

# Installation Checklist

**Squire 20.1.11**

Last updated 2022-02-10

# Table of Contents

Preface .....	1
Foreword.....	1
Licence.....	1
Warranty .....	1
Responsibilities .....	2
Contacting Vector Informatik GmbH Product Support.....	2
Getting the Latest Version of this Manual .....	2
1. Introduction .....	3
The Squire Ecosystem .....	3
Frequently Asked Questions .....	4
2. Preparing your Machine for Squire .....	9
Installation Prerequisites .....	9
Supported Operating Systems .....	9
For All Systems.....	9
Packages for Windows.....	10
Packages for CentOS and Red Hat Enterprise Linux.....	10
Packages for Ubuntu.....	10
Packages for other Linux distributions.....	11
Third-Party Plugins and Applications .....	11
Requirements for LDAP Integration .....	11
Requirements for TeamForge Integration .....	12
Appendix A: Licences .....	13
Software Licence Agreement.....	13
Redistributed Software.....	19
Index.....	21

# Preface

© 2021 Vector Informatik GmbH - All rights reserved - <https://www.vector.com/> - This material may not be reproduced, displayed, modified or distributed without the express prior written permission of the copyright holder. Squire is protected by an Interdeposit Certification registered with Agence pour la Protection des Programmes under the Inter Deposit Digital Number IDDN.FR.001.390035.001.S.P.2013.000.10600.

## Foreword

This edition of the Installation Checklist was released by Vector Informatik GmbH.

It is part of the user documentation of the Squire software product edited and distributed by Vector Informatik GmbH.

For information on how to use and configure Squire, the full suite of manuals includes:

User Manual	Target Audience
<a href="#">Squire Installation Checklist</a>	New users before their first installation
<a href="#">Squire Installation and Administration Guide</a>	IT personnel and Squire administrators
<a href="#">Squire Getting Started Guide</a>	End users, new users wanting to discover Squire features
<a href="#">Squire Command Line Interface</a>	Continuous Integration Managers
<a href="#">Squire Configuration Guide</a>	Squire configuration maintainers, Quality Assurance personnel
<a href="#">Squire Eclipse Plugin Guide</a>	Eclipse IDE users
<a href="#">Squire Reference Manual</a>	End Users, Squire configuration maintainers
<a href="#">Squire API Guide</a>	End Users, Continuous Integration Managers
<a href="#">Squire Software Analytics Handbook</a>	End Users, Quality Assurance personnel



You can also use the online help from any page when using the Squire web interface by clicking **? > Help**.

## Licence

No part of this publication may be reproduced, transmitted, stored in a retrieval system, nor translated into any human or computer language, in any form or by any means, electronic, mechanical, magnetic, optical, chemical, manual or otherwise, without the prior written permission of the copyright owner, Vector Informatik GmbH. Vector Informatik GmbH reserves the right to revise this publication and to make changes from time to time without obligation to notify authorised users of such changes. Consult Vector Informatik GmbH to determine whether any such changes have been made. The terms and conditions governing the licensing of Vector Informatik GmbH software consist solely of those set forth in the written contracts between Vector Informatik GmbH and its customers. All third-party products are trademarks or registered trademarks of their respective companies.

## Warranty

Vector Informatik GmbH makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Vector Informatik GmbH shall not be liable for errors contained herein nor for incidental or consequential damages in connection with the furnishing, performance or use of this material.

This edition of the Installation Checklist applies to Squire 20.1.11 and to all subsequent releases and modifications until otherwise indicated in new editions.

## Responsibilities

Approval of this version of the document and any further updates are the responsibility of Vector Informatik GmbH.

## Contacting Vector Informatik GmbH Product Support

If the information provided in this manual is erroneous or inaccurate, or if you encounter problems during your installation, contact Vector Informatik GmbH Product Support: <https://portal.vector.com/>

You will need a valid customer account to submit a support request. You can create an account on the support website if you do not have one already.

For any communication:

- **support@vector.com**
- **Vector Informatik GmbH Product Support**

Vector Informatik GmbH - Holderäckerstr. 36 / 70499 Stuttgart - Germany

## Getting the Latest Version of this Manual

The version of this manual included in your Squire installation may have been updated. If you would like to check for updated user guides, consult the Vector Informatik GmbH documentation site to consult or download the latest Squire manuals at <https://support.squoring.com/documentation/latest>. Manuals are constantly updated and published as soon as they are available.

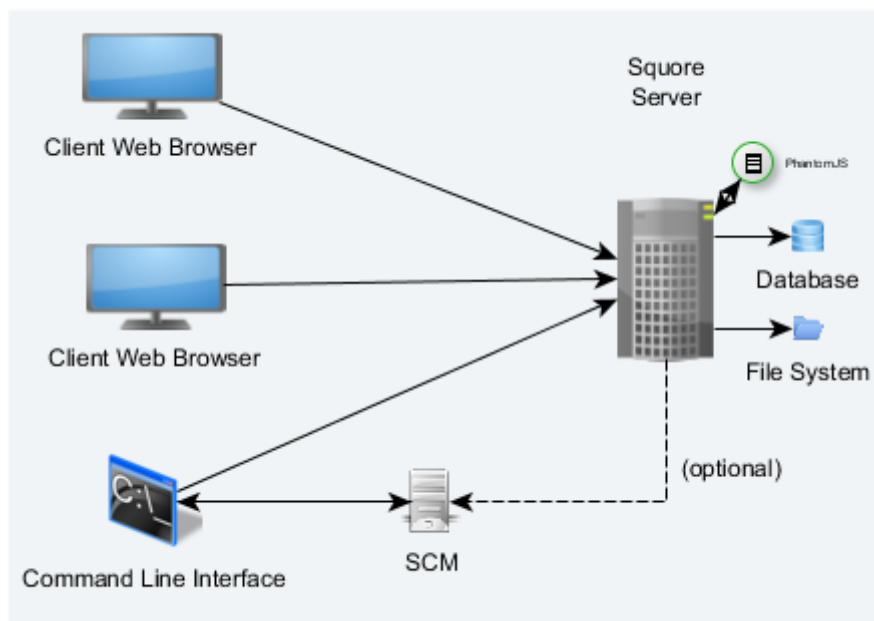
# Chapter 1. Introduction

This document is a checklist and FAQ sheet you can use to make sure that your environment is ready for installing Squire. The first chapter covers common pre-requisites and concepts of the Squire architecture. Each of the following chapters covers more specific topics that you only need to read about if you are interested in using the proposed solutions.

## The Squire Ecosystem

Squire is based on a traditional 3-tier architecture consisting of:

- A database and a data folder for storing project data
- An application server running the main application, the licence server and a distribution of PhantomJS
- Clients that can interact with the server: a front-end accessible through a Web Browser and a Command Line Interface (Squire CLI) to interact with the server from a console



*The Squire Architecture*

As shown in the schema above, Squire Server can provide analysis results to clients without having access to any source code, in scenarios where the analysis is carried out on a client machine with access to the SCM repository, as is the case in most Continuous Integration environments.

If you are planning to access source code hosted in Subversion, Git, ClearCase, CVS, Integrity, Perforce, TFS or Synergy repositories, a command line client for this repository must be available on the machine where the Squire analysis is carried out. For complete information about all installation pre-requisites, consult [Installation Prerequisites](#).

Squire allows analysing source code in the following programming languages: ABAP, Ada, C, C++, MindC, C#, COBOL, Java, JavaScript, Fortran 77, Fortran 90, Objective-C, PHP, PL/SQL, Python, T-SQL, Visual Basic .NET, XAML, Lustre.

# Frequently Asked Questions

The following is a list of frequently asked questions when considering a Squire installation.

## Is the Squire solution self-hosted? Does it require internet access?

Squire can be installed within your network and does not require internet access.

## How does licencing work?

A Squire Server installation contains a licence file which defines how many active users are allowed to use the software. The licence is specific to the server machine's hardware and will have to be changed for a new one if the installation moves to a new environment.

It is not necessary to identify the machines where clients will run ahead of time.

You can find a complete description of Squire licencing in the Installation and Administration Guide in the section called [The Squire Licence Server](#).

## What languages does the source code parser support?

The following languages are available by default: ABAP, Ada, C, C++, MindC, C#, COBOL, Java, JavaScript, Fortran 77, Fortran 90, Objective-C, PHP, PL/SQL, Python, T-SQL, Visual Basic .NET, XAML, Lustre

You can add basic support for additional languages by consulting the Configuration Guide's [Adding More Languages to Squan Sources](#) section.

## How can I import results from a tool that is not available in Squire?

If you have results from a tool not currently supported by Squire's Software Analytics analysis model, our open Data Provider framework lets you write more plugins so you can import your results. Consult the Configuration Guide's [Creating a form.xml for your own Data Providers, Repository Connectors and Export Definitions](#) section for more information.

## How often do you release new versions?

Squire releases use the following naming pattern: *<major>.<minor>.<patch>*.

A new major release of Squire comes out every year with new features and an improved analysis model. One or two minor releases may also come out during the year to bring new functionality without impacting the the analysis model used in the existing major version.

Patch releases for maintained versions are released usually every 4 to 8 weeks to fix issues reported by customers. These patch releases are fully compatible with the major or minor release they bring fixes for. They do not affect your existing analysis results, unless explicitly stated in our release notes. A patch release never contains new features or functionality, only bug fixes.



You can keep track of release announcements by following our **Release** space on the support wiki at <https://openwiki.squaring.com>.

## How often should I upgrade?

If you install a production server that runs a major or minor release, it is recommended to upgrade regularly to the latest patch release available for this Squire version to benefit from the most recent bug fixes.

Before upgrading to a new minor or major release, we recommend replicating your environment to test the migration. While our releases are always backwards-compatible, some of your analysis

results may be impacted by changes and improvements to the analysis model in new major or minor versions.

## How easy is it to upgrade my Squore installation?

Our upgrade script or installer takes care of deploying the new version of the software and migrating your existing data and configuration if needed in one simple process.



Before upgrading your Squore installation, always check the release notes on our support wiki at <https://openwiki.squoring.com>.

Upgrade instructions can be found in the Installation and Administration Guide, in the section called [Upgrading from a Previous Version](#).

## What documentation is available?

A list of all existing user manuals and their target audience is available in the [Preface](#).

Online help is also available on every page of the Squore web interface.

If you are new to Squore, you should probably start with the [Installation and Administration Guide](#) to install Squore and move on to [Getting Started Guide](#) to discover all the features that Squore has to offer.

## What information does your dashboard show?

The main page of the web interface is the Explorer, which consists of tabs showing the following information:

- **Dashboard:** shows the score card and displays data on charts. Depending on your project role, you may have access to different dashboards.
- **Action Items:** displays the action plan computed by the Decision Model to improve the quality of your project
- **Highlights:** displays flat lists of artefacts that match conditions defined in each highlight definition
- **Findings:** displays the list of violations in the project
- **Documents:** offers a selection of reports and data exports in various output formats
- **Forms:** displays a list of project attributes that can be modified after an analysis
- **Indicators:** lists the key performance indicators and their associated performance levels computed for the current artefact
- **Measures:** lists the metrics and their computed values for the current artefact
- **Comments:** displays comments and discussion threads started about the current artefact and its descendants



An artefact is a component of your project, to which Squore attaches measures, indicators, textual information and findings. Each artefact is rated by the analysis model, and can consist of several sub-artefacts. When analysing source code, the smallest artefact created by the source code parser is usually a function artefact, which is a child of a class artefact, which is itself a child of a file artefact.

To learn more about the Explorer, consult the [Getting Started Guide](#) or use the online help directly from the web interface.

## Why are there three trees in the web interface?



Squore uses a triple drill-down navigation. Each tree in the web interface allows you to access different information:

- **The Project Portfolios** list all projects and versions, grouped by analysis model and group.
- **The Artefact Tree** displays all the artefacts of your project. Clicking an artefact updates the Explorer to display the information about the selected artefact, and its descendants where applicable.
- **The Indicator Tree** displays details about how the rating for the currently selected artefact is computed. You can click any metric in the tree to see how it is computed, and how it has evolved since the previous analyses of your project.

## How are artefacts rated?

The Software Analytics analysis model is the result of Vector Informatik GmbH's years of experience in the field of software quality and provides the best rating for your software development or system engineering projects. Squore aggregates metrics from a source code parser, various code analysis tools, requirement and test management systems, bugtrackers and version control systems to provide you with an impartial assessment of the quality and progress of your projects through standardised dashboards and reports.

The analysis model is highly customisable so it can adapt to your process and the tools you use in-house.

Your project is broken down into a hierarchy of artefacts of different types, which are evaluated using base measures that are aggregated into key performance indicators. The rating is computed differently for each type of artefact so that all aspects of your project weigh in appropriately overall for the entire project.

## Does Squore provide an API?

A REST API is available. You can find the full API documentation [here](#).

## How do I ensure secure communication between a Squore server and its clients?

You should enable and configure SSL on your server. Follow the Installation and Administration Guide's [instructions](#) for securing your traffic.

Note that you may also need to import your server's certificate into your client's java keystore to run analyses from the command line after securing your server, as explained in the section called [Import a private key and a certificate](#) in the Installation and Administration Guide.

## Is there a command line interface?

Yes. We provide a command line interface so you can automate analyses and integrate them into your continuous integration process. Find out more in the [Command Line Interface](#) manual.

## Are projects analysed automatically when I change source code?

Squore analyses are triggered by humans via the web interface. A command line interface is available so you automate analyses. The command line syntax is described in the [Command Line Interface](#) manual so you can integrate Squore in your continuous integration system or even launch analyses when checking in source code.

## Are my analysis results public? Who can view my source code?

Squore uses a role-based access model, so you decide who gets to see your projects and even if they can launch new analyses or view source code. For a full descriptions of the permissions available to users, check out the Getting Started Guide's [User Project Roles](#) section.





Squore administrators have access to the full list of existing projects on a server so they can administer them and resolve issues.

## Why is there a database and a projects folder?

Part of your data is stored in a database and another part in a folder on the Squore Server's hard drive. Both locations must be backed up together simultaneously to ensure backups can be restored. The database stores the metrics, findings and the artefacts, while the data folder mostly contains information about project configuration and data used to assess the stability and level of cloning in the source code.

## What is Squore doing with my source code? Is it stored in the Squore database?

Squore does not store any of your source code in its database. Each artefact in your project only stores a reference to the location of the source code that was analysed. Users with appropriate access may request to view source code referenced in an analysis, which will prompt them for valid credentials for your repository.

During an analysis, source code is retrieved via a SVN checkout or a git clone and stored into a temporary folder that is deleted when it is no longer needed.

In sensitive environments, it is even possible to run analyses locally on a client so that the source code never reaches the Squore Server. See more about [Common Deployment Scenarios](#) in the Installation and Administration Guide.

## Can I modify the dashboards?

Squore's dashboards, but also its decision model, highlight and export definitions are fully configurable. The [Configuration Guide](#) showcases all the possible customisations so you can modify the default model, or even build a model from scratch.

Instead of modifying the default configuration, Squore lets you override the default model so you can keep your customisations when you upgrade to a new version. For more details about overriding the default configuration, refer to the Configuration Guide's [Customising the Squore Configuration](#).

## What are the known limitations of Squore? How many projects and versions can I create? How many metrics? How many charts?

You do not have to worry about how many metrics and indicators you can add to your model. The Squore database can handle over  $10^{18}$  metrics.

Dashboards can be extended to include many charts, as they are rendered client-side and should not have a meaningful impact on server performance.

Consult the section called [Sizing Squore Server and Database](#) in the Installation and Administration Guide for more information about database and disk usage.

## How do I read and analyse server logs?

Squore runs in a WildFly instance which the installer deploys for you. As a result, some WildFly server codes can be quite useful to analyse issues:

- WFLYSRV0049: WildFly application server startup launched
- WFLYSRV0010: application (squore-server, squore-license) deployed successfully - message displayed during startup
- WFLYSRV0025: WildFly application server started successfully
- WFLYSRV0026: WildFly application server started with errors

- WFLYSRV0028: application (square-server, square-license) stopped successfully - message displayed during shutdown
- WFLYSRV0050: WildFly application server shutdown successfully
- <TIMESTAMP> INFO [com.squoring.square.server.BootstrapBean] (ServerService Thread Pool-\d+) {} Square started—Square is available for users to log in

Other useful log messages include:

- <TIMESTAMP> INFO User <user> authenticated / unauthenticated - for user login/logout actions
- <TIMESTAMP> INFO {\taskId\} <project> semaphore acquired / released - indicated the beginning / end of a task of a project (analysis or deletion)
- <TIMESTAMP> WARN {taskId} No data found for this measure : '<MEASURE\_ID>' for type '<TYPE>' - indicates that no Data Provider provided a value for a metric, when the analysis model expected one. Cases where this is unexpected should be investigated.
- <TIMESTAMP> WARN {taskId} Measure '<MEASURE\_ID>' that occurs x times is not defined in model - indicates that a Data Provider tried to import a metric that is not defined in the analysis model. Review your analysis model to check if the reported metric should be defined.
- <TIMESTAMP> WARN {taskId} On measure <MEASURE\_ID> (<ARTEFACT\_TYPE>) > On computation: <COMPUTATION> > x/y artefacts use the default value - indicates that the default value for was used for a metric. Cases where this is unexpected should be investigated.
- <TIMESTAMP> WARN ARJUNA012117 - indicates that a timeout happened. This can mean a timeout when clicking a control in the web interface, or a timeout during a project task (analysis or deletion).



The server log is only accessible to Square administrators, but Square project owners also have access to logs specific to their own projects via the *Projects* page in the web interface.

# Chapter 2. Preparing your Machine for Squire

This chapter only covers how to ensure that your environment can be prepared for a Squire installation. For actual instructions on how to install Squire itself, refer to the full Installation and Administration Guide.

## Installation Prerequisites

### Supported Operating Systems

The following is a list of the officially supported and tested operating systems:



A 64-bit version of the OS is required

- CentOS 7
- CentOS 8
- Ubuntu 18.04 LTS
- Ubuntu 20.04 LTS
- Windows 10
- Windows Server 2016

The following is a list of the operating systems that are not regularly tested but are known to be working:

- Red Hat Enterprise Linux 7
- Fedora 29
- SuSe Linux 11.1
- Ubuntu Server 16.04
- Windows 7
- Windows 8
- Windows Server 2008 R2
- Windows Server 2012 R2

### For All Systems

For a successful installation of Squire, you will need:

- A Java Runtime Environment version 8 or 11 (64-bits) (other versions are not supported)
- At least 4 GB of space available on the disk for a full installation with demo projects
- At least 8 GB of RAM on the server machine



Keep in mind that the requirements above are the strict minimum. In production, Squire Server generally runs on a dedicated machine. A performant configuration is usually:

- 16 threads CPU.
- 64GB of RAM.
- SSD hard drives.

Squire reserves 25% of the available RAM of the machine to the database and another 25% to the server. External processes (like Checkstyle or FindBugs) running on the same machine as Squire may add to the amount of RAM required for analysing source code. Linux is known to offer better performances than Windows when running Squire. For a production database, you should plan a minimum of 20 GB of disk space.

## Packages for Windows

## Packages for CentOS and Red Hat Enterprise Linux

On Red Hat Enterprise Linux and CentOS (6.5, 7.1 and 8), the dependencies are satisfied by the following packages:

### Mandatory packages:

- **java-1.8.0-openjdk**
- **perl**
- **perl-Date-Calc**
- **perl-Digest-SHA**
- **perl-JSON**
- **perl-libwww-perl**
- **perl-Time-HiRes**
- **perl-XML-Parser**
- **fontconfig**
- **tcl**

For more information about how to install the Extra Packages for Enterprise Linux (EPEL) repository, consult <https://fedoraproject.org/wiki/EPEL>.

## Packages for Ubuntu

On Ubuntu 16.04 LTS, 18.04 LTS and 20.04 LTS, the dependencies are satisfied by the following packages:

### Mandatory packages:

- **libdate-calc-perl**
- **libhttp-message-perl**
- **libjson-perl**
- **libwww-perl**
- **libxml-parser-perl**
- **openjdk-8-jre**
- **perl**

- tcl

## Packages for other Linux distributions

On Linux platforms, the following must be installed before installing Squire:

- **Perl** version 5.10.1 or greater including the following extra-modules:
  - Mandatory packages:
    - **Date::Calc** [module details]
    - **Digest::SHA** [module details]
    - **HTTP::Request** [module details]
    - **JSON** [module details]
    - **LWP** [module details]
    - **LWP::UserAgent** [module details]
    - **Time::HiRes** [module details]
    - **XML::Parser** [module details]



If some of these modules are not available as packages on your operating system, use your perl installation's cpan to install the modules. Using the OS packages is recommended, as it avoids having to reinstall via cpan after upgrading your version of perl.

- **Tcl** version 8.5 or greater,

## Third-Party Plugins and Applications

If you have deployed some third-party tools on Squire Server, they will automatically be downloaded to your client when you launch the client synchronisation script.

AntiC and Cppcheck on Linux also require special attention: Cppcheck must be installed and available in the path, and antiC must be compiled with the command:



```
cd addons/Antic_auto/bin/ && gcc antic.c -o antic
```

For more information, refer to the Command Line Interface Manual, which contains the full details about special installation procedures for Data Providers and Repository Connectors.

## Requirements for LDAP Integration

In order to configure Squire to integrate with your LDAP Server, you should make sure that you have access to the following information:

- The address of the LDAP server you want to connect to Squire.
- The section(s) of the directory that contain the users that should be allowed to log into Squire.
- The login and password of a user account allowed to browse the section(s) of the directory mentioned above.
- Basic knowledge of your directory structure. Note that Squire was tested with Microsoft Active Directory on Windows Server 2008 and OpenLDAP on Ubuntu 12.04.

If you need to obtain this information to a system administrator, ask for these details:

- **java.naming.provider.url**: The URL of the directory server.
- **baseCtxDN**: The fixed DN of the context to start the user search from.
- **bindDN**: The DN used to bind against the ldap server for the user and roles queries. This is some DN with read/search permissions on the baseCtxDN and rolesCtxDN values.
- **bindCredential**: The password for the bindDN
- **baseFilter**: The search query sent by Squire to the LDAP server when authenticating. If the password is correct and the search returns true, the user is allowed to log into Squire. The default query checks that the login exists, but you can change it to check that the login is valid and that the user is part of a specific group for example, using the syntax *& condition1) (condition2*. For more information about LDAP query syntax, refer to [https://technet.microsoft.com/en-us/library/aa996205\(v=exchg.65\).aspx](https://technet.microsoft.com/en-us/library/aa996205(v=exchg.65).aspx). Note that the **&** characters must be written as an entity (**&amp;**) in the settings file.
- **rolesCtxDN**: The fixed DN of the context to search for user roles. This is required to exist, even though it is not used by Squire at the moment.
- **userCompositeName (optional)**: the field in the LDAP account that Squire will import and user as the user's full name.
- **userMail (optional)**: the field in the LDAP account that Squire will import and user as the user's e-mail address.
- **userOrganizationUnit (optional)**: the field in the LDAP account that Squire will import and user as the user's department.
- **userId (optional)**: the field in the LDAP account that Squire will use as a the final user login to create the account or log into the application. When no value is specified for this field, Squire uses the login as typed by the user on the login page.



Using this field helps avoiding confusion with mixed-case logins. Squire Server considers **demo** and **Demo** as two separate users by default. By specifying that the login is taken from a specific field from your directory, you ensure that the same account is used no matter what case was used in the login form.

## Requirements for TeamForge Integration

In order to configure Squire to integrate with TeamForge, you will need to provide the following Teamforge details:

- **TeamForge Server URL (without / at the end)**, for example *http://localhost*.
- **TeamForge Server Name (will appear in Squire export format list)**, for example *TeamForge*.
- **TeamForge artifact default status at export**, for example *Open*.
- **TeamForge artifact default priority at export**, for example *4*.
- **SvnEdge Main Repository URL**, for example *http://localhost/svn*.
- **SvnEdge Viewer (viewvc) URL**, for example *http://localhost/viewvc*.
- **External System Id of SvnEdge (for ex: exsy1001)**, for example *exsy1011*.

The rest of the configuration is carried out on the TeamForge side with data obtained from Squire after you create some projects. You can see this information in the full Installation and Administration Guide

# Appendix A: Licences

## Software Licence Agreement

Squore Software

End-User License and Support Agreement

Please read this document carefully. This is a legal agreement by which Vector Informatik GmbH ("Vector") permits use of its Software products ("Squore Software"). The user ("Customer") accepts the terms of this Agreement by taking any or all of the following actions: (a) by signing an order form or purchase order referencing either this Agreement or a Technical and Financial proposal issued by Vector (an "Order Form"), (b) by opening the package containing the Software, and/or (c) by installing the Software on a computer ("Target Hardware").

CUSTOMER CONSENTS TO BE LEGALLY BOUND BY THESE TERMS. IF CUSTOMER DOES NOT AGREE TO ALL OF THE TERMS OF THIS AGREEMENT, CUSTOMER MUST NOT USE THE SOFTWARE AND MUST RETURN IT, INCLUDING ANY PRINTED ASSOCIATED DOCUMENTATION, TO VECTOR WITHIN FOURTEEN (14) DAYS TO RECEIVE A FULL REFUND OF THE PURCHASE PRICE.

### 1) DEFINITION

(a) "Squore Software" includes (i) the Software identified in an Order Form or delivered with this Agreement; (ii) any authorized copies thereof; (iii) all related documentation ("Documentation") delivered with or included in that software; and (iv) any update to that software that Customer may receive from Vector.

(b) Squore Software is based on a traditional 3-tier architecture consisting of:

- . a database and a data folder for storing project and user management data
- . an application server running the "Squore Server Software" and the "License Server Software"
- . a client front-end accessible through a Web Browser and a Command Line Interface to interact with the application server from a client machine

(c) Target hardware ("Target Hardware") is uniquely identified by the hardware specification and the operating system running on it.

### 2) LICENSE

Vector grants to Customer, and Customer accepts from Vector, a non-exclusive and non-transferable right and license to use the Squore Software that is specified in the Order Form and/or that accompanies this Agreement, but only (i) in accordance with the related Documentation, (ii) subject to Customer's payment of applicable license fees and (iii) subject to the terms and conditions specified below.

Customer agrees that Customer does not have, and does not hereby acquire, any title or rights of ownership in any Squore Software or, except for the license rights hereby granted, any right to use, copy, transfer or disclose all or any portion of any Squore Software. The Squore Software is protected by copyright laws and international treaties.



### 3) FEES

The fees for the license under this Agreement are set forth in the applicable Order Form or, if no Order Form exists, in the applicable Technical and Financial proposal issued by Vector, or, if no proposal exists, then in accordance with Vector current list prices.

### 4) USE

(a) Unless otherwise stated in a special agreement, all dissemination or commercial exploitation of Squire Software results is strictly forbidden.

(b) The license granted by this Agreement is a license under which a maximum number of active users and projects specified in the Order Form may use the Squire Software. An "Active User" is a physical user registered in the Squire Software database. Active Users are not shared among several Squire software databases. A user is active if any activity has been recorded by the Squire Server Software in the past 6 months. Activities include remote project creation, viewing of analysis results, and e-mail notification.

(c) The management and regulation of Active Users is managed by the License Server Software hosted on the Target hardware designated by the Customer.

(d) Except for continuous integration purpose, it is strictly forbidden to share the same Squire Software login between different physical users.

(e) Customer will ensure that at least one of its employees has completed the two days on-site training course "Adminstrating Squire software" given by a Squire Software certified trainer, and that such trained employee(s) are the people within Customer's organization who are responsible for interactions with Squire on maintenance and support matters.

(f) Customer may make a reasonable number of back-up or archival copies of the Software. Customer will reproduce all confidentiality and proprietary notices on each of these copies and maintain an accurate record of the location of each of these copies.

(g) Customer will not:

- . Reverse compile, disassemble, or otherwise reverse engineer any Squire Software, or allow anyone else to do so (except only to the extent such prohibition is contrary to applicable law).

- . Remove or destroy any proprietary markings or legends or any encrypted license keys or similar security devices placed upon or contained in any Squire Software.

- . Modify or adapt the Squire Software or create a derivative work based on or incorporate the Squire Software into or with other software.

- . Unless otherwise stated in a special agreement, distribute, sublicense, share, display, or in any manner make the Squire Software available to any third party, with or without compensation.

- . Use all or any part of the Squire Software to create other software a principal purpose of which is to perform the same or similar functions as, or to replace any component of, the Squire Software.

### 5. MAINTENANCE AND SUPPORT

(a) Depending on the type of licenses bought by the Customer, Vector will provide support and maintenance services according to the following schedule:

i. In case of perpetual licenses bought by Customer

. Maintenance fees will be charged in addition to the price of the purchase of the Squore Software licenses. The annual initial amount of maintenance fees will be calculated on the basis of a 20% percentage of the net list price of the software licenses purchased by the Customer.

. After one year, Maintenance and Support services will be renewed by tacit agreement of the parties, for annual periods. Before the anniversary date of each license for which Support Services are in effect, Vector shall advise the Customer of the applicable Maintenance and Support Service fees for the coming year.

. The termination of maintenance contract will be effective only by sending a registered letter with acknowledgement of receipt denouncing the contract at least 60 days before the end of the period of validity.

ii. In case of software licenses subscription

If the license to use the software is subject to a periodic subscription, the maintenance cost for these licenses is included in the subscription price. The maintenance is so provided during all the duration of the subscription, and stops automatically at the end of the subscription validity.

(b) Conditions. Maintenance and support services as defined hereafter in paragraph 5-(c) are applicable subject to the following conditions:

. Squore Software is covered by a valid maintenance contract for all acquired Squore software licenses.

. Squore Software was not modified by the Customer.

. The version of the installed Squore Software corresponds to one of the two latest annual major updates distributed by Vector.

. Customer engages to comply with the normal use of the software, strictly comply with the instructions given by Vector and to respect all provisions in the present agreement.

. Customer shall nominate from among its staff a technical coordinator and an alternate coordinator at the Customer Site(s), with up to date knowledge of Squore licensed products usage and sufficient technical knowledge to interact with Vector support staff. In case of change of the coordinators, the Customer will provide written notification to Vector.

(c) Maintenance and support services include:

. Assistance from support: the online support (hot line) is available during Vector normal business hours from 9 am to 6 pm (Central European Time) to answer the questions of the technical coordinator when technical facts encountered in the use of the Squore Software. Support will help to identify problems and provide, where appropriate, temporary fixing patches. Contact information for support is:

- support site: <https://portal.vector.com>

- email: [support@vector.com](mailto:support@vector.com)

. Corrective maintenance: the corrective maintenance includes the development, to the extent commercially reasonable, of workarounds or program fixes for malfunctions submitted by Customer. Are considered as malfunctions recognized or reproducible defects resulting in distorted results compared to those defined in the software Manual and not coming from non-observance of the instructions of the said Manual.

. Updates: updates include the delivery of successive versions of the software, being either due to bug fixes or to enhancements of performances or features (this delivery does not necessarily provide new additional features). Customer will destroy any prior version before installing a new update.

. Rehost: any change of "Target hardware" implies a change of license keys and shall be subject to the prior written authorization of Vector and to the signature by the

Customer of a letter of destruction of all the license files already installed.

d) Maintenance and support services do not include:

- . Time spent, after request of the Customer by Vector staff not directly attributable to maintenance services: search for non-reproducible anomalies, malfunction due to non-compliance with the Manual without the Software itself being an issue, unavailability of the system, operating activities prior to the intervention (such as preliminary backup ...).

- . Installation of the Software by Vector.

- . Additional services which do not fall within the scope of maintenance services as defined above in paragraph 5-(c).

- . On-site support: Vector may offer on-site support to Customer at additional charges

e) Procedure for the submission of requests:

- . To be taken into account, Customer requests shall be sent using the support site at <https://portal.vector.com> or by email at [support@vector.com](mailto:support@vector.com)

- . Customer agrees to give, in support of a request for correction due to an anomaly, any information likely to facilitate the search for the causes of this anomaly, and to give for free to Vector an open access to its premises and development stations in the day and hours necessary to perform the contract, and to ensure the conservation, under the conditions of appropriate security and condition of the latest version, of the sources of programs in case of this is necessary to enable Vector to perform its maintenance services.

- . Regarding correction of anomalies, Vector is committed to act as soon as possible to correct the anomalies detected. Vector will transfer to the Customer either a technique to bypass the anomaly or a patch of necessary corrections or a new version of the software.

- . Any issues not resolved with the initial response will be investigated using the data provided. Below are the targeted response times for continued investigations.

- Blocker: 1 day, daily update. A request is "blocking" when the incident has a significant impact with a risk of operating loss or when datas are corrupted. The significant impact is appreciated by Vector.

- Serious: 2 days, weekly update

- Major: 3 days, monthly update

- Minor: 1 week, monthly update

(f) Limitations

- . Vector is expressly subject to an obligation of means.

- . Vector is not required to (i) develop and release any, or any particular type of enhancements or (ii) customize the enhancements to satisfy Customer's particular requirements.

- . The Updates will not include any upgrade or new version of the Products that Vector decides, in its sole discretion, to make generally available as a separately priced item.

- . Vector will be released from any responsibility in case of breach by the Customer of any provision of this maintenance terms and conditions.

## 6. WARRANTIES AND REMEDIES

(a) Limited Warranty. Vector warrants that it has the right to (i) enter into this Agreement and (ii) grant the licenses hereunder. Vector also warrants that the Square Software will perform substantially as described in the Documentation during a 90 days Warranty Period. Customer acknowledges that (i) the Products may not satisfy all of Customer's requirements and (ii) the use of the Products may not be uninterrupted or

error-free.

(b) Remedies. Vector or its representative will correct or replace any defective Software. Customer acknowledges that this paragraph sets forth Customer's exclusive remedy, and Vector exclusive liability, for any breach of warranty or other duty related to the quality of the Products.

(c) Disclaimer. EXCEPT AS EXPRESSLY PROVIDED IN THIS AGREEMENT OR REQUIRED BY APPLICABLE LAW, ALL WARRANTIES, CONDITIONS, REPRESENTATIONS, INDEMNITIES AND GUARANTEES WITH RESPECT TO THE PRODUCTS, WHETHER EXPRESS OR IMPLIED, ARISING BY LAW, CUSTOM, PRIOR ORAL OR WRITTEN STATEMENTS BY VECTOR, ITS REPRESENTATIVES OR OTHERWISE (INCLUDING, BUT NOT LIMITED TO ANY WARRANTY OF MERCHANTABILITY, SATISFACTION AND FITNESS FOR A PARTICULAR PURPOSE) ARE HEREBY OVERRIDDEN, EXCLUDED AND DISCLAIMED. IN NO EVENT SHALL THE AGGREGATE LIABILITY OF VECTOR TO CUSTOMER ON ACCOUNT OF ANY MATTER ARISING WITH RESPECT TO THE SQUORE SOFTWARE EXCEED THE LICENSE FEES PAID BY CUSTOMER UNDER THIS AGREEMENT.

(d) Infringement Indemnity. If an action is brought against Customer claiming that the Product infringes a patent, trade secret or copyright, Vector will defend Customer at Vector expense and, subject to this Section, pay the damages and costs finally awarded against Customer in the infringement action, but only if (i) Customer notifies Vector promptly upon learning that the claim might be asserted, (ii) Vector has sole control over the defense of the claim and any negotiation for its settlement or compromise, and (iii) Customer takes no action that is contrary to Vector interest. If a claim described above may be or has been asserted, Customer will permit Vector, at Vector option and expense, to (A) procure the right to continue using the Product, (B) replace or modify the Product to eliminate the infringement while providing functionally equivalent performance, or (C) accept the return of the Product and refund to Customer the License Fee actually paid to Vector for such Product, less depreciation based on a 5-year straight-line-depreciation schedule.

Vector shall have no indemnity obligation to Customer under this Section if the patent or copyright infringement claim results from (i) a correction or modification of the Product not provided by Vector, (ii) the failure to promptly install an Update or Enhancement at Vector direction with knowledge that installation thereof would have avoided the infringement or (iii) the combination of the Product with other non-Vector software or (iv) any unauthorized use of the Squore Software, or (v) any version of the Software other than the latest update offered by Vector to Customer at no additional charge.

## 7. LIMITATION OF LIABILITY

UNDER NO CIRCUMSTANCES WILL VECTOR OR ITS REPRESENTATIVES BE LIABLE FOR ANY CONSEQUENTIAL, INDIRECT, SPECIAL, PUNITIVE OR INCIDENTAL DAMAGES OR LOST PROFITS, WHETHER FORESEEABLE OR UNFORESEEABLE, BASED ON CUSTOMER'S CLAIMS OR THOSE OF ITS CUSTOMERS (INCLUDING, BUT NOT LIMITED TO, CLAIMS FOR LOSS OF DATA, GOODWILL, USE OF MONEY OR USE OF THE PRODUCTS, INTERRUPTION IN USE OR AVAILABILITY OF DATA, STOPPAGE OF OTHER WORK OR IMPAIRMENT OF OTHER ASSETS), ARISING OUT OF BREACH OR FAILURE OF EXPRESS OR IMPLIED WARRANTY, BREACH OF CONTRACT, MISREPRESENTATION, NEGLIGENCE, STRICT LIABILITY IN TORT OR OTHERWISE. IN NO EVENT WILL THE AGGREGATE LIABILITY WHICH VECTOR OR ITS REPRESENTATIVES MAY INCUR IN ANY ACTION OR PROCEEDING EXCEED THE LICENSE FEES ACTUALLY PAID BY CUSTOMER FOR THE SPECIFIC PRODUCT THAT DIRECTLY CAUSED THE DAMAGE. THIS SECTION WILL NOT APPLY ONLY WHEN AND TO THE EXTENT THAT APPLICABLE LAW SPECIFICALLY REQUIRES LIABILITY, DESPITE THE FOREGOING EXCLUSION AND LIMITATION.

## 8. OWNERSHIP

All trademarks, service marks, patents, copyrights, trade secrets and other proprietary rights in or related to the Products are and will remain the exclusive property of Vector, whether or not specifically recognized or perfected under local applicable law. Customer will not take any action that jeopardizes Vector proprietary rights or acquire any right in the Products, except the limited use rights specified in Section 4. Vector will own all rights in any copy, translation, modification, adaptation or derivation of the Products, including any improvement or development thereof.

## 9. CONFIDENTIALITY

(a) Confidentiality. Customer acknowledges that the Products constitute and incorporate confidential and proprietary information developed or acquired by or licensed to Vector. Customer will take all reasonable precautions necessary to safeguard the confidentiality of the Products, including at a minimum those taken by Customer to protect Customer's own confidential information. Customer will not allow the removal or defacement of any confidentiality or proprietary notice placed on the Products. The placement of copyright notices on these items will not constitute publication or otherwise impair their confidential nature.

(b) Disclosure. Customer will not disclose, in whole or in part, the Products or any portion thereof or other information that has been designated as confidential to any individual, entity or other person, except to those of Customer's employees or consultants who require access for Customer's authorized use of the Products, provided such consultants agree in writing to comply with the use and non-disclosure restrictions applicable to the Products under this Agreement. Customer acknowledges that any unauthorized use or disclosure of the Products may cause irreparable damage to Vector. If an unauthorized use or disclosure occurs, Customer will immediately notify Vector and take, at Customer's expense, all steps which may be available to recover the Products and to prevent their subsequent unauthorized use or dissemination. Vector agrees to take the same action regarding any information designated in writing as proprietary which it receives from Customer ("Customer Information").

(c) Limitation. Neither Vector nor Customer will have any confidentiality obligation with respect to any portion of the Products or Customer Information that (i) the receiving party knew or independently developed before receiving such Products or Customer Information under this Agreement, (ii) the receiving party lawfully obtained from a third party under no confidentiality obligation, or (iii) became available to the public other than as a result of any act or omission by the receiving party or any of receiving party's employees or consultants.

## 10. TERMINATION

Customer may terminate this Agreement or any Order Form, without right to refund, by notifying Vector of such termination and returning the Product and copies thereof to Vector. Vector may terminate this Agreement, upon reasonable notice and without judicial or administrative resolution, if Customer or any of Customer's employees or consultants breach any material term or condition hereof. This Agreement will terminate automatically if Customer becomes insolvent or enters into bankruptcy, suspension of payments, moratorium, or any other proceeding that relates to insolvency

or protection or creditors' rights.

Upon the termination of this Agreement for any reason, all rights granted to Customer hereunder will cease, and Customer will stop using Squore Software, return or destroy all copies and so certify to Vector in writing. The provisions of Sections 6-8 will survive the termination of this Agreement.

#### 11. INSPECTION

During the term of this Agreement, Vector or its representative, if in receipt of credible evidence of non-compliance, may, upon prior notice to Customer, inspect the files, computer processors, equipment and facilities of Customer during normal working hours to verify Customer's compliance with this Agreement. While conducting such inspection, Vector or its representative will be entitled to copy any item that Customer may possess in violation of this Agreement, without disruption of Vector business and violation of Laws and Regulations.

#### 12. ASSIGNMENT

Customer shall not assign, delegate or otherwise transfer this Agreement or any of its rights or obligations hereunder without Vector prior approval which shall not be unreasonably withheld.

#### 13. MISCELLANEOUS

. Any terms and conditions of any unilateral letter, memorandum, purchase order or other writing issued by Customer shall not be binding on Vector. Any waiver or modification of this Agreement will not be effective unless executed in writing and signed by an authorized representative of Vector and Customer. This Agreement will bind Customer's successors-in-interest.

. This Agreement will be governed by and interpreted in accordance with the laws of Germany. If any provision of this Agreement is held to be unenforceable, in whole or in part, such holding will not affect the validity of the other provisions of this Agreement, unless the Parties in good faith deem the unenforceable provision to be essential, in which case either Party may terminate this Agreement effective immediately upon notice to the other Party. This Agreement constitutes the complete and entire statement of all conditions and representations of the agreement between Vector and Customer with respect to its subject matter and supersedes all prior writings or understandings.

## Redistributed Software

Redistributed Software	Version	License File
-----		
Wildfly	10.1.0.Final	lgpl-2.1
PostgreSQL	8.4	postgresql-bsd
Perl	5.12.3	al
TCL	8.5	tcltkl,tcllib
PhantomJS	2.1.1	bsd3

Components	Sub component	Version	License File
------------	---------------	---------	--------------

-----			
Richfaces		4.5.17	lgpl-2.1
PostgreSQL JDBC Driver		42.0.0	bsd2
Oracle JDBC driver		12.1.02	otn
Omnifaces		2.6	apache-2.0
ANTLR		3.1	antlr3
JasperReports Library		4.8	lgpl-2.1
Checkstyle		5.6	lgpl-2.1
JTcl		2.8	jtcl, amd, itcl, janino,
tcllib, tcltk, ucb			
log4j		1.2.17	apache-2.0
Apache Commons	commons-lang3	3.1	apache-2.0
Apache HttpComponents	httpclient	4.1.2	apache-2.0
Apache HttpComponents	httpmime	4.1.2	apache-2.0
Apache XML Graphics	batik-transcoder	1.7	apache-2.0
Bouncy Castle	bcmail-jdk16	1.46	bouncy-castle
com.beust	jcommander	1.48	apache-2.0
com.google.collections	google-collections	1.0	apache-2.0
com.googlecode.juniversalchardet	juniversalchardet	1.0.3	mpl1.1
com.sun.mail	javax.mail	1.5.3	gf
commons-cli	commons-cli	1.2	apache-2.0
commons-collections	commons-collections	3.2.2	apache-2.0
javax.enterprise	cdi-api	1.2	apache-2.0
javax.validation	validation-api	1.1.0.Final	apache-2.0
net.java.dev.jna	jna	4.1.0	lgpl-2.1
net.sf.jsci	jsci	1.2	lgpl-2.1
net.sf.saxon	saxon-xom	8.7	mpl1.0
nux	nux	1.6	nux
org.glassfish	javax.json	1.0.4	gf
org.jdom	jdom2	2.0.5	jdom
Resources		Version	License File
-----			
CodeMirror		4.4.0	mit
font-awesome		4.7.0	mit, ofl-1.1
JavaScript InfoVis Toolkit		2.0.1	mit
jquery		1.12.3	jquery
jquery-mobile		1.4.2	jquery-mobile
jquery-ui-resizable		1.11.4	mit, jquery
lodash.js		4.17.11	mit
notify.js		0.4.2	mit
gridstack.js		1.0.0	mit
spectrum.js		1.8.0	mit



# Index

## A

Architecture, 3

## C

condition1) (condition2, 12

## D

Disk Space, 9

## J

Java, 9

## L

Licences, 13

## M

Memory, 9

## P

PhantomJS, 3

Prerequisites, 9

## R

Redistributed Software, 13

## S

Server

    Application Server, 3

    Software Licence Agreement, 13