

CSC343 Assignment 3

Wendy Moniuk - 996659343

March 19, 2014

1. I am assuming that emails are unique among guests and hosts.

Every country and every city in my database has at least one guest and at least one host.

A host may have had no guests or many guests that have requested a couchsurf. A guest may have sent none or many couchsurf requests.

A member may have many friends or no friends.

A event may have zero invites or many invites associated with it. A member could be invited to zero or many events.

An event may have nobody going to it, or many going. A member could go to no events or many events.

A group must have at least one member.

A member can join zero or more groups.

A group may have zero or more messages. A member can post no messages or many messages. A post must be associated with at least one member and at least one group

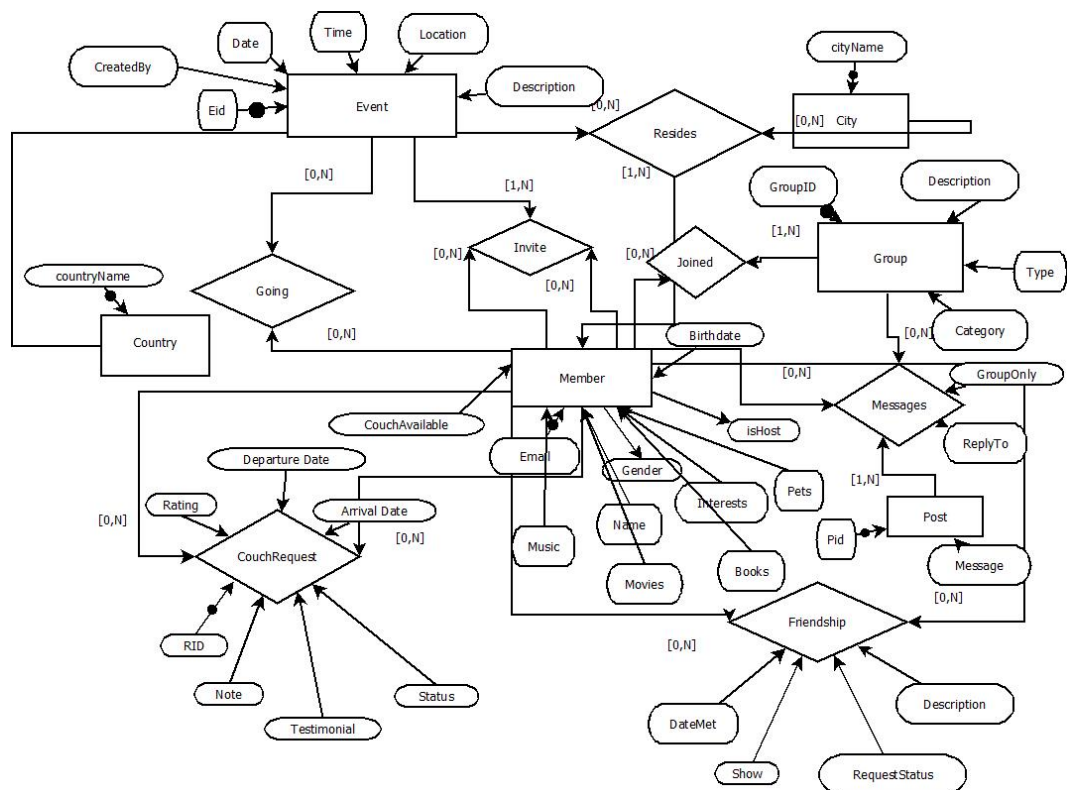
The Administrator is someone who has access to the database but they do not need information stored about them.

Search is a function implemented on the database and does not need any stored data.

The overall credibility score is calculated by the app and not stored in the database, it uses an aggregate function over the rating.

If a post is the first in a chain of replies it lists itself as what it is replying to.

2. See figure.
3. Member(email, gender, birthdate, isHost, name, pets, interests, music, movies, couchAvailable)
Country(countryName)
City(cityName)
Group(GroupID, Desc, type, category)
Event(Eid, **CreatedBy**, date, time, location, description)
Post(pid, message)



CouchRequest(RID, **guest**, **host**, status, departureDate, arrivalDate, rating, note, testimonial)

Residesin(mid, **country**, **city**)

Friendship(member1, member2, dateMet, Show, RequestStatus, Description)

Joined(mid, **groupID**)

Messages(mid, pid, **groupID**, groupOnly, replyTo)

Invite(eid, **inviter**, **invitee**)

Going(eid, mid)

```
4. CREATE TABLE member{
  email varchar(50) PRIMARY KEY,
  gender char,
  birthdate date,
  isHost boolean,
  name varchar(50),
  pets varchar(30),
  interests varchar(50),
  music varchar(50),
  movies varchar(50),
  couchAvailable boolean
};
```

```

CREATE TABLE country{
countryName varchar(20) PRIMARY KEY
};

CREATE TABLE city{
cityName varchar(20) PRIMARY KEY
};

CREATE TABLE group(
groupid integer PRIMARY KEY,
desc varchar(100),
type char(7), CHECK type = 'PRIVATE' OR type = 'PUBLIC',
category varchar(20)
);

CREATE TABLE event(
eid integer PRIMARY KEY,
createdBy varchar(50) REFERENCES member(email),
date date NOT NULL,
time time NOT NULL,
location varchar(50) NOT NULL,
description varchar(100)
);

CREATE TABLE post(
pid integer PRIMARY KEY,
message varchar(300)
);

CREATE TABLE couchrequest(
rid integer PRIMARY KEY,
guest varchar(50) REFERENCES member(email),
host varchar(50) REFERENCES member(email),
status char(8),
CHECK status = 'ACCEPTED' or status = 'PENDING' or status = 'DECLINED'
or status = 'MAYBE',
departureDate date,
arrivalDate date,
rating integer,
CHECK rating >=1 and rating <=5,
note varchar(50),
testimonial varchar(100)
);

CREATE TABLE residesin(
mid varchar(50) REFERENCES member(email),
country varchar(20) REFERENCES country,
city varchar(20) REFERENCES city
);

```

```

CREATE TABLE friendship(
member1 varchar(50) REFERENCES member(email),
member2 varchar(50) REFERENCES member(email),
dateMet date,
show boolean,
requeststatus char(5),
check requeststatus = 'YES' or requeststatus = 'NO'
or requeststatus = 'MAYBE',
description varchar(100)
);

```

```

CREATE TABLE joined(
mid varchar(50) REFERENCES member(email),
groupID integer REFERENCES group
);

```

```

CREATE TABLE messages(
mid varchar(50) REFERENCES member(email),
pid integer REFERENCES post,
groupID integer REFERENCES group,
grouponly boolean,
replyto integer REFERENCES post
);

```

```

CREATE TABLE invite(
eid integer REFERENCES event,
inviter varchar(50) REFERENCES member(email),
invitee varchar(50) REFERENCES member(email)
);

```

```

CREATE TABLE going(
eid integer REFERENCES event,
mid varchar(50) REFERENCES member(email)
);

```