UNICESUMAR – UNIVERSIDADE CESUMAR
CENTRO DE CIÊNCIAS EXATAS TECNOLÓGICAS E AGRÁRIAS
CURSO DE GRADUAÇÃO EM ENGENHARIA DE SOFTWARE

FÓRUM BIMESTRAL

MARCOS ROBERTO DE ANDRADE DOS SANTOS

MARINGÁ – PR
2020

Linguagem Python.

Para acessar o repositório e acessar os dados:

1. Vá para https://github.com/python/cpython .

2. Pressione o reportório no canto superior direito.

3. Quando perguntado onde bifurcar o repositório, escolha bifurcá-lo para seu nome de usuário.

4. Seu repositório será criado em .https://github.com/<username>/cpython

5. Clone seu repositório do GitHub (substitua <username>pelo seu nome de usuário):

$ git clone git@github.com:<username>/cpython.git

(Você pode usar URLs baseados em SSH ou HTTPS.)

6. Configure um upstream para o contole remoto:

$ cd cpython

$ git remote add upstream git@github.com:python/cpython.git

7. Verifique se sua configuração está correta:

$ git remote -v

origin git@github.com:<your-username>/cpython.git (fetch)

origin git@github.com:<your-username>/cpython.git (push)

upstream git@github.com:python/cpython.git (fetch)

upstream git@github.com:python/cpython.git (push)

Para solicitar um pull

1. Atualize os dados do seu upstream no repositório:

git fetch upstream

2. Crie um novo branch em seu clone local:

git checkout -b <branch-name> upstream/master

3. Faça alterações no código e use e para vê-las.git statusgit diff

4. Certifique-se de que as alterações estejam corretas e não causem nenhuma falha no teste:

make patchcheck

./python -m test

5. Quando estiver satisfeito com as alterações, adicione os arquivos e confirme-os:

git add <filenames>

git commit -m '<message>'

6. Em seguida, envie seu trabalho para o fork do GitHub:

git push origin <branch-name>

7. Finalmente vá em frente : você verá uma caixa com o branch que você acabou de pressionar e um botão verde que permite criar uma solicitação de pull no repositório oficial do CPython.https://github.com/<your-username>/cpython

8. Quando as pessoas começarem a adicionar comentários de revisão, você pode abordá-los mudando para o seu branch, fazendo mais alterações, comprometendo-os e empurrando-os para atualizar automaticamente seu PR:

git checkout <branch-name>

make changes and run tests

git add <filenames>

git commit -m '<message>'

git push origin <branch-name>

- Se um desenvolvedor central que está revisando seu PR enviou um ou mais commits para seu ramo de RP, depois de verificar seu ramo e antes de editar, execute:

git pull origin <branch-name> # pull = fetch + merge

- Se você fez alterações locais que não foram enviadas para o fork e há conflitos de mesclagem, o git irá avisá-lo sobre isso e entrar no modo de resolução de conflito. Consulte Resolvendo conflitos de mesclagem abaixo.

Se o tempo passar e houver conflitos de mesclagem com o branch master, o GitHub mostrará um aviso para esse fim e você pode ser solicitado a resolver isso. Combine as alterações do branch master enquanto resolve os conflitos localmente:

git checkout <branch-name>

git pull upstream master # pull = fetch + merge

resolve conflicts: see "Resolving Merge Conflicts" below

git push origin <branch-name>

Depois que seu PR for aceito e mesclado, você pode excluir a filial :

git branch -D <branch-name> # delete local branch

git push origin -d <branch-name> # delete remote branch

Resolvendo conflitos

Ao mesclar alterações de ramificações diferentes (ou variantes de uma ramificação em repositórios diferentes), as duas ramificações podem conter alterações incompatíveis em um ou mais arquivos. Eles são chamados de “conflitos de mesclagem” e precisam ser resolvidos manualmente da seguinte forma:

1. Verifique quais arquivos têm conflitos de mesclagem:

git status

2. Edite os arquivos afetados e traga-os ao estado final pretendido. Certifique-se de remover os “marcadores de conflito” especiais inseridos pelo git.

Confirme os arquivos afetados:

git add <filenames>

git merge --continue

3. Ao executar o comando final, git pode abrir um editor para escrever uma mensagem de confirmação. Geralmente, não há problema em deixar como está e fechar o editor.

Referencias:

https://devguide.python.org/

https://devguide.python.org/#status-of-python-branches

https://devguide.python.org/pullrequest/#step-by-step-guide