Grameen Guru

**Presented By:**

**Lakshmi Ambica Vedula-Stanley College of Engineering and Technology for Women-Artificial Intelligence and Data Science**

**OUTLINE**

* **Problem Statement -** Rural students, especially post 10th/12th, lack personalized career guidance and face language barriers accessing online info.
* **Proposed System/Solution - A web-based chatbot that answers student queries in Telugu, Hindi, and English using IBM Watsonx and translation services.**
* **System Development Approach** - Tech Stack:  
  • Python (Flask)  
  • HTML/CSS (Frontend)  
  • IBM Watsonx.ai (LLM)  
  • IBM Cloud Services  
  • DeepTranslator (Language handling)
* **Algorithm & Deployment - User selects language & asks question**
* **Translated → Prompt sent to Watsonx**
* **AI generates answer → Translated back**
* **Response + Chat history shown**
* **Deployed locally using Flask server**
* **Result (Output Image)**

**A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.**

* **Conclusion**
* **Grameena Guru bridges language and tech gaps, helping rural students make informed decisions through AI-powered guidance.**
* **Future Scope**
* **📱 Mobile App • 🗣️ Voice Input • 🎯 Personalized career tracking • 📍 Region-specific opportunities**
* **References**
* **• IBM Watsonx.ai Docs  
  • IBM Cloud Services  
  • DeepTranslator (Python)  
  • Flask Official Docs**

# PROBLEM STATEMENT

In many rural and semi-urban areas, students often lack access to proper career guidance after completing their 10th or 12th grade. Due to limited digital exposure, language barriers, and absence of mentors, students face confusion in choosing suitable courses or career paths.

To overcome this challenge, it is important to provide students with accurate, understandable, and language-inclusive information. With the rise of AI and multilingual technologies, creating a smart, accessible assistant becomes essential. The crucial part is delivering correct answers in the student’s preferred language at the right time, reducing misinformation and indecision.

# PROPOSED SOLUTION

* **🔷 Proposed System: Grameena Guru – Multilingual Career Guidance Chatbot for Rural Students**
* The proposed system aims to address the lack of accessible and localized career guidance in rural areas by creating an AI-powered assistant that can answer students' career-related questions in **Telugu, Hindi, and English**. The assistant uses **IBM Watsonx.ai** hosted on **IBM Cloud** for generating intelligent responses and **translation APIs** to support multilingual interaction.
* **🔸 Components:**
* **Data Collection:**
* Collect and curate commonly asked questions by rural students (e.g., courses after 10th/12th, skill-based jobs, etc.).
* Gather expert answers and guidance in simple language tailored for regional understanding.
* **Natural Language Processing (NLP):**
* Integrate **IBM Watsonx Foundation Models** for understanding queries and generating answers.
* Use a **translation layer** to convert queries into English for processing, and then translate answers back into Telugu or Hindi as per user selection.
* **Multilingual Interaction:**
* Accept inputs in **Telugu, Hindi, or English** and respond in the same language.
* Use language codes and auto-translation to maintain accuracy and cultural relevance.
* **Frontend UI:**
* A **simple, mobile-friendly interface** where users can type questions, select language, and view response and chat history.
* Responses and past interactions are stored session-wise for a consistent user experience.
* **Deployment:**
* Deploy the app using **Flask on IBM Cloud**, ensuring availability and scalability.
* Make the chatbot accessible via web browsers with minimal data requirements for rural users.
* **Evaluation:**
* Evaluate chatbot performance based on response quality, language accuracy, and user feedback.
* Continuously update training prompts and translation mapping for improved accuracy and relevance.
* **🔸 Result:**
* A working chatbot named **Grameena Guru** that provides accurate, personalized career advice in 3 languages.
* Enables rural students to make informed decisions after 10th/12th with confidence.

# SYSTEM APPROACH

The **system approach** outlines the strategy, technologies, and components used to develop the multilingual AI chatbot. The goal is to ensure a smooth and accessible experience for rural students seeking career guidance.

**🔸 System Requirements:**

* **Operating System:** Windows/Linux (for development and deployment)
* **Backend Framework:** Flask (Python micro web framework)
* **Cloud Platform:** IBM Cloud (for deploying Watsonx.ai models and app hosting)
* **Frontend:** HTML, CSS (for a simple, responsive UI)
* **Language Support:** English, Telugu, Hindi
* **Internet Connection:** Required for API calls and model responses

**🔸 Libraries/Packages Required:**

* flask – for building the web interface and handling routes
* ibm\_watsonx\_ai – to access IBM Watsonx Foundation Models
* deep-translator – to perform multilingual translations between English, Hindi, and Telugu
* dotenv – to manage sensitive keys and project credentials (optional for security)
* httpx==0.23 – compatible version required by IBM SDK
* gunicorn (optional) – for production-level deployment on IBM Cloud or Render

# ALGORITHM & DEPLOYMENT

# The project utilizes IBM Watsonx.ai Foundation Model (Granite-3B/8B Instruct) for natural language understanding and response generation. This large language model (LLM) is selected for its high accuracy, multilingual support, and ability to understand complex queries from students in English, Telugu, and Hindi.

# Additionally, the Deep Translator API is integrated for accurate input/output translation, enabling interaction in native languages while the core processing happens in English.

# 🔸 Data Input:

# User question (in English, Telugu, or Hindi)

# Selected input language

# Translated version (to English, for processing)

# Optional: Metadata like timestamp (for history storage)

# 🔸 Processing Pipeline:

# User Input Handling: The user submits a question and selects a language (English/Hindi/Telugu).

# Translation (if needed): Non-English inputs are translated to English using deep-translator.

# Watsonx.ai Inference: The translated input is passed to the IBM Foundation Model to generate a relevant, career-guiding response.

# Translation (Back to User Language): The output from the AI model is translated back to the user’s selected language.

# Storage: Question and response pairs are stored in server memory (or database in future) for viewing history.

# 🔸 Deployment:

# Platform: IBM Cloud (or local server during testing)

# Frontend: HTML & CSS-based responsive UI

# Backend: Python (Flask)

# Model Hosting: IBM Watsonx.ai (Granite Model)

# Deployment Tools: Gunicorn (for scalable server deployment)

# RESULT

The **Grameena Guru** chatbot was successfully developed and tested using the **IBM Watsonx.ai Granite model** integrated with **Deep Translator API**. It supports multilingual question handling and provides career suggestions for students in **Telugu, Hindi, and English**.

**✅ Performance Highlights:**

* **Accuracy:** The responses generated were contextually accurate and relevant to education- and career-related questions.
* **Language Understanding:** Successfully handled input and output in 3 languages.
* **User Experience:** Real-time responses with intuitive and user-friendly frontend.
* **History Tracking:** Maintains previous Q&A pairs in session.

# CONCLUSION

The **Grameena Guru** project successfully demonstrates the use of **AI and multilingual processing** to guide rural students in their career decisions. The chatbot provides accurate, context-aware responses in **Telugu, Hindi, and English**, ensuring inclusivity for students from different linguistic backgrounds.

This solution highlights the **power of IBM Watsonx.ai** for natural language generation and shows how **language translation tools** can bridge the communication gap in rural education.

**✅ Key Outcomes:**

* Enabled **real-time AI responses** to education and career-related queries.
* Supported **three-language interaction**, increasing accessibility.
* Maintained **chat history** for reference and continuity.

**⚠️ Challenges Faced:**

* Handling translation inconsistencies in regional languages.
* Ensuring accurate context in multi-language questions.

**🔧 Potential Improvements:**

* Adding **voice-based input/output** for low-literacy users.
* Expanding to support **regional dialects**.
* Training a **custom fine-tuned model** for rural educational datasets.

Grameena Guru sets the foundation for AI-based educational support in rural India, promoting **awareness, opportunity, and guidance** for underserved students.

**FUTURE SCOPE**

The **Grameena Guru chatbot** presents vast potential for further development and scalability to benefit more students across regions.

**🔮 Planned Enhancements:**

* **Voice Integration**: Implementing speech-to-text and text-to-speech features for users with low literacy.
* **Mobile App Deployment**: Creating a lightweight Android app for wider rural reach.

**🌍 Regional Expansion:**

* Extend support for **more Indian languages** and dialects to cover students from other states.
* Deploy in **multiple districts and states**, integrated with local education boards.

**📊 Data & Intelligence:**

* Integrate with **career portals, job data, and government schemes** for real-time recommendations.
* Build a **recommendation engine** using past questions and outcomes.

**🚀 Emerging Technology Adoption:**

* Utilize **edge computing** to make the system work offline or in low-connectivity areas.
* Experiment with **advanced LLMs** and domain-specific AI models for higher accuracy in responses.

These future improvements will strengthen the chatbot’s **impact on rural education** and ensure **sustained scalability and accessibility** across India.

# REFERENCES

1. IBM Watsonx.ai Documentation – IBM Cloud  
   https://www.ibm.com/cloud/watsonx-ai
2. Flask Web Framework – Official Documentation  
   https://flask.palletsprojects.com/
3. DeepL and Google Translate APIs – Language Translation Tools  
   https://www.deepl.com/docs-api  
   https://cloud.google.com/translate
4. Career Guidance and Counselling in Rural India – Research Article  
   https://ijcrt.org/papers/IJCRT2105285.pdf
5. Hugging Face Transformers – Large Language Models  
   https://huggingface.co/docs/transformers/index
6. HTML, CSS, and Bootstrap – Frontend Web Development  
   https://developer.mozilla.org/en-US/docs/Web
7. Python Official Documentation – Language Reference  
   https://docs.python.org/3/
8. Articles on Rural Education & Digital Empowerment



A blue and green rectangular background

AI-generated content may be incorrect.

## 

A pink cover with text

AI-generated content may be incorrect.

## IBM CERTIFICATIONS

## A screenshot of a computer AI-generated content may be incorrect.

**THANK YOU**