RESTAURANT AUTOMATION SOFTWARE

Problem Statement:

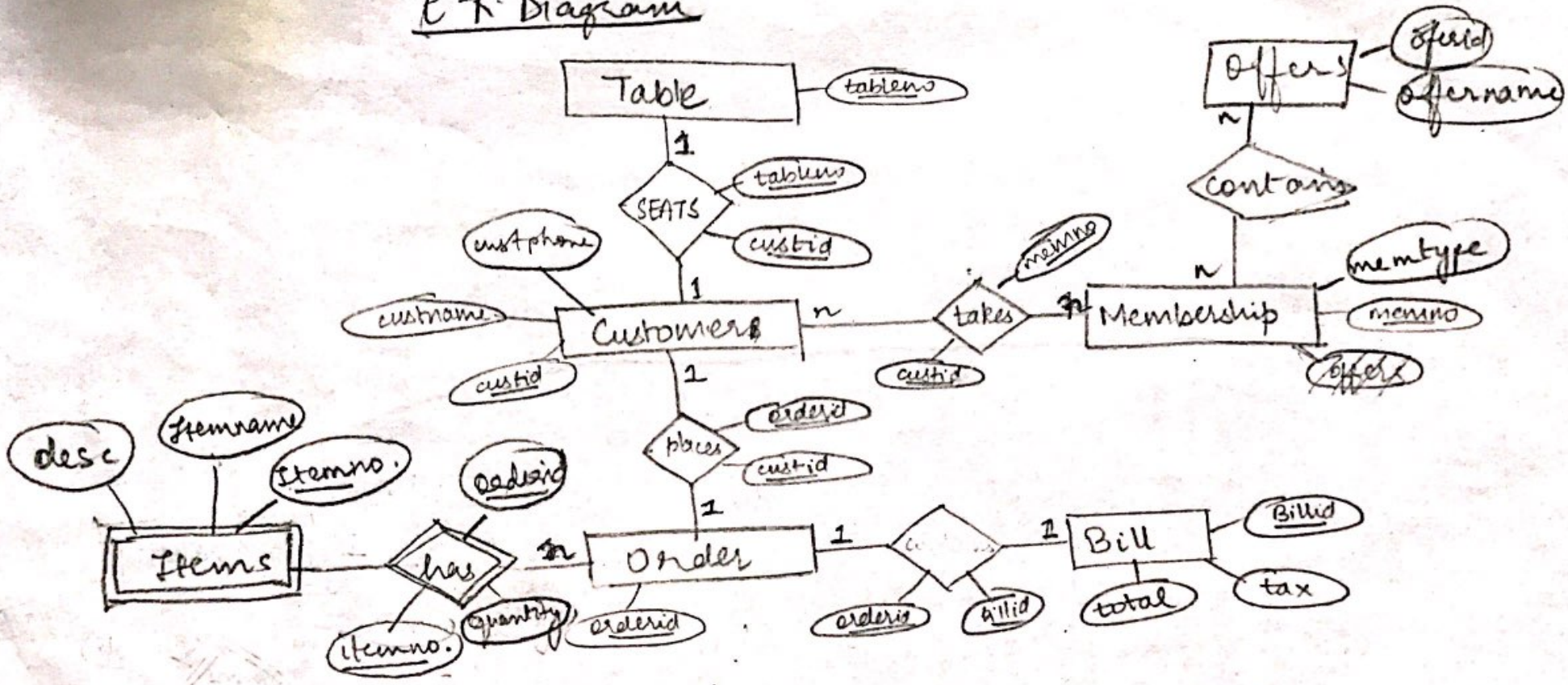
Problems faced by traditional restaurant management systems:

- Coordination of work activities
- Recognizing trends to take adv. of bestsellers
- Record maintenance

This project will computerize all restaurant operations so that it stores all information regarding customer's orders & staff activity. Hosts will be able to view table status with the click of a button. The waiters will be able to enter the customer orders quickly and share it electronically sent to the kitchen. The kitchen staff (cooks) will be able to view incoming orders & notify the assigned waiter where the food is ready. Bus boys will be able to view real-time floor status allowing them to know which tables are empty/occupied & clean/dirty. All of the restaurant information is organised & stored in the system database for viewing and analysis of revenue and revenue percentage per menu item & menu item popularity.

Scanned with CamScanner

E-R Diagram

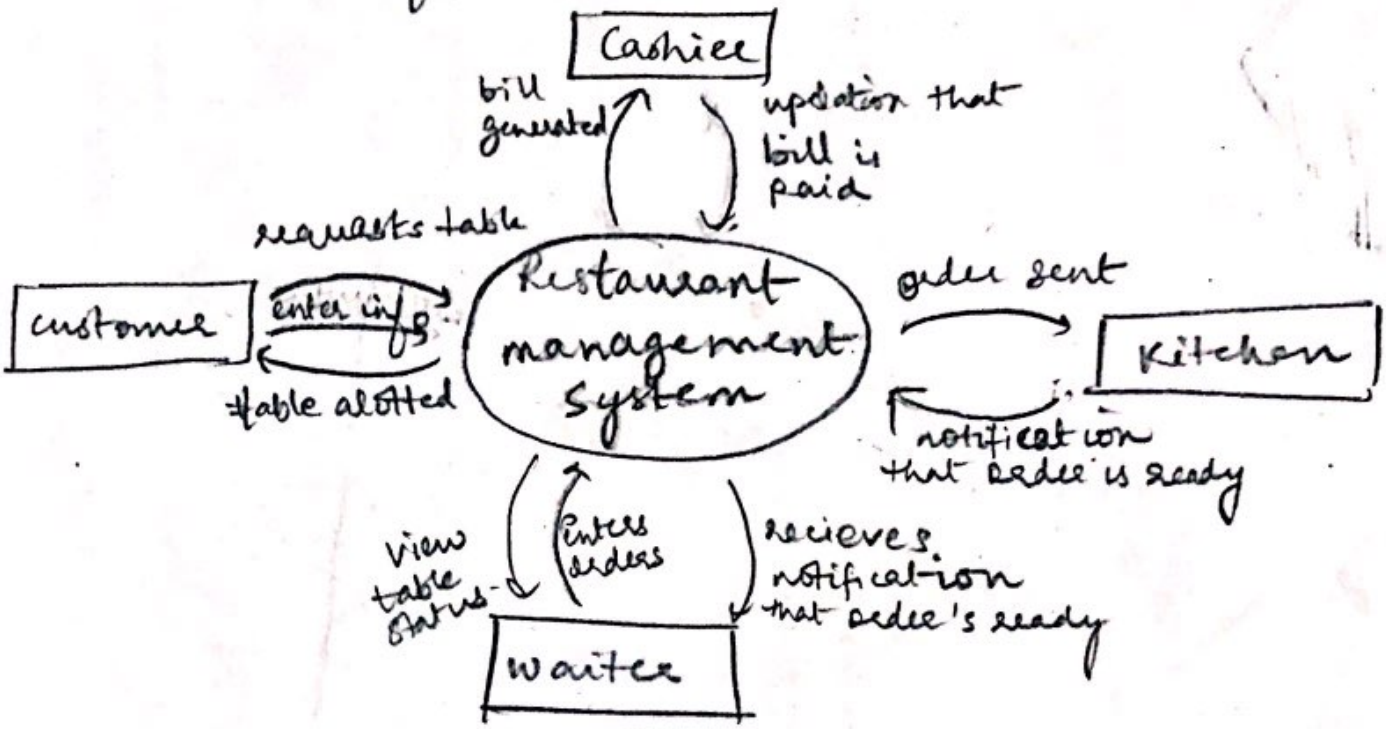


for 2:1:1 cardinality?

ER ?

Scanned with CamScanner

Context diagram

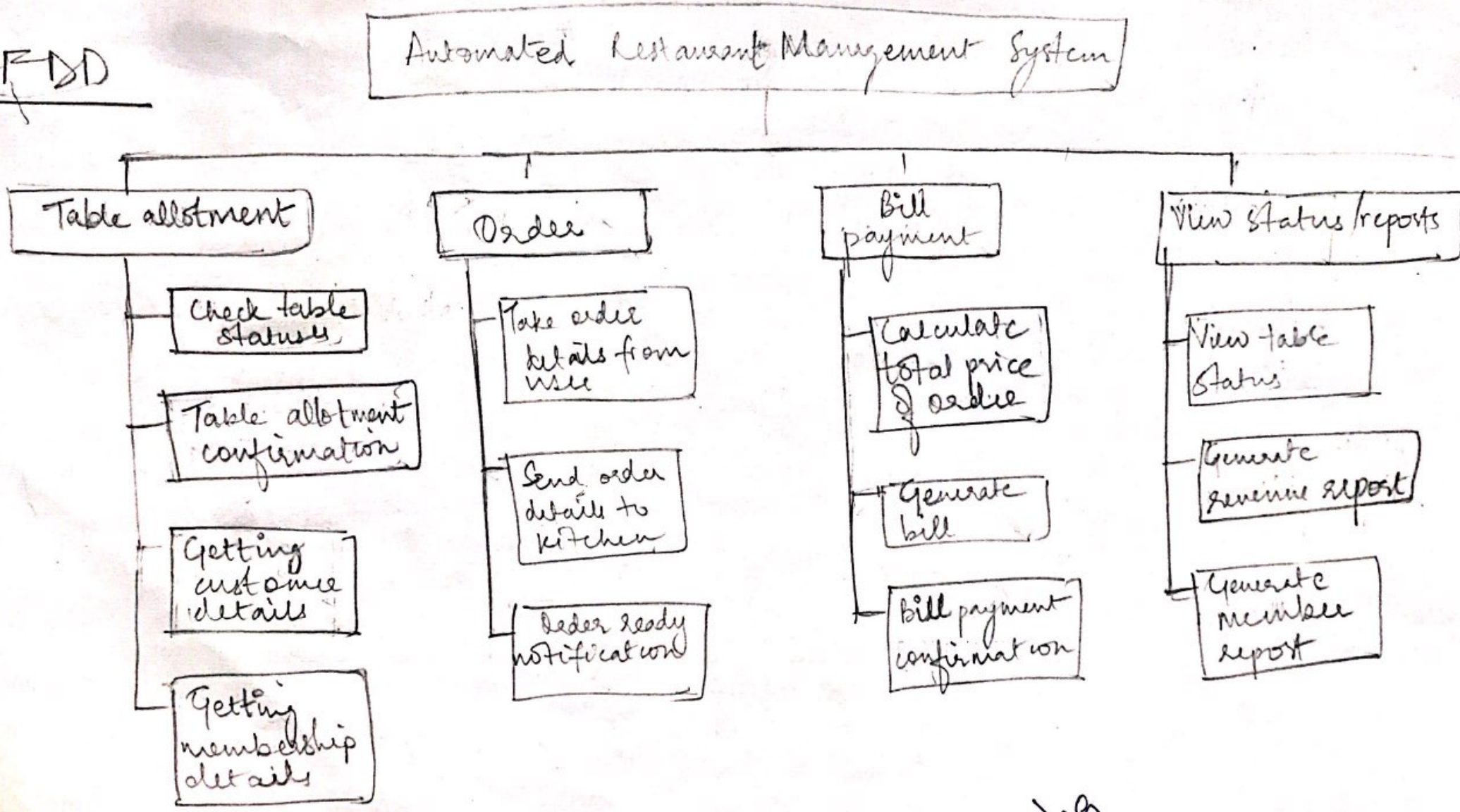


Event list

(1) customer asks for table.	Request for table [IN] Table allotted [OUT] customer info entered [IN]
(2) Waiter enters orders.	Order entered [IN] Order sent to kitchen [OUT]
(3) Waiter needs to see table status	Request for table status [IN] Table status given [OUT]
(4) Kitchen notifies that order is ready	Order status updated [IN] Notification sent to waiter [O
(5) Cashier needs to generate bill.	Bill is calculated & generated [O Status of order payment updated [IN]

Scanned with CamScanner

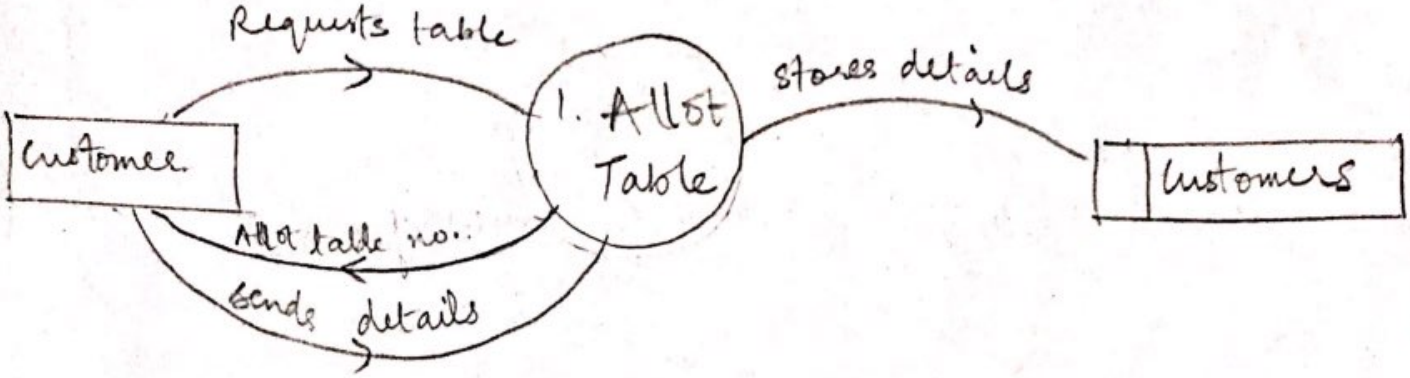
FDD

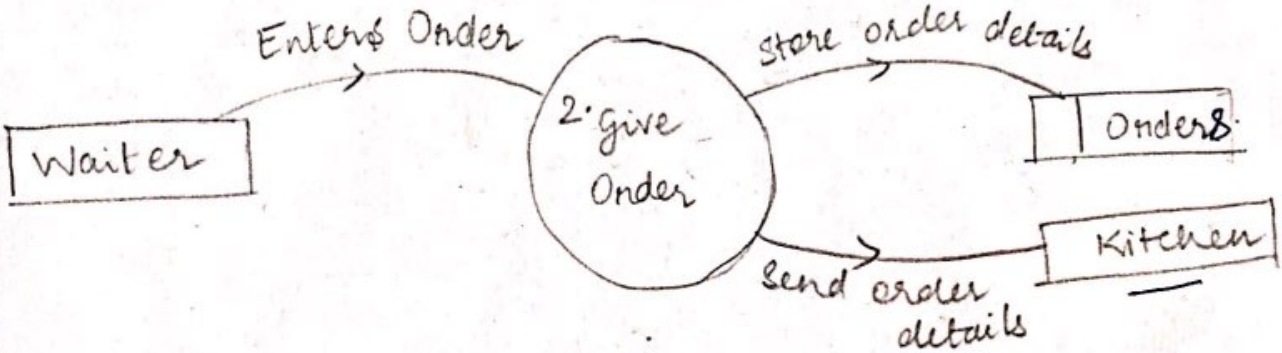


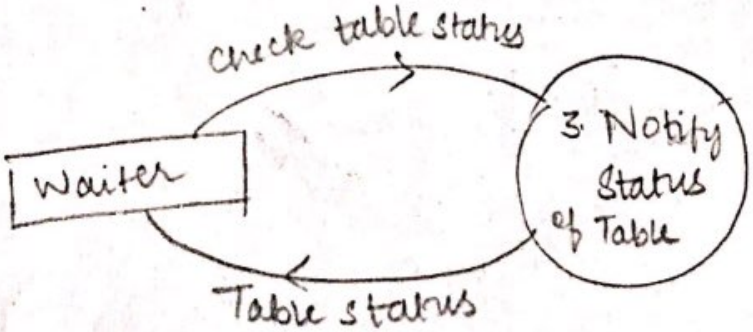
29/3/19

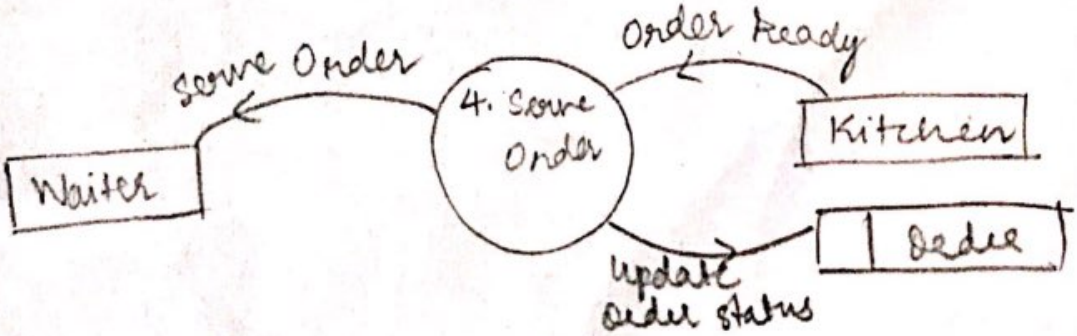
Scanned with CamScanner

Level 1 DFD

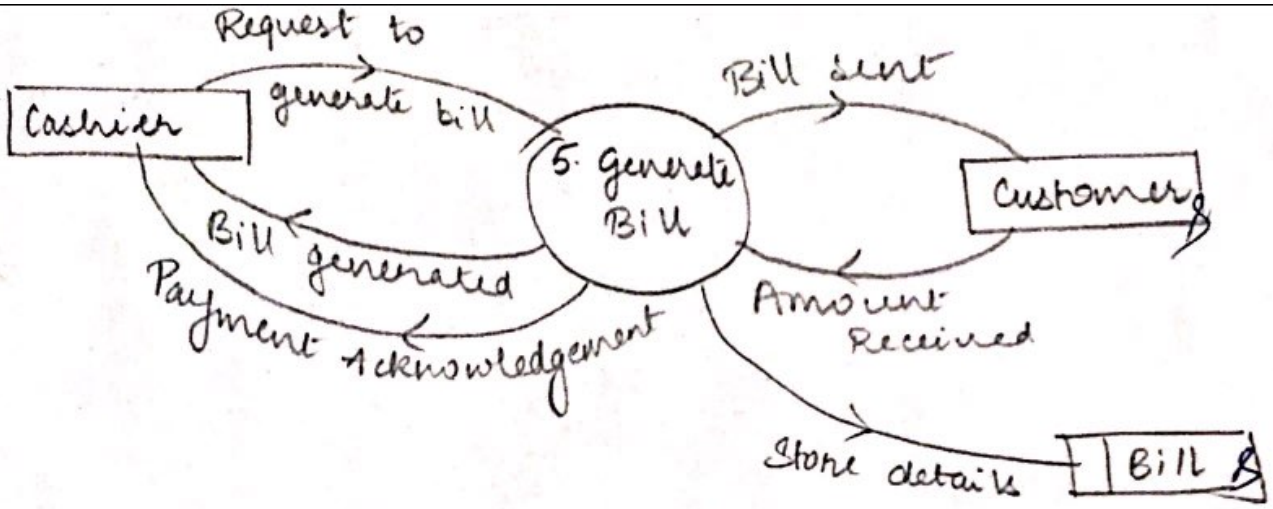








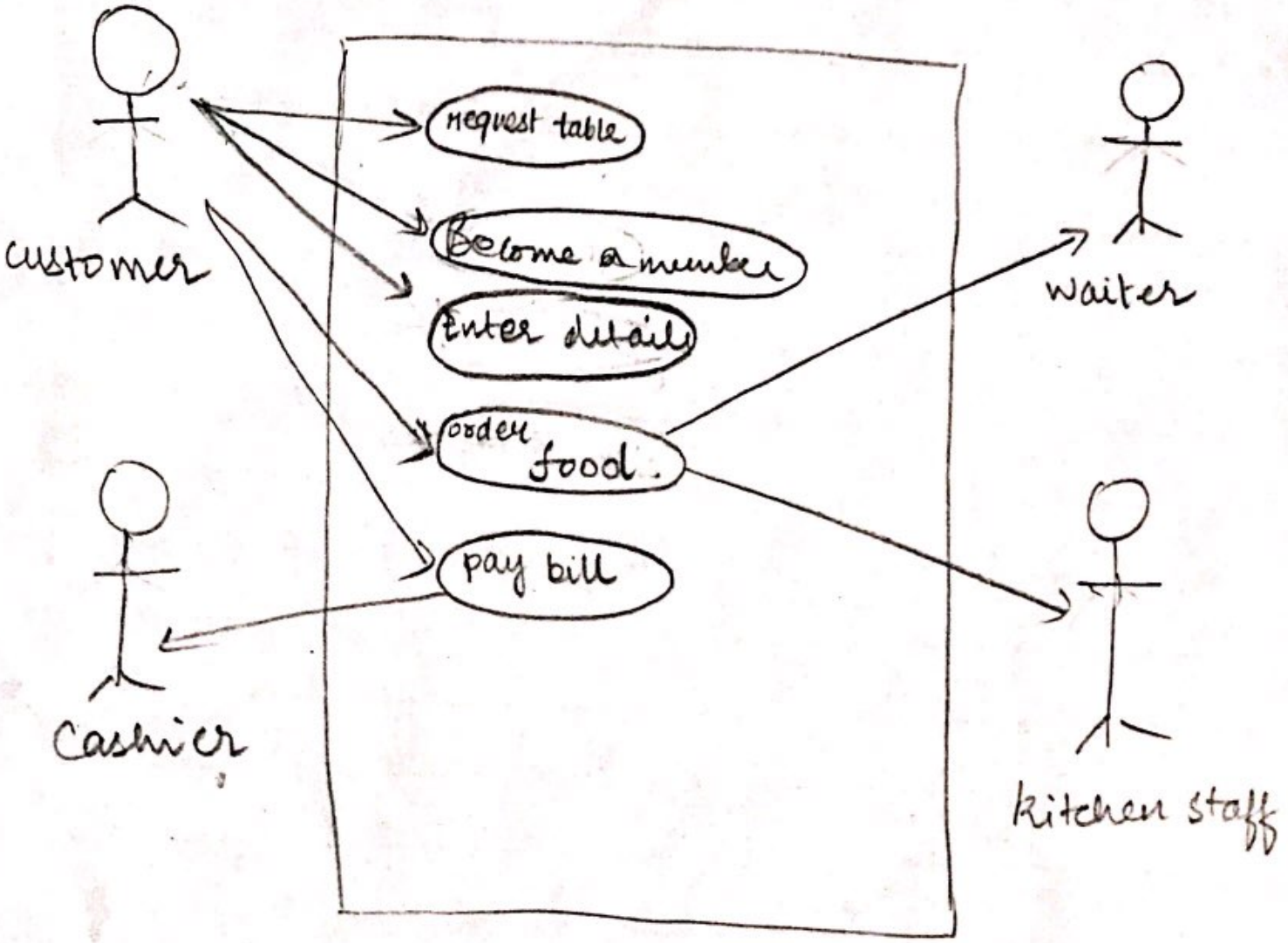
Scanned with CamScanner



29/3/19

Scanned with CamScanner

USE CASE DIAGRAM



NARRATIVE

29/3/1

UC1 Request Table

(1) Introduction : The use case ... request for a table.

(2) Presentation : Customer hasn't been allotted a table already.

(3) Pre-condition : ...

Scanned with CamScanner

UC1 - Request Table

① Introduction : This use case creates a request for a table.

② Actors : Customer

③ Pre condition : Customer hasn't been allotted a table already.

④ Post condition : System displays waiting time / table number allotted.

⑤ Event flow :

Basic flow -

1) Customer requests table.
2) Table is allotted & waiter is assigned to customer.

Alternative flow -

1) Customer requests table.
2) All tables are occupied. System shows waiting time.

Scanned with CamScanner

UC2- Enter details

(1) Introduction : This use case lets the customer enter his/her details. i.e. phone no., name, address

(2) Actors : Customer

(3) Precondition : Customer is not a registered member & has been allotted a table.

(4) Post condition : System displays membership options.

(5) Event flow :

Basic flow →

1) User enters his/her details into the system (name, phone no., address).
2) Details are saved to the customers data store along with an automatically allocated customer id
3) System displays membership schemes.

Alternative flow →

1) If user is already a registered member, only enter the customer id.
2) Retrieve customer details from database.

Scanned with CamScanner

UC3 - Become a member

(1) Introduction : This use case registers a new member into the system.

(2) Actors : Customer

(3) Precondition : Customer has entered details into the system and is not an already registered member.

(4) Post condition : System automatically gives appropriate offers according to the membership chosen by the customer.

(5) Event flow

Basic flow -

1) System displays different membership options & their offers.
2) Customer chooses one option.
3) Offers associated with the membership are displayed when customer places order.

Scanned with CamScanner

UC1 = Order food

(1) Introduction: This use case places an order & saves order details into orders data store.

(2) Actors: customer, Waiter, Kitchen staff

(3) Pre-condition: Customer has entered his/her details & allotted a table. Customer has been shown Menu ~~items~~ by the waiter.

(4) Post-condition: The Kitchen staff (Chef) is notified about any incoming order. The chef notifies the assigned waiter when food is ready.

(5) Event flow:

Basic flow →

1) Waiter shows menu to the customer on system
2) Customer gives the order to the waiter, who saves the order details into data store.
3) The system notifies the chef of the new order.

Alternative flow →

1) Waiter shows menu to customer on system.
2) Customer places order to waiter but the order is out of stock. Waiter notifies customer about non-availability & suggests substitute.
3) System notifies chef of the order.

Scanned with CamScanner

UC5 - Pay bill

(1) Introduction: this use case records payment of bill by customer to the cashier.

(2) Actors: Customer, Cashier

(3) Pre-condition: The customer's order has been served.

(4) Post-condition: Feedback is taken from the customer

(5) Event flow:

- Basic flow -

1) System calculates bill amount from order details & applies offers/discounts.

2) The calculated amount is stored into bills data store and displayed to customer along with order details.

3) The customer pays the amount to the cashier by cash

Scanned with CamScanner

DATA DICTIONARY

letter = [A/B/c .... Y/Z]

digit = [0/1/2 .... 8/9]

character = [letter, digit, special characters]

Table

tableID = 5{character}6

Customer

custID: 5{character}10

custname: fname + lname

fname: 2{letter}20

lname: 2{letter}20

custphone: countrycode + 10{digit}10

Membership

memtype: 10{characters}15

memno: 5{digit}6

offerID: 5{digit}7

offername: 10{characters}20

Order

orderID: 3{digit}7

itemID: 4{digit}5

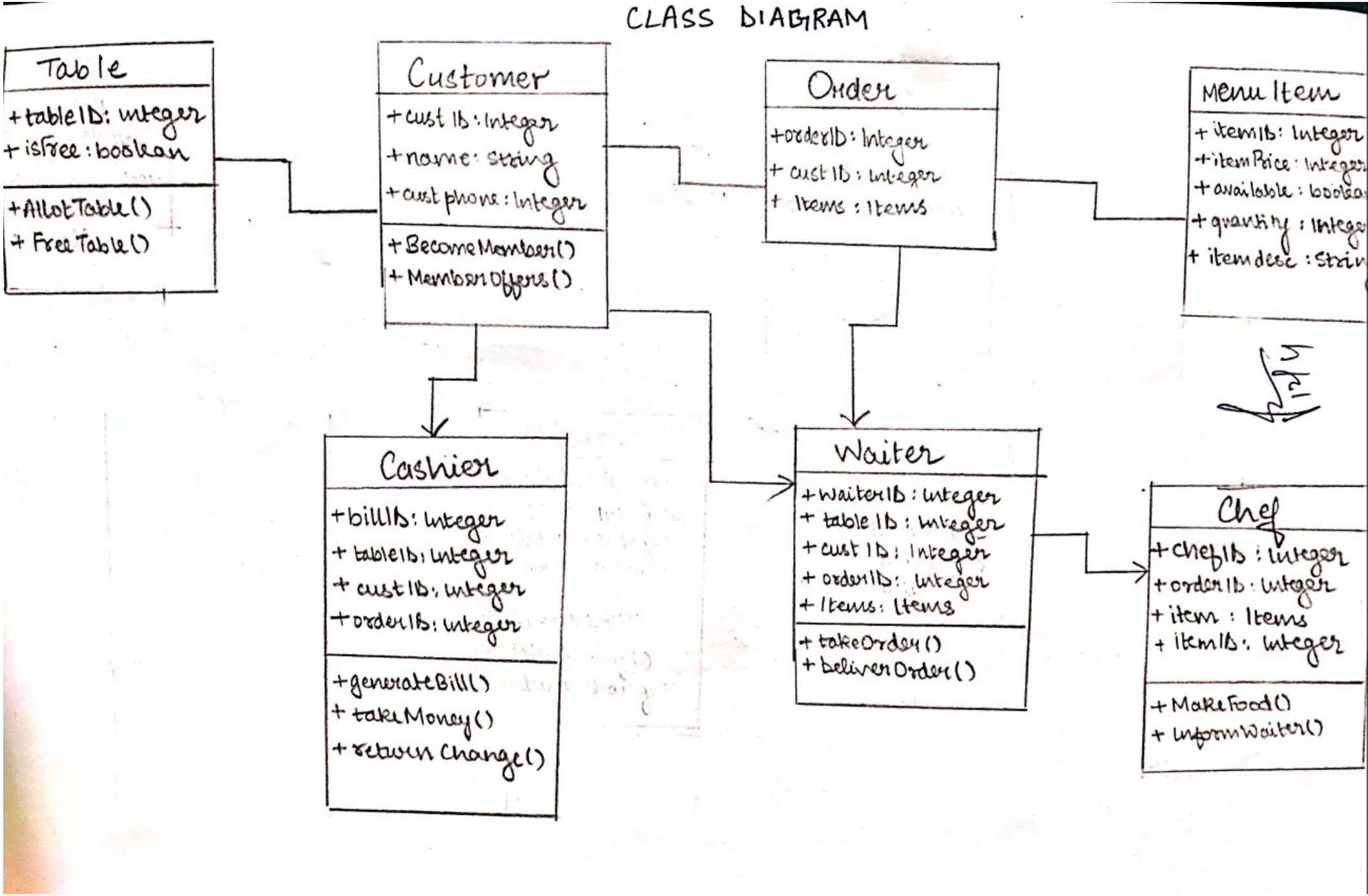
quantity: 2{character}4

itemname: 6{letter}20

description: 9{character}20

Scanned with CamScanner

CLASS DIAGRAM



Scanned with CamScanner