INTRODUCTION

PROJECT PROFILE

Airline Reservation System is software which is helpful for ticketing manager as well as the customers. In the later system all the activities were done manually. It was very time consuming and costly. Our Airline Reservation System deals with the various activities related to the Flights.
There are mainly 3 modules in this software

- Flight Reservation module
- Flight Cancellation Module
- Flight Postpone Module

In the Software only user with the legal username and password can sign in. A ticketing manager can book, cancel or postpone any flight for any customer. Flights are booked through Flight Reservation Module in which all the details regarding customer and his flight are entered. A receipt no. is provide to every customer which is unique for each customer and with the help of which cancellation and postpone of flight can be done.

OBJECTIVE

- Our software will perform and fulfill all the tasks that any customer would desire.
- Our software system mainly deals with customers booking and cancelling of flight in the airlines.
- In this ticketing system customer can even postpone their flights.
- The various features added to the project provide all the functions to make the task easy to perform.

SYSTEM ANALYSIS

EXISTING SYSTEM:

System Analysis is a detailed study of the various operations performed by a system and their relationships within and outside of the system. Here the key question is- what all problems exist in the present system? What must be done to solve the problem? Analysis begins when a user or manager begins a study of the program using existing system.

During analysis, data collected on the various files, decision points and transactions handled by the present system. The commonly used tools in the system are Data Flow Diagram, interviews, etc. Training, experience and common sense are required for collection of relevant information needed to develop the system. The success of the system depends largely on how clearly the problem is defined, thoroughly investigated and properly carried out through the choice of solution. A good analysis model should provide not only the mechanisms of problem understanding but also the frame work of the solution. Thus it should be studied thoroughly by collecting data about the system. Then the proposed system should be analyzed thoroughly in accordance with the needs. System analysis can be categorized into four parts.

- ✓ System planning and initial investigation
- ✓ Information Gathering
- ✓ Applying analysis tools for structured analysis
- ✓ Feasibility study
- ✓ Cost/ Benefit analysis.

In the current system we need to keep a number of records related to the customer and want to enter the details of the customer manually. In this system only the ticketing manager views the details of the customer and they can edit the date of flight of the customer or can delete the flight. This is time consuming and has much cost.

PROPOSED SYSTEM

In our proposed system we have the provision for adding the details of the customers by ticketing manager with the help of predefined format and drop-down lists. So the overhead of the ticketing manager is become less. Another advantage of the system is that it is very easy to edit the details of the customers and delete a ticket entry when it found unnecessary.

Our proposed system has several advantages

- User friendly interface
- Fast access to database
- Less error
- More Storage Capacity
- Look and Feel Environment
- Quick transaction

All the manual difficulties in managing the customer details in a airline reservation database have been rectified by implementing computerization.

PROJECT PLANNING

FEASIBILITY ANALYSIS

Whatever we think need not be feasible .It is wise to think about the feasibility of any problem we undertake. Feasibility is the study of impact, which happens in the organization by the development of a system. The impact can be either positive or negative. When the positives nominate the negatives, then the system is considered feasible. Here the feasibility study can be performed in two ways such as technical feasibility and Economical Feasibility.

Technical Feasibility

We can strongly says that it is technically feasible, since there will not be much difficulty in getting required resources for the development and maintaining the system as well. All the resources needed for the development of the software as well as the maintenance of the same is available in the organization here we are utilizing the resources which are available already.

Economical Feasibility:

Development of this application is highly economically feasible .The organization needed not spend much money for the development of the system already available. The only thing is to be done is making an environment for the development with an effective supervision. If we are doing so, we can attain the maximum usability of the corresponding resources .Even after the development, the organization will not be in condition to invest more in the organization .Therefore, the system is economically feasible.

HARDWARE CONFIGURATION

Processor	**:**	**Pentium III 630MHz**
RAM	**:**	**128 MB**
Hard Disk	**:**	**20GB**
Monitor	**:**	**15” Color monitor**
Key Board	**:**	**122 Keys**

SOFTWARE CONFIGURATION

Operating System : Windows NT,
Windows 98,
Windows XP.

Language : Java 2 Runtime Environment

Database : MS Access2003.

SYSTEM REQUIREMENTS

This management system can be used in windows 98, Windows2000, Windows XP and Windows NT, supported for other platform such as Macintosh and UNIX.

The system must be running Windows 98, Windows 98 or Windows NT4.0 operating system and must meet the following hardware requirements.

- For Windows 95 based computers , a 486 / 66 MHz or higher processor with 8MB
- For Windows 98 based computers , a 500/88MHz or higher processor with 32 Mb of RAM
- For Windows NT based computers , a 488 / 66 MHz or higher processor with 16 MB of RAM
- For Windows 200 based computers , a 700/850 MHz or higher processor with 512 MB of Ram

SYSTEM DESIGN

JAVA

The entire component has been developed using Java technology. Java has been chosen as the platform because of its feature rich nature. The Java Platform provides robust end-to-end solutions for networked applications as well as a trusted standard for embedded applications. So Java was a natural choice for development process.

Characteristics of Java

Object Oriented:

Java is object oriented to the truest sense of the word. Everything in Java is represented as objects. Variables and methods both are encapsulated in objects. Java is the purest object-oriented language.

Robust:

Java is a very robust language owing to the following features:

- Excellent exception handling facilities.
- Memory management relief for the user. User does not have to worry about allocation and de-allocation of memory.
- Strict compile-time and runtime checks for data types.

Portable and Architecture-neutral (Platform Independent):

Java is portable and platform independent so much that they satisfy "write once; run anywhere, anytime, forever". This feature is implemented in the following ways:

- Compiler generates machine independent byte-code instructions which can be run on any machine supporting **Java Virtual Machine**.
- Size of primitive data type is machine independent.

Multithreaded:

- Programs can do many things simultaneously using different threads.
- Provides a solution for multi-process synchronization.
- Allows the creation of networked and interactive programs.

Distributed:

- Open access to remote objects by the use of RMI(Remote Method Invocation).
- Brings a level of abstraction to client/server programming.

Secure:

- Security is achieved by confining a java program to the java execution environment and not allowing access to other parts of the user computer.
- Absence of pointers provides memory related security as encroachment of memory is avoided Proper measures for prevention of viral infection and malicious intent.

Dynamic and Extensible:

- Facilitates linking in of new classes, objects and methods.
- Supports native methods (methods written in other languages like C, C++).
- Programs carry with them a substantial amount of runtime type information that is used to verify and resolve accesses to objects at run-time.

High Performance:

Just-In-Time (JIT) compilers are used to convert byte-code into native machine code resulting in very high performance. These JIT compilers can be used on a real time, piece by piece demand basis to perform on-the-fly compilation of byte-code into native-code.

Compilation and Interpretation

Java programs are implemented as a two-stage system.

Compilation: Source code to byte-code and not machine instructions.

Interpretation: Byte-code to machine code (for any system that supports using JVM)

Thus cross-platform programs can be written.

JAVA SWINGS

"Swing" refers to the new library of GUI controls (buttons, sliders, checkboxes, etc.) that replaces the somewhat weak and inflexible AWT controls.

The Swing classes eliminate Java's biggest weakness: its relatively primitive user interface toolkit. Java Swing helps you to take full advantage of the Swing classes, providing detailed descriptions of every class and interface in the key Swing packages. It shows you how to use all of the new components, allowing you to build state-of-the-art user interfaces and giving you the context you need to understand what you're doing. It's more than documentation; *Java Swing* helps you develop code quickly and effectively.

1. Main New Features

- **Lightweight**. Not built on native window-system windows.
- **Much bigger set of built-in controls**: Trees, image buttons, tabbed panes, sliders, toolbars, color choosers, tables, text areas to display HTML or RTF, etc.
- **Much more customizable**. Can change border, text alignment, or add image to almost any control. Can customize how minor features are drawn. Can separate internal representation from visual appearance.
- **"Pluggable" look and feel**. Can change look and feel at runtime, or design own look and feel.
- **Many miscellaneous new features**. Double-buffering built in, tool tips, dock able tool bars, keyboard accelerators, custom cursors, etc.

2. Components are named JXxx.

E.g. JFrame, JPanel, JApplet, JDialog, JButton, etc.

J Component

The J Component class is the root of the Visual component class hierarchy in JFC. All Swing components are implemented as subclass of J components class, which inherits from the Container class. Swing component inherit the following functionality from J Component

a). Borders- Using the setBorder() method, you can specify the border that a component displays around its edges. You can specify that component have extra space around its edges using an EmptyBorder instance.

b). Double Buffering- It can improve the appearance of frequently changing components. Now you do not have to write the double buffering code because Swing provides it for you. By default Swing components are double buffered.

c). Tool Tips- By specifying a string with the setToolTipNext() method, you can provide help to users of a components. When the cursor pauses over the components, the specified string is displayed in small window near the component.

d). Look and Feel- Subject to the security restriction, you can choose the look and feel used by all Swing components by invoking the UIManager.setLookAndFeel() method.

3. There is an almost-equivalent Swing component for most AWT components.

4. Instead of adding components directly to frames or applets, we can use the content pane.

- Add to content pane via `getContentPane().add`
- Replace content pane via `setContentPane`

5. Model-View-Controller architecture let us change the internal data representation for lists, trees, tables, etc.

6. Swing was in the com.sun.java.swing package in beta releases of 1.2. Switched to javax.swing in 1.2 final.

7. Default "look and feel" is a Java-specific one.

- Need special call to get native look
- Default called "Java look & feel"

8. Mixing AWT and Swing is doomed.

AWT components are always on top, and z-ordering problems catch you in many unexpected ways. Stick with the AWT or move completely to Swing.

JDBC

The JDBC is a set of the database access classes. The very term JDBC stands for "Java Database Connectivity". It was developed by Java Soft. JDBC technology is an API (Application Program Interface) that allows virtual access to any tabular data source from the Java programming language by means of some connecting software called Drivers. It provides cross-DBMS connectivity to a wide range of SQL databases. JDBC defines a set of interfaces to enable developers to access data independently of the actual database product used to store the data. JDBC allow Java applets, Servlets, and application to access data in famous database management systems.

It also provides access to other tabular data sources, such as spreadsheets or flat files. The JDBC API allows developers to take advantage of the Java platform's "Write Once, Run Anywhere" capabilities for industrial strength, cross-platform applications that require access to enterprise data. With a JDBC technology-enabled driver, a developer can easily connect all corporate data even in a heterogeneous environment. The JDBC API is the industry standard for database-independent connectivity between the Java programming language and a wide range of databases. The JDBC API makes it possible to do three things:

- Establish a connection with a database or access any tabular data source
- Send SQL statements
- Process the results

Steps in using JDBC

1) Create a Connection type of object (A) denoting a connection to the database.
2) Create a Statement type of object (B) using the A.
3) Use B to execute either update the database or send a query request.
4) The result of the query operation in step 3 is a Result Set type of object(C)
5) C is actually a small table (D) consisting of the result of the query.
6) D can be handled according to the user needs
7) Close C,B and A

JDBC ARCHITECTURE

The JDBC API contains two major sets of interfaces:

- JDBC API for application writers
- Lower-level JDBC driver API for driver writers

JDBC drivers are divided into four types or levels. Each type defines a JDBC driver implementation with increasingly higher levels of platform independence, performance, deployment administration. The four types are:

- Type 1: JDBC-ODBC Bridge
- Type 2: Native-API/partly Java driver
- Type 3: Net-protocol/all-Java driver
- Type 4: Native-protocol/all-Java driver

Type1:JDBC-ODBC Bridge

The type 1 driver, JDBC-ODBC Bridge, translates all JDBC calls into ODBC (Open Database Connectivity) calls and sends them to the ODBC driver. As such, the ODBC driver, as well as, the client database code, must be present on the client machine.

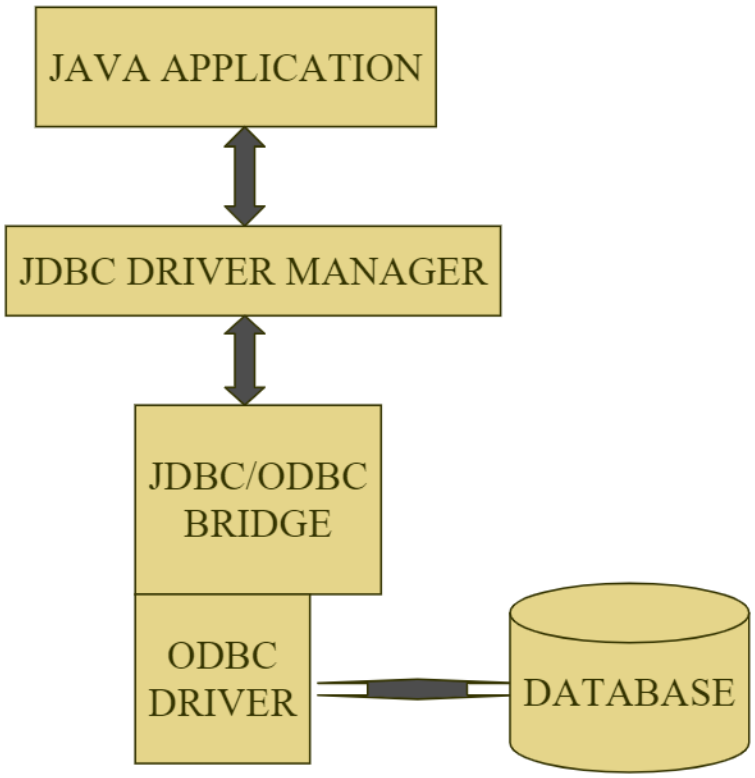


Figure 1. Type 1: JDBC-ODBC Bridge

Pros: The JDBC-ODBC Bridge allows access to almost any database, since the database's ODBC drivers are already available. Type 1 drivers may be useful for those companies that have an ODBC driver already installed on client machines.
Cons: The performance is degraded since the JDBC call goes through the bridge to the ODBC driver, then to the native database connectivity interface. The result comes back through the reverse process. Considering the performance issue, type 1 drivers may not be suitable for large-scale applications.

The ODBC driver and native connectivity interface must already be installed on the client machine. Thus any advantage of using Java applets in an intranet environment is lost, since the deployment problems of traditional applications remain. We have used JDBC-ODBC Bridge Type Drivers.

INPUT DESIGN

Input design is the process of converting user-oriented input to a computer based format. Input design is a part of overall system design, which requires very careful attention .Often the collection of input data is the most expensive part of the system. The main objectives of the input design are …

1. Produce cost effective method of input
2. Achieve highest possible level of accuracy
3. Ensure that the input is acceptable to and understood by the staff.

Input Data

The goal of designing input data is to make entry easy, logical and free from errors as possible. The entering data entry operators need to know the allocated space for each field; field sequence and which must match with that in the source document. The format in which the data fields are entered should be given in the input form .Here data entry is offline; it makes use of processor that accepts commands and data from the operator through a key board. The input required is analyzed by the processor. It is then accepted or rejected. Input stages include the following processes

- Data Recording
- Data Transcription
- Data Conversion
- Data Verification
- Data Control
- Data Transmission
- Data Correction

One of the aims of the system analyst must be to select data capture method and devices, which reduce the number of stages so as to reduce both the changes of errors and the cost .Input types, can be characterized as.

- External
- Internal
- Operational
- Computerized
- Interactive

Input files can exist in document form before being input to the computer. Input design is rather complex since it involves procedures for capturing data as well as inputting it to the computer.

OUTPUT DESIGN

Outputs from computer systems are required primarily to communicate the results of processing to users. They are also used to provide a permanent copy of these result for latter consultation .Computer output is the most important and direct source of information to the users. Designing computer output should proceed in an organized well throughout the manner. The right output must be available for the people who find the system easy o use. The outputs have been defined during the logical design stage. If not, they should defined at the beginning of the output designing terms of types of output connect, format, response etc,

Various types of outputs are

- External outputs
- Internal outputs
- Operational outputs
- Interactive outputs
- Turn around outputs

All screens are informative and interactive in such a way that the user can full fill his requirements through asking queries.

DATABASE DESIGN

The general theme behind a database is to handle information as an integrated whole. A database is a collection of interrelated data stored with minimum redundancy to serve many users quickly and effectively. After designing input and output, the analyst must concentrate on database design or how data should be organized around user requirements. The general objective is to make information access, easy quick, inexpensive and flexible for other users. During database design the following objectives are concerned:-

- Controlled Redundancy
- Data independence
- Accurate and integrating
- More information at low cost
- Recovery from failure
- Privacy and security
- Performance
- Ease of learning and use

TABLES USED

Tab1

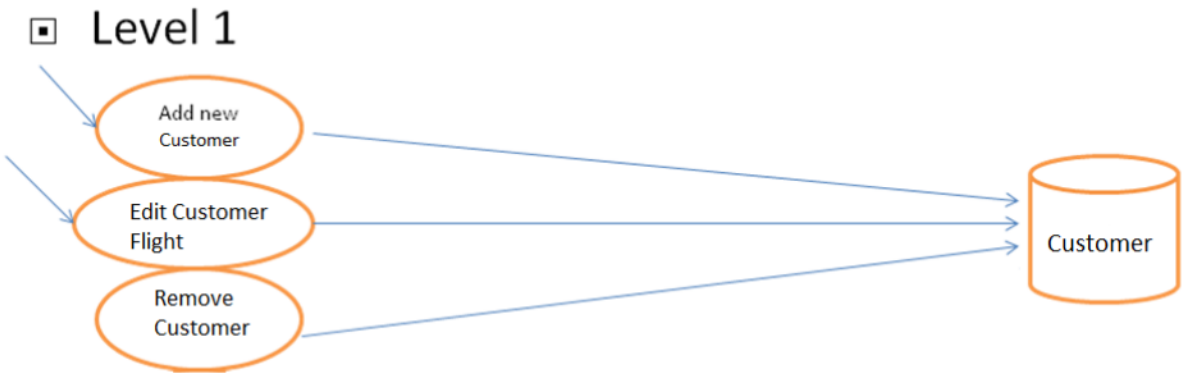
Field Name	Data Type	Description
ReciptNo	Text(50)	Primary Key
Name	Text(50)	-
Airlines	Text(15)	-
Class	Text(10)	-
Age	Text(50)	-
From	Text(50)	-
To	Text(50)	-
Date	Text(50)	-
Sex	Text(50)	-

Tab2

Field Name	Data Type	Description
Username	Text(25)	Primary Key
Password	Text(15)	-

DATA FLOW DIAGRAM

Context Diagram



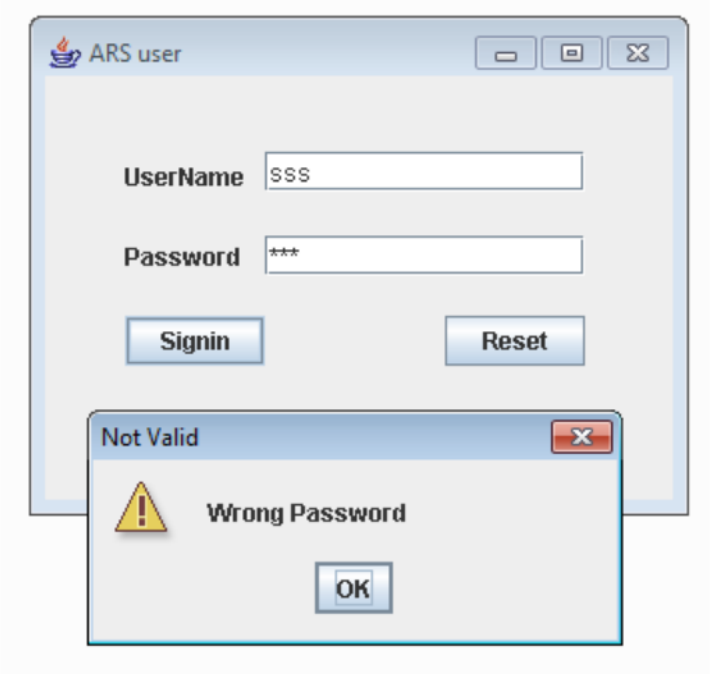
SCREEN LAYOUT

Login:

This is the login window, here the ticketing manager enter the username and password. Only the ticketing manager has the username and can access the software system.

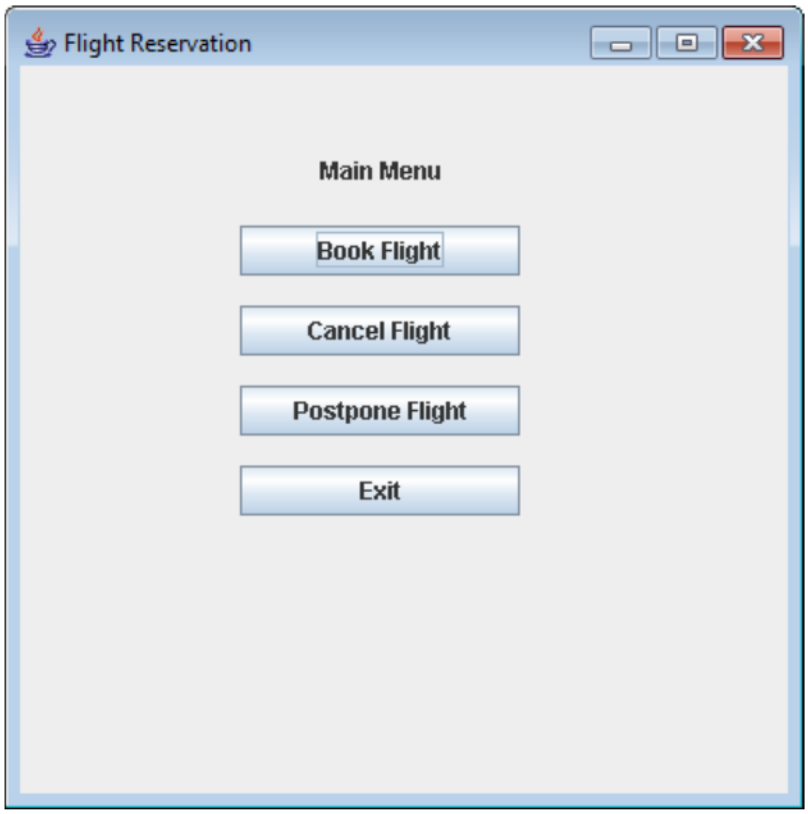
On Wrong password

If the ticketing manager enters the wrong password a new window pops out telling that you have entered wrong password.



Main Menu

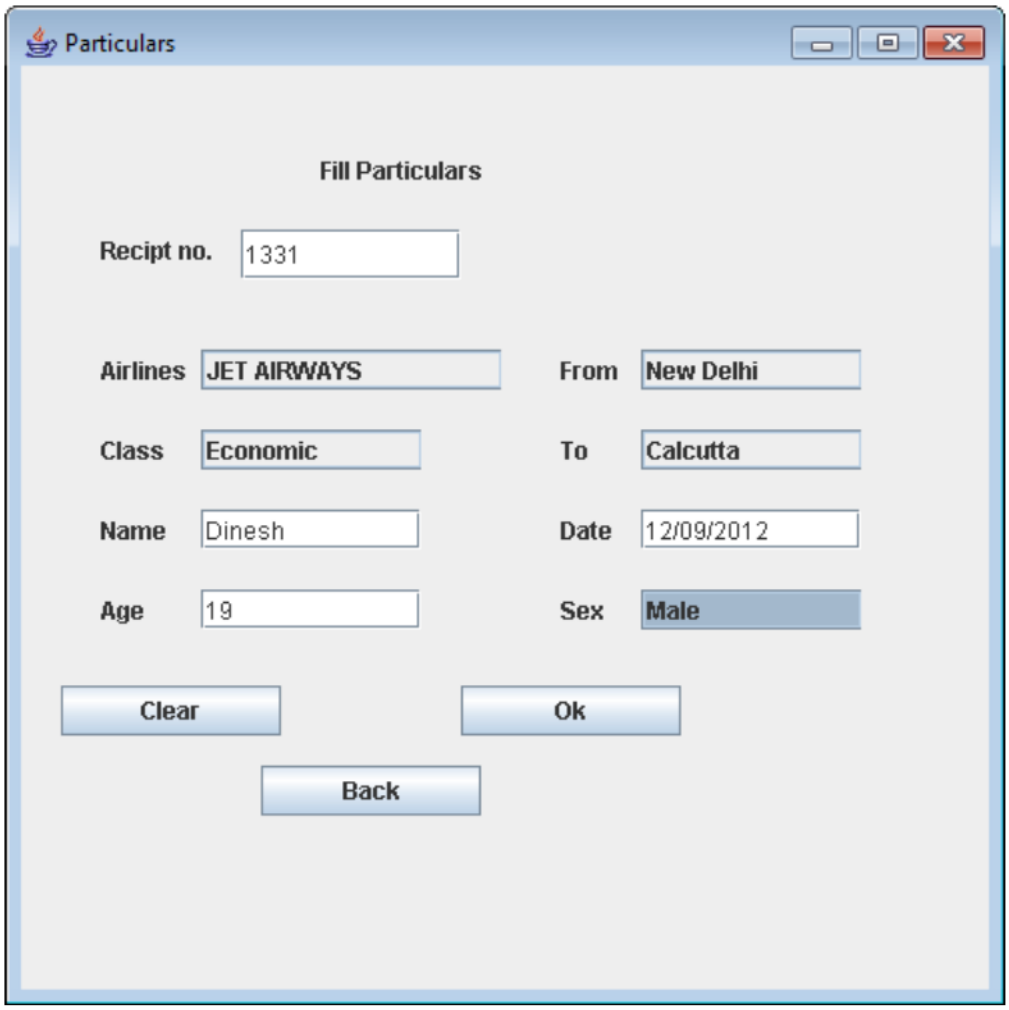
The main menu consist of four options for the ticketing manager, were the manager can Book a flight, cancel a flight and postpone a flight according to the customer requirements. Last option is the Exit.



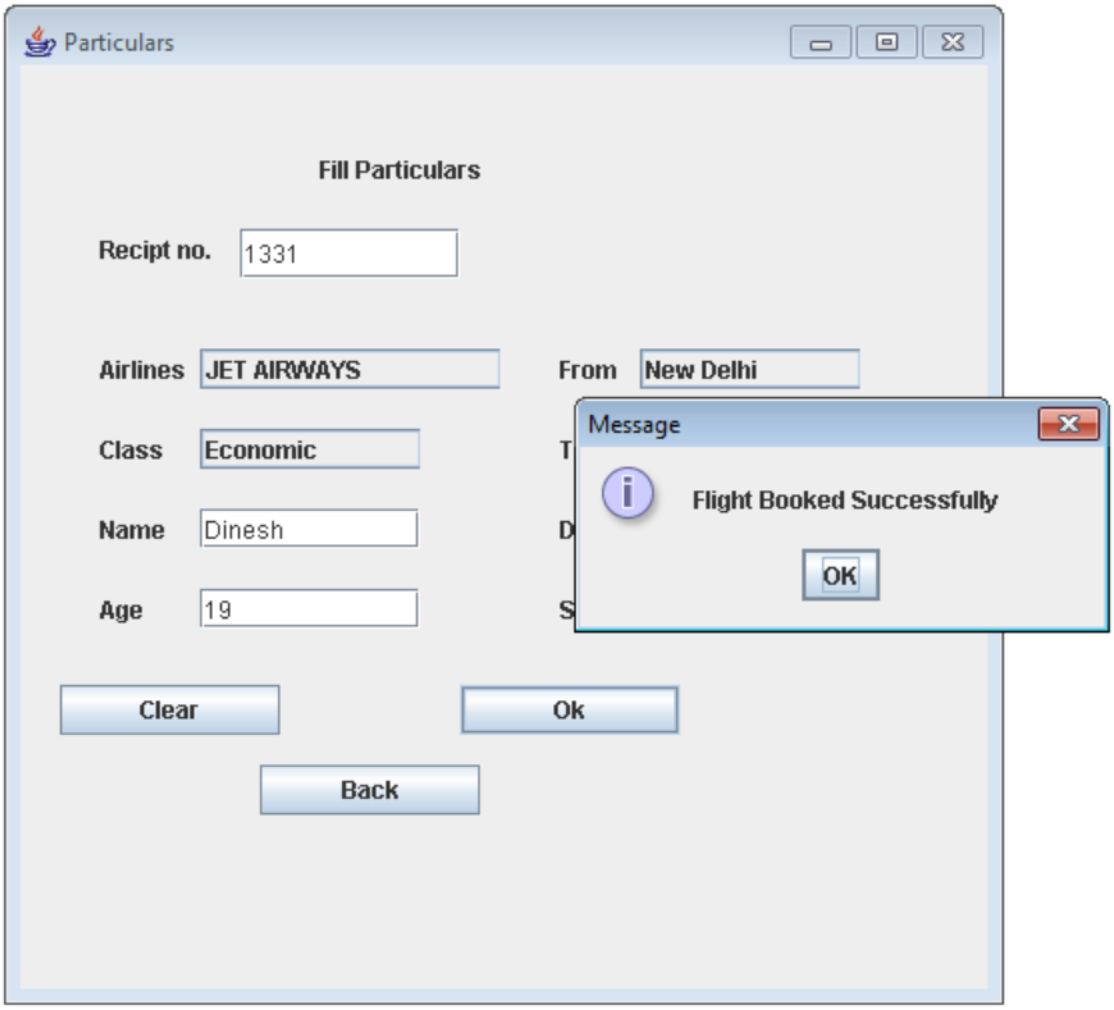
Customer Reservation

In customer reservation window, we can see different option .the ticketing manager is required to fill these selected option.

First the ticketing manager enters a receipt no. and fills the required option according to The customer need like airlines, destination, class, date, name, age. sex.

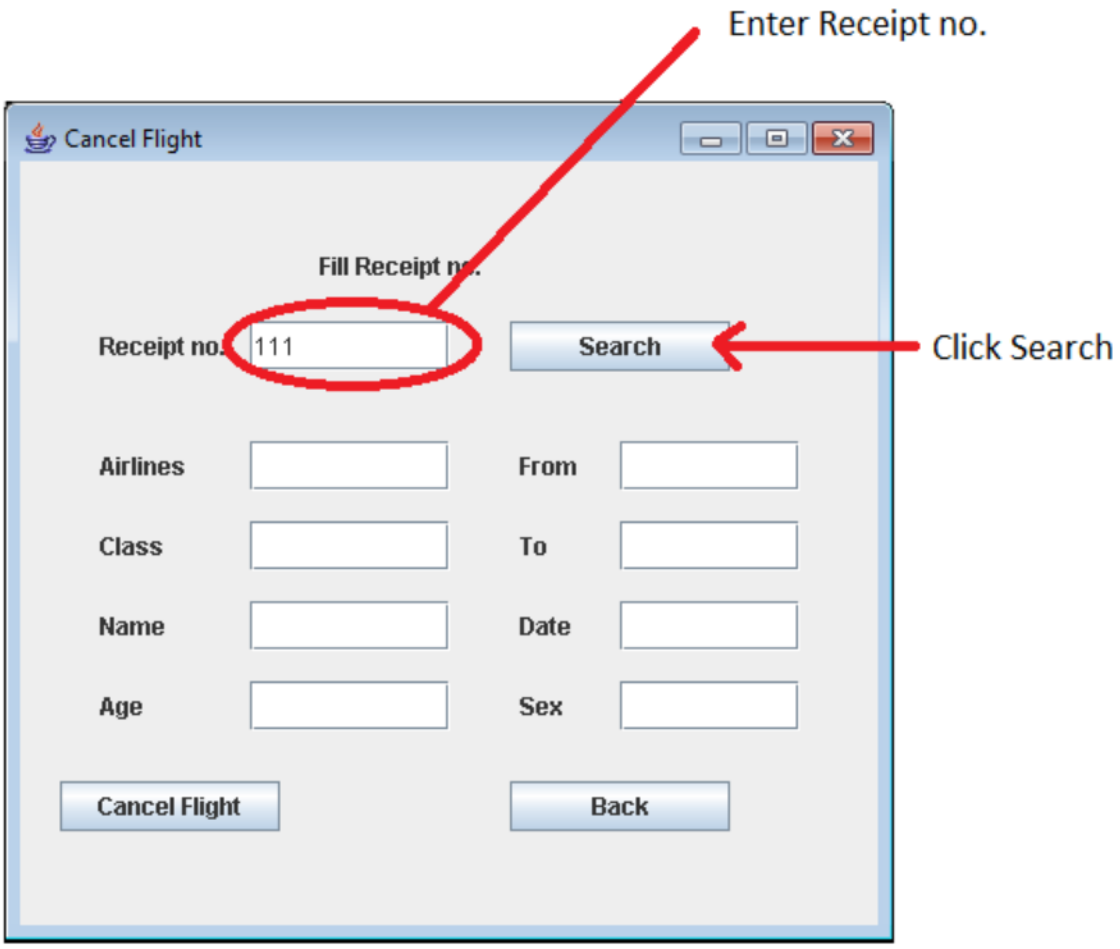


Flight Reservation

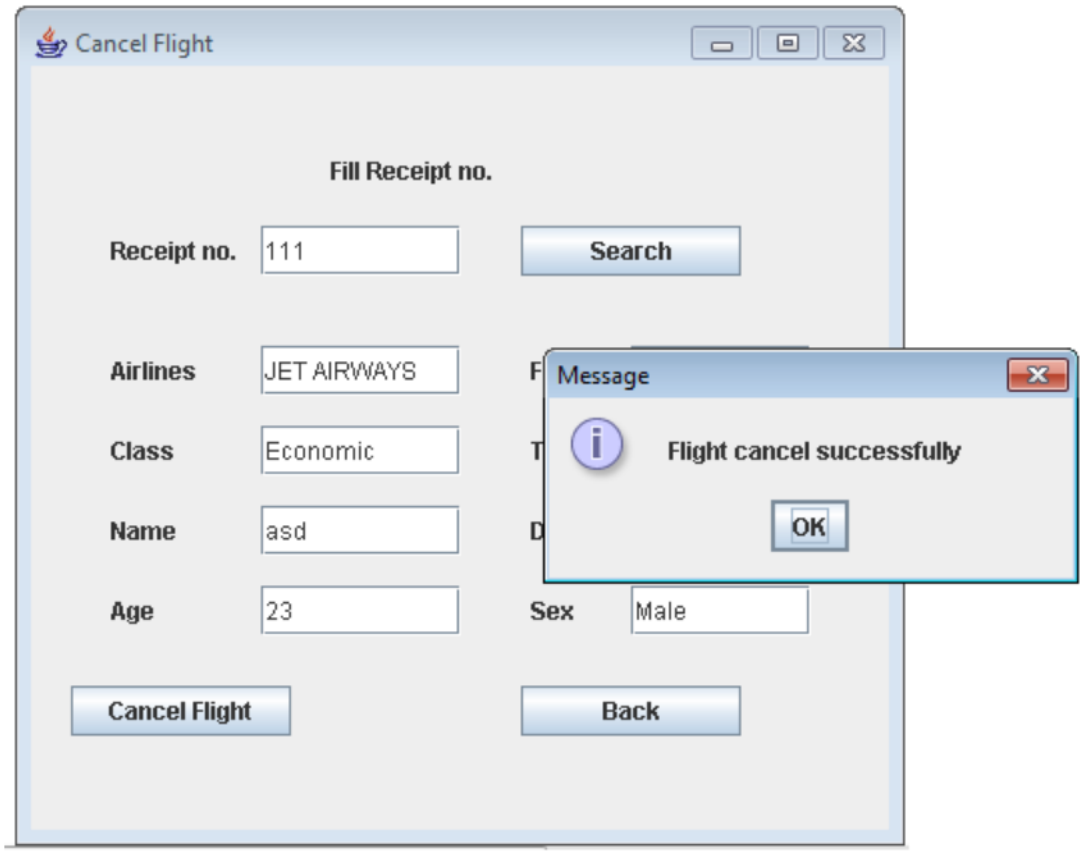


Flight Cancellation

Search

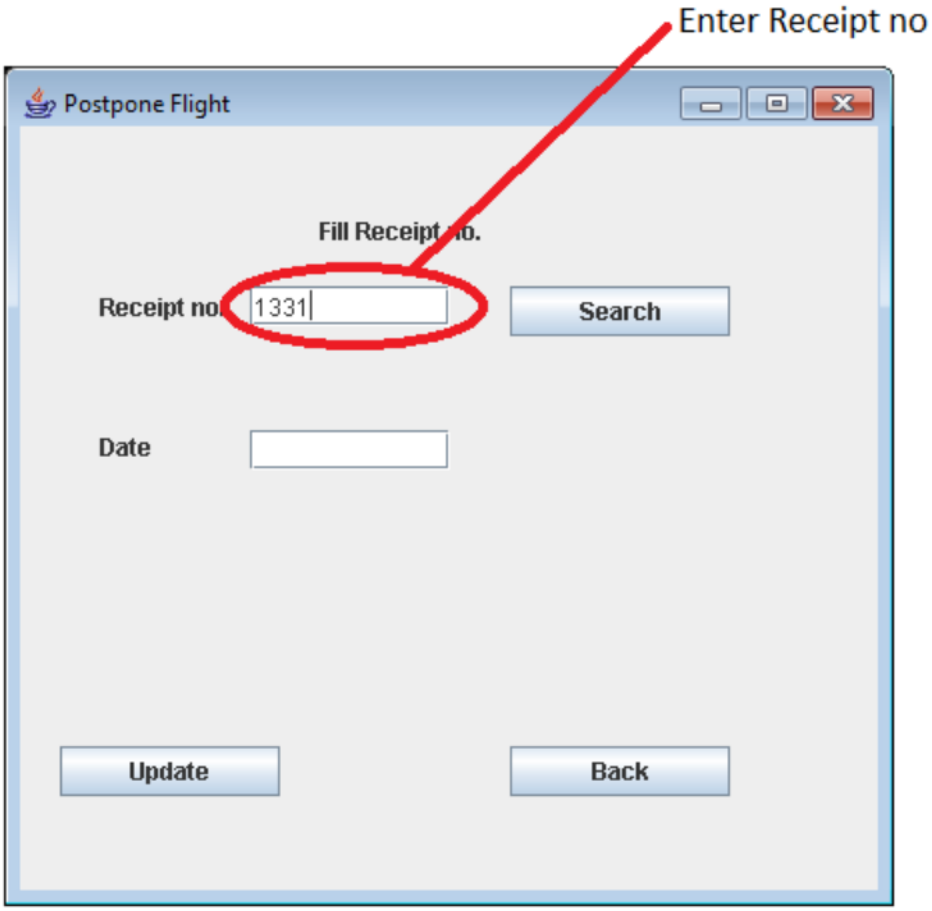


Cancel Flight

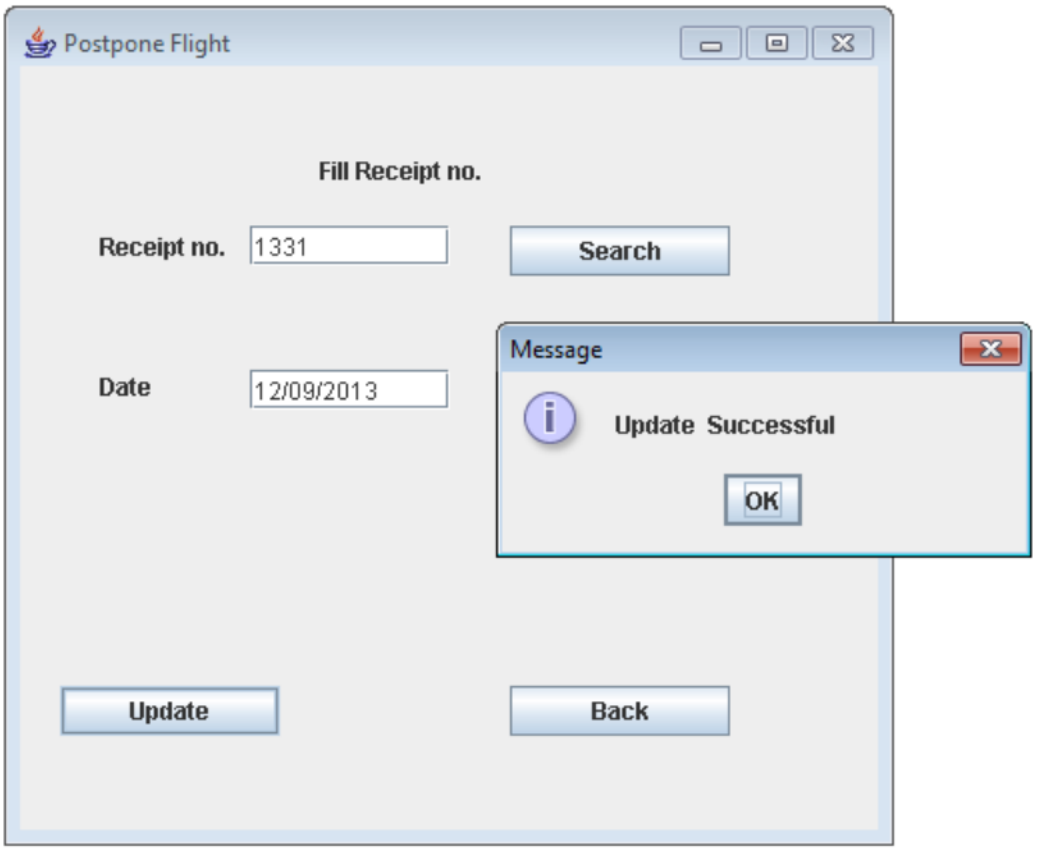


Postpone Flight

Search



Update


Postpone Flight
Fill Receipt no.
Receipt no.
1331
Search
Date
12/09/2013
Message
Update Successful
OK
Update
Back

SYSTEM IMPLEMENTATION

Implementation is the stage in the project where the theoretical design is turned into a working system. The implementation phase constructs, installs and operates the new system. The most crucial stage in achieving a new successful system is that it will work efficiently and effectively.

There are several activities involved while implementing a new project. They are

- End user training
- End user Education
- Training on the application software
- System Design
- Parallel Run and To New System
- Post implementation Review

End user Training:

The successful implementation of the new system will purely upon the involvement of the officers working in that department. The officers will be imparted the necessary training on the new technology

End User Education:

The education of the end user start after the implementation and testing is over. When the system is found to be more difficult to understand and complex, more effort is put to educate the end used to make them aware of the system, giving them lectures about the new system and providing them necessary documents and materials about how the system can do this.

Training of application software:

After providing the necessary basic training on the computer awareness, the users will have to be trained upon the new system such as the screen flows and screen design type of help on the screen, type of errors while entering the data, the corresponding

validation check at each entry and the way to correct the data entered. It should then cover information needed by the specific user or group to use the system.

Post Implementation View:

The department is planning a method to know the states of t he past implementation process. For that regular meeting will be arranged by the concerned officers about the implementation problem and success.

CONCLUSION

Based on the preparation & analysis of Airline Reservation System, it can be concluded that Airline Reservation system is better than the manual System can be used by the ticketing manager to make transactions for the customers regarding flights.

The objective of software planning is to provide a frame work that enables the manger to make reasonable estimates made within a limited time frame at the beginning of the software project and should be updated regularly as the project progresses.

BIBLIOGRAPHY

1) http://www.javaworld.com/javaworld/jw-01-1998/jw-01-bookreview.html
2) *Database Programming with JDBC and Java by O'Reilly*
3) Head First Java 2nd Edition
4) http://www.jdbc-tutorial.com/
5) Java and Software Design Concepts by APress