



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
FACULTAD DE INGENIERÍA



Sistema de Vuelos

Proyecto Final



ELABORADO POR:

Manuel Salvador Cabrera Sánchez

Luis Sergio Cano Olguín

Eduardo Enrique López Salinas

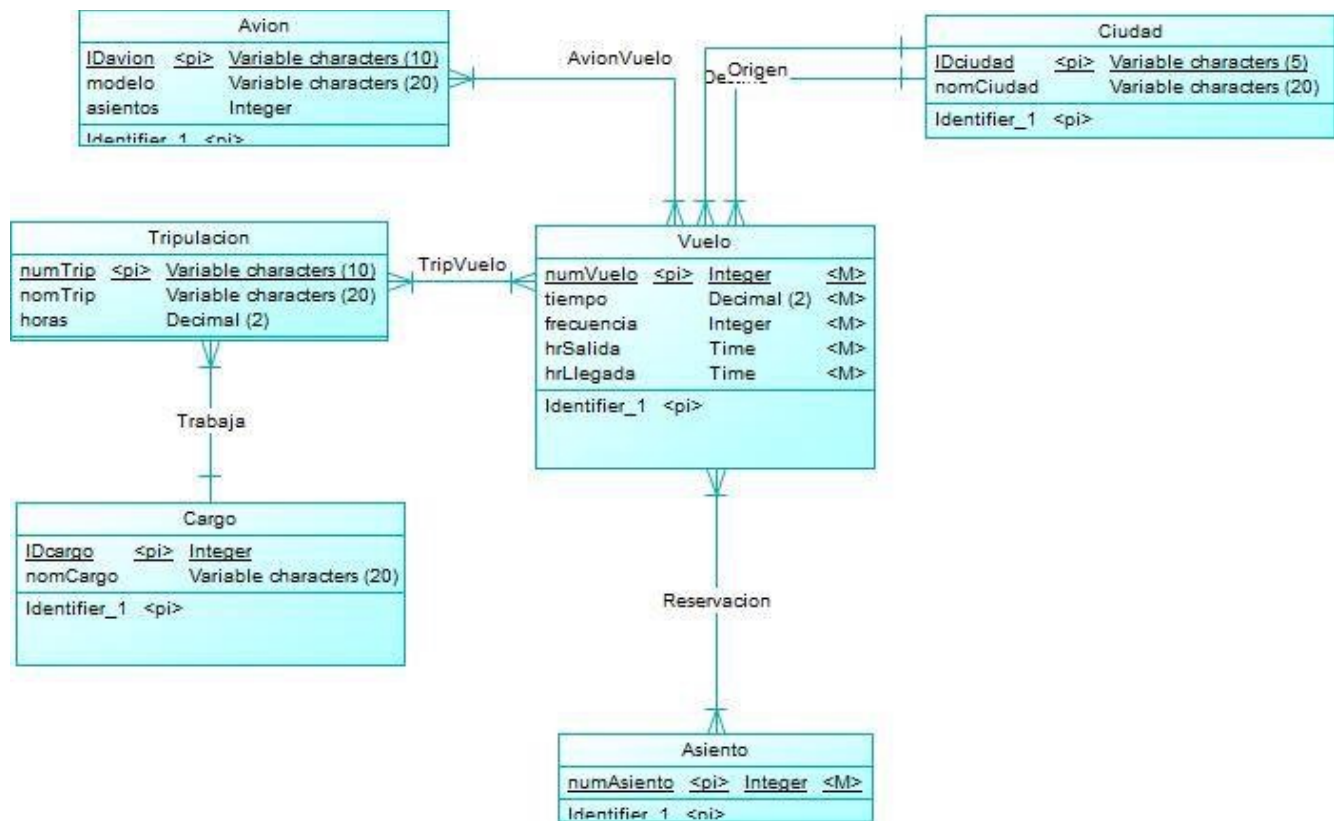
Bases de Datos - Grupo 4
Dra. María del Pilar Ángeles

5 de diciembre de 2017

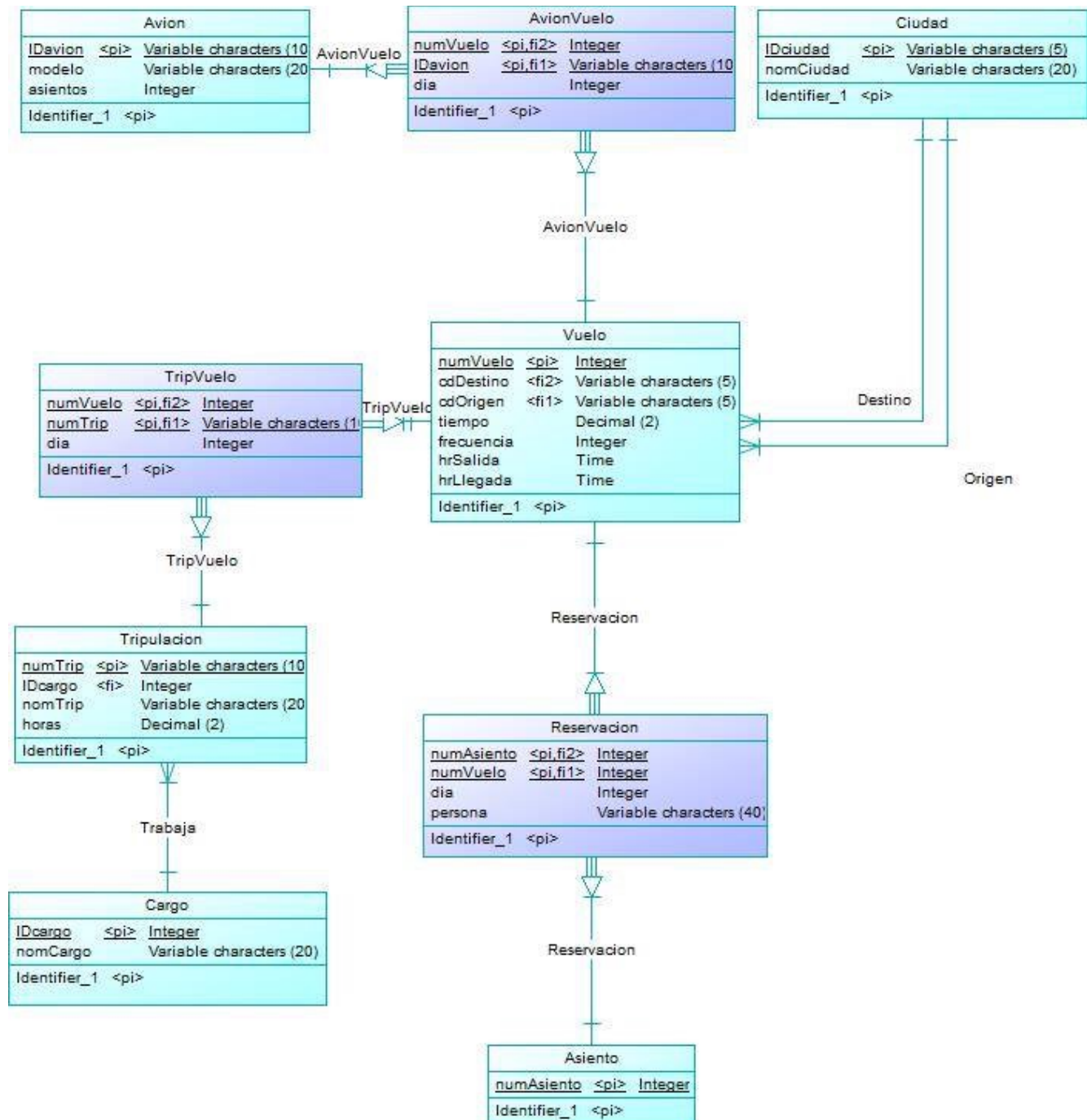
Hoja de Requerimientos	3
Diagrama Entidad - Relación	4
Modelo Relacional	5
Programas en SQL	6
1) Creación de base de datos	6
2) Creación de tablas	6
3) Llaves primarias y foráneas	7
4) Integridad de dominio y Reglas de negocio	8
5) Alta de registros a las tablas	9
6) Creación de índices	12
7) Procedimiento almacenado para Inserción	12
8) Procedimiento almacenado para Actualización	12
9) Procedimiento almacenado para Eliminación	13
10) Triggers para el control de Integridad	13
11) Consultas solicitadas en los Requerimientos	15
Reflexiones sobre el proyecto	18
Nuestra experiencia en la elaboración del proyecto	18
Resultados alcanzados con el equipo de trabajo	18
Impacto de la solución en el ámbito económico, ambiental y social	18
Fuentes de Consulta	19

Hoja de Requerimientos

Diagrama Entidad - Relación



Modelo Relacional



Programas en SQL

1) Creación de base de datos

```
CREATE DATABASE SISTEMA_VUELOS  
GO
```

2) Creación de tablas

```
CREATE TABLE Avion (IDavion varchar(10) NOT NULL,modelo varchar(20) NOT  
NULL,asientos int NOT NULL)
```

```
CREATE TABLE Ciudad (IDciudad varchar(5) NOT NULL,nomCiudad varchar(20)  
NOT NULL)
```

```
CREATE TABLE Vuelo (numVuelo int NOT NULL,tiempo decimal(2) NOT  
NULL,frecuencia int NOT NULL,hrSalida time NOT NULL,hrLlegada time NOT  
NULL,cdOrigen varchar(5) NOT NULL,cdDestino varchar(5) NOT NULL)
```

```
CREATE TABLE AvionVuelo (IDavion varchar(10) NOT NULL,numVuelo int NOT  
NULL,dia int NOT NULL)
```

```
CREATE TABLE Cargo (IDcargo int NOT NULL,nomCargo varchar(20) NOT NULL)
```

```
CREATE TABLE Tripulacion (numTrip varchar(10) NOT NULL,nomTrip varchar(20)  
NOT NULL,horas decimal(2) NOT NULL,IDcargo int NOT NULL)
```

```
CREATE TABLE TripVuelo (numTrip varchar(10) NOT NULL,numVuelo int NOT  
NULL,dia int NOT NULL)
```

```
CREATE TABLE Asiento (numAsiento int NOT NULL)
```

```
CREATE TABLE Reservacion (numVuelo int NOT NULL,numAsiento int NOT  
NULL,dia int NOT NULL,persona varchar(40) default '_')
```

```
GO
```

3) Llaves primarias y foráneas

```
ALTER TABLE Avion  
ADD CONSTRAINT pk_avion PRIMARY KEY (IDavion)
```

```
ALTER TABLE Ciudad  
ADD CONSTRAINT pk_ciudad PRIMARY KEY (IDciudad)
```

```
ALTER TABLE Vuelo  
ADD CONSTRAINT pk_vuelo PRIMARY KEY (numVuelo)  
ADD CONSTRAINT fk_origen FOREIGN KEY (cdOrigen) REFERENCES  
Ciudad(IDciudad)  
ADD CONSTRAINT fk_destino FOREIGN KEY (cdDestino) REFERENCES  
Ciudad(IDciudad)
```

```
ALTER TABLE AvionVuelo  
ADD CONSTRAINT pk_av PRIMARY KEY (IDavion,numVuelo)  
ADD CONSTRAINT fk_ava FOREIGN KEY (IDavion) REFERENCES Avion(IDavion)  
ADD CONSTRAINT fk_avv FOREIGN KEY (numVuelo) REFERENCES  
Vuelo(numVuelo)
```

```
ALTER TABLE Cargo  
ADD CONSTRAINT pk_cargo PRIMARY KEY (IDcargo)
```

```
ALTER TABLE Tripulacion  
ADD CONSTRAINT pk_trip PRIMARY KEY (numTrip)  
ADD CONSTRAINT fk_cargo FOREIGN KEY (IDcargo) REFERENCES Cargo(IDcargo)
```

```
ALTER TABLE TripVuelo  
ADD CONSTRAINT pk_tv PRIMARY KEY (numTrip,numVuelo)  
ADD CONSTRAINT fk_tvt FOREIGN KEY (numTrip) REFERENCES  
Tripulacion(numTrip)  
ADD CONSTRAINT fk_tvv FOREIGN KEY (numVuelo) REFERENCES  
Vuelo(numVuelo)
```

```
ALTER TABLE Asiento  
ADD CONSTRAINT pk_asiento PRIMARY KEY (numAsiento)
```

```
ALTER TABLE Reservacion
ADD CONSTRAINT pk_res PRIMARY KEY (numVuelo,numAsiento)
ADD CONSTRAINT fk_resv FOREIGN KEY (numVuelo) REFERENCES
Vuelo(numVuelo)
ADD CONSTRAINT fk_asiento FOREIGN KEY (numAsiento) REFERENCES
Asiento(numAsiento)
```

```
GO
```

4) Integridad de dominio y Reglas de negocio

```
ALTER TABLE Avion
ADD CONSTRAINT chk_asientos CHECK (asientos>0)
```

```
ALTER TABLE Vuelo
ADD CONSTRAINT chk_frec CHECK (frecuencia BETWEEN 1 AND 5)
ADD CONSTRAINT chk_destino CHECK (cdDestino <> cdOrigen)
ADD CONSTRAINT chk_llegada CHECK (hrLlegada > hrSalida)
ADD CONSTRAINT chk_tiempo CHECK (tiempo>0)
```

```
ALTER TABLE Cargo
ADD CONSTRAINT chk_cargo CHECK (cargo BETWEEN 1 AND 4)
```

```
ALTER TABLE AvionVuelo
ADD CONSTRAINT chk_dia_av CHECK (dia>=1 AND dia<=356)
```

```
ALTER TABLE Tripulacion
ADD CONSTRAINT chk_horas CHECK (horas>0)
```

```
ALTER TABLE TripVuelo
ADD CONSTRAINT chk_dia_tv CHECK (dia>=1 AND dia<=356)
```

```
ALTER TABLE Reservacion
ADD CONSTRAINT chk_dia_res CHECK (dia>=1 AND dia<=356)
```

```
ALTER TABLE Asiento
ADD CONSTRAINT chk_asiento CHECK (numAsiento>0)
```

```
GO
```

5) Alta de registros a las tablas

```
INSERT INTO Avion VALUES('AV001','DC9',20)
INSERT INTO Avion VALUES('AV002','Boeing 797',20)
INSERT INTO Avion VALUES('AV003','Boeing 747',100)
INSERT INTO Avion VALUES('AV004','Boeing 777',238)
INSERT INTO Avion VALUES('AV005','Boeing 737',130)
INSERT INTO Avion VALUES('AV006','Boeing 787',226)
INSERT INTO Avion VALUES('AV007','Boeing 757',140)
INSERT INTO Avion VALUES('AV008','Boeing 727',128)
INSERT INTO Avion VALUES('AV009','A320',220)
INSERT INTO Avion VALUES('AV010','A380',550)
GO
```

```
INSERT INTO Ciudad VALUES('MEX','Ciudad de Mexico')
INSERT INTO Ciudad VALUES('GUA','Guadalajara')
INSERT INTO Ciudad VALUES('MON','Monterrey')
INSERT INTO Ciudad VALUES('HER','Hermosillo')
INSERT INTO Ciudad VALUES('VER','Veracruz')
INSERT INTO Ciudad VALUES('CUN','Cancun')
INSERT INTO Ciudad VALUES('YML','Montreal')
INSERT INTO Ciudad VALUES('OTW','Ottawa')
INSERT INTO Ciudad VALUES('TRT','Toronto')
INSERT INTO Ciudad VALUES('NWY','Nueva York')
GO
```

```
INSERT INTO Vuelo VALUES(180,1.0,1,'12:20:00','15:20:00','MEX','VER')
INSERT INTO Vuelo VALUES(181,2.0,2,'11:00:00','13:00:00','VER','MEX')
INSERT INTO Vuelo VALUES(182,3.0,3,'15:35:00','19:05:00','GUA','MON')
INSERT INTO Vuelo VALUES(183,2.0,4,'09:00:00','11:00:00','HER','MEX')
INSERT INTO Vuelo VALUES(184,2.0,5,'12:00:00','14:00:00','CUN','MON')
INSERT INTO Vuelo VALUES(185,6.0,1,'14:10:00','20:10:00','HER','MON')
INSERT INTO Vuelo VALUES(186,7.0,2,'03:55:00','11:55:00','MEX','YML')
INSERT INTO Vuelo VALUES(187,6.0,3,'12:20:00','17:20:00','OTW','MON')
INSERT INTO Vuelo VALUES(188,5.0,4,'10:00:00','16:00:00','CUN','TRT')
INSERT INTO Vuelo VALUES(189,3.0,5,'17:40:00','19:40:00','NWY','MEX')
GO
```

```
INSERT INTO AvionVuelo VALUES('AV001',180,1)
INSERT INTO AvionVuelo VALUES('AV002',182,115)
INSERT INTO AvionVuelo VALUES('AV003',186,55)
INSERT INTO AvionVuelo VALUES('AV004',181,39)
INSERT INTO AvionVuelo VALUES('AV005',185,96)
INSERT INTO AvionVuelo VALUES('AV006',183,96)
INSERT INTO AvionVuelo VALUES('AV007',180,50)
INSERT INTO AvionVuelo VALUES('AV008',180,51)
INSERT INTO AvionVuelo VALUES('AV009',187,3)
INSERT INTO AvionVuelo VALUES('AV010',188,14)
GO
```

```
INSERT INTO Cargo VALUES(1,'Piloto')
INSERT INTO Cargo VALUES(2,'Copiloto')
INSERT INTO Cargo VALUES(3,'Ingeniero')
INSERT INTO Cargo VALUES(4,'Sobrecargo')
GO
```

```
INSERT INTO Tripulacion VALUES('TRIP001','Hernandez',20.0,1)
INSERT INTO Tripulacion VALUES('TRIP002','Sosa',16.0,2)
INSERT INTO Tripulacion VALUES('TRIP003','Cabrera',43.0,3)
INSERT INTO Tripulacion VALUES('TRIP004','Sanchez',22.0,4)
INSERT INTO Tripulacion VALUES('TRIP005','Cano',60.0,1)
INSERT INTO Tripulacion VALUES('TRIP006','Olguin',56.0,2)
INSERT INTO Tripulacion VALUES('TRIP007','Lopez',13.0,3)
INSERT INTO Tripulacion VALUES('TRIP008','Salinas',40.0,4)
INSERT INTO Tripulacion VALUES('TRIP009','Ventura',36.0,1)
INSERT INTO Tripulacion VALUES('TRIP010','Maxime',10.0,2)
GO
```

```
INSERT INTO TripVuelo VALUES('TRIP001',181,39)
INSERT INTO TripVuelo VALUES('TRIP007',186,55)
INSERT INTO TripVuelo VALUES('TRIP003',186,55)
INSERT INTO TripVuelo VALUES('TRIP004',183,96)
INSERT INTO TripVuelo VALUES('TRIP008',185,96)
INSERT INTO TripVuelo VALUES('TRIP005',180,50)
INSERT INTO TripVuelo VALUES('TRIP009',180,51)
INSERT INTO TripVuelo VALUES('TRIP002',180,1)
INSERT INTO TripVuelo VALUES('TRIP006',182,115)
INSERT INTO TripVuelo VALUES('TRIP010',188,14)
GO
```

```
INSERT INTO Asiento VALUES(01)
INSERT INTO Asiento VALUES(02)
INSERT INTO Asiento VALUES(03)
INSERT INTO Asiento VALUES(04)
INSERT INTO Asiento VALUES(05)
INSERT INTO Asiento VALUES(06)
INSERT INTO Asiento VALUES(07)
INSERT INTO Asiento VALUES(08)
INSERT INTO Asiento VALUES(09)
INSERT INTO Asiento VALUES(10)
INSERT INTO Asiento VALUES(11)
INSERT INTO Asiento VALUES(12)
INSERT INTO Asiento VALUES(13)
INSERT INTO Asiento VALUES(14)
INSERT INTO Asiento VALUES(15)
INSERT INTO Asiento VALUES(16)
INSERT INTO Asiento VALUES(17)
INSERT INTO Asiento VALUES(18)
INSERT INTO Asiento VALUES(19)
INSERT INTO Asiento VALUES(20)
GO
```

```
INSERT INTO Reservacion VALUES(182,06,115,'Karla Rosas')
INSERT INTO Reservacion VALUES(182,07,115,'Edwin Diaz')
INSERT INTO Reservacion VALUES(182,08,115,'Avery Price')
INSERT INTO Reservacion VALUES(182,09,115,'Raquel Bautista')
INSERT INTO Reservacion VALUES(182,10,115,'Raul Sosa')
INSERT INTO Reservacion VALUES(180,16,50,'Mariana Rodriguez')
INSERT INTO Reservacion VALUES(180,17,50,'Patricia Padilla')
INSERT INTO Reservacion VALUES(180,18,50,'Samantha Gasga')
INSERT INTO Reservacion VALUES(180,19,50,'Jonathan Luna')
INSERT INTO Reservacion VALUES(180,20,50,'Eowyn Frozen')
GO
```

6) Creación de índices

```
CREATE UNIQUE NONCLUSTERED INDEX avion_ind  
ON Avion(IDavion)
```

```
CREATE UNIQUE NONCLUSTERED INDEX ciudad_ind  
ON Ciudad(IDciudad)
```

```
CREATE UNIQUE NONCLUSTERED INDEX vuelo_ind  
ON Vuelo(numVuelo)
```

```
CREATE UNIQUE NONCLUSTERED INDEX trip_ind  
ON Tripulacion(numTrip)
```

```
GO
```

7) Procedimiento almacenado para Inserción

```
CREATE Procedure sp_inserta  
(@IDavion varchar(10),  
 @modelo varchar(20),  
 @asientos int)  
AS  
BEGIN  
    INSERT INTO Avion ([IDavion], [modelo], [asientos])  
    VALUES (@IDavion, @modelo, @asientos)  
END  
GO
```

8) Procedimiento almacenado para Actualización

```
CREATE procedure sp_actualiza  
(@IDavion varchar(10),  
 @modelo varchar(20),  
 @asientos int)  
AS  
UPDATE Avion SET @modelo = modelo, @asientos = asientos  
WHERE @IDavion = IDavion
```

9) Procedimiento almacenado para Eliminación

```
CREATE procedure sp_borra
(@IDavion varchar(10),
 @modelo varchar(20),
 @asientos int )
AS
DELETE FROM Avion
WHERE @IDavion = IDavion
AND @modelo = modelo
AND @asientos = asientos

GO
```

10) Triggers para el control de Integridad

/* NOTA: Programamos este trigger para controlar los asientos ocupados, pero nos daba problemas al insertar en Reservacion y decidimos eliminarlo.

```
CREATE TRIGGER ocupado
ON Reservacion
FOR INSERT AS
IF ((SELECT COUNT(*) FROM Reservacion,inserted WHERE
Reservacion.numVuelo=inserted.numVuelo AND
Reservacion.numAsiento=inserted.numAsiento AND
Reservacion.dia=inserted.dia)>(0))
BEGIN
rollback tran
print "El asiento ya esta ocupado, elija otro"
END
```

```
GO
```

```
*/
```

```
/* Control del número total de asientos por cada avión */  
CREATE TRIGGER num_asiento  
ON Reservacion  
FOR INSERT AS  
IF (SELECT asientos FROM Avion WHERE IDavion IN(SELECT IDavion FROM  
AvionVuelo,inserted WHERE  
AvionVuelo.numVuelo=inserted.numVuelo))<(SELECT numAsiento FROM  
inserted)  
BEGIN  
rollback tran  
print "El asiento no existe en este avion"  
END
```

```
/* Evitamos borrar una ciudad que está registrada como origen o destino */  
CREATE TRIGGER delete_cd  
ON Ciudad  
FOR DELETE AS  
IF (SELECT COUNT(*) FROM Vuelo,deleted WHERE  
Vuelo.cdOrigen=deleted.IDciudad OR Vuelo.cdDestino=deleted.IDciudad)>0  
BEGIN  
ROLLBACK TRAN  
PRINT "No puedes eliminar una ciudad con vuelos asociados a ella"  
END
```

```
/* Evitamos borrar un avión que tiene vuelos pendientes, si se desea eliminar  
primero se deberá cambiar el avión para dichos vuelos */  
CREATE TRIGGER delete_av  
ON Avion  
FOR DELETE AS  
IF (SELECT COUNT(*) FROM AvionVuelo,deleted WHERE  
AvionVuelo.IDavion=deleted.IDavion)>0  
BEGIN  
ROLLBACK TRAN  
PRINT "No puedes eliminar un avion con vuelos asociados a el"  
END
```

/* Eliminación en cascada de la relación de una tripulación a un vuelo, cuando la tripulación es eliminada */

```
CREATE TRIGGER delete_trip
ON Tripulacion
FOR DELETE AS
DELETE TripVuelo
FROM TripVuelo,deleted
WHERE TripVuelo.numTrip=deleted.numTrip
```

/* Eliminación en cascada de la relación de un vuelo a un avión, las reservaciones y la tripulación, cuando el vuelo es eliminado */

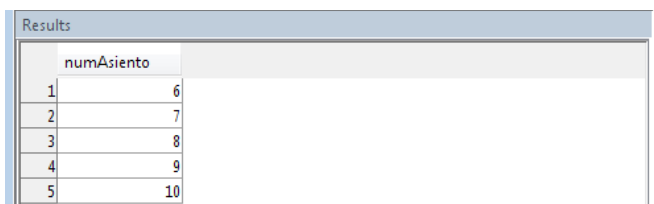
```
CREATE TRIGGER delete_vuelo
ON Vuelo
FOR DELETE AS
DELETE AvionVuelo
FROM AvionVuelo,deleted
WHERE AvionVuelo.numVuelo=deleted.numVuelo
DELETE TripVuelo
FROM TripVuelo,deleted
WHERE TripVuelo.numVuelo=deleted.numVuelo
DELETE Reservacion
FROM Reservacion,deleted
WHERE Reservacion.numVuelo=deleted.numVuelo
```

GO

11) Consultas solicitadas en los Requerimientos

i) ¿Qué asientos hay libres el día 115 para volar de Guadalajara a Monterrey?

```
SELECT numAsiento
FROM Reservacion, Vuelo
WHERE Reservacion.numVuelo=Vuelo.numVuelo
HAVING cdOrigen='GUA' AND cdDestino='MON' AND dia=115
GO
```

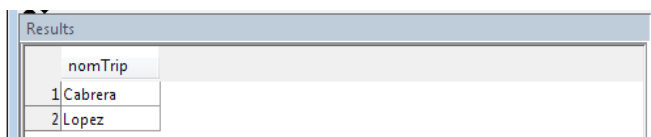


Results	
numAsiento	
1	6
2	7
3	8
4	9
5	10

(Resultados de la consulta)

ii) ¿Quiénes son los ingenieros de vuelo que el día 24 de febrero volarán más de 4 horas?

```
SELECT nomTrip
FROM Tripulacion
WHERE numTrip IN(
    SELECT numTrip
    FROM TripVuelo
    WHERE dia=55
    AND numVuelo IN(
        SELECT numVuelo
        FROM Vuelo
        WHERE tiempo>4))
```

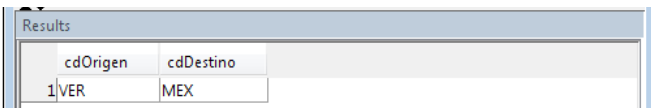


nomTrip
1 Cabrera
2 Lopez

(Resultados de la consulta)

iii) ¿Trabaja el capitán Hernández hoy (hoy es el día 39)? En caso de que haya trabajado ¿en qué ciudades puedo encontrarlo?

```
SELECT cdOrigen,cdDestino
FROM Vuelo
WHERE numVuelo IN(
    SELECT numVuelo
    FROM TripVuelo
    WHERE dia=39
    AND numTrip IN(
        SELECT numTrip
        FROM Tripulacion
        WHERE nomTrip='Hernandez'))
```



cdOrigen	cdDestino
1 VER	MEX

(Resultados de la consulta)

iv) ¿Quiénes son los sobrecargos que salen de Hermosillo el día 96?

```
SELECT nomTrip
FROM Tripulacion
WHERE numTrip IN(
    SELECT numTrip
    FROM TripVuelo
    WHERE dia=96
    AND numVuelo IN(
        SELECT numVuelo
        FROM Vuelo
        WHERE cdOrigen IN(
            SELECT IDciudad
            FROM Ciudad
            WHERE nomCiudad='Hermosillo'))))
```

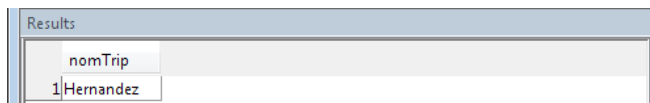


nomTrip
1 Sanchez
2 Salinas

(Resultados de la consulta)

vi) ¿Qué pilotos tripularán en el año sólo aviones con capacidad para 160 o más pasajeros?

```
SELECT nomTrip
FROM Tripulacion
WHERE IDcargo=1
AND Tripulacion.numTrip IN(
    SELECT numTrip
    FROM TripVuelo
    WHERE numVuelo IN(
        SELECT numVuelo
        FROM AvionVuelo
        WHERE IDavion IN(
            SELECT IDavion
            FROM Avion
            WHERE asientos>=160)))
```



nomTrip
1 Hernandez

(Resultados de la consulta)

Reflexiones sobre el proyecto

Nuestra experiencia en la elaboración del proyecto

La experiencia fue bastante buena, pues aplicamos todo lo aprendido a lo largo de todo el curso; asimismo, aprendimos a utilizar una herramienta nueva como Power Designer de la cual no teníamos ningún conocimiento previo.

De igual forma la contribución de todos los miembros del equipo facilitó la realización y rapidez de este proyecto.

Resultados alcanzados con el equipo de trabajo

Si bien la organización en nuestro equipo no fue lo suficientemente sólida, pudimos sacar el proyecto adelante. Realizamos todos los puntos indicados, sin embargo, tuvimos algunas dificultades al implementar dos de las consultas pedidas en los requerimientos. De cualquier modo, quedamos muy satisfechos con los resultados ya que este proyecto es un claro reflejo de los conocimientos adquiridos y a su vez queda demostrado el alto nivel de enseñanza por parte de la profesora.

Impacto de la solución en el ámbito económico, ambiental y social

Este proyecto podría contribuir enormemente al ámbito social, ya que presenta un claro registro y control de vuelos nacionales e internacionales. Posiblemente el diseño de la base de datos podría mejorarse y hacerse más eficiente; el que desarrollamos cumple con las funciones básicas basadas en la hoja de requerimientos, posteriormente y con un poco más de conocimiento podría ser optimizado.

Fuentes de Consulta

Notas de Bases de Datos. Recopilado por María del Pilar Ángeles. Facultad de Ingeniería, Universidad Nacional Autónoma de México. 2013.

Referencia de Transact-SQL. TechNet, Microsoft. Consultado Diciembre 2017.
[[https://technet.microsoft.com/es-es/library/ms189826\(v=sql.90\).aspx](https://technet.microsoft.com/es-es/library/ms189826(v=sql.90).aspx)]

Transact-SQL Users Guide. Infocenter, Sybase. Consultado Diciembre 2017.
[<http://infocenter.sybase.com/help/index.jsp?topic=/com.sybase.infocenter.dc32300.1570/html/sqlug/title.htm>]