







# 软工III-第二次迭代

软件学院-2018










# 迭代一情况总结

- 目前Gitlab上共上传了58个项目
  - 有少部分同学的用户名未包含学号，请修改
- 助教正在逐一检查系统、文档和Gitlab操作轨迹
  - 尚未出结果，以最终分数为准
- 多数同学在Gitlab上的操作存在问题
  - 将逐一指出，请在迭代二时严格遵守（详见“实践要求.pdf”文件，有明确的要求）！
- 请在之后的迭代中准备单元测试，方便助教检查

# 问题1：仓库结构设置有误

Name	Last Update
 counting-stars_phase_I	2 days ago
 counting-stars_phase_III_docs	16 days ago
 counting-stars_phase_II_docs	16 days ago
 counting-stars_phase_I_docs	2 days ago
 .gitignore	2 days ago
 README.md	19 days ago


## 问题2：没有使用Maven管理外部依赖jar包

Name	Last Update
..	
 javax.annotation.jar	4 days ago
 javax.ejb.jar	4 days ago
 javax.jms.jar	4 days ago
 javax.persistence.jar	4 days ago
 javax.resource.jar	4 days ago
 javax.servlet.jar	4 days ago
 javax.servlet.jsp.jar	4 days ago
 javax.servlet.jsp.jstl.jar	4 days ago
 javax.transaction.jar	4 days ago










## 问题3：文档出现在代码仓库中

Name	Last Update
 测试文档.pdf	2 days ago
 迭代一计划文档.pdf	7 days ago
 迭代二计划文档.pdf	7 days ago
 部署文档.pdf	2 days ago
 需求规格说明文档.pdf	7 days ago
 项目启动文档.pdf	7 days ago
 项目设计文档.pdf	7 days ago














# 问题4：文档本身没有按规范设置，文档分多人版本

 metting record	9 days ago
 系统用例图.asta	14 days ago
 系统用例图.png	14 days ago
 迭代I配置文档.pdf	3 days ago
 迭代一需求规格文档.doc	3 days ago
 迭代一需求规格文档.pdf	3 days ago
 迭代一项目总结 [redacted].txt	3 days ago
 迭代一项目总结 [redacted].txt	2 days ago
 迭代一项目总结 [redacted].txt	2 days ago
 迭代一项目总结 [redacted].txt	2 days ago
 迭代一项目计划及执行文档.pdf	3 days ago
 项目启动文档.docx	14 days ago
 项目启动文档.pdf	14 days ago

问题5：由于缺少Maven的管理，Node.js的node\_modules文件夹也上传了










Name	Last Update
..	
 bin	6 days ago
 node_modules	4 days ago
 public	a day ago
 routes	3 days ago
 views	a day ago
 README.md	7 days ago
 app.js	3 days ago
 package-lock.json	6 days ago
 package.json	5 days ago

# 使用依赖管理工具后不会上传










..	
 build	3 days ago
 config	3 days ago
 src	3 days ago
 static	8 days ago
 test	8 days ago
 .babelrc	8 days ago
 .editorconfig	8 days ago
 .eslintignore	8 days ago
 .eslintrc.js	8 days ago
 .gitignore	8 days ago
 .postcssrc.js	8 days ago
 README.md	8 days ago
 index.html	8 days ago










# 推荐实践1：文档充分，结构合理，文档分为md和pdf两个版本

Name	Last Update
 Delivery	2 days ago
 Design	3 days ago
 Requirement	8 days ago
 Test	3 days ago
 logo1.png	9 days ago
 小组和个人总结.md	2 days ago
 小组和个人总结.pdf	2 days ago
 项目启动文档.md	9 days ago
 项目启动文档.pdf	9 days ago

# 推荐实践2：使用Maven管理依赖

Name	Last Update
..	
 .idea	3 days ago
 .mvn/wrapper	7 days ago
 src	3 days ago
 Server.iml	6 days ago
 UserData.txt	3 days ago
 counts.iml	5 days ago
 mvnw	8 days ago
 mvnw.cmd	8 days ago
 pom.xml	6 days ago

# 推荐实践3：使用Gradle管理项目

Name	Last Update
 gradle/wrapper	12 days ago
 src	2 days ago
 ui	2 days ago
 .gitignore	8 days ago
 build.gradle	4 days ago
 gradlew	12 days ago
 gradlew.bat	12 days ago

# 通俗理解Maven和Gradle

## 为什么用ant

打包的时候要做很多事，比如说配置文件中的用户名和密码，你本地开发和程序实际运行时的内容肯定不一致，所以需要有两个文件，开发的时候用一个，实际运行的时候用一个。这样每次打包的时候都要重复的做这些事，ant可以让你用xml的形式把这些事情写成脚本，然后自动运行。

## 为什么用maven

你的项目要用很多jar包，比如你写日志要用个log4j吧，连数据库要用个connector吧。这年头写程序不用个spring都不好意思出门，下载下来的spring包打开一看，密密麻麻的好几十个jar，不知道用哪个不用哪个。而且，即便是你一狠心把这些jar包都放进来，很大可能性还是不能运行，因为还要依赖其它的jar包。哪天想升级个spring小版本，所有jar包都要重来一遍，你也不记得哪些是由于spring引进来的jar包了。

所以有了maven，你只要配置一下说我要用spring 3，所有jar包都给你下载好了，你直接运行就行了。赶明儿想升级版本，直接把3改成4，新的jar包也给你下载好了。

## 然后？

用了maven，jar包是方便了，但我打包的时候还是有好多事要做啊，然后你会发现maven实在是不知道怎么做这些事，于是开始怀念ant。

gradle就是又能干maven的活，又能干ant的活，用groove语言写脚本，表达能力还更强。

就这意思。

# 迭代二要求

- 完善管理系统
  - 具有管理员、众包工人（Worker）、众包发起者（Requestor）等角色众包
    - 发起者提交众包需求，发起众包请求
    - 系统进行众包分发
    - 众包工人完成标注任务
- 完善奖励系统。
  - 众包发起方可以提供一定额度的积分奖励
  - 众包工人完成任务后可以获得积分奖励，规则在迭代二可自行定义
- 系统可视化数据统计展示
  - 对不同角色展示统计数据例如但不限于：个人任务完成情况，众包发起方单个任务的完成情况，系统管理员总体的任务完成情况等

# 迭代二检查点

- 能够注册众包发起者、众包工人
- 众包发起者能够提交众包需求
- 系统能够创建并分发众包任务
- 众包工人能够接收并完成众包任务
- 众包发起者能够为众包任务设定积分奖励
- 众包工人能够获得一定的任务积分奖励
- 众包工人能够查看自己任务、奖励、群体排名等统计信息
- 众包发起者能查看自己任务数、完成进度、参与者等统计信息
- 系统管理员能够查看系统的用户数、任务数、进行中、已完成等统计信息

# 检查点注释

- 系统分发任务可以有多种形式。不限于系统指定分派给众包工人；众包工人自由选取众包任务。
- 不允许使用数据库。
- 需要详细的部署使用文档。