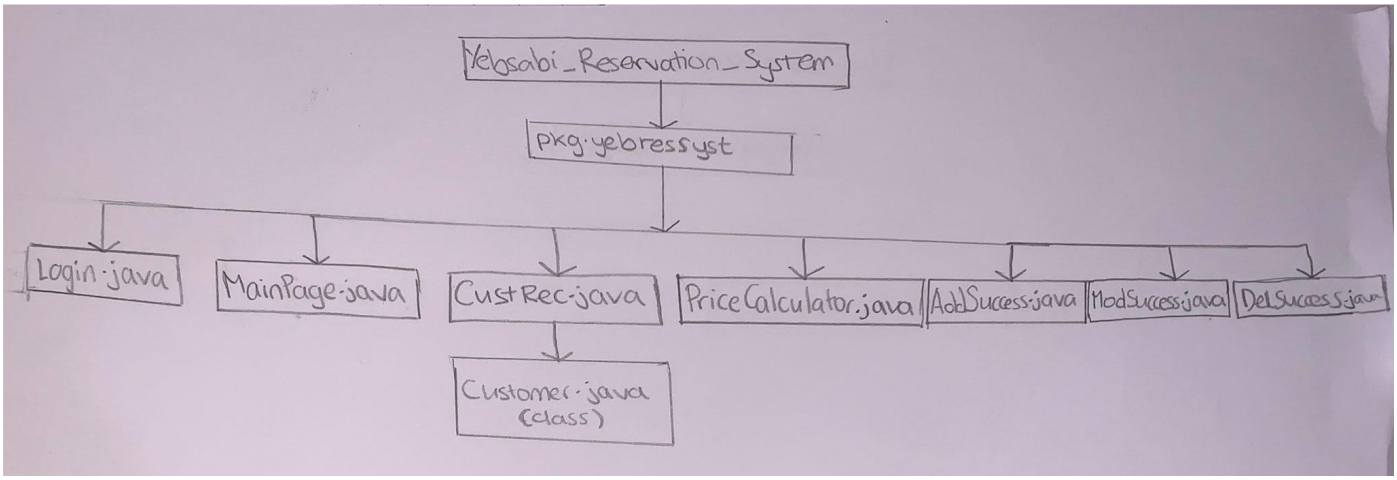
Criterion B: Design

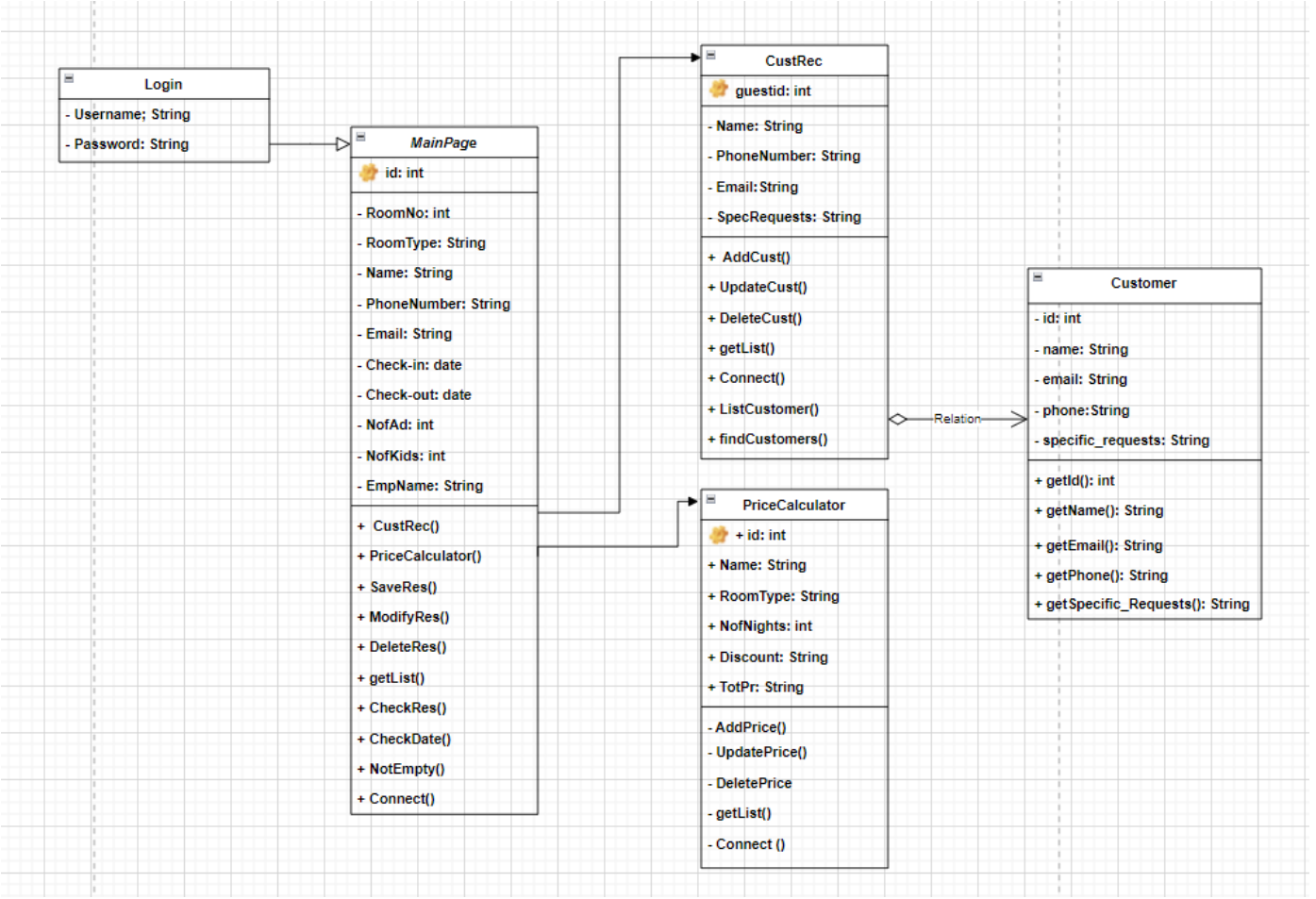
Structure:



Within the 'Yebsabi Reservation System' project, there'll be one package named 'pkg.yebsabireservations'. Within this package, there will be seven forms, within which there will be 7 classes.
Login.java main class- Responsible for ensuring that the user is an employee. The MainPage.java main class-, where the employee can make, modify, and delete a reservation.
CustRec.java main class- is responsible for saving guest information.
Customer.java main class- ithin the CustRec.java form, there is a class which is necessary for the search button.
The 'Price Calculator' form which is responsible for calculating the price for a guest.

Ruth Bekalu Criterion B

UML Diagrams for Classes mentioned above

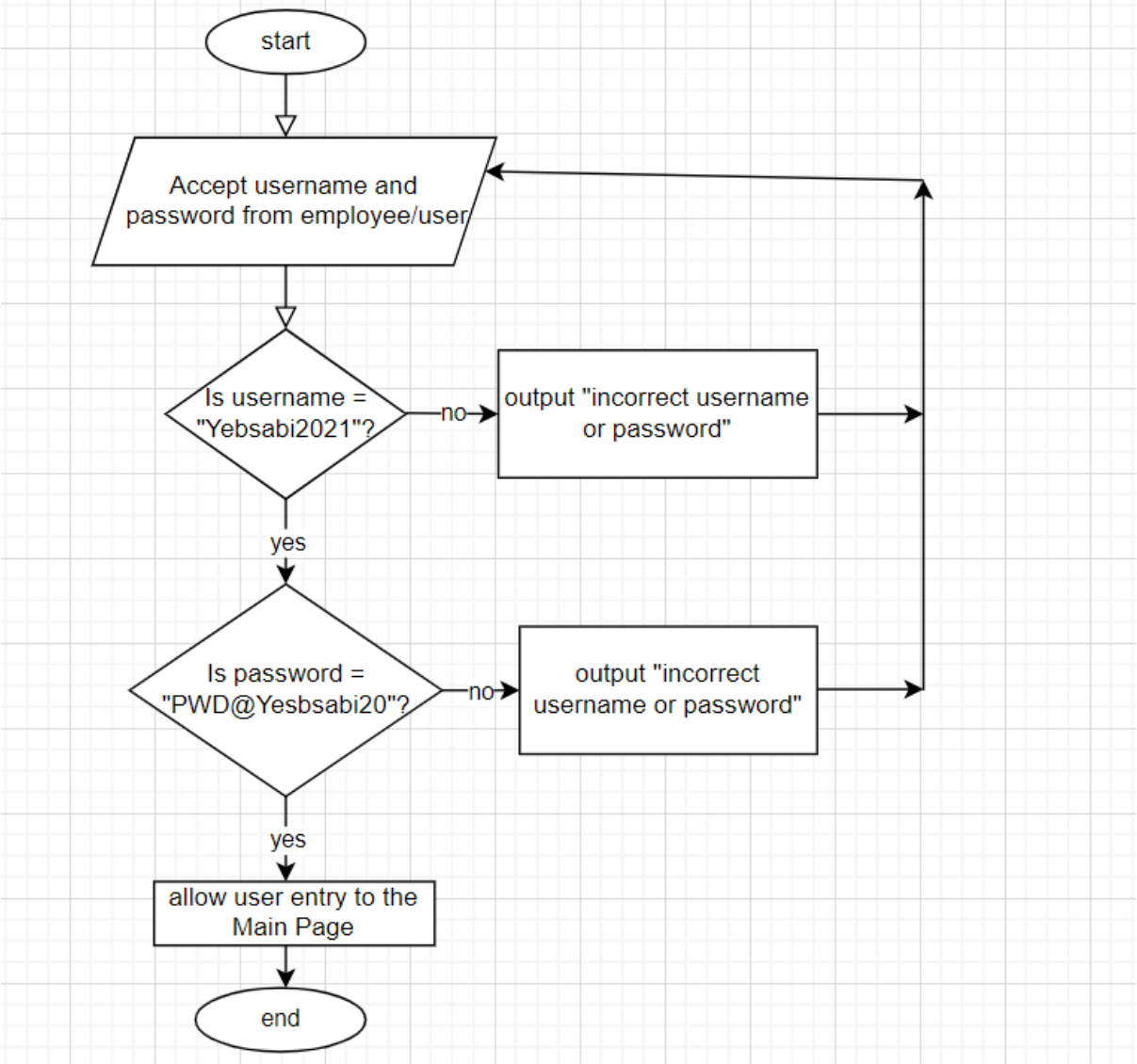


Ruth Bekalu Criterion B

Flowcharts

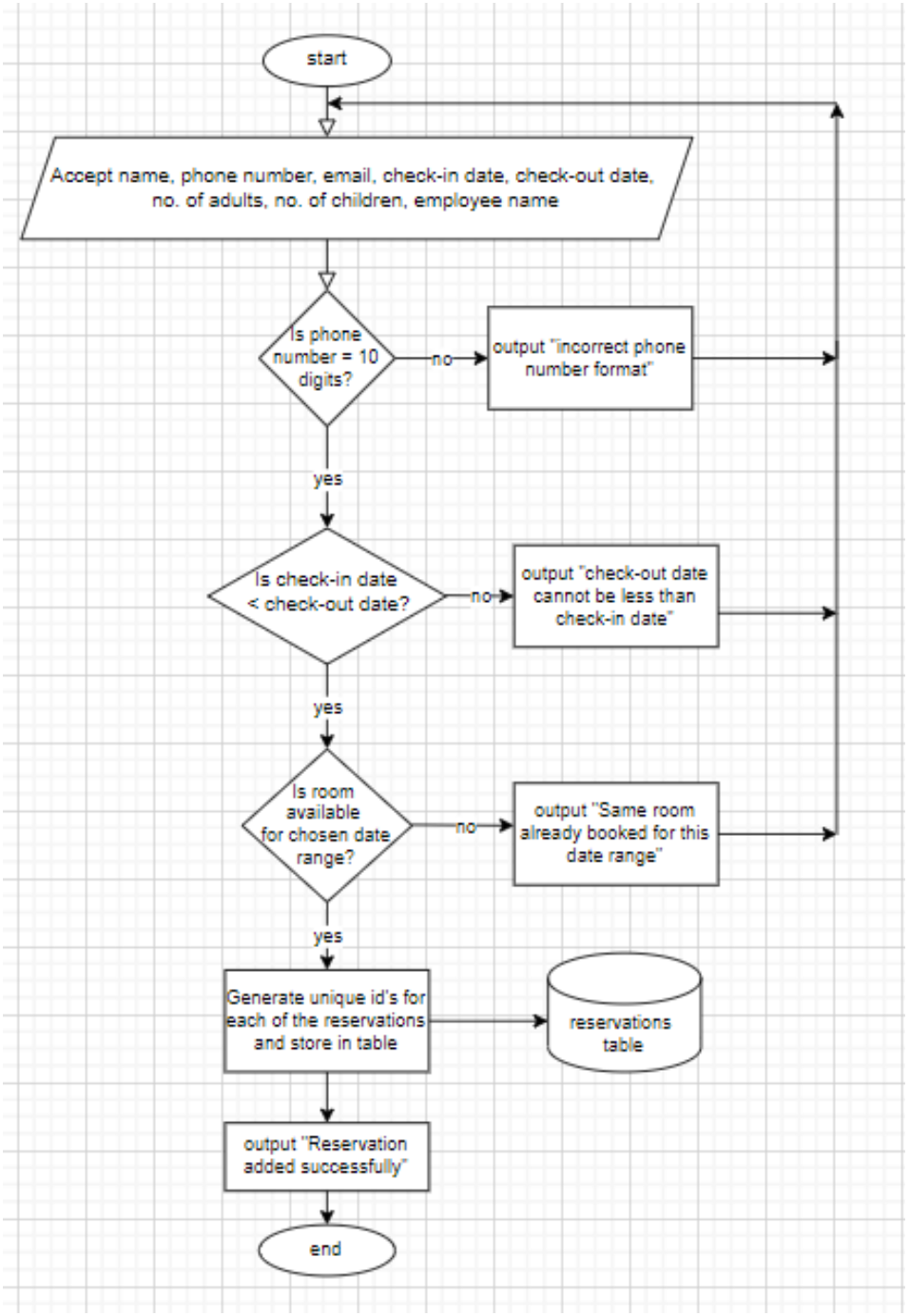
Main Page Flowcharts:

Flowchart number 1 (Login): This flowchart shows the processes behind the login page. The employee would input the username and password and, if correct, it will redirect the user to the main page. If not, an error message will be displayed.

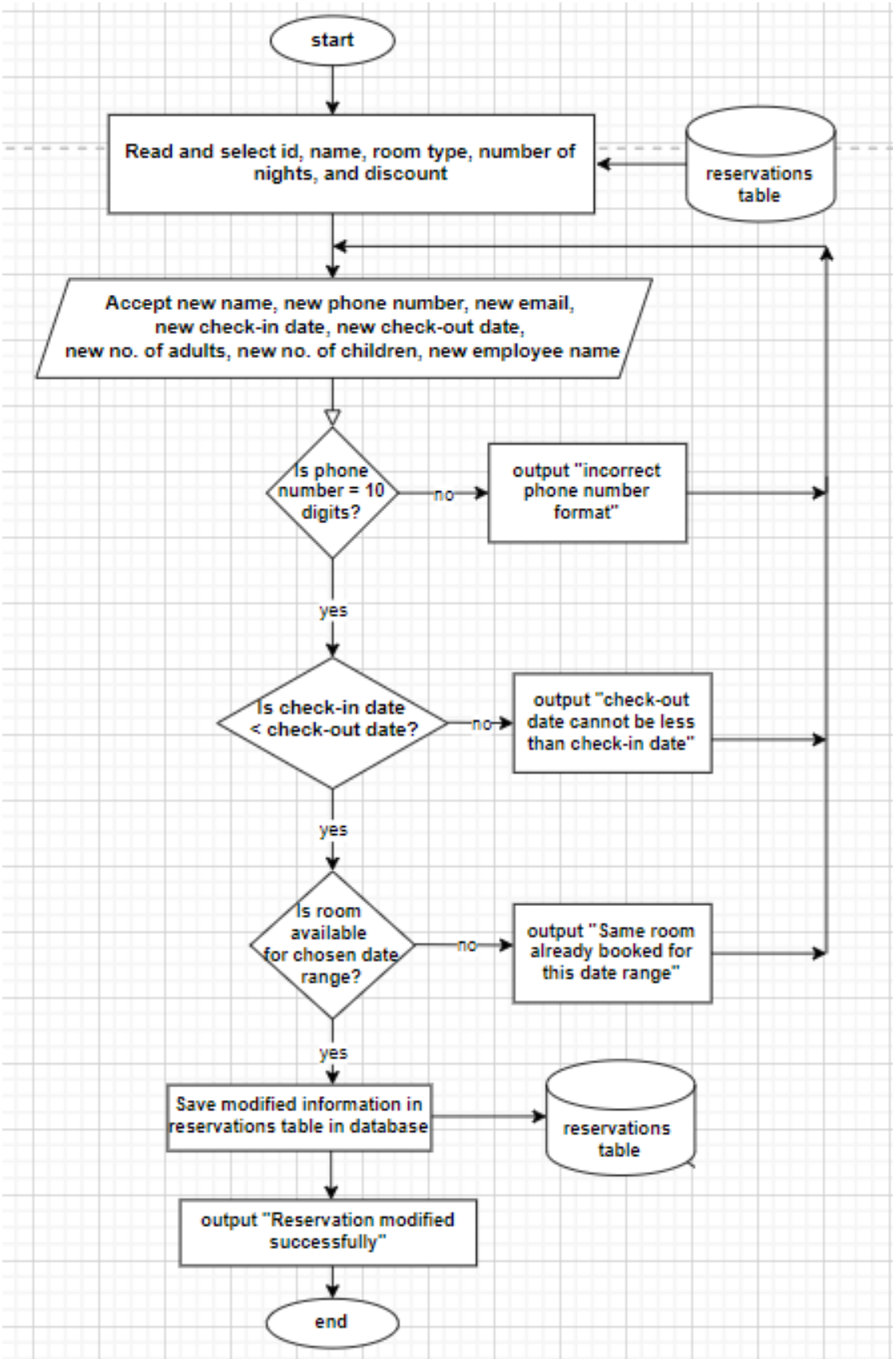


Ruth Bekalu Criterion B

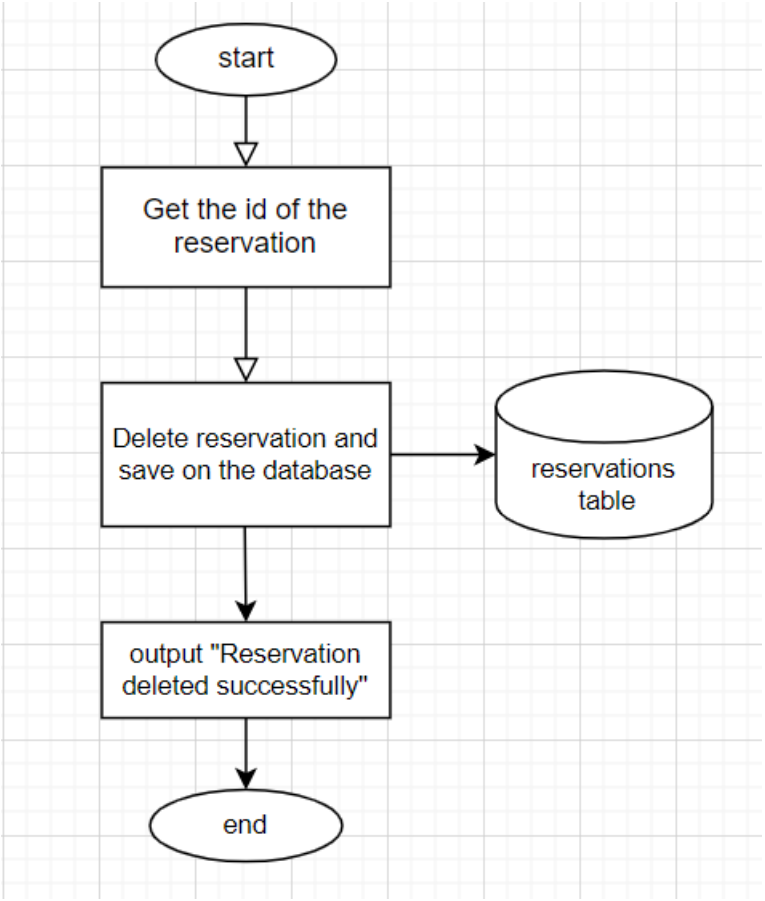
Flowchart number 2 (New Reservation): The method chosen for this reservation system is inputting all necessary guest information and saving as a new reservation on a table in the database. As shown by Flowchart number one, the employee will input all of the customer's information and reserve a room. If phone number format or order of dates are inputted incorrectly, an error message will be displayed. Furthermore, if the same room is booked for the same date range, an error message will be displayed.



Flowchart number 3 (Modify Reservation): This flowchart is a similar one to Flowchart number 21. But, this flowchart outlines how the program modifies a reservation. It shows the steps taken by the employee when modifying a reservation.

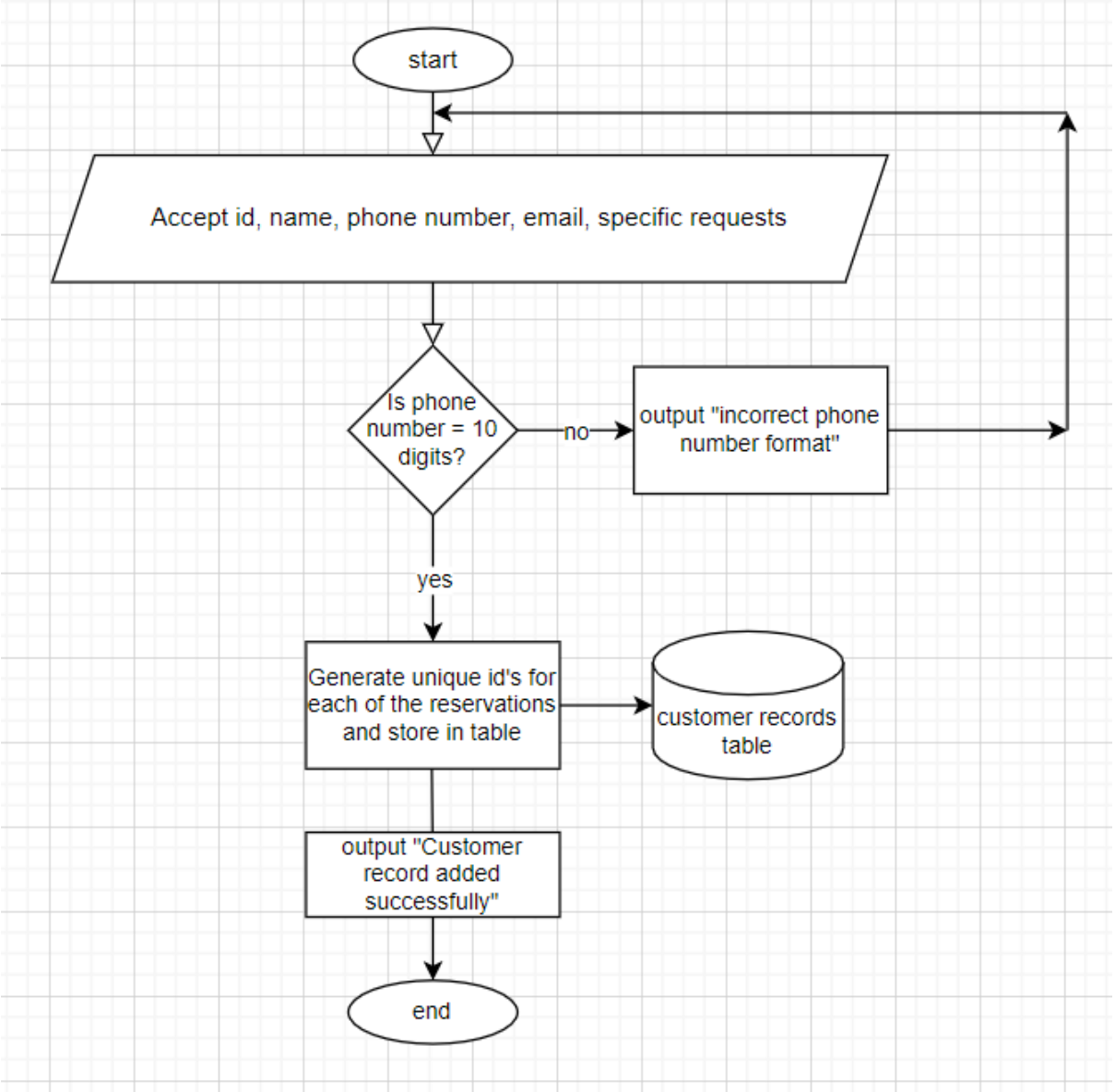


Flowchart number 4 (Cancel Reservation): This flowchart is a similar one to Flowchart number 1. But, this flowchart outlines how the program cancels a reservation. It shows the steps taken by the employee when cancelling a reservation.

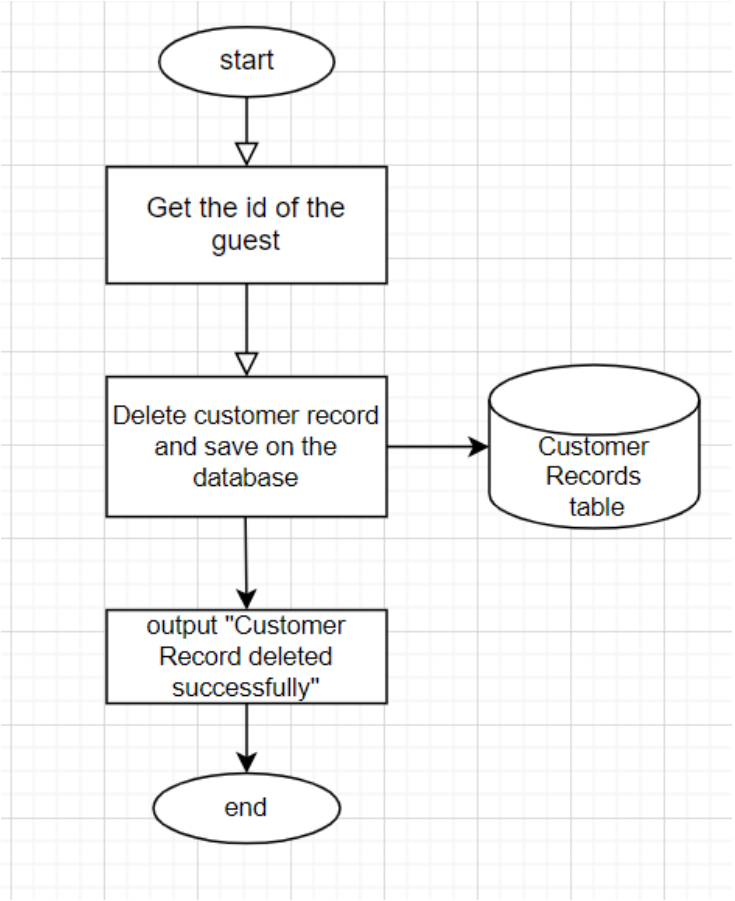


Manage Customer Records Flowcharts

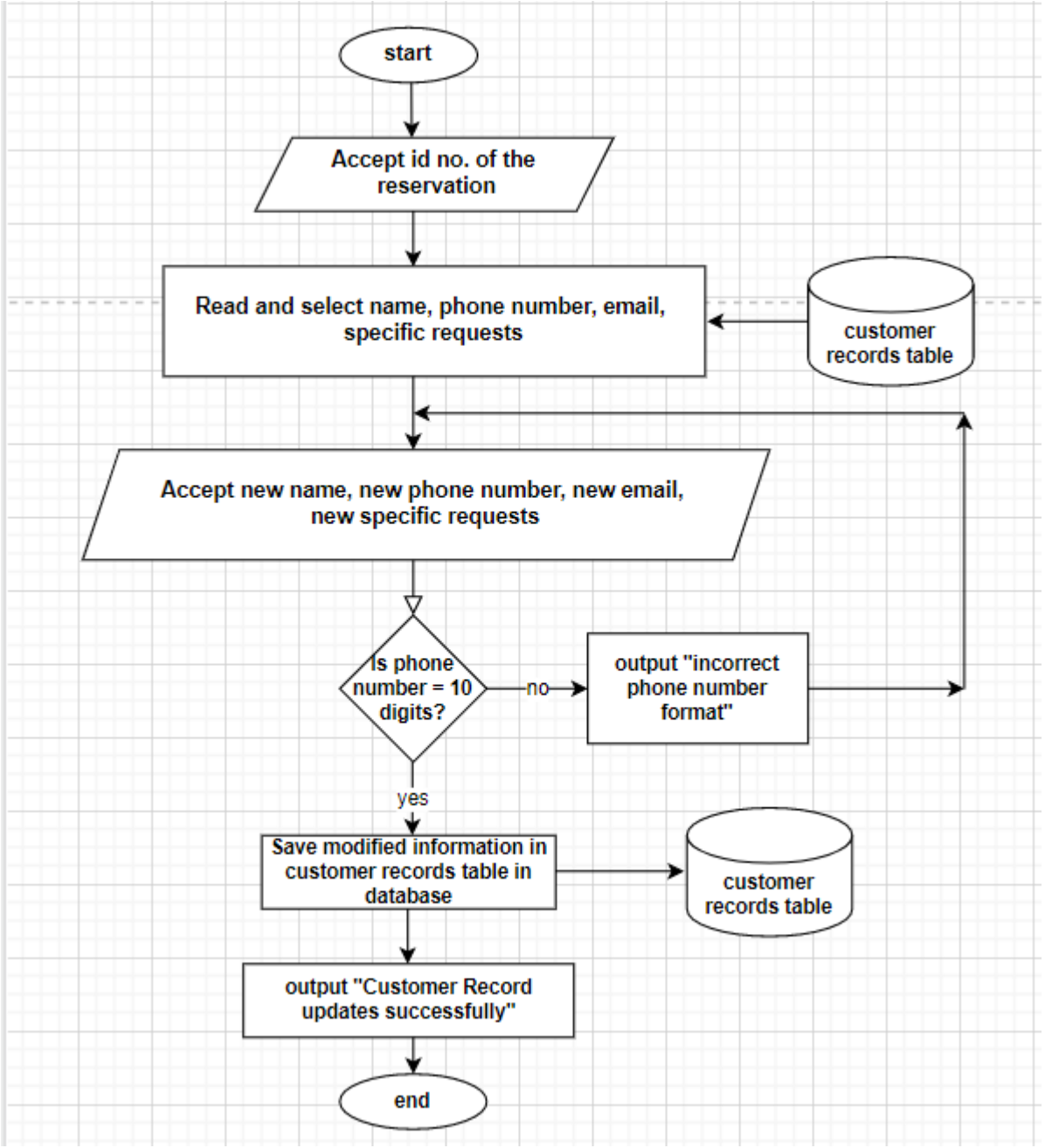
Flowchart number 5 (Add Customer Record): This flowchart is similar to flowchart number one. It shows the necessary steps that need to be taken in order to add a new customer record. The user inputs customer's information (Name, Email, Phone) and saves it to the table in the database.



Flowchart number 6 (Delete Customer Record): This flowchart is a similar one to Flowchart number 3. But, this flowchart outlines how the program deletes a record in the customer records table. It shows the steps taken by the employee when deleting a record.



Flowchart number 7 (Modify Customer Record): This flowchart is a similar one to Flowchart number 2. But, this flowchart outlines how the program modifies a record in the Customer Records table. It shows the steps taken by the employee when modifying a Customer Record.

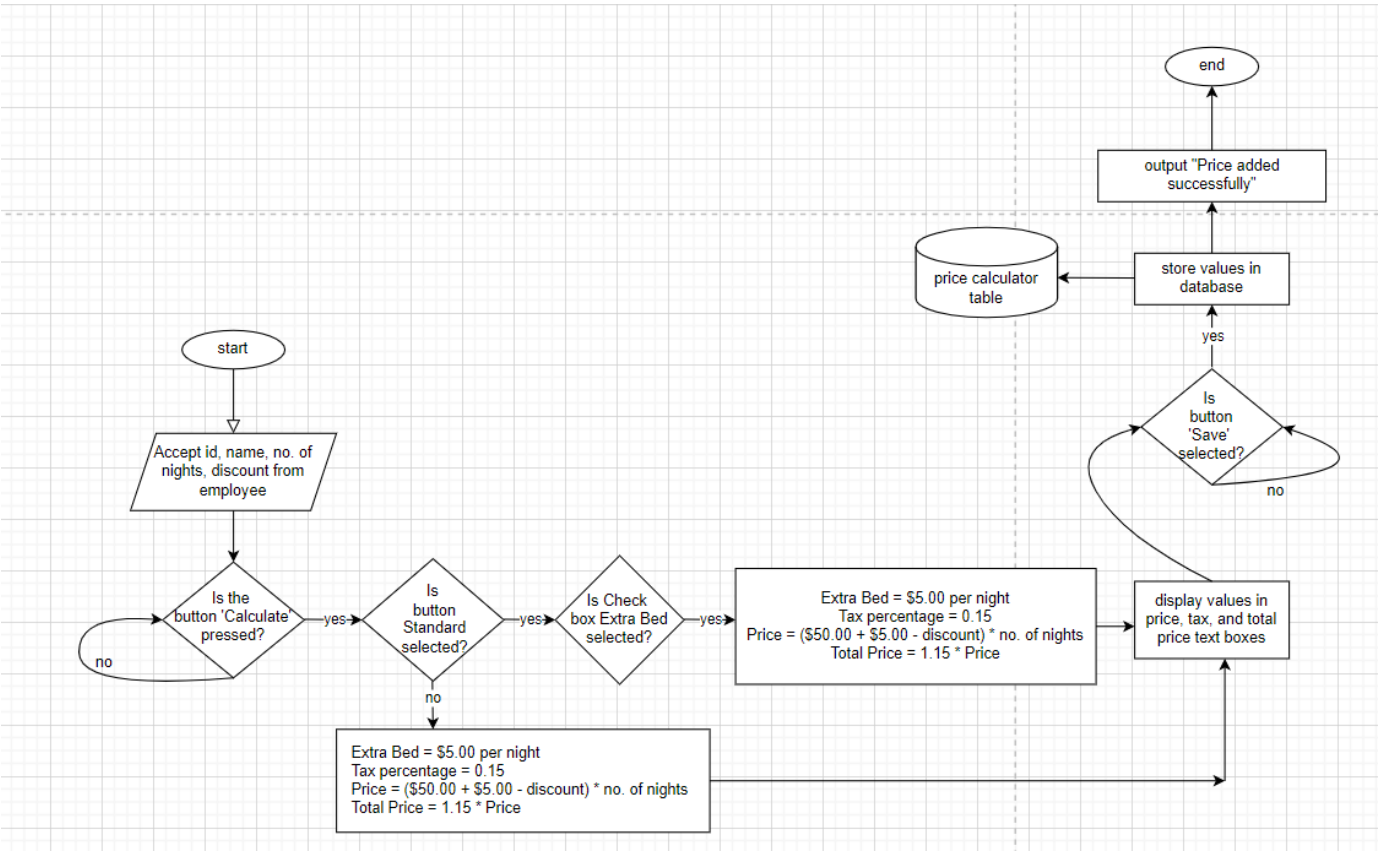


Ruth Bekalu Criterion B

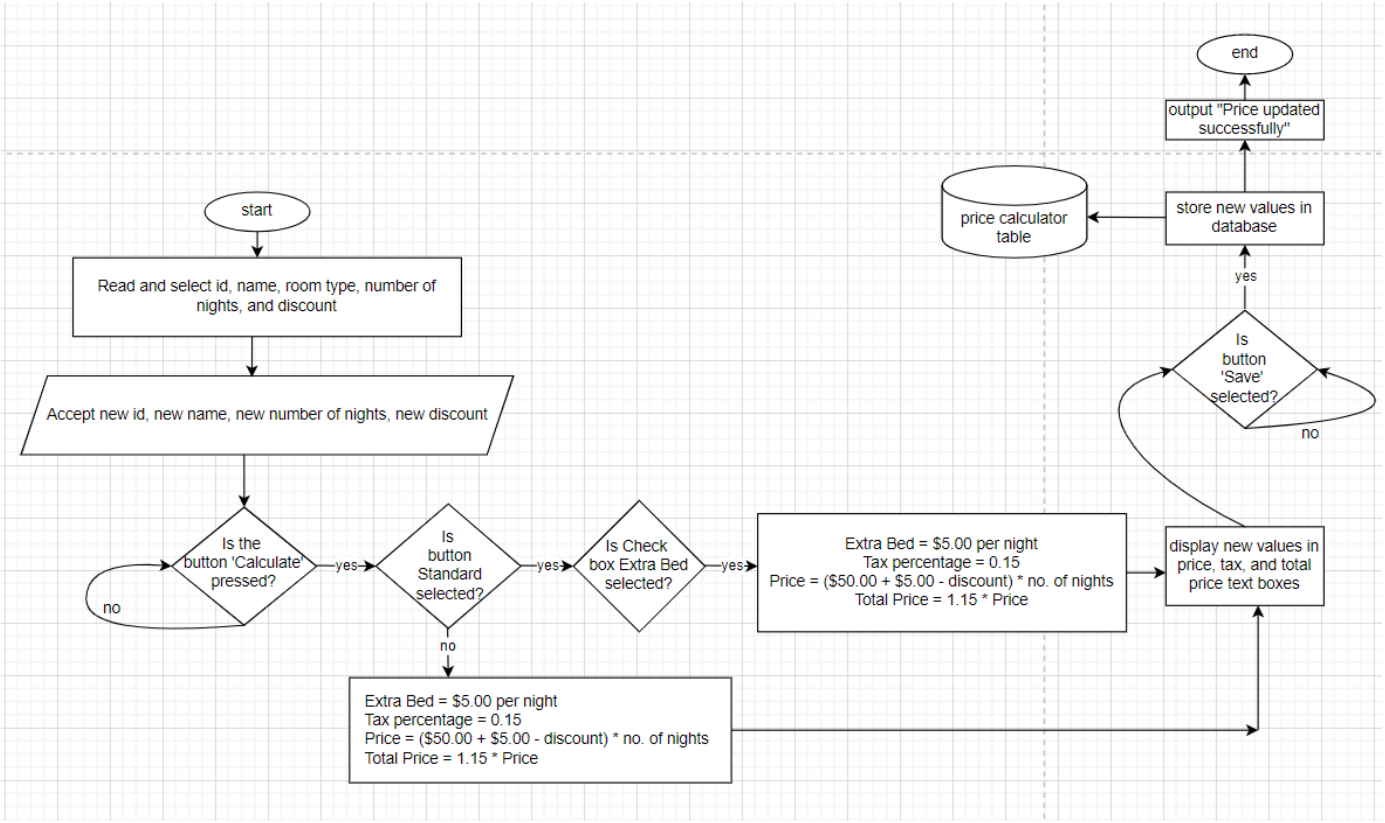
Price Calculator Flowcharts

.

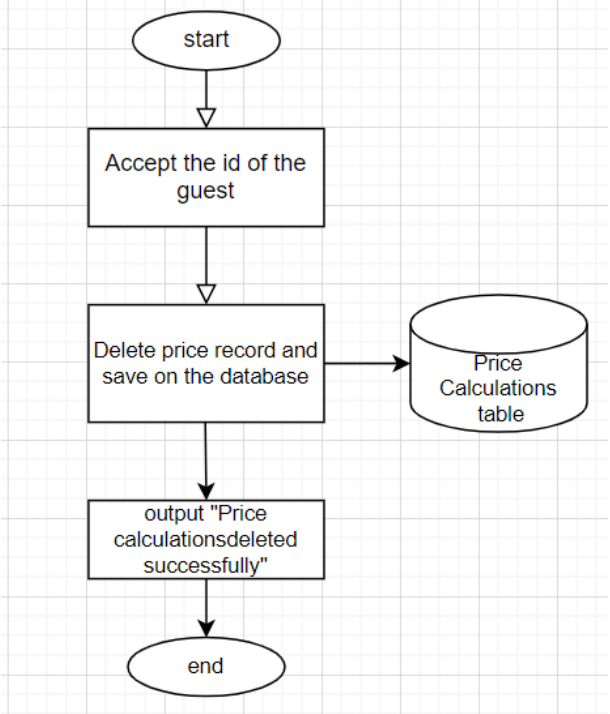
Flowchart number 7: This flowchart shows all the calculations done when displaying the total price of the reservation. It uses totalling to calculate the price for each type of room. It also uses totalling to calculate the price of an extra bed, if the customer chooses to have one, and to calculate the total price including 15% VAT. It also reduces the amount of discount inputted by the employee from the price per night. It then shows how the save button stores the values into the database.



Flowchart 8: Modification of Price Calculations: This flowchart is similar to flowchart number 7, except it reads/selects the values on the table and displays them on the text boxes. The employee then just alters the desired values by redoing the processes outlined in flowchart number 7.



Flowchart 9: Delete Price Calculations: Deletes the records from the table in the database, using the unique id.



Ruth Bekalu Criterion B

Database table designs

PK = Primary Key AI = AUTO_INCREMENT

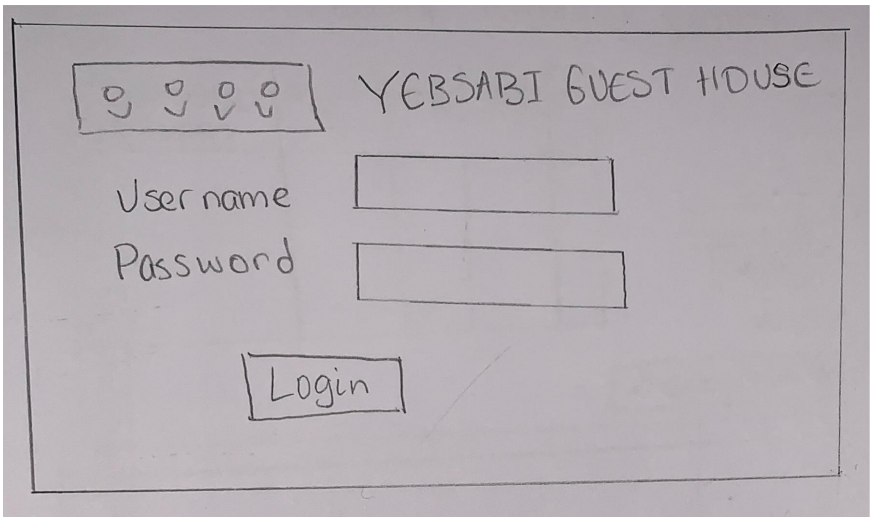
reservations **reservations**
id: int(11) PK, AI
room_no: varchar(255)
room_type: varchar(255)
Name: varchar(255)
Phone_number: varchar(255)
Email: varchar(255)
checkin_date: date
checkout_date: date
no_of_adults: int(10)
no_of_kids: int(10)
emp_name: varchar(255)

reservations **customer**
id: int(11) PK
name: text
email: text
phone: text
specific_requests: text

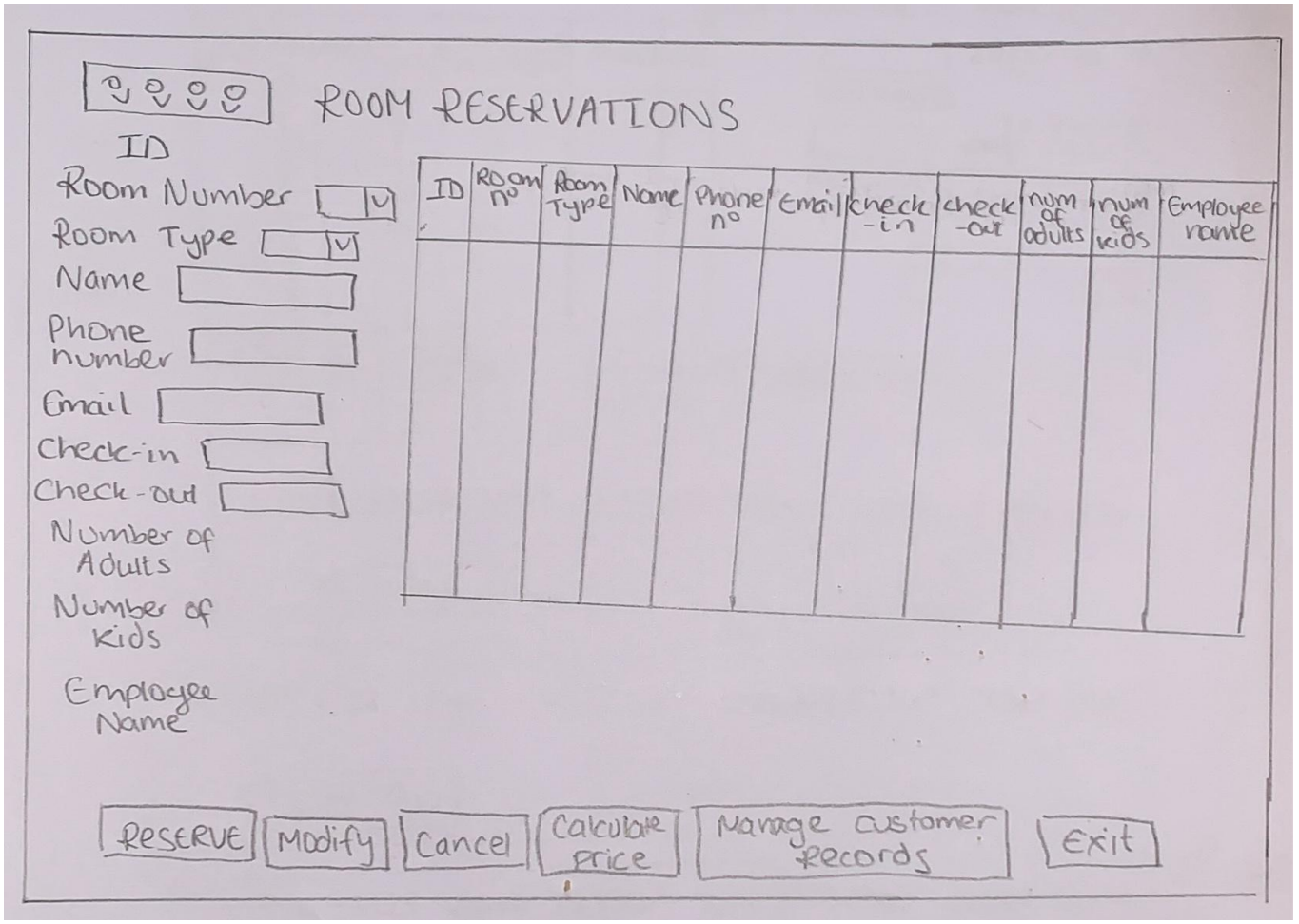
reservations **price_calc1**
guestid: int(11) PK
name: varchar(100)
room_type: varchar(255)
number_of_nights: varchar(100)
discount: varchar(100)
total_price: varchar(100)

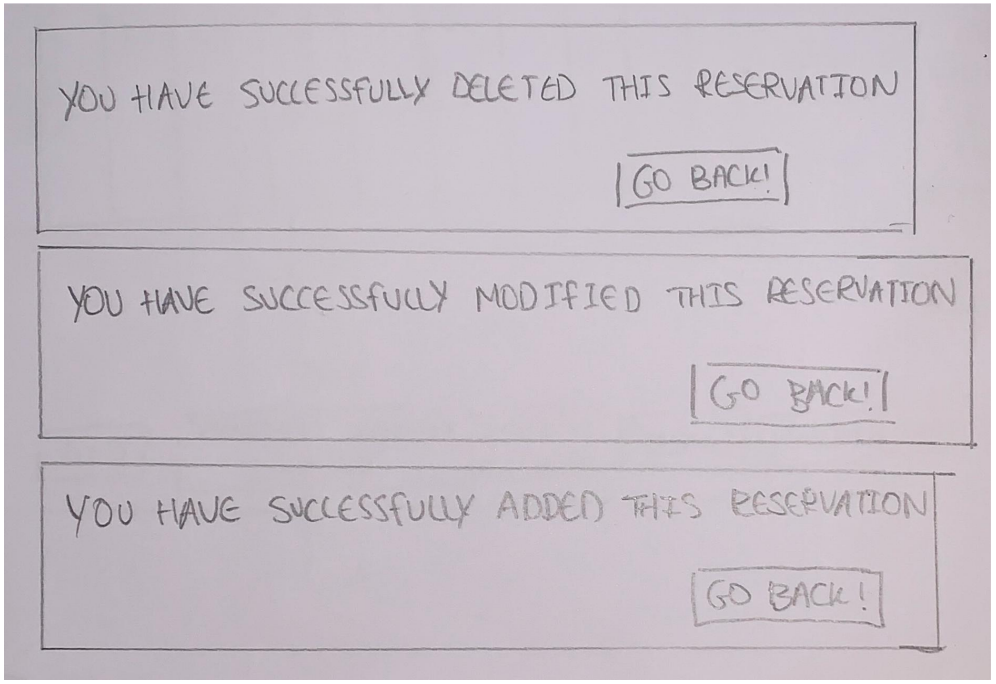
Ruth Bekalu Criterion B

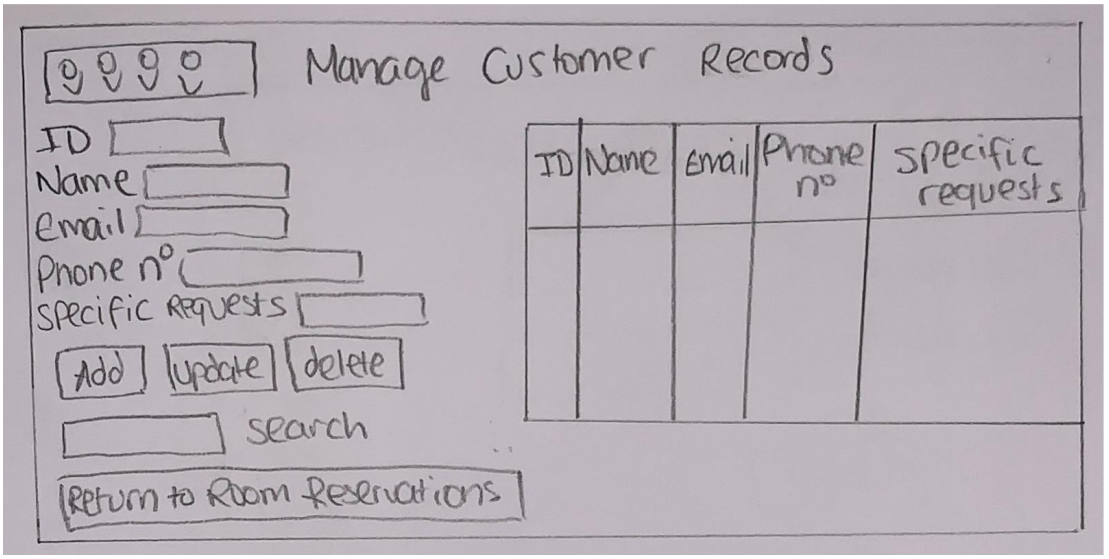
Design of Panels

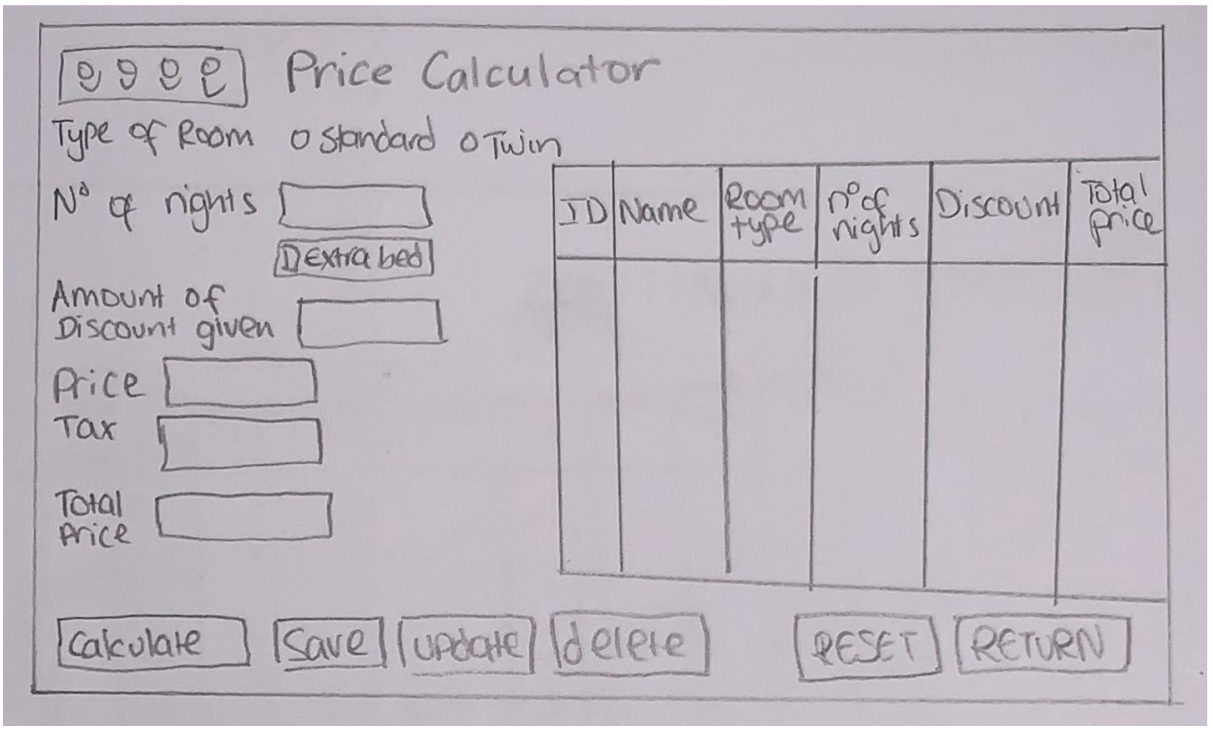


Ruth Bekalu Criterion B





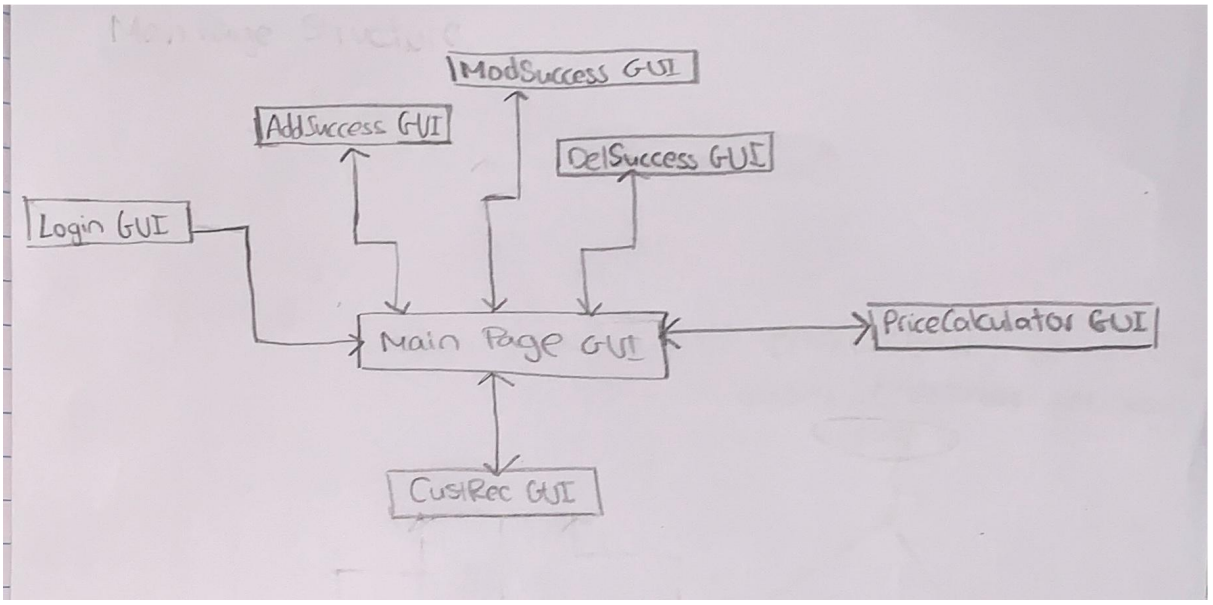

Manage Customer Records
ID
Name
Email
Phone n°
Specific Requests
Add
update
delete
search
Return to Room Reservations
ID
Name
Email
Phone n°
specific requests


Price Calculator
Type of Room
o standard
o Twin
N° of nights
Extra bed
Amount of Discount given
Price
Tax
Total Price
ID
Name
Room type
n° of nights
Discount
Total price
Calculate
Save
update
delete
RESET
RETURN

Ruth Bekalu Criterion B

Main Page GUI Structure

The following shows all the forms coming to or from the Main Page. The Price Calculator form, CustRec form, AddSuccess form, ModSuccess form, and Del Success form will all have return buttons, so the user can go to the forms from the main page and can also go back to the main page. The Login form can only go to the Main Page; the user cannot go back to the login from the Main



Main Class Components

Variable Name	Data Type Description	Description
RoomNo_Combo	jComboBox	Select Room Number
RoomType_Combo	jComboBox	Choose between choices in drop down list
CancelRes_Bt	jButton	Cancels selected reservation
ModifyRes_Bt	jButton	Modifies selected reservation
Reserve_Bt	jButton	Go to Reserve GUI screen
Clear_Bt	jButton	Clears all reservations
DateChooser1	jDateChooser	Input date
Name_Text	jTextField	Enter name of guest
Phonenumber_Text	jTextField	Enter phone number of guest
Email_Text	jTextField	Enter Email of guest
NoofAdults_Text	jTextField	Enter number of adults

Ruth Bekalu Criterion B

NoofKids_Text	jTextField	Enter number of kids
Employeename_Text	jTextField	Enter name of employee making reservation
CalcPrice_btn	jButton	Takes to PriceCalculator form
btn_ManCustRec	jButton	Takes to Cust Rec Form
tbl_res	jTable	Displays reservations table on database

Manage Customer Records

Variable Name	Data Type Description	Description
tfName	jTextField	Enter Name of Guest
tfemail	jTextField	Enter Email of Guest
tfPhone	jTextField	Enter Phone Number of Guest
btn_Add	jButton	Saves customer record to customer table in database
btn_Update	jButton	Modifies records in customer table in database
btn_Delete	jButton	Deletes records in customer table in database
btn_RetToCR	jButton	Takes user to Main Page form
search_btn	jButton	Allows user to search for information on database
search_txt	jTextField	User can type in information to be searched for
tbl_cust	jTable	Displays customer table on database

Calculate Price (Main Classes)

Variable Name	Data Type	Description
calculate_btn	jButton	Displays values in appropriate text boxes
save_btn	jButton	Saves values to database

return_btn	jButton	Go to previous GUI screen
reset_btn	jButton	Clears all values
roomtype_cb	jComboBox	Selects type of room
Exbed_cb	jCheckBox	Selects Extra Bed value
nofnights_txt	jTextField	Enter number of nights guest is staying
discount_txt	jTextField	Input amount of discount
tblprice	jTable	Displays price_calc table on database

Test Plan

1) Unit Testing - Tests that verify the functionality of a specific section of code/component of the program.
2) Code Review – Review of the code to eliminate errors and enhance program performance.
3) Integration Testing – Testing that verifies that the interfaces between components in a system function properly.
4) System Testing – Tests the final completely integrated system to verify that it meets the requirements stated by the client.
5) Acceptance Testing - test conducted by the client to determine if the requirements of his contract are met.

Action to be tested	Test Method	Success Criteria no. & purpose
GUI functionality check	Unit Testing Run the program and check if each button works.	(All numbers) Checks if each button works without any errors.
Prevention of Double Booking	Unit Testing Checks if the “Reserve” button works.	(No. 4) GUI that checks if unavailable date is selected is displayed and, if so, error message is displayed.
New Reservation check	Unit Testing Checks if the new reservation GUI works by running and adding a new reservation.	(No. 1)New reservation is added successfully.
Modify Reservation check	Unit Testing Checks if Modify reservation GUI and modifications are saved.	(No. 2) User can successfully modify reservations.
Cancel Reservation check	Unit Testing Checks if Cancel Reservation GUI works and deletes a reservation.	(No. 3) User can successfully delete a reservation.
Add Customer Records Check	Unit Testing Checks if CustMain GUI works by running and allowing user to add information.	(No. 5) New Customer is added successfully and user can modify and delete their information.
Modify Customer Records Check	Unit Testing Checks if the Update GUI works and modifications are saved.	(No. 6) User can successfully modify Customer Records.
Delete Customer Records Check	Unit Testing Checks if the Delete GUI	(No. 7) User can successfully delete a

	works and deletes a customer record.	customer record.
Serialization check	Unit Testing Checks if all data is saved for future use by exiting and reopening.	(No. 10) User can access all data even after exiting
Username and Password check	Unit Testing Checks if correct password and username works by running and checking if log in is allowed.	(No. 11) User can log in using the same username and password.
Price calculation check	Unit Testing Checks if Total Price and Total Price + VAT correctly calculates by checking manually with a calculator.	(No. 9) Total Price and Total Price + VAT is successfully displayed according to client's information.
Saving Price Check	Unit Testing Checks if PriceCalculator GUI works by running and allowing user to add price information.	(No. 10) New Price Record is added successfully.
Modify Price Check	Unit Testing Checks if the Update GUI works and modifications are saved.	(No, 10) User can successfully modify Price Records.
Delete Price Check	Unit Testing Checks if the Delete GUI works and deletes a customer record.	(No. 10) User can successfully delete a price record.
Code efficiency and accuracy check.	Code Review Checks if any error messages are displayed on code.	(all numbers) No syntax, logical, or compiling errors.
Checks to see if all GUI work properly together.	Integration Testing Checks if buttons don't interfere with each other and complete their function without compromising other GUI's functions.	(all numbers) User can switch between different GUI without any errors.
Final Compilation Test	System Testing	(all numbers) Requirements met
Test conducted by client	Acceptance Testing Client uses the product and checks if there's anything they want to change.	(all numbers) Client Satisfied with product

Ruth Bekalu Criterion B