Assignment No 02

Aim:

To implement static application security testing using SonarQube.

Tools used

1. SonarQube (Community).
2. SonarScanner.
3. SonarScanner for Maven
4. Jenkins
5. Code repository.

Theory:

Static Application Security Testing (SAST) inspects source code, bytecode, or configuration files without executing the program to identify vulnerabilities (e.g., SQL injection, XSS, hard-coded secrets), code smells, bugs, and maintainability issues early in the SDLC. SAST finds issues at the code-level and provides traceable rule-based findings.

SonarQube is a code quality and SAST platform that:

- Parses code using language analyzers and applies rule engines to detect security hotspots, bugs, vulnerabilities, and code smells.
- Aggregates metrics (coverage, duplications, complexity) and enforces **Quality Gates** (pass/fail criteria).
- Integrates with CI to scan every commit / PR and gives actionable issue lists and remediation guidance.
- Supports extensible rule sets (OWASP, CWE mappings) and security-focused profiles (e.g., SAST-focused Quality Profiles).

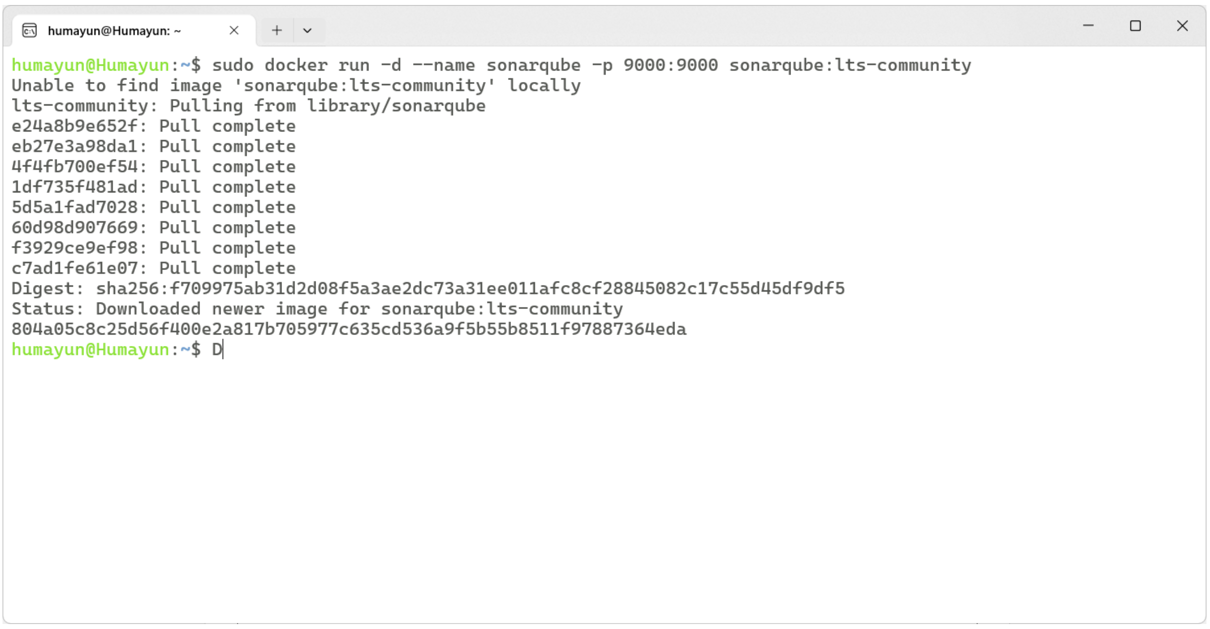
Why use SonarQube in SAST pipeline?

- Centralizes findings across languages/projects.
- Enforces automated quality/security gates to stop bad code merging.
- Provides developer-friendly guidance and prioritization (severity, reliability, remediation effort).
- Integrates with IDEs so developers can fix issues before pushes.

Steps

Step 1: Open Terminal

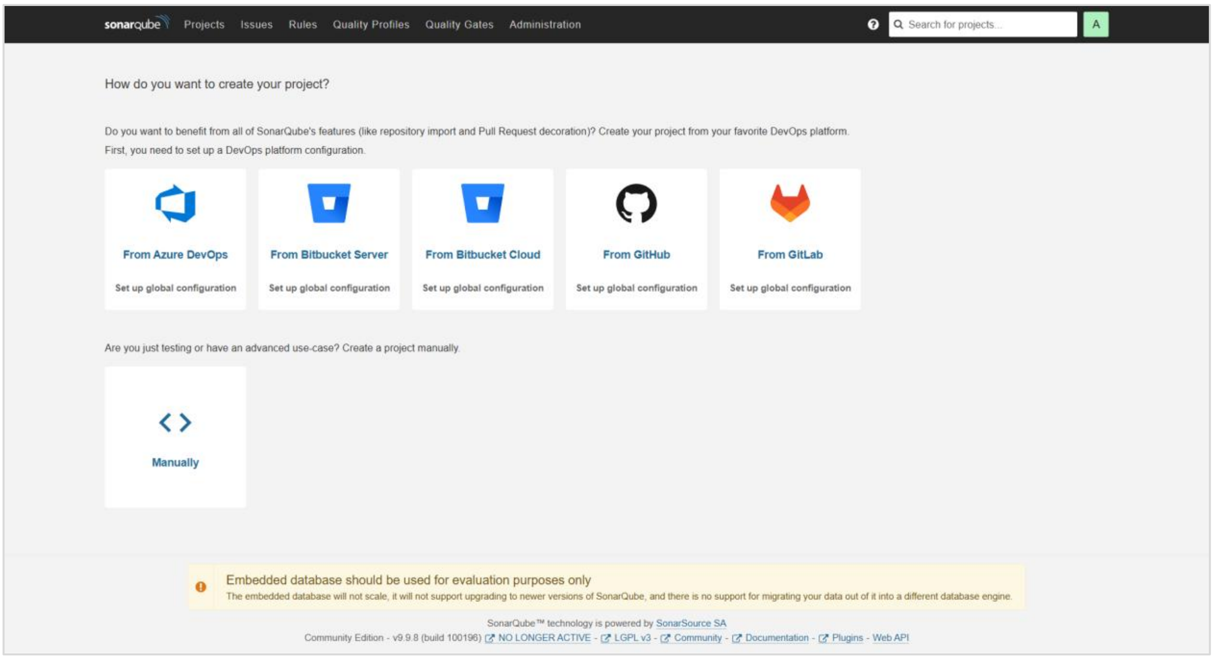
- Launch the terminal on your system (Ubuntu or WSL on Windows).
- Make sure Docker is installed and running properly.



```
humayun@Humayun:~$ sudo docker run -d --name sonarqube -p 9000:9000 sonarqube:lts-community
Unable to find image 'sonarqube:lts-community' locally
lts-community: Pulling from library/sonarqube
e24a8b9e652f: Pull complete
eb27e3a98da1: Pull complete
4f4fb700ef54: Pull complete
1df735f481ad: Pull complete
5d5a1fad7028: Pull complete
60d98d907669: Pull complete
f3929ce9ef98: Pull complete
c7ad1fe61e07: Pull complete
Digest: sha256:f709975ab31d2d08f5a3ae2dc73a31ee011afc8cf28845082c17c55d45df9df5
Status: Downloaded newer image for sonarqube:lts-community
804a05c8c25d56f400e2a817b705977c635cd536a9f5b55b8511f97887364eda
humayun@Humayun:~$ D
```

Step 2: Pull and Run SonarQube Container

- Execute the following Docker command to download and start the **SonarQube LTS Community Edition** container:



How do you want to create your project?

Do you want to benefit from all of SonarQube's features (like repository import and Pull Request decoration)? Create your project from your favorite DevOps platform.
First, you need to set up a DevOps platform configuration.

From Azure DevOps
Set up global configuration
From Bitbucket Server
Set up global configuration
From Bitbucket Cloud
Set up global configuration
From GitHub
Set up global configuration
From GitLab
Set up global configuration

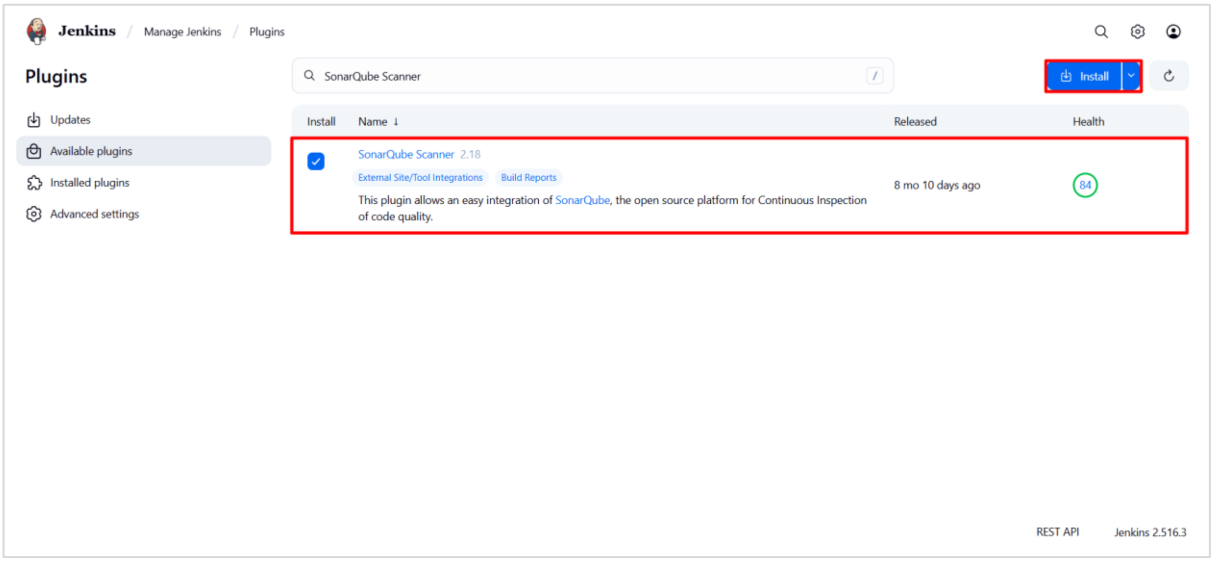
Are you just testing or have an advanced use-case? Create a project manually.

Embedded database should be used for evaluation purposes only
The embedded database will not scale, it will not support upgrading to newer versions of SonarQube, and there is no support for migrating your data out of it into a different database engine.

SonarQube™ technology is powered by SonarSource SA
Community Edition - v9.9.8 (build 100196) NO LONGER ACTIVE - LGPL v3 - Community - Documentation - Plugins - Web API

Step 3: Wait for Image Download and Setup

- Docker will automatically:
  - Pull the necessary image layers (as shown in your screenshot: "Pull complete" messages).
  - Assemble and start the SonarQube container.
- Once completed, you'll see an output like:



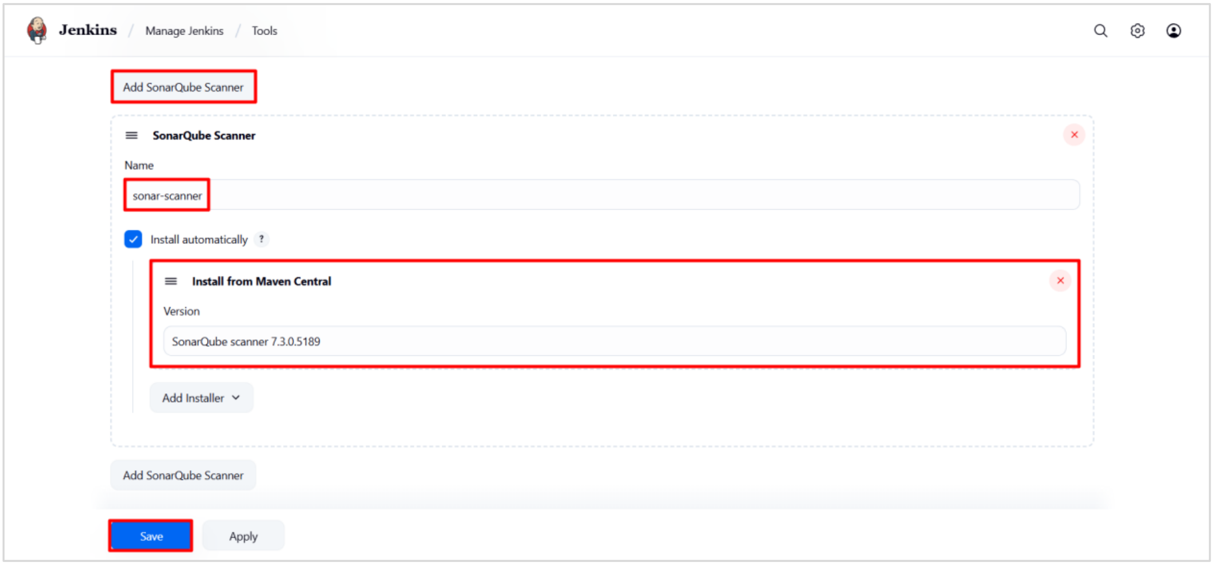
SonarQube Scanner

Install	Name ↓	Released	Health
✓	SonarQube Scanner 2.18 External Site/Tool Integrations Build Reports This plugin allows an easy integration of SonarQube, the open source platform for Continuous Inspection of code quality.	8 mo 10 days ago	84

REST API Jenkins 2.516.3

Step 4: Add SonarQube Scanner:

- Add SonarQube Scanner in the tools
- Add Maven3 in the tools



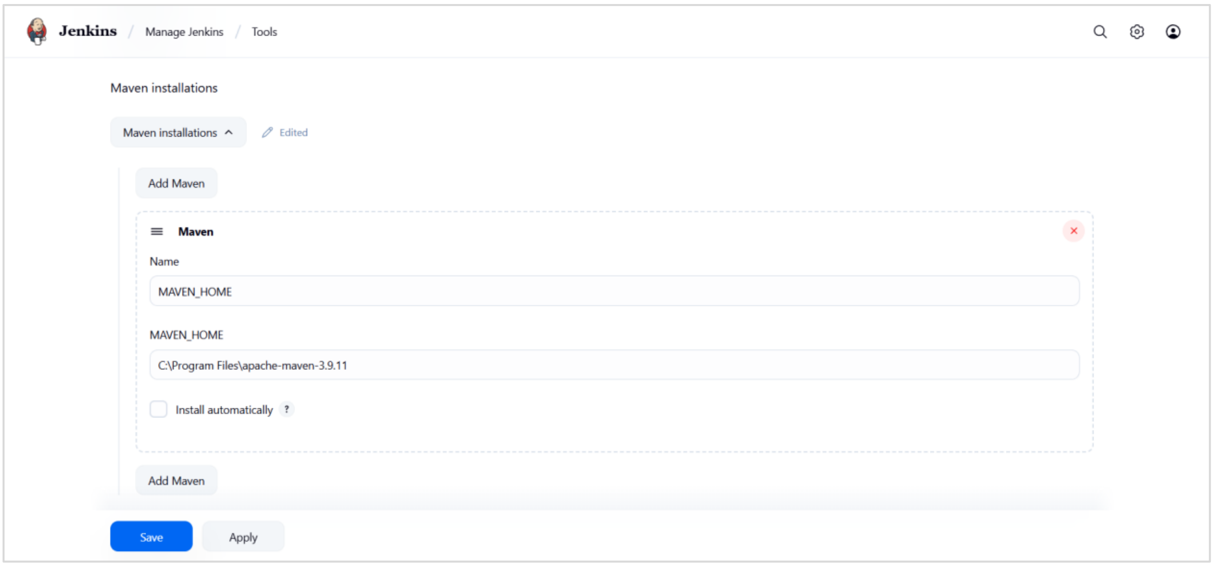
SonarQube Scanner

Name: sonar-scanner

✓ Install automatically

Install from Maven Central

Version: SonarQube scanner 7.3.0.5189



Maven installations

Maven installations — Edited

Maven

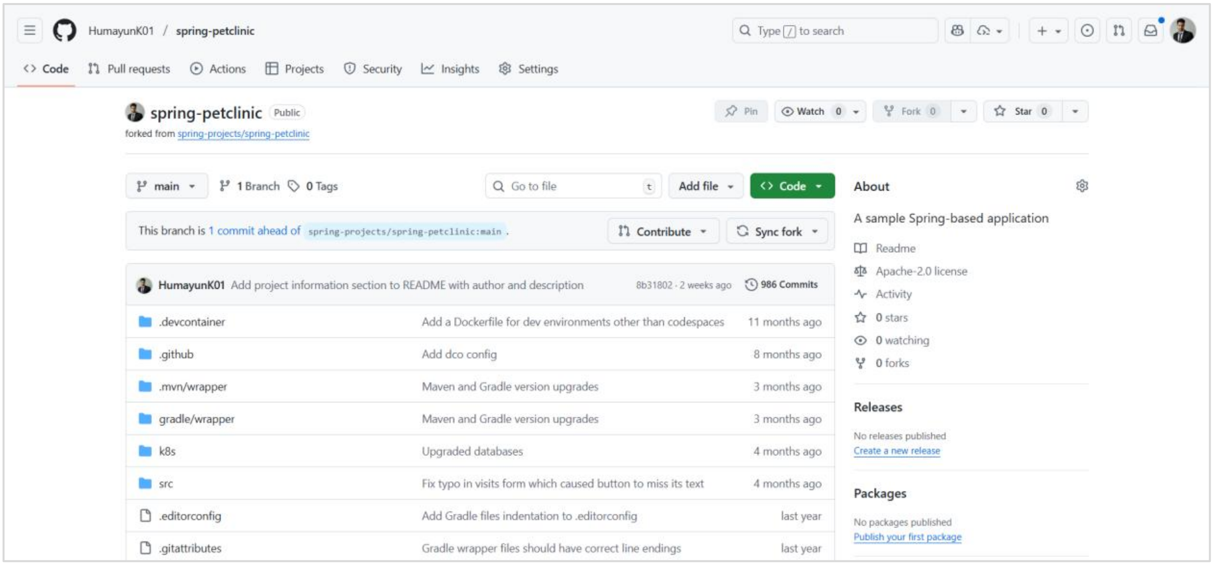
Name: MAVEN_HOME

MAVEN_HOME: C:\Program Files\apache-maven-3.9.11

Install automatically

Step 6: Create a New Project

1. Click **“Create new project”** in the SonarQube dashboard.
2. Enter a **Project Key** and **Project Name**.
3. Generate a **token** (this will be used by SonarScanner).



spring-petclinic Public
forked from spring-projects/spring-petclinic

This branch is 1 commit ahead of spring-projects/spring-petclinic:main .

HumayunK01 Add project information section to README with author and description 8b31802 · 2 weeks ago 986 Commits

.devcontainer	Add a Dockerfile for dev environments other than codespaces	11 months ago
.github	Add dco config	8 months ago
.mvn/wrapper	Maven and Gradle version upgrades	3 months ago
gradle/wrapper	Maven and Gradle version upgrades	3 months ago
k8s	Upgraded databases	4 months ago
src	Fix typo in visits form which caused button to miss its text	4 months ago
.editorconfig	Add Gradle files indentation to .editorconfig	last year
.gitattributes	Gradle wrapper files should have correct line endings	last year

About

A sample Spring-based application

Readme
Apache-2.0 license
Activity
0 stars
0 watching
0 forks

Releases

No releases published
Create a new release

Packages

No packages published
Publish your first package

Step 8: Run Static Code Analysis

Run the scanner from your project directory:

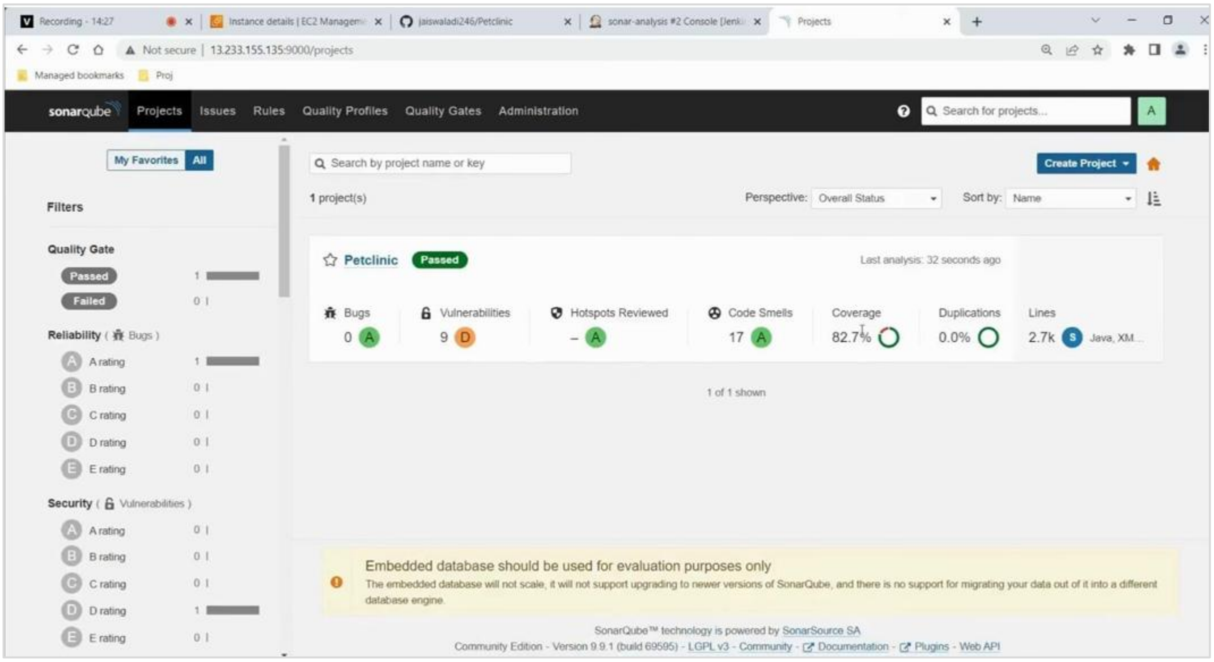
Once completed, return to the SonarQube dashboard to view detailed results, including:

- **Vulnerabilities**
- **Bugs**
- **Code Smells**

- **Security Hotspots**

Step 9: Review and Interpret Results

- Go to your project in the dashboard.
- Review findings and **fix vulnerabilities or issues** based on the detailed report.



1 project(s)

Quality Gate: Passed 1; Failed 0

Reliability (Bugs): A rating 1; B rating 0; C rating 0; D rating 0; E rating 0

Security (Vulnerabilities): A rating 0; B rating 0; C rating 0; D rating 1; E rating 0

Petclinic Passed — Last analysis: 32 seconds ago

Bugs	Vulnerabilities	Hotspots Reviewed	Code Smells	Coverage	Duplications	Lines
0 A	9 D	– A	17 A	82.7%	0.0%	2.7k S Java, XM...

1 of 1 shown

Embedded database should be used for evaluation purposes only

The embedded database will not scale, it will not support upgrading to newer versions of SonarQube, and there is no support for migrating your data out of it into a different database engine.

SonarQube™ technology is powered by SonarSource SA

Community Edition - Version 9.9.1 (build 69595) - LGPL v3 - Community - Documentation - Plugins - Web API

Conclusion

Implementing SAST with SonarQube gives you automated, repeatable detection of security flaws and quality regressions early in the development lifecycle. The recommended flow is: run local scans → integrate Sonar in CI for PR and branch scans → enforce Quality Gates → provide developers with IDE feedback via SonarLint → triage and fix vulnerabilities first. This setup reduces risk, shortens remediation time, and raises overall code health.