**Jobs Board - Bryan Poulsen**

It has need to be redone for quite awhile. I made a version in ruby on rails, but we are not using it because it isn’t ready for our platform. 
http://jobs.dixie.red

Think it would be a good one to revisit. We have new features and specifications for it as well now.

**Code Camp Rookie kit - Curtis Larsen**

I've got an existing project, the Code Camp rookie kit that needs some significant work done. I don't know if that would meet the needs of your class or not, since it's not a from-scratch project.

The project is a client/server game written completely in Python 2. Some potential changes:

1. Server already runs on *nix. Make it also run on Windows.
2. Game already has fixed size arena. Make the arena larger, with scrolling viewport.
3. Game attack/defense are already fixed according to mathematical formula. Make a more flexible attack/defense system such that plug-n-play configurations can be submitted to change game play.
4. Create a working tournament system.
5. Create a spectator client to watch on-going games.
6. Make the game display "skins" easily shareable.

... Those are just some possibilities off the top of my head.

**Book/Resource Checkout - Rachel Ramsay**

I would like a system to check out reference books from my personal library for student use. I have a hard time keeping track of who I've lent them to and getting them back. I envision using a bar code scanner and be able to scan the book to check it in and out and send a message to a student when I need them to return it.

**Binary - Gary Cantrell**

If they can read and write binary, we've got stuff.

**Clicker System - Russ Ross**

I'd like a clicker system that uses mobile phones or other internet connected devices and integrates with Canvas.

So a student would log in at the beginning of class (this could be a way of taking attendance, too), and the instructor could pose questions to the class periodically. Students would enter a response on their phone, and the responses would be compiled and presented to the instructor. Grades would then be posted back to Canvas automatically, with some control over how the grades are calculated (maybe 50% participation, 50% getting the right answer, for example). It would also be nice to be able to have students group up and enter a group answer that then applied to each of their accounts. Different kinds of questions would be nice, such as multiple choice, fill in the blank, etc. The instructor should be able to create new questions on the fly pretty easily.

The integration with Canvas would be similar to CodeGrinder, so I'd be able to coach them on how to do that part of it.

**4H Projects**

There are always several projects brewing here.

**Build your own**
You can create a project on your own. It should be big enough that one person can not do it in one semester. But a team can. It should not be commercial. There are just too many legal complications.