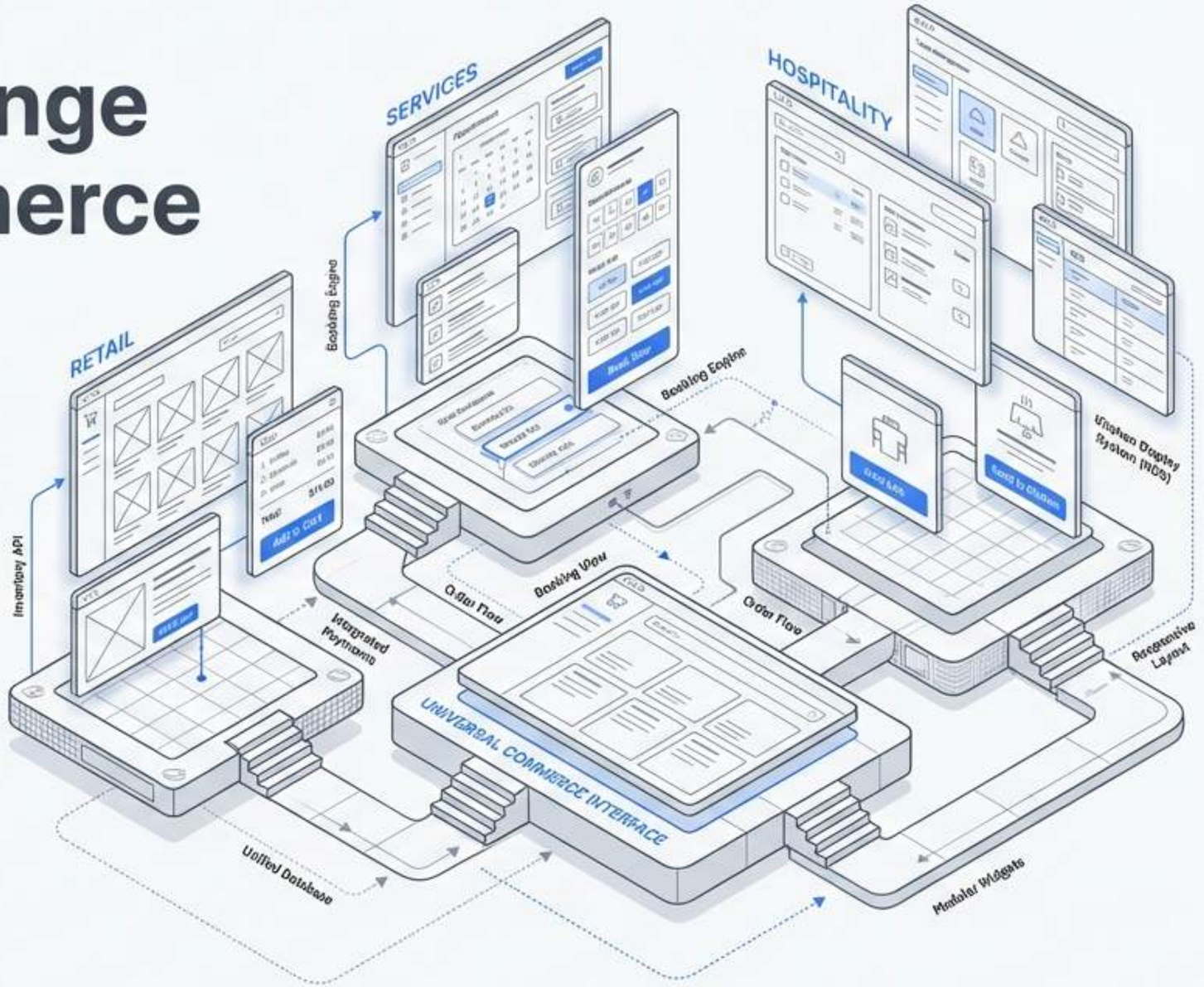
UX CASE STUDY

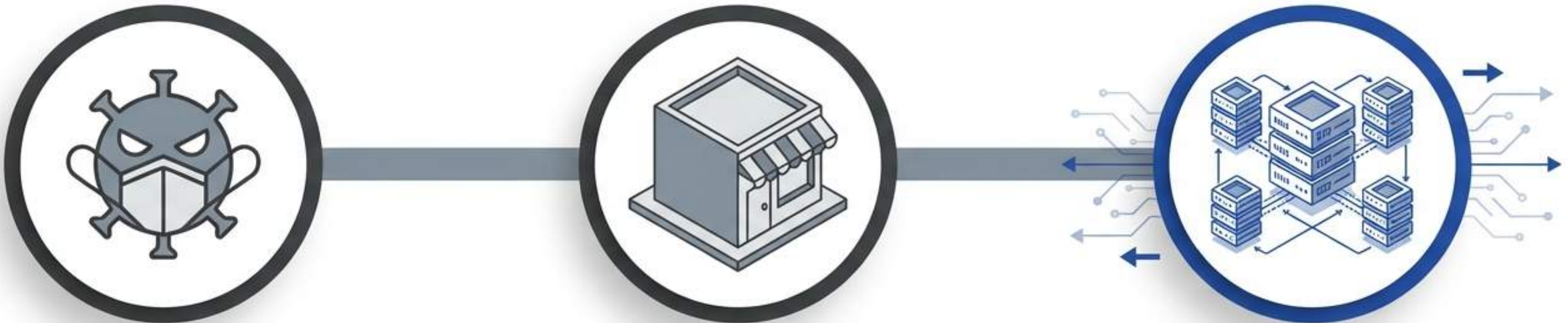
# eStore: The Challenge of Universal Commerce

From a Digital Menu to a Multi-Vertical Marketplace for QueueBuster POS.


Designing a single interface that accommodates the distinct logic of Retail, Services, and Hospitality.

# The Pivot: Survival Driven Evolution



### 2020 Context

Response to Covid-19. Digital Menu & Contactless Ordering.

### The Pivot Goal

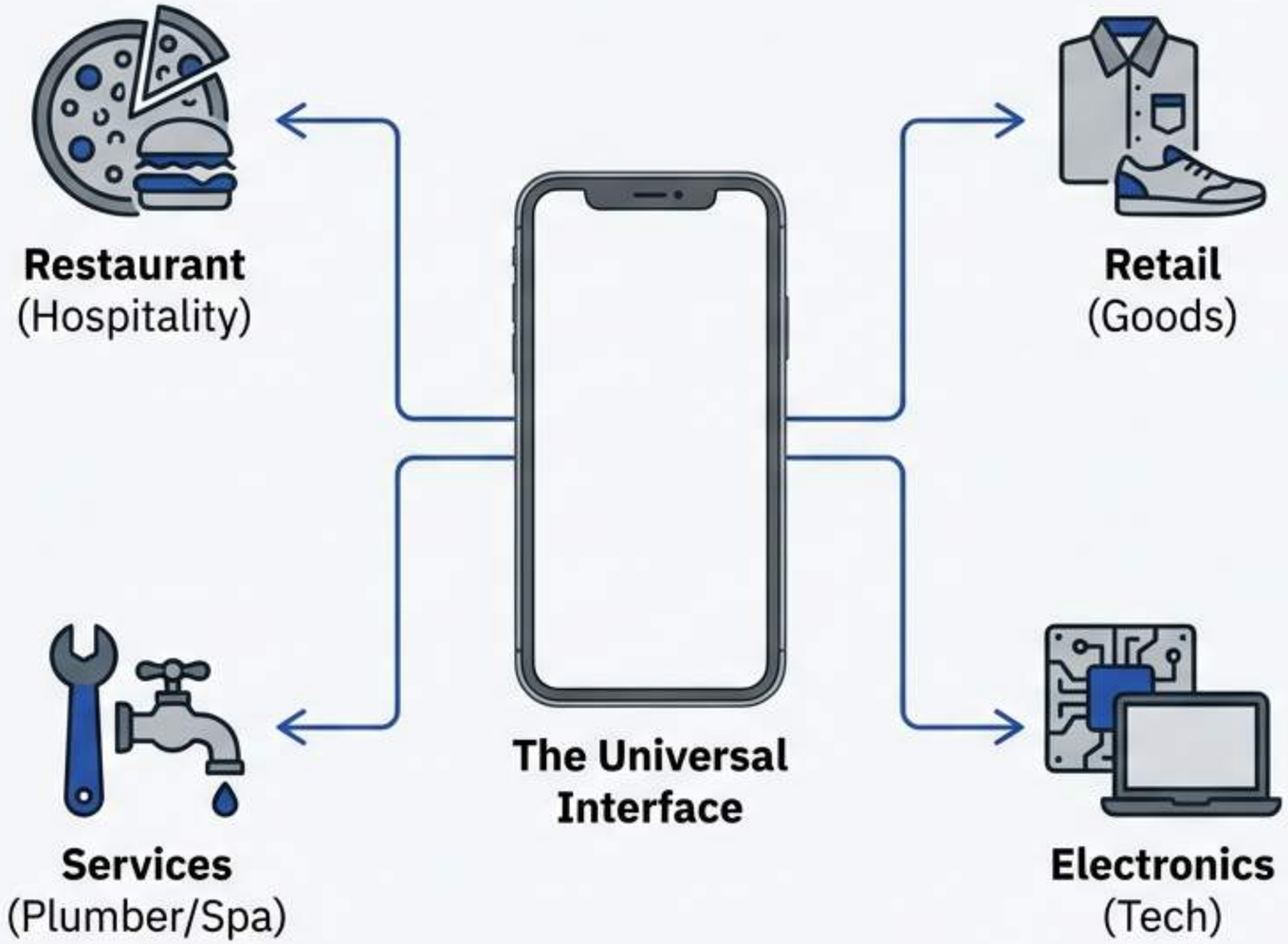
Enable QueueBuster POS customers to sell online.

### The Reality

**Full E-commerce Portal. Payments, Inventory, Universal Logic.**

The project began with a singular focus on safety—contactless ordering—but evolved into a foundational business requirement. The challenge was no longer just about ordering food; it was about building a survival engine for merchants across incompatible verticals.

# The Paradox of the Universal Interface



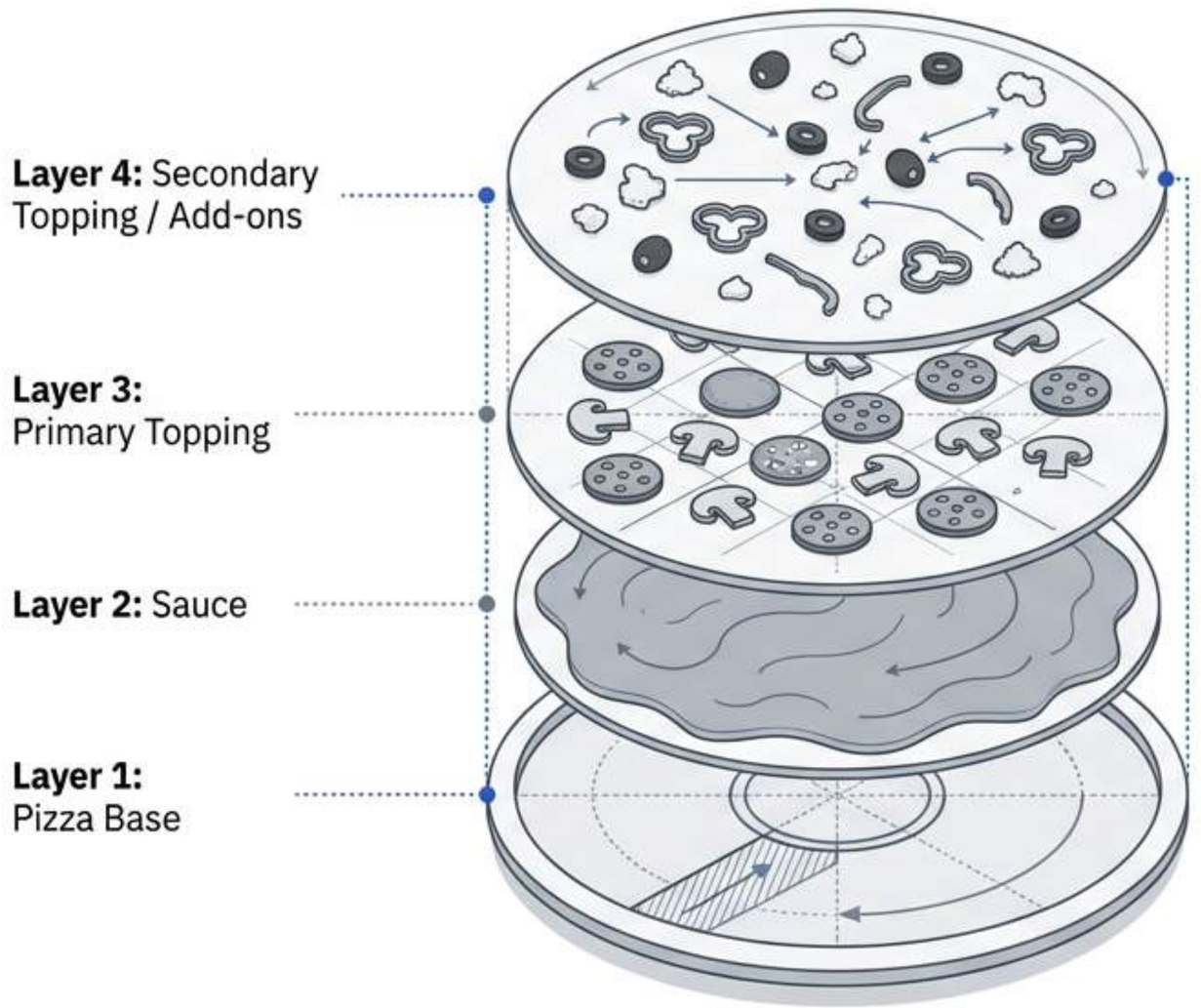

The project became complex when the interface had to be designed for different industries simultaneously.

How do you build a single UI that accommodates products and services that share no structural logic? A spa appointment is not sold the same way as a smartphone.

# Deep Dive: The Logic of a Pizza

**THE LOGIC TREE**

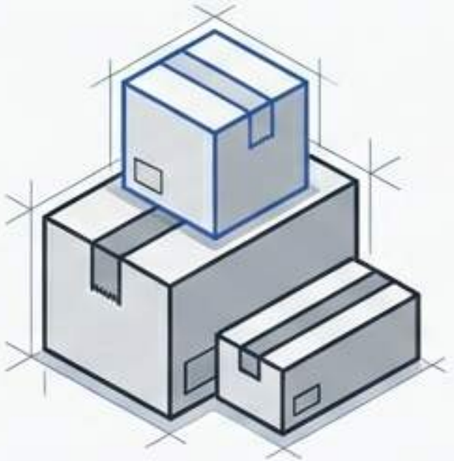
- **Nested Add-ons:** Options exist inside layers.
- **Dependencies:** Some choices are mandatory; others are optional.
- **Conflicts:** There can be a product where one add-on cannot be added with another.




To the user, it is just dinner. To the system, it is a complex dependency tree.

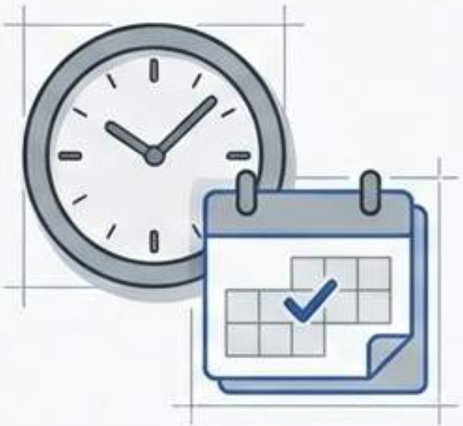
# Tangible Goods vs. Intangible Services

## Retail & Electronics



- Fixed Stock Inventory
- Defined Variants (Colour, Brand, Size)
- Specific Hierarchy
- Physical Delivery

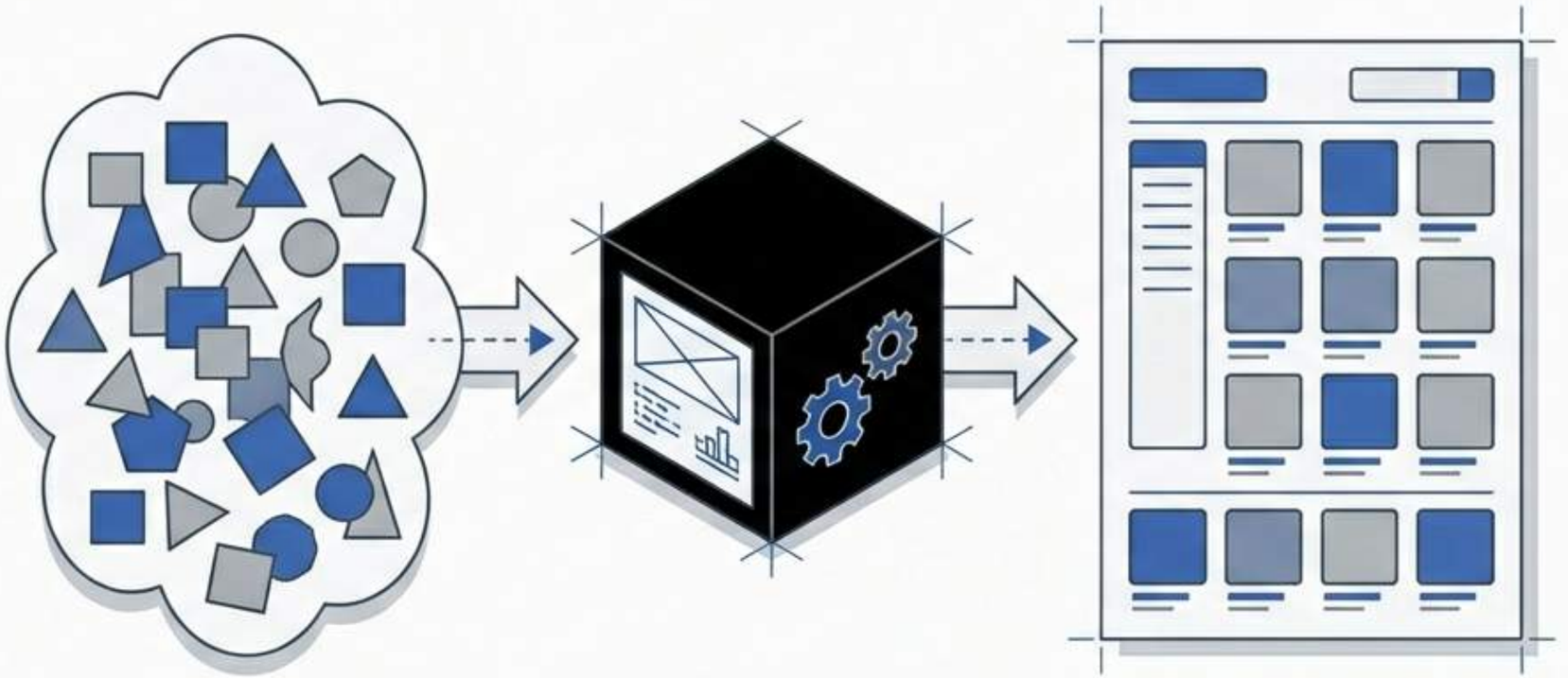
## Services (Spa, Plumber, Electrician)



- Time-based Availability
- Abstract Descriptions
- No Physical Inventory
- Service Radii / Location based

Key Design Question: How does the same ‘Add to Cart’ button function for a physical television and an hour of a plumber’s time?

# Constraint I: The Loss of Inventory Control



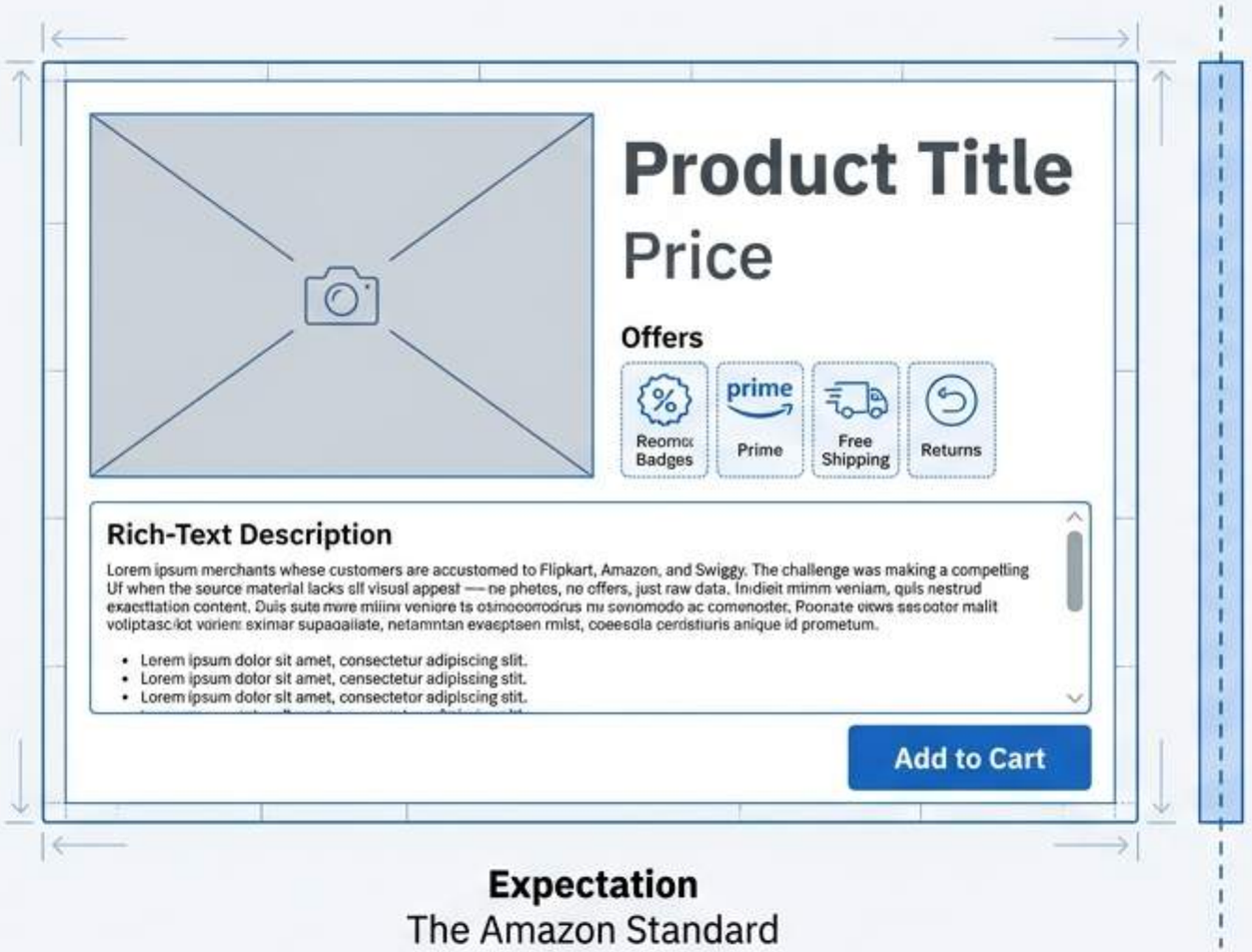
Merchant Data
(Uncontrolled Input)

eStore Engine

Customer
Interface

- Unlike a curated store (e.g., Apple), eStore does not control the merchandise.
- **Constraint:** We cannot control the number of categories, sub-categories, brands, or variants.
- **Risk:** The UI breaks if the merchant adds too much data (500 categories) or too little (1 category).

# Constraint II: Designing for Minimum Viable Data



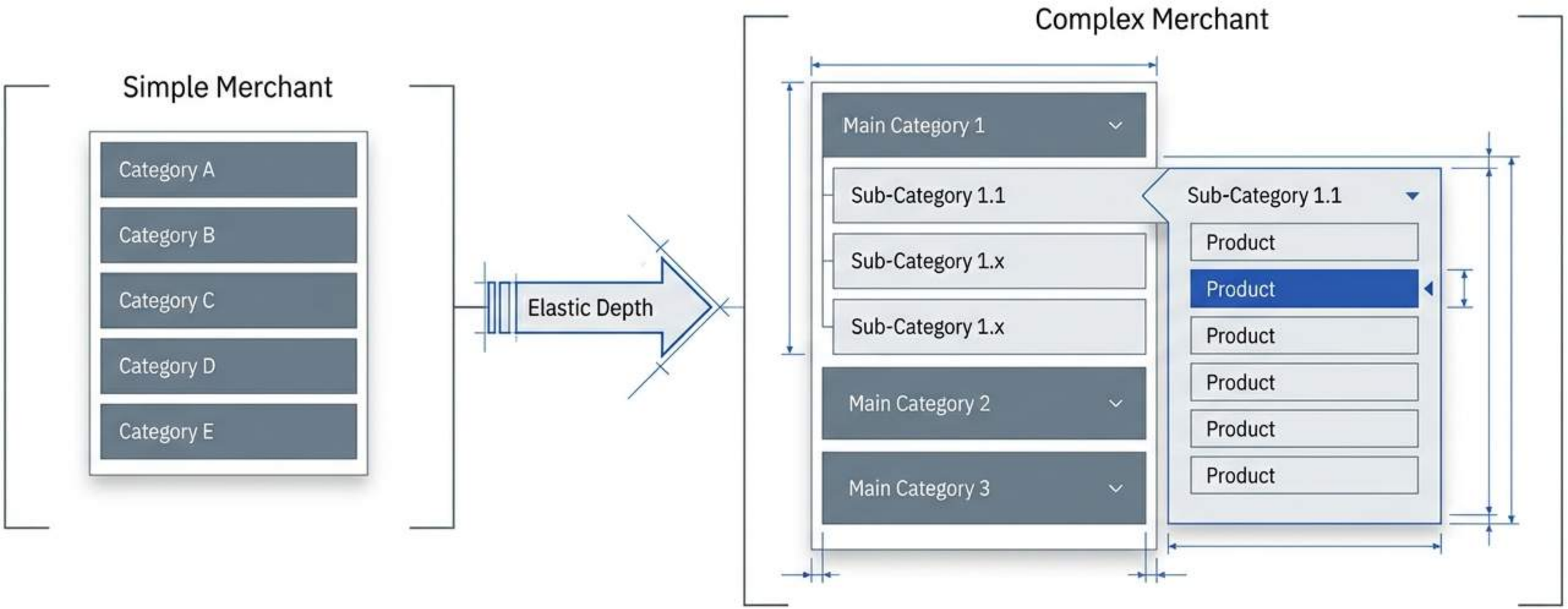

**Expectation**
The Amazon Standard


**Reality**
The QueueBuster Reality

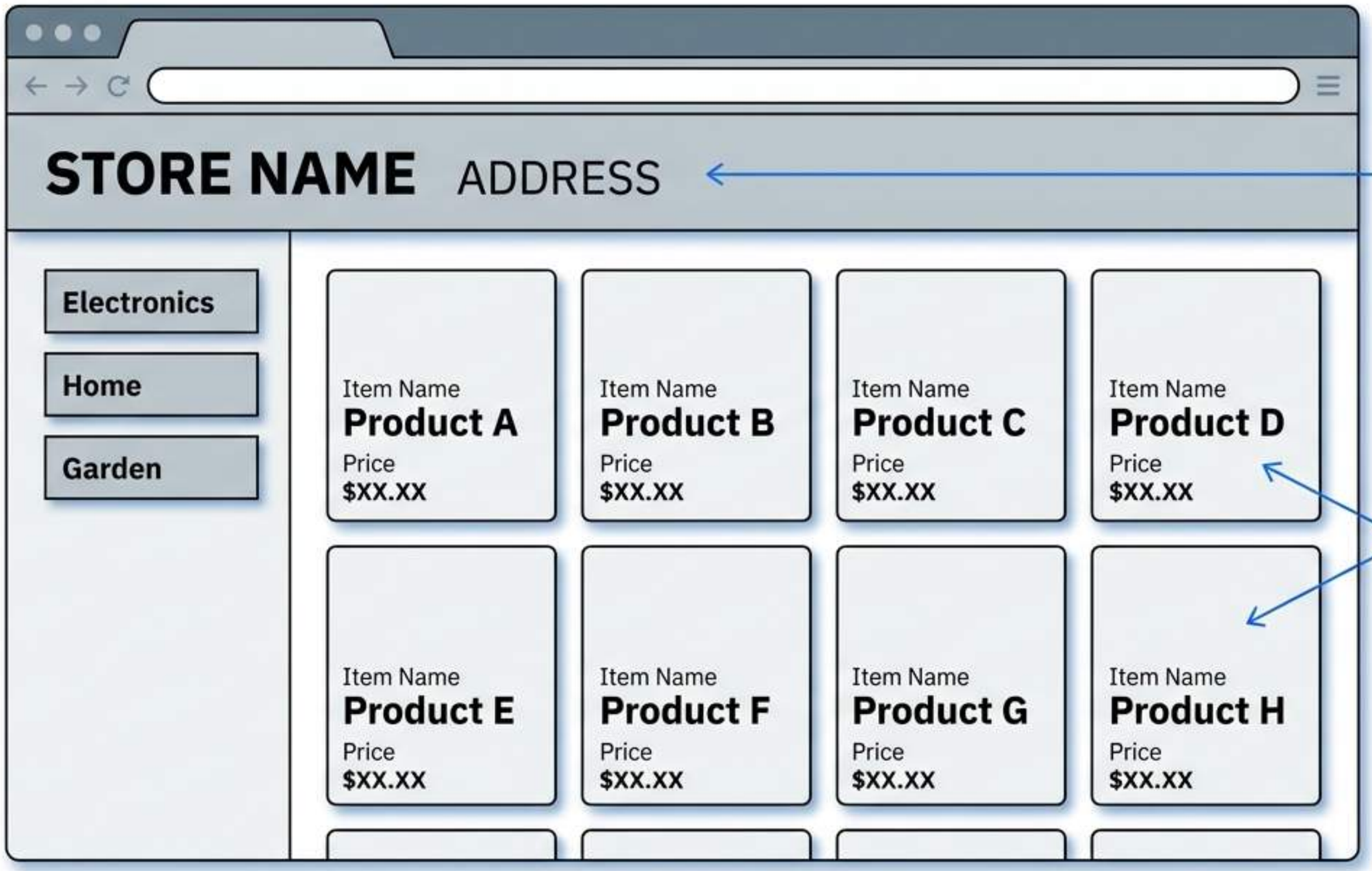
We were targeting merchants whose customers are accustomed to Flipkart, Amazon, and Swiggy. The challenge was making a compelling UI when the source material lacks all visual appeal—no photos, no offers, just raw data.

# The Architecture of Elastic Categorisation

The basis of e-commerce is the categorisation of products. The solution required a hierarchy that could expand or contract based on the merchant's sophistication. The complexity is hidden until the data requires it to surface.



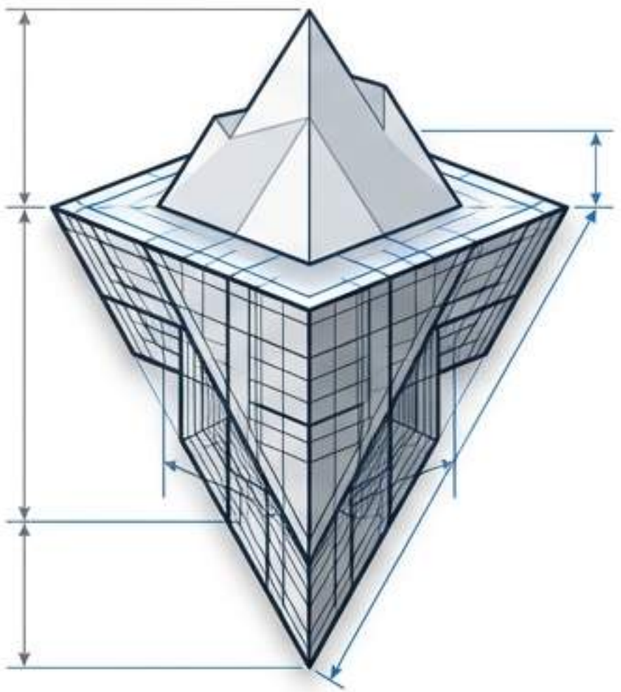
# The Solution: Web Interface



# The Solution: Mobile Experience

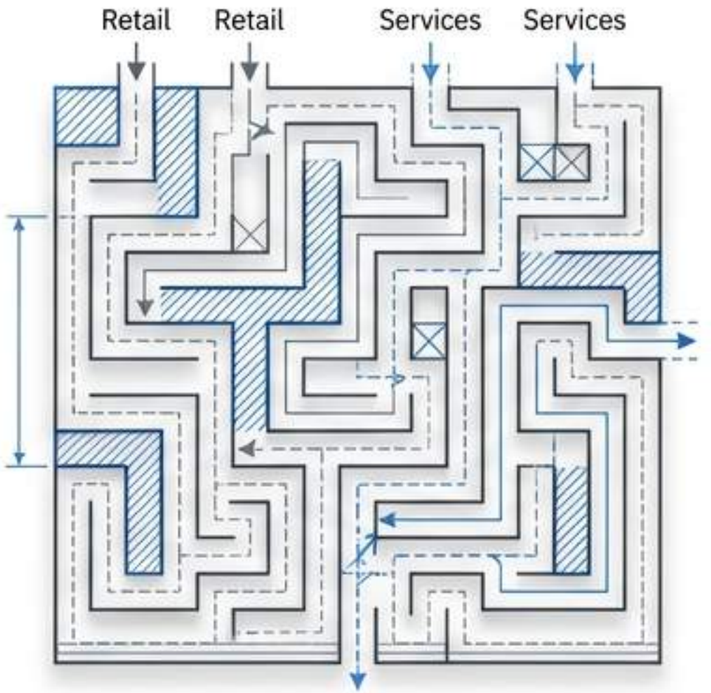

Translating complex dependency trees into a linear mobile flow.

# Strategic Learnings



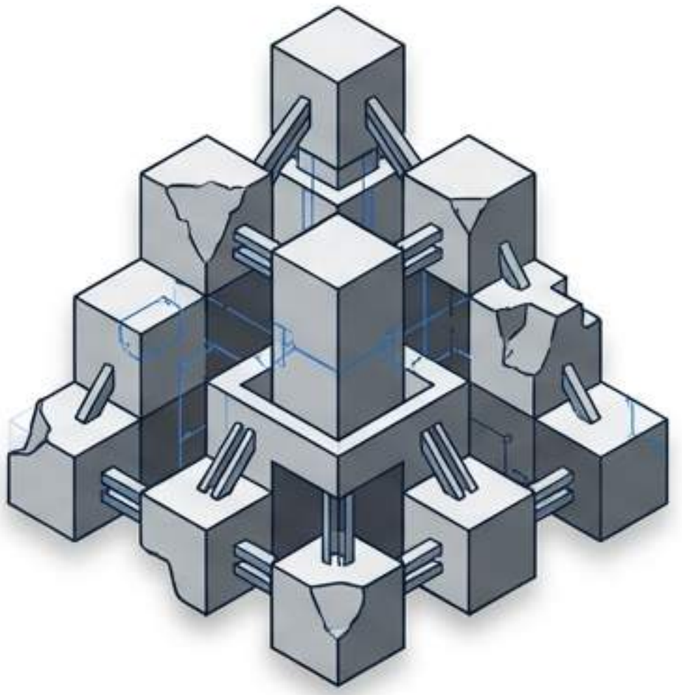
## The Illusion of Simplicity

It looked easy to design an e-commerce interface, but back-o-end logic is deceptively deep.




## Edge Cases are the Foundation

You cannot design the ‘happy path’ first when industries (Retail vs. Services) contradict each other.



## Data Agnosticism

A scalable platform must function beautifully even when the input data is ugly or incomplete.

“It looked easy to design... but deep down when you work with teams it is different.”

— Madhavan, Product Designer

CONCLUSION: True UX capability is not just about making things look good;
it is about organising the chaos of the real world into a system that feels simple.