

# **StoryGPT : An LLM-based learning tool for specially-abled children**

Submitted in partial fulfillment of the requirements of the  
degree

## **BACHELOR OF ENGINEERING IN COMPUTER ENGINEERING**

By

**Sarang Pavanaskar D12A/51**

**Akshat Mahajan D12A/40**

**Tanmay Maity D12A/42**

**Mohit Vaidya D12A/60**

Name of the Mentor

**Dr. Sujata Khedkar**



**Vivekanand Education Society's Institute of Technology,**

An Autonomous Institute affiliated to University of Mumbai  
**HAMC, Collector's Colony, Chembur,**

**Mumbai-400074**

**University of Mumbai (AY 2024-25)**

# **CERTIFICATE**

This is to certify that the Mini Project entitled “ **StoryGPT : An LLM-based learning tool for specially-abled children** ” is a bonafide work of **Sarang Pavanaskar (D12A-51), Askhat Mahajan (D12A-40), Tanmay Maity (D12A-42), Mohit Vaidya (D12A-60)** submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the degree of “**Bachelor of Engineering**” in “**Computer Engineering**” .

**(Dr. Sujata Khedkar)**

Mentor

**(Dr. Nupur Giri)**

Head of Department

**(Dr. J.M. Nair)**

Principal

# **Mini Project Approval**

This Mini Project entitled “ **StoryGPT : An LLM-based learning tool for specially-abled children** ” by **Sarang Pavanaskar (D12A-51), Askhat Mahajan (D12A-40), Tanmay Maity (D12A-42), Mohit Vaidya (D12A-60)** is approved for the degree of **Bachelor of Engineering in Computer Engineering**.

## **Examiners**

**1.....**  
(Internal Examiner Name & Sign)

**2.....**  
(External Examiner name & Sign)

Date:

Place:

# Contents

<b>Abstract</b>	<b>ii</b>
<b>Acknowledgments</b>	<b>ii</b>
<b>List of Figures</b>	<b>iii</b>
<b>List of Tables</b>	<b>iv</b>
<b>1 Introduction</b>	<b>1</b>
<b>1.1 Introduction</b>	
<b>1.2 Motivation</b>	
<b>1.3 Problem Statement &amp; Objectives</b>	
<b>1.4 Organization of the Report</b>	
<b>2 Literature Survey</b>	<b>5</b>
<b>2.1 Survey of Existing System</b>	
<b>2.2 Limitation Existing system or Research gap</b>	
<b>2.3 Mini Project Contribution</b>	
<b>3 Proposed System</b>	<b>9</b>
<b>3.1 Introduction</b>	
<b>3.2 Architectural Framework / Conceptual Design</b>	
<b>3.3 Algorithm and Process Design</b>	
<b>3.4 Methodology Applied</b>	
<b>3.5 Hardware &amp; Software Specifications</b>	
<b>3.6 Experiment and Results for Validation and Verification</b>	
<b>3.5 Result Analysis and Discussion</b>	
<b>3.6 Conclusion and Future work</b>	
<b>4 References</b>	<b>25</b>

# **Abstract**

This project aims to revolutionize how educational narratives are crafted and delivered to teach social norms and appropriate behaviors to specially-abled children. By leveraging advancements in artificial intelligence, particularly Large Language Models (LLMs) and generative AI, this project seeks to automate the creation of personalized stories. These stories, designed to address specific situations and challenges, will serve as valuable tools for parents, educators, and caregivers. By offering personalized content in various formats such as text, audio, images, and videos, the project aims to provide an engaging and accessible learning experience for children, fostering their understanding of social norms in a creative and meaningful way.

# **Acknowledgement**

We would like to sincerely thank everyone who helped us complete this project. First, we are very grateful to our mentor, Dr. Sujata Khedkar, for her constant support, guidance, and helpful advice throughout the project.

We also want to express our gratitude to our Head of Department, Dr. Nupur Giri, and Deputy Head of Department, Dr. Gresha Bhatia, for providing us with the opportunity to work on this project. We are thankful to our college, VESIT (Vivekanand Education Society's Institute of Technology), for providing us with a conducive environment for learning and innovation..

## List of Figures

Figures	Description
Fig 01	Block Diagram
Fig 02	Data Flow Diagram
Fig 03	Story Generation using Gemini model
Fig 04	Story Generation using Fine Tuned Model
Fig 05	Image prompt generation
Fig 06	Starter Screen
Fig 07	Home Page
Fig 08	Story Output Screen
Fig 09	Audio Output Screen
Fig 10	Video Output Screen
Fig 11	PDF Output Screen
Fig 12	generate_story function
Fig 13	genearte_image_prompt function
Fig 14	generate_content function
Fig 15	split story into parts
Fig 16	generate images using image prompts
Fig 17	audio generation - 1
Fig 18	audio generation - 2
Fig 19	folder structure

## List of Tables

Tables	Description
Table 01	Literature Survey
Table 02	Limitations in the existing systems
Table 03	Experiments & results

# **1. Introduction**

This chapter introduces the StoryGPT project, focusing on the need for personalized tools to help specially-abled children learn social skills. It highlights the limitations of traditional methods and proposes an AI-driven system using Large Language Models (LLMs) and generative AI to create engaging, multimedia-rich stories. The chapter outlines the problem of manually creating social stories and presents the project's objectives, which include automating the story creation process and offering accessible, tailored content. It also provides an overview of the report's structure, detailing the key sections on research, system design, implementation, and future work.

## **1.1 Introduction**

The ability to learn and understand social norms is a critical aspect of a child's development, yet for specially-abled children, acquiring these skills can be particularly challenging. Traditional educational methods often fall short in providing the tailored support and engagement these children need to thrive in social situations. As technology advances, there is a growing opportunity to transform the way we approach teaching social behaviors. This project aims to harness the power of Large Language Models (LLMs) and generative AI technologies to create personalized stories that are not only informative but also engaging for specially-abled children. These stories will be designed to address real-life scenarios and challenges, offering a more accessible and interactive learning experience in formats such as text, audio, images, and videos. By automating the story creation process and personalizing the content to each child's unique needs, this project seeks to provide a valuable resource for parents, educators, and caregivers, ultimately enhancing the educational support available for children with special needs.

## **1.2 Motivation**

The motivation behind this project comes from the need to provide specially-abled children with more personalized, accessible, and effective tools for learning social norms and appropriate behaviors. Traditional teaching methods often lack the adaptability required to meet the unique learning needs of these children, leaving gaps in their ability to navigate social situations confidently. With the growing capabilities of artificial intelligence, particularly Large Language Models (LLMs) and generative AI, there is now an opportunity to create highly customized educational content that can engage children in ways that conventional approaches cannot. Personalized stories that incorporate relevant situations, challenges, and characters can help children better understand and practice social behaviors in a safe, supportive environment. By delivering these stories in various formats—text, audio, images, and videos—this project aims to make learning more immersive, enjoyable, and tailored to each child's abilities. This initiative not only seeks to support children's social development but also to empower parents, educators, and caregivers with a powerful, innovative tool to enhance learning outcomes.

## **1.3 Problem Statement**

Manually creating Social Stories for children with special needs is a labor-intensive and challenging process that demands significant time and effort from parents and educators. Traditional Social Stories often fall short in providing the necessary customization to address specific situations, behaviors, and learning needs, which limits their effectiveness. Additionally, relying solely on text-based content fails to capture the attention of children with varying learning preferences, reducing the overall impact of these stories. The absence of integrated multimedia elements such as audio, images, and videos further diminishes the engagement and effectiveness in teaching social norms and behaviors. Moreover, parents and educators face difficulties in regularly creating and updating these stories, making it hard to consistently meet the evolving needs of the children. This problem underscores the need for a more efficient, customizable, and engaging approach to creating Social Stories that can adapt to the unique requirements of each child.

## **Objectives**

- To develop an AI-driven system that automates the creation of personalized educational stories tailored to the unique needs of specially-abled children.
- To produce these stories in multiple formats—text, audio, images, and videos—to cater to different learning preferences and enhance accessibility.
- To address specific social challenges and situations faced by specially-abled children through the narrative content of the stories.
- To provide parents, educators, and caregivers with an effective tool to support the social development of specially-abled children.
- To innovate educational technology by integrating Large Language Models (LLMs) and generative AI for personalized learning solutions.

## **1.4 Organization of the Report**

This report is structured to provide a comprehensive overview of the StoryGPT project:

**Chapter 1:** Provides a detailed introduction to the project, including the motivation, problem statement, and objectives, establishing the context for the rest of the report.

**Chapter 2:** Conducts a thorough literature survey, reviewing existing systems, identifying their limitations, and highlighting the research gaps that StoryGPT addresses.

**Chapter 3:** Describes the proposed system in detail, covering the architectural framework, algorithm design, process design, and methodologies applied, along with hardware and software specifications.

**Chapter 4:** Focuses on the implementation and testing of the StoryGPT system, presenting experimental results, validation, and verification processes.

**Chapter 5:** Concludes the report, summarizing the key findings and offering suggestions for future work to further enhance the system.

## 2. Literature Review

This chapter provides a comprehensive literature review, examining existing systems and methodologies related to automated storytelling, generative AI, and text-to-image synthesis. The chapter is divided into three sections: a survey of existing systems, identifying the methodologies, advantages, and limitations of recent projects; an analysis of the research gaps and limitations within current systems; and the unique contributions of this mini project, StoryGPT, which aims to address several identified gaps by offering a more personalized, multimedia-rich, and automated storytelling experience.

### 2.1 Survey of Existing System

Year	Title	Objective	Methodology	Advantages	Limitations
2024	Sarid: Arabic Storyteller Using a Fine-Tuned LLM and Text-to-Image Generation	To develop an Arabic story generation model using a fine-tuned LLM and text-to-image generator	Collected a dataset of 527 Arabic children's stories through web scraping, Preprocessed and fine-tuned the Davinci model using the dataset, Generated visual elements using Midjourney, Integrated user inputs into the story generation process through a website	Gives stories based on user inputs such as theme, tone, and character, Produces high-quality images corresponding to the stories, Ensures character consistency throughout the story.	Requires careful prompt engineering to maintain consistency in visuals., Challenges in character consistency without the use of specific seeds, Dependence on the quality of the fine-tuned model and dataset.
2024	A survey of Generative AI Applications	To survey and categorize over 350 generative AI applications, providing insights into state-of-the-art technologies across various domains.	Organized into 15 categories such as text, images, video, and more, detailing current technologies and tools used.	Extensive coverage; organized framework; aids in identifying current technologies and trends.	Rapid tech evolution might make some details outdated; broad scope may be overwhelming for beginners.
2022	Deep Learning Methodology Converts Text to Image	To explore the application of Generative Adversarial Networks (GANs) for text-to-image synthesis, focusing on the challenges of translating textual descriptions into visual content and the utilization of deep learning techniques to enhance image generation	Utilized GANs with TensorFlow, NLTK, and PySimpleGUI; employed GAN-CLS algorithm; trained on Oxford-102 flower dataset with generator and discriminator models	Effective text-to-image synthesis; improved user interaction with GUI; deep learning efficiency.	High computational cost; time-consuming training; challenges in text interpretation and coherence.

2024	LLM-DetectAIVe: a Tool for Fine-Grained Machine-Generated Text Detection	To classify texts into four categories: human-written, machine-generated, machine-humanized, and human-polished.	<ul style="list-style-type: none"> <li>- Creation of a new dataset combining human-written and various machine-generated texts.</li> <li>- Training multiple detectors using models like RoBERTa, DeBERTa, and DistilBERT.</li> <li>- Evaluation using domain-specific and universal detectors.</li> </ul>	<ul style="list-style-type: none"> <li>- Offers fine-grained detection, improving the accuracy and relevance of detecting AI intervention in text.</li> <li>- Incorporates a multi-way classification system, offering more nuanced detection compared to binary classification.</li> </ul>	<ul style="list-style-type: none"> <li>- Domain-specific detectors require users to select the domain before detection.</li> <li>- May face challenges in cross-domain scenarios due to varying text features.</li> </ul>
2024	HEART-felt Narratives: Tracing Empathy and Narrative Style in Personal Stories with LLMs	To empirically examine the relationship between narrative style and empathy using LLMs and large-scale crowdsourcing studies.	The authors introduced a novel taxonomy (HEART) and used LLMs to extract narrative elements, followed by a large-scale crowdsourcing study with 2,624 participants.	The methodology allows for the analysis of complex narrative elements and their impact on empathy using advanced AI techniques.	The study relies heavily on LLMs, which may have limitations in fully capturing the nuances of human narrative empathy.
2024	How Well Can Vision Language Models See Image Details?	Investigate how well Vision-Language Models (VLMs) perceive image details beyond the semantic level using a pixel value prediction task (PVP).	Fine-tune VLMs on the PVP task by adapting the connection module, language model, and visual encoder, then evaluate performance on image-language tasks.	Significant improvement in detailed image perception tasks, with notable boosts in referring image segmentation and video game decision-making performance.	Existing VLMs struggle with precise pixel value prediction when only fine-tuning the connection module and language model.
2023	Muse: Text-To-Image Generation via Masked Generative Transformers	To develop Muse, a more efficient text-to-image Transformer model that achieves state-of-the-art (SOTA) image generation performance.	Muse is trained on a masked modeling task using discrete image tokens and text embeddings from a pre-trained large language model, enabling efficient parallel decoding and fine-grained language understanding.	Muse outperforms diffusion and autoregressive models in efficiency and image quality, supports advanced image editing tasks without fine-tuning, and achieves SOTA results on benchmarks like CC3M and COCO.	Performance and efficiency gains may vary across different datasets, and applicability beyond tested benchmarks needs exploration.

Table no. 01 : Literature Survey

## 2.2 Limitation in Existing system or Research gap

Aspect	Limitation/Research Gap in Existing System
Story Customization	Most existing systems lack the ability to personalize social stories based on age, comprehension level, or specific needs of the child, making stories less effective.
Multimedia Integration	Current tools typically focus on text-based stories and do not integrate multimedia elements like videos, images, and audio in a seamless, user-friendly way.
Cultural and Linguistic Adaptability	Current systems lack the ability to adapt social stories for diverse cultural backgrounds and multiple languages, limiting their global applicability.
Automation	The creation of social stories is largely a manual, time-consuming process involving specialists like psychologists, with minimal automation for content generation.
Real-time Feedback	Existing systems do not offer mechanisms for gathering real-time feedback from parents or educators to improve or adapt stories dynamically.
Dataset Availability	Publicly available datasets of social stories are limited, and there is no organized corpus to systematically train machine learning models for personalized story generation.
Story Categorization	Current solutions do not categorize stories by themes (e.g., hospital visits, school behavior) or emotional situations, limiting their applicability to diverse scenarios.
Cross-format Consistency	Systems lack consistency when generating stories across formats (text, video, audio), resulting in varying levels of quality and cohesion between story components.
User Interaction and Usability	Existing systems often have complex interfaces, making it difficult for non-technical users (parents, teachers) to create and adapt stories effectively.
Scalability	Tools for social stories generation are not scalable to handle diverse user bases, particularly when creating personalized, multimedia-rich stories for various use cases.

Table no. 02 : Limitations in the existing systems

## 2.3 Mini Project Contribution

This project contributes to the domain of automated storytelling by addressing the limitations found in existing systems:

**Automated Story Generation:** StoryGPT introduces an AI-based system that dynamically generates personalized stories based on user inputs, eliminating the need for manual story creation.

**Image Integration:** Unlike existing systems, StoryGPT seamlessly integrates AI-generated images into the story, ensuring that visuals are relevant to the narrative and enhance the overall storytelling experience.

**Streamlined User Experience:** The project utilizes Streamlit to provide an intuitive, user-friendly interface, allowing users to easily navigate between input, generation, and output pages without complex steps.

**Dynamic Story Segmentation:** StoryGPT automatically splits the generated stories into meaningful segments, inserting appropriate images between sections to create a more engaging and structured narrative flow.

**Customization and Interactivity:** The system allows for a high degree of customization, offering users the flexibility to tailor stories to their preferences and interact with the generated content seamlessly.

### **3. Proposed System**

This chapter provides a detailed explanation of the system's architecture, algorithm, and design process. The proposed solution leverages advanced Large Language Models (LLMs) and generative AI technologies to generate personalized social stories, incorporating multimedia elements like text, audio, images, and video. The system enables caregivers to input specific behavioral challenges and preferences, and in response, creates tailored stories that are both engaging and accessible. This chapter also covers the technical aspects of the project, including hardware and software specifications, experiments conducted, and results achieved, with a focus on refining the system for improved efficiency and accuracy. Lastly, the chapter concludes with a discussion on future work, emphasizing scalability, multilingual support, and collaboration with educators to maximize the system's impact.

#### **3.1 Proposed solution**

The proposed idea is to build a system that uses Large Language Models (LLMs) and generative AI to create personalized stories for specially-abled children. Parents, teachers, or caregivers can input the child's needs, likes, and challenges into the system. Based on this, the AI will generate customized stories in different formats like text, audio, images, and videos. These stories will be easy to understand, matched to the child's learning level, and designed to help them learn good social behaviors through fun and engaging storytelling. Interactive elements will also be added to keep the child involved.

The system will allow users to create stories for different real-life situations, helping children practice how to behave in various scenarios. It will also adapt over time by using feedback from the child's responses to make the stories even better. The aim is to give caregivers a strong tool to support the child's social development. In the future, the solution can be expanded into apps or educational platforms to reach more children and make personalized AI learning available to a wider audience.

### 3.2 Architectural Framework / Conceptual Design

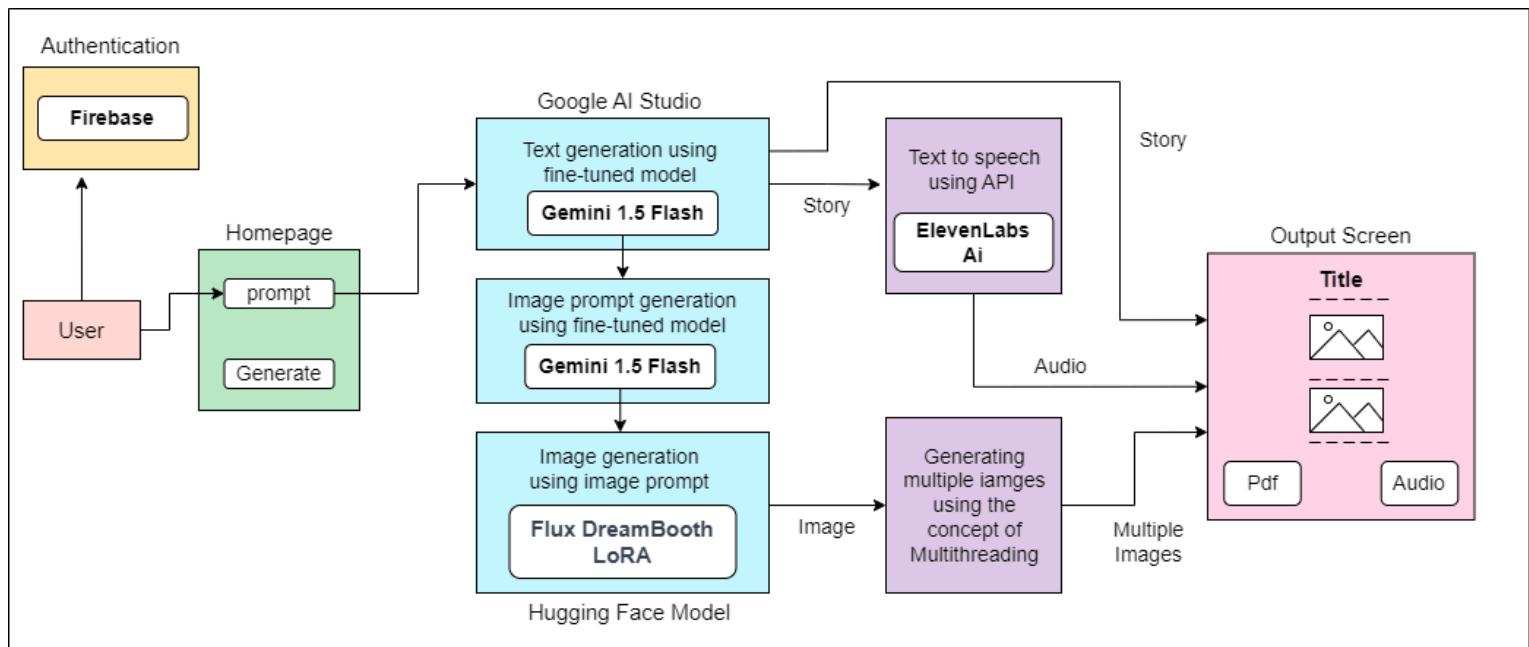


Fig. no. 01 : Block Diagram

### 3.3 Algorithm and Process Design

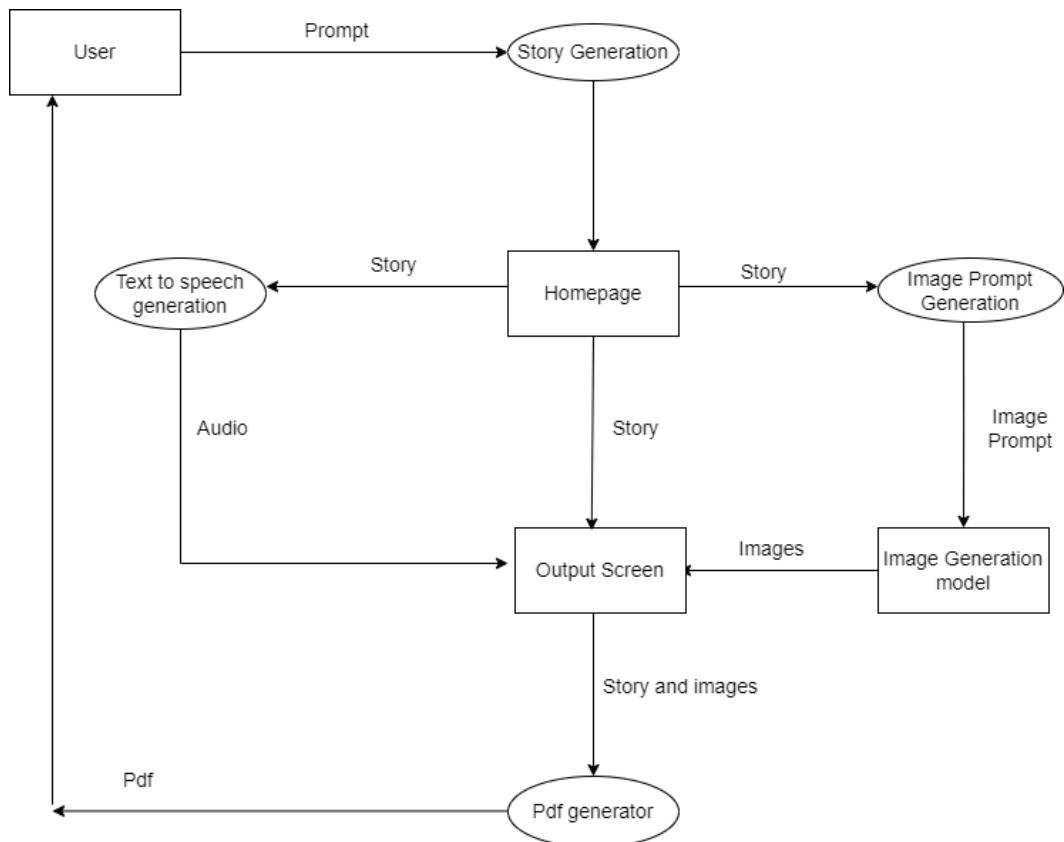


Fig. no. 02 : Data Flow Diagram

## **3.4 Methodology Applied**

**Problem Identification and Research :** The first step involved conducting extensive research to identify the challenges faced by parents, educators, and specially-abled children in traditional learning environments. This research focused on understanding the limitations of current educational tools in teaching social norms and behaviors, particularly the lack of personalized content that can engage children with different learning abilities.

**Data Collection and Preparation :** A curated dataset of social stories was developed, focusing on real-world scenarios that specially-abled children often encounter. This dataset forms the foundation for both the story generation and image prompt creation processes, ensuring the relevance and effectiveness of the content generated by the system.

### **AI Model Development:**

**Gemini Model for Story Generation :** The Gemini model was fine-tuned using the curated dataset to generate personalized social stories based on user input. This model enables the generation of narratives that address specific behavioral challenges and learning needs, ensuring each story is tailored to the child's context.

**Gemini Model for Image Prompt Generation:** A second instance of the Gemini model was fine-tuned to generate detailed image prompts from the curated dataset. These prompts are designed to be forwarded to the image generation model, ensuring that the images created are story-specific and visually aligned with the narrative.

#### **1. Image Generation:**

Flux Dream Booth LoRA: The Flux Dream Booth LoRA model was employed for generating personalized, high-quality images based on the prompts provided by the Gemini model. These images enhance the visual storytelling experience for children.

2. **Multithreading for Multiple Image Generation:** To improve efficiency and handle the simultaneous creation of multiple images, multithreading techniques were implemented. This approach ensures faster image generation, allowing multiple story-specific images to be processed concurrently.
3. **Audio Generation:** Using the Eleven Labs API, the generated stories were converted into audio format, providing an additional modality for children who

benefit from auditory learning. This step ensures that the stories are accessible to a wider range of learning preferences, offering a more engaging experience through voice narration.

4. **PDF Output Generation:** Python libraries were used to provide a downloadable PDF output of the generated story and images. This feature allows caregivers and educators to access the stories in a traditional, printable format, which can be shared or used offline as needed.
5. **Video Integration:** A system will be developed to combine the generated stories, images, and audio into a cohesive video format. This integrated approach creates immersive, multimedia stories that can effectively capture and maintain the child's attention, offering a richer learning experience.

### **3.5 Hardware & Software Specifications**

High-end computer with the following configuration :

- Processor: Intel Core i5 or i7 (7th Generation or later)
- RAM: 8GB or more
- Storage: 256GB or higher
- Graphics card: Dedicated GPU with at least 4GB VRAM (for multimedia processing) or
- Cloud GPU like Google Colab

Software Techstack :

- Gemini 1.5 Flash (fine-tuned) for text and image prompt generation
- Flux.1- Schnell model for image generation
- Frontend: Streamlit python library
- Authentication & Database: Firebase
- External libraries: Langchain, Streamlit, MoviePy, etc.

### 3.6 Experiment and Results for Validation and Verification

Experiment	Result
Using llama 2 for story generation	<ul style="list-style-type: none"><li>As llama 2 is a big model with a large amount of parameters, it takes long time for text generation</li><li>Not producing expected output</li></ul>
Using Gemini fine tuning	<ul style="list-style-type: none"><li>A lot of description was required along with story topic for getting desired output</li></ul>
Using fine tuned gemini model	<ul style="list-style-type: none"><li>Achieved desired output after fine tuning the model with dataset containing around 100 stories</li></ul>
Using Stable Diffusion	<ul style="list-style-type: none"><li>Took around 40-50 minutes for generating one single image</li></ul>
Using Midjourney	<ul style="list-style-type: none"><li>Using their API key was paid and quite expensive as well as there were limits on number of images that could be generated on a single day</li></ul>
Using flux-lego-lora-dream booth	<ul style="list-style-type: none"><li>Achieved desired output with accurate and good quality images</li><li>Capable of generating multiple images at the same time</li><li>Took around 1-2 minutes to generate all the images</li></ul>

*Table no. 03 : Experiments & results*

## 3.7 Result Analysis and Discussion

### Story Generation : Using Gemini 1.5 Flash Model

This is the base model which generates the output in a long story format which is not desirable for the generation of social stories . Hence we fine tune it to get desired results.

The screenshot shows the Gemini 1.5 Flash interface. At the top, there's a title "Lily's Lesson: Respecting Elders" with a pencil icon. Below that, a dropdown menu says "Gemini 1.5 Flash" with a dropdown arrow. To the right are three icons: a double arrow, a gear, and an X. Underneath, a section titled "System Instructions" has a "User" button. The user input is "Generate a social story on topic Respecting Elders". The model response is timestamped "6.5s" and includes a warning icon. The generated story is titled "The Day I Learned About Respecting Elders". The story content is as follows:

Once upon a time, there was a little girl named Lily who loved to play outside. She had a big, beautiful backyard filled with flowers and trees, and she spent hours running, jumping, and laughing. One day, Lily was playing tag with her friends when she ran into a group of older people sitting on a bench under a shady tree.

Lily was so excited, she didn't stop running. She bumped into the bench, knocking over a bag of knitting needles. The old woman sitting on the bench gasped and looked upset.

"Oh, I'm so sorry!" Lily exclaimed, rushing to help pick up the needles. "I didn't see you there!"

At the bottom, there's a text input field with the placeholder "Type something".

Fig no. 03 : Story Generation using Gemini model

This is the fine tuned version of the Gemini 1.5 Flash model and gives the desired output which can be used by other models for image and audio generation . This model is fine tuned using a dataset .

The screenshot shows a user interface for generating stories. At the top, there is a dropdown menu labeled "social\_stories" with a downward arrow. To the right of the menu are three icons: a double-headed arrow, a gear, and a close button. Below this header, there are two sections: "User" and "Model". The "User" section contains the prompt "Generate a social story on topic Respecting Elders". The "Model" section displays the generated response: "Sophia always shows respect to her elders by being polite and listening to what they have to say. She helps them if they need assistance and remembers that they have lived longer, so she values their experiences and wisdom. Sophia understands that showing respect is important because it makes her elders feel loved and appreciated. When she is older, Sophia hopes people will show the same respect to her, and she will continue to be a good example for others." At the bottom right of the interface are three buttons: a plus sign inside a circle, a star inside a circle labeled "Run", and a circular arrow icon.

Fig no. 04 : Story Generation using Fine Tuned Model

## Image prompt generation

This is the base model which generates the image prompts in a long descriptive format which is not desirable for the generation of relevant images as the image generating model expects short and descriptive image prompts . Hence we fine tune it to get better results.

Sophia's Respectful Nature ✍

Gemini 1.5 Flash ▼ ↔️ ⚙️ ✕

System Instructions

**User**

Sophia always shows respect to her elders by being polite and listening to what they have to say. She helps them if they need assistance and remembers that they have lived longer, so she values their experiences and wisdom. Sophia understands that showing respect is important because it makes her elders feel loved and appreciated. When she is older, Sophia hopes people will show the same respect to her, and she will continue to be a good example for others.

**Model** 1.4s

This is a beautiful and positive description of Sophia's character! It highlights her:

- **Respectful nature:** She demonstrates respect through her polite behavior, active listening, and willingness to help.
- **Appreciation for elders:** She recognizes their experience and wisdom, understanding that they have valuable insights to offer.
- **Empathy and consideration:** She understands the importance of making elders feel valued and appreciated.
- **Commitment to role modeling:** She aims to be a good example for others and hopes to receive the same respect she shows.

Type something

This is the similar fine tuned version of the Gemini 1.5 Flash model and gives the desired image prompt which can be used by image model for image generation . This model is fine tuned using a dataset which consists of story and its corresponding image prompts.

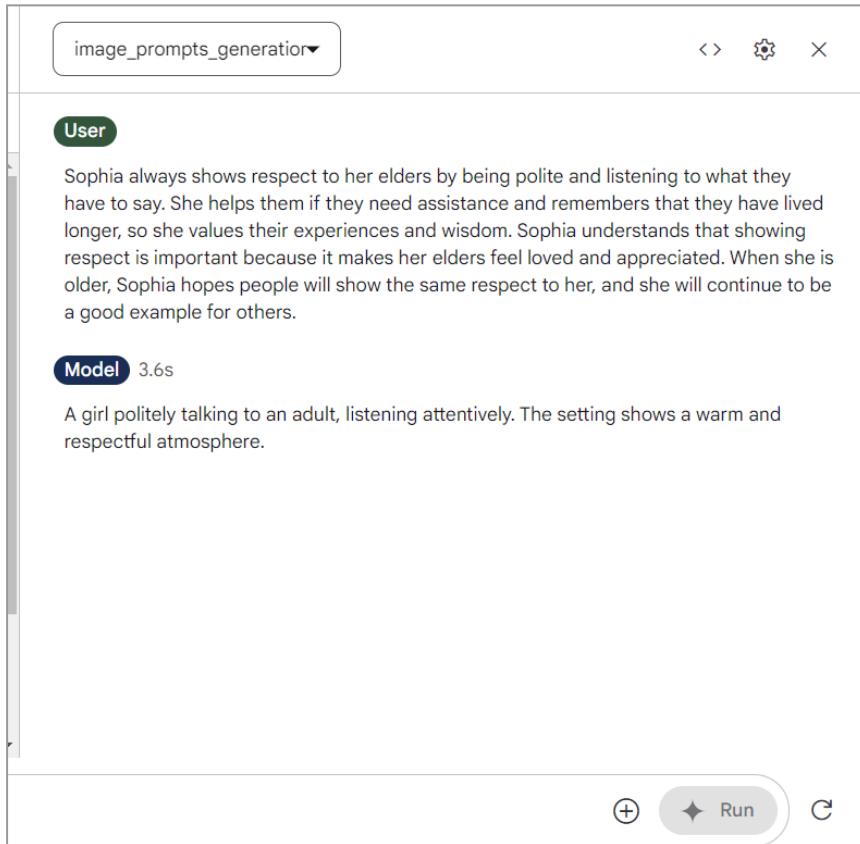


Fig no. 05 : Image prompt Generation

## Implementation (Output)

This is the first screen that pops up when you click on the website link. We have also added an animation to this page, which enhances user engagement and captures attention right from the start.



Fig no. 06 : Starter Screen

This is the homepage where users can navigate to other pages and also can give input prompts and press the generate button to generate the story.

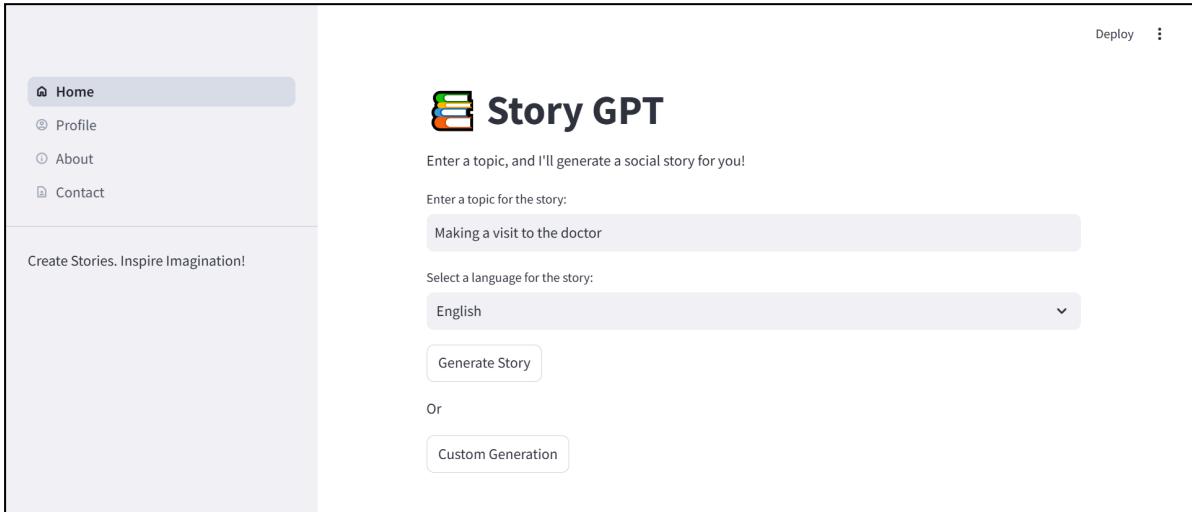


Fig no. 07 : Home Page

After clicking on generate story, We get directed to this page where our text generation model and image generation model combine to give a multiple image and text media output

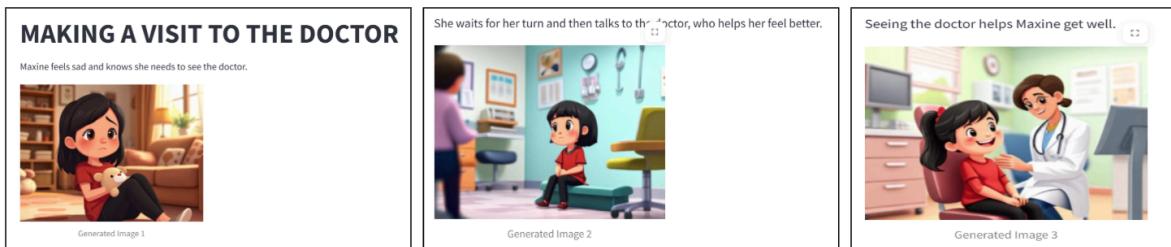


Fig no. 08 : Story Output Screen

This is the section within the output screen which contains the Audio generation button to get the audio output of social story.

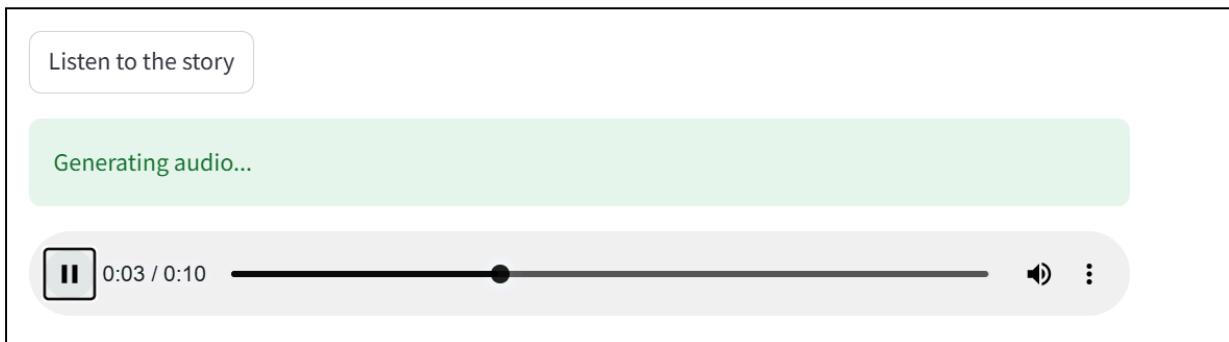
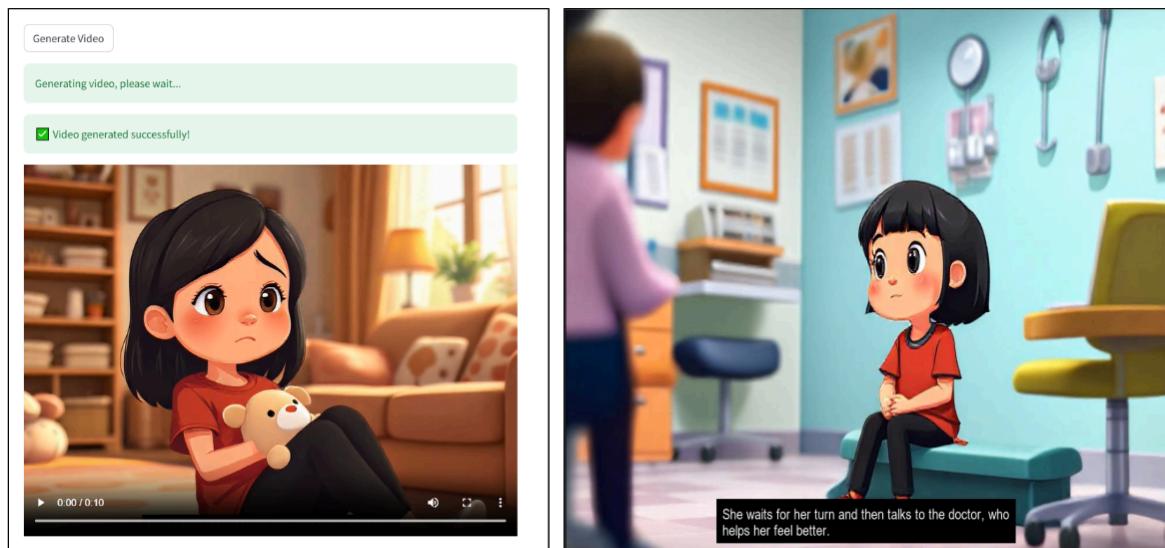


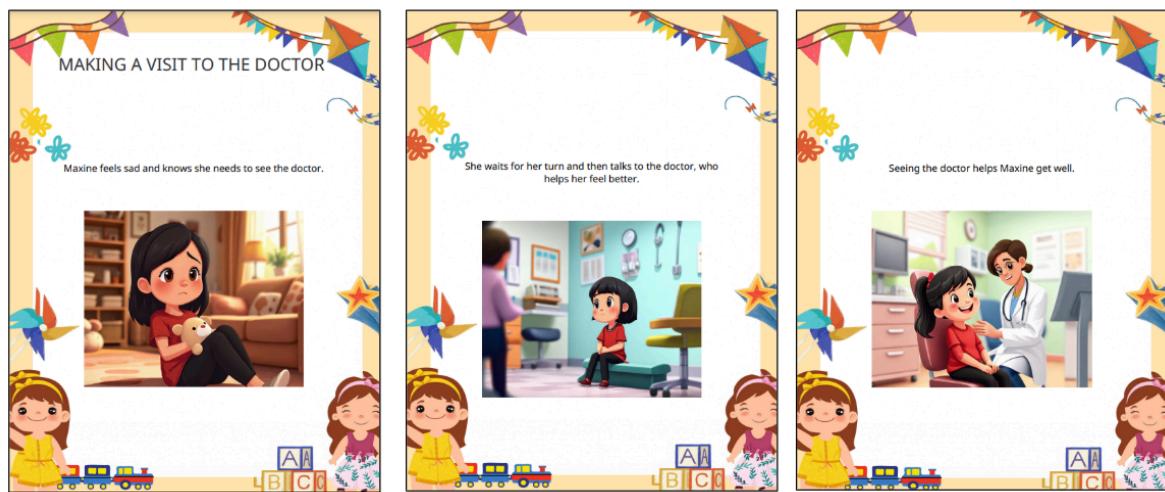
Fig no. 09 : Audio Output Screen

This is the screen where users can obtain a video output to the social story.



*Fig no. 10 : Video Output Screen*

This is the pdf output which is downloaded into the downloads folder of your PC.



*Fig no. 11 : Pdf format output*

## Implementation (Codes And Algorithm) :

### Function to generate story

```
FUNCTION generate_story(topic)
    // Step 1: Initialize the generative model
    SET story_model = CreateGenerativeModel("tunedModels/socialstories-my8gxcmyb2vx")

    // Step 2: Start a new chat session
    SET story_session = story_model.StartChat(history=[])

    // Step 3: Generate a story based on the provided topic
    SET story = story_session.SendMessage("Generate a social story on topic " + topic).text

    // Step 4: Extract story components
    SET story_list = get_story_list(story)

    // Step 5: Return the results
    RETURN {
        'story': story,
        'story_list': story_list
    }
END FUNCTION
```

Fig no. 12 : generate\_story function

### Function to generate image prompt

```
FUNCTION generate_image_prompt(sentence)
    // Step 1: Initialize the generative model for image prompts
    SET image_prompt_model = CreateGenerativeModel("tunedModels/imagepromptsgeneration-n56rol")

    // Step 2: Start a new chat session for generating image prompts
    SET image_prompt_session = image_prompt_model.StartChat(history=[])

    // Step 3: Generate an image prompt based on the provided sentence
    SET image_prompt = image_prompt_session.SendMessage(sentence).text

    // Step 4: Return the generated image prompt
    RETURN image_prompt
END FUNCTION
```

Fig no. 13 : generate\_image\_prompt function

## Function to generate story and image in proper format

```
FUNCTION generate_content(topic)
    // Step 1: Generate the story data using the given topic
    SET story_data = generate_story(topic)

    // Step 2: Extract the list of sentences from the generated story
    SET story_sentences = story_data['story_list']

    // Step 3: Initialize an empty list for image prompts
    SET image_prompts = []

    // Step 4: Generate image prompts for each sentence in the story
    FOR i FROM 0 TO LENGTH(story_sentences) - 1 DO
        SET prompt = generate_image_prompt(story_sentences[i])
        APPEND prompt TO image_prompts
    END FOR

    // Step 5: Print each generated image prompt
    FOR EACH prompt IN image_prompts DO
        PRINT prompt
        PRINT '\n'
    END FOR

    // Step 6: Return the results including story, story list, and image prompts
    RETURN {
        'story': story_data['story'],
        'story_list': story_data['story_list'],
        'image_prompts': image_prompts
    }
END FUNCTION
```

Fig no. 14 : generate\_content function

```
FUNCTION get_story_list(story)
    // Step 1: Split the story into sentences based on the period character
    SET sentences = story.split('.')

    // Step 2: Remove any leading/trailing whitespace and filter out empty sentences
    SET sentences = [s.strip() FOR EACH s IN sentences IF s.strip() IS NOT EMPTY]

    // Step 3: Add a period to the end of each sentence
    SET sentences = [s + '.' FOR EACH s IN sentences]

    // Step 4: Print the list of sentences for debugging purposes
    PRINT sentences

    // Step 5: Return the list of sentences
    RETURN sentences
END FUNCTION
```

Fig no. 15 : split story into parts

## Function for image generation

```
FUNCTION generate_image(prompt)

    // Step 1: Define the query function to send requests
    FUNCTION query(payload)
        TRY
            PRINT "Sending payload: " + payload // Log the payload being sent
            SET response = POST(API_URL, headers=headers, json=payload) // Send POST request
            PRINT "Response status code: " + response.status_code // Log the response status
            response.raise_for_status() // Raise an error for bad responses (4xx and 5xx)
            RETURN response.content // Return the response content
        EXCEPT requests.exceptions.HTTPError AS http_err
            PRINT "HTTP error occurred: " + http_err // Log HTTP errors
            PRINT "Response text: " + response.text // Print response text for more details
            RETURN None // Return None on error
        EXCEPT Exception AS err
            PRINT "Other error occurred: " + err // Log other errors
            RETURN None // Return None on error
        END TRY
    END FUNCTION

    // Step 2: Call the query function with the prompt
    SET image_bytes = query({"inputs": prompt})

    // Step 3: Check if image_bytes is valid and return the image
    IF image_bytes IS NOT None THEN
        RETURN Image.open(io.BytesIO(image_bytes)) // Return PIL Image object
    ELSE
        PRINT "Failed to generate image for prompt: " + prompt // Log failure message
        RETURN None // Return None on failure
    END IF
END FUNCTION
```

Fig no. 16 : generate images using image prompts

## Function for audio generation

```
SET tts_url = "https://api.elevenlabs.io/v1/text-to-speech/" + VOICE_ID + "/stream"

    // Step 2: Prepare the headers for the API request
    SET headers = {
        "Accept": "application/json",
        "xi-api-key": xi_api_key // Include the API key for authorization
    }

    // Step 3: Define the payload data for the TTS request
    SET data = {
        "text": text, // Text to be converted to speech
        "model_id": "eleven_multilingual_v2", // Specify the model to use
        "voice_settings": {
            "stability": 0.8, // High stability for clear speech
            "similarity_boost": 0.9, // High resemblance to the base voice
            "style": 0.2, // Low expressiveness for neutral tone
            "use_speaker_boost": False // No extra intensity
        }
    }
```

Fig no. 17 : audio generation-I

```

// Step 4: Make the POST request to generate audio
SET response = POST(tts_url, headers=headers, json=data, stream=True)

// Step 5: Check if the response is OK
IF response.ok THEN
    // Step 5a: Create an in-memory binary stream to hold the audio data
    SET audio_stream = CreateBytesIO() // Initialize an empty byte stream

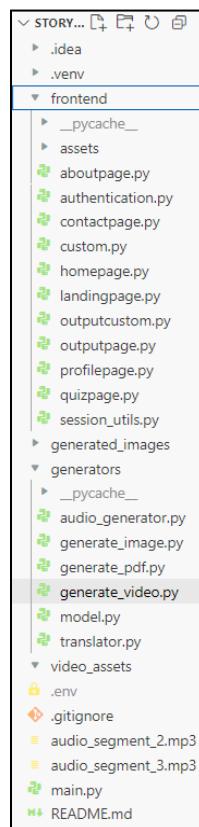
    // Step 5b: Write the audio data in chunks to the stream
    FOR EACH chunk IN response.iter_content(chunk_size=1024) DO
        audio_stream.write(chunk)
    END FOR

    // Step 5c: Move the cursor to the beginning of the stream
    audio_stream.seek(0)
    RETURN audio_stream // Return the audio stream
ELSE
    PRINT "Error: " + response.text // Log error details
    RETURN None // Return None on failure
END IF
END FUNCTION

```

*Fig no. 18 : audio generation - 2*

## Folder Structure of the project



*Fig no. 19 : folder structure*

- All the frontend files are kept inside the frontend folder .
- All the files containing the models and apis to generate text, images, audio and pdfs are kept inside the generators folder .
- All the credentials of the various apis , api keys and secret ids are kept in the .env file .
- And the main.py file contains the starter code responsible for user authentication .

### **3.8 Conclusion and Future work**

Our project aims to create a transformative tool that enhances social skill development for children with special needs through personalized, AI-driven educational content combined with immersive experiences. We will utilize advanced AI tools like Gemini and diffusers to create tailored social stories and images that meet each child's specific learning and sensory needs. This innovative approach addresses the challenges these children face in traditional learning environments and offers a flexible solution suitable for various educational and therapy settings. As we continue to improve and develop our prototype, our primary goal is to help children with special needs navigate social situations more effectively, ultimately enhancing their quality of life. This project represents a significant step toward making learning easier, more engaging, and more effective for those who need it most. Through ongoing development, testing, and collaboration, we believe our solution will have a positive impact on the lives of children and their families.

### **Future Scope**

1. Multilingual Audio Generation: Expand audio capabilities to support multiple languages for broader accessibility.
2. NLP Integration: Implement natural language processing to highlight specific words in the text, enhancing engagement and user-friendliness.
3. Video Creation: Develop a process to generate videos from text and images, incorporating multimedia elements to enrich storytelling.
4. Testing and Optimization: Conduct thorough testing of the entire workflow to improve quality and user experience based on feedback.
5. Collaboration with Educators: Partner with educators and therapists to ensure the tool meets the needs of students and can be effectively integrated into educational settings.

## 4. References

- [1] M. Alabdulrahman, R. Khayyat, K. Almowallad, and Z. Alharz, "Sarid: Arabic storyteller using a fine-tuned LLM and text-to-image generation," in *Proc. of the 2024 16th International Conference on Computer and Automation Engineering (ICCAE)*, 2024.
- [2] C. Gou, A. Felemban, F. F. Khan, D. Zhu, J. Cai, H. Rezatofighi, and M. Elhoseiny, "How Well Can Vision Language Models See Image Details?," *arXiv preprint arXiv:2408.03940*, 2024.
- [3] M. Konenkov, A. Lykov, D. Trinitatova, and D. Tsetserukou, "VR-GPT: Visual Language Model for Intelligent Virtual Reality Applications," *arXiv preprint arXiv:2405.11537*, 2024.
- [4] H. Chang, H. Zhang, J. Barber, A. J. Maschinot, J. Lezama, J. Lu, M.-H. Yang, K. Murphy, W. T. Freeman, M. Rubinstein, Y. Li, and D. Krishnan, "Muse: Text-To-Image Generation via Masked Generative Transformers," *arXiv preprint arXiv:2301.00704*, 2023.
- [5] J. Izquierdo-Domenech, J. Linares-Pellicer, and I. Ferri-Molla, "Virtual Reality and Language Models, a New Frontier in Learning," [Online]. Available: <https://drive.google.com/file/d/1DXdgXvhx9zYKlvvZM4kYvRxIPykeRgJg/view>
- [6] J. Shen, J. Mire, H. W. Park, C. Breazeal, and M. Sap, "HEART-felt Narratives: Tracing Empathy and Narrative Style in Personal Stories with LLMs," *arXiv preprint arXiv:2405.17633*, 2024.
- [7] "LoRA: Low-Rank Adaptation of Large Language Models," *arXiv preprint arXiv:2106.09685*, 2021.
- [8] "MuRIL: Multilingual Representations for Indian Languages," *arXiv preprint arXiv:2103.10730*, 2021.
- [9] H. Chang, H. Zhang, J. Barber, A. J. Maschinot, J. Lezama, J. Lu, M.-H. Yang, K. Murphy, W. T. Freeman, M. Rubinstein, Y. Li, and D. Krishnan, "Muse: Text-To-Image Generation via Masked Generative Transformers," *arXiv preprint arXiv:2301.00704*, 2023.
- [10] "Cross-Lingual Conversational Speech Summarization with Large Language Models," *arXiv preprint arXiv:2408.06484*, 2024.

- [11] "CT-Eval: Benchmarking Chinese Text-to-Table Performance in Large Language Models," *arXiv preprint arXiv:2405.12174*, 2024.
- [12] J. Shen, J. Mire, H. W. Park, C. Breazeal, and M. Sap, "HEART-felt Narratives: Tracing Empathy and Narrative Style in Personal Stories with LLMs," *arXiv preprint arXiv:2405.17633*, 2024.
- [13] "LLM-DetectAIve: a Tool for Fine-Grained Machine-Generated Text Detection," *arXiv preprint arXiv:2408.04284v1*, 2024.
- [14] "LLaVA-Surg: Towards Multimodal Surgical Assistant via Structured Surgical Video Learning," *arXiv preprint arXiv:2408.07981*, 2024.
- [15] M. Konenkov, A. Lykov, D. Trinitatova, and D. Tsetserukou, "VR-GPT: Visual Language Model for Intelligent Virtual Reality Applications," *arXiv preprint arXiv:2405.11537*, 2024.

# Social Stories Generator : An LLM-Based Learning Tool for Specially-Abled Children

Sujata Khedkar

*VES Institute of Technology*

sujata.khedkar@ves.ac.in

Mohit Vaidya

*VES Institute of Technology*

2022.mohit.vaidya@ves.ac.in

Sarang Pavanaskar

*VES Institute of Technology*

2022.sarang.pavanaskar@ves.ac.in

Akshat Mahajan

*VES Institute of Technology*

2022.akshat.mahajan@ves.ac.in

Tanmay Maity

*VES Institute of Technology*

2022.tanmay.maity@ves.ac.in

**Abstract -** The proposed system presents a novel AI-driven approach to creating personalized social stories for children with autism spectrum disorder (ASD). This system integrates five key stages: text generation, image generation, audio generation, PDF generation, and video generation. By fine-tuning large language models (LLMs) such as Gemini on a dataset of over 100 social stories, the system customizes the generated stories to match each child's specific needs. To improve social understanding and behavior of a child, the system integrates multimedia elements such as images, audio, and video. This approach demonstrates how generative AI can support education for children with special needs by automating the manual process of creating social stories and providing a personalized, multi-sensory learning experience.

**Keywords -** Large Language Models, Generative AI, Social Stories, Image Generation, Video Generation, Text generation, Fine-tuning.

## I. INTRODUCTION

Social Stories are a helpful tool for children with special needs, especially those with Autism Spectrum Disorder (ASD), to understand social situations and learn how to respond appropriately. These stories explain specific situations, guide children on the right way to behave, and show the expected outcomes. They play an important role in developing social and communication skills. However, creating Social Stories manually takes a lot of time, is often rigid, and can be difficult to tailor to each child's needs. This lack of personalization makes them less effective.

With the growth of Artificial Intelligence (AI) and Large Language Models (LLMs), it is now possible to improve

how Social Stories are created and used. AI-powered tools can automate story generation while adding personalized text, images, and audio, making them more engaging for children with different learning styles.

The proposed system uses AI models like Gemini to generate text-based stories and FLUX-1.schnell model to create images that go along with them. This allows for a fully customized experience that fits each child's learning needs.

The main aim of this project is to make Social Stories easier to create and more adaptable. By using AI to generate stories with multimedia elements, we hope to improve how children with special needs learn social skills. This paper will explain the technical details of our Social Stories Generator, including how we prepared the data, fine-tuned the AI models, and designed the system to be simple for parents, teachers, and caregivers to use.

## II. RELATED WORK

*A. Sarid: Arabic Storyteller Using a Fine-Tuned LLM and Text-to-Image Generation (2024)*

Sarid is a project focused on developing an Arabic story generation model using a fine-tuned language model and a text-to-image generator. The team collected a dataset of 527 Arabic children's stories and fine-tuned OpenAI's Davinci model to generate stories based on user inputs such as theme, tone, and character.

Visual elements are created using Midjourney, ensuring high-quality images that align with the generated stories. While the system maintains character consistency, it requires careful prompt engineering to do so.

Some challenges arise when specific seeds are not used, leading to inconsistencies in character continuity. Additionally, the model's performance depends on the quality of the dataset and the fine-tuning process [1].

#### B. Deep Learning Methodology Converts Text to Image (2022)

This study examines the use of Generative Adversarial Networks (GANs) for converting textual descriptions into visual content. Researchers implemented GAN-CLS, a text-to-image generation algorithm, using the Oxford-102 flower dataset in TensorFlow to create realistic images from text.

The study highlights GANs' potential in producing visually coherent images that align with textual inputs. However, challenges such as high computational costs, long training times, and difficulties in ensuring text-to-image coherence were observed. These findings emphasize the complexity of text-to-image generation, particularly when applied to descriptive narratives [2].

#### C. Muse: Text-To-Image Generation via Masked Generative Transformers (2023)

Muse, a Transformer-based model, was designed for efficient text-to-image generation. It uses a masked modeling approach with discrete image tokens and text embeddings from a pre-trained large language model. Muse supports parallel decoding, which enhances both performance and efficiency compared to diffusion and autoregressive models.

The model achieved state-of-the-art results in text-to-image generation tasks and demonstrated advanced image editing capabilities without requiring additional fine-tuning. While Muse performs well on benchmarks such as CC3M and COCO, its ability to generalize across broader datasets remains an area for further research [3].

#### D. LLM-DetectAIve: A Tool for Fine-Grained Machine-Generated Text Detection (2024)

LLM-DetectAIve was created to detect and classify texts as human-written, machine-generated, machine-humanized, or human-polished. The project introduced a new dataset that combined various types of human and machine-generated texts. Detectors such as RoBERTa, DeBERTa, and DistilBERT were used to improve the fine-grained detection capabilities. Although the tool provides a nuanced classification system, challenges arise in cross-domain scenarios where text features vary significantly. The study demonstrated the effectiveness of domain-specific detectors but also

highlighted the need for users to select the correct domain for optimal detection performance [4].

### III. DATASET

A fundamental step of our social story generator was the collection of an appropriate dataset for the fine-tuning of the Gemini model. This consisted of gathering a dataset through web scraping and data processing to align it with the required format for model fine-tuning. To fine-tune a Gemini model, we need to have an input and its corresponding output. We collected the dataset, preprocessed it, and organized it in a table with the prompts and stories, making it possible to fine-tune the Gemini model.

#### A. Data Collection

We started by collecting a comprehensive dataset. The dataset for this project was gathered from a combination of publicly available social stories and online educational resources. This diverse collection of data provided a broad range of social scenarios and behavioral lessons that children may encounter in their daily lives, including themes such as Communication skills, Appropriate behavior in social settings, Emotional regulation, Personal safety.

The dataset for the Social Stories Generator was organized in a simple format with two main parts: a title and the corresponding story. We collected various social stories, focusing on situations that children with special needs, especially those with Autism Spectrum Disorder (ASD), commonly face. Each entry in the dataset had a title, like "*Going to School*" or "*Making New Friends*," followed by a story that explained how to handle that situation in a way the child could understand. This structure was used to fine-tune the model, enabling it to generate personalized stories based on specific scenarios provided by users.

#### B. Prompts Formulation

The next phase in the fine-tuning process for the Gemini model involved the careful creation of prompts, which were essential for guiding the model in generating both text and image prompts.

To formulate these prompts, we first gathered essential story elements such as titles, descriptions, and relevant narrative information from a curated dataset. In the case of image prompts, we ensured that the description aligned with key visual features of the story, such as characters and settings. For instance, character attributes like appearance, clothing, and background environment were

integrated into the prompt to guide the image generation model in producing corresponding illustrations.

This dual-layered prompt creation, comprising text and image guidance, allowed the Gemini model to generate cohesive narratives and guide the FLUX model to create visually consistent images.

#### IV. MODEL CONSTRUCTION

This section explains the model architecture and optimization process, which involves fine-tuning a large language model (LLM) to create personalized social stories and using AI tools to add images and audio.

##### A. Fundamental Model

Our approach uses Gemini 1.5 Flash, an LLM known for its proficiency in natural language generation and interpretation. This model is based on few-shot learning capabilities and extensive pre-training, allowing it to create meaningful and contextually appropriate social stories.

To ensure the generated stories align with the needs of children with Autism Spectrum Disorder (ASD), Gemini 1.5 Flash has been fine-tuned on a dataset of over 100 social stories. These stories are designed to assist children with special needs, particularly those with ASD, in understanding social cues and behaviors.

##### B. Fine Tuning

Fine-tuning Gemini for social story generation is done in two phases, keeping in mind the specific needs of children with ASD.

The first phase focuses on text generation. The model is trained using a dataset of 100 social stories that cover various social situations, behavioral cues, and cultural differences. This helps the model create meaningful and relevant stories.

In the second phase, the model is fine-tuned to generate image prompts. It analyzes the story text to extract key themes and descriptions, then creates structured prompts for image generation.

##### C. Image Generation

We integrate image generation into the storytelling process using the FLUX-1.schnell model, which is highly effective at generating detailed, visually appealing illustrations based on text prompts. The image generation process ensures that the visuals align with the story's narrative and emotional context. It begins with Gemini

1.5 Flash, which creates structured image prompts based on the story. These prompts are then processed by FLUX-1.schnell to generate high-quality illustrations that match the storyline, character development, and emotions conveyed in the text.

Each story scene is accompanied by a corresponding image that helps children visualize the social context described in the story. Consistency in character design and scene development is maintained throughout the story.

We prompt the Flux model with detailed descriptions derived from the story's narrative to ensure that the generated images align with the storyline and character progression.

##### D. Audio Generation

To support children who learn better through auditory input, the system includes text-to-speech (TTS) conversion using the ElevenLabs API.

The TTS model adjusts pitch, tone, and speed to create clear and engaging narration. Emotionally expressive voice synthesis enhances the storytelling experience, helping children connect with the story. The generated audio can be played within the application or downloaded for offline use, providing flexibility for different needs.

##### E. Multilingual Translation

To make social stories accessible to a wider audience, the system includes multilingual translation, supporting Hindi and Marathi alongside English. The translation process uses Google Translator from the Deep Translator library to convert the generated stories while maintaining accuracy and contextual relevance.

Once translated, the text is processed through the ElevenLabs API, which generates speech synthesis for the translated versions. This ensures that both text and audio are available in multiple languages, allowing non-English-speaking children and caregivers to engage with the content in their preferred language. By integrating multilingual support, the system enhances accessibility and inclusivity for diverse users.

##### F. Quiz Generation

The system includes an interactive quiz module powered by Gemini 1.5 Flash. This feature creates assessments based on each generated story, helping caregivers and educators evaluate a child's understanding. The content dynamically adapts to match the story, making it relevant to the social lessons presented.

## G. PDF Generation and Download

For offline access, the system provides an option to generate a downloadable PDF of the social story. The PDF includes the full story and illustrations creating a complete learning resource.

## H. Video Generation

The system includes video generation using the MoviePy Python library. This feature combines text, images, and audio into an animated storytelling video, providing an alternative to static stories.

The process begins by arranging the generated images according to the story's sequence. The narrated audio is then synchronized with the visuals. Subtitles are added for better engagement.

The final video is available along with subtitles, allowing users to download or stream it directly from the platform. By converting social stories into interactive videos, the system offers a visually engaging learning tool for children with ASD.

The system follows a modular architecture, integrating multiple AI-driven components for the generation of social stories.

## V. PIPELINE

**User Authentication:** Managed via Firebase to ensure secure access. Users log in to access the homepage, where they can choose between Title Prompt Generation, allowing them to enter a prompt to generate a story, or Custom Input, where they can manually enter text or upload a PDF file.

**Story Generation and Processing:** Google AI Studio processes the input using Gemini 1.5 Flash for text generation, image prompt formulation, and quiz development. The system ensures coherence and relevance in the generated content.

**Audio Narration and Translation:** The ElevenLabs API converts the generated text into speech, providing a natural-sounding narration. The system also supports multilingual translation using Google Translator from the Deep Translator library, enabling story conversion into Hindi and Marathi with both text and voice support.

**Output and User Interaction:** The final content is made available in multiple formats, including PDF (containing the story, images), audio (narration for auditory learners), video (integrating text, images, and narration) and an interactive quiz (to assess comprehension and reinforce learning).

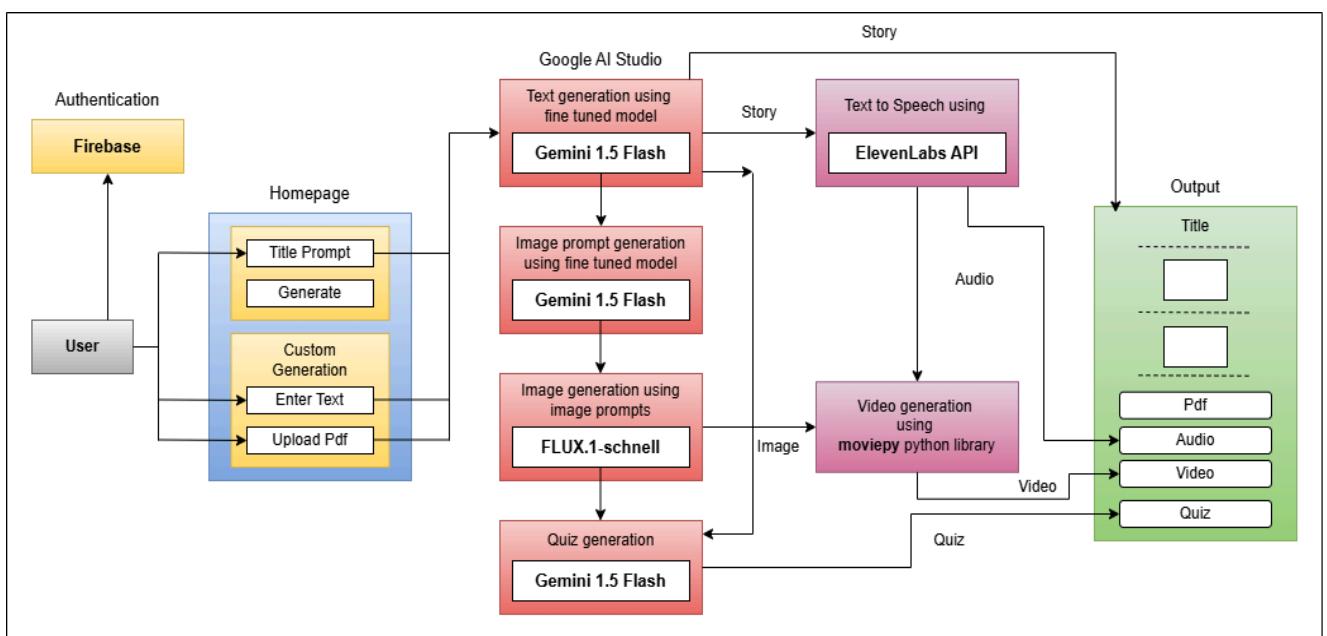


Fig. 1 Pipeline

## VI. RESULTS

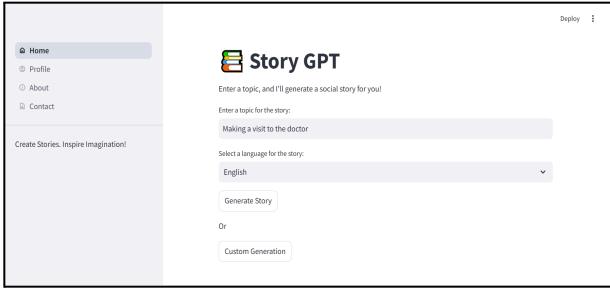


Fig. 2 Homepage of social story generator

### MAKING A VISIT TO THE DOCTOR

Maxine feels sad and knows she needs to see the doctor.



Generated Image 1

Fig. 3 Story text and image generation

Listen to the story

Generating audio...

0:03 / 0:10

Fig. 4 Audio output of social story

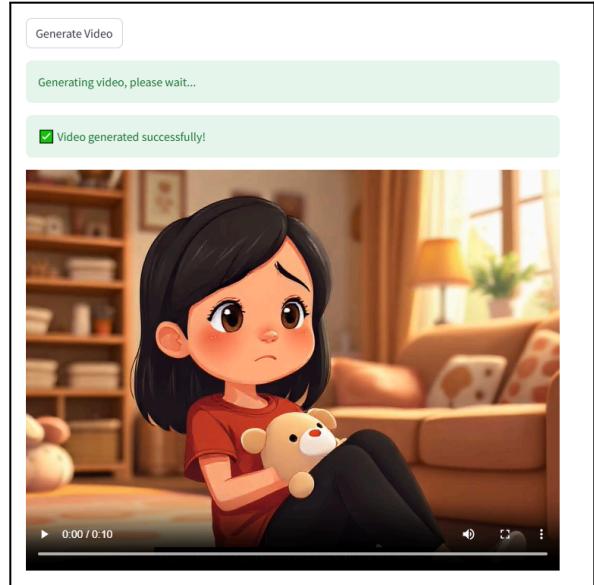


Fig. 5 Video output of social story

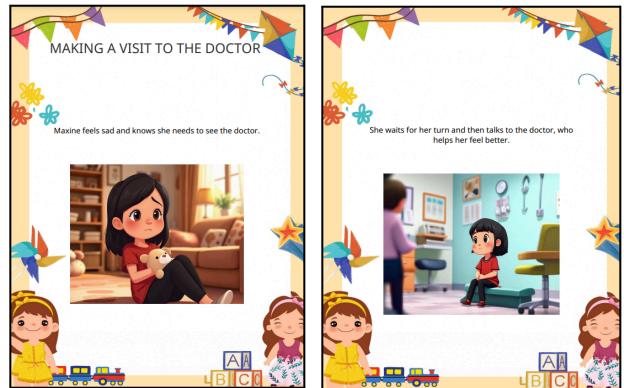


Fig. 6 PDF output of social story

Fig. 2 displays the homepage, where users can select a language, enter a prompt, and generate a personalized story, making the system easy to navigate. Fig. 3 illustrates the story text and image generation, ensuring visually immersive storytelling by combining AI-generated illustrations with meaningful narratives that enhance comprehension and engagement.

The system's text-to-speech feature as shown in Fig. 4 converts the generated story into natural narration, making it accessible to auditory learners while improving

the overall storytelling experience. Fig. 5 highlights the video output, which integrates text, images, and audio into a multimedia format, making learning more engaging and interactive. Lastly, Fig. 6 presents the PDF output, enabling offline accessibility for educators, therapists, and caregivers, ensuring that the generated stories can be used across different environments, including classrooms and therapy sessions.

## Quiz Page

Answer the following questions:

**Q1: How does Maxine feel before seeing the doctor? **

Select an answer for Q1

- Scared
- Sad
- Happy
- Confused

[Submit Q1](#)

**Q2: What happens after Maxine waits?**

Select an answer for Q2

- She talks to the doctor
- She leaves
- She waits more
- She plays

[Submit Q2](#)

Fig. 7 Quiz assessment based on generated social story

## Custom Story Generation

Choose input method:

- Enter text
- Upload PDF

Enter your text:

Adam loves playing video games. Sometimes they make him mad. Rowan gives Adam some important tips to de-escalate when this happens.

Enter the topic of the story:

Select a language for the story:

English

[Generate Story](#)

[Back to Home](#)

Fig. 8 Customized Social Story Generation by entering text or uploading any pdf

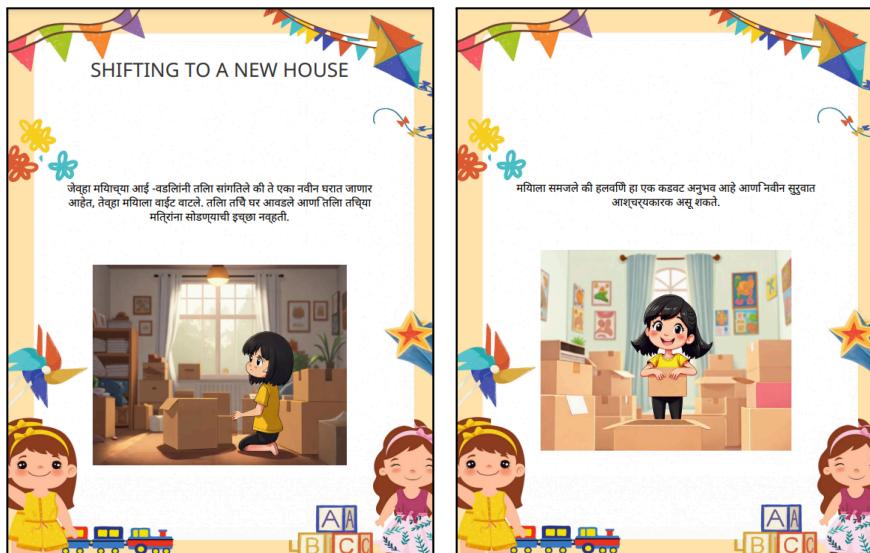


Fig. 9 Example of Multilingual (Marathi) output of social story

The quiz assessment feature, as shown in Fig. 7, evaluates comprehension by generating quizzes based on the created social story. This helps reinforce learning by prompting children to reflect on the story's content, making the experience more engaging and educational.

The system also supports custom story generation, as depicted in Fig. 8, allowing users to manually enter text or

upload a PDF to create a personalized social story. This feature provides flexibility in content creation, catering to diverse user needs and preferences. Educators, therapists, and parents can customize stories to align with specific learning objectives or address particular social situations. Furthermore, the multilingual output capability is demonstrated in Fig. 9, showcasing the system's ability to translate stories into different languages, such as Hindi and Marathi.

## VII. CONCLUSION

The Social Stories Generator is a product of Generative AI designed to help children with special needs, especially those with Autism Spectrum Disorder (ASD), learn social skills through simple, customized stories. It uses language models like Gemini to create text-based stories and adds images and audio to make learning more engaging.

This tool helps caregivers and teachers of children with autism by reducing the effort needed to create social stories. Features like visual support and multiple languages make it easier for more children to understand and use.

Future improvements can include dynamic video generation, improved personalization, an upgraded user interface, and more real-life situations in social stories.

## ACKNOWLEDGEMENTS

We express our sincere gratitude to Dr. Sujata Khedkar, our project mentor, for her invaluable guidance, continuous support, and insightful feedback throughout this research. Her knowledge and encouragement have been instrumental in shaping our work.

We also extend our heartfelt appreciation to Dr. Nupur Giri, Head of the Department of Computer Engineering, and Dr. Gresha Bhatia, Deputy Head of the Department, for providing us with the necessary resources, a conducive research environment, and their constant motivation.

We also extend our gratitude to Vivekanand Education Society's Institute of Technology (VESIT) for offering us the platform to develop this project.

## REFERENCES

- [1] M. Alabdulrahman, R. Khayyat, K. Almowallad, and Z. Alharz, "Sarid: Arabic storyteller using a fine-tuned LLM and text-to-image generation," in *2024 16th International Conference on Computer and Automation Engineering (ICCAE)*, 2024.
- [2] C. Gou, A. Felemban, F. F. Khan, D. Zhu, J. Cai, H. Rezatofighi, and M. Elhoseiny, "Deep learning methodology converts text to image," in *Proceedings of the International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.
- [3] H. Chang, H. Zhang, J. Barber, A. J. Maschinot, J. Lezama, J. Lu, M.-H. Yang, K. Murphy, W. T. Freeman, M. Rubinstein, Y. Li, and D. Krishnan, "Muse: Text-to-image generation via masked generative transformers," *arXiv preprint arXiv:2301.00704*, 2023.
- [4] J. Wei, L. Wang, and D. Li, "LLM-DetectAIve: A tool for fine-grained machine-generated text detection," *arXiv preprint arXiv:2408.04284v1*, 2024.
- [5] J. Shen, J. Mire, H. W. Park, C. Breazeal, and M. Sap, "HEART-felt narratives: Tracing empathy and narrative style in personal stories with LLMs," *arXiv preprint arXiv:2405.17633*, 2024.
- [6] X. Hu, Y. Shen, Z. Tan, R. Yang, and J. Yang, "LoRA: Low-Rank Adaptation of Large Language Models," *arXiv preprint arXiv:2106.09685*, 2021.
- [7] R. Khapra, P. Bhowmick, A. Kunchukuttan, and M. Murthy, "MuRIL: Multilingual Representations for Indian Languages," *arXiv preprint arXiv:2103.10730*, 2021.
- [8] H. Chang, H. Zhang, J. Barber, A. J. Maschinot, J. Lezama, J. Lu, M.-H. Yang, K. Murphy, W. T. Freeman, M. Rubinstein, Y. Li, and D. Krishnan, "Muse: Text-to-image generation via masked generative transformers," *arXiv preprint arXiv:2301.00704*, 2023.
- [9] Z. Zheng, B. He, and H. Lin, "Cross-lingual conversational speech summarization with large language models," *arXiv preprint arXiv:2408.06484*, 2024.
- [10] H. Wang, Y. Liu, and X. Sun, "CT-Eval: Benchmarking Chinese text-to-table performance in large language models," *arXiv preprint arXiv:2405.12174*, 2024.
- [11] J. Shen, J. Mire, H. W. Park, C. Breazeal, and M. Sap, "HEART-felt narratives: Tracing empathy and narrative style in personal stories with LLMs," *arXiv preprint arXiv:2405.17633*, 2024.
- [12] J. Wei, L. Wang, and D. Li, "LLM-DetectAIve: A tool for fine-grained machine-generated text detection," *arXiv preprint arXiv:2408.04284v1*, 2024.
- [13] S. Zhang, Y. Chen, and T. Liu, "LLaVA-Surg: Towards multimodal surgical assistant via structured surgical video learning," *arXiv preprint arXiv:2408.07981*, 2024.

**Industry / Inhouse:  
Research / Innovation:**

**Project Evaluation Sheet 2024-25(Sem 6)**

**Class: D12A/B/C**

(68)

Title of Project (Group no): 08 / Social stories Generator, An LLM-Based Learning Tool for Specially-abled children

Mentor Name & Group Members: Dr. Sujata Khedkar, Savang (D12A-5), Akshat (D12A-40), Tamay (D12A-42), Mohit (D12A-6), Vaibhav (D12A-7), Nidhi (D12A-8)

Review of Project Stage I	Engineering Concepts & Knowledge (5)	Interpretation of Problem & Analysis (5)	Design / Prototype (5)	Interpretation of Data & Dataset (3)	Modern Tool Usage (5)	Societal Benefit, Safety Consideration (2)	Environ- ment Friendly (2)	Ethics Team work (2)	Presenta- tion Skills (3)	Applied Engg & Mgmt principles (3)	Life - long learning (3)	Profes- sional Skills (5)	Innov- ative Appr oach (5)	Total Marks (50)
Review of Project Stage I	4	5	5	5	5	4	2	2	2	3	3	4	46	

Comments:

Good work.

Name & Signature  
Reviewer

Dr. Sujata Khedkar

Review of Project Stage I	Engineering Concepts & Knowledge (5)	Interpretation of Problem & Analysis (5)	Design / Prototype (5)	Interpretation of Data & Dataset (3)	Modern Tool Usage (5)	Social Benefit, Safety Consideration (2)	Environ- ment Friendly (2)	Ethics Team work (2)	Presenta- tion Skills (3)	Applied Engg & Mgmt principles (3)	Life - long learning (3)	Profes- sional Skills (5)	Innov- ative Appr oach (5)	Total Marks (50)
Review of Project Stage I	4	5	5	3	4	2	2	2	3	3	3	4	4	46

Comments:

Date: 01/03/2025

Jyoti D. D.  
Name & Signature  
Reviewer

**Industry / Inhouse:  
Research / Innovation:**

**Project Evaluation Sheet 2024-25(Sem 6)**

**Class: D12A/B/C**

08

Dr. Sujata Kedkar

Title of Project (Group no): 08 : Social Stories Generator: An LLM-Based Learning Tool for Specially-Abled children  
Mentor Name & Group Members: Dr. Sujata Kedkar, Sarang Pawaraskar, Akshat Mahajan, Tamay Mehta, Mohit Vaidya  
D12A-51, D12A-40, D12A-42, D12A-60

	Engineering Concepts & Knowledge	Interpretation of Problem & Analysis	Design / Prototype	Interpretation of Data & Dataset	Modern Tool Usage	Societal Benefit, Safety Consideration	Environment Friendly	Ethics	Team work	Presentation Skills	Applied Engg & Mgmt principles	Life-long learning	Professional Skills	Innovative Approach	Total Marks
Review of Project Stage 1	5	5	5	3	5	2	2	2	2	2	3	3	4	5	49
Comments:	<i>Good work.</i>														

*Dr. Sujata Kedkar*  
Name & Signature Reviewer1

	Engineering Concepts & Knowledge	Interpretation of Problem & Analysis	Design / Prototype	Interpretation of Data & Dataset	Modern Tool Usage	Societal Benefit, Safety Consideration	Environment Friendly	Ethics	Team work	Presentation Skills	Applied Engg & Mgmt principles	Life-long learning	Professional Skills	Innovative Approach	Total Marks
Review of Project Stage 1	5	5	5	3	5	2	2	2	2	2	3	3	4	5	49
Comments:	<i>Good work.</i>														

*Prateek D. Joshi*  
Name & Signature Reviewer2

Date: 01/04/2025