**Design**

**1.Brand Name:**

**JUST RENT IT!**

**2.Business Process / Transactions**

The objective here is to develop a website where people who want to rent an apartment near the campus can easily get access to all the information they need online. In order to do it,we have to build a database where all the data is well aggregated and classified in the system by deciding related entities and its attributes and analyzing the relationship between them.

Following shows our business process:

1. Each company is described by a unique ID, company name, phone number, email address and website.
2. Each renter is described by a unique ID, first name, last name, phone number, email address and rentertype (employment status).
3. Each apartment is described by a unique apartment ID, name, phone number, renting website, apartment address, apartment open date, distance to campus and google review.
4. Each room is described by room number, room type, listed price and furnishing status.
5. Each employee is described by a unique ID, FirstName, last name, Phone number, email address and working position.
6. Each neighborhood is described by a unique ID, name, the number of health center, shopping center, entertainment place, dining place, transportation stops, employers and crime index.
7. An apartment can have many rooms, and a room must be in one and only one apartment.
8. One apartment can be owned by one or more companies, and one company can own zero or more apartments.
9. One employee can only manage one apartment but an apartment may be managed by one or multiple employees.
10. A neighborhood can have zero or many apartments but an apartment is only in one neighborhood.
11. For each room rented by the renter, there can be one or many companies responsible for signing the lease for that room.
12. For each room owned by the company, there can be zero or more renters who rent the rooms.
13. For each company and a renter, he or she can rent zero or more rooms.
14. Each lease has its own attributes: lease total price, deposit amount, sign date, lease start date, lease end date.
15. One renter can live with zero or more corenter.

**3. ER Schema**

**3.1 Entities, Attributes and Primary keys**

Apartment (**aptId**,aptName*,* aptPhoneNo, aptAddress, aptWebsite, aptOpenDate, aptCampsDistance, aptGoogleReview)

Company (**cpyId,** cpyName, cpyPhoneNo, cpyEmail, cpyWebsite)

Renter (**rntId**, rntName,-rntFirstName,-rntLastName, rntPhoneNo, rntEmail, rntType)

Employee (**empId,** empName,-empFirstName,-empLastName, empPhoneNo, empEmail, empPosition)

Room (**romId,** romType, romPrice, romStatus)

Neighborhood(**nghId,** nghName, nghHealthCenter, nghShopping, nghEntertainment, nghDinning, nghtransportation, nghEmployers, nghCrimeIndex)

**3.2 Relationships**

Corent: unary relationship

1 Renter to 0 or many Renters

Own: binary relationship

1 Apartment to 1 or more Company

1 Company to 0 or more Apartments

Manage: binary relationship

1 Apartment to 1 or more Employees

1 Employee to 1 Apartment

Has: binary relationship

1 Apartment to 1 or more Rooms

1 Room to 1 Apartment

Locate: binary relationship

1 Apartment to 1 Neighborhood

1 Neighborhood to 0 or more Apartments

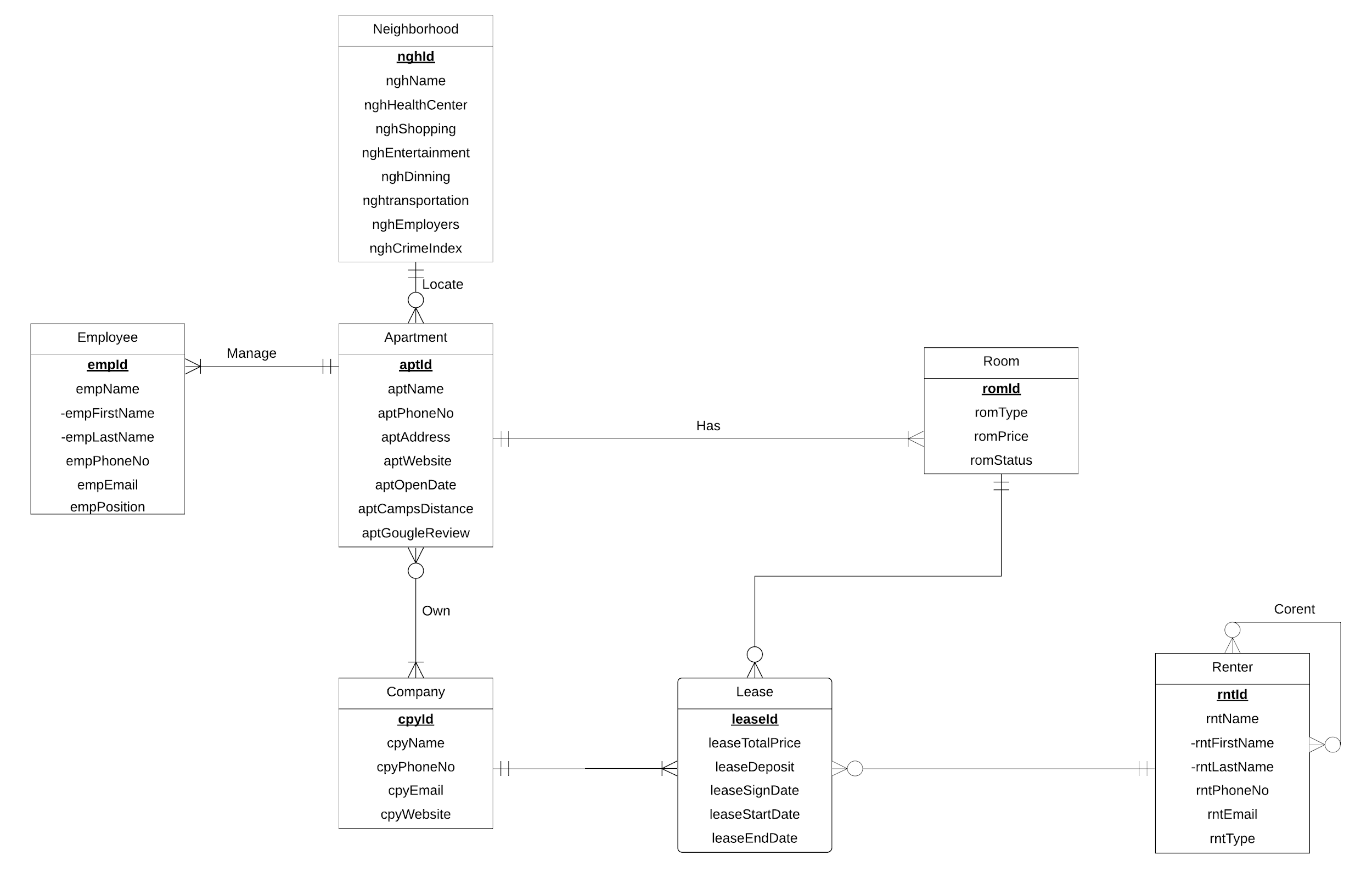
Lease: ternary relationship

1 Company and 1 Room to 0 or more Renters

1 Company and 1 Renter to 0 or more Rooms

1 Room and 1 Renter to 1 or many Companies

**3.3 ER Diagram**



**Proposal**

**1. Mission Statement**

In order to set up a website where all the information about the apartments close to campus are conveniently accessible to all students and workers, we have to build a database where all the data is well aggregated and classified in the system by deciding related entities and its attributes and analyzing the relationship between them.

Here are two main steps we take: Firstly, we display the entities and their relationship in the ER diagram and build accordingly relational model. Secondly, we create tables and insert the data by implementing SQL queries. Thirdly, we use Select queries to deal with some business transactions and also present the results. After this,we build a website and connect the website with tableau so that all the information is available online for all students and workers.

**2.Mission Objective**

The objective here is to develop a website where people who want to rent an apartment near the campus can easily get access to all the information they need online,which will save them large amounts of search time and provide them with better housing resources.

We give some details regarding:

[1] What is the neighborhood that has cheaper apartments?

[2] What are the prices of apartments according to their furnituring status?

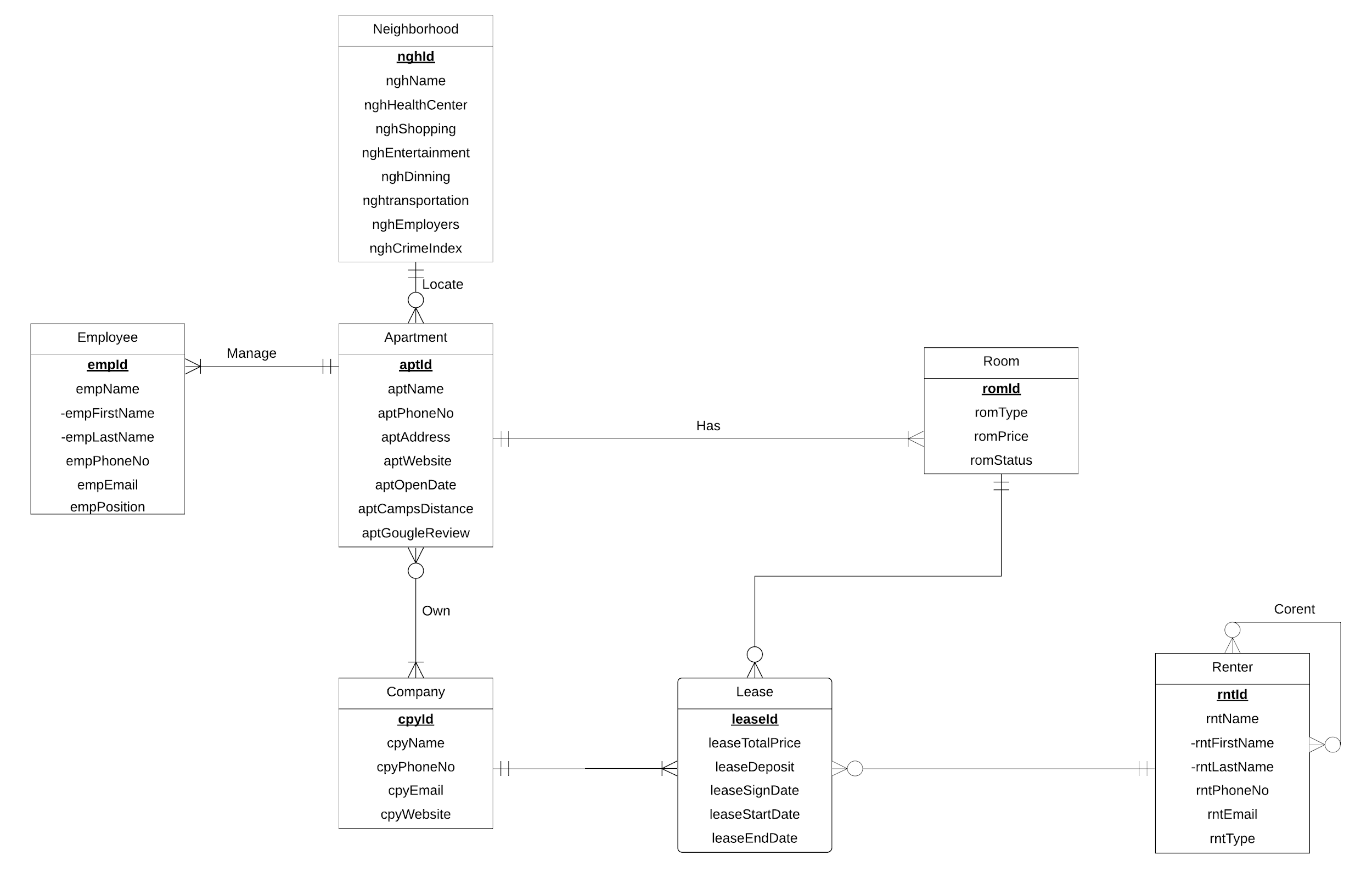
[3] What are the apartments that renters can have easy access to health centers and entertainment facilities?

[4] Which neighborhood is a better choice for renters who care much about the distance to school?

[5] Which are the apartments that have relatively high google reviews?

[6]Which neighborhood should a renter choose if he or she wants to live in a safer apartment?

**3.ER Diagram**



**4.Relations**

Apartment (**aptId**,*nghId*, aptName, aptPhoneNo, aptAddress, aptWebsite, aptOpenDate, aptCampsDistance, aptGoogleReview)

Renter (**rntId**, rntFirstName,rntLastName, rntPhoneNo, rntEmail, rntType)

Corent(***rntId,corntId***)

Company (**cpyId,** cpyName, cpyPhoneNo, cpyEmail, cpyWebsite)

Own(***aptId***,***cpyId***)

Employee (**empId,** *aptId,* empFirstName,empLastName, empPhoneNo, empEmail, empPosition)

Room (**romId,***aptId*, romType, romPrice, romStatus)

Lease (**leaseId,** *cpyId, romId, rntId*, leaseTotalPrice, leaseDeposit, leaseSignDate, leaseStartDate, leaseEndDate)

Neighborhood(**nghId,** nghName, nghHealthCenter, nghShopping, nghEntertainment, nghDinning, nghtransportation, nghEmployers, nghCrimeIndex)

**5.Functional Dependency**

aptId—> nghId, aptName, aptPhoneNo, aptAddress, aptWebsite, aptOpenDate, aptCampsDistance, aptGoogleReview

rntId—> rntFirstName,rntLastName, rntPhoneNo, rntEmail, rntType

rntId,corntId—>

cpyId—> cpyName, cpyPhoneNo, cpyEmail, cpyWebsite

aptId,cpyId—>

empId—> aptId, empFirstName,empLastName, empPhoneNo, empEmail, empPosition

romId—> aptId, romType, romPrice, romStatus

leaseId—> romId, cpyId, rntId, leaseTotalPrice, leaseDeposit, leaseSignDate, leaseStartDate, leaseEndDate

nghId—> nghName, nghHealthCenter, nghShopping, nghEntertainment, nghDinning, nghtransportation, nghEmployers, nghCrimeIndex

## 6.Normalization

Apartment (**aptId**, *nghId*, aptName, aptPhoneNo, aptAddress, aptWebsite, aptOpenDate, aptCampsDistance, aptGoogleReview)=3NF

Renter (**rntId**, rntFirstName,rntLastName, rntPhoneNo, rntEmail, rntType)=3NF

Corent(***rntId,corntId***)=3NF

Company (**cpyId,** cpyName, cpyPhoneNo, cpyEmail, cpyWebsite)=3NF

Own(***aptId***,***cpyId***)=3NF

Employee (**empId,** *aptId,* empFirstName,empLastName, empPhoneNo, empEmail, empPosition)=3NF

Room ( **romId,** *aptId*, romType, romPrice, romStatus)=3NF

Lease (**leaseId,** *cpyId, romId, rntId*, leaseTotalPrice, leaseDeposit, leaseSignDate, leaseStartDate, leaseEndDate)=3NF

Neighborhood(**nghId,** nghName, nghHealthCenter, nghShopping, nghEntertainment, nghDinning, nghtransportation, nghEmployers, nghCrimeIndex) =3NF

**7.Business Rules:**

[1]. When the information about the apartment is deleted or updated, the ownership information should be deleted or updated from the database.

[2]. When the information about the company is deleted or updated, the ownership should be deleted or updated from the database.

[3]. When the information about the apartment is deleted or updated, the employee information should be unchanged in the database.

[4]. When the information about the apartment is deleted or updated, the rooms should be deleted or updated in the database.

[5]. When the lease signed by the company and renter on a room expires, the company, the renter and the apartment information can not be changed or deleted from the database.

[6]. When the information about the neighborhood is deleted or updated, the location information should be deleted or updated from the database.

[7]. When the correnter information is updated or changed, the original renter information cannot be changed or updated.

**8.Referential integrity:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Relation** | **Foreign Key** | **Base**  **Relation** | **Primary Key** | **Business Rule** | **Constraint: ON DELETE** | **Business Rule** | **Constraint: ON UPDATE** |
| Own | aptId | Apartment | aptId | R1 | CASCADE | R1 | CASCADE |
| Own | cpyId | Company | cpyId | R2 | CASCADE | R2 | CASCADE |
| Corent | rntId | Renter | rntId | R7 | NO ACTION | R7 | NO ACTION |
| Corent | corntId | Renter | rntId | R7 | NO ACTION | R7 | NO ACTION |
| Employee | aptId | Apartment | aptId | R3 | NO ACTION | R3 | NO ACTION |
| Room | aptId | Apartment | aptId | R4 | CASCADE | R4 | CASCADE |
| Lease | cpyId | Apartment | aptId | R5 | NO ACTION | R5 | NO ACTION |
| Lease | romId | Room | romId | R5 | NO ACTION | R5 | NO ACTION |
| Lease | rntId | Renter | rntId | R5 | NO ACTION | R5 | NO ACTION |
| Apartment | nghId | Neighborhood | nghId | R6 | CASCADE | R6 | CASCADE |

**9.Sample Data:**

Apartment (**aptId**, *nghId*, aptName, aptPhoneNo, aptAddress, aptWebsite, aptOpenDate, aptCampsDistance, aptGoogleReview)

('APT000001','NGH000001', 'Alloy by Alta', ' 8446853293', '[www.alloy.com](http://www.alloy.com)', '4700 Berwyn House Rd, College Park, MD 20740', 'www.AlloybyAlta.com','')

Renter (**rntId**, rntFirstName,rntLastName, rntPhoneNo, rntEmail, rntType)

('00001', 'Jason', 'Lee', '234-567-8900', '[Jason@rhsmith.umd.edu](mailto:Jason@rhsmith.umd.edu)', 'student')

Corent(***rntId,corntId***)

('00001','00002')

Company (**cpyId,** cpyName, cpyPhoneNo, cpyEmail, cpyWebsite)

('00011','Apple','301-999-7999', 'Jame@rhsmith.umd.edu', 'www.apple.com')

Own(***aptId***,***cpyId***)

('01234', '00011')

Employee (**empId,** *aptId,* empFirstName,empLastName, empPhoneNo, empEmail, empPosition)

('00123','01234' ,'Amy','Cook', '240-314-1234', 'Amy@gmail.com', 'Manager')

Room ( **romId,** *aptId,* romType, romPrice, romStatus)

('23913', '01234', '2b2b', 890, 'unfurnished')

Lease (**leaseId,** *cpyId, romId, rntId*, leaseTotalPrice, leaseDeposit, leaseSignDate, leaseStartDate, leaseEndDate)

('28530', '00011', '23913', '00001', 890, 1780, '2019-09-13', '2019-09-20', '2020-09-20')

Neighborhood(**nghId,** nghName, nghHealthCenter, nghShopping, nghEntertainment, nghDinning, nghtransportation, nghEmployers, nghCrimeIndex)

('54321', 5, 4, 6, 8, 4,8,50)