# 03.06.24

1. [John] Struggle with introducing new concepts and keeping up with older paradigms, half-baked solutions (i.e. inconsistency python tests, overlap between integration and unit tests)
    a. On-going technical debt being paid
    b. Switch to pytests and fixtures
    c. Struggle to understand what is used code and what is unused
    d. Pylint rules
    e. [Diego] Created a backlog item in Todo in Devex & Quality Board
2. [Justin] pre-commit hook to check staged files for typescript errors
3. [Max]
    a. pip-compile-multi is broken https://github.com/apache/superset/pull/27404
    b. should we drop pip-compile-multi (are there alternatives?)
    c. Go back to a simpler setup with pip-compile and see if that is viable
    d. Get dependabot to support pip-compile-multi
4. [Max] https://github.com/apache/superset/issues/26997
    a. Ideally, two setup docker-compose up (deterministic on the branch), and docker-compose interactive that would build the docker deterministic to the branch and mounts a local folder to run deps locally
    b. Improve the docs

# 10.18.22

1. API to Client contracts
    a. inv

# 08/26/22

1. [Hugh] Client to API contracts (thrift, protos, etc)
    a. Having a more robust framework to solidify data we are passing from the backend to the frontend.
    b. [Eric] manual ways on finding ways source of truth on
    c. [Arash] annotating the fields so users know what it represents
    d. [John] are there mechanism for defining the objects once
        i. Can we not be a airport terminal and actually close out
            1. Moving logic from the client (owner checks) to the backend?
            2. With this project can we really close this out and make 1 API the source of truth vs. having n different ways of doing something
            3. Is graphql an option? (might be controversial [Evan])
        ii. [John] What is our north star when it comes developing this picture

1. [Eric] we need to break down the problems we are trying to solve technology aside

iii. Offsite for doing a deep dive on projects

# 06/19/22

2. [Hugh] Write a SIP on enhancing the backend test suite
3. [John] Github workflows vs. tox
    a. Local and remove development is different at the moment
    b. Tox has the github workflow plugins to make more standardized for
4. [John] Using new system help with managing package dependencies
    a. Using poetry for example
5. [Eric] Working FE design system structure to fix circular dependencies
    a. Can we make CI not run when for certain file changes

Action Items:

1. [John] to write SIP on standardizing GH actions with local development
2. [Hugh] Write a SIP on enhancing the backend test suite
3. [Hugh] Move tickets to github project

# 05/31/22

- [Hugh] Breakdown of upgrading testing suite
    - First task we are working on
    - What is the difference between unit vs integration test. We need to clearly define this for developers (specifically in our documentation)
    - Find doc that clearly states what roles have what permissions
        - Like if i'm admin what am i able to do and if i'm a gamma what can i do?
        - Lets add this to the docs to clearly have alignment
        - ```
          # This is the sqlite/mysql syntax. GROUP_CONCAT has different but
          equivalent syntax in postgresql - string_agg().
          select avm.name view_menu, ap.name permission,
          group_concat(ar.name, ', ') "role"
          from ab_permission ap
          join ab_permission_view apv on ap.id = apv.permission_id
          join ab_view_menu avm on avm.id = apv.view_menu_id
          join ab_permission_view_role apvr on apv.id =
          apvr.permission_view_id
          join ab_role ar on ar.id = apvr.role_id
          group by avm.name, ap.name
          order by "role", avm.name, ap.name;
          ```
    - How are we thinking about the file structures when adding testing
        - superset/a/b/c.py -> tests/unit/superset/a/b/c.py
        - Is there a standardize way of putting test in certain places?
          ```

        - How are we handling namespaces? (look into how pytest recommends)
- Action Items
    - [Hugh] Breakdown epic into smaller task + make a public project for this

# 05/17/22

- [Hugh] Superset x GH Codespaces
    - putting together a wiki on how you could setup codespace as your dev env
    - Can we leverage better data examples for this as well or setting up “superset” DB connection
- [Jesse] Running integrations test messing up examples db
    - Sometimes it adds data and doesn’t rollback which is affecting the examples db
    - This is altering the state of the app which is making development hard
    - Some tests are using examples, can we rewrite them so they are encapsulated and not using db examples (used for dev + manual testing)?
        - Make sure all test have a rollback when doing a create/POST
    - Investigate what is the proper pattern for integration test and setting up rollbacks
        - Can we use fixtures just like the unit_test examples
    - We need to prioritize test before we do a rewrite and allow us
        - Get test suite to run locally with no errors
        - Audit which test don’t have rollbacks properly set
            - Can these test be moved to unit test?

# 03/22/22

- [Arash] Adding search to docusarus
- [Micheal] Review for best practices for documentation
    - Everyone one take time to review this
- Remove official docker image since Community Growth is going to own this
    - Do we need to add official images to be in AWS or Google Cloud
- [Hugh] to switch meeting time to thursday at 11am EST (8am PST)

# 03/08/22

- Working through breaking down task in operation doc
- [Evan] try and figure out how 1% better every time we leave these meetings
- Added more task to the board and continued to breakdown other task to be more actionable
    - Most of the task were focused around updating documentation and creating wikis to make it easier to debug superset.

Click here to view initial task breakdown doc