**Proposal Document:**

As members of the Android Team, Tim Schonberger, Cody Rozic, Mike Colistro, and Gary Graham, will be responsible for the development of a fully-fledged android application, used for interaction with the web API developed by the API/Database team. The overriding purpose of the team is to create what will be the major part of the application that actually interfaces with users, along with what the front-end web team creates. Our place in the overall project is to create a functional and well-polished application that not only performs all necessary tasks, but also has a very high level of usability and is intuitive to new users.

**Scope of the application:**

The android application must replicate the vast majority of features found on the web side. It is possible due to the complexity of particular tasks, that they may not be replicated onto the device. Regardless, this entails at least the following:

* Registration to the application.
  + Registration exists for the purposes of notification of a match between buyer and seller.
  + Registration information will be passed back over to the server for verification. Nothing is done locally.
  + Must be encrypted before transfer over the line.
* Login to the application.
  + Must be encrypted over the line.
* Peruse list of currently available books and their associated pictures and courses:
  + List is loaded through the web API. Data scraped by the web scraping team is passed to us in JSON or some other format, and displayed by department heading.
  + Different organizational options depending on year level of the class or anything like that.
* Possibility of locally cached course list after first load, with it updating periodically, and manually when it is perused.
* Search Sorting.
  + By course number.
  + By book name.
  + By currently available listings.
* The ability to have a listing found by any of it’s information, not just title.
* In all different views, have the ability to tap on a “BUY” or “SELL” button, that allows for speed in usability.
  + “Buy” button automatically creates an entry in the “Wants” table of the database.
  + “Sell” button gets more information, like edition number, desired price, and desired method of contact, and then creates a listing in the “Listings” table.
  + When there is a match between the two tables, both users are notified.
* Option to sell a book that is not necessarily in our list, via ISBN(and potentially the android barcode scanner).

**By Midterm:**

* Register/login
* peruse list of available books
* Get in contact with listing owner.
* Input course number, get book.

**By end of term:**

* Display full course list with ability to look through old editions.
* In course list, have an "I want to sell this book/I'm looking for this book" buttons.
* Option to add books not in the list, by ISBN(camera scanner)
* More sophisticated Searching
* Polished UI

**Time estimates:**

* Partially functional non-database-touching prototype: 35 hours
* Functional but ugly database-utilising application: 80 hours
* Polished application: 120 hours.

The following is a small mockup of the UI generated to give the other teams an idea of roughly how we intend the application to look.

