实 验 指 导 书

**《软件测试技术》课程**

实验 **1** 完成一个简单的测试过程

（共 2 学时）（作业提交时间2024.4.7日前）

**1.** 实验目的

（1）对软件测试有一个直观的感受。

（2）对软件测试过程有一个完整的感受。

（3）加强对基本概念的理解。

**2.** 实验内容

（1）针对 [https://www.saucedemo.com](https://www.saucedemo.com/) 进行分析。

（2）先做一些初步的功能测试分析，如了解功能操作的路径，有哪些输入数据，有哪 些特殊、异常的数据或操作。

（3）基于上述的分析，像用户使用产品操作软件一样，进行手工测试，发现缺陷并记 录。

（4）完成一个非规范的测试报告。

**3.** 实验过程

（1）选定一个被测系统或被测模块（统称为“被测对象 ” ）；

（2）针对被测对象进行简单的分析，明确要测试的子功能及其功能点（更细的划 分），并记录下来每个功能点（测试项）；

（3）针对每个功能点，了解其应用场景、操作和输入/输出数据，并做记录；

（4）根据上述记录，尽可能覆盖其应用场景、完成各种操作和数据的输入，并检查输 出的结果；

（5）在测试过程中尽可能做些异常操作（如连续单击）、输入一些特殊的数据，如输 入特别长的字符串、输数字的地方输字母、输入空值、 输入边界值等；

（6）当检查结果不对时，记录异常情况或缺陷；

（7）最后整理上述记录，思考和总结整个过程，形成一个完整的报告。

**4.** 实验结果

交付最后整理的测试报告，包括功能点、测试场景、输入数据、异常/缺陷等记录列 表，以及自己的感受和总结。

**实验 2 单元测试实验**

（共 2 学时）（作业提交时间2024.4.21日前）

使用 JUnit 工具，针对 Spring Unit Testing 控制器代码中 ItemController 类进行测试，编 写对应的测试类以完成单元测试，最终提交测试代码。

|  |
| --- |
| package com.sprint.unittesting.unittesting.controller;  import java.util.List;  import org.springframework.beans.factory.annotation.Autowired;  import org.springframework.web.bind.annotation.GetMapping;  import org.springframework.web.bind.annotation.RestController;  import  com.sprint.unittesting.unittesting.business.ItemBusinessService;  import com.sprint.unittesting.unittesting.model.Item;  @RestController  public class ItemController {  @Autowired  private ItemBusinessService businessService;  @GetMapping("/dummy-item")  public Item dummyItem() {  return new Item(1, "Ball", 10, 100);  }  @GetMapping("/item-from-business-service")  public Item itemFromBusinessService() {  Item item = businessService.retreiveHardcodedItem();  return item;  }  @GetMapping("/all-items-from-database")  public List<Item> retrieveAllItems() {  return businessService.retrieveAllItems();  }  } |

**实验 3 系统功能测试**

（共 4 学时）（作业提交时间2024.5.5日前）

**1.** 实验目的

（1）巩固所学到的测试方法；

（2）提高实际的测试能力。

**2.** 实验前提

（1）理解 6.1 节所描述的功能测试分析思路；

（2）熟悉测试用例设计测试方法（第 3 章内容）；

（3）会统一给出一个被测试的 Web 应用系统（SUT）。

**3.** 实验内容

针对被测试的 Web 应用系统进行功能测试，发现其存在的缺陷。

**4.** 实验环境

（1）每 3～5 个学生组成一个测试小组。

（2）SUT是给出的一个在“Java课设 ”课程中完成的应用系统。

（3）每人或每两人有一台 PC，安装了 3 种浏览器，如 IE 、Firefox、Chrome。 （4）网络连接，能够访问被测系统。

**5.** 实验过程

（1）小组讨论，分析 SUT，确定测试的范围， 列出测试项（功能点）。

（2）按功能点分工，如有 12 功能点，则每组四人，每个学生分到 3 个功能点。

（3）基于本章所学的测试方法，每个学生设计 20 个或更多的测试用例，这些用例要求 相对关键， 对功能点的验证有效，或发现 Bug 的可能性更高。

（4）每个学生向本组其他学生讲解自己是如何设计这些测试用例的，他学生提出意 见，然后大家讨论， 最后修改、完善之前写的测试用例。

（5）先选择 Firefox 执行全部测试用例，然后根据自己判断，决定在 IE 、Chrome 上执 行哪些测试用例。学生甲可以执行自己设计的测试用例，也可以交叉执行，即学生甲执行学 生乙设计的测试用例、学生乙执行学生甲设计的测试用例。

（6）记录所发现的缺陷。

（7）基于 Selenium＋Webdriver，将之前设计的测试用例转化为测试脚本，如 Java 格式 的脚本，在 Eclipse 的 JUnit 框架上执行和调试这些脚本。

（8）最后写出一个完整的测试报告，包括分析思路、功能点清单、如何设计测试用例、 脚本开发遇到哪些问题、测试环境、测试结果、Bug 列表、其它的体会感想等。

**6.** 实验结果

交付测试用例、测试脚本，以及完整的测试报告。

**实验 4 性能测试**

（共 4 学时）（作业提交时间2024.5.19日前）

**1.** 实验目的

（1）巩固所学到的系统性能测试方法；

（2）提高使用系统性能测试工具的能力。

**2.** 实验前提

（1）掌握系统性能测试方法；

（2）熟悉系统性能测试过程和工具使用的基本知识；

（3）选择一个被测试的移动应用系统（SUT）。

**3.** 实验内容

（1）针对被测试的移动应用系统的后端（Web 服务器、应用服务器、数据库服务器 等）进行性能测试。

（2）针对被测试的移动应用系统的前端（App）进行性能测试。

**4.** 实验环境

（1）每 3～5 个学生组成一个测试小组。

（2）SUT 安装在一台或多台独立的服务器上，也可以是外部系统。

（3）每人或每两人有一台安装了 Java 运行环境的 PC。

（4）准备一台基于 Android 的手机或基于 iOS 的 iPhone 手机。

（5）网络连接，能够访问 SUT。

**5.** 后端性能实验过程

（1）小组讨论性能测试方案和小组成员分工。

（2）下载后端性能测试工具 JMeter 或 nGrinder。

（3）部署 JMeter 或 nGrinder 分布式测试环境，有控制器、多个测试机。 （4）选择 SUT 多个关键性的页面，录制或开发脚本。

（5）脚本参数化：测试数据文件配置、用户自定义变量等。

（6）采样器：覆盖两种协议（如 HTTP 、JDBC 或 JMS）。

（7）针对 HTTP 协议，需要设置断言、Cookie 管理、默认值等。

（8）测试多组负载，如并发用户数为 100 、500 、1000 等。

（9）根据聚合报告、图形结果等，进行结果分析。

**6.** 前端性能实验过程

（1）登录 PerfDog 官网， 下载正确的桌面应用程序。

（2）USB 连接手机，自动检测添加手机到应用列表中。

（3）设定测试模式（USB 模式测试或 WIFI 模式测试）。

（4）选择被测试的 App 应用开始测试。

（5）测试过程中采集性能数据。

（6）保存测试结果。

（7）进行测试数据统计与分析。

（8）编写测试报告。

**7.** 实验结果

（1）记录测试完整过程（工具安装、环境设置、负载及其模式设置、脚本录制和开发、监听器、结果分析），包括脚本文件。

（2）提交前、后端性能测试报告（word 格式），描述所做的测试、遇到的问题、负载模 式、结果分析等，包括主要工具执行截图等。

**实验 6 部署自动化测试框架**

（共 4 学时）（作业提交时间2024.5.31日前）

**1.** 实验目的

（1）巩固所学到的自动化测试框架相关知识；

（2）提高自动化测试的动手能力。

**2.** 实验前提

（1）了解自动化测试框架的构成和常见的自动化测试框架；

（2）访问 Robot Framework 官方网站：<https://robotframework.org/>；

（3）熟悉 Linux 系统或其他适合部署自动化测试框架的环境。

**3.** 实验内容

（1）安装自动化测试框架。

（2）设置和集成相应的测试工具。

（3）简单使用自动化测试框架

**4.** 实验环境

（1）每 3～5 个学生组成一个测试小组；

（2）准备 2-3 台 PC 和一台服务器；

（3）网络连接，能够访问服务器和外部互联网资源。

**5.** 实验过程

（1）小组讨论部署方案和小组成员分工；

（2）如果没有 Python，安装 Python；

（3）下载安装 Robot framework ，如 pip install robotframework；

（4）安装 Robot framework 的 GUI 界面，如 wxPython+ robotframework-ride；

（5）安装第三方库（Robot framework 插件），如 SeleniumLibrary、HTTP RequestsLibrary、 AppiumLibrary 、RESTinstance 等；

（6）启动 Robot framework（RF），使用 RF 内建库，完成一些基本的操作（如时间日期、过程、操作系统等）、RF 自身的关键字驱动脚本。

（7）结合实验 3，实现 RF 的集成，生成测试报告；

（8）基于 RF 和第三方库完成接口测试的脚本等，生成测试报告。

**6.** 实验结果

交付完整的实验报告，包括记录安装和使用完整过程，遇到的问题以及如何解决问题的， 使用的效果，包括关键使用的界面截图等。

**提交地址**：

* 学院  [ftp://211.87.227.230:230/](%20ftp://211.87.227.230:230/%20%20%20)  找到王筱婷/2024春季学期/软件测试 目录。
* 账号student、密码sc.sdu.edu.cn

**命名要求**：个人文件以学号姓名方式命名。小组文件以组长学号姓名方式命名。只要求提交一个文件的不用打包，如上传多个文件，请打包。

**分组要求**：自愿分组，组长上传作业时用一个单独文件写明组员学号班级姓名和工作量等信息。工作量组长为1，其他成员可设为0.x，1.x等。

**其他注意事项**：

* 实验1与课本的区别是我们限定了具体的测试网站[https://www.saucedemo.com](https://www.saucedemo.com/)
* 实验2与课本要求一致
* 实验3到时候会提供一个Web应用系统给大家测试，不要自选
* 实验4与课本要求一致，自选移动应用系统
* 课本上实验5不做要求，
* 实验6与课本要求一致，但要注意要求（7）中需结合实验3
* 每个实验占比20%