

# 《软件工程》实践课程大纲

## 一、教材：《软件工程》（第4版）齐治昌 谭庆平 宁洪 高等教育出版社 2019.3

《软件工程》课程教材，强调理论与实践相结合，软件工程的技术、方法与工具相结合，系统地介绍了 UML 和面向对象软件开发方法，以及软件开发的需求、设计、实现、测试、交付、维护、软件度量以及软件项目管理等。

## 二、学时：20 理论+12 实验，学分 2

## 三、课程成绩：

**个人作业（20%）：**Educoder 线上作业（包括软件工程相关概念、理论题和应用测试题）[www.educoder.net](http://www.educoder.net)，注册登录，教师和学生以课堂邀请码进入不同课堂，学生完成每个章节作业后，系统自动批改评分。

**项目作业（80%）：**以小组形式完成一个软件项目的设计与开发。**班级内**自行组队（4 人左右），在 Gitee 平台上记录软件过程及管理，最后每组进行项目汇报，并提交一个项目报告（有机组合 Gitee 平台上提交的相关文档），相关格式可参考“《软件工程》项目报告撰写内容及规范化参考样本”，注意：报告中的体会部分需要每个组员根据自己的项目分工进行撰写。

## 四、课程安排

课程以理论带动实践的形式开展，以软件项目开发的框架为指导，运用软件工程的方法引导学生科学地进行软件项目的设计与实现，并指导学生运用目前先进的软件开发和管理的工具。

### 1、第一次课（4 个理论学时）

讲授内容：软件工程背景、概念、软件过程和软件生命周期模型

学生完成：运用 Gitee 平台记录项目立项（<https://gitee.com>），各班学委收集课堂分组选题情况，每组给出 Gitee 平台主页名称。完成主要工作包括：

- （1）运用头脑风暴进行选题；
- （2）用 mindmap 画所定选题的思维导图；
- （3）相应的会议记录、照片；
- （4）确定需求分析的 NABCD 模型。

### 2、第二次课（4 个理论学时）

讲授内容：如何运用 UML 构建需求分析模型

点评各组的项目立项，分析利弊

学生完成：（1）掌握如何运用墨刀进行快速原型系统的构建；

(2)掌握 Gitee 平台中如何运用 WIKI 进行文档的记录(markdown 模式)；

(3)如何运用 processon 进行在线画图。

### 3、第三次课（2 个理论学时+2 个实践学时）

讲授内容：软件项目管理、软件成本计算、人员安排、时间规划等

**第一次沟通：**老师与各小组进行选题确定以及项目规划的讨论

学生完成：（1）快速原型系统；

（2）需求规格说明报告（包括需求分析模型（UML）、软件计划甘特图、任务分配表），发布在 Gitee 平台上，老师进行点评。

### 4、第四次课（4 个理论学时）

讲授内容：软件架构设计、详细设计理论

### 5、第五次课（（4 个实践学时）

**第二次沟通：**老师与各小组进行项目最新进展的讨论

学生完成：（1）软件项目的架构设计、数据结构及算法设计；

（2）提交软件设计报告；

（3）编码实现软件项目的相关功能，运用 Gitee 平台进行代码的迁入迁出与管理，提交源代码及可执行程序。

### 6、第六次课（2 个理论学时+2 个实践学时）

讲授内容：软件测试相关理论与方法

### 7、第七次课（4 个实践学时）

学生完成：（1）软件项目的功能测试与性能测试；

（2）提交测试报告；

（3）提交用户说明；

（4）公开发布软件应用、并收集用户反馈进行分析。

### 8、第八次课（4 个理论学时）

学生分组对所开发的软件进行公开的讲解与展示（每组 8 分钟），全程专人记录，老师点评。

# 《软件工程》实践课程重点操作说明

以 <https://gitee.com/cshust/sedemo> 为例。

## 一、项目立项

### 1、任务：

(1) 开一次头脑风暴确认一个项目，项目选题参考请看本文档“软件工程项目选题参考”

(2) 对该项目进行 NABCD 分析

### 2、成果提交：

(1) 以 wiki 文档的方式记录项目立项过程

(2) 参会者，角色，时间

(3) 开会的照片

(4) 头脑风暴的 mindmap 图

(4) 最终项目的 NABCD 模型

### 3、提示：

Gitee 上 wiki 使用的是文件格式为 Markdown，编写前需要掌握一定的 Markdown 知识。下面两个基础教材，任选一个学习即可。

<https://www.jianshu.com/p/335db5716248>

<https://www.runoob.com/markdown/md-tutorial.html>

mindmap 图可以用很多软件生成，网上有很多，如果不想安装，也可以使用在线的工具生成，例如 [processon.com](http://processon.com)。

## 4、NABCD 模型介绍

### (1) 概述

1) N(Need 需求)你的创意解决了用户的什么需求?

这里的需求不是去细化一个软件的需求，而是一个理由，为什么要做这个软件。回答应该是最重要的那个，先确定这个最重要的理由，后面的就是围绕这

一个理由而进行，时刻记着，这个是你的原始目的。一般来说这个也是用户的痛点。

2) A(Approach 做法)你有什么招数来解决用户的痛苦或问题？

这些招数不光是技术上的，也可以是商业模式上的，地狱的，人脉的，行业的。

3) B(Benefit 好处)你这个产品或服务会给用户带来什么好处？

4) C(Competitors 竞争)你的产品有没有类似的竞争者，他们的产品怎么样？

5) D(Delivery 推广)你如何推销你的产品？

## (2) 案例

下面给出一个案例让大家参考：

### 游戏攻略搜寻平台：

1) N(Need 需求)：玩家用户在进行硬核游戏体验时，常常会遇到无法通关副本，而且没有时间去研究如何去快速通关并获得副本奖励的方法，用户希望通过一个可靠且方便的平台快速准确的找到适合自己的通关方法；

2) A(Approach 方法)：平台通过招募专业玩家进行攻略制作或者与已经制作好攻略的人沟通转载其攻略。对游戏进行划分，使用户可以快速的找到自己所需要的攻略；

3) B(Benefit 好处)：用户可以通过平台快速的找到自己所需要的攻略，提高自己的游戏体验，也可以逐渐加入到制作者的队伍中，体验攻略制作的乐趣。攻略制作者可以通过在平台上上传的攻略的到应得的收益，可以提高制作者的工作动力与效率，使得制作出来的攻略更好；

4) C(Competitors 竞争)：目前存在许多视频门户网站，其中有不少游戏攻略向视频，但是不集中，用户难在同一个门户网站找到多种游戏的攻略，但是因为这些网站已经经营很长时间，存在许多的老用户，平台前期发展较难，竞争者较多；

5) (Delivery 推广)：联合游戏厂商进行推广，让官方推荐我们的平台给玩家使用。

其中还可加入创新方法：

平台不仅仅使用户可以快速的搜寻自己想要的攻略，同时也可以让用户无条件的加入制作者的队伍，体验更多乐趣。

NABCD 分析对格式并无特定要求，只要能回答模型中的问题即可。

## 二、原型设计

### 1、任务：

- (1) 使用墨刀设计软件的原型
- (2) 原型需要包括一个完整的可交互流程

### 2、成果提交：

- (1) 以 wiki 文档的方式记录原型截图
- (2) 原型的可访问地址

### 3、提示：

软件原型设计工具很多，推荐使用墨刀的原因主要是墨刀提供了很多已有的模板，并且可以直接发布在线可交互原型，而且免费。如果你有熟悉的工具也可以使用。建议先看看模板库，挑选一个和你的设计相近的模板，在此基础上修改，可以节约大量的时间。

## 三、进度计划

### 1、任务：

- (1) 对项目进行功能分解，分解后的任务数不少于 30 个
- (2) 将任务分配到个人，并计划你的开发时间
- (3) 用甘特图记录小组的任务计划结果

### 2、成果提交：

- (1) 进度计划放入 Wiki，包括甘特图

### 3、提示：

首先进行任务分配，合理安排项目的时间进度，每个人都需要领到 5-10 个任务，这个粒度比较合适，粒度太小没有必要，太大则难以保证进度的监控，大家可以体会一下。甘特图的绘画工具可以用 **project** 等软件，也可以采用在线的工具，例如

<https://www.mindsup.com.cn/>

#### 四、软件设计与实现

##### 1、任务

(1) 软件项目的架构设计、数据结构及算法设计、实现

##### 2、成果提交：

(1) 提交软件设计报告

(2) 编码实现软件项目的相关功能，运用 **Gitee** 平台进行代码的迁入迁出与管理，提交源代码及可执行程序

##### 3、提示

项目的实现语言并没有做严格的限定，但会重点关注项目实施的工作量（代码行数），**建议尽量不要选择能运用现有模块做游戏开发的项目。**

#### 五、软件发布

##### 1、任务：

(1) 做一个网页介绍自己的软件，包含截图、使用介绍、获取方法，提示提供准确的获取方法，在交叉评估环节需要保证其余同学可以评估你的软件

(2) 鼓励正式上架自己的产品

##### 2、成果提交：

(1) 利用 **Gitee** 的 **page** 服务，发布你的网站

(2) 如果是用别的服务器发布，也请将发布地址，放入项目的首页

(3) 本任务在最后一次实验课之前，必须完成

##### 3、提示：

在项目网站介绍上应该给出明确的获取所开发软件的方法，如果是手机软件请给出二维码，如果是单机软件给出下载链接，网站应用给出链接地址，然后对

软件给出一定的使用介绍。网页的发布可以用自己的方法，也可以直接用 Gitee 的 Page 服务，如何用 Page 服务发布网站，请自行学习，在我校的计算机基础课程仓库中也有一个说明。发布的网站保留至少 1 个月，以备检查。

## 软件工程项目选题参考

### 1、学生选题系统

相关课程课设老师很苦恼，每次在课设选题时，都有学生抱怨选不到想要的题目。如何决定学生的选题顺序成了个很大的问题。如何公正公开的解决这个问题呢？

提示：表面上只是一个如何决定次序的问题，但是题库的维护，学生的注册，选题过程的展示，学生如何参与如何操作，都是个大问题。需求分析可以邀请老师参加哦。

### 2、儿童学习

小明的妈妈近期很郁闷，小明上小学后，老师要求每天完成 30 道口算题，每天出题改题成了一个大负担，“有个可以出题的软件就好了”小明妈妈这么想。真的可以做出一个解决小明妈妈烦恼的软件吗？

提示：小明现在在一年级，小明妈妈可是希望买一个软件可以用三年的。你的用户可以询问任何一个小朋友在读书的家长，看看他们对这个软件有什么需求。

注意：该题目的完成形式可以是 App，小程序，但是必须要完成最终的发布，可以让普通用户通过互联网直接试用。

### 3、区块链（通道）游戏

区块链（Blockchain）是一种提供数据可信性支持的基础设施，但目前其缺点在于事务处理能力较低。试结合区块链的可信特性，开发一款五子棋（或其它）游戏。

要求：（1）在游戏结束时，能够使用区块链正确结算游戏积分；（2）并尽量避免游戏给底层链系统带来较大的事务处理负担。

提示：（1）分析现有区块链游戏的设计思路；（2）了解通道技术是如何降低区块链系统事务负载的。

#### **4、基于零知识证明的身份验证小工具**

零知识证明技术能够在不泄露关键用户信息的情况下，证明信息的存在性或正确性，在隐私保护领域中极具价值。试基于默克树（或其它数据结构），构建一套用户身份验证机制，使得用户在不暴露其公钥（视作用户 ID）的情况下，亦能证明其属于某一（用户）集合。

要求：（1）假设用户的公钥已经被预先处理并包含在表征（用户）集合的信息中；（2）在验证过程中，公钥不能暴露给任何第三方。

#### **5、P2P 存储文件下载工具**

在 P2P 存储系统中，文件被分片保存在互联网连接的众多对等节点上。试开发一款文件下载小工具，能够基于文件分片的位置信息与实时网络状况，尽可能快的下载到完整文件。

要求：（1）以提升用户体验为目标设计软件；（2）考虑对等节点不稳定，且文件分片包含纠删码冗余（Reed-Solomon Codes，简称 RS 码）的情况。

提示：了解 RS 码的特性——当文件被分成  $k$  个（相同大小的）分片，且此  $k$  个分片共同生成了  $m$  个冗余分片时，这  $k+m$  个分片中的任意  $k$  个都能够修复剩余所有的（ $m$  个）分片。

#### **6、自主选题**

题目要有新意，并与老师进行商讨，充分体现软件工程的方法。



## 《软件工程》课程项目评价指标

评价指标	满分	评价标准
问题概述、需求分析 (20%)	100	NABCD 模型描述 (30%) 需求分析 (70%)
原型系统设计、概要设计、详细设计 (35%)	100	原型系统设计 (30%) 概要设计 (30%) 详细设计 (40%)
编码与测试 (30%)	100	编码规范, 代码量, 运用 Gitee 平台进行管理 (50%) 测试计划与测试用例 (50%)
功能创意 (10%)	100	很有特色, 抓眼球: 90+ 较有特色, 有实用价值: 80+ 特色一般, 有一定实用价值: 80-
真实用户反馈 (5%)	100	有用户反馈: 80, 反馈充分: 80+, 用户反馈很少: 80-。
逾期扣分	10	逾期提交: 2/天。超过 5 天者本次实验记 0。
<p style="text-align: center;"><b>综合成绩=项目成绩×95%+考勤×5%</b></p> <p style="text-align: center;">项目成绩=(Σ需求分析×20%+设计×35%+实现×30%+创意×10%+反馈×5%-逾期扣分)</p>		