**CSS 475 Group Project**

Completed Project Due: 16 March 2021

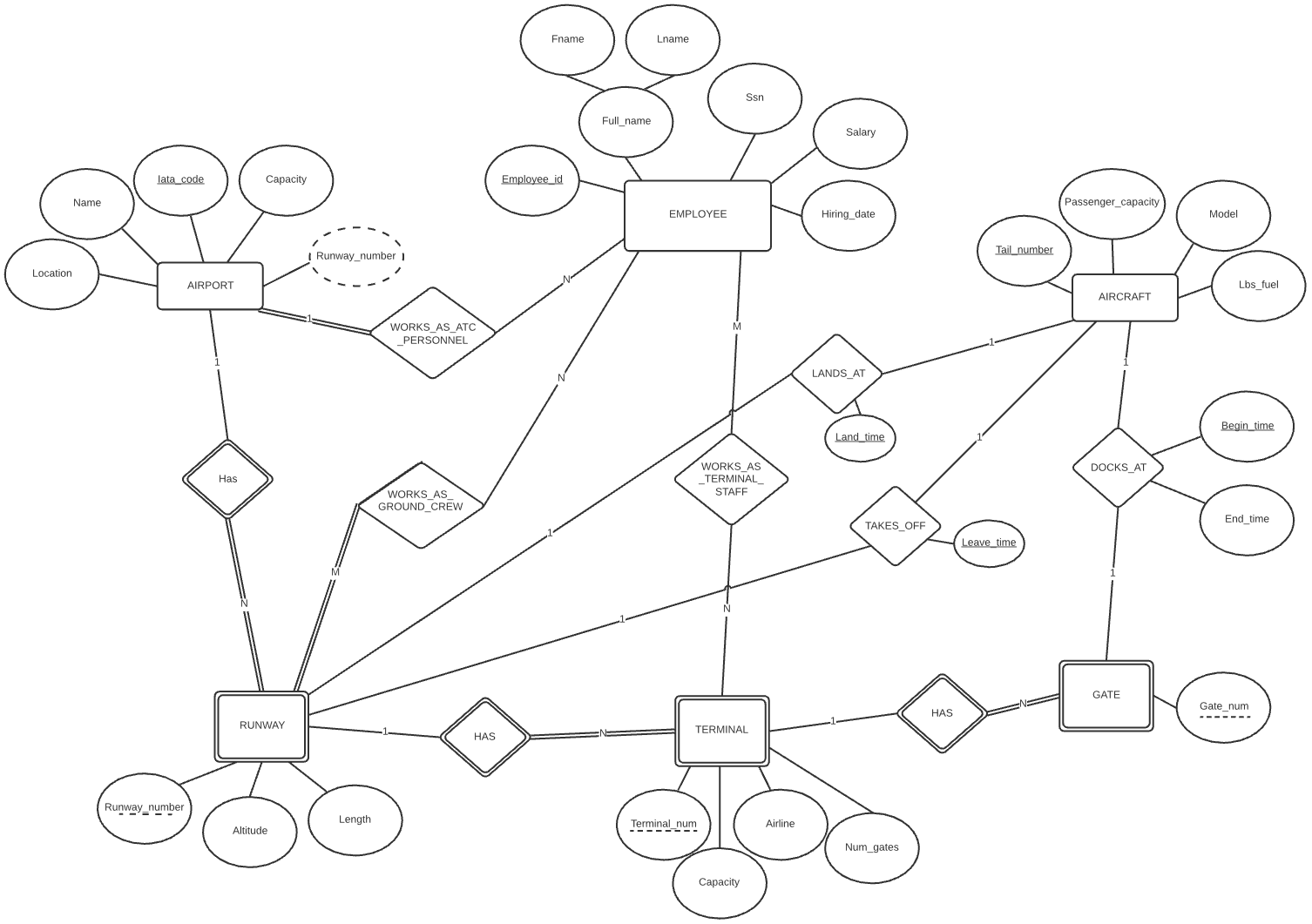
Participants: Christopher Hovsepian, Ethan Mo, Tanner Hanay, Marci Ma, Xue Guan

Project: Airport Database

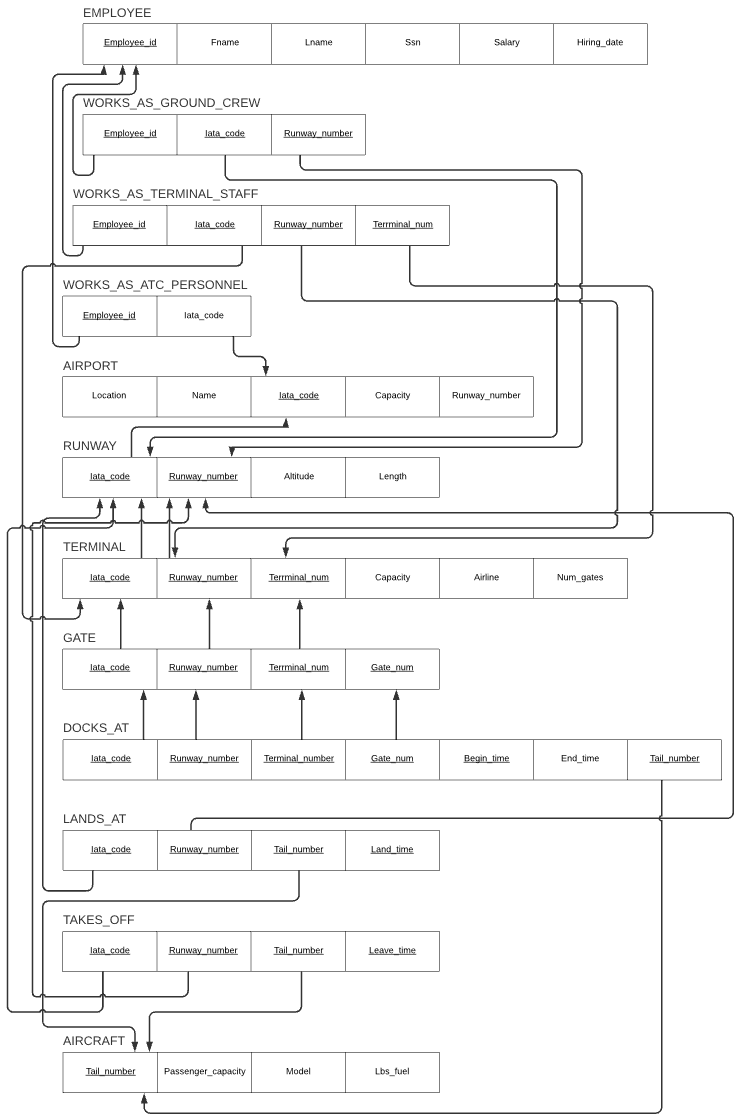
**Assumptions**:

* Ground crew can walk around to any gate attached to the runway they are assigned to.
* Because Gate is a weak entity to terminal and runway, the aircraft will land at a runway….
* 1 airport has multiple runways, 1 runway has multiple terminals, 1 terminal has multiple gates
* Employees could be assigned to runway and terminal

**Entity Relationship Diagram**



**Relational Data Model**



**Project Evaluation**:

Since the assignment was assigned we met for three hours every Friday to complete each task.

What went right:

We used lucidchart, google docs, and discord to collaborate on this project. Google docs was used for writing SQLite statements and these discussions. Lucidchart was used for making the ER diagram and Relational Data Model. Discord was used for its voice chat and for sharing links. We were able to meet every week and had no schedule issues. We planned out what we were going to do each week, and were able to finish the portion that we planned. We submitted it on time.

What went wrong:

Nothing, we are amazing. On a serious note, we did not run into any serious issues. We had to change a few things about our ER diagram once we started our Relational Data Model which waisted some time. When making our SQL statements we had some issues with the datetime format, we did research about how to use it while creating tables and queries, and resolved them eventually.

What would we do differently:

We should check if all the entities on the ER diagram can be mapped to the relational Data Model, so that we can avoid editing the ER diagram while doing the Relational Data Model.

**NF Disscussion**:

**EMPLOYEE** meets all the requirements for BCNF. Employee\_id is the key, and Ssn is a secondary key. There are no nested attributes, it meets full FD, and only the key can be used to determine the non-keys. This meets BCNF because the left side is only a candidate key.

**WORKS\_AS\_GROUND\_CREW** meets all the requirements for BCNF. Because all attributes are used as keys, it meets 1, 2, 3NF, and BCNF.

**WORKS\_AS\_TERMINAL\_STAFF** meets all the requirements for BCNF. Because all attributes are used as keys, it meets 1, 2, 3NF, and BCNF.

**WORKS\_AS\_ATC\_PERSONNEL** meets all the requirements for BCNF. Because all attributes are used as keys, it meets 1, 2, 3NF, and BCNF.

**AIRPORT** meets all the requirements for BCNF. Iata\_code is the primary key, and Location and Name together are a secondary key. There are no nested attributes, it meets full FD, and only the keys can be used to determine the non-keys. The left side is a candidate key only, so it meets BCNF.

**RUNWAY** meets all the requirements for BCNF. There are no nested attributes, it meets full FD, and only the keys can be used to determine the non-keys, not vice versa.

**TERMINAL** meets all the requirements for BCNF. There are no nested attributes, it meets full FD, and only the keys can be used to determine the non-keys, not vice versa.

**GATE** meets all the requirements for BCNF. Because all attributes are used as keys, it meets 1, 2, 3NF, and BCNF.

**DOCKS\_AT** meets all the requirements for BCNF. Because all but one attributes are used as keys, it meets 1, 2, 3NF. Because the key is on the left, and the only attribute is on the right, this meets BCNF.

**LANDS\_AT** meets all the requirements for BCNF. Because all attributes are used as keys, it meets 1, 2, 3NF, and BCNF.

**TAKES\_OFF** meets all the requirements for BCNF. Because all attributes are used as keys, it meets 1, 2, 3NF, and BCNF.

**AIRCRAFT** meets all the requirements for BCNF. There are no nested attributes, it meets full FD, and only the keys can be used to determine the non-keys, there's no non-key determining key.

**SQL Statement:**

|  |  |
| --- | --- |
| **SQL Statement** | **Purpose** |
| Select tail\_number, land\_time  From LANDS\_AT  Where Land\_time > '2020-03-05 00:00:00' AND  Land\_time < '2020-03-06 00:00:00'; | List of flights that arrived that day |
| Select Tail\_number, Leave\_time  From TAKES\_OFF  Where Leave\_time > '2020-03-05 00:00:00' AND  Leave\_time < '2020-03-06 00:00:00'; | List of fights that left that day |
| Select Tail\_number  From DOCKS\_AT  Where End\_time is null; | List Aircraft tail numbers that are currently docked to a terminal |
| Select tail\_number  From LANDS\_AT  Where tail\_number not in (Select tail\_number From DOCKS\_AT); | List Aircraft tail numbers that are waiting to dock with the terminal |
| Select avg(salary)  From employee e, WORKS\_AS\_ATC\_PERSONNEL  where e.Employee\_id = w.Employee\_id; | Average salary of air traffic control workers |
| Select count(iata\_code)  from TAKES\_OFF  Where Leave\_time > '2020-02-26 00:00:00' AND  Leave\_time < '2020-03-04 00:00:00'; | Number of flight departures over the last week. |
| Select Employee\_id, fname, lname  from employee  where salary < 40000; | List the employee name and employee ID having salary under $40,000 |
| Select Employee\_id, Salary  From EMPLOYEE  Order by Salary desc; | List all the employe ID numbers and orders them by salary decreasingly |
| Select gate\_num, airline  From GATE g, TERMINAL t  Where g.runway\_number = '34R' and g.Runway\_number = t.Runway\_number  Group by g.Terminal\_num; | List the gate number and airline info, for aircraft that landed at runway ‘34R’ and is grouped by terminal number. |

**/// SQLite**

.open finalproject.db

.mode column

.headers on

pragma table\_info(EMPLOYEE);

pragma table\_info(WORKS\_AS\_GROUND\_CREW);

pragma table\_info(WORKS\_AS\_TERMINAL\_STAFF);

pragma table\_info(WORKS\_AS\_ATC\_PERSONNEL);

pragma table\_info(AIRPORT);

pragma table\_info(RUNWAY);

pragma table\_info(TERMINAL);

pragma table\_info(GATE);

pragma table\_info(DOCKS\_AT);

pragma table\_info(LANDS\_AT);

pragma table\_info(TAKES\_OFF);

pragma table\_info(AIRCRAFT);

drop table EMPLOYEE;

drop table WORKS\_AS\_GROUND\_CREW;

drop table WORKS\_AS\_TERMINAL\_STAFF;

drop table WORKS\_AS\_ATC\_PERSONNEL;

drop table AIRPORT;

drop table RUNWAY;

drop table TERMINAL;

drop table GATE;

drop table DOCKS\_AT;

drop table LANDS\_AT;

drop table TAKES\_OFF;

drop table AIRCRAFT;

create table EMPLOYEE

(

Employee\_id char(10) not null,

Fname varchar(15) not null,

Lname varchar(15) not null,

Ssn char(9) not null,

Salary decimal(10,2),

Hiring\_date date(8),

Primary Key (Employee\_id),

Unique (Ssn)

);

create table WORKS\_AS\_GROUND\_CREW

(

Employee\_id char(10) not null,

Iata\_code varchar(4) not null,

Runway\_number varchar(5) not null,

Primary Key (Employee\_id, Iata\_code, Runway\_number),

Foreign key (Employee\_id)

references EMPLOYEE(Employee\_id)

on update cascade on delete set null,

Foreign key (Iata\_code, Runway\_number)

references RUNWAY(Iata\_code, Runway\_number)

on update cascade on delete set null

);

create table WORKS\_AS\_TERMINAL\_STAFF

(

Employee\_id char(10) not null ,

Iata\_code varchar(4) not null ,

Runway\_number varchar(5) not null ,

Terminal\_num int not null,

PRIMARY KEY (Employee\_id, Iata\_code, Runway\_number, Terminal\_num),

Foreign key (Employee\_id) references EMPLOYEE(Employee\_id) on update cascade on delete set null,

Foreign key (Iata\_code, Runway\_number, Terminal\_num)

references TERMINAL(Iata\_code, Runway\_number, Terminal\_num)

on update cascade on delete set null

);

create table WORKS\_AS\_ATC\_PERSONNEL

(

Employee\_id char(10) not null,

Iata\_code varchar(4) not null,

PRIMARY KEY(Employee\_id),

Foreign key(Iata\_code) references Airport(Iata\_code) on update cascade on delete cascade,

Foreign key (Employee\_id) references EMPLOYEE(Employee\_id) on update cascade on delete set null

);

create table AIRPORT

(

Location varchar(50),

Name varchar(50) not null,

Iata\_code varchar(4),

Capacity INT,

Runway\_number,

primary key(Iata\_code),

unique(Name)

);

create table RUNWAY

(

Iata\_code varchar(4) not null ,

Runway\_number varchar(5) not null ,

Altitude int,

Length int,

Primary key (Iata\_code, Runway\_number),

Foreign key (Iata\_code) references AIRPORT(Iata\_code) on update cascade on delete cascade

);

create table TERMINAL

(

Iata\_code varchar(4) not null,

Runway\_number varchar(5) not null,

Terminal\_num int not null,

Capacity int,

Airline varchar(30),

Num\_gates int,

Primary key(Iata\_code, Runway\_number, Terminal\_num),

Foreign key(Iata\_code, Runway\_number)

references RUNWAY(Iata\_code, Runway\_number)

on update cascade on delete cascade

);

create table GATE

(

Iata\_code char(4) not null,

Runway\_number varchar(5) not null,

Terminal\_num int not null,

Gate\_num varchar(3),

primary key (Iata\_code, Runway\_number, Terminal\_num, Gate\_num),

Foreign key (Iata\_code, Runway\_number, Terminal\_num)

references TERMINAL(Iata\_code, Runway\_number, Terminal\_num)

on update cascade on delete cascade

);

create table DOCKS\_AT

(

Iata\_code varchar(4) not null ,

Runway\_number varchar(5) not null ,

Terminal\_num int not null,

Gate\_num varchar(3),

Begin\_time datetime not null,

End\_time datetime,

Tail\_number varchar(10)not null,

Primary Key (Iata\_code, Runway\_number , Terminal\_num, Gate\_num, Begin\_time, Tail\_number),

Foreign key (Iata\_code, Runway\_number , Terminal\_num, Gate\_num)

references GATE(Iata\_code, Runway\_number , Terminal\_num, Gate\_num)

on update cascade on delete set null,

Foreign key (Tail\_number) references AIRCRAFT(Tail\_number) on update cascade on delete set null

);

create table LANDS\_AT

(

Iata\_code varchar(4) not null,

Runway\_number varchar(5) not null,

Tail\_number varchar(10) not null,

Land\_time datetime not null,

PRIMARY KEY(Iata\_code, Runway\_number, Tail\_number, Land\_time),

Foreign key(Iata\_code, Runway\_number) references RUNWAY(Iata\_code, Runway\_number) on update cascade on delete cascade

Foreign key(Tail\_number) references AIRCRAFT(Tail\_number) on update cascade on delete cascade

);

create table TAKES\_OFF

(

Iata\_code varchar(4) not null,

Runway\_number varchar(5) not null,

Tail\_number varchar(10) not null,

Leave\_time datetime,

PRIMARY KEY(Iata\_code, Runway\_number, Tail\_number, Leave\_time),

Foreign key(Iata\_code, Runway\_number) references RUNWAY(Iata\_code, Runway\_number) on update cascade on delete cascade

Foreign key(Tail\_number) references AIRCRAFT(Tail\_number) on update cascade on delete cascade

);

create table AIRCRAFT

(

Tail\_number varchar(10),

Passenger\_capacity int,

Model varchar(10),

Lbs\_fuel int,

primary key (Tail\_number)

);

**//// Populating the Tables**

.tables

Insert into EMPLOYEE

values

('9123123123', 'Bob', 'Smith', '000111222', 40000.00, 2015-01-01),

('8323123123', 'Billybob', 'Smith', '333111555', 45000.00, 2016-11-05),

('8323123555', 'Adam', 'Smith', '444777555', 47000.00, 2013-05-22),

('9090123120', 'Sponge', 'Bob', '888444000', 37000.00, 2020-06-24);

Insert into WORKS\_AS\_GROUND\_CREW

Values

('9123123123', 'SEA', '34R'),

('8323123123', 'SEA', '34R');

Insert into WORKS\_AS\_TERMINAL\_STAFF values ('8323123555', 'SEA', '34R', '5');

Insert into WORKS\_AS\_ATC\_PERSONNEL values ('9090123120', 'SEA');

insert into AIRPORT

values

('Seattle', 'Seattle-Tacoma International Airport', 'SEA', 15000, 3),

('Atlanta', 'Hartsfield-Jackson Atlanta International Airport', 'ATL', 20000, 5),

('New York', 'John F. Kennedy International Airport', 'JFK', 18000, 4);

Insert into RUNWAY

values

('SEA', '34R', 385, 11901);

Insert into TERMINAL

values

('SEA', '34R', 1, 6000, 'Alaska Airlines', 10);

Insert into GATE

values

('SEA', '34R', 5, 'A10'),

('SEA', '34R', 5, 'A11'),

('SEA', '34R', 5, 'A12'),

('SEA', '34R', 2, 'B5'),

('SEA', '34R', 2, 'B6');

Insert into DOCKS\_AT

values

('SEA', '34R', 1, 'A12', '2020-03-05 02:48:10', NULL, 'FG12451'),

('SEA', '34R', 1, 'B6', '2020-03-05 01:28:13', NULL, 'GWA1512'),

('SEA', '34R', 1, 'B5', '2020-03-05 11:28:13', '2020-03-05 02:38:22', 'GH21512');

insert into LANDS\_AT

values

('SEA', '34R', 'FG12451', '2020-03-05 02:43:10');

insert into TAKES\_OFF

values

('SEA', '34R', 'FG12451', '2020-03-05 03:39:10');

insert into AIRCRAFT

values

('FG12451', 150, 'B734', 1000),

('GWA1512', 200, 'B737', 1500),

('GH21512', 100, 'AC90', 900),

('JHU6823', 90, 'CL2T', 500),

('OIU16191', 10, 'CRJ9', 600);

**//// SQL statements (that provide functionality)**

Select tail\_number, land\_time

From LANDS\_AT

Where Land\_time > '2020-03-05 00:00:00' AND

Land\_time < '2020-03-06 00:00:00';

//List of flights that arrived that day

Select Tail\_number, Leave\_time

From TAKES\_OFF

Where Leave\_time > '2020-03-05 00:00:00' AND

Leave\_time < '2020-03-06 00:00:00';

//List of fights that left that day

Select Tail\_number

From DOCKS\_AT

Where End\_time is null;

//Aircraft tail numbers that are currently docked to a terminal

Select tail\_number

From LANDS\_AT

Where tail\_number not in (Select tail\_number From DOCKS\_AT);

//Aircraft tail numbers that are waiting to dock with the terminal

Select avg(salary)

From employee e, WORKS\_AS\_ATC\_PERSONNEL w

where e.Employee\_id = w.Employee\_id;

//Average salary of air traffic control workers (aggregation)

Select count(iata\_code)

from TAKES\_OFF

Where Leave\_time > '2020-02-26 00:00:00' AND

Leave\_time < '2020-03-04 00:00:00';

//Number of flight departures over the last week. (aggregation)

Select Employee\_id, fname, lname

from employee

where salary < 40000;

//Select from the ID number column, first name, last name from employee table, having salary under $40,000

Select Employee\_id, Salary

From EMPLOYEE

Order by Salary desc;

// this lists all the employe ID numbers and orders them by salary decreasingly

Select gate\_num, airline

From GATE g, TERMINAL t

Where g.runway\_number = '34R' and g.Runway\_number = t.Runway\_number

Group by g.Terminal\_num;

// List the gate number and airline info, for aircraft that landed at runway ‘34R’ and is grouped by terminal number.

Select \* from EMPLOYEE;

select \* from AIRCRAFT;

select \* from AIRPORT;

select \* from DOCKS\_AT;

select \* from EMPLOYEE;

select \* from GATE;

select \* from LANDS\_AT;

select \* from RUNWAY;

select \* from TAKES\_OFF;

select \* from TERMINAL;

select \* from WORKS\_AS\_ATC\_PERSONNEL;

select \* from WORKS\_AS\_GROUND\_CREW;

select \* from WORKS\_AS\_TERMINAL\_STAFF;