URL Shortner Application

Requirements

The requirements are gathered in this document : docs/01_requirements/README.md

Design

The design is described in this document : docs/02_design/README.md

Usage

This is a quick guide to how to run the programs locally from source (a more detailed guide will be available soon).

Requirements

To run this project, you need to have the following tools installed on your machine :

- Golang (1.20)
- Make
- Docker

How to test the platform locally ?

1. Go to the root of the project.

There should be the following files:

- `env` => configuration, used in development mode
- `redirection_400.html` => html page to be shown by the redirection server if customer or short url are not found or if they are disabled
- `redirection_500.html` => html page to be show by the redirection server if there is an unexpected internal error

2. Open three separate terminal windows.

3. In terminal 1 : Start postgres database using Docker :

```
docker run -e POSTGRES_PASSWORD=password -e POSTGRES_USER=postgres -
```

```
p 5433:5432  postgres
```

4. Connect to the database with the tool of your choice (dbeaver, pgadmin, etc.).

```
- Host : *localhost*
- Port : *5433*
- Database : *postgres*
- Username : *postgres*
- Password : *postgres*
```

5. Run the initialization script that you can find here:

```
<project>/db/migration/000001_init_schema.up.sql
```

6. In terminal 2 : Start the Short API Server on port 8081 :

```
        go run ./cmd/api-server/ --port 8081 --host localhost
```

Then Short API server is now accessible on : http://localhost:8081/docs

7. In terminal 3 : Start the Redirection server on port 8085 :

```
    go run ./cmd/redirection-server/ --port 8085 --host localhost
```

8. Using the http://localhost:8081/api/docs or using cUrl:

8.1 Create a new customer (Admin API key is 1234):

```
    curl -X POST \
    -H "Content-Type: application/json" \
    -H "X-API-KEY: 1234" \
    -d '{
    "username": "johnsmith",
    "email": "johnsmith@example.com"
    }' \
    http://localhost:8081/api/customer/
```

8.2 Create a new short url for the customer, by using the API key returned by the previous call.

```
curl -X POST \
-H "Content-Type: application/json" \
-H "X-API-KEY: {customer API generated in step 8.1}" \
-d '{
    "long_url": "https://www.example.com"
}' \
http://localhost:8081/api/short-url/
```

8.3 Try to access the short url generated by the previous call, you should be redirected to the long URL.

9. Check the database to see the data that has been created and modified

Environment Variables

This is a list of the environment variables that can be used to configure the application.

- DB_DRIVER=postgres
- DB_SOURCE= for example `postgresql://postgres:password@localhost:5433/postgres?sslmode=disable`
- ADMIN_API_KEY= for example `1234`
- REDIRECTION_SERVER_BASE_URL= for our example `http://localhost:8085`
- REDIRECTION_404_PAGE=, default `redirection_404.html`
- REDIRECTION_500_PAGE=, default: redirection_500.html

Code Structure

TODO