

# **LIBRARIAN'S GUIDE**

*to documentation*

**Kaitlyn Tierney**  
[krtierney.com](http://krtierney.com)

# **Software engineer characteristics**

# Software engineer characteristics

- Like board games



# Software engineer characteristics

- Like board games
- **Learn new things**

# Software engineer characteristics

- Like board games
- Learn new things
- **Look things up**

# Software engineer characteristics

- Like board games
- Learn new things
- Look things up
- **Use computers**

# Software engineer characteristics

- Like board games
- Learn new things
- Look things up
- Use computers
- **Distill vague requests into actionable queries**

# Software engineer characteristics

- Like board games
- Learn new things
- Look things up
- Use computers
- Distill vague requests into actionable queries
- **Love booleans and databases**



# Software engineer characteristics

- Like board games
- Learn new things
- Look things up
- Use computers
- Distill vague requests into actionable queries
- Love booleans and databases
- **Good at documentation**

**Let's talk about your  
documentation problem**

**It's hard**

**It takes time**

**It's hard to maintain**

**We don't see the value**

**Too much of a good thing**

**Can't find what you need**





# Discoverability



# ~~Software engineer~~ Librarian characteristics

- Like board games
- Learn new things
- Look things up
- Use computers
- Distill vague requests into actionable queries
- Love booleans and databases
- Good at documentation

# ~~Software engineer~~ Librarian characteristics

- Like board games
- Learn new things
- Look things up
- Use computers
- Distill vague requests into actionable queries
- Love booleans and databases
- Good at documentation
- **Master's degree in library science**





















# Engineering Wiki

↩ 2 backlinks

All things engineering: processes, best practices, setup guides, and more!





## Guides & Processes

---

-  [Using the Wiki](#)
-  [Onboarding](#)
-  [Incident Response](#)
-  [Data security](#)






## Codebase

---

-  [Development Lifecycle](#)
-  [Engineering Guidelines](#)
-  [Applications](#)
-  [Services](#)





## Testing

---

-  [Automated Testing](#)
-  [Browser support](#)
-  [TODO: How to QA](#)
-  [Penetration testing](#)
-  [A/B testing](#)


## People

---

-  [Who's Who](#)
-  [Progression & Promotions](#)
-  [Tech Leading at Farewill](#)
-  [Company Rituals](#)






## Collaboration

---

-  [Engineering on Slack](#)
-  [Proposals and ADRs](#)
-  [Skills Matrix](#)

## Infrastructure

---

-  [Continuous Integration](#)
-  [Domain and DNS setup](#)
-  [Heroku](#)
-  [AWS](#)
-  [Data storage](#)

# Librarian tips

**0. Take a backup**

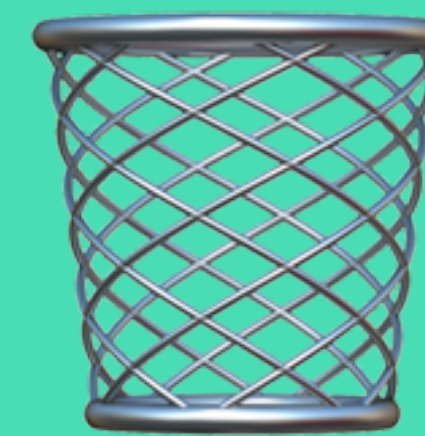
**1. Start sorting**



wiki



archive



bin



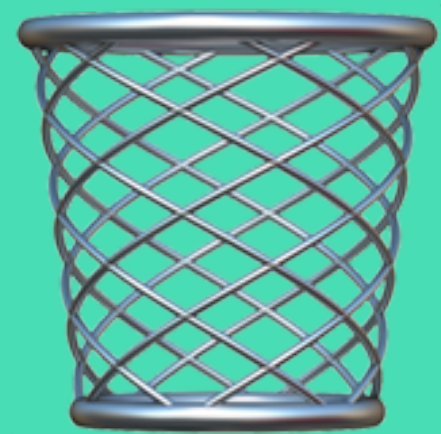




wiki



archive



bin

# Digital ephemera

# Working docs

▼ Meeting notes archive

Previous meetings ▾

### Engineering Guild meetings

Aa -	Person	Recording Link
@April 25, 2023	[Redacted]	<a href="https://drive.google.com/file/d/1">https://drive.google.com/file/d/1</a>
@April 11, 2023	[Redacted]	
@April 4, 2023	[Redacted]	<a href="https://drive.google.com/file/d/1">https://drive.google.com/file/d/1</a>
@March 21, 2023	[Redacted]	<a href="https://drive.google.com/file/d/1">https://drive.google.com/file/d/1</a>
@March 14, 2023	[Redacted]	
@March 7, 2023	[Redacted]	<a href="https://drive.google.com/file/d/1">https://drive.google.com/file/d/1</a>
@February 28, 2023	[Redacted]	<a href="https://drive.google.com/file/d/1">https://drive.google.com/file/d/1</a>
🍌 @February 21, 2023	[Redacted]	<a href="https://drive.google.com/file/d/1">https://drive.google.com/file/d/1</a>
🍷 @February 14, 2023	[Redacted]	<a href="https://drive.google.com/file/d/1">https://drive.google.com/file/d/1</a>
7 February 2023	[Redacted]	<a href="https://drive.google.com/file/d/1">https://drive.google.com/file/d/1</a>
31st January 2023	[Redacted]	<a href="https://drive.google.com/open?i">https://drive.google.com/open?i</a>
24th January 2023	[Redacted]	<a href="https://drive.google.com/file/d/1">https://drive.google.com/file/d/1</a>
@January 17, 2023	[Redacted]	<a href="https://drive.google.com/file/d/1">https://drive.google.com/file/d/1</a>
@January 10, 2023	[Redacted]	<a href="https://drive.google.com/file/d/1">https://drive.google.com/file/d/1</a>
@January 3, 2023	[Redacted]	NA - meeting cancelled
@December 20, 2022	[Redacted]	<a href="https://drive.google.com/file/d/1">https://drive.google.com/file/d/1</a>
Le 13 décembre	[Redacted]	<a href="https://drive.google.com/file/d/1">https://drive.google.com/file/d/1</a>
29th November 2022	[Redacted]	<a href="https://drive.google.com/file/d/1">https://drive.google.com/file/d/1</a>
@November 22, 2022	[Redacted]	<a href="https://drive.google.com/file/d/1">https://drive.google.com/file/d/1</a>
@November 15, 2022	[Redacted]	<a href="https://drive.google.com/file/d/1">https://drive.google.com/file/d/1</a>
@November 8, 2022	[Redacted]	<a href="https://drive.google.com/file/d/1">https://drive.google.com/file/d/1</a>
@November 1, 2022	[Redacted]	<a href="https://drive.google.com/file/d/1">https://drive.google.com/file/d/1</a>

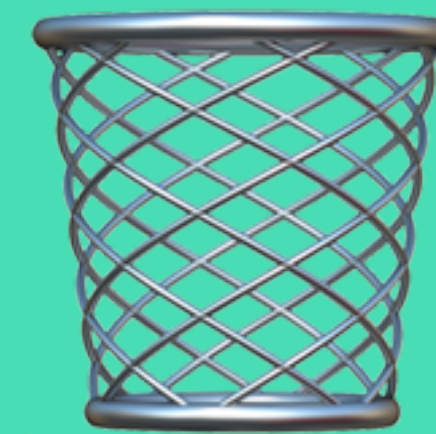
COUNT 80



wiki



archive



bin

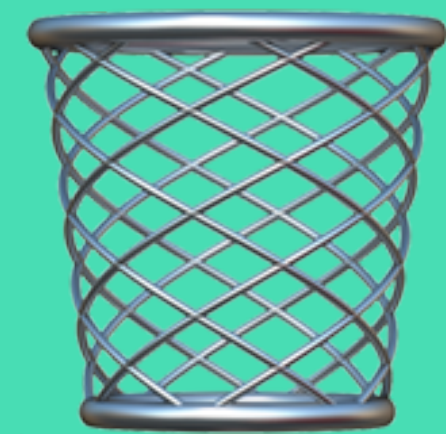




wiki



archive


























bin

## 2. Create your taxonomy



# Engineering Wiki <Example>

↩ 2 backlinks

-  [Onboarding](#)
-  [Who's Who](#)
-  [Progression & Promotions](#)
-  [Tech Leading at Farewill](#)
-  [Company Rituals](#)
-  [Engineering on Slack](#)
-  [Proposals and ADRs](#)
-  [Skills Matrix](#)
-  [Continuous Integration](#)
-  [Domain and DNS setup](#)
-  [Heroku](#)
-  [AWS](#)
-  [Data storage](#)
-  [Incident Response](#)
-  [Data security](#)
-  [Development Lifecycle](#)
-  [Engineering Guidelines](#)
-  [Applications](#)
-  [Services](#)
-  [Automated Testing](#)
-  [Browser support](#)
-  [TODO: How to QA](#)
-  [Penetration testing](#)







# Engineering Wiki

↩ 2 backlinks

All things engineering: processes, best practices, setup guides, and more!





## Guides & Processes

---

-  [Using the Wiki](#)
-  [Onboarding](#)
-  [Incident Response](#)
-  [Data security](#)






## Codebase

---

-  [Development Lifecycle](#)
-  [Engineering Guidelines](#)
-  [Applications](#)
-  [Services](#)





## Testing

---

-  [Automated Testing](#)
-  [Browser support](#)
-  [TODO: How to QA](#)
-  [Penetration testing](#)
-  [A/B testing](#)




## People

---

-  [Who's Who](#)
-  [Progression & Promotions](#)
-  [Tech Leading at Farewill](#)
-  [Company Rituals](#)






## Collaboration

---

-  [Engineering on Slack](#)
-  [Proposals and ADRs](#)
-  [Skills Matrix](#)

## Infrastructure





---

-  [Continuous Integration](#)
-  [Domain and DNS setup](#)
-  [Heroku](#)
-  [AWS](#)
-  [Data storage](#)

**Look for patterns**





## Guides & Processes

---

-  [Using the Wiki](#)
-  [Onboarding](#)
-  [Incident Response](#)
-  [Data security](#)






## Codebase

---

-  [Development Lifecycle](#)
-  [Engineering Guidelines](#)
-  [Applications](#)
-  [Services](#)





## Testing

---

-  [Automated Testing](#)
-  [Browser support](#)
-  [TODO: How to QA](#)
-  [Penetration testing](#)
-  [A/B testing](#)




## People

---

-  [Who's Who](#)
-  [Progression & Promotions](#)
-  [Tech Leading at Farewill](#)
-  [Company Rituals](#)






## Collaboration

---

-  [Engineering on Slack](#)
-  [Proposals and ADRs](#)
-  [Skills Matrix](#)

## Infrastructure





---

-  [Continuous Integration](#)
-  [Domain and DNS setup](#)
-  [Heroku](#)
-  [AWS](#)
-  [Data storage](#)

**Content follows form**





## Guides & Processes

---

-  [Using the Wiki](#)
-  [Onboarding](#)
-  [Incident Response](#)
-  [Data security](#)






## Codebase

---

-  [Development Lifecycle](#)
-  [Engineering Guidelines](#)
-  [Applications](#)
-  [Services](#)





## Testing

---

-  [Automated Testing](#)
-  [Browser support](#)
-  [TODO: How to QA](#)
-  [Penetration testing](#)
-  [A/B testing](#)



## People

---

-  [Who's Who](#)
-  [Progression & Promotions](#)
-  [Tech Leading at Farewill](#)
-  [Company Rituals](#)






## Collaboration

---

-  [Engineering on Slack](#)
-  [Proposals and ADRs](#)
-  [Skills Matrix](#)





## Infrastructure

---

-  [Continuous Integration](#)
-  [Domain and DNS setup](#)
-  [Heroku](#)
-  [AWS](#)
-  [Data storage](#)





## Guides & Processes

---

-  [Using the Wiki](#)
-  [Onboarding](#)
-  [Incident Response](#)
-  [Data security](#)






## Codebase

---

-  [Development Lifecycle](#)
-  [Engineering Guidelines](#)
-  [Applications](#)
-  [Services](#)





## Testing

---

-  [Automated Testing](#)
-  [Browser support](#)
-  [TODO: How to QA](#)
-  [Penetration testing](#)
-  [A/B testing](#)




## People

---

-  [Who's Who](#)
-  [Progression & Promotions](#)
-  [Tech Leading at Farewill](#)
-  [Company Rituals](#)






## Collaboration

---

-  [Engineering on Slack](#)
-  [Proposals and ADRs](#)
-  [Skills Matrix](#)





## Infrastructure

---

-  [Continuous Integration](#)
-  [Domain and DNS setup](#)
-  [Heroku](#)
-  [AWS](#)
-  [Data storage](#)





## Guides & Processes

---

-  [Using the Wiki](#)
-  [Onboarding](#)
-  [Incident Response](#)
-  [Data security](#)






## Codebase

---

-  [Development Lifecycle](#)
-  [Engineering Guidelines](#)
-  [Applications](#)
-  [Services](#)





## Testing

---

-  [Automated Testing](#)
-  [Browser support](#)
-  [TODO: How to QA](#)
-  [Penetration testing](#)
-  [A/B testing](#)



## People

---

-  [Who's Who](#)
-  [Progression & Promotions](#)
-  [Tech Leading at Farewill](#)
-  [Company Rituals](#)






## Collaboration

---

-  [Engineering on Slack](#)
-  [Proposals and ADRs](#)
-  [Skills Matrix](#)

## Infrastructure





---

-  [Continuous Integration](#)
-  [Domain and DNS setup](#)
-  [Heroku](#)
-  [AWS](#)
-  [Data storage](#)







## Guides & Processes

---

-  [Using the Wiki](#)
-  [Onboarding](#)
-  [Incident Response](#)
-  [Data security](#)






## Codebase

---

-  [Development Lifecycle](#)
-  [Engineering Guidelines](#)
-  [Applications](#)
-  [Services](#)





## Testing

---

-  [Automated Testing](#)
-  [Browser support](#)
-  [TODO: How to QA](#)
-  [Penetration testing](#)
-  [A/B testing](#)




## People

---

-  [Who's Who](#)
-  [Progression & Promotions](#)
-  [Tech Leading at Farewill](#)
-  [Company Rituals](#)






## Collaboration

---

-  [Engineering on Slack](#)
-  [Proposals and ADRs](#)
-  [Skills Matrix](#)





## Infrastructure

---

-  [Continuous Integration](#)
-  [Domain and DNS setup](#)
-  [Heroku](#)
-  [AWS](#)
-  [Data storage](#)





## Guides & Processes

---

-  [Using the Wiki](#)
-  [Onboarding](#)
-  [Incident Response](#)
-  [Data security](#)






## Codebase

---

-  [Development Lifecycle](#)
-  [Engineering Guidelines](#)
-  [Applications](#)
-  [Services](#)





## Testing

---

-  [Automated Testing](#)
-  [Browser support](#)
-  [TODO: How to QA](#)
-  [Penetration testing](#)
-  [A/B testing](#)




## People

---

-  [Who's Who](#)
-  [Progression & Promotions](#)
-  [Tech Leading at Farewill](#)
-  [Company Rituals](#)






## Collaboration

---

-  [Engineering on Slack](#)
-  [Proposals and ADRs](#)
-  [Skills Matrix](#)





## Infrastructure

---

-  [Continuous Integration](#)
-  [Domain and DNS setup](#)
-  [Heroku](#)
-  [AWS](#)
-  [Data storage](#)





## Guides & Processes

---

-  [Using the Wiki](#)
-  [Onboarding](#)
-  [Incident Response](#)
-  [Data security](#)






## Codebase

---

-  [Development Lifecycle](#)
-  [Engineering Guidelines](#)
-  [Applications](#)
-  [Services](#)





## Testing

---

-  [Automated Testing](#)
-  [Browser support](#)
-  [TODO: How to QA](#)
-  [Penetration testing](#)
-  [A/B testing](#)



## People

---

-  [Who's Who](#)
-  [Progression & Promotions](#)
-  [Tech Leading at Farewill](#)
-  [Company Rituals](#)






## Collaboration

---

-  [Engineering on Slack](#)
-  [Proposals and ADRs](#)
-  [Skills Matrix](#)





## Infrastructure

---

-  [Continuous Integration](#)
-  [Domain and DNS setup](#)
-  [Heroku](#)
-  [AWS](#)
-  [Data storage](#)





## Guides & Processes

---

-  [Using the Wiki](#)
-  [Onboarding](#)
-  [Incident Response](#)
-  [Data security](#)






## Codebase

---

-  [Development Lifecycle](#)
-  [Engineering Guidelines](#)
-  [Applications](#)
-  [Services](#)





## Testing

---

-  [Automated Testing](#)
-  [Browser support](#)
-  [TODO: How to QA](#)
-  [Penetration testing](#)
-  [A/B testing](#)



## People

---

-  [Who's Who](#)
-  [Progression & Promotions](#)
-  [Tech Leading at Farewill](#)
-  [Company Rituals](#)






## Collaboration

---

-  [Engineering on Slack](#)
-  [Proposals and ADRs](#)
-  [Skills Matrix](#)





## Infrastructure

---

-  [Continuous Integration](#)
-  [Domain and DNS setup](#)
-  [Heroku](#)
-  [AWS](#)
-  [Data storage](#)





## Guides & Processes

---

-  [Using the Wiki](#)
-  [Onboarding](#)
-  [Incident Response](#)
-  [Data security](#)






## Codebase

---

-  [Development Lifecycle](#)
-  [Engineering Guidelines](#)
-  [Applications](#)
-  [Services](#)





## Testing

---

-  [Automated Testing](#)
-  [Browser support](#)
-  [TODO: How to QA](#)
-  [Penetration testing](#)
-  [A/B testing](#)



## People

---

-  [Who's Who](#)
-  [Progression & Promotions](#)
-  [Tech Leading at Farewill](#)
-  [Company Rituals](#)






## Collaboration

---

-  [Engineering on Slack](#)
-  [Proposals and ADRs](#)
-  [Skills Matrix](#)

## Infrastructure

---

-  [Continuous Integration](#)
-  [Domain and DNS setup](#)
-  [Heroku](#)
-  [AWS](#)
-  [Data storage](#)

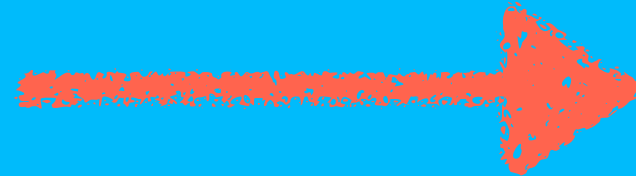
**Define topic headings**

# 3. Set your style

# Engineering Wiki <Example>

↙ 2 backlinks

- 📄 [Onboarding](#)
- 📄 [Who's Who](#)
- 📄 [Progression & Promotions](#)
- 📄 [Tech Leading at Farewill](#)
- 📄 [Company Rituals](#)
- 📄 [Engineering on Slack](#)
- 📄 [Proposals and ADRs](#)
- 📄 [Skills Matrix](#)
- 📄 [Continuous Integration](#)
- 📄 [Domain and DNS setup](#)
- 📄 [Heroku](#)
- 📄 [AWS](#)
- 📄 [Data storage](#)
- 📄 [Incident Response](#)
- 📄 [Data security](#)
- 📄 [Development Lifecycle](#)
- 📄 [Engineering Guidelines](#)
- 📄 [Applications](#)
- 📄 [Services](#)
- 📄 [Automated Testing](#)
- 📄 [Browser support](#)
- 📄 [TODO: How to QA](#)
- 📄 [Penetration testing](#)
- 📄 [A/B testing](#)



# Engineering Wiki <Example>

↙ 2 backlinks

## Guides & Processes

- 📄 [Using the Wiki](#)
- 📄 [Onboarding](#)
- 📄 [Incident Response](#)
- 📄 [Data security](#)

## Codebase

- 📄 [Development Lifecycle](#)
- 📄 [Engineering Guidelines](#)
- 📄 [Applications](#)
- 📄 [Services](#)

## Testing

- 📄 [Automated Testing](#)
- 📄 [Browser support](#)
- 📄 [TODO: How to QA](#)
- 📄 [Penetration testing](#)
- 📄 [A/B testing](#)

## People

- 📄 [Who's Who](#)
- 📄 [Progression & Promotions](#)
- 📄 [Tech Leading at Farewill](#)
- 📄 [Company Rituals](#)

## Collaboration

- 📄 [Engineering on Slack](#)
- 📄 [Proposals and ADRs](#)
- 📄 [Skills Matrix](#)

## Infrastructure

- 📄 [Continuous Integration](#)
- 📄 [Domain and DNS setup](#)
- 📄 [Heroku](#)
- 📄 [AWS](#)
- 📄 [Data storage](#)



# Engineering Wiki <Example>

🔗 2 backlinks

## Guides & Processes

- 📄 [Using the Wiki](#)
- 📄 [Onboarding](#)
- 📄 [Incident Response](#)
- 📄 [Data security](#)

## Codebase

- 📄 [Development Lifecycle](#)
- 📄 [Engineering Guidelines](#)
- 📄 [Applications](#)
- 📄 [Services](#)

## Testing

- 📄 [Automated Testing](#)
- 📄 [Browser support](#)
- 📄 [TODO: How to QA](#)
- 📄 [Penetration testing](#)
- 📄 [A/B testing](#)

## People

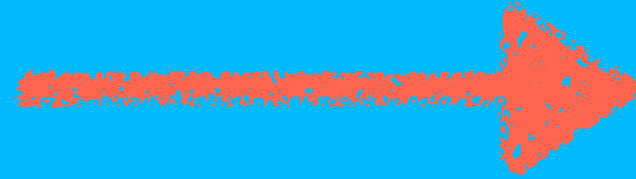
- 📄 [Who's Who](#)
- 📄 [Progression & Promotions](#)
- 📄 [Tech Leading at Farewill](#)
- 📄 [Company Rituals](#)

## Collaboration

- 📄 [Engineering on Slack](#)
- 📄 [Proposals and ADRs](#)
- 📄 [Skills Matrix](#)

## Infrastructure

- 📄 [Continuous Integration](#)
- 📄 [Domain and DNS setup](#)
- 📄 [Heroku](#)
- 📄 [AWS](#)
- 📄 [Data storage](#)



# Engineering Wiki <Example>

🔗 2 backlinks

## Guides & Processes

- 🚚 [Using the Wiki](#)
- 🚚 [Onboarding](#)
- 🚚 [Incident Response](#)
- 🚚 [Data security](#)

## Codebase

- 🔄 [Development Lifecycle](#)
- 🚗 [Engineering Guidelines](#)
- 🧩 [Applications](#)
- 📁 [Services](#)

## Testing

- 🚗 [Automated Testing](#)
- 🌐 [Browser support](#)
- 📝 [TODO: How to QA](#)
- 🔒 [Penetration testing](#)
- 📄 [A/B testing](#)

## People

- 👤 [Who's Who](#)
- 🚀 [Progression & Promotions](#)
- 👤 [Tech Leading at Farewill](#)
- 👤 [Company Rituals](#)

## Collaboration

- 👥 [Engineering on Slack](#)
- 💡 [Proposals and ADRs](#)
- 👤 [Skills Matrix](#)

## Infrastructure

- 🌐 [Continuous Integration](#)
- 🌐 [Domain and DNS setup](#)
- 📄 [Heroku](#)
- 📄 [AWS](#)
- 📄 [Data storage](#)



# Engineering Wiki <Example>

🔗 2 backlinks

## Guides & Processes

- 🚚 [Using the Wiki](#)
- 🚚 [Onboarding](#)
- 🚚 [Incident Response](#)
- 🔒 [Data security](#)

## Codebase

- 🔄 [Development Lifecycle](#)
- 🚚 [Engineering Guidelines](#)
- 🧩 [Applications](#)
- 📦 [Services](#)

## Testing

- 🚚 [Automated Testing](#)
- 🌐 [Browser support](#)
- 📄 [TODO: How to QA](#)
- 🔒 [Penetration testing](#)
- 📄 [A/B testing](#)

## People

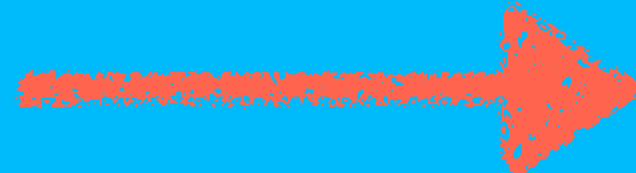
- 👤 [Who's Who](#)
- 🚀 [Progression & Promotions](#)
- 👤 [Tech Leading at Farewill](#)
- 👤 [Company Rituals](#)

## Collaboration

- 👤 [Engineering on Slack](#)
- 💡 [Proposals and ADRs](#)
- 👤 [Skills Matrix](#)

## Infrastructure

- 🔄 [Continuous Integration](#)
- 🌐 [Domain and DNS setup](#)
- 🏠 [Heroku](#)
- 📄 [AWS](#)
- 📄 [Data storage](#)



# Engineering Wiki <Example>

🔗 2 backlinks

All things engineering: processes, best practices, setup guides, and more!

## Guides & Processes

- 🚚 [Using the Wiki](#)
- 🚚 [Onboarding](#)
- 🚚 [Incident Response](#)
- 🔒 [Data security](#)

## Codebase

- 🔄 [Development Lifecycle](#)
- 🚚 [Engineering Guidelines](#)
- 🧩 [Applications](#)
- 📦 [Services](#)

## Testing

- 🚚 [Automated Testing](#)
- 🌐 [Browser support](#)
- 📄 [TODO: How to QA](#)
- 🔒 [Penetration testing](#)
- 📄 [A/B testing](#)

## People

- 👤 [Who's Who](#)
- 🚀 [Progression & Promotions](#)
- 👤 [Tech Leading at Farewill](#)
- 👤 [Company Rituals](#)

## Collaboration

- 👤 [Engineering on Slack](#)
- 💡 [Proposals and ADRs](#)
- 👤 [Skills Matrix](#)

## Infrastructure

- 🔄 [Continuous Integration](#)
- 🌐 [Domain and DNS setup](#)
- 🏠 [Heroku](#)
- 📄 [AWS](#)
- 📄 [Data storage](#)

**Optimise for readability**

# Incident Response

Owner: Kaitlyn Tierney  
Last edited time: May 19, 2023 10:47 AM

11 backlinks

Spotted an incident in progress? Check out the [Quick Guide to incident.io!](#)

[Something is wrong with production](#)

[Oh no, I've deployed a bad change](#)

[Something is literally on fire](#)

[When we know a traffic spike is expected...](#)

[Who is on-call?](#)

Further reading

An incident occurs when something goes wrong with our products, services or internal tools. To respond effectively, we seek understand what happened, and fix it. We record what has happened, and try to learn from it to prevent the same or similar thing going wrong in future.

This full process—identifying a problem, fixing it, and using our learnings to mitigate against future occurrences—is what we mean when we talk about "incident response".

Because we have responsibilities to various regulatory authorities, it's essential that we follow our documented procedures for incident response. **We must respond to incidents in a timely manner**, involving stakeholders across the business to ensure our response is adequate and compliant with our regulatory requirements.

For external services, we have a Slack channel for [Third Party Service Alerts](#) and another for [#service-notifications](#) which receive automated notifications about deprecations and issues.

[Something is wrong with production](#)

[Quick Guide to incident.io](#)

[1 Triaging an incident](#)

[2 Responding to an active incident](#)

[3 Post-incident response](#)

[Incident post-incident reviews](#)

[Run Books](#)

[Oh no, I've deployed a bad change](#)

[Something is literally on fire](#)

[Data Recovery](#)

[When we know a traffic spike is expected...](#)

[Big traffic plan](#)

[Who is on-call?](#)

[Out of hours production alerts](#)

[Something is wrong with production](#)

[Quick Guide to incident.io](#)

[1 Triaging an incident](#)

[2 Responding to an active incident](#)

[3 Post-incident response](#)

[Incident post-incident reviews](#)

[Run Books](#)

[Oh no, I've deployed a bad change](#)

[Rolling back changes](#)

## Example incidents

- We deploy a new version of our Wills app that breaks a customer journey, we need to identify the problem, debug it and revert the change.
- Google docs has an issue that means we can't create new docs and therefore we can't create new wills for customers. We need to find a work around so that our will writers can continue to serve customers until Google docs recovers.
- We realise that a database instance is about to run out of disk space and we need to take action urgently to prevent disruption to online services.

## What is not an incident

The most common type of issue which wouldn't be classed as an incident is something where something isn't working on one individual's computer.

In which case it's probably something for [#tech-support](#).

1. My computer can't connect to the internet, so I can't work.
2. I am locked out of my account, and it's disrupting my work.

Next step: [2](#) Responding to an active incident

# 2

## Responding to an active incident

1 backlink

If you're not yet sure of the scale and severity of the incident, go back to [1](#) Triaging an incident.

We use [incident.io](#) to manage our incident response. Check out the [Quick Guide to incident.io](#) to get started.

[Creating a new incident](#)

[Helping out on an existing incident](#)

[Acting as an Incident Lead](#)

[Assign other roles](#)

[Coordinate the response](#)

[Update the incident status](#)

[Complete the post-incident review](#)

[Acting as Communications Lead](#)



# Using the Wiki

[Contributing](#)

[Content guidelines](#)

[Good wiki content can be...](#)

## Contributing

A good wiki is a collaborative effort. Here are a few ways you can help:

1. If you spot something outdated, incorrect, or missing here, take a crack at fixing it yourself.
2. If you'd like a hand or have any questions, pop into [#eng-wiki](#) on Slack and our resident doc-enthusiasts will be happy to help.
3. Try to stick to the content guidelines below—they're designed to keep our docs accessible to everyone and as useful and maintainable as possible
4. To avoid things getting too cluttered, consider asking in Slack for a second opinion before adding new top-level docs to the [wiki's main page](#).

## Content guidelines

- This wiki is predominantly for engineers and engineering managers, but we want to make it approachable and understandable. A technical product owner or engineer just starting out should be able to have their questions answered by reading a relevant wiki page when one is available. (When one isn't available, we may want to document the answer for the next person who has the same question)
- Feel free to use technical language as-needed. But aim for readability where possible: short, declarative sentences.
- Use the industry-standard terms to make things easier to search for.
- Try to strike a balance between keeping things concise and providing enough context.
- Feel free to link out to external docs for tools we use. But when you spot dead links, try to update them.
- Consider the information architecture of topics ordered in the sidebar when adding a new page

## Good wiki content can be...

- Evergreen, or things that change infrequently
- Impacting multiple product teams or related to multiple codebases
- Referred or shared with others often

## Content guidelines

- This wiki is predominantly for engineers and engineering managers, but we want to make it approachable and understandable. A technical product owner or engineer just starting out should be able to have their questions answered by reading a relevant wiki page when one is available. (When one isn't available, we may want to document the answer for the next person who has the same question)
- Feel free to use technical language as-needed. But aim for readability where possible: short, declarative sentences.
- Use the industry-standard terms to make things easier to search for.
- Try to strike a balance between keeping things concise and providing enough context.
- Feel free to link out to external docs for tools we use. But when you spot dead links, try to update them.
- Consider the information architecture of topics ordered in the sidebar when adding a new page

## Good wiki content can be...

- Evergreen, or things that change infrequently
- Impacting multiple product teams or related to multiple codebases
- Referred or shared with others often

**4. Find what you need**





**Create a finding aid**

# Proposals

Default view Technical +

Date submitted 1 rule Status: Done + Add filter

Proposal	Authors	Date
Upgrade Backstage Staging DB Tier	Kaitlyn Tierney	March 1
Use Heroku Redis for all apps	Kaitlyn Tierney	January
GraphQL for customer-facing APIs in Backstage	Kaitlyn Tierney	October
Introducing a merge freeze tool into our projects		Septem
Backstage codebase ownership		May 20,
Storybook for Backstage		March 3
Start tackling tech debt (Wills & LPAs team)		March 1
Add a configuration management tool (Doppler)		March 1
Rich text		Februar
Transfer ownership of Spotlight transformations		January
Rename master branches to main	Kaitlyn Tierney	January
Cypress testing in farewill.com		January
Heroku vs Netlify for CMS Preview Mode - Proposal		January
Improve incident management + use an incident management tool		

	Date of occur...	Severity	Codebases	3rd parties	Author
outages	April 6, 2022	Minor	all	CircleCI	
counts and	April 4, 2022	Major	wills fro	farewill.	backsta

## Meeting notes archive

Previous meetings

### Engineering Guild meetings

	Person	Recording Link
@April 25, 2023		https://drive.google
@April 11, 2023		
@April 4, 2023		https://drive.google
@March 21, 2023		https://drive.google
@March 14, 2023		
@March 7, 2023		https://drive.google
@February 28, 2023		https://drive.google
@February 21, 2023		https://drive.google
@February 14, 2023		https://drive.google
7 February 2023		https://drive.google
31st January 2023		https://drive.google
24th January 2023		https://drive.google
@January 17, 2023		https://drive.google
@January 10, 2023		https://drive.google
@January 3, 2023		NA - meeting canc
@December 20, 2022		https://drive.google
Le 13 décembre		https://drive.google
29th November 2022		https://drive.google
@November 22, 2022		https://drive.google
@November 15, 2022		https://drive.google
@November 8, 2022		https://drive.google
@November 1, 2022		https://drive.google

COUNT 80

Add cover Hide description

## Docs

Use this template to organise documents like technical specs, architecture overviews, and project kickoff notes. Add additional database properties as you see fit.

Click All Docs to switch between different views.

All Docs Created By Me +

Created + Add filter

- React-scripts v5 issues
- Heroku Salesforce Cost Optimisation Meeting Notes
- Wrap up list
- Heroku Tier Pricing Changes
- Further suggestions on probate-cases typescript work
- Q4 - Senior Engineers Team Decisions
- Platform Eng Info Dump - Notes
- Cloudflare Multiple Users
- AWS LIDS Meeting notes
- Dependa-day™ Guidance
- 2022 Q3 OKR Planning
- Introductory meeting with AWS
- IBM Cloud Certificate Discovery
- Onboarding retro notes
- Offboarding queries
- Questions for onboarding
- Technical Principles
- Refreshing Papertrail Credentials
- Chats about migrating Wills DB to Backstage
- Roadmap for migrating Wills DB to Backstage
- Alternatives to Heroku
- Migrating wills MongoDB to a different DBaaS provider

**Optimise search results**

# Add metadata



## Software Architecture

Owner: Kaitlyn Tierney | Last edited time: Today 10:41 AM | Tags: **Diagrams**, Architecture

1 backlink

This page collects links to architectural diagrams and other documents relevant to our overall software architecture, or the architecture of individual projects. For more fine-grained architectural discussion, please see our database of [Architecture Decision Records](#).

### Architectural Overview

[Architecture Diagram for Backstage on Miro](#)

Created by [redacted], c. 2022

Contains details about Backstage, Farewill.com, Wills app, MongoDB, Redis, Postgres

[backstage-architecture.png](#)

[backstage-get-contacts.png](#)

### Diagrams on Github

We have a [GitHub repository for diagrams](#), which can be helpful for visualising how services are [connected](#). It contains sequence diagrams which are created using mermaid. Diagrams in this repository were primarily created or updated prior to 2021.

### Services

[An overview of internal and external services on Miro](#)

Created by [redacted], c. 2021

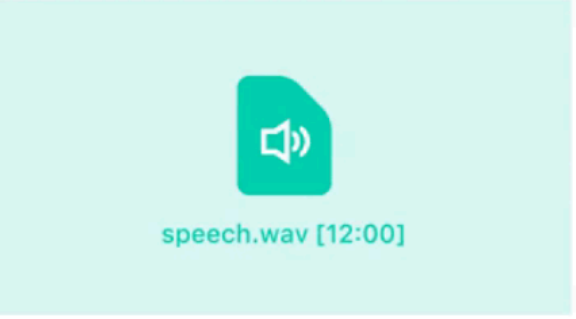
## Nobody transcribes like Descript

Automatic transcription with industry-leading accuracy. Near-instant turnaround. And it costs pennies per minute.

TRANSCRIPTION

- A** Automatic
- W** White Glove
- M** Multitrack
- B** Import Transcript


### Automatic Transcription



speech.wav [12:00]

95% accurate and ready in a few minutes.  
Good for transcribing files quickly.

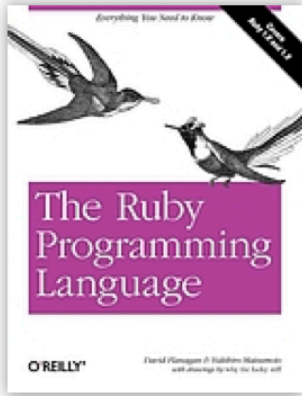
# Add transcripts



## Aiko

AI-powered audio transcription

# Use keywords



**The Ruby programming language** ★★★★★☆ [35 reviews](#)

Authors: [David Flanagan](#), [Yukihiro Matsumoto](#)

Summary: A guide to Ruby programming covers such topics as datatypes and objects, expressions, classes and modules, control structures, and the Ruby platform

Print Book, English, 2008

Edition: 1st ed [View all formats and editions](#)


Publisher: O'Reilly, Beijing, 2008

Physical Description: xi, 431 pages : illustrations ; 24 cm

ISBN: 9780596516178, 0596516177

OCLC Number / Unique Identifier: 178724855


Subjects: [Object-oriented programming \(Computer science\)](#)  
[Programmation orientée objet \(Informatique\)](#)  
[Ruby \(Computer program language\)](#)  
[Ruby \(Langage de programmation\)](#)  
[Ruby \(programspråk\)](#)



Contents lists available at [ScienceDirect](#)

## NeuroImage

journal homepage: [www.elsevier.com/locate/neuroimage](http://www.elsevier.com/locate/neuroimage)



**Slow phase-locked modulations support selective attention to sound**

Magdalena Kachlicka<sup>a</sup>, Aeron Laffere<sup>a</sup>, Fred Dick<sup>a,b</sup>, Adam Tierney<sup>a,\*</sup>


<sup>a</sup> Department of Psychological Sciences, Birkbeck, University of London, Malet Street, London WC1E 7HX, England  
<sup>b</sup> Division of Psychology & Language Sciences, UCL, Gower Street, London WC1E 6BT, England

**ARTICLE INFO**

**Keywords:**  
Attention  
Auditory  
Temporal  
EEG

**ABSTRACT**

To make sense of complex soundscapes, listeners must select and attend to task-relevant streams while ignoring uninformative sounds. One possible neural mechanism underlying this process is alignment of endogenous oscillations with the temporal structure of the target sound stream. Such a mechanism has been suggested to mediate attentional modulation of neural phase-locking to the rhythms of attended sounds. However, such modulations are compatible with an alternate framework, where attention acts as a filter that enhances exogenously-driven neural auditory responses. Here we attempted to test several predictions arising from the oscillatory account by playing two tone streams varying across conditions in tone duration and presentation rate; participants attended to one stream or listened passively. Attentional modulation of the evoked waveform was roughly sinusoidal and scaled with rate, while the passive response did not. However, there was only limited evidence for continuation of modulations through the silence between sequences. These results suggest that attentionally-driven changes in phase alignment reflect synchronization of slow endogenous activity with the temporal structure of attended stimuli.



**But don't rely on search  
exclusively**

**Redundancy = reliability**



**What tool to use?**

## Wikis

It's hard to move fast if you can't find anything. Centralize all your knowledge in Notion.

[Explore →](#)



- Acme Inc. ↕
- Search
- Updates
- All teamspaces
- Settings & members

Teamspaces

- General
- Engineering
- > Engineering wiki**
- > Meeting notes
- > Projects
- Design
- Product
- Marketing
- Sales
- HR & Legal 🔒



# Engineering wiki

All things engineering: processes, best practices, setup guides, and more!

## Guides & Processes

- 🚜 [Getting started](#)
- 🤖 [Engineering guidelines](#)
- 🔄 [Development lifecycle](#)
- 🚢 [Getting started](#)

## Codebase

- ✅ [Code reviews](#)
- 🔗 [React](#)
- 🖥️ [Backend](#)
- 📁 [Engineering directory](#)

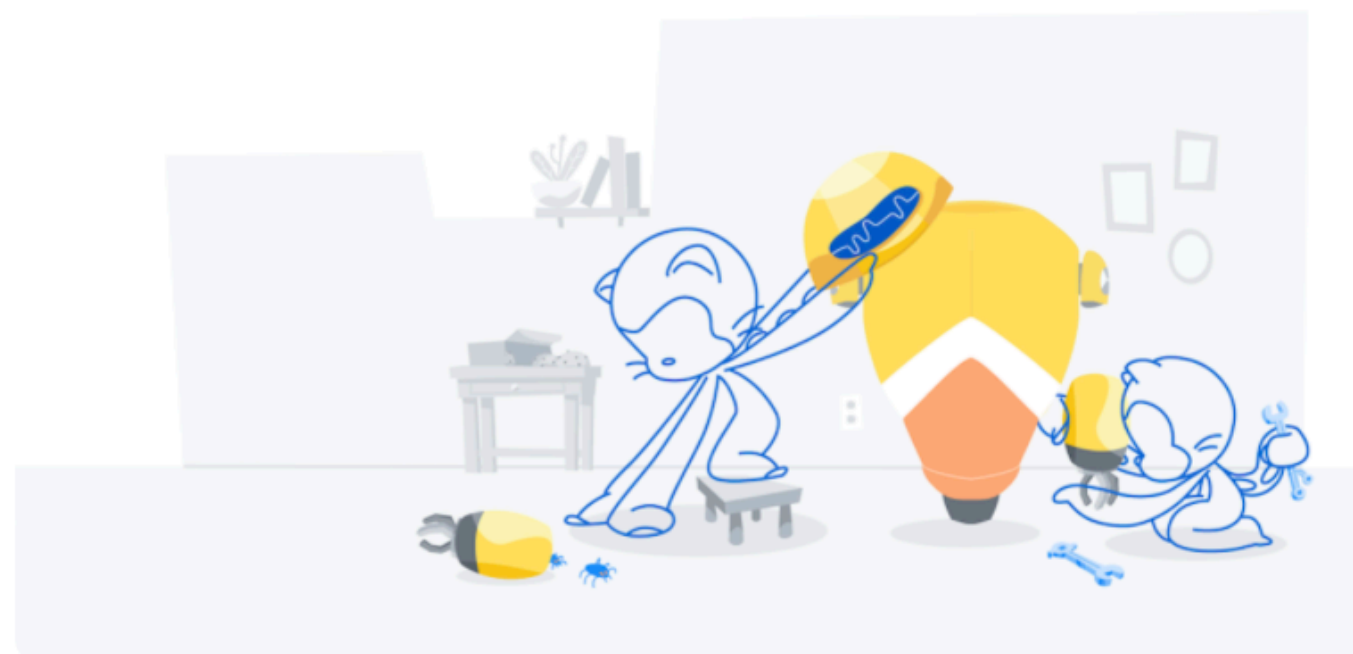
🗃️ All 📅 Timeline 📊 Board

## 🚗 Projects

Aa Name 👤 Product owner 👤 Team

GitHub Blog Search

# Search Results for: *wiki*

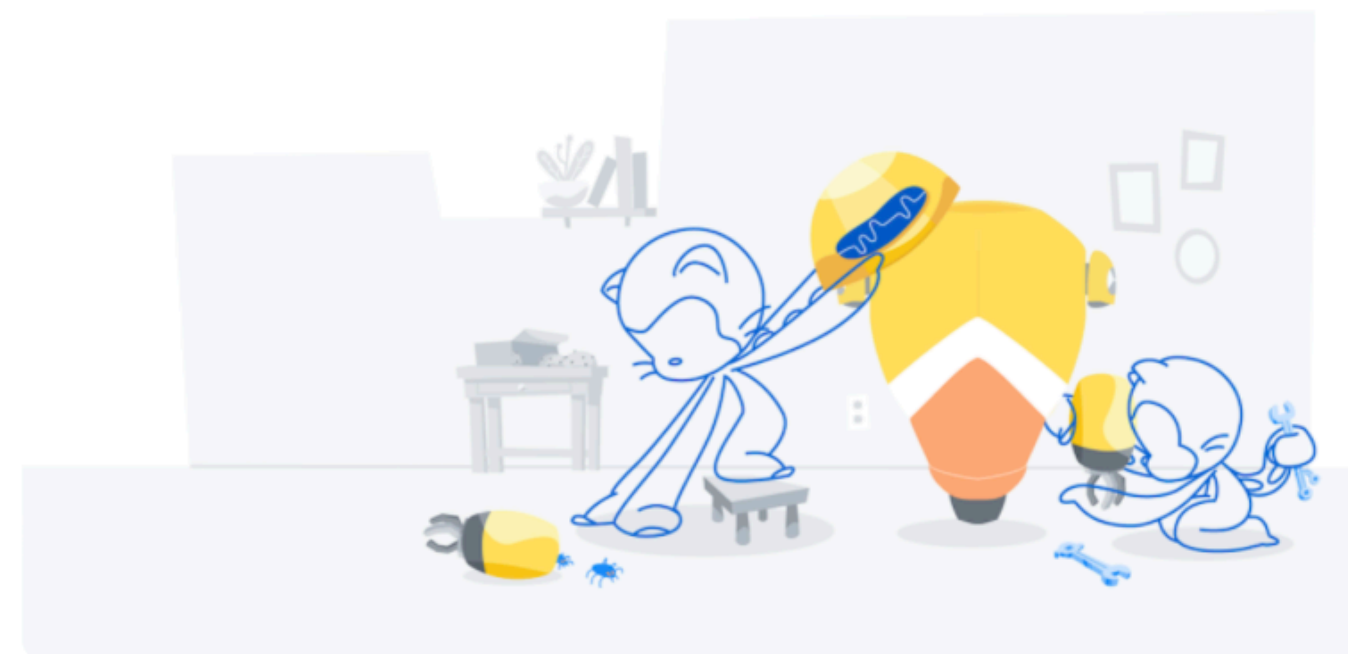


Changelog

## Wikis now support math and Mermaid diagrams

Wikis now support math and Mermaid diagrams

**Kevin Duck** August 9, 2022

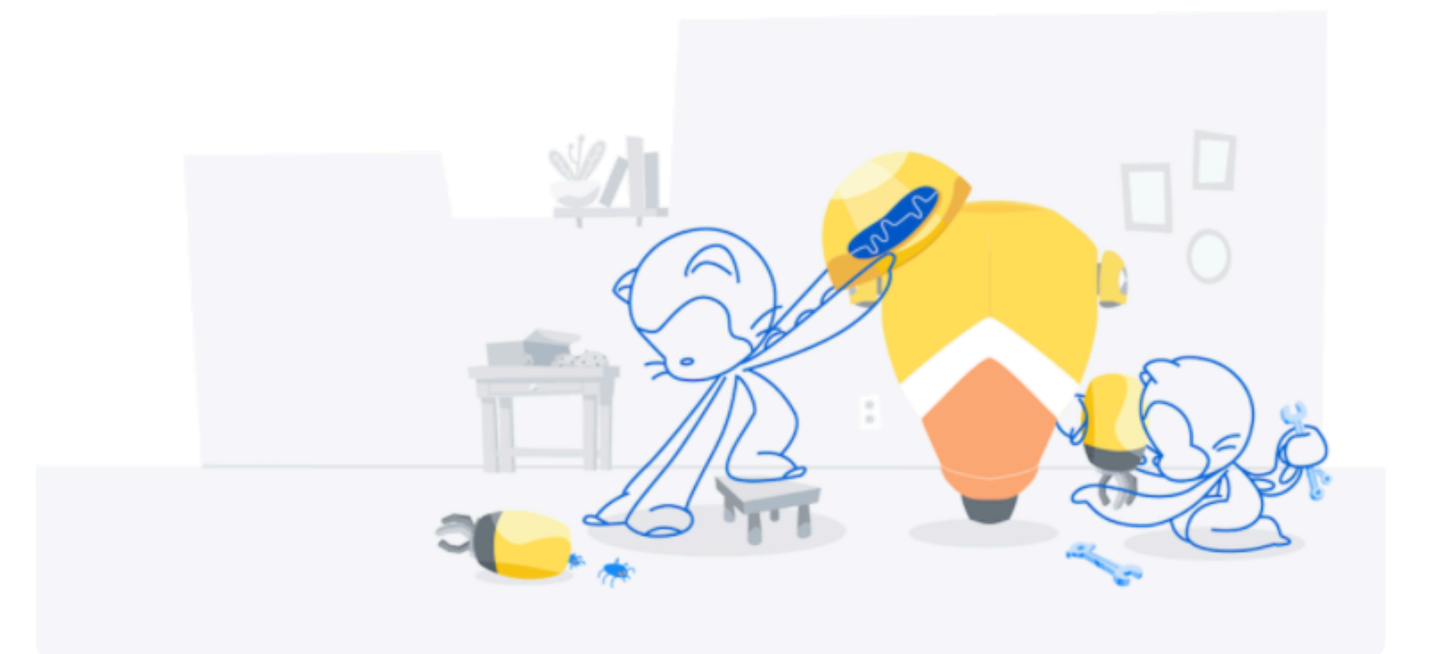


Changelog

## Upload images to Wiki pages

Upload images to Wiki pages

**Sidi Merzouk** February 14, 2022



Changelog

## Table of content for Wikis

Table of content for Wikis

**phanatic** August 19, 2021

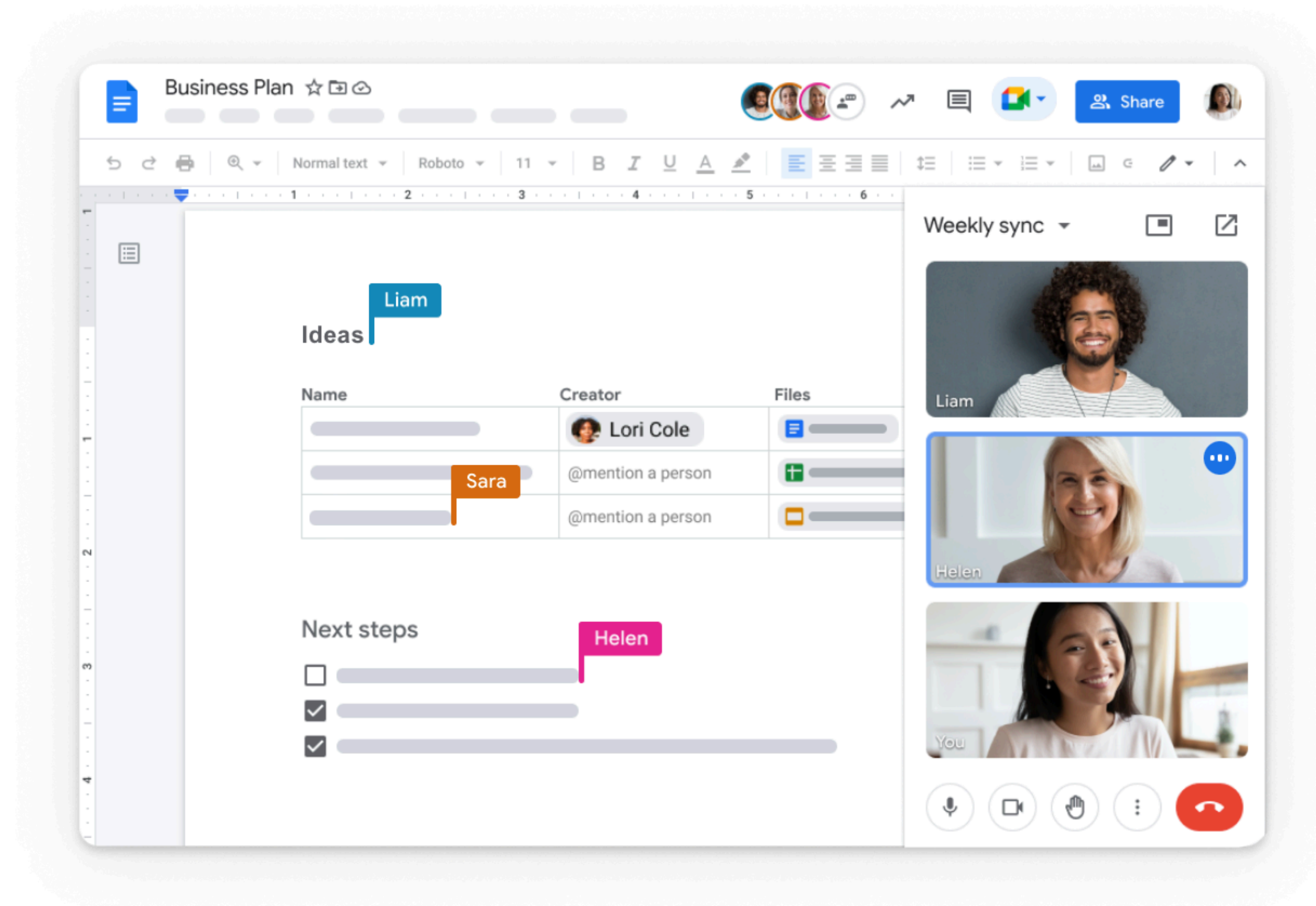
# Build your best ideas together, in Google Docs

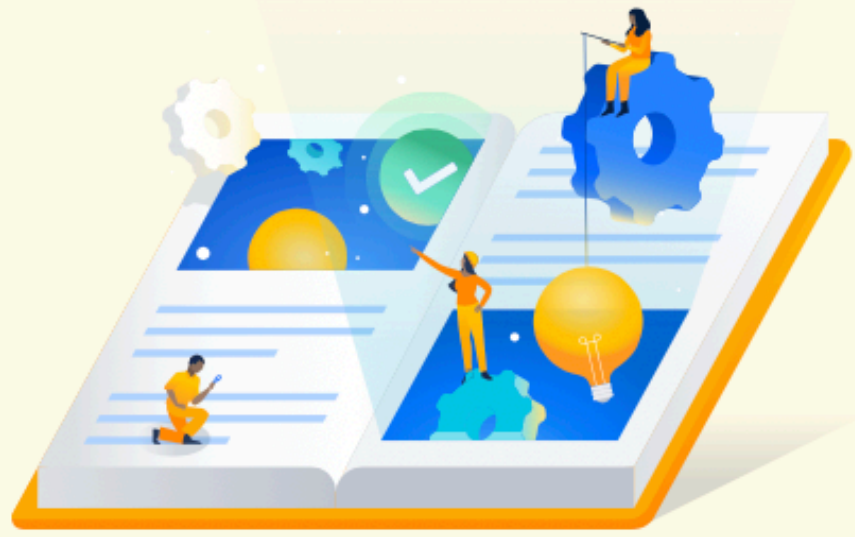
Create and collaborate on online documents in real-time and from any device.

[Try Docs for Work](#)

[Go to Docs](#)

Don't have an account? [Sign up for free](#)

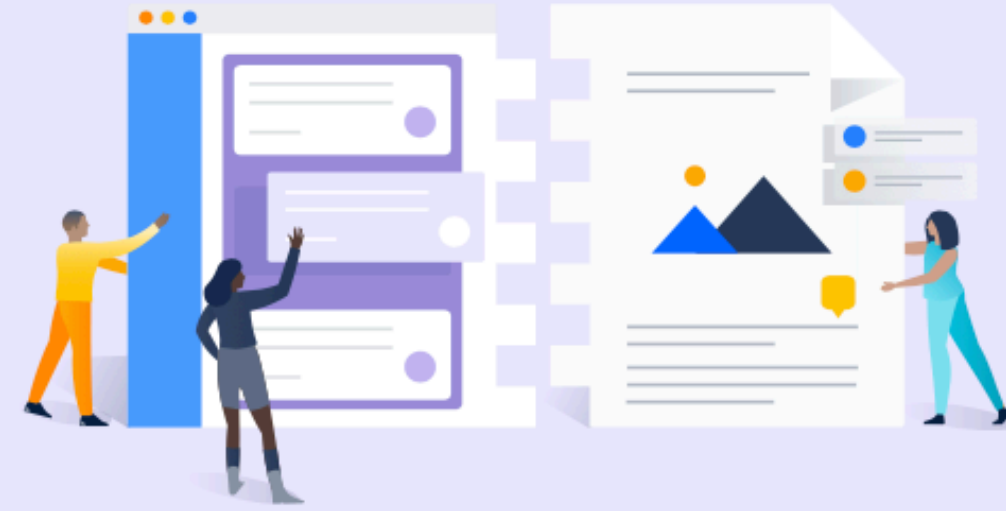




## A shared knowledge hub for every team

Make it easy for teams to find and contribute critical company knowledge.

[View use case →](#)



## Streamline work with Confluence and Jira

Connect Confluence to Jira Software to organize, manage, and ship projects faster.

[View use case →](#)



## Take the work out of teamwork

Collaborate openly and efficiently without meetings or emails.

[View use case →](#)

## How would you like to use Confluence?

Explore all the ways all kinds of teams use Confluence as their collaborative knowledge hub.

[See all use cases →](#)

**5. Share widely**

**Start small**

**Solicit feedback**



**Find your champions**


**Trust in the process**

**Show your work**

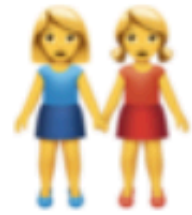
  3 days ago

I think there's still outstanding questions that don't have well formed answers yet, but I know they're being discussed / in the backlog currently. So I can update the docs after they're worked through (edited)

 **Kaitlyn Tierney** < 1 minute ago

@ I think the question you had is pretty straightforward but wasn't very clear from the existing doc, so I just added this paragraph to the top to make things a bit more clear! Let me know if you think that helps or if there's anything else you'd like the doc to cover.

Screenshot 2023-05-15 at 09.52.58.png ▼



## Couples wills

← 1 backlink

### Introduction

In April 2019 we launched "couples wills" - an extension of the online will product. A "couples will" is where one user pays for two wills (and two subscriptions if they opt in to unlimited changes). Their partner can then submit their own will for checking without entering any card

✚ Ask AI Text ↗ Link Comment B I U S <> √x A @ ...

A customer will only have the option to pay for two wills at checkout *if they have invited their partner*. If they have invited a partner, they still have the option to pay for a single will instead. At checkout, there is a toggle allowing them to choose whether they'd like to pay for only a single will, or to pay the couple's price for two wills.

Prior to April 2019, a user could share their 50% discount code with their partner, meaning the total for two wills was £90 + £45 = £135.

 25 minutes ago

Hey! A couple of wills app questions:

- as a user in the wills app, can you purchase a couples will at checkout **without** having invited a partner?
- as a user in the wills app that has invited a partner, can i then choose to only purchase my own single will at checkout, and allow the invited partner to take care of their own purchase?

2 replies

 13 minutes ago


- as a user in the wills app, can you purchase a couples will at checkout **without** having invited a partner?

No, you have to have invited them to Farewill (it's triggered by the `linked_user` document). Even if the user has told us they're married / cohabiting / etc, if their partner isn't invited to Farewill, they cannot purchase a couples will.

- as a user in the wills app that has invited a partner, can i then choose to only purchase my own single will at checkout, and allow the invited partner to take care of their own purchase?

Yes, there is a toggle that allows the user to switch between paying for both / paying for only their will. The default is paying for both.

 **Kaitlyn Tierney** 1 minute ago

Hey @ we have some product documentation around [Couples Wills here on Notion](#). Could you take a look through there and see if:

- a) your questions would have been answered by reading it, and
- b) update the docs if not, so that it's documented if anyone else has the same questions later on?





# Outreach

**Pairing: not just for code!**



**Incentivise**

# Share ownership

**Reap the rewards**

0. Take a backup
1. Start sorting
2. Create your taxonomy
3. Set your style
4. Find what you need
5. Share widely

**Focus on the *why***

Example 1

# **ADR advice process**




## **A Commune in the Ivory Tower? - A new approach to architecture decisions**

Andrew Harmel-Law introduces a mindset and an associated set of practices which do away with the traditional idea of “Architects” while bringing the practice of “Architecture” to the fore.



... / Proposals / Adopting the "Advice Process" for ... Edited Oct 21



## Adopting the "Advice Process" for technical decisions

Authors: Kaitlyn Tierney, Helena Thompson

Date submitted: June 14, 2022

Size: Small

Status: Done

Type: Process

Scope: Engineering

## Supporting changes

This updated ADR process would include the following small changes:

- Updating the ADR template to include a new section for "Advice", in which advice can be recorded.
- Updating the Engineering Guild meeting template to include a section for ADRs to be discussed.
- Storing all ADRs in Notion, instead of in a project's repository, for better visibility and to make it easier to capture which ADRs have a 'Proposed' status.
- Updating our ADR documentation to reflect these changes and to highlight great example ADRs for engineers to draw inspiration from.

This is a low-risk initiative. If we try it for a few months and it doesn't add value, or engineers don't seem to get any benefit or enjoyment out of it, or no learnings result, then we can abandon it. But we think it's a good bet that discussing how we plan, architect, and build software, as a group, will lead to better software with more internal consistency, and happier, more engaged engineers, and will cut down on siloes of knowledge within teams.

## Proposal

Thus, we propose to introduce a lightweight version of the "Advice Process"[2] into our engineering workflow. The original "Advice Process" as defined by its authors is as follows:

- **The Rule:** anyone can make an architectural decision.
- **The Qualifier:** before making the decision, the decision-taker must consult two groups: The first is everyone who will be meaningfully affected by the decision. The second is people with expertise in the area the decision is being taken.

We'd like to bring this in even for smaller, non-architectural decisions—most particularly when they impact on *shared codebases* like Backstage, but also any time it seems like a decision could spark interesting technical discussion, or might benefit from some wider consultation. Realistically, we're not often (hopefully) architecting new systems, but we do have a lot of little decisions that are made on a regular basis, all of which can add up to making our codebases less easy to understand, or harder to maintain, if we don't agree on how to approach things.

## Suggested Implementation

If we agree on the proposal in principle, we have mapped out a suggested way of implementing the "Advice Process" into our existing practices.

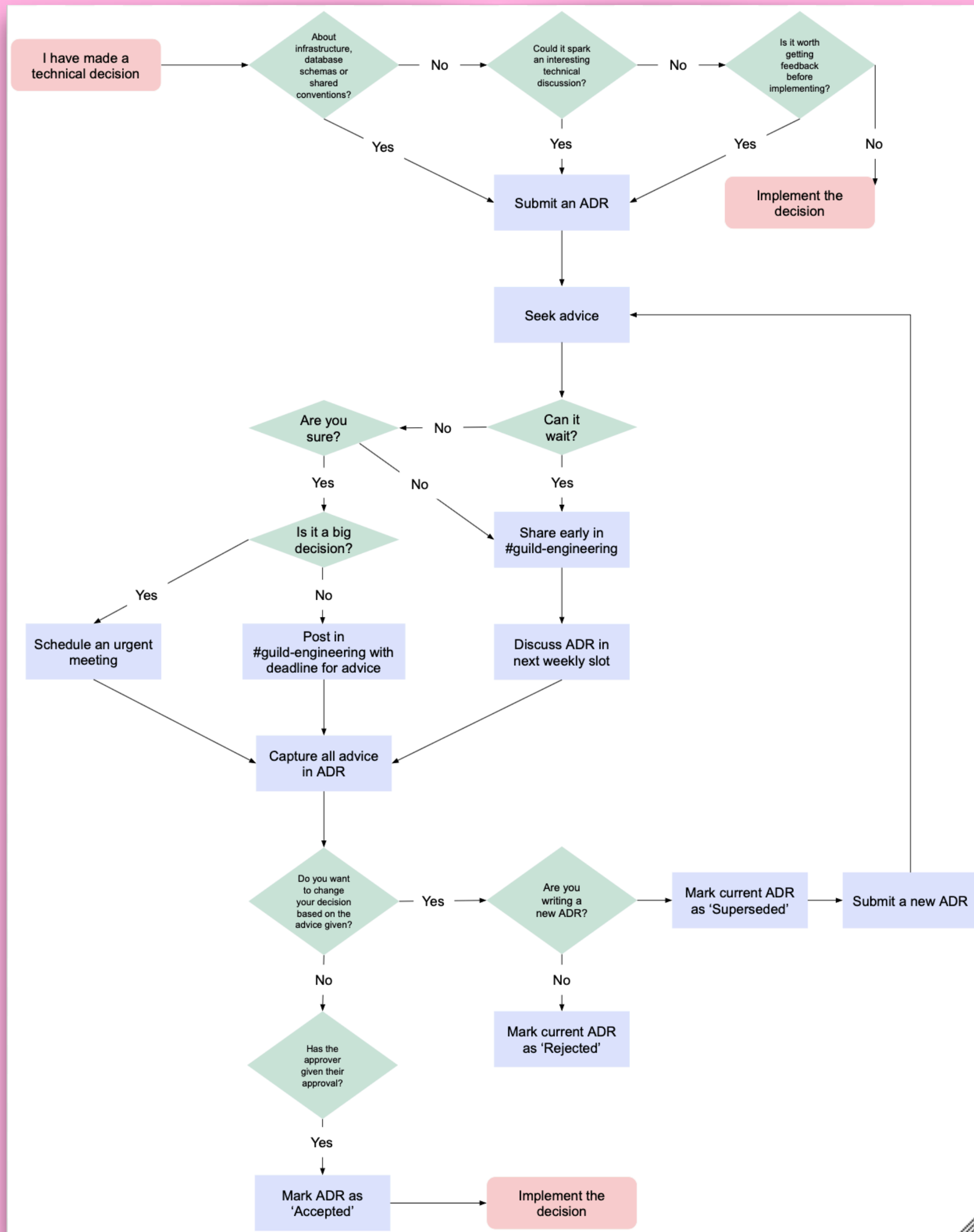
## Updated ADR process

We could update our ADR process to facilitate these consultations. This would entail:

1. For each decision, write a lightweight ADR (using our existing [Architecture Decision Records](#) database). These can be short and sweet, and ideally would be written about any software design decision that impacts on other engineers—anything from adding a new column or table to a database, to introducing a new pattern or shared type to a codebase.
2. Once a 'Draft' ADR is ready to be shared, it is 'Proposed' and posted in #guild-engineering. This gives people an opportunity to think about the ADR and leave advice ahead of discussing in person.
3. All ADRs that are 'Proposed' are then discussed in the next Engineering Guild meeting. The ADR author gives an overview, clarifies any existing advice and seeks further advice.

*For more urgent work, schedule an advisory meeting, inviting any engineers who care to attend, and any other affected people.*





# Architecture Decision Records

Owner: Kaitlyn Tierney  
 Verification: Until July 31, 2023  
 Last edited time: Today 12:10 PM

4 backlinks Add comment

We use [architecture decision records \(ADRs\)](#) to propose and document technical decisions we make on our projects.

Each ADR has an identical format, comprised of the following sections:

- **Title** - Short description stating the architecture decision
- **Status** - Draft, Proposed, Accepted, Superseded, Rejected
- **Approver** - The person whose approval is required for acceptance
- **Context** - Motivation behind a decision
- **Decision** - Decision and corresponding justification
- **Consequences** - Impact of a decision, whether positive, negative or neutral
- **Advice** - Capturing advice given when the ADR is proposed

## When should I write an ADR?

As a result of a successful proposal ([Adopting the "Advice Process" for technical decisions](#)), we're trialing writing ADRs for a wider pool of decisions (not just big architectural decisions). This is to help promote more technical discussions guild-wide which should level up the quality of our software as well as provide more opportunities for learning and development.

An ADR should be written *before* implementation so that we're not wasting engineering time. That said, a prototype implementation (i.e. not production-ready code) can be useful evidence in your ADR to illustrate an approach and its trade-offs. Prototyping is a great way of failing and learning quickly to gain confidence in our decisions.

We should write an ADR whenever we're proposing to change our:

- Infrastructure (example: [Heroku Advanced Scheduler](#))
- Database schema, for major architectural changes (example: [Representing crematoria providers in the database](#))
- Shared conventions (example: [Set port numbers in Doppler for local development](#))

We also encourage ADRs to be written *whenever* a decision could spark interesting technical discussion or might benefit from wider consultation.

Although it might seem like additional effort to write an ADR, it should save us a bunch more time down the line! It might take ~30 minutes to write an ADR but that



**Kees Heuperman** 13:58

Having a record of the motivation behind decisions is incredibly valuable



**Alexandra O'Mahony** 14:44

I'm a big convert to ADRs! It's helpful for the person working on it to have their thoughts organised in one place and the feedback I've received has had a massive impact on my personal development and on making better decisions as a team (edited)

is that too wordy??

I LOVE ADRs!



**Stephanie Ye** 13:57

“Why on earth did they build it like this???” is a question every programmer has screamed silently into the void. With ADRs, you can at least do your screaming in a more informed manner.

Example 2

# Applying for regulation



# Software Development Lifecycle

6 backlinks

Last updated: 21/12/2022

This page documents our Software Development Lifecycle. The requirements listed here exist to ensure we meet the requirements that allow us to be compliant with PCI-DSS and regulations.

The requirements outlined by this document should be followed at all times. If you have questions or suggestions, drop a message in #guild-engineering.

This page uses synced blocks which are embedded across other relevant engineering-related documentation pages. As a result, this page has restricted write access to prevent any accidental changes being made (via editing the content of the synced blocks in the other pages) without review. If you'd like to suggest a change to anything in this document please reach out to

[Code storage](#)

[Secret management](#)

[Shared logins](#)

[Development machines](#)

[Dependency management](#)

[Secure development](#)

[Branch Protection Rules](#)

[Writing code](#)

[Testing](#)

[Pull requests](#)

## Code storage

1. All of our code is stored in projects on GitHub under the Farewill organisation.
2. Projects are marked as "private" by default, unless the VP Engineering has discussed and agreed that a project may be made public.

⚠ Changing the visibility of an existing project from private to public requires planning to ensure that any potentially confidential information that could lead to security or privacy issues is scrubbed from the repositories code and commit history.

## Secret management

See [Doppler](#).

1. All new projects should store environment variables and secrets in [Doppler](#). This helps us ensure that we have an easily auditable trail of any changes made to these values.
2. If we become aware that any secrets relating to any of our production environments have been inadvertently leaked, firstly, **don't panic!**
  1. Raise an incident so that appropriate remedial action can be taken and so that we have an audit trail of the event.
  2. Ensure that all potentially compromised secrets are rotated.

# Secure development and deployment guidance

Owner



Last edited time

November 22, 2022 4:09 PM

2 backlinks

This page contains information from our [Software Development Lifecycle](#). Please ensure you are familiar with and comply with the requirements outlined there to help keep our systems secure.

It is important that we consider security as a primary concern throughout our development and deployment processes.

## Requirements






1. Ensure security is considered as part of each piece of product work. If there are likely to be potential security related decisions made as part of a piece of work, record the potential risks, options and chosen approach in [an ADR](#).
2. If you discover a vulnerability [raise an incident](#) to handle investigating and performing the remedial action required.
3. If you are concerned about a potential vulnerability or other security issue, discuss with the relevant product engineering team or the guild immediately so we can investigate and (if necessary) respond and mitigate the issue (raising an incident if necessary).
4. Ensure pull request templates in GitHub contain a checklist item reminding engineers to consider (and document) any implications on security when completing the self-review checklist for a pull request.
5. Any potential security risk(s) identified at any point in time should be logged in the [Risk register](#).

Example 3

# The QA Stub






## Codebase

---

-  [Quick Reference CLI](#)
-  [Development Lifecycle](#)
-  [Engineering Guidelines](#)
-  [Applications](#)
-  [Services](#)





## Testing

---

-  [Automated Testing](#)
-  [Browser support](#)
-  [TODO: How to QA](#)
-  [Penetration testing](#)
-  [A/B testing](#)









## Collaboration

---

-  [Engineering on Slack](#)
-  [Proposals and ADRs](#)
-  [Product Announcements](#)
-  [Skills Matrix](#)

## Infrastructure

---

-  [Continuous Integration](#)
-  [Domain and DNS setup](#)
-  [Heroku](#)
-  [Docker](#)
-  [AWS](#)
-  [IBM Cloud](#)
-  [Data storage](#)
-  [Terraform](#)



# TODO: How to QA

Owner: Kaitlyn Tierney  
Last edited time: March 9, 2022 10:30 AM

💡 This template explains our QA process for shipping bug-free software.

## QA Philosophy

### Processes

[Making Code Changes](#)

[Reviewing Code](#)

[QA](#)

## QA Philosophy

+ ☰ Write about your approach to QA and why it's critical to success.

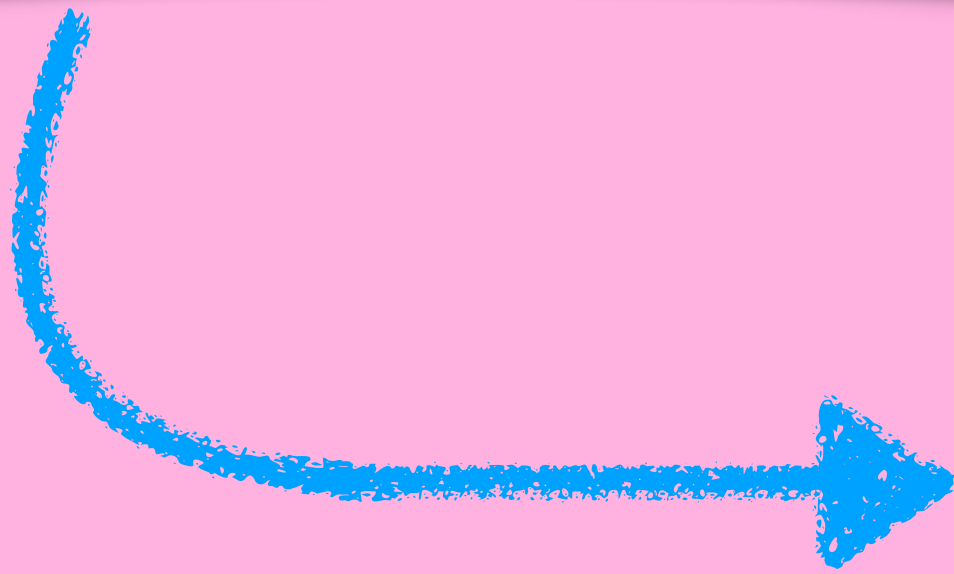
## Processes

### Making Code Changes

- Test PRs rigorously before requesting review.
- Include test cases you've checked for in your PR description.

### Reviewing Code





- **2023-03-14 10:12** : The new probate Acquisition Flow was put live - **This is the point at which the incident began**
- **2023-03-22 12:16** : it was discovered that ads were still pointing to old pages
- **2023-03-22 19:50** : it was discovered that the CTAs on the old pages were not working (unresponsive when clicked)
- **2023-03-22 20:15** : all ads were switched to point towards the new pages

### What could have gone better?

1. We should have a single well-tested component for links, instead, we have multiple components with confusing behaviour. This will be easier to fix when we remove the unused landing pages and drastically decrease the amount of existing components and lines of code.
2. We should have tested the PR better during PR review
3. We should have confirmed that the expected changes in the ad account had been executed
4. We should have end-to-end journey tested when the ads performance dropped

### Commercial Impact

36% below target on prospects during the period in question, having been only 4% under target on prospects in the preceding 7 days. Also 60% below target on prospects in the days following the changes. **Approximate revenue impact of £59k.**





## What could have gone better?

1. We should have a single well-tested component for links, instead, we have multiple components with confusing behaviour. This will be easier to fix when we remove the unused landing pages and drastically decrease the amount of existing components and lines of code.
2. We should have tested the PR better during PR review
3. We should have confirmed that the expected changes in the ad account had been executed
4. We should have end-to-end journey tested when the ads performance dropped

0. Take a backup
1. Start sorting
2. Create your taxonomy
3. Set your style
4. Find what you need
5. Share widely

0. Take a backup

**1. Start sorting**

2. Create your taxonomy

3. Set your style

4. Find what you need

5. Share widely

**Just make a start**

**It will never be perfect**

**...Preferably today**

**Kaitlyn Tierney**  
**k@krtierney.com**