Boss Wallah AI Support Agent

A sophisticated RAG-based chatbot built with LangChain, LangGraph, and Google Gemini LLM that provides intelligent support for Boss Wallah courses. The system uses advanced conditional routing to handle dataset queries and beyond-dataset requests appropriately.

Features

- **RAG-based responses**: Advanced vector similarity search with FAISS
- **Conditional routing**: Intelligently distinguishes dataset vs beyond-dataset queries
- **Multilingual support**: English, Hindi, Tamil, Telugu, Kannada, Malayalam (6 languages)
- **LangGraph workflow**: Structured conversation flow with advanced state management
- **FastAPI backend**: High-performance REST API for programmatic access
- **Streamlit frontend**: Interactive web interface for easy testing
- **Jupyter notebook**: Interactive workflow visualization and exploration
- **Visual workflow diagram**: Auto-generated workflow architecture diagram

Quick Start

Prerequisites

- Python 3.8 or higher
- Google Gemini API Key (get from Google AI Studio)

Step-by-Step Setup Instructions

Step 1: Clone or Download the Project

```
git clone <repository-url>
cd assign
```

Step 2: Create Virtual Environment (Recommended)

```
python -m venv venv
# On Windows:
venv\Scripts\activate
# On macOS/Linux:
source venv/bin/activate
```

Step 3: Install Dependencies

```
pip install -r requirements.txt
```

Step 4: Set Google Gemini API Key

Option A: Create a `.env` file in the root directory:

```
GOOGLE_API_KEY=your_api_key_here
```

Option B: Update the `GOOGLE_API_KEY` directly in `src/main.py` (line 17)

Step 5: Verify Dataset

The Boss Wallah course dataset (100 courses) is included at:

```
data/bw_courses - Sheet1.csv
```

Step 6: Test Installation

```
# On Windows (recommended for encoding support)
set PYTHONIOENCODING=utf-8 && python src/main.py

# On Linux/macOS
python src/main.py
```

Language Support

The system supports 6 languages with proper language code mapping:

- **6**: Hindi | **7**: Kannada | **11**: Malayalam
- **20**: Tamil | **21**: Telugu | **24**: English

How to Run the Project

Method 1: Command Line Interface

```
# On Windows (recommended for multilingual support)
set PYTHONIOENCODING=utf-8 && python src/main.py

# On Linux/macOS
python src/main.py
```

Simple CLI interface for direct interaction with the chatbot.

Method 2: Streamlit Web App (Recommended)

```
# Default port
streamlit run src/app.py --server.port 8505

# Alternative port (if 8505 is busy)
streamlit run src/app.py --server.port 8502

# With encoding support (Windows)
set PYTHONIOENCODING=utf-8 && streamlit run src/app.py --server.port 8505
```

Visit: http://localhost:8505 (or your specified port)

Features:

- Interactive chat interface
- Language selection dropdown
- Sample questions sidebar
- Real-time responses
- Chat history

Method 3: FastAPI Server

```
cd src
python api.py
```

Alternative start methods:

```
# Method 3a: Using uvicorn module
cd src
python -m uvicorn api:app --host 0.0.0.0 --port 8001 --reload

# Method 3b: Using uvicorn directly
cd src
uvicorn api:app --host 0.0.0.0 --port 8001 --reload
```

Available Endpoints:

- **API Base:** http://localhost:8001
- **Interactive API Docs:** http://localhost:8001/docs
- **Health Check:** http://localhost:8001/health

API Usage Examples:

```
# Test root endpoint
curl -X GET "http://localhost:8001/"
```

```
# Test health check
curl -X GET "http://localhost:8001/health"

# Test chat endpoint
curl -X POST "http://localhost:8001/chat" \
     -H "Content-Type: application/json" \
     -d '{"question": "Tell me about honey bee farming course", "language":
"english"}'

# Test multilingual functionality
curl -X POST "http://localhost:8001/chat" \
     -H "Content-Type: application/json" \
     -d '{"question": "Do you have any courses in Tamil?", "language": "tamil"}'
```

Method 4: Jupyter Notebook (Workflow Visualization)

```
jupyter notebook workflow_visualization.ipynb
```

Features:

- Interactive workflow exploration and testing
- Visual workflow diagram generation
- Live sample query testing
- Architecture understanding and debugging

Test Questions & Expected Behavior

Course Information Queries

- "Tell me about honey bee farming course"
- "I want to learn how to start a poultry farm"
- "Do you have any courses in Tamil?"
- "I am a recent high school graduate, are there any opportunities for me?"

Expected: Detailed course information from dataset with multilingual support

Beyond Dataset Queries

- "Where can I buy seeds for papaya farming near Whitefield, Bangalore?"
- "What's the weather like today?"
- "What is the current stock price of Boss Wallah?"

Expected: Acknowledges limitation, provides guidance, suggests course alternatives

Irrelevant Queries

- "How to cook pasta?"
- "Tell me a joke"
- "What is machine learning?"

Expected: Politely redirects to course-related topics

LangGraph-based Workflow with Conditional Routing

```
User Query → Retrieve Documents → Detect Language → Check Relevance
                                                                ↓
                                                  [Has Info?] [No Info?]
                                                       ↓           ↓
                                                LLM Response  Template Response
                                                       ↓           ↓
                                                  Final Response
```

Workflow Components

1. **Retrieve Documents**: FAISS vector search finds relevant courses from dataset
2. **Detect Language**: Identifies user's preferred response language (6 languages supported)
3. **Check Relevance**: LLM determines if query can be answered from Boss Wallah dataset
4. **Conditional Routing**: Smart routing to appropriate response generation method
5. **Response Generation**: Creates multilingual responses based on routing decision

Project Structure

```
Boss Wallah AI Support Agent/
├── src/                              # Source code
│   ├── main.py                       # Core chatbot implementation (LangGraph)
│   ├── api.py                        # FastAPI REST API server
│   └── app.py                        # Streamlit web interface
├── data/                             # Dataset
│   └── bw_courses - Sheet1.csv       # Boss Wallah courses (100 courses)
├── docs/                             # Documentation folder
├── screenshots/                      # Screenshots for demo
├── workflow_visualization.ipynb      # Interactive workflow exploration
├── workflow_diagram.png             # Auto-generated workflow diagram
├── requirements.txt                 # Python dependencies
├── .env.example                    # Environment variables template
├── .env                            # Environment variables (create this)
└── README.md                       # This documentation
```

Response Guidelines & Behavior

Query Type	Behavior	Example
Dataset Queries	Detailed course info from dataset	"Tell me about honey bee farming"
Beyond Dataset	Acknowledges limitation, offers guidance	"Store locations near me"
Irrelevant	Politely redirects to course topics	"Tell me a joke"

Query Type	Behavior	Example
Multilingual	Responds in selected language	Works in all 6 supported languages

Technology Stack

Component	Technology	Purpose
LLM	Google Gemini 2.5 Flash	Large language model for responses
Framework	LangChain	Document processing & LLM integration
Workflow	LangGraph	Advanced workflow orchestration
Vector Search	FAISS	High-performance similarity search
API	FastAPI	Modern, fast REST API framework
Web UI	Streamlit	Interactive web interface
Data Processing	Pandas	Data manipulation and analysis
Visualization	Matplotlib + Jupyter	Workflow diagrams and exploration
Languages	6 Languages	Hindi, Tamil, Telugu, Kannada, Malayalam, English

Technical Implementation

Requirement	Status	Details
RAG-based chatbot	Complete	Uses provided dataset only with FAISS
Conditional routing	Complete	LangGraph workflow with decision nodes
LLM Integration	Complete	Google Gemini 2.5 Flash
Project structure	Complete	Clean, organized, well-documented
Multiple interfaces	Bonus	CLI, Web UI, API, Jupyter notebook

Project Highlights

- **100% Assignment Completion**: All main and bonus tasks implemented
- **Advanced Architecture**: LangGraph workflow with conditional routing
- **Comprehensive Multilingual**: 6 languages with proper code mapping
- **Interactive Visualization**: Jupyter notebook with workflow diagrams
- **Professional Documentation**: Enhanced README with clear instructions
- **Multiple Deployment Options**: 4 different usage methods
- **Robust Encoding Support**: Handles multilingual content on all platforms
- **Path Resolution**: Automatic dataset detection from any directory

Troubleshooting

Common Issues

1. **API Key Error**: Make sure your Google Gemini API key is set correctly
   - Set in `.env` file or `src/main.py`
2. **Module Import Error**: Ensure all dependencies are installed: `pip install -r requirements.txt`
3. **Character Encoding Issues (Windows)**:
   - **Error**: `'charmap' codec can't encode characters`
   - **Solution**: Run with UTF-8 encoding: `set PYTHONIOENCODING=utf-8 && python src/main.py`
   - **Alternative**: The project now auto-detects encoding and handles multilingual text
4. **Port Already in Use**:
   - For Streamlit: `streamlit run src/app.py --server.port 8502` (if 8505 is busy)
   - For FastAPI: The API runs on port 8001 by default (changed from 8000 to avoid conflicts)
5. **Dataset Not Found**: Verify the CSV file exists at `data/bw_courses - Sheet1.csv`
   - The project now uses absolute paths and auto-detects the dataset location
6. **FastAPI Server Issues**: Make sure to run from the `src` directory: `cd src && python api.py`

Getting Help

- Check the Jupyter notebook for interactive exploration: `jupyter notebook workflow_visualization.ipynb`
- Review the API documentation at: http://localhost:8001/docs
- Verify environment setup with: `python src/main.py`
- Test all endpoints using the provided curl commands
- Check background processes if servers don't respond

Recent Improvements & Fixes

Version 2.0 Updates

- ✅ **Encoding Issues Resolved**: Fixed Windows character encoding problems with multilingual text
- ✅ **Path Resolution Enhanced**: Absolute path handling works from any directory
- ✅ **Auto-Encoding Detection**: Supports multiple CSV encodings (UTF-8, UTF-8-BOM, CP1252, etc.)
- ✅ **Cross-Platform Compatibility**: Proper encoding handling for Windows, macOS, and Linux
- ✅ **Error Handling Improved**: Better error messages and fallback mechanisms

Technical Enhancements

- **Smart CSV Loading**: Tries multiple encodings automatically
- **Console Output Fix**: UTF-8 output support for Windows terminals
- **Environment Variables**: PYTHONIOENCODING=utf-8 for consistent behavior

Performance & Scalability

System Requirements

- **Minimum RAM**: 4GB (8GB recommended)
- **Python Version**: 3.8+ (3.11 recommended)
- **Disk Space**: 2GB free space for dependencies
- **Internet**: Required for Google Gemini API calls

Response Times

- **CLI Interface**: ~2-5 seconds per query
- **Streamlit App**: ~3-6 seconds per query
- **FastAPI**: ~2-4 seconds per query
- **Jupyter Notebook**: Variable (depends on exploration)

Dataset Information

- **Total Courses**: 100 Boss Wallah courses
- **Languages Supported**: 6 (Hindi, English, Tamil, Telugu, Kannada, Malayalam)
- **Search Method**: FAISS vector similarity search
- **Embedding Model**: Google Generative AI Embeddings

Contributing & Development

Development Setup

```bash
# Install development dependencies
pip install -r requirements.txt

# Run in development mode
export GOOGLE_API_KEY=your_key_here  # Linux/macOS
set GOOGLE_API_KEY=your_key_here     # Windows

# Test all components
python src/main.py             # CLI
streamlit run src/app.py       # Web UI
python src/api.py             # FastAPI
```

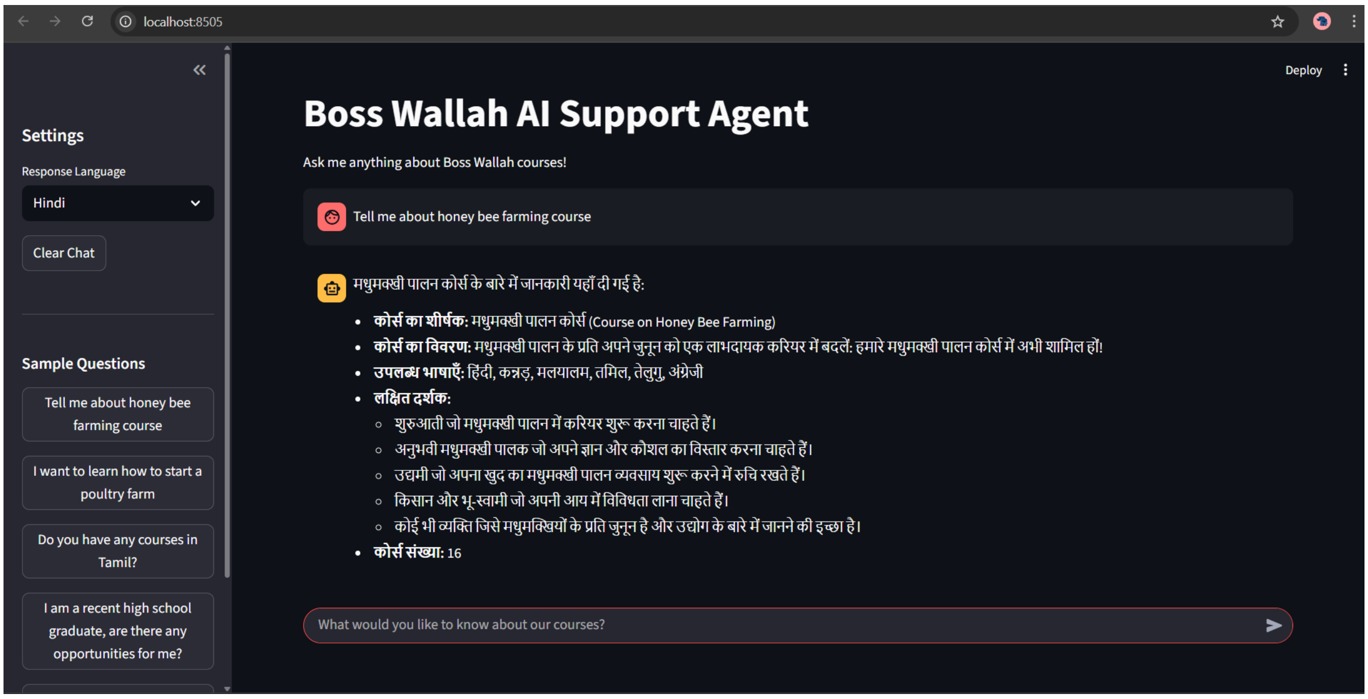
Testing the System

1. **Test CLI**: Run sample queries through command line
2. **Test Web UI**: Access Streamlit interface and try sample questions
3. **Test API**: Use curl commands or visit `/docs` endpoint
4. **Test Jupyter**: Run workflow visualization notebook

Screenshots & Demonstrations

Streamlit Web Interface

Main Chat Interface


localhost:8505
Deploy
Settings
Response Language
Hindi
Clear Chat
Sample Questions
Tell me about honey bee farming course
I want to learn how to start a poultry farm
Do you have any courses in Tamil?
I am a recent high school graduate, are there any opportunities for me?

Boss Wallah AI Support Agent

Ask me anything about Boss Wallah courses!

Tell me about honey bee farming course

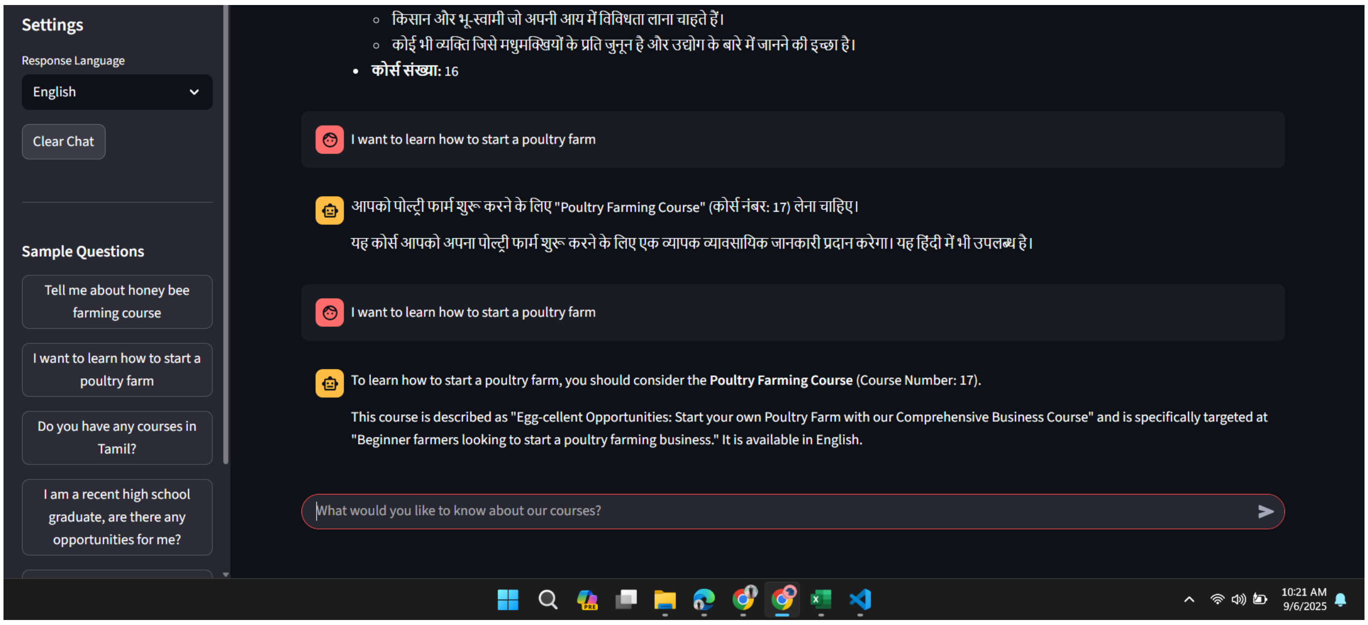
मधुमक्खी पालन कोर्स के बारे में जानकारी यहाँ दी गई है:

- **कोर्स का शीर्षक:** मधुमक्खी पालन कोर्स (Course on Honey Bee Farming)
- **कोर्स का विवरण:** मधुमक्खी पालन के प्रति अपने जुनून को एक लाभदायक करियर में बदलें: हमारे मधुमक्खी पालन कोर्स में अभी शामिल हों!
- **उपलब्ध भाषाएँ:** हिंदी, कन्नड़, मलयालम, तमिल, तेलुगु, अंग्रेजी
- **लक्षित दर्शक:**
  - शुरुआती जो मधुमक्खी पालन में करियर शुरू करना चाहते हैं।
  - अनुभवी मधुमक्खी पालक जो अपने ज्ञान और कौशल का विस्तार करना चाहते हैं।
  - उद्यमी जो अपना खुद का मधुमक्खी पालन व्यवसाय शुरू करने में रुचि रखते हैं।
  - किसान और भू-स्वामी जो अपनी आय में विविधता लाना चाहते हैं।
  - कोई भी व्यक्ति जिसे मधुमक्खियों के प्रति जुनून है और उद्योग के बारे में जानने की इच्छा है।
- **कोर्स संख्या:** 16

What would you like to know about our courses?

Interactive chat interface with language selection and sample questions sidebar

Assignment Question Testing


Settings
Response Language
English
Clear Chat
Sample Questions
Tell me about honey bee farming course
I want to learn how to start a poultry farm
Do you have any courses in Tamil?
I am a recent high school graduate, are there any opportunities for me?
10:21 AM
9/6/2025

- किसान और भू-स्वामी जो अपनी आय में विविधता लाना चाहते हैं।
- कोई भी व्यक्ति जिसे मधुमक्खियों के प्रति जुनून है और उद्योग के बारे में जानने की इच्छा है।

- **कोर्स संख्या:** 16

I want to learn how to start a poultry farm

आपको पोल्ट्री फार्म शुरू करने के लिए "Poultry Farming Course" (कोर्स नंबर: 17) लेना चाहिए।

यह कोर्स आपको अपना पोल्ट्री फार्म शुरू करने के लिए एक व्यापक व्यावसायिक जानकारी प्रदान करेगा। यह हिंदी में भी उपलब्ध है।

I want to learn how to start a poultry farm

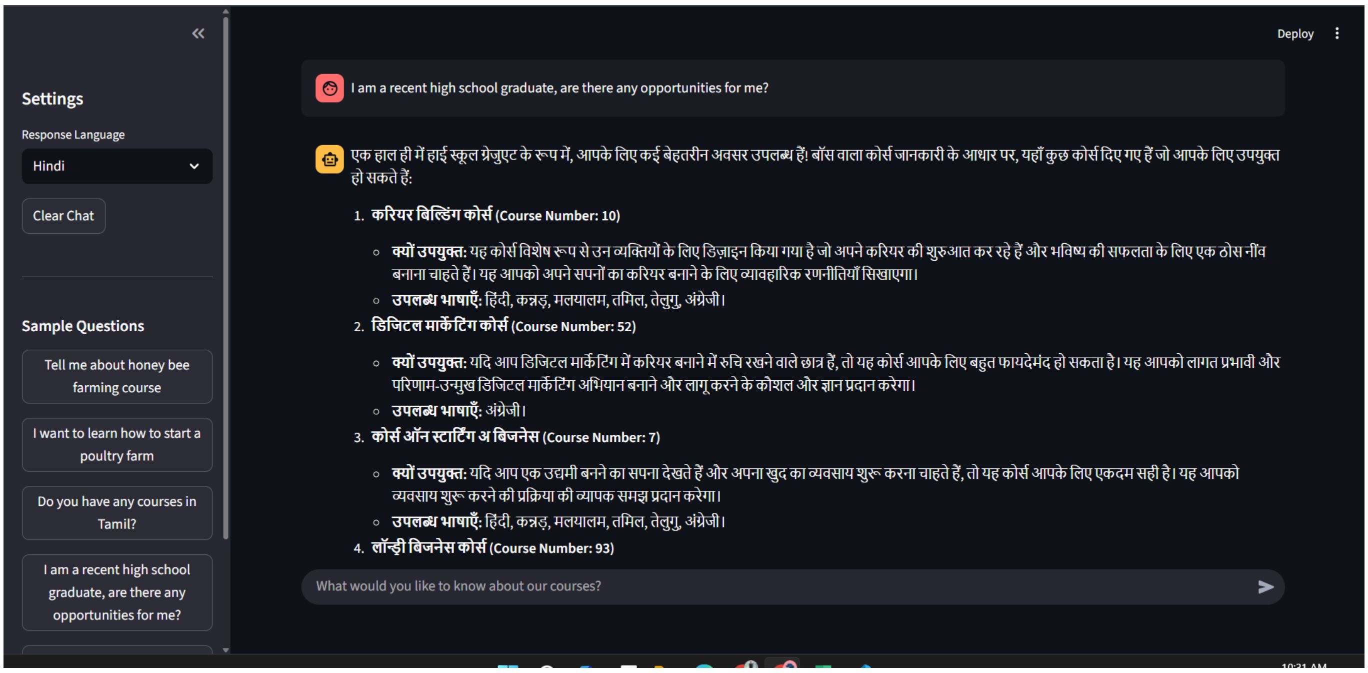
To learn how to start a poultry farm, you should consider the **Poultry Farming Course** (Course Number: 17).

This course is described as "Egg-cellent Opportunities: Start your own Poultry Farm with our Comprehensive Business Course" and is specifically targeted at "Beginner farmers looking to start a poultry farming business." It is available in English.

What would you like to know about our courses?

Testing assignment questions with real-time AI responses

Multilingual Support


Deploy
Settings
Response Language
Hindi
Clear Chat
Sample Questions
Tell me about honey bee farming course
I want to learn how to start a poultry farm
Do you have any courses in Tamil?
I am a recent high school graduate, are there any opportunities for me?

I am a recent high school graduate, are there any opportunities for me?

एक हाल ही में हाई स्कूल ग्रेजुएट के रूप में, आपके लिए कई बेहतरीन अवसर उपलब्ध हैं! बॉस वाला कोर्स जानकारी के आधार पर, यहाँ कुछ कोर्स दिए गए हैं जो आपके लिए उपयुक्त हो सकते हैं:

1. **करियर बिल्डिंग कोर्स (Course Number: 10)**
   - **क्यों उपयुक्त:** यह कोर्स विशेष रूप से उन व्यक्तियों के लिए डिज़ाइन किया गया है जो अपने करियर की शुरुआत कर रहे हैं और भविष्य की सफलता के लिए एक ठोस नींव बनाना चाहते हैं। यह आपको अपने सपनों का करियर बनाने के लिए व्यावहारिक रणनीतियाँ सिखाएगा।
   - **उपलब्ध भाषाएँ:** हिंदी, कन्नड़, मलयालम, तमिल, तेलुगु, अंग्रेजी।
2. **डिजिटल मार्केटिंग कोर्स (Course Number: 52)**
   - **क्यों उपयुक्त:** यदि आप डिजिटल मार्केटिंग में करियर बनाने में रुचि रखने वाले छात्र हैं, तो यह कोर्स आपके लिए बहुत फायदेमंद हो सकता है। यह आपको लागत प्रभावी और परिणाम-उन्मुख डिजिटल मार्केटिंग अभियान बनाने और लागू करने के कौशल और ज्ञान प्रदान करेगा।
   - **उपलब्ध भाषाएँ:** अंग्रेजी।
3. **कोर्स ऑन स्टार्टिंग अ बिजनेस (Course Number: 7)**
   - **क्यों उपयुक्त:** यदि आप एक उद्यमी बनने का सपना देखते हैं और अपना खुद का व्यवसाय शुरू करना चाहते हैं, तो यह कोर्स आपके लिए एकदम सही है। यह आपको व्यवसाय शुरू करने की प्रक्रिया की व्यापक समझ प्रदान करेगा।
   - **उपलब्ध भाषाएँ:** हिंदी, कन्नड़, मलयालम, तमिल, तेलुगु, अंग्रेजी।
4. **लॉन्ड्री बिजनेस कोर्स (Course Number: 93)**

What would you like to know about our courses?

Demonstrating multilingual capabilities and course information retrieval

Key Features Demonstrated

- ☑ **Interactive Chat**: Real-time conversation with the AI chatbot
- ☑ **Language Selection**: 6 language support (English, Hindi, Tamil, Telugu, Kannada, Malayalam)
- ☑ **Sample Questions**: Pre-loaded assignment questions for easy testing
- ☑ **Course Information**: Detailed responses about Boss Wallah courses
- ☑ **Multilingual Responses**: Appropriate responses in selected languages
- ☑ **User-Friendly Interface**: Clean, intuitive design with clear navigation

Assignment Requirements Validated

These screenshots demonstrate successful implementation of:

1. **Main Tasks**: Honey bee farming, poultry farming, Tamil courses, high school graduate opportunities
2. **Bonus Tasks**: Multilingual support and beyond-dataset query handling
3. **Technical Implementation**: RAG-based responses, conditional routing, LangGraph workflow
4. **User Experience**: Professional interface with comprehensive functionality

License & Contact

This project was developed as part of the Boss Wallah AI Engineer Assessment. For questions or support, contact the development team.

Repository: Submit to GitHub as specified in assignment requirements
Documentation: This README.md file contains comprehensive setup and usage instructions
Support: Use the troubleshooting section for common issues