### 《软件工程课程设计》课程教学大纲

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 英文课程名 | Software Engineering Curriculum Design | | 总 学 时 | | 40 | 学 分 | 2 |
| 课程编码 | G726037 | | 理论教学学时 | |  | 适用专业 | 软件工程 |
| 课程类别（请在课程所属类别栏注明选修或必修） | 通识课程 |  | 实践  教学  学时 | 实验学时 | 40 | 先修课程 | C/C++程序设计，数据结构，软件工程，计算机科学导论，高等数学 |
| 大类基础课程 |  | 上机学时 |  | 开课学院（部） | 计算机科学与技术学院、软件学院 |
| 专业课程 | 必修 | 其它 |  | 基层教学组织 | 软件工程系列课程教学团队 |

一、课程简介

该课程是一门针对软件工程专业的本科生专业必修课，是高等学校软件工程专业教学计划中综合实践教学的核心课程之一。它具有较强的理论性和综合性，不仅需要软件开发过程和管理的理论和方法等知识，还需要问题建模、程序设计、文档撰写等工程实践知识，同时还和实际软件开发相结合。从课程地位上来说，该课程是一门综合运用先修基础课程所学知识解决实际问题的重要课程。综合运用学生在前面掌握的专业基础知识，利用开放资源进行应用系统的设计及开发，从一个系统性的综合案例设计及实现过程，锻炼学生的创新及开发实践能力。

# 二、教学目标

2.1 课程教学目标掌握软件需求分析和设计的方法，以及UML建模技术，能够在软件工程过程中使用常用建模工具对复杂软件系统进行需求分析、模块分解和设计，并掌握相关文档的编写方法。

1. 培养学生掌握应用系统的基本开发流程和相关技术，深入理解面向对象程序设计、开发过程的管理和质量控制等知识，加深对数学建模的认识和理解，提升计算思维能力。
2. 基于教师给定或自设的课题，锻炼学生充分利用互联网等其他开放式资源，能够独立完成系统的分析与设计，实现创新能力的实践和培养。

2.2 课程目标与毕业要求（指标点）对应关系

该课程支撑以下毕业要求和具体细分指标点：

【毕业要求3】设计/开发解决方案：能够设计针对复杂软件工程问题的解决方案，设计满足特定需求的系统、单元（部件）或工艺流程，并能够在设计环节中体现创新意识，考虑社会、健康、安全、法律、文化以及环境等因素。

支撑指标点3.5：具备创新意识，能够在软件设计中发现创新点，并掌握基本的创新方法。

【毕业要求10】沟通：能够就复杂软件工程问题与业界同行及社会公众进行有效沟通和交流，包括撰写报告和设计文稿、陈述发言、清晰表达或回应指令。并具备一定的国际视野，能够在跨文化背景下进行沟通和交流。

支撑指标点10.3：具有良好的沟通能力，能够就复杂软件工程问题清晰表达见解、陈述发言，并与业界同行及社会公众进行有效交流。

【毕业要求11】项目管理：理解并掌握工程管理原理与经济决策方法，并能在多学科环境中应用。

支撑指标点11.3：具有在多学科环境中应用工程管理原理与经济决策方法管理复杂软件工程项目的能力。

【毕业要求12】终身学习：具有自主学习和终身学习的意识，有不断学习和适应发展的能力。

支撑指标点12.1：能够自主进行文献检索和资料查询，及时获取和跟踪计算机领域的前沿技术和最新进展。

本课程目标与毕业要求（指标点）的对应关系如表1所示。

表1 课程目标与毕业要求（指标点）的对应关系

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 课程目标 | 毕业要求指标点 | 教学环节 | | | | |
| 课堂授课 | 实验 | 作业 | 课堂测验 | 课堂讨论 |
| **目标1**：掌握软件需求分析和设计的方法，以及UML建模技术，能够在软件工程过程中使用常用建模工具对复杂软件系统进行需求分析、模块分解和设计，并掌握相关文档的编写方法。 | 指标点11.3：具有在多学科环境中应用工程管理原理与经济决策方法管理复杂软件工程项目的能力。  指标点12.1：能够自主进行文献检索和资料查询，及时获取和跟踪计算机领域的前沿技术和最新进展。 |  | √ |  |  | √ |
| **目标2**：培养学生掌握应用系统的基本开发流程和相关技术，深入理解面向对象程序设计、开发过程的管理和质量控制等知识，加深对数学建模的认识和理解，提升计算思维能力。 | 指标点10.3：具有良好的沟通能力，能够就复杂软件工程问题清晰表达见解、陈述发言，并与业界同行及社会公众进行有效交流。  指标点11.3：具有在多学科环境中应用工程管理原理与经济决策方法管理复杂软件工程项目的能力。 |  | √ |  |  | √ |
| **目标3**：基于教师给定或自设的课题，锻炼学生充分利用互联网等其他开放式资源，能够独立完成系统的分析与设计，实现创新能力的实践和培养。 | 指标点3.5：具备创新意识，能够在软件设计中发现创新点，并掌握基本的创新方法。  指标点10.3：具有良好的沟通能力，能够就复杂软件工程问题清晰表达见解、陈述发言，并与业界同行及社会公众进行有效交流。 |  | √ |  |  | √ |

# 三、课程教学内容及学时分配

## 1．实践教学安排

3.1理论教学安排

本课程不安排理论教学，所有课时均用于实践教学。

3.2实践教学安排

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 序号 | 项目名称 | 学时 | 类型 | 每组人数 | 教学要求  (应明确教学重点、难点和教学方法) | 学生任务 | |
| 作业要求 | 其他要求(自学/讨论） |
| 1 | 创新实践选题及方案论证 | 6 | 设计 | 3-5人 | 教学重点：  教师给定部分题目，鼓励学生自主立题。  教学难点：  指标点3.5、10.3、11.3、12.1  教学方法：讲授 + 演示教学 + 讨论 + 任务驱动式教学。 | 完成相关系统的分析、设计与实现，撰写实验报告。 | 自学：  熟悉所选用的建模工具、开发平台，自学其相关资料；  根据自己应用系统的具体需求，进行系统的分析与设计，并进行系统开发、测试等工作。  讨论：分析、讨论和构思团队的系统需求、各自的分工及其设计思路。 |
| 2 | 选题分组及编程环境设置 | 10 | 设计 | 3-5人 | 教学重点：  学生自主组队，方案讨论及编程环境设置。任务分解。  教学难点：  指标点3.5、10.3、11.3、12.1  教学方法：讲授 + 演示教学 + 讨论 + 任务驱动式教学。 |
| 3 | 创新实践选题及方案论证 |  | 设计 | 3-5人 | 教学重点：  完成分组和选题过程，开展项目研究及编程。  教学难点：  指标点3.5、10.3、11.3、12.1  教学方法：讲授 + 演示教学 + 讨论 + 任务驱动式教学。 |
| 4 | 选题分组及编程环境设置 |  | 设计 | 3-5人 | 教学重点：  学生自主组队，方案讨论及编程环境设置。任务分解。  教学难点：  指标点3.5、10.3、11.3、12.1  教学方法：讲授 + 演示教学 + 讨论 + 任务驱动式教学。 |
| 5 | 阶段进展检查及指导 |  | 设计 | 3-5人 | 教学重点：  检查阶段进度并指导学生持续开展项目研究及代码开发。教师按项目参与程度对每人给出阶段分。  教学难点：  指标点3.5、10.3、11.3、12.1  教学方法：讲授 + 演示教学 + 讨论 + 任务驱动式教学。 |
| 6 | 自主编程 |  | 设计 | 3-5人 | 教学重点：  分组开展程序设计与开发，自主讨论项目方案。  教学难点：  指标点3.5、10.3、11.3、12.1  教学方法：讲授 + 演示教学 + 讨论 + 任务驱动式教学。 |
| 7 | 阶段进展检查及指导 |  | 设计 | 3-5人 | 教学重点：  检查阶段进度并指导学生持续开展项目研究及代码开发。教师按项目参与程度对每人给出阶段分。  教学难点：  指标点3.5、10.3、11.3、12.1  教学方法：讲授 + 演示教学 + 讨论 + 任务驱动式教学。 |
| 8 | 自主编程 |  | 设计 | 3-5人 | 教学重点：  分组开展程序设计与开发，自主讨论项目方案。  教学难点：  指标点3.5、10.3、11.3、12.1  教学方法：讲授 + 演示教学 + 讨论 + 任务驱动式教学。 |
| 9 | 阶段进展检查及指导 |  | 设计 | 3-5人 | 教学重点：  检查阶段进度并指导学生持续开展项目研究及代码开发。教师按项目参与程度对每人给出阶段分。  教学难点：  指标点3.5、10.3、11.3、12.1  教学方法：讲授 + 演示教学 + 讨论 + 任务驱动式教学。 |
| 10 | 自主编程 |  | 设计 | 3-5人 | 教学重点：  分组开展程序设计与开发，自主讨论项目方案。  教学难点：  指标点3.5、10.3、11.3、12.1  教学方法：讲授 + 演示教学 + 讨论 + 任务驱动式教学。 |
| 11 | 阶段进展检查及指导 |  | 设计 | 3-5人 | 教学重点：  检查阶段进度并指导学生持续开展项目研究及代码开发。教师按项目参与程度对每人给出阶段分。  教学难点：  指标点3.5、10.3、11.3、12.1  教学方法：讲授 + 演示教学 + 讨论 + 任务驱动式教学。 |
| 12 | 自主编程 |  | 设计 | 3-5人 | 教学重点：  分组开展程序设计与开发，自主讨论项目方案。  教学难点：  指标点3.5、10.3、11.3、12.1  教学方法：讲授 + 演示教学 + 讨论 + 任务驱动式教学。 |
| 13 | 阶段进展检查及指导 |  | 设计 | 3-5人 | 教学重点：  检查阶段进度并指导学生持续开展项目研究及代码开发。教师按项目参与程度对每人给出阶段分。  教学难点：  指标点3.5、10.3、11.3、12.1  教学方法：讲授 + 演示教学 + 讨论 + 任务驱动式教学。 |
| 14 | 自主编程 |  | 设计 | 3-5人 | 教学重点：  分组开展程序设计与开发，自主完成项目工作，撰写结题报告及答辩ppt。  教学难点：  指标点3.5、10.3、11.3、12.1  教学方法：讲授 + 演示教学 + 讨论 + 任务驱动式教学。 |
| 15 | 项目结题答辩 |  | 设计 | 3-5人 | 教学重点：  分组选择代表结题答辩，提交程序源码及实验报告。  教学难点：  指标点3.5、10.3、11.3、12.1  教学方法：讲授 + 演示教学 + 讨论 + 任务驱动式教学。 |

四、考核方式及成绩评定方式

考试方式为考查，参考实验教学平时、系统实现及创新实践报告和结题答辩进行最终考核和成绩评定。成绩采用五分制：优秀、良好、中等、及格和不及格，其中平时成绩占10%，创新实践报告及结题答辩占40%，系统实现占50%。创新实践报告内容包括：1. 选题意义及研究现状 2. 课题研究内容及分析 3. 系统实现方法设计、系统实现及分析 4. 系统运行截图 5.核心模块源代码 6.参考文献。

# 五、教材、课程网址及参考书目

# 无指定教材，选题后，由教师指导查阅相关公开资源及专业文献。

**执笔者：董天阳，王婷**

**审核者：田贤忠**

**课程教学团队成员：江颉，董天阳，王婷，汤颖，张繁**