Field Guide to AI-Driven Refactoring Strategies

Author: Alex Bunardzic – *AI & Refactoring Strategist*

A concise, 10-page PDF covering test-driven navigation, mutation testing, and AI pairing patterns. Practical strategies to accelerate and improve software quality Perfect for engineering leaders who need results in 90 days.

Page 1 — Cover Page

- Big title, your name + role
- Eye-catching AI + code refactoring visual
- Subheading: “Practical strategies to accelerate and improve software quality”

Page 2 — Introduction

- Purpose of the guide
- Why AI is transforming code refactoring
- High-level benefits

Page 3 — The Refactoring Mindset

- Principles of clean, maintainable code
- The importance of iterative improvement
- Role of AI as a “pair programmer”

Page 4 — Common Code Smells

- Definitions + examples (duplicated code, large classes, long methods, etc.)
- How AI can detect them

Page 5 — AI Tools Overview

- Current AI tools for refactoring
- Key capabilities and limitations
- Evaluation checklist

Page 6 — AI-Assisted Refactoring Process

- Step-by-step process integrating AI
- When to trust AI vs. manual review
- CI/CD integration tips

Page 7 — Case Study

- Before-and-after example of a codebase
- Metrics: complexity reduction, maintainability index improvement

---

Page 8 — Challenges & Risks

- Over-refactoring

- AI hallucinations and incorrect changes
- Security and licensing concerns

Page 9 — Best Practices

- Guidelines for safe AI refactoring
- Version control strategies
- Team training and adoption tips

Page 10 — Closing & Resources

- Final takeaway
- Recommended books, blogs, tools
- Contact information
