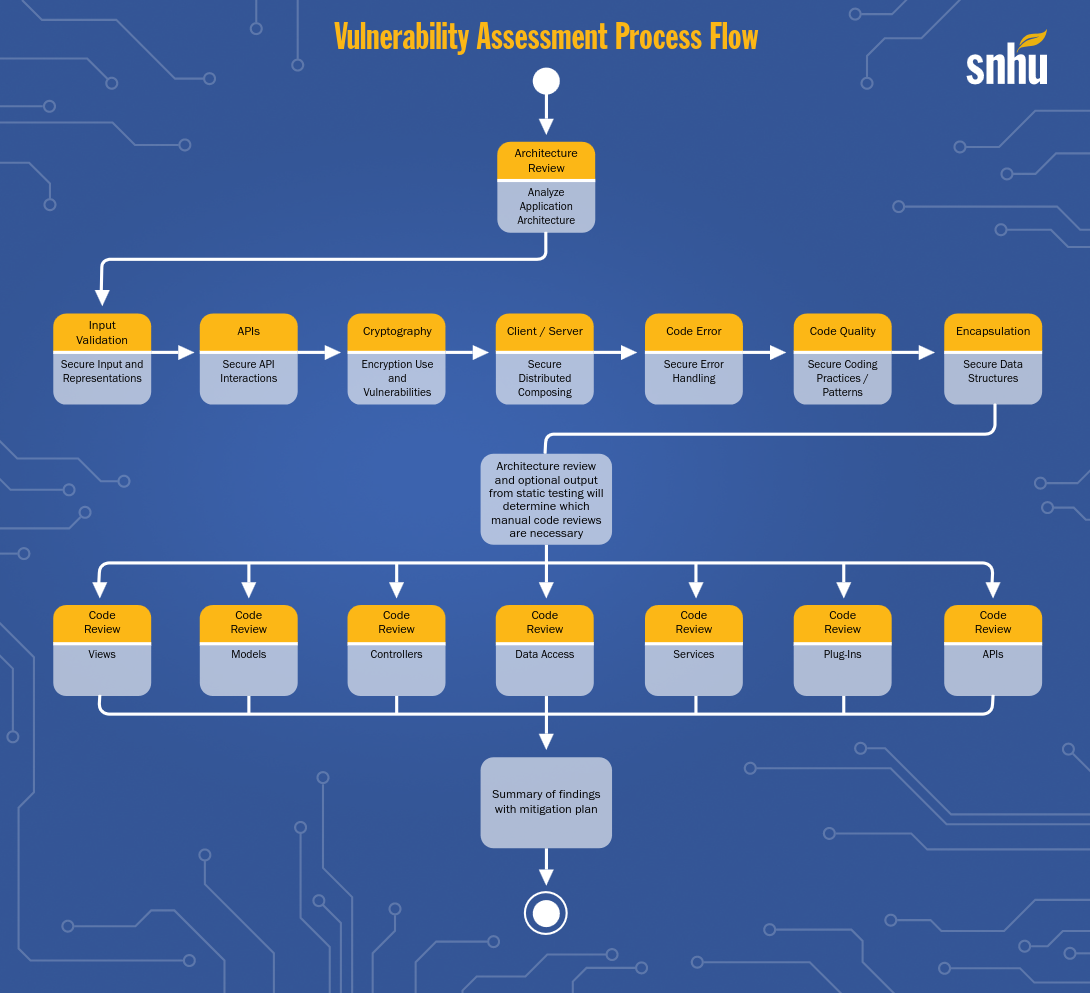
# Module Two Written Assignment Template

**Scenario**

You’re a senior software developer in a team of software developers. You’re responsible for a complex web application that uses Spring Framework. The team has been tasked with implementing an expressive command input function for the application. You are told the team is currently using Version 2.6.5 of the spring-data-rest-webmvc in Spring Framework. You also want to use the Spring Expression Language to accomplish the task.



## Areas of Security

* Input validation
* APIs
* Code quality

## Areas of Security Justification

The application needs a secure means to verify input. The spring-data-rest-webmvc is a web API requiring a secure means of communication and data transfer. Both the API and Input validation security concerns involve code quality.

## Code Review Summary

During the code review I have identified the following possible vulnerabilities:

* GreetingController class is not marked final. This leaves the class open to a malicious subclass adding finalizers, cloning, and overriding random methods.
* Input validation is not implemented. Input should be validated to prevent denial of service attacks and similar malfeasance. In particular the scan data in GreetingController is attached directly to a string and added a clear text format.
* Consider declaring any classes that are not defined in the API as package-private.

## Mitigation Plan

I recommend the following mitigation activities to address these concerns:

* GreetingController class should be marked final
* Validation should be implemented for API endpoints. @NotNull @Valid and other annotations should be used to restrict the values that can be provided. In particular the length of the string should be restricted.
* Review classes to determine if public access is required or if a more restrictive level is sufficient.