



Top 10 reasons OpenText Functional Testing for Developers is a must-have

Constantly changing applications, diverse technologies, and the need for swift, high-quality results when developing software can be daunting.

Unlock the power of seamless software delivery and discover how OpenText™ Functional Testing for Developers tackles these pain points head-on. Dive into the top ten reasons why this shift-left functional test automation tool is essential for DevTesters and engineers, accelerating productivity and ensuring code quality in every development cycle.

- 1 **Shift-left testing:** Accelerate development by testing simultaneously with application creation—enhancing collaboration and boosting productivity.
- 2 **Multi-language support:** Write test scripts in Java, JavaScript, or C#—ensuring flexibility and ease for developers using diverse languages.
- 3 **IDE integration:** Seamlessly integrate with popular IDEs like Visual Studio, Eclipse, and IntelliJ—enabling developers to work in their preferred environment.
- 4 **CI server compatibility:** Support for Jenkins, TeamCity, Bamboo, and Azure DevOps ensures effortless integration into continuous integration pipelines for streamlined workflows.
- 5 **Enhanced Selenium integration:** Expand Selenium’s power with OpenText™ Functional Testing for Developers API extension—facilitating quick and robust test script creation.
- 6 **Comprehensive reporting:** HTML run reports provide detailed insights into test execution, aiding developers in swiftly identifying errors and improving code quality.
- 7 **Service virtualization:** Save time and effort by simulating backend APIs and Virtual Services—enabling efficient test automation against virtualized backend services.
- 8 **Object identification tools:** Maintain consistency in testing models by adapting to changes in the underlying application—ensuring stability and reliability.

Highlights

- Boost productivity and accelerate delivery with shift-left test automation tools.
- Speed up test execution while improving code quality.
- Improve visibility of automated test execution with quality reports.
- Increase developer efficiency with a broad ecosystem of integrations.

9 **Parallel testing:** Run multiple tests concurrently on various environments, be it physical, virtual, or dockerized—ensuring efficient use of time and resources.

10 **Mobile testing integration:** Seamlessly integrate with UFT Mobile for end-to-end automation of mobile applications—providing developers with comprehensive mobile testing capabilities.

Learn more about [OpenText Functional Testing for Developers](#) ›

“With OpenText Functional Testing for Developers, we release higher quality software and deliver new functionality faster and more frequently. It gives us a great platform to expand our thinking around Agile processes and DevOps. We want to increase our application coverage and integration and have created a scalable test framework to achieve this.”

Fabrizio Sunna
Test Automation Business Lead
Helsana Group