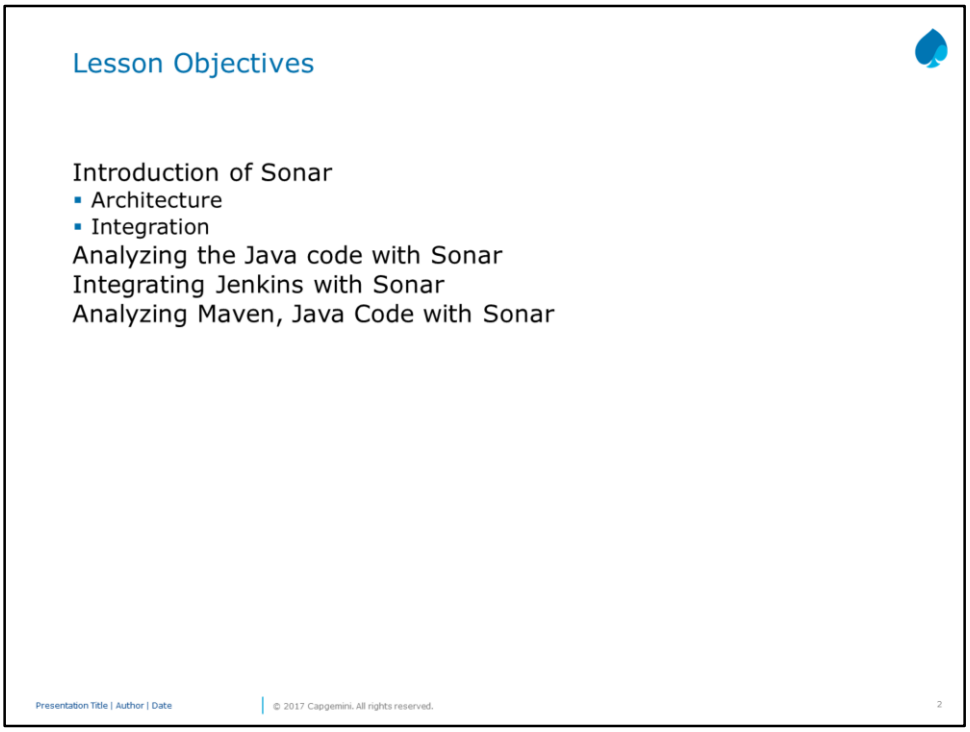
Instructor Notes:

Add instructor notes here.

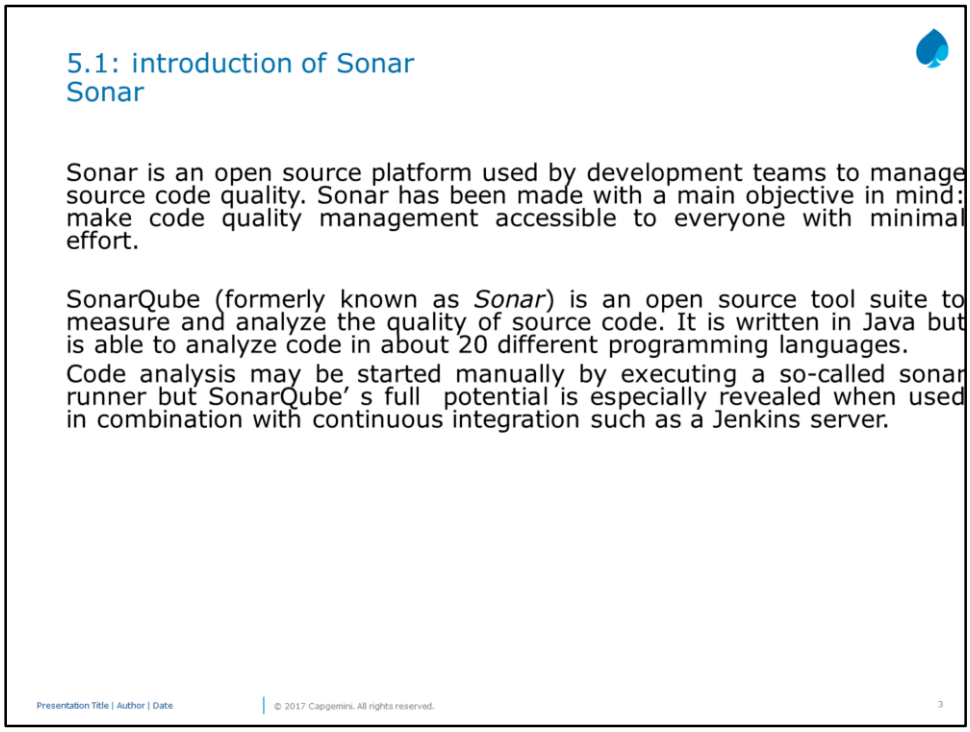
Instructor Notes:

Add instructor notes here.



Instructor Notes:

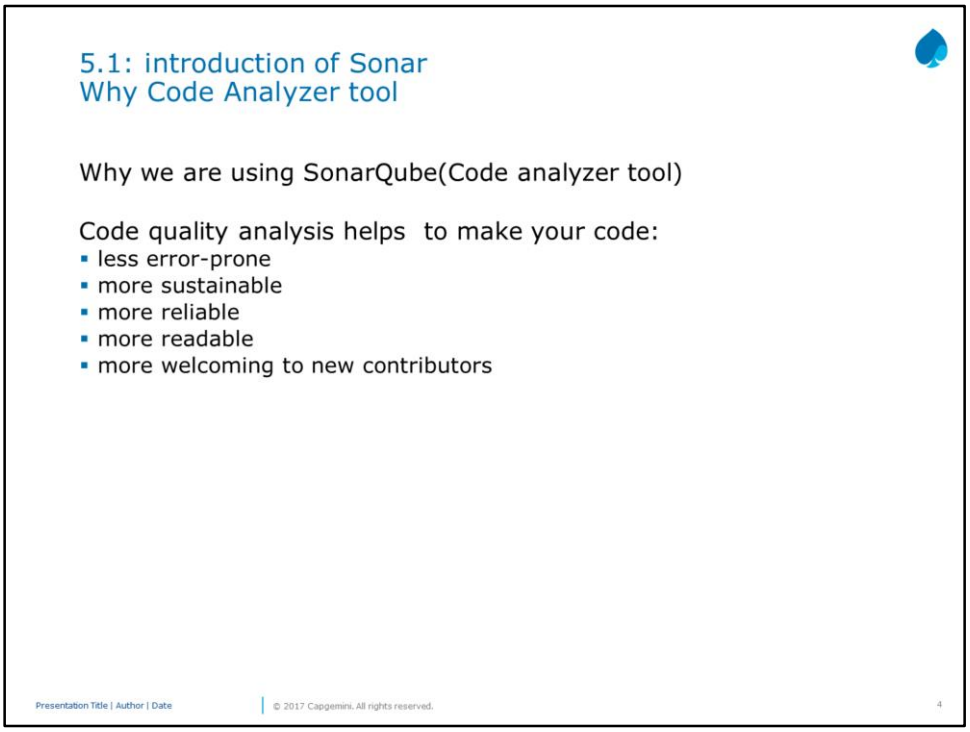
Add instructor notes here.



Add the notes here.

Instructor Notes:

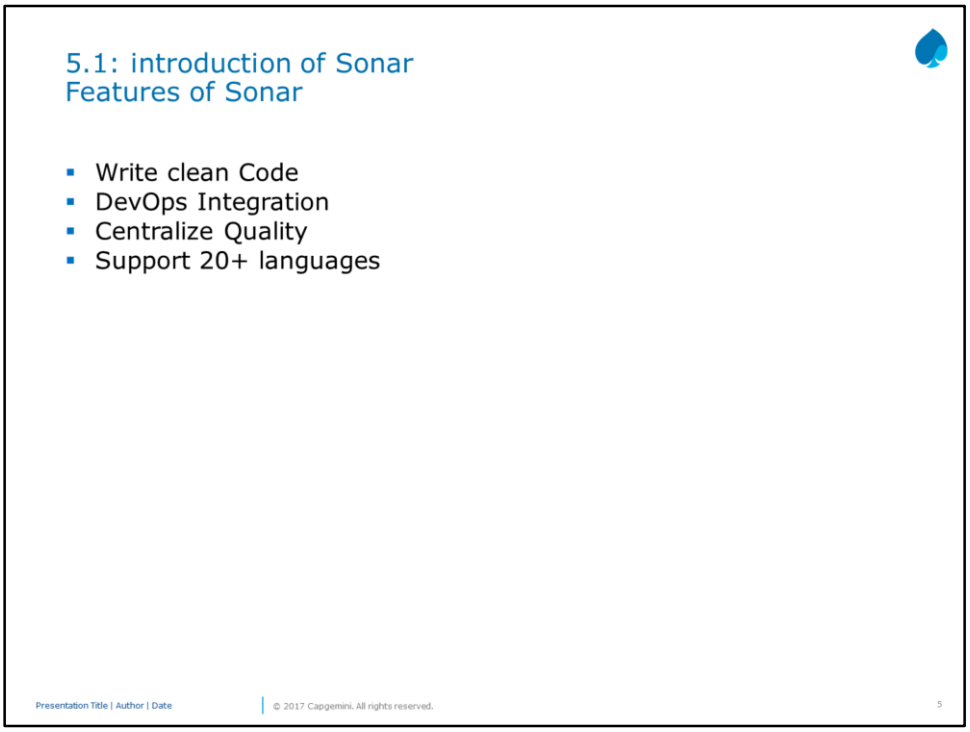
Add instructor notes here.



Code quality analysis mainly relies on a set of tools that look at your code and give you hints. The most famous tools are Findbugs, PMD, Checkstyle but also code coverage tools such as Jacoco. JDT itself provides very powerful quality checks, but there are not enabled by default. You should go to Error/Warnings in preferences and replace all "ignore" by "Warning".

Instructor Notes:

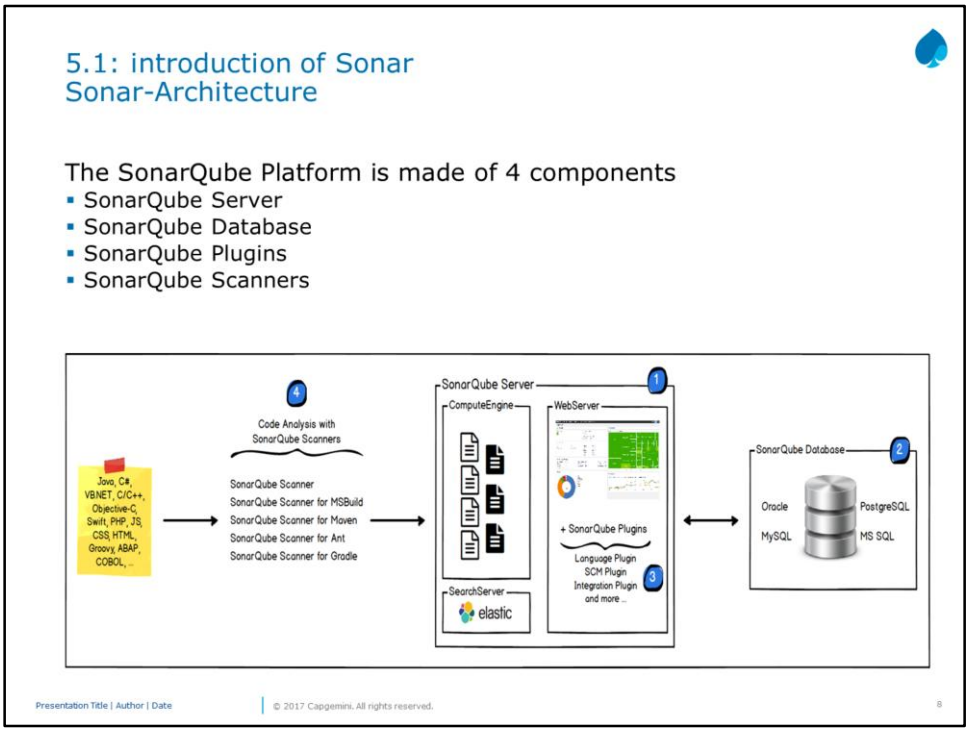
Add instructor notes here.



Write clean code
Overall health
Your project home page shows where you stand in terms of quality in a glimpse of an eye. This main page also shows you an immediate sense of the good results achieved over time.
Focus on the Leak
The water leak paradigm is a simple yet powerful way to manage code quality: quality of new - changed and added - code should be put under control before anything else. Once that Leak is under control, code quality will start improving mechanically. In SonarQube, the Leak is a built-in concept that you can't miss. Once you've had a look at this yellow area on the left of your project home page, you will always remain focused on it to not miss any new issues.
Enforce Quality Gate
With SonarQube, a developer has everything at hand to take ownership of the quality of his code. To fully enforce a code quality practice across all teams, you need to set up a Quality Gate. This core concept of SonarQube is a set of requirements that tells whether or not a new version of a project can go into production. SonarQube's default Quality Gate checks what happened on the Leak period and fails if your new code got worse in this period.
Analyze pull requests
Once you have SonarQube in place, you will quickly want to make sure you add as few issues as possible to your code base. To shorten the feedback loop so you don't have to wait for new analyses to be available on SonarQube, you can set up the analysis of your pull requests. Analyses will be run on your feature branches without being pushed to SonarQube, giving you the opportunity to fix issues before they ever reach SonarQube!

Instructor Notes:

How sonar works give idea.



The SonarQube Platform is made of 4 components:
One **SonarQube Server** starting 3 main processes:

- a **Web Server** for developers, managers to browse quality snapshots and configure the SonarQube instance
- a **Search Server** based on Elasticsearch to back searches from the UI
- a **Compute Engine Server** in charge of processing code analysis reports and saving them in the SonarQube Database

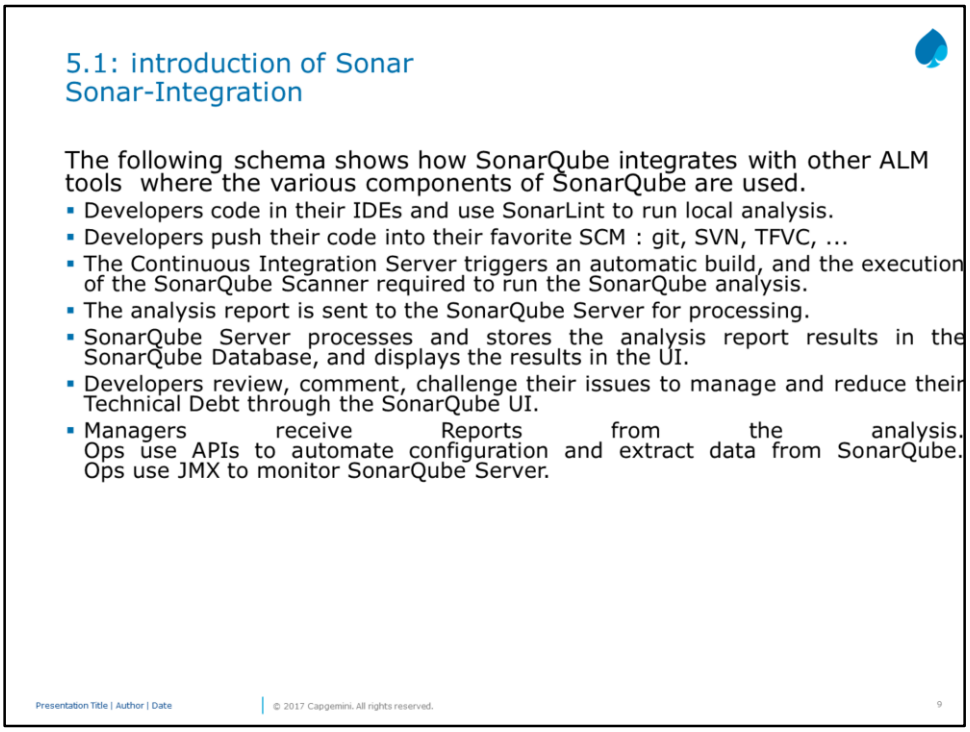
One **SonarQube Database** to store:

- the configuration of the SonarQube instance (security, plugins settings, etc.)
- the quality snapshots of projects, views, etc.

Multiple **SonarQube Plugins** installed on the server, possibly including language, SCM, integration, authentication, and governance plugins
One or more **SonarQube Scanners** running on your Build / Continuous Integration Servers to analyze projects

Instructor Notes:

Add instructor notes here.



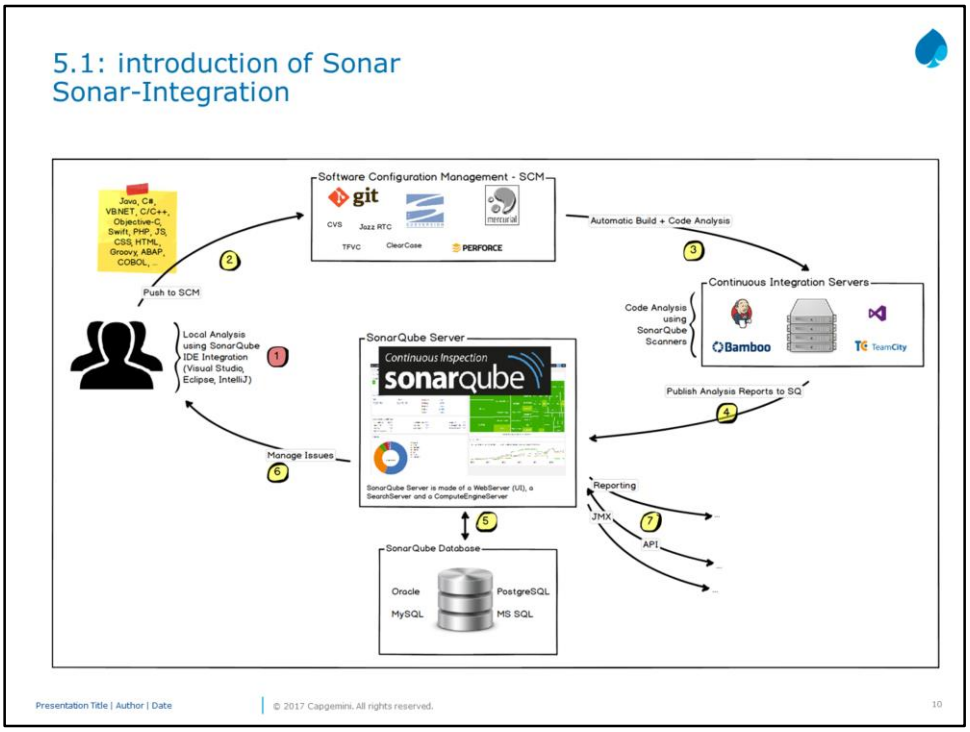
The SonarQube Platform is made of 4 components:
One **SonarQube Server** starting 3 main processes:
- a **Web Server** for developers, managers to browse quality snapshots and configure the SonarQube instance
- a **Search Server** based on Elastic search to back searches from the UI
- a **Compute Engine Server** in charge of processing code analysis reports and saving them in the SonarQube Database

One **SonarQube Database** to store:
- the configuration of the SonarQube instance (security, plugins settings, etc.)
- the quality snapshots of projects, views, etc.

Multiple **SonarQube Plugins** installed on the server, possibly including language, SCM, integration, authentication, and governance plugins
One or more **SonarQube Scanners** running on your Build / Continuous Integration Servers to analyze projects

Instructor Notes:

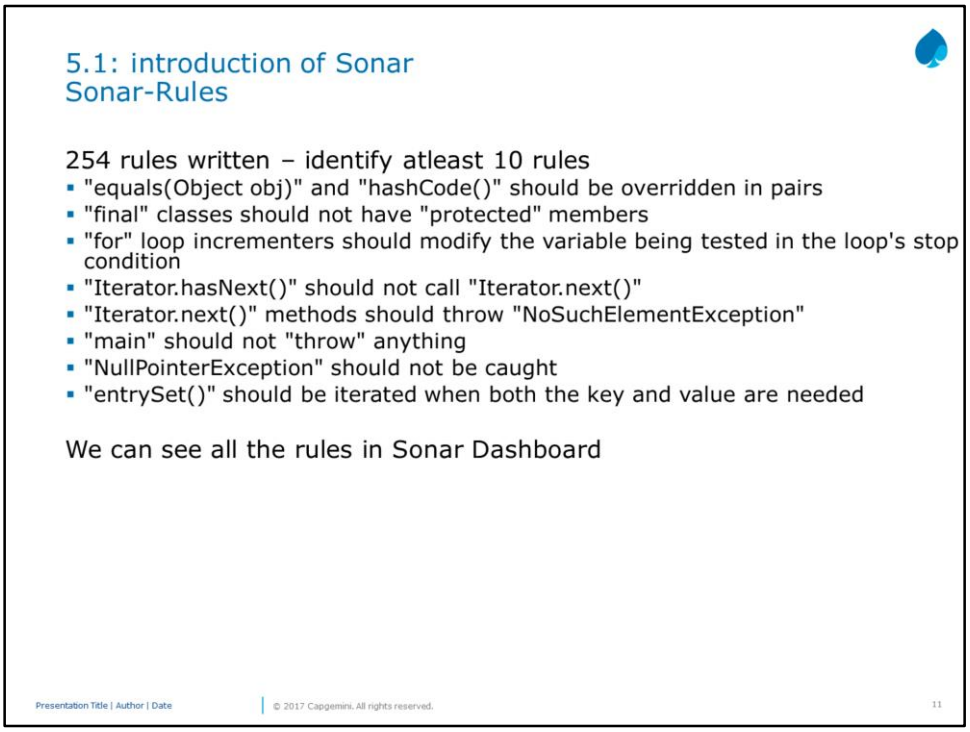
Give idea of flow



About Machines and Locations
The SonarQube Platform cannot have more than one SonarQube Server and one SonarQube Database.
For optimal performance, each component (server, database, scanners) should be installed on a separate machine, and the server machine should be dedicated.
SonarQube Scanners scale by adding machines.
All machines must be time synchronized.
The SonarQube Server and the SonarQube Database must be located in the same network
SonarQube Scanners don't need to be on the same network as the SonarQube Server.
There is **no communication** between **SonarQube Scanners** and the **SonarQube Database**.

Instructor Notes:

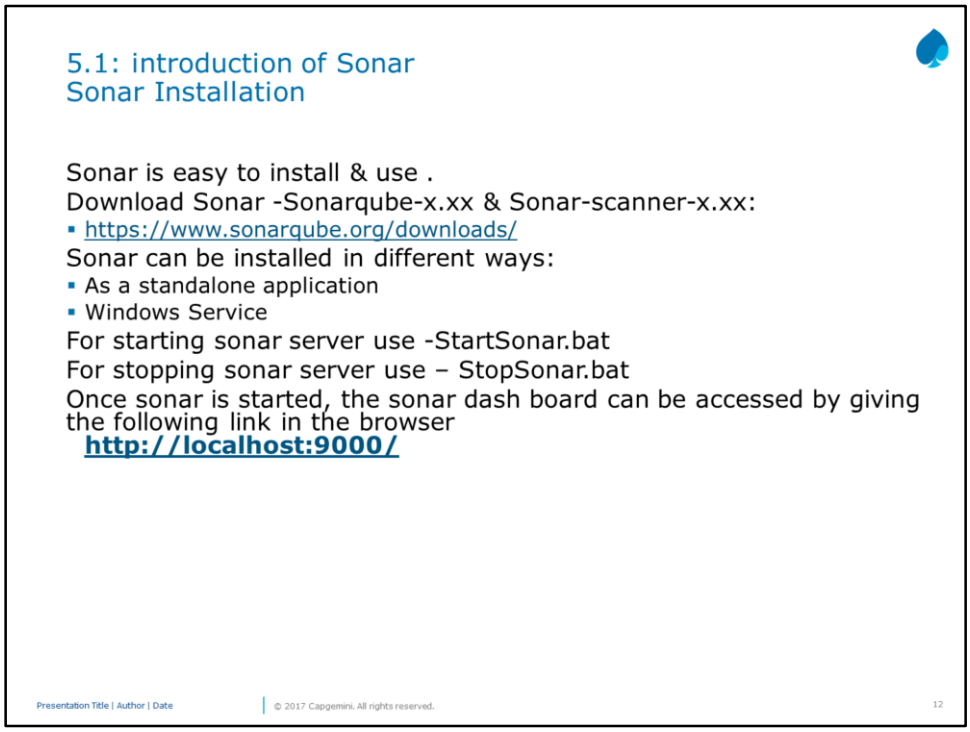
Add instructor notes here.



About Machines and Locations
The SonarQube Platform cannot have more than one SonarQube Server and one SonarQube Database.
For optimal performance, each component (server, database, scanners) should be installed on a separate machine, and the server machine should be dedicated.
SonarQube Scanners scale by adding machines.
All machines must be time synchronized.
The SonarQube Server and the SonarQube Database must be located in the same network
SonarQube Scanners don't need to be on the same network as the SonarQube Server.
There is **no communication** between **SonarQube Scanners** and the **SonarQube Database**.

Instructor Notes:

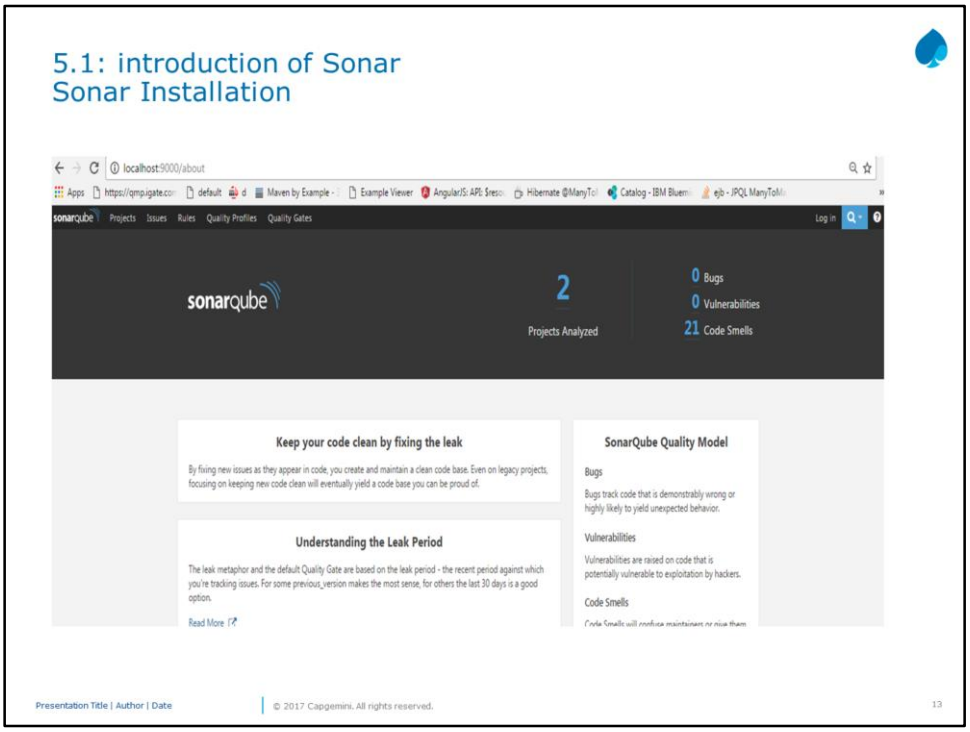
Give Demo example



Add the notes here.

Instructor Notes:

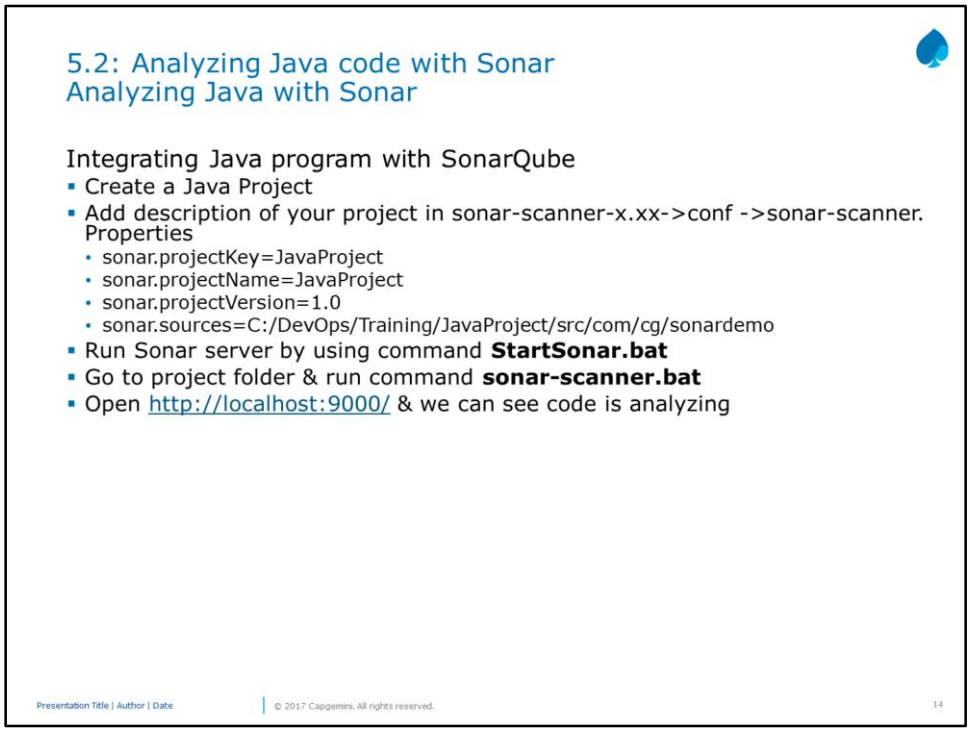
Add instructor notes here.



Add the notes here.

Instructor Notes:

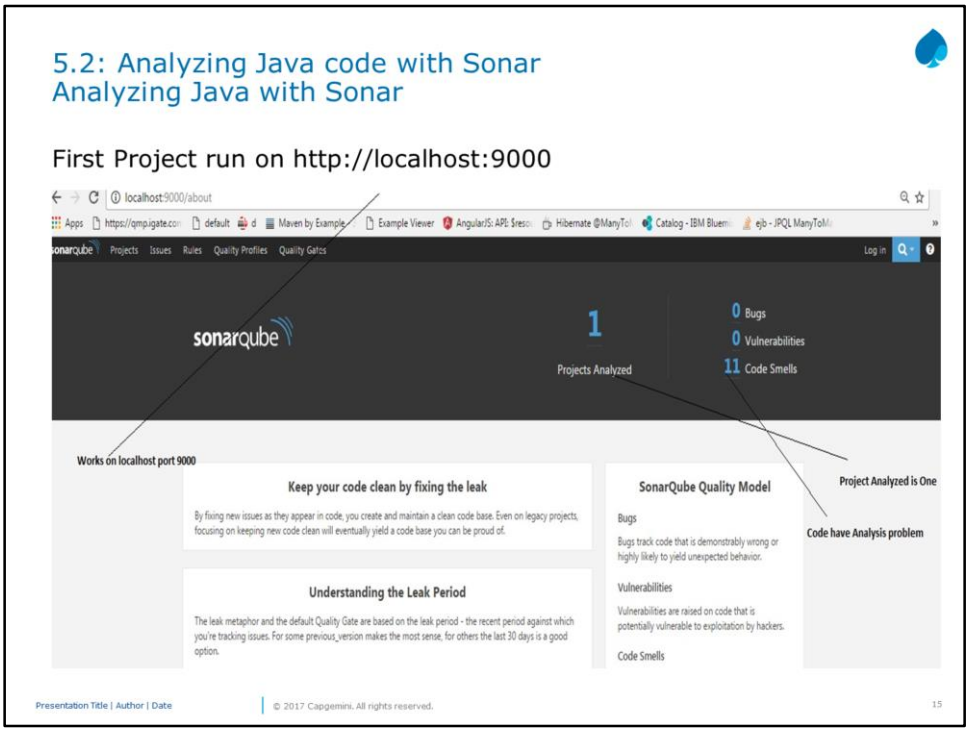
Add instructor notes here.



Add the notes here.

Instructor Notes:

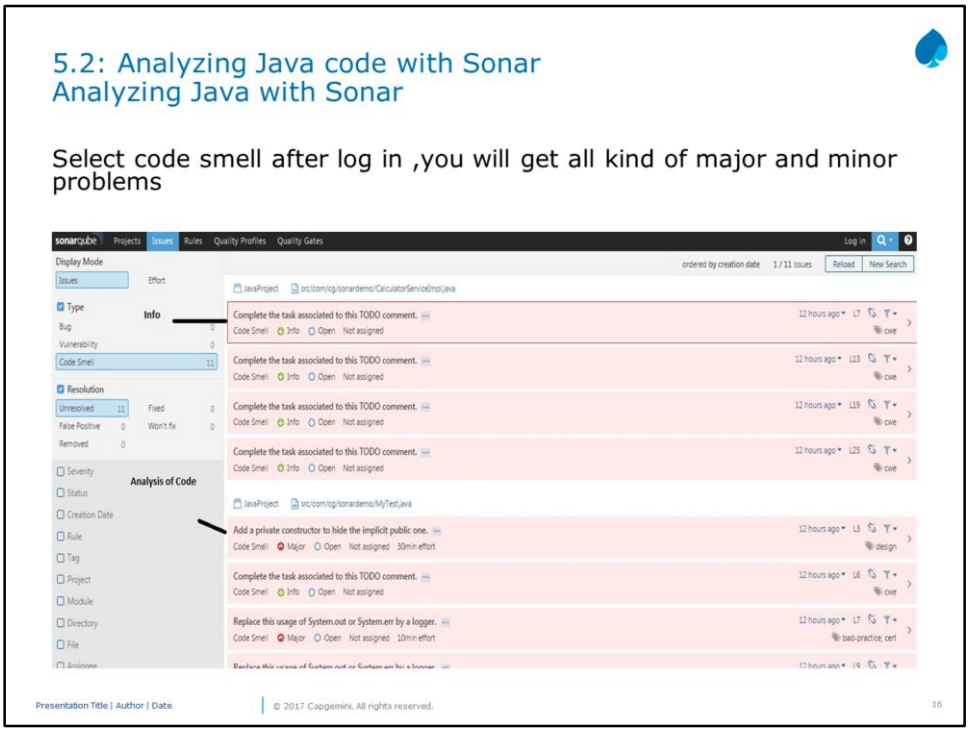
Add instructor notes here.



Add the notes here.

Instructor Notes:

Give Demo example

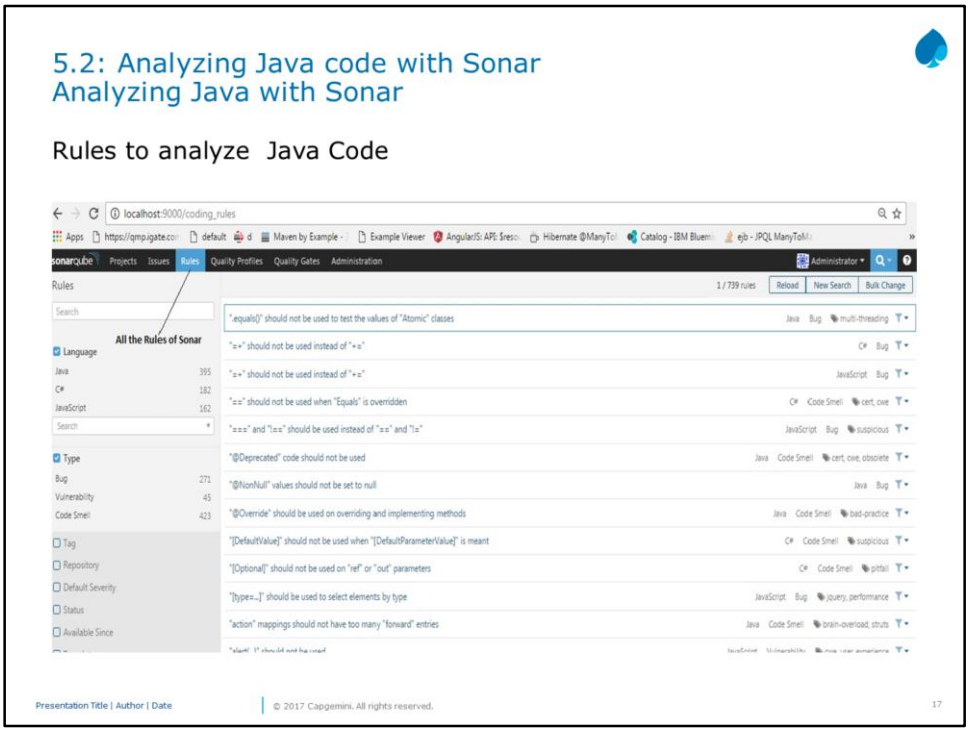


Each issue has one of five severities:
BLOCKER
Bug with a high probability to impact the behavior of the application in production: memory leak, unclosed JDBC connection, ... The code MUST be immediately fixed.
CRITICAL
Either a bug with a low probability to impact the behavior of the application in production or an issue which represents a security flaw: empty catch block, SQL injection, ... The code MUST be immediately reviewed.
MAJOR
Quality flaw which can highly impact the developer productivity: uncovered piece of code, duplicated blocks, unused parameters, ...
MINOR
Quality flaw which can slightly impact the developer productivity: lines should not be too long, "switch" statements should have at least 3 cases, ...
INFO
Neither a bug nor a quality flaw, just a finding.
Technical Review
Confirm, False Positive, Won't Fix, Change Severity, and Resolve fall into this category, which presumes an initial review of an issue to verify its validity. Assume it's time to review the technical debt added in the last review period - whether that's a day, a week, or an entire sprint. You go through each new issue and do one:

Confirm - By confirming an issue, you're basically saying "Yep, that's a problem." Doing so moves it out of "Open" status to "Confirmed".
False Positive - Looking at the issue in context, you realize that for whatever reason, this issue isn't actually an ~~issue~~, erm... "problem." It's not actually a problem. So you mark it False Positive and move on. Requires Administer Issues permission on the project.
Won't Fix - Looking at the issue in context, you realize that while it's a valid issue it's not one that actually needs fixing. In other words, it represents accepted technical debt. So you mark it Won't Fix and move on. Requires Administer Issues permission on the project.
Change Severity - This is the middle ground between the first two options. Yes, it's a problem, but it's not as bad a problem as the rule's default severity makes it out to be. Or perhaps it's actually far worse. Either way, you adjust the severity of the issue to bring it in line with what you feel it deserves. The marker in the drilldown will change to show the new severity immediately, but the change won't be reflected in your issue counts until after the next analysis. Requires Administer Issues permission on the project.
Resolve - If you think you've fixed an open issue, you can Resolve it. If you're right, the next analysis will move it to closed status. If you're wrong, its status will go to re-opened.
.

Instructor Notes:

Add instructor notes here.



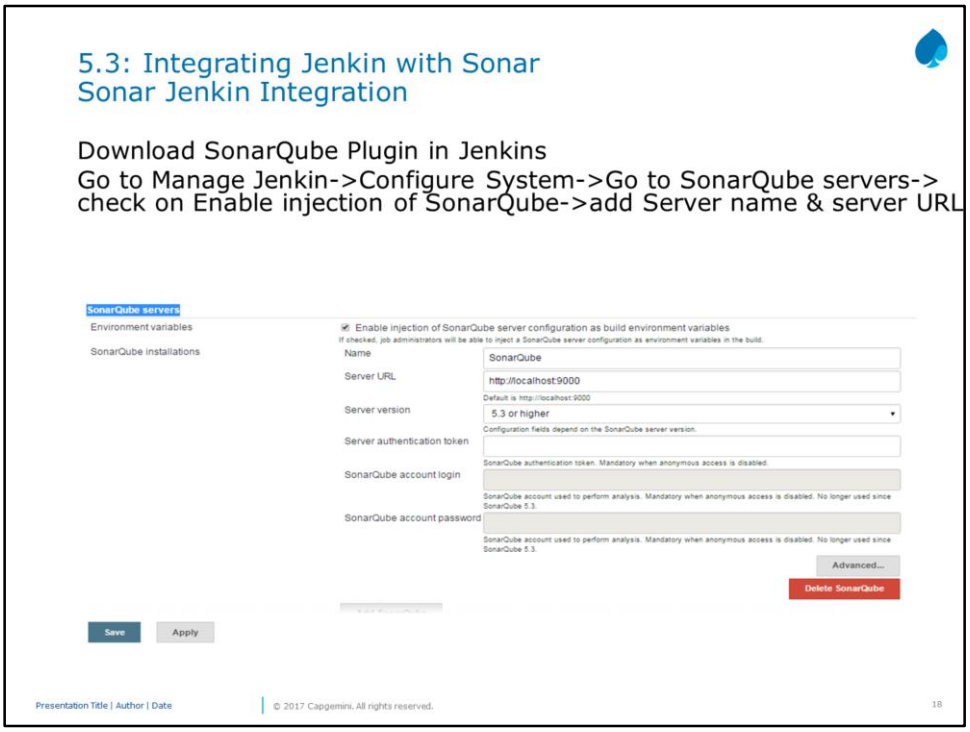
Click on the top "Rules" menu item to enter the world of rules. By default, you will see all the available rules, with the ability to narrow the selection based on search criteria in the left pane:
Language: the language to which a rule applies.
Type: Bug, Vulnerability or Code Smell rules
Tag: it is possible to add tags to rules in order to classify them and to help discover them more easily.
Repository: the engine that contributes rules to SonarQube.
Default Severity: the original severity of the rule - as defined by the plugin that contributes this rule.
Status: rules can have 3 different statuses:

Beta: The rule has been recently implemented and we haven't gotten enough feedback from users yet, so there may be false positives or false negatives.
Deprecated: The rule should no longer be used because a similar, but more powerful and accurate rule exists.
Ready: The rule is ready to be used in production.

Available Since: date when a rule was first added on the SonarQube instance. This is useful to list all the new rules since the last upgrade of a plugin for instance.
Template: display rule templates that allow to create custom rules (see later on this page).
Quality Profile: inclusion in or exclusion from a specific profile
To see the details of a rule, either click on it, or use the right arrow key. Along with basic rule data, you'll also be able to see which, if any, profiles it's active in and how many open issues have been raised with it.
The 2 following actions are available only if you have the right permissions ("Administer Quality Profiles and Gates"):
Add/Remove Tags:
It is possible to add existing tags on a rule, or to create new ones (just enter a new name while typing in the text field).
Note that some rules have built-in tags that you cannot remove - they are provided by the plugins which contribute the rules.
Extend Description:
Extending rule descriptions is useful to let users know how your organization is using a particular rule for instance or to give more insight on a rule.
Note that the extension will be available to non-admin users as a normal part of the rule details.

Instructor Notes:

Add instructor notes here.



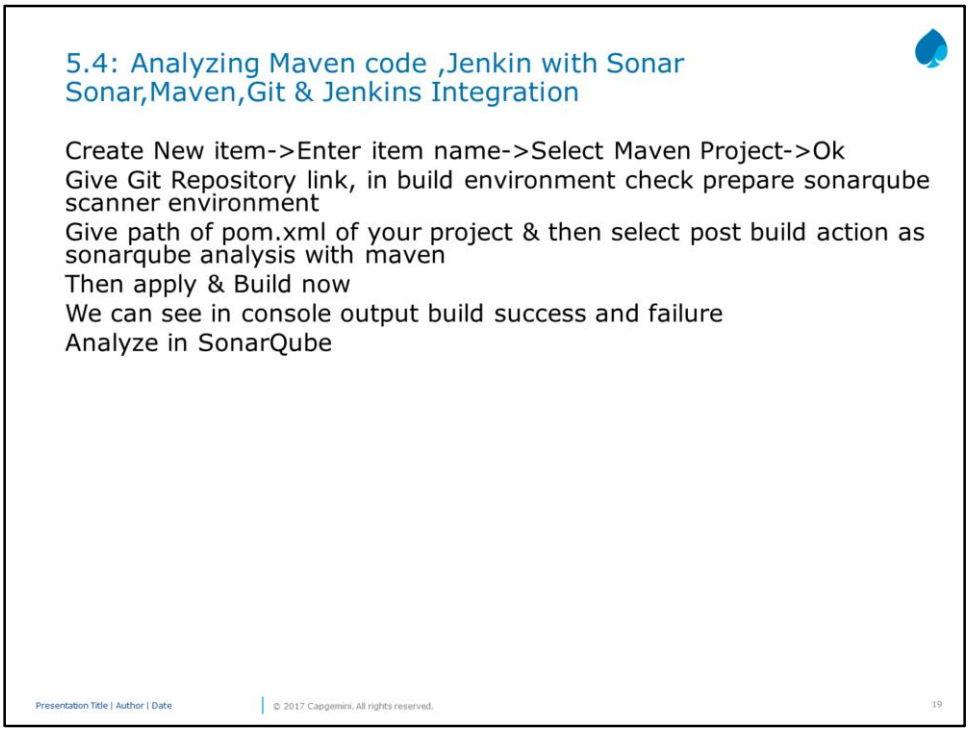
Click on the top "Rules" menu item to enter the world of rules. By default, you will see all the available rules, with the ability to narrow the selection based on search criteria in the left pane:
Language: the language to which a rule applies.
Type: Bug, Vulnerability or Code Smell rules
Tag: it is possible to add tags to rules in order to classify them and to help discover them more easily.
Repository: the engine that contributes rules to SonarQube.
Default Severity: the original severity of the rule - as defined by the plugin that contributes this rule.
Status: rules can have 3 different statuses:

Beta: The rule has been recently implemented and we haven't gotten enough feedback from users yet, so there may be false positives or false negatives.
Deprecated: The rule should no longer be used because a similar, but more powerful and accurate rule exists.
Ready: The rule is ready to be used in production.

Available Since: date when a rule was first added on the SonarQube instance. This is useful to list all the new rules since the last upgrade of a plugin for instance.
Template: display rule templates that allow to create custom rules (see later on this page).
Quality Profile: inclusion in or exclusion from a specific profile
To see the details of a rule, either click on it, or use the right arrow key. Along with basic rule data, you'll also be able to see which, if any, profiles it's active in and how many open issues have been raised with it.
The 2 following actions are available only if you have the right permissions ("Administer Quality Profiles and Gates"):
Add/Remove Tags:
It is possible to add existing tags on a rule, or to create new ones (just enter a new name while typing in the text field).
Note that some rules have built-in tags that you cannot remove - they are provided by the plugins which contribute the rules.
Extend Description:
Extending rule descriptions is useful to let users know how your organization is using a particular rule for instance or to give more insight on a rule.
Note that the extension will be available to non-admin users as a normal part of the rule details.

Instructor Notes:

Add instructor notes here.



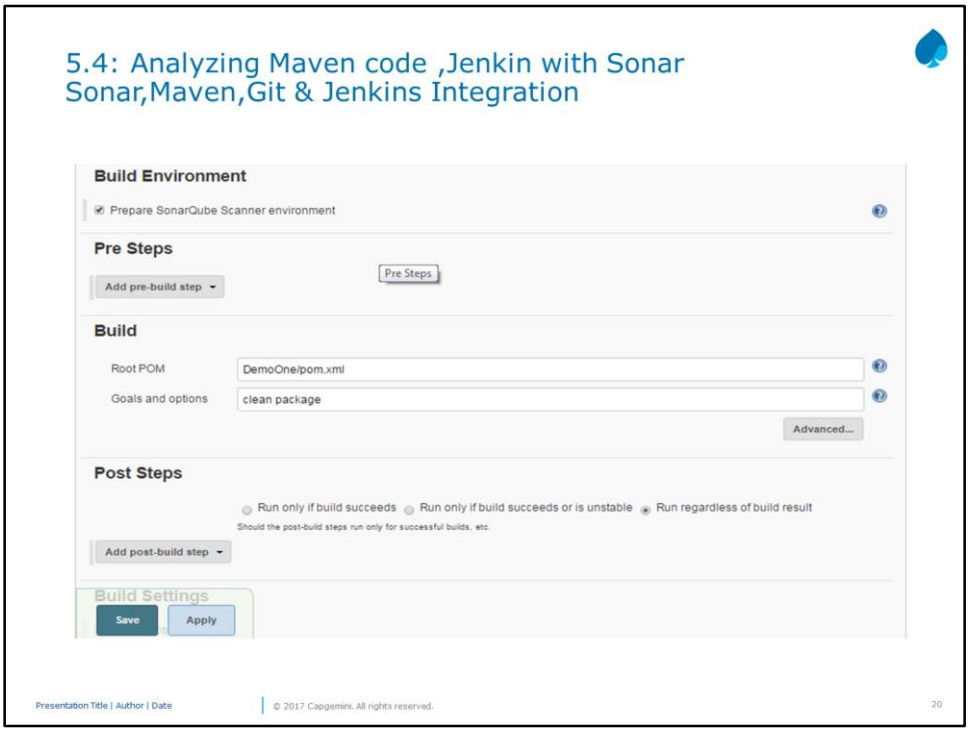
Click on the top "Rules" menu item to enter the world of rules. By default, you will see all the available rules, with the ability to narrow the selection based on search criteria in the left pane:
Language: the language to which a rule applies.
Type: Bug, Vulnerability or Code Smell rules
Tag: it is possible to add tags to rules in order to classify them and to help discover them more easily.
Repository: the engine that contributes rules to SonarQube.
Default Severity: the original severity of the rule - as defined by the plugin that contributes this rule.
Status: rules can have 3 different statuses:

Beta: The rule has been recently implemented and we haven't gotten enough feedback from users yet, so there may be false positives or false negatives.
Deprecated: The rule should no longer be used because a similar, but more powerful and accurate rule exists.
Ready: The rule is ready to be used in production.

Available Since: date when a rule was first added on the SonarQube instance. This is useful to list all the new rules since the last upgrade of a plugin for instance.
Template: display rule templates that allow to create custom rules (see later on this page).
Quality Profile: inclusion in or exclusion from a specific profile
To see the details of a rule, either click on it, or use the right arrow key. Along with basic rule data, you'll also be able to see which, if any, profiles it's active in and how many open issues have been raised with it.
The 2 following actions are available only if you have the right permissions ("Administer Quality Profiles and Gates"):
Add/Remove Tags:
It is possible to add existing tags on a rule, or to create new ones (just enter a new name while typing in the text field).
Note that some rules have built-in tags that you cannot remove - they are provided by the plugins which contribute the rules.
Extend Description:
Extending rule descriptions is useful to let users know how your organization is using a particular rule for instance or to give more insight on a rule.
Note that the extension will be available to non-admin users as a normal part of the rule details.

Instructor Notes:

Give Demo example



Click on the top "Rules" menu item to enter the world of rules. By default, you will see all the available rules, with the ability to narrow the selection based on search criteria in the left pane:
Language: the language to which a rule applies.
Type: Bug, Vulnerability or Code Smell rules
Tag: it is possible to add tags to rules in order to classify them and to help discover them more easily.
Repository: the engine that contributes rules to SonarQube.
Default Severity: the original severity of the rule - as defined by the plugin that contributes this rule.
Status: rules can have 3 different statuses:

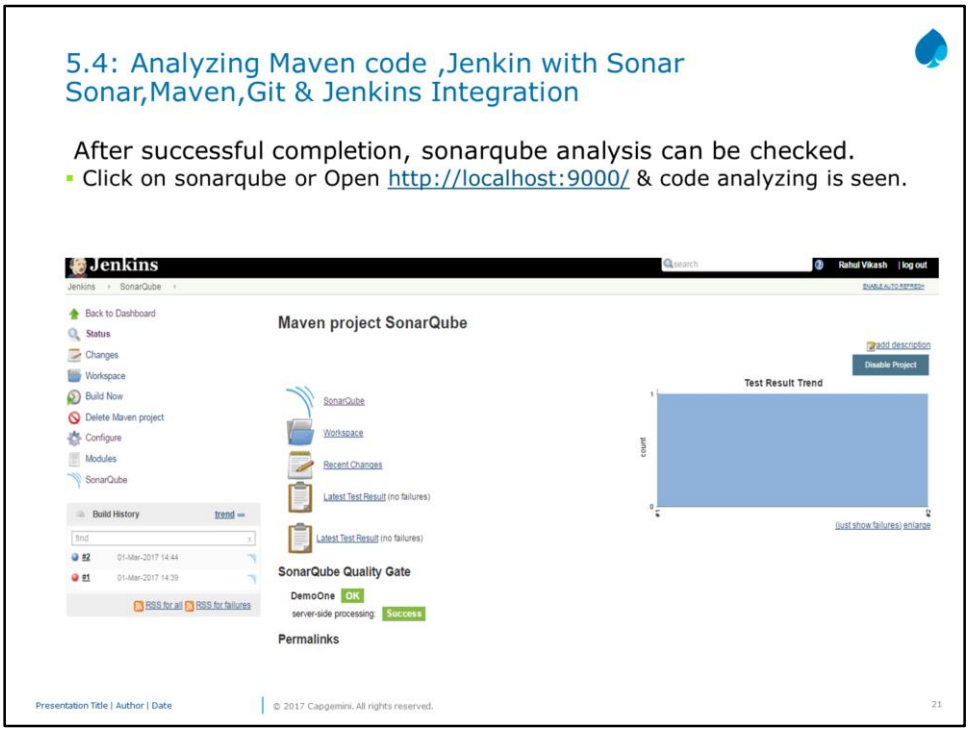
Beta: The rule has been recently implemented and we haven't gotten enough feedback from users yet, so there may be false positives or false negatives.
Deprecated: The rule should no longer be used because a similar, but more powerful and accurate rule exists.
Ready: The rule is ready to be used in production.

Available Since: date when a rule was first added on the SonarQube instance. This is useful to list all the new rules since the last upgrade of a plugin for instance.
Template: display rule templates that allow to create custom rules (see later on this page).
Quality Profile: inclusion in or exclusion from a specific profile
To see the details of a rule, either click on it, or use the right arrow key. Along with basic rule data, you'll also be able to see which, if any, profiles it's active in and how many open issues have been raised with it.
The 2 following actions are available only if you have the right permissions ("Administer Quality Profiles and Gates"):
Add/Remove Tags:
It is possible to add existing tags on a rule, or to create new ones (just enter a new name while typing in the text field).
Note that some rules have built-in tags that you cannot remove - they are provided by the plugins which contribute the rules.
Extend Description:
Extending rule descriptions is useful to let users know how your organization is using a particular rule for instance or to give more insight on a rule.
Note that the extension will be available to non-admin users as a normal part of the rule details.

Instructor Notes:

Give Demo example

.



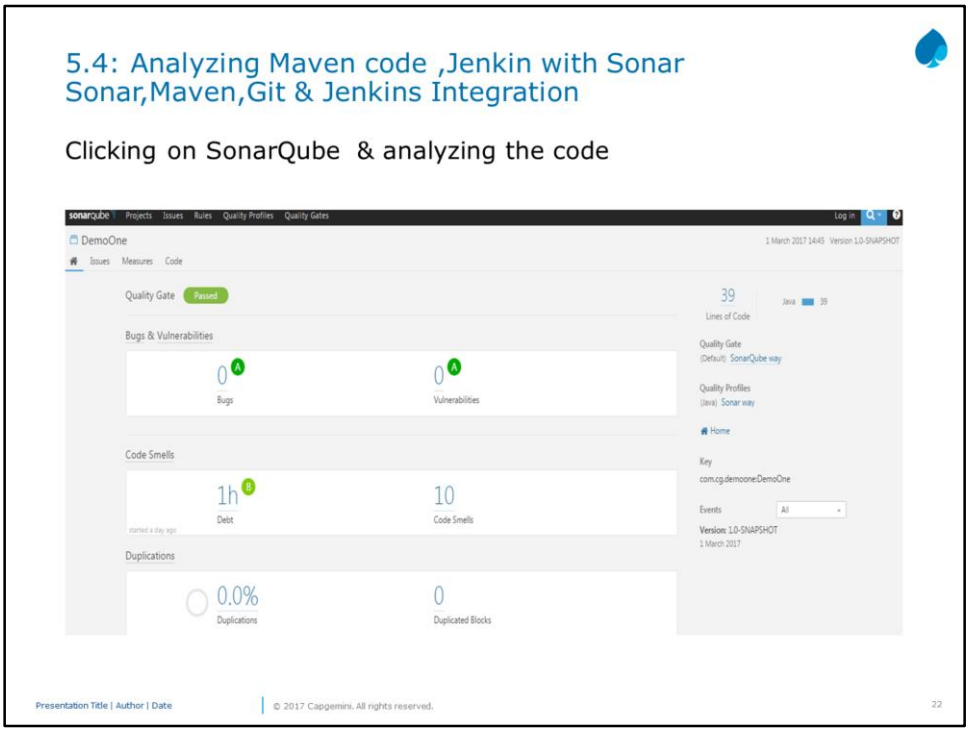
Click on the top "Rules" menu item to enter the world of rules. By default, you will see all the available rules, with the ability to narrow the selection based on search criteria in the left pane:
Language: the language to which a rule applies.
Type: Bug, Vulnerability or Code Smell rules
Tag: it is possible to add tags to rules in order to classify them and to help discover them more easily.
Repository: the engine that contributes rules to SonarQube.
Default Severity: the original severity of the rule - as defined by the plugin that contributes this rule.
Status: rules can have 3 different statuses:

Beta: The rule has been recently implemented and we haven't gotten enough feedback from users yet, so there may be false positives or false negatives.
Deprecated: The rule should no longer be used because a similar, but more powerful and accurate rule exists.
Ready: The rule is ready to be used in production.

Available Since: date when a rule was first added on the SonarQube instance. This is useful to list all the new rules since the last upgrade of a plugin for instance.
Template: display rule templates that allow to create custom rules (see later on this page).
Quality Profile: inclusion in or exclusion from a specific profile
To see the details of a rule, either click on it, or use the right arrow key. Along with basic rule data, you'll also be able to see which, if any, profiles it's active in and how many open issues have been raised with it.
The 2 following actions are available only if you have the right permissions ("Administer Quality Profiles and Gates"):
Add/Remove Tags:
It is possible to add existing tags on a rule, or to create new ones (just enter a new name while typing in the text field).
Note that some rules have built-in tags that you cannot remove - they are provided by the plugins which contribute the rules.
Extend Description:
Extending rule descriptions is useful to let users know how your organization is using a particular rule for instance or to give more insight on a rule.
Note that the extension will be available to non-admin users as a normal part of the rule details.

Instructor Notes:

Give Demo example



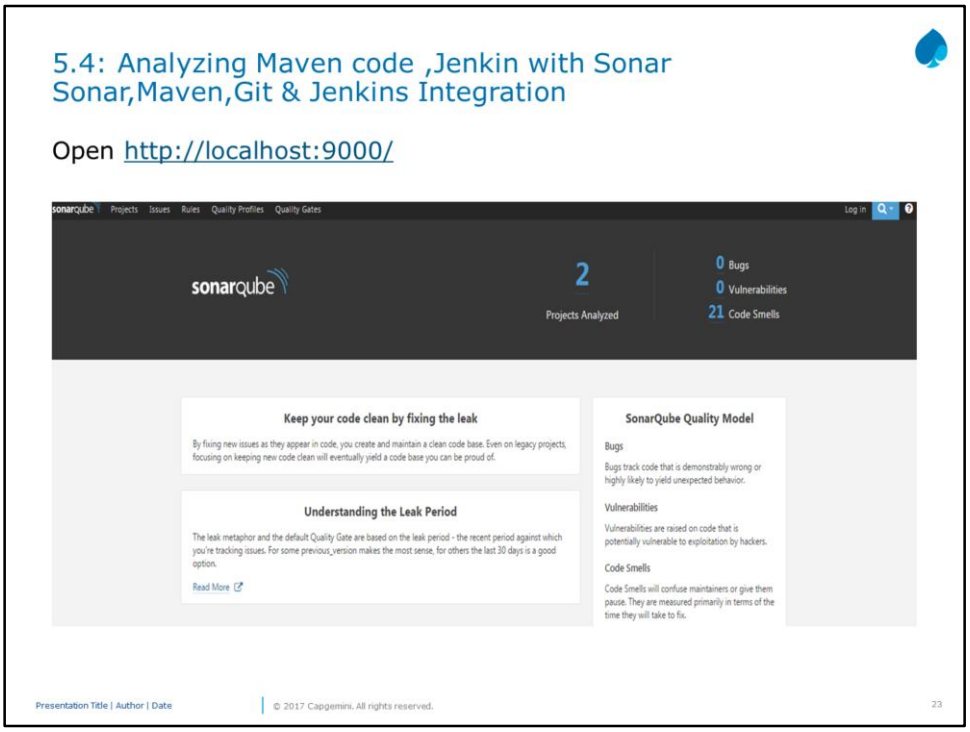
Click on the top "Rules" menu item to enter the world of rules. By default, you will see all the available rules, with the ability to narrow the selection based on search criteria in the left pane:
Language: the language to which a rule applies.
Type: Bug, Vulnerability or Code Smell rules
Tag: it is possible to add tags to rules in order to classify them and to help discover them more easily.
Repository: the engine that contributes rules to SonarQube.
Default Severity: the original severity of the rule - as defined by the plugin that contributes this rule.
Status: rules can have 3 different statuses:

Beta: The rule has been recently implemented and we haven't gotten enough feedback from users yet, so there may be false positives or false negatives.
Deprecated: The rule should no longer be used because a similar, but more powerful and accurate rule exists.
Ready: The rule is ready to be used in production.

Available Since: date when a rule was first added on the SonarQube instance. This is useful to list all the new rules since the last upgrade of a plugin for instance.
Template: display rule templates that allow to create custom rules (see later on this page).
Quality Profile: inclusion in or exclusion from a specific profile
To see the details of a rule, either click on it, or use the right arrow key. Along with basic rule data, you'll also be able to see which, if any, profiles it's active in and how many open issues have been raised with it.
The 2 following actions are available only if you have the right permissions ("Administer Quality Profiles and Gates"):
Add/Remove Tags:
It is possible to add existing tags on a rule, or to create new ones (just enter a new name while typing in the text field).
Note that some rules have built-in tags that you cannot remove - they are provided by the plugins which contribute the rules.
Extend Description:
Extending rule descriptions is useful to let users know how your organization is using a particular rule for instance or to give more insight on a rule.
Note that the extension will be available to non-admin users as a normal part of the rule details.

Instructor Notes:

Give Demo example



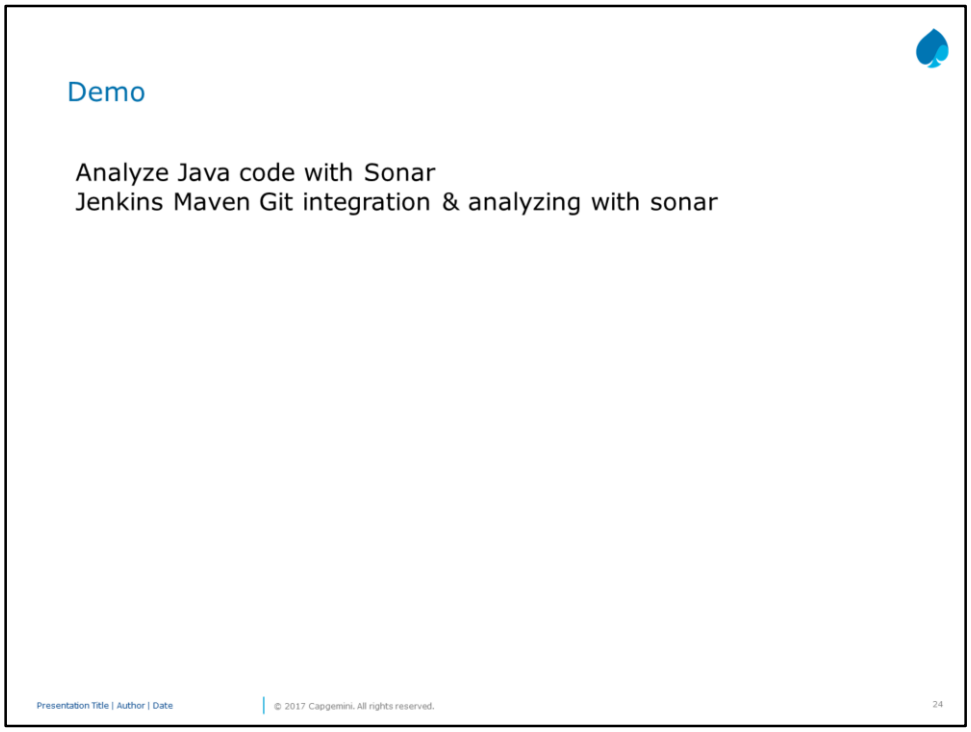
Click on the top "Rules" menu item to enter the world of rules. By default, you will see all the available rules, with the ability to narrow the selection based on search criteria in the left pane:
Language: the language to which a rule applies.
Type: Bug, Vulnerability or Code Smell rules
Tag: it is possible to add tags to rules in order to classify them and to help discover them more easily.
Repository: the engine that contributes rules to SonarQube.
Default Severity: the original severity of the rule - as defined by the plugin that contributes this rule.
Status: rules can have 3 different statuses:

Beta: The rule has been recently implemented and we haven't gotten enough feedback from users yet, so there may be false positives or false negatives.
Deprecated: The rule should no longer be used because a similar, but more powerful and accurate rule exists.
Ready: The rule is ready to be used in production.

Available Since: date when a rule was first added on the SonarQube instance. This is useful to list all the new rules since the last upgrade of a plugin for instance.
Template: display rule templates that allow to create custom rules (see later on this page).
Quality Profile: inclusion in or exclusion from a specific profile
To see the details of a rule, either click on it, or use the right arrow key. Along with basic rule data, you'll also be able to see which, if any, profiles it's active in and how many open issues have been raised with it.
The 2 following actions are available only if you have the right permissions ("Administer Quality Profiles and Gates"):
Add/Remove Tags:
It is possible to add existing tags on a rule, or to create new ones (just enter a new name while typing in the text field).
Note that some rules have built-in tags that you cannot remove - they are provided by the plugins which contribute the rules.
Extend Description:
Extending rule descriptions is useful to let users know how your organization is using a particular rule for instance or to give more insight on a rule.
Note that the extension will be available to non-admin users as a normal part of the rule details.

Instructor Notes:

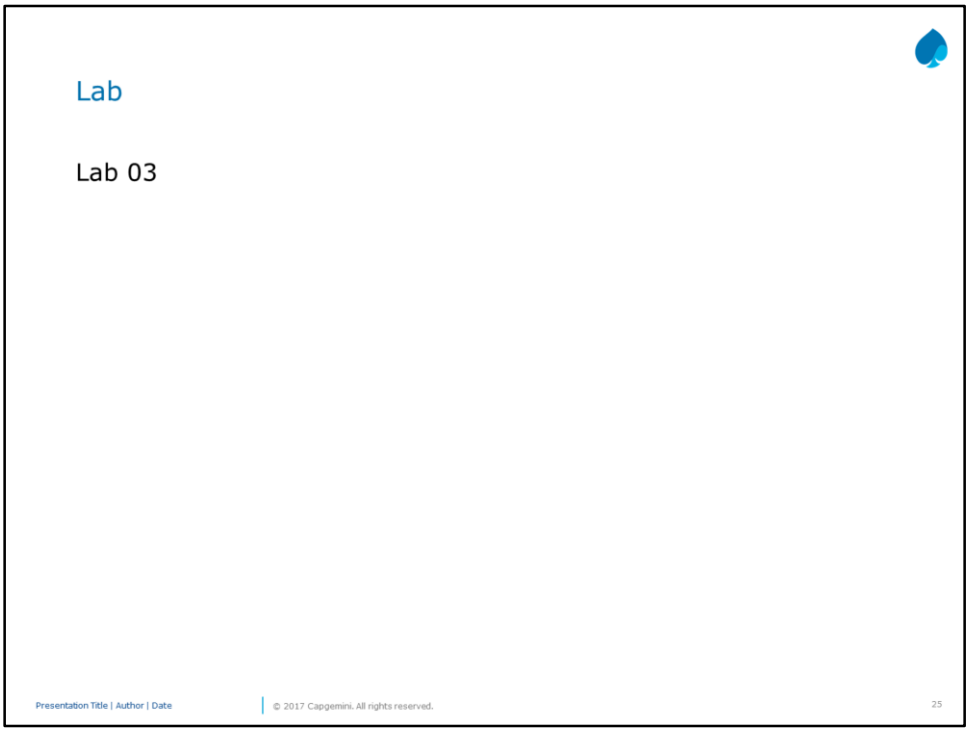
Add instructor notes here.



Add the notes here.

Instructor Notes:

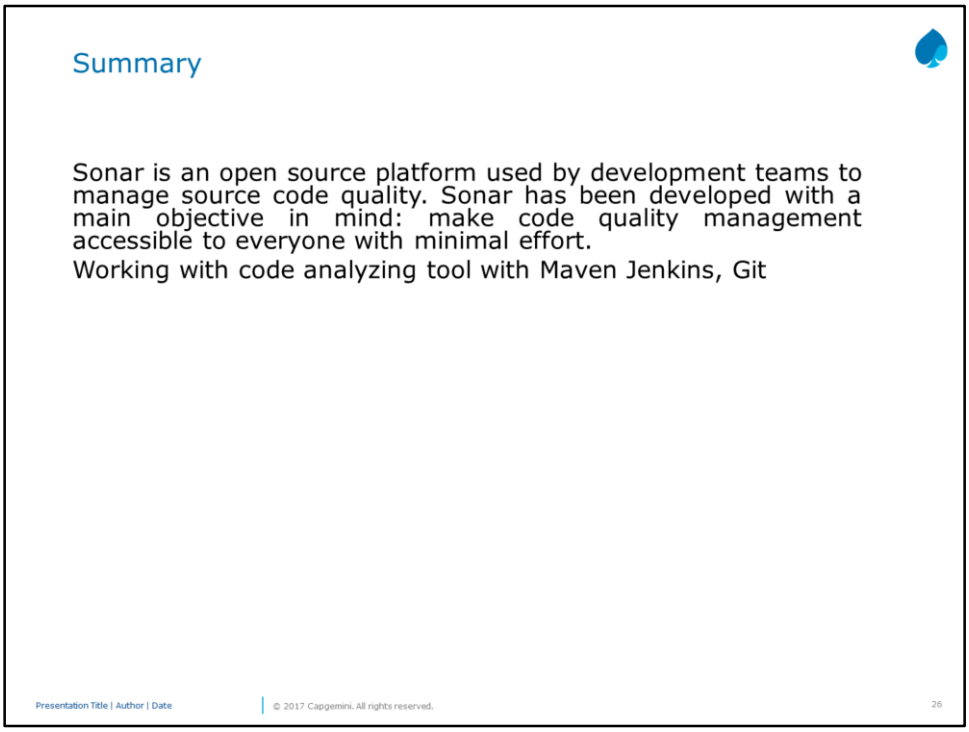
Add instructor notes here.



Add the notes here.

Instructor Notes:

Add instructor notes here.



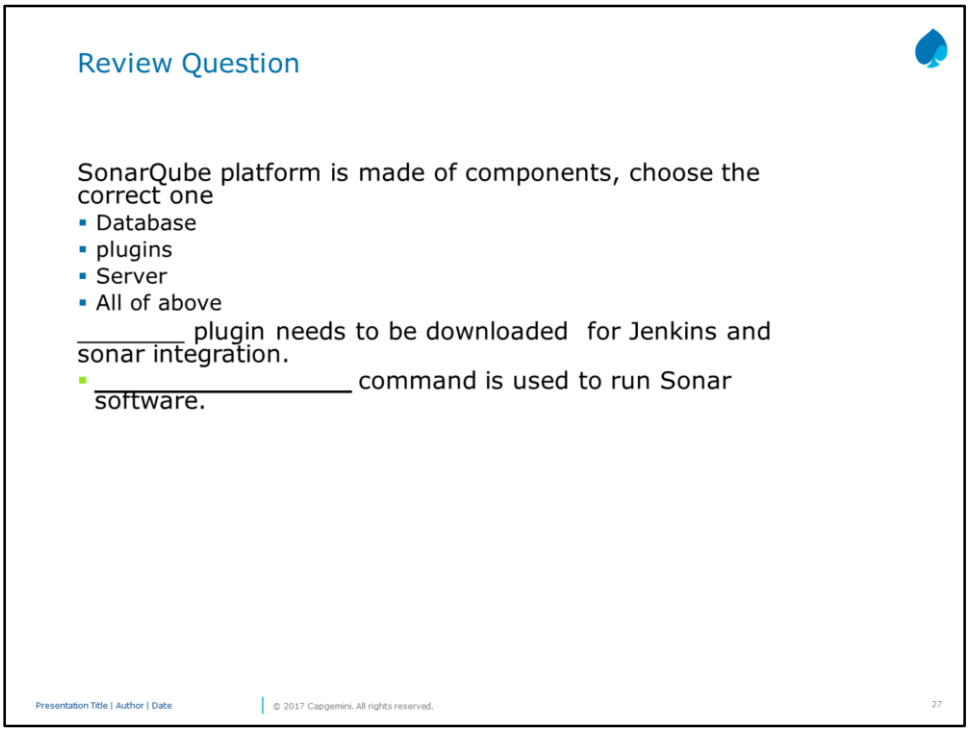
Add the notes here.

Instructor Notes:

Q1.All of above.

Q2. Sonarqube

Q3StartSonar.bat



Add the notes here.