Week 4 Assignment: AI in Software Engineering

Theme: Building Intelligent Software Solutions

Part 1 - Theoretical Analysis (30%)

Q1: AI-driven code generation tools

AI code-completion tools like GitHub Copilot reduce development time by suggesting code snippets based on context. They allow developers to focus on logic rather than boilerplate, saving keystrokes and reducing context switching. Limitations include poor handling of context boundaries, potential legal/IP issues, and sometimes generating incorrect or insecure code.

Q2: Supervised vs Unsupervised Bug Detection

Supervised learning uses labeled bug data, achieving high precision on known bugs. Unsupervised learning detects anomalies in logs or code structure without labeled data, identifying new or unseen bugs. Supervised is best for regression detection; unsupervised excels in anomaly-rich environments with sparse labels.

Q3: Bias Mitigation in UX

AI in user experience personalization can amplify biases if certain user groups are underrepresented. Bias mitigation ensures fairness and prevents feedback loops that exclude minority users. Tools like IBM AI Fairness 360 help detect and correct such biases.

Case Study: AIOps in DevOps

AIOps improves deployment efficiency by automating anomaly detection and dynamic resource allocation. Example 1: Predictive gating stops flaky builds. Example 2: Smart Kubernetes scaling improves CI/CD speed and stability.

Part 2 - Practical Implementation (60%)

Task 1: Code Completion

Compared a manual sort_dicts function with Copilot-generated version. Copilot used itemgetter for speed. Manual code was more readable but less optimized. Copilot saved time but needed review.

Task 2: Automated Testing

Used Selenium IDE with Testim to automate login tests. AI-based selectors adapted to DOM changes. Captured results showed higher resilience and reduced flakiness compared to manual scripts.

Task 3: Predictive Analytics

Used Breast Cancer dataset to predict issue priority. Preprocessed data and trained a RandomForestClassifier.

Week 4 Assignment: AI in Software Engineering

Theme: Building Intelligent Software Solutions

Achieved accuracy ~94% and F1-score ~93% on test set. Notebook contains code and evaluation metrics.

Part 3 - Ethical Reflection (10%)

The model may inherit dataset biases, leading to unfair priority assignments. Underrepresented teams may suffer. IBM AI Fairness 360 can assess and mitigate such bias using equal opportunity and reweighing techniques.

Bonus - Innovation Challenge

AutoDoc-GPT: A commit hook using LLMs to generate documentation from code diffs. Saves developer time, improves documentation quality, and integrates with CI pipelines. Impact: Higher traceability and reduced tech debt.