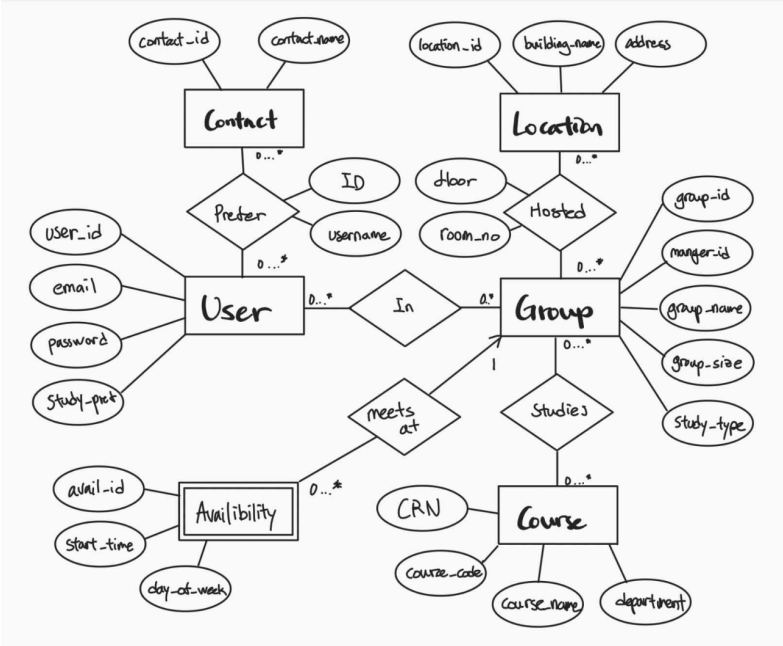
1+3. ER Diagram:



2. Entities and Relationships

Ideology:

All of the entities that we have are independent categories that are not dependent on other entities. Hence we use many-to-many relationships to describe how they relate to each other.

Entities:

1) User: Stores user profile data
2) Contact: methods of contact (social media, messaging services) the user uses, which helps other user contact the person to be added to the study group
3) Group: Formed study groups
4) Course: Stores university course data that users can filter by when creating or joining
5) Location: places to meet at for studying

Weak Entities

1) Availability: Stores availability of users to schedule group meetings.

Relationships/Cardinality:

- User and Contact: One user could have several forms of contacts, but one method of contact could have several users using it (Many-Many).
- User and Group: One User could be in multiple (study) Groups, but one Group could have many users as members (Many-Many).
- Group and Location: One group could meet up at multiple locations on campus, but one location could have several groups meeting there (Many-Many).
- Group and Course: One group could be studying for several courses, but one course could have several groups that study for it (Many-Many).
- Group and Availability: One group has a list of available times to meet at (One-Many).

4. Normalization (3NF):

Reference: U=User, Con=Contact, G=Group, Cou=Course, L=Location, A=Availability

Given: R(U, Con, G, Cou, L, A)

- U->G
- G->A

LEFT	MIDDLE	RIGHT	NONE
U	G	A	Con, Cou, L

1) Find minimal superkey

UConCouL+ = UConCouLGA

2) Compute set of minimal bases for FDs

2.1) RHS is Singleton (Yes)

2.2) Remove unnecessary attributes from LHS

U->G: G+=A. No removal.

G->A: A+=A. No removal.

2.3) Remove redundant FDs (using remaining other FDs)

U->G: U+=U. No removal.

G->A: G+=G. No removal.

3) Create a relation for every minimal basis FD

X(U,G); Y(G,A)

BUT it is missing Con, Cou, and L!

4) Add any of the candidate keys as a relation if we don't have a candidate key as a subset of any of the resulted relations in 3)

Add in M(U, Con), N(G, Cou), O(G, L)

So the minimal basis is X(U,G); Y(G,A), M(U, Con), N(G, Cou), O(G, L), which our ER diagram adheres to.

5. Relational Schema:

```
User (
    user_id INT PK,
    email VARCHAR,
    password CHAR(100),
    study_pref CHAR(100)
)

Group (
    group_id INT PK,
    manager_id INT [FK to User.user_id],
    group_name VARCHAR,
    group_size INT,
    study_type CHAR(100),
)

Membership (
    user_id INT PK [FK to User.user_id],
    group_id INT PK [FK to Group.group_id],
    role VARCHAR
)

Availability (
    avail_id INT PK,
    start_time TIME,
    day_of_week DATE,
    group_id INT [FK to Group.group_id]
)

Course (
    CRN INT PK,
    course_code CHAR(7),
    course_name VARCHAR,
```

```
    instructor VARCHAR,
    department VARCHAR,
)


Group_Courses (
    group_id INT [FK to Group.group_id],
    course_id INT [FK to Courses.course_id],
)

Location (
    location_id INT PK,
    building_name VARCHAR,
    address VARCHAR,
)

Group_Location (
    group_id INT [FK to Group.group_id],
    location_id INT [FK to location_id],
    floor INT,
    room_no INT,
)

Contact (
    contact_id INT PK,
    contact_name VARCHAR,
)

User_Contact (
    user_id INT [FK to User.user_id],
    contact_id INT [FK to Contact.contact_id],
    ID VARCHAR,
```

```
    user_name VARCHAR,
)
```