Meeting Notes from 11/25/2015

Discussion:
Demo the MVP and discuss features:
- Autocomplete for searching courses; we looked through the catalog to get data.
- Noticed that the snippet currently appears as large as the amount of content inside it.
- Snippets now have markdown supported using external library. We might try to improve styling/add toolbar.

For the final version of the app:
- What happens after clicking subscribe button? Perhaps a toast pops up to notify the user that he is now subscribed to the course. Also, if user is already subscribed, button click should unsubscribe.
- We will look into certificates for user authentication.
- We will have real data on user home page.
- Use socket.io for real time updating of the feed and tracking number of active users in a session.
- Flag snippets that are bad/incorrect; display number of flags for a snippet.
- Unlocking: hide most of the snippets in the feed, until a user has contributed 3 good snippets.
- Add a filtering feature that allows users to view other users' stashes.
- If implementing fixed size snippets, we might need another view to show all the content inside a snippet.

Some design decisions:
- Currently snippets are immutable. One downside is that if you authored the snippet and made a mistake, you can't edit it.
- If someone else has already created a session recently, notify a user when they attempt to create another session for the same class.
- Consider how to navigate backwards
- When a user has 4 classes, consider which sessions to display on the user home page (maybe the most recent ones). How do we want to display different sessions for different courses: same list or tabs by course?
- Why do we not automatically create sessions at the class start/end time? Classes can have variable times, could be cancelled, and have in class exams. That is why we are going to have warning displayed if a user tries to create session for a lecture that already has a session recently created.

TA ideas:
- Do we want a feed for the entire class?
- Reminder for testing:
    o Tests for model
    o Test UI manually and explain what/how we tested
    o Look at security and how the app protects against attacks

Plan:

Top features for final:
- Connecting with real data
- Markdown
- Unlocking/hiding
- Flagging
- Filtering stashes by user