Contents

RaimaAI: The World's First Autonomous Functional Safety (FuSa) Engineer

Executive Summary

RaimaAI represents a paradigm shift in functional safety engineering, providing autonomous capabilities that accelerate safety-critical software development while ensuring compliance with the most stringent industry standards including DO-178C Level A and ISO 26262 ASIL D. By leveraging advanced AI technologies, RaimaAI automates the entire functional safety lifecycle from requirements to certification.

Core Capabilities

1. Automated Test Generation

Unit Test Generation

- **100% MC/DC Coverage**: Automatically generates test cases achieving Modified Condition/Decision Coverage
- **Boundary Value Analysis**: Identifies edge cases and corner conditions
- **Equivalence Partitioning**: Optimizes test suite efficiency
- **State Machine Testing**: Covers all state transitions and invalid state handling

Integration Test Generation

- **Interface Testing**: Validates all module interactions
- **Data Flow Testing**: Ensures proper data propagation across components
- **Timing Analysis**: Verifies real-time constraints
- **Fault Injection**: Tests error handling and recovery mechanisms

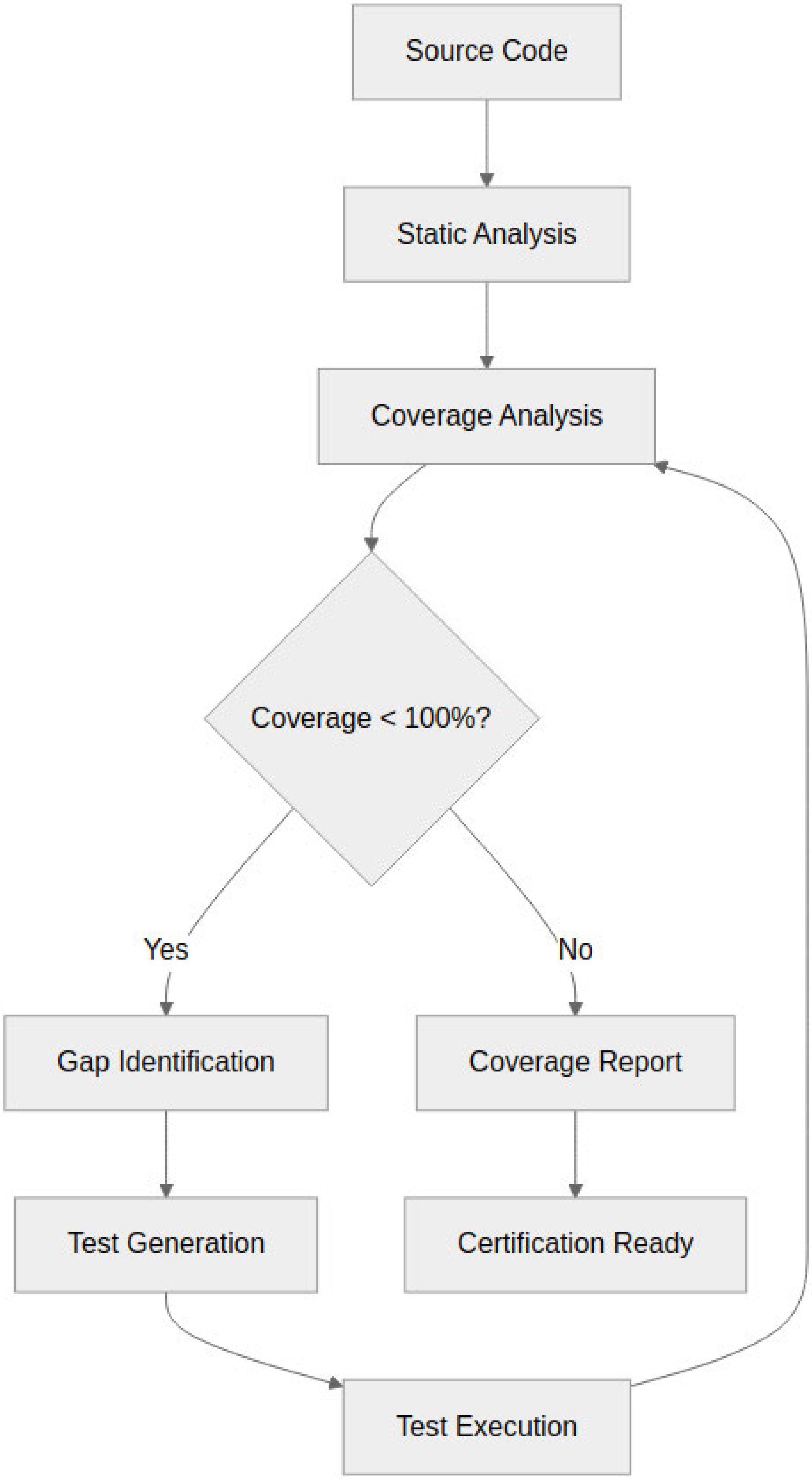

Figure 1: Diagram

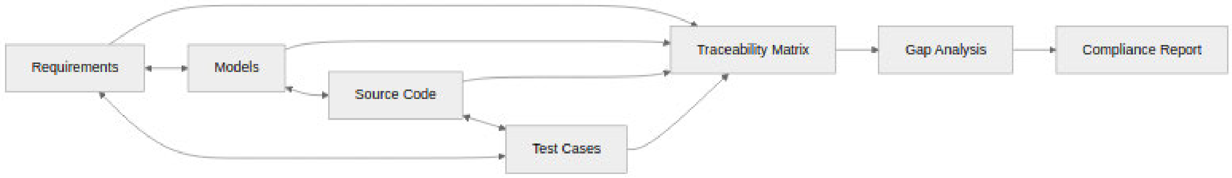

Figure 2: Diagram

2. Coverage Gap Analysis and Completion

3. Multi-Domain Traceability

4. Static Analysis Automation

MISRA 2025 Compliance

- Real-time code analysis during development
- Automated fix suggestions
- Deviation documentation
- Rule mapping to safety requirements

AUTOSAR C++14 Compliance

- Architecture conformance checking
- Coding guideline enforcement
- Memory safety validation
- Concurrency analysis

5. Git Workflow Integration

Automated Documentation Generation

1. Control Flow Analysis (CFA)

RaimaAI automatically: - Generates control flow graphs for all functions - Identifies cyclomatic complexity - Detects unreachable code - Validates decision coverage

2. Data Flow Analysis (DFA)

- Variable lifecycle tracking
- Use-definition chain analysis
- Data dependency graphs
- Information flow security analysis

3. Architecture Analysis

Architecture Metrics

- **Cohesion**: Measures module focus and single responsibility
- **Coupling**: Analyzes inter-module dependencies
- **Connectedness**: Evaluates component relationships

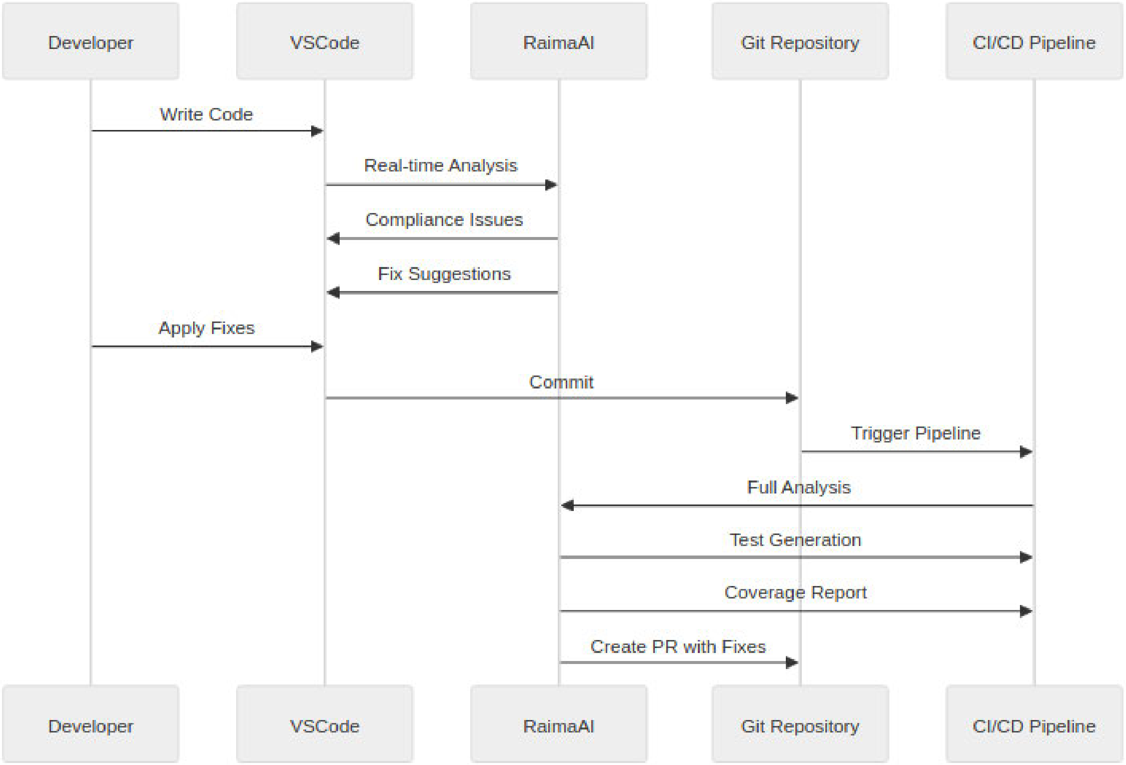

Figure 3: Diagram

- **Stability**: Assesses change impact and maintainability

Safety Analysis Automation

1. HARA (Hazard Analysis and Risk Assessment)

2. FMEA (Failure Mode and Effects Analysis)

Automated generation includes: - Component failure modes - Failure effects and causes - Detection methods - Risk Priority Numbers (RPN) - Mitigation strategies

3. Requirements and Design Documentation

System Requirements Specification (SRS)

- Functional requirements extraction
- Non-functional requirements identification
- Safety requirements derivation
- Traceability to hazards

Software Design Documentation (SDD)

- High-level architecture
- Module specifications

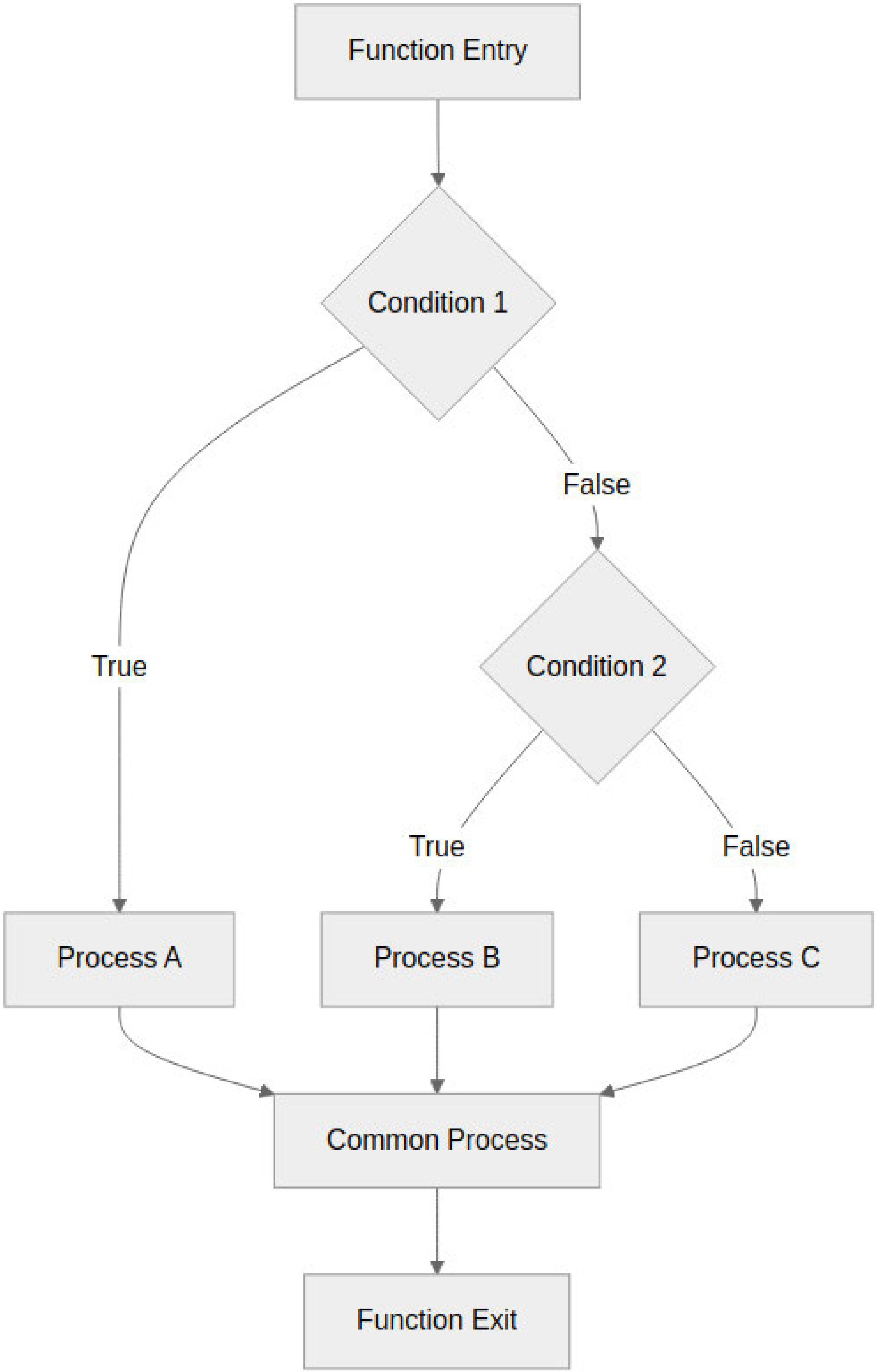

Figure 4: Diagram

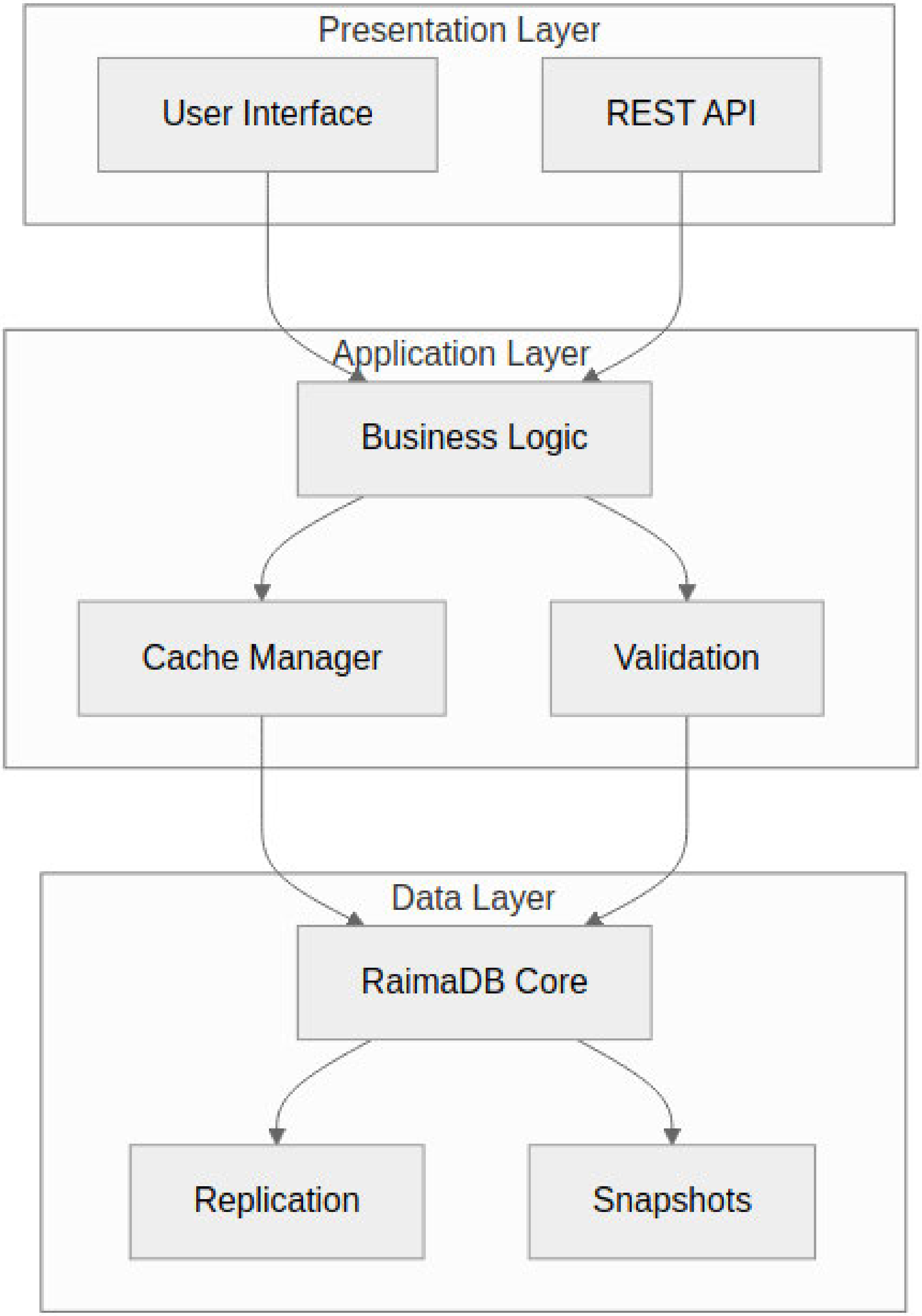

Figure 5: Diagram

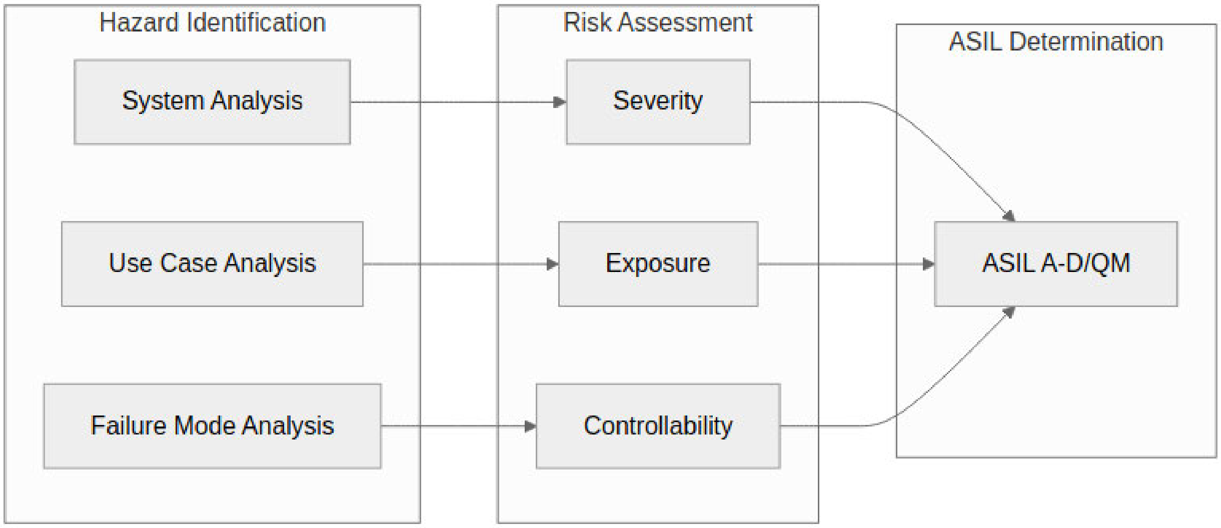

Figure 6: Diagram

- Interface definitions
- Design rationale with safety considerations

Time Zero Developer Experience

VSCode Integration

Workflow Benefits

1. **Immediate Feedback**: Issues detected as code is written
2. **Contextual Fixes**: AI understands code intent and suggests appropriate fixes
3. **Learning System**: Improves suggestions based on accepted/rejected fixes
4. **Seamless Integration**: Works within existing development workflows

Certification Support

DO-178C Level A Compliance

- **Automated Evidence Generation**
  - Test procedures and results
  - Coverage analysis reports
  - Traceability matrices
  - Review checklists
- **Verification Activities**
  - Requirements-based testing
  - Structural coverage analysis
  - Source code to object code analysis
  - Timing and stack analysis

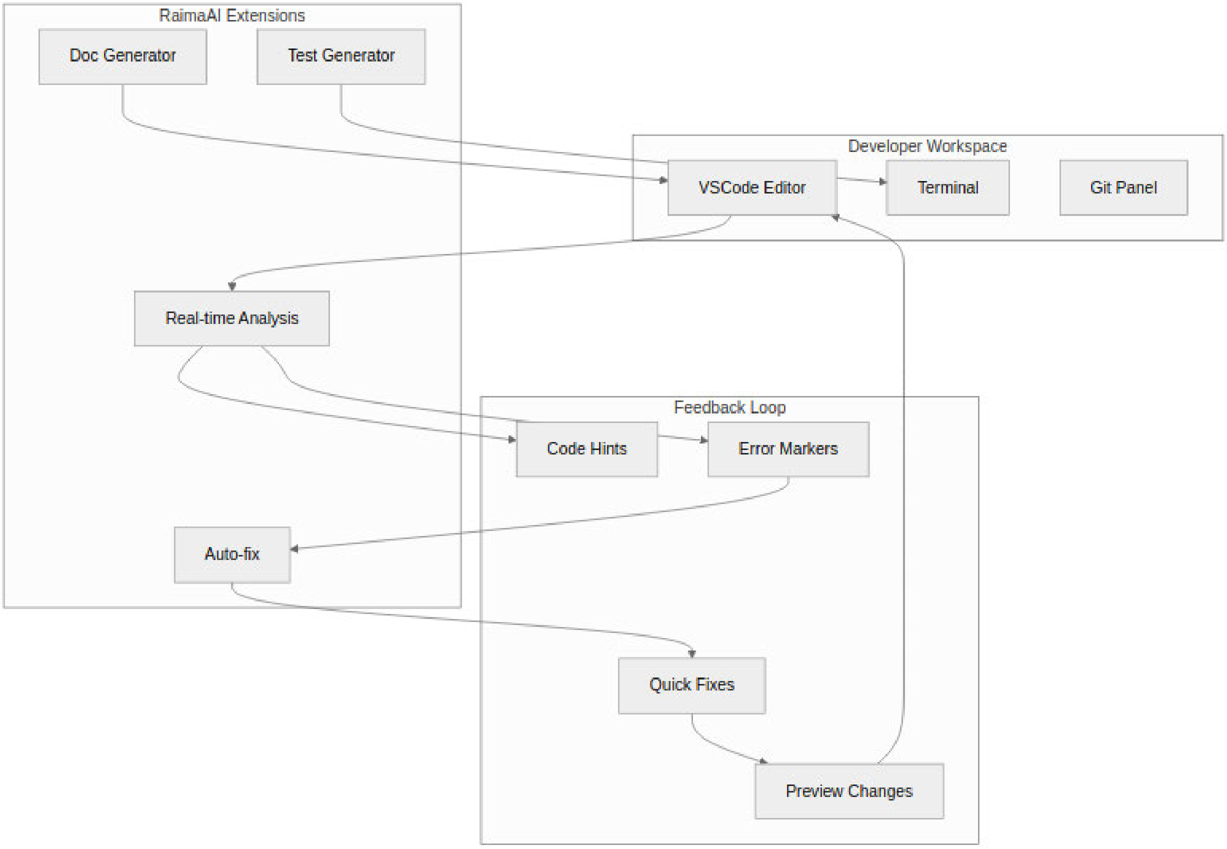

Figure 7: Diagram

ISO 26262 ASIL D Compliance

- **Work Product Generation**
  - Software safety requirements
  - Software architectural design
  - Unit/Integration test specifications
  - Verification reports
- **Tool Qualification**
  - Tool Confidence Level (TCL) assessment
  - Validation suite execution
  - Tool error detection capabilities
  - Qualification documentation

Implementation Architecture

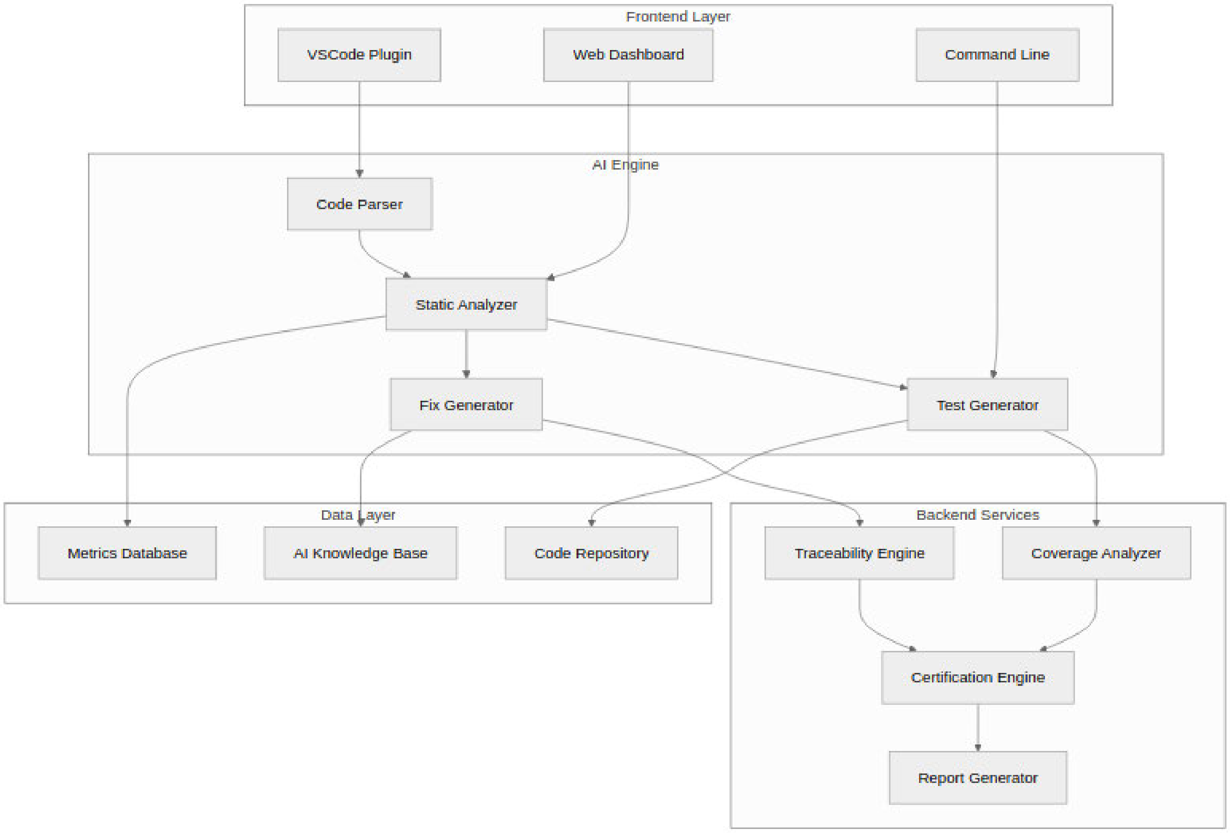

Figure 8: Diagram

Key Differentiators

1. Autonomous Operation

- Minimal human intervention required
- Self-improving through machine learning

- Proactive issue detection and resolution

2. Comprehensive Coverage

- Entire safety lifecycle support
- Multiple standard compliance
- All documentation automated

3. Developer-Centric Design

- Integrates into existing workflows
- Reduces cognitive load
- Accelerates development cycles

4. Certification-Ready Output

- Pre-validated against standards
- Auditor-friendly documentation
- Complete evidence packages

ROI and Benefits

[redacted]

[redacted]

[redacted]

[redacted]

[redacted]

[redacted]

[redacted]

[redacted]

[redacted]
- Automated threat modeling
- Predictive defect analysis

Phase 2: Extended Standard Support

- IEC 61508 SIL 3/4
- EN 50128 SIL 3/4
- IEC 62304 Class C

Phase 3: Full Lifecycle Automation

- Requirements generation from specifications
- Automated safety case construction
- Continuous certification maintenance

Conclusion

RaimaAI transforms functional safety engineering from a manual, error-prone process into an automated, reliable, and efficient workflow. By combining cutting-edge AI technology with deep domain expertise in safety-critical systems, RaimaAI enables organizations to achieve certification faster, maintain compliance continuously, and deliver safer products to market.

RaimaAI: Automating Safety, Accelerating Innovation