EXPERIMENT-14

NAME: SHAIKH MUBASHIRA TUFEL AHMED
ROLL NO: 612055 **COURSE:** ADVANCE DEVOPS(ITL504)
BRANCH: T.E. INFORMATION TECHNOLOGY (SEM 5)

CASE-STUDY [SONARQUBE]

Q1. What is SonarQube? Why use SonarQube?

➔SonarQube is an open-source platform developed by SonarSource for continuous inspection of code quality. Sonar does static code analysis, which provides a detailed report of bugs, code smells, vulnerabilities, code duplications.

SonarQube (formerly Sonar) is an open-source platform developed by SonarSource for continuous inspection of code quality to perform automatic reviews with static analysis of code to detect bugs, code smells on 29 programming languages.

SonarQube is a Code Quality Assurance tool that collects and analyzes source code, and provides reports for the code quality of your project. It combines static and dynamic analysis tools and enables quality to be measured continually over time.

Q2. What is software quality measurement ?

➔In Software Engineering, Software Measurement is done based on some Software Metrics where these software metrics are referred to as the measure of various characteristics of a Software. In Software engineering Software Quality Assurance (SAQ) assures the quality of the software.

Q3. What is Static and Dynamic Code Analysis?

➔Dynamic analysis is the testing and evaluation of an application during runtime. Static analysis is the testing and evaluation of an application by examining the code without executing the application. Many software defects that cause memory and threading errors can be detected both dynamically and statically.

For example, static code analysis is a form of white-box testing that can help identify security issues in source code. On the other hand, dynamic code analysis is a form of black-box vulnerability scanning that allows software teams to scan running applications and identify vulnerabilities.

Q4. What are the benefits of using SonarQube?

❖**Sustainability -** Reduces complexity, possible vulnerabilities, and code duplications, optimising the life of applications.

❖**Increase productivity** - Reduces the scale, cost of maintenance, and risk of the application; as such, it removes the need to spend more time changing the code

❖**Quality code** - Code quality control is an inseparable part of the process of software development.

❖**Detect Errors** - Detects errors in the code and alerts developers to fix them automatically before submitting them for output.

❖**Increase consistency** - Determines where the code criteria are breached and enhances the quality.

❖**Business scaling** - No restriction on the number of projects to be evaluated.

❖**Enhance developer skills** - Regular feedback on quality problems helps developers to improve their coding skills.