

Bakery Management System

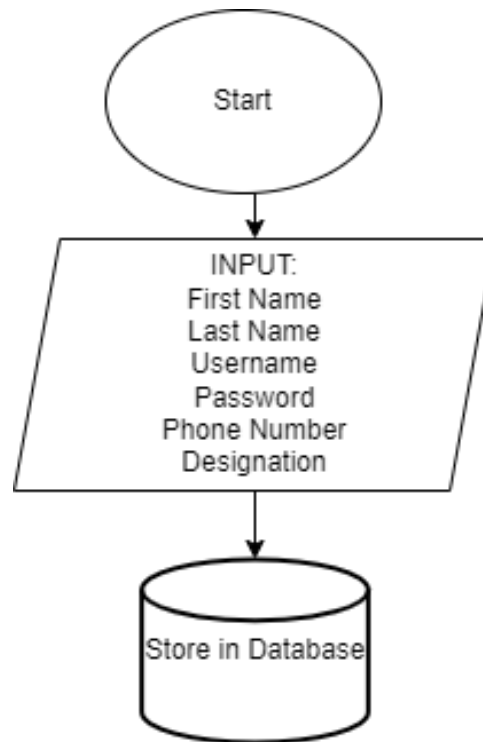
Criterion B: Design

Contents

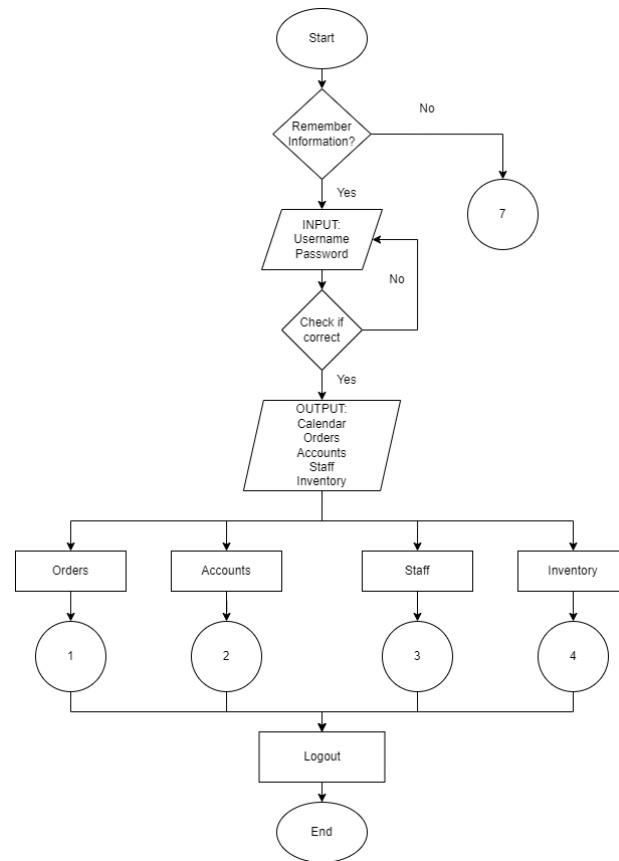
System Flowchart	2
Registration	2
For the Admin	3
For the Staff	3
Modular Diagrams	8
For the Admin	8
For the Staff	9
Screen Designs	10
Login Screen	10
Register Screen	10
Forgot Information Page	11
Admin Screens	11
Staff Screens	15
Database	17
UML Diagrams	19
Admin	19
Staff	21
Validation	22
Test Plan	22
Testing Functionality of database:	23

System Flowchart

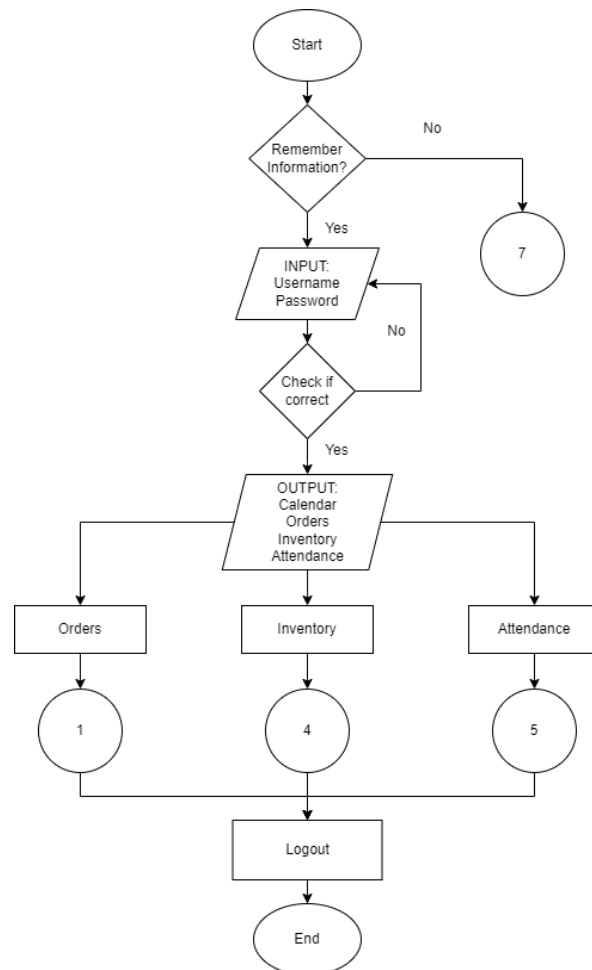
Registration

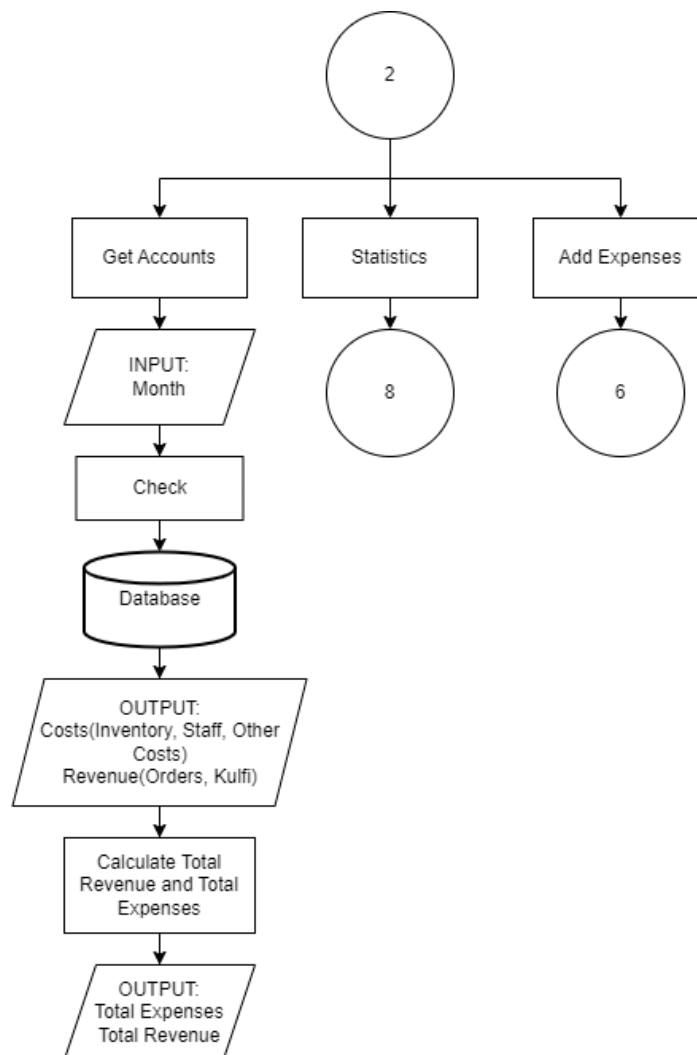
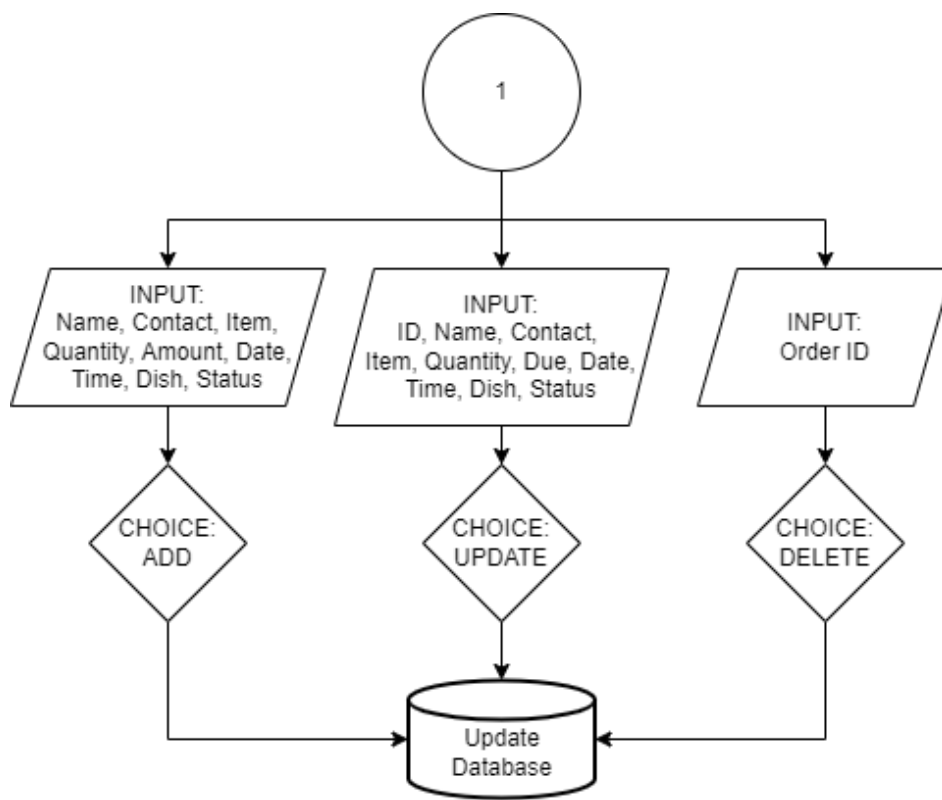


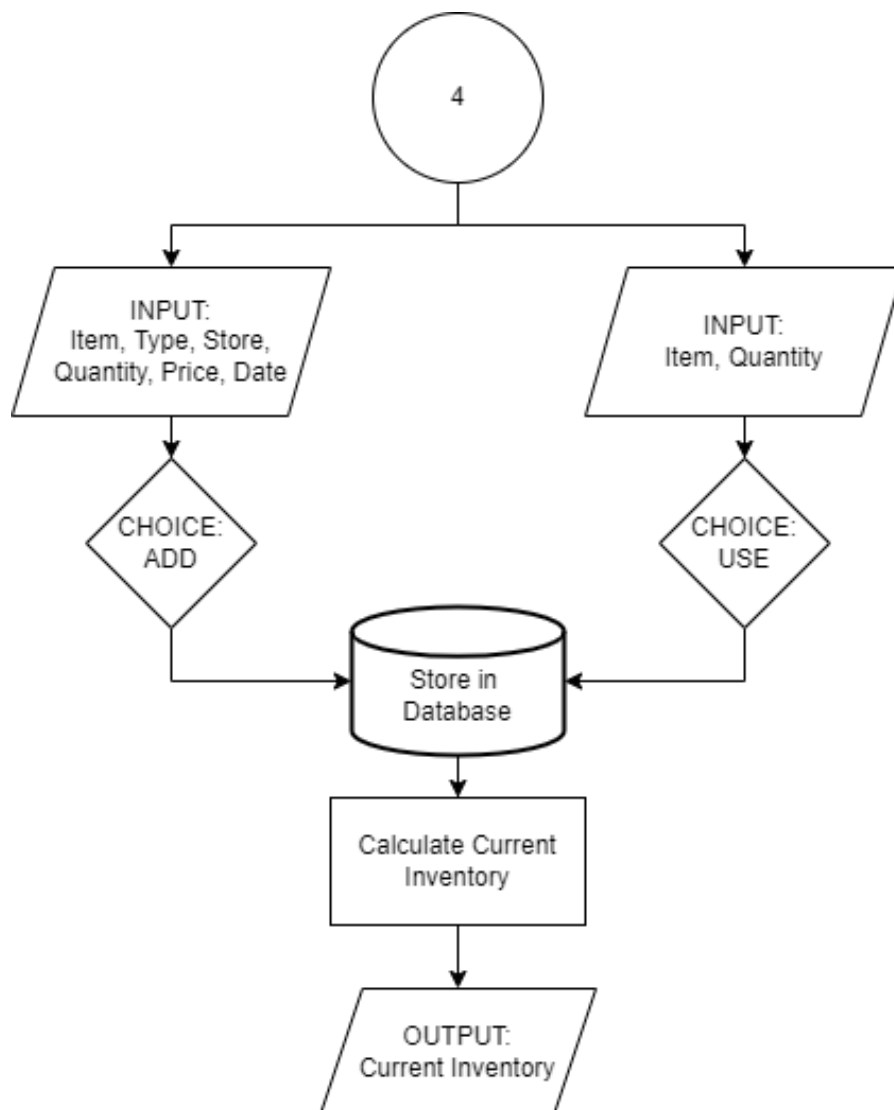
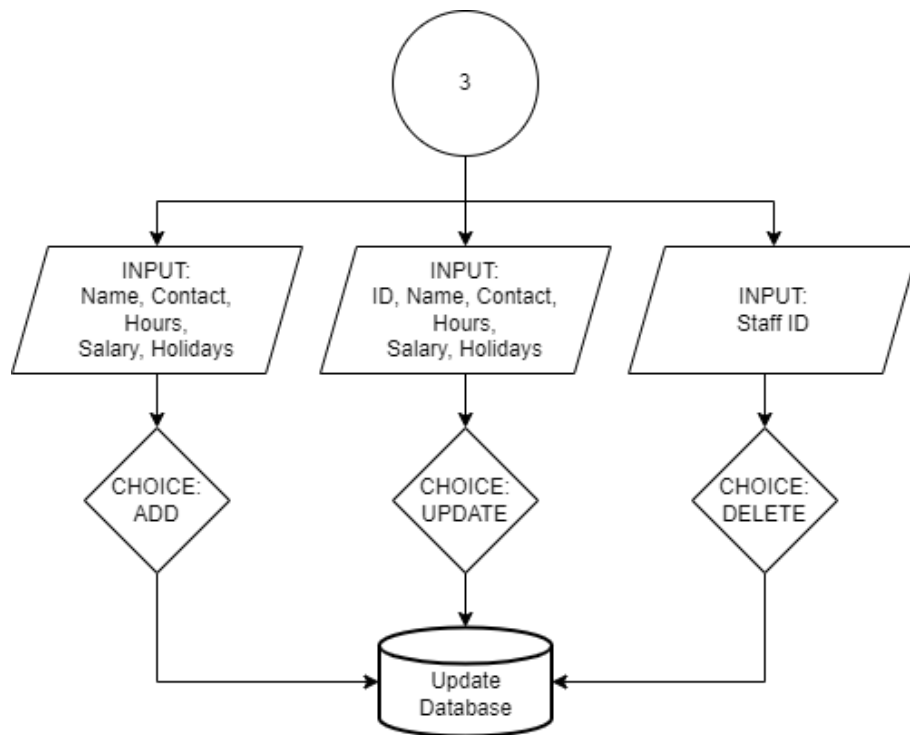
For the Admin

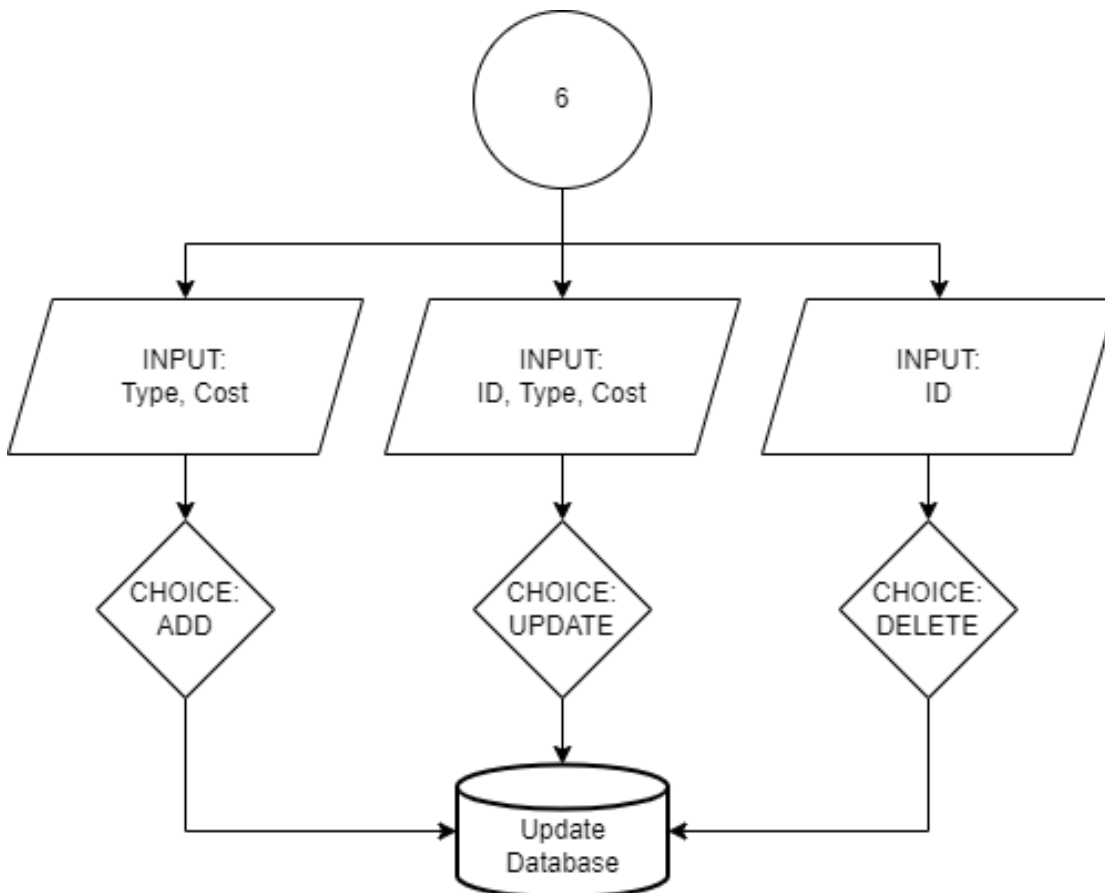
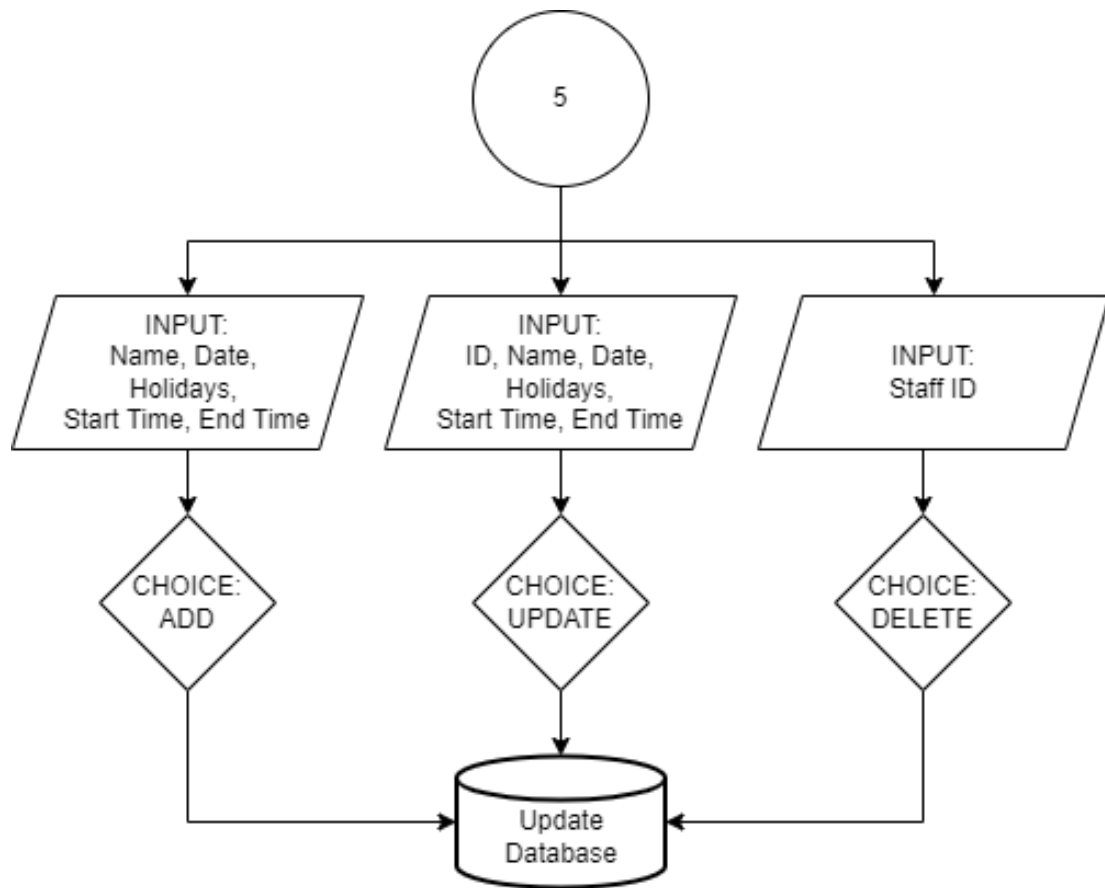


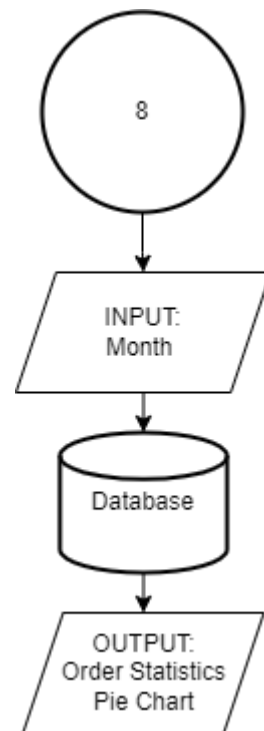
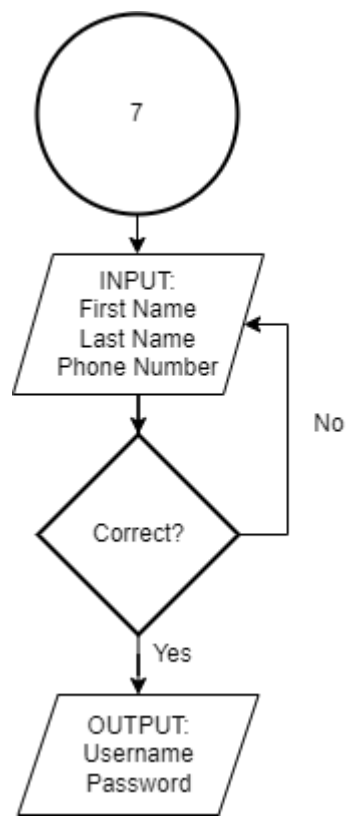
For the Staff



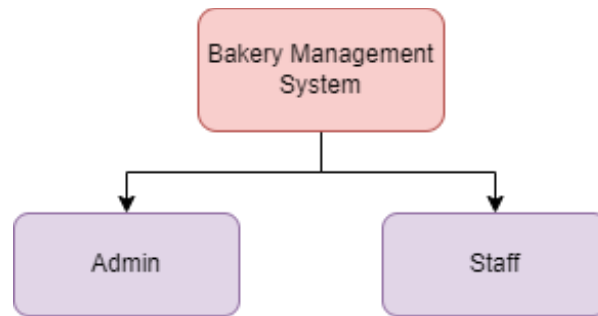




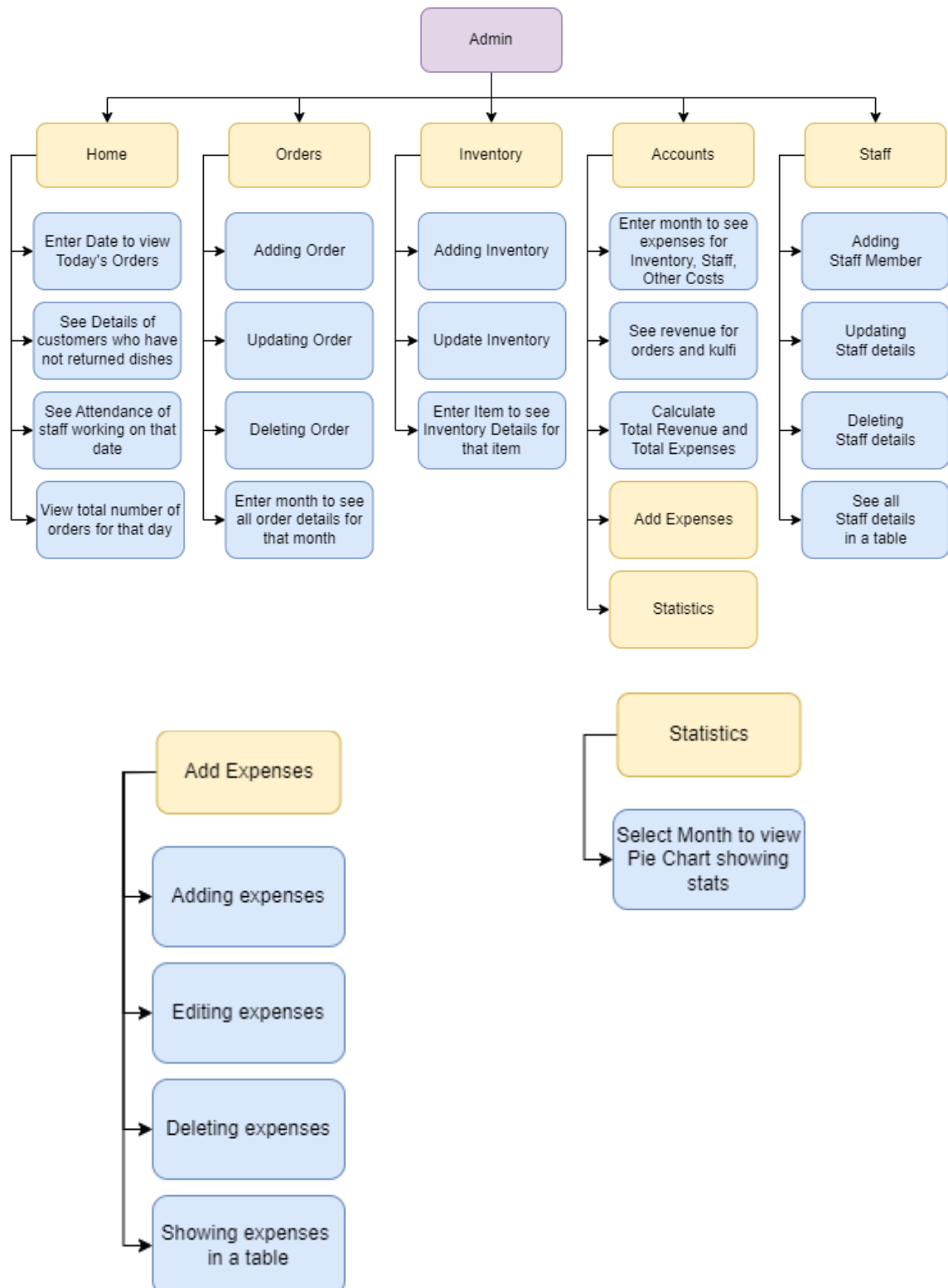




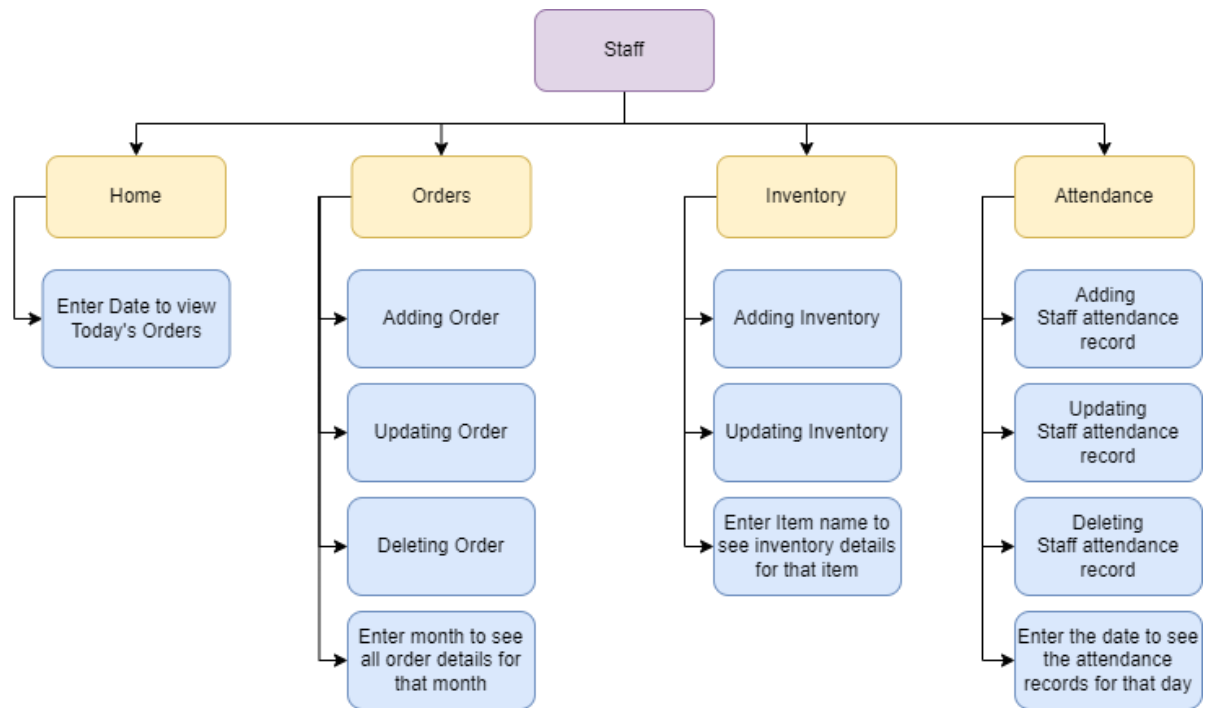
Modular Diagrams



For the Admin

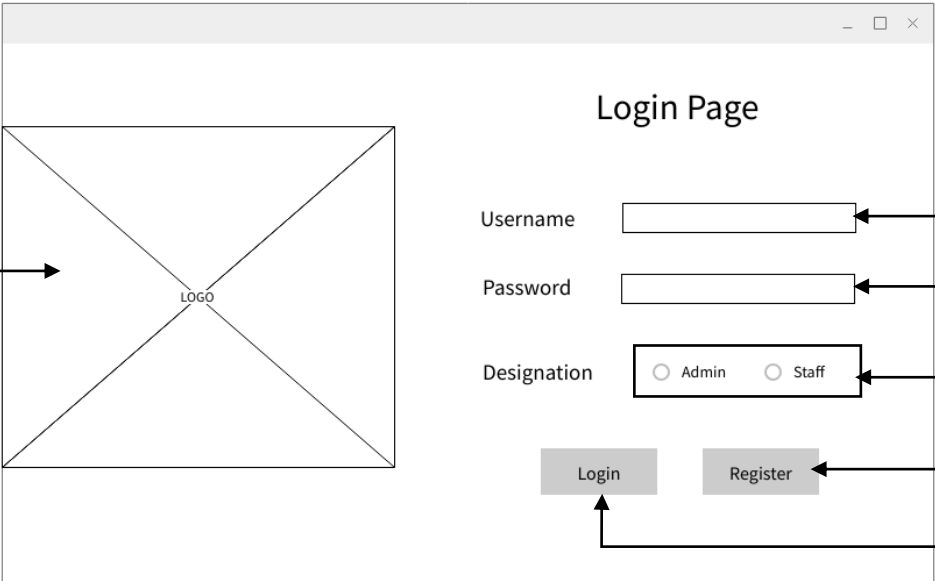


For the Staff



Screen Designs

Login Screen



The Login Page UI design features a window with a title bar and a main content area. On the left, there is a large square placeholder for a logo, labeled "LOGO" in the center. An arrow labeled "Picture" points to this placeholder. On the right, the title "Login Page" is centered. Below the title, there are three input fields: "Username", "Password", and "Designation". The "Designation" field contains two radio buttons labeled "Admin" and "Staff". Below these fields are two buttons: "Login" and "Register". Arrows point from labels to these elements: "Text Box" for Username, "Text Box" for Password, "Radio Buttons" for Designation, and "Button" for both Login and Register.

Picture →

LOGO

Username

Password

Designation ☐ Admin ☐ Staff

Login Register

Text Box

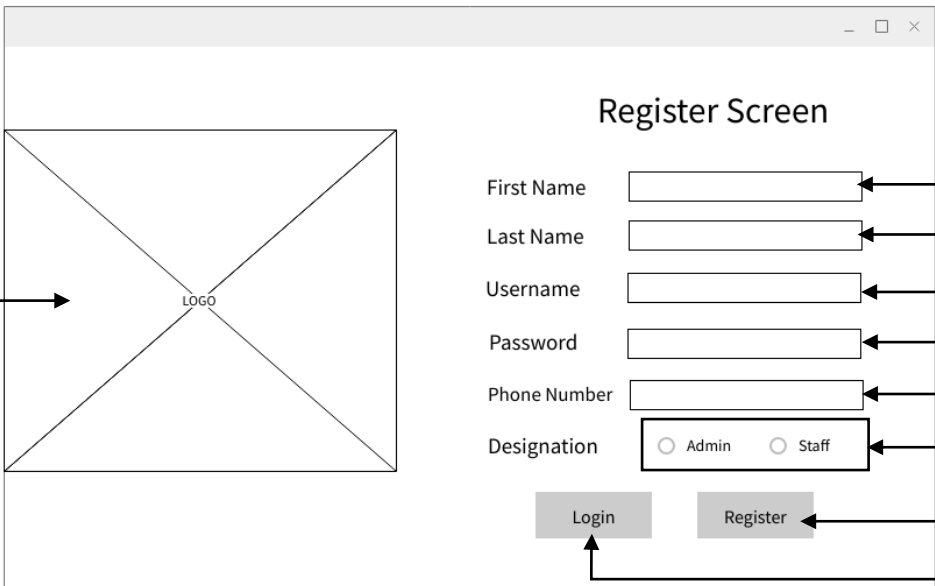
Text Box

Radio Buttons

Button

Button

Register Screen



The Register Screen UI design features a window with a title bar and a main content area. On the left, there is a large square placeholder for a logo, labeled "LOGO" in the center. An arrow labeled "Picture" points to this placeholder. On the right, the title "Register Screen" is centered. Below the title, there are five input fields: "First Name", "Last Name", "Username", "Password", and "Phone Number". The "Designation" field contains two radio buttons labeled "Admin" and "Staff". Below these fields are two buttons: "Login" and "Register". Arrows point from labels to these elements: "Text Box" for First Name, Last Name, Username, Password, and Phone Number; "Radio Buttons" for Designation; and "Button" for both Login and Register.

Picture →

LOGO

First Name

Last Name

Username

Password

Phone Number

Designation ☐ Admin ☐ Staff

Login Register

Text Box

Text Box

Text Box

Text Box

Text Box

Radio Buttons

Button

Button

Forgot Information Page

The diagram illustrates the layout of the 'Forgot Information Page'. It features a window with a title bar. On the left side of the window is a 'Back Button' with a left-pointing arrow and the text 'Back to Accounts'. On the right side is a 'Picture' of a square button with an 'X' and the text 'LOGC'. The main content area is titled 'Enter Information'. It contains three text input fields labeled 'First Name', 'Last Name', and 'Contact', each with a corresponding label to its left. Below these is a 'Text Box' label pointing to the input area. Further down are labels for 'Username' and 'Password', followed by a 'Button' labeled 'Get Details'.

Admin Screens

Admin Dashboard

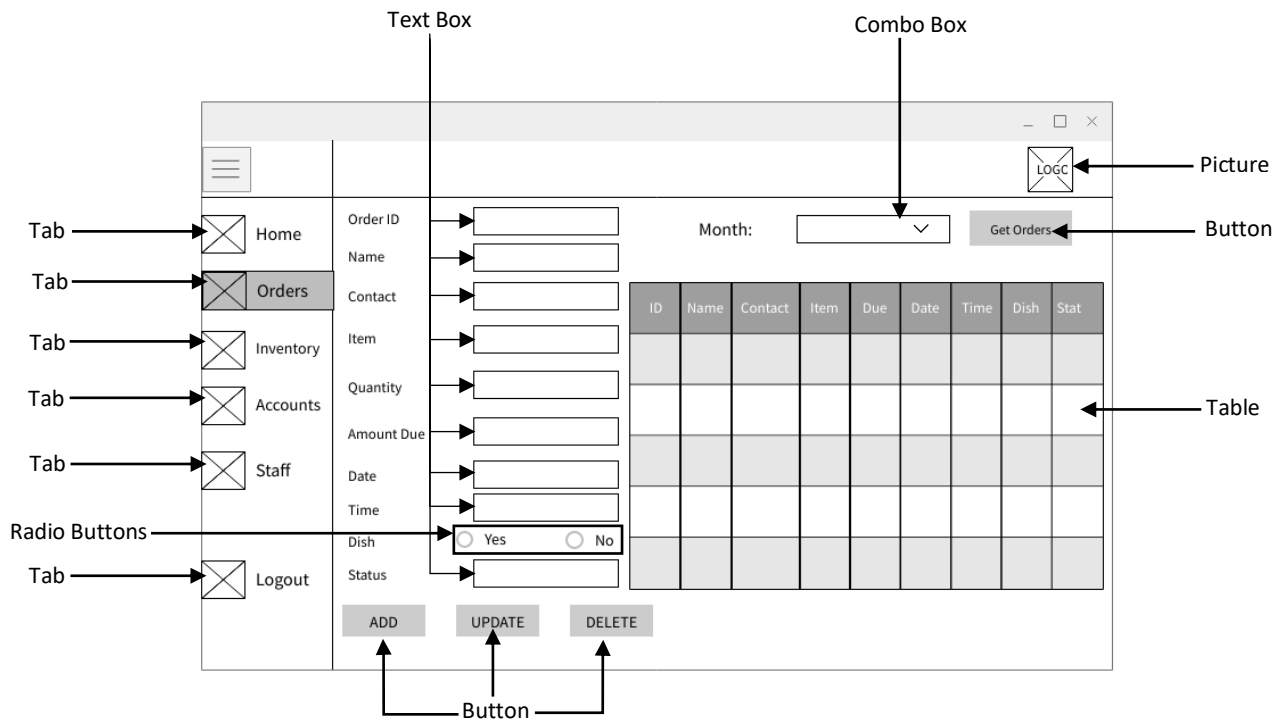
The diagram illustrates the layout of the 'Admin Dashboard'. It features a window with a title bar. On the left side is a vertical sidebar with a 'Tab' menu containing six items: 'Home', 'Orders', 'Inventory', 'Accounts', 'Staff', and 'Logout', each with a corresponding icon. The main content area is divided into several sections. At the top right is a 'Picture' of a square button with an 'X' and the text 'LOGC'. Below this is a 'Text Box' labeled 'Date:' followed by a date input field. Below the date field is a 'Button' labeled 'Get Records'. The main content area is divided into three main sections: 'Today's Orders:', 'Dishes:', and 'Attendance:'. 'Today's Orders:' contains a table with columns 'ID', 'Name', 'Item', and 'Contact'. 'Dishes:' contains a table with columns 'ID', 'Name', 'Number', and 'Contact'. 'Attendance:' contains a table with columns 'ID', 'Name', 'Start Time', and 'End Time'. A 'Panel' labeled 'Total Orders' is located to the right of the 'Today's Orders' table. A 'Table' label points to the 'Dishes' table, and another 'Table' label points to the 'Attendance' table.

ID	Name	Item	Contact

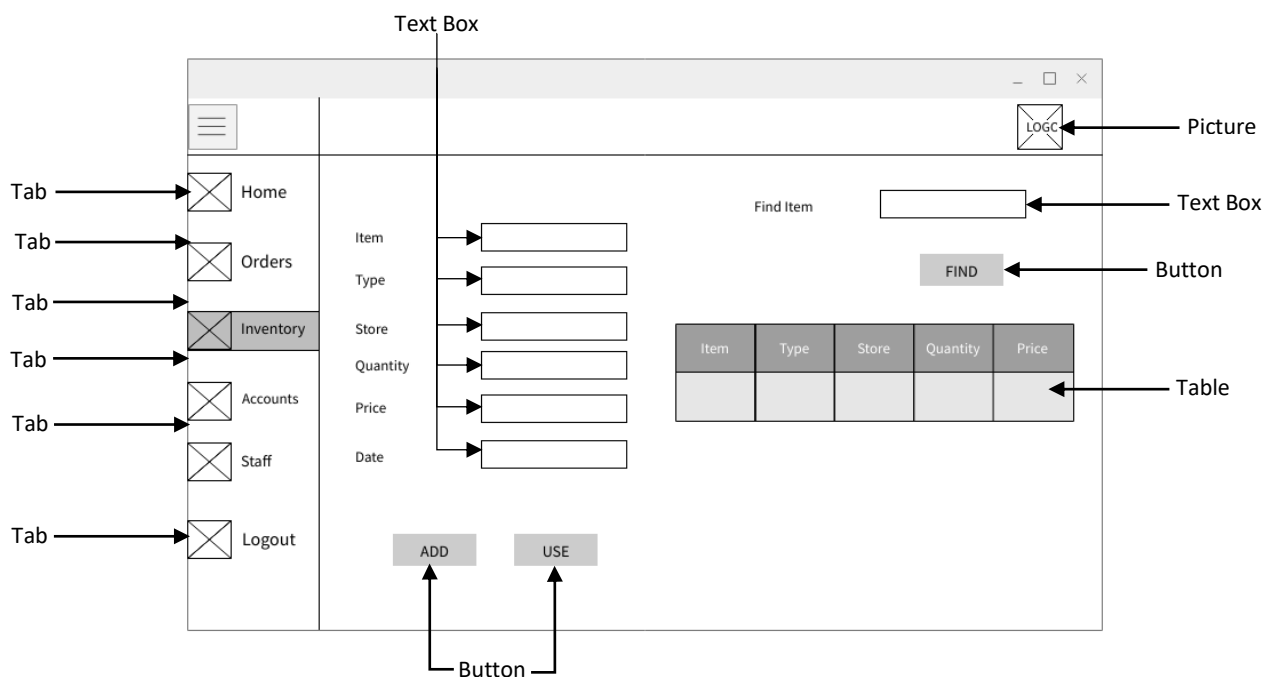
ID	Name	Number	Contact

ID	Name	Start Time	End Time

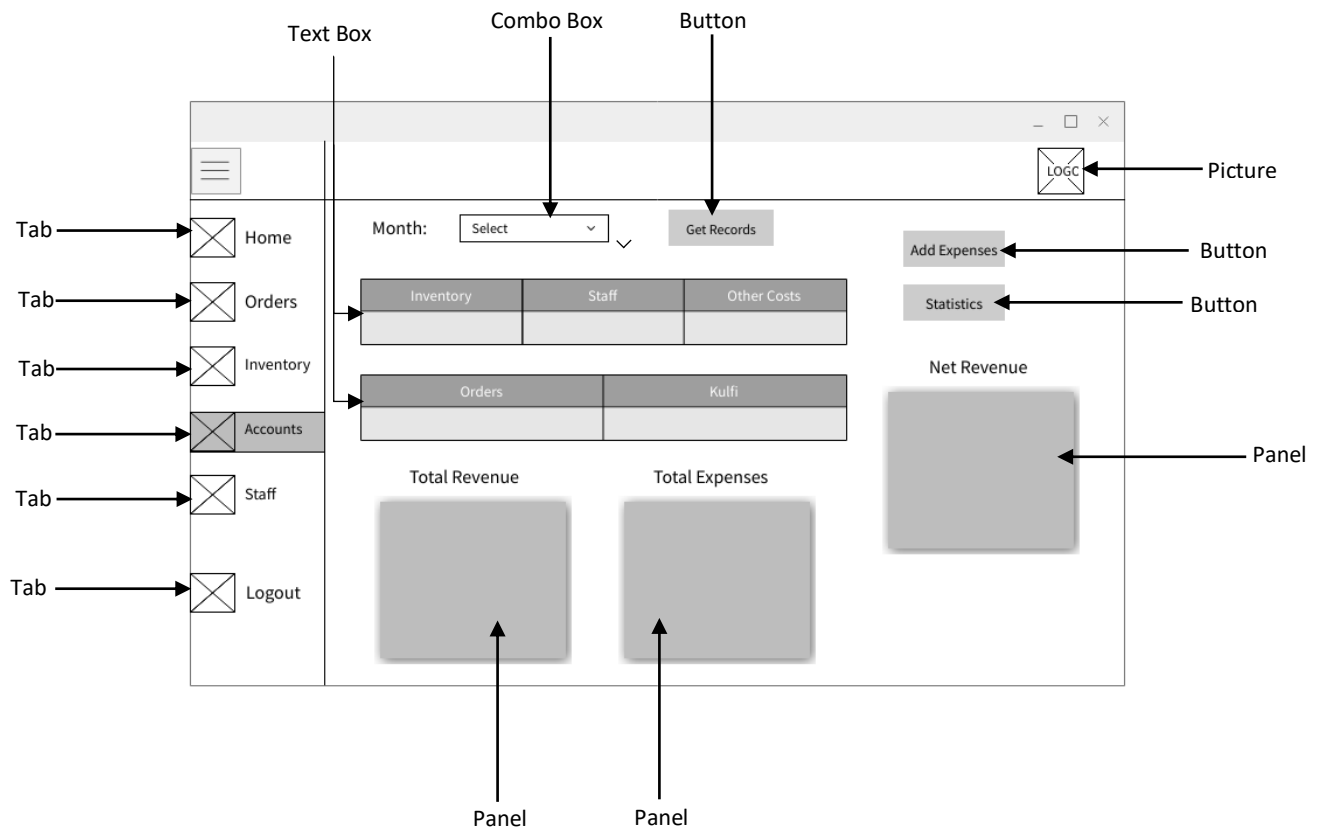
Admin Orders



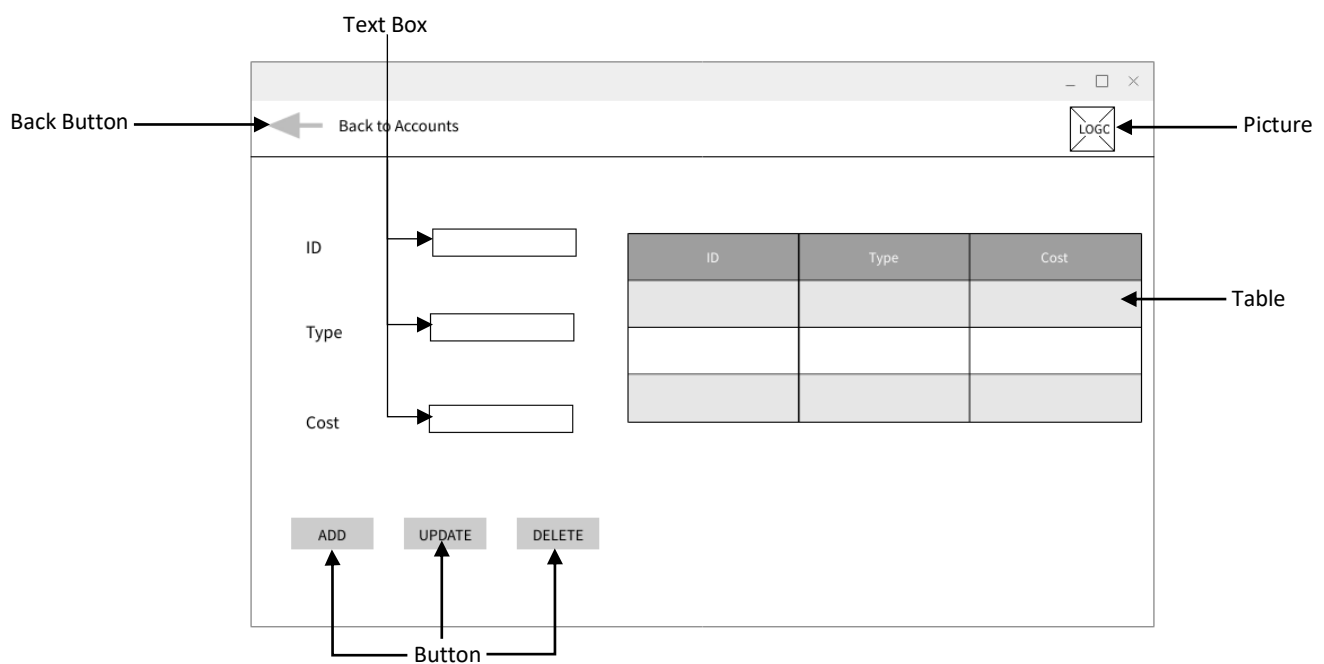
Admin Inventory



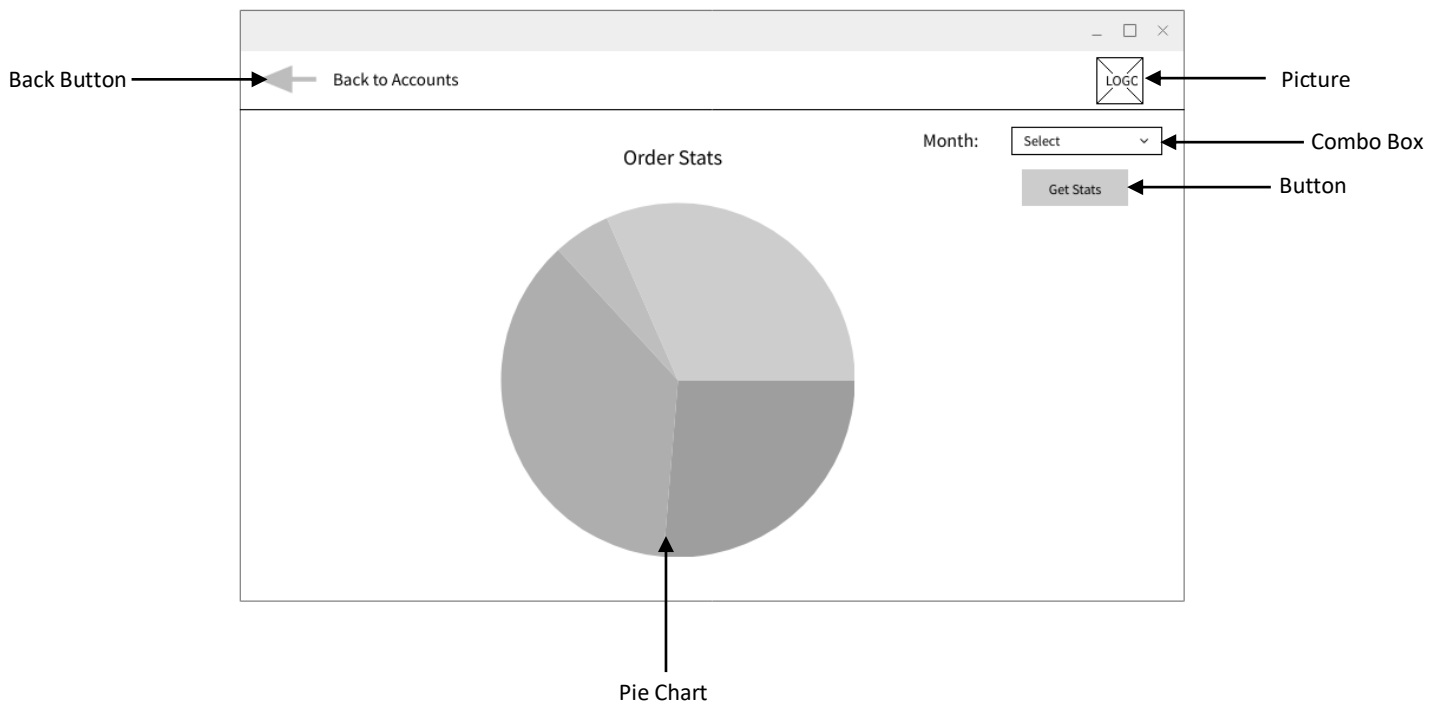
Admin Accounts



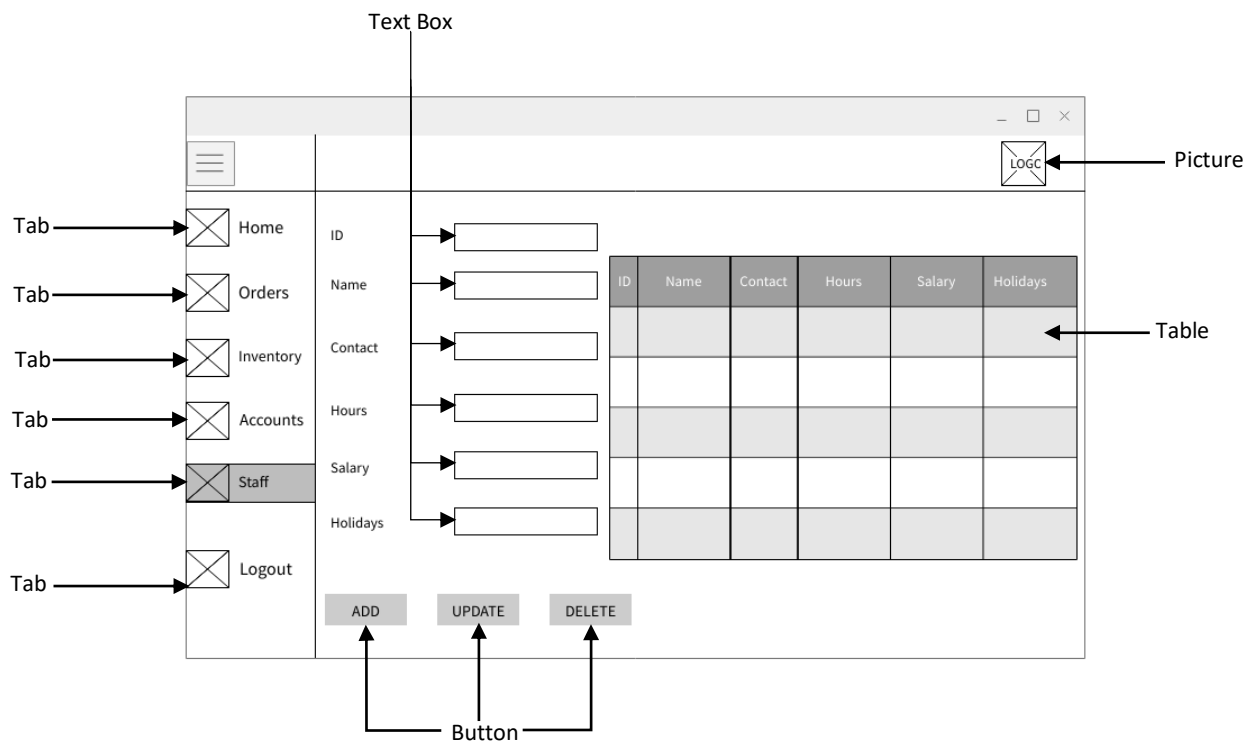
Admin Other Costs



Accounts Stats

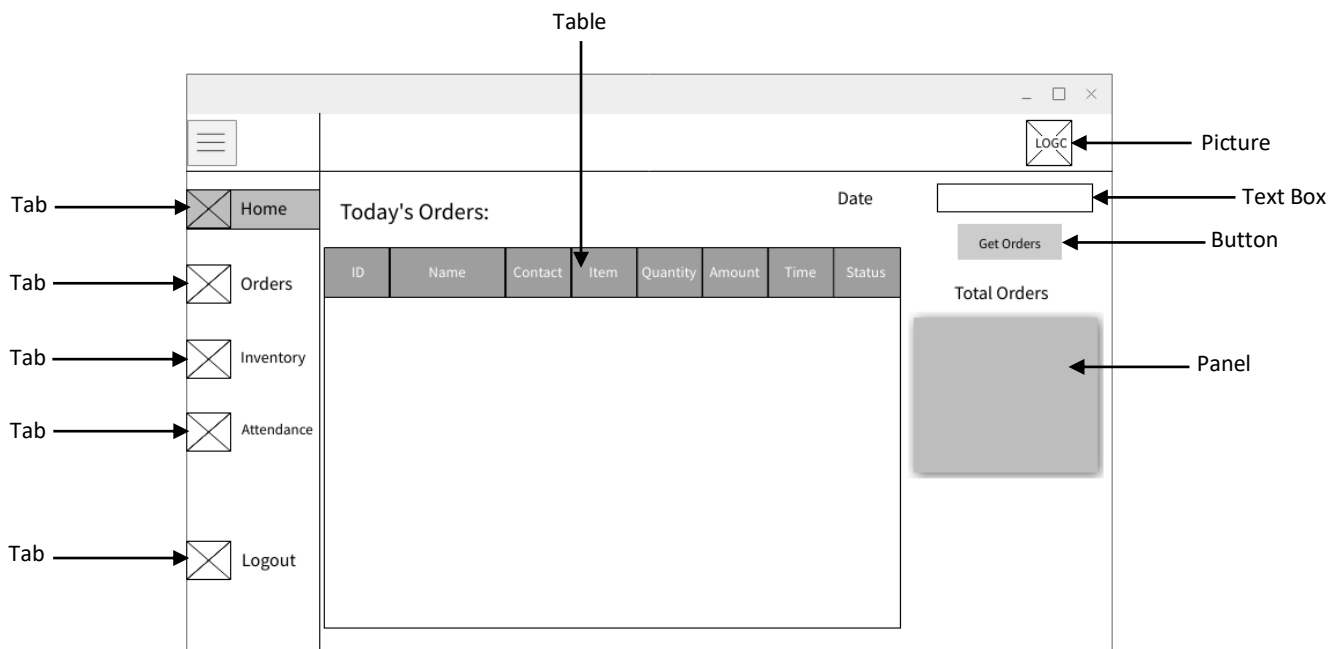


Admin Staff

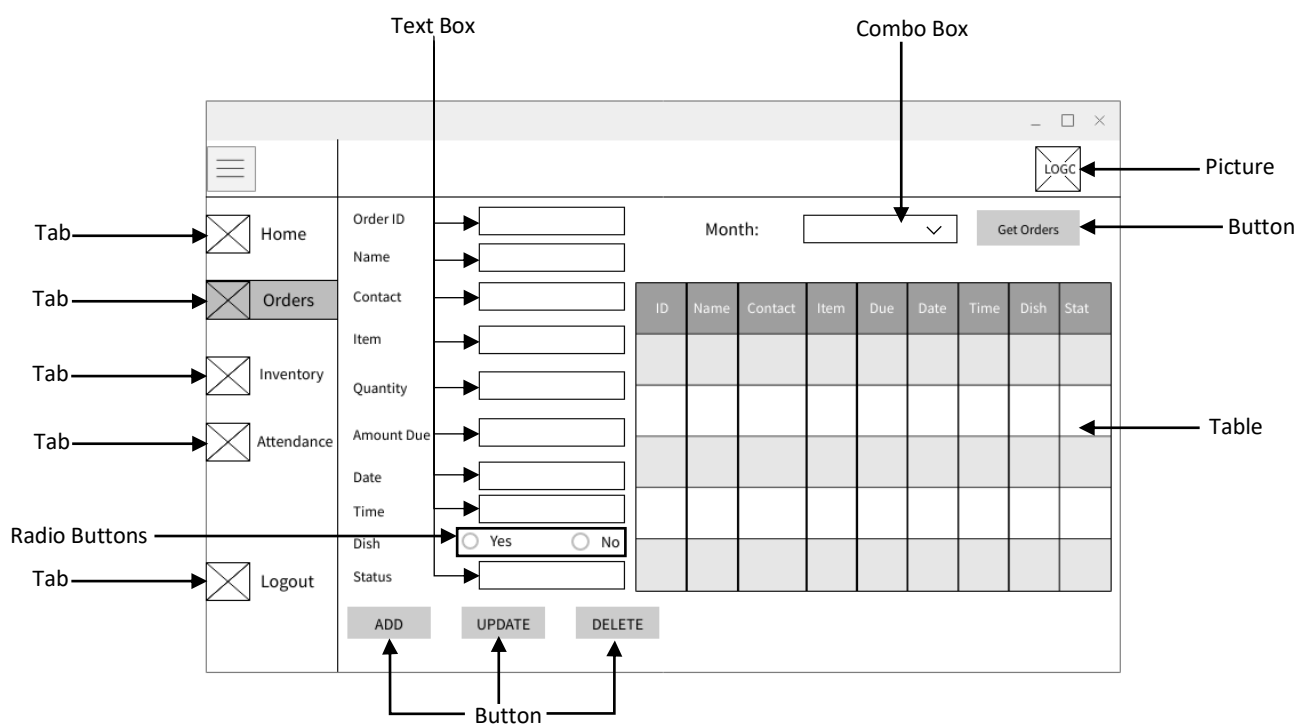


Staff Screens

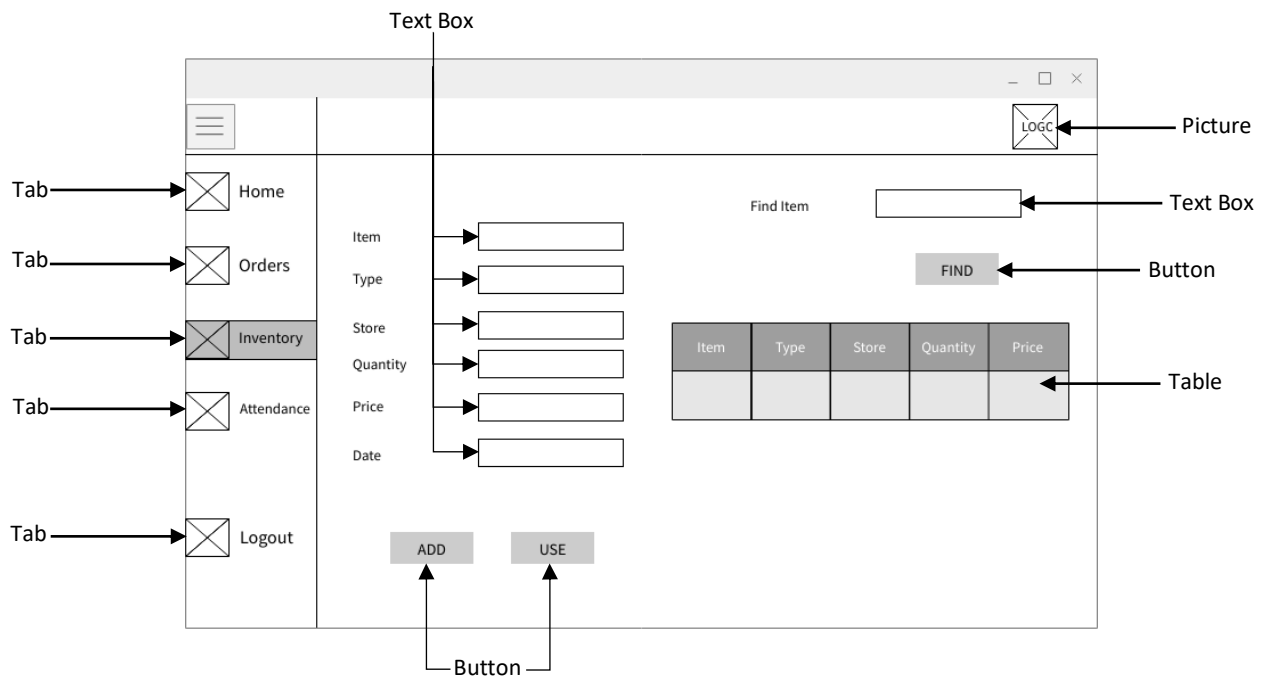
Staff Dashboard



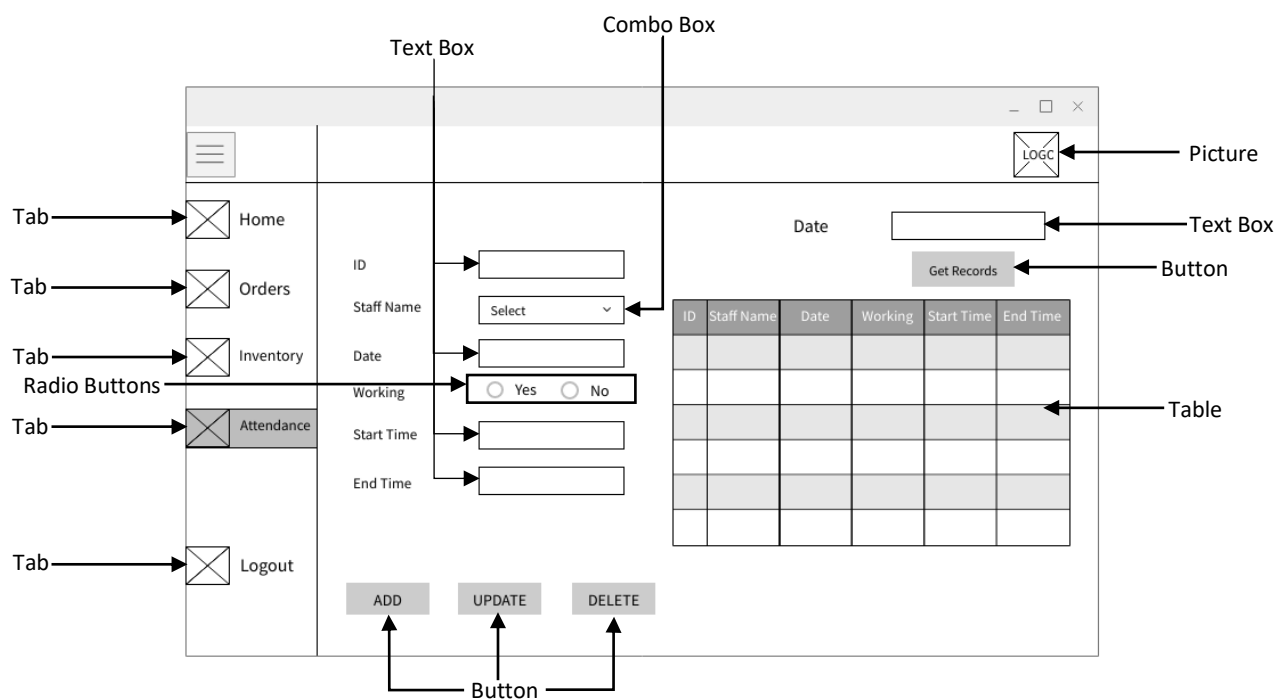
Staff Orders



Staff Inventory



Staff Attendance



Database

users table

Field Name	Data Type	Description	Other Information
id	int(11)	Unique ID for each user	Primary Key, Not Null
first_name	varchar(50)	First Name of the User	Not Null
last_name	varchar(50)	First Name of the User	Not Null
username	varchar(50)	Name used for login	Not Null
password	varchar(50)	Password for login	Not Null
phone_number	varchar(10)	Contact Information of the user	Not Null
designation	varchar(15)	Position of the user(Admin/Staff)	Not Null

order_details table

Field Name	Data Type	Description	Other Information
Order_ID	int(11)	Unique ID for each order	Primary Key, Not Null
Customer_Name	varchar(100)	Name of the customer	Not Null
Customer_Contact	varchar(10)	Contact Information for the customer	Not Null
Item	varchar(50)	The item that was ordered	Not Null
Quantity	int(11)	The quantity that was ordered	Not Null
Amount_Due	float	Total amount due for the order	Not Null
Delivery_Date	varchar(15)	Date the order is due	Not Null
Delivery_Time	varchar(15)	Time the order is due	Not Null
Dish	varchar(11)	Whether the order is in a dish that is to be returned	Not Null
Stat	varchar(50)	Status of the order	Not Null
Month	varchar(50)	Month the order is due	Not Null, Not shown in the program

inventory table

Field Name	Data Type	Description	Other Information
Item_ID	int(11)	Unique ID for each item	Primary Key, Not Null
Item	varchar(250)	Name of the Item	Not Null
Type	varchar(100)	Ingredient/Material	Not Null
Store	varchar(200)	The store the item was bought from	Not Null
Quantity	int(11)	The quantity that was bought	Not Null
Price	float	Price per unit for the item	Not Null
Date	varchar(50)	Date the item was bought	Not Null
Month	varchar	Month the item was purchased in	Not Null, Not shown in the program

used_inventory

Field Name	Data Type	Description	Other Information
ID	int	Unique ID for each item	Primary Key, Not Null
Item	varchar	Name of the item	Not Null
Quantity	int	Quantity of the item bought	Not Null

staff table

Field Name	Data Type	Description	Other Information
Staff_ID	int(11)	Unique ID for each staff member	Primary Key, Not Null
Staff_Name	varchar(100)	Name of the staff	Not Null
Contact	varchar(10)	Contact Information for the staff	Not Null
Hours	int(11)	Hours worked by the staff	Not Null
Salary	float	Salary of the staff	Not Null
Holidays	int(11)	Number of Holidays remaining	Not Null

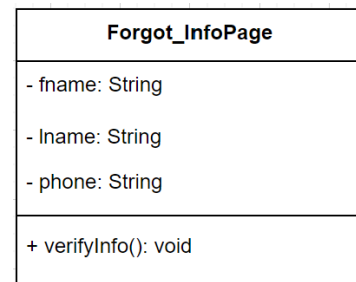
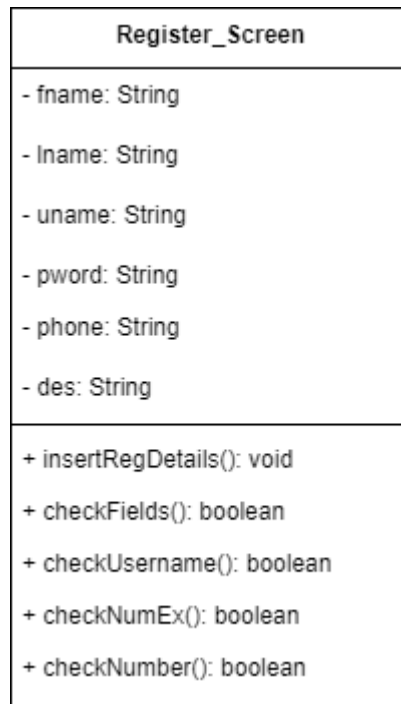
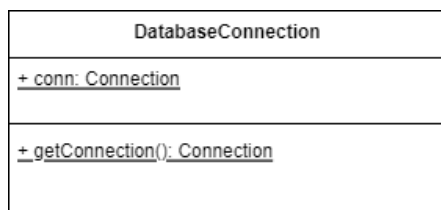
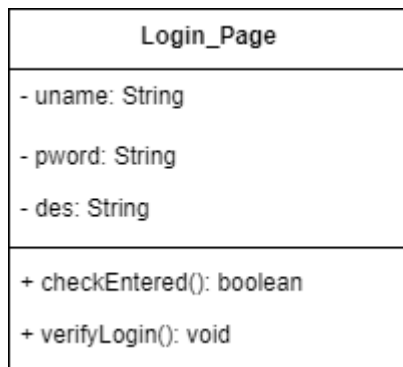
attendance table

Field Name	Data Type	Description	Other Information
ID	int(11)	Unique ID for each staff member	Primary Key, Not Null
Name	varchar(50)	Name of the staff	Not Null
Date	varchar(25)	Date	Not Null
Working	varchar(10)	Whether the staff is working	Not Null, Yes/No
Start_Time	varchar(25)	Time this staff member started working	Not Null
End_Time	varchar(25)	Time this staff member stopped working	Not Null

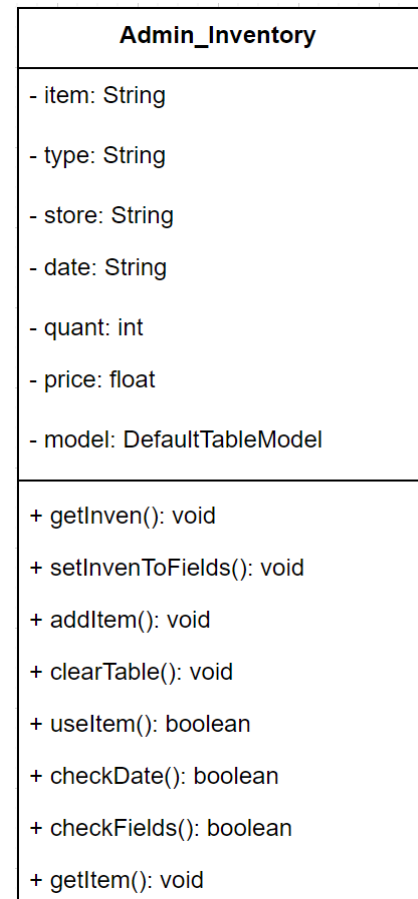
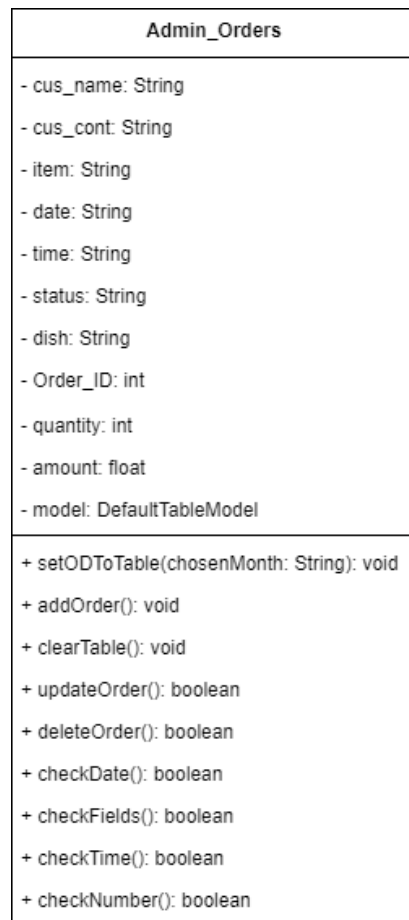
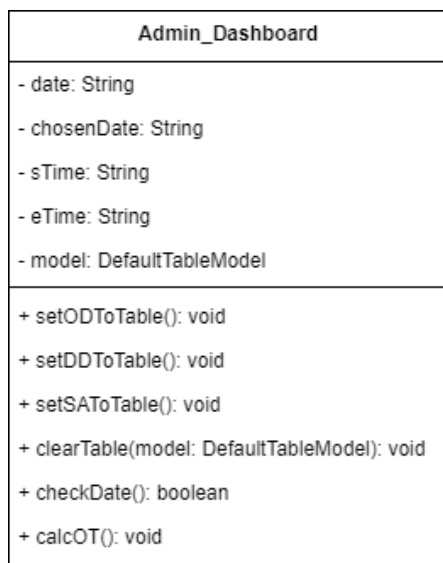
costs table

Field Name	Data Type	Description	Other Information
ID	int(11)	Unique ID for each cost	Primary Key, Not Null
Type	varchar(100)	What the cost is	Not Null
Cost	float	The amount for this cost	Not Null

UML Diagrams



Admin



Admin_Accounts
<ul style="list-style-type: none"> - totalOC: float - totalInven: float - totalStaffSal: float - totalExpen: String - totalExpenFlo: float - totalORev: float - totalKulRev: float - totalReven: String - totalRevenFlo: float - netRev: String - netRevFlo: float - model: DefaultTableModel
<ul style="list-style-type: none"> + setTotExp(): void + setTotRev(): void + clearTable(model: DefaultTableModel): void + calcTotalOC(): void + calcTotalInven(): void + calcTotalStaSal(): void + calcTotalORev(): void + calcTotalKulRev(): void

Accounts_OtherCosts
<ul style="list-style-type: none"> - type: String - id: int - cost: float - model: DefaultTableModel
<ul style="list-style-type: none"> + setExpToTable(): void + addExpense(): void + clearTable(): void + updateExpense(): boolean + deleteExpense(): boolean + checkFields(): boolean

Accounts_Stats
<ul style="list-style-type: none"> - chosenMonth: Strin
<ul style="list-style-type: none"> + setPieChart(): void

Admin_Staff
<ul style="list-style-type: none"> - name: String - cont: String - id: int - hours: int - holis: int - salary: float - model: DefaultTableModel
<ul style="list-style-type: none"> + setStaffToTable(): void + addStaff(): void + clearTable(): void + updateStaff(): boolean + deleteStaff(): boolean + checkNumber(): boolean + checkFields(): boolean

Staff

StaffDash
- date: String - model: DefaultTableModel
+ setODToTable(): void + clearTable(): void + checkDate(date: String): boolean

StaffOrders
- cus_name: String - cus_cont: String - item: String - date: String - time: String - status: String - dish: String - Order_ID: int - quantity: int - amount: float - model: DefaultTableModel
+ setODToTable(chosenMonth: String): void + addOrder(): void + clearTable(): void + updateOrder(): boolean + deleteOrder(): boolean + checkDate(): boolean + checkTime(): boolean + checkNumber(): boolean + checkFields(): boolean

StaffInventory
- item: String - type: String - store: String - date: String - quant: int - price: float - model: DefaultTableModel
+ getInven(): void + setInvenToFields(): void + addItem(): void + clearTable(): void + useItem(): boolean + checkDate(): boolean + checkFields(): boolean + getItem(): void

StaffAttendance
- name: String - date: String - dateE: String - sTime: String - eTime: String - work: String - id: int - model: DefaultTableModel
+ setSAtToTable(): void + addAtten(): void + clearTable(): void + updateAtten(): void + deleteAtten(): void + checkDate(date: String): boolean + checkTime(time: String): boolean + checkFields(): boolean + getStaNames(): void

Validation

Field Name	Data Validation Type	Explanation
Username	<ul style="list-style-type: none"> Uniqueness Check 	Doesn't exist in database
Phone Number	<ul style="list-style-type: none"> Length check Uniqueness check Type check 	Must be exactly 10 characters long; must be integers; must not already exist in the database
Date	<ul style="list-style-type: none"> Form check Type check Range check 	Must be in DD/MM/YY form; must be numbers or '/'; dd must be <= 31, mm must be <= 12
Time	<ul style="list-style-type: none"> Form check Range check Type check 	must be in HH:MM am/pm form; HH must be <= 12, MM must be <= 59
All Fields	<ul style="list-style-type: none"> Presence Check 	All fields must contain valid information

Test Plan

Field	Test Data	Type	Explanation
Phone Number	668190826	Abnormal	Must exactly 10 characters
Phone Number	88209871678	Abnormal	must be exactly 10 characters
Phone Number	88920098j7	Abnormal	must be only integers
Phone Number	7075879337	Normal	10 characters; only integers; should be accepted
Date	45/10/22	Abnormal	DD must be <= 31
Date	12/98/22	Abnormal	MM must be <= 12
Date	31/12/22	Extreme	should be accepted
Date	4/9/22	Abnormal	must have 8 characters in the form DD/MM/YY
Date	04/09:22	Abnormal	must be 8 characters; must split into 3 by '/'
Date	04/09/22	Normal	within range; 8 characters; in DD/MM/YY; should be accepted
Time	32:12 am	Abnormal	HH must be <= 12
Time	09:89 pm	Abnormal	MM must be <= 59
Time	08:45 on	Abnormal	must be either am or pm
Time	12:34	Abnormal	must have am or pm
Time	12:59 pm	Extreme	should be accepted
Time	04:37 am	Normal	should be accepted
Time	10:14 pm	Normal	should be accepted
All fields	" "	Abnormal	all information must be entered

Testing Functionality of database:

1. Register using new username and phone number then try reregistering using same details (it should check database and not permit registering using same/preexisting details)
2. Login using credentials (checks database to match username to password and designation; enter correct and see if allowed to login then enter incorrect and see if not allowed to login)
3. Enter date in dashboard (checks database to give all details for that specific date only)
4. Add order/attendance/staff/inventory/expenses (adds in database and then shows in table connected to database)
5. Updates order/attendance/staff/expenses (updates in database then shows in connected table)
6. Use item (adds in used inventory table in database and removes same amount from table connected to database)
7. Delete order/staff/attendance (removes from database then shows connected table without entry)
8. Enter month to get orders and accounts for that month (checks database then displays results for only those months in tables connected to database)
9. Enter item name to check stock for that item (checks database then displays stock for that item)