Ismailova Dayana

Rysbekov Ansar

Amangeldinova Saltanat

ABOUT US

EcoBeko is a eco-activists oriented social network, where in addition to an ordinary online social interactions features, new ones are added!

Our goal is to build a social network with eco-specific features that could unite people, help contacting big organizations and just knowing about all the helpful activities that people around the world could share with simple articles.

There are multiple ways of doing that: thoughtful consuming, collecting/sorting trash, planting trees, helping animals, engineering agricultural lands and etc.

API consumer (or Client) is accessing the Reverse-Proxy Nginx server for resources like: static files, API

Nginx server serves static files and proxies the requests into internal REST API server

PM2 runs multiple instances of REST API (Nodes), creates single endpoint for Nginx to forward to, then it balances the requests between created nodes. Sessions (organized by Redis) and PostgreSQL are shared across all Nodes

As a UI, we choose to build a web application with Vue.js

Node.js is an obvious choice

Database

Postgres SQL is one of the most popular SQL database. It's Open Source nature brings a community, that any project definitely wants. Postgres is known to be a highly fault-tolerant and while comparing to the other SQL solutions, it's easy to setup and get start with.

API docs will be build with Swagger and will be available at /api endpoint

users

GET	/api/users/info	Get user info from token
POST	/api/users/authenticate	Authorize user
POST	/api/users	Create user
GET	/api/users	Get users by filter
PUT	/api/users/{id}	Update user

During the third phase, we had some problems
First, due to the specificity of our project, we did not have the opportunity to select a dataset, so we generated it, hence another problem occurred due to the fact that we were drawing the diagram only in the 4th phase, and in the 3rd phase it was not quite clear what kind of data. we need.

Also, initially we did not understand correctly what kind of questions we should build, and only when we figured out relational algebra did it become clear which queries we should write.