# CS 305 Module Two Written Assignment: Alexis Prazak

## Areas of Security

* Input Validation
* Cryptography
* APIs
* Code Quality
* Code Error

## Areas of Security Justification

**Code quality:** It's important to minimize bugs in your code. While having perfect code is practically impossible, bugs shouldn't be ignored just because the code runs well.

**Input Validation:** It's easy for bad actors to insert malicious inputs if there aren’t strict validation requirements. In the current code our spring framework uses an expression parser to get content from the console with little validation. We are also really on an outdated API for input handling.

**APIs:** Secure API interactions are already a concern for our application. It would be smart to secure the ones currently in our project. Focusing on fixing the Spring Framework should be done before adding others, like the RESTful API.

**Cryptography:** Since our user data will be shared with outside APIs this data needs to be encrypted. Even if a breach happens because of another API, our software would still be held responsible for any damages.

**Code Error:** As input validation is added we will need to add logic for error handing. This means making sure users can't bypass errors.

## Code Review Summary

In the GreetingController class we see how the program parses information.

A screen shot of a computer program

Description automatically generated

In pom.xml we can see the current version of our spring framework. We should compare this information with the current release using spring’s online documentation.

**A screen shot of a computer

Description automatically generated**

## Mitigation Plan

The following steps should be taken to improve security:

* Update the spring framework API to version 6.2. This will remedy known vulnerabilities in previous versions. It is good practice to keep any APIs updated to their latest version.
* Create input validation. The software should default to denial when validating inputs.
* Add a central point for validation/error handling to GreetingController.java. It's important that these checks happen at a single point.