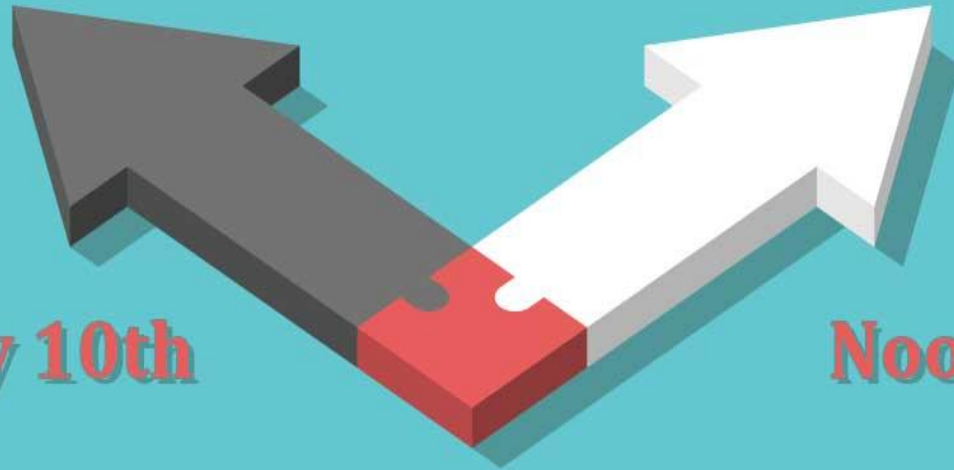




Quality Leadership, Testing, & Governance
Tactics that *Make or Break* Your
Progressive Delivery System



February 10th

Noon, Eastern

The background features a complex network of thin grey lines connecting various points, forming a web-like structure. Scattered throughout are numerous triangles of different sizes and orientations, some solid and some outlined. The overall aesthetic is modern and technical.

Quality Leadership, Testing, and Governance Tactics that Make or Break Your Progressive Delivery System

Jeff Sing
Sr Software QA Engineering Manager, Iterable



Jeff Sing

- Sr QA Manager @Iterable
- 15 Years in QA in a variety of industries
- Progressive Delivery Champion while working at Optimizely



jeff.sing@iterable.com



<https://www.linkedin.com/in/jeffsing/>





Quality Leadership, Testing, and Governance Tactics that Make or Break Your **Progressive Delivery** System



Progressive Delivery

“... a new new basket of skill and technologies concerned with modern software development, testing and deployment. I am thinking of Canarying, **Feature Flags**, A/B testing at scale.”

- James Monk, RedMonk



The Independent

Netflix finally rolling out feature that means users have to browse at all

Netflix is finally rolling out a "shuffle" feature that means users have to browse at all

Dec

Netfli
Early

"You'll
availab
1 week

Eng

Netfli

It's tes
Netflix
1 day a

Tut

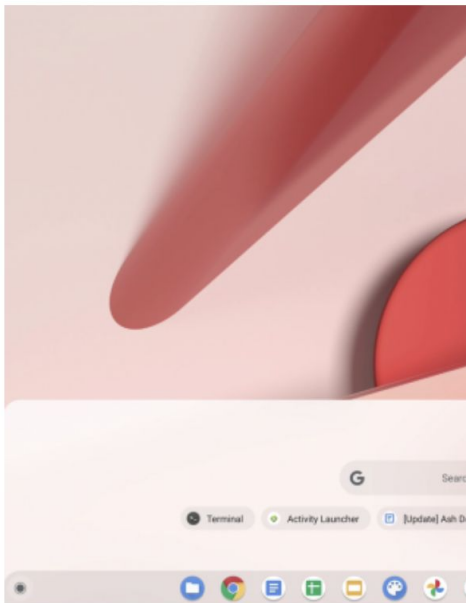
Netfli
Stop

Netfli
if they
1 day a

Kal

Netfli

The sti



A comparison of the new system ther

- **WebUI dark mode**

- A new Chrome flag implements **feature flag** does not switch on with the force dark mode. Enabl

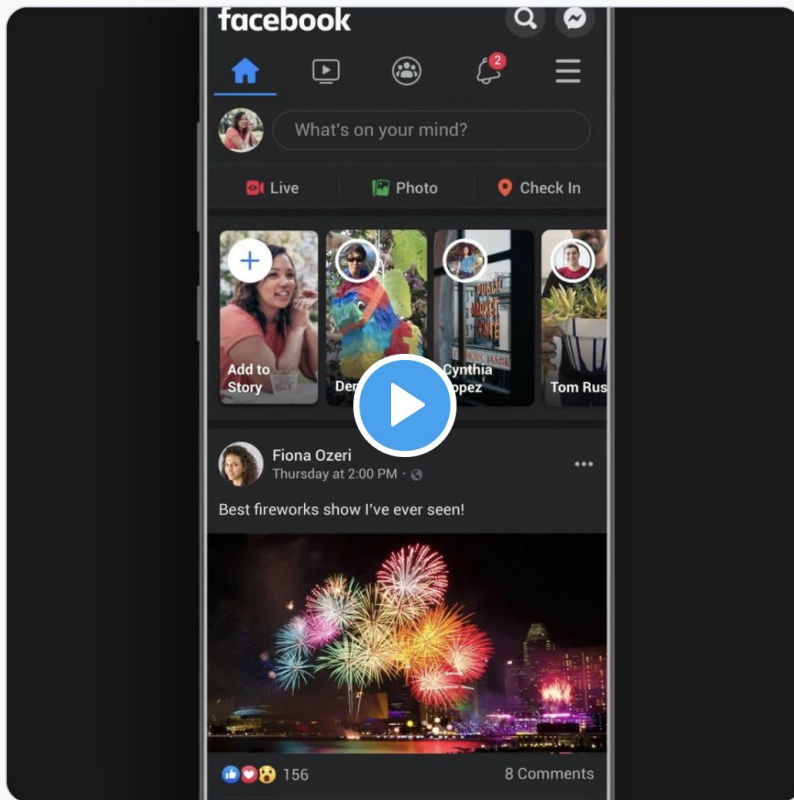


Jane Manchun Wong
@wongmjane



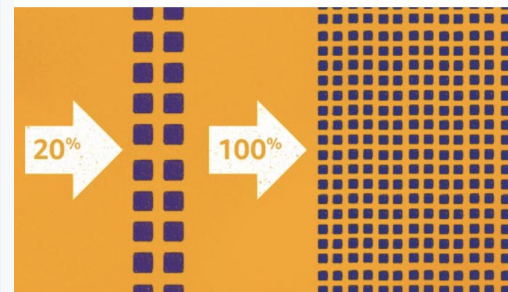
Facebook is publicly testing Dark Mode! 🌙

You asked it, I chatted with Facebook's [@alexvoica](#) about it and here's a video made in collaboration with Facebook!



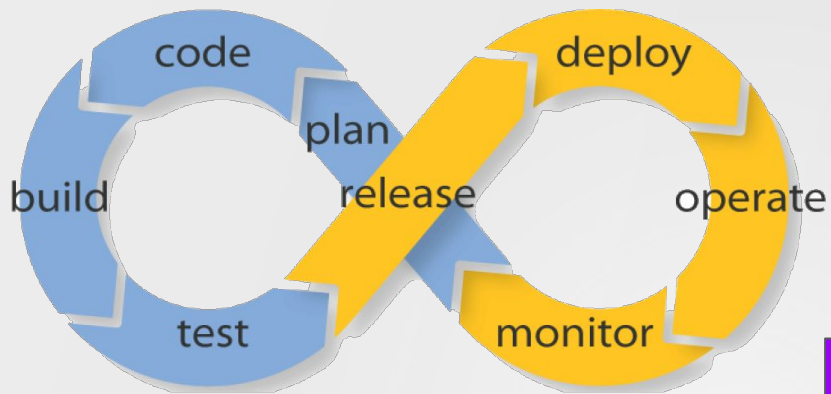
ivery

date Experience

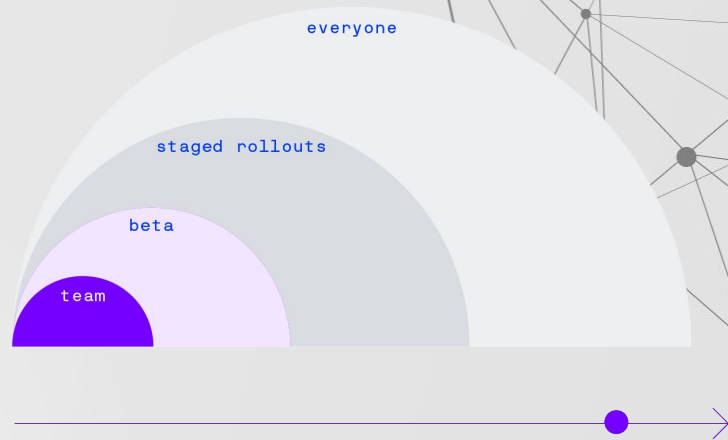
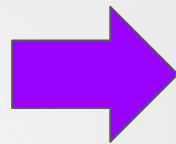


ow available. Staged rollouts provide a way to limit the updates to your customers in stages. This feature allows to 100% of the install base. Learn more about how to get

f your customers and increase that percentage before nes, it's better if only 1% of users experience this rather hat have historically yielded 10,000+ unique app installs r app in the Reporting metrics in the [Amazon Developer](#)



Continuous Delivery helps your team move faster by making each change small and manageable.



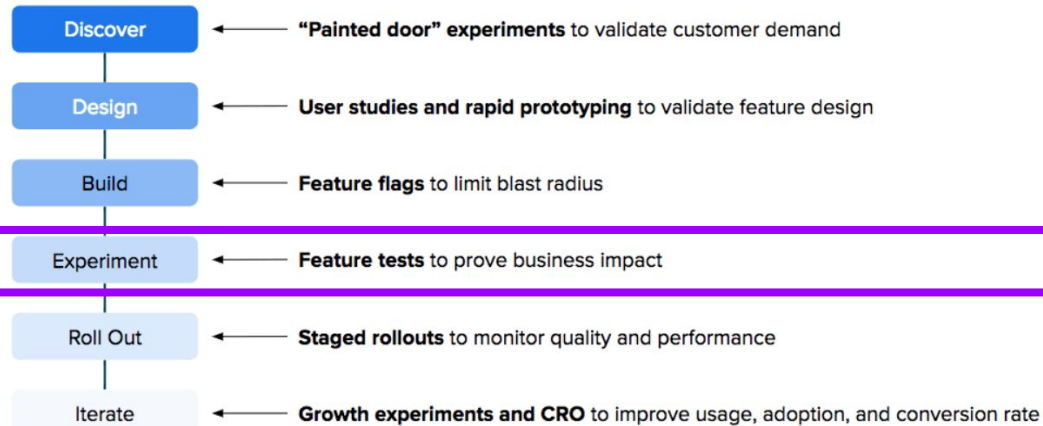
Progressive Delivery helps your team move faster by reducing the risk of each change by controlling the audience exposed.

Data Driven Software Development

Did I build the Product right?

Did I build the right Product?

Experiment-driven product development lifecycle



What happens when you don't
make good data driven decisions?

Snapchat's redesign cost it millions of users

By Ashley Carman | @ashleyrcarman | Aug 7, 2018, 4:44pm EDT



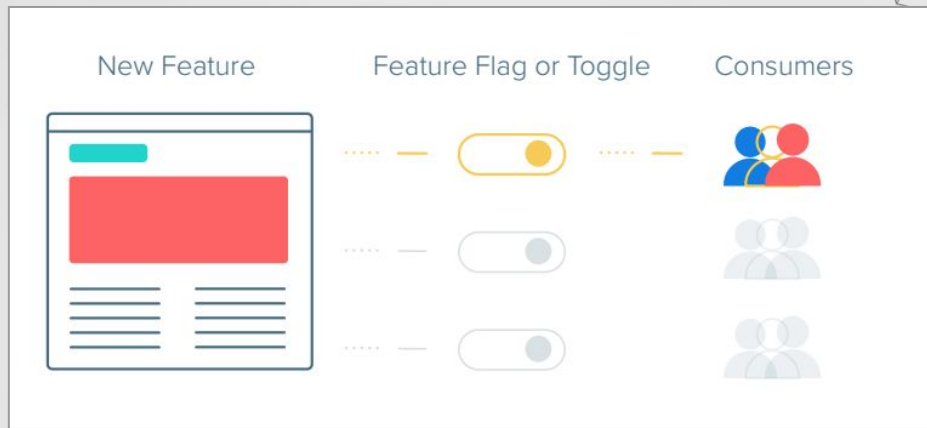
Listen to this article



SHARE



So... Feature Flags?

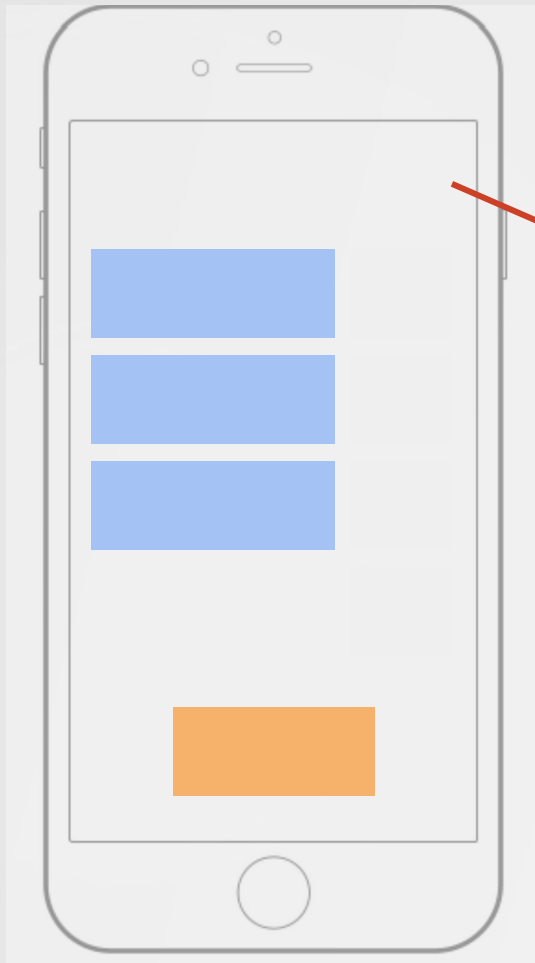


```
if feature_flag is 'on':  
    do_something()  
else:  
    do_something_else()
```



Feature Rollout

Disable header in
production, no code
change.



mobile_app_header

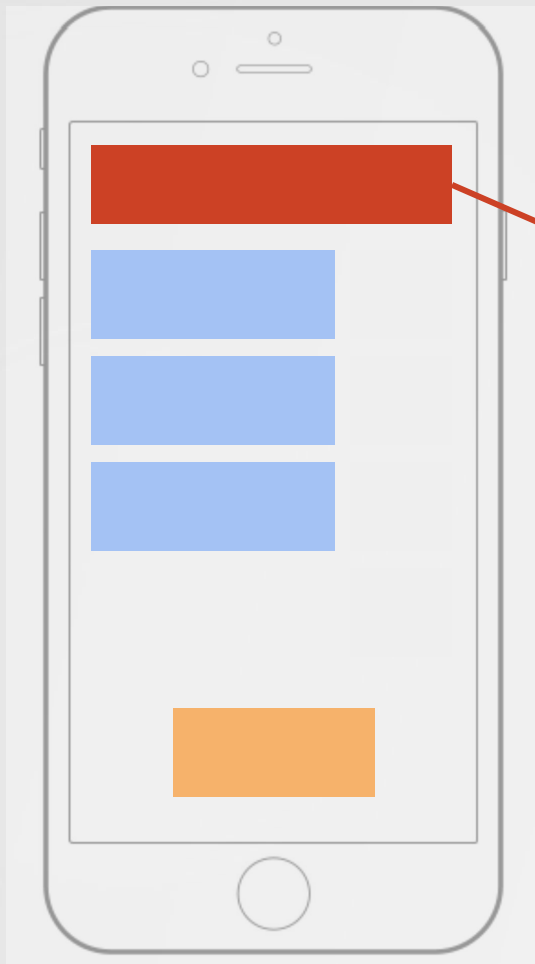
enabled: False

header_color: #F829EA
header_sticky: True
header_height: 20
header_message: 'We love
our users <3'



Feature Rollout

Slowly rollout new header to only certain customers.



mobile_app_header

enabled: True

header_color: #F829EA
header_sticky: True
header_height: 20
header_message: 'We love
our users <3'

Software Development Life Cycle



Build & Test

- Features don't need to be completed to get merged
- No wasted time untangling messy merge conflicts



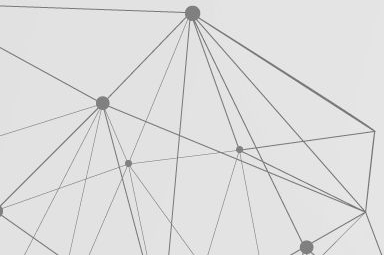
Deploy

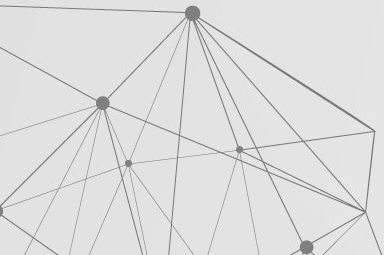
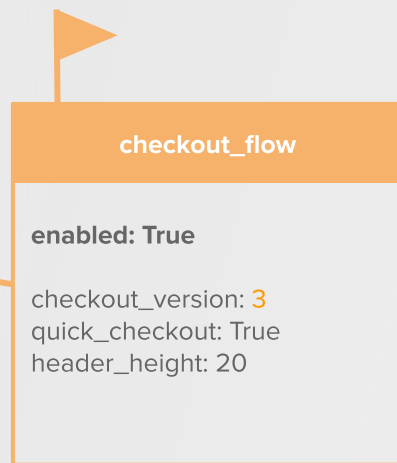
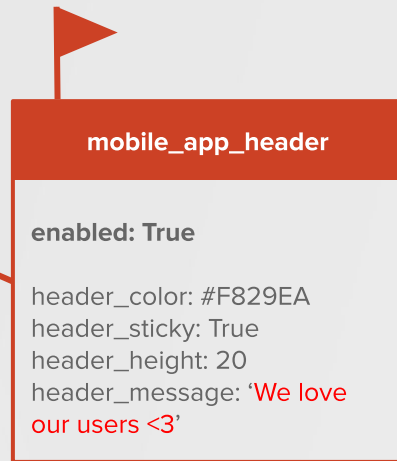
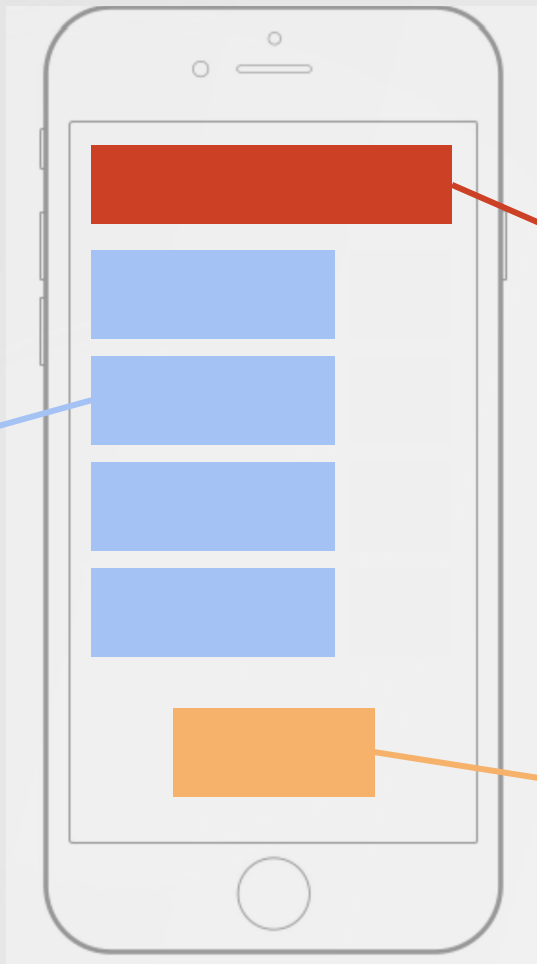
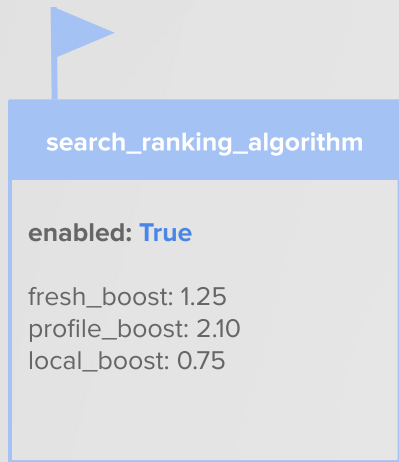
- Ship faster by decoupling deploys from releases to shorten dev cycle
- Deploy frequently and safely



Release

- Empower teams to manage their own releases
- Quickly rollback production incidents (way faster MTTR)







Quality Leadership, **Testing**, and **Governance Tactics** that Make or Break Your Progressive Delivery System

Why governance and testing?

Mission Control



Governance Tactics



Feature Flags

Damage Control



Testing



What could go wrong?

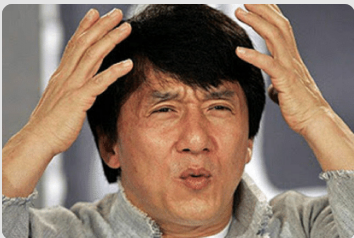
DZone > DevOps Zone > Feature Toggles are one of the Worst kinds of Technical Debt

Feature Toggles are one of the Worst kinds of Technical Debt

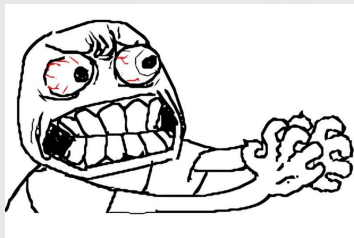
Technical debt is pretty bad, and feature toggles are some of the most horrible examples of technical debt.



by Jim Bird  MVB · Aug. 11, 14 · DevOps Zone · Opinion



What does this flag do?



Who turned off my flag?



Who touched my flag?

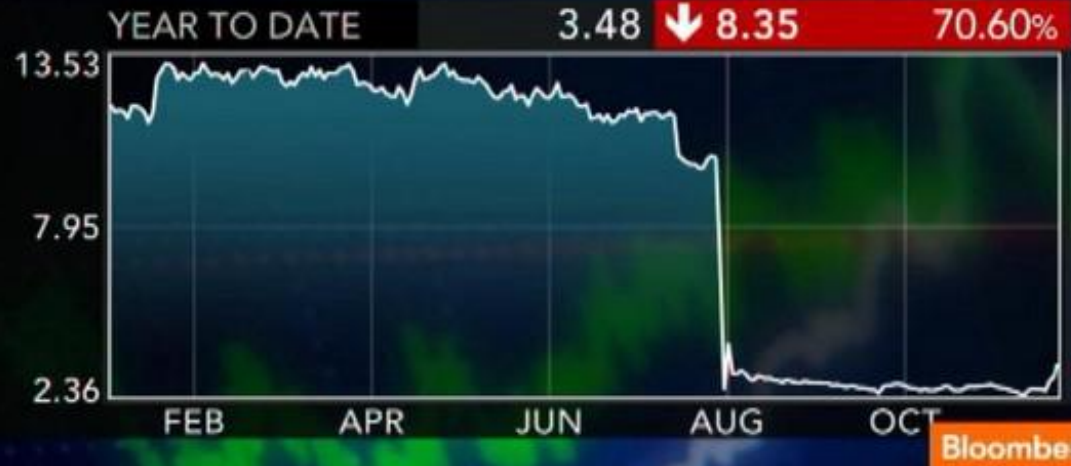
Horror Story Time!

Knight Capital Group

Knight

Type	Subsidiary
Traded as	NYSE: KCG , until July 1, 2013
Industry	Financial Services
Fate	Acquired by Getco LLC in 2013, forming KCG Holdings
Founded	1995
Headquarters	Jersey City, New Jersey , United States
Key people	Thomas Joyce, chairman and chief executive officer
Products	Market making and trading
Revenue	\$1.404 billion USD (2011)
Net income	\$115.2 million USD (2011)
Number of employees	1,418 (2012)
Website	www.knight.com

Knight Capital Group



Feature Flag Governance



Flag Info

General Info

- What does Flag do when it's On/Off
- What env is it deployed?

Visibility

- Who needs to know if this is On/Off?
- Who is monitoring metrics?



Access Level

Who is allowed to make changes to this feature flag?

How do we track this?



Risk Level

How dangerous is this to roll out?

What happens if something catastrophic happens to this flag?



Exit Strategy

When is this feature flag no longer useful?

When it is no longer useful, who is removing it?

Example: Governance

Flag Info

1. Feature On Behavior
2. Feature Off Behavior
3. Who actually owns this Feature Flag

[Feature Flag] changelog_publish

Attach Create subtask Link issue Attach Lucidchart Diagram

Description

Feature ON:

The changelog is published to the google pub/sub. `account_id` is used as an attribute for feature flag evaluation. Currently, the audience is set to `everyone` which means that changelog is published to every request which has an `account_id`. The feature flag is intended to be used as a kill switch.

Feature OFF:

The changelog is not published to the google pub/sub and is only logged.

Off Location (optional):

`...py#L209`

Risk to Roll Back Feature:

We'd lose the ability to stop publishing changelogs if something bad happens.

Exit Criteria:

1 month after Change history is GA

Expiration Date:

1 month after Change history is GA

(If you have questions on how to fill this out, go/how_to_ff)

Ready to be Removed

Feature Key changelog_publish

Risk to Roll Back Medium

Expiration Duration 30 days

Linked Experiment Name None

Creation Date 2019/04/04

Flag Type Feature Rollout

Assignee

Reporter

Epic Link CH 2.0 M7 - Change Hist...

Labels feature_flag_short_term

Automation ⚡ Rule executions

Slack 🗨 Open issue activity

Sentry 🐞 Linked Issues

Zendesk Support 🗨 Linked Tickets

Show 5 more fields

Components, Fix versions, Sprint, Development and Original Estimate

Created September 9, 2019, 2:37 PM

Updated August 19, 2020, 1:34 PM

Configure

Example: Governance

Risk Level

1. Can I roll back?
2. What happens if this flag fails?

Attach

Create subtask

Link issue

Attach Lucidchart Diagram

Description

Feature ON:

The changelog is published to the google pub/sub. `account_id` is used as an attribute for feature flag evaluation. Currently, the audience is set to `everyone` which means that changelog is published to every request which has an `account_id`. The feature flag is intended to be used as a kill switch.

Feature OFF:

The changelog is not published to the google pub/sub and is only logged.

GIT Location (optional):

Risk to Roll Back Feature:

We'd lose the ability to stop publishing changelogs if something bad happens.

Expiration Date:

1 month after Change history is GA

(If you have questions on how to fill this out, go/how_to_ff)

Ready to be Removed

Feature Key

changelog_publish

Risk to Roll Back

Medium

Expiration Duration

30 days

Linked Experiment Name

None

Creation Date

2019/04/04

Flag Type

Feature Rollout

Assignee

Reporter

Epic Link

CH 2.0 M7 - Change Hist...

Labels

feature_flag_short_term

Automation

Rule executions

Slack

Open issue activity

Sentry

Linked Issues

Zendesk Support

Linked Tickets

Show 5 more fields

Components, Fix versions, Sprint, Development and Original Estimate

Created September 9, 2019, 2:37 PM

Updated August 19, 2020, 1:34 PM

Configure

Example: Governance



EXIT

Exit Strategy

1. When should I remove this?
2. When can I remove this?
3. Who should be removing this?

[Feature Flag] changelog_publish

Attach Create subtask Link issue Attach Lucidchart Diagram

Description

Feature ON:

The changelog is published to the google pub/sub. `account_id` is used as an attribute for feature flag evaluation. Currently, the audience is set to `everyone` which means that changelog is published to every request which has an `account_id`. The feature flag is intended to be used as a kill switch.

Feature OFF:

The changelog is not published to the google pub/sub and is only logged.

GIT Location (optional):

`...py#L209`

Risk to Roll Back Feature:

We'd lose the ability to stop publishing changelogs if something bad happens.

Exit Criteria:

1 month after Change history is GA

Expiration Date:

1 month after Change history is GA

(If you have questions on how to fill this out, go/how_to_ff)

Ready to be Removed

Feature Key changelog_publish

Risk to Roll Back Medium

Expiration Duration 30 days

Linked Experiment Name None

Creation Date 2019/04/04

Flag Type Feature Rollout

Assignee

Reporter

Epic Link CH 2.0 M7 - Change Hist...

Labels feature_flag_short_term

Automation Rule executions

Slack Open issue activity

Sentry Linked Issues

Zendesk Support Linked Tickets

Show 5 more fields

Components, Fix versions, Sprint, Development and Original Estimate

Created September 9, 2019, 2:37 PM

Updated August 19, 2020, 1:34 PM

Configure

Testing Strategy

**Manual
Testing**

Test Type

High Level Strategy

Manual Test

Only most critical variations.

E2E Test

Focus only on important variations and test application still works if all features are on/off

UI Tests

Integration Test

Mock and stub to control feature states. Focus on individual code paths to ensure proper integration and business logic

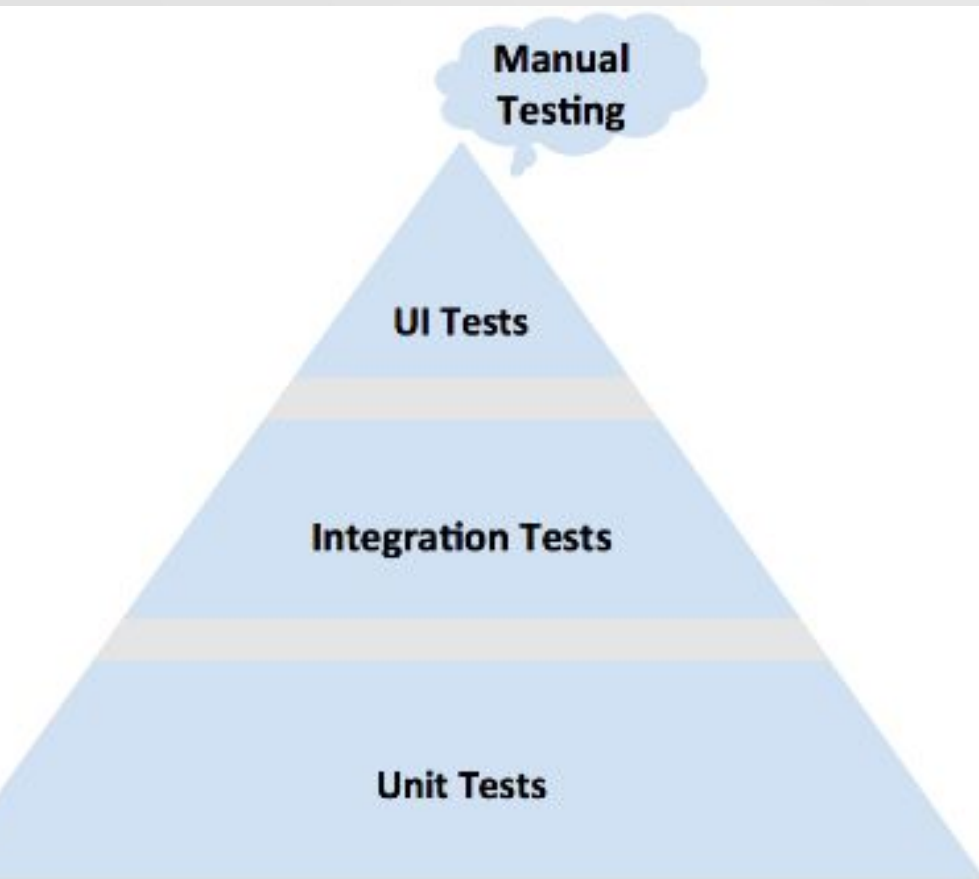
Integration Tests

Unit Test

Each code path should have its own set of independent unit tests. Ensure high code coverage, just as if you didn't have any feature flags in your codebase.

Unit Tests

Testing Strategy



Test Type	High Level Strategy
Manual Test	Only most critical variations.
E2E Test	Focus only on important variations and test application still works if all features are on/off
Integration Test	Mock and stub to control feature states. Focus on individual code paths to ensure proper integration and business logic
Unit Test	Each code path should have its own set of independent unit tests. Ensure high code coverage, just as if you didn't have any feature flags in your codebase.

General E2E Automation Test Strategy

Test Runner + INDETERMINISM =



General E2E Automation Test Strategy

Remove INDETERMINISM

1. Special Test User
2. Test Cookie
3. Test Query Parameter to force a particular version





General E2E Automation Test Strategy

```
7   cy.get('#astronaut')
8     .should('not.exist')
9   })
10
11  it('Validate that the astro_boy feature is not enabled with incorrect query parameter', function () {
12    // Visit app
13    cy.visit('/?cypress=off')
14
15    // Validate feature is disabled
16    cy.get('#astronaut')
17      .should('not.exist')
18  })
19
20  it('Validate that the astro_boy feature is enabled', function () {
21    // Visit app with audience query parameter
22    cy.visit('/?cypress=on')
23
24    // Validate feature is enabled
25    cy.get('#astronaut')
26      .should('exist')
27  })
28
29  it('Validate button is clickable', function () {
30    // Visit app with audience query parameter
31    cy.visit('/?cypress=on')
32
33    // Validate feature is enabled
34    cy.get('button')
35      .click()
```

WHAT IF I TOLD YOU

**THERE ARE MORE THAN ONE TYPE
OF FEATURE FLAGS AND YOU NEED
TO TEST/GOVERN THEM ALL DIFFERENTLY?**

Feature Flag Types



Feature Rollout

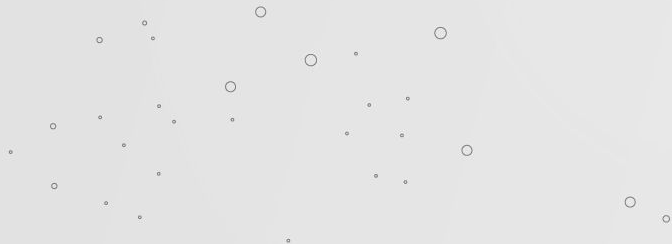
For deploying a new feature that will be permanent.



Experimentation

Perform A/B/n tests against our deployed features.

Short Term

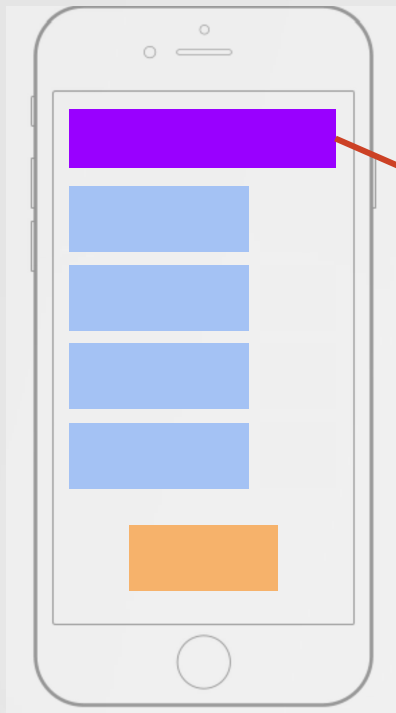




Experimentation

Does header "We love our users <3" drive more conversions vs "Hi, welcome?"

50%



mobile_app_header

enabled: True

header_color: #F829EA

header_sticky: True

header_height: 20

header_message: 'We love our users <3'

mobile_app_header

enabled: True

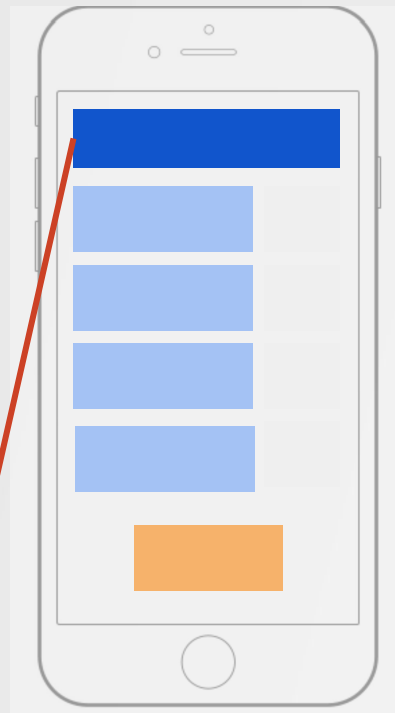
header_color: #F829EA

header_sticky: True

header_height: 20

header_message: 'Hi, welcome?'

50%



Feature Flag Types



Feature Rollout

For deploying a new feature that will be permanent.



Experimentation

Perform A/B/n tests against our deployed features.

Short Term



Long Term

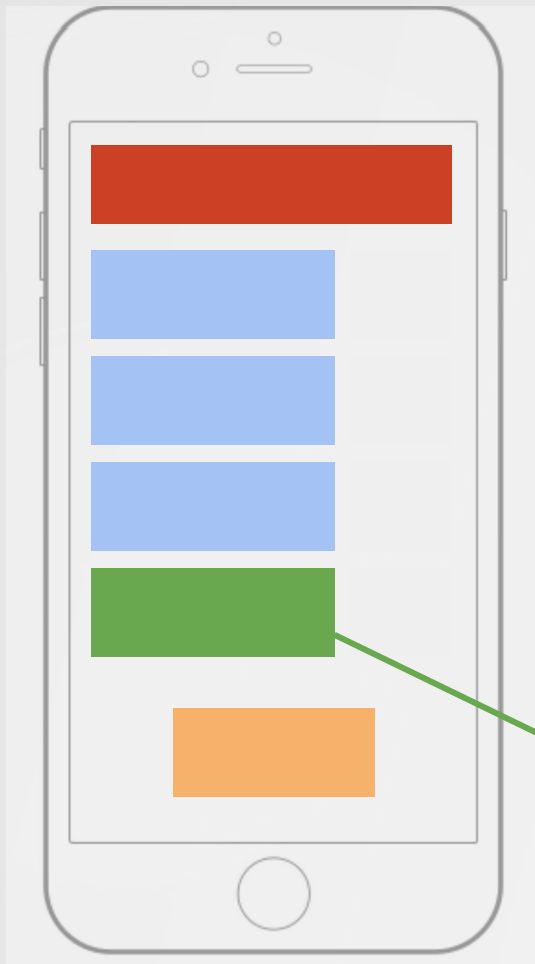
Permission

Change product experience that certain users receive



Permissions

Is user_tier type premium? If so, allow them to access the premium content.



premium_view

enabled: True

user_tier: premium
header_height: 20

Feature Flag Types



Feature Rollout

For deploying a new feature that will be permanent.



Experimentation

Perform A/B/n tests against our deployed features.

Short Term

Long Term



Permission

Change product experience that certain users receive



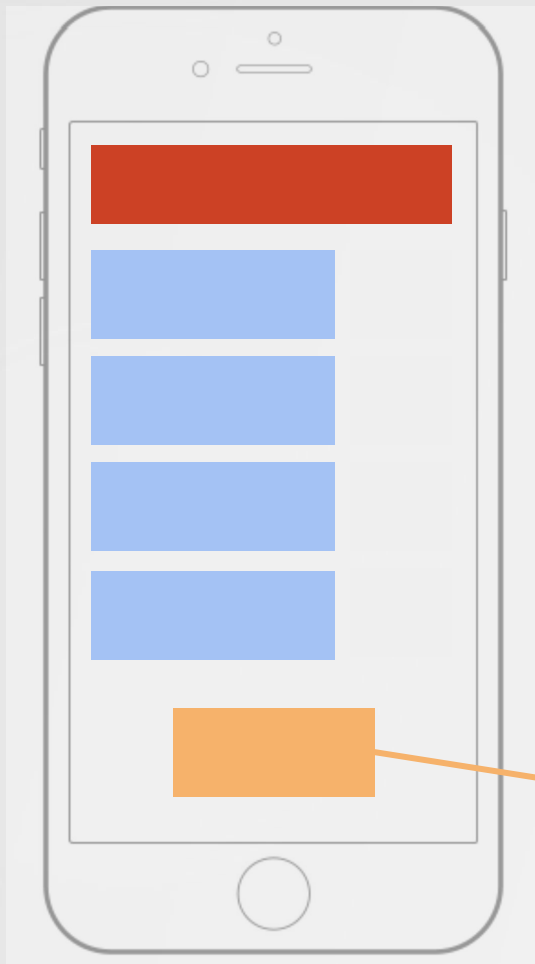
Operation

Controls operational aspects of your system; allows control of certain operational features in your production environment



Operations

Change the
checkout_flow
dynamically without
having to deploy.



checkout_flow

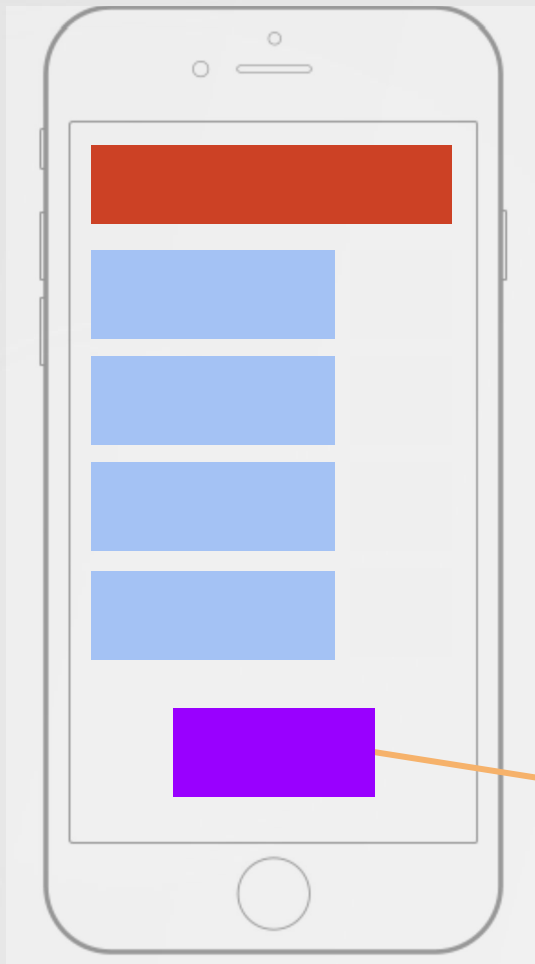
enabled: True

checkout_version: 3
quick_checkout: True
header_height: 20



Operations

Change the
checkout_flow
dynamically without
having to deploy.



checkout_flow

enabled: True

checkout_version: **4**
quick_checkout: True
header_height: 20

Feature Flag Types



Feature Rollout

For deploying a new feature that will be permanent.



Experimentation

Perform A/B/n tests against our deployed features.

Short Term

Long Term



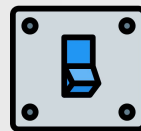
Permission

Change product experience that certain users receive




Operation

Controls operational aspects of your system; allows control of certain operational features in your production environment



Circuit Breaker

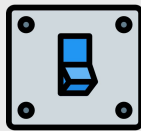
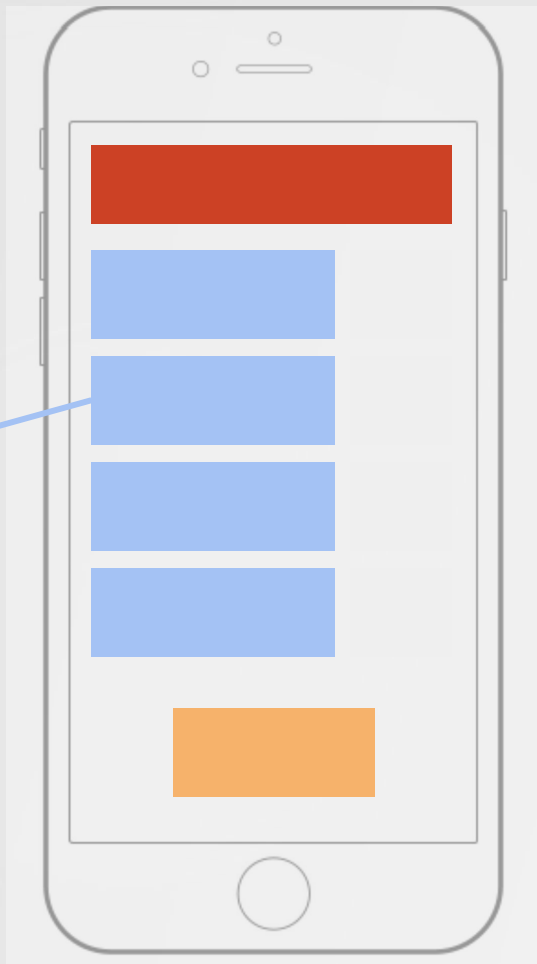
Turn on/off certain features or conditions in your production environment.



search_ranking_algorithm

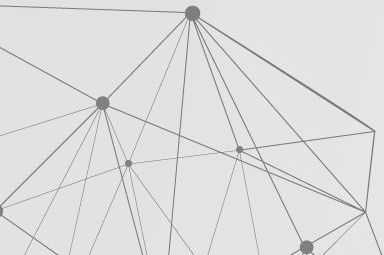
enabled: **True**

fresh_boost: 1.25
profile_boost: 2.10
local_boost: 0.75



Circuit Breaker

Turn on/off
search_ranking_alogrithm.



Same but Very Different...

Mission Control

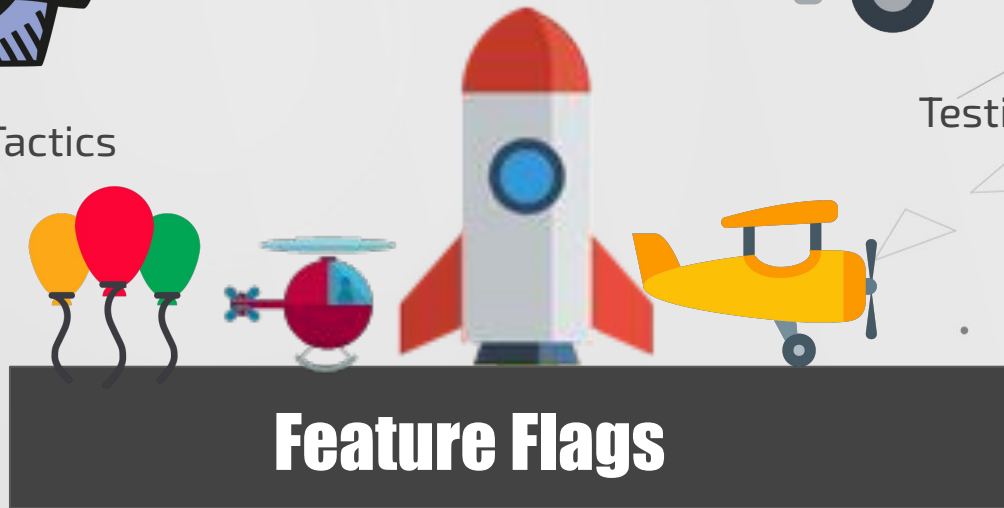


Governance Tactics

Damage Control



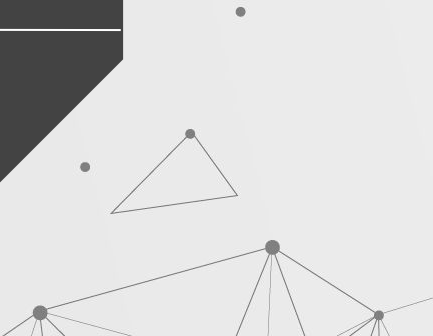
Testing



Feature Flags

Feature Flag Governance

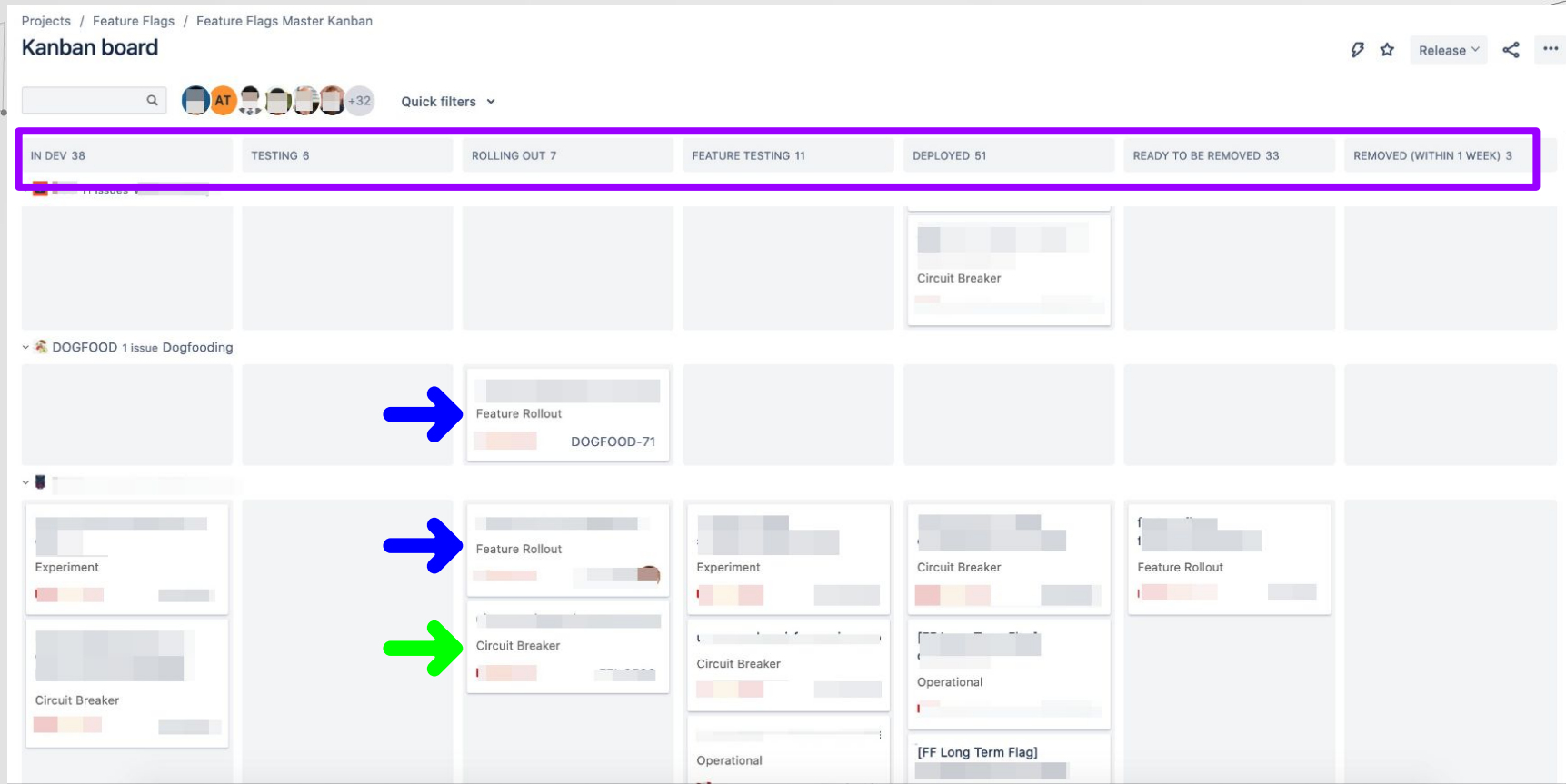
	Flag Info	Access Level	Risk Level	Exit Strategy
Experiment	Hypothesis driven	Product/Eng	Low	Experiment End



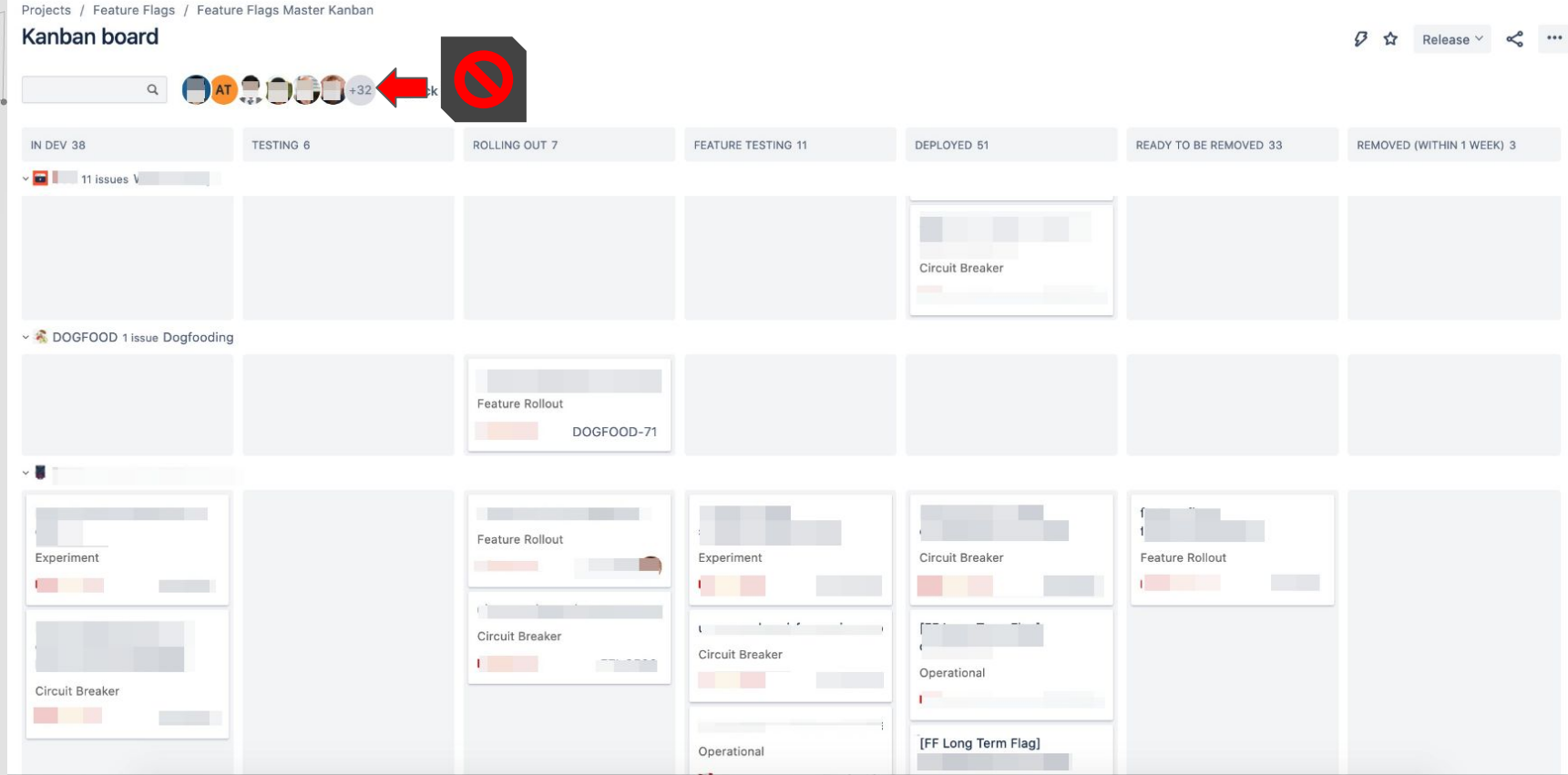
Feature Flag Governance

	Flag Info	Access Level	Risk Level	Exit Strategy
Experiment	Hypothesis driven	Product/Eng	Low	Experiment End
Permission	Business Tier Provisioning Info	CS/AE/GTM Restrictive Access	High	Rare
Operational	Eng Runbook	Eng	High/Medium	If Feature Deprecates
Circuit Breaker	Eng Runbook	Eng	Low	If Feature Deprecates

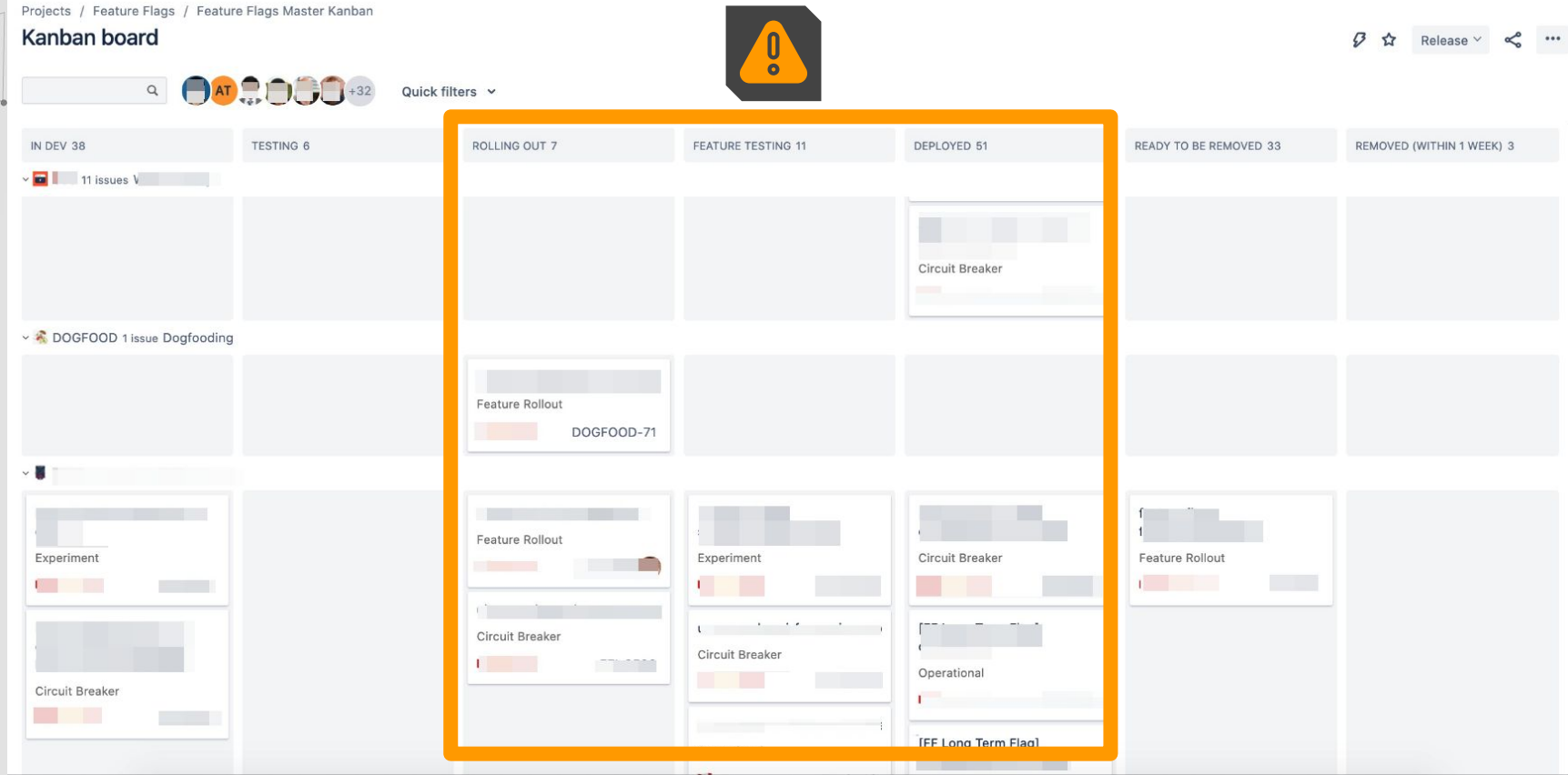
Example: Governance (Flag Info)



Example: Governance (Access)



Example: Governance (Risk)







Example: Governance (Removal)

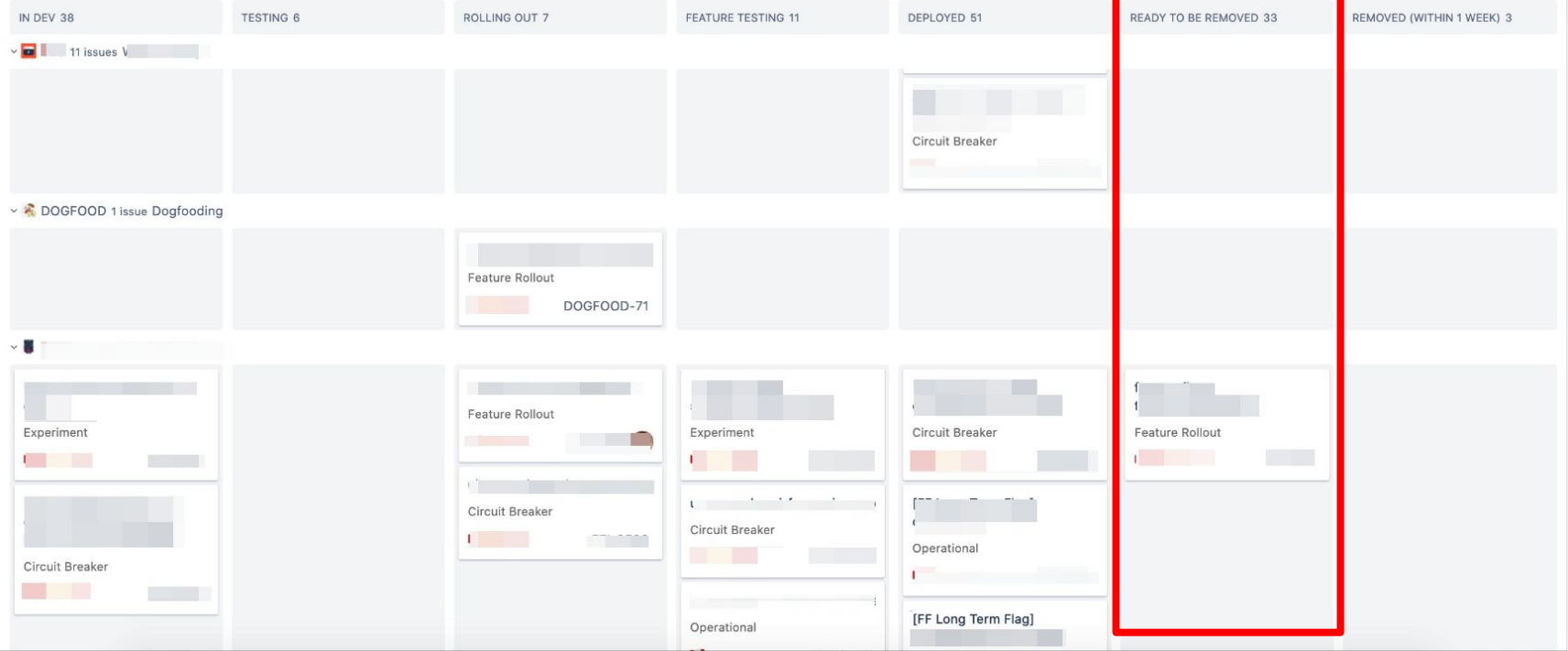
Projects / Feature Flags / Feature Flags Master Kanban

Kanban board

 Quick filters ▾



  Release ▾  



Permission: E2E Test Strategy



Permissions

1. Checking that "provisioning" is correct
2. Automation should check all paths
3. High sensitivity in regression that we don't remove a customers features

```
1 describe('Rollouts Experiments Dashboard Tests', function
2   before(function () {
3     cy.createAccount(Cypress.env('free_rollout_account'))
4     .saveGeneratedTimeBasedIds('featureKey')
5     .saveGeneratedTimeBasedIds('featureKey1')
6     .saveGeneratedTimeBasedIds('experimentKey')
7     .saveGeneratedTimeBasedIds('experimentKey1')
8   })
```

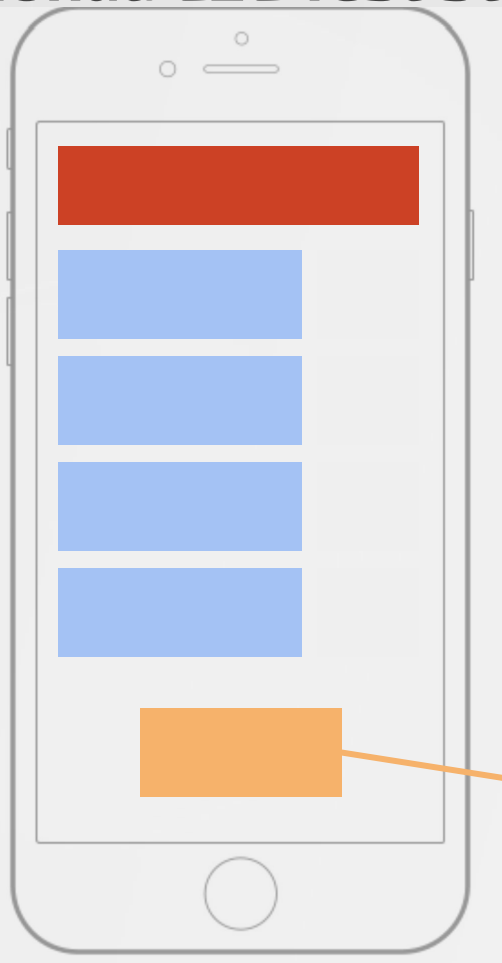
```
1 describe('Full Stack Permissions Experiments Dashboard Tests', fu
2   before(function () {
3     cy.createAccount(Cypress.env('full_stack_account'))
4     .saveGeneratedTimeBasedIds('featureKey')
5     .saveGeneratedTimeBasedIds('featureKey1')
6     .saveGeneratedTimeBasedIds('experimentKey')
7     .saveGeneratedTimeBasedIds('experimentKey1')
8   })
```

Operational: E2E Test Strategy



Operational

1. Validate “default” condition
2. Strategy to validate operational values may vary...
3. Doesn't always make sense to automate all paths...

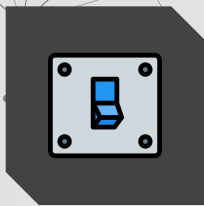


checkout_flow

enabled: True

checkout_version: 3
quick_checkout: True
header_height: 20

Circuit Breaker: E2E Test Strategy



▶ Circuit Breaker

1. Regression Automation:
Product still works if flag
is ON, and still works if
OFF
2. 3rd Party Integrations -
usually manually tested

```
self.template_values['enable_segment_snippet'] =  
[...].is_feature_enabled('enable_segment_snippet', (...)(_user_id))
```

POWERING MODERN DEPLOYMENT

Feature Flags used correctly are an incredibly powerful engineering tool.

01

CONTROLLED BLAST RADIUS

Quality champions rejoice! Plus our MTTR should be factors better!

02

THOUGHTFUL IMPLEMENTATION

Ensure your organization has a strong process for releasing/managing/removing your feature flag, or else technical debt can punish you.

03

TAKEAWAYS

04

FEATURE FLAG DIFFERENTIATION

Not all Feature Flags are alike, why would you treat them so?

05

DETERMINISTIC TESTING

Ensure you are properly testing your feature flags and that you have good feature coverage for each path.

06

QUALITY LEADERSHIP

Every organization needs leadership around governance and testing.

RESOURCES

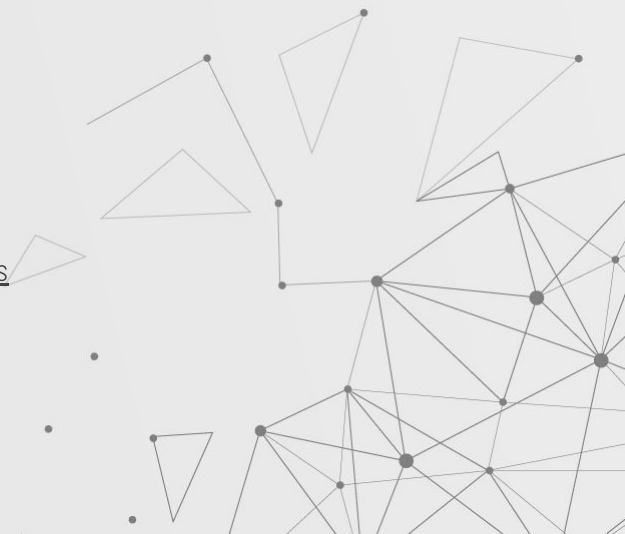
Interested in learning more about Feature Flags? Or trying them?

VENDORS

- Optimizely (Free Rollouts): <https://www.optimizely.com/rollouts/>
- LaunchDarkly: <https://launchdarkly.com/>
- Split: <https://split.io/>

FURTHER READING

- Feature Toggles: <https://martinfowler.com/articles/feature-toggles.html>
- Getting Started With Feature Flags: <https://dzone.com/refcardz/getting-started-with-feature-flags>
- Unleash (open source Feature Flag): <https://github.com/Unleash/unleash>



Stay in Touch



jeff.sing@iterable.com



<https://www.linkedin.com/in/jeffsing/>

CREDITS: This presentation template was created by **Slidesgo**, including icons by **Flaticon**, and infographics & images by **Freepik**.

Please keep this slide for attribution.