Gieß den Kiez

- Building the Backend
- 12.05.2021

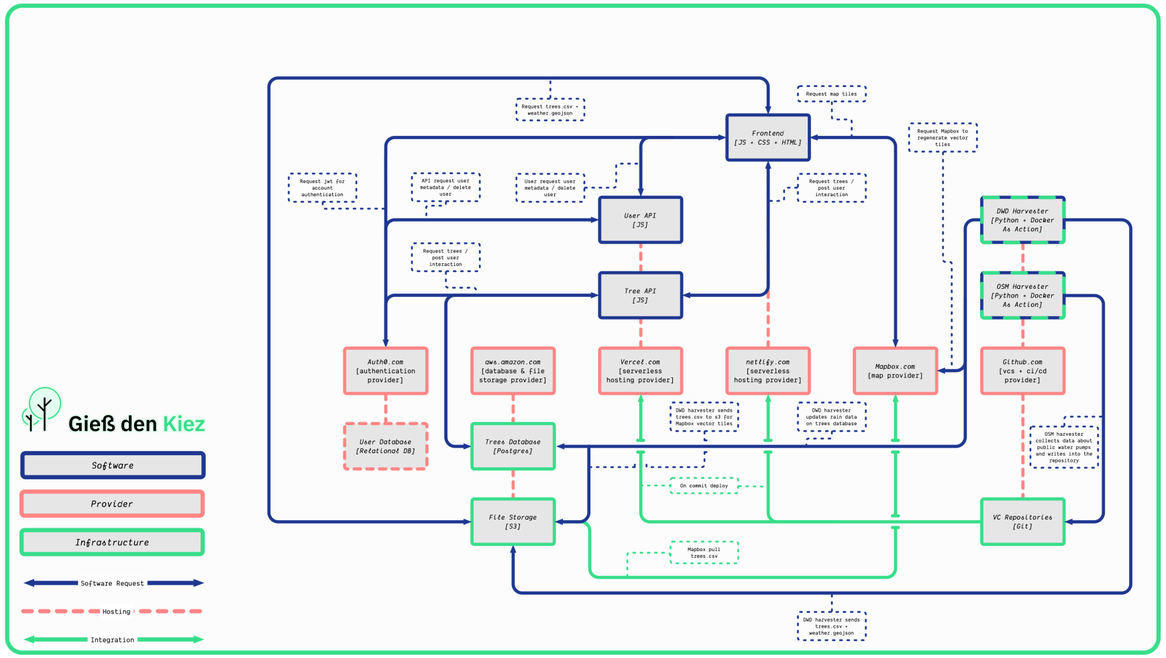
FABIAN MORÓN ZIRFAS

- Creative Technologist
- @Ideation & Prototyping Lab
- @Technologiestiftung Berlin

T.O.C.

- Backend Überblick
- Software + Services
- Wo ist der Sourcecode?
- Wie fange ich an?
- Beispiel DB & API in 6 Schritten
- Q & A

BACKEND ÜBERBLICK


Gieß den Kiez
Software
Provider
Infrastructure
Software Request
Hosting
Integration
Request trees.csv + weather.geojson
Request map tiles
Frontend
[JS + CSS + HTML]
Request Mapbox to regenerate vector tiles
Request jwt for account authentication
API request user metadata / delete user
User request user metadata / delete user
Request trees / post user interaction
User API
[JS]
DWD Harvester
[Python + Docker As Action]
Request trees / post user interaction
Tree API
[JS]
GSM Harvester
[Python + Docker As Action]
Auth0.com
[authentication provider]
aws.amazon.com
[database & file storage provider]
Vercel.com
[serverless hosting provider]
netlify.com
[serverless hosting provider]
Mapbox.com
[map provider]
Github.com
[vcs + ci/cd provider]
User Database
[Relational DB]
Trees Database
[Postgres]
DWD harvester sends trees.csv to s3 for Mapbox vector tiles
DWD harvester updates rain data on trees database
GSM harvester collects data about public wikei pumps and writes into the repository
On commit deploy
File Storage
[S3]
VC Repositories
[Git]
Mapbox pull trees.csv
DWD harvester sends trees.csv + weather.geojson

VORRAUSSETZUNGEN

Tool	Kommentar
nvm	Verwaltung von Node.js Versionen
asdf	Verwaltung von CLI Versionen
Git	Versionskontrolle
Node.js	Ausführung
AWS CLI	Abhängigkeit für Terraform
Terraform	Erzeugung von Infrastruktur
auth0.com Account	Auth Provider
vercel.com Account	Hosting Provider
netlify.com Account	Hosting Provider
mapbox.com Account	Karten Provider
AWS Account	DB und Datei Speicherung Provider
GitHub Account	VCS + CI/CD

WO IST DER SOURCECODE?

Provider	Infrastruktur	Repository (https://github.com/technologiestiftung/)
Mapbox	Karten	
Auth0	Autentifiziereng	
GitHub	Versionskontrolle & CI/CD	
~~AWS~~	~~Datenbank~~	giessdenkiez-de-aws-rds-terraform
AWS	Datei Speicherung	giessdenkiez-de-aws-s3-terraform
DWD	Regendaten	giessdenkiez-de-dwd-harvester
OSM	Wasserpumpendaten	giessdenkiez-de-osm-pumpen-harvester
Vercel	Backend Hosting	tsb-trees-api-user-management
Vercel	Backend Hosting	giessdenkiez-de-postgres-api
Netlify	Frontend Hosting	giessdenkiez-de

WIE FANGE ICH AN?

▶▶ ▶ ZUM WIKI ▶ ▶▶

BEISPIEL DB & API IN 6 SCHRITTEN

🚨 ACHTUNG 🚨

Hier könnten Drachen hausen!

1. DATENBANK ERZEUGEN

AWS ODER NICHT?

RENDER.COM

- Username
- Passwort
- Host
- Port
- Datenbank Name

ALS `postgresql` CONNECTION STRING

```
postgresql://[USER]:[PASSWORD]@[HOST]:[PORT]/[DATABASE]?schema=[SCHEMA]
```

2. AUTH0.COM API

- Audience
- Issuer
- JWKSUri

3. QUELLCODE

```
git clone https://github.com/technologiestiftung/giessdenkiez-de-postgres-api.git gdk-api
cd gdk-api
npm ci
```

3.1 ENVIRONMENT VARIABLEN

```
cp .env.sample .env
```

in .env

```
# this is for the local dev environmet
port=5432
user=fangorn
database=trees
password=ent
host=localhost
# this is for prisma - the pattern is
# postgresql://USER:PASSWORD@HOST:PORT/DATABASE?schema=SCHEMA
DATABASE_URL="postgresql://fangorn:ent@localhost:5432/trees?schema=public"
# you will find these in auth0.com
jwksuri=https://your-fancy-tenant.eu.auth0.com/.well-known/jwks.json
audience=your-audience
issuer=https://your-fancy-tenant.eu.auth0.com/
```

4. TABELLEN & DATEN

```
npx prisma db push --preview-feature --skip-generate
npx prisma db seed --preview-feature
```

5. DEPLOY

```
npx vercel
```

5.1 ENVIRONMENT VARIABLEN

```
# the user for the postgres db
npx vercel env add user
# the database name
npx vercel env add database
# the database password
npx vercel env add password
# the host of the db, aws? render.com? localhost?
npx vercel env add host
# defaults to 5432
npx vercel env add port
# below are all taken from auth0.com
npx vercel env add jwksuri
npx vercel env add audience
npx vercel env add issuer
```

5.2 DEPLOY

6. TEST

```
code --install-extension humao.rest-client
code docs/api.http
```

6.1 TEST AUTH

6.1.1 AUTH0 APPLICATION

6.1.2 ENVIRONMENT VARIABLEN

in `.env`

```
# These can be obtained from Auth0 if you create a new machine to machine
# application that has access to your API
client_id=abc123
client_secret=abc123
```

6.1.3 TOKEN HOLEN

```
code docs/api.http
```

6.1.4 AUTHENTIFIZIERTE ANFRAGE

in `.env`

```
# below varaibles are for testing the api only
# this token can be obtained by running the POST request to
# see docs/api.http for more info
# https://giessdenkiez.eu.auth0.com/oauth/token
token=a.b.c
```

Q & A

DANKE

für Ihre Aufmerksamkeit