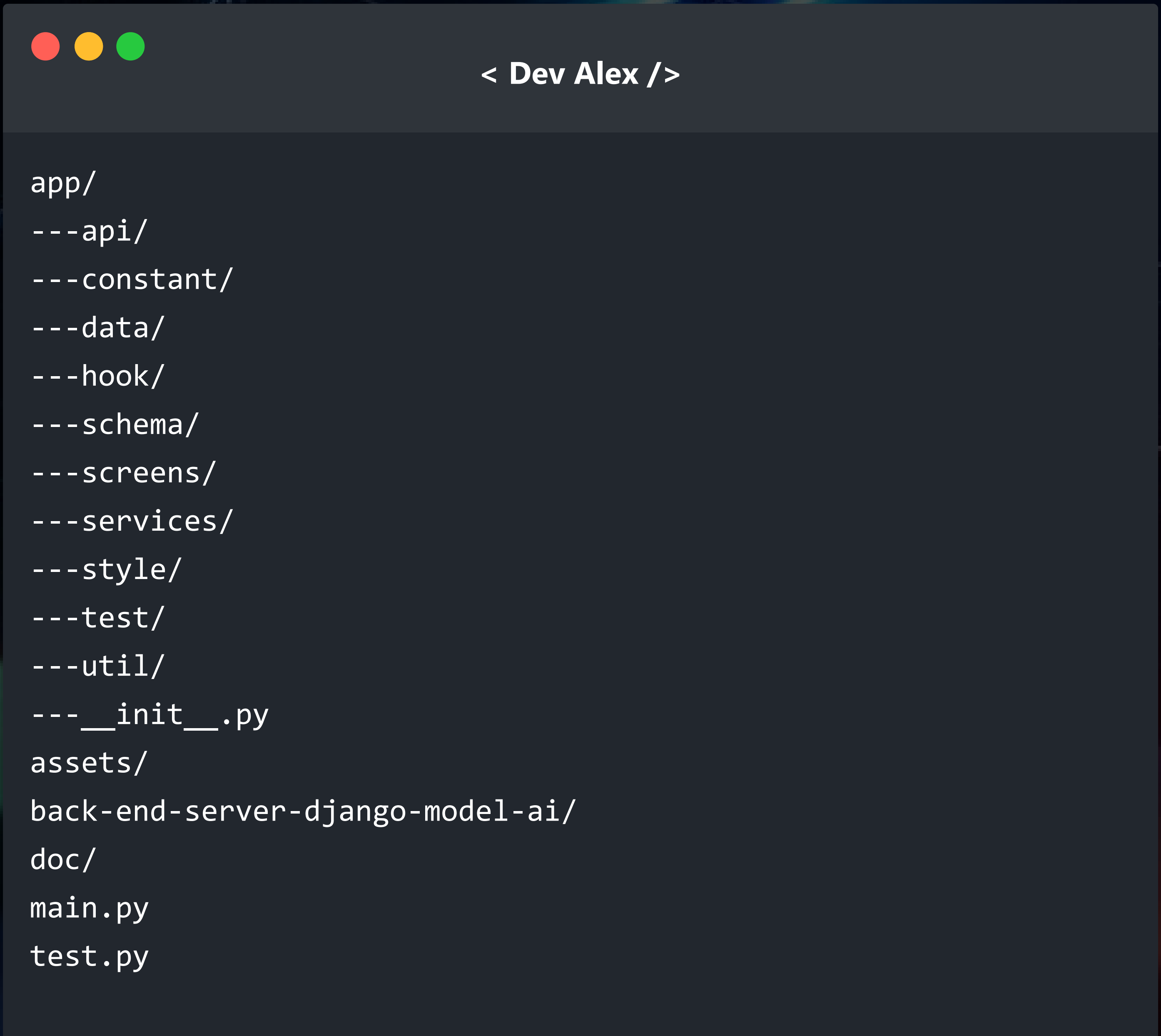
BUILDING EDU PROGRAMING COURSES MOBILE APP USING FLET FRAMEWORK

TEAM DEV

- Nguyễn Phương Anh Tú (Leader)
- Đinh Thành Đức (Co-op)
- Lê Quốc Thắng (Co-op)

STRUTURE OF PROJECT



< Dev Alex />

```
app/
---api/
---constant/
---data/
---hook/
---schema/
---screens/
---services/
---style/
---test/
---util/
---__init__.py
assets/
back-end-server-django-model-ai/
doc/
main.py
test.py
```

```
README.md
README.pdf
```

1. **app/**: This is the main directory for the application code.
   - **api/**: Contains the API-related code, such as endpoints and views.
   - **constant/**: Stores constant values used across the application.
   - **data/**: Contains data-related files, possibly including data models or static data.
   - **hook/**: Includes custom hooks if using a framework that supports hooks.
   - **schema/**: Stores schema definitions, likely for database models or API request/response schemas.
   - **screens/**: Contains screen components, typically used in front-end frameworks like React Native or Flutter.
   - **services/**: Holds service-related code, such as API calls, business logic, or external service integrations.
   - **style/**: Includes styling-related files.
   - **test/**: Contains test cases for the application.
   - **util/**: Stores utility functions or helper modules.
   - **init.py**: A Python file to mark this directory as a package.
2. **assets/**: Stores static assets such as images, fonts, or other media files.
3. **back-end-server-django-model-ai/**: Contains the back-end server code, likely a Django application with some AI model integration.
4. **doc/**: Includes documentation files for the project.
5. **main.py**: The main entry point of the application.
6. **test.py**: A script for running tests.
7. **README.md**: A markdown file containing the project description, setup instructions, and other relevant information.
8. **README.pdf**: A PDF version of the README file, potentially containing the same information as README.md.

I. OVERALL THIS APPLICATIONS

- UI: using Flet Framework to build Mobile App (build Flutter App) - language Python
- Back End: using Django Framework to build Server Side API and apply model AI sentiment - language Python
- Database: using PostgreSQL to store data, register FREE on render: https://render.com/
- Hosting: using Render to deploy Django API and PostgreSQL Database: https://render.com/
- Model AI Sentiment intergrated in Back End: using underthesea library to build model AI Sentiment - language Python: https://github.com/undertheseanlp/underthesea