

GitLab使用说明

项目统一使用Git作为版本管理工具，我们搭建了一个gitlab系统，地址为<http://114.55.35.12/>，请及时注册账号。

注册规范说明

1，注册请使用学院邮箱，非学院邮箱无效。在注册完成后，会统一确认，我们会给出一个截止日期，截止日期前注册完，之后会统一确认账号。

2，name和username使用院邮前缀，即 `**14` 的格式，便于后面识别。

3，在git上编辑自己的个人信息页面 <http://114.55.35.12/profile> 在其他栏填完的基础上，在最后Bio条目输入框要求输入以下信息，所有的信息按照示例格式填写：

需输入内容：

姓名,性别,学号,以前是否有过编程经验,多久编程经验(年),使用的编程语言,以什么语言为主,之后想进一步学习的编程语言,开发喜好(是喜欢单枪匹马还是喜欢团队合作),是否有写单元测试的习惯,觉得测试是否重要, 软工一,二的分数(如不想写可以-1替代),是不是本项目的 manager。

示例如下：

XiaoZhang#Male#MG1532001#NO#3.5#java,python,shell#python#python,perl#team|individure#No#YES#83#78#NO

网页如下示例格式

Bio

XiaoZhang#Male#MG1532001#NO#3.5#java,python,shell#python#python,perl
#team|individure#No#YES#83#78#NO

Tell us about yourself in fewer than 250 characters.

Save changes

使用中的一些规范

设置SSH Key

注册完成后，只是个人页面，注意这里类似github，需要在个人页面中设置 `ssh key` 来保证后面能正常提交，具体设置就不多说了，和github类似，网上有很多教程文章。例如可参考 <http://www.tuicool.com/articles/BVJjiez>

分组

之后完成课程组队分组后，请每个小组创建自己的 `group`，命名规范为

```
<小组负责人学号>_cseiii_<项目选题>
```

项目选题为 `GitMining` 或 `AnyQuant` (注意大小写), 例如

```
00001_cseiii_GitMining
```

同时在小组介绍中加上组内所有成员的姓名和学号，便于识别。建好小组后将组内成员拉进小组内，注意同一个人不能在多个小组中(自己定义的小组除外)。

后面项目提交就以小组为标示，同时也会统计组内各成员的代码和贡献程度。除此外的私有项目不计算在内。

项目说明

- 项目设为注册用户可见, **Internal级别**，开放代码。
- 在项目中，需要在根目录下建立一个 `blog` 文件夹，这个作为每周团队项目进度说明，每周需要将进度提交到这个目录下的新文件中，以周为时间单位。说明主要包括这周项目做了什么，下周项目准备做什么以及在项目中遇到了哪些问题。

提交说明

1, 提交代码时规范的 commit log 信息

commit log 格式如下:

```
提交类型:对本次提交的概括性概述
```

这里的提交类型分4类:

- * 修复bug `Fix bug`;
- * 开发工作 `Development`;
- * 文档编辑工作 `Document`;
- * 其他 `Other`;

示例:

```
Fix bug: there are some logical errors when calculate the cosine distance, have fixed these er
```

2, 每周更新自己的项目进展

在本项目的主页面下新建一个文件夹project progress，该文件夹存放每周的项目进展，要求每周的进展以文本文件的形式上传，文本文件的命名格式为

第几周_日期_项目进展.txt（只能以英文形式命名如1_2016-02-28_progress.txt）。在该文件中需记录每个人本周

示例如下：

```
who: 张三
when: 第5周
what: 初步完成注册登录界面的开发
problem: 不能很好的控制如何让用户在注册时输入有效数据
todo: 预计下周加入约束以很好地验证用户输入数据是否有效
```

3，记录团队交流情况

在本项目的主页面下新建一个文件夹communication，该文件夹存放每周的团队交流情况，将交流内容以文本文件的形式上传，文本文件的命名格式为

第几周_日期_团队交流.txt（只能以英文形式命名如1_2016-02-28_communication.txt）

交流的内容需包括以下几个方面:什么时候,谁和谁,就什么问题进行了多长时间的交流以及达成了什么决定。

示例如下：

```
who: 张三, 李四, 王二（以，分隔人名）
when: 第1周
what: 讨论项目如何进行分工
how long: 1.5h（以小时为单位）
arrived solutions: 张三做前台，李四做后台，王二专门负责文档编辑
```

下面是一个实际截图



其他说明

以下说明仅作参考使用，不是强制使用：

使用中推荐做法是master主干，分支做实验性的开发，每次开发完成后，比如每次迭代，将此次迭代完成的内容merge到master上，也就是说保证master分支上的项目一直是可部署和可用的。最终结束后我们会统一检查master上的项目。