

# Danger: smarter, faster and more efficient code reviews

Camila Maia

#### Who am I?



### loadsmart

Backend Engineer in the Mobile
 Squad



Sinatra

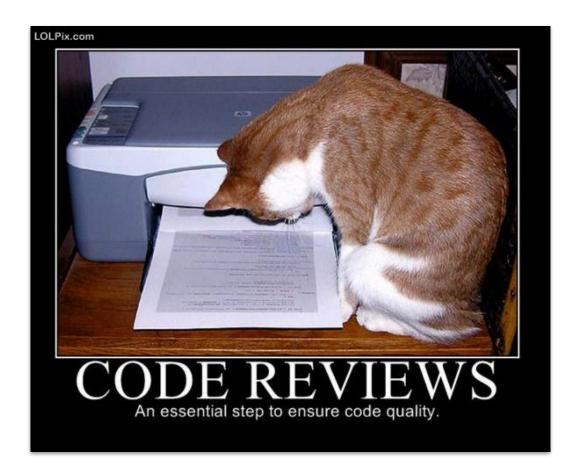








#### **Code Reviews**





#### **Code Reviews:**

They take time!





# How can we make code reviews smarter, faster and more efficient?

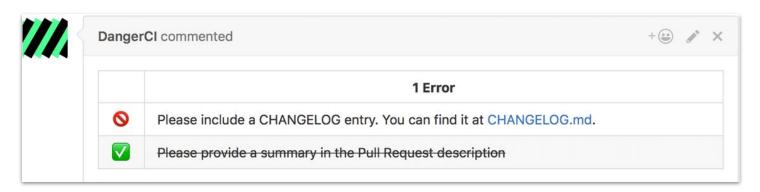
#### Automation

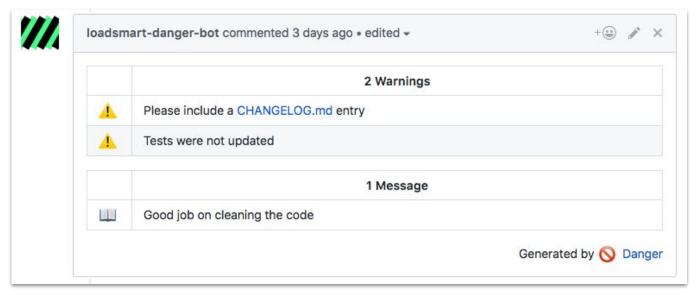














- An automated reviewer of your code
- Ruby gem
- Runs after your CI automating your team's conventions surrounding code review





**Peer Review Tools** 



Source Control Management (SCM)



**CI Systems** 

### It is as generic as you want

Like Unit Tests, but for your Team Culture.

Formalize your Pull Request etiquette.



- More precision on PRs
- Take less time on code reviews

# With Danger, you can leave humans to think about harder problems.



#### How does it work?

#### Domain Specific Languages (DSL's)

MESSAGING

GIT

DANGER

SCM

```
# Make it more obvious that a PR is a work in progress and shouldn't be
merged yet
if github.pr title.downcase.include? "[wip]"
    warn("PR is classed as Work in Progress")
end
# Warn when there is a big PR
warn("Big PR") if git.lines of code > 500
```

warn("Please, provide a description to your PR") if github.pr body.empty?

# Warn if there's no description

0

#### IMPORT FROM A LOCAL FILE REFERENCE

danger.import\_plugin("danger/plugins/watch\_plugin.rb")

RETURNS THE NAME OF THE CURRENT SCM PROVIDER BEING USED.

danger.scm\_provider

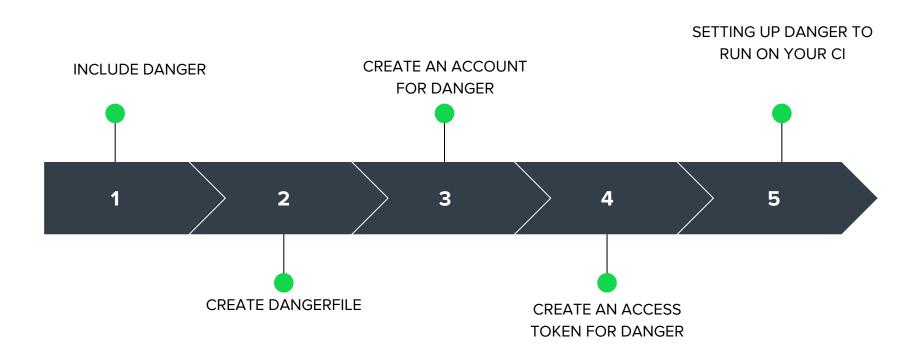
#### **Plugins**

ANDROID LINT APKANALYZER PROSE AUTO LABEL CHANGELOG CLOROX **ESLINT** COMMIT LINT CONFLICT CHECKER **DUPLICATE LOCALIZABLE STRINGS** FINDRUGS JAZZY **JENKINS** JIRA JUNIT LGTM MENTION MISSED LOCALIZABLE STRINGS PRONTO RAILS BEST PRACTICES REEK RUBOCOP SHELLCHECK PEP8 SIMPLECOV JSON SLACK SLATHER SLIM LINT SWIFTLINT TAILOR THE CODING LOVE TODOIST TRANSIFEX **XCODE SUMMARY** XCODEBUILD XCOV HLINT **WELCOME MESSAGE** CHECKSTYLE FORMAT TEXTLINT



### How to integrate it?

### Setup





### **Include Danger**



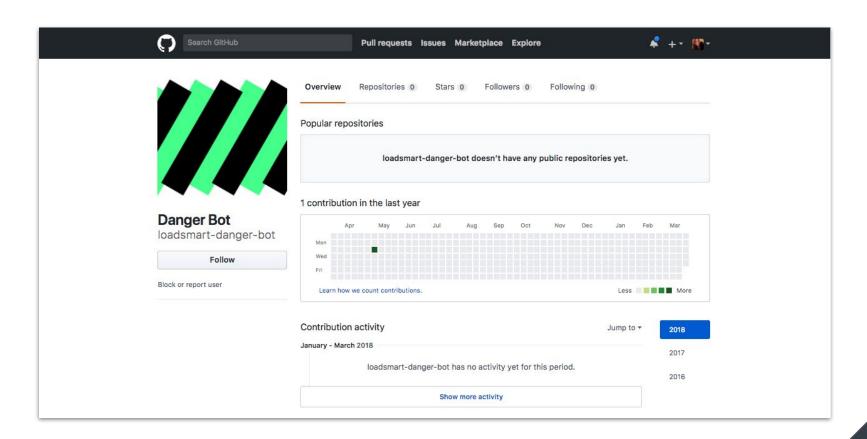
#### **Create a Dangerfile**



#### **DON'T ALLOW A FILE TO BE DELETED**

deleted = git.deleted\_files.include? "my/favourite.file"
fail "Don't delete my precious" if deleted

#### Create an Account and an Access Token for Danger

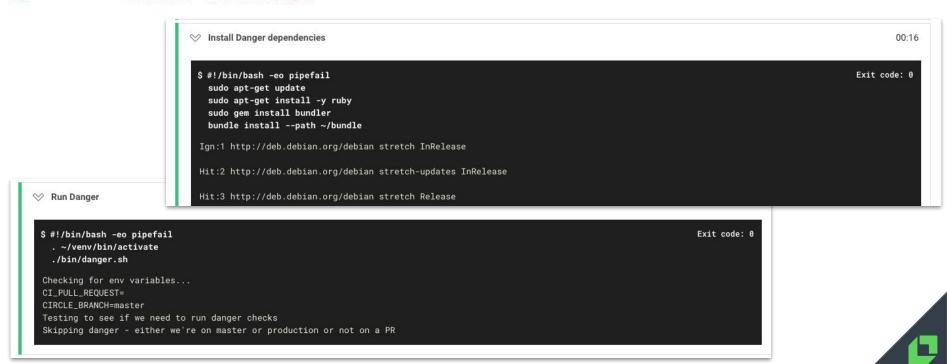


#### Setting up Danger to Run on your Ci

#### test:

#### override:

- bundle exec danger





### wetransfer











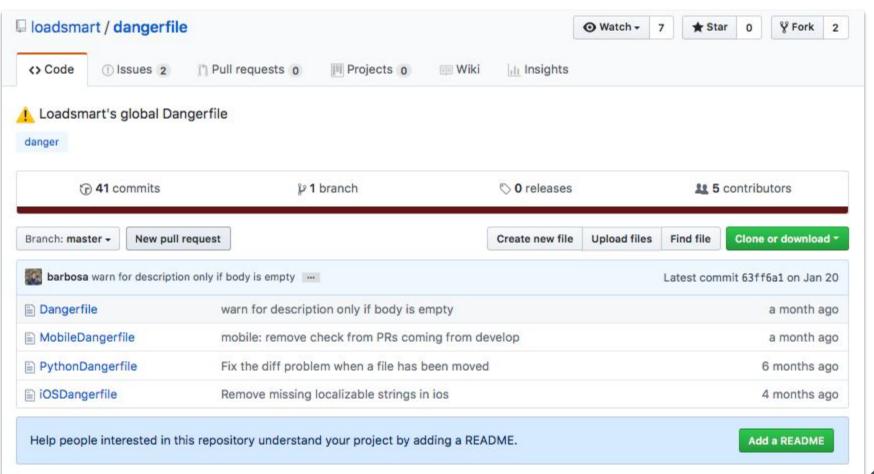
### **Our Experience**

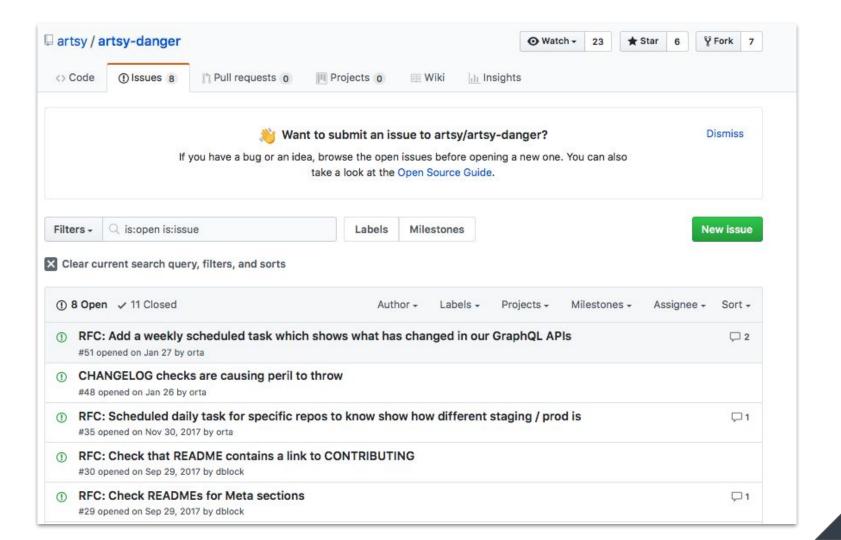
- Gradual integration works better
- The culture is the most important part
- It takes time to coworkers get engaged



### **Our Experience**

- Pair programming
- Check other projects
- It is awesome for open source projects
- Inheritance might be not that easy





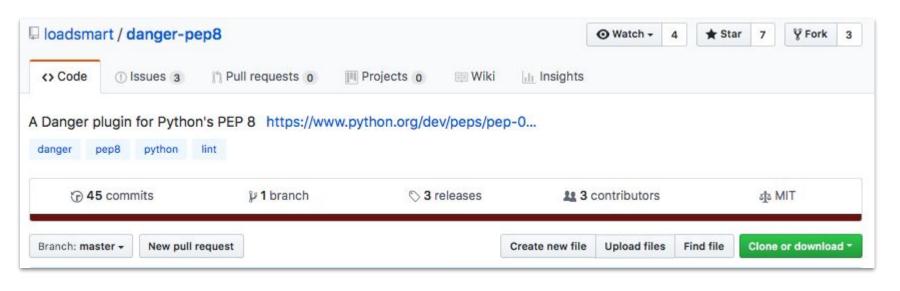


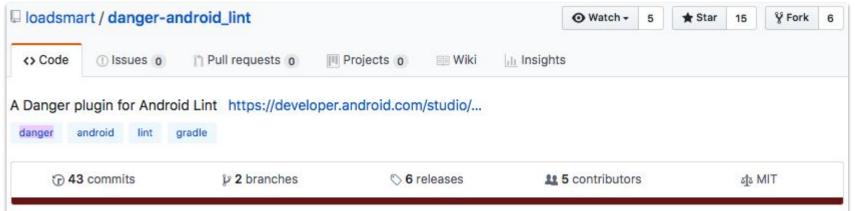


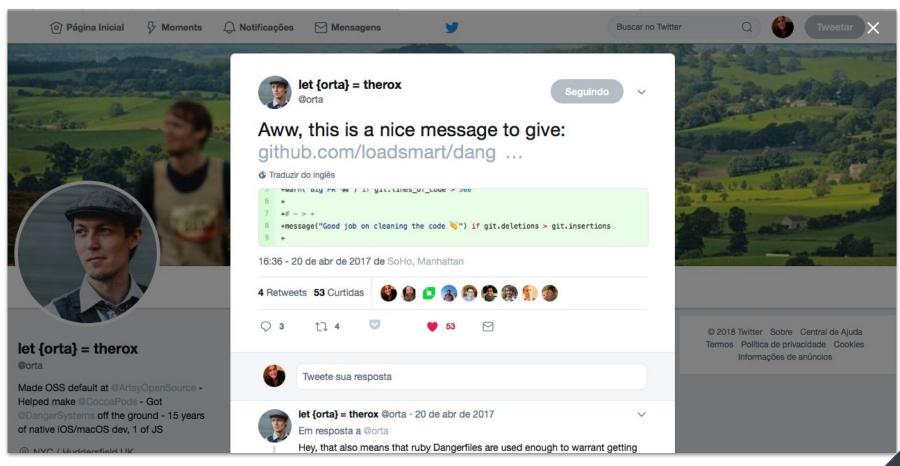




### Community









What is next?



What?

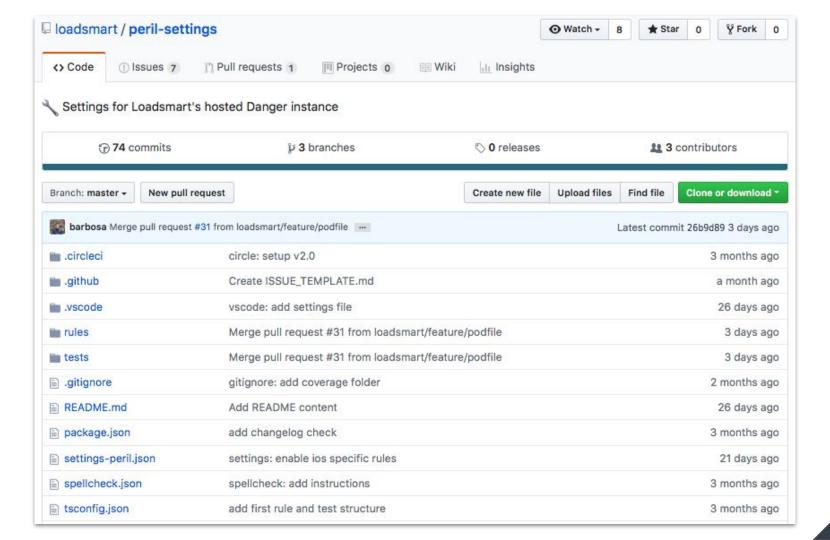
- Hosted Danger
- Typescript
- Danger JS
- It is a really recent project



- More flexibility:
  - Scheduler, for example
- Settings are more powerful
  - Composition besides inheritance



- Easier to test
- Installation can literally be a single click on the website
- It is a GitHub App (not an account)
  - no need to consider scope for tokens
  - no need to ensure bot has access to repo



# Stop saying "you forgot to ..." in code review



# THANK YOU FOR YOUR TIME

#### **WE ARE HIRING!**

https://loadsmart.com/careers/



