Installation Manual

AT-AT Installation Manual

This document guides you through the complete local installation and execution of the **API Threat Assessment Tool (AT-AT)**.

Prerequisites

- **Node.js** ≥ 16
- **Python** ≥ 3.10
- **npm** ≥ 8
- **Git** 2.44+
- Recommended: PowerShell (Windows) or bash (Linux/macOS)

1. Clone the Repository

```
git clone https://github.com/YourOrg/API-Threat-Assessment-Tool.git
cd API-Threat-Assessment-Tool
```

2. Create `.env` Files

Each of the three components — `frontend`, `api`, and `backend` — require `.env` files. Below are their templates:

frontend/.env

```
SUPABASE_URL=https://your-project.supabase.co
SUPABASE_KEY=your-supabase-service-key
JWT_SECRET=your-secret
FRONTEND_URL=http://localhost:3000
PORT=3000
HOST=0.0.0.0
DANGEROUSLY_DISABLE_HOST_CHECK=true
DOCKER=TRUE
```

api/.env

```
SUPABASE_URL=https://our-link.supabase.co
SUPABASE_KEY=your-supabase-service-key
JWT_SECRET=your-secret
FRONTEND_URL=http://localhost:3000
PORT=3001
GMAIL_USER=at.at.noreply@gmail.com
GMAIL_CLIENT_ID=client_id
GMAIL_CLIENT_SECRET=client_secret
GMAIL_REFRESH_TOKEN=refresh_token
DOCKER=TRUE
```

backend/.env

```
PORT=9011
FRONTEND_URL=http://localhost:3000
SUPABASE_URL=https://our-link.supabase.co
SUPABASE_KEY=your-supabase-service-key
JWT_SECRET=your-secret
FRONTEND_URL=http://localhost:3000
DOCKER=TRUE
```

3. Repository Layout

```
/api/              Node/Express public API (OpenAPI served here)
/backend/          Python service(s) for scanning/analysis
/docs/             SRS, Service Contracts, OpenAPI (openapi.yaml), manuals
.github/workflows/ CI pipelines
```
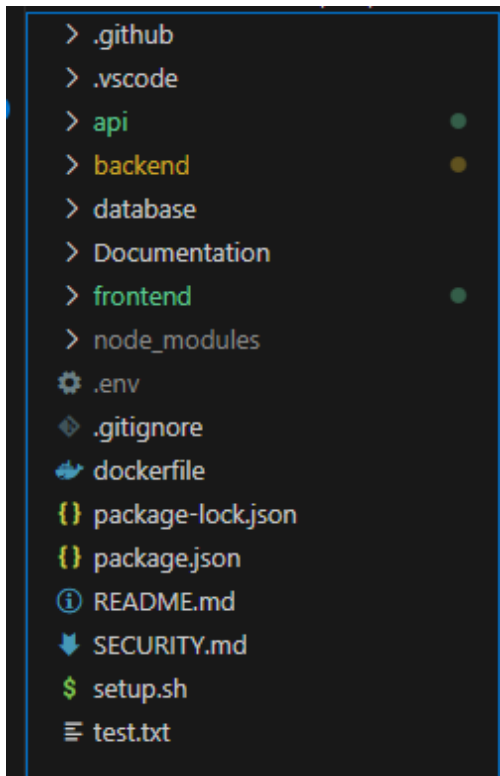

Setup and Running

To run everything at once use

```
cd frontend
npm start
```

This will start the necessary requirements, however you can laucnh each individually.

```
> frontend@0.1.0 start
> concurrently "npm run start:api" "npm run start:client"

[0]
[0] > frontend@0.1.0 start:api
[0] > cross-env-shell "cd ../api && npm start"
[0]
[1]
[1] > frontend@0.1.0 start:client
[1] > react-scripts start
[1]
[0]
[0] > api-threat-assessment-backend@1.0.0 start
[0] > node index.js
[0]
[0] 🚀 Server running at http://localhost:3001
[0] ⚡ Engine will auto-start when needed
[0] 📧 Email configured: none
[0] 🔒 Google OAuth: Enabled
[1] Attempting to bind to HOST environment variable: 0.0.0.0
```

```
[1] Starting the development server...
[1]
[1] Compiled successfully!
[1]
[1] You can now view frontend in the browser.
[1]
[1]    http://localhost:3000
[1]
[1] Note that the development build is not optimized.
[1] To create a production build, use npm run build.
[1]
[1] webpack compiled successfully
```

4. Setup and Run Backend (Python)

```
cd backend
python -m venv venv
.\venv\Scripts\Activate       # Windows
# or: source venv/bin/activate  # macOS/Linux


pip install -r requirements.txt
python main.py
```

5. Setup and Run API (Node.js)

```
cd api
npm install
node index.js
```

6. Setup and Run Frontend (React)

```
cd frontend
npm install
npm run start:client
```

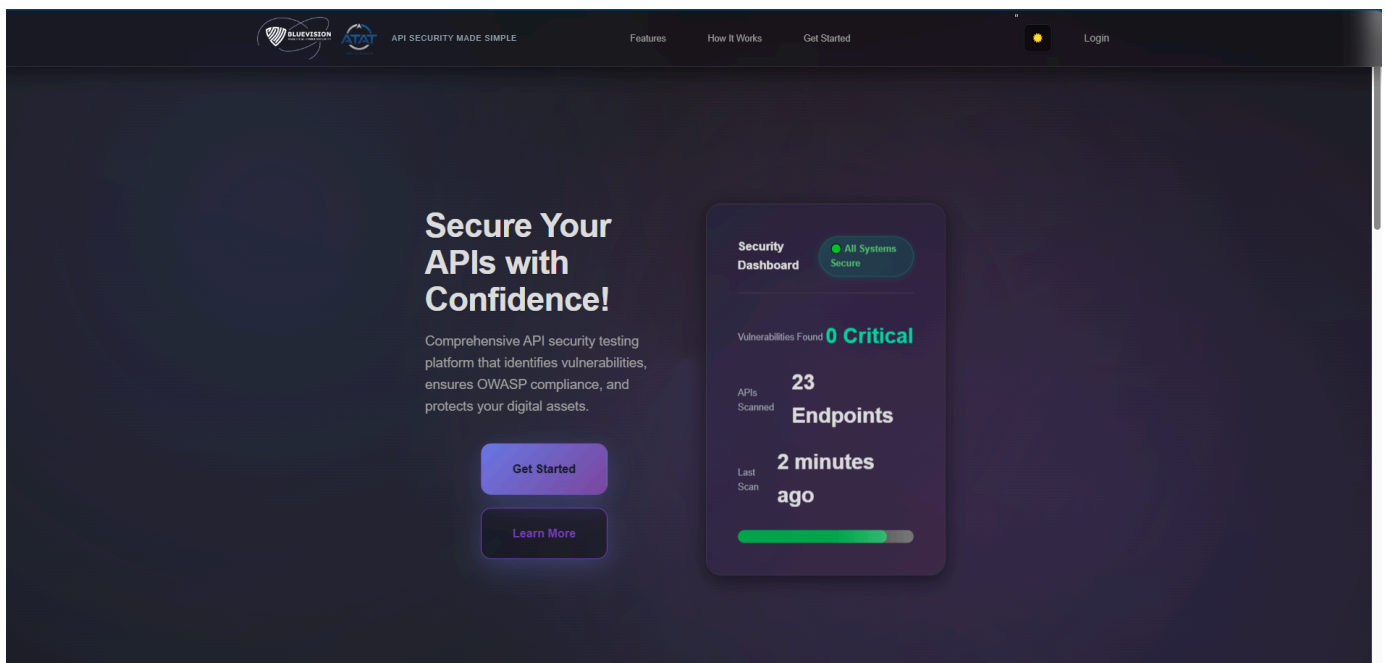If the `start:client` script fails with a host error, ensure your `.env` includes:

```
HOST=0.0.0.0
DANGEROUSLY_DISABLE_HOST_CHECK=true
```

7. Test in Browser

Open:

```
http://localhost:3000
```

You should see the AT-AT UI.

- Log in
- Upload a spec to test backend connectivity
- Verify `localhost:3001` and `localhost:9011` are active

---

8. Docker

```
docker build -f dockerfile -t atat .
docker run --rm -p 3000:3000 -p 3001:3001 -p 9011:9011 atat
```

Common Troubleshooting

Issue	Fix
`allowedHosts` error	Add `HOST=0.0.0.0` and `DANGEROUSLY_DISABLE_HOST_CHECK=true`
Supabase not loading	Ensure all `.env` files have correct `SUPABASE_URL` and `SUPABASE_KEY`
Python crash on `dotenv`	Run `pip install python-dotenv`

Final Notes

- GitHub Actions CI/CD is available and runs tests on pull requests.

- This setup is for local development.