gen-ner-ui-umbrella

fetch submodules

1. clone project and fetch submodules
   1. ocr-pipeline
   2. generic-ner-ui
   3. generic-ner-ui_headless

```
git submodule init
git submodule update
```

platform support

- check system architecture: `$ arch`

```
BOW_ARCH=$(arch)
echo $BOW_ARCH
## x86_64
## arm64
```

build & run locally

1. start all containers

```
BOW_ARCH=$(arch)
echo $BOW_ARCH
docker-compose -f docker-compose.$BOW_ARCH.yml up --build rabbitmq
postgres database minio keycloak generic-ner-ui ocr-pipeline
```

2. http://localhost:8080/auth/
   1. create testuser/testuser
   2. copy public key
   3. update *CONFIG_KEYCLOAK_PUBLIC_KEY* in *docker-compose.yml*
3. restart containers
   1. in case of re-start – without *--build* parameter
4. http://localhost:8000/bow/main/
   1. Welcome to BOW 👋

CVL Server Configuration

services running on CVL

service	port	access	access	domain	description
generic-ner-ui	8000	all users	public	bow.vhh.cvl.tuwien.ac.at/bow/main/	BOW user interface for data upload and retrieval, push tasks into rabbitmq queue
keycloak	8080	ingo & admin	public	auth.bow.vhh.cvl.tuwien.ac.at	user management for generic-ner-ui
minio	9000	wireguard configs	WireGuard vpn	minio.vhh.cvl.tuwien.ac.at	object storage, retrieve data from ocr_pipeline, provide data to generic-ner-ui
rabbitmq	15672	wireguard configs	WireGuard vpn	rabbitmq.vhh.cvl.tuwien.ac.at	manage tasks from generic-ner-ui
ocr_pipeline	-	-	-	-	process tasks from rabbitmq queue; read/write data from minio
mysql	3306	-	-	-	stores generic-ner-ui task metadata
postgres	5432	-	-	-	stores keycloak users

update git project & run on CVL

```
### connect
ssh cvl

### BOW — DBS (5 containers)
tmux a -t cli-session
cd /home/project/generic-ner-ui-umbrella
git pull
docker-compose -f docker-compose.x86_64.yml up rabbitmq postgres database
minio keycloak

### BOW — UI (1 container)
tmux a -t ui-session
cd /home/project/generic-ner-ui-umbrella/generic-ner-ui/
git pull
docker-compose -f docker-compose.x86_64.yml up generic-ner-ui

### BOW — OCR (1 container)
tmux a -t ocr-session
cd /home/project/generic-ner-ui-umbrella/ocr-pipeline
git pull
docker-compose -f docker-compose.x86_64.yml up ocr-pipeline
```

git authenticate

```
git pull
# _name_
# _token_
# login enter token git: settings > developer > token classic >
(re)generate > paste
```

tmux sessions

```
## create new session
tmux new -s cli-session
tmux new -s ui-session
tmux new -s ocr-session

## start sample-server on session
cd /data/project/service_example
python3 -m http.server --bind localhost 8080

## attach
tmux a -t cli-session
tmux a -t ui-session
tmux a -t ocr-session
```

```
## list
tmux ls

## exit tmux
Ctrl+b d or Ctrl+b :detach
```

storage attached to containers

```
df -h /data
# /dev/mapper/bowdata-data  590G  121G  439G  22% /data

docker system df -v
# ...
# generic-ner-ui-umbrella_dbdata         1     54.4MB
# generic-ner-ui-umbrella_migrations     1     6.211kB
# generic-ner-ui-umbrella_miniodata      1     99.34GB
# generic-ner-ui-umbrella_nerdbdata      1     1.918GB
```

wireguard configuration(s)

directory:

```
ls /home/project/wireguard
# jweber_bow.conf
# izechner_bow.conf
```

adapting the yaml file

docker-compose.x86_64.yml

```
# run localhost
- CONFIG_KEYCLOAK_HOST=localhost:8080
- CONFIG_KEYCLOAK_ACCESS_HOST=keycloak:8080

# run live on cvl
- CONFIG_KEYCLOAK_HOST=auth.bow.vhh.cvl.tuwien.ac.at
- CONFIG_KEYCLOAK_ACCESS_HOST=keycloak:8080
- PROXY_ADDRESS_FORWARDING=true
- REDIRECT_SOCKET=proxy-https
- KEYCLOAK_FRONTEND_URL=https://auth.bow.vhh.cvl.tuwien.ac.at/auth

# on any deployment update after first keycloak run:
- CONFIG_KEYCLOAK_PUBLIC_KEY=...
```