| | |
|---|---|
| Reg. NO. | 212223220105 |
| NAME. | Shanmugakarthik. |
| Subject | UI and UX Design. |
| Slot | L4X3-4. |
| DATE. | 19/8/2025 |

Hands-on UI/UX Experiments

a. Explore the fundamentals of UI and UX.

⇒ Core definitions

* UX (user Experience): End-to-end experience a person has with a product/service: usefulness, ease efficiency, delight.

* UI (user Interface): The visual/interactive layer user touch: layout, colors, typography, components, states.

⇒ UX building blocks.

* Research: interviews, surveys, contextual inquiry, analytics review.

* synthesis: Personas, jobs-to-be-done, problem statements, journey maps.

* IA & flows: information architecture, navigation patterns, task flows.

* Interaction Design: states, feedback, error handling, micro-interactions.

* Prototyping: low-fi (paper/wireframes) → hi-fi (interactive).

* Testing & Iteration: usability tests, A/B tests, heuristic evaluations.

=> UI building blocks.

* visual hierarchy: size, weight, spacing to guide attention
* Typography: 2-3 type scales (eg. 12/14/16/20/24/32), line-height 1.4-1.6
* Color & Contrast: primary, neutrals, semantic (success/warn/error); aim contrast ≥ 4.5:1 for body text.
* Layout & Grids: 4/8px spacing system; 12-Column grid for web, 4/8 columns for mobile.
* Components & states: buttons (default/hover/active/disabled), inputs, cards, modals, toasts.
* Accessibility: Keyboard navigation, focus order, labels/alt text, motion sensitivity.

=> quick reference: Common UX heuristics

* visibility of system status
* user Control & freedom.
* Error prevention.
* flexibility & efficiency.
* Help & documentation.
* Consistency & standards.
* Match between system & real world
* Recognition rather than recall.

Differentiate between UI and UX using real-world case studies.

=> Case study template (use for any product).

* Context: product, audience, primary jobs-to-be-done.

* UI Assessment: Color system, typography, spacing, component consistency density.

* UX Assessment: discoverability, navigation clarity, task steps, feedback, error handling, Performance.

* findings: what works vs what hurts.

* Opportunity List: quick wins (1-2 weeks) vs. strategic (quarter+).

* Metrics to move: conversion, activation, retention, task success, CSAT/SUS.

=> Example 1: Food delivery APP (e.g. Zomato / Swiggy).

* UI: appetizing imagery, prominent CTAs, clear price & rating chips, use of cards for scannability.

* UX: search → filter → item → customize → cart → pay → track; live ETA; reorder shortcuts.

* UI win: badges (veg / Egg), Price clarity and Card

* UX win: Saved addresses & Payment; order tracking map; retry payment flows.

=> Example 2: Ride-hailing (e.g. uber/ola)

* UI: map centric, bottom sheet for options, large tap targets.

* UX: Pickup confirmation, fare estimate, wait time, driver tracking, cancellation flow.

* Common friction: Ambiguous Pickup Points, solution; landmark Picker and text to driver.

=> Example 3: E-Commerce (eg. Amazon/ FlipKart).

* UI: Dense info on Product Pages; Stores, Price, offers, delivery dates.

* UX: 1-Click/ quick buy, persistent Cart, address & Payment vault, returns Policy Clarity.

* Trade-off: high info density (UI) vs faster decision making (UX) for power shoppers.

Illustrate the relationship between UI and UX in design.

⇒ How they fit together

* UX defines: Problems, audience, journeys, flows, requirements / Content priorities.

* UI Expresses: these decisions via visual language & interaction patterns.

* feedback loop: UI choices reveal new UX issues (e.g. button label truncates → rethink copy IA).

⇒ Flow → Screen mapping (example: checkout)

| How step (UX) | user goal | Design implication (UI). |
|---|---|---|
| Cart Review | verify items/price | Emphasize totals, editable quantity, lightweight images. |
| Address | Confirm destination. | Input masks, address suggestions, map picker, progress stepper. |
| Payment | Choose & confirm. | Tokenized saved methods, default selection, error states, trust badges. |
| Review | final check. | order summary, delivery window, terms, primary CTA prominent. |

| | | |
|---|---|---|
| Confirmation | Clear next steps. | Receipt, tracking link, share button, support access. |

=> Good/Bad Combinations.

* Good UI + Bad UX: Beautiful form but 6 steps, confusing validation.

* Bad UI + Good UX: Cluttered visuals but short; forgiving flow.

* Ideal: Short flows, clear feedback, accessible visuals, Consistent patterns.

Demonstrate various UI/UX design tools in detail.

⇒ Figma (recommended for both UI & UX).

Core Skills to Practice.

* Auto Layout: responsive buttons/cards/lists; Padding & gap; min/max constraints

* Components & variants: Create Button with size (S/M/L) + states; use properties for icons & labels.

* Styles & Tokens: set color styles (primary/secondary/semantic), text styles (H1-Body-Caption)

* Grids & spacing: apply 8px spacing; 12-col grid (web) or 4/8-col (mobile).

* Prototyping: link frames, overlay modals, scrolling containers, interactive components (toggles, switches).

* Design systems: publish libraries, usage guidelines, naming conventions.

* Dev Handoff: Inspect Panel, CSS/iOS/Android code snippets, redlines.

⇒ Figjam / Miro (collaboration & research)

* use for: brainstorming, affinity mapping, journeys, architecture, prioritization.

* Templates: Crazy 8s, Impact/Effort Matrix,

Priority Pyramids, Card sorts.

⇒ Prototyping Animations( Principle, Pratopie, figma smart Animate).

* Build micro-interactions: PWI - to refresh Card swipe, success checkmark.

* validate feel/timing before engineering effort.

⇒ Research & Testing tooling.

* Planning: research plan, screener, Consent.

* Remote testing: unmoderated tasks with Maze/useberry; surveys with forms/ Typeform.

* Instrumentation: analytics events for key actions; heatmaps/ session replay (eg. Hotjar).

* Accessibility Checks: Axe DevTools, Light house, Screen ~~replay~~ readers (NVDA/ Voice over), keyboard-only Pass.

⇒ Handoff & front-end Alignment.

* Design tokens → map to Code (CSS variables).

* Story book for component documentation; zeplin for specs if needed.