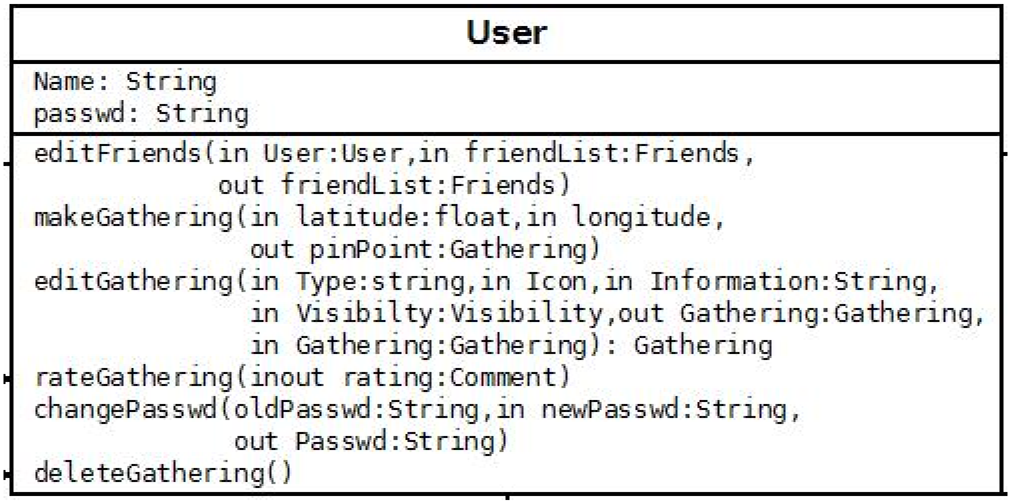
OCELOT

Ocelot Soft works Inc.
1601 N Pine St. Rolla, MO. 65401
Theodore Hunter, Matthew Lindsay, Hilary Hamlin
Project Planning
Gathering Application Spatial Map
October 29th, 2013.

Gathering Application Spatial Map

This project is a tool that lists gatherings spatially rather than chronologically. A map application will allow users, via either internet or phone, to see and place markers on a map for various types of social gatherings. Each marker lists pertinent information such as the address, time of the gathering, and type of gathering. The marker can either be deleted or it will disappear upon gathering completion.

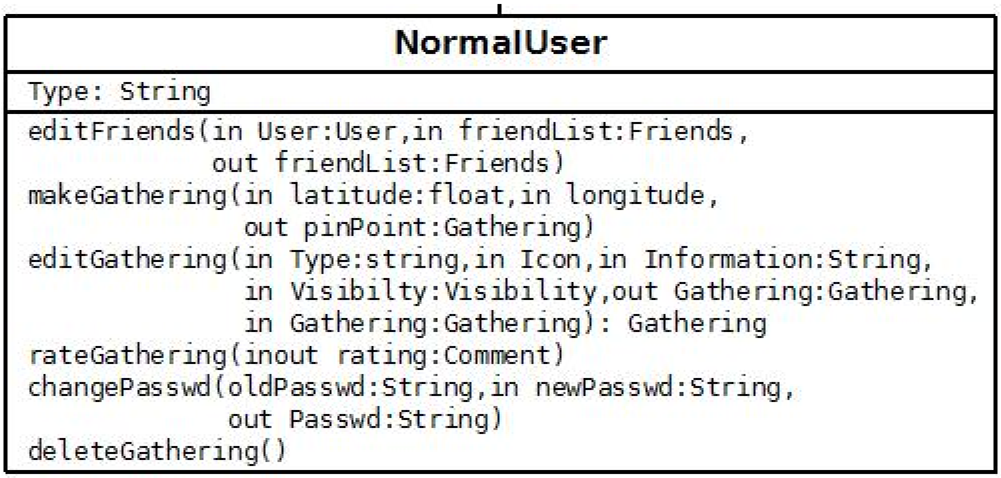
USER



```
User
Name: String
passwd: String
editFriends(in User:User,in friendList:Friends,
            out friendList:Friends)
makeGathering(in latitude:float,in longitude,
              out pinPoint:Gathering)
editGathering(in Type:string,in Icon,in Information:String,
              in Visibilty:Visibility,out Gathering:Gathering,
              in Gathering:Gathering): Gathering
rateGathering(inout rating:Comment)
changePasswd(oldPasswd:String,in newPasswd:String,
             out Passwd:String)
deleteGathering()
```

The Gathering Application Spatial Map is user driven. The program does little on its own, without the user interaction it is just a map. A user must log in to use the application thus every User requires a password and unique user name. The password can be changed but the user has to know their password to be able to change it. A user can select from zero to an infinite number of friends from the list of other users A logged in user can create one gathering at a time, each gathering is displayed as a simple point on an Open Layers map so it requires an input of the latitude and longitude position. This can be gathered by tracking a mouse click or tap on a touchscreen, the map knows the latitude and longitude where it is clicked. Once the gathering is made the user should be able to edit their own gathering, it is here that they can set specific information for their gathering. If a user so chooses they can comment or rate on a gathering. There are no limits to where or how many comments a user can leave on a house. If the user made the gathering by mistake or the user chooses to cancel the gathering they can delete it.

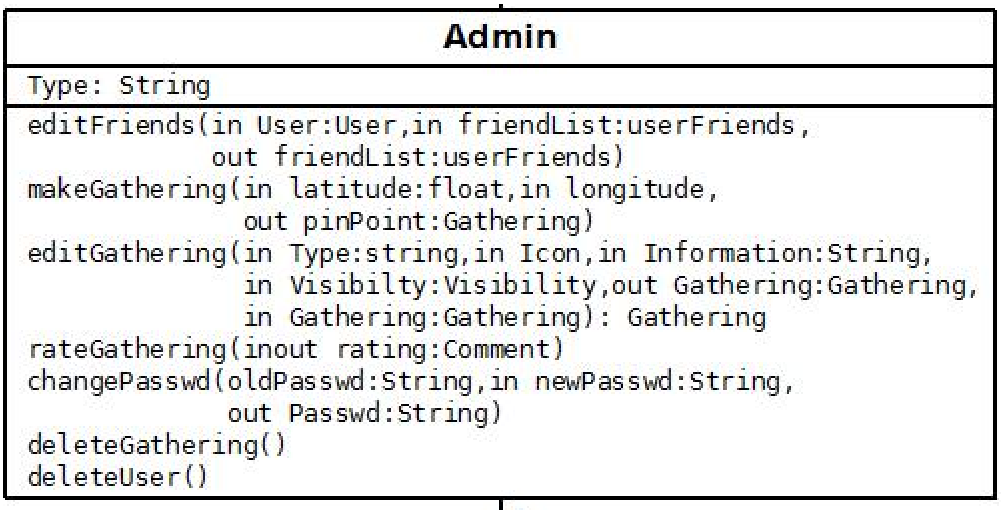
NORMAL USER



```
NormalUser
Type: String
editFriends(in User:User,in friendList:Friends,
            out friendList:Friends)
makeGathering(in latitude:float,in longitude,
              out pinPoint:Gathering)
editGathering(in Type:string,in Icon,in Information:String,
              in Visibilty:Visibility,out Gathering:Gathering,
              in Gathering:Gathering): Gathering
rateGathering(inout rating:Comment)
changePasswd(oldPasswd:String,in newPasswd:String,
             out Passwd:String)
deleteGathering()
```

Users come in two flavors the first being a normal user. They inherit all the functionalities of user. The only difference being a there is now ownership. A normal user's gatherings are protected, meaning only they can edit or delete their content. The only thing a normal user can do to a gathering they did not make is leave a comment or rating. A normal user can edit and remove their friends, edit their password, edit or delete their gathering, and finally edit or delete their ratings and comments.

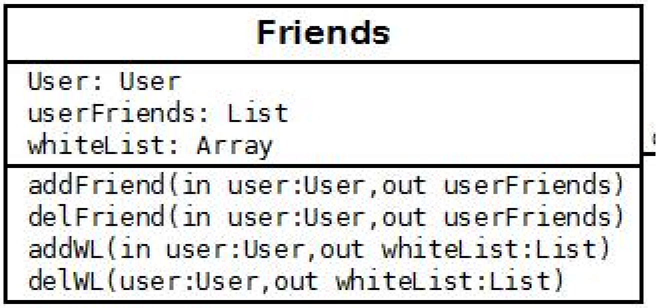
ADMIN



```
Admin
Type: String
editFriends(in User:User,in friendList:userFriends,
            out friendList:userFriends)
makeGathering(in latitude:float,in longitude,
              out pinPoint:Gathering)
editGathering(in Type:string,in Icon,in Information:String,
              in Visibilty:Visibility,out Gathering:Gathering,
              in Gathering:Gathering): Gathering
rateGathering(inout rating:Comment)
changePasswd(oldPasswd:String,in newPasswd:String,
             out Passwd:String)
deleteGathering()
deleteUser()
```

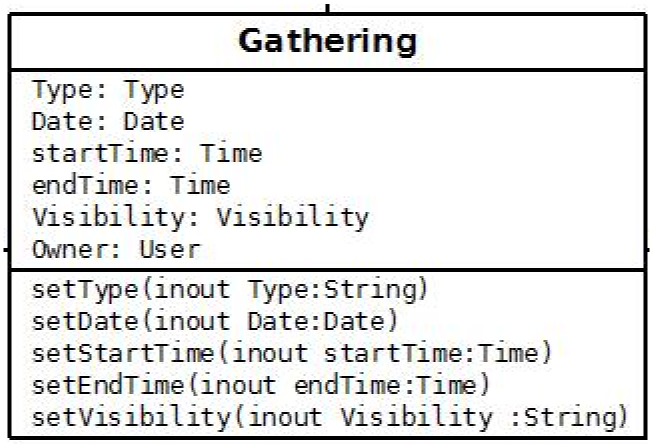
The second type of user is the admin. The admin inherits all functions from user but an admin has extended privileges and thus has the ability to manage and edit content. They have the power to remove any gathering, comment, rating or user as they see fit. An admin user can edit and remove any other user, they can edit and remove not only their friends but any user’s friends, they can edit their own password or another user’s password, since we have no password recovery system this is the only way to let a user who has forgotten their password back into the system. Without this functionality a user that forgot their password would have to make a new account. An admin can edit or delete any user’s gathering, and finally an admin can edit or delete any ratings and comments. Most of the admins powers will be used to ensure the map is not spammed with fake accounts, fake gatherings or inappropriate content.

FRIENDS


Friends
User: User
userFriends: List
whiteList: Array
addFriend(in user:User,out userFriends)
delFriend(in user:User,out userFriends)
addWL(in user:User,out whiteList:List)
delWL(user:User,out whiteList:List)

A user can add friends; this list can later be used to edit visibility in a gathering. Friends can be selected from a list of other users. Once added as a user's friend they can see gatherings listed by that user as 'friends only'. To make a more exclusive list a user can add some of their friends to a white list. These white listed friends are chosen from a list of friended users. The user's white listed friends can see gatherings listed by that user as 'white list only'. The same goes for the user they can see friend only gatherings listed by other users who have added them as a friend and they can see the white list gatherings of their friends who have added them as a whitelisted friend.

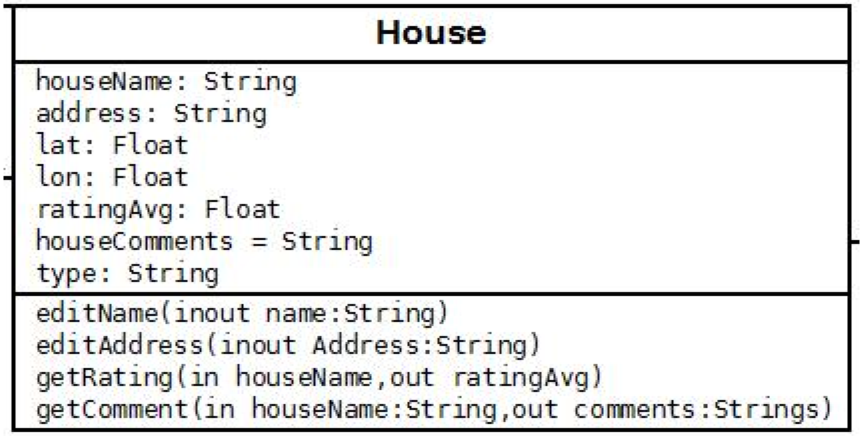
GATHERING



```
Gathering
Type: Type
Date: Date
startTime: Time
endTime: Time
Visibility: Visibility
Owner: User
setType(inout Type:String)
setDate(inout Date:Date)
setStartTime(inout startTime:Time)
setEndTime(inout endTime:Time)
setVisibility(inout Visibility :String)
```

A user can only make one gathering at a time; once a gathering is placed it can be edited. The owner field is set upon creation and used to protect the gathering. Only the gathering creator, or an admin, can edit the gathering. The owner of the gathering can decide what kind of gathering they want to host, what day the gathering is on, when the gathering starts, when the gathering ends, where the gathering is, and who can actually see the gathering on the map. A user can select the type of gathering from a list of admin made gatherings, such as birthday party or study group. Each type of group has an icon associated it, changing the type will affect the icon that appears on the map. Visibility can be set to public, where everyone can see the gathering on the map, friends, where only users on the owner's friend list can see the gathering on the map, and whitelist, where only friends on the owner's white list can see the gathering on the map. Gatherings only run for a finite time, as of right now it is limited to one day from a start time, when the user wants guests to arrive, to an end time, when the user wants guests to leave. Once the end time on the specified date has past the gathering expires and is deleted from the database. Since a user can only have one gathering at a time they have to either delete their gathering or wait for it to expire before they can make a new gathering.

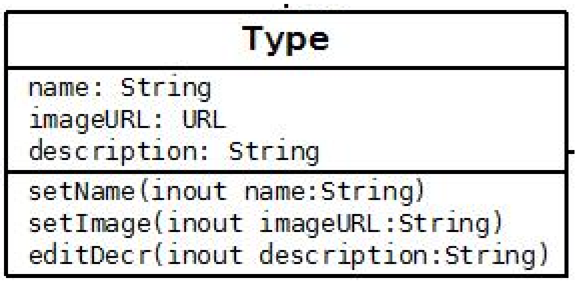
HOUSE



```
House
houseName: String
address: String
lat: Float
lon: Float
ratingAvg: Float
houseComments = String
type: String
editName(inout name:String)
editAddress(inout Address:String)
getRating(in houseName,out ratingAvg)
getComment(in houseName:String,out comments:Strings)
```

To make gathering information more robust a user can create a house and select it so the gathering has a physical location attached to it. The house can only be selected if its latitude and longitude is within a certain error range of the house. This should somewhat keep users from abusing other houses. Since gatherings disappear after they are scheduled to end the house keeps the average rating and comments. The rating average is calculated by adding up the all the ratings scores, numbers from zero to five, a house has received and dividing it by the number of ratings. When a gathering is made with an existing house the overall rating and comments are added to the information so a user looking at a gathering can make an educated decision whether they want to attend based on other user's previous experiences. The house is also where the address is stored. The address is not used for directions, such a task being beyond the scope of the project. The address it is kept in one field as a string and used for reference, if a user wants to use the address for directions, sadly, they would have to use another service.

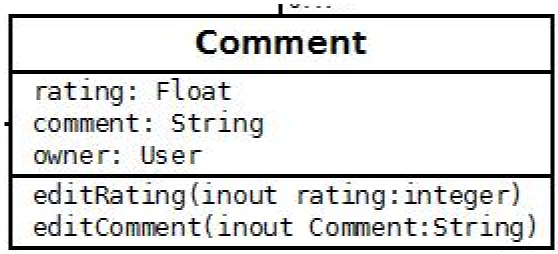
TYPE



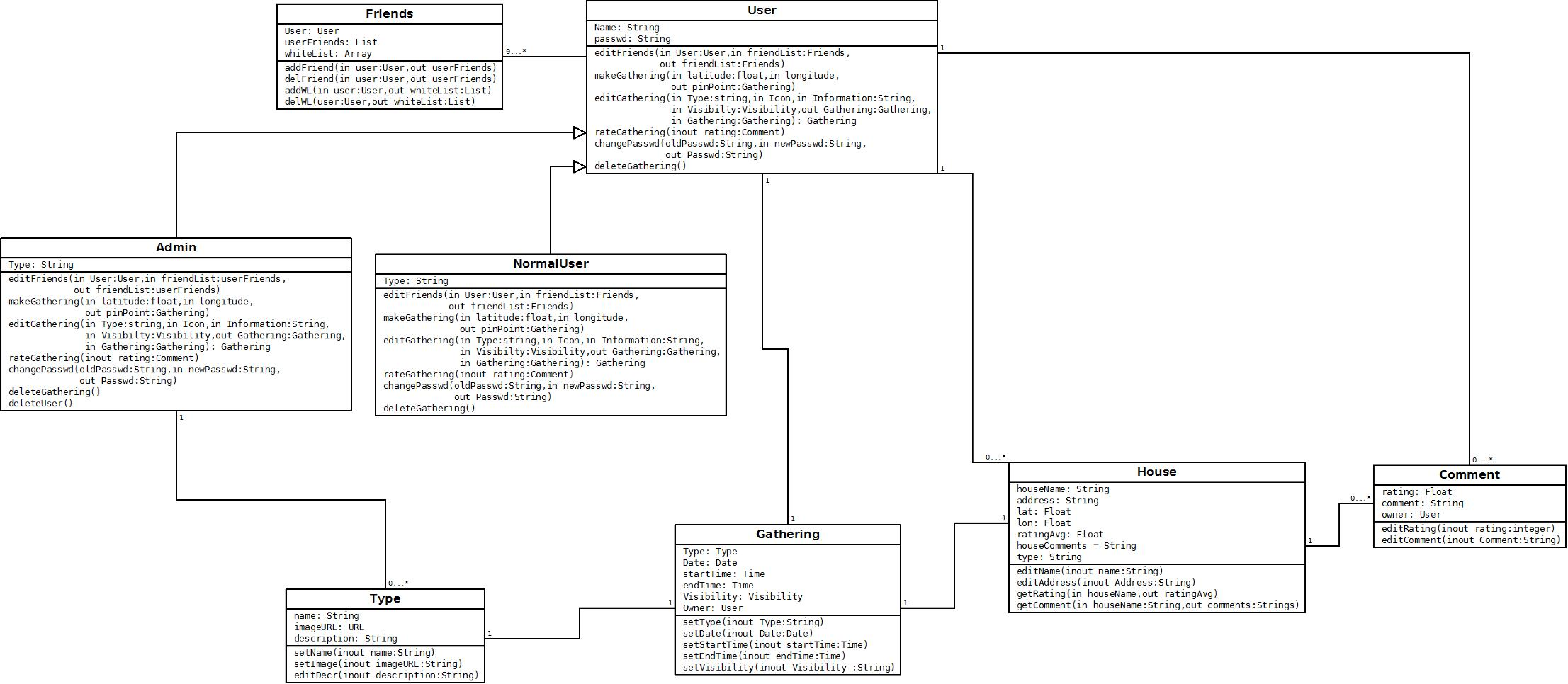
```
Type
name: String
imageURL: URL
description: String
setName(inout name:String)
setImage(inout imageURL:String)
editDecr(inout description:String)
```

Every gathering must have a type. The type contains a name, a description to describe the nature of the gathering and a unique icon. When a user chooses a type for their gathering the change is reflected on the map. The generic gathering point is replaced by the unique icon coupled with the type. A normal user can only select a type through their gatherings. Type creation is left up to the admins of the program. They can make a type by setting the name, giving the type a concise description and listing the URL to some unique image. By keeping the type a protected item we can stop normal users from creating types with inappropriate images and descriptions from polluting the map.

COMMENT


Comment
rating: Float
comment: String
owner: User
editRating(inout rating:integer)
editComment(inout Comment:String)

The only actions a normal user can take on a gathering that user does not own is to leave comments and ratings. The comments and ratings are not attached to the gathering since the gathering is deleted after it expires; rather they are attributed with the house a gathering is built on. The rating will be done graphically using stars; a user can decide how much they liked a certain gathering based on their experiences. They can leave zero to five stars. These numbers appear with the comment along with the user who made them. So users can quickly gauge any gathering an average score is calculated from these ratings and boldly displayed with the gathering’s information. Any comments attached to that specific house will also be displayed with gathering information so any user can view them. The only people who can edit or delete a comment or rating are the comment or rating owner or a user with admin privileges. Some comments and ratings may be deleted by an admin if they contain lewd or inappropriate remarks.


Friends
User: User
userFriends: List
whiteList: Array
addFriend(in user:User,out userFriends)
delFriend(in user:User,out userFriends)
addWL(in user:User,out whiteList:List)
delWL(user:User,out whiteList:List)
0...*
User
Name: String
passwd: String
editFriends(in User:User,in friendList:Friends,
out friendList:Friends)
makeGathering(in latitude:float,in longitude,
out pinPoint:Gathering)
editGathering(in Type:string,in Icon,in Information:String,
in Visibilty:Visibility,out Gathering:Gathering,
in Gathering:Gathering): Gathering
rateGathering(inout rating:Comment)
changePasswd(oldPasswd:String,in newPasswd:String,
out Passwd:String)
deleteGathering()
1
1
1
Admin
Type: String
editFriends(in User:User,in friendList:userFriends,
out friendList:userFriends)
makeGathering(in latitude:float,in longitude,
out pinPoint:Gathering)
editGathering(in Type:string,in Icon,in Information:String,
in Visibilty:Visibility,out Gathering:Gathering,
in Gathering:Gathering): Gathering
rateGathering(inout rating:Comment)
changePasswd(oldPasswd:String,in newPasswd:String,
out Passwd:String)
deleteGathering()
deleteUser()
1
NormalUser
Type: String
editFriends(in User:User,in friendList:Friends,
out friendList:Friends)
makeGathering(in latitude:float,in longitude,
out pinPoint:Gathering)
editGathering(in Type:string,in Icon,in Information:String,
in Visibilty:Visibility,out Gathering:Gathering,
in Gathering:Gathering): Gathering
rateGathering(inout rating:Comment)
changePasswd(oldPasswd:String,in newPasswd:String,
out Passwd:String)
deleteGathering()
0...*
0...*
House
houseName: String
address: String
lat: Float
lon: Float
ratingAvg: Float
houseComments = String
type: String
editName(inout name:String)
editAddress(inout Address:String)
getRating(in houseName,out ratingAvg)
getComment(in houseName:String,out comments:Strings)
1
1
0...*
Comment
rating: Float
comment: String
owner: User
editRating(inout rating:integer)
editComment(inout Comment:String)
1
Gathering
Type: Type
Date: Date
startTime: Time
endTime: Time
Visibility: Visibility
Owner: User
setType(inout Type:String)
setDate(inout Date:Date)
setStartTime(inout startTime:Time)
setEndTime(inout endTime:Time)
setVisibility(inout Visibility :String)
1
1
0...*
Type
name: String
imageURL: URL
description: String
setName(inout name:String)
setImage(inout imageURL:String)
editDecr(inout description:String)
1