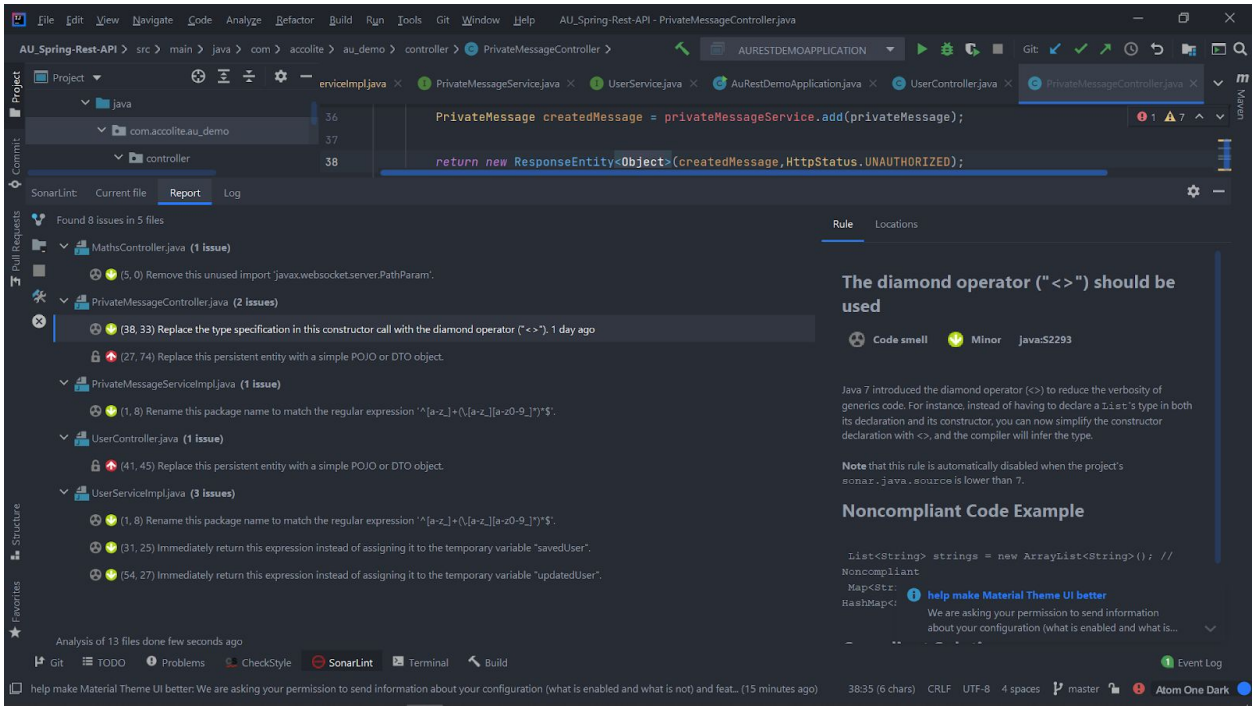
Java Code Quality Assignment

Prabhakar

Code Analyzers:
SonarLint:

- Install in Intellij File->settings->plugin->type sonarlint and install it.
- Once the installation is over, need to restart the IDE.
- Scanned project with 8 class files and encountered few issues



Package names should comply with a naming convention:

- Shared coding conventions allow teams to collaborate efficiently. This rule checks that all package names match a provided regular expression.

Code : `package com.accolite.au_demo.service.Impl;`

Local variables should not be declared and then immediately returned or thrown:

- Declaring a variable only to immediately return or throw it is a bad practice.

Code : `User updatedUser = userRepository.save(user);`

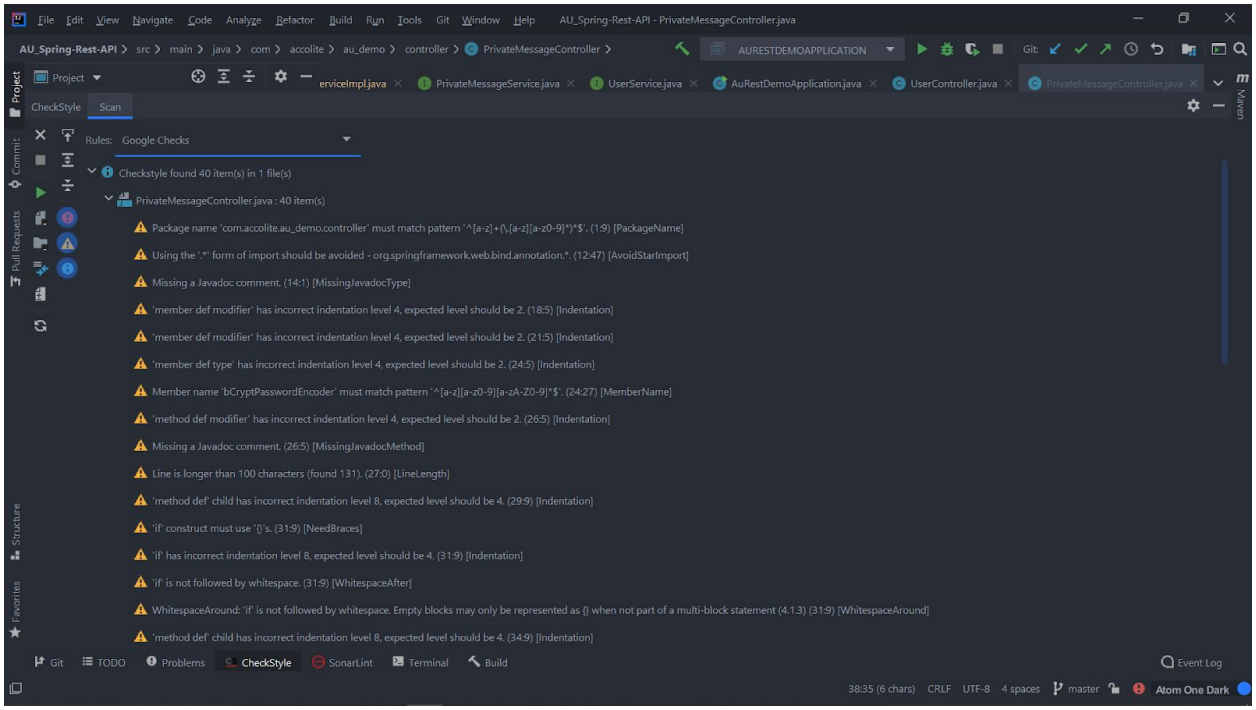
The diamond operator ("<>") should be used:

- Java 7 introduced the diamond operator (<>) to reduce the verbosity of generics code. For instance, instead of having to declare a List's type in both its declaration and its constructor, you can now simplify the constructor declaration with <>, and the compiler will infer the type.

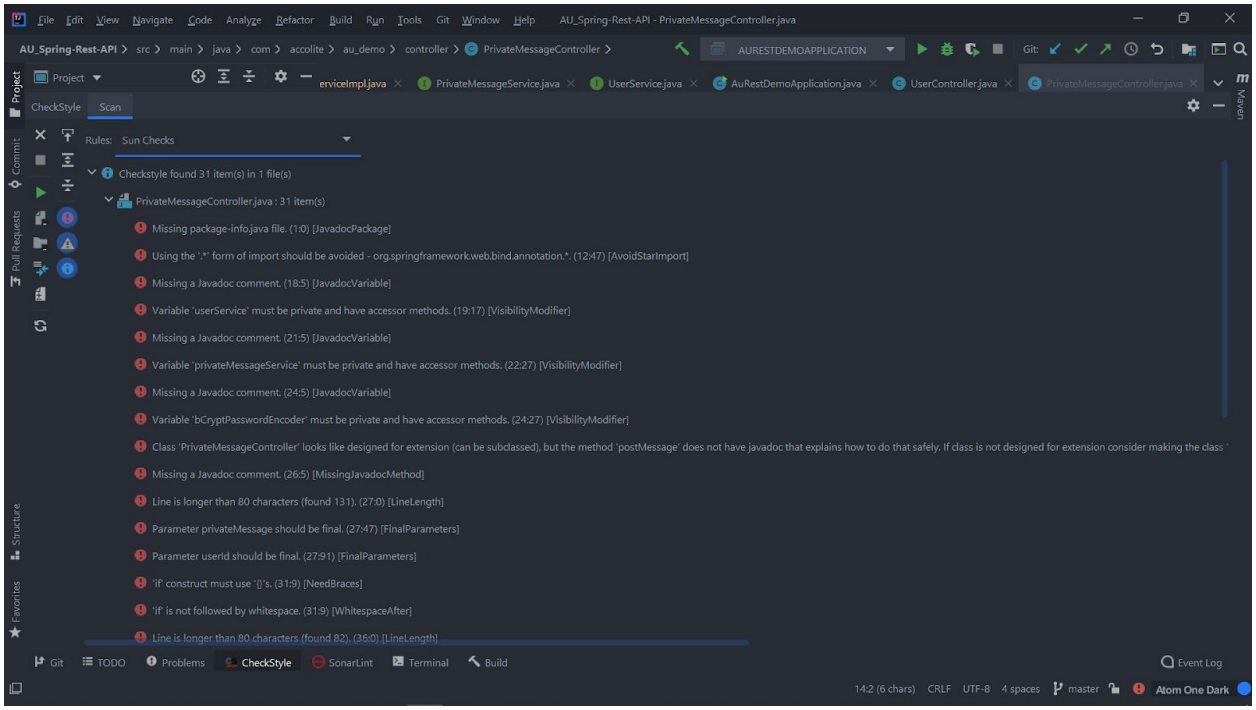
Code: `return new ResponseEntity<Object>(createdMessage,HttpStatus.UNAUTHORIZED);`

CheckStyle:

- Install in Intellij File->settings->plugin->type sonarlint and install it.
- Once the installation is over, need to restart the IDE.
- Scanned google check a class file and got 40 warnings



- Scanned with sun check and got 31 errors

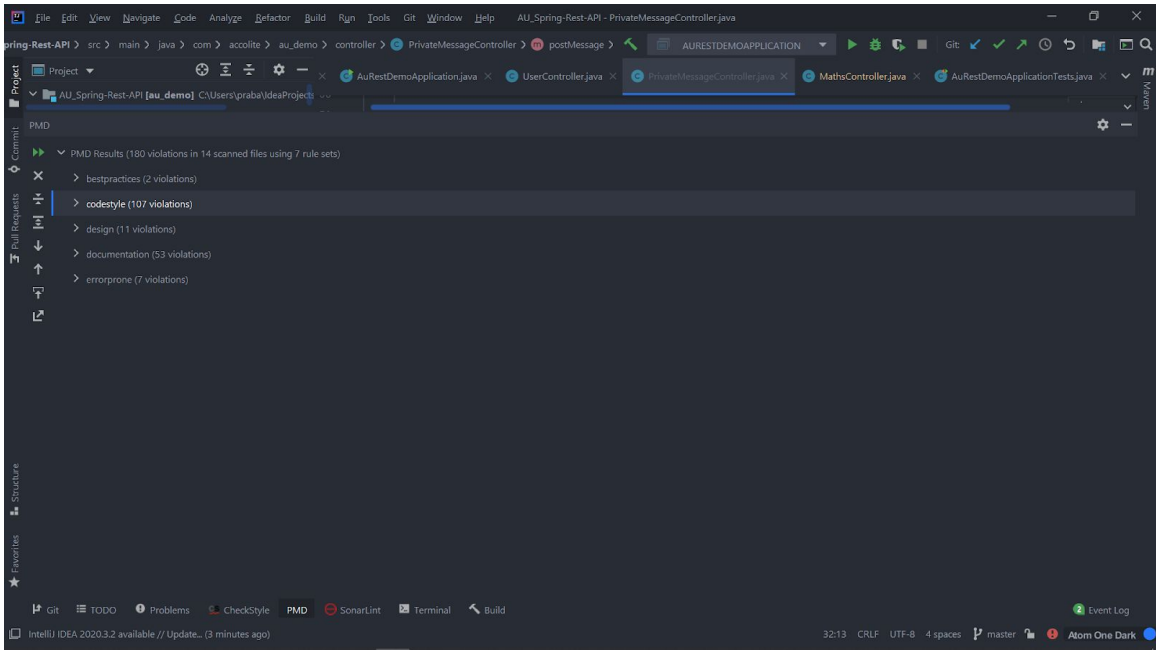


PMD:

- A plugin to run static analysis using PMD in intelliJ.
- Install in Intellij File->settings->plugin->type sonarlint and install it.
- Once the installation is over, need to restart the IDE.
- Scanned the project with 14 files and got 180 violations.

Violations:

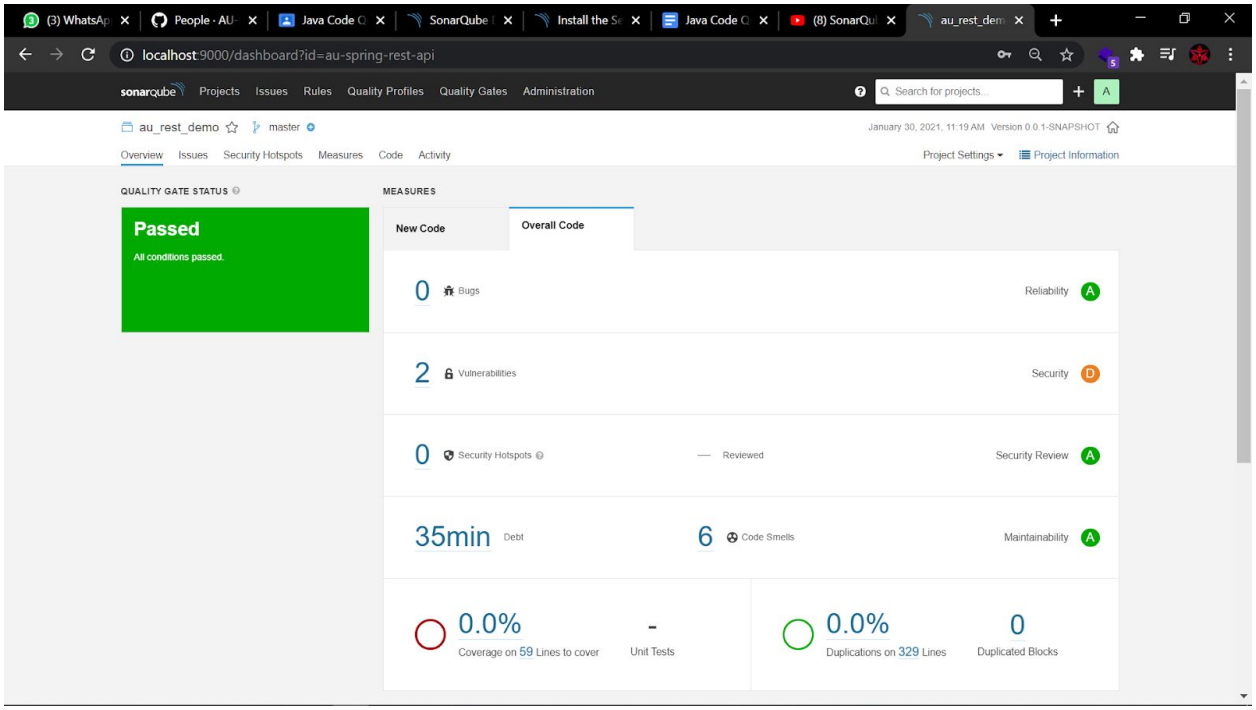
- Best practices
- Code style
- Design
- Documentation
- error prone



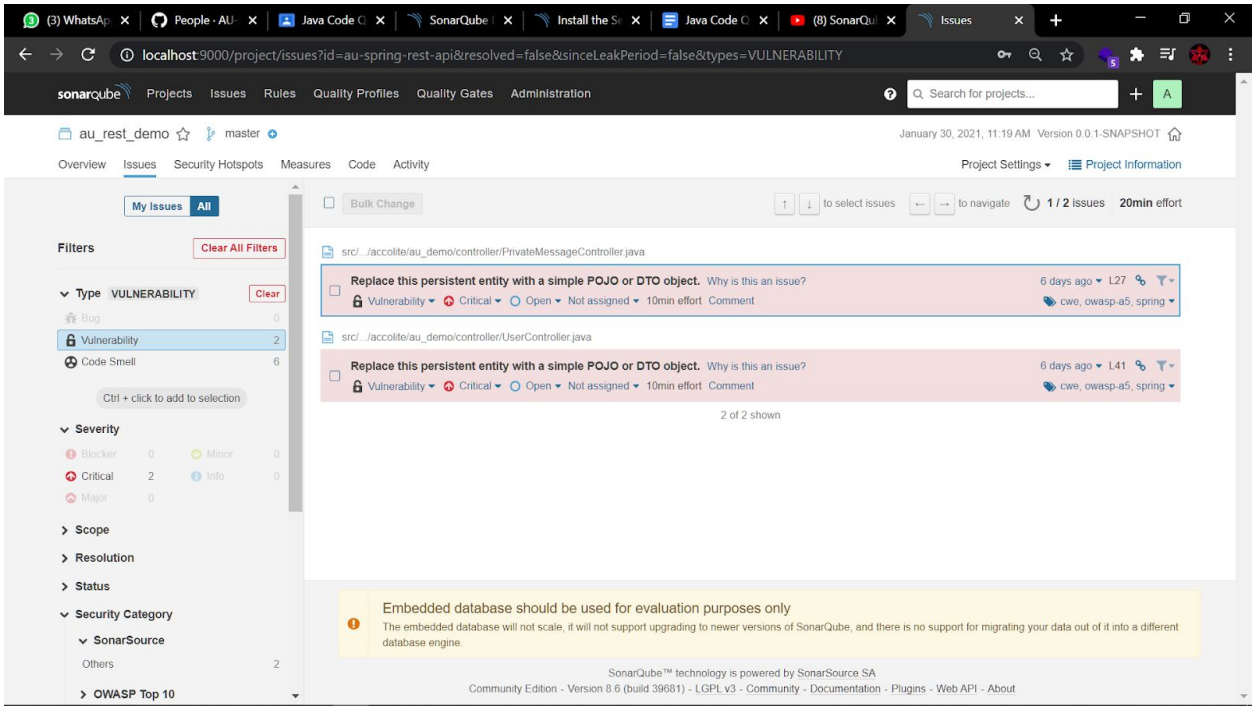
SonarQube:

- sonarQube is installed and configured in /conf/wrapper.conf
- Scanned the project with 7 java class files.

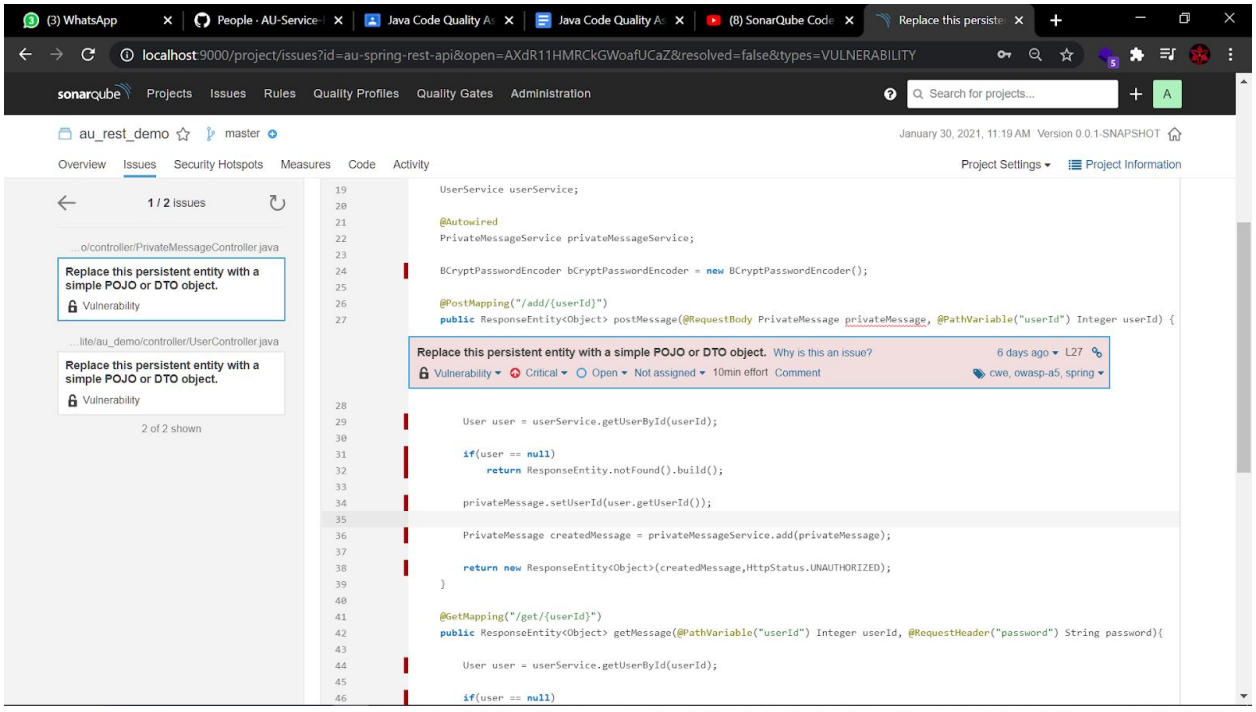
SonarQube home:



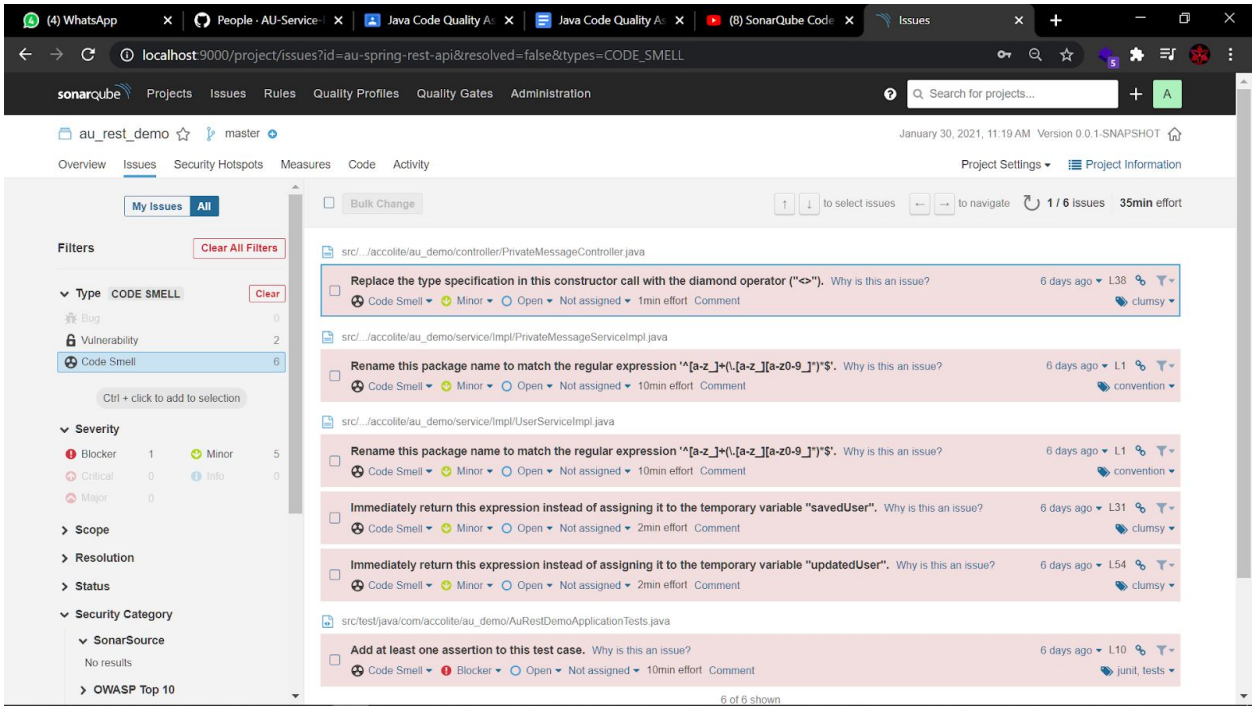
Vulnerabilities:



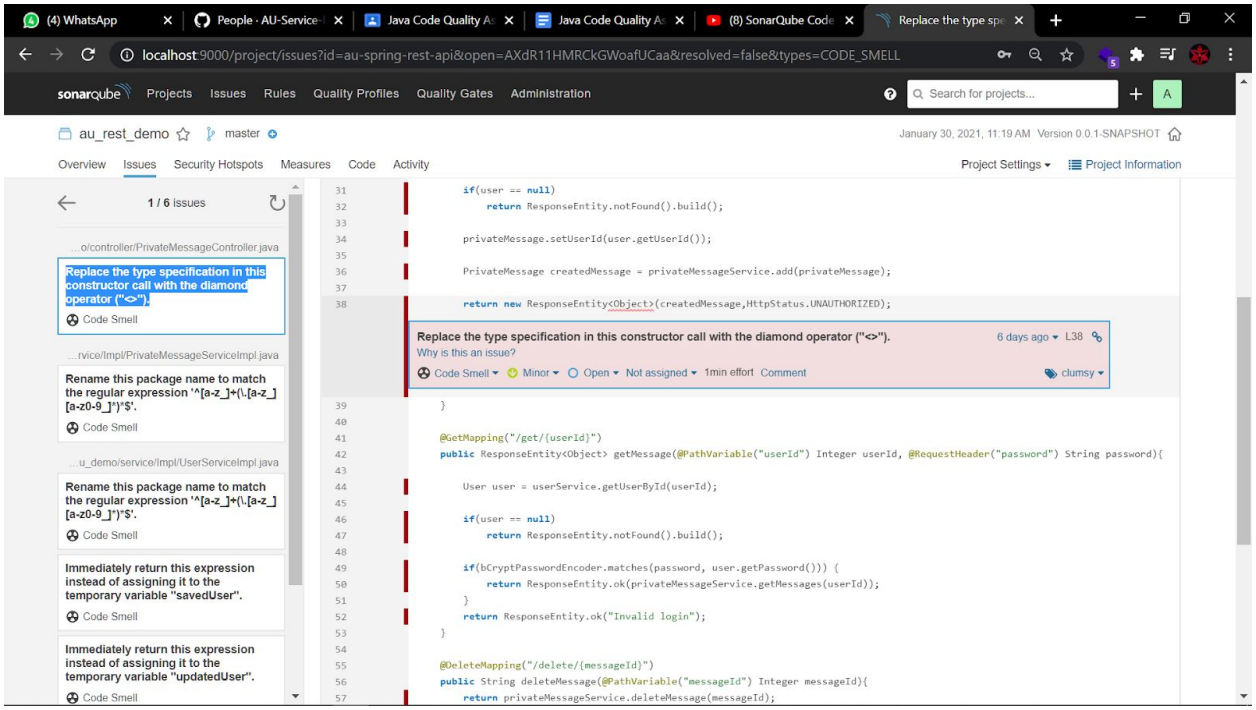
Replace this persistent entity with a simple POJO or DTO object:



Code Smell:



Replace the type specification in this constructor call with the diamond operator ("<>"):



CWE - Common Weakness Enumeration:

- CWE™ is a community-developed list of software and hardware weakness types.
- It serves as a common language, a measuring stick for security tools, and as a baseline for weakness identification, mitigation, and prevention efforts.

Benefits:

- Reducing time and effort to triage, validate, and prioritize reports;
- Improving the alignment of expectations for report resolution and bounty payouts;
- Improving interoperability with external systems;
- Improving interoperability with custom internal taxonomies;
- Unlocking advanced analytics.

OWASP Top 10:

The OWASP Top 10 is a standard awareness document for developers and web application security. It represents a broad consensus about the most critical security risks to web applications.

Security Risks and Prevention

1. Injection

Prevention - Application security testing

2. Broken Authentication and Session Management

Prevention- Multi-factor authentication

3. Sensitive Data Exposure

Prevention - Encryption of data at rest and in transit

4. XML External Entity

Prevention - Static application security testing (SAST)

5. Broken Access Control

Prevention - Penetration testing

6. Security Misconfiguration

Prevention - Dynamic application security testing (DAST)

7. Cross-Site Scripting

Prevention - Developer training complements security testing

8. Insecure deserialization

Prevention - Application security tools can detect deserialization flaws

9. Using Components With Known Vulnerabilities

Prevention - Software composition analysis

10. Insufficient Logging and Monitoring

Prevention - Think like an attacker

CERT

- The CERT Division is a leader in cybersecurity. They partner with government, industry, law enforcement, and academia to improve the security and resilience of computer systems and networks.
- They study problems that have widespread cybersecurity implications and develop advanced methods and tools to counter large-scale, sophisticated cyber threats.

Functionalities of area CERT serve:

- Collection,analysis and dissemination of information on cyber incidents.
- Forecast and alerts of cyber security incidents
- Emergency measures for handling cyber security incidents
- Coordination of cyber incident response activities.
- Issue guidelines,advisories,vulnerability notes and whitepapers relating to information security practices,procedures, prevention,response and reporting of cyber incidents.
- Such other functions relating to cyber security as may be prescribed

SANS 25

The SANS Top 25 Most Dangerous Software Errors is a list of the most widespread and critical errors that can lead to serious vulnerabilities in software (please note: not all vulnerability types apply to all programming languages). The vulnerabilities include insecure interaction between components, risky resource management, and porous defenses.

Errors:

- Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')
- Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection')
- Buffer Copy without Checking Size of Input ('Classic Buffer Overflow')
- Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')
- Missing Authentication for Critical Function

- Missing Authorization
- Use of Hard-coded Credentials
- Missing Encryption of Sensitive Data
- Unrestricted Upload of File with Dangerous Type
- Reliance on Untrusted Inputs in a Security Decision
- Execution with Unnecessary Privileges
- And so on.