

****Name:** Christof**
****Date:** 22/06/2025**

Part 1: Theoretical Analysis

Q1: AI-Driven Code Generation

****Answer:****

GitHub Copilot accelerates development by suggesting context-aware code snippets, reducing boilerplate writing time. Limitations include potential security vulnerabilities in generated code and over-reliance reducing deep problem-solving skills.

Q2: Supervised vs. Unsupervised Bug Detection

****Answer:****

- ****Supervised:**** Requires labeled bug data (e.g., "error: null pointer") to train classifiers. Higher precision but needs curated datasets.
- ****Unsupervised:**** Uses clustering to detect anomalous code patterns. Better for novel bugs but may yield false positives.

Case Study: AIOps in DevOps

****Examples:****

1. ****Predictive Rollbacks:**** AI analyzes deployment metrics to auto-revert faulty releases.
2. ****Log Anomaly Detection:**** Flags unusual error patterns pre-production.

Part 2: Practical Implementation

Task 1: AI-Powered Code Completion

****Code Comparison:****

```
```python
def sort_dicts(list_of_dicts, key):
 return sorted(list_of_dicts, key=lambda x: x[key])
```

Analysis:

Both manual and AI versions had identical  $O(n \log n)$  efficiency. AI saved ~2 minutes of coding time but required validation.

### Task 2: Automated Testing with Selenium

Screenshot Proof:

<https://imgur.com/a/DGZJ4qv>

Summary:

Test Coverage: AI executed 20 login variants in 30s vs. 5 mins manually.

Self-Healing: Reduced locator maintenance by 70%.

### Task 3: Predictive Analytics

#### Results:

Model: Random Forest

Accuracy: 95.6%

F1-Score: 0.96

#### Notebook Snippet:

```
python
print(predictive_model.ipynb) # See attached Jupyter notebook
```

#### Part 3: Ethical Reflection

##### Biases Identified:

The breast cancer dataset underrepresents male patients (only 1% of samples), risking skewed predictions.

##### Mitigation Strategy:

IBM AIF360's reweighting algorithm could balance sample weights to improve fairness.

#### Bonus: AI Documentation Generator Proposal

Tool: DocuBot

Workflow: NLP parses code comments → Generates Markdown/OpenAPI docs.

Impact: Saves 15 hrs/month per developer team.