1.1.2 Build Tools

Required Build Tools

1. Maven
2. Eclipse
3. Bitbucket
4. Jenkins CI
5. AdmitOne Maven Repository

Static Code Analysis

This project will utitlize the following automated analysis tools:

1. Code Formatting - formatting java source code using the Eclipse code formatter
2. CheckStyle - automates the process of checking Java code
3. SonarLint and SonarQube - identifies and helps you fix quality and security issues
4. OWASP Dependency Checking - attempts to detect publicly disclosed vulnerabilities contained within a project's dependencies

Maven Plugins

All Java artifacts will implement the following Maven build plugins and reporting.

1. Code Formatting.

Use the maven build plugin formatter-maven-plugin configured to use **JavaLambdaFormat** style. This style configuration is provied by the **admitone:code-quality-config** resource plugin.

Standard setup for formatting code is

```xml
<!-- Automatically reformat source code to Java Lambda Format -->
<plugin>
    <groupId>net.revelc.code.formatter</groupId>
    <artifactId>formatter-maven-plugin</artifactId>
    <version>2.18.0</version>
    <executions>
        <execution>
            <goals>
                <goal>format</goal>
            </goals>
            <configuration>
                <configFile>eclipse/JavaLambdaFormat.xml</configFile>
            </configuration>
        </execution>
```

```xml
            </executions>
            <configuration>
                <encoding>UTF-8</encoding>
                <excludes>
                    <exclude>**/*Test.java</exclude>
                    <exclude>**/*_Spec.java</exclude>
                </excludes>
            </configuration>
            <dependencies>
                <dependency>
                    <groupId>admitone</groupId>
                    <artifactId>code-quality-config</artifactId>
                    <version>2022.03.01</version>
                </dependency>
            </dependencies>
        </plugin>
```

2. Setting up Checkstyle

Checkstyle will help educate and enforce our coding standards. You can set up your IDE to use Checkstyle to examine code for conformance to the standards. Learn more about the checks or Google the error message to find out why it complains about certain things.

```xml
    <!-- Verify code matches the AdmitOne Rout66 Java Lambda Function
 Requirements-->
    <plugin>
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-checkstyle-plugin</artifactId>
        <version>3.1.2</version>
        <configuration>
            <configLocation>checkstyle.xml</configLocation>
        </configuration>
        <dependencies>
            <dependency>
                <groupId>com.puppycrawl.tools</groupId>
                <artifactId>checkstyle</artifactId>
                <version>10.0</version>
            </dependency>

            <dependency>
                <groupId>admitone</groupId>
                <artifactId>code-quality-config</artifactId>
                <version>2022.03.01</version>
            </dependency>
        </dependencies>
    </plugin>
```

3. SonarLint and SonarQube with SonarSannar

Execute the SonarLint and SonarQube analysis via a regular Maven SonarScanner.

TODO: add setup here

4. OWASP Dependency Checking

*TODO_*: add setup here

OWASP Dependency-Check

Eclipse

TODO

Jenkins

URL - http://admitone.ci:50080/jenkins/

AdmitOne Repository

TODO:

URL - http://admitone.repo:52180/