V Semester B.Tech. (IT)  
 ICT 3136 DATABASE SYSTEMS LAB MINI-PROJECT IMPLEMENTATION DOCUMENT

Travel Desi

[Ticket Booking System]

A PROJECT REPORT

Submitted By

Mohammed Afnan Althaf Samad : 200911006

Ali Murtaza Rizvi : 200911027

Abstract:

Ticket Booking System is a computerized system used to store and retrieve information and conduct ticket reservation related to air travel. The project is aimed at exposing the relevance and importance of Airline Reservation Systems. It is projected towards enhancing the relationship between customer and airline agencies using ARS, thus making it convenient for the custoners to book flights.

Introduction:

The main purpose of this software is to reduce the manual errors involved in the airline reservation process and make it convenient for the customers to book the flights as when they require such that they can utilize this software to make reservations, add flights and add passengers.

The Travel Desi booking system allows the admin operating a travel agency to add flights to the database along with their Flight Name, Source, Destination, Price per ticket, Class of seats available and a few other details. It also has a feature to add passenger details to the database, the details can be added when a customer comes to the agency for the first time, on later occasions the same entry can be reused thus not causing data redundancy. And finally, the agent can also book tickets that the passenger needs by selecting the arrival and departure airports, just enter the customer id and then enter the seat id, proceed to click on book ticket.

The system also allows the User to search for flights that are available between the two travel cities, namely the "Departure City" and "Arrival City" for a particular departure and arrival dates. The system displays all the flight's details such as flight no, name, price, and duration of journey etc.

After search the system display list of available flights and allows customer to choose a particular flight. Then the system checks for the availability of seats on the flight. If the seats are available, then the system allows the passenger to book a seat. Otherwise, it asks the user to choose another flight.

Technology Used:

C# on Visual Studio for the front end

MS SQL for the database

Multiple Windows Forms are present, each performing specific functions.

Objectives:

* Develop an ecosystem for Ticket Booking
* Create an interface for Ticket Management by a Admin of a Travel Agency or a User
* Manage Flight Details and Passenger Details

Functional and Non-Functional Requirements:

Functional Requirements:

This database application will make it easy for the user to book flights and get the reservation details, add/view flights, add/view passengers, and book tickets.

Non-Functional Requirements:

* Efficiency Requirement: When the Ticket Booking System is implemented users will easily be able to add their details and then find the flights of their choice and then book them.
* Reliability Requirement: The system should accurately perform User Login, Flight Details, User Details and Ticket Booking
* Usability Requirement: The system is designed for a user-friendly environment for the User.
* Implementation Requirement: In implementing the whole system, it uses C# using Visual Studio front end which will be used for database connectivity, and the backend i.e., the database part is developed using MS SQL.
* Delivery Requirement: The whole system is expected to be delivered in four months' time with a two-week evaluation by the project guide.

Software and Hardware Requirements:

Software Requirements:

• Operating system- Windows 10 is used as the operating system as it is stable and supports more features and is more user friendly

• Database MS SQL – MS SQL is used as a database as it is easy to maintain and retrieve records by simple queries which are in the English language which are easy to understand and easy to write.

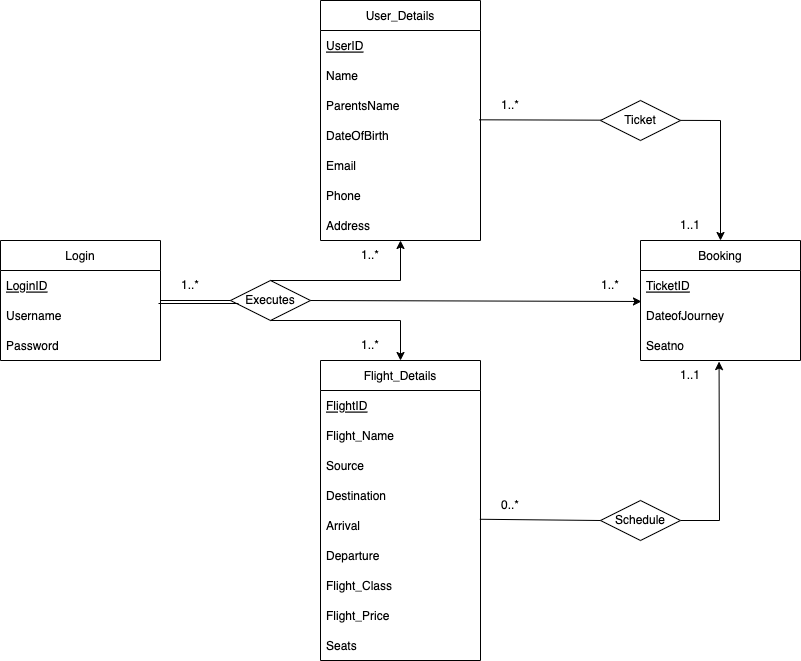
• Development tools and Programming language- The front-end is built using Visual Studio C#

Hardware Requirements:

Intel Core i7 generation is used as a processor because it is fast than other processors and provides reliable and stable and we can run our pc for a long time. By using this processor, we can keep on developing our project without any worries. Ram 1 GB is used as it will provide fast reading and writing capabilities and will in turn support processing.

Methodology:

ER Diagram:



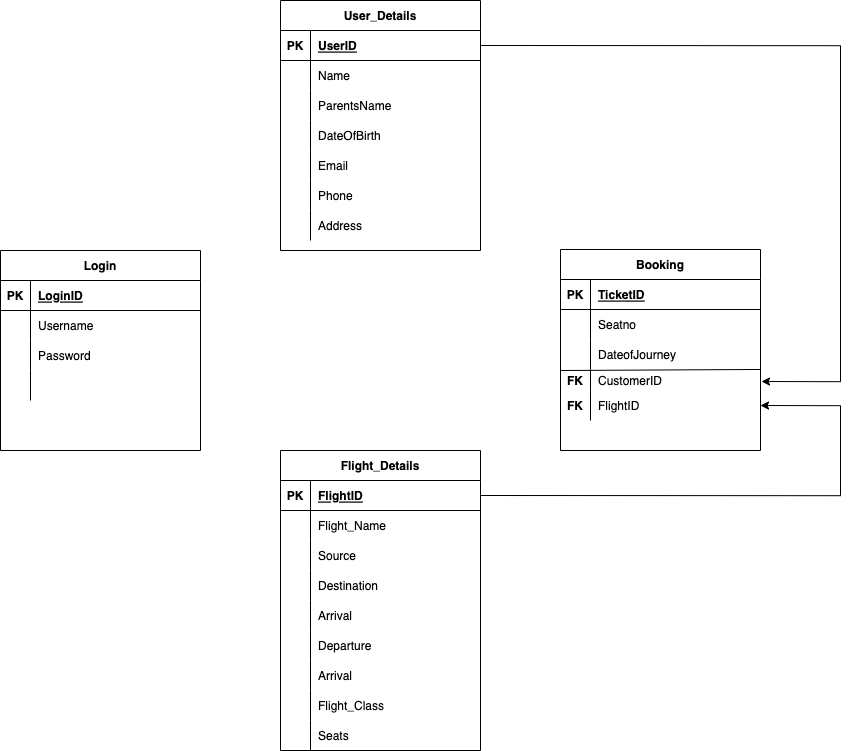
Relations: The ER diagram has 3 relations, which are as follows

* Executes - The other 3 entities can be executed once the login has succeeded.
* Ticket - The Passenger is related to the Booking entity with a Ticket which the passenger receives for travelling.
* Schedule - The flights are related to Booking through its Schedule.

Entity Sets: The ER Diagram of our project possess 4 entity sets which are as follows

* Login: This entity has the login data of all the users who have an account in the system, users the login using their Username and Password.
* User\_Details: This entity has the details of all the passengers with attributes such as Name, Parents Name, DOB etc.
* Flight\_Details: This entity has the details of all the flights with attributes such as Flight Name, Source, Destination, Arrival, Departure etc.
* Booking: This entity deals with booking the tickets and has records of all the tickets that have been reserved.

Schema Diagram:



Normalization:

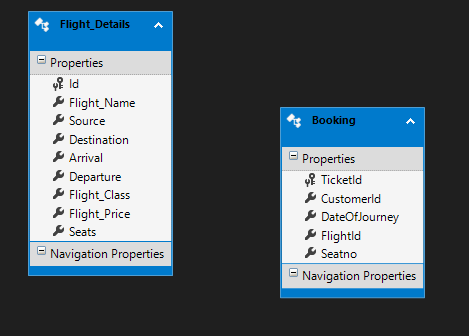
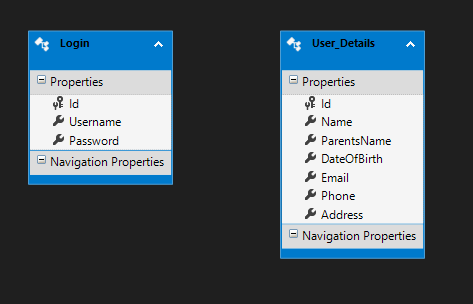
1st Form: All the relations are in 1st form of normalization because it contains atomic values. All the attributes of a relation do not hold multiple values. It holds only single-valued attribute.

2nd Form: All the relations are in 2nd form of normalization because they are in 1st form as well as all non-key attributes are fully functional dependent on the primary key. In fact this implies there is no Partial Dependency.

3rd Form: All the relations are in 3nd form of normalization because they are in 2nd normal form and no transitive dependency exists for a non-prime attribute

Implementation:

Tables: 4 tables are present in the database, which are as follows



Login:

CREATE TABLE [dbo].[Login] (

[Id] INT IDENTITY (1, 1) NOT NULL,

[Username] NVARCHAR (50) NULL,

[Password] NVARCHAR (50) NULL,

PRIMARY KEY CLUSTERED ([Id] ASC)

);

User\_Details:

CREATE TABLE [dbo].[User\_Details] (

[Id] INT IDENTITY (1, 1) NOT NULL,

[Name] NVARCHAR (50) NULL,

[ParentsName] NVARCHAR (50) NULL,

[DateOfBirth] DATETIME NULL,

[Email] NVARCHAR (50) NULL,

[Phone] NVARCHAR (50) NULL,

[Address] NVARCHAR (MAX) NULL,

PRIMARY KEY CLUSTERED ([Id] ASC)

);

Flight\_Details:

CREATE TABLE [dbo].[Flight\_Details] (

[Id] INT IDENTITY (1, 1) NOT NULL,

[Flight\_Name] NVARCHAR (50) NULL,

[Source] NVARCHAR (50) NULL,

[Destination] NVARCHAR (50) NULL,

[Arrival] NVARCHAR (50) NULL,

[Departure] NVARCHAR (50) NULL,

[Flight\_Class] NVARCHAR (50) NULL,

[Flight\_Price] DECIMAL (18) NULL,

[Seats] INT NULL,

PRIMARY KEY CLUSTERED ([Id] ASC)

);

Booking:

CREATE TABLE [dbo].[Booking] (

[TicketId] INT IDENTITY (1000, 1) NOT NULL,

[CustomerId] INT NULL,

[DateOfJourney] DATETIME NULL,

[FlightId] INT NULL,

[Seatno] INT NULL,

PRIMARY KEY CLUSTERED ([TicketId] ASC)

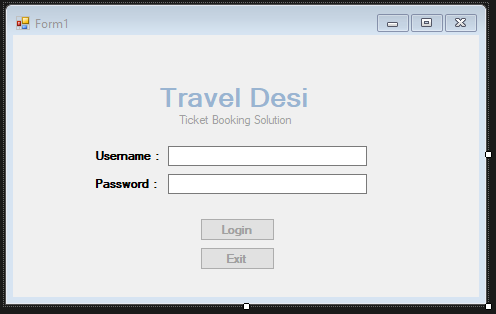
);

Forms:

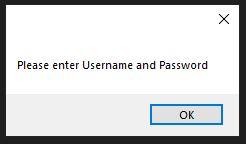
* Form1:
* Mainform:
* Flight\_Information:
* User:
* Search\_Passenger:
* Ticket\_Booking:

Snapshots:

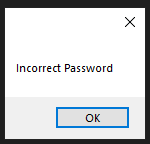
* Login Page – Windows Form for the user to Login



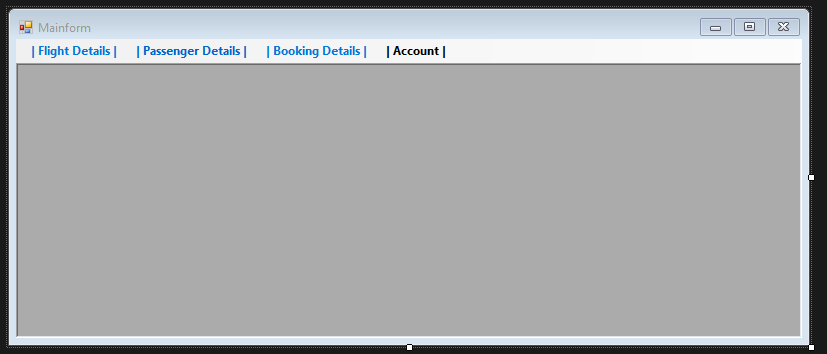
* Dialog Box – In case Username or Password is not entered



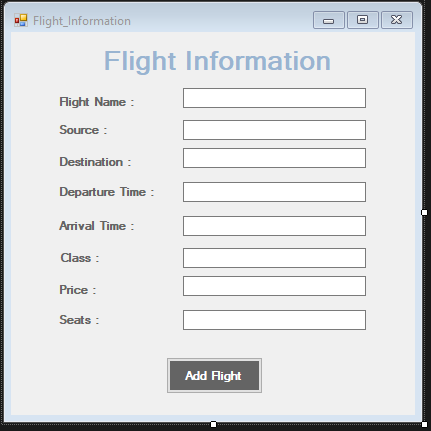
* Dialog Box – In case wrong Password is entered



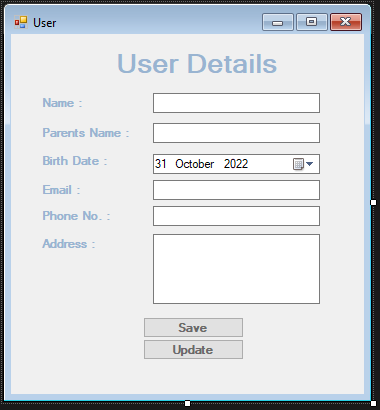
* Main Form – Windows Form through which the user can access all the other functionalities through the Menu Bar. This form is a MdiContainer.

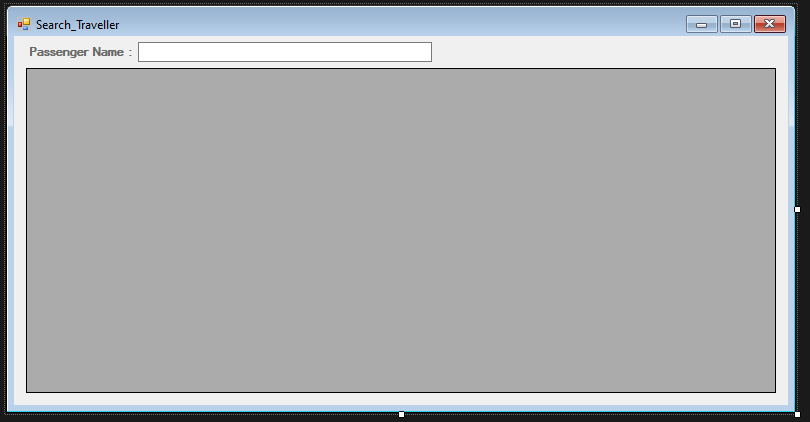


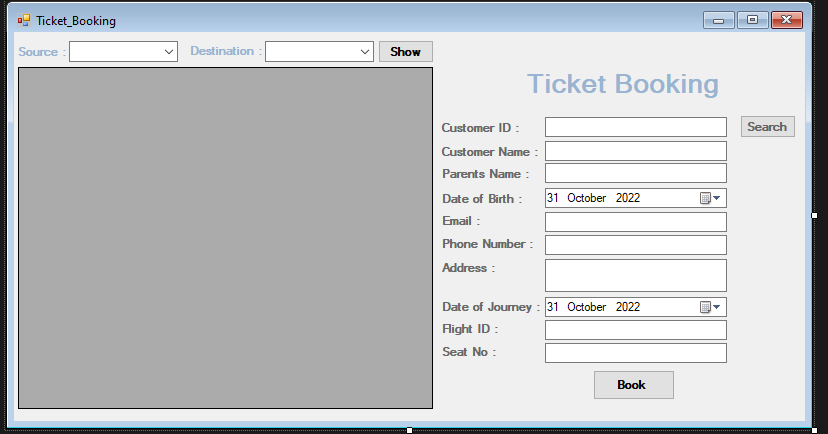
* Flight Information – Windows Form which is used to input details of a new flight



* User Information – Windows Form which is used to input details of a new passenger or update details of an existing passenger



* Search Traveller – Windows Form which is used to search for the details of an existing passenger by using the passenger's name
* Ticket Booking – Windows Form responsible for Booking of tickets for customers



Conclusion:

Travel Desi (Ticket Booking system) project was created as a part of DBMS Lab V sem. The system lets users who have an account the database to login. It can take add flight details, add passenger details, and book tickets. It is an efficient system to make ticket reservation easier. Admins can add flight details, add passengers, and then book tickets. Users can add passengers and then book tickets for them.

Overall, the project was an impeccable learning experience and a good lesson in teamwork for which we are thankful to the faculty.

Mohammed Afnan Althaf Samad – Worked on Frontend, Backend, ER diagram and Report

Ali Murtaza Rizvi – Worked on ER Diagram, Schema Diagram and Tables

**THANK YOU**