

基于小米便签的阅读维护——案例使用说明书

1. 案例概述

软件工程系列课程的主要目标是提高学生对复杂软件系统和较大规模软件的开发素养和能力。传统的软件工程课程实践教学主要是软件项目实践，学生在已经学习大规模软件开发方法和技术的基础上，运用所学知识设计开发一个具有一定规模的真实软件。但由于学生没有接触过真实的优秀软件项目，个人认识还停留在几十上百行的小规模代码中，因此在实际项目开发中还是不知道该如何开发，软件工程方法的应用趋于形式。同时，软件工程方法的很多经验和技巧只有让学生见识优秀软件工程师所开发的优秀软件项目，才能让学生在理论学习阶段与软件工程理论和方法产生共鸣，并在项目实践阶段模仿优秀软件工程师的设计开发方法，提高学生开发较大规模高质量软件的能力。

针对软件工程课程实践教学中学生缺乏大规模高质量软件开发经验的问题，我们提出让学生阅读和维护具有一定规模的高质量开源软件，通过尽早接触优秀的高质量软件来增强学生工程意识。基于这种思想，设计提出了在软件工程课程实践教学中的应用开源软件阅读的基本方案。开源软件阅读维护与课程其他内容之间的关系如图 1 所示，整个课程分为三个阶段：开源软件阅读维护、软件工程理论教学和软件项目实践。

开源软件阅读维护阶段，学生通过阅读维护具有一定规模的高质量开源软件，首先从感性上接触和认识较大规模高质量软件，并从中学习高质量软件的开发方法和原则。具体来说，该阶段要求学生通过开源软件的阅读提取出其需求和设计。由于还没有系统学习软件工程的理论和方法，学生可能会出现无从下手，或者描述不清晰、不规范等情况。这些也正是学生课程理论学习阶段需要解决的问题。有了这些体验，学生就会带着问题去深入学习软件工程方法和技术，并可以作为案例来辅助软件工程理论方法和技术的教学。学生在掌握需求分析、软件设计和代码测试的方法与技术之后可以进一步完善从小米便签中提取出来的软件需求与设计。同时，在开源软件阅读维护和软件工程理论教学部分所学习到的经验和方法又可以进一步指导后期的创意软件项目实践。三个阶段由浅入深、相辅相成。通过这三个阶段，系统培养学生大规模高质量软件的开发能力和素养，从而实现课程实践教学目标。

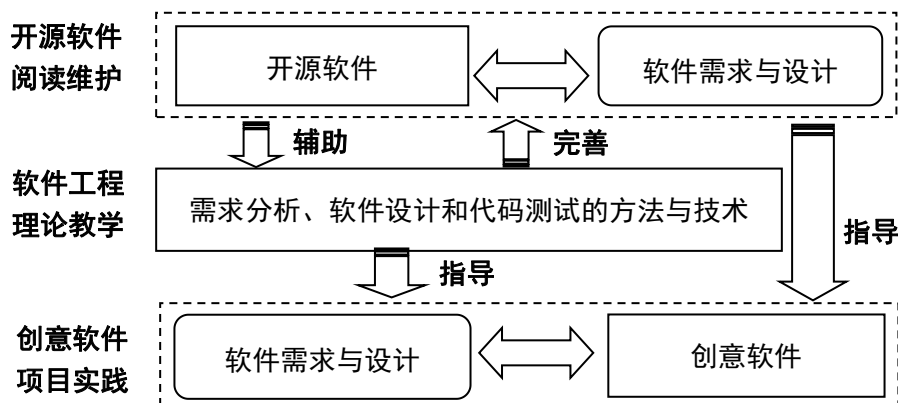


图 1 开源软件阅读维护在软件工程课程教学中的定位

作为软件工程课程实践教学阅读用的开源软件，需要具备如下几个特征。

特征之一：面向实际应用。当前比较热门的开源软件往往都采用了最先进的方法和技术。由于学生在后期软件项目实践时通常都会参照和模仿被阅读的开源软件进行开发，面向实际应用的软件能让学生学习到最新软件开发方法和技术。

特征之二：代码质量高。首先，该软件必须具有高质量的软件设计，包括软件体系结构、类设计等。其次，该软件必须具有非常规范的编码风格，让学生通过阅读高质量代码，意识到自己代码的缺陷，并养成良好的编码习惯。

特征之三：代码规模适中。代码规模太小不足以让学生体会到软件工程的思想，但模太大又会给学生造成一定的压力和负担。在阅读完该开源软件之后，学生应该能较好地驾驭该软件。通常来说被阅读开源软件的规模在一万行代码左右比较合适。

小米便签是小米公司 MIUI 团队开发的一款手机 App。Android 编程是目前非常热门的一项编程技术，便签也是大家广为熟悉的一款日常使用软件，大部分学生对该软件都比较感兴趣。该软件包含大概一万行代码，代码规模适中，适合学生在课程实践中开展阅读。由于是由小米专业开发团队精心为 Android 初学者所打造的一款软件，该软件不管从设计还是从代码的角度来说都具有较高质量。

2. 案例教学目标

本案例的主要目标是通过让学生阅读维护优秀软件工程师所开发的优秀软件项目让学生建立对较大规模高质量软件的感性认识，为软件工程方法和技术的理论教学和后期的软件项目实践打下基础。同时在此过程中培养学生对较大规模复杂软件的分析、设计、和开发能力，以及基于 Git 的团队协作开发和复用开源软件实现软件功能的能力。

3. 案例准备

学生需要会基本的程序设计，最好会 Java 编程，有一定的程序编程经验。

4. 案例教学组织方式

为引导学生一步步阅读开源软件，将开源软件阅读分为三个阶段：代码泛读、代码精读以及代码维护，如图 2 所示。三个阶段计划总共安排 12 周时间。

1) 代码泛读（4 周）

代码泛读的主要目标是让学生在较短的时间内快速掌握该软件的功能和概要设计，包括软件的体系架构、模块划分等。要求学生 2 人一组，采取结对的方式完成对整个小米便签近 1 万行代码的泛读。学生需要通过泛读小米便签代码给出小米便签的：

- **用例图：**通过用例图来描述小米便签的主要功能以及它们之间的关系；
- **体系结构图：**通过绘制体系结构图来了解整个软件的总体设计思路；
- **类之间的调用关系图：**通过绘制类之间的调用关系图来掌握小米便签的具体设计；
- **核心类的主要作用：**通过给出核心类的主要作用来进一步加深对软件设计的理解。

2) 代码精读（4 周）

代码精读的目标是在学生已经基本掌握软件整体架构的基础上，进一步体会软件的详细设计、代码规范、实现方法和技巧等。要求学生 2 人一组，采取结对的方式开展代码精读。考虑到时间的问题，每组只需要完成部分模块约 3000 行代码的精读。小米便签代码没有代码注释，学生需要在精读过程中对精读代码进行标注，包括对相应的类、类的成员变量、成员函数、以及关键代码进行标注。总的来说，在小米便签的代码精读过程中，学生需要给出：

- **关键用例的顺序图和详细类图：**通过绘制顺序图和详细类图理解关键用例的实现过程。
- **精读部分的代码标注：**通过对代码进行标注加深与评估学生对代码的理解程度；
- **代码编写规范和编程技巧总结报告：**通过撰写编写规范和技巧报告让学生对学习到的代码编程方法进行归纳总结。

3) 代码维护（4 周）

代码维护是在学生已经掌握整个软件的整体架构，也对某一部分代码非常熟悉的前提下，要求学生在该开源软件的基础上新增新的功能、对代码进行测试、或者修改维护已有的功能。要求新添加代码的质量与原来的代码质量保持一致，学生在代码维护阶段可以利用网上已有的开源软件代码。代码维护阶段是对前两个阶段所学知识和方法的一次初步运用，也为后面软件项目实践中开发高质量软件打下基础。在代码维护过程中学生需要给出：

- **新增或修改的软件功能说明：**包括新增的软件功能说明以及实现方案；
- **新增或修改的软件代码：**要求采用团队协作开发的方式通过 git 提交。

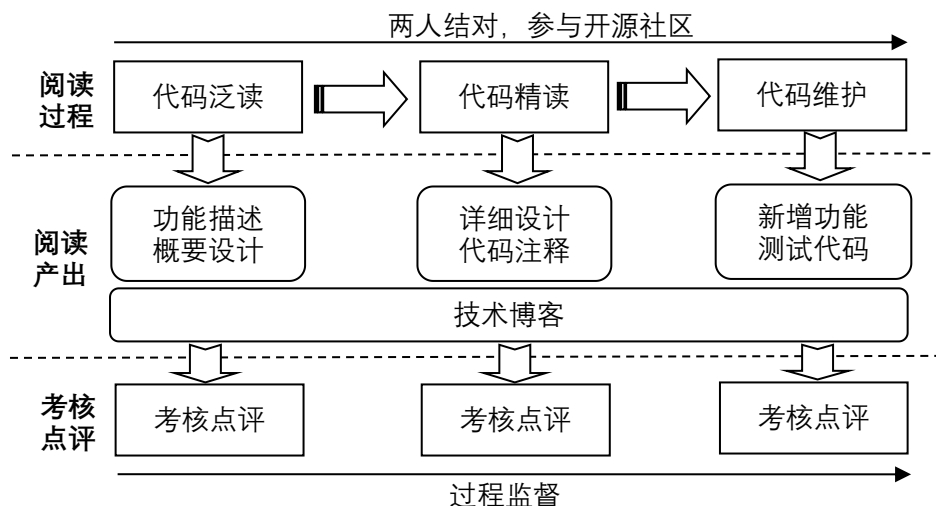


图 2 开源软件阅读维护组织方式

为督促学生积极开展代码阅读工作，也为了总结自己学习到的知识，还要求学生每周都需要撰写一篇技术博客，主要描述自己一周里面学习到的内容，例如关于协作开发、开源软件复用、需求描述和项目设计、以及自己遇到的问题和解决方案等。每个阶段结束之后都会对该阶段的成果进行考核点评，给出各小组各阶段的成绩，对存在的问题进行点评，并要求学生在点评之后进一步完善相关工作。整个过程中都会持续对学生的任务完成情况进行监督，包括跟踪大家的平台活跃情况、技术博客撰写情况、成果完成质量、新增代码质量和规模等。

5. 开放性问题

通过对小米便签进行阅读维护，学生还需要思考下面一些问题：为实现小米便签的功能，小米便签的整体实现为什么要这么设计，这么设计有什么好处，还存在一些什么可以改进的地方等。