

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

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 APIChrome/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