

Human-Computer Interaction

CSE566 • Week 1

Foundations, people and design

Mehmet Göktürk

Department of Computer Engineering



The purpose of a system is the human activity it enables.

Communication, learning, work and enjoyment provide reasons to build technology. Processing power and feature counts describe the equipment. The human outcome determines whether that capability is useful.

A longstanding design problem



People must operate equipment in real environments.

Human-machine interaction spans industrial controls, transport, household appliances and computing. New technologies change the available actions, but people still need to learn, operate and recover from difficulties.

Hürriyet Anasayfa > Tüm Gündem Haberleri

Önceki Haber | Sonraki Haber

Dev kazanda korkunç ölüm

Paylaş B S f t Arşivime Ekle Yazdır

Ömer ERDEM/İSTANBUL, (DHA)

ATAŞEHİR'de bir geri dönüşüm fabrikasında işçi olarak çalışan 22 yaşındaki İbrahim Bektaş, temizlik yaptığı plastik geri dönüşüm kazanının aniden çalışması ile feci şekilde can verdi.

Olay saat 15.00 sıralarında Ataşehir Ferhatpaşa Mahallesi Akdeniz Caddesi 61. Sokakta bulunan Hür Kağıtçılık firmasında meydana geldi. Eski plastik parçaların hammaddeye dönüştürüldüğü kazanda temizlik yapan İbrahim Bektaş'ı fark etmeyen bir çalışan, çalıştırma butonuna bastı. Dev bıçakların çalışması ile İbrahim Bektaş feci şekilde can verdi.

Olay yerine gelen itfaiye ekiplerinin çalışması ile parçalar halinde çıkan İbrahim Bektaş'ın cesedi, Adli Tıp Kurumu Morguna kaldırıldı. Olayla ilgili operasyon başlatan polis, bir ihmal olup olmadığını değerlendirirken, İbrahim Bektaş'ın çalışma arkadaşları ifade için polis merkezine götürüldü.

30 Eylül 2009



Safe interaction includes cleaning and maintenance.

The worker was cleaning inside an industrial machine when it started. How should controls, interlocks and maintenance procedures prevent operation while someone is inside? The design must account for the whole task, including cleaning and servicing.

Air Inter 148: similar displays, different meanings



-3.3° is a descent angle. -33 represented a descent rate of 3,300 ft/min.

The same display and controls served two modes: flight path angle (FPA) and vertical speed (V/S). In the Air Inter accident, investigators considered inadvertent selection of V/S highly probable. A small visual difference represented a very different aircraft command.

FAA Lessons Learned: Air Inter 148. Original FPA / V/S displays.

The redesign: make the difference visible



**After the accident, -33 became
-3300 in vertical-speed mode.**

Two additional zeros made the vertical-speed value more distinct from the flight-path angle. The display now exposed the full rate in feet per minute. Design lesson: when modes change the meaning of an input, make that difference easy to see.

FAA Lessons Learned: modified Flight Control Unit after the accident.

Simple physical arrangements can help



Control arrangement can influence how a person operates equipment.

Look at the blade area, guard and separated controls. Requiring both hands at designated controls can keep them away from the cutting zone. Physical arrangement and interlocks can make safe operation part of the action itself.



More capable equipment can create more demands on its operators.

People have limited attention, memory and physical reach. Training and external aids can improve performance. Design should account for these capabilities instead of assuming that increased machine capability automatically improves task performance.

More functions, more interaction decisions



Radio controls

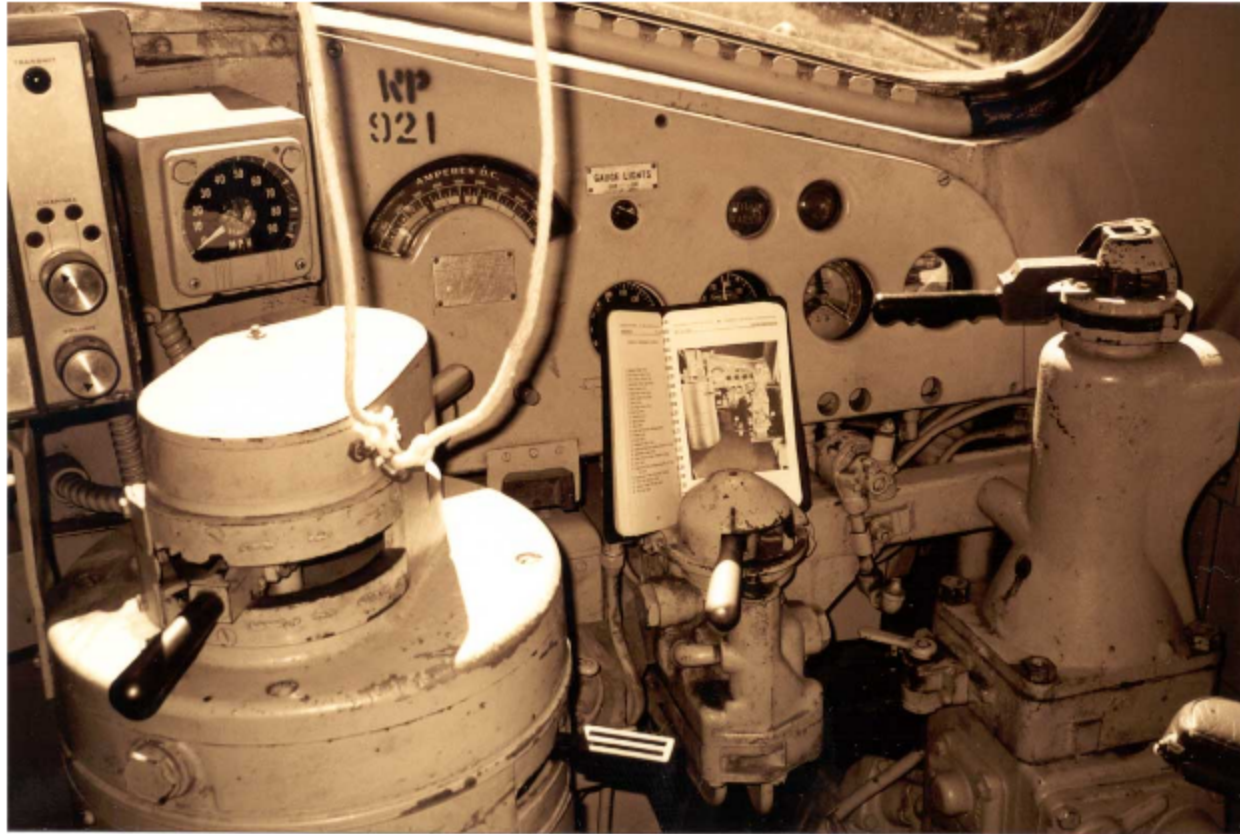


Radio and digital functions

Additional functions change what the user must identify and select.

Compare visible choices, labels and grouping for a particular task, such as changing a station while attention is divided. Added capability and ease of operation are separate design questions.

Beyond the technical threshold



A working mechanism is only part of a working system.

Engineering establishes what the equipment can do. Human-centered design examines whether people can use that capability reliably in the intended setting. Interface decisions belong in requirements and evaluation as well as implementation.

Ergonomics questions

- Reach, posture and force
- Workload and conditions
- Organization of work

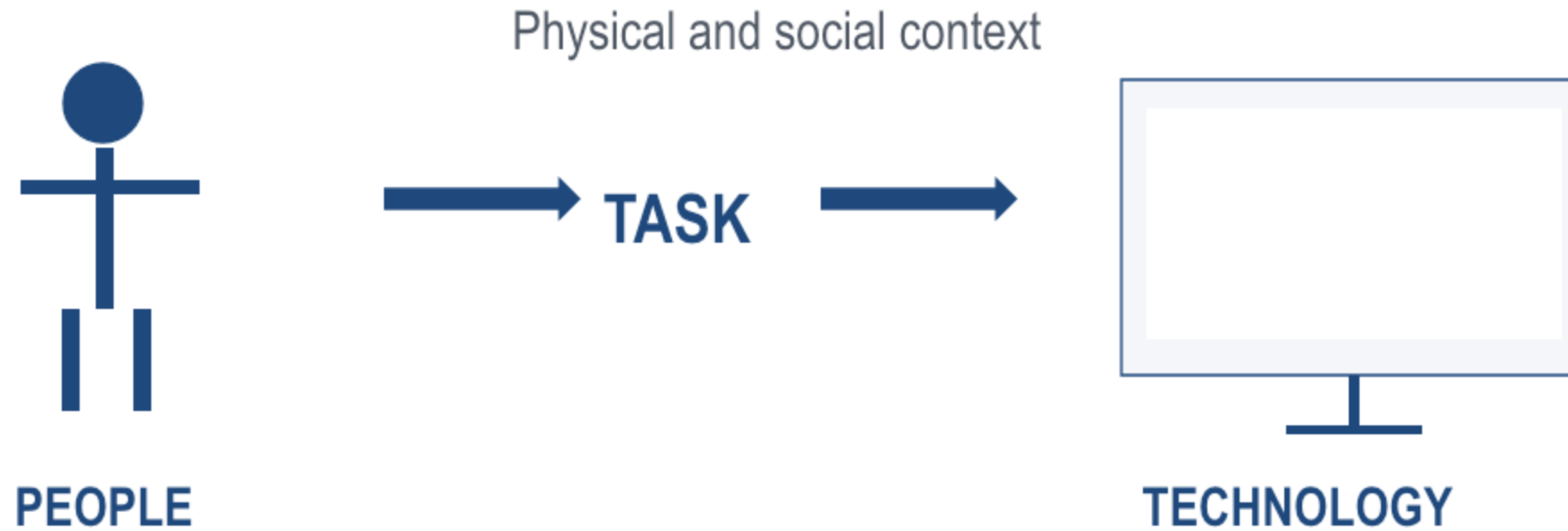
HCI questions

- Actions and information
- Interaction and feedback
- Use and evaluation

Both study the fit between people, activities and designed systems.

Ergonomics includes physical, cognitive and organizational aspects of work. HCI focuses on interactive computing systems. A touchscreen appliance, vehicle or workplace can require both perspectives.

HCI connects people, tasks and technology



HCI studies interactive computing systems for human use.

Design, implementation and evaluation must account for people's capabilities and experience, the task, and the physical and social setting.

More capability changes the interaction



Mechanical controls



Electronic programs



Phone-mediated interaction

The task may stay familiar while controls, modes and dependencies multiply.

Useful questions include: What must the user choose? What state is visible? Does a second device help, or introduce another dependency?

Familiar task, unfamiliar labels



Recognizing an appliance is not the same as understanding its controls.

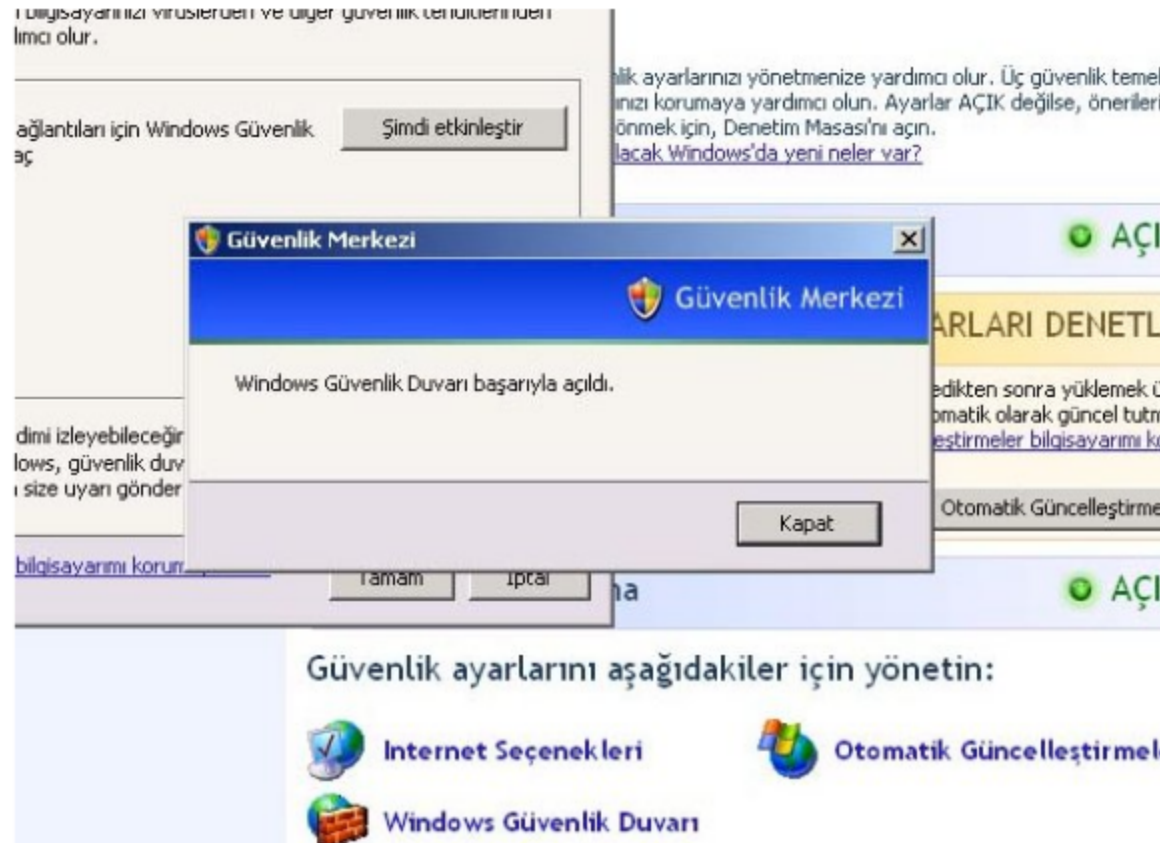
Language, conventions and prior experience shape interpretation. An icon may help, but its meaning also has to be learned. Evaluate the interface with the people who will use it.



The operator serves a wider task and other people.

In a service setting, the interface supports a conversation, records and decisions. The system must fit the employee's work and the customer's needs. Screen layout alone does not describe the complete interaction.

The cost of confusing interaction



Messages can interrupt work without making the next action clear.

This historical Windows example provides several prompts and settings. Identify the user's immediate goal and the information needed to continue. Training, repeated attempts and support calls are costs to investigate.

From Computer Desktop Encyclopedia
© 2007 The Computer Language Company Inc.



A useful interface supports a particular activity.

The click wheel combines browsing, selection and adjustment in a compact control. Consider finding an album and changing volume: which movements, labels and feedback support each task? The value of a control comes from how it supports the activity.

Consequences of design choices

Person	Completion, effort, confidence, recovery
Organization	Training, support, rework, adoption
Wider setting	Access, coordination, consequences of error

Usability affects both the person and the organization.

Establish which outcomes matter before choosing measures. An interface can save operator time while shifting work to support staff or customers.

A documented control-transfer problem

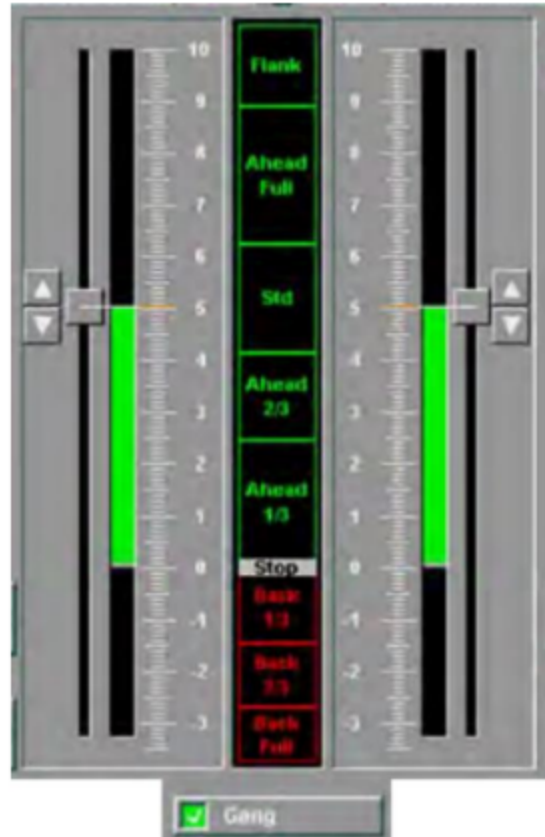


USS John S McCain collision, 2017. Report published in 2019.

The NTSB found that the control-system design increased the likelihood of operator errors. Its probable-cause finding also addressed Navy oversight, training and operating procedures. Design was part of a larger system of causes.

NTSB MAR-19/01 (2019), Figure 5; analysis pp. 32–33.

Feedback includes physical information

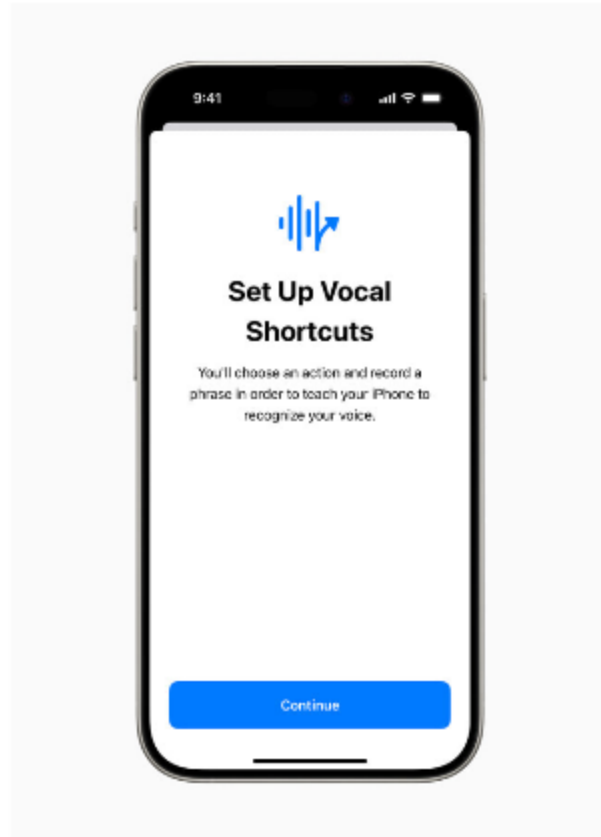


The report examined thrust transfer and missing tactile feedback.

Port and starboard thrust could be transferred separately. Touchscreen throttles lacked the physical feedback of mechanical controls. Operators needed to identify who controlled each side and how much thrust was applied. The interface must support both feedback and coordination.

NTSB MAR-19/01 (2019), Figure 6; analysis pp. 32–33.

Access can change the available action



Vocal Shortcuts: a 2024 accessibility example.

Apple introduced custom spoken utterances for triggering actions. This extends available input options. Whether a particular action works well still depends on the person, recognition behavior and setting.

Apple Newsroom, 15 May 2024. Vocal Shortcuts settings.

Useful capability

- Find rooms
- Check availability
- Request a reservation

Usable interaction

- Choose the correct room
- Understand the status
- Complete or recover

Does it support the goal? Can the intended user accomplish that goal?

A room directory may be useful for finding a location but insufficient for making a reservation. A reservation feature can exist and still be difficult to use.

Three dimensions of usability

Effectiveness
Correct and complete
outcome

Efficiency
Resources relative to
outcome

Satisfaction
Experience of using
the system

Usability concerns specified users, goals and context.

These dimensions answer different questions. A fast error is not successful performance; confidence is not proof of correctness.

Goal	Print the intended document
Accuracy	Correct file, pages and settings
Completeness	All required pages collected

Did the user achieve the specified goal accurately and completely?

An accepted command or a correctly stored request may be only an intermediate step. Define the result the person needs before declaring success.

Time	Choosing, correcting, waiting, collecting
Effort	Searching, entering and checking
Resources	Paper, repeated jobs, staff assistance

What resources did successful completion require?

Count time and effort in relation to the outcome. A short interaction that produces the wrong document is not efficient success. Distinguish active interaction from waiting for the service.

Performance and reported experience answer different questions.

A person may finish a task but feel uncertain, frustrated or dependent on assistance. Ask about experience and observe performance separately. Confidence does not prove correctness. A low error count does not establish a satisfying experience.

Experience extends beyond a successful click



A technically successful task can still feel uncertain or stressful.

Expectations before use, the interaction itself, and consequences afterward can all matter. Reported experience complements task performance; it does not replace it.

Operational definitions make usability assessable

Effectiveness	Was the intended reservation actually confirmed?
Efficiency	What time and effort did completion require?
Satisfaction	How did the participant describe the experience?

Task: reserve the correct room for the correct date.

Define the task boundary before measuring it. Time spent waiting for approval and time spent operating the interface may need separate treatment.

Correct software can still be confusing

System evidence

- Request stored correctly
- No data corruption
- Correct availability rules

Interaction evidence

- Status understood
- Correct task completed
- Errors recoverable

Technical correctness and understandable interaction need different evidence.

Both matter. A perfect status display cannot fix a lost booking, and a correct database does not guarantee that people understand its state.

Tradeoffs depend on the task

Routine, reversible action

- Low consequence
- Frequent repetition
- Fast correction available

Consequential action

- Costly or external result
- Scope needs checking
- Recovery may be limited

A design can improve one outcome while making another harder.

Confirmation can prevent an expensive mistake but add effort to frequent routine actions. Evaluate the balance for the actual task, its consequences and the intended users.

First use, practiced use and return after a long gap differ.

A clear first encounter matters for occasional users. Experts may benefit from shortcuts and stable commands. An evaluation should specify the experience level and usage frequency it addresses.

A goal describes an intended outcome. A task describes the work needed to reach it.

An operation is an action through the interface, such as selecting a file or pressing Print. Task analysis connects these operations to the larger goal, including prerequisites and completion checks.

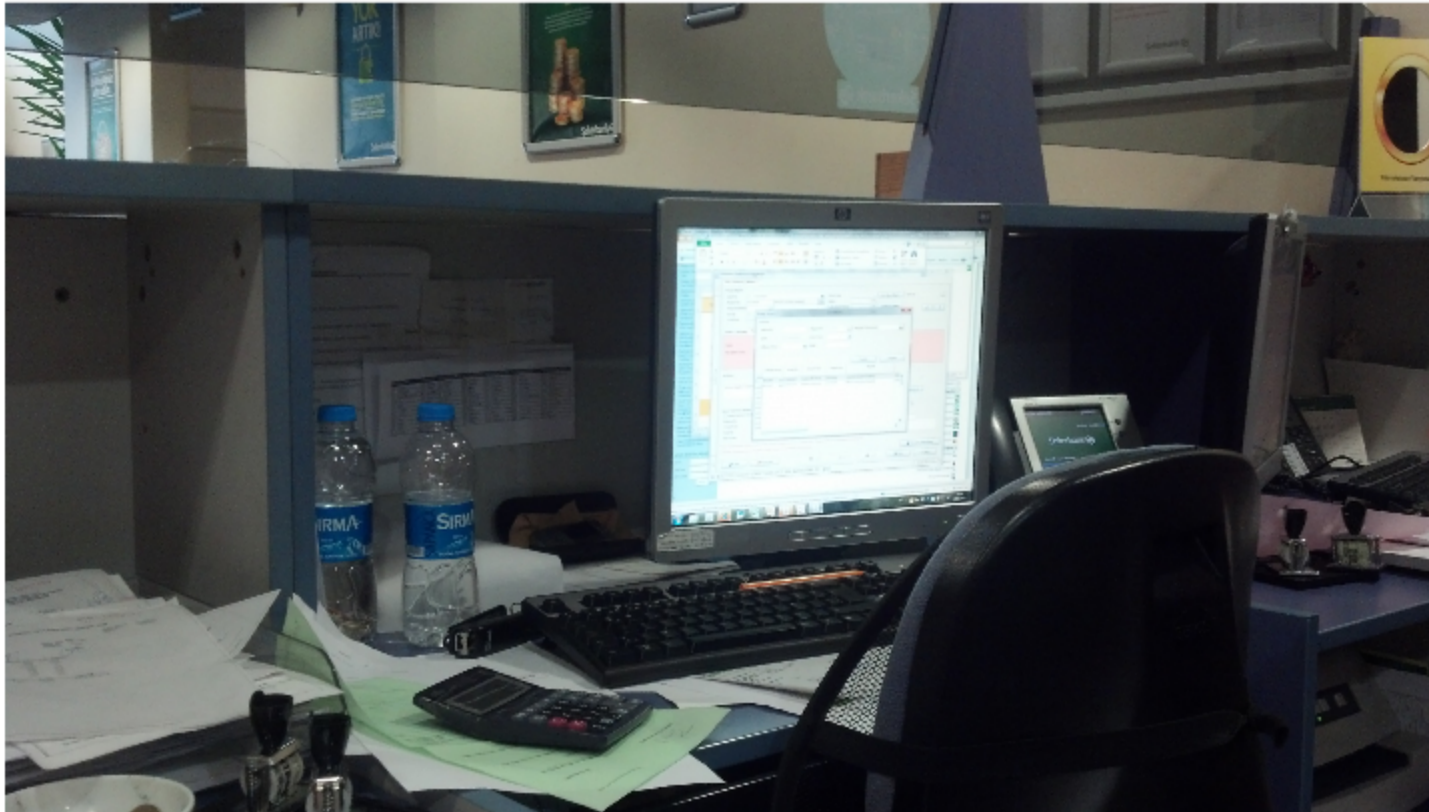
Different users, different requirements

Domain expert	Knows the work but may be new to the system
System expert	Knows the interface but may lack domain knowledge
Occasional user	Must resume after forgetting details

Computer experience does not establish knowledge of this task.

Consider domain knowledge, language, physical capabilities and frequency of use. These dimensions can vary independently. Avoid designing around a single imaginary average user.

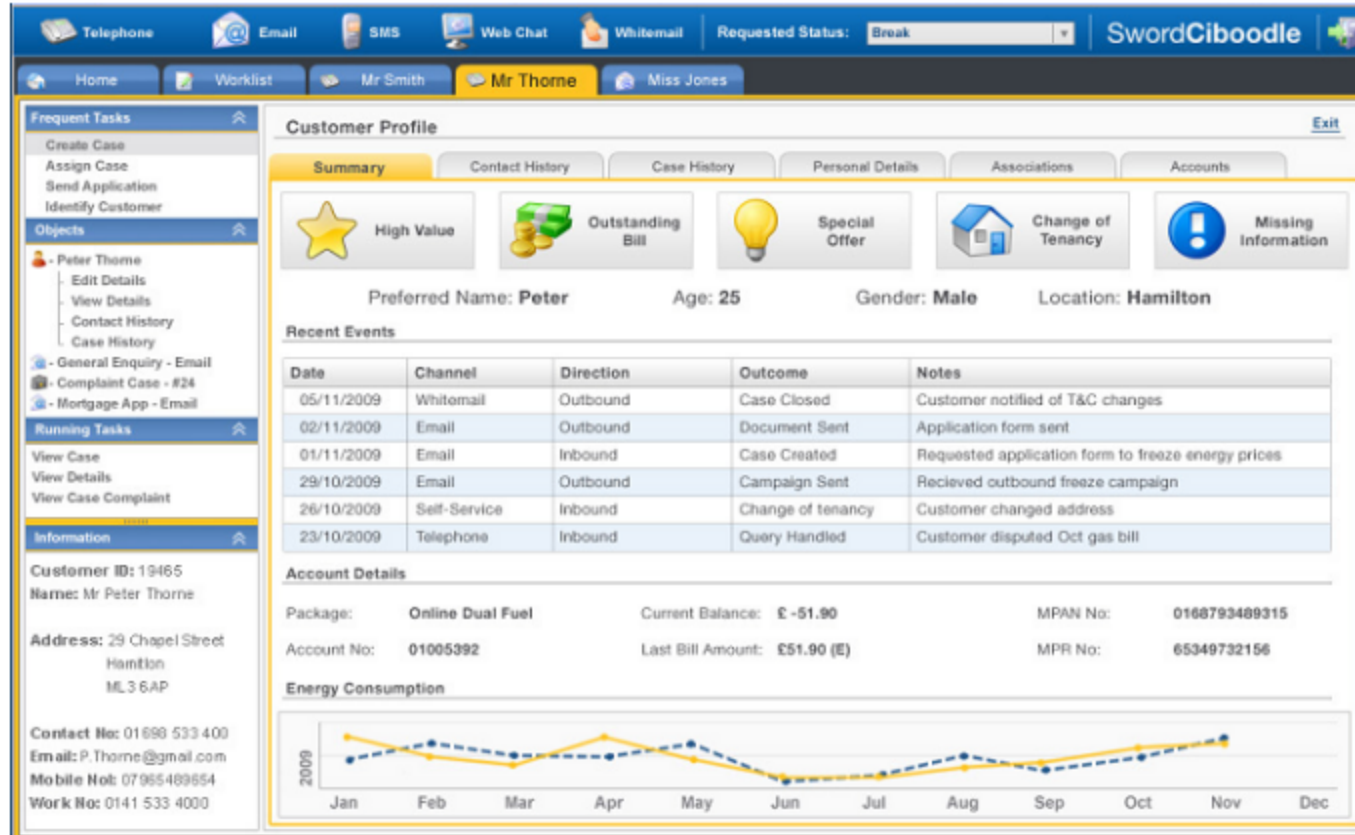
A workstation is also a workplace



Documents, interruptions and other people are part of the task.

Work continues around the screen: referring to paper, answering questions, switching activities and coordinating with someone nearby. Understanding this setting helps determine what information should stay visible and how an interrupted task can be resumed.

The developer's screen is not the whole workplace



A rich interface can support work or impose extra searching.

This historical service interface includes customer records and multiple views. Which information is necessary for the current task? Observe coordination with conversation, paper and interruptions before deciding what to simplify.

People affected beyond the operator

Operator	Performs the interaction
Recipient	Receives the service or its consequences
Support and administration	Maintains access, resolves failures, sets policy

The operator may not be the only stakeholder.

Design decisions distribute work and consequences. Identify these relationships explicitly instead of treating every stakeholder as an identical end user.

Quiet, prepared use

- Time to read
- Stable workspace
- Help available

Busy service point

- Other people waiting
- Divided attention
- Limited time to retry

An interface can be easy in one setting and difficult in another.

A quiet desk and a busy service point differ in interruptions, privacy and available time. A successful demonstration does not by itself establish field performance.

Requirements need a source

Need	Avoid collecting the wrong print job
Possible requirement	Identify owner and document before release
Evidence to seek	Observed mix-ups and interpretation

Connect a user need to a verifiable outcome.

A requested feature may be one solution to a deeper need. Ask what problem it addresses, then consider alternatives. Keep observed needs separate from assumptions.

A scenario makes assumptions inspectable

A concrete scenario names the person, goal, setting and relevant conditions.

An occasional user needs two double-sided copies of a report before a meeting, at a shared printer, with other people waiting. Selection, quantity, privacy and completion now have context. The scenario remains a design hypothesis until checked with users.

Observation	A person presses Print twice
Possible explanations	No feedback, uncertain progress, accidental repeat
Next evidence	Observe timing and ask what they expected

An explanation of difficulty should be testable.

“The control is confusing” needs a mechanism and an observation. Different explanations can suggest different redesigns. Test the explanation before investing in a polished solution.

Interaction styles distribute work differently

Command

```
> reserve room A  
--date Monday  
--time 14:00
```

Direct selection

Room A • Monday

14:00

15:00

Delegation

Find a room for Monday.

Plan: check availability,
then request a booking.

Commands, direct selection and goal-based delegation distribute work differently.

Each requires something from the user: knowing a syntax, recognizing an object or inspecting a proposed plan. None removes the need for feedback and control.

Commands provide a compact language for specifying actions.

Commands support precise, repeated and composable operations. They require knowledge of syntax, names and available options. Discoverability and error feedback affect learning and recovery. Expertise changes the balance between speed and recall.



Menus and selectors expose possible choices.

A visible list reduces the need to recall a command. Users must still understand labels and find the relevant option. This appliance dial mixes categories and settings. Which task distinctions does the arrangement communicate?



The control can correspond to the object being changed.

Seat-shaped controls offer a concrete example of correspondence. Direct manipulation more generally uses visible objects, actions and observable results. Resemblance alone does not establish every direction, limit or consequence.

Form-based interaction: fields, actions and status

Room A • Reading group

Day

Monday

Time

14:00–15:00

Request submitted

[Return home](#)

A student needs a room for a reading-group meeting.

The screen acknowledges a request.
What result can the student safely infer from it?

Fictional booking interface

Speech interaction changes the demands

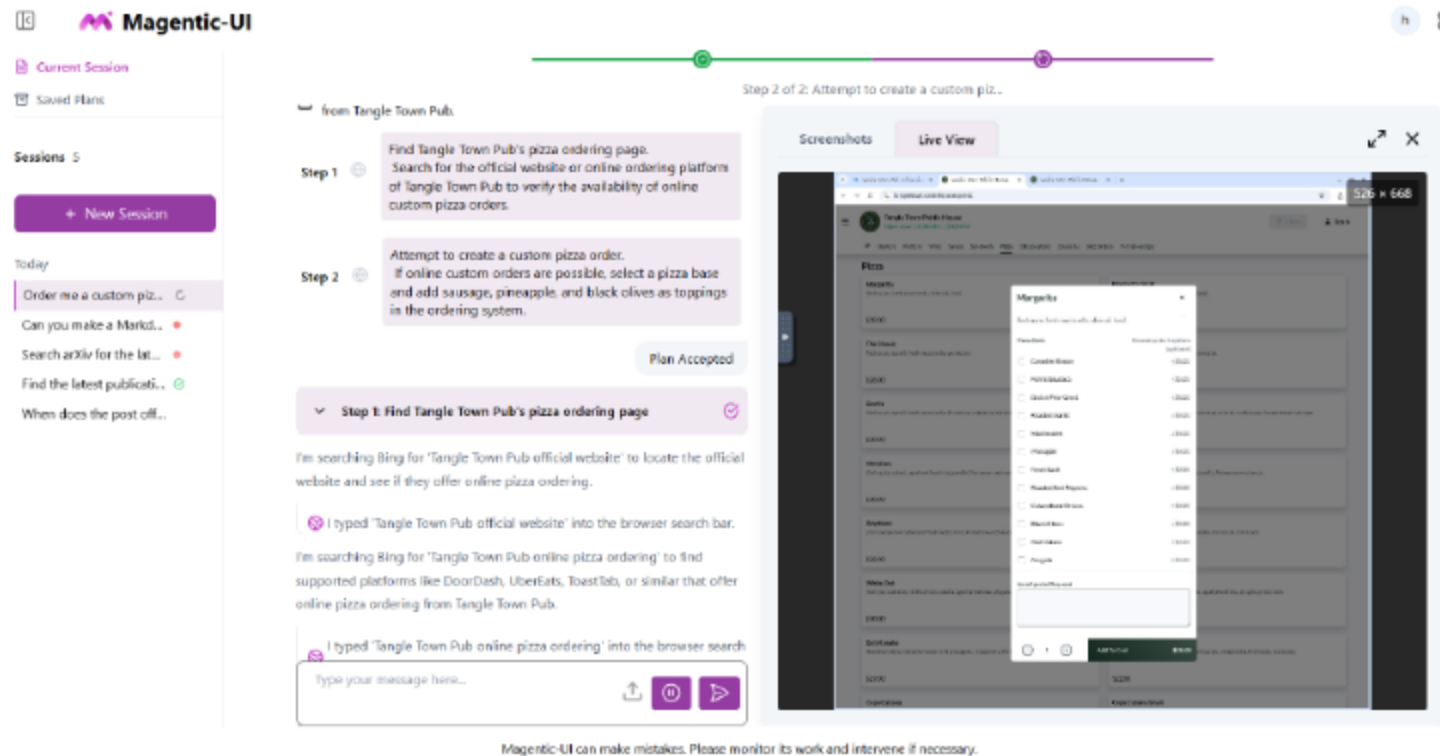


A spoken action still needs discoverability and feedback.

What phrases are available? Was the utterance recognized? Can the person cancel or correct the action? Speech provides another channel, with questions about privacy, environmental noise and recognition errors.

Apple Newsroom, 15 May 2024. Vocal Shortcuts settings.

Delegation to a software agent

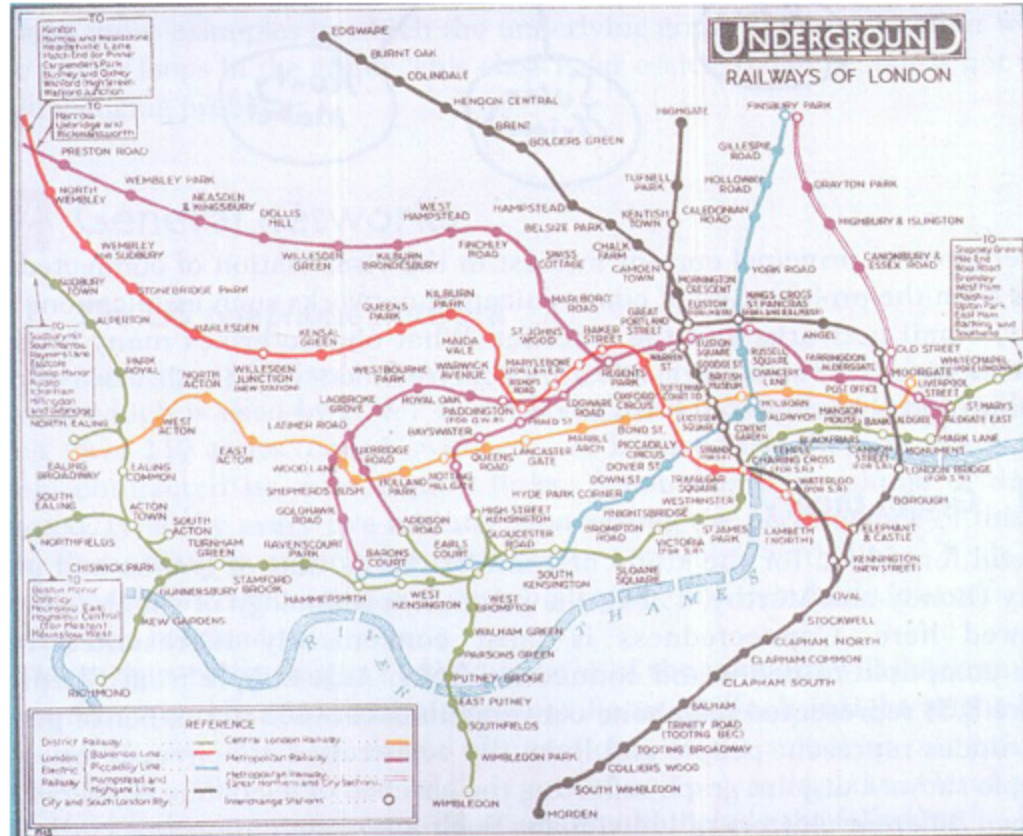


Magentic-UI • Microsoft Research • 2025

The interface places a visible plan beside an active browser workspace. Users can inspect the intended action and its progress, and intervene while the agent works.

Mozannar et al. (2025), Magentic-UI, p. 4, Fig. 2

Representation changes the available task



A geographically grounded map preserves spatial relationships.

This historical London Underground map represents a transport network. Locating a station in the city and planning a transfer are different tasks. What information does this representation make easy to inspect?

Brad Hemminger, INLS 507, Introduction and Framework, slides 57–58.

A network diagram emphasizes connections



A schematic map can simplify line following and interchange decisions.

Compare route planning with judging street distance. Straightened lines and clear interchange symbols emphasize connections, while distorted geography makes distance harder to estimate. A representation helps by preserving the relationships needed for the task.

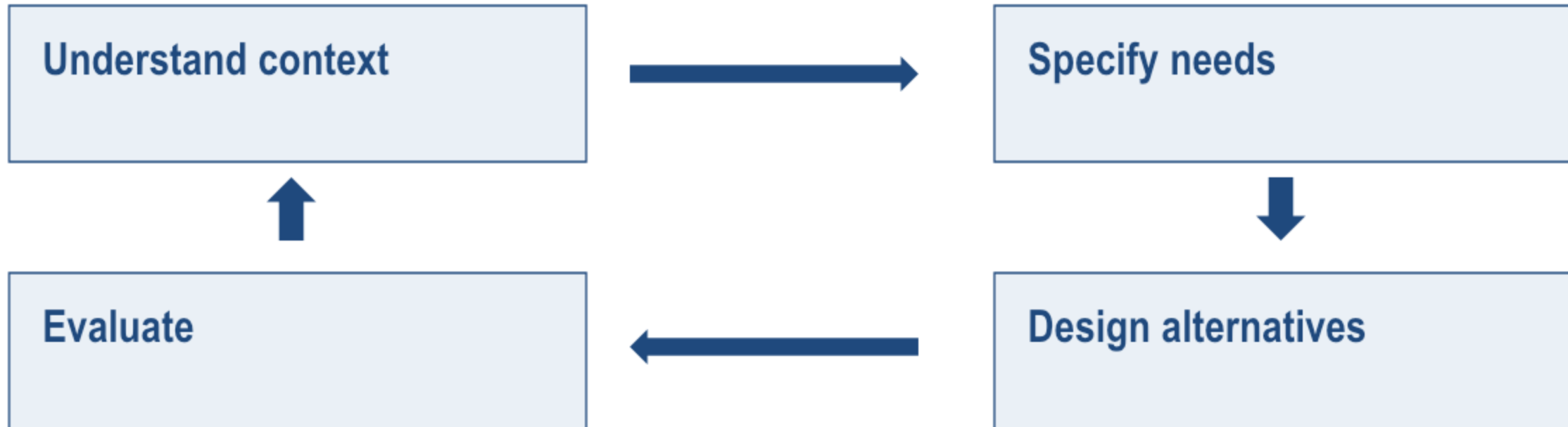
Choosing an interaction style

Specify	Can the user express the intended action?
Check	Can the user determine the current state?
Recover	Can the user correct or revise a decision?

Choose according to task demands and user capabilities.

Systems often combine styles. A voice request can produce a visual draft, a menu can reveal commands, and a direct adjustment can show a numeric value. Evaluate the complete interaction.

Human-centered design is iterative

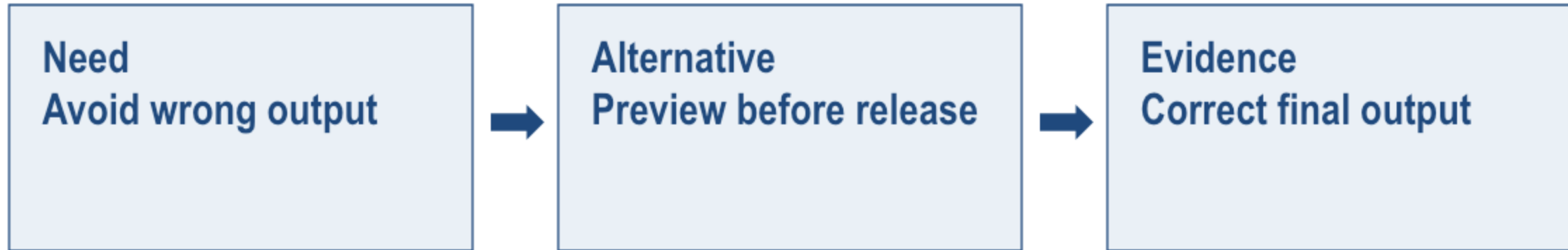


Evaluation can change the requirements as well as the design.

Understand context and needs, create alternatives, and evaluate them. Return to the relevant earlier step when evidence exposes an incorrect assumption.

Understanding the work shapes the requirements.

User involvement and evaluation belong throughout development. Early evidence can change what should be built. Styling a finished implementation cannot repair every mismatch between the system and the work.



A design decision should have a reason that can be evaluated.

Start with the user need, consider multiple solutions and identify evidence that could discriminate between them. A polished prototype is not evidence of success.

Defaults first

- Fewer initial decisions
- Visible settings needed
- Easy correction needed

Explicit choices first

- More initial decisions
- Important scope exposed
- Extra effort possible

Exploring alternatives tests assumptions in the first idea.

Preselecting common settings may reduce effort but hide inappropriate defaults. Requiring every setting each time may expose assumptions but slow routine use. Compare consequences before choosing.

Represent enough of the interaction to investigate an uncertainty.

Paper sketches can test terminology and sequence. Interactive prototypes can test state transitions. A working implementation may be needed for timing or hardware behavior. Match fidelity to the question instead of treating realism as the main objective.

Evaluation methods provide different evidence

Inspection	Potential issues against principles or tasks
Observed task use	Actions, completion, difficulties, explanations
Field study	Actual work, interruptions and coordination

Inspection, observed use and field work complement one another.

Expert review identifies plausible problems. Observation shows how people act and interpret. Field work reveals conditions a controlled session may omit. None automatically replaces the others.

Evaluation may reveal an incorrect assumption about the work.

If users need to release a job from another location, moving a button does not address the requirement. Return to the relevant earlier decision, revise the design and evaluate again. Iteration responds to evidence.

Classroom case: a shared printing kiosk



Obtain the right output and leave the device ready for others.

An occasional user needs two double-sided copies of a report before a meeting. Other people are waiting. The task includes choosing the right file and settings, checking progress, collecting the output and leaving the kiosk ready for the next person.

AI-generated illustration • Classroom printing scenario

The task and its boundaries

Input	Correct six-page report
Settings	Two copies, double-sided
Outcome	Two complete copies collected by their owner

Print two double-sided copies of the intended six-page report.

Each complete duplex copy normally uses three sheets, so the intended output is six sheets total. Assume duplex support and no blank separators. The task ends after checking and collecting both copies.

What would count as success?

Effectiveness	Correct file, two complete duplex copies
Efficiency	Task time, corrections and repeated jobs
Experience	Reported certainty, effort and frustration

Define measures before comparing designs.

Distinguish selection mistakes from printing failures. Record assistance and repeated attempts. Ask about confidence after checking the output.

A hypothetical observation

Action	Presses Print, waits, presses Print again
Outcome	Two submitted jobs
Question	What did the person believe after the first press?

The user repeats Print while the first job is processing.

What would explain the second press: no acknowledgement, unclear progress, or an attempt to print another copy?
The explanation determines the redesign. Observe the timing and ask what the person expected.

A proposed redesign

Before release	Document, pages, copies, duplex setting
After release	Job identity and current status
Recovery	Correct or cancel when possible

Show document and settings before release, then show job status.

A preview supports checking the file and scope. Job status distinguishes queued, printing and ready. If another press could duplicate the job, expose that consequence. Cancellation must reflect whether printing can still stop.

An evaluation that tests the proposal

Hold consistent	Task, document and output requirements
Observe	Repeated release, mistakes, assistance, outcome
Ask	Meaning of queued, printing and ready

Can occasional users obtain the output without duplicate submission?

Give occasional users the same printing task and check the final output. Record duplicate jobs, mistakes, assistance and completion time. Ask what queued, printing and ready meant to them; compare their interpretations with the actual job state.

A redesign still has tradeoffs

More checking

- May prevent costly mistakes
- Adds interaction time
- Requires understandable detail

Faster release

- Supports frequent routine use
- Depends on visible defaults
- Needs effective recovery

The task determines how much checking is useful.

A preview can prevent a wrong job but add time. Confirmation on every action can become routine clicking. Accessibility, privacy and recoverability remain requirements even when one design is faster.

Connect the goal, difficulty, design proposal and evaluation.

Duplicate submission motivates clearer job status and a visible consequence of resubmitting. Evaluation checks duplicate jobs and state interpretation. The reasoning transfers to booking and delegated actions while preserving the differences between domains.

The fourteen-week teaching sequence

1 Foundations	2 Human capabilities	3 Users & tasks	4 Interaction design	5 Prototyping	6 Experiments	7 Midterm
8 Accessibility	9 Mobile / embodied	10 Visualization	11 Human–AI	12 Collab. & trust	13 Paper talks	14 Paper talks

Human capabilities, design methods and contemporary applications

Weeks 1–6 and 8–12 are lecture weeks. Week 7 is the midterm; Weeks 13–14 are reserved for paper presentations. The final exam is outside this teaching sequence.

30%

Midterm

40%

Final

15%

Assignments

15%

Presentation

Assignments: W2 W4 W6 W8 W10

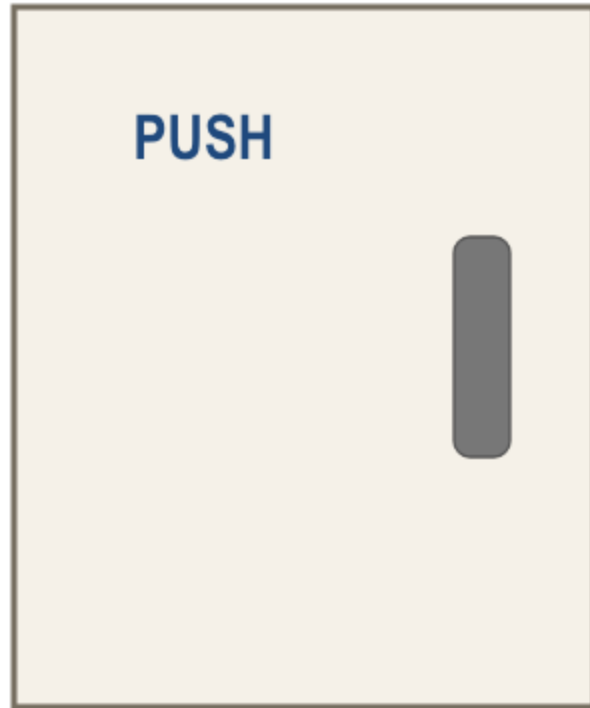
Five assignments, beginning in Week 2

Midterm 30%; final exam 40%; assignments 15%; paper presentation 15%. Assignments fall in Weeks 2, 4, 6, 8 and 10, worth 3% each. No quizzes or Week 12 assignment. Papers are distributed after the midterm.

Original material and complementary sources

Göktürk, Lecture 1: motivation, appliances and workplace examples.
Georgia Tech CS/PSY 6750, Stasko et al. (2007): goals and evaluation.
James Tam, CPSC 481: user involvement and design process.
Brad Hemminger, INLS 507: visual representation and tasks.
BEA Air Inter inquiry; NTSB MAR-19/01 (2019).
Apple accessibility announcement (2024); Magentic-UI report (2025).
Specific source paths and URLs accompany the slide notes.

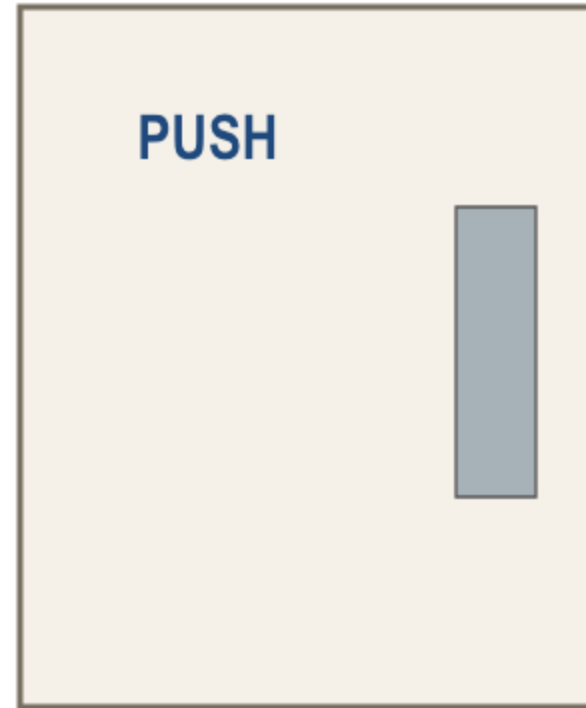
Closing video: confusing doors



Pull-like handle

It's not you. Bad doors are everywhere.

[Open video on YouTube](#)



Broad push plate

Vox / 99% Invisible. Identify the misleading cue, propose a change, and explain why an instruction may compete with physical form.

Perception establishes what reaches the person. Memory connects it with experience.

Working memory supports holding and manipulating information during a task. Long-term memory supports learned knowledge and skills. Mental models are explanations people use to predict behavior. The refrigerator case follows these foundations in Week 2.