**WEEKLY STATUS REPORT FOR THE WEEK ENDING ON  
*Jeremy Scott, Chad Reynolds, Darcie Odom***

February 27, 2015

***ACTIVITIES:***- Meeting on February 22:

* Discussion on some of the details in the use cases, how they should be split up and whether some of them should be condensed together.
* Initial work done on database design, deciding what data will need to be stored at a minimum.
* Deciding on what should be accomplished for the prototype, what will be most useful to flesh out in a prototype to help us make design decisions for the final version. We decided that a basic navigation skeleton should be created for the Android application, a working connection with the server should be demonstrated, and the server should be able to retrieve data from the database. Testing if the application-server and server-database connections function as we expect will either reinforce our design decisions or guide us towards an improved design.

***ACCOMPLISHMENTS:***

* Editing and improvements on use case descriptions
* Drafts of sequence diagrams to gain an idea of the number and types of object we will need
* Database relations have been initally drafted (see below)

***ISSUES/PROBLEMS:***

Deciding on how to implement the application-to-server connection

***PLANS:***

* Meeting again next week
* Drafting sequence diagrams
* Android application navigation with screen mock-ups
* Server providing data to the application
* Database tables created

**DB Relations**

User {userName(PK), firstName, lastName, password, bioBlurb, email, universityID(FK), accountStatus}

Messages {msgID(PK), sendingUser, receivingUser, body, subject, datetime}

UserBlock {userName(PK, FK), blockedUserName(PK, FK)}

MatchResponses {userName(FK), suggestedUserName(FK), response}

Class {classID(PK), universityID(FK), className}

University {universityID(PK), universityName}

UserEnrollment (User-Professor-Class) {userName(PK, FK), teachingID(PK, FK), enrollmentStatus}

Professor {professorID(PK), professorFName, professorLName}

Teaching (Professor-Class) {teachingID(PK), classID(FK), professorID(FK)}

PasswordReset {resetID(PK), userName(FK), resetCode, expirationTime}