****

**员工年度总结表**

**（review流程：入职每满一年当月填写总结表，发次月薪资前出review结果）**

**填表时间： 2023 年 4 月 17 日**

|  |  |  |  |
| --- | --- | --- | --- |
| **基本信息** | | | |
| 姓 名 | 兰杨能 | 入职时间 | 2020.04.22 |
| 项 目 | RemarkableFoods QA Automation | 岗 位 | JAVA Engineer |
| **员工年度总结** | | | |
| 感谢在公司度过的三年时光。回顾2022年，我非常感激前端组同事和领导们对我的技术支持和帮助，这让我不仅在技术方面得到了提升，还在工作软实力方面有了显著的进步。  **在过去的一年中，我的主要工作职责包括：**   1. 编写高质量的Java代码，包括设计、开发、测试和维护 2. 与团队成员紧密合作，参与需求分析、系统设计、代码评审等工作。 3. 解决技术问题和缺陷，确保系统的稳定性和可靠性。 4. 参与系统架构设计和优化，提高系统的性能和扩展性。 5. 学习和研究新技术，为团队提供技术支持和建议。 6. 参与项目管理、进度控制、风险评估等工作。 7. 培训和指导新员工，提高团队整体技术水平。   **在过去的一年中，我取得了以下工作成果：**   1. 完成allure测试报告的生成速率从1～2分钟的时间到秒级别的提升。 2. 完成UI比对从测试过程中剥离出来，减少测试时间，降低appcenter测试时长的消费。 3. 完成测试平台2.0第一版的原型设计&流程图&时序图的设计。 4. 完成测试设备种类的可扩展性和设备的可插拔性。 5. 完成并行测试，提升测试效率，从每台设备需要花费6个小时左右，如果平台的设备数量有4台，预估跑完测试只需要花费6/4=1.5个小时，达到测试时长=（原先测试时长/平台设备数量）的提升。 6. 完成测试用例的解析速率的提升，减少服务器内存和性能的消费。 7. 完成测试结果准确性的提升，做到提早发现业务app的缺陷。 8. 完成测试自动化程度的提升。   总体来说，在测试平台的开发中，通过提升测试效率，自动化程度，减少手动操作等方面的贡献，大大提高了测试的效率和准确性。  **在过去的一年中，我参与了以下重要项目：**  **项目：测试平台1.0**  我的角色：JAVA后台开发 & WEB前端开发  我的贡献：   1. 完成Allure测试报告的管理，设计支持app端和web端的Allure测试报告 2. 完成allure测试报告的生成速率从1～2分钟的时间到秒级别的提升，因为测试报告是临时存放在测试服务器的磁盘上的，且我们的服务部署是CICD，重启服务器的时候需要重新处理报告。 3. 完成UI比对报告的管理，支持测试开发便捷管理UI标准图库。 4. 解决UI对比算法导致服务器栈溢出的问题 5. 将UI比对从测试过程中剥离出来，节省测试时间，减少appcenter测试时长的消费。 6. 完成打包服务的管理，提供release包的IPA&APP&APK的三种类型的测试包，满足了不同的测试需求和场景。 7. 完成代理第三方API，让测开可以方便地调用和模拟第三方业务的接口，保证了测试的正确性和稳定性。   **项目：测试平台2.0**  项目角色：JAVA后台开发 &  架构助理  项目贡献：   1. 收集和分析需求：在需求阶段，通过和测试团队的沟通，和以往自己的测试经验，收集并分析测试开发团队的需求和期望，完成测试平台2.0第一版的原型设计&系统架构图&流程图&时序图的设计。 2. 提升设备资源的利用率：对于接入测试平台的操作系统，平台会主动去发现该操作系统上已安装或已连接的ios和android设备，包括虚拟机和真机，完成测试设备种类的可扩展性和设备的可插拔性。 3. 提早发现APP的缺陷：在以往的测试过程中，测试开发人员需要手动到打包服务器下载APP包到本地，再放到测试脚本中进行测试，可能导致使用旧的APP进行测试，在测试平台进行测试，可以保证每次都以系统中最新的APP包进行测试，完成测试结果准确性的提升。 4. 提升自动化程度：因为测试过程手工参与度过高，容易出现操作失误，如接收测试报告地址选错、测试设备选错等因素导致测试需要重新执行，一台设备错误测试浪费的时间最大是6个小时，在平台可以通过建立测试计划，定义JOB定期跑测试，完成杜绝手工错误带来的影响。 5. 提升测试效率：因为每个迭代留给测试开发团队的测试时间只有2天，随着测试用例的增多，测试设备的增多，原本跑在一台设备上跑一批测试用例的时间大概是6个小时，在平台里面，假设同种设备数量是N，跑完一批 case需要的时间预估可以达到6/N，团队内现有4台mac os操作系统，跑完一批case从原来的6个小时缩减到6/4=1.5个小时，接入系统的设备数量越多，跑完一次测试需要的时间越短，完成测试效率质的提升。 6. 提升测试用例的解析速率，原先使用类加载的方式，耗费内存资源和性能，需要花费至少1分钟，到现在通过asm字节码解析class文件获取测试用例，达到5秒级别完成解析，并且随着测试用例的增加，效果越明显。   **在过去的一年中，我表现出了以下优点：**   1. 需求更改调整时候，从框架到细节去确认需求，先确定好框架，等整体方案拍定，再去确定细节，逐一确认。 2. 技术设计方案输出的时候，要把可能被问到的问题，都思考清楚，怕忘记的话就用笔记记录下来。 3. 挖掘潜在需求，实际需求：对于他人提出的需求，要考虑这样设计的原因，是否合理，实际上他想要的东西是什么。 4. 会议纪要，对会议讨论的东西，记录下来发给大家二次确认。 5. 思维方式：从点性思维到线性思维，再到面性思维，也就是说从强调细节，再到整体流程，再到多个方案之间利弊分析、选择的过程，要具备面性思维。   **在过去的一年中，我也存在以下不足：**   1. 在技术提升方面，我还需要进一步学习和掌握一些新的技术和框架，以满足公司未来的发展需求。 2. 在团队合作方面，我还需要进一步提升自己的沟通和协调能力，以更好地与其他团队成员协作完成任务。   **为了改进我的工作表现，在未来的一年中，我将：**   1. 加强对新技术和框架的学习，不断提升自己的技术水平，以更好地适应公司未来的发展需求。 2. 提高自己的沟通和协调能力，积极参与团队合作，与其他团队成员协作完成任务。 3. 注重个人时间管理，合理规划工作时间，提高工作效率和质量。 4. 加强对项目进度和风险的管理，及时发现和解决可能出现的问题，确保项目按计划进行。 5. 加强对代码质量的管理和评估，通过代码审查和单元测试等方式确保代码的质量和可维护性。 6. 积极参加培训和技术交流活动，与其他同行进行交流和学习，不断提升自己的专业知识和技能。   **对团队和公司建议**   1. 建议明确工作流程，并为每个人提供流程培训，以确保大家能够严格按照流程进行工作。例如，需要共同讨论和制定需求的拍定和完成时间，并确立明确的功能验收流程。 2. 建议强调职责分明，为每个项目指定明确的责任人，并规范开发流程，以确保不会随意修改非自主负责项目的代码。同时，也需要遵守开发规范，保证代码的质量和可读性。   Thank you for the three years you have spent in the company. Looking back on 2022, I am very grateful to my colleagues and leaders in the front-end group for their technical support and help, which has not only improved my technology, but also made significant progress in my soft power at work.  In the past year, my main job responsibilities include:  1. Write high-quality Java code, including design, development, testing and maintenance  2. Work closely with team members to participate in requirements analysis, system design, code review, etc.  3. Solve technical problems and defects to ensure the stability and reliability of the system.  4. Participate in system architecture design and optimization to improve system performance and scalability.  5. Learn and research new technologies, and provide technical support and suggestions for the team.  6. Participate in project management, progress control, risk assessment and other work.  7. Train and guide new employees to improve the overall technical level of the team.  In the past year, I have achieved the following work results:  1. The generation rate of the allure test report has been improved from 1 to 2 minutes to the second level.  2. The completion of UI comparison is separated from the testing process, reducing testing time and reducing the consumption of appcenter testing time.  3. Complete the prototype design & flow chart & sequence diagram design of the first version of the test platform 2.0.  4. Complete the scalability of test equipment types and the pluggability of equipment.  5. To complete the parallel test and improve the test efficiency, it takes about 6 hours for each device. If there are 4 devices on the platform, it is estimated that it only takes 6/4=1.5 hours to run the test, and the test duration= (Original test duration/number of platform devices) has been improved.  6. Complete the improvement of the analysis rate of the test case and reduce the consumption of server memory and performance.  7. Complete the improvement of the accuracy of the test results, so as to discover the defects of the business app early.  8. Complete the improvement of test automation.  Generally speaking, in the development of the test platform, the efficiency and accuracy of the test are greatly improved by improving the test efficiency, automation, and reducing the contribution of manual operations.  In the past year, I have participated in the following important projects:  Project: Testbed 1.0  My role: JAVA background development & WEB front-end development  My contribution:  1. Complete the management of the Allure test report, design and support the Allure test report on the app side and the web side  2. Complete the increase in the generation rate of allure test reports from 1 to 2 minutes to the second level, because the test report is temporarily stored on the disk of the test server, and our service deployment is CICD, and it needs to be restarted when the server is restarted. Processing report.  3. Complete the management of UI comparison reports, support test development and convenient management of UI standard library.  4. Solve the problem of server stack overflow caused by UI comparison algorithm  5. Separate the UI comparison from the testing process to save testing time and reduce the consumption of appcenter testing time.  6. Complete the management of packaging services and provide three types of test packages of IPA&APP&APK for release packages to meet different test requirements and scenarios.  7. Complete the proxy third-party API, so that the test can easily call and simulate the interface of the third-party business, ensuring the correctness and stability of the test.  Project: Testbed 2.0  Project role: JAVA background development & architecture assistant  Project contribution:  1. Collect and analyze requirements: In the requirements stage, through communication with the test team and previous test experience, collect and analyze the needs and expectations of the test development team, and complete the prototype design & system architecture diagram of the first version of the test platform 2.0 & Flowchart & Sequence Diagram Design.  2. Improve the utilization of device resources: For the operating system connected to the test platform, the platform will actively discover the installed or connected ios and android devices on the operating system, including virtual machines and real machines, and complete the test equipment types Scalability and pluggability of devices.  3. Early detection of APP defects: In the past testing process, test developers need to manually download the APP package from the packaging server to the local, and then put it into the test script for testing, which may lead to using the old APP for testing. Testing can ensure that the latest APP package in the system is used for testing every time, and the accuracy of test results can be improved.  4. Improve the degree of automation: Because the manual participation in the test process is too high, operational errors are prone to occur. For example, the wrong address for receiving the test report, the wrong test equipment and other factors cause the test to be re-executed. The time wasted by a wrong test on one device is at most In 6 hours, you can create a test plan on the platform and define JOB to run tests regularly to eliminate the impact of manual errors.  5. Improve test efficiency: Because the test time left for the test development team for each iteration is only 2 days, with the increase of test cases and test equipment, the original time to run a batch of test cases on one device is about 6 hours. In the platform, assuming that the number of devices of the same type is N, the estimated time required to run a batch of cases can reach 6/N. There are currently 4 mac os operating systems in the team. After running a batch of cases, the original The 6 hours reduced to 6/4=1.5 hours, the more devices connected to the system, the shorter the time required to run a test, and the improvement of test efficiency and quality.  6. Improve the parsing rate of test cases. The original method of class loading consumes memory resources and performance, and it takes at least 1 minute. Now, the test case is obtained by parsing the class file through asm bytecode, and the parsing is completed in 5 seconds, and With the increase of test cases, the effect is more obvious.  Over the past year, I have demonstrated the following strengths:  1. When the requirements are changed and adjusted, confirm the requirements from the framework to the details, first determine the framework, wait for the overall plan to be finalized, and then confirm the details one by one.  2. When outputting the technical design plan, think clearly about the questions that may be asked, and record them in notes if you are afraid of forgetting them.  3. Excavate potential needs and actual needs: For the needs put forward by others, consider the reasons for this design, whether it is reasonable, and what he actually wants.  4. Minutes of the meeting, record the things discussed at the meeting and send them to everyone for second confirmation.  5. Way of thinking: From point-based thinking to linear thinking, and then to face-to-face thinking, that is to say, from emphasizing details, to the overall process, to the process of pros and cons analysis and selection among multiple solutions, you must have face-to-face thinking .  In the past year, I also have the following deficiencies:  1. In terms of technology improvement, I still need to further study and master some new technologies and frameworks to meet the company's future development needs.  2. In terms of teamwork, I still need to further improve my communication and coordination skills to better collaborate with other team members to complete tasks.  In order to improve my work performance, in the coming year, I will:  1. Strengthen the study of new technologies and frameworks, and constantly improve their technical level to better meet the company's future development needs.  2. Improve your communication and coordination skills, actively participate in teamwork, and cooperate with other team members to complete tasks.  3. Pay attention to personal time management, plan working time reasonably, and improve work efficiency and quality.  4. Strengthen the management of project progress and risks, discover and solve possible problems in time, and ensure that the project is carried out as planned.  5. Strengthen the management and evaluation of code quality, and ensure the quality and maintainability of code through code review and unit testing.  6. Actively participate in training and technical exchange activities, communicate and learn with other peers, and continuously improve their professional knowledge and skills.  Advice to teams and companies  1. It is recommended to clarify the work process and provide process training for everyone to ensure that everyone can work in strict accordance with the process. For example, it is necessary to jointly discuss and formulate the determination and completion time of requirements, and establish a clear functional acceptance process.  2. It is recommended to emphasize clear responsibilities, designate a clear responsible person for each project, and standardize the development process to ensure that the code of non-self-responsible projects will not be modified at will. At the same time, it is also necessary to abide by the development specifications to ensure the quality and readability of the code. | | | |