# Database I – CSE5330

# Nisa Management System

## Lakshmi Meghana 1001417274

## Shripal Shah 1001238188

**Nisa Management System** is a web management system aimed for the business owner(s) to manage business by tracking orders, employees, and customers.

It will also allow the owner to monitor employee’s progress and add bonuses for every order placed. This app will motivate employee to stay on top of their game by being incentivized for every order placed by them or by the customers they introduce.

# Project Description Assignment

**ORGANIZATIONAL OVERVIEW**

‘NISA CREATIONS’ is a company owned by Mr. Samir R. Shah. The company manufactures Jute Bags, purses, and various other jute items. The company is based in Mumbai and Kolkata. Mr. Samir Shah handles the business in Mumbai and Mr. Paresh Shah in Kolkata. The jute items are then sold in wholesale to their customers.

**DESCRIPTION OF THE CURRENT SYSTEM**

Their current system of work is very much manual, devoid of computers and is completely paper based. Big registers are maintained for every transaction. All the details must be maintained minutely and must be taken care of avoiding manual errors.

The assistants handle the manual system of taking orders and managing them. There is a possibility of these assistants making mistakes by taking wrong input or any sort of calculation error. Preserving the records of each customer, orders of the customer, maintaining the records of each employee, reports for analysis etc. must be done manually. In short a bulk of paper work is done to maintain records of transactions and orders.

Also managing assistants becomes a tedious job for the owners of the organization.

***Problem***

A typical business must manage the following key areas:

**Employees-** To maintain and manage its employees

**Customers-** To keep an active track and provide high quality services to its customers.

**Products-** To keep track of what products are being manufactured and the demand of new products in the market.

**Orders-** To keep a track on the orders placed by its customers, and monitor the progress of orders placed.

**App Summary**

**Requirements of what data must be represented**

* Firstly, the system should maintain all the records about the employees working in the organization.
* The System should be able to add new employees, edit employees’ details and can remove the employees’ record as and when required.
* The system should isolate the order reservation information reserved by an individual employee from the other employee.

The system should also maintain security so that no intruder can enter the system without valid login credentials.

* The system should store and display the details about the products which the organization manufactures, store and display the information of customers who reserves the order through employee as well as through admin and should be able to add/remove the customers’ and products details.

**End Users:**

* Owner
* Employee
* Customer

**Queries**

**Owner**

1. Display the list of employees whose incentive is greater than $5000 for a month.
2. List the products ordered by the customers.
3. Get the name and cost of the products whose size is L.
4. List the customers who have placed orders.
5. Display the top 10 customers.
6. Add, Update, and Delete an Employee.
7. Add, Update, and Delete Customer.
8. Add, Update, and Delete a Product.
9. Add an order for a customer whose id is c009, delete the order or update the order where order id is o066.

**Employee**

1. Display the product details and the sales cost in USD in the orders ordered by his customer.
2. List the products that are in high-demand in the market.
3. Display the customer’s and order’s details where order quantity in more than 5000 piece.
4. Display the customer who hasn’t placed any order for more than one month.

**Customer**

1. Display my orders sorted by the latest one first.
2. Create an order for a product.

**Assumptions about the database**

***App Functions***

**Owner View:**

* The web-application will allow the owner to maintain order, product, employee, customer information in the system’s database.
* The business owner will be allowed to Insert, Update, and Delete employee’s, customer’s, product’s, and order’s information into the database.
* The web-application will allow the owner to monitor the employee’s performance and let him incentivize the employees.
* The web-application will let the owner to provide feedback to the employee.

**Employee View:**

* The web-application will allow the employee to maintain order, product, and customer information in the system’s database.
* The employee will be allowed to Insert, Update, and Delete customer’s, product’s, and order’s information into the database.
* The web-application will allow the employee to view feedback given by the owner.
* The application will allow employees to place orders on behalf of customers

**Customer View:**

* The web-application allows the customer to add new orders, and view previous orders

***Entity types***

* **Owner**
* **Employee**
* **Customer**
* **Order**
* **Product**
* **Login\_details**
* **Size\_cost**
* **Employee\_customer**
* **Incentive**

**Relationships:**

* Owner adds employees 1..m
* Employee adds customers 1..m
* Customer creates account 1..1
* Customer orders product 1..m
* Owner incentivize Employees 1..m
* Owner adds products 1..m

|  |  |  |  |
| --- | --- | --- | --- |
| Entity | Description | Attributes | Primary Key |
| Customer | This entity contains all the basic details of the customers. | C\_fname -varchar(20)  customer\_lname -varchar(20)  customer\_contactno -int(12)  customer\_emailid -varchar(20)  customer\_nationalid  customer\_bankid -varchar(20)  customer\_address - -varchar(20) | customer\_id |
| Owner | This entity contains all the basic details of the owner | owner\_id - varchar (10)  owner\_name - varchar(20)  owner\_number- int(12)  owner\_address- varchar(20) | Owner\_id |
| Login\_details | This is the superclass of  Customer, employee and owner entities. This stores the login details | userid -varchar(26)  password - varchar(26) | User\_id |
| Employee | This entity stores the information about employees | Employee\_id – varchar(26)  Employee\_number – int(10)  Employee\_address – varchar(26)  Employee\_dateofbirth– varchar(26)  Employee\_name- varchar(26)  Employee\_department- varchar(26) | Employee\_id |
| Order | This entity stores order details | Order\_id – varchar(26)  Product\_item\_no- int(20)  Customer\_id – varchar(26)  Order\_date- varchar(26)  Order\_quantity-int(20) | Order\_id |
| Product | This entity stores the details of products manufactured by the owner | product\_id - varchar(10)  product\_name -varchar(20)  product\_color-varchar(20) | product\_id |
| Size\_cost | This entity type stores the size and cost relationship | Product\_item\_no- int(20)  Product\_id- varchar(20)  Product\_size- varchar(20)  Product\_cost- int(20) | Product\_item\_no |
| Employee\_customer | This entity stores the information about employee and the customer | Employee\_id - varchar(20)  Customer\_id- varchar(20) | Customer\_id |
| Incentive | This entity stores the information about the incentive of individual employee for a month | Incentive\_id – varchar(20)  Employee\_id – varchar(20)  Incentive\_amount\_usd – int(5) | Incentive\_id |

***User Roles***

***Owner***

This user can view, add, update, and delete items to key areas, assign completion states to orders. This group will be open to all owners. They will log in using their username and password

***Employee***

The Employee can view, add, update, and delete items to key areas except the Employee entity type.

***Customer***

This user can place orders.

**Platform**

To facilitate the expected active business demands, this will be a web-application with its first iteration aimed at web-login interface.

**Internal Functions**

*AddEmployee*

This will add a new Employee-User’s name and profile information in the database. This function will let the employee to access the system database.

*AddCustomer*

This will add a new Customer’s name and profile information in the database.

*AddOrder*

This will add a new order placed by the customer. User can select from a predefined list of options of products for the order.

*Modifyorder*

This will allow the user to edit existing order.

*MarkOrderAsComplete*

This puts an order in a complete state.

*GenerateIncentive*

This will calculate incentive for the employees on every order placed for the customer

*GenerateReport*

This will generate reports on past orders*.*

***Conclusion***

From the phase 1 of the project, we have learned how to gather requirements from the user. Since the project is to be delivered to the client, we got practical exposure on how to interact with them.

For getting better output, we will be using Agile SCRUM methodology.

This system will automate the traditional methods of maintaining files for storing the documents.