**Department of Computer Science & Engineering**

**Veer Surendra Sai University of Technology**

**INVENTORY MANAGEMENT**

**Software Engineering Lab**

**Submitted By:**

**Sudipta Kumar Nayak: 1802041033**

**Aitik Dandapat: 1802041034**

**Udit Kumar Panda: 1802041035**

**Date: 18/10/2020**

**Inventory Management System**

**Inventory:** We can refer inventory to all the items, goods, merchandise, andmaterials held by a business organization for selling in the market to earn some profit. We can also say that, Inventory is:

1. An asset, tangible or intangible,
2. An asset that can be realized for revenue generation or has a value for exchange, or
3. An asset which is in process but is meant for sale in the market.

For example, a medical store buys bulk of different medicine from different pharmaceutical company and earns profit by selling into the market.

**Different types of User:**

|  |  |
| --- | --- |
| **Types Of Users** | **User Description** |
| **Inventory Manager** | Stock clerks unload products from delivery trucks and arrange them in a supermarkets’ storeroom.  They transfer items from the store room to store shelves.  They also periodically take stock a stores’ inventory. |
| **Customer/ Retail Shopkeeper** | They order the goods from the warehouse.  Also have the facility to reorder the same good, or cancel the order as per their needs. |
| **Security Dept.** | They maintain the security of the organization.  They maintain the security of the org.  They constantly monitor the supermarket from every corner of the org.  Report of any suspicious activities which are taking place inside of the supermarket. |
| **Cashier** | Ringing up branches for customers.  When customers pay with cash, cashiers must make correct changes. |
| **Delivery Manager** | They responsible for the delivery of the products ordered by the customers.  Keep an order registry of the same. |
| **Customer Helpers** | Responsible for giving advice and product details to the customer inside a supermarket. |

**User Features:**

* **Reorder point:** Should inventory reach a specific threshold, a company's ***inventory management system can be programmed to tell managers to reorder that product***. This helps companies avoid running out of products or tying up too much capital in inventory.
* **First Time Order:** A company's ***inventory management system can be programmed to tell managers to order a particular product***.
* **Cancel Order:** A company's ***inventory management system can be programmed to tell managers to cancel a particular product***.
* **Asset Tracking:** When a product is in a warehouse or store, it can betracked via its barcode and/or other ***tracking criteria***, such as serial ***number, lot number or revision number***. Nowadays, inventory management software often utilizes barcode, radio-frequency identification (RFID), and/or wireless tracking technology.
* **Service Management:** Companies that are primarily ***service-oriented rather than product-oriented*** can use inventory management software to track the cost of the materials they use to provide services, such as cleaning supplies. This way, they can attach prices to their services that reflect the ***total cost*** of performing them.
* **Product Identification:** Barcodes are often the means whereby ***data on products and orders*** are inputted into inventory management software. A barcode reader is used to ***read barcodes and look up information on the products*** they represent. Radio-frequency identification (RFID) tags and wireless methods of product identification are also growing in popularity.
* ***Inventory Optimization*:** A fully automated demand forecasting andinventory optimization system to attain key inventory optimization metrics such as ***Reorder Print, Order Quantity, Lead Demand, Stock Cover, Accuracy.***
* **Cost Saving:** A company's inventory represents one of its largestinvestments, along with its workforce and locations. Inventory management software helps companies cut ***expenses by minimizing the amount of unnecessary parts and products in storage***. It also helps companies keep **l*ost sales to a minimum by having enough stock on hand to meet demand***.
* ***Increased Efficiency*:** Inventory management software often allows forautomation of many inventory-related tasks. For example, software can **automatically collect data, conduct calculations, and create records**. This not only results in time savings, cost savings, but also increases business efficiency.
* **Warehouse Organization:** Inventory management software can helpdistributors, wholesalers, manufacturers and retailers ***optimize their warehouses***. Eg: If certain products are often sold together or are more popular than others, those products can be grouped together or placed near the delivery area to speed up the process of picking.

**Feasibility Report of Inventory management Software**

Feasibility study in software engineering is the study of the software to evaluate feasibility of the proposed project or system.

Types of Feasibility study:

1. **Technical Feasibility –** In Technical feasibility current resources both hardware software along with required technology are analyzed / assessed to develop project.
2. **Operational Feasibility -** In Operational Feasibility degree of providing service to requirements is analyzed along with how much easy product will be to operate and maintenance after deployment.

|  |  |  |  |
| --- | --- | --- | --- |
| Feature | Technical Feasible | Operational Feasible | Remarks |
| Registration Portal | Yes | Yes |  |
| Reorder Point | Yes | Yes |  |
| Order Point | Yes | Yes |  |
| Cancel | Yes | yes |  |
| Asset Tracking | Yes | Yes |  |
| Service Management | Yes | Yes |  |
| Product Identification | No | Yes | Currently we don’t have particular hardware resource to perform this task |
| Inventory Optimization | Yes | Yes |  |
| Cost Saving | Yes | Yes |  |
| Increased Efficiency | Yes | Yes |  |
| Warehouse Organization | Yes | Yes |  |

**SDLC Model for Inventory Management Software**

The **Iterative Waterfall model** will be suitable life cycle model for inventory management software. The reasons behind choosing waterfall model are,

1. **Iterative** **Waterfall model** is usually chosen for low risk and small scale projects; thus this model has been identified as the fitting option for inventory management system.

2. It is easy to manage and understand because of its simplicity.

3. Since Waterfall model is easy to manage and is very simple to understand and use, thus the employee of any organization will be able to collaborate with the developer.

4. The requirements for implementing inventory management system are certain and less complex.

5. Ample of resources are available for developing inventory management software.

6. Also it conserves the resources to cut down the cost of development and maintenance.

7. Iterative waterfall model feedback path from one phase to its preceding phase allows correcting the errors that are committed and these changes are reflected in the later phases.

**Effort and development time calculation using the COCOMO model:**

This project will use an **basic**​ **COCOMO model** with​**​ organic model** to calculate effort and development time.

The estimated effort and scheduled time for the project are given by relation:

Effort (E) = a\*(KLOC)bPer Month

Scheduled Time (D) = c\*(E)dMonths(M)

Where,

* **E =** Total effort required for the project in Man-Months (MM).
* **D =** Total time required for project development in Months (M).
* **KLOC =** the size of the code for the project in Kilo lines of code.
* **a, b, c, d** **=** The constant parameters for a software project.

Features:

* Registration Portal – 0.5 KLOC
* Reorder Point – 0.3 KLOC
* Order Point – 0.4 KLOC
* Cancel – 0.2 KLOC
* Asset Tracking – 0.4 KLOC
* Service Management – 0.6 KLOC
* Inventory Optimization – 0.6 KLOC
* Cost Saving – 0.4 KLOC
* Increased Efficiency – 0.4 KLOC
* Warehouse Organization – 0.6 KLOC

Expected lines of code ​​≈ ​**5000 = 5kloc**

According to our model:

Effort (E) = a\*(KLOC)​b​ Per Month

Development Time(D) = c\*(Effort) ​d​ Month

Where

a=2.4, b=1.05, c=2.5, d=0.38

KLOC (Number of line of code in kilo) = 5

Effort = 2.4 \* (5)1​.05​ = 13.00 PM

Development time = 2.5 \* (13.00)​0.38​= 6.06 ≈​ 6.626 M ≈ 6.6 M

**Estimation of all the parameters for project duration estimation using PERT chart technique:**

Expected time for each task using **PERT** chart **= (O+4\*L+P)/6**

where, 0 = Optimistic Estimate

L = Likely Estimate

P = Pessimistic Estimate

Design Database

21,27,33

Code Database

23,37,51

Integration and Testing

33,43,53

Finish

0

Code GUI

16,19,22

Design GUI

13,18,23

Specification

9, 16, 23

Write User Manual

33, 38, 43

**Expected time for each task :**

Specification = (9 + 4x16 + 23)/6 = 16

Write user manual = (33 + 4x38 + 43)/6 = 38

Integrate and Testing = (33 + 4x43 + 53)/6 = 43

Design GUI = (13 + 4x18 + 23)/6 = 18

Design database = (21 + 4x27 + 33) /6= 27

Code database = (23 + 4x37 + 51)/6 = 37

Code GUI = (16 + 4x19 + 22)/6 = 19

**[Software Requirements Specification](https://en.wikipedia.org/wiki/Software_requirements_specification)**

**Introduction:**

**Purpose :**

The purpose of this document is to provide details on the existing problem and to present the design of the proposed ‘Inventory Management System’. This will provide solutions to the inventory related issues which the Warehouse Organizers are facing in maintaining, tracking and monitoring their inventory. This document will give a detailed description of the system functionalities, analysis of requirements, constraints in the system, technology, economic feasibility, context diagram and proposed plan to develop the system.

**Overview :**

The project aims at providing an efficient interface to the warehouses for managing their goods inventory based on each item sold. The basic idea involved here is that each item is linked to its atomic goods which are stored in a database. At the end of each day, the system analyzes the total sale of menu items and proportionately deducts appropriate amount from the resource database. Then it compares the current available resources with the threshold level of each goods. If it finds that certain goods are below the threshold, it will generate a purchase order for those item(s) and send it to the manager (admin) for approval.

We also propose to include a special feature “Prediction”. This feature keeps track of any upcoming occasions, climatic changes and special events that may influence inventory needs for the upcoming week. The system will then predict the required resources for these events based on previously accumulated information/knowledge. It will now generate an updated purchase order in accordance with the predictions.

The product also aims to keep track of the shelf life of resources. If any resource nears the end of its shelf life, it would intimate to the manager (admin) the details of the quantity that is near its expiration date. The warehouse must function efficiently, the goods must be tracked correctly, timely orders must be sent out to the vendors, and the inventory must be maintained and updated at all times.

**Environment Characteristics:**

**Hardware:**

* 20 GB HDD Free Space
* 256 MB RAM
* Pentium IV or above Processor
* Monitor

**Peripherals:** Mouse, Keyboard

**People:**

* Inventory Manager
* Security Department
* Customer/ Retail Shopkeeper
* Delivery Manager
* Cashier
* Customer Helpers

**Functional Requirement**

A functional requirement defines a function of a system or its component. The system should aim at providing functions that would help the user in managing and controlling inventory in an efficient manner.

**A> Inventory Manager:**

R.1.1 **Authentication**

Description: This function should enable the administrator, users/ faculty to log into the system. This is a mandatory requirement because every user who wants to access the inventory system should have a user id and password. The system should validate the login details of the user entering them

Input: User id: Employee id

Password: DOB\_Firstname

Output: Correct credentials or not

R.1.2 **Report Generation:**

Description: Monthly, Quarterly or Yearly reports need to be generated for auditing purpose.

Input: Enters the details of product like items left, item sold, rise or fall in the sell of the item compared to last report.

Output: Can be able to print the report in csv or pdf format.

R.1.3 **Data Integration:**

Description: To maintain a common database according to the inventory, inventory manager has access to write, update or delete the item from the database according to sale.

R.1.3.1 **Add data**

Input: Details of item (Item id, price, quantity, date of manufacture, etc.)

Output: Item added

R.1.3.2 **Update data**

Input: Item Id of the item with updated values if that item

Output: Details updated

R.1.3.3 **Delete data**

Input: Details of item (Item id)

Output: Data is updated by deleting the item.

R.1.3.4 **View data**

Input: Enter start date and end date of data manufacturing.

Output: View (Shows the current view of the database)

**B> Security Department:**

R.2.1: **Authentication**

Description: This function should enable the administrator, users/ faculty to log into the system. This is a mandatory requirement because every user who wants to access the inventory system should have a user id and password. The system should validate the login details of the user entering them

Input: User id: Employee id

Password: DOB\_Firstname

Output: Correct credentials or not

R.2.2 **Data Integration:**

Description: The system will provide a tabular view as well as a graphical view of the availability of all the equipment in the inventory.

R.2.2.1 **View data**

Input: Enter start date and end date of data manufacturing.

Output: View (Shows the current view of the database)

R.2.3 **Report Alert:**

Description: This system creates alert to the whole management as soon as the security department experiences unusual behavior in the inventory or missing item

Input: Enter the alert report

Output: It alerts the whole management team

**C> Cashier:**

R.3.1: **Authentication**

Description: This function should enable the administrator, users/ faculty to log into the system. This is a mandatory requirement because every user who wants to access the inventory system should have a user id and password. The system should validate the login details of the user entering them

Input: User id: Employee id

Password: DOB\_Firstname

Output: Correct credentials or not

R.3.2 **Data Integration:**

Description: The system will provide a tabular view as well as a graphical view of the availability of all the equipment in the inventory.

R.3.3.1 **Update data**

Input: Item Id of the item with updated values (payment details) if that item

Output: Payment details updated

R.3.2.2 **View data**

Input: Enter start date and end date of data manufacturing.

Output: View (Shows the current view of the database)

**D> Delivery Manager:**

R.4.1: **Authentication**

Description: This function should enable the administrator, users/ faculty to log into the system. This is a mandatory requirement because every user who wants to access the inventory system should have a user id and password. The system should validate the login details of the user entering them

Input: User id: Employee id

Password: DOB\_Firstname

Output: Correct credentials or not

R.4.2 **Data Integration:**

Description: The system will provide a tabular view as well as a graphical view of the availability of all the equipment in the inventory and access to update the data as soon its delivered.

R.4.3.1 **Update data**

Input: Item Id of the item with updated values (delivery status) if that item

Output: Status updated

R.4.2.2 **View data**

Input: Enter start date and end date of data manufacturing.

Output: View (Shows the current view of the database)

**E> Customer Helpers:**

R.5.1: Authentication

Description: This function should enable the administrator, users/ faculty to log into the system. This is a mandatory requirement because every user who wants to access the inventory system should have a user id and password. The system should validate the login details of the user entering them

Input: User id: Employee id

Password: DOB\_Firstname

Output: Correct credentials or not

R.5.2 **Data Integration:**

Description: The system will provide a tabular view as well as a graphical view of the availability of all the equipment in the inventory and access to update the database if any issue faced by the customer.

R.5.2.1 **Update data**

Input: Item Id of the item with updated values (queries) if that item

Output: Queries added

R.5.2.2 **View data**

Input: Enter start date and end date of data manufacturing.

Output: View (Shows the current view of the database)

R.5.3 **Email Notification:**

Description: Once the equipment is borrowed both the borrower and lender should have a record of the transaction.

Input: Send an email to customer regarding their issue or payment related queries/remainder.

Output: Triggers the mail and drops it in respective inboxes.

**F> Customer / Retail Shopkeeper**

R.6.1: Authentication

Description: This function should enable the administrator, users/ faculty to log into the system. This is a mandatory requirement because every user who wants to access the inventory system should have a user id and password. The system should validate the login details of the user entering them

Input: User id:Email id

Password: 8 character long(comprising of 1 capital letter, 1 small letter, 1 numeric character and 1 special character)

Output: Correct credentials or not

R.6.2 **Ordering Portal**

Description: It helps for ordering, reordering and canceling the order according to customer wish.

R.6.2.1 **Order**

Input: List of items

Output: Places order and changes the database simultaneously.

R.6.2.2 **Reorder**

Input: Select the previously ordered items

Output: Reorders the same thing, which u have ordered earlier.

R.6.2.3 **Cancel Order**

Input: Select the ordered items for deletion

Output: Cancels the order, and deletes the data from the database

R.6.3 **Help Center**

Description: Helps to contact the help center of warehouse for any issue

Input: Send email to the customer care.

Output: Drops an email to the customer care of the warehouse

**4. Nonfunctional Requirements**

Non-functional requirements define how well the system needs to function.

**Hardware Interfaces :**

The hardware interface for the user would be any PC having any OS and a minimum 10 GB HDD. The main interface would be monitor, keyboard and mouse.

**User Interfaces:**

1. The system should be ***user-friendly*** for the administrator and users. Buttons should clearly define the function which it will perform.

2. The system should provide a ***help option*** which will help first-time users to understand the system. The Help option should include all the functions which are included on the pages.

3. During transaction, the table should be locked in the database to ***avoid any deadlocks or inconsistency*** in the system.

4. When the system takes long time to retrieve any page then it should ***prompt a message*** ‘Processing. Please wait for few more seconds'.

**Security:**

5. The system should have protection from unauthorized users**. *Password of the User should be encrypted***. Eg: \*\*\*\*\*.

6. Pages for data manipulations in the system should be invisible for the users with managerial authorization roles.

**Software Interfaces**

Inventory Management will use database for storing and management of records. So an access to the database management system is required. When such an event occurs the system establishes connection to the database management system, once the connection is created; the client program can communicate with the database management system.

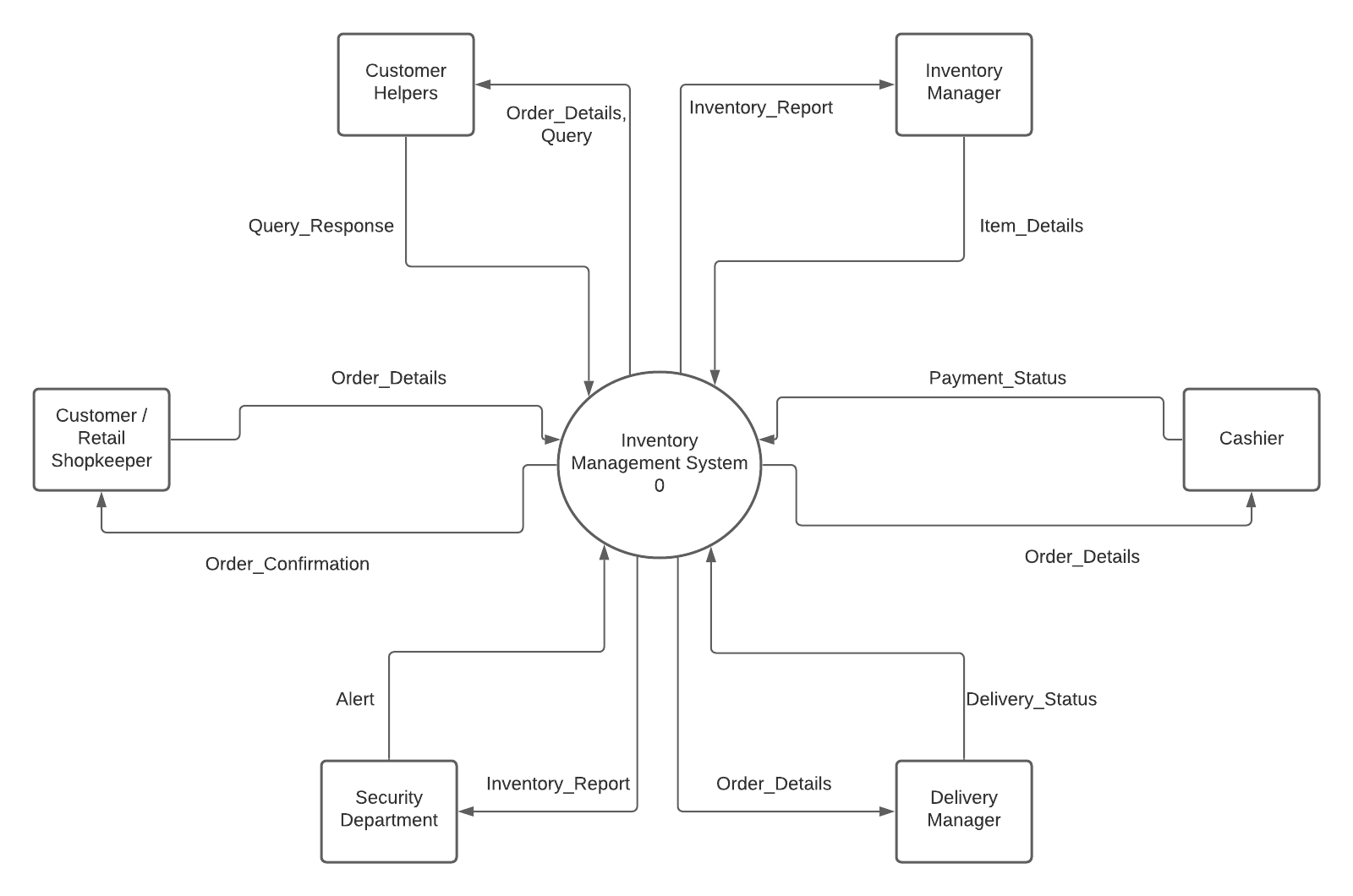
5. Behavioral Descriptions

a. System States

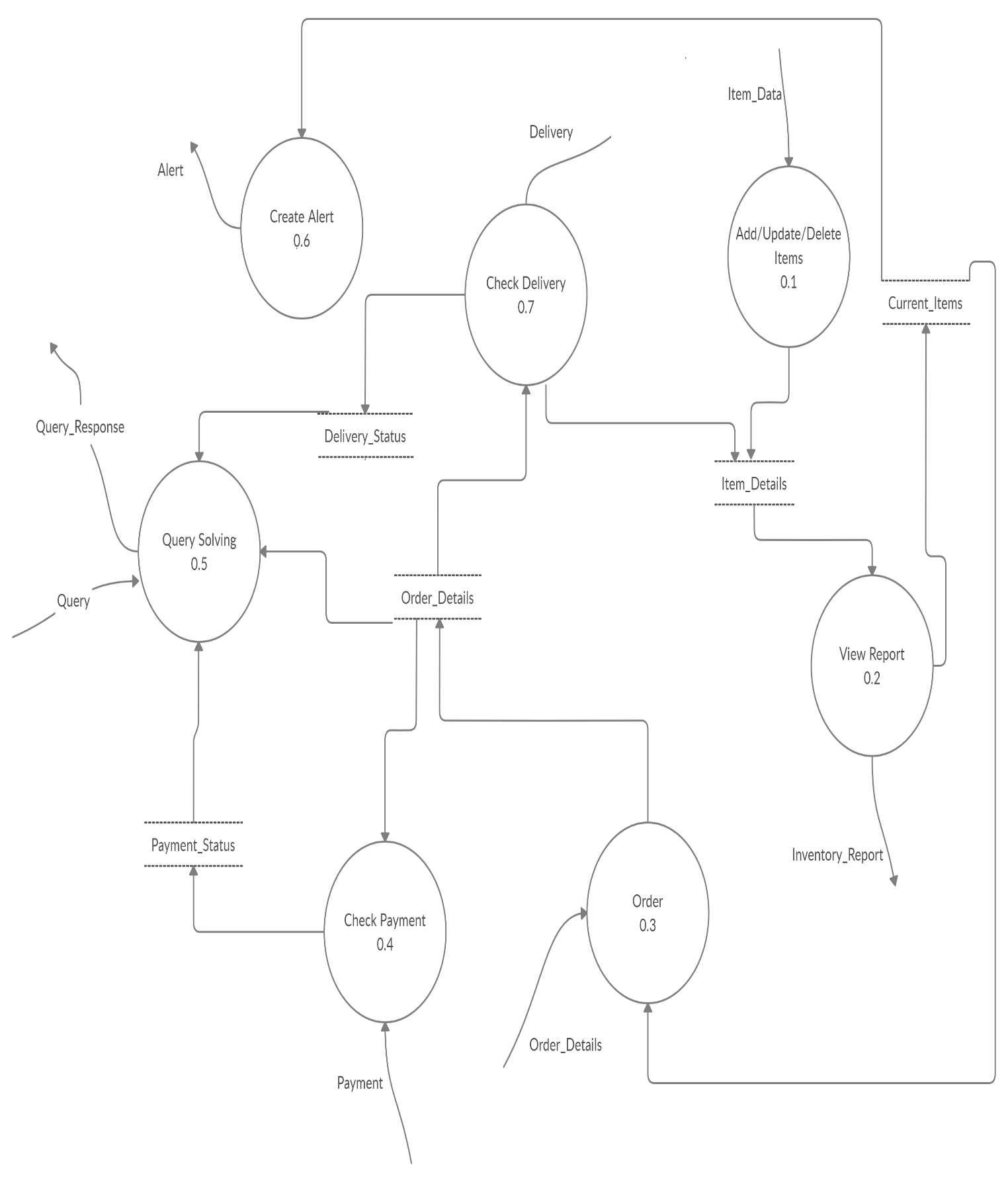
b. Events and Actions

**Data Flow Diagrams**

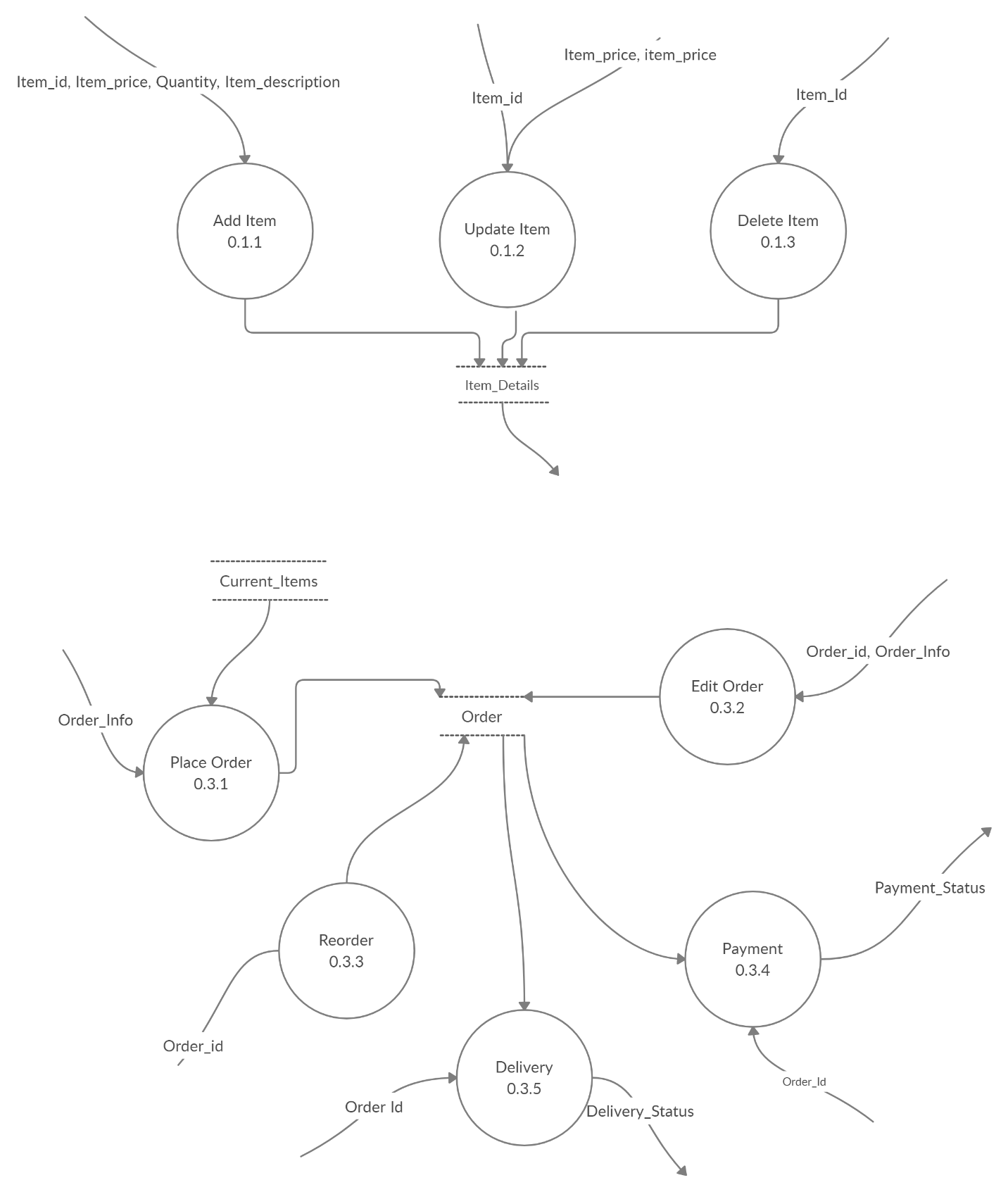
1. **Level 0 DFD**



1. **Level 1 DFD**



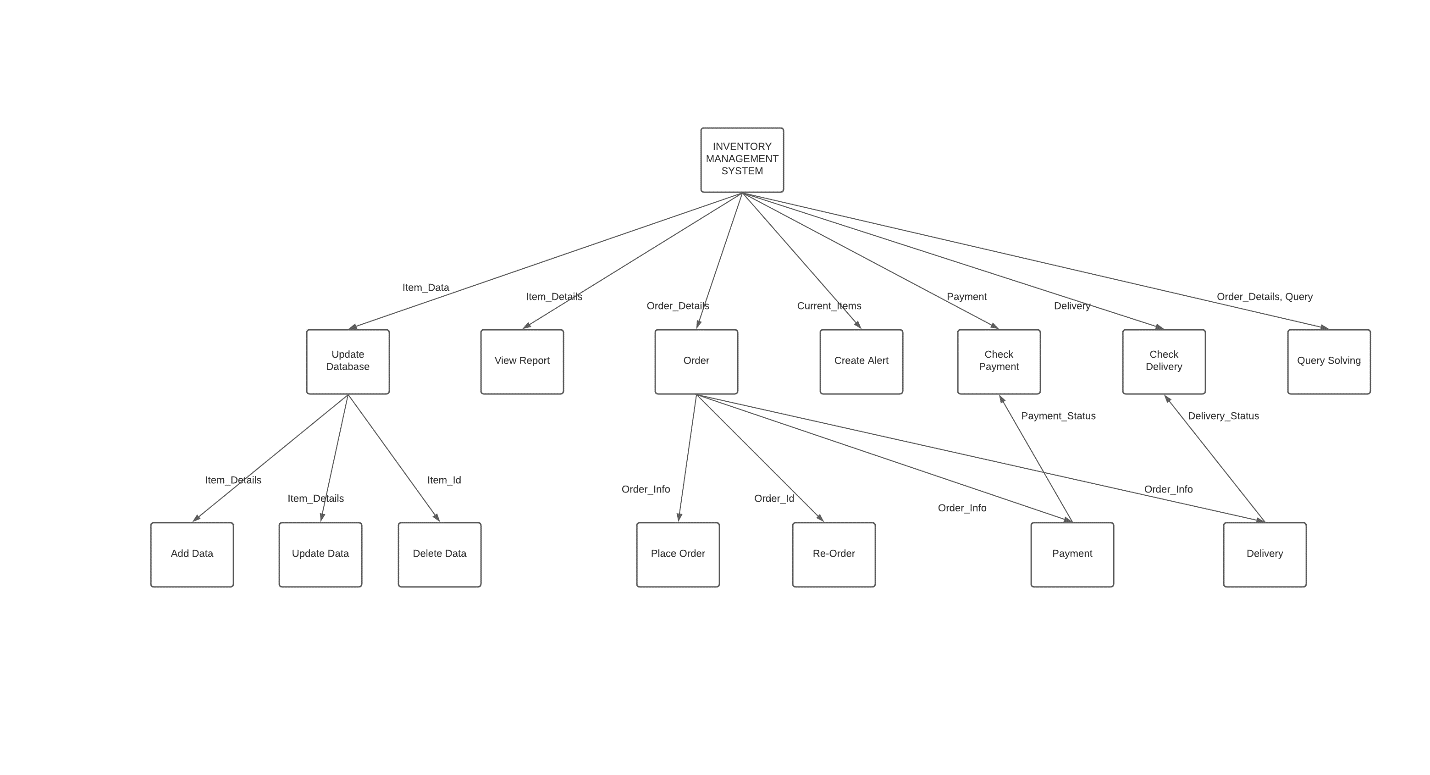
1. **Level 2 DFD**

****

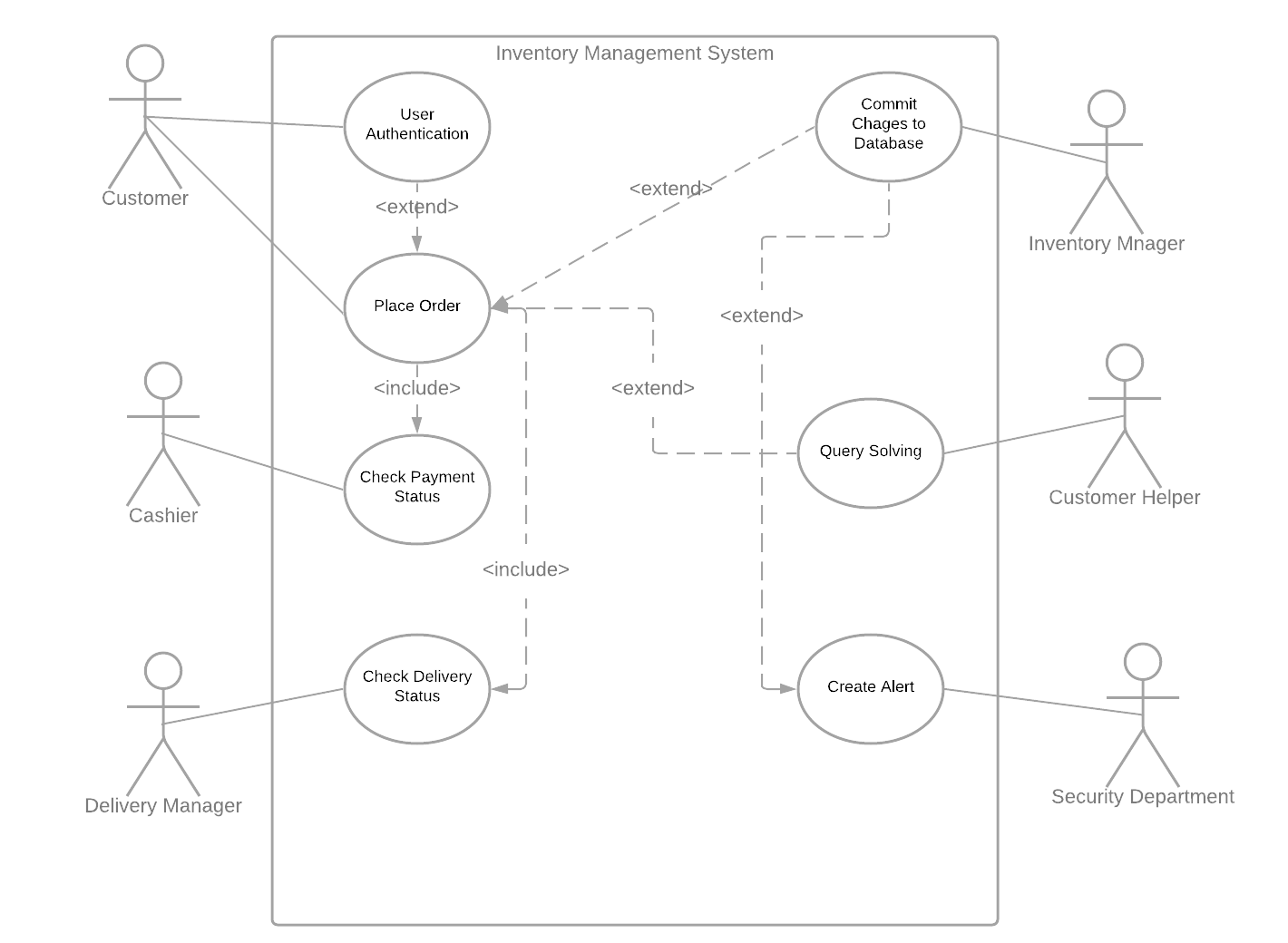
**Data Dictionary**

* Item\_Details: Item\_id + Item\_Price + Item\_Description + Item\_Quantity
* Item\_id: String
* Item\_Price: Integer
* Item\_Description: String
* Current\_Items: {Item\_Details}\*
* Order\_Details: Customer\_Name + Date + Time + Customer\_Id + Payment Details + Invoice
* Customer\_Name: String
* Date: Year + Month + Day
* Year: Integer
* Month: Integer
* Day: Integer
* Time: Hour + Minute
* Hour: Integer (0-23)
* Minutes: Integer (0-59)
* Order\_Id: integer
* Order\_Info: String
* Delivery\_Status: String
* Payment\_Status: String
* Payment\_Details: [Card, Cash, Online]Card: Card\_Number + Customer\_Name + CVV
* Cash: Amount
* Online: UPI-id
* Amount: FloatInvoice: {Order number + {Item\_id + Item\_Price + Quantity}\* + Customer\_Id}
* Query: String
* Query\_Response: String
* Alert: String

**Structure Chart**

****

**Use Case Diagram**

****

**Use Case Description:**

1. **User Authentication**

Actors : Customer

Preconditions : Customer should have an unique e-mail

Post conditions: Unique Customer ID will be generated

1. **Place Order**

Actors : Customer

Preconditions :

* Customer should have an account
* Items should be available

Post conditions : Unique Order ID will be generated

1. **Updation of delivery status**

Actors : Delivery Manager

Preconditions : Deliver person can track the order

Post conditions : Delivery status will be updated

1. **Updation of payment status**

Actors : Cashier

Preconditions : Accepts the payment and generates bill

Post conditions : Payement status will be updated

1. **Query Solving**

Actors : Customer Helper

Preconditions : Have access to the order placed.

Post conditions : Query Resolved or not.

1. **Create Alert**

Actors : Security Department

Preconditions : Have access to the items present in inventory

Post conditions : If mismatched found with database, creates alert

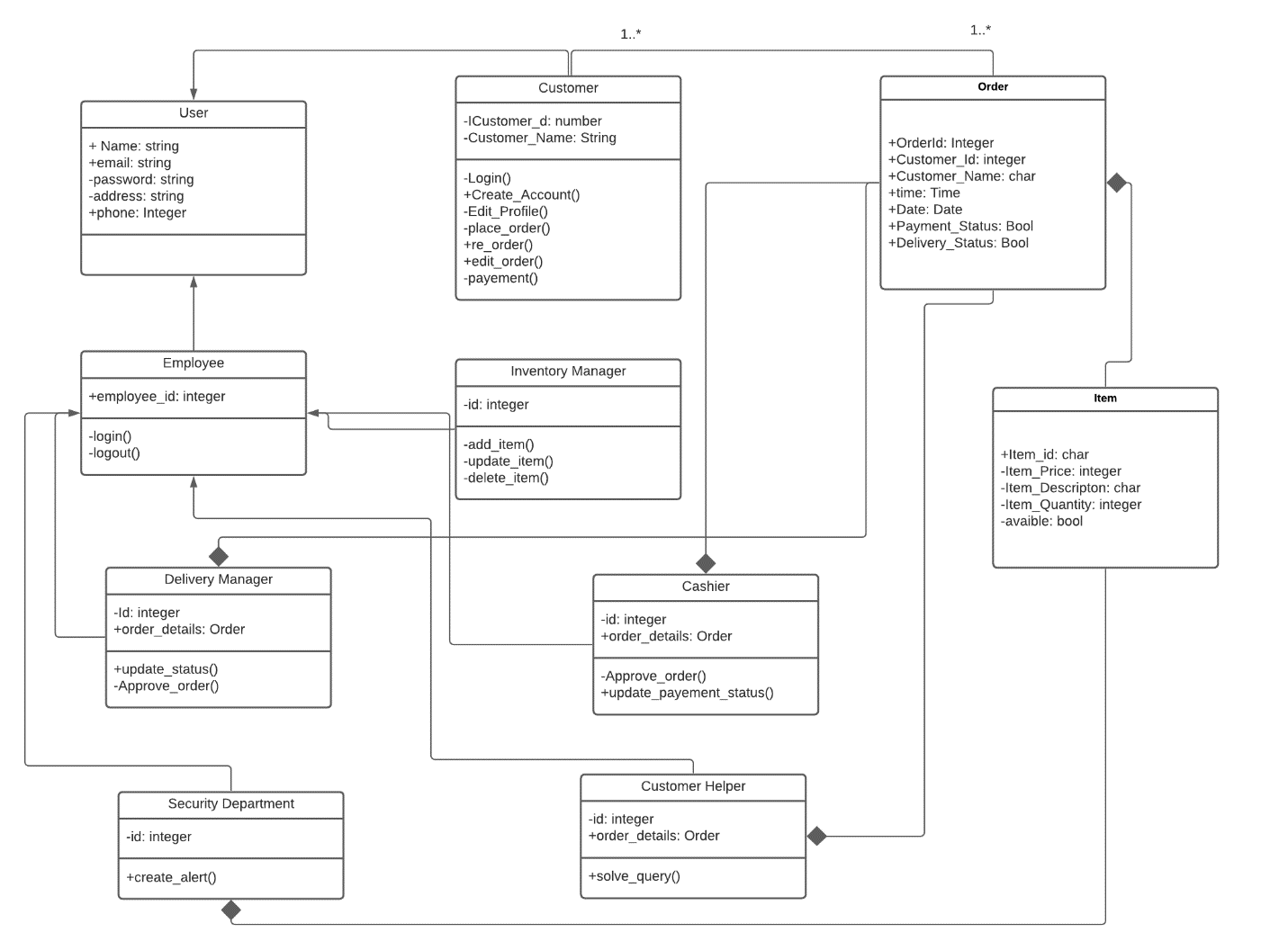
1. **Database update**

Actors : Inventory Manager

Preconditions : Fetches raw item

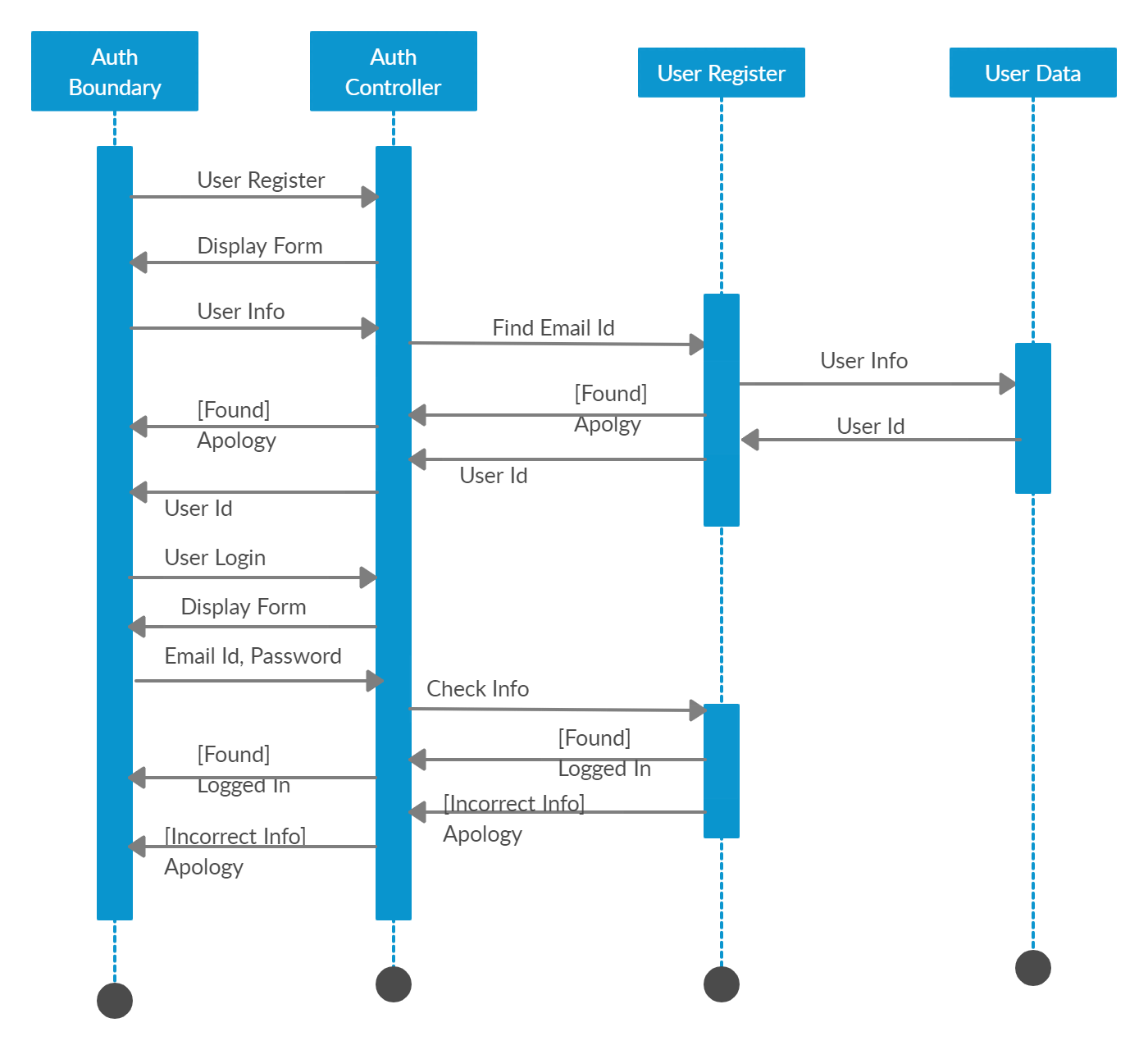
Post conditions : Updates the database (Add/ update/ delete item)

**Class Diagram:**

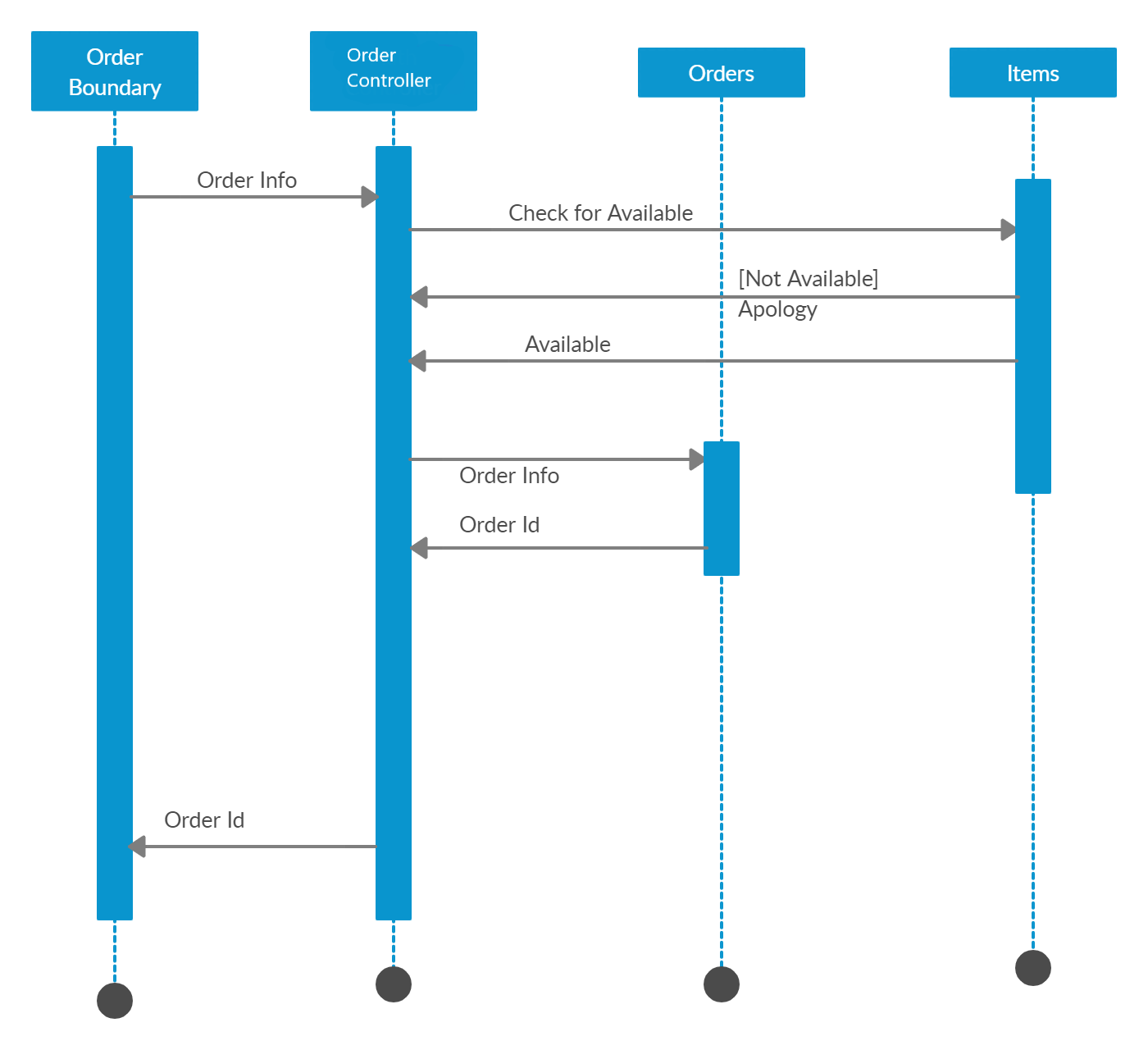
****

**Sequence Diagram of each Use Case:**

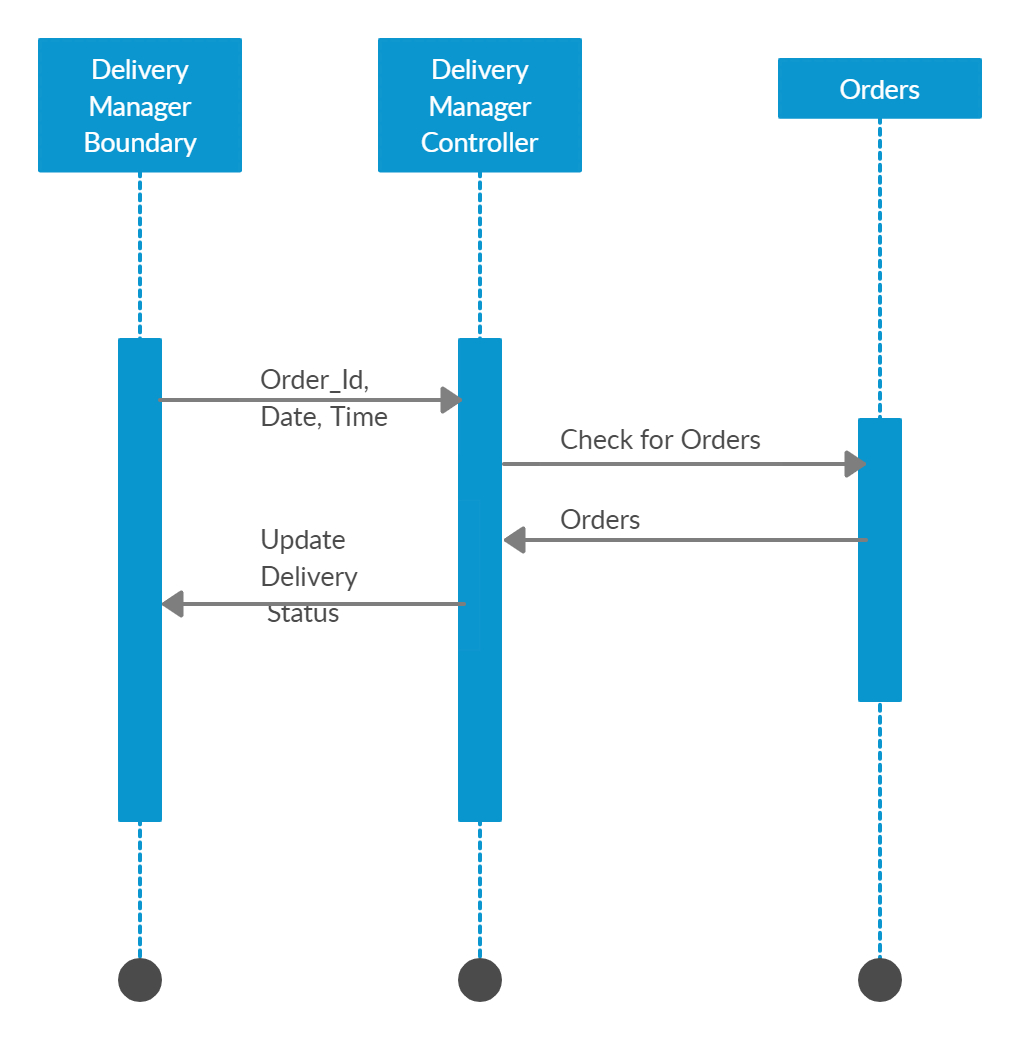
1. **User Authentication**

****

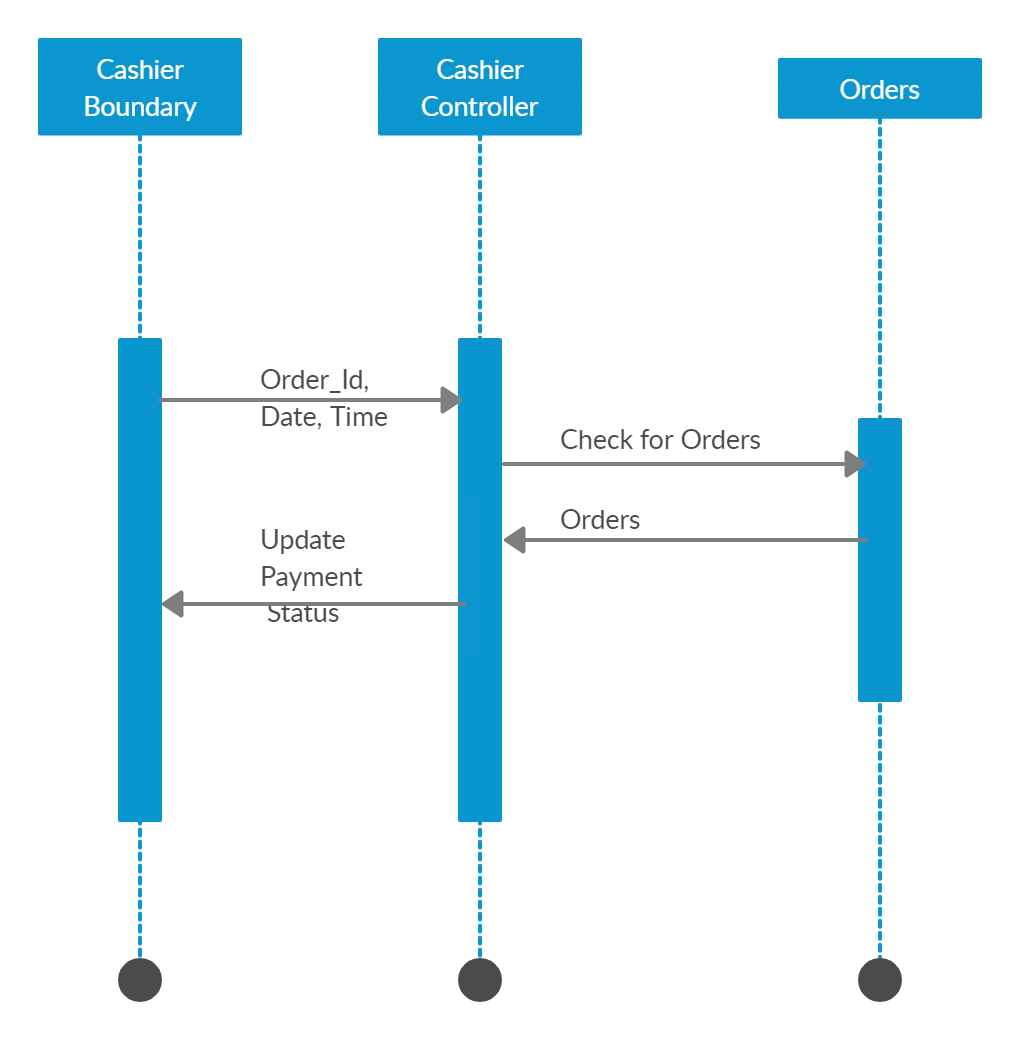
1. **Place Order**

****

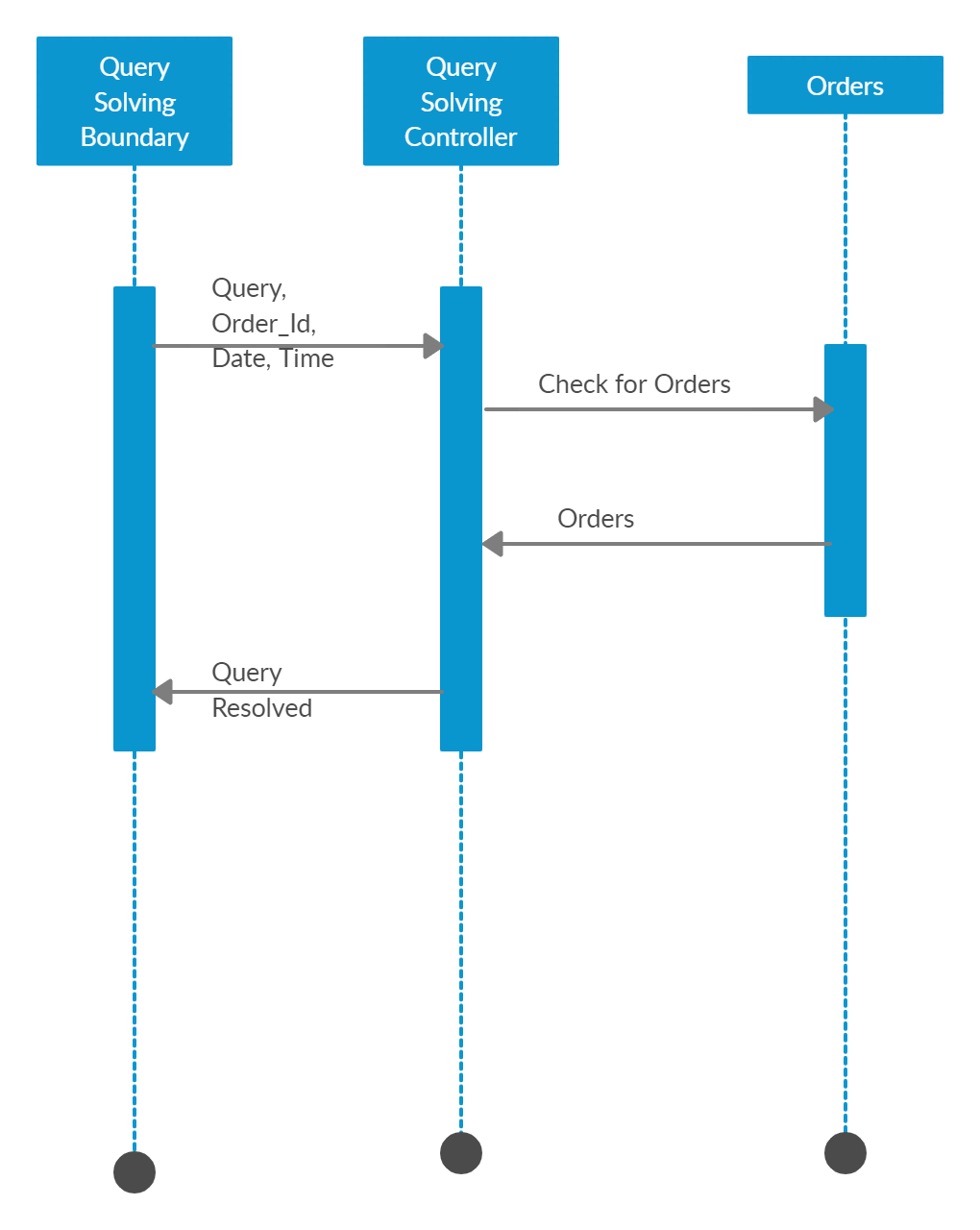
1. **Updation of delivery status**

****

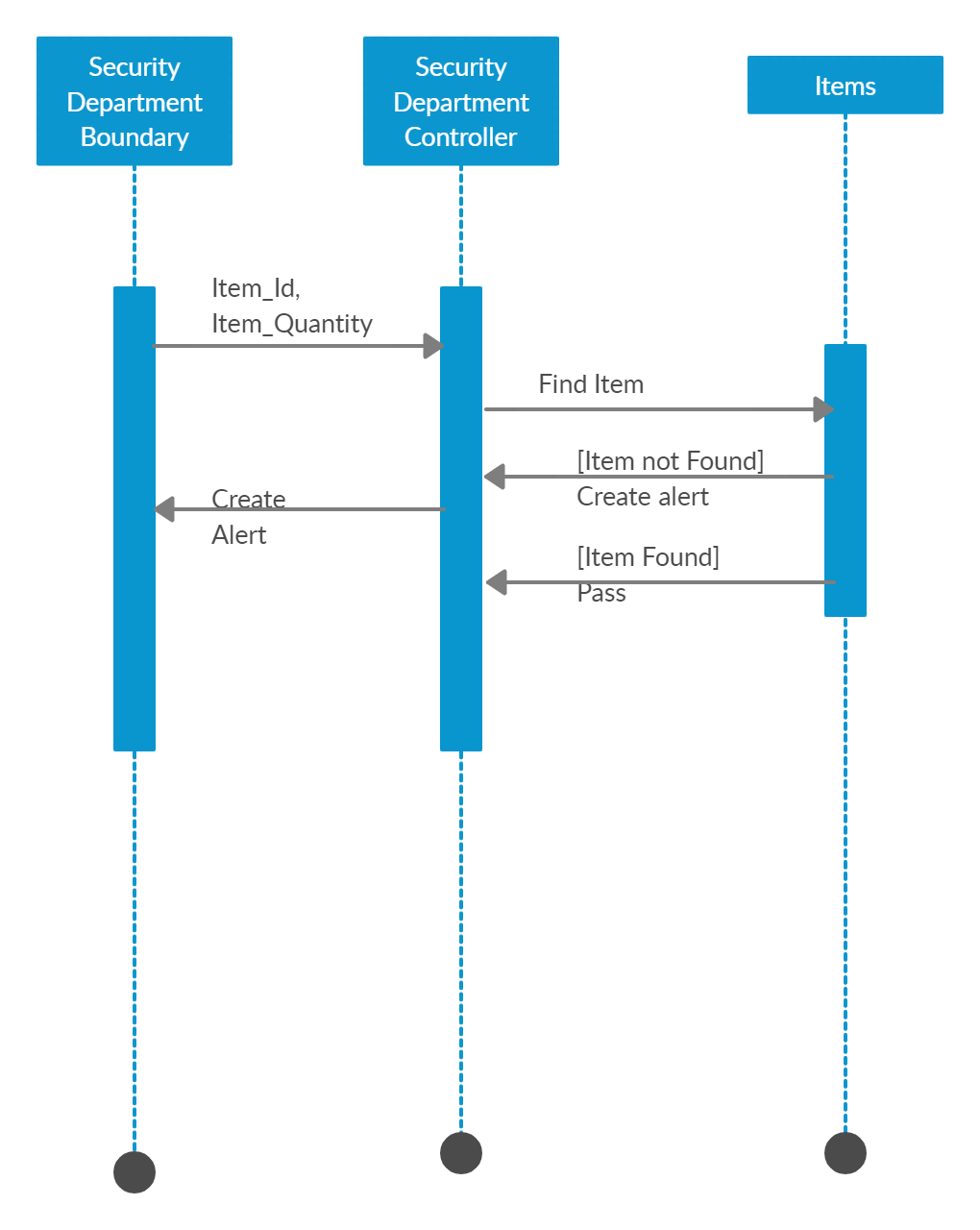
1. **Updation of payment status**

****

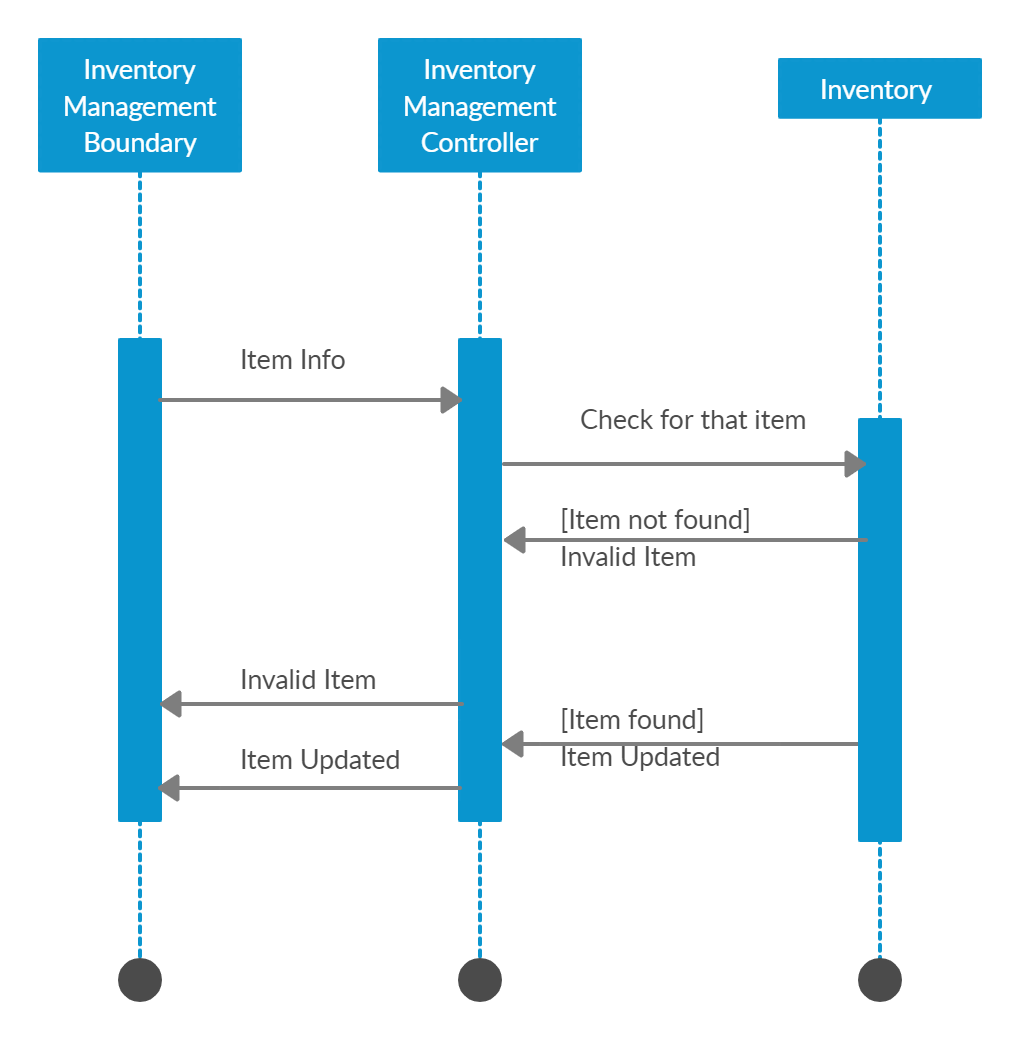
1. **Query Solving**

****

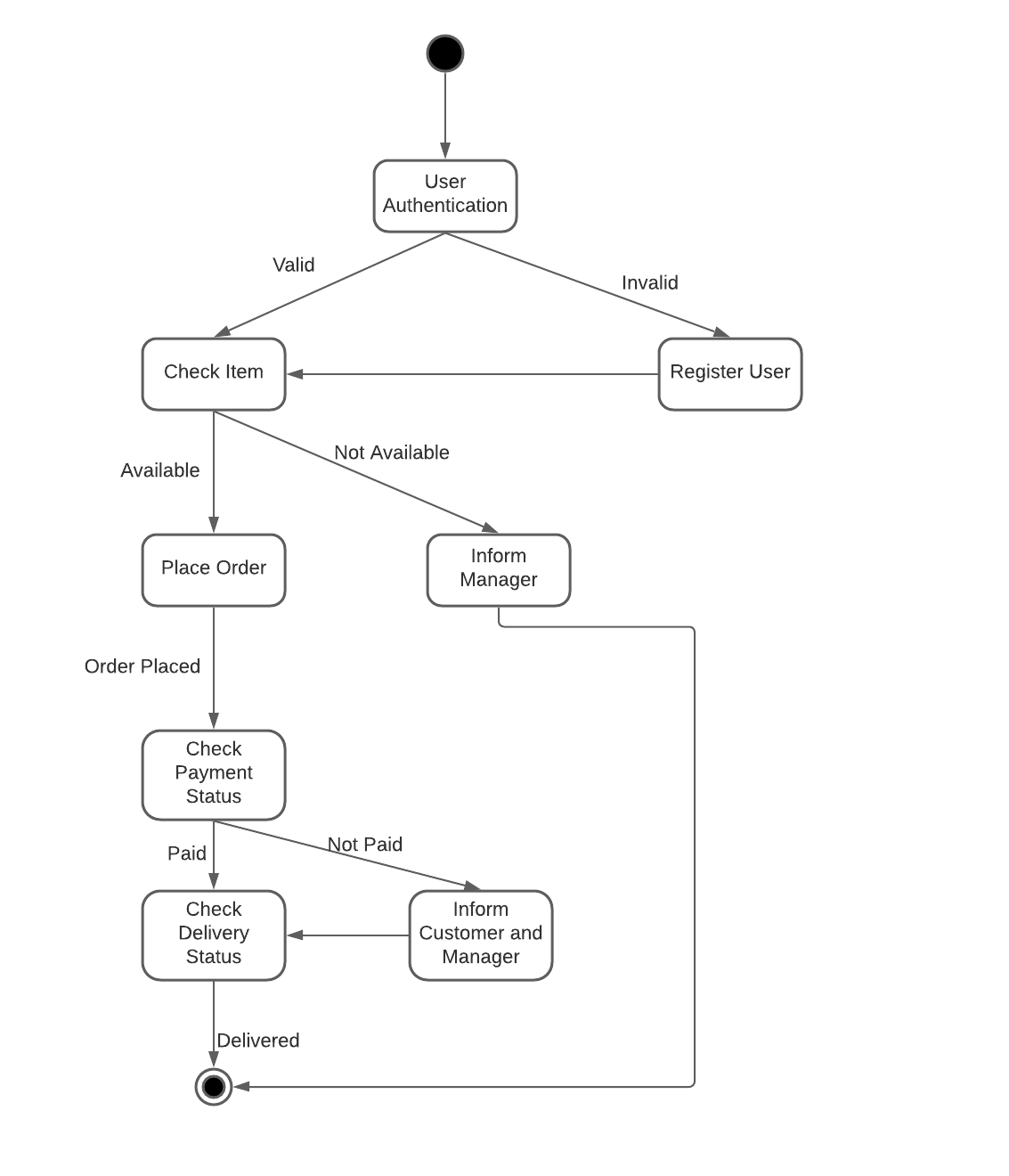
1. **Create Alert**

****

1. **Database Update by Inventory Manager**

****

**State Chart Diagram:**

****

**Submitted by:**

Sudipta Kumar Nayak: 1802041033

Aitik Dandapat: 1802041034

Udit Kumar Panda: 1802041035

