

Software Product Information

Fabasoft app.test 2017 June Release

Valid from June 25, 2017

Fabasoft[®]

Copyright ©

Fabasoft R&D GmbH, Linz, Austria, 2017.

All rights reserved. All hardware and software names used are registered trade names and/or registered trademarks of the respective manufacturers.

No rights to our software or our professional services, or results of our professional services, or other protected rights can be based on the handing over and presentation of these documents.

Contents

1 Fabasoft app.test	4
1.1 Product Editions	4
1.2 Compare Editions	4
2 Fabasoft app.test Studio	5
2.1 Test Recorder	5
2.2 Point and Click Recorder	6
2.3 Multi-Browser Support	7
2.4 Dialog and Script Error Handler	7
2.5 Control Specification Designer	8
2.6 Test Player	9
2.7 Internationalization of Tests	9
3 Console Player	9
4 Continuous Integration	10
4.1 Commander	10
4.2 Remote Agent	10
4.3 Ant Tasks	10
4.4 Infrastructure Manager	11
5 Reporting	11
5.1 HTML Reporting	11
5.2 BIRT Reporting Support	12
5.3 Database Reporting	13
6 Supported Platforms and Minimum System Requirements	14
6.1 Microsoft Windows Environment	14
6.2 Linux Environment	14
6.3 Apple OS X Environment	15
6.4 Additional Conditions	15
6.5 Supported Editions of Third-Party Products	15
6.6 Minimum Requirements	16

1 Fabasoft app.test

Fabasoft app.test is a web and business app testing tool for the quality assurance of applications: whether on the web, on a hard disk or in the cloud, app.test generates reports together with a screenshot of each problem that arises.

- It's easy to create and expand your own tests.
- The reports are permanently available and completely tailored to individual requirements.
- In addition, Fabasoft app.test integrates seamlessly into your continuous integration system.

Fabasoft app.test detects programming errors – ensuring your developments shine in front of your customers.

1.1 Product Editions

Fabasoft app.test is available in three product editions.

Fabasoft app.test primo is available for free providing a limited set of features and community support.

Fabasoft app.test premium is a commercial edition providing a comprehensive set of features and enhanced support services.

Fabasoft app.test data center is a commercial edition providing the whole set of features and full priority support services. It is focused on managing test resources in data centers and testing Fabasoft Folio and Fabasoft eGov-Suite.

1.2 Compare Editions

The product capabilities are bundled by product editions.

Fabasoft app.test Feature	Fabasoft app.test primo	Fabasoft app.test premium	Fabasoft app.test data center
Fabasoft app.test Studio	X ⁽¹⁾	X	
Test Recorder	X	X	
Point and Click Recorder	X	X	
Multi-Browser Support	X ⁽²⁾	X	
Dialog and Script Error Handler	X	X	
Control Specification Designer	X	X	
Internationalization of Tests	X	X	
Test Player	X	X	
Console Player		X	
Continuous Integration			

Commander	X	X	X
Remote Agent		X	
Ant Tasks			X
Infrastructure Manager			X
Reporting			
HTML Reporting	X	X	X
BIRT Reporting Support		X	X
Database Reporting			X

- (1) Max. 1 session/client
 Max. 500 executions/file,
 Max. 10 test files (Test, Use Case, Sequence)/project
 (2) Free for the latest supported Mozilla Firefox

2 Fabasoft app.test Studio

At a glance

- Record, manage and analyze tests
- Lay the foundation for high quality website tests
- Prepare the automated test execution
- Analyze, improve and debug tests

Fabasoft app.test Studio is the tool to manage software quality, providing the functionality to manage the full test cycle: managing test projects, recording and enhancing tests, analyzing reports and preparing full automation and CI integration.

No matter how complex test projects are, you are always in control. A simple click on the error message changes the test player to a test recorder so that you can immediately analyze and solve problems. Debugging tests has never been easier.

Fabasoft app.test comes with a full set of reports. Furthermore it comes with the ready to use BIRT report designer.

2.1 Test Recorder

At a glance

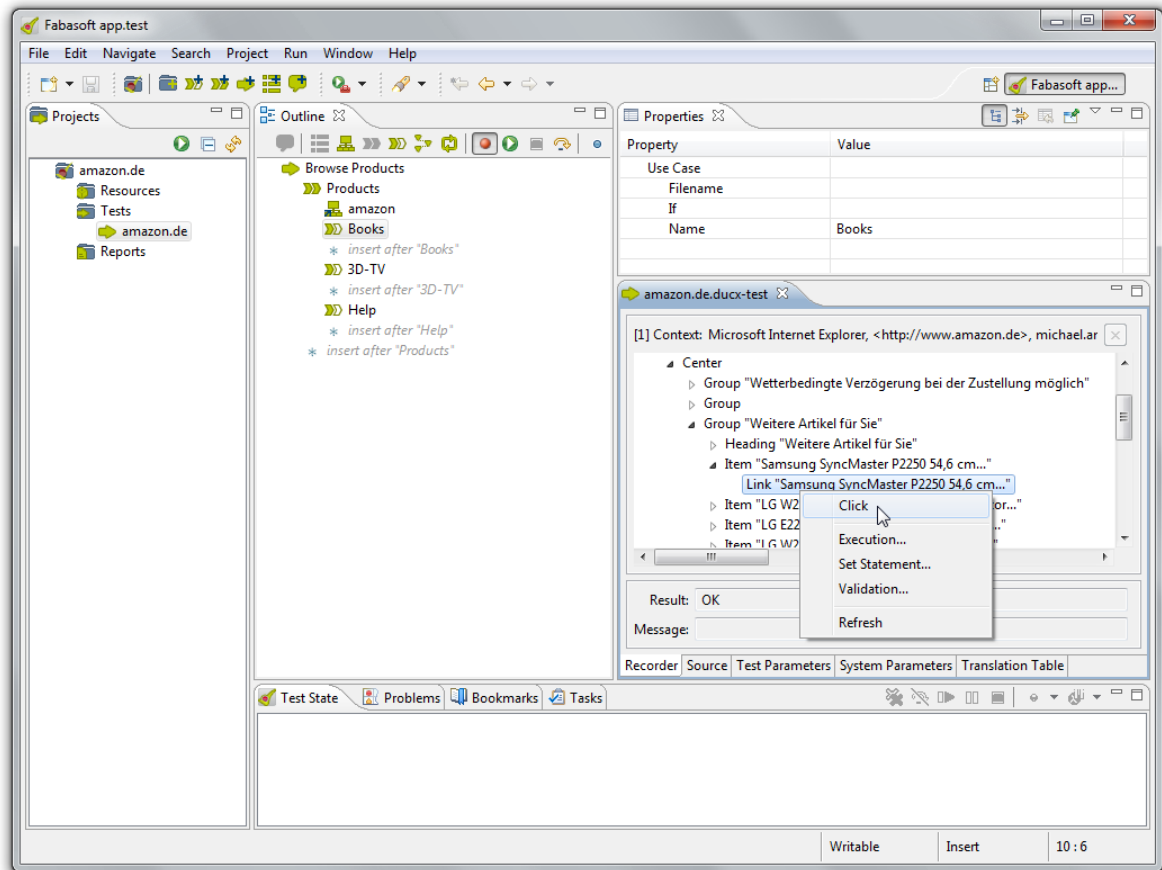
- Record test steps
- Insert statements to assign values
- Perform validations

Fabasoft app.test Recorder provides the tool to create valuable and effective tests. Based on the tree view of a web page or an application, highly specific actions can be performed. Parameters, validations as well as translations can be integrated and managed.

With the Test Recorder you can record tests by using the “Control Tree”. The tree view of the available controls (boxes, buttons, links, ...) allows performing and recording actions, statements as

well as validations. By utilizing the context menu in the tree you create tests by seamlessly inserting test statements. The steps are performed and recorded immediately.

With advanced options like “if statements”, “timeout” definitions or “waits” you can enhance your test executions exactly as you need them. Inserting “Set” and “Validation statements” helps you tune your tests and bring them up to a higher level of quality.



Recording a Click using the Test Recorder

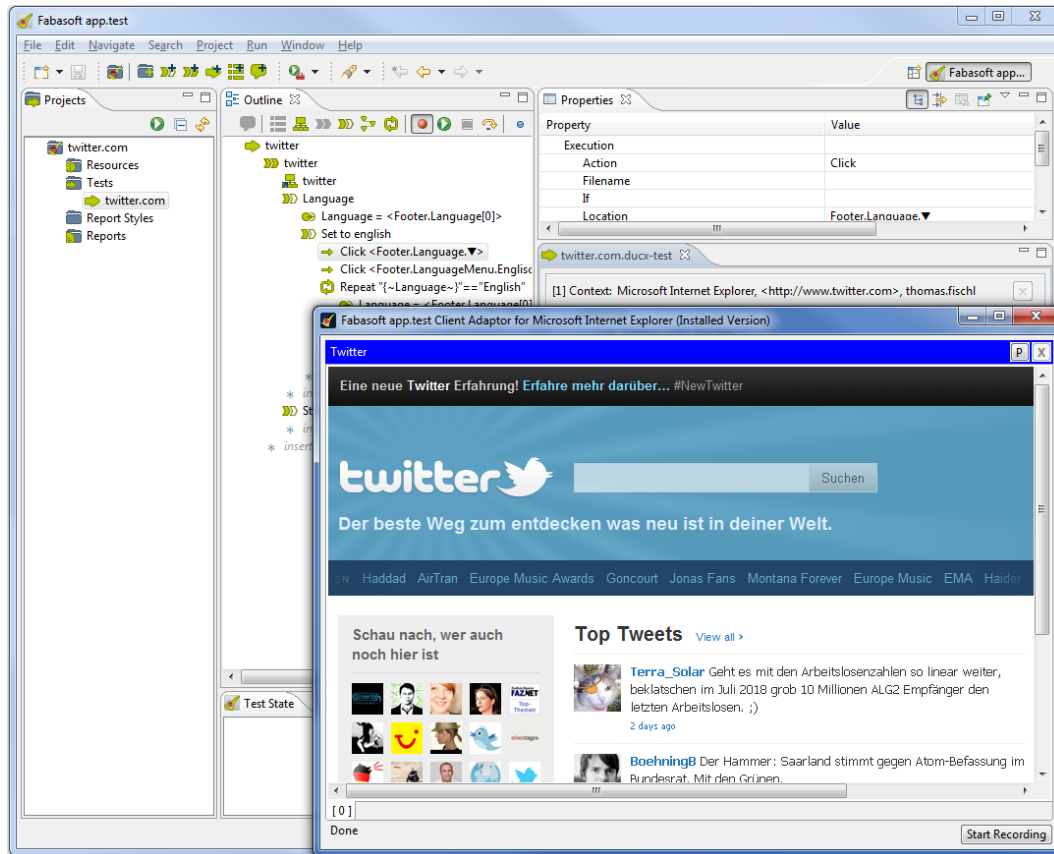
2.2 Point and Click Recorder

At a glance

- Record test steps while working with a web application
- Allow others to record tests
- Insert test statements into an existing test
- Get to know the structure of a website by exploring the recorded statements

The Point and Click Recorder makes recording tests as easy as working with an application. This fast and comfortable way to record tests allows anyone to create tests without prior knowledge of a testing application. Tests are recorded while developing or clicking through a website.

The tested website or web application is embedded in the Point and Click Recorder window. Right after hitting the “Start Recording” button, actions are recorded. As we want to ensure that tests survive changes on the user interface, they are recorded in an understandable way. Everything you do is recorded and displayed as an execution in the Test Recorder.



Recording a Click using the Point and Click Recorder

2.3 Multi-Browser Support

At a glance

- Running tests with Microsoft Internet Explorer on Microsoft Windows Systems
- Performing the same test with several versions of Mozilla Firefox without installing them
- Using the same tests for each browser

Web applications are used with several browsers and platforms. Therefore we need to make sure that the tests are executed on several browsers and platforms.

The variety of web browsers increases steadily. As web applications should be accessible through various web browsers and on different platforms, we need to ensure that the application works properly in any environment. Therefore Fabasoft app.test supports many commonly used web browsers and operating systems (see chapter 6 Supported Platforms and Minimum System Requirements). When testing with Mozilla Firefox, no installation of any version is required – tests can be performed immediately.

Note: The included Mozilla Firefox versions correspond to the respective state of development of the web browser and may contain vulnerabilities. Therefore test only trusted sites.

2.4 Dialog and Script Error Handler

At a glance

- Identification and reporting of Script Errors

- Handling and reporting of all arriving dialogs and message boxes

Dialogs and script errors can be found in many websites. Script errors or other interfering elements can block the execution of tests. Dialogs especially are often used by web application developers for debugging purposes or for passing information to the user. These windows are partly modal, and this would otherwise result in interruptions to the test. Fabasoft app.test, however, identifies those windows or script errors and handles them to ensure that the test continues without interruption.

All dialog and script errors are documented in the reports.

2.5 Control Specification Designer

At a glance

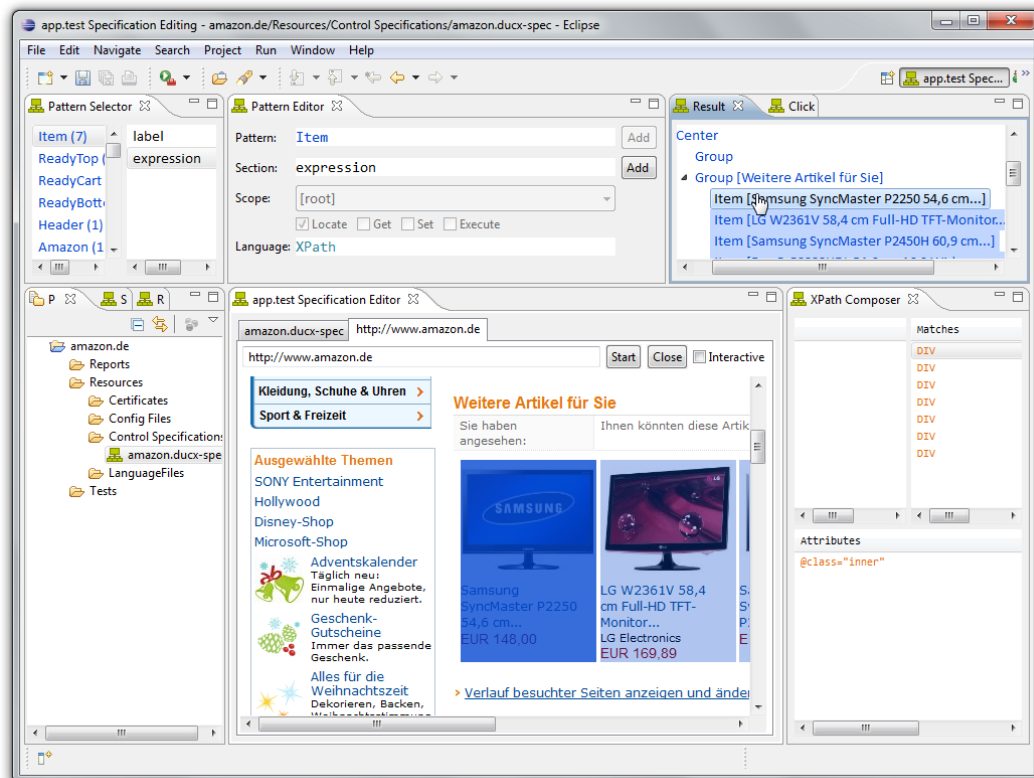
- Point-and-click IDE to provide reusable tests for any web page
- Integrated editor for designing XPath expressions to locate elements

The Control Specification Designer provides an easy way to make a website testable. Even complex website structures can be described by clicking through, making web tests understandable for everybody.

Making a web application testable is a matter of clicking. Simply by “showing” the editor the parts of the website that belong together, so called “patterns” are generated. By applying them throughout the websites’ structure, tests become readable, understandable and reusable. The specification file can be integrated into the tests immediately.

When recording a test with Fabasoft app.test, the specification file is used and all actions are performed for the “understandable and readable parts”. This way complex website tests become humanly accessible.

Note: The Control Specification Designer is only available for Microsoft Windows and Microsoft Internet Explorer.



2.6 Test Player

At a glance

- Running tests unattended
- Executing tests on several platforms
- Seamlessly switching from test to debugging mode to improve tests

The Fabasoft app.test Player allows for tests to be run in an unattended environment. It also allows running the same test on many platforms with various browsers on a regular basis. When using the Test Player in Fabasoft app.test Studio, immediate debugging of executed tests is possible to solve any problems.

In the Test Player recorded or imported tests can be played. If tests need to be modified, e.g. in the case of an error, it is possible to switch seamlessly to the Test Recorder and refine your test. Afterwards you can switch back to the Test Player and continue playing the test or restart it. You could call it a kind of “debugging” for tests.

2.7 Internationalization of Tests

At a glance

- Running the same test in a multilingual environment
- No need to update tests after translating applications to other languages

Multilingual environments often cause duplication of acceptance tests when operating on the GUI. Fabasoft app.test provides the solution by providing translation files with multilingual strings. When recording tests, these strings are examined and statements are automatically recorded, independent of language.

Recording acceptance tests on the graphical user interface takes place in one language, but applications often support different languages. Therefore tests can be recorded in one language and executed in another. This provides a way to test international websites or applications in multiple languages without having to record tests for each language.

3 Console Player

At a glance

- Starting tests from command line
- Running tests in batch mode with threads
- Generating reports in various formats

Running tests is just a matter of calling a batch file. With Fabasoft app.test Console Player you can execute tests on clients by simply calling the Console Player and passing the required information. This is how to integrate automated acceptance testing into a build cycle.

With the Console Player, testing websites or web applications on different platforms (e.g. Microsoft Windows or Linux) can be automated. The Console Player offers the possibility of starting tests on different physical clients directly from the command line. Tests can be started in a certain order and executed with different users.

4 Continuous Integration

4.1 Commander

At a glance

- Running preparation tasks on machine instances
- Setting up remote applications
- Copying and extracting files
- Synchronizing workspace between machine instances

The Fabasoft app.test Commander is the tool to prepare a complete Continuous Integration environment. This includes setting up applications, performing administrative tasks, transferring files between instances and synchronizing files.

By using the Fabasoft app.test Commander installing applications, configuring systems or synchronizing files between machine instances happens in just the same way as testing. By using the same Domain-Specific Language (DSL) as when recording a test, you can transfer files between systems or execute applications. These tasks can be integrated into an Acceptance Test to dynamically prepare the environment as required in the test. To do so, operating system commands like `move`, `delete` or `copy` can be executed, Ant-Scripts can be called and the workspace can be synchronized.

4.2 Remote Agent

At a glance

- Running tests on distributed environments without installing Fabasoft app.test Studio
- Automated update prior to starting tests

The Remote Agent allows running parallel tests on different hosts. Whether the client is a virtual machine or physical computer with either Microsoft Windows or Linux installed, tests can be instantly executed.

The Fabasoft app.test Agent provides the necessary tool to run the same tests in parallel using different platform and web browser combinations (see chapter 6 Supported Platforms and Minimum System Requirements). For remote testing purposes all that is required is to set up the Fabasoft app.test Agent on the computer where you want to execute or record the test. To start the tests remotely you can use the Console Player.

4.3 Ant Tasks

At a glance

- Seamless integration of Fabasoft app.test into Ant based build scripts
- Loading data into the database after running tests

With Ant Tasks Fabasoft app.test can be directly integrated into the build process. Tests can be executed in parallel on multiple computers with various users.

Ant Tasks are used to run tests based on the Console Player. By using Ant Tasks you can immediately integrate any tests into the build cycle. After running tests by specifying a file set you can summarize test reports into one single report very simply. With a specified stylesheet, e.g. XSLT or BIRT, Ant Tasks can be used to transform and send out reports via e-mail.

4.4 Infrastructure Manager

At a glance

- Preparation of virtual machines (VMs) before starting tests
- Balancing tests over many machine instances
- Executing application setups to prepare servers and clients

The Fabasoft app.test Infrastructure Manager provides a set of tools to automate recurring tasks prior to starting them. This includes dynamically requesting resources (VMs) based on specified requirements, cloning them, running setups, copying files and finally cleaning up.

With the Infrastructure Manager test resources in the data centers can be managed. Whether you need clients or servers for testing, they can be requested dynamically. Even preparing the infrastructure elements by performing setups, copying files or running scripts is possible. The infrastructure manager takes care of balancing requested resources over many physical machines. When requesting infrastructure, it is provided based on the specified requirements.

These requirements could look like

- I need 2 instances with CentOS
- I need 2 clients with Microsoft Windows 7
- I need 1 client with Microsoft Windows 8.1, Microsoft Internet Explorer 11 and Microsoft Office 2013 installed

The infrastructure manager takes care of the provisioning of virtual machine instances based on your templates as well as cleaning up after the job is finished.

5 Reporting

5.1 HTML Reporting

At a glance

- Transforming test reports into readable HTML
- Generating overview and detail reports
- Transforming reports into a JUnit compliant XML format to be used in a Continuous Integration (CI) environment like Hudson
- Transforming reports to match your corporate design by specifying XSL files

With HTML Reporting it is easy to analyze the results of your tests. Clearly structured, including screenshots and used test parameters, they provide every scrap of information - as detailed as you wish- to evaluate your tests.

Reports are available for performing HTML overview and detail reporting as well as XML JUnit reporting. By modifying the XSL files you can easily change the appearance of the reports to match your corporate design requirements.

When running complex tests a report summary is available also. This summary combines the results of all test reports and generates one common report, which contains links to the detail reports and provides a quick overview over the executed test.

Test:	EditionFolio_21_Outgoing_and_E-Mail_1_Main [Run: 1]
Project:	
Start time:	2010-11-22 15:10:33
End time:	2010-11-22 15:44:22
Duration:	00:33:48.648
Client Environment:	192.168.107.158
Web Server Environment:	192.168.104.239
Version:	FSC 10.3.6.47

	Total	Ok	Warning	Not ok	Not Executed
Sequences:	2	1		1	
Use Cases:	19	16		1	2
Context switches:	8	5		3	
Repeats:	10	10			
Executions:	623	582		36	5
Set-Statements:	95	90		3	2
Validation:	10	8		2	

Name	Time	Result
ParameterTable	Date: 2010-11-22 Start: 15:10:33 End: 15:10:33 Duration: 00:00:00.422	Ok
Set parameter token value 0003 2010-11-22 [15 10 33] orig value {~scope~} {~timestamp~} line 3	Date: 2010-11-22 Start: 15:10:33 End: 15:10:33 Duration: 00:00:00.016	Ok
Set parameter swctoken value 0003_20101122_151033	Date: 2010-11-22	

HTML Report

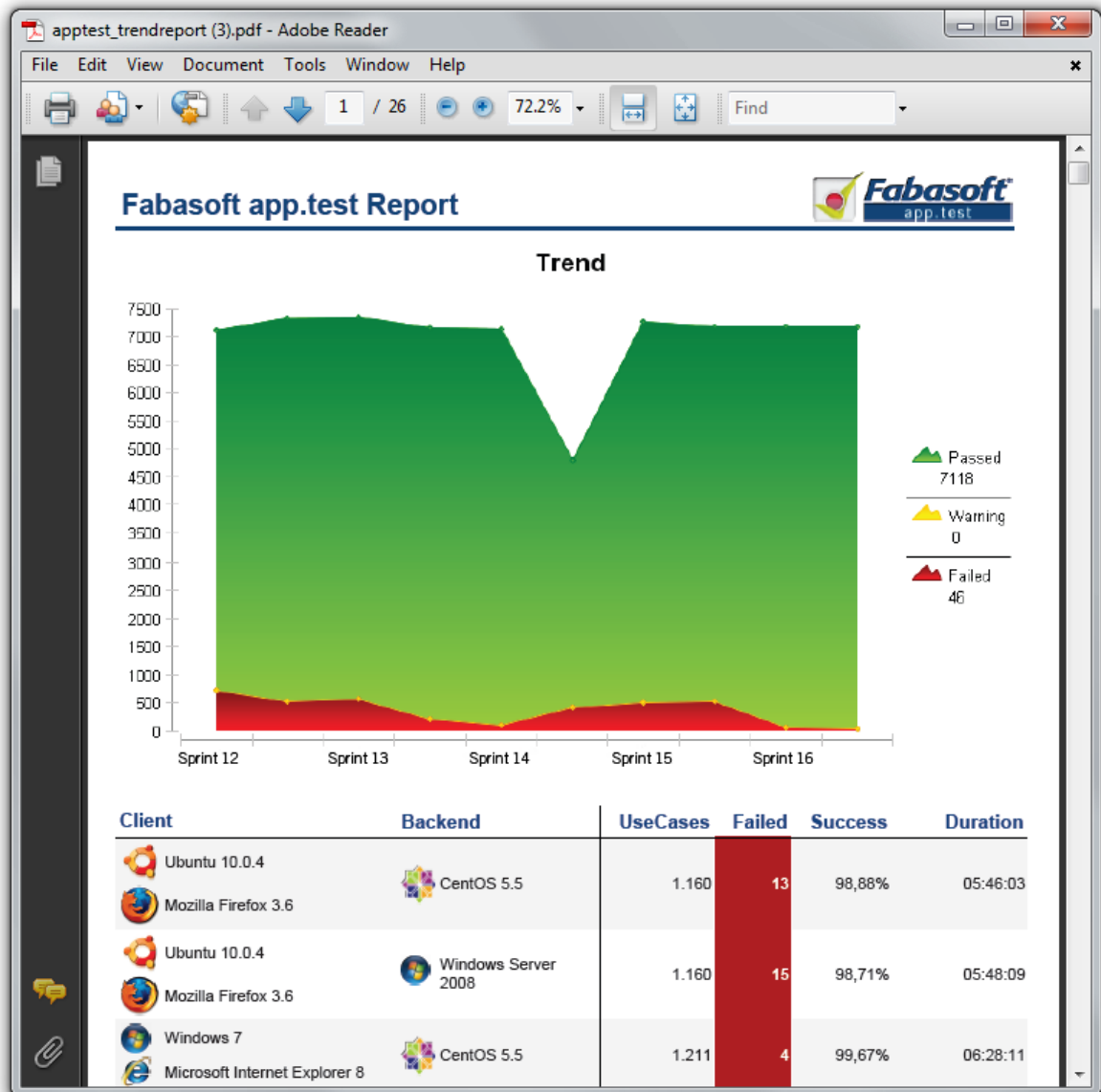
5.2 BIRT Reporting Support

At a glance

- Transforming reports into PDF, HTML, Microsoft Word or Microsoft Excel by using BIRT
- Designing individual reports for visualizing test trends or other key indicators

Reports are the most important way to visualize test results. Therefore Fabasoft app.test provides out-of-the box reports as well as predefined templates for BIRT.

Eclipse BIRT provides a great way to design reports for various formats. This includes HTML as well as PDF, Microsoft Excel, Microsoft PowerPoint or Microsoft Word. They can easily be designed in Fabasoft app.test and seamlessly integrated into the test cycle.



Trend Report created using BIRT

5.3 Database Reporting

At a glance

- Loading report data to relational databases
- Using Microsoft SQL-Server or PostgreSQL databases
- Merging the infrastructure with the test results to enhance information detail

Analyzing test results from various perspectives is supported by Fabasoft app.test Database Reporting. This allows for loading test results into a relational database. All database tables are created automatically.

Databases are the right way to generate key performance indicators. This allows for the filtering of report data based on various conditions. Questions like: "Is the system faster on a server with Microsoft Windows or Red Hat Enterprise Linux" as well as "How many executions are slower than 5 seconds" can be answered.

6 Supported Platforms and Minimum System Requirements

When using Fabasoft app.test 2017 for automatic testing of Fabasoft Folio or Fabasoft eGov-Suite following versions are supported:

- Fabasoft Folio 2012 to 2016
- Versions of Fabasoft eGov-Suite that are based on the above mentioned versions of Fabasoft Folio

6.1 Microsoft Windows Environment

Fabasoft app.test Studio and Fabasoft app.test Agent

Operating System	Architecture
Microsoft Windows 7 Enterprise SP1 or Microsoft Windows 8.1 Enterprise or Microsoft Windows 10 Enterprise	32 bit (x86) oder 64 bit (x64) 32 bit (x86) oder 64 bit (x64) 32 bit (x86) oder 64 bit (x64)
Software Requirements	
Oracle Java SE Runtime Environment 8 Update 131 (JRE) (x86)	
Microsoft Internet Explorer 11.0 (x86)	

Running Tests With Apache Ant
Apache Ant 1.9.4

6.2 Linux Environment

Fabasoft app.test Agent

Operating System	Architecture
Ubuntu 17.04	32 bit (x86)
Software Requirements	
Oracle Java SE Runtime Environment 8 Update 131 (JRE)	

Running Tests With Apache Ant

Apache Ant 1.9.4

6.3 Apple OS X Environment

Fabasoftware app.test Studio and Fabasoftware app.test Agent

Operating System	Architecture
Apple macOS Sierra 10.12	64 bit (x64)
Software Requirements	
Oracle Java SE Runtime Environment 8 Update 131 (JRE)	
Apple Safari 10.1	

Running Tests With Apache Ant
Apache Ant 1.9.4

6.4 Additional Conditions

- The minimum hardware and software requirements for any future versions of this product may differ from the requirements of this version.
- A certain version of a third-party product integrated into any Fabasoftware software product is supported by Fabasoftware as long as the software product itself is supported by the third-party itself. For instance a particular version of an integrated Microsoft software product is only supported by Fabasoftware until the date on which the Microsoft “Extended Support Phase” for this product version ends.
- Third-party products that are required or supported by Fabasoftware products, proper licensing and installation of them, necessary tests for product release as well as corresponding manufacturer support are not included in the scope of supply and services and therefore they are not subject to warranty of Fabasoftware regarding functionality, mode of operation or features. Fabasoftware is not responsible for errors or malfunction, which are caused by third-party software products and/or products that are not included in the scope of supply and services.
- Fabasoftware software products are only supported and tested on the mentioned reference system environments. In addition to the third-party product editions listed in the reference system environments the editions listed in chapter “Supported Editions of Third-Party Products” are supported but not tested.

6.5 Supported Editions of Third-Party Products

Following editions of third-party products are supported. Explicitly tested are only those editions that are listed in the reference system environments.

- The third-party product Microsoft Windows 7 SP1 is supported in the following editions: Enterprise and Ultimate.

- The third-party product Microsoft Windows 8.1 is supported in the following editions: Pro and Enterprise.
- The third-party product Microsoft Windows 10 is supported in the following editions: Pro and Enterprise.

6.6 Minimum Requirements

Architecture	CPU	RAM	Disk Space
32 bit (x86)	1,6 GHz	1 GB	1 GB
64 bit (x64)	1,6 GHz	1 GB	1 GB