Technological University of the Philippines

Ayala Boulevard, Ermita, Manila

**TEAM PROJECT**

**Information Management**

Submitted by:

**LEADER:**

BALANGA, KRISTAN LLOYD B.

**ASST. LEADER:**

AVELLANA, DOROTHEA

**MEMBERS:**

CAYCO, SHEKINAH MARIE D.V

LEONARDO, NYL RAEMON S.

**2nd Semester**

**S.Y 2022-2023**

Submitted to:

**Prof. DELA CRUZ, FRANCIS L.**

Faculty in Charge

**TEAM PROJECT Information Management**

**Semester 2 AY 2022-2023**

**Case Study**

**Case Narration Section:**

1. Happy Travel (HAT) Is a web-based company that has decided to expand its business by providing its registered customers with a one-stop shop flight information system to any destination in the world. The database will assist the customer in planning his/her trip in advance as every registered airline with Happy Travel sends its entire fleet’s flight information, which then is posted on the website. This information is updated regularly to ensure the accuracy of the information.
2. However, the company is also accepting transactions over the counter to serve customers who do not have access to the Internet. They can fill up forms to populate the company database at the end of the day. These forms contain passengers’ data and information for the flight they wish to avail.
3. On any given day, several airlines have flights to the same destination. On these flights, information about the plane such as the plane model, seating capacity, and amenities on board, is provided.
4. Information for each flight includes the flight schedule; the customer can view in advance the departure and arrival time of the flight, where the flight originated from, and its destination.
5. The customer can also check whether the flight is fully booked by looking at the seat availability of the chosen flight. The seats are classified into 2 classes, namely first and economy class. The number of seats per class differs on every plane. The seat price is determined by the seat class it belongs to, the first class costs more than the economy class. The airline companies have agreed to make the prices uniform so that whichever airline the customer chooses, the price is the same. The customer flying first class gets to board from the lounge section of the boarding area, whereas the customer flying on economy class will board through the normal check-in desk.

**Query Description Section** – (HAT Information Management Requirements)

QD-Sec1. Route and Destination of a Flight

|  |
| --- |
| SELECT origin, destination FROM Flight  WHERE flight\_number = <>; |

QD-Sec2. First Class Seat/s available on a particular flight

|  |
| --- |
| SELECT COUNT(\*) AS available\_seats  FROM Seat  WHERE flight\_number = <> AND class = 'First Class' AND is\_available = true; |

QD-Sec3. Flights scheduled on a given date

QD-Sec4. Flights going to a given destination

QD-Sec5. Price of a given flight

QD-Sec6. Plane to use for a given flight

QD-Sec7. Planes that will fly on a given schedule/s.

QD-Sec8. Arrival time of certain types of planes.

QD-Sec9. The departure time of a flight to a particular destination.

QD-Sec10.A number of seats available for a particular flight.

QD-Sec11. List of Passengers boarding to lounge section.

QD-Sec12. List of Passengers who make first-class seat reservations.

QD-Sec13. Seating plan of a given flight.

**GROUP TASKS:**

1. Enumerate the data that needs to be accommodated in your database system.

**Entity Name:** flight, counter, destination, passenger, first class passenger.

**Attributes of Flight**: flight number, flight name, departure date, departure time, price, to from, arrival date, number of seats.

**Attributes of Counter**: counter number, counter name.

**Attributes of Passenger**: passenger id, name, from, destination, boarding time, boarding date.

**Attributes of First-class Passenger**: passenger id, name, from, destination, boarding time, boarding date.

**Attributes of Lounge**: name, lounge id.

**Attributes of Destination**: country, city, name, destination.

1. Draw the Conceptual Schema using the ER-Diagram.

A picture containing screenshot, circle, black and white, graphics

Description automatically generated

1. Draw the Logical Design (Conceptual Schema + attributes) using the ERD.

A picture containing diagram, line, text, parallel

Description automatically generated

1. Transform your Database's Logical Design to its equivalent Relational Schema.

**COUNTER**

**FLIGHT**

Number of Seat Price

Arrival Date Departure Time

Arrival Time Departure Date

From Flight name

To Flight Number

=

Counter Number

Name

>

>

Fill

Form

=

Search and

Book Flight

Can have

>

**PASSENGER**

>

**DESTINATION**

Boarding Time Passenger ID

Boarding Data Name

Destination From

>

Country Name

City Destination ID

**FIRST CLASS PASSENGER**

>

Passenger ID Destination

Name Boarding Date

Boarding Time From

Search and

Book Flight

>

**LOUNGE**

Name

Lounge ID

Access to

=

1. Determine and specify the functional dependencies that may exist in your relations.

**FLIGHT ID -> DEPARTURE DATE, BOARDING DATE, ARRIVAL DATE** represents a partial dependency because **DEPARTURE DATE, BOARDING DATE, AND ARRIVAL DATE** rather than on the entire primary key composed of **FLIGHT ID AND PASSENGER ID**

**PRICE -> NO. OF SEAT** represents a transitive dependency because **PRICE** depends on an attribute **(NO. OF SEAT)** that is not part of a primary key.

**FLIGHT ID, PASSENGER ID -> DEPARTURE DATE, BOARDING DATE, ARRIVAL DATE, DESTINATION ID** represents a set of proper functional dependencies, because **DEPARTURE DATE, BOARDING DATE, ARRIVAL DATE, and DESTINATION ID** depend on the primary key composed of **FLIGHT ID** and **PASSENGER ID.**

1. Perform Normalization to relations with functional dependencies.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| FLIGHT ID | DEPARTURE DATE | BORADING DATE | ARRIVAL DATE | FLIGHT NAME | PRICE | NO. OF SEAT |

|  |  |
| --- | --- |
| DEPARTURE DATE | DEPARTURE TIME |

|  |
| --- |
| PRIMARY KEY |

|  |  |
| --- | --- |
| BOARDING DATE | BOARDING TIME |

|  |  |
| --- | --- |
| ARRIVAL DATE | ARRIVAL TIME |

|  |
| --- |
| FOREIGN KEY |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| PASSENGER ID | DESTINATION ID | FROM | NAME | LOUNGE ID | TO |

|  |  |
| --- | --- |
| NAME | BOARDING PASS |

|  |  |  |  |
| --- | --- | --- | --- |
| DESTINATION ID | DESTINATION | COUNTRY | CITY |

|  |  |
| --- | --- |
| COUNTER ID | COUNTER NAME |

|  |  |
| --- | --- |
| LOUNGE ID | LOUNGE NAME |

1. List all the derived final well-structured relations.
2. Write the SQL query using the final well-structured relations to generate the information requirements in the Query Description Section.