

Gerenciamento de Configuração do Mozilla Firefox

1. Contribuição

O Firefox possui uma imensidão de Bugs que os voluntários podem escolher para tentar fornecer uma solução em uma plataforma chamada Bugzilla. Após a escolha, o desenvolvedor aplica para receber a aprovação do dono do módulo o qual o Bug pertence, caso seja aprovado o desenvolvedor pode fazer o pull do código e começar a trabalhar em cima do problema. Todas as mudanças realizadas no código são rastreadas por uma tag de bug Report no Bugzilla. Além disso, o projeto Mozilla ainda traz a possibilidade de contribuição através de outros serviços como tradução, testes, design e outras categorias.

Por último, caso o desenvolvedor tenha desenvolvido um patch e queira submetê-lo para o Mozilla, ele deverá solicitar um review do seu código.

2. Code Review

Esse é o mecanismo básico do Mozilla para validar o design e a implementação de novos patches, ajudando a manter a consistência no design e nas práticas de implementação em todos os módulos do Mozilla. O código deve ser commitado utilizando o seguinte formato: *'Número do Bug - O que o seu patch faz; r?reviewer'*. Para que um código possa ser aceito como uma contribuição ele deve ser revisado e aprovado pelo dono do módulo onde esse código será enviado. Logo, este código será identificado com uma tag "r=[name] para cada módulo que ele irá afetar.

Um code review normalmente irá focar no design do patch proposto, implementação e utilidade para resolução de um problema previamente relatado. Os reviewers do Mozilla normalmente irão focar nos seguintes atributos: review de metas, API/design review, review de manutenibilidade, review de segurança, review de integração, review de testes, review de performance e review de licença.

Além disso, o Mozilla fornece um checklist de boas práticas para os desenvolvedores terem maior facilidade ao submeterem seus patches. Essa lista contém práticas como: correção, qualidade, estilo, segurança, privacidade, leaks de privacidade, impacto na performance, problemas de threading, compatibilidade, documentação e etc.

3. Testes

Após o código ser aprovado pelo code reviewer, significa que o código está pronto para passar pelos testes automatizados do Mozilla. Caso aprovado, o código está pronto para ser incorporado a feature a qual ele pertence.

Symbol	Name	File type	Platform		Process		Environment		Privilege		What is tested
			Desktop	Mobile	Parent	Child	Shell	Browser Profile	Low	High	
ES	ESLint	JS	Yes	Yes	N/A		Terminal	–	N/A		Javascript is analysed for correctness.
mocha(EPM)	ESLint-plugin-mozilla	JS	Yes	No	N/A		Terminal	–	N/A		The ESLint plugin rules.
f8	flake8	Python	Yes	Yes	N/A		Terminal	-	N/A		Python analysed for style and correctness.
W	wpt lint	All	Yes	No	N/A		Terminal	–	N/A		web-platform-tests analysed for style and manifest correctness
android-checkstyle	android-checkstyle	Java	No	Yes	N/A		Terminal	-	N/A		Java analysed for coding style consistency.
android-lint	android-lint	Java	No	Yes	N/A		Terminal	-	N/A		Java analysed for common Android coding errors.
android-findbugs	android-findbugs	Java	No	Yes	N/A		Terminal	-	N/A		Java analysed for common Java coding errors.
WR(tidy)	WebRender servo-tidy	All	Yes	No	N/A		Terminal	-	N/A		Code in gfx/wr is run through servo-tidy

Figura 2. Exemplos de testes automatizados do Mozilla

4. Branches

O Firefox possui 4 canais/branches, sendo eles:

- Firefox Release/mozilla-release: possui apenas o release oficial.
- Firefox Beta/mozilla-beta: utilizado para testar a próxima versão do firefox antes que ela se torne o release oficial. Esse repositório também é construído a partir do branch Beta
- Firefox Nightly/mozilla-central: contem os releases que possuem features experimentais. Para que
- Firefox ESR/mozilla-esr68: contem os releases oficiais para Organizações

O Firefox Nightly é abastecido por todas as novas mudanças, features e patches que são enviadas para o repositório mozilla-central. O Nightly solta suas novas releases a cada 12 horas, podendo ocorrer releases adicionais caso problemas ocorram. A cada 4 semanas, ocorre um merge das branches mozilla-central e mozilla-beta, fazendo com que a mozilla-beta branch passe para uma fase onde pode receber apenas patches focados em estabilizar essa nova release. Pouco antes desse merge, ocorre o “congelamento leve” do código do Nightly, impedindo que desenvolvedores soltem novos códigos que sejam arriscados para a estabilidade e qualidade geral do Firefox.

Então, o Firefox Beta é disponibilizado 3 vezes por semana, totalizando 9 betas por ciclo, desde que não ocorram imprevistos. Ao final do ciclo Beta, a build final é validada pelo time de QA, que irá por fim classificar aquela build como pronta para ser lançada no mozilla-release branch.

Main repositories

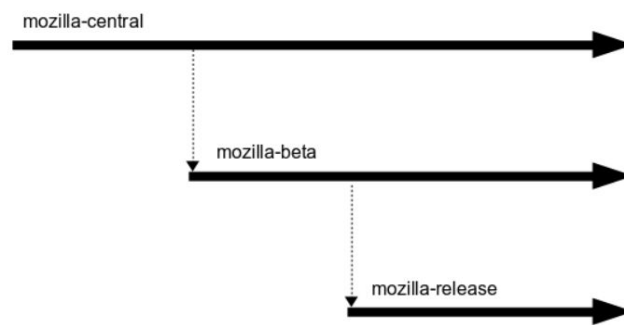


Figura 1. Mozilla Firefox Branch Tree