

COMP 4959: Lab Assignment 4

The purpose of this exercise is to provide a JSON REST API to a supervised `Mydb` server using the Phoenix framework.

For this lab, you are asked to turn the `Mydb` server from Lab 3 of COMP4958 into a supervised server that is automatically re-started when it crashes. However, you do not need to create a “store” to back up the server. Note that in order to be supervised, the server must be registered (as its PID changes when it is re-started). Put the implementation in the module `Mydb.Server`.

Besides the usual functions to start/stop the server, `Mydb.Server` needs to provide the following functions:

```
store(key, value) => :ok
find(key) => value | nil
match(value) => [key1, ..., keyN].
list() => [{key1, value1}, ..., {keyN, valueN}]
```

The first 3 functions are basically the same as before. Note, however, that there is a change in the return value of `find`. The fourth function returns a list of all key-value pairs.

For the REST API, each of the above operations is mapped to a GET/POST request to a URL as follows:

```
POST  /api/db                (store)
GET    /api/db/find/<key>     (find)
GET    /api/db/match/<value>  (match)
GET    /api/db               (list)
```

The following interaction shows some requests & responses, assuming the server is listening at port 4000 on localhost:

```
$ curl http://localhost:4000/api/db/
{"data": [{"key": "bart", "value": "simpson"}, {"key": "lisa", "value": "simpson"}, {"key": "monty", "value": "burns"}]}
$ curl http://localhost:4000/api/db/find/monty
{"data": "burns"}
$ curl http://localhost:4000/api/db/find/homer
{"data": null}
$ curl http://localhost:4000/api/db/match/simpson
{"data": ["bart", "lisa"]}
$ curl -X POST http://localhost:4000/api/db -H 'Content-type: application/json' -d '{"key": "homer", "value": "simpson"}'
{"data": {"key": "homer", "value": "simpson"}}
$ curl http://localhost:4000/api/db/find/homer
{"data": "simpson"}
```

Note that, in order to make the output clearer, a newline character has been added after each response.

Provide some tests for your JSON API.