COMP322

From who to what.

Scenarios and measurable requirements.

Personas tell you who. Scenarios tell you what they are trying to do in context. Requirements turn that into something you can build, and test.

Hisham Ihshaish

Birzeit University - Department of Computer Science

A small detour....

We are taking these two topics out of textbook order. Here is why, openly.

WHAT ROGERS/SHARP/PREECE EXPECT

Chapter 1 - what is interaction design?
Chapter 2 - conceptualising interaction (Norman)
Chapter 3 - cognitive aspects
Chapter 4 - social interaction
Chapter 5 - emotional interaction
Chapter 6 - interfaces
Chapter 7 - data gathering
Chapter 9 - the IxD process
Chapter 10 - establishing requirements
Chapter 11 - design, prototyping

WHAT PHASE 1 NEEDS BY 23d MAY

Personas - done last session

Scenarios - today

Measurable usability requirements - today

Heuristic evaluation - already covered

Exploratory interviews

The chain you are building.

Each link constrains the next. Break a link and the design loses its grounding.

PERSONA

Who

Who is this design for?

Done last session

SCENARIO

What

What are they trying to do, in context?

REQUIREMENT

How well

What must our design provide, measurably?

TEST

Did it?

Did the design actually deliver?

Phase 3

By the end of this hour you will be able to:

01

Write a scenario

Not a feature list. Not a wish. A short narrative grounded in a specific persona, with a specific context and a specific goal. You will know the four tests of a good one.

02

Lift a scenario into requirements

Carroll's claim analysis: each scenario implies things the design must do. You will know how to make those things measurable and how to map each to one of the six ISO 9241-11 usability goals.

03

Trace the chain

Persona to scenario to requirement to test. You will know how to write requirements that point back, so a marker (or a stakeholder, or your future self) can audit your design decisions.

Quick reminder - meet Lina.

From your "Interactive Companion" on the Course page: She is the persona running through today's worked examples - here is the card.

PERSONA - LINA KHOURY

LK

Lina Khoury

Second-year Computer Science student

Anchored by **P3 (Lina, 20, Ramallah)** - the participant closest to the cluster centroid

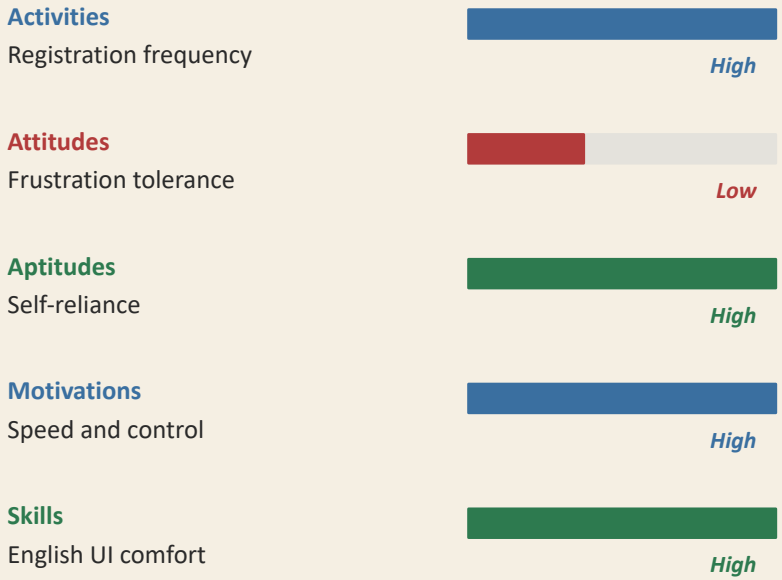
Summary

Registers electives on her phone, often in 10-15 minute gaps between classes. Tech-comfortable, English-fluent, prefers efficient flows. Frustrated by vague "course full" messages with no waitlist, and by slow advisor approvals that interrupt her registration sessions.

> *"I just want to know quickly if a course is full or not - and if it is, when can I try again."*

CLUSTER EVIDENCE

3 participants (P1, P2, P3) anchored by P3. Centroid scores on the five behavioural variables:



More depth: In previous lectures and the interactive walkthrough on the course site. *There is also a second persona (Maryam, cautious delegator) - she is on the handbook but does not feature in today's worked examples.*

What happens when the chain breaks.

PERSONA

Lina, second-year CS student

Registers on her phone between classes. Frustrated by slow loads and vague "course full" messages. Wants advisor confirmation in one flow.

SCENARIO

Adding an elective between classes

Lina has 15 minutes between classes. She opens Ritaj on her phone, scans suggested electives that fit her timetable, adds one, gets advisor confirmation 90 seconds later.

REQUIREMENT

The system should use dark mode

***Dark mode** is more popular among young people and improves the user experience. The system shall support a dark theme.*

WHERE THE CHAIN BROKE

Lina's interview never mentioned dark mode. The scenario does not need dark mode to succeed. Neither persona nor scenario justifies this requirement.

This is the single most common Phase 1 failure: requirements that float free of the user research that came before them. The team had real personas and real scenarios, then wrote requirements from intuition or trend-watching.

PART ONE OF TWO

Scenarios.

What your persona is trying to do, in context, told as a story.

What a scenario actually is.

And why Carroll's tradition matters more than the textbook lets on.

JOHN M. CARROLL - MAKING USE, MIT PRESS, 2000

*"A scenario is a **story** about **people** and their **activities**. It includes a setting, agents or actors, who typically have goals or objectives, and a **plot** or sequence of actions and events. Scenarios are made of natural-language description, not formal notation. They are concrete, particular, and easy to dispute."*

Four things this definition fixes:

Particular

About one specific person doing one specific thing. Not 'users register for courses'. Lina, on Tuesday, on her phone.

Contextual

Includes setting. Where she is, what device she has, what else is happening. Context shapes what counts as success.

Narrative

Told as a story with a beginning, middle, end. Not a bullet list. Not a flowchart. Prose with a plot.

Disputable

Specific enough to be wrong. A teammate can read it and say 'no, Lina would not do that'. Vague scenarios cannot be disputed and so cannot be improved.

Three flavours of scenario, three uses.

Cooper, Reimann et al. (About Face, 4th ed., 2014) distinguish three. You will likely use two of them in Phase 1.

Goal scenario

Phase 1 - early design

Describes a persona pursuing a goal at the highest level. Abstracts away from any specific interface. Focuses on motivation and outcome.

Example

Lina wants to add an elective to her timetable without conflicting with her existing schedule, before her friends fill up the popular sections.

What you write in Phase 1.

Task scenario

Phase 2 - design refinement

Walks through a specific task on a specific (proposed) interface. Names the buttons, the screens, the sequence. Tied to a prototype.

Example

Lina opens the redesigned Ritaj. She taps 'Suggested Electives'. She sees three that fit her timetable. She taps the first. She confirms.

What you write in Phase 2 to defend your prototype.

Use scenario

Phase 3 - evaluation

The task script you give to participants in a usability test. Outcome-focused, neutral, does not lead the user toward any particular button.

Example

You want to add an elective that fits your timetable. Use the prototype to do this.

What you give a test participant in Phase 3.

In your project you will write three GOAL scenarios for Phase 1. Task and use scenarios come later.

Where the scenario actually comes from.

Three sources, synthesised into one short story. You are not inventing - you are integrating.

01 Persona pattern

from clustering analysis

Who the protagonist is. Their context, tech literacy, frustrations, representative quote. The persona supplies the WHO.

Example

Lina, second-year CS, registers on her phone between classes, frustrated by vague "course full" messages.

02 Interview evidence

from the transcripts

Specific moments participants described. Real phrases, real situations, real frustrations. Interviews supply the TEXTURE.

Example

P2: "I check between my algorithms class and the next one - 15 minutes - then it says course full but no waitlist."

03 Task knowledge

from existing-system research

What tasks the system supports. Your team walked through the current Ritaj, read its docs, watched users register. This supplies the WHAT.

Example

Ritaj supports: search electives, view capacity, add to plan, check conflicts, advisor approval workflow.

synthesised into

ONE GOAL SCENARIO

Lina's scenario takes the protagonist from **(1)**, a moment of use described in **(2)**, and a task supported by **(3)** - and writes a short story about Lina doing that task in that context, in the REDESIGNED system.

The scenario is not quoted from any one interview. It is the synthesis.

Lina's scenario, fully written.

Notice what each phrase is doing.

Scenario - Adding an elective between classes

It is 10:45 am on a Tuesday *and Lina has 15 minutes between her algorithms tutorial and her next class.* ***She opens the redesigned Ritaj on her phone.*** *Her dashboard already shows three suggested electives that fit her timetable. She taps the first, sees its current capacity ("28 of 30"), and adds it.* ***The system warns her that one elective in her plan now overlaps with the new one and offers to swap.*** *She accepts the swap, confirms, and gets a push notification 90 seconds later: "Advisor approved. Schedule updated." She closes the app and walks to her next class.*

WHAT EACH PHRASE DOES

10:45 am on a Tuesday

Sets temporal context. Fifteen minutes is a constraint on the design.

Opens on her phone

Sets device context. Small screen, thumb interaction, possibly walking.

Warns and offers to swap

The plot point - the moment the design has to handle a conflict gracefully.

Three phrases of context, one plot point. The scenario is 95 words. Most of what makes it good is what is NOT in it.

Four tests of a good scenario.

Apply these before you submit. If any answer is no, rewrite.

01 Can you point to the persona?

The protagonist must be a named persona from your Phase 1, with their voice and context. If you could replace the name and the scenario still works, it is not grounded.

02 Is the context specific?

Time, place, device, what else is happening. If it could happen 'sometime, somewhere', it is not a scenario - it is a flow.

03 Is there a plot point?

A moment of decision, conflict, or non-trivial choice. If the path is straight from start to finish, the design is not being asked to do anything interesting.

04 Could a teammate dispute it?

Specific enough that someone reading it could say 'no, that is not what Lina would do'. Disputability is what lets the scenario be improved.

These four tests are how your marker will read your Phase 1 scenarios. Apply them yourself before submission.

Carroll's move: from scenario to claims.

The reason requirements can feel disconnected.

SCENARIO (recap)

"She opens Ritaj on her phone… three suggested electives that fit her timetable… taps the first, sees its current capacity ("28 of 30")… system warns about a schedule overlap and offers to swap… confirms… push notification: "Advisor approved.""

Each phrase makes a claim about what the design must do.

"on her phone"

The design must work on small screens, with thumb interaction.

"suggested electives that fit her timetable"

The design must filter electives by schedule fit, automatically.

"its current capacity ("28 of 30")"

The design must show real capacity, not just "available".

"warns her about a schedule overlap"

The design must detect overlaps and surface them at decision time.

"offers to swap"

The design must propose concrete resolutions, not just flag problems.

"push notification: Advisor approved"

The design must complete the loop with the advisor asynchronously, fast.

Claims are the connective tissue. They become measurable requirements next.

PART TWO OF TWO

Requirements.

What your design must provide. Measurable. Testable. Traceable.

Requirements engineering's blind spot.

Why classical requirements engineering misses what HCI cares about.

SOFTWARE ENGINEERING TRADITION

"The system shall..."

Treats the system as the subject.

Behaviours, functions, data, performance. Inherited from Boehm, Sommerville, IEEE 830.

Example

"The system shall return search results in under 2 seconds."

Optimises for: buildability, testability against the system.

Misses: what the user is trying to achieve, in what context.

HCI TRADITION

"The user shall be able to..."

Treats the user as the subject.

Effectiveness, efficiency, satisfaction, in a specified context (ISO 9241-11, 2018).

Example

"A second-year student shall complete the add-elective task in under 4 minutes with at most 1 error."

Optimises for: testability against the user, in context.

What it commits to: the user's success is the metric, not the system's.

In Phase 1 you write requirements the HCI way. Use them in Phase 3 to test whether your design actually delivered.

Two kinds of requirement, two roles.

You will write mostly the non-functional kind for Phase 1.

FUNCTIONAL REQUIREMENTS

What the system must DO

Features, behaviours, actions. If absent, the system cannot do the task at all. Often derived from your scenarios as capabilities the design must provide.

Example: *"The system shall display current course capacity (e.g. 28 of 30) on every course card."*

NON-FUNCTIONAL REQUIREMENTS

What "well" looks like - the qualities of doing it

Performance, learnability, accessibility, error tolerance, satisfaction. These map directly to the six ISO 9241-11 usability goals (next slide). The Phase 1 rubric is asking for these specifically.

Example: *"A second-year student shall complete the add-elective flow in under 4 minutes with at most 1 error."* (Maps to **efficiency** - ISO 9241-11.)

The rubric's "measurable usability requirements" = non-functional requirements mapped to ISO usability goals.

Measurable means testable.

Map every requirement to one of the six ISO 9241-11 goals. If it doesn't map, rewrite it.

E Effectiveness

Can users complete the task at all?

MEASURE BY

task completion rate, error rate

F Efficiency

How fast / how much effort?

MEASURE BY

time on task, number of clicks

S Satisfaction

How does it feel?

MEASURE BY

SUS, Likert, qualitative quotes

L Learnability

How fast to become productive?

MEASURE BY

time-to-first-success, errors-on-first-use

M Memorability

After a week away, do they remember?

MEASURE BY

performance on second session vs first

R Errors/safety

How often, how serious, recovery?

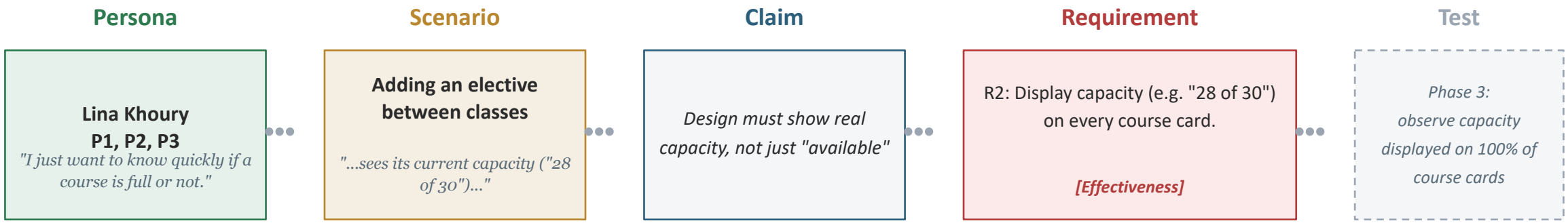
MEASURE BY

error frequency, severity, recovery time

Each Phase 1 requirement must name which goal it serves. 5-7 requirements; cover at least 4 of the 6 goals.

Traceability: the line back.

Every requirement points to a scenario, every scenario to a persona.



Your Phase 1 marker reads top-to-bottom: persona, scenarios, requirements.

If R2 has "(scenario: Adding an elective)" and "[Effectiveness]" in brackets, the marker can trace it back in one read. ***If not, they assume it came from nowhere - and mark accordingly.***

In Phase 3 the same trace lets you measure each requirement against the actual prototype. *Without the trace, Phase 3 becomes guesswork.*

Lina's seven requirements, traced.

From your handbook. Notice the ISO mapping covers 5 of 6 goals.

#	REQUIREMENT	ISO 9241-11 GOAL
R1	Users can complete the "add elective" flow in under 4 minutes with at most 1 error.	***Efficiency***
R2	The system displays current course capacity (e.g. "28 of 30") on every course card; never just "available".	***Effectiveness***
R3	When a schedule conflict is detected, the system offers a specific resolution within 1 second.	***Effectiveness***
R4	Users can return to and complete a registration session up to 24 hours later without re-doing earlier steps.	***Memorability***
R5	All primary actions are reachable on a 360px-wide screen without horizontal scrolling.	***Effectiveness***
R6	Error messages name the failed field and suggest a concrete fix (no generic "invalid input").	***Learnability***
R7	SUS score >= 70 across at least 3 think-aloud participants in Phase 3.	***Satisfaction***

Coverage: Effectiveness (3), Efficiency (1), Memorability (1), Learnability (1), Satisfaction (1). Errors-and-safety not covered - if the design has any risky actions, add an 8th requirement.

What the Phase 1 rubric is actually looking for.

Translated from rubric language into what you should do.

SCENARIOS

15 points

What the rubric says: *"Three scenarios describing how the personas would realistically use the redesigned system, written as short narratives."*

What it actually means:

Three goal scenarios (not task, not use). Realistic = particular. Specific = contextual. Connected = clear persona trace. Plot point in each. *Length: 80-150 words each.*

MEASURABLE USABILITY REQUIREMENTS

15 points

What the rubric says: *"5-7 requirements expressed as measurable targets, mapped to the six ISO 9241-11 usability goals."*

What it actually means:

5-7 non-functional requirements, each with:
(1) measurable metric with target value; (2) tagged ISO 9241-11 goal; (3) traceable back to a scenario (e.g. "from S2").
Cover at least 4 of the 6 goals. Include one SUS-based satisfaction requirement.

Combined: 30 of 100 Phase 1 points. Plus indirect contribution to coherence (5 points).

LIVE IN-CLASS EXERCISE

Now you write one.

Phones out. 15 minutes. Two rounds.

Join the live session.

Scan the code, choose a short alias, you're in.

[QR CODE]

(generated and inserted at lecture time)

Or go to:

ihshaish.github.io/birzeit/live

Two rounds, 12 minutes total.

Round 1: write a scenario. Round 2: write a requirement from it.

ROUND 1 - 5 MINUTES

Write a scenario.

PERSONA: Layla (engaged power user, banking app).

SITUATION: Layla just got paid on the 5th and needs to send 800 NIS to her landlord by tomorrow.

Write a 2-3 sentence goal scenario that captures:

- the temporal context
- the device
- a plot point (something that could go wrong, or a decision)
- a concrete outcome

Submit on your phone. We will vote on three live.

ROUND 2 - 5 MINUTES

Write a requirement.

We pick the winning scenario from round 1. From that scenario, write ONE measurable requirement:

- a specific metric (number, time, threshold)
- tag the ISO 9241-11 goal it serves
- keep it under 30 words

Example shape:

Layla shall complete the transfer in under 90 seconds with at most 1 confirmation tap. [Efficiency]

Submit and we vote on whether it is genuinely measurable.

Timer starts when I advance the live page. The same page on your phone will count down.

Recap.

01 The chain is the point

Persona to scenario to requirement to test. Break any link and Phase 1 loses marks.

02 Scenarios are stories with plot points

Particular, contextual, narrative, disputable. Carroll's tradition, not requirements engineering's.

03 Claims bridge scenarios to requirements

The connective tissue most teams skip. Every meaningful phrase in your scenario makes a claim about what the design must do.

04 Requirements are user-centred, measurable, ISO-mapped

5-7 non-functional, covering >= 4 of 6 ISO 9241-11 goals. Numbers, not adjectives.

NEXT SESSION: Back to textbook order - "What is interaction design?". Then Chapter 2 - Norman's models.

References.

Cited in this session. Cite these when drawing on their ideas in your reports.

Carroll, J. M. (2000). *Making Use: Scenario-Based Design of Human-Computer Interactions.* MIT Press. The canonical text on scenario-based design. Chapters 2-4 develop the framework used in today's lecture.

Carroll, J. M., & Rosson, M. B. (1992). *Getting around the task-artifact cycle: how to make claims and design by scenario.* ACM Transactions on Information Systems, 10(2), 181-212. The original claims-analysis paper.

Cooper, A., Reimann, R., Cronin, D., & Noessel, C. (2014). *About Face: The Essentials of Interaction Design, 4th edition.* Wiley. Chapter 4 distinguishes goal, task, and use scenarios.

ISO 9241-11:2018. . *Ergonomics of human-system interaction - Part 11: Usability: Definitions and concepts.* International Organization for Standardization. The canonical six usability goals.

Rogers, Y., Sharp, H., & Preece, J. (2024). *Interaction Design: Beyond Human-Computer Interaction, 6th edition.* Wiley. Chapter 10 on establishing requirements; Chapter 11 on design, prototyping and construction.

Sommerville, I. (2016). *Software Engineering, 10th edition.* Pearson. Chapter 4 on requirements engineering from the SE tradition - useful for contrast.

Nielsen, J. (2012). *Usability 101: Introduction to Usability.* Nielsen Norman Group. Practical operationalisations of the ISO usability goals.