****

# Practices for Secure Software Report

Table of Contents

[Document Revision History 3](#_Toc102040754)

[Client 3](#_Toc102040755)

[Instructions 3](#_Toc102040756)

[Developer 4](#_Toc102040757)

[1. Algorithm Cipher 4](#_Toc102040758)

[2. Certificate Generation 4](#_Toc102040759)

[3. Deploy Cipher 4](#_Toc102040760)

[4. Secure Communications 4](#_Toc102040761)

[5. Secondary Testing 4](#_Toc102040762)

[6. Functional Testing 4](#_Toc102040763)

[7. Summary 4](#_Toc102040764)

[8. Industry Standard Best Practices 4](#_Toc102040765)

## Document Revision History

| **Version** | **Date** | **Author** | **Comments** |
| --- | --- | --- | --- |
| **1.0** | **12-9-23** | **Andreya Colvin** |  |

## Client



## Developer

Andreya Colvin

## Algorithm Cipher

After researching and looking over the needs of Artemis Financial I believe the best and most efficient algorithm cipher to use is Advanced Encryption Standard (AES). AES is recognized as secure and robust, and it has also been adopted as the standard by the government and the entire world. The purpose of the hash functions and bit levels are to ensure confidentiality. Because AES operates with 128 bits and key sizes of 128, 192, and 256 bits this is what makes it secure with encryption and decryption, it uses the same key for both. The use of random numbers, symmetric versus non-symmetric keys, and so on, are all part of generating secure encryption keys.

## Certificate Generation

A screenshot of a computer

Description automatically generated

## Deploy Cipher

A screenshot of a computer

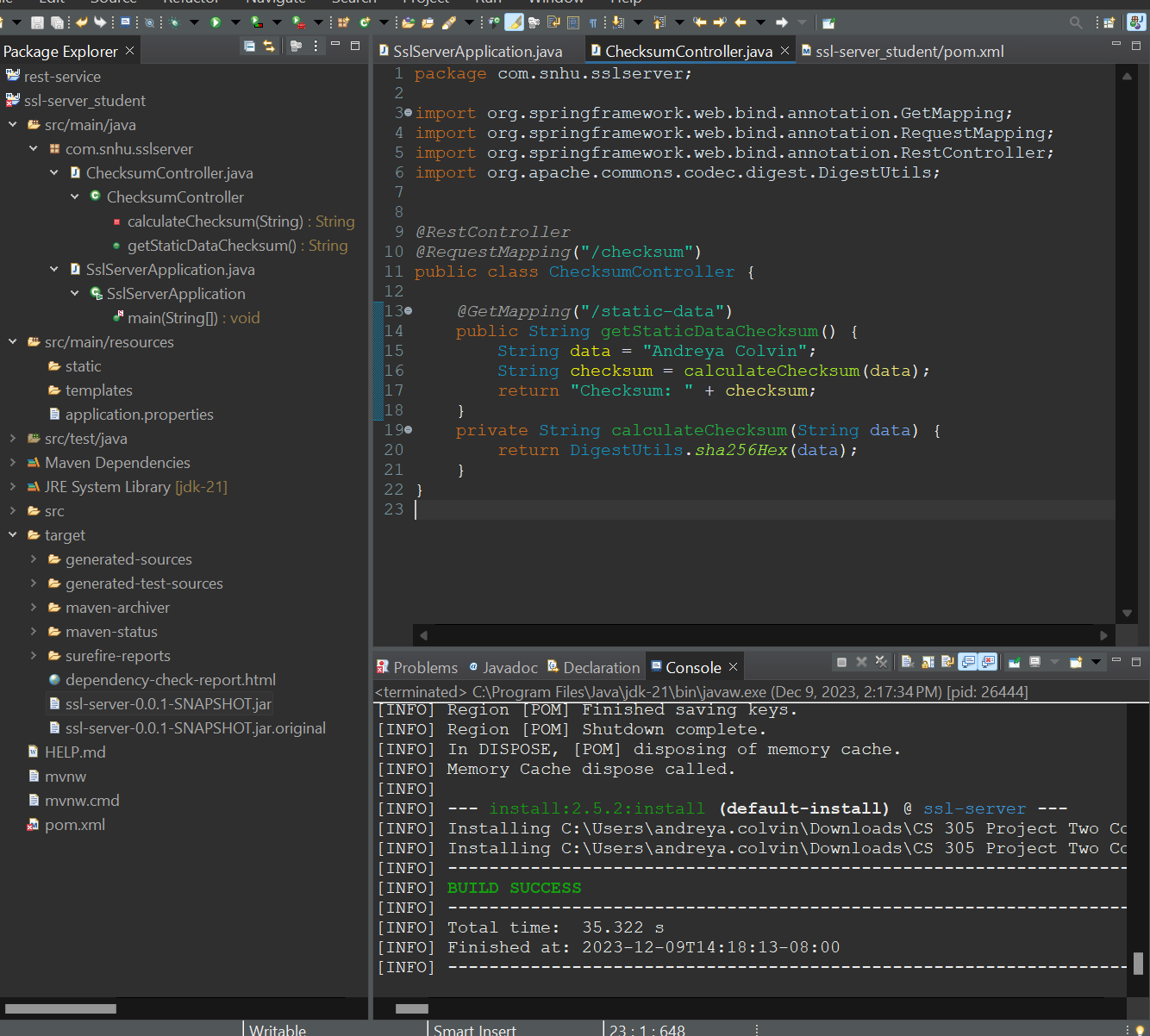
Description automatically generated

## Secure Communications

A screenshot of a computer

Description automatically generated

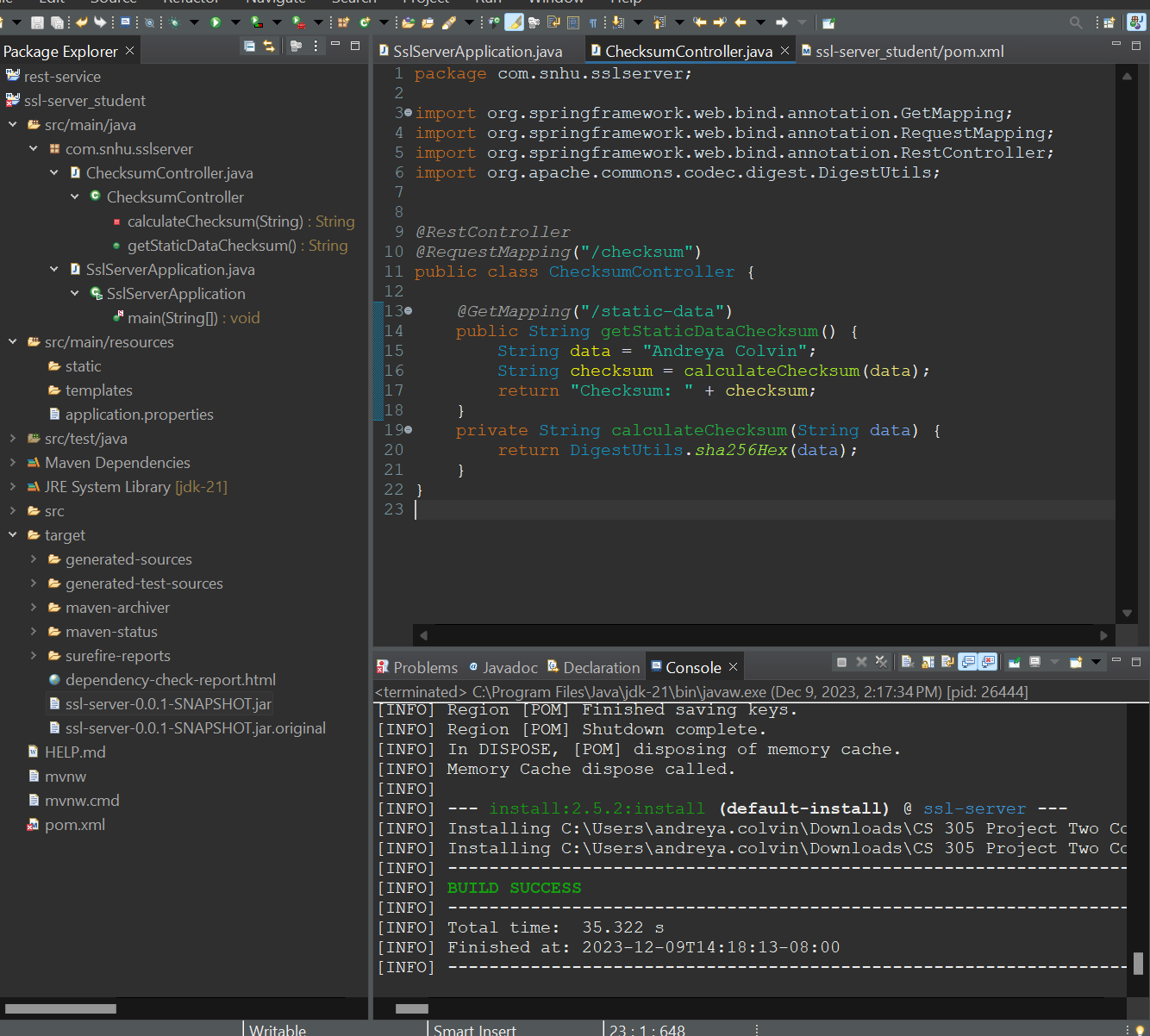
## Secondary Testing



A screenshot of a computer

Description automatically generated

## Functional Testing



## Summary

My code had to be refactored to enhance the security posture. The refactoring process aimed to address the vulnerabilities as well as to improve the over all security of the software application. Some key areas to highlight included input validation, secure communication, cryptographic operations, and integrating security protocols.

Using the vulnerability assessment process flow diagram, I followed as – input validation, secure communication, cryptographic operations, OQSAP dependency check. This is also how I added layers of security, I implemented strict input validation to mitigate risks, then added secure communication to ensure HHTPS was encrypting data, next using the cryptographic operations to have the hash algorithm (256-SHA) calculate checksums, then finally the dependency check to identify potential vulnerabilities.

## Industry Standard Best Practices

In order to maintain current security, I had to implement the standard practices which again consisted of input validation, HTTPS, SHA-256, and the dependency check. Using all of these high-quality standard practices allowed me to keep the company’s software application in a wellbeing state.

This kept the value of the company’s overall wellbeing by using risk mitigation which helped prevent and address the common security vulnerabilities. Staying in regulatory compliance which meets the organizations standards and regulations. This also kept the company’s reputation high because of how they take security seriously and do whatever they can to keep it implemented. With having such a competitive landscape, it was imperative to keep this company in good standing so they can have a high success rate.