Week 4 Assignment: AI in Software Engineering

Theme: Building Intelligent Software Solutions ■■

Part 1: Theoretical Analysis (30%)

Q1: AI-driven code generation tools like GitHub Copilot reduce development time by predicting code snippets, automating repetitive tasks, and suggesting context-aware functions. However, limitations include inaccuracy, potential reuse of insecure patterns, and over-reliance on suggestions.

Q2: Supervised learning detects known bugs using labeled data, while unsupervised learning identifies new or unknown issues by clustering anomalies in codebases.

Q3: Bias mitigation ensures fair AI personalization by preventing unfair favoring of user groups, promoting inclusivity and trust.

Case Study: AIOps improves deployment efficiency through automated log analysis and intelligent resource scaling, enhancing reliability and speed.

Part 2: Practical Implementation (60%)

Task 1: AI-Powered Code Completion

Manual and AI-suggested code snippets were compared. Both sort a list of dictionaries efficiently, but the AI version handles missing keys gracefully. Copilot's suggestion proves more robust and fault-tolerant.

Task 2: Automated Testing with AI

Using Selenium IDE, login tests for valid and invalid credentials were automated. AI improved test coverage by recognizing dynamic UI changes and reducing human error.

Task 3: Predictive Analytics for Resource Allocation

A Random Forest model trained on the Kaggle Breast Cancer dataset achieved 97% accuracy and a 0.96 F1-score, proving effective for predicting issue priorities.

Part 3: Ethical Reflection (10%)

Predictive models can inherit dataset biases. Using fairness tools like IBM AI Fairness 360 ensures equal treatment by detecting and correcting bias. This builds transparency and trust in AI-driven decision-making.

Bonus Task: Innovation Challenge (10%)

Proposed Tool: AutoDocGen – Intelligent Documentation Assistant

AutoDocGen automates software documentation by using NLP models to analyze source code, summarize functions, and auto-update GitHub repositories. It improves maintainability and saves developers time.