









-  [Back-End\(https://imasters.com.br/back-end\)](https://imasters.com.br/back-end)
-  [Mobile\(https://imasters.com.br/mobile\)](https://imasters.com.br/mobile)
-  [Front End\(https://imasters.com.br/front-end\)](https://imasters.com.br/front-end)
-  [DevSecOps\(https://imasters.com.br/devsecops\)](https://imasters.com.br/devsecops)
-  [Design & UX\(https://imasters.com.br/design-ux\)](https://imasters.com.br/design-ux)
-  [Data\(https://imasters.com.br/data\)](https://imasters.com.br/data)
-  [APIs e Microserviços\(https://imasters.com.br/apis-microservicos\)](https://imasters.com.br/apis-microservicos)
-  [IoT e Makers\(https://imasters.com.br/iot-makers\)](https://imasters.com.br/iot-makers)

## DESENVOLVIMENTO

DESENVOLVIMENTO (HTTPS://IMASTERS.COM.BR/DESENVOLVIMENTO)

# Como contribuir com um projeto no GitHub

12 OUT, 2015

ROGÉRIO PRADO DE JESUS  
([HTTPS://IMASTERS.COM.BR/PERFIL/ROGERIOP](https://imasters.com.br/perfil/rogeriop))  
Tem 13 artigos publicados com 19515 visualizações desde 2013



### PUBLICIDADE



ROGÉRIO PRADO DE JESUS ([HTTPS://IMASTERS.COM.BR/PERFIL/ROGERIOPRADOJ](https://imasters.com.br/perfil/rogeriopradoj))

13 

é desenvolvedor e consultor de TI. Analista na Caixa Econômica Federal, Consultor na Quatro Elementos Gestão da Informação e Proprietário da Pradoj.com. Tecnólogo em Processamento de Dados pela FATEC-SP, trabalha com PHP desde 2009, entusiasta da comunidade PHPSP. Mais informações: <http://rogeriopradoj.com>

LEIA MAIS ([HTTPS://IMASTERS.COM.BR/PERFIL/ROGERIOPRADOJ](https://imasters.com.br/perfil/rogeriopradoj))

11 OUT, 2016

Como contribuir com o Homebrew para macOS (<https://imasters.com.br/desenvolvimento/como-contribuir-com-o-homebrew-para-macos>)

12 ABR, 2016

Gerenciamento de recursos de hardware no Docker e instalação de pacotes de sistema operacional no docker-machine (<https://imasters.com.br/devsecops/gerenciamento-de-recursos-de-hardware-no-docker-e-instalacao-de-pacotes-de-sistema-operacional-no-docker-machine>)

22 OUT, 2015

Meu primeiro plugin para o Vagrant, o vagrant-ip-show, mostra a lista de IPs da VM (<https://imasters.com.br/devsecops/meu-primeiro-plugin-para-o-vagrant-o-vagrant-ip-show-mostra-a-lista-de-ips-da-vm>)

Eu já estou há alguns meses tentando escrever um guia com dicas sobre como contribuir com open source [neste meu gist](https://gist.github.com/rogeriopradoj/9c2208b50bcb1f047d19) (<https://gist.github.com/rogeriopradoj/9c2208b50bcb1f047d19>), mas até agora não consegui terminar...

No entanto, em 24/09/2015, o Rob Allen, mais conhecido no Twitter como @akrabat (<https://twitter.com/akrabat>), escreveu um artigo sobre o tema: The beginner's guide to contributing to a GitHub project (<http://akrabat.com/the-beginners-guide-to-contributing-to-a-github-project/>). Conversei com ele para pedir a autorização para traduzir seu texto em português do Brasil, e a resposta foi positiva!

Então vamos lá! Este é um guia sobre como contribuir como um projeto open source que utiliza o GitHub. Ele é baseado principalmente no que acompanhei sobre a forma como operam o Zend Framework (<http://framework.zend.com/>), o Slim Framework (<http://www.slimframework.com/>) e o joind.in (<https://joind.in/>). No entanto, este é um guia genérico, sendo assim, leia o README de seus projetos para ver o que for específico.

## Passo 1: Defina uma cópia de trabalho (working copy) no seu computador

Primeiramente, você precisa de um fork local do projeto na sua máquina, vá direto no GitHub e aperte o botão “fork”. Ele criará uma cópia do repositório em sua própria conta do GitHub, e você verá um aviso de que ele foi forkado abaixo do nome do projeto:

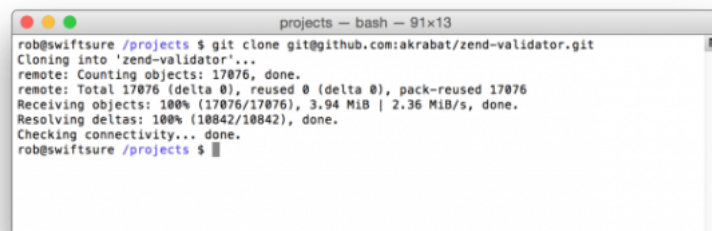


(<https://static.imasters.com.br/wp-content/uploads/2015/10/github-2.png>)

Agora, você precisa de uma cópia local. Procure por “HTTPS clone URL” ou “SSH clone URL” do lado direito do site e use esse endereço para fazer o clone local usando um terminal:

```
1 | $ git clone git@github.com:akrabat/zend-validator.git
```

O resultado será parecido com este aqui:



(<https://static.imasters.com.br/wp-content/uploads/2015/10/github-3.png>)

Entre no diretório do novo projeto:

```
1 | $ cd zend-validator
```

Por fim, você precisa definir um novo remoto (remote) apontando para o projeto original. Dessa forma, você consegue trazer as mudanças e colocá-las dentro de sua cópia local. Acesse o link do repositório original – ele está marcado com “Forked from” no topo da página do GitHub. Isso vai te levar para a página principal do GitHub do projeto, onde você encontra a “HTTPS clone URL” ou a “SSH clone URL” e deve usá-la para criar o novo remoto, que chamaremos de upstream.

```
1 | $ git remote add upstream git@github.com:zendframework/zend-validator.git
```

Agora você tem dois remotos para esse projeto no disco:

o origin, que aponta para seu fork do projeto no GitHub. Você tem acesso de leitura e gravação nesse remoto.

o upstream, que aponta para o repositório principal do projeto no GitHub. Você só tem acesso de leitura nesse remoto.

## Passo 2: Faça suas modificações

Essa é a parte divertida onde você começa a contribuir com o projeto. Em geral, é melhor começar arrumando um problema que está te atrapalhando ou algum bug que você encontrou no issue tracker do projeto. Se estiver procurando um lugar para começar, vários projetos usam a marcação “easy pick” label (<http://seld.be/notes/encouraging-contributions-with-the-easy-pick-label>) (ou alguma variação) para indicar que a issue pode ser resolvida por alguém novo no projeto.

## Branch!

A regra número um é colocar cada pedaço do seu trabalho em seu próprio branch. Se o projeto estiver usando o [git-flow](http://nvie.com/posts/a-successful-git-branching-model/) (<http://nvie.com/posts/a-successful-git-branching-model/>), ele terá tanto um branch master quanto um branch develop. A regra padrão é que se você estiver consertando um bug, você criará um branch a partir do master, e se você estiver adicionando uma nova funcionalidade, criará um branch a partir do develop. Se o projeto tiver apenas o branch master, é de lá que você criará o branch novo. Alguns projetos, como o Slim, usam os nomes dos branches baseados em um número de versão (2.x e 3.x na situação deles). Nesse caso, escolha o branch que for relevante.

Neste exemplo, vamos supor que você está arrumando um bug no zend-validator, então vamos fazer um branch a partir do master:

```
1 | $ git checkout master
2 | $ git pull upstream master && git push origin master
3 | $ git checkout -b hotfix/readme-update
```

Primeiramente, vamos garantir que estamos no branch master. Dessa forma o comando `git pull` irá sincronizar nossa cópia local com o projeto upstream, e o `git push` irá sincronizá-lo com nosso projeto forkado no GitHub. Por fim, criamos nosso novo branch. Você pode nomear o branch como quiser, mas ajuda se o nome for significativo. Incluir o número da issue também é útil. Se o projeto usar o `git-flow` tal qual o zend-validator, existem convenções de nomenclatura para prefixar os branches com “hotfix/” ou “feature/”.

Agora você pode fazer suas alterações.

Tenha certeza de que você apenas arrume o código onde estiver trabalhando. Não ceda à tentação de arrumar outras coisas que for achando durante suas alterações, pois assim seu PR (Pull Request) provavelmente será rejeitado. Certifique-se de que você faça commits em blocos lógicos. Cada uma das mensagens de commit deve ser sensata. Leia o artigo do Tim Pope [A Note About Git Commit Messages](http://tbaggery.com/2008/04/19/a-note-about-git-commit-messages.html) (<http://tbaggery.com/2008/04/19/a-note-about-git-commit-messages.html>) (ou, se preferir em português do Brasil, desde 2011 em RogerioPradoJ.com: [Uma Nota Sobre as Mensagens do Git Commit](http://rogeriopradoj.com/2011/11/29/uma-nota-sobre-as-mensagens-do-git-commit/) (<http://rogeriopradoj.com/2011/11/29/uma-nota-sobre-as-mensagens-do-git-commit/>)).

## Passo 3: Crie o PR (Pull Request)

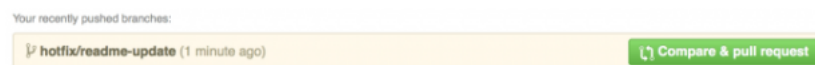
Para criar um PR, você precisa fazer o push de seu branch para o remoto origin e depois apertar alguns botões no GitHub.

Para fazer o push de um branch novo:

```
1 | $ git push -u origin hotfix/readme-update
```

O comando cria um branch em seu projeto no GitHub. A flag `-u` faz a amarração desse branch com seu remoto; assim, no futuro, você pode simplesmente digitar `git push origin`.

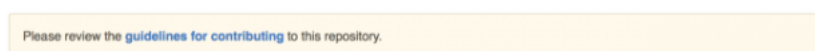
Volte ao navegador e acesse o fork do seu projeto (<https://github.com/akrabat/zend-validator> no meu caso) e você verá que seu novo branch está listado no topo com um conveniente botão “Compare & pull request”:



(<https://static.imasters.com.br/wp-content/uploads/2015/10/github-4.png>)

Vá em frente e aperte o botão!

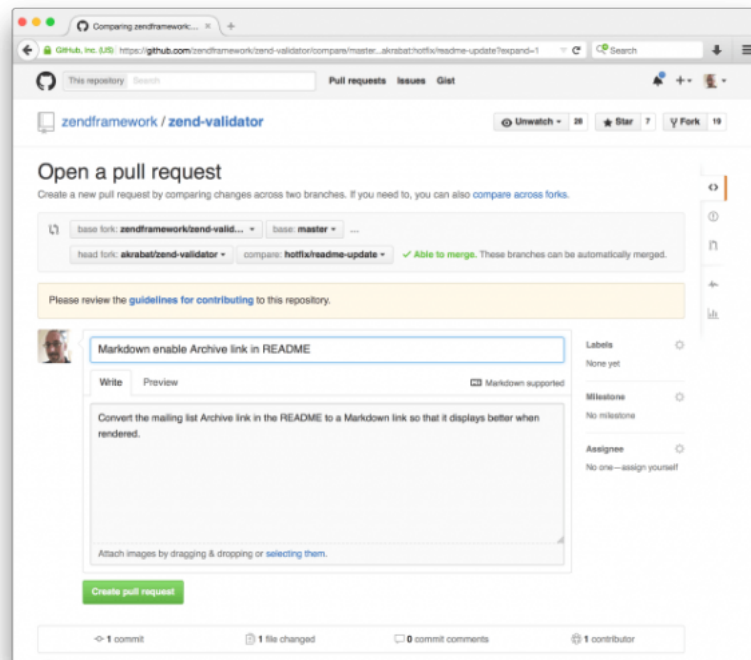
Se você vir uma caixa amarela como esta:



(<https://static.imasters.com.br/wp-content/uploads/2015/10/github-5.png>)

Clique no link que te levará ao arquivo CONTRIBUTING do projeto e o leia! Ele contém informação valiosa sobre como trabalhar com a base de código do projeto e ajudará para que sua contribuição seja aceita.

Na página a seguir, assegure que o “base fork” aponta para o repositório e para o branch correto. Então, certifique-se de fornecer um título bom e sucinto para seu pull request e explique por que você o criou na caixa de descrição. Adicione todos os números de issue caso os tenha.



(<https://static.imasters.com.br/wp-content/uploads/2015/10/github-6.png>)

Se você rolar a tela um pouco, verá um diff das suas alterações. **Verifique mais de uma vez se ele contém o que era esperado.**

Quando estiver satisfeito, aperte o botão “Create pull request” e você terá terminado.

## Passo 4: Revisão dos mantenedores

Para que seu trabalho seja integrado ao projeto, os mantenedores fazem a revisão do que você fez e então solicitam alterações ou fazem o merge.

O artigo da Lorna Mitchell [Code Reviews: Before You Even Run The Code](http://www.lornajane.net/posts/2015/code-reviews-before-you-even-run-the-code) (<http://www.lornajane.net/posts/2015/code-reviews-before-you-even-run-the-code>) trata dos pontos com que os mantenedores se preocupam. Por isso, vá lá dar uma lida e assegure que você facilitou as coisas para os mantenedores o máximo possível.

## Resumindo

Isso é tudo! As partes fundamentais são as seguintes:

Faça o fork do projeto & o clone local.

Crie um remoto upstream e sincronize com sua cópia local antes de criar o branch.

Faça um branch para cada pedaço separado de trabalho.

Faça as alterações, escreva [boas mensagens de commit](https://blogs.gnome.org/danni/2011/10/25/a-guide-to-writing-git-commit-messages/) (<https://blogs.gnome.org/danni/2011/10/25/a-guide-to-writing-git-commit-messages/>) e leia o arquivo CONTRIBUTING quando ele existir.

Faça o push para seu repositório origin.

Crie em novo PR (Pull Request) no GitHub.

Responda a todos os feedbacks recebidos durante a [revisão do código](http://www.lornajane.net/posts/2015/code-reviews-before-you-even-run-the-code) (<http://www.lornajane.net/posts/2015/code-reviews-before-you-even-run-the-code>).

Se você quiser contribuir com um projeto open source, considere o [joind.in](https://github.com/joindin/joindin-web2#joindin) (<https://github.com/joindin/joindin-web2#joindin>)!

—

É isso pessoal! Agradeço muito ao Rob Allen (<http://akrabat.com/>) pelo artigo original (<http://akrabat.com/the-beginners-guide-to-contributing-to-a-github-project/>) e autorização para tradução.

Para quem quiser mais ideias de projetos opensource para contribuir, dê uma olhada no [#30contribs](http://rogeriopradoj.com/2015/06/24/30contribs-cheguei-nos-30-e-quero-de-presente-pull-requests-e-contribicoes-para-projetos-open-source-e-da-comunidade/) (<http://rogeriopradoj.com/2015/06/24/30contribs-cheguei-nos-30-e-quero-de-presente-pull-requests-e-contribicoes-para-projetos-open-source-e-da-comunidade/>) e também no [@yourfirstpr](https://twitter.com/yourfirstpr) (<https://twitter.com/yourfirstpr>), que te ajudam a começar a contribuir.

O mais legal vai ser daqui a pouco você ter tantas contribuições que nem vai lembrar de quando foi sua primeira vez. Para te ajudar a recordar, você pode usar o <http://firstpr.me/> (<http://firstpr.me/>)!



(<https://static.imasters.com.br/wp-content/uploads/2015/10/github-7.png>)

Até mais!





Artigo original em <http://akrabat.com/the-beginners-guide-to-contributing-to-a-github-project/> (<http://akrabat.com/the-beginners-guide-to-contributing-to-a-github-project/>).



De 0 a 10, o quanto você recomendaria este artigo para um amigo?



#### ARTIGOS PUBLICADOS POR ESTE AUTOR

- ROGÉRIO PRADO DE JESUS ([HTTPS://IMASTERS.COM.BR/PERFIL/ROGERIOPRADOJ](https://imasters.com.br/perfil/rogeriopradoj))  
11 OUT, 2016▶
- Como contribuir com o Homebrew para macOS (<https://imasters.com.br/desenvolvimento/como-contribuir-com-o-homebrew-para-macos>)
- ROGÉRIO PRADO DE JESUS ([HTTPS://IMASTERS.COM.BR/PERFIL/ROGERIOPRADOJ](https://imasters.com.br/perfil/rogeriopradoj))  
12 ABR, 2016▶
- Gerenciamento de recursos de hardware no Docker e instalação de pacotes de sistema operacional no docker-machine (<https://imasters.com.br/devsecops/gerenciamento-de-recursos-de-hardware-no-docker-e-instalacao-de-pacotes-de-sistema-operacional-no-docker-machine>)
- ROGÉRIO PRADO DE JESUS ([HTTPS://IMASTERS.COM.BR/PERFIL/ROGERIOPRADOJ](https://imasters.com.br/perfil/rogeriopradoj))  
22 OUT, 2015▶
- Meu primeiro plugin para o Vagrant, o vagrant-ip-show, mostra a lista de IPs da VM (<https://imasters.com.br/devsecops/meu-primeiro-plugin-para-o-vagrant-o-vagrant-ip-show-mostra-a-lista-de-ips-da-vm>)
- ROGÉRIO PRADO DE JESUS ([HTTPS://IMASTERS.COM.BR/PERFIL/ROGERIOPRADOJ](https://imasters.com.br/perfil/rogeriopradoj))  
4 SET, 2015▶
- Como fazer squash de commits no Git (<https://imasters.com.br/desenvolvimento/como-fazer-squash-de-commits-no-git>)
- ROGÉRIO PRADO DE JESUS ([HTTPS://IMASTERS.COM.BR/PERFIL/ROGERIOPRADOJ](https://imasters.com.br/perfil/rogeriopradoj))  
11 DEZ, 2014▶



ROGÉRIO PRADO DE JESUS ([HTTPS://IMASTERS.COM.BR/PERFIL/ROGERIOPRADOJ](https://imasters.com.br/perfil/rogeriopradoj))  
10 NOV, 2014



Testando rapidamente projetos PHP, com Vagrant, com Docker ou com Servidor Web Embutido (<https://imasters.com.br/back-end/testando-rapidamente-projetos-php-com-vagrant-com-docker-ou-com-servidor-web-embutido>)



SAIBA MAIS  
([HTTPS://IMASTERS.COM.BR/PERFIL/ROGERIOPRADOJ](https://imasters.com.br/perfil/rogeriopradoj))

Rogério Prado de Jesus

🔗 13 Artigo(s)

é desenvolvedor e consultor de TI. Analista na Caixa Econômica Federal, Consultor na Quatro Elementos Gestão da Informação e Proprietário da Pradoj.com. Tecnólogo em Processamento de Dados pela FATEC-SP, trabalha com PHP desde 2009, entusiasta da comunidade PHPSP. Mais informações: <http://rogeriopradoj.com>

0 comentários

Classificar por Mais antigos



Adicionar um comentário...

Plugin de comentários do Facebook

Este projeto é oferecido pelas empresas



(<https://www.cartaoelo.com.br>)



([https://www.linkedin.com/jobs/search/?f\\_C=247645&location=Mundialmente&locationId=OTHERS.worldwide](https://www.linkedin.com/jobs/search/?f_C=247645&location=Mundialmente&locationId=OTHERS.worldwide))



(<http://www.movile.com/jobs>)

Este projeto é mantido e patrocinado pelas empresas



(<http://developers.original.com.br>)



(<https://desenvolvedores.cielo.com.br/api-portal/>)



([https://www.dialhost.com.br/?utm\\_campaign=patrocinio\\_iMasters&utm\\_source=imasters](https://www.dialhost.com.br/?utm_campaign=patrocinio_iMasters&utm_source=imasters))



([https://www.fiap.com.br/?utm\\_source=imasters&utm\\_medium=logorodape&utm\\_campaign=imasters2018](https://www.fiap.com.br/?utm_source=imasters&utm_medium=logorodape&utm_campaign=imasters2018))



([http://gama.academy/pt/programas/experienciam\\_gama\\_experience-sp-devs&utm\\_medium=cpc&utm\\_source=iMasters\\_site&utm\\_content=vag](http://gama.academy/pt/programas/experienciam_gama_experience-sp-devs&utm_medium=cpc&utm_source=iMasters_site&utm_content=vag))



([http://www.hostgator.com.br/?utm\\_source=imasters2018&utm\\_medium=logorodape&utm\\_campaign=imasters2018](http://www.hostgator.com.br/?utm_source=imasters2018&utm_medium=logorodape&utm_campaign=imasters2018))



(<https://www.idexo.com.br/>)



(<http://www.impacta.com.br/>)



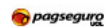
(<https://www.kinghost.com.br/>)



(<http://lambda3.com.br/>)



([https://www.locaweb.com.br/?utm\\_campaign=portalimasters&utm\\_source=portalimasters&utm\\_medium=nev](https://www.locaweb.com.br/?utm_campaign=portalimasters&utm_source=portalimasters&utm_medium=nev))



(<https://pagseguro.uol.com.br/>)



(<https://www.userede.com.br/>)



([https://br.resellerclub.com/webpro?utm\\_source=imasters](https://br.resellerclub.com/webpro?utm_source=imasters))



(<https://uolhost.uol.com.br/>)



(<http://www.zarpsystem.com.br/>)

Este projeto é apoiado pelas empresas



(<https://imasters.tech/>)



(<http://portal.embratel.com.br/cloud/index.php>)



(<https://gerencianet.com.br/lp/imasters/>)



(<https://pagar.me/>)

## ASSINE NOSSA Newsletter

Fique em dia com as novidades do iMasters! Assine nossa newsletter e receba conteúdos especiais curados por nossa equipe



Qual é o seu e-mail?

ASSINAR



SOBRE O IMASTERS ([HTTPS://IMASTERS.COM.BR/P/SOBRE-O-IMASTERS](https://imasters.com.br/p/sobre-o-imasters))

POLÍTICA DE PRIVACIDADE ([HTTPS://IMASTERS.COM.BR/P/POLITICA-DE-PRIVACIDADE](https://imasters.com.br/p/politica-de-privacidade))

FALE CONOSCO ([HTTPS://IMASTERS.COM.BR/FALE-CONOSCO/](https://imasters.com.br/fale-conosco/))

QUERO SER AUTOR ([HTTPS://IMASTERS.COM.BR/P/QUERO-SER-AUTOR](https://imasters.com.br/p/quero-ser-autor))

FÓRUM ([HTTPS://FORUM.IMASTERS.COM.BR/](https://forum.imasters.com.br/))

7MASTERS ([HTTPS://SETEMASTERS.IMASTERS.COM.BR/](https://setemasters.imasters.com.br/))

AGENDA ([HTTPS://IMASTERS.COM.BR/AGENDA/](https://imasters.com.br/agenda/))

IMASTERS.COM ([HTTPS://IMASTERS.COM/](https://imasters.com/))