

Code Reviews con git avanzado

Pepe Doval

SetPay

24 de febrero de 2016



Esta obra está suxeita á licencia Reconocimiento-CompartirIgual 4.0 Internacional de Creative

Commons. Para ver unha copia desta licencia, visite <http://creativecommons.org/licenses/by-sa/4.0/>.

Índice

1 Code Reviews

- Introducción
- Que é unha Code Review
- Como facer unha Code Review

2 Git Avanzado

- Entendendo git
- Construíndo a historia
- Utilidades para review
- Boas prácticas

Code Reviews

Canta xentiña!



Son un mal programador



Figura: <http://csgardner.com/blog/2015/06/11/managing-software-developers/>

Pero tamén fun un crack



Figura: <https://www.apico.net/blog/how-bad-software-developer-are-you.html>

...é dicir, fun Junior

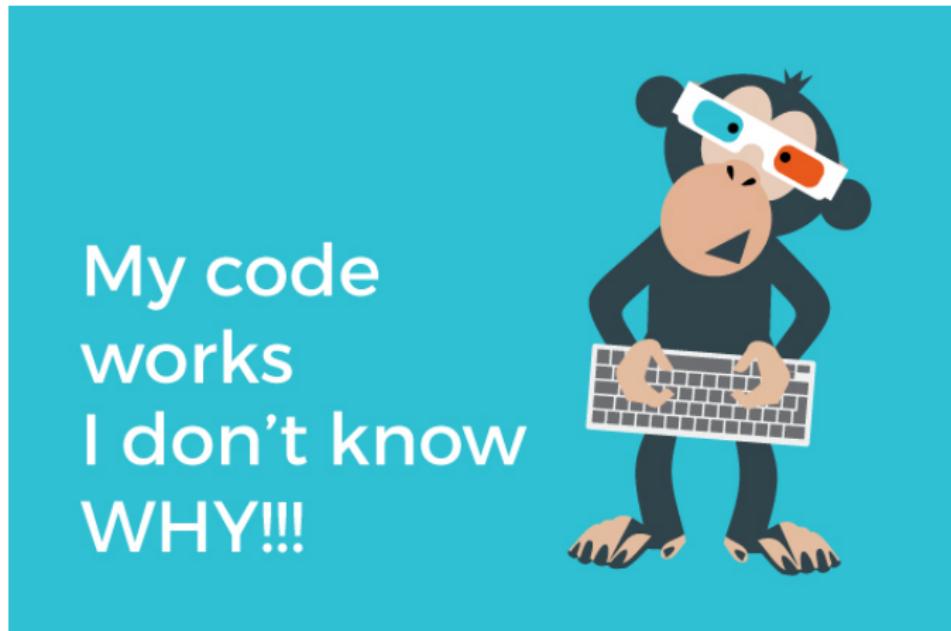


Figura: <https://www.apico.net/blog/how-bad-software-developer-are-you.html>

De que vai construír software



Figura: Wikipedia

Abracemos o cambio

O que hoxe está ben, mañá estará mal.

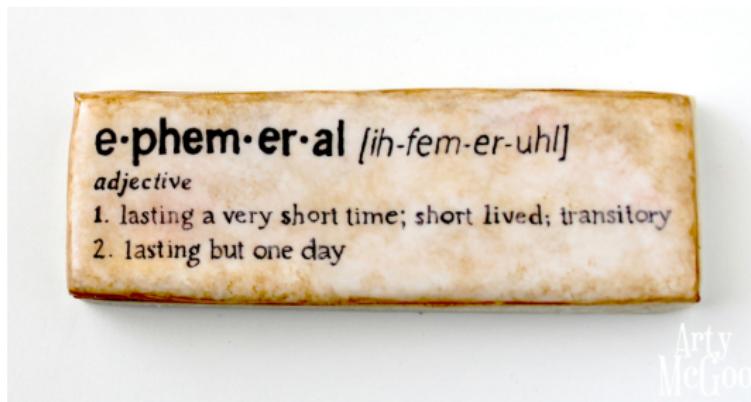


Figura: <http://www.artymcgoo.com/blog/2014/5/15/ephemeral-art>

Por que revisar código

- Atopar erros
- Mellorar o código
- Compartir coñecemento
- Compartir decisións
- Mellorar a comunicación do equipo

Mars Climate Orbiter



Figura: <http://visionlearningcommunity.blogspot.com/2012/09/tragedies-in-science-crash-of-mars.html>

O día que non comezou a Terceira Guerra Mundial

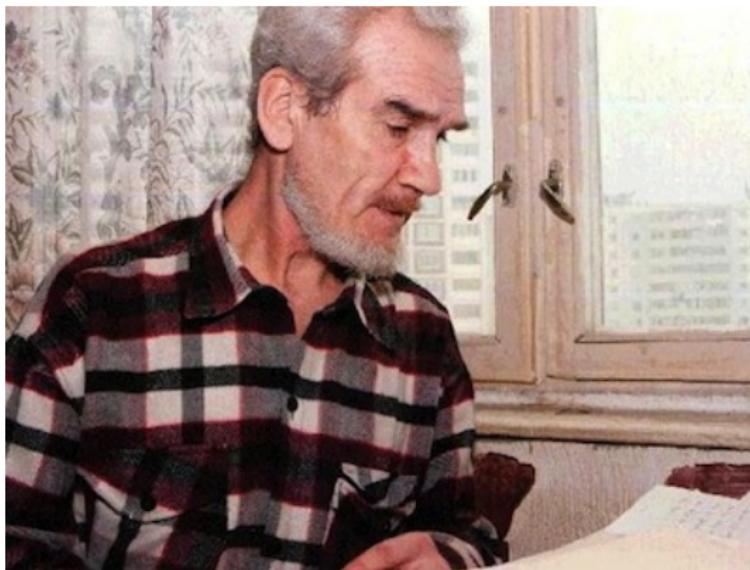


Figura: http://www.brightstarsound.com/world_hero/petrov_expanded_photo.html

Se non queredes revisar código...



Figura: Wikipedia

Que é unha Code Review



Figura: <https://www.youtube.com/watch?v=rHVIFOB5BpU>

Fases da Review

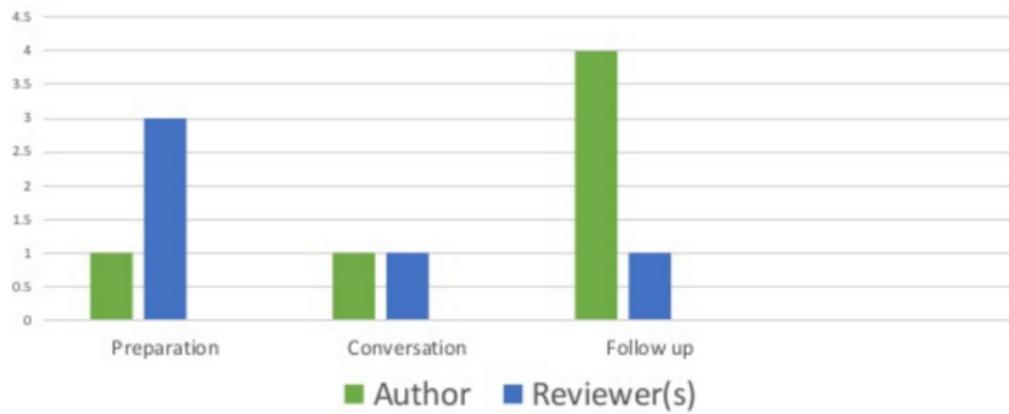


Figura: <http://www.slideshare.net/jprusakova/effective-codereview-color>

Limitade o tempo



Figura: <https://sites.psu.edu/siowfa14/2014/10/19/why-am-i-so-tired/>

Facede unha checklist

Add a to-do in "Review Verification Checklist copy"...

- The branches to review have the same name in all affected projects
- The branches to review are properly rebased to integration and have no expected conflicts to merge
- The resolved issue was included in the "unreleased" section of the CHANGELOG
- If there are format changes in source code they must be isolated in a separate commit
- The feature is deployed or easily deployable to a personal space, "How to test" if needed, apk for Android
- If any manual step is required to deploy the feature into production, it's included in the proper table in Manual Steps to Deploy
- All tests for the project are passing: gradle test + mocha
- Any new migration is reversible. Any required manual step with rollback is included in Manual Steps to Rollback Features
- If the issue is a bug, a regression test is included (see Testing Guidelines)
- The feature has been sufficiently documented
- The code meets our quality standards
- All review comments or questions are answered either on Bitbucket or a dedicated HipChat room

Figura: My Review Checklist

Medide



Figura: Wikipedia

Que midades!

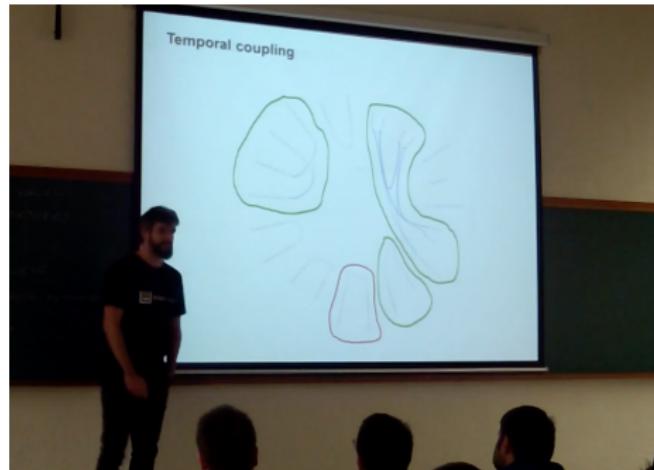


Figura: Your Code as a Crime Scene - <https://vimeo.com/154470784>

Medir é divertido, en serio



Outra métrica sinxela



Jef C^{laes}
@JefClaes



Seguir

Hypothesis: your core domain can be found in 1.
the code that changes most often 2. the
questions customers ask you most often

Ver traducción

RETWEETS

7

ME GUSTA

9



23:07 - 13 dic. 2015



...



Tips para reviewers

- Limita o tempo
- Fai unha checklist
- Mide para decidir que abordar

Fai pulls pequenas



Figura: <https://www.youtube.com/watch?v=rHVIFOB5BpU>

Revisa ti antes

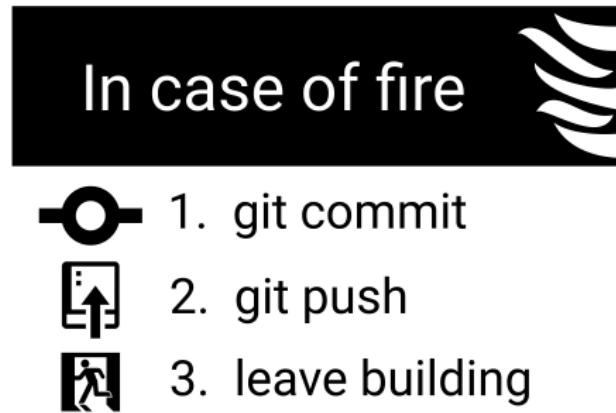


Figura:

https://www.reddit.com/r/ProgrammerHumor/comments/3nc531/in_case_of_fire

Automatiza

Nothing can stop automation



Figura: <http://www.runmode.com/automationhumor.html>

Fai tests de regresión

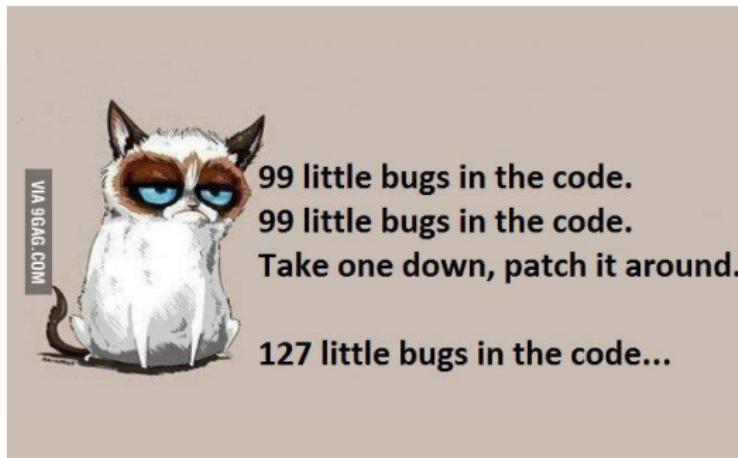


Figura: <http://9gag.com/gag/a5dDw9o/99-little-bugs-in-the-code>

Tips para submitters

- Fai pulls pequenas
- Revisa ti antes
- Automatiza
- Fai tests de regresión

Big Brother vs Ego



Figura: Wikipedia + <http://www.sergerente.net/la-trampa-del-ego>

Acción vs visión

«Vision without action is a dream. Action without vision is simply passing the time. Action with Vision is making a positive difference.» Joel Barker

Lei de Conway

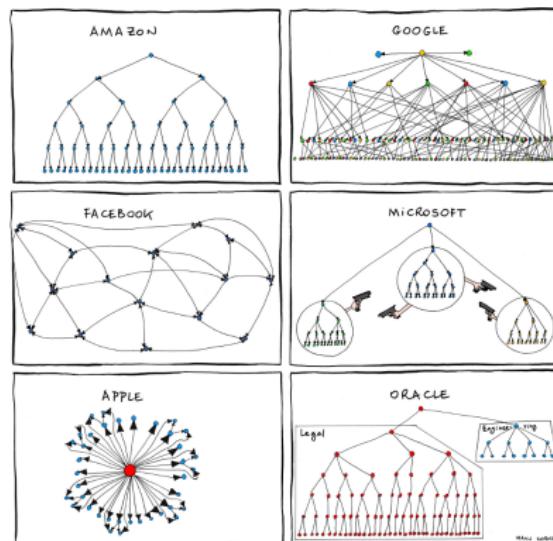


Figura: <http://www.sixteensmallstones.org/conways-law-wios-laws-software/>

Exceso de optimismo



Figura: <https://mahifx.com/blog/aspects-of-behavioural-psychology-in-trading>

Lei de Parkinson da Trivialidade



Figura: http://www.zj86.cn/page/what-is-this_.net/fr/definition/triviality

Tips

- Chámase Code Review, non People Review
- Medide en vez de controlar
- Revisade todos
- Preguntade moito "por que"
- Discutide en persoas
- En lugar de ter razón, usade guías de estilo, métricas, ferramentas, ...
- Como submitter, ofrece contexto
- Como reviewer, pregunta alternativas

Ideas para mellorar o código

- Single Responsibility
- Open/Close
- Duplicación
- Eficiencia
- Manexo de errores
- Regra do boy scout

Algúns malos olores

- Nomes
- Lonxitudes
- Comentarios
- Número de parámetros
- Lexibilidade/estilo

E o ingrediente segredo...



Figura: <http://kungfupanda.wikia.com/>

Git Avanzado



Táboa de contidos

1 Code Reviews

- Introdución
- Que é unha Code Review
- Como facer unha Code Review

2 Git Avanzado

- Entendendo git
- Construíndo a historia
- Utilidades para review
- Boas prácticas

Git é complicado?



chi wai lau
@tabqwerty



Seguir

"git gets easier once you get the basic idea that branches are homeomorphic endofunctors mapping submanifolds of a Hilbert space."

RETWEETS

2.301

ME GUSTA

1.023



23:28 - 9 mar. 2011



...

Figura: <https://twitter.com/tabqwerty/status/45611899953491968>

Por que git é diferente?

- Non é complicado, é potente
- Debedes desaprender Subversion (ou o que toque)
- Debedes decidir como contar a historia

man

man git

man git-log

man git-cherry-pick

Conceptos esenciais

- SHA-1
- Grafo Dirixido Acíclico
- Working Directory
- Staging Area (index)

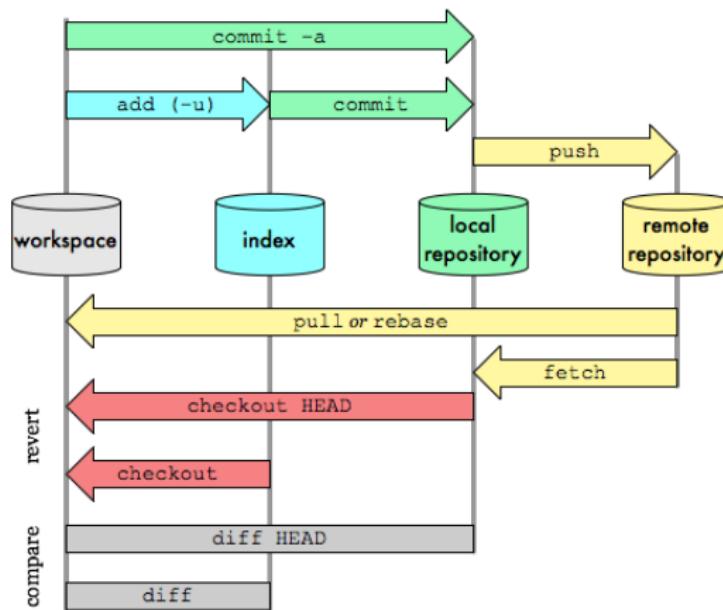
Vocabulario básico

- Commit
- Punteiros: branches, tags, HEAD
- Checkout
- Fast forward
- Merge
- Rebase
- Conflicto
- Reset

Isto ten que quedar claro

Git Data Transport Commands

<http://csteelle.com>



Para referenciar revisiones e paths

```
git show 5f2df0d025026aa8785bde8cb32270aea3bac570
git show 5f2df0d
git show HEAD
git show HEAD~1
git show 5e75cda..ca1fc43
git show :/Merged
git checkout master
git checkout master master
git checkout -- master
man git-rev-parse
```

Para preparar bons commits

```
git add -p  
git add -e  
git rm  
git mv  
git diff  
.gitignore
```

Para rescribir a historia

```
git commit --amend  
git stash  
git revert  
git cherry-pick  
git rebase -i
```

Para traballar en paralelo

```
git checkout -b mybranch master~1
git format-patch 5e75cda..ca1fc43
git apply 0001-Upgrade-version.patch
git remote -v
git remote add/rm
git push/pull
git init --bare
git pull --rebase
git push origin :master
```

Que é e como facer unha pull request



Figura: <http://memegenerator.net/instance/62024625>

Para investigar na historia

```
git log --author=Pepe
git log -Sconsole.log
git blame de168bf9~1 README.md
git bisect
```

Configuración avanzada

```
git config  
.gitconfig  
alias
```

Hooks

```
$ ls .git/hooks/
```

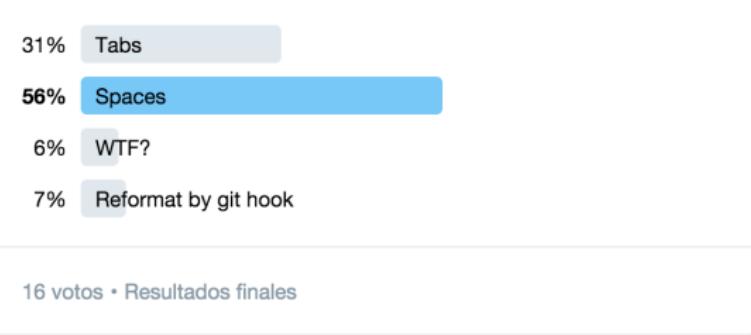
applypatch-msg.sample
commit-msg.sample
post-update.sample
pre-applypatch.sample
pre-commit.sample

pre-push.sample
pre-rebase.sample
prepare-commit-msg.sample
update.sample

Un dos grandes problemas da humanidade resolto!

 José Doval @pepelou · 18 feb.

Tabs or spaces?



Opción	Porcentaje
Tabs	31%
Spaces	56%
WTF?	6%
Reformat by git hook	7%

16 votos • Resultados finales

◀ ↻ 3 ❤️ ⚡ ...

Boas prácticas

- Commits atómicos
- Commits frecuentes
- Non commitear traballo a medias
- Pasa os tests antes de commitear
- Escribe boas mensaxes de commit
- Usa branches, son gratis
- Usade todos o mesmo workflow no equipo

Dúvidas?



Figura: <http://memegenerator.net/instance/62024625>