MSSE672 – Component Based Software

Student: John Michael Kreski

Instructor: Mohammad Abu Matar

Assignment: Week 8 - JUnit Test Execution – Spring’s IoC / DI

Date: 08/24/2025

File Name: HWUnitTestExecution.doc

## Summary

This week’s programming assignment focused on refactoring and validating the use of Spring’s Dependency Injection (DI) and Inversion of Control (IoC) throughout the application. No new business logic or services were introduced, and no modifications were made to existing logic that would affect functionality or require additional test coverage.

As such:

* **No new unit tests were created.**
* All previously implemented tests remain relevant and continue to pass.
* The structure and logic of the application were preserved; only the project organization, annotations, and DI-related configurations were refined.

This decision aligns with the scope of the Week 8 objective: to demonstrate effective use of Spring’s IoC container and DI mechanisms, rather than introducing new functional logic requiring additional test coverage.

**Notes**

* Full src/test/java directory is included in the submission archive to provide continuity across weeks.
* This week’s submission focused on **Spring-based Dependency Injection (DI)**
* All previously submitted JUnit tests (e.g., for controller, service, and repository layers) remain valid and continue to pass. These tests were documented in prior submissions (Weeks 1–7).
* The full source code, including test classes, configuration files, and documentation, is available in the Git repository accompanying this project.
* Key updates this week include:
  + Finalized Spring @Component and @Autowired annotations across layers
  + Centralized socket authentication logic via a OncePerRequestFilter
  + Cleaned and documented file structure (see README.md and docs.md)