encrypt username/pw (e, ω)

user, pass

4 logins
↳ Check if any are the same?

Probably should do:

clients / capital one ⟶ this can be done at top?

Potentially release a bunch of libraries?
↳ Need to think how they are used

At the end of the day, it's the clients responsibility to conform to what the server can do.

So lets work on the server :

Will need to have tests too. Anyway

- Lets think
  - ↳ will need to get :
    - Institution ⟶ Capital One - directly
                  ⟶ Plaid - through plaid item
    - Plaid Items
    - Accounts
    - Transactions

Okay, sure, but... What is it going to look like?

Declare interface which all must support
  - Get (requested from server)
  - Set (requested to server)

Probably should try Nested Templates
| ↳ What defines an institution, make sure value is there.
| x
↓
Lets see...

Really this is part of hyperverse
   Should be hyperverse / financials / connector
                                      / client
   |

↓

Need to be extended while still conforming to <u>Entity</u>

↓

How to have plaid_item_id, but as metadata?

Every metadata/application gets its own tables? Hmmm...

When we get:

Should use other GQL library?

　No it's way is just as ugly. Makes sense but still ugly.

　Not sure how I would do different yet...

Maybe　let's not make too generic just yet.

Agh.

└→ Queries are specific to clients.

　└→ However clients shouldn't maintain structs.

　　└→ Define structs then use in client for now.

Not very happy right now. Want more done.

　• Capital One Tx data in.

　• Plaid Tx data in.

　No more thinking of abstractions.
　Just get it in.

Really do want to do that.

Also want to get some level of script.

　└→ GitHub static site generator?

　　For now no. But let's think about that
　　going forward.

1. On save .md ⟹ .html w/ proper formatting.
   ↳ inotifywatch
      ↳ on change

On file save
   ↳ pandoc

Really pub-sub?

Basically IPC saying I'll subscribe to these updates but you publish them.
this establishes intermediary representation.