COS 436 / ELE 469
Spring 2013

# Course Project Details
# +
# P1: Group Brainstorm

Due 2/22/13 at 11:59 PM

## Overview

You will form a group, brainstorm project ideas, and select an idea for your project. You will have time in lab (February 13, 15, 18) to work on this. You may have to meet outside of lab, as well.

## Project Requirements

1. You will be working in **groups of 4**. Because this is a huge class, please do not form groups of 3 unless absolutely necessary (i.e., because the number of students in your lab section is not divisible by 4), and then do so only with prior permission. You will be working with the same project group for all components of the course project, as well as for the subsequent labs (beginning next week with Lab 1: Resistance is Not Futile).

2. In your project, you must **design and build a prototype system that involves tangible, embedded, gestural, non-visual, or other off-the-desktop/not-just-mobile-or-tablet interactions.** The project assignments will lead you through the process of designing, prototyping, refining, and evaluating this interface.

Types of applications that are OK include:

- Arduino-based applications
- Kinect-based applications (** only a limited number of groups will be able to borrow Kinects, so you may have to be flexible)
- Other physical/gestural computing platforms (with prior approval)
- An application that is a hybrid of a desktop/web/mobile app, plus Arduino/Kinect/etc.

Try to think out of the box. Don't think in a device-centered way, and don't think only of things that existing applications do. Instead, focus on the things that people do every day and how tangible, gestural, embedded, or other non-standard interfaces could be used to facilitate these tasks. This approach will help you identify tasks and then the target users that perform those tasks. Or you could start with a set of target users, think of tasks they perform and applications that would facilitate those tasks. In this case, you would think of the target users first and then their tasks. Either way, you should talk to some potential users to figure out the kinds of problems they run into.

Your application must make sense as a tangible/gestural/embedded/etc. application. Don't build something that would be better off as a regular desktop or mobile app.

3. Your application **must address a real-world problem**. It cannot just be a game, though "games with a purpose" are OK. Musical or other creativity-supporting interfaces may be OK, as long as you can tell us what real-world problem you're solving (e.g., perhaps it's an instrument for someone who is paralyzed, or perhaps it's a device to help an acoustic performer practice more effectively).

4. Your application must be designed for a **specific target group of users**. College students, online shoppers, and small business owners are very large categories of users. You should be far more specific. It's better to be too specific than not specific enough. Keep in mind that you'll need to have access to people in your target group, or people who can reasonably stand in for people in your target group, so maybe it's unwise to choose "President of the USA" or "Chuck Norris" as your target group. At the same time, we're practicing designing for others, so "Computer Science Undergrads at Princeton" is not an acceptable target group, either.

5. It must be **feasible to make a working prototype of your system between now and May.** Don't be too ambitious. You have limited time to work on the project, and the goal of the course is to iterate, test and improve users' experience of your design, not to produce the most elaborate experience. Your application can be extremely simple. Make sure you're realistic about what can be done in a semester. This is an exercise in prototyping applications that could really be built, not in science fiction. Brainstorm! Give every idea a chance, no matter how strange at first.

6. It must be **feasible to create a working prototype using a reasonable budget**. We will fund a maximum of $100/group in reusable supplies (e.g., sensors, motors, other parts that are not glued or permanently soldered) + $10/group of non-reusable supplies (glued, soldered, etc.). This budget does not include the Arduino or lab kits: you may incorporate any element of your lab kit without it coming out of the budget as long as it is **reusable** after the semester. You may also incorporate sensors and actuators from the labs for "free" as long as no gluing or soldering is done.

You are welcome to buy your own parts for the project if you want to keep the project at the end of the course, but you will still have to submit a budget, adhere to the $100 + $10 spending limit, and stick to the items you specify in the budget.

## The Assignment

1. Spend at least one hour brainstorming a list of at least 50 numbered ideas (aim for more). Each idea has to be described in one full sentence (don't just list an abstract title like "Cooking application"—better: "Chef's assistant that tracks proper cooking times and helps chef maintain awareness of nutrition content.") Be visual during your brainstorm—make plenty of sketches on paper or a whiteboard. Take pictures of those sketches.

2. Choose one idea to work on and write a short explanation of why the group picked it from amongst all the possibilities in your list. This should be one short paragraph. Note: this idea may develop and change later; it's your best idea at this time. **If your chosen idea is a Kinect-based application, choose a second-best idea that does not use the Kinect** in case we don't have enough Kinects for everyone.

3. Now write a longer project description. This description should include:

- Target User Group: Your target user group should be sensible (people you have access to) and not trivial (CS undergrads is not a challenge). Describe the user group in enough detail that you can easily separate the group from other types of people. Then include details about their needs and wants. (one paragraph)
- Problem Description & Context: The problem description should be short and specific about the high-level goals of the project. The problem should be described in terms of user activities and situations where the problem occurs, and what aspects of the situation might be improved with a technical solution. Avoid describing or suggesting a solution at this stage that will hamper your design thinking when you actually start solving the problem. What aspects of the situation might influence the problem solution? Think about location, time, environmental factors, etc. Then think about aspects of the user group, their education, available time, motivation, values, etc. What related or complementary solutions exist already? (one paragraph)
- Why is the technology platform you've chosen (i.e., Arduino-based system, Kinect-based system, hybrid desktop/Arduino/whatever system) offer an appropriate solution for the problem? (one paragraph)
- Include sketches in your description of the project. Diagrams and figures will help you reduce the number of words you need to write and are likely to make your description clearer.

4. Create one blog post or page for this project assignment. **Use the "Project1" category** when you create the post. Include on this blog post:

- The **names** of everyone in your group
- A **group name** (think of a good one)
- Your brainstorming list from Part 1, with pictures of your sketches (each with a brief caption)
- Your 1 paragraph explaining your idea choice in Part 2. (If your first choice is a Kinect app, include a second paragraph explaining your second-best idea.)

- Your project description in part 3, including a separate section for each of: the target user group, the problem description and context, the justification of the technology platform. Include photos of sketches, with captions.

5. Fill out the form at the URL below to make your group official and point the TAs to your blog:
https://docs.google.com/forms/d/16NS_QFPTTAoz3JMR2P4jgZoS7yq1M1u0sdYWHVzdffU/viewform

## Grading Rubric

### Brainstorming and Idea Selection

| Criterion | Poor solution 0% | Acceptable Solution 50% | Great Solution 100% |
|---|---|---|---|
| Did you list at least 50 ideas? (5pts) | No, < 10 | No, ~30 ideas | Yes, 50+ |
| Were you visual? (2pts) | No sketches | Few sketches | Yes, sketches for > 10% of ideas |
| Did you explore a diverse set of ideas and also build on ideas? (1pt) | No build & jump | Limited build & jump | Yes, both build & jump |
| How novel and creative were the ideas? (2pts) | Most or all applications exist, are common | Some novelty, many apps already exist | Many new and unusual ideas |

### Project Description

| Criterion | Poor solution 0% | Acceptable Solution 50% | Great Solution 100% |
|---|---|---|---|
| Target Users: Did you define & describe a concrete target user community? (2pts) | Missing or ill-defined | Too broad | Yes, concrete and narrow |
| Problem description: Did you clearly describe the problem? (2pts) | No, poorly written or missing | Some problems with writing or argument | Yes |
| Why the chosen platform/technology? Did you make a clear case why the application is uniquely suited to using this type of platform? (2pts) | No, missing or app is simple copy of desktop app | Not entirely convincing | Yes |
| Novelty/creativity: How novel and creative is your final idea? (4pts) | Many apps of this type already exist | Limited novelty | Novel and creative |

**FAQ**

Q: Is ___[insert idea here]___ feasible?
A: Check out Sparkfun and Adafruit for lists of sensors and actuators if you want to know what can be sensed or actuated, and at what cost. There are all sorts of ways you might sense a user's actions and the larger world. Post to Piazza if you want some help determining what is feasible.

Q: What sorts of projects have been done in the past?
A: For inspiration, you might check out the following webpages. (Note that not all of these projects would make appropriate 436 projects, of course.)

COS 436 in 2008:
http://www.cs.princeton.edu/courses/archive/fall08/cos436/FinalProj08.html

Tangible User Interfaces course at Berkeley:
http://courses.ischool.berkeley.edu/i290-4/f08/index.html%3Fq=node%252F5.html
COS 160, User Interface Design, at Berkeley:
http://vis.berkeley.edu/courses/cs160-sp12/wiki/index.php/Main_Page
Check out their Kinect projects at
http://vis.berkeley.edu/courses/cs160-sp12/wiki/index.php/Final_Poster_Presentation
but DO NOT steal their ideas!!

Q: Why can't we just make desktop/mobile apps?
A: Although the core practices you'll be employing on this project (brainstorming, prototyping, evaluation, refinement) can be applied to interfaces on any platform, you have plenty of other opportunities to develop your desktop, web, and mobile programming skills (e.g., COS 333). In this course, you'll be developing a new skill set and hopefully be challenged to think about computing in a new way.

**Acknowledgements**
Much of this assignment is adapted from Maneesh Agrawala's CS 160 course at UC Berkeley.