High-Level Overview

Application Summary

ACT (Automated Code Translation) is a developer-facing tool designed to accelerate the modernization of legacy systems by translating **Delphi Pascal code into modern C#**. The tool is powered by **Retrieval-Augmented Generation (RAG)** using OpenAI's LLMs, which improves translation quality by grounding generations in real Delphi-C# examples.

The application provides:

- An intuitive web interface for uploading or pasting Delphi code.
- Fast, accurate C# translations using AI.
- A secure login system via **OIDC authentication** (powered by Authentik).
- A translation history dashboard with features like **filtering, editing, deleting, and copying** translations.
- Persistent storage of translations and user sessions using **CouchDB** and **PostgreSQL**.

Target Users:
Developers, maintainers, or contractors tasked with migrating legacy Delphi systems to C# within defense or enterprise software environments.

Business Value:

- Saves engineering time by automating routine code translation tasks.
- Reduces translation errors by grounding responses in domain-relevant examples.
- Provides secure user management and traceable translation logs for auditing and debugging.

Current Status

Area	Status	Notes
Core Translation	✅ Complete	Uses FAISS + OpenAI + curated examples for accuracy.

Authentication (OIDC)	✅ Complete	Integrated with Authentik; securely manages user sessions.
Frontend UI	✅ Complete	Fully responsive HTML/CSS/JS UI. Includes tooltips and feedback UX.
Translation History	✅ Complete	Users can save, edit, delete, filter, and revisit past translations.
File Upload Support	✅ Complete	Supports .dpr files via drag-and-drop and browsing.
Feedback System	✅ Complete	Emails routed to team via 2msgactdev@gmail.com.
Local Deployment	✅ Complete	Documented Docker setup with setup.sh automation script.
CI/CD or Hosting	❌ Not Implemented	Meant for local or internal server deployment only.
User Management UI	❌ Not Implemented	Admin console exists via Authentik, but no app-level user roles.

Known Limitations & Notes:

- **Accuracy:** While the AI model performs well with common patterns, edge cases may require human post-processing.
- **Lack of testing –** No testing implemented, unable to test translation accuracy because no access to testing dataset unknown to RAG. Testing roadmap is provided in this folder.
- **No real-time collaboration** – each user session is isolated.
- **Not cloud-hosted** — requires local setup or internal infrastructure support (documented in deployment section).

🛠️ Admin Guide

🔐 Managing Users (via Authentik Admin Panel)

All user authentication and authorization are handled by **Authentik**, accessible at:

http://<your-host>:9000/if/admin/

Once authenticated as an admin:

☑ *Tasks You Can Perform:*

- **View all users**
- **Revoke access** to individual accounts
- **Assign roles or group permissions**
- **Manually reset passwords**
- **Monitor login attempts and activity logs**

🔧 *Useful Authentik Sections:*

Section	Description
Users	View, add, or deactivate user accounts
Groups	Organize users and assign role-based access
Flows	Configure authentication flows like signup/login
Tokens	View and revoke issued authentication tokens
Events	Track login activity and admin-level actions

⚙ System Settings

There is **no custom admin UI built into the ACT application**. All operational configuration (such as allowed redirect URIs, client secrets, or scopes) must be managed through:

- **Authentik Admin Panel** (for login + identity)
- **.env file** (for API keys, database URLs, Authentik endpoints)
- **Docker Compose** (for container orchestration and updates)

To change default credentials, update:

- COUCHDB_USER and COUCHDB_PASSWORD
- POSTGRES_USER and POSTGRES_PASSWORD
- Authentik's user roles or secrets via the admin panel or authentik_dump.sql

💼 Admin Tasks Overview

Task	Where to Manage
Revoke user access	Authentik Admin > Users
Change environment config	.env file (then restart containers)
Restart a service (e.g., Auth)	docker restart <container-name>
Monitor API or translator	Terminal logs / Docker container logs
Rotate OpenAI API key	Update .env → OPENAI_API_KEY

❓ Frequently Asked Questions & Troubleshooting

🔄 "I can’t log in to the application"

- ✅ Check if Authentik is running at localhost:9000
- ✅ Confirm that your .env file points to the correct OIDC endpoints
- ✅ Verify hosts file contains the alias 127.0.0.1 authentik (or custom hostname)

⚠️ "Translation fails or takes too long"

- Check your OpenAI API key is valid and not rate-limited.
- Make sure faiss_index.index and indexed_metadata.jsonl exist in the Docker container.
- Restart the translator service:

```
docker compose restart translator
```

🛑 "Containers won’t start"

- Run: docker compose down followed by docker compose up --build

- Check for syntax errors in .env or docker-compose.yml
- Make sure Docker Desktop is running in the background

🗂 "No translations are showing in File Drive"

- Confirm that:
  - The translations CouchDB is seeded or has data
  - The backend server (Node.js) is running and connected to the database
- Open console logs with:

```
docker logs app
```

🧪 "How do I test this app on a new machine?"

1. Clone the repo
2. Setup Docker and .env
3. Run docker compose up --build
4. Open http://localhost:3000