Description of the class diagram:

1. Registration
   1. registerUser(): This is a method that will be invoked when a user clicks on “Register Me”. The parameters being passed would be the OSU Email ID used to register
   2. validateFields(): This method will be used to validate the email that the user enters. This will check the format of the email-ID, and also if the user has a valid OSU Email ID. Parameters: OSU\_Email\_ID
   3. sendActivationMail(): This method is invoked to send an activation email to the user while registering. The user will be successfully registered in the BucKonnect DB only after he receives this email and clicks on the confirmation link. Parameters: OSU\_Email\_ID.
2. LoginInfo:
3. login(): This method is invoked when user enters the Username and password and then clicks on “Login” button. This method will then check the validity of the user by looking up to see if the username and password combination is correct. If it is correct, the profile page of the user will load, else it will throw an error saying the combination is wrong.

Parameters: username, password

1. User:
2. searchUser(): This method is invoked when user clicks on a button on the screen that will enable the user to search for other BucKonnect users. The user could enter the first name or the last name of the user that he intends to search for.

Parameter: username (string)

1. searchGroup():This method is invoked when user clicks on a button on the screen that will enable the user to search for BucKonnect groups. The user could enter the name or the interest of the group that he intends to search for.

Parameter: groupname (string) interest (string)

1. searchEvent():This method is invoked when user clicks on a button on the screen that will enable the user to search for BucKonnect events. The user could enter the name of the event that he intends to search for.

Parameter: eventname (string)

1. joinGroup(): Method allows user to join a group. When a user clicks on this, an email form opens up and the user would be able to send an email to the admin of the group he intends to join.

Parameter: groupid

1. reportAbuse():

Parameters: OSU\_Email\_ID of user to be reported

1. blockUser():

Parameters: OSU\_Email\_ID of user to be blocked.

1. createGroup(): New group class object is created and fields are accordingly populated.

Parameter: GroupId

1. createEvent(): New event class object is created and fields are accordingly populated.

Parameter: EventID

1. PrivacySettings:
2. updatePrivacySettings(): Allows for user to set and update Permissions such as call, text and view permissions.
3. GeneralSettings:
4. updateGeneralSettings(): Allows for user to set and update general basic settings such as contact information, major, interests, etc..
5. Calendar:
6. addEntry()
7. removeEntry()
8. updateEntry()

All the above methods will allow the user to add/remove/update entries in their calendars respectively.

Parameters: CalendarEntry, OSU\_Email\_ID

1. Group and Event:
2. addPost()
3. removePost()

a and b allow user to add and remove posts within the group / event.

1. Admin and PEC:
2. addUser()

Allows the admin / PEC to add users to the group / event

1. addSA() , addSEC()

Allows PA to add an SA to a group and a PEC to add an SEC to an event.

1. revokeSA(), revokeSEC()

Allows the PA to revoke the rights of an SA and a PEC to revoke the rights of an SEC.

1. blockUser()

Enables the Admin / PEC to block users from the group / event.

1. updateDetails()

Enables the Admin/PEC to update details of the group/event.

Parameter: groupID,EventID

Parameters for a-d : OSU\_Email\_ID

1. Post:
2. editPost():

Allows users to edit posts that have been already posted by them.

1. comment():

Allows users to comment on posts on the group / event.