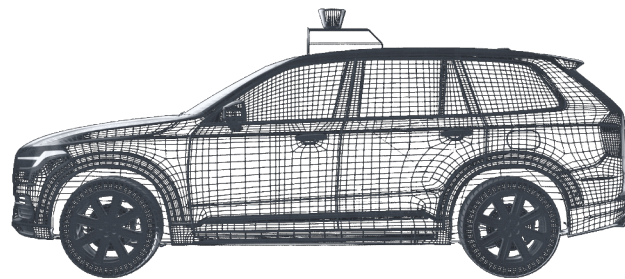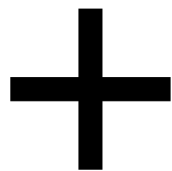# Farmacy Food Architecture Proposal

Farmacy Food Kata

Farmacy Food is a tech-enabled healthy food startup that creates tasty meals around people's dietary needs and active lifestyles to support their overall well-being.

Their mission is to make health and wellness radically affordable and accessible.

Problem Statement

A "ghost kitchen" needs a system to allow users to have visibility of what items are available, purchase, and pick up items at any one of their points of sale.

Requirements

Users: Dozens of automated fridges and representative run kiosks, thousands of customers.

1. Must integrate with 3rd party smart fridges to obtain inventory and purchase activity Smart Fridges Produce item inventory levels and purchases.
2. Must integrate with point of sale system at kiosks
3. Mobile and Web accessible
4. Support providing feedback on items of verified purchases and in app surveys Accept coupons and promotional pricing
5. Send inventory updates to central kitchen
6. Would like to allow multiple vendors to offer items through points of sale
7. Wants to harvest data to provide personalized recommendations based on users health goals, purchase history, and item ratings

User Stories

Alice is a Farmacy Food chef.

Claire is a Farmacy Food business analyst.

Edward is a Farmacy Food delivery person.

Mark is a Farmacy Food nutritionist.

Scott is an occasional Farmacy Food eater, but not a member.

Jennifer is a Farmacy Food subscriber, she likes to eat healthy.

Barbara works at the corner coffee shop.

Challenges

- Time To Market
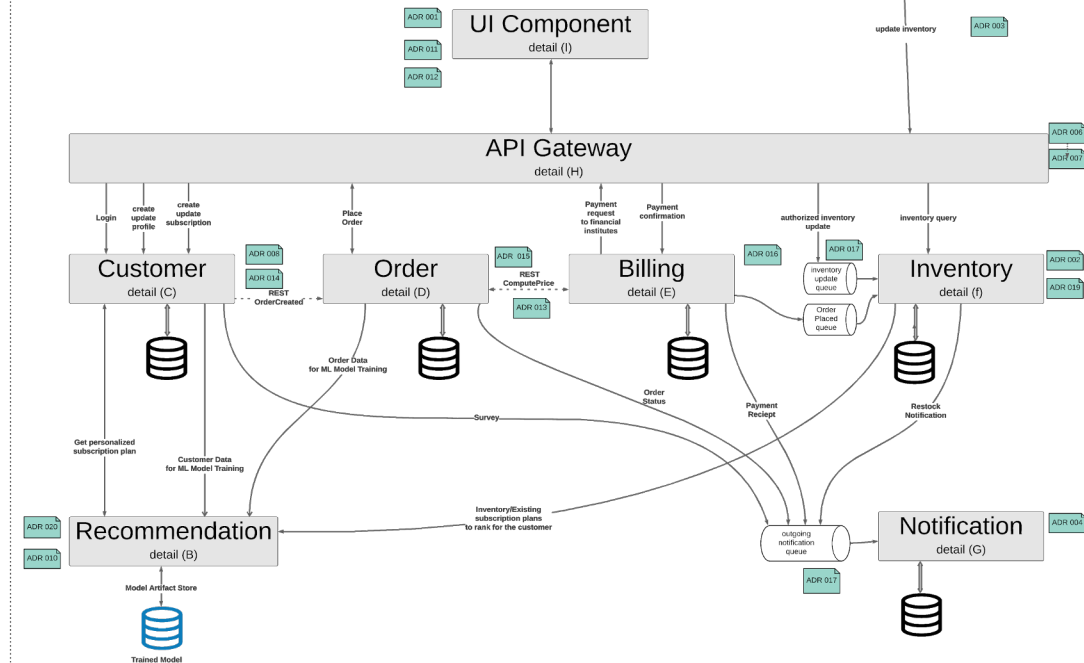- 3rd Party Integrations
- Data Security
- Future Growth

Farmacy Food System Architecture Diagram

Hardware and 3rd party hardware

3rd Party Fridge

SmartFridge

FarmacyFood Software System

ADR 001
ADR 011
ADR 012

UI Component
detail (I)

ADR 003

update inventory

ADR 006
ADR 007

API Gateway
detail (H)

Login

create update profile

create update subscription

Place Order

Payment request to financial institutes

Payment confirmation

authorized inventory update

inventory query

ADR 008
ADR 014

Customer
detail (C)

REST OrderCreated

ADR 015

Order
detail (D)

REST ComputePrice

ADR 013

ADR 016

Billing
detail (E)

ADR 017

inventory update queue

Order Placed queue

ADR 002
ADR 019

Inventory
detail (f)

Order Data for ML Model Training

Order Status

Payment Reciept

Restock Notification

Get personalized subscription plan

Customer Data for ML Model Training

Survey

Inventory/Existing subscription plans to rank for the customer

ADR 020
ADR 010

Recommendation
detail (B)

outgoing notification queue

ADR 004

Notification
detail (G)

ADR 017

Model Artifact Store

Trained Model

Decisions:

- [Microservice Based](#)
- [S.O.L.I.D Base](#)
- Scalable Domains
- [Architecture Characteristics](#)
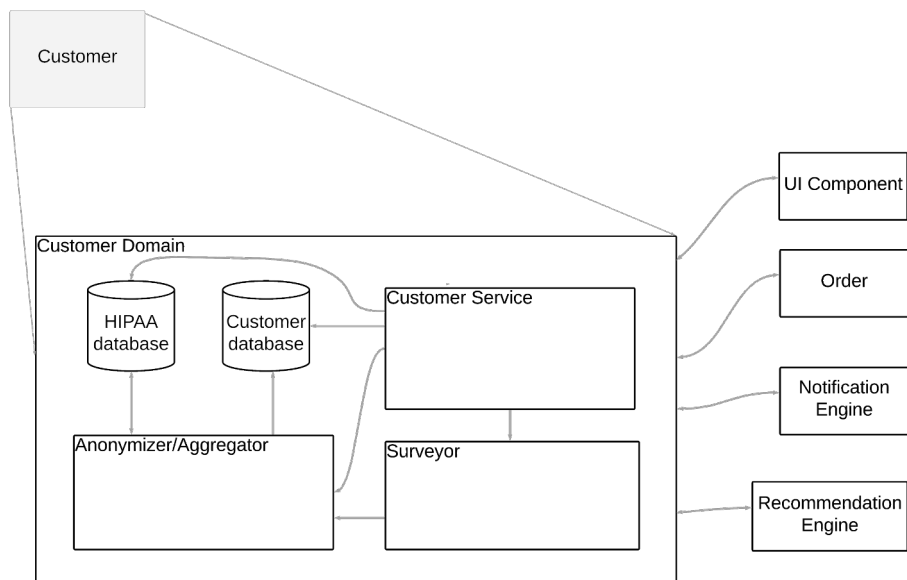
[selfdriventeam](#)

Customers

Decisions:
- [PII](#) must be anonymized
- [REST API](#) between components
- [3rd party health hooks into the customer info](#)
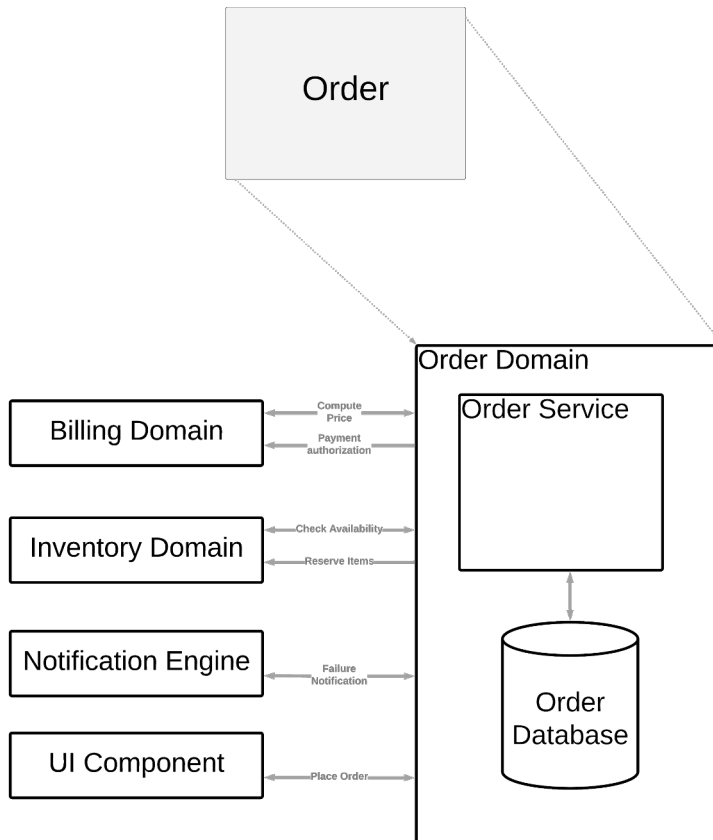
Characteristics:
- Adaptable
- Security
- Extensible

Orders

Decisions:
- No separate delivery component

Characteristics:
- Reliable
- Durable
- Low latency

Order

Order Domain

Order Service

Order Database

Billing Domain
- Compute Price
- Payment authorization

Inventory Domain
- Check Availability
- Reserve Items

Notification Engine
- Failure Notification

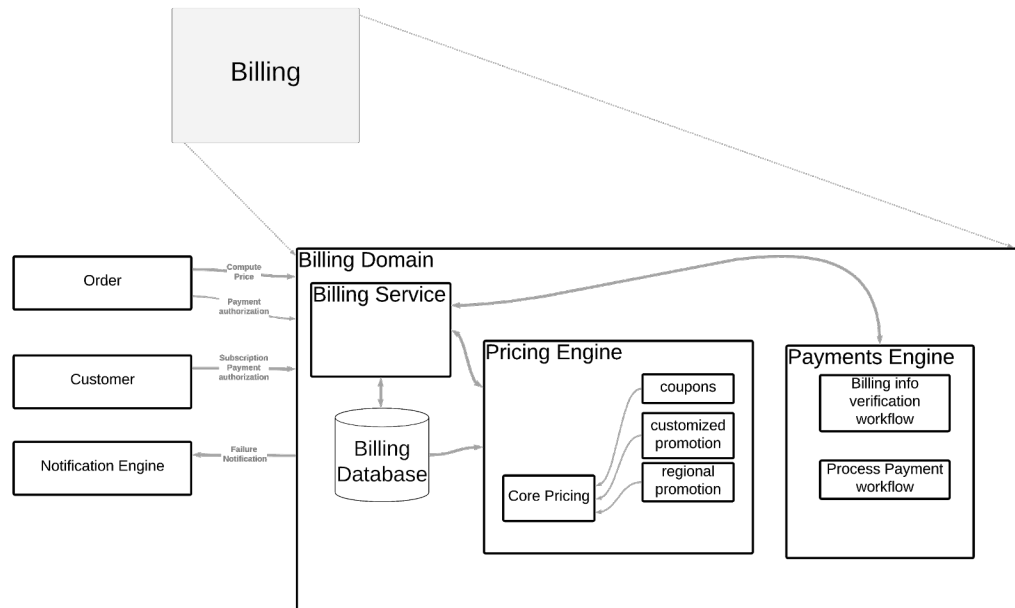**UI Component**
- Place Order

Billing

Decisions:
- [Separate payments engine for billing](#)

Characteristics:
- Durable
- Security
- Accurate
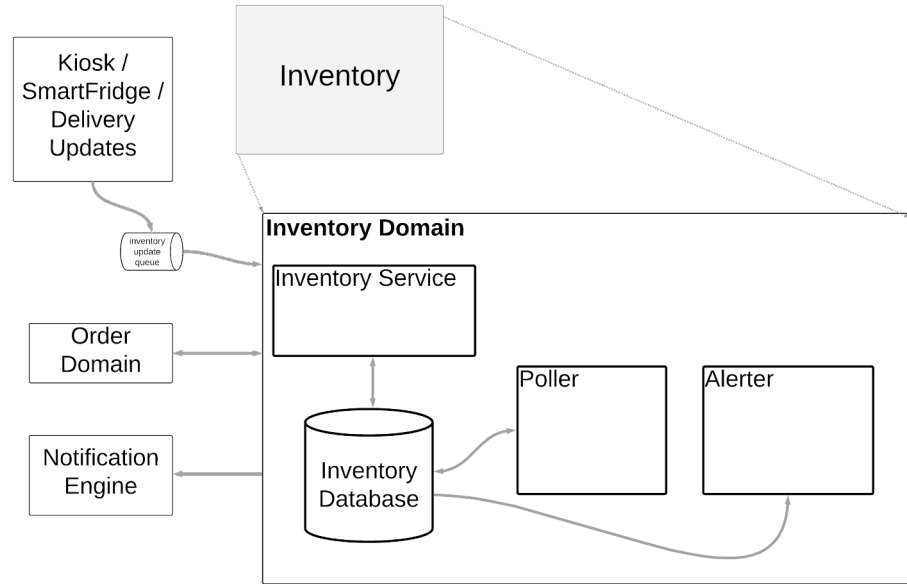- Customizable (Coupon and promotions)

Inventory

Decisions:
- [Use queue to update the inventory and external notification](#)
- [Stock monitoring and inventory update mechanism](#)

Characteristics:
- Trackable
- Accurate
- Consistent
- Customizable (3rd party vendor)

Kiosk / SmartFridge / Delivery Updates

Inventory

Inventory Domain

Inventory Service

inventory update queue

Order Domain

Notification Engine

Inventory Database

Poller
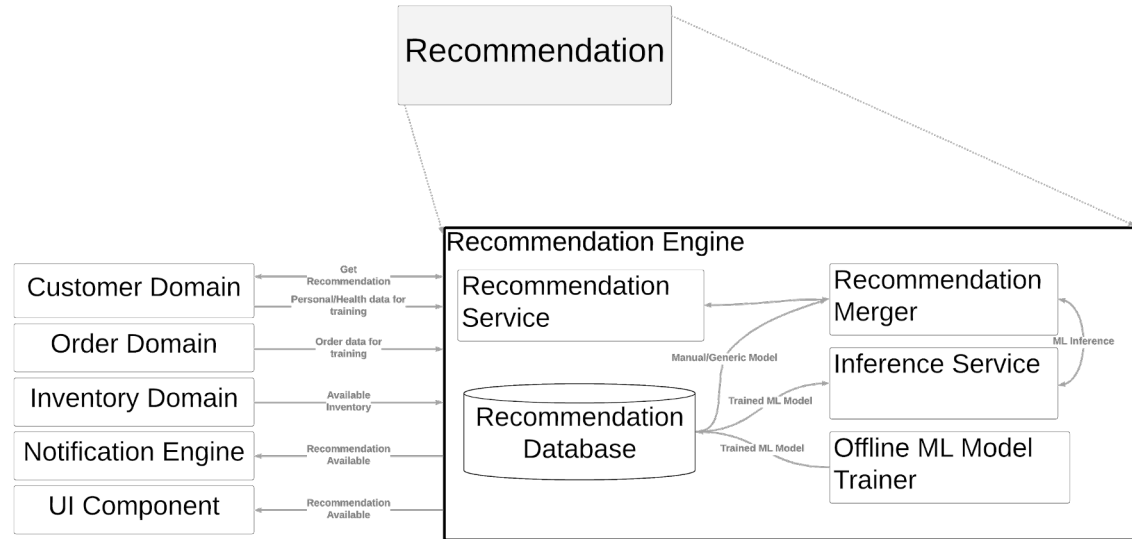
Alerter

selfdriventeam

Recommendations

Decisions:
- [Hybrid approach for recommendation component](#)
- [Recommendation engine is a batch system](#)

Characteristics:
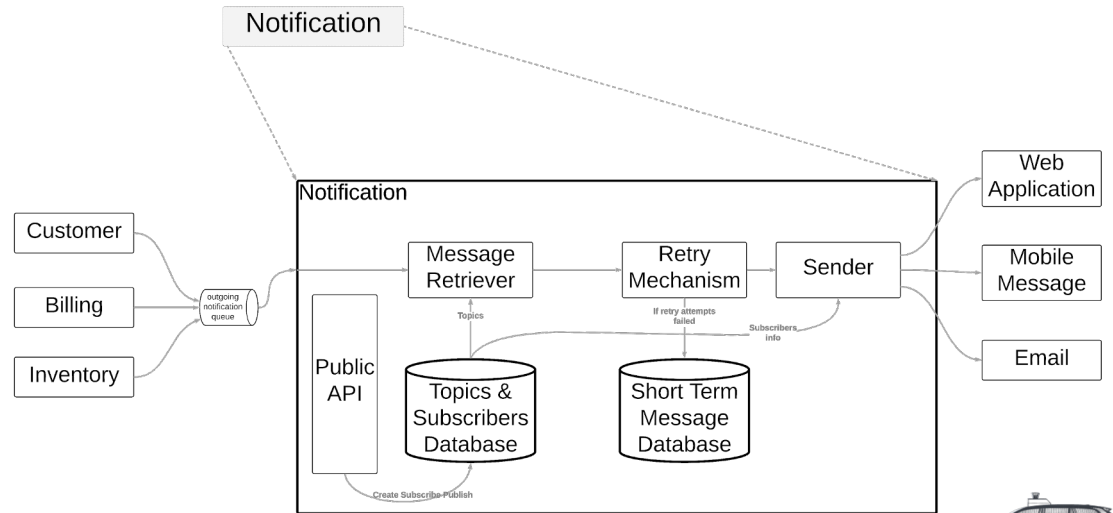- Security
- Customizable

Notifications

Decisions:
- [Use a centralized notification for external communication](#)

Characteristics:
- Scalable
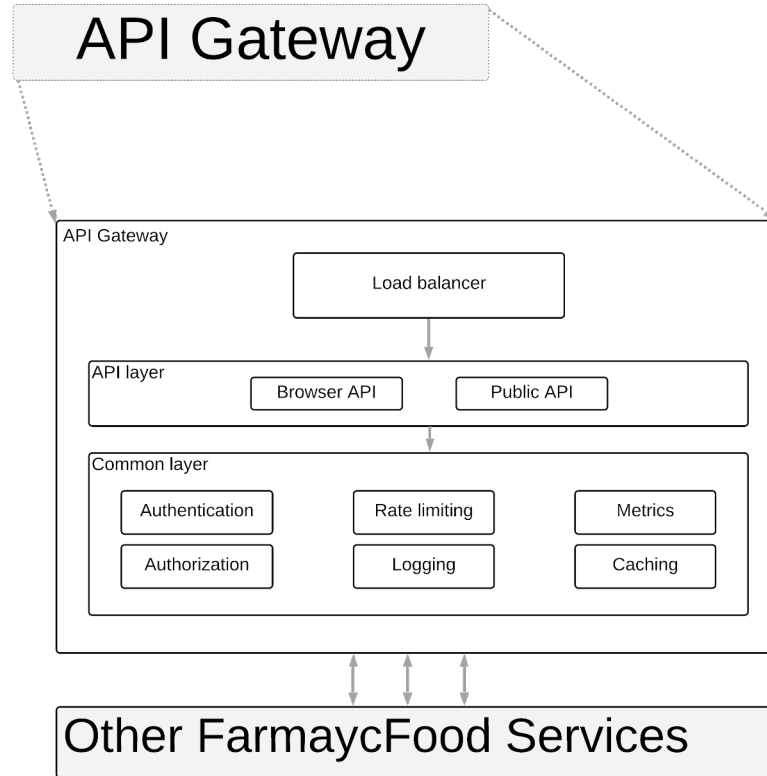- Highly Available
- Durable
- Reliable

Gateway/UI

Decisions:
- [Using External Identity Provider](#)
- [Component level authorization rules for access control](#)
- [Use mobile friendly web app](#)

Characteristics:
- Gateway:
  - Security
  - Available
  - Reliable
- UI
  - Responsive and reliable
  - Flexible
  - Accessibility

API Gateway

API Gateway

| Load balancer |

API layer

| Browser API | Public API |

Common layer

| Authentication | Rate limiting | Metrics |
| Authorization | Logging | Caching |

Other FarmaycFood Services

selfdriventeam

Ready for Today
Ready for the Future

- Time To Market
  - Casual Users
  - Subscription Users
  - Employees
- 3rd Party Integrations
  - POS
  - Partners

- Data Security
  - Financials
  - PII
- Future Growth
  - New Partners
  - Market Penetration
  - Geographic Expansion

selfdriventeam

Thank You

Additional Info
- O'Reilly Kata Page
- FarmacyFoods
- SelfDrivenTeam Github Repo

SelfDrivenTeam:
- Alex Torok
- Ankit Aggarwal
- Leon Rosenshein
- Shaw Xu
- Z Wang

selfdriventeam