

UNIT 7: DESIGN EVALUATIONS · 6 HOURS

Design Evaluations

Design evaluations in UI/UX are structured processes used to measure how easily, efficiently, and satisfyingly users interact with a product.

Unit 7 Syllabus & Learning Roadmap

Official Tribhuvan University BCA 2nd Semester curriculum breakdown (6 Hours):

Section	Curriculum Topic	Core Practical Skills & Interactive Demos
7.1	Formative and summative evaluation	Timing & purpose · Chef's soup metaphor · Cost curve · [Demo] Mode Lens & Scenario Classifier
7.2	Usability testing: Moderated vs Unmoderated	Facilitator roles · Think-Aloud protocol · Remote vs Lab · [Demo] Live Testing Simulator & Advisor
7.3	Analyzing test results & gathering insights	Qual vs Quant · Affinity maps · Nielsen 0–4 Severity · [Demo] Findings Analyzer & 2×2 Matrix
7.4	Expert analysis, user participation, iterative & paradigms	Cognitive walkthrough · Nielsen's 5–user law · 4 paradigms · [Demo] 5–User Curve Simulator
7.5	DECIDE evaluation framework, heuristic evaluation	6–step DECIDE roadmap · 10 Nielsen Heuristics · [Demo] 10 Heuristics Inspector & Audit Lab
7.6	Task analysis and performance metrics	HTA tree · ISO 9241–11 · Completion % · Time · [Demo] Real Working 10–Item SUS Calculator

What is Design Evaluation? & Why We Evaluate

Design evaluations in UI/UX are structured processes used to measure how easily, efficiently, and satisfyingly users interact with a product.

1. Easily (Effectiveness)

Can users learn the interface and accomplish their tasks accurately without errors or roadblocks?

2. Efficiently (Speed & Effort)

How quickly and with how few steps, clicks, and mental cognitive effort can goals be reached?

3. Satisfyingly (Experience)

Does the interaction feel comfortable, intuitive, trustworthy, and aesthetically pleasing?

Boehm's 100x Cost Curve

Economic Value

Fixing a flaw on a wireframe costs \$1. Fixing it in code costs \$10. Fixing it after release costs \$100+ in lost users and emergency patches!

The Core Rule

Psychology

"You are NOT your user." Developers understand system logic; real users act on their own mental models, habits, and assumptions.

Formative vs. Summative Evaluation

Feature	Formative Evaluation	Summative Evaluation
When it happens	Early and continuously throughout the design and iteration phase.	At the end of a development cycle, right before or after launch.
Primary Goal	To find what is broken, why it fails, and how to fix it.	To measure how well the final design performs against benchmarks or competitors.
Data Type	Mostly qualitative (user frustrations, verbal feedback, behavioral observations), though can be quantitative.	Strictly structured and quantitative (task completion rates, time-on-task, error rates).
Sample Size	Small (often 3 to 5 users per testing cycle).	Large and representative (15–20+ users for statistical reliability).
Methods	Wireframe testing, cognitive walkthroughs, heuristic evaluations, and think-aloud protocols.	Benchmark testing, unassisted task performance, and standardized satisfaction surveys.
Audience	The internal design and product team.	Stakeholders, executives, or external regulators.

7.2 Usability Testing

Usability Testing involves real users interacting with your product while the product designers observe and gather feedback. Usability testing helps you pinpoint problem areas and drive user-centered design for an accessible end product that works effectively.

When to Do Usability Testing:

Stage 1

Before you start designing

Test competitor products or existing systems to uncover real user pain points, baseline habits, and unmet needs.

Stage 2

Once you have a wireframe or prototype

Validate information hierarchy, navigation flows, and user mental models early when design changes are inexpensive.

Stage 3

Right before launch

Evaluate the near-final product against benchmarks to ensure critical flows succeed with zero catastrophic blockers.

Stage 4

Regularly, after launch

Continuously monitor UX health, test newly released features, avoid regressions, and adapt as user expectations change.

7.2 Moderated vs. Unmoderated Testing

The two major operational formats for running usability tests:

Human Facilitator Present



Moderated Usability Testing

A trained facilitator actively guides the participant, provides task instructions, answers questions, and asks follow-up probes in real time (in-person or via live video call).

- Pros: Can unblock stuck users, ask deep probing questions ("What did you expect to happen when you tapped that?"), reads micro-expressions and hesitation.
- Cons: Expensive, time-consuming, potential facilitator bias, small sample sizes (5–8 participants).
- Best for: Early prototypes, complex enterprise software, novel workflows.

Automated / Self-Guided



Unmoderated Usability Testing

Participants complete pre-configured tasks independently on their own devices using automated software (e.g., Maze, UserTesting, Lookback) with zero human intervention.

- Pros: Highly scalable (50–100+ users in 24 hours), cheaper, natural home environment, zero observer bias.
- Cons: Cannot probe deeper into "why", participants may misunderstand tasks or abandon if stuck, lower qualitative depth.
- Best for: Statistical validation, simple e-commerce flows, measuring time-on-task.

7.2 Testing Formats & Decision Matrix

Combining Moderation (Moderated vs Unmoderated) with Location (In-Person vs Remote):

Dimension	In-Person Moderated	Remote Moderated	Remote Unmoderated
Environment	Usability lab, college lab, conference room.	Participant’s home/desk via Zoom, Meet, Teams.	Participant’s own device at their own time.
Facilitator Interaction	Direct face-to-face dialogue, full body language.	Voice & screen-share interaction over video.	Zero live interaction; automated task prompts.
Equipment Needed	Lab cameras, dedicated PCs, mobile test rigs.	Webcam, microphone, screen recording tool.	Automated testing SaaS platform (Maze, Useberry).
Geographic Reach	Local only (must travel to lab).	Global / International participants.	Global / Async 24-hour testing.
Cost & Turnaround	Highest cost (\$\$\$), 2–3 weeks.	Medium cost (\$\$), 1 week.	Lowest cost (\$), 24–48 hours.

7.3 Analyzing Test Results: Qual vs. Quant Data

Effective UX analysis combines Qualitative Insights (understanding why users struggle) with Quantitative Metrics (measuring how much users struggle).

Qualitative Data (The "Why")

Observational & Descriptive

- Verbal Quotes: "I didn't realize I had to scroll to see the payment options."
- Behavioral Signals: Frowning, squinting, frantic clicking, cursor hesitation, backtracking.
- Mental Model Misalignments: Expecting "Search" to filter within a page instead of global database.
- Collection Method: Notes, screen recordings, think-aloud transcripts.

Quantitative Data (The "What / How Much")

Numerical & Statistical

- Task Completion Rate: 72% of participants completed checkout successfully.
- Time on Task: Average 48 seconds to find the syllabus PDF.
- Error Frequency: Average 2.4 misclicks per participant.
- Standardized Scales: System Usability Scale (SUS) score = 74.5/100.

7.3 The 5–Step UX Data Analysis Workflow

From raw observations to engineering action items:

Step 1

1. Log Data



Compile timestamped notes, quotes, video timestamps, and task completion records.

Step 2

2. Affinity Map



Group individual observations into thematic clusters: Navigation, Forms, Content, Visual hierarchy.

Step 3

3. Severity Rate



Assign Jakob Nielsen's 0–4 severity scale based on frequency, impact, and persistence.

Step 4

4. Formulate Fix



Translate findings into actionable recommendations (Problem + Evidence + Proposed Design Fix).

Step 5

5. Prioritize



Plot recommendations onto an Impact vs. Effort 2x2 matrix to schedule immediate development.

7.3 Jakob Nielsen's 5-Point Severity Rating

Severity Rating: Allocates limited development resources to fixing the most damaging usability flaws first. Governed by 3 factors: Frequency (how common?), Impact (how hard to overcome?), and Persistence (does it repeat?).

Level	Severity Classification	Description & User Impact	Development Action
0	Not a Usability Problem	Participant personal preference or stylistic critique with no functional impact.	Do not fix.
1	Cosmetic Problem Only	Minor visual inconsistency (e.g., button margin off by 4px). Does not hinder task.	Fix only if extra time is available before release.
2	Minor Usability Problem	Causes slight hesitation or minor delay, but users easily recover on their own.	Low priority; assign to normal sprint backlog.
3	Major Usability Problem	Significantly impairs task completion; many users get confused or need assistance.	High priority; must be fixed before final product release.
4	Usability Catastrophe	Complete blocker! Users cannot complete their primary goal (e.g., checkout crash).	Emergency Blocker! Mandatory fix before any deployment.

7.4 Expert Analysis vs. User Participation

Evaluation methods divide into two broad paradigms: Inspection Methods (conducted by UX experts without real users) and Empirical Methods (conducted with actual representative end users).

Inspection Paradigm

1. Expert Analysis

- Core Methods: Heuristic Evaluation, Cognitive Walkthrough, Pluralistic Walkthrough.
- Who Evaluates: 3 to 5 HCI / UX usability experts.
- Advantages: Fast, highly cost-effective, requires no participant recruitment or incentives, catches up to 75% of obvious layout and heuristic flaws.
- Limitations: Experts are not real end users; risks expert bias, misses obscure real-world workflows, and produces false alarms.

Empirical Paradigm

2. User Participation

- Core Methods: Usability testing, contextual inquiry, field studies, focus groups, interviews.
- Who Evaluates: Real representative target users (e.g. students, doctors, shoppers).
- Advantages: Uncovers true behavioral patterns, genuine emotional reactions, and real-world task context.
- Limitations: Slower, requires recruitment budget, logistics, scheduling, and ethics oversight.

7.5 The DECIDE Evaluation Framework

DECIDE Framework: Developed by Jenny Preece, Yvonne Rogers, and Helen Sharp. A systematic 6–step checklist to guide practitioners in planning, executing, and analyzing design evaluations.

Step	DECIDE Stage	Key Questions & Practical Actions
D	Determine the overall goals	Why are we evaluating? (e.g. improve mobile checkout conversion by 20%).
E	Explore specific questions	Break goals into testable questions (e.g. Can users locate the voucher box?).
C	Choose evaluation methods	Select paradigms & techniques (Heuristic inspection + 6 lab usability tests).
I	Identify practical issues	Recruit target participants, prepare facilities, equipment, and budget.
D	Decide on ethical issues	Informed consent forms, data anonymity, right to withdraw without penalty.
E	Evaluate, analyze & present	Filter data, compute task success/SUS, prioritize fixes for developers.

7.6 Task Analysis: Hierarchical Task Analysis (HTA)

Task Analysis: The formal study of how users perform tasks to achieve their goals.

Hierarchical Task Analysis (HTA): Decomposes a high-level goal into a hierarchy of sub-tasks and elementary operations, governed by conditional Plans.

HTA Structure: Order Momo on Mobile App

0. Goal: Order Momo Online

1. Locate & Select Restaurant

→ 1.1 Open app & search 'Momo'

→ 1.2 Filter by top rating (>4.5★)

2. Customize Item & Add to Cart

→ 2.1 Choose C-Momo vs Steamed

→ 2.2 Select spice level (Spicy)

3. Checkout & Payment

→ 3.1 Confirm delivery address

→ 3.2 Pay via digital wallet (eSewa/Khalti)

Plan Specification

The Rules Governing Order:

Plan 0: Do 1 → Do 2 → Do 3 in sequence.

Plan 2: Do 2.1; if custom addon required, do 2.2; then tap Add to Cart.

Plan 3: If address saved, do 3.2; else do 3.1 first.

Why Conduct Task Analysis?

- Pinpoints cognitive bottlenecks and unnecessary friction steps.
- Provides the blueprint for writing realistic usability test tasks.
- Identifies where users make frequent slips or navigation errors.

7.6 Usability Performance Metrics: ISO 9241-11

ISO 9241-11 Standard: Usability is measured across 3 fundamental pillars:

1. Effectiveness (Can users succeed?) · 2. Efficiency (At what cost/time?) · 3. Satisfaction (How did it feel?).

Pillar 1

Effectiveness

Accuracy and completeness with which users achieve goals.

Task Success Rate:
 $(\text{Successes} / \text{Total Tasks}) \times 100\%$

Benchmark: Global average task completion rate is ~78% across enterprise software.

Pillar 2

Efficiency

Resources expended in relation to accuracy and completeness.

- Time on Task: Seconds to complete.
- Task vs Error Time: Productive work vs struggle time.
- Error Rate: Unintentional slips & invalid form submissions.

Pillar 3

Satisfaction

Freedom from discomfort and positive attitudes towards use.

- SUS (System Usability Scale): 10-item standard (avg 68).
- SEQ (Single Ease Question): 1-7 scale after each task.
- CSAT & NPS: Customer satisfaction and recommendation index.

7.6 System Usability Scale (SUS) Deep Dive

System Usability Scale (SUS): Created by John Brooke in 1986. A standardized, technology-agnostic 10–statement questionnaire answered on a 5–point Likert scale (1 = Strongly Disagree to 5 = Strongly Agree).

Brooke's Alternating Scoring Formula

How to Calculate a SUS Score:

Questions alternate between positive (odd) and negative (even) phrasing to combat acquiescence response bias:

- Odd Questions (1, 3, 5, 7, 9): Contribution = Score - 1
- Even Questions (2, 4, 6, 8, 10): Contribution = 5 - Score
- Final Calculation: Sum all 10 contributions and multiply by 2.5:

SUS Score = (Sum of Contributions) × 2.5

SUS Benchmarks & Grading Curve

Interpreting the 0–100 Score:

⚠ Critical Exam Note: A SUS score is a percentile ranking, NOT a percentage!

SUS Score	Letter Grade	Usability Meaning
80.3 – 100	Grade A	Excellent / Users will advocate
68.0 – 80.2	Grade B/C	68 is Industry Average!
51.0 – 67.9	Grade D	Marginal / High friction
0.0 – 50.9	Grade F	Catastrophic UX failure

7.6 [Interactive Lab] Live Usability Metrics & SUS

Calculate Task Success Rate & compute real System Usability Scale (SUS) scores dynamically:

Pillar 1: Effectiveness

Task Success Rate Calculator

Attempted Tasks:

Succeeded Tasks:

20

16

Success Rate:

80%

Acceptable (Near Industry Average)

Pillar 3: Satisfaction

Brooke Algorithm

Live SUS Instrument Simulator

1. I'd like to use this system frequently:

☐

1

☐

3

☒

5

2. I found the system unnecessarily complex:

☒

1

☐

3

☐

5

3. I thought the system was easy to use:

☐

1

☐

3

☒

5

4. I'd need technical support to use this:

☒

1

☐

3

☐

5

Computed
SUS Score:

70

Grade C (Average / Acceptable)

System Usability Scale benchmark standard is 68.
Usable, but has noticeable friction.

Summary: Golden Rules of Design Evaluation

BCA exam quick-reference study notes:

5 Essential Do's

- 1. Evaluate Early & Formatively: Catch layout flaws on paper wireframes before code is written.
- 2. Test 5 Users Iteratively: 3 iterative rounds of 5 users beat 1 round of 15 users every time.
- 3. Follow Think-Aloud Protocol: Let users verbalize their mental model without interruption.
- 4. Use the DECIDE Framework: Ensure goals, questions, logistics, and ethics are rigorously planned.
- 5. Respect Ethics: Obtain informed consent and assure participants the software is on trial, not them.

5 Critical Don'ts

- 1. Don't Lead the User: Never say "Can you see that nice blue checkout button?" Give neutral goals.
- 2. Don't Rely Solely on Opinions: Trust observational behavior (DO) far more than subjective praise (SAY).
- 3. Don't Defend Your Design: If a user fails to find the menu, accept the failure and fix the menu.
- 4. Don't Confuse SUS with %: A SUS score of 68 is average (Grade C), not failing!
- 5. Don't Skip Severity Rating: Prioritize fixes based on frequency and impact rather than fixing cosmetics.

UNIT 7 COMPLETED

Design Evaluations Complete!

← Unit 6: Wireframing & Prototyping

↺ Restart Unit 7 Slides