4.8. DESIGN SYSTEMS

What is a Design System?

A **Design System** is a collection of reusable UI components, design rules, guidelines, colors, fonts, and patterns used to create consistent and user-friendly websites and applications.

In simple words: A Design System is a set of rules and ready-made design elements that help designers and developers create applications with the same look and feel.

Example

Suppose we are designing a college application. We use the same:

- Colors on every page.
- Fonts and text sizes.
- Button styles.
- Icons and images.
- Navigation menus.
- Spacing and layouts.

All these elements together form a Design System.

Main Components of a Design System

1. **Design Tokens:** Colors, fonts, spacing and shadows.
2. **UI Components:** Buttons, cards, forms and icons.
3. **UI Patterns:** Login, search and checkout layouts.
4. **Typography:** Font styles and sizes.
5. **Color System:** Primary, secondary and background colors.
6. **Documentation:** Rules and guidelines for using designs.

Benefits

- Maintains consistency.
- Saves time.
- Reduces design errors.
- Makes components reusable.
- Improves user experience.
- Makes maintenance easier.

Real-Life Examples

- Google Material Design
- Apple Human Interface Guidelines
- Microsoft Fluent Design System
- IBM Carbon Design System

Design Systems in UI/UX

A Design System is a collection of reusable design components, guidelines, patterns and rules that help designers and developers create consistent, scalable and user-friendly digital products.



Consistent
Scalable
Accessible
Better UX

1. Introduction

A Design System is a structured collection of reusable design components and guidelines used to create consistent, scalable and user-friendly interfaces.

Example: Same button style, font, colors, icons, forms and navigation throughout all screens.

2. Why Do We Need It?

Without Design System

Login Screen
Registration Screen
Profile Screen
Payment Screen

With Design System

DESIGN SYSTEM
Buttons
Colors
Typography
Consistent UI

3. Goals

1. Consistency
2. Efficiency
3. Scalability
4. Usability
5. Accessibility
6. Collaboration
7. Maintainability
8. Quality

4. Components

DESIGN SYSTEM

Design Tokens	Components	Patterns
• Colors • Typography • Spacing • Shadows ...	• Buttons • Cards • Forms • Navigation ...	• Login • Checkout • Search ...

Documentation + Guidelines

5. Design Tokens

Smallest reusable values in a design system.

Examples:

- Primary color #2563EB
- Font size 16px
- Border radius 8px
- Spacing 16px
- Button height 48px

Benefits:

- Easy to maintain
- Consistent design
- Faster development
- Easy theme changes
- Supports multiple platforms

6. Color System

Primary Secondary	• Secondary • Supporting actions
Success	• Successful operation
Warning	• Warning messages
Error	• Error messages
Background	• Page background
Surface	• Cards and panels
Text	• Text • Content
Disabled	• Disabled controls

7. Typography

H1	32px	Bold
H2	24px	Bold
H3	20px	Semi-Bold
Body	16px	Regular
Caption	12px	Regular

Includes: font family, size, weight, line height, letter spacing, heading styles, body text styles, caption styles.

8. Spacing System

4 8 12 16 24 32 48

Heading
16px
Paragraph
24px
Button

9. UI Components

- Buttons
- Text fields
- Checkboxes
- Radio buttons
- Dropdowns
- Cards
- Navigation bars
- Tabs
- Modals
- Tooltips
- Alerts
- Badges
- Progress indicators

10. Component States

- Default
- Hover
- Focus
- Active / Pressed
- Disabled
- Selected
- Loading
- Error
- Success

Button Example

LOGIN
LOGIN
LOGIN
LOGIN
✓ Logged in

11. UI Patterns

- Login pattern
- Search pattern
- Checkout pattern
- Navigation pattern
- Form pattern
- Onboarding pattern
- Filter pattern
- Pagination pattern
- Error-handling pattern

12. Layout & Grid System

Desktop (1440px) Tablet (768px) Mobile (320px)

- Grid columns
- Margins
- Gutters
- Containers
- Breakpoints
- Alignment

13. Iconography

Search Home Settings Profile Delete

- Style
- Size
- Stroke width
- Color
- Spacing
- Usage guidelines

14. Accessibility

- Color contrast
- Keyboard accessibility
- Focus indicators
- Screen readers
- Touch targets

Accessible label: [x] Close

Instead of: [x]

15. Responsive Design

Mobile 320px Tablet 768px Desktop 1440px

16. Design Libraries

Figma Design System

- Colors
- Typography
- Spacing
- Buttons
- Forms
- Cards
- Navigation
- Modals
- Icons

17. Documentation

Component: Button
Usage: Primary actions

Do: Submit
Don't: Submit SUBMIT submit

- Variants
- States
- Accessibility
- Do's & Don'ts
- Code examples

18. Workflow

Research → Tokens → Components → Patterns
Test ← Document ← Library ← Use & Maintain

19. Design System vs Style Guide

Style Guide	Design System
Visual styles	Complete design language
Colors, typography, logo rules	Colors, typography, components, patterns
Basic guidelines	Interaction rules
Limited documentation	Extensive documentation
Mostly visual	Visual + functional

Style guide = how things look
Design system = how things look, behave and be reused.

20. Benefits

For Designers
Faster design, reusable components, fewer errors

For Developers
Reusable code, clear specs, faster implementation

For Users
Familiar interface, better usability, accessibility

For Organizations
Faster development, reduced effort, consistent brand

21. Real-Time Example – E-Commerce

E-COMMERCE DESIGN SYSTEM

Design Tokens	Components	Patterns
• Colors • Typography • Spacing • Shadows	• Buttons • Cards • Forms • Navigation	• Product Card • Checkout • Login • Payment

Consistent experience across all pages

22. Popular Design Systems

Google Material Design
Apple Human Interface Guidelines
Microsoft Design
IBM Carbon Design System
Shopify Polaris

23. Maintenance

Create → Use → Collect Feedback
Update → Improve → Identify Problems

Review, remove, add, fix accessibility, update documentation.

24. Governance

- Who can create?
- Who approves?
- Naming rules
- Version management
- Communicate changes
- Handle outdated components

Team: Design Lead, UX Designers, UI Designers, Developers, Accessibility Experts

25. In Figma

Design System
Colors
Typography
Spacing
Buttons
Forms
Cards
Navigation
Alerts
Modals
Icons
Responsive Components

26. College App Example

Colors: Primary, Success, Error, Background

Typography: H1 24px Bold; H2 20px Semi-Bold; Body 16px Regular; Caption 12px Regular

Components: Button, Card, Text Field, Dropdown, Navigation Bar, Profile Card, Notification

Patterns: Login, Attendance, Fee Payment, Exam Results, Course Registration

27. Key Principles

1. Consistent
2. Reusable
3. Scalable
4. Accessible
5. Flexible
6. Documented
7. Easy to maintain
8. Developer-friendly
9. User-centered
10. Continuously updated

28. Design System – Summary

Design Tokens (Colors, Typography, Spacing, Shadows) + Components (Buttons, Cards, Forms, Modals) + Patterns (Login, Checkout, Navigation, Search)

Documentation + Component Library

Consistent UI/UX Product

29. Key Takeaway

"A Design System is the foundation for creating consistent, scalable, accessible and efficient digital products."

It connects design principles + reusable components + design tokens + interaction patterns + documentation + development, allowing teams to create better user experiences faster.

4.9. COMPONENT LIBRARIES

What is a Component Library?

A **Component Library** is a collection of reusable UI elements such as buttons, forms, cards, icons, menus, and navigation bars that designers and developers can use to build websites and applications.

In simple words: A Component Library is a collection of ready-made design elements that can be reused on different pages instead of creating them again and again.

Example: Suppose we are creating a college application. We need:

- Login buttons
- Registration forms
- Navigation bars
- Student information cards
- Search boxes
- Icons

Instead of designing each element separately, we create them once and reuse them wherever needed. This collection is called a Component Library.

Types of Components

1. **Buttons:** Login, Submit, Cancel.
2. **Forms:** Text fields, checkboxes, dropdowns.
3. **Cards:** Student cards, product cards.
4. **Navigation:** Menus, tabs, navigation bars.
5. **Icons:** Search, home, settings.
6. **Modals:** Pop-up windows and dialog boxes.

Benefits of Component Libraries

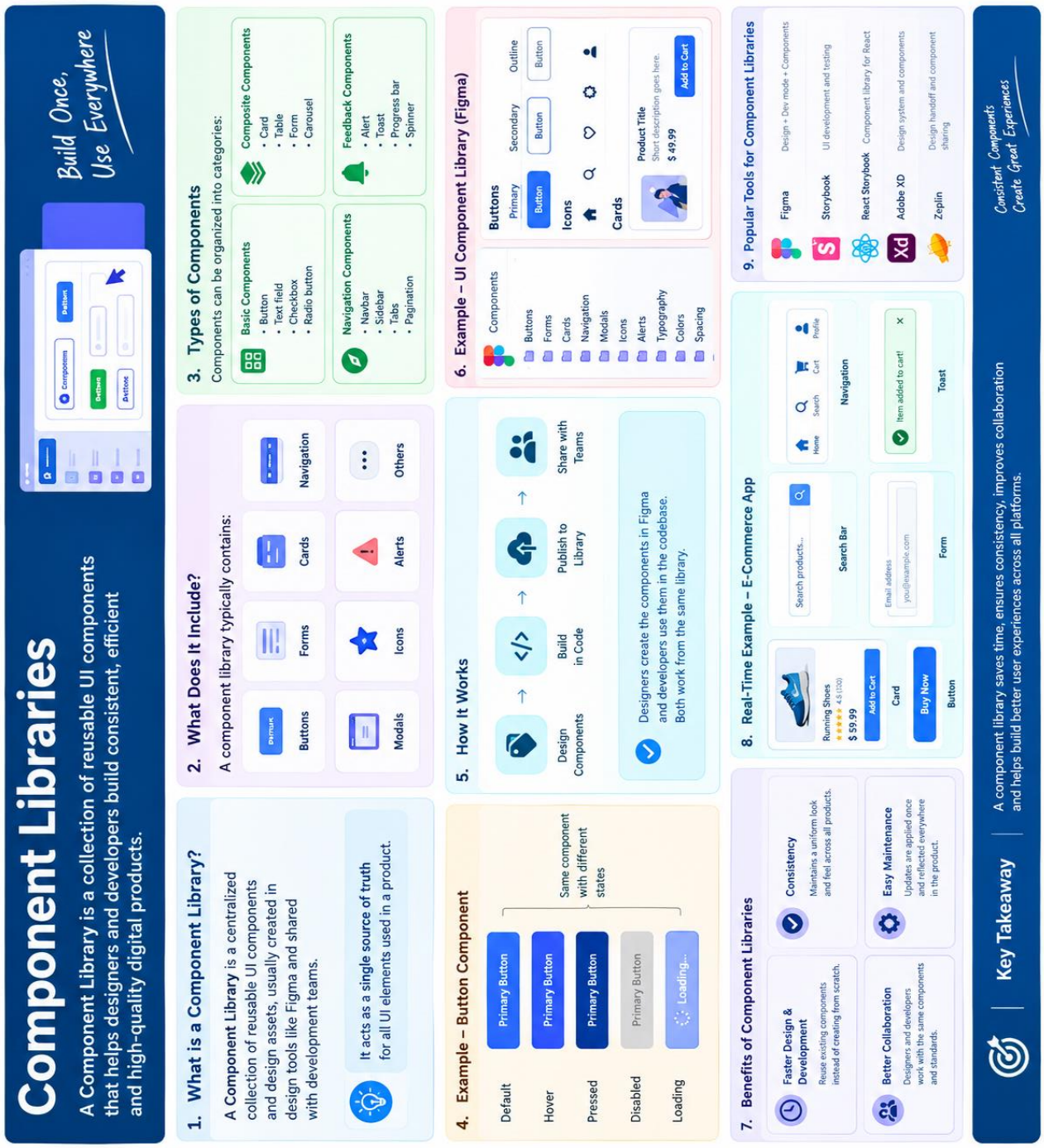
- Saves time and effort.
- Maintains design consistency.
- Allows component reuse.
- Reduces design errors.
- Makes updates easier.
- Improves teamwork between designers and developers.

Popular Tools

- Figma
- Storybook
- Bootstrap
- Material UI

Design System vs Component Library

Design System	Component Library
Complete set of design rules and guidelines	Collection of reusable UI elements
Includes colors, fonts, spacing and patterns	Includes buttons, cards, forms and icons
Defines how the product should look and behave	Provides ready-made components
Broader concept	Part of a design system



4.10. INTERACTIVE PROTOTYPING

What is Interactive Prototyping?

Interactive Prototyping is the process of creating a clickable and working model of a website or mobile application to show how it will look and behave before actual development.

In simple words: Interactive Prototyping allows users to click buttons, navigate between pages, fill forms, and experience how an application works without building the actual application.

Example

Suppose we are designing a college application.

An interactive prototype allows users to:

1. Click the Login button.
2. Enter username and password.
3. Click the Submit button.
4. Navigate to the Home page.
5. Open the Student Profile.
6. View the Timetable.

Main Elements of Interactive Prototyping

1. **Screens:** Different pages of an application.
2. **Buttons:** Allow users to perform actions.
3. **Links:** Connect different screens.
4. **Navigation:** Helps users move between pages.
5. **Interactions:** Define what happens when users click or tap.
6. **Animations:** Add movement and visual effects.

Steps in Interactive Prototyping

1. Understand user requirements.
2. Create wireframes.
3. Design UI screens.
4. Connect screens using links.
5. Add interactions and animations.
6. Test the prototype.
7. Collect feedback and improve the design.

Benefits of Interactive Prototyping

- Helps understand the application flow.
- Improves user experience.

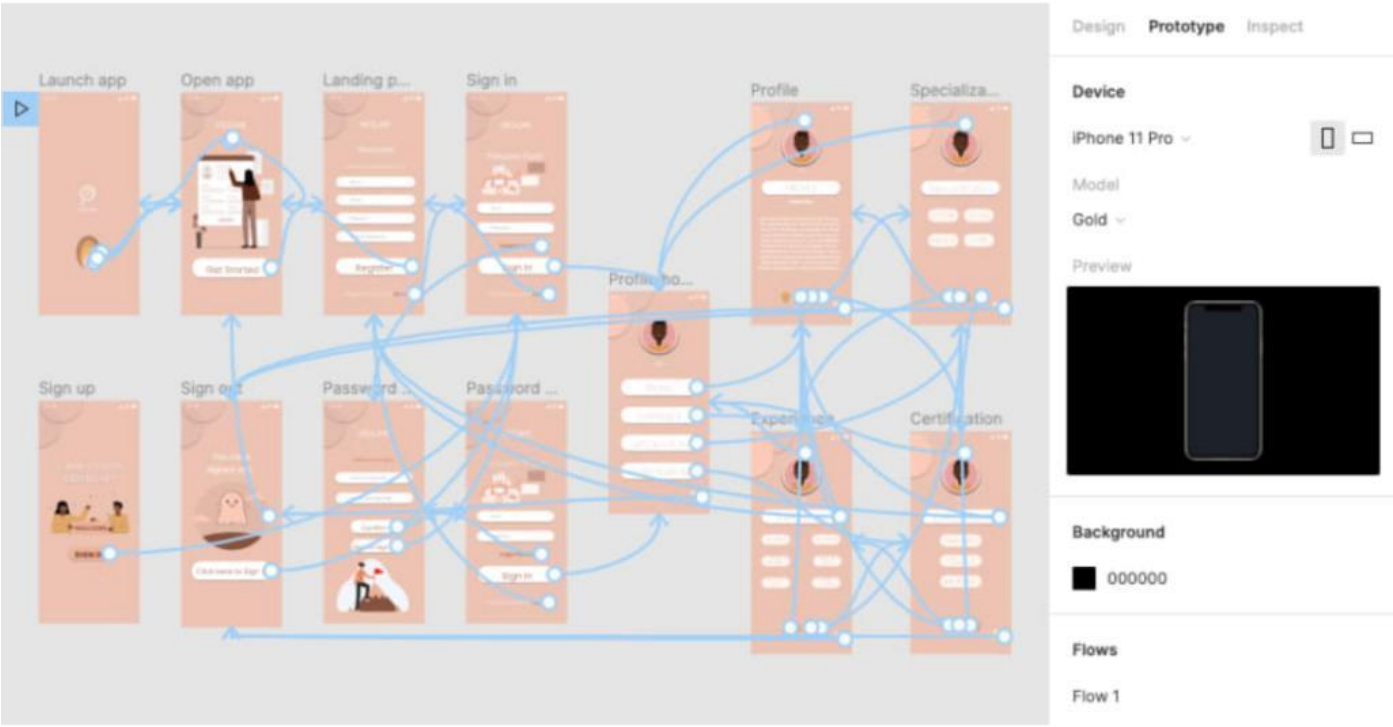
- Identifies design problems early.
- Saves development time.
- Makes it easier to collect feedback.
- Helps explain ideas to clients and developers.

Popular Tools

- Figma
- Adobe XD
- Sketch
- InVision

Static vs Interactive Prototype

Static Prototype	Interactive Prototype
Shows only the design	Shows design and interactions
Cannot click or navigate	Can click and navigate
Looks like a picture	Feels like a real application
Limited user testing	Better user testing



Interactive Prototype in UI/UX Design

A clickable model that shows how users navigate and interact with a product.



1. What is an Interactive Prototype?

An interactive prototype is a working model of a website, mobile app or software that allows users to click, tap, swipe, scroll and navigate between screens before the actual product is developed.

- It shows both the appearance and behaviour of the interface.
- It helps test the user flow, find usability problems, collect feedback and reduce development cost.

2. Why It is Important

- Test the user flow
- Check if navigation is easy
- Demonstrate how the application works
- Identify usability problems early
- Collect user and stakeholder feedback
- Reduce development errors and cost

3. Design Process

User Research → Ideas → Wireframe → UI Design → Interactive Prototype → User Testing → Refinement → Development

4. Static Design vs Interactive Prototype



Static Design
- Looks like a picture
- No real interaction
- Only shows appearance

Interactive Prototype
- Clickable and navigable
- Shows behaviour
- Connects multiple screens

Static Design = Appearance
Interactive Prototype = Appearance + Behavior + Navigation

5. Components of an Interactive Prototype

Screens	Different pages or states of the application.
Hotspots	Interactive areas (buttons, icons, links, etc.).
Interactions	User actions (tap, swipe, drag, scroll, etc.).
Transitions	How screens change from one to another.
Animations	Provide visual feedback.
User Flow	Sequence of steps to complete a task.

6. Interactions

User Action	Example
Tap / Click	Tap Login
Swipe	Swipe image gallery
Drag	Drag a slider
Scroll	Scroll product list
Hover	Hover over menu
Long press	Long press an item
Type	Enter username
Select	Select payment method

7. Triggers and Actions

Trigger	Action
Tap "Login"	Navigate to Home Screen
Swipe left	Show next image
Tap "Add to Cart"	Change cart quantity (0 → 1)

8. Types of Triggers

1. **Tap / Click** → press a button
2. **Drag** → move a slider
3. **Swipe** → switch between screens
4. **Hover** → show menu
5. **Keyboard Input** → enter text

9. Transitions

Common transition types:
- Instant
- Dissolve
- Slide left / right
- Slide up / down
- Smart animation
- Fade



10. Animation

Examples:
- Button press
- Loading animation
- Menu opening
- Card expansion
- Image transition
- Notification popup
- Progress indicator
- Page transition



11. User Flow Example (Online Pharmacy)



12. High-Fidelity Prototype



Includes:
- Realistic colors
- Typography
- Images & icons
- Buttons & forms
- Navigation
- Animations
- Microinteractions
- Responsive layouts
- Real content

13. Interactive Prototyping Tools

Figma | Adobe XD | Sketch | Framer

14. Figma Workflow (Basic)

1. Create frames (screens)
2. Design the UI
3. Open Prototype mode
4. Select interactive element
5. Create connection (link)
6. Define trigger (tap, etc.)
7. Define action (navigate)
8. Choose transition
9. Preview and test

15. Prototype Testing

Example Task:
"Find a laptop under ₹50,000 and add it to the cart."

Observe:
- Can users find the search option?
- Is navigation clear?
- Are labels understandable?
- Can users complete the task?
- Where do users make mistakes?

16. Advantages & Limitations

Advantages
- Early error detection
- Better user experience
- Saves development time
- Better communication
- Easy modification
- Demonstrates user flow
- Supports usability testing

Limitations
- Requires considerable design time
- Not the actual application
- Complex backend not shown
- May create false expectations
- Difficult to maintain many screens and interactions

17. Real-Time Applications

- **E-Commerce** (Home → Category → Product → Cart → Payment)
- **Banking** (Login → Transfer → Confirm → Success)
- **Food Delivery** (Restaurant → Menu → Cart → Payment → Tracking)
- **Travel** (Search → Destination → Hotel → Booking → Confirm)
- **Healthcare** (Login → Appointment → Doctor → Date → Confirm)
- **Education** (Login → Course → Lesson → Quiz → Result)
- **Women's Safety** (Home → Emergency → SOS → Location Sharing → Alert)

18. Classroom Activity – Food Delivery Prototype

Create 6 screens:
1. Home
2. Restaurant
3. Food Details
4. Cart
5. Payment
6. Order Confirmation

User Flow:
Home → Restaurant → Food Details → Cart → Payment → Order Confirmation

Add interactions:
- Quantity control (− 1 +)
- Swipe image gallery



19. Learning Outcomes

After this activity, students can:
- Explain interactive prototyping
- Identify prototype components
- Create a user flow
- Connect multiple UI screens
- Define triggers and actions
- Apply transitions and animations
- Design microinteractions
- Build a clickable prototype
- Conduct basic usability testing
- Refine design using feedback

20. Simple Definition

Interactive prototyping is the process of creating a clickable model of a digital product that simulates how users navigate and interact with the final product before actual development.

UI Design (What it looks like) + UX Design (How it works for the user) = Interactive Prototype (How it looks, works and responds)

Better Designs → Better Experiences → Happier Users

4.11. MOCKUPS AND ANIMATIONS

1. What are Mockups?

- A **Mockup** is a visual representation of a website or mobile application that shows how the final product will look.
- It displays the actual design, including colors, fonts, images, buttons, icons, and layouts.
- A Mockup is a realistic picture of how a website or application will look before it is developed.

Example

Suppose we are designing a college application. A mockup shows:

- Login page design.
- Button colors and styles.
- College logo.
- Student dashboard.
- Navigation menu.
- Images and fonts.

Features of Mockups

1. Shows the visual appearance of an application.
2. Includes colors and typography.
3. Displays images and icons.
4. Shows button and layout designs.
5. Helps visualize the final product.

Benefits of Mockups

- Helps understand the final design.
- Improves communication with clients.
- Identifies visual design problems.
- Saves development time.
- Helps collect feedback.

2. What are Animations?

Animations are visual effects that create movement or transitions between elements in a website or mobile application.

In simple words: Animations make an application look more attractive and interactive by adding movement to buttons, images, menus, and screens.

Example

In a mobile application:

- A button changes color when clicked.
- A loading icon rotates.
- A menu slides from the side.
- A screen smoothly moves to another screen.
- A notification appears on the screen.

Types of Animations

1. **Loading Animation:** Shows that data is loading.
2. **Transition Animation:** Smooth movement between screens.
3. **Hover Animation:** Changes appearance when the mouse moves over an element.
4. **Button Animation:** Provides feedback when a button is clicked.
5. **Scroll Animation:** Elements move or appear while scrolling.
6. **Micro-interactions:** Small animations that respond to user actions.

Benefits of Animations

- Makes the application attractive.
- Improves user experience.
- Provides visual feedback.
- Makes navigation smoother.
- Helps users understand actions.
- Makes the interface feel interactive.

3. Difference Between Mockups and Animations

Mockups	**Animations**
Show the visual design.	Show movement and effects.
Mostly static.	Dynamic and moving.
Focus on appearance.	Focus on interaction and movement.
Show colors, fonts and layouts.	Show transitions and actions.
Help visualize the final product.	Improve user experience.

4. Popular Tools

- Figma – UI mockups and animations.
- Adobe Photoshop – Visual mockups.
- Adobe After Effects – Advanced animations.
- Framer – Interactive designs and animations.

Mockups and Animations in UI/UX Design

Mockups show how it looks. Animations show how it moves.

1. What are Mockups?

A mockup is a high-fidelity visual representation of a design. It shows how the final product will look with real colors, typography, images and components. It helps stakeholders understand the appearance of the product before development.

A mockup is like a "real picture" of the product, but it is not clickable or interactive.

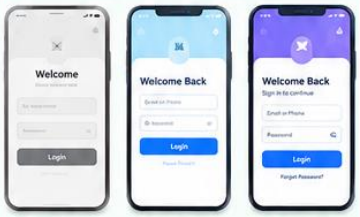
2. Types of Mockups

Low-Fidelity Mockup
- Simple layout
- No real colors
- Basic shapes

Mid-Fidelity Mockup
- Refined layout
- Placeholder images
- Some real colors

High-Fidelity Mockup
- Final look & feel
- Real images
- Typography, icons
- Detailed UI components

3. Example – Mockups



Low-Fidelity (Wireframe) | Mid-Fidelity (Design Mockup) | High-Fidelity (Realistic Mockup)

4. What are Animations?

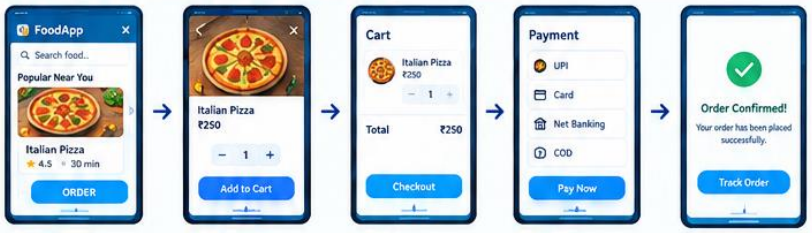
Animations are visual effects that show movement and change in the interface. They make the application feel more natural, provide feedback and improve user experience by guiding the user's attention.

Animations should be subtle, useful and support the user's understanding, not just for decoration.

5. Types of Animations

1. **Button Press** — Add to Cart → Added ✓ — Shows action feedback
2. **Loading Animation** — Indicates process is ongoing
3. **Page Transition** — Smoothly moves between screens
4. **Menu Opening** — Home, Profile, Settings — Reveals hidden options
5. **Image Transition** — Slides or fades between images
6. **Notification Popup** — New message! — Alerts the user
7. **Progress Indicator** — 1 2 3 4 — Shows completion status
8. **Microinteractions** — 128 — Small responses to user actions

6. Example – Animations in a Food Delivery App



1. Tap "ORDER" (Button animation)
2. Add to Cart (Icon animation)
3. Checkout (Transition animation)
4. Pay Now (Loading animation)
5. Confirmation (Success animation)

7. Common Animation Tools

- **Figma** — Built-in prototyping; Easy to use; Great for teams
- **Adobe XD** — Interactive prototypes; Motion presets; Integration with Creative Cloud
- **Sketch** — Vector design; Simple animations; Popular for UI design
- **Framer** — Advanced animations; Interactive web prototypes; Code export

8. Best Practices for Animations

- Keep animations short (0.1s – 0.5s).
- Use consistent timing and easing.
- Make them purposeful (provide feedback).
- Maintain performance (avoid lag).
- Follow platform guidelines (iOS / Android / Web).
- Use familiar patterns (e.g., slide, fade, scale).
- Test on real devices.

Good vs Bad
- ✓ Smooth, natural, helpful
- ✗ Too long, flashy, distracting

> "Good animation disappears. It feels natural and helps the user."

9. Mockup vs Animation

Feature	Mockup	Animation
Purpose	Shows appearance	Shows movement and interaction
Interactivity	Not interactive	Interactive (clickable)
Elements	Colors, typography, images, components	Transitions, effects, feedback, motion
Tools	Figma, Adobe XD, Sketch, etc.	Figma, Adobe XD, Framer, etc.
Example	A static login screen	Button click → next screen

10. Mockups and Animations in Real-World Applications

- **E-commerce** — Product pages, cart flow, checkout journey
- **Banking** — Login, balance, transactions, notifications
- **Food Delivery** — Browse, add to cart, order tracking
- **Travel** — Search, booking, confirmation
- **Healthcare** — Appointments, reports, reminders
- **Education** — Course flow, quiz feedback
- **Women's Safety** — SOS, alert, location sharing

11. Summary

Mockups + Animations = Better UX

Mockups: Show how it looks
Animations: Show how it moves
Better UX: Create a real user experience

Together, mockups and animations make your design come to life!

Design → Visualize → Animate → Delight Users

4.12. DESIGN HANDOFF

1. What is Design Handoff?

- **Design Handoff** is the process of transferring the final UI/UX design from designers to developers so they can convert the design into a working website or mobile application.
- Design Handoff means giving all design files, images, colors, fonts, measurements, and instructions to developers so they can build the application correctly.

2. Example

Suppose a designer creates a college application in Figma.

The designer shares the following with the developer:

- Login page design.
- Button styles and colors.
- Font styles and sizes.
- Images and icons.
- Screen layouts.
- Navigation flow.
- Spacing and measurements.

The developer uses these details to develop the actual application.

3. Important Elements of Design Handoff

1. **Design Files:** Final UI designs and layouts.
2. **Assets:** Images, logos, icons and illustrations.
3. **Typography:** Font family, size and weight.
4. **Color Codes:** Exact colors used in the design.
5. **Spacing:** Margins, padding and gaps.
6. **Measurements:** Height, width and alignment.
7. **Interactions:** Button clicks, hover effects and animations.
8. **Responsive Design:** Mobile, tablet and desktop layouts.
9. **Documentation:** Instructions and design guidelines.

4. Steps in Design Handoff

1. Complete the UI/UX design.
2. Organize design files and components.
3. Export images, icons and other assets.
4. Provide colors, fonts and measurements.
5. Explain interactions and navigation.

6. Share the design with developers.
7. Developers start implementation.
8. Review the developed application and provide feedback.

5. Benefits of Design Handoff

- Improves communication between designers and developers.
- Reduces development mistakes.
- Saves time.
- Maintains design consistency.
- Helps developers understand the design.
- Improves final product quality.

6. Popular Tools for Design Handoff

- Figma – Sharing designs and inspecting specifications.
- Zeplin – Sharing design specifications and assets.
- Adobe XD – Sharing design prototypes and specifications.
- Avocode – Design inspection and developer collaboration.

7. Designer vs Developer Responsibilities

Designer	**Developer**
Creates UI designs	Develops the application
Provides colors and fonts	Implements colors and fonts
Creates prototypes	Implements interactions
Shares images and icons	Uses assets in development
Explains design guidelines	Follows design guidelines

8. Example – College App

In a college application design handoff:

- Designer creates the login screen.
- Designer shares the Figma file.
- Developer checks button sizes and colors.
- Developer uses the provided images and icons.
- Developer creates the actual login page.
- Designer checks whether the final page matches the design.

Design Handoff in UI/UX Design

Bridging the gap between design and development.

1. What is Design Handoff?

Design handoff is the process of transferring design files, specifications and assets from designers to developers so that the final product can be built accurately and efficiently.

It ensures that the **design intent**, **visual style** and **interaction details** are clearly understood by the development team.

2. Why is it Important?

- Reduces misunderstandings
- Saves time and effort
- Maintains design consistency
- Helps in accurate implementation
- Improves collaboration between designers and developers
- Ensures a better user experience

3. What is Included in a Handoff?

- **Design Files** (Figma, XD, Sketch, etc.)
- **Assets** (images, icons, logos, etc.)
- **Typography** (fonts, sizes, styles)
- **Colors** (hex codes, style guide)
- **Components** (buttons, inputs, cards, etc.)
- **Specifications** (spacing, behavior, notes)

4. Types of Handoff

1. Design Handoff
Visual and style information

- Design files
- UI assets
- Color palette
- Typography
- Component library
- Style guide

2. Interaction Handoff
Behavior and functionality details

- User flows
- Screen transitions
- Microinteractions
- Animation details
- Linking between screens
- Responsive behavior

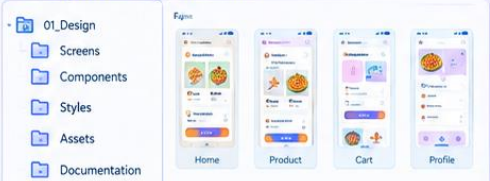
5. Tools for Design Handoff

Figma	• Design files, prototypes, assets • Inspect & share feature
Adobe XD	• Design specs, links, assets • Prototype sharing
Sketch	• Design files, symbols, assets • Export options
Zeplin	• Style guide, assets, specs • Developer handoff

6. Figma – Handoff Features

- **Inspect** — View exact measurements, colors, fonts and assets
- **Share** — Send a link to developers
- **Design System** — Manage components and styles
- **Code View** — Get CSS, iOS, Android code snippets
- **Comments** — Add notes and feedback

7. Design File Organization



8. Key Details to Share

Category	Examples
Spacing	Margin, padding, gap (e.g., 16px)
Colors	#3B82F6, #FFFFFF, #1F2937
Typography	Inter, 16px, Medium, 24px, Bold
Components	Button, Input, Card, Nav Bar
Interactions	Tap, Swipe, Hover, Scroll
Transitions	Slide, Fade, Smart Animate
Assets	Icons, Images, Logos
Responsive	Mobile, Tablet, Desktop

9. Handoff Process

1. **Organize design files** — Use clear layers, groups and naming.
2. **Prepare assets and specs** — Export images, icons, colors, typography.
3. **Share with developers** — Use Figma/Zeplin or a shared document.
4. **Clarify doubts** — Discuss and resolve queries.
5. **Track and iterate** — Update files based on feedback.

10. Best Practices

- Use consistent naming conventions
- Maintain proper hierarchy
- Follow a design system
- Add clear annotations
- Keep communication open

11. Real-World Example

Designer creates UI in Figma → **Handoff** with specs & assets → **Developer** builds the product

Good handoff = Faster development + Fewer bugs + Consistent user experience

12. Summary

Design handoff is an essential step in the UI/UX process. It ensures that the design vision is clearly communicated to the development team with all the necessary files, assets and specifications. A smooth handoff leads to accurate implementation, better collaboration and a high-quality final product.

Design + Share + Build = Great User Experience