Database Designs and Management

MIS 3123~ Project  
May 3rd 2009

Documentation for

Hotel Database Application

Implementation of a Database system for Real world Business

Table of Content

[I. Summary 3](#_Toc52873031)

[II. Introduction 4](#_Toc52873032)

[III. Queries and Reports 7](#_Toc52873033)

[IV. Entity Relationship Diagram 9](#_Toc52873034)

[V. User Manual 10](#_Toc52873035)

[VI. Apendix 19](#_Toc52873036)

# Summary

Hotel Database Application Summary

This project is a desktop database application. The design of the database application allows users of a hotel to capture information from its operations, and convert them into meaningful data. The data collected will be interpreted by management to improve the hotel operation, and in turn, maximizing the profitability of the hotel. The features of the application include user friendly form interface and simply one click reports creations.

# Introduction

Introduction to Hotel Database Project

This project was prepared in Microsoft Access 2007 and Visual Studio 2008 Express Edition. The VB coding language was used to create the desktop application and the queries was generated using SQL scripting language within the VB codes of the desktop application.

The main purpose of this database application is to provide a user friendly environment for hotel employees to capture information about their guest and provide accurate reports for management. This database application will help owners of small and medium size hotels in their every day operation.

Every hotel is operated on number of rooms they have in Inventory. They also generate revenue through sale of rooms. The beginning process of sale of room starts with reservation.

This application will assist managers to track the amount of reservation the hotels receives per day. Also, it will allow the managers to create a report that will show reservation information for next day of business. This will help manager to prepare in advance for future guests. In addition, this application will allow managers to administer hotel employee. Manager will be able to restrict certain features of this application to different groups of employees. For instance, Receptionist access will be limited to the receptionist menu. In the receptionist menu the employee could only add information pertaining to check in and check out. Where as, “Admin’s” can add new employees to the database, manage payroll and create user names and passwords for employees.

They key feature of this application is the user friendly environment for the hotel owners and their employees. The application can be installed on the computers of the hotels. After installation is completed the program can be ready to use instantly. The application does not require employees to have technical savvy skills to use the application.

# Tables within the Database

Overview of the tables in the Microsoft Access database: The type of tables and why they were created.

**Check-in & Check-out Table**

These are the base operations of any working hotel. It’s the main foundation for answering the reason a database is created for a hotel. The main point of a hotel is to incorporate their guests into rooms and they must reserve a date for the hotel of their choice. Within a database, our tables include the name of the person and the date that they will be arriving which is the most crucial information in the reservation.

**Guest Table**

Our customer table is all the relevant information pertaining to the person who is reserving the room. Where they are coming from and who they are is mandatory information to make sure that if the customer has a problem we can make sure that we have credible information to reach them and correct a mistake.

**Reservation Table**

The Reservation table stores information about when a room is assigned to a guess and for how long it assigned for. This is basically our inventory. All of our rooms that are taken by a customer will be documented in this table. We have duration of stay and which type of room is taken as well.

**Room Table**

In a hotel, there are several different types of rooms that can be occupied. This is the table that showcases our rooms, how much money each room will cost, and the recommended capacity.

**Payment Table**

Each customer will be tied to the payment table. This table is where we store all the financial information we need on the guest. We included how they will be paying for there room and the final price for their rooms as well.

**Employee Table**

This is our table to provide the inner workings of the hotel. This table shows the employees we have working to keep the stability of the hotel up. We basically just keep the personal information of our employees here.

# Tables within the Database Cont…

**Payroll Table**

The inclusion of payroll is to provide our employees with a realistic salary to compensate for their specific roles in the corporate ladder. We have pay deductions and hourly rate provided to enhance how realistic our company runs.

**Users Table**

Finally we have our users table. This table shows the members that have access to our database. Not much information is shown here about those individuals, that information you can see in the employee table, but their name, password, and job.

# Queries and Report’s

An overview of the queries and reports used in the Hotel Database Application

**Queries**

We have several queries showing details as to what a hotel needs. We chose to do two types of queries integral to the business of running an actual hotel.

Next Day Arrival: This query will show in detail the data needed to forecast that week of business and show the necessary information the hotel we need to know. We included customer information as well as room type so that the staff of that hotel can keep track of room inventory. Next day arrival is very much needed so that the hotel that would use this type of database can keep track of customers and their expected arrival.

**Payment Query**

In this query it will show data about payment and room options. This data is necessary to keep track of sales and the type of rooms customers prefer. Broken down into cash, check, or credit we can see the preference of the customer as to which they pay with compared to the amount being spent. Also, we can see the rooms being rented during the time of season and have a forecast as to what rooms will be used the most that time of the year. On top of that information we can see what rooms generated the most amount of revenue for the year. With this the hotel can generate an idea as to what rooms need to be expanded across the hotel and raise or lower the room rates. Useful forecasting queries will benefit hotels.

**Payroll**

Within this query we can see employee information and amounts that are being given to the hotel’s employees. We used basic calculations in this query to display Net Pay and Total Pay. This is useful to keep track of employees and their salaries of hours worked.

**Reports**

The reports decided to be used within our database centered on the two most important parties of a hotel; the customer and the employees. We created a Customer Receipt and a Payroll Report to keep track of hotel spending and hotel income as generated by the renting of rooms and the loss by wages given.

Customer Receipt: The most useful report we could see to be the most practical is a customer receipt. We used this report to show the customer their own information, nights stayed, payment type, and amount owed/received.

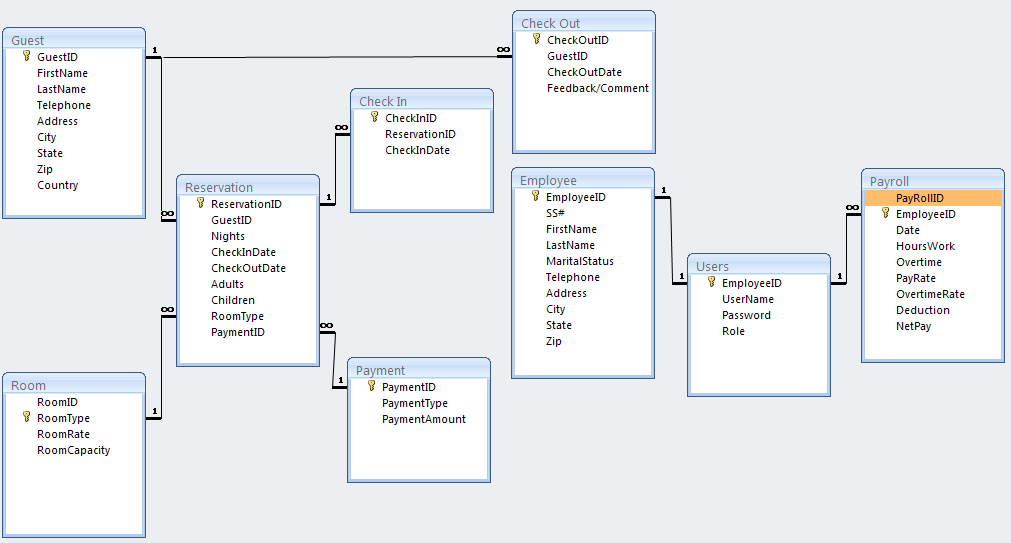
# Queries and Report’s Cont…

This is useful for the hotel and customer to show what the hotel has about them and that the hotel has the customer in their database and that they are be taken

Care of. The customer receipt is to be used for records for that hotel to help forecast the years service and what is needed to be adjusted for the next year’s business, such as raising or lowering room rates, adding extensions to the hotel (more rooms, more amenities), and hiring more staff.

Payroll Report: Just like the query, this report shows an ongoing and instantly updated, by week, of payroll expense. This can be used to keep a watchful eye over hotel spending on hotel employees and hours worked. This payroll will be able to keep track of individual employees and the actual hours worked to help with evaluations later on in the future. Instead of looking at a query you just have one report in front of you to look at and gauge the employee’s stats.

# Entity-Relationship Diagram



# User Manual

Overview about how to use the Hotel Database Application

**Starting program**

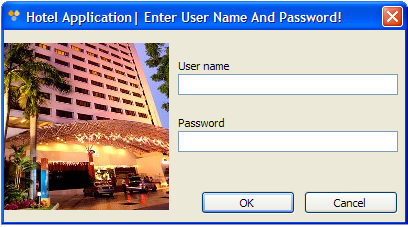


**Figure 1.0** shows the initial screen of after the after application is launched.

The Picture above (**Figure 1.0**) is the splash screen that is use to display a short introduction to the program.

**Login into Application**

After splash screen loads a login form will prompt the user to enter their username and password. This is illustrated in the **Figure 1.1** below.

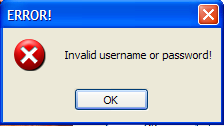


**Figure 1.1** shows the login screen that prompts the user for login credentials.

# User Manual Cont…

**Authentication of login**

If login credential does not match the data the users table in the **hotelapp\_db database user table** then user is prompted by an Error message. The Error message is illustrated in **figure 1.2.**

****

**Figure 1.2** invalid username and password Error message.

Login credential is authenticated by a SQL script that selects from the Role column in the users table where the username entered in the username textbox ‘txtu’ is equal to username in the users table and the password users table is equal to text entered in the password textbox ‘txtp’. This is illustrated in the VB code below.

Dim con As New OleDbConnection("Provider=Microsoft.jet.oledb.4.0;data source=E:\database project\Hotelapp\_db.mdb")

Dim cmd As OleDbCommand = New OleDbCommand("SELECT Role FROM Users WHERE UserName = '" & txtu.Text & "' AND Password = '" & txtp.Text & "'", con)

con.Open()

If the above criteria is not met then message box (**Figure 1.2**) is displayed.

An “IF” statement is used in the VB Code to show the error message. This is illustrated below.

If (sdr.Read() = False) Then

MessageBox.Show("Invalid username or password!", "ERROR!", MessageBoxButtons.OK, MessageBoxIcon.Error)

# User Manual Cont…

**Restricting Users Access**

Based on the roles of users specific access privileges were granted to each form in the application. **Table 1.0** below shows the access granted to different roles.

**Table 1.0** illustrate the users roles and the type of access granted

|  |  |  |
| --- | --- | --- |
| **User Roles** | Description of Access | Table Access |
| Receptionist | Add New, Delete, Edit Reservation Data and Print | Guest, Check In, Reservation and Check Out |
| Manager | Generate and Print Reports | Manager Form/Report Menu |
| Admin | Add New, Delete, Edit Employee Data | Employee, Users, and Payroll |

The different access to the tables was achieved by making forms in the application specific to the different user roles. For instance, Receptionist user role shows Reservation form, Manager user role shows Manager form and the Admin user role show Admin Form. The different forms content table adapters and binding source that connected the form fields with the tables in the database. The VB codes below were used to show the different form base on the login user role.

role = sdr.GetString(0)

If role = "manager" Then

Dim managerForm As New managerForm

managerForm.Show()

With Me

.Hide()

.txtp.Text = ""

.txtu.Text = ""

End With

ElseIf role = "receptionist" Then

Dim mainform As New MainForm

mainform.Show()

With Me

.Hide()

.txtp.Text = ""

.txtu.Text = ""

End With

ElseIf role = "admin" Then

Dim adminform As New AdminForm

adminform.Show()

With Me

.Hide()

.txtp.Text = ""

.txtu.Text = ""

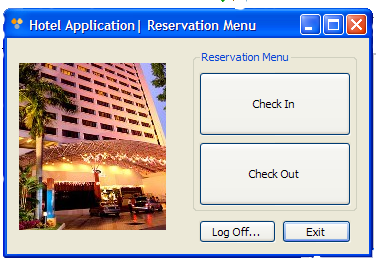
End With

End If

# User Manual Cont…

**Different Form Base on User Role**

**Figure 1.3** shows Reservation Menu the Members of the receptionist role have access to the Reservation member. The members of the Admin Role have access to the Administrator Menu illustrated in **Figure 1.4**. And the members of the Manager Role have access to the Report Menu shown in **Figure 1.5**.

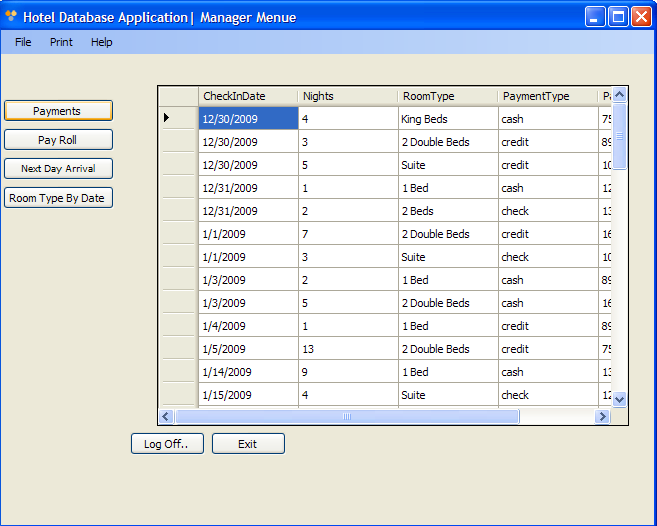
**Figure 1.3** sh****ows the Reservation Menu

****

**Figure 1.4** shows the Administrator Menu

# User Manual Cont…

**Figure 1.5** shows the Manager (Report) Menu

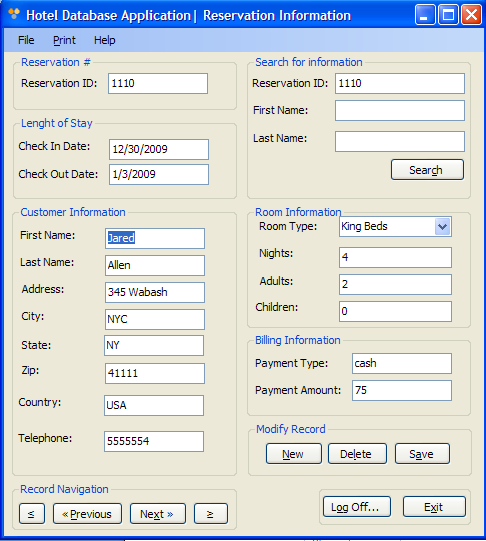


There are two forms that can be derived from the Reservation Menu. They are the Check In Form and the Check Out Form. These forms are used to collect reservation information when a customer checks in to the hotel and check out from the hotel. The check out form is primary used to collect feed back from the customers. These Forms are illustrated in **Figure 1.6** and **Figure 1.7**.

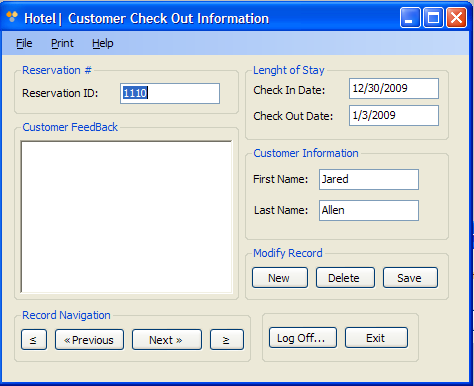
The form provides a user friendly interface for the employees of the hotel. They can browse through records by using the Navigate Buttons at the bottom of **Figure 1.6** and Figure **1.7**. In addition to the navigation buttons, an employee can find for records from the search box provided on the Check In Form. They also have the option of using the buttons on the form to Add, Delete, and Save records or using the menu strip at the top of the forms. The Employee can also Exit or log Off from application using the buttons at the button of the form or the menu strip at the top of the Form. If Employees click on the Exit Button they will be prompted by a message box to confirm Exit. The message box is show in **Figure 1.8**

# User Manual Cont…

**Figure 1.6** shows the Check In Form

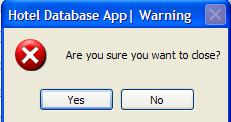


**Figure 1.7** shows the Check Out Form



# User Manual Cont…

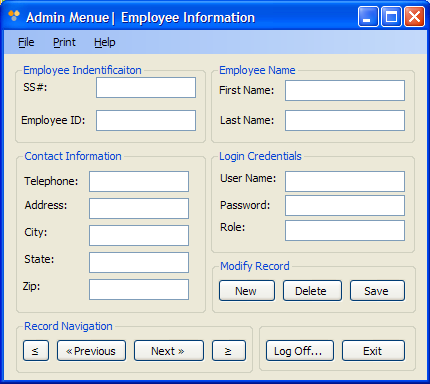
**Figure 1.8** shows the Exit confirmation Error Message Box



**Figure 1.8** is used in all forms in the applications.

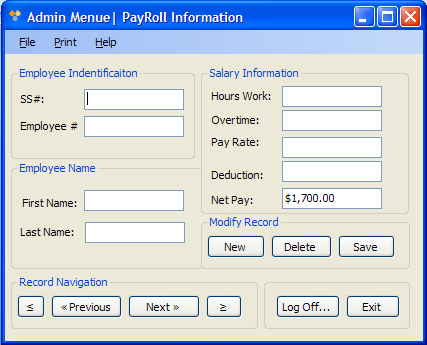
The Administrator Menu Form allows the Admin’s to manage employees information and their payroll. This is illustrated in **Figure 1.9** and **Figure 2.0**. The features are similar to that of reservation; however, the information manage is that of employees.

**Figure 1.9** shows the form used to administer Employee Information



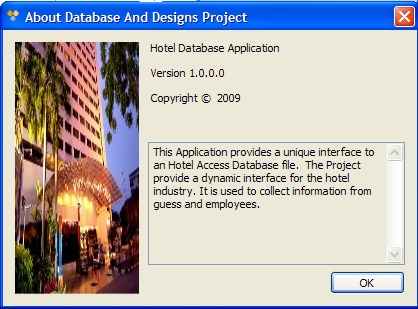
# User Manual Cont….

**Figure 2.0** shows the form used to calculate employees payroll



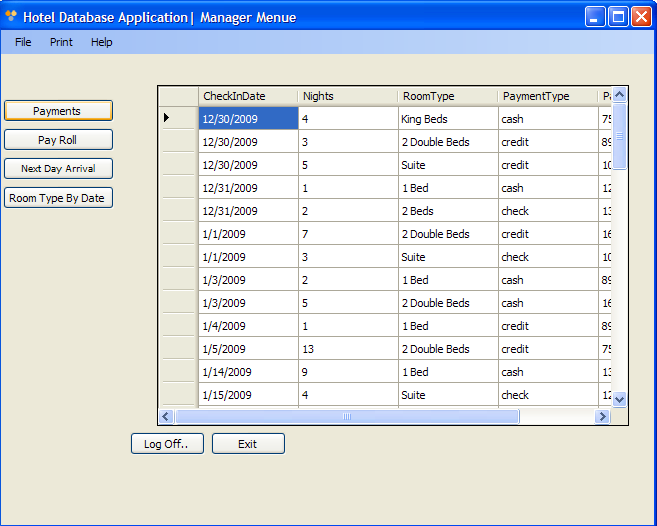
The “About Box” form is can be access from all the forms within the application. It can be obtain by clicking on the **Help** link found in the menu strip at the top of every form. This is illustrated in **Figure 2.1.**

**Figure 2.1** show the About “Box”



# User Manual Cont…

The Report menu gives the manager the ability to retrieve information from the all the tables within the database. The reports are generated by clicking on the different button on the left of the form this is illustrated in **Figure 2.2**.

**Figure 2.2** show the Manager Report menu

# 

# Appendix

**Application Source Code:**

Login Form Source Code

Imports System.Data.OleDb

Public Class Login

Private Sub OK\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles OK.Click

Dim con As New OleDbConnection("Provider=Microsoft.jet.oledb.4.0;data source=E:\database project\Hotelapp\_db.mdb")

Dim cmd As OleDbCommand = New OleDbCommand("SELECT Role FROM Users WHERE UserName = '" & txtu.Text & "' AND Password = '" & txtp.Text & "'", con)

con.Open()

Dim sdr As OleDbDataReader = cmd.ExecuteReader()

Dim role As String

Try

'Password verification and go to main form.

If (sdr.Read() = False) Then

MessageBox.Show("Invalid username or password!", "ERROR!", MessageBoxButtons.OK, MessageBoxIcon.Error)

Me.txtp.Text = ""

Else

role = sdr.GetString(0)

' User Role verification.

If role = "manager" Then

Dim managerForm As New managerForm

managerForm.Show()

'hide login form and clear login info incase of loggoff

With Me

.Hide()

.txtp.Text = ""

.txtu.Text = ""

End With

ElseIf role = "receptionist" Then

Dim mainform As New MainForm

mainform.Show()

'hide login form and clear login info incase of loggoff

With Me

# Appendix Cont…

Login Form Source Code

.Hide()

.txtp.Text = ""

.txtu.Text = ""

End With

ElseIf role = "admin" Then

Dim adminform As New AdminForm

adminform.Show()

'hide login form and clear login info incase of loggoff

With Me

.Hide()

.txtp.Text = ""

.txtu.Text = ""

End With

End If

End If

Catch ex As Exception

MessageBox.Show("Authentication Failed...")

End Try

If con.State <> ConnectionState.Closed Then

con.Close()

End If

con.Close()

End Sub

' Cancel button

Private Sub Cancel\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Cancel.Click

Me.Close()

End Sub

End Class

# Appendix Cont…

Login Reservation Form Source Code

Public Class MainForm

'show check in form.

Private Sub checkInButton\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles checkInButton.Click

Me.Hide()

ReservationForm.Show()

End Sub

'show check out form.

Private Sub checkOutButton\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles checkOutButton.Click

Me.Hide()

CheckOutForm.Show()

End Sub

' Return user to login form.

Private Sub logOffButton\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles logOffButton.Click

Login.Show()

Me.Hide()

End Sub

Private Sub ReservationsToolStripMenuItem\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)

Me.Hide()

ReservationForm.Show()

End Sub

Private Sub ManagerReportToolStripMenuItem\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)

Me.Hide()

CheckOutForm.Show()

End Sub

Private Sub ExitToolStripMenuItem\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)

Dim response As String

response = MsgBox("Are you sure you want to close application?", 20, "Warning")

If response = vbYes Then

Me.Close()

End If

# Appendix Cont…

Login Reservation Form Source Code

End Sub

Private Sub LogOffToolStripMenuItem\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)

Login.Show()

Me.Hide()

End Sub

Private Sub AboutToolStripMenuItem\_Click(ByVal sender As Object, ByVal e As System.EventArgs)

AboutBox1.Show()

End Sub

End Class

# Appendix Cont…

Login Check In Form Source Code

Imports System.Data.OleDb

Public Class ReservationForm

Private Sub searchButton\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles searchButton.Click

End Sub

Private Sub ReservationForm\_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load

'TODO: This line of code loads data into the 'Hotelapp\_dbDataSet.Room' table. You can move, or remove it, as needed.

Me.RoomTableAdapter.Fill(Me.Hotelapp\_dbDataSet.Room)

Me.Check\_OutTableAdapter.Fill(Me.Hotelapp\_dbDataSet.Check\_Out)

Me.Check\_InTableAdapter.Fill(Me.Hotelapp\_dbDataSet.Check\_In)

Me.GuestTableAdapter.Fill(Me.Hotelapp\_dbDataSet.Guest)

Me.PaymentTableAdapter.Fill(Me.Hotelapp\_dbDataSet.Payment)

Me.ReservationTableAdapter.Fill(Me.Hotelapp\_dbDataSet.Reservation)

End Sub

Private Sub newButton\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles newButton.Click

'add new record to database.

ReservationBindingSource.AddNew()

RoomBindingSource.AddNew()

Check\_OutBindingSource.AddNew()

Check\_InBindingSource.AddNew()

GuestBindingSource.AddNew()

PaymentBindingSource.AddNew()

End Sub

Private Sub saveButton\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles saveButton.Click

'save record to database.

Validate()

ReservationBindingSource.EndEdit()

RoomBindingSource.EndEdit()

Check\_OutBindingSource.EndEdit()

Check\_InBindingSource.EndEdit()

GuestBindingSource.EndEdit()

PaymentBindingSource.EndEdit()

TableAdapterManager.UpdateAll(Me.Hotelapp\_dbDataSet)

MessageBox.Show("Record saved")

End Sub

Private Sub deleteButton\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles deleteButton.Click

'Delete record from database.

Dim response As String

response = MsgBox("The current record is about to be deleted!", 20, "Hotel Database App| Warning")

If response = vbYes Then

ReservationBindingSource.RemoveCurrent()

GuestBindingSource.RemoveCurrent()

RoomBindingSource.RemoveCurrent()

PaymentBindingSource.RemoveCurrent()

Check\_InBindingSource.RemoveCurrent()

Check\_OutBindingSource.RemoveCurrent()

GuestBindingSource.EndEdit()

RoomBindingSource.EndEdit()

PaymentBindingSource.EndEdit()

Check\_InBindingSource.EndEdit()

Check\_OutBindingSource.EndEdit()

TableAdapterManager.UpdateAll(Me.Hotelapp\_dbDataSet)

MessageBox.Show("Record Deleted")

End If

End Sub

Private Sub firstButton\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles firstButton.Click

'move to first record in database.

ReservationBindingSource.MoveFirst()

GuestBindingSource.MoveFirst()

RoomBindingSource.MoveFirst()

PaymentBindingSource.MoveFirst()

Check\_InBindingSource.MoveFirst()

Check\_OutBindingSource.MoveFirst()

End Sub

Private Sub previousButton\_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles previousButton.Click

'move to previous record in database.

ReservationBindingSource.MovePrevious()

GuestBindingSource.MovePrevious()

# Appendix Cont…

Login Check In Form Source Code

RoomBindingSource.MovePrevious()

PaymentBindingSource.MovePrevious()

Check\_InBindingSource.MovePrevious()

Check\_OutBindingSource.MovePrevious()

End Sub

Private Sub nextButton\_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles nextButton.Click

'move to next record in database.

ReservationBindingSource.MoveNext()

GuestBindingSource.MoveNext()

RoomBindingSource.MoveNext()

PaymentBindingSource.MoveNext()

Check\_InBindingSource.MoveNext()

Check\_OutBindingSource.MoveNext()

End Sub

Private Sub lastButton\_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles lastButton.Click

'move to last record in database.

ReservationBindingSource.MoveLast()

GuestBindingSource.MoveLast()

RoomBindingSource.MoveLast()

PaymentBindingSource.MoveLast()

Check\_InBindingSource.MoveLast()

Check\_OutBindingSource.MoveLast()

End Sub

Private Sub exitButton\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles exitButton.Click

'return to main menu

Dim response As String

response = MsgBox("Are you sure you want to close?", 20, "Hotel Database App| Warning")

If response = vbYes Then

Me.Hide()

MainForm.Show()

Me.Hide()

End If

End Sub

Private Sub Button4\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles loggOffButton.Click

Login.Show()

Me.Hide()

End Sub

# Appendix Cont…

Login Check In Form Source Code

Private Sub NewToolStripMenuItem\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles NewToolStripMenuItem.Click

'add new record to database.

ReservationBindingSource.AddNew()

RoomBindingSource.AddNew()

Check\_OutBindingSource.AddNew()

Check\_InBindingSource.AddNew()

GuestBindingSource.AddNew()

PaymentBindingSource.AddNew()

End Sub

Private Sub SaveToolStripMenuItem\_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles SaveToolStripMenuItem.Click

Validate()

ReservationBindingSource.EndEdit()

RoomBindingSource.EndEdit()

Check\_OutBindingSource.EndEdit()

Check\_InBindingSource.EndEdit()

GuestBindingSource.EndEdit()

PaymentBindingSource.EndEdit()

TableAdapterManager.UpdateAll(Me.Hotelapp\_dbDataSet)

MessageBox.Show("Record saved")

End Sub

Private Sub DeleteToolStripMenuItem\_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles DeleteToolStripMenuItem.Click

'Delete record from database.

Dim response As String

response = MsgBox("The current record is about to be deleted!", 20, "Hotel Database App| Warning")

If response = vbYes Then

ReservationBindingSource.RemoveCurrent()

GuestBindingSource.RemoveCurrent()

RoomBindingSource.RemoveCurrent()

PaymentBindingSource.RemoveCurrent()

Check\_InBindingSource.RemoveCurrent()

Check\_OutBindingSource.RemoveCurrent()

GuestBindingSource.EndEdit()

RoomBindingSource.EndEdit()

PaymentBindingSource.EndEdit()

Check\_InBindingSource.EndEdit()

Check\_OutBindingSource.EndEdit()

TableAdapterManager.UpdateAll(Me.Hotelapp\_dbDataSet)

MessageBox.Show("Record Deleted")

End If

End Sub

# Appendix Cont…

Login Check In Form Source Code

Private Sub LogOffToolStripMenuItem\_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles LogOffToolStripMenuItem.Click

Me.Hide()

Login.Show()

End Sub

Private Sub ExitToolStripMenuItem\_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles ExitToolStripMenuItem.Click

'return to main menu

Dim response As String

response = MsgBox("Are you sure you want to close?", 20, "Hotel Database App| Warning")

If response = vbYes Then

Me.Hide()

MainForm.Show()

MainForm.Show()

Me.Hide()

End If

End Sub

Private Sub PrintCustomerInfoToolStripMenuItem\_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles PrintCustomerInfoToolStripMenuItem.Click

'print Customer Information.

End Sub

Private Sub PrintReceiptToolStripMenuItem\_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles PrintReceiptToolStripMenuItem.Click

'Print Customer Receipt.

End Sub

Private Sub AboutToolStripMenuItem\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles AboutToolStripMenuItem.Click

'show about information.

AboutBox1.Show()

End Sub

End Class