
A Quick Tour of HCI

SWE233: Intelligent User Interfaces

<https://dayenam.com/teaching/swe233-fall2025/>

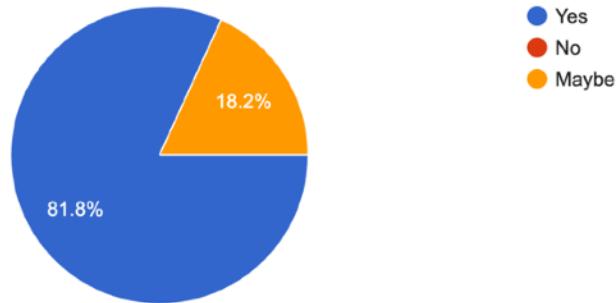
Daye Nam
Fall 2025

Part of this lecture adapted materials from CMU's 05-863 (Fall 2018) Lecture 1

Intro Survey

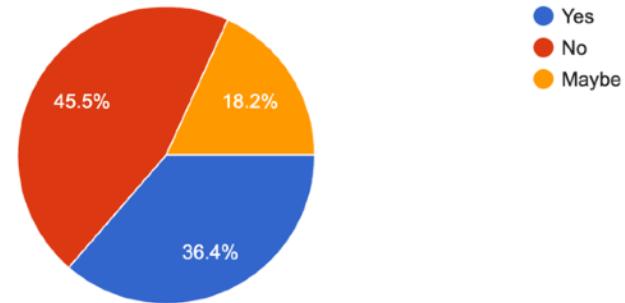
Have you ever taken an AI/ML/NLP course?

11 responses



Have you ever taken an HCI/empirical methods course?

11 responses



Will have 3 overview lectures on HCI/empirical methods/evaluation.
Will read and discuss many new AI papers!

Course infrastructures and logistics

Infrastructure/source of truth

Course website: schedule, slides, syllabus, office hours

<https://dayenam.com/teaching/swe233-fall2025/>

Canvas: homework, grades, other material

Slack: communication and collaboration



Paper Presentation & Discussion

We will read and discuss systems with intelligent user interfaces, mainly within the software engineering domain

Presentaion

Each student will present at least one paper ~~one~~ **TWO PAPERS** over the quarter. As presenters, you will do a presentation of the paper and connect the paper to many aspects we cover in the first part of the quarter.

Paper selection

You will bid for a paper you want to present in the next class.
Bid for papers that are close to your potential project topic.

Learning Goals

After today's class, you should be able to

Understand the importance and need for HCI

Understand the challenges in designing good UIs

Identify and describe UI/UX issues

Why Do We Care About HCI?

Human Computer Interaction

Is to make computers that are *useful and usable and effective* for users

Accomplishes the goal by designing and building better interaction

Draws on *computer science*, psychology, design, social sciences to understand user needs and behaviors

Why Do We Care About Users?

With user-centered design, we can

Help novices become more effective quicker

Make experts more efficient

Reduce errors

Productivity and satisfaction

Increase sales / reduce cost for customer support, ...

Removing a step during eCommerce purchase increased sales by 45%

= \$300,000,000 / year

Reduce redesign costs

Competitive Differentiation

Apple Products - iPhone & iPad



<https://www.macworld.com/article/186335/original-iphone-review-2.html>



<https://www.wired.com/story/apple-ipad-turns-10/>

Uber / Lyft

Uber Ride Drive Business Uber Eats About ▾

EN Help

Ride

Request a ride

Pickup location

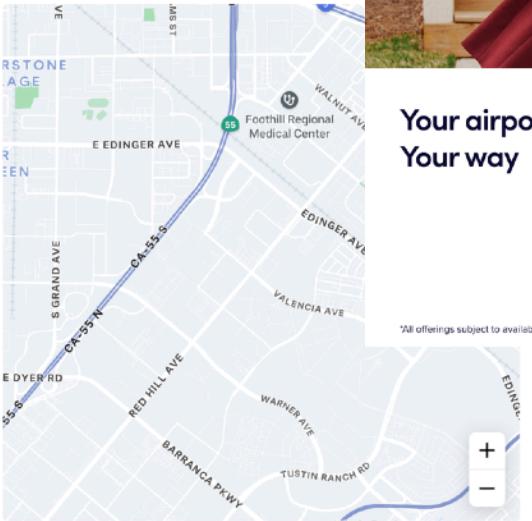
Dropoff location

Date Time

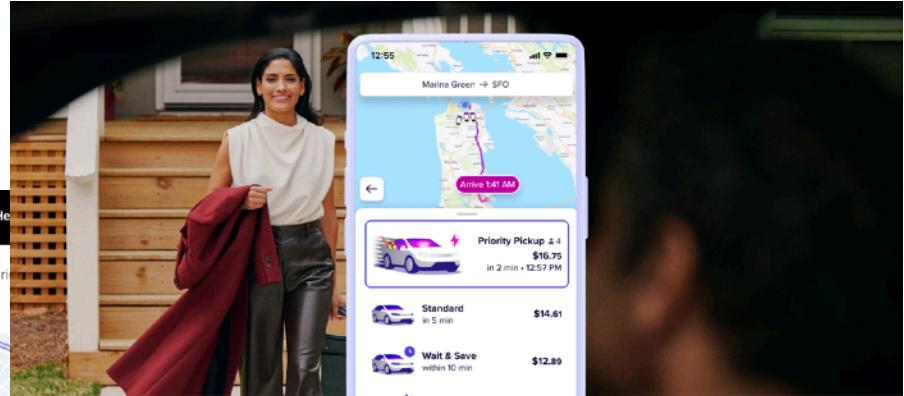
Destination suggestions

- Marriott
18000 Von Karman Ave, Irvine, CA
- Renaissance Hotels
4500 Macarthur Blvd, Newport Beach, CA

[See prices](#)



A detailed map of a city street grid. A blue line indicates the route from 'Marina Green' to 'SFO'. Street names visible include E EDINGER AVE, S GRAND AVE, CA-55 S, RED HILL AVE, BARRANCA PKWY, TUSTIN RANCH RD, VALENCIA AVE, and WALNUT AVE. A green circle marks the pickup point at 'Marina Green'. A small note at the bottom right of the map says: 'All offerings subject to availability.'



RIDE SOONER WITH PRIORITY PICKUP*

Choose our fastest pickup option when you need to travel as quickly as possible.

[Start your journey.](#)

<https://www.lyft.com/>

<https://www.uber.com/>

GitHub Copilot

General Availability

GitHub Copilot for Xcode code completion

The screenshot shows the Xcode interface with the GitHub Copilot feature active. On the left, the Project Navigator displays a project structure for 'DemoApp' with files like 'ContentView.swift', 'DemoAppTests.swift', and 'DemoAppUITests.swift'. In the center, the Editor pane shows a portion of 'ContentView.swift':

```
1 import SwiftUI
2
3 struct ContentView: View {
4     @State private var isRed: Bool = false
5
6     var body: some View {
7         VStack {
8             Image(systemName: "globe")
9                 .imageScale(.large)
10                .foregroundColor(isRed ? .red : .blue)
11
12            Text("Hello, World!")
13            Toggle("Red Hold ⌂ for full suggestion")
14            .padding()
15
16            Button("Tap Me") {
17                ...
18            }
19        }
20    }
21}
```

A tooltip from GitHub Copilot appears over the 'Toggle' line, reading: 'Red Hold ⌂ for full suggestion'. The GitHub Copilot logo is visible in the top right corner of the Xcode window.

<https://github.blog/changelog/2025-02-14-code-completion-in-github-copilot-for-xcode-is-now-generally-available/>

Ford

Ford recalls SUVs because drivers are accidentally turning them off

By Chris Isidore @CNNMoney January 6, 2015: 7:54 AM ET

 Recommend 118



<http://money.cnn.com/2015/01/06/autos/ford-push-button-ignition-recall/index.html>

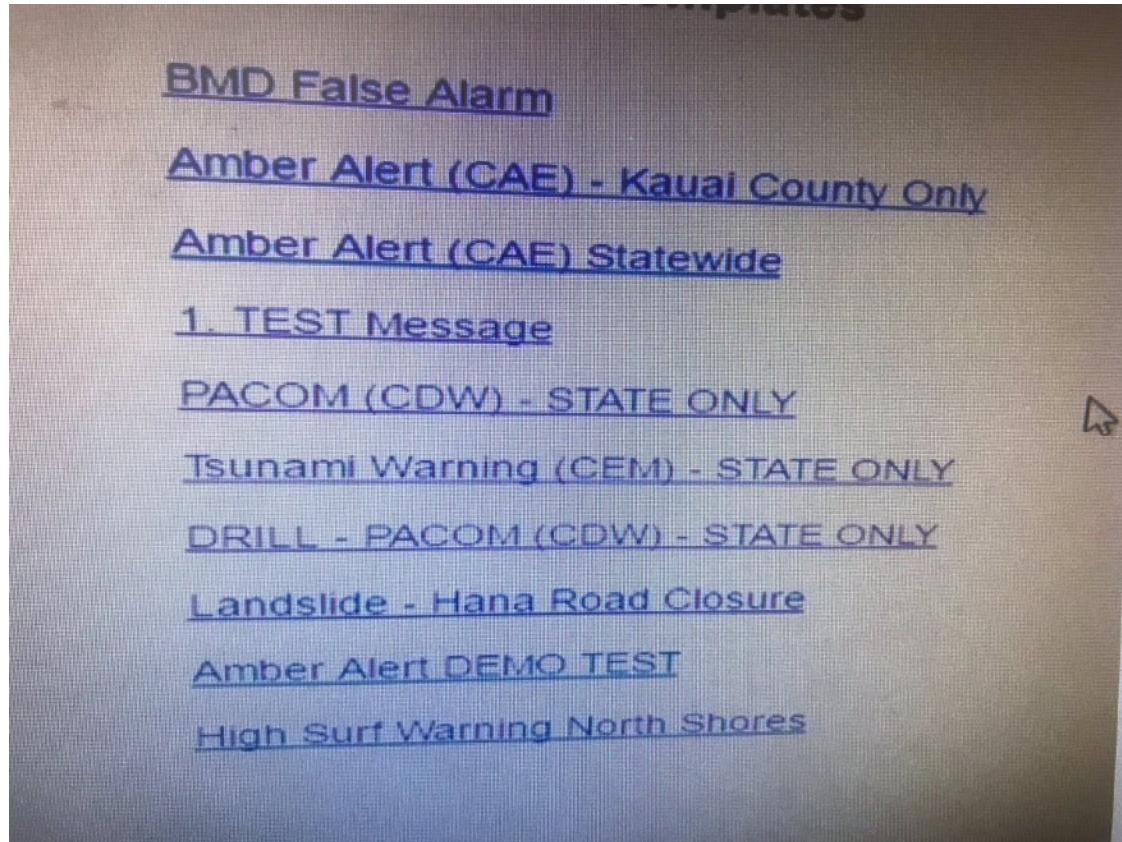
The start-stop button is nestled at the bottom of the gear selector column.

Florida Ballots in 2000

Confusion over Palm Beach County ballot



Hawaii Missile Threat False Alarm in 2018



Activity

As a group of 2-3,

1. Pick 1 product that you had UI/UX issues
2. Describe the issues
3. Think about how you would fix them

Why Do WE Need to Learn HCI?

Great user experience only comes about through constant diligence and attention

There are well-defined methods and techniques

- Not just opinions, luck, domain-experience

- HCI-trained people build better interfaces

- Exposure to different kinds of interfaces, problems

- User model, not system model

- Guidelines

Why are UIs Difficult to Design?

What is the “User Interface”?

Everything the user encounters

Functionality & Usefulness

Content

Presentation

Layout

Navigation

Speed of response

Emotional Impact

Documentation & Help

All Influences the Usability

Learnability

Efficiency

Memorability

User errors

Satisfaction

...

User Experience (UX)

More than usability

Emotion, Heritage

Fun, Style, Art

Branding, Reputation

Political, social personal connections

Beyond just the product itself – “Service Design”

Why Hard to Design UIs?

No silver bullet

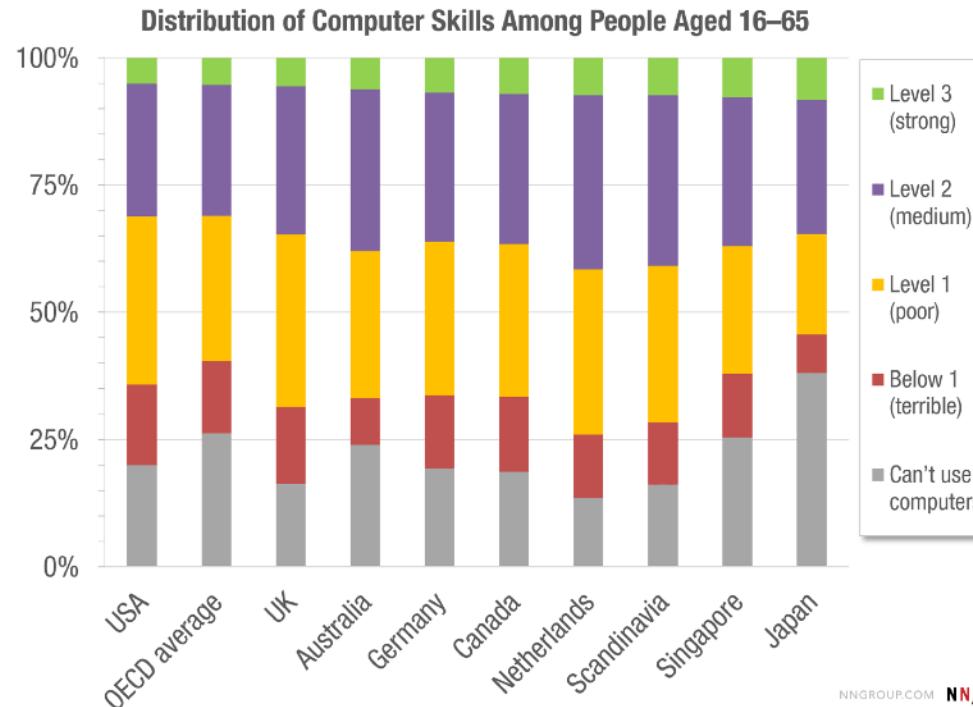
Seems easy, common sense, but seldom done right

Once done right, however, seems “obvious”

User Interface design is a creative process

Designers (you) ≠ Users

Designers (you) ≠ Users



<https://www.nngroup.com/articles/computer-skill-levels/> (Nov 13, 2016)

Why Hard to Design UIs?

Designers have difficulty thinking like users

Often need to understand task domain

Can't "unlearn" something

Specifications are always wrong:

"Only slightly more than 30% of the code developed in application software development ever gets used as intended by end-users."

Hugh Beyer and Karen Holtzblatt, "Contextual Design: A Customer-Centric Approach to Systems Design,"
ACM Interactions, Sep+Oct, 1997, iv.5, p. 62.

Need for prototyping and iteration



A
BIRD
IN THE
THE BUSH

Why Hard to Design UIs?

All design/development involves trade-offs

- Add features

- Test/fix bugs

- Test/fix usability

- Cost

- Time-to-ship

- ...

Why Do WE Need to Learn HCI?

Great user experience only comes about through constant diligence and attention

There are well-defined methods and techniques

Not just opinions, luck, domain-experience

HCI-trained people build better interfaces

Exposure to different kinds of interfaces, problems

User model, not system model

Guidelines

THIS DESIGN IS BASED ON
THE USER RESEARCH
I MADE.

WHAT KIND OF RESEARCH
DID YOU CONDUCT?

USABILITY TESTING.

HOW MANY PEOPLE
WERE TESTED?

ONE

WHO WAS IT?

A COWORKER...

WAS THIS
COWORKER YOU?

YES.



Usability Methods

Contextual Inquiry
Contextual Analysis (Design)
Paper prototypes
Think-aloud protocols
Heuristic Evaluation
Affinity diagrams (WAAD)
Personas
Wizard of Oz
Task analysis
Cognitive Walkthrough
KLM and GOMS (CogTool)
Video prototyping
Body storming
Expert interviews
Information Architecture Diagrams

A/B studies
Questionnaires
Surveys
Storyboards
Journey maps
Prioritization Matrices
Log analysis
Focus groups
Card sorting
Diary studies
Improvisation
Use cases
Scenarios
Cognitive Dimensions
“Speed Dating”
Journey Maps
...

Administrivia

Problem Identification (due: Oct 8)

Submit a short abstract (300 words) that covers the following:

Problem Definition

Target Users

Intelligent Solutions

Please talk to me if you need help!

ABSTRACT

Code-generating large language models map natural language to code. However, only a small portion of the infinite space of naturalistic utterances is effective at guiding code generation. For non-expert end-user programmers, learning this is the challenge of *abstraction matching*. We examine this challenge in the specific context of data analysis in spreadsheets, in a system that maps the user's natural language query to Python code using the Codex generator, executes the code, and shows the result. We propose *grounded abstraction matching*, which bridges the abstraction gap by translating the code back into a systematic and predictable naturalistic utterance. In a between-subjects, think-aloud study ($n=24$), we compare grounded abstraction matching to an ungrounded alternative based on previously established query framing principles. We find that the grounded approach improves end-users' understanding of the scope and capabilities of the code-generating model, and the kind of language needed to use it effectively.

Paper Presentation Bidding (due: Oct 2 1 am)

Submit your favorite 5 paper titles in order of preference