

Screen Sizes, Responsive Design & Interactive Elements

Designing fluid digital interfaces across devices, touch ergonomics, responsive typography, and tactile UI controls.

BCA 2nd Semester

UI/UX Design

Interactive Slides

Screen Size

Physical vs. Logical Viewport

Screen Size in UI/UX design refers to the physical dimensions and digital resolution (measured in pixels) of display screens across various devices. In UI/UX, screen size matters because the same design needs to work seamlessly across different devices and screen dimensions.



Mobile Phones

- Touch-first navigation
- One-handed thumb zone
- Vertical portrait ratio

58%+ Web Traffic



Tablets & Foldables

- Hybrid touch & keyboard
- 2-column adaptive grids
- Frequent screen rotations

Multi-Orientation



Laptops

- Mouse/trackpad precision
- Interactive hover states
- Multi-column navigation

Dense Information



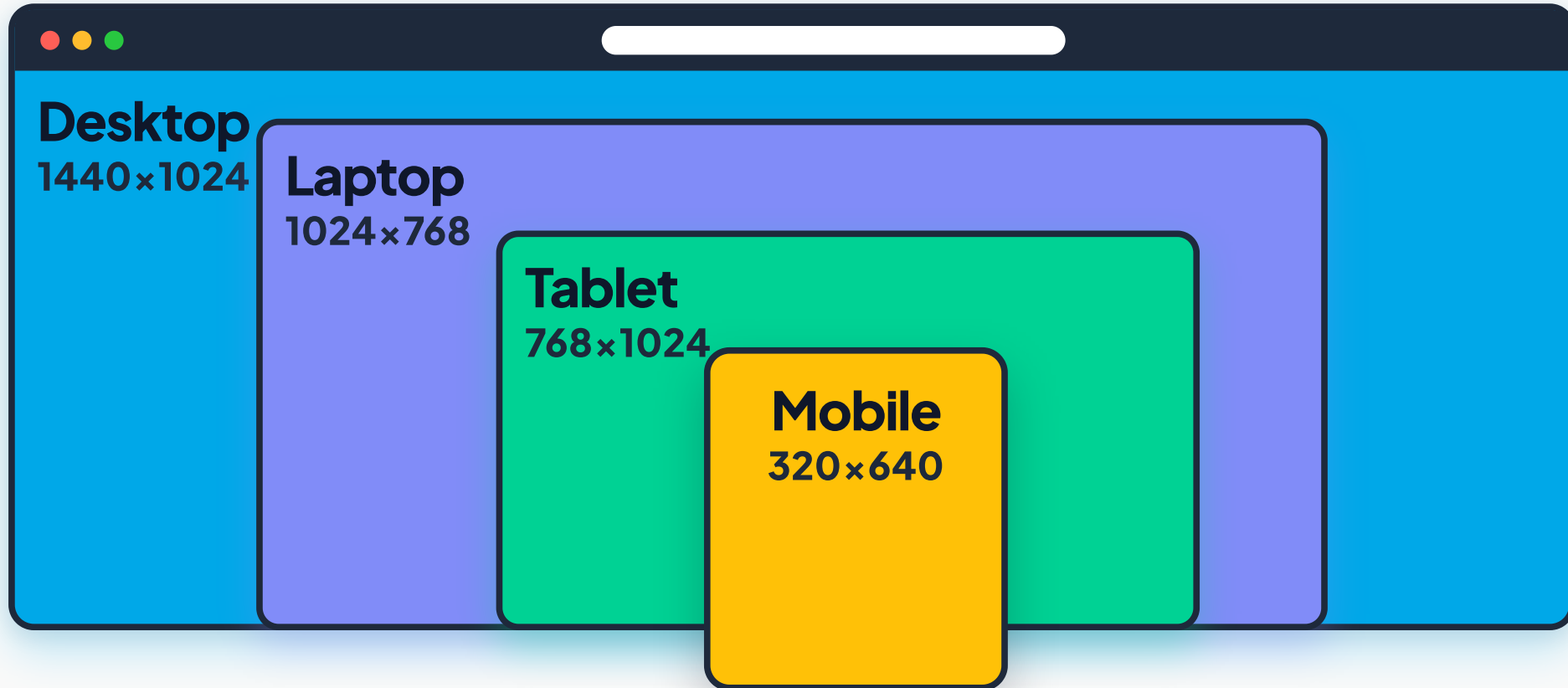
Desktop & Ultra-wide

- Requires max-width limits
- Prevents overly long lines
- Optimal reading comfort

Max Content Container

Common screen sizes

Viewport Scale Comparison



Desktop ·
1440×1024
Max Content
Container

Laptop ·
1024×768
Multi-Column
Layout

Tablet ·
768×1024
Dual-Orientation
Mode

Mobile ·
320×640
One-Hand Touch
Zone

Physical Pixels vs. CSS Pixels

Hardware vs. Reference Dimension

A **Physical Pixel (Hardware Pixel)** is the smallest physical light-emitting element on a display screen. A **CSS Pixel (Reference Pixel)** is an abstract unit of measurement defined by the W3C as 1/96th of an inch at arm's length.

Device Pixel Ratio (DPR)

To make text crisp on high-density displays (Apple Retina, Super AMOLED), operating systems bundle multiple physical pixels into a single logical CSS pixel.

$$\text{DPR} = \frac{\text{Physical Device Pixels}}{\text{CSS Reference Pixels}}$$

- Standard Screen (1x): 1 CSS px = 1 Physical px
- Retina Display (2x): 1 CSS px = 2x2 = 4 Physical px
- Modern Smartphone (3x): 1 CSS px = 3x3 = 9 Physical px

Real-World Example: iPhone 14 Pro

Property	Hardware Spec	CSS Reality
Width	1,179 physical px	393 CSS px
Height	2,556 physical px	852 CSS px
Pixel Ratio	DPR = 3.0 (3x scale)	

⚠ Critical Rule for UI Designers: **Never design mockups using 1179 physical pixels!** Always design at 393px (1x logical points) so fonts and buttons are sized to real human fingertips.

What is Responsive Web Design?

Core Philosophy & 3 Pillars

Responsive Web Design (RWD) is a web development approach that makes web pages render smoothly and remain fully usable across all screen sizes, from mobile phones to ultra-wide desktop monitors.

Why is a responsive website important?

5 Core Benefits

- Enhances user experience
- Improves efficiency
- Boosts conversions
- Ensures accessibility
- Decreases costs



1. Fluid Grid Layouts

Percentages, Flex & Grid

Elements are sized in proportional units (`%`, `vw`, `fr`) rather than rigid pixels, allowing layouts to compress and expand fluidly.



2. Flexible Media

Fluid Image Constraints

Images, SVGs, and video containers scale within their parent bounding boxes using `max-width: 100%` and `height: auto` to prevent overflow.



3. CSS Media Queries

Breakpoint Triggers

Conditional CSS blocks (e.g. `@media (min-width: 768px)`) that trigger layout transformations when viewport thresholds are crossed.

Responsive vs. Adaptive vs. Mobile-First

Strategy	How It Works	Pros & Trade-offs
Responsive Design (RWD)	Single fluid layout that continuously stretches and contracts across all infinite widths between breakpoints.	Industry Gold Standard Flawless on any custom window or foldable device; requires thoughtful CSS architecture.
Adaptive Design (AWD)	Detects specific device widths (e.g. 320, 768, 1024) and loads distinct static layouts. "Jumps" between states.	Legacy / Phasing Out Easier to bolt onto legacy desktop sites; awkward on in-between tablet or folding screens.
Mobile-First Strategy	Author base styles for the smallest mobile screen first, then layer enhancements using <code>min-width</code> media queries.	Modern Best Practice Faster mobile loading, clean progressive enhancement, zero desktop CSS bloat on phones.

💡 **Why Mobile-First CSS?** Coding desktop-first (with `max-width`) requires writing complex code and then continuously *undoing* properties (e.g. `display: none`, resetting margins) on mobile. Mobile-first starts clean

Fluid Typography: CSS `clamp()`

Mathematical Scaling

Fluid Typography uses mathematical viewport functions to smoothly scale text between a defined minimum and maximum size as the screen resizes, eliminating jarring font-size jump steps at arbitrary breakpoints.

LIVE CALCULATOR

Simulate Viewport:

900px

Rendered Heading at Current Viewport:

Responsive UI Design

Computed Size: **30px (1.88rem)**

```
font-size: clamp(18px, 12px + 2vw, 36px);
```

Scaling Fluently

Anatomy of `clamp(MIN, VAL, MAX)`

- 1. Minimum (18px):** Floor limit. On small phones (320px), text will never shrink smaller than 18px.
- 2. Preferred (12px + 2vw):** The dynamic scaling formula. As viewport expands, `2vw` increases by 2px per 100px screen width.
- 3. Maximum (36px):** Ceiling limit. On ultra-wide 4K monitors, text will never blow up into an absurd billboard font.

✨ Also works on: `padding: clamp(1rem, 4vw, 3.5rem);` and `gap: clamp(8px, 2vw, 24px);`

Responsive Media & Images

Bandwidth & CLS Prevention

Responsive Media ensures images and video players adapt in resolution, file weight, and artistic framing without causing **Cumulative Layout Shift (CLS)**.

Resolution Switching: `srcset`

Sends lightweight compressed images to mobile 3G connections and 4K assets only to desktop monitors:

```

```

Browser picks optimal asset based on DPR and viewport width.

Art Direction: `<picture>`

Changes the *cropping and framing* of the image so a landscape photo crops to a close-up portrait on mobile:

```
<picture>
  <!-- Mobile: Close-up square crop -->
  <source media="(max-width: 640px)" srcset="crop-square.jpg">
  <!-- Desktop: Full wide panoramic -->
  
</picture>
```

 **Prevent Layout Shift (CLS):** Always declare `aspect-ratio: 16 / 9;` or explicit `width` & `height` so space is reserved.

Responsive Navigation Patterns

Desktop to Mobile Thumb Zone

Navigation architecture must adapt from wide horizontal screen space to constrained vertical thumb reaches.



Horizontal Navbar

Desktop ($\geq 1024\text{px}$)

All primary links exposed in top row. Highest visibility, zero clicks required to discover sections.



Hamburger Drawer

Mobile ($< 768\text{px}$)

Hides secondary links inside an off-canvas drawer. Saves screen space but reduces link discovery.



Bottom Tab Bar

Mobile Apps & PWAs

3 to 5 core destinations pinned to the bottom thumb zone. 80%+ higher engagement than hidden menus.



Priority+ Menu

Adaptive Navigation

Shows as many links as fit the current width, and moves excess items automatically into a "More ▼" dropdown.

What are Interactive Elements?

UI Controls & Perceptible Feedback

An **Interactive Element (UI Control)** is any user interface component that accepts user input, commands, or gestures (clicks, taps, typing, swiping) and provides immediate perceptible visual and system feedback.

1. Action Controls

Trigger immediate system actions, form submissions, or view navigation.

Buttons

Hyperlinks

FABs

Icon Buttons

2. Input & Selection

Collect user preferences, text, numeric values, or boolean toggles.

Text Inputs

Checkboxes

Radio Buttons

Switches

3. Feedback & Overlays

Confirm interactions, guide workflows, and communicate system status.

Modals

Bottom Sheets

Toasts

Spinners

The 6 Essential States of UI Controls

Interactive State Lab

Interactive elements are not static pictures. They must communicate status through 6 distinct interactive states:

LIVE STATE LAB

1. Default
(Rest)

2. Hover
(Pointer)

3. Active
(Pressed)

4. Focus
(Keyboard)

5. Loading
(Busy)

6. Disabled
(Inactive)

1. Default / Rest State

Submit Order

john@kbc.edu.np



```
/* Rest State */
.btn-primary {
  background: #2563eb;
  color: #ffffff;
  box-shadow: 0 1px 3px rgba(37,99,235,0.3);
  cursor: pointer;
}
```

Touch vs. Mouse Pointer Interaction

Ergonomic Divergence

Interaction Factor	Desktop Mouse / Trackpad	Mobile Touchscreen
Input Precision	Sub-pixel precision (1×1 px pointer cursor). Can click microscopic 12px links.	Fingertip pad is 10–11 mm wide (approx. 44–48 CSS pixels). Highly imprecise.
Hover State	Full <code>:hover</code> support. Can inspect tooltips and drop-down previews.	Zero hover! Tapping triggers click immediately. Hover menus break!
Screen Occlusion	Tiny cursor arrow leaves 99.9% of the screen fully visible.	The user's hand and palm physically block the bottom half of the screen.
Gestures	Click, double click, right-click context menu, scroll wheel, drag & drop.	Tap, double-tap, swipe, flick, pinch-to-zoom, pull-to-refresh, long press.

Action Controls: Button Hierarchy

Emphasis & Decision Speed

Visual hierarchy prevents decision paralysis. A well-designed screen has only **one primary action** at any given time.

Primary Action

Primary (High)

Solid Saturated

Reserved for the main goal of the screen (e.g. "Save", "Checkout", "Publish"). Max 1 per view.

Secondary Action

Secondary (Med)

Outlined / Light Fill

Alternative choices alongside the primary goal (e.g. "Filter", "Export PDF", "Back").

Ghost / Text

Tertiary (Low)

Border-free Text

Subtle non-distracting actions (e.g. "Cancel", "Learn more", "Skip intro").

Delete Project

Danger (Destructive)

Red Warning Tone

Irreversible or dangerous actions (e.g. "Delete account"). Requires modal confirmation.

 **The Button vs. Link Rule:** Use a `<button>` when triggering a system operation, submitting data, or toggling UI. Use an `<a href="...">` only when navigating to a new URL destination.

Selection Controls: Radio - Checkbox - Switch

Input Patterns & Mental Models

Choosing the correct selection control matches user expectations for **single selection**, **multiple selection**, or **instant state toggle**.

☐ Radio Button

Mutually Exclusive (1 of N)

Use when the user must pick **exactly one** option from a visible list of 2 to 5 items (e.g. "Shipping Method: Standard / Express").

Selection cannot be unselected back to empty without reset.

☒ Checkbox

Multiple or Binary Choice

Use for picking **zero, one, or multiple** independent items from a list, or for binary consent (e.g. "I accept the Terms & Conditions").

Typically requires clicking a "Save" or "Submit" button.

☒ Toggle Switch

Instant State Change

Use for physical on/off settings that take **immediate effect** without requiring a "Submit" or "Save" button (e.g. "Dark Mode", "Airplane Mode").

Digital equivalent of a physical wall light switch.

Feedback & Micro-Interactions

Perceptible System Confirmation

Micro-interactions are contained product moments that accomplish a single task (e.g. liking a post, dragging to refresh, toggling a favorite). They provide perceptible confirmation that the system heard the user.

Inline Validation

✓ Username available

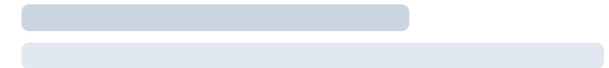
Validate inputs *as the user types or leaves the field* (on blur) rather than making them submit the whole form to discover errors.

Toasts & Snackbars

Email archived UNDO

Transient non-blocking notifications confirming operations. Always provide a brief "Undo" window for high-friction safety.

Skeleton Loaders



Mimic the shape of incoming content. Research shows skeleton loaders feel **30% faster** to users than circular spinners.

Section 5.3 Completed!

You have covered Screen Sizes, Viewports, Responsive Design, CSS clamp(), Thumb Ergonomics, and Interactive Element States.

[← Back to Unit 5 Hub](#)

[§5.1 Color Scheme](#)

[§5.2 Typography](#)