Lecture-12: HCI processes in a nutshell

CS698Y: HAI

Logistics

- Homework up, as announced earlier, due on 18th October.
- Immediately following, next assignment will involve explanations on the UI and a user evaluation.
- Mid-sem copies available to see on Friday during the cancelled class time – 1030-1130, KD102.
- No final project – the continued homework treated as final project.

Final grading scheme

- In-class assignments: 10%
- Quizzes: 15%
  - 2 more coming your way – one next Monday on explainability techniques.
- Homework assignments: ~~20%~~ 10%
- Mid-semester exam: 25%
- Final project: ~~30%~~ 40%--the continuing assignment
  - Will include 4 milestones in total + a final presentation.

So far…

- HAX principles
- FARE
- Explainability & transparency
- Going forward…
  - Need finding
  - Evaluating AI user interfaces
  - Tied to a model process

How is software built (SDLC)?

- Planning & Analysis / requirements gathering
- Design → UI/UX + software architecture & design
- Development
- Testing
- Deployment
- Maintenance, rinse & repeat

What about ML/AI systems?

- How do we gather requirements / analyze in routine systems?
- How do we do that for ML/AI systems?

- Analyze business processes
- Take to stakeholders for needs
- See if it fits one of these categories

(Aka, can AI solve it?)

Detection

Identification

Estimation

Forecast / Predict

Compare

Discover

Generate / Answer

Act

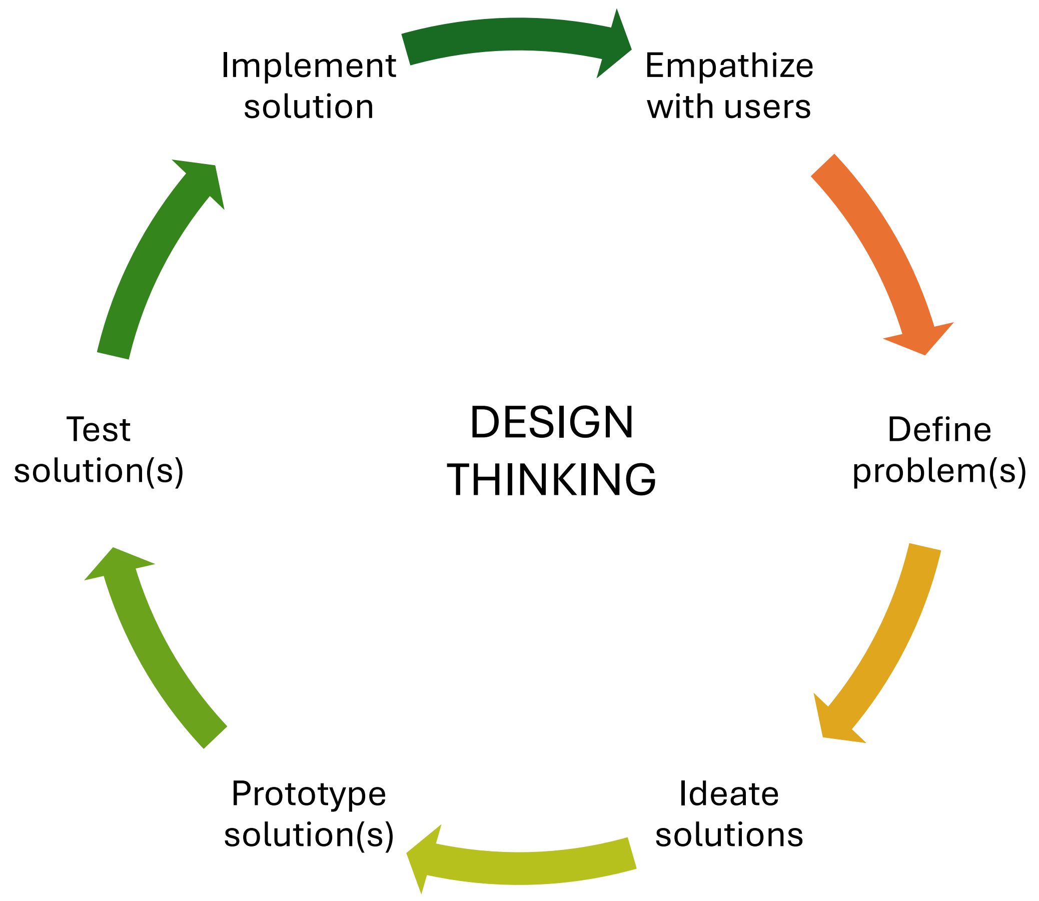
Analysis

- Data availability – incl. regulations on data
- Who is responsible for decisions (how much automation?)
- Who will use the system?
    - What are their needs?
    - Ask in terms of business, rather than tech!

Design

- Think in terms of overall solution, not just the AI bit
- How to present results, how it will work with other systems, where data comes from, live data vs. not, etc.
- Confirm with users / customers – validation (happens all the time in SE)
- Catch: People are not yet very familiar with what ML systems can do, and we don’t yet know what all works or not.

Designing right!



Empathize with users

- Methods: interviews, observations
- Understand processes, workflows and pain points
- Hard for users to specify requirements, since people don’t understand AI capabilities widely, or haven’t used them yet.
- Rule of thumb: seek to understand users’ job, roles, processes, do not think of what to build, yet.

Problem definition

- Look for inefficiencies, errors, in the processes.
- Rule of thumb: At this stage, think about what to help out with, list all of them [not just AI].
- Then see where AI can help, and where other kinds of systems together can.
  - Look for the CMU's list of 10 AI opportunities
  - See if an ML solution can be built when such opportunities arise
  - Think about roles of users, do not let them feel disenfranchised
  - Who is responsible for the system's decisions/suggestions?
  - Consider data availability

Ideation

- Generating ideas for solutions (often more than one)
- In the case of AI-based solutions, consider the following:
  - What kind of model(s)
  - Where the data will come from, privacy, etc.
  - What kind of UI / who will use it, when, how
- Also consider non-AI based solutions, and evaluate which way to go, and for what.
- At this point, we have a first set of requirements
  - Need to validate them with user(s) / stakeholder(s), before building

Prototyping

- When feasibility is unclear
- When you want to communicate to user(s) what the system will do (and they can’t imagine yet, otherwise)
    - Similar to pictures say a 1000 words!

Not a full system, though → costly, especially when we don’t know yet if that is what people want.

Prototypes capture salient aspects of a system for obtaining feedback

Helps think through details (e.g., fitting in overall workflow, estimating time/effort/costs, integrating with existing systems, etc.)

How do we prototype?

- Prototype entire system, not just model
- Think about how it integrates with existing processes/systems, not in silo.
- In prototyping ML/AI systems, consider:
  1. Availability of data
     1. Synthetic data
     2. Data from another similar case
     3. Old data, small subset
  2. Model prototype
     1. Build a toy model
     2. Wizard of Oz
  3. UI prototype
     1. Paper prototypes
     2. Sketch screens
     3. Wizard of oz, with real-like interactive screens (e.g., chat window)

Evaluating prototypes

- Have we built the right thing? → requirements / process understanding is right?
  - [equivalent to requirements validation in SE]
  - Basically, Is it useful?
  - How? Go back to customers → show prototype → tell them an alternative way of doing their tasks (in context of overall process) → get feedback.
  - Always requires customers / users!
- Have we built the thing right?
  - Basically, is it usable
  - How? → with & without users

Usability evaluation

- Next class.