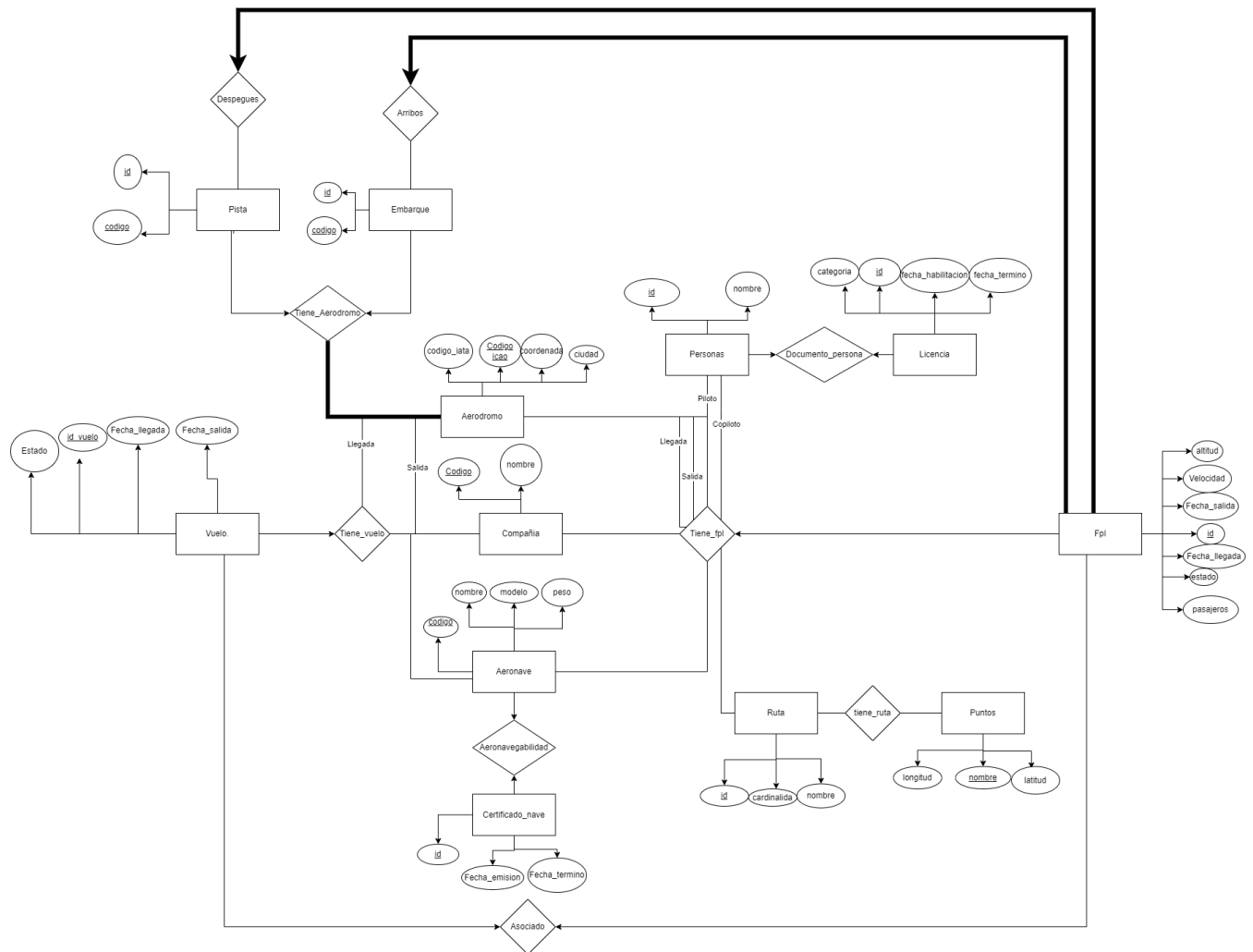


Entrega 2.



Es necesario destacar que el esquema entregado tiene en consideración todo el problema, no solo la parte entregada en los datos, por lo que se encuentra mucho más completo, listo para ser utilizado al momento de tener toda la información disponible. (dejaré la imagen en formato png junto al informe.) Notar que en el esquema no se encuentran todos los atributos de las tablas, pero si se encuentran especificados en el esquema relacional.

El esquema que se utilizó para esta entrega es el siguiente:

Vuelo(id: int, fecha_de_salida: date, fecha_de_llegada: date, estado: string)

Aeródromo(id : int,codigo_ICAO : string, código_IATA : string, atitud: int, longitud: int, ciudad : string, pais: string)

Fpl(id : int, fecha_de_salida : date, fecha_de_llegada : date, velocidad : int, altitud : int, tipo_de_vuelo : string, numero_de_pasajeros : int, estado: string)

Asociado(id.Vuelo : int, id.Fpl : int)

Aeronave(certificado_id: int, fecha_habilitacion: date, fecha_termino: date, id: int, nombre: string, modelo: string, peso: int, codigo : int)

Tiene_vuelo(Vuelo.id: int, Compañías.codigo: string, Aerodromo(salida).codigo_ICAO: string, Aerodromo(llegada).codigo_ICAO: string, Aeronave.codigo: int)

Tiene_fpl(Vuelo.id: int, Compañías.codigo: string, Aerodromo(salida).codigo_ICAO: string, Aerodromo(llegada).codigo_ICAO: string, Aeronave.codigo: int, Persona(piloto).id : int, Persona(copiloto).id: int, Ruta.id: int)
Pilotos(certificado_id: int, fecha_habilitacion: date, fecha_termino: date, categoria:string, pasaporte:string)

Adicionalmente, dejamos estas tablas, por completitud, que deben ser utilizadas para una modelación completa del problema que se nos presenta dado en la entrega 1.

Arribos (Embarque.id : int, Embarque.codigo: int, Fpl.id : int)
Compañías aéreas(nombre: string, código :string)
Despegues (Pista.id : int, Pista.codigo : int, Fpl.id : int)
Pista(id: int, codigo: int)
Aerodromo_tiene_P(Pista.id : int, Pista.codigo : int, Aerodromo.codigo_ICAO: string)
Aerodromo_tiene_E(Embarque.id : int, Embarque.codigo : int, Aerodromo.codigo_ICAO: string)
Embarque(id:int, codigo: int)
Persona(nombre: string, id: int) Solo dan los pasaportes
Licencia(id: int, fecha_de_habilitación: date, fecha_de_término: date, categoria: string)
Documento_persona(Persona.id : int, Licencia.id: int)
Ruta(id: int, nombre: string, cardinalidad: int)
Punto(nombre: string, latitud: float, longitud: float)
Tiene_ruta(Ruta.id: int, Punto.nombre: string)

Validación modelo.

Nuestro modelo se encuentra en 3NF, ya que al crear las tablas para nuestra modelación, utilizando las entregadas en los datos de la entrega, no preservamos todas las dependencias y tenemos cierto grado de redundancia. Por que existe redundancia y porque existen tablas que no preservan las dependencias no puede ser BCNF. Por ejemplo, nuestra tabla Aerodromo_tiene_p se compone de dos atributos que no mantuvieron las dependencias originales, entre ellos son independientes. Otro ejemplo, el atributo estado se repite en varias tablas, por ejemplo, en fpl y vuelo, esto es redundancia, algo que no es aceptable en BCNF, pero si en 3NF.

Además, dentro de las dependencias lógicas que podríamos intentar proponer, se tiene que X siempre será una super llave, por lo que cumpliríamos las condiciones para que el modelo este en 3NF.

En la tabla Aerodromo_tiene_E, tampoco se mantienen las dependencias originales, los atributos son independientes entre sí. La tabla Despegue va a repetir varias veces los códigos de pista, ya que diferentes fpl pueden llegar a una misma pista, sin embargo no nos fue posible separar la tabla y generar alguna dependencia o relacionarlos con otros atributos, no pudimos eliminar la redundancia.

En el resto de las tablas que no fueron mencionadas tienen super llaves en cualquier dependencia que se pueda inferir, por lo tanto están en 3NF.

Consultas utilizadas en pagina web.

1.SELECT * FROM vuelo WHERE estado LIKE 'pendiente';

2.SELECT * FROM vuelo, (SELECT id FROM aerodromo WHERE icao LIKE UPPER('\$origen')) AS t1, (SELECT id FROM aerodromo WHERE icao LIKE UPPER('\$destino')) AS t2, tiene_vuelo
WHERE vuelo.id =tiene_vuelo.idvuelo AND tiene_vuelo.salida=t1.id AND
tiene_vuelo.llegada=t2.id AND vuelo.estado='aceptado';

3.SELECT id, categoria, pasaporte FROM documento_p WHERE inicio<='\$fecha' and
termino>'\$fecha';

4. no realizado

5.SELECT aeronave, COUNT(aeronave) FROM vuelo, tiene_vuelo WHERE vuelo.id
=tiene_vuelo.idvuelo AND
vuelo.realizado='realizado' AND vuelo.fecha_salida>='\$inicio' AND
vuelo.fecha_llegada<='\$final' GROUP BY aeronave;

SELECT aeronave, COUNT(aeronave) FROM fpl, tiene_fpl, (SELECT fplid FROM asociado)
AS t1 WHERE fpl.id = tiene_fpl.fplid AND
fpl.realizado='realizado' AND fpl.fecha_salida>='\$inicio' AND fpl.fecha_llegada<='\$final'
AND fpl.id NOT IN (t1.fplid) GROUP BY aeronave;

En este se realizaron dos consultas ya que tenemos, separados los vuelos de los fpl y es necesario contabilizar el viaje de naves en ambos, destacar que solo se contabilizan los fpl que no son vuelos, por lo que si se quiere el total de vuelos de una nave se suman ambos solamente.

6. (SELECT * FROM (SELECT pasaporte from documento_p) AS p1, (SELECT pasaporte
from documento_p) AS p2 WHERE p1<>p2 EXCEPT (SELECT piloto, copiloto FROM
tiene_fpl WHERE piloto<>copiloto)) EXCEPT (SELECT piloto,copiloto FROM tiene_vuelo
WHERE piloto<>copiloto);