

## 内容目录

|   |   |
|---|---|
| 1.根据 comments, 向 gerrit push 了修改。 .....     | 1 |
| 2.使用.gitignore 设置忽略文件时做的总结: .....           | 1 |
| 3.gerrit 代码 review 的过程:.....                | 1 |
| 4.关于 git 的新的理解: 使用 git 是通过散列算法来管理文件的; ..... | 4 |

今日总结:

## 1.根据 comments, 向 gerrit push 了修改。

## 2.使用.gitignore 设置忽略文件时做的总结:

举例:

忽略 class 文件: \*.calss

忽略 target 文件夹: target/

忽略 target 文件夹下的所有文件: target/\*

忽略 target 文件夹下的文件但不包括 one.txt: !target/one.txt

注意点:

target/

!target/one.txt

这种写法是有问题的。首先 target/忽略了整个 target 文件夹, 后面无论怎么排除 target 文件夹下的某些文件, 都是无效的。

target/\*

!target/bin/one.txt

这种写法也是有问题的, 也会导致 one.txt 仍然被忽略。只对 target 文件夹下的文件有效, 对 target 文件夹下的其他文件夹无效。

如果需要添加多个仍会被 track 的文件, 则需要一行一行写出, 不可以在一行追加:

target/\*

!target/one.txt twqo.txt three.txt

以上情况都会导致!之后的文件仍被排除在外, 也就是 git 将不会 track。

## 3.gerrit 代码 review 的过程:

changes 有三种形式: open, Merged, Abandoned。


open 的 change 是被 push 到缓存区但是没有被 review 的;

merged 的 change 是被 review 之后并且 submit 的;

abandoned 的 change 是被禁止掉的 change, 也可以通过 RESTORE 恢复至 open 状态。

如何进行 review：


可以点击 ADD REVIEWER 添加 reviewer。

|                    |  |
|--------------------|--|
| Updated            | Jul 24   |
| Owner              |  zhaofs0921 |
| Assignee           | Add assignee...  |
| Reviewers          | <a href="#">ADD REVIEWER</a>   |
| CC                 | <a href="#">ADD CC</a>   |
| Project            | <a href="#">automotive/manifest</a>  |
| Branch             | <a href="#">master</a>   |
| Topic              | <a href="#">ADD TOPIC</a>  |
| Hashtag            | <a href="#">ADD HASHTAG</a>  |
| CPPCheck           |  |
| Code-Review        |  |
| CommitMessageCheck |  |
| ...                |  |

点击 REPLY 按钮进行 REVIEW 打分：

1 (Abandoned): test

latedJul 24

ner zhaofs0921

igneeAdd assignee...

iewers[ADD REVIEWER](#)  
[ADD CC](#)

ject[automotive/manifest](#)


nch[master](#)

REPLY

test

Change-Id: Iac8e5235fcdf2d11ad05387f7

EDIT

Owner majc0806

ReviewersAdd reviewer...

CC

qinsl0106 <qinsl0106@thundersoft.com>

Say something nice...

☐ Preview formatting

CPPCheck

-10+1

Verified

Code-Review

-2-10+1+2

Looks good to me, approved

CommitMessageCheck

-10+1

Verified

JavaCheck

-10+1

Verified

Verified

-10+1

Verified

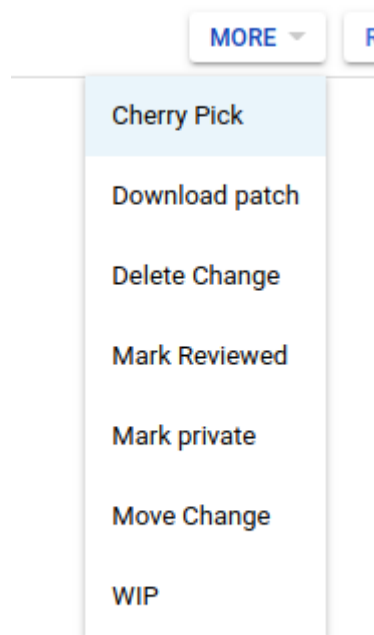
CANCEL

SEND

当全部通过之后，就可以进行 submit：



更多操作部分：



Cherry Pick 可以选择把当前分支的

更改 cherry pick 到其他分支；

Download patch 可以下载 diff 文件至本地；

Delete change 用于将这个 change 直接删除（和 Abandon 不同，Abandon 可以恢复，delete 不可以恢复）；

WIP(Working In Progress)：当一个 change 被标记为 WIP 之后，在页面会出现 WIP 字样：

| Subject |           | Status | Owner      |
|---------|-----------|--------|------------|
| ★       | [Feature] | WIP    | 👤 majc0806 |
| ★       | [Feature] |        | 👤 majc0806 |

其他几个功能暂时不知道作用。

## 4.关于 git 的新的理解：

### 使用 git 是通过散列算法来管理文件的；

使用 git add 命令添加一个文件到暂存区，那么对应的.git/objects/文件夹下会生成对应的文件夹，类似于下图，是以散列码的前两个字符作为文件夹的名字，而散列码是通过对文件的内容进行散列算法得来的，进入某个文件夹（如 78），发现剩下的散列码组成了文件名：

```
总用量 44
drwxrwxr-x 11 majc0806 majc0806 4096 8月 16 19:40 ./
drwxrwxr-x  8 majc0806 majc0806 4096 8月 16 19:40 ../
drwxrwxr-x  2 majc0806 majc0806 4096 8月 16 16:15 1f/
drwxrwxr-x  2 majc0806 majc0806 4096 8月 16 16:45 4a/
drwxrwxr-x  2 majc0806 majc0806 4096 8月 16 19:40 61/
drwxrwxr-x  2 majc0806 majc0806 4096 8月 16 19:38 78/
drwxrwxr-x  2 majc0806 majc0806 4096 8月 16 16:45 86/
drwxrwxr-x  2 majc0806 majc0806 4096 8月 16 16:16 b7/
drwxrwxr-x  2 majc0806 majc0806 4096 8月 16 16:15 e6/
drwxrwxr-x  2 majc0806 majc0806 4096 8月 16 16:06 info/
drwxrwxr-x  2 majc0806 majc0806 4096 8月 16 16:06 pack/

总用量 12
drwxrwxr-x  2 majc0806 majc0806 4096 8月 16 19:38 ./
drwxrwxr-x 11 majc0806 majc0806 4096 8月 16 19:40 ../
-r--r--r--  1 majc0806 majc0806  17 8月 16 19:38 981922613b2afb6025042ff6bd878ac1994e85
```

明日计划：

熟悉 repo；

从 android source 网站熟悉 aosp 的开发 workflow；