

POM设计模式在Web UI自动化测试教学中的能力培养研究

卢慧雅¹, 郝志卿²

(1. 北京信息科技大学 管理科学与工程学院; 2. 北京浩泰思特科技有限公司, 北京 100192)

摘要: 针对软件测试课程Web UI自动化测试教学中传统非POM设计模式代码可维护性差、复用率低的问题,以校企合作的合同管理系统为测试对象,探究POM模式的教学价值。使用案例教学、项目实践与对比分析多种方法,将POM设计模式与pytest框架相结合,设计“知识讲解—环境配置—页面类开发—测试框架搭建”阶梯式教学方案,强化模式对比重构与用例分层设计。通过过程化考核数据分析,学生对设计模式的理解、代码维护灵活性、用例复用能力和团队项目完成质量均有提升,为软件测试实践课程改革与能力培养研究提供可借鉴范式。

关键词: POM设计模式; Web UI自动化测试; Pytest框架; 能力培养

DOI: 10.11907/rjtk.251491

扫描二维码阅读全文:

中图分类号: G642

文献标识码: A

文章编号: 1672-7800(2026)009-0207-06



Research on Competence Development Using POM Design Pattern in Web UI Automation Testing Teaching

LU Huiya¹, HAO Zhiqing²

(1. College of Management Science and Engineering, Beijing Information Science and Technology University;

2. Beijing Haotaisite Technology Co., Ltd., Beijing 100192, China)

Abstract: Aiming at the problems of poor maintainability and low reuse rate of traditional non-Page Object Model (POM) design pattern code in the Web UI automation testing module of the software testing curriculum, this research explores the teaching value of the POM pattern using a contract management system from industry-academia cooperation as the test object. The study employs multiple methods, including case-based teaching, project practice, and comparative analysis, integrating the POM design pattern with the pytest framework to design a step-by-step teaching plan encompassing "knowledge explanation, environment configuration, page class development, and test framework construction." This approach emphasizes pattern comparison, refactoring, and layered test case design. Analysis of process-based assessment data shows improvements in students' understanding of design patterns, flexibility in code maintenance, ability to reuse test cases, and quality of team project completion. The study provides a replicable paradigm for reforming software testing practice courses and researching competence development.

Key words: POM design pattern; Web UI automated testing; Pytest framework; capability cultivation

0 引言

伴随着互联网技术的飞速发展,Web服务软件的数量越来越多,功能也愈加复杂,开展Web UI自动化测试已经

成为提高软件质量的一个重要环节^[1-2]。传统非POM (Page Object Model)模式测试代码存在明显缺陷:线性脚本的页面操作和业务逻辑耦合过深,如果页面上的元素发生任何改动,需要逐个对引用的代码进行级联修改^[3];常规的测试类代码由于没有对页面元素进行封装而导致代

收稿日期: 2025-09-04

基金项目: 北京信息科技大学高教研究课题(2024GJYB03); 北京信息科技大学人工智能专项项目(2025JGAI19)

作者简介: 卢慧雅(1973-),女,硕士,北京信息科技大学管理科学与工程学院实验师,研究方向为信息系统自动化测试技术;郝志卿(1981-),男,北京浩泰思特科技有限公司高级工程师,研究方向为软件测试方法在行业中的应用实践。本文通讯作者: 卢慧雅。

码不易阅读、难以复用和维护。

POM 设计模式最初由马丁·福勒提出,并在 Selenium 自动化测试框架中得到了广泛应用,成为一种非常主流的自动化测试设计模式^[4-5]。特性为松耦合且内聚性高,页面元素和操作逻辑有效分离,给 Web UI 自动化测试提供了高效且可维护的解决方案。

国外研究方面,文献[6]描述了基于 QMOOD(Quality Model of Object-Oriented Design)模型的页面对象与剧本设计模式质量量化分析方法,比较两种模式在电商网站测试中的表现,为测试工程师选择设计模式、优化自动化测试提供客观依据。文献[7]提出一种基于“定位—执行—验证”的三层架构设计模式,通过分离元素定位、操作执行与结果验证逻辑,解决在复杂系统测试中的代码冗余与维护难题。上述研究均聚焦工程实践效率优化,未结合教学场景设计阶梯式实践案例,也未形成适配学生能力培养的教学指导方案。

国内研究方面,文献[8]提出一种基于优化差异 DOM 树的网页元素重定位方法,该方法主要用于提升重定位效率,同时也是目前 Web UI 自动化测试的研究热点之一。该方法未形成配套的教学案例与实践指导,难以直接用于高校软件测试课程的实验教学。文献[9]介绍了基于 Pytest 的自动化测试平台,对学生采用按需指导的方式,让学生按照给定模板编写自动化测试用例,以实现自动化测试过程,但在教学应用中,对学生自主设计测试框架、优化代码结构的创新实践引导不足。文献[10]阐述了自动化测试技术应用于各场景的实施策略,表明应重视脚本可维护性和复用性,在测试环境中结合实际情况不断调整测试方法以适应项目更新迭代需求,但未深入探讨如何将策略拆解为适配高校教学的知识点。

对于高校的教学研究而言,POM 模式教学体系研究相关资料多来自于开发者社区等技术平台,尤其在非 POM 模式与 POM 模式的对比教学、POM 模式与大语言模型辅助测试等前沿技术融合方面,成熟的教学方案与实践案例较少。主要原因在于,企业先进技术与高校教学内容衔接不足,企业中的 POM 模式应用案例未能及时转化为教学资源;软件测试领域缺乏标准化教学案例库,使得教师在开展 POM 模式教学时缺乏有效的教学素材。目前,已有研究缺乏对学生能力培养效果的全面评价维度,仅从代码实现正确性等单一角度评估学生学习成果,忽视了学生问题解决、团队协作、知识迁移等能力发展,这些共同形成了“技术成果落地教学”的研究缺口。

鉴于此,本文系统对比非 POM 模式和 POM 模式的差异,深入探究 POM 模式在测试教学中的独特价值,以提升学生实践操作与问题解决能力,为高校 Web UI 自动化测试教学提供有益范例,符合学科前沿发展趋势与行业实际需求。

1 POM 设计模式

1.1 非 POM 模式与 POM 设计模式比较

非 POM 模式测试代码存在显著弊端,主要体现为如下 3 个方面:

(1)可维护性差。线性脚本将页面操作与测试逻辑紧密结合,常规测试类代码也没有对页面元素进行有效封装,使得测试代码维护性较差。页面元素定位方式改变后,所有涉及该元素的代码都需要修改,容易出现遗漏。

(2)复用性不高。不同测试用例中如果有相同页面操作,需要重复编写代码,导致代码冗余度高,相关代码难以得到有效复用。

(3)扩展性不足。系统功能增加,测试需求变更时,非 POM 模式的代码结构会愈发混乱,代码修改难度较大,很难快速适应变化。

两种模式结构比较如图 1 所示,POM 模式将代码拆成“页面对象类+测试用例类”两层,页面对象类专门存放元素定位表达式与操作方法定义,测试用例类聚焦于测试流程编排^[11-12]。通过调用页面对象类的标准化方法执行测试,无需关注页面元素的具体实现细节,让元素变更影响“锁死”在页面对象类里,从架构层面切断元素变更对测试用例的直接影响。页面元素与操作分离后,代码的可维护性有了显著提升:页面元素若发生变更,只需修改对应页面对象类中的代码,其他测试用例不用调整。并且,页面对象类能在多个测试用例里复用,提升了开发效率,充分呈现松耦合、高内聚的特点。

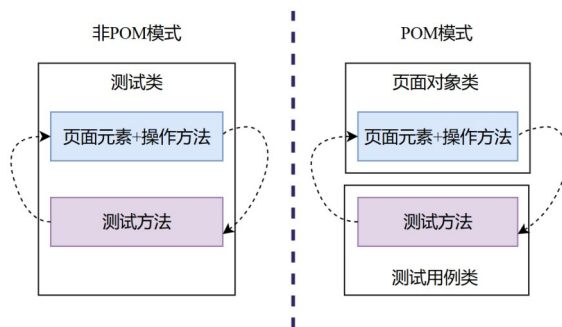


Fig. 1 Structure comparison of two modes

图1 两种模式结构比较

1.2 POM 模式原理

从设计理论层面分析,POM 模式遵循“开闭原则”和“单一职责原则^[13]”。“开闭原则”强调软件实体应对扩展开放、对修改关闭。在 POM 模式中,当 Web 页面元素发生变化,仅需修改对应页面对象类中的元素定位相关代码,而不影响其他测试用例,实现对修改的关闭;新增测试场景时,通过添加新的页面对象类或测试方法加以灵活扩展,满足对扩展的开放。“单一职责原则”要求一个类只负责一项职责,POM 模式将页面元素定位和操作逻辑封装在独立

的页面对象类中,每个类职责单一明确,逻辑结构清晰。

1.3 POM 设计模式适用性

POM 设计模式高度适配 Web UI 自动化测试,其结构清晰,方便理解和维护,目录结构如图 2 所示。该目录结构直观呈现了 POM 模式“页面层与测试层分离”的核心架构,可帮助教学场景中的学生快速理解模式的分层逻辑,为后续实践环节的环境配置、代码开发提供清晰的结构参考。

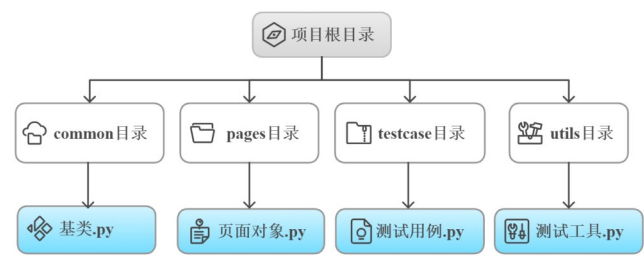


Fig. 2 Directory structure of POM design pattern
图 2 POM 设计模式目录结构

POM 设计模式适用于 Web UI 自动化测试,原因如下:

- (1)应对页面频繁变化。POM 模式将页面元素定位和操作封装在页面类中,Web 页面元素变化时仅需修改对应页面类,无需改动测试代码。
- (2)提升测试用例复用。Web UI 自动化测试里,同一页面常会被多个测试用例操作。POM 模式的页面类能够在不同的测试用例中使用,减少代码重复编写,提升开发效率,符合行业高效测试需求。
- (3)促进团队成员协作。POM 模式分离了页面元素和操作逻辑,不同的测试人员可以维护不同页面类,提高了成员之间的协作度,使得大家能够发挥各自所长。

2 研究方法与具体实施

2.1 研究方法

本文采用案例教学法、项目实践法和对比分析法相结

合的方式展开^[14],利用合同管理系统作为测试对象,搭建基于 POM 设计模式的 Web UI 自动化测试教学方案,并且进行项目实践。将 POM 模式应用前后学生对知识的掌握情况进行对比分析,从而评估 POM 模式应用效果。教学环节中,重点对比非 POM 模式与 POM 模式的差异,通过实际案例演示非 POM 模式在页面元素变化时,代码修改的复杂程度,以及 POM 模式提升代码复用、降低冗余的情况。

2.2 教学方案设计

按照布鲁姆教育目标分类法^[15],教学团队构建了“工具级(应用)→平台级(分析)→代码级(创造)”三级能力矩阵。其中,Web UI 自动化测试实践对应“创造”维度,学生在理解、应用知识的基础上进行创新性的实践操作,以实现技术能力提升。

2.2.1 教学目标设定

- (1)知识目标。全面理解 POM 模式核心理念,熟练运用该方法对页面元素及操作进行封装和优化,显著提高代码可维护性和复用性。了解学科前沿领域有关人工智能辅助自动化测试的新技术。
- (2)能力目标。学生能够熟练运用 POM 模式与 Pytest 框架,构建并实现 Web UI 自动化测试框架;能够编写高质量的测试用例,熟练掌握测试报告生成及分析方法;具备将非 POM 模式迁移优化为 POM 模式的能力;能够依据实际项目需求,灵活选取合适的测试策略。
- (3)素质目标。培养学生提升自动化测试综合素养,具备在自动化测试领域持续探索,并运用学科前沿技术达到满足行业实际需求意识与能力。

2.2.2 教学内容设计

选取校企合作项目中的合同管理系统作为测试对象。该系统包含登录、查询、新建、作废、评审等核心功能模块,预设典型缺陷,为教学提供丰富素材。教学内容主要包含 POM 设计模式和 Pytest 框架,如表 1 所示。

Table 1 Description of teaching content design
表 1 教学内容设计说明

教学阶段	教学内容
知识讲解	POM 设计模式原理;Web UI 自动化测试概念与方法;自动化测试工具的使用。非 POM 模式与 POM 模式之间的代码结构、维护成本和复用性对比案例;Pytest 框架的特性及其应用方法
环境搭建与配置	Python 开发环境搭建,Selenium 库的安装与配置,以及浏览器驱动配置;Pytest 框架安装与基础配置
页面类设计与实现	以合同管理系统页面为例,讲解页面类设计原则与方法,包括元素定位、操作封装;将非 POM 模式代码重构为 POM 模式的实践,并详细阐述重构前后差异
测试用例编写	依据合同管理系统功能需求设计测试用例,学习运用 Pytest 框架与 POM 模式对测试用例进行执行;分析非 POM 模式与 POM 模式下测试用例编写的差异;掌握 Pytest 的参数化测试、夹具等特性,以有效管理测试数据并优化执行流程
测试框架搭建	以 POM 模式为基础,利用 Pytest 框架搭建 Web UI 自动化测试框架,包含测试用例的组织、数据管理和报告生成等内容;探讨在框架中如何预留接口,以满足行业对测试框架扩展性的要求
项目实践与优化	学生分组完成合同管理系统 Web UI 自动化测试项目,优化测试框架;分析非 POM 模式与 POM 模式在项目中的应用成效

2.2.3 教学方法与实施

(1)理论教学。运用课堂讲授、多媒体演示等方法,将实际案例与人工测试、非 POM 模式和 POM 模式的差异相

结合,直观呈现传统测试方式的不足,从而激发学生的学习兴趣。详细阐述 POM 设计模式的原理,Web UI 自动化测试基本概念与方法,以及自动化测试工具的运用,助力

学生理解抽象的理论知识。比较非 POM 模式和 POM 模式的代码片段,让学生直观体会到两者在结构与可维护性方面的差异。

(2)实践教学。借由案例剖析、项目实践等手段,使学生在实际操作中学会 POM 模式运用和 Web UI 自动化测试技能。教师讲解非 POM 模式脚本如何实现合同管理系统部分功能测试、非 POM 模式代码重构为 POM 模式的方法、演示页面类的设计与实现过程,以及使用 Pytest 编写和执行测试用例的方法。学生在教师的指导下开展实践操作,完成从非 POM 模式到 POM 模式的过渡练习。期间,教师记录学生遇到的问题及解决方法,引导学生对比两种模式在实际项目中的表现,总结 POM 模式的优势。

(3)小组协作。学生分组开展合同管理系统自动化测试项目,各小组负责不同功能模块,选择测试模式、设计执

行测试用例并生成报告。教师在项目中引导学生思考不同模式的应用场景和优化策略,培养学生的团队协作能力和沟通能力,观察学生在团队项目中承担的角色、与成员的协作情况,并记录团队讨论的过程。项目结束后,每个小组进行答辩汇报,分享项目中对不同测试模式的应用体会。教师对学生的项目进行点评总结,重点分析非 POM 模式与 POM 模式在实际项目中的效果对比,并给出参考改进建议。

3 评价结果

3.1 评价指标体系构建

从知识掌握、技能提高、团队协作和创新能力这 4 个方面构建评价指标体系,具体如表 2 所示。

Table 2 Evaluation indicator system

表 2 评价指标体系

评价指标	具体内容
知识掌握	POM 设计模式之原理、Web UI 自动化测试概念与方法以及自动化测试工具的使用;非 POM 模式与 POM 模式的区别;Pytest 框架的原理与应用
技能提升	页面类设计的实现能力、测试用例编写技巧、基于 POM 和 Pytest 框架的测试搭建能力,以及测试报告生成与分析能力;非 POM 模式向 POM 模式转变的迁移实践能力
团队协作	小组分工协作情况、沟通交流能力、问题解决能力;在团队中运用学科前沿的项目管理理念以增强协作效能的表现
创新能力	项目实践中的创新思路方法、测试框架优化与改进能力;不同测试模式应用场景的创新思考

3.2 数据收集

在教学改革实践过程中,收集丰富的教学过程数据,包括课堂测验、作业、项目实践等。过程化的考核方式能够更全面地反映学生的学习状况和能力水平,有利于提高学生的学习积极性和综合素质。各部分考查分配比例如表 3 所示。

Table 3 Data sources and examination ratios

表 3 数据来源与考查比例

数据来源	考察范围	类型	比例/%
课堂测验	POM 设计模式原理、Pytest 框架基础概念	选择题	5
课堂测验	分析非 POM 模式和 POM 模式的优缺点,POM 模式在提高代码可维护性方面的原理	简答题	5
作业	POM 模式练习:页面类代码规范性	编程题	15
作业	POM 模式练习:测试用例设计覆盖度	编程题	15
作业	POM 模式练习:测试报告准确性		
作业	(作业设计阶梯式考核方案,量化学生对 POM 模式原理的理解深度)	网页	10
项目实践	测试框架代码质量	工程	20
项目实践	测试框架功能实现	工程	20
项目实践	团队协作与答疑		
项目实践	(如果学生使用了版本控制系统对项目进行精细化管理,酌情加分)	文档	10

基于表 3 中各数据来源的考查比例,将原始考核总分等比例折算为百分制,教学过程数据经过汇总统计后,各项分值如表 4 所示。

Table 4 Itemized summary of teaching process data

表 4 教学过程数据分项汇总

数据来源	折算				主要存在问题
	后总	后平	后最	后最	
	分	均分	高分	低分	
课堂测验	10	7	10	5.5	对 POM 模式页面类封装的理解存在薄弱点,仅能简单复述理论,无法结合具体测试场景展开分析,原理表述缺乏连贯性
作业	40	33	39	26	页面类与测试类设计存在耦合现象;测试用例仅覆盖正常业务流程,未考虑异常情况;测试报告缺乏关键错误场景截图,部分数据统计存在偏差,完整性与准确性不足
项目实践	50	40	48	33	测试框架未充分提取公共代码,存在部分冗余;框架功能未完全覆盖合同管理系统所有核心模块,存在功能遗漏;团队内部任务分配不均衡,协作效率较低,对项目进度产生一定影响

3.3 评价结果分析

3.3.1 知识掌握情况

课堂测验数据表明,学生对 POM 设计模式原理、线性脚本和 POM 模式差异的掌握较为深入,但在 Pytest 框架高级特性应用方面,有些学生对参数化复杂场景设置、自定义夹具等理解不够,得分率还较低,说明该部分内容的教学深度或者实践引导有所不足。

3.3.2 技能提升情况

作业及项目实践结果显示,学生在 POM 模式应用技能上呈现阶梯式提升,具体表现为:

(1)功能模块重构耗时显著缩短。超过一半的学生能够独立将线性脚本重构为 POM 模式,根据每个学生对相应环节的耗时反馈,教师针对性指导,推动效率提升。基础练习期,因学生对分层逻辑不熟悉,教师补充 BasePage 公共基类模板,拆解“元素定位提取→页面方法封装→测试用例调用”三步流程;中期针对重复编写页面操作代码问题,通过“登录页面类复用至查询模块”案例演示复用逻辑;后期引导学生复用前期代码,搭配小组互助解决个性化问题,最终使其他功能模块重构的平均耗时从最初的 8 h 逐步缩短至 4 h,反映出学生对 POM 模块化规范的掌握不断深化。

(2)测试用例复用率稳步提升。从各阶段学生提交的作业代码分析,引入 POM 模式初期,测试用例复用率仅 15%,多为重复编写“输入用户名”“点击提交”等操作;随着“页面类统一维护”教学强化,中期复用率升至 30%;综合项目实践阶段,复用率稳定在 40%~50%,基本实现模块用例跨场景复用。

(3)测试用例错误率与覆盖率改善显著。从错误率看,教学初期非 POM 模式下,合同管理系统核心模块测试用例平均错误率为 25%,主要集中于“元素定位代码重复写入测试类”“页面元素变更导致多组用例同步报错”等耦合类错误。采用 POM 模式后,错误率降至 10%,且错误类型转为“参数传递偏差”“断言逻辑疏漏”,较少出现耦合类错误,重复修改成本大幅降低。从覆盖率看,往届学生对核心模块的平均功能覆盖率为 75%,多集中在正常流程;本届 POM 模式组通过页面类按模块分工开发,平均覆盖率提升至 90%,其中“空密码登录”“已作废合同评审”等异常场景覆盖率从 20% 提升至 60%,测试完整性显著提升。

页面类与测试类设计秉持“分而治之”的理念,参考代码如图 3 所示。相较于往届学生使用线性脚本完成项目实践情况,本届学生使用了 POM 模式的小组,在系统页面元素变更时,只需修改对应页面的对象,充分体现了 POM 模式的低耦合优势。

3.3.3 团队协作情况

项目实践过程中,团队协作评价结果呈现积极态势,小组分工合理得分率相对较高。成员能根据自身技术特长,依靠协作迅速定位问题并给出解决方案,表明团队对问题解决能力较强。然而,仍存在少数小组任务分配不均衡、沟通延迟等情况,对整体项目进度产生了一定影响。通过对 3 届学生的 6 项指标进行综合评估,绘制效能雷达图,清晰展示不同批次学生在使用不同脚本模式下的综合能力情况,如图 4 所示。在“截止日期前提前完成时间”方面,使用了 POM 模式的小组平均提前 7 天完成项目,而传统模式组提前完成时间仅为 1 天;“问题解决效率”和“代码复用率”等指标也有明显提升,充分表明 POM 模式在提升项目综合质量方面具有显著优势。

```
def insert_contract(self):
    self.by_xpath(self.btn_insert_xpath).click()

# 单选按钮选择第1条记录进行作废
def invalid_contract(self):
    # 1第1个单选按钮单击
    self.by_xpath(self.input_firstradio_xpath).click()

    # 2定位作废合同按钮,单击
    if self.element_is_enabled(self.btn_invalid_xpath):
        self.by_xpath(self.btn_invalid_xpath).click()
        time.sleep(1)

# 单击退出
def logout(self):
    # 定位退出元素,单击
    self.by_xpath(self.btn_logout_xpath).click()
```

(a) Page class code example
(a) 页面类代码示例

```
# 5-作废
@pytest.mark.run(order=5)
@allure.title("作废合同UI测试用例")
def test_invalid(self, get_driver):
    driver = get_driver

    # 创建列表页面对象
    listpage = ListPage(driver)

    # 调用作废函数
    listpage.invalid_contract()

    time.sleep(2)

    # 断言作废后的记录状态为invalid, 作废按钮变灰
    assert listpage.by_xpath(listpage.first_contract_status_xpath).text == '已作废'
    assert listpage.by_xpath(listpage.invalid_btn_xpath).is_enabled() == False

    time.sleep(2)
```

(b) Test class code example
(b) 测试类代码示例

Fig. 3 Page class and test class
图 3 页面类和测试类

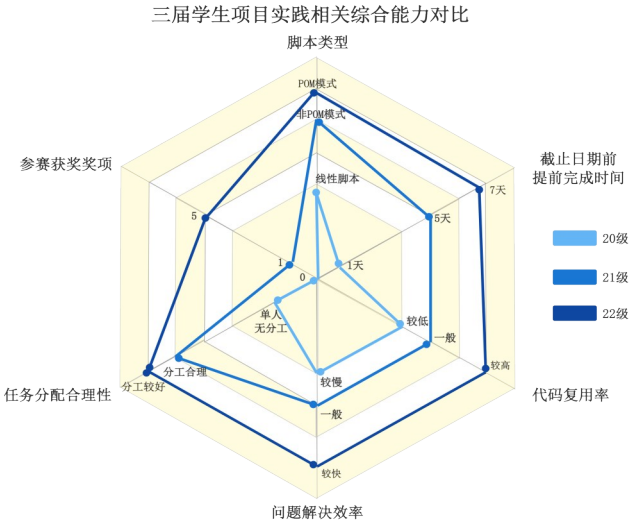


Fig. 4 Comparison of comprehensive project practice abilities
图 4 项目实践综合能力比较

3.3.4 创新能力情况

创新能力评价中,学生展现出多样化的探索。12 个实践小组中有 3 组提出结合 POM 模式与数据驱动测试,使用 Excel 或者 yaml 文件以管理测试数据,完成同一测试用例对多组数据的自动化验证,表明这些学生在培养技术灵活运用能力与创新思维方面成效较为显著。本届部分学生

结课后积极参与学科竞赛,在“全国大学生软件测试大赛”中,荣获省部级三等奖及以上共5项,彰显了学生将所学知识转化为实践创新的能力。

4 结语

本文提出一种基于POM设计模式和Pytest框架相结合的教学方案,经教学对比、实践验证发现,与传统教学模式相比,本文方案可以更好地提高学生的理论知识掌握程度、技能应用能力、团队协作精神和创新能力。学生能够依据POM模式的分层解耦思想,快速识别系统中的页面元素与业务逻辑,将积累的测试脚本进行合理改造与复用,有效缩短系统测试周期,保障了测试质量。同时,POM模式相比于非POM模式具有代码更易维护、复用性更好、测试执行效率更高等优点,可将该模式作为Web UI自动化测试教学实践的一种方式。

但同时也存在一定局限性:①样本范围小,只针对一个班,无法涵盖到不同基础的学生;②时间周期短,没有涉及POM模式与CI/CD流程的相关内容;③缺乏其他领域Web应用的测试案例,“多场景迁移能力培养”存在薄弱之处。后续研究将扩大样本范围,对多所院校、多专业同时进行比较实验研究;加长周期,增加其他领域拓展性实践环节;探究人工智能技术与POM模式、Pytest框架相融合的方式方法^[16],更好地满足行业对复合型人才的需求。

此外,高校应开展对比教学,以具体实例引导学生理解非POM模式与POM模式的应用场景。同时,强调项目驱动教学要关注理论联系实际,以解决复杂问题为目标,提升学生的实践能力。教师要时刻紧跟技术发展前沿,及时更新知识体系,将大语言模型、机器学习等方面的新知识融入课堂教学,提高学生的技术敏感度与创新能力。并且,加强校企合作,将企业项目融入教学活动,双方共同参与教学环节,培养出更多具有竞争力的软件测试人才。

参考文献:

- [1] LI H, YU H Y, ZHANG S P, et al. Web service software automation testing technology based on framework[J]. Ordnance Industry Automation, 2024, 43(8):43-46, 79.
黎晖, 于宏宇, 张绍平, 等. 基于框架的Web服务软件自动化测试技术[J]. 兵工自动化, 2024, 43(8): 43-46, 79.
- [2] LI Y T, HUANG F L. Design and development of web-based automated testing tools[J]. Industrial Control Computer, 2024, 37(11):56-58.
李玉涛, 黄飞龙. 基于Web自动化测试工具的设计与开发[J]. 工业控制计算机, 2024, 37(11):56-58.
- [3] CHEN J, SHEN Y F, ZHANG Y. Design of lightweight embedded software automatic testing framework[J]. Computer Engineering and Design, 2024, 45(3):940-947.
陈建, 沈亚峰, 张谊. 轻量级嵌入式软件自动化测试框架设计[J]. 计算机工程与设计, 2024, 45(3):940-947.
- [4] FOWLER M. PageObject [EB/OL]. <https://martinfowler.com/bliki/PageObject.html>.
- [5] MA L X, LIU H Y, ZHANG B Q. Application of web automation testing framework based on selenium and pytest[J]. Computer Knowledge and Technology, 2023, 19(34):52-54.
马立鑫, 刘海燕, 张兵权. 基于Selenium和Pytest的Web自动化测试框架应用[J]. 电脑知识与技术, 2023, 19(34):52-54.
- [6] YUNIASRI D, BADRIYAH T, SAADAH U. A comparative analysis of quality page object and screenplay design pattern on web-based automation testing[C]//Istanbul: International Conference on Electrical, Communication and Computer Engineering (ICECCE), 2020.
- [7] VADAN A M, MICLEA L C. Locate, execute, expect design pattern[C]//Cluj-Napoca: IEEE International Conference on Automation, Quality and Testing, Robotics(AQTR), 2022.
- [8] CHEN J S, PENG L F. A web element relocation method for web UI automated test script repair[J]. Journal of Changchun Normal University, 2023, 42(8):54-59.
陈俊生, 彭莉芬. 一种用于Web UI自动化测试脚本修复的网页元素重定位方法[J]. 长春师范大学学报, 2023, 42(8):54-59.
- [9] HUANG F L, ZHANG H X, LIU Z L. Automated testing platform based on pytest testing framework[J]. Industrial Control Computer, 2025, 38(2):60-62.
黄飞龙, 张焕欣, 刘泽林. 基于Pytest测试框架的自动化测试平台[J]. 工业控制计算机, 2025, 38(2):60-62.
- [10] YANG S. Application strategy of automated testing technology in software development[J]. China Information Times, 2025(3):231-233.
杨帅. 软件开发中自动化测试技术的应用策略[J]. 中国信息界, 2025(3):231-233.
- [11] MYLI B B. Automation test ix: page object pattern[EB/OL]. https://blog.csdn.net/myli_binbin/article/details/112796131.
MYLI B B. 自动化测试之九: Page Object模式[EB/OL]. https://blog.csdn.net/myli_binbin/article/details/112796131.
- [12] MYLI B B. Automation test x: pytest unit testing framework [EB/OL]. https://blog.csdn.net/myli_binbin/article/details/113037165.
MYLI B B. 自动化测试之十: pytest单元测试框架[EB/OL]. https://blog.csdn.net/myli_binbin/article/details/113037165.
- [13] GUO R. Brief discussion on design principles in software design patterns [J]. Digital World, 2015(11):5-6.
郭荣. 浅谈软件设计模式中的设计原则[J]. 数码世界, 2015(11):5-6.
- [14] JIANG X T, HU Z H, HUANG H. Exploration of case study teaching approach of computer majors in school-enterprise cooperation under the background of emerging engineering education[J]. Software Guide, 2023, 22(6):158-161.
蒋湘涛, 胡志浩, 黄辉. 新工科背景下计算机专业校企合作案例教学方式探索[J]. 软件导刊, 2023, 22(6):158-161.
- [15] BAIDU ENCYCLOPEDIA. Bloom taxonomy of educational objectives[EB/OL]. <https://baike.baidu.com/item/布鲁姆教育目标分类法/15725290>.
百度百科. 布鲁姆教育目标分类法[EB/OL]. <https://baike.baidu.com/item/布鲁姆教育目标分类法/15725290>.
- [16] LIU L L, CEN B Z. Application of artificial intelligence technology in software automated testing[J]. Software, 2025, 46(2):110-112.
刘丽琳, 岑柏滋. 人工智能技术在软件自动化测试中的运用[J]. 软件, 2025, 46(2):110-112.

(责任编辑:孙娟)