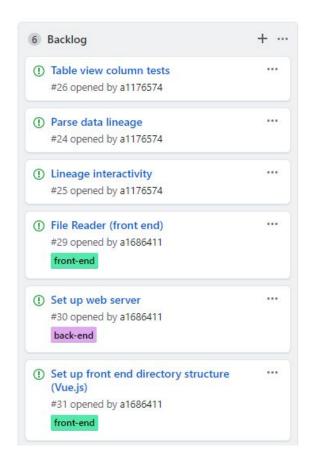# Task Board Snapshot

6   Backlog   + ...

(!) Table view column tests   ...
#26 opened by a1176574

(!) Parse data lineage   ...
#24 opened by a1176574

(!) Lineage interactivity   ...
#25 opened by a1176574

(!) File Reader (front end)   ...
#29 opened by a1686411
`front-end`

(!) Set up web server   ...
#30 opened by a1686411
`back-end`

(!) Set up front end directory structure (Vue.js)   ...
#31 opened by a1686411
`front-end`

4   Sprint Backlog   + ...

(!) Group Meetings and Team Member Roles   ...
#16 opened by a1706579
`reports`
⍦ Initial Report Submission

(!) Team Report Drafting   ...
#17 opened by a1706579
`reports`
⍦ Initial Report Submission

(!) Organise ongoing meetings   ...
#6 opened by a1686411
`project-management`

(!) Parse query files   ...
#7 opened by a1704822

12   In Progress   + ..

(!) Week 3 Snapshot   ...
#10 opened by a1704822

(!) User Base   ...
#13 opened by a1706579
`reports`
⍦ Initial Report Submission

(!) Customer Q&A   ...
#12 opened by a1706579
`reports`
⍦ Initial Report Submission

(!) Select web server technologies   ...
#2 opened by a1686411
`back-end` `initial-setup`
⍦ Select Architecture

(!) Generate decision matrix for technologies (back end)   ...
#9 opened by a1704822
`back-end` `project-management`
⍦ Select Architecture

(!) Tech Stack and Standards   ...
#15 opened by a1706579
`reports`

Task Board Snapshot

12 In Progress

⚠ Configure GitHub to only merge when all unit tests pass ···
#21 opened by a1686411
`initial-setup`
🏷 Configure GitHub

⚠ Project Vision ···
#11 opened by a1706579
`reports`
🏷 Initial Report Submission

⚠ Generate decision matrix for technologies (front end) ···
#8 opened by a1704822
`project-management`
🏷 Select Architecture

⚠ Select front end frameworks ···
#3 opened by a1686411
`initial-setup`
🏷 Select Architecture

⚠ Gather questions to send to Propic via Ines ···
#19 opened by a1686411

⚠ Set up OverLeaf integration with GitHub ···

7 Done

⊘ Detail Software Architecture ···
#14 opened by a1706579
`reports`
🏷 Initial Report Submission

🗒 Moving issues into this column has been ··· automated - no need to manually drag and drop, now when an issue is closed, it will automatically be moved here.

Added by a1686411

⚠ Create Initial Report ···
#1 opened by a1690417
`reports`
🏷 Initial Report Submission

⊘ Set up a virtual environment for the repo ···
#20 opened by a1686411
`documentation`

1 linked pull request ⌄

⊘ Decide on definition of "done" ···
#5 opened by a1686411
`initial-setup` `project-management`

⊘ Set up unit testing ···
#4 opened by a1686411
`initial-setup`

1 linked pull request ⌄

⊘ Configure GitHub review policy ···
#18 opened by a1686411
`initial-setup`
🏷 Configure GitHub

User Stories In Scope for Sprint 1

Sprint 1 Goals

- To configure the GitHub repository, and set up the project backlog with tasks.
- To develop the necessary backend to receive and parse an SQL script to output the associated columns, tables and views.
- To compose a preliminary report describing the proposed architecture and workflows.

Primary Tasks:

- Configure python virtual environment for the repository.
- Set up unit testing with a minimal working example.
- Generate decision matrix for frontend and backend frameworks.
- Select frontend and backend frameworks.
- Determine coding standards.
- Parse query files.

Statement of work completed

The work items and tasks completed up to 1700 hours on 14/08/20 is shown in the task-board snapshots, and detailed further here. The primary goal for week 1 of sprint 1 was to configure all project management tools and resources including:

- Initialising the git repository.
- Creating a backlog of tasks and issues
- Agreeing upon and setting recurring sprint meetings (kick-off and retrospective) and stand-ups.

The frontend decision matrix was developed assessing three main technologies: Angular, React and Vue. These technologies were assessed on 6 criteria. From this it has been decided that Vue will be used. Coding standards have also been agreed upon and style guides will include PEP for Python and Airbnb for JavaScript with corresponding linters being configured.

Additionally, we organised an all hands for week 2 of sprint 1 to determine how to parse query files, and also aimed to break the files down and create a set of tasks to complete.

Work Remaining

- Break down the 'parse query files' task
- Stand up backend to retrieve input SQL script and produce appropriate output.
- Complete initial report

Blockers

- Github wiki down (minor)

Definition of done

For an issue to be considered to be done, where appropriate and necessary it must pass all linting, be reviewed by at least two other team members, unit tests, and have appropriate test coverage. We nominally aim for at least 80% test coverage of code.