CS 352

Database Management Systems

TERM PROJECT

FINAL REPORT

May 13, 2011

Airline Company Data Management System

Group 43

Ahmet Alp Balkan	Ecem Ünal	Emre Ekmekçi
20800978	20702525	20800227
a_balkan@ug	ecem_u@ug	e_ekmekci@ug

Table of Contents

Project Description

In this project, we designed and implemented a database management system for airline companies. The system is mainly a web application that is used by the passengers of the Airline Company, and administrative & sales staff. Such a database management system is necessary for the companies in order to keep track of their flights, planes, sales and crew.

There are two main user of the system, which makes use of different aspects of the application. The customers of the airline company use the system to see all upcoming flights (or specific flights chosen by themselves), to select a flight and to buy a ticket for the selected flight. The customers need to be logged on to the system to perform all of the operations specified, and if they do not have an account, they can easily sign up.

The sales staff of the company uses the system to perform check-in operations of the passengers. They can check-in passengers to flights whose flight IDs are known using

passengers' seat number, and while checking in they can associate luggage with the passengers. When the operation is complete, they can print the flight cards. To complete these operations, staff must be logged on to the system.

Executives have many roles in the system mainly consisting of managing employees (both sales staff and crew), flights, airports, cities, fleet and reporting the current situation of the company. Management of employees includes adding new employees to the system, updating their details such as their salaries and types. In addition to that, administrative staff can add new cities to database and edit their names. They can also add new airports and associate them with one of the cities that exist in the database, and update the information of an existing airport such as its code or its name. Executives can add new planes and edit their name, model and capacity information. After adding airports, planes and crew to the system, executives can plan new routes and edit their details including duration of the route, departure and arrival points. Whenever a route exists in the system, they may now add a new flight using the existing route and adding departure date and time, flight fare, plane, and captain pilot, assistant pilot and hostess information. This information can be updated anytime before the flight takes place. Also executives have the permission to delete employees, planes, cities, airports, routes or flights from the system permanently.

There also exist reports about flights, payments, tickets sold over time, staff and their salaries, monthly revenues, which are available to only executives. In order to perform all of these operations, executives must be logged on to the system.

List Of Tables

Entities

- city
- airport
- route
- plane
- ground_staff
- flight_staff
- customer
- flight
- payment
- luggage

Relations

- flight_crew
- booking
- booking_luggage

User's Manual

For Customers

In order to see upcoming flights or buy tickets as the passengers of the airline company, users must have a valid account in our web application. If they are not a member yet, they can sign-up by clicking “New Customer?” link located on the main screen. After being directed on the sign-up page, users must pick a unique username and password for them and they must specify their telephone numbers. Once they entered all necessary information to the system, they can complete their registration by clicking “Proceed” button. All registered users can sign-in in the main screen by using the information they provided while registering. After signing-in to the system by entering their username and password and pressing “Proceed”, all customers must see a flight selection screen now. They now have two options; clicking “See all upcoming flights” link shows all the flights

the company has from now on. The other option is selecting a range for departure date and departure/destination cities and pressing continue. If all the entered inputs are valid, then the customer must see some flights according to the availability of flights.

After listing all possible flights in either selection, the customer can select a flight to buy a ticket to it by checking the “Buy” option and clicking “Continue” button. If there are seats available, customer now can enter their credit card information and buy ticket. They should keep their seat number in order to check-in to the system before the flight happens. In any stage the customer have the right to return back to the main screen by clicking “Airline Company” link located on the top of the site.

For Sales Staff

Sales staff can only login by the username and password given by administrative staff. After logging in to the system using their login info, salesman should see a dashboard screen where they can perform one of the two options: one is listing the details of any scheduled flight, and the other is checking-in a customer where the number of the flight and the seat of the customer to be checked-in is known.

If sales staff selects a flight number from the list and clicks on the “Generate Report” button, they may see the route, departure date, captain and assistant pilots and the hostess of the flight as well as the customers who bought ticket for this flight with their seat number.

If there is a customer who wants to be checked-in to a flight with or without their luggage, sales staff must select the number of flight from the list and provide the seat number of the customer. If the customer has luggage(s) then salesman should also provide their heights in metric kilograms. After entering all necessary information to the system and pressing “Check-in” button, the travel pass of the customer is shown on the screen. The travel pass includes the name and the seat number of the passenger, the route, departure date and time of the flight and the heights of the luggage of the passenger. This pass should be printed and given to passenger to be used by him/her while heading off the plane.

For Executive Staff

Administrative staff has mainly two options: managing the system and reporting the situation of the company. After signing in to the system, all of the options that executives can perform have corresponding links on the main screen. Clicking “Cities”, “Airports”, “Airplanes”, “Routes”, “Flights” or “Crew” link leads to corresponding manage screen where executives can add new cities, airports, airplanes, routes, flights or crew, modify existing ones or delete them. While adding or modifying, they should meet the requirements of them appropriately.

For example adding a new flight with a date before now is not a valid flight, hence cannot be added to the system. Or creating a route between same cities is not valid; therefore it should not be accepted as well. Same rules follow while modifying existing ones.

Reporting has three options: “Average Staff Salary Report”, “Fullness Report of Upcoming Flights” and “Total Revenue per Month”. To learn the average salaries that each staff type get, executives should choose average staff salary report. The report computes averages for executives, pilots, and hostesses and sales staff separately. Fullness report lists all of the upcoming flights and number of tickets sold for all of the flights with a fullness rate. Also there is a revenue report for executives to see total monthly incomes and expenses of the company. Pressing “Print Report” button located on the bottom of the screen can print reports out.

Implementation Details

Back-end

In local test environment of the project, we have used PHP 5.1.2 and MySQL 5.5.9. However in the deployment environment, there is PHP 5.2.12 and MySQL 5.0.51a. Our server is located at Florida, US.

The reason we choose PHP and MySQL is they can cooperate very easily since PHP has built-in support for MySQL and we have previous know-how on this area. PHP is a strong language for scripting such these proof-of-concept purposes, since it does not compile; just modify and upload via FTP. On the other hand MySQL is a very handy database and we prefer it since it has InnoDB engine, which provides foreign keys, indexes and many

constraints in high level.

We did not use any other libraries or frameworks to provide database connection or executing queries. Our project is written in pure PHP. We have used command line mysql, mysqldump tools and Sequel, phpMyAdmin high level MySQL management tools.

Front-end

The user interface of our project is a web page interface, which is coded with HTML 5 and CSS 3. It supports recently released browser versions supporting HTML 5 and JavaScript.

We use jQuery JavaScript framework to handle many user interface interactions, events and form integrity checks. jQuery is a very handy off-the-shelf JavaScript framework to cope with DOM objects and animations on the page. We used it to perform field controls on user-filled forms and user interface transitions.

Licences

jQuery is licensed under GNU General Public License v2.
MySQL is licensed under GNU General Public License.
PHP is licensed under The PHP License v3.01 http://www.php.net/license/3_01.txt

Advanced Database Features Used

Secondary Indexes

We have indexes on many fields to provide search operations quicker and less expensive on the database tier. We have HASH index on equality comparison fields and BTREE index on range query fields.

In MySQL, foreign key and primary key fields are indexed, by default.

```
CREATE INDEX city_name ON city(name(5)) USING BTREE
CREATE INDEX route_points ON route(departure,destination) USING HASH;
CREATE INDEX route_flights ON flight(route) USING HASH;
CREATE INDEX route_dates ON flight(flight_date) USING BTREE; //range
CREATE INDEX staff_credentials ON ground_staff(name,passowrd) USING HASH;
CREATE INDEX customer_credentials ON customer(name,password) USING HASH;
```

Views

We have a view to lists departure and destination airports of routes separated by a dash (-) along with their id. Used for representational listings in the GUI.

```
CREATE VIEW route_listing AS
(SELECT r.id, CONCAT(f.name, '-', t.name) AS route_name FROM route r, airport f,
airport t WHERE r.departure=f.id AND r.destination=t.id);
```

Results:

id	route_name	departure	destination	duration
8	Ataturk-Esenboga	9	7	120
16	Charles De Gaulle-Esenboga	10	7	195
27	Sabiha Gokcen-Esenboga	11	7	45
7	Esenboga-Ataturk	7	9	120
15	Charles De Gaulle-Ataturk	10	9	180
13	Ataturk-Heathrow	9	12	190
17	Charles De Gaulle-Heathrow	10	12	75
28	Sabiha Gokcen-Heathrow	11	12	190

Triggers

In prior versions of MySQL, there were CHECK statements allowed in DDL queries to check constraints on fields upon insertion. However they are replaced with triggers in MySQL 5. We have the following check operations implemented with triggers (SQLs omitted here):

- **route:** departure and arrival fields cannot be the same.
- **flight:** departure_date field cannot be in the past.
- **booking:** booking_date field cannot be in the past. Cannot make booking to a past date.