# CS 305 Project One Template

## Document Revision History

| **Version** | **Date** | **Author** | **Comments** |
| --- | --- | --- | --- |
| **1.0** | **05/22/2025** | **Steven Klabunde** |  |

## Client



## Instructions

Submit this completed vulnerability assessment report. Replace the bracketed text with the relevant information. In this report, identify your security vulnerability findings and recommend the next steps to remedy the issues you have found.

* Respond to the five steps outlined below and include your findings.
* Respond using your own words. You may also include images or supporting materials. If you include them, make certain to insert them in the relevant locations in the document.
* Refer to the Project One Guidelines and Rubric for more detailed instructions about each section of the template.

## Developer

Steven Klabunde

**1. Interpreting Client Needs**

Determine your client’s needs and potential threats and attacks associated with the company’s application and software security requirements. Consider the following questions regarding how companies protect against external threats based on the scenario information:

* What is the value of secure communications to the company?
* Are there any international transactions that the company produces?
* Are there governmental restrictions on secure communications to consider?
* What external threats might be present now and in the immediate future?
* What modernization requirements must be considered, such as the role of open-source libraries and evolving web application technologies?

Because Artemis Financial is involved in the financial planning of its customers, they have access to a lot of confidential information. They may have access to account numbers, social security numbers, and personal information such as addresses and birthdates. If someone were able to hack into the system, they may be able to take advantage of that and ruin customers’ credit, steal money, or steal identities. In order for Artemis Financial to protect their clients, they need to make sure that all transactions and communications are safe and secure. At this point, we have not been told whether any international transactions will be performed but it would be prudent for us to assume that may happen at some point. This could be an international client or a client with a foreign bank account. There may even be some international stock trading. Financial advisors are strictly monitored by numerous agencies because of the confidential information held. We will want to make sure that all protocols follow the requirements of the SEC, HIPPA, and any other applicable agencies. The biggest external threat would be cyberattacks with an individual or group of individuals looking to access account information. There could also be a risk of a denial of service when trying to make transactions which could definitely affect the quality of the financial management. Any outside programs that are linked into the server could also pose a risk. Open source libraries should be included in code in order to lessen the amount of code. With that though, you lose some control over the code. Therefore, additional precautions need to be taken. That is also the case with web applications that the system is working with. Even if there was no threat previously, an update to that application could change that.

**2. Areas of Security**

Refer to the vulnerability assessment process flow diagram. Identify which areas of security apply to Artemis Financial’s software application. Justify your reasoning for why each area is relevant to the software application.

All areas of the vulnerability assessment should be addressed in order to verify everything is as secure as possible. One of the most vulnerable processes is input validation. Since this is where an attacker has their way in is though input, making sure that they are not altering any other code or gaining administrative access is crucial to keeping the software safe. This is crucial with the information being stored. Following very closely with that is the API. You want to make sure that the application interface is secure because there is communication between the API and other software. Assessing the cryptography vulnerability involves looking at the encryption being used. Encryption plays an important role in verifying that all data remains safe. Both incoming and outgoing encryption should be assessed. Another of the most important processes to access is the client/server. This is where the information coming from the user is transmitted to the server. This is a key step where an attacker can be stopped with the appropriate actions. The next two areas go hand in hand but are different. Code errors involve how errors including incoming errors are handled whereas code quality involves looking at how securely the code is written. Lastly, there is the area of encapsulation. Encapsulation can play an important part of security as it determines what objects and methods are private versus public and therefore how it can be modified. As this is showing, all of these are important and none of them can be ignored throughout the development process.

**3. Manual Review**

Continue working through the vulnerability assessment process flow diagram. Identify all vulnerabilities in the code base by manually inspecting the code.

* Input validation - When looking at the input, there is not any input validation. We would be looking for length limits and potential character limits
* Cryptography - There is no evidence of encryption in the code. There is no java.crypto package listed.
* Code Errors - Although one try catch statement is used, more could be used such as in the rest service application. This would help eliminate code error.
* Encapsulation - In customer, the account balance method should be a private method in order to keep that information protected from being changed manually.
* Client/Server - There is no login requirement with a password verification that verifies a secure password
* Code quality – There is a lack of code comments in the code. This could lead to another developer changing the code in an unsafe manner.
* API - It does not appear that requests are run through HTTPS

**4. Static Testing**

Run a dependency check on Artemis Financial’s software application to identify all security vulnerabilities in the code. Record the output from the dependency-check report. Include the following items:

* The names or vulnerability codes of the known vulnerabilities
* A brief description and recommended solutions provided by the dependency-check report
* Any attribution that documents how this vulnerability has been identified or documented previously,

|  |  |  |  |
| --- | --- | --- | --- |
| **Dependency** | **Vulnerability Codes** | **Description** | **Recommended Solution** |
| [bcprov-jdk15on-1.46.jar](file:///C:\Users\Sklab\OneDrive\Desktop\Eclipse%20Files\rest-service\target\dependency-check-report.html#l1_991c96a4e31e6c19e2b9136c8955bd423f2dc4c7) | [**CVE-2016-1000338**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2016-1000338)  **CVE-2016-1000342** (OSSINDEX)  **CVE-2016-1000343** (OSSINDEX)  **CVE-2024-29857** (OSSINDEX)  **CVE-2016-1000341** (OSSINDEX)  **CVE-2016-1000345** (OSSINDEX)  **CVE-2024-30171** (OSSINDEX)  [**CVE-2020-15522**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2020-15522)  **CVE-2020-0187** (OSSINDEX)  [**CVE-2023-33202**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2023-33202)  **CVE-2020-26939** (OSSINDEX)  **CVE-2023-33201** (OSSINDEX)  [**CVE-2015-7940**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2015-7940)  **CVE-2018-5382** (OSSINDEX)  **CVE-2013-1624** (OSSINDEX)  **CVE-2016-1000346** (OSSINDEX)  **CVE-2015-6644** (OSSINDEX) | The software does not properly ensure that the certificate is associated with the host and does not fully validate encoding of signature. | It appears that the Bouncy Castle software may be out of date. |
| [hibernate-validator-6.0.18.Final.jar](file:///C:\Users\Sklab\OneDrive\Desktop\Eclipse%20Files\rest-service\target\dependency-check-report.html#l3_7fd00bcd87e14b6ba66279282ef15efa30dd2492) | **CVE-2023-1932** (OSSINDEX)  [**CVE-2020-10693**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2020-10693) | A flaw was found in the isValid method | May need to apply a FixPack |
| [jackson-databind-2.10.2.jar](file:///C:\Users\Sklab\OneDrive\Desktop\Eclipse%20Files\rest-service\target\dependency-check-report.html#l5_0528de95f198afafbcfb0c09d2e43b6e0ea663ec) | [**CVE-2020-25649**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2020-25649)  [**CVE-2020-36518**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2020-36518)  [**CVE-2021-46877**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2021-46877)  [**CVE-2022-42003**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-42003)  [**CVE-2022-42004**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-42004)  [**CVE-2023-35116**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2023-35116) | Flaw in Jackson Databind allows a Stack Overflow exception, may cause a denial of service and resource exhaustion | Upgrade software to most current version |
| [log4j-api-2.12.1.jar](file:///C:\Users\Sklab\OneDrive\Desktop\Eclipse%20Files\rest-service\target\dependency-check-report.html#l10_a55e6d987f50a515c9260b0451b4fa217dc539cb) | [**CVE-2020-9488**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2020-9488) | Improper Certification Validation | Upgrade Apache software |
| [logback-classic-1.2.3.jar](file:///C:\Users\Sklab\OneDrive\Desktop\Eclipse%20Files\rest-service\target\dependency-check-report.html#l12_7c4f3c474fb2c041d8028740440937705ebb473a) | [**CVE-2023-6378**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2023-6378)  [**CVE-2021-42550**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2021-42550) | Deserialization of Untrusted Data allowing attacker to send poisonous data | Upgrade to a correct logback version |
| [logback-core-1.2.3.jar](file:///C:\Users\Sklab\OneDrive\Desktop\Eclipse%20Files\rest-service\target\dependency-check-report.html#l13_864344400c3d4d92dfeb0a305dc87d953677c03c) | [**CVE-2023-6378**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2023-6378)  [**CVE-2021-42550**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2021-42550)  **CVE-2024-12798** (OSSINDEX)  **CVE-2024-12801** (OSSINDEX) | Deserialization of Untrusted Data allowing attacker to send poisonous data | Upgrade to a correct logback version |
| [snakeyaml-1.25.jar](file:///C:\Users\Sklab\OneDrive\Desktop\Eclipse%20Files\rest-service\target\dependency-check-report.html#l15_8b6e01ef661d8378ae6dd7b511a7f2a33fae1421) | [**CVE-2022-1471**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-1471)  [**CVE-2017-18640**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2017-18640)  [**CVE-2022-25857**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-25857)  [**CVE-2022-38749**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-38749)  [**CVE-2022-38751**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-38751)  [**CVE-2022-38752**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-38752)  [**CVE-2022-41854**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-41854)  [**CVE-2022-38750**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-38750) | Constructor class does not restrict types | Upgrade SnakeYami to version 2.0 or above |
| [spring-boot-2.2.4.RELEASE.jar](file:///C:\Users\Sklab\OneDrive\Desktop\Eclipse%20Files\rest-service\target\dependency-check-report.html#l16_225a4fd31156c254e3bb92adb42ee8c6de812714) | [**CVE-2023-20873**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2023-20873)  [**CVE-2022-27772**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-27772)  [**CVE-2023-20883**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2023-20883) | Older version of Spring Boot is susceptible to security bypass and temporary directory highjacking | Upgrade Spring Boot to 3.0.6+ or 2.7.11+ |
| [spring-boot-starter-web-2.2.4.RELEASE.jar](file:///C:\Users\Sklab\OneDrive\Desktop\Eclipse%20Files\rest-service\target\dependency-check-report.html#l17_ec75d01d212b5229c16d872fb127744c0ed46ed8) | [**CVE-2023-20873**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2023-20873)  [**CVE-2022-27772**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-27772)  [**CVE-2023-20883**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2023-20883) | Older version of Spring Boot is susceptible to security bypass and temporary directory highjacking | Upgrade Spring Boot to 3.0.6+ or 2.7.11+ |
| [spring-context-5.2.3.RELEASE.jar](file:///C:\Users\Sklab\OneDrive\Desktop\Eclipse%20Files\rest-service\target\dependency-check-report.html#l18_7750c95c96c7a1885c8b1b503ba915bc33ca579a) | [**CVE-2022-22965**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-22965)  [**CVE-2021-22118**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2021-22118)  [**CVE-2020-5421**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2020-5421)  [**CVE-2022-22950**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-22950)  [**CVE-2022-22971**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-22971)  [**CVE-2023-20861**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2023-20861)  [**CVE-2023-20861**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2023-20861)  [**CVE-2022-22968**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-22968)  [**CVE-2022-22970**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-22970)  [**CVE-2021-22060**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2021-22060)  [**CVE-2021-22096**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2021-22096)  **CVE-2025-22233** (OSSINDEX) | May be vulnerable to remote control execution via data binding or to privilege escalation | Upgrade Spring Framework to current version |
| [spring-core-5.2.3.RELEASE.jar](file:///C:\Users\Sklab\OneDrive\Desktop\Eclipse%20Files\rest-service\target\dependency-check-report.html#l19_3734223040040e8c3fecd5faa3ae8a1ed6da146b) | [**CVE-2022-22965**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-22965)  [**CVE-2021-22118**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2021-22118)  [**CVE-2020-5421**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2020-5421)  [**CVE-2022-22950**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-22950)  [**CVE-2022-22971**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-22971)  [**CVE-2023-20861**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2023-20861)  [**CVE-2023-20863**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2023-20863)  [**CVE-2022-22968**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-22968)  [**CVE-2022-22970**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-22970)  [**CVE-2021-22060**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2021-22060)  [**CVE-2021-22096**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2021-22096) | May be vulnerable to remote control execution via data binding or to privilege escalation | Upgrade Spring Framework to current version |
| [spring-expression-5.2.3.RELEASE.jar](file:///C:\Users\Sklab\OneDrive\Desktop\Eclipse%20Files\rest-service\target\dependency-check-report.html#l20_d0c6bb10758805b2153c589686b8045554bfac2d) | [**CVE-2022-22965**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-22965)  [**CVE-2021-22118**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2021-22118)  [**CVE-2020-5421**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2020-5421)  [**CVE-2022-22950**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-22950)  [**CVE-2022-22971**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-22971)  [**CVE-2023-20861**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2023-20861)  [**CVE-2023-20863**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2023-20863)  **CVE-2024-38808** (OSSINDEX)  [**CVE-2022-22968**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-22968)  [**CVE-2022-22970**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-22970)  [**CVE-2021-22060**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2021-22060)  [**CVE-2021-22096**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2021-22096) | May be vulnerable to remote control execution via data binding or to privilege escalation | Upgrade Spring Framework to current version |
| [spring-web-5.2.3.RELEASE.jar](file:///C:\Users\Sklab\OneDrive\Desktop\Eclipse%20Files\rest-service\target\dependency-check-report.html#l21_dd386a02e40b915ab400a3bf9f586d2dc4c0852c) | [**CVE-2016-1000027**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2016-1000027)  [**CVE-2022-22965**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-22965)  **CVE-2024-38809** (OSSINDEX)  **CVE-2024-22243** (OSSINDEX)  **CVE-2024-22262** (OSSINDEX)  [**CVE-2021-22118**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2021-22118)  **CVE-2024-38828** (OSSINDEX)  [**CVE-2020-5421**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2020-5421)  [**CVE-2022-22950**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-22950)  [**CVE-2022-22971**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-22971)  [**CVE-2023-20861**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2023-20861)  [**CVE-2023-20863**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2023-20863)  [**CVE-2022-22968**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-22968)  [**CVE-2022-22970**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-22970)  [**CVE-2021-22060**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2021-22060)  [**CVE-2021-22096**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2021-22096) | Spring Framework suffers from a potential remote code execution | Upgrade Spring Framework to current version |
| [spring-webmvc-5.2.3.RELEASE.jar](file:///C:\Users\Sklab\OneDrive\Desktop\Eclipse%20Files\rest-service\target\dependency-check-report.html#l22_745a62502023d2496b565b7fe102bb1ee229d6b7) | [**CVE-2022-22965**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-22965)  **CVE-2024-38816** (OSSINDEX)  [**CVE-2021-22118**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2021-22118)  [**CVE-2020-5421**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2020-5421)  [**CVE-2022-22950**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-22950)  [**CVE-2022-22971**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-22971)  [**CVE-2023-20861**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2023-20861)  [**CVE-2023-20863**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2023-20863)  [**CVE-2022-22968**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-22968)  [**CVE-2022-22970**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-22970)  [**CVE-2021-22060**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2021-22060)  [**CVE-2021-22096**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2021-22096) | May be vulnerable to remote control execution via data binding or to privilege escalation | Upgrade Spring Framework to current version |
| [tomcat-embed-core-9.0.30.jar](file:///C:\Users\Sklab\OneDrive\Desktop\Eclipse%20Files\rest-service\target\dependency-check-report.html#l23_ad32909314fe2ba02cec036434c0addd19bcc580) | [**CVE-2020-1938**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2020-1938)  [**CVE-2024-52316**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2024-52316)  [**CVE-2025-24813**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2025-24813)  [**CVE-2025-31651**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2025-31651)  [**CVE-2020-11996**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2020-11996)  [**CVE-2020-13934**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2020-13934)  [**CVE-2020-13935**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2020-13935)  [**CVE-2020-17527**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2020-17527)  [**CVE-2021-25122**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2021-25122)  [**CVE-2021-41079**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2021-41079)  [**CVE-2022-29885**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-29885)  [**CVE-2022-42252**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-42252)  [**CVE-2023-44487**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2023-44487)  [**CVE-2023-46589**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2023-46589)  [**CVE-2024-24549**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2024-24549)  [**CVE-2024-38286**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2024-38286)  [**CVE-2020-9484**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2020-9484)  [**CVE-2021-25329**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2021-25329)  [**CVE-2021-30640**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2021-30640)  [**CVE-2024-23672**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2024-23672)  [**CVE-2022-34305**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-34305)  [**CVE-2023-41080**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2023-41080)  [**CVE-2021-24122**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2021-24122)  [**CVE-2021-33037**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2021-33037)  [**CVE-2023-42795**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2023-42795)  [**CVE-2023-45648**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2023-45648)  [**CVE-2024-21733**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2024-21733)  [**CVE-2019-17569**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2019-17569)  [**CVE-2020-1935**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2020-1935)  [**CVE-2020-13943**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2020-13943)  [**CVE-2023-28708**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2023-28708)  [**CVE-2021-43980**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2021-43980) | Connection to Apache Tomcat may not be secure | Need to verify connection is secure |
| [tomcat-embed-websocket-9.0.30.jar](file:///C:\Users\Sklab\OneDrive\Desktop\Eclipse%20Files\rest-service\target\dependency-check-report.html#l25_33157f6bc5bfd03380ebb5ac476db0600a04168d) | [**CVE-2020-1938**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2020-1938)  [**CVE-2024-52316**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2024-52316)  [**CVE-2025-24813**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2025-24813)  [**CVE-2025-31651**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2025-31651)  [**CVE-2020-8022**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2020-8022)  [**CVE-2020-11996**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2020-11996)  [**CVE-2020-13934**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2020-13934)  [**CVE-2020-13935**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2020-13935)  [**CVE-2020-17527**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2020-17527)  [**CVE-2021-25122**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2021-25122)  [**CVE-2021-41079**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2021-41079)  [**CVE-2022-29885**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-29885)  [**CVE-2022-42252**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-42252)  [**CVE-2023-44487**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2023-44487)  [**CVE-2023-46589**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2023-46589)  [**CVE-2024-24549**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2024-24549)  [**CVE-2024-38286**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2024-38286)  [**CVE-2020-9484**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2020-9484)  [**CVE-2021-25329**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2021-25329)  [**CVE-2021-30640**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2021-30640)  [**CVE-2024-23672**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2024-23672)  [**CVE-2022-34305**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-34305)  [**CVE-2023-41080**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2023-41080)  [**CVE-2021-24122**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2021-24122)  [**CVE-2021-33037**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2021-33037)  [**CVE-2023-42795**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2023-42795)  [**CVE-2023-45648**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2023-45648)  [**CVE-2024-21733**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2024-21733)  [**CVE-2019-17569**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2019-17569)  [**CVE-2020-1935**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2020-1935)  [**CVE-2020-13943**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2020-13943)  [**CVE-2023-28708**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2023-28708)  [**CVE-2021-43980**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2021-43980) | Connection to Apache Tomcat may not be secure | Need to verify connection is secure |

**5. Mitigation Plan**

Interpret the results from the manual review and static testing report. Then identify the steps to mitigate the identified security vulnerabilities for Artemis Financial’s software application.

[Include your findings here.]

* As it appears that the Bouncy Castle software is out of date, will want to upgrade to the latest version.
* As it appears that the logback software is out of date, will want to upgrade to the correct version.
* As it appears that the SnakeYami software is out of date, will want to upgrade to the latest version.
* As it appears that the Springboot software is out of date, will want to upgrade to the correct version.
* Input validation: In greeting, lines 8 – 9 need to validate input, create a length limit, potentially blacklisting and whitelisting
* API: As shown above, make sure all the software is up to date to make sure that the communications between the software is safe
* Cryptography: Make sure that all requests are coming through HTTPS
* Make sure to use an encryption library such as Apache that is up to date
* Client/Server: DocData lines 21 – 30, make sure that user passwords are verified and denied access if incorrect
* Code Error: Include more try/catch statements with methods, such as myDateTime lines 14 -15
* Code Quality: Add in more comments, only comments found were in DocData. The rest should have comments as well
* Encapsulation: Customer – method on line 12 changed to private instead of public

**Citations**

CS 305 Project One Guidelines and Rubric. <https://learn.snhu.edu/d2l/le/content/1918286/viewContent/40563416/View>

Manico, J. & Detlefsen, A. (2014) Iron-Clad Java. <https://learning.oreilly.com/library/view/iron-clad-java/9780071835886/?sso_link=yes&sso_link_from=SNHU>