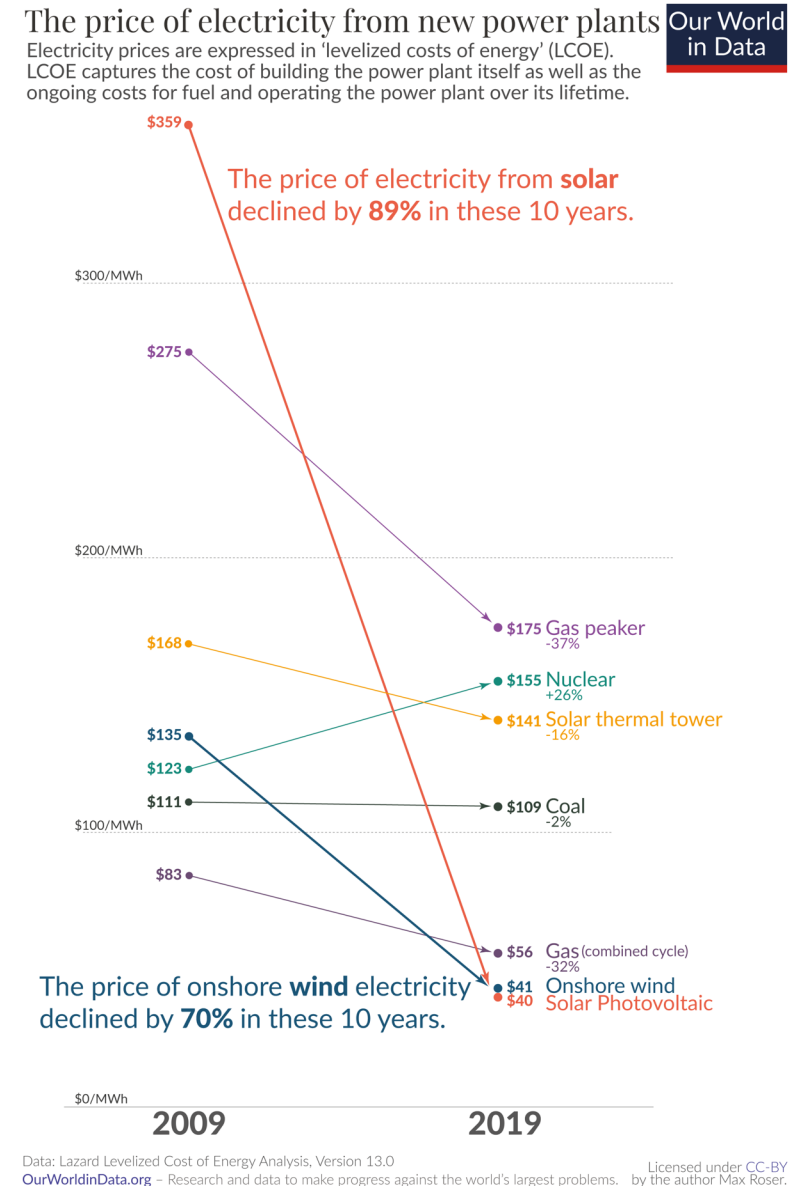# Bank.Green for Engineers

Q1 2023
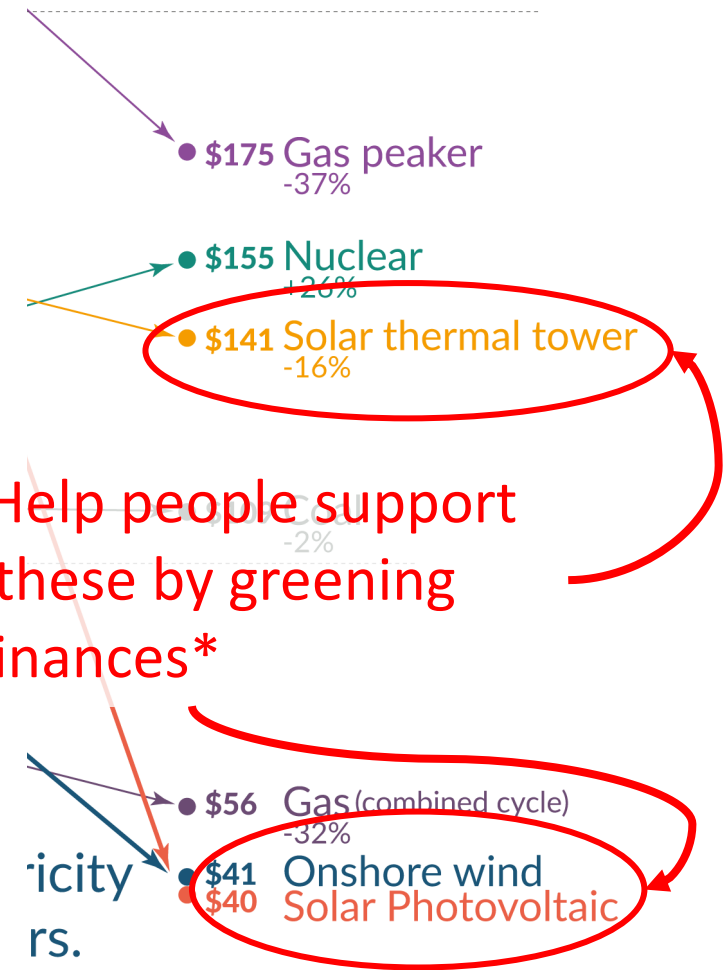
Bank.Green Vision

- To accelerate the advent of a sustainable future by…
  - evaluating and promoting green banks and their products,
  - protecting consumers [from greenwashing], while
  - mobilising them to fight climate change by greening their finances

The price of electricity from new power plants
Electricity prices are expressed in 'levelized costs of energy' (LCOE).
LCOE captures the cost of building the power plant itself as well as the
ongoing costs for fuel and operating the power plant over its lifetime.

Our World in Data

$359 — The price of electricity from **solar** declined by **89%** in these 10 years.

$300/MWh

$275

$200/MWh

$175 Gas peaker -37%
$168
$155 Nuclear +26%
$141 Solar thermal tower -16%
$135
$123
$111
$109 Coal -2%
$100/MWh

$83

$56 Gas (combined cycle) -32%
$41 Onshore wind
$40 Solar Photovoltaic

The price of onshore **wind** electricity declined by **70%** in these 10 years.

$0/MWh

2009   2019

Data: Lazard Levelized Cost of Energy Analysis, Version 13.0
OurWorldinData.org – Research and data to make progress against the world's largest problems.
Licensed under CC-BY by the author Max Roser.

Bank.Green Vision

- To accelerate the advent of a sustainable future by…
  - evaluating and promoting green banks and their products,
  - protecting consumers [from greenwashing], while
  - mobilising them to fight climate change
    - By greening their finances

$175 Gas peaker
-37%

$155 Nuclear
-20%

$141 Solar thermal tower
-16%

-2%

Tl;dr: Help people support these these by greening their finances*

$56 Gas (combined cycle)
-32%

$41 Onshore wind

$40 Solar Photovoltaic

ricity

rs.

* This chart is only for electricity, but also applies to other industries. What really matters for energy and other industries is the relative cost of polluting industries vs their sustainable counterparts.

"to accelerating the advent of a sustainable future…"

Starting with bank loans to the energy sector

- Banks provide the majority of energy financing
  - True both for fossil fuels and sustainable energy
- We want
  - More bank loans to sustainable energy
  - Fewer bank loans to fossil fuels

How to achieve? (theory of change)

- **The obvious thing (doesn't work): Consumer pressure from bank switching**
  - Moving accounts is slow and difficult
  - Relatively few people will do it
  - Banks' marketing departments sometimes care. Their finance departments do not.
- **The non-obvious solution: Demonstrating business feasibility**
  - Big banks lend to fossil fuels because they feel that they need to for profitability
  - Bank.Green needs to demonstrate that it's more profitable to lend sustainably
- **How to demonstrate business feasibility**
  - Verify when "green" finance is truly green
    - Fossil free certification for banks
    - Exclude and ostracize greenwashing
  - Facilitate green lending
  - Show that green finance wins customers
  - Publish and publicize results

Typical tech volunteering at non-technical nonprofits

Requests by day:

1. "Fix the printer"

2. "Fix Excel macros"

3. "Make an app encouraging 5000 people to spontaneously organise"

4. Hack the CIA

No In-between

Many non-technical collaborators imagine that you think, look, smell, and will act like this 🙁 ->

Bank.Green Tech Volunteering

Use your hard skills

Learn new skills & technologies

See your impact

Most effective use of your effort

Tech Setup

Bi-Weekly Plannings	Groomed Backlog
Backend: Python Django + SQLite	Frontend: Vue.js + Nuxt
GraphQL API	Open-Source Repos, Open Data

Upcoming Initiatives

TESTIMONIALS

BETTER BANK RATINGS

BANK CUSTOMIZABILITY

Testimonials

- Users tell us about their bank switching experience
  - From what bank to what bank?
  - Why switch?
  - Pain points of switching
  - How is the new bank?

- Bank.Green shares data with banks on…
  - Customers leaving/joining for environmental reasons
  - Products their customers wish they had (Fossil Free Alliance banks only)
  - About their user experience (Fossil Free Alliance banks only)

- Tech Work:
  - Mostly frontend (vue.js) with typeform integrations

Testimonials (Tech Details)

- Data collection via typeform (embed if possible)
- Various ways to enter typeform
  - Bottom of "great" rated bank pages
  - Contact us section
  - Etc
- Mostly vue.js work

Better Bank Ratings

- Updated system for rating banks
  - What organizations control lending with customer deposits?
  - Who do they lend to?
  - How environmental is their lending criteria?

- Bank.Green displays some of data on its website

- Tech Work:
  - Finance & Admin: Devising systems to collect
  - Backend: Assigning ratings

Bank Ratings
(Tech Details)

- Much work depends on how the ratings are designed (in progress)

- Some data will need to be collected from the outside world. In the backend, this likely means:
  - Additional record attributes
  - Improved tools for merging and associating records
  - Repetitive jobs will likely be required to collect and associate data

- Displaying this data on the frontend (Vue.js)
  - This will bleed somewhat into the next initiative. Some data should still be displayed in this initiative though

Bank Customizability

- Allow editors to add custom information on more discoverable bank subpages
  - Highlight positive or negative actions
  - Highlight unique services that green banks offer
  - Highlight accredidations (i.e. B-Impact)
  - Easier editing via headless CMS
  - Tweaks for bots

- Tech Work:
  - Frontend: Tighter integration (vue.js) with prismic CMS
  - Frontend: Higher lighthouse scores for bank pages (vue.js + nuxt.js)
  - Backend: Communication between Django and headless CMS (prismic.io)

Bank Customizability (Tech Details)

- Two pieces of work:
  - connecting the python-django backend with prismic.io headless CMS
  - A design framework for what may need to be added to bank pages

- Some loadtime and lighthouse score improvements are included here, but can be moved to other initiatives if necessary
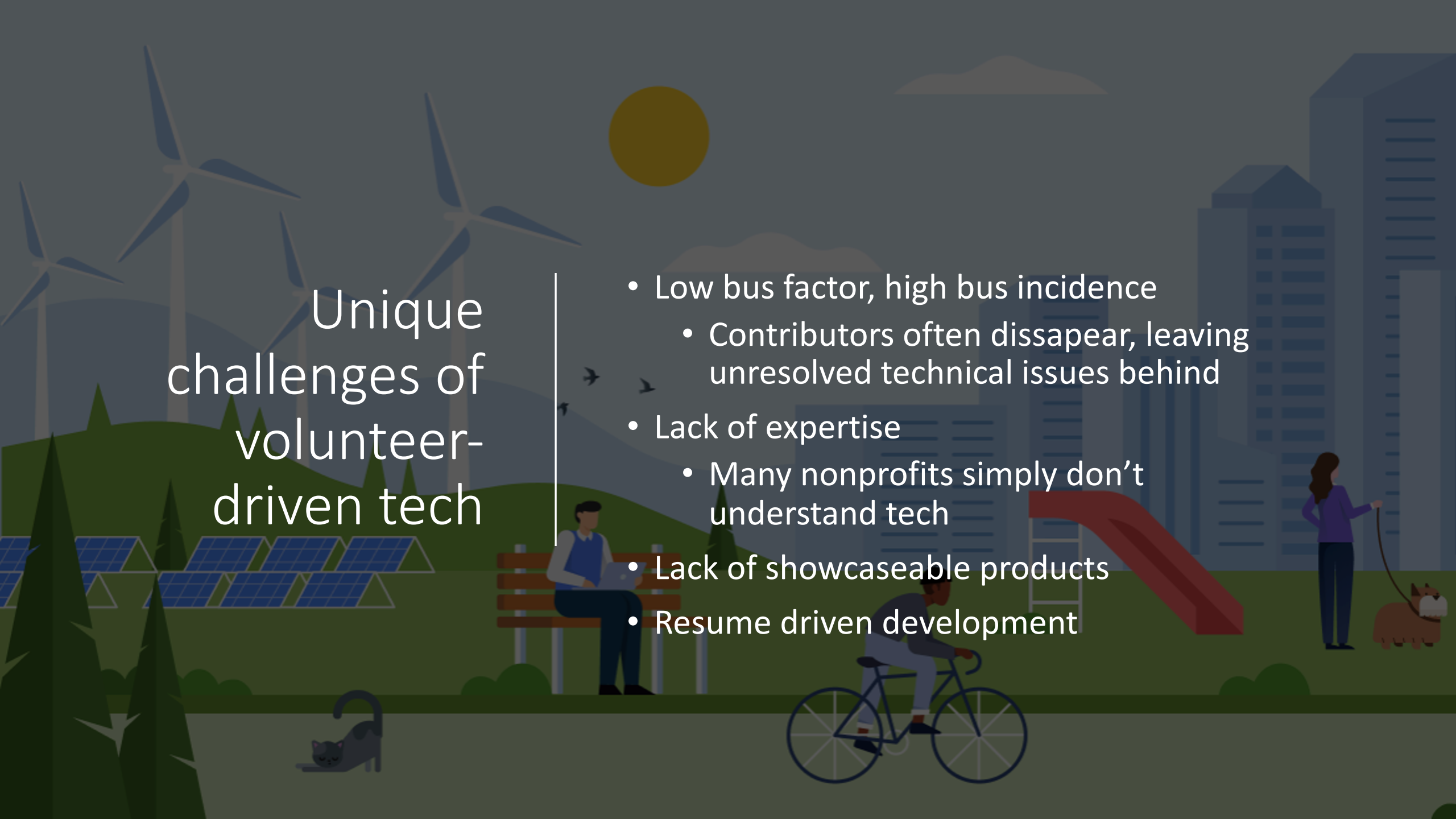
Bank.Green Engineering Team

Challenges

Guidelines

Vision

Unique challenges of volunteer-driven tech

- Low bus factor, high bus incidence
  - Contributors often dissapear, leaving unresolved technical issues behind
- Lack of expertise
  - Many nonprofits simply don't understand tech
- Lack of showcaseable products
- Resume driven development

Bank.Green Guidelines for Engineering Culture

Delivery includes collaboration and explanation
- Explaining your work to at least one colleague
- Technically present large changes to colleagues

Embrace ignorance
- Ask questions frequently in Slack
- Mark them resolved as necessary
- Err on the side of over-asking

Teach
- Share what you learned, even if it's redundant
- Repeat yourself
- Err on the side of repetition and redundancy

Communicate frequently
- Request and offer help/calls when possible
- Say when you will respond
- Prioritize synchronous communication when possible

Be proud of your work
- Feel free to add yourself to the Bank.Green team page,
  - link to your profile/website, and
  - link your profile/website to Bank.Green

Choose well-known technologies
- Boring and bulletproof > cutting-edge future abandonware
- Use Bank.Green as an opportunity to learn proven but new-to-you tech

Promise less. Deliver more
- undelivered promises have negative value
- Keep your promises manageable and deliver more if possible.

Engineering Team Vision

- Staffing
  - 2-3 high-availability devs
  - 2-3 more limited availability devs
  - 1 product owner to help with productization and prioritization
  - 1 UX designer (high availability, low hours)
- Technology
  - Continue with Vue.js-Nuxt, Python-Django, SQLite, and Prismic.io CMS
  - Potential for additional data ingestion and processing for financial ratings
- Rituals
  - Review + Sprint planning 2x per month (recommended attendance)
  - Retro 1x per month (recommended attendance)

Collaboration

- Active and friendly Slack
- Easy to ask questions of high-availabiliy devs
- Video and voice call

Developer Availability & Expectations

	Inactive Dev	**Limited Availability Dev**	**High Availability Dev**
Responsibilities	- 0-2 hr /week - answer dev questions on areas of expertise	- 2-8 hours / week - <u>initially</u>: completing low-priority non-blocking tickets - <u>later</u>: completing higher priority and potentially blocking tickets, possibly owning low-priority projects	- 8-20 hours / week - <u>initially</u>: populating backog, planning sprints, completing high priority and potentially-blocking tickets - <u>later</u>: owning high priority projects
Tooling & Access			
Github	Write Access	Write Access <u>expectation</u>: - push branches, open PR's	Maintain Access <u>expectation</u>: - all of the previous - review, merge into main, and deploy
Slack	Member	Member <u>expectation</u>: - respond to DM's and and when tagged - read and react (using emojis) to channel messages	Member <u>expectation</u>: - all of the previous - read, react, and respond to channel messages
Linear	No Access	Member <u>expectation</u>: - own tickets	Admin <u>expectation</u>: - all of the previous - own projects
Meetings			
Retros	optional	optional <u>expectation</u>: - proactively communicate issues impacting you to high availability devs before meeting	recommended but not required <u>expectation</u>: - proactively communicate issues impacting you and other developers to someone attending the meeting or in the meeting directly
Planning + Refinement	optional	optional <u>expectation</u>: - communicate and commit to velocity/availability before planning - Add unrefined tickets to backlog	required <u>expectation</u>: - Add refined tickets to backlog - participate in engineering discussion - actively refine tickets in owned projects