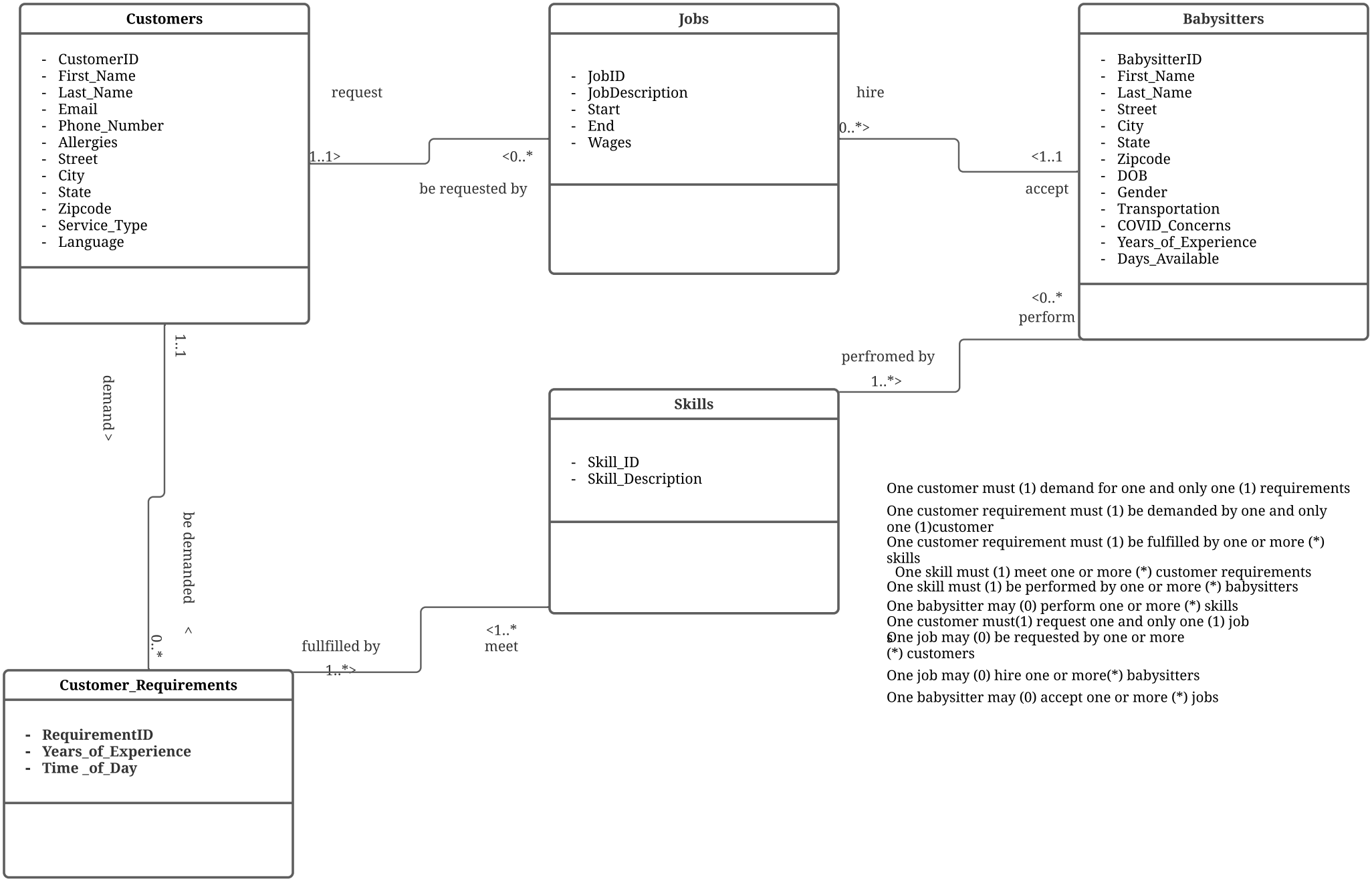

One customer must (1) demand for one and only one (1) requirements

One customer requirement must (1) be demanded by one and only one (1)customer

One customer requirement must (1) be fulfilled by one or more (*) skills

One skill must (1) meet one or more (*) customer requirements

One skill must (1) be performed by one or more (*) babysitters

One babysitter may (0) perform one or more (*) skills

One customer must(1) request one and only one (1) job

One job may (0) be requested by one or more (*) customers

One job may (0) hire one or more(*) babysitters

One babysitter may (0) accept one or more (*) jobs

NORMALIZATION PROCESS

Customer (CustomerID, First_Name, Last_Name, Email, Phone, Allergies, Street, City, State, Zipcode, Service_Type, Language)
Key: CustomerID
FD1: CustomerID -> First_Name, Last_Name, Email, Phone, Allergies, Street, City, State, Zipcode, Service_Type, Language
FD2: Zipcode -> City, State

1NF: Yes it is in 1NF because it meets all the requirements.
2NF: Yes it is in 2NF because there are no partial key dependencies.
3NF: No because there is a transitive dependency.
solution: split into two new relation, COPY Zipcode and REMOVE City, State.

Zipcode(Zipcode, City, State)
key: Zipcode
FD1: Zipcode -> City, State
1NF: Yes it is in 1NF because it split from a relation.
2NF: Yes it is in 2NF because there are no partial key dependencies.
3NF: Yes because there are no transitive dependencies.

CustomerA (CustomerID, First_Name, Last_Name, Email, Phone, Allergies, Street, Zipcode, Service_Type, Language)
key: CustomerID
FD1: CustomerID -> First_Name, Last_Name, Email, Phone, Allergies, Street, Zipcode, Service_Type, Language
1NF: Yes it is in 1NF because it split from a relation.
2NF: Yes it is in 2NF because there are no partial key dependencies.
3NF: Yes because there are no transitive dependencies.

Customer_Requirements (RequirementID, Years_of_Experience, Time_of_Day, CustomerID)
Key: RequirementID
FD1: RequirementID -> Years_of_Experience, Time_of_Day, CustomerID

1NF: Yes it is in 1NF because it meets all the requirements.
2NF: Yes it is in 2NF because there are no partial key dependencies.
3NF: Yes because there are no transitive dependencies.

Skills (Skill_ID, Skill_Description)
Key: Skill_ID
FD1: Skill_ID -> Skill_Description

1NF: Yes it is in 1NF because it meets all the requirements.
2NF: Yes it is in 2NF because there are no partial key dependencies.
3NF: Yes because there are no transitive dependencies.

Babysitter (BabysitterID, Street, City, State, Zipcode, First_Name, Last_Name, DOB, Gender ,Transportation, COVID_Concerns, Years_of_Experience, Days_Available)
Key: BabysitterID
FD1: BabysitterID -> Street, City, State, Zipcode, First_Name, Last_Name, DOB, Gender, Transportation, COVID_Concerns, Years_of_Experience, Days_Available
FD2: Zipcode -> City, State

1NF: Yes it is in 1NF because it meets all the requirements.
2NF: Yes it is in 2NF because there are no partial key dependencies.
3NF: No because there is a transitive dependency.
solution: split into two new relation, COPY Zipcode and REMOVE City, State.

Note: We already have a ZipCode relation from when the Customer relation was split up. So we re-use that ZipCode relation. There is no need to create a second ZipCode relation.

BabysutterA (BabysitterID, Street, Zipcode, First_Name, Last_Name, DOB, Gender ,Transportation, COVID_Concerns, Years_of_Experience, Days_Available)
key: BabysitterID
FD1: BabysitterID -> Street, Zipcode, First_Name, Last_Name, DOB, Gender ,Transportation, COVID_Concerns, Years_of_Experience, Days_Available

1NF: Yes it is in 1NF because it meets all the requirements.
2NF: Yes it is in 2NF because there are no partial key dependencies.
3NF: Yes because there are no transitive dependencies.

Jobs (JobID, JobDescription, Start_Date, End_Date, Wages, Babysitter_ID)
Key: JobID
FD1: JobID -> JobDescription, Start_Date, End_Date, Wages, Babysitter_ID

1NF: Yes it is in 1NF because it meets all the requirements.
2NF: Yes it is in 2NF because there are no partial key dependencies.
3NF: Yes because there are no transitive dependencies.

Customer_Requirement_Skills (RequirementID, Skill_ID)
Key: RequirementID, Skill_ID

Babysitter_Skills (Babysitter_ID, Skill_ID)
Key: Babysitter_ID, Skill_ID

Final relation:
Zipcode(Zipcode, City, State)

CustomerA (CustomerID, First_Name, Last_Name, Email, Phone, Allergies, Street, Zipcode, Service_Type, Language)

Customer_Requirements (RequirementID, Years_of_Experience, Time_of_Day, CustomerID)

Skills (SkillID, Skill_Description)

BabysutterA (BabysitterID, Street, Zipcode, First_Name, Last_Name, DOB, Gender ,Transportation, COVID_Concerns, Years_of_Experience, Days_Available)

Jobs (JobID, JobDescription, Start_Date, End_Date, Wages, BabysitterID)

Customer_Requirement_Skills (RequirementID, SkillID)

Babysitter_Skills (BabysitterID, SkillID)

Create tables

```
CREATE TABLE Zipcodes
(
    Zipcode          VARCHAR(10) NOT NULL
        CONSTRAINT pk_Zipcodes PRIMARY KEY,
   City              VARCHAR(35) NOT NULL,
   State             VARCHAR(35) NOT NULL
) ;


CREATE TABLE Customers
(
    CustomerID   VARCHAR(10) NOT NULL
        CONSTRAINT pk_Customers PRIMARY KEY,
   First_Name     VARCHAR(35) NOT NULL,
   Last_Name      VARCHAR(35) NOT NULL,
   Email          VARCHAR(100) NOT NULL,
   Phone_Number          VARCHAR(11) NOT NULL,
   Allergies       VARCHAR(100),
   Street          VARCHAR(100) NOT NULL,
   Zipcode         VARCHAR(10) NOT NULL,
   Service_Type    VARCHAR(100),
   Language       VARCHAR(100)
) ;


CREATE TABLE Customer_Requirements
(
    RequirementID          NUMBER NOT NULL
```

```
        CONSTRAINT pk_Customer_Requirements PRIMARY KEY,
    Years_of_Experience                 VARCHAR(100),
    Time_of_Day              VARCHAR(100) NOT NULL,
    CustomerID               VARCHAR(10) NOT NULL
) ;



CREATE TABLE Skills
(
     SkillID          NUMBER NOT NULL
        CONSTRAINT pk_Skills PRIMARY KEY,
    Skill_Description                VARCHAR(200) NOT NULL
) ;



CREATE TABLE Babysitters
(
     BabysitterID   VARCHAR(10) NOT NULL
        CONSTRAINT pk_Babysitters PRIMARY KEY,
    First_Name      VARCHAR(35) NOT NULL,
    Last_Name       VARCHAR(35) NOT NULL,
    Date_of_Birth       DATE,
    Gender          VARCHAR(35),
    Street          VARCHAR(100) NOT NULL,
    Zipcode         VARCHAR(10) NOT NULL,
    Transportation    VARCHAR(100),
    COVID_Concerns    VARCHAR(100) NOT NULL,
    Years_of_Experience  VARCHAR(100),
    Days_Available       VARCHAR(100)
```

```
) ;



CREATE TABLE Jobs
(
    JobID          NUMBER NOT NULL
        CONSTRAINT pk_Jobs PRIMARY KEY,
   JobDescription                VARCHAR(100),
   Start_Date          DATE,
   End_Date            DATE,
   Wages               NUMBER,
   BabysitterID                VARCHAR(10) NOT NULL
) ;
```

```
ALTER TABLE

ALTER TABLE Customers
  ADD CONSTRAINT fk_customers_zipcodes
    FOREIGN KEY (Zipcode)
      REFERENCES Zipcodes (Zipcode)


ALTER TABLE Babysitters
  ADD CONSTRAINT fk_babysitters_zipcodes
    FOREIGN KEY (Zipcode)
      REFERENCES Zipcodes (Zipcode)
ALTER TABLE Babysitters
  ADD CONSTRAINT fk_Babysitters_skillID
    FOREIGN KEY (SkillID)
      REFERENCES Skills (SkillID)


ALTER TABLE Customer_Requirements
  ADD CONSTRAINT fk_customers_CustomerID
    FOREIGN KEY (CustomerID)
      REFERENCES Customers (CustomerID)
ALTER TABLE Customer_Requirements
  ADD CONSTRAINT fk_Customer_SkillID
    FOREIGN KEY (SkillID)
      REFERENCES Skills (SkillID)
```

```
ALTER TABLE Jobs
  ADD CONSTRAINT fk_Babysitter_Jobs
    FOREIGN KEY (BabysitterID)
      REFERENCES Babysitters (BabysitterID)
ALTER TABLE Jobs
  ADD CONSTRAINT fk_Job_CustomerID
    FOREIGN KEY (CustomerID)
      REFERENCES Customers (CustomerID)


ALTER TABLE Skills
ADD CONSTRAINT fk_Skill_RequirementID
    FOREIGN KEY (RequirementID)
      REFERENCES Customer_Requirement (RequirementID)
ALTER TABLE Skills
ADD CONSTRAINT fk_Skill_BabysitterID
    FOREIGN KEY (BabysitterID)
      REFERENCES Babysitters (BabysitterID)
```