如何修改Sourcetree已提交的注释信息

- <u>1 需求说明</u>
- <u>2 实现思路</u>
- <u>3 A1.修改本地最后一次提交的注释(Sourcetree)</u>
- <u>4 A2.修改本地最后一次提交的注释(命令行)</u>
- <u>5 B.修改以前提交的注释</u>
- 6 如何把历史的注释记录修改推送到远端
 - <u>6.1 需求说明</u>
 - <u>6.2 解决方式</u>



• 有时候我们在进行版本控制提交日志时,不小心写错了提交的说明日志,希望进行修改

2 实现思路

0

- 在本地通过 Git 的命令行的方式修改
- 再根据需要,强制推送到远端的仓库

3 A1.修改本地最后一次提交的注释 (Sourcetree)

• 可见下图, 最新的一条版本控制提交的内容的说明信息为错误的信息, 希望改对



• 步骤 1:选择 File Status, 打勾右下角的 Amend latest commit 选项,在输入框会显示最近一次提交的信息,可直接修改

| 0 | | Pending files, sorted by file status ∨ ≡ ∨ Search | Q |
|---|-------------------|--|--------|
| | File Status | 1. 选择 | 2 |
| | History Search | $\nabla_{0}\rho$ q | 2 |
| | ✓ ູ່ງ BRANCHES | Nothing to commit | \sim |
| | O main | Den in Explorer | a ` |
| | > 🚫 TAGS | | |
| | | | 2 |
| | ✓ origin | | |
| | HEAD | Zhiyi Yuan <zhiyi.yuan@br-automation.com></zhiyi.yuan@br-automation.com> | 15 V |
| | main | | |
| | master | 5. 社区主制正门音 | |
| | nextBranch | 2.打勾此项 | |
| | | Amend latest commit Commit | nit |

• 步骤 2: 修改完成后, 点击提交

| 0 | Zhiyi Yuan <zhiyi.yuan@br-automation.com></zhiyi.yuan@br-automation.com> |
|------------------------|--|
| | 正确的内容 1. 修改完成 2. 点击提交 |
| | Push changes immediately to origin/main |
| ● 在 Hi | story 中可见最后一次提交的注释已被修改 |
| 0 | WORKSPACE All Branches Show Remote Branches Date Order V |
| | Graph Description File Status O Description |
| | History ・ Vorigin/main Vorigin/HEAD ??新的文件增加了 |
| 4 A2 | .修改本地最后一次提交的注释(命令行) |
| • 可见 | 下图,最新的一条版本控制提交的内容的说明信息为日语,我们希望改为中文 |
| 0 | All Branches 🗸 🗹 Show Remote Branches 🛛 Date Order 🗸 |
| | Graph |
| | o pmain 1t 新しいファイルが追加されました fl origin/main 1t のrigin/HEAD 増加一行測试 |
| | 测试第一个文件 |
| 步骤 | 1: 点击 Sourcetree 右上角的 Terminal 按钮, 输入 git commitamend |
| 0 | |
| | |
| | Git-flow Remote Terminal Explorer Settings |
| | |
| | MINGW64:/c/Users/yuanz/Desktop/GitComme — |
| | yuanz@CNSHAN747 MINGW64 ~/Desktop/GitComment (main) \$ git commitamend |
| | |

• 步骤 2: 出现可修改的页面,此时的页面的数据修改需要使用 VI 编辑器命令进行操作



• 步骤 2-A:

0

0

0

0

| 🚸 MINGW64:/c/Users/yuanz/Desktop/GitComme 🗕 | \times |
|--|---------------|
| 新しいファイルが追加されました | ~ |
| <pre># Please enter the commit message for your changes. L starting # with '#' will be ignored, and an empty message abor e commit. #</pre> | ines ts th |
| # Date: Wed Nov 8 17:58:29 2023 +0800 # | |
| <pre># On branch main # Your branch is ahead of 'origin/main' by 1 commit. # (use "git push" to publish your local commits) #</pre> | |
| <pre># Changes to be committed: # new file: "\346\226\260\345\242\236\344\270 344\270\252\346\226\207\344\273\266.md" #</pre> | \200\ |
| * ? ? ? ? | |
| ~ .git/COMMIT_EDITMSG [unix] (18:12 08/11/2023) 1, <comment .git="" 14l,="" 473b<="" [unix]="" commit_editmsg"="" td=""><td>1 All</td></comment> | 1 All |
| 确认输入法为英文输入法,按 📋 ,即可见屏幕最下方出现 INSERT | 信息 |
| ~ | |
| .git/COMMIT_EDITMSG [unix] (18:12 08/11/2023) 1 INSERT | ,1 All |

• 步骤 2-B: 光标移动在需要修改的文字, 进行信息修改

| 🚸 MINGW64:/c/Users/yuanz/Desktop/GitComme — 🛛 🛛 🛛 | |
|--|--|
| 新的文件增加了 | |
| <pre># Please enter the commit message for your changes. Lines starting # with '#' will be ignored, and an empty message aborts th e commit. #</pre> | |
| <pre><it (18:12="" 08="" 11="" 2023)1,22-15="" <="" [unix]="" commit_editmsg[+]="" pre="" top=""></it></pre> | |

• 步骤 2-C: 当修改完成后,确认当前输入法在英文,按 ESC 按钮,即退出编辑模式,如下图所示



• 步骤 2-D:确保输入法为英文输入模式,输入:wq,按回车按钮

| MINGW64:/c/Users/yuanz/Desktop/GitComme | — | | \times |
|---|--------------------|--------------------|--------------|
| 新的文件增加了 | | | ~ |
| <pre># Please enter the commit message for your starting # with '#' will be ignored, and an empty r e commit. #</pre> | r chang message | ges. Li e abort | ines s th |
| <it (18:12="" 08="" 11="" <="" [unix]="" commit_editmsg[+]="" td=""><td>/2023):</td><td>1,19-13</td><td>В Тор</td></it> | /2023): | 1,19-13 | В Тор |
| :wq | | | - N |

- 步骤 3: 此时已经修改完成, 点击 Fecth 按钮刷新一下日志提交列表
 - ↓ ↑ ↓ Pull Push Fetch
- 可见日志信息的第一行信息已经被修改成功

0

0

0

| 0 | All Branches | ✓ ✓ Show Remote Branches | Date Order 🖂 |
|---|--------------|------------------------------|--------------|
| | Graph | | |
| | Ŷ | O 🔰 main 1t 新的文件增加] | 7 |
| | ŧ | 😰 origin/main 🛛 😰 origin/HEA | AD 增加一行测试 |

5 B.修改以前提交的注释

• 若想修改之前提交的 Commit 记录内容,例如下图中的第三行提交信息



将 Show Remote Branches 取消打钩,可方便查看在本地的分支操作

- 步骤 1: 点击 Sourcetree 软件的右上角 Terminal 按钮, 输入 git rebase -i HEAD~3
 - 其中数字指的是倒数第 n 次提交记录的注释, 可配合上图理解含义。



- 步骤 2: 出现指定修改条目的窗口, 编辑选择需要修改注释的记录。
 - 注意,需要修改的注释的条目,需要确保在第一行,如果不是,请参考步骤 ERROR,退出

| MINGW64:/c/Users/yuanz/Desktop/GitComme | — | | × | |
|---|--------|---------|-----|---|
| pick 1ee524 7 ?????测试第一个文件 pick d0e7992 增加一行测试 pick 6c0df24 ??新的文件增加了 | | | | 1 |
| # Rebase f719a476c0df24 onto f719a47 (3 # # Commands: | comma | nds) | | |
| # p, pick <commit> = use commit @@@</commit> | | | | |
| <se-merge (17:20="" 09<="" [unix]="" git-rebase-todo="" td=""><td>9/11/2</td><td>023)1,1</td><td>Тор</td><td></td></se-merge> | 9/11/2 | 023)1,1 | Тор | |
| | | | - N | 1 |

• 步骤 2-A: 确认当前输入法为英文输入法, 按 i,进入编辑模式, 将第一行的 pick 修改为 edit



0

0



• 步骤 2-B: 切换至英文输入法,按 ESC 退出编辑,输入:wq 保存

| 🚸 MINGW64:/c/Users/yuanz/Desktop/GitComme — 🛛 🛛 🗙 | |
|--|--------|
| edit 1ee5247 ?????测试第一个文件 pick d0e7992 增加一行测试 pick 6c0df24 ??新的文件增加了 | ^ |
| # Rebase f719a476c0df24 onto f719a47 (3 commands) # # Commands: | |
| # p, pick <commit> = use commit @@@</commit> | |
| <merge (17:20="" 09="" 11="" 2023)1,4="" [unix]="" git-rebase-todo[+]="" td="" top<=""><td></td></merge> | |
| :wq 🔶 | \sim |

步骤 3: 输入 git commit -- amend ,进入注释修改流程,详情参考 A.修改本地最后一次提交的注释 章节的步骤2

| 🚸 MINGW64:/c/Users/yuanz/Desktop/GitComme — 🛛 🛛 🛛 | < |
|---|---|
| You can amend the commit now, with | ^ |
| git commitamend | |
| Once you are satisfied with your changes, run | |
| git rebasecontinue | |
| yuanz@CNSHAN747 MINGW64 ~/Desktop/GitComment (main REBASE | |
| \$ git commitamend | ~ |

。 修改如下图所示

0

Г



• 步骤 4: 输入 git rebase -- continue 退出



• 步骤 5: 此时已经修改完成, 点击 Fecth 按钮刷新一下日志提交列表

| € | ĵ⁰ | ٢ | |
|------|------|-------|--|
| Pull | Push | Fetch | |

。 此时可见提交记录的注释, 以前的信息已经被修改了。

| Current Branch $$ | | Show Remote Branches | | Date Order $ \sim $ | | | | | |
|---------------------------------------|--|----------------------|--|---------------------|---|----------------------|--|--|--|
| Graph | | | | | | | | | |
| ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● | | | | | | | | | |
| | | | | | • | First try Git commit | | | |
| | | | | | | | | | |

- 步骤 ERROR: 若在步骤 2发现输入错数字,则按照以下流程退出
 - 步骤 ERROR-A:确保自己不在编辑模式,使用英文输入法,输入:q
 - 。 步骤 ERROR-B: 输入 git rebase --abort 退出

6 如何把历史的注释记录修改推送到远端

6.1 需求说明

0

• 当修改了历史的记录注释信息

| Current Bra | nch 👻 🗌 Show Remote Branches 🛛 Date Order 👻 |
|-------------|---|
| Graph | |
| | O 🔰 main 4t 44 ??新的文件增加了 |
| • | 增加一行测试 修 207 |
| • | ?????测试第一个文件 |
| • | 第一次提交 First try Git commit |
| • | Add 1 |
| | · · · · · · · · · · · · · · · · · · · |

• 打开 Show Remote Branches ,会发现有远程云端的日志与本地的日志有较大差异。

| Current Brand | ch ∨ 🗹 Show Remote Branches 🛛 Date Order ∨ |
|---------------|---|
| Graph | |
| | ▶ main 4t 41 ??新的文件增加了 增加一行测试 ?????测试第一个文件 第一次提交 First try Git commit ▶ origin/HEAD ??新的文件增加了 增加一行测试 ?????测试第一个文件 First try Git commit Add 1 |

6.2 解决方式

.

- 在Terminal下, 输入 git push --force
- MINGW64:/c/Users/yuanz/Desktop/GitComment
 yuanz@CNSHAN747 MINGW64 ~/Desktop/GitComment (main)
 \$ git push --force
 再次 Fetch 后发现日志信息完全一致了
 Current Branch ~ ✓ Show Remote Branches Date Order ~
 Graph Description
- 2 main 2 origin/main 2 origin/HEAD ??新的文件增加了 增加一行测试 ?????测试第一个文件 第一次提交 First try Git commit Add 1
- ● 需要确保你强制 push 之前没有人提交代码,如果在你 push 之前有人提交了新的代码到远程仓库,然后你又强制 push,那么他们在云端的提交会被你的强制更新覆盖!