《软件测试分析》实验报告

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 姓名 | 李爽 | 班级 | 22级软件工程专升本1班 | | | 学号 | 2206831522 |
| 实验名称 | 实验4、检查点 | | | | | | |
| 实验时间 | 2023年 4 月 20日 | | | 成绩 |  | | |
| **实验目的：**  ① 理解检查点的概念和loadrunner中插入检查点的两种方式：手工方式和利用菜单或者工具条；  ② 掌握loadrunner中插入文本检查点：web\_reg\_find()函数和图片检查点：web\_image\_check()函数；  ③ 掌握loadrunner中迭代次数的设置；  **实验环境：**  **Win10 X64**  **VMware** LoadRunnner **实验内容：**  ①录制loadrunner自带的一个订票系统，协议选择WEB（HTTP/HTML）选项，操作：一名为jojo的顾客，订阅一张从London飞往Paris航班的机票，并备份脚本信息；  ②为了回放时能够订阅往返于不同城市的机票，对备份的脚本信息中的出发城市和到达城市分别进行参数化设置。然后从【design】->【parameters】->【parameters list】打开参数列表，分别为出发城市和到达城市构建数据源，选择合适数据分配方法（Sequential）、数据更新方式（Each iteration），到达城市参数列表的Fist data设置为1，出发城市参数列表的Fist data设置为2，保存脚本；（实验3操作）  备注信息：  出发和到达城市列表：Denver、Frankfurt、London、Los Angeles、Paris、San Francisco、Seattle、Sydney、Zurich  ③在【runtime-setting】中设置迭代次数为5，且把脚本中登录操作的相关代码放置在vuser\_init中，并利用lr\_output\_message()函数输出信息“这里是init部分”，同时把退出相关代码放在vuser\_end中，并利用lr\_output\_message()函数输出信息“这里是end部分”；  ④在备份的脚本中合适的位置插入文本检查点和判断语句，判断购票支付是否成功，若成功请输出信息“订票成功，出发城市为\*\*\*，到达城市为\*\*\*”，否则输出信息“订票失败，请核实!”（\*\*\*为订阅机票的实际出发和到达城市）；  ⑤保存脚本，关闭日志，回放脚本并查看结果是否成功订阅机票。即一名为jojo的顾客，订阅5张从不同出发地飞往不同到达地的机票。同时回放日志信息查看输出信息是否正确，即  这里是init，  订票成功出发城市为Denver，到达城市为Frankfurt  订票成功出发城市为Frankfurt，到达城市为London  订票成功出发城市为London，到达城市为Los Angeles  订票成功出发城市为Los Angeles，到达城市为Paris  订票成功出发城市为Paris，到达城市为San Francisco  这里是end.  **实验过程及结果记录：**  1.脚本录制    2.对出发和到达城市进行参数化设置      3.设置回放次数    4.对脚本进行拆分      5.修改脚本，在订票之前插入文本检查函数如下    6.保存脚本，回放查看运行结果 | | | | | | | |
| **实验总结：**  在这个实验中，我使用了LoadRunner来录制订票脚本，并且添加了文本检查点以确保应用程序在运行时正常工作。  首先，我打开了LoadRunner的录制器，并选择了我的浏览器作为目标应用程序，然后开始录制脚本。  在录制脚本的过程中，我模拟了一次用户订票的过程。这包括了浏览网站、搜索航班、订票等一系列步骤。  在录制脚本后，我使用LoadRunner的脚本编辑器添加了文本检查点。这个检查点会比较应用程序返回的文本和我预先定义的标准文本。如果这两个文本不匹配，那么脚本会停止运行并报告错误。  接下来，我运行了我的脚本，以确保它在正常情况下能够正确运行。我发现文本检查点确保了应用程序的正确性，并且脚本能够按照预期的方式运行。  总的来说，这个实验让我学会了如何使用LoadRunner录制脚本和添加文本检查点。这些技能对于我的职业发展非常有用。 | | | | | | | |