## 

# Cuny Card Center

# Database Project

**Prepared By:**

Shereen Thomas

Jamie Bondie

Muqsit Momin

Jawad Chowdhurry

# Table Of Content

1. **Scenario**

1. [**System Analysis and ER Model**](#_heading=h.1t3h5sf)

* Problem Statement
* Assumption about the scenario
* Relationship Sentences
* Entity and Attributes
* E-R Model

1. [**Logical Modeling and Normalization**](#_heading=h.1y810tw)**.**

* Logical Modeling
* Normalization

1. Assumptions
2. Process

* Functional Dependencies
* Normalization Process

**IV.** [**Creating the Database Schema with Structured Query language**](#_heading=h.19c6y18)

* Creating tables using SQL code

1. Adding Primary key
2. Adding Foreign Key

* Database Schema
* Relationship View

# 

# Scenario

The following materials are the documentation of the design and development of a database application to support the Cuny Card Center. The project begins by identifying the entities and attributes followed by designing an E-R model. The E-R model proceeds through Logical (Relational) modeling and Normalization. Finally, the database schema is implemented using SQL codes. Assumptions are mentioned where necessary.

# System Analysis and ER Model

**Problem Statement:**

The Cuny Card center wants to distribute ID cards to all persons safely and effectively. A person can be a student, alumni, visitor, and/or employee. A person must provide a Government ID and a Photo along with their Designation to submit a request for a CUNY ID Card to be disbursed. In order to obtain their ID Card they also need to provide either a mailing address or the physical address of a CUNY Card center located throughout campus.

**Assumptions about the scenario:**

* Distribution method could be mail or in-person
* Cuny Card Center has multiple locations (Addresses)
* Student or Employee must Request an ID with their Designation (Student, Employee, Alumni, etc.)
* A Person must have the appropriate Role to perform actions

**Relationship Sentences:**

* There are multiple CUNYCard Centers.
* A Person may issue an ID Card to a Persons’ mailing Address or physically at the Address of a CUNYCard Center
* A CUNYCard Center can only have one Address; one for each location.
* A Person can have a mailing address and/or a physical address
* An ID Card indicates the Person’ Designation at CUNY, so an employee that is also a student could request multiple ID Cards, one for each Designation
* A Designation indicates a Persons’ affiliation with CUNY, i.e. ‘Student’, ‘Alumni’, ‘Employee’, etc.
* A Person Requests ID Cards, they may make multiple Requests and be associated with several ID Cards.
* A Request must be approved/verified to issue an ID with a Persons’ Designation, Since this is a database that is for issuing ID Cards, not maintaining Student and Employee Information; In this case, we are assuming that we need to have these Designations validated by a Person (perhaps they are manually checking another system or in the future Person could be leveraged for an integration, i.e. Person.Name = ‘PeopleSoft’)
* A Person approves/verifies the Designation on a Request (perhaps they are manually checking another system or in the future Person could be leveraged for an integration)
* A Person distributes an ID Card at the physical Address for a Cuny Card Center or to a Person's mailing address, i.e. picked up in person at Cuny Card Center or mailed.
* Any Person may make Requests, only some Person’s have System\_Role’s that allow them to be Approver’s and/or Disbursers.
* One Government ID is needed to process a Request.
* One Photo is needed to process a Request; One Photo is used for a CUNY Card ID
* A Person can have multiple System Roles or none; a System Role can be assigned to multiple Persons
* A System Role has only one Designation; A Designation can have multiple System Role’s.
* The System Role entity is a unique identifier for system permissions as it relates to Person and Designation, i.e. Approval and Disbursement actions

**Entity and Attributes**

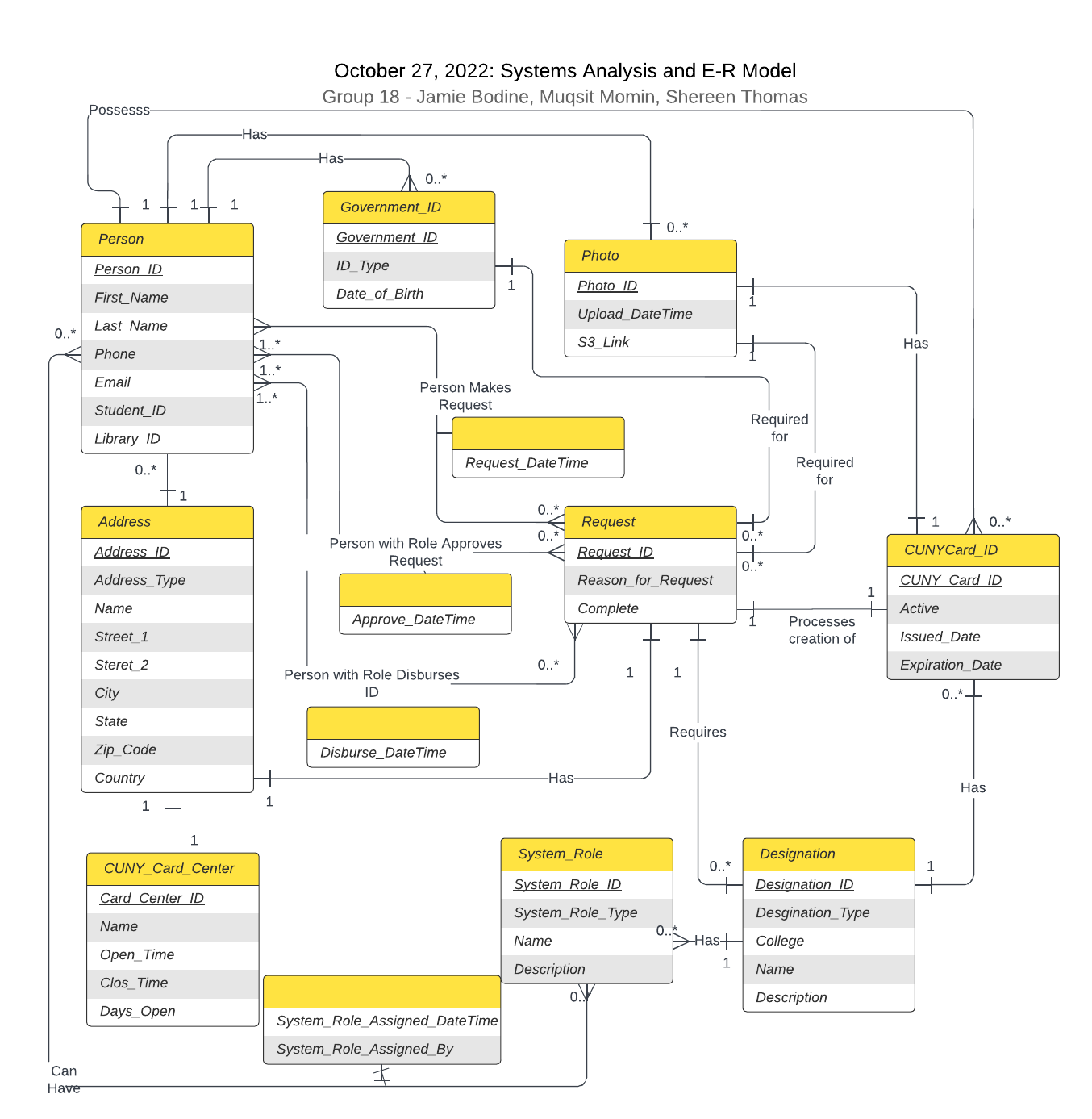
Our First Iteration

| **Entities** | **Attributes** |
| --- | --- |
| CUNYCard Center | Card\_Center\_ID, Name, OpenTime, CloseTime, DaysOpen |
| CUNY Card ID | CUNY\_Card\_ID, Active, IssuedDate, ExpirationDate |
| Issuer | Person\_ID, First\_Name, Last\_Name, Phone, Email |
| Requestor | RequestorNum, First\_Name, Last\_Name, Designation\_Title |
| Address | Address\_ID, Type, Name, Street\_1, Street\_2, City, State, Zip\_Code |
| Request | Request\_Num, RequestDateTime, Reason\_for\_Request,Complete |
| Designation | Designation\_ID, Title, Name, Description |
| Government ID | Govenment\_ID, ID\_Type, Date\_of\_Birth |
| Photo | Photo\_ID, Upload\_DateTime, S3\_Link |

Final Iteration

| **Entities** | **Attributes** |
| --- | --- |
| CUNYCard Center | Card\_Center\_ID, Name, OpenTime, CloseTime, DaysOpen |
| CUNY Card ID | CUNY\_Card\_ID, Active, IssuedDate, ExpirationDate |
| Person | Person\_ID, First\_Name, Last\_Name, Phone, Email, Student\_ID, Library\_ID |
| Address | Address\_ID, Address\_Type, Name, Street\_1, Street\_2, City, State, Zip\_Code, Country |
| Request | Request\_ID, Reason\_for\_Request,Complete, Request\_DateTime, Approve\_DateTime, Disburse\_DateTime |
| Designation | Designation\_ID, Designation\_Type, Name, College, Description |
| Government\_ID | Govenment\_ID, ID\_Type, Date\_of\_Birth |
| System\_Role | System\_Role\_ID, Sytem\_Role\_Type, Name, Description, Assigned\_DateTime, Assigned\_By |
| Photo | Photo\_ID, Upload\_DateTime, S3\_Link, Person\_ID (fk) |

**E-R Model**

****

# Logical Modeling and Normalization

# Logical Modeling

| Relation | Attributes |
| --- | --- |
| CUNY\_Card\_Center | Card\_Center\_ID, Name, Open\_Time, Close\_Time, Days\_Open, Address\_ID (fk) |
| CUNYCard\_ID | CUNY\_Card\_ID, Active, Issued\_Date, Expiration\_Date, Person\_ID (fk), Designation\_ID (fk), Photo\_ID (fk) |
| Person | Person\_ID, First\_Name, Last\_Name, Phone, Email, Student\_ID, Library\_ID, Address\_ID (fk) |
| Person\_Request | Person\_ID (fk), Request\_ID (fk), Request\_DateTime |
| Person\_Approve | Person\_ID (fk), Request\_ID (fk), Approve\_DateTime |
| Person\_Disburse | Person\_ID (fk), Request\_ID (fk), Disburse\_DateTime |
| Address | Address\_ID, Address\_Type, Name, Street\_1, Street\_2, City, State, Zip\_Code, Country |
| Request | Request\_ID, Reason\_for\_Request,Complete, Designation\_ID (fk), Government\_ID (fk), Photo\_ID (fk), CUNY\_Card\_ID (fk), Address\_ID (fk) |
| Designation | Designation\_ID, Designation\_Type, Name, College, Description |
| Government\_ID | Govenment\_ID, ID\_Type, Date\_of\_Birth, Person\_ID (fk) |
| System\_Role | System\_Role\_ID, Sytem\_Role\_Type, Name, Description, Designation\_ID (fk) |
| Person\_System\_Role | Person\_ID (fk), System\_Role\_ID (fk), Assigned\_DateTime, Assigned\_By |
| Photo | Photo\_ID, Upload\_DateTime, S3\_Link, Person\_ID (fk) |

**Normalization**

**Assumptions about normalization exercise:**

* All tables not included were considered normalized after our first iteration
* Intended to demonstrate how we worked through this process
* Conceptualization necessitated expansion of attributes during this process

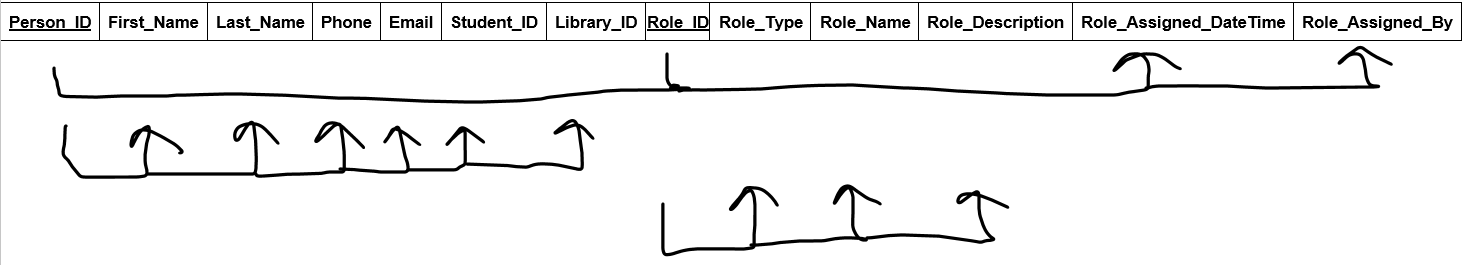
**Process:**

1. Our Person entity started here…

| Relation | Attributes |
| --- | --- |
| Person | Person\_ID, First\_Name, Last\_Name, Phone, Email, Approver (T/F), Disburser (T/F) |

1. We realized our attributes needed to be conceptualized differently…

| Relation | Attributes |
| --- | --- |
| Person | Person\_ID, First\_Name, Last\_Name, Phone, Email, Student\_ID, Library\_ID, System\_Role\_ID, System\_Role\_Type, System\_Role\_Name, System\_Role\_Description, System\_Role\_Assigned\_DateTime, System\_Role\_Assigned\_By |

1. Then we normalized… 

\*we also updated our attribute names a little after we normalized, changes not account for in above

**Functional dependencies**

**Listing all the functional dependencies of the table. State any assumptions you make about the data shown in this table (if any).**

Keys: Person\_ID, System\_Role\_ID

FD1: Person\_ID, System\_Role\_ID→ System\_Role\_Assigned\_DateTime, System\_Role\_Assigned\_By

FD2: Person\_ID→ First\_Name, Last\_Name, Phone, Email, Student\_ID, Library\_ID

FD3: System\_Role\_ID→ System\_Role\_Type, System\_Role\_Name, System\_Role\_Description

**Normalization Process**

**1NF:** PersonRole(Person\_ID, First\_Name, Last\_Name, Phone, Email, Student\_ID, Library\_ID,

System\_Role\_ID, System\_Role\_Type, System\_Role\_Name, System\_Role\_Description, System\_Role\_Assigned\_DateTime, System\_Role\_Assigned\_By)

\*In the 1NF repeating groups are removed. Functional dependencies are then identified.

**2NF**: Person(Person\_ID, First\_Name, Last\_Name, Phone, Email, Student\_ID, Library\_ID)

Role(System\_Role\_ID, System\_Role\_Type, System\_Role\_Name, System\_Role\_Description, System\_Role\_Assigned\_DateTime, System\_Role\_Assigned\_By)

\*In the 2NF partial dependent attributes are removed from the relation and put in a new relationship with a copy of their determinant.

**3NF**: Person(Person\_ID, First\_Name, Last\_Name, Phone, Email, Student\_ID, Library\_ID)

Role(System\_Role\_ID, System\_Role\_Type, System\_Role\_Name, System\_Role\_Description)

Person\_System\_Role(Person\_ID (FK), System\_Role\_ID (FK), System\_Role\_Assigned\_DateTime, System\_Role\_Assigned\_By)

\*In the 3NF dependent attributes are removed from the relation and put in a new relationship with a copy of their determinant

1. And landed on this…

| Relation | Attributes |
| --- | --- |
| Person | Person\_ID, First\_Name, Last\_Name, Phone, Email, Student\_ID, Library\_ID, Address\_ID (FK) |
| Role | System\_Role\_ID, System\_Role\_Type, Name, Description |
| Person\_Role | Person\_ID (FK), System\_Role\_ID (FK), System\_Role\_Assigned\_DateTime, System\_Role\_Assigned\_By |

# Creating the Database Schema with Structured Query language

### **The following SQL code creates the tables and adds the PRIMARY KEY constraint to each one:**

CREATE TABLE CUNY\_Card\_Center (

Card\_Center\_ID INT NOT NULL,

Name VARCHAR (45) NULL,

Open\_Time VARCHAR (45) NULL,  
Close\_Time VARCHAR (45) NULL,

Days\_Open VARCHAR (45) NULL,

PRIMARY KEY (Card\_Center\_ID) );

CREATE TABLE CUNYCard\_ID (

CUNY\_Card\_ID INT NOT NULL,

Active VARCHAR (10) NULL,

Issued\_Date DATE NULL,

Expiration\_Date DATE NULL,

PRIMARY KEY (CUNY\_Card\_ID) );

CREATE TABLE Person(

Person\_ID INT NOT NULL,

First\_Name VARCHAR (45) NULL,

Last\_Name VARCHAR (45) NULL,

Phone INT NULL,

Email VARCHAR(100) NULL,

Student\_ID VARCHAR (10) NULL,

Library\_ID VARCHAR (10) NULL,

PRIMARY KEY (Person\_ID) );

CREATE TABLE Person\_Request (

Request\_DateTime DATETIME NULL );

CREATE TABLE Person\_Approve (

Request\_DateTime DATETIME NULL );

CREATE TABLE Person\_Disburse(

Request\_DateTime DATETIME NULL) ;

CREATE TABLE Address (

Address\_ID INT NOT NULL,

Address\_Type VARCHAR (20) NULL,

Name VARCHAR (45) NULL,

Street\_1 VARCHAR (45) NULL,

Street\_2 VARCHAR (45) NULL,

City VARCHAR (45) NULL,

State VARCHAR (10) NULL,

Zip\_Code INT NULL,

Country VARCHAR (45) NULL,

PRIMARY KEY (Address\_ID) );

CREATE TABLE Request (

Request\_ID INT NOT NULL,

Reason\_for\_Request VARCHAR(150) NULL,

Complete BIT NULL,

PRIMARY KEY (Request\_ID) );

CREATE TABLE Designation (

Designation\_ID INT NOT NULL,

Designation\_Type VARCHAR(25) NULL,

Name VARCHAR(40) NULL,

College VARCHAR(50) NULL,

Description VARCHAR(100) NULL,

PRIMARY KEY (Designation\_ID));

CREATE TABLE Government\_ID (

Govenment\_ID INT NOT NULL,

ID\_Type INT NOT NULL,

Date\_of\_Birth DATE,

PRIMARY KEY (Government\_ID) );

CREATE TABLE System\_Role

System\_Role\_ID INT NOT NULL,

Type INT NOT NULL,

Name VARCHAR(30) NULL,

Description VARCHAR(100) NULL,

PRIMARY KEY (System\_Role\_ID));

CREATE TABLE Person\_System\_Role (

Assigned\_DateTime DATETIME NOT NULL,

Assigned\_By INT NOT NULL );

CREATE TABLE Photo (

Photo\_ID INT NOT NULL,

Upload\_DateTime DATETIME NOT NULL,

S3\_Link VARCHAR(100) NOT NULL,

PRIMARY KEY (Photo\_ID) );

**Adding Foreign Keys:**

The following SQL codes add FOREIGN KEY constraints to link the tables together

**We followed the example provided (**[**holowczak.com/the-hair-salon-database-project/5/**](https://holowczak.com/the-hair-salon-database-project/5/)**), but could not make it work in Access to implement this database. Below is that work (at the end is an example of what was used to implement this in Access)**

Alter TABLE CUNY\_Card\_Center

ADD CONSTRAINT fk\_CUNY\_Card\_Center\_Address

FOREIGN KEY (Address\_ID)

REFERENCES Address(Address\_ID)

Alter TABLE CUNYCard\_ID

ADD CONSTRAINT fk\_CUNYCard\_Person

FOREIGN KEY (Person\_ID)

REFERENCES Person(Person\_ID)

Alter TABLE CUNYCard\_ID

ADD CONSTRAINT fk\_CUNYCard\_Designation

FOREIGN KEY (Designation\_ID)

REFERENCES Designation(Designation\_ID)

Alter TABLE CUNYCard\_ID

ADD CONSTRAINT fk\_CUNYCard\_Photo

FOREIGN KEY (Photo\_ID)

REFERENCES Photo(Photo\_ID)

Alter TABLE Person\_Request

ADD CONSTRAINT fk\_Person\_Request\_Person

FOREIGN KEY (Person\_ID)

REFERENCES Person(Person\_ID)

Alter TABLE Person\_Request

ADD CONSTRAINT fk\_Person\_Request\_Request

FOREIGN KEY (Request\_ID)

REFERENCES Request(Request\_ID)

Alter TABLE Person\_Approve

ADD CONSTRAINT fk\_Person\_Request\_Person

FOREIGN KEY (Person\_ID)

REFERENCES Person(Person\_ID)

Alter TABLE Person\_Approve

ADD CONSTRAINT fk\_Person\_Request\_Request

FOREIGN KEY (Request\_ID)

REFERENCES Request(Request\_ID)

Alter TABLE Person\_Disburse

ADD CONSTRAINT fk\_Person\_Request\_Person

FOREIGN KEY (Person\_ID)

REFERENCES Person(Person\_ID)

Alter TABLE Person\_Disburse

ADD CONSTRAINT fk\_Person\_Request\_Request

FOREIGN KEY (Request\_ID)

REFERENCES Request(Request\_ID)

Alter TABLE Request

ADD CONSTRAINT fk\_Request\_Designation

FOREIGN KEY (Designation\_ID)

REFERENCES Designation (Designation\_ID)

Alter TABLE Request

ADD CONSTRAINT fk\_Request\_Government\_ID

FOREIGN KEY (Government\_ID)

REFERENCES Government\_ID(Government\_ID)

Alter TABLE Request

ADD CONSTRAINT fk\_Request\_Photo

FOREIGN KEY (Photo\_ID)

REFERENCES Photo (Photo\_ID)

Alter TABLE Request

ADD CONSTRAINT fk\_Request\_CUNYCard\_ID

FOREIGN KEY (CUNY\_Card\_ID)

REFERENCES CUNYCard\_ID(CUNY\_Card\_ID)

Alter TABLE Request

ADD CONSTRAINT fk\_Request\_Address

FOREIGN KEY (Address\_ID)

REFERENCES Address(Address\_ID)

Alter TABLE Government\_ID

ADD CONSTRAINT fk\_Government\_ID\_Person

FOREIGN KEY (Person\_ID)

REFERENCES Person(Person\_ID)

Alter TABLE System\_Role

ADD CONSTRAINT fk\_System\_Role\_Designation

FOREIGN KEY (Designation\_ID)

REFERENCES Designation(Designation\_ID)

Alter TABLE Person\_System\_Role

ADD CONSTRAINT fk\_ Person\_System\_Role\_Person

FOREIGN KEY (Person\_ID)

REFERENCES Person(Person\_ID)

Alter TABLE Person\_System\_Role

ADD CONSTRAINT fk\_System\_Role

FOREIGN KEY (System\_Role\_ID)

REFERENCES System\_Role(System\_Role\_ID)

Alter TABLE Photo

ADD CONSTRAINT fk\_Photo\_Person

FOREIGN KEY (Person\_ID)

REFERENCES Person(Person\_ID)

**Example of what was used to implement this in Access, this is just one snippet, but we had to create the column for the foreign key… and then create the relationship, this is also the reason we implemented the database in this sequence Tables-> Columns-> Relationship… each had to exist to do the next step.**

ALTER TABLE System\_Role

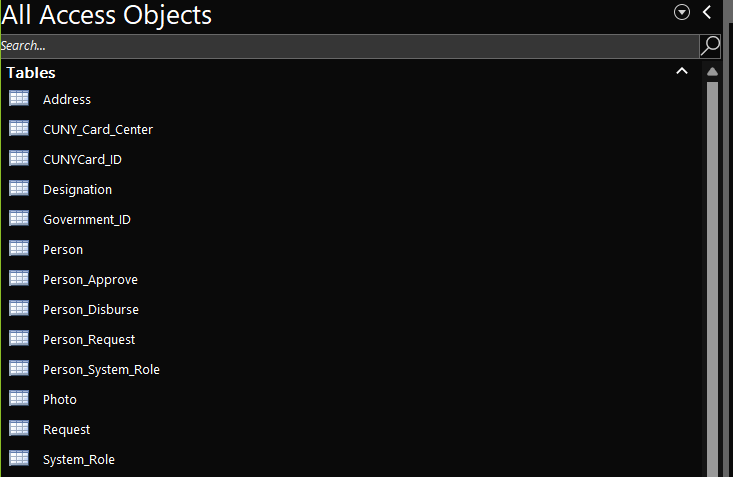
ADD COLUMN Designation\_ID INT;

ALTER TABLE System\_Role

ADD FOREIGN KEY (Designation\_ID) REFERENCES Designation;

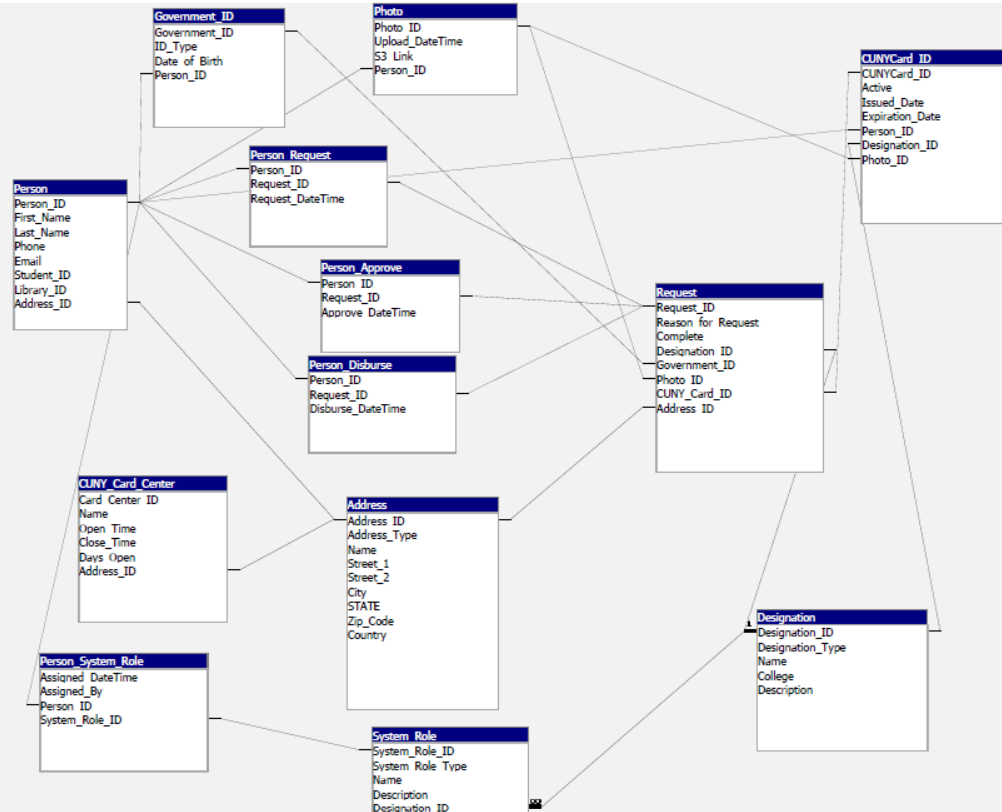
**Database Schema:**

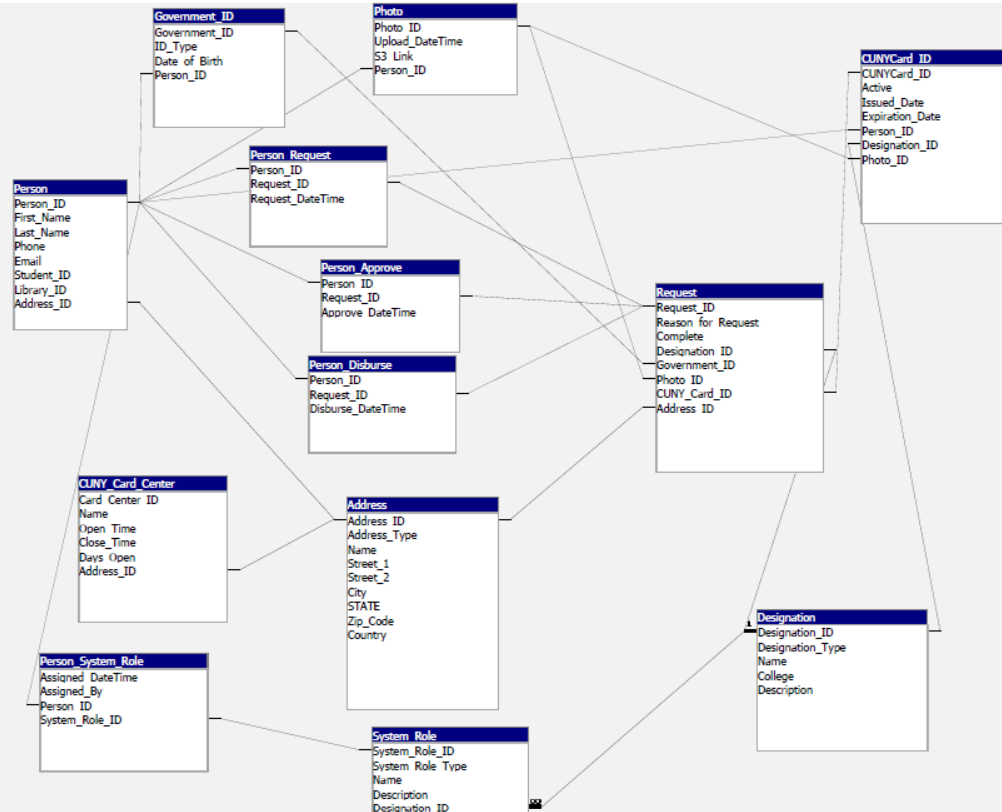
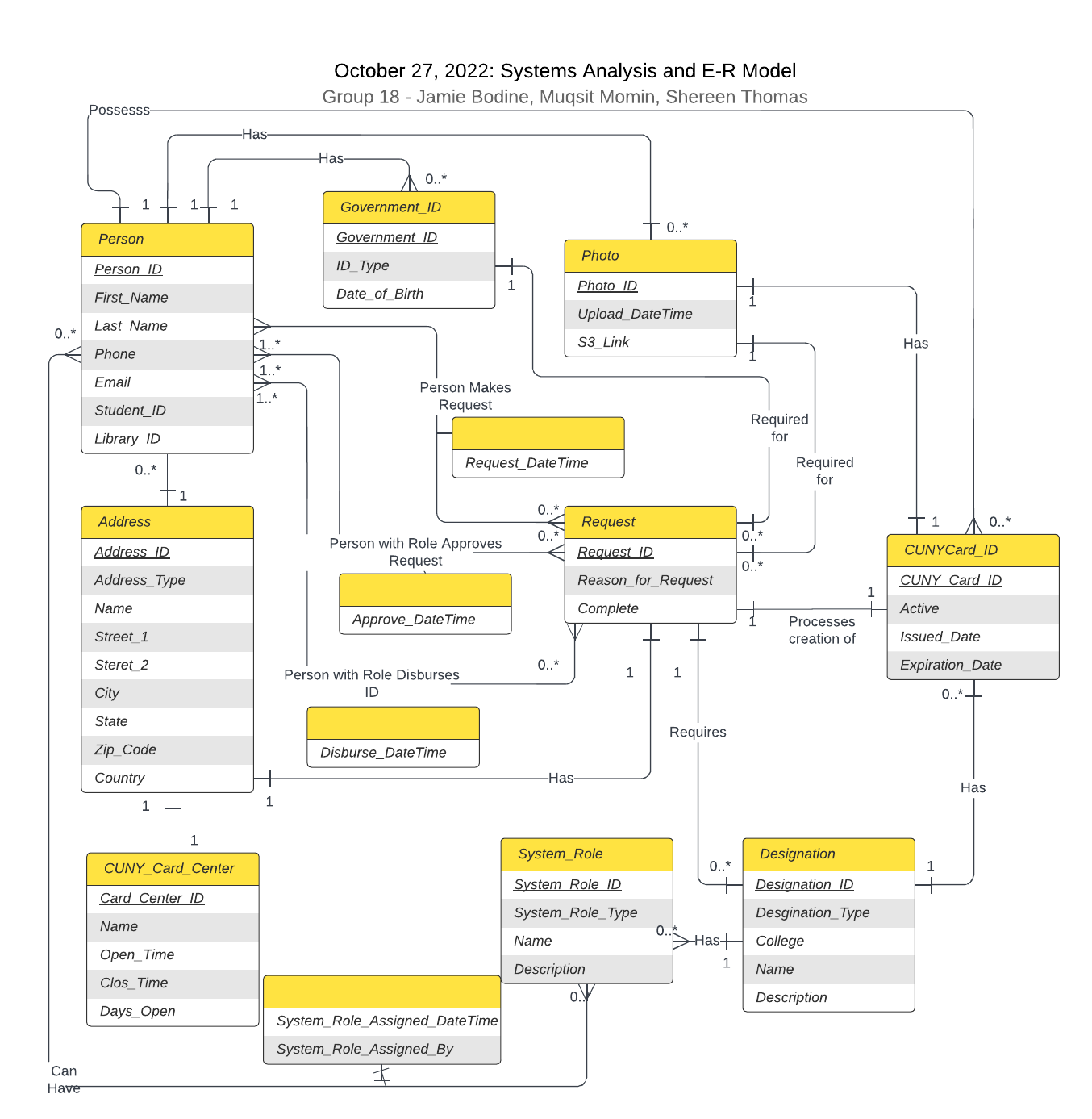
**After creating the table and adding the foreign key constraints, the database schema now looks like the following:**



**Relationship View:**

Using the Relationship View under Database Tools, we can see the relationships (foreign keys) between the tables:

****

****