LIBRARY DATABASE APPLICATION

DESCRIPTION:

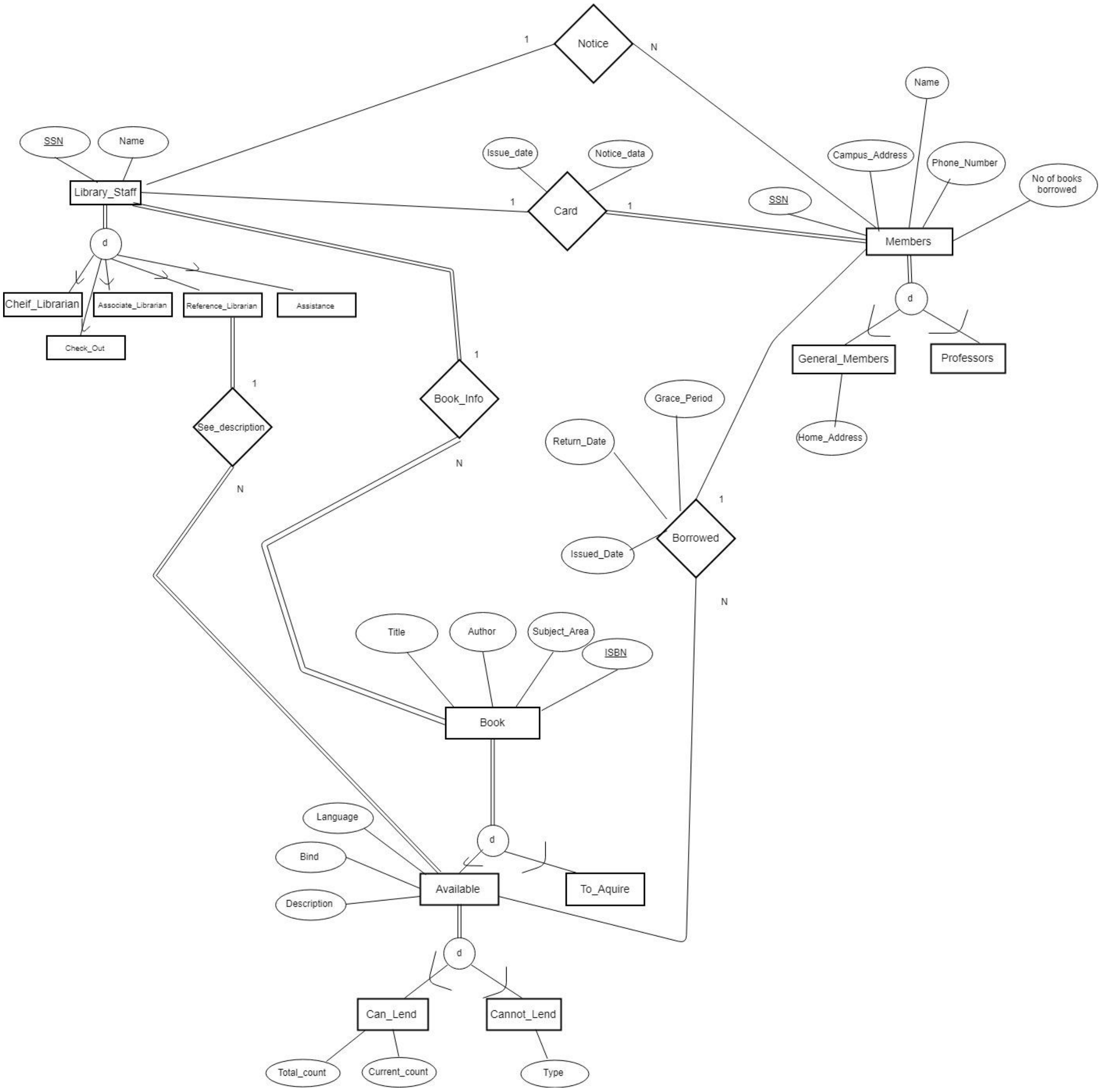
This project gives a detail description of design and implementation of library management system.

- STEP1: A detail EER is drawn from the given description for a Library management system. And with the assumptions that are made.
- STEP2: The projected EER is converted into relational database schema and implemented in SQL.
- STEP3 : Sample data has been created for created schema. And Random Queries were run for testing.

INTRODUCTION:

This is a general Library management system. Which gives a deep description about how a total library management system takes place.

EER DIAGRAM:



1.1 Entities:

a. BOOKS

- Each book a unique Id called ISBN.
- Each book has an author name, Subject Area, Title
- The book is available or not is also noted.
- There is a wanted attribute which holds the books that the librarian wants to acquire.

b. LIB_STAFF

- A unique id SSN for each staff.
- And every member holds a Given name.

c. MEMBERS

- An unique ID SSN given for each member
- Campus Address for each member
- Phone number od each member
- Number of books borrowed by each member
- Card Expiry date of each member

1.2 Relations:

A. **NOTICE**

- A librarian sends notice to all those who are due to return their books and about the expiry of their identity card
- This is a 1: N relation as each member can get many notices depending upon the number of books they borrowed.

B. **CARD**

- A card is issued to all those who are the member of the library.
- A professor is already registered as he is registered to the university.
- This is a 1: 1 relation as each member will only have one card and it will be issued by one staff member

C. **BOOK INFO**

- The book information such as the number of books is available, how many books are out of stock, etc. are only accessed by librarians
- So they are given to a particular type of library member.
- This is 1: N relation as each staff member can get the details of N number of books

D. **BORROWED**

- Books are borrowed by the members.
- Professors can also borrow books with a different due date.
- The borrow relation has relation attributes of return date, grace period and the expected return date of the book.
- This is a M: N relation. Since any number of members can borrow any number of books.

E. GET DESCRIPTION

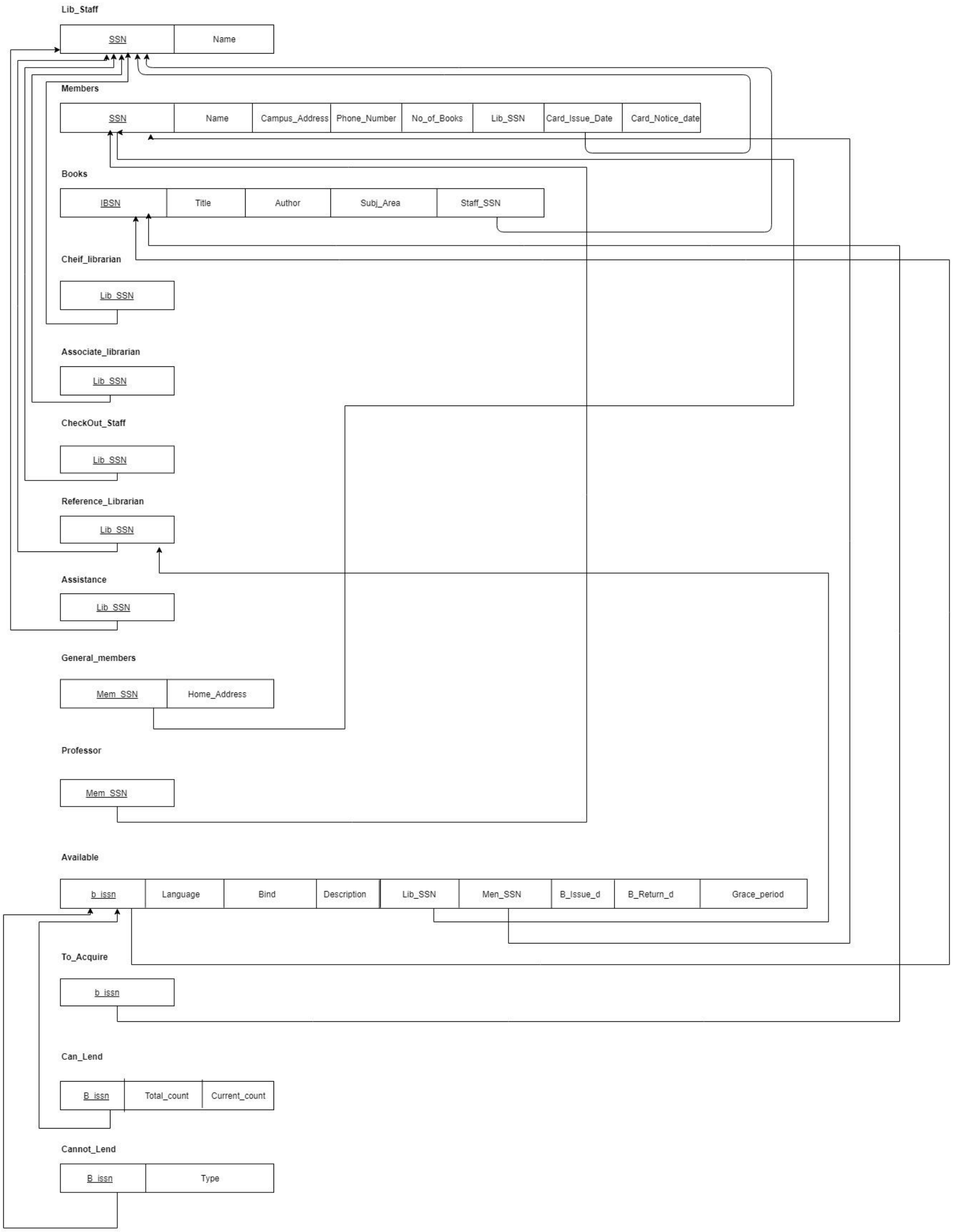
- Since the description can only be viewed only by reference library staff it is given a separate relation to a book description.
- This is given a 1: N relation as one reference librarian can access multiple book descriptions.

1.3 Assumptions:

1. Each Library staff can only have one designation.
2. There is only one person per designation in Library staff.
3. Only the Members that are not professors will have the home address and the professors will only have campus address.
4. We have the ISBN number for the books that the librarians want to acquire.

RELATIONAL SCHEMA:

The following is the converted schema from the above EER diagram



Lib_Staff

SSN	Name

Members

SSN	Name	Campus_Address	Phone_Number	No_of_Books	Lib_SSN	Card_Issue_Date	Card_Notice_date

Books

IBSN	Title	Author	Subj_Area	Staff_SSN

Cheif_librarian

Lib_SSN

Associate_librarian

Lib_SSN

CheckOut_Staff

Lib_SSN

Reference_Librarian

Lib_SSN

Assistance

Lib_SSN

General_members

Mem_SSN	Home_Address

Professor

Mem_SSN

Available

b_issn	Language	Bind	Description	Lib_SSN	Men_SSN	B_Issue_d	B_Return_d	Grace_period

To_Acquire

b_issn

Can_Lend

B_issn	Total_count	Current_count

Cannot_Lend

B_issn	Type

CREATE TABLE STATEMENTS in MYSQL:

```
Create TABLE Library_Staff (SSN int NOT NULL, Name varchar(255), PRIMARY
KEY(SSN));

Create TABLE Members (SSN int NOT NULL, Campus_Address varchar(255), Name
varchar(255), Phone_No CHAR(10), No_of_Books int, Lib_SSN int, Card_Issue_date
date, Card_Notice_date date, PRIMARY KEY(SSN), FOREIGN KEY (Lib_SSN)
REFERENCES library_staff(SSN));

Create TABLE books (ISBN int NOT NULL,
 Title varchar(50),
 Author varchar(30),
 Subject_Area varchar(30),
 Staff_SSN int,
 PRIMARY KEY(ISBN),
 FOREIGN KEY (Staff_SSN) REFERENCES library_staff(SSN));

 Create TABLE cheif_librarian (lib_SSN int, FOREIGN KEY(lib_SSN) REFERENCES
library_staff(SSN));

 Create TABLE Associate_librarian (lib_SSN int, FOREIGN KEY(lib_SSN)
REFERENCES library_staff(SSN));

 Create TABLE check_out_staff (lib_SSN int, FOREIGN KEY(lib_SSN) REFERENCES
library_staff(SSN));

 Create TABLE ref_librarian (lib_SSN int, FOREIGN KEY(lib_SSN) REFERENCES
library_staff(SSN));

 Create TABLE Assitance (lib_SSN int, FOREIGN KEY(lib_SSN) REFERENCES
library_staff(SSN));

 Create TABLE general_members (Mem_SSN int,Home_Address varchar(255),
FOREIGN KEY(Mem_SSN) REFERENCES members(SSN));

 Create TABLE professor (Mem_SSN int, FOREIGN KEY(Mem_SSN) REFERENCES
members(SSN));

 Create TABLE available (b_isbn int NOT NULL,lang varchar(255),bind varchar(15),
descri text, lib_SSN int, mem_SSN int, book_issue_date date, b_return_date date,
grace_period int, FOREIGN KEY(mem_SSN) REFERENCES
members(SSN),FOREIGN KEY(lib_SSN) REFERENCES ref_librarian(lib_SSN) );
```

```
 Create TABLE To_acquire (b_ISBN int, FOREIGN KEY(b_ISBN) REFERENCES
books(ISBN));

 ALTER TABLE available ADD FOREIGN KEY (b_ISBN) REFERENCES books(ISBN);

Create TABLE can_lend (b_ISBN int NOT NULL,total_count int, current_cout int,
FOREIGN KEY(b_ISBN) REFERENCES available(b_isbn));

Create Table cannot_lend(b_isbn int NOT NULL, type varchar(30), FOREIGN
KEY(b_isbn) REFERENCES available(b_isbn));
```