

AI 赋能与产业项目牵引的软件 工程实践教学改革探索

巩庆志 王科 朱毅

大连外国语学院软件学院, 大连 116044

摘要 针对当前软件工程实践教学存在真实工程场景缺失、教学环节衔接割裂、AI 工具应用不规范等问题, 本文以设施农业智能控制系统产业项目为载体, 探索 AI 赋能与产业项目双向牵引的教学改革路径。构建课程实践、课程设计、综合实训、毕业设计、创新创业五段贯通的实践体系, 将 AI 工具分层规范化融入需求分析、架构设计、代码开发、功能测试等软件开发全流程。同时创新学期开放式硬件支持、优秀学生辅导导师制、分阶段工程化评价三项配套机制。教学实践表明, 该模式有效打破理论与实践壁垒, 规范学生 AI 辅助开发行为, 强化团队协作与系统集成能力, 全面提升学生工程实践素养, 为应用型高校软件工程实践教学改革提供参考。

关键字 软件工程, 人工智能, 产业项目, 五段贯通, 实践教学改革

Exploration on the Reform of Software Engineering Practical Teaching Empowered by AI and Driven by Industrial Projects

Qingzhi Gong Ke Wang Yi Zhu

School of Software, Dalian University of Foreign Languages
Dalian 116044, China

Abstract—Aiming at the problems in current software engineering practical teaching, such as the lack of real engineering scenarios, fragmented teaching links, and non-standard application of AI tools, this paper takes the industrial project of intelligent control system for facility agriculture as the carrier, and explores the teaching reform path driven by AI empowerment and industrial projects. A five-stage integrated practice system encompassing course practice, curriculum design, comprehensive training, graduation project, and innovation and entrepreneurship is constructed, in which AI tools are hierarchically and normatively integrated into the whole software development process, including requirement analysis, architecture design, code development, and functional testing. Meanwhile, three supporting mechanisms are innovated: semester-long open hardware support, an outstanding-student tutoring system, and phased engineering evaluation. Teaching practice shows that this model effectively breaks down the barrier between theory and practice, standardizes students' AI-assisted development behavior, strengthens teamwork and system integration ability, and comprehensively improves students' engineering practice literacy, providing a reference for the reform of software engineering practical teaching in application-oriented universities.

Keywords—software engineering, artificial intelligence, industrial projects, five-stage progression, practical teaching reform

1 引言

以 ChatGPT、GitHub Copilot、Cursor、Claude 等为代表的 AI 开发工具正在重塑软件工程实践方式。产业场景中, AI 已从编码辅助扩展到需求分析、设计、测试与调试等环节, 这使高校软件工程实践教学面临新的调整要求^[1-2]。

传统教学长期以手写代码和结果验证为中心, 而在 AI 能够高效生成常规代码的背景下, 教学重点应从“写出程序”转向“理解需求、把握设计、审查结果、保障质量”^[3]。与此同时, 实践项目虚化、教学环节割裂、课后实践不足、评价偏重结果等问题仍较突出,

学生容易借助 AI 完成任务, 却未形成完整工程认知与质量判断能力^[4]。

为此, 本文依托编者研发的设施农业智能控制系统产品项目, 提出 AI 赋能与产业项目牵引相结合的软件工程实践教学改革思路, 构建课程实践、课程设计、综合实训、毕业设计、创新创业五段贯通式训练结构, 为应用型本科院校提供可操作参考。

2 AI 时代软件工程实践教学面临的核心挑战

2.1 AI 工具融入实践教学的三种失当状态

当前高校在 AI 工具融入软件工程实践教学时, 主要存在 3 种失当状态。

(1) 封堵式: 因担心抄袭而简单禁止使用 AI, 结果使学生与企业真实开发方式脱节^[5]。

(2) 放任式: 允许使用但缺少规范指导, 学生直接提交 AI 生成结果, 难以形成代码理解和问题分析能力。

(3) 贴标签式: 仅在课程表述中加入 AI 元素, 未将其真正嵌入工程训练流程, 改革流于形式。

3 种状态的共同根源在于: 目前高校软件工程教育界尚未形成对“AI 如何恰当融入工程训练”的清晰方法论, 缺乏从课程设计到评价机制的完整可操作范式^[8]。

2.2 传统实践教学存在的问题

除 AI 融入困境外, 传统实践教学还存在以下问题。

(1) 项目工程性不足。训练多停留在小程序验证, 缺乏真实需求约束、接口边界和异常场景^[9]。

(2) 环节割裂。课程实验、课程设计、综合实训和毕业设计相互分离, 学生难以形成连续工程认知。

(3) 持续实践条件不足。设备和平台主要限于课堂使用, 难以支撑反复调试和迭代。

(4) 评价偏重结果。只看程序是否运行, 难以识别学生在分析、设计、测试和 AI 使用中的真实能力。

2.3 AI 时代软件工程人才能力模型的新要求

面向 AI 时代, 软件工程实践教学不仅要关注编码能力, 还要突出“会分析、会借助、会判断、会验证”的综合工程素养^[9]。结合软件工程专业培养要求和 AI 辅助开发特征, 本文将核心能力概括为以下 5 个方面:

- (1) 需求理解与业务分析能力;
- (2) AI 工具使用与 Prompt 表达能力;
- (3) AI 生成结果的审查与判断能力;
- (4) 系统架构与模块设计能力;
- (5) 工程规范、测试验证与质量控制能力。

上述能力由“会做题”转向“会做项目”, 也是后文设计实践教学改革的主要依据。面向 AI 时代, 软件工程实践教学应强化理论与应用融合、过程与结果并重, 构建多元化、工程化的培养与评价体系^[8,9]。

3 AI 赋能的软件工程五段贯通式实践教学设计

3.1 核心理念: 工具链融入工程全流程

本文的核心改革理念是: AI 工具不应成为实践教学中的“装饰性要素”或“禁用对象”, 而应作为工程工具链的组成部分, 有序融入软件工程实践教学全过程。其要点主要体现在以下三个方面:

一是贯穿需求、设计、编码、测试和联调全流程; 二是在使用 AI 提升效率的同时, 坚持学生主导设计、审查和验证; 三是依托真实项目场景, 让学生认识 AI 在硬件接口、通信协议和异常工况处理上的边界^[7]。这样既回应了产业开发方式变化, 也避免把 AI 简单等同于“自动生成代码”。

3.2 实训载体: 设施农业智能控制系统

本文选取编者在实际场景中开发的设施农业智能控制系统作为综合实训的核心载体。该系统具备真实工程训练所需的完整特征: 技术链条覆盖嵌入式终端(单片机程序、传感器驱动、设备控制固件)、通信层(MQTT/HTTP 协议、数据上传、指令下发)、云平台服务层(后台 API、数据存储、业务逻辑)和用户交互层(微信小程序、Web 监控界面)4 个层级, 构成完整的信息物理融合系统。

系统的典型功能模块包括: 大棚通风自动控制(基于温湿度阈值的风口开闭联动逻辑)、多传感器环境参数监测(温度、湿度、PH、EC)、远程设备控制与状态回传(MQTT 协议远程指令下发与实时状态上报)、历史数据存储与趋势分析(时序数据设计与查询优化)、微信小程序监控端(设备状态展示、参数远程配置、异常报警推送)。这些模块涵盖了需求分析、接口设计、数据建模、并发控制、异常处理、安全认证等软件工程核心问题, 具有高度的教学适配性。

3.3 体系结构: 五段贯通, AI 分层融入

在此核心理念基础上, 构建课程实践、课程设计、综合实训、毕业设计、创新创业五段贯通体系, 并在各阶段根据训练目标分层融入 AI 工具, 形成由浅入深的 AI 辅助工程能力培养路径, 分为以下 5 个阶段:

(1) 第一阶段为课程实践阶段, 工程训练重心是基础技能与工程场景感知。AI 辅助代码补全、Prompt 规范训练、AI 对比实验(辅助 vs 独立)作为 AI 工具链融入要点。以辅助认知、理解为主。

(2) 第二阶段是课程设计阶段, 训练重心是模块化小系统设计与实现, AI 协同接口设计、AI 生成框架+人工审查、代码审查报告训练。以协同设计、代码审查为核心。

(3) 第三阶段为综合实训阶段, 训练重心是完整系统团队协作开发, AI 辅助需求建模、AI 驱动测试用例生成、全链条代码审查规范、AI 辅助联调诊断, 以全链条融入, 判断力优先。

(4) 第四阶段是毕业设计阶段: 训练重心是深化研究与系统优化, 融入要点是 AI 辅助文献调研梳理、AI 驱动性能分析、AI 辅助论文润色 (边界规范), 以研究性融合、边界规范为主。

(5) 第五阶段是创新创业阶段: 训练重心是成果转化与创新探索, 融入要点是 AI 辅助市场分析、AI 驱动快速原型设计、AI 支撑产品迭代方案, 以主导性使用、创新导向为主。

4 五段贯通实践教学体系详细设计

4.1 课程实践: 在真实工程场景中建立 AI 工具认知

课程实践阶段重点是让学生在真实场景任务中建立对 AI 工具的正确认知。

围绕传感器数据解析、阈值控制等基础任务, 将抽象语法训练转化为与农业控制场景相关的问题求解, 使学生在入门阶段就接触业务约束。相比单纯的语言语法练习, 这类任务更容易帮助学生理解“程序为什么这样写”。

本阶段通过“独立完成—AI 辅助—对比分析”的方式, 要求学生检查并修改 AI 生成代码, 理解其适用边界, 同时学习基本 Prompt 表达和结果校验方法。

4.2 课程设计: AI 协同下的模块化系统设计训练

课程设计阶段从单点代码实现转向模块化小系统设计, 训练学生在 AI 协作条件下完成需求分析、接口设计和代码框架构建。

任务可设置为数据采集与存储、设备控制逻辑、数据上传与断线重连、Web 监控页面等模块, 兼顾基础要求与扩展要求。通过模块化拆分, 学生能够在相对清晰的边界内完成从需求到实现的闭环训练。

教学中坚持“学生先分析、AI 后辅助、结果再审查”的路径, 要求学生在使用 AI 建议后说明修改内容及原因, 以强化设计判断力。

4.3 综合实训: AI 全链条融入的团队工程实践

综合实训阶段以设施农业智能控制系统为完整项目, 组织学生开展团队化工程实践, 是五段体系的核心环节。

4.3.1 项目结构与团队角色分工

项目按终端、后端、前端和测试等方向分工, 覆盖需求分析、架构设计、编码实现、测试联调等典型岗位任务。

4.3.2 AI 工具链在实训各阶段的具体融入方式

在需求阶段, 学生可借助 AI 梳理功能点和用例, 但必须结合现场约束补充边界条件^[6]。

在架构阶段, 学生先提出备选方案, 再借助 AI 进行比较分析, 最终独立完成决策说明。

在编码阶段, 可使用 Copilot、通义灵码等工具, 但要求做到“能解释、会测试、留记录”, 防止直接照搬。

在测试阶段, 重点检查 AI 生成用例对异常工况和边界条件的覆盖不足; 在联调阶段, 学生可利用 AI 辅助排错, 但必须自行验证诊断结论^[7]。

4.3.3 工程规范与过程管理要求

综合实训全程配套版本管理、接口文档、周报提交和阶段答辩等要求, 保证 AI 使用始终处于工程规范框架内, 使学生把“借助 AI 完成任务”逐步转化为“按工程流程完成任务”。

4.4 毕业设计: AI 辅助下的深化研究与系统优化

毕业设计阶段, 学生选取综合实训中值得深入研究的工程问题进行深化探索。典型选题方向包括: 基于机器学习的环境参数异常检测算法 (以历史数据训练模型替代固定阈值判断); 通风控制策略的多目标优化 (兼顾节能与温湿度调控效果的算法设计); 多分区温室设备联动控制的分布式协调机制; 低功耗传感节点的通信可靠性改进方案等。这类选题来源真实、问题清晰, 既有工程实践深度, 又具备一定研究探索空间。

在 AI 工具融入方面, 毕业设计阶段重点探索 AI 辅助学术研究的规范使用: 使用 AI 辅助文献调研和知识梳理; 使用 AI 辅助分析实验数据, 获取统计建议和可视化方案; 使用 AI 辅助论文语言润色。通过边界清晰的 AI 辅助规范, 培养学生在研究工作中负责任使用 AI 的习惯。

4.5 创新创业: AI 驱动的成果延伸与能力进阶

对于综合能力较强的学生, 鼓励将综合实训或毕业设计的优秀成果延伸为大学生创新创业训练项目或学科竞赛作品。在此阶段, AI 工具的使用可以更加主导: 用 AI 辅助市场需求分析、技术路线图生成和产品迭代方案的快速原型设计。

5 三项配套实训模式创新机制

5.1 学期开放式硬件支持: 打破时间边界的持续实践

软件工程能力的形成依赖持续的工程实践，传统“上课领设备、下课交设备”的模式将实践压缩在课堂时间内，严重制约了学生的自主探索与迭代能力。为此，探索“学期开放式硬件支持”机制：在学期初将单片机开发板、传感器模块（温湿度、CO₂、光照等）、通信模块（Wi-Fi、4G 等）等实验硬件发放给学生，整个学期内由学生自主保管和使用。

这一做法尤其有助于学生在真实硬件环境中发现 AI 生成代码与接口时序、通信指令、现场干扰之间的不匹配问题，从而提升独立排错能力，也让实践训练从课堂演示延伸到持续迭代。实践观察表明，采用开放式硬件支持机制后，学生在课后主动调试和功能扩展的时间显著增加，部分学生自发在 AI 生成代码基础上扩展出原题目未要求的功能，展现出真正的工程探索意识。

5.2 优秀学生辅导导师制：学长引领与 AI 工具经验传递

从上一轮课程中遴选优秀学生担任辅导导师，负责技术答疑、经验分享和过程指导，形成朋辈支持机制。

该机制既能帮助新生更快掌握 AI 辅助开发中的常见问题，也能促使辅导学生在讲解与审查过程中进一步巩固自身工程理解，从而形成“教师—学长—学生”多层次支持。

5.3 分阶段工程化评价：面向真实实训过程的能力考核

在 AI 工具广泛进入学习过程后，单纯以“代码是否运行”为依据，已难以准确反映学生的软件工程实践能力。对于真实项目实训而言，学生能力不仅体现在功能实现结果上，更体现在任务分解、接口遵循、联调排错、文档表达和独立说明等方面。结合应用型本科院校学生基础差异较大的实际，本文将评价方式由“结果单点评分”调整为“分阶段实施、过程记录支撑、个人与团队结合”的工程化评价。具体通过以下 5 个模块实现评价：

（1）阶段过程考核（25%）：关注周报、Git 提交、任务推进和 AI 使用记录；

（2）模块实现质量（25%）：关注功能完整性、代码可读性、异常处理和接口规范；

（3）联调与测试能力（20%）：关注接口联调、故障定位和边界测试；

（4）文档与答辩表现（15%）：关注需求说明、测试记录和现场说明能力；

（5）团队贡献评价（15%）：结合互评和教师观察，评价协作配合与任务承担情况。

6 教学实践成效与问题反思

6.1 教学实践成效

本文所述改革自 2023 年起在软件工程专业综合实训环节试点，2024 年起推广至课程实践与课程设计全链条，目前已覆盖两届学生（2021 级、2022 级），参与学生累计逾 200 人。以下从学生竞赛成果、工程能力指标和课堂行为变化三个维度呈现主要成效。

（1）竞赛成果：获奖层级与数量双提升

改革实施后，学生在主要学科竞赛中的获奖数量与层级均呈明显上升趋势。以改革前（2022 年）为基准，2023 年至 2025 年各主要竞赛获奖情况如表 1 所示。

表1 近三年主要竞赛获奖情况统计

竞赛名称	2022 年 (改革前)	2023年	2024年	2025年
省级计算机设计大赛	二等奖 3项	一等奖 1项、二等奖 5项	一等奖 2项、二等奖 6项，4项入围国赛	一等奖 4项、二等奖 8项，8项入围国赛
“挑战杯”竞赛	铜奖 2项	银奖 1项、铜奖 3项	金奖 1项、银奖 4项	— (届次未到)
“蓝桥杯”(省级)	约 35 项	约 50 项	约 60 项	约 65 项 (含国家级 30 余项，累计)

(2022—2025)

注：2022 年数据为改革前基准值，部分数据为区间估算；“蓝桥杯”国家级奖项为累计口径。

从竞赛成果趋势看，省级计算机设计大赛入围国赛数量由改革前 0 项增至 2025 年的 8 项，增幅显著；“挑战杯”由铜奖级别跃升至金奖，项目主题亦从单一技术演示转向乡村振兴、智慧农业等具有完整工程系统的落地场景，体现出学生综合工程能力的实质性提升。

（2）工程能力指标：课程内可测维度的改善

在综合实训环节，对两届参与学生进行了多维度过程性评估，重点考察接口设计、异常处理、AI 使用规范性及团队交付质量四项指标。表 2 列出了改革前后的对比数据。

表2 综合实训关键能力指标改革前后对比

考察指标	改革前 (2022 级)	改革后 (2023 级)	变化幅度
主动完成接口设计与异常处理的学生比例	约38%	约54%	+16% (绝对提升)
团队项目完整交付率 (含测试文档与答辩)	约61%	85%以上	提升约24个百分点
AI使用记录与说明提交合规率	未做要求	约79%	新建指标，首轮达标率较高
课后自主调试时长 (周均，开放硬件组)	约1.5小时	约 4.2 小时	提升约180%

注：课后自主调试时长通过学期末问卷调查收集（有效回收率 87%），其余指标来自教师评价记录。

上述数据中，接口设计与异常处理达标率提升 16%，反映了五段贯通体系中“模块化任务分工+联调验证”机制对系统工程思维的有效促进。完整交付率从 61%升至 85%以上，说明分阶段评价与开放硬件支持联合发力，有效改善了项目“虎头蛇尾”的传统问题。课后自主调试时长的大幅提升，则直接印证了开放硬件支持机制打破时间边界的预期效果。

(3) AI 使用行为变化：从“直接求码”到“带约束协同验证”

通过对综合实训 AI 使用记录的分析（2023 级，n=112），学生 AI 辅助开发行为呈现出可量化的结构性转变。改革前学生 AI 使用行为以“直接提交生成结果”为主，占观察记录的约 63%；改革后，“提出需求约束+生成+人工审查修改”的规范流程比例上升至约 71%，而“直接照搬 AI 输出”的比例下降至约 18%（见图 1）。这一转变说明，AI 工具规范融入机制与分阶段评价标准，切实改变了学生对 AI 的依赖方式，使其从被动接受生成结果转向主动管理 AI 辅助过程。

综合以上三个维度，本改革模式在竞赛成果、工程过程能力和AI使用规范性上均有可量化的改善表现。

需要指出的是，上述数据来源于两届学生的教学实践记录，规模尚有限，部分指标（如AI使用行为分类）依赖人工观察，存在一定主观性。后续将进一步规范数据收集方式，推进更系统的效果评估。

6. 2 存在问题与进一步改进方向

实施中也暴露出一些现实问题，这些问题也决定了该模式在二本院校推广时必须坚持“小步推进、逐项完善”的思路。

(1) 学生基础差异较大，部分学生仍容易把 AI 当作替代完成任务的工具，前置课程中的接口阅读、日志分析和调试训练仍需加强。

(2) 真实项目开发周期与课程学时之间存在矛盾，部分任务只能做到能演示、稳定性不足，后续需通过任务前移和模块复用缓解压力。

(3) 指导力量仍显不足。此类实训同时涉及软件、嵌入式、通信和行业场景知识，单一教师难以覆盖完整链条。

(4) 开放式硬件支持提高了管理与维护成本，后续需完善设备台账、借还机制和分层维护制度。

7 结束语

AI 辅助开发工具的普及，为软件工程实践教学改革提供了新的契机。本文依托设施农业智能控制系统，构建了 AI 赋能与产业项目牵引相结合的软件工程实践教学改革方案。实践表明，只有将 AI 置于真实工程约束、过程要求和评价机制之中，才能真正促进学生形成需求分析、模块开发、系统联调和问题迭代能力。后续仍需在前置能力衔接、指导团队建设和案例复用等方面持续完善。生成式 AI 与产业项目深度融合的教学改革，契合新工科与人工智能教育发展要求^[8,9]，可为应用型高校软件工程实践教学提供可复制路径。

参 考 文 献

[1] GitHub. The State of the Octoverse 2023[R]. San Francisco: GitHub Inc., 2023.

[2] Stack Overflow. Developer Survey 2024[EB/OL]. (2024-05-20) [2024-09-10]. <https://survey.stackoverflow.co/2024>.

[3] 林健. 深入扎实推进新工科建设—新工科研究与实践项目的组织和实施 [J]. 高等工程教育研究, 2017(5): 18-31.

[4] Kazemitabaar M, Chow J, Ma C K T, et al. Studying the Effect of AI Code Generators on Supporting Novice Learners in Introductory Programming[C]//Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems. New York: ACM, 2023: 1-23.

[5] Prather J, Reeves B N, Denny P, et al. It’s Weird That it Knows What I Want: Usability and Interactions with Copilot for Novice

- Programmers[J]. ACM Transactions on Computer-Human Interaction, 2023, 31(1): 1-31.
- [6] 教育部高等学校软件工程专业教学指导分委员会. 高等学校软件工程专业规范[M]. 北京: 高等教育出版社, 2011.
- [7] Pressman R S, Maxim B R. Software Engineering: A Practitioner's Approach[M]. 8th ed. New York: McGraw-Hill, 2014.
- [8] 李亚坤, 颜荣恩, 杨波, 等. 生成式人工智能背景下高校软件工程课程的教学改革与探索[J]. 计算机技术与教育学报, 2024, 12(5): 8-12.
- [9] 余超, 冯旸赫, 张俊格. “人工智能”课程教学模式改革及创新实践[J]. 计算机技术与教育学报, 2022, 10(4): 42-45.