

Build Documentation

Local setup:

Prerequisites

Make sure the following are installed on your machine:

- **Node 20**
- **Docker**
- **Docker Compose**

```
npm ci
cd ./apps/analyzer/metadata_analyzer ; poetry install
```

- `npm ci` : dependency install
- copy `.env.example` file in backend and rename to `.env` (adjust database properties according to database setup if necessary)
- copy `.env.example` file in analyzer and rename to `.env` (adjust port properties according to backend setup if necessary)

Running the code locally:

- `npm run be` : run backend individually
- `npm run fe` : run frontend individually
- `npm run py` : run python app
- `npm run all` : run backend, frontend and python module

Generating database migrations:

- the entity files need to be annotated with `@Entity(<table-name>)`
- append the entity file to the `entities` array in `db-config.service.ts`
- run the following command to generate a migration file:
 - `nx run metadata-analyzer-backend:migrations:generate --name <migration-name>`
- append the generated file to the `migrations` array in `db-config.service.ts`

Running tests

- backend: `nx run metadata-analyzer-backend:test`
- frontend: `nx run metadata-analyzer-frontend:test`

- python: `nx run metadata-analyzer:test`

Mailing

Fill the `.env` file with the mailing information

Hint: For gmail you have to generate an app password, which you have to use as password in the `.env` file

For changes in templates being used, you have to restart the backend

Installing new dependencies

in python app

When working with python dependencies, first cd into the `analyzer` folder.

Install a new dependency

```
poetry add <dependency-name>
```

Remove a dependency

```
poetry remove <dependency-name>
```

Using docker

1. Clone the repository:

```
git clone https://github.com/amosproj/amos2024ws02-backup-metadata-analyzer.git
```

2. Change directory:

```
cd ./amos2024ws02-backup-metadata-analyzer/
```

3. Setup .env files:

```
cp .env.docker.example .env.docker
```

4. Copy database dump into project:

Copy the database dump `.dmp` file in the projects root folder and rename it to `db_dump.sql`

5. Clean Docker node_modules:

```
docker volume rm amos2024ws02-backup-metadata-analyzer_mono-node-modules
```

6. Build Docker container:

```
docker compose --env-file .env.docker build --no-cache
```

7. Start Docker container:

```
docker compose --env-file .env.docker up
```

8. Stop Docker Container:

```
docker compose --env-file .env.docker down
```