

---

# 协同系统 V2.1 用户手册

## 目录

协同系统 V2.1 用户手册 .....	1
一、系统概述 .....	4
1.1 系统用户 .....	4
1.2 系统组成 .....	4
1.3 主要业务功能概述 .....	4
1.3.1 课程教学部分 .....	4
1.3.2 敏捷实训部分 .....	5
1.4 系统版本 .....	7
1.5 联系我们 .....	7
二、模块介绍 .....	7
2.1 登录和主页 .....	7
2.1.1 平台地址 .....	7
2.1.2 登录界面 .....	7
2.1.3 平台首页 .....	8
2.1.4 账号管理 .....	8
2.2 教学-课程主页 .....	9
2.3 教学-课程资源 .....	10
2.4 教学-课程团队 .....	10
2.4.1 团队申请流程 .....	10

---

2.4.2 申请加入团队 .....	11
2.4.3 创建团队 .....	14
2.4.4 团队主页面 .....	15
2.5 教学-课程作业 .....	16
2.6 教学-课程考试 .....	19
2.7 实训-日志 .....	22
2.8 实训-故事地图 .....	23
2.8.1 故事地图看板 .....	23
2.8.2 史诗/用户故事 .....	24
2.9 实训-迭代 .....	25
2.10 实训-缺陷 .....	27
2.11 实训-统计 .....	29
2.12 实训-测试 .....	30
2.12.1 测试流程 .....	30
2.12.2 组内测试-测试用例 .....	30
2.12.3 组间测试-测试计划 .....	31
2.12.4 组间互评 .....	33
2.13 实训-代码评审&&代码仓库 .....	37
2.13.1 评审准备 .....	37
2.13.2 代码仓库设置 .....	38
2.13.3 人工评审 .....	39
2.13.4 自动评审 .....	41

---

2.14 实训-编译部署 .....	44
2.14.1 凭证管理 .....	44
2.14.2 主机管理 .....	45
2.14.3 任务管理 .....	46
2.14.4 任务成功执行的前提 .....	48
2.14.5 后端项目 .....	49
2.14.6 前端项目 .....	49

---

## 一、系统概述

### 1.1 系统用户

**系统管理员**是系统的管理者，负责管理所有用户的信息，管理所有课程信息；

**教师**是课程的管理者，负责管理和维护自己课程下的人员、团队、资源、作业、题库、考试以及敏捷实训部分的项目、迭代和互测等相关功能；

**学生**是课程的参与者，在课程部分：能够下载课程资源、维护课程团队、参与课程作业、参加课程考试、查看课程成绩等；在敏捷实训部分：能够查看操作日志、维护故事地图、管理迭代信息、维护缺陷看板、查看统计图表、参与自测互测、维护代码仓库、进行代码评审、线上编译部署等。

本用户手册仅面向学生角色的业务功能。

### 1.2 系统组成

本系统分为两大板块：**课程教学部分**和**敏捷实训部分**。

课程教学部分主要面向学期内的课程和教学工作。实现了线上的**资源分享**、**作业提交**、**课程考试**以及**成绩评价**等主要业务功能。

敏捷实训部分主要面向学期后的软件工程实践课的实践任务。实现了线上的**项目管理和版本控制**，包括迭代、看板（故事、任务、缺陷）、测试（组内测试、组间互评）、仓库、评审、部署等一系列主要业务功能。

### 1.3 主要业务功能概述

#### 1.3.1 课程教学部分

一门课程从创建到结束，中间将经历以下几个阶段。

首先，由系统管理员创建课程，并添加课程教师。

其次，课程教师进入课程主页，修改课程信息，导入课程学生名单。学生确认自己的账号登录，可以看到该课程，并根据教师或助教的要求，完善个人信息，组建课程团队（是否需要团队请听从授课老师的安排），这些是准备工作。

---

此后，每次线下授课结束后，教师可以通过本平台上传课程资料、发布课程作业；相应的，学生可以进行课程资料的下载，课程作业的提交。

最后，当课程临近结束时，教师可以开放课程考试。学生可以根据考试要求，在规定时间内进行线上考试。考试结束后，教师可以为学生分配课程成绩，学生可以查看自己的成绩，平台将提供一些辅助性的分析。（考试部分也可能是平时的依此测验）

### 1.3.2 敏捷实训部分

一次软件工程综合实践课，以集中实训的形式开展（以下简称“实训课”）。参与实训课的前提：

- （1）学生加入该门课程，并已经完善个人信息（email 信息是必要的）；
- （2）团队已完全分配完毕，且教师冻结了所有团队。

项目：实训的开展是以“项目”为基础的。在实训课开始时，教师会为所有的团队建立团队项目。在一次实训课程中，团队和项目是相互唯一对应的，即一个团队有且只有一个项目，一个项目能且只能隶属于某一个团队。实训课接下来的所有过程，包括迭代、看板、统计、测试、评审、部署、以及日志，都是为了推进项目而设置和开展的。

迭代：“迭代”是开发过程中的基本单位，一个项目可能会经过多次迭代完成，而每次迭代团队都需要向用户交付一个有价值的项目，并通过多次迭代最终完成满足用户需求的成品。由于平台是面向软件工程实训课的，因此，所有团队的迭代周期应保持同步。教师通过平台为所有团队创建一次迭代，此时迭代就正式开始了。在一次迭代中，团队成员可以利用看板直观地控制迭代进度，也可以通过统计来把握项目和团队成员的状态，可以利用缺陷看板进行缺陷管理，也可以创建测试计划进行组内测试，可以进行评审检查代码质量，也可以利用平台编译部署，实时交付用户需求的产品。

看板：实训项目通过看板来实现可视化进度管理，包括故事看板、任务看板、缺陷看板。通过看板，团队成员可以快速地了解每个用户故事、任务、缺陷、测试用例的负责人和状态等信息。团队中所有成员的进度可以直观地体现在看板上，使开发过程对小组透明化，提高团队成员间沟通的效率。

---

**故事和任务：**故事是敏捷软件开发中使用的工具，用于从最终用户的角度捕获软件功能的描述。故事分为史诗故事和用户故事，史诗故事粒度较大，用户故事粒度较小，史诗故事可以是一系列用户故事的集合。用户故事可以继续拆分成任务，任务则不可再分，是实现某一个功能点的最小单位（例如：实现某一个功能点是一个用户故事，进而可以拆分成前端开发任务和后端开发任务）。值得注意的是，故事和任务之间存在一定的关系，但并非极其严格的依赖关系。一个用户故事下的所有任务全部完成，是该用户故事完成的必要不充分条件。

**统计：**统计部分包括项目或迭代中的燃尽图和情绪图两个部分。项目开始后，团队成员可以通过每日更新的用户故事进展所生成的燃尽图等信息了解到项目的进展情况，以帮助项目负责人掌控进度。此外，团队管理员也可以通过小组成员每天提交的情绪数据所生成的情绪图来了解小组成员的心理状态，提前发现潜在的问题。

**测试：**项目测试对于开发过程来说，是必要的。其中，测试计划是测试的基本单位，测试用例是测试的最小单位。测试用例是与用户故事相关联的，但这种关系属于多对多的关系，即一个测试用例可以关联多个用户故事，一个用户故事也可以被多个测试用例关联。同样的，测试计划与测试用例之间也属于多对多关系，即一次测试计划可以包含多个测试用例，一个测试用例也可以从属于不同的测试计划。需要根据团队项目的具体，由团队管理者需求自主决定。

**评审：**团队可以使用评审部分对代码质量进行控制。评审部分包含人工评审和自动评审（静态代码审查）。人工评审以 Gitlab 中的 MR 作为评审粒度，评审者可以查看 MR 中的文件 diff 信息以及 commit 信息来选择 MR 对应推进的用户故事，并回答可修改性、可测试性等相关问题最后反馈给提交者；自动评审是团队成员可以根据语言等需求选择适当的工具对 Gitlab 保存的代码进行静态检查，团队可以根据检查结果对代码进行进一步的改进。

**部署：**协同教学平台敏捷实训模式的编译部署功能可以帮助学生更加便利地完成软件项目的交付，避免每次进行项目的改动都要进行一次手动编译部署，并提供给教师管理功能，使得授课教师能随时查看学生的项目完成情况。学生可以在平台注册开通代码仓库并完成开发过程中的代码版本控制，在完成一个迭代周

---

期后，可以通过编译部署管理系统创建一个构建部署任务，任务的内容包括代码仓库地址、编译指令和开放端口等信息。创建完毕后即可手动点击触发执行，或者在主分支发生变动时系统自动触发执行。执行完毕后，学生的软件项目即完成了部署，学生可以根据服务器的 IP 地址访问。如果失败了，则可以通过历史记录找到当次的控制台输出，查明原因。

## 1.4 系统版本

当前版本为 **V2.1**，目前不包括成绩部分、通知部分。

## 1.5 联系我们

使用本系统的过程中如有任何问题，请直接联系管理员（微信：L-18743236295），电话：15652573922。

# 二、模块介绍

## 2.1 登录和主页

### 2.1.1 平台地址

<http://219.224.166.3:9996/login>

### 2.1.2 登录界面



图 2.1.1

用户输入用户名和密码登录，通常用户名和密码均为当前学号。如图 2.1.1 所示。如果无法登录，请联系系统管理员。

### 2.1.3 平台首页

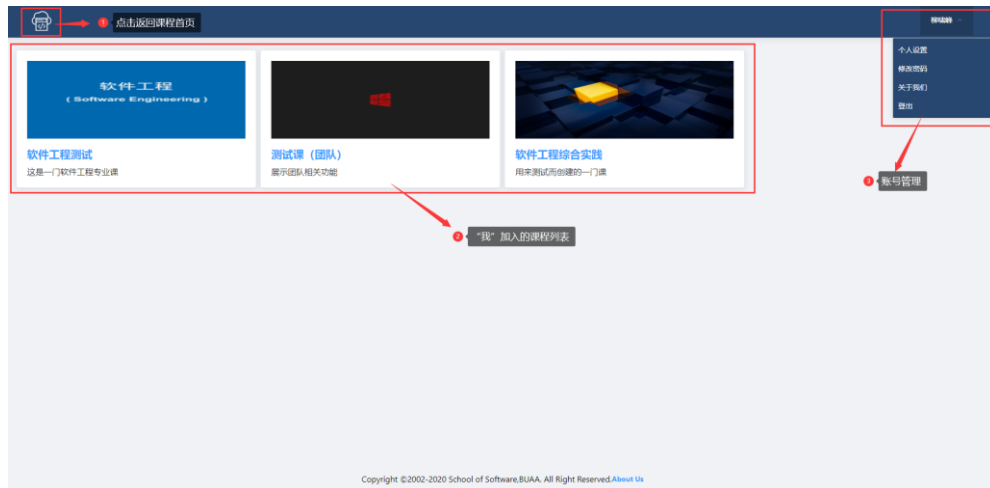


图 2.1.2

用户登录后，进入首页，可以看到三个部分的内容，如图 2.1.2。左上角图标为返回首页按钮。右上角的菜单是账号管理，包括个人设置、修改密码、关于我们以及登出。首页的中间内容是当前学生所加入的课程列表。点击课程卡片将进入对应的课程主页。

### 2.1.4 账号管理

The screenshot shows the 'User Settings' (用户设置) page. It has a dark blue header with a home icon and a 'Settings' (设置) link. The main content area contains several input fields: 'Email' (邮箱) with the value 'test@buaa.com', 'Name' (姓名), 'Phone' (手机号), 'Password' (密码), 'Student ID' (学号), and 'Student Number' (学号). There is a blue 'Save' (保存) button at the bottom. The footer contains copyright information for the School of Software, BUAA.



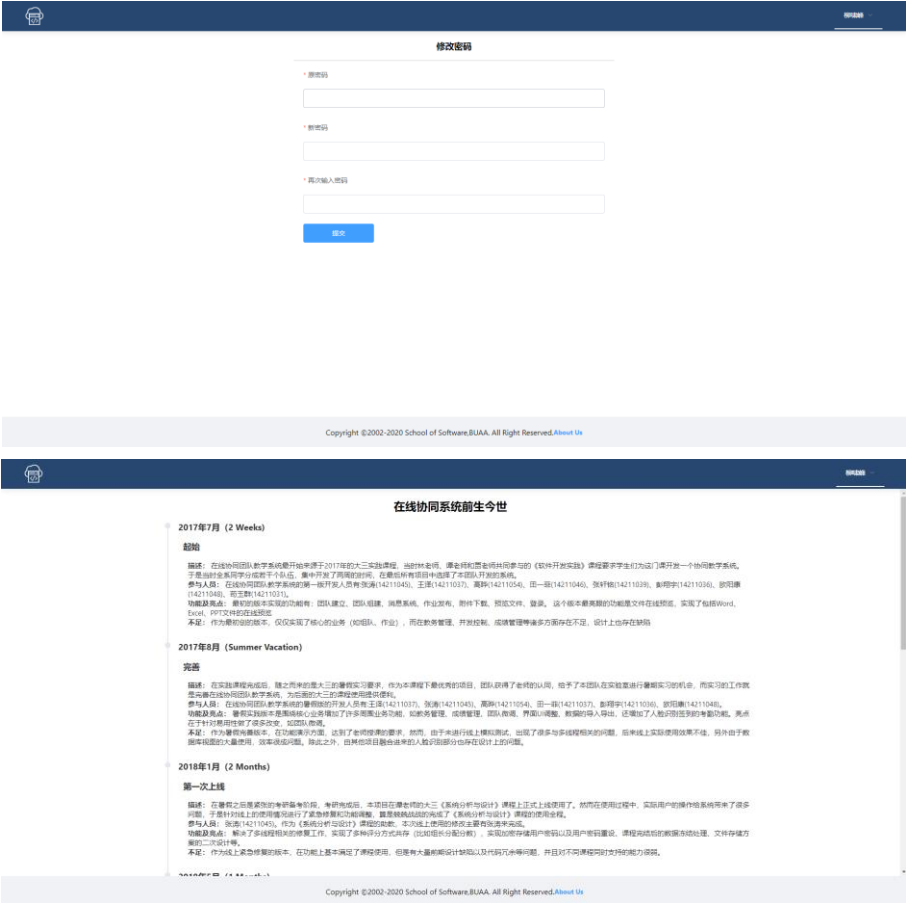


图 2.1.3

以上图 2.1.3 三张图分别为用户管理、修改密码、和关于我们。

## 2.2 教学-课程主页

从首页点击某个课程卡片，可以进入对应的课程主页。如下图 2.2.1 所示。

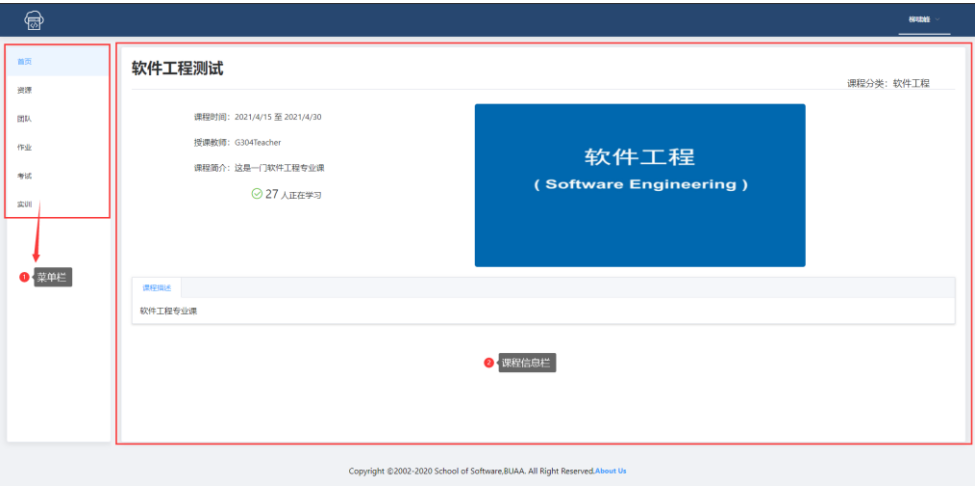


图 2.2.1

其中，课程信息包括课程名称、课程分类、课程时间、授课教师、课程简介、课程描述以及封面图片，对于以上信息，学生仅有查看功能。

从左侧的菜单栏可以进入对应的功能页，文档接下来将展开说明。

## 2.3 教学-课程资源

从课程主页左侧菜单栏中点击资源按钮，将进入课程资源页面。如下图 2.3.1 所示。

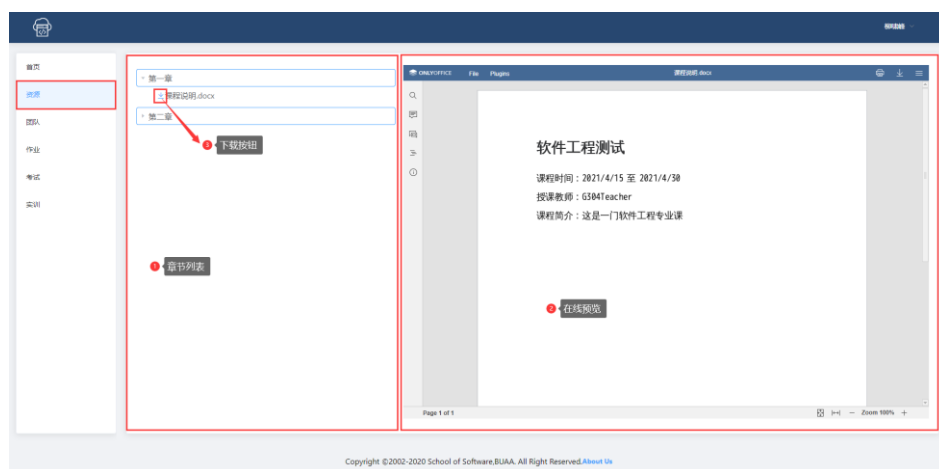


图 2.3.1

该页面分为左右两部分，左侧为章节资源列表，点击对应文件名可以在右侧在线预览文件内容，目前仅 word、xlsx、ppt 以及视频支持在线预览；点击文件名左侧的下载按钮们进行文件下载。

## 2.4 教学-课程团队

### 2.4.1 团队申请流程

**学生默认是没有团队的**，在进入团队界面后，可以选择**创建团队**，成为团队管理员；也可以选择已有的团队列表中选择某一个团队**发出申请**，并等待该团队管理员审核。**在审核期间，该生可以不断地向其他团队发出申请，直至成功加入任何一个团队为止。**成为团队管理员之后，将等待团队成员申请入队，团队管理员有权利决定是否通过某些申请。

当然教师用户有权微调任何一个团队，以保证教学团队的合理性。

此外团队相关操作每次更新都会通过系统通知，推送到相关学生用户。（当前版本不包括通知功能）。

学生申请团队流程图，如下图 2.4.1 所示。

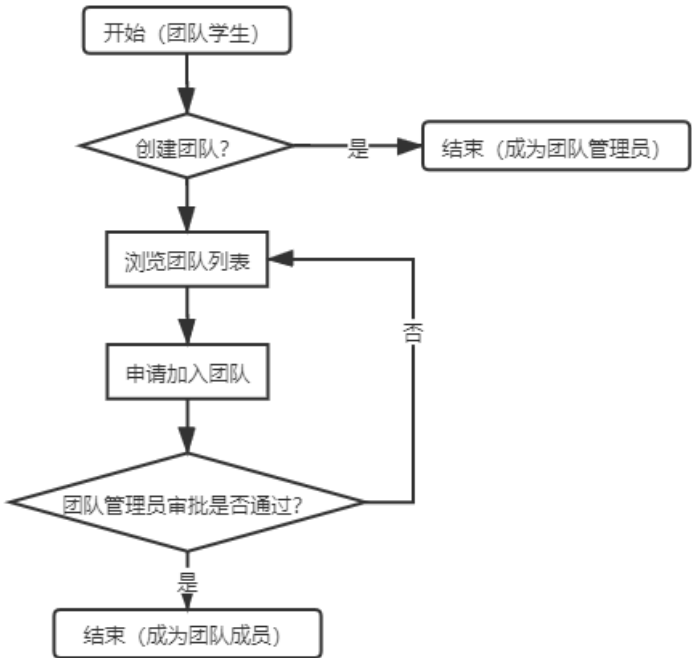


图 2.4.1

### 2.4.2 申请加入团队

当一个未加入任何团队的学生进入到团队页面时，会看到如下图 2.4.2 所示的页面。

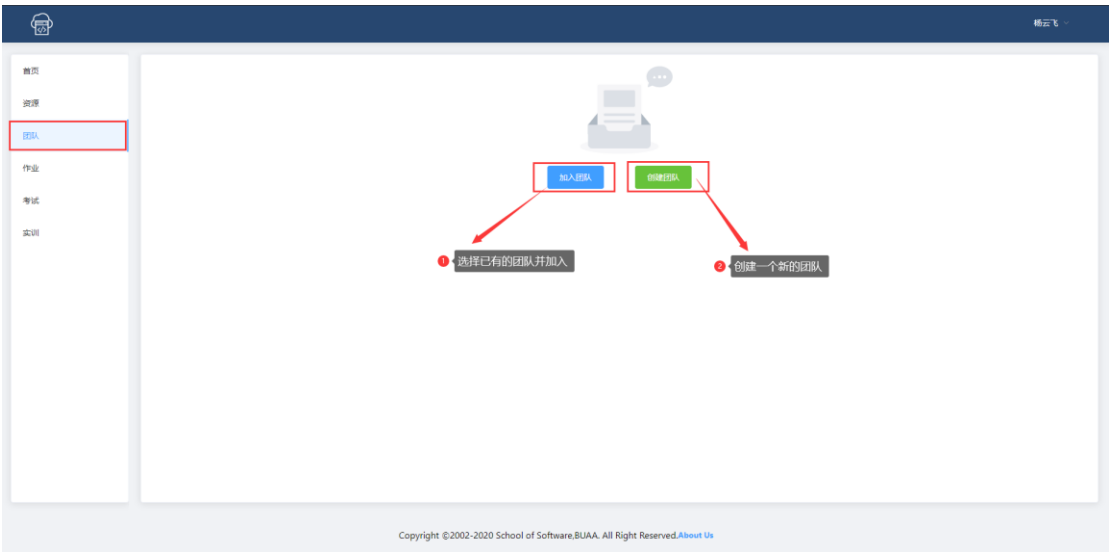


图 2.4.2

可以看到，学生可以选择已有的团队进行加入，或是创建一个新的团队，成为团队管理员。

当学生点击左侧“加入团队”按钮时，可以查看当前所有队伍列表，如下图 2.4.3 所示。

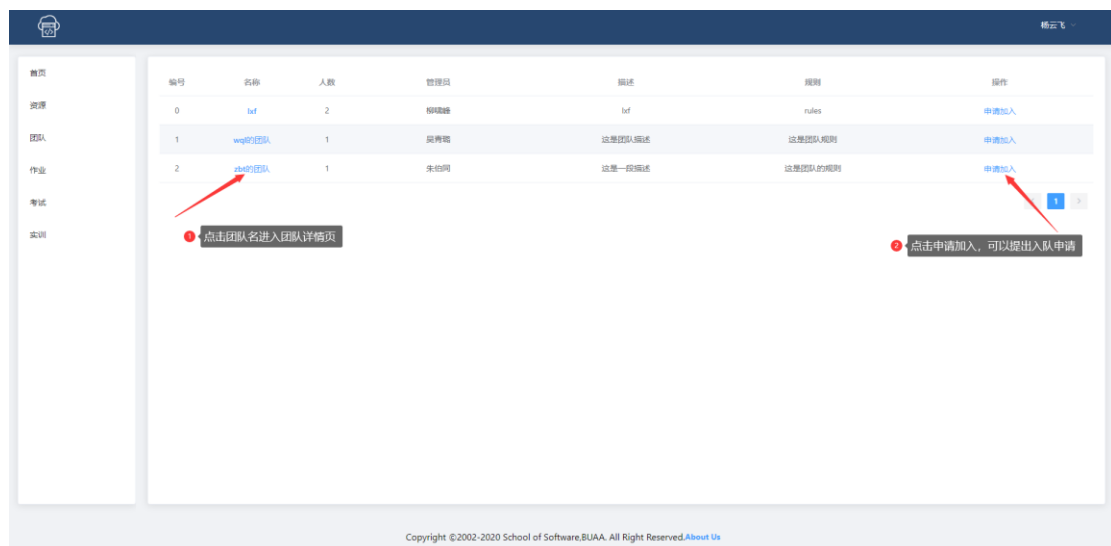


图 2.4.3

点击左边团队名称，可以进入到对应团队的详情页面，包括团队名称、团队描述、团队规则、团队成员等信息。如下图 2.4.4 所示。

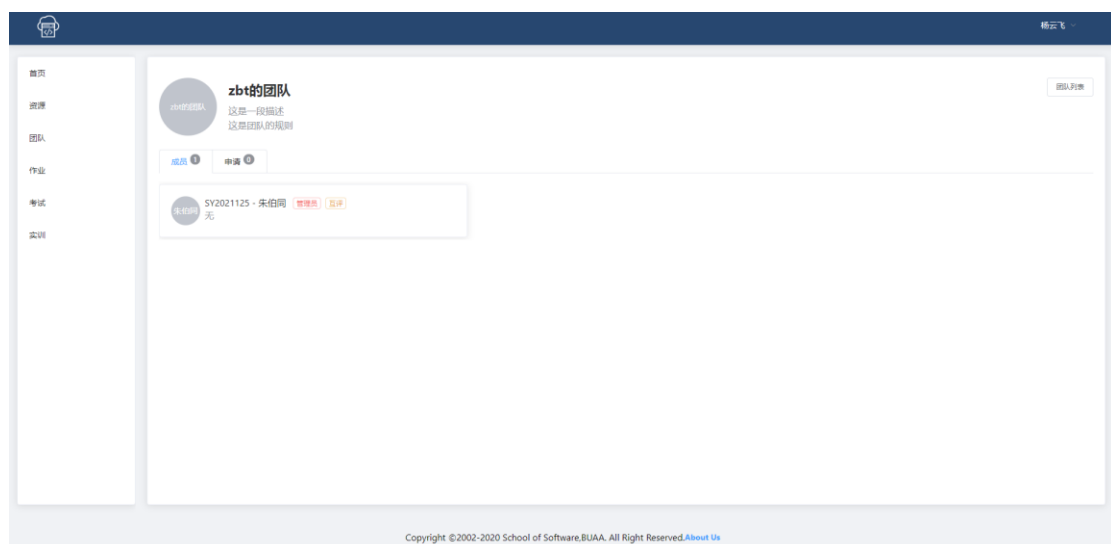


图 2.4.4

在团队列表页（图 2.4.3）点击右侧申请加入按钮，可以提出加入团队申请。

如下图 2.4.5 所示。

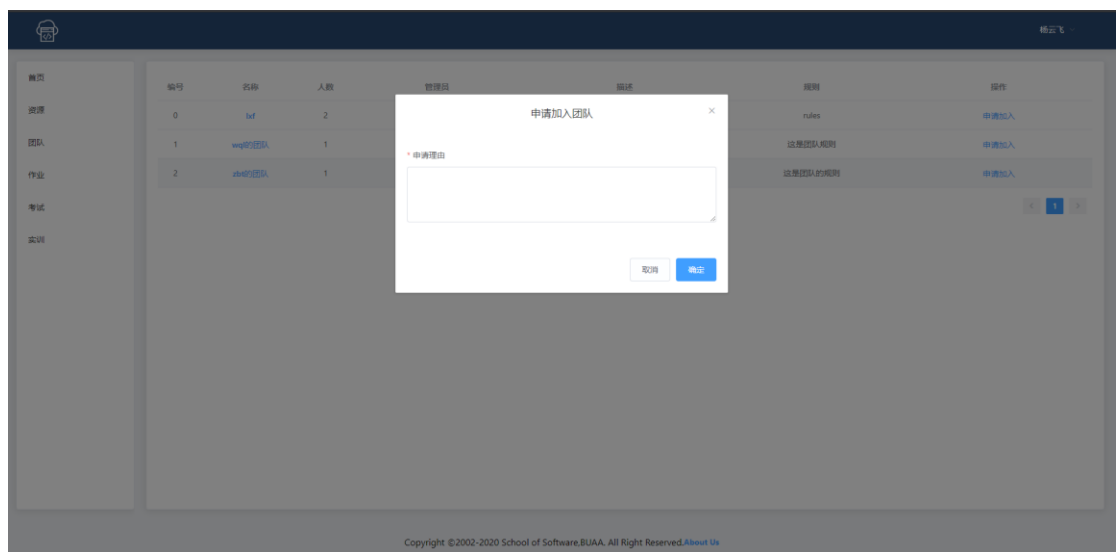


图 2.4.5

输入申请理由后，点击确认，则申请消息将被成功发送给该团队管理员。如下图 2.4.6 所示。

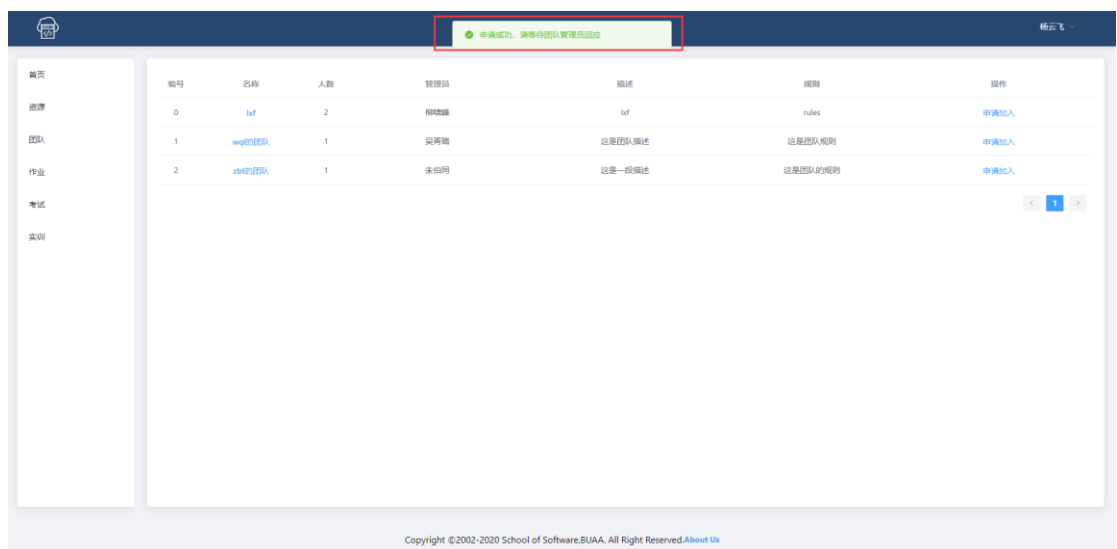


图 2.4.6

团队队长可以对申请信息进行操作：同意入队，或者拒绝入队（**✓按钮为通过申请，×按钮为拒绝申请**）。如下图 2.4.7 所示。

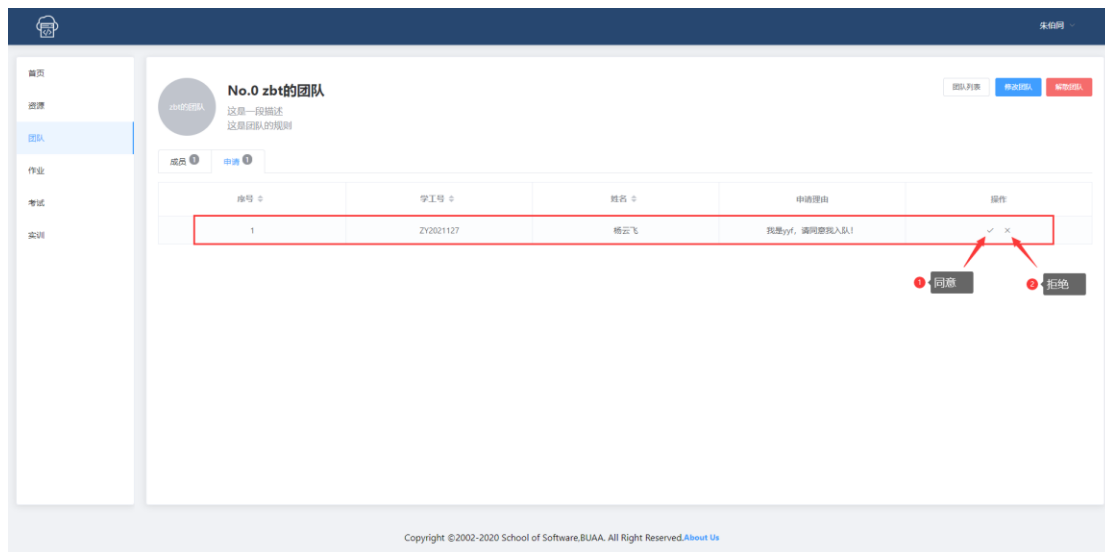


图 2.4.7

点击同意后，队伍人员随之更改。如下图 2.4.8 所示。

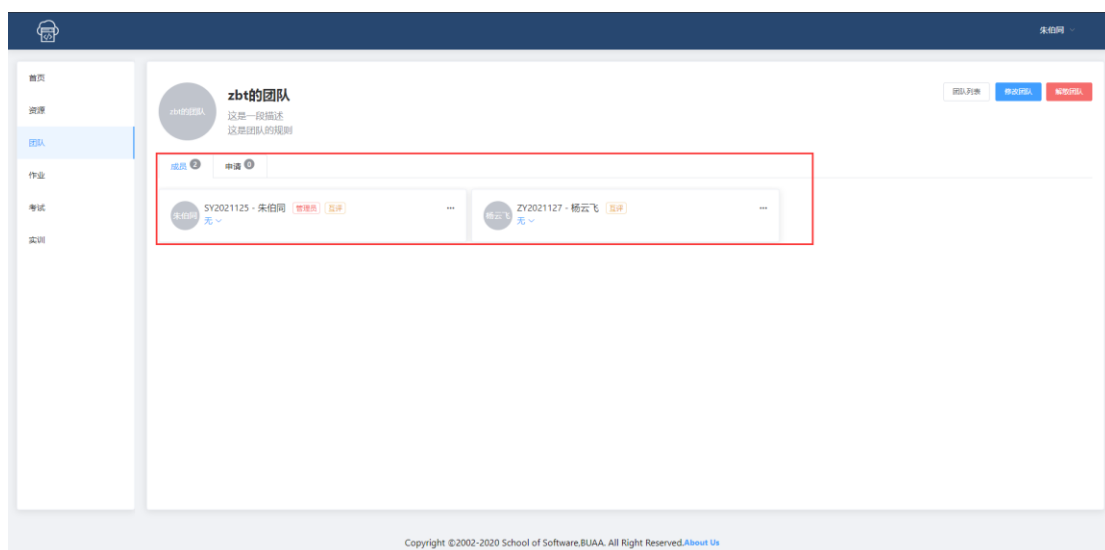


图 2.4.8

### 2.4.3 创建团队

未加入任何团队的学生可以创建团队。如下图 2.4.9 所示。

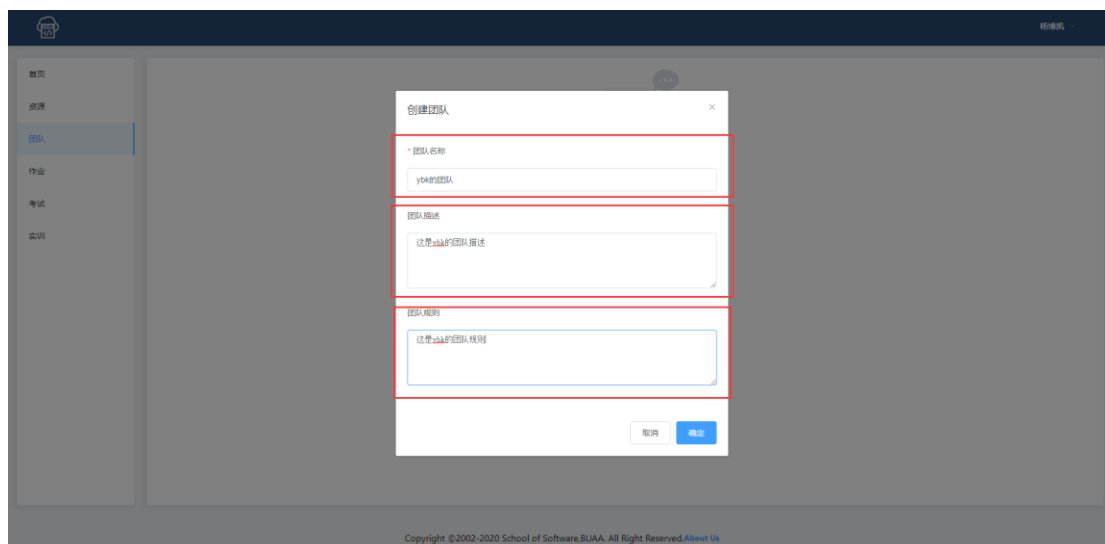


图 2.4.9

输入团队名称、团队描述、团队规则，点击确定，将会在团队列表中生成一条记录，并且团队处于等待其他学生加入的状态。如下图 2.4.10 所示。

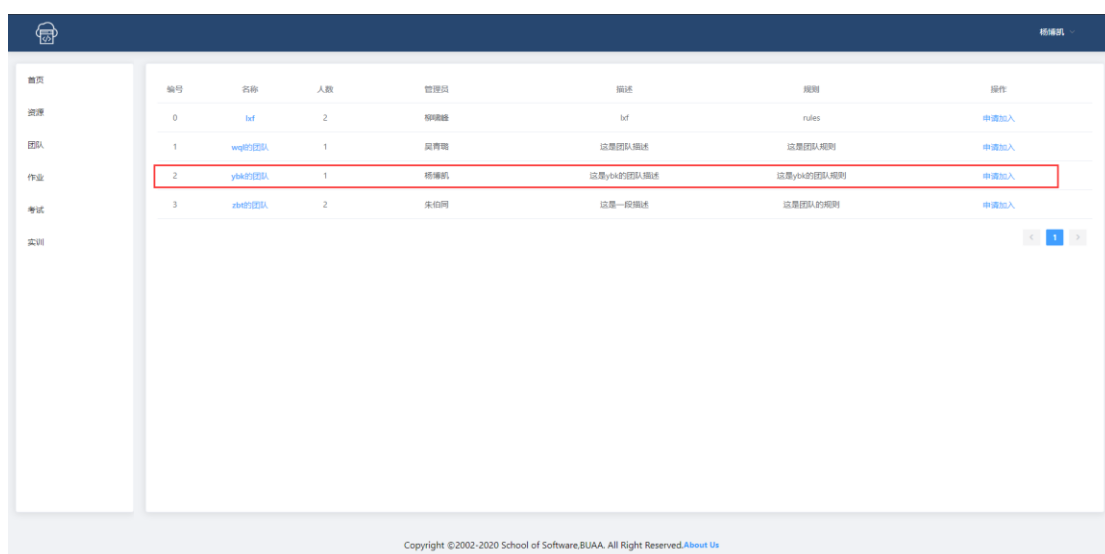


图 2.4.10

## 2.4.4 团队主页面

已加入团队的学生点击左侧菜单栏“团队”按钮后，将直接进入到自己的团队主页。如下图 2.4.11 所示。

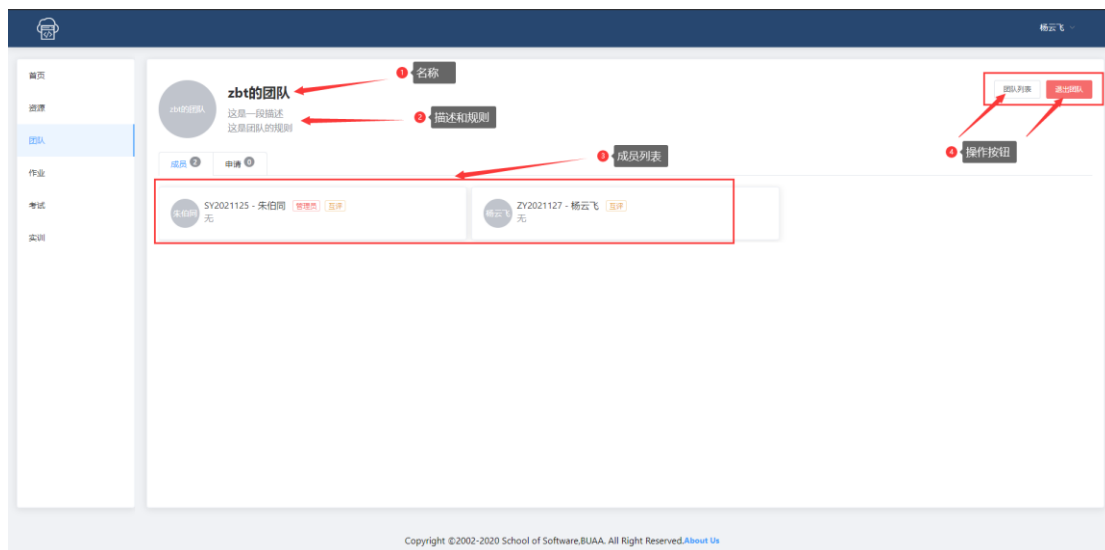


图 2.4.11

团队主页之内，除了团队基础信息之外，右上角还有一些功能操作，例如：返回团队列表，以及退出团队。团队管理员进入团队主页后，还拥有管理团队的部份权限，如下图 2.4.12 所示。

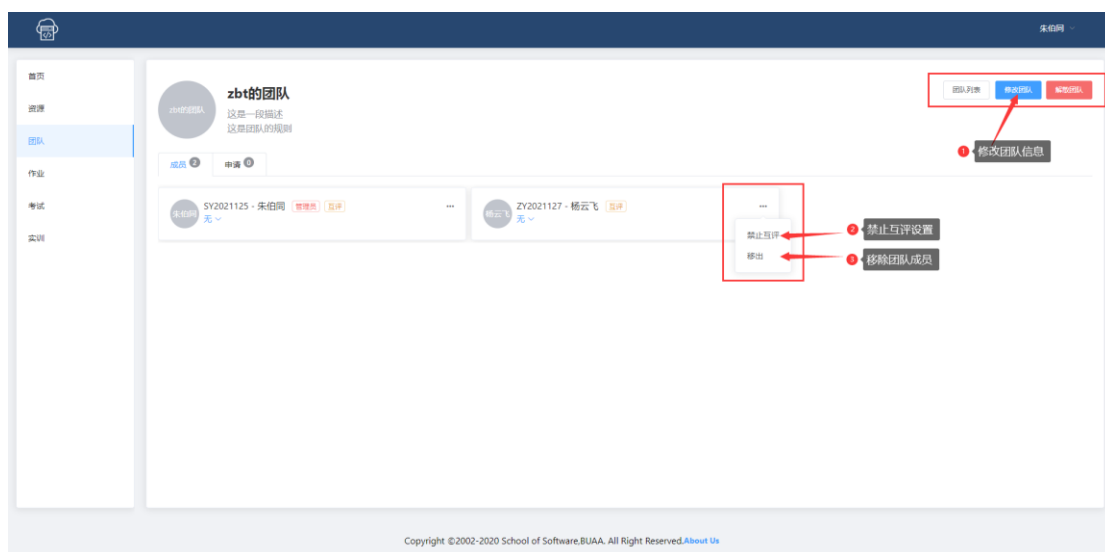


图 2.4.12

## 2.5 教学-课程作业

学生点击左侧菜单栏“作业”按钮，可以进入当前课程作业列表页。如图 2.5.1 所示。



序号	标题	类型	状态	截止时间	评测
1	第一次作业 (互评)	个人作业	进行中	2021-05-14 00:00 至 2021-05-19 00:00	<a href="#">查看互评</a> <a href="#">查看总分</a>
2	第一次作业	个人作业	进行中	2021-05-14 00:00 至 2021-05-19 00:00	<a href="#">查看互评</a> <a href="#">查看总分</a>
3	团队第一次作业 (互评)	团队作业	进行中	2021-05-14 00:00 至 2021-05-19 00:00	<a href="#">查看互评</a> <a href="#">查看总分</a>
4	团队第一次作业	团队作业	进行中	2021-05-14 00:00 至 2021-05-19 00:00	<a href="#">查看互评</a> <a href="#">查看总分</a>

图 2.5.1

作业的种类根据人数限制可以分成**个人作业**和**团队作业**，根据教师需求可以分成**可互评作业**和**不可互评作业**。在作业列表中，可以看到每一个作业的标题、类型（个人作业、团队作业）、状态（未开始、进行中、互评中、已结束）、起止时间、以及两个功能操作（查看互评和查看总分）。由于当前版本还不包括成绩部分的相关功能更，因此目前“评阅”中的“查看总分”并没有实装。

学生选择某一次作业，点击进入作业详情页面，来查看这次作业的详细信息，下载作业附件，并提交作业。如下图 2.5.2 所示。



图 2.5.2

作业详情页面由上到下的内容依次为：作业简介、作业附件、MD 编辑器、提交部分。点击上传附件可以选择本地文件上传至系统（**注意，在未点击提交作业前，系统不会保存你上传的文件！**），点击提交作业，会提示提交成功。如下图 2.5.3 所示。

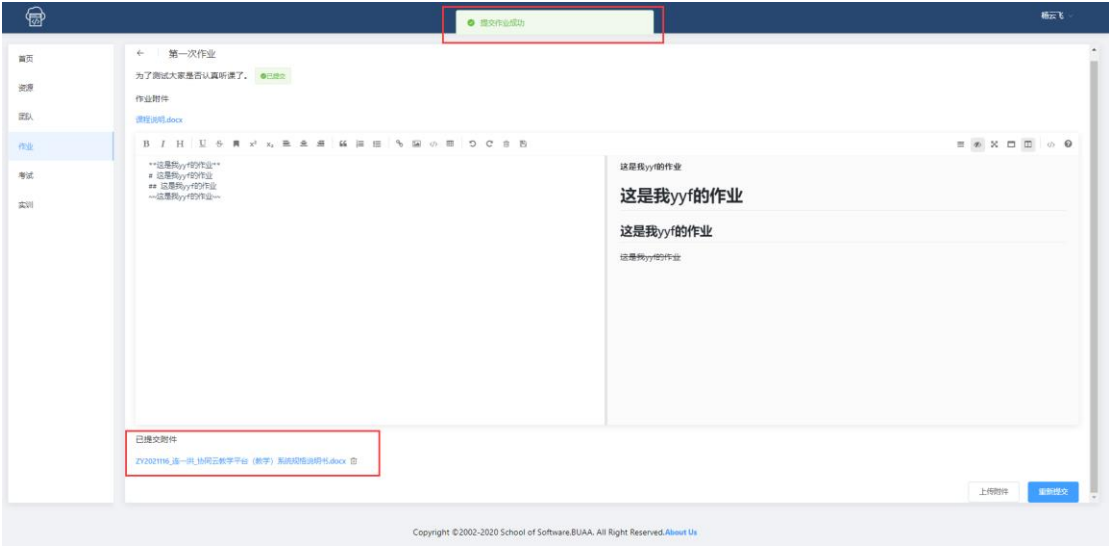


图 2.5.3

此时左上角变成已提交状态，鼠标悬停右下角“重新提交”会进行提示。如图 2.5.4 所示。

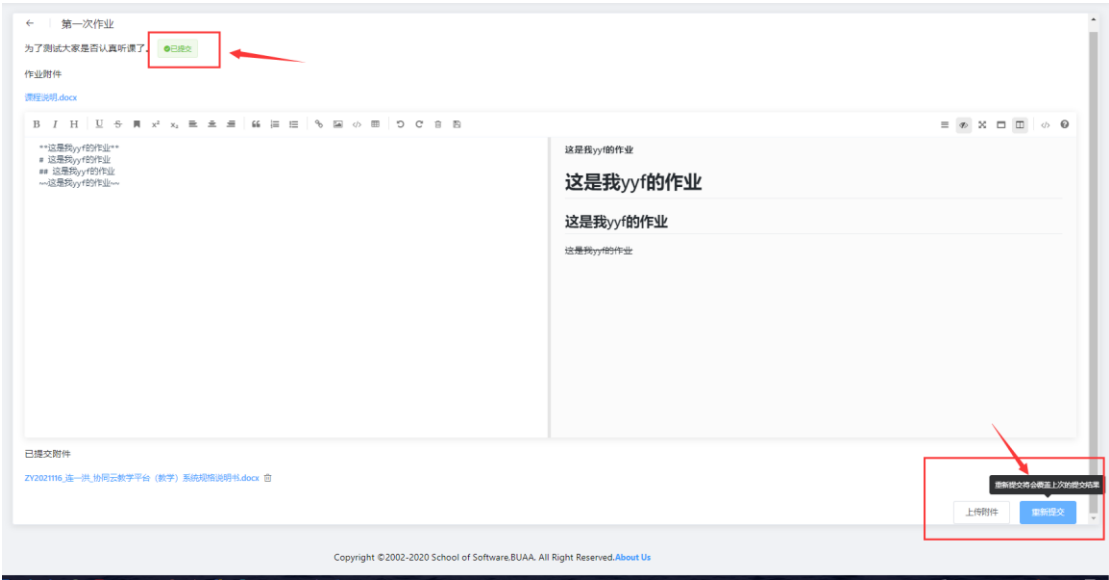


图 2.5.4

点击“查看互评”按钮，可以进入到互评界面。可以看到当前学生用户所需要的评价的作业列表，包括被评人的提交情况以及评分。如下图 2.5.5 所示。

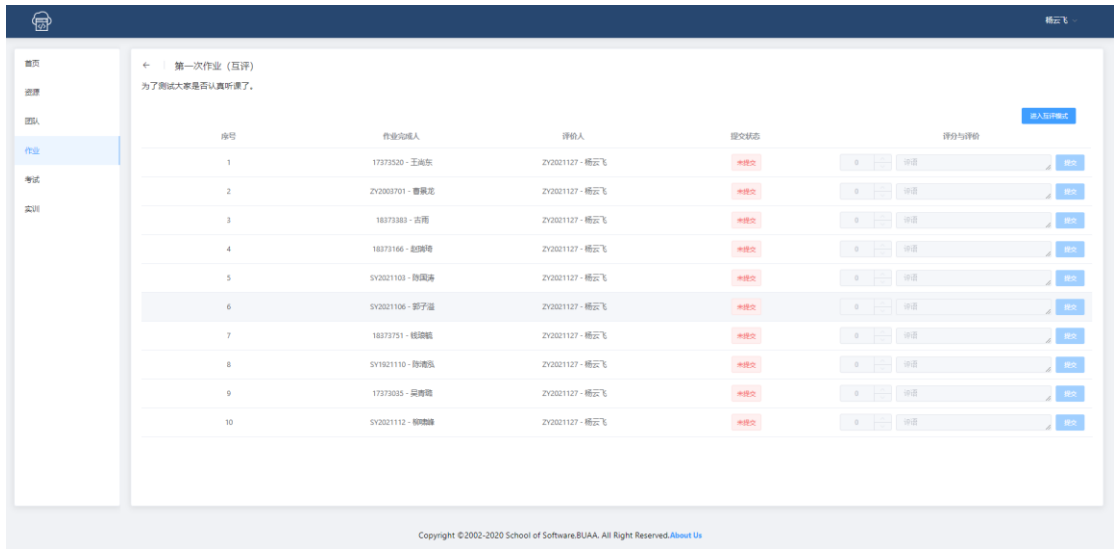


图 2.5.5

团队作业只能够由团队队长进行提交，提交过程与个人作业相一致。

## 2.6 教学-课程考试

学生点击菜单栏考试选项，进入到考试列表，可以查看到查看所有的考试信息。其中包括考试标题、起始时间、考试得分、以及功能操作栏。如下图 2.6.1 所示。

考试标题	开始时间	结束时间	考试得分	操作
4.28考试	2021-04-28 11:44	2021-04-28 00:00	暂未开放成绩查看	<a href="#">请签到</a> <a href="#">进入考试</a> <a href="#">查看考试结果</a>
5.4考试	2021-05-04 14:10	2021-05-04 00:00	暂未开放成绩查看	<a href="#">已签到</a> <a href="#">进入考试</a> <a href="#">查看考试结果</a>
5.13教师手册	2021-05-13 10:59	2021-05-13 11:01	暂未开放成绩查看	<a href="#">已签到</a> <a href="#">进入考试</a> <a href="#">查看考试结果</a>
11	2021-05-13 11:24	2021-05-13 12:23	暂未开放成绩查看	<a href="#">请签到</a> <a href="#">进入考试</a> <a href="#">查看考试结果</a>

7条/页

[<](#) [1](#) [>](#)

图 2.6.1

学生点击功能操作栏中的“签到”按钮进行签到，在考试时间开始时可以通过操作栏中“进入考试”按钮进入考试。考试界面如下图 2.6.2 所示。左上方点击返回列表按钮即可返回考试库，左侧可以查看试卷名称、考试剩余时间，点击答题卡中对应的数字将链接到对应的题目。

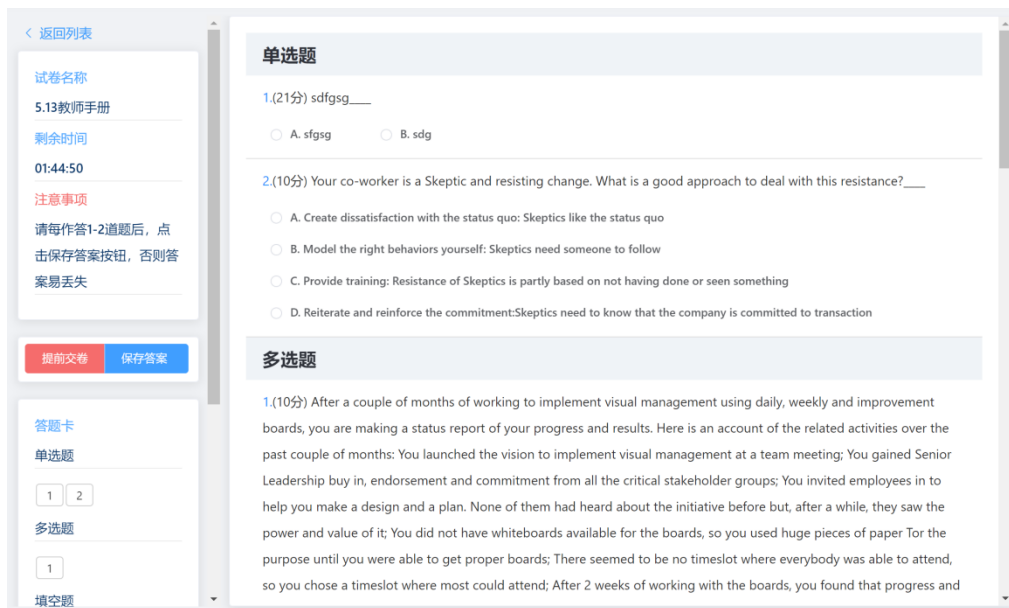


图 2.6.2

试卷中，对于单选题目与多选题目，点击对应的选项即可完成作答，对于填空题需要将答案用逗号分隔开，如下图 2.6.3 三图所示。

单选题

1.(21分) sdfgsg\_\_\_\_  
☒ A. sfgsg ☐ B. sdg

2.(10分) Your co-worker is a Skeptic and resisting change. What is a good approach to deal with this resistance?\_\_\_\_  
☒ A. Create dissatisfaction with the status quo: Skeptics like the status quo  
☐ B. Model the right behaviors yourself: Skeptics need someone to follow  
☐ C. Provide training: Resistance of Skeptics is partly based on not having done or seen something  
☐ D. Reiterate and reinforce the commitment:Skeptics need to know that the company is committed to transaction

填空题

1.(10分) 软件质量保证中，使用了\_\_\_\_思想和\_\_\_\_思想  

请输入填空题答案，如有多个空请用逗号分隔开

简答题

1.(1分) 简述产品质量的属性  

请输入答案

图 2.6.3

点击保存答案按钮将保存当前作答。同时在上方提示保存答案成功。点击提前交卷按钮会弹出提示框，**点击确认后将无法返回该考试页面**。如下图 2.6.4 所示。



图 2.6.4

学生点击图 2.6.1 功能操作栏中的“查看考试结果”按钮可查看对应的考试结果，**当且仅当教师批改完成后学生才可进行查看考试结果**。当教师允许查看考试结果时，学生点击可查看题目解析，参考答案等信息。如下图 2.6.5 所示。

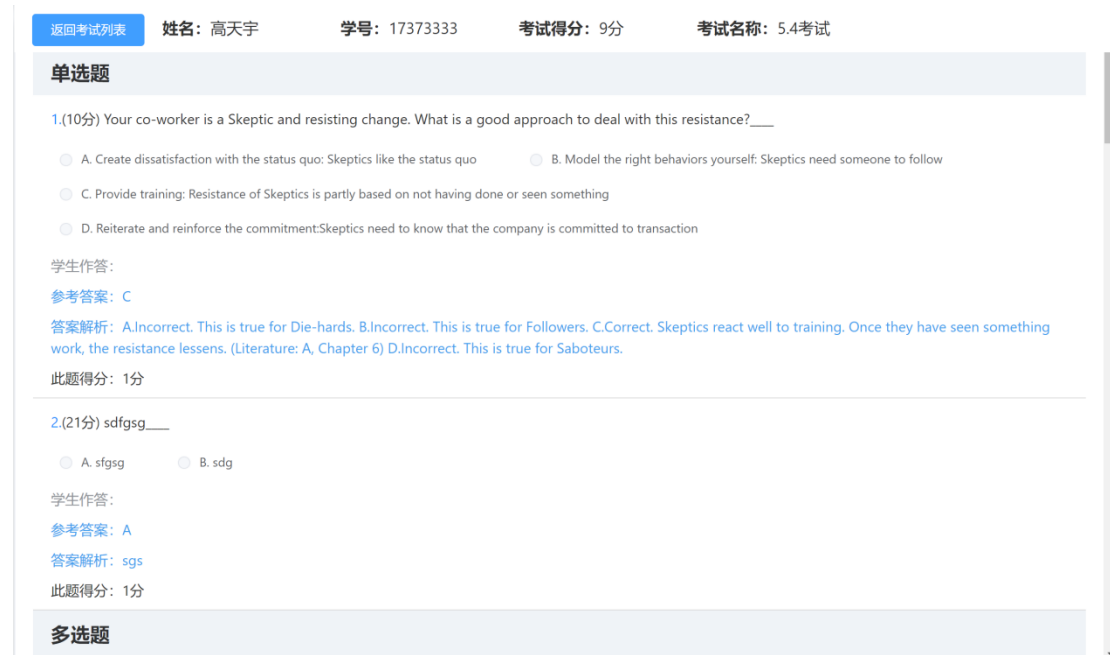


图 2.6.5

当教师不允许查看考试结果时，学生无法查看题目解析、参考答案等信息，但是可在考试栏目中查看该考试得分。如下图 2.6.6 所示。

不允许查看考试结果!

考试标题	开始时间	结束时间	考试得分	操作
4.28考试	2021-04-28 11:44	2021-04-28 00:00	暂未开放成绩查询	<a href="#">请签到</a> <a href="#">进入考试</a> <a href="#">查看考试结果</a>
5.4考试	2021-05-04 14:10	2021-05-04 00:00	0	<a href="#">已签到</a> <a href="#">进入考试</a> <a href="#">查看考试结果</a>

图 2.6.6

2.7 实训-日志

学生通过点击页面菜单栏的“实训”，即可进入敏捷实践部分，实训部分默认页面是日志记录，以展示项目成员的相关操作，具体为对故事、任务和缺陷的编辑操作等，也可以对某一段时间的日志进行查询。下面对故事地图的编辑在日志中均会体现。页面下方有加载更多，可以看到更多的操作条例。如图 2.7.1 所示。

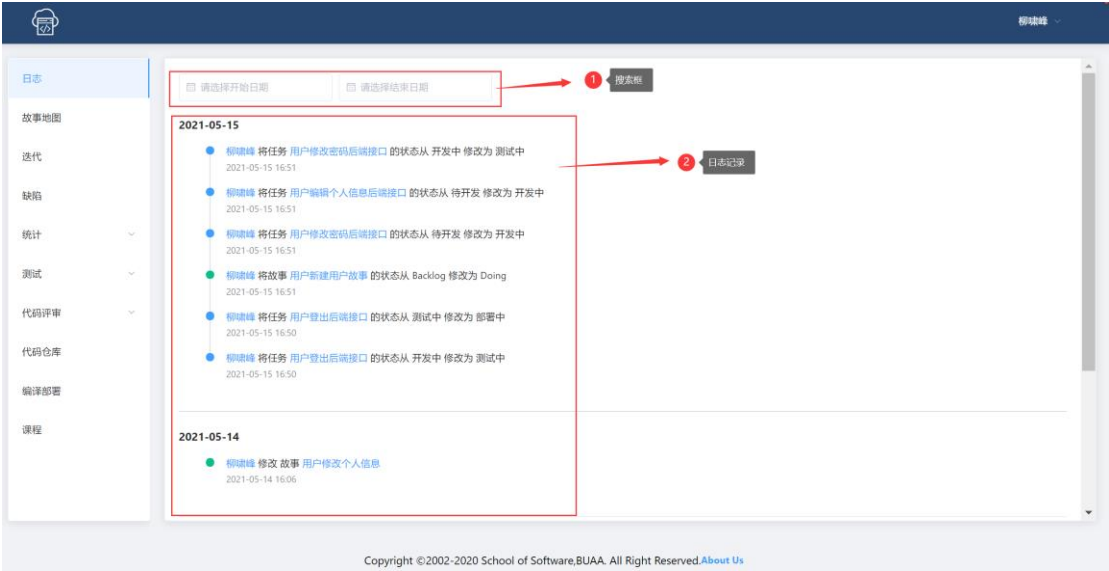


图 2.7.1

## 2.8 实训-故事地图

### 2.8.1 故事地图看板

学生点击故事地图进入故事地图页面，可以看到全部的用户故事分布在不同迭代的行和史诗故事的列中，可以点击史诗故事或用户故事卡片打开具体信息。可通过拖动改变用户故事卡片所属的行列。如下图 2.8.1 所示。



图 2.8.1

用户可以点击看板上方的按钮添加史诗故事、以及用户故事，新建用户故事如下图 2.8.2 所示，输入标题、描述、史诗故事、迭代等基础信息后点击新建，则会新建至对应位置。若不指定史诗故事，则会新建到“其他”一行；若不指定迭代，则会新建到“为规划的故事”一行。

## 2.8.2 史诗/用户故事

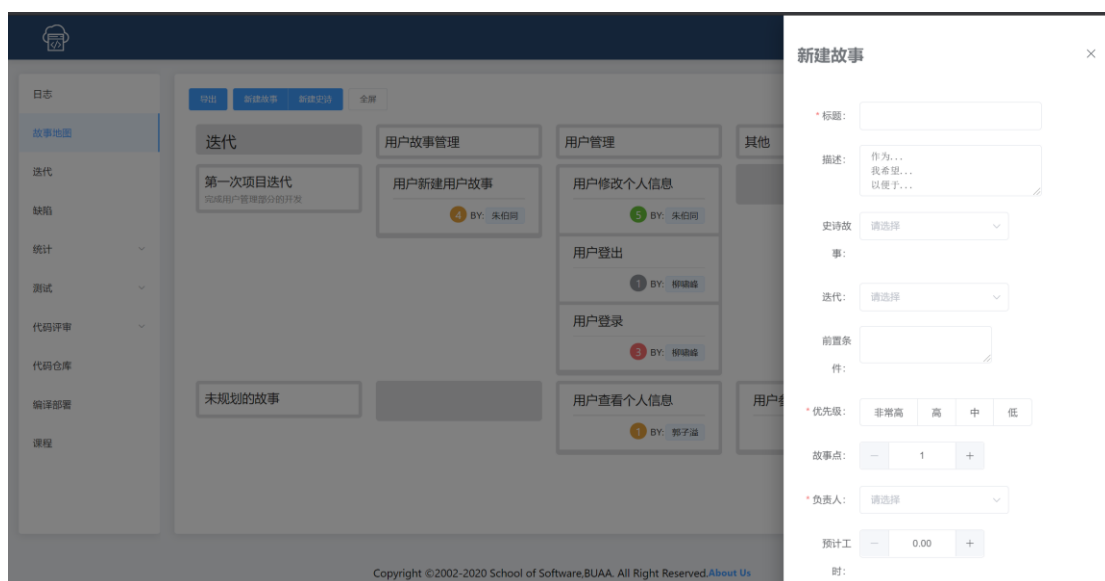


图 2.8.2

点击看板中的故事卡片，将显示故事详情，如下图 2.8.3 所示。团队学生可以对所有故事进行管理。

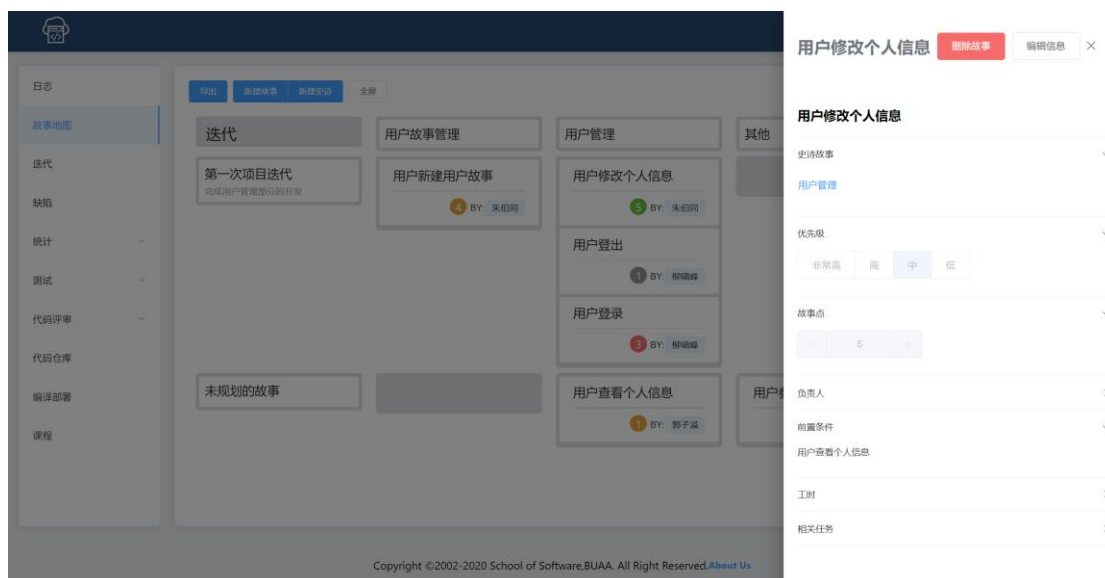


图 2.8.3

点击左上角“导出”按钮，将会导出 excel 格式的故事列表文件。



## 2.9 实训-迭代

点击菜单栏“迭代”，进入迭代列表页面，列表中展示所有迭代及迭代的信息。如下图 2.9.1 所示。

标题	目标	状态	开始时间	结束时间
软件工程综合实训的第一次迭代	for test <a href="#">↗</a>	已结束	2021-03-18 00:00	2021-03-25 00:00
软件工程综合实训的第二次迭代	for test again <a href="#">↗</a>	已结束	2021-03-25 00:00	2021-04-14 00:00

图 2.9.1

点击“目标”一列中的修改按钮，可以修改迭代目标信息，点击保存按钮提交修改。点击迭代标题可以进入迭代内部看板和图表。如下图 2.9.2 所示。



图 2.9.2

故事看板展示迭代中所有的用户故事，分布在所属的状态列中，可以点击新建故事添加用户故事，看板可以通过拖动改变用户故事状态。类似的，点击任务看板展示迭代中的所有任务，分布在所属的状态列中，可以点击新建任务添加任务，看板可以通过拖动改变任务状态。

**注意：故事和任务是有关联的！当任务所属的故事未开始时，将无法提前更改任务状态！当任务所属的故事已结束时，将无法再次更改任务的结束状态！当故事所包含的任务未全完成时，将无法结束该故事！**

如下图 2.9.3 所示。



图 2.9.3

点击右上的列表模式打开列表模式，展示了用户故事和任务的树状关系，蓝色底色的为任务，白色底色的为用户故事。如下图 2.9.4 所示。

第一次项目迭代						看板模式	列表模式	情绪图	燃尽图
标题	描述	状态	优先级	负责人	点数				
用户修改个人信息	用户修改个人信息	Doing	中	朱伯同	3				
用户修改密码后端接口	用户修改密码后端接口	待开发	高	朱伯同	3				
用户编辑个人信息后端接口	用户编辑个人信息后端接口	待开发	高	朱伯同	1				
用户登出	用户登出系统	Doing	低	柳啸峰	1				
用户登出后端接口	用户登出后端接口	开发中	中	柳啸峰	1				
用户登录	用户成功登录系统	Done	非常高	柳啸峰	3				
用户登录后端接口	用户登录后端接口	已完成	非常高	柳啸峰	1				
用户登录前端界面	用户登录前端界面	已完成	低	柳啸峰	2				
用户新建用户故事		Backlog	高	朱伯同	4				

图 2.9.4

点击情绪图查看本次迭代时间段的情绪图，在上方今日心情选择选项即可上传今日心情。也可查看情绪图的表格模式。如下图 2.9.5 两图所示。





图 2.9.5

点击燃尽图可以查看本次迭代的燃尽图。如下图 2.9.6 所示。

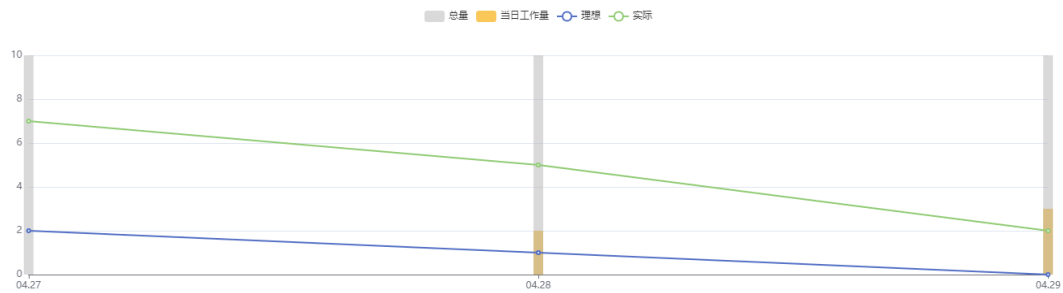


图 2.9.6

## 2.10 实训-缺陷

学生用户可以点击左侧菜单栏中的“缺陷”进入缺陷看板。看板展示项目的所有缺陷，分布在所属的状态列中，可以点击新建缺陷添加缺陷，看板可以通过拖动改变缺陷状态。如下图 2.10.1 所示。



图 2.10.1

点击左上角“新建缺陷”，输入标题、描述、优先级和负责人，可以新建一条缺陷记录。如下图 2.10.2 所示。创建成功的缺陷卡片默认放置在初始状态。

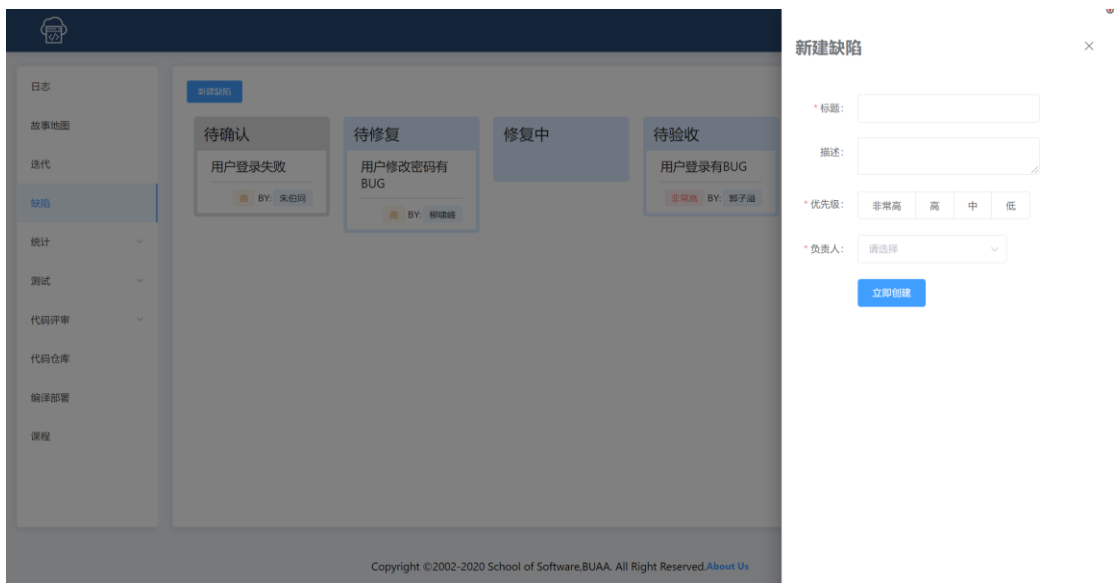


图 2.10.2

点击缺陷看板中的缺陷卡片，可以查看缺陷详情。以便于团队学生对缺陷的管理。如下图 2.10.3 所示。



图 2.10.3

## 2.11 实训-统计

统计这部分将情绪图和燃尽图独立出来，以便更方便查看和监控团队和项目的实时状态。点击左侧统计子菜单中的情绪图可以查看整个项目持续时间的情绪图，点击燃尽图可以查看整个项目持续时间的燃尽图。如下 2.11.1 所示。

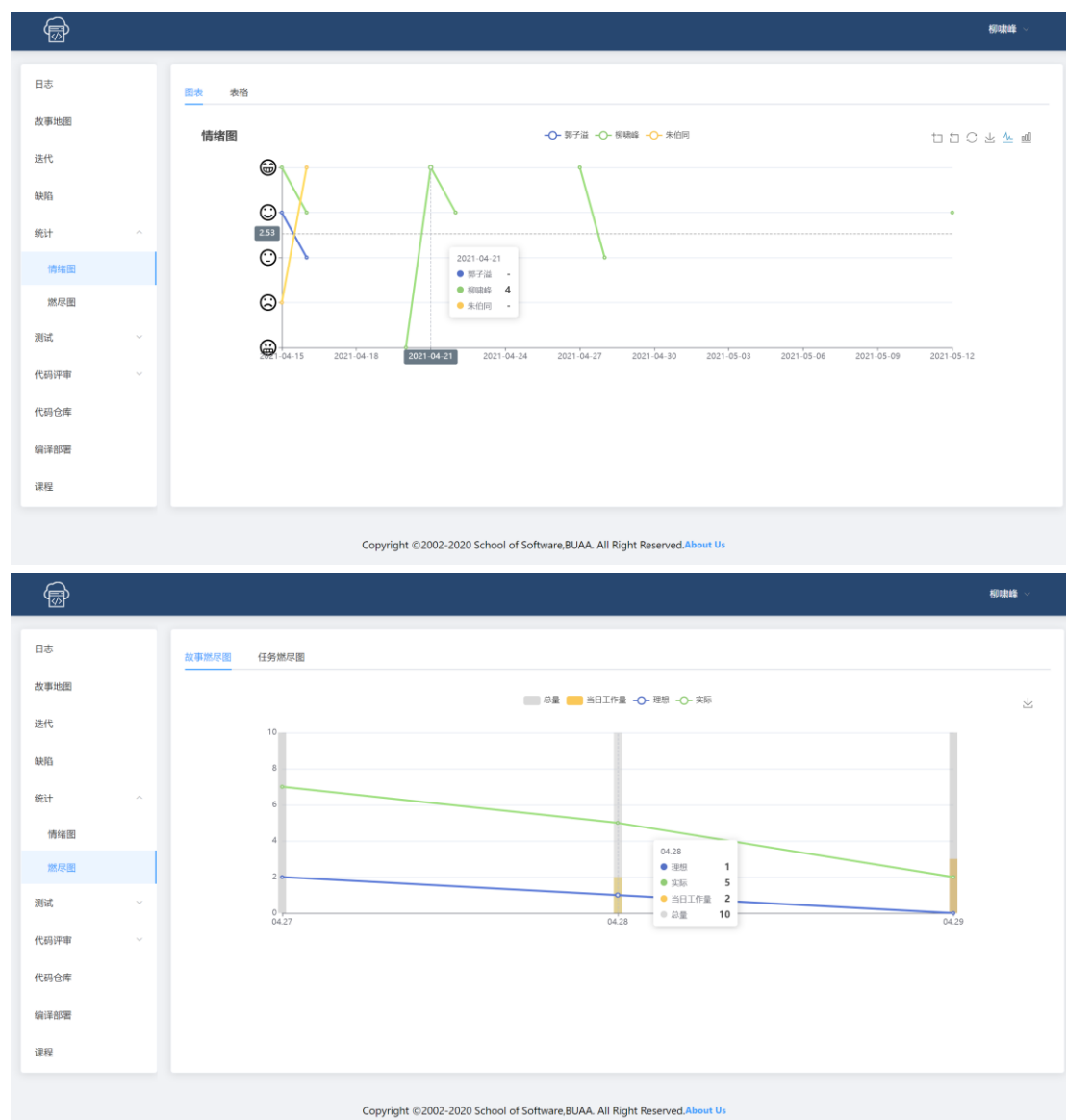


图 2.11.1

## 2.12 实训-测试

### 2.12.1 测试流程

学生用户从左侧菜单栏可以进入测试部分。测试部分共分为两个模块：**组内测试**和**组间互评**。组内测试即为团队内测试，来判断测试用例的通过与否，并提出缺陷。

组间互评共有三个身份：**教师**，**被测团队**（开发团队）和**测试团队**，其中被测团队和测试团队是相对而言的。在一次互测中，同一个团队既是测试团队，可以测试其他团队的项目；同时也是被测团队，自己的项目也会被其他组测评。

组间互评具体流程如下图 2.12.1。

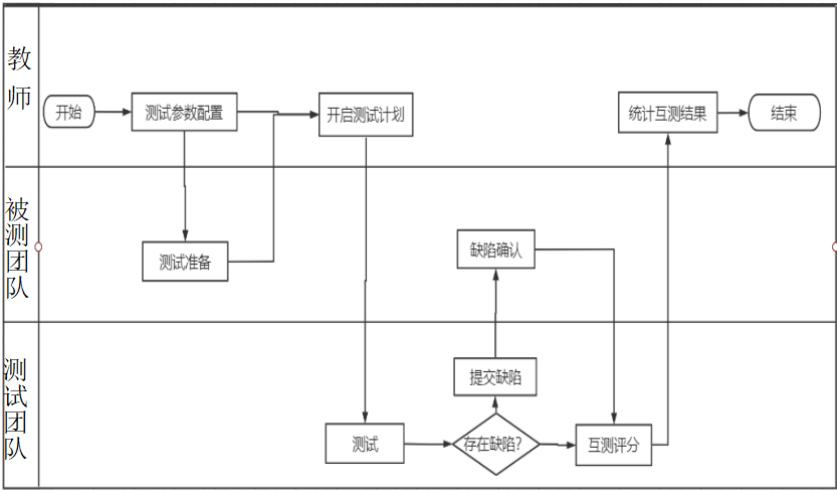


图 2.12.1

### 2.12.2 组内测试-测试用例

学生用户点击左侧导航栏测试->测试用例，即可查看测试用例列表，包括用例名、用例级别、所属故事以及预期等详细信息。如下图 2.12.2 所示。



图 2.12.2

可以通过点击“添加用例按钮”、点击测试用例列表操作“修改”“删除”按钮,实现对测试用例的基本增删改查功能。此处不过多赘述,如下图 2.12.3 所示。

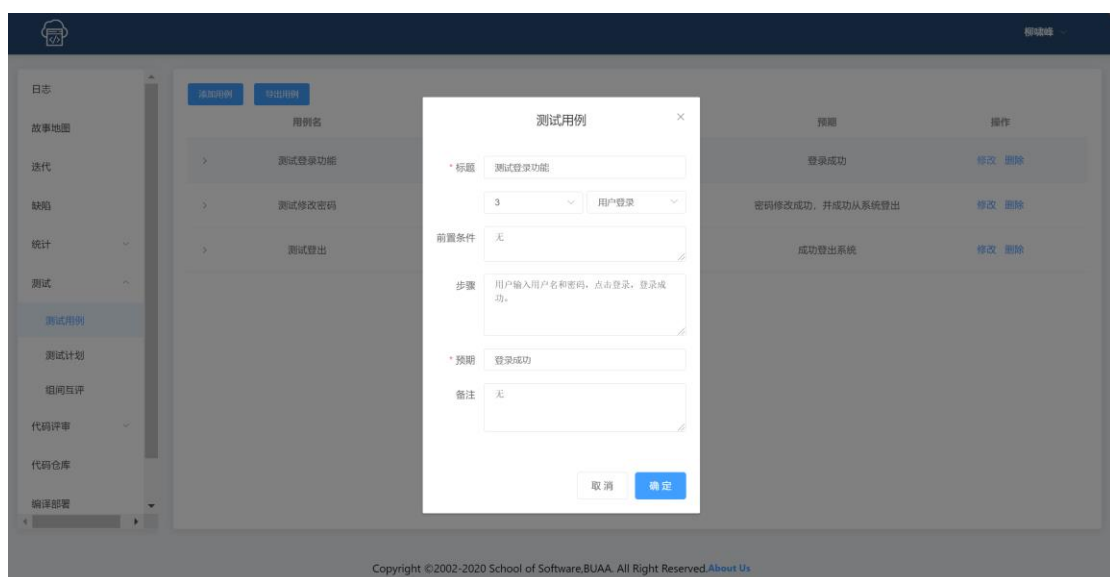


图 2.12.3

### 2.12.3 组间测试-测试计划

点击左侧导航栏测试->测试用例便可查看测试计划列表。其中包括测试计划名称、负责人、状态、起止时间等基本信息。每一个测试计划标题下都通过不同颜色,标记了该计划包含的测试用例的通过情况,鼠标悬停即可查看。并且通过相应的按钮,实现对测试计划的基础增删改查功能。如下图 2.12.4 所示。

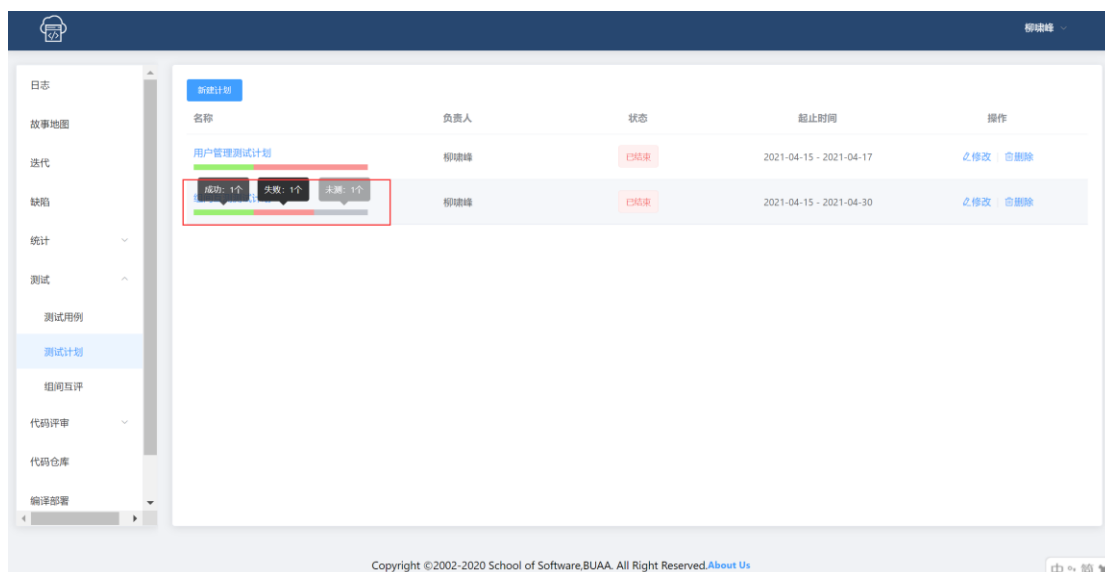


图 2.12.4

点击某一测试计划名称，将进入测试计划详情页面，可以看到该测试计划包含的所有测试用例。其中绿色对号代表该用例测试通过，红色错号代表该测试用例测试不过。如下图 2.12.5 所示。



图 2.12.5

针对某一测试用例，学生可提交测试结果成功/失败并给出说明，可以通过点击“新建缺陷”按钮针对未通过测试用例提出缺陷，并在缺陷看板上生成一条缺陷记录。如图 2.12.6 所示。



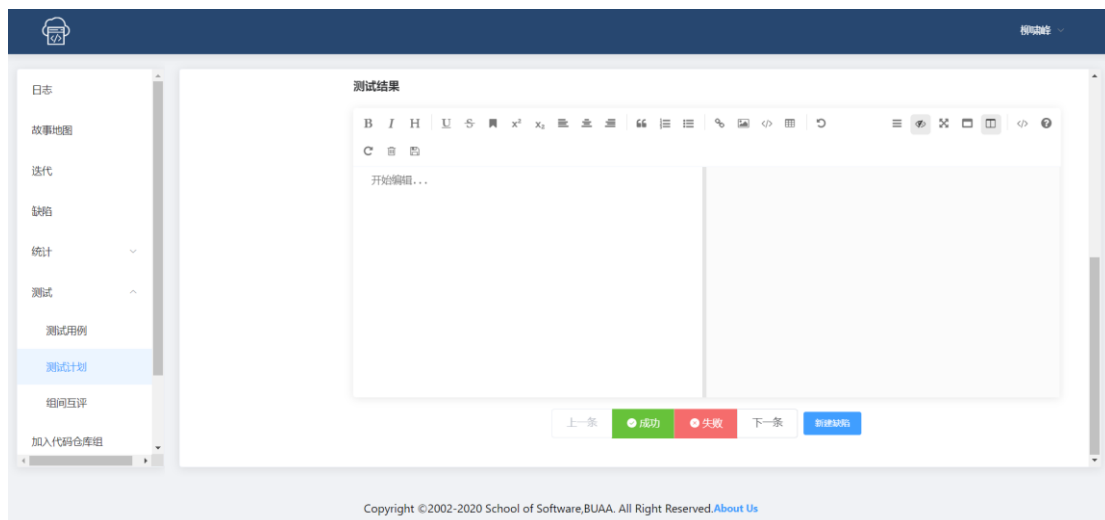


图 2.12.6

## 2.12.4 组间互评

互测开始时，教师会提前创建好互测信息和相关规则和说明，分配好互测团队，此为互测的准备工作。对于学生来说，一次互测将分为两个阶段，分别为提交阶段以及互评阶段。其中，提交界面仅在提交阶段开放，互评和评分界面仅在开放。

被测团队点击导航栏测试->组间互评，可以看到当前互测列表。如图 2.12.7 所示。

标题	互测组数量	状态	提交时间	互测时间	操作
> 互测提交阶段02	3	提交阶段	2021-05-14 00:00 至 2021-05-16 00:00	2021-05-17 00:00 至 2021-05-18 00:00	提交   开始互评   查看评分
> 测试计划04111545	4	互评结束	2021-04-11 00:00 至 2021-04-11 16:00	2021-04-11 16:00 至 2021-05-10 00:00	提交   开始互评   查看评分
> 第一次迭代互测	3	互评结束	2021-03-18 00:00 至 2021-03-21 00:00	2021-03-22 00:00 至 2021-03-30 00:00	提交   开始互评   查看评分
> 测试计划04111545	4	互评结束	2021-04-11 00:00 至 2021-04-11 05:00	2021-04-11 06:00 至 2021-05-10 00:00	提交   开始互评   查看评分
> 互测提交阶段测试	1	提交阶段	2021-05-13 00:00 至 2021-05-16 00:00	2021-05-17 00:00 至 2021-05-18 00:00	提交   开始互评   查看评分
> 第三次迭代互测	3	互评结束	2021-04-01 00:00 至 2021-04-10 00:00	2021-04-11 00:00 至 2021-04-12 00:00	提交   开始互评   查看评分
> 测试互测20210411	4	互评结束	2021-04-11 00:00 至 2021-04-11 04:00	2021-04-11 05:00 至 2021-05-12 12:00	提交   开始互评   查看评分

图 2.12.7

正在提交阶段的互测，被测团队可以点击“提交”按钮，进入提交页面，填写好项目网址，上传用户手册，**选择需要提交的测试计划并给出相关说明**。如图 2.12.8 所示。

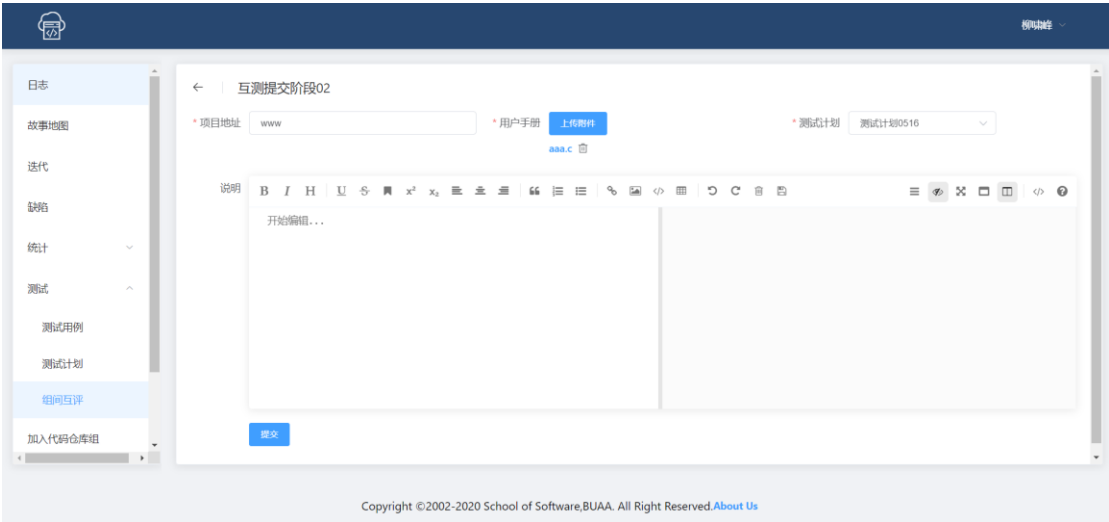


图 2.12.8

待所有团队提交结束后，互测将进入到互评阶段。此时测试团队点击互测列表页（图 2.12.7）中的“开始互评”按钮进入到互评页面。在此界面可查看被测团队所提交的测试计划中的测试用例列表，测试团队需要根据实际情况给出成功/失败结果。

如图 2.12.9 所示。



## 评语

请输入评语

上一条

成功

失败

下一条

新建互测缺陷

图 2.12.9

测试团队也可以针对失败的测试用例可新建互测缺陷。当测试团队为被测团队提出缺陷时，被测团队将会收到一条缺陷确认信息。

被测团队点击互测列表页（图 2.12.7）中的“查看评分”按钮进入到评分结果页面，其中在缺陷界面中，可以看到测试组为本组提出的缺陷，并有权接受/拒绝测试组提出的缺陷，并进行反馈。若被测团队接受了该缺陷，则会在缺陷看板上生成一条记录；否则被测组需要给出拒绝理由，并且缺陷记录将不会生成。如图 2.12.10 所示。

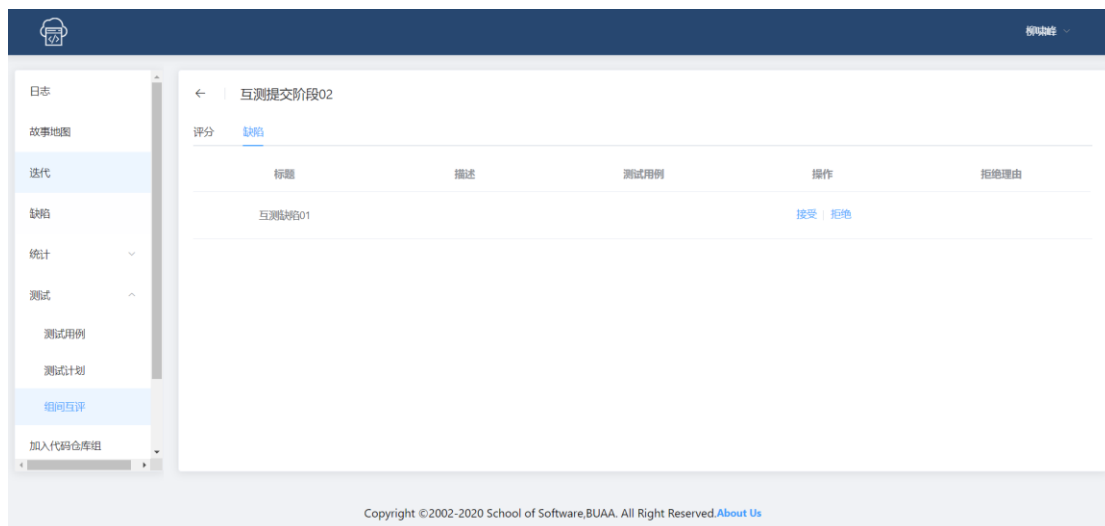


图 2.12.10

被测团队无论对这条缺陷接受与否，测试团队都能够收到一条反馈信息。描述如下。

测试团队可以点击中顶栏小标题“缺陷”进入“我提出的缺陷”界面，可以看到本小组为不同团队测试时提出的缺陷情况，包括缺陷标题、缺陷描述、缺陷状态。

其中，缺陷的状态分为“未确认”、“已拒绝”和“已接受”三种。**值得注意的是，“操作”一列是测试组对被测组这一条确认信息的认同情况，即是否认同被测组所做的选择。**

如图 2.12.11、图 2.12.12 所示。

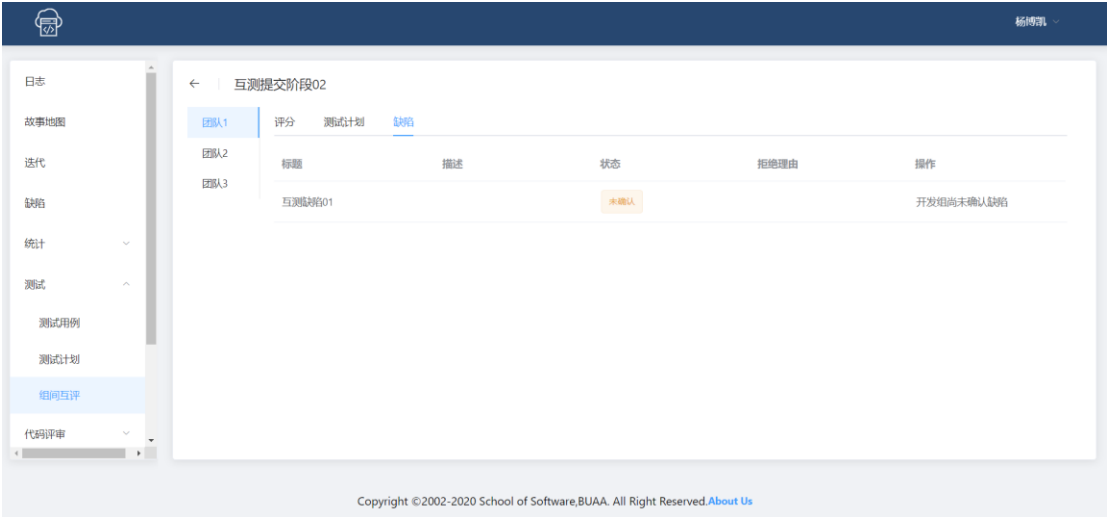


图 2.12.11 被测组未确认状态

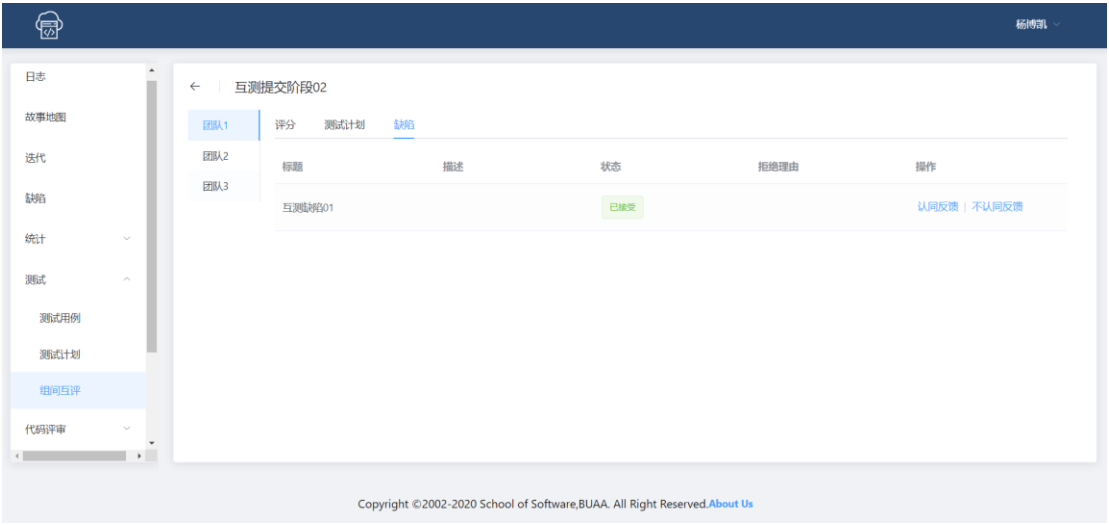


图 2.12.12 被测组确认后的状态

测试组在所有团队所有用例均测试完毕后，在“评分界面”对其进行打分和评语。被测组可以点击互测列表页（图 2.12.7）中的“查看评分”按钮进入到评分结果页面，进行查看。如下图 2.12.13 所示。



图 2.12.13

## 2.13 实训-代码评审&&代码仓库

### 2.13.1 评审准备

学生在进入系统后，如果未创建 Gitlab 账号，会在进入课程首页的时候收到提醒，如下图 2.13.1 所示。



## 课程综合实践

图 2.13.1

进入个人信息页面后需要**首先填写 email，该项为必填项**，随后补全其他信息。在学生填完信息点击“提交”按钮后，个人信息已修改完毕且已经创建 Gitlab 账号。**在教师冻结学生团队之后**，学生即可选择加入 Gitlab 中自己团队所对应的 group，在实训页面的侧边栏中点击“加入代码仓库”即可顺利加入本课程下所在团队的 group，同时侧边栏中也会发生变化，原有的“加入代码仓库”会变为“代码评审”以及“代码仓库”。

### 2.13.2 代码仓库设置

学生进入代码仓库页面后可以创建和查看代码仓库。点击页面左上角的“新建仓库”填写仓库的名称以及描述即可顺利新建仓库。**注意每个团队的仓库最大数目为 4 个。**如图 2.13.2 所示。



图 2.13.2

创建成功后点击仓库的卡片即可进入仓库的 Gitlab 界面，方便进行进一步的操作。如图 2.13.3、2.13.4 所示。

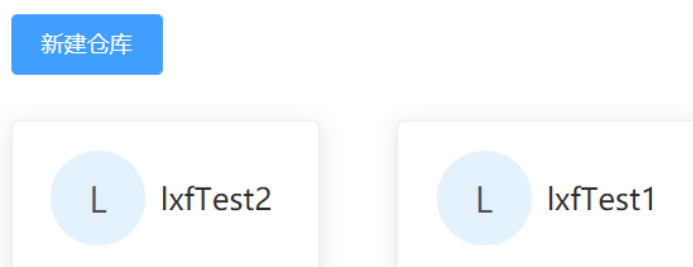


图 2.13.3 仓库列表

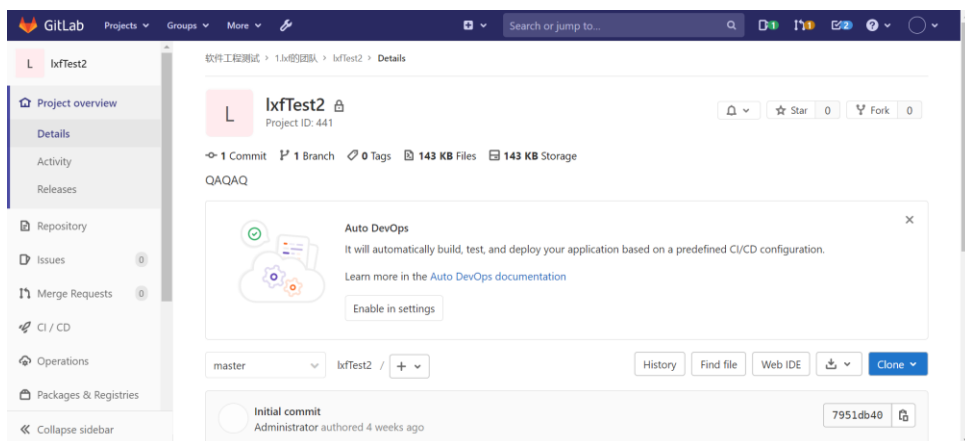


图 2.13.4 Gitlab 界面

### 2.13.3 人工评审

点击实训页面侧边栏“代码评审”下的“人工评审”，即可进入 MR 列表页面。MR 信息包括 MR 名称、所属仓库、状态以及提交时间和关闭时间这几个属性，状态共有三种：开启的，已合并和已关闭。如图 2.13.5 所示。

标题	所属仓库	状态	提交时间	合并时间
Lyh	lxfTest1	已关闭	2021-04-15 15:37	-
Lzc	lxfTest1	开启的	2021-04-15 10:31	-
Update GitProjectController.java	lxfTest1	已合并	2021-04-15 10:17	2021-04-15 10:26

图 2.13.5

点击 MR 列表中的名称即可进入 MR 详细信息页面，该页面中可以看到本次 MR 所有变动文件的 diff 信息以及 commit 信息。如图 2.13.6 所示。

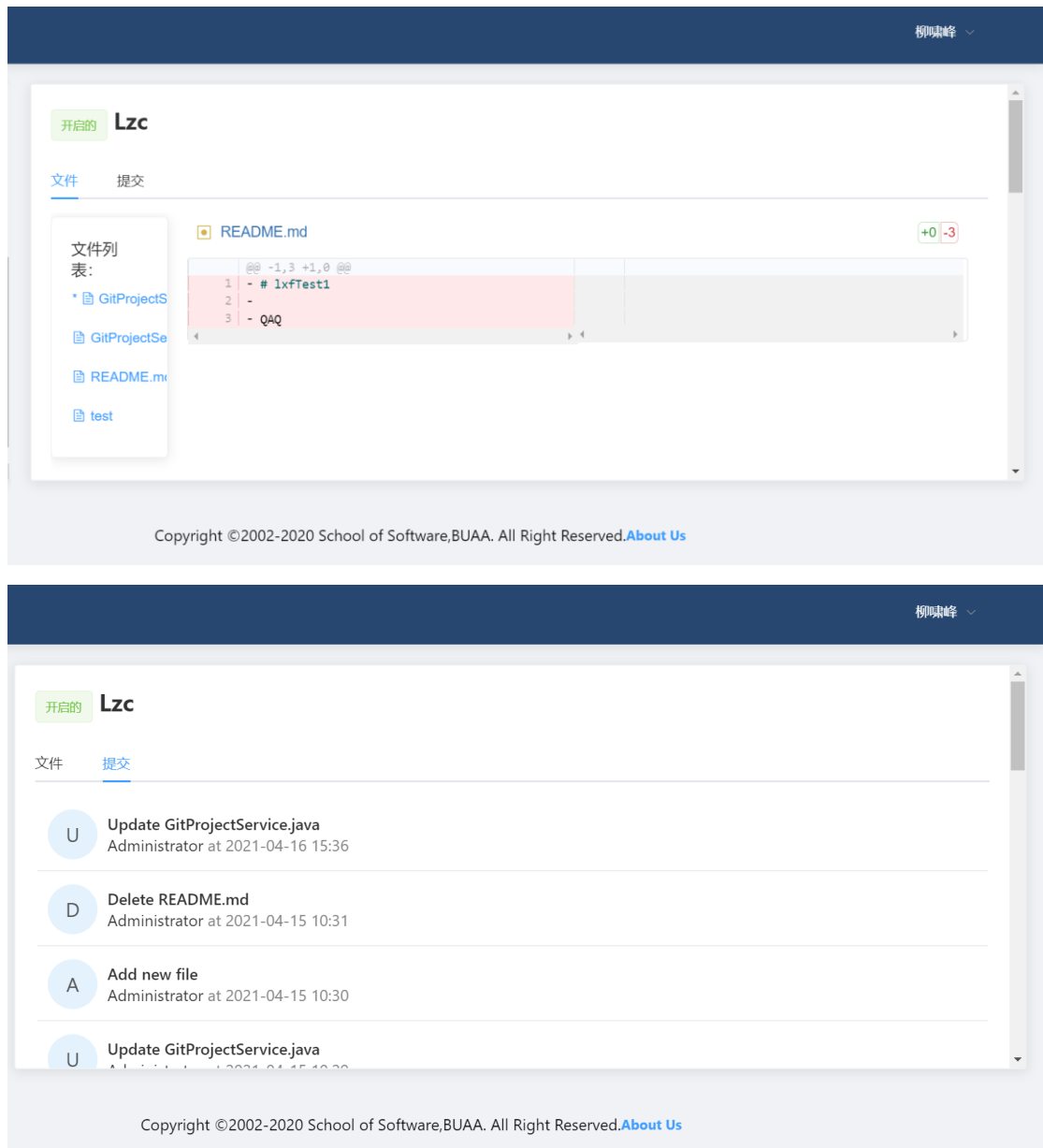


图 2.13.6

页面下方首先是本次 MR 关联的用户故事，每个 MR **最多可关联 3 个**用户故事，评审者可以从上方的 commit 信息中获取；然后是可修改性、可测试性、可模块化等五个维度的具体扩展问题以及对应回答，最后是评审者的评论。如图 2.13.7 所示。



柳啸峰

\* 所属用户故事:

用户修改个人信息 用户登出 用户登录

\* 你修改/新增这些代码时工作量如何?

☐ 实际上修改并未达成目标,基本上白费功夫

☐ 超出了开发计划的时间修改完成,或者修改了预想以外的大量代码

☐ 能够预想到需要变动的代码,在开发计划的时间内完成

☒ 代码变动非常简单,改动操作基本不耗费时间

\* 你对本次代码提交的信心如何?

☐ 手工测了也不能放心提交

Copyright ©2002-2020 School of Software,BUAA. All Right Reserved.[About Us](#)

柳啸峰

☐ 需要明确的测试数据/复现方式,并在本地运行代码/查看日志来确认/调试关键系统行为

☒ 只需要口头的粗略描述,不需运行代码,通过IDE就可定位关键系统行为

\* 若这些代码与复用相关/计划进行复用(不限于本项目范围),你复用代码的方式?

☐ 只是理论上可以复用,实际上不可行/代价太大/不如重写

☐ 通过复制粘贴、修改某些特定点来复用

☒ 通过给使用者口耳相传的方式,能够以调用的形式运用

☐ 看到就会用,随便其他开发者自己琢磨,不太可能用错

\* 评价:

test  
by: 柳啸峰

Copyright ©2002-2020 School of Software,BUAA. All Right Reserved.[About Us](#)

图 2.13.7

如果本次 MR 已经被评审那么进入详细信息后会显示这些评审信息,未被评审的话则可以供评审者进行本次评审。

#### 2.13.4 自动评审

点击侧边栏的“代码工具”即可进入自动评审页面。点击左上方的“添加任务”,选择对应的代码仓库、分支以及检查工具,即可创建新的检查任务, **注意一个仓库的一个分支只能对应一个任务,不可重复创建**。如图 2.13.8 所示。

创建代码审查任务

\* 项目仓库

请选择代码仓库

\* 分支选择

请选择代码仓库

检查工具

请选择

取消

确定

图 2.13.8

点击左上方的“代码工具介绍”，即可查看可选代码工具的相关介绍，学生可根据自己的具体需求选择对应工具。如图 2.13.9 所示。

柳啸峰

代码工具介绍

格式检查	Checkstyle	Checkstyle是一种开发工具，可帮助程序员编写符合编码标准的Java代码。它使检查Java代码的过程自动化，从而使人们不必执行这项无聊（但很重要）的任务。这使其成为想要实施编码标准的项目的理想选择。 官网： <a href="https://checkstyle.sourceforge.io">https://checkstyle.sourceforge.io</a>
Python	Bandit	Bandit是一种旨在查找Python代码中常见安全问题的工具。为此，Bandit处理每个文件，从中构建AST，然后针对AST节点运行适当的插件。完成扫描所有文件后，将生成报告。Bandit最初是在OpenStack安全项目中开发的，后来又归还给PyCQA。 官网： <a href="https://github.com/PyCQA/bandit">https://github.com/PyCQA/bandit</a>
	Duplication	Duplication是Code Climate中检测复制代码的工具，引擎使用较为简单的AST语法树来进行代码对比。

Copyright ©2002-2020 School of Software,BUAA. All Right Reserved.[About Us](#)

图 2.13.9

在点击任务左侧的“开始”按钮即可运行 job，用户也可以点击右侧的“修改任务”修改使用的代码检查工具。点击“删除任务”即刻删除本次任务。任务右侧的状态栏显示本次 job 的运行情况，开始运行后状态按钮会显示为“执行中”，如果执行成功状态按钮会变为“SUCCESS”，失败则会变为“FAILURE”。如图 2.13.10。

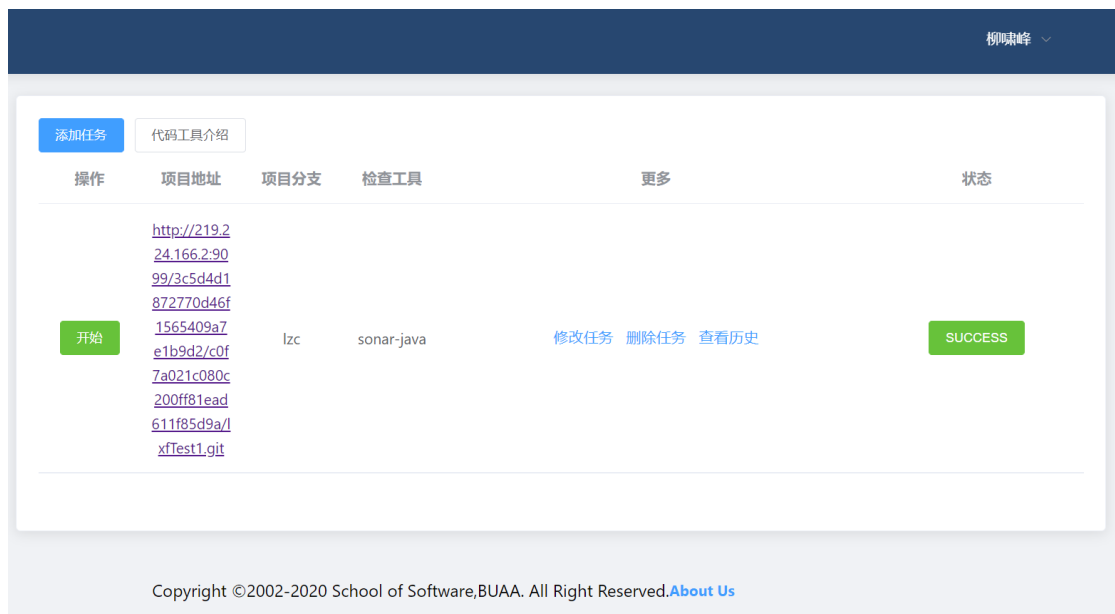


图 2.13.10

点击任务栏右侧的“SUCCESS”状态按钮，即可下载本次检查的结果。检查结果用 html 文件的形式展现。如图 2.13.11 所示。

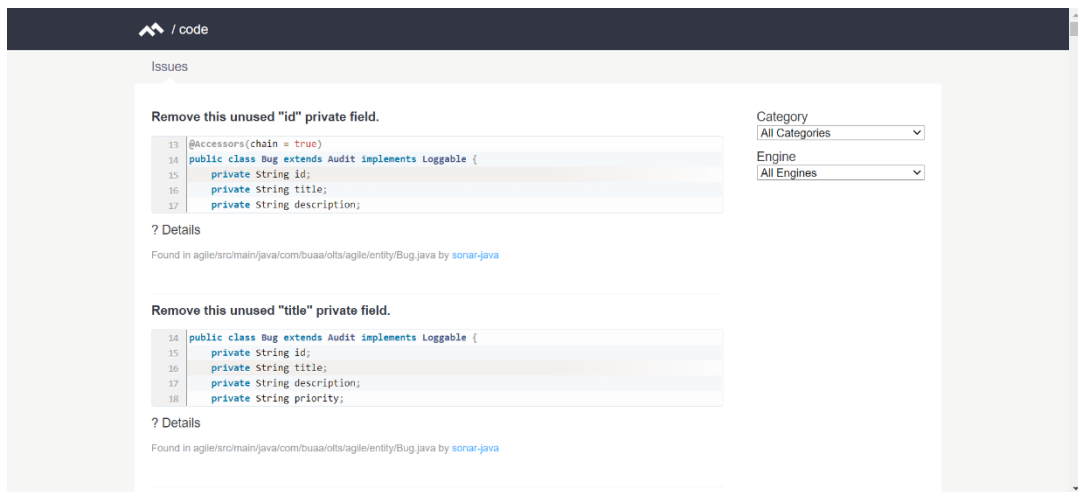


图 2.13.11

点击任务栏右侧的“查看历史”按钮即可查看本次 job 的历史检查记录，包括检查时间以及结果，点击记录即可下载该次记录的检查结果 html 文件。如图 2.13.12 所示。

---

← | a4c41f165ff5d84191b0db12e7d42651lxfTest1lzc

历史记录

2021/4/29下午4:29:42 SUCCESS

2021/4/28上午10:17:13 SUCCESS

2021/4/28上午10:15:08 SUCCESS

图 2.13.12

## 2.14 实训-编译部署

编译部署功能可以帮助学生更加便利地完成软件项目的交付，避免每次进行项目的改动都要进行一次手动编译部署，并提供给教师管理功能，使得授课教师能随时查看学生的项目完成情况。学生可以在平台注册开通代码仓库并完成开发过程中的代码版本控制，在完成一个迭代周期后，可以通过编译部署管理系统创建一个构建部署任务，任务的内容包括代码仓库地址、编译指令和开放端口等信息。创建完毕后即可手动点击触发执行，或者在主分支发生变动时系统自动触发执行。执行完毕后，学生的软件项目即完成了部署，学生可以根据服务器的 IP 地址访问。如果失败了，则可以通过历史记录找到当次的控制台输出，查明原因。

### 2.14.1 凭证管理

点击侧边栏的编译部署按钮，进入编译部署页面。接着点击添加凭证按钮，输入 Gitlab 账号和密码，然后点击确认提交。系统提示成功后即成功绑定了 Gitlab 凭证。如图 2.14.1 所示。

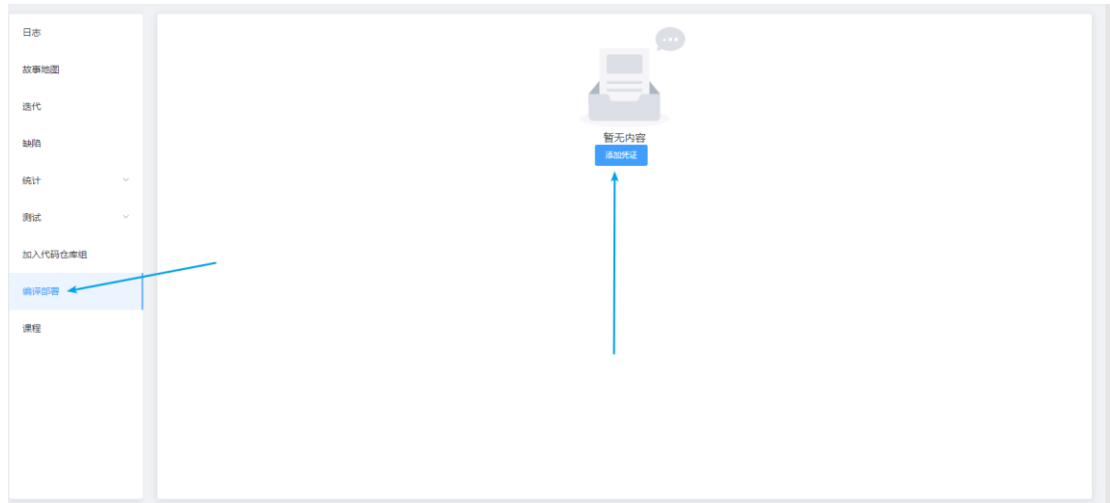
The image shows a dialog box titled 'gitlab凭证' (GitLab Credential) with a close button (X) in the top right corner. Inside the dialog, there are two input fields. The first field is labeled '\* 用户名' (Username) and contains the text 'name'. The second field is labeled '\* 密码' (Password) and contains a series of dots, indicating a masked password. To the right of the password field is an eye icon. At the bottom of the dialog, there are two buttons: '取消' (Cancel) and '确定' (Confirm). The '确定' button is highlighted with a blue background.

图 2.14.1

### 2.14.2 主机管理

点击主机管理进入主机管理页面。点击添加主机按钮，输入云主机的 IP、自己任取的主机名和 SSH 登录用户名、登录密码信息，点击确定提交。如图 2.14.2、2.14.3 所示。



图 2.14.2

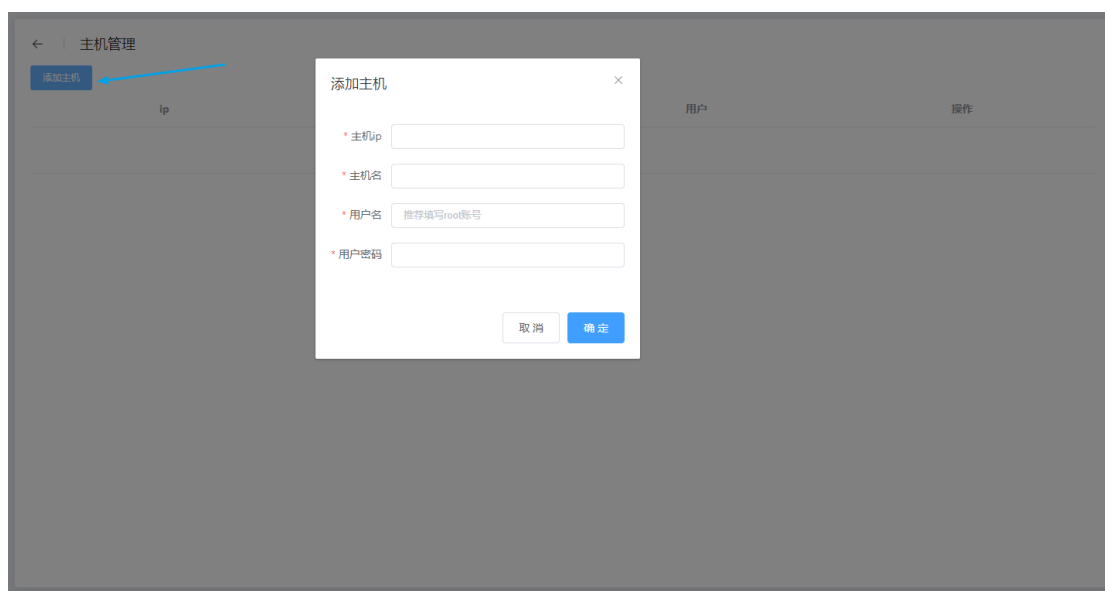


图 2.14.3

### 2.14.3 任务管理

点击添加任务按钮，在弹出框中选择要部署的项目的仓库地址，选择该仓库要部署的分支。接着需要为该任务命名，**注意此名称为任务的唯一标志，多个任务之间不可重名**，之后输入编译指令，指令执行的目录为代码仓库中代码文件的根目录。接着选择要部署的目标主机，以及需要开放的端口。点击新增端口可以增加一个要开放的端口，点击删除可以删除端口。如图 2.14.4 所示为部署一个典型的 React 项目的示例。

\* 项目仓库

lxfTest2: http://219.224.166.2

\* 分支选择

dev

\* 任务名称

frontendproject

\* 部署主机

2c4g: 10.251.254.54

\* 端口0

80

删除

新增端口

取消

确定

图 2.14.4

创建完成后，可以点击开始按钮执行任务，此时最右边的状态会显示执行中；执行完毕会显示相应的结果。如图 2.14.5 所示。

操作	项目名	项目地址	任务名称	更多	状态
<div>开始</div> <div>停止</div>	lxf的团队的项目	<a href="http://219.224.166.2:9099/3c5d4d1872770446f1565409a7e1b3d2/c0f7a021c080c200f81ead6511f85d9a/lxfTest2.g">http://219.224.166.2:9099/3c5d4d1872770446f1565409a7e1b3d2/c0f7a021c080c200f81ead6511f85d9a/lxfTest2.g</a>	frontendproject	<a href="#">修改任务</a> <a href="#">删除任务</a> <a href="#">查看历史</a>	<div>未执行</div>

图 2.14.5

若要修改任务点击修改任务按钮即可，修改弹窗中的信息并点击确认即可提交修改；点击删除任务按钮可以删除当前任务。点击状态显示按钮可以查看上一次执行的控制台输出。如图 2.14.6 所示。





### 2.14.5 后端项目

采用单体服务的后端项目需在项目的根目录放置一个 Dockerfile，下两图

2.14.8 以 Spring Boot 为例。示例地址 <https://gitlab.com/KurusuSaber/spring-boot-demo>

Name	Last commit	Last update
src	hello world	4 months ago
.gitignore	hello world	4 months ago
Dockerfile	Add new file	2 months ago
mvnw	hello world	4 months ago
mvnw.cmd	hello world	4 months ago
pom.xml	hello world	4 months ago

Dockerfile 94 Bytes

1

From openjdk:8

2

add /target/demo-0.0.1-SNAPSHOT.jar app.jar

3

EXPOSE 10000



4

cmd java -jar app.jar

图 2.14.8

### 2.14.6 前端项目

基于 node 的前端项目，如 react 和 vue，需要在项目的根目录放置 Dockerfile 文件和 app.js 文件，下图 2.14.9 是以一个 react 项目为例。参考示例地址 <https://gitlab.com/KurusuSaber/react-demo>

 **Dockerfile**  294 Bytes

```
1  # build stage
2  FROM node:lts-alpine as build-stage
3  WORKDIR /app
4  COPY package*.json ./
5  RUN npm install
6  COPY . .
7  RUN npm run build
8
9  # production stage
10 FROM nginx:stable-alpine as production-stage
11 COPY --from=build-stage /app/build /usr/share/nginx/html
12 EXPOSE 80
13 CMD ["nginx", "-g", "daemon off;"]
```

图 2.14.9