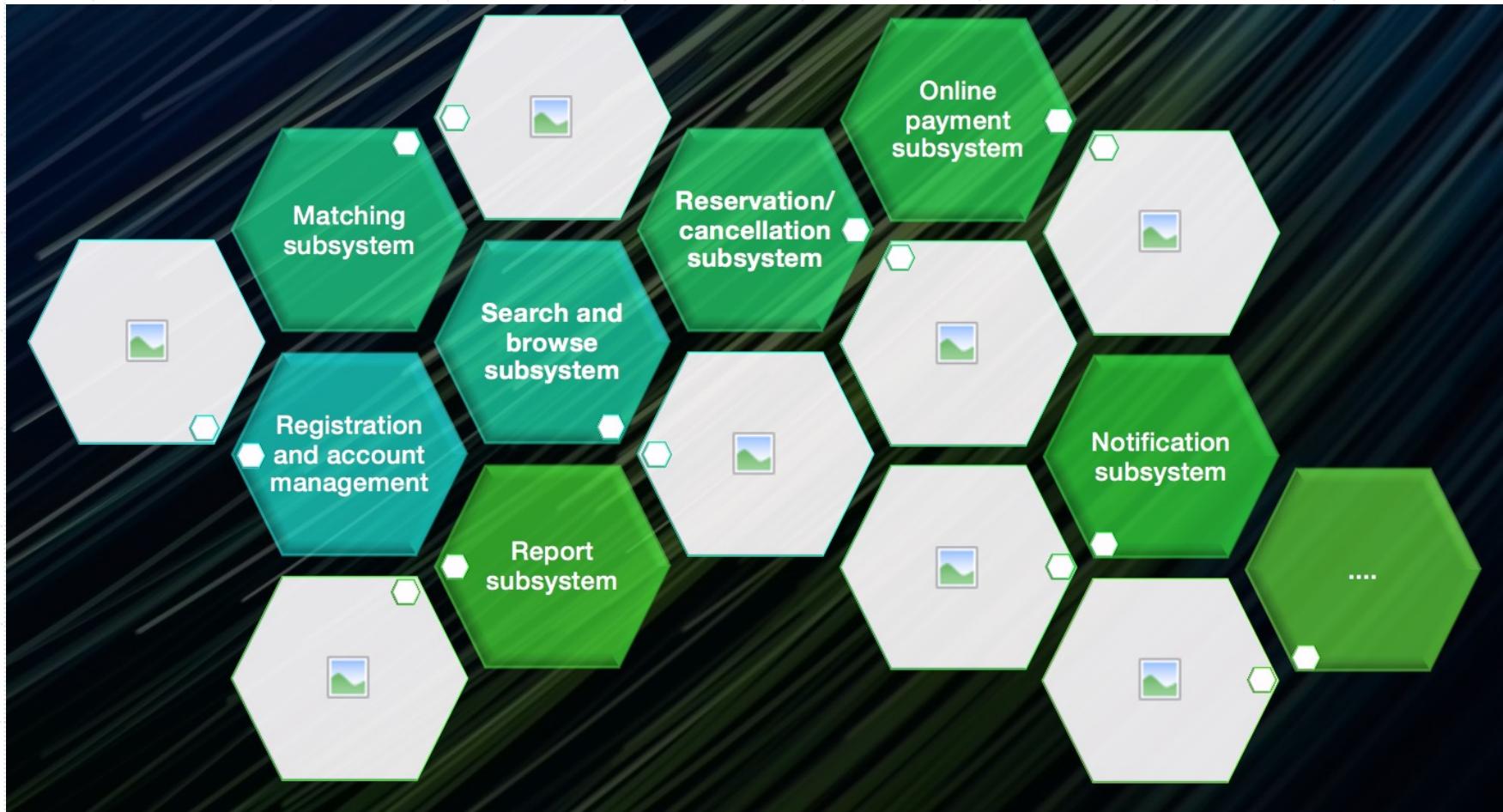


Continuous Integration (CI) and Continuous Delivery (CD)

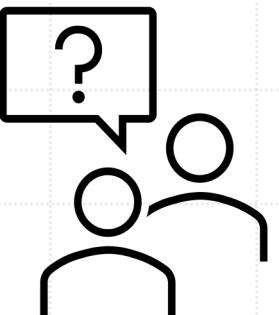
Software Engineering II
2566/2



What are subsystems within your term project?



What else?



How are you developing term project

The image shows a Trello Agile Sprint Board titled "Agile Sprint Board". The board is divided into several columns: "Story", "Completed Story", "Tumor Board Sprint Backlog", "Sprint 8 (29/01 - 11/02)", and "In progress". Each column contains cards representing tasks or stories. The "Story" and "Completed Story" columns have cards for doctors wanting to upload VCF files, see patient medical possibility, view evidence details, discuss with associated doctors, and submit treatment decisions. The "Tumor Board Sprint Backlog" and "Sprint 8" columns contain cards for users querying evidence data via API endpoints, adding patient data, having infrastructure setup correctly, getting up-to-date data from OncoKB, having CRUD API endpoints for OncoKB, handling cases for fatal server-side errors, and managing CORS. The "In progress" column has cards for reading maf files and editing patients on the server-side. A large orange callout bubble points from the bottom right towards the center of the board, containing the text: "Trello, Slack, Notion, Jira, Asana, ClickUp, etc."

<https://trello.com/c/KG8bF8wV/73-story07-as-a-doctor-i-want-to-see-my-patient-medical-possibility>

<https://trello.com/b/HijFnL6B/agile-sprint-board>

Which tools are you using?



Which coding tools are you using?



How do you enable version control?



How do you build your system?



How do you test your system?



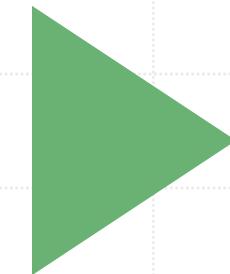
How do you deploy your system?

Which tools are your friends using in term project?

ขอให้แต่ละกลุ่ม ใส่ข้อมูล ในคอลัมน์นี้ต่อไปนี้
ก่อนวันที่ 28 กพ 66 โดยสามารถเขียนแบบ free text

Section		Group	multirepo	monorepo	DB	Dockerfile	docker-compose	Deployment environment เช่น Vercel	Git workflow	CI/CD tools เช่น Argo NGINX	Swagger	Automated tests:
			yes/no	yes/no	Firebase, Mong	yes/no	yes/no	AWS, GCP, Azure, Huawei, Vercel	yes/no	ArgoCD/etc...	yes/no	yes/no
33	1	No	Yes	Yes	postgresql	Yes	Yes	VPS Server	Yes	Github Actions	No	Yes
	2	No	Yes	Yes	postgreSQL, S3	No	Yes	Railway	Yes	Github Actions	No	Not yet
	3	No	Yes	Yes	MongoDB	Yes	Yes	AWS	Yes		No	Not yet
	4	Yes	No	No	MongoDB	Yes	Yes		Yes		-	-
1	5	No	Yes	Yes	MongoDB	No	No	Vercel	Yes	Github Actions	No	Yes
	6	-	-	-		-	-		-		-	-
	7	No	Yes	Yes	MongoDB	Yes	Yes	AWS	Yes	Github Actions	No	Not yet
	8	-	-	-		-	-		-		-	-
2	9	No	Yes	Yes	postgreSQL, S3	Yes	Yes	DigitalOcean	Yes	Github Actions	No	Yes
	10	-	-	-		-	-		-		-	-
	11	Yes	No	No	MongoDB	Yes	Yes	Vercel (Frontend) , AWS Lightsail (Backend), Cloudflare	Yes	Github Actions	No	Yes
	12	-	-	-		-	-		-		-	-
3	13	No	Yes	Yes	postgreSQL	Not yet	Not yet	Vercel	Yes		No	Yes
	14	Yes	No	No	postgreSQL	-	Yes		Yes	Github Action	No	Yes
	15	(CU Freelance)	Yes	No								
					- PostgreSQL - MongoDB - Redis - Opensearch							
								GCP, AWS				

Example from last year: https://docs.google.com/spreadsheets/d/12gBK3i5IDL_zZYyFzq8V4PE6P751HsGXaLwBwMkh4Gw/edit#gid=0
https://docs.google.com/spreadsheets/d/19_iZA43x5D-hH7nchlGY3tfAaFOOnM3Psfvh5-Ktx8U/edit#gid=0



HOW CAN YOU
CONTINUOUSLY
INTEGRATE YOUR
SUBSYSTEMS AND
DELIVER FOR
EACH SPRINT
REVIEW?

HOW CAN YOU
CONTINUOUSLY
INTEGRATE YOUR
BUG FIXES / NEW
FUNCTIONALITY
TO YOUR
LAUNCHED
PRODUCT?

How can you deliver a new feature to the production quickly?



Outlines

- Continuous Integration (CI) and Continuous Delivery (CD)
- DevOps
- GitHub Actions (as example tool for CI/CD)

This process in conjunction with other philosophies and software development methods that were gaining traction in 1990s, later formed the [Agile Manifesto](#).

Brief history

- 1960s-1980s [Waterfall model](#)
- Early 1990s [Extreme Programming \(XP\)](#), centered of [test-driven development \(TDD\)](#), which helped shape what it would be for today CI.
 - Announces **automated processes** to frequently integrate all code across developing teams, with the intention of delivering code that could be sent to the production at any given time.
 - Resulted in many builds per day.
 - Helps improve the predictability and efficiency of the software development life cycle.
 - Allows constant interactions with users with ever-changing needs.



Agile

- Agile encompasses many practices that are
 - Lightweight
 - Code- and people-oriented
 - Highly adaptive and built frequently

Based on these premises, CI and CD have gained more space among software developers across the globe.



Main idea behind CI (Continuous Integration)

- **Automated processes** should be in place to test and build software many times a day.
- This is important because **bugs are commonly introduced at the intersection and integration of two difference pieces of codes**.
- CI practices manage processes, which **allows software engineers to focus on the code itself**.



CD (Continuous Delivery)

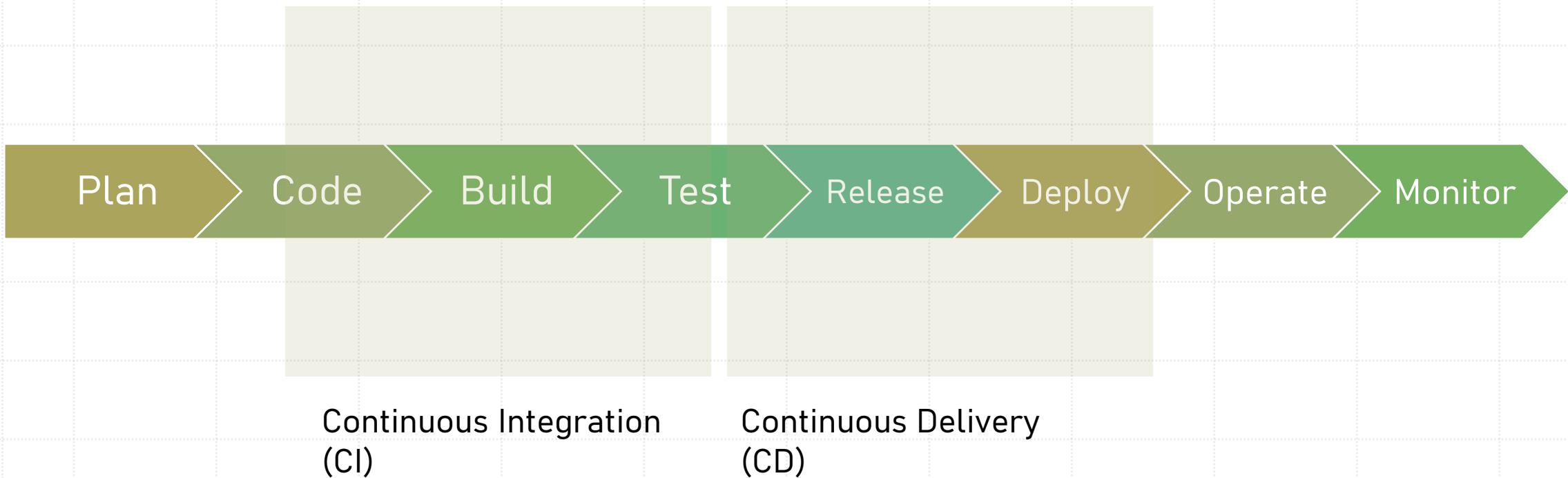
- CD focuses on **gathering all changes to code – new features, bug fixes, and configuration changes** – and sending them to users – or production safely, sustainability, and as quickly as possible.



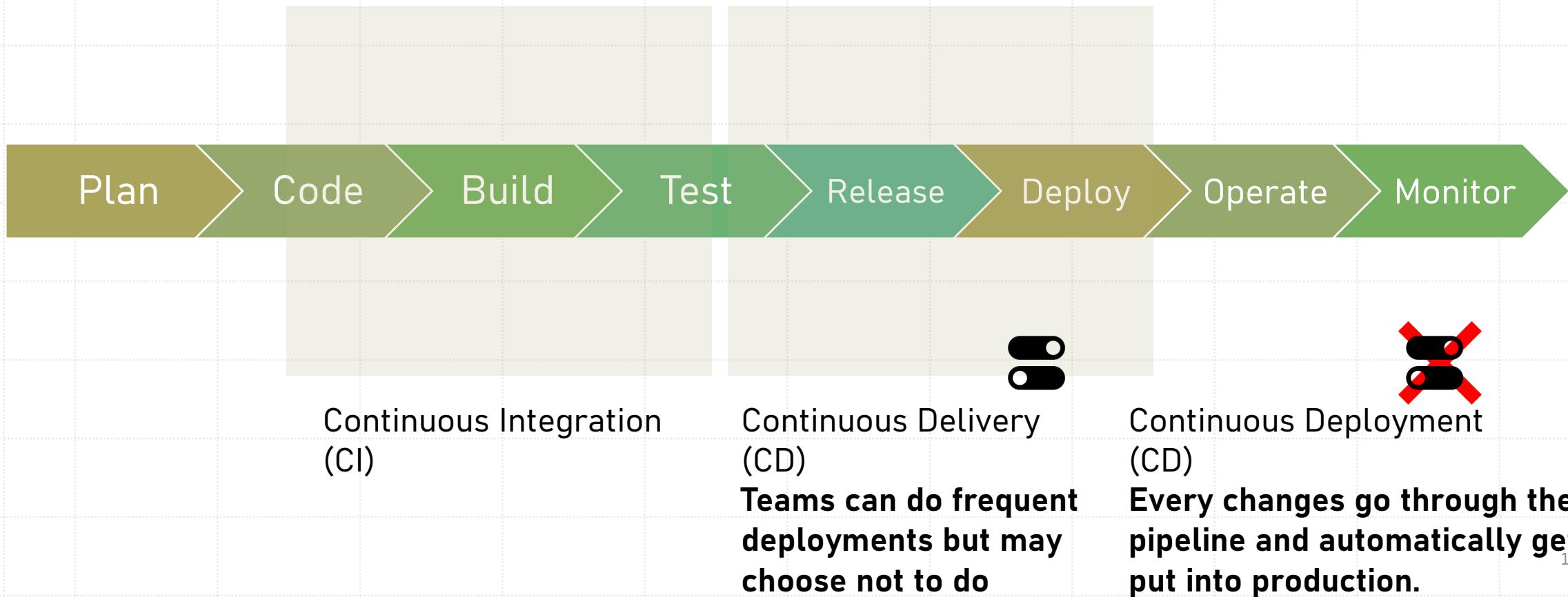
CI : Merge Code In

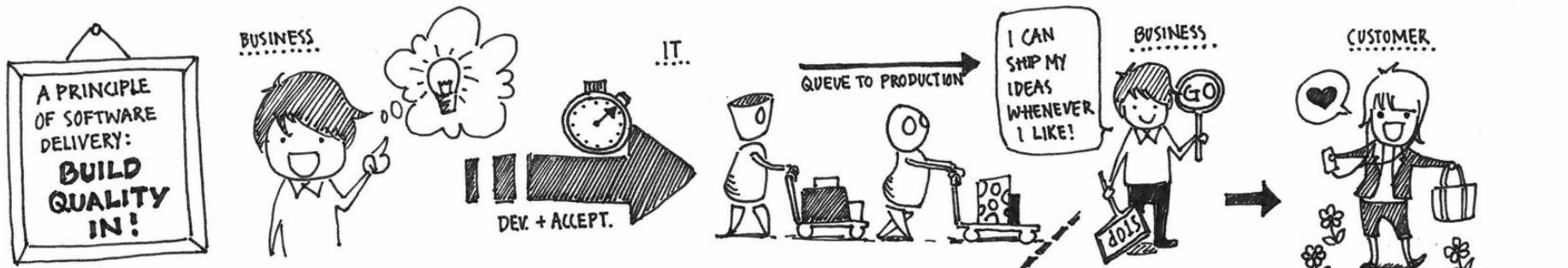
CD : Release Code Out

CI/CD



Differences between Continuous Delivery and Continuous Deployment





CONTINUOUS DELIVERY

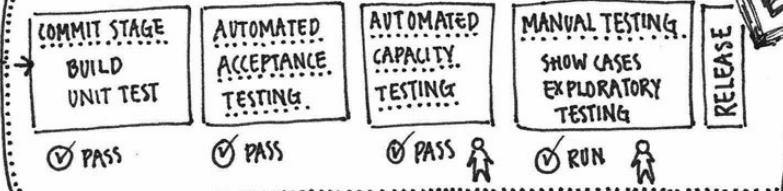
BY JEZ HUMBLE & DAVID FARLEY



A CLOSER LOOK

- COMMIT STAGE
 - CREATING EXECUTABLE CODE MUST WORK.
 - VERIFIES THAT THE SYNTAX OF YOUR SOURCE CODE IS VALID
- UNIT TEST PASS
- FULFILL CERTAIN QUALITY CRITERIA SUCH AS TEST COVERAGE AND OTHER TECHNOLOGY-SPECIFIC METRICS

KEY pattern DEPLOYMENT PIPELINE



EXAMPLE

DONE MEANS RELEASED
CHANGE

CHANGE IN

- EXECUTABLE CODE
- CONFIGURATION
- HOST ENVIRONMENT
- DATA

CHANGED PIPELINE 1
CHANGED PIPELINE 2
CHANGED PIPELINE 3

CREATE NEW INSTANCE OF PIPELINE

FEEDBACK

• ANY CHANGE IS A TRIGGER • FAST • ACT ON IT

BENEFITS



EMPOWERED - IN CONTROL
LOW STRESS - SMALL RELEASES

REDUCING ERRORS
- CONFIG MGT.
- VERSION CONTROL

DEPLOYMENT FLEXIBILITY
- EASY TO START APPLICATION IN NEW ENVIRONMENT

PRACTICE MAKES Perfect

SEEMS LIKE THE AUTHORS CAN'T STRESS IT ENOUGH. IT'S EVERYWHERE THROUGHOUT THIS BOOK.

AUTOMATE ALMOST EVERYTHING

VERSION CONTROL

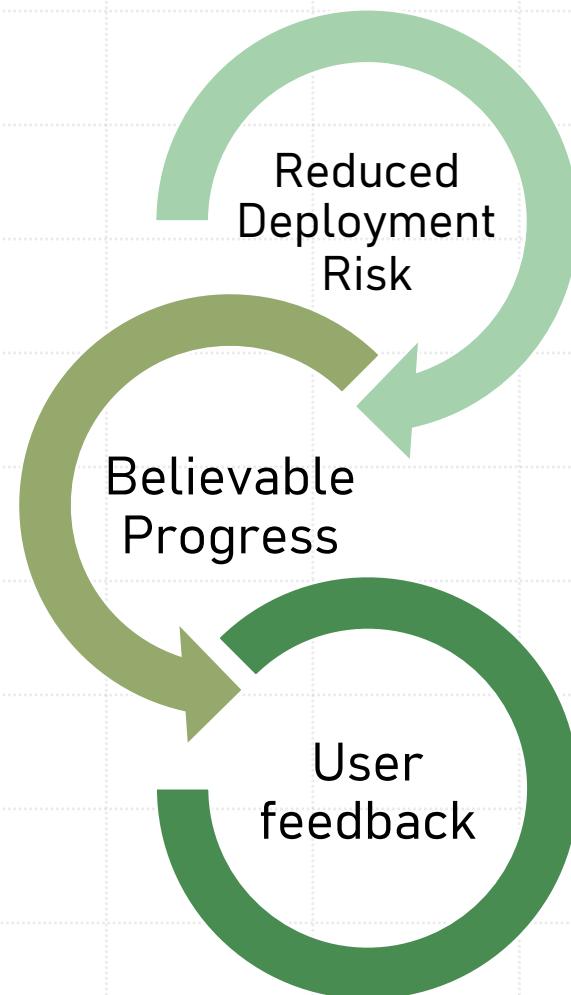
“
ENCOURAGING GREATER COLLABORATION
BETWEEN EVERYONE INVOLVED IN
SOFTWARE DELIVERY IN ORDER
TO RELEASE VALUABLE
SOFTWARE FASTER
AND MORE RELIABLY.
”

If it hurts, do it more frequently

(CC) BY-SA

Nhan Ngo

Principle benefits of continuous delivery



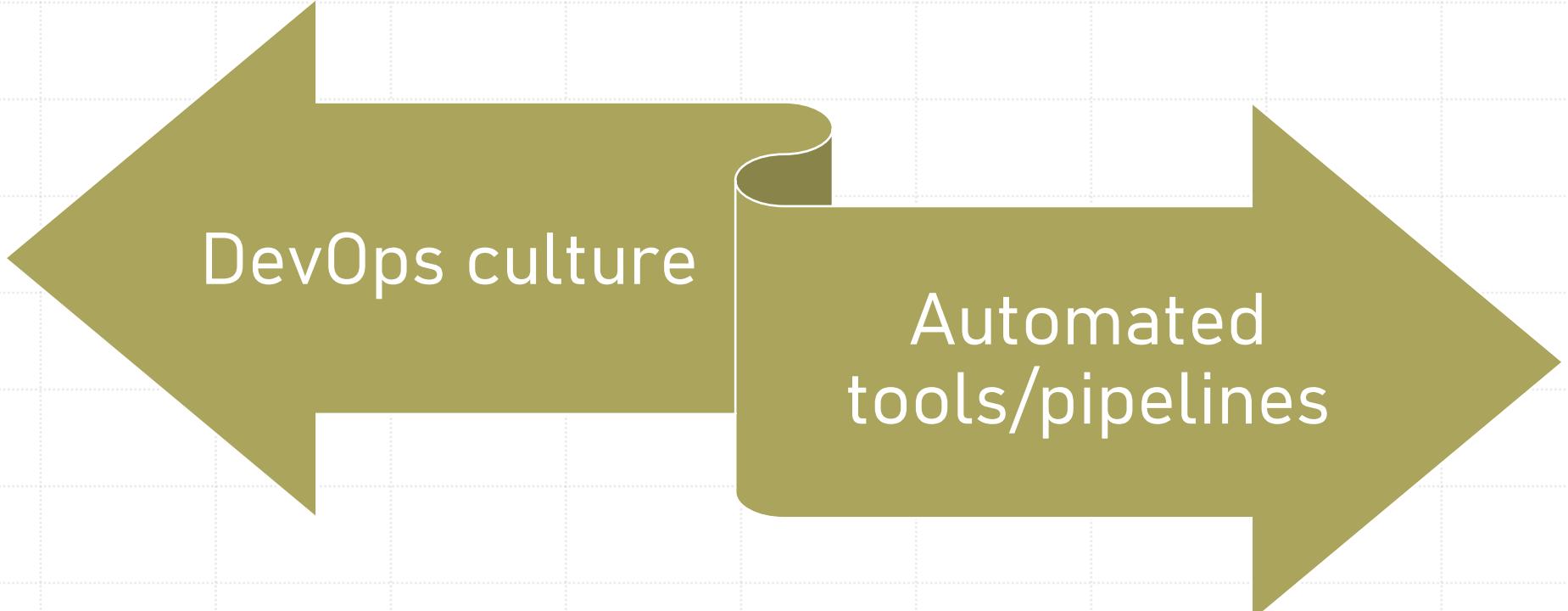


CI/CD can be both powerful and challenging

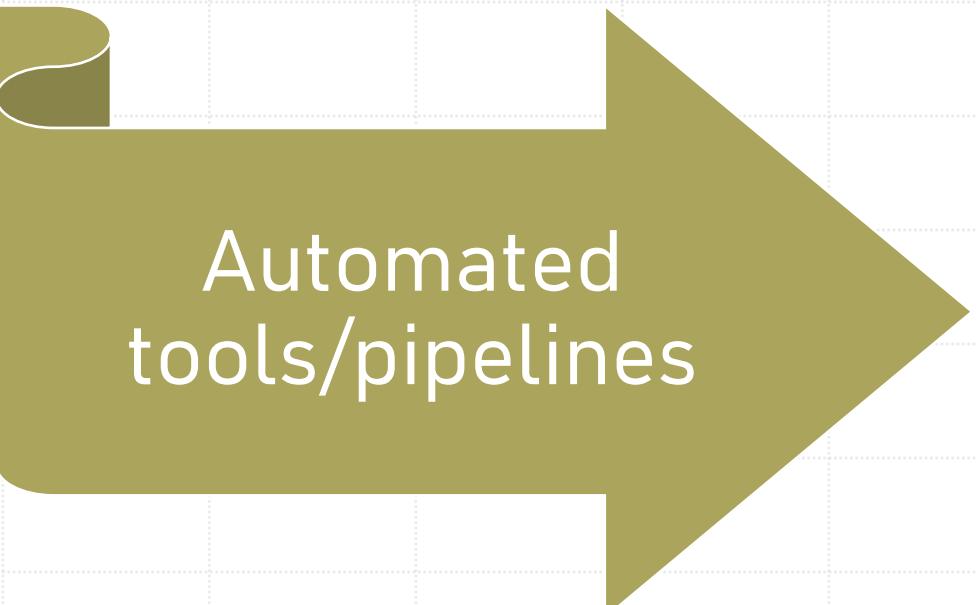
- Shift in culture that must accompany the adoption of these practices.
- When combined and adopted across a company, they can help ensure high quality, lightweight, and adaptive software development.



How to achieve continuous delivery



DevOps culture



Automated
tools/pipelines

What is DevOps?

The diagram illustrates the DevOps concept through four interconnected components:

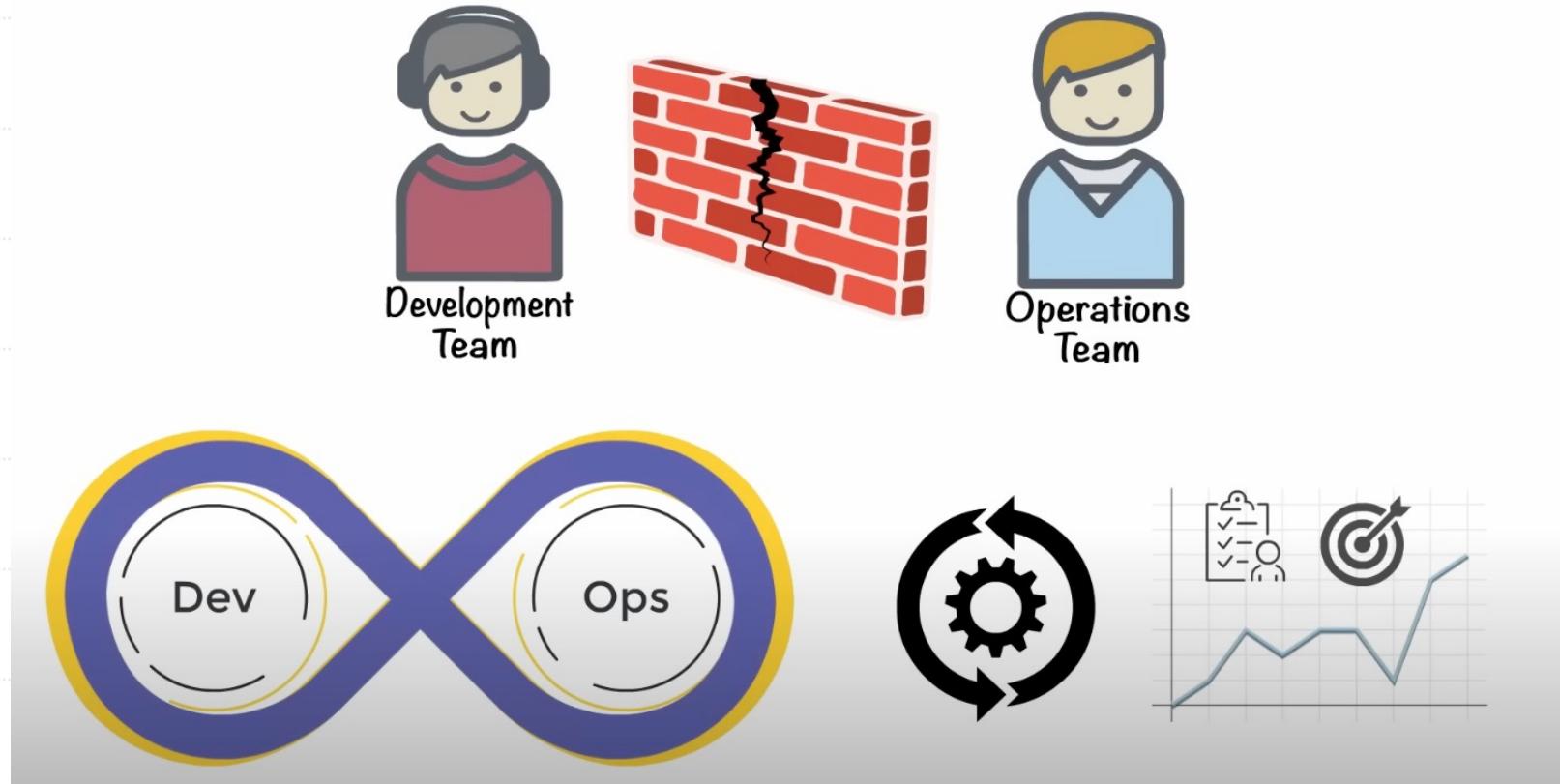
- TIME TO MARKET:** A character holds two signs: "OLD CODE" and "NEW FEATURES".
- PENDING CODE:** A character holds signs for "OLD CODE", "PENDING CODE", and "NEW PRODUCTS".
- DEVELOPMENT ENVIRONMENT:** Represented by a cloud containing a network of five interconnected boxes.
- PRODUCTION ENVIRONMENT:** Represented by a cloud containing a more complex network of ten interconnected boxes.
- Operations:** A character stands next to a stack of seven server racks, with a clock icon above them. The text "99.9% UPTIME" is displayed.

A large bracket on the left groups the first three components under the heading "DevOps Introduction".

YouTube video controls are visible at the bottom left, showing the video is at 0:01 / 7:06.

https://www.youtube.com/watch?v=_I94-tJlovg

What is DevOps?



<https://www.youtube.com/watch?v=Xrgk023l4lI>



What is DevOps?

Technical terms

DevOps is a combination of cultural philosophy, set of practices, and tools that integrate and automate between software development and IT operations team. It emphasizes rapid software development by team empowerment, cross-functional collaboration, simplified communication, and technology automation.

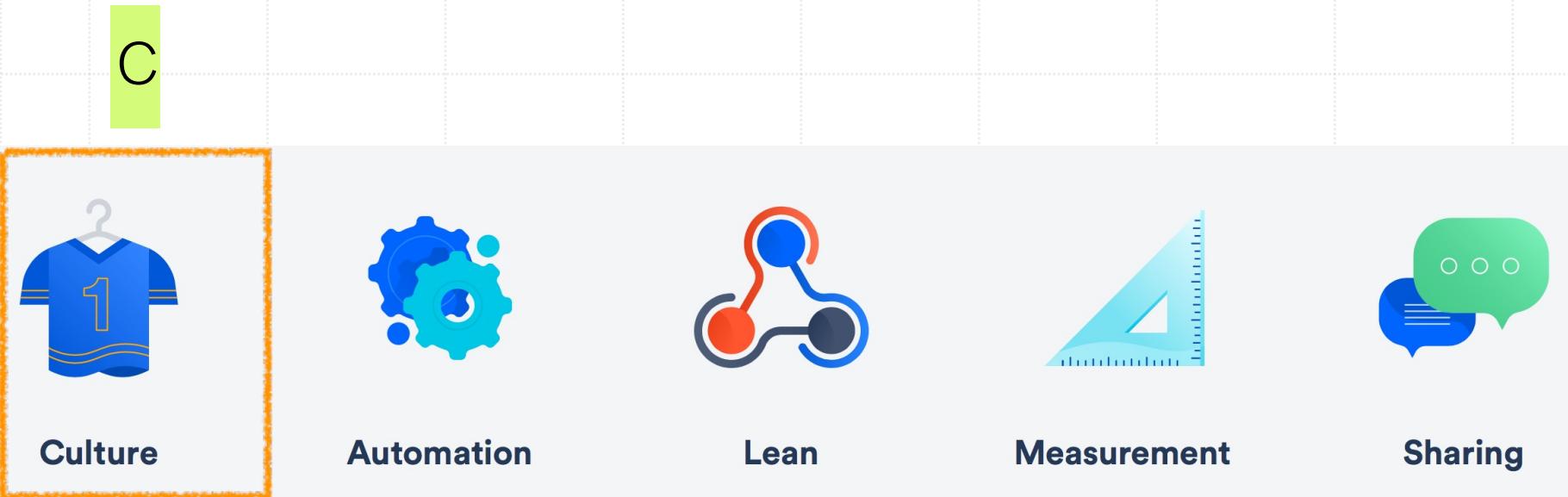


What is DevOps?

The scientists from [CSIRO](#) and [SEI](#) define DevOps as:

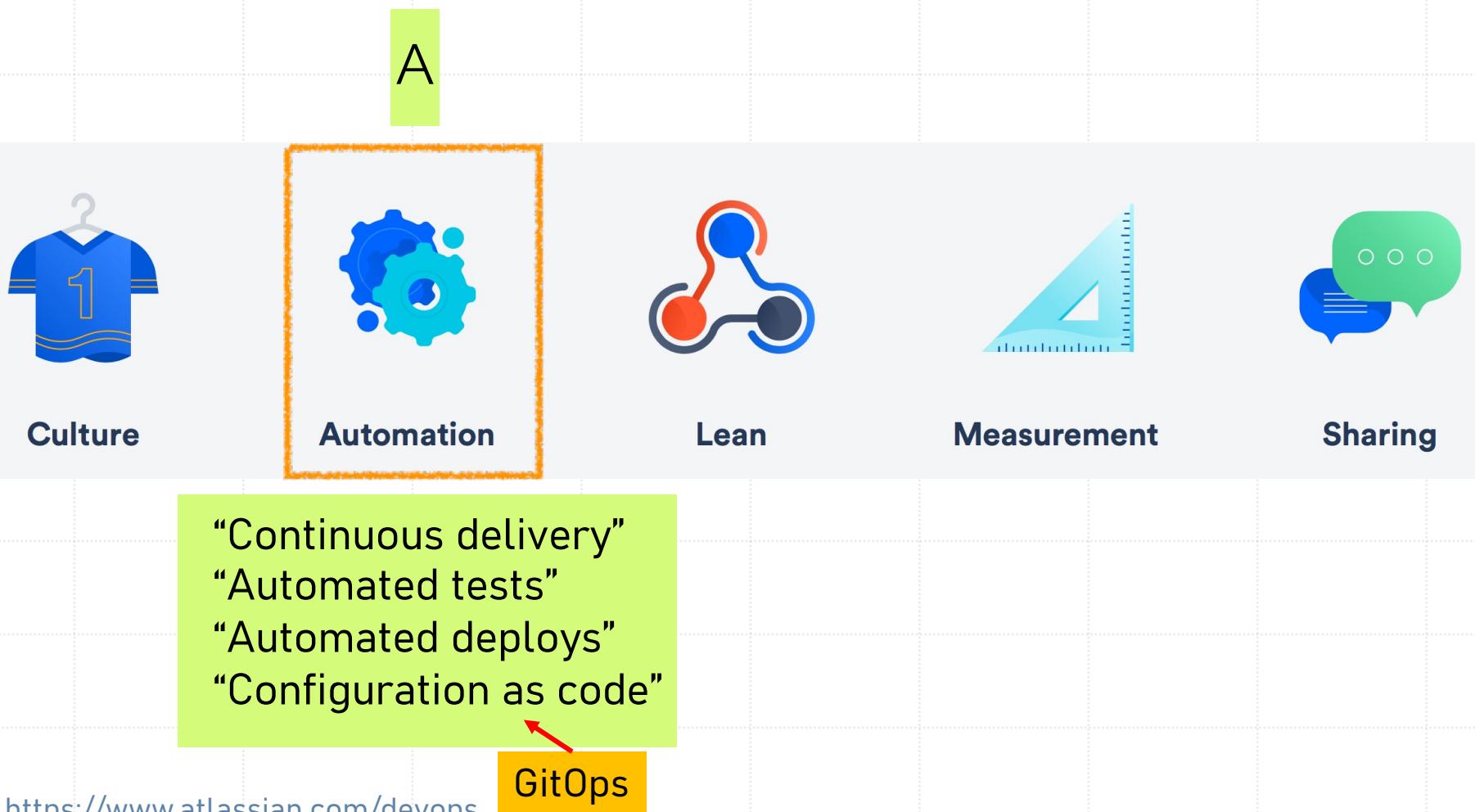
DevOps is a set of practices to reduce the turnaround time from committing a change to a system and implementing that change into a production environment while maintaining high quality.

The CALMS Framework for DevOps

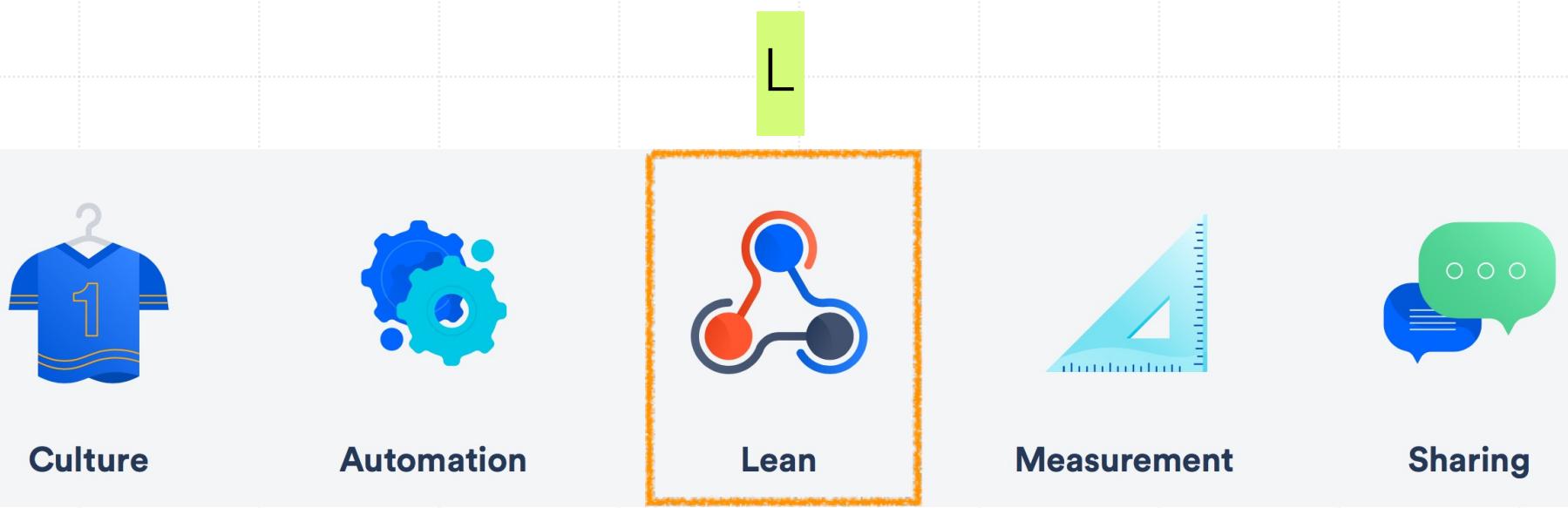


"collaboration"
"cross-functional collaboration"

The CALMS Framework for DevOps

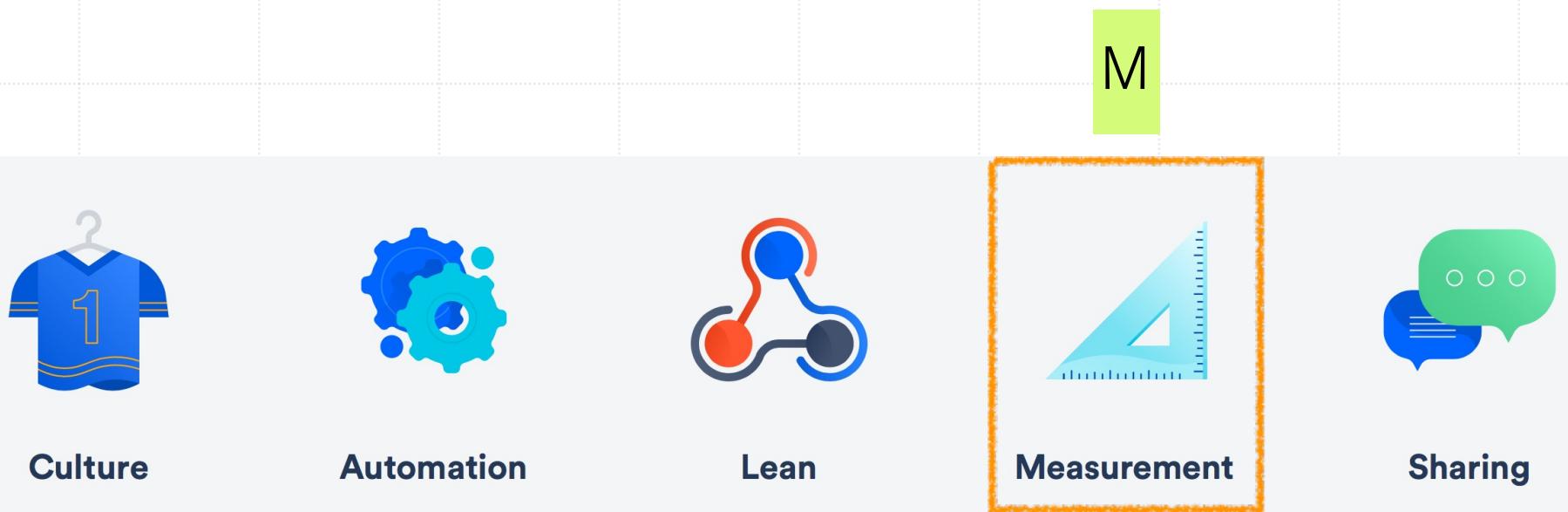


The CALMS Framework for DevOps



“continuous improvement”
being agile, eliminate
low-value activities
“embracing failure”

The CALMS Framework for DevOps



How long did it take to go from development to deployment?
How often do recurring bugs or failures happen?
How long does it take to recover after a system failure?
How many people are using your product right now?
How many users did you gain / lose this week?
Direct and indirect feedbacks

The CALMS Framework for DevOps



Culture



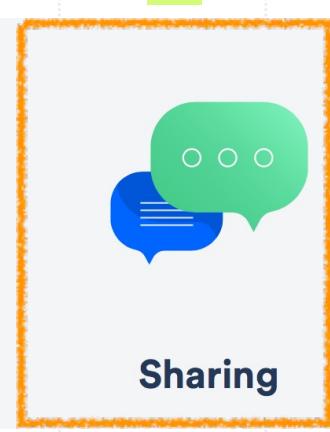
Automation



Lean



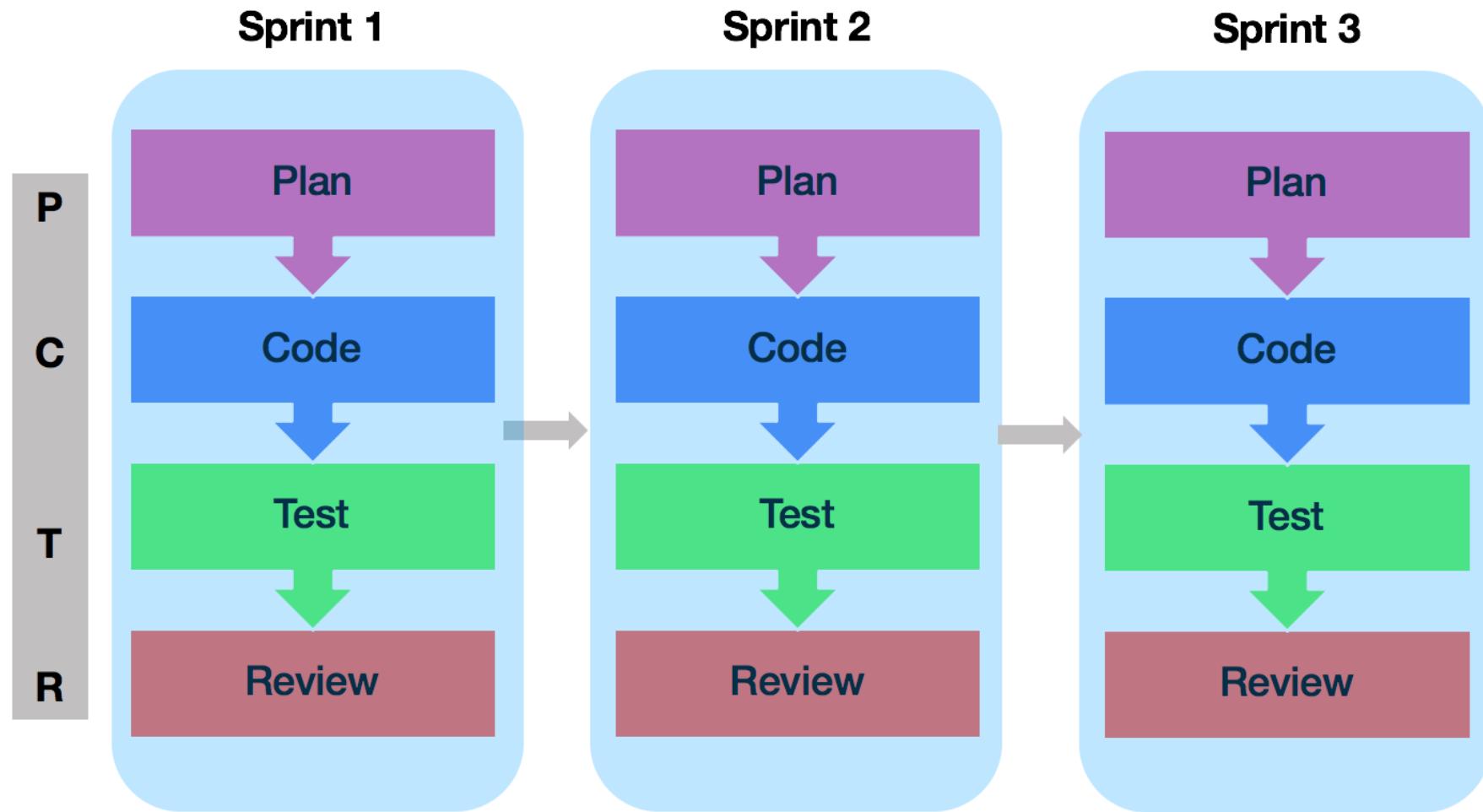
Measurement



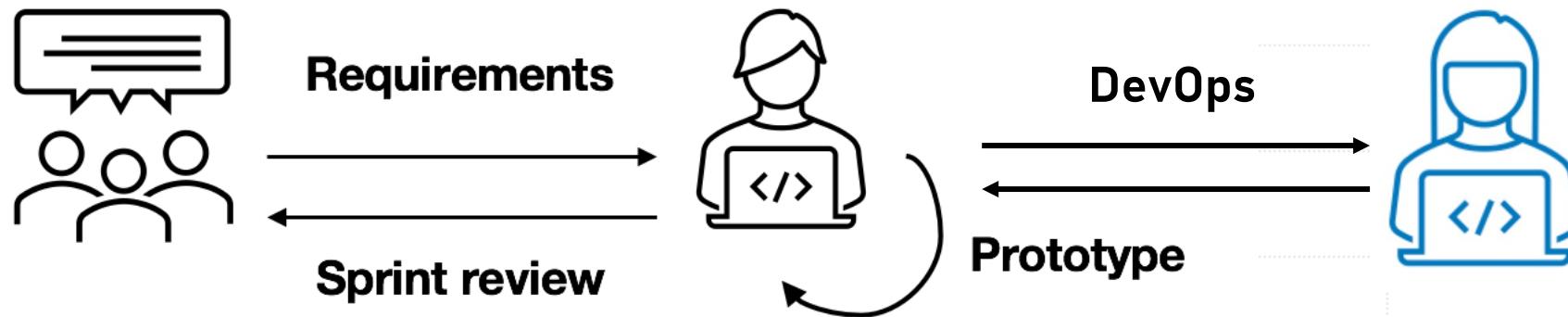
Sharing

same people who build an application should be involved in shipping and running it

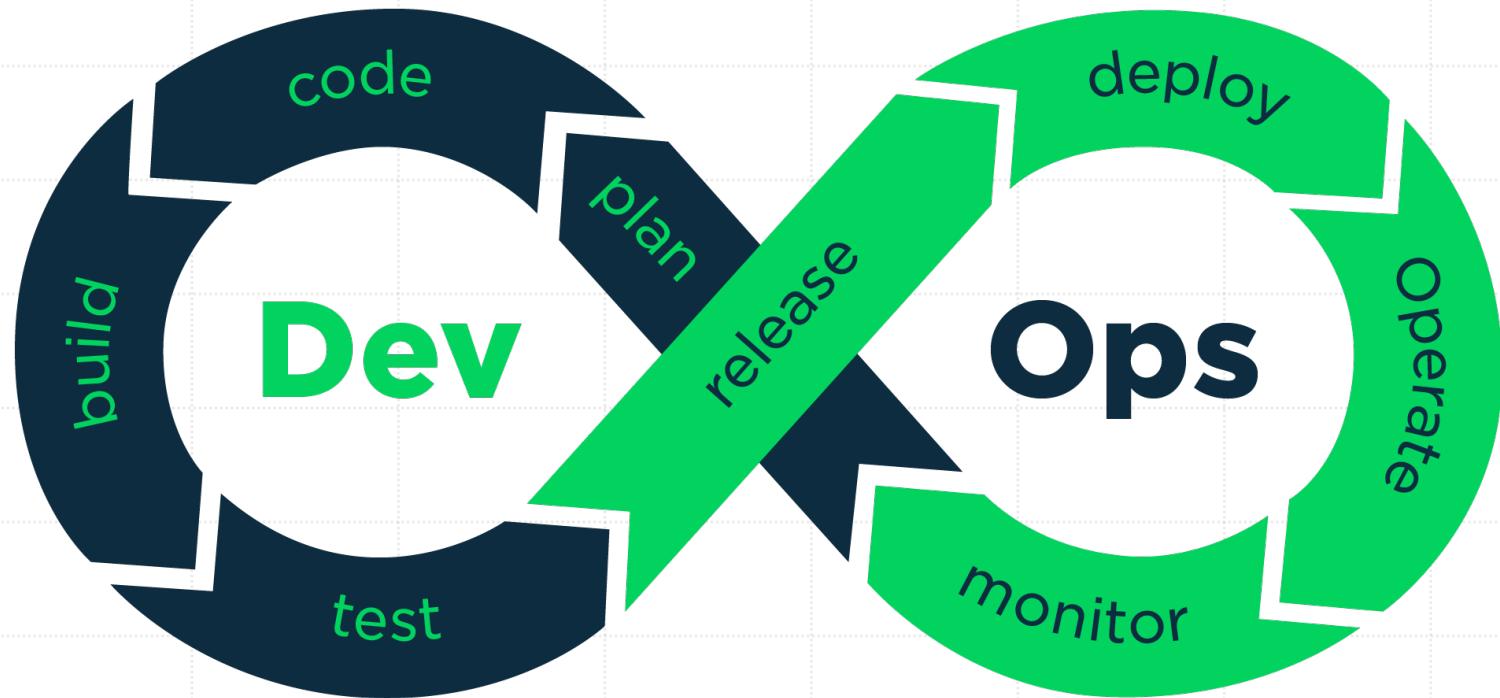
Agile model



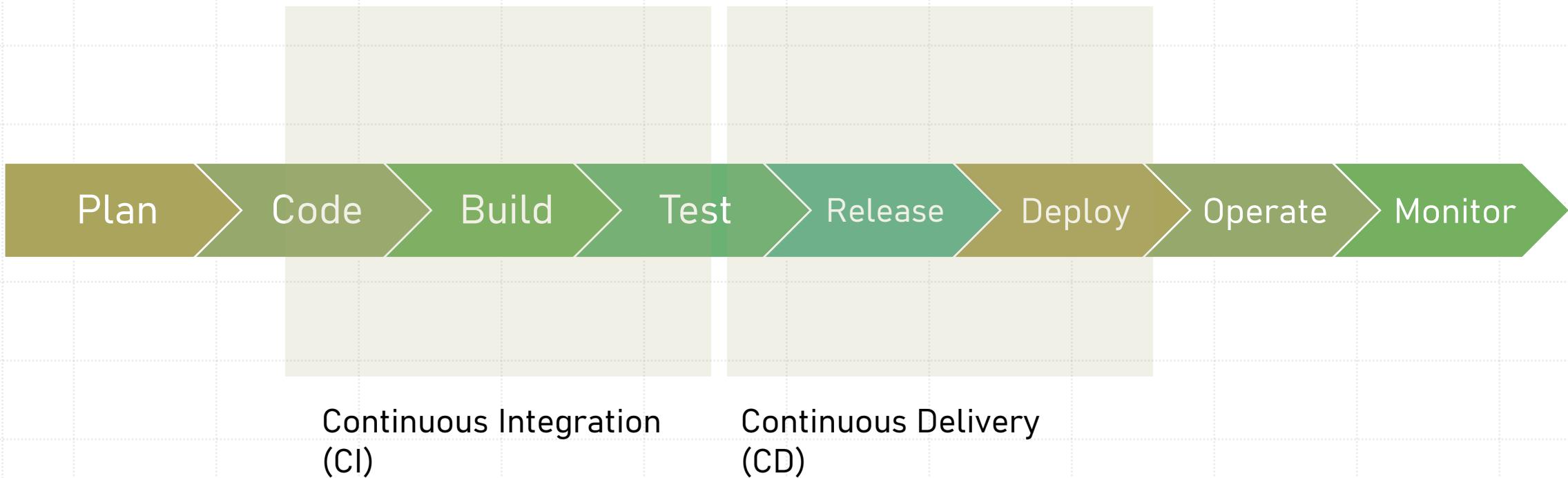
Agile and DevOps



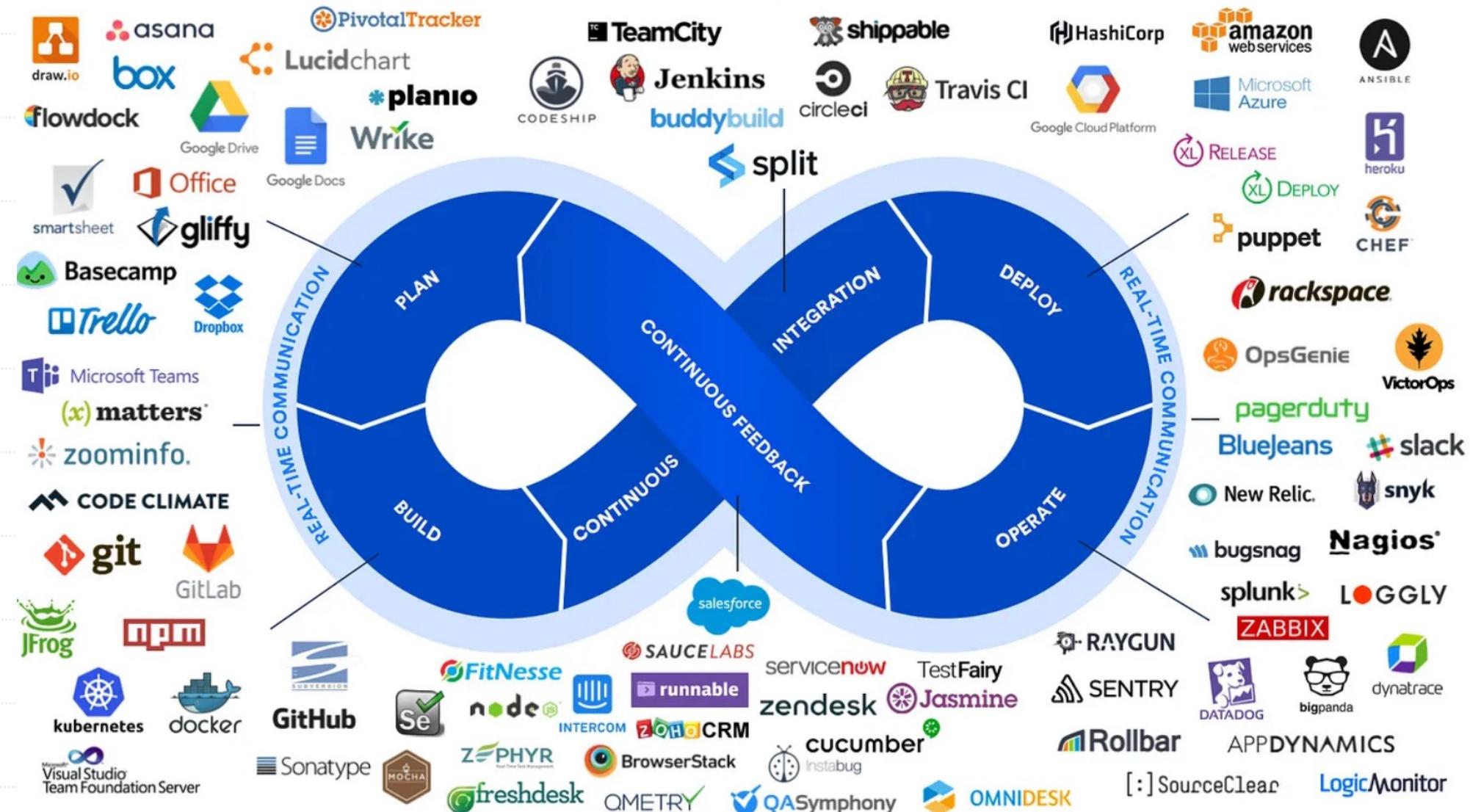
DevOps phases



CI/CD



DevOps Tools





“Even with the best tools, DevOps is just another buzzword if you don't have the right culture.”

Rouan Wilsenach

- Increased collaboration
- Shared responsibility
- No silos
- Autonomous teams
- Building quality into the development process
- Feedbacks
- Automation



Features Actions Packages Security Codespaces Copilot Code review Search Issues Discussions



GitHub Actions

Automate your workflow from idea to production

GitHub Actions makes it easy to automate all your software workflows, now with world-class CI/CD. Build, test, and deploy your code right from GitHub. Make code reviews, branch management, and issue triaging work the way you want.

[Get started with Actions >](#)

Questions? [Contact Sales >](#)

A background graphic featuring several white cards with rounded corners, some containing small icons like a red plus sign or a green checkmark. The cards are arranged in a overlapping, slightly tilted manner, suggesting a workflow or process.

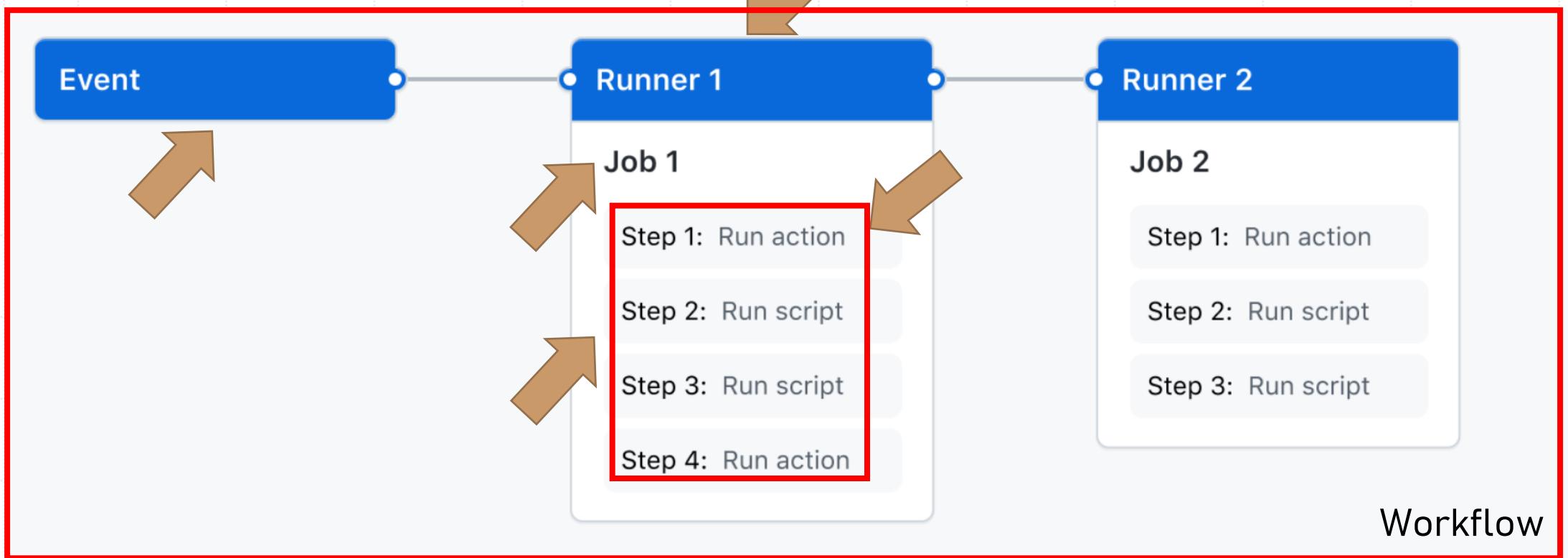
<https://github.com/features/actions>



GitHub Actions

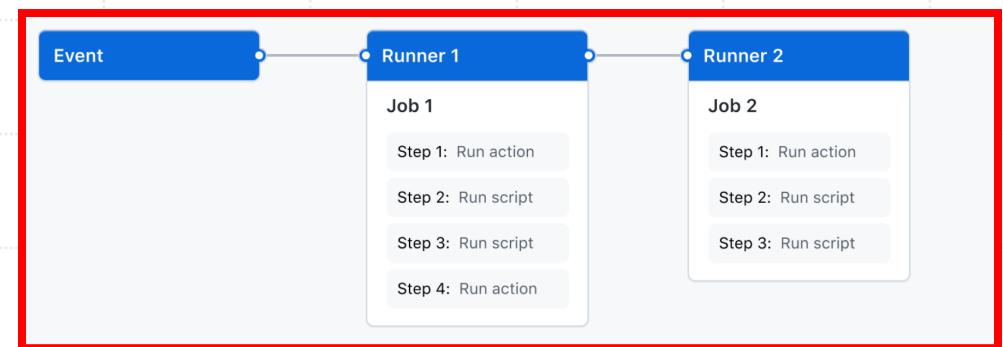
- GitHub Actions is a CI/CD platform.
- We can create workflows that build and test every pull request and deploy merged pull requests to production.
- GitHub Actions goes beyond just DevOps as it also allows to run workflows related to other web hook events in git repo such as issues events.

The components of GitHub Actions



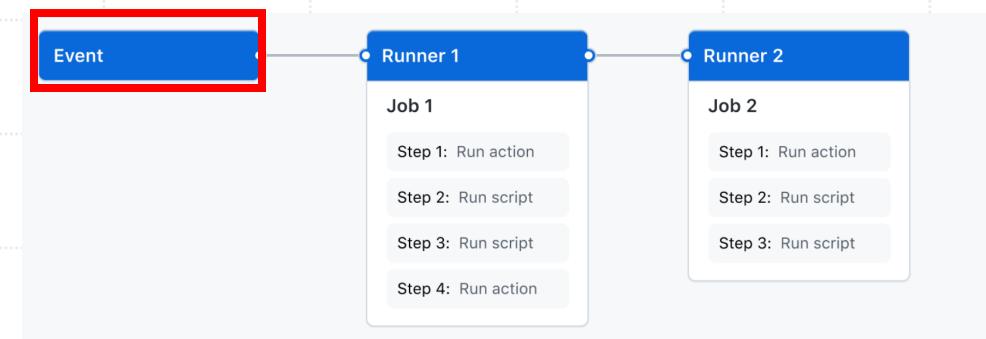
Workflow

- A workflow is a configurable automated process that will run on one or more jobs.
- We can define a workflow, in a YAML file, under the .github/workflow in the repository.
- We can reference a workflow from another workflow, keep in mind to reusing a workflow if possible.



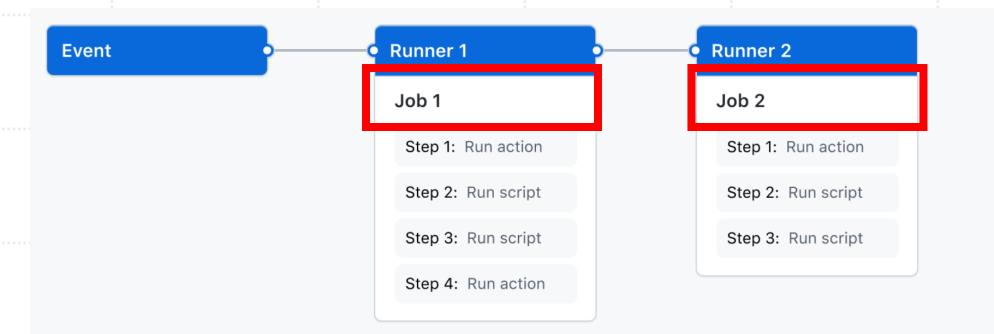
Events

- An event is a specific activity in a repository that triggers a workflow to run.
- Examples of events are pull request, push a commit, opens an issue, schedule a crontab, etc.



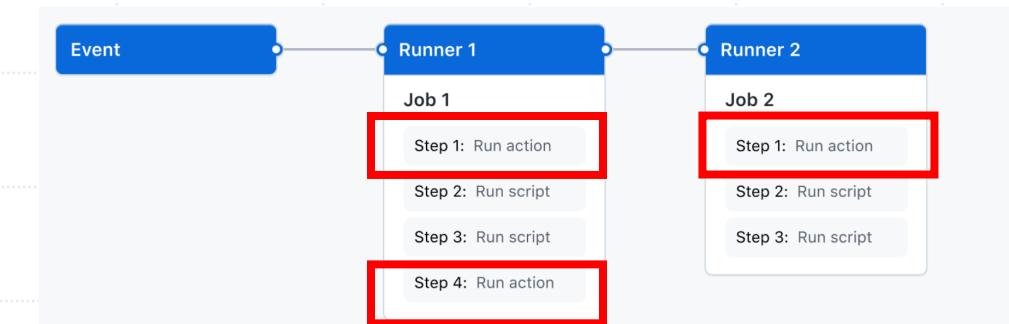
Jobs

- A job is a set of steps in a workflow that execute on the same runner.
- Each step is either (1) a **shell script** or (2) an **action**.



Actions

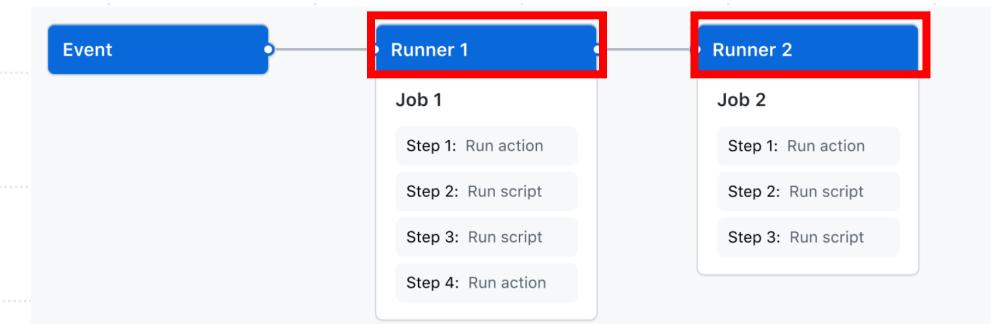
- An action is a custom application for the GitHub Actions platform that performs a complex but frequently repeated task.
- Use an action to reduce the amount of repetitive code.
- Examples of [actions](#) are checkout your code from git repository from GitHub, setup node.js environment, build and push docker image, and many more....



<https://docs.github.com/en/actions/learn-github-actions/understanding-github-actions>

Runners

- A runner is a server that run the workflows when they are triggered.
- Each runner can run a single job at a time.
- Ubuntu Linux, Microsoft Windows, and macOS runners are provided by GitHub.
- We can also host our own runner also.



Standard GitHub-hosted runners for public repositories

Virtual Machine	Processor (CPU)	Memory (RAM)	Storage (SSD)	OS (YAML workflow label)	Notes
Linux	4	16 GB	14 GB	ubuntu-latest , ubuntu-22.04 , ubuntu-20.04	The ubuntu-latest label currently uses the Ubuntu 22.04 runner image.
Windows	4	16 GB	14 GB	windows-latest , windows-2022 , windows-2019	The windows-latest label currently uses the Windows 2022 runner image.
macOS	3	14 GB	14 GB	macos-latest , macos-12 , macos-11	The macos-latest workflow label currently uses the macOS 12 runner image.
macOS	4	14 GB	14 GB	macos-13	N/A
macOS	3 (M1)	7 GB	14 GB	macos-14 [Beta]	N/A

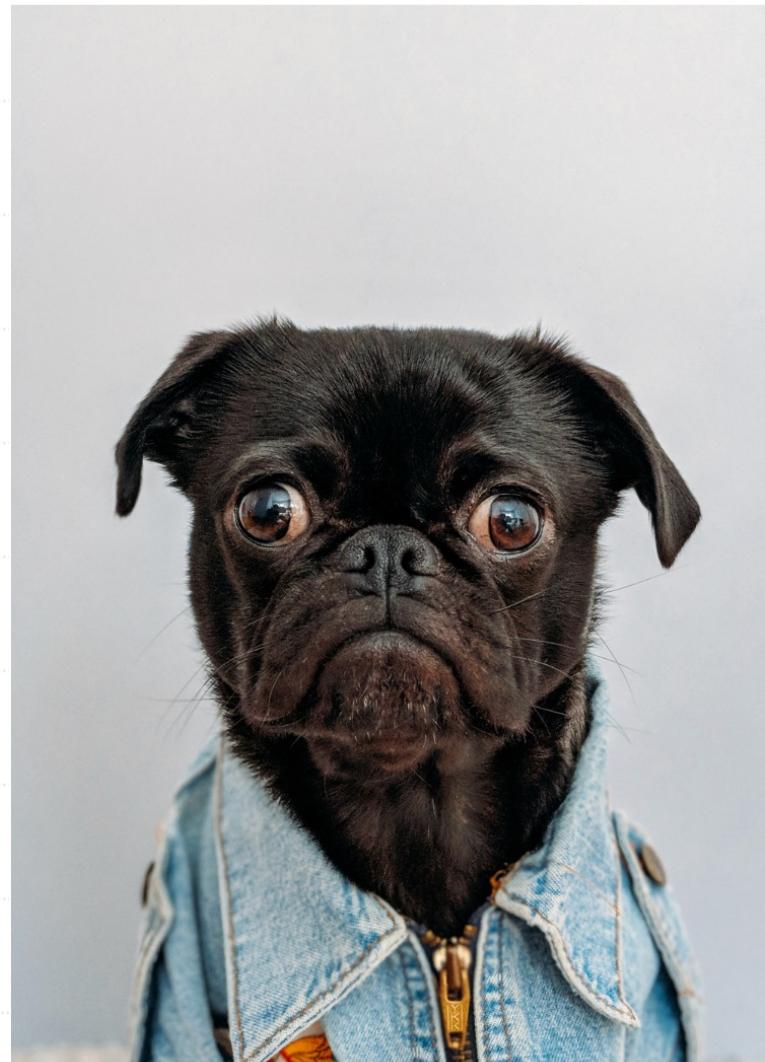
GitHub Actions Example

- <https://github.com/wichadak/GitHubActionDemo1>
 1. **Build a simple return dayOfWeek and monthOfYear with automated test with jest**
 2. **Deploy the app on Firebase**
- Thank <https://github.com/fireship-io/225-github-actions-demo>
- Thank <https://www.youtube.com/watch?v=P0x0LmiknJc> for connecting GitHub Actions with Firebase

You can clone the original code from this this git repo.

What day is it? 😕

Today is Monday



Clone it.

To run the code from <https://github.com/fireship-io/225-github-actions-demo> locally, we can use Firebase hosting. To use the Firebase hosting, we run the following commands.

@localmachine

```
npm i firebase-tools -D
./node_modules/.bin/firebase login:add duangdao.wic@cp.eng.chula.ac.th
./node_modules/.bin/firebase login:use duangdao.wic@cp.eng.chula.ac.th
./node_modules/.bin/firebase init hosting
./node_modules/.bin/firebase init emulators
npm ci
./node_modules/.bin/firebase emulators:start
```

```
GitHubActionDemo1/node_modules/.bin/firebase emulators:start
```

Original codes did not come with the getMothOfYear.



Then,...

What we want?
1. Run automated test and
build when pull request.

{ } package.json U X

{ } package.json > ...

```
1  {
2    "name": "devops",
3    "version": "1.0.0",
4    "description": "",
5    "main": "index.js",
6    ▶ Debug
7    "scripts": {
8      "test": "jest",
9      "build": "webpack src/app.js -o public/bundle.js",
10     "deploy": "firebase deploy --token \"$FIREBASE_TOKEN\" --only hosting"
11   },
12   "keywords": [],
13   "author": "",
14   "license": "ISC",
15   "devDependencies": {
16     "firebase-tools": "^13.6.0",
17     "jest": "^25.1.0",
18     "webpack": "^4.42.0",
19     "webpack-cli": "^3.3.11"
20   },
21   "dependencies": {
22     "terser": "^4.6.6"
23 }
```



Then,...

What we want?

1. Run automated test and build when pull request.

Push the original code to git repo.

The screenshot shows a GitHub repository page for 'GitHubActionDemo1' owned by 'wichadak'. The 'Actions' tab is highlighted with a red box and a large orange arrow pointing towards it from the left. The repository is public. The main content area displays a list of recent commits:

File	Message	Time
public	adding original codes to the git repo	17 minutes ago
src	adding original codes to the git repo	17 minutes ago
README.md	Update README.md	14 minutes ago
firebase.json	adding original codes to the git repo	17 minutes ago
package-lock.json	adding original codes to the git repo	17 minutes ago
package.json	adding original codes to the git repo	17 minutes ago

A green callout bubble on the right side of the screen contains the text: "Create the first workflow file." A green diagonal bar also points towards the 'Actions' tab.

A screenshot of a GitHub repository page for 'wichadak / GitHubActionDemo1'. The page includes a header with a search bar, a profile picture, and navigation links for Code, Issues, Pull requests, Actions (which is underlined in red), Projects, Wiki, Security, Insights, and Settings. The main content area features a section titled 'Get started with GitHub Actions' with a sub-instruction 'Build, test, and deploy your code. Make code reviews, branch management, and issue triaging work the way you want. Select a workflow to get started.' Below this is a button labeled 'Skip this and set up a workflow yourself →' which is highlighted with a red box and has a brown arrow pointing towards it from the left. A search bar for workflows is also present. To the right, there is a large orange callout bubble containing the text: 'There are a lot of workflow templates here... Check them out.'

Get started with GitHub Actions

Build, test, and deploy your code. Make code reviews, branch management, and issue triaging work the way you want. Select a workflow to get started.

Skip this and [set up a workflow yourself →](#)



Search workflows

Suggested for this repository

Jekyll using Docker image

By GitHub Actions

Package a Jekyll site using the jekyll.builder Docker image.

[Configure](#)

HTML

Publish Node.js Package to GitHub Packages

By GitHub Actions

Publishes a Node.js package to GitHub Packages.

[Configure](#)

JavaScript

Publish Node.js Package

By GitHub Actions

Publishes a Node.js package to npm.

[Configure](#)

JavaScript

wichadak / GitHubActionDemo1

Type / to search

Code Issues Pull requests 1 Actions Projects Wiki Security Insights Settings

GitHubActionDemo1/.github/workflows/integrate.yml in main

Cancel changes Commit changes... ←

Edit Preview Spaces 2 No wrap

```
1 name: Node Continuous Integration
2
3 on:
4   pull_request:
5     branches: [ main ]
6
7 jobs:
8   test_pull_request:
9     runs-on: ubuntu-latest
10    steps:
11      - uses: actions/checkout@v4
12      - uses: actions/setup-node@v4
13        with:
14          node-version: 16
15      - run: npm ci
16      - run: npm test
17      - run: npm run build
```

Marketplace Documentation

Search Marketplace for Actions

Featured Actions

Cache By actions ★ 4.2k

Cache artifacts like dependencies and build outputs to improve workflow execution time

Setup Java JDK By actions ★ 1.4k

Set up a specific version of the Java JDK and add the command-line tools to the PATH

wichadak / GitHubActionDemo1

Type / to search

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

GitHubActionDemo1/.github/workflows/main.yml

Edit Preview

```
1 name: Node Continuous Integration
2
3 on:
4   pull_request:
5     branches: [ main ]
6
7 jobs:
8   test_pull_request:
9     runs-on: ubuntu-latest
10    steps:
11      - uses: actions/checkouts@v2
12      - uses: actions/setup-node@v2
13        with:
14          node-version: 16
15        - run: npm ci
16        - run: npm test
17        - run: npm run build
```

Commit changes

Commit message

Create integrate.yml

Extended description

Add an optional extended description..

Commit directly to the main branch

Create a new branch for this commit and start a pull request

Learn more about pull requests

Cancel Commit changes

Cancel changes Commit changes...

Documentation

Search Marketplace for Actions

Selected Actions

Setup Node.js environment By actions ★ 3.5k

Setup a Node.js environment by adding problem matchers and optionally downloading and adding it to the PATH

Upload a Build Artifact By actions ★ 2.8k

Upload a build artifact that is used by subsequent workflow steps

 main  GitHubActionDemo1 / .github / workflows / 

Go to file



Add file 

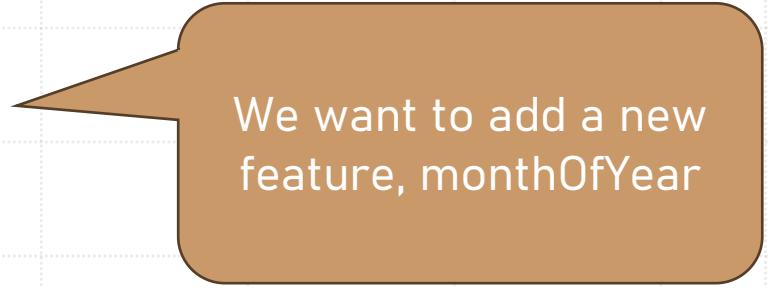


 **wichadak** Create integrate.yml

5eb43c5 · now  History

Name	Last commit message	Last commit date
 ..		
 integrate.yml	Create integrate.yml	now

@ localmachine under the folder GitHubActionDemo1
cloned from the git repo



We want to add a new feature, monthOfYear

Run the following command:

```
git checkout -b addMonthOfYear  
git status
```

Add feature addMonthOfYer to src/app.js and add test of this addMothnOfYear to src/app.test.js

```
git add .  
git commit -m "update app.js and app.test.js"  
git push --set-upstream origin addMonthOfYear
```

wichadak / GitHubActionDemo1

Type ⌘ to search

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

GitHubActionDemo1 Public

Pin Unwatch 1 Fork 0 Star 0

addMonthOfYear had recent pushes 19 seconds ago

Compare & pull request

About

No description, website, or topics provided.

main Go to file + <> Code

wichadak Create integrate.yml 5eb43c5 · 37 minutes ago 3 Commits

.github/workflows Create integrate.yml 37 minutes ago

public adding original codes to the git repo 1 hour ago

src adding original codes to the git repo 1 hour ago

README.md Update README.md 1 hour ago

firebase.json adding original codes to the git repo 1 hour ago

package-lock.json adding original codes to the git repo 1 hour ago

package.json adding original codes to the git repo 1 hour ago

Readme Activity 0 stars 1 watching 0 forks

Releases No releases published [Create a new release](#)

Packages No packages published [Publish your first package](#)

▼ 9 src/app.js

```
... @@ -4,10 +4,17 @@ const dayOfTheWeek = (date = new Date()) => {
 4   4     return days[date.getDay()];
 5   5   }
 6   6
 7 + const monthOfTheYear = (date = new Date()) => {
 8 +   const months = ['January', 'February', 'March', 'April', 'May', 'June', 'July', 'August',
 9 +     'September', 'October', 'November', 'December'];
 10 +   return months[date.getMonth()];
 11 + }
 7 12
 8 13   try {
 9 14     document.getElementById('day').innerText = dayOfTheWeek();
10 +   document.getElementById('month').innerText = monthOfTheYear();
11 16   } catch(err) {}
12 17
13 18
13 - exports.dayOfTheWeek = dayOfTheWeek;
  ⊖
19 + exports.dayOfTheWeek = dayOfTheWeek;
20 + exports.monthOfTheYear = monthOfTheYear;
```

✓ 7  src/app.test.js 

• • •

```
...     ... @@ -1,7 +1,12 @@
1     1     const { dayOfTheWeek } = require('./app');
2 +   const { monthOfTheYear } = require('./app');

2     3
3     4
4     5     test('getDay returns the long-format day of the week', () => {
5       6       const day = dayOfTheWeek( new Date('3/11/2020') );
6       7       expect( day ).toBe('Wednesday');

7     - });
    - ⊖
8     + );
9     + test('getDay returns the month of the year', () => {
10    +   const month = monthOfTheYear( new Date('3/11/2020') );
11    +   expect( month ).toBe('November');
12    + );
```



Add a title

update app.js and app.test.js

Add a description

Write

Preview

H

B

I

E

<>

🔗

≡

≡

≡

📎

@

...

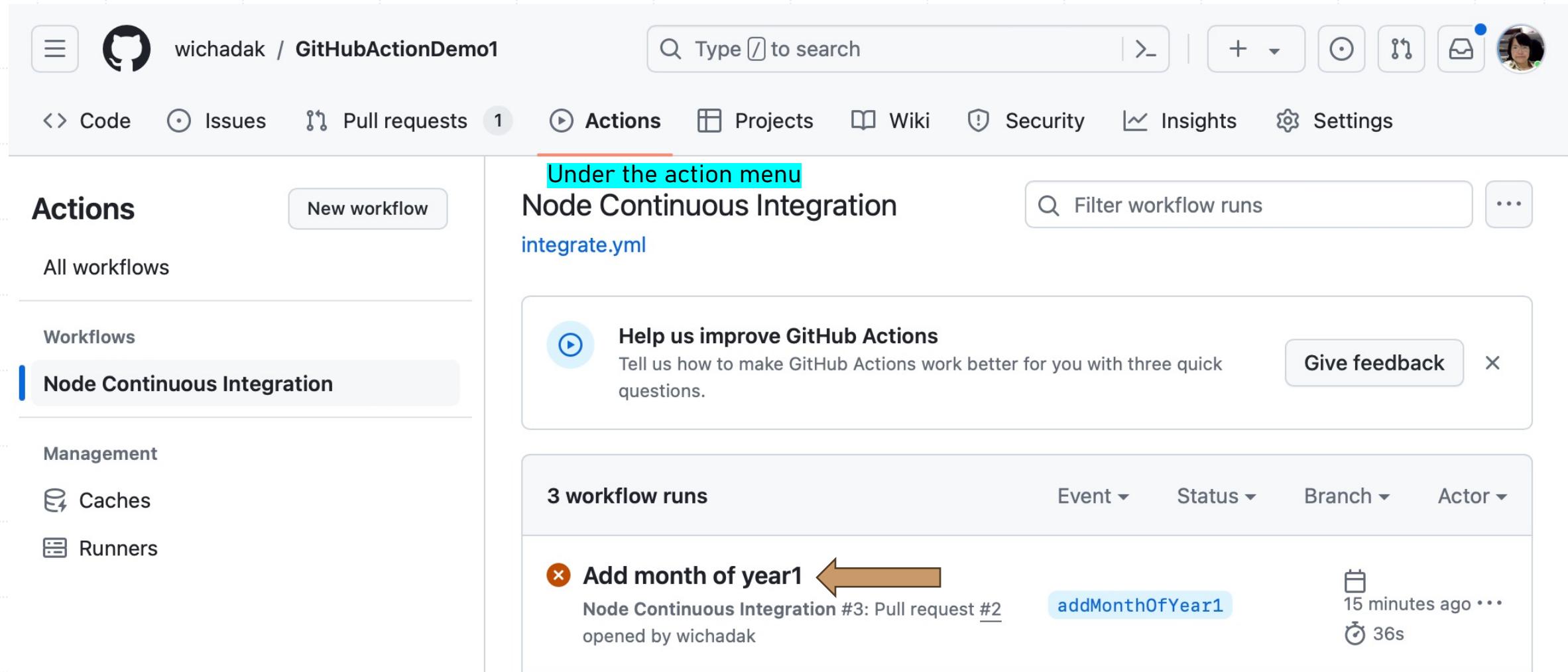
Add your description here...

Markdown is supported

Paste, drop, or click to add files

Create pull request





wichadak / GitHubActionDemo1

Type ⌘ to search

Code Issues Pull requests 1 Actions Projects Wiki Security Insights Settings

← Node Continuous Integration

Add month of year1 #3

Re-run jobs ...

Summary

Triggered via pull request 16 minutes ago

wichadak opened #2 addMonthOfYear1 Status Failure Total duration 36s

Artifacts

test_pull_request

Run details

Usage

Workflow file

integrate.yml
on: pull_request

test_pull_request 28s

A large orange arrow points from the "Failure" status text to the "test_pull_request" step in the workflow log.

← Node Continuous Integration

✖ Add month of year1 #3

Re-run jobs

...

Summary

Jobs

✖ test_pull_request

Run details

⌚ Usage

↳ Workflow file

test_pull_request

failed 16 minutes ago in 28s

Beta Give feedback Search logs

- > ✓ Set up job
- > ✓ Run actions/checkout@v4
- > ✓ Run actions/setup-node@v4
- > ✓ Run npm ci
- > ✖ Run npm test
- > ⏺ Run npm run build
- > ⏺ Post Run actions/setup-node@v4
- > ✓ Post Run actions/checkout@v4
- > ✓ Complete job

Summary

Jobs

test_pull_request

Run details

Usage

Workflow file

test_pull_request

failed 17 minutes ago in 28s

Beta

Give feedback

Search logs

Run npm test

```
10      x getDay returns the month of the year (3ms)
11
12      ● getDay returns the month of the year
13
14          expect(received).toBe(expected) // Object.is equality
15
16          Expected: "November"
17          Received: "March"
18
19          9 | test('getDay returns the month of the year', () => {
20          10 |     const month = monthOfTheYear( new Date('3/11/2020') );
21          > 11 |     expect( month ).toBe('November');
22          |
23          12 | });
24          13 |
25
26          at Object.<anonymous> (src/app.test.js:11:21)
27          at processTicksAndRejections (node:internal/process/task_queues:96:5)
28
29 Test Suites: 1 failed, 1 total
30 Tests:       1 failed, 1 passed, 2 total
31 Snapshots:   0 total
32 Time:        0.807s
33 Ran all test suites.
34 Error: Process completed with exit code 1.
```

@local machine

Fixed the
app.test.js in local
machine

Then, git add, git
commit, and git push
again

Add month of year1 #2

[Edit](#)[Code](#)[Open](#)wichadak wants to merge 4 commits into `main` from `addMonthOfYear1`[Conversation](#) 0[Commits](#) 4[Checks](#) 1[Files changed](#) 2[+14 -2](#)[Changes from 1 commit](#)[File filter](#)[Conversations](#)[Jump to](#)[Review in codespace](#)[Review changes](#)[fixed app.test.js](#)[◀ Prev](#) [Next ▶](#)[addMonthOfYear1 \(#2\)](#)

wichadak committed 1 minute ago

commit 82c1bcac745e0b4aad3d2ae31ca610f8c35f39b8

[src/app.test.js](#)[...](#)

```
@@ -8,5 +8,5 @@ test('getDay returns the long-format day of the week', () => {
 8   8   });
 9   9   test('getDay returns the month of the year', () => {
10   10   const month = monthOfTheYear( new Date('3/11/2020') );
11 - 11   expect( month ).toBe('November');
11 + 11   expect( month ).toBe('March');
12   12   });

```

Add month of year1 #2

[Edit](#)[Code](#)[Open](#)wichadak wants to merge 4 commits into `main` from `addMonthOfYear1`

Conversation 0

Commits 4

Checks 1

Files changed 2

+14 -2

✓ fixed app.test.js 82c1bca

[Re-run all jobs](#)

✓ Node Continuous Integration
on: pull_request

test_pull_request Beta

succeeded 1 minute ago in 30s

[Search logs](#)

✓ test_pull_request

- > ✓ Set up job
- > ✓ Run actions/checkout@v4
- > ✓ Run actions/setup-node@v4
- > ✓ Run npm ci
- > ✓ Run npm test
- > ✓ Run npm run build
- > ✓ Post Run actions/setup-node@v4
- > ✓ Post Run actions/checkout@v4
- > ✓ Complete job

github action
successfully run



Then,...

What we want?
2. Deploy the app on
Firebase automatically

wichadak / GitHubActionDemo1

Type / to search

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

GitHubActionDemo1 Public

Pin Unwatch 1 Fork 0 Star 0

main Go to file Code

wichadak Update README.md da7ca6b · 14 minutes ago 2 Commits

public adding original codes to the git repo 17 minutes ago

src adding original codes to the git repo 17 minutes ago

README.md Update README.md 14 minutes ago

firebase.json adding original codes to the git repo 17 minutes ago

package-lock.json adding original codes to the git repo 17 minutes ago

package.json adding original codes to the git repo 17 minutes ago

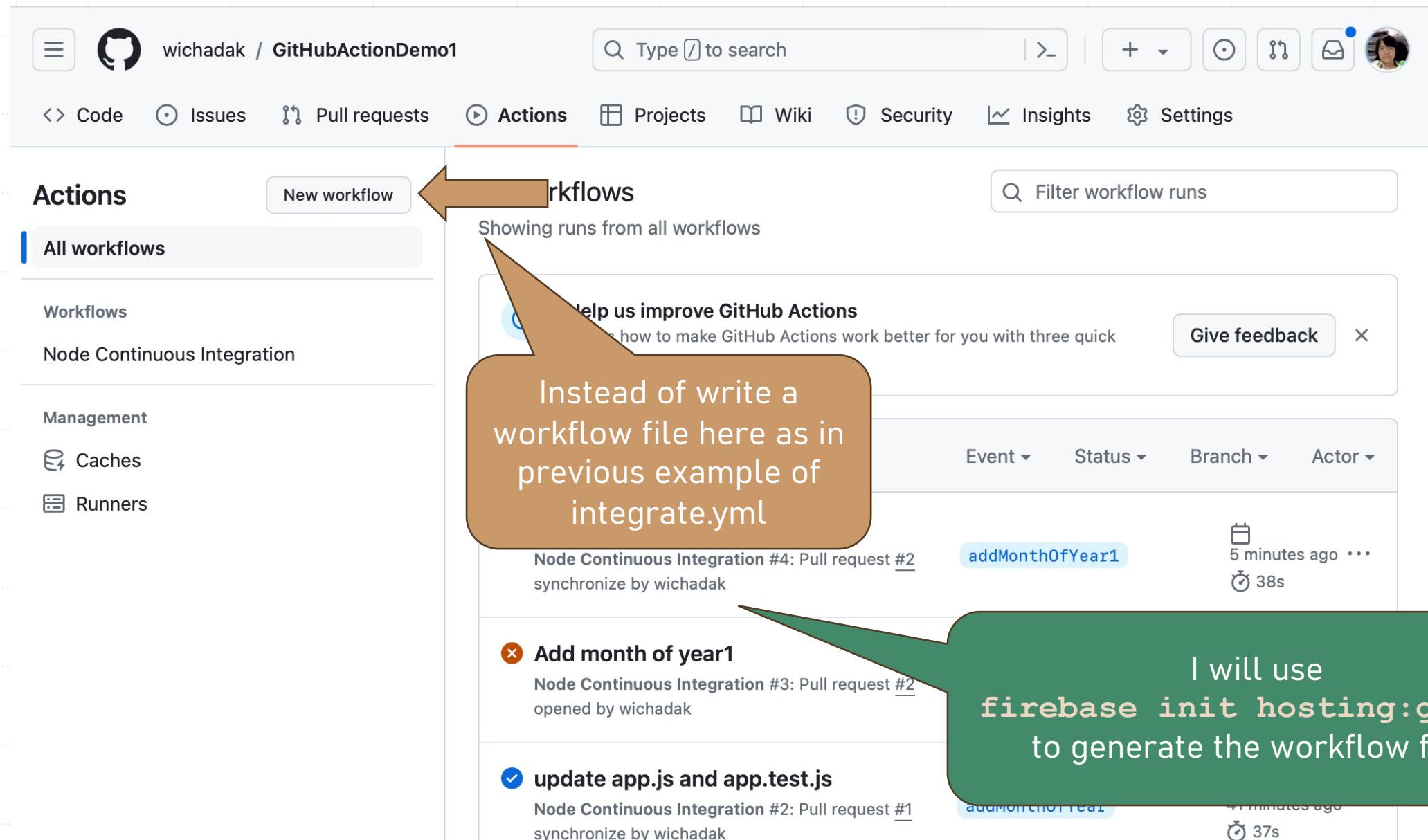
About

No description, website, or topics provided.

Readme Activity 0 stars 1 watching 0 forks

Releases

No releases published Create a new release



Instead of write a workflow file here as in previous example of integrate.yml

I will use
firebase init hosting:github
to generate the workflow files

@local machine

```
npm i firebase-tools -D # D for local within node_modules to avoid sudo issue  
.node_modules/.bin/firebase login:add duangdao.wic@cp.eng.chula.ac.th  
.node_modules/.bin/firebase login:use duangdao.wic@cp.eng.chula.ac.th  
.node_modules/.bin/firebase init hosting:github
```

```
(base) ~/Desktop/2110336_Workshops/CICD/GitHubActionDemo1 $ ./node_modules/.bin/firebase init hosting:github  
(node:77585) [DEP0040] DeprecationWarning: The `punycode` module is deprecated. Please use a userland alternative instead.  
(Use `node --trace-deprecation ...` to show where the warning was created)
```

```
##### #### ##### #### ##### #### ##### ####  
## ## ## ## ## ## ## ## ## ## ## ## ##  
##### ## ##### ## ##### ## ##### ## #####  
## ## ## ## ## ## ## ## ## ## ## ## ##  
## ##### ## ## ##### ## ##### ## ## #####
```

```
You're about to initialize a Firebase project in this directory:
```

```
[ /Users/duangdaowichadakul/Desktop/2110336_Workshops/CICD/GitHubActionDemo1
```

```
Before we get started, keep in mind:
```

- * You are initializing within an existing Firebase project directory

==== Project Setup

```
First, let's associate this project directory with a Firebase project.  
You can create multiple project aliases by running firebase use --add,  
but for now we'll just set up a default project.
```

```
i Using project cicd1-8748d (CICD1)
```

==== Hosting:github Setup

```
i Detected a .git folder at /Users/duangdaowichadakul/Desktop/2110336_Workshops/CICD/GitHubActionDemo1  
i Authorizing with GitHub to upload your service account to a GitHub repository's secrets store.
```

```
Visit this URL on this device to log in:
```

```
https://github.com/login/oauth/authorize?client\_id=89cf50f02ac6aaed3484&state=663413293&redirect\_uri=http%3A%2F%2Flocalhost%3A9005&scope=read%3Auser%20repo%20public\_repo
```

```
Waiting for authentication...
```

```
Waiting for authentication...

✓ Success! Logged into GitHub as wichadak

[?] For which GitHub repository would you like to set up a GitHub workflow? (format: user/repository) wichadak/GitHubActionDemo1

✓ Created service account github-action-779858803 with Firebase Hosting admin permissions.
✓ Uploaded service account JSON to GitHub as secret FIREBASE_SERVICE_ACCOUNT_CICD1_8748D.
ℹ You can manage your secrets at https://github.com/wichadak/GitHubActionDemo1/settings/secrets.

[?] Set up the workflow to run a build script before every deploy? Yes
[?] What script should be run before every deploy? npm ci && npm run build

✓ Created workflow file /Users/duangdaowichadakul/Desktop/2110336_Workshops/CICD/GitHubActionDemo1/.github/workflows/firebase-hosting-pull-request.yml
[?] Set up automatic deployment to your site's live channel when a PR is merged? Yes
[?] What is the name of the GitHub branch associated with your site's live channel? main

✓ Created workflow file /Users/duangdaowichadakul/Desktop/2110336_Workshops/CICD/GitHubActionDemo1/.github/workflows/firebase-hosting-merge.yml

ℹ Action required: Visit this URL to revoke authorization for the Firebase CLI GitHub OAuth App:  

https://github.com/settings/connections/applications/89cf50f02ac6aaed3484
ℹ Action required: Push any new workflow file(s) to your repo

ℹ Writing configuration info to firebase.json...
ℹ Writing project information to .firebaserc...
```

```
npm i firebase-tools -D # D for local within node_modules to avoid sudo issue  
./node_modules/.bin/firebase login:add duangdao.wic@cp.eng.chula.ac.th  
./node_modules/.bin/firebase login:use duangdao.wic@cp.eng.chula.ac.th  
./node_modules/.bin/firebase init hosting:github
```

Then, we will get two workflow files.

```
.github/workflows/firebase-hosting-merge.yml  
.github/workflows/firebase-hosting-pull-request.yml
```

We also will get the updated

`firebase.json` and `.firebaserc`

```
git add . # to dd these four files to github actions of this repo  
git commit -m "..."  
git push
```

Summary

Jobs

build_and_preview

Deploy Preview

Run details

⌚ Usage

📄 Workflow file

build_and_preview

succeeded 2 minutes ago in 53s

Beta

Give feedback

Search logs



> Run npm ci && npm run build

20s

< Run FirebaseExtended/action-hosting-deploy@v0

29s

```
1   ▼ Run FirebaseExtended/action-hosting-deploy@v0
2     with:
3       repoToken: ***
4       firebaseServiceAccount: ***
5
6       projectId: cicd1-8748d
7       expires: 7d
8       entryPoint: .
9       firebaseToolsVersion: latest
10  ▼ Verifying firebase.json exists
11    firebase.json file found. Continuing deploy.
12  ▼ Setting up CLI credentials
13    Created a temporary file with Application Default Cred...
14  ▼ Deploying to Firebase preview channel pr1-addMonthOfY...
15    /usr/local/bin/npx firebase-tools@latest hosting:chan...
16    cicd1-8748d --json
17    npm WARN exec The following package was not found and is required by your script: firebase-tools@13.6.0
18    ***  
19      "status": "success",
20      "result": ***  
21      "cicd1-8748d": ***  
22      "site": "cicd1-8748d".
23      "url": "https://cicd1-8748d--pr1-addmonthofyear-xt02jftm.web.app",  
24      "version": "",  
      "expireTime": "2024-04-08T16:11:01.463027982Z"
```

Successfully deployed
to Firebase.



GitHub Actions (another example)

build and test a very simple GO app

- <https://github.com/cucpbioinfo/SimpleGO>
- <https://docs.github.com/en/actions/automating-builds-and-tests/building-and-testing-go>

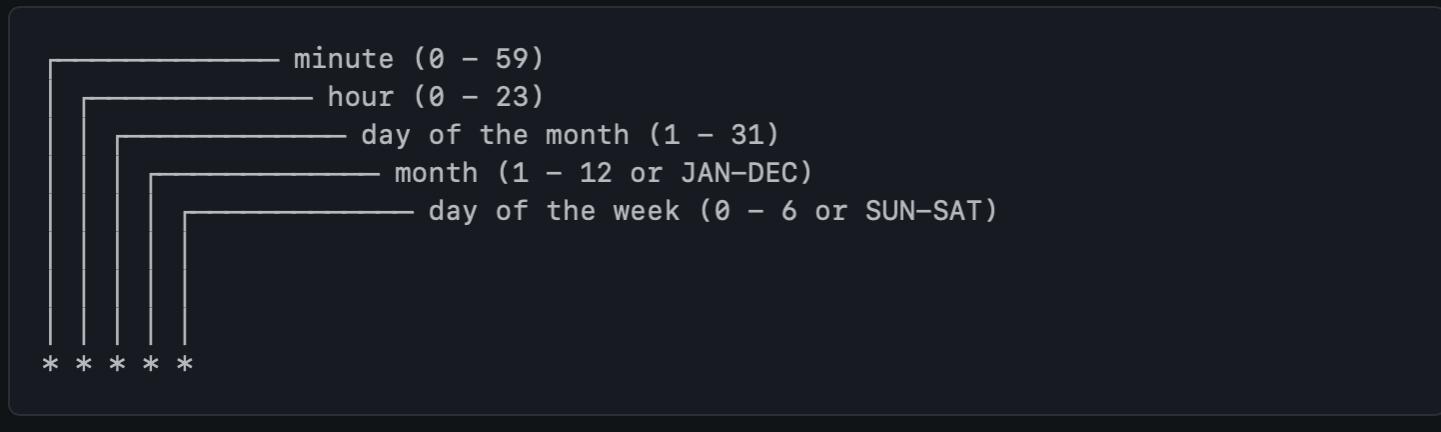
Schedule background jobs

```
1 name: Export Firestore Data
2
3   on:
4     schedule:
5       - cron: '0 0 * * *'
6
7   jobs:
8     backup:
9       runs-on: ubuntu-latest
10      steps:
11        - uses: GoogleCloudPlatform/github-actions/setup-gcloud@master
12        - uses: actions/setup-node@master
13          with:
14            service_account_key: ${{secrets.GCP_SA_KEY}}
15            export_default_credentials: true
16        - run: gcloud config set project $PROJECT_ID
17        - run: gcloud firestore export $BUCKET
```

Schedule

```
1 name: Export
2
3 on:
4   schedule:
5     - cron: '0 0 * * *'
6
7 jobs:
8   backup:
9     runs-on: ubuntu-latest
10    steps:
11      - uses: GoogleCloudPlatform/github-actions/setup-gcloud@master
12      - uses: actions/setup-node@master
13        with:
14          service_account_key: ${{secrets.GCP_SA_KEY}}
15          export_default_credentials: true
16      - run: gcloud config set project $PROJECT_ID
17      - run: gcloud firestore export $BUCKET
```

Cron syntax has five fields separated by a space, and each field represents a unit of time.



Action Composition

```
name: "Publish to Docker"
description: "Pushes built artifacts to Docker"

inputs:
  registry_username:
    description: "Username for image registry"
    required: true
  registry_password:
    description: "Password for image registry"
    required: true

runs:
  using: "composite"
  steps:
    - uses: docker/setup-buildx-action@v1

    - uses: docker/login-action@v1
      with:
        username: ${{inputs.registry_username}}
        password: ${{inputs.registry_password}}

    - uses: docker/build-push-action@v2
      with:
        context: .
        push: true
        tags: user/app:latest
```

```
on: [push]

jobs:
  publish:
    runs-on: ubuntu-latest
    steps:
      - uses: actions/checkout@v2
      - uses: my-org/publish-docker@v1
        with:
          registry_username: ${{secrets.REGISTRY_USERNAME}}
          registry_password: ${{secrets.REGISTRY_PASSWORD}}
```

<https://github.blog/changelog/2021-08-25-github-actions-reduce-duplication-with-action-composition/>
<https://docs.github.com/en/actions/creating-actions/creating-a-composite-action>

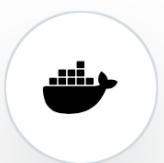
GitHub Actions marketplace

The screenshot shows the GitHub Marketplace search results for "Actions". The search bar at the top contains the query "sort:popularity-desc" and the sorting option "Sort: Most installed/starred". The left sidebar lists various categories: Types, Apps, Actions (selected), Categories, API management, Chat, Code quality, Code review, Continuous integration, Dependency management, Deployment, Deployment Protection Rules, IDEs, Learning, Localization, Mobile, Monitoring, Project management, Publishing, and Recently added. The main content area displays five GitHub Actions cards:

- TruffleHog OSS** by trufflesecurity: Creator verified by GitHub. Scan Github Actions with TruffleHog. 13.7k stars.
- yq - portable yaml processor** by mikefarah: create, read, update, delete, merge, validate and do more with yaml. 10.6k stars.
- Gosec Security Checker** by securego: Runs the gosec security checker. 7.4k stars.
- Metrics embed** by lowlighter: An infographics generator with 40+ plugins and 300+ options to display stats about your GitHub account. 12.3k stars.
- Super-Linter** by super-linter: Creator verified by GitHub. It is a simple combination of various linters, written in bash, to help validate your source code. 9.1k stars.
- OpenCommit — improve commits with AI** by di-sukharev: Replaces lame commit messages with meaningful AI-generated messages when you push to remote. 5.3k stars.

<https://github.com/marketplace?type=actions>

[Marketplace](#) / [Actions](#) / Build and push Docker images



GitHub Action

Build and push Docker images

v5.3.0

Latest version

release v5.3.0

GitHub Action to build and push Docker images with [Buildx](#) with full support of the features provided by [Moby BuildKit](#) builder toolkit. This includes multi-platform build, secrets, remote cache, etc. and different builder deployment/namespacing options.

```
/usr/bin/docker buildx build --file ./test/Dockerfile --iidfile /tmp/docker-build-push-0hL396/iidfile --label org.opencontainers.image.url=https://github.com/docker/build-push-action --label org.opencontainers.image.source=https://github.com/docker/build-push-action --label org.opencontainers.image.revision=8c76bb76c29ef1a323204893a39e7e0413e831d3 --label org.opencontainers.image.licenses=/metadata-file --push ./test

#1 [internal] load build definition from Dockerfile
#1 transferring dockerfile: 74B done
#1 DONE 0.0s

#2 [internal] load .dockerignore
#2 transferring context: 2B done
#2 DONE 0.0s

#3 [internal] load metadata for docker.io/library/alpine:latest
#3 DONE 0.1s

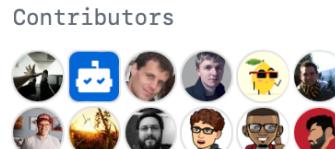
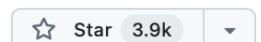
#4 [1/2] FROM docker.io/library/alpine@sha256:21a3deaa0d32a8057914f36584b5288d2e5ecc984380bc0118285c70fa8c9300
#4 resolve docker.io/library/alpine@sha256:21a3deaa0d32a8057914f36584b5288d2e5ecc984380bc0118285c70fa8c9300 done
#4 DONE 0.0s

#5 [2/2] RUN echo "Hello world!"
#5 CACHED
```

Use latest version



[Learn more about verified Actions.](#)



Categories

Continuous integration

Container CI

Links

⊕ docker/build-push-action

• Open issues

➡ Pull requests

```
name: ci

on:
  push:
    branches:
      - 'main'

jobs:
  docker:
    runs-on: ubuntu-latest
    steps:
      -
        name: Set up QEMU
        uses: docker/setup-qemu-action@v3
      -
        name: Set up Docker Buildx
        uses: docker/setup-buildx-action@v3
      -
        name: Login to Docker Hub
        uses: docker/login-action@v3
        with:
          username: ${{ secrets.DOCKERHUB_USERNAME }}
          password: ${{ secrets.DOCKERHUB_TOKEN }}
      -
        name: Build and push
        uses: docker/build-push-action@v5
        with:
          push: true
          tags: user/app:latest
```



Some other examples

- <https://github.com/cucpbioinfo/PTTEP-BIF-V3-Frontend/blob/dev/.github/workflows/deploy-docker-image.yml>
- <https://github.com/cucpbioinfo/PTTEPBIF-V3-oapi-codegen/tree/main/.github/workflows>
- <https://github.com/ProtEngPlus/proteng-kubeflow/blob/main/.github/workflows/build-push-consumer.yaml>



YAML validators

- <https://yamllint.readthedocs.io/en/stable/>
- <https://codebeautify.org/yaml-validator> (online)
- <https://www.yamllint.com/> (online)



What else?



GitHub Actions เป็น server less
ในขณะที่ Jenkins ต้องมี server

Jenkins

Build great things at any scale

The leading open source automation server, Jenkins provides ~~hundreds of plugins to~~ support building, deploying and automating any project.

We stand with the people of Ukraine. Please assist humanitarian efforts for the Ukrainian people and those affected by the military invasion of Ukraine by supporting international aid organizations, including the International Committee of the Red Cross

Documentation

Download

Jenkins Stories!

We are looking for experiences of Jenkins users from around the world showcasing how they are building, deploying, and automating great software with Jenkins. Check out their user stories and share yours



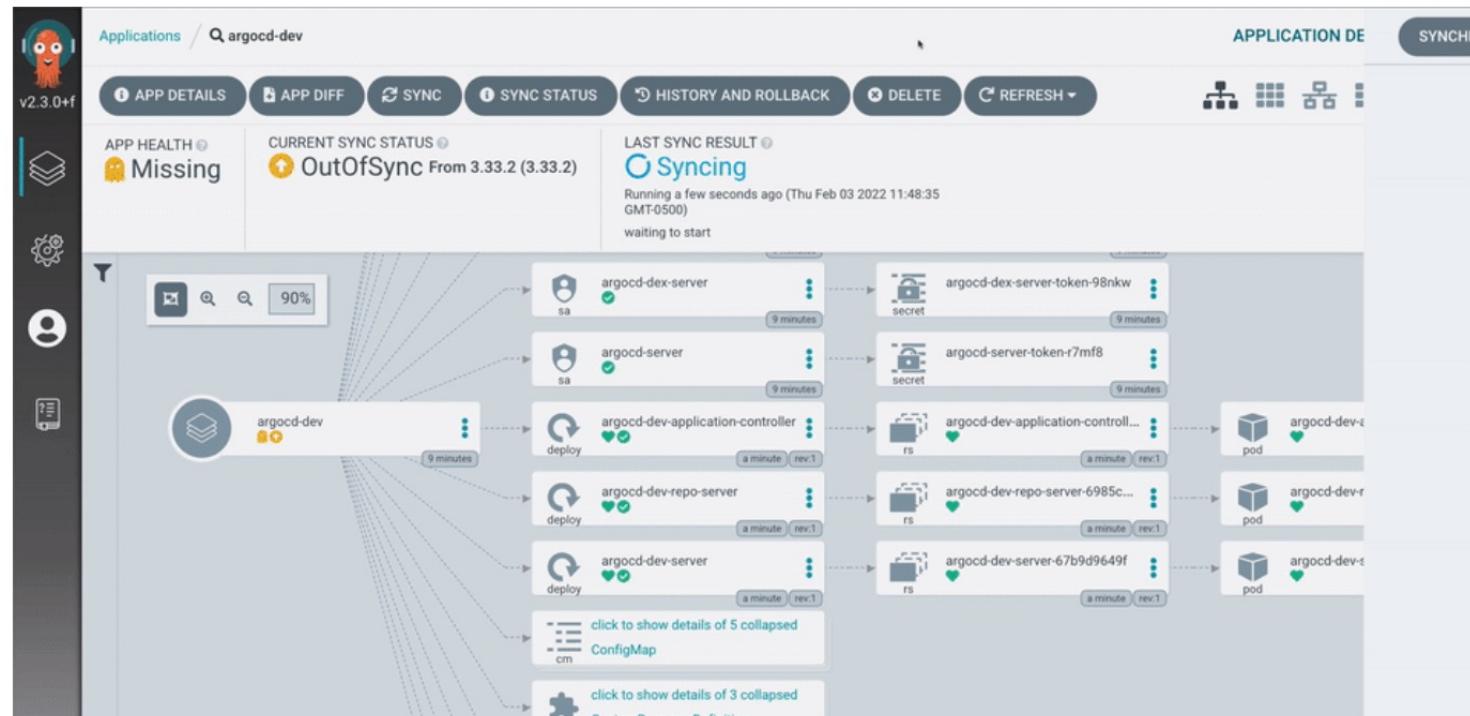
**Argo CD - Declarative GitOps CD for Kubernetes**[Overview](#)[Understand The Basics](#)[Core Concepts](#)[Getting Started](#)[Operator Manual](#)[User Guide](#)[Developer Guide](#)[FAQ](#)[Security Considerations](#)[Support](#)[Roadmap](#)[Releases](#) ▾[Blog](#) ▾

Overview

**Table of contents**[What Is Argo CD?](#)[Why Argo CD?](#)[Getting Started](#)[Quick Start](#)[How it works](#)[Architecture](#)[Features](#)[Development Status](#)[Adoption](#)

What Is Argo CD?

Argo CD is a declarative, GitOps continuous delivery tool for Kubernetes.



Argo CD Architecture

Codes /
Environment configurations (Manifest files)

