

Major Project Presentation  
on  
**AI Story Telling Application**  
in

**Sumedha Polawar**                      **121B1D004**

**Aditi Ingale**                              **121B1D059**

**Shivani Bahirat**                      **121B1D060**

**Prasad Mangnale**                      **121B1D067**

(A. Y. 2024-25)

Under the guidance of  
**Prof. Vinayak Malavade**



Department of Computer Engineering (Regional Language)  
**PCET's Pimpri Chinchwad College of Engineering**

# Contents

- Introduction
- Motivation
- Objective
- Literature Review
- Database Design
- Proposed System
- Project Plan
- Algorithm Details
- System Architecture
- UML Diagrams
- Testing
- Conclusion
- Paper Publication Details
- References

# Introduction

- Modifying AI-generated narratives to conform to Marathi and English audiences' cultural values, customs, and storytelling preferences.
- Addressing language issues, such as grammatical and structural differences between Marathi Hindi and English storytelling .
- Developing qualitative and quantitative criteria to evaluate the effectiveness creativity and user engagement of AI- generated multi language stories
- According To Users Mood Generating the Abhang's in Marathi .

# Motivation

- ◆ **Bridging the Language and Cultural Gaps**

- ❖ Encourage cross-cultural storytelling to bring together distinct language communities.

- ◆ **Encourages Interactive Learning.**

- ❖ Use AI-generated stories to make language learning more interesting and effective.

- ◆ **Advancing AI in Digital Storytelling**

- ❖ Leverage NLP and machine learning to generate meaningful, context-aware narratives.

# Objectives

- To create an AI-powered interactive storytelling platform that supports Marathi, Hindi, and English, allowing for seamless multilingual storytelling.
- To increase user engagement, provide realistic and engaging story experiences that dynamically adjust to users' preferences and inputs.
- To provide accessibility through a user-friendly interface, the platform should be accessible for people of different ages and languages.

# Literature Review

Reference 1	AI Based Multilingual Chatbot: Advancing Higher Education in Rural Communities
Objectives	The AI-based multilingual chatbot provides a user-friendly interface with support for multiple local languages, ensuring accessibility for rural students. Optimized response generation enables quick and relevant answers, empowering students with personalized educational support through AI and NLP technologies.
Proposed Solution	The proposed solution involves developing an AI-powered multilingual chatbot for rural communities, using NLP to provide educational support in local languages. It offers personalized assistance, leveraging machine learning for accurate query responses.
Results	The AI-based multilingual chatbot improved user interaction and achieved over 90% efficiency in delivering accurate responses. User feedback was highly positive, confirming its effectiveness and ease of use.
Advantages	The AI-based multilingual chatbot enhances accessibility by providing educational support in local languages with a user-friendly interface.
Limitations	The chatbot may have content gaps, language limitations, and require internet access, posing challenges for rural users.

Reference 2	Multi-Lingual DALL-E Storytime Noga Mudrik, Adam S. Charles 22 Dec 2022
Objectives	The Multi-Lingual DALL-E Storytime framework reduces AI language bias and enhances storytelling with coherent image sequences. It visualizes non-English content dynamically while allowing user customization.
Proposed Solution	The Multi-Lingual DALL-E Storytime framework enables coherent storytelling by generating sequential images for non-English narratives. It allows user customization, maintains a consistent artistic style, and enhances accessibility for non-English speakers.
Results	The framework successfully generated coherent visualizations for non-English texts, improving AI accessibility for diverse users. It maintained a consistent artistic style and allowed dynamic storytelling across multiple frames..
Advantages	The framework reduces language bias in AI-generated images, making storytelling accessible to non-English speakers. Users can customize settings for consistency, ensuring tailored storytelling experiences.
Limitations	The model relies on translations, which may lead to inaccuracies in meaning and representation.

Reference 3	MahaSQuAD: Bridging Linguistic Divides in Marathi Question-Answering Ruturaj Ghatage,Aditya Kulkarni,Rajlaxmi Patil,Sharvi Endait,Raviraj Joshi 20 Apr 2024
Objectives	MahaSQuAD aims to create a Marathi question-answering dataset by translating SQuAD into Marathi. The project addresses the lack of robust NLP resources for low-resource languages.
Proposed Solution	A robust translation approach is implemented to accurately map answers within translated passages. The dataset includes 118,516 training, 11,873 validation, and 11,803 test samples. Multiple models, including MahaBERT and mBERT, are trained and evaluated for performance.
Results	The dataset significantly improved Marathi question-answering capabilities, with monolingual models outperforming multilingual ones. MahaBERT and MahaRoBERTa achieved high accuracy, showing better contextual understanding. The approach proved effective in bridging linguistic gaps in Marathi NLP.
Advantages	MahaSQuAD enhances NLP accessibility for Marathi speakers by providing a large, structured dataset. The scalable translation approach can be extended to other low-resource languages.
Limitations	Automated translation may introduce minor inaccuracies and contextual mismatches. The dataset relies on machine translation, which may not always capture linguistic nuances. Maintaining answer span alignment across translations remains a challenging task.



Reference 4	Together We Can: Multilingual Automatic Post-Editing for Low-Resource Languages Sourabh Deoghare,Diptesh Kanojia,Pushpak Bhattacharyya 23 Oct 2024 (Cornell University)
Objectives	This study enhances machine translation quality for low-resource Indo-Aryan languages using multilingual Automatic Post-Editing (APE) . The research integrates Quality Estimation (QE) for better domain adaptation and translation correction.
Proposed Solution	A multilingual APE model is developed with synthetic Hindi-Marathi and Marathi-Hindi APE triplets to enable cross-linguistic transfer. The framework integrates multitask learning with QE for improved accuracy. Data augmentation and domain adaptation techniques further refine translation quality.
Results	The multilingual APE model outperformed single-language APE models, achieving up to 2.5 TER point improvements. Multitask learning boosted translation correction by over 1.29 TER points, with data augmentation adding further refinements.
Advantages	The system improves translation accuracy by leveraging linguistic similarities across languages.The approach is scalable, providing a foundation for improving translations in other low-resource languages .
Limitations	Translation errors may persist due to limited authentic training data. Over-correction can occur, leading to unnecessary modifications. The approach's effectiveness for languages outside the Hindi-Marathi pair remains uncertain and requires further study.

# Database Design :

Key Fields in Firebase :

Field Name	Description
user_id	A unique ID for each session or user interaction.
age_group	The selected age bracket (e.g., 5–9, 10–14).
language	The language in which the story is requested (English/Hindi/Marathi).
story_category	The genre or theme of the story (e.g., Moral, Adventure, Fantasy).
generated_story	The final AI-generated story content.

For Abhang :

Field Name	Description
abhang_id	Unique identifier for each Abhang.
sant_name	Name of the poet-saint (e.g., Tukaram, Namdev, Dnyaneshwar, Eknath).
emotion_tag	Emotion category (e.g., Happy, Motivational, Devotional, Sad).
abhang_text	Complete poetic verse in Marathi.

# Proposed System

## ❑ Module-wise Implementation

Each module is developed separately:

- **User Input** – collects user preferences
- **AI Story Generator** – uses AI to create stories
- **Text-to-Speech** – narrates stories with voice control
- **Abhang Recommender** – fetches emotional poetry
- **Video WebView** – plays curated YouTube content

## ❑ Firebase Integration

- Real-time database used for storing user data, stories, links
- Optional authentication for personalization
- Enables dynamic content delivery

## ❑ Testing & Optimization

- Functional testing across devices and Android versions
- Multilingual validation and performance tuning
- TTS voice clarity, AI story appropriateness

# Project Plan

Phase	Tasks	Expected Duration
Phase1: Requirement Analysis	Gather functional & non-functional requirements, feasibility study, risk assessment	2 Weeks
Phase 2: System Design	Architecture design, technology stack selection, database schema design	2 Weeks
Phase 3: Model Development & Training	Train & fine-tune Gemini Pro 1.5 for storytelling and manually entering abhang in firebase.	3 Weeks
Phase4: Web & Hardware Integration	Develop Flask API & integrate with frontend UI	4 Weeks
Phase5: Testing & Validation	AI accuracy, performance, and security testing	3 Weeks
Phase6: Deployment& Optimization	Firebase hosting setup & cloud function scaling	2 Weeks
Phase7: Final Review & Documentation	Prepare reports, user manuals, sustainability & risk assessments	2 Weeks
Phase8: Presentation & Feedback	Demonstration, final evaluation, and project submission	1 Week

# Algorithm Details :

## NLP-Based Story Generation Algorithm (Generative AI):

Key Steps:

**1. User Input Collection:** a. Accepts inputs such as:

i. Age Group (e.g., 3–7, 8–12) ii. Preferred Language (English, Hindi, Marathi)

**2. Prompt Formation:**

a. Converts the collected user inputs into a structured and context-rich prompt.

b. Example: "Generate a moral story for an 8-year-old in Marathi."

**3. Story Generation:**

a. The generated prompt is passed to the NLP model.

b. The model generates a complete story aligned with the selected parameters.

**4. Post-Processing:**

a. Adjusts text length and formatting.

b. Ensures clear paragraph separation and age-appropriate vocabulary.

**5. Output Dispatch:**

a. The final story is: i. Displayed on the mobile app screen for reading. ii. Sent to the TTS module for audio narration. This algorithm supports multilingual storytelling and adapts dynamically based on user preferences, ensuring a personalized experience.

# **Text-to-Speech (TTS) Conversion Algorithm:**

## **Key Steps:**

### **1. Story Input:**

- a. Receives story text from the NLP module once generation is complete.

### **2. Language Configuration:**

- a. Automatically detects and configures the language (English, Hindi, or Marathi) for narration.

### **3. Voice Output:**

- a. Invokes the TextToSpeech.speak() method to start reading the story aloud.
- b. Voice pitch and speed can be adjusted for better clarity and listening experience.

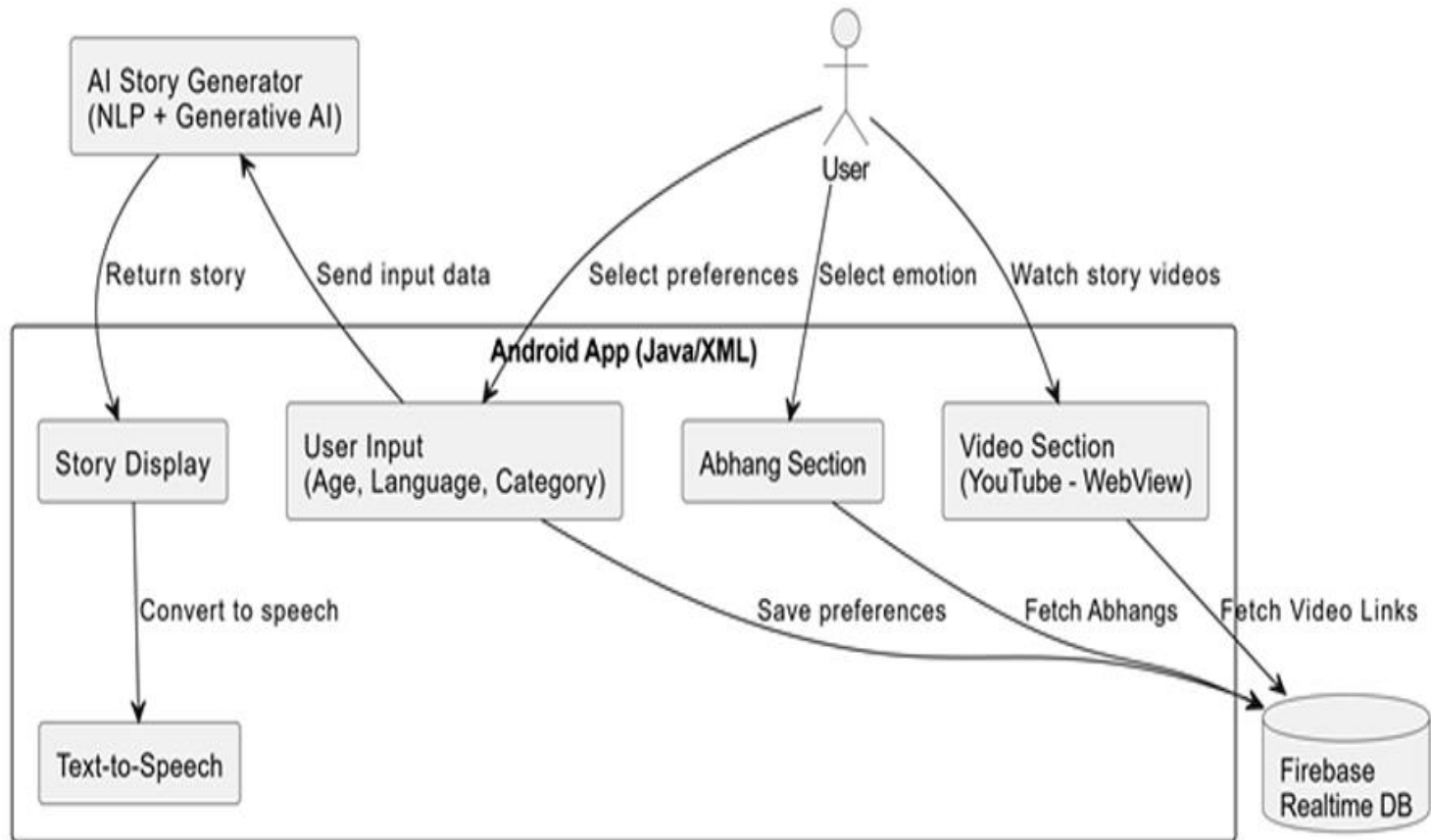
### **4. User Controls:**

- a. Provides buttons to:
  - i. Play
  - ii. Pause
  - iii. Stop narration

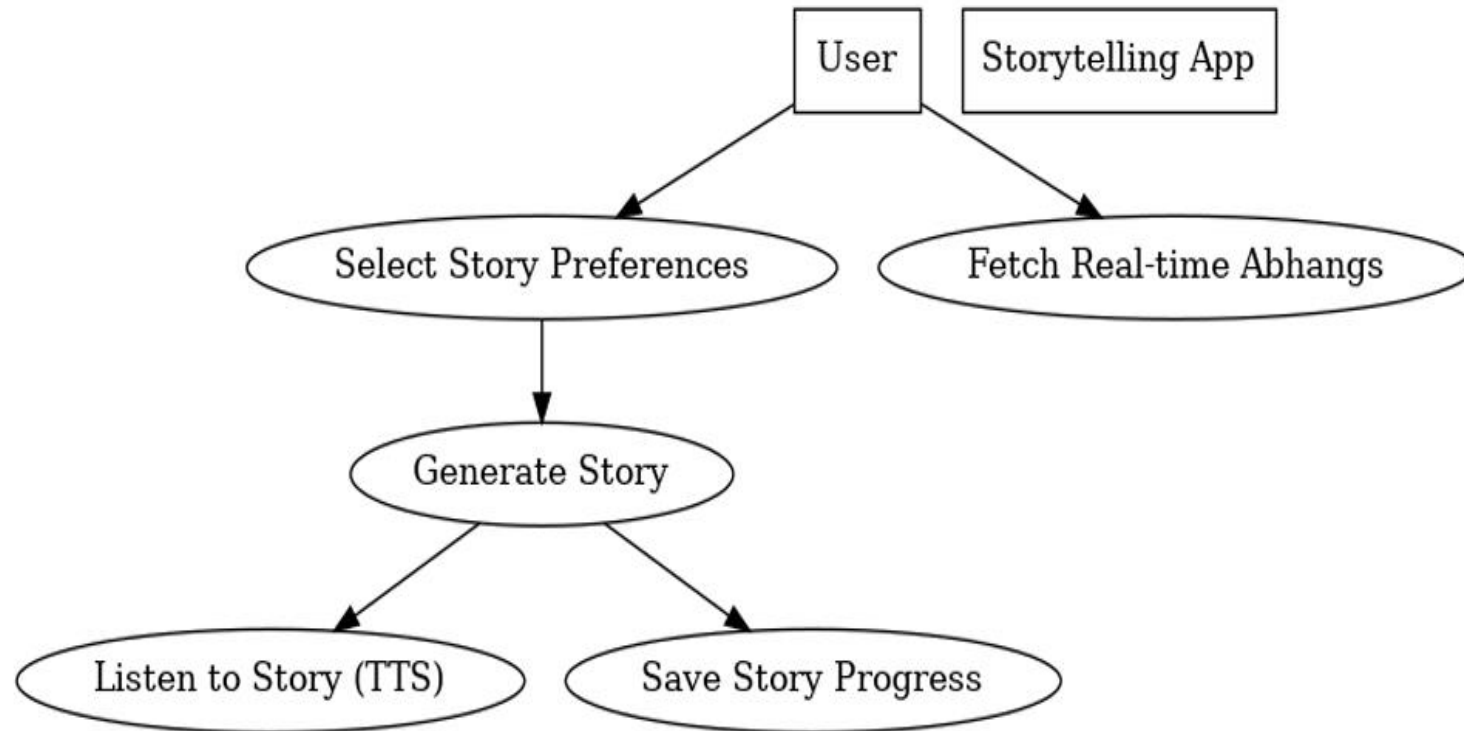
### **5. Resource Management:**

- a. On completion or stop, the TTS engine is properly shut down to free resources and avoid memory leaks.

# System Architecture

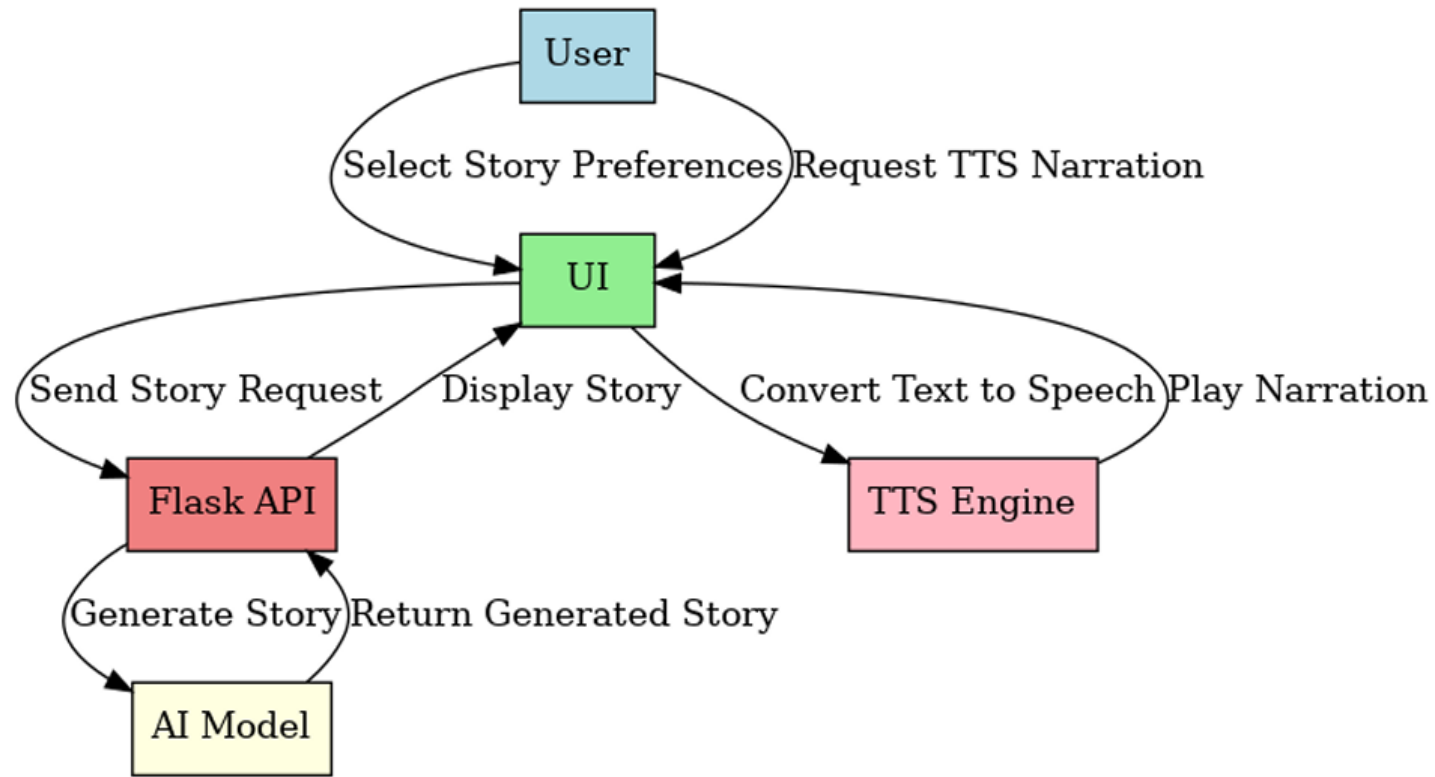


# Use Case Diagram





# Sequence Diagram



# Unit Testing

Case ID	Test Scenario	Test Steps	Expected Result	Actual Result	Status
UT-01	AI Model generates accurate responses	1. Input text prompt 2. Run Gemini Pro 1.5 3. Check output	AI-generated content is contextually relevant	Story generated correctly	Pass
UT-02	API correctly processes AI requests	1. Send request to API 2. Verify response	API receives and processes request	API logged request successfully	Pass
UT-03	Firebase stores and retrieves real-time data	1. Store AI-generated response in Firebase 2. Retrieve data	Data is stored and retrieved instantly	Firebase updated in real time	Pass
UT-04	AI fails to generate content for ambiguous input	1. Input vague prompt 2. Run model	AI provides at least a partial response	Model to generate text	Pass
UT-05	API request to Firebase fails	1. Send request to Firebase and it updates	Data is updated successfully	Database updated	Pass

# System Testing

Test Case ID	Test Scenario	Test Steps	Expected Result	Actual Result	Status
ST-01	Full system runs end-to-end successfully	1. User inputs query 2. AI generates text 3. Response stored in Firebase	AI-generated content is displayed in UI	End-to-end process worked correctly	Pass
ST-02	Multiple users receive unique responses	1. Two users input different prompts 2. Verify AI-generated text	Each user receives unique content	Different responses generated	Pass
ST-03	Response time meets performance goals	1. Measure time from input to response	Response generated in <2 sec	AI responded in 1.7 sec	Pass
ST-04	AI struggles with complex prompts	1. Input complex story scenario 2. Check AI response	AI generates relevant response	Model provided output	Pass
ST-05	Firebase fails to update real-time data	1. AI generates response 2. Flask updates Firebase 3. Check stored data	Data is stored successfully	update in Firebase	Pass

# Conclusion

- Encourages language learning and cultural preservation, creating a respect for traditional storytelling in the digital era.
- Existing AI models, cloud-based deployment, and scalable design enable future expansion.
- Potential uses include education, entertainment, and cross-cultural communication, making the system useful to a wide range of user groups.

# Paper Publication Details

The screenshot shows a Gmail interface on a Windows desktop. The browser address bar displays the email URL: `mail.google.com/mail/u/0/?tab=rm&ogbl#inbox/FMfcgzQZV/wgPMGWMshJHbpkQxjGV5GV`. The Gmail sidebar on the left shows the 'Inbox' with 7,306 messages. The main email content is from 'Microsoft CMT' with the subject '9th International Conference on Control Communication, Computing and Automation : Submission (418) has been created.' The email body contains the following details:

- Hello,**
- The following submission has been created.**
- Track Name:** Cognitive Computing and Machine Learning
- Paper ID:** 418
- Paper Title:** Story Telling Android App
- Abstract:**

This project presents a feature-rich Storytelling Android Application developed using Java, XML, and Firebase Realtime Database. The app leverages Generative AI integrated with Natural Language Processing (NLP) models to automatically generate stories based on the user's age group, preferred language (English, Hindi, Marathi), and category such as Adventure, Moral Values, and more. Each generated story is converted into audio using Text-to-Speech (TTS) technology, offering an engaging and interactive experience for users of all ages. The app includes a special "Abhang Recommend" section that suggests abhangs (devotional or emotional poetry) categorized by themes like Motivation, Happiness, Sadness, and others. A dedicated Video Tab is also integrated, where top Youtube storytelling videos are displayed using Webview, enhancing the app's multimedia capabilities. The real-time data handling with Firebase ensures seamless performance, making the app an intelligent, multilingual storytelling solution ideal for both educational and entertainment purposes.
- Created on:** Sat, 12 Apr 2025 03:36:44 GMT
- Last Modified:** Sat, 12 Apr 2025 03:36:44 GMT
- Authors:**
  - shivanibahiratla@gmail.com (Primary)
- Secondary Subject Areas:** Not Entered

The Windows taskbar at the bottom shows the time as 09:08 on 12-04-2025, with various application icons and system status indicators.

# References

- [1] M. Ghosh, S. Chakraborty and A. Banerjee, "Automatic Story Generation Using Deep Learning Techniques," 2020 International Conference on Computational Intelligence and Communication Technology (CICT), Ghaziabad, India, 2020, pp. 1-6.
- [2] S. V. Patel and R. A. Pandya, "Text-to-Speech Synthesis Using Android Platform," 2019 IEEE International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS), Coimbatore, India, 2019, pp. 1-5.
- [3] T. Nguyen, M. Tran and D. Pham, "Multilingual Natural Language Processing for Low-Resource Languages: A Survey," IEEE Access, vol. 8, pp. 26171–26189, 2020.
- [4] A. R. Dey and P. Dutta, "Firebase as BaaS for Android App Development," 2021 IEEE International Conference on Computational and Intelligent Techniques in Science, Engineering and Technology (CITSET), 2021, pp. 173–177.
- [5] L. Liu and Y. Xu, "YouTube Video Recommendation Using User Preference and Content Analysis," 2021 IEEE International Conference on Big Data and Smart Computing (BigComp), 2021, pp. 219–224.

**Thank you**