SDLC Fundamentals: Methodologies & Best Practices

Optimising Software Delivery through Structured Lifecycle Management

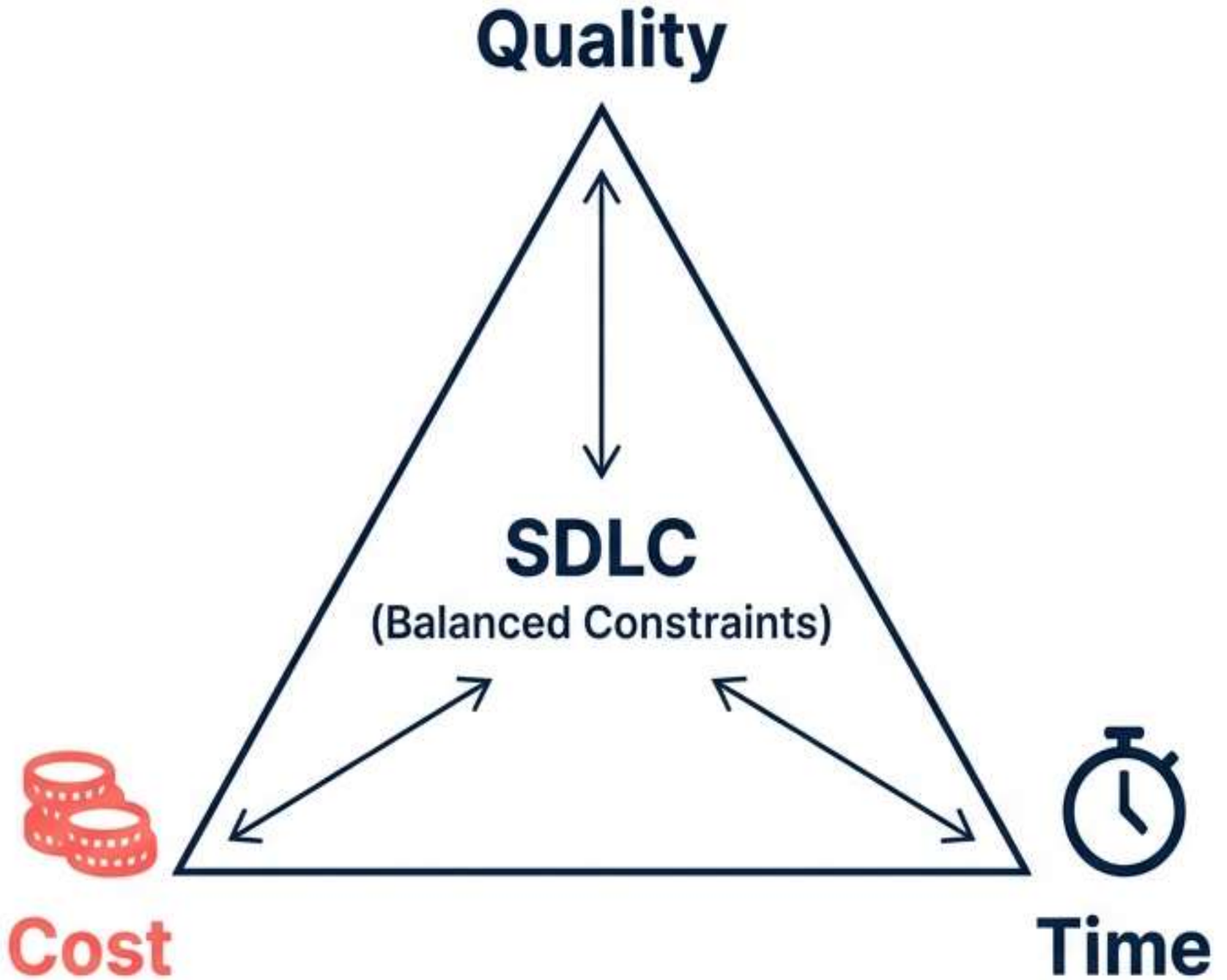
The Strategic Value of SDLC

Testleaf
Always Ahead

Core Definition

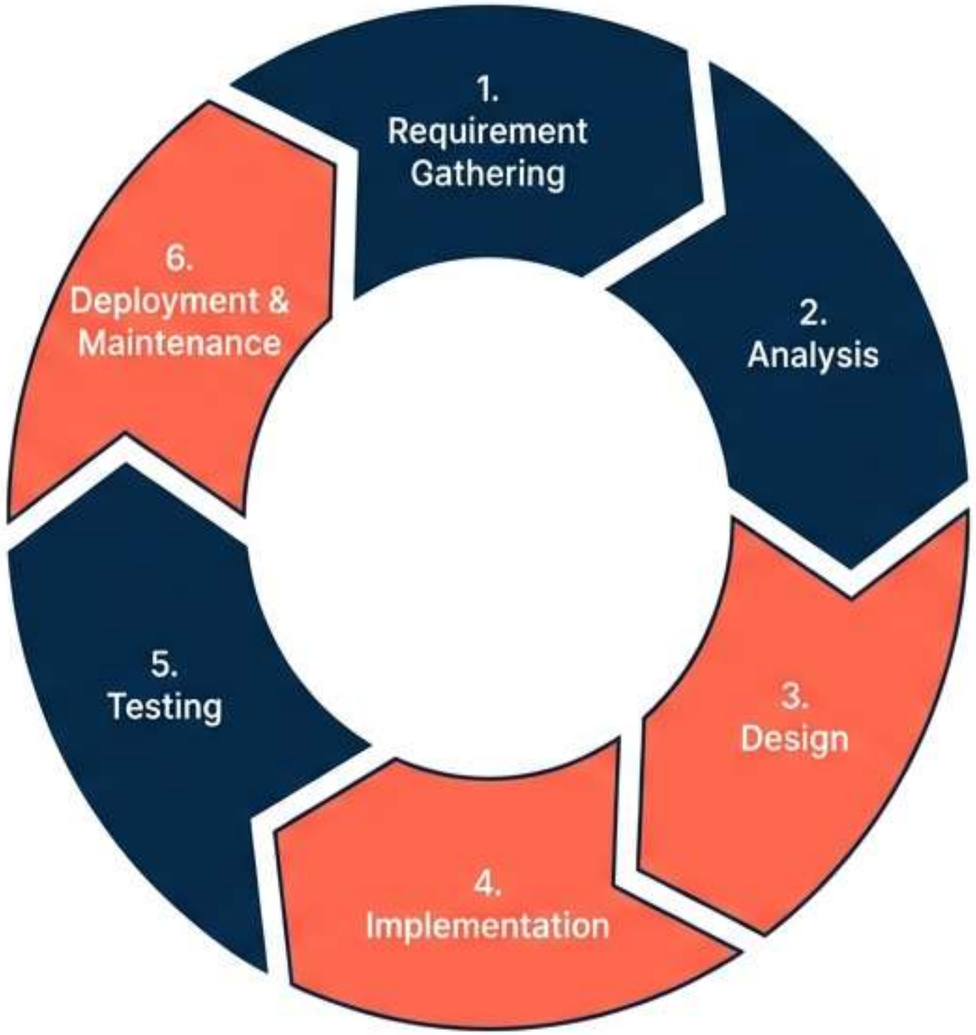
A systematic framework designed to produce high-quality software at the lowest cost in the shortest timeframe.

Why It Matters: Without SDLC, development is chaotic, unmeasurable, and prone to scope creep.



Testleaf
Always Ahead

The Six Pillars of the Lifecycle



Requirements Engineering: The Foundation

- The Foundation: This phase dictates the success of the entire project.
- Outcome: A signed-off SRS document serving as the single source of truth.

Design, Develop, Deliver

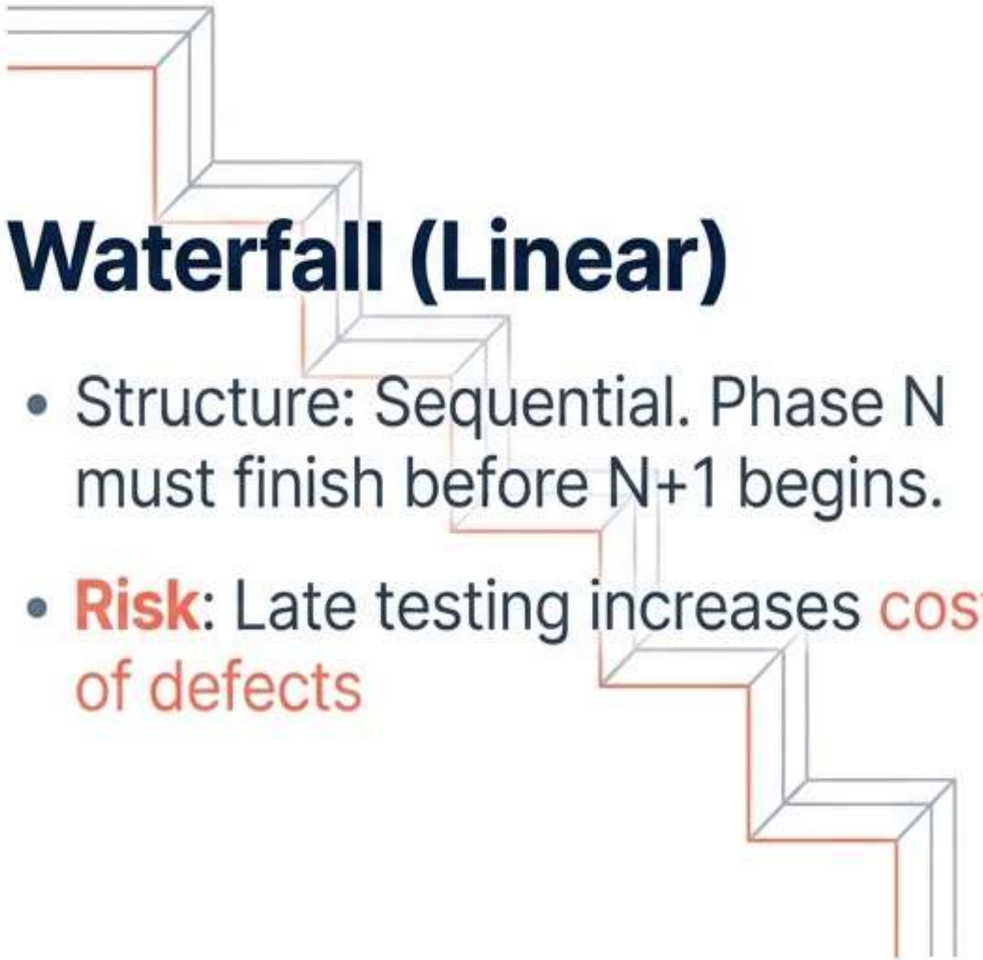
Design Phase (The Blueprint)

- **High-Level Design (HLD):** Macro-architecture, solution overview. Created by Solution Architects.
- **Low-Level Design (LLD):** Micro-logic, detailed module specs. Created by Senior Developers.

Development & Testing

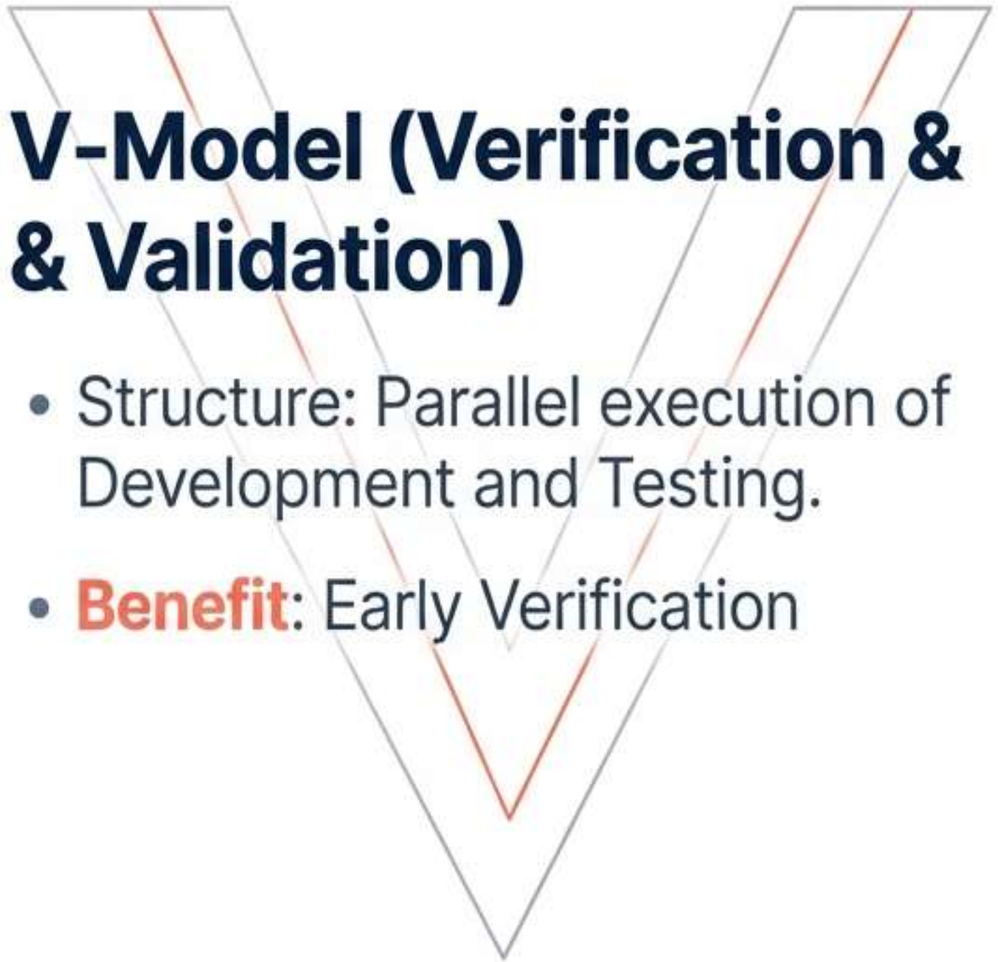
- **Development:** Transforming blueprints into executable code.
- **Testing:** Iterative bug finding and fixing loop (Dev <-> QA).

Comparative Methodologies: Waterfall vs. V-Model



Waterfall (Linear)

- Structure: Sequential. Phase N must finish before N+1 begins.
- **Risk**: Late testing increases cost of defects



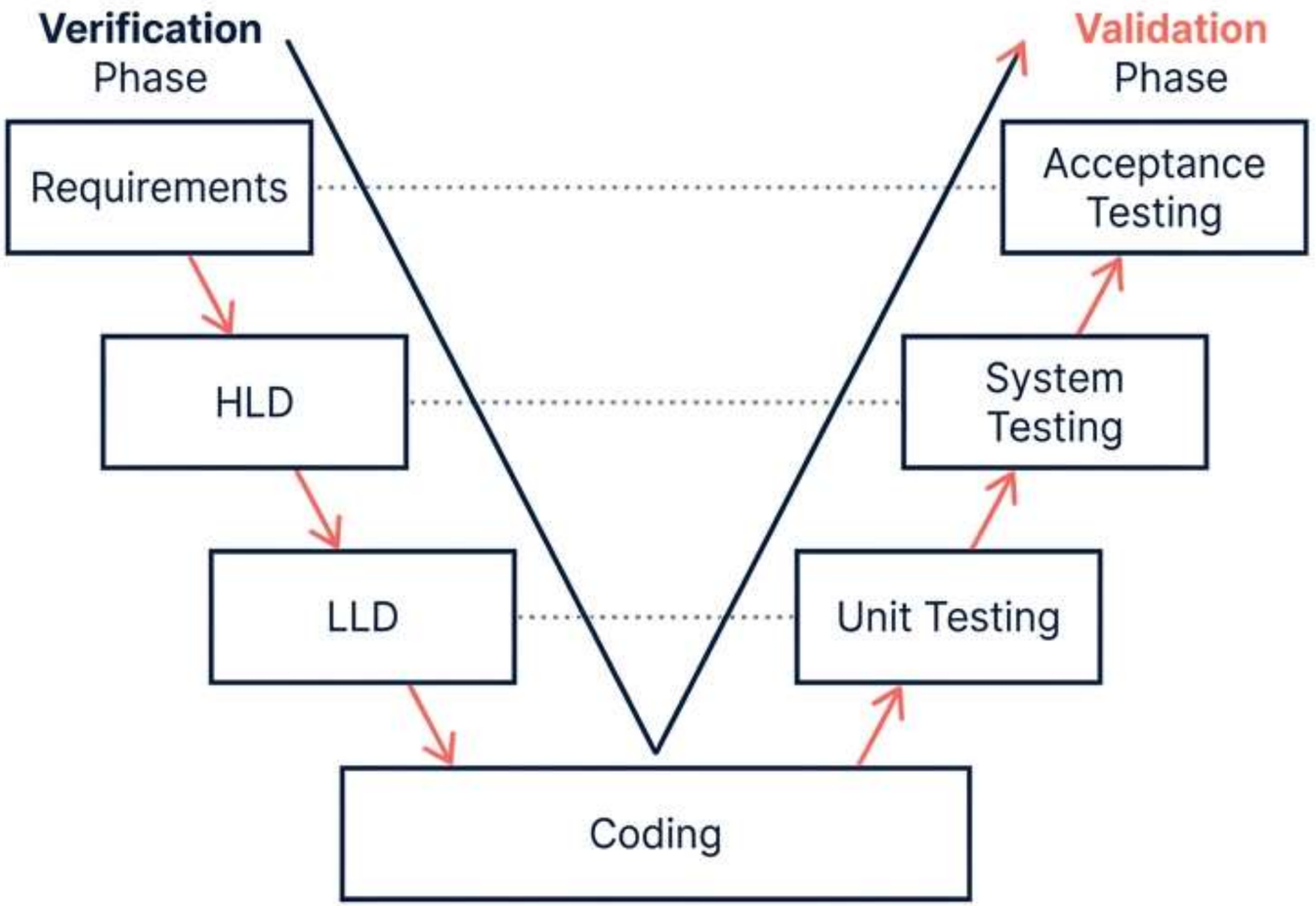
V-Model (Verification & & Validation)

- Structure: Parallel execution of Development and Testing.
- **Benefit**: Early Verification

Verification (Static):
Are we building the product right? (Reviews)

Validation (Dynamic):
Are we building the right product? (Execution)

The V-Model Strategy: Verification vs. Validation



Modern Testing Strategy: Shift Left & Shift Right

Testleaf
Always Ahead

Shift Left (Development)

Review Requirements

Unit Tests

Shift Right (Production)

A/B Testing

User Feedback

Benefit: Drastic reduction in bug-fix costs (Proactive).

Benefit: Validates user experience under real load (Real-World).

Executive Summary & Key Competencies

- **Structure is Mandatory:** SDLC acts as the backbone, balancing Cost, Quality, and Time.
- **Documentation is Critical:** BRS, SRS, HLD, and LLD are the bridges between stakeholders and developers.
- **Testing is Continuous:** Involves both static Verification (Reviews) and dynamic Validation (Execution).
- **Adaptability:** Choose Waterfall for stability; adopt Shift Left/Right for modern quality assurance.