# GitLab ---創建和接受合併請求 Merge Requests

當我們分支上有一更改文件需要合到保護分支master的時候，我們需要用到

Merge Requests (簡稱MR)

## 【提交 Merge Request 申請進行code review】

為了強化代碼記錄的可讀性並協助 Code review 的執行，通過參考已有流程方案，設定一種適合的 Git 流程方案。

### 流程步驟

1. 新建分支
2. 提交 commit 記錄
3. 合併 commit 記錄
4. 推送到對應的遠程倉庫
5. 提交 Merge Request 申請

### 第一步：新建分支

每次開發新功能，都應該從master分支新建一個功能分支。

|  |
| --- |
| $ git checkout -b case-KJDS-00000 |

### 第二步：提交 commit 記錄

添加功能後進行 commit，commit 時需要攜帶足夠的修改信息。

|  |
| --- |
| $ git add .  $ git commit -m "header: some message" |

### 第三步：合併多個 commit 記錄

|  |
| --- |
| 假設在 feature 功能分支上新增了 4 條開發記錄：  commit 34d364d9d51dc94b264e99f7a92add50dd2c3987  Author: Aeo <A\_doz@126.com>  Date: Sun Jun 25 12:27:02 2016 +0800  misc: forth commit  commit 27322cb4b3f99226ffa98240460b90d92ed55a17  Author: Aeo <A\_doz@126.com>  Date: Sun Jun 25 12:26:42 2016 +0800  misc: third commit  commit 405b957a96a7dbe352cf7da9a422312a735f6081  Author: Aeo <A\_doz@126.com>  Date: Sun Jun 25 12:26:16 2016 +0800  misc: second commit  commit cc12fc86a7738ee2f9a8a48c31a9435232c2b08f  Author: Aeo <A\_doz@126.com>  Date: Sun Jun 25 12:25:53 2016 +0800  misc: first commit |

現在我們已經完成了 feature 分支的開發和聯調，需要把代碼合併到當前主開發分支。

此時可以合併一些不必要的 commit 使主開發分支保持干淨，優先推薦 git rebase 進行合併操作。

如果你想要合併最後 4 個 commit，你可以運行下列命令：

|  |
| --- |
| $ git rebase -i HEAD~4  這時 git 會打開一個互動界面，方便用戶對歷史提交進行修改：  pick cc12fc8 misc: first commit  pick 405b957 misc: second commit  pick 27322cb misc: third commit  pick 34d364d misc: forth commit  # Rebase 2763481..34d364d onto 2763481 (4 command(s))  #  # Commands:  # p, pick = 使用該提交  # r, reword = 使用該提交，但需要編輯提交信息  # e, edit = 使用該提交，但此處暫停並提供修改機會  # s, squash = 使用該提交，但合併到上一個提交記錄中  # f, fixup = 類似 squash，但丟棄當前提交記錄的提交信息  # x, exec = 執行 shell 命令  # d, drop = 移除當前提交 |

我們需要的是利用 squash 或者 fixup 進行分支合併：

|  |
| --- |
| pick cc12fc8 misc: first commit squash 405b957 misc: second commit squash 27322cb misc: third commit squash 34d364d misc: forth commit |

若使用 fixup 的話，則直接丟棄其它記錄，省去下一步操作。但使用 squash 時，則需要在下一步中對這 4 條 commit 信息進行修改和保存：

|  |
| --- |
| # This is a combinatin of 4 commits.  # This is the 1st commit message:  misc: first commit  # This is the commit message #2:  misc: second commit  # This is the commit message #3:  misc: third commit  # This is the commit message #4:  misc: forth commit |

編輯並保存後，通過 git log 可以看到僅剩一條 commit 記錄：

|  |
| --- |
| # 此時的 commit SHA 與之前的並不相同  commit da473276aa981f6e29577aa09a525109971547f2  Author: Aeo <A\_doz@126.com>  Date: Sun Jun 25 12:50:53 2016 +0800  misc: first commit |

*rebase 的風險： 使用 rebase 需要遵循一條准則：*

*一旦分支中的提交對象發布到公共倉庫，就千萬不要對該分支進行 rebase 操作。*

*簡單來說，當合併 commit 記錄中混雜著他人的 commit 記錄，此時就不可以再對這部分 commit 記錄做合併操作。*

但只要新開分支且保持分支獨立開發，混雜的情況就不存在。

### 第四步：推送到對應的遠程倉庫

完成 commit 記錄的合併後，此時就可以合併到目標分支的最新代碼，避免後續提交 Merge Request 時存在沖突。

|  |
| --- |
| $ git push origin case-KJDS-00000 |

### 第五步：提交 Merge Request 申請

當完成功能後提交到遠程後，需要經過下述流程 (圖文見下節)：

1. 提交 Merge Request：

通過 ”＋Create Merge Request” 按鈕創建一個 Merge Request；

1. 必須指定 ”Assignee”：

指定需要 review 你代碼的人員，禁止指定自己；

1. 更改 “Target branch”：

改變為需要合併進去的目標分支；

1. 設置合併後刪除被合併分支的選項：

勾選 ”Remove source branch when merge request is accepted.” 選項，在合併完成後刪除源分支，控制分支總數量；

1. 使用 ”Submit Merge Request” 提交合併請求。

完成上述設置後，相關同學將會收到郵件通知，此時可進入 GitLab 進行 code review。

如果發現問題則對問題代碼進行點評並拒絕關閉申請，反之則通過合併申請。

合併申請通過後留下的 Merge Request 記錄，也就記錄了 code reviewer 。

### 註：

拉取遠程代碼時，使用 git pull --rebase 就可以保持提交歷史記錄的整潔。

## 【GitLab創建MR】

### 登錄gitlab服務器，並擁有主幹分支repoter以上的權限，點擊Create Merge Request.

一張含有 文字 的圖片

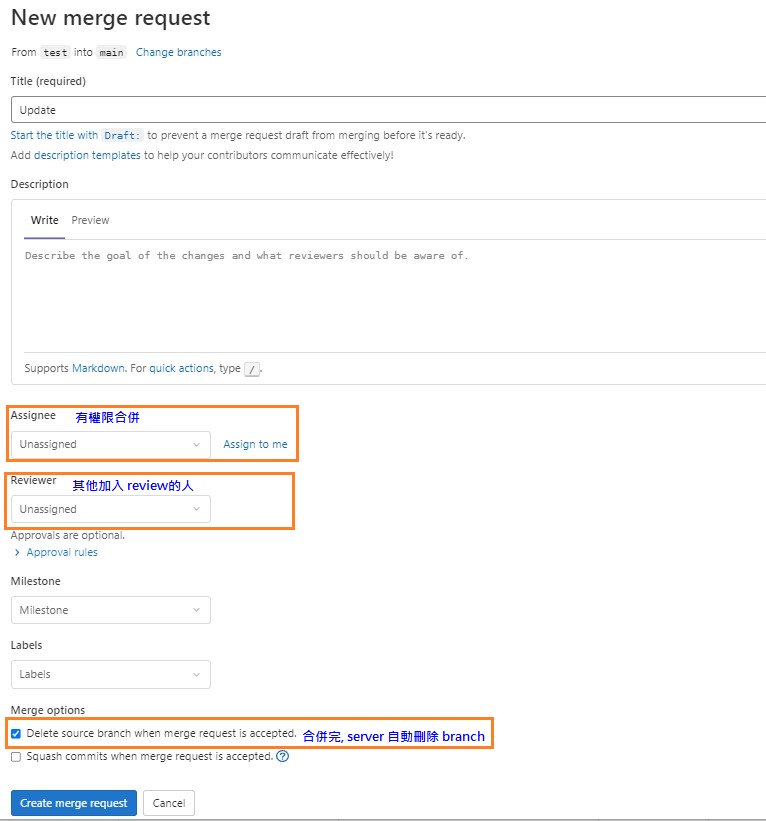
自動產生的描述

### 選擇dev分支的，合入到master分支

一張含有 文字 的圖片

自動產生的描述

### 3.填寫MR的信息，指定審核人（Maintainer以上的權限）



### 4.需要審核人去審核，權限不夠merge是灰色的

一張含有 文字 的圖片

自動產生的描述

## 【GitLab審核MR】

### 以審核人賬號登錄，會發現右上角有一個MR提示通知

一張含有 文字 的圖片

自動產生的描述

### 2.點進這個MR，發現Merge可以選擇了

一張含有 文字 的圖片

自動產生的描述

### 3.合併成功

一張含有 文字 的圖片

自動產生的描述