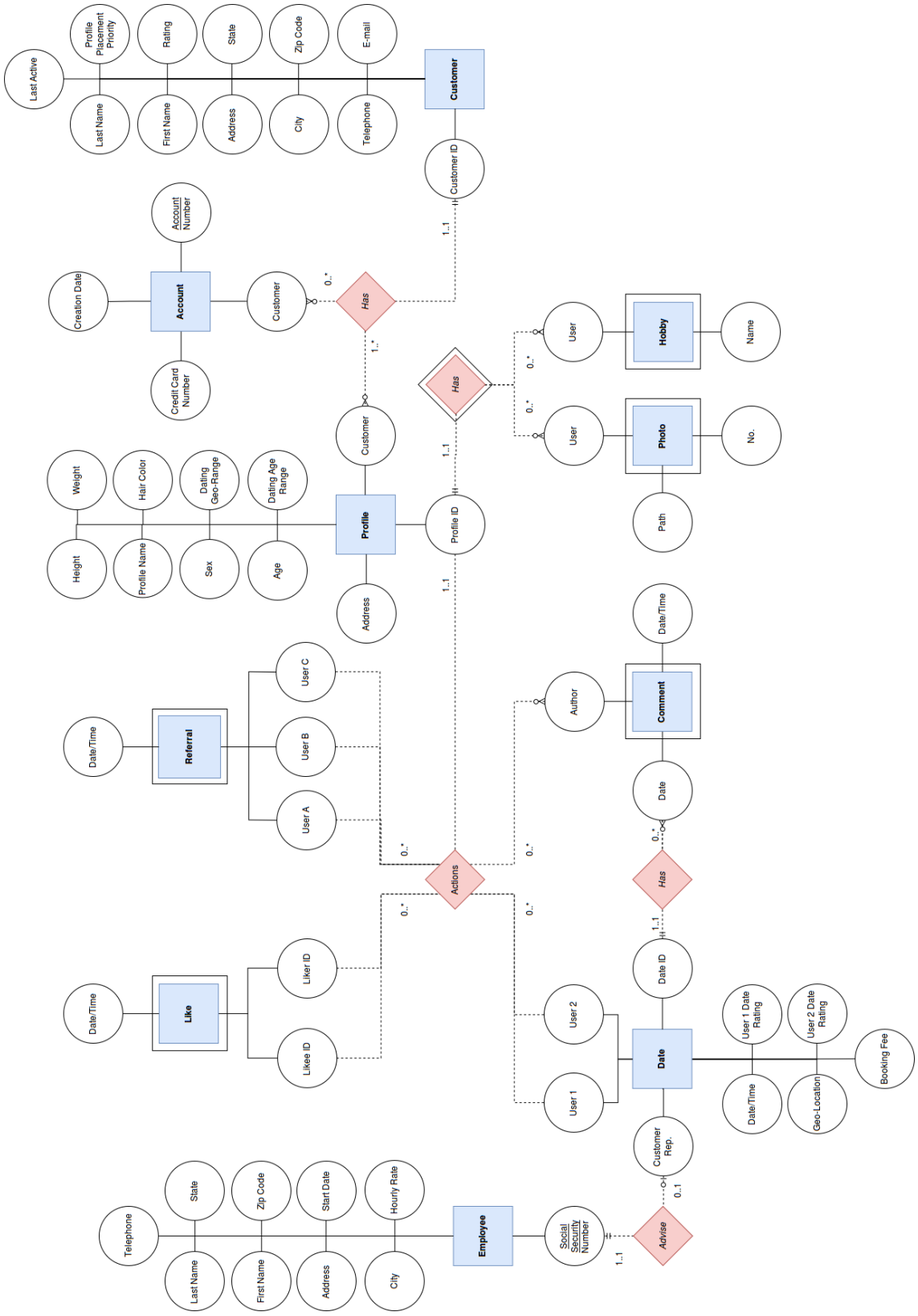
Jon Reyes 109750965 jon.reyes@stonybrook.edu
Joon Lee joon.lee.1@stonybrook.edu
Billy Schweigert

CSE 305 – Project Assignment 1

Team/Dating Site Name

JewelBond - “Discover the Key to the Heart. Love’s Treasure Awaits”

ER Diagram



Relational Model
SQL Description

```
CREATE TABLE Customer(
        CustomerId                      INTEGER,
        LastName                        CHAR(25),
        FirstName                       CHAR(25),
        Address                         CHAR(50),
        City                            CHAR(50),
        State                           CHAR(2),
        ZipCode                         CHAR(10),
        Telephone                       CHAR(10),
        Email                           CHAR(255),
        ProfilePlacementPriority        PPP,
        Rating                          Rating,
        LastActive                      DATETIME,
        PRIMARY KEY(CustomerId)
)

CREATE TABLE Account(
        AccountNumber                   INTEGER,
        Customer                        INTEGER,
        CreationDate                    DATE,
        CreditCardNumber                CHAR(16),
        PRIMARY KEY(AccountNumber),
        FOREIGN KEY(Customer) REFERENCES Customer(CustomerId)
)

CREATE TABLE Profile(
        ProfileId                       CHAR(24),
        ProfileName                     CHAR(24),
        Customer                        INTEGER,
        Age                             INTEGER,
        Address                         CHAR(50),
        Sex                             Sex,
        Height                          DOUBLE,
        Weight                          DOUBLE,
        HairColor                       INTEGER,
        DatingGeoRange                  INTEGER,
        DatingAgeRangeBegin             INTEGER,
        DatingAgeRangeEnd               INTEGER,
        PRIMARY KEY(ProfileId),
        FOREIGN KEY(Customer) REFERENCES Customer(CustomerId)
)

CREATE TABLE Photo(
        PhotoNo                         INTEGER,
        User                            CHAR(24),
        Path                            CHAR(255),
        PRIMARY KEY(PhotoId)
        FOREIGN KEY(User) REFERENCES Profile(ProfileId)
)

CREATE TABLE Hobby(
        User                            CHAR(24),
        Name                            CHAR(20),
        PRIMARY KEY(HobbyId)
        FOREIGN KEY(User) REFERENCES Profile(ProfileId)
)
```

```
CREATE TABLE Like(
        LikerId                                 CHAR(24),
        LikeeId                                 CHAR(24),
        DateTime                                DATETIME,
        PRIMARY KEY(LikerId,LikeeId),
        FOREIGN KEY(LikerId) REFERENCES Profile(ProfileId)
        FOREIGN KEY(LikeeId) REFERENCES Profile(ProfileId)
)

CREATE TABLE Referral(
        UserA                                   CHAR(24),
        UserB                                   CHAR(24),
        UserC                                   CHAR(24),
        DateTime                                DATETIME,
        PRIMARY KEY(UserA,UserB,UserC),
        FOREIGN KEY(UserA) REFERENCES Profile(ProfileId)
        FOREIGN KEY(UserB) REFERENCES Profile(ProfileId)
        FOREIGN KEY(UserC) REFERENCES Profile(ProfileId)
)

CREATE TABLE Date(
        DateId                                  INTEGER,
        User1                                   CHAR(24),
        User2                                   CHAR(24),
        CustomerRep                             INTEGER,
        DateTimeStart                           DATETIME,
        DateTimeEnd                             DATETIME,
        GeoLocation                             CHAR(?),
        BookingFee                              DOUBLE,
        User1DateRating                         Rating,
        User2DateRating                         Rating,
        PRIMARY KEY(DateId),
        FOREIGN KEY(User1) REFERENCES Profile(ProfileId),
        FOREIGN KEY(User2) REFERENCES Profile(ProfileId),
        FOREIGN KEY(CustomerRep) REFERENCES Employee(SSN)
)

CREATE TABLE Comment(
        Author                                  CHAR(24),
        Date                                    INTEGER,
        DateTime                                DATETIME,
        PRIMARY KEY(Author,Date,DateTime),
        FOREIGN KEY(Date) REFERENCES Date(DateId)
)

CREATE TABLE Employee(
        SSN                                     INTEGER,
        LastName                                CHAR(25),
        FirstName                               CHAR(25),
        Address                                 CHAR(50),
        City                                    CHAR(50)
        State                                   CHAR(2),
        ZipCode                                 CHAR(10),
        StartDate                               DATE,
        HourlyRate                              DOUBLE,
        PRIMARY KEY(SSN))

CREATE DOMAIN Sex CHAR(1)
        CHECK( VALUE IN('M','F'))
CREATE DOMAIN PPP CHAR()
        CHECK( VALUE IN('Super','Good','User'))
```

CREATE DOMAIN Rating CHAR(9)
CHECK(VALUE IN('Excellent','Very Good','Good','Fair','Poor'))

Rationale/Explanation

From the specification, obvious entities, corresponding attributes, and keys were simply produced accordingly. For example, for the most part, categories of data are listed as entities and items of data are listed as attributes This includes Customer, Profile, Referral, Like, Date, Employee.

Examining further, items of data that would become set valued attributes were converted to individual entities, since it is best for a column to only hold one value. This is with the exception of Physical Characteristics which was broken down into individual attributes of height, weight, and hair color within Profile, since these attributes have a 1 to 1 relationship. New additional entities include Account, Photo, Hobby, and Comment.

All together the design has the following entities:
- Customer
- Profile
- Referral
- Like
- Date
- Employee
- Account
- Photo
- Hobby
- Comment

Next, entity relationships are examined.

Customers Have
- Customers can have one or more Profiles.
- Customers can have one or more Accounts.
- Profiles and Accounts must be associated with a Customer.

Profiles Have
- Profiles contain zero or more Photos.
- Profiles contain zero or more Hobbies.
- Photos and Hobbies must be associated with a Profile.

Profile Actions
- Profiles can Like other Profiles.
- Profiles can make a Referral for two other Profiles to Date;
- Profiles can Date other Profiles.
- Profiles can Comment on Dates.

Dates Have Comments

Employees Supervise Dates
- Dates must be supervised by an Employee.
- Employee may be vacant and does not always have to be supervising a Date.

Then, entity types.

Strong entities are independent of other entities and uniquely identified by a single attribute primary key.

Strong Entities Include:
- Employee
  - Employees can exist w/o requiring the Supervision of a Date
- Customer
  - Customers are uniquely identified by Customer Id independent of Profiles and Accounts

Weak entities are uniquely identified by a composite primary key that includes an attribute which is a foreign key reference to the primary key of a Strong entity which their existence is dependent upon.

Weak Entities Include:
- Likes
  - Like(LikeeId,LikerId) depends upon Profile(ProfileId)
- Referrals
  - Referral(UserA,UserB,UserC) depends upon Profile(ProfileId)
- Comments
  - Comment(Author,Date,DateTime) depends upon Profile(ProfileId) and Date(DateId)
- Photos
  - Photo(User,No) depends upon Profile(ProfileId)
- Hobbies
  - Hobby(User,Name) depends upon Profile(ProfileId)

Not Sure:
- Profiles - Strong?
  - Profile have Id but must be associated with Customer
- Accounts - Weak?
  - Accounts have AccountNumber but cannot exist w/o Customer owner
- Dates - Weak?
  - Dates have Id but cannot exist without participating Profiles

Finally, domains:

Sex
- Must be Male or Female.

Profile Placement Priority
- Categories include:
  - Super-User, Good-User, and User-User

Rating
- Scale from:
  - Poor, Fair, Good, Very Good, Excellent

Collaboration Plan

Three-Way Division of Labor
(Somewhat Equal)

- One Teammate will handle aspects associated w/ Customers and Account transactions.
- One Teammate will handle aspects associated w/ Profile and Actions.
- One Teammate will handle aspects associated w/ Dates, Employee Supervision, and Comment feature.

	Attributes	*Teammate*
Customer	12	A
Account	4	A
	16	
Profile	11	B
Referral	4	B
Like	3	B
Photo	3	B
Hobby	2	B
	23	
Employee	10	C
Date	9	C
Comment	3	C
	22	
Total	61	