Human Computer Interaction

User-centred design and usability heuristics

Hisham Ihshaish | Birzeit University

COMP322 | Based on: Interaction Design, 6th Ed. (Rogers, Sharp, Preece)

Today's agenda

01 **Recap**

Key ideas from Lecture 1 in 60 seconds

02 **Why do we need a process?**

A real failure that motivates UCD

03 **User-centred design (UCD)**

ISO 9241-210, its principles, and the iterative cycle

04 **Usability engineering**

Making usability measurable, not aspirational

05 **Nielsen's 10 usability heuristics**

Practical guidelines for evaluating any interface

Lecture 1 in 60 seconds

HCI	The study of how people interact with technology, and how to design better interactions
Design matters	Poor design causes real harm (Therac-25, Hawaii missile alert). Good design is invisible.
Usability	Effectiveness, efficiency, learnability, memorability, error tolerance, satisfaction (ISO 9241-11)
UX	The total experience: usability plus emotions, aesthetics, and context
Norman's principles	Affordances, signifiers, mapping, feedback, constraints, conceptual models

Why do we need a design process?

Case study: Healthcare.gov (October 2013)

The US government launched Healthcare.gov, a website for citizens to purchase health insurance. Budget: over $500 million. On launch day, of the millions who tried to use it, only 6 people successfully enrolled. The site crashed repeatedly, forms lost data, error messages were incomprehensible, and the user flow required 76 screens to complete a task that should have taken minutes.

Post-mortem analysis revealed: no user testing before launch, no iterative design, requirements defined by policy teams without consulting end users.

What went wrong?

They skipped the process. No understanding of users. No iterative prototyping. No evaluation until it was too late. Half a billion dollars later, they had to rebuild from scratch using the very methods we are about to learn.

Source: GAO-14-694, US Government Accountability Office (2014)

User-centred design

Putting users at the heart of every decision

What is user-centred design?

> *An approach to interactive system development that focuses specifically on making systems usable. It is a multi-disciplinary activity, which incorporates human factors and ergonomics knowledge and techniques.*
>
> ISO 9241-210:2019

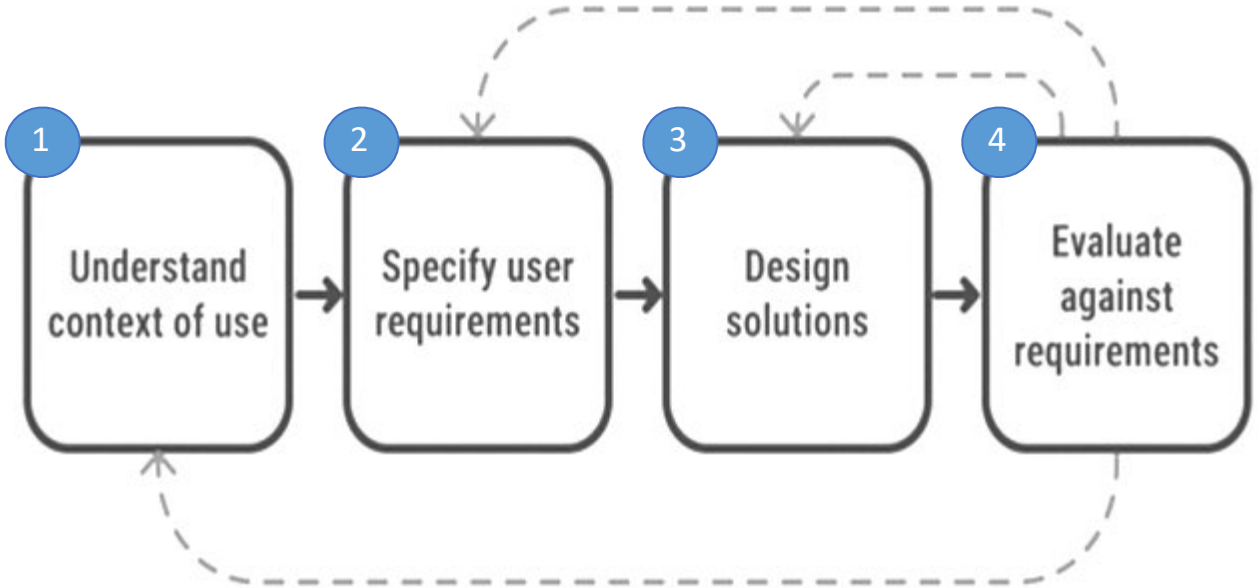
In practice, UCD means:

- Start with the users: who they are, what they need, and how they work
- Design around them, test with them, iterate until it works
- Treat this as a cycle, not a one-off handover of requirements

The term 'user-centred design' was coined by Donald Norman and Stephen Draper in their 1986 book at UC San Diego. ISO standardised the process in 1999 (updated 2010, then 2019).

The UCD cycle (ISO 9241-210)

Four activities, repeated until the design meets user needs



Based on ISO 9241-210:2019, Figure 1

Six principles of UCD

ISO 9241-210:2019, Section 4

1. The design is based upon an explicit understanding of users, tasks, and environments.
2. Users are involved throughout design and development.
3. The design is driven and refined by user-centred evaluation.
4. The process is iterative.
5. The design addresses the whole user experience.
6. The design team includes multidisciplinary skills and perspectives.

Compare Healthcare.gov: principles 1, 2, 3, and 4 were all violated.

Usability engineering

Making usability measurable, not aspirational

Usability engineering in practice

Set targets before you design. Verify them after.

A vague goal: *"The system should be user-friendly."*

A measurable target: *"A new user completes checkout in under 90 seconds with zero errors."*

Example: usability targets for a food delivery app

Usability goal	Measurable target
Learnability	New user places first order within 3 minutes without help
Efficiency	Returning user reorders last meal in under 15 seconds
Error tolerance	Address entry error rate below 5%
Satisfaction	Post-task satisfaction averages 4/5 or higher

These targets map directly to the six usability goals from Lecture 1 (ISO 9241-11).

Nielsen's 10 usability heuristics

Rules of thumb for evaluating any interface

Jakob Nielsen

Heuristics 1 to 3

H1 Visibility of system status

وضوح حالة النظام

The system should always keep users informed about what is going on, through appropriate feedback within reasonable time.

Good: WhatsApp's blue ticks: sent, delivered, read. A progress bar during uploads.

Bad: A form that submits silently with no confirmation or loading indicator.

H2 Match between system and real world

التوافق مع العالم الحقيقي

The system should speak the users' language with familiar words and concepts, rather than system-oriented terms. Follow real-world conventions.

Good: A shopping cart icon for purchases. A recycle bin for deleted files.

Bad: 'Exception 0x80070005: Access Denied' shown to a non-technical user.

H3 User control and freedom

تحكم المستخدم وحريته

Users often perform actions by mistake. Provide a clearly marked 'emergency exit' without extended dialogue. Support undo and redo.

Good: Gmail's 'Undo Send' (5-30 second window). Ctrl+Z in editors. Browser back button.

Bad: A multi-step form with no back button. Deleting a file with no way to recover it.

Heuristics 4 to 6

H4 Consistency and standards

الاتساق والمعايير

Users should not wonder whether different words, situations, or actions mean the same thing. Follow platform conventions.

Good: The floppy disk icon for 'Save' across all apps. Underlined blue text = hyperlink.

Bad: 'Log in', 'Sign in', and 'Access account' used for the same action on different pages.

H5 Error prevention

منع الأخطاء

Even better than good error messages is a design that prevents problems from occurring in the first place. Eliminate error-prone conditions.

Good: Greying out unavailable options. Confirmation before permanent deletion. Auto-formatting phone numbers.

Bad: A 'Delete all' button placed right next to 'Save' with identical styling and no confirmation.

H6 Recognition rather than recall

التعرف بدلا من التذكر

Minimise the user's memory load. Make objects, actions, and options visible. Do not require remembering information across screens.

Good: Dropdown menus showing all choices. 'Recently opened' file lists. Search suggestions as you type.

Bad: A system requiring you to type a product code from memory instead of a searchable list.

Heuristics 7 to 9

H7 Flexibility and efficiency of use

المرونة وكفاءة الاستخدام

Accelerators, unseen by novices, may speed up expert interaction. Allow users to tailor frequent actions.

Good: Keyboard shortcuts (Ctrl+C). VS Code's command palette (Ctrl+Shift+P). Mobile gestures.

Bad: A system forcing experts through a 5-step wizard for an action they perform 50 times a day.

H8 Aesthetic and minimalist design

التصميم الجمالي والبسيط

Dialogues should not contain irrelevant or rarely needed information. Every extra element competes with the relevant ones.

Good: Google's homepage: one search bar. Apple product pages: focused, clean, intentional.

Bad: A government portal with 50+ links, flashing banners, and three navigation menus.

H9 Help users with errors

مساعدة المستخدمين مع الأخطاء

Error messages should be in plain language (no codes), precisely indicate the problem, and constructively suggest a solution.

Good: 'Password must be at least 8 characters. You entered 5.' Specific and actionable.

Bad: 'Error 500: Internal Server Error' with no explanation or next steps.

Heuristic 10

H10 Help and documentation

المساعدة والتوثيق

Even though it is better if systems are usable without documentation, help should be easy to search, task-focused, and concise.

Good: Duolingo's learning-by-doing onboarding. Slack's contextual tooltips on first use.

Bad: A 200-page PDF manual with no search function and no table of contents.

Why is H10 last?

Nielsen placed it last deliberately. If you need extensive documentation, it often means H1 through H9 were not followed well enough. Documentation should be a safety net, not a crutch for poor design.

All 10 heuristics at a glance

H1 Visibility of system status

H2 Match between system and real world

H3 User control and freedom

H4 Consistency and standards

H5 Error prevention

H6 Recognition rather than recall

H7 Flexibility and efficiency of use

H8 Aesthetic and minimalist design

H9 Help users with errors

H10 Help and documentation

First published: Molich & Nielsen (1990). Refined: Nielsen (1994). Unchanged for 30+ years.

Visibility of System Status

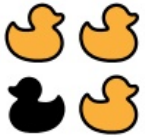
1

Match Between System & Real World

2

User Control And Freedom

3

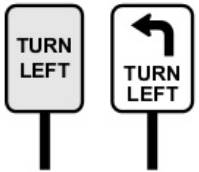


Consistency And Standards

4

Error Prevention

5

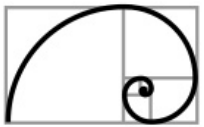


Recognition Rather Than Recall

6

Flexibility And Efficiency of Use

7



Aesthetic And Minimalistic Design

8

Help Users With Errors

9

Help And Documentation

10

Excerpt from: https://medium.com/@hazlazuardi/how-my-team-uses-nielsens-10-usability-heuristics-in-our-application-865fef19220d
Also visit: https://www.nngroup.com/articles/ten-usability-heuristics/ for tips and guidelines.

Name that heuristic

For each situation, identify the violated heuristic. Post your answer in the chat.

1	You delete an important email. There is no 'Undo' option and no trash folder. The email is gone forever.	**Answer: H3**
2	A banking app shows your balance but uses the label 'Net Disposable Liquidity' instead of 'Available Balance.'	**Answer: H2**
3	You fill in a long form and click submit. A blank white page appears. No message, no spinner. Did it work?	**Answer: H1**

Bonus: Can you suggest a design fix for each one?

Norman's principles vs Nielsen's heuristics

Two complementary tools, not two versions of the same thing

Norman's 6 principles

Purpose: Understand WHY a design works or fails

Origin: Cognitive science (Norman, 1988/2013), built on Gibson's ecological psychology (1979)

Scope: All designed objects: doors, stoves, apps, cars

Use: Generative. Helps you CREATE good designs

Question: *"Does this design match how humans think and perceive?"*

Nielsen's 10 heuristics

Purpose: Check WHETHER an interface has usability problems

Origin: Empirical research. Factor analysis of 249 usability problems (Nielsen, 1994a, CHI '94)

Scope: Interactive systems: apps, websites, software

Use: Evaluative. Helps you FIND problems in designs

Question: *"Does this interface follow established usability guidelines?"*

Where they overlap

Norman's "feedback" relates to Nielsen's H1 (visibility of system status). Norman's "constraints" relates to H5 (error prevention). But Norman explains the psychology; Nielsen provides the checklist.

Analogy: Norman is the physics of how a bridge carries weight. Nielsen is the safety inspection checklist.

What you learned thus far

UCD process	A formal ISO standard (9241-210) with six principles and four iterative activities: understand, specify, design, evaluate.
Usability engineering	Set measurable targets before design. Verify them after. Originated by Nielsen (1993).
10 heuristics	Nielsen & Molich (1990), refined via factor analysis of 249 problems (Nielsen, 1994). A practical checklist for any interface.

Next session

From principles to practice

Applying everything from previous Lectures:

- How to conduct a heuristic evaluation (Nielsen's method, **severity ratings**)
- Personas and scenarios: modelling your users (Cooper, 1999)
- Accessibility and inclusive design (WCAG 2.2, universal design)
- Hands-on activity: spot the violations

Suggested reading:

Nielsen's 10 Heuristics: nngroup.com/articles/ten-usability-heuristics/

Rogers, Sharp, Preece. Interaction Design, Chapter 1.

Norman, D. The Design of Everyday Things, Chapters 1-2.

References

Rogers, Y., Sharp, H. & Preece, J. (2024). Interaction Design, 6th Ed. Wiley.

Norman, D. & Draper, S. (1986). User Centered System Design. Lawrence Erlbaum.

ISO 9241-210:2019. Human-centred design for interactive systems.

ISO 9241-11:2018. Usability: Definitions and concepts.

Nielsen, J. (1993). Usability Engineering. Morgan Kaufmann.

Molich, R. & Nielsen, J. (1990). Improving a human-computer dialogue. Comm. ACM, 33(3), 338-348.

Nielsen, J. (1994a). Enhancing the explanatory power of usability heuristics. Proc. CHI '94, ACM, 152-158.

Nielsen, J. (1992). Finding usability problems through heuristic evaluation. Proc. CHI '92, ACM, 373-380.

Miller, G.A. (1956). The magical number seven. Psychological Review, 63(2), 81-97.

Shneiderman, B. (1987). Designing the User Interface. Addison-Wesley.

GAO (2014). Healthcare.gov: Ineffective Planning and Oversight. Report GAO-14-694.