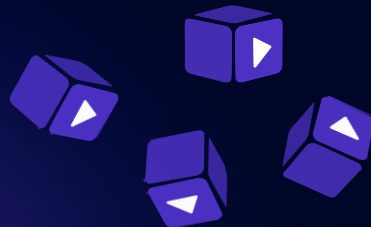


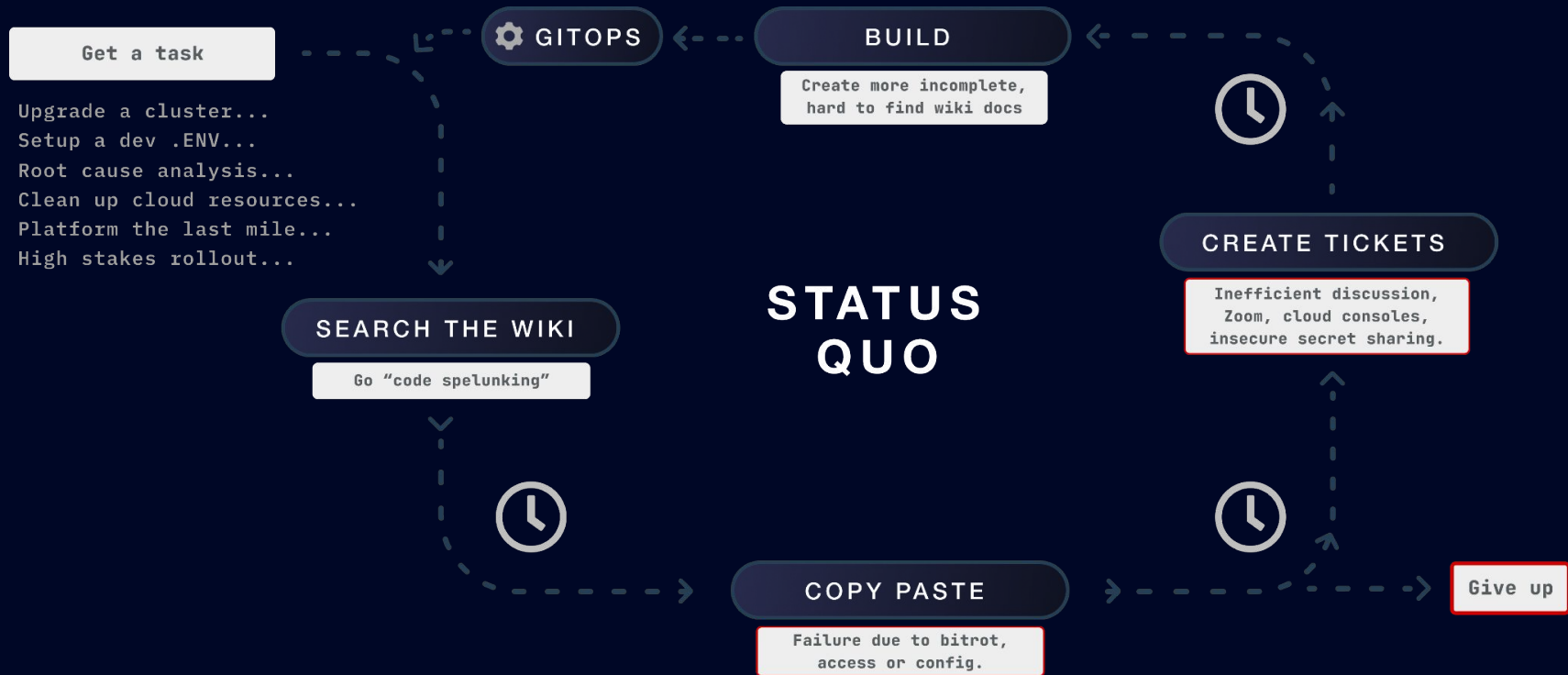
# DevOps Notebooks Built with Markdown

Codify your operational workflows as cloud-native,  
testable, interactive documentation.

[runme.dev](https://runme.dev)



# Outdated documentation is a problem



# What do docs *really* need?

Docs deps are complex in our cloud-native world.

## Scripts

- Executable commands/code
- Terminal output/artifacts
- Prompts and behavior

## .ENV

- Configuration
- Secrets

## Documentation

- Instructions
- Commands & Code
- References to:
  - bash scripts
  - docs, IaC
  - cloud dashboards

(docs today)

## Integrations

- Cloud consoles
- Cloud-native CLI's
- Operational SaaS API's
- Runme AI for authoring

## System

- Dependencies
- Networking
- Local permissions
- Shell configuration





# Introduction

25k+

VS Code Installs

1k+

Github Stars

700

Discord users

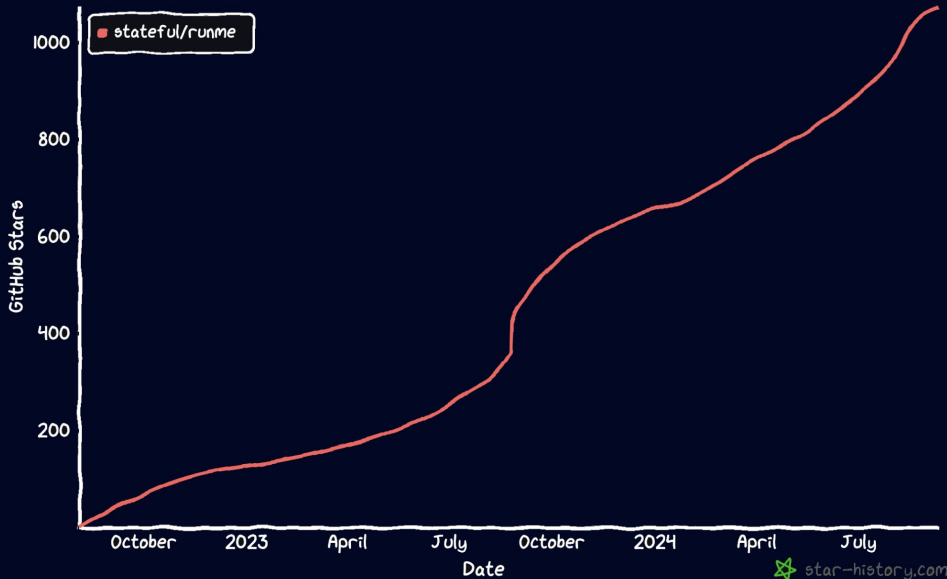
Open-source core  
tech, transforms  
between Markdown  
and Notebooks.

Notebooks are cloud provider  
aware, cloud-native  
interactive interfaces.

**runme.dev**



Star History

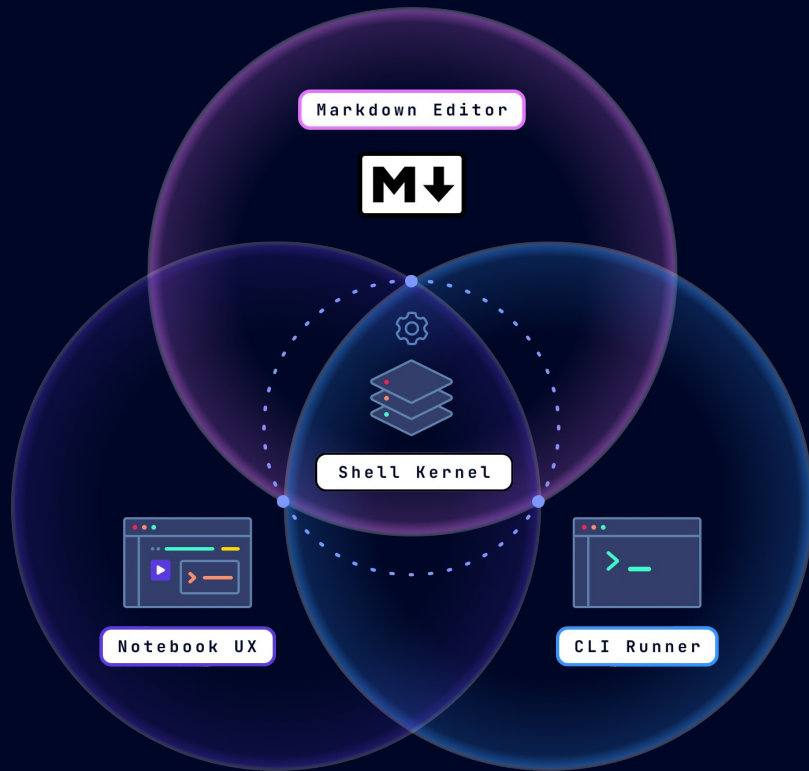


star-history.com




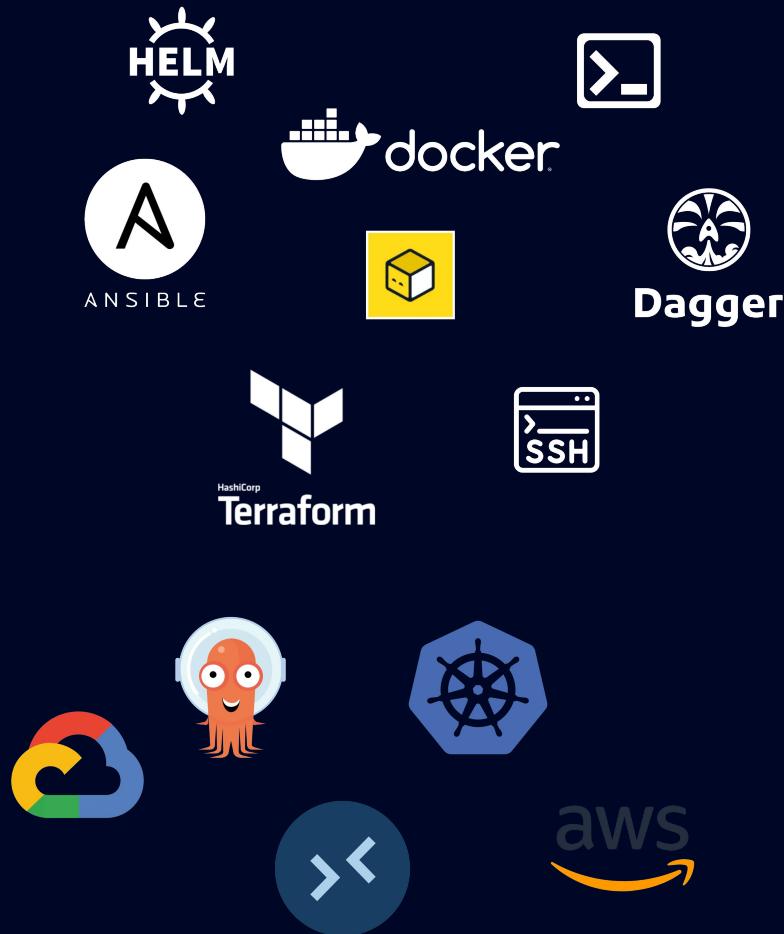
# Architecture

- Repos: stateful/runme, stateful/vscode-runme
- A serializer that transforms Markdown into executable cells with input and output
- A universal task runner that's native to markdown and works with all language-specific runners
- A portable runner interface that supports multimodal clients
- A kernel that retains state across execution in sessions akin to a terminal
- A raw-Markdown editor (inside VS Code) client for the runner
- A CLI client for the runner
- A notebook client (inside VS Code) for the runner
- A visual Markdown viewer and editor (inside VS Code)
- A cloud-native web component renderer that can embed any cloud resource



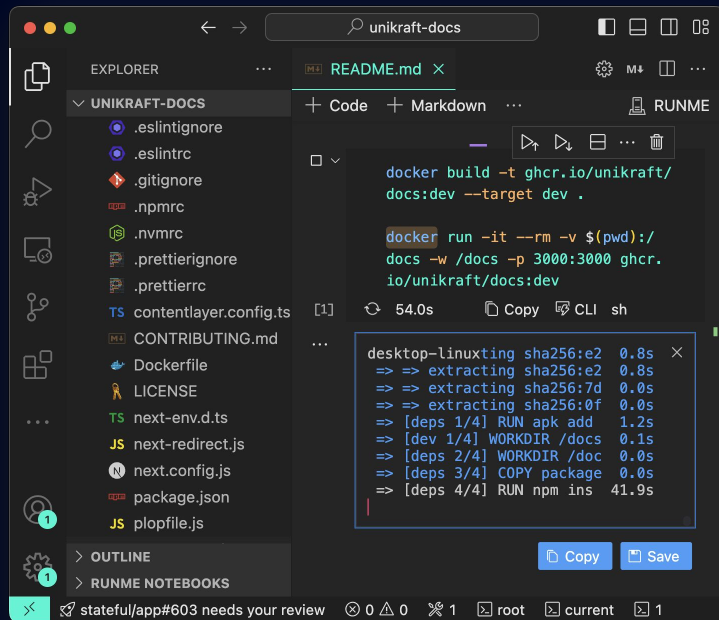
# Key features

- Create workflows from Markdown
- Configure exec behavior
- Use your tasks (make, npm, etc.)
- Runnable cells in any language
- Cloud integrations
- Custom output rendering
- Define and validate your .ENV
- VS Code/Codespace 
- CLI, Github Action (Test in CI/CD)



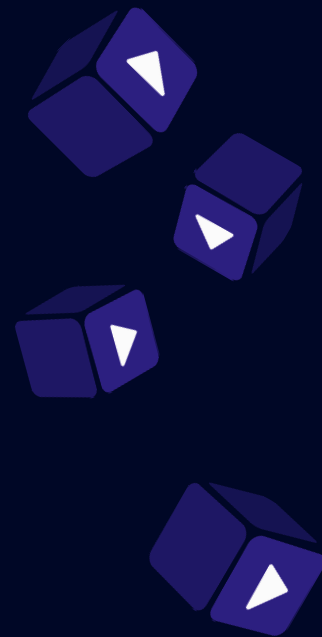
Demo

# Transforming a Runbook



# Roadmap

- .ENV types, validation and forking
- Front-matter UI (doc wide settings)
- CLI runner V2
- Cloud integration surface area
- SaaS integrations
- More shells
- Windows (non WSL)





# stateful

Cloud services for self-serve  
DevOps, and collaboration.

Find, launch, run and debug your  
internal eng docs as Notebooks.

Shameless plug for [stateful.com](https://stateful.com)

The screenshot displays the stateful.com web application interface. On the left is a sidebar with navigation options: Home, ARTIFACTS (Markdowns, Notebooks, Commands, Conversations, Bookmarks), DISCOVER (Search, Projects, Chat (AI)), OBSERVE (Activity, Analytics), and MANAGE (Settings). The main content area is titled 'Notebooks' and shows a list of notebooks with columns for name, age, and a 'main' button. The notebooks listed are: new.md (2w), infra.md (3w), outputs.md (3w), tf.md (3w), CONTRIBUTIN... (1m), example.md (1m), and neovim.md (1m). The 'new.md' notebook is selected, and its preview is shown on the right. The preview content includes a 'Google Cloud Login' section with instructions and a code snippet, and an 'Initialize Terraform' section with instructions and a code snippet. The right sidebar shows details for the selected notebook, including 'Add description', 'Created' (3 weeks ago), 'File' (docs/infra.md), 'Repository' (git@me:stateful), 'Branch' (main), 'Commit' (b63955c161f), 'Identity Type' (lifecycle), 'Identity Value' (01HPHHQ3J7), 'Session ID' (01J44RF5V5), and a 'Share on Slack' button.

stateful

Cloud services for self-serve  
DevOps, and collaboration.

Find, launch, run and debug your  
internal eng docs as Notebooks.

Shameless plug for [stateful.com](https://stateful.com)

stateful

Notebooks

new.md 2w  
infra.md 3w  
outputs.md 3w  
tf.md 3w  
CONTRIBUTIN... 1m  
example.md 1m  
neovim.md 1m

Google Cloud Login

Log in to your Google Cloud account.  
This step is essential to grant the  
necessary permissions for managing  
Google Cloud resources.  
\$ gcloud auth login  
Note: Remember to use your company's  
email address when prompted.

Initialize Terraform

Initialize Terraform with the following  
command:  
\$ terraform init  
To verify your setup, run:  
\$ terraform refresh

# Ran on 2024-08-01 10:40:40-03:00  
Acquiring state lock. This may  
take a few moments...

Details Conversation (0)

Add description

Created 3 weeks ago

File docs/infra.md

Repository git@me:stateful

Branch main

Commit b63955c161f

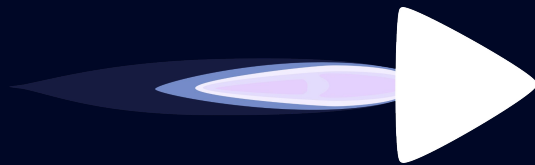
Identity Type lifecycle

Identity Value 01HPHHQ3J7

Session ID 01J44RF5V5

Add a tag

Share on Slack



Join us! 😊

