

## **Group Project Description**

### **CSCI 3308-F13**

All students in Software Engineering Methods and Tools are required to complete a group project. Each group will submit a project proposal, which must meet certain minimum standards and have that proposal approved by Judy. Groups may choose to do the default project as described below or propose a project of their own description. Project proposals are due by midnight Friday, Nov 8. Groups opting to do the default project described below need only submit a list of group members along with a statement that they are doing the default. Others need to submit the information described in Section I.E.

## **I. LOGISTICS**

### **A. Due Date/Time**

Project is due at scheduled time for final exam (this project replaces a final exam in the course). Due dates for intermediate deliverables are provided in the Deliverables section below.

### **B. Submission**

Intermediate project deliverables will be demonstrated to TAs during scheduled Status Check Sessions. Final project will be deployed to Heroku and source code pushed to GitHub. Groups will make one submission to moodle that includes links to their Heroku and GitHub repositories.

### **C. Grading**

Total project grade is 100 pts. Distributed as follows:

- 10 pts for identifying team by end of lab on Nov 6
- 10 pts for project proposal submitted on or before deadline
- 60 pts (15 pts for each) intermediate sets of deliverables
- 20 pts for final project submission

Going beyond original project plan may be rewarded with extra-credit (points being deposited in the extra-credit pot)

### **C. Intermediate Project Status Meetings**

Intermediate Status Check Meetings will be held weekly. See Deliverables schedule below. Groups will sign up on the moodle scheduler for one Status Meeting slot, which will be the group's recurring meeting time for the entire duration of the project. Status meeting slots will be scheduled during the TAs usual lab time. The rest of lab time will be a group work session focused on the deliverables for the following meeting.

### **D. Groups**

Groups will be self-forming. Allowable groups will have 2 to 3 students, all of whom are able to attend the same lab session. Note that it is not required that all group members are registered for the same lab, but only that they can attend the same lab for rest of the semester for status meetings. If a student does not find him/herself in a group by Wednesday, Nov 6 they should attend their regularly scheduled lab and the lab instructor will place them in a group.

### **E. Project Proposals**

Groups will submit a Project Proposal by midnight, Friday November 8. Groups opting for the default project need only submit a list of group member names along with a statement that they will be doing the default project.

The proposal should be similar to the default project description provided below and will include the following:

- (a) Brief overview of the project goals.
- (b) List of views, which must number at least 5. Each view must be described with general capabilities and constraints.
- (c) A list of stakeholders or user roles for such a system
- (d) The names of each group member.
- (e) Convincing statement that the group members have the necessary skills to complete the project.

### **F. Development Tools**

- (a) Project must be created in Ruby on Rails. Groups may use whichever version of the framework and language that they prefer.
- (b) Teams must use Pivotal Tracker
- (c) Teams must use Cucumber
- (d) Teams must use a unit testing framework, not necessarily Rspec
- (e) Teams must use a code-coverage tool, not necessarily simplecov

## **II. PROJECT DELIVERABLES FOR ALL PROJECTS**

Groups need not view the schedule below as a guide for pacing their project but rather as hard deadlines for completing that aspect of the project. In other words, groups may well work at an accelerated pace, making more progress on their project before the deliverable due date.

### **A. Project Proposal due by midnight, Friday, Nov 8**

See Part I.E above.

### **B. Storyboards due at Wednesday status meeting, Nov 13**

Create storyboards or other form of lo-fi sketches to represent the views. Come to agreement on UI.

### **C. List of Models due at Wednesday status meeting, Nov 13**

Identify list of models along with model attributes and relationships among models.

### **D. User stories due at Wednesday status meeting, Nov 20**

- Create user stories to represent the interests of all stakeholders listed in the proposal.
- User stories must be created in Pivotal Tracker

### **E. Make estimate of weekly progress based on story points due at Wednesday status meeting, Nov 20**

### **E. BDD due by Wednesday status meeting, December 4**

Write cucumber features for user stories and write code to make them pass.

**F. Use TDD to develop controllers - show progress during status meeting on December 11**

Test cases must be available for view by TAs and unit testing framework must be in use (not necessarily Rspec). The ability to apply (or description of having applied) refactoring must be demonstrated.

**F. Final Project Submission by time scheduled for final exam**

Push project to GitHub and Deploy to Heroku

### **III. DEFAULT PROJECT**

For the default final project the assignment is for students to work in groups to create a simple clone of Facebook using Ruby on Rails. This application is to have similar functionality to what Facebook (then known as [TheFacebook.com](http://TheFacebook.com)) launched with in 2004. The following are the bare minimum requirements:

- 1) Create user stories for all the functionality for the website. Additionally, before coding, do a low-fi sketch of your user interfaces. These are expected to change as the project evolves so do not worry about updating them. These are to be done in order to have some direction when starting.
- 2) Create views for Logging In, Creating an Account, the Newsfeed, the Profile Page, Searching / Adding Friends, Profile Preferences
  - a) Creating an Account and Login are pretty self explanatory views. After logging in a user should go to their Newsfeed view.
  - b) The Profile Preferences view should allow the user to enter basic information in text areas such as Interests and Quotes. Additionally, it should allow users to view their friends (their names should be hyperlinks to their profile pages) as well as have the option to de-friend individuals that they no longer wish to be friends with. This view should also allow the user to clear off their entire wall. This view is also where users can accept or deny friend requests.
  - c) The Profile Page view should display information about the user, as well as contain a Wall where the user or friends can post information as well as read the contents of the wall. Remember, the users' names who wrote on the wall should all be hyperlinks to their profile pages. This content should not be viewable by users who are not in the person's social network. If a user that is not in the user's social network attempts to view the user's Profile Page it should default to simply the user's name, with a button to send a friend request.
  - d) The Search for Friend / Add Friend view should allow users to type in a last name, and then it will display all the users with that last name and allow the user to click a button to send a friend request to that user.
  - e) The Newsfeed view should show all the things posted on a users wall as well as anything posted on all of the user's friends' walls. Again, remember the users' names should all be hyperlinks to their profile pages! This page should also contain links to Profile Preferences, Search / Add Users, and the user's own Profile Page.

In general, log on to Facebook and see how it works! This is a good guide for your project.

Things that you are NOT required to implement for this project:

- Users having profile pictures
- Groups, or any sort of Page for interests or likes. The Profile Page's Interests section will simply be a text field where users enter their interests. This is what Facebook had in 2004!
- Chatting or instant messaging
- Any sort of extended privacy (other than what's stated above), all friends can view anything on a friends page, and post to any friends wall.
- The ability to post pictures to a wall
- "Like" functionality, or the ability to delete only certain things off of a wall. (being able to clear the whole wall is required)
- Photos and photo albums
- Timeline
- Notes
- Tagging users
- Any sort of email notification
- "In a relationship with..." functionality
- Hashtags
- The ability to comment on anything written on the wall

This is a project that really encourages creativity! Is there something in Facebook which really annoys you? Then go ahead and change it so that it works better! Don't like blue as the color? Use red! You have the freedom to create the Facebook you always wanted. Additionally extra credit is available on this project. Any of the things that you do not have to implement can be added as extra features, not to mention anything else you can think up. It is really up to you to let your technical skills and creativity shine.