

软件项目开发实习课程指导书

1. 课程愿景

软件项目开发实习课程以企业级项目实战为载体，系统构建学生的全栈工程能力与软件工程思维。课程深度还原真实项目场景，通过团队协同开发机制，使学生在角色轮转（如产品经理、前端开发、后端架构、测试工程师等）、敏捷需求迭代、分布式代码协作及持续集成部署的完整链路中，建立对软件工程全生命周期的立体认知。

（1）以 Git 为核心的版本管理系统作为技术协作基座，不仅承载着分支策略管理、冲突智能解构、代码质量审查、变更历史追溯等工程实践，更通过与 Gitee、GitHub 等企业级平台的对接，让学生提前掌握工业化开发范式——基于代码托管平台的协作开发，可直接转化为就业市场或科研深造中的项目经验证明。

（2）通过企业项目驱动式教学，学生将经历从 0 到 1 的完整开发周期，在解决真实业务问题的过程中，形成"架构设计-技术选型-编码实现-质量保障"的闭环思维。这种以战代练的培养模式，既帮助学生构建可迁移的技术认知框架，又培养其自主解决复杂工程问题的能力，为技术生涯的长期发展筑牢根基，实现从校园到毕业设计甚至到职场的无缝衔接。

2. 课程安排

2.1 理论学习安排

5 月 27 日~6 月 2 日

基于“U+平台”，自选项目，根据班级码加入平台课程，完成视频课程学习。在学习过程中，涉及企业项目开发流程、陌生的知识点（如 Vue、Spring Boot、Spring Cloud、Docker）等，自查资料进行学习，并在小黄本中做好学习笔记。

2.2 团队实战安排

6 月 3 日~6 月 12 日：团队协作完成项目的开发。

要求：

团队人数：人数控制在 6 人以内；同一指导老师的班级允许跨班组队。

关于项目：完成平台所选项目剩余内容，也可自选题目，重新开发。

3. 注意事项

3.1 关于“U+”平台

（1）学生登录的前提是在平台开通账号，如果有学生登录之后，身份显示 U+平台，不显示太原理工大学软件学院，请及时反馈指导老师，指导老师将名单反馈到高程昕老师处，联系平台为同学们开通账号；

（2）学生登录之后，点击【学生工作台】→【加入课程】→【输入班级码】，加入课程中，课程选择如表 3-1 所示。

表 3-1 项目列表

项目名	项目简介	技术栈	班级码
基于数字经济的农产品融销一体化平台	当前，我国农产品交易仍深受传统单边市场模式的束缚，导致农作物的销售状况不尽如人意。这一问题不仅制约了农业的发展，更对广大农户的生计带来了严峻挑战。由于农户的收入普遍偏低，他们的偿债能力也相对较弱，这使得他们在申请农资贷款时，常常面临繁琐的程序和严格的审核标准。为了破解这一困局，急需开发一款数字农产品融资营销一体化平台。这个平台将致力于打破传统模式的束缚，实现数字经济与农业的深度融合。在这个平台上，农户可以便捷地申请农资贷款，减轻经济压力；通过精准营销，帮助农户扩大农产品的销售渠道，提高销售效率。平台同时提供一系列增值服务，如用户间自由交互、专家技术指导、个性化农业发展等。在疫情期间，平台更是能够发挥重要作用，通过线上销售、无接触配送等方式，帮助农户更好地经营农产品，保障他们的生计。通过基于数字经济的农产品融销一体化平台的推广和应用，我们有望为农户带来更加广阔的市场前景和更可观的经济收益，推动农业产业的持续健康发展。	Spring Boot Vue	WK5N05
垂直社交类移动应用-兴趣小队	兴趣小队系统是一个多平台应用系统，包括 Web 端、桌面端和鸿蒙手机 APP 端。用户可以随时随地使用鸿蒙手机进行兴趣活动发起，寻找具有相同或相似兴趣的人群；用户还可以通过兴趣圈分享活动，吸引相同爱好者查看、报名、评论和点赞活动。管理员可以通过 Web 浏览器管理和维护鸿蒙手机 APP 中的用户信息、活动信息和用户评论信息。运维人员可以使用桌面端应用查看服务器的运行状态、数据库连通性和项目操作日志，还可以自定义服务器预警规则，便于快速发现和解决服务器问题，保障系统的安全运行。	HarmonyOS、 Web、Swing	R3NBR4

表 3-1（续）

Music Dreamer 悦享音乐	<p>随着网络媒体的快速发展，出现了各类音乐网站，多以当红歌手作品为主，进驻门槛较高，网络音乐爱好者无法自由分享创作的音乐作品。Music Dreamer 悦享音乐平台为用户带来最佳音乐体验的同时，简化了音乐作品分享流程，使得音乐创作者可以更好更快的分享自己所创作的音乐。</p> <p>平台采用 Spring Cloud Alibaba 微服务技术、Spring Boot 框架技术、VueJS 前端框架开发技术，前台功能包括歌曲播放、歌曲推荐与收藏、歌单推荐与收藏、用户收听历史记录、消息通知等功能；后台功能包括用户管理、歌手管理、歌曲管理等功能，真正实现音乐悦享。</p>	微服务、Spring Cloud、Spring Boot、Vue	69WL1U
--------------------	---	----------------------------------	--------

3.2 关于 gitee 代码托管平台

（1）是什么

Gitee（码云）是由开源中国于 2013 年推出的基于 Git 的代码托管平台，提供稳定的 Git 仓库服务，支持无限公共仓库和一定数量的私有仓库，确保代码的安全存储与版本控制。同时，它还集成了项目管理、代码审查、持续集成/持续部署（CI/CD）等核心功能。团队可以通过任务看板、里程碑跟踪等功能实现高效协作；利用 Pull Request 机制进行代码评审，提升代码质量；借助内置的 Gitee Go 工具，实现自动化构建、测试与部署，加速软件交付周期。堪称中国本土化的“GitHub”。

（2）使用规则（在第二周发布 git 教学视频）

①如图 3-1 所示，安装 Git 应用（如果在 idea 或 VSCode 中集成 git 插件，也可以直接用）

下载git: <https://git-scm.com/downloads>



图 3-1 git 应用安装

②创建 Gitee 仓库：

- A. 浏览器搜索 `https://gitee.com/`，进入 gitee 官网，进行注册和登录。
- B. 组长进入个人首页，点击右上角“+”，创建个人仓库，如图 3-2 所示。



图 3-2 创建仓库入口

- C. 组长填写仓库信息，包含仓库名称（建议为项目名）、URL、仓库介绍、对外权限，如图 3-3 所示。填写时，若只愿小组内成员可见，可将仓库设为私有。



图 3-3 创建仓库详情填写

- D. 创建后，可先关注如图 3-4 所示红框圈住的部分，在本地找到项目所在跟路径，设置好远程仓库信息（方便日后开发时，直接推送和拉取代码）。其中，`git config --global user.name` 和 `git config --global user.email` 为提交时的用户名和 `git` 注册邮箱信息。



图 3-4 本地仓库用户信息设置

- E. 如图 3-5 所示，当远程仓库创建后，需要创建本地仓库和远程仓库同步信息。进入到本地项目（项目名与远程仓库名一致），执行第三行及往后的指令。

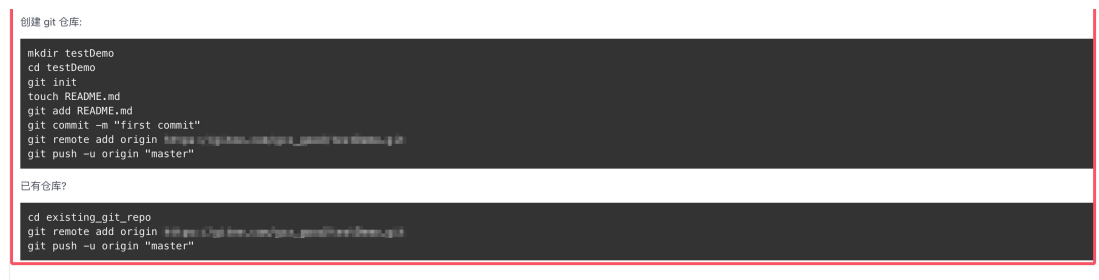


图 3-6 本地仓库关联远程仓库并推送代码

F. 如图 3-7 所示，远程仓库完成同步。



图 3-7 远程内容同步

③管理成员：可以按照如图 3-8 的顺序，添加项目组成员，可以通过链接、直接添加 gitee 用户、通过仓库等方式邀请成员。



图 3-8 成员管理

④创建分支：可以按照如图 3-9 所示的顺序，为项目组成员创建各自的分支。



图 3-9 创建分支

创建分支时，建议基于 master 创建 develop 分支；基于 develop 分支创建项目组成员的开发分支，如图 3-10 所示，具体内容详见视频课。



图 3-10 分支层级建议

(3) 常见指令

①首次下载远程项目

`git clone xxxxxxxxxxxxxxxxx`: 第一次远程登录需要用户名密码, 其中用户名为注册邮箱前缀; 密码为注册密码。

②提交相关:

`git add -A`: 将工作区更新的+删除的变化, 提交到暂存区

`git commit -a -m“记录变化的描述信息”`: 暂存区保存文件提交到本地仓库

`git push`: 本地仓库提交到远程仓库

③拉取代码:

`git pull`: 将远程仓库的代码拉回来

④其他:

`git branch`: 查看当前分支

`git checkout 分支名`: 切换分支到其他分支

`git reset --hard`: 代码回滚到修改提交之前

`git log`: 查看提交记录

`git reset --hard 提交记录字符串`: 代码回滚到某个提交历史节点

`git merge 分支名`

(4) 一般性流程 (以实际需要为主, 具体可看视频课)

① 切换主开发分支 (假设名称是 `develop`), 拉取最新代码

`git checkout develop`

`git pull`

- ② 切换个人开发分支（假设名称是 `feat_gcx`），合并最新代码

```
git checkout feat_gcx
```

```
git branch (这步可有可无，确认一下是否切过来了)
```

```
git merge develop
```

- ③ 若没有冲突，可以直接推到远程仓库

```
git push
```

- ④ 若有冲突，则先把冲突部分删除,再进行提交

```
git add -A
```

```
git commit -a -m "XXX"
```

```
git push
```

- ⑤ 开启劳累的一天

- ⑥ 提交代码，和 4 步骤一模一样。