PARASOFT

Parasoft Jtest

Java开发人员生产力解决方案

借助AI驱动的自动化,更智能地工作

Parasoft Jtest通过深入的静态分析和AI驱动的自动化单元测试，加速Java软件开发，确保交付的软件可靠又安全且易于维护。在优化开发人员生产力的同时，最大限度地提高质量并降低业务风险。将Jtest无缝集成到您的IDE (IntelliJ、Eclipse或VS Code)中以支持开发人员的工作流程，或者集成到CI/CD管道中进行分析和报告。

使用静态代码分析快速分析Java代码的关键缺陷和安全漏洞。利用预防性(模式)和检测性(基于流的)静态分析技术，以及一套全面的代码度量指标。获得基于AI的建议，了解优先解决的违规问题以及应分配修复这些违规问题的开发人员。将Jtest与OpenAI/ Azure OpenAI提供程序集成，使用AI生成的修复措施加速对静态分析违规的补救。通过针对CERT、CWE、OWASP、PCI DSS等的合规性检查来验证代码的可靠性。自动生成合规性验证文档。

对新代码和修改后的代码进行高效的单元测试和集成测试，并自动生成强大的单元测试用例。Jtest的AI驱动单元测试助手自动生成一套优化、有意义、可维护的测试用例，包括模拟和断言。通过支持开发人员辅助的工作流，团队可以高效地构建和配置额外的单元测试来填补覆盖率缺口。利用Jtest的OpenAI/Azure OpenAI集成，根据用户提供的自然语言提示来分析和重构现有的JUnit测试，使用户能够根据自己的需求更轻松、更快速地修改单元测试。

通过在开发人员IDE中自动、持续地运行受代码更改影响的单元测试，提高开发人员的生产力和效率。Parasoft Jtest的集成Java开发测试解决方案帮助组织降低风险，降低成本，提高生产力，并实现行业合规目标。

全面且可配置的报告功能使开发人员和管理人员能够了解和优先处理代码库中检测到的错误。将CI/CD管道(GitLab、GitHub和Azure DevOps)中的静态分析和代码覆盖率导入IDE中。作为强大的Parasoft持续质量平台的一部分，与Parasoft DTP的集成使团队能够从所有测试实践中获得代码覆盖率的可见性，包括单元测试、API测试、UI测试和手动测试。此外，针对新代码和修改后的代码测量代码覆盖率，以确保即使总体代码覆盖率低也有足够的覆盖率。

提高开发人员的工作效率

- 分析代码，提高团队对代码质量的信心和安全性，并符合编码标准。
- 通过快速且易于使用的AI 驱动单元测试生成，加速敏捷开发并提高代码覆盖率。
- 高效地测试新的和遗留的代码，定位影响软件安全性、质量和可靠性的缺陷。
- 通过与CI/CD集成实现自动化回归测试，查看静态分析和覆盖率结果。
- 在IDE和CI/CD管道中通过测试影响分析获得对更改代码的即时反馈。

> 我们的开发人员花费在修复生产问题上的时间减少了，将更多的时间用于编写代码。除此之外，随着开发人员编写更多的测试，希望他们能在进入QA之前能更快地找到漏洞。 ——Fitch Solutions

优化 Java开发

申请DEMO，了解如何改进Java应用程序的安全性和质量。

自动化代码分析和合规性报告

通过静态代码分析和SAST将安全和质量缺陷的预防左移。Parasoft Jtest提供超越了开源解决方案的能力，全面审计代码，发现从API误用到安全漏洞的各种问题，帮助确保代码符合行业标准。

深度代码分析为Java提供了1000多条内置的Java静态分析规则，包括CERT、CWE、OWASP、PCI DSS和其他安全标准，支持对缺陷进行精确的检测，能够做到安全预防和暴露质量缺陷问题。

分析和优化代码

在不执行代码的情况下暴露bug。通过将静态代码分析直接集成到您的开发工作流中，提前满足合规性要求。使用流分析来模拟复杂的应用程序执行路径，并识别可能触发运行时缺陷的路径。对照定义的阈值分析代码度量，找出超出范围的值。

自定义规则

定义特定于组织的准则和编码标准。根据您的开发环境和实践，创建基于自定义模式的规则。

消除缺陷

通过完整的路径分析快速发现代码缺陷，以实现准确的违规检测。利用Jtest的AI生成的建议加速代码修复，直接在IDE中获得即时反馈。

利用AI优化问题修复

利用AI来识别优先级问题，并基于过去的分类操作自动将违规分配给最适合修复该违规的开发人员，从而加速静态分析违规的分类。使用DTP的CVE匹配建议，可以轻松判断安全漏洞是真是问题还是误报。通过遵循AI生成的建议，可以更快地解决违规问题。

测量和管理

通过集中报告和AI驱动的分析，团队可以对代码库中检测到的错误进行优先排序。

事半功倍的单元测试

通过简单的一键操作自动化测试创建，以创建和维护有意义的JUnit测试。Jtest的单元测试助手使用AI自动创建JUnit测试，并提供建议和快速修复，以简化新代码和修改代码的测试创建。Parasoft Jtest使开发人员能够优化验证其代码所需的时间，并有助于确保软件的每个部分都能按设计运行，并保持持续的质量。

生成有意义的JUnit测试

将AI和自动化引入单元测试的过程。借助实时、上下文感知的帮助，轻松创建、评估和增强JUnit测试，让开发人员可以专注于测试的业务逻辑。

批量创建单元测试

快速生成测试套件，通过批量创建来覆盖广泛的代码库，从而显著提高代码覆盖率。利用现有的JUnit测试套件，并通过测试用例克隆和自动变异来扩展它们，以覆盖更多的用例。

实现代码覆盖率目标

使用Jtest的开发人员辅助工作流进一步扩展测试套件，以实现更高的覆盖率。使用快速修复和批量操作来覆盖未测试的代码区域，增加代码覆盖率并启动您的测试套件。除了整体代码覆盖率之外，还要跟踪修改代码的覆盖率。

利用OpenAI/Azure OpenAI集成

通过利用Jtest与OpenAI/Azure OpenAI的集成，以及使用人工编写的自然语言提示，根据特定要求分析和重构现有测试，从而增强单元测试。

Parasoft Corporation
www.parasoftchina.cn
上海市黄浦区打浦路1号金玉兰广场1004室
邮箱：200023
Sales: (+86) 021 6093 2819
Email: info-china@parasoft.com