

## PART ONE. TASK 1: THEORETICAL QUESTIONS

**Q1: How do AI-driven code generation tools reduce development time, and what are their limitations?**

### Benefits:

- **Speedy boilerplate creation** – Tools like GitHub Copilot can quickly generate common code structures like API endpoints or tests.
- **Reduced cognitive load** – Developers can focus on logic and architecture while the tool handles repetition.
- **In-context suggestions** – Real-time help boosts productivity and accelerates learning.

### Limitations:

- **Contextual gaps** – Generated code may not fully align with your specific logic or requirements.
- **Overreliance** – Can lead to blindly accepting insecure or inefficient code.
- **Licensing issues** – Some generated code might originate from sources with restrictive licenses.
- **Weak at architecture** – It assists with syntax, not high-level system design or architectural decisions.

---

**Q2: Compare supervised and unsupervised learning in the context of automated bug detection.**

| Feature       | Supervised Learning                    | Unsupervised Learning                 |
|---------------|--|---------------------------------------|
| Training Data | Labeled examples (buggy vs clean code) | Unlabeled code                        |
| Approach      | Classification or regression           | Clustering, anomaly detection         |
| Use Cases     | Detecting known bugs                   | Spotting unusual behavior or new bugs |

|            |  |  |
|------------|--|--|
| Advantages | High accuracy on known problems              | Discover hidden patterns, handles novel bugs |
| Drawbacks  | Needs lots of labeled data, weak on unknowns | May flag false positives, lacks clear labels |

---

### Q3: Why is bias mitigation critical when using AI for user experience personalization?

- **Fairness** – Without mitigation, certain groups may be unfairly favored or excluded.
  - **Inclusivity** – Prevents cultural, gender, or accessibility biases from affecting experience quality.
  - **Trust and reputation** – Biased experiences erode user trust and damage brand reputation.
  - **Legal compliance** – Bias in personalization may violate anti-discrimination laws or ethical standards.
- 

## 2. Case Study Analysis

### Article: "AI in DevOps: Automating Deployment Pipelines"

**Q: How does AIOps improve software deployment efficiency? Provide two examples.**

#### 1. Predictive Failure Detection

AIOps tools analyze metrics and logs to detect early signs of failure (e.g., memory spikes, slow queries). This allows for proactive intervention before users are affected.

#### 2. Automated Rollbacks

If a deployment causes issues, AIOps systems can detect anomalies and automatically revert to the last stable version—reducing downtime and human error.

#### Summary:

AIOps makes DevOps smarter by enabling **real-time insights**, **self-healing systems**, and **faster, more reliable deployments**.

## PART TWO. TASK 2:

### SELENIUM IDE CHROME EXTENSION TESTING

**Website used:** <https://practicetestautomation.com/practice-test-login/>

I used Selenium IDE to automate login testing for valid and invalid credentials on a public test website. The test cases included inputting known credentials, submitting the login form, and verifying the presence of either a success message or an error message.

AI plugins for Selenium IDE (or AI-driven test tools like Testim.io) greatly enhance traditional test automation. They can automatically detect changes in the UI, such as altered button IDs or dynamic layouts, and adapt the tests without human intervention. This “self-healing” capability reduces the need to constantly update test scripts as the application evolves.

AI also suggests test cases by analyzing usage data or inspecting patterns, thus improving test coverage—something that’s often missed in manual testing. Overall, AI-driven automation leads to more reliable, scalable, and maintainable testing workflows, especially in fast-paced agile development environments.

#### I. Valid Credentials Test

# Logged In Successfully

**Congratulations student. You successfully logged in!**

Log out



# Test login

This is a simple Login page. Students can use this page to practice writing simple positive and negative Login tests. Login functionality is something that most of the test automation engineers need to automate.

Use next credentials to execute Login:

Username: **student**

Password: **Password123**

Username

Password



• Selenium IDE is recording...

## II. Invalid Credentials Test

# Test login

This is a simple Login page. Students can use this page to practice writing simple positive and negative Login tests. Login functionality is something that most of the test automation engineers need to automate.

Use next credentials to execute Login:

Username: **student**

Password: **Password123**

Username

wronguser

Password

.....

Submit

Your username is invalid!

### Test case 1: Positive Login test

1. Open page
2. Type username **student** into Username field
3. Type password **Password123** into Password field



• Selenium IDE is recording...

## **PART THREE. Ethical Reflection.**

When deploying the predictive model from Task 3 in a real company setting, several ethical concerns emerge—especially around bias and fairness.

### **1. Potential Biases in the Dataset:**

Although we used a medical dataset for simulation, in a real-world company scenario, the data used to predict issue priority could be biased. For example:

- Underrepresented teams or departments might have fewer logged issues, making the model less accurate for them.
- Historical labels for “priority” may reflect managerial bias (e.g., prioritizing issues from senior staff or certain departments).
- Features such as time of submission, reporter role, or project type could introduce systemic bias if they correlate with team identity or status.

### **2. Fairness Tools – IBM AI Fairness 360:**

To mitigate these risks, tools like IBM AI Fairness 360 (AIF360) can be used to:

- Audit the model for disparate impact across teams, roles, or departments.
- Apply fairness metrics (e.g., equal opportunity, disparate impact ratio).
- Rebalance or reweigh data using algorithms like Reweighting or Disparate Impact Remover.
- Continuously monitor fairness in production with explainability modules.

### **Conclusion:**

Ethical AI deployment means going beyond accuracy. It requires tools, policies, and vigilance to ensure the model doesn’t unintentionally reinforce existing inequalities.

## **PART FOUR. BONUS PROPOSAL.**

### **Code Whisperer DX – AI-Powered Developer Experience Optimizer**

---

#### **Purpose**

Code Whisperer DX is an AI-powered tool that analyzes not just the logic of a codebase, but the emotional and cognitive impact it has on developers. It identifies areas that are overly complex, hard to maintain, or mentally taxing—then offers improvements that enhance developer experience (DX). Think of it as Grammarly meets therapy, but for your source code.

---

#### **Workflow**

##### **1. Ingest Codebase**

- Parses repositories including commit history, code comments, and structure.
- Scores functions, components, and files based on readability, churn, and clarity.




##### **2. Developer Sentiment Integration**

- Gathers optional, anonymous developer feedback on code complexity and stress levels.
- Tracks developer frustration signals from IDEs (e.g., hesitation, frequent undo, debug loops).

##### **3. DX Insights Engine (AI Core)**

- Uses NLP and code embeddings to detect:
  - Mental fatigue hotspots
  - Overengineered logic
  - Vague or misleading naming

##### **4. Recommendations**

- Suggests refactors that improve clarity and reduce mental load.
  - Flags problem areas with emoji-coded DX alerts:  
 Too Clever    Needs Cleanup    Elegant Simplicity
- 

## Impact

- Reduces burnout by identifying stress-inducing code patterns.
  - Improves long-term maintainability and onboarding.
  - Enhances team morale by aligning code quality with human experience.
- 

## Tagline:

*“Because writing code is human. So fixing it should be too.”*