《程序设计与软件工程实践》教学大纲

课程代码：TELE2136

课程名称：程序设计与软件工程实践 大纲执笔人：徐清源

英文名称：Programming and Software Engineering Practice审批人：

课程学时（周数）：54(14周) 实践学时（周数）：

课程学分：3

适用专业：通信工程

**一、课程性质和教学目标**

**课程性质**：本课程是通信工程专业一门重要的专业课，是通信工程专业必修课。本课程采用教师知识讲授为主，实践为辅的教学模式，旨在让学生掌握软件工程的概念和应用。软件已经成为电子信息技术和互联网领域的灵魂，传统通信工程专业教育造成学生的软件开发能力与培养目标之间有很大的差距。在学生具备了基本的编程、计算机及网络知识后，本课程旨在培养他们的程序设计技能，为未来的工程实践中硬件向软件转变的趋势做好准备。

**课程目标**：修完本课程，学生应能具备如下能力。

1. 能够利用软件工程的分析和设计方法，分析具体工程的需求，并转换为相应模块的设计。
2. 能独立完成小规模软件设计工作
3. 能利用网络资源和开源软件,辅助软件设计，能够利用各种现代设计开发工具。
4. 能够组成有效团队，合理分工，参与团队软件开发。

**二、课程目标与毕业要求的对应关系**

|  |  |  |
| --- | --- | --- |
| 毕业要求 | 指标点 | 课程目标 |
| 1、工程知识 | 1-2掌握计算机软硬件基础知识，具备对工程问题进行软硬件分析与设计的基本能力； | 课程目标1 |
| 3、设计/开发解决方案 | 3-2能适当考虑社会、健康、安全、法律、文化及环境因素，根据设计目标进行需求分析，设计解决方案。 | 课程目标2 |
| 5、使用现代工具 | 5-1能合理使用现代信息技术工具； | 课程目标3 |
| 9、个人和团队 | 9-2能胜任团队成员的角色与责任，组织团队成员开展工作，完成团队分配的工作。 | 课程目标4 |

**三、课程教学内容及学时分配**（重点内容：★；难点内容：Δ）

1. **概述（6学时）（支撑课程目标1）**
	1. 软件与硬件
	2. 软件危机与软件工程
	3. 软件生命周期★
	4. 软件过程模型★
* **目标及要求：**
1. 通过概述的介绍，使得学生了解软件类型和软件工程基本概念
2. 理解软件的完整生命周期，以及了解各种过程模型；
* **作业内容：**

强化对软件生命周期认识，各种过程模型以及应用场合。

* **讨论内容：**

常用软件或者工具的软件类型，个人软件编程方面的体验和困惑。

* **自学拓展：**

小组确定，讨论实践工程项目选题。

1. **可行性分析（3学时）（支撑课程目标1）**
	1. 可行性分析任务
	2. 可行性分析过程
	3. 成本效益分析★
* **目标及要求：**
1. 了解可行性分析的重要性，具体步骤和内容
* **作业内容：**

计算利率对投入产出的影响，进行成本效益分析。

* **讨论内容：**

项目中需要考虑的具体问题：技术、社会、环境以及效益等。

* **自学拓展：**

考虑实践工程项目的可行性。

1. **需求分析（9学时）（支撑课程目标1，2，3，5）**
	1. 需求分析任务
	2. 需求获取方法
	3. 数据流图与数据字典★Δ
	4. 状态转移图★Δ
	5. 其他图形工具
	6. 验证软件需求
* **目标及要求：**
1. 理解需求分析的重要性，方法和内容
2. 掌握需求分析的表达方法和工具
3. 掌握数据流图以及状态转移图的使用方法
4. 了解其他需求分析表达工具
* **作业内容：**

加强数据流图的使用能力，状态转移图的使用方法。以及使用工具发现需求分析中存在问题等。

* **讨论内容：**

需求分析如何提出，如何根据实际项目从不同渠道获取需求。数据流绘制过程中常出现的问题。如何从用户需求描述中提炼状态和状态转移。

* **自学拓展：**

分析项目的不同相关人的不同需求。并使用工具表达。

1. **总体设计（6学时）（支撑课程目标1，2，3）**
	1. 设计过程
	2. 设计原理和设计规则★
	3. 软件结构
	4. 软件结构绘制工具：层次图和HIPO图★
	5. 面向数据流设计方法★Δ
* **目标及要求：**
1. 理解总体设计原则以及应用方法
2. 了解软件结构类型
3. 掌握软件结构绘制工具使用方法
4. 掌握数据流设计方法
* **作业内容：**

加强对设计原理的理解，面向数据流方法的使用，软件结构表达的方法等。

* **讨论内容：**

如何利用用户需求进行软件设计，如何提出不同设计方案，不同设计方案的分析对比。

* **自学拓展：**

根据项目的用户需求提出不同解决方案，并做对比分析，使用合适工具表达。

1. **详细设计（6学时）（支撑课程目标1，2，3，4，5）**
	1. 结构化设计方法
	2. 人机界面设计
	3. 过程设计工具：程序流程图，判定表，判定树等
	4. 程序复杂度度量
* **目标及要求：**
1. 掌握结构化设计方法★Δ
2. 掌握程序流程图的使用★Δ
3. 掌握判定表和判断树的使用方法
4. 了解程序复杂度的度量
* **作业内容：**

加强对结构化设计方法的理解，熟练绘制程序流程图以及判定表的使用等。

* **讨论内容：**

如何将非结构化设计转换为结构化设计。如何保证程序流程图的结构化。如何将判定表和判断树转换成程序流程图。。

* **自学拓展：**

根据项目的解决方案，进行小组分工，并完成部分模块的详细设计，使用合适工具表达。

1. **编码和测试（6学时）（支撑课程目标1，2，3）**
	1. 文档化编程风格
	2. 程序设计语言
	3. 软件测试基础：单元测试、集成测试、确认测试
	4. 黑盒测试★Δ
	5. 白盒测试★Δ
	6. 调试
* **目标及要求：**
1. 了解文档化编程风格
2. 掌握覆盖的概念
3. 掌握等价类划分和边界值分析方法
4. 了解调试过程和途径
* **作业内容：**

加强对各种覆盖的理解，并据此设计测试用例等。

* **讨论内容：**

讨论黑盒测试和白盒测试之间的关系，以及是否可互相取代。良好的编码风格能够带来什么好处。

* **自学拓展：**

针对项目中的模块，提出测试用例。阅读开源代码了解编码风格。

1. **软件项目管理（6学时）（支撑课程目标3，4，5）**
	1. 估算软件规模
	2. 工作量估计
	3. 进度计划安排：甘特图和工程网络★Δ
	4. 人员组织
	5. 软件配置管理
* **目标及要求：**
1. 了解软件规模与工作量关系
2. 能够在有限资源情况下合理安排进度计划
3. 了解软件配置管理方法
* **作业内容：**

通过使用甘特图和工程网络图加深理解资源对进度的影响。

* **讨论内容：**

讨论不同资源情况下对进度安排的影响，以及如何优化进度安排，发现关键路径。

* **自学拓展：**

针对项目中的模块，提出项目进度计划。

1. **面向对象方法学（3学时）（支撑课程目标1）**
	1. 面向对象方法学概述
	2. 对象概念：面向对象分析、设计与实现
	3. UML工具
* **目标及要求：**
1. 了解面向对象方法
2. 了解UML工具
* **作业内容：**

通过使用UML工具描述对象。

* **讨论内容：**

讨论结构化设计和面向对象设计的不同。

* **自学拓展：**

将项目中用户需求使用面向对象方法描述。

**四、考核和成绩评定**

1.考核环节

课程开始后以分组形式进行项目设计活动，加强个人设计能力和团队合作能力培养。考核以设计为主，考试为辅的形式。

各部分比例为：平时10%，课程报告50%，考试40%。

其中课程报告包括：项目需求分析报告15%，项目设计测试报告35%

小组同学合作完成项目设计，成员分工明确。每个同学需独立提交课程报告，报告中需要指出个人在团队中的角色，完成的任务模块在系统中的位置，和其他成员模块的关系等。

2.课程目标达成与考核成绩比例

|  |  |  |
| --- | --- | --- |
| 教学目标 | 毕业要求指标点 | 成绩比例 % |
| 平时成绩 | 期中成绩 | 期末成绩40% |
| 作业 | 项目需求分析报告 | 总体设计报告 | 详细设计与测试报告 | 期末考试 |
| 35教学目标1 | 工程知识1.2  | 6 |  | 6 | 8 | 15 |
| 29教学目标2 | 设计/开发解决方案3.2  |  | 14 |  |  | 15 |
| 21教学目标3 | 使用现代工具5.1  | 4 |  | 8 | 4 | 5 |
| 15教学目标4 | 个人和团队9.2 |  | 2 | 4 | 4 | 5 |

3.评价标准

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 分数构成 | 评价内容 | 指标点 | 满分 | 　 |
| 4分 | 3分 | 2分 | 1分 | 0分 |
| 项目需求分析报告16% | 文献数及综述 | 3.2 | 1 | 　 | 　 | 　 | 有 | 无 |
| 系统功能要求列表和应用背景关系 | 3.2 | 4 | 100%符合应用要求、功能具体、可测试 | 80%符合应用要求 | 50%符合应用要求 | 20%符合应用要求 | 无功能列表 |
| 列出角色数 | 9.2 | 2 | 　 | 　 | >2个 | 2个 | 0~1个 |
| 列出性能要求 | 3.2 | 1 | 　 | 　 | 　 | 有 | 无 |
| 用户角度功能模型 | 3.2 | 4 | 内容一致层次分明符号规范 | 内容一致有层次符号规范 | 内容基本一致有层次符号基本规范 | 内容不全面无层次符号不规范 | 无模型 |
| 数据模型 | 3.2 | 4 | 内容一致层次分明符号规范 | 内容一致有层次符号规范 | 内容基本一致有层次符号基本规范 | 内容不全面无层次符号不规范 | 无模型 |
| 总体设计报告18% | 设计者角度功能模型 | 5.1 | 2 | 　 | 　 | 设计者角度符号规范 | 不是设计者角度符号部分不规范 | 无模型 |
| 数据模型 | 5.1 | 2 | 　 | 　 | 设计者角度符号规范 | 不是设计者角度符号部分不规范 | 无模型 |
| 软件结构 | 5.1 | 4 | 内容一致结构演化过程正确 | 内容一致结构演化过程基本正确 | 内容基本一致有结构演化过程 | 内容基本一致无结构演化过程 | 无模型 |
| 软件结构优化 | 1.2 | 2 | 　 | 　 | 有优化引用原理正确 | 有优化 | 无优化 |
| 模块接口定义数 | 1.2 | 4 | 100%且无错误 | 100% | >50% | >20% | 无 |
| 个人负责模块定义 | 9.2 | 4 | 100%且无错误 | 100% | >50% | >20% | 无 |
| 详细设计与测试报告16% | 模块流程图 | 5.1 | 4 | 100%且无错误 | 100% | >50% | >20% | 无 |
| 模块测试用例 | 1.2 | 4 | 100%且无错误 | 100% | >50% | >20% | 无 |
| 界面或接口协议设计 | 1.2 | 4 | 100%且无错误 | 100% | >50% | >20% | 无 |
| 多模块集成测试用例 | 9.2 | 4 | 100%且无错误 | 100% | >50% | >20% | 无 |

**七、教学教材、指导书及主要参考书**

教材：软件工程导论（第6版） 张海潘 牟永敏 清华大学出版社 2013-8

**参考书**

1. 软件工程实践者的研究方法 （美）Roger S Pressman 机械工业出版社 2011-7
2. 软件工程（第3版） （美）Shari Lawrence Pfleeger 人民邮电出版社2007-5
3. 邹欣. 构建之法：现代软件工程. 人民邮电出版社, 第二版, 2015-7.
4. 软件工程教程 孙涌 机械工业出版社 2010-4
5. Andrew Hunt. 程序员修炼之道：从小工到专家. 电子工业出版社, 2011-1.
6. Andy Hunt. 程序员的思维修炼：开发认知潜能的九堂课.人民邮电出版社, 2010.
7. [美] 乔治·莱考夫, 马克·约翰逊. 我们赖以生存的隐喻. 浙江大学出版社, 2015-4.
8. Leo Hui. 程序员的自我修养.
https://www.gitbook.com/read/book/leohxj/a-programmer-prepares
9. [美] 史蒂夫·迈克康奈尔. 代码大全. 电子工业出版社, 第2版, 2006-3.
10. [美] Bruce Eckel. Java编程思想. 机械工业出版社, 第4版, 2007-6.
11. Cay S. Horstmann, Gary Cornell. Java核心技术·卷1：基础知识. 机械工业出版社, 第9版, 2013-11.
12. Cay S. Horstmann, Gary Cornell. Java核心技术·卷2高级特性. 机械工业出版社,第9版, 2014-3.
13. [美] 泽德 A. 肖(Zed A. Shaw). “笨办法”学Python. 人民邮电出版社, 第3版, 2014-10.
14. Christine Hofmeister, Robert Nord, Dilip Soni.实用软件体系结构. 北京:电子工业出版社,2004.
15. William E. Shotts, Jr. 快乐的Linux命令行. Second Internet Edition,
http://billie66.github.io/TLCL/
16. Arnold Robbins, Elbert Hannah, Linda Lamb. 学习vi和Vim编辑器. 东南大学出版社, 2011-3.

八、其他

无