SonarQube Project Analysis

Pre-Requisites:

1. You have Java JDK 17 installed on your machine. (See Troubleshooting below).
   - Check via: *java –version*
   - If you are doing it on Lab machine, then you need to have Java JDK 17 rather than 11 (already installed).
2. Your Uranus application is ready to compile or verify.
   - Check crypto-back via: *mvn clean install* OR *mvn clean verify*.
   - If you are on lab machines, then set up your MySQL connection applications.properties to seng-sec-25.seng.uvic.ca and use your username, database name and password provided. Otherwise set it to your MySQL on your own machine.
   - You may use the one I have already provided to avoid multiple errors. (https://gitfront.io/r/ashiqullahmg/8wWxtYEEyYPh/cryptonight-final/).

Analyzing Your Project: (Backend)

1. Once your SonarQube instance is up and running, log in to http://localhost:9000 using System Administrator credentials: login: admin password: admin
2. Now that you're logged in to your local SonarQube instance, let's analyze our project:
3. Select Create new local project.
4. Give your project a Project key and a Display name as “crypto-back” for backend code analysis and select Next.
5. In second step select “Use the global setting” and click on “Create Project”.
6. In next step “Analysis Method”, select locally again.
7. Under Provide a token, select Generate a token. Give your token a name if you want, select Generate, and click Continue.
8. Select your project's main language under Run analysis on your project, for the backend choose Maven, for this you don’t need the sonar-scanner, since it downloads it automatically, but when you want to analyze the front-end which is a TS project, then you may select “Other (for JS, TS, Go, Python, PHP, ...), which needs sonar-scanner, that you have already installed on your machine last time.
9. In “Execute the Scanner for Maven” copy the command and run that command in your crypto-back, the backend of your project directory. The command looks like this. (See troubleshooting).
   - mvn clean verify sonar:sonar \
     -Dsonar.projectKey=ucrypt-backend \
     -Dsonar.projectName='ucrypt-backend' \
     -Dsonar.host.url=http://localhost:9000 \
     -Dsonar.token=sqp_3b3f567f390fcef3f2d6415c3863asdfde2f4b9c8f4a
   - Wait for the command to execute successfully. Once the execution is done, you will see the results on your SonarQube project.
10. Repeat this for frontend “UCryptPortal) as well, make sure to select Other… in step 8 for that.

Analyze SonarQube Reports:

Go to issues tab in the report and analyze the report as below:

1. Issue Identification: Review issues reported by SonarQube focusing solely on vulnerabilities and security hotspots. Ignore bugs and code smells unless they directly impact security.
2. Issue Analysis: Determine if each reported issue is a false positive or a genuine vulnerability. Utilize the details panel (accessed via the ellipsis icon next to the issue's title) for a deeper understanding of each issue.
3. Classification Update: Re-classify issues based on their impact and likelihood using the Impact/Likelihood matrix to correct any potential misclassifications by the tool.
4. Documentation: Document the rationale behind each classification and action taken for future reference.

Possible Troubleshooting:

1. On windows you may copy the above command and paste it in any browser's URL (address) bar to remove the line breaks, copy it from the address bar and then run it in traditional command prompt (CMD).
2. If you face any java related error, have Java JDK 17 on your machine level.
3. If you face any database error, you may double check your MySQL connection. In case you get schema migration error, delete database and create again.
4. To have a complete static code analysis and to avoid verifying (this will also help you skip the updating java JDK to version to 17), remove the *clean verify* from the command above:
    - mvn sonar:sonar \
      -Dsonar.projectKey=ucrypt-backend \
      -Dsonar.projectName='ucrypt-backend' \
      -Dsonar.host.url=http://localhost:9000 \
      -Dsonar.token=sqp_3b3f567f390fcef3f2d6fasdf415c386bdf4b9c8f4a