🎓 CMU-Africa Information Assistant - Setup Checklist

Pre-Installation Checklist

- [ ] Python 3.9+ installed (`python --version`)
- [ ] pip installed (`pip --version`)
- [ ] Git installed (if cloning from repository)
- [ ] 4GB+ RAM available
- [ ] Internet connection active

API Keys Required

- [ ] **Pinecone API Key** - Get it here (https://www.pinecone.io/)
- Sign up for free account
- Create API key in dashboard
- Note your environment (e.g., us-east-1)
- [ ] **OpenAI API Key** - Get it here (https://platform.openai.com/)
- Sign up for account
- Add billing information
- Create API key in API section

Installation Steps

1. Environment Setup

```
- [ ] Create virtual environment: python -m venv venv
- [ ] Activate virtual environment:
      Linux/Mac: source venv/bin/activate
      Windows: venv\Scripts\activate
- [ ] Verify activation: which python (should show venv path)
```

2. Install Dependencies

```
- [ ] Install packages: pip install -r requirements.txt
- [ ] Verify installation: pip list
```

Expected packages:
- streamlit
- sentence-transformers
- torch
- pinecone-client
- openai

- deep-translator
- pandas, numpy

3. Configuration

```
- [ ] Copy .env.example to .env: cp .env.example .env
- [ ] Edit .env file with your actual API keys
- [ ] Verify .env exists: ls -la .env
```

Your `.env` should look like:

```
PINECONE_API_KEY=pc-xxxxxxxxxxx
PINECONE_ENVIRONMENT=us-east-1
PINECONE_INDEX_NAME=cmu-africa-kb
OPENAI_API_KEY=sk-xxxxxxxxxxx
```

4. Initialize Knowledge Base

```
- [ ] Run initialization: python init_knowledge_base.py
- [ ] Wait for completion (2-5 minutes)
- [ ] Verify success message appears
```

Expected output:

```
✅ Loaded 29 entries
✅ Embedding model loaded (dimension: 384)
✅ Vector store initialized
✅ Index ready
✅ Successfully uploaded 29 vectors to Pinecone
✅ Knowledge base initialization completed successfully!
```

5. Launch Application

```
- [ ] Start Streamlit: streamlit run app.py
- [ ] Browser opens automatically (http://localhost:8501)
- [ ] See "System Online" in sidebar
```

6. Test Basic Functionality

```
- [ ] Ask a test question (e.g., "What are the bus schedules?")
- [ ] Verify response appears
- [ ] Check sources are shown
- [ ] Test thumbs up/down feedback
- [ ] Try different language (French or Kinyarwanda)
```

7. Test Admin Panel

```
- [ ] Click "Admin Panel" in sidebar
- [ ] View entries in "View Entries" tab
- [ ] Test adding a new entry
- [ ] Test editing an entry
- [ ] Run re-indexing
- [ ] Verify changes appear in main chat
```

Troubleshooting Checklist

Issue: Dependencies won't install

- [ ] Check Python version (must be 3.9+)
- [ ] Update pip: pip install –upgrade pip
- [ ] Try installing one by one
- [ ] Check for error messages in output

Issue: "Pinecone API key not found"

- [ ] Verify .env file exists in project root
- [ ] Check file is named exactly ".env" (not .env.txt)
- [ ] Verify PINECONE_API_KEY line has no spaces around =
- [ ] Try printing: python -c "from dotenv import load_dotenv; load_dotenv(); import os; print(os.getenv('PINECONE_API_KEY'))"

Issue: "Index does not exist"

- [ ] Run init_knowledge_base.py first
- [ ] Check Pinecone dashboard for index
- [ ] Verify index name matches .env
- [ ] Check Pinecone environment matches .env

Issue: "OpenAI API error"

- [ ] Verify API key is correct
- [ ] Check billing is set up on OpenAI account
- [ ] Verify you have API credits
- [ ] Check API usage limits

Issue: Model download fails

- [ ] Check internet connection
- [ ] Wait longer (first download can take 5+ minutes)
- [ ] Check disk space (models ~500MB)
- [ ] Try different model in .env

Issue: Streamlit won't start

- [ ] Verify virtual environment is activated
- [ ] Check port 8501 is not in use
- [ ] Try different port: streamlit run app.py –server.port 8502
- [ ] Check for error messages

Issue: App shows “System not initialized”

- [ ] Check both API keys are set in .env
- [ ] Verify init_knowledge_base.py ran successfully
- [ ] Check Streamlit logs for errors
- [ ] Restart the application

Performance Verification

Expected Performance

- [ ] Initial load: 10-30 seconds (model loading)
- [ ] Query response: 2-5 seconds
- [ ] Admin re-indexing: 1-3 minutes
- [ ] Translation: < 1 second

If Slow

- [ ] Check internet connection
- [ ] Verify sufficient RAM available
- [ ] Reduce top_k in queries
- [ ] Use smaller embedding model
- [ ] Check API rate limits

Production Deployment Checklist

Before Deploying

- [ ] Test thoroughly in local environment
- [ ] Update knowledge base with real data
- [ ] Configure production API keys
- [ ] Set up monitoring/logging
- [ ] Configure backup strategy
- [ ] Add authentication (especially for admin)
- [ ] Set up SSL/HTTPS
- [ ] Configure domain DNS

Deployment Options

- [ ] Option 1: Streamlit Cloud (easiest)
- [ ] Option 2: Docker container
- [ ] Option 3: VPS/Cloud server
- [ ] Option 4: Kubernetes cluster

Post-Deployment

- [ ] Verify application is accessible
- [ ] Test all features in production
- [ ] Monitor error logs
- [ ] Set up uptime monitoring
- [ ] Configure backup schedule
- [ ] Document admin procedures

Maintenance Checklist

Weekly

- [ ] Check application logs for errors
- [ ] Monitor API usage and costs
- [ ] Review user feedback
- [ ] Check Pinecone index health

Monthly

- [ ] Update knowledge base content
- [ ] Re-index if significant changes
- [ ] Review and update dependencies
- [ ] Check for security updates
- [ ] Analyze usage patterns

Quarterly

- [ ] Major knowledge base review
- [ ] Update embedding model if needed
- [ ] Review and optimize performance
- [ ] Update documentation
- [ ] Plan feature enhancements

Support Resources

Documentation

- [ ] README.md - Complete setup guide
- [ ] .env.example - Configuration template
- [ ] This checklist - Step-by-step verification

External Resources

- [ ] Pinecone Documentation: https://docs.pinecone.io/
- [ ] OpenAI Documentation: https://platform.openai.com/docs
- [ ] Streamlit Documentation: https://docs.streamlit.io/
- [ ] Sentence Transformers: https://www.sbert.net/

Getting Help

- [ ] Check error messages carefully
- [ ] Review logs: ~/.streamlit/logs/
- [ ] Search GitHub issues
- [ ] Check Stack Overflow
- [ ] Contact support team

Success Criteria

Your setup is complete when:
- ✅ Application starts without errors
- ✅ System status shows “Online”

- ✅ Questions receive relevant answers
- ✅ Sources are displayed correctly
- ✅ Feedback buttons work
- ✅ Admin panel is accessible
- ✅ Can add/edit/delete entries
- ✅ Re-indexing works
- ✅ Multi-language works

Next Steps After Setup

1. **Customize Knowledge Base**
   - Replace sample data with real CMU-Africa information
   - Add more categories if needed
   - Update regularly

2. **User Training**
   - Train staff on admin panel
   - Create user guide
   - Demonstrate features

3. **Monitoring**
   - Set up error tracking
   - Monitor user engagement
   - Collect feedback

4. **Optimization**
   - Fine-tune response quality
   - Adjust embedding model if needed
   - Optimize performance

5. **Enhancement**
   - Add more features based on feedback
   - Integrate with other systems
   - Expand language support

Questions or Issues?
Refer to README.md for detailed troubleshooting or contact the development team.