Balasubramaniam Ramasamy

version  1.0

“DOCS CHATBOT” Deployment instructions

Contents

[1. Revision History 3](#_Toc183271479)

[2. Overview 3](#_Toc183271480)

[a. Frontend: 3](#_Toc183271481)

[b. Backend 1 3](#_Toc183271482)

[c. Backend 2 3](#_Toc183271483)

[d. Database 3](#_Toc183271484)

[3. Prerequisites 3](#_Toc183271485)

[System Requirements: 4](#_Toc183271486)

[Tools: 4](#_Toc183271487)

[Servers: 4](#_Toc183271488)

[4. Frontend Deployment 4](#_Toc183271489)

[Install Dependencies: 4](#_Toc183271490)

[Build the Angular App: 4](#_Toc183271491)

[Serve the Angular App: 5](#_Toc183271492)

[Environment Configuration: 5](#_Toc183271493)

[5. FastAPI Backend Deployment 5](#_Toc183271494)

[Clone the Repository: 5](#_Toc183271495)

[Set Up Virtual Environment: 6](#_Toc183271496)

[Environment Variables: 6](#_Toc183271497)

[Database Migration: 6](#_Toc183271498)

[Run FastAPI: 6](#_Toc183271499)

[Development: 6](#_Toc183271500)

[Production (with Gunicorn): 7](#_Toc183271501)

[Serving with Nginx (Optional): Configure Nginx as a reverse proxy to FastAPI: 7](#_Toc183271502)

[6. Node.js Backend Deployment 7](#_Toc183271503)

[Clone the Repository: 7](#_Toc183271504)

[Install Dependencies: 7](#_Toc183271505)

[Environment Variables: 7](#_Toc183271506)

[Run the App: 8](#_Toc183271507)

[Development: 8](#_Toc183271508)

[Production: 8](#_Toc183271509)

[Serve with Nginx (Optional): Configure Nginx as a reverse proxy: 8](#_Toc183271510)

[7. Database Setup 9](#_Toc183271511)

[SQLite: 9](#_Toc183271512)

[PostgreSQL/MySQL (for production): 9](#_Toc183271513)

[8. Testing 9](#_Toc183271514)

[Frontend: 9](#_Toc183271515)

[Unit Tests: 9](#_Toc183271516)

[End-to-End Tests: 9](#_Toc183271517)

[Backend: 9](#_Toc183271518)

[FastAPI: 9](#_Toc183271519)

[Node.js: 9](#_Toc183271520)

[9. Snapshots for Verification UI and Functional Verification 10](#_Toc183271521)

[a. Register Screen 10](#_Toc183271522)

[b. Login Screen 10](#_Toc183271523)

[c. Documents Screen 11](#_Toc183271524)

[d. Query Screen 11](#_Toc183271525)

[e. Document Upload Screen 11](#_Toc183271526)

[f. Logout Screen 12](#_Toc183271527)

[10. Monitoring and Maintenance 12](#_Toc183271528)

[11. Common Issues and Debugging 12](#_Toc183271529)

[CORS Errors: 12](#_Toc183271530)

[Database Connection Issues: 12](#_Toc183271531)

[Static File Issues: 12](#_Toc183271532)

[12. README’s 12](#_Toc183271533)

[13. Support 13](#_Toc183271534)

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Author | Remarks | Version |
| 23/11/2024 | Balasubramaniam Ramasamy | Initial Version | 1.0 |
|  |  |  |  |

# Overview

Hello, welcome to Docs Chatbot.

The system architecture:A screenshot of a chatbot

Description automatically generated

1. Frontend: Angular application
2. Backend 1: FastAPI (Python-based API for handling document uploads, queries, and embeddings)
3. Backend 2: Node.js (Authentication and user management)

d. Database: SQLite (for development) or any production-grade DB (e.g., PostgreSQL, MySQL)

# 3. Prerequisites

List software, tools, and environment setups required:

## System Requirements:

OS: Linux/Windows/MacOS (recommend Linux for production)

RAM: At least 4 GB for local, 8+ GB for production

## Tools:

Angular CLI: npm install -g @angular/cli

Node.js: Version >= 20.0

Python: Version >= 3.10

Npm: Version >= 10.0

Database:

SQLite (for development) or PostgreSQL/MySQL for production

Package managers:

npm (for Node.js and Angular)

pip or pipenv (for Python)

## Servers:

Nginx or Apache (optional, for serving Angular and acting as reverse proxy)

Gunicorn or uvicorn for serving FastAPI

# 4. Frontend Deployment

## Install Dependencies:

Clone the repository: git clone <frontend-repo-url>

Navigate to the project folder: cd <project-folder>

Install dependencies: npm install

## Build the Angular App:

Development: ng serve

Production: ng build --prod --output-path=dist

## Serve the Angular App:

For development: Use ng serve (default runs on http://localhost:4200).

For production:

Copy the contents of dist to your server.

Configure Nginx or Apache to serve the static files from dist.

## Environment Configuration:

Use environment.ts and environment.prod.ts for API URLs and environment-specific settings.

Sample Nginx Configuration:

nginx

Copy code

server {

listen 80;

server\_name your-domain.com;

root /var/www/your-angular-app/dist;

index index.html;

location / {

try\_files $uri /index.html;

}

}

# 5. FastAPI Backend Deployment

## Clone the Repository:

git clone <fastapi-repo-url>

cd <project-folder>

## Set Up Virtual Environment:

bash

Copy code

python -m venv venv

source venv/bin/activate

pip install -r requirements.txt

***Note:*** *You can also use conda to setup environment and run FastAPI*

## Environment Variables:

Create .env file with:

env

Copy code

DATABASE\_URL=sqlite:///./database.sqlite

SECRET\_KEY=your-secret-key

## Database Migration:

Use alembic or SQLAlchemy for migrations:

bash

Copy code

alembic upgrade head

## Run FastAPI:

### Development:

bash

Copy code

uvicorn main:app --reload --host 0.0.0.0 --port 8000

### Production (with Gunicorn):

bash

Copy code

gunicorn -k uvicorn.workers.UvicornWorker main:app --bind 0.0.0.0:8000

### Serving with Nginx (Optional): Configure Nginx as a reverse proxy to FastAPI:

nginx

Copy code

server {

listen 80;

server\_name your-domain.com;

location / {

proxy\_pass http://127.0.0.1:8000;

proxy\_set\_header Host $host;

proxy\_set\_header X-Real-IP $remote\_addr;

}

}

# 6. Node.js Backend Deployment

## Clone the Repository:

git clone <nodejs-repo-url>

cd <project-folder>

## Install Dependencies:

npm install

## Environment Variables:

Create .env file with:

env

Copy code

PORT=3000

DATABASE\_URL=sqlite:///./auth-database.sqlite

JWT\_SECRET=your-secret-key

## Run the App:

### Development:

bash

Copy code

npm run dev

### Production:

bash

Copy code

npm start

### Serve with Nginx (Optional): Configure Nginx as a reverse proxy:

nginx

Copy code

server {

listen 80;

server\_name your-domain.com;

location /auth {

proxy\_pass http://127.0.0.1:3000;

proxy\_set\_header Host $host;

proxy\_set\_header X-Real-IP $remote\_addr;

}

}

# 7. Database Setup

## SQLite:

Auto-created for both FastAPI and Node.js if used as sqlite:///./database/database.sqlite.

No further setup required.

## PostgreSQL/MySQL (for production):

Create databases for FastAPI and Node.js.

Update the DATABASE\_URL in .env files:

env

Copy code

DATABASE\_URL=postgresql://user:password@localhost/dbname

# 8. Testing

## Frontend:

Unit Tests: ng test

End-to-End Tests: ng e2e

## Backend:

FastAPI: Use pytest to run tests:

bash

Copy code

pytest

Node.js: Use jest or mocha to run tests:

bash

Copy code

npm test

# 9. Snapshots for Verification UI and Functional Verification

## a. Register Screen

A screenshot of a computer

Description automatically generated

b. Login Screen A screenshot of a computer

Description automatically generated

## c. Documents Screen

A screenshot of a computer

Description automatically generated

## d. Query Screen

A screenshot of a computer

Description automatically generated

## e. Document Upload Screen

A screenshot of a computer

Description automatically generated

f. Logout ScreenA screenshot of a computer

Description automatically generated

# 10. Monitoring and Maintenance

Use tools like PM2 (for Node.js) and supervisord (for Python FastAPI) to keep backend services running.

Set up logging:

FastAPI: Use Python's logging module.

Node.js: Use a logger like winston.

Monitor server health with tools like UptimeRobot or Prometheus.

# 11. Common Issues and Debugging

## CORS Errors:

Ensure CORS is enabled in both FastAPI and Node.js.

## Database Connection Issues:

Verify DATABASE\_URL and database permissions.

## Static File Issues:

Ensure Nginx or Apache is correctly serving Angular files.

By following these steps, you ensure a smooth deployment of the Angular frontend and backend services, making the application production-ready.

# 12. README’s

There are four readme files. Please, check them individually and update them if needed.

The folders are:

1. .\docs-chatbot\README.md
2. .\docs-chatbot\frontend\README.md
3. .\docs-chatbot\backend-nodejs\README.md
4. .\docs-chatbot\backend-fastapi\README.md

# 13. Support

In case of any help needed, please, feel free to create an issue in GitHub against the respective repository.