Grading Feedback:

Rubric - just because you do everything right, does not mean perfect score
solid = B, really good = B+, go out of your way = A
There is no proper set of deliverables.

STEAM in Section 2. 204.

Sprint has been extended by a week because we did not have enough time.

Impress Sneha by going way beyond!

Let's discuss what we did:

- Olivia and Lindsay worked on UML rough draft
  - Brainstormed back end and algorithm
  - Research, looking through how to use a google maps API to do calculations for distance, locations, times.
  - See if any algorithm online for google pool or LYFT etc
- Lindsay did a usability study on friends, paper prototypes project description
- Sophie, making the rough UI coding
  - Html, css, bootstrap, angular (maybe) *front end

- back end - use Node.
In Web Dev, MEAN stack. HTML.
Maybe back end RoR?

Java - google maps lib.
Lib has nice APIs, wrappers, really easy to implement that (has some exceptions).

If we use Java, we all know it, so can all help out.

Good group:
UI with dummy data for navigation.
Descriptions on tech stack, licensing, uploaded doc on confluence, researched algo, in git, pushed in diff branches,

JIRA is for task tracking. Push it to in progress to track that. Actual research, if u have stuff put it in git so that TAs can see.

What will we achieve in the next week:

- Research with mapping (Michael has experience with google maps API)
- Plan!
- How to implement specific Objects in the overall UI.
  - Users, functionality, what they have, what parts

    - Data, what we can manipulate as admin

- figure out the tech stack!
- class diagram (michael)
- dummy data - try using
- research on the different types of testing

Tech stack - think about the test frameworks.
For front end, have testing frameworks.
For React its ReactUtils.
For HTML- ?

Capybara?

Due date for this sprint Oct 27.

ASK THE CLIENT:
What are the extra permissions an admin should have, that regular end user does not have?

Admin - see the list of users logged into your app.
End User -
*might be a group of admins doing work and they all have the same functions.

ONce we have API
Write the interface for the method, push
Test cases
Then start impl.
Make sure have the same sequence of commits in git

Push tests
Then push the implementation.

Tests should be completed.
Write a dummy api.
Signature is void. Dont do any processing.
All tests fail.
Then all tests should pass once implement.

Can have a test branch.
*****DO NOT WRITE CODE BEFORE TESTING******

Interface: have an interface file, then API file, no implementation in the method. If it returns an int return 0, if its void do nothing, etc. stubs.

JIRA

Mass Open Cloud aka MOC (like AWS) create instances deploy whatever u want like Jenkins use it as a software.

Remote server

If we all have Windows and a server runs on linux, create an instance, have a VM in the cloud. Instead of having local.
Final app should be deployed in there so that they can use it.

Jenkins config. Should be given to us or told how to configure it

If not enough time to do the testing for front end, at least write a test spec.