How to build software, not write code

The non-coding skills are at least as important as knowing how to code

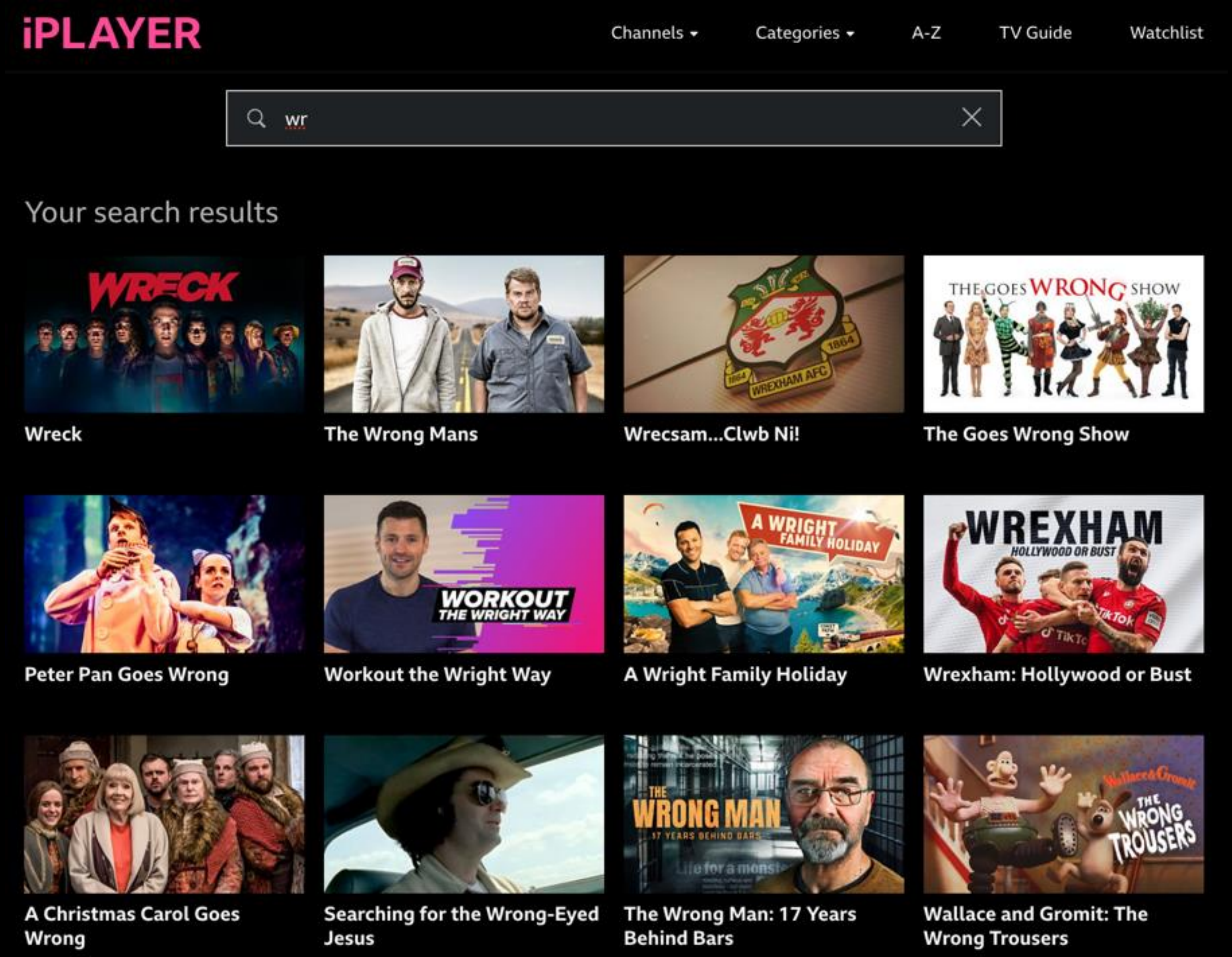
(maybe more)

Hi! I’m Will

Maru

Mochi

BBC



wr

Your search results

Wreck

The Wrong Mans

Wrecsam...Clwb Ni!

The Goes Wrong Show

Peter Pan Goes Wrong

Workout the Wright Way

A Wright Family Holiday

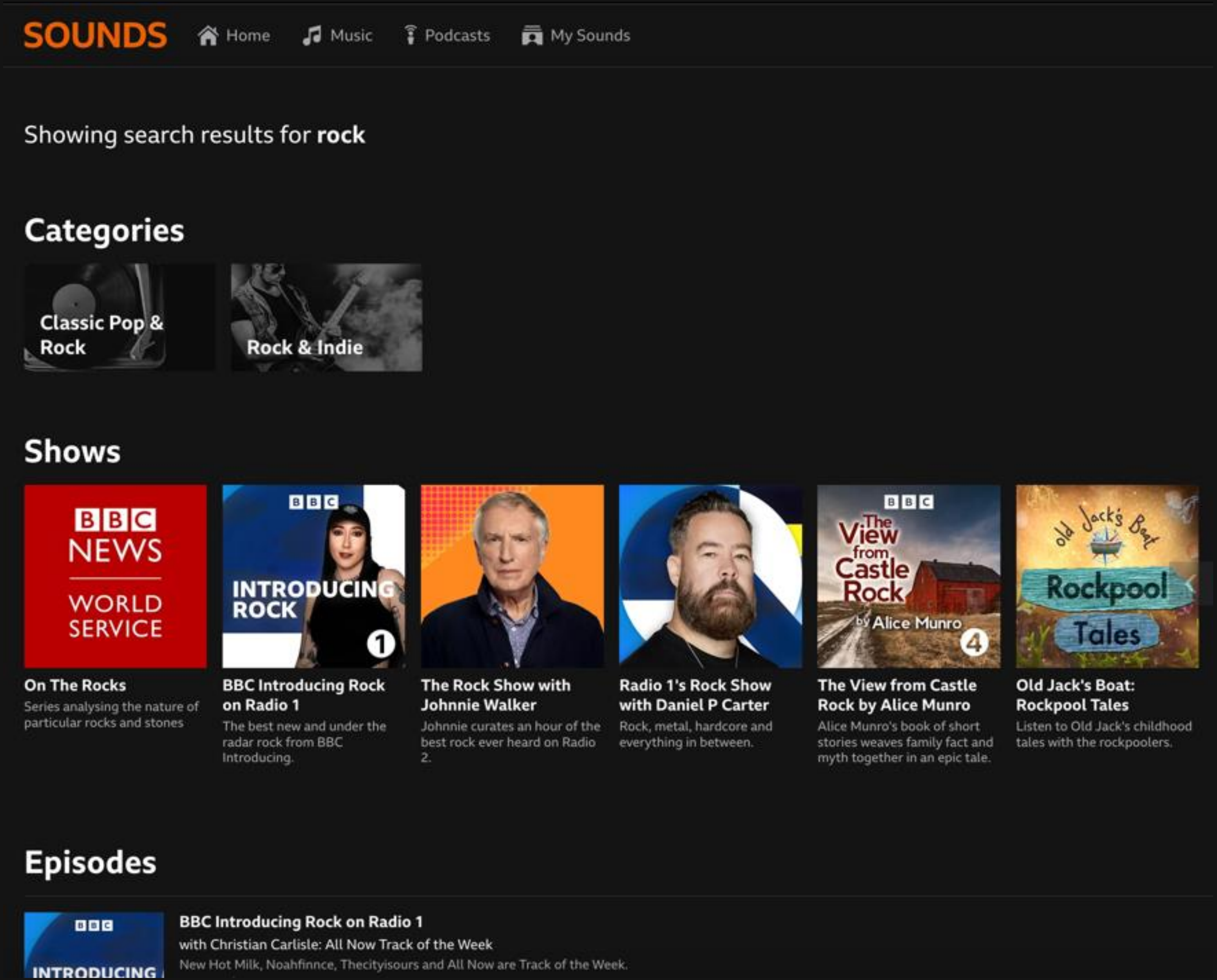
Wrexham: Hollywood or Bust

A Christmas Carol Goes Wrong

Searching for the Wrong-Eyed Jesus

The Wrong Man: 17 Years Behind Bars

Wallace and Gromit: The Wrong Trousers



Showing search results for **rock**

Categories

Classic Pop & Rock

Rock & Indie

Shows

On The Rocks
Series analysing the nature of particular rocks and stones

BBC Introducing Rock on Radio 1
The best new and under the radar rock from BBC Introducing.

The Rock Show with Johnnie Walker
Johnnie curates an hour of the best rock ever heard on Radio 2.

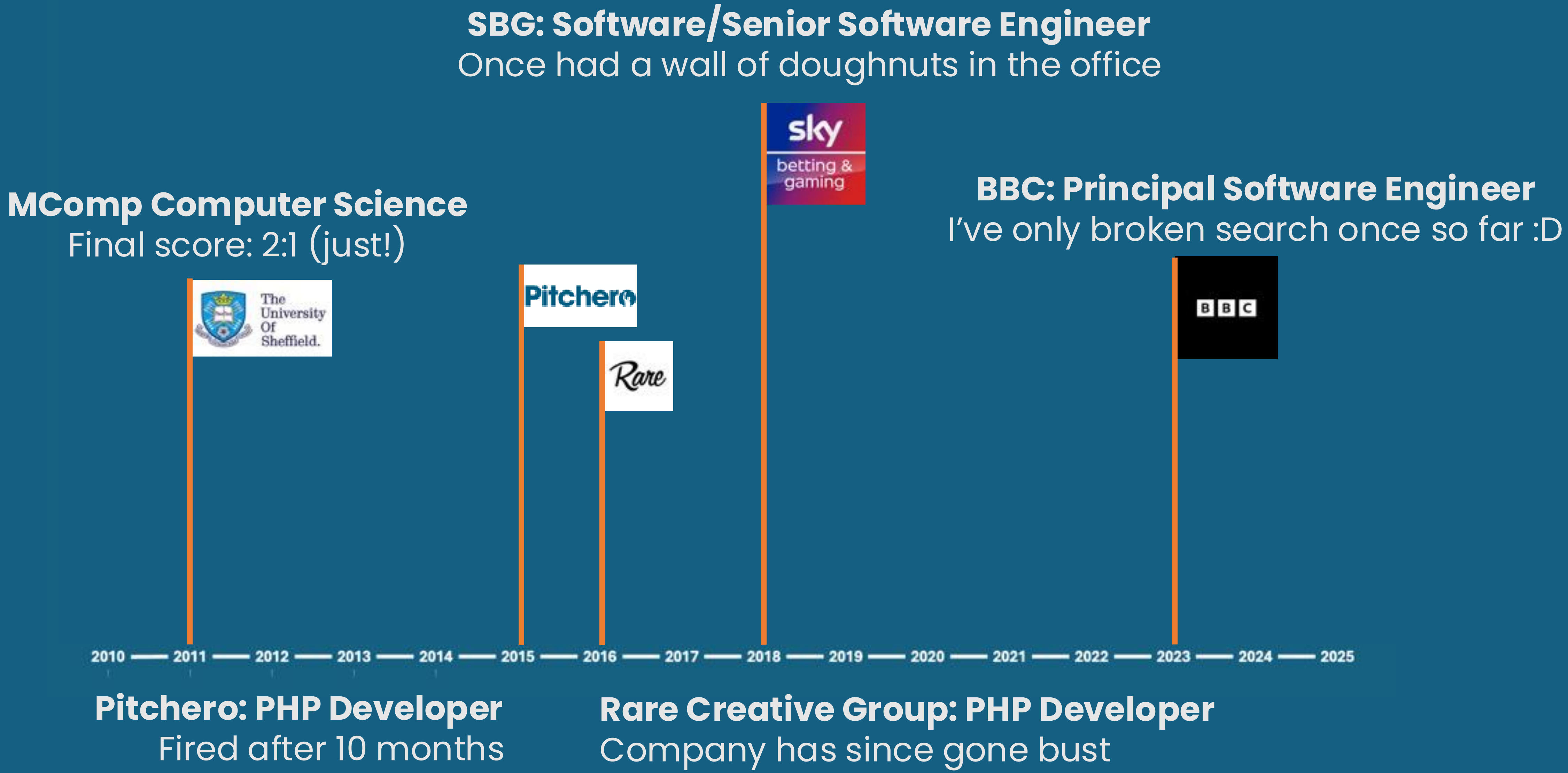
Radio 1's Rock Show with Daniel P Carter
Rock, metal, hardcore and everything in between.

The View from Castle Rock by Alice Munro
Alice Munro's book of short stories weaves family fact and myth together in an epic tale.

Old Jack's Boat: Rockpool Tales
Listen to Old Jack's childhood tales with the rockpoolers.

Episodes

BBC Introducing Rock on Radio 1
with Christian Carlisle: All Now Track of the Week
New Hot Milk, Noahfinnce, Thecityisours and All Now are Track of the Week.



SBG: Software/Senior Software Engineer
Once had a wall of doughnuts in the office

MComp Computer Science
Final score: 2:1 (just!)

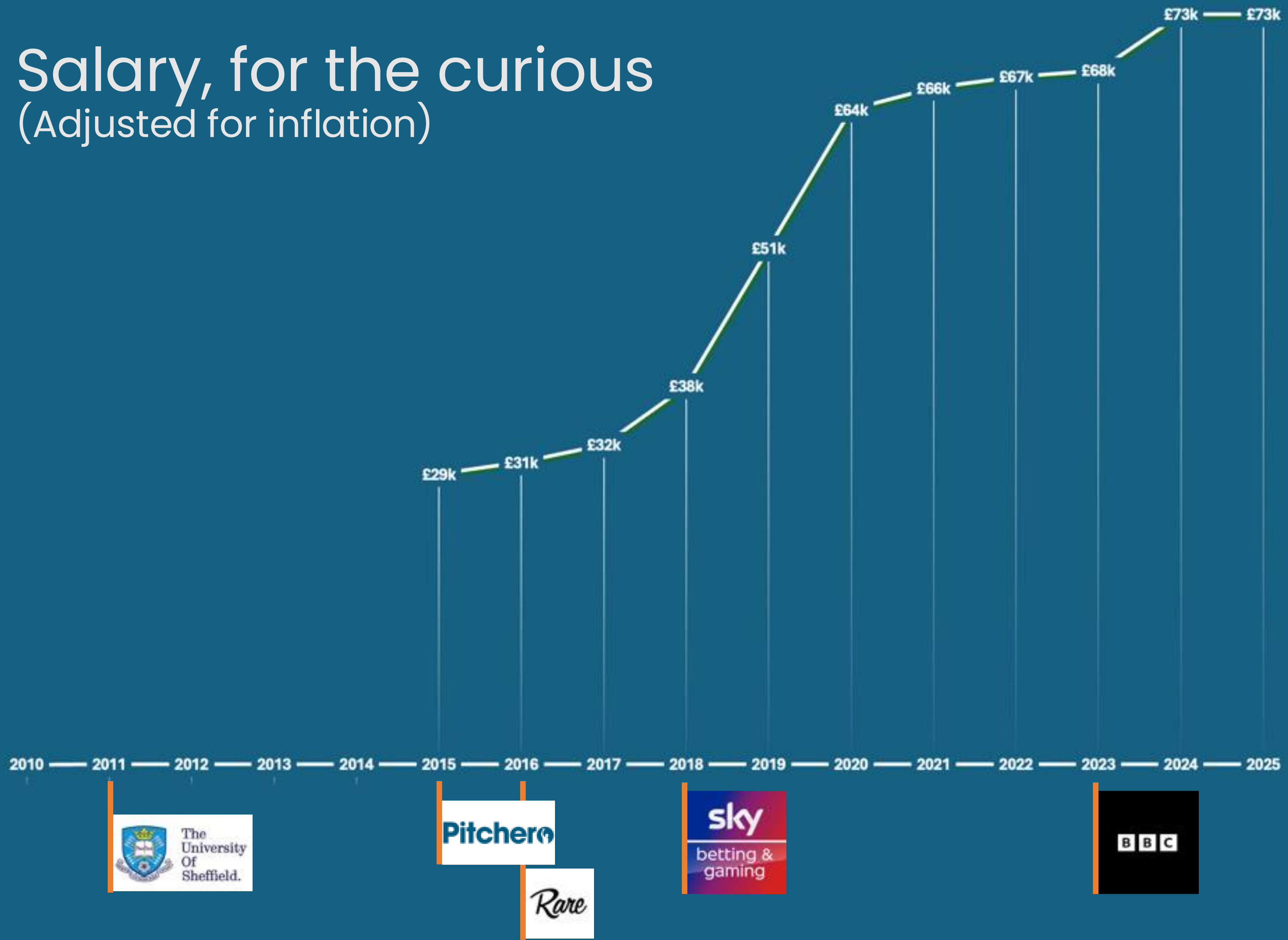
BBC: Principal Software Engineer
I've only broken search once so far :D

Pitchero: PHP Developer
Fired after 10 months

Rare Creative Group: PHP Developer
Company has since gone bust

Salary, for the curious

(Adjusted for inflation)



findyourjam.team

Open source team building tool

dotwo.games

My games jam projects and prototypes

Squarepinski

(Under development)

Before we begin, two definitions

Pragmatic

dealing with things sensibly and realistically;
based on practical rather than theoretical considerations

Dogmatic

inclined to lay down principles as undeniably true;
black-and-white view on things

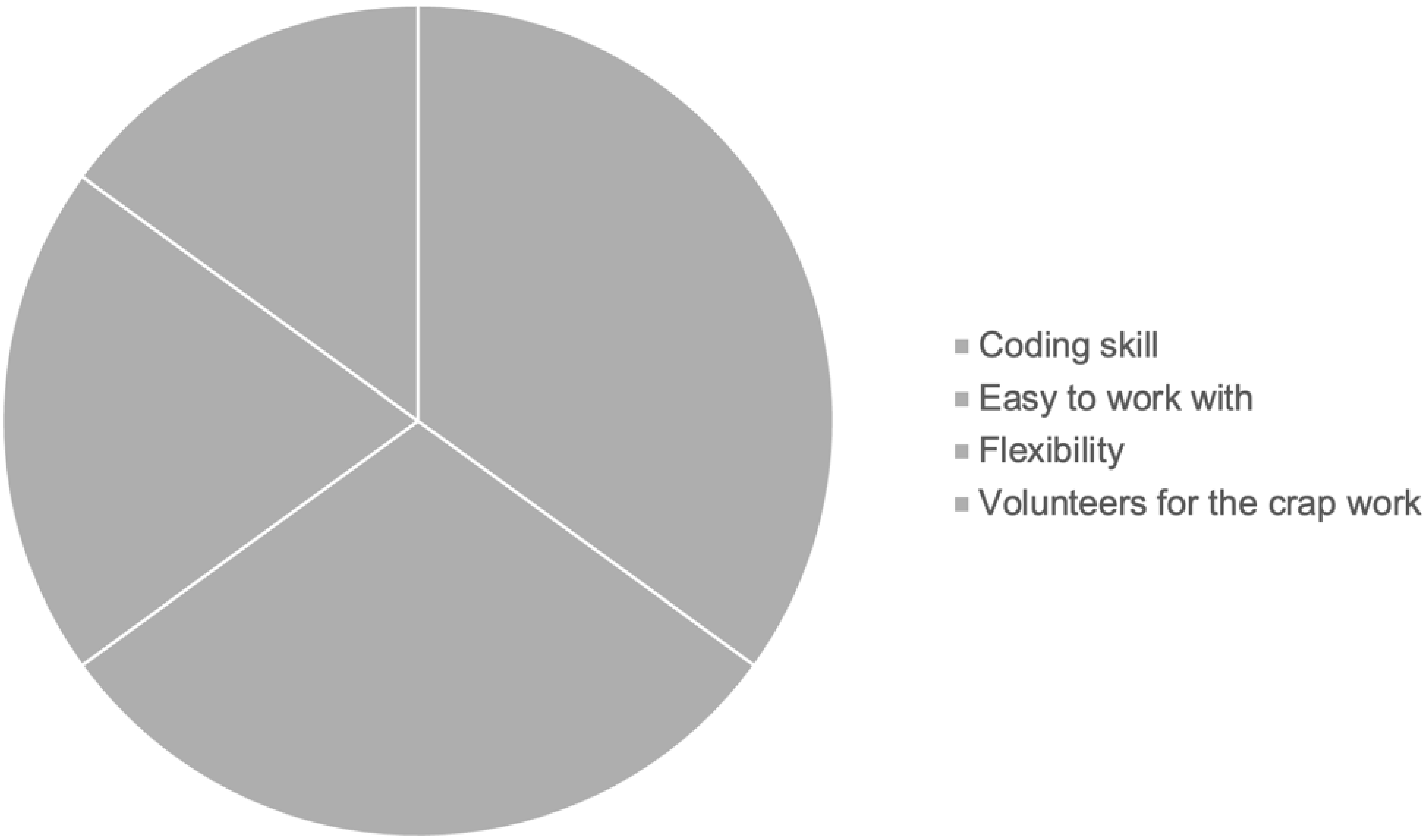
What are we going to talk about?

What are we going to talk about?

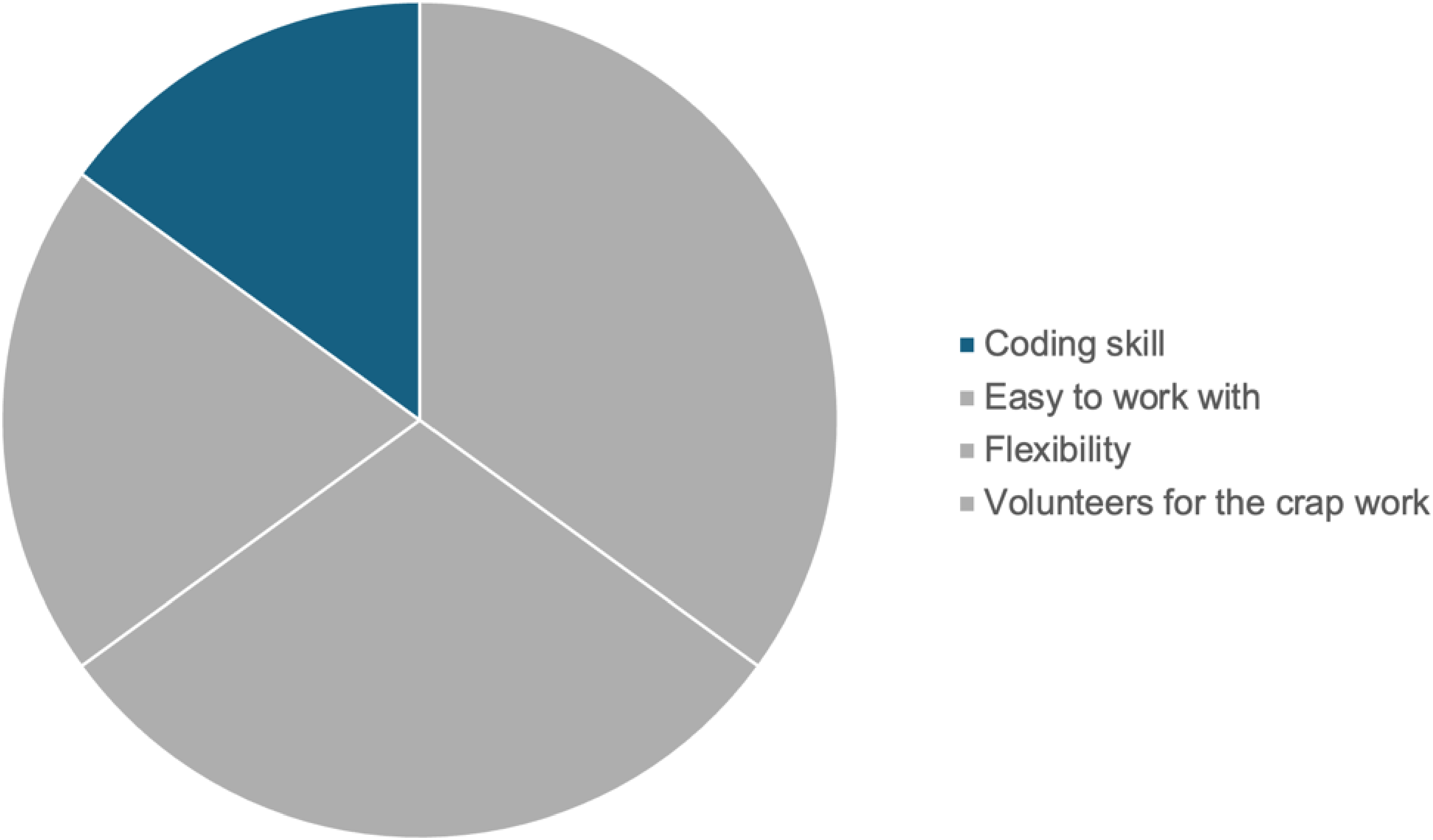
- What makes a good software engineer
- Other people's code sucks (and what do about it)
- Use git! Even on solo projects!
- How to review code
- Why tests are important, actually
- How to make your Software Hut experience more smooth

What makes a good software engineer?

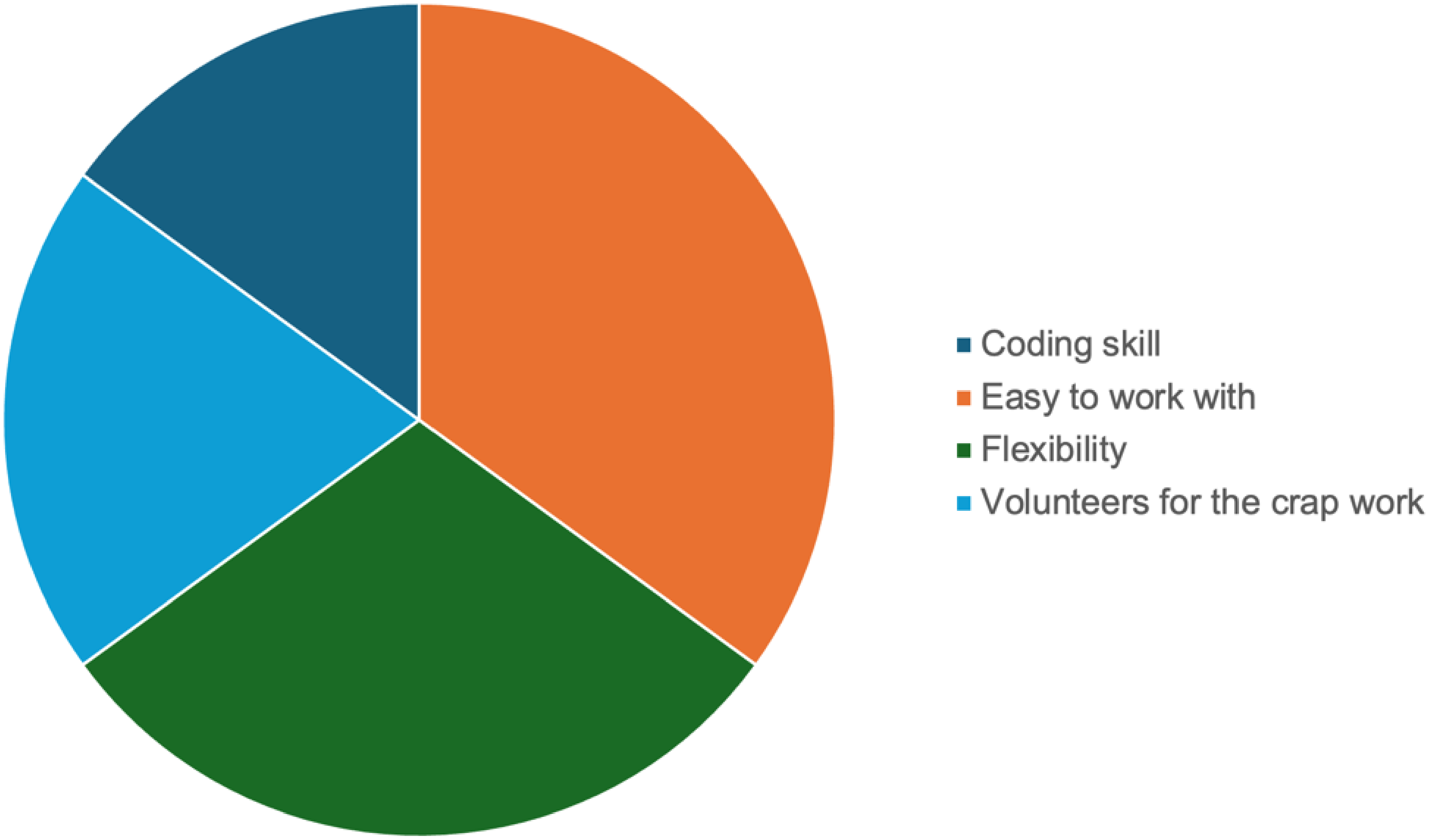
What makes a good software engineer?



What makes a good software engineer?



What makes a good software engineer?



Audience Participation

Question 1/3

Hands up if you’ve written code

(This isn’t a trick question)

Question 2/3

Hands up if you’ve worked with other people’s code

Question 3/3

Hands up if working on other people's code absolutely sucked

Why other people’s code is awful

- It’s too terse
- It’s too verbose
- It uses too many abstractions
- It doesn’t use enough abstractions
- The author didn’t write enough tests
- The author wrote way too many tests
- …

Why other people’s code is awful

- I don’t understand it

- I think the developer made poor decisions

Code is very subjective

Writing code is like writing a group essay

You **don’t** need to be identical

You **do** need to be similar

Every choice is a trade-off

Language A vs. Language B

Abstractions for flexibility vs. Overly loose code

The list is genuinely endless

Most of the time, it honestly doesn’t matter

Being *Most Correct™* doesn't matter!

- You spend far more time reading code than writing code

- Optimise for developer experience
  - CPU is cheap
  - RAM is cheap
  - Your time is not cheap

- Following existing rules > doing things *Better But Different*

A quick word on writing ***quality*** code and tests

(They are not the focus of this talk)

The only rule I recommend to people

1. Make It Work

2. Make It Right

3. Make It Fast

Attributed to Kent Beck, part of the Unix Way

I don’t have enough time to dig into detail today

There’s a wealth of information online

Why write code that *Good Enough* ?

- Code doesn’t exist in a vacuum

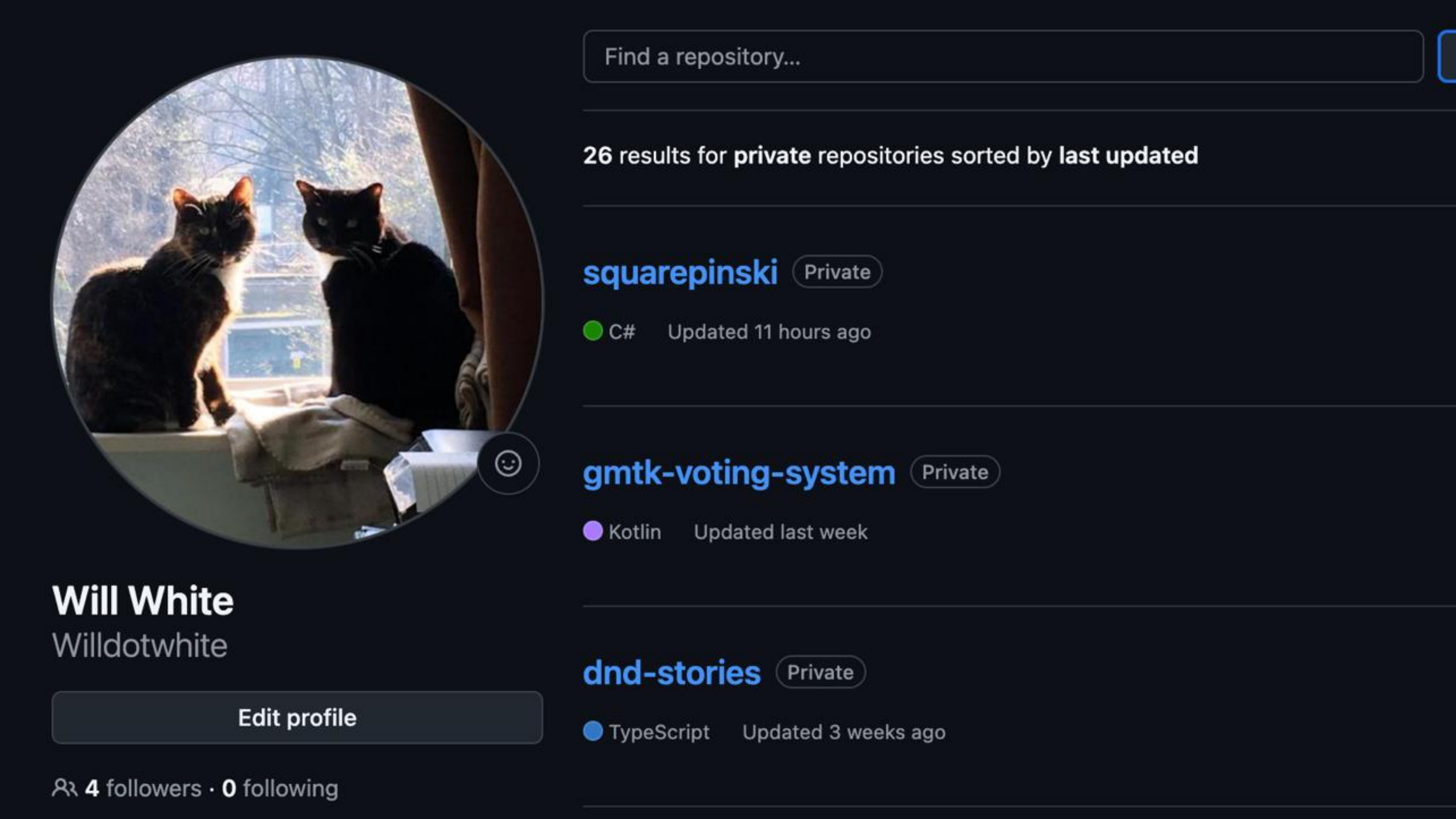
- Your code serves a business goal

- Balance “shovel out garbage” and
  “spend weeks writing a single module”

Git and friends

Doesn’t it take loads of time?

- Not as much as it saves!
- It takes seconds to commit a change
- It can (and will) save you days of effort when things go wrong
- Past You and Future You are totally different people!



Find a repository...

26 results for **private** repositories sorted by **last updated**

squarepinski Private

● C# Updated 11 hours ago

gmtk-voting-system Private

● Kotlin Updated last week

dnd-stories Private

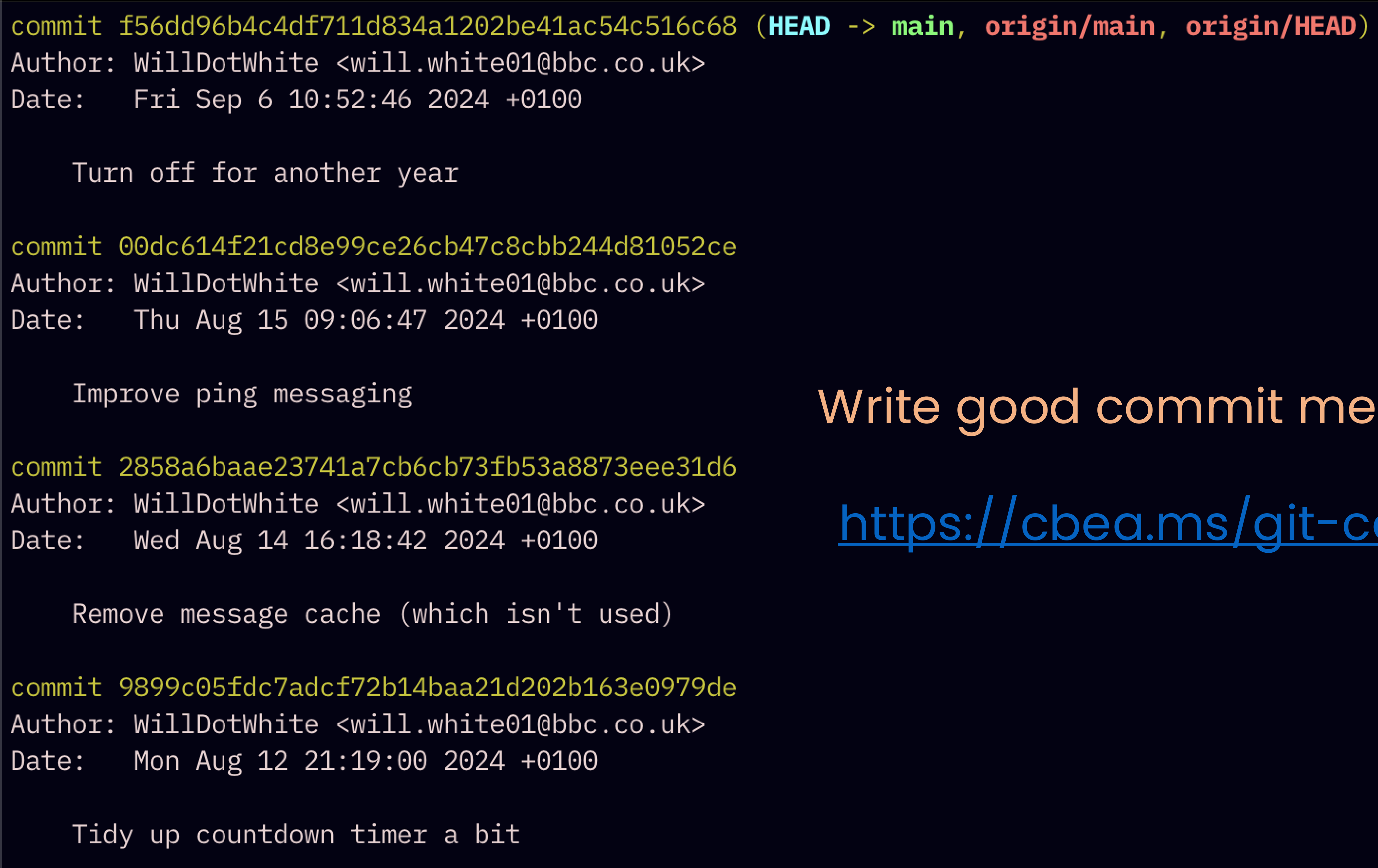
● TypeScript Updated 3 weeks ago

Will White

Willdotwhite

Edit profile

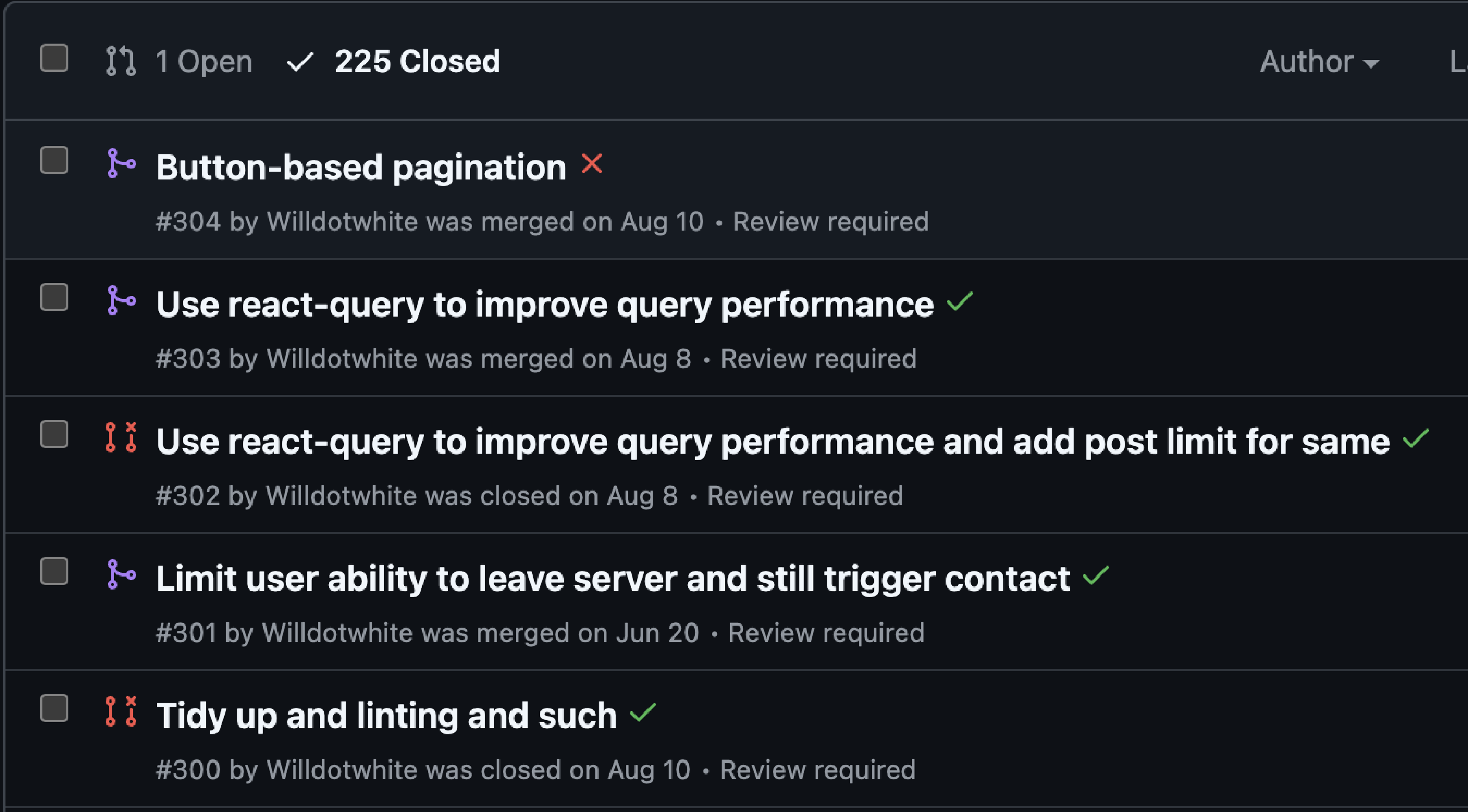
4 followers · **0** following



```
commit f56dd96b4c4df711d834a1202be41ac54c516c68 (HEAD -> main, origin/main, origin/HEAD)
Author: WillDotWhite <will.white01@bbc.co.uk>
Date:   Fri Sep 6 10:52:46 2024 +0100

    Turn off for another year

commit 00dc614f21cd8e99ce26cb47c8cbb244d81052ce
Author: WillDotWhite <will.white01@bbc.co.uk>
Date:   Thu Aug 15 09:06:47 2024 +0100

    Improve ping messaging

commit 2858a6baae23741a7cb6cb73fb53a8873eee31d6
Author: WillDotWhite <will.white01@bbc.co.uk>
Date:   Wed Aug 14 16:18:42 2024 +0100

    Remove message cache (which isn't used)

commit 9899c05fdc7adcf72b14baa21d202b163e0979de
Author: WillDotWhite <will.white01@bbc.co.uk>
Date:   Mon Aug 12 21:19:00 2024 +0100

    Tidy up countdown timer a bit
```

Write good commit messages!

https://cbea.ms/git-commit/



□ 1 Open ✓ 225 Closed Author ▾ L

□ **Button-based pagination** ✗

#304 by Willdotwhite was merged on Aug 10 • Review required

□ **Use react-query to improve query performance** ✓

#303 by Willdotwhite was merged on Aug 8 • Review required

□ **Use react-query to improve query performance and add post limit for same** ✓

#302 by Willdotwhite was closed on Aug 8 • Review required

□ **Limit user ability to leave server and still trigger contact** ✓

#301 by Willdotwhite was merged on Jun 20 • Review required

□ **Tidy up and linting and such** ✓

#300 by Willdotwhite was closed on Aug 10 • Review required

Reviewing Code

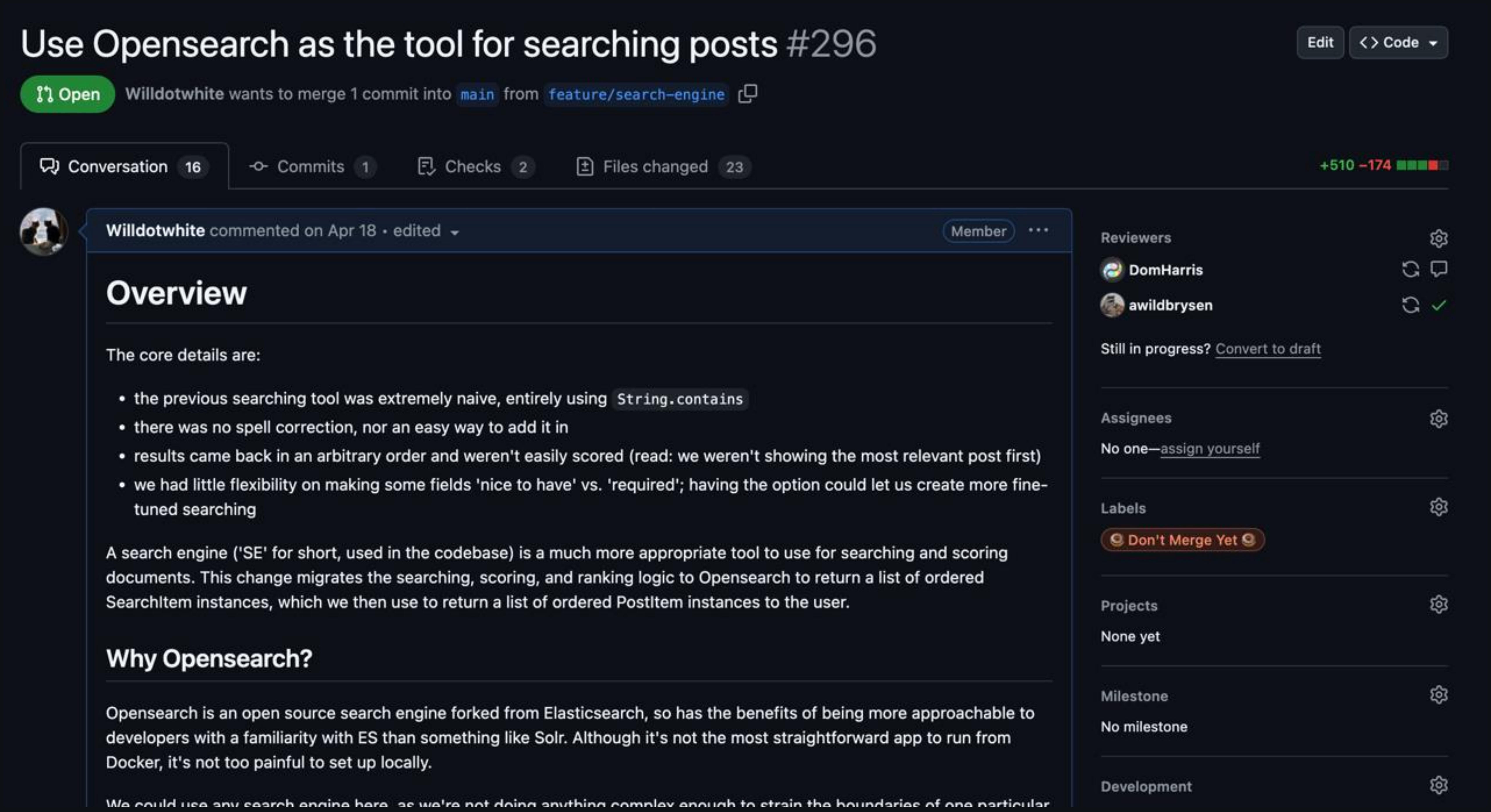
How to review code

- Expect:
  - to share your knowledge
  - to learn from others
  - for it to be uncomfortable at first

- You are not your code

- Reviews are never “reviewers vs. authors”
  - They’re “reviewers + authors vs. software is hard to build”

But seriously,
how do you review code?



Use Opensearch as the tool for searching posts #296

Open Willdotwhite wants to merge 1 commit into `main` from `feature/search-engine`

Willdotwhite commented on Apr 18 • edited

Overview

The core details are:

- the previous searching tool was extremely naive, entirely using `String.contains`
- there was no spell correction, nor an easy way to add it in
- results came back in an arbitrary order and weren't easily scored (read: we weren't showing the most relevant post first)
- we had little flexibility on making some fields 'nice to have' vs. 'required'; having the option could let us create more fine-tuned searching

A search engine ('SE' for short, used in the codebase) is a much more appropriate tool to use for searching and scoring documents. This change migrates the searching, scoring, and ranking logic to Opensearch to return a list of ordered SearchItem instances, which we then use to return a list of ordered PostItem instances to the user.

Why Opensearch?

Opensearch is an open source search engine forked from Elasticsearch, so has the benefits of being more approachable to developers with a familiarity with ES than something like Solr. Although it's not the most straightforward app to run from Docker, it's not too painful to set up locally.

We could use any search engine here, as we're not doing anything complex enough to strain the boundaries of one particular

Reviewers

DomHarris

awildbrysen ✓

Still in progress? Convert to draft

Assignees

No one—assign yourself

Labels

Don't Merge Yet

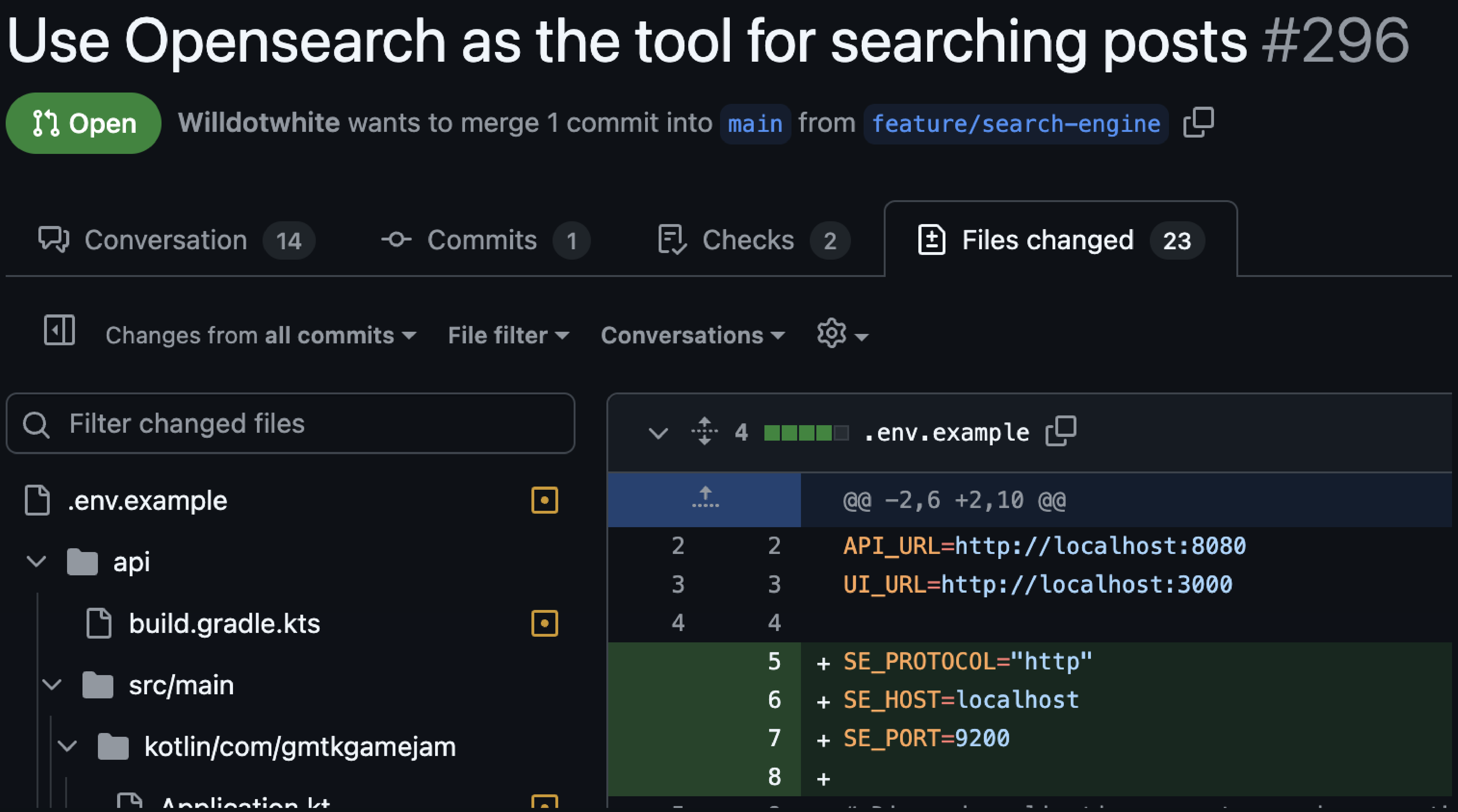
Projects

None yet

Milestone

No milestone

Development



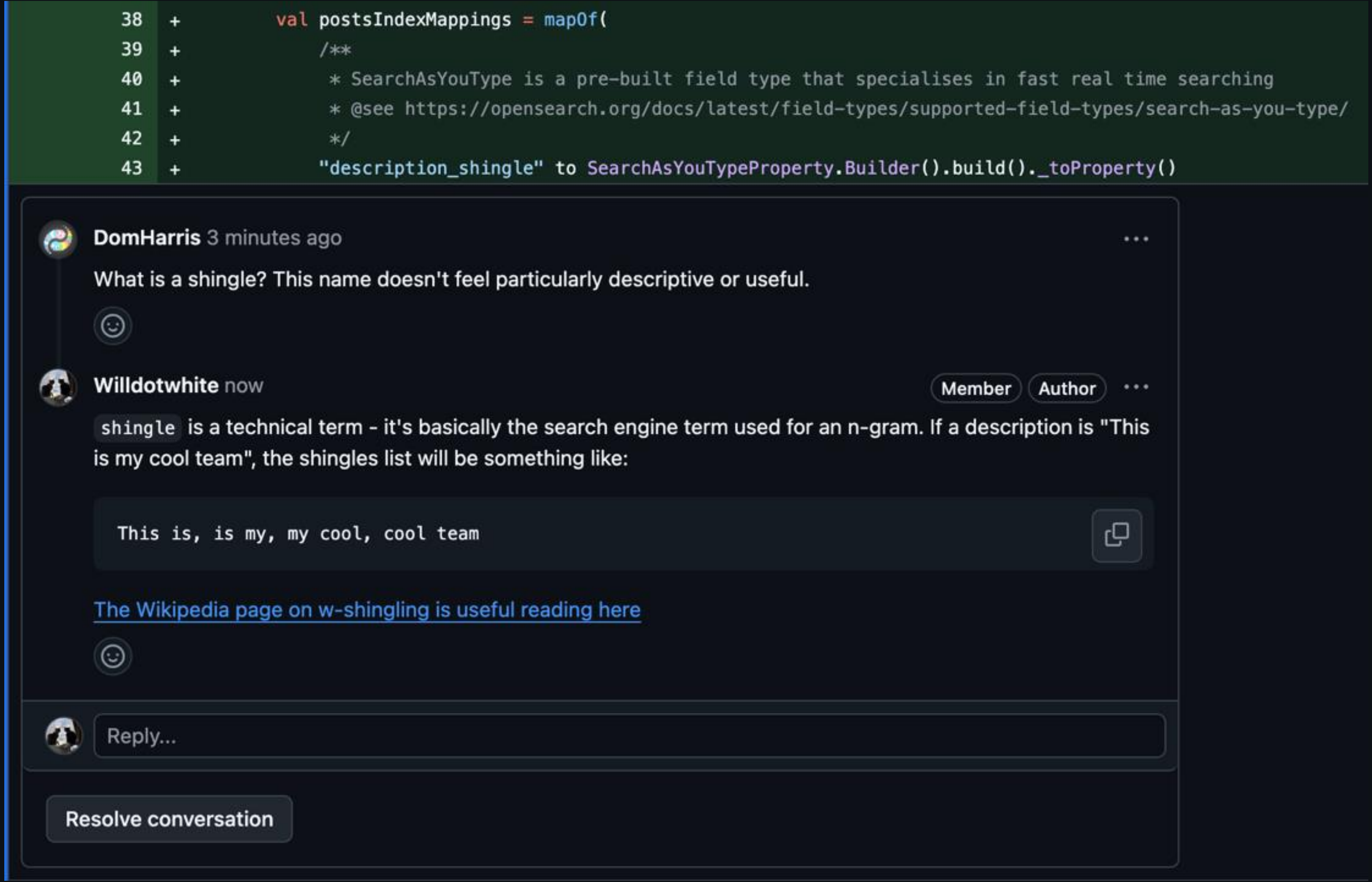
Use Opensearch as the tool for searching posts #296

Open Willdotwhite wants to merge 1 commit into `main` from `feature/search-engine`

- .env.example
- api
  - build.gradle.kts
  - src/main
    - kotlin/com/gmtkgamejam
      - Application.kt

4 .env.example

```
@@ -2,6 +2,10 @@
 API_URL=http://localhost:8080
 UI_URL=http://localhost:3000

+SE_PROTOCOL="http"
+SE_HOST=localhost
+SE_PORT=9200
+
```



```
+        val postsIndexMappings = mapOf(
+            /**
+             * SearchAsYouType is a pre-built field type that specialises in fast real time searching
+             * @see https://opensearch.org/docs/latest/field-types/supported-field-types/search-as-you-type/
+             */
+            "description_shingle" to SearchAsYouTypeProperty.Builder().build()._toProperty()
```

DomHarris 3 minutes ago

What is a shingle? This name doesn't feel particularly descriptive or useful.

Willdotwhite now

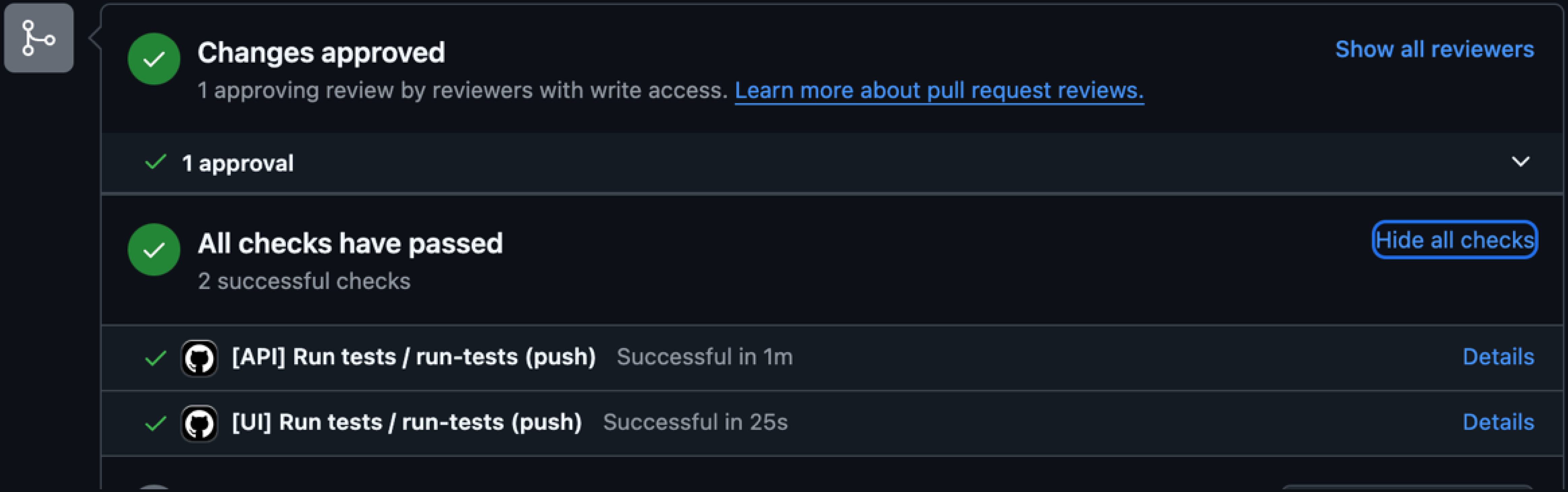
`shingle` is a technical term - it's basically the search engine term used for an n-gram. If a description is "This is my cool team", the shingles list will be something like:

```
This is, is my, my cool, cool team
```

The Wikipedia page on w-shingling is useful reading here

How tests get used

(This used to be the CI/CD section,
but I had to cut it because of the time constraint)



✓ **Changes approved**

1 approving review by reviewers with write access. Learn more about pull request reviews.

Show all reviewers

✓ 1 approval

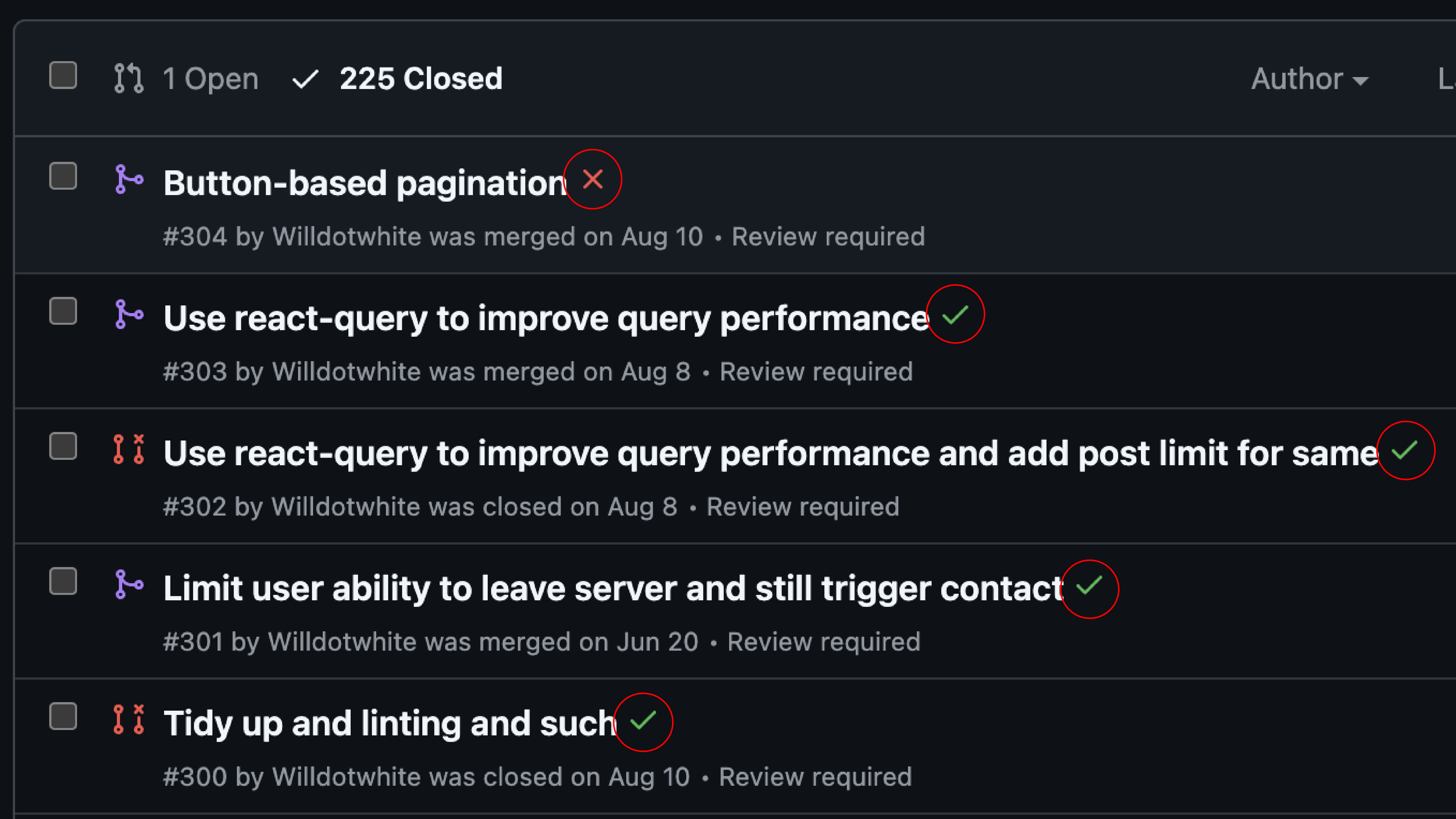
✓ **All checks have passed**

2 successful checks

Hide all checks

✓ **[API] Run tests / run-tests (push)** Successful in 1m Details

✓ **[UI] Run tests / run-tests (push)** Successful in 25s Details



□ 1 Open ✓ 225 Closed Author ▾ L

□ **Button-based pagination** ✗

#304 by Willdotwhite was merged on Aug 10 • Review required

□ **Use react-query to improve query performance** ✓

#303 by Willdotwhite was merged on Aug 8 • Review required

□ **Use react-query to improve query performance and add post limit for same** ✓

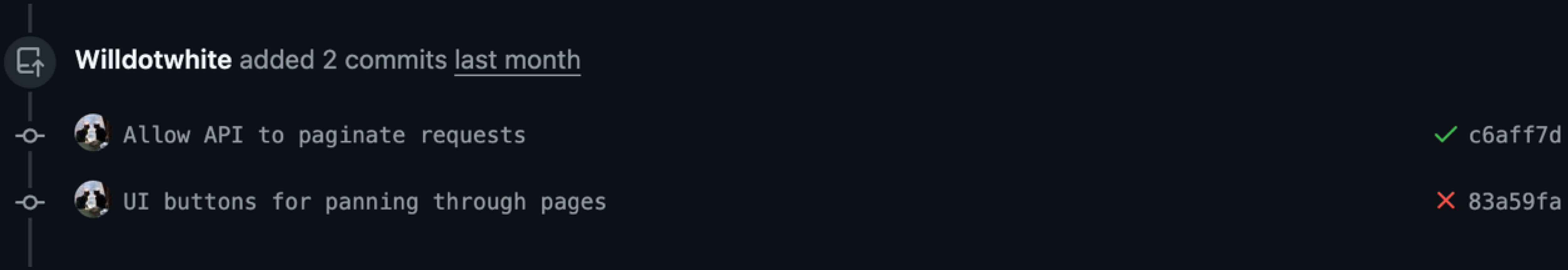
#302 by Willdotwhite was closed on Aug 8 • Review required

□ **Limit user ability to leave server and still trigger contact** ✓

#301 by Willdotwhite was merged on Jun 20 • Review required

□ **Tidy up and linting and such** ✓

#300 by Willdotwhite was closed on Aug 10 • Review required

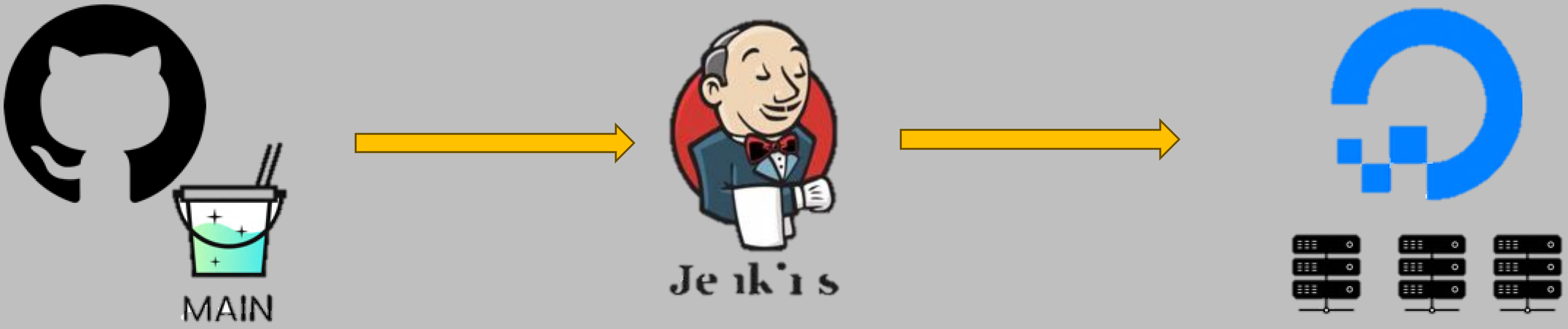


Willdotwhite added 2 commits last month

Allow API to paginate requests ✓ c6aff7d

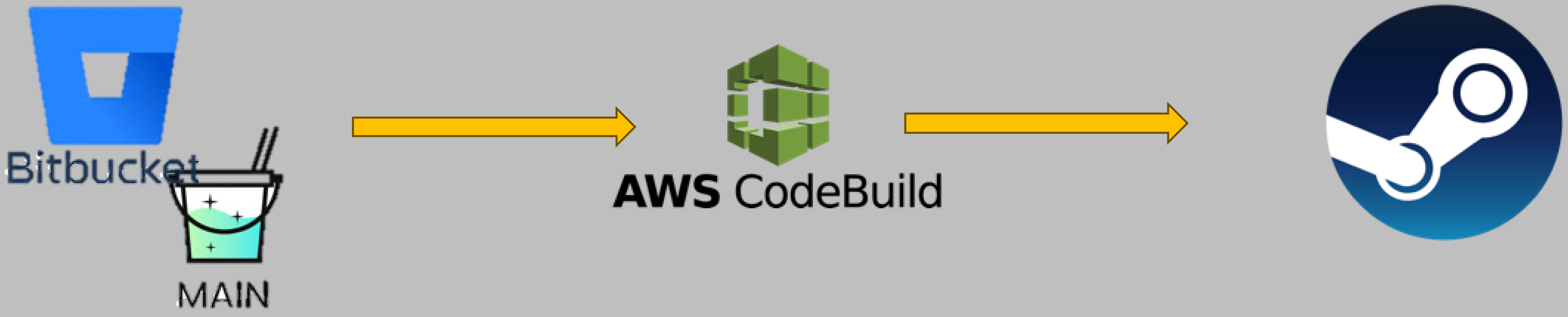
UI buttons for panning through pages ✗ 83a59fa

… and when it’s merged?



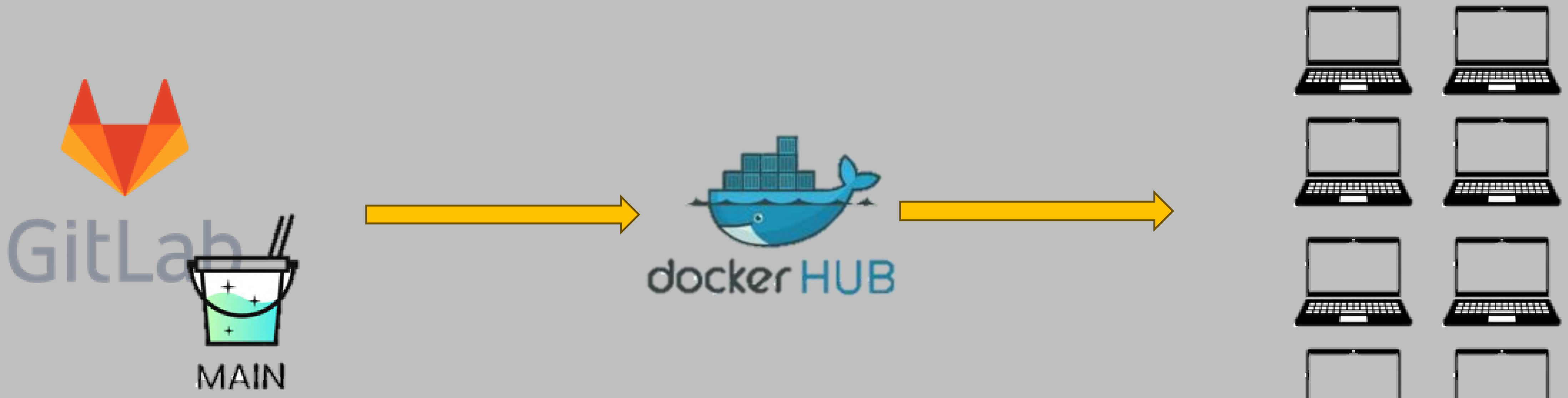
Option 1/many

… and when it’s merged?



Option 2/many

… and when it's merged?

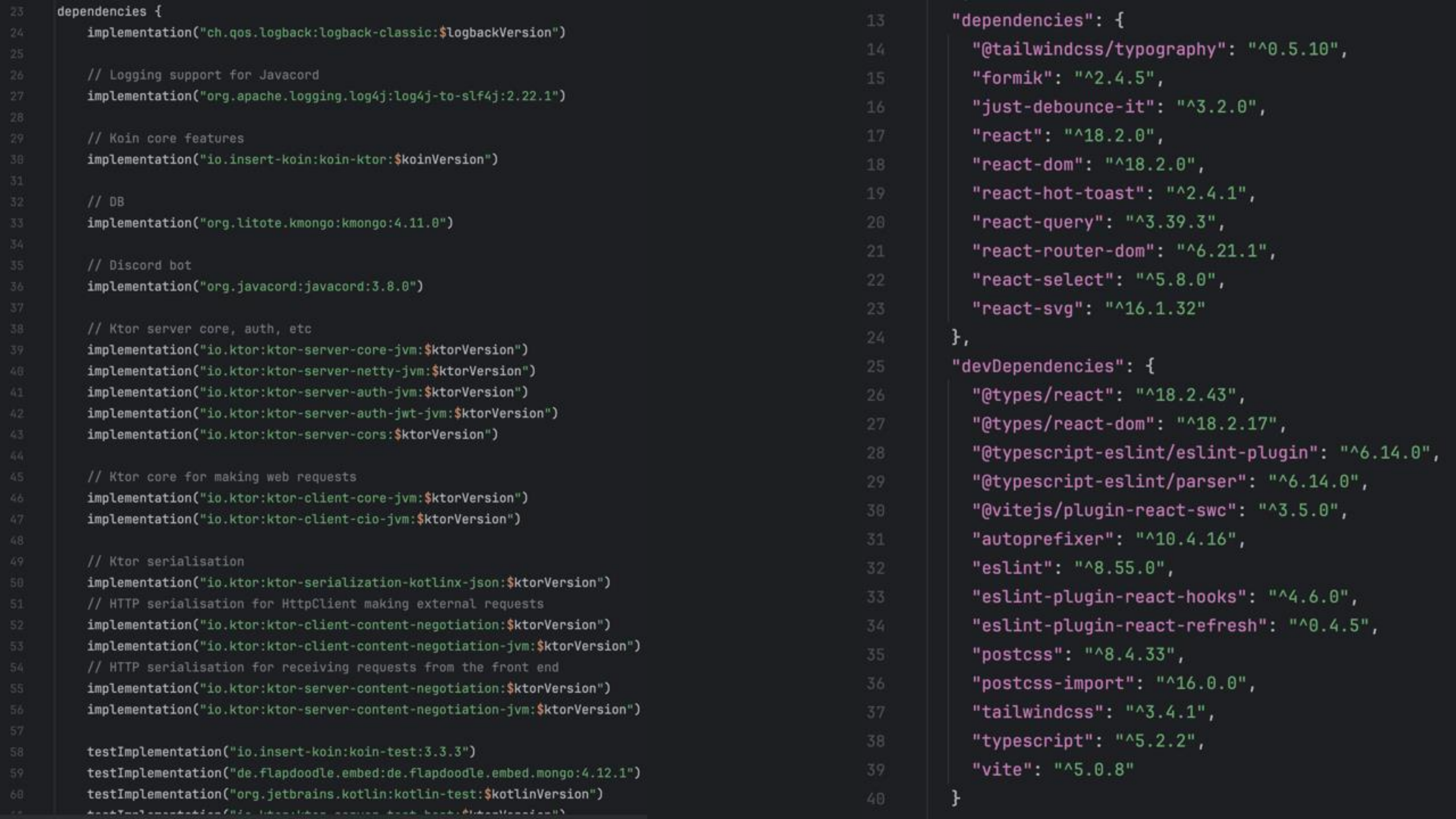


Option 3/many

A quick note on libraries and frameworks

Why use libraries and frameworks?

- Don’t reinvent the wheel!
- Use other people’s code
- Save your time, focus on your work
- Their code is better than yours
  - (because they’ve spent longer getting it wrong)



```kotlin
dependencies {
    implementation("ch.qos.logback:logback-classic:$logbackVersion")

    // Logging support for Javacord
    implementation("org.apache.logging.log4j:log4j-to-slf4j:2.22.1")

    // Koin core features
    implementation("io.insert-koin:koin-ktor:$koinVersion")

    // DB
    implementation("org.litote.kmongo:kmongo:4.11.0")

    // Discord bot
    implementation("org.javacord:javacord:3.8.0")

    // Ktor server core, auth, etc
    implementation("io.ktor:ktor-server-core-jvm:$ktorVersion")
    implementation("io.ktor:ktor-server-netty-jvm:$ktorVersion")
    implementation("io.ktor:ktor-server-auth-jvm:$ktorVersion")
    implementation("io.ktor:ktor-server-auth-jwt-jvm:$ktorVersion")
    implementation("io.ktor:ktor-server-cors:$ktorVersion")

    // Ktor core for making web requests
    implementation("io.ktor:ktor-client-core-jvm:$ktorVersion")
    implementation("io.ktor:ktor-client-cio-jvm:$ktorVersion")

    // Ktor serialisation
    implementation("io.ktor:ktor-serialization-kotlinx-json:$ktorVersion")
    // HTTP serialisation for HttpClient making external requests
    implementation("io.ktor:ktor-client-content-negotiation:$ktorVersion")
    implementation("io.ktor:ktor-client-content-negotiation-jvm:$ktorVersion")
    // HTTP serialisation for receiving requests from the front end
    implementation("io.ktor:ktor-server-content-negotiation:$ktorVersion")
    implementation("io.ktor:ktor-server-content-negotiation-jvm:$ktorVersion")

    testImplementation("io.insert-koin:koin-test:3.3.3")
    testImplementation("de.flapdoodle.embed:de.flapdoodle.embed.mongo:4.12.1")
    testImplementation("org.jetbrains.kotlin:kotlin-test:$kotlinVersion")
    testImplementation("io.ktor:ktor-server-test-host:$ktorVersion")
```

```json
  "dependencies": {
    "@tailwindcss/typography": "^0.5.10",
    "formik": "^2.4.5",
    "just-debounce-it": "^3.2.0",
    "react": "^18.2.0",
    "react-dom": "^18.2.0",
    "react-hot-toast": "^2.4.1",
    "react-query": "^3.39.3",
    "react-router-dom": "^6.21.1",
    "react-select": "^5.8.0",
    "react-svg": "^16.1.32"
  },
  "devDependencies": {
    "@types/react": "^18.2.43",
    "@types/react-dom": "^18.2.17",
    "@typescript-eslint/eslint-plugin": "^6.14.0",
    "@typescript-eslint/parser": "^6.14.0",
    "@vitejs/plugin-react-swc": "^3.5.0",
    "autoprefixer": "^10.4.16",
    "eslint": "^8.55.0",
    "eslint-plugin-react-hooks": "^4.6.0",
    "eslint-plugin-react-refresh": "^0.4.5",
    "postcss": "^8.4.33",
    "postcss-import": "^16.0.0",
    "tailwindcss": "^3.4.1",
    "typescript": "^5.2.2",
    "vite": "^5.0.8"
  }
```

Some final quick-fire thoughts

How can you make your Software Hut experience not terrible?

Be pragmatic about what you contribute

Be moderate in what you expect from others

You might want to discuss…

You might want to discuss…

- Will you use version control?
  - **YES, USE VERSION CONTROL!**
  - I recommend **git** and **trunk-based development**
    - This is the most common practice for version control
  - Write good commit messages: https://cbea.ms/git-commit/

You might want to discuss…

- Will you use version control? (Yes)
- What happens if you can’t agree on something?
  - Do you vote? Toss a coin? Ask an unbiased peer?
  - Look up “Disagree and commit”
  - It’s impossible to do this *after* the big argument

You might want to discuss…

- Will you use version control? (Yes)
- What happens if you can’t agree on something?
- How will you coordinate your work?
  - JIRA tickets? Group DM chat?
  - Will you pair program?
  - How/when will you review pull requests?

Stretch goals

- Consider using:
  - Pre-commit hooks
  - PR checks to run tests
  - Code linting tools
  - CI/CD

- Check out open source projects!
  - They’re a great place to learn

Thank you for your time!

These slides will be available on **badlydone.dev/presentations**

Any questions?