ITSE 1331 Semester Group Project (Fall 2016)

Contents

List of Diagrams

The diagrams provided were developed by the systems analyst that consulted on this project and determined how the current system works and how the new system that your team is building should look.

Data Flow Model Diagrams

Based on the current system.

Assumptions

Based on the current system.

Entity Relationship Diagram

Based on the current system.

Business Problems

Based on the current system.

System Vision Document

Systems analyst documentation of the new system.

Activity Diagram

Systems analyst documentation of the new system.

Use Case Diagrams

Systems analyst documentation of the new system.

Complete Use Case Descriptions

Systems analyst documentation of the new system.

Brief Use Case Descriptions

Systems analyst documentation of the new system.

Domain Class Diagram
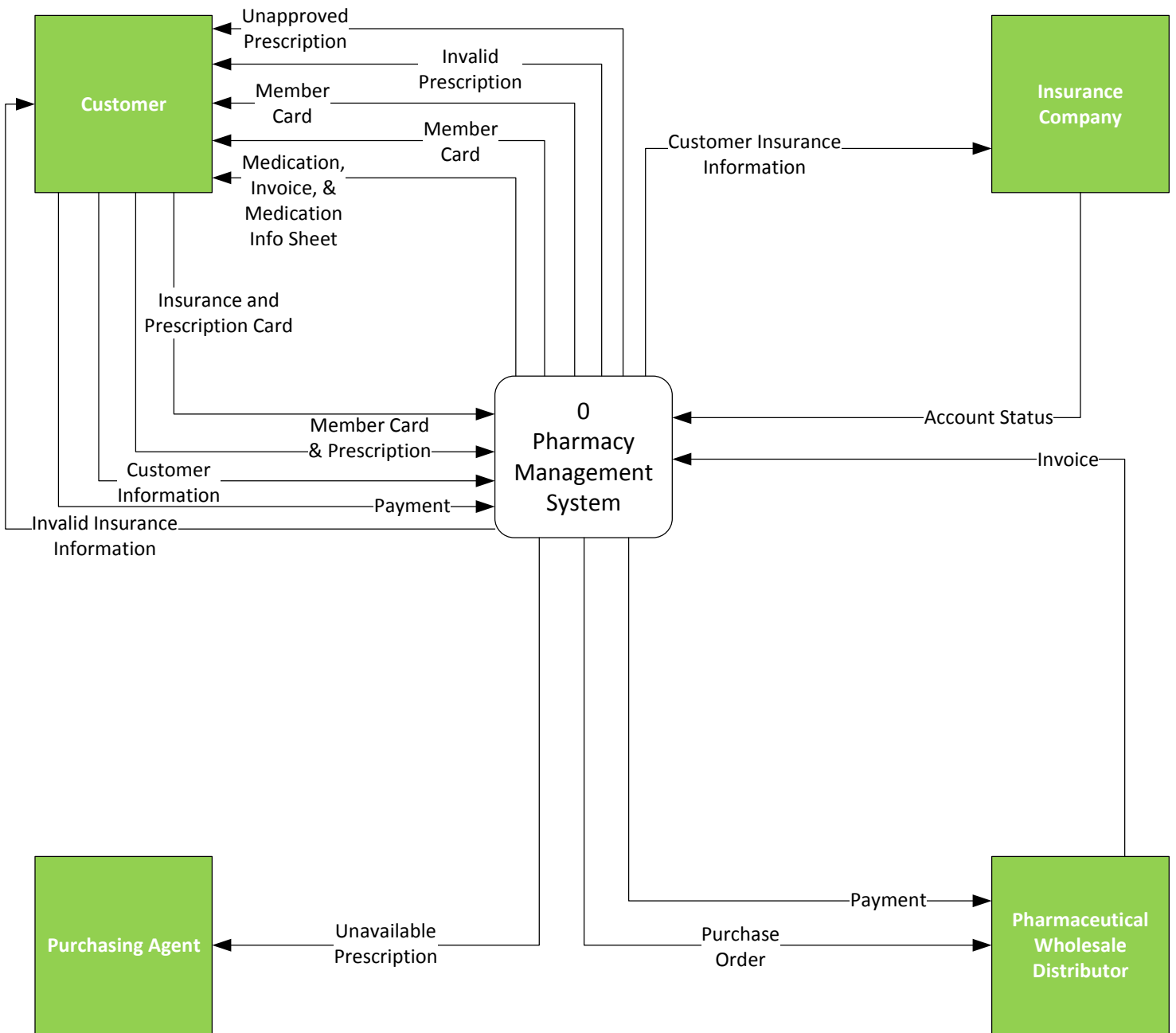
Systems analyst documentation of the new system.

Sequence Diagram

Systems analyst documentation of the new system.

Design Class Diagram

Systems analyst documentation of the new system.

Context Diagram

Context Diagram showing the "0 Pharmacy Management System" process at the center, connected to four external entities: Customer, Insurance Company, Purchasing Agent, and Pharmaceutical Wholesale Distributor.

Customer flows:
- Unapproved Prescription (to Customer)
- Invalid Prescription (to Customer)
- Member Card (to Customer)
- Member Card (to Customer)
- Medication, Invoice, & Medication Info Sheet (to Customer)
- Insurance and Prescription Card (to System)
- Member Card & Prescription (to System)
- Customer Information (to System)
- Payment (to System)
- Invalid Insurance Information (to Customer)

Insurance Company flows:
- Customer Insurance Information (to Insurance Company)
- Account Status (to System)

Pharmaceutical Wholesale Distributor flows:
- Invoice (to System)
- Payment (to Distributor)
- Purchase Order (to Distributor)

Purchasing Agent flows:
- Unavailable Prescription (to Purchasing Agent)

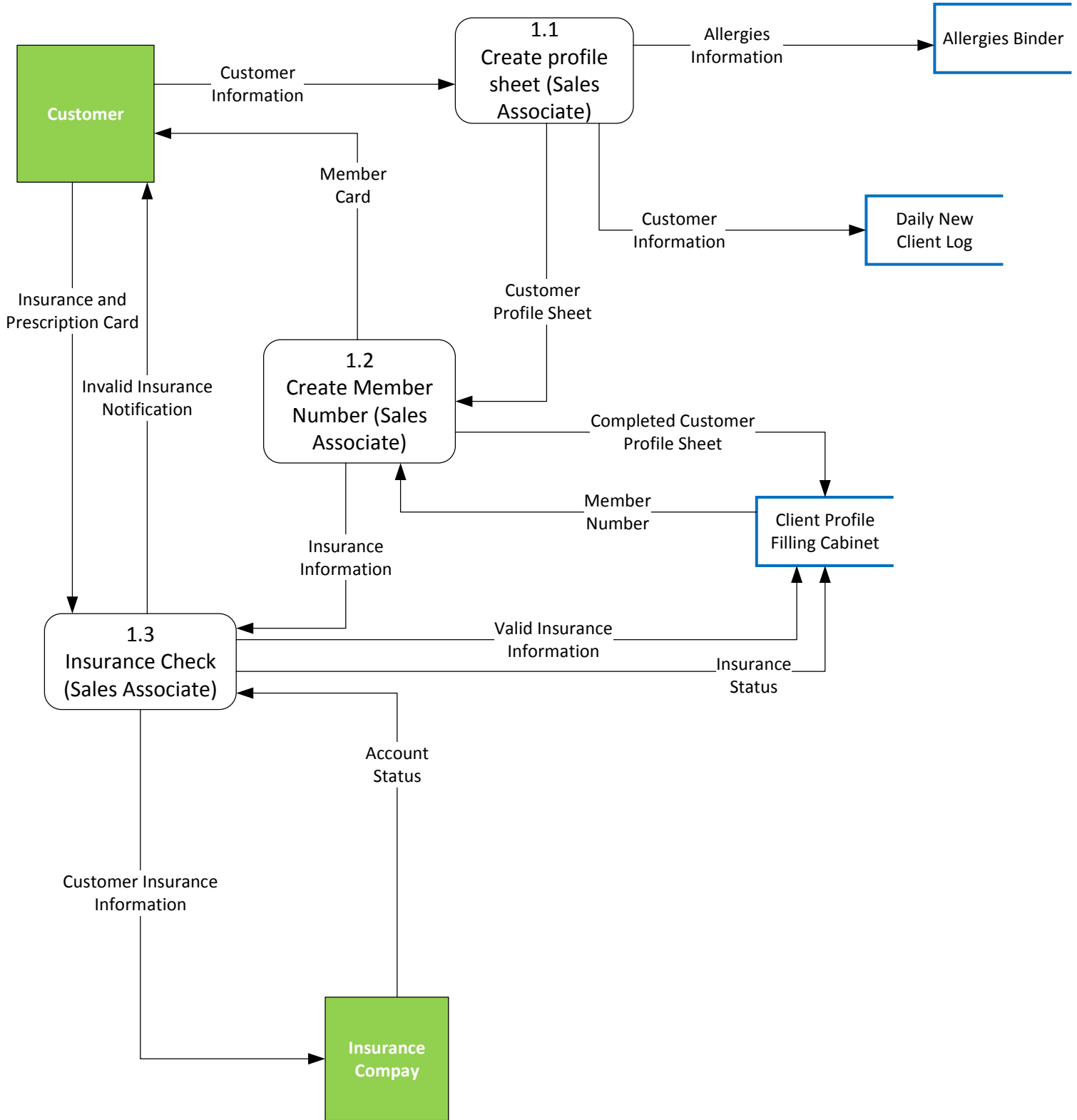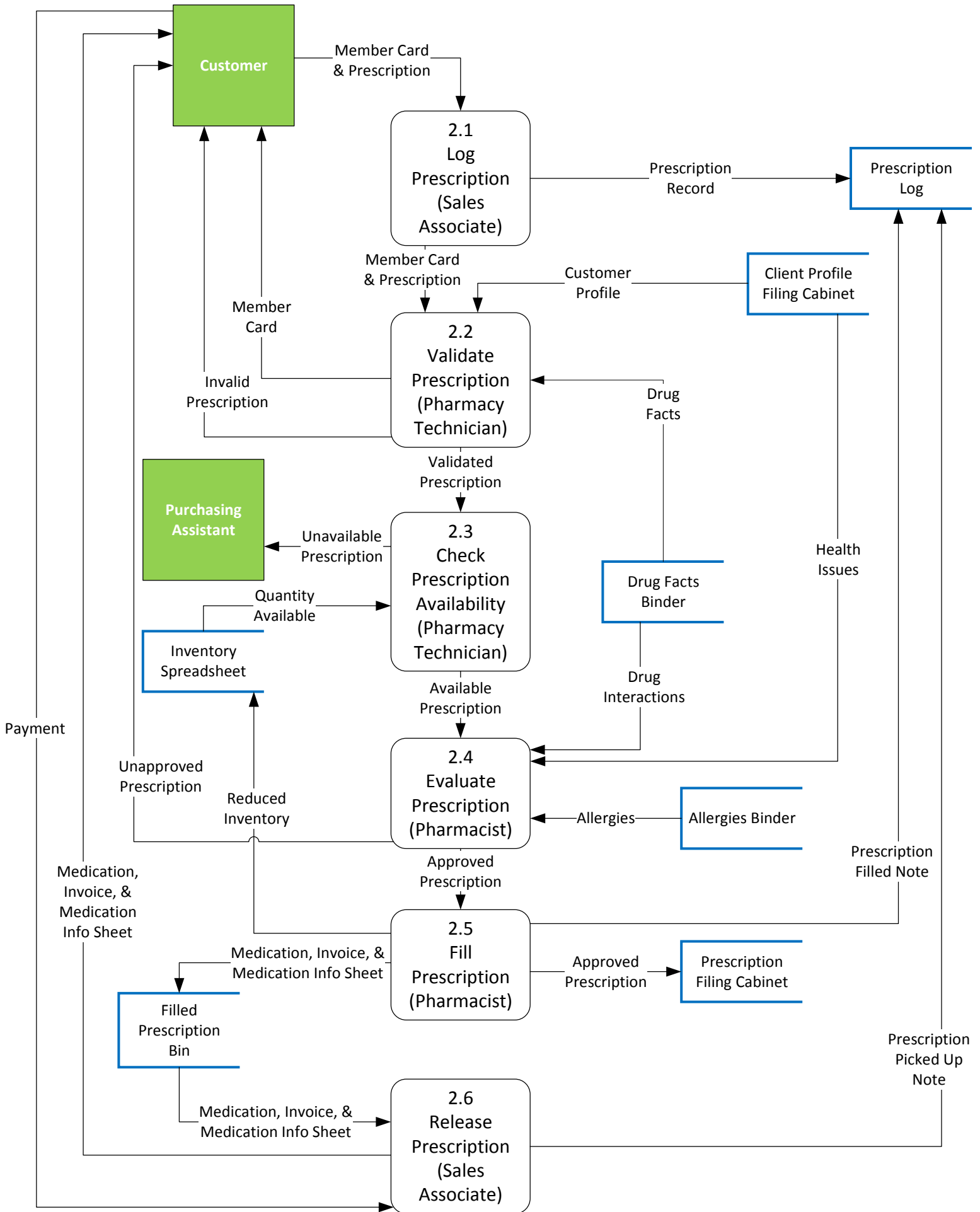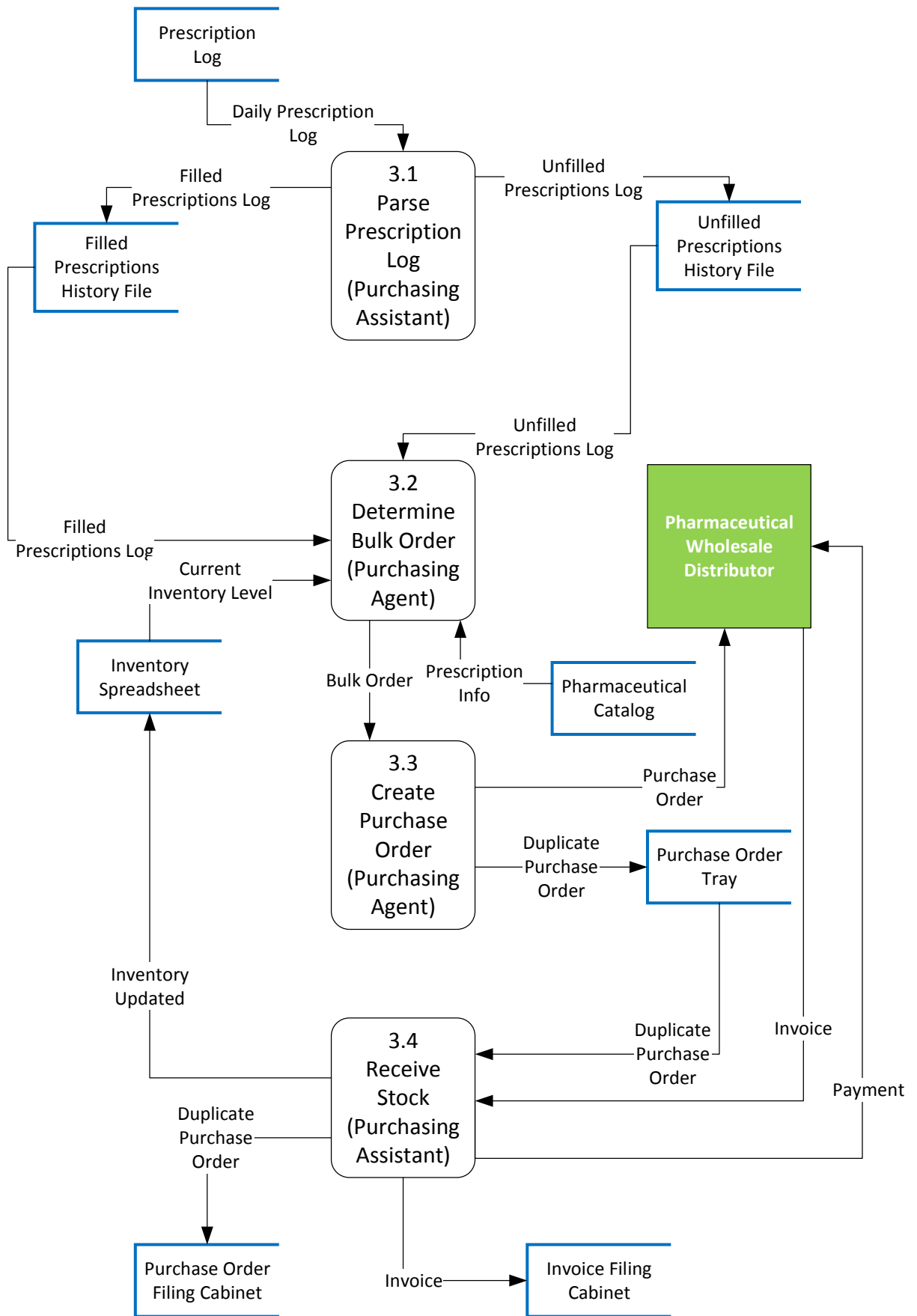# Level 0 Diagram

Level 1: Process 1.0 – Account Setup

Level 1: Process 2.0 – Prescription Fulfillment

Level 1: Process 3.0 – Bulk Purchasing

Prescription Log

Daily Prescription Log

Filled Prescriptions Log

3.1 Parse Prescription Log (Purchasing Assistant)

Unfilled Prescriptions Log

Unfilled Prescriptions History File

Filled Prescriptions History File

Unfilled Prescriptions Log

3.2 Determine Bulk Order (Purchasing Agent)

Filled Prescriptions Log

Current Inventory Level

Pharmaceutical Wholesale Distributor

Inventory Spreadsheet

Bulk Order

Prescription Info

Pharmaceutical Catalog

3.3 Create Purchase Order (Purchasing Agent)

Purchase Order

Duplicate Purchase Order

Purchase Order Tray

Inventory Updated

Duplicate Purchase Order

Invoice

3.4 Receive Stock (Purchasing Assistant)

Duplicate Purchase Order

Payment

Purchase Order Filing Cabinet
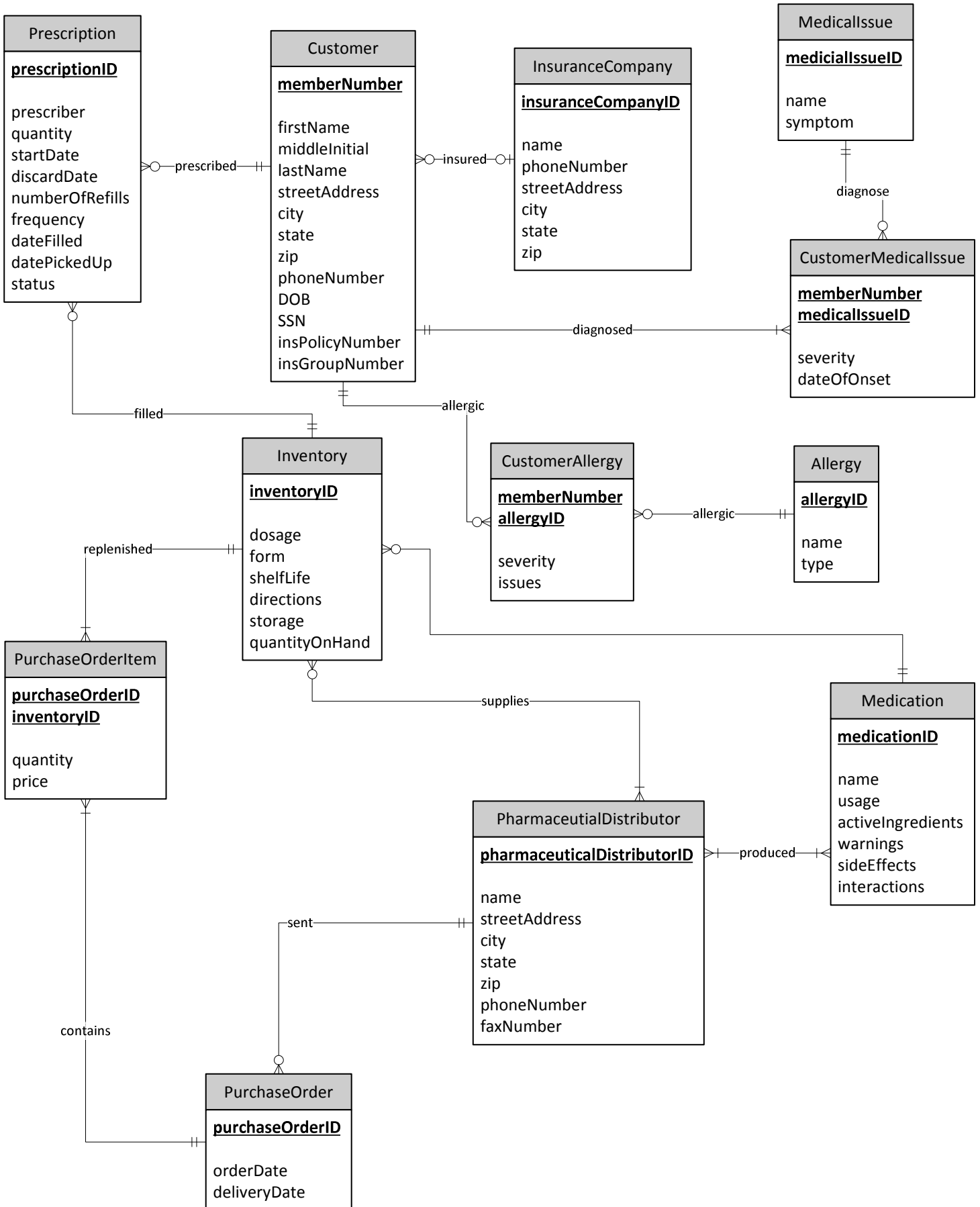
Invoice

Invoice Filing Cabinet

Assumptions

Level 1: Process 2.0 – Prescription Fulfillment

- Returning Membership Card to Customer (Process 2.2)
- Unavailable Prescription to Purchasing Assistant (Process 2.3)
- Release of Prescription by Sales Associate to customer (Process 2.6)
  - Collecting the Filled Prescription from the Filled Prescription Bin (Process 2.6)
  - Collecting the Payment from the customer (Process 2.6)
  - Prescription Picked Up by Customer (Process 2.6)
  - Logging Prescription Picked Up in Prescription Log (Process 2.6)

Level 1: Process 3.0 – Bulk Purchasing

- File Purchase Order in Purchase Order Filing Cabinet (Process 3.4)
- File Invoice in Invoice Filing Cabinet (Process 3.4)

ERD – Current System

Prescription
prescriptionID

prescriber
quantity
startDate
discardDate
numberOfRefills
frequency
dateFilled
datePickedUp
status

Customer
memberNumber

firstName
middleInitial
lastName
streetAddress
city
state
zip
phoneNumber
DOB
SSN
insPolicyNumber
insGroupNumber

InsuranceCompany
insuranceCompanyID

name
phoneNumber
streetAddress
city
state
zip

MedicalIssue
medicialIssueID

name
symptom

CustomerMedicalIssue
memberNumber
medicalIssueID

severity
dateOfOnset

Inventory
inventoryID

dosage
form
shelfLife
directions
storage
quantityOnHand

CustomerAllergy
memberNumber
allergyID

severity
issues

Allergy
allergyID

name
type

PurchaseOrderItem
purchaseOrderID
inventoryID

quantity
price

Medication
medicationID

name
usage
activeIngredients
warnings
sideEffects
interactions

PharmaceutialDistributor
pharmaceuticalDistributorID

name
streetAddress
city
state
zip
phoneNumber
faxNumber

PurchaseOrder
purchaseOrderID

orderDate
deliveryDate

Relationships: prescribed, insured, diagnose, diagnosed, filled, allergic, replenished, allergic, supplies, produced, sent, contains

Account Setup

- Customer data is stored in numerous locations, rather than in one location

- Redundant ways of identifying a customer is created

- Manually looking up the next member number is tedious, and error prone

Prescription Fulfillment

- Company doesn't verify insurance with customer to ensure it's still accurate

- Company doesn't verify insurance with insurance company to ensure it is still valid

- Pharmacy tech looks in too many places to validate a prescription which is error prone

- There is no provision for what to do if a medication is unavailable

- Customer has to resolve too many issues that the pharmacy could handle providing better customer service (issues with prescription being dealt with doctor by pharmacy)

- Manual inventory reduction is error prone due to miscalculations/transposed numbers

- Single bin for filled medications is unorganized and could result in loss medications

- No provision for if a medication is not picked up after being filled

- No provision for the process of picking up a prescription

- No provisions are made for a doctor to phone, fax, or electronically file a prescription

- Better privacy checks should be included to verify the person dropping off the prescription and picking it up is who they say they are

- No provision for if a member forgets their card

Bulk Purchasing

- Manual process for determining bulk order is error prone

- Information for bulk order is found in too many different places

- Purchase orders left in a tray are easily lost, resulting in loss of order proof

<div align="center">**Paisano Pharmacy System**
System Vision Document</div>

Paisano Pharmacy has fallen behind with adapting to current advances in business management. Most business operations today are purely automatized, but the pharmacy's current system is based on manual paperwork. Since the system has been paper based, we believe they have not been as efficient and user-friendly as they could be with a modern system. If the system converts to a paperless approach and embraces digital technology, they will be able to avoid various unnecessary tasks they currently have. This change would be beneficial for everyone, pharmacists as well as customers.

System Capabilities

This document identifies the required system capabilities at a high level. Later documents will specify the detailed requirements. These capabilities are:
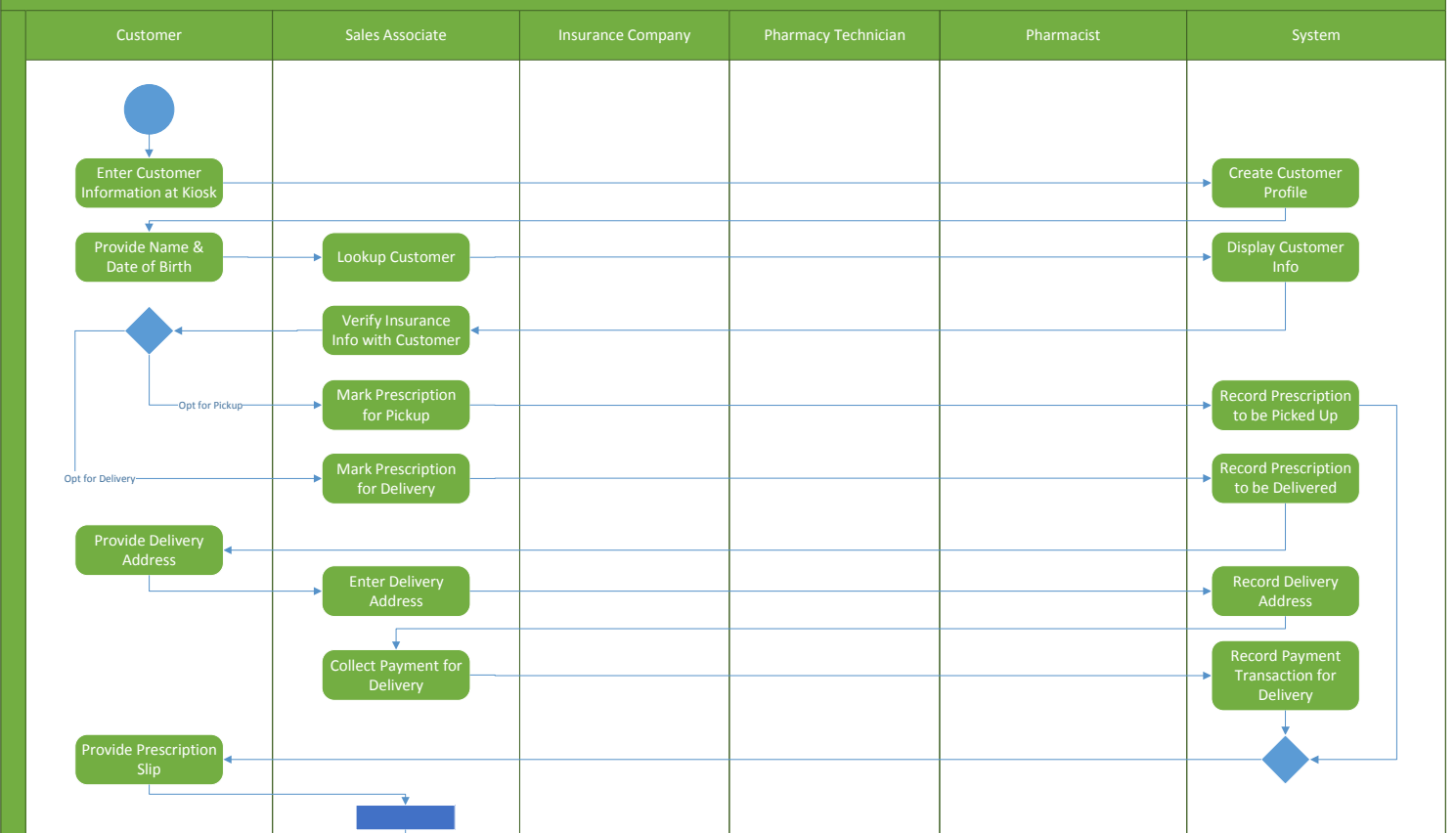
- Customer data management system for Pharmacists
    - Input customer data into single file which they can later update if necessary
    - Assign member number automatically
    - Ability to add other necessary notes relating their customers
- Customer features of the new system
    - In-store kiosk for new customers to input their information
    - Mobile application for customers to request prescription refills
    - Prescription delivery to customer
    - Point rewards system for frequent customers
- Pharmacist features of the new system
    - Shared database for all Paisano Pharmacy locations
    - Cloud prescription storage to scan and file prescription slips online
    - Provide options to receive a prescription – paper prescription slip, doctor's office fax, doctor's office e-prescription filling.
    - Automated inventory check, customer data check, and prescription/medication data check
    - Robotic prescription filling
    - Ability to order multiple prescriptions at one time
    - Check insurance every time a prescription order is placed

Business Benefits

The primary business benefit of these capabilities would be to reduce data redundancy, increase the pharmacy's efficiency, and improve the customer experience. The specific benefits include:

Cut cost by not creating the unnecessary physical member card
Reduce waiting time with new customer kiosk
Save time and money by not using paper forms when creating accounts
Ability to serve customers without social security numbers
Joining all pharmacy locations' databases together to increase efficiency and customer satisfaction
Increase speed of refilling prescription from new inventory management system
Reduce human errors in data entry and prescription fulfillment
Added security for customer files with regulated access of employees to system

Activity Diagram - Set up Contract

Customer	Sales Associate	Insurance Company	Pharmacy Technician	Pharmacist	System
●					
Enter Customer Information at Kiosk					Create Customer Profile
Provide Name & Date of Birth	Lookup Customer				Display Customer Info
◆	Verify Insurance Info with Customer				
Opt for Pickup	Mark Prescription for Pickup				Record Prescription to be Picked Up
Opt for Delivery	Mark Prescription for Delivery				Record Prescription to be Delivered
Provide Delivery Address					
	Enter Delivery Address				Record Delivery Address
	Collect Payment for Delivery				Record Payment Transaction for Delivery
Provide Prescription Slip					◆

Use Case Diagram – Pharmacy System
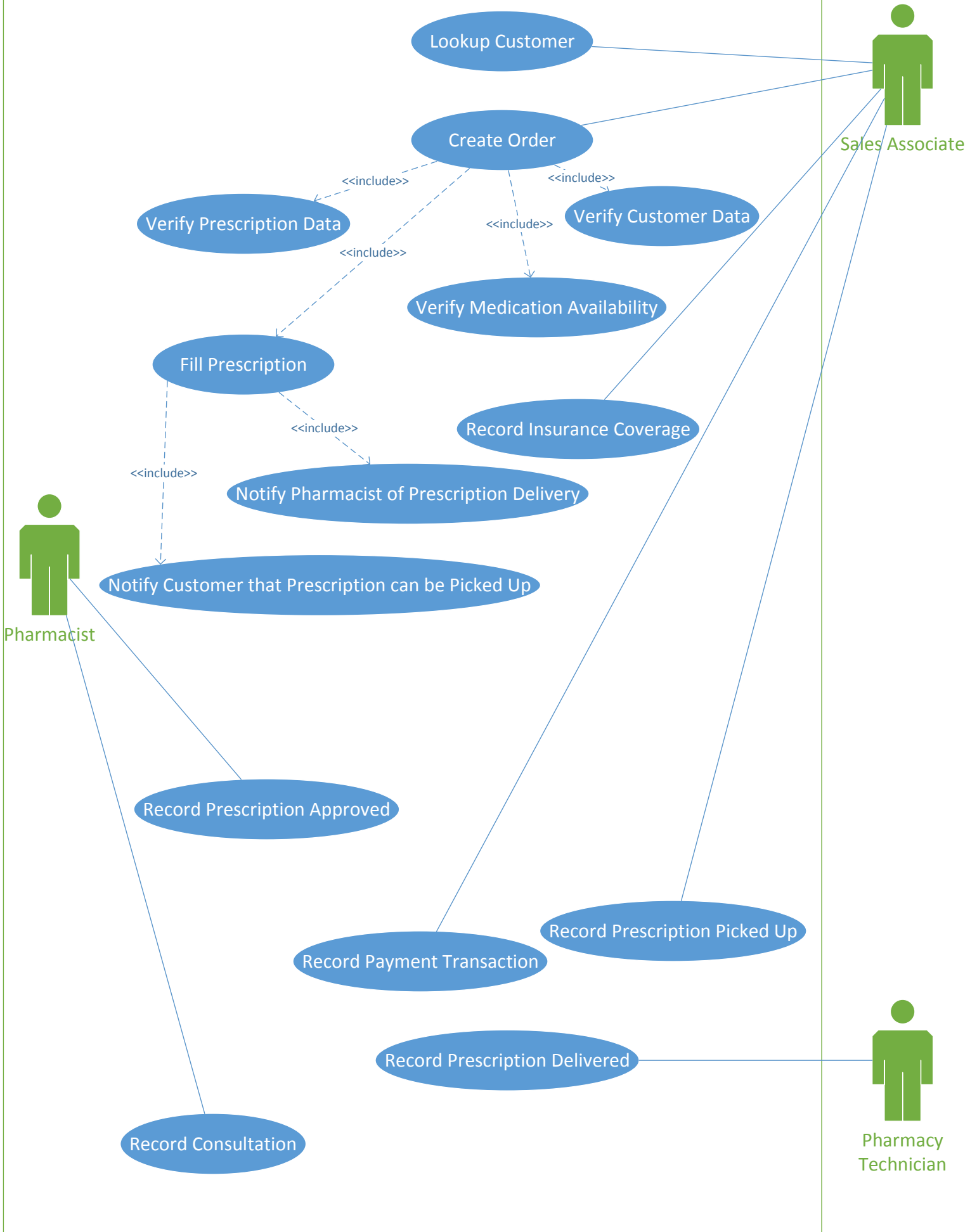
Use Case Diagram - Prescription Refill Mobile App
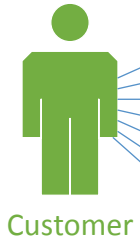
Customer

- View Customer Profile
- Update Customer Profile
- Order Prescription Refills
- Look up Order Status
- Contact Pharmacy
- View Points Earned
- View Prescription History

Use Case Diagram – Kiosk

Customer

Create Customer Profile

Update Customer Profile

Order Prescription Refill

Look Up Order Status

View Points Earned

View Prescription History

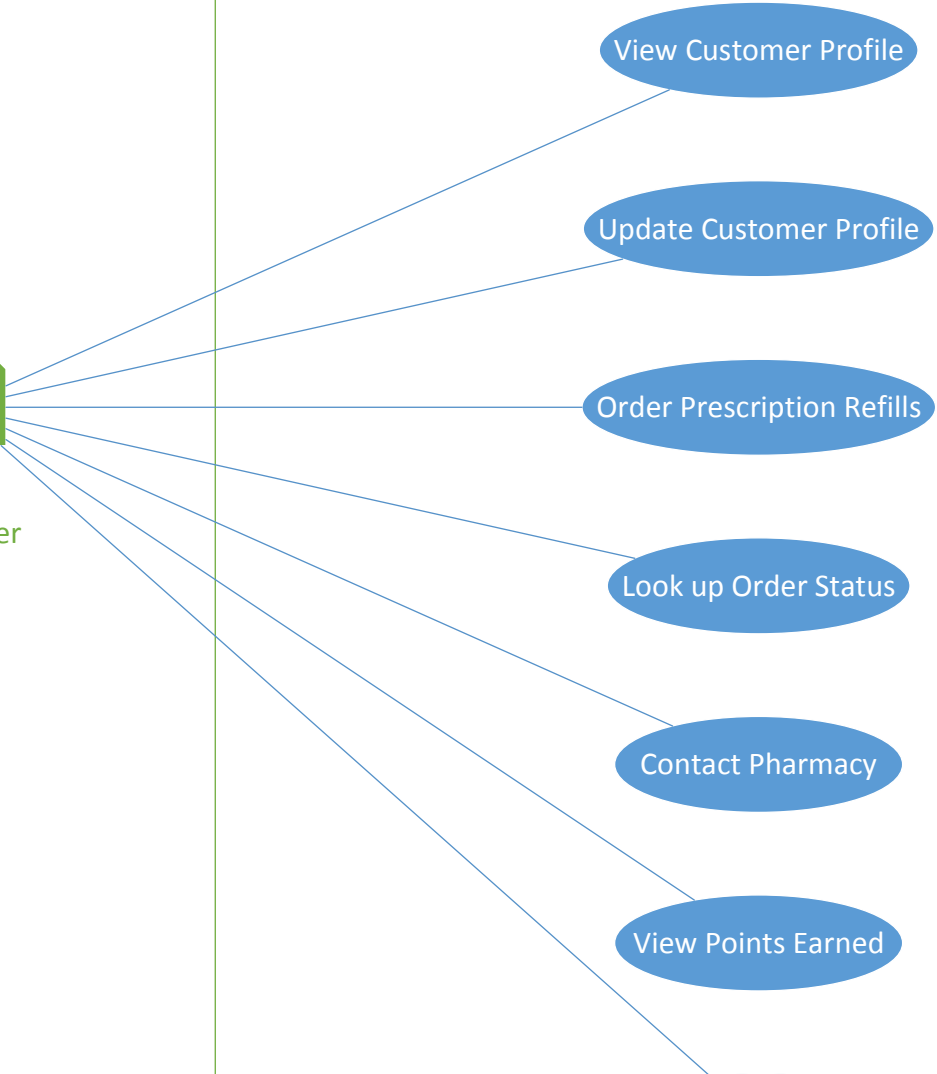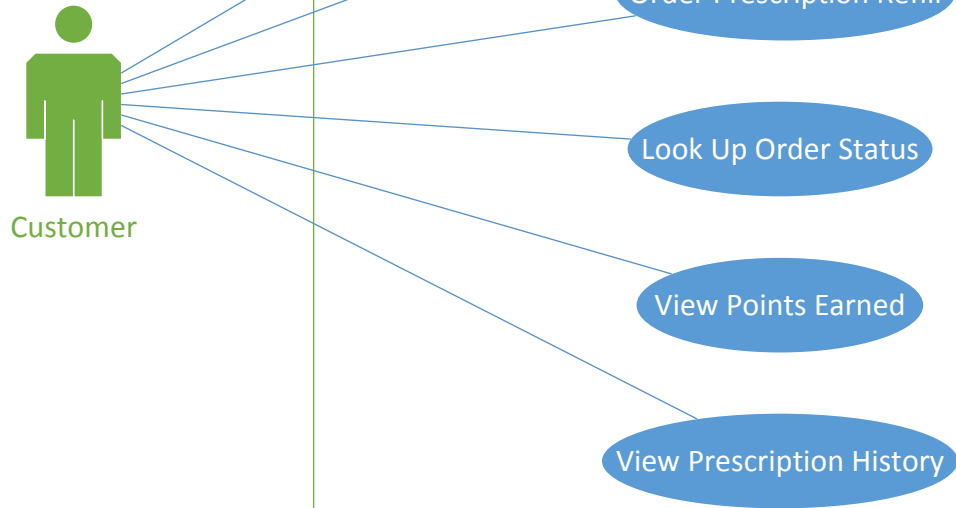# Use Case – Create Customer Profile

Use case name:	Create customer profile	
Scenario:	Create profile for new customer	
Triggering event:	New customer wants to set up account	
Brief description:	Customer creates new profile by entering their basic information and also their medical and insurance information	
Actors:	Customer	
Related use cases:	View customer profile; Update customer profile	
Stakeholders:	Sales and Marketing	
Preconditions:	Kiosk must be available; insurance company data must exist; database must be available	
Postconditions:	Customer must be created and saved; insurance must be validated; medical information must be created and saved; customer, medical, and insurance is associated	
Flow of activities:	Actor: 1 Customer indicated desire to create customer profile	System: 1.1 System creates new customer profile
		1.2 System prompts customer for medical information
	2 Customer enters medical information	2.1 System creates medical records
		2.2 System prompts customer for insurance information
	3 Customer enters insurance information	3.1 System creates insurance information
		3.2 System verifies insurance information
		3.3 System returns customer profile details
Exception conditions:	1.1 Basic customer information is invalid or incomplete 2.1 Medical information incomplete 3.2 Insurance isn't valid	

Use Case – Create Order

Use case name:	Create order
Scenario:	Create new order
Triggering event:	Customer wants to place a prescription order or a refill
Brief description:	Customer requests new order, the sales associate places the order
Actors:	Customer and Sales Associate
Related use cases:	Verify prescription data, verify customer data, verify prescription availability, fill prescription
Stakeholders:	Sales, Accounting, Pharmacist
Preconditions:	Customer profile must exist; customer must have prescription slip
Postconditions:	Prescription is entered and associated with customer; new order information is stored.
Flow of activities:	Actor: 1. Customer gives sales associate prescription slip.	System: 1.1 System prompts for customer information
	2. Sales associate looks up customer profile.	2.1 System locates customer profile
		2.2 System prompts sales associate for order
	3. Sales associate enters order information.	3.1 System creates order
		3.2 System adds prescription to order
		3.2 System prompts sales associate for submit
	4. Sales associate submits order.	4.1 System submits order
		4.2 System returns order confirmation
Exception conditions:	2.1 Customer profile cannot be found 3.2 Order information is incomplete	

Use Cases of Pharmacy System

Lookup Customer

Sales Associate enters customer first name and last name; system retrieves customer profile and displays profile on screen

Verify Customer Data

Sales Associate verifies DOB, address, phone number, and insurance information with customer; if any corrections are needed, Sales Associate enters the corrections and system verifies that the data entered is valid data.

Verify Prescription Data

Sales Associate enters prescription information; system verifies that a medication exists by that name and verifies that the entered dosage and drug form exist for that medication.

Verify Prescription Availability

Sales Associate enters prescription information; system checks to see if the medication is available to fill the prescription.

Fill Prescription

Sales Associate enters prescription information; system checks prescription data and prescription availability. After pharmacist has recorded prescription as approved, the system fills the prescription.

Record Insurance Information

Sales Associate enters insurance information, and the system records the coverage information after verifying that the insurance company is valid.

Notify Customer that Prescription can be Picked Up

System notifies customer that prescription is available for pick up.

Notify Pharmacist of Prescription Delivery

Pharmacy Technician records deliver of prescription to customer; system marks prescription as delivered and notifies the pharmacist via the system that the prescription has been delivered.

Record Payment Transaction

Sales Associate enters payment information, and the system records the payment transaction.

Record Prescription Picked Up

Sales Associate enters that that the prescription was picked up, and the system marks prescription as picked up.

Record Prescription Approved

Pharmacist reviews prescription, drug interactions, and allergies; pharmacist then records prescription approval and the system records and stores the information.

Record Consultation

Pharmacist enters consultation information, and the system records and stores the information.

Record Prescription Delivered

Pharmacy Technician enters deliver information, and the system records and stores the delivery information.

Use Cases for Customer Mobile App

View Customer Profile

User/actor enters username and password; system verifies credentials and the system retrieves and displays customer profile

Update Customer Profile

User/actor enters username and password; system verifies credentials and displays current record information; and customer enters new profile data, and the system replaces the old profile data with the new profile data.

Order Prescription Refills

User/actor selects prescription; the system retrieves order data; actor indicates refill desired, and the system creates a refill record for the order and customer.

Look Up Order Status

User/actor enters username and password; system verifies credentials and the system retrieves and displays order data and status.

Contact Pharmacy

User/actor selects method of contact; if user selects phone number, the app will initiate a phone call to the pharmacy. If user selects email address, they will be taken to an email composition screen to compose an email. If user selects form, they enter the information into a form.

View Points earned

User/actor enters username and password; system verifies credentials, and the system retrieves points earned by member.

View Prescription History

User/actor enters username and password; system verifies credentials, and the system retrieves past prescriptions by member.

Use Cases for Kiosk

Create Customer Profile

User/actor enters new profile data, creates a username and password, and the system assigns member number and creates a customer profile.

Update Customer Profile

User/actor enters username and password; system verifies credentials and display current record information; and customer enters new profile data, and the system replaces the old profile data with the new profile data.

Order Prescription Refills

User/actor selects prescription; the system retrieves order data; actor indicates refill desired, and the system creates a refill record for the order and customer.

Look Up Order Status

User/actor enters username and password; system verifies credentials and the system retrieves and displays order data and status.

View Points earned

User/actor enters username and password; system verifies credentials, and the system retrieves points earned by member.

View Prescription History

User/actor enters username and password; system verifies credentials, and the system retrieves past prescriptions by member.

Domain Class Diagram

Prescription
- prescriptionID
- prescriber
- startDate
- discardDate
- numberOfRefills
- frequency
- dateFilled
- datePickedUp
- status

prescribed

0..*   1..1

Customer
- memberNumber
- firstName
- lastName
- middleInitial
- streetAddress
- city
- state
- zip
- phoneNumber
- DOB
- SSN
- pointsEarned

insured

1..1   0..*

CustomerInsuranceCompany
- memberNumber
- insuranceCompanyID
- insPolicyNumber
- insGroupNumber

insures

0..*   1..1

InsuranceCompany
- insuranceCompanyID
- name
- phoneNumber
- streetAddress
- city
- state
- zip

1..1

diagnosed

CustomerMedicalIssue
- memberNumber
- medicalIssueID
- severity
- dateOfOnset

1..*

diagnose

0..*   1..1

MedicalIssue
- medicalIssueID
- name
- description

0..*

1..1

allergic ▶

0..*

CustomerAllergy
- memberNumber
- allergyID
- severity
- symptoms

allergic

0..*   1..1

Allergy
- allergyID
- name
- type

0..1

OrderDelivery
- orderDeliverID
- deliveryAddress
- deliveryPhoneNumber

1..1

1..1

delivered

Medication
- medicationID
- name
- usage
- activeIngredient
- warnings
- sideEffects
- interactions

checked

1..1   0..*

Item
- itemID
- dosage
- form
- shelfLife
- directions
- storage
- quantitiyOnHand
- pricePerUnit

1..1

Order
- orderID
- dateOrdered
- pickupMethod
- paymentMethod
- consultationDate

required

1..1   1..*

PaymentTransaction
- receiptNumber
- amount
- paymentType
- datePaid

1..1

includes

1..*

includes

0..*

OrderItem
- orderID
- itemID
- prescriptionApproved

Sequence Diagram – Create Order

Design Class Diagram

<<controller>> Order Handler

+lookUpCustomer()
+pickUpMethod()
+recordPrescription()
+completeOrder()
+recordPayment()

Prescription

-prescriptionID : Integer{key}
-prescriber : String
-startDate : Date
-discardDate : Date
-numberOfRefills : Integer
-frequency : String
-dateFilled : Date
-datePickedUp : Date
-status : String

+recordPrescription()

Customer

-memberNumber : Integer{key}
-firstName : String
-lastName : String
-middleInitial : String
-streetAddress : String
-city : String
-state : String
-zip : Integer
-phoneNumber : Integer
-DOB : Date
-SSN : Integer
-pointsEarned : Integer

+verifyCustomer()
+createOrder()

CustomerInsuranceCompany

-memberNumber : Integer
-insuranceCompanyID : Integer
-insPolicyNumber : String
-insGroupNumber : String

InsuranceCompany

-insuranceCompanyID : String{key}
-name : String
-phoneNumber : Integer
-streetAddress : String
-city : String
-state : String
-zip : Integer

+verifyInsurance()

CustomerMedicalIssue

-memberNumber : Integer{key}
-medicalIssueID : String{key}
-severity : String
-dateOfOnset : Date

MedicalIssue

-medicalIssueID : String{key}
-name : String
-description : String

CustomerAllergy

-memberNumber : Integer{key}
-allergyID : String{key}
-severity : String
-symptoms : String

Allergy

-allergyID : String{key}
-name : String
-type : String

OrderDelivery

-orderDeliverID
-deliveryAddress
-deliveryPhoneNumber

Medication

-medicationID : String{key}
-name : String
-usage : String
-activeIngredient : String
-warnings : String
-sideEffects : String
-interactions : String

+getMedName()
+updateQty()

Item

-itemID : Integer{key}
-dosage : String
-form : String
-shelfLife : String
-directions : String
-storage : String
-quantitiyOnHand : Integer
-pricePerUnit : Decimal

+getPricePerUnit()

Order

-orderID : Integer
-dateOrdered : Date
-pickupMethod : String
-paymentMethod : String
-consultationDate : Date

+pickUpMethod()
+addItem()
+completeOrder()
+makePayment()

PaymentTransaction

-receiptNumber : Integer{key}
-amount : Decimal
-paymentType : String
-datePaid : Date

+createPayment()

OrderItem

-orderID : Integer
-itemID : Integer
-prescriptionApproved : String