**Why I Chose These Security Tools**

1. Detekt with Custom Security Rules

I chose Detekt because it will scan our Kotlin code and catch security issues before they end up in our final app.

This will help our app by:

* Finding potential security vulnerabilities like hardcoded credentials or insecure random numbers
* Enforcing our custom security rules consistently across our entire codebase
* Making our code more maintainable and secure from the start

2. OWASP Dependency Check

I selected OWASP Dependency Check because it will protect our app from vulnerabilities in third-party libraries.

This will benefit our app by:

* Automatically scanning all our dependencies for known security issues
* Alerting us immediately if we add a library with security flaws
* Helping us justify our dependency choices based on security considerations

3. MobSF (Mobile Security Framework)

I included MobSF because it will provide mobile-specific security testing for our app.

This will improve our app by:

* Testing for Android-specific security issues like insecure data storage
* Identifying potential privacy leaks in our final APK
* Ensuring our app's permissions and communication channels are properly secured

4. SonarQube

I implemented SonarQube because it will give us a comprehensive security dashboard for our app.

This will enhance our app development by:

* Providing metrics and trends showing our security posture over time
* Combining results from our other security tools in one place
* Helping us demonstrate the security quality of our app to stakeholders

By implementing these tools, our app will be protected from multiple angles, making it much harder for attackers to find and exploit vulnerabilities. This comprehensive approach will give our users the secure experience they deserve and help us meet the requirements for the secure programming minor.