Schema Mockup

**Bold** = primary key  
Underline = foreign key

User(**userID**, email, password, facebookID, active, isVerified, creationDate)

Community(**communityID**, name)

Event(**eventID**, name, startDate, endDate, longitude, latitude, description, ecmID, active, creationDate)

EventCommunityMap(**ecmID**, ecm, communityID)

UserCommunityMap(**ucmID**, userID, communityID, creationDate)

User(**userID**, email, password, facebookID, active, isVerified, creationDate)

A tuple in this relation represents the user’s login information to the app. The *userID* is the unique id of the user (e.g. 4) and is a unique integer. The *email* and *password* is the email of the user (e.g. email@email.com) and the password of the user (e.g. “password”) and are strings no more than 255 characters long. The *facebookID* is an id that will reference a user to their Facebook account so that the app can connect to their Facebook (e.g. ???). The *active* field is the flag that enables or disables a user account and is either ‘Y’ or ‘N’. The *isVerified* field is the flag that determines if a user account is verified through their email and is either ‘Y’ or ‘N’. The *creationDate* is a date and time string showing the creation date of the user account (e.g. “2016-10-10 18:00:39”) in the format YYYY-MM-DD HH:MM:SS.

Community(**communityID**, name)

A tuple in this relation represents all the communities available in our app whether they are default communities or user created communities. The *communityID* is the unique id of the community (e.g. 12) and is an integer. The *name* is the name of the category (e.g. “Hockey”) and is a string no more than 255 characters long.

Event(**eventID**, name, startDate, endDate, longitude, latitude, description, ecm, active, creationDate)

A tuple in this relation represents all the past and present events that have been created. The *eventID* is the unique id of the event (e.g. 111) and is an integer. The *startDate* and *endDate* are the start date and time, and the end date and time of the event respectively which will be a string (e.g. “2016-10-10 18:00:39”) in the format YYYY-MM-DD HH:MM:SS. The *longitude* and *latitude* field will be the location of the event stored as a double (e.g. 43.869464). The *description* of the event will display any additional information about the event and is a string no more than 2048 characters long. The *ecm* is an integer that which map the event to the communities that are tagged to the event. The *active* field is the flag that determines if the event is still active and is either ‘Y’ or ‘N’. The *creationDate* is a date and time string showing the creation date of the event (e.g. “2016-10-10 18:00:39”) in the format YYYY-MM-DD HH:MM:SS.

EventCommunityMap(**ecmID**, ecm, communityID)

A tuple in this relation represents the mapping of an event to its communities. The *ecmID* is the unique integer id of the mapping. The *ecm* is the integer value that will map the event to the community. The *communityID* is the integer id of the community that is being mapped. This will allow us to find all the communities that are tagged to an event.

Example tuples:  
(1, 2, 3)  
(2, 2, 22)  
(3, 4, 5)

UserCommunityMap(**ucmID**, userID, communityID, creationDate)

A tuple in this relation represents the mapping of a user to its communities. The *ucmID* is a unique integer id of the mapping. The *userID* is the integer id of the user that has joined this community. The *communityID* is the integer id of the community that is being mapped to the user. The *creationDate* is a date and time string showing the creation date of the event (e.g. “2016-10-10 18:00:39”) in the format YYYY-MM-DD HH:MM:SS.