AXIOMATIC DESIGN · COLLECTIVE SYSTEM DESIGN · AI

Axiomatic Design Foundations for AI-Driven Collective System Design

David S. Cochran, Ph.D. and **Joseph Smith** — Purdue University Fort Wayne · System Design, LLC

PURDUE UNIVERSITY FORT WAYNE Center of Excellence in Systems Engineering

Table of Contents

OPENING

- The Challenge — The Valley of Design Death

CSD FOUNDATIONS

- The Flame Model of a System
- The Flame Model — Diagnosis to Design
- The CSD 12-Step Process
- Axiom 1 and Design Predictability

A SINGLE SOURCE OF TRUTH

- Common Terminology — ISO/IEC/IEEE 42010
- The Consistent Design Artifact Model

AI: KEEPER OF THE FLAME

- AI Across the Four Layers
- RAG AI Architecture
- Case Studies — Healthcare and Mechatronics

AI IN ENGINEERING EDUCATION

- The CSD Requirements Ledger
- Inside the Ledger — Feature Tour
- The Student as System Architect

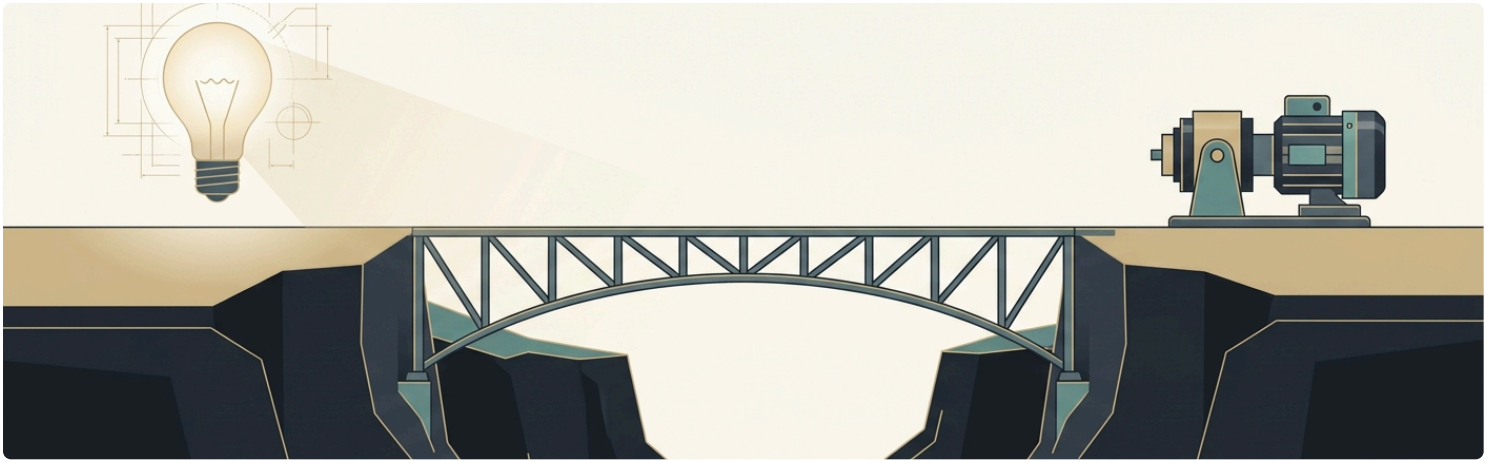
CLOSE

- Thank You · Discussion

Figure numbers (Fig. 1–7) match the companion ICAD paper; slides present them in teaching order, so they may appear out of sequence.

The Challenge: From Design Intent to Realization

FROM DESIGN INTENT TO PHYSICAL REALIZATION



WHERE STUDENTS GET STUCK

Students struggle to cross the **“valley of design death”** — from **design intention to realization** — with an **over-reliance on CAD and simulation tools** without an understanding of the issues required to build things. The result is **last-minute trial and error**.

OUR THESIS

The **CSD 12-Step process** is a **“liberating structure”** that preserves design intent by distilling it into a **Consistent Design Artifact Model (CDAM)**:

- Merges **Axiomatic Design** with **Systems Engineering**
- Starts from **solution-neutral** requirements
- Produces a **machine-readable** artifact for **RAG-driven AI**

At the heart of every great engineering feat lies **Design Intent** — captured, after Nam P. Suh, as independent **Functional Requirements (FRs)** — turning “complex,” unpredictable systems into predictable, improvable engineered ones.

The Flame Model of a System

A SYSTEM AS AN INTEGRATED WHOLE

Collective System Design (CSD) represents a system not as isolated tasks but as an **integrated whole**.

Tone establishes **collective agreement** about what the system must accomplish.

Thinking requires a **consistently-used ontology** to express design information and artifacts — which is where **Axiomatic Design** fits.

Structure is derived from the **Axiomatic Design Decomposition** and the other viewpoints that express the design. Applies to a product design or an enterprise design.

Work generally implements the **leaf-level solutions** of the **Axiomatic Design Decomposition**. The physical realization and sustainability of a system results from establishing **standard work** to define “normal.”

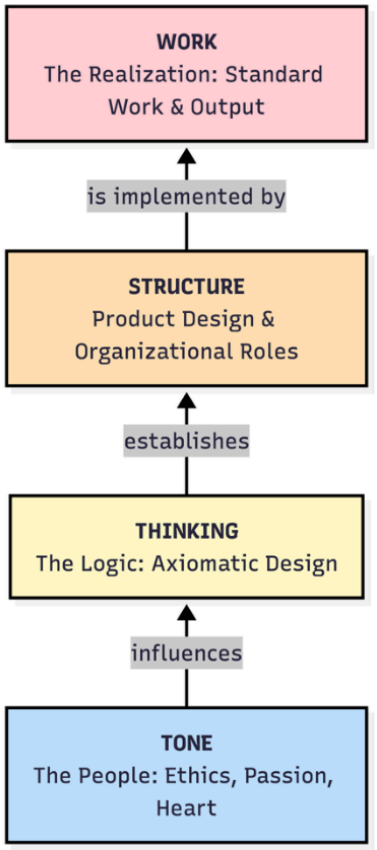


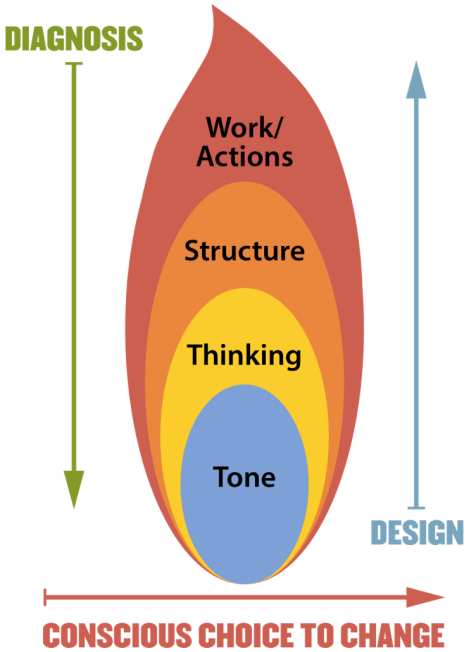
FIG. 1 — THE FLAME MODEL

> Tone sets Thinking; Thinking sets Structure; Structure is realized as Work. People are not the cause of system failures — **poorly designed systems fail people.**

The Flame Model — Diagnosis to Design

DIAGNOSIS ←

Trace symptoms **inward** — from Work to root cause — by asking **5 Whys** at each layer of the flame.



THE FLAME MODEL OF A SYSTEM

→ DESIGN

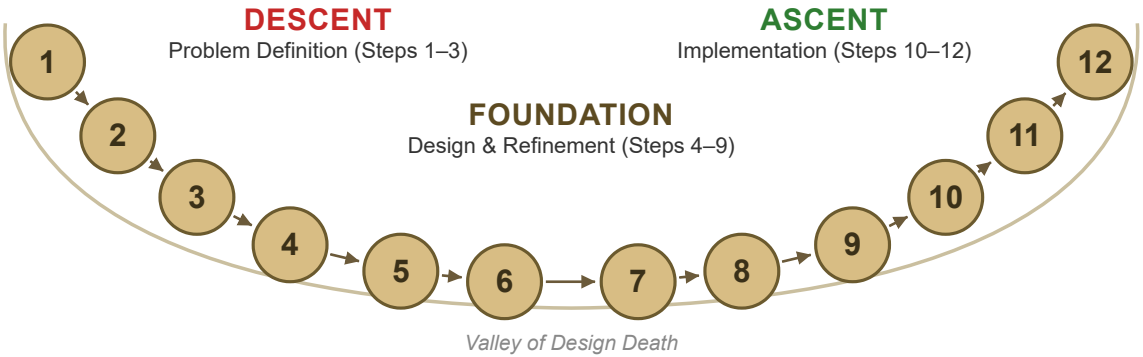
Build **outward** from **Tone** — only once mindset is set can “we” design together.

KEY POINTS

- System Design relies on the people within a system being able to work together — which requires a **conscious understanding of the mindset and attitudes** used to approach each other in making design decisions.
- The benefit of **Axiomatic Design** is that it forces a team to **define a solution (a design decision) to achieve a function** of the design.
- **Lasting change requires a conscious choice** to change existing behaviors in each of the four layers represented by the Flame Model of a system.

THE TWELVE STEPS — A U FROM PROBLEM TO REALIZATION

The **Descent** defines the problem solution-neutrally; the **Foundation** designs and refines at the bottom (the "Valley of Design Death"); the **Ascent** verifies and realizes. Step names match the **Requirements Ledger**.



DESCENT · Problem (1–3)

1. Project Intro & Background
2. Problem Definition
3. Conceptual Alternatives & Selection

FOUNDATION · Design (4–9)

4. Design Decomposition
Analysis, Simulation, Mini-Prototype
5. Design & Process FMEA
6. Detailed Design
7. Design for X
8. Verification Test Plan
9. Validation Test Plan

ASCENT · Realize (10–12)

10. Prototype Build
11. Standard Work
12. V&V + Final Cost

KEY POINTS

- A **design artifact** is any work product that expresses part of the design — an FR, an FRm, a PS, a risk, a verification, or a diagram — in a defined, reusable form.
- Artifacts produced on the descent and the verification on the ascent are bound to the same **Consistent Design Artifact Model (CDAM)** populated at the bottom.
- The purpose of the CSD 12-Step is to provide a **methodology for design that yields predictable results and mitigates unstructured or biased solution selection**.

Axiom 1 and Design Predictability

INDEPENDENCE GOVERNS EVERY STEP

Although only Step 4 is **Design Decomposition**, **Axiom 1** governs all twelve steps — via the recursive **zigzag** between the **Functional** and **Physical** domains.

> **Axiom 1 — Maintain Independence of the Functional Requirements.**

After Nam P. Suh, The Principles of Design (Oxford University Press, 1990).

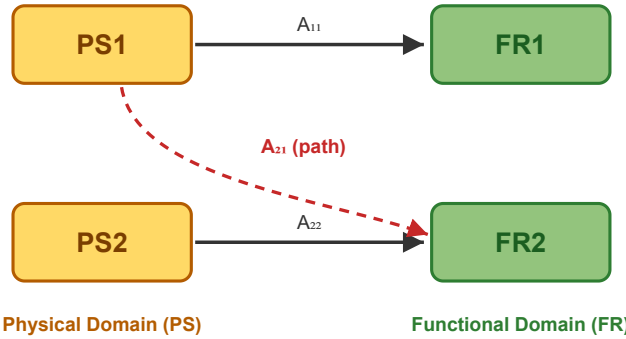
A **decoupled** design yields an **upper- or lower-triangular matrix** — which gives a **predictable implementation sequence** (right).

THE DESIGN EQUATION {FR} = [A]{PS}

$$\begin{bmatrix} FR1 \\ FR2 \end{bmatrix} = \begin{bmatrix} A_{11} & 0 \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} PS1 \\ PS2 \end{bmatrix}$$

lower-triangular

PATH-DEPENDENT IMPLEMENTATION



Fix **PS1** first → then **PS2** — A_{21} sets the order

LOWER-TRIANGULAR = PREDICTABLE SEQUENCE

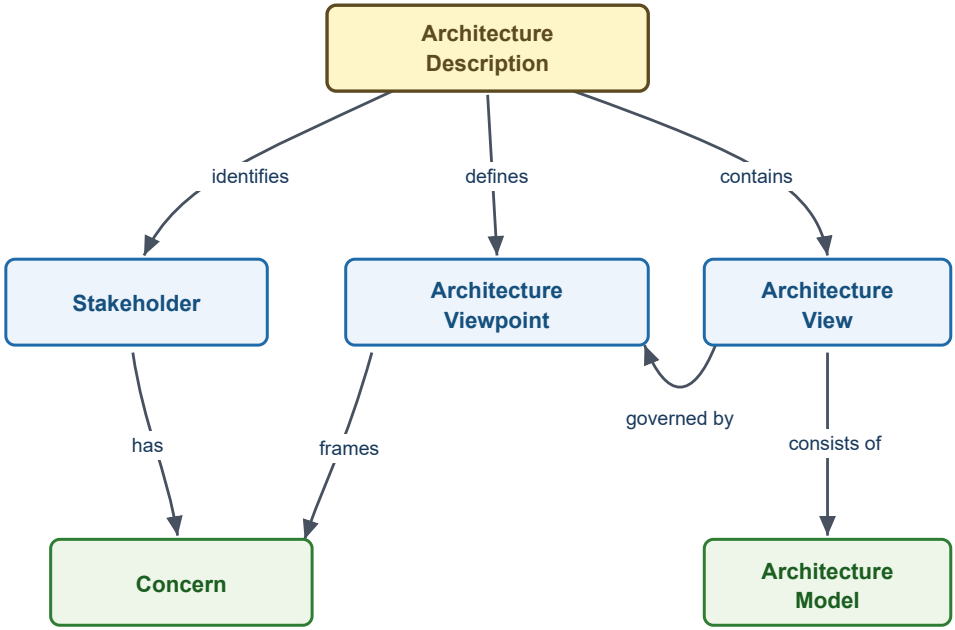
KEY POINTS

- The choice of the **PS (Physical Solution)** to achieve an **FR (Functional Requirement)** is a decision made by the **designer(s)**.
- **Diagonal** = uncoupled (most predictable); **triangular** = decoupled, predictable *if the sequence is respected*; **full** = coupled, with unpredictable cross-talk.

A COMMON TERMINOLOGY

A single source of truth needs **shared words**. ISO/IEC/IEEE 42010 — the architecture-description standard — supplies the ontology CSD uses, separating the **Architecture** from the **Architecture Description** that expresses it.

Each **Architecture Viewpoint** frames a stakeholder's **concern** — and **Axiomatic Design is one such viewpoint**.



Axiomatic Design = one Viewpoint that frames a Stakeholder's Concerns

ISO/IEC/IEEE 42010 CONCEPTUAL MODEL

ISO/IEC/IEEE 42010:2011 — Systems and software engineering — Architecture description.

KEY POINTS

- **Architecture** — the fundamental concepts and properties of a system in its environment, embodied in its elements, relationships, and the principles of its design and evolution.
- **Architecture Description** — the work product (the set of artifacts) used to express an architecture.
- **Axiomatic Design is one viewpoint** used to frame a stakeholder's concerns; **42010 is a liberating structure** that lets designers frame concerns with multiple viewpoints.

ONE SHARED MODEL — A SINGLE SOURCE OF TRUTH

The **CDAM** is **one shared, agreed model** that every discipline reads from and writes to — so mechanical, electrical, and software teams don't drift into separate, conflicting artifacts (i.e., documents).

The AI **reviews changes and flags conflicts** — e.g. an FR with no measure, a likely FR–PS coupling, an interface with no FR — and **people make the edits**.

A structured model the AI can navigate to run **automated design audits** — keeping the design honest as it grows.

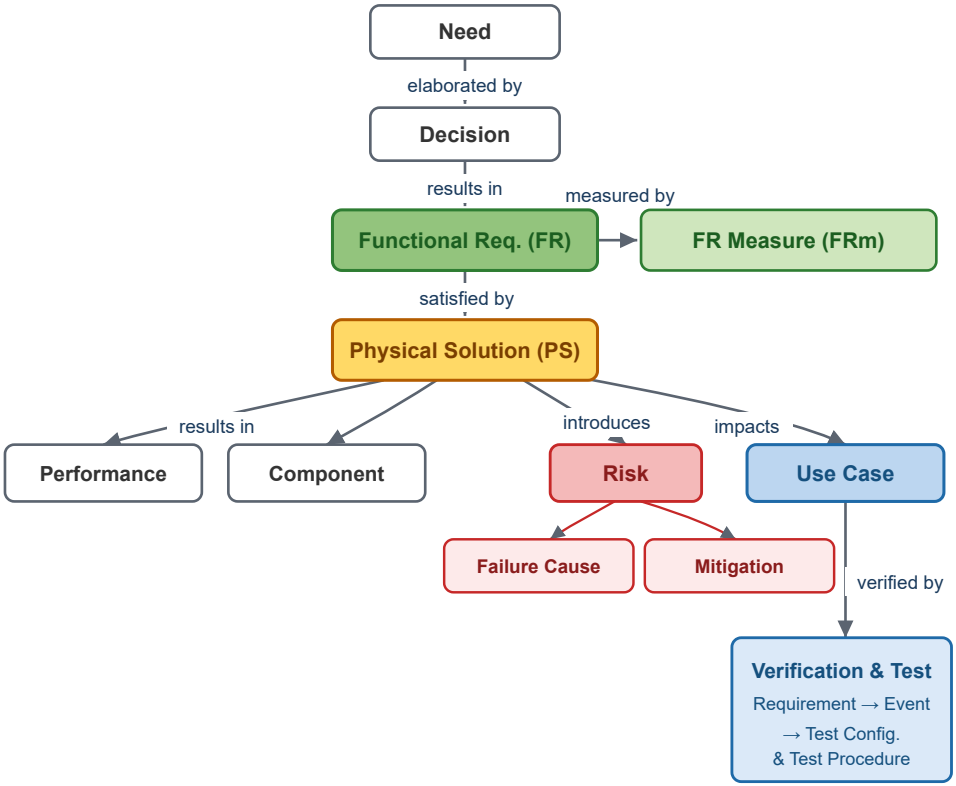


FIG. 3 — THE CDAM INFORMATION MODEL

ONE AI, FOUR ROLES — ONE PER LAYER OF THE FLAME

WORK Implementation Partner	Across the valley of design death, it **sequences the build by path-dependency** — PS1 verified before PS2.
STRUCTURE Interdisciplinary Glue	Uses the CDAM to connect mechanical, electrical and software — **warns when a change in one domain breaks a requirement in another**.
THINKING Enforcer of Rigor	Won’t validate a design that breaks Independence; catches a coupling **at the cost of a conversation, not a build**.
TONE Psychological Safety	Audits the **logic, never the person** (“FR2 is coupled to PS1 via A_{21}”) — this is how it drives out fear.

Tone at the base of the flame → Work at the tip · read-only early, an active partner only at realization

IN PRACTICE — HOW WE DEPLOY IT (INPUT · PROCESS · OUTPUT)

INPUT — information governed and structured by the CDAM + 12 Steps → **PROCESS** — the 12 Steps structure the thinking; they don’t replace it → **OUTPUT** — capability beyond what is possible without AI

KEY POINTS

- The goal is to ensure that **Axiomatic Design practitioners use consistent terminology** across projects.
- That we **embrace multiple viewpoints** within the engineering lifecycle.
- With the emergence of the most powerful tool ever conceived, that we practice a **conscious tone of well-being with each other**.

AN AI AGENT, GROUNDED ON THE CDAM

A student or engineer query goes to an **AI agent grounded on the CDAM** and the primer. ("**RAG-based**" just means *retrieval-augmented* — the agent answers from **our own model and corpus**, not generic training.)

Because the knowledge is **structured**, the agent **reviews and explains** rather than guesses:

- Detects coupling and checks FR / PS mapping
- Flags missing measures and interfaces
- Explains, in the terms of the CSD 12-Step process

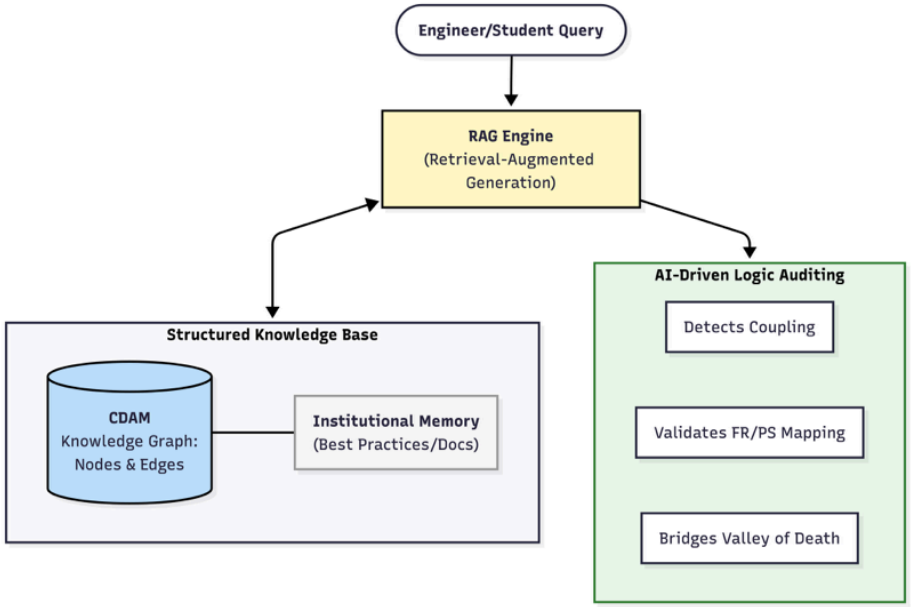


FIG. 7 — THE AI AGENT OVER THE CDAM

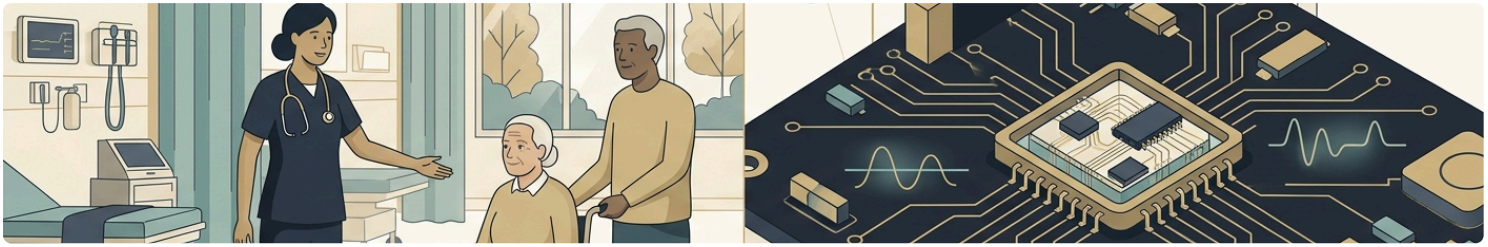
> The agent **reviews and flags**; the **student decides and edits**.

KEY POINTS — WHAT THE CDAM GIVES STUDENTS AND DESIGNERS

- One shared model means **less rework and fewer late surprises**.
- The design can be **checked against the process instantly, 24/7**.
- Design knowledge **carries from one project (and team) to the next**.

ONE LOGIC, TWO VERY DIFFERENT DOMAINS

The same decoupling logic translates high-level intent into measurable requirements *before any physical solution is chosen* — whether the system is human-centric or hardware-centric.



	Healthcare — ER Triage	Mechatronics — HVAC PLC
Functional Requirement (FR)	Categorize patient acuity	Transmit digital data reliably
FR Measure (FRm)	Time to triage ≤ 5 min	Packet delivery ≥ 98%
Physical Solution (PS)	Rapid Triage protocol	Narrowband BFSK Modem
AI's role	Design Auditor and Synthetic Implementation Partner across a hospital network	Technical Assistant cross-checking datasheets vs. IEC 61000-4-6

Franciscan Health (Q4 2025): decoupled logic ensured triage speed did not compromise clinician-matching accuracy — design patterns scaled across hospitals without intent drift.

Residential HVAC: RAG retrieved the IEC standard and modem datasheets together — minutes instead of hours of manual cross-referencing.

Same CDAM grammar, two domains — the "Thinking" layer stays rigorous across both.

AI in Engineering Education: The Requirements Ledger

Senior Design and SE

THE CDAM, MADE USABLE FOR STUDENTS

The **CSD Requirements Ledger** brings the CDAM to life for **senior design** and SE education — one place where every FR, FRm, PS, risk, and verification is a **single source of truth**. The AI agent:

- Guides students through the **12 Steps, viewpoint-by-viewpoint**
- **Audits coupling** as the design is entered (reviews, never authors)
- Keeps **one shared model** across mechanical, electrical and software
- A logic-driven **co-pilot, available 24/7**
- **States what a good FR looks like**
- **Derives / suggests FRs** from the System Boundary and Use Case diagrams
- **Points to the solutions and interfaces to build early**

THE LEDGER AT WORK — CATCHING A SOLUTION IN DISGUISE

A STUDENT ENTERS AN "FR"

FR3: "Use a **12 V Li-ion battery pack**" ⚠

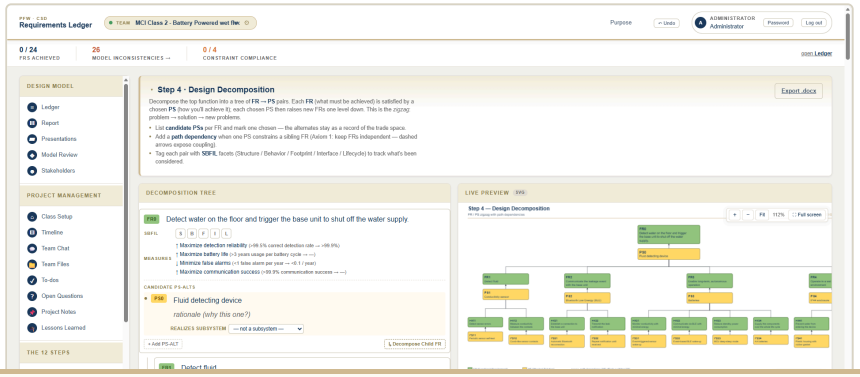
AI REVIEW — IN REAL TIME

That's a **Physical Solution**, not a Functional Requirement — it names the part before the function. An FR states **what the system must do**, solution-neutrally. Define the function first; the PS follows.

CORRECTED IN THE LEDGER

FR3: Store electrical energy

PS3: 12 V Li-ion battery pack ✓



OUR CONSCIOUS APPROACH TO AI IN ENGINEERING EDUCATION

The **SE Center** is using AI technologies to:

- **Build engineering tools** that help focus students' time and energy on **designing and building working systems**
- **Provide real-time feedback** on the design model from **trained AI agents**
- **Inhibit the use of AI to do "the thinking"** by providing a **process** that guides decision-making and implementation, yet requires engineers to do the thinking and learning

Inside the CSD Requirements Ledger

One Model · Many Viewpoints

1 LIVE PROGRESS & HEALTH

FRs achieved, model inconsistencies and constraint compliance — updated live as engineers design.

2 SINGLE SOURCE OF TRUTH

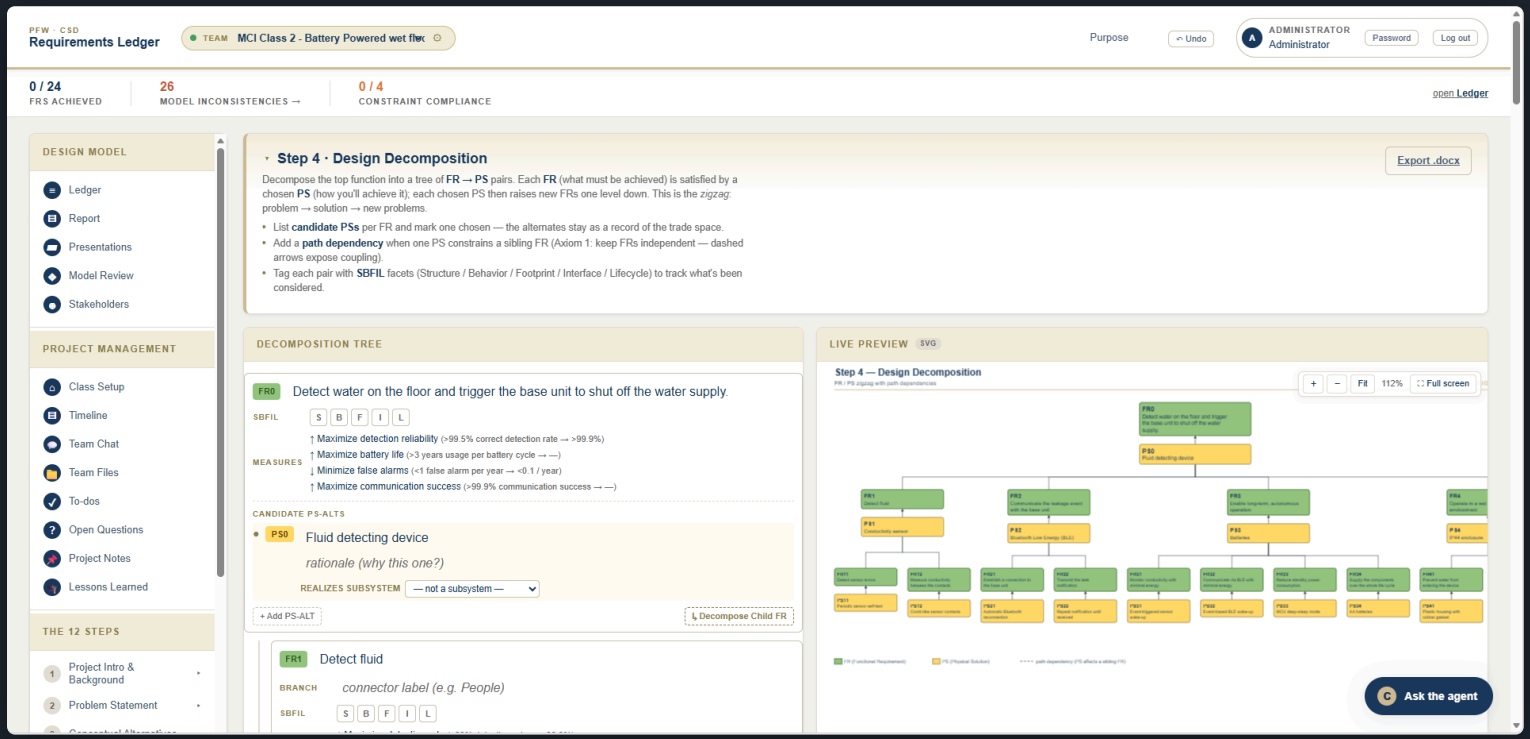
Every FR, FRm, PS, risk and verification in one shared model — the ledger.

3 AD DECOMPOSITION BUILDER

The FR→PS Axiomatic Design decomposition, built from student-authored inputs.

4 REPORT BUILDER

Documentation auto-populated from the model — one click to export a PDF.



5 PRESENTATION BUILDER

A live, editable presentation view — generated from the model.

6 PROJECT MANAGEMENT

Teams, timeline, to-dos, open questions and team chat — the project, organized.

7 ASK THE AGENT

A CSD Primer–grounded co-pilot, 24/7 — it reviews the design, never authors it.

FROM PART-PICKING BIAS TO SYSTEM ARCHITECT

- A **solution-neutral start** reduces ambiguity in every later stage
- Axiom 1 keeps designs **predictable and improvable**
- A **single source of truth** (CDAM) + **common terminology** (ISO 42010) defeat the silo effect
- A grounded **AI agent (RAG-based)** audits logic — it does not replace judgment

CSD relies on a **CDAM and structured viewpoints** to do design; the **Requirements Ledger** gives a user-friendly **AI agent (RAG-based)** to implement CSD on that foundation — shifting students from **part-picking** to proactive **System Architects**.

THANK YOU · DISCUSSION

Axiomatic Design Foundations for AI-Driven Collective System Design

A logic-based bridge from design intent to physical realization — the **CDAM** and structured viewpoints as a single source of truth, implemented by a **grounded AI agent**.

AUTHORS
David S. Cochran · Joseph Smith

ONLINE
sysdesign.org

CONTACT
dscochra@purdue.edu