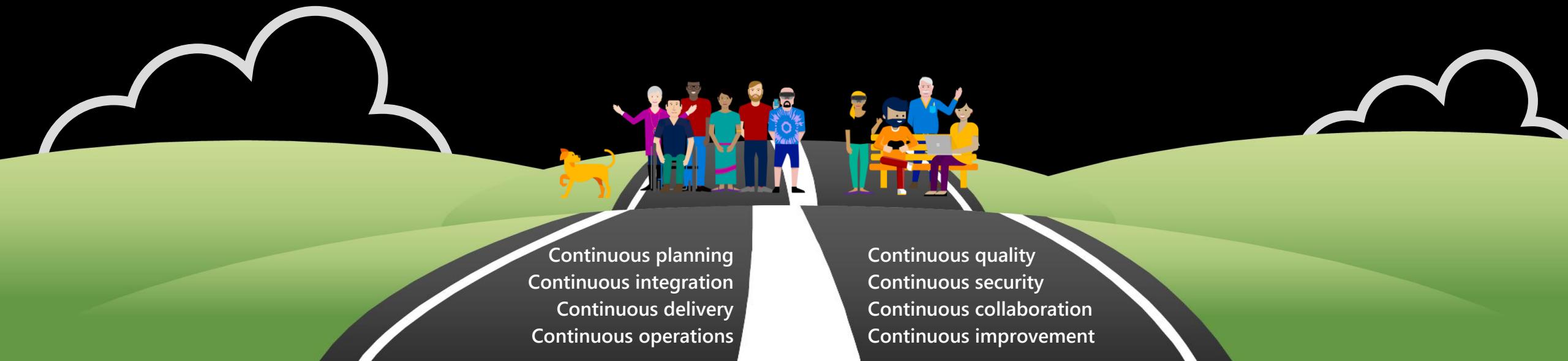
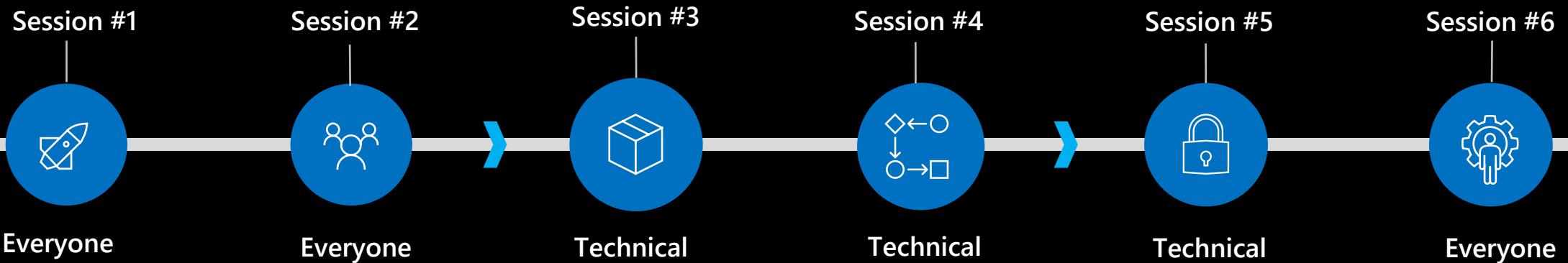
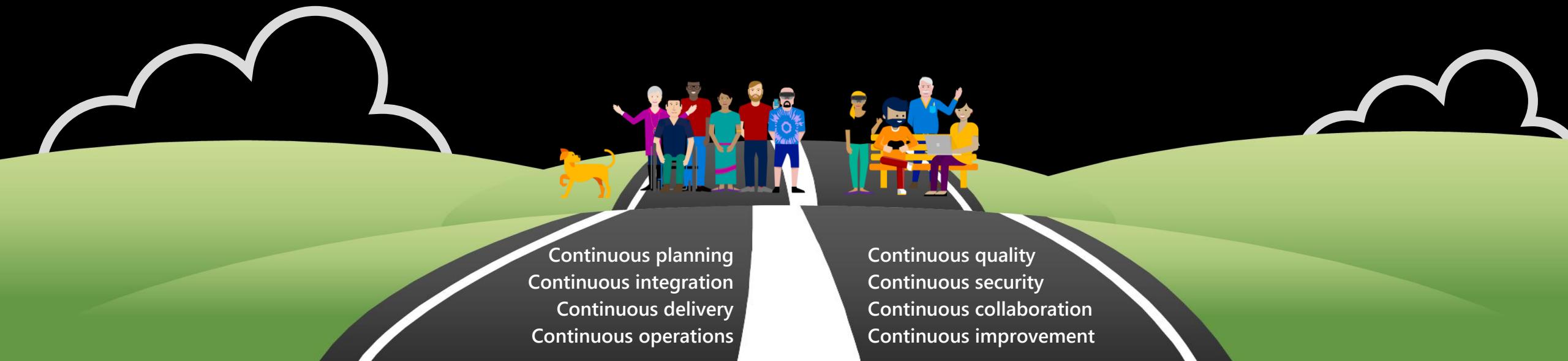
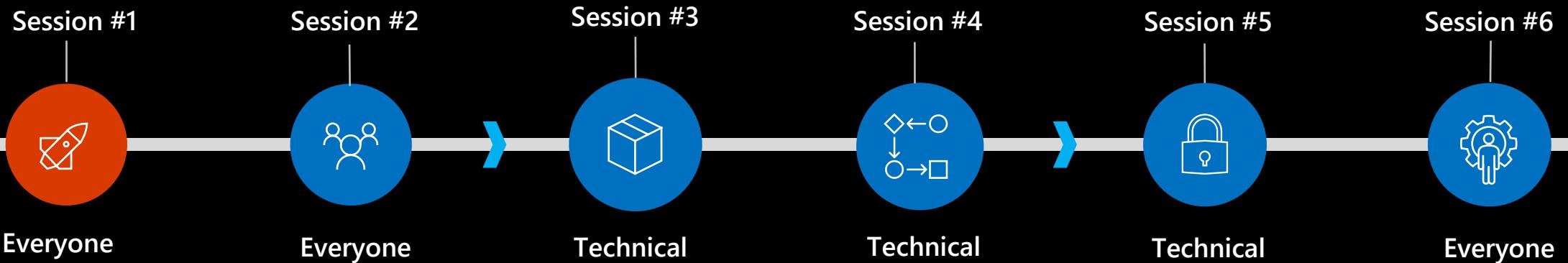


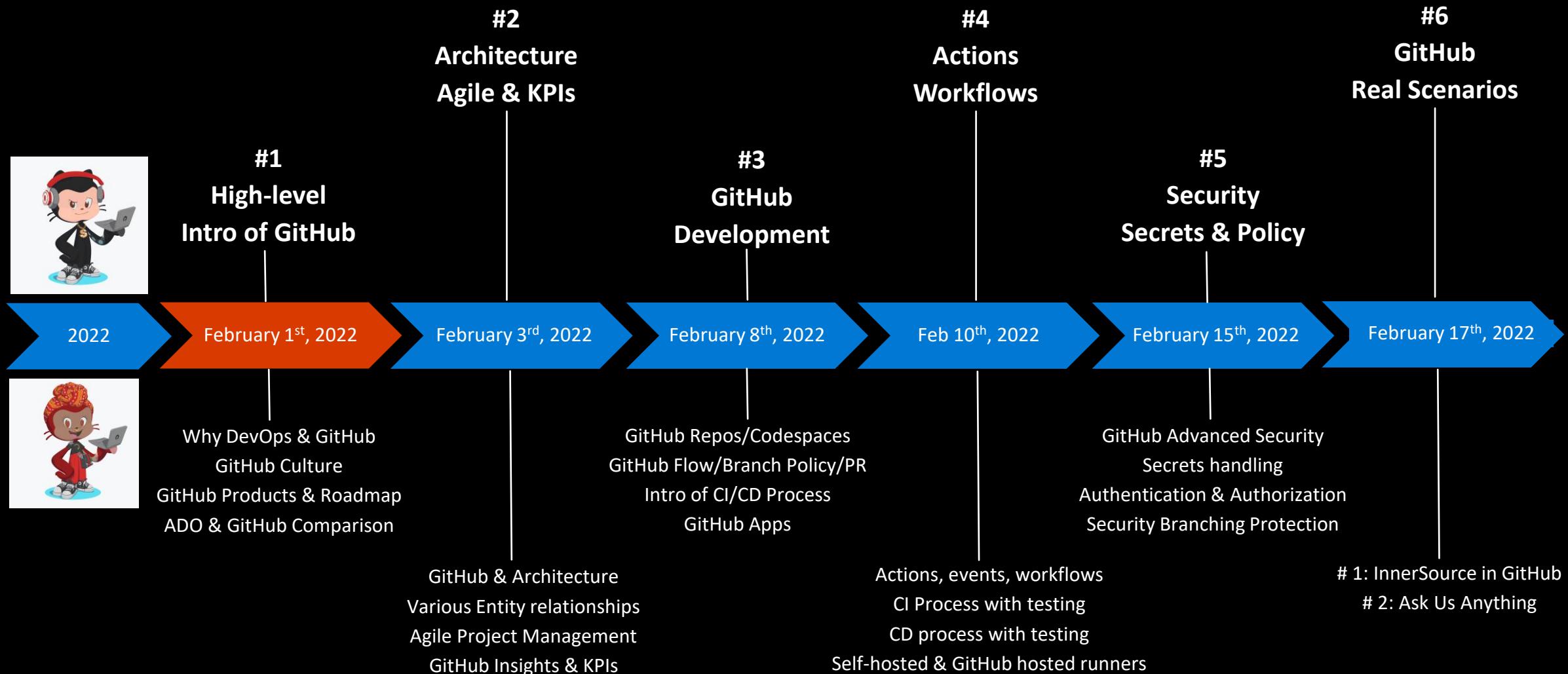
GitHub Bootcamp By DevOps Dojo



GitHub Bootcamp By DevOps Dojo



GitHub Bootcamp – February 1st, 2022



The Topics of the Day – Session #1

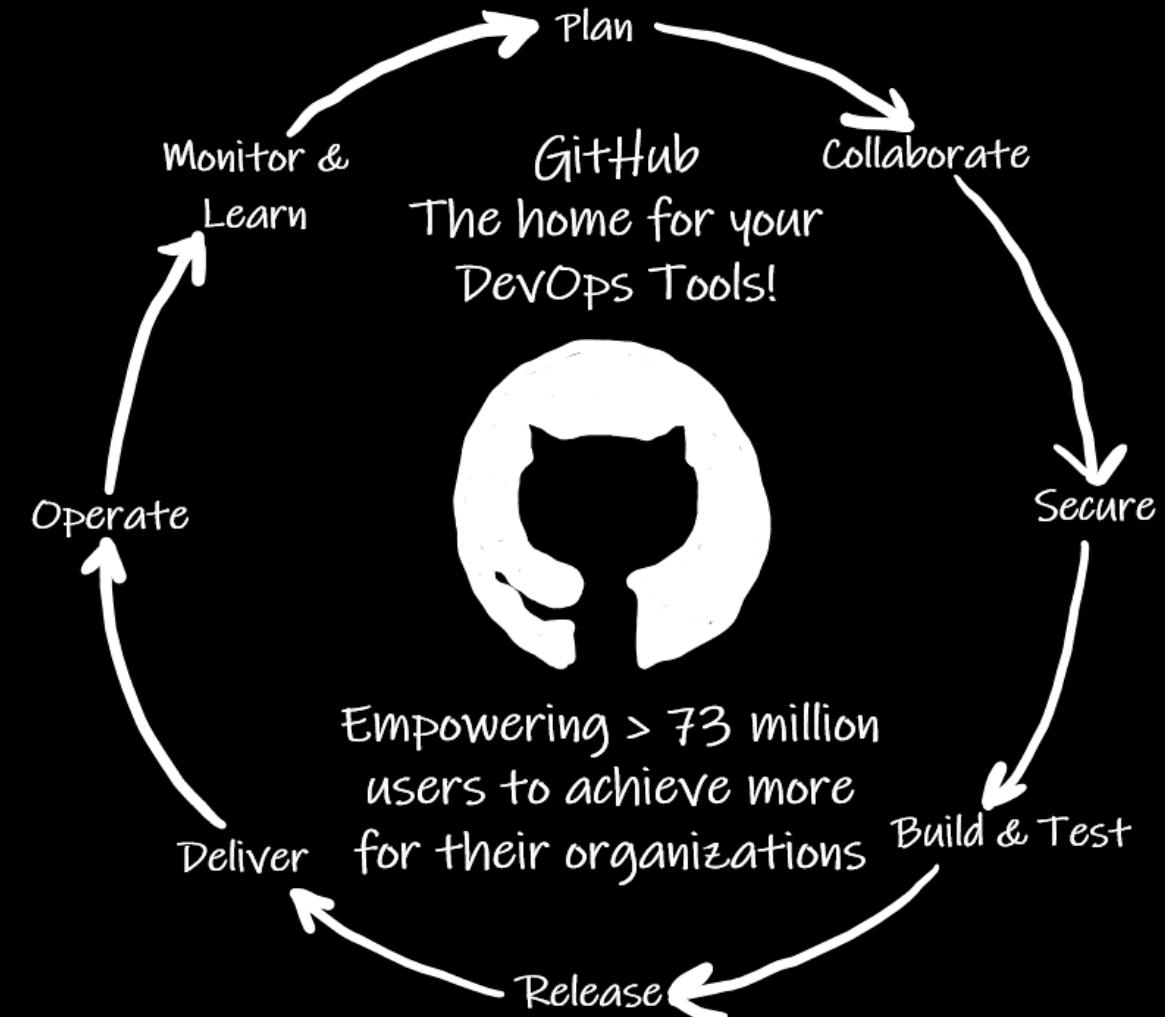
1 Why DevOps & GitHub

2 GitHub Culture

3 GitHub Products & Roadmap

4 Azure DevOps & GitHub Comparison

5 Questions



[Margarita Sanz]



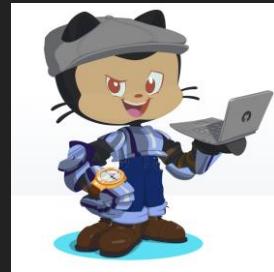
[Deep Mehta]



[Harleen Kaur]



[Nithyanathan R]



[Charlie Gu]



[Yue Sheng]



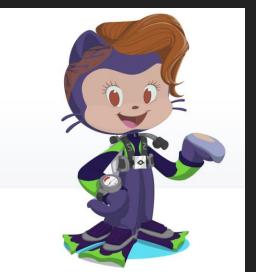
[Garry Trinder]



[Dave Burnison]



[April Edwards]



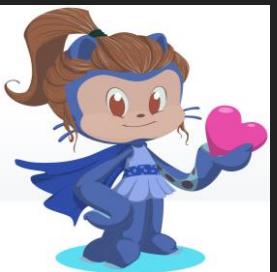
[Pierre Donyegro]



[Kan Tang]



[Beste Altinay]



[Ronghua Chen]



[Giulia Cupani]



[Kory Tuyo]



[Piyush Jain]



[Aakanksha]



[Abdeslam M]



WHO WE ARE

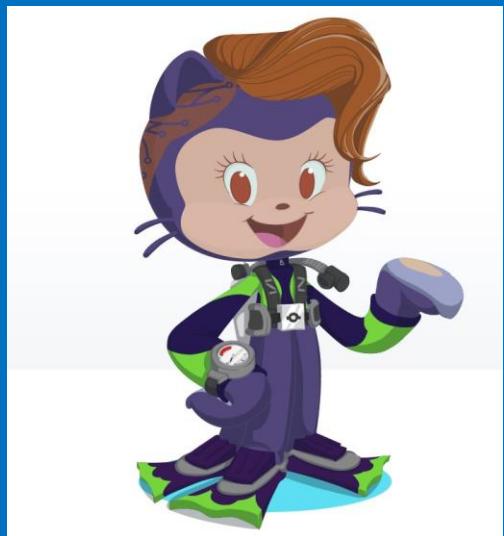
GitHub Subject Matter Experts

- **Dave Burnison**
- **Dave McKinstry**
- **James Garcia**
- **Bryton Herlong**
- **Mickey Gousset**
- **Briana Swift**



Session #1 Presenters

[April Edwards]



[Garry Trinder]



[Pierre Donyegro]



[Kan Tang]





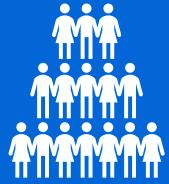
Presented by [April]

Why DevOps & GitHub

GitHub & Dojo Presentation 2022 (15 minutes)

Why GitHub?

#1



**Brand & OSS &
Innersource &
Community**

#2



**Developer
Platform &
Innovation**

#3



**Security
Platform &
Features**

#4



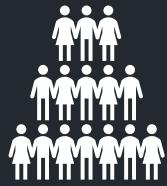
**Microsoft
Strategy on
1ES & GitHub**

GitHub Culture



Why GitHub #1 – Brand, OSS, Innersource & Community

#1



**Brand & OSS &
Innersource &
Community**

#2



**Developer
Platform &
Innovation**

#3



**Security
Platform &
Features**

#4



**Microsoft
Strategy on
1ES & GitHub**

DevOps Culture



GitHub is the largest developer community on Earth

73M+

developers

200M+

private and public
repositories

1,000s

top open source
communities

2.6B+

contributions
per year

4M+

organizations

84%

Fortune 500
companies

The most innovative companies



coinbase



NETFLIX

shopify

slack



stripe

twilio

Uber

The most open source projects



APACHE
HTTP SERVER PROJECT



GraphQL



npm

TensorFlow

Innersource with GitHub

Enabling open-source culture and best practices inside your organization

Increased collaboration

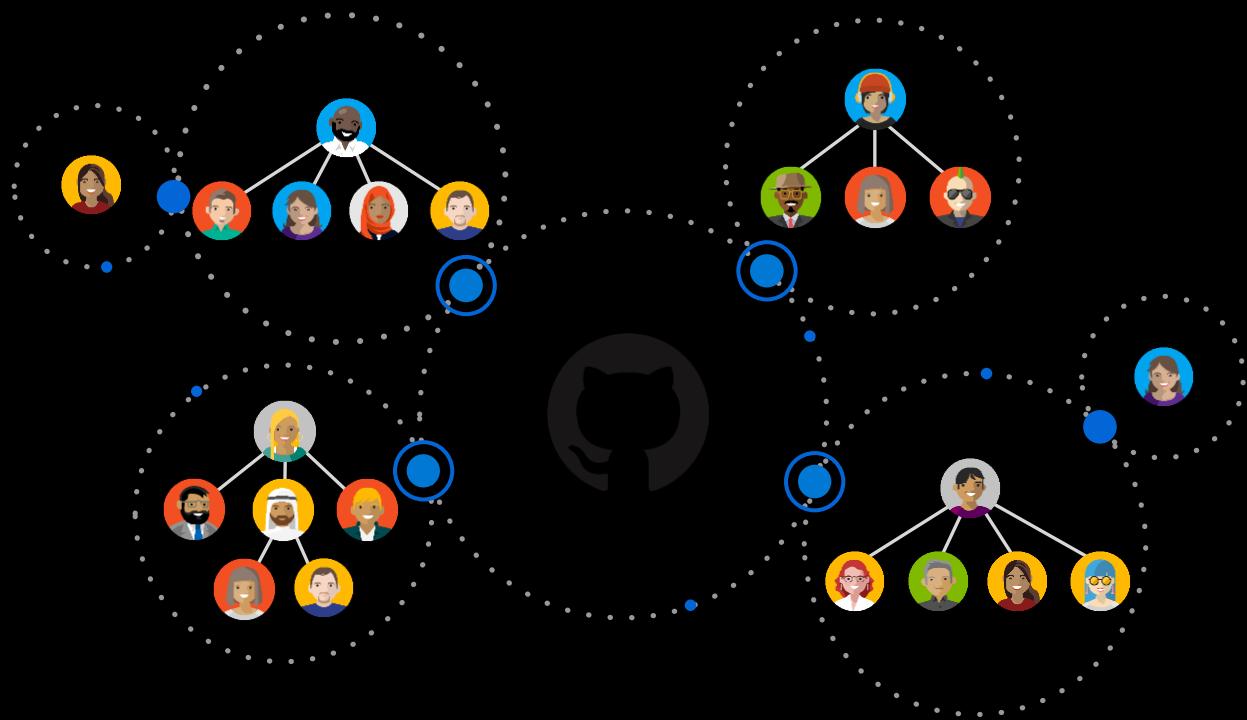
Encourage teams to collaborate within your organization using the same processes and practices as open-source communities

Breaking silos

Simplified collaboration across teams, sharing of knowledge, improved code reuse, and secured workflows

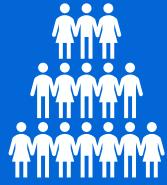
Higher developer satisfaction

Leveraging inner source and open-source practices increases developers' satisfaction, enabling them to work on interest projects and increase their skills



Why GitHub #2 – Developers & Innovation

#1



**Brand & OSS &
Innersource &
Community**

#2



**Developer
Platform &
Innovation**

#3



**Security
Platform &
Features**

#4



**Microsoft
Strategy on
1ES & GitHub**

GitHub Culture



GitHub



Collaboration

Attract and retain the best talent
with tools developers know and love



Security

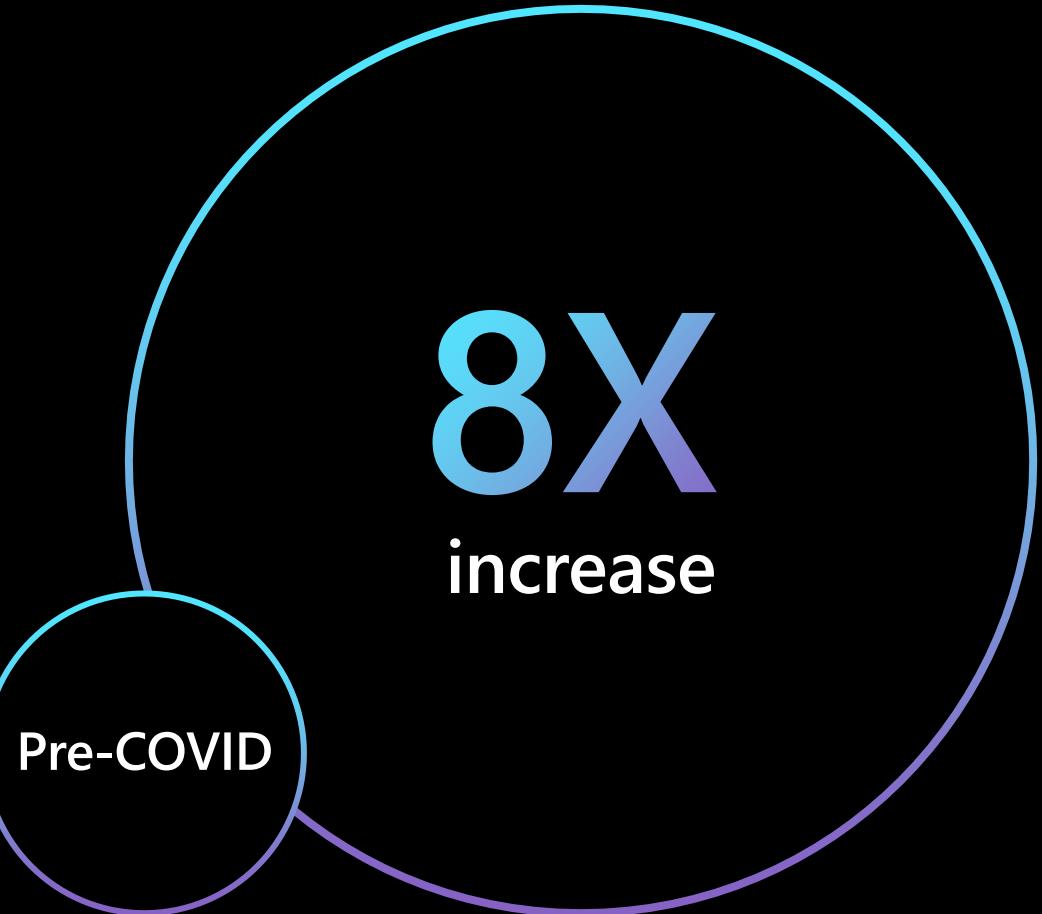
Securely integrate open source
code in your software



DevOps

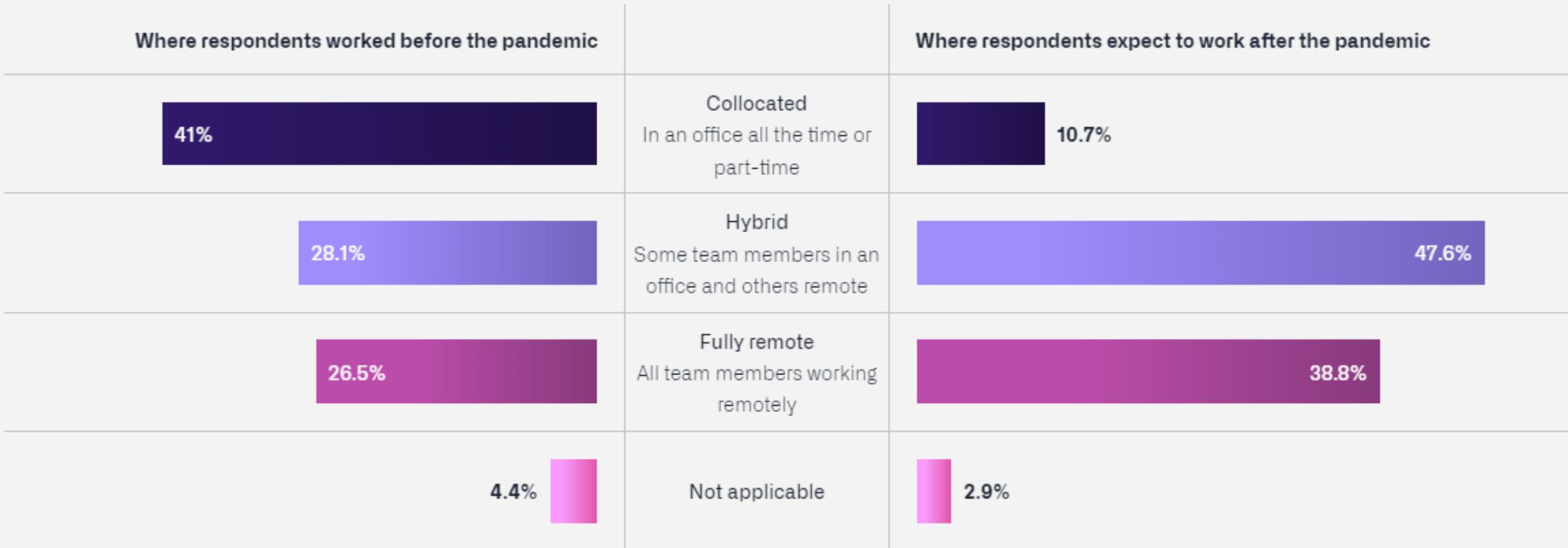
Automate workflows
from code to any cloud

Remote job postings for developers



May 2019–Apr 2020 vs. May 2020–Apr 2021
Source: LinkedIn

Work before and after the pandemic

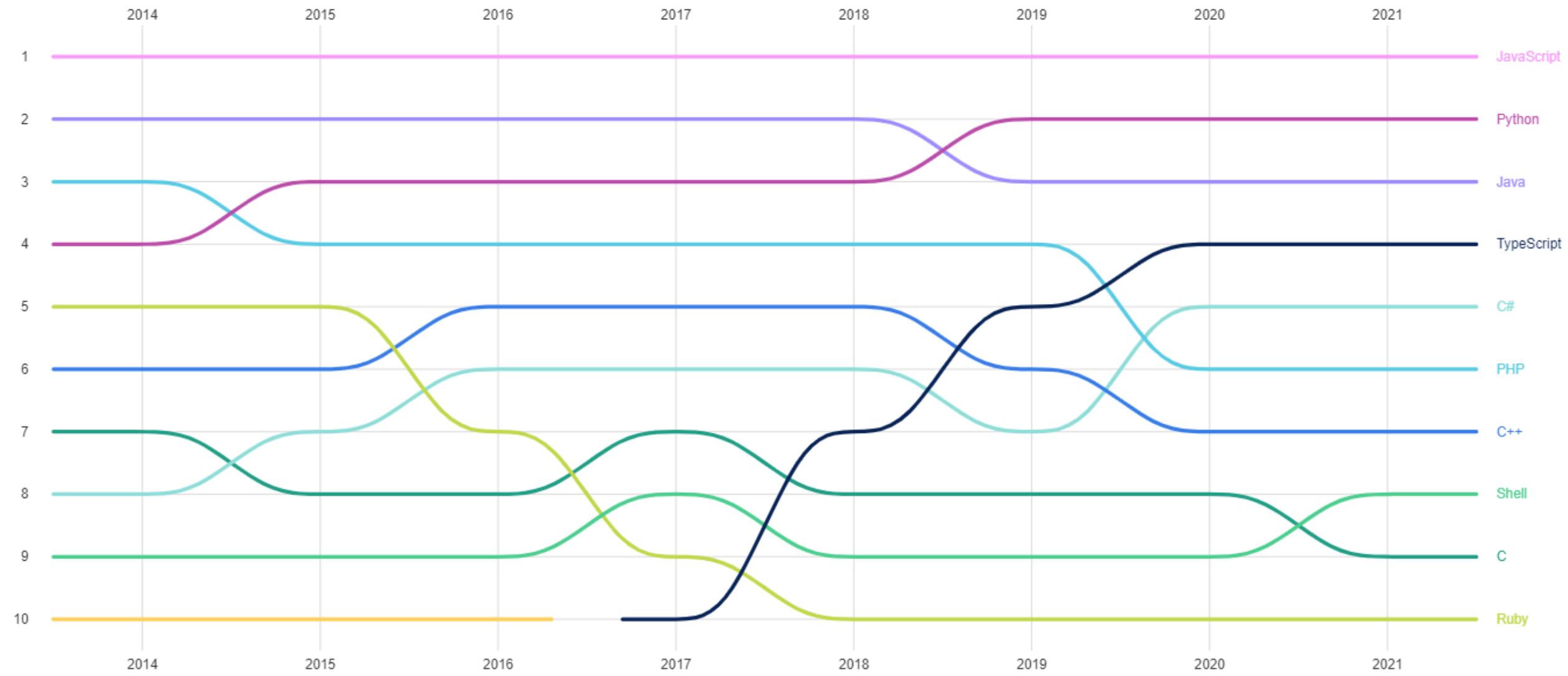


Monthly Engaged Users on GitHub & ADO



Source: GitHub & Microsoft

Top languages over the years



Why GitHub #3 – Security Platform & Features

#1



**Brand & OSS &
Innersource &
Community**

#2



**Developer
Platform &
Innovation**

#3



**Security
Platform &
Features**

#4

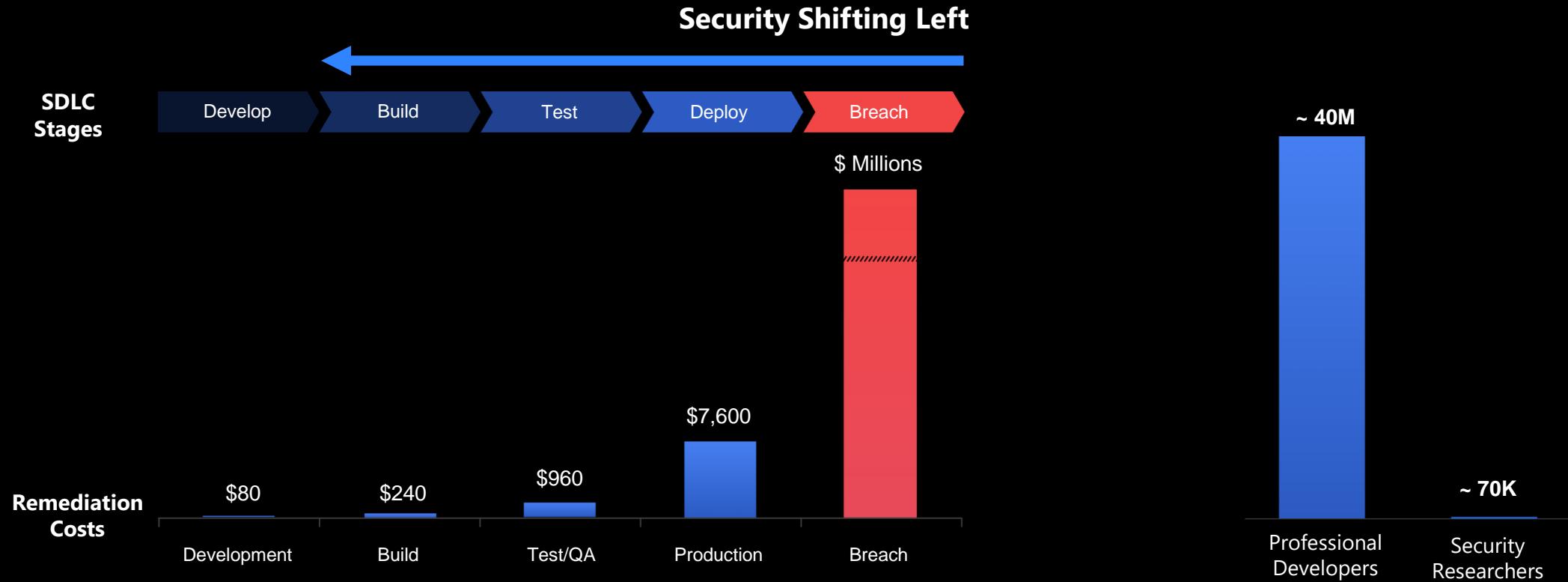


**Microsoft
Strategy on
1ES & GitHub**

GitHub Culture



Shift security left with GitHub Advanced Security



Vastly more cost effective to remediate during development

570X more developers than security researchers

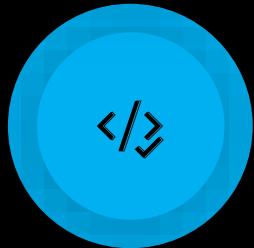


GitHub Advanced Security



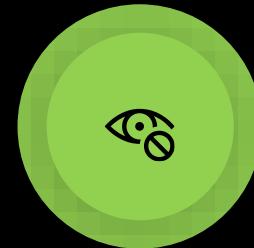
Dependency scanning

Integrated review of dependencies alerts and security updates



Code scanning

Extensible framework for code scanning integrated within the developer workflow. Backed by industry-leading CodeQL engine



Secret scanning

Scanning for leaked secrets in public and private repos partnership with 40+ providers

Security at every step of the DevOps lifecycle

Pre-commit

Threat modeling

Secure coding standards

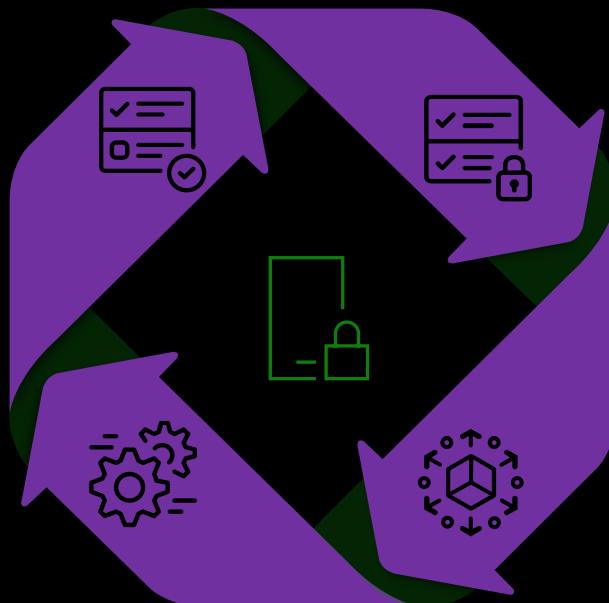
Peer review

Operate and monitor

Continuous monitoring

Threat intelligence

Blameless postmortems



Commit (CI)

Static code analysis

Dependency management

Credential scanning

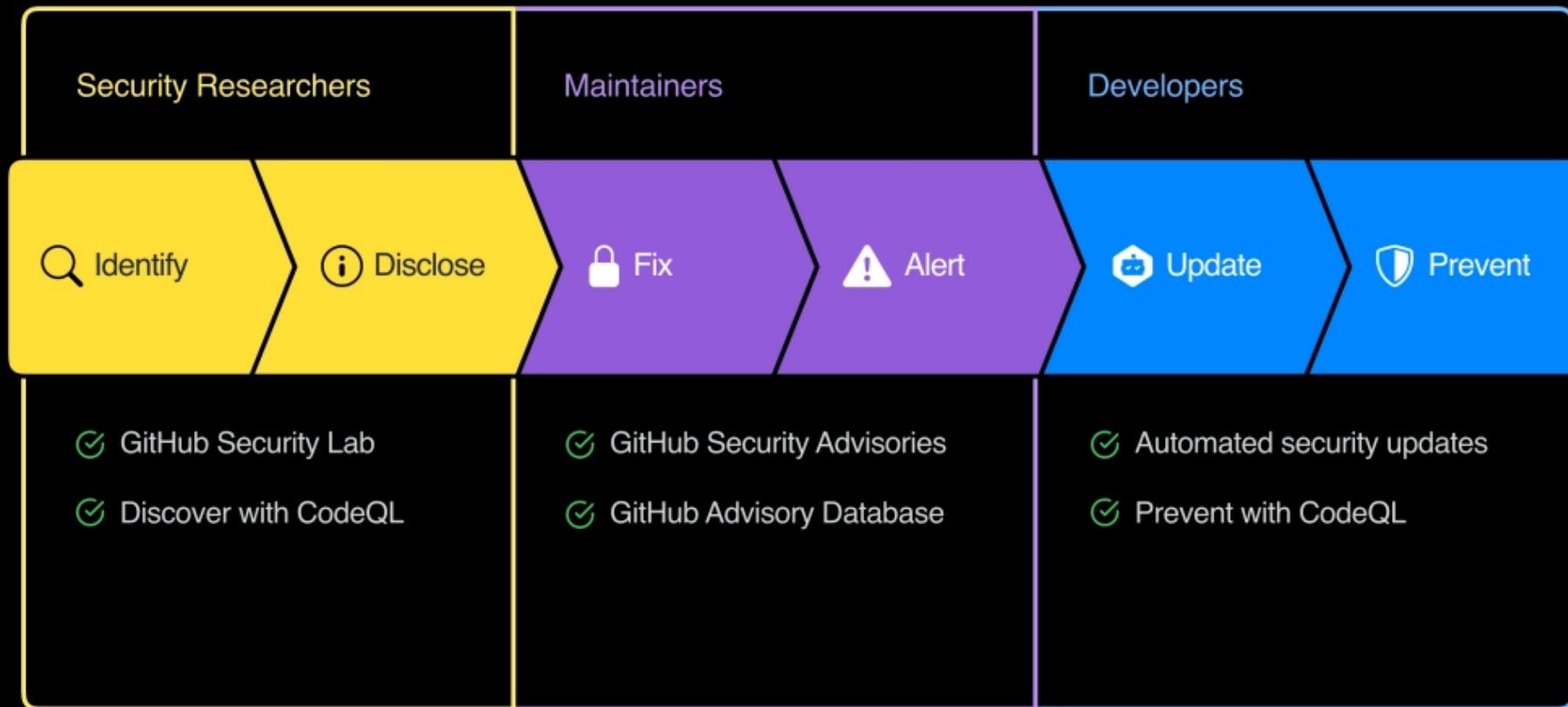
Deploy (CD)

Infra as code (IaC) scanning

Cloud configuration checks

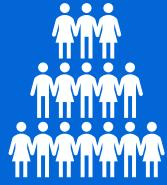
Security acceptance tests

Security functions by personas



Why GitHub #4 – Microsoft Investment

#1



Brand & OSS &
Innersource &
Community

#2



Developer
Platform &
Innovation

#3



Security
Platform &
Features

#4



Microsoft
Strategy on
1ES & GitHub

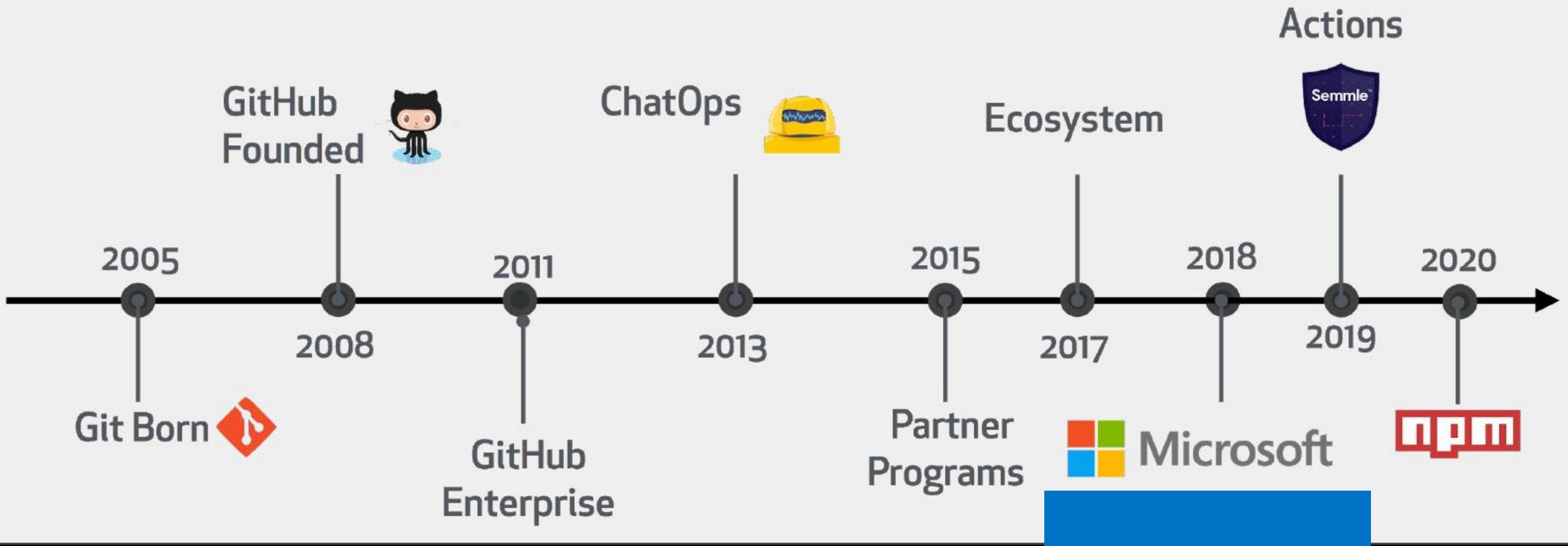
GitHub Culture



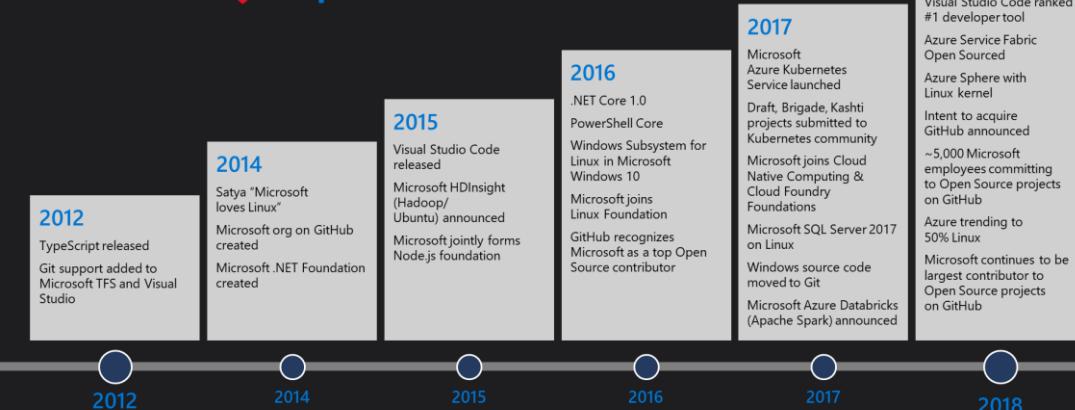


“...one of the things GitHub does is it
drives the agility of your
development...**GitHub is a way of life.**”

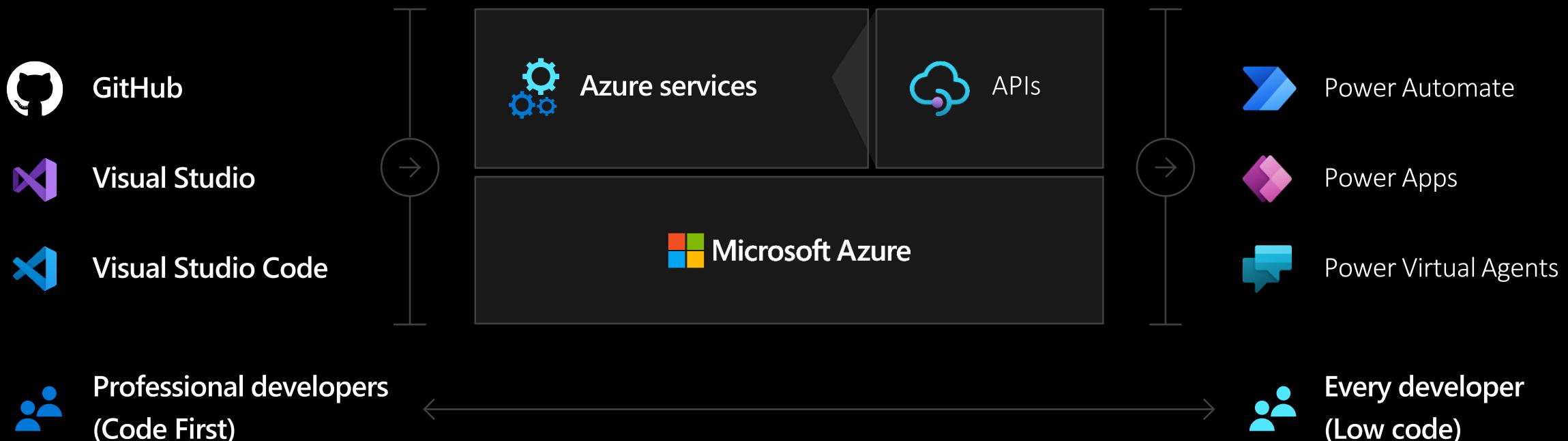
Satya Nadella
Chief Executive Officer
Microsoft Corporation



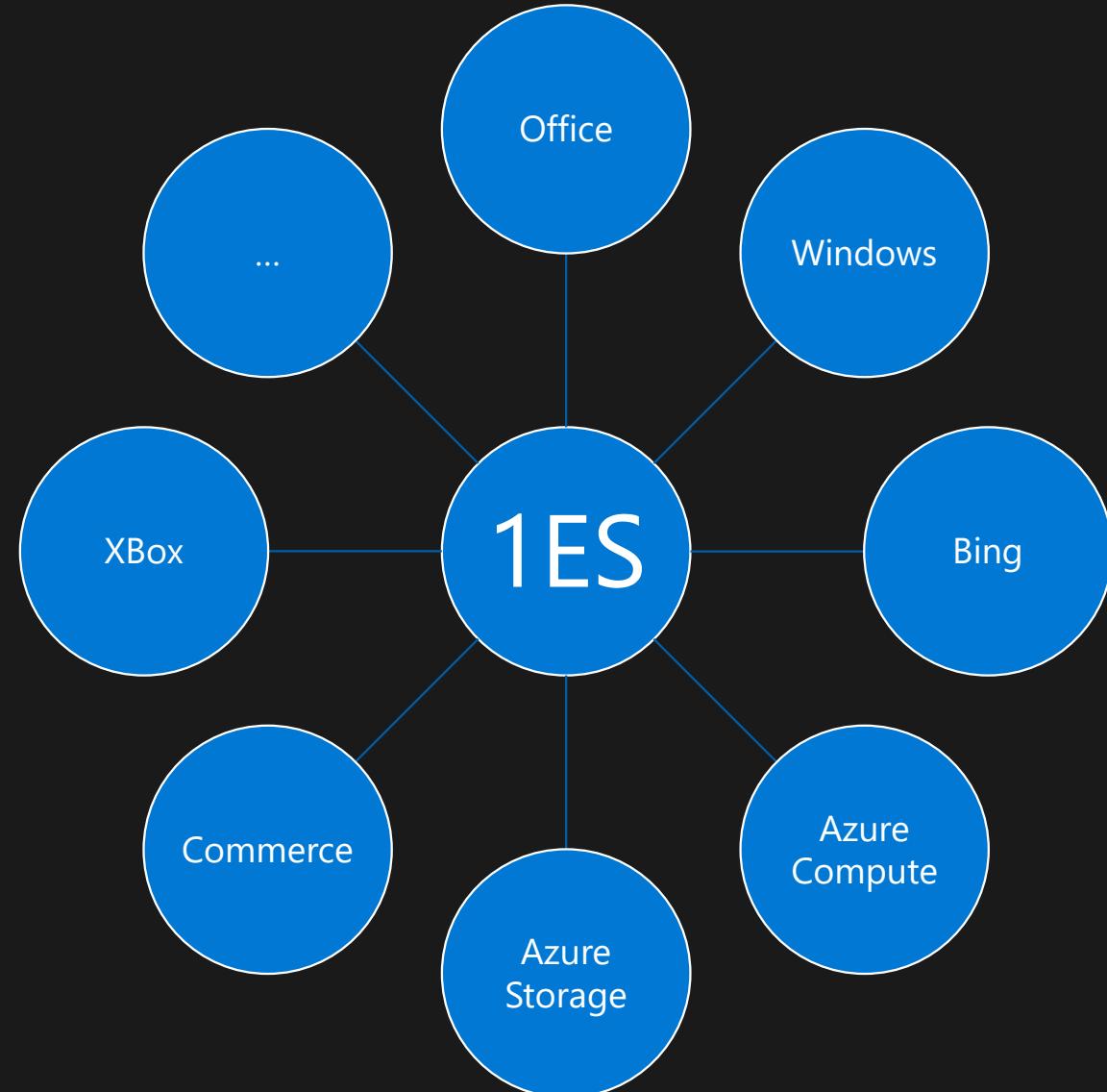
Microsoft ❤️ Open Source



Complete toolchain for Developers



1ES & Engineering System Landscape at MSFT



1ES on GitHub

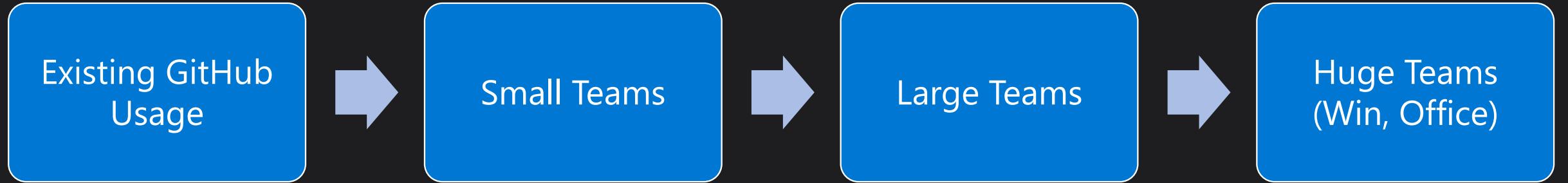


Karl Piteira
1ES PRINCIPAL GROUP
PM MANAGER

“

GitHub and Azure DevOps joined forces and are merging their efforts to make GitHub the best DevOps tool for the Enterprise. To achieve that goal, Microsoft will be the leading customer to use GitHub.

Using GitHub will be the opportunity to promote more collaboration through Inner-source software development practices and building on the wild success of GitHub in the Open-source.



GitHub inside Microsoft (GiM) Strategy

Experience Migration Sequence

Build a Secure and Compliant end-to-end experience

DevOps across Microsoft

→ <http://aka.ms/DevOps-Stories>

165K

Monthly Engaged
Users (MEU)

27+k

Repos: GitHub
& Azure Repos

10.1M

Builds per month

3.4M

Git commits
per month

373K

Pull Requests
completed per month

1.7M

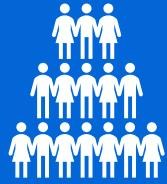
Work items
created per month

5,700,000

Releases in August, 2021

Why does GitHub matter to YOU?

#1



**Brand & OSS &
Innersource &
Community**

#2



**Developer
Platform &
Innovation**

#3



**Security
Platform &
Features**

#4



**Microsoft
Strategy & App
Innovate & 1ES**

GitHub Culture





Presented by [Garry]

GitHub Culture

GitHub & Dojo Presentation 2022 (15 minutes)

Speed of change

Technology

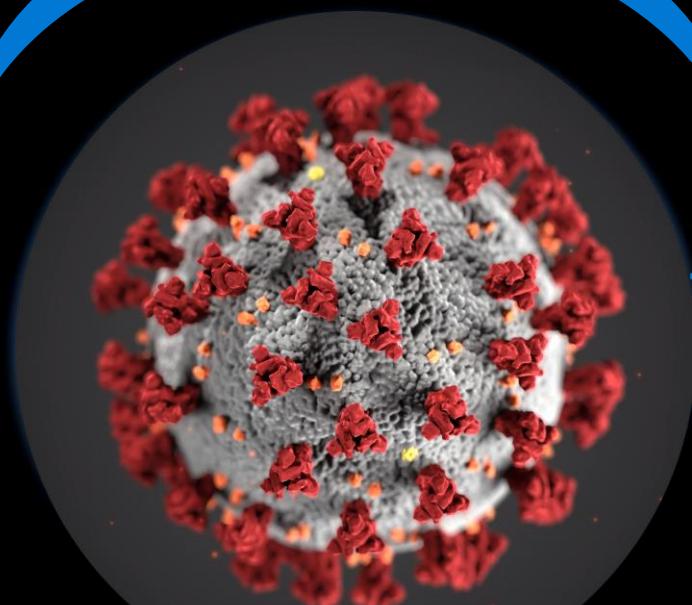
Commerce

Infrastructure

Governance

Culture

Nature

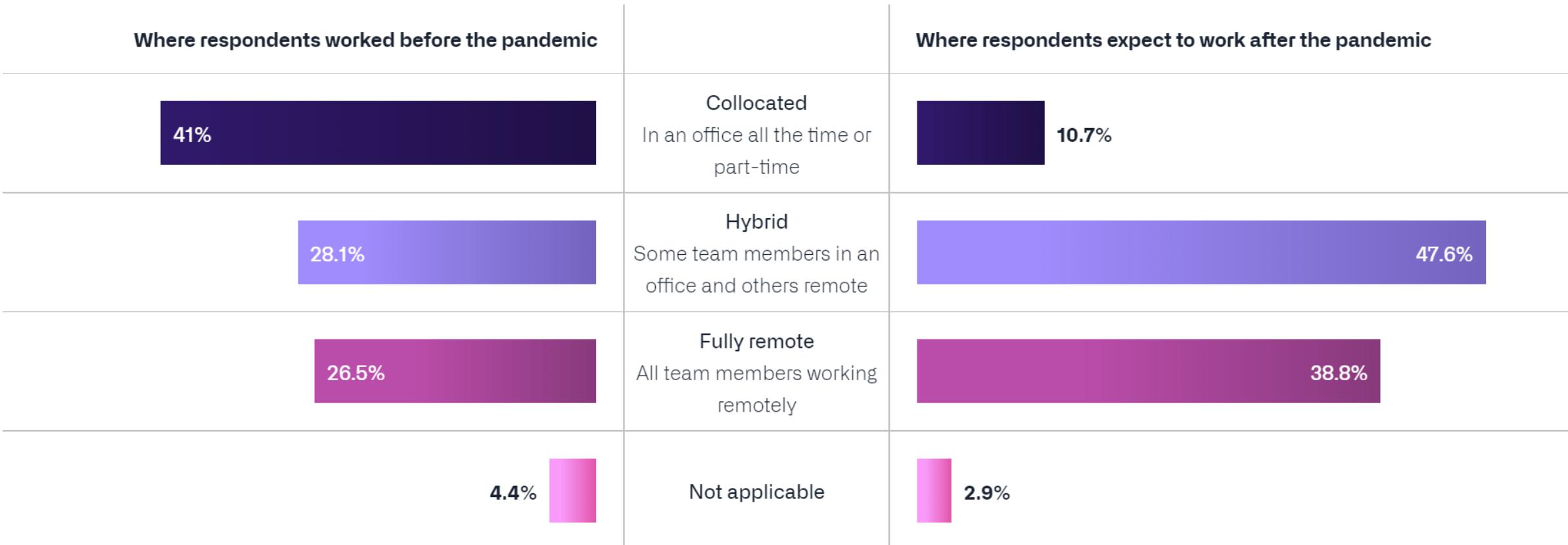


Work is shifting

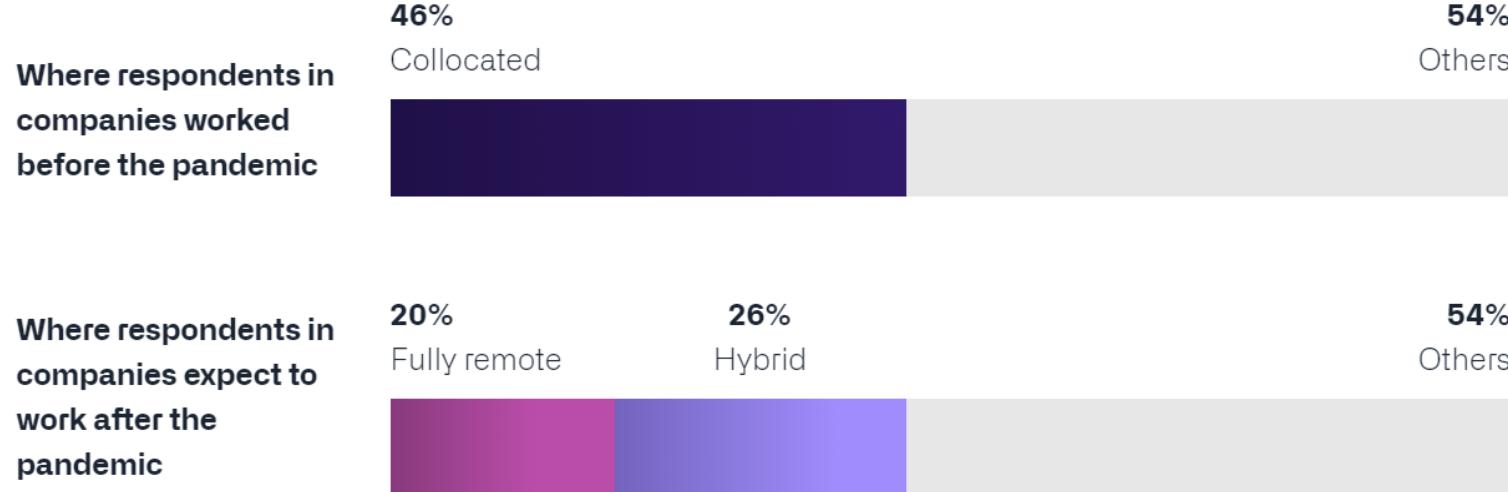
Work before and after the pandemic



We present one decimal for simplicity; there may be a rounding difference of 1%.



Work is shifting



When drilling down into the data we found the difference was more pronounced for folks in companies. Of those respondents working in companies, 46% who previously worked collocated now expect to work in a fully remote (20%) or hybrid (26%) environment.

Westrum Culture & High Performing Teams

Pathological



- Low cooperation
- Messengers shot
- Responsibilities shirked
- Bridging discouraged
- Failure → scapegoating
- Novelty crushed

Bureaucratic



- Modest cooperation
- Messengers neglected
- Narrow responsibilities
- Bridging tolerated
- Failure → justice
- Novelty problems

Generative



- High cooperation
- Messengers trained
- Risks are shared
- Bridging encouraged
- Failure → inquiry
- Novelty implemented

Role of culture



A positive team culture mitigates burnout during challenging circumstances.

From State of DevOps 2021 report

Team culture makes a large difference to a team's ability to deliver software and meet or exceed their organizational goals.



Inclusive teams with a generative culture experienced less burnout during the COVID-19 pandemic.

GitHub supports high performing teams



Collaborative
culture



Documentation,
documentation,
documentation ...



Automate all the
things!



Empowering
developers





Trust and respect contribute to a strong culture

Pull requests are how development teams coordinate and build software together

Collaborative culture

Teams with high trust are more likely to have a healthy collaborative culture -- 2x more likely in companies, and 3x more likely in open source.





**Documentation,
documentation,
documentation...**

Documentation is good for productivity and culture -- it's a win-win

Developers see about a 50% productivity boost when documentation is up-to-date, detailed, reliable, and comes in different formats (e.g., articles, videos, forums).

Document everything, if its not written down, it doesn't get done!

“The palest ink is better than the best memory.”





**Automate all
the things!**

Remove friction and repetitive tasks using GitHub Actions

Teams merge almost 2x more pull requests per day than before (61% increase) and they merge 31% faster using Actions.

Teams perform 27% better in open source and 43% better at work, and developers report higher fulfilment.

Good automation helps teams communicate better and more clearly

**Automation and community help us build better,
together.**





Empowering developers

Better tools helps developers feel empowered to do their work and feel fulfilled.

Developers are almost 60% more likely to feel equipped to do their job when they can easily find what they need.

Plus, they can get an 11% productivity bump simply by having a team repo that is easy to search

Reuse of code is easy and doesn't introduce friction. Developers' performance at work can increase by up to 87% when reusing others



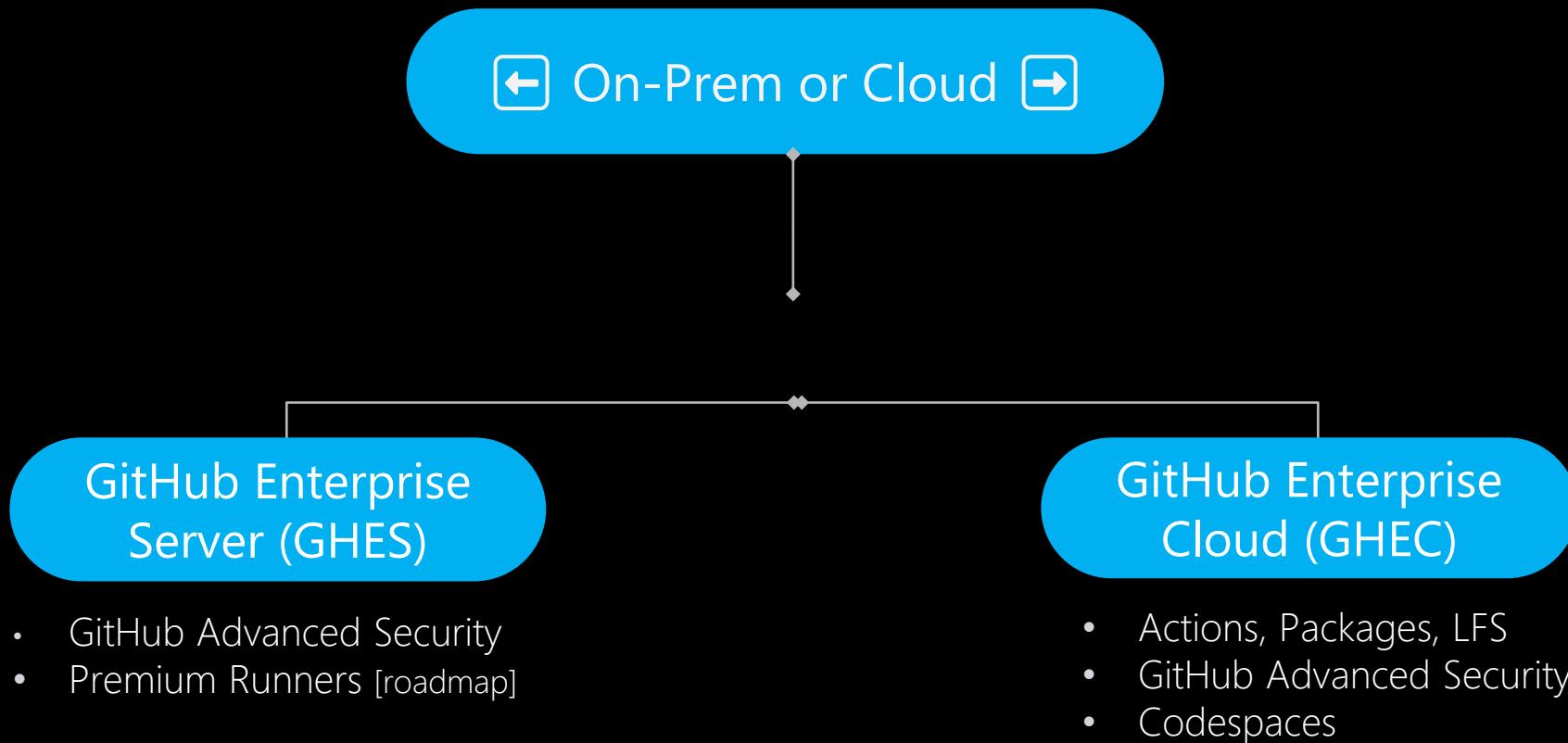


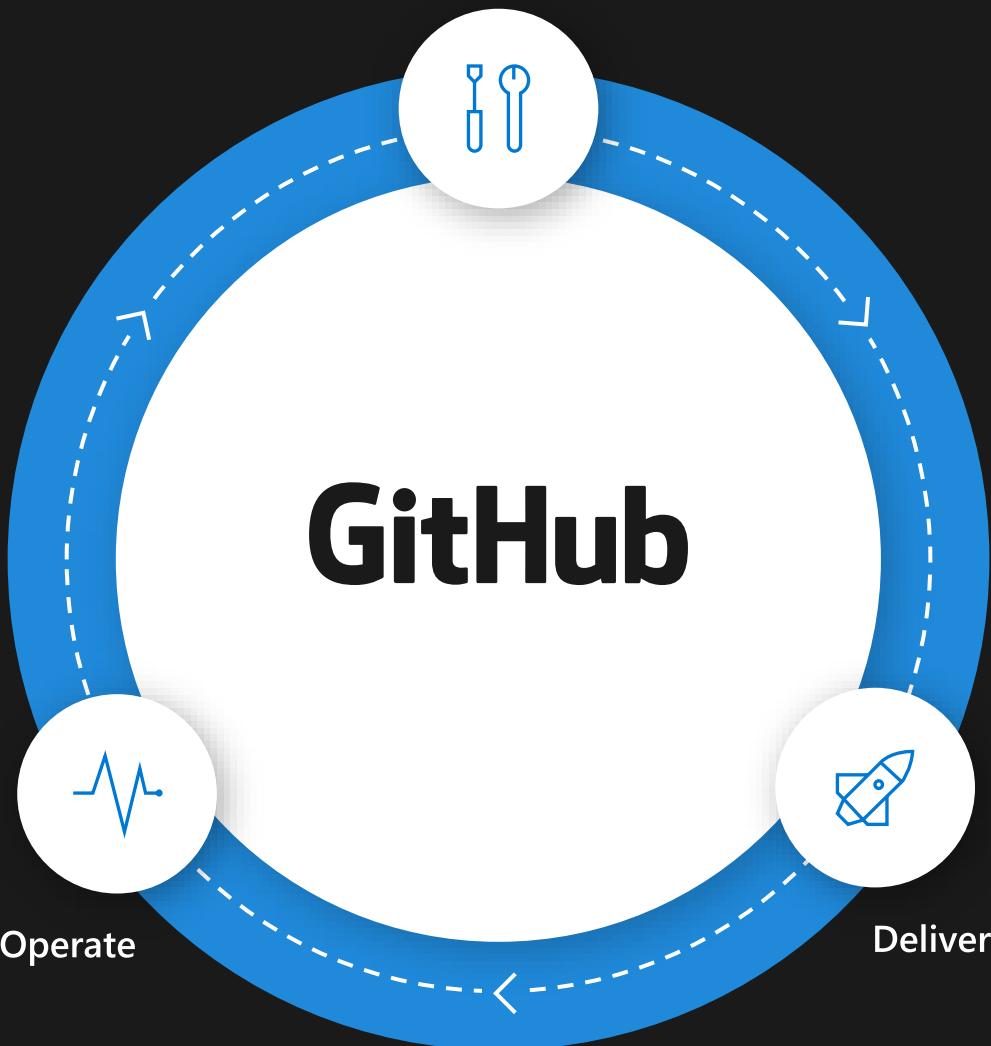
Presented by [Pierre]

GitHub Products & Roadmap

GitHub & Dojo Presentation 2022 (15 minutes)

Qualification Criteria





GitHub Enterprise Cloud

- GitHub Issues
- GitHub Projects
- GitHub Codespaces
- GitHub Copilot
- GitHub Repos
- GitHub Actions
- GitHub Packages
- GitHub Apps
- GitHub Advanced Security
 - GitHub Code Scanning
 - GitHub Secret Scanning
 - GitHub OSS Scan
- GitHub Mobile
- GitHub Discussions

Planning

Introducing the new GitHub Issues

Project planning for developers

Create issues, break them into tasks, track relationships, add custom fields, and have conversations. Visualize large projects as spreadsheets or boards, and automate everything with code.

Switch between Boards & Tables

Built like a spreadsheet, project tables give you a live canvas to filter, sort, and group issues and pull requests. Tailor them to your needs with custom fields and saved views.

Issues, wherever you look

Issues can be viewed, created, and managed in your browser, your favorite terminal, or on your phone or iPad.

[Introducing the new GitHub Issues](#) | [The GitHub Blog](#) &
[Issues Archives](#) | [The GitHub Blog](#)

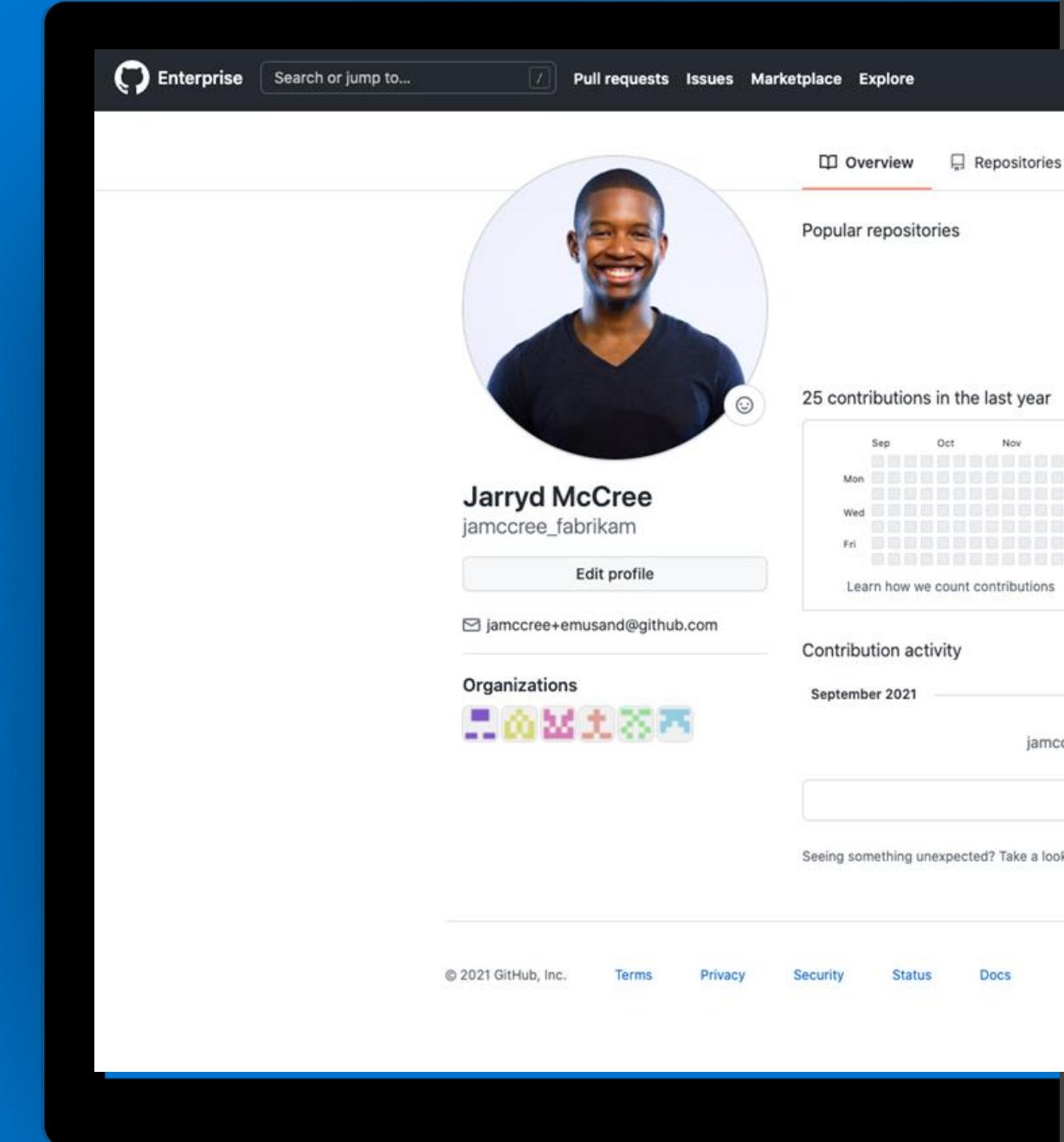
OctoArcade Invaders			
The Plan	Game loop Backlog	Standup	New view
Title	Team	Status	
Prototype 🛠️ 3			
1 Game brief and go-no-go	Producers 🎬	Comp	
2 Engine prototype (physics, rendering)	Engine ⚙️	Comp	
3 Initial concept art	Art 🌈	Comp	
+ Add item			
Beta 🌱 5			
4 Integrate with Leaderboard Service	Game Loop 📈	Not St	
5 Creative design update to aliens for variety	Art 🌈	Plann	
6 🚨 Updates to alien, beam, and cannon sprites	Art 🌈	Buildi	
7 🚨 Update to collision logic	Engine ⚙️	Buildi	
8 🚨 Improve alien respawn rate	Game Loop 📈	Behin	
+ Add item			
Launch 🚀 6			
9 Interviews with media outlets	Producers 🎬	Not St	
10 🚨 Save score across levels	Game Loop 📈	Not St	

Enterprise Managed Users

Enterprise owned user accounts in the GitHub Cloud

- Full lifecycle ownership of user accounts
- Automated account management and team syncing with SCIM
- SAML backed and fully auditable
- No public repositories

[Enterprise managed users are now generally available for GitHub Enterprise Cloud | The GitHub Blog](#)



Custom Repository Roles

Create your own repository access roles

- Users and teams can have just the repository permissions they need to do their jobs
- Roles govern access to repositories, but are created at the organization level
- 40 permissions available today, with more to come based on feedback

[Enterprise organizations can now create custom repository roles | GitHub Changelog](#)

The screenshot shows the GitHub interface for managing repository roles. On the left, a sidebar lists various account settings: Account settings, Profile, Billing & plans, Member privileges, Organization security, Security & analysis, **Repository roles** (which is selected), Verified & approved domains, Audit log, Sponsorship log, Webhooks, Third-party access, Installed GitHub Apps, Scheduled reminders, Repository topics, Repository defaults, Deleted repositories, and Projects. To the right, the main content area is titled "Repository roles / Edit role". It says "Roles are used to grant access and permissions for teams and members. Begin adding permissions to create a role that fits your needs. [Learn more](#)". There is a "Name" field containing "Security Engineer", a "Description" field containing "To triage security scan results", and a note "A short description who this role is for or what permissions it grants". Below this, there is a section titled "Choose a role to inherit" with the note "All custom roles must inherit the permissions of a default role." A radio button for "Read" is unselected, while one for "Triage" is selected (indicated by a blue border). Other options include "Write" and "Maintain". Under "Add Permissions", there is a search bar with "code" typed in. A list of permissions follows: "Read code scanning results" (unchecked), "Write code scanning results" (unchecked), "Delete code scanning results" (unchecked), and "Assign or remove a user (inherited from Triage)". Below this is another section for "Issue Permissions" with the option "Close an issue (inherited from Triage)".

Codespaces

Your instant dev environment

Code without compromise

Code, build, test, debug, and deploy with a complete development environment in your browser.

Contribute to repos quickly

Reduce the time to set up dev environments from hours per week to a couple of minutes.

Extend and customize

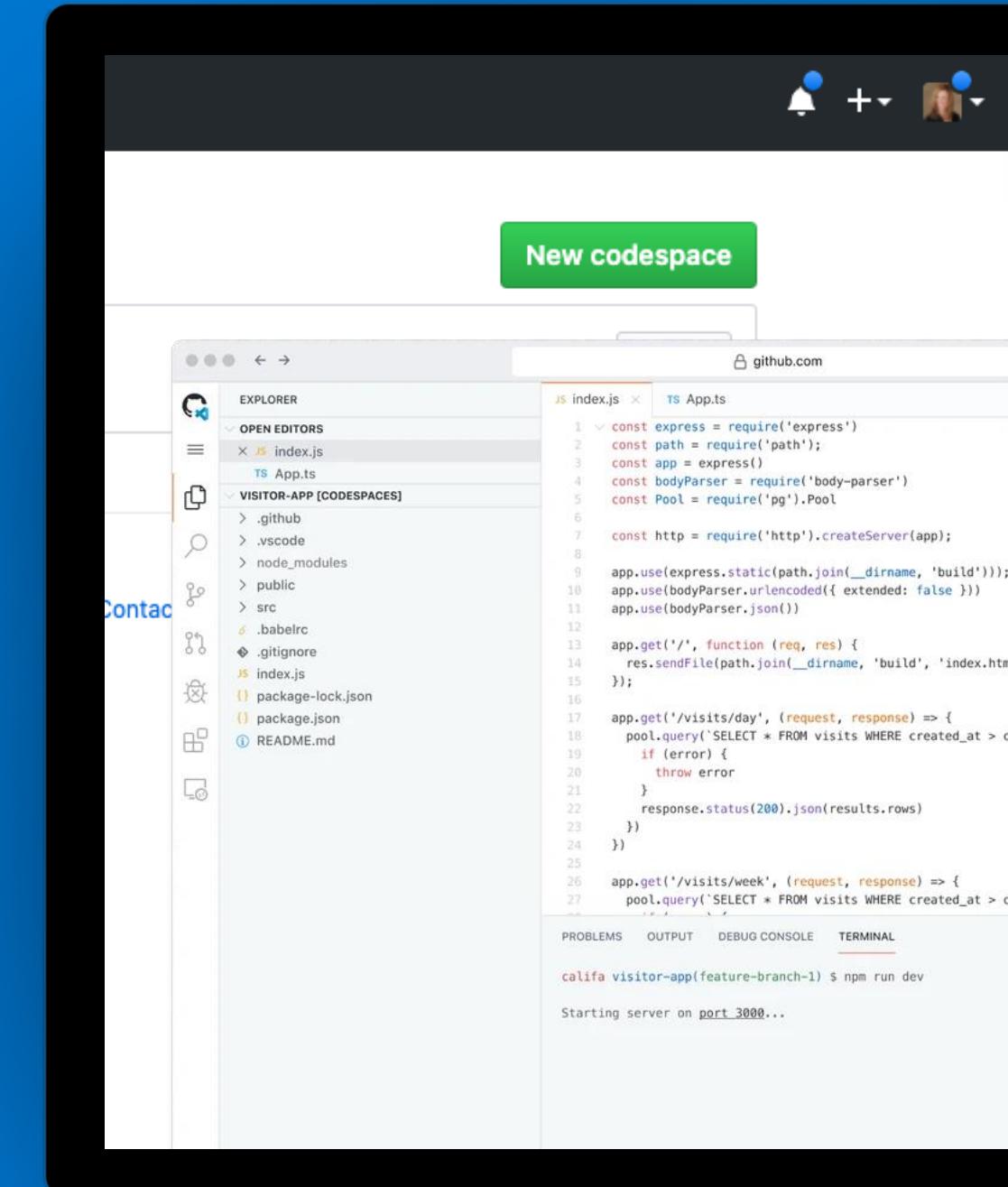
Configure your editor with dotfiles and VS Code extensions to create a consistent environment in every Codespace.

Developer choice

Powered by web-based version of VS Code or, use with your desktop IDE.

<https://github.com/features/codespaces> &

<https://github.blog/2021-08-11-githubs-engineering-team-moved-codespaces>



GitHub Copilot

Your AI pair programmer

What is GitHub Copilot?

An AI pair programmer that helps you write code faster and with less work. GitHub Copilot draws context from comments and code, and suggests individual lines and whole functions instantly

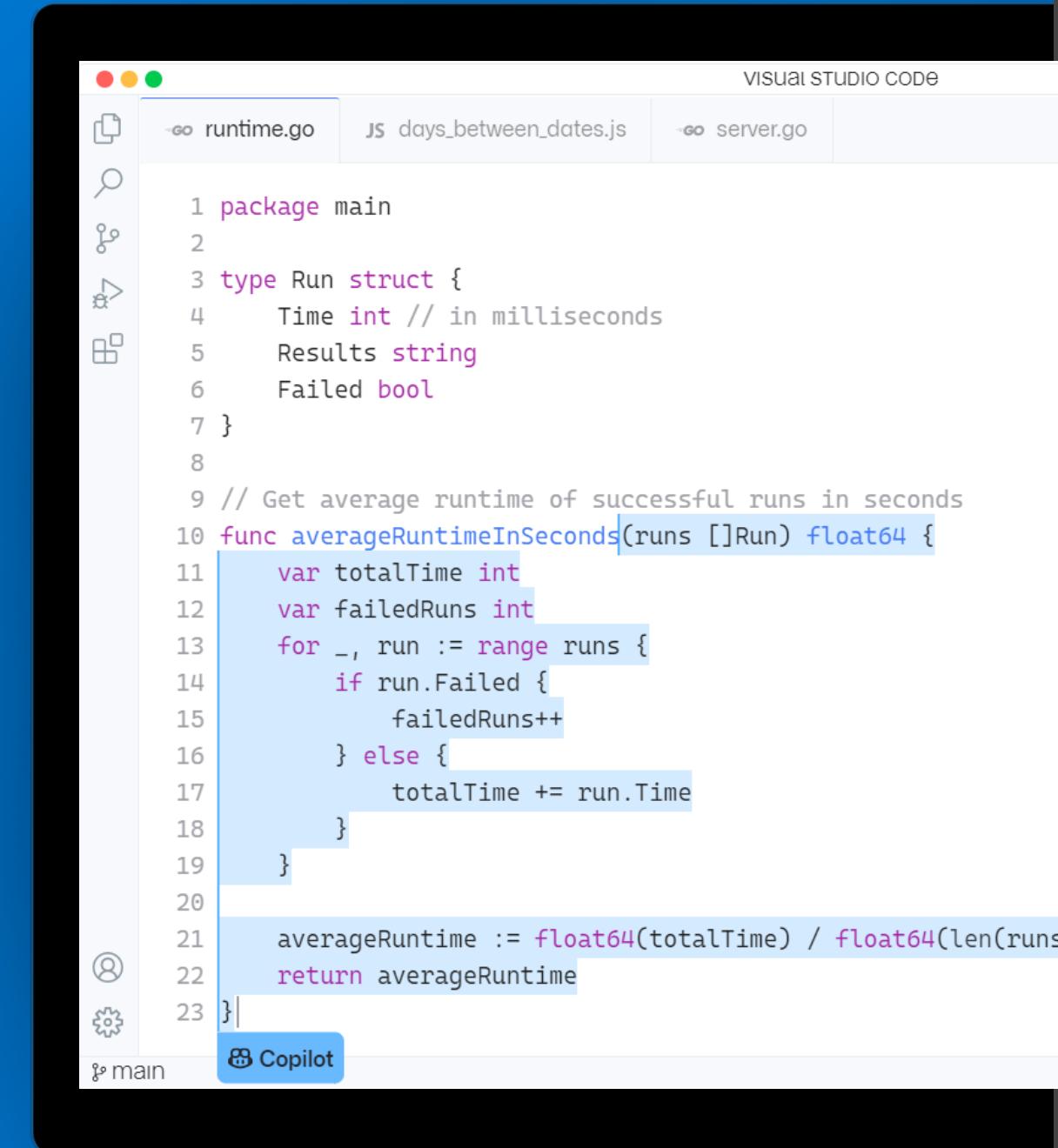
Convert comments to code.

Write a comment describing the logic you want and let GitHub Copilot assemble the code for you.

Tests without the toil.

Tests are the backbone of any robust software engineering project. Import a unit test package and let GitHub Copilot suggest tests that match your implementation code.

[Introducing GitHub Copilot: your AI pair programmer](#)



The screenshot shows a dark-themed instance of Visual Studio Code. In the top right corner, it says "VISUAL STUDIO CODE". The left sidebar has icons for file, search, file history, and a folder labeled "main". The main editor area has three tabs: "runtime.go" (selected), "days_between_dates.js", and "server.go". The "runtime.go" tab contains Go code:

```
1 package main
2
3 type Run struct {
4     Time int // in milliseconds
5     Results string
6     Failed bool
7 }
8
9 // Get average runtime of successful runs in seconds
10 func averageRuntimeInSeconds(runs []Run) float64 {
11     var totalTime int
12     var failedRuns int
13     for _, run := range runs {
14         if run.Failed {
15             failedRuns++
16         } else {
17             totalTime += run.Time
18         }
19     }
20
21     averageRuntime := float64(totalTime) / float64(len(runs))
22     return averageRuntime
23 }
```

A blue callout box labeled "Copilot" points to the line "averageRuntime := float64(totalTime) / float64(len(runs))".

GitHub Packages

Enterprise-grade repository to securely manage dependencies.

Single, secure login for code and packages

Public and private registries, with support for multiple formats and ecosystems.

Chain of custody

Find packages on GitHub, understand details and dependencies.

Innersource and collaboration

Centralized, searchable source for approved and secure packages within your organization.

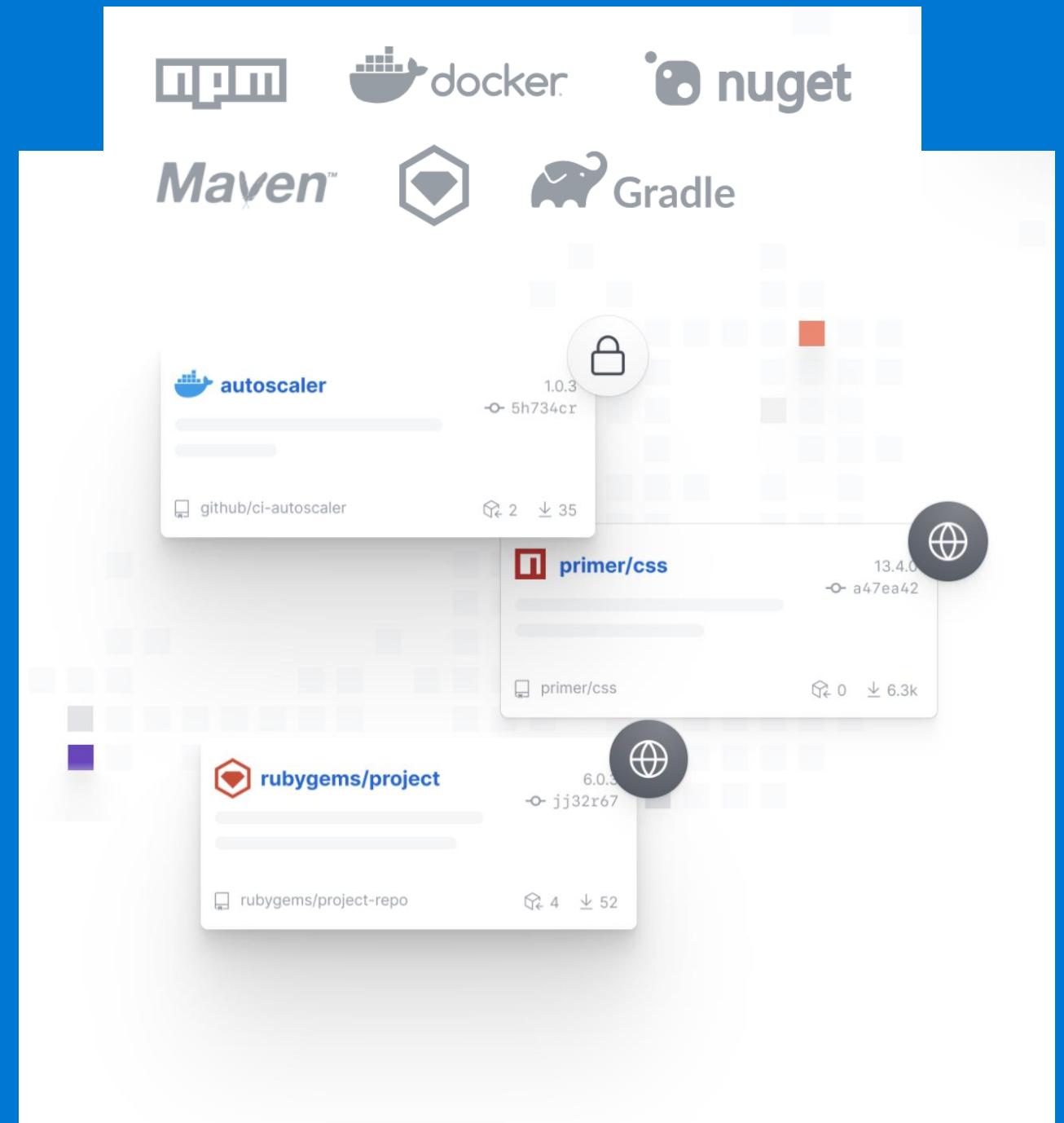
CI/CD and Dependency Management

Works seamlessly with GitHub Actions.

Also works with your existing CI tool.

Use as a **Container Registry** with Helm Charts.

<https://docs.github.com/en/packages>



Code Scanning

Preventing vulnerabilities from the start

Find and fix vulnerabilities fast

Find and fix vulnerabilities before they are merged into the code base with automated CodeQL scans.

Community of top security experts

Community-driven query set powers every project with a world-class security team. Or, create your own custom queries.

Integrated with developer workflow

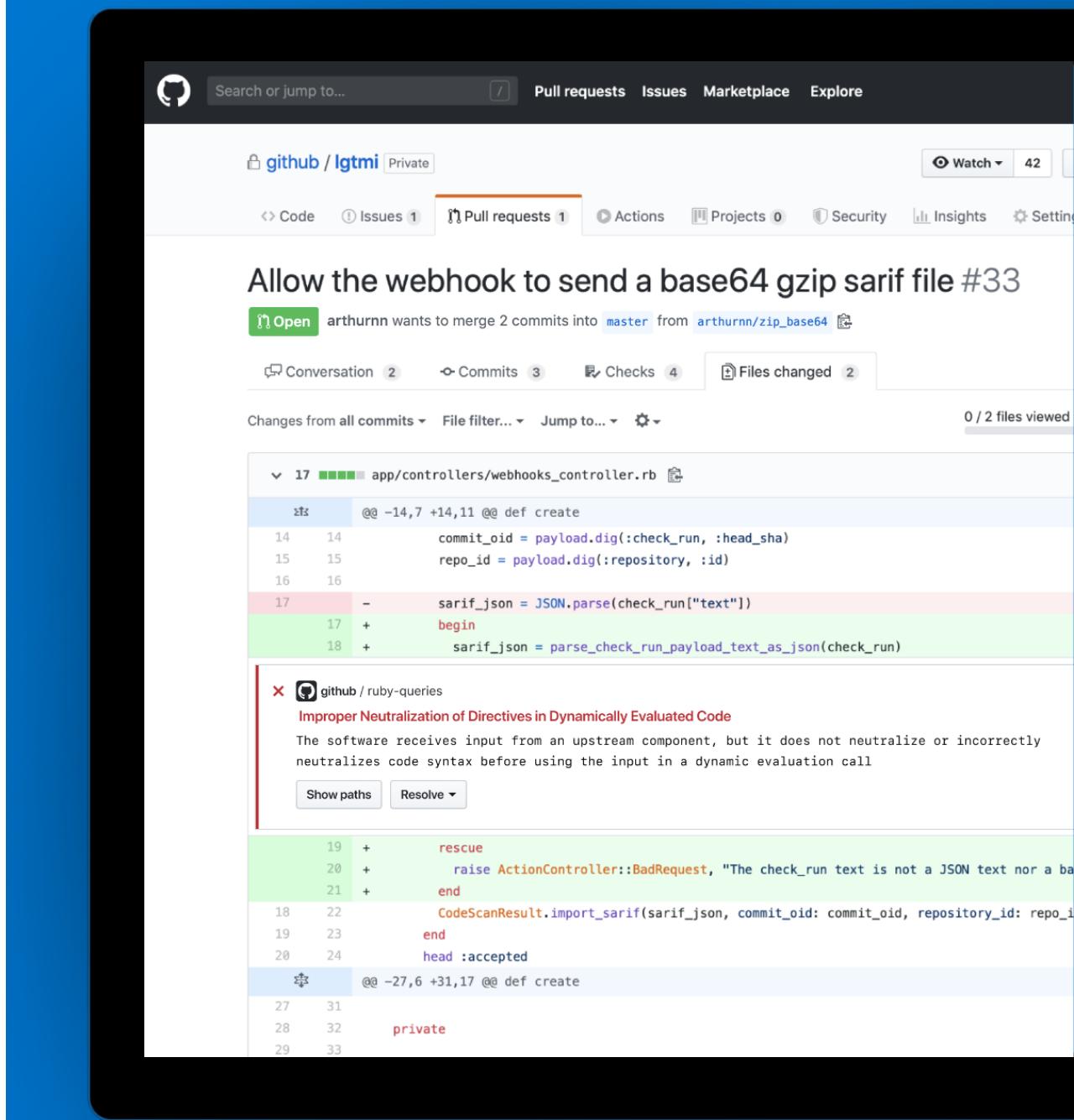
Integrate security results directly into the developer workflow for a frictionless experience and faster development.

Extensible

Plug other SAST tools into the same developer workflow.

[&](https://github.com/features/security)

<https://github.blog/2020-09-30-code-scanning-is-now-available/>



Secret Scanning

Keeping your secrets, a secret

Identifies secrets as early as possible

Finds secrets the moment they are pushed to GitHub and immediately notifies developers when they are found. Scan your entire git history.

Community of secret scanning partners

For every commit made to your repository, and its full git history, we'll look for secret formats from secret scanning partners

Supports both public and private repos

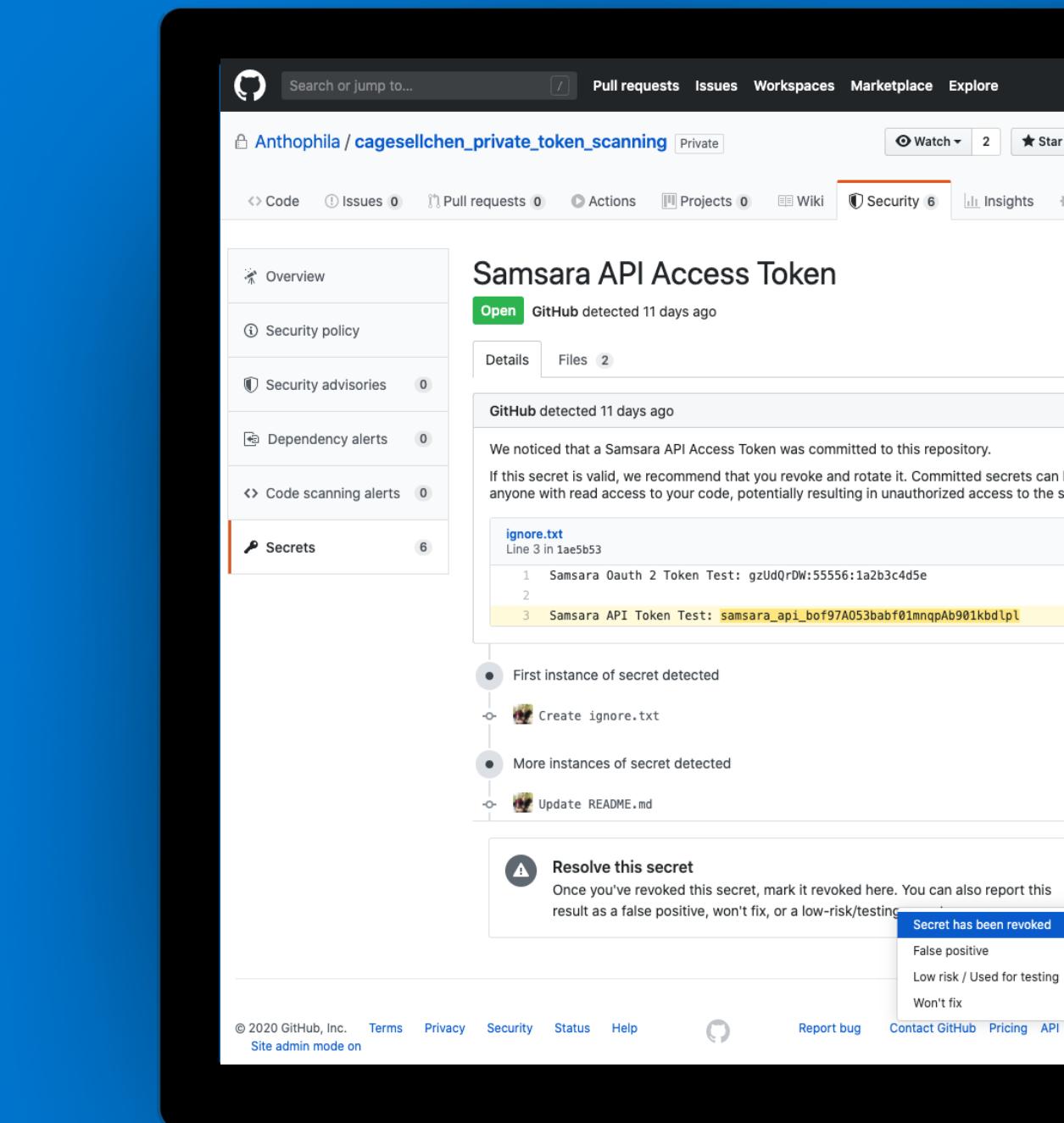
Secret scanning watches both public and private repos for potential secret vulnerabilities.

Protection from exposed secrets

Automatically disable or suspend secrets from [45+ service providers](#) as soon as they are committed.

The provider list [continues to grow](#).

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



Dependency Scanning & Auto Security Updates

Securing the software supply chain

Dependency graph

See the packages your project depends on, the repositories that depend on them, and any vulnerabilities detected in their dependencies.

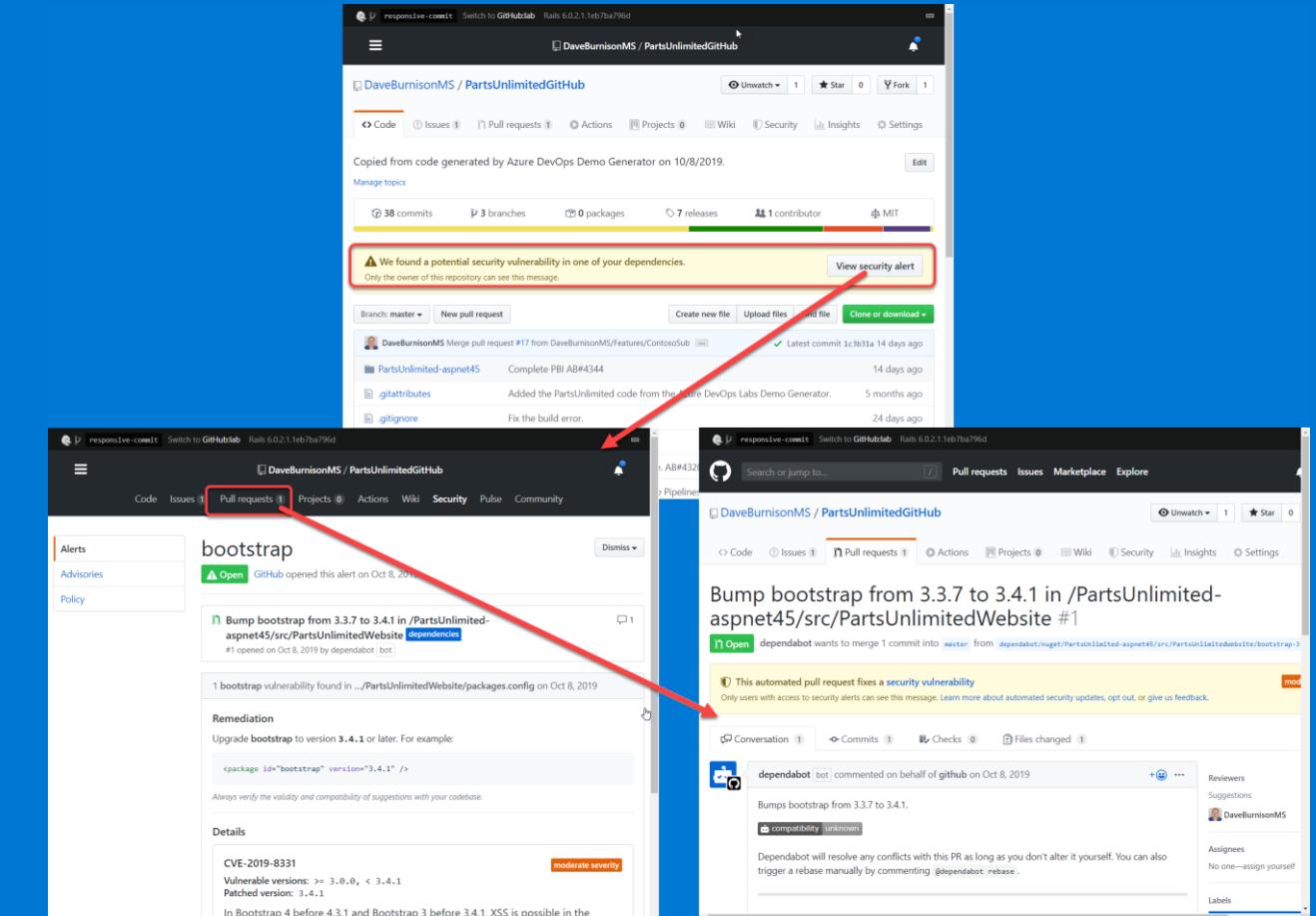
Dependabot alerts

Get notified when there are new vulnerabilities affecting your repositories. GitHub detects and alerts users to vulnerable dependencies in public and private repos.

Dependabot security and version updates

Keep your supply chain secure and up-to-date by automatically opening pull requests that update vulnerable or out-of-date dependencies.

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



Automated Delivery

Automating workflows from code to cloud

Accelerate delivery through automation

Automation triggers for 20+ project events allows for automation beyond just CI/CD to any available API

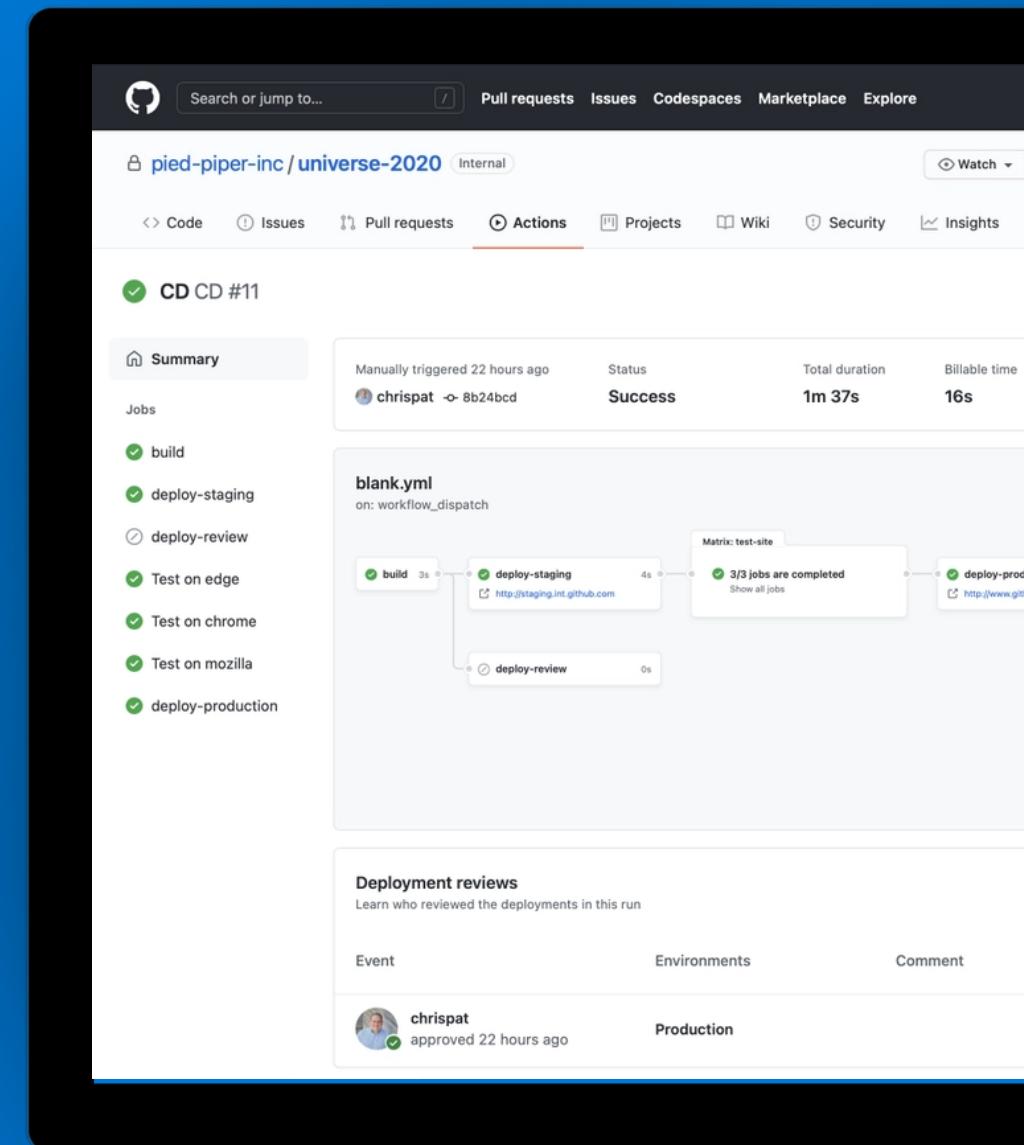
Simple and easy to use

Configuration based on YAML with a host of sample workflows to learn from and get started

Global community for actions

Thousands of open-source Actions, maintained by the community and by companies offering integrations, including Microsoft Azure

<https://github.com/features/actions> &
[New from Universe 2020: Environments, Manual Approvals and Workflow Visualization](#)



GitHub Actions for Azure

Over 120 ready-to-use actions for Azure

Automate workflows:

Easily automate your code-to-cloud workflows on any GitHub event, such as code commits, creation of pull requests, or new GitHub releases. Deploy apps across popular languages and frameworks or, running on any OS.

Shift security and compliance left:

Automate governance, security, and compliance into the early stages of the software development life cycle. Write infrastructure configurations, release pipelines, compliance, and security policies "as code" to enable continuous improvement and greater transparency.

Operate seamlessly:

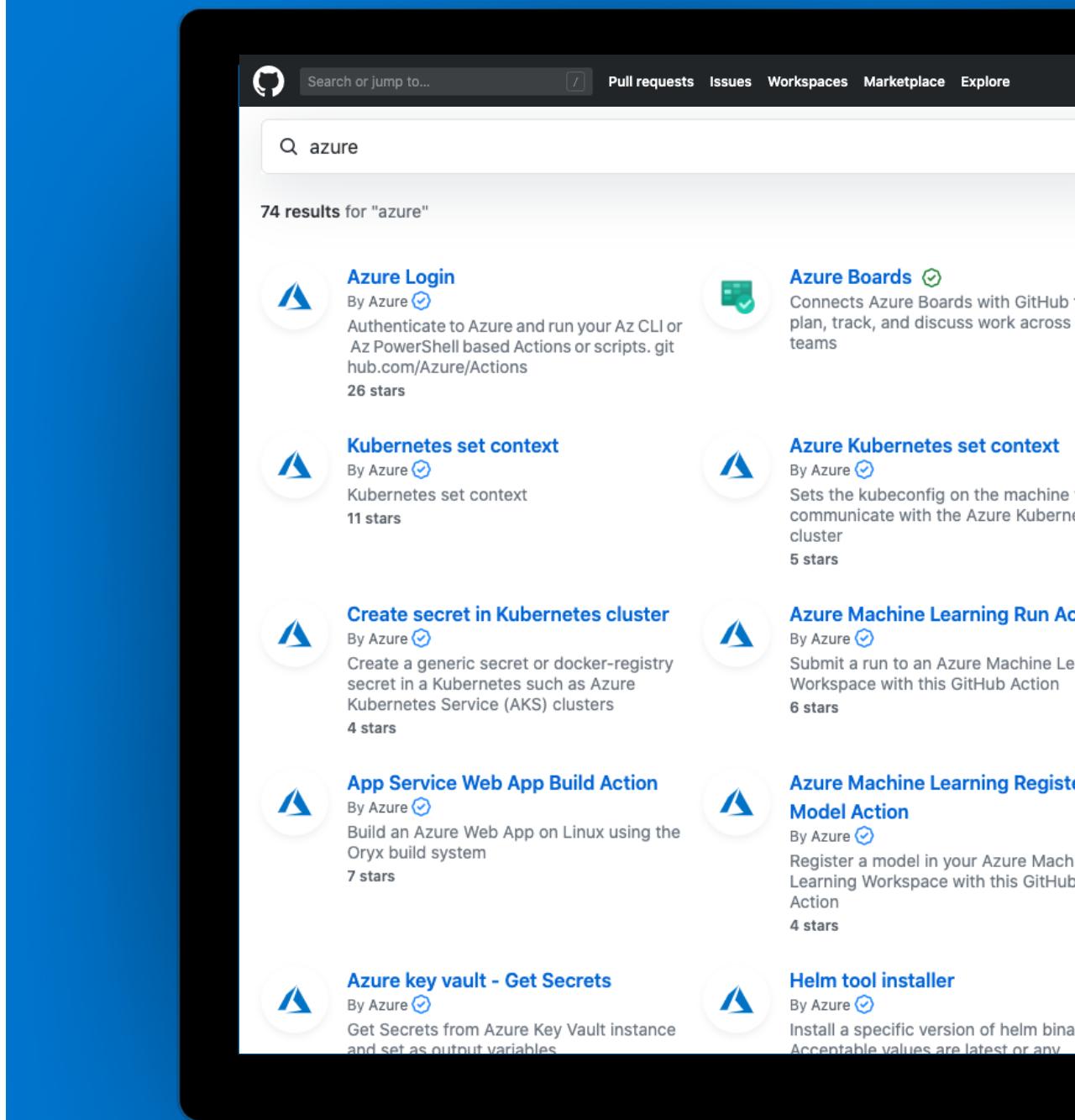
Shift from manual management and operation of IT environments to enable seamlessly automated IT management (GitOps) that is controllable, collaborative, and repeatable at enterprise scale.

Accelerate developer velocity:

Deploy using popular developer tools integrated with GitHub Actions. Significantly reduce ramp-up time, avoid frequent context switching, and help teams be more productive with built-in extensions into Visual Studio Code, Azure CLI, and Azure Portal.

<https://azure.github.io/actions>

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



GitHub Mobile

Collaborate and develop, on-the-go

Stay connected

Stay in touch with your team and the GitHub Community, right from your mobile device

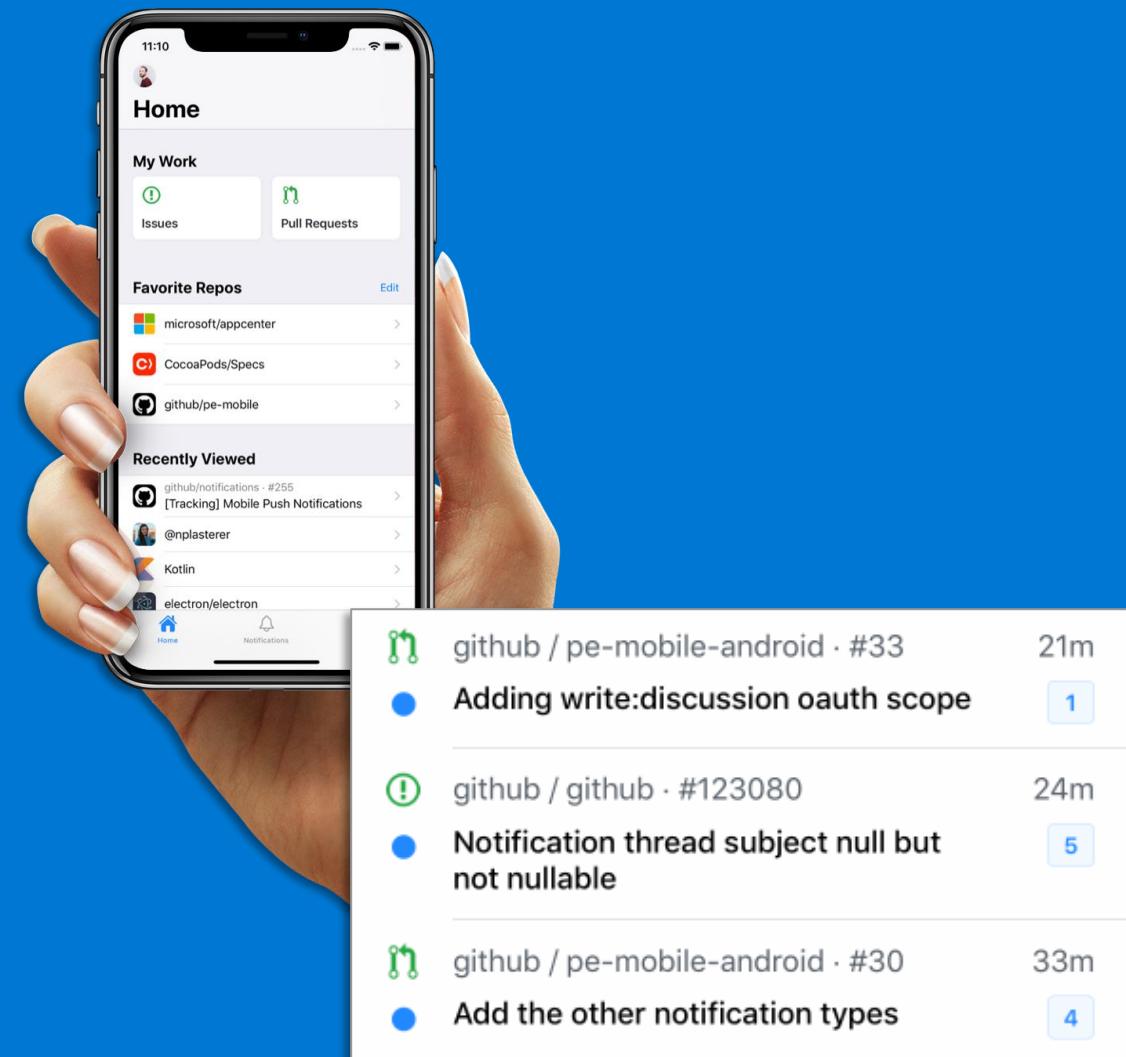
Collaborate from anywhere

Collaborate on Issues and Pull Requests with a totally redesigned experience. Comment, react, and merge code.

Breeze through your workflow

Get your inbox to zero in no time—swipe to finish a task or save the notification to return to it later.

<https://github.com/mobile/>



Discussions

Join the conversation

Open conversations about code and community

New threaded format makes it easy to brainstorm feature ideas, help new users get their bearings, and collaborate on software.

Perfect for Q&A

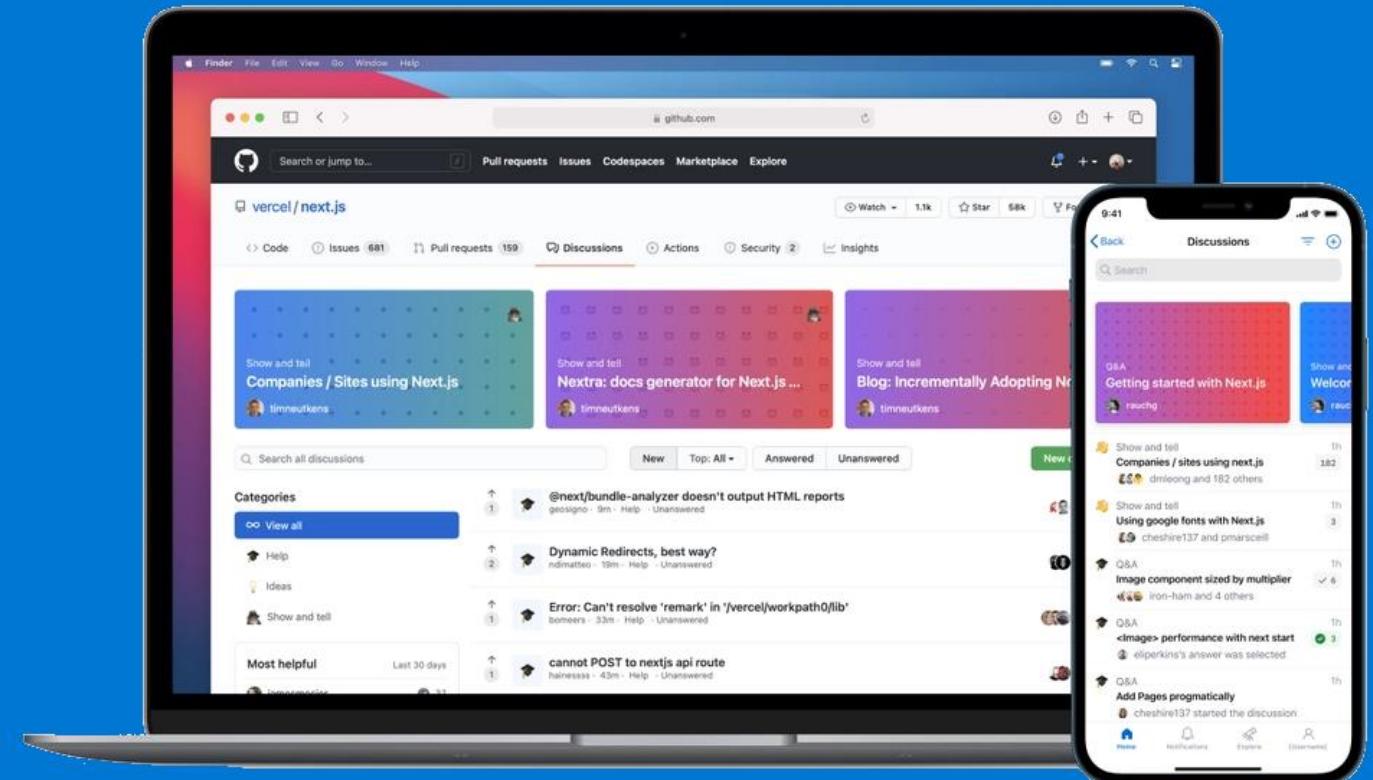
Questions can be marked as answered, so over time a community's knowledge base grows naturally

Familiar GitHub experience

Discussions live in your project repository, so they're accessible where your community is already working together.

<https://docs.github.com/en/discussions>,

Example: <https://github.com/github/feedback/discussions>



GitHub public roadmap

Search or jump to... / Pull requests Issues Marketplace Explore

GitHub public roadmap

Board Table Generally av... Cloud (githu... Server (GHE) GitHub Adv... GitHub Issues Give feedback

Filter by keyword or by field

Q4 2021 – Oct-Dec 31

- roadmap #248
Releases Refresh and Automatically Generated Release Notes (Beta)
all beta cloud code-to-cloud shipped
- roadmap #247
Releases Refresh and Automatically Generated Release Notes (GA)
all cloud code-to-cloud ga shipped
- roadmap #238
Security Manager org-level role - cloud beta
all beta cloud security & compliance shipped
- roadmap #219
Optional expiration for Personal Access Tokens (Server)
ecosystem ga github enterprise server shipped
- roadmap #215
Dependency graph: Dependencies API (Private Beta)
all beta cloud security & compliance shipped
- roadmap #212

Q1 2022 – Jan-Mar 49

- roadmap #257
Actions: Reusable workflows on GHES (Beta)
actions beta code-to-cloud github enterprise server
- roadmap #240
Actions: macOS 12 (Monterey) on GitHub-hosted runners (Private Beta)
actions all beta cloud code-to-cloud
- roadmap #220
Actions: Limit self-hosted Runners to specific workflows
actions cloud code-to-cloud ga github enterprise server
- roadmap #216
Audit log API on GHAE and GHES
ae beta github ae github enterprise security & compliance server
- roadmap #213
GHEC Audit Log Streaming General Availability
admin-cloud cloud ga github enterprise security & compliance shipped
- roadmap #193

Q2 2022 – Apr-Jun 18

- roadmap #256
Actions: Reusable workflows on GHES (GA)
actions code-to-cloud ga github enterprise server
- roadmap #255
Actions: Limit self-hosted Runners to specific workflows on GHES
actions code-to-cloud ga github enterprise server
- roadmap #254
Actions: Use actions from internal repositories on GHES (Beta)
actions beta code-to-cloud github enterprise server
- roadmap #237
Git LFS moves to metered billing
cloud ga github enterprise github team other
- roadmap #195
API Versioning (GA)
all ecosystem ga
- roadmap #58

Q3 2022 – Jul-Sep 1

- roadmap #207
Kotlin security analysis support in CodeQL code scanning (beta)
ae beta codeql github advanced security security & compliance server

Future 46

- roadmap #246
Billing Usage API
admin-cloud beta billing cloud github enterprise
- roadmap #245
ISO 27001 Certification for Dotcom and GitHub Enterprise Cloud
admin-cloud all cloud ga security & compliance
- roadmap #218
Create a branch from an issue
all in design planning
- roadmap #217
Updated billing overview page
admin-cloud all billing ga
- roadmap #211
Pull request revisions and improved workflows
all beta code
- roadmap #206
Swift security analysis support in CodeQL code scanning (beta)
ae beta codeql

Roadmap UI walkthrough

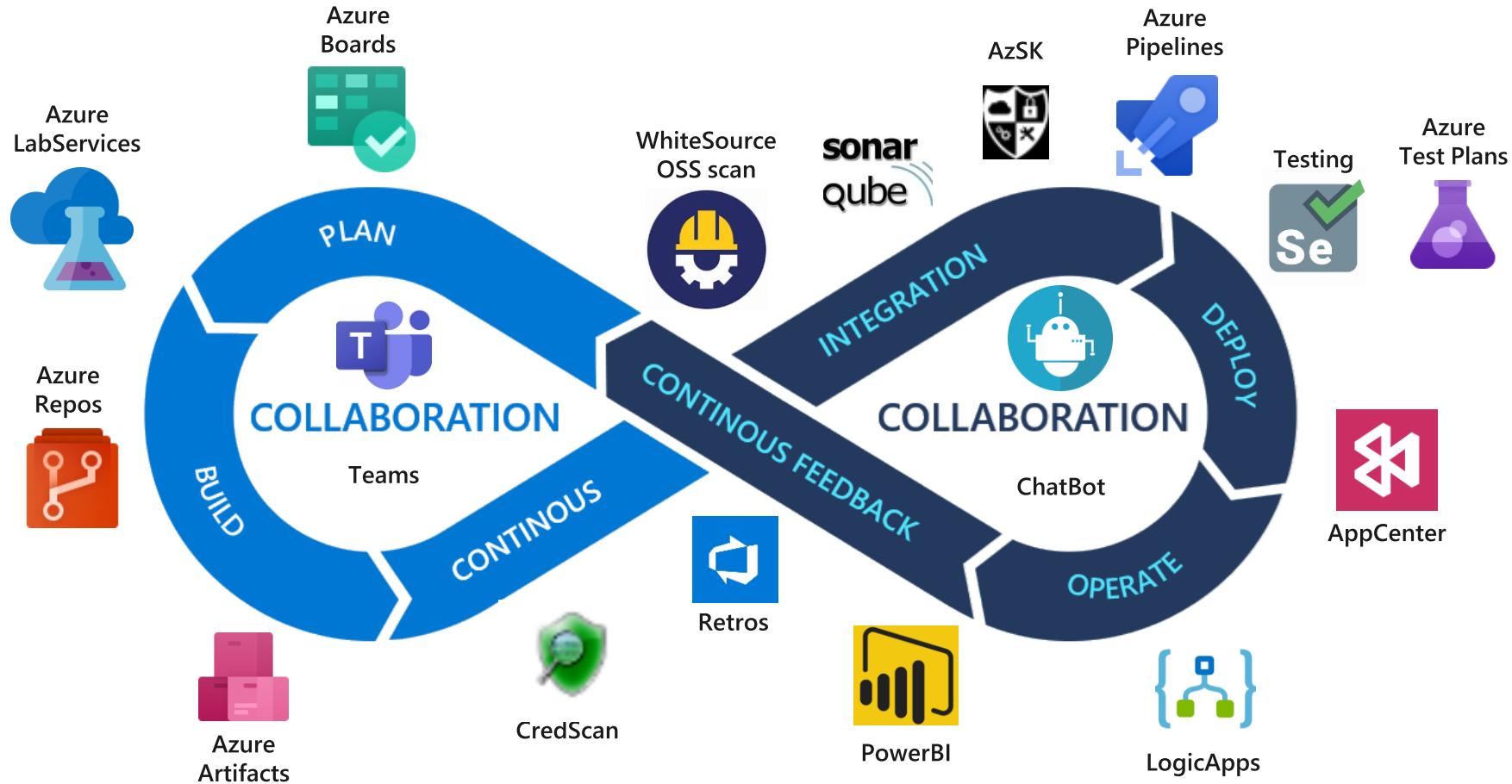


Presented by [Kan]

Azure DevOps & GitHub Comparison

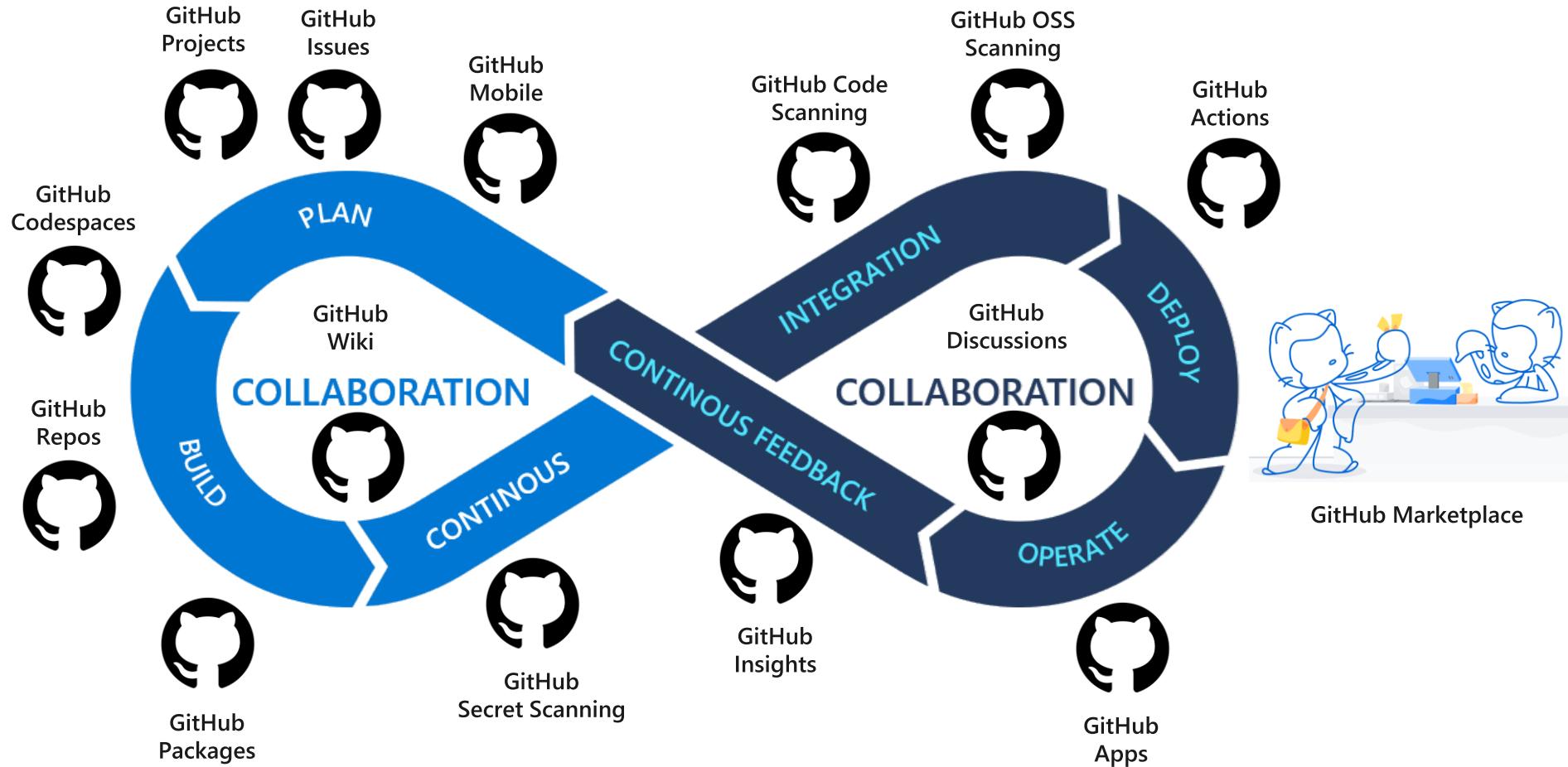
GitHub & Dojo Presentation 2022 (15 minutes)

Reference Implementation with Azure DevOps

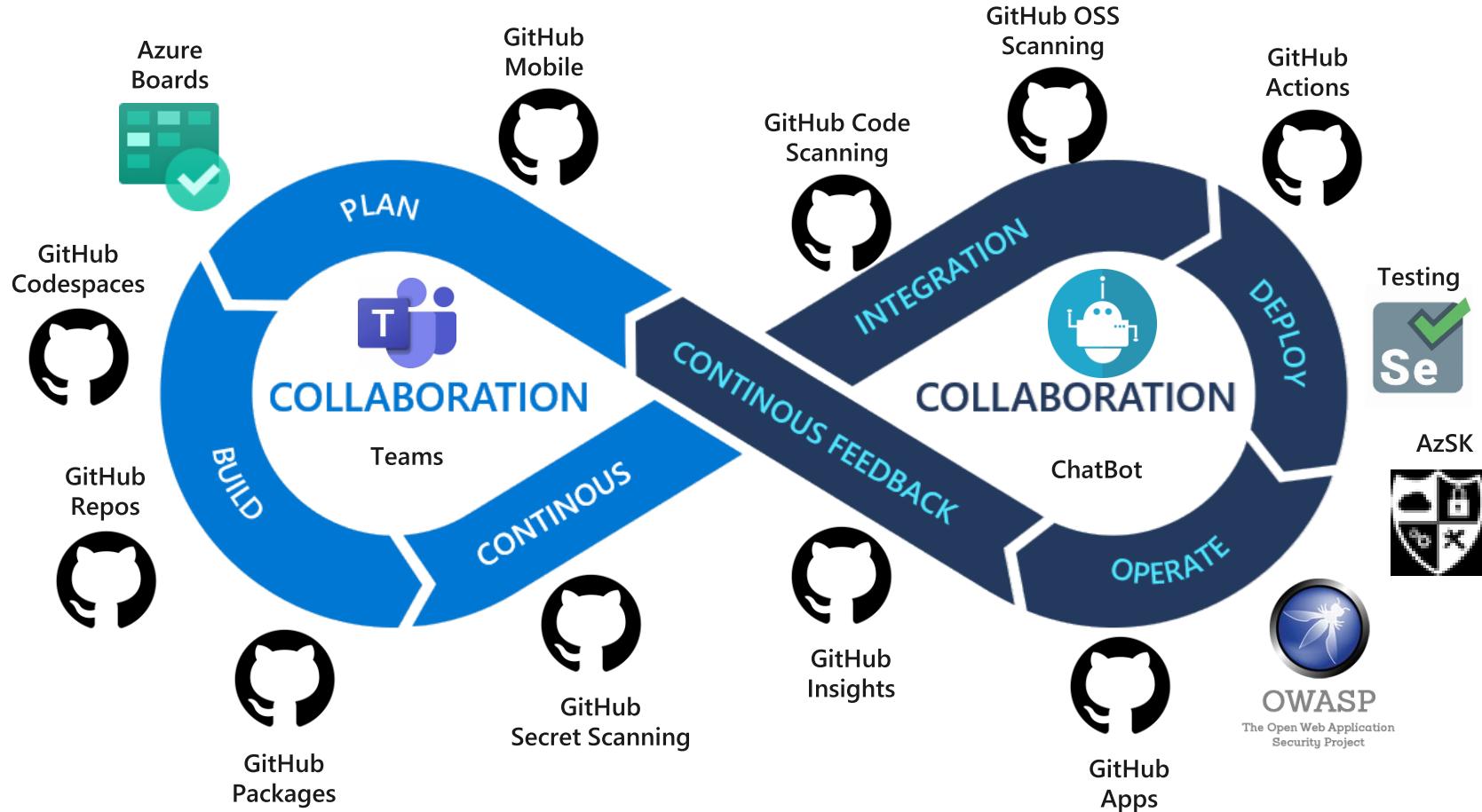


 Microsoft Azure

Reference Implementation with GitHub



Reference Implementation with GitHub + Azure DevOps

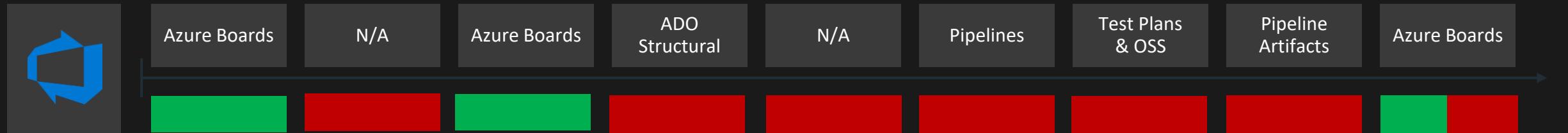
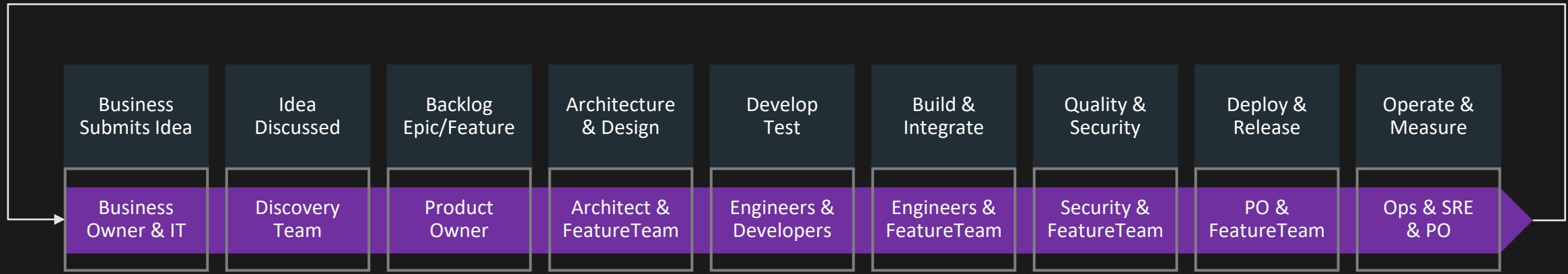


 Microsoft Azure

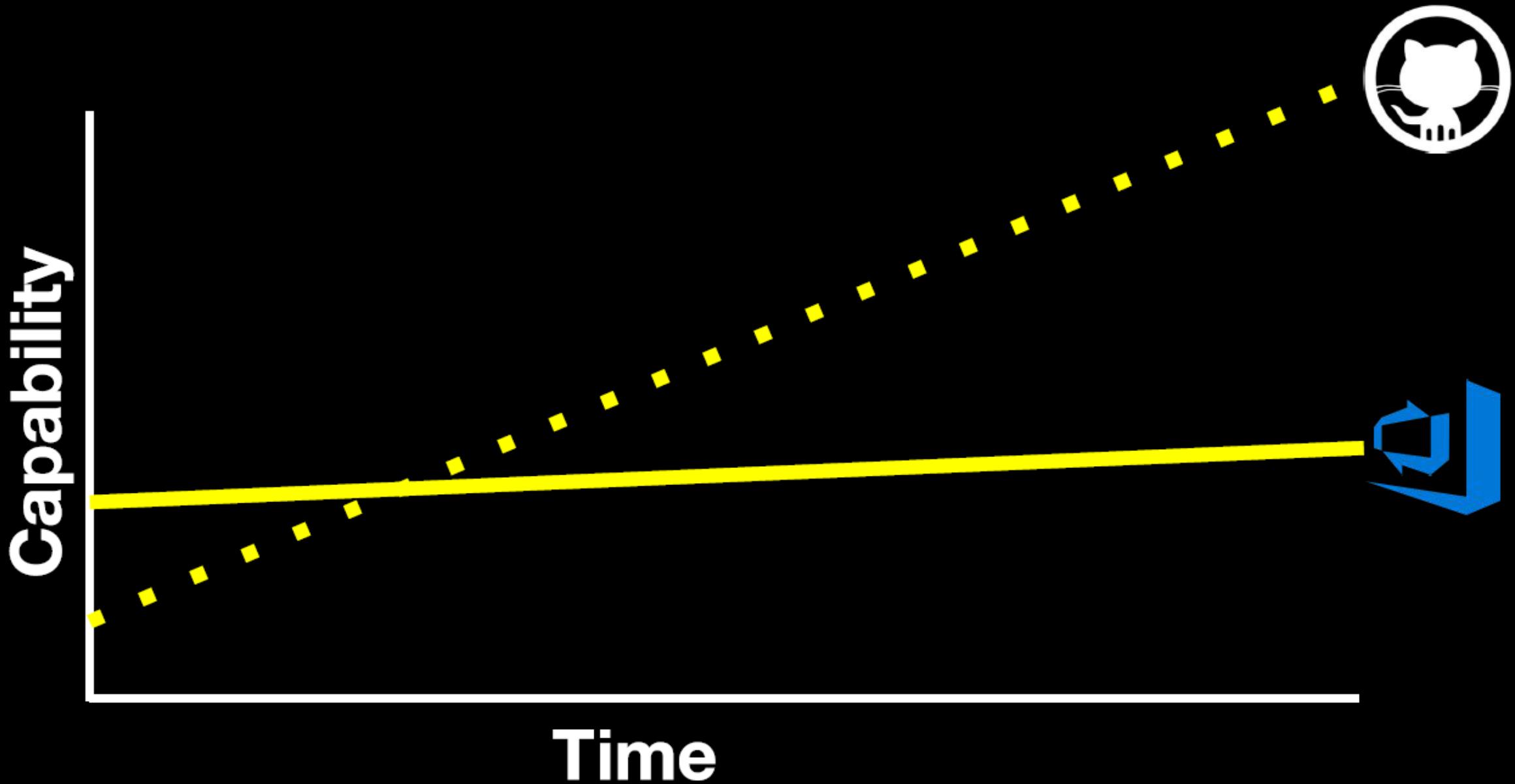
GitHub vs. Azure DevOps as of Now

1st choice

2nd choice



The Future



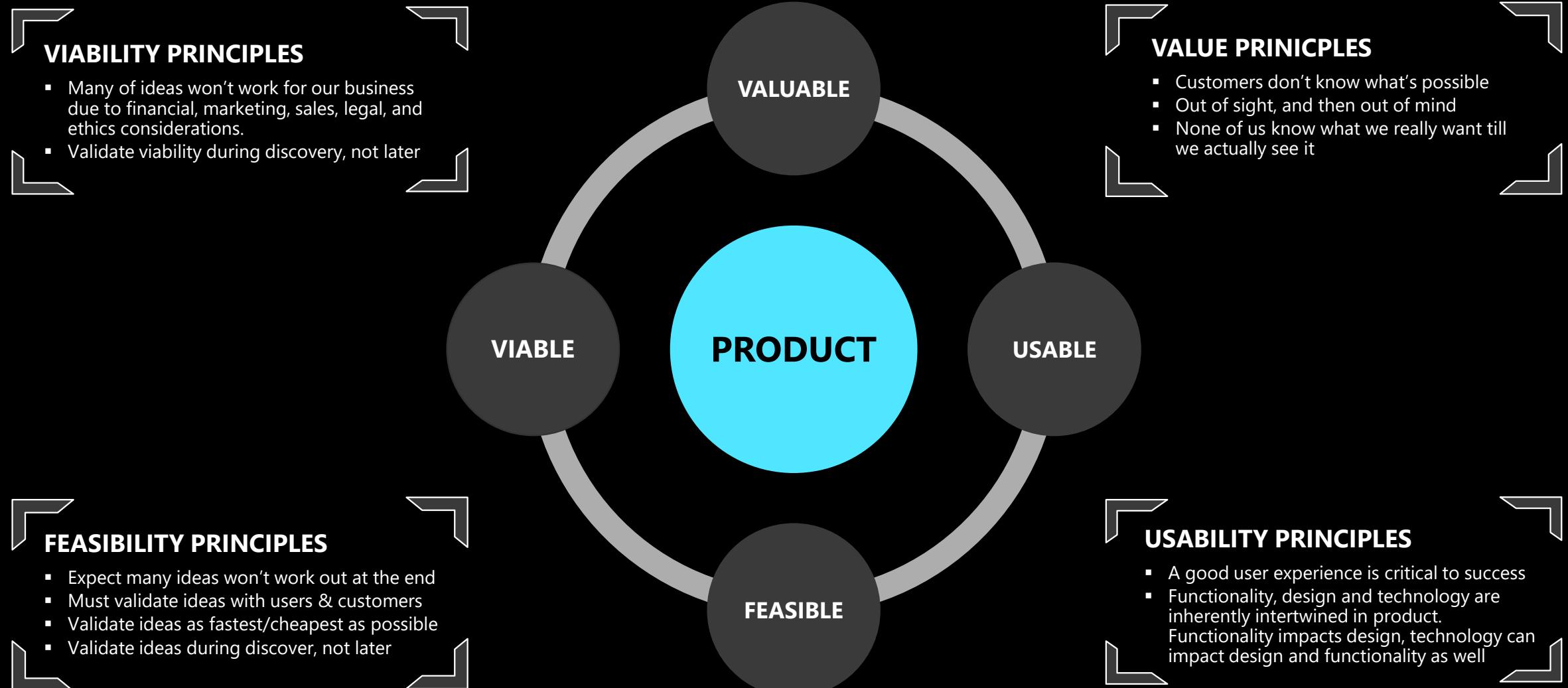
What problems do we try to solve in DevOps?



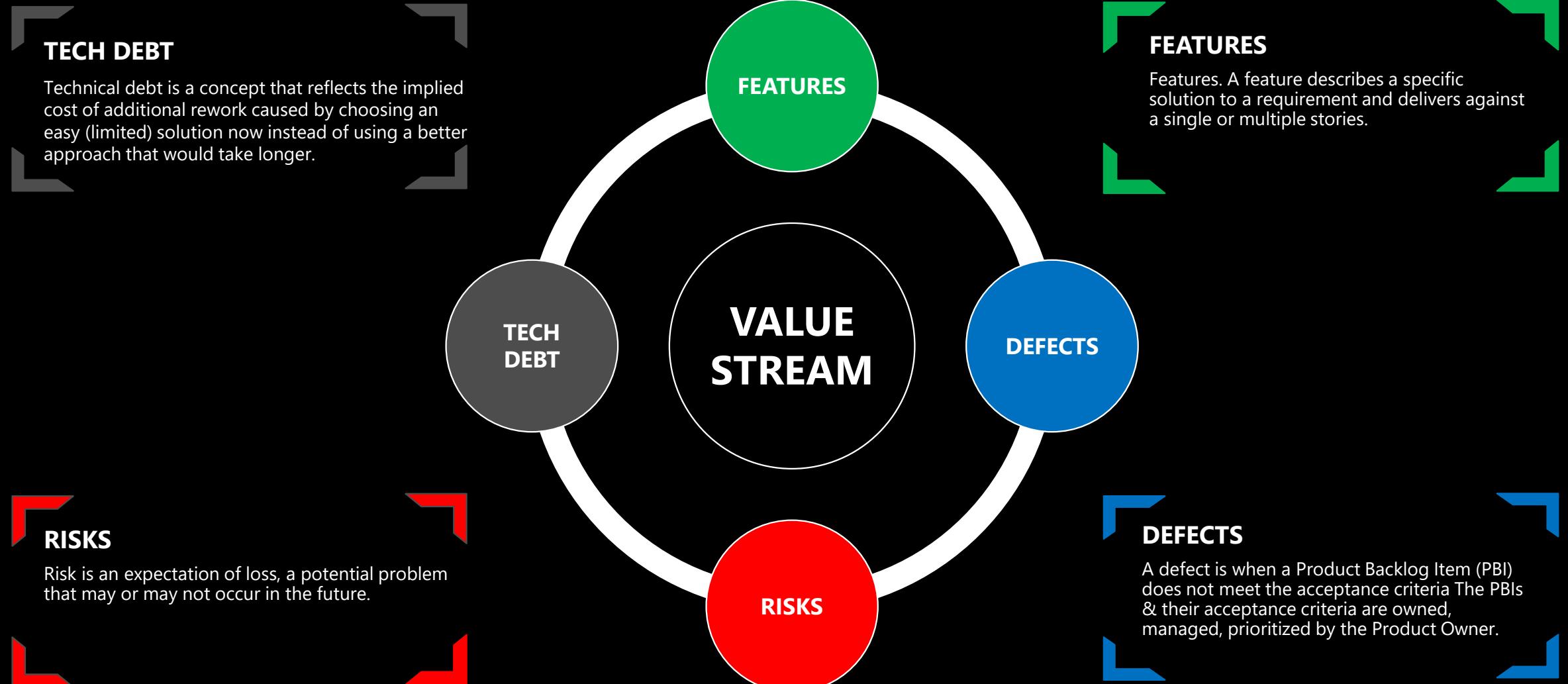
"If I had only one hour to save the world, I would spend fifty-five minutes defining the problem, and only five minutes finding the solution."

- Albert Einstein

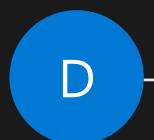
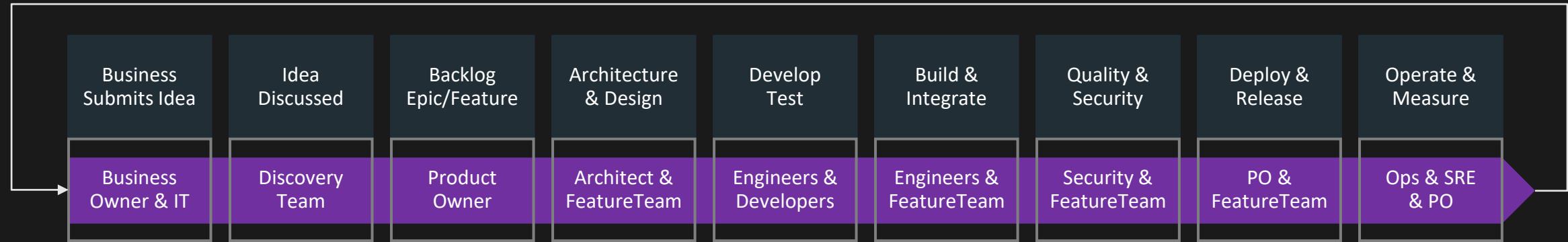
DevOps and Product



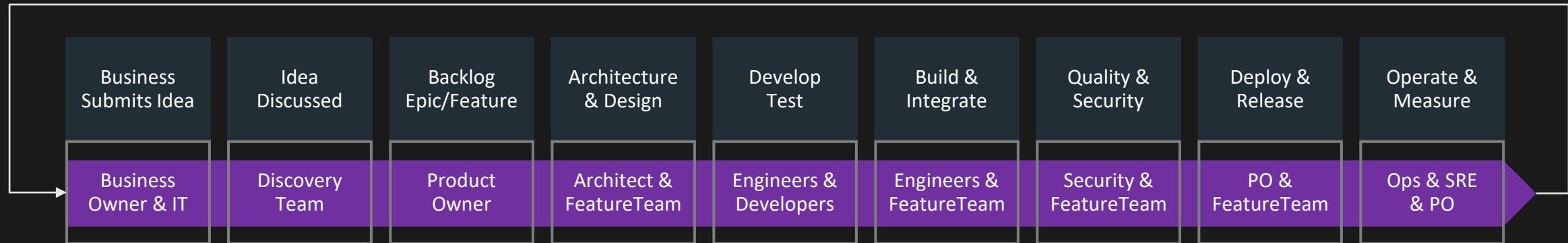
Flow Distribution



DevOps Value Stream Flow



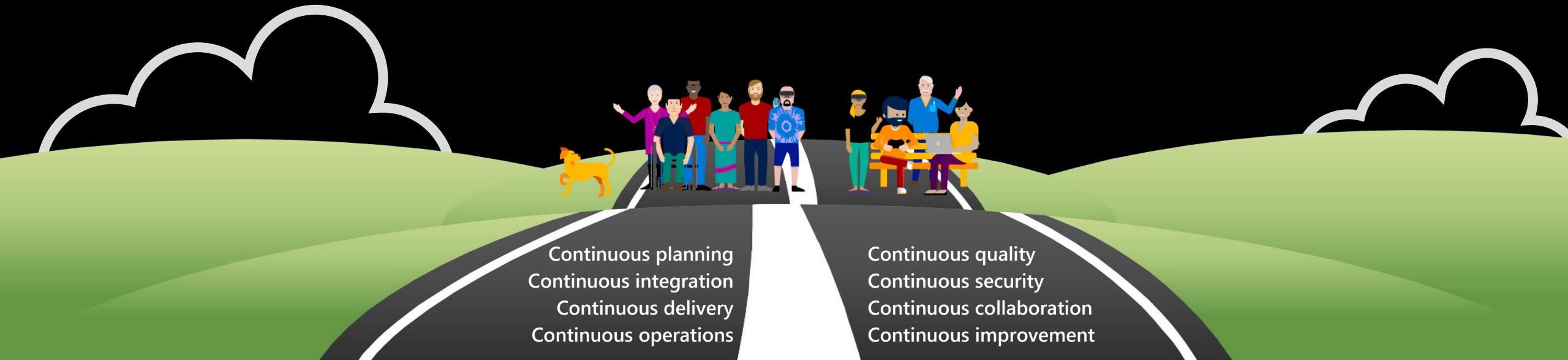
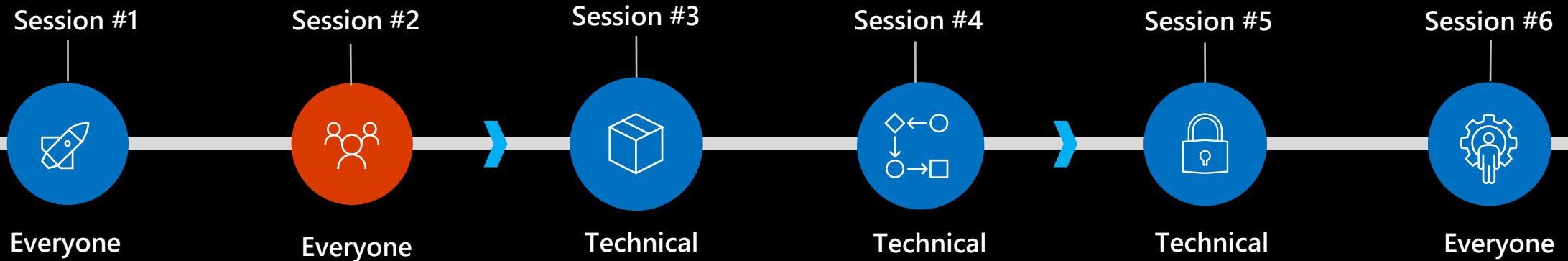
GitHub-enabled Value Stream Flow



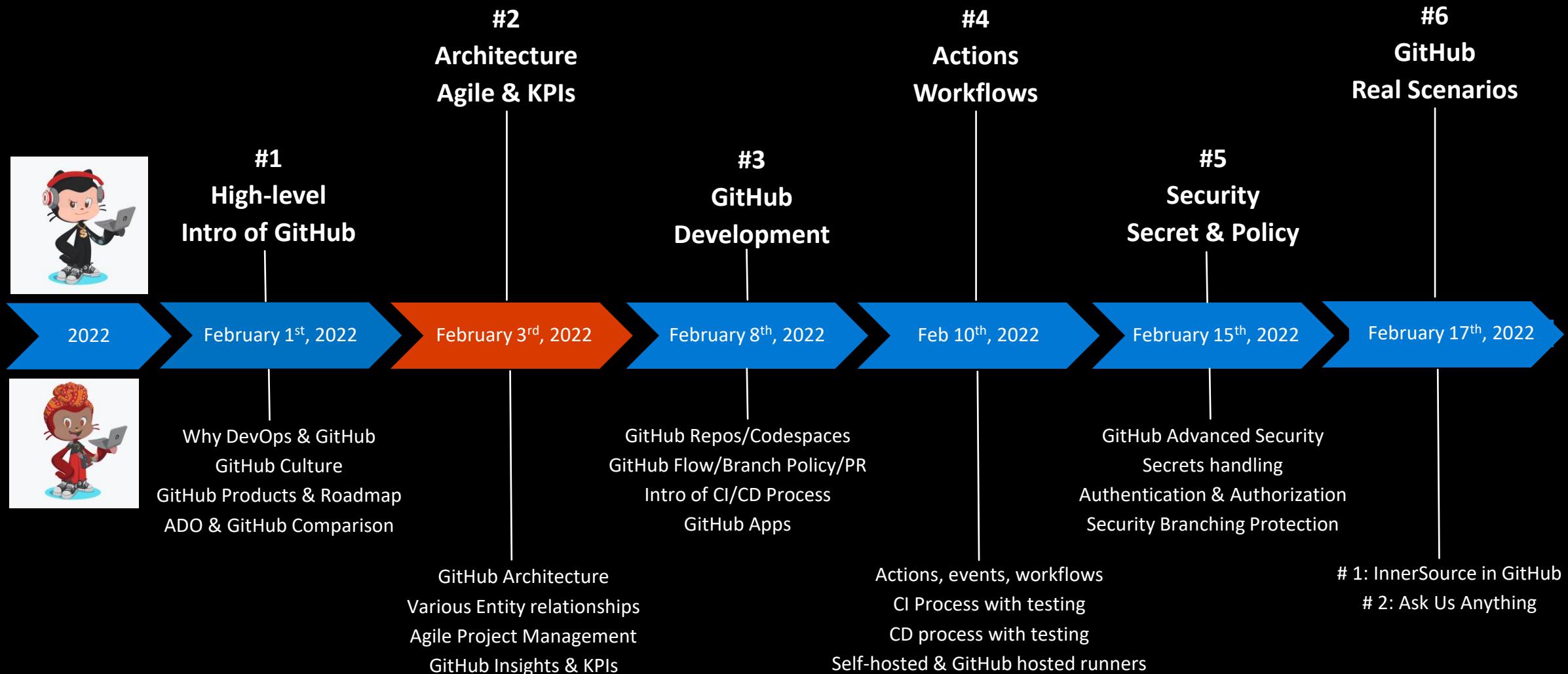


Questions

Session #2: Architecture, Agile Project Management, KPIs



GitHub Bootcamp – February 3rd, 2022



The Topics of the Day – Session #2

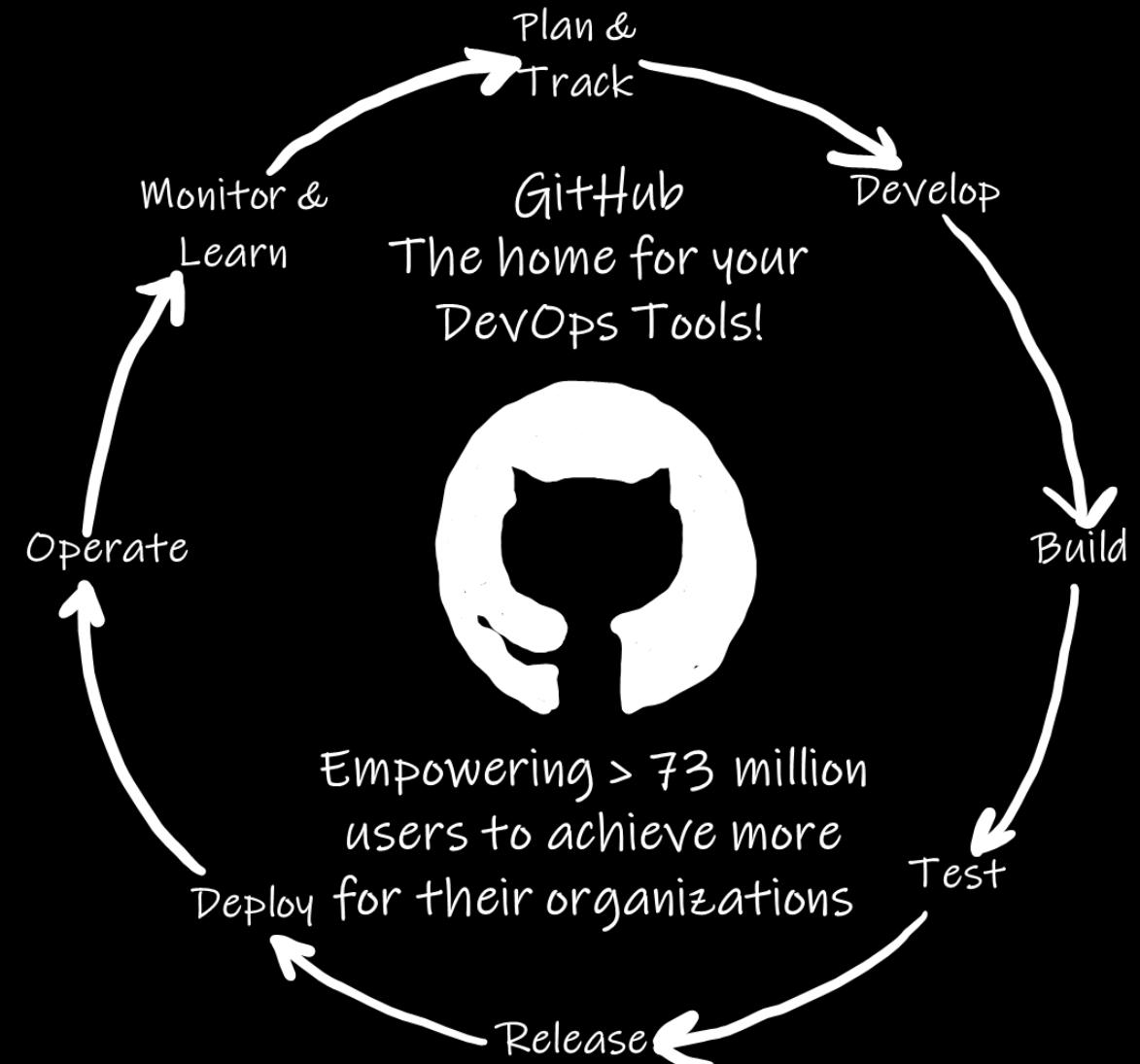
1 GitHub Architecture

2 GitHub Agile Management

3 GitHub Insights

4 DevOps KPIs with Grafana

5 Questions



CALL TO ACTION

Connect with Microsoft to discuss your needs:

MSUSDev@microsoft.com

Learn more about DevOps Dojo:

<https://aka.ms/dojo-intro>

Start a free trial today:

https://github.com/organizations/enterprise_plan



Thank You

