# AutoSecure Agent – High Level Design (Enhanced)

## Overview

AutoSecure Agent is an AI-powered developer assistant designed to automate the detection and remediation of security vulnerabilities and deprecated code in Java-based enterprise applications.

It intelligently upgrades libraries, replaces insecure or outdated methods, and validates code integrity — all within the GitHub workflow. Designed with modern DevSecOps principles in mind, this agent empowers organizations like Danske Bank to maintain secure, up-to-date applications without manual overhead.

## Key Objectives

• Detect vulnerabilities (CVEs) and deprecated Java/Spring methods in existing projects   
• Suggest and apply secure, up-to-date alternatives using AI   
• Automatically test and validate code to prevent breakages   
• Open detailed Pull Requests summarizing changes and benefits   
• Seamlessly integrate with GitHub, CI/CD, and enterprise policies

## Architecture Overview

The AutoSecure Agent is composed of modular, pluggable services:

1. \*\*GitHub Integration Module\*\* – Fetches source code and manages pull requests.  
2. \*\*Static Analyzer Engine\*\* – Scans for known vulnerabilities and deprecated APIs.  
3. \*\*AI Refactor Assistant\*\* – Uses GPT-4 to recommend safe code replacements.  
4. \*\*Refactoring Engine\*\* – Applies LLM recommendations using JavaParser or Spoon.  
5. \*\*Validator Engine\*\* – Builds and tests code using Maven/Gradle.  
6. \*\*PR Automation\*\* – Creates rich pull requests with fix summaries and logs.  
7. \*\*UI/CLI Interface\*\* (Optional) – Allows developers to trigger, preview, and validate fixes manually.

## Technology Stack

| Component | Technology |  
|------------------------|----------------------------------|  
| Language | Java, Python |  
| AI Backend | OpenAI GPT-4 API |  
| Code Parsing | JavaParser, Spoon |  
| Vulnerability Scanner | OWASP Dependency Check, Snyk |  
| Build Tools | Maven, Gradle |  
| Testing Framework | JUnit, TestNG |  
| Git Integration | GitHub API, GitHub CLI, Actions |  
| Interface | React (Web UI), Node.js (CLI) |

## Data Flow

1. \*\*Trigger\*\*: Repo is input via UI or CLI, or by GitHub webhook  
2. \*\*Scan\*\*: Static Analyzer identifies outdated code and vulnerabilities  
3. \*\*Analyze\*\*: AI Assistant evaluates unsafe patterns and provides migration paths  
4. \*\*Refactor\*\*: Refactoring Engine safely updates code  
5. \*\*Validate\*\*: Maven/Gradle build & unit tests confirm stability  
6. \*\*PR Create\*\*: Auto-generated PR includes summary, diffs, and changelog

## Security Considerations

• GitHub access is restricted via scoped tokens or GitHub Apps   
• Code is never sent to LLMs in entirety — only necessary snippets   
• Logs and rollback options maintained for every change   
• Agent does not execute external scripts, sandboxed runtime enforced

## Implementation Roadmap

\*\*Phase 1: MVP (Hackathon)\*\*   
- Detect CVEs and deprecated code   
- Generate and apply AI-based fix suggestions   
- Submit Pull Request with changelog

\*\*Phase 2: Enterprise-Ready Integration\*\*   
- Jenkins/GitHub CI integration   
- Team approval workflows

\*\*Phase 3: Extensibility & Language Support\*\*   
- Node.js, Python language support   
- Plugin ecosystem for custom security rules

## Value Proposition

• Save developer time by automating routine security upgrades   
• Reduce risk by quickly remediating vulnerabilities   
• Improve code quality and future-proof enterprise applications   
• Ensure compliance and auditability via automated logs and PRs   
• Empower secure software delivery across teams and pipelines

## Conclusion

AutoSecure Agent brings intelligence, automation, and security into the developer workflow. By acting as a Copilot for secure upgrades and refactors, it reduces manual effort, improves quality, and builds trust in every release.

In the context of Danske Bank and similar institutions, AutoSecure Agent strengthens application security posture, drives engineering efficiency, and paves the way for continuous, compliant modernization.