CALIFORNIA STATE UNIVERSITY, SAN BERNARDINO
SCHOOL OF COMPUTER SCIENCE & ENGINEERING

Database Design Project

PROJECT OPTION

FastCabs

TEAM NAME

Dunder Mifflin

TEAM MEMBERS

Alvarez,Paul
Nahhas,Philip
Awad,Saker

Project Description: FastCabs

A private taxi company called FastCabs was established in Glasgow in 1992. Since then, the company has grown steadily and now has offices in most of the main cities of Scotland. However, the company is now so large that more and more administrative staff are being employed to cope with the ever-increasing amount of paperwork. Furthermore, the communication and sharing of information within the company is poor. The Director of the company, Paddy MacKay feels that too many mistakes are being made and that the success of his company will be short-lived if he does not do something to remedy the situation. He knows that a database could help in part to solve the problem and has approached you and your team to help in creating a database application to support the running of FastCabs.

The Director has provided the following brief description of how FastCabs operates.

Each office has a Manager; several taxi owners, drivers and administrative staff. The Manager is responsible for the day-to-day running of the office. An owner provides one or more taxis to FastCabs and each taxi is allocated for use to a number of drivers. The majority of owners are also drivers.

FastCab taxis are not available for hire by the public hailing a taxi in the street but must be requested by first phoning the company to attend a given address.

There are two kinds of clients, namely private and business. The business provided by private clients is on an ad hoc basis. The details of private clients are collected on the first booking of a taxi. However, the business provided by business clients is more formal and involves agreeing a contract of work with the business. A contract stipulates the number of jobs that FastCabs will undertake for a fixed fee.

When a job comes into FastCabs the name, phone number and contract number (when appropriate) of the client is taken and then the pick-up date/time and pick-up/drop-off addresses are noted. Each job is allocated a unique jobID. The nearest driver to the pick-up address is called by radio and is informed of the details of the job.

When a job is completed the driver should note the mileage used and the charge made (for private clients only). If a job is not complete, the reason for the failed job should be noted.

The Director has provided some examples of typical queries that the database application for FastCabs must support.

Part I: Conceptual Model

Conceptual Model PART A – Relationships/Entities

Mission Statement

The purpose of the FastCabs database system is to maintain the data that is used and generated to support the taxi cabs service for our clients,drivers,and owners and to facilitate sharing of information within the company.

Data Requirements

1.Office

FastCabs has offices in various locations. Each office is allocated members of staff.This include a manager,who manages an office,a driver who driver the taxi, and a owners who own the taxi being driven and may also be drivers themselves.The data describing a office includes a unique office number, address (street, city, and postcode), telephone number, and the total amount of staff employed.Every office can have multiple clients and managed by one manager

2. Staff

Members of staff come in various roles that they can be assigned.In general all employees have a some shared data regardless of position. The data stored regarding each member of staff includes staff ID, name,sex,age, address, position, salary,office , and the details of the branch office at which a member of staff is currently working. The staff ID is unique across all offices of FastCabs.

3.Driver

Each office has at least one driver to complete each job.Each driver can be assigned to different Taxi.The data stored for each driver are unique DriverID,OfficeNO,OfficeLoc,TaxiID,and the total number of jobs completed.Each driver can be allocated to multiple drivers and be assigned multiple jobs.

4.Manager

Members of staff with the role of Managers are responsible for the day-to-day activities of their managed office.The data stored for managers are the officeID,which specifies which office they manage, and employees managed.OfficeID is unique across all offices.

5.Owner

Members of staff with the role of Owners are responsible for the purchasing and owning the taxi cabs driven by the FastCabs company.THe data stored for owners are total taxis owned,OwnerID ,which is a unique ID used to denote taxis owned. Owner may also be Drivers and may own multiple taxis at once.

6 .Taxi

Each office has various taxis that are used to complete the jobs.THe data stored for each taxi are the Owner,Number of drivers,A unique TaxiID,Drive that is currently assigned,and the totaljobs completed.Every taxi can only be owned by one owner but can be driven by many drivers hence the unique TaxiID.

7.Client

Client Stores the general details of a client.There are two types of Client Private and Business Client each being a subclass of Client.The data stored for each client is   a phone number,to contact client.The Client belongs to at least one office.Client needs to be either Private or Business

8.Private

A client who is a Private client.The client details are taken immediately and the job is done on a case per case basis.So data stored for a private client is the info pertaining to the client.Therefore, Name and a client ID is assigned to the private client.

9.Business

A client who is a Business client.The client details are taken to used in the contract for the jobs.So data stored a business client is the info pertaining to the client's contract.They are Business Name and address is assigned to the business client.Business can only have jobs done through a contract.

10.Contract

For jobs done for a Business the terms must be agreed upon ahead of time.So a contract detailing all the info of said jobs must be made.The data stored on the Contract are a unique contract number,,the number of jobs that were agreed upon,the fee for the jobs ,and the total miles driven for the contract

11.Job

The office clients can have jobs assigned to them and the job can be assigned to one driver as well as having one driver per job.The data stored for each job are a unique jobId that identifies each jobs,pickup date, pickup time,pick up add, pick up drop,miles driven for the job,the driverId of the assigned driver,the taxiID of the assigned taxi,and the status of the job

Transaction Requirements

1. Data entry

1. Enter the details of a new office
2. Enter the details of a new member of staff at an office
3. Enter the details of a contract between a client and Business
4. Enter the details for a new taxi and the owner.
5. Enter the details of a job for a Private Client
6. Enter the details of a new client.

7.Enter the details of new taxi Owner

8.Enter the details of the new Driver details

9.Enter the details of a new taxi

10.Enter details of a new manager

2. Data update/deletion

1. Update/delete the details of an office.
2. Update/delete the details of a member of staff at an office.
3. Update/delete the details of a given contract at a given branch for a specific Business Client.
4. Update/delete the details of of the Owner of a Taxi
5. Update/delete the details of a Taxi.
6. Update/delete the details of a Driver.
7. Update/delete the details of a client either Private or Business.
8. Update/delete the details of a maangert.
9. Update/delete the details of a job.

3. Data queries

a) The names and phone numbers of the Managers at each office.

b) The names of all female drivers based in the Glasgow office.

c) The total number of staff at each office.

d) The details of all taxis at the Glasgow office.

e) The total number of W registered taxis.

f) The number of drivers allocated to each taxi.

g) The name and number of owners with more than one taxi.

h) The full address of all business clients in Glasgow.

i) The details of the current contracts with business clients in Glasgow.

j) The total number of private clients in each city.

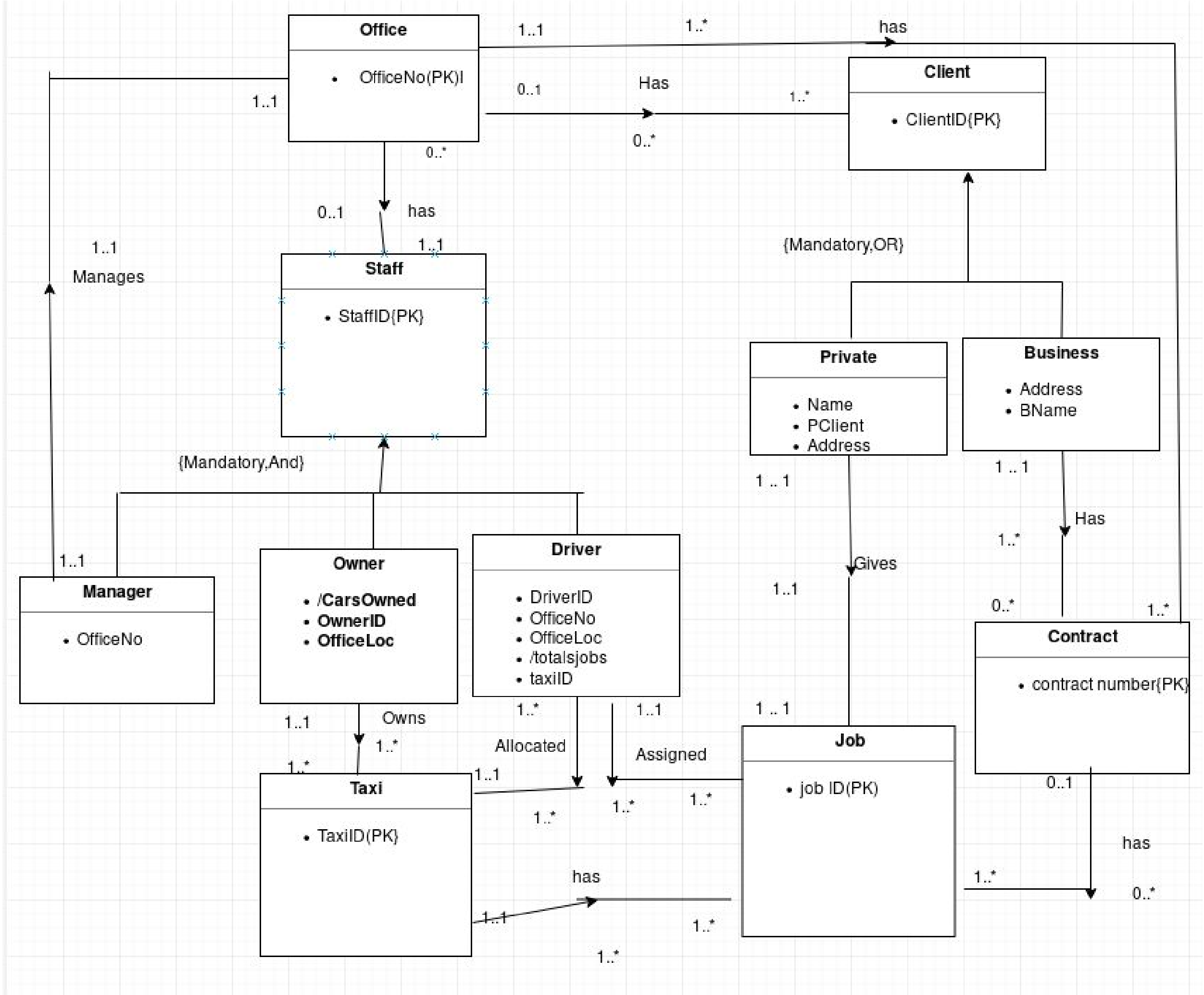
k) The details of jobs undertaken by a driver on a given day.

l) The names of drivers who are over 55 years old.

m) The names and numbers of private clients who hired a taxi in November 2000.

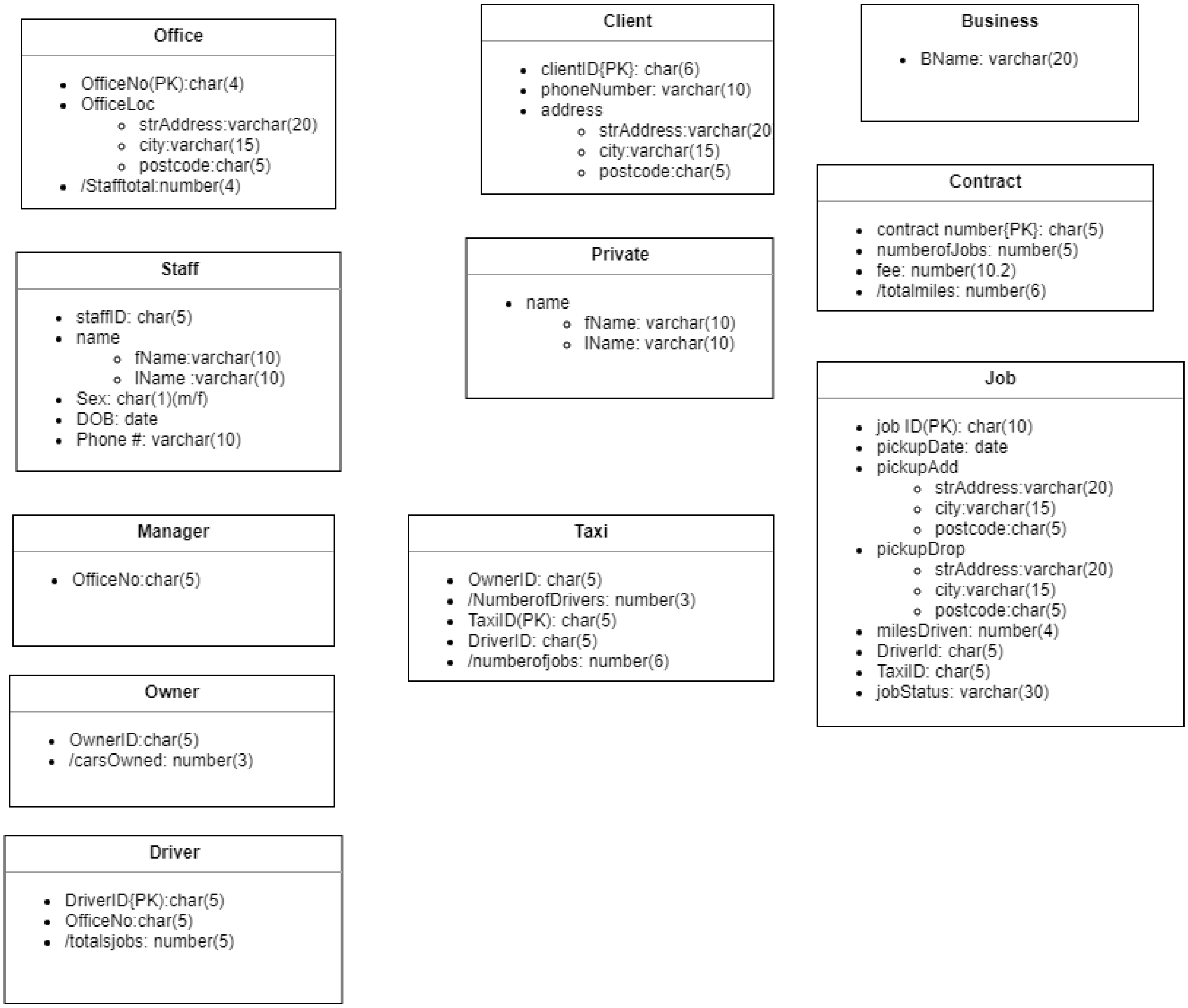
n) The names and addresses of private clients who have hired a taxi more than three times.
o) The average number of miles driven during a job.
p) The total number of jobs allocated to each car.
q) The total number of jobs allocated to each driver.
r) The total amount charged for each car in November 2000.
s) The total number of jobs and miles driven for a given contract.

D.Entity – Relationship (ER) Diagram



Conceptual Model PART B – Entities/Attributes

Entities



Office

- OfficeNo(PK):char(4)
- OfficeLoc
    - strAddress:varchar(20)
    - city:varchar(15)
    - postcode:char(5)
- /Stafftotal:number(4)

Client

- clientID{PK}: char(6)
- phoneNumber: varchar(10)
- address
    - strAddress:varchar(20
    - city:varchar(15)
    - postcode:char(5)

Business

- BName: varchar(20)

Contract

- contract number{PK}: char(5)
- numberofJobs: number(5)
- fee: number(10.2)
- /totalmiles: number(6)

Staff

- staffID: char(5)
- name
    - fName:varchar(10)
    - lName :varchar(10)
- Sex: char(1)(m/f)
- DOB: date
- Phone #: varchar(10)

Private

- name
    - fName: varchar(10)
    - lName: varchar(10)

Job

- job ID(PK): char(10)
- pickupDate: date
- pickupAdd
    - strAddress:varchar(20)
    - city:varchar(15)
    - postcode:char(5)
- pickupDrop
    - strAddress:varchar(20)
    - city:varchar(15)
    - postcode:char(5)
- milesDriven: number(4)
- DriverId: char(5)
- TaxiID: char(5)
- jobStatus: varchar(30)

Manager

- OfficeNo:char(5)

Taxi

- OwnerID: char(5)
- /NumberofDrivers: number(3)
- TaxiID(PK): char(5)
- DriverID: char(5)
- /numberofjobs: number(6)

Owner

- OwnerID:char(5)
- /carsOwned: number(3)

Driver

- DriverID{PK}:char(5)
- OfficeNo:char(5)
- /totalsjobs: number(5)

Data Dictionary

Entity Name	Attributes	Description	Data Type & length	Null
Office	**OfficeNo{PK}**	unique Office ID	char(4)	No
	officeLoc			
	strAddress	Street address	varchar(20)	No
	city	Office Location	varchar(15)	No
	postcode	Office Zipcode	char(5)	No
	/staffTotal	Derived number of staff assigned to office	number(4)	No
Staff	StaffID{PK}	Unique Staff ID	char(5)	No
	Name			
	fName	first name	varchar(10)	No
	lName	last name	varchar(10)	No
	sex	gender	char(1) (M/F)	No
	DOB	Date of birth	date	No
	Phone #	Phone number	varchar(10)	No
Manager	OfficeNo	Office number the manager is assigned to		
Owner	OwnerID	Unique Owner ID	char(5)	No
	/carsOwned	Derived number of cars owned	number(3)	No
Driver	DriverID{PK}	Unique Driver ID	char(5)	No
	OfficeNo	OfficeNo that the driver is assigned to	char(5)	No
	/totaljobs	Derived # of total jobs driver has	number(5)	Yes

		completed		
Taxi	TaxiID {PK}	Unique taxi ID	char(5)	No
	/NumberofDrivers	Derived number of drivers allocated taxi	number(3)	Yes
	OwnerID	ID of owner of taxi	char(5)	No
	DriverID	ID of current driver of taxi	char(5)	Yes
	/numberofjobs	# of jobs the taxi has been given	number(6)	Yes
Job	jobID{pk}	Unique job ID	char(5)	No
	pickupDate	Date of pick up	char(8)	Yes
	pickupADD	Pick up address		
	strAdress	Street address	varchar(20)	No
	city	Pick up Location	varchar(15)	No
	postcode	Pick up Zipcode	char(5)	No
	pickupDrop	Drop off address		
	strAdress	Street address	varchar(20)	No
	city	Drop off Location	varchar(15)	No
	postcode	Drop off Zipcode	char(5)	No
	milesDriven	Miles driven for private job	number(4)	Yes
	chargeMade	Charge made for private job	varchar(4)	Yes
	DriverId	Unique ID of driver assigned to job	char(5)	No
	jobStatus	If job failed, reason for failure	varchar(30)	Yes
	contract	If business client, contains contractNo	char(5)	Yes
	clientID	If private client,	char(6)	Yes

		contains clientID		
Client	clientID{PK}	Unique client ID	char(6)	No
	phoneNumber	Phone number of client	varchar(10)	No
	address			
	strAddress	Street address	varchar(20)	
	city	Client Location	varchar(15)	
Private	name			
	fName	first name	varchar(10)	No
	lName	last name	varchar(10)	No
Business	bName	Business name	varchar(20)	No
Contract	contractNo{PK}	Unique contract number	char(5)	No
	noJob	Number of jobs	number(5)	No
	fee	Agreed upon fee for contract	number(10.2)	No
	/totalMiles	Derived value for total miles driven for contract	number(6)	Yes
	clientID{PK}	Unique client ID	char(6)	No
	OfficeNo	unique Office ID	char(4)	No

C. BUSINESS RULES

1. Office:
   - Each office is managed by one manager, who is also a staff member
   - Each office has many drivers and owners
   - Each branch has a telephone
   - Each office has a unique ID
   - Each office has many clients and contracts
2. Staff:
   - There are three types of staff: Managers,Owners, and drivers.
   - Each staff member must be registered with an Office, and one office only at a time.

- Each Manager can manage one office

3.Manager

- Every Manager manages one Office
- Every manager is member of staff

4.Owner

- Every owner has a total number of cars to be displayed
- Every owner as a unique ID
- Each owner can own multiple cars

5.Driver

- Each Driver has a unique ID
- Every Driver is associated with an office
- Driver can drive multiple cars but can only drive one at a time per job
- Driver can have multiple jobs

6. Client

- Each client must be either Private or Business
- Every Client needs a number
- A client must belong to one office

7.Business

- Business has an address and an ID
- Each Business can have multiple contracts
- Business' can only have jobs done through contracts

8.Private Client

- Client needs to have a name and an ID
- Each private client can have one 1 job at a time

9.Taxi

- Each Taxi can be driven by multiple drivers but only one at a time
- Each taxi can have 1 or more jobs assigned overall but only one at a time

10.Job

- Multiple jobs can be assigned to a contract
- Each job can only have one of each attribute per job
- Each job can only have one Taxi assigned to it
- Each job can only have one driver assigned to it

11.Contract

- Contract can only be assigned to Business Clients
- Each contract can have multiple jobs
- Each Contract has unique ID
- Each contract is assigned to one Office

k. Common Comments on Project – Conceptual Model Phase

A.Difficulty in properly identifying all the attributes associated with an entity and how they all relate to one another.Reading the project description carefully to determine the attributes.In addition,the eer diagram complicated the diagram abit and the entities.Initially confused on how to treat a subclass

B.Likes: how all the general structure of the database is laid out, the project description provides all the information needed for this step, the entities and relationships are clearly defined

Dislikes: Some entities have a large number of attributes and some of these attributes were difficult to derive.

C. The most challenging aspect of this part of the project was organizing all of the entities and their correct attributes. Also figuring out the relationships and multiplicities between all the entities and their subentities.

D. A suggestion would be to make a more clear project description a lot of vagueness

I. Personal Comments on Project – Conceptual Model Phase

A. Paul Alvarez

1.The main parts I contributed were the Data requirement,Transaction requirements and the Relationship Diagram.The data requirement and transaction requirement both were related and both allowed me to get more insight on what was required of the database

2.Phillip and Saker works allowed another perspective on my areas usually expanding upon my own work.

B. Phillip Nahhas

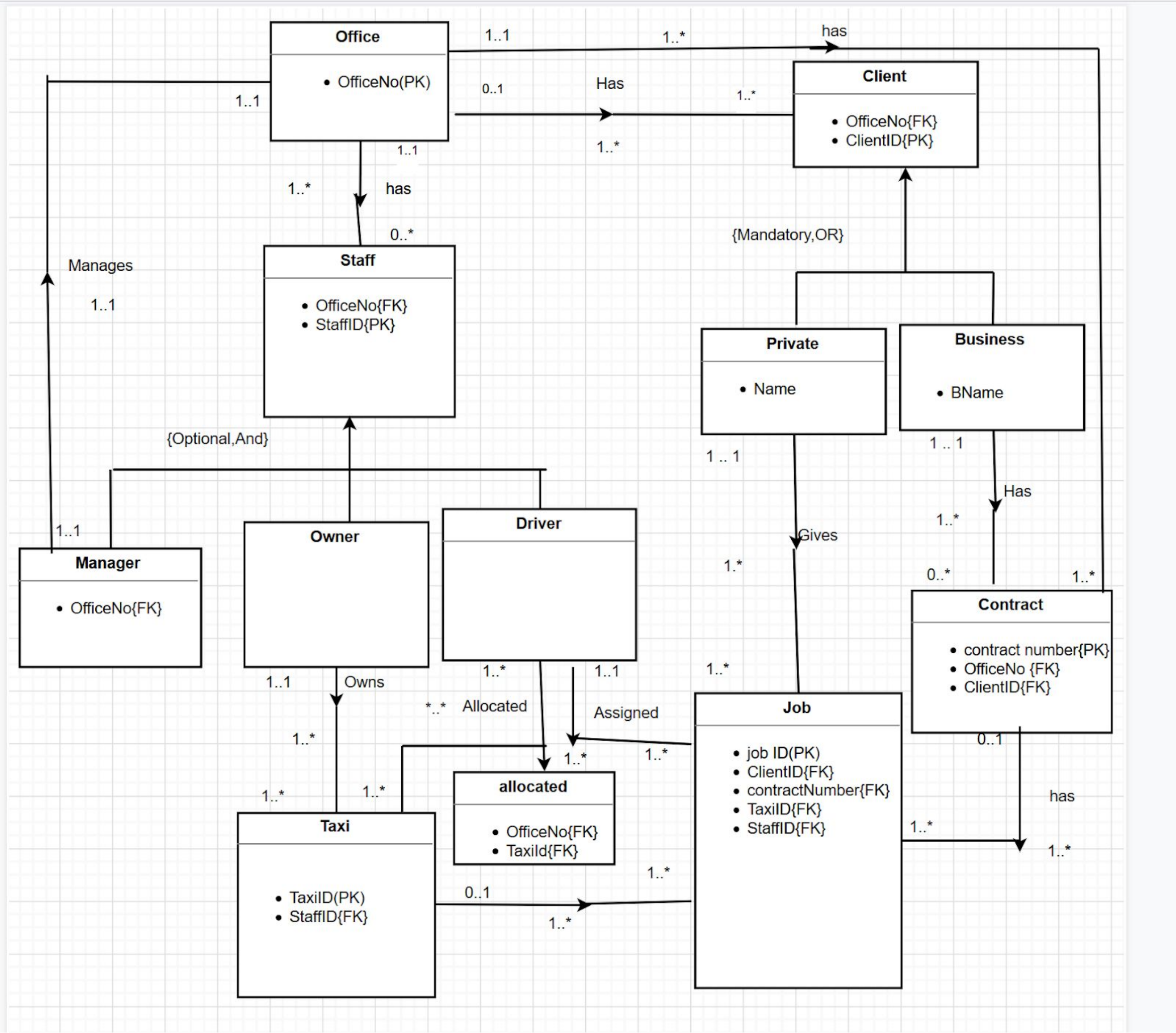
1. The main parts I contributed to were the EER diagram, the Entities and their Attributes and the Data Dictionary. These parts were interesting to work with because each part built upon the last until we ended up with a complete description of the data required for the project.
2. The Data requirements, transaction requirements and business rules that my partners worked on were very helpful to guiding the structure of the ER diagram and ensuring the entities had all the correct attributes they needed.

C.Saker Awad

1. I contributed by adding to the sections made by more partners.I added more attributes to the entities
2. The work done by my partners helped by all stacking on each other

PART II: RELATIONAL MODEL

A.Global ER / Relation Diagram



B.Relations / Attributes

Office(OfficeNo,strAddress,city,zipcode,postcode) Primary Key OfficeNo Derived stafftotal(Count(staff.StaffID))	Staff(StaffID,OfficeNo,fname,lname,sex,DOB,Pho) Primary Key StaffID Foreign Key OfficeNo references Office(OfficeNo)
Staffdetails(Stataffdetails(StaffId,bonus,DriverID) Primary Key StaffID Foreign Key StaffID references Staff(StaffID) Derived carsOwned(Count(Taxi.OwnerID== Owner.OwnerID) Derived totaljobs(Count(job.staff_id =Staff.staff_id)	ClientPrivate(ClientID,phoneNo,straddress ,city,name) Primary Key ClientID Foreign Key OfficeNo references Office(OfficeNo)
ClientBusiness(ClientID,Bname,phoneNo,strAddress,city) Primary Key ClientID Foreign Key OfficeNo references Office(OfficeNo)	Job(jobID,pickupDate,strAddress,city,postCode,milesDriven,chargemade,jobStatus,,TaxiID,ClientID,contractNumber, StaffID) Primary Key jobID Foreign Key TaxiID references Taxi(TaxiID) Foreign Key ClientID references ClientPrivate(ClientID) Foreign key contractNumber references Contract(contractNumber) Foreign Key StaffID references Staff(StaffID)
Contract(contractNumber,noJobs,fee) Primary Key contractNumber Foreign Key OfficeNo references Office(OFficeNo) Foreign Key clientID references ClientBusiness(clientID) Derived totalmiles(JOB.SUM(milesDriven))	Allocated(TaxiID,StaffID,allocatedDate) Primary Key TaxiID,StaffID Foreign Key TaxiID references Taxi(TaxiID) Foreign Key StaffID references Staff(StaffID)
Taxi(TaxiID,ownerID,DriverID,StaffID) Primary Key TaxiID FOreign Key StaffId references Staff(staffID) Derived numberofJobs(job.count(job.TaxiID=Taxi.TaxiID)	

Derived numberofdrivers(SUM(Staff.DriverID=driver.driverID)	

Common Comments on Project

A. Determining which was the parent and child entities and then creating the proper relational schema was a bit difficult.Then putting the proper primary key and foreign keys.Resolved this by looking over the notes and comparing it with the provided example
B. Since it was all up to our implementation sometimes it was difficult to gauge if the work was done properly could use a bit more clarification on the proper ways of doing a relational schema table
C. Most challenging was reassessing the previous sections work and deciding what needed to stay and what needed to go.Making the relational schema showed us that some parts previously had data redundancy.
D. Some more clear guidelines on the relational table

I. Personal Comments on Project

A. Paul Alvarez

1.I made adjustments to the er diagram adding missing tables and also figured out the Parent and child entities from the multiplicity and provided some of the initial work on the relational schema

2.Phillip and Saker works allowed another perspective on my areas usually expanding upon my own work.

B. Saker Awad

1. I assisted Paul with his work, tried to give as much knowledge as I can.
2. Paul adjusted the er diagram, adding bits and pieces to the tables. Philip helped .

C.Phillip Nahhas

1. My main contributions to this phase of the project was updating the data dictionary to correct our mistakes from phase 1. I also assisted with some of the entries in the Relations / attributes.
2. Paul and Saker’s contributions helped with my understanding of the project and relational schema.

CALIFORNIA STATE UNIVERSITY, SAN BERNARDINO
SCHOOL OF COMPUTER SCIENCE & ENGINEERING

Database Design Project

CSE 572 Spring 2019

PROJECT OPTION
Dream home

PART III: IMPLEMENTATION IN ORACLE

Project Tables

A. Dictionary

```
--fast cabs report
set pause off
spool fastcabsreport.txt
SELECT * FROM user_tab_comments;
SELECT * FROM user_col_comments WHERE table_name ='OFFICE';
SELECT * FROM user_col_comments WHERE table_name ='STAFF';
SELECT * FROM user_col_comments WHERE table_name ='STAFFDETAILS';
SELECT * FROM user_col_comments WHERE table_name ='CLIENTPRIVATE';
SELECT * FROM user_col_comments WHERE table_name ='CLIENTBUSINESS';
SELECT * FROM user_col_comments WHERE table_name ='CONTRACT';
SELECT * FROM user_col_comments WHERE table_name ='JOB';
SELECT * FROM user_col_comments WHERE table_name ='ALLOCATED';
SELECT * FROM user_col_comments WHERE table_name ='TAXI';

desc Office;
desc Staff;
desc Staffdetails;
desc ClientPrivate;
desc ClientBusiness;
desc Contract;
desc Job;
desc Allocated;
desc Taxi;

SELECT * FROM Office;
SELECT * FROM Staff;
SELECT * FROM Staffdetails;
SELECT * FROM ClientPrivate;
SELECT * FROM ClientBusiness;
SELECT * FROM Contract;
SELECT * FROM Job;
SELECT * FROM Allocated;
SELECT * FROM Taxi;

SPOOL OFF
SET PAUSE ON
```

```
TABLE_NAME                      TABLE_TYPE COMMENTS
------------------------------ ----------- ----------------------------------------------------------------
TAXIJOB                        VIEW        Total number of jobs allocated to each Taxi
TAXI                           TABLE       Taxi cab information
STAFF                          TABLE       All active employees in the COMPANY database
PRIVATECITY                     VIEW        Total number of private clients in each city
OFFICE                         TABLE       Fast Cab office information
NOSTAFF                        VIEW        Number of staff at each branch
JOB                            TABLE       All jobs in the COMPANY database
FONTANACONTRACTS VIEW                      Business Contract Info for Fontana Office
FEMALEDRIVERS                  VIEW        Names and Info for all Female Drivers
DRIVERJOB                      VIEW        Total number of jobs allocated to each driver
CONTRACTTOTALS                 VIEW        Total number of jobs and miles driven for a given contract
CONTRACTMILES                  VIEW        Total Jobs and Miles Driven for each contract
CONTRACT                       TABLE       Business Contract Information
CLIENTPRIVATE                  TABLE       All private clients in the COMPANY database
CLIENTBUSINESS                 TABLE       Business Client
AVGMILES                       VIEW        Average number of miles driven during a job
ALLOCATEDTAXI                  VIEW        Number of Drivers Allocated to each taxi
ALLOCATED                      TABLE       All drivers allocated to taxi

18 rows selected.

TABLE_NAME                      COLUMN_NAME      COMMENTS
------------------------------ ---------------------- --------------------------------------------------
OFFICE                         OFFICENO          Unique Office NO
OFFICE                         STRADDRESS        Office Street Address
OFFICE                         CITY              Office City Location
OFFICE                         POSTCODE          Office Zipcode

4 rows selected.


STAFF                          STAFFID           Employee’s unique staffID
STAFF                          FNAME             Employee’s first name
STAFF                          LNAME             Employee’s last name
STAFF                          SEX               Employee’s sex
STAFF                          DOB               Employee’s date of birth
STAFF                          PHONE             Employee’s phone number
STAFF                          OFFICENO          Employee’s office

7 rows selected.

STAFFDETAILS                   STAFFID           Staff ID
STAFFDETAILS                   BONUS             Staff Bonus
STAFFDETAILS                   DRIVERID          Staff Driver ID
```

```
3 rows selected.

CLIENTPRIVATE       CLIENTID           Client’s unique client ID
CLIENTPRIVATE      PHONENO            Clients phone number
CLIENTPRIVATE      STRADDRESS         Client’s address
CLIENTPRIVATE      CITY               Client’s city
CLIENTPRIVATE      NAME               Client’s name
CLIENTPRIVATE      OFFICENO            Client officeNo

6 rows selected.

CLIENTBUSINESS     CLIENTID           Unqiue ID of Client
CLIENTBUSINESS     BNAME              Name of Business
CLIENTBUSINESS     PHONENO            Business phone #
CLIENTBUSINESS     STRADDRESS         Business Street Address
CLIENTBUSINESS     CITY               Business City
CLIENTBUSINESS     OFFICENO           Client officeNo

6 rows selected.

CONTRACT           CONTRACTNO         Unique Contract Number
CONTRACT           NOJOBS             Number of Jobs
CONTRACT           FEE                Agreed upon fee for contract
CONTRACT           OFFICENO           Associated office number
CONTRACT           CLIENTID           Associated client ID

5 rows selected.


JOB                JOBID              Job’s unique client ID
JOB                PICKUPDATE         Date job is done
JOB               STRADDRESS          Job locations
JOB               CITY                Job’s city
JOB               POSTCODE            Job’s postcode
JOB               MILESDRIVEN         Job’s mile driven
JOB               CHARGEMADE          Jobs fee
JOB               TAXIID              Job’s assigned taxi
JOB               CLIENTID            Job’s assigned client
JOB               CONTRACTNO          Job’s contract number
JOB               STAFFID             Job’s assigned staff
JOB               JOBSTATUS           Client’s address

12 rows selected.

ALLOCATED          TAXIID             Drivers allocated Taxi
ALLOCATED          STAFFID            Taxis allocated driver
ALLOCATED          ALLOCATEDDATE      Date driver is assigned to a driver
```

```
3 rows selected.

TAXI                TAXIID              Unique Taxi ID
TAXI                STAFFID             ID of owner
TAXI                DRIVERID            ID of current Driver

3 rows selected.
```

B. SQL Data Definition

```
-- CALIFORNIA STATE UNIVERSITY, SAN BERNARDINO
-- DEPARTMENT OF COMPUTER SCIENCE
-- Course: CSE572
-- Student Names: Phillip Nahhas, Paul Alvarez
--
-- FASTCABS DATABASE PROJECT
-- CREATION OF TABLES

SET CONSTRAINTS ALL DEFERRED;
DROP TABLE Office CASCADE CONSTRAINTS;
CREATE TABLE Office(
        officeNo CHAR(4)  ,
        strAddress VARCHAR(20) UNIQUE NOT NULL,
        City VARCHAR(15)   NOT NULL,
        postcode char(5) NOT NULL,
        CONSTRAINTS Office_PK PRIMARY KEY (officeNo)
);

DROP TABLE Staff CASCADE CONSTRAINTS;
CREATE TABLE Staff(
        staffId CHAR(5) ,
        fname VARCHAR(10) NOT NULL,
        lname VARCHAR(10) NOT NULL,
        sex   char(1) NOT NULL,
        DOB   DATE NOT NULL,
        Phone VARCHAR(10) NOT NULL,
        officeNo CHAR(4),
        CONSTRAINTS Staff_PK PRIMARY KEY (staffID)
);

DROP TABLE Staffdetails CASCADE CONSTRAINTS;
CREATE TABLE Staffdetails(
        staffId CHAR(5) ,
        bonus VARCHAR(5),
        DriverID CHAR(5),
        officeNo CHAR(4),
        CONSTRAINTS Staffdetails_PK PRIMARY KEY(staffID)
);

DROP TABLE ClientPrivate CASCADE CONSTRAINTS;
CREATE TABLE ClientPrivate(
        ClientID CHAR(5),
        phoneNo VARCHAR(10),
        strAddress VARCHAR2(20) NOT NULL,
        city VARCHAR(15) NOT NULL,
        name VARCHAR(20),
        officeNo CHAR(4),
        CONSTRAINTS ClientPrivate_PK PRIMARY KEY (ClientID)
);

DROP TABLE ClientBusiness CASCADE CONSTRAINTS;
CREATE TABLE ClientBusiness(
        ClientID CHAR(5),
        Bname VARCHAR(20),
        phoneNo VARCHAR(10) NOT NULL,
        strAddress VARCHAR(20) NOT NULL,
        city VARCHAR(15),
      officeNo CHAR(4),
        CONSTRAINTS ClientBusiness_PK PRIMARY KEY (ClientID)
);

DROP TABLE Job CASCADE CONSTRAINTS;
CREATE TABLE Job(
        jobID CHAR(5),
        pickupDate DATE,
        strAddress VARCHAR(20) NOT NULL,
        city VARCHAR(15) NOT NULL,
        postCode CHAR(5),
        milesDriven NUMBER(4) NOT NULL,
        chargemade VARCHAR(4) NOT NULL,
        TaxiID CHAR(5),
        ClientID CHAR(5),
        contractNo CHAR(5),
        staffID CHAR(5),
        jobStatus VARCHAR2(30),
        CONSTRAINTS Job_PK PRIMARY KEY (jobID)
);

DROP TABLE Contract CASCADE CONSTRAINTS;
CREATE TABLE Contract(
        contractNo CHAR(5),
        noJobs NUMBER(5),
        fee NUMBER(10,2) NOT NULL,
        officeNo CHAR(4),
        ClientID CHAR(5)NOT NULL,
        CONSTRAINTS Contract_PK PRIMARY KEY (contractNo)
);

DROP TABLE Allocated CASCADE CONSTRAINTS;
CREATE TABLE Allocated(
        TaxiID CHAR(5),
        staffID CHAR(5),
        allocatedDate DATE NOT NULL,
        CONSTRAINTS Allocated_PK PRIMARY KEY (TaxiID,StaffID)
);
```

```
DROP TABLE Taxi CASCADE CONSTRAINTS;
CREATE TABLE Taxi(
        TaxiID CHAR(5),
        staffID CHAR(5),
        DriverID CHAR(5),
        CONSTRAINTS Taxi_PK PRIMARY KEY (TaxiID)
);

SET CONSTRAINTS ALL IMMEDIATE;
```

```
-- CALIFORNIA STATE UNIVERSITY, SAN BERNARDINO
-- DEPARTMENT OF COMPUTER SCIENCE
-- Course: CSE572
-- Student Names: Phillip Nahhas, Paul Alvarez
--
-- FASTCABS DATABASE PROJECT
-- ALTERING TABLE ADDING CONSTRAINTS

SET CONSTRAINTS ALL DEFERRED;

ALTER TABLE Staff
   ADD CONSTRAINTS Staff_sex_check  CHECK(sex IN('F','M'));

ALTER TABLE Staffdetails
         ADD CONSTRAINTS Staffdetails_staffID_FK FOREIGN KEY (staffID) REFERENCES
Staff(staffID)
         ON DELETE CASCADE DEFERRABLE INITIALLY IMMEDIATE;

ALTER TABLE Staffdetails
         ADD CONSTRAINTS Staffdetails_officeNo_FK FOREIGN KEY (officeNo)
REFERENCES Office(officeNo)
         ON DELETE CASCADE DEFERRABLE INITIALLY IMMEDIATE;

ALTER TABLE ClientPrivate
         ADD CONSTRAINTS ClientPrivate_officeNo_FK FOREIGN KEY(officeNo)
REFERENCES Office(officeNo)
         ON DELETE SET NULL DEFERRABLE INITIALLY IMMEDIATE;

ALTER TABLE ClientBusiness
         ADD CONSTRAINTS ClientBusiness_officeNo_FK FOREIGN KEY(officeNo)
REFERENCES Office(officeNo)
         ON DELETE SET NULL DEFERRABLE INITIALLY IMMEDIATE;

ALTER TABLE Staff
   ADD CONSTRAINTS Staff_officeNo_FK FOREIGN KEY(officeNo) REFERENCES
Office(officeNo)
      ON DELETE CASCADE DEFERRABLE INITIALLY IMMEDIATE;

ALTER TABLE Job
         ADD CONSTRAINTS Job_TaxiID_FK FOREIGN KEY(taxiID) REFERENCES
Taxi(TaxiID)
         ON DELETE SET NULL DEFERRABLE INITIALLY IMMEDIATE;

ALTER TABLE Job
         ADD CONSTRAINTS Job_clientID_FK FOREIGN KEY(ClientID) REFERENCES
ClientPrivate(ClientID)
         ON DELETE SET NULL DEFERRABLE INITIALLY IMMEDIATE;
```

```
ALTER TABLE Job
        ADD CONSTRAINTS Job_contractNo_FK FOREIGN KEY(contractNo) REFERENCES
Contract(contractNo)
        ON DELETE SET NULL DEFERRABLE INITIALLY IMMEDIATE;

ALTER TABLE Job
        ADD CONSTRAINTS Job_StaffID_FK FOREIGN KEY(StaffID) REFERENCES
Staff(StaffID)
        ON DELETE SET NULL DEFERRABLE INITIALLY IMMEDIATE;

ALTER TABLE Contract
        ADD CONSTRAINTS Contract_OfficeNo_FK FOREIGN KEY(officeNo) REFERENCES
Office(officeNo)
        ON DELETE SET NULL DEFERRABLE INITIALLY IMMEDIATE;

 ALTER TABLE Contract
        ADD CONSTRAINTS Contract_ClientID_FK FOREIGN KEY(ClientID) REFERENCES
ClientBusiness(ClientID)
        ON DELETE SET NULL DEFERRABLE INITIALLY IMMEDIATE;

ALTER TABLE Allocated
        ADD CONSTRAINTS Allocated_TaxiID_FK FOREIGN KEY(TaxiID) REFERENCES
Taxi(TaxiID)
        ON DELETE SET NULL DEFERRABLE INITIALLY IMMEDIATE;

ALTER TABLE Allocated
        ADD CONSTRAINTS Allocated_StaffID_FK FOREIGN KEY(StaffID) REFERENCES
Staff(StaffID)
        ON DELETE SET NULL DEFERRABLE INITIALLY IMMEDIATE;

ALTER TABLE Taxi
        ADD CONSTRAINTS Taxi_StaffID_FK FOREIGN KEY(StaffID) REFERENCES
Staff(StaffID)
        ON DELETE SET NULL DEFERRABLE INITIALLY IMMEDIATE;


SET CONSTRAINTS ALL IMMEDIATE;
```

```
-- CALIFORNIA STATE UNIVERSITY, SAN BERNARDINO
-- DEPARTMENT OF COMPUTER SCIENCE
-- Course: CSE572
-- Student Names: Phillip Nahhas, Paul Alvarez
--
-- FASTCABS DATABASE PROJECT
-- TABLE AND COLUMN COMMENTS

--Table Office
COMMENT ON TABLE OFFICE IS 'Fast Cab office information';
COMMENT ON COLUMN Office.officeNo IS 'Unique Office NO';
COMMENT ON COLUMN Office.strAddress IS 'Office Street Address';
COMMENT ON COLUMN Office.city IS 'Office City Location';
COMMENT ON COLUMN Office.postcode IS 'Office Zipcode';

--Table Staffdetails
COMMENT ON TABLE Staffdetails IS 'Staff Details';
COMMENT ON COLUMN Staffdetails.StaffID IS 'Staff ID';
COMMENT ON COLUMN Staffdetails.bonus IS 'Staff Bonus';
COMMENT ON COLUMN Staffdetails.DriverID IS 'Staff Driver ID';

--Table ClientBusiness
COMMENT ON TABLE ClientBusiness IS 'Business Client';
COMMENT ON COLUMN ClientBusiness.ClientID IS 'Unqiue ID of Client';
COMMENT ON COLUMN ClientBusiness.Bname IS 'Name of Business';
COMMENT ON COLUMN ClientBusiness.phoneNo IS 'Business phone #';
COMMENT ON COLUMN ClientBusiness.strAddress IS 'Business Street Address';
COMMENT ON COLUMN ClientBusiness.city IS 'Business City';
COMMENT ON COLUMN ClientBusiness.officeNo IS 'Client officeNo';

--Table Contract
COMMENT ON TABLE Contract IS 'Business Contract Information';
COMMENT ON COLUMN Contract.contractNo IS 'Unique Contract Number';
COMMENT ON COLUMN Contract.noJobs IS 'Number of Jobs';
COMMENT ON COLUMN Contract.fee IS 'Agreed upon fee for contract';
COMMENT ON COLUMN Contract.officeNo IS 'Associated office number';
COMMENT ON COLUMN Contract.clientID IS 'Associated client ID';

--Table Taxi
COMMENT ON TABLE Taxi IS 'Taxi cab information';
COMMENT ON COLUMN Taxi.TaxiID IS 'Unique Taxi ID';
COMMENT ON COLUMN Taxi.staffID IS 'ID of owner';
COMMENT ON COLUMN Taxi.DriverID IS 'ID of current Driver';
COMMENT ON COLUMN Taxi.allocatedDate IS 'Date the taxi is Allocated to driver';

--Table Staff
COMMENT ON TABLE Staff IS 'All active employees in the COMPANY database';
COMMENT ON COLUMN Staff.StaffID IS 'Employee’s unique staffID';
COMMENT ON COLUMN Staff.fname IS 'Employee’s fisrt name';
COMMENT ON COLUMN Staff.lname IS 'Employee’s last name';
COMMENT ON COLUMN Staff.sex IS 'Employee’s sex';
COMMENT ON COLUMN Staff.DOB IS 'Employee’s date of birth';
COMMENT ON COLUMN Staff.Phone IS 'Employee’s phone number';
COMMENT ON COLUMN Staff.officeNo IS 'Employee’s office ';

--Table ClientPrivate
COMMENT ON TABLE ClientPrivate IS 'All private clients in the COMPANY database';
COMMENT ON COLUMN ClientPrivate.ClientID IS 'Client’s unique client ID';
COMMENT ON COLUMN ClientPrivate.phoneNo IS 'Clients phone number';
COMMENT ON COLUMN ClientPrivate.strAddress IS 'Client’s address';
COMMENT ON COLUMN ClientPrivate.city IS 'Client’s city';
COMMENT ON COLUMN ClientPrivate.name IS 'Client’s name';
COMMENT ON COLUMN ClientPrivate.officeNo IS 'Client officeNo';

--Table Job
COMMENT ON TABLE Job IS 'All jobs in the COMPANY database';
COMMENT ON COLUMN Job.jobID IS 'Job’s unique client ID';
COMMENT ON COLUMN Job.pickupDate IS 'Date job is done';
COMMENT ON COLUMN Job.strAddress IS 'Job locations';
COMMENT ON COLUMN Job.city IS 'Job’s city';
COMMENT ON COLUMN Job.postCode IS 'Job’s postcode';
COMMENT ON COLUMN Job.milesDriven IS 'Job’s mile driven';
COMMENT ON COLUMN Job.chargemade IS 'Jobs fee';
COMMENT ON COLUMN Job.jobStatus IS 'Client’s address';
COMMENT ON COLUMN Job.TaxiID IS 'Job’s assigned taxi';
COMMENT ON COLUMN Job.ClientID IS 'Jobs’s assigned client';
COMMENT ON COLUMN Job.contractNo IS 'Jobs’s contract number';
COMMENT ON COLUMN Job.StaffID IS 'Job’s assigned staff';

--Table Allocated
COMMENT ON TABLE Allocated IS 'All drivers allocated to taxi';
COMMENT ON COLUMN Allocated.TaxiID IS 'Drivers allocated Taxi';
COMMENT ON COLUMN Allocated.StaffID IS 'Taxis allocated driver';
COMMENT ON COLUMN Allocated.allocatedDate IS 'Date driver is assigned to a driver';

--View Comments
COMMENT ON MATERIALIZED VIEW noStaff IS 'Number of staff at each branch';
COMMENT ON MATERIALIZED VIEW privateCity IS 'Total number of private clients in each
city';
COMMENT ON MATERIALIZED VIEW femaleDrivers IS 'Names and Info for all Female
Drivers';
COMMENT ON MATERIALIZED VIEW contractMiles IS 'Total Jobs and Miles Driven for each
contract';
COMMENT ON MATERIALIZED VIEW fontanaContracts IS 'Business Contract Info for Fontana
Office';
```

```
COMMENT ON MATERIALIZED VIEW allocatedTaxi IS 'Number of Drivers Allocated to each taxi';
COMMENT ON MATERIALIZED VIEW driverJob IS 'Total number of jobs allocated to each driver';
COMMENT ON MATERIALIZED VIEW taxiJob IS 'Total number of jobs allocated to each Taxi';
COMMENT ON MATERIALIZED VIEW avgMiles IS 'Average number of miles driven during a job';
COMMENT ON MATERIALIZED VIEW contractTotals IS 'Total number of jobs and miles driven for a given contract';
```

C. Table Structures

```
SQL> desc Office;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 OFFICENO                                  NOT NULL CHAR(4)
 STRADDRESS                                NOT NULL VARCHAR2(20)
 CITY                                      NOT NULL VARCHAR2(15)
 POSTCODE                                  NOT NULL CHAR(5)

SQL> desc Staff;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 STAFFID                                   NOT NULL CHAR(5)
 FNAME                                     NOT NULL VARCHAR2(10)
 LNAME                                     NOT NULL VARCHAR2(10)
 SEX                                       NOT NULL CHAR(1)
 DOB                                       NOT NULL DATE
 PHONE                                     NOT NULL VARCHAR2(10)
 OFFICENO                                           CHAR(4)

SQL> desc Staffdetails;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 STAFFID                                   NOT NULL CHAR(5)
 BONUS                                              VARCHAR2(5)
 DRIVERID                                           CHAR(5)
 OFFICENO                                           CHAR(4)

SQL> desc ClientPrivate;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 CLIENTID                                  NOT NULL CHAR(5)
 PHONENO                                            VARCHAR2(10)
 STRADDRESS                                NOT NULL VARCHAR2(20)
 CITY                                      NOT NULL VARCHAR2(15)
 NAME                                               VARCHAR2(20)
 OFFICENO                                           CHAR(4)

SQL> desc ClientBusiness;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 CLIENTID                                  NOT NULL CHAR(5)
 BNAME                                              VARCHAR2(20)
 PHONENO                                   NOT NULL VARCHAR2(10)
 STRADDRESS                                NOT NULL VARCHAR2(20)
 CITY                                               VARCHAR2(15)
 OFFICENO                                           CHAR(4)

SQL> desc Contract;
```

```
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 CONTRACTNO                               NOT NULL CHAR(5)
 NOJOBS                                             NUMBER(5)
 FEE                                       NOT NULL NUMBER(10,2)
 OFFICENO                                           CHAR(4)
 CLIENTID                                  NOT NULL CHAR(5)

SQL> desc Job;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 JOBID                                     NOT NULL CHAR(5)
 PICKUPDATE                                         DATE
 STRADDRESS                                NOT NULL VARCHAR2(20)
 CITY                                      NOT NULL VARCHAR2(15)
 POSTCODE                                           CHAR(5)
 MILESDRIVEN                               NOT NULL NUMBER(4)
 CHARGEMADE                                NOT NULL VARCHAR2(4)
 TAXIID                                             CHAR(5)
 CLIENTID                                           CHAR(5)
 CONTRACTNO                                         CHAR(5)
 STAFFID                                            CHAR(5)
 JOBSTATUS                                          VARCHAR2(30)

SQL> desc Allocated;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 TAXIID                                    NOT NULL CHAR(5)
 STAFFID                                   NOT NULL CHAR(5)
 ALLOCATEDDATE                             NOT NULL DATE

SQL> desc Taxi;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 TAXIID                                    NOT NULL CHAR(5)
 STAFFID                                            CHAR(5)
 DRIVERID                                           CHAR(5)

SQL> spool off
```

D. Table Contents

```
-- CALIFORNIA STATE UNIVERSITY, SAN BERNARDINO
-- DEPARTMENT OF COMPUTER SCIENCE
-- Course: CSE572
-- Student Names: Phillip Nahhas, Paul Alvarez
--
-- FASTCABS DATABASE PROJECT
-- INSERTION OF DATA INTO TABLES

SET CONSTRAINTS ALL DEFERRED;

INSERT INTO Office VALUES ('B001','1234 Foothill Ave','Fontana', '92336');
INSERT INTO Office VALUES ('B002','6789 Baseline Ave','Moreno Valley', '92551');

INSERT INTO Staff VALUES ('S0001','David','Anguiano','M','20-Feb-90','3425555','B001');
INSERT INTO Staff VALUES ('S0002','Felipe','Valadez','F','20-Mar-86','4435555','B001');
INSERT INTO Staff VALUES ('S0003','Hector','Madrigal','M','13-Oct-91','1235555','B002');
INSERT INTO Staff VALUES ('S0004','Rocio','Tabatai','F','10-Sep-87','8795555','B002');
INSERT INTO Staff VALUES ('S0005','Danielle','Nolan','F', '13-Mar-65','5675555','B001');

INSERT INTO Staffdetails VALUES ('S0004',NULL,'D0001', NULL);
INSERT INTO Staffdetails VALUES ('S0002',NULL,'D0002',NULL);
INSERT INTO Staffdetails VALUES ('S0005',NULL,NULL,'B001');
INSERT INTO Staffdetails VALUES ('S0001',NULL,NULL,'B002');

INSERT INTO ClientPrivate VALUES ('C0001','9095551000','123 University Ave','San
Bernardino','Jan Levninson','B001');
INSERT INTO ClientPrivate VALUES ('C0002','9095551001','456 Citrus Ave','Fontana','Oscar
Martinez', 'B002');
INSERT INTO ClientPrivate VALUES ('C0003','9095551002','789 Siera Ave','Rancho','Holly
Flax','B001');
INSERT INTO ClientPrivate VALUES ('C0004','9095551003','321 Day Creek
Ave','Riverside','Creed Braton', 'B002');

INSERT INTO ClientBusiness VALUES('C1001','Apple','9515551234','111 Tyler
Ave','Riverside','B001');
INSERT INTO ClientBusiness VALUES('C1002','Dell','9515551111','222 Kendal Ave','San
Bernardino','B002');
INSERT INTO ClientBusiness VALUES('C1003','Intel','9515552222','333 Highland
Ave','Victorville','B001');

INSERT INTO Contract VALUES ('A0001',100,50,'B001','C1001');
INSERT INTO Contract VALUES ('A0002',200,50,'B002','C1002');
INSERT INTO Contract VALUES ('A0003',300,50,'B001','C1003');



INSERT INTO Job   VALUES ('J0001','08-Jan-19','111 First
Ave','Fontana','92336',10,'25','T0001','C0001',NULL,'S0004','Complete');
```

```
INSERT INTO Job   VALUES ('J0002','08-Jan-19','222 Second
Ave','Victorville','92356',40,'50','T0001',NULL,'A0001','S0004','Complete');
INSERT INTO Job   VALUES ('J0003','22-May-19','333 Third
ST','Upland','92367',10,'25','T0001','C0003',NULL,'S0002','Incomplete, Taxi flat tire');
INSERT INTO Job   VALUES ('J0004','22-May-19','444 Fourth
St','Riverside','92054',20,'50','T0001',NULL,'A0002','S0002','Complete');
INSERT INTO Job   VALUES ('J0005','01-Jun-19','555 Fifth St','Moreno
Valley','92059',80,'200','T0002','C0004',NULL,'S0004','Complete, Overtime');
INSERT INTO Job   VALUES ('J0006','20-Apr-19','666 Sixth
St','Ontario','92330',5,'50','T0004',NULL,'A0003','S0002','Complete');



INSERT INTO Allocated  VALUES ('T0001','S0004','08-Jan-19');
INSERT INTO Allocated VALUES ('T0001','S0002','22-May-19');
INSERT INTO Allocated VALUES ('T0002','S0004','01-Jun-19');
INSERT INTO Allocated VALUES ('T0004','S0002','20-Apr-19');

INSERT INTO Taxi  VALUES ('T0001','S0004','D0001');
INSERT INTO Taxi VALUES ('T0002','S0004','D0002');
INSERT INTO Taxi VALUES ('T0003','S0004','D0002');
INSERT INTO Taxi VALUES ('T0004','S0004','D0002');

SET CONSTRAINTS ALL IMMEDIATE;

SET PAUSE OFF

SELECT * FROM Office;
SELECT * FROM Staff;
SELECT * FROM Staffdetails;
SELECT * FROM ClientPrivate;
SELECT * FROM ClientBusiness;
SELECT * FROM Contract;
SELECT * FROM Job;
SELECT * FROM Allocated;
SELECT * FROM Taxi;

SET PAUSE ON

SQL> @fastcabselect

OFFI STRADDRESS          CITY            POSTC
---- -------------------- --------------- -----
B001 1234 Foothill Ave   Fontana         92336
B002 6789 Baseline Ave   Moreno Valley   92551


STAFF FNAME      LNAME      S DOB      PHONE      OFFI
```

```
----- ---------- ---------- - --------- ---------- ----
S0001 David      Anguiano   M 20-FEB-90 3425555    B001
S0002 Felipe     Valadez    F 20-MAR-86 4435555    B001
S0003 Hector     Madrigal   M 13-OCT-91 1235555    B002
S0004 Rocio      Tabatai    F 10-SEP-87 8795555    B002
S0005 Danielle   Nolan      F 13-MAR-65 5675555    B001


STAFF BONUS DRIVE OFFI
----- ----- ----- ----
S0004       D0001
S0002       D0002
S0005             B001
S0001             B002


CLIEN PHONENO    STRADDRESS           CITY            NAME                 OFFI
----- ---------- -------------------- --------------- -------------------- ----
C0001 9095551000 123 University Ave   San Bernardino  Jan Levninson        B001
C0002 9095551001 456 Citrus Ave       Fontana         Oscar Martinez       B002
C0003 9095551002 789 Siera Ave        Rancho          Holly Flax           B001
C0004 9095551003 321 Day Creek Ave    Riverside       Creed Braton         B002


CLIEN BNAME                PHONENO    STRADDRESS          CITY            OFFI
----- -------------------- ---------- ------------------- --------------- ----
C1001 Apple                9515551234 111 Tyler Ave       Riverside       B001
C1002 Dell                 9515551111 222 Kendal Ave      San Bernardino  B002
C1003 Intel                9515552222 333 Highland Ave    Victorville     B001


CONTR     NOJOBS        FEE OFFI CLIEN
----- ---------- ---------- ---- -----
A0001        100         50 B001 C1001
A0002        200         50 B002 C1002
A0003        300         50 B001 C1003


JOBID PICKUPDAT STRADDRESS          CITY            POSTC MILESDRIVEN CHAR
----- --------- ------------------- --------------- ----- ----------- ----
TAXII CLIEN CONTR STAFF JOBSTATUS
----- ----- ----- ----- ------------------------------
J0001 08-JAN-19 111 First Ave       Fontana         92336          10 25
T0001 C0001       S0004 Complete

J0002 08-JAN-19 222 Second Ave      Victorville     92356          40 50
T0001       A0001 S0004 Complete
```

```
J0003 22-MAY-19 333 Third ST         Upland         92367          10 25
T0001 C0003       S0002 Incomplete, Taxi flat tire


JOBID PICKUPDAT STRADDRESS           CITY           POSTC MILESDRIVEN CHAR
----- --------- -------------------- --------------- ----- ----------- ----
TAXII CLIEN CONTR STAFF JOBSTATUS
----- ----- ----- ----- ------------------------------
J0004 22-MAY-19 444 Fourth St        Riverside      92054          20 50
T0001       A0002 S0002 Complete

J0005 01-JUN-19 555 Fifth St         Moreno Valley   92059          80 200
T0002 C0004       S0004 Complete, Overtime

J0006 20-APR-19 666 Sixth St         Ontario        92330           5 50
T0004       A0003 S0002 Complete


6 rows selected.


TAXII STAFF ALLOCATED
----- ----- ---------
T0001 S0004 08-JAN-19
T0001 S0002 22-MAY-19
T0002 S0004 01-JUN-19
T0004 S0002 20-APR-19


TAXII STAFF DRIVE
----- ----- -----
T0001 S0004 D0001
T0002 S0004 D0002
T0003 S0004 D0002
T0004 S0004 D0002

SQL> spool off
```

Project Queries

A. Queries and Views

```
-- CALIFORNIA STATE UNIVERSITY, SAN BERNARDINO
-- DEPARTMENT OF COMPUTER SCIENCE
-- Course: CSE572
-- Student Names: Phillip Nahhas, Paul Alvarez
--
-- FASTCABS DATABASE PROJECT
-- CREATION OF VIEWS

SET PAUSE OFF

--Number of staff at each branch
DROP VIEW noStaff;
CREATE VIEW noStaff AS
        SELECT COUNT(staffId) AS NO_of_Staff, officeNo FROM Staff GROUP BY officeNo;
SELECT * FROM noStaff;

--Total number of private clients in each city
DROP VIEW privateCity;
CREATE VIEW privateCity AS
        SELECT COUNT(ClientID) AS NO_of_Clients, city FROM ClientPrivate GROUP BY city;
SELECT * FROM privateCity;

--Names and Info for all Female Drivers
DROP VIEW femaleDrivers;
CREATE VIEW femaleDrivers AS
        SELECT s.staffID, s.fname, s.lname, s.sex, s.Phone, s.DOB, s.officeNo, d.DriverID
FROM Staff s, StaffDetails d WHERE s.staffID IN (SELECT staffID FROM StaffDetails WHERE
DriverID is NOT NULL) AND s.sex IN 'F' AND s.staffID = d.staffID ORDER BY s.staffID;
SELECT * FROM femaleDrivers;

--Total Jobs and Miles Driven for each contract
DROP VIEW contractMiles;
CREATE VIEW contractMiles AS
        SELECT contractNo, COUNT(contractNo) AS NO_of_Jobs, SUM(milesDriven) AS
Total_Miles_Driven FROM Job WHERE contractNo IS NOT NULL GROUP BY contractNo
ORDER BY contractNo;
SELECT * FROM contractMiles;

--Business Contract Info for Fontana Office
DROP VIEW fontanaContracts;
CREATE VIEW fontanaContracts AS
        SELECT contractNo, noJobs, Fee, ClientID, officeNo FROM Contract WHERE officeNo
IN 'B001';
SELECT * FROM fontanaContracts;

--Number of Drivers Allocated to each taxi
DROP VIEW allocatedTaxi;
CREATE VIEW allocatedTaxi AS
```

```
        SELECT COUNT(staffID) AS NO_of_Drivers, TaxiID FROM Allocated GROUP BY
TaxiID ORDER BY TaxiID;
SELECT * FROM allocatedTaxi;

--Total number of jobs allocated to each driver
DROP VIEW driverJob;
CREATE VIEW driverJob AS
        SELECT COUNT(jobID) AS NO_of_jobs, staffID FROM Job GROUP BY staffID ORDER
BY staffID;
SELECT * FROM driverJob;

--Total number of jobs allocated to each Taxi
DROP VIEW taxiJob;
CREATE VIEW taxiJob AS
        SELECT COUNT(jobID) AS NO_of_jobs, TaxiID FROM Job GROUP BY TaxiID ORDER
BY TaxiID;
SELECT * FROM taxiJob;

--Average number of miles driven during a job
DROP VIEW avgMiles;
CREATE VIEW avgMiles AS
        SELECT avg(milesDriven) AS AVG_miles_driven FROM Job;
SELECT * FROM avgMiles;

--Total number of jobs and miles driven for a given contract
DROP VIEW contractTotals;
CREATE VIEW contractTotals AS
        SELECT contractNo ,COUNT(jobID) AS NO_of_Jobs, SUM(milesDriven) AS
Total_miles_driven FROM Job WHERE contractNo IS NOT NULL GROUP BY contractNo
ORDER BY contractNo;
SELECT * FROM contractTotals;

SET PAUSE ON
```

B. Results

```
SQL> @testFastcabs

View dropped.


View created.


NO_OF_STAFF OFFI
----------- ----
          2 B002
          3 B001


View dropped.


View created.


NO_OF_CLIENTS CITY
------------- ---------------
            1 Rancho
            1 Fontana
            1 Riverside
            1 San Bernardino


View dropped.


View created.


STAFF FNAME      LNAME      S PHONE      DOB       OFFI DRIVE
----- ---------- ---------- - ---------- --------- ---- -----
S0002 Felipe     Valadez    F 4435555    20-MAR-86 B001 D0002
S0004 Rocio      Tabatai    F 8795555    10-SEP-87 B002 D0001


View dropped.


View created.


CONTR NO_OF_JOBS TOTAL_MILES_DRIVEN
----- ---------- ------------------
```

```
A0001          1         40
A0002          1         20
A0003          1          5


View dropped.


View created.


CONTR     NOJOBS        FEE CLIEN OFFI
----- ---------- ---------- ----- ----
A0001        100         50 C1001 B001
A0003        300         50 C1003 B001


View dropped.


View created.


NO_OF_DRIVERS TAXII
------------- -----
            2 T0001
            1 T0002
            1 T0004


View dropped.


View created.


NO_OF_JOBS STAFF
---------- -----
         3 S0002
         3 S0004


View dropped.


View created.
```

```
NO_OF_JOBS TAXII
---------- -----
         4 T0001
         1 T0002
         1 T0004


View dropped.


View created.


AVG_MILES_DRIVEN
----------------
            27.5


View dropped.


View created.


CONTR NO_OF_JOBS TOTAL_MILES_DRIVEN
----- ---------- ------------------
A0001          1                 40
A0002          1                 20
A0003          1                  5

SQL> spool off
```

IV. Comments on Project - Implementation Phase

A. One difficulty we faced during this phase was making sure all the data and data types were consistent between Tables. We resolved this by constantly referencing and double checking the data dictionary from phase I.

B. Likes: One part i liked about this part of the project was that all of our planning finally came together and we were able to see the finished product
Dislikes: One dislike about this phase is the amount of sql code we need to write to fully implement each table with the correct relations, comments, insertions, and queries.

C. The most challenging aspect of this part of the design was that we had to be sure that all of our data dictionary and EER diagram relations were correct from the previous phases in order for the implementation phase to be accurate and function correctly.

D. On suggestion to improve this phase of the project would be to possibly separate it into smaller sections. Implementing the entire database in one phase is a lot of work and it is easy to get burned out writing so many sql statements at once.

V.Personal Comments on Project

A. Phillip Nahhas

1. The main parts i contributed to for this phase of the project were to the table alterations/ constraints, parts of the table comments, parts of the data insertion, views/queries, and the implementation and reporting of the database.
2. The contributions of my partners parts allowed us to divide the workload and combine them at the end without problem.

B. Paul Alvarez

1. The parts I contributed to third phase the table relations , part of the table comments, part of the data insertion, and some of the table views.
2. My groups mates allowed for the more even redistribution of the project worldload where each of us would get a portion of the work and complete and then combining it for the final project.

C.Saker Awad

1.I contributed by adding to the sections made by more partners.I added more views ,some relational table and some insertion data
2.The work done by my partners helped by all stacking on each other after having worked on them separately