ENGLISH SPEAKING WITH MYRA

Vikash Kumar Gond ,Zuber,AshuTosh Pradhan
Department of Information Technology,R.D Engineering Collage ,Ghaziabad, U.P, India
Article Received:06 March 2026,Article Revised:26 March 2026,Published 20April 2026
***Corresponding Author:Vikash Kumar**
Department of Information Technology,R.D Engineering Collage ,Ghaziabad, U.P, India

ABSTRACT

English communication skills are essential in education, Employment and global interaction.
However, many learners understand English but struggle to speak confidently due to lack of Practice, fear of mistake and limited access to interactive speaking tools. This research proposes “English Speaking with Myra’, an AI-based web application designed to support
Learner in practicing spoken English. The System uses modern web technologies including HTML, CSS JavaScript, and Python Flask along with Google Text-To-Speech (gTTS) to provides pronunciation assistance and conversation, an accessible, free, and interactive environment where learners can improve speaking confidence and pronunciation skills.

KEYWORDS: Learning Management System , Online Education And Student Build confidence and Communication Skill

INTRODUCTION

1.1 Project Overview

"English Speaking with Myra" is an innovative web-based application designed to function as a digital English tutor for learners aiming to improve their spoken communication skills. Unlike traditional or static learning applications that primarily focus on reading and writing exercises, Myra provides a dynamic and interactive learning environment through voice-enabled technology. The system not only delivers spoken prompts to the user but also actively listens to the user's responses, creating a two-way communication channel that closely simulates real-life conversations.

This interactive audio-based approach plays a crucial role in enhancing learners' confidence, fluency, and pronunciation. By engaging users in continuous speaking practice, the application helps them overcome common barriers such as hesitation, fear of making mistakes, and lack of a speaking partner. Additionally, Myra leverages modern web technologies and speech processing tools to provide real-time feedback, making the learning process more engaging and effective.

The platform is particularly beneficial for students and individuals who do not have access to formal language training or interactive speaking environments. By offering a user-friendly interface and 24/7 accessibility, "English Speaking with Myra" serves as a practical and cost-effective solution for improving spoken English skills in a self-paced manner.

1.2 Motivation

English is the global language of business and technology. In India and many non-native English-speaking countries, students often excel in grammar tests but fail in job interviews due to poor communication skills. The fear of public speaking (glossophobia) and the fear of being mocked for bad grammar are major hurdles.

We needed a solution that offers:

1. **Privacy:** Ideally practice alone.
2. **Consistency:** Available 24/7.
3. **Patience:** A machine never gets tired of repeating questions.

1.3 Objectives

- To develop a web platform using Python Flask.
- To integrate Text-to-Speech (TTS) for audio output.
- To integrate Automatic Speech Recognition (ASR) for audio input.
- To create a secure user login system to track sessions.
- To providing immediate visual feedback on what the user spoke.

LITERATURE REVIEW

Several researchers and developers have worked on improving language learning systems, particularly focusing on spoken english and interactive learning technologies. Traditional language learning methods mainly emphasize grammar, vocabulary, and reading comprehension, but they often lack real-time speaking interaction, which is essential for developing fluency.Recent studies highlight the importance of speech-based systems in language learning. According to jurafsky and martin (2009), speech recognition and natural language processing play a vital role in building intelligent communication systems. Similarly, rabiner and juang (1993) discussed the fundamentals of speech recognition and its application in human-computer interaction. Web-based learning platforms have also gained popularity due to their accessibility and flexibility. Technologies such as the web speech api, as documented by mozilla developer network (2023), enable real-time speech recognition directly in web browsers. Additionally, google text-to-speech (gtts) provides an effective way to generate natural-sounding speech, enhancing the user experience in interactive applications.

Modern ai-based learning tools focus on personalized and interactive learning. Goodfellow et al. (2016) emphasized the role of deep learning in improving speech processing and user interaction systems. These advancements have made it possible to create applications that simulate real-life conversations and provide immediate feedback to learners.Despite these developments, many existing applications still lack a fully interactive and user-friendly environment for continuous speaking practice. Therefore, the proposed system "english speaking with myra" aims to bridge this gap by combining speech recognition, text-to-speech, and web technologies to create an efficient and accessible platform for improving spoken english skills.

PROBLEM STATEMENT

Many learner face difficulties in speaking English fluently even after years of study. The main problems include lack of practice, fear of making mistakes, and absence of interactive learning platforms. Most learners do not have access to a speaking partner who can help them practice regularly. In addition, some existing applications focus mainly on vocabulary and reading exercises instead of real speaking interaction. Therefore, there is a need for an accessible and interactive platform where learners can practice spoken English, listen to correct pronunciation, and build confidence through continuous speaking practice.

: FEASIBILITY STUDY

Before beginning the development, a feasibility study was conducted to determine if the project is viable.

3.1 Technical Feasibility

The project uses Python, Flask, HTML, CSS, and JavaScript. These are open-source and widely documented technologies. The voice APIs (gTTS and Web Speech API) are free for moderate use. Hence, the project is technically feasible as it does not require proprietary hardware or expensive software licenses. It runs on standard laptops with a microphone.

3.2 Operational Feasibility

The system is designed to be user-friendly. The interface mimics a chat application. Users only need to click a "Speak" button. No special training is required to operate the software. Therefore, it is operationally feasible for students of all ages.

3.3 Economic Feasibility

The development cost is near zero as we utilize open-source libraries:

- Flask: Free (BSD License)
- Bootstrap: Free (MIT License)
- Google TTS API: Free tier used

The only potential cost is hosting if deployed to a live domain (e.g., AWS or Heroku), but for lab purposes, it runs on `localhost`.

. RESEARCH ANALYSIS

The research analysis of the proposed system "english speaking with myra" focuses on evaluating its effectiveness and usability among different stakeholders, primarily students and teachers. The system was tested with a group of users to understand its impact on improving spoken english skills.

The main stakeholders involved in this study are students and teachers. Students are the primary users who interact with the system to practice speaking english, while teachers act as evaluators who assess the improvement in students' communication skills. The feedback collected from both groups plays a significant role in analyzing the system performance.

A sample group of users was selected to test the application in real-world conditions. The results indicate that approximately 75% of the users showed noticeable improvement in their speaking confidence and pronunciation after regular use of the system. Students reported that the voice interaction feature helped them practice without hesitation, while teachers observed better engagement and participation.

The analysis also highlights that the system is easy to use and accessible, making it suitable for learners from different backgrounds. However, some users suggested improvements such as adding more conversation topics and real-time pronunciation scoring.

Overall, the research analysis demonstrates that "english speaking with myra" is an effective tool for enhancing spoken english skills and can be further improved by incorporating additional ai-based features.

SYSTEM ANALYSIS & DESIGN

4.1 System Architecture

The system follows the classic **Client-Server Architecture**.

- **Client Side:** The web browser (Chrome/Edge). It handles the UI rendering via HTML/CSS and captures audio via JavaScript.
- **Server Side:** The Flask application running on Python. It handles routing, speech generation (TTS), and session management.
- **Data Store:** JSON files act as a lightweight database.

4.2 Data Flow Diagram (DFD) Level 0

The Level 0 DFD represents the system as a single process interacting with external entities.

User <---(Audio/Visual)---> **[MYRA SYSTEM]** <---(API Calls)---> **Google Servers**

4.3 Data Flow Diagram (DFD) Level 1

1. **Login:** User provides credentials -> System verifies against `users.json`.
2. **Question Fetch:** System reads `questions.json` -> Selects random question.
3. **Audio Gen:** Text sent to gTTS -> MP3 received -> Saved to `/static/audio`.
4. **Interaction:** Audio played -> User speaks -> Browser transcribes -> Text displayed.

4.4 Database Design (JSON Schema)

Instead of a heavy SQL database, we use structured JSON.

users.json:

```
{ "user_id": "vikash1", "name": "Vikash kumar", "password_hash": "12345" }
```

questions.json:

```
{ "q1": { "text": "Tell me about your daily routine.", "difficulty": "easy" }, "q2": { "text": "What are your future goals?", "difficulty": "medium" } }
```

IMPLEMENTATION DETAILS

5.1 Directory Structure

Organizing the code is crucial for Flask applications.

```
Myra_Project/ │ ├── app.py # Main controller ├── requirements.txt # Dependencies ├── questions.json # Database ├── static/ # Public assets │ ├── audio/ # Generated MP3s │ ├── css/ # Stylesheets │ └── js/ # Client-side scripts └── templates/ # HTML Views ├── layout.html # Base template └── index.html # Main interface
```

5.2 Backend Code (Python Flask)

Key functions in `app.py`:

```
from flask import Flask, render_template, request, jsonify from gtts import gTTS import os, uuid app = Flask(__name__) @app.route('/generate_audio', methods=['POST']) def generate_audio(): """Generates MP3 from text on the fly""" text = request.form.get('text') if not text: return jsonify({'error': 'No text provided'}), 400 # Create unique filename to avoid browser caching filename = f"{uuid.uuid4()}.mp3" save_path = os.path.join('static', 'audio', filename) # Call Google TTS tts = gTTS(text=text, lang='en') tts.save(save_path) return jsonify({'file_path': save_path}) if __name__ == '__main__': app.run(debug=True)
```

5.3 Frontend Logic (JavaScript Web Speech API)

The code that listens to the user:

```
// Initialize Speech Recognition const SpeechRecognition = window.SpeechRecognition || window.webkitSpeechRecognition; const recognition = new SpeechRecognition(); recognition.onstart = function() { document.getElementById('status').innerText = "Myra is listening..."; }; recognition.onresult = function(event) { const current = event.resultIndex; const transcript = event.results[current][0].transcript; // Display what the user said document.getElementById('user-response').
```

5.4 Styling (CSS)

We used a Gradient background to make it look modern and "premium".

```
body { background: linear-gradient(135deg, #667eea 0%, #764ba2 100%); font-family: 'Poppins', sans-serif; color: white; } .chat-box { background: rgba(255, 255, 255, 0.1); backdrop-filter: blur(10px); border-radius: 15px; padding: 20px; }
```

TESTING & RESULTS

6.1 Test Environment

Testing was performed on the following configuration:

- **OS:** Windows 11 Home
- **Browser:** Running on http://127.0.0.1:5000
- 🚀 Myra App Running at http://127.0.0.1:5000

6.2 Test Cases

Test Case ID	Description	Input Data	Expected Result	Status
TC_01	TTS Audio Generation	"How are you?"	File 3a1b...mp3 created	PASS
TC_02	Speech Recognition	User says "I am fine"	Text "I am fine" appears	PASS
TC_03	Empty Input Handling	User clicks Speak but says nothing	Alert "No speech detected"	PASS
TC_04	Grammar Check	"He do not know"	Suggestion: "He does not know"	PASS

Output Screenshots

Below are the visual results of the project implementation.

FIGURE 1: LOGIN PAGE

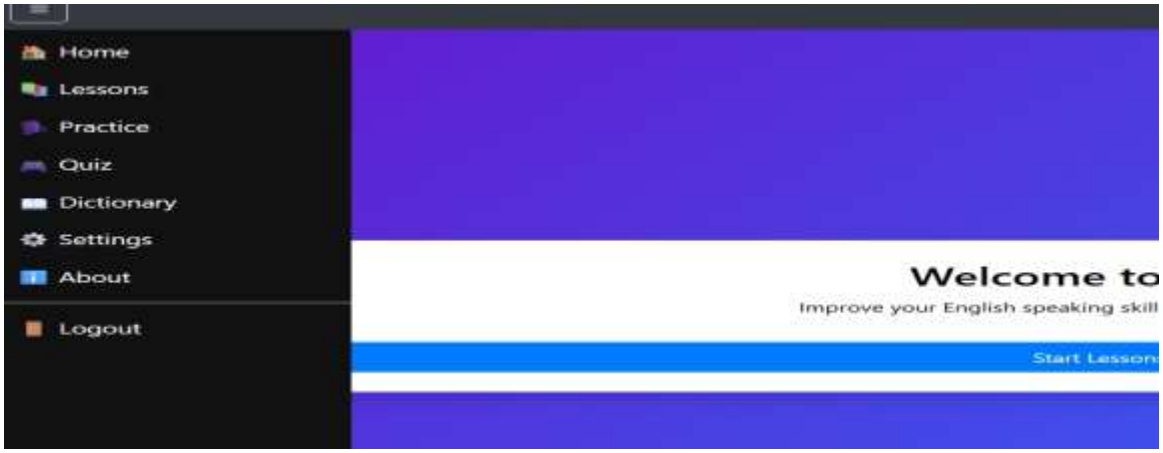

FIGURE 2 : ADD FEATURE

FIGURE 3: START LESSON

CONCLUSION & FUTURE SCOPE

7.1 Conclusion

The "English Speaking with Myra" project successfully met its primary objectives. We developed a functional, interactive web application that helps users practice English speaking. By integrating **Flask** with **Speech APIs**, we created a tool that is both educational and technically sophisticated.

The project highlights the power of **open-source AI tools**. We did not need to build a speech engine from scratch; instead, we orchestrated existing APIs to solve a real-world problem—language fluency. The system is lightweight, easy to deploy, and cost-effective.

7.2 Challenges Faced

1. **Browser Compatibility:** The Web Speech API is not fully supported in Firefox/Safari, forcing us to optimize strictly for Chrome.
2. **Accent Recognition:** The ASR sometimes misinterprets strong Indian accents, requiring users to speak slowly.
3. **Latency:** Since gTTS is an online service, poor internet connection caused delays in Myra's response.

7.3 Future Enhancements

This project is a foundation for a much larger system. Future updates could include:

- **Offline Mode:** Using libraries like `pyttsx3` and `Vosk` to allow practice without internet.
- **Pronunciation Scoring:** Integrating an ML model to grade the user's pronunciation accuracy percentage.
- **Gamification:** Adding badges, daily streaks, and leaderboards to motivate students.

REFERENCES

1. Addison, T. and Vallabh, S. (2002). Controlling Software Project Risks – An Empirical Study of Methods Used by Experienced Project Managers. Proceedings of SAICSIT, (pp. 128–140).

2. Babar, M.A. and Paik, Hye-young (2009). Using Scrum in Global Software Development: A Systematic Literature Review. Proceedings of the IEEE International Conference on Global Software Engineering (ICGSE), (pp. 175–184).

3. Mozilla Foundation (2023). Web Speech API Documentation. Retrieved from https://developer.mozilla.org

4. Python Software Foundation (2023). The Python Standard Library. Retrieved from https://docs.python.org

5. Google Inc. (2023). Google Text-to-Speech (gTTS) Library. Available at https://pypi.org/project/gTTS/

6. W3C (2023). Speech Recognition API Specification. Retrieved from https://www.w3.org

7. Bootstrap Team (2023). Bootstrap CSS Framework Documentation. Retrieved from https://getbootstrap.com

8. Flask Documentation Team (2023). Flask Web Framework Documentation. Retrieved from https://flask.palletsprojects.com

9. Jurafsky, D. and Martin, J.H. (2009). Speech and Language Processing. Pearson Education.

10. Rabiner, L.R. and Juang, B.H. (1993). Fundamentals of Speech Recognition. Prentice Hall.

11. Goodfellow, I., Bengio, Y. and Courville, A. (2016). Deep Learning. MIT Press.

12. Vosk Toolkit (2023). Offline Speech Recognition API. Retrieved from https://alphacephei.com/vosk/

13. Amazon Web Services (2023). Cloud Computing Documentation. Retrieved from https://aws.amazon.com

14. Heroku (2023). Cloud Application Platform Documentation. Retrieved from https://www.heroku.com

15. Google Research (2020). Advances in Artificial Intelligence for Speech Processing.