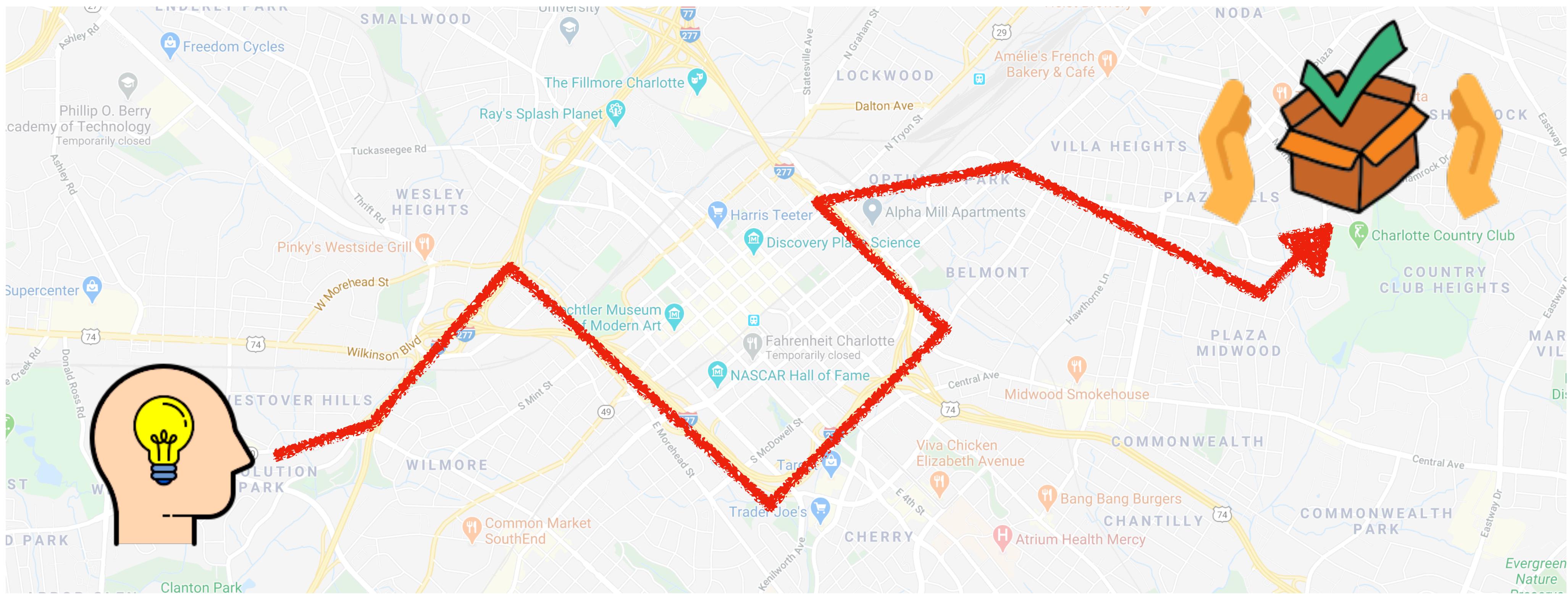


Software Architecture

A story about business value and trade-offs

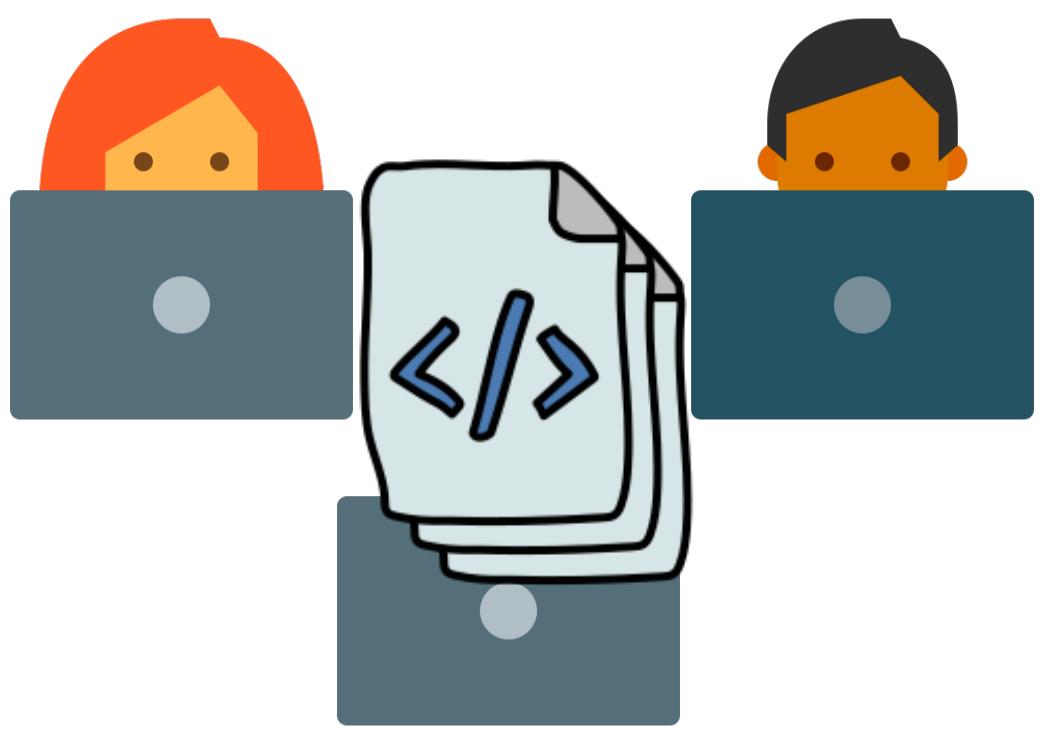


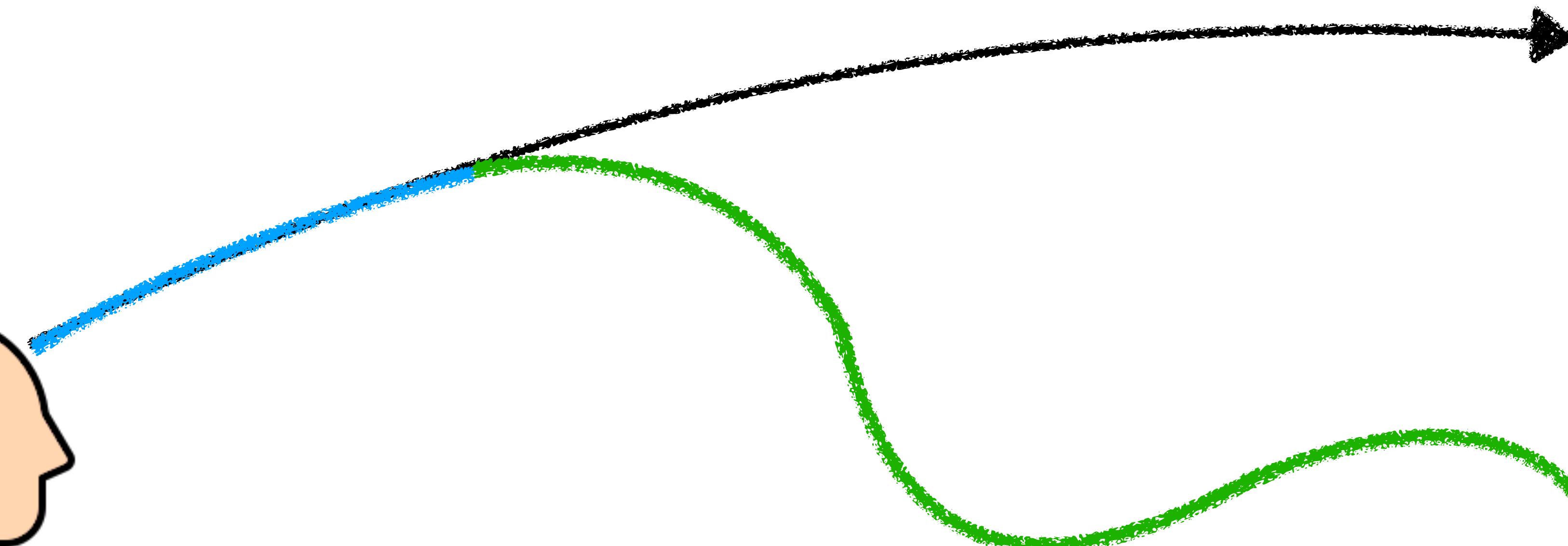
About me

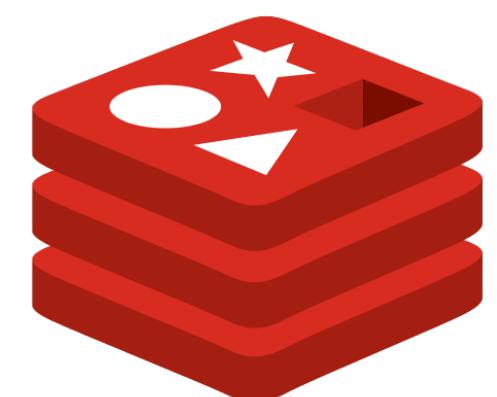


- Senior Software Engineer
- 14 years working in the software industry
- Passionate about software development









Couchbase

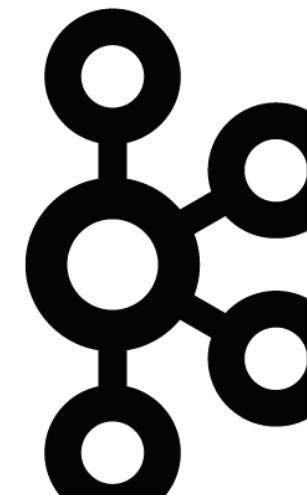
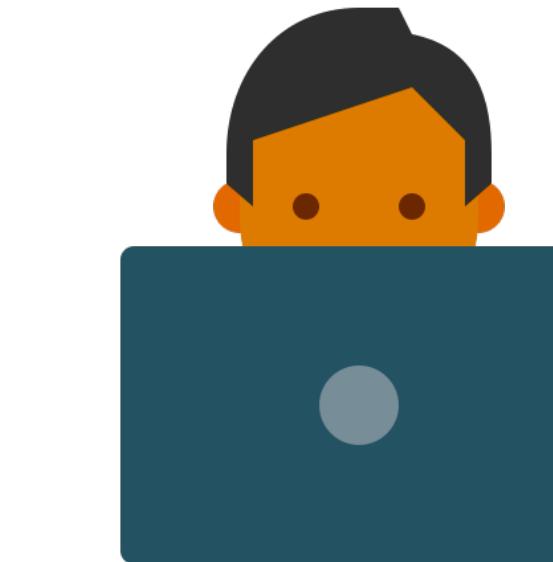


Istio

redis

Azure

Jenkins



neo4j



docker

okta

mongoDB



dynatrace

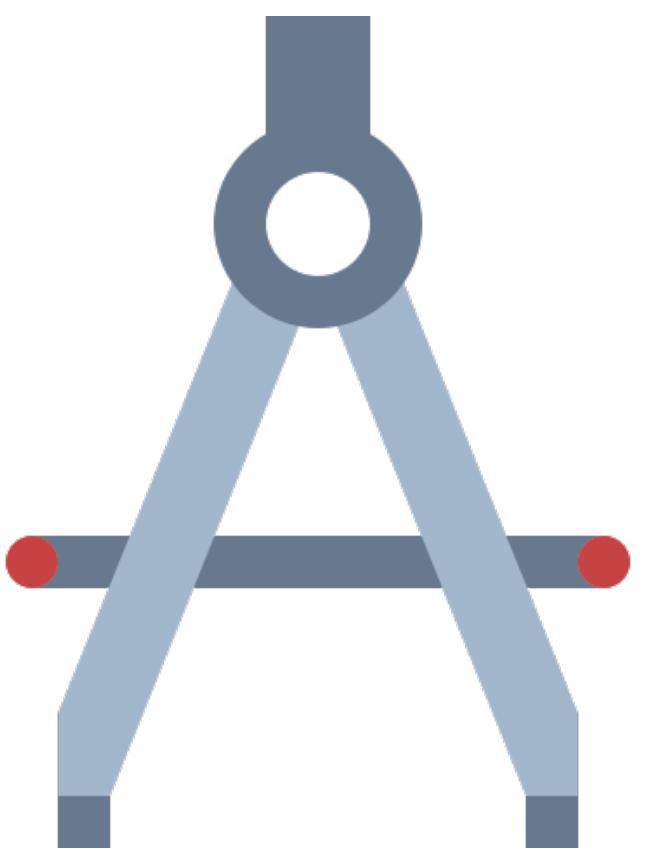
Know the enemy and know yourself,
in a hundred battles you will never be in peril

Sun Tzu - *The Art of War*

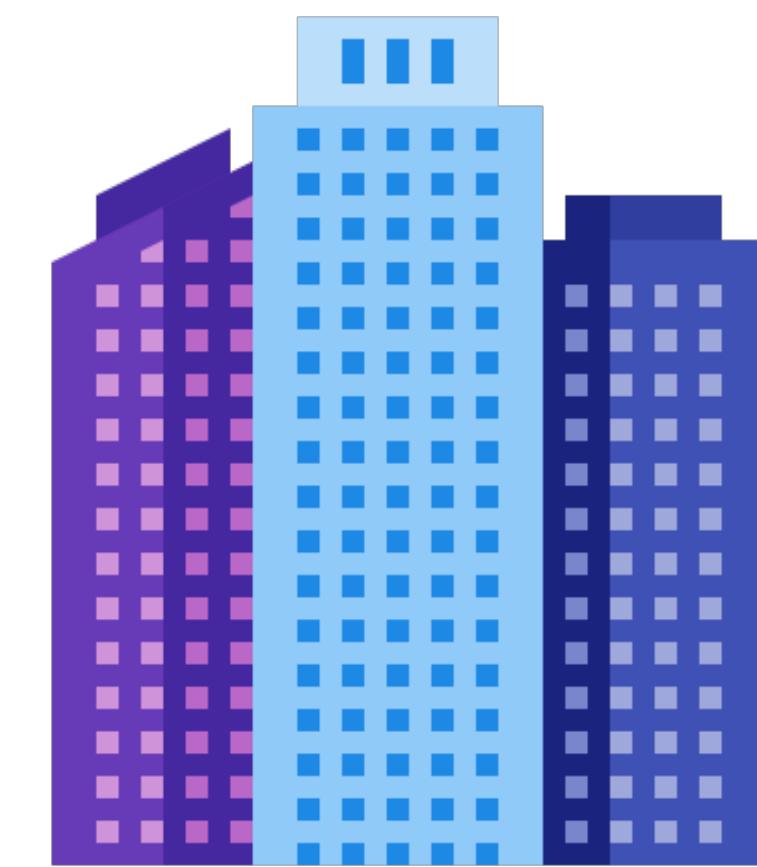
What to Build



Design



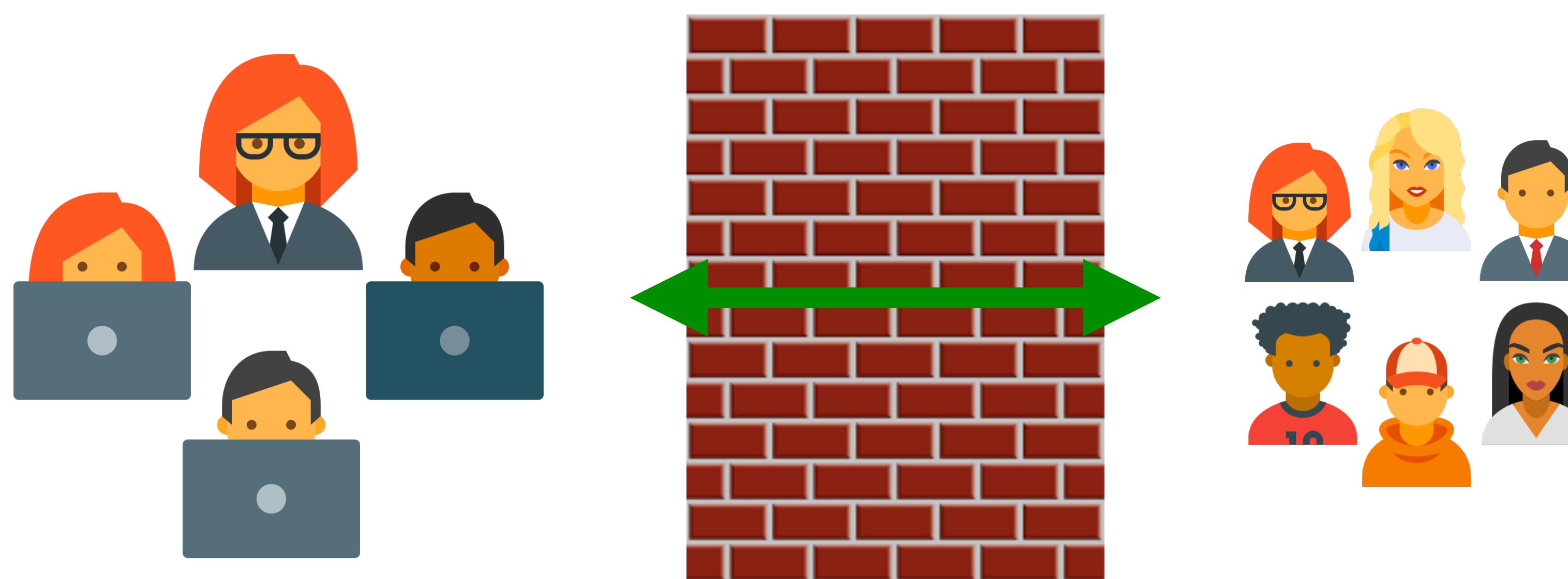
Architecture



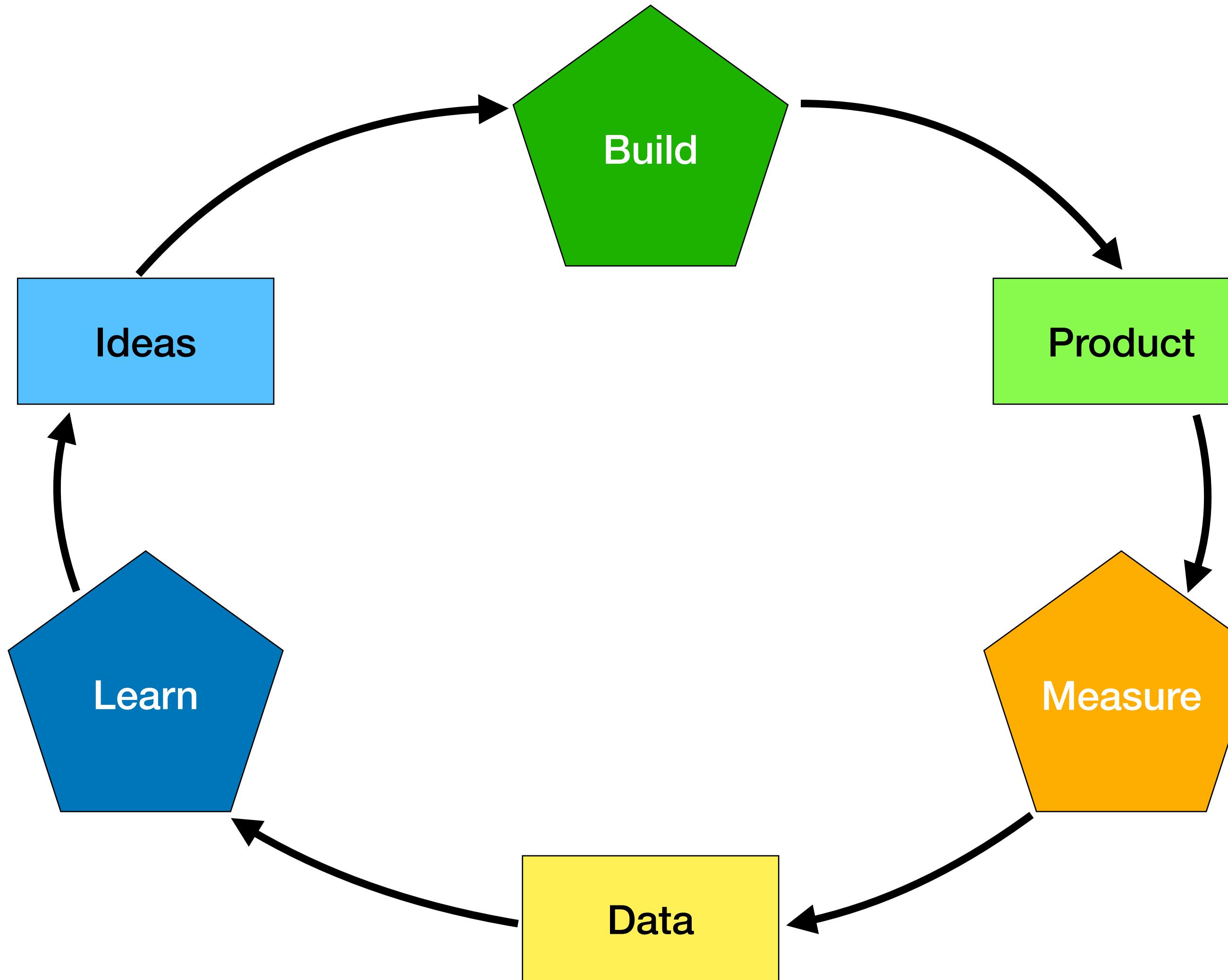
What to Build



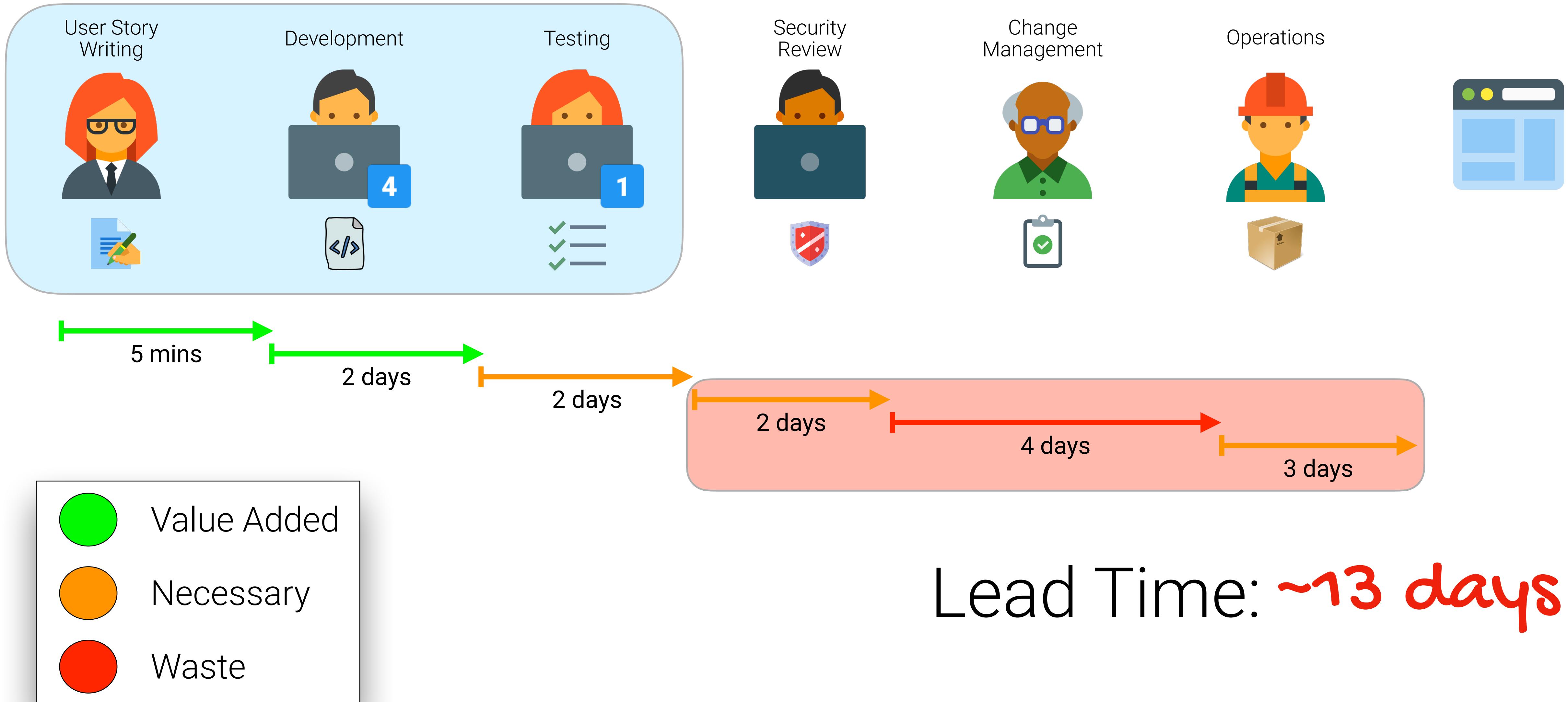
Know your customer



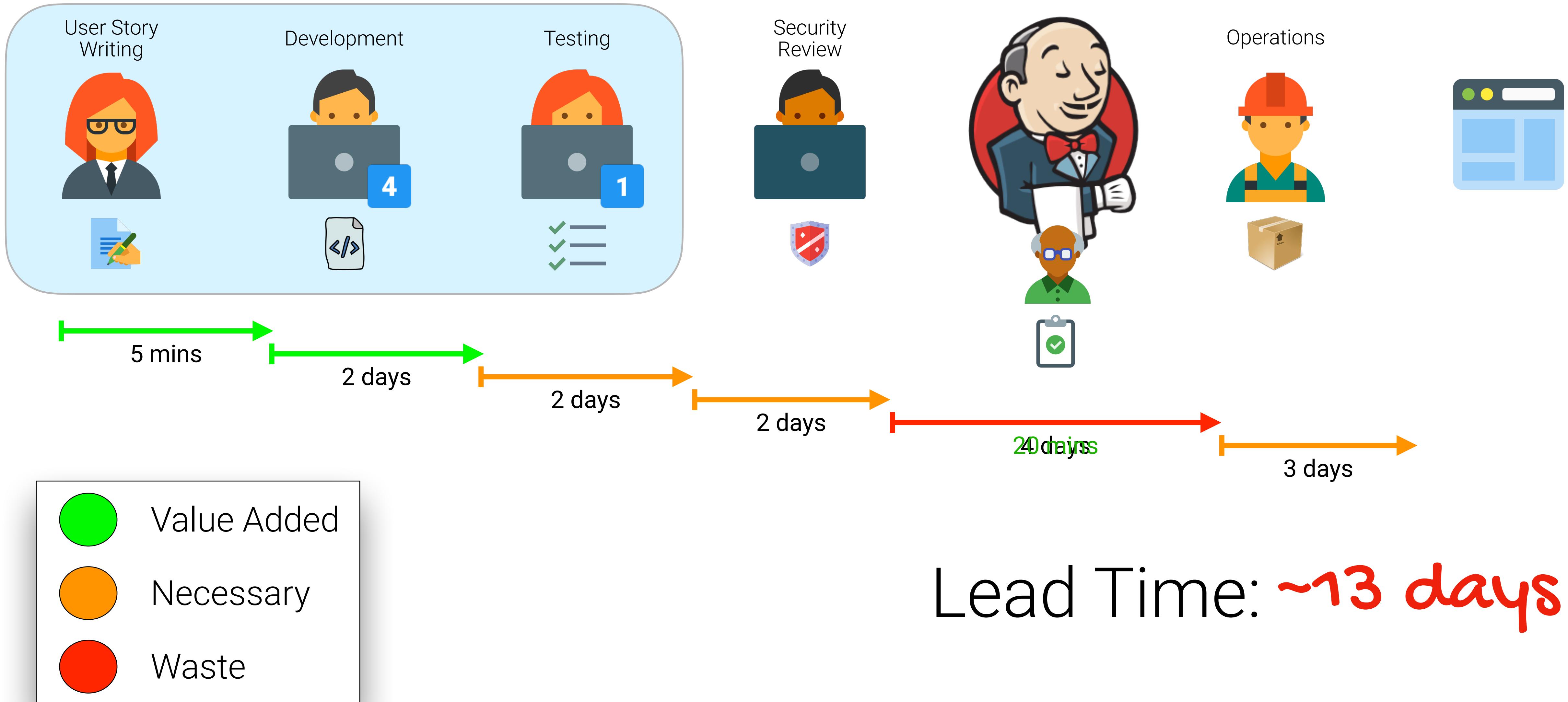
Lean Manufacturing



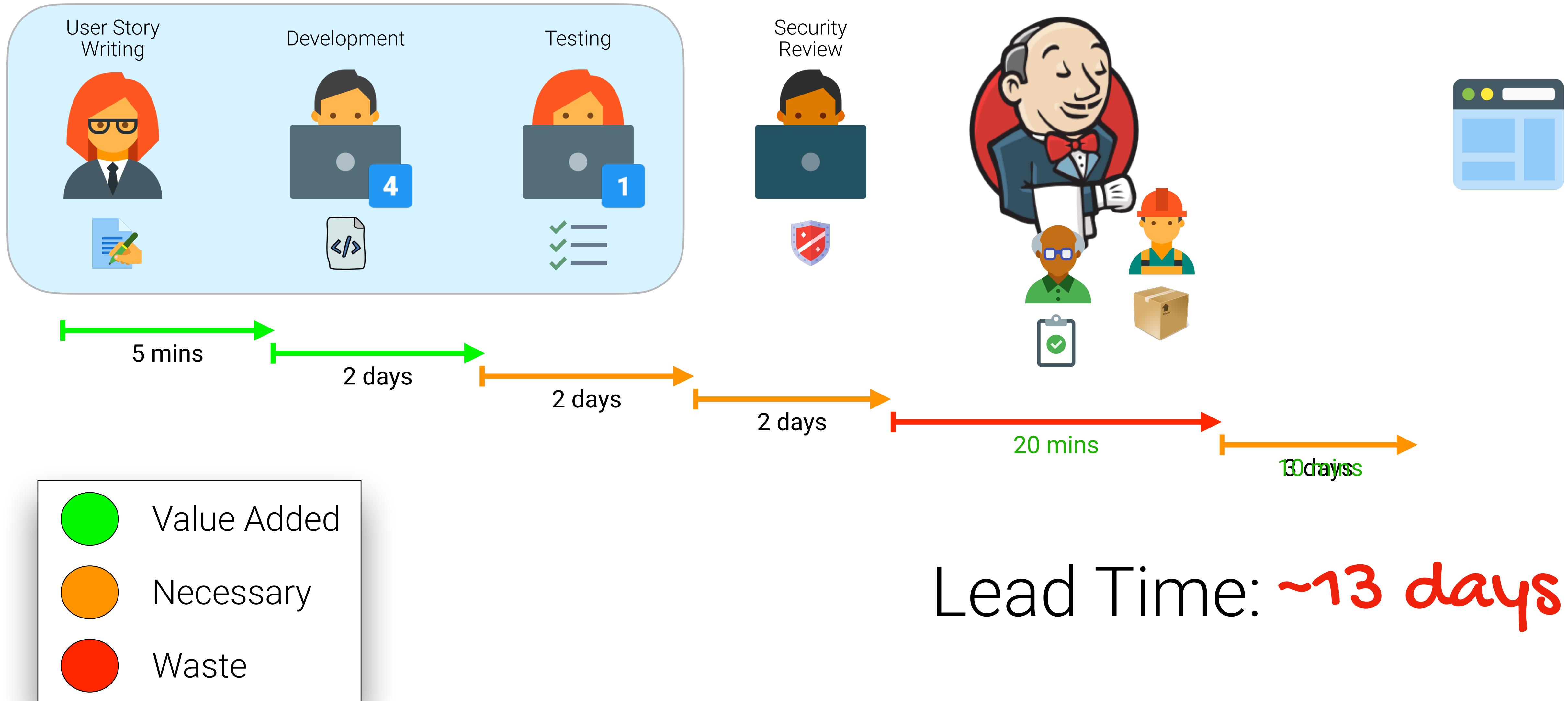
Value Stream Mapping



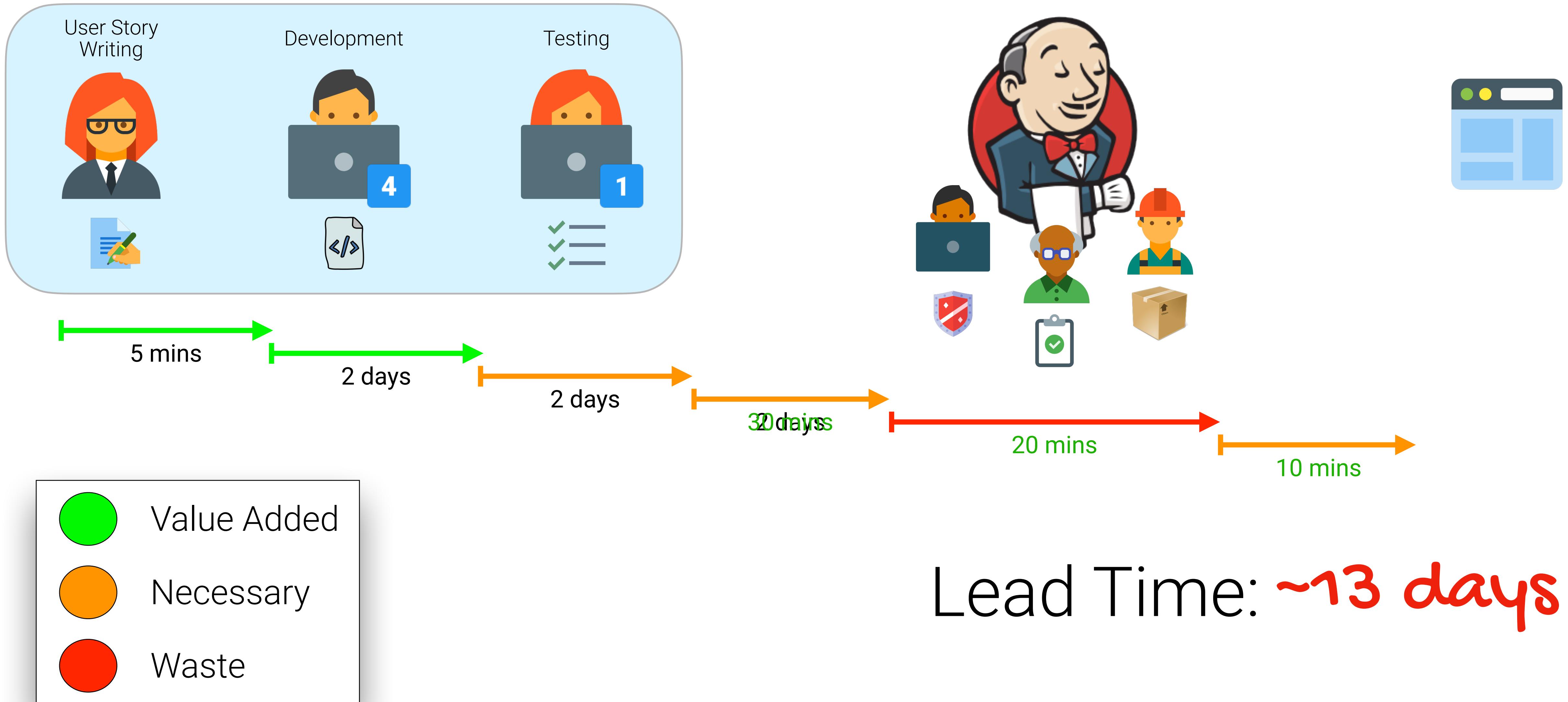
Value Stream Mapping



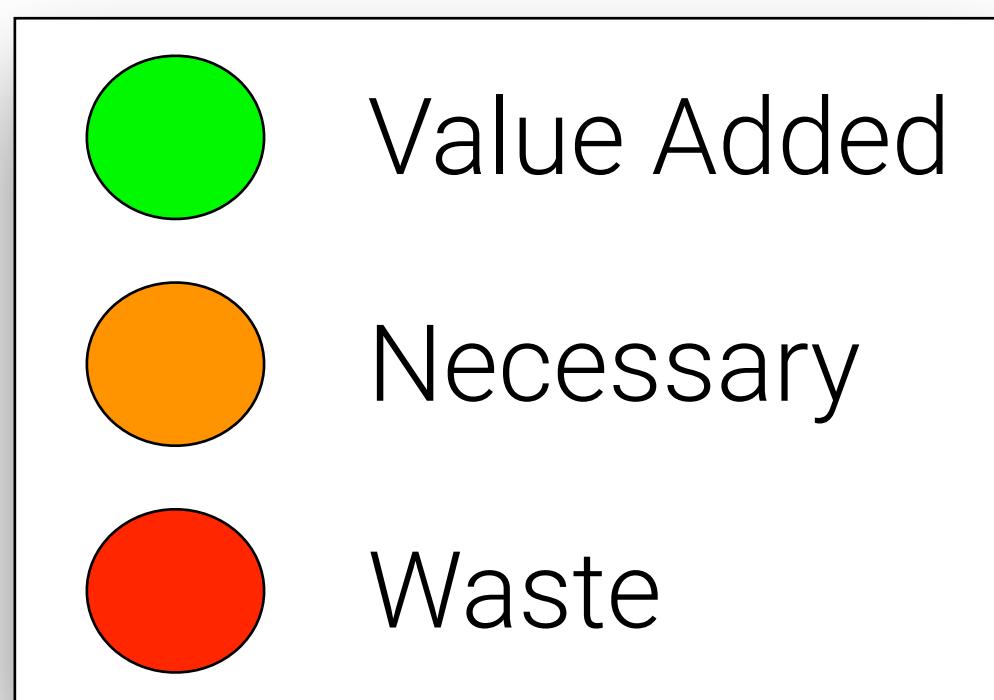
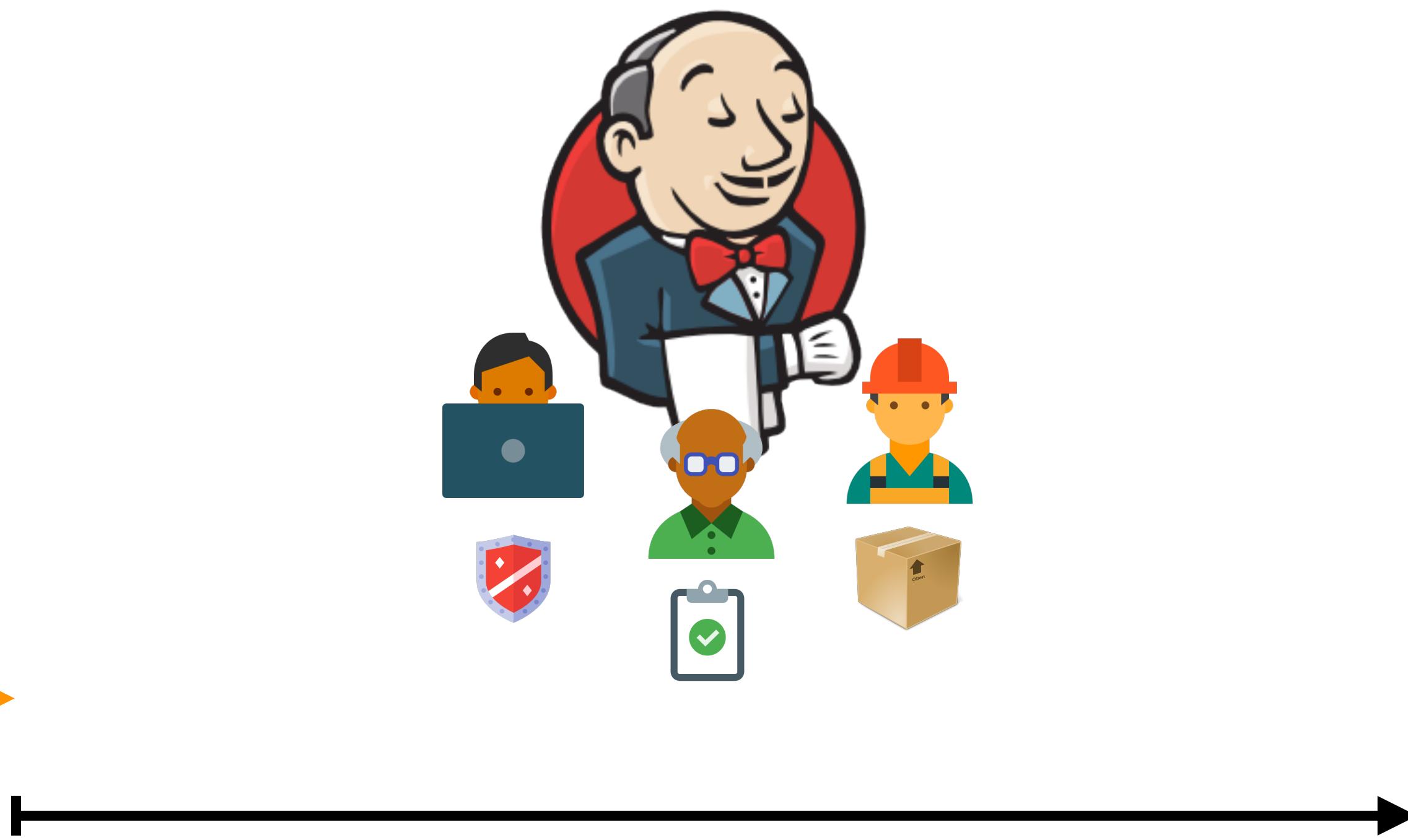
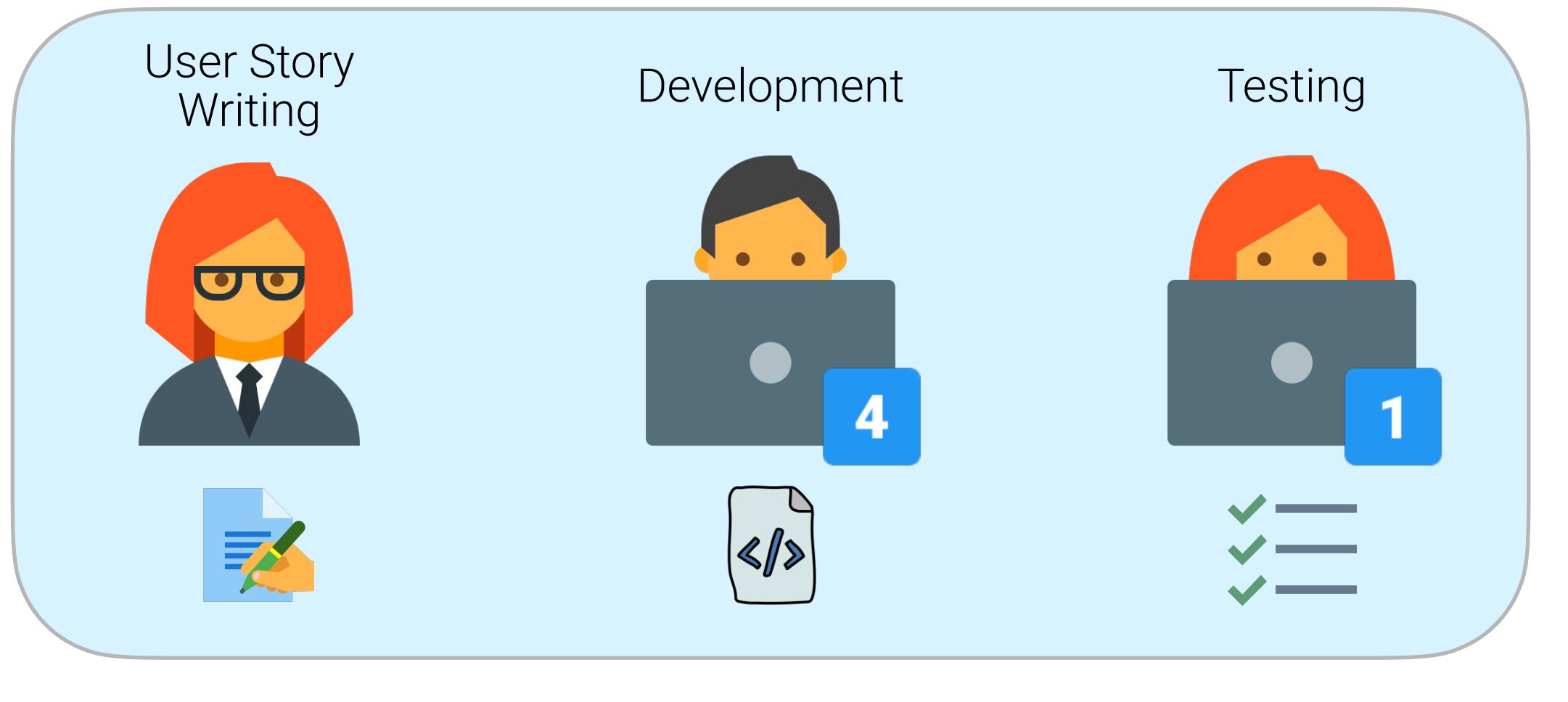
Value Stream Mapping



Value Stream Mapping



Value Stream Mapping

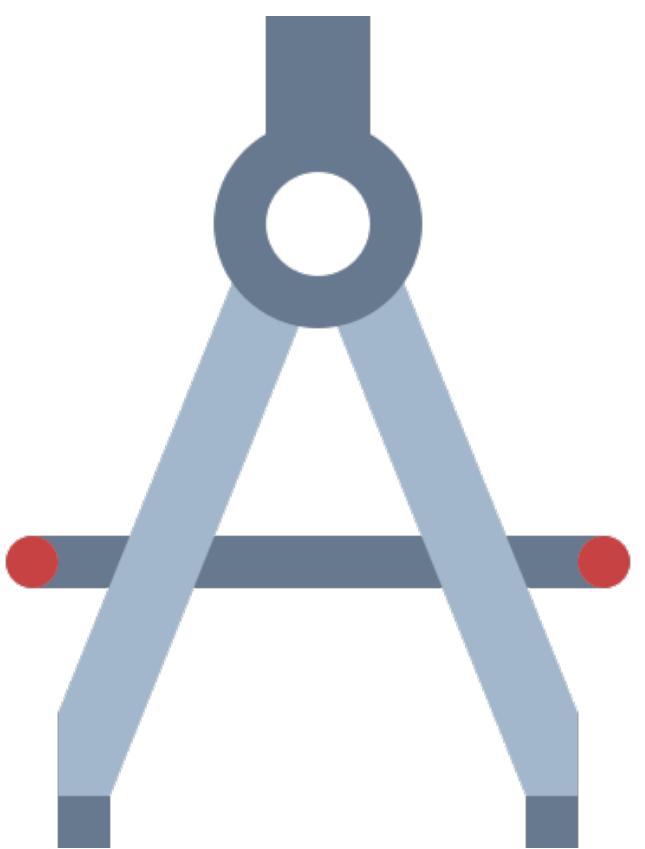


Lead Time: ~43 days

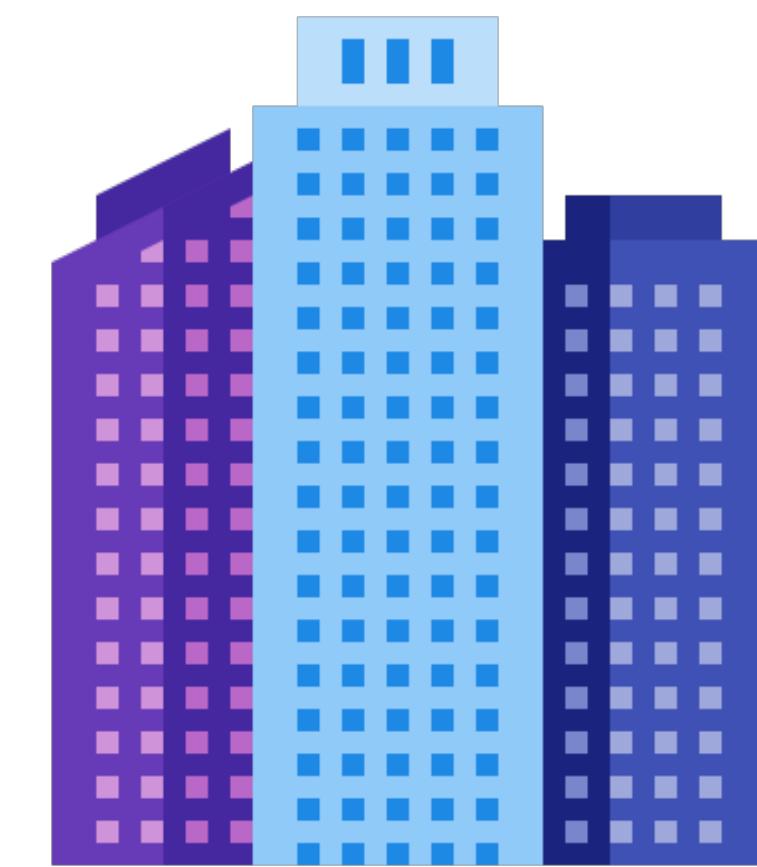
What to Build



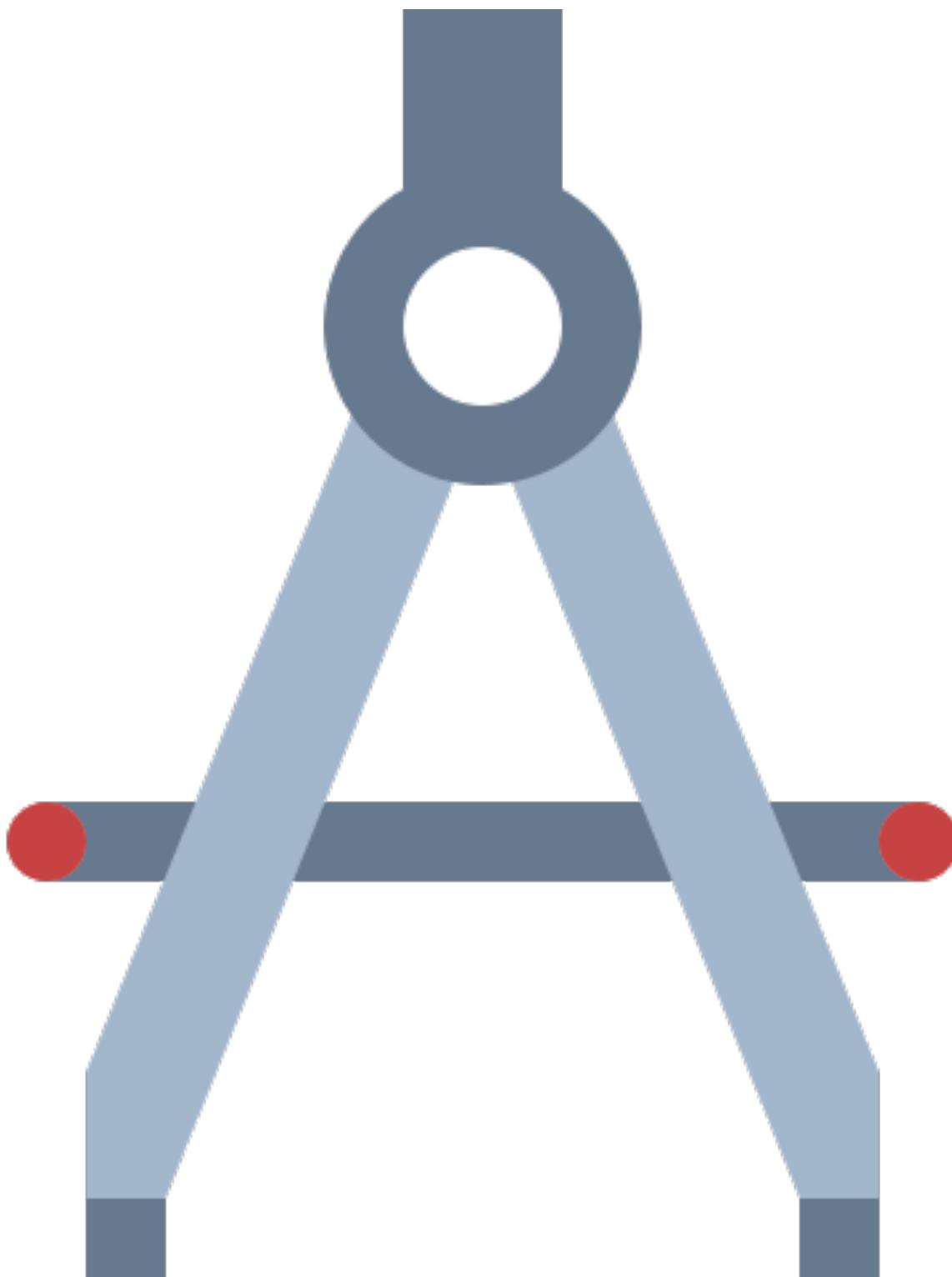
Design



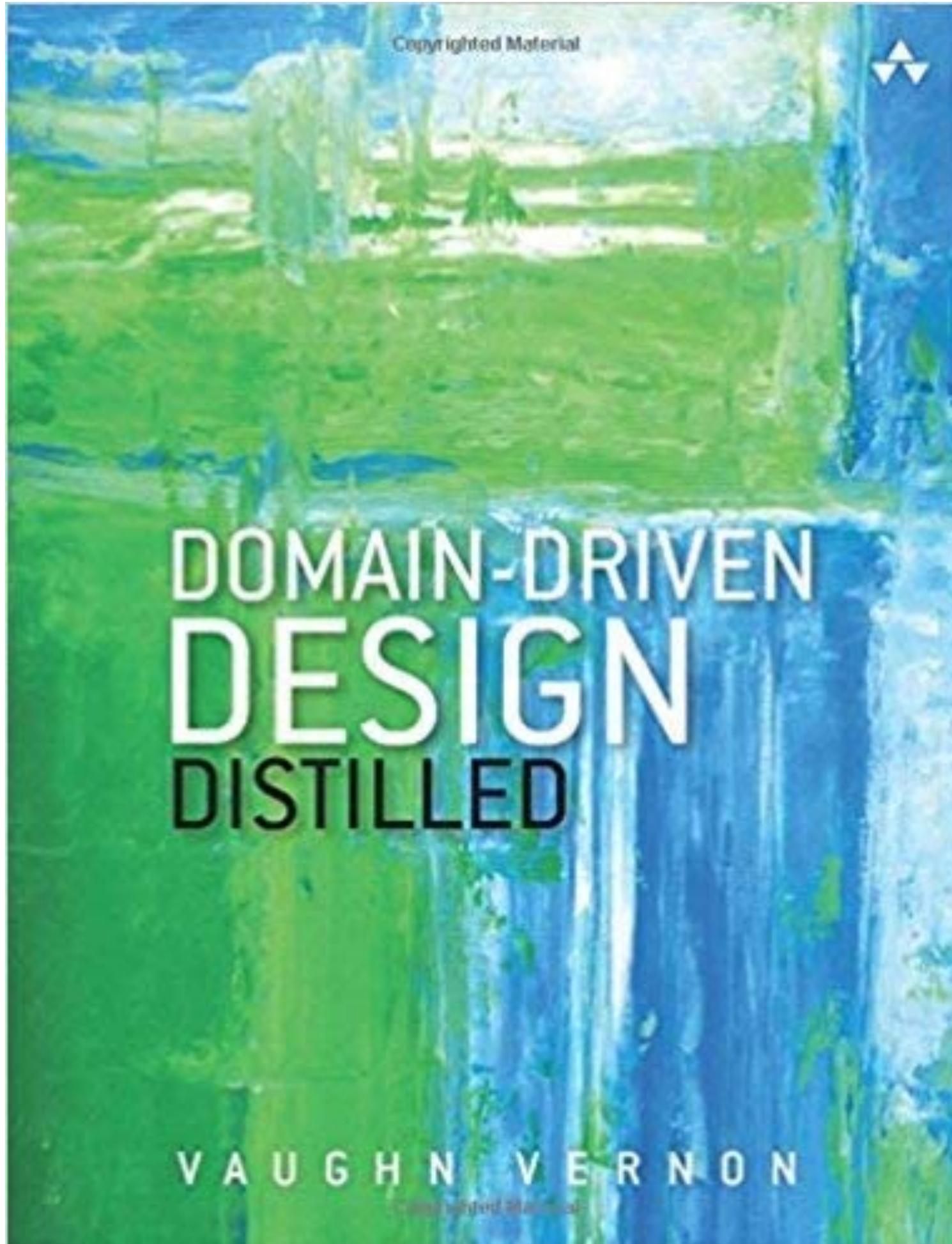
Architecture



Design



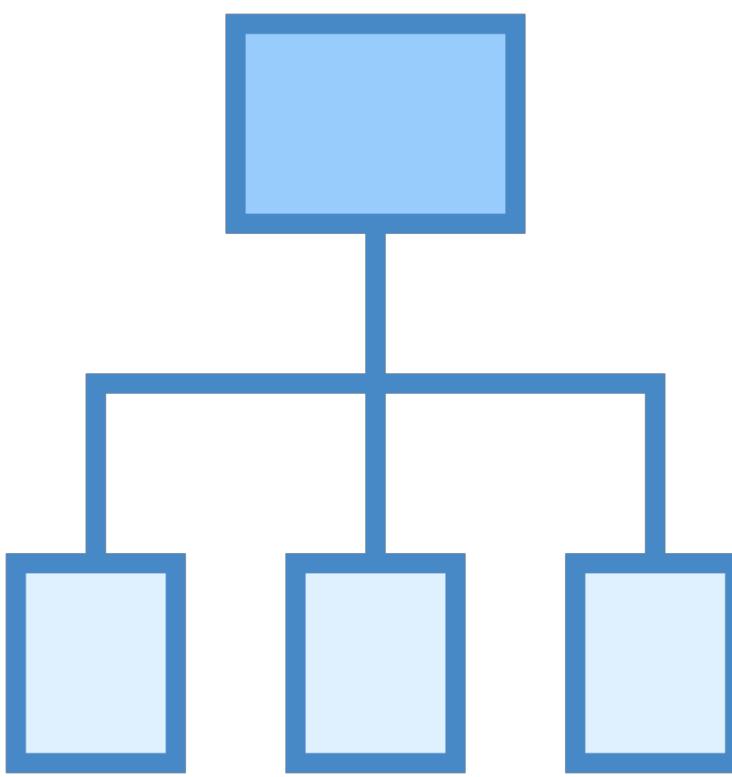
Domain Driven Design



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

Domain Driven Design

Ubiquitous Language



Diagrams



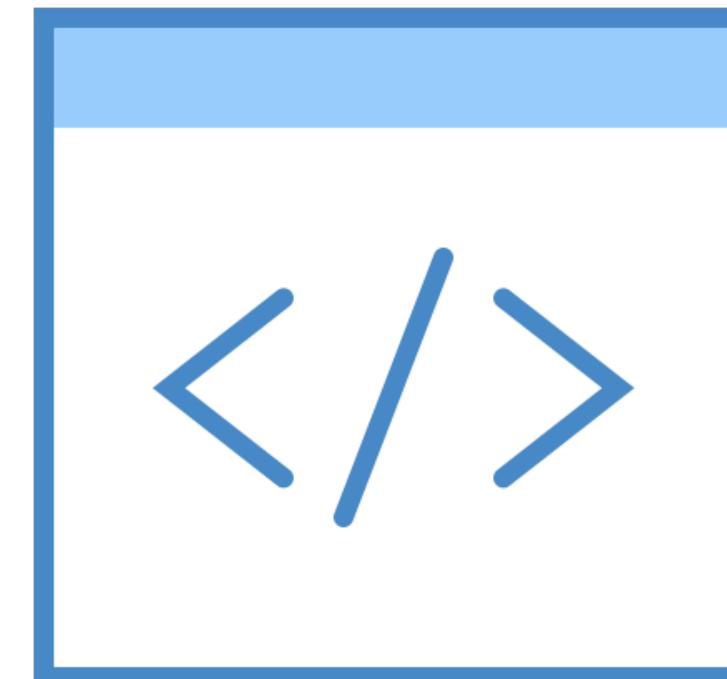
Documentation

thinking

involvement
implement
reach probably health
doctor years
problem state program cost
technology information
different started
system talking
happened back
done even kinds right
position put set care
learning new stuff
call challenge good support
keep service way may
challenge process nows network
available bit issue clinical
challenge service may area
challenge giving ways anything
challenge want times room
challenge actually using
challenge side means change
challenge means telestroke
challenge telemedicine trying connect
challenge taking telehealth part
challenge video school medical department
challenge providers equipment
challenge sort something person meetings
challenge everything interesting



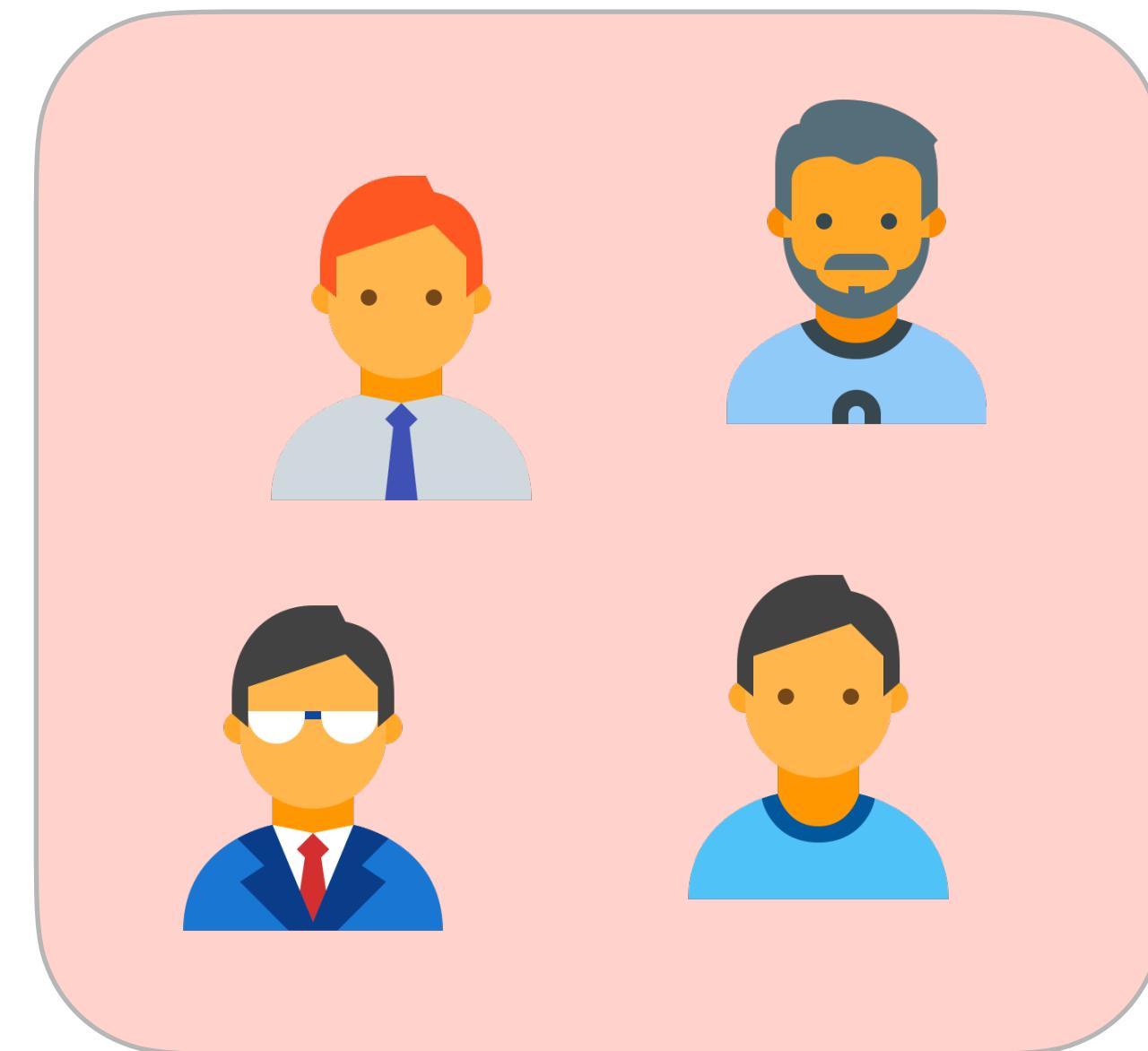
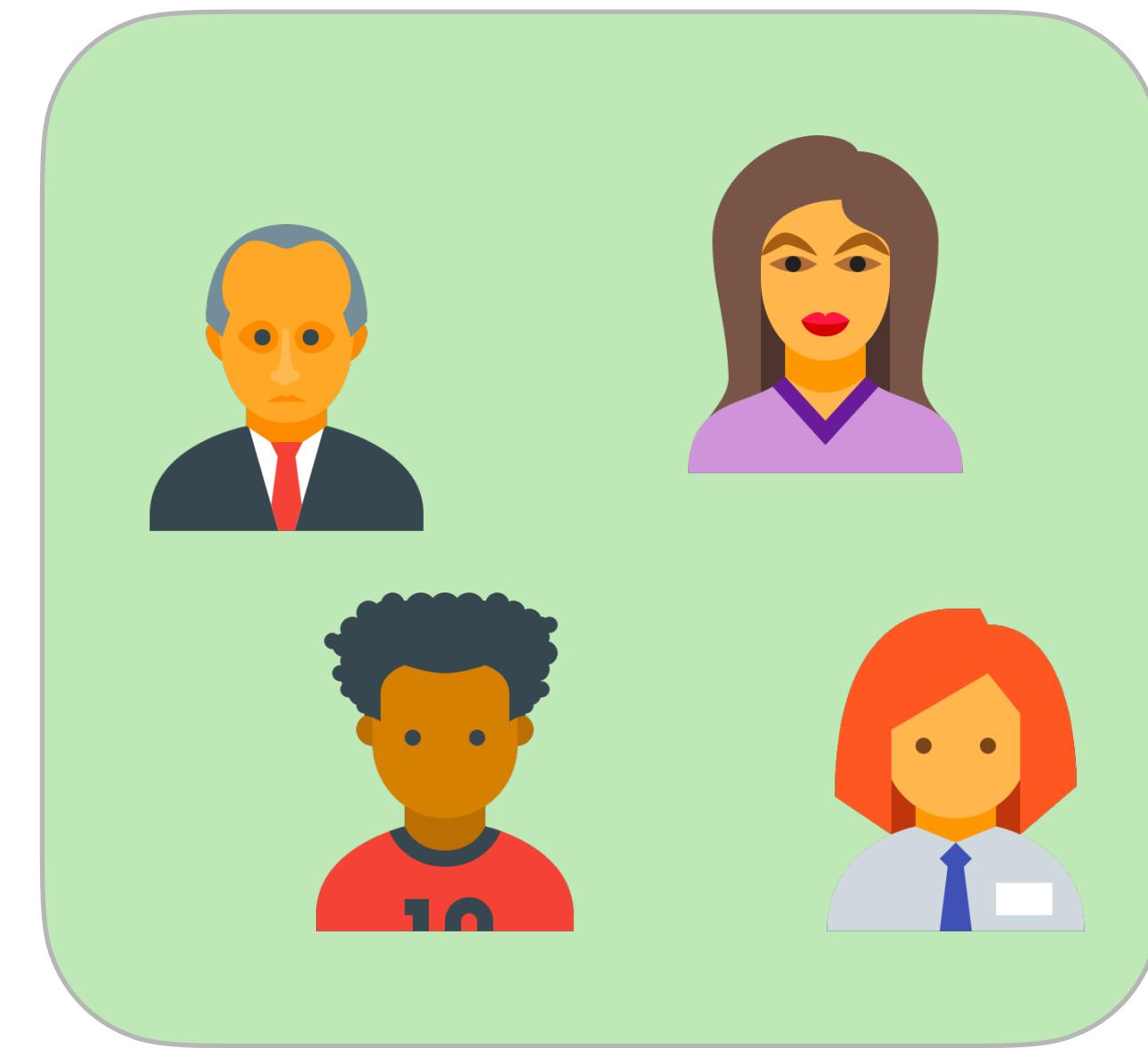
User Stories



Code

Strategic Design

Bounded Context



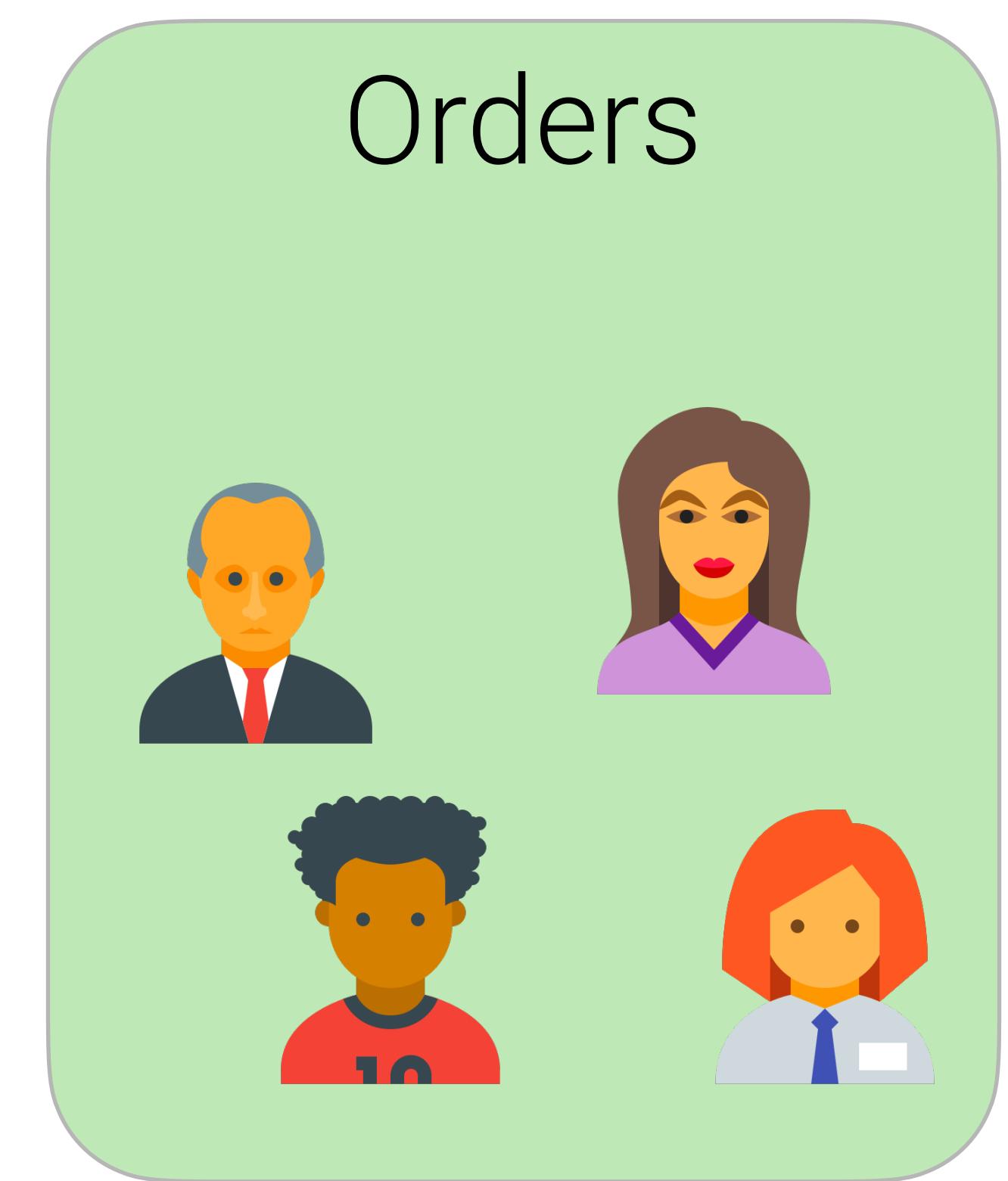
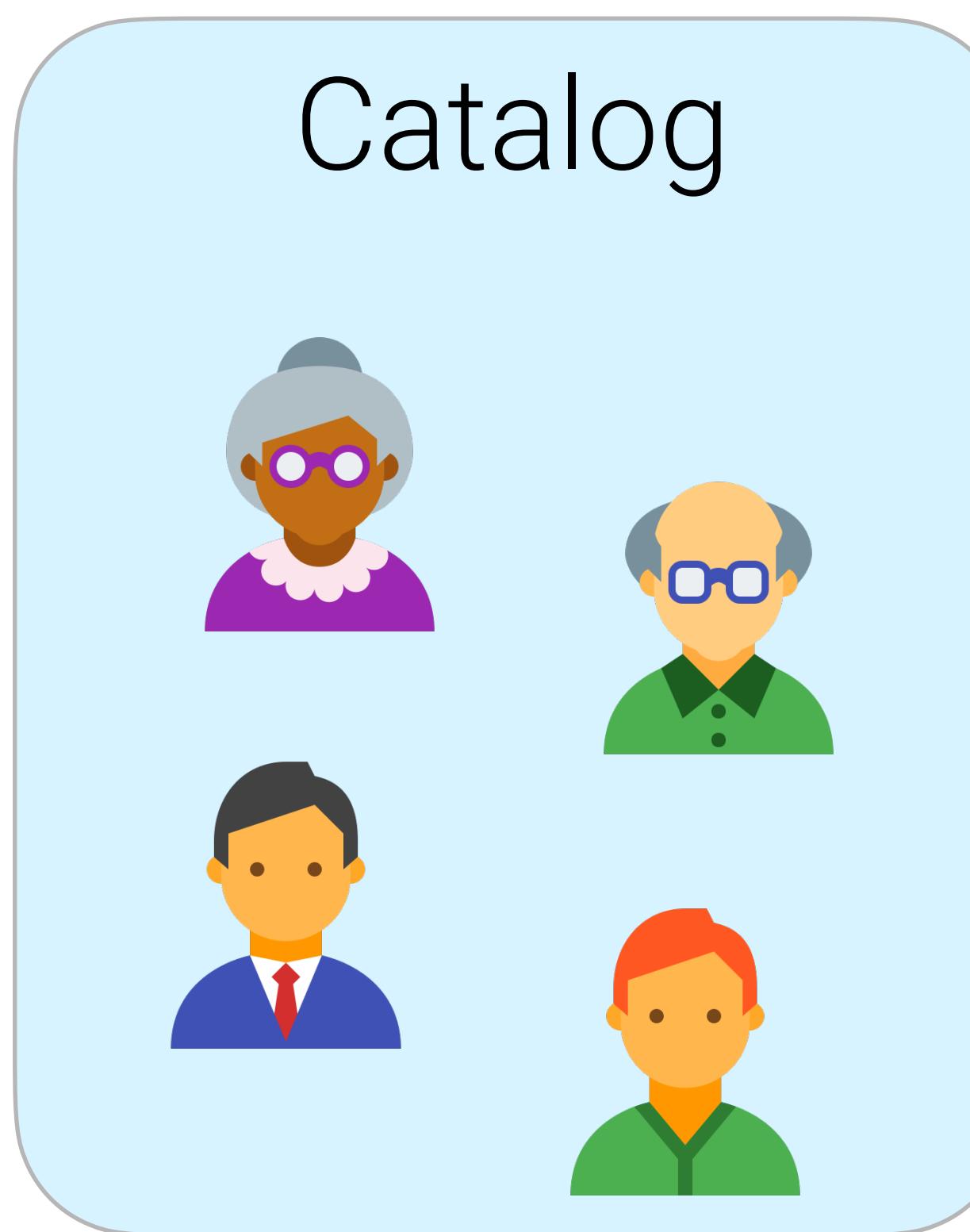
Strategic Design

Bounded Context



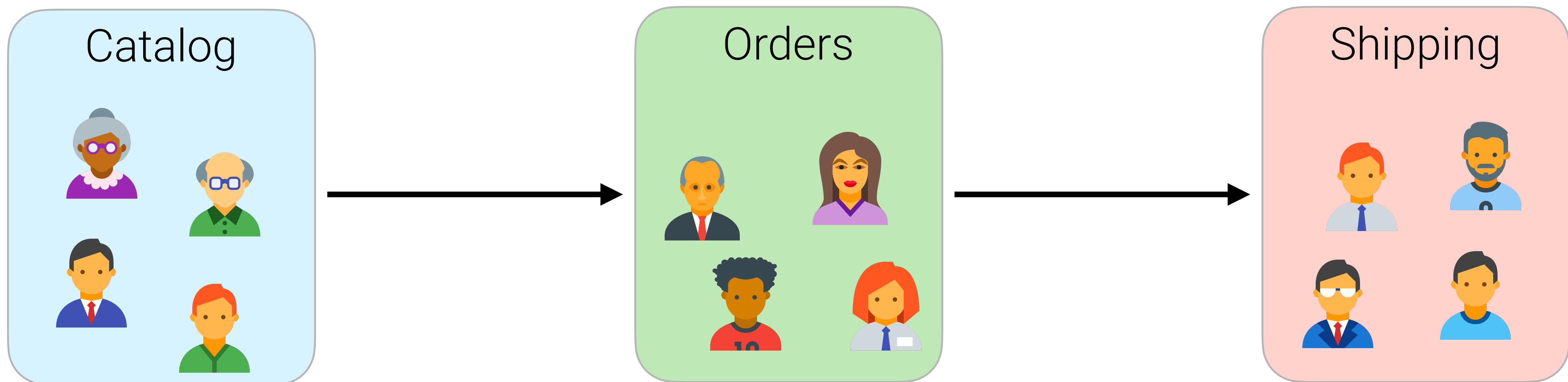
Strategic Design

Bounded Context



Strategic Design

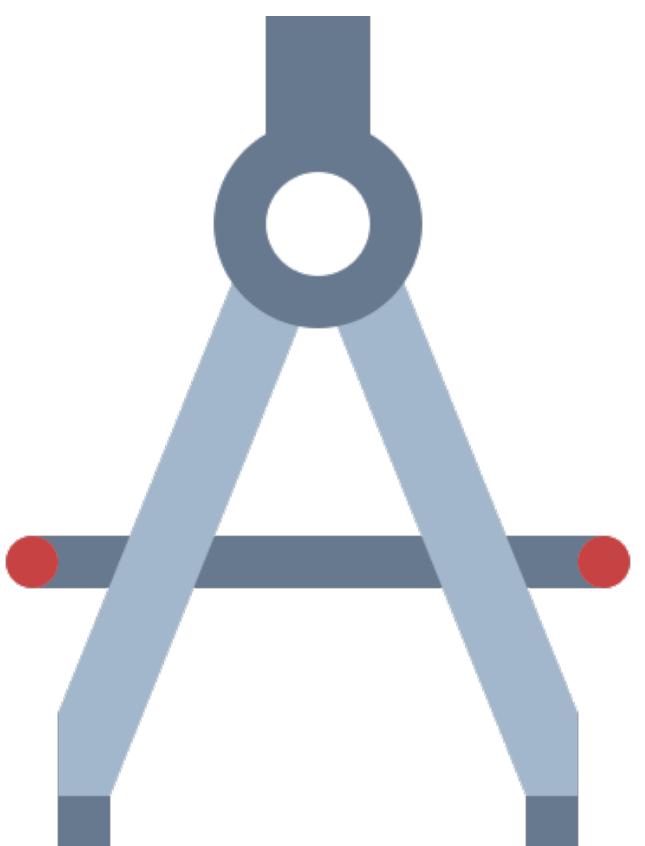
Context Mapping



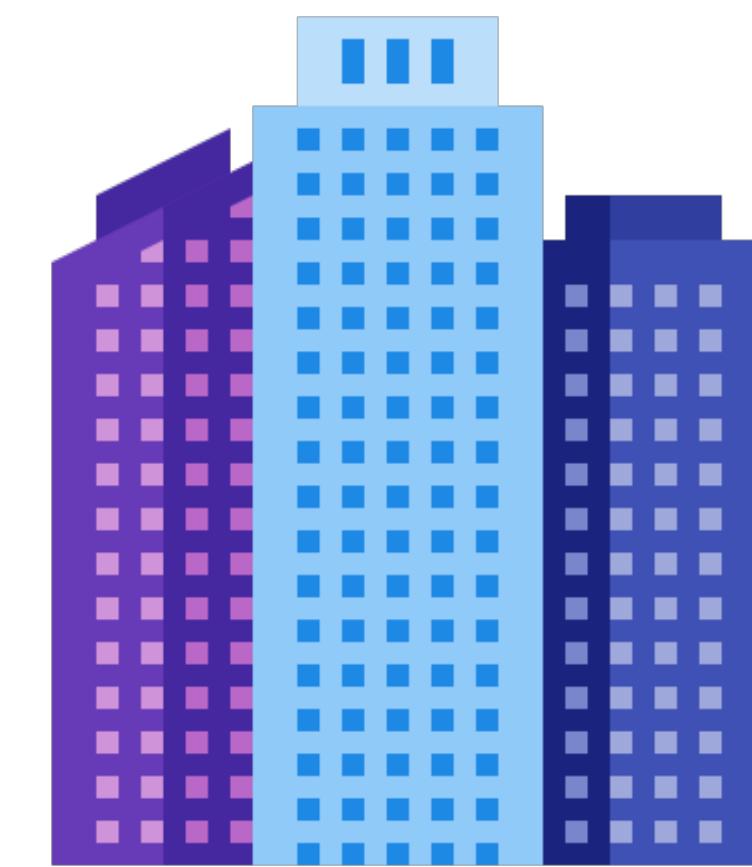
What to Build



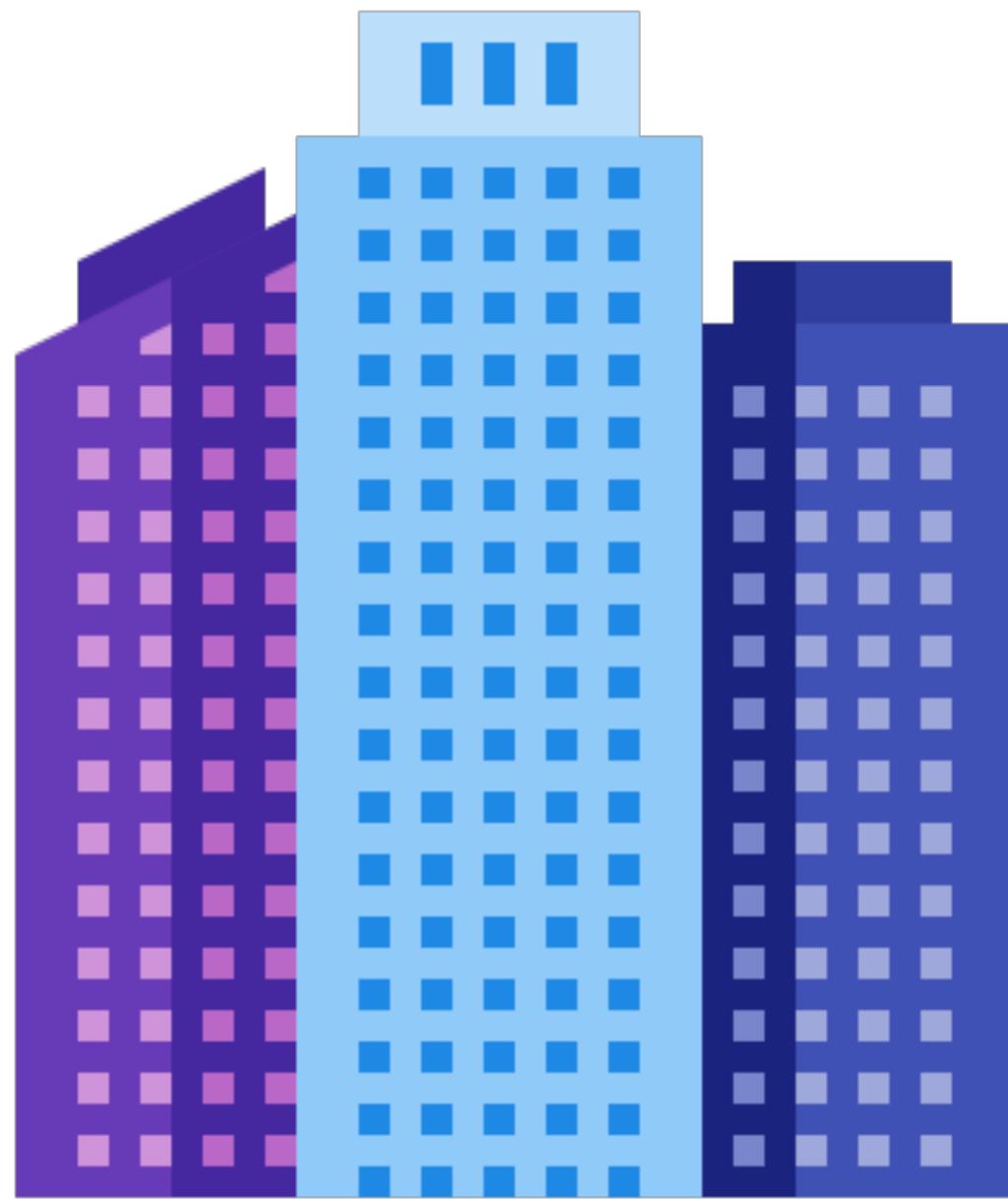
Design



Architecture

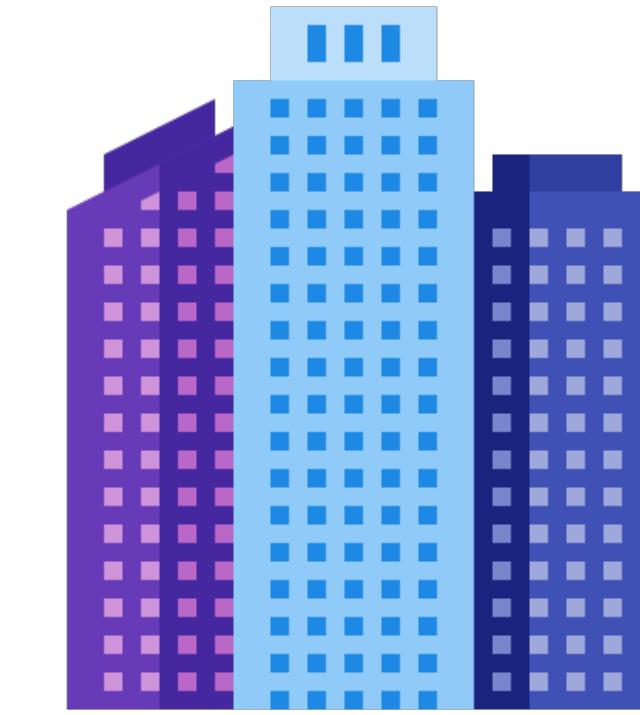


Architecture



Choose an architecture style

- Your Domain
- Strategic Goals
- Trade-offs for architectural characteristics



Neal Ford / Mark Richards
Fundamentals of Software Architecture

Step 1

Choose Architectural Characteristics

Architectural Characteristics

Scalability

Usability

Reliability

Agility

Portability

Durability

Resilience

Traceability

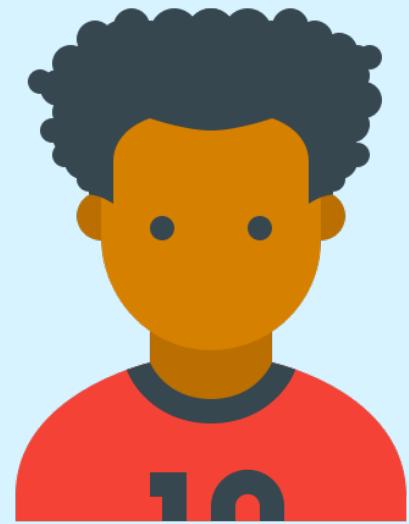
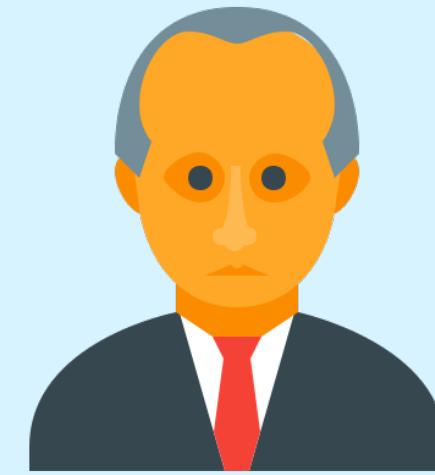
Security

Evolvability

Step 2

Monolithic vs Distributed

Catalog



Monolithic vs Distributed

Catalog

Availability

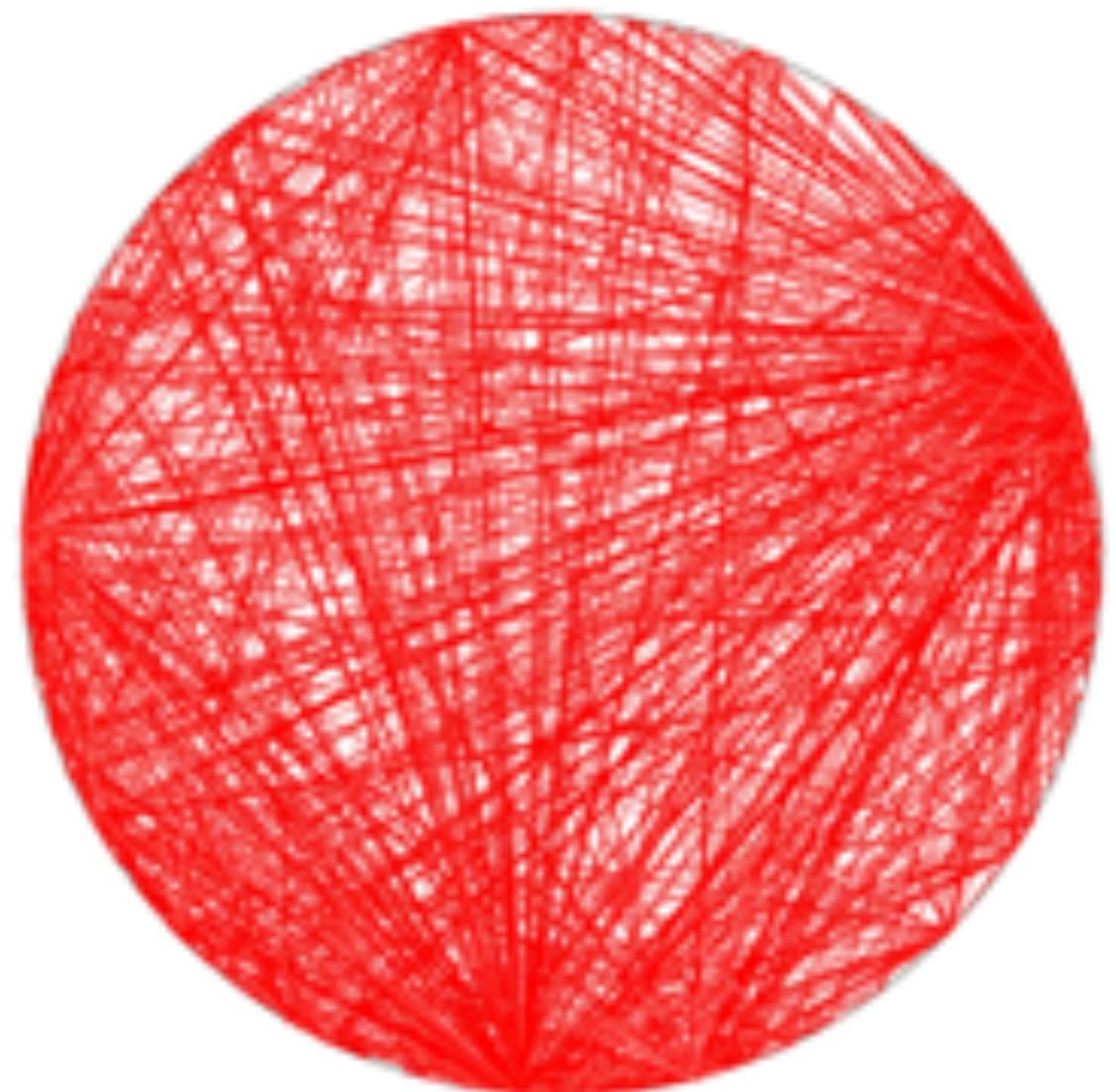
Correctness

Durability

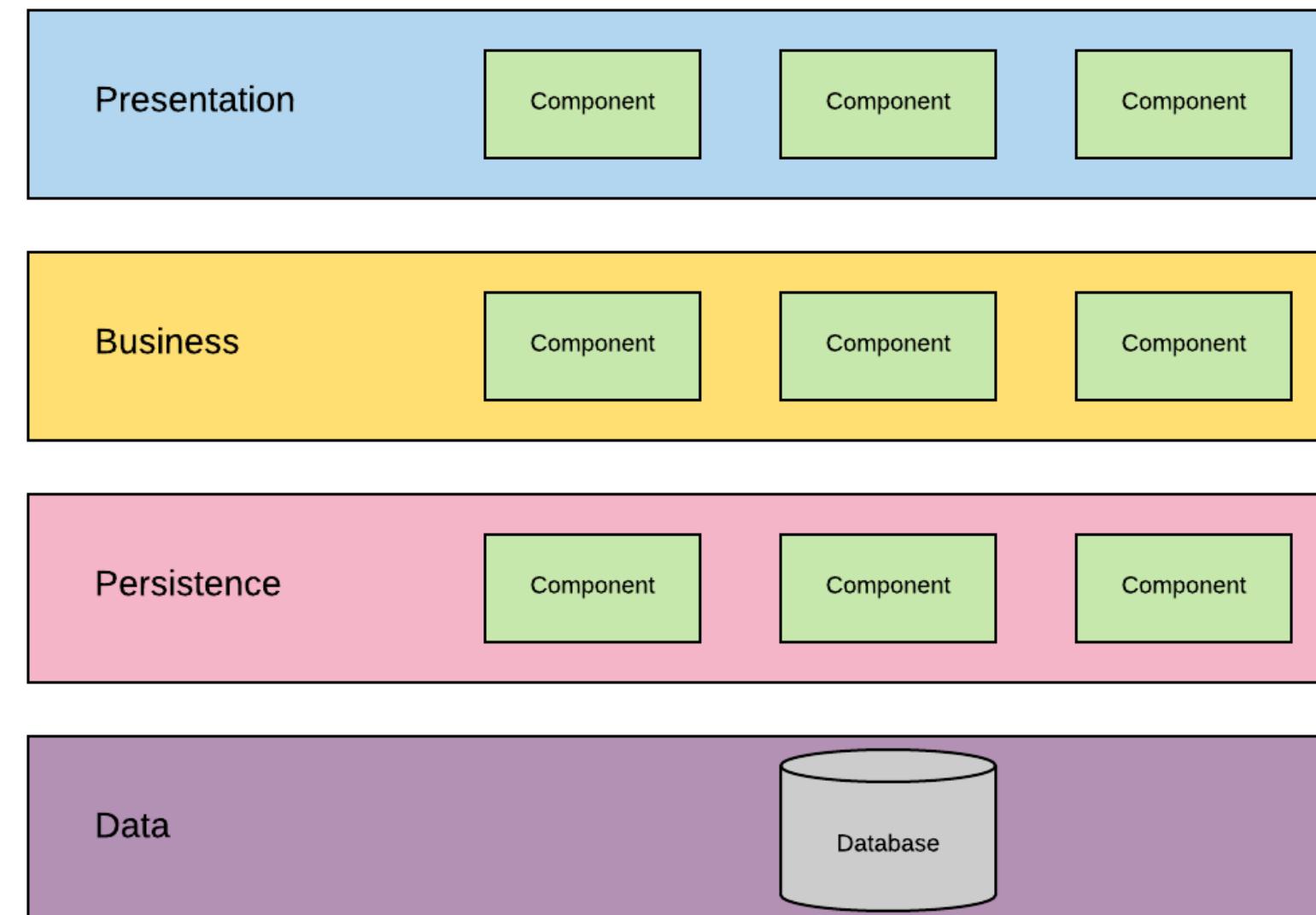


Monolithic Architecture Styles

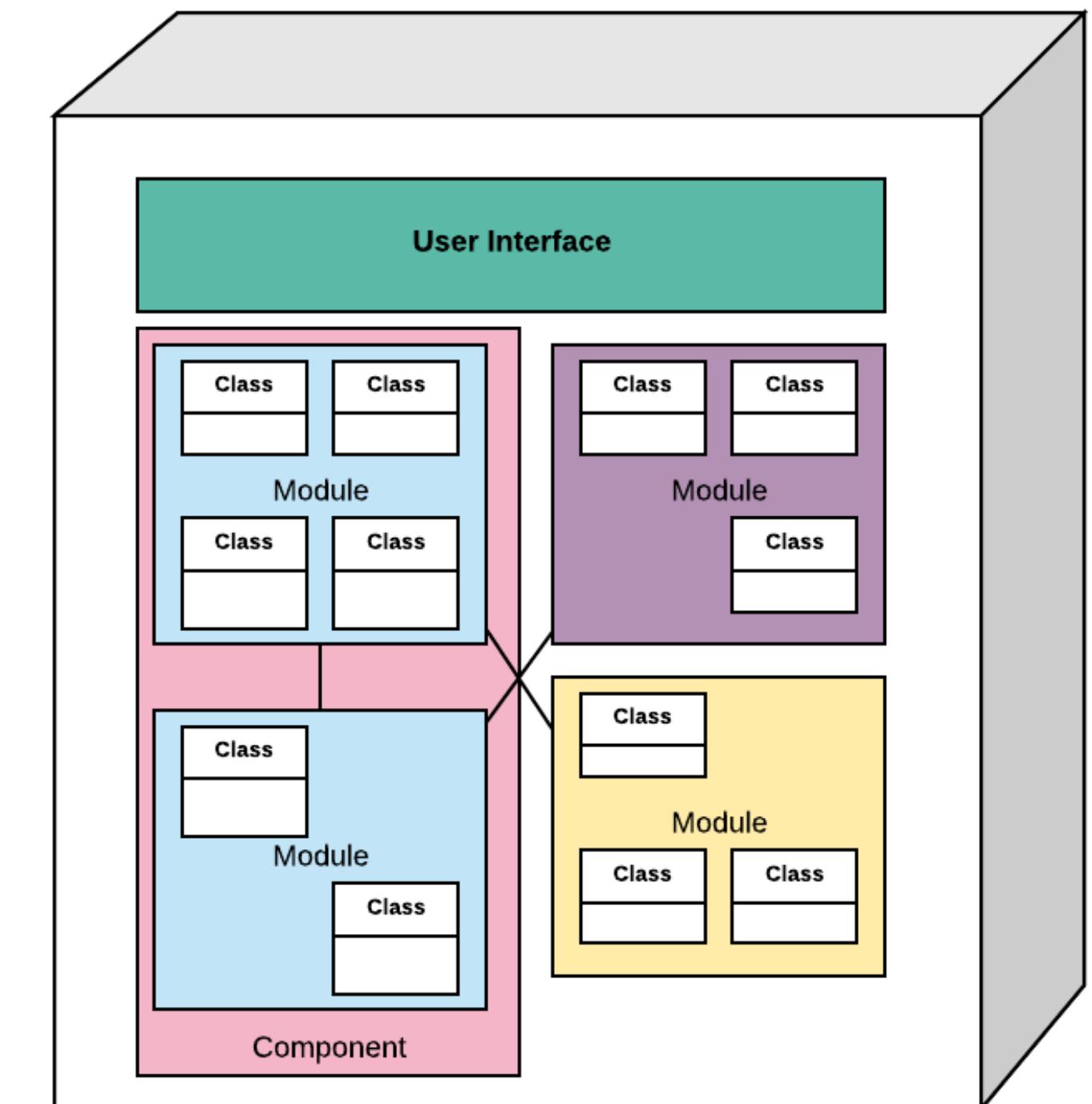
Big Ball of Mud



Layered Architecture

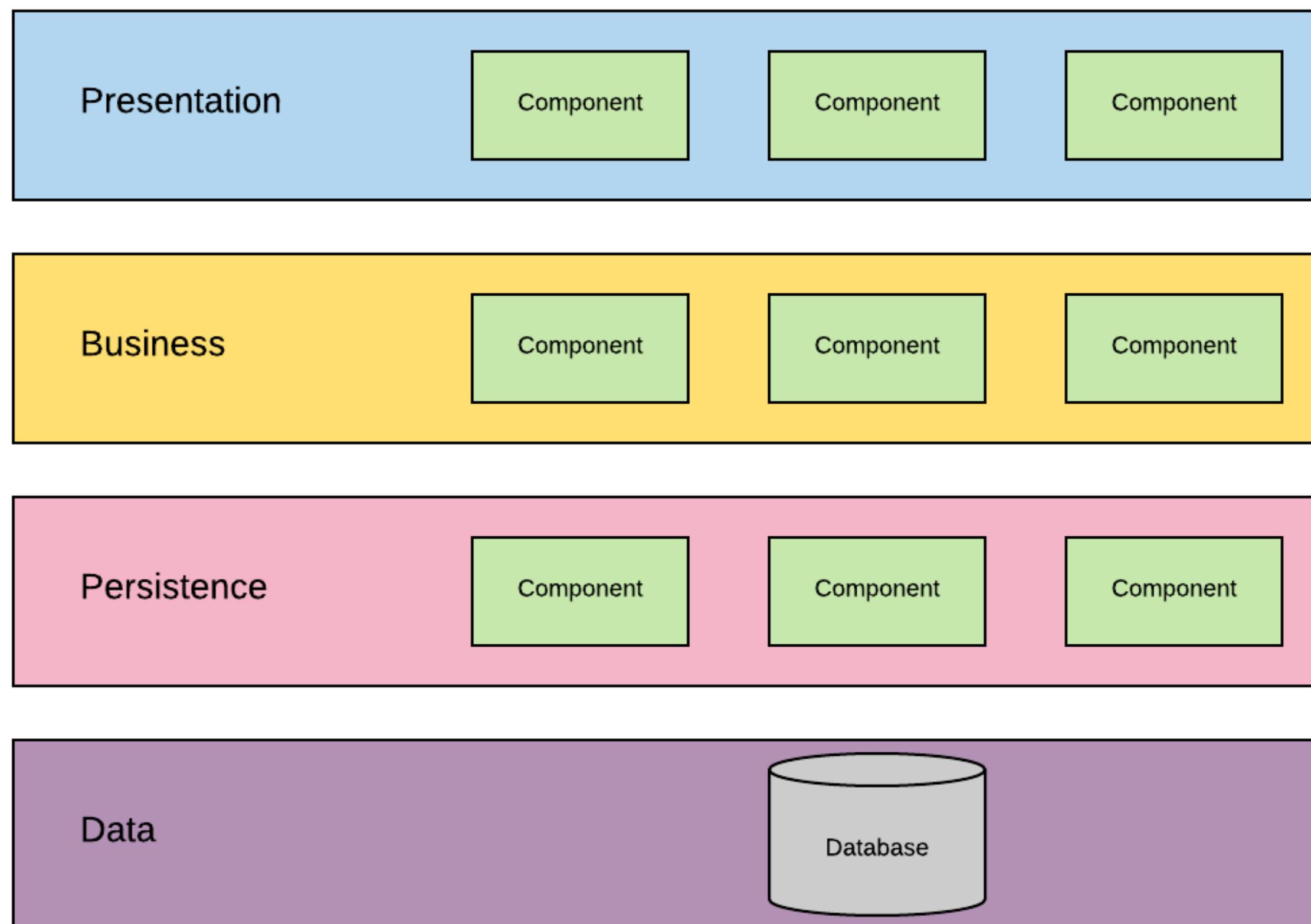


Modular Monolith

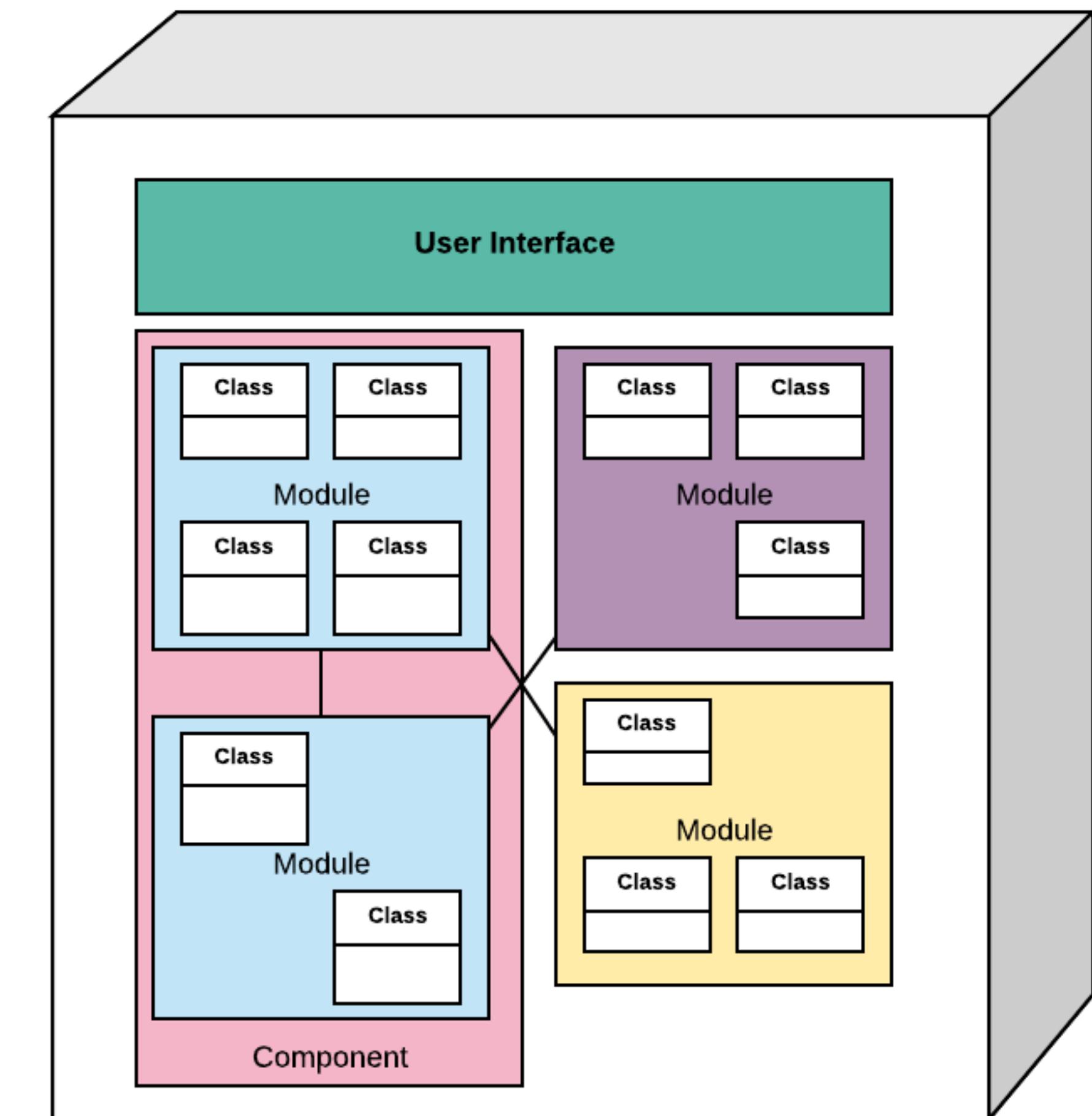


Monolithic Architecture Styles

Layered Architecture



Modular Monolith



Monolithic Architecture Styles

Catalog

Availability

Correctness

Durability

Catalog

User Interface

Accessibility

Simplicity

Usability

Reporting

Accuracy

Integrity

Reliability

Product Catalog

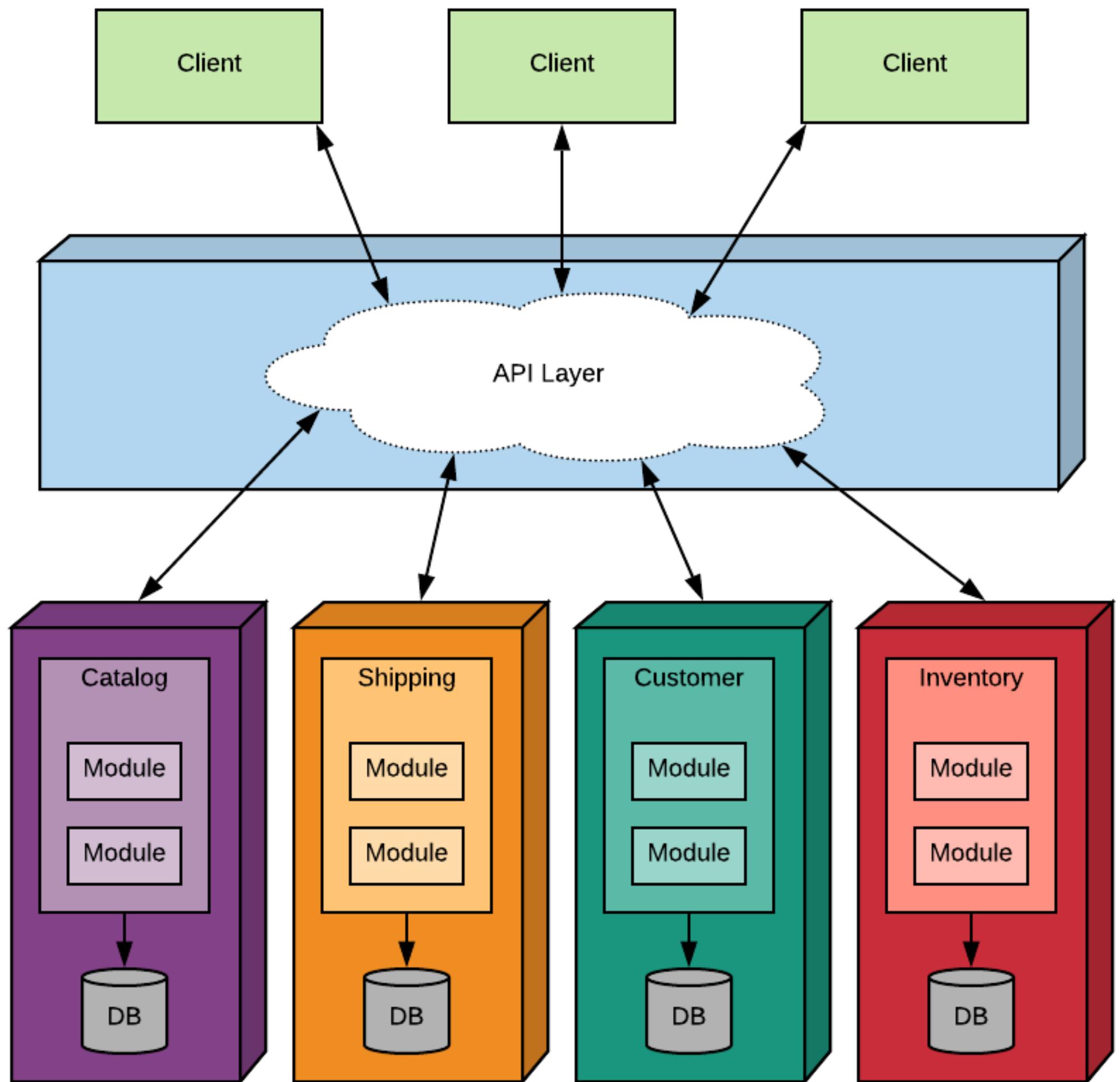
Availability

Correctness

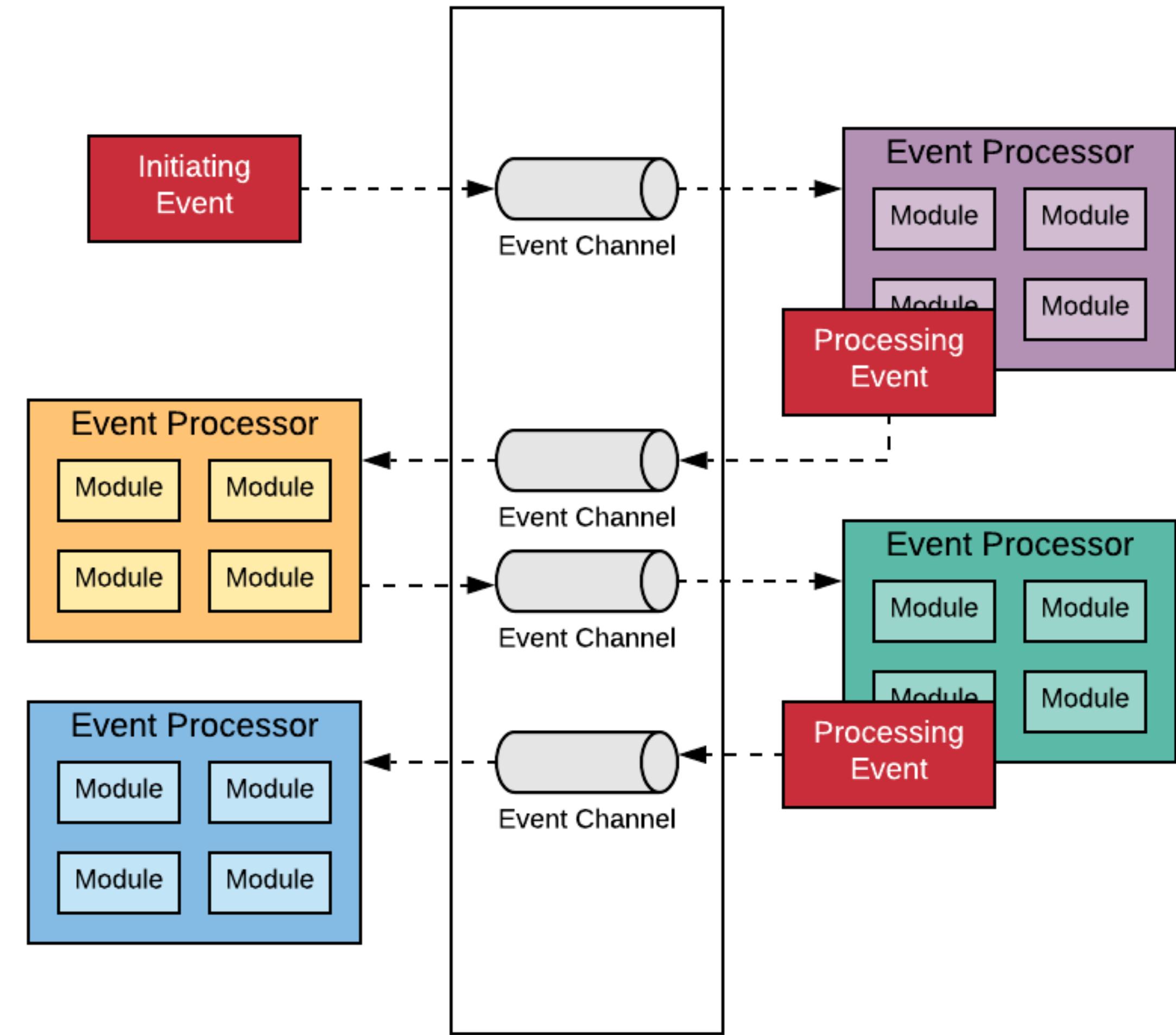
Durability

Distributed Architecture Styles

Microservices Architecture



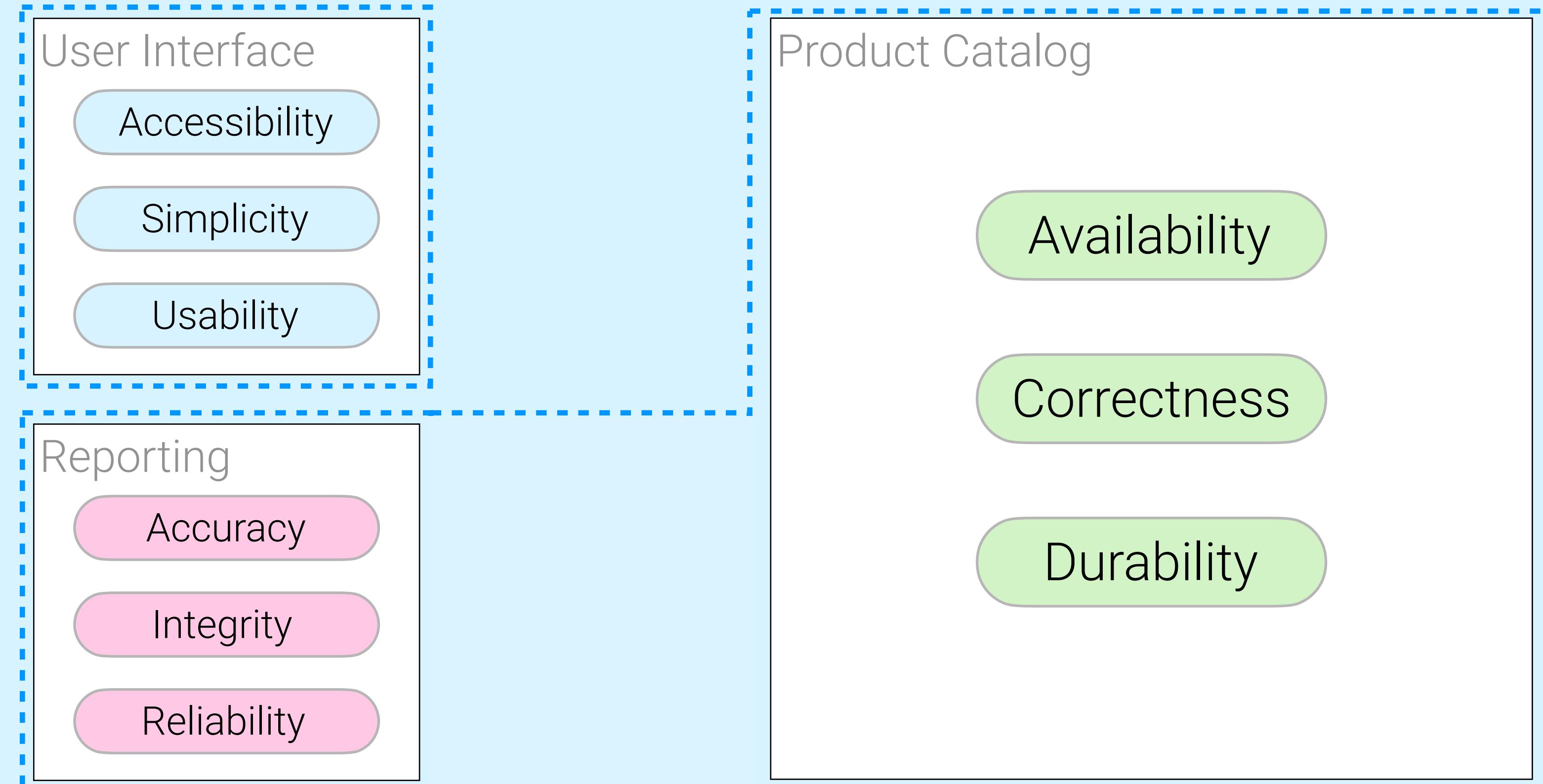
Event-Driven Architecture



Step 3

Determine Architecture Quantum

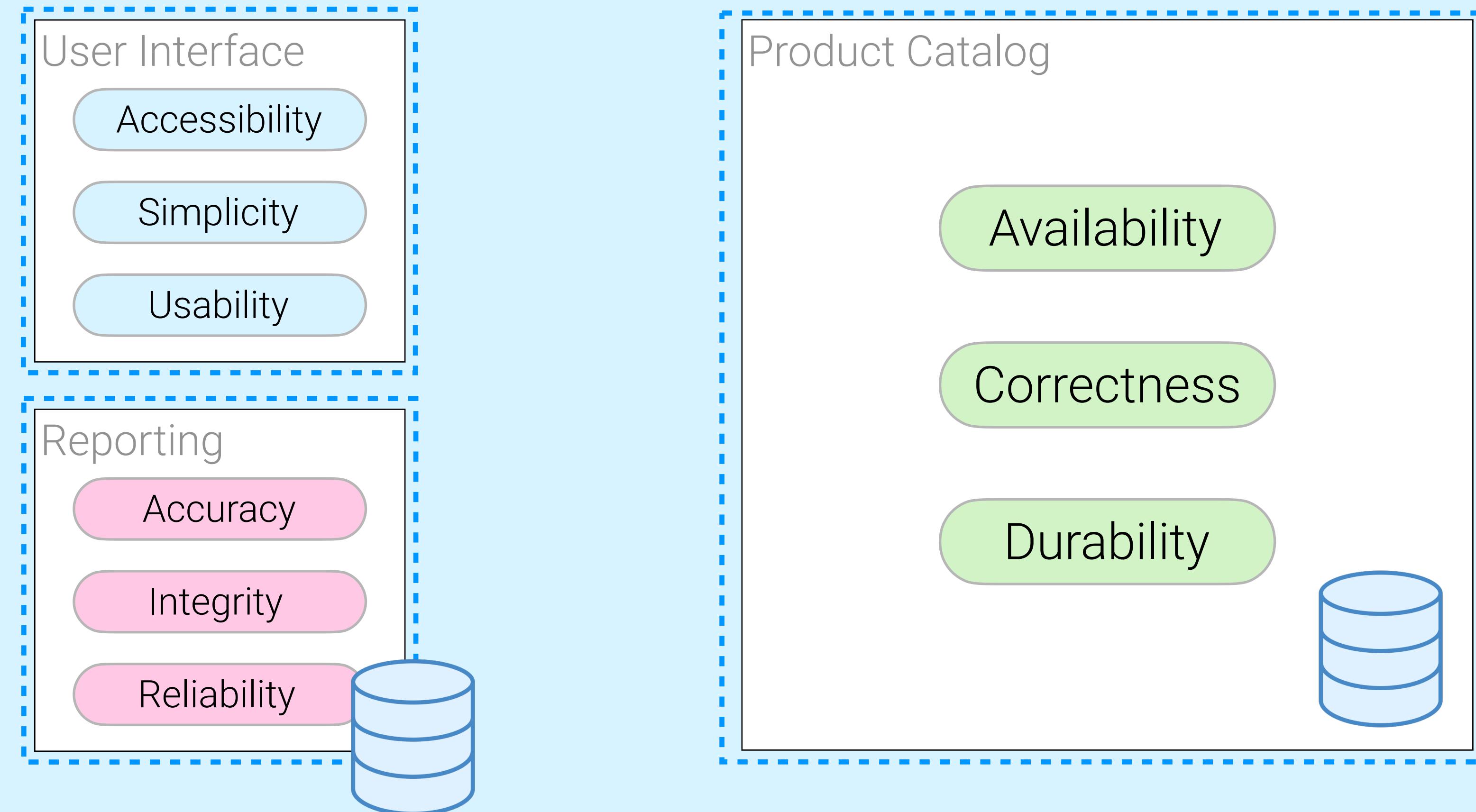
Catalog



Step 4

Determine Persistence

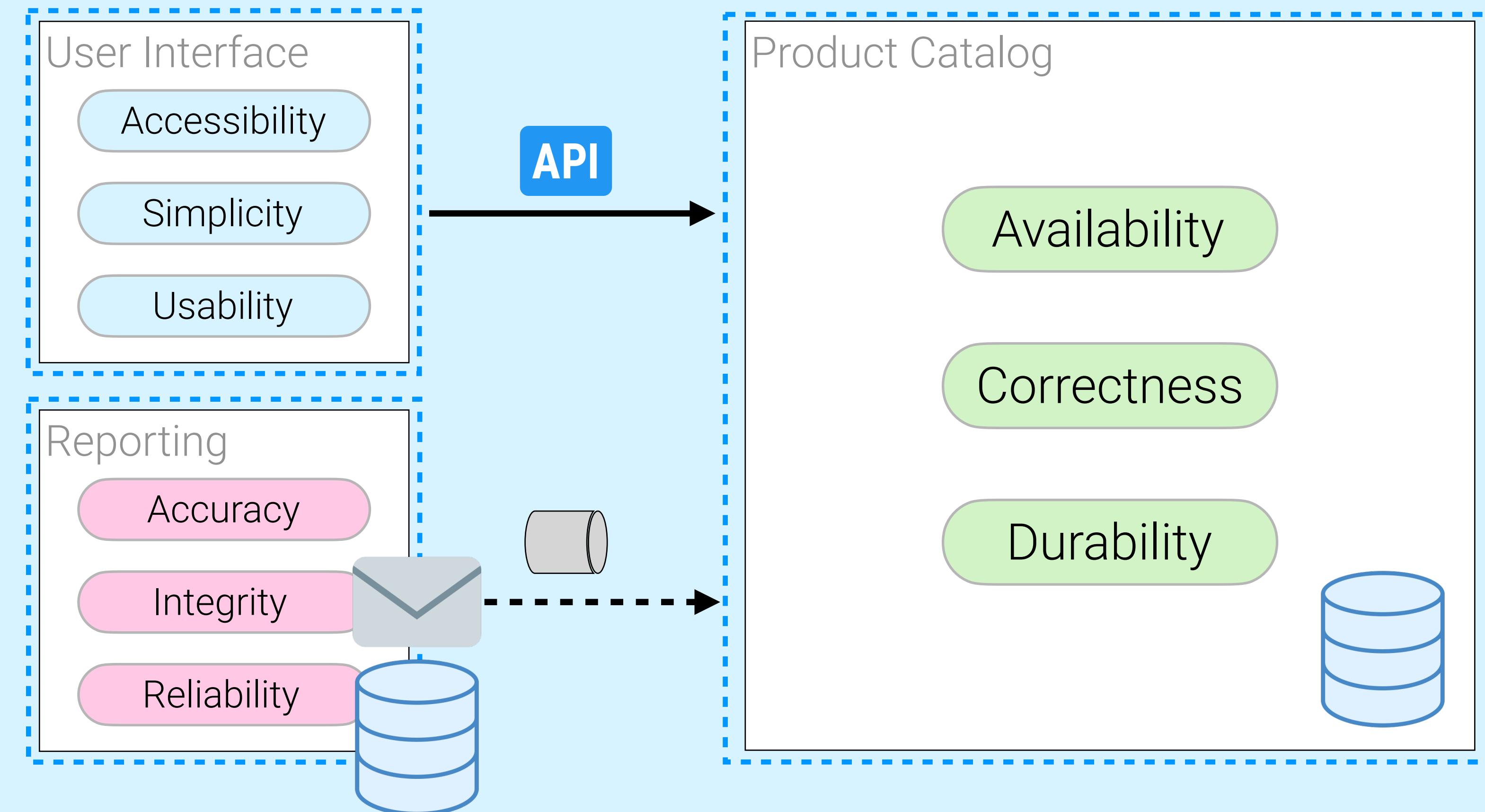
Catalog



Step 5

Determine Communication Style

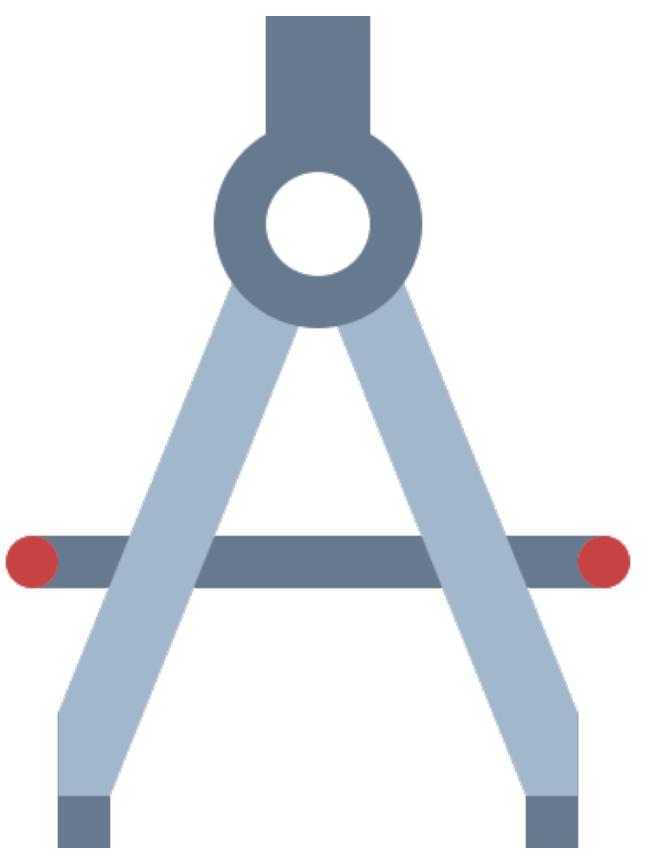
Catalog



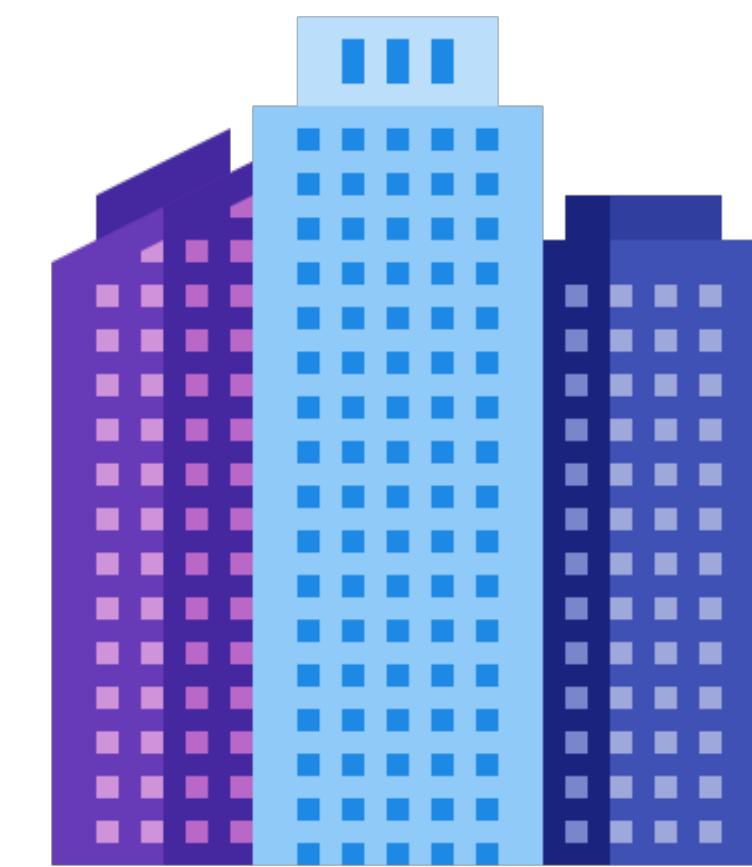
What to Build



Design



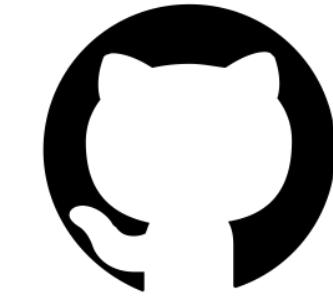
Architecture



Q & A



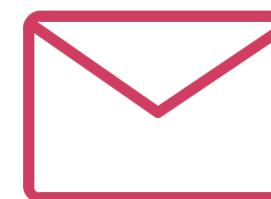
<https://abel.dev>



<https://github.com/abel-fresnillo/slides>



@abel_fresnillo



me@abel.dev