

API DESIGN

using REST + GraphQL

ENDPOINTS

GitHub token input
Repo input (url)

get set

Outputs

License
Average Bug Time
Num Contributors
Correctness Metric

http get, http post(?), set, get

other needs

source (npm, Github?)
is valid URL
stored { URL
Alt { URL
License Bool
Name?
UUID ID
last updated

Indexable List of URL structs

CRUD + Query

Calling APIs in Go using REST + GraphQL:

Owners: Anna (Ben)

Time: 6 hours from Anna

Success Criteria: Given a GitHub personal token (classic) and a GitHub repository link, return and parse all the APIs to get the following information:

License of the repository, Average time bug tickets are opened, Number of contributors in the past year.

entities

url (holds outputs, stores repos in
github)

url

api (CRUD)
register handlers?

struct resource
serve
logger

repo

another CRUD??
interface??

New Repo func

CRUD + Query + Count

service

interface of Query?

add data structs (typed)

e.g.

type URL struct {
entity URL

}

Validate?

another CRUD?