Research Duck Fed System

Functional Requirement

A scientist is trying to understand how ducks are being fed in parks around the world. She wants to collect the following information:

- What time the ducks are fed
- What food the ducks are fed
- Where the ducks are fed
- How many ducks are fed
- How much food the ducks are fed

The scientist would like to crowdsource this information by creating a web application where people can submit these data points. The scientist would like to be able to view all submissions (please note that user management/authentication is not a requirement/expectation.)

Non Functional Requirement

- Able to support people around the world (Scalable)
- Apply Security Policy
- Automate Deployment
- Code test for E2E Testing e.g using Cypress

Capacity Estimation

Request

Countries in the world: 195
Average Users per country: 1000
Average Request per day: 195 * 1000 = 195,000
Average Request per second: 195,000 / 86400 = 2.25/s ≈ 3/s
Estimate Retrieve data average 10 page per day

Storage

Document size ~16MB
Transaction per day: 195000 * 16MB = 3.12TB
Transaction 5 years: 5616TB = 5.6PB

Bandwidth

195000 * 16MB / 86400s = 36.1 MBps

API design

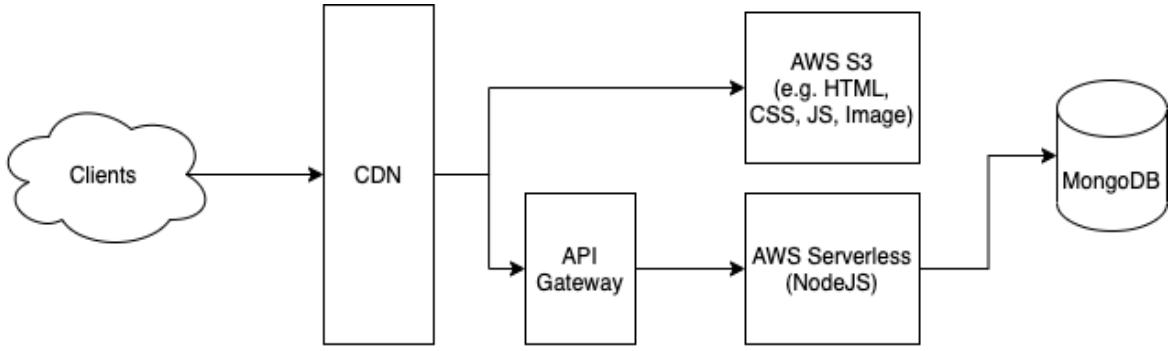
GET feed
POST feed

Data model/schema

Duck_Fed

- datetime
- Location (lat, lng, address)
- duck_number
- food_type
- food_weight
- created_date

System high level system design



Architectural design

NextJS

Builds on top of React. It's easy one to work with dynamic route, data fetching, layouts, image optimizing, SEO etc. Learn more

NextJS API

Our website is a simple request server-side to retrieve data. NextJS helps us to easily develop. Able to deploy as serverless functions with AWS easily. Learn more

MongoDB

RDBMS is a good one but has difficulties scaling with sharding. For saving a lot of research data. I think NOSQL is more suitable. That's why I use document stores - MongDB with flexible schemas. Learn more

REST API

Even Though REST API has fixed data structures. It's not flexible like GraphQL. But our web application is less data structures. Lest API endpoint. I think REST API is Easy to understand and implement. Learn more

Appendix

-