

软工平台介绍

计 43 李明杰
计 44 叶德铭

- 网址 `http://166.111.227.241:8000/`

软工平台 首页 通知 GitLab Jenkins 项目分析 表的项 选择项目 关于 汉语

登录

清华大学

学号 2014011396 请使用清华学号登录

密码 *****

登录

- 目前仅支持清华账号验证登陆，密码采用 bcrypt 加密存入数据库
- 右上角支持语言切换

- 首页显示通知及一些统计信息，首页中统计的 commit 过滤掉 commit-message 小于 10 字节的 commit



选择项目

- 右上角 Exit 按钮离开当前队伍
- 填写队名，志愿等信息后可创建队伍
- 队员项填入选入队员学号，仅当该名同学不属于任何队伍时才会被拉入队伍

选择项目

Exit

所属队伍

队名	队长	队员1	队员2	队员3	队员4
大明队	叶德铭/2014011398				

创建队伍:

队长: 2014011398

队名: 大明队

队员1

队员2

队员3

队员4

一志愿1 1

一志愿1 2

一志愿2 3

二志愿1 4

二志愿2 5

二志愿3 6

创建队伍

志愿统计(上次更新:,每四小时更新)

编号	项目名称	助教	招募组数	一志愿	二志愿
1	小明的project	小明	4	0	0

所有队伍:

队名	队长	队员1	队员2	队员3	队员4
大明队	叶德铭/2014011398				

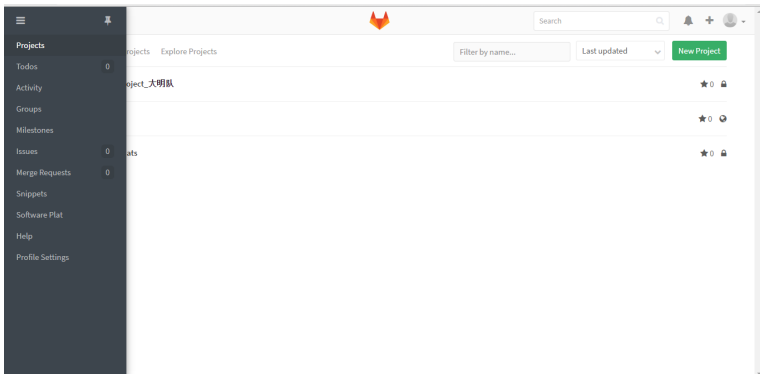
- 志愿信息每 4h 统计一次
- 队伍信息实时显示在下方表格

- 每支队伍分别拥有 3 个一志愿和 3 个二志愿，必须填写满这 6 个志愿不得留空（尽量减少有队伍分不到项目的情况）
- 志愿抽签方法：
- 随机打乱队伍顺序后，对每支队伍使用一志愿与项目之间进行二分图匹配，优先匹配每支队伍的一志愿 1，再是一志愿 2，一志愿 3... 接着没选上项目的组和剩余项目再进行二分图匹配，优先匹配二志愿 1，再是二志愿 2，二志愿 3..
- 志愿分配原则：满足尽量多的队伍的一志愿，且必须为完备匹配，有更好算法的同学请联系我

所有队伍:

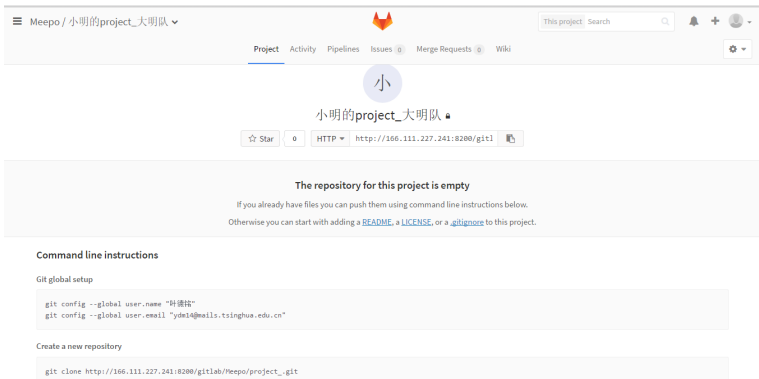
队名	队长	队员1	队员2	队员3	队员4	分组结果
大明队	叶德铭/2014011390					1

- 抽签完成后结果会在同一页面进行公示，并在队长的 gitlab 上创建项目。
- 名称为项目 gitlab 名（助教设定）+ 下划线 + 队伍名称
- 队员会自动 fork 该项目并成为该项目的 Master



- 用户名和密码的修改的网页上的修改页面已被去除
- 用 api 更改密码导致无法登陆的同学请自行改回来

在本地创建 git 仓库



- 按页面下方提示操作，有关 git 的使用请参见 google/软工课件等

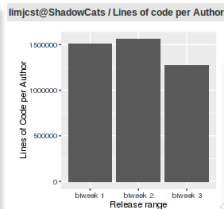
- Codeface 为一个大型项目分析工具，现已删减并修改成软工特供版。
- 点击小图可以看到该图的说明

limjst@ShadowCats / Commits biweek 3

	Name	Number of commits
1	陈文潇	58.00
2	isl	22.00
3	iamlockelightning	20.00
4	ZhouJianyu	20.00
5	mfhraven	16.00

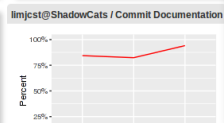
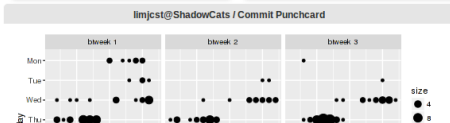
limjst@ShadowCats / Changes biweek 3

	Name	Added LOC	Deleted LOC
1	陈文潇	489438.00	5211
2	iamlockelightning	371146.00	935
3	mfhraven	15734.00	9
4	ZhouJianyu	1594.00	15
5	isl	98.00	



Lines of code per Author

Number of lines of code in the project divided by the number of Authors for a given cycle. Gives an indication of the maintenance burden of individual author.



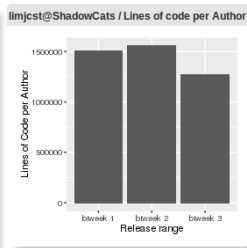
- 以两周为一个 range（但目前每个交界的 commit 会在两个 range 中重复统计）
- 从左至右：
该 range 中 commit 数排行，该 range 中修改行数排行。每个 range 中平均一个参与者代码行数

limjst@ShadowCats / Commits biweek 3

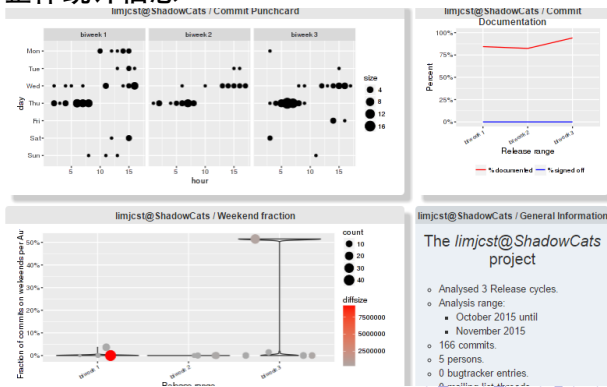
	Name	Number of commits
1	陈文潇	58.00
2	isl	22.00
3	iamlockelighting	20.00
4	ZhouJianyu	20.00
5	mfhraven	16.00

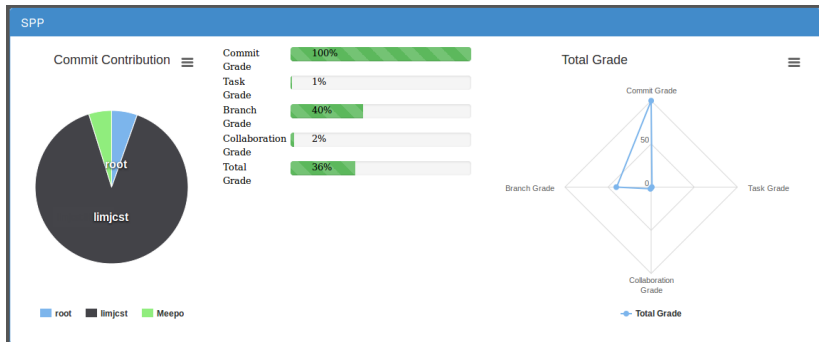
limjst@ShadowCats / Changes biweek 3

	Name	Added LOC	Deleted LOC
1	陈文潇	489438.00	5211
2	iamlockelighting	371146.00	935
3	mfhraven	15734.00	9
4	ZhouJianyu	1594.00	15
5	isl	98.00	



- 左上: 分星期几的 commit 的统计, 如左图中可以看出周三周四为该项目 commit 数的集中期
- 右上: 红线, commit message ≥ 10 bytes 的 commit 比例 (commit -m); 蓝线, commit 签名比例 (commit -s)
- 左下: 参与者在周末工作的比例
- 右下: 整体统计信息





评分结果仅供参考，具体细则参见“关于”页面

提交评分

- 提交评分用于评价项目成员各自的贡献。
- 提交积分的计算方式鼓励深思熟虑且为队友提供充分信息的提交。建议每次提交前使用git diff查看更改的内容，不使用-m等在命令行添加提交信息的选项以便在编辑器中充分描述本次提交。
- 每位成员在起止时间内的每个提交为该成员积 $e^{-\Delta t - \Delta l}$ 分。其中 Δt 为该提交与前一个提交的时间间隔减去一个小时，若差为正数则 Δt 取 0； Δl 为 20 减去提交信息的字符数，若差为正数则 Δl 取 0。小组内成员的积分除以组内积分的和即获得提交贡献。
- 最终，提交评分为

$$Contribution * (n + 2)$$

其中，Contribution 为成员的提交贡献，n 为组内小组人数。

- **注意：每个提交的作者由将其推到仓库时所用的用户确定！**

任务评分

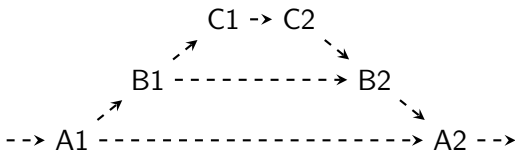
- 任务评分用于评价开发进程的任务分配情况。
- 任务评分的计算鼓励每位组员都分配到合适的任务。如果一项任务很快被完成了，组长应分配新的任务；如果一项任务长时间没有被完成，应该关闭任务后新开一个任务，并酌情更改任务内容。
- 项目开发过程中的所有任务都应该使用 GitLab 内的 Issue 功能来提出并分配给成员，任务评分即统计分配给各个成员的 Issue 的打开时间对项目开发进程的覆盖情况。以每个 Issue 建立时间中最早的一个和更新时间最晚的一个限定项目的开发时间。
- 最终，任务评分为

$$1 - \frac{\sum \Delta t + \sum \Delta p}{nT}$$

其中， Δt 是成员长于三天的未被 Issue 活跃时间覆盖的时间段， Δp 是 Issue 打开时间超出两周的部分， n 为小组人数， T 为项目开发时间。

分支评分

- 分支评分用于评价开发进程的分支使用情况。
- 分支评分的计算希望组员掌握处理合并冲突和使用分支的方法。
- 在起止时间内的提交树上，合并数的最小的分支含有三个合并时，可获得 40% 的分数，用于评价对合并冲突的处理能力；当提交树含有下图所示模式时，可获得另 60% 的分数，用于评价对分支的使用能力。



- **注意：请不要使用变基操作以免影响得分！**

合作评分

- 合作评分用于评价开发进程中小组成员的合作情况。
- 合作评分的计算希望小组成员的提交评分不应相差太大。
- 合作评分计算如下，

$$\frac{0.5 - \frac{\sqrt{\sum (CG - \overline{CG})^2}}{n-1}}{0.5 - 0.08}$$

其中， CG 是小组成员的提交评分， $\frac{\sqrt{\sum (CG - \overline{CG})^2}}{n-1}$ 是其标准差，计算结果小于 0 时取 0，大于 1 时取 1。当四人小组内只有一人有提交记录时，提交评分分布 (1, 0, 0, 0) 的标准差为 0.5，该式为 0；当三人小组的提交评分分布为 (0.35, 0.4, 0.25) 时，标准差为 0.076，得分为 1。

综合评分是上述四种评分的均值。

- Jenkins 是一个开源软件项目，旨在提供一个开放易用的软件平台，使软件的持续集成变成可能。



- 每次在 gitlab push 或 merge 后能自动触发 jenkins build 代码，支持 java,c++,ant

Enter an item name

小明的project

Required field



构建一个自由风格的软件项目

这是Jenkins的主要功能,Jenkins将会结合任何SCM和任何构建系统来构建你的项目,甚至可以构建软件以外的系统

- 具体方法参见“关于页面”的 Jenkins Jenkins:

Jenkins是一个开源软件项目，旨在提供一个开放易用的软件平台，使软件的持续集成变成可能。

1. 点击右上角sign up注册，点击log in，可勾选“Remember me on this computer”以便以后直接登录。
2. 点击左边列表中的New Item新建项目，输入项目名称，选择“Freestyle project”进行项目配置，点击项目配置页面每个单项右侧问号可获取详情。
3. 勾选“Enable project-based security”进行项目权限管理，User/group to add中加入该项参与人员的用户名并赋予权限（注意要把自己加进去），最右按钮为反选
4. 在Source Code Management中选择Git，输入仓库地址，如：<http://166.111.227.241:8200/gitlab/>.../xxx.git
5. 在Build Triggers选择构建触发器，建议选择Build when a change is pushed to GitLab。GitLab CI Service URL: xxx.另，Poll SCM为定时构建
6. 在Build中选择构建方法，如：(1)Execute shell并输入java test.java*执行java test (2)Invoke Ant并选择Ant Version，Target默认为build.xml ...
7. 在Post-build Actions中选择构建之后的操作，如：E-mail Notification，勾选“Send e-mail for every unstable build”并填入邮箱地址(多个邮箱空格分隔)，这样在构建失败时将通知邮件把错误信息发送至指定邮箱
8. 若选择Build when a change is pushed to GitLab。GitLab CI Service URL: xxx，须在gitlab项目设置中的webhooks新建一项，url为xxx并勾选Push Event和Merge Request events，点击Add webhook后可以在相应hook点击事件进行测试

谢谢大家！