



## Sprint 4 Team Report

**Team: 108**

**Repo URL:** <http://github.ccs.neu.edu/cs5500/Team-108-SP19>

**Reviewer:** Robert Bronstein

**Score:** 97

### Comments

My only comment with respect to demo was at some point in the demo there was an error message (I think it merely said "Registration Failed") but didn't say how or why. Though I feel like I've probably mentioned to you not giving too much information away about the system on exception messages as it is a security risk to do so, this I think is the opposite extreme of being too opaque. If the user doesn't have some high level understanding of what went wrong, they won't be able to take corrective action. They most likely will not be able to view stored information in the database in order to determine what transpired.

A few code quality notes:

Some unused methods as before.

A note about the IUserGroup interface...looking at this it seems that there was some expectation that there would be base methods common to both a user and a group, but none were ever supplied (well actually there was at least one, namely getId but it is listed twice [once in each child method instead]).

I would either:

1. Delete the IUserGroupInterface
2. Refactor methods common to user and group into the IUserGroup Interface

For only something small like getId, I would question the utility of a shared base user interface (except insofar as it leaves open the possibility of using the composite pattern) though.

GroupJPA's removeUserFromGroup method is a little on the larger side and may be a candidate for refactoring (e.g. by factoring out the check to see if someone is a moderator).

Same for InviteJPA's searchInvitebyGroupCode method.

UserService checkString method appears to be validating password policy was followed but has no documentation.

ProfileController search entity method returns null (I would throw an exception for something not



implemented yet instead.).