GUI-Testtool for Java, Web, Android & native Windows Applications

	QF-TEST The Tool for Professional UI Test Automation	✓
Features		
GUI-technology of the application under test	**Java applications**: Swing, JavaFX, AWT, SWT, Eclipse Plug-Ins, RCP, Applets, JavaWebStart, RIA, ULC, CaptainCasa, JavaFX SubScene Components, JIDE Common Layer	
	Web applications: Browsers: Chrome, Firefox, Opera, Safari, Edge (Chromium based), Microsoft Edge Legacy, Internet Explorer; Headless Browser Versions of Chrome, Firefox and Edge (Chromium based) HTML 5, AJAX: QF-Test completely supports frameworks like Angular, React and Vue.js and also other UI-toolkits like GWT, Smart GWT, ExtGWT, ExtJS, ICEfaces, jQueryUI, jQueryEasyUI, KendoUI, PrimeFaces, Qooxdoo, RAP, RichFaces, Vaadin and ZK. Further toolkits can be integrated with little effort e.g. SAP UI5, Siebel Open UI and Salesforce. Testing of Electron and Webswing applications is supported as well.	
	Android apps: Native Android apps, mobile web apps, and hybrid apps on Android 7 and later on real devices and with the Emulator from Android Studio.	
	Windows applications: classic Win32, .NET (often developed with C#), Windows Presentation Foundation (WPF), Windows Forms, Windows Apps / Universal Windows Platform (UWP) with XAML control elements, modern C++ applications (e.g. with Qt)	
	Hybrid systems with the combination of multiple GUI technologies, as well as embedded browser components (JavaFX WebView, JXBrowser, SWT-Browser)	
	PDF documents can be tested like a normal application (textual and graphical checks for individual elements)	
GUI support depending on operating system	**Java applications:** **Swing and JavaFX**: Windows, Linux, Unix, macOS **SWT**: Windows, Linux-GTK; Solaris-GTK upon request	
	Web applications: Windows, Linux, macOS	
	Windows applications: Windows	
Testing principle	Capturing function (Capture/Replay) for direct and efficient generation of test sequences for further processing into more complex test cases with flow control, parameterization, modularization and extended scripting possibilities. Everything can be customized.	
Testing structure	Clear and concise thanks to graphical representation of test cases and action nodes in tree structure. Projects can be built up modularly through test suites and libraries.	
Component recognition, robustness of tests	Stable component recognition independent of geographical properties, as well as complex elements like dynamic trees and tables. Tests are robust and tolerant of changes in the GUI.	
Reusability, maintenance effort	High reusability of test building blocks through modular design e.g. through procedures, encapsulation of component accesses etc.	
Data driven testing	Direct import of CSV or Excel files, use of SQL database queries, XML files. Any other sources can be included via script extension.	

Keyword-driven testing/ Behavior-driven testing	Use of keywords for implementation and control of test cases, also through the means of external test specification documents or tools (e.g. Excel or test management tools)	
Load and performance testing	Load and performance testing through synchronized, parallel execution, even on multiple machines. For web in combination with tools like JMeter or NeoLoad.	
Docker containers	QF-Test supports containerized Testing via Docker	
Protocols, test documentation, reports	Clear, detailed logs incl. screenshots of the error situation are always generated. Configurable reports in various formats (HTML, XML, JUnit), test and procedure documentation can be generated at the push of a button or automatically.	
JIRA /REST	JIRA and JIRA PlugIns like TestRail, Zephyr, X-Ray, TM4J can be integrated through REST, often also with CI Tools like Jenkins.	
Extendability through scripting	Free function extendablity and customer specific checks/actions through embedded scripting (Jython, Groovy and JavaScript). Full access to all objects of the application (SUT) via scripting API and execution of own code in the application or browser.	
Test management	Basic functionality for smaller projects integrated in QF-Test. Integration available or possible for: ALM/QualityCenter by MicroFocus/HP, TestBench by Imbus, QMetry, Klaros by Verit, TestLink, IBM Rational Quality Manager, Jira and Jira PlugIns like TestRail, Zephyr, X-Ray, TM4J.	
Continuous integration	Flexible integration in build tools: i.e. Jenkins, GitLab CI/CD, Bamboo, Travis CI, Circle CI, TeamCity, Ant, Maven, CruiseControl	
Virtual desktops	i.e. Citrix, VMware, VirtualBox	
Version management	Good versioning capability e.g. via Git, SVN/subversion, CVS, Mercurial through XML format of the relevant files.	
Error tracking	Can be connected using open interfaces and REST, e.g. Jira, MantisBT, Bugzilla	
Robot Framework	Robot Framework keywords can be simply implemented and executed as QF test procedures.	
Test execution		
Test preparation	A quickstart assistend supports the generation of a suitable starting sequence for the application under test, depending on the underlying GUI technology.	
Test case preparation	Dependence management for test preparation and followup for independently executable test cases, including automatic error handling.	
Verification points	Direct capture of standard checks, customer specific verifications can be variably implemented through scripting.	
Image comparisons	Direct capture of image checks possible. Numerous algorithms also for blurred image comparisons - convenient Diff-View for checking in case of deviations.	
Object mapping	Component information is saved centrally during capture in a separate area in the testsuite, can be edited anytime. Mechanisms for reference search and automatic update.	
Intelligent Object Location	With SmartIDs, components can be addressed directly on the basis of characteristic properties, i. e. the associated label. Setting a scope allows to restrict the object search, e.g. in case of multiple elements.	
Generic components	The mapping of GUI specific components into generic components (buttons, text fields...) allows the reuse of tests across technologies as well as the use of generally valid actions without having to capture every single component.	
Test execution via command line	Execution in batch mode possible with extensive configuration options via command line parameters, also for integration in build environments.	
Remote execution	Test execution also possible on remote computers in Daemon mode.	

Error handling	Automatic error handling guarantees continuation of the overall test run without interruption. Errors are protocoled for later analysis.	
Test debugger	Full debugger functionality including break points and variable analysis.	
User-friendliness		
Ease of use	Simple and intuitive use with clearly structured for convenient editing of test cases, e.g. copy/paste and drag&grop; capture/replay for a quick start.	
Required previous knowledge	No programming knowledge necessary for standard use For scripting knowledge of the standard scripting languages Jython, Groovy and JavaScript useful.	
Teamwork	Modular structure of test suites and imported libraries in conjunction with version management allows efficient teamwork.	
Licenses and prices		
Product variants	QF-Test is available variably configurable in different product variants for the supported GUI technologies (Swing, JavaFX, SWT, Web and Windows) - purchase only what you need.	
Lincense	**Developer license -** to create (and execute) test cases and **Runtime-lincense -** to execute (nightly) tests.	
Lincense mechanism	All licenses are floating (i.e. not bound to one specific person). Standard licenses work within a (local) network; for cross-network use we offer a license server.	
Purchase or rent	We offer a purchase option and a yearly rent, for load test licenses shorter terms are also available.	
Support and service		
Download and free testing	Free demo version anonymous with the download. Demo version runs without license registration. Free test license for 4 weeks including free support.	
Installation	Easy installation on Windows, macOs and Linux with just a few clicks.	
Support	Direct support from QF-Test developers and testers in German, English or French.	
Maintenance contract	Maintenance contract (support + upgrades) included on a yearly base Email or phone support available	
Training, consulting	Individual training or consulting at your site. Regular open standard training at QFS. All offers also possible online.	
Documentation	Extensive manual, self start tutorial, videos, blog, online search function, online help with a right click in QF-Test, general and technical FAQ.	