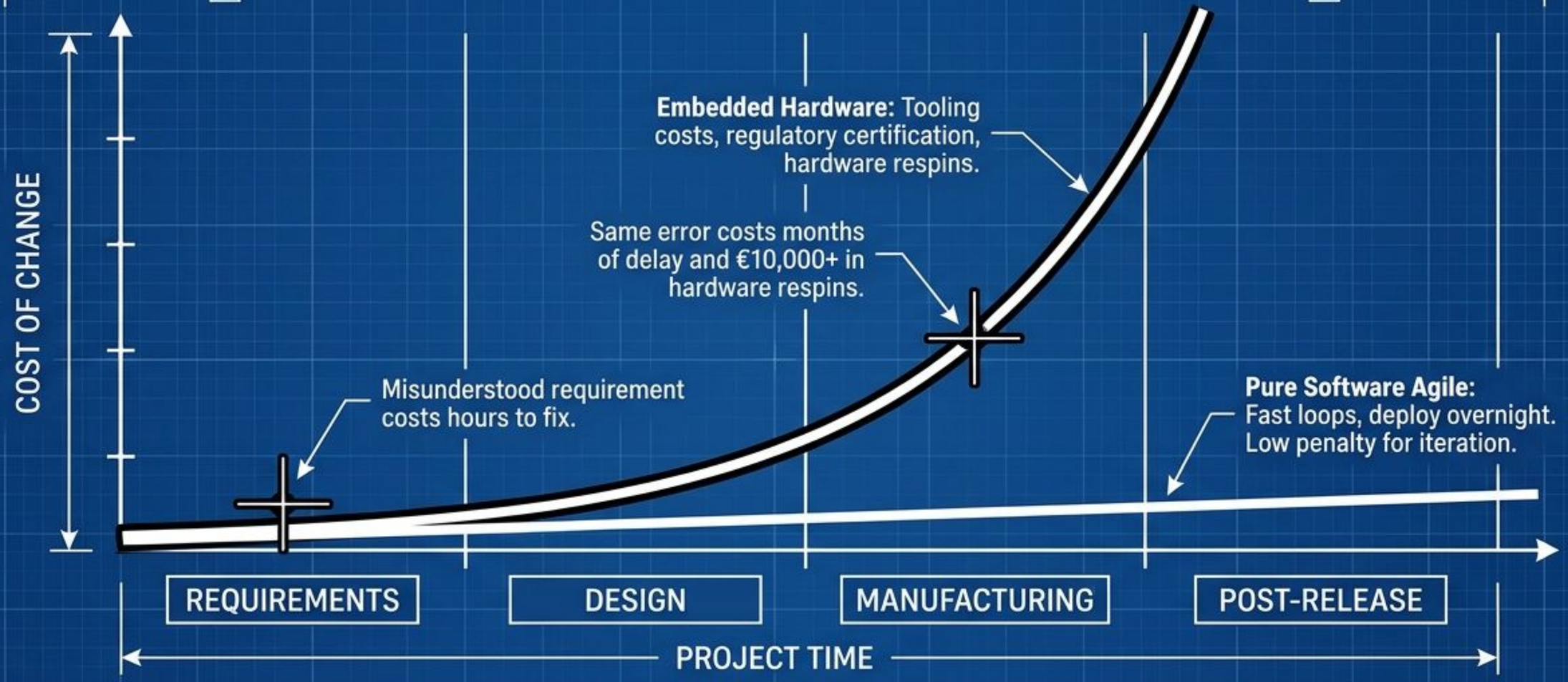
Requirements Analysis for Embedded Products

Turning product concepts into buildable, defensible systems.

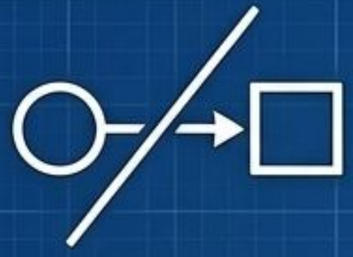
Computerguided Systems | Pre-Development Methodology

THE HIDDEN TRAP OF "JUST START BUILDING"



TAKEAWAY: In hardware, uncertainty does not resolve itself through iteration. The disciplined work we do up front is not a delay—it is the risk reduction that makes development predictable.

The Three Immutable Principles



1. Separate What from How

Preserve design freedom. Define product intent before prescribing technical implementation.

Example:

"Wireless configuration" (Product Need) vs. "Bluetooth" (implementation Detail).

2. Make it Verifiable

A requirement that cannot be tested is a wish.

Force precision and surface testing infrastructure needs (e.g., EMC labs, altitude chambers) early.

3. Confront Risk Early

Deliberately seek out uncertainty.

Classify technical limits and prototype them now.

Do not wait for late-stage development to discover a hardware assumption failed.

WHO NEEDS STRUCTURED REQUIREMENTS?

01

Concept Phase

Trigger: You have a slide deck, sketches, or meeting notes.

Need: Translate raw ideas into concrete functions to understand cost, feasibility, and timelines.

02

Draft Specification

Trigger: You have a Customer Requirements Specification (CRS).

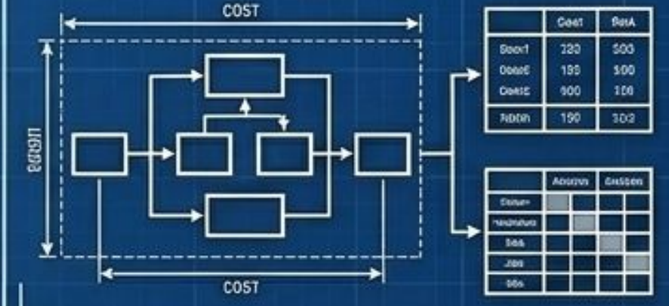
Need: The document feels complete but needs to be rigorously structured, reviewed, and made defensible as an engineering contract.

03

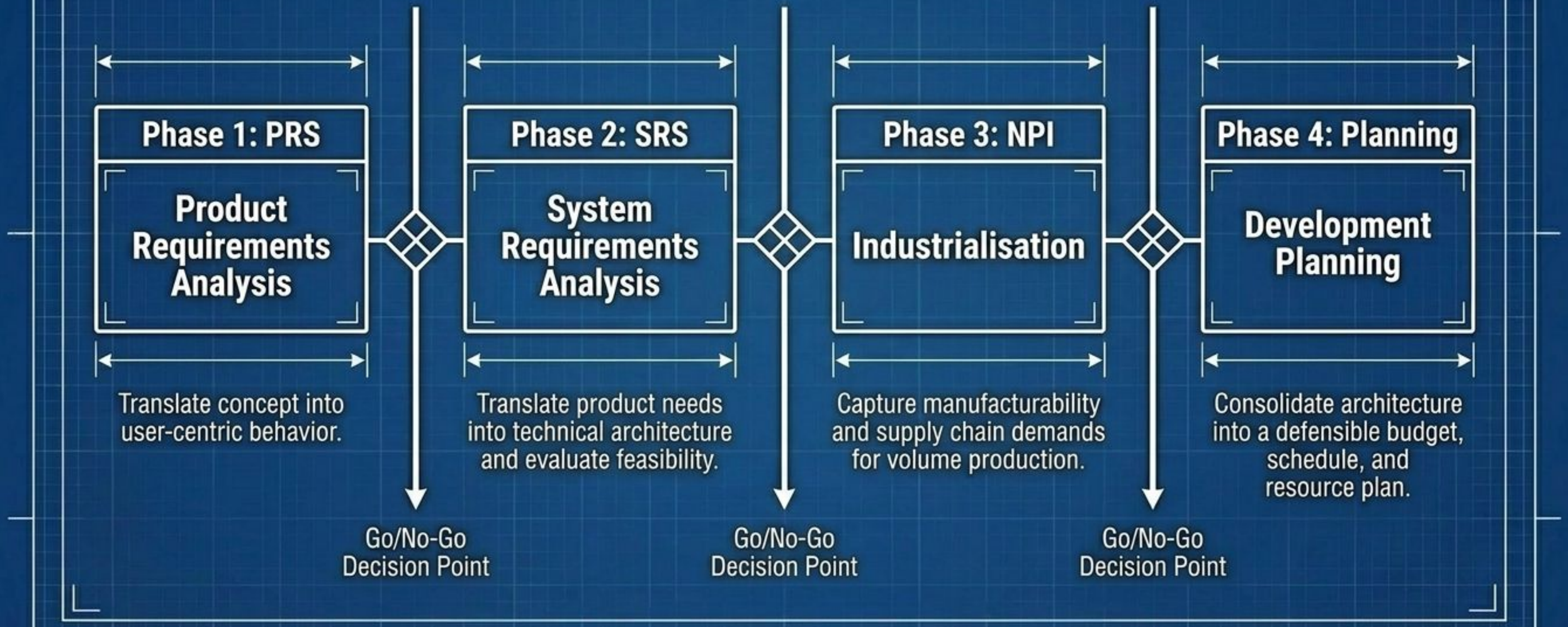
Defensible Estimate

Trigger: You must present a project plan to a board or customer.

Need: Real numbers grounded in real architecture, component selection, and design risk—not top-down guesses.



The 4-Phase Pre-Development Framework



Phase 1: Product Requirements Analysis (PRS)

PRS DOCUMENT

CONTEXT DIAGRAM

Key Insight: Solution Neutrality

The PRS defines **What** the product does and **How** it behaves from an end-user perspective. Design decisions are deliberately excluded.

Human Actors

Defining exactly who—or what—interacts with the system.

External Systems

Defining boundaries, not pin assignments.

Main Product Objective

The fundamental need the product addresses.

Artifacts

Data and physical objects crossing the boundary.

If a requirement cannot be traced back to this defined context, it is out of scope.

THE PRS CORE: USE CASES & SCENARIOS

THE CONCEPT

USE CASES

The high-level intent (e.g., "Configure Network Settings").

SCENARIOS

The pathways to achieve the objective.

SUCCESS SCENARIO

The primary path under normal conditions. (One per use case).

ALTERNATIVE SUCCESS

Achieving the goal via different means or conditions.

FAILURE SCENARIOS

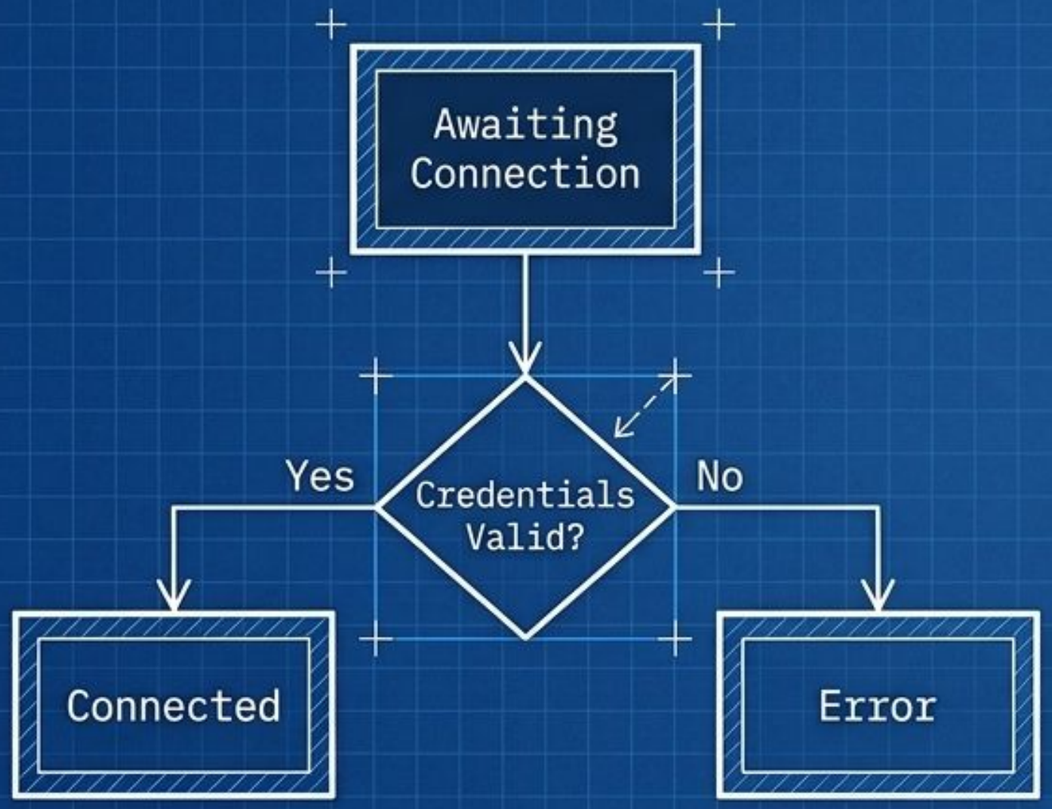
Situations where the objective fails (e.g., timeouts, hardware faults). System recovery must be defined.

THE RULE OF SHARED FUNCTIONALITY

Repeated interaction sequences across different scenarios are systematically extracted into 'Partial Scenarios' to prevent duplication and ensure perfect consistency across the specification.

State Modeling & Functional Extraction

State Modeling



Functional Extraction Table

How narratives become engineering contracts.

Trigger	Actor Action / External Event (e.g., User enters credentials).
System Response	What the system does (e.g., System validates against database).
Extracted Requirement	**REQ-PRS-FR-001: When credentials are submitted, the system shall validate them and transition to the Connected state within 200ms.**

Non-Functional Requirements (NFRs) & Prioritization

	Essential →	Secondary →	Nice-to-have →	Emerging →	Future
Performance	●	●	✓	●	●
Reliability	●	●	✓	●	●
Regulatory compliance	●	✓	●	●	●
Environmental conditions	●	✓	●	✓	●
Power constraints	●	●	●	✓	●

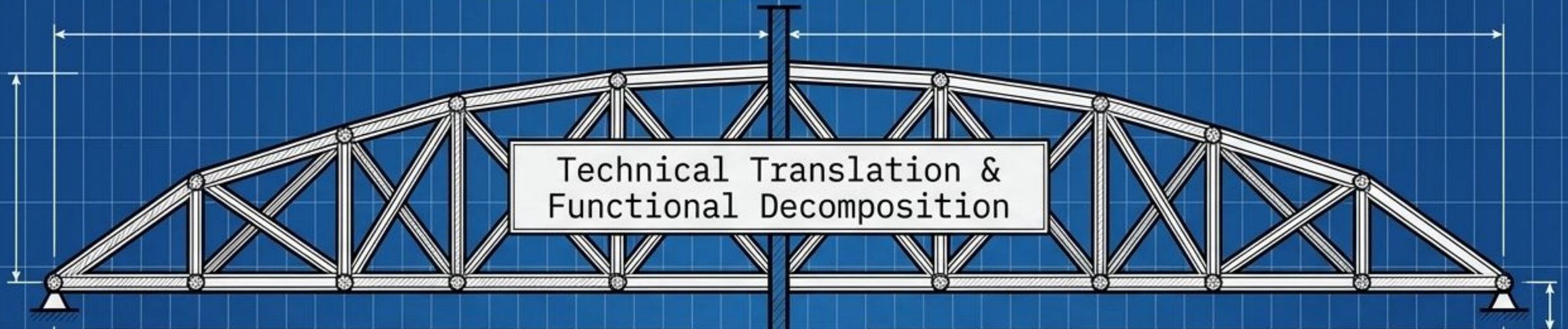
Core Principle

Every NFR requires an **explicit rationale**. A constraint without a documented rationale is arbitrary. We force justification for every threshold to prevent unnecessary engineering costs.

Dependency Consistency Check

We automatically verify that high-priority features do not secretly rely on low-priority (deferred) requirements.

Phase 2: System Requirements Analysis (SRS)



PRS: The World of Intent

Crossing the boundary from What to How.

- User needs
- Functional scenarios
- Environmental constraints

SRS: The World of Implementation

- System architecture
- Hardware/software mapping
- Technical feasibility
- Design risk

Key Insight

The System Requirements Analysis (SRS) evaluates architectural options and technologies. If a product requirement is technically infeasible or economically impractical, we catch it here—before a single CAD model or PCB is designed.

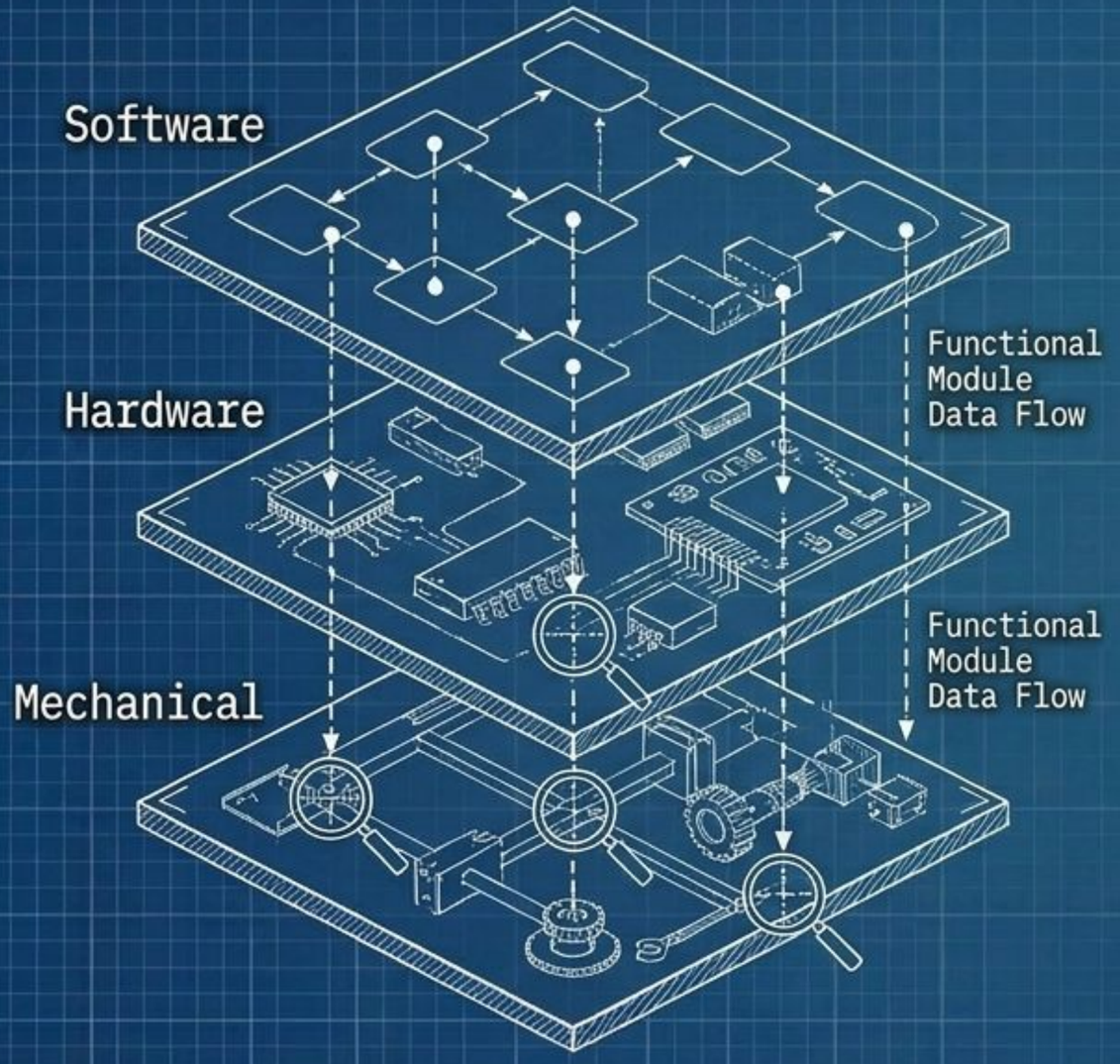
Concept Analysis & Feasibility Classification

How functional modules are mathematically evaluated for risk.

	Trivial Concepts	Balanced Concepts	Exploratory Concepts
Characteristics	Single reasonable, standard solution. Established building blocks.	Multiple viable paths exist.	High uncertainty, pushing known technical limits, unproven technology.
Action	Straightforward documentation; minimal effort required.	Triggers a structured trade-off analysis (Performance vs. Cost vs. Power vs. Availability) to select the optimal path.	Triggers targeted proof-of-concept prototyping to validate feasibility and retire risk before committing to the architecture.

FEASIBILITY DIAGNOSTIC MATRIX - DWG NO. 003 - REV A

Device Architecture & Design Risk (DFMEA)



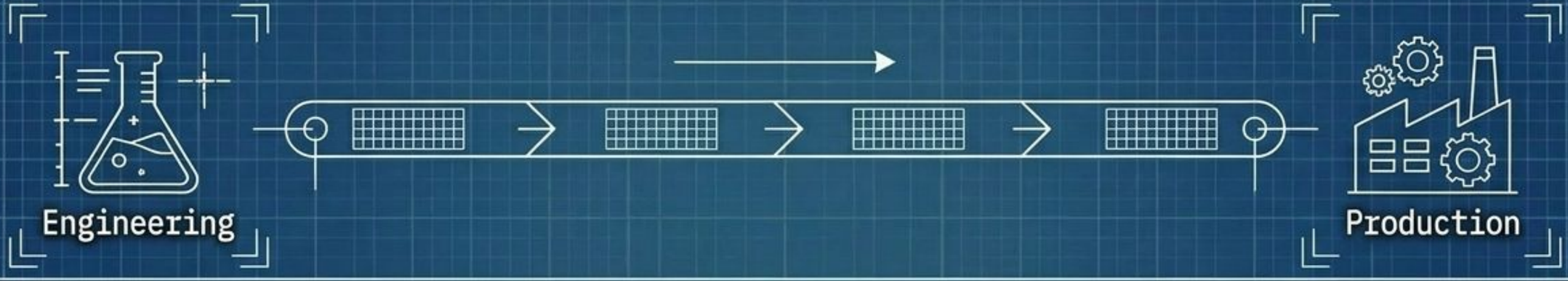
DFMEA Process Breakdown

1. Systematically identify how modules or interfaces fail.
2. Assess severity, occurrence, and detectability.
3. Mandate architectural mitigations (redundancy, fallbacks, new requirements) before detailed design begins.

Takeaway: We hunt for single points of failure while changes cost hours of documentation, not months of redesign.

Phase 3: Industrialisation Analysis

Focus Shift: From ‘Can it be built?’ to ‘Can it be manufactured reliably at volume?’



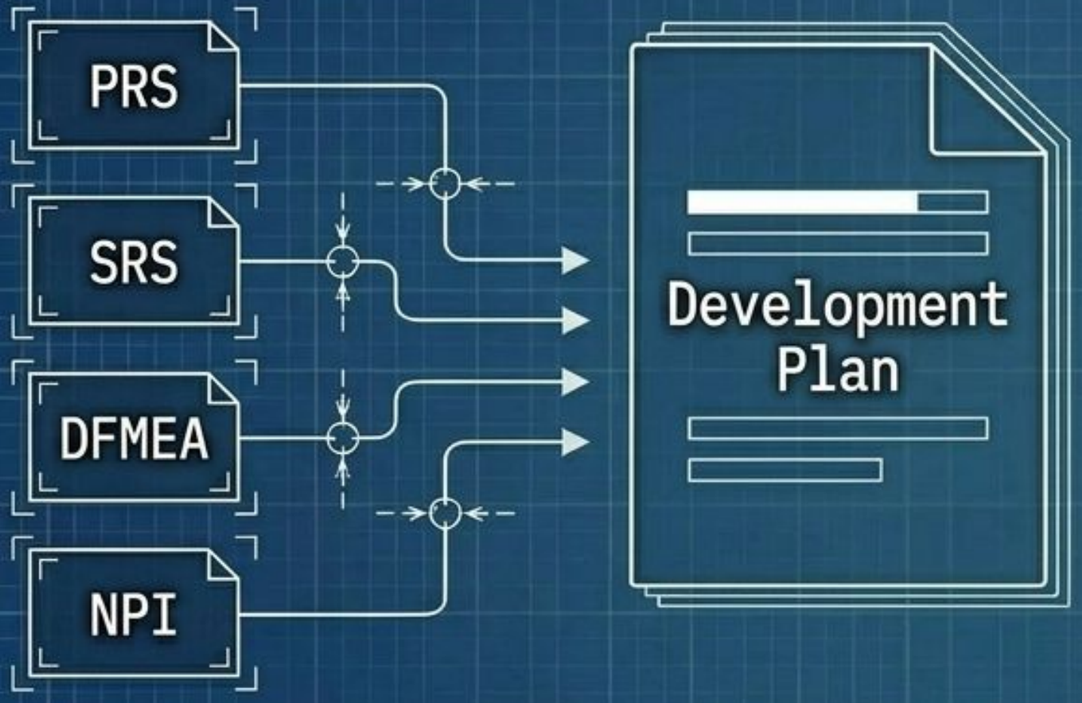
Key Components of the NPI (New Product Introduction) Spec

Manufacturing Concept	Quality & Test	Supply Chain & EMS
Validating critical assembly, calibration, and packaging methods.	Establishing production and end-of-line (EOL) test strategies.	Evaluating Electronic Manufacturing Services (EMS) partners and mitigating component availability and obsolescence risks.

FEASIBILITY DIAGNOSTIC MATRIX - DWG NO. 003 - REV A

Phase 4: Product Development Planning

Consolidated Dashboard



The Output (The Defensible Estimate)

Concept BoM

Preliminary Bill of Materials based on actual architecture, not guesses.

Resource Plan

Competency mapping and work breakdown structure (WBS).

Schedule & Milestones

Concrete development phases and decision gates.

An estimate without clear requirements is a guess. We deliver a project estimation grounded in evidence, giving you numbers you can stand behind.

FEASIBILITY DIAGNOSTIC MATRIX - DWG NO. 004 - REV A