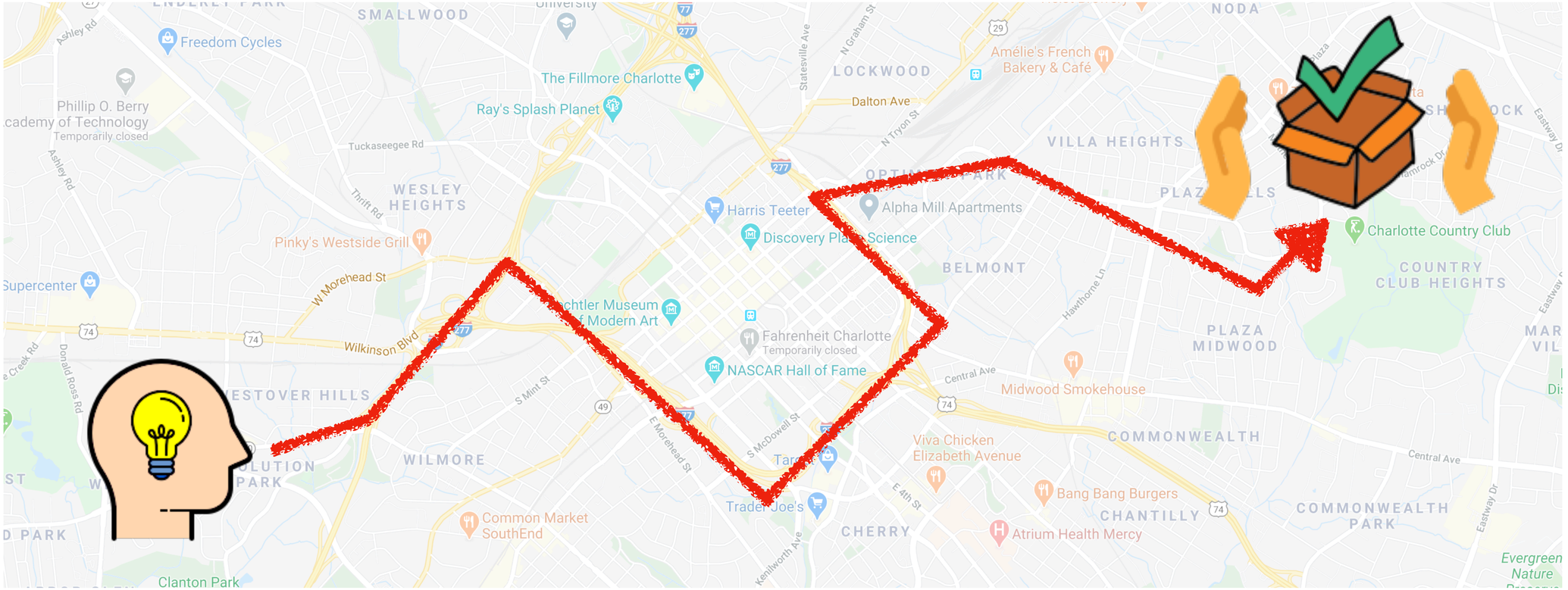
Software Architecture

A story about business value and trade-offs



About me

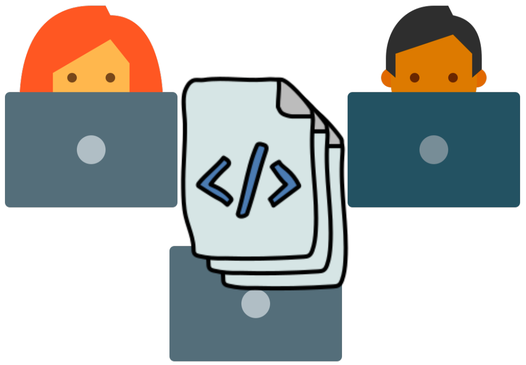
- Senior Software Engineer
- 14 years working in the software industry
- Passionate about software development

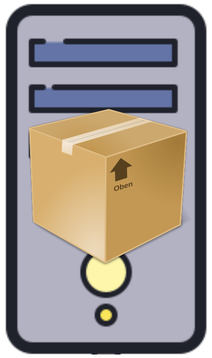
Ge CA

LOWE'S

WELLS FARGO

GE Aviation



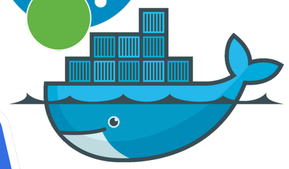

Oben

cassandra

Couchbase

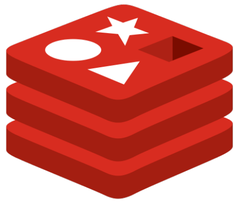
neo4j

Istio



docker

Spinnaker



redis

Azure

okta

mongoDB

Jenkins

kubernetes

dynatrace

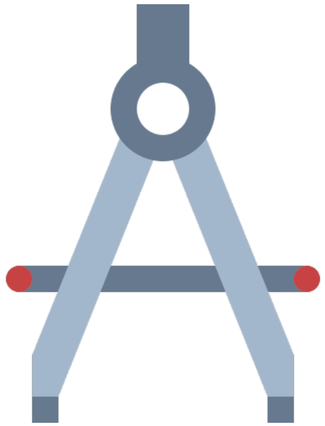
Know the enemy and know yourself,

in a hundred battles you will never be in peril

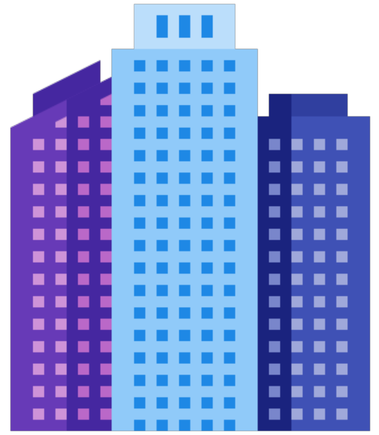
Sun Tzu - *The Art of War*

What and Why

Design

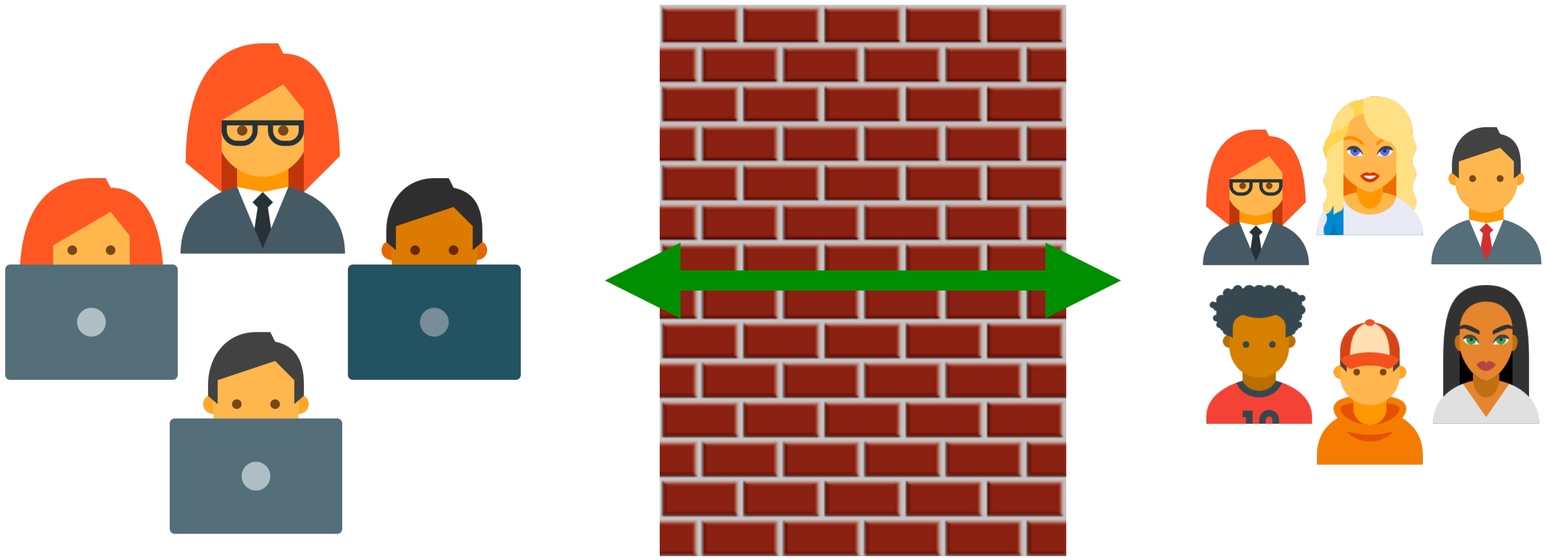


Architecture

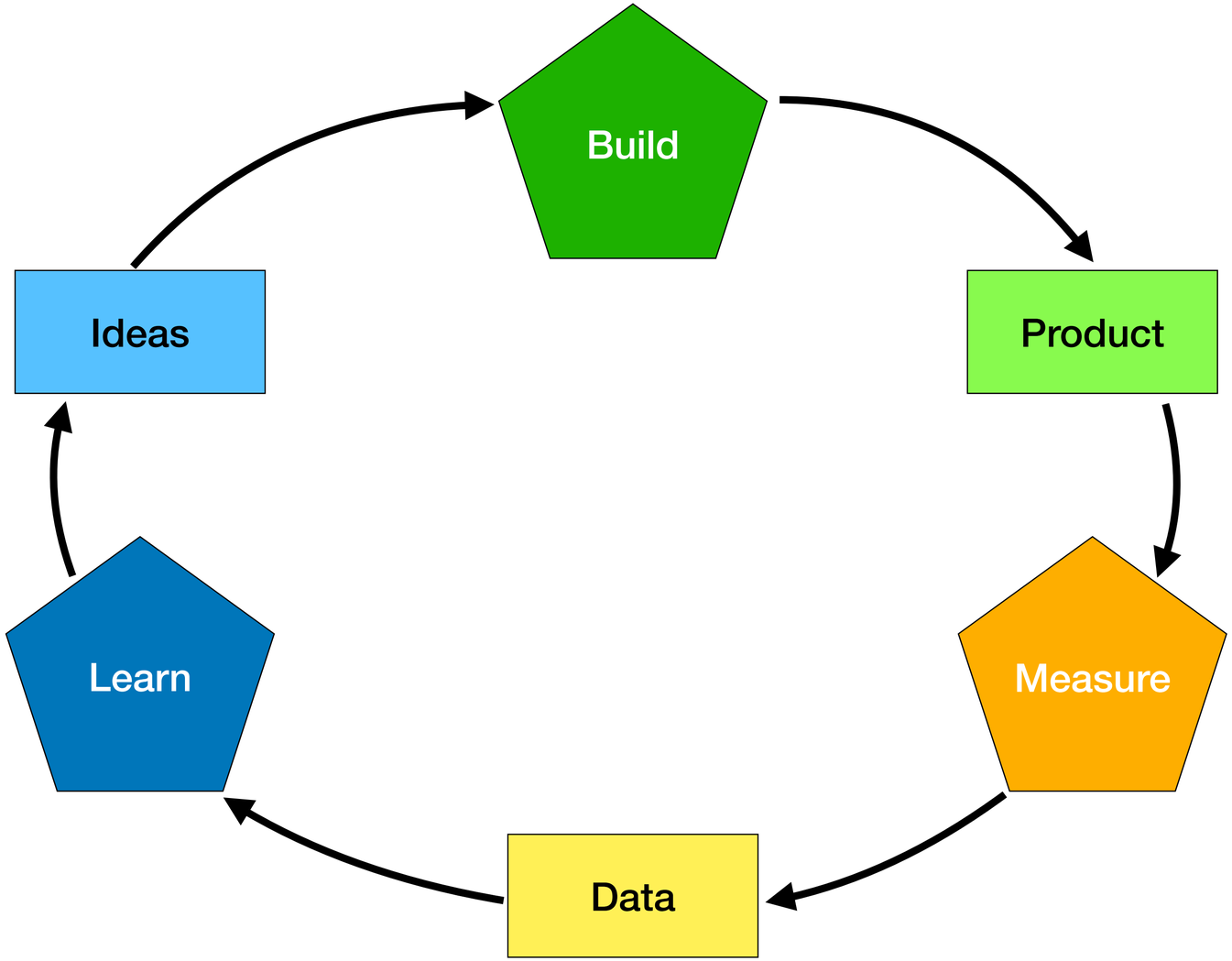


What and Why

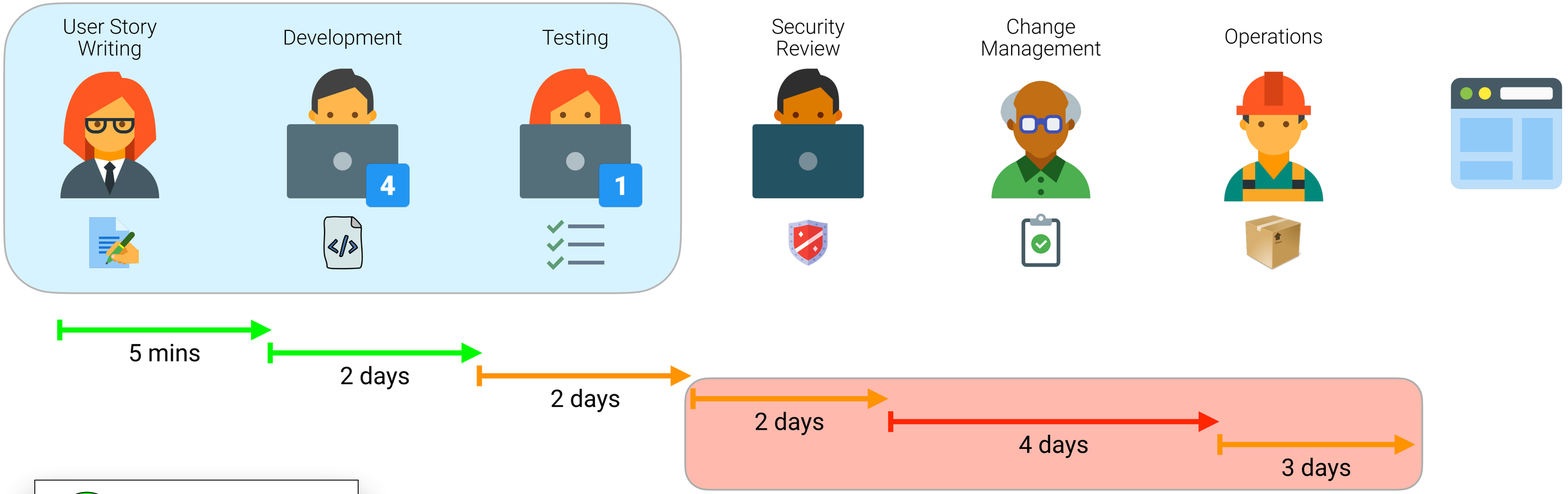
Know your customer



Lean Manufacturing



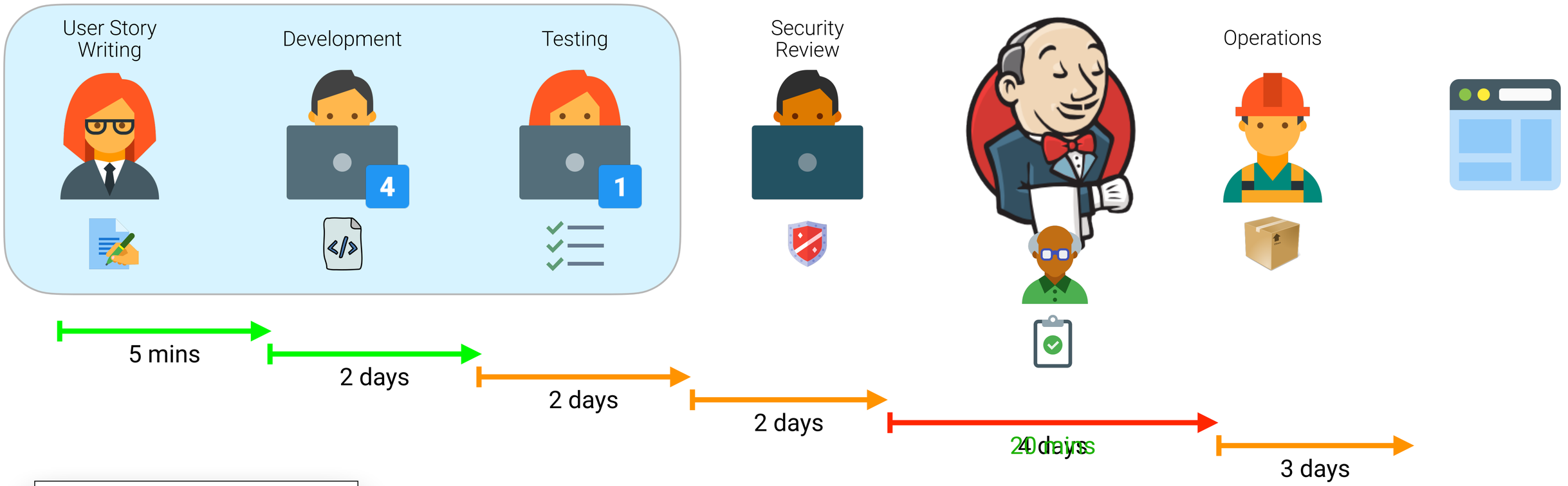
Value Stream Mapping



- Value Added
- Necessary
- Waste

Lead Time: ~13 days

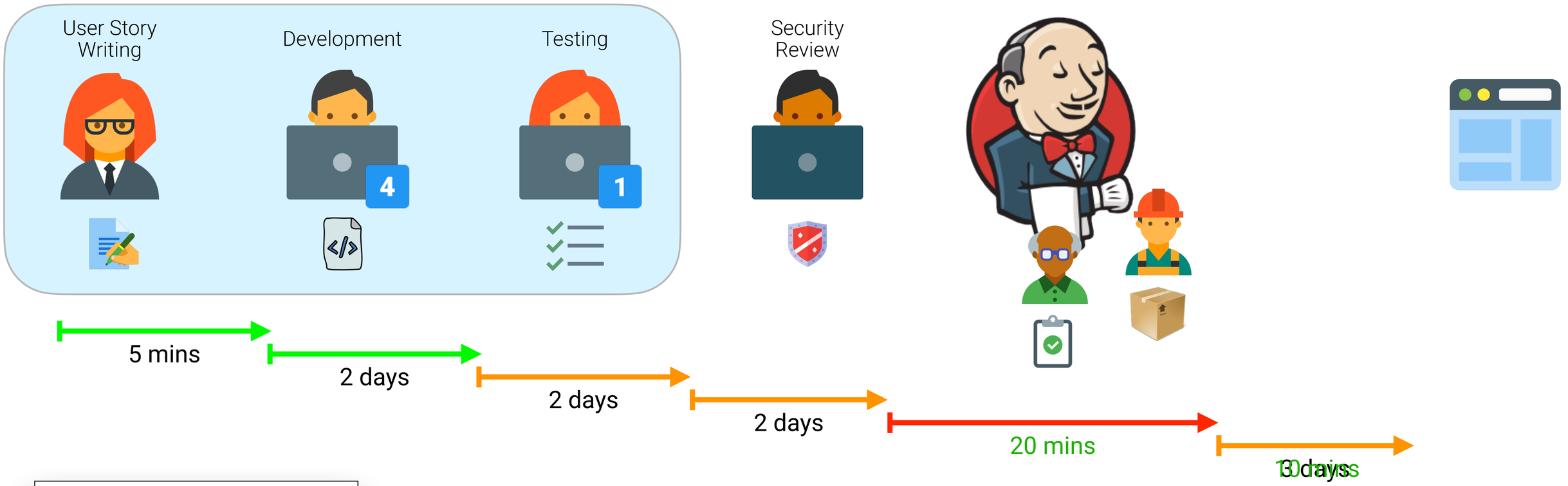
Value Stream Mapping



- Value Added
- Necessary
- Waste

Lead Time: ~13 days

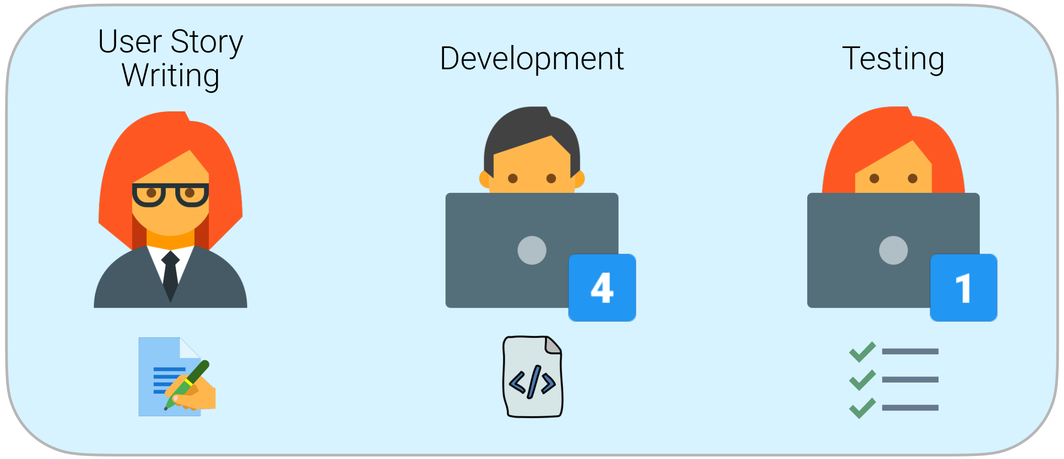
Value Stream Mapping



- Value Added
- Necessary
- Waste

Lead Time: ~13 days

Value Stream Mapping



5 mins

2 days

2 days

30 mins

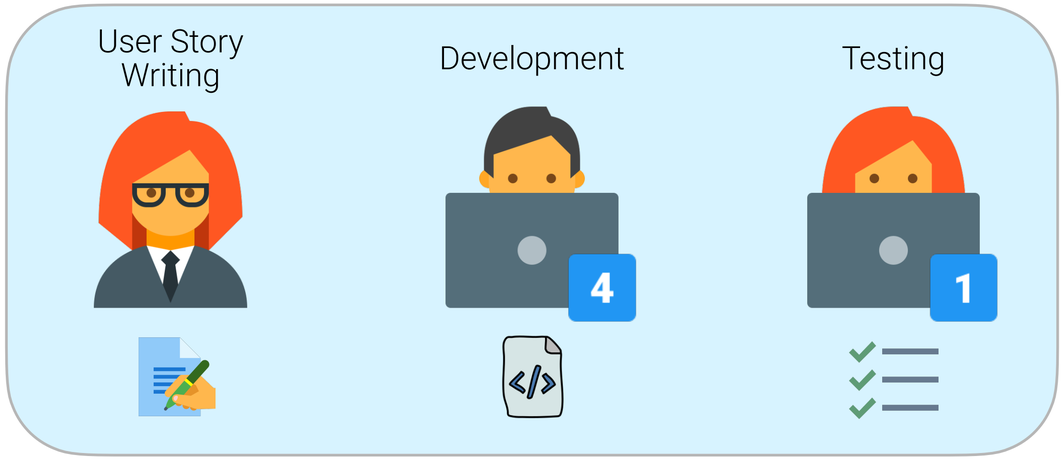
20 mins

10 mins

- Value Added
- Necessary
- Waste

Lead Time: ~13 days

Value Stream Mapping



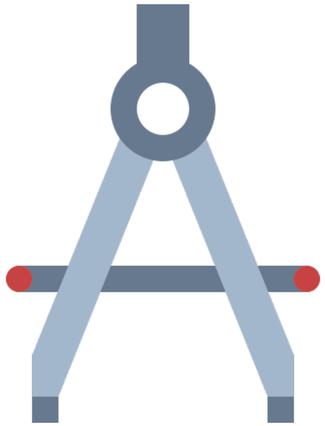
60 mins

- Value Added
- Necessary
- Waste

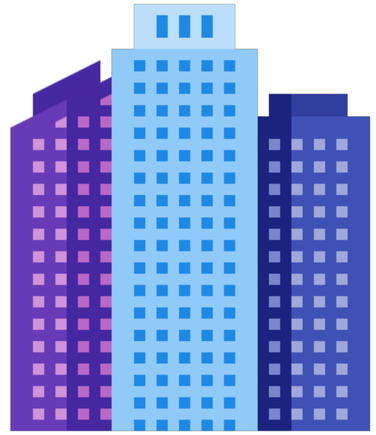
Lead Time: ~43days

What and Why

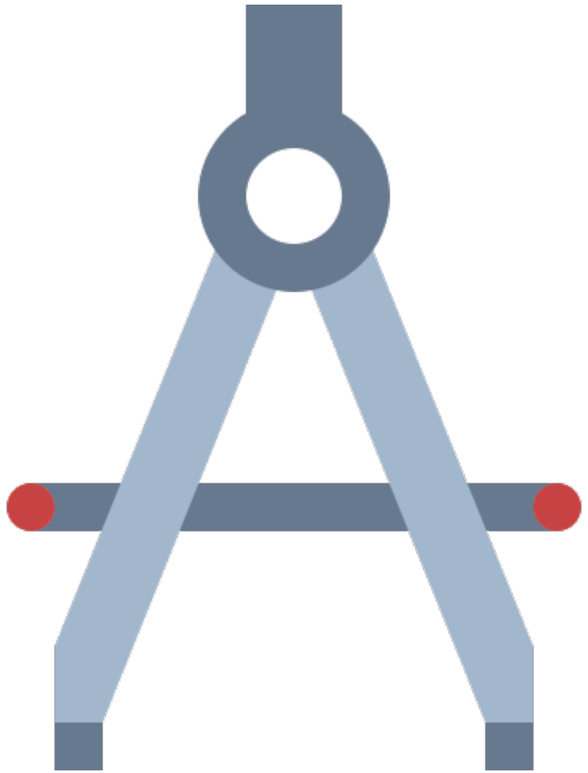
Design



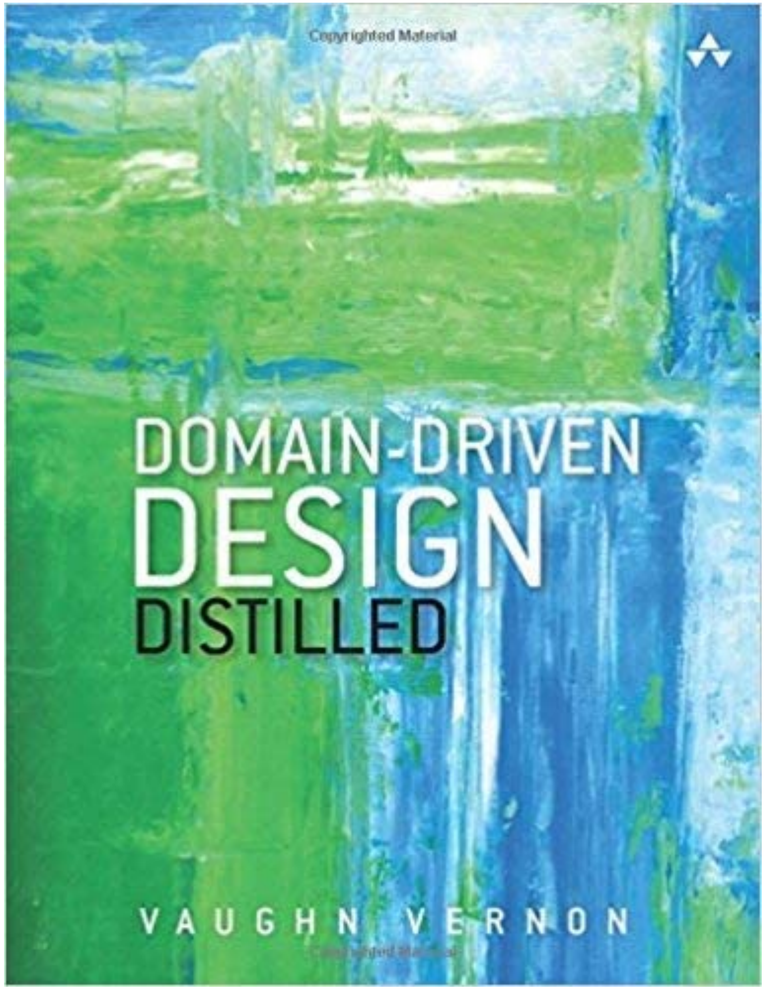
Architecture



Design



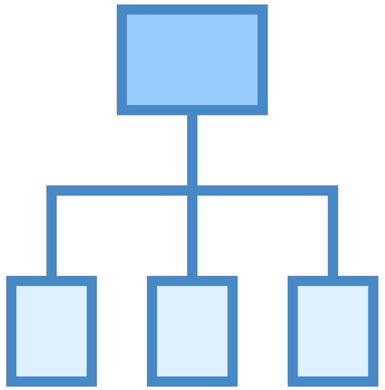
Domain Driven Design



- Strategic Design
- High Level Component View
- Great for distributed architectures

Strategic Design

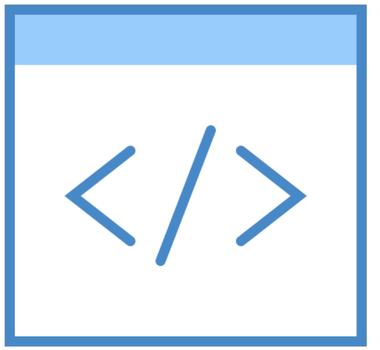
Ubiquitous Language



Diagrams

Documentation

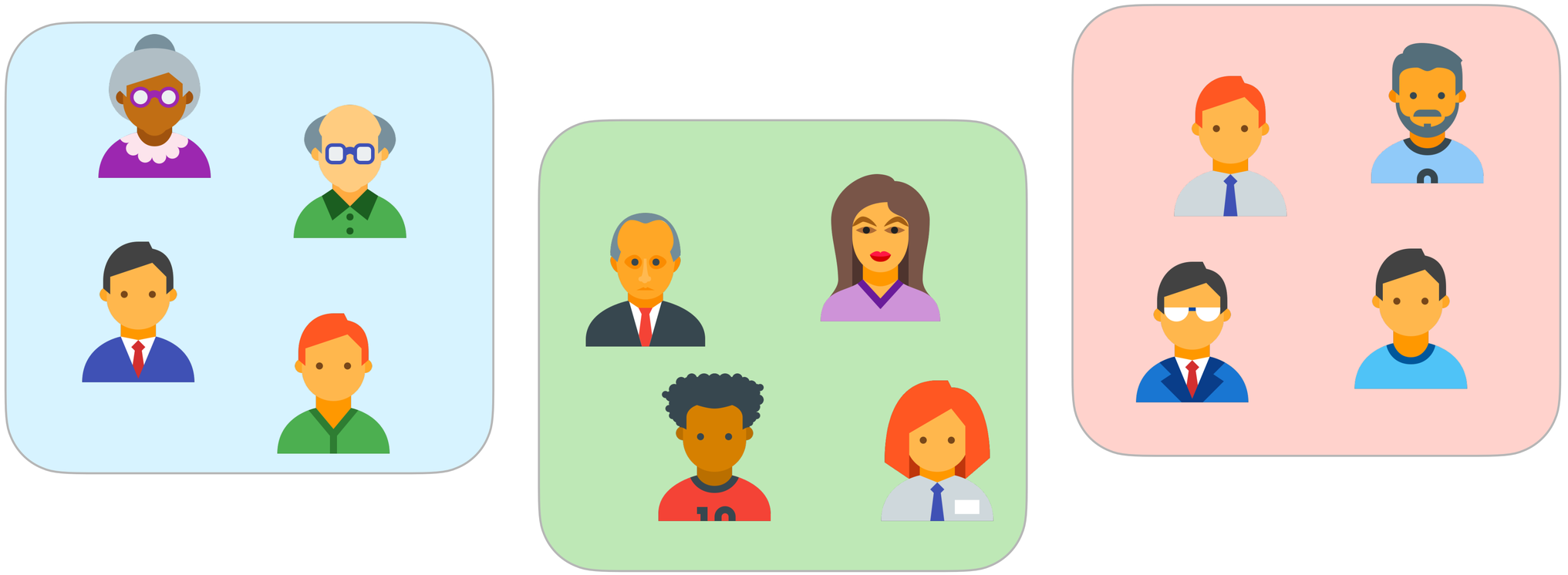
User Stories



Code

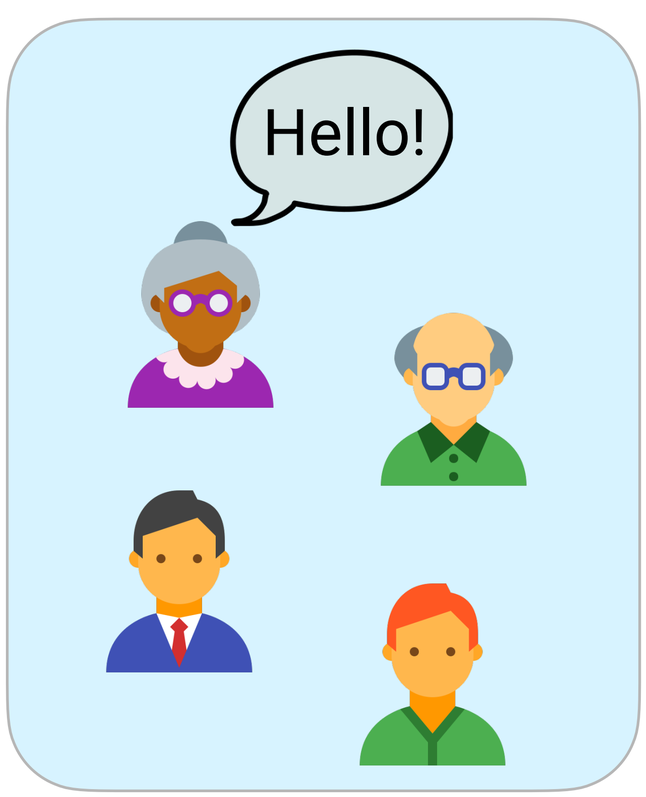
Strategic Design

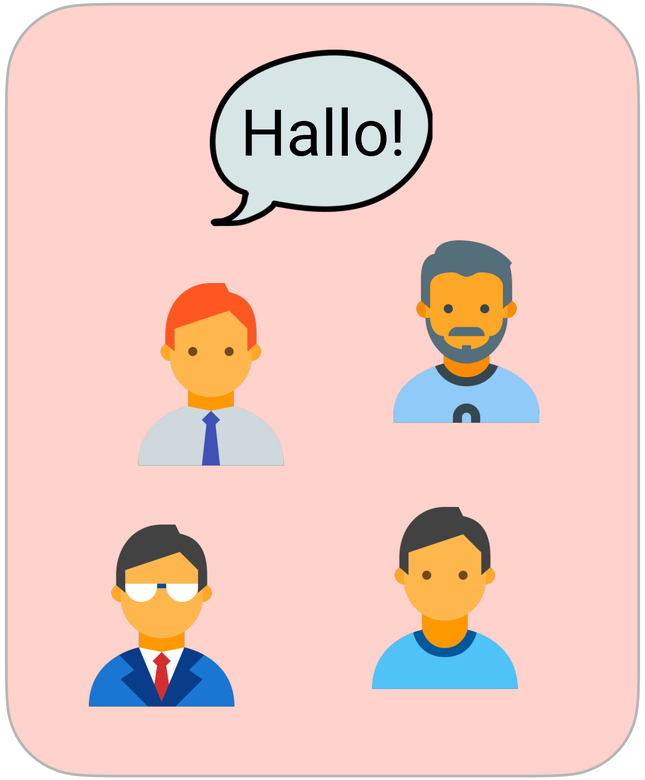
Bounded Context



Strategic Design

Bounded Context





Strategic Design

Bounded Context

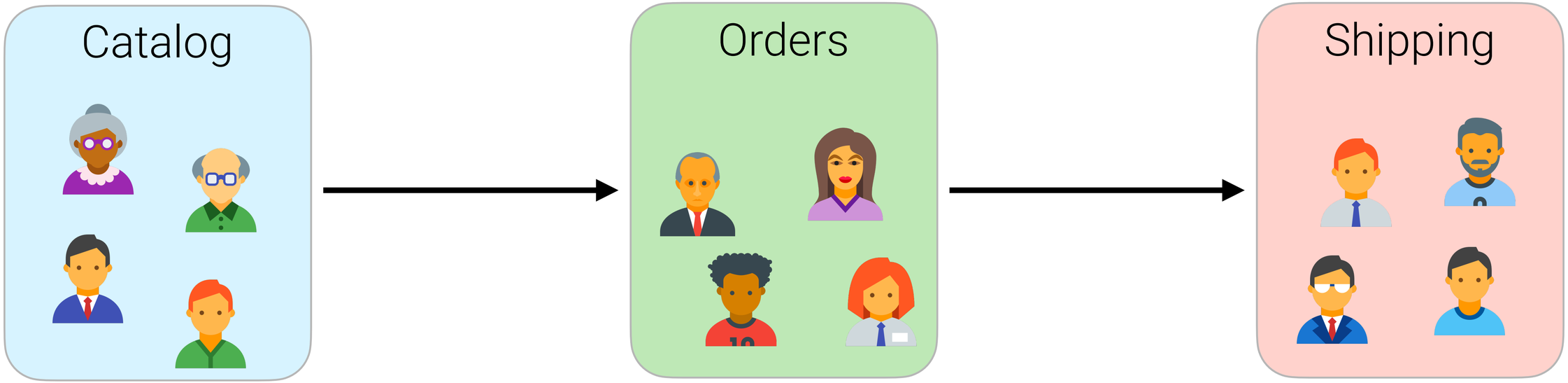
Catalog

Orders

Shipping

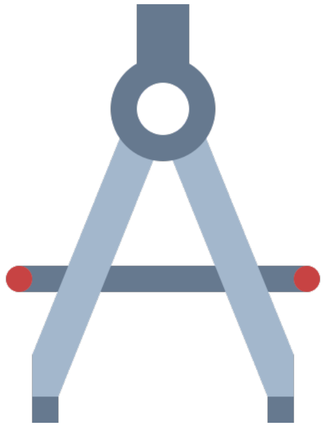
Strategic Design

Context Mapping

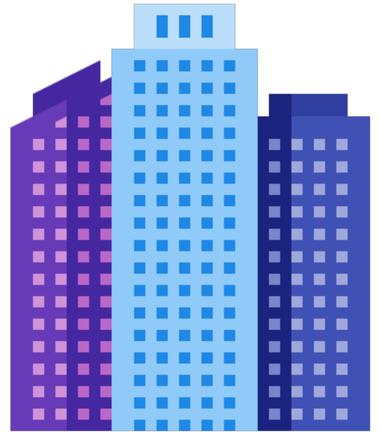


What? and Why?

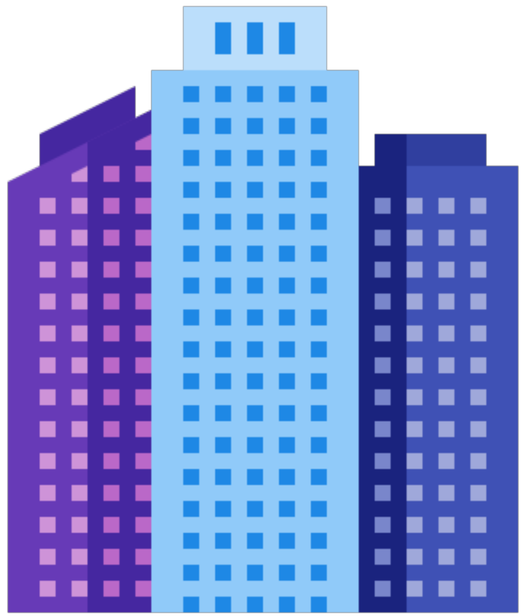
Design



Architecture

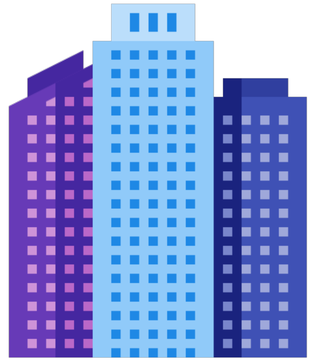


Architecture



Choose an architecture style

- **Strategic** Goals
- **Domain** considerations
- Trade-offs for **architecture characteristics**



Neal Ford / Mark Richards
Fundamentals of Software Architecture

Step 1

Choose Architectural Characteristics

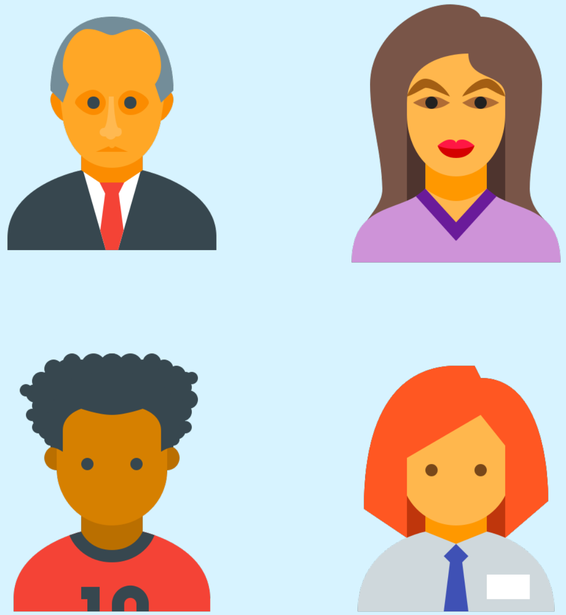
Architectural Characteristics

Scalability
Usability
Reliability
Agility
Portability
Durability
Resilience
Traceability
Security
Evolvability

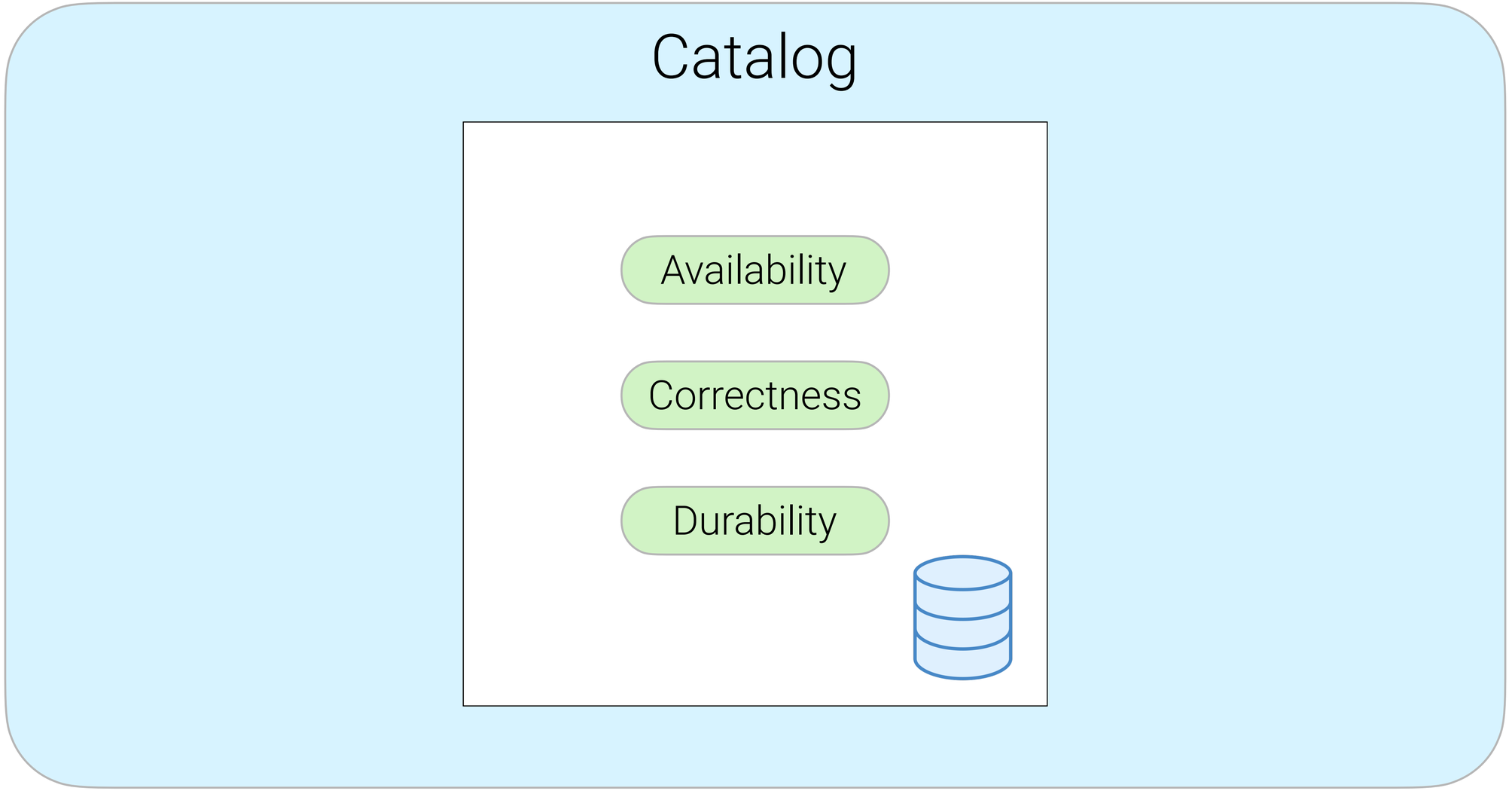
Step 2

Monolithic vs Distributed

Catalog

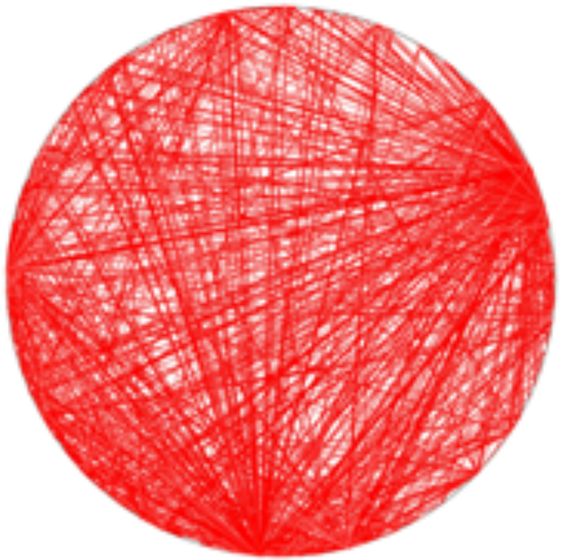


Monolithic vs Distributed

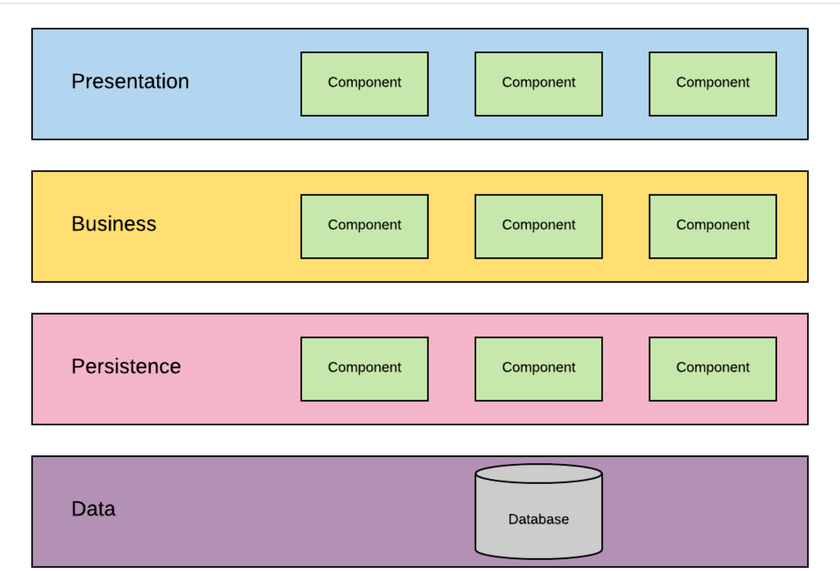


Monolithic Architecture Styles

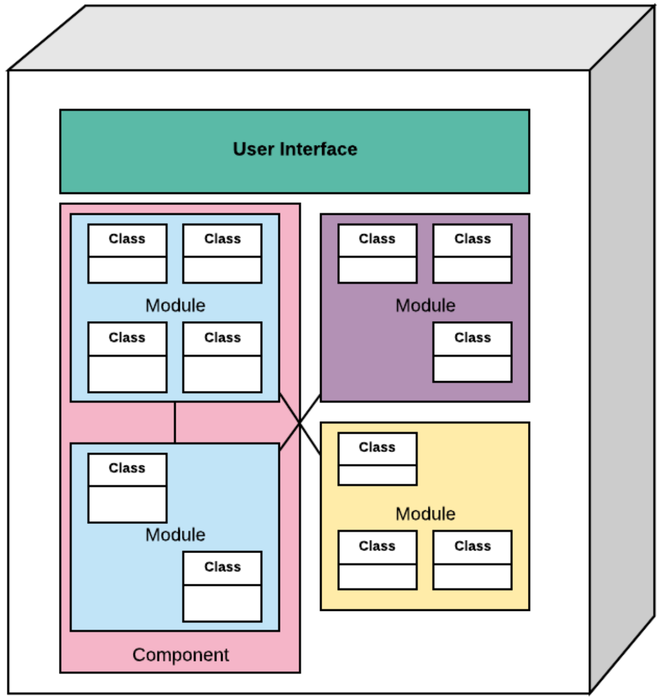
Big Ball of Mud



Layered Architecture

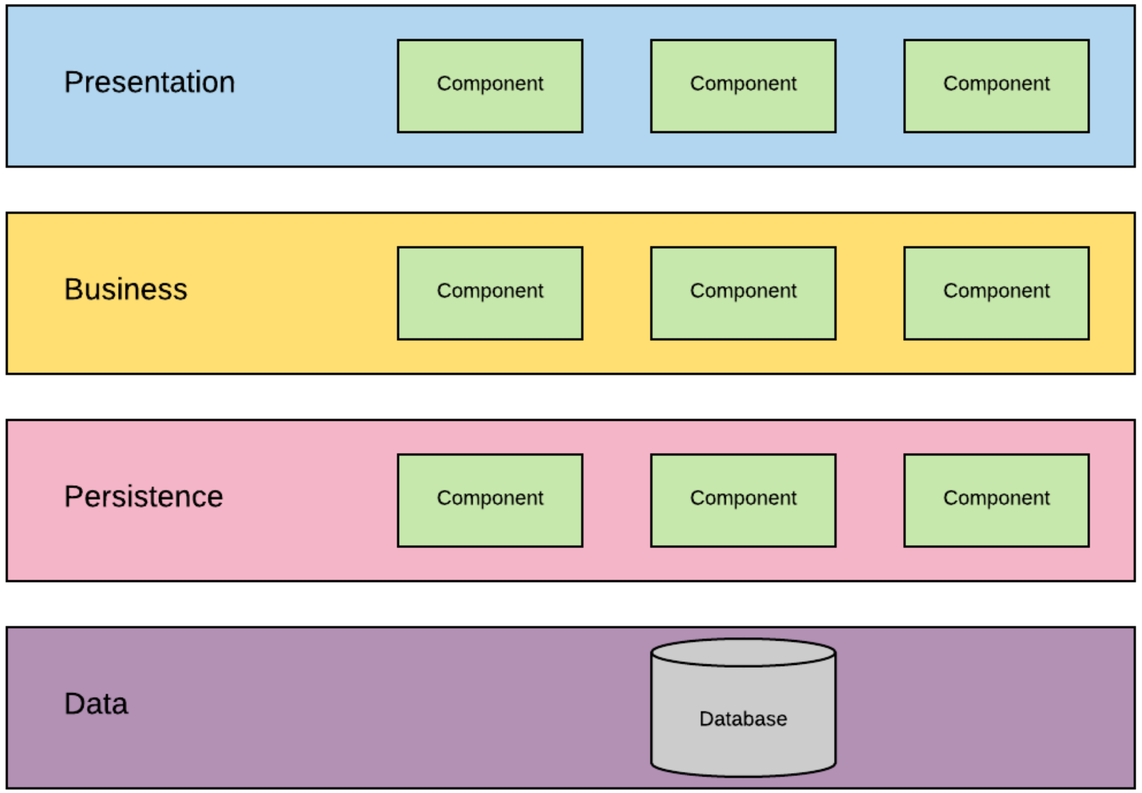


Modular Monolith

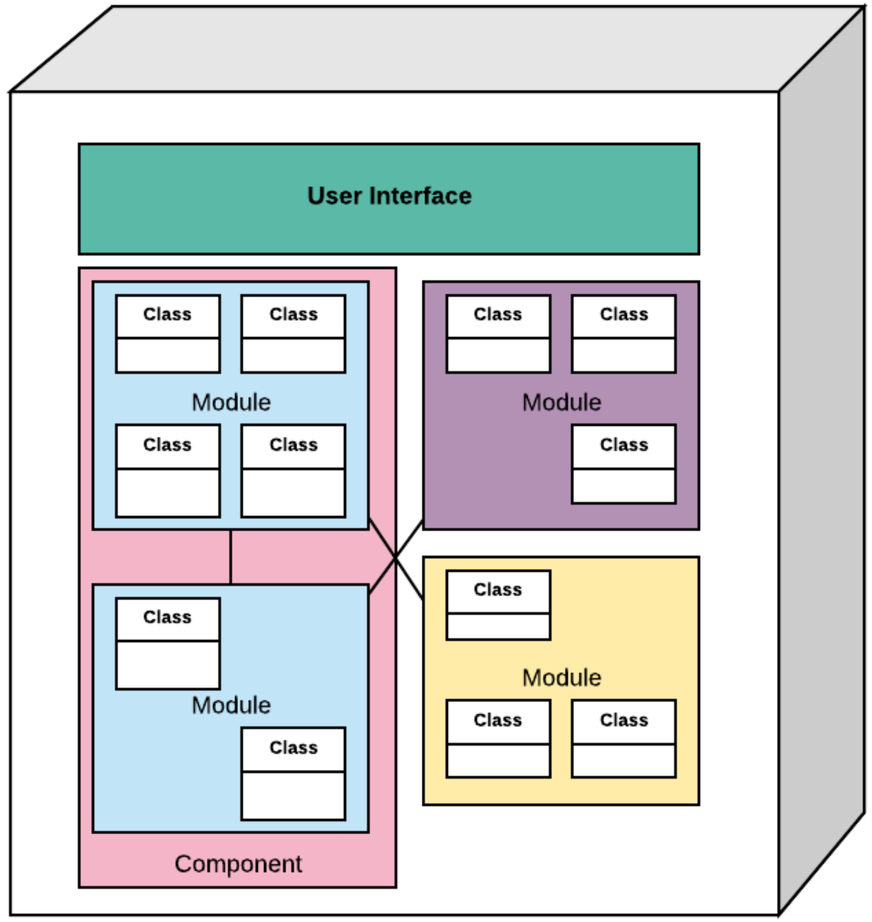


Monolithic Architecture Styles

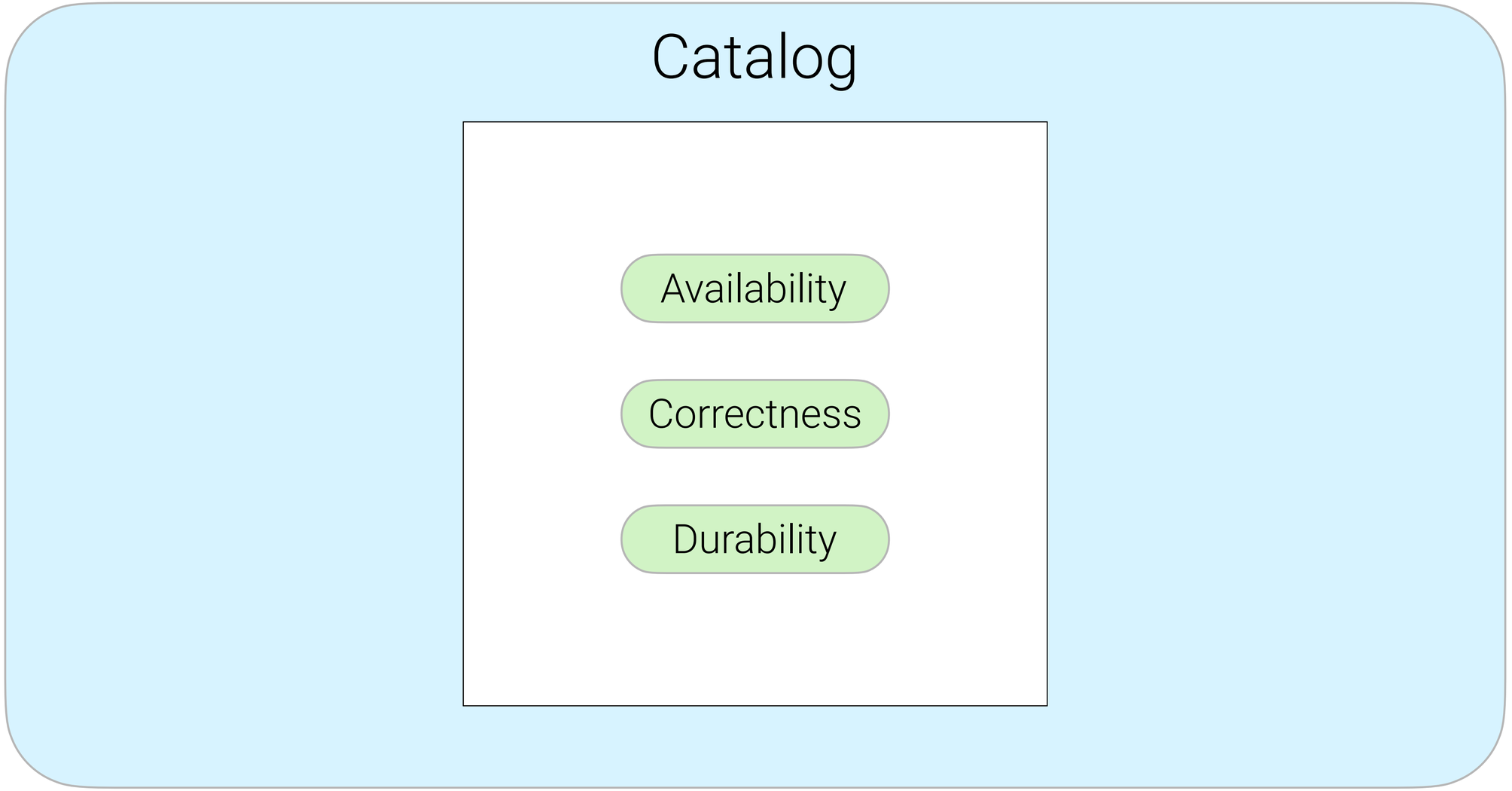
Layered Architecture

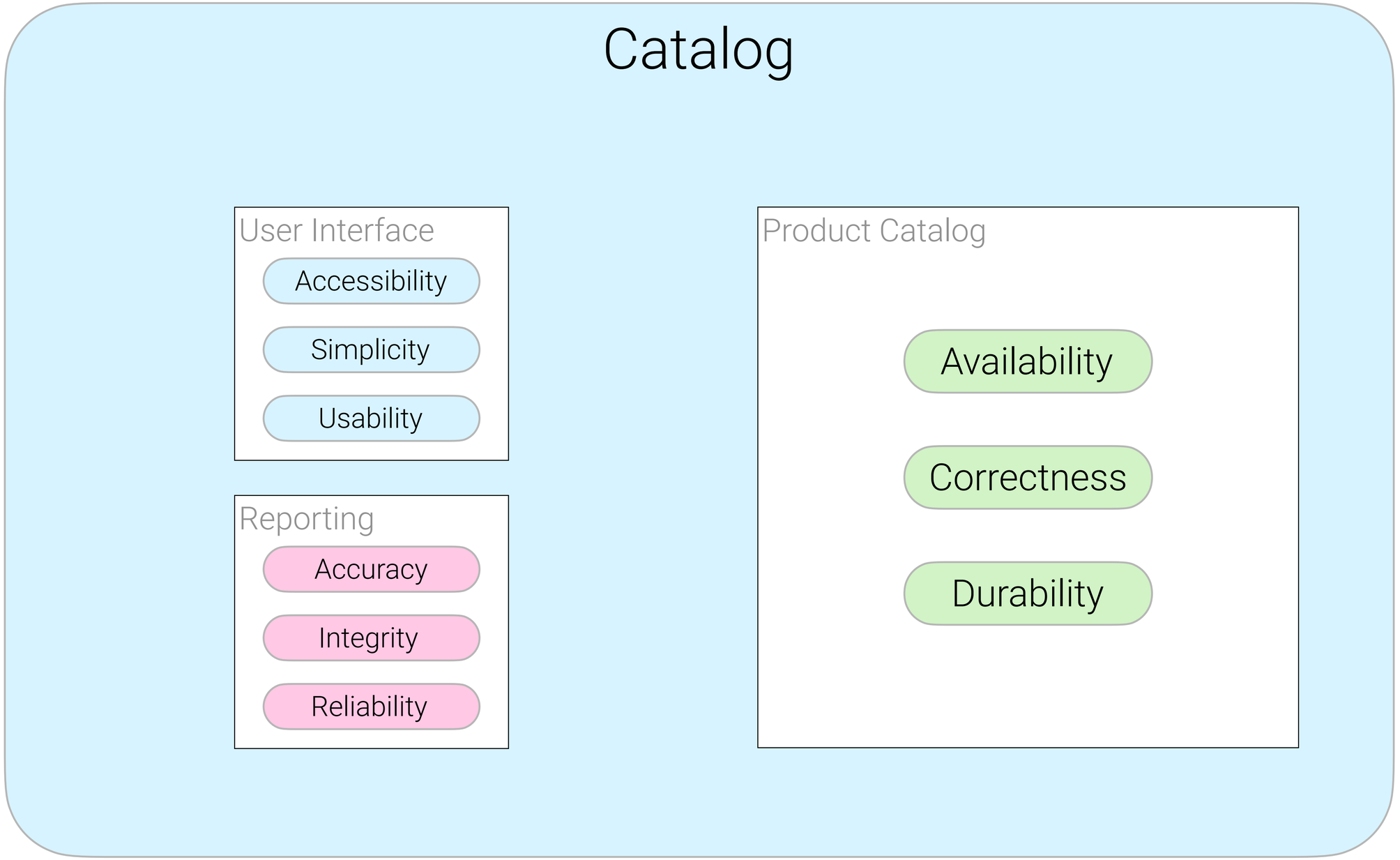


Modular Monolith



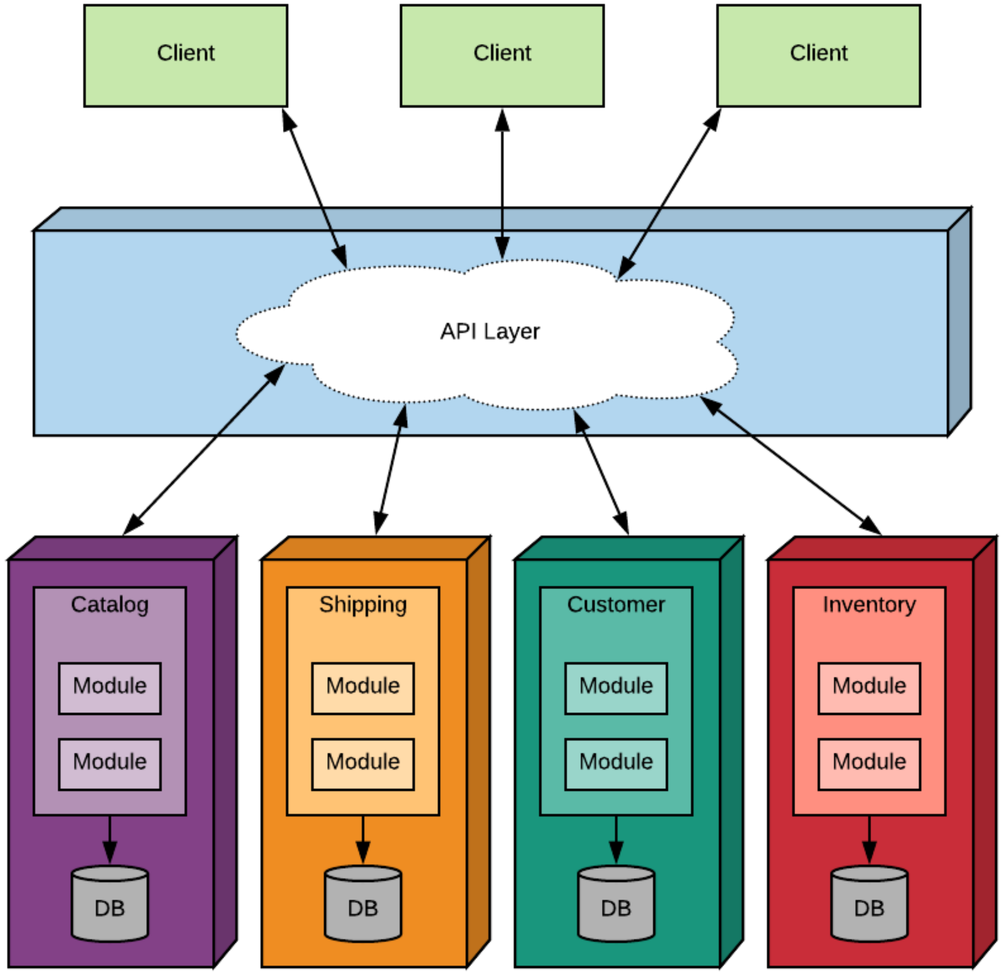
Monolithic Architecture Styles



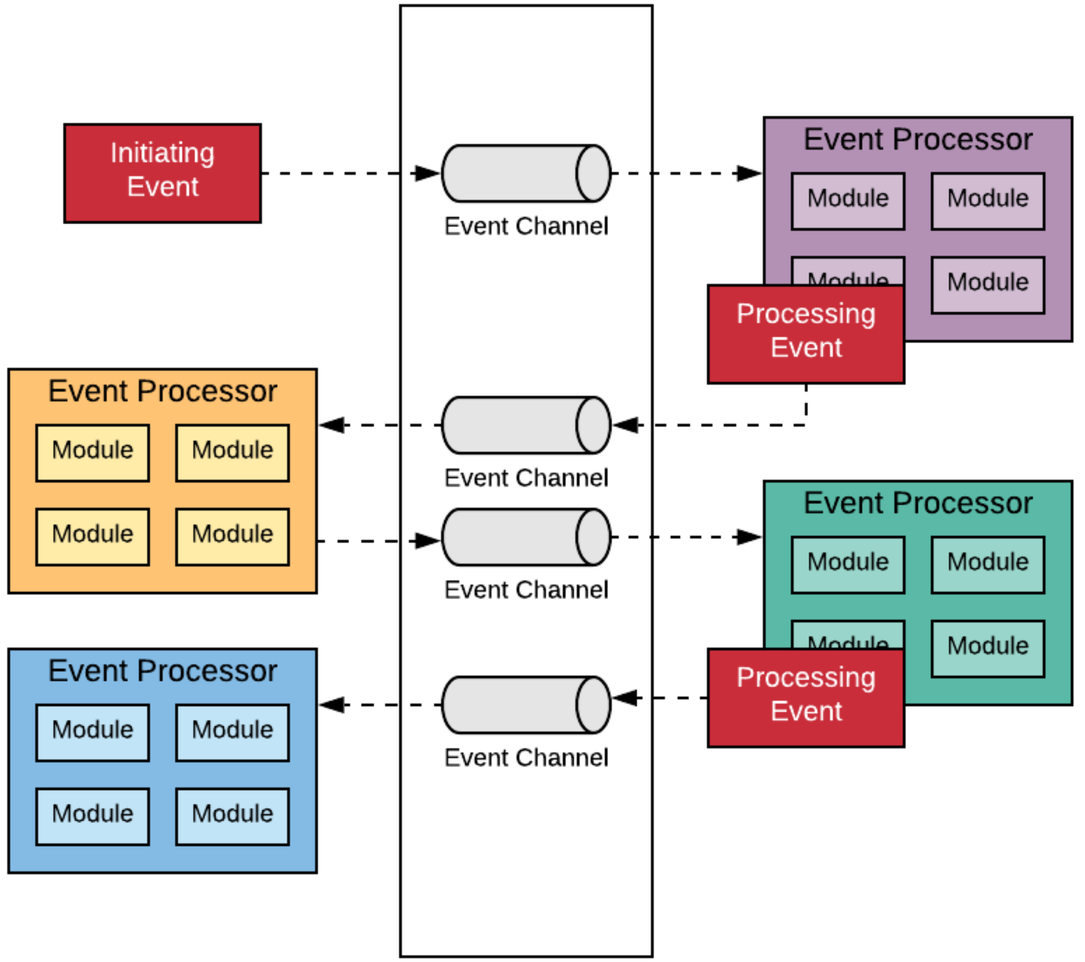

Catalog
User Interface
Accessibility
Simplicity
Usability
Reporting
Accuracy
Integrity
Reliability
Product Catalog
Availability
Correctness
Durability

Distributed Architecture Styles

Microservices Architecture

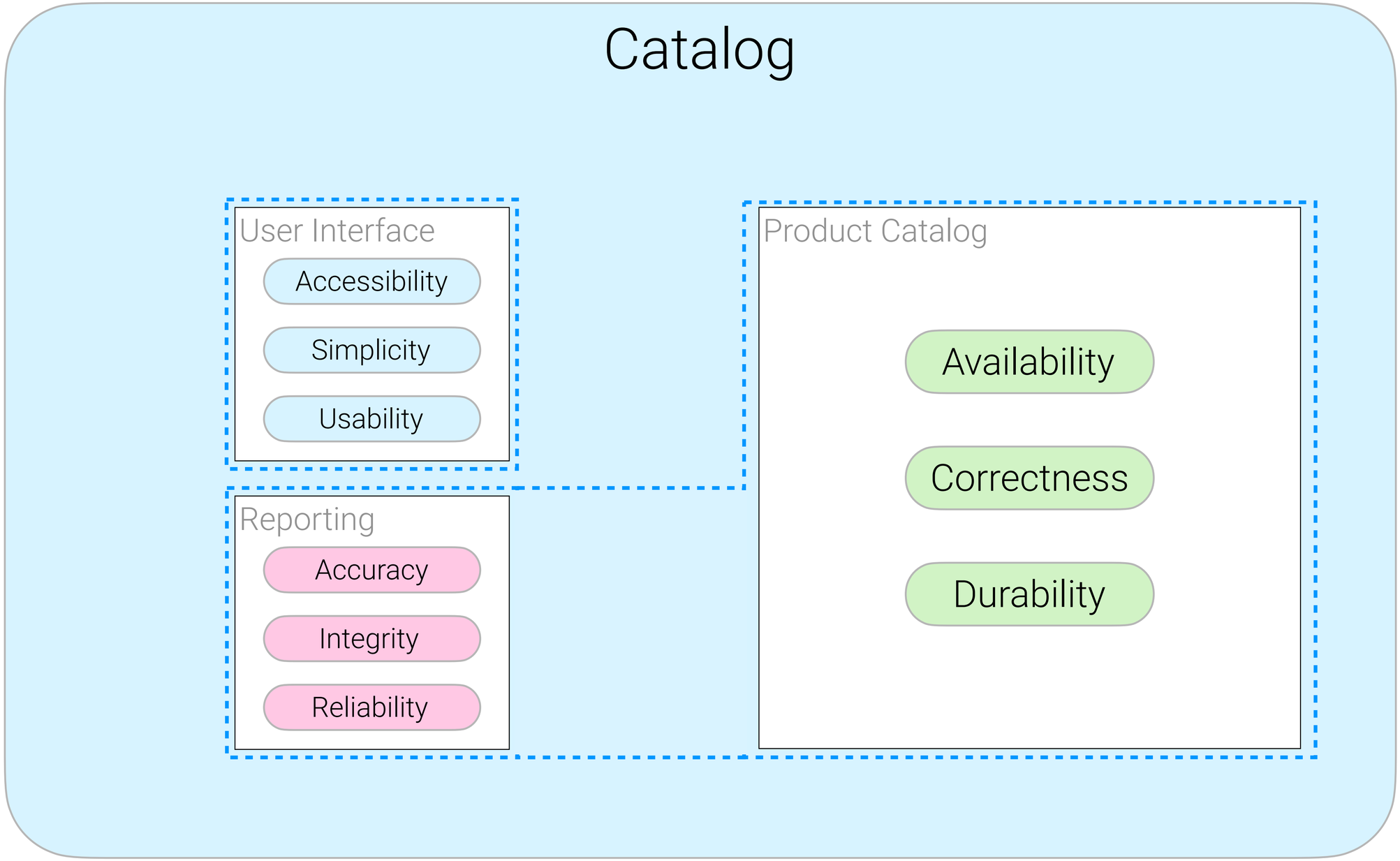


Event-Driven Architecture



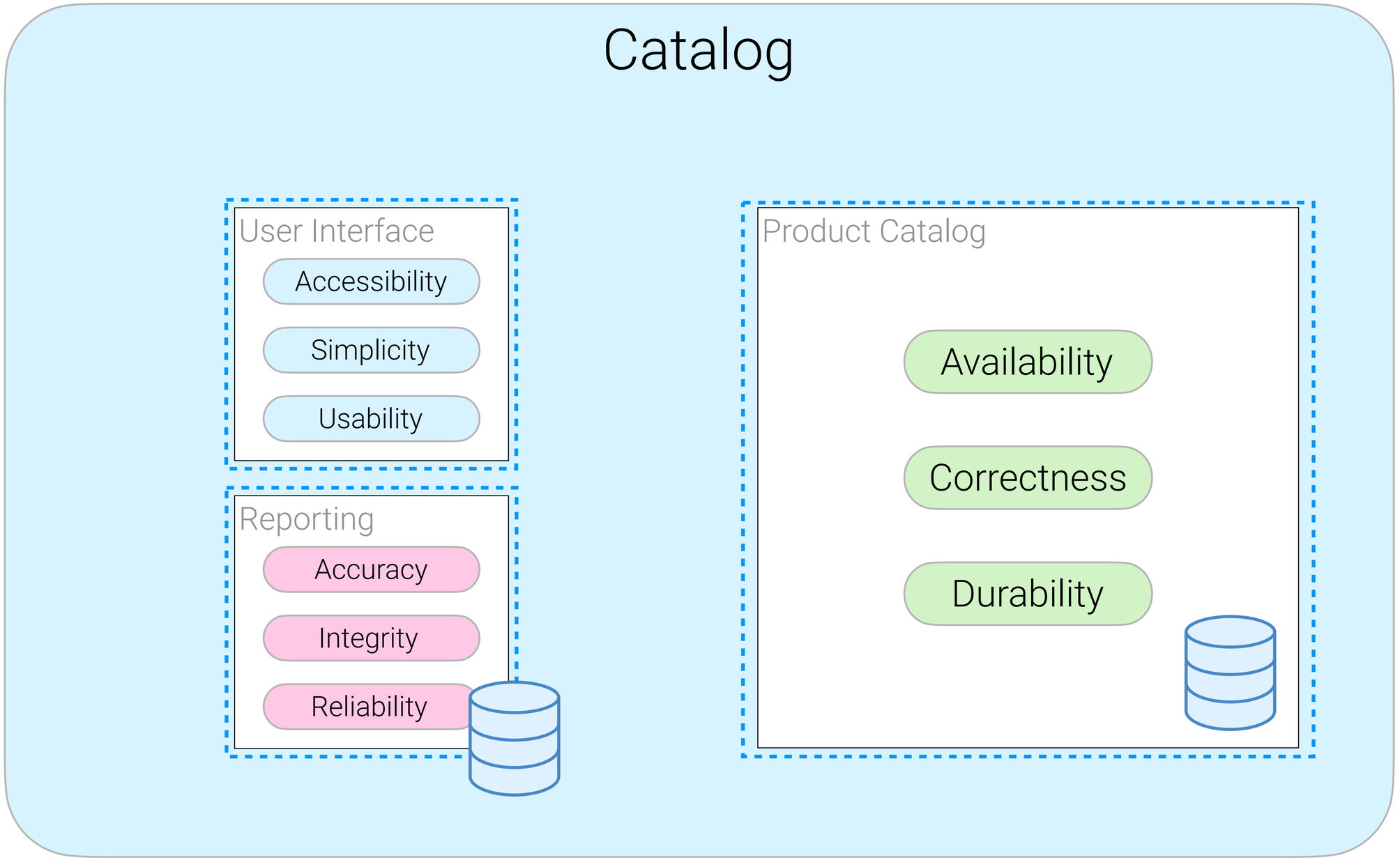
Step 3

Determine Architecture Quantum


Catalog
User Interface
Accessibility
Simplicity
Usability
Reporting
Accuracy
Integrity
Reliability
Product Catalog
Availability
Correctness
Durability

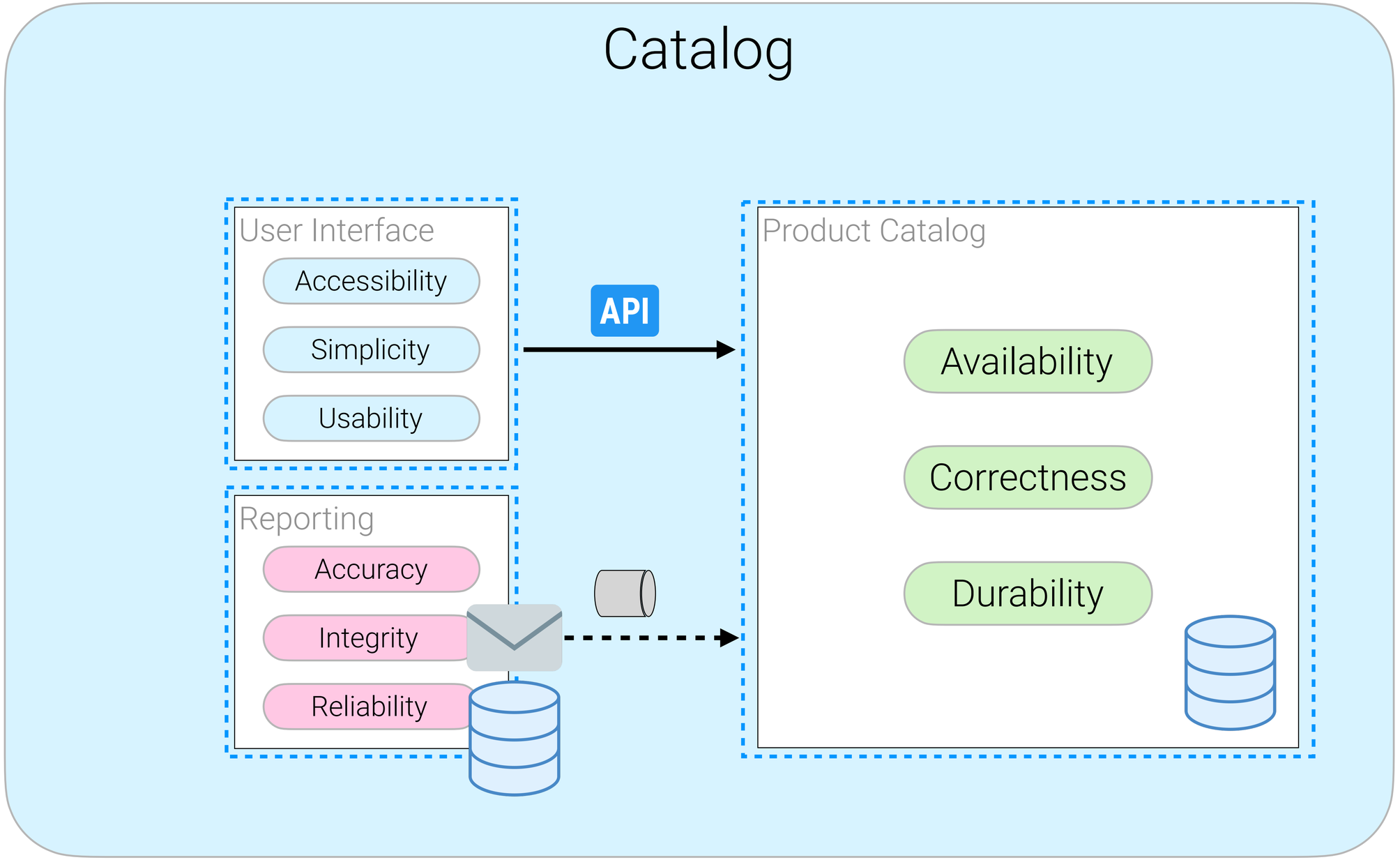
Step 4

Determine Persistence


Catalog
User Interface
Accessibility
Simplicity
Usability
Reporting
Accuracy
Integrity
Reliability
Product Catalog
Availability
Correctness
Durability

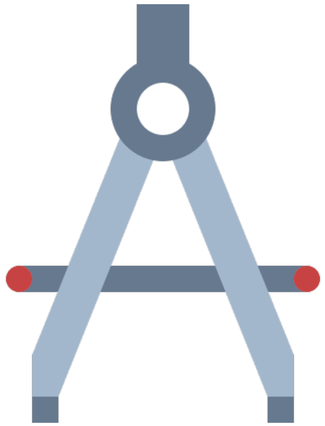
Step 5

Determine Communication Style

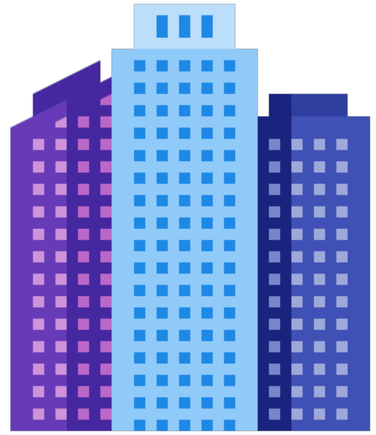

Catalog
User Interface
Accessibility
Simplicity
Usability
API
Reporting
Accuracy
Integrity
Reliability
Product Catalog
Availability
Correctness
Durability

What and Why

Design



Architecture



Q & A

 https://abel.dev

 https://github.com/abel-fresnillo

 @abel_fresnillo

 me@abel.dev