



# My App Broke

And it's all the API's fault





# Shy Ruparel

Developer Evangelist | Contentful

@ShyRuparel


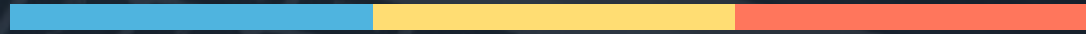
He/Him



Don't let a CMS get in the way of shipping software.

Contentful provides a content infrastructure that enables teams to power content in any digital product.

# Let's talk about Software Engineering

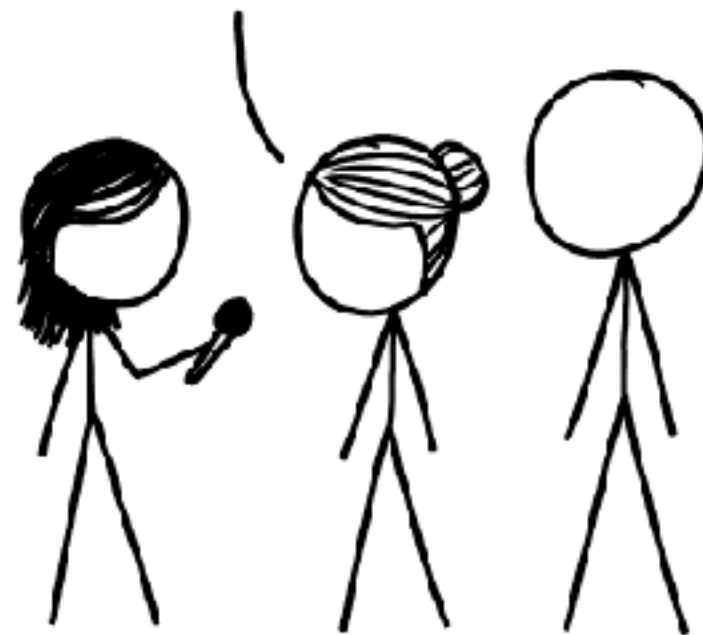
The background is a dark, semi-transparent image of a workspace. It features a laptop with a screen displaying lines of code, a hand holding a smartphone, and a pair of glasses resting on a desk. The overall aesthetic is professional and tech-oriented.

Regardless of  
what you make

Something is  
gonna break

ASKING AIRCRAFT DESIGNERS  
ABOUT AIRPLANE SAFETY:

NOTHING IS EVER FOOLPROOF,  
BUT MODERN AIRLINERS ARE  
INCREDIBLY RESILIENT. FLYING IS  
THE SAFEST WAY TO TRAVEL.



ASKING BUILDING ENGINEERS  
ABOUT ELEVATOR SAFETY:

ELEVATORS ARE PROTECTED BY  
MULTIPLE TRIED-AND-TESTED  
FAILSAFE MECHANISMS. THEY'RE  
NEARLY INCAPABLE OF FALLING.



ASKING SOFTWARE  
ENGINEERS ABOUT  
COMPUTERIZED VOTING:

THAT'S TERRIFYING.



WAIT, REALLY?

DON'T TRUST VOTING SOFTWARE AND DON'T  
LISTEN TO ANYONE WHO TELLS YOU IT'S SAFE.

WHY?

I DON'T QUITE KNOW HOW TO PUT THIS, BUT  
OUR ENTIRE FIELD IS BAD AT WHAT WE DO,  
AND IF YOU RELY ON US, EVERYONE WILL DIE.



THEY SAY THEY'VE FIXED IT WITH  
SOMETHING CALLED "BLOCKCHAIN."

AAAAA!!!

WHATEVER THEY SOLD  
YOU, DON'T TOUCH IT.  
BURY IT IN THE DESERT.  
WEAR GLOVES.



# Stuff breaks



Tamara McCleary  
@TamaraMcCleary

Follow

The cause of the [#Amazon](#) Web Services failure that resulted in parts of the internet failing? A typo. [techspot.com/news/68381-cau](https://techspot.com/news/68381-cau) ... [#awsoutage](#)



12:07 AM - 4 Mar 2017

19 Retweets 7 Likes



MacBook



#kimsconvenience

You can do better






**Use APIs**



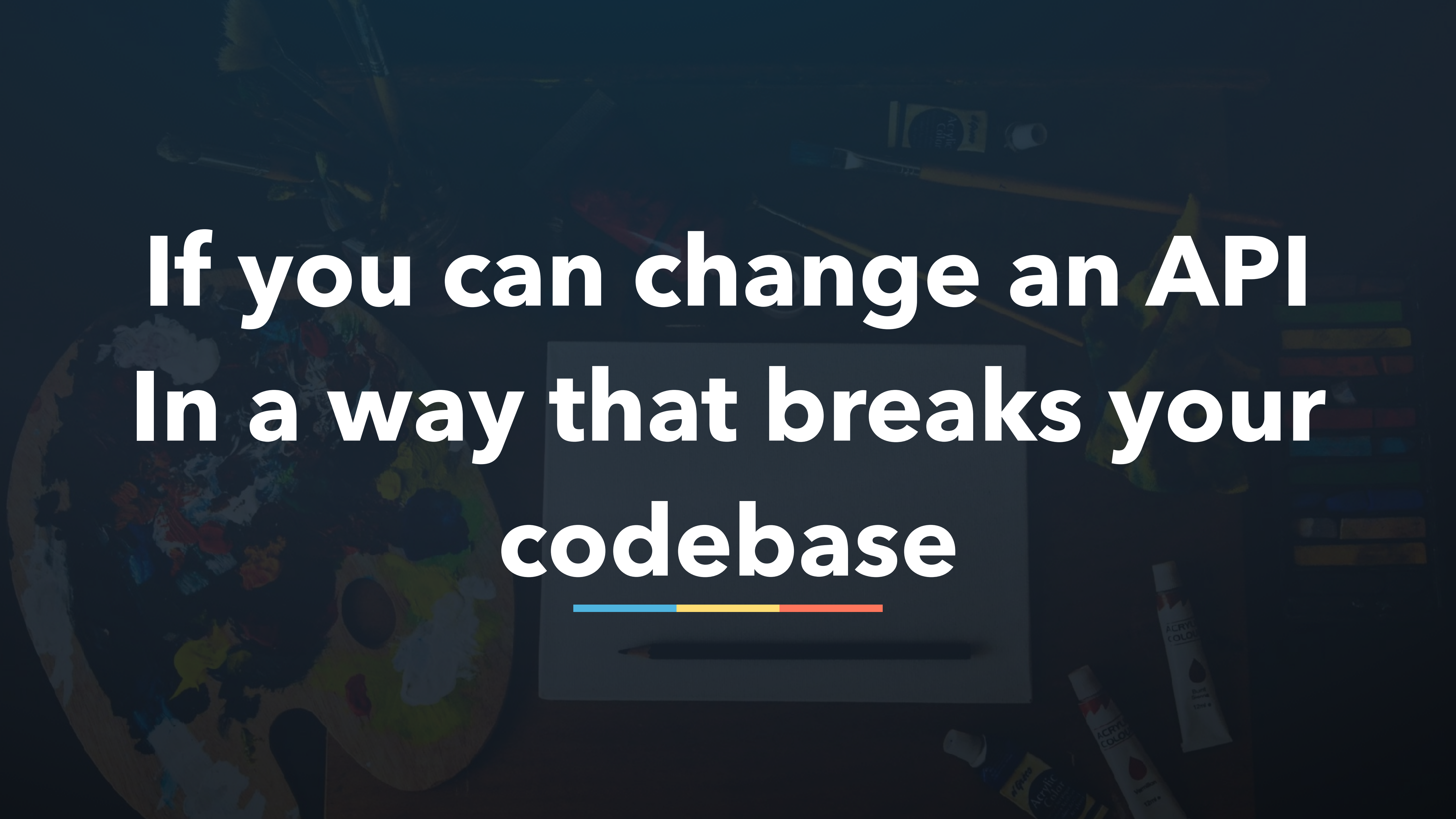
**Writes APIs**



A woman with long hair, wearing a dark blazer, is standing in profile and writing on a whiteboard with a blue marker. The whiteboard has some faint, illegible writing on it. The background is a dimly lit room with a window on the right. The overall image has a dark, blue-tinted overlay.

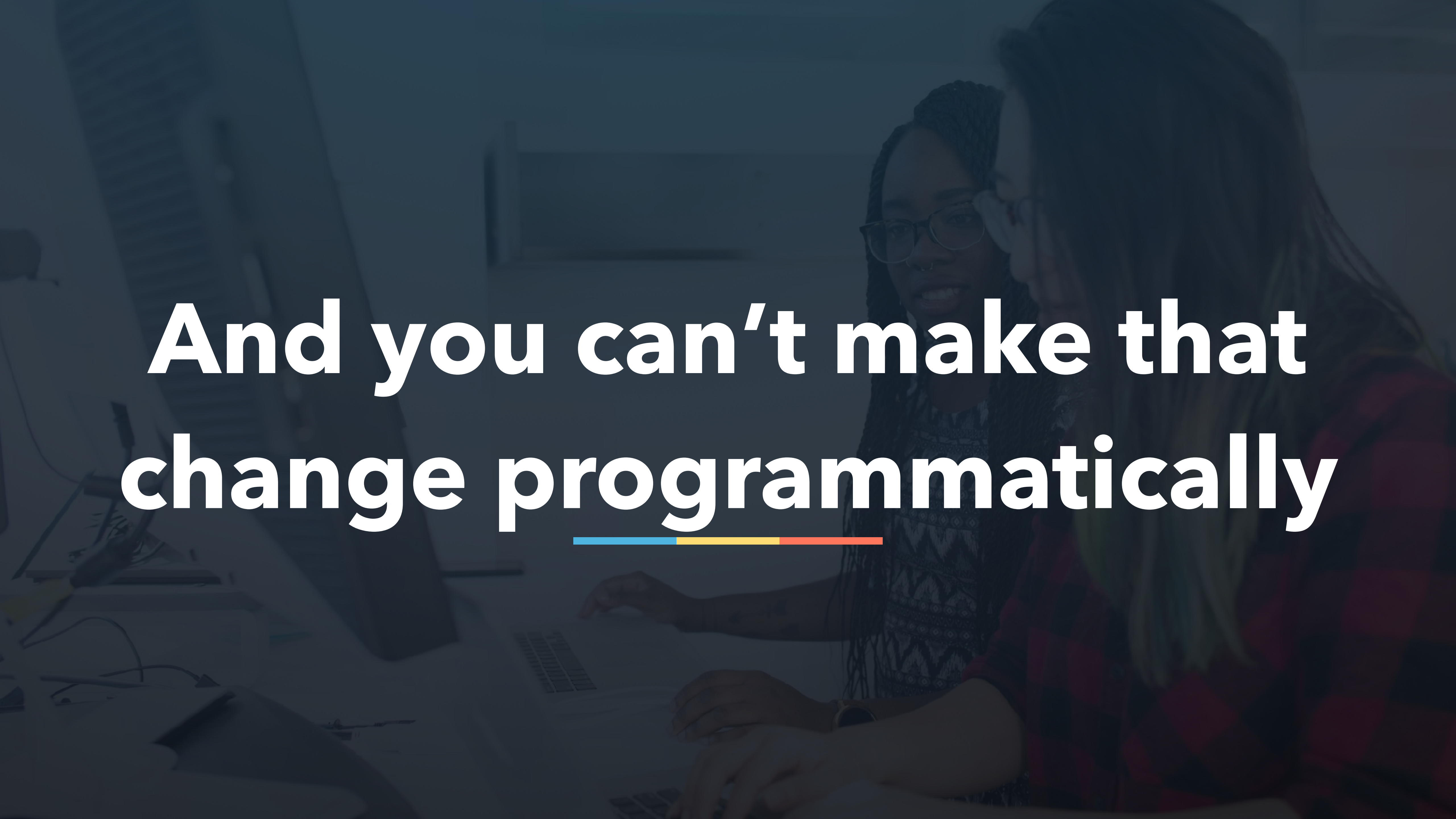
**How do we  
Minimize risk?**

---

The background is a dark, artistic composition featuring a palette with various colors of paint, several tubes of acrylic paint, and brushes. The overall aesthetic is creative and artistic.

**If you can change an API  
In a way that breaks your  
codebase**





**And you can't make that  
change programmatically**

---

#kimsconvenience

Get. Out.



# Let's take a look at Databases

A hand holding a pen over a document with the text "What W... Google Do?" and "GOOGLE DO?" visible. The document is part of a book or manual, and the hand is positioned as if about to write or highlight. The background is dark and out of focus, showing a person's face and hands.



# DON'T FUCK IT UP

Critical to business

Store user data

In Sync with Codebase

Outages translate to \$\$ lost

**Production should be  
Non-Trivial to change**



**Fun Fact:**  
**SQL is 44 years old**





# Database Migrations

No more developers manually alternating a DB

Schema alterations as code

Version Control for Migrations

Repeatable and Testable

Cool.

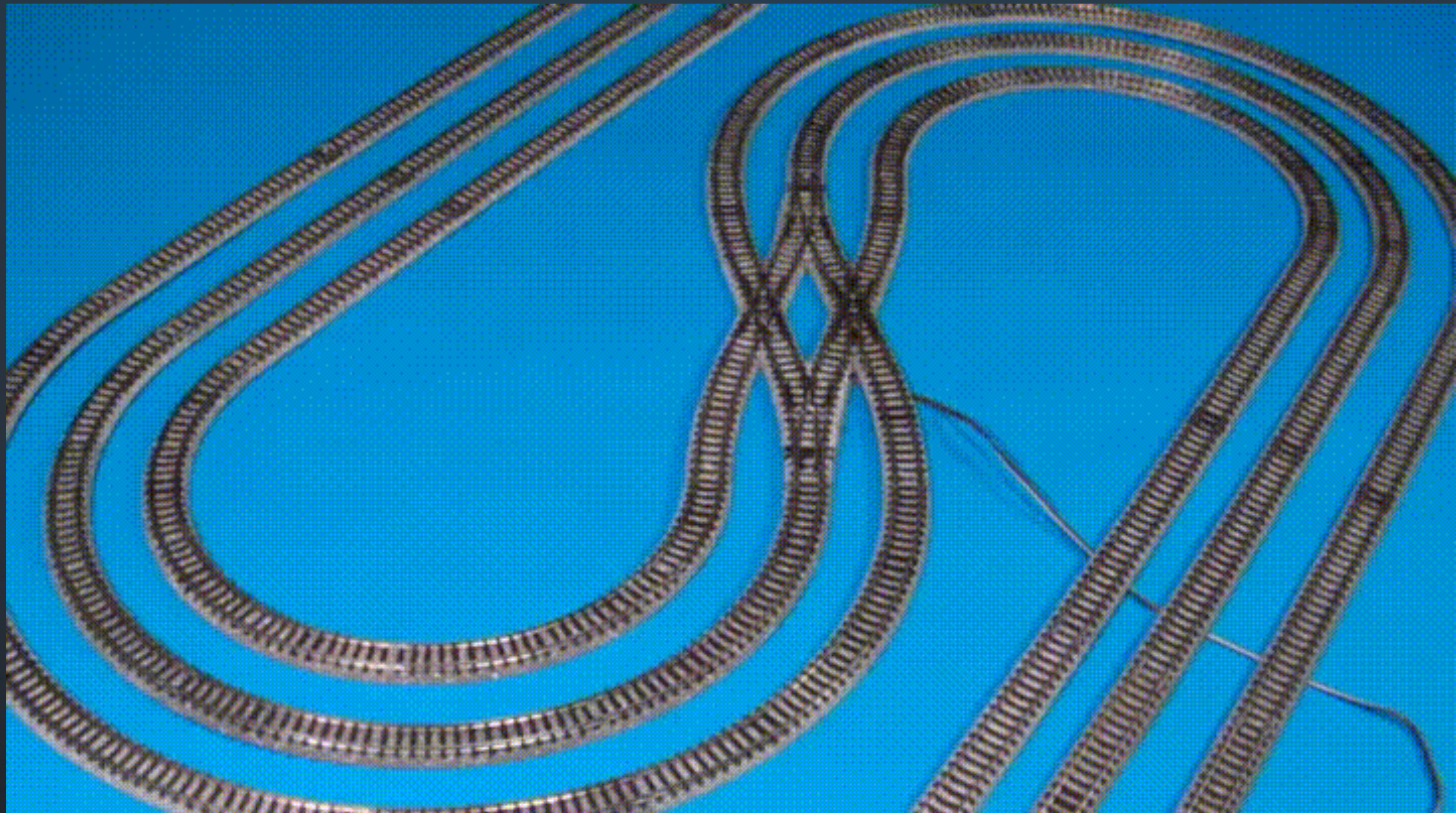
Can we talk about APIs again?




#kimsconvenience

I don't think so





**Can we learn anything from  
Continuous Integration?**



# Simple CI Pipeline



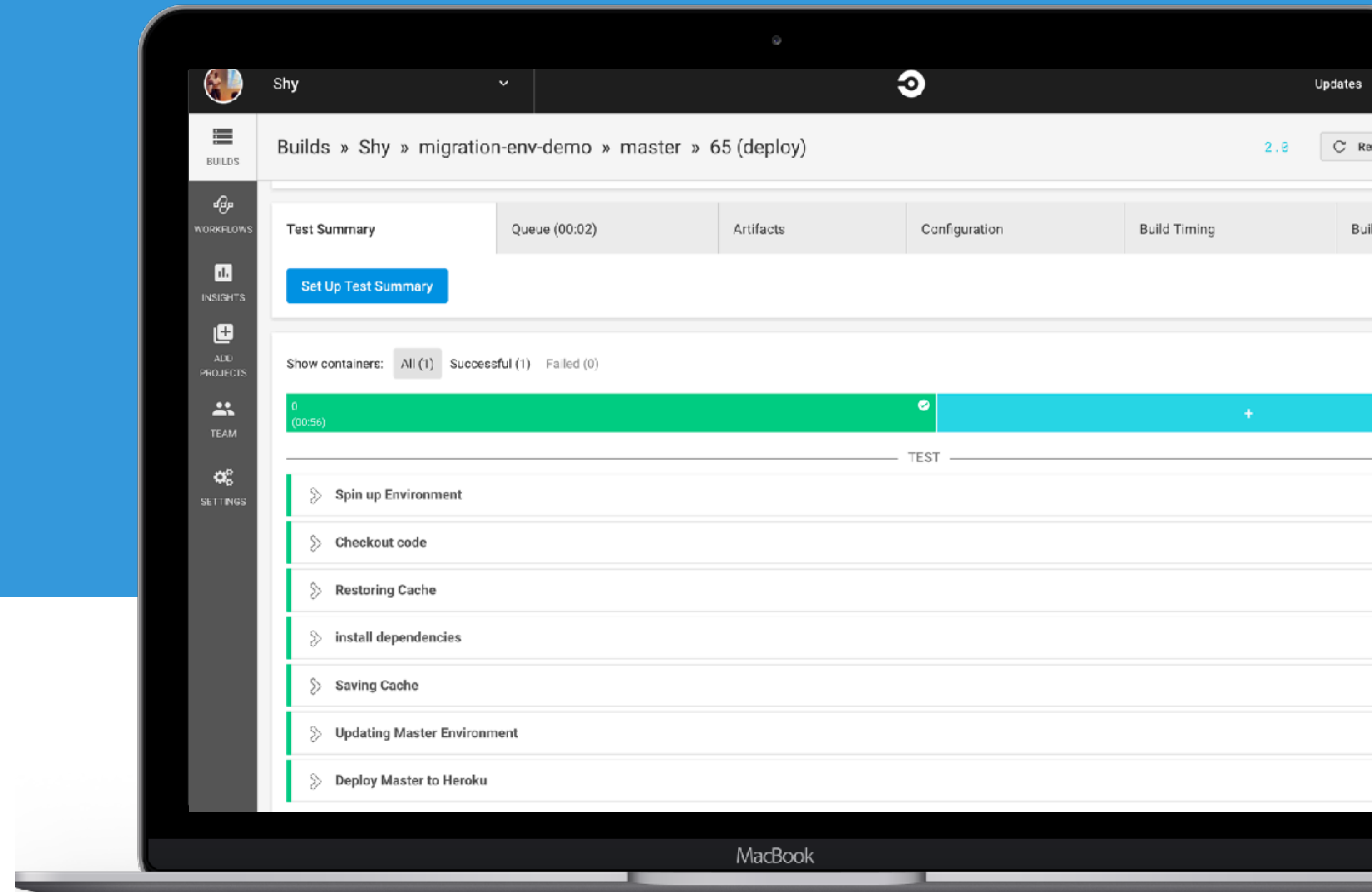
Build

Test

Deploy

# WHAT CI DOES

- Forces testing and checks on changes
- Automates deployment
- Cancelable
- Repeatable





**Saves Time**



**Creates Trust**



A dark, atmospheric photograph of a space shuttle launch. The shuttle is positioned vertically in the center, with a large plume of white smoke and fire at its base. The background is a deep blue, suggesting a twilight or night sky. The overall mood is dramatic and industrial.

**Process is key**

---

**Ok.**

**Now we can talk about**

**External APIs**

---

#kimsconvenience

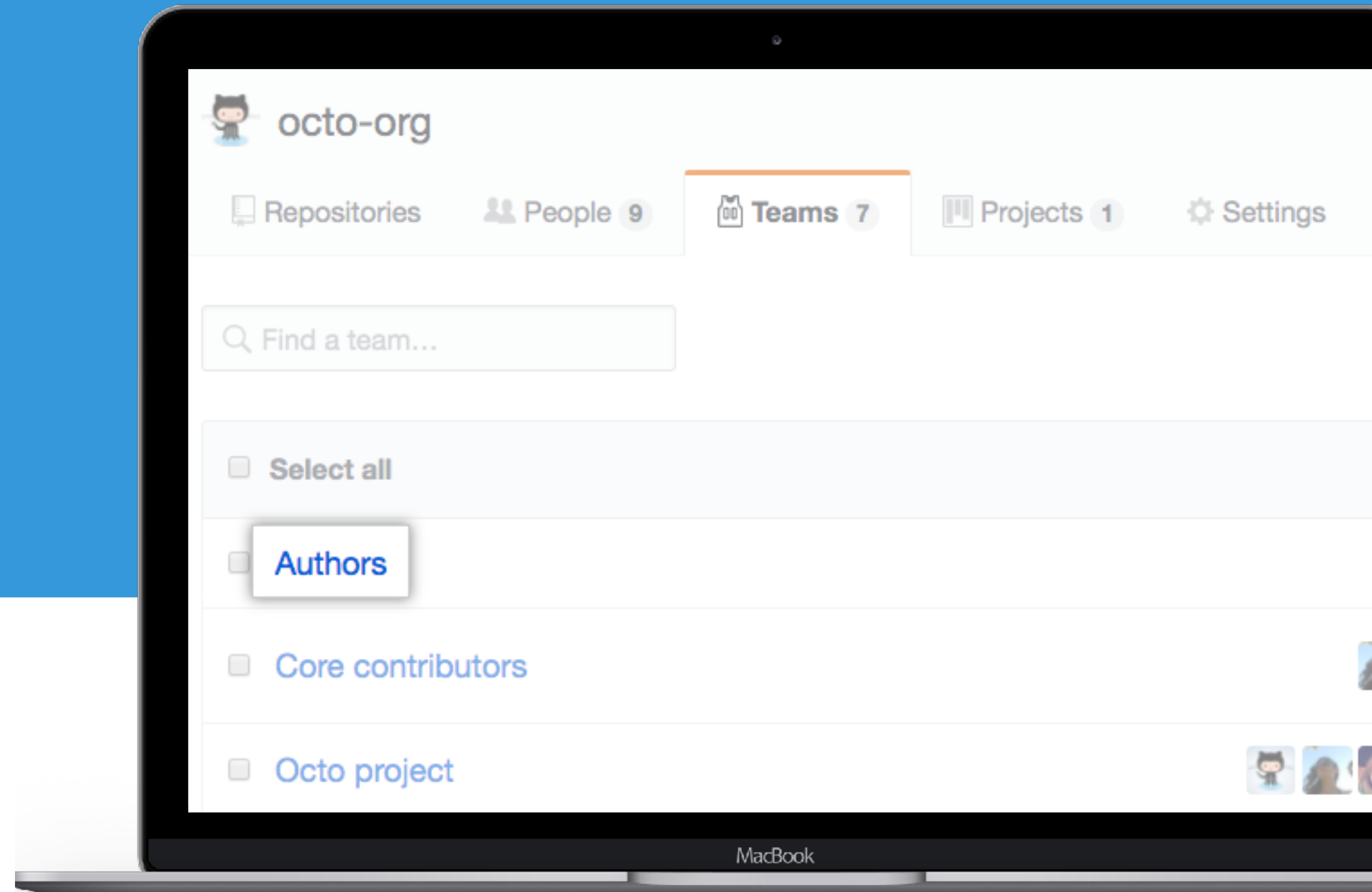
Okay! Now we talking!



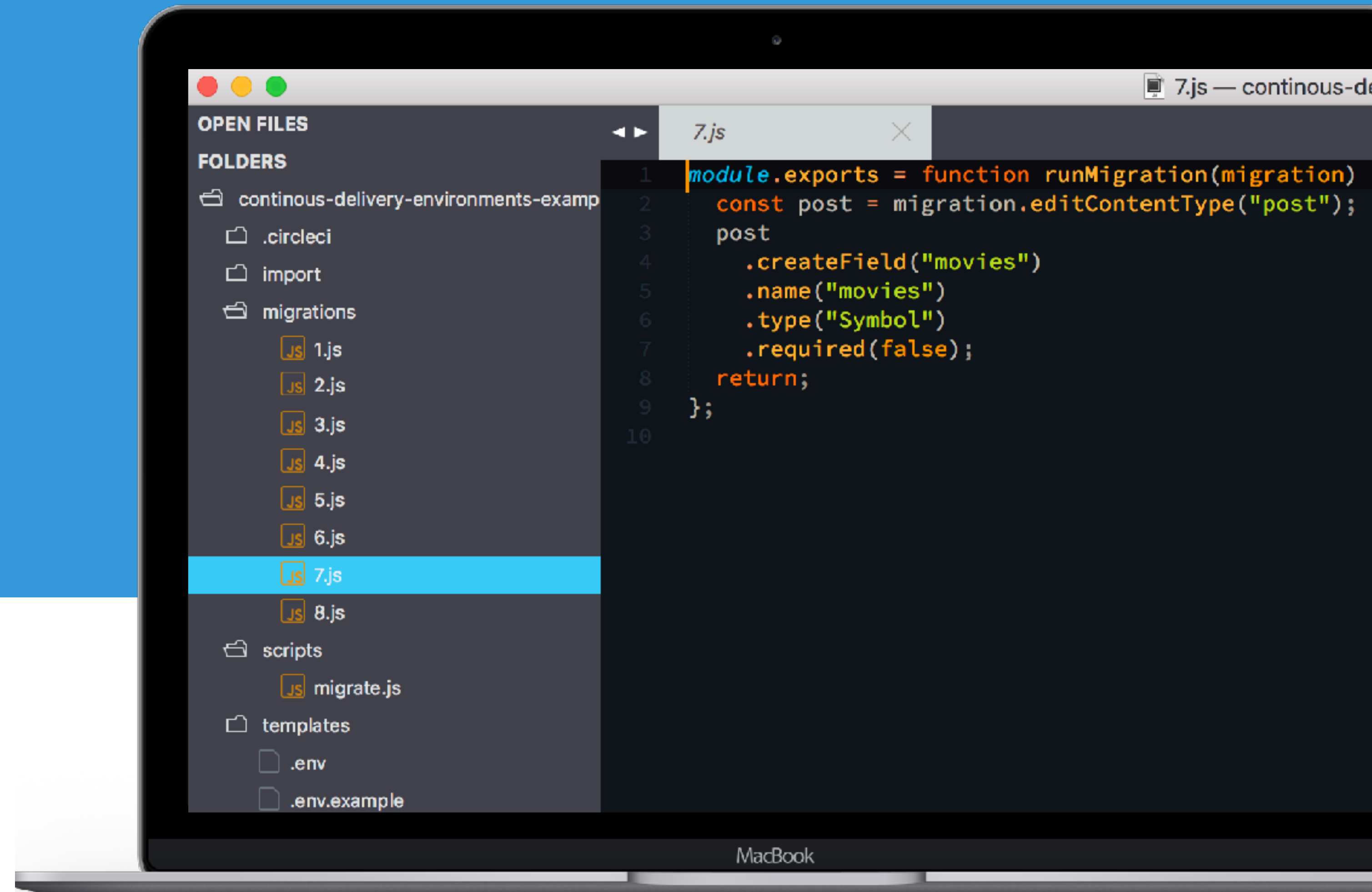
# Treat APIs like Databases



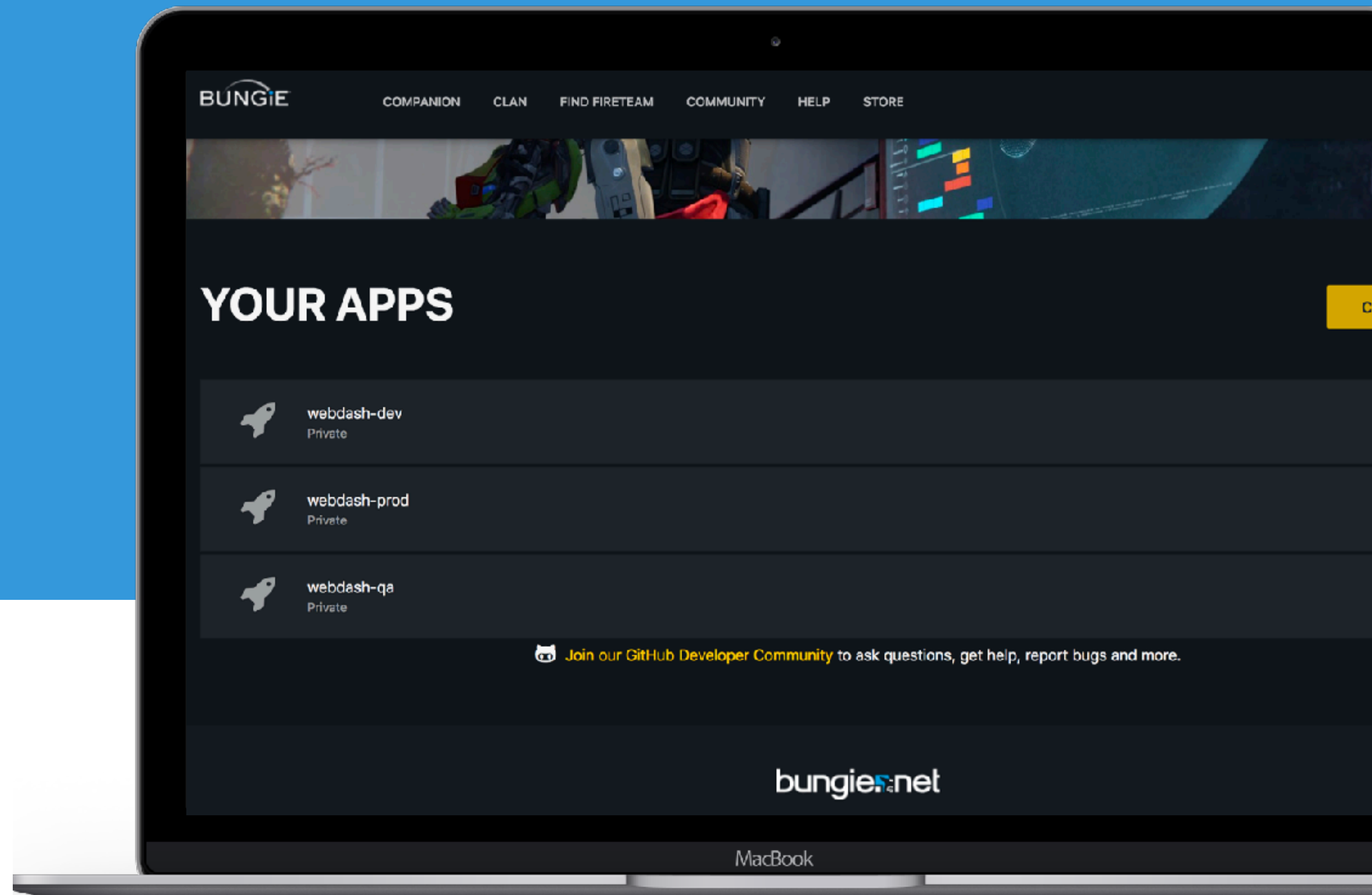
**MAKE PRODUCTION  
NON-TRIVIAL TO EDIT**



# WRITE/ENABLE PROGRAMMATIC API MIGRATIONS



# API KEYS FOR EVERY BRANCH



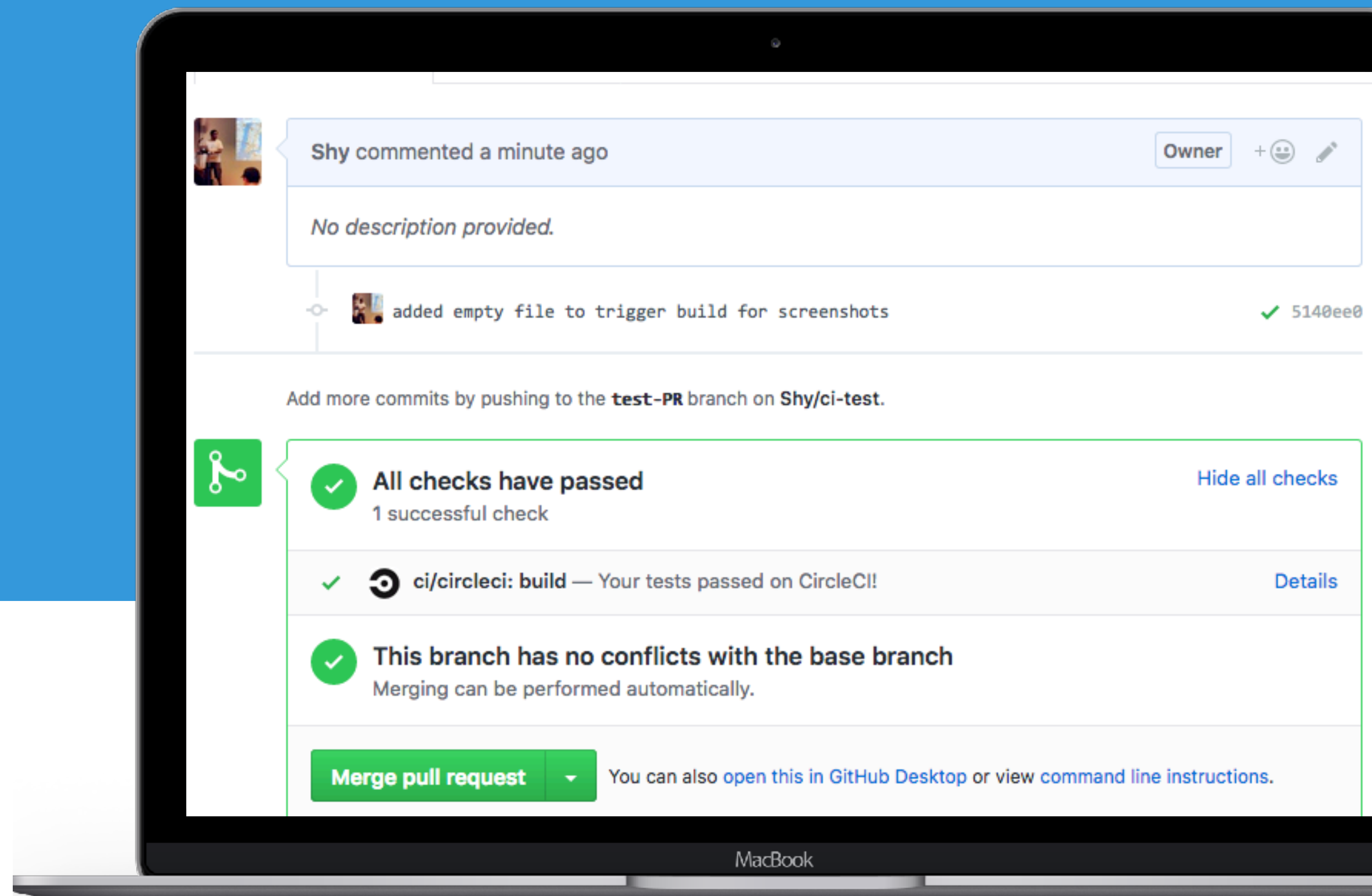


# Automate your changes via Continuous Integration

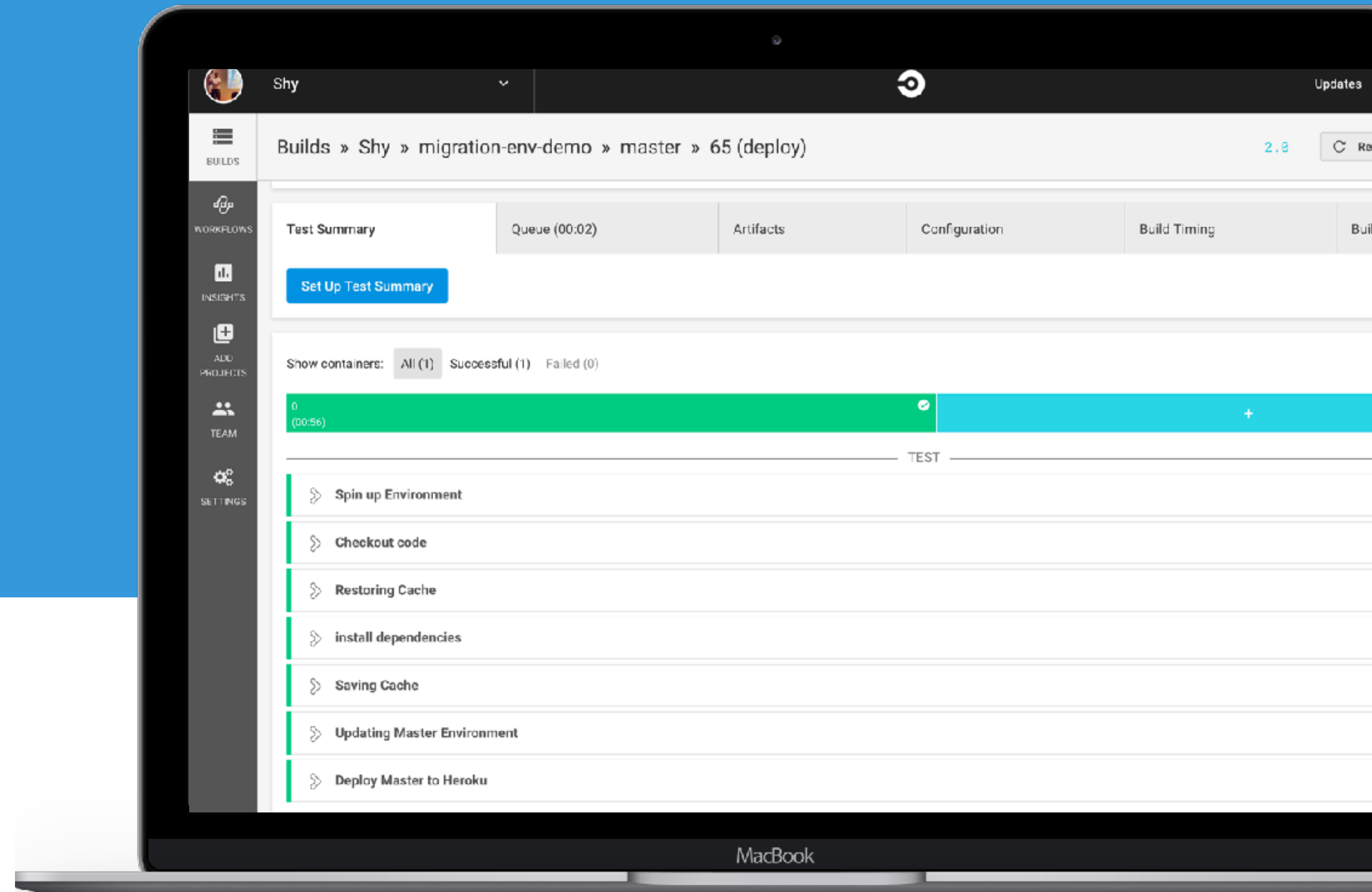
---



# TEST YOUR APIS



# MAKE CHANGE PART OF DEPLOYMENT



# Less Simple CI Pipeline

---

Build

- *Use Branch API Key*
- *Implement API Configuration Changes*

Test

- *Run tests that use that API*

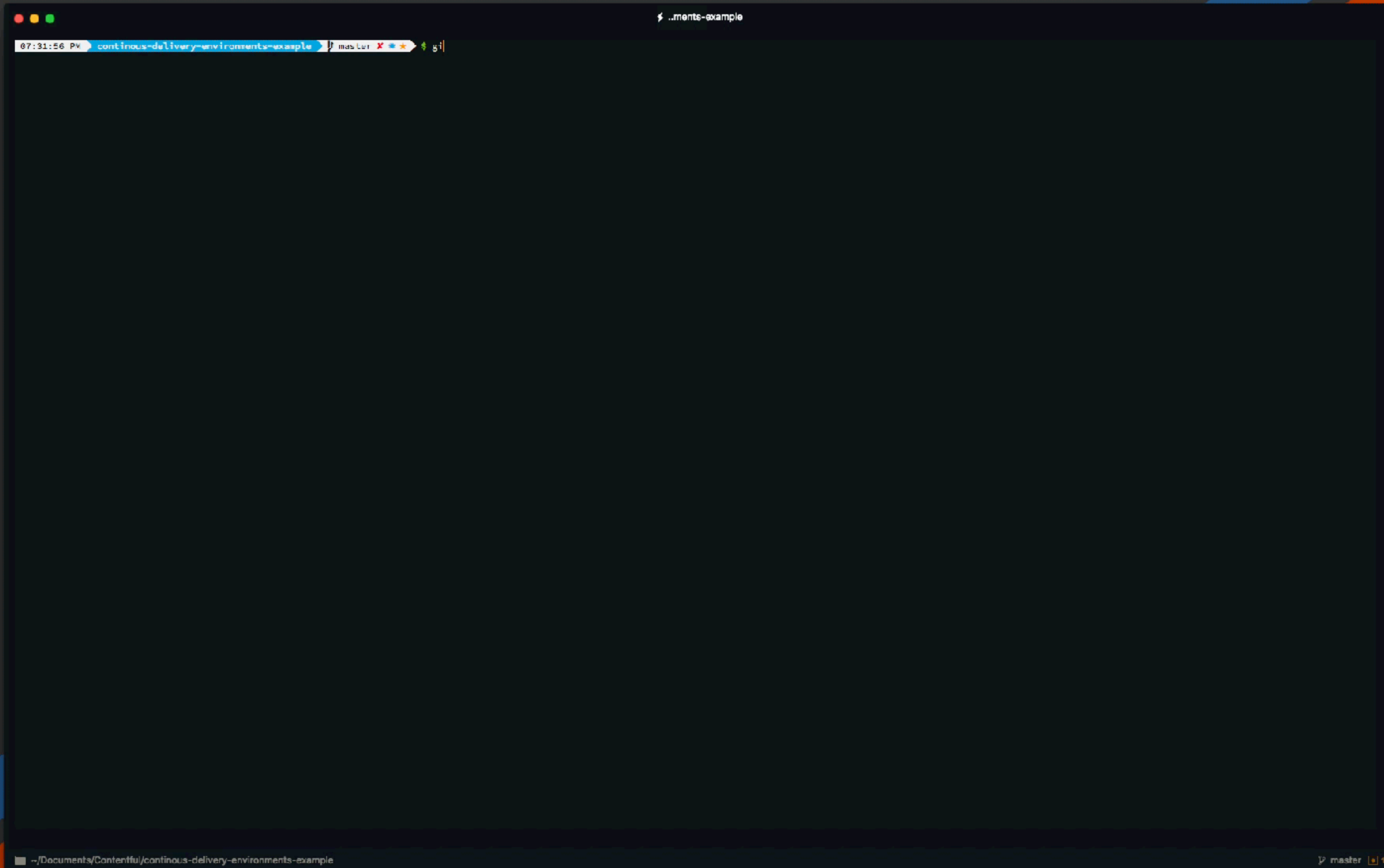
Deploy

- *Update Production API Key*



# Let's look at a practical Workflow

---





# Shy Ruparel

---

@ShyRuparel

[github.com/shy/talks](https://github.com/shy/talks)

[Image Credits](#)

[wocintechchat.com](http://wocintechchat.com)

Kim's Convenience

XKCD