

FAQs - Frequently Asked Questions

This page intends to collect more frequently asked questions from our users and customers.

Please, use search below to find specific help topics in our [Documentation](#) or browse through our [FAQs list](#)

- [Kiuwan FAQs - Frequently Asked Questions](#)
- [Where can I find useful Kiuwan resources](#)
- [What Security and Quality Standards are supported by Kiuwan](#)
 - [CWE Common Weakness Enumeration](#)
 - [How does Kiuwan help me to comply with OWASP](#)
 - [OWASP Top 10 for 2017](#)
 - [OWASP Benchmark and Kiuwan](#)
 - [How does Kiuwan help with security in Cobol, RPG and ABAP](#)
 - [Why Kiuwan Code Security vulnerabilities do not match Code Analysis Security defects ?](#)
- [Which are the main Indicators provided by Kiuwan](#)
 - [What's the meaning of Risk Index and how is calculated](#)
- [How does Kiuwan help me to make decisions on how to fix my application](#)
- [How does Kiuwan help me to manage and make strategic decisions on my Application Portfolio](#)
 - [How can I obtain aggregated indicators and metrics for applications Governance](#)
 - [How does Kiuwan help me to build Decision Quadrants on my Application Portfolio](#)
 - [How does Kiuwan help me to manage Software Vendors](#)
- [Does Kiuwan let me to manage software development Life Cycle](#)
 - [What about Team management](#)
- [What are the requirements to use Kiuwan](#)
- [Can I use Kiuwan without uploading my code](#)
 - [How can I download Kiuwan Local Analyzer](#)
- [What programming languages are supported by Kiuwan](#)
 - [What is the Kiuwan support for SQL](#)
 - [Does Kiuwan support mobile technologies](#)
 - [Does Kiuwan support PHP Symfony](#)
 - [How to analyze ABAP code](#)
 - [How to analyze C/C++ code](#)
 - [How to analyze OracleForms code](#)
 - [Does Kiuwan detect duplicated code](#)
- [How to use Kiuwan in Continuous Integration \(Jenkins, for example\)](#)
 - [How could I integrate Kiuwan results in my own Dashboard](#)
 - [Does Kiuwan integrate with JIRA](#)
 - [Does Kiuwan allow the development of custom quality rules](#)
 - [Does Kiuwan integrate results from 3rd party static analyzers](#)
 - [Does Kiuwan integrate with LDAP](#)
- [What is a Kiuwan application](#)
- [What is a Kiuwan Model? Should I start from scratch](#)
 - [How can I deactivate a rule](#)
 - [How can I mute defects](#)
- [How to manage Kiuwan defects when I do not completely agree with them](#)

Kiuwan FAQs - Frequently Asked Questions

Where can I find useful Kiuwan resources

- [Kiuwan Web site](#)
 - <https://www.kiuwan.com/>
- [Kiuwan Product Documentation:](#)
 - <https://www.kiuwan.com/docs>
- [Kiuwan Blog:](#)
 - <https://www.kiuwan.com/blog>
- [Troubleshooting Guide:](#)
 - [Troubleshooting](#)
- [Kiuwan Technical Support:](#)
 - [Contact Kiuwan Technical Support](#)

What Security and Quality Standards are supported by Kiuwan

Kiuwan provides out-of-the-box "rules" to accomplish your applications security and quality with industry standards, such as CWE, OWASP, CERT-Java/C/C++, SANS-Top25, WASC, PCI-DSS, NIST, MISRA, BIZEC, ISO/IEC 25000 and ISO/IEC 9126.

Please visit our blog post on [Security standards in software development](#)

CWE Common Weakness Enumeration

The Common Weakness Enumeration (**CWE**) is an extension of the Common Vulnerabilities and Exposures (CVE) list compiled by MITRE, a federally-funded, non-profit organization that manages research and development centers supporting government agencies like Homeland Security.



Kiuwan provides full support and mapping for CWE list.

You can find further info at:

- <https://www.kiuwan.com/blog/cwe-common-weakness-enumeration/>
- [Kiuwan and CWE](#)

How does Kiuwan help me to comply with OWASP

OWASP is an international non-profit organization dedicated to analyzing, documenting and spreading the principles for the safe and vulnerability-free software development.

They produce a document called **OWASP Top 10**. As they say: "The OWASP Top Ten is a powerful awareness document for web application security. The OWASP Top Ten represents a broad consensus about what the most critical web application security flaws are. **Adopting the OWASP Top Ten is perhaps the most effective first step** towards changing the software development culture within your organization into one that produces secure code".

You can visit our [Kiuwan Blog](#) and learn how Kiuwan can help you on assessing and fixing your security vulnerabilities according to OWAS Top Ten:

- [OWASP Top 10: how to discover vulnerabilities in your Java applications](#)
- [OWASP Top 10: how to discover vulnerabilities in your C# applications](#)

OWASP Top 10 for 2017

For 2017, the OWASP Top 10 Most Critical Web Application Security Risks are:

- A1 Injection (<https://www.kiuwan.com/blog/owasp-top-10-a1-injection>)
- A2 Broken Authentication and Session Management (<https://www.kiuwan.com/blog/owasp-top-10-2017-a2>)
- A3 Sensitive Data Exposure (<https://www.kiuwan.com/blog/owasp-top-10-2017-a3-sensitive-data-exposure-identify-your-weaknesses/>)
- A4 XML External Entities (XXE) (<https://www.kiuwan.com/blog/a4-xml-external-entities-xxe/>)
- A5 Broken Access Control (<https://www.kiuwan.com/blog/owasp-top-10-2017-a5-broken-access-control/>)
- A6 Security Misconfiguration (<https://www.kiuwan.com/blog/owasp-top-10-2017-a6-security-misconfiguration/>)
- A7 Cross-Site Scripting (XSS) (<https://www.kiuwan.com/blog/owasp-top-10-2017-a7-cross-site-scripting-xss/>)
- A8 Insecure Deserialization (<https://www.kiuwan.com/blog/owasp-top-10-2017-a8-insecure-deserialization/>)
- A9 Using Components with Known Vulnerabilities (<https://www.kiuwan.com/blog/owasp-top-10-2017-a9-using-components-with-known-vulnerabilities/>)
- A10 Insufficient Logging and Monitoring (<https://www.kiuwan.com/blog/owasp-top-10-2017-a10-insufficient-logging-monitoring/>)

You can browse Kiuwan rules by OWAS Top10 to find what Kiuwan rules complies with what OWASP security risks.

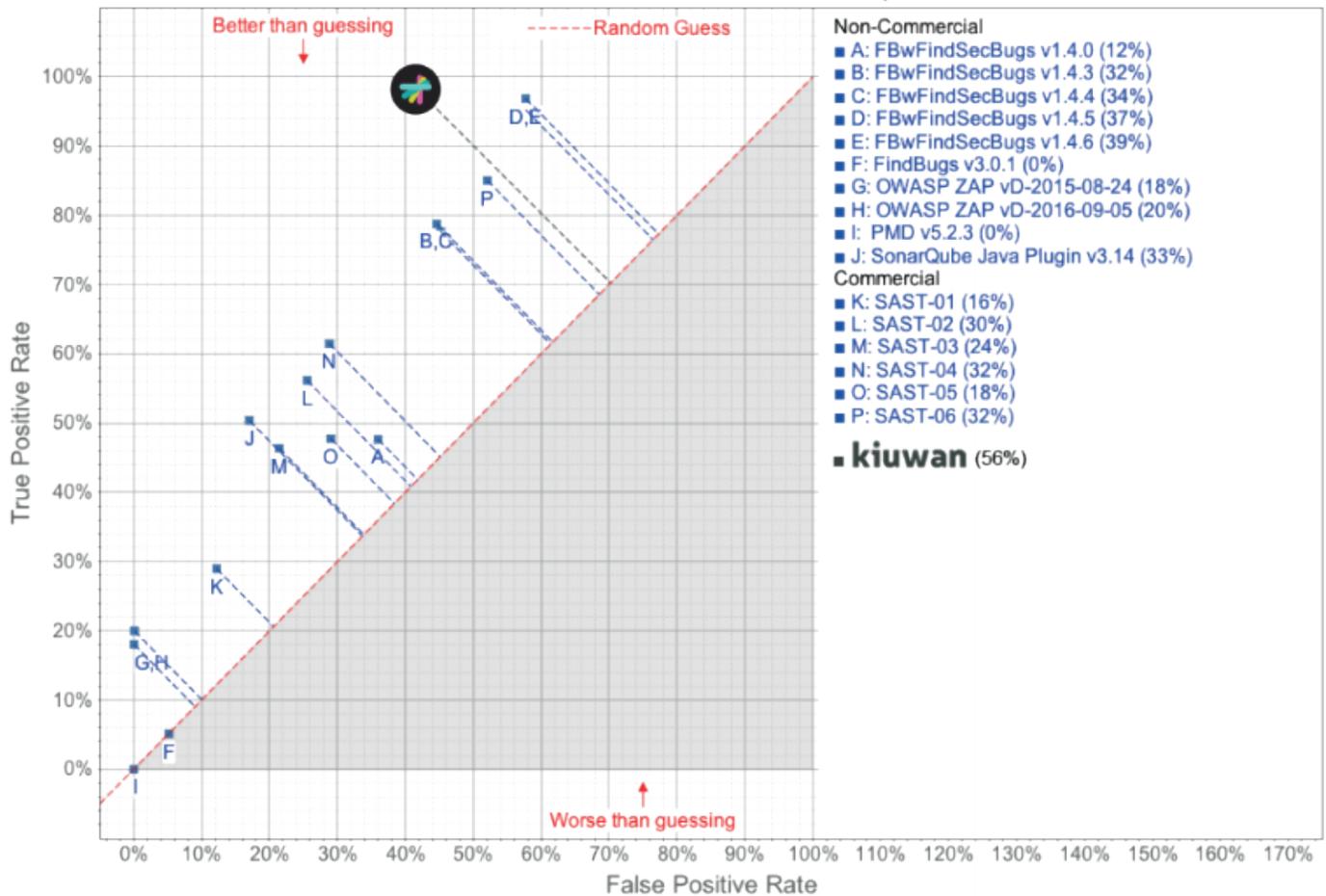
OWASP Benchmark and Kiuwan

The **OWASP Benchmark** is a *test suite designed to evaluate the coverage and accuracy of automated vulnerability detection tools*.

We have run **Kiuwan** against the **OWASP Benchmark** test cases and here you have the results (compared to open and commercial tools).

Kiuwan is right up there detecting almost 100% of true positives !!

OWASP Benchmark Results Comparison



The Benchmark contains *thousands of test cases* that are fully runnable and exploitable.

It considers *11 different types of vulnerabilities*, including several injection types such as XSS, weak encryption or trust boundary. For every type, the test cases have real vulnerabilities (true positives) and fake vulnerabilities (false positives) to challenge the tools.

Are you interested to know more detail on OWASP Benchmark & Kiuwan?

Please, read <https://www.kiuwan.com/blog/owasp-benchmark-kiuwan/>

Do you want to build and run by yourself the OWASP Benchmark with Kiuwan?

Then, have a look at <https://www.kiuwan.com/blog/owasp-benchmark-diy/>

How does Kiuwan help with security in Cobol, RPG and ABAP

Kiuwan provides for security vulnerabilities detection rules that cover so-called "legacy" languages.

Please visit below links for further info:

- Security in business-oriented languages: COBOL and RPG
- Security in business-oriented languages: ABAP

Why Kiuwan Code Security vulnerabilities do not match Code Analysis Security defects ?

Kiuwan Code Security checks not only for directly related security defects, but also for defects that indirectly might lead to a security flaw.

Then, Kiuwan Code Security reports as "vulnerabilities" not only defects coming from rules belonging to Security, but also defects coming from other characteristics that might indirectly lead to security issue.

If you open your Model and check "Only Code Security rules", you could browse all the rules that will produce defects considered as vulnerabilities.

For further info, please visit [Kiuwan Vulnerability Types](#)

Which are the main Indicators provided by Kiuwan

Kiuwan provides indicators for:

- **Software characteristics**
 - Security, efficiency, maintainability, reliability and portability
- **Global Indicator**
 - It is calculated as weighted average of the above software characteristics through a complex algorithm that has into account the severity of the defects, the weight of the category in which the defect is, the analyzed code volume and the criticality of the language for Kiuwan user. Kiuwan allows to "customize" this algorithm by modifying its level of demand, the weights of the category and the priority of the rules.
- **Effort to Target**
 - The amount of work effort needed to reach the defined goal. Objectives are defined at application level. These objectives are configurable. CQM has a repair effort assigned for each one of the more than 4,000 rules it incorporates. The sum of the repair efforts of each defect indicates the time needed to reach the targets.
- **Risk Index**
 - It is a summary index that concentrates all evidence found in the application source code and could be understood as the risk associated to the software defects found related to defined goals and effort to reach them. See below for further details.



Please, have a look at our [Kiuwan's Blog post](#) for a practical approach to understand Kiuwan Indicators

What's the meaning of Risk Index and how is calculated

Risk index represents the potential problems that you are assuming by not paying attention to the security and quality of your source code. In other words: how far you are (measured in effort) to get an acceptable level.

Risk index calculation concentrates all the evidence found in the source code of your application, and is calculated combining Global Indicator, Effort to Target and Code Size.

Therefore, if you have poor global indicator, but the effort needed to get better is low, you are not assuming a high risk in this application because you are going to repair your defects easily. But if the effort needed to get better is very high, your risk index will be high, too.

There's no simple "adequate" or critical threshold for Risk Index. As a rule of thumb, any value greater than 0 should be "observed", as it means that, based on defined goals, actions should be done to decrease it.

Pay attention to risk index evolution over time and use it as a metric to compare against multiple applications.

How does Kiuwan help me to make decisions on how to fix my application

Once you have obtained security and quality metrics and defects of your application, the most probably questions you will have will be some of the following:

- Where should I start to improve?
- How much time does it take me to repair each one of them?
- Which are the optimal path and action plan to reach my quality goals?
- I only have 20 hours to fix errors before the next delivery. What should I fix to aim the best possible quality?

Kiuwan provides a module to create **Action Plans**, i.e. a concrete and defined set of goals and actions to be performed on your application. Once defined, you will be able to share it (by exporting to PDF or as Jira issues) and track progress based on analysis results.

In order to create an Action Plan, you can follow two different approaches.

Based on current analysis defects, you might build your action plan based completely on your more important criteria (high security vulnerabilities, available man-power, etc.).

A different approach might be asking Kiuwan to build an optimized action plan. For that, Kiuwan provides a complete module (“**What-If**”) where Kiuwan will provide to you with concrete actions depending on the strategy you prefer: either setting specific goals for metrics (e.g. what are the actions and effort to reach a target of 90 in Security?) or specifying the effort you may invest in remediation tasks (e.g. with 40 h, how can I obtain the best gain?).

Whatever be the approach, once you decide the right simulation scenario, Kiuwan generates an Action Plan for you to implement the simulation. After generation of that action plan, you will be able to **track the Remediation Progress** (i.e. the consecution of that action plan).

Please visit [Kiuwan Code Analysis](#) and [Action Plans](#) for further info.

How does Kiuwan help me to manage and make strategic decisions on my Application Portfolio

Kiuwan is the only cloud solution that allows you to take decisions about your application portfolio, compare them, see the evolution, explore the best repair scenario to fit your needs and resources and help to decide if an application has to be conserved, optimized or replaced.

By using [Kiuwan Governance Doc](#) you will get:

- Complete visibility of your **entire** application portfolio
- Objective information to negotiate your **SLA's**
- Measure external **providers**, understand their path from a unique vantage point.

[Kiuwan Governance Doc](#) lets you make informed decisions through Enterprise Software Analytics tools such as:

- **Decision quadrants**
 - Detect risky applications using different decision quadrants
- **Evolution**
 - Predictive analytics of the evolution of your application's portfolio.
- **Activity**
 - Record the activity of your development teams and software vendors, both in application and maintenance projects or change requests.

Please visit [Kiuwan Governance](#) and [Kiuwan Governance doc site](#) for further info.

How can I obtain aggregated indicators and metrics for applications Governance

Kiuwan provides software analytics for a single application (Application), a group of applications (Portfolio) or the whole set of applications (Enterprise) levels.

At **Application Level**, you obviously will get information about the selected application.

At **Enterprise Level**, you will get aggregated information for all the applications.

At **Portfolio Level**, you will get metrics based on any grouping criteria you can define. To group applications by some defined criteria, first you have to decide the criteria by which you want to group your applications. For example, the software development team, the software vendor that has developed the application, the application business value –by default already available on Kiuwan-; or any other criteria relevant for you or your organization.

Once you have decided this, you are able to create a portfolios group with the possible values of the specific criteria it can take (portfolio). For example, for business value, the possible values could be: critical, high, medium, low and very low.

Afterwards, you can assign any application to an existing portfolio, establishing the value it will have for that application. My on-line banking application, for instance, might belong to business-critical applications portfolio.

By grouping the applications into portfolios, you can manage the analytics (security, productivity, activity, evolution, etc.) of your applications at any level, which means that Kiuwan calculates all indicators for the portfolios and the portfolios groups, based on the applications data.

Kiuwan provides a **Governance** module where you can fully exploit your whole enterprise software analytics data, letting you to drive conclusions and insights that will help you in your decision making process.

Please visit [Kiuwan Governance](#) and [Kiuwan Governance doc site](#) for further info.

How does Kiuwan help me to build Decision Quadrants on my Application Portfolio

Kiuwan provides **Decision Quadrants** both at Portfolio and Global Levels.

At Portfolio Level, Kiuwan positions portfolios or applications in a graph with 4 quadrants: Replace, Remediate, Observe and Conserve; based on their exposure to development risk and quality. This way you can decide how to improve the health of your portfolios or applications.

At Global Level, Kiuwan provides four decision quadrants:

- **Business Value Decision Quadrant**
 - Aimed to identify those applications in your portfolio that require immediate action based on their criticality for the business and their exposure to any of risks you are facing: Global Risk (Risk index), Failure Probability (Production Risk), Maintenance (Development Risk) and Security Risk.
- **Production Decision Quadrant**
 - Aimed to identify those applications in your portfolio that could cause problems in production (reliability issues), and if they will be able to recover from these errors easily.
- **Development Decision Quadrant**
 - Aimed to identify those applications in your portfolio exposed in the midterm given the difficulty and associated cost to maintain them.
- **Security Decision Quadrant**
 - Aimed to identify those applications in your portfolio that are exposed to potential internal or external attacks, that can compromise the integrity of your organization, and if these potential vulnerabilities can be easily corrected.

Please visit [Kiuwan Governance](#) and [Kiuwan Governance doc site](#) for further info.

How does Kiuwan help me to manage Software Vendors

By using Kiuwan Governance, you will be able to

- Quantify : Define service-level agreements (SLA) to be met by each provider and verify compliance
- Decide : Informed data and vendor ranking
- Negotiate : Unique vantage point to negotiate SLA's

Please visit [Kiuwan Governance](#) and [Kiuwan Governance doc site](#) for further info.

Does Kiuwan let me to manage software development Life Cycle

[Kiuwan Life Cycle](#) lets you to sensibly reduce development time, testing and integration. Audit, monitor and automatically analyze change requests within their respective environments.

Life Cycle puts you in full control of your application's deliveries from the start:

- State management for change requests or development projects (in progress, resolved, etc).
- Decide whether a new version of an application is ready to promote based on its status.
- Automate the entire process making Kiuwan connect with your continuous integration system (eg. Jenkins).

Kiuwan Life Cycle lets you:

- **Compare baseline modifications** in order to detect new defects during the development process.
- **Define checkpoints and audits** tailored to each type of project or change request.
- **Check control points** continuously during the construction or maintenance phases to ensure that applications do not degrade over time after modifications.

Please visit [Kiuwan Life Cycle](#) and [Kiuwan Life Cycle doc site](#) for further info.

What about Team management

Kiuwan Life Cycle allows to :

- Define permissions and roles for your users

- Control what information and what actions every member of the team can perform, reporting non-compliance and effort required to repair deliveries.
- Control the work being done by each development team or each software vendor

Please visit [Kiuwan Life Cycle](#) and [Kiuwan Life Cycle doc site](#) for further info.

What are the requirements to use Kiuwan

Kiuwan is a cloud-based solution, so to access Kiuwan website you only need Internet access to following URLs:

- Kiuwan dashboard and web site:
 - <https://www.kiuwan.com>
 - <https://api-iam.intercom.io>
 - <https://static.intercomassets.com>
 - <https://js.intercomcdn.com>
 - <https://nexus-websocket-a.intercom.io/>
- Additionally, if you uses Kiuwan Local Analyzer (see next section on using kiuwan without upload source code)
 - <https://api.kiuwan.com>

Can I use Kiuwan without uploading my code

There is no need to upload your code if you do not want to do it for security reasons.

You can **analyze your code locally** downloading the Local Analyzer, running it in your infrastructure and uploading (encrypted) the results of your analysis, process them securely and seeing the results in Kiuwan cloud as if you had uploaded the code.

Secure Socket Layer (**SSL**) technology protects information sent to Kiuwan using encryption and authentication server both of your computer and data between the data center, ensuring that your data in transit is safe, secure and available only to registered users in your organization.

Kiuwan has been designed from the beginning as a **SaaS solution** and, therefore, it is "multitenant", which prevents that anyone but the owner of an application and its data could access to data that are not theirs (malicious users, hacker, back doors, system errors, information in logs, etc.). Kiuwan also offers an **On Premise version** if you want to install it completely at your own facilities. Please contact [Kiuwan Technical Support](#) for on-premise solution.

When using Local Analyzer, analysis is performed locally in customer computer and results are sent to Kiuwan site. No source file is uploaded to Kiuwan, only the results of the analysis (defects, metrics, etc.).

In order to see where in the code errors or security vulnerabilities are, you can choose:

- To upload the lines of code where they are located (Kiuwan shows them in defects report).
- Not to upload anything of your code at all (in this case, Kiuwan shows just the line numbers where errors or vulnerabilities have been found).

Visit [Kiuwan Local Analyzer](#) for further info.

How can I download Kiuwan Local Analyzer

To download Kiuwan Local Analyzer you must have an active **Kiuwan Account**.

If you do not have yet, you can *signup for a free trail* account at <https://www.kiuwan.com/signup/free/>

Once you have your Kiuwan Account activated, create your first application (if not already done) and click on **Analyze** and decide how to analyze the code:

1. **in the cloud**, "uploading" the code of your application, or
2. **locally**, by downloading and installing the **Kiuwan Local Analyzer** on a local machine and sends the results to the server

Visit [Analysis Management](#) to find additional info.

Visit [Kiuwan Local Analyzer](#) to find installation instructions and requirements.

What programming languages are supported by Kiuwan

Please visit [Kiuwan Supported Technologies](#) to view the list of languages supported by Kiuwan so far.

For other languages not included, please contact [Kiuwan Technical Support](#).

What is the Kiuwan support for SQL

Kiuwan performs SQL analysis by providing specific support for PL-SQL and Transact-SQL, Informix and Sybase.

Kiuwan also provides support to analyze SQL code embedded into Java and Cobol source code.

Does Kiuwan support mobile technologies

Android is supported (specific Java rules for Android, besides Java general rules), as well as Objective-C (iPhone and iPad).

Does Kiuwan support PHP Symfony

For PHP analyses, Kiuwan provides a specific model (different from CQM) only for PHP Symfony with PHP, Javascript and Symphony rules. There are also Symfony rules in the rules library in case you need to add them to your own model.

How to analyze ABAP code

Basically, to analyze ABAP code in Kiuwan is firstly needed to extract ABAP source code from SAP server. After extracting the code, zip it an upload it to the cloud or analyze it locally.

To make these steps easy, Kiuwan provides a [\(old\) ABAP source code extractor](#).

You can find detailed guides about SAP and Kiuwan in the ABAP Series in our blog:

1. [ABAP Code Quality & Security Vulnerabilities detection](#)
2. [Static analysis for ABAP](#)
3. [ABAP: continuous analysis with Kiuwan](#)
4. [Perform Kiuwan analysis in your ABAP Development Life Cycle](#)

How to analyze C/C++ code

C and C++ are fully supported by Kiuwan. You can analyze C/C++ code in the cloud, of course. But C and C++ is heavily dependant on compiler directives that are required to fine tune the analysis. Typical cases are macros and header files resolution.

To fully support these options, you should use Kiuwan Local Analyzer to properly configure those options.

Please visit [Configuring C or Cpp analysis](#) for further info.

You can also find usefull and step-to-step guide on how to configure C/C++ analysis in the following posts of [Kiuwan Blog](#)

1. [Analyzing C and C++ with Kiuwan](#)
2. [C/C++ analysis with Kiuwan, part II](#)

How to analyze OracleForms code

To analyze OracleForms code, the Forms binary files (*.fmb) must be converted to XML format (you can use Oracle conversion tool). After extracting the code, zip it an upload it to the cloud or analyze it locally.

Please visit [Analyzing Oracle Forms](#) for further info.

Does Kiuwan detect duplicated code

Reusing code is usual in software development, but this practice makes the code **less maintainable**, besides introducing defects. That's why we have the Kiuwan **Clone Detector**.

Please visit [Avoid duplicated code with Kiuwan Clone Detector](#) for further info.

How to use Kiuwan in Continuous Integration (Jenkins, for example)

Developers and integrators can connect to Kiuwan by different means.



Continuous Integration

Please visit [Developers - Integrations](#) for a full listing of possibilities.

How could I integrate Kiuwan results in my own Dashboard

Kiuwan provides a [REST API](#) in case you need to extract analysis results from Kiuwan.

It is useful to integrate Kiuwan information in your own dashboard, or (for example) if you want to block a promotion between environments based on the analysis results.

Does Kiuwan integrate with JIRA

Defect founds by Kiuwan and incorporated to an Action Plan can generate automatically incidences in JIRA, accelerating the step between the certification of an application and the remediation of the founded issues. Please, visit [Action Plans - Export to JIRA](#) and [Jira integration](#) for further info.

Does Kiuwan allow the development of custom quality rules

Yes. Besides available rules, Kiuwan provides fully support for the definition, creation and analysis of custom rules.

With Kiuwan Local analyzer, we will have access to Kiuwan Rule Developer, a GUI tool that will help you to create, run and edit rules that can be executed in a Kiuwan analysis. After creating, running and debugging our custom rules with Rule Developer, Kiuwan allows you to install and use them in a Quality Model, fully integrating your custom rules with the rules provided by Kiuwan library.

Please visit [Rule development](#) for further info.

You can also find usefull and step-to-step guide on how to develop and run custom rules in the following posts of [Kiuwan Blog](#)

1. [Rules development \(I\): Where do we start?](#)
2. [Rules development \(II\): basis for implementation](#)
3. [Rules development \(III\): debugging custom rules](#)
4. [Rules development \(IV\): Basic API – navigating through the AST](#)
5. [Rules development \(V\): Query API](#)
6. [How to create new zero-code Kiuwan rules](#)

Does Kiuwan integrate results from 3rd party static analyzers

Yes. Kiuwan can execute rules from PMD, CheckStyle and FindBugs.

But Kiuwan also allows you to import results from any other static analyzer.

You only need to transform those reports to Kiuwan input format and attach those results to local analyzer execution.

You can find examples of transformers for HP Fortify and MS-FxCop on: <https://github.com/kiuwan/thirdparty-report-importer>

Please visit below links for further info:

- [Importing data from PMD, Checkstyle, Findbugs and checkKingQA](#)
- [Third party analyzers](#)
- [How to integrate Ruby in Kiuwan](#)

Does Kiuwan integrate with LDAP

Yes. If your company has a corporate authentication service implemented, your users and passwords will most probably be stored in an Active Directory, an OpenLDAP or an IBM Tivoli. If that is your case, you don't want to have a different password for your Kiuwan account. By integrating Kiuwan with your **LDAP** service, you get to delegate your company users authentication in it. Kiuwan allows you to configure the authentication of your account users with your own LDAP service.

Most of our customers use Microsoft Active Directory as a repository for user credentials. However, visit [How to configure Kiuwan authentication with your own LDAP service](#) for a procedure valid to integrate Kiuwan with any other users repository, even if it's not based on LDAP.

What is a Kiuwan application

A kiuwan application is the piece of code you want to analyze and track. It can be all the code of your company's new web app, a single module of your application you want to track specifically, that open source project you are working on, etc.

A Kiuwan application can contain as many languages as you wish. Kiuwan will analyze them all and will give you relevant info for each language and for whole application, as well.

What is a Kiuwan Model? Should I start from scratch

To analyze an application source code, it is necessary to configure a Quality Model. It requires a great knowledge on the repository of hundreds of rules that help you to validate the code, how to select and parameterize them... The same task is needed for the set of metrics that Kiuwan supports. And still remain some other configuration details for other Kiuwan indicators.

To help this process, Kiuwan provides Checking Quality Model for Software (CQM). CQM is a model for assessing the internal security and quality of a software product, designed by Optimyth and available 'out-of-the-box' in Kiuwan, so that users can begin to analyze the quality of their code immediately and, once known the methodology behind code quality certification, they will be able to "calibrate" the model, develop new models from it or from scratch, etc.

Visit [Models Manager User Guide](#) for further info.

How can I deactivate a rule

If you are using a custom model (i.e. some model created by you) the process is quite straightforward. Just:

1. find the rule in your model (Model Management -> Rules),
2. click on the circle icon in the Active column to deactivate the rule, and
3. publish again the model.

But if you are using CQM, take in mind that CQM is a read-only model, so you should first create your own model (it might be done as a copy of CQM) and proceed as above to deactivate the rule.

In case you need help on how to create a new model, please visit [Advanced Model Management](#).

How can I mute defects

For Kiuwan, a defect is a violation to a rule defined in the model for a specific language and a software characteristic. Kiuwan provides a full list of all detected defects found in the source code of an application.

Defects found on generated code can be ignored (muted) through Kiuwan Defect Mute functionality. For each muted defect, you can specify a reason why it is muted, as well as a comment. For example, if it is a false positive in a file, a line of code that must not be taken into account, a file that has too many defects of the same type or a line of code that has been generated and does not have to be taken into account, or any other basis that is reasonable for the user.

Through this functionality you can mute defects for the whole application, a specific file or simply for a specific line of code in a file by dragging and dropping the rule that triggers the defects, the file or the line of code. You can perform this action in muted defects tab on Defects screen.

You can check our blog post about [Suppress false positives in your code analysis](#)

How to manage Kiuwan defects when I do not completely agree with them

Some times you might consider as defects some of the defects found by Kiuwan.

What to do in those cases ? Please have a look at [How to manage Kiuwan defects when I do not completely agree with them](#)