

AI Travel Planner Submission

GitHub Repository: <https://github.com/your-account/AI-Travel-Planner>

README Summary

AI Travel Planner AI

PythonFastAPI AI Docker PDF

##

- ** / ** Web Speech API API
- **AI** LLM Mock DashScope OpenAI//
- ****
- ****JWT
- **** Key
- **Docker & CI/CD** Dockerfiledocker-compose GitHub Actions
- **PDF** GitHub README submission.pdf

##

- ****FastAPISQLAlchemy (Async)Pydantic SettingsJWT (python-jose)Passlib
- **** SQLite `DATABASE_URL` Supabase/PostgreSQL
- ****Jinja2 + JS
- **AI** DashScope / OpenAI / Mock
- **** Web Speech API IAT REST API
- ****DockerDocker ComposeGitHub Actions

##

...

- app/ # FastAPI
- api/ # REST API
- core/ #
- db/ # &
- models/ # SQLAlchemy
- repositories/ #
- schemas/ # Pydantic
- services/ # LLM / Speech / Planning
- static/ # CSSJS
- templates/ # Jinja2
- docs/ # PDF
- .github/workflows/ # GitHub Actions CI
- Dockerfile
- docker-compose.yml
- requirements.txt
- README.md

.env.example

```

##

1. \*\*\*\*

```bash

python -m venv .venv

.\venv\Scripts\activate # Windows PowerShell

pip install -r requirements.txt

cp .env.example .env

```

2. \*\*\*\*

```bash

uvicorn app.main:app --reload

```

3. \*\*\*\*

[http://localhost:8000](http://localhost:8000)

4. \*\*API\*\*

- FastAPI Docs: `http://localhost:8000/docs`

- Redoc: `http://localhost:8000/redoc`

##

| | | |

| --- | --- | --- |

| `ENVIRONMENT` | development/production | development |

| `SECRET\_KEY` | JWT | change-me |

| `ACCESS\_TOKEN\_EXPIRE\_MINUTES` | Token | 1440 |

| `DATABASE\_URL` | sqlite / postgres / supabase | sqlite+aiosqlite:///./travel\_planner.db

|

| `LLM\_PROVIDER` | `mock` / `dashscope` / `openai` | mock |

| `LLM\_MODEL` | LLM | qwen-turbo |

| `LLM\_API\_KEY` | LLM Key 3 | |

| `LLM\_ENDPOINT` | LLM Endpoint | |

| `SPEECH\_PROVIDER` | `web` / `iflytek` | web |

| `IFLYTEK\_APP\_ID` | App ID | |

| `IFLYTEK\_API\_KEY` | API Key | |

| `AMAP\_API\_KEY` | JS API Key | |

> \*\*\*\* API Key Git `/settings` Key

## &

\_ \*\*\*\*

- Web Speech API Chrome/Edge HTTPS localhost
- `.env` `IFLYTEK\_APP\_ID` `IFLYTEK\_API\_KEY`

\_ \*\*\*\*

- `.env` `AMAP\_API\_KEY`

## LLM

1. \*\*Mock\*\*

2. \*\*DashScope\*\*

- `.env` `LLM\_PROVIDER=dashscope` `LLM\_API\_KEY`
- `LLM\_ENDPOINT`

3. \*\*OpenAI\*\*

- `LLM\_PROVIDER=openai` `LLM\_API\_KEY` `LLM\_ENDPOINT` Azure

##

-  
-  
-

## Docker

###

```bash

docker build -t ai-travel-planner:latest .

```

### docker-compose

```bash

docker compose up --build

```

`http://localhost:8000`

## GitHub Actions

`.github/workflows/docker-publish.yml` Secrets

- `ACR\_USERNAME` / `ACR\_PASSWORD`
- `ACR\_NAMESPACE` / `ACR\_REPOSITORY`

push `main` `registry.cn-hangzhou.aliyuncs.com/<namespace>/<repository>:<git-sha>`

##

-

```
```bash
python -m compileall app
```
```

- `pytest` LLM mock API

## PDF

`scripts/generate\_pdf.py` `docs/submission.pdf` GitHub README

```
```bash
python scripts/generate_pdf.py --repo https://github.com/<your-account>/AI-Travel-Planner
```
```

PDF

##

- 1.
2. /
- 3.
- 4.

---

README issue