

CS 330 Course Project: Understanding and Innovation: Intermittent Computing
Chien – Autumn Quarter 2016
October 25, 2016

Key Project Deadlines

(Please read this entire document carefully for assignment descriptions)

- I. Project Plan (2-3 students) Team+Platform+Idea, **due Thursday, November 3**
- II. Project “Pitch”, **in-class, November 11 (this is a friday makeup class)**
- III. Full Project Plan, and Design (idea, usage, functional description, rough SW architecture, tools, schedule, development/test plan, demo), **Tuesday, November 22**
- IV. Project Presentations and Demos, Last week of classes, **Tuesday, December 6 (during finals week)**
- V. Final Reports, **Thursday, December 8 (full writeup)**

Motivation and Focus

As you’ve learned Intermittent Computing is a broad trend, manifest in Spot instances, Peer to Peer, Serverless, and even Zero-carbon Cloud. Having learned about the technology and trends, you now have the opportunity to exercise your creativity and imagination, as well as technical skills in a course project. The challenge is to balance your creativity and insight for where all of this technology is going with what can be realistically achieved with current technology and infrastructure. And, to do all that within a 10-week quarter. Think creatively, plan carefully and cleverly, execute proactively to showcase your ideas!

Your project should explore some interesting property or implication of Intermittent Computing. It should include a couple of key elements;

- Building a working system (demo) that involves aggressive use of intermittent resources
- Pick a place where intermittence makes a BIG difference
- Understand (and systematically study) the intermittence properties that make the problem hard, how parametrically (and by distribution) these intermittence properties affect the system design, performance, overhead
- Study how far this can be pushed. How difficult a distribution? How extreme resource impulses? What are the costs/limits?

We are giving you a fair amount of freedom, so work with your teammates (and Professor Chien) to shape your project to be feasible, yet do something great!

Project Assignments Detail

- I. **Project Plan+Team+Platform+Idea**: 2-3 students
 - 3-pager, with the following content
 - Name and emails for members (and a team name!)
 - the software platform and if appropriate the hardware platform for your project
 - the idea – described succinctly in a paragraph
 - 1.5 pages outline of the plan and experiments
 - list of questions you hope to be able to answer

II. Project “Pitch”+Background, 10-minute in-class presentation,

- Purpose: Crystallize your idea, Articulate why its interesting, Why its possible
- Format: 4 slides (Team, Problem/Opportunity, Idea, Background, Why Possible), 10 minute presentation, class-wide feedback
- Detail
 - o Present the context and motivation for the project
 - o Present the idea: To focus the idea and shape possible realizations, answer critical questions about the project – what is the “big idea”? Why is it compelling? What will you learn from it? What are the hardest technical challenges? Be as specific as possible
 - o Put the work in context, what is already known? What is novel about your project? What implications will it have?
 - o Invite the audience to help you; do they have expertise, critiques that would improve the project? what questions would do you wish they could answer?
 - o These elements are the classical elements of explaining and communicating an idea in a technical setting

III. Project Plan and Design (plan, data, platforms, rough SW architecture, schedule, dev&test plan, demo sketch and plan) ,

- Purpose: Nail down feasibility, Establish responsibilities for Key Elements/Challenges/Risks, Plan their Early Resolution, Plan time for Build/Eval/Rebuild
- Format: 1 page team & contact info, 3 pages motivation and background/prior work, 1 page project risks, 3-5 pages software/system design, 2-4 pages plan, schedule, including if appropriate team assignments, 1 page planned demonstration, 1 page references, 1 page – anything else relevant
- Detail
 - o Take your idea to the next level of specificity, make the design tangible, develop a clear design (path) to realization, apply rigorous assessment to the effort and challenges for each part.
 - o What has been done in this area before? How will your work be different (a twist, different sensors, novel domain, etc.)
 - o What are the risks to failure in your project? Assumptions? Technical challenges.
 - o What is the simplest realization and demonstration to explore the research question? What are essential capabilities and technologies? In the class, what is your planned demonstration?
 - o Put together a project plan – both to explore and test the implementation approach and key assumptions. A key strategy is to eliminate all of the technical risk and usage risk as soon as possible. The plan should include opportunities for checkpoint demos (integration of partial functionality). Lay out the tools, plan and schedule, and anything else you think is important to communicate.

IV. Project Presentations and Demo

- Purpose: Showcase your project and demonstrate both that it works, but the mobile experience and interaction it enables. Demonstrations are a critical element of mobile experience, interactions, etc.
- Format: 10 minute pitch and demonstration, revised “pitch” slides, to include how it works, and the “demo” scenario – and what they should see work
- Detail: scheduled by signup

V. Final Project Writeup

- Purpose: Document your project and what you learned. Insights for Intermittent Computing, insights for your project area, and also for you about how to undertake and plan class (and perhaps longer term research) projects for maximum success and impact.
- **Format and Content:** (15-20 pages) Using the plan and presentation materials, document the project – design, execution, results, software if appropriate. Analyze your technical results, and also explore what **you** learned from doing it. Be sure to spend time to summarize your learnings from the last week or two! (often the most fruitful). For example, what did you learn about planning and executing a complex project? Working in teams? What would you do differently next time?