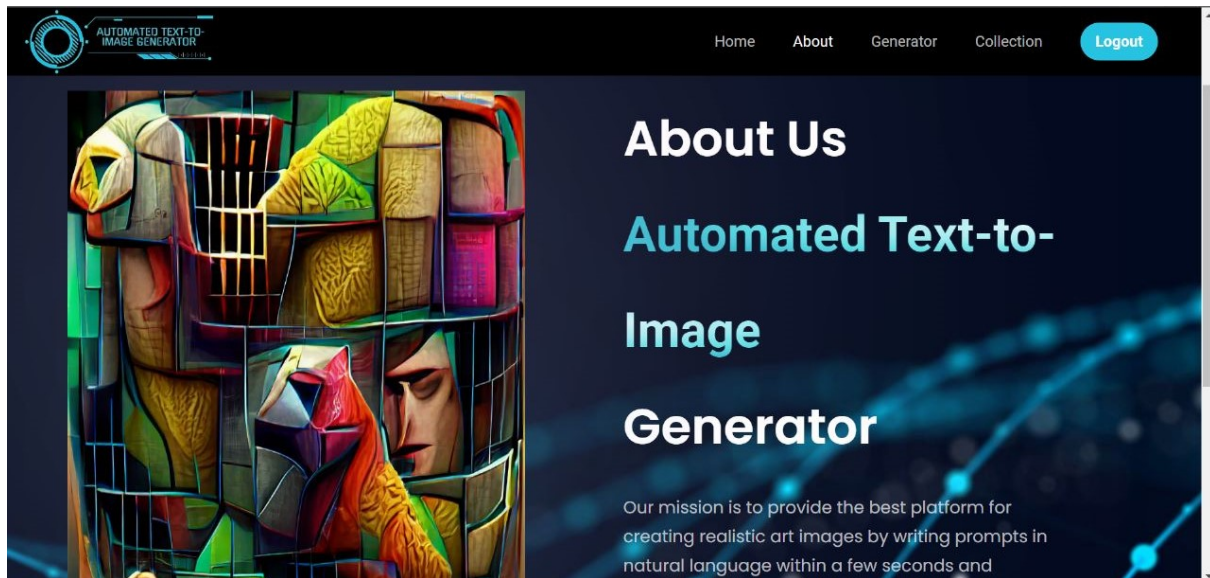


Figure 6.3: About Us

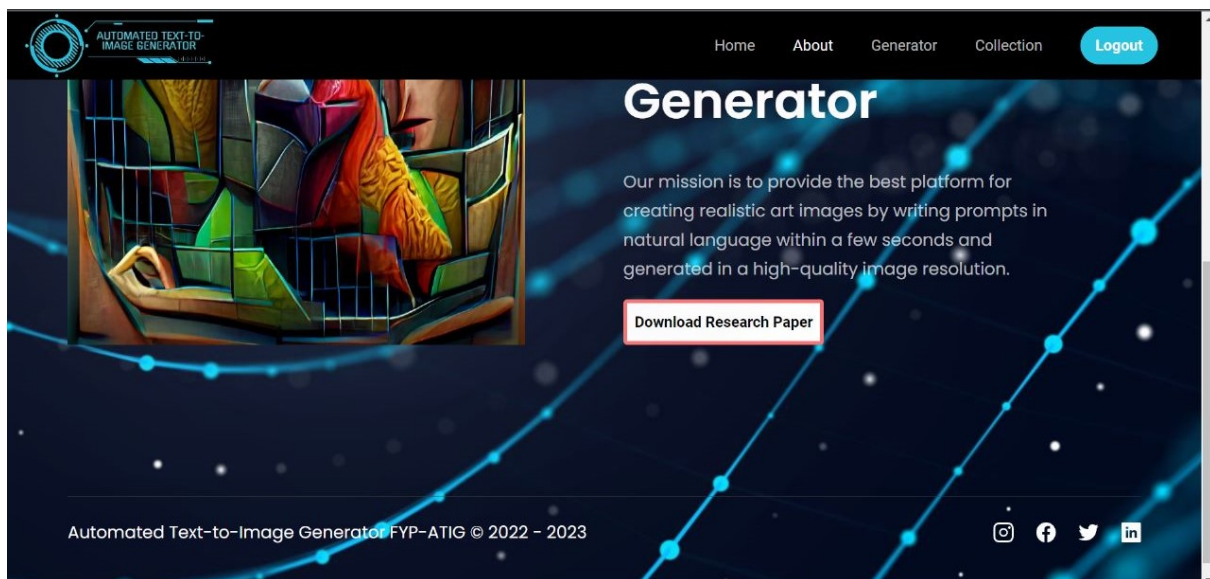
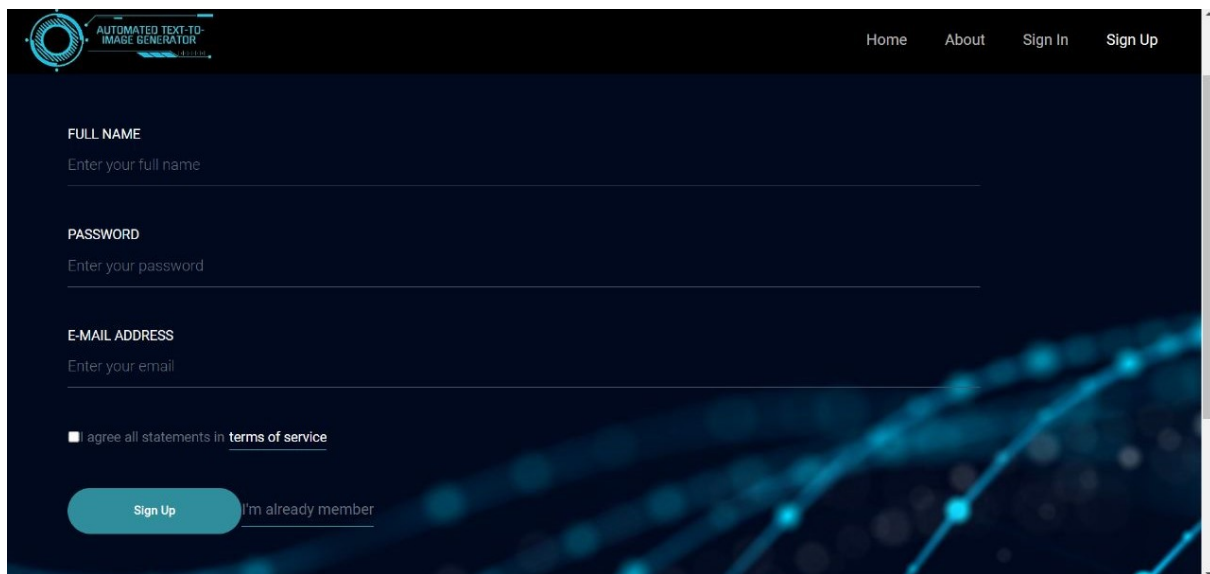


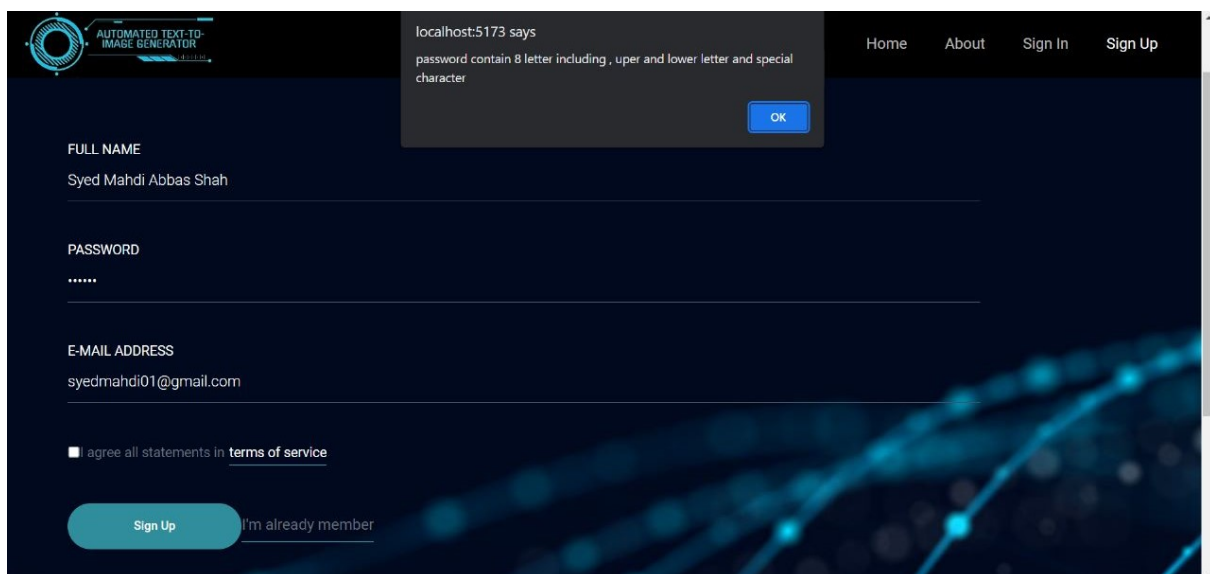
Figure 6.4: About Us

**6.0.1.3 Sign-up Interface** The Figure 6.5 is representing the "Sign-up" screen of our Website, figure 6.6 shows the functionality of sign-up form and figure 6.7 shows the save data successfully.



The image shows a web application's sign-up interface. At the top left is a logo for 'AUTOMATED TEXT-TO-IMAGE GENERATOR'. The top navigation bar includes links for 'Home', 'About', 'Sign In', and 'Sign Up'. The main form area has three input fields: 'FULL NAME' with the placeholder 'Enter your full name', 'PASSWORD' with the placeholder 'Enter your password', and 'E-MAIL ADDRESS' with the placeholder 'Enter your email'. Below these fields is a checkbox labeled 'I agree all statements in [terms of service](#)'. At the bottom of the form are two buttons: a teal 'Sign Up' button and a link 'I'm already member'.

Figure 6.5: Sign-up



This image shows the same sign-up form as Figure 6.5, but with sample data entered and a validation message. The 'FULL NAME' field contains 'Syed Mahdi Abbas Shah', the 'PASSWORD' field contains '\*\*\*\*\*', and the 'E-MAIL ADDRESS' field contains 'syedmahdi01@gmail.com'. The checkbox 'I agree all statements in [terms of service](#)' is checked. A modal message box is displayed at the top center, stating 'localhost:5173 says password contain 8 letter including , uper and lower letter and special character' with an 'OK' button. The 'Sign Up' button and 'I'm already member' link are still visible at the bottom.

Figure 6.6: Sign-up

localhost:5173 says  
Auth Model Saved in mongodb

Home About Sign In Sign Up

**FULL NAME**  
Syed Mahdi Abbas Shah

**PASSWORD**  
\*\*\*\*\*

**E-MAIL ADDRESS**  
syedmahdi01@gmail.com

☒ agree all statements in [terms of service](#)

**Sign Up** [I'm already member](#)

Figure 6.7: Sign-up

**6.0.1.4 Sign-in Interface** The Figure 6.8 is representing the "Sign-in" screen of our Website and figure 6.9 shows the functionality of sign-in form.

Home About Sign In Sign Up

**E-MAIL ADDRESS**  
Enter your email

**PASSWORD**  
Enter your password

**Sign In** [Create an account](#)

Figure 6.8: Sign-in



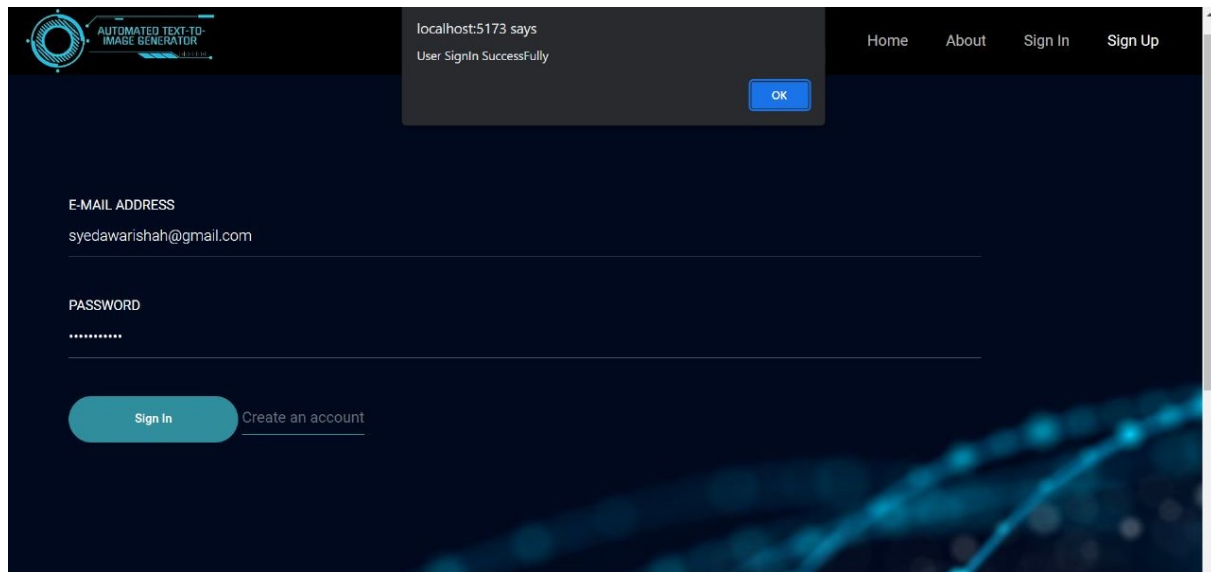


Figure 6.9: Sign-in

## 6.0.2 Main Module Interface

**6.0.2.1 Image Generator Interface** The Figure 6.10 is representing the "Generator" screen of our Website and Figure 6.11 shows the footer side of the page.

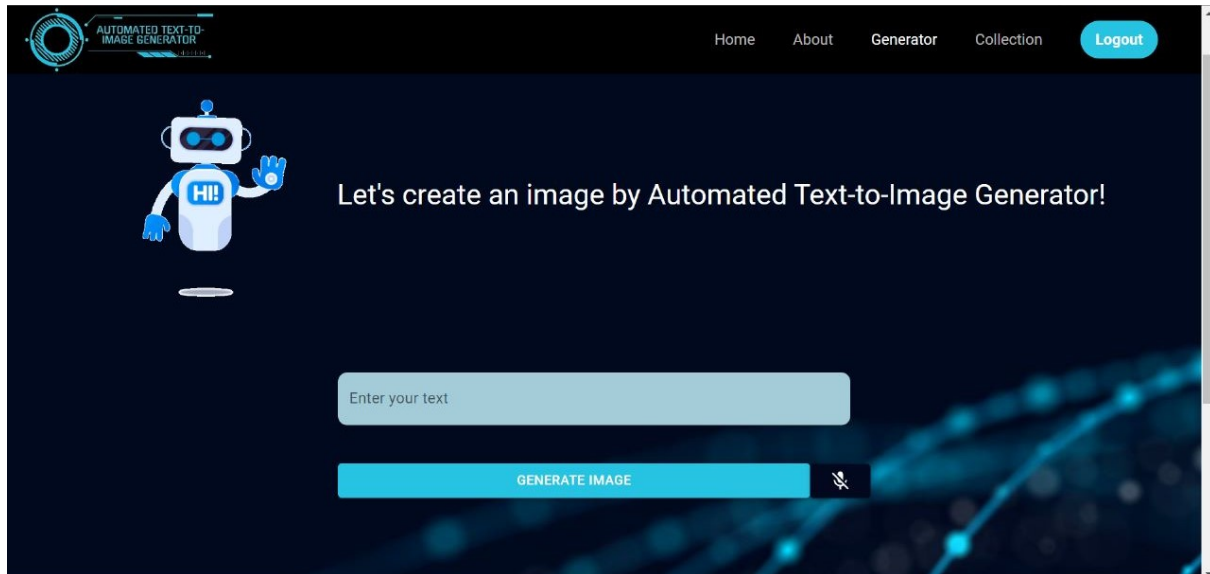


Figure 6.10: Generator

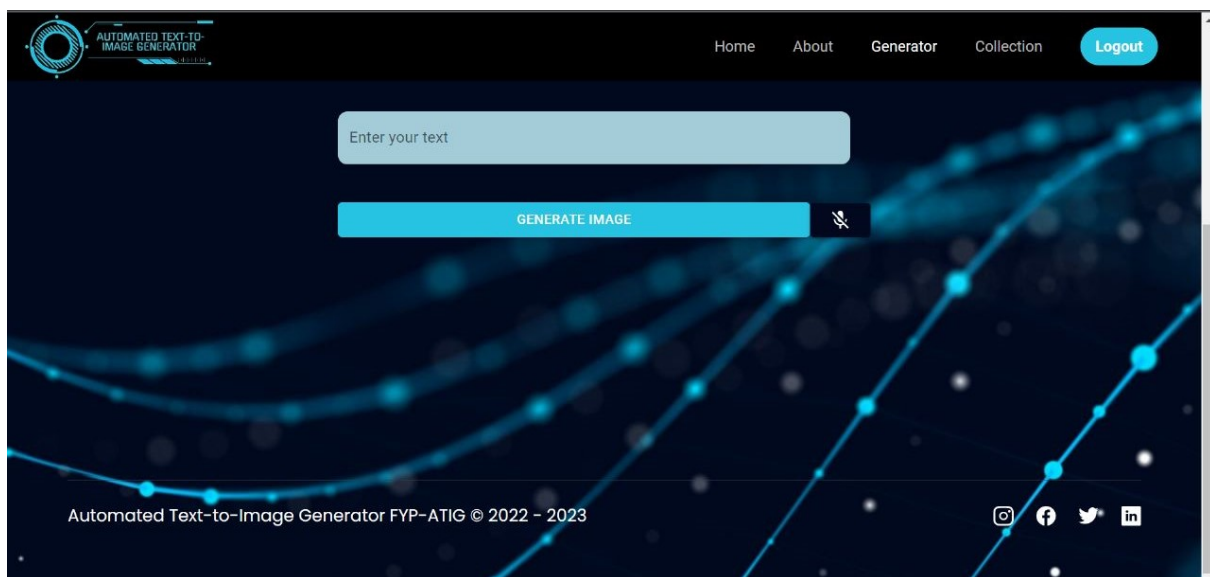


Figure 6.11: Generator

**6.0.2.2 Collection Page Interface** The Figure 6.12 is representing the "Collection" screen of our Website and Figure 6.13 shows the footer side of the page.

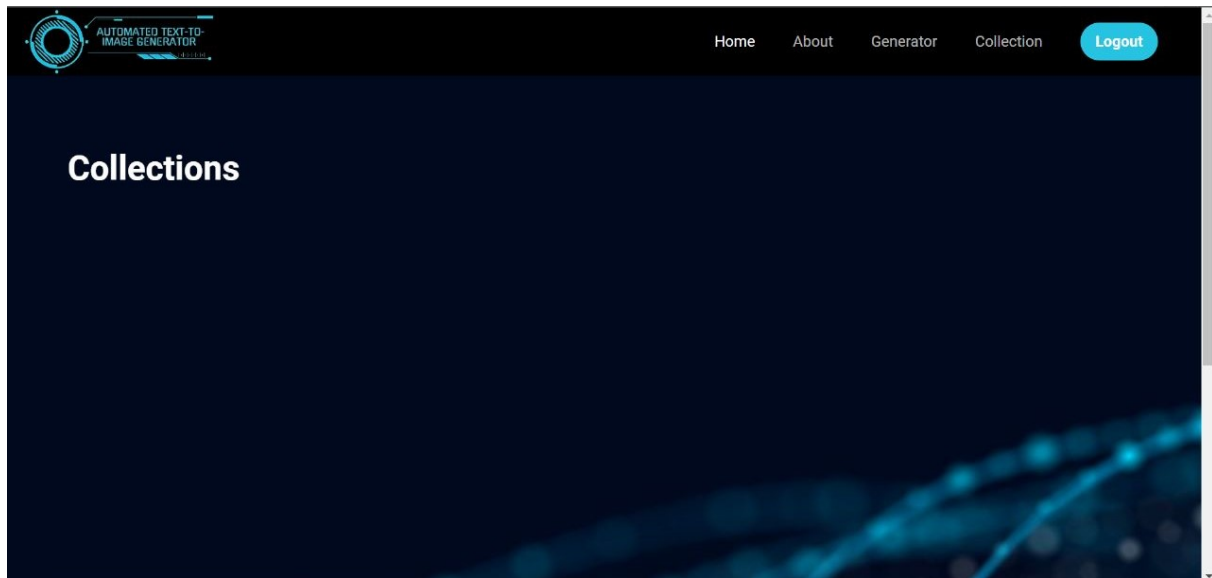


Figure 6.12: My Collection

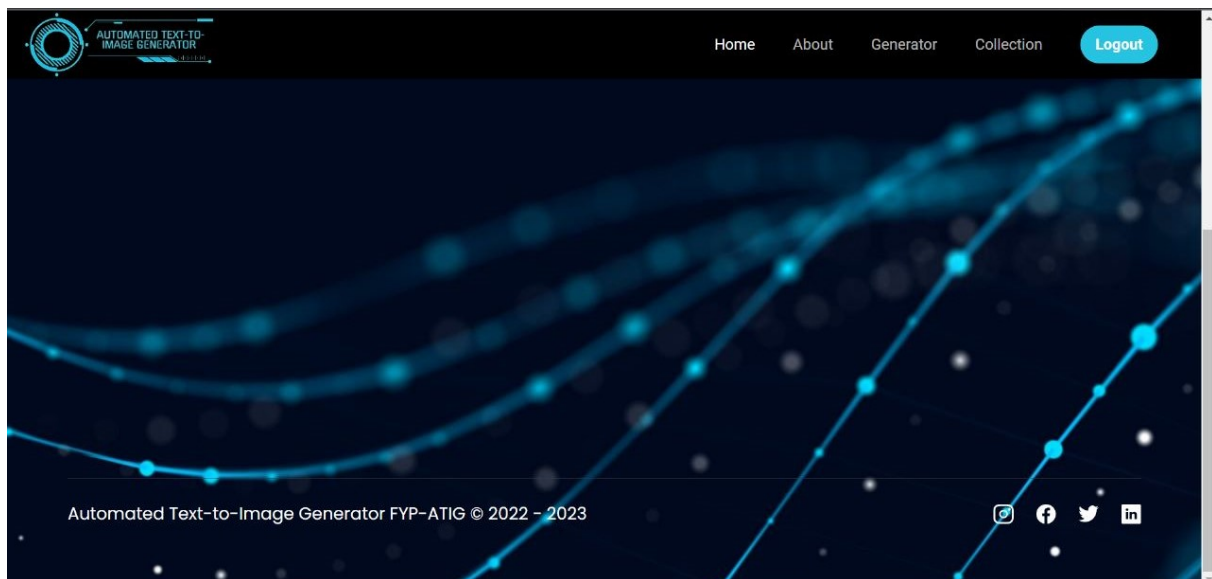


Figure 6.13: My Collection

### 6.0.3 Systems Working Interface

**6.0.3.1 Image Generator Working** The Figure 6.14 is representing the functionality of the "Generator" screen of our Website and Figure 6.15 shows the working of "Save In Collection" button.

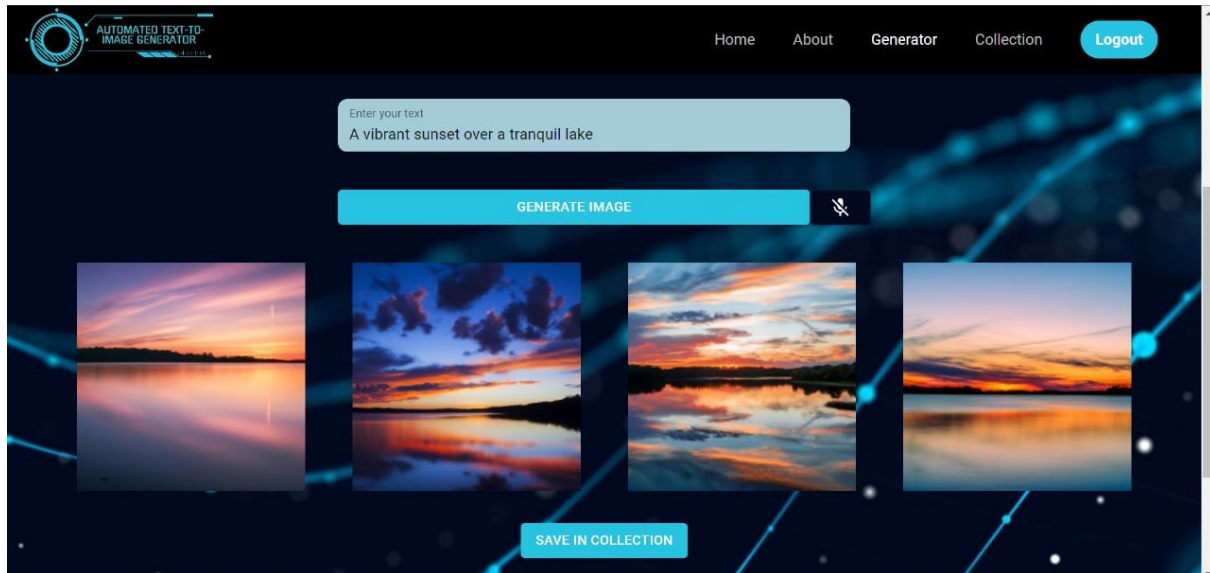


Figure 6.14: Generator

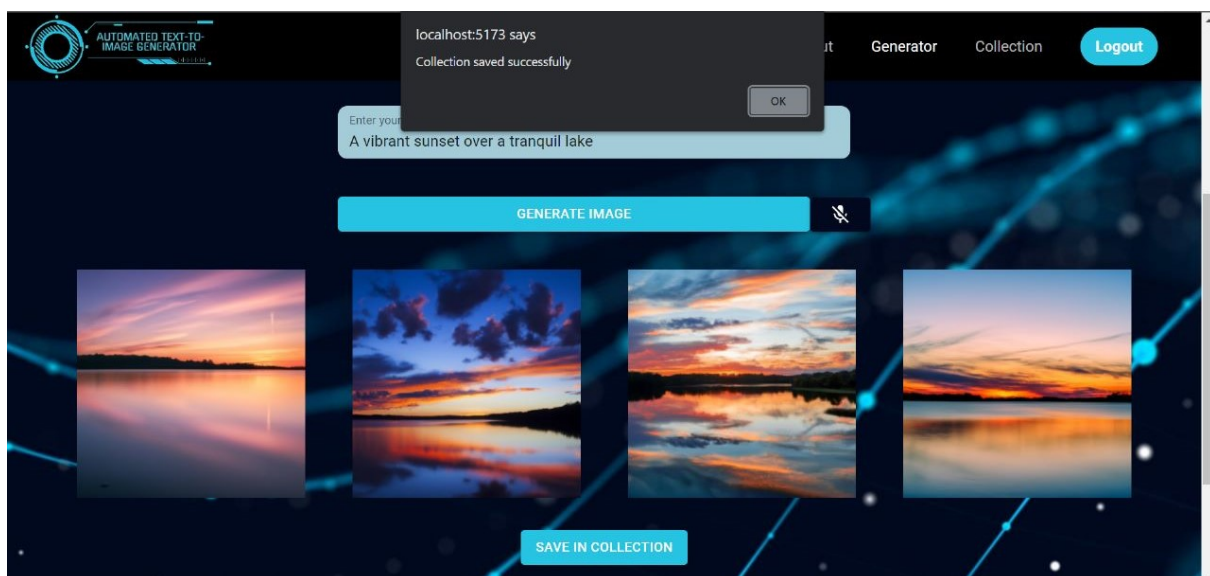


Figure 6.15: Generator

**6.0.3.2 Collection Working** The Figure 6.16 is representing the "Collection" screen of our Website.

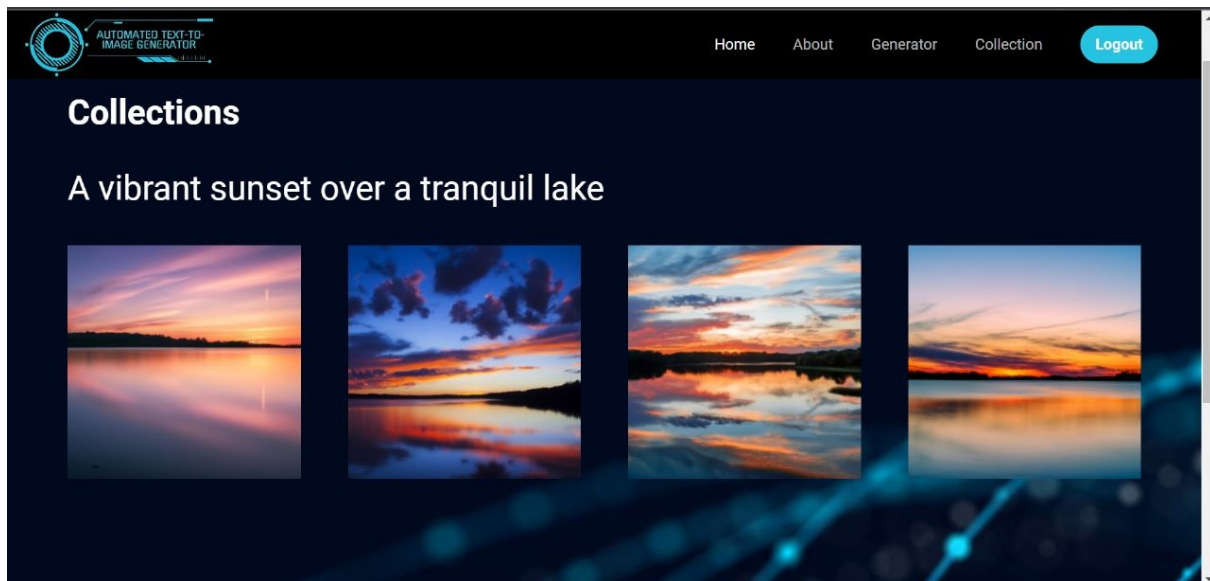


Figure 6.16: My Collection

**6.0.3.3 Image Generator Working With Long Text** The Figure 6.17 is representing the "Generator" screen working by large written text description of our Website.

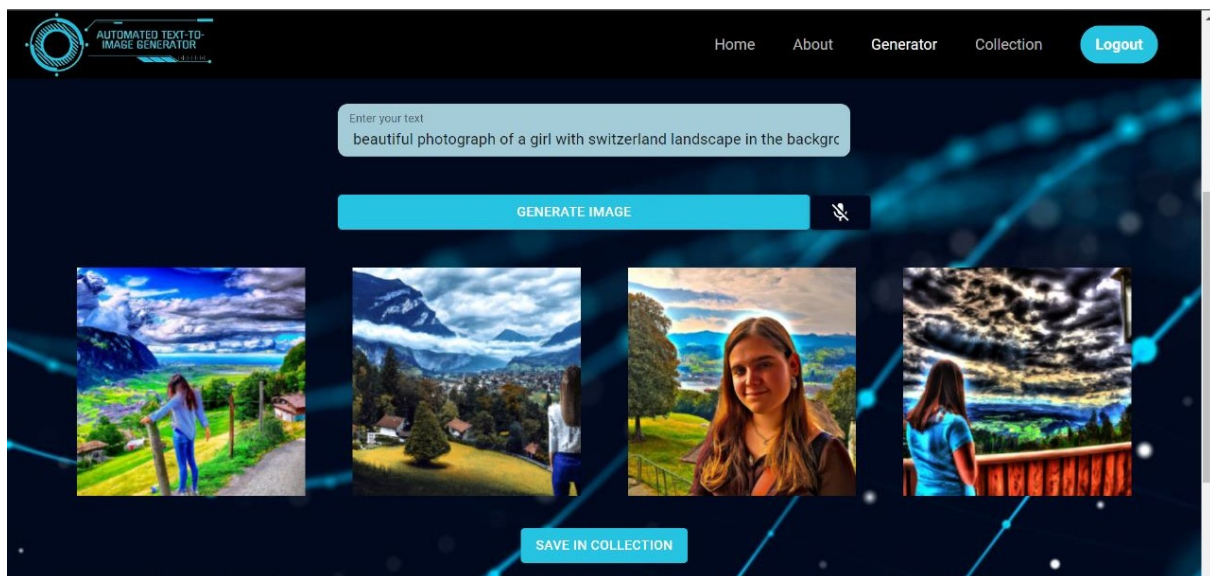


Figure 6.17: Generator



**6.0.3.4 Collection Working** The Figure 6.18 is representing the "Collection" screen of our Website.

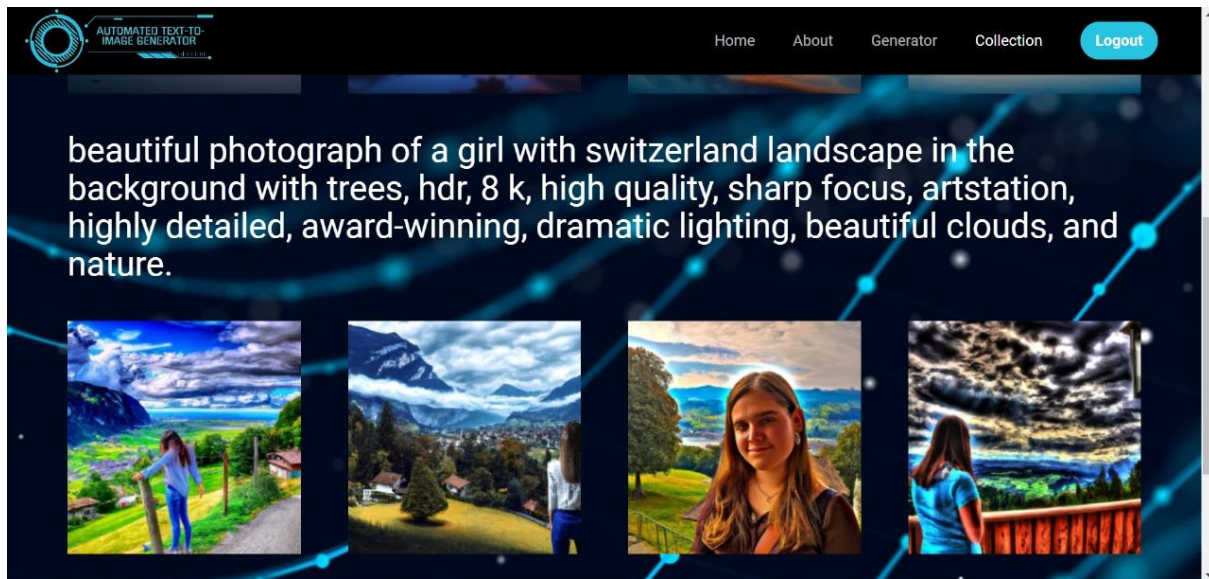


Figure 6.18: My Collection

## 6.0.4 Speech To Text Interface

**6.0.4.1 Image Generator Interface** The Figure 6.19 is representing the "Generator" screen in which images can generate by speech to text technology in outand Figure 6.20 shows the prompt message.

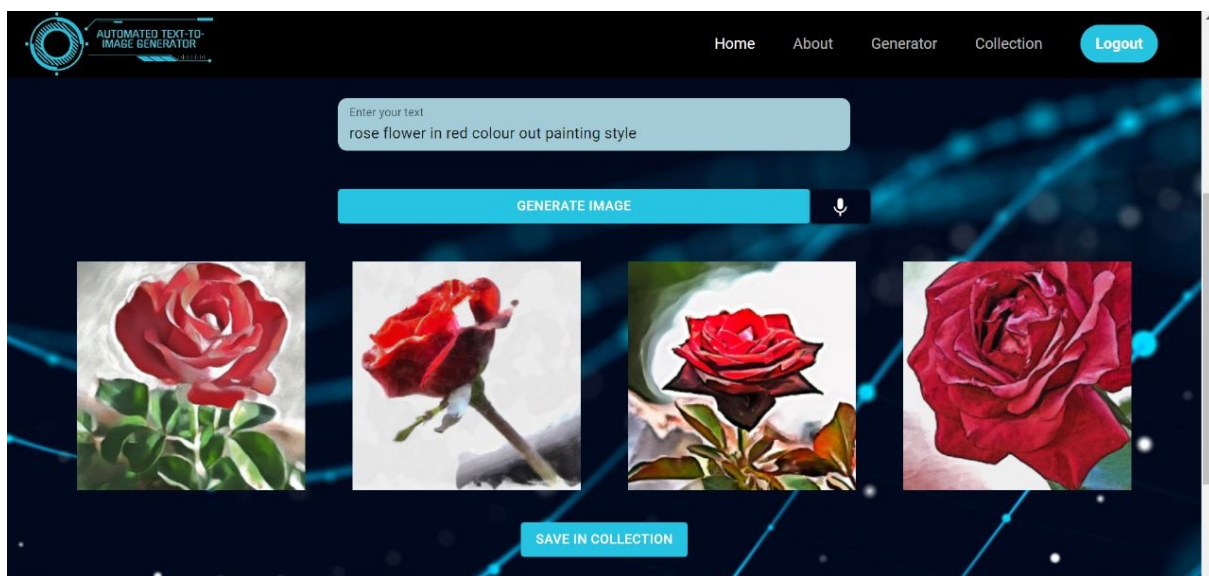


Figure 6.19: Mic Testing

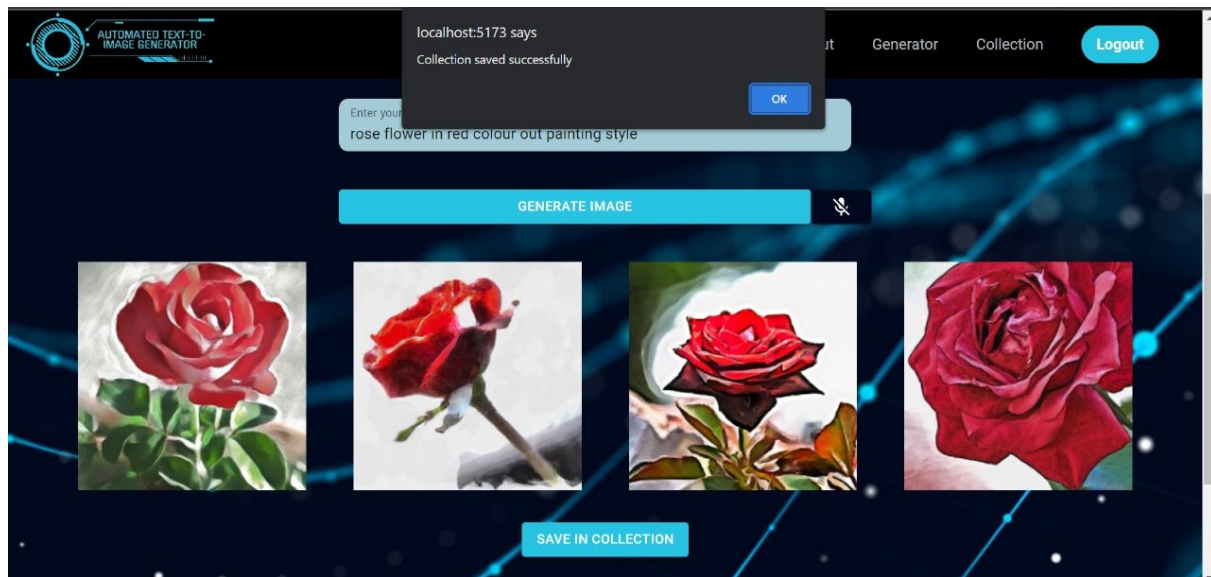


Figure 6.20: Save in Collection

**6.0.4.2 Saved Collection Interface** The Figure 6.21 is representing the "Collection" screen of our Website.

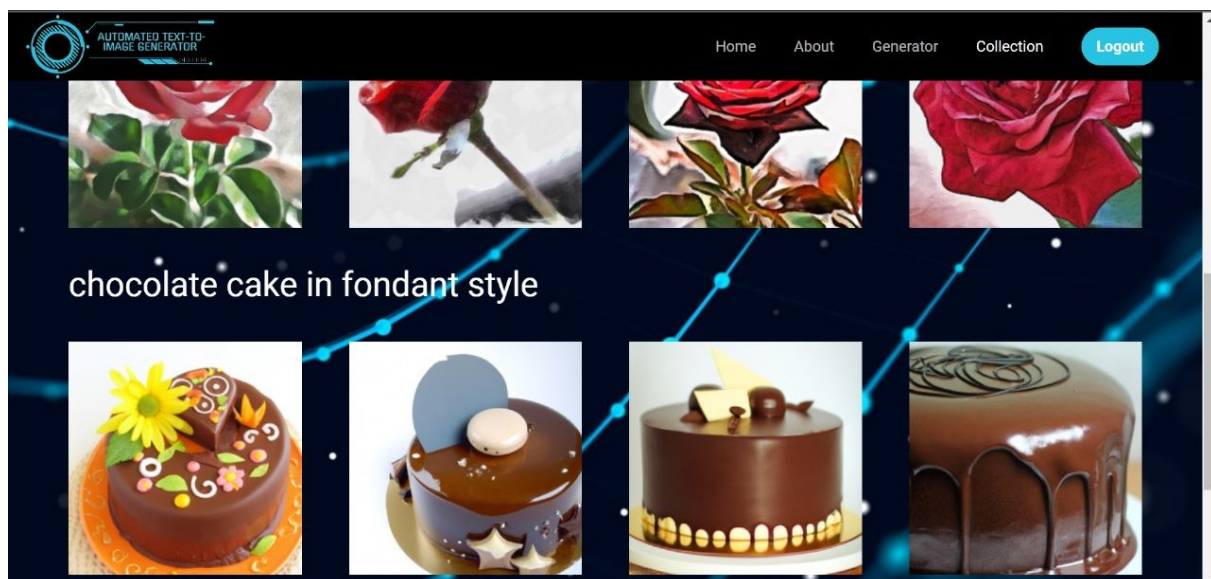


Figure 6.21: My Collection

# References

- [1] Open AI. (January, 2021). DALL.E 2 [Online]. Available: <https://openai.com/dall-e-2/>. Research Papers: <https://arxiv.org/pdf/2204.06125.pdf>
- [2] Dave Rogenmose. (November, 2008). Jasper [Online]. Available: <https://www.jasper.ai/free-trial?> Research Paper: <https://www.scijournal.org/articles/jasper-ai-review>
- [3] Angus Russell. (November, 2019) Nightcafe [Online]. Available: <https://creator.nightcafe.studio/create-nft-art?> Research Paper: <https://dataconomy.com/2022/10/nightcafe-ai-image-generator-creator-how/>
- [4] Boris Dayma. (June, 2022). Craiyon, formerly DALL-E mini [Online]. Available: <https://www.craiyon.com/>
- [5] Wombo Canadian. (October 13, 2021). Dream by Wombo [Online]. Available: <https://dream.ai/>
- [6] Mo Kahn. (August, 2021). starryai [Online]. Available: <https://starryai.com/>
- [7] Engineer Alexander Mordvintsev. (July, 2015). Deep Dream Generator [Online]. Available: <https://deepdreamgenerator.com/>
- [8] Kevin Baragona. (June 11, 2018). Deep AI [Online]. Available: <https://deepai.org/machine-learning-model/text2img>
- [9] Chengdu Hengtu. (2009). Fotor [Online]. Available: <https://www.fotor.com/features/ai-art-generator/>