Name: Kaushal Kotkar Department: Cyber Security

Div: BE-15 Roll No: 54

Subject: DSO

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Experiment No. – 9** | | | | |
| **Date of Performance:** | 09/09/2024 | | | |
| **Date of Submission:** | 16/09/2024 | | | |
| Program Execution/ formation/ correction/  ethical practices (06) | Timely Submission  (01) | Viva (03) | Experiment Total (10) | Sign with Date |
|  |  |  |  |  |

**Experiment No. 9**

**Aim:** To implement Static Application Security Testing using SonarQube.

**Lab Outcome:** Use Sonarqube and snyk to perform code quality checks and Threat Dragon to create threat models to identify threats in the system

**Theory:**

**Static Application Security Testing (SAST):** SAST is a security testing method that analyzes the source code, bytecode, or binary code of an application for security vulnerabilities and coding flaws. It identifies potential security issues early in the development process.

**SonarQube:** SonarQube is an open-source platform for continuous inspection of code quality, security, and maintainability. It offers static code analysis and security scanning capabilities.

**Why Use SonarQube:** SonarQube helps identify security vulnerabilities, code smells, and bugs in the source code, providing developers with actionable feedback to improve code quality and security. It supports a wide range of programming languages and integrates well into DevOps pipelines.

**Steps Followed:**

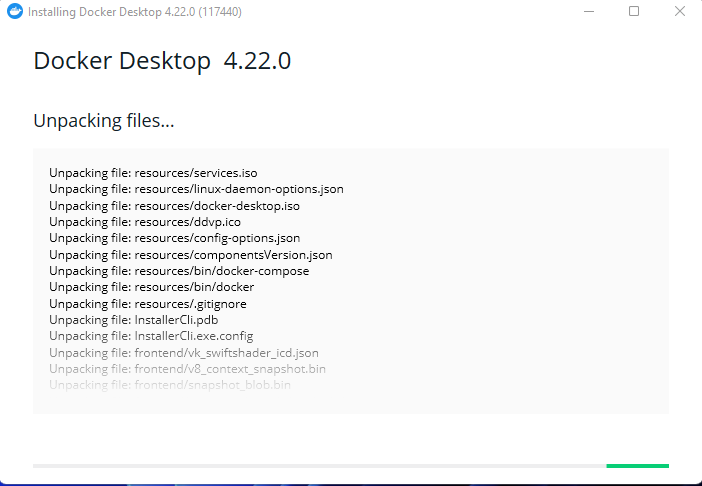
1. **Install Docker:**
   * Download and install Docker from the official website for your operating system.
2. **Pull SonarQube Image:**
   * Open a terminal or command prompt and use the command: ***docker pull sonarqube***to download the SonarQube Docker image from Docker Hub.
3. **Run SonarQube Container:**
   * Start a SonarQube container with the command:

*docker run -d --name sonarqube -p 9000:9000 -p 9092:9092 sonarqube*

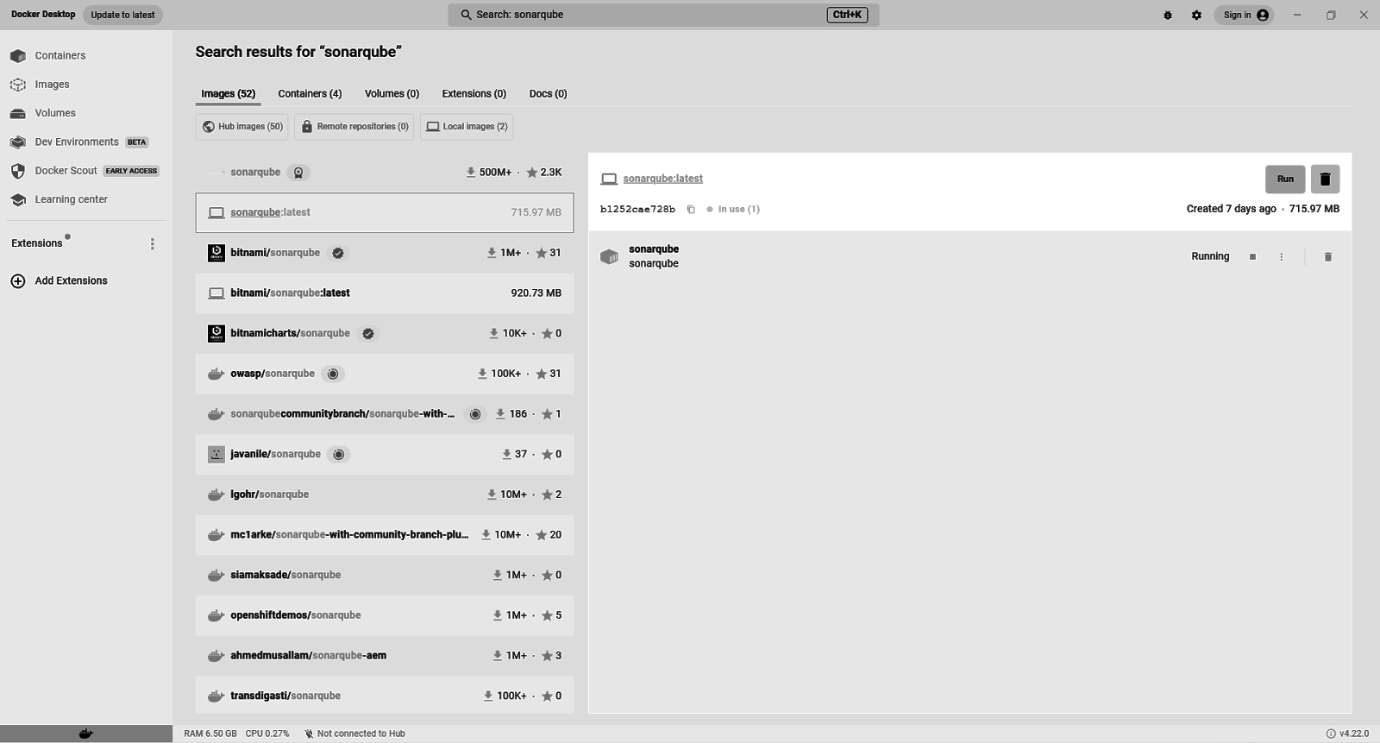
1. **Access SonarQube Web Interface:**
   * Open your web browser and go to [http://localhost:9000](http://localhost:9000/) to access the SonarQube web interface.
2. **Log in to SonarQube:**
   * Use the default credentials (admin/admin) to log in. You'll be prompted to change the password.
3. **Create a Project in SonarQube:**
   * Click on "Create New Project" and provide project details, such as project key and name. Click "Set Up" to create the project.
4. **Generate an Authentication Token (Optional):**
   * If needed, generate an authentication token for analysis. This is useful for private repositories or specific setups. You can generate a token in your SonarQube profile settings.
5. **Analyze Code with Specific Build Tools:**
   * Depending on your project's programming language, analyze the code using specific build tools. Ensure that the SonarQube scanner (sonar-scanner) is downloaded and added to your system's environment variables.
6. **View Analysis Results:**
   * After running the code analysis, review the results and insights provided by SonarQube in the web interface.

**Output:**

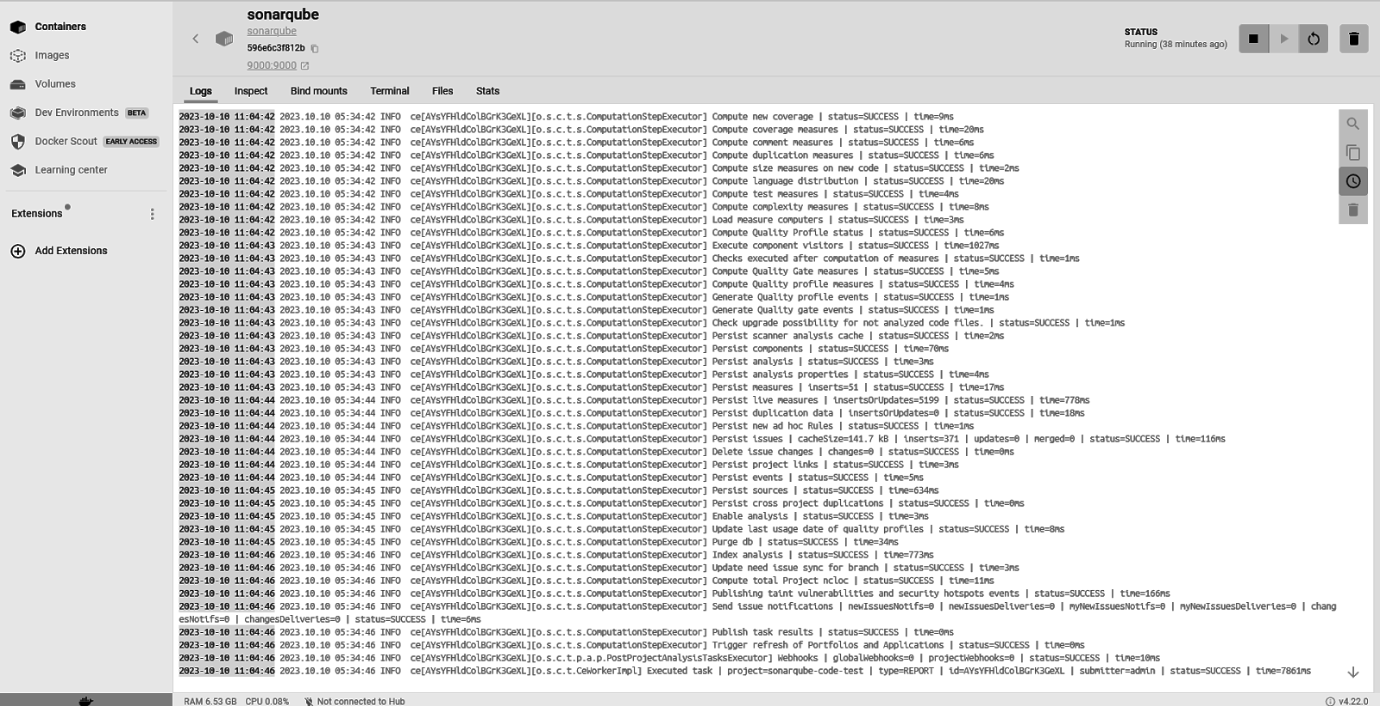
1. Install Docker



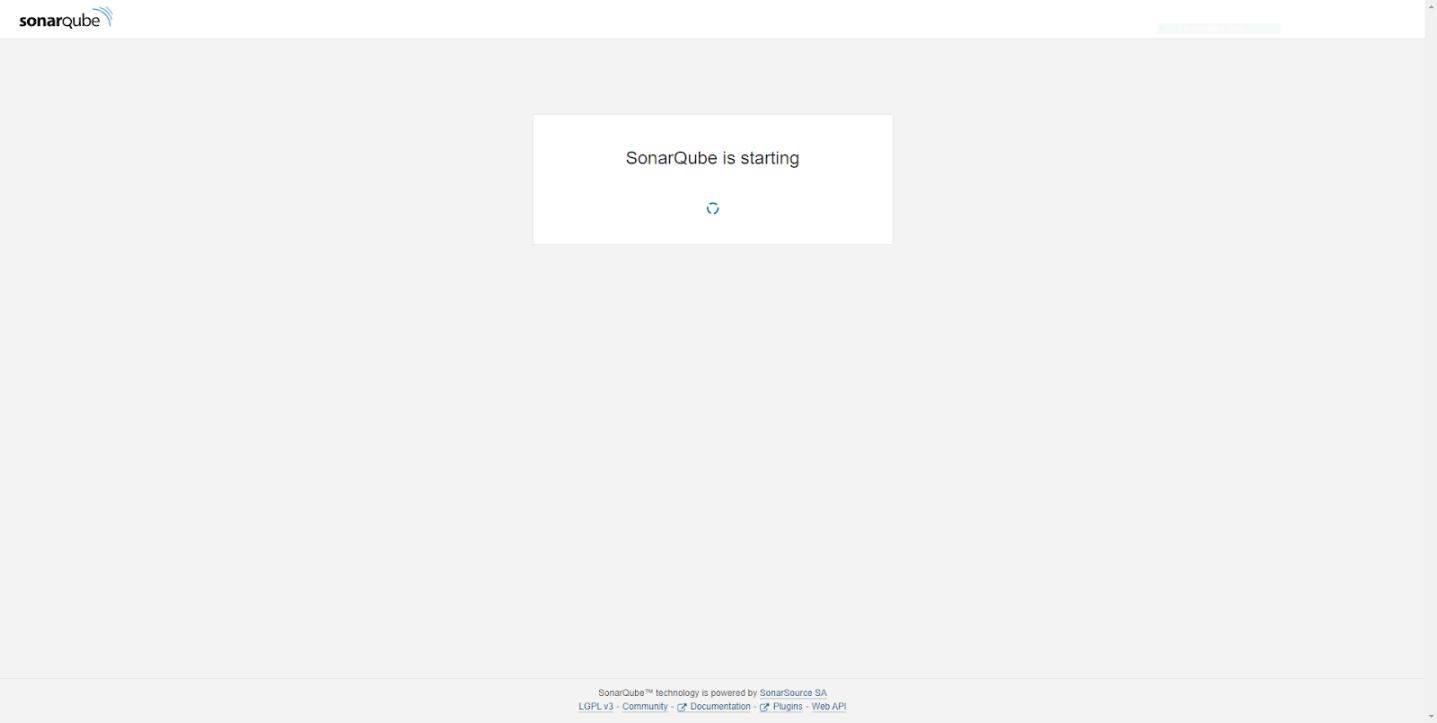
2. Pull SonarQube Image



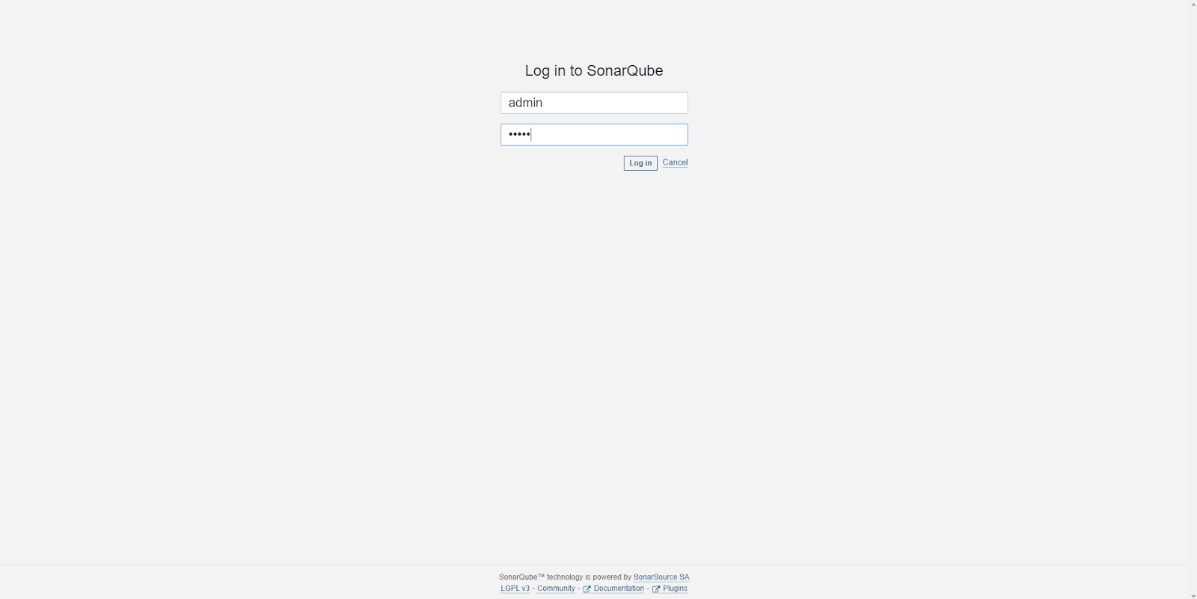
3. Run SonarQube Container



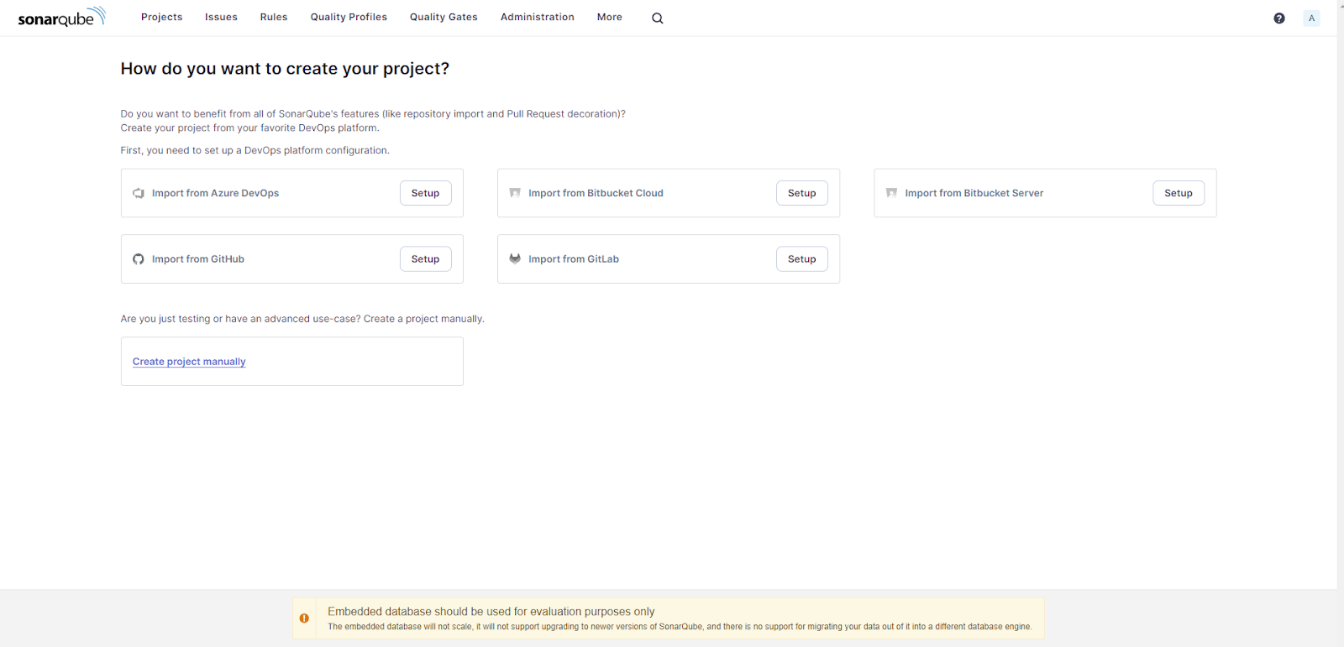
4. Access SonarQube Web Interface

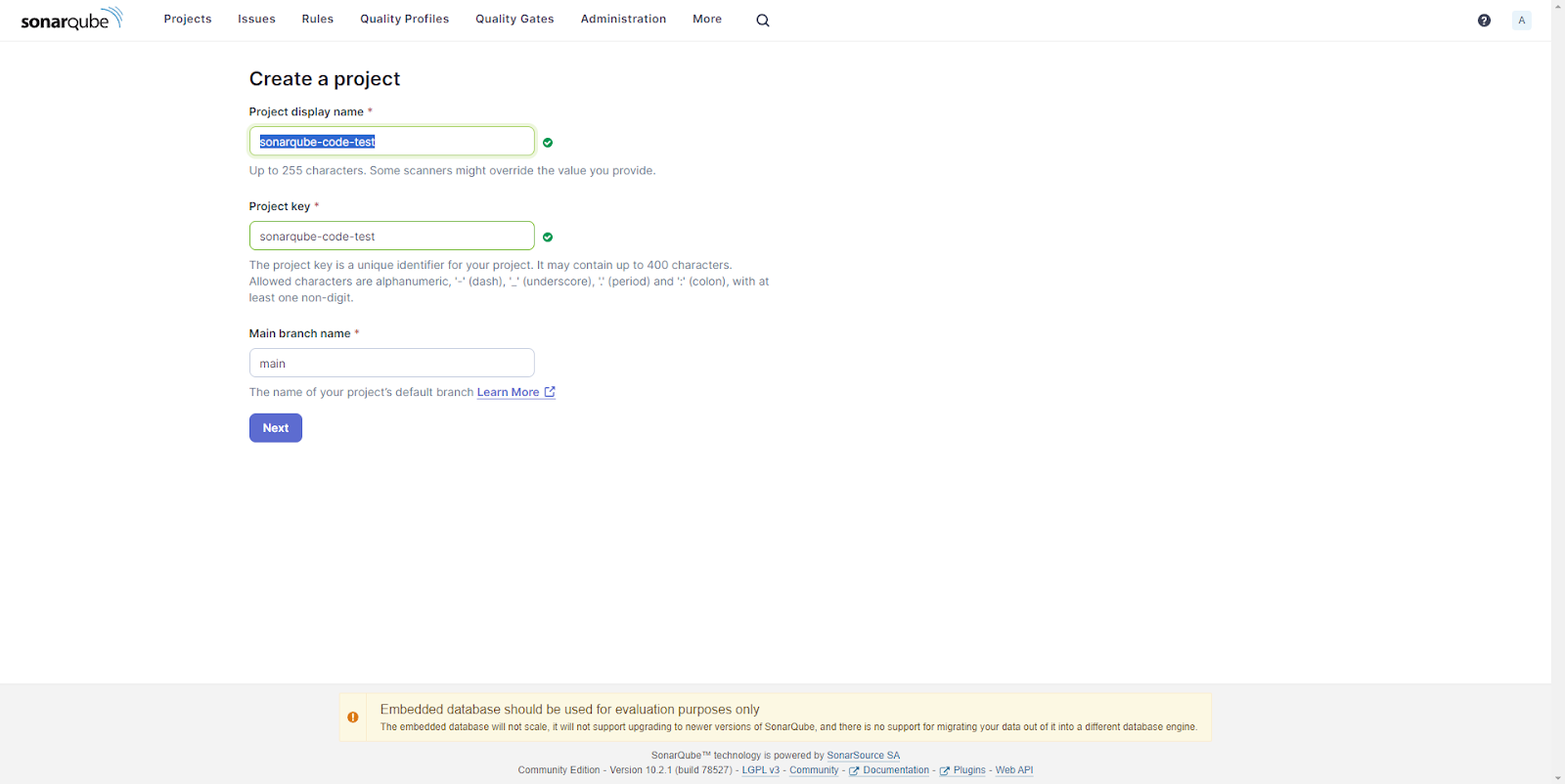


5. Log in to SonarQube

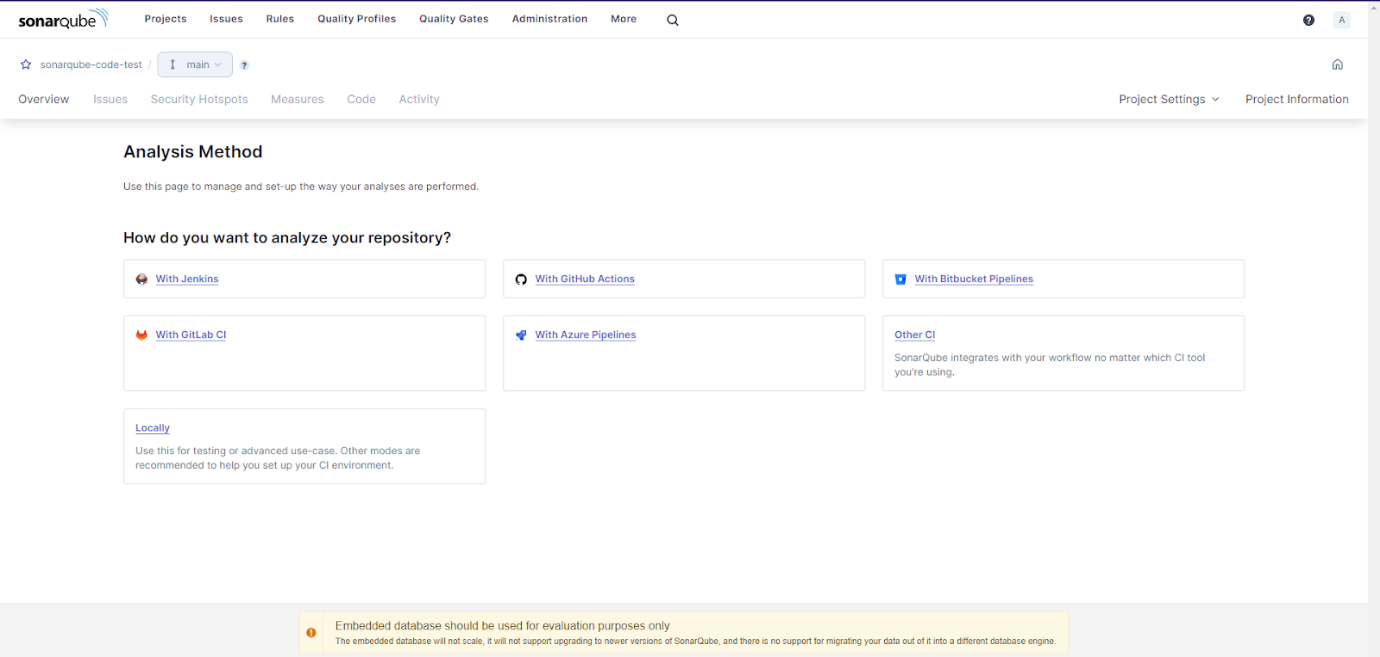


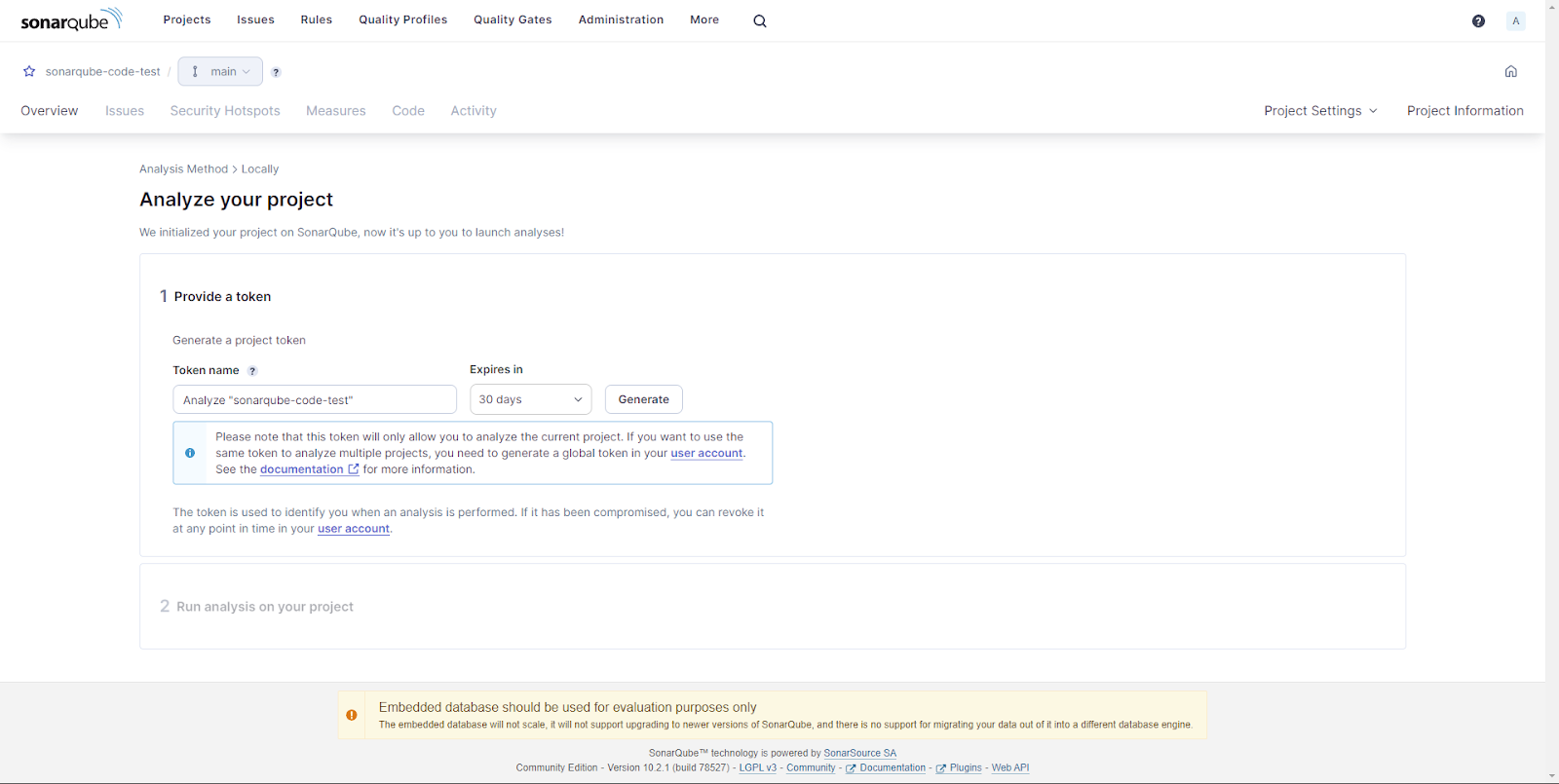
6. Create a Project in SonarQube



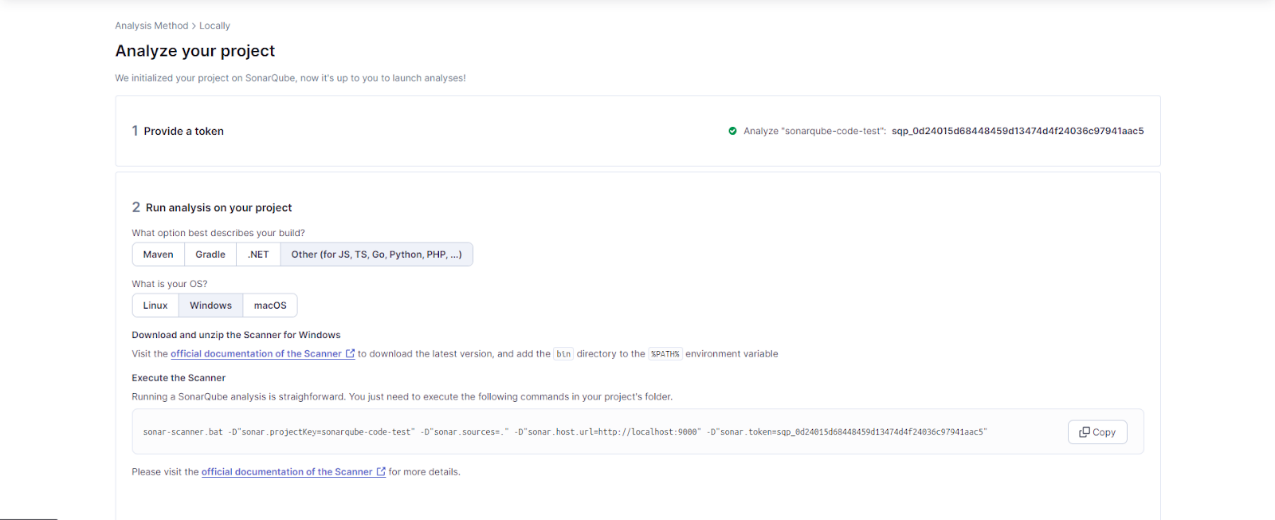


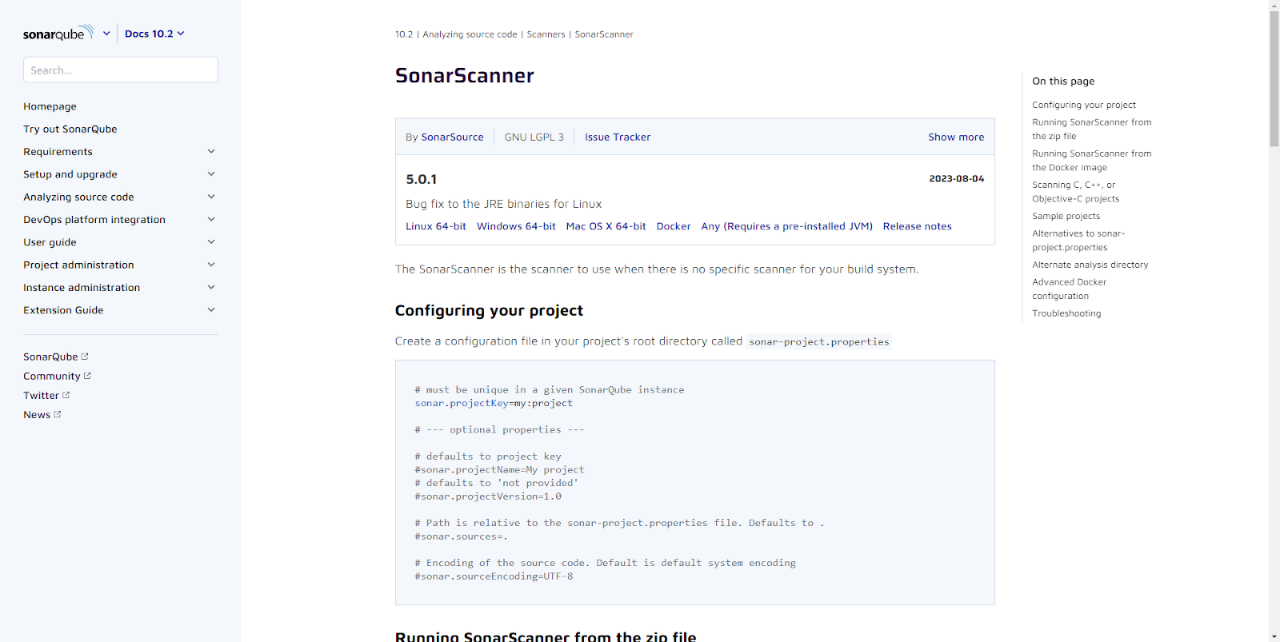
7. Generate an Authentication Token (Optional)





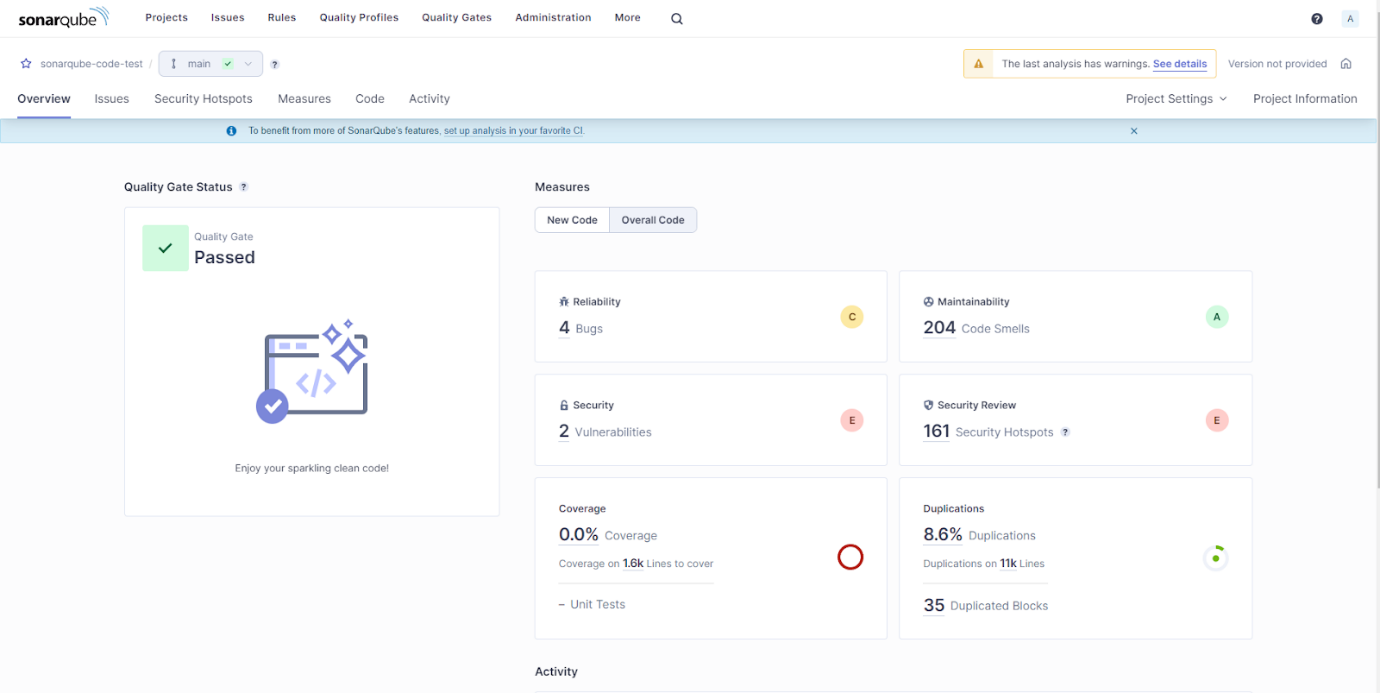
8. Analyze Code with Specific Build Tools

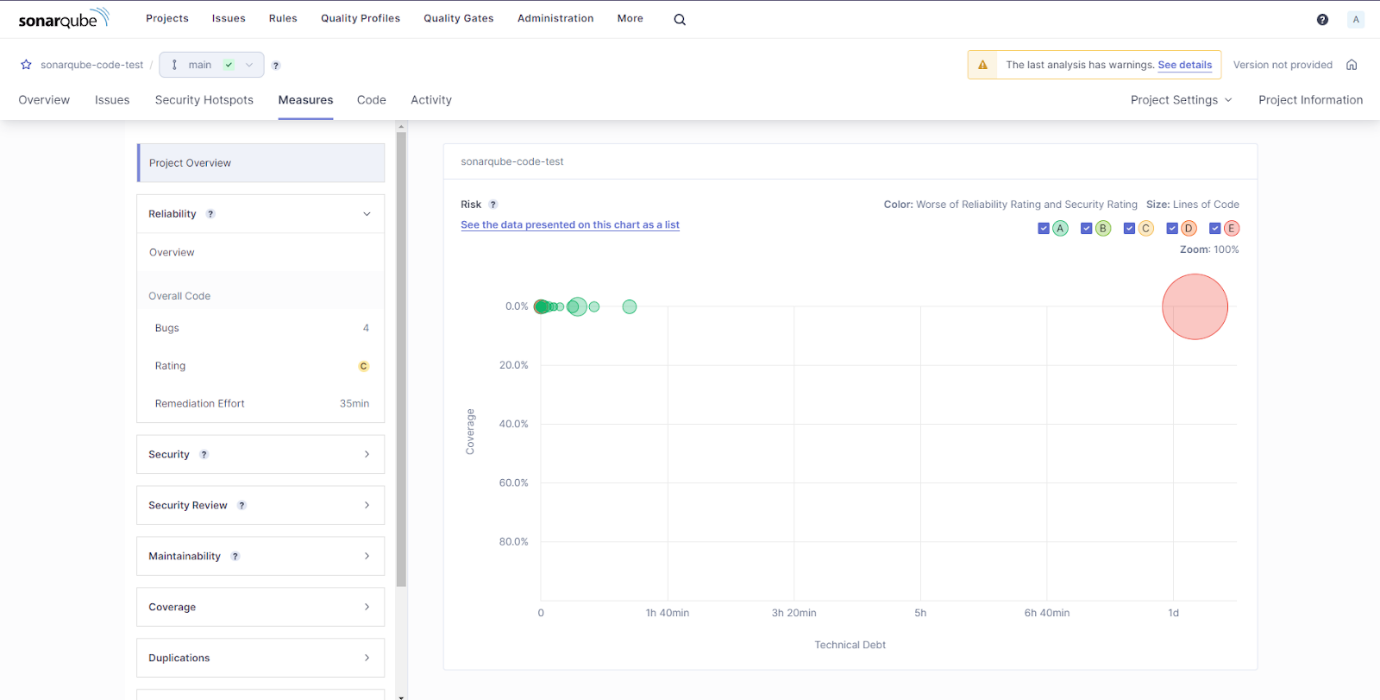






9. View Analysis Results





**Conclusion:**

In this experiment, we successfully implemented Static Application Security Testing (SAST) using SonarQube. By integrating SonarQube into our development workflow, we can proactively identify and address security vulnerabilities and code quality issues in our software projects, leading to more secure and maintainable code.