
MEDICINE DELIVERY MANAGEMENT SYSTEM

Medicine Delivery Management System

Submitted To
Kishan Kumar Ganguly
Lecturer
Institute of Information Technology,
University of Dhaka

Submitted By
BSSE 0903
BSSE 0906
BSSE0908
BSSE0917
BSSE0919
BSSE0927

Date of Submission: December 3, 2019



Institute of Information Technology
University of Dhaka

Contents

Contents

ARCHITECTURAL DESIGN	4
Archetype	5
Analyzing Class	5
Refining the Architecture into Components	6
Instantiations of the System	6
COMPONENT-LEVEL DESIGN	7
Analysis Class – Pharmacy	8
Message Passing (Pharmacy)	9
Analysis Class – Customer	10
Message Passing (Customer)	11
Analysis Class – Deliverer	12
Message Passing (Deliverer)	13
Analysis Class – Medicine Order	14
Message Passing (Medicine Order)	15
Message Passing (Feedback)	17
State Chart Fragment for Classes	18
Deployment diagram	23
Activity diagram	24
Chapter Three: User Interface Design	35
Interface Analysis	35
User Analysis	35
Task Analysis	37

Table of Figures

Figure 1 Architectural Context Diagram	4
Figure 2 Archytype	5
Figure 3 Instantiations	6
Figure 4 Analysis Class – Pharmacy	8
Figure 5 Message Passing (Pharmacy)	9
Figure 6 Analysis Class – Customer	10
Figure 7 Message Passing (Customer)	11
Figure 8 Analysis Class – Deliverer	12
Figure 9 Message Passing (Deliverer)	13
Figure 10 Analysis Class – Medicine Order	14
Figure 11 Message Passing (Medicine Order)	15
Figure 12 Analysis Class (Feedback)	16
Figure 13 Message Passing (Feedback)	17
Figure 14 Pharmacy State Chart	18
Figure 15 Customer	19
Figure 16 Deliverer	20
Figure 17 Product	21
Figure 18 Feedback	22
Figure 19 Deployment	23
Figure 20 communicateTopharmacy	24
Figure 21 communicationToCustomer	25
Figure 22 communicationTodeliverer	26
Figure 23 Deliver Medicine	27
Figure 24 Medicine Info Upload	28
Figure 25 Order Notification	29
Figure 26 Prescription Completion	30
Figure 27 Search Medicine	31
Figure 28 Search User Location	32
Figure 29 Sign In	33
Figure 30 Sign Up	34
Figure 31 Welcome Page	40
Figure 32 Sign In Page	41
Figure 33 Sign Up Page	42
Figure 34 Customer	44
Figure 35 Pharmacy Owner	44
Figure 36 Medicine	45
Figure 37 Profile	46
Figure 38 Upload Prescription	48
Figure 39 Location	48
Figure 40 Search Drug	49

ARCHITECTURAL DESIGN

At the architectural design level, a software architect uses an architectural context diagram (ACD) to model the manner in which software interacts with entities external to its boundaries.

Referring to the figure, systems that interoperate with the target system (the system for which an architectural design is to be developed) are represented as

- **Superordinate systems**—those systems that use the target system as part of some higher-level processing scheme.
- **Subordinate systems**—those systems that are used by the target system and provide data or processing that are necessary to complete target system functionality.
- **Peer-level systems**—those systems that interact on a peer-to-peer basis (i.e., information is either produced or consumed by the peers and the target system).
- **Actors**—entities (people, devices) that interact with the target system by producing or consuming information that is necessary for requisite processing. Each of these external entities communicates with the target system through an interface

1. Architectural Context Diagram

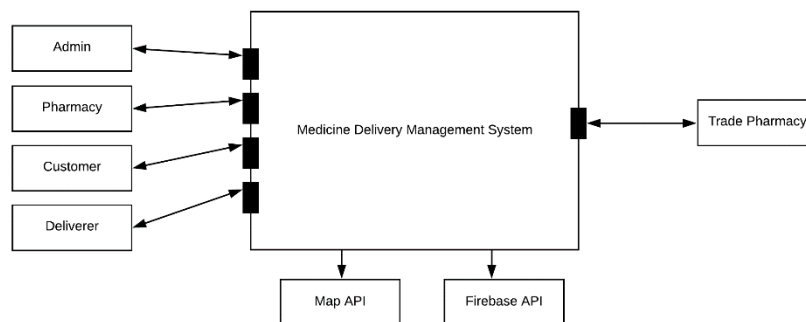


Figure 1 Architectural Context Diagram

Actors: Pharmacy Owner, Customer, Admin and deliverer they are the actors. They use the system by producing/consuming information through the system

Super Ordinate: There is no super ordinate systems.

Sub Ordinate: Map API, Firebase API are the sub ordinate system, because our MediPro depends on these APIs.

Peer-Level Systems: Trade Pharmacy and MediPro are fully independent. But they create interface on their system

Archetype

An archetype is a class or pattern that represents a core abstraction that is critical to the design of an architecture for the target system. In general, a relatively small set of archetypes is required to design even relatively complex systems. The target system architecture is composed of these archetypes, which represent stable elements of the architecture but may be instantiated many different ways based on the behavior of the system.

Analyzing Class

Pharmacy: An abstraction that represents receive order, give medicine information and serve the deliverer.

Customer: An abstraction that represents order the medicine through the system.

Deliverer: An abstraction that deliver the medicine

Feedback: An abstraction where user can give feedback using this app.

MedicineOrder: An abstraction where keep medicine information and store updated medicine information.

Archy Type

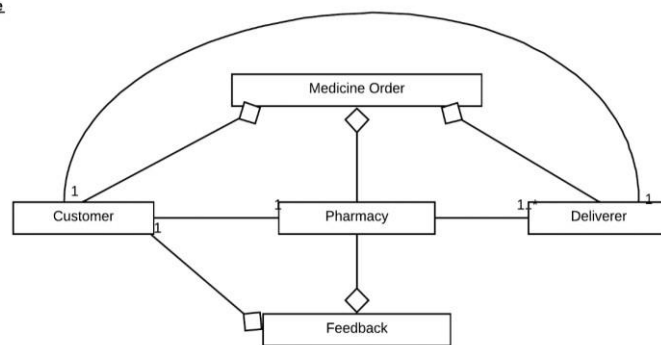


Figure 2 Archytype

Refining the Architecture into Components

As the software architecture is refined into components, the structure of the system begins to emerge. But how are these components chosen? In order to answer this question, you begin with the classes that were described as part of the requirements model. These analysis classes represent entities within the application (business) domain that must be addressed within the software architecture. Hence, the application domain is one source for the derivation and refinement of components. Another source is the infrastructure domain. The architecture must accommodate many infrastructure components that enable application components but have no business connection to the application domain. For example, memory management components, communication components, database components, and task management components are often integrated into the software architecture.

Instantiations of the System

The architectural design that has been modeled to this point is still relatively high level. The context of the system has been represented, archetypes that indicate the important abstractions within the problem domain have been defined, the overall structure of the system is apparent, and the major software components have been identified. However, further refinement (recall that all design is iterative) is still necessary.

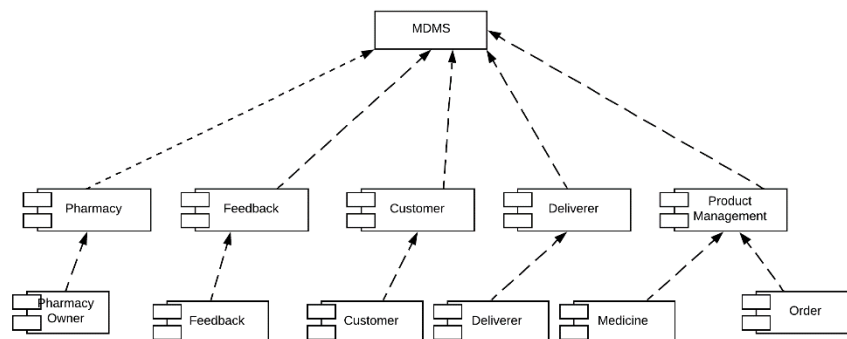


Figure 3 Instantiations

COMPONENT-LEVEL DESIGN

Component-level design occurs after the first iteration of architectural design has been completed. At this stage, the overall data and program structure of the software has been established. The intent is to translate the design model into operational software. But the level of abstraction of the existing design model is relatively high, and the abstraction level of the operational program is low. The translation can be challenging, opening the door to the introduction of subtle errors that are difficult to find and correct in later stages of the software process. In a famous lecture, Edsger Dijkstra, a major contributor to our understanding of software design, stated.

Analysis Class – Pharmacy

Analysis Class (1)

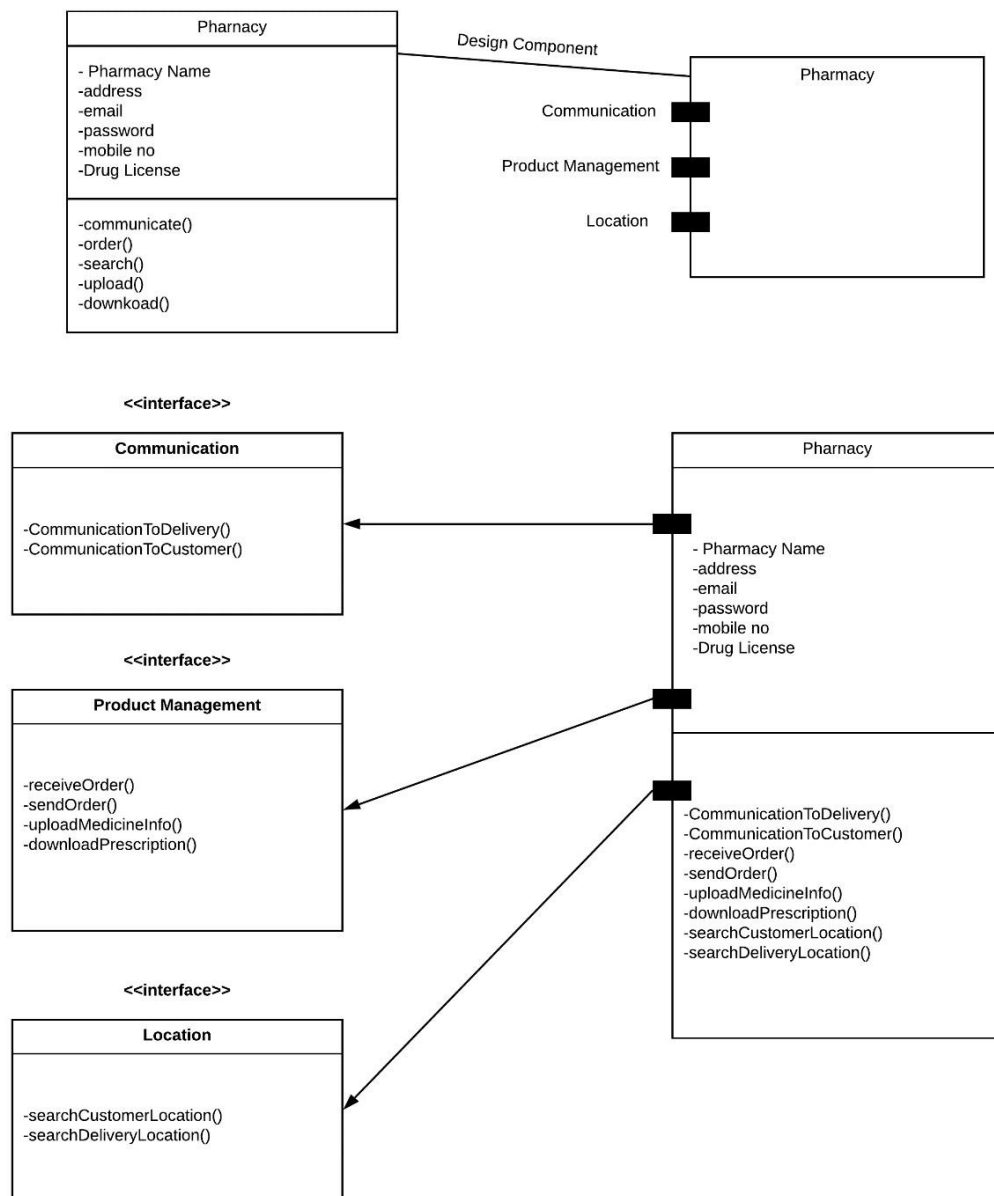


Figure 4 Analysis Class – Pharmacy

Message Passing (Pharmacy)

3a Message Passing

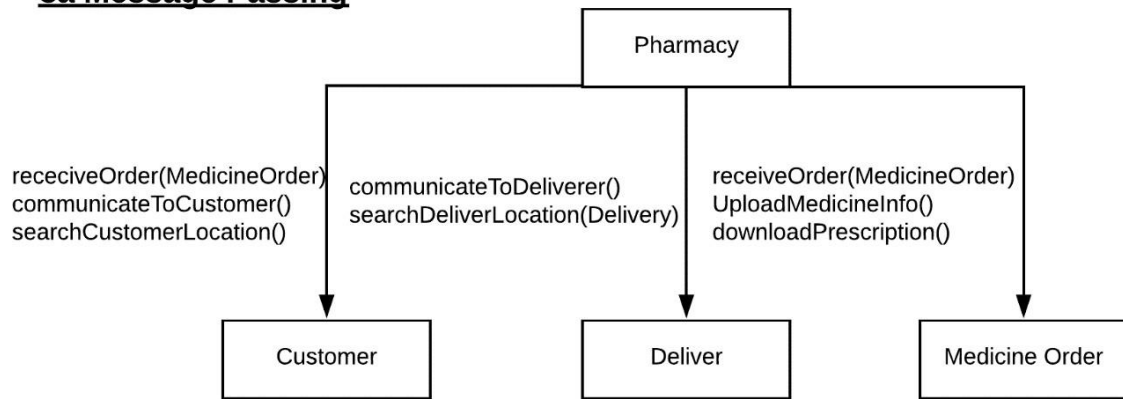


Figure 5 Message Passing (Pharmacy)

Analysis Class – Customer

Analysis Class (2)

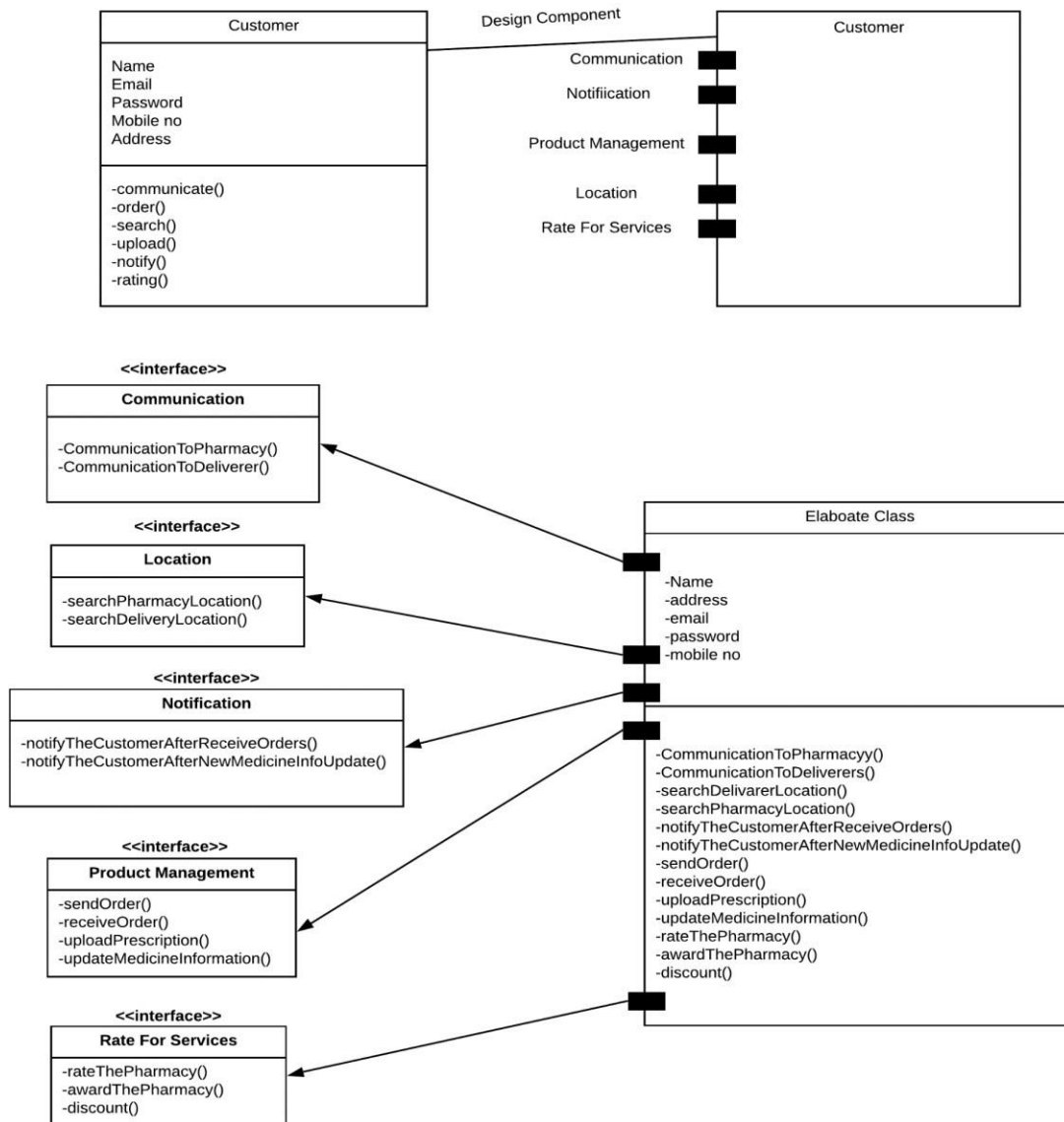


Figure 6 Analysis Class – Customer

Message Passing (Customer)

3a Message Passing

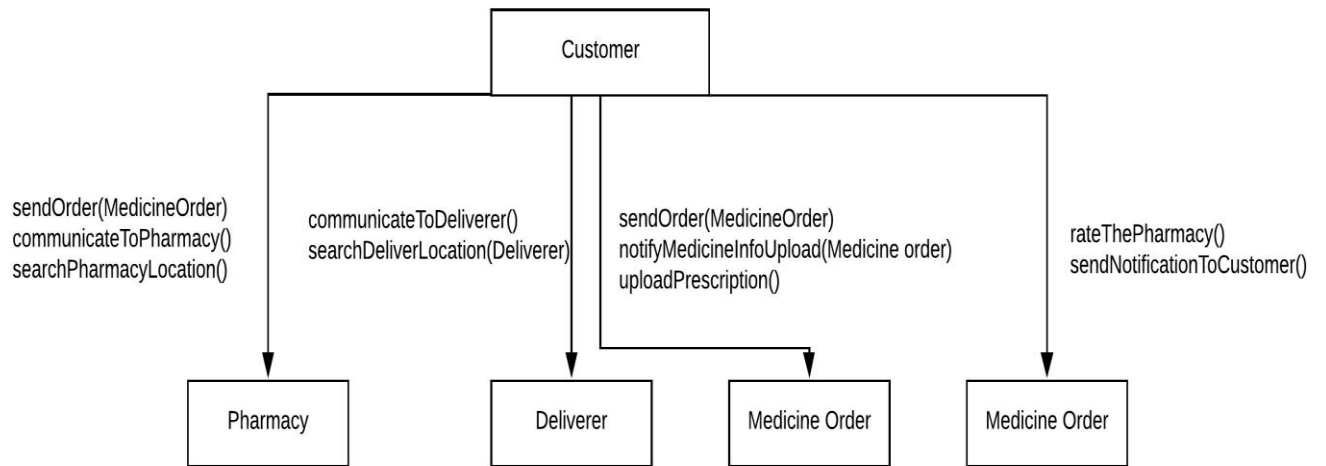


Figure 7 Message Passing (Customer)

Analysis Class – Deliverer

Analysis Class (3)

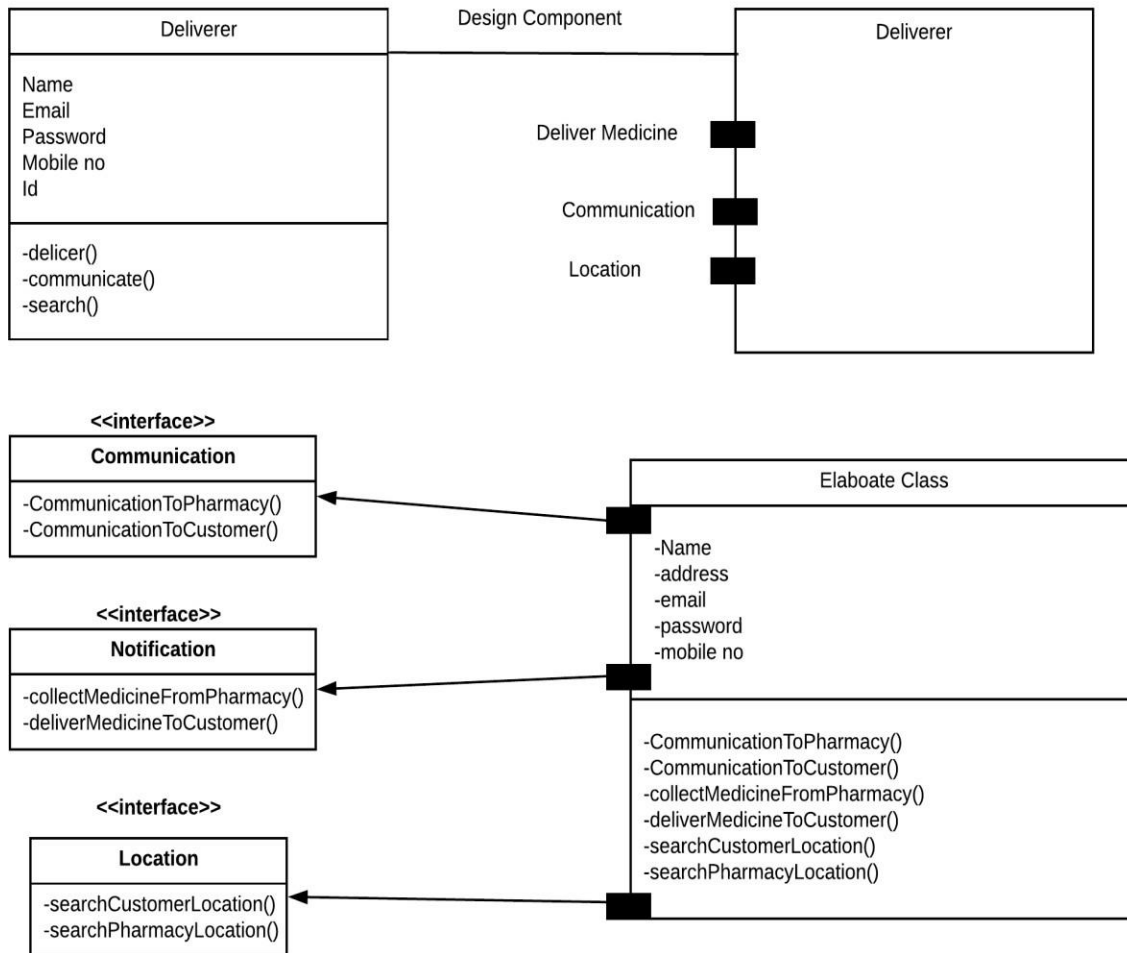


Figure 8 Analysis Class – Deliverer

Message Passing (Deliverer)

3a Message Passing

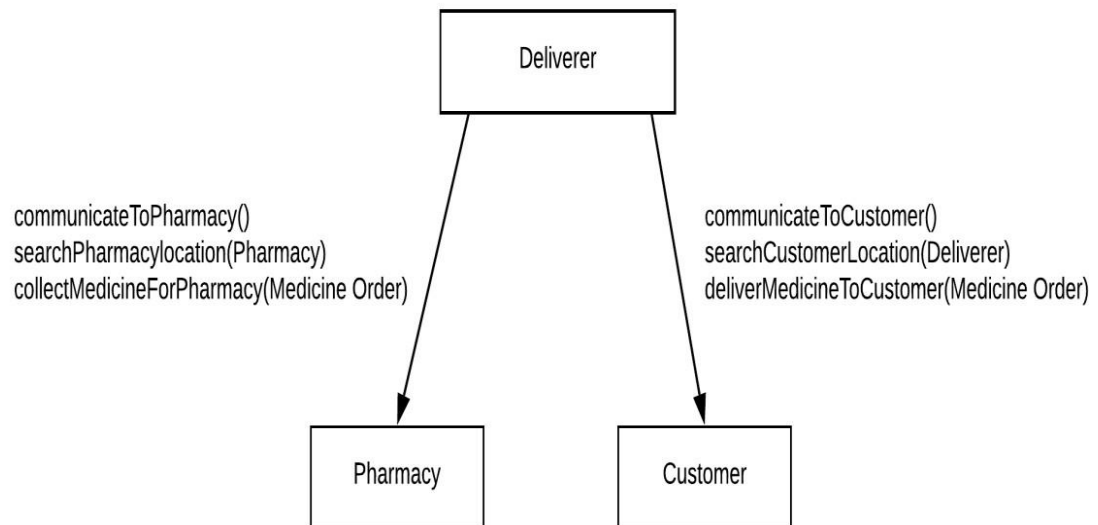


Figure 9 Message Passing (Deliverer)

Analysis Class – Medicine Order

Analysis Class (4)

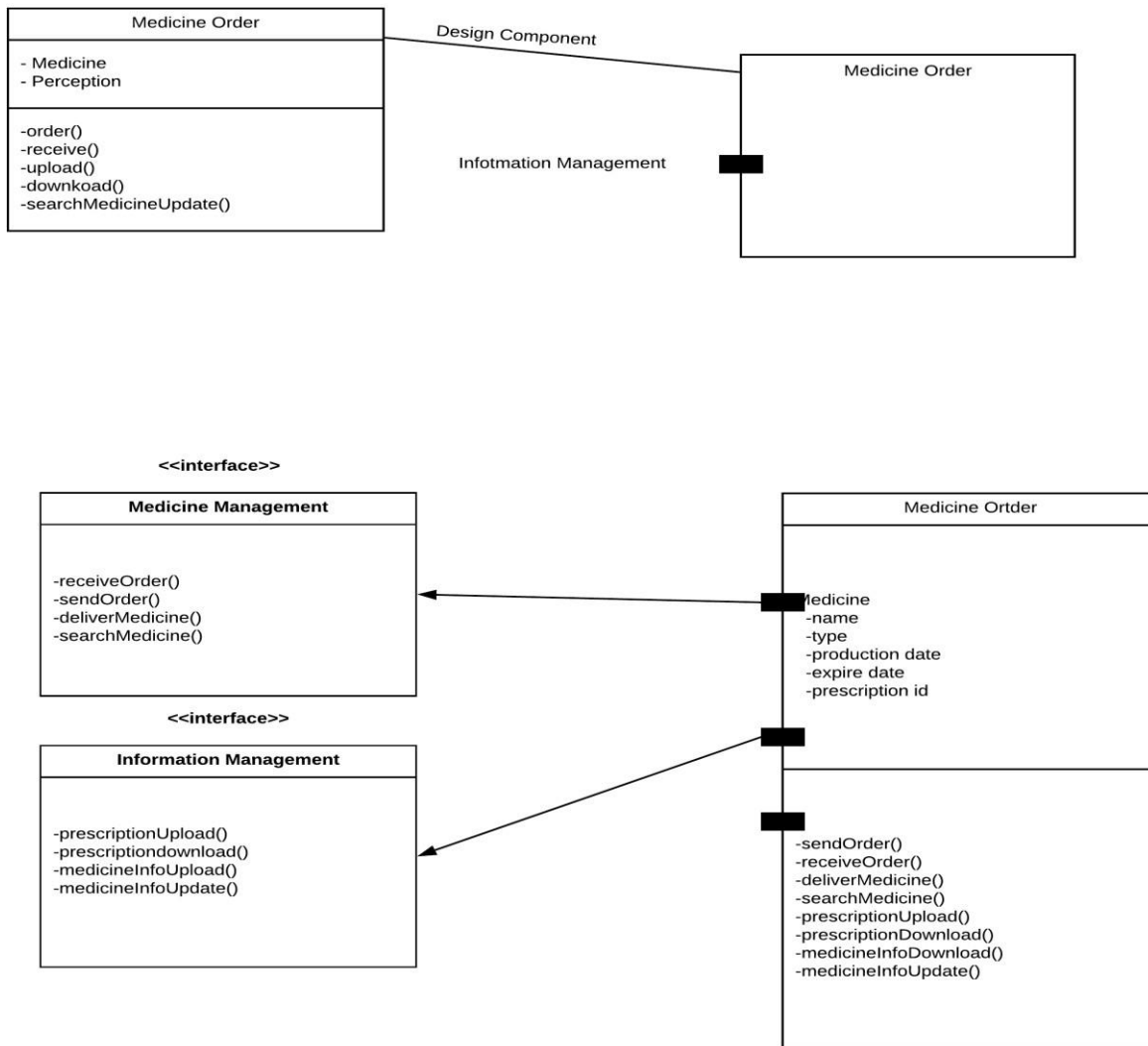


Figure 10 Analysis Class – Medicine Order

Message Passing (Medicine Order)

3a Message Passing

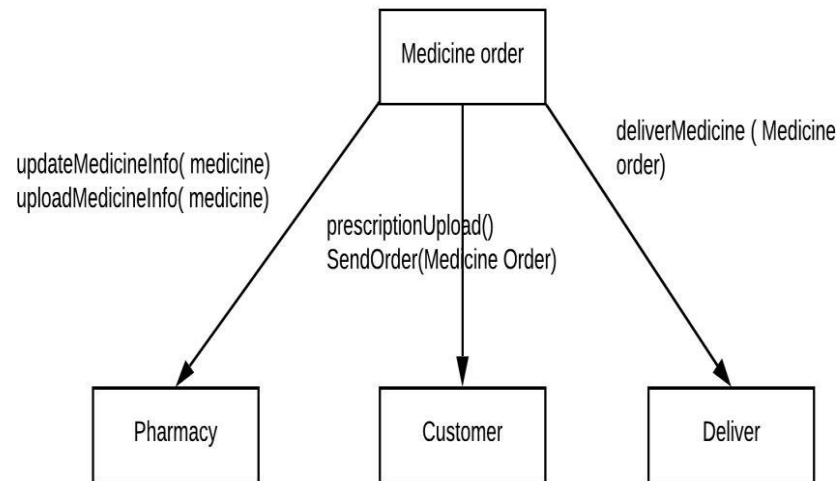


Figure 11 Message Passing (Medicine Order)

Analysis Class (Feedback)

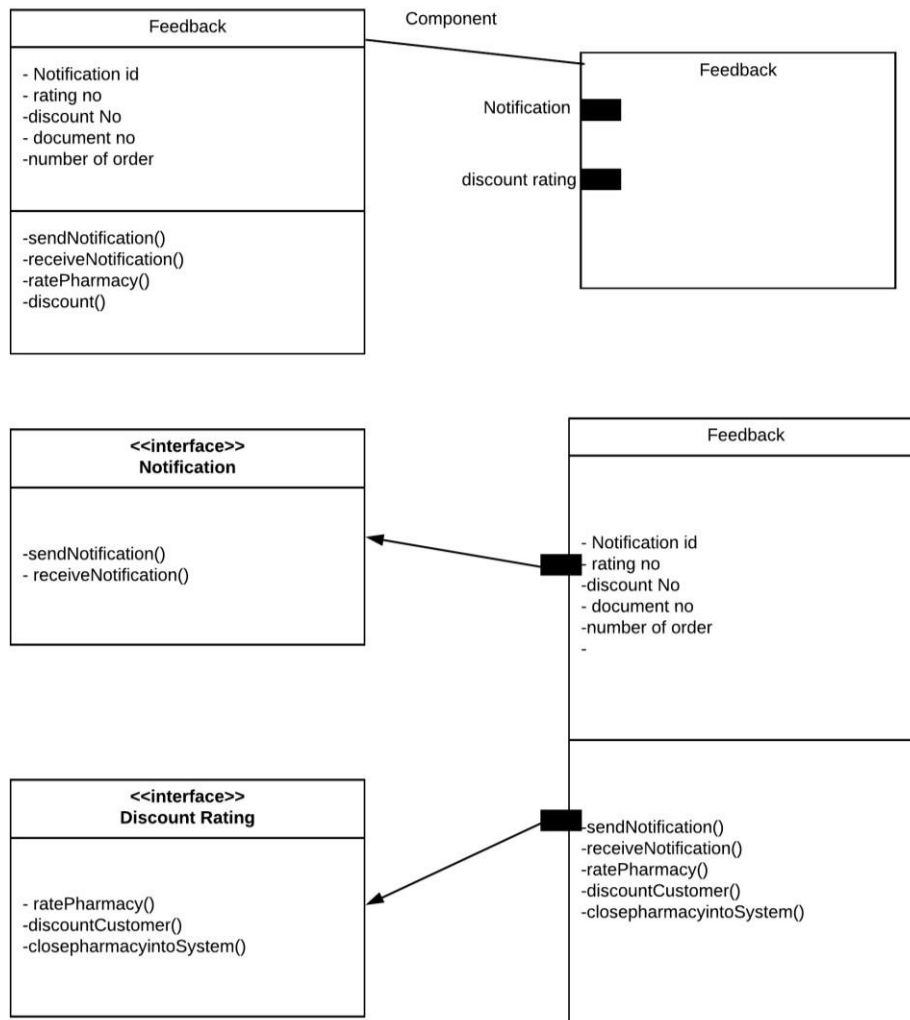


Figure 12 Analysis Class (Feedback)

Message Passing (Feedback)

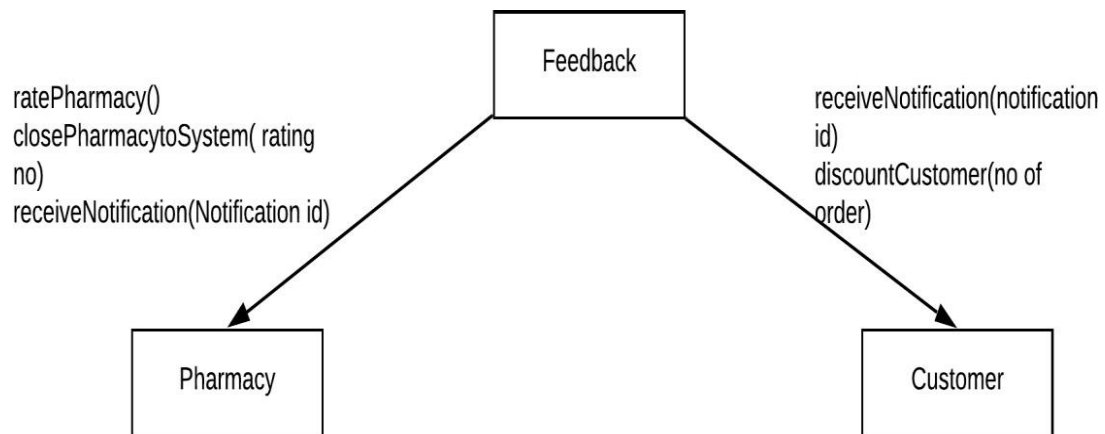


Figure 13 Message Passing (Feedback)

State Chart Fragment for Classes

Statechart diagram is one of the five UML diagrams used to model the dynamic nature of a system. They define different states of an object during its lifetime and these states are changed by events. Statechart diagrams are useful to model the reactive systems.

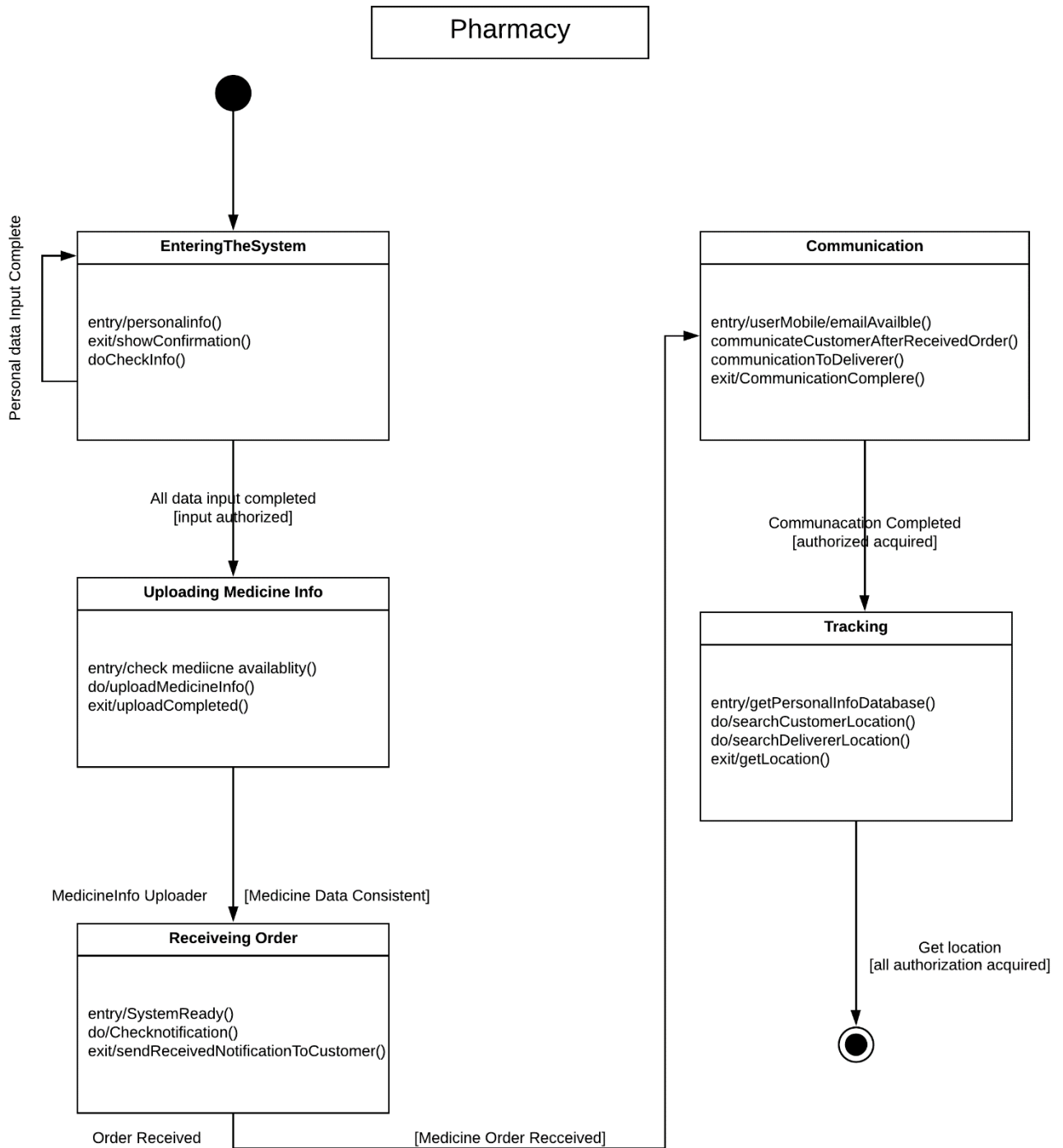


Figure 14 Pharmacy State Chart

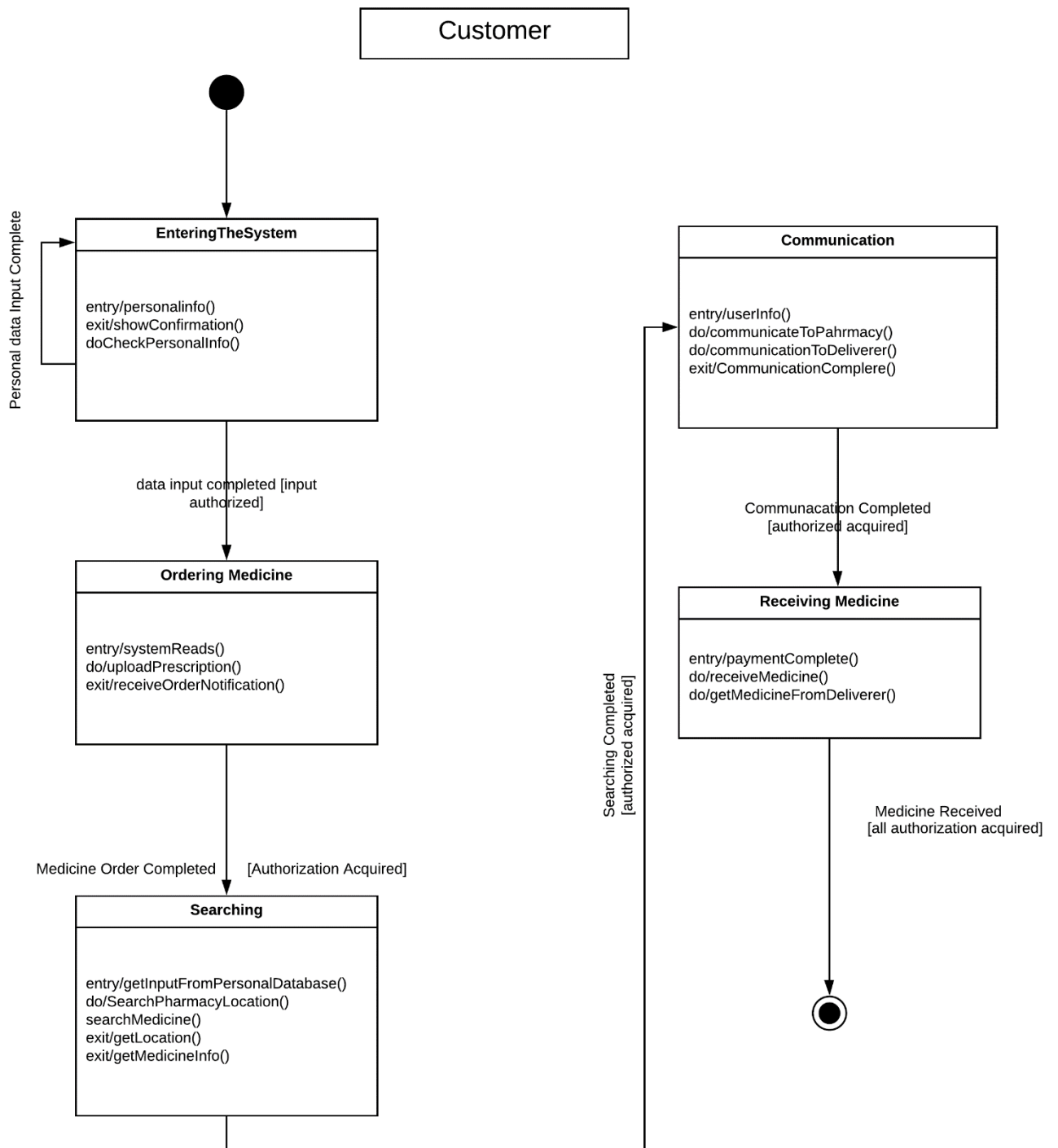


Figure 15 Customer

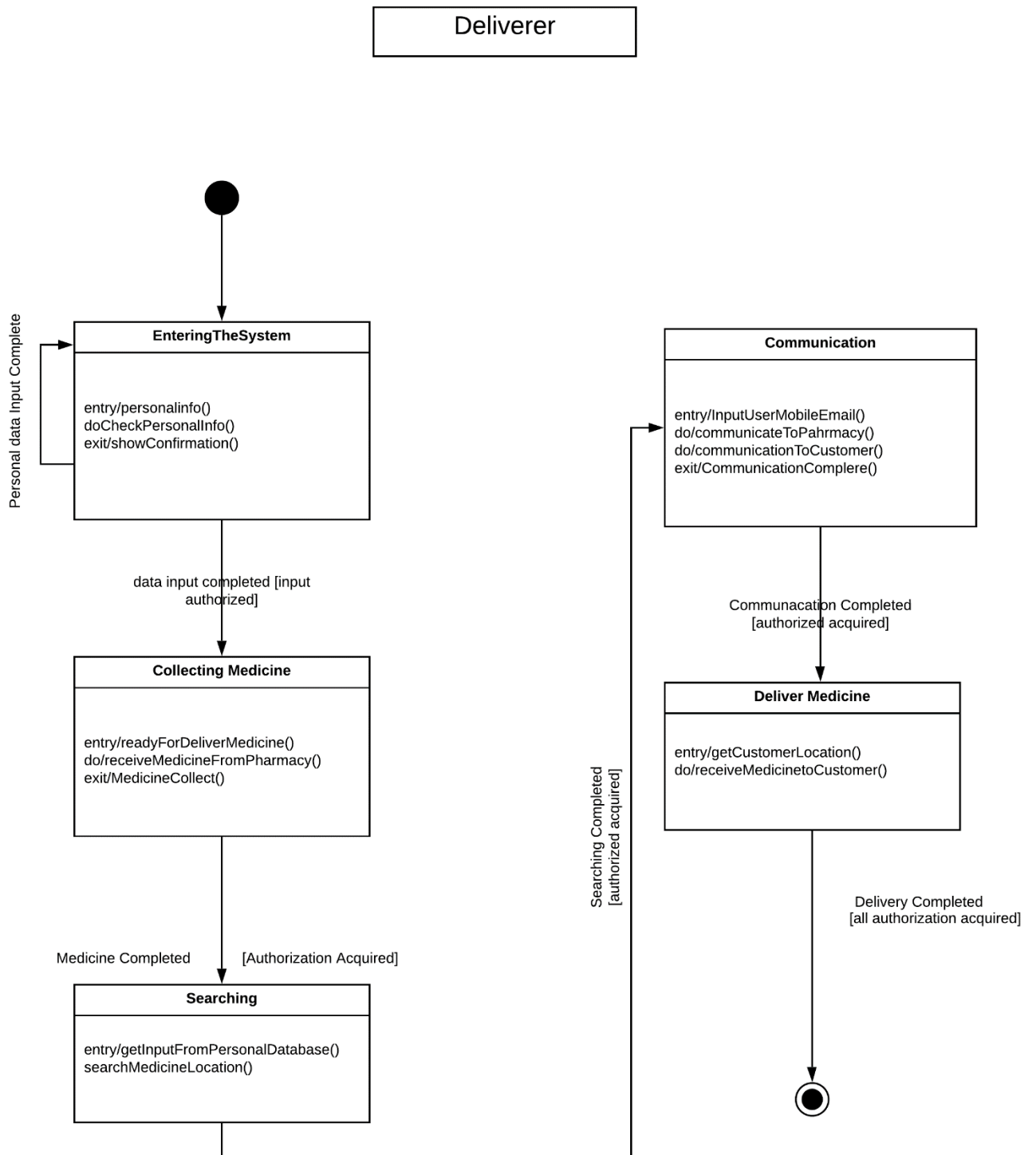


Figure 16 Deliverer

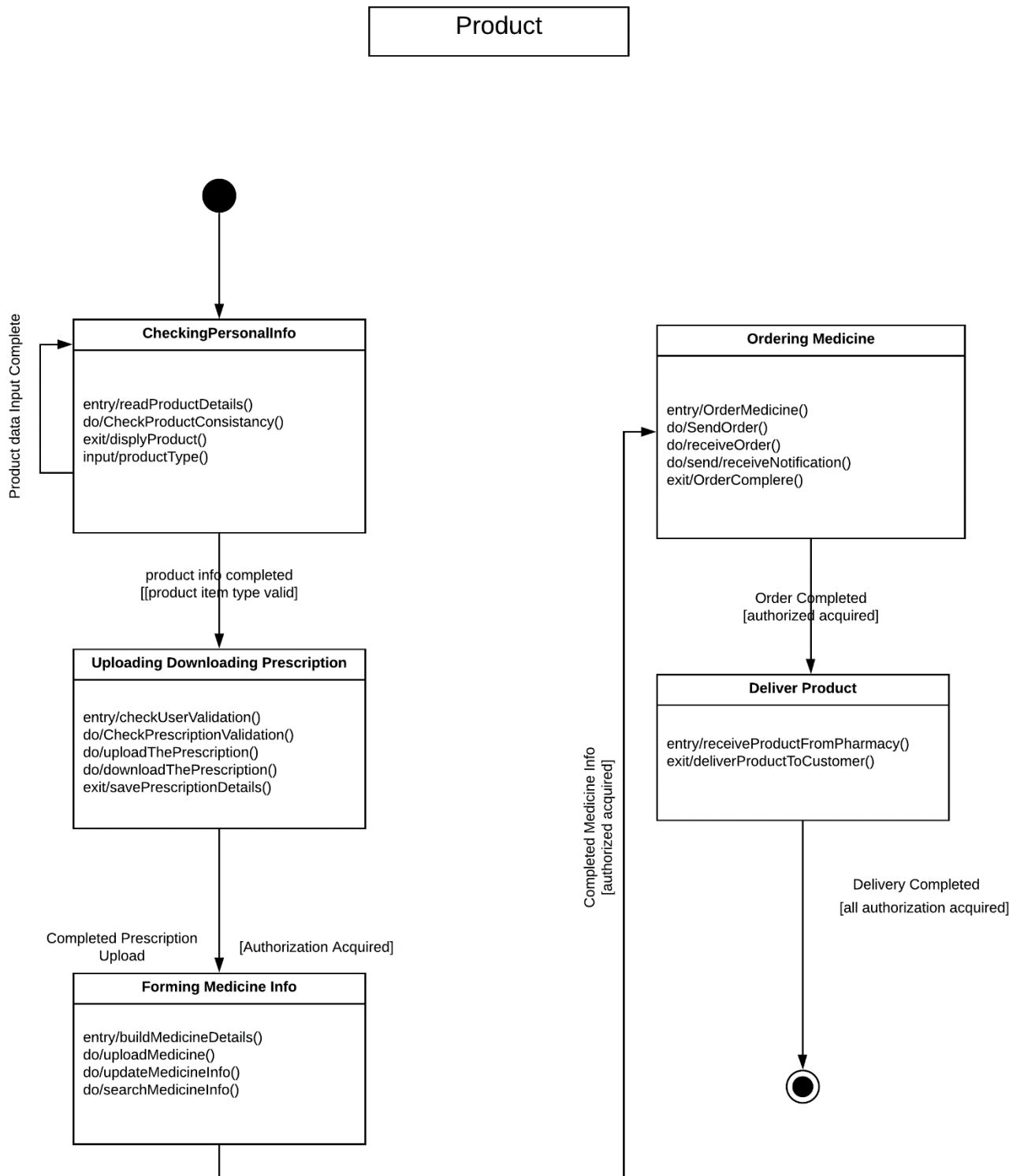


Figure 17 Product

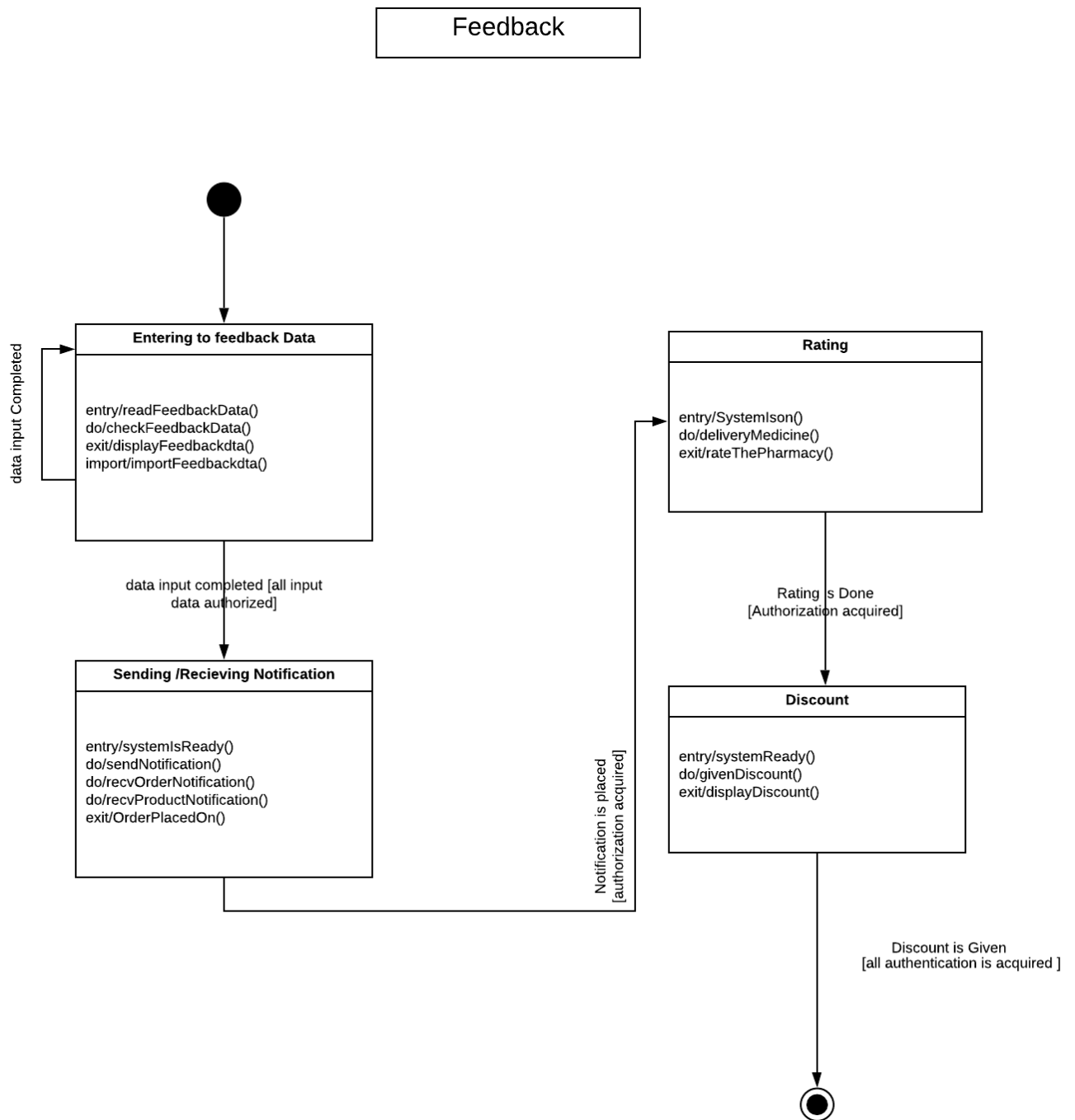


Figure 18 Feedback

Deployment diagram

A deployment diagram in the Unified Modeling Language models the physical deployment of artifacts on nodes. To describe a web site, for example, a deployment diagram would show what hardware components exist, what software components run on each node, and how the different pieces are connected.

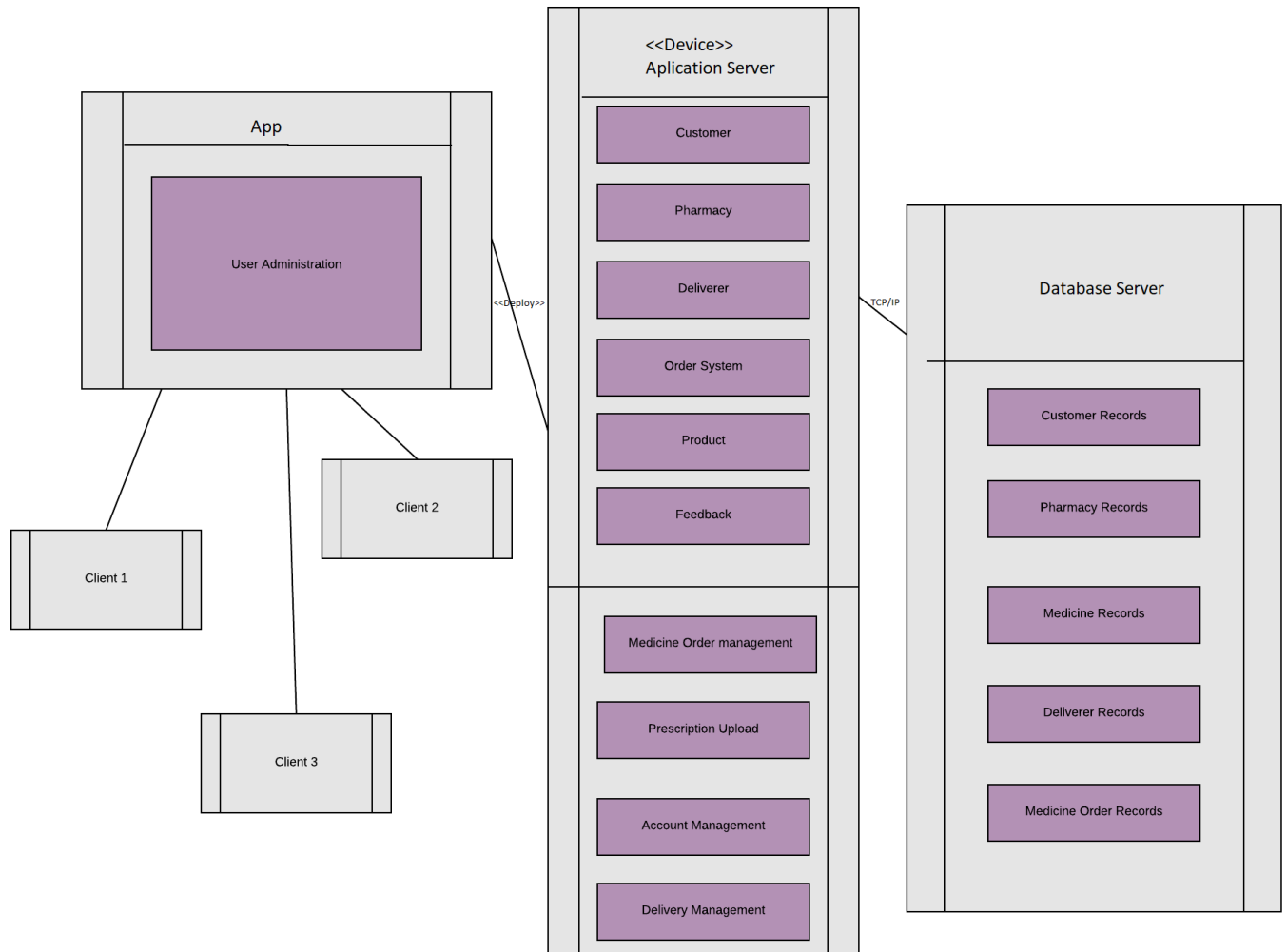


Figure 19 Deployment

Activity diagram

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency.

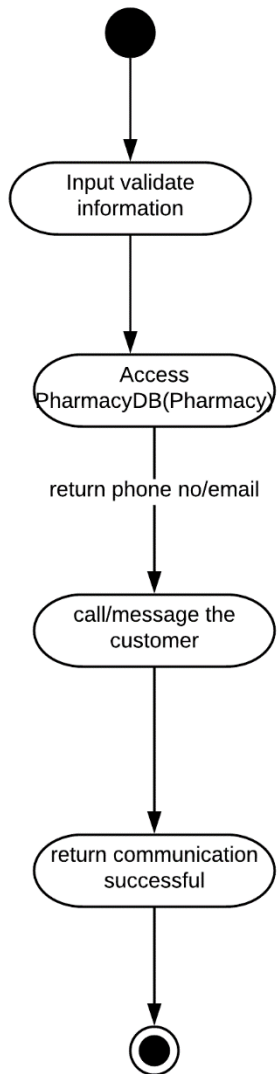


Figure 20 communicateTopharmacy

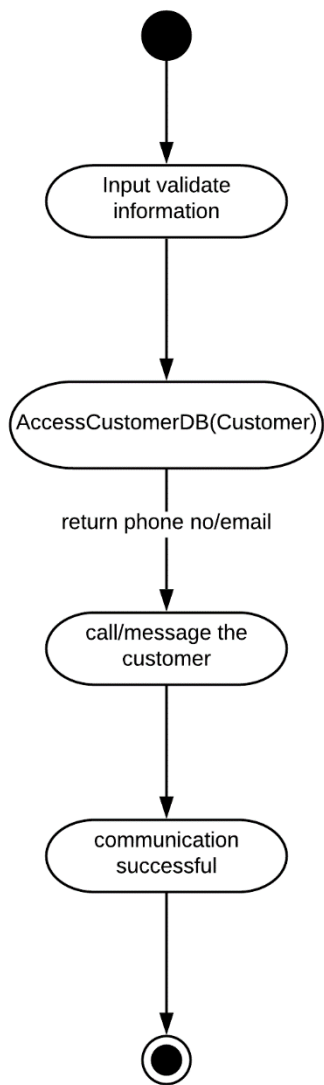


Figure 21 *communicationToCustomer*

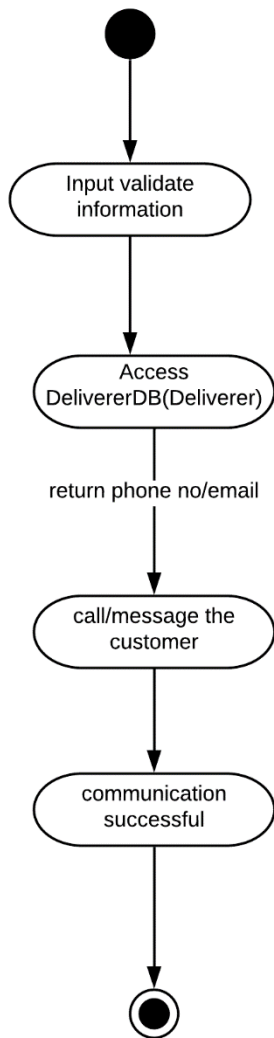


Figure 22 *communicationTodeliverer*

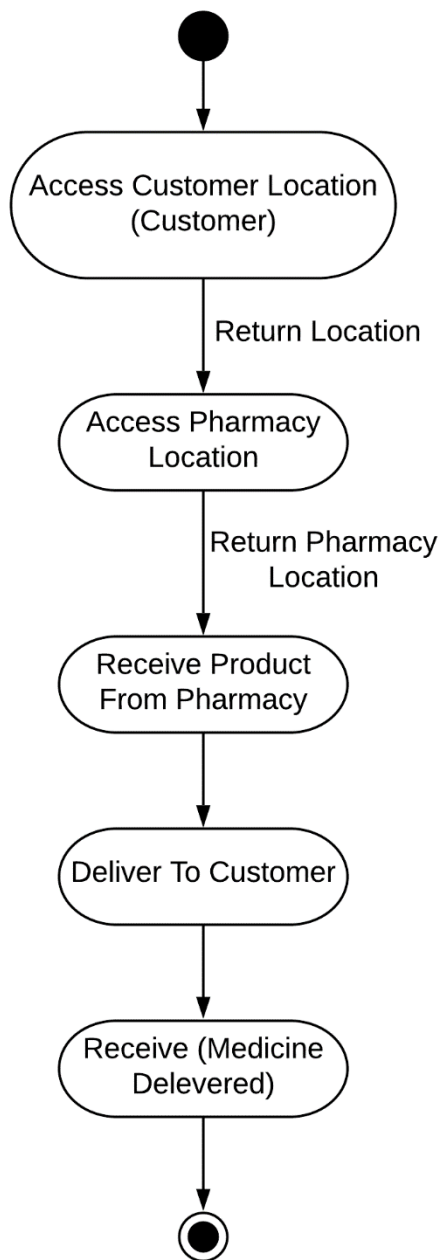


Figure 23 Deliver Medicine

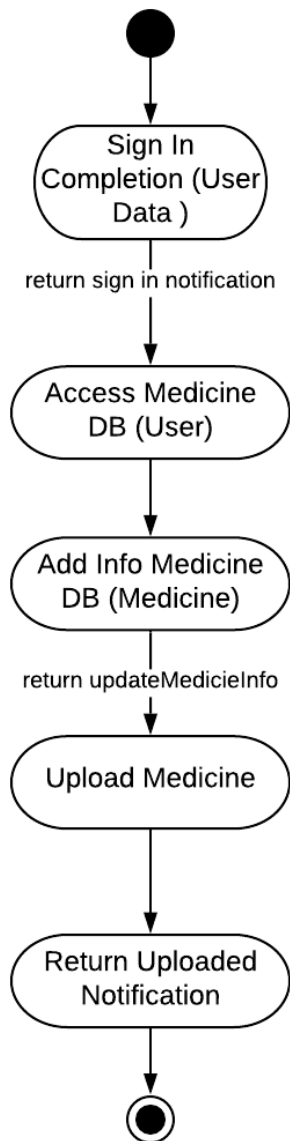


Figure 24 Medicine Info Upload

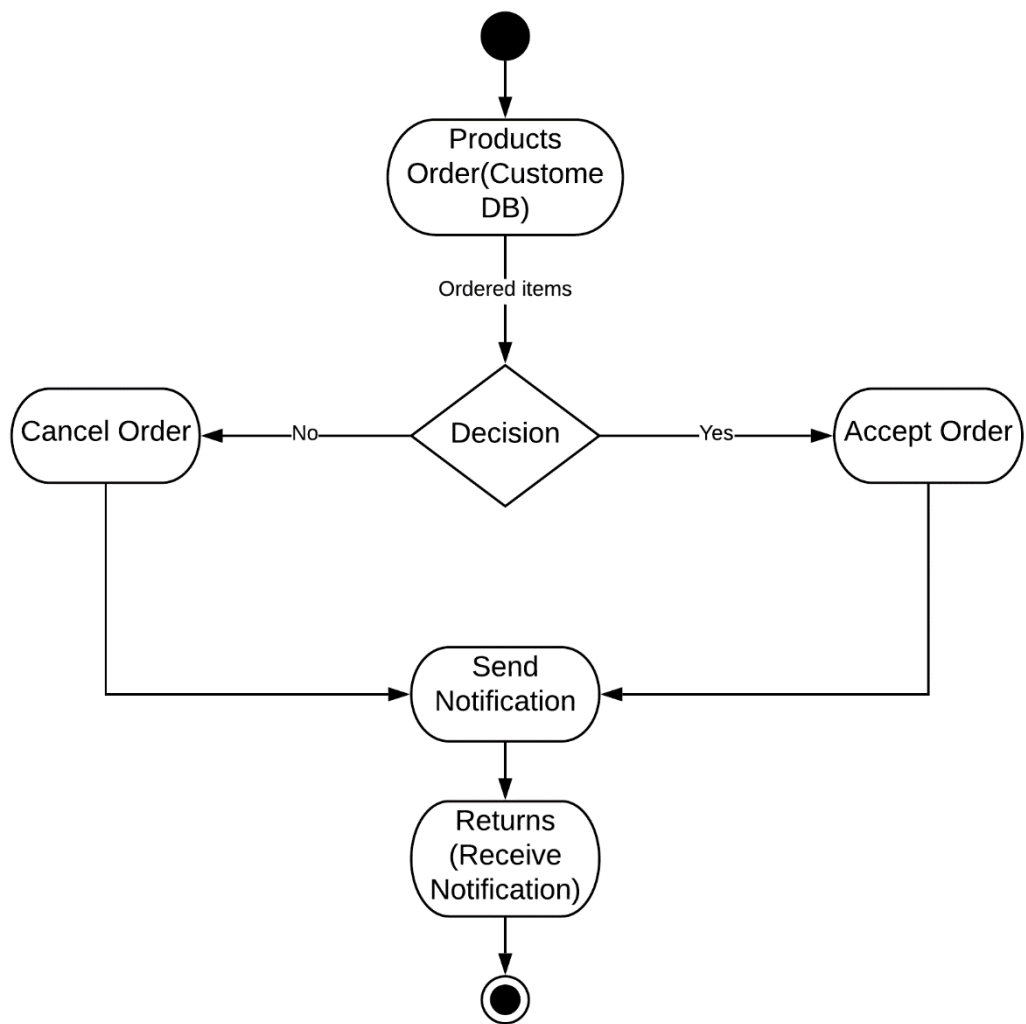


Figure 25 Order Notification

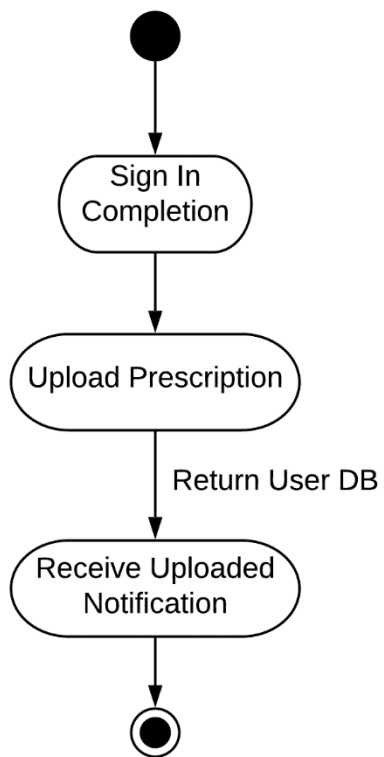


Figure 26 Prescription Completion

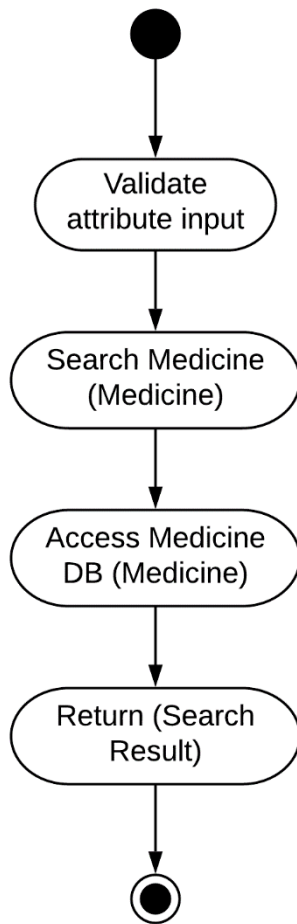


Figure 27 Search Medicine

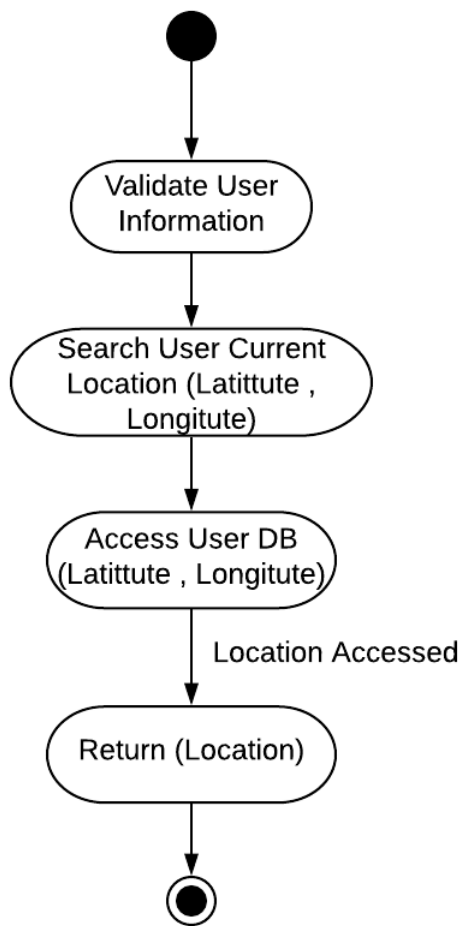


Figure 28 Search User Location

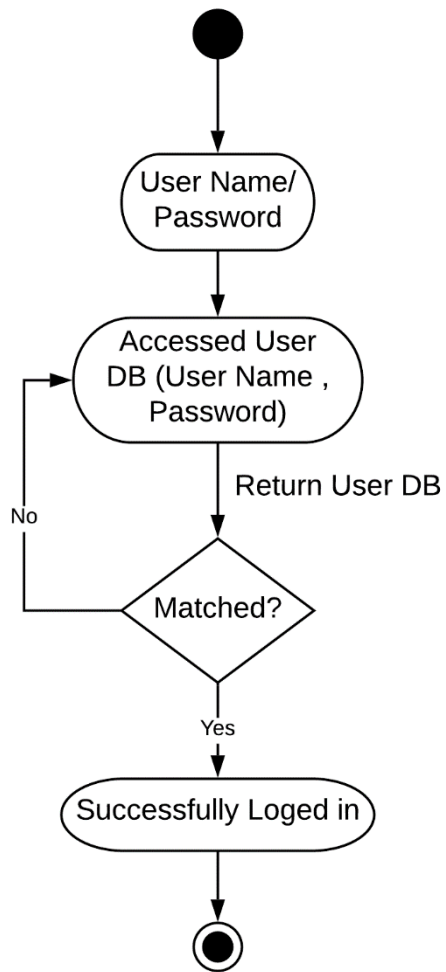


Figure 29 Sign In

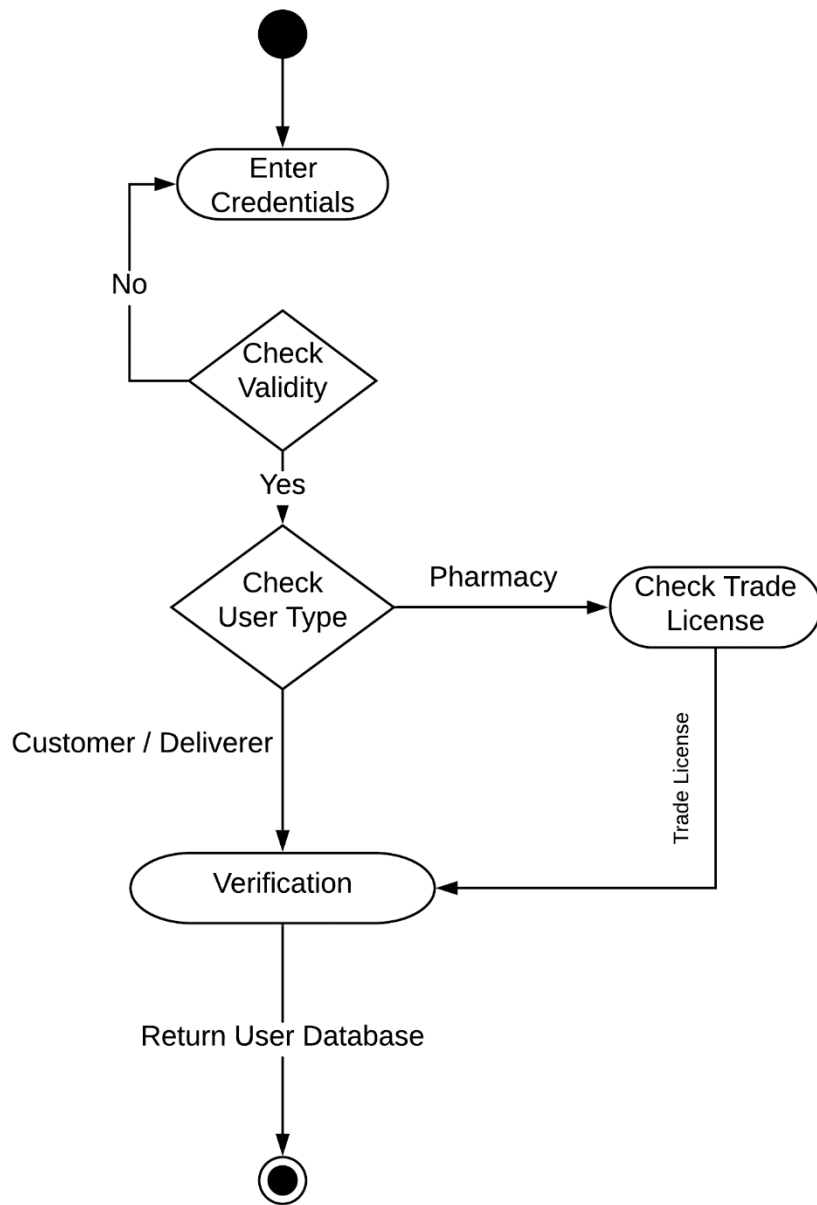


Figure 30 Sign Up

Chapter Three: User Interface Design

User Interface Design is the design of websites, computers, appliances, machines, mobile communication devices, and software applications with the focus on the user's experience and interaction.

The Golden Rules:

Three golden rules for interface design are:

1. Place the user in control.
2. Reduce the user's memory load.
3. Make the interface consistent.

These golden rules actually form the basis for a set of user interface design principles that guide this important aspect of software design.

Interface Analysis

We divide interface analysis into the following part:

1. User analysis
2. Task analysis

User Analysis

There are two steps in this part-

1. Identify user
2. Know user

Identify User

From the requirement and specification document, we have found three types of user-

1. Pharmacy Owner
2. Customer
3. Admin
4. Deliverer

Know User

User Type	Characteristics	Value
Pharmacy Owner	Age	25-40
	Skills	Average
	Domain expert	Yes
	Application expert	No
	Frequency of use	Frequently
	Consequence of a mistake	High
	General computer experience	No
Customer	Age	20-50
	Skills	Average
	Domain expert	Yes
	Application expert	No
	Frequency of use	Occasionally
	Consequence of a mistake	Medium
	General computer experience	Yes
Admin	Age	30-40
	Skills	Above Average
	Domain expert	Yes
	Application expert	Yes
	Frequency of use	Frequently
	Consequence of a mistake	No
	General computer experience	Yes
Deliverer	Age	20-30
	Skills	Below Average
	Domain expert	Yes
	Application expert	No
	Frequency of use	Frequently
	Consequence of a mistake	High
	General computer experience:	No

Table 1 Know Users

Task Analysis

Admin

Tasks	Process Goal	Preconditions	Subtasks
Verify User	Checking validity for Account creation of Pharmacy Owner, Customer and deliverer	Logged in as admin	<ul style="list-style-type: none">• Accept request• Decline request• Ban user
Verify Pharmacy	Checking legal documents of Vehicle	Logged in as admin	<ul style="list-style-type: none">• Add rating• Accept Pharmacy• Reject Pharmacy
Promotion	Increase Sales	Logged in as admin	<ul style="list-style-type: none">• Rating Promotion• Taking Feedbacks

Table 2 Task Analysis Admin

Customer

Tasks	Process Goal	Precondition	Subtasks
Account Creation	Creating use account	Customer must be having NID	<ul style="list-style-type: none">• Registration• Sign In• Sign Up
Update Profile	Change any field of profile information	Logged in as customer	<ul style="list-style-type: none">• Change field data• Update profile picture
Order	Ordering medicines	Logged in as customer	<ul style="list-style-type: none">• Upload Prescription• Search Pharmacy• Check medicine• Check Pharmacy Rating
Payment	Pay the bill	Logged in as customer	<ul style="list-style-type: none">• Cash on Deliver• Apply Promo Code

Table 3 Task Analysis Customer

Pharmacy Owner

Tasks	Process Goal	Precondition	Subtasks
Register Pharmacy	Register pharmacy to the services of MediPro	Pharmacy must have legal license	<ul style="list-style-type: none"> • Registration • Sign In • Sign Up
Track Deliverer	Track deliverer so that the pharmacy can always keep track of their medicine	Logged in as Pharmacy Owner	<ul style="list-style-type: none"> • Track Order • Communicate with Deliverer • Search Deliverer Location • Send Notification
Receive Order	Receive order from the customer	Logged in as Pharmacy Owner	<ul style="list-style-type: none"> • Receive Order • Download Prescription • Communicate with customer • Send notification
Check Medicine	Check the availability of the medicine and its expiration	Logged in as Pharmacy Owner	<ul style="list-style-type: none"> • Check medicine availability • Search Medicine • Check Expire date
Restock Medicine	Restock medicines that have expired or out of stock	Logged in as Pharmacy Owner	<ul style="list-style-type: none"> • Check medicine availability • Check Expire date • Restock medicine • Throw away all expired medicine • Send order notice to supplier
Update Medicine Log	Update medicine availability log in MediPro	Logged in as Pharmacy Owner	<ul style="list-style-type: none"> • Check medicine availability • Check Expire date • Update medicine log • Send notification

Table 4 Task Analysis Pharmacy Owner

Deliverer

Tasks	Process Goal	Precondition	Subtasks
Register	Register delivery van to the services of MediPro	Must have driving license	<ul style="list-style-type: none">• Registration• Sign In• Sign Up
Collect Medicine	Collect medicine from Pharmacy	Logged in as deliverer	<ul style="list-style-type: none">• Communicate with pharmacy• Search pharmacy location• Collect medicine from Pharmacy• Send notification
Deliver Medicine	Deliver medicine to customer	Logged in as deliverer	<ul style="list-style-type: none">• Communicate with customer• Search customer location• Deliver medicine to customer• Send notification
Payment	Receive medicine bill payment from customer if payment type is "Pay by Cash"	Logged in as deliverer	<ul style="list-style-type: none">• Receive cash• Send notification• Receive customer delivery receipt• Deliver cash to pharmacy

Table 5 Task Analysis Deliverer

Interface Design Steps of MediPro

The MediPro logo is displayed in a bold, sans-serif font. The word "Medi" is in a dark blue color, and "Pro" is in a reddish-orange color. The logo is centered within a blue decorative shape at the top of the page.

MediPro

Choose Who You are

Customer

Pharmacy Owner

Figure 31 Welcome Page

MediPro

Welcome to MediPro

Email

Password

Sign In

Create New Account

Figure 32 Sign In Page

MediPro

Create a new account ...

User Name

Email

Password

Confirm Password



I agree with the term & conditions

Sign UP

Figure 33 Sign Up Page

MediPro

Customer



Profile



Prescription



Search



Track

Figure 34 Customer



Pharmacy Owner



Profile



Medicine



Search



Track

Figure 35 Pharmacy Owner

MediPro

Medicine List

ATTACH FILE



Figure 36 Medicine

MediPro

Profile



Name

Age

Address

Contact No

Figure 37 Profile

MediPro

PRESCRIPTION

ATTACH FILE



Figure 38 Upload Prescription

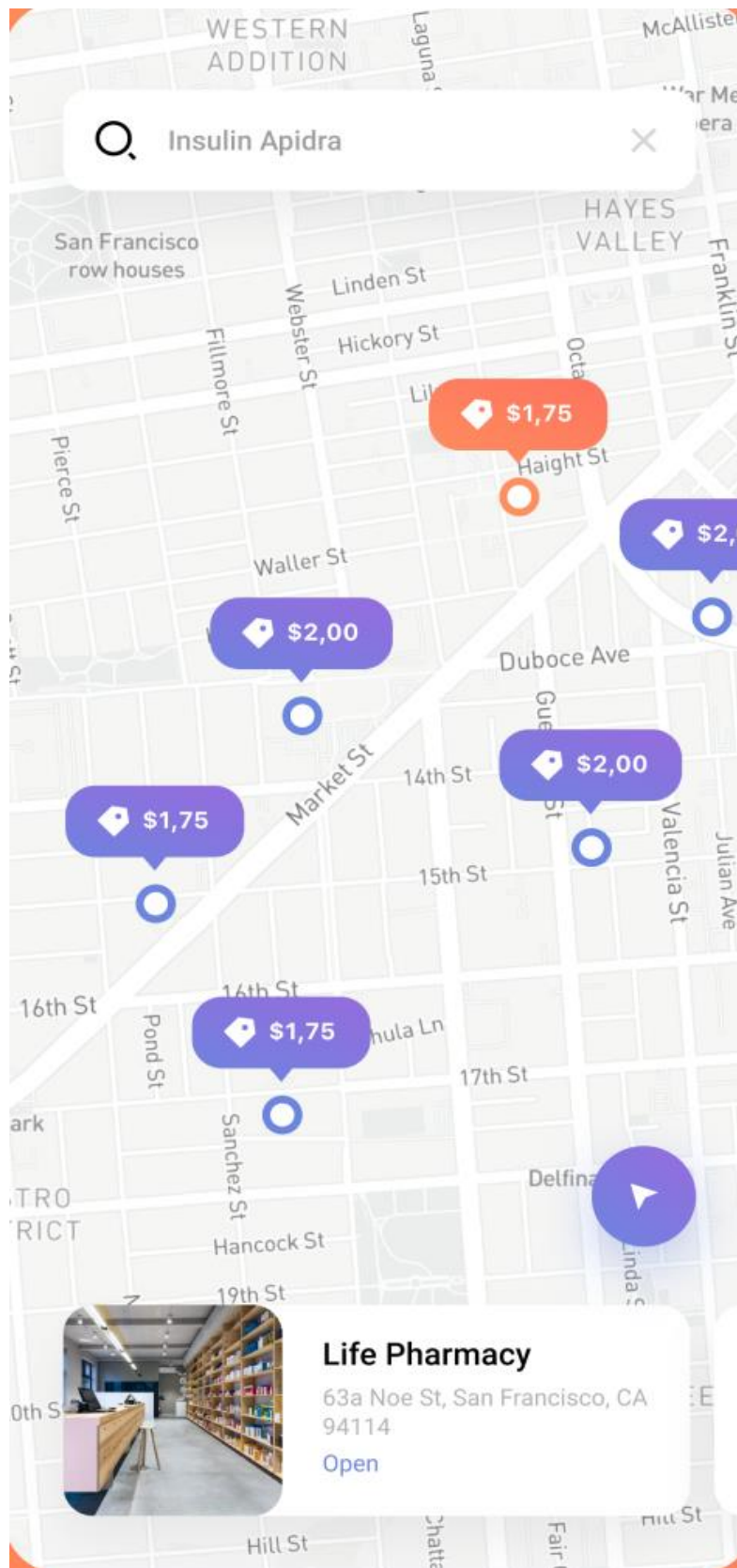


Figure 39 Location



San Francisco ▼

Search for drugs

🔍 Search

Categories

Anesthetic

Medical
Equipment

Ant

Best Pharmacies



Aster Pharmacy



Life Pharmacy



Figure 40 Search Drug