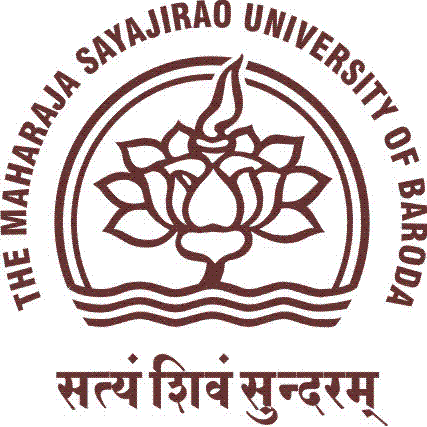
**THE MAHARAJA SAYAJIRAO UNIVERSITY OF**

**BARODA**



A BUSINESS MODELLING PROJECT REPORT

On

**Café Management System**

Guided By

Ms. Poonam Shah

In fulfilment for the award of the degree

Of

***BACHELOR OF COMPUTER APPLICATIONS***

In

The Department Of Computer Applications

**27th November, 2022**

# Team Members

**Group No: 10**

|  |  |  |
| --- | --- | --- |
| Sr No. | Name | PRN |
| 1 | Vandit Harshit Shah | 2020033800115133 |
| 2 | Dhwani Chauhan | 2020033800116032 |
| 3 | Drashti Barot | 2020033800007314 |
| 4 | Krushanu Bhatt | 2020033800114667 |

# Index

|  |  |  |
| --- | --- | --- |
| Sr No. | Title | Page No. |
| 1 | Definition | 4 |
| 2 | Description | 4 |
| 3 | Objective | 5 |
| 4 | About Our System | 6 |
| 5 | Stake Holders | 6 |
| 6 | Use Case Diagram | 7 |
| 7 | Activity Diagram | 10 |
| 8 | Sequence Diagram | 18 |
| 9 | Class Diagram | 25 |
|  |  |  |
|  |  |  |
| 12 | Work Distribution | 29 |
| 13 | Conclusion | 30 |

# DEFINITION

Every year our nations Food Industry is increasing by 12-14%. And every restaurant and cafes give around 13-17% of there earnings as salary to servants and other working people.

We made a project of 78-degree café to manage their café. The software we made is in the form of website. The main purpose to make management of the café more smoother as well as more efficient using this technology.

# **DESCRIPTION**

# 78 degree café website needs no servant to interact with the customer for the order and order conformance procedure. Customer can scan the QR code which is placed on the table they are. After scanning they will be able to select item they desire, A customer can order as many times they want.

All the orders will be seen by the manager and also the Head Chief order procedure will be seen by the manager. Along with this manager can manage table (create new table, delete if not required or update table details). In addition, he will also be able to maintain the Menu item and inventory by adding new category of food, adding new food item, change the pricing and will also get the notification of food item that will be less than 3 in quantity left.

Head Chief can see the orders which are placed by the customer and then can update the order status to served when the order is been served.

# **OBJECTIVE**

* Red Sky Blood Bank insures to supply required blood components on time so that no life suffers due to lack of blood components.
* Red Sky Blood Bank provides its services to the people in need of blood with utmost care and safety.
* Red Sky makes sure that donated blood, are safe before they are used in blood transfusions and other medical procedures.
* Recipient can interact with the system and can request for particular blood type.
* Donor can register for blood donation by interacting with the system. All required test are done before and after blood donation
* Red Sky make the blood stocks more systematic, manageable and secured.

# **ABOUT OUR SYSTEM**

The project is made using the MERN stack (MongoDB, Express JS, React JS, Node JS) Technology.

For front-end we have used React, For Back-end Node JS and Express JS and for Database Mongo DB. The sole aim of the project is to make the café management easier. The website when started shows the login page. Manager has its own credentials to login and Head Chief has its own credentials to login.

Both have different working space and both can access the at the same time.

# **STACKHOLDERS / CONTRIBUTOR**

* Customer
* Manager
* Head Chief

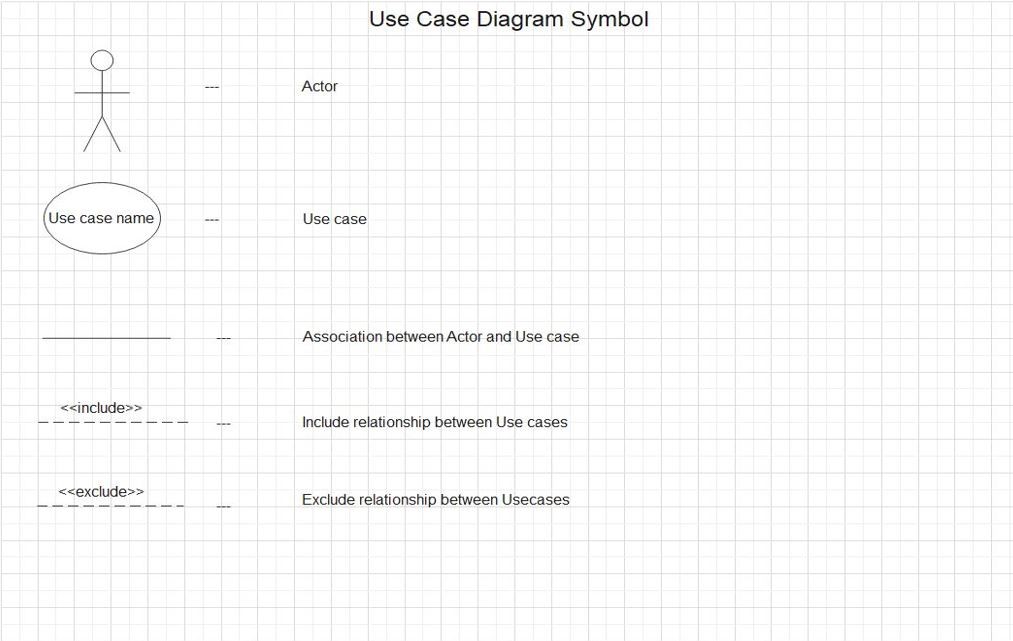
# **UML DIAGRAMS**

**USE CASE DIAGRAM**

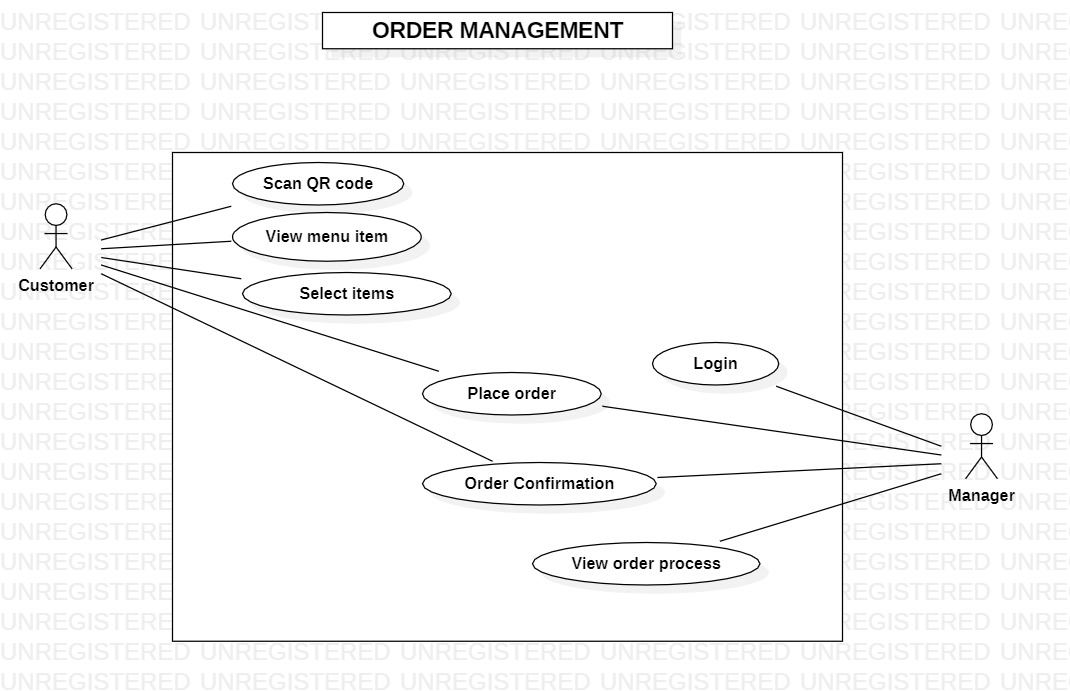
A *use case* specifies the behavior of a system or a part of a system, and is a description of a set of sequences of actions, including variants, that a system performs to yield an observable result of value to an actor

**ACTORS**

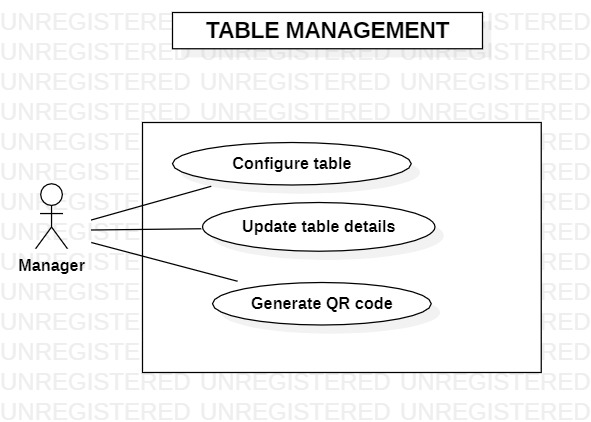
* Customer
* Manager
* Head Chief



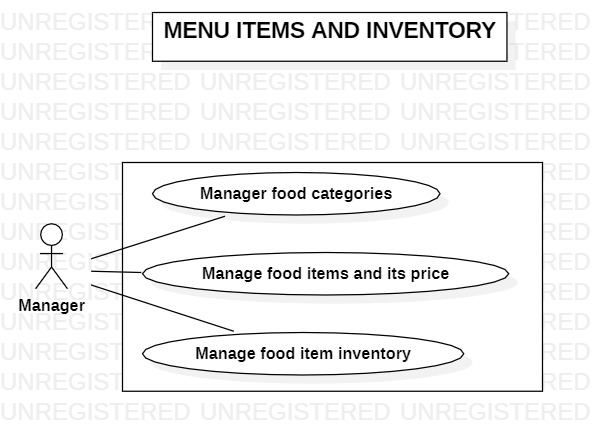
**Order Management Use Case**



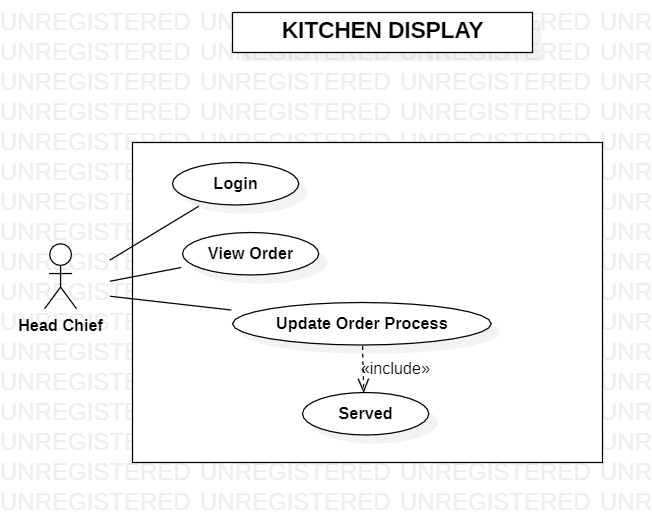
**Table Management Use Case**



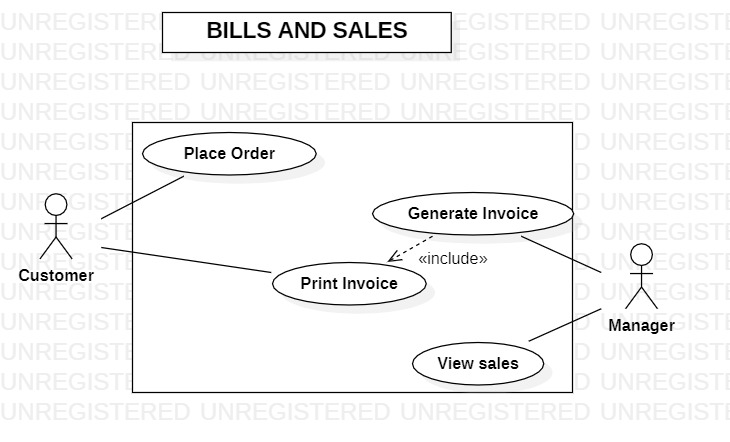
**Menu Item and Inventory Use Case**

****

**Kitchen Display Use Case**

****

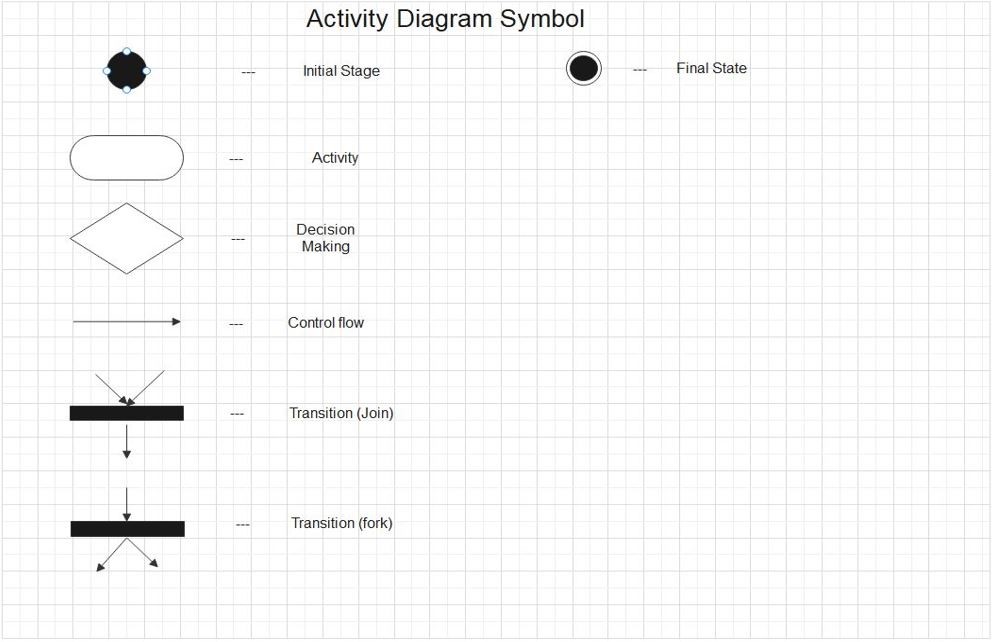
**Bills and Sales Use Case**

****

|  |  |  |
| --- | --- | --- |
| Actor | Action Performed | Description |
| Customer | Scan QR code | The first thing to order is to scan QR code from the table |
|  | View Menu Item | After scanning customer will be able to see the menu items available |
|  | Select items | After viewing customer can order items |
|  | Place Order | Place Order button will be pressed |
|  | Order Confirmation | After selecting items customer will have to confirm the order |
|  | Print Invoice | Lastly customer can View and Print the Invoice |
| Manager | Login and Register | First Manager has to login to manage the system |
|  | Place Order | Manager will get all information of order placed by the customer |
|  | View Order Process | The order is in process or been served – Both can be known by the manager as updated by the Head Chief |
|  | Configure Table | New Table Configuration is possible |
|  | Generate QR code | Each table will have unique QR code |
|  | Update Table details | Table details such as table size can be updated if require in future |
|  | Manage Food Category | New Food category can be generated |
|  | Manage Food item and its Price | New Food item can be added or previous food item pricing can be updated |
|  | Manage Food item Inventory | If any food Item is below 3 than it will give notification |
|  | Generate Invoice | Manager can generate invoice |
|  | View Sales | Whole Sales of the Café is been recorded |
| Head Chief | Login | To manage orders Head Chief has to Login to the system. |
|  | View Order | Order Tickets are viewed by the Head Chief |
|  | Update Order Process | Head Chief can Update Order process by clicking on served. |

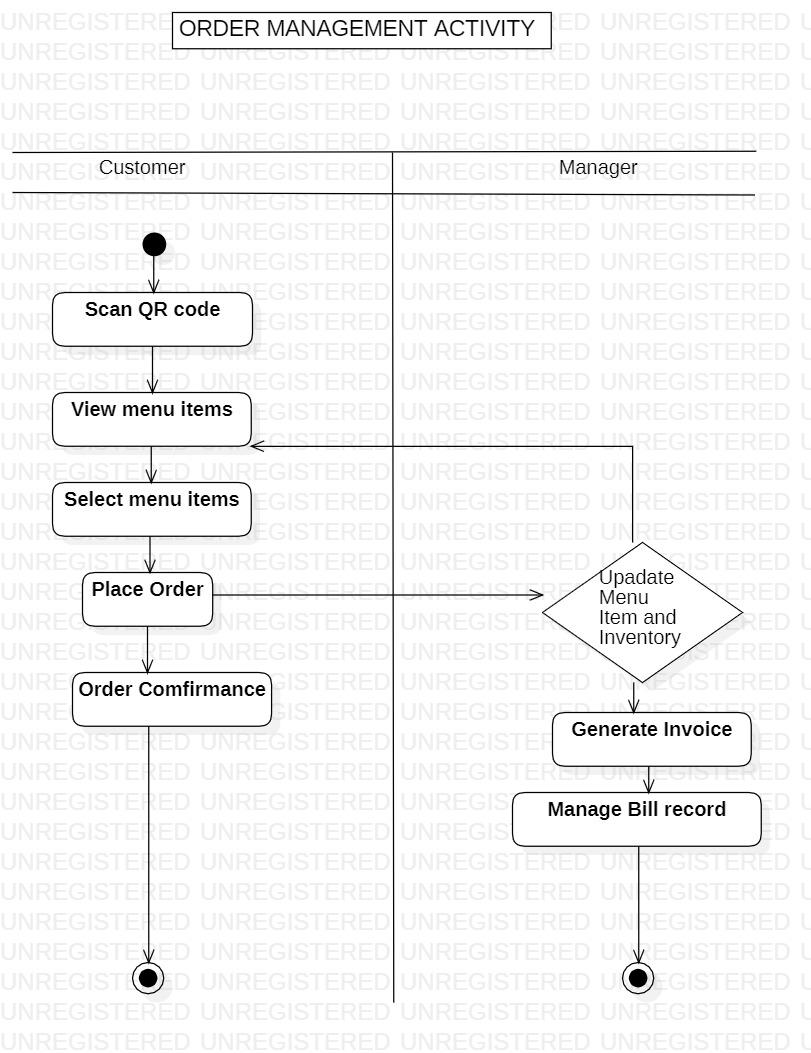
**ACTIVITY DIAGRAM**

An activity diagram is a behavioural diagram i.e. it depicts the behaviour of a system. An activity diagram portrays the control flow from a start point to a finish point showing the various decision paths that exist while the activity is being executed.



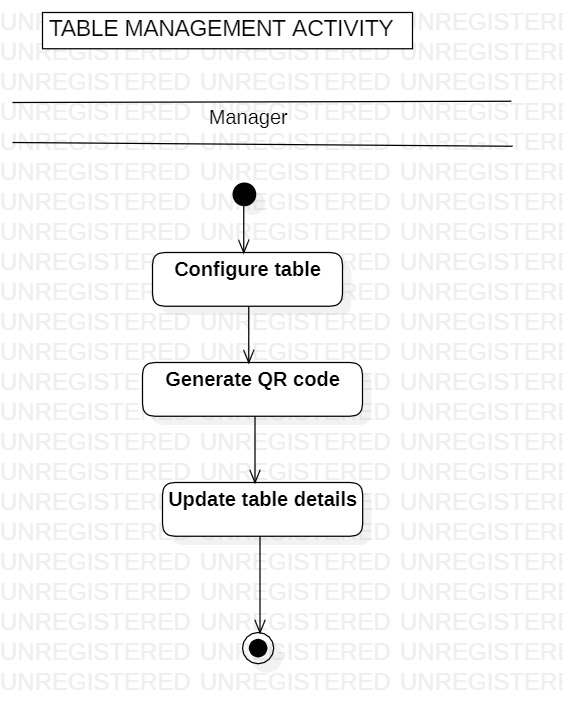
**MODULES**

1. **Order**
2. **Table Configuration**
3. **Menu Items and Inventory**
4. **Kitchen Display**
5. **Order**

****

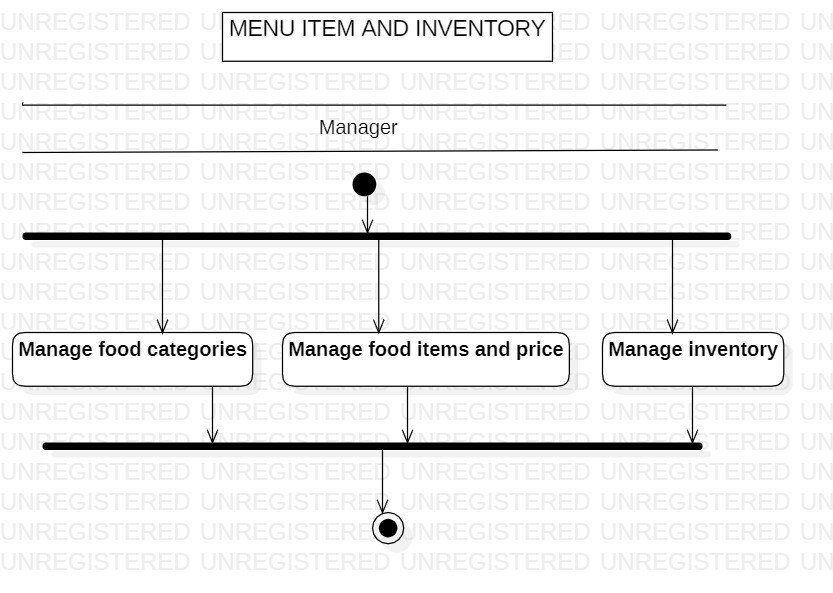
* Customer has to Scan the QR code first, than Menu items are visible from that customer will select the required items and place order. Order confirmance will done and Manager will also confirm the order. If required customer can all Print Invoice.

1. **Table Configuration**

****

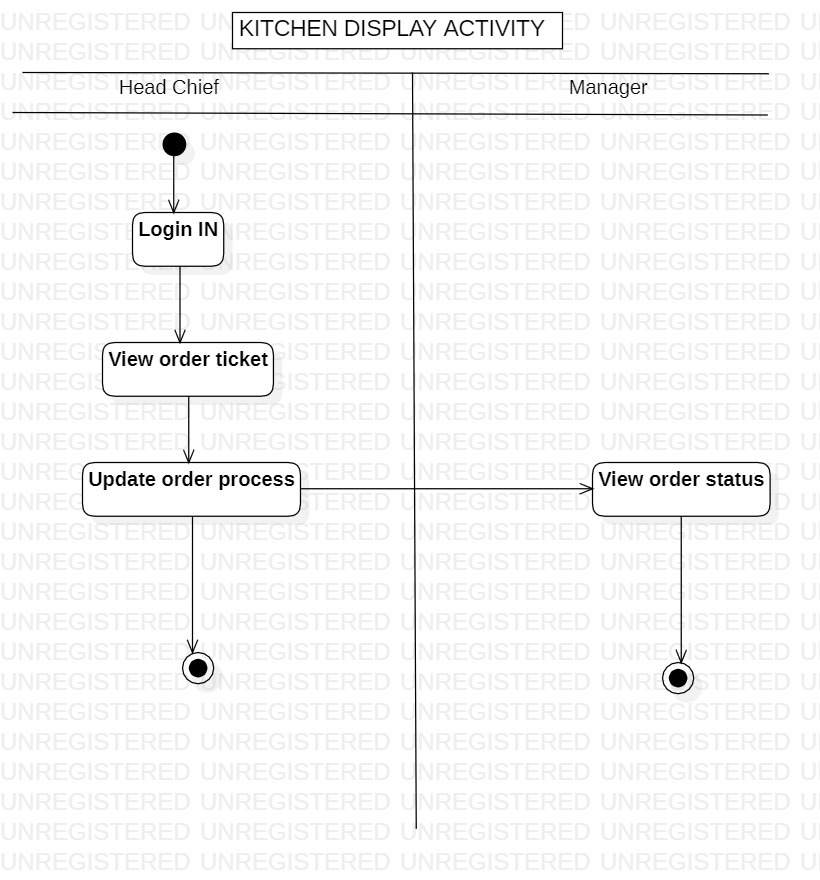
* To generate QR code manager will configure and enter all details required and then generate QR code.

1. **Menu Item and Inventory**

****

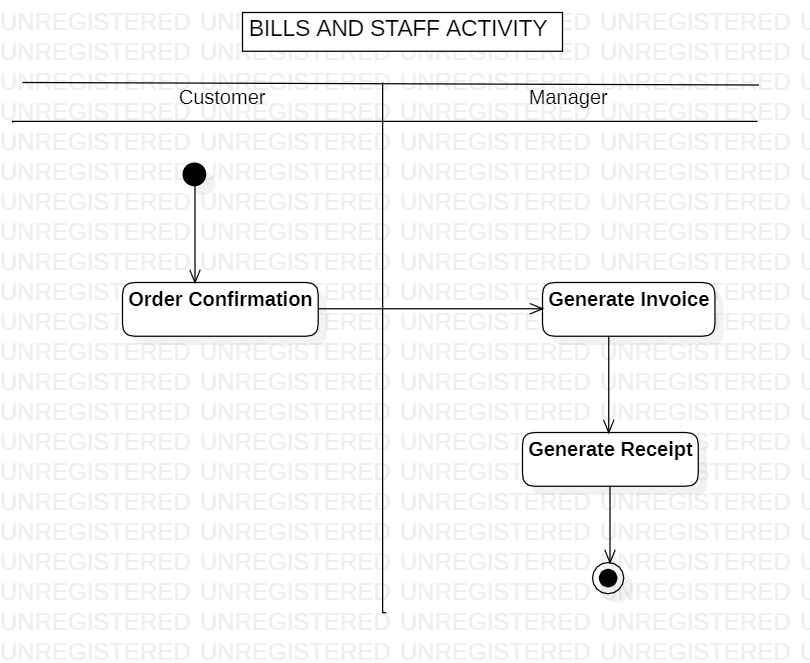
* Manager can simultaneously manage Food categories, food items and price and Inventory.

1. **Kitchen Display**

****

* Head Chief will first login and then can view order tickets and Update Order process when done.

1. **Bills and Sales**

****

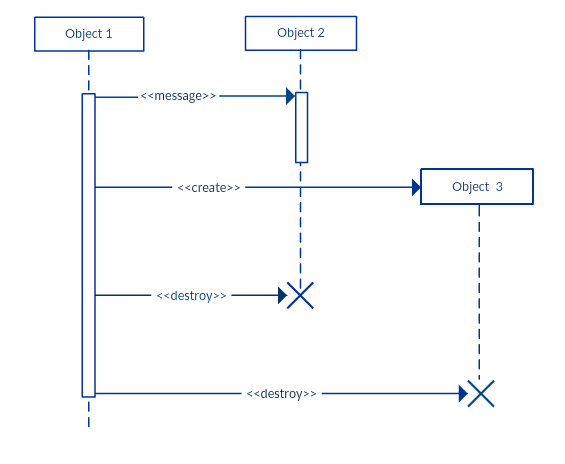
* After the confirmation of order customer can view their order and can also download the receipt.

**SEQUENCE DIAGRAM**

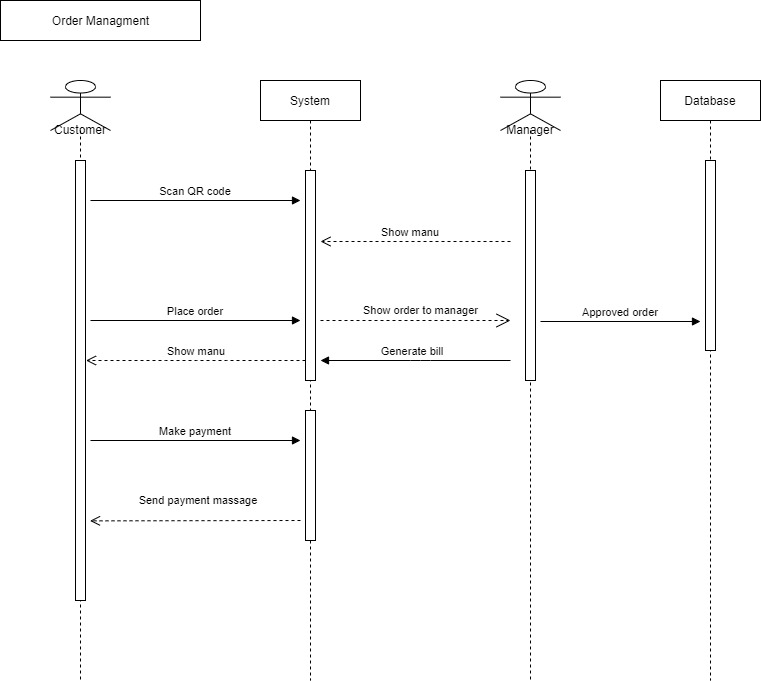
A sequence diagram is a type of interaction diagram because it describes how—and in what order—a group of objects works together. These diagrams are used by software developers and business professionals to understand requirements for a new system or to document an existing process

**MODULES**

1. **Order**
2. **Table**
3. **Menu**
4. **Kitchen**

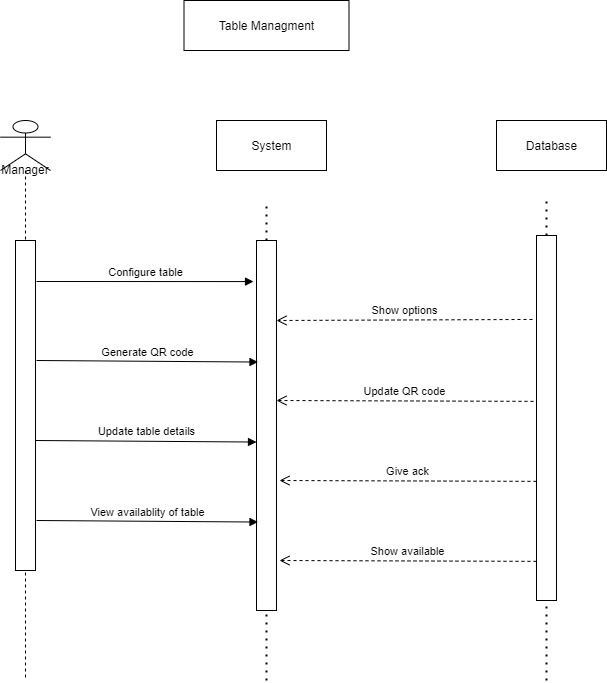


1. **Order**



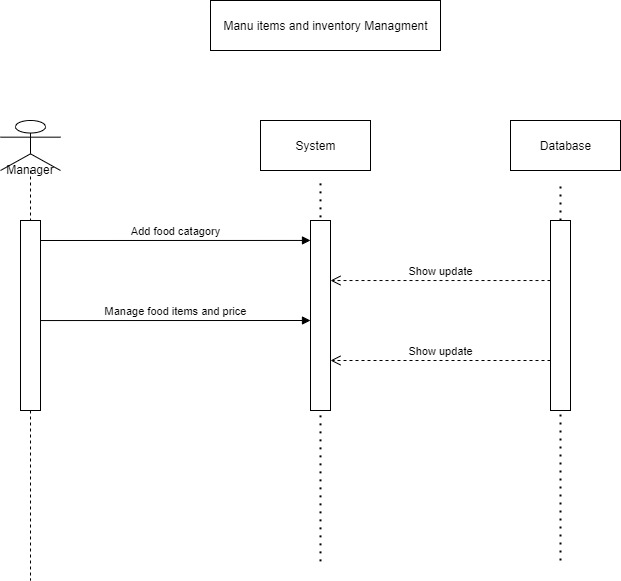
* If the user is new, he must register in the system first. He will enter all required details, the details will be verify and will be added to the Database and it will show the message of Registration Successful. If the user is already registered the he can login into the system, he will enter the credential provided, and the system will verify the credential from database, if its correct then only the user will be able to logged into the system

1. **Table**



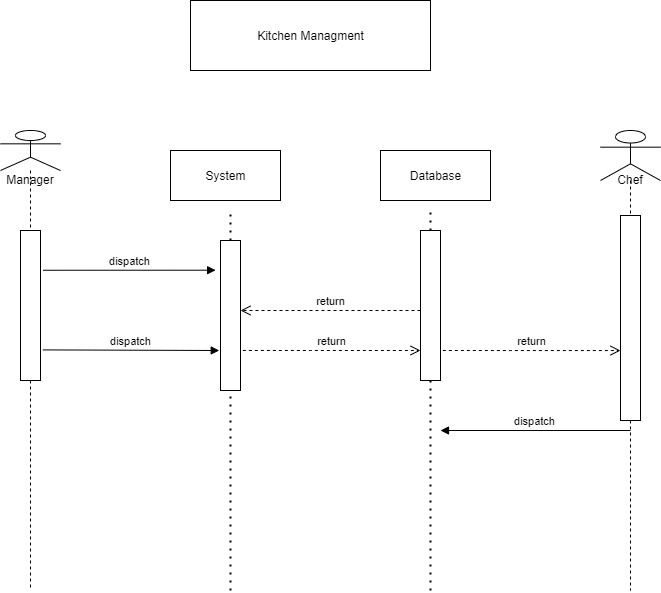
* To modify account details user must login into the system first, so the user will enter credential and will login to the system. Once the user is logged in he can view its account detail, he will modify the details and will submit updated details, the modified details will be entered into the database by system and the message will be show – Details Updated, after that the user can view its updated details.

1. **Menu**



* To donate blood, the donor must undergo some necessary testing, the reports will be generated and will entered into the system by Medical Staff. If the blood is safe, the Medical Staff will collect the blood from the Donor and the two parallel actions will performed. The Medical Staff will perform required testing on blood and will generate the report which will be added into Donors database, and on the other hand the Medical Staff will request the Admin to Update the Inventory, the Admin will update the inventory. If the blood is not safe, Medical Staff won’t collect blood from the Donor

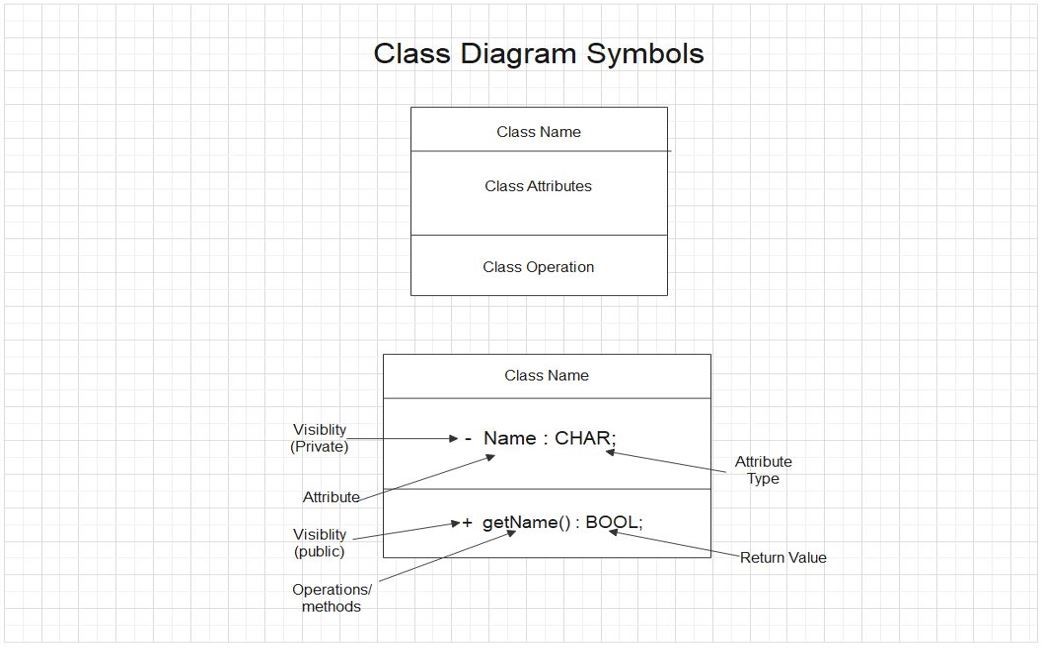
1. **Kitchen**



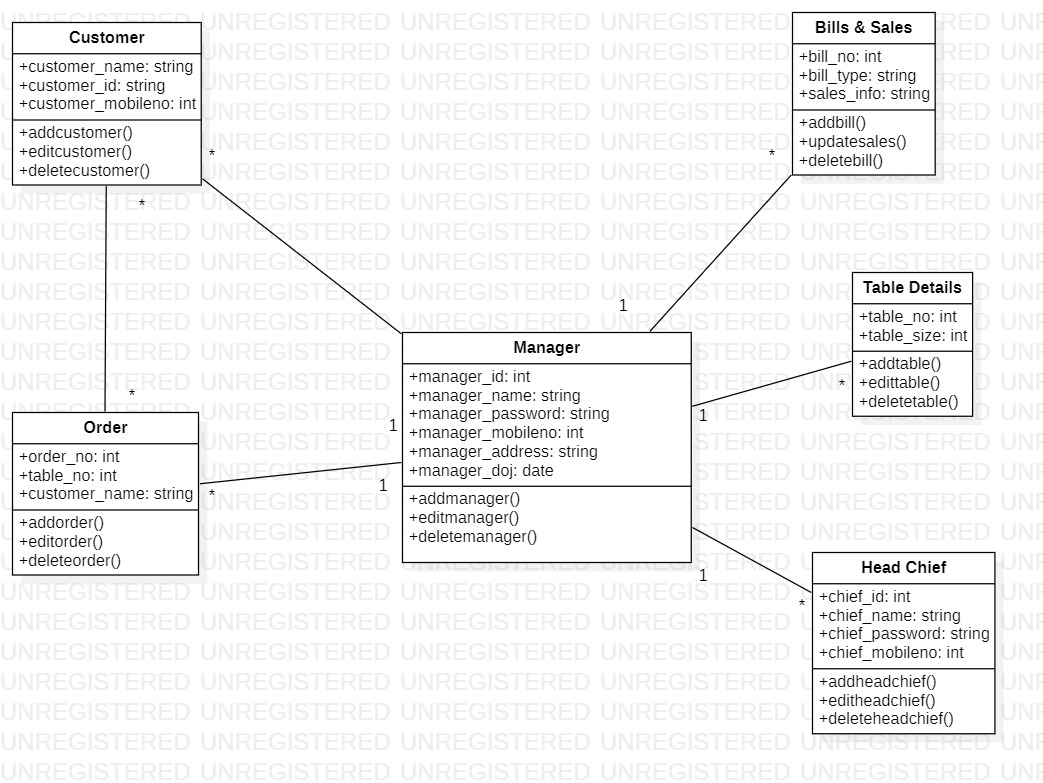
* Only admin can manage Blood Inventory, manage include viewing, adding, deleting, modifying inventory. If the Admin want to view the inventory, he will select the Blood Type in the system and the system will fetch the data of inventory from the database, then the data will be shown to the admin. If the admin want to modify the data, admin will select the data and will enter updated data. The system will then updated the inventory in the database and the message will be shown about Inventory Updated, then admin can view updated inventory.

**CLASS DIAGRAM**

The class diagram is the main building block of object-oriented modelling. It is used for general conceptual modelling of the structure of the application, and for detailed modelling, translating the models into programming code. Class diagrams can also be used for data modelling.



**CLASS DIAGRAM**



* There are six classes in our system : Customer, Order, Manager, Bills and Sales, Table Details, Head Chief.

**Work Distribution**

|  |  |
| --- | --- |
| Activity | Names |
| Use Case Diagram | Dhwani, Vandit, Drashti, Krushanu |
| Activity Diagram | Dhwani, Vandit, Drashti, Krushanu |
| Sequence Diagram | Dhwani, Vandit, Drashti, Krushanu |
| Class Diagram | Dhwani, Vandit, Drashti, Krushanu |
| System Front-end | Dhwani, Vandit, Drashti, Krushanu |
| System Back-end | Dhwani, Vandit, Drashti, Krushanu |
| System DataBase | Dhwani, Vandit, Drashti, Krushanu |
| Documentation | Dhwani, Vandit, Drashti, Krushanu |
| Presentation | Dhwani, Vandit, Drashti, Krushanu |

**CONCLUSION**

To conclude, this business modelling project delivers a documentation on the “Red Sky” Blood bank system. Also, it delivers a business implementation strategy, supported by an assessment model and practical guideline of the enterprise.

Red Sky Blood Bank provides Blood bags to the people in need. Safety precautions are taken at every step of the process. And aims to fulfil maximum number of blood requests so that no life is lost due to lack of blood.