Social Computing Capstone

CSE 481p | Spring 2023

Amy X. Zhang
Instructor
Assistant Professor | University of Washington, Allen School of Computer Science & Engineering

Kevin Feng
TA
Ph.D. Student | University of Washington, Human-Centered Design & Engineering

Schedule for today's class

- Introduction to social computing (20 min)
- Go around and do intros! (everyone 1 min) (~30 min)
    - For those of you who didn't get to making an intro slide - add it in quickly at: tinyurl.com/SoCoSp23intros
- Overview of syllabus and course calendar (20 min)
- Socializing (if we have time at the end)

What is social computing?

It is the study of how technology shapes human social interaction (and society at large),

how humans repurpose and shape technology for social interaction,

emergent effects and coevolution from the ***combination*** of the two,

and the design of new technology that mediates human social interaction

computer science design social science

Social Computing Systems

work, collaboration, production, productivity

social support, leisure, fun, interpersonal communication

Venmo
Github
Wikipedia
Youtube
Instagram
ebay
Stack Overflow
Twitter
Twitch
TikTok
AirBnB
Google Docs
Quora
WeChat
WhatsApp
Minecraft
Amazon
Yelp
Slack
Reddit
Pokemon Go
Google Calendar
Piazza/Ed
Zoom
iMessage
WoW
Google Sheets
Canvas
email
Discord
Snapchat
Uber
Notion
blockchain
Facebook
Figma
Fortnite
cryptocurrency
Tinder
GroupMe

What is *social computing system* design?

Increasingly, we are fashioning social environments **online**.

Social computing system design asks how to fashion those environments to support the participants in achieving their goals.

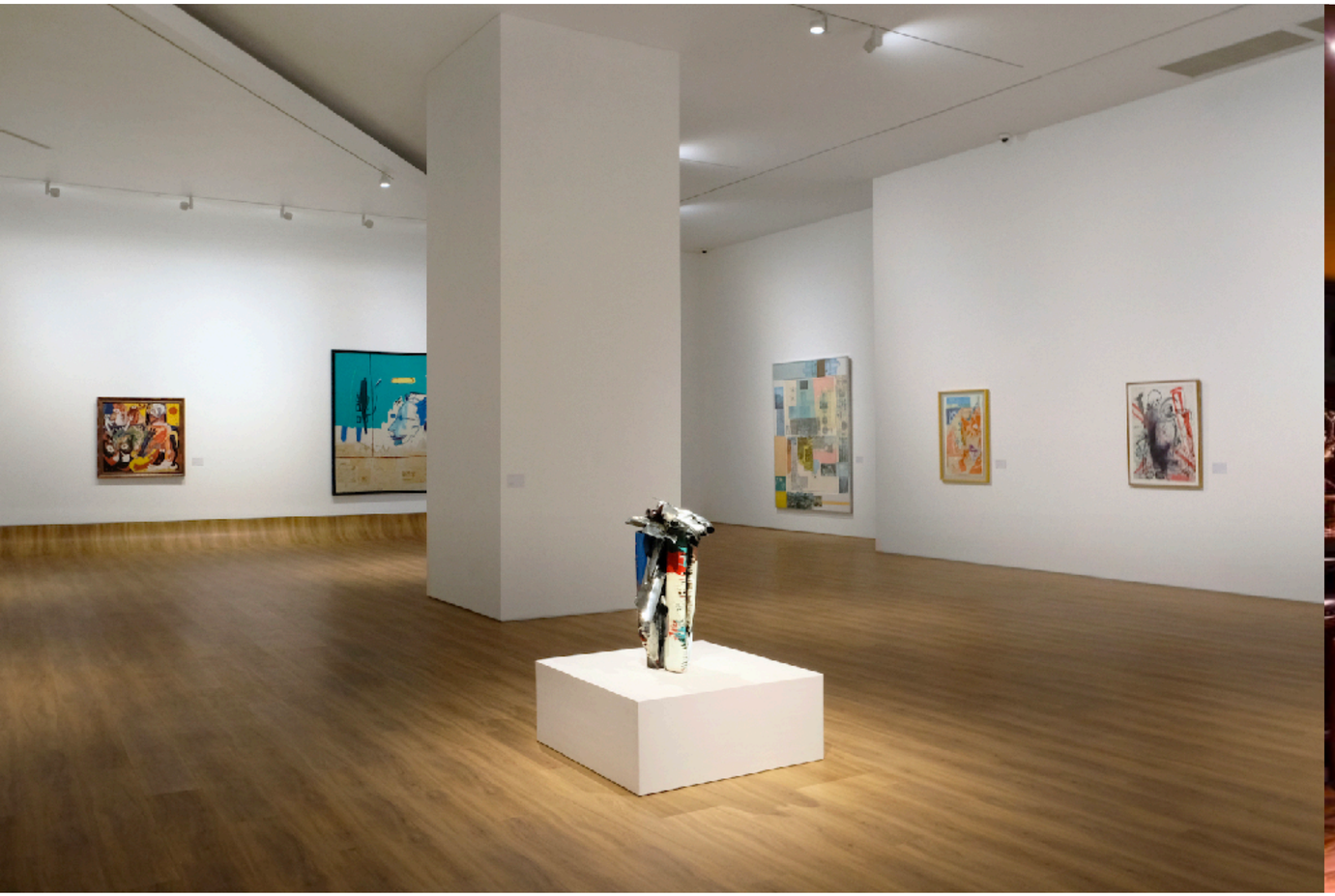
How do we cross the chasm between the social interactions that the group wants to support, and the computer interactions that we have at our disposal or could invent? [Ackerman 2000]

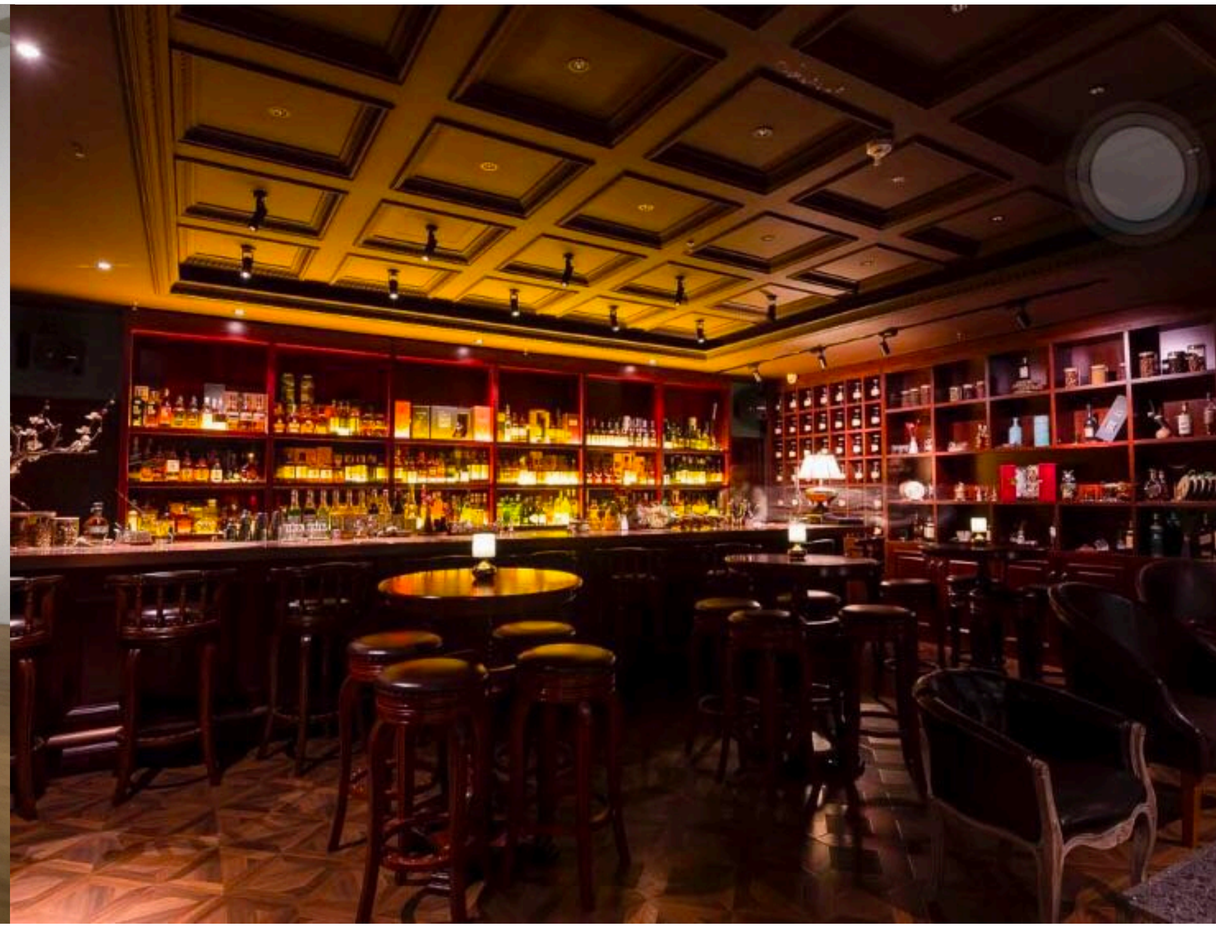
Ackerman, Mark S. "The intellectual challenge of CSCW: The gap between social requirements and technical feasibility." *Human–Computer Interaction* 15.2-3 (2000): 179-203.

computer science

design

social science





What's different about designing for online vs offline?

The offline world has laws of nature and physics that govern physical spaces.

The online worlds has technological laws—the limits of what technology makes possible—that are being rewritten all the time.

But we do bring all of ***ourselves*** online, including our physical bodies, our learned social habits, our understanding of societal norms, our emotional baggage, our prejudices and biases, etc.

Social problems => social computing problems

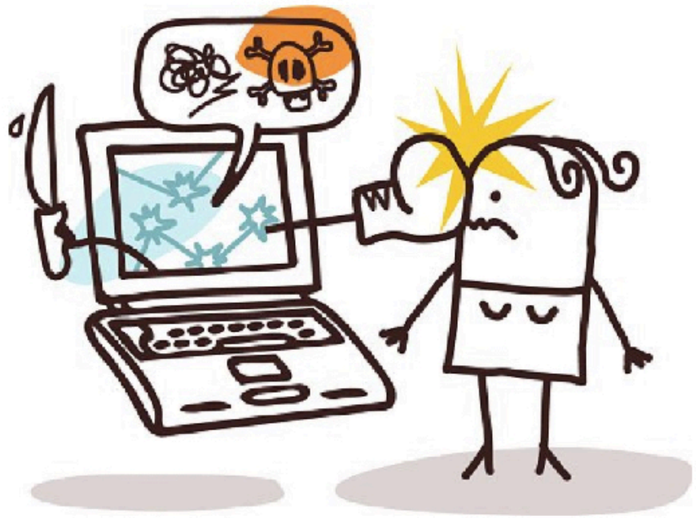
Thus, online social systems reflect all the social problems and needs we have in offline society.

1) Social media:
    a. harassment, bullying
    b. misinformation, conspiracies
    c. hate speech, trolling
    d. polarization, addiction, scams, etc.

2) Collaboration and productivity tools:
How to not curl into an anxious ball getting through your email, missed Slack messages, Zoom meetings, browser tabs, etc.

(+) How to...make friends, learn new skills, fall in love, find employment, get support, play together, make things together, etc.

Social problems => social computing problems



Are we doomed to replicate all our offline societal problems?

As designers and builders of social computing systems where so much of the “architecture” of the space is up to us, can we imagine a better future than this?

Maybe! Perhaps through good design, we can encourage certain behaviors and discourage others for pro-social outcomes?

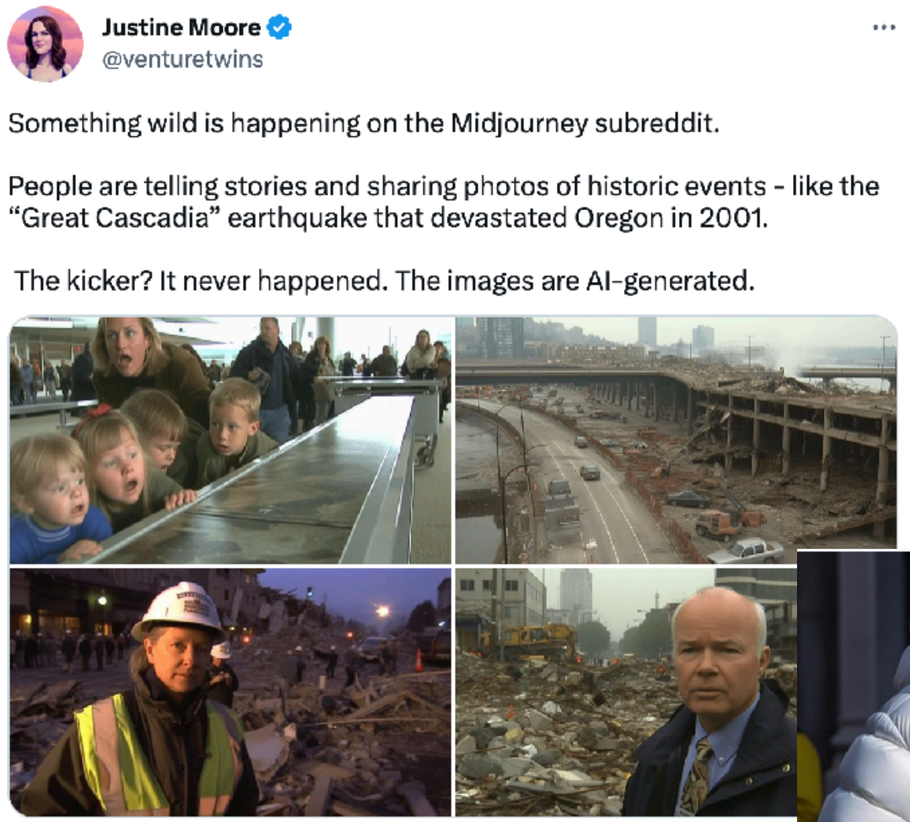
So are we’re going to solve all of society’s ills online? And create a digital utopia? Sadly, no.

beware of *technosolutionism*

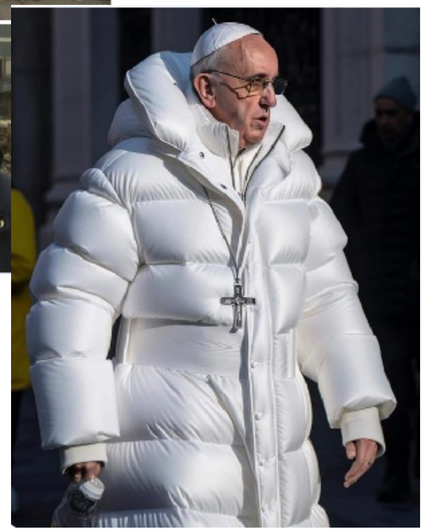
There is not an easy fix to big complex social problems through technology alone.

Social media platforms could be designed to be better for society... but their problems are also often symptoms of other deeper issues that changing the tech cannot solve.

Social computing problems => social problems



Examples: AI "deepfakes" and generative images



With new technology, we may introduce NEW problems that don't happen offline. We can also exacerbate existing social problems and make them WORSE online.

Sometimes the creators of new systems that create problems are acting nefariously. Oftentimes, new technology has both good and bad uses. Other times, problems are unanticipated and emergent.

Why is social computing design hard?

The Daily Dot

Culture

The demise of a social media platform: Tracking LiveJournal's decline

Aja Romano—

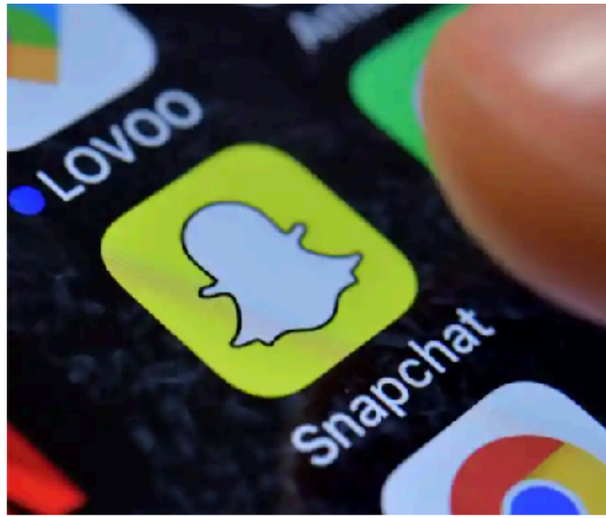
VANITY FAIR

HIVE

From the Magazine

"MEN ARE SCUM": INSIDE FACEBOOK'S WAR ON HATE SPEECH

The company blew it on privacy and fake news. Can it do better against trolls and racists? An



Snapchat

Snapchat update: mo 800,000 angry users s petition to change rec

sociotechnical:

the holistic, interconnected contribution of technology and the social systems that operate and interact with it

All social computing systems are sociotechnical. You cannot divorce the two.

Why is social computing design hard?

The sum is not just the addition of the parts.

Never just transfer social features from one app into another. It’s not just about whether you have points, or friend/follow models, or real names or pseudonyms.

The same code can produce very different social outcomes.

social norms:

The unwritten rules of beliefs, attitudes, and behaviors that are considered acceptable in a particular social group or culture. Social norms are the accepted standards of behavior of social groups.

As social beings online, our behavior is regulated both by social norms and by technology.

Good design takes into account *both*. You can write code to influence behavior…up to a point. You can also do things to establish norms that then influence behavior.

Why is social computing design hard?

How do you design a social computing systems that helps promote the behaviors that the group wants to see in the system?

What about a design makes people...
Feel safe?
Post funny memes?
Engage in thoughtful discussion?

Why is social computing design a serious responsibility?

These systems present an opportunity for us to imagine a more {thoughtful, deliberative, fun, emotionally connected, empathic, just} society. However, they can also have the opposite effect.

What power do you have as a creator, and what responsibility do you have when creating? How do we draw on positive opportunities without unleashing Pandora's Box?

Your job in this class is to put yourself into that role of a designer of a social space to make something new. Get used to making tough tradeoffs and clarifying your values. Our technology is not neutral [Winner 1980].

Winner, Langdon. "Do artifacts have politics?." *Daedalus* (1980): 121-136.

Intros!

Let’s have a ~3 minute bathroom break.

For those of you who didn’t get to making an intro slide - add it in quickly at: tinyurl.com/SoCoSp23intros

Course Overview

This class is a capstone. What does that mean?

We'll spend a little bit of time every week talking about social computing topics. But I'm not going to lecture at you much.

Instead, this class is going to be oriented around *doing*. The main work involved in this class is a major capstone project where you'll work in groups to apply the skills you've gained in prior courses towards designing and implementing a social computing system!

Along the way, you'll learn and apply some software engineering, project management, UX research and design, and presentation skills.

By the end of this class, you will...

- understand the major issues and societal debates facing social computing systems today and the state of the art in how to address them understand
- what are important considerations for designing a social computing system
- be able to critically evaluate social computing systems you encounter in the world

- know how to use an iterative and human-centered design workflow to design and build a complex user-facing social system
- understand and be able to select and apply appropriate user research and testing methods
- apply the computer science skills you have gained in other courses towards engineering a complete project, including code and design specifications and feasibility analysis
- practice working with other students in a team and gain strategies for successful teamwork and project management

We expect from you…

(This is me influencing the social norms of this class!)

Your involvement in creating an engaged and positive class environment. That includes your active and enthusiastic participation in class discussions, presentations, and feedback. And it includes assuming good intentions to foster a welcoming space during discussion and sharing.

Your equal contribution to your team’s workload, openness to communication, and willingness to be a team player. We will have regular check-ins with your team to set team expectations.

Timely communication with the staff on any issues that may crop up, absences, illnesses, and other extenuating circumstances. We’re here to help you, and we will work with you and your situation, whatever it may be.

Policy on missed class

In-person attendance and participation is mandatory and part of your grade (10%).

By default, this course will not be recorded, since much of it is discussion based and group based. We understand you may get sick or have something that is difficult to miss. Please email us regarding any excused absences.

Those who miss class with an excused absence are still expected to make up whatever they missed. For instance, if there was group work happening, you should make up for this by doing extra work for your group outside of class. I will share any slides from class on the website to help you make it up. Remember though that lecture is only a tiny part of the class.

Course Structure

Meetings: TTh, 11:30AM - 12:50PM Parrington Hall 220
Class will involve a mix of lecture, discussion of readings and assignments, class activities, guest speakers, group work time, group presentations.

No exam or tests of any kind.

5 individual assignments (mostly short reflections). Except for A1, you will post these on Ed and read and comment on each other's posts.

1-2 readings per class (mostly news articles/opinion pieces). We will read them and collaboratively add comments in the margins using Hypothes.is, and then discuss them in class.

9 group assignments related to the capstone project, going from idea all the way to a working digital prototype, launch website, and video demo!

Social Computing Systems used in this class

Course website: https://courses.cs.washington.edu/courses/cse481p/23sp/

Ed - where you'll post reflections and where I'll post announcements

hypothes.is - for comments on readings in the margins. We'll install this together in a future class (browser plugin)

Github - where your group will collaborate on code and you'll keep track of deliverables

Canvas - only for submitting links to group submissions and receiving grades. All individual/group/reading assignments with deadlines are also there to help remind you.

Google Drive - where most of your group submissions will actually be stored (we will create group folders for each group)

Grading

10% - Attendance and participation

10% - Annotate readings before class with comments
  12 days with readings, 1 point for each, ignoring up to 2 missing ones

13% - 5 individual reflection assignments
  3%, 2.5%, 2.5%, 2.5%, 2.5%

67% - 9 group assignments for the capstone
- 5%: user research report
- 3%: pitch presentation
- 9%: low fidelity prototype
- 10%: code and design specification
- 3%: midterm presentation
- 9%: user testing
- 18%: digital prototype
- 7%: video demo
- 3%: website

For each group assignment you will also turn in an individual contribution statement.

Capstone Project

Groups of 3-4 (preferably 4): we will help facilitate group formation

Your goal: design, build, launch, and test an idea for the design of a social computing system

This can be a whole new way of doing a social activity (building a system for performing a social activity from the ground up) or a change to an existing social system (Slack but with a bot that pairs up team members to help them get to know each other).

You can choose to use any (user-facing) technology you've learned in a prior class - web dev, app dev, machine learning, NLP, vision, visualization, graphics, VR/AR, haptics. You can stretch yourself a little here but do NOT take on a technology you know nothing about, or you will be in a lot of pain later in the class when implementing...

Social Computing topics

Still somewhat in flux, but broadly broken up into three sections

Concepts and Overview
Today's intro
Landscape of social computing systems
- Why do people use them?
- What's broken about them?
How to prototype social computing systems

Issues of Social Computing Systems
Engagement
Online Social Identity
Quantification
Monetization
Ethics

Social Computing Systems Today and Tomorrow
Commercial Content Moderation
Community Norms
"Democratization" and Crowdsourcing
Being On All the Time
Public Interest Technology
Algorithmic Feed Curation

For Thursday:

Assignment A1: Pitch 3 project ideas will be due.

Assignment spec is on Canvas and on the website.

Think about your goals for this class and the topics within social computing that interest you to come up with three project ideas. After coming up with three project ideas, create a Google slides presentation to present your three project ideas (submit this link on Canvas by 11:00 AM Thurs).

We will be placing all the slides into one big slideshow, and on Thursday, you will each pitch your project ideas to the class for 1 minute.

Remember that the main purpose of this assignment is to **help you find group members that are interested in the same area**. So don't worry if your ideas are still fuzzy. Think about topics that you are excited about and pitch to your classmates to get them excited about it as well!

Socializing Time

To help facilitate you all forming groups, we're going to do a social icebreaking activity.

Find another person or a group of 3. Go around, introduce yourselves, and tell each other **your first internet memory**. Once you all are done, if there's time left over, break up your group and join another one, and do it again!