

## Participant Course Summary

**Intelligent Engineering with AI** is a two-day course that integrates AI tools with traditional Test-Driven Development (TDD) practices. By leveraging AI technologies, this course will enhance software engineers' productivity and code quality.

### Course Objectives

- Understand the fundamentals of Test-Driven Development (TDD).
  - Learn how to use AI coding assistants like GitHub Copilot and ChatGPT.
  - Develop skills for writing, generating, and refactoring code using AI tools.
  - Master paired programming with AI for enhanced collaboration.
  - Address issues associated with working in legacy code.
- 

### Key Topics Covered

#### 1. Introduction to TDD and AI Tools

- Basics of TDD and its cycle: Red, Green, Refactor.
- Hands-on exercises with GitHub Copilot and ChatGPT.
- Practical pair programming exercises.

#### 2. AI-Driven Testing and Code Quality

- Automating test generation and execution with AI.
- Using AI for code refactoring and optimization.
- Documenting code using AI tools.

#### 3. Advanced TDD Practices

- Working with mocks, stubs, fakes, and test doubles.
- Techniques for building testable code and refactoring legacy code.
- Applying design patterns and recognizing code smells with AI assistance.

#### 4. Software Craftsmanship and Extreme Programming (XP)

- SOLID principles and XP practices.
- Clean code concepts and refactoring techniques.
- Practical exercises like the Fizz Buzz Kata and Bowling Kata.

#### 5. Building and Deploying Applications

- Developing a simple API application.
- Setting up continuous integration and deployment using GitHub Actions.
- Hands-on exercises in pair programming.

## Learning Outcomes

### Participants will be able to:

- Write clean, maintainable code with the help of AI tools.
- Understand and document code effectively using AI.
- Implement TDD practices to improve software quality.
- Utilize AI for code generation, refactoring, and testing.
- Collaborate efficiently using paired programming techniques.

### Participants will become familiar with:

- GitHub Copilots and ChatGPT.
  - Write tests first and use multiple paired testing techniques.
  - Modifying software design using a test-first approach.
- 

## Course Schedule

### Day 1: Introduction to TDD and AI Tools

#### *Morning Session:*

1. Welcome and Introductions
2. Introducing Tyler Morgan, the Intelligent Engineering Course Assistant
3. Course Objectives and Structure
4. Introduction to TDD with Fizz Buzz Kata
5. Practical Exercise: Fizz Buzz Kata (Parse String Time) with TDD and AI Tools
6. Software Craftsmanship and Extreme Programming
7. Wrap-Up and Q&A

#### *Afternoon Session:*

1. Overview of AI Tools and Current Capabilities
2. Bowling Kata
3. Refactoring Techniques and Clean Code Concepts
4. Introduction to the Roman Numeral Calculator Kata
5. Practical Exercise: Roman Numeral Calculator Kata with TDD and AI Tools
6. Wrap-Up and Q&A

## Intelligent Engineering with AI

### Day 2: Advanced TDD and Software Craftsmanship with AI Integration

#### *Morning Session:*

1. Recap and Introduction
2. Gilded Rose Kata with TDD and AI Tools
3. Code Smells and Design Patterns
4. Wrap-Up and Q&A

#### *Afternoon Session:*

1. Introduction to Prompt Engineering
  2. Stubs, Mocks, Fakes, and Doubles
  3. Building a Simple API Application
  4. Deploying the Application using GitHub Actions
  5. Wrap-Up and Q&A
- 

## Conclusion

This course comprehensively introduces integrating AI tools with traditional software engineering practices, focusing on TDD. By the end of the course, participants will be equipped with the knowledge and skills to enhance their coding practices and productivity using AI technologies.