Arabic Learning Assistant Platform - Complete Guide

Table of Contents

1. Prerequisites
2. Setup Instructions
3. Running the Application
4. Feature Verification
5. Troubleshooting
6. Deployment
7. Development Workflow

Prerequisites

- Python 3.9+ installed
- Stable internet connection
- YouTube video URLs with Arabic content
- Terminal/Command Prompt access

Setup Instructions

1. Clone Repository

```
git clone https://github.com/yourusername/arabic-learning-
        assistant.git
cd arabic-learning-assistant
```

2. Create Virtual Environment

```
python -m venv venv
# Activate environment:
# Linux/Mac: source venv/bin/activate
# Windows: venv\Scripts\activate
```

3. Install Dependencies

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

4. Configure Directories

```
mkdir -p transcripts structured_data
```

5. Verify Structure

```
.
├── backend/
├── frontend/
├── transcripts/
├── structured_data/
├── main.py
└── requirements.txt
```

Running the Application

```
streamlit run main.py
```

First-Time Setup Checklist:

- Open browser to `localhost:8501`
- Select “1. Chat with Arabic GPT” in sidebar
- Test with: “ما معنى كلمة ‘مكتبة’؟”
- Verify response about library meaning appears

Feature Verification

Feature	Test Method	Expected Result
Basic Chat	Ask “كيف أقول شكرًا بالإنجليزية؟”	English translation of “thank you”
Transcript Download	Paste Arabic YouTube URL in Raw Transcript	Text appears in left panel
RAG System	Query “ما هي عاصمة السعودية؟”	Riyadh mentioned in response
Structured Data	Check `structured_data/` after processing	CSV/JSON files created

---

Troubleshooting

Common Issues:

Transcript Download Fails

- Verify video has captions
- Try different YouTube URL
- Check network connection

Slow Model Responses

Modify in `backend/chat.py`:

```
def generate_response(...):
    inference_config = {"max_length": 50, "temperature": 0.7}
```

ChromaDB Errors

```
pip install --upgrade chromadb
rm -rf chroma-data  # Reset vector store
```

Missing Dependencies

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

Deployment

Streamlit Cloud Deployment

1. Create account at streamlit.io
2. Connect GitHub repository
3. Configure settings:
    - Python version: 3.9
    - Main file path: `main.py`
4. Deploy!

Requirements:

- Repository must be public for free tier
- Keep transcripts under 100MB

Development Workflow

```
graph TD
    A[Local Development] --> B[Test Features]
    B --> C{Passed All Tests?}
    C -->|Yes| D[Commit Changes]
    C -->|No| E[Debug Issues]
    D --> F[Push to GitHub]
    F --> G[Cloud Deployment]
    G --> H[User Feedback]
    H --> A
```

Important Notes

- All components use free-tier resources
- No API keys required
- Data persists only during active session
- Reset environment by deleting venv and recreating