

Table of Contents

INPUT ELEMENTS

Text Input

Text Input (Error)

Primary Button

Secondary Button

Checkbox

Radio Button

NAVIGATION ELEMENTS

Segmented Control

Chip Group

Tabs

FEEDBACK ELEMENTS

Error State

UI/UX Reference Guide

A comprehensive guide for communicating design requirements using proper terminology


Introduction

This reference guide provides the terminology, specifications, and code snippets for common UI elements. Use this guide to communicate design changes clearly and effectively with your development team.

Each element includes:

- Visual example
- Technical description
- CSS properties
- How to request changes
- Common states

Pro Tip:

Use the copy button () to easily copy the technical descriptions and paste them into your communications.

Input Elements

enter email

Text Input [Input Element](#)

Technical Description

 Copy

Text input field with light border and placeholder text in default state

CSS Properties

 Copy

```
.text-input {  
  border: 1px solid rgba(0,0,0,0.1);  
  border-radius: 8px;  
  padding: 16px;  
  width: 100%;  
  font-size: 16px;  
  transition: border-color 0.2s ease;  
}
```

```
.text-input::placeholder {  
  color: rgba(0,0,0,0.3);  
}  
  
.text-input:focus {  
  outline: none;  
  border-color: rgba(0,0,0,0.2);  
  box-shadow: 0 0 0 1px rgba(0,0,0,0.05);  
}
```

How to Request Changes

 Copy

Instead of saying: "Make the text box pop more"

Say: "Increase input field border opacity to 20% and add a subtle 2px

Instead of saying: "Make the placeholder text easier to read"

Say: "Adjust placeholder text opacity to 40% to improve readability w/

Common States

- **Default:** Light border, placeholder text
- **Focus:** Darker border, possible subtle shadow
- **Filled:** Contains user-entered text
- **Disabled:** Reduced opacity, often with background color change
- **Error:** Red/error color border or outline

2nd attempt

Text Input (Error State) [Validation Element](#)

Technical Description

 Copy

Text input with error validation state and prominent error stroke. The

CSS Properties

 Copy

```
.text-input.error {  
  border: 2px solid #FF3B30; /* iOS-style system red */  
  border-radius: 8px;  
  padding: 16px;  
  width: 100%;  
  font-size: 16px;  
  box-shadow: 0 0 0 1px rgba(255, 59, 48, 0.2); /* Subtle error glow */  
}  
  
.text-input.error:focus {  
  outline: none;  
  box-shadow: 0 0 0 2px rgba(255, 59, 48, 0.2);  
}  
  
/* Optional error message styling */  
.error-message {  
  color: #FF3B30;  
  font-size: 14px;  
  margin-top: 4px;  
}
```

How to Request Changes

 Copy

Instead of saying: "The red outline is too harsh"

Say: "Reduce the error state border thickness to 1px and decrease color"

Instead of saying: "Add a little message when there's an error"

Say: "Include an inline error message component below the input field"

Implementation Notes

- Error states should be accompanied by descriptive error messages whenever possible
- Consider adding a subtle shake animation on submission for additional feedback
- Maintain consistent padding despite the thicker border to prevent layout shifts
- Ensure color contrast meets accessibility standards (4.5:1 minimum ratio)

Next

Primary Button Action Element

Technical Description

 Copy

Primary action button with solid background fill and high contrast text

CSS Properties

 Copy

```
.button-primary {
  background-color: #0070F3; /* Brand primary blue */
  color: white;
  border: none;
  border-radius: 8px;
  padding: 12px 24px;
  font-size: 16px;
  font-weight: 500;
  text-align: center;
  cursor: pointer;
  transition: background-color 0.2s, transform 0.1s, box-shadow 0.2s;
  box-shadow: 0 2px 4px rgba(0, 0, 0, 0.1);
}
```

```
}

.button-primary:hover {
  background-color: #0060D9;
  box-shadow: 0 4px 6px rgba(0, 0, 0, 0.12);
}

.button-primary:active {
  transform: translateY(1px);
  box-shadow: 0 1px 2px rgba(0, 0, 0, 0.1);
}

.button-primary:disabled {
  background-color: #B3D4FF;
  cursor: not-allowed;
  box-shadow: none;
}
```

How to Request Changes

 Copy

Instead of saying: "Make the button more clicky"

Say: "Enhance the primary button's interactive feedback by increasing

Instead of saying: "The button should feel more important"

Say: "Increase the visual weight of the primary button by adjusting th

Common States

- **Default:** Solid fill, base elevation
- **Hover:** Slightly darker color, increased elevation
- **Active/Pressed:** Darker color, reduced elevation, slight scale down
- **Disabled:** Desaturated color, no elevation, no hover effects
- **Loading:** Same as default but with spinner/loading indicator

Secondary Button Action Element

Technical Description



Secondary action button with outline or subtle fill styling. This butt

CSS Properties



```
.button-secondary {
  background-color: white;
  color: #333333;
  border: 1px solid rgba(0, 0, 0, 0.1);
  border-radius: 8px;
  padding: 12px 24px;
  font-size: 16px;
  font-weight: 500;
  text-align: center;
  cursor: pointer;
  transition: background-color 0.2s, border-color 0.2s, transform 0.1s;
  box-shadow: 0 1px 2px rgba(0, 0, 0, 0.05);
}

.button-secondary:hover {
  background-color: #F5F5F5;
  border-color: rgba(0, 0, 0, 0.2);
  box-shadow: 0 2px 4px rgba(0, 0, 0, 0.08);
}

.button-secondary:active {
  transform: translateY(1px);
  box-shadow: none;
  background-color: #EEEEEE;
}
```

```
.button-secondary:disabled {
  background-color: #F5F5F5;
  color: #BBBBBB;
  border-color: rgba(0, 0, 0, 0.05);
  cursor: not-allowed;
  box-shadow: none;
}
```

How to Request Changes

 Copy

Instead of saying: "The button looks too plain"

Say: "Enhance the secondary button's visual definition by increasing k

Instead of saying: "I can't tell when I'm hovering over the button"

Say: "Strengthen the secondary button's hover state feedback by implem

Variant Options

- **Ghost:** No border, only hover/active states show background
- **Outlined:** Color border with transparent background
- **Subtle:** Very light background fill in brand color
- **Text button:** No background or border, only text color changes

☒ Remember me

Checkbox [Input Element](#)

Technical Description

 Copy

Binary selection control with toggleable checked state. The checkbox a

CSS Properties

 Copy

```
.checkbox-container {  
  display: flex;  
  align-items: center;  
  gap: 8px;  
  cursor: pointer;  
}
```

```
.checkbox-input {  
  position: absolute;  
  opacity: 0;  
  height: 0;  
  width: 0;  
}
```

```
.checkbox-custom {  
  width: 20px;  
  height: 20px;  
  border: 1px solid rgba(0, 0, 0, 0.2);  
  border-radius: 4px;  
  display: flex;  
  align-items: center;  
  justify-content: center;  
  transition: all 0.2s;  
  background: white;  
}
```

```
.checkbox-input:checked + .checkbox-custom {  
  background-color: #0070F3;  
  border-color: #0070F3;  
}
```

```
.checkbox-input:checked + .checkbox-custom::after {  
  content: '';  
  width: 5px;  
  height: 10px;  
  border: solid white;  
  border-width: 0 2px 2px 0;
```

```
transform: rotate(45deg);
margin-top: -2px;
}

.checkbox-input:focus + .checkbox-custom {
  box-shadow: 0 0 0 2px rgba(0, 112, 243, 0.3);
}

.checkbox-label {
  font-size: 16px;
  color: #333;
  user-select: none;
}
```

How to Request Changes



Instead of saying: "The checkbox is too small"

Say: "Increase the checkbox dimensions to 24x24px to improve touch target"

Instead of saying: "I want a cooler animation when checking"

Say: "Implement a subtle scale-in animation for the checkmark (transform: scale(1.1))"

Common States

- **Unchecked:** Empty square with border
- **Checked:** Filled square with checkmark
- **Indeterminate:** Partially selected state (horizontal line)
- **Focus:** Visible focus ring around the checkbox
- **Disabled:** Reduced opacity, non-interactive

☐ Option 1

☐ Option 2

Radio Button Input Element

Technical Description



Single-selection control for mutually exclusive options. Radio buttons

CSS Properties



```
.radio-container {
  display: flex;
  align-items: center;
  gap: 8px;
  cursor: pointer;
  margin-bottom: 8px;
}

.radio-input {
  position: absolute;
  opacity: 0;
  height: 0;
  width: 0;
}

.radio-custom {
  width: 20px;
  height: 20px;
  border: 1px solid rgba(0, 0, 0, 0.2);
  border-radius: 50%;
  display: flex;
  align-items: center;
  justify-content: center;
  transition: all 0.2s;
  background: white;
}

.radio-input:checked + .radio-custom {
  border-color: #0070F3;
}
```

```
.radio-input:checked + .radio-custom::after {
  content: '';
  width: 10px;
  height: 10px;
  border-radius: 50%;
  background-color: #0070F3;
}

.radio-input:focus + .radio-custom {
  box-shadow: 0 0 0 2px rgba(0, 112, 243, 0.3);
}

.radio-label {
  font-size: 16px;
  color: #333;
  user-select: none;
}
```

How to Request Changes



Instead of saying: "The selected circle is too small"

Say: "Increase the inner selected circle diameter to 60% of the container"

Instead of saying: "The spacing between options is wrong"

Say: "Adjust the vertical spacing between radio options to 16px to improve readability"

Accessibility Considerations

- Always wrap radio buttons in a fieldset with a descriptive legend
- Ensure sufficient color contrast between the selected indicator and background
- Provide keyboard navigation support (arrow keys) between options
- Include visible focus states for keyboard users
- Ensure touch targets are at least 44×44px for mobile users

Navigation Elements

Preview

Code

Thinking

Segmented Control [Navigation Element](#)

Technical Description

 Copy

Primary segmented control with active state indicator. This horizontal

CSS Properties

 Copy

```
.segmented-control {
  display: flex;
  background-color: #F2F2F7; /* iOS light gray */
  border-radius: 9999px; /* Fully rounded */
  padding: 4px;
  width: 100%;
}
```

```
.segment {
  flex: 1;
  text-align: center;
  padding: 8px 16px;
  border-radius: 9999px;
  font-size: 14px;
  font-weight: 500;
  transition: all 0.2s ease;
  cursor: pointer;
  user-select: none;
}
```

```
.segment.active {
  background-color: white;
  color: #333333;
```

```
    box-shadow: 0 1px 2px rgba(0, 0, 0, 0.1);
}

.segment:not(.active) {
    color: #8E8E93; /* iOS secondary text */
}

.segment:not(.active):hover {
    background-color: rgba(255, 255, 255, 0.5);
}
```

How to Request Changes

 Copy

Instead of saying: "Make the tabs more obvious"

Say: "Increase the contrast between active and inactive segments by using a secondary text color"

Instead of saying: "I want the tabs to animate when switching"

Say: "Implement a smooth sliding indicator that transitions between active and inactive states"

Common Variations

- **iOS Style:** Pill-shaped with filled active state (as shown)
- **Material Style:** Underline indicator for active state
- **Button Group:** Square edges with distinct button appearance
- **Text Only:** No container background, just text color/weight changes
- **Icon+Text:** Includes icons alongside labels for visual enhancement

Reset M

Web

Tablet

iPhone

Chip Group [Selection Element](#)

Technical Description

Horizontal selector chips with subtle elevation. These compact, pill-s

CSS Properties

```
.chip-group {
  display: flex;
  gap: 8px;
  overflow-x: auto;
  padding: 4px 0;
  scrollbar-width: none; /* Hide scrollbar for Firefox */
}

.chip-group::-webkit-scrollbar {
  display: none; /* Hide scrollbar for Chrome/Safari */
}

.chip {
  background-color: #F2F2F7;
  border-radius: 9999px;
  padding: 8px 16px;
  font-size: 14px;
  font-weight: 500;
  color: #333333;
  white-space: nowrap;
  cursor: pointer;
  user-select: none;
  transition: all 0.2s;
  box-shadow: 0 1px 2px rgba(0, 0, 0, 0.05);
}

.chip:hover {
  background-color: #E5E5EA;
  box-shadow: 0 2px 4px rgba(0, 0, 0, 0.1);
}

.chip.selected {
  background-color: #0070F3;
  color: white;
}
```

```
box-shadow: 0 2px 4px rgba(0, 112, 243, 0.2);  
}
```

How to Request Changes

 Copy

Instead of saying: "The chips look too flat"

Say: "Increase the chip elevation with a more pronounced shadow (0 2px 4px rgba(0, 112, 243, 0.2))"

Instead of saying: "Make it clearer which chip is selected"

Say: "For the selected chip state, implement a 2px accent color border"

Common Use Cases

- **Filtering:** Allow users to filter content by specific attributes
- **Selection:** Select options from a horizontally arranged set
- **Tags:** Display attributes or categories associated with content
- **Input:** Show selected values in a multi-select field
- **Navigation:** Provide horizontally scrolling navigation options

Home

Explore

Notifications

Tabs Navigation Element

Technical Description

 Copy

Horizontal tab navigation with underline indicator. This element provides a simple and intuitive way to switch between different views or sections of an application.

CSS Properties


```
.tab-container {
  display: flex;
  border-bottom: 1px solid #E5E5EA;
  width: 100%;
  position: relative;
}

.tab {
  padding: 12px 16px;
  font-size: 14px;
  font-weight: 500;
  color: #8E8E93;
  cursor: pointer;
  transition: all 0.2s;
  position: relative;
  white-space: nowrap;
  user-select: none;
}

.tab.active {
  color: #0070F3;
}

.tab.active::after {
  content: '';
  position: absolute;
  bottom: -1px;
  left: 0;
  right: 0;
  height: 2px;
  background-color: #0070F3;
  border-radius: 1px 1px 0 0;
}

.tab:hover:not(.active) {
  color: #666666;
  background-color: rgba(0, 0, 0, 0.02);
}

/* Optional sliding indicator */
```

```
.tab-indicator {  
  position: absolute;  
  bottom: 0;  
  height: 2px;  
  background-color: #0070F3;  
  transition: transform 0.3s ease;  
}
```

How to Request Changes



Instead of saying: "The tabs need to be more noticeable"

Say: "Increase tab text contrast by using darker inactive text color"

Instead of saying: "Make the blue line slide between tabs"

Say: "Implement an animated tab indicator with a 300ms cubic-bezier(0.

Tab Variations

- **Text Only:** Simple underline indicator (as shown)
- **Icon+Text:** Includes icons above or beside text for visual cues
- **Container Tabs:** Fully contained tabs with background fills
- **Scrollable Tabs:** Horizontally scrolling when many options exist
- **Bottom Tabs:** Mobile navigation pattern at screen bottom

Feedback Elements



Invalid email address

Please enter a valid email address to continue.

Error State [Feedback Element](#)

Technical Description

 Copy

Error notification component with contextual styling. This component c

CSS Properties

 Copy

```
.error-message {
  display: flex;
  align-items: flex-start;
  padding: 16px;
  border-radius: 8px;
  background-color: rgba(255, 59, 48, 0.08); /* iOS red with low opaci
  border: 1px solid rgba(255, 59, 48, 0.2);
  color: #D70015; /* Accessible red */
  margin-bottom: 16px;
}

.error-icon {
  width: 20px;
  height: 20px;
  margin-right: 12px;
  flex-shrink: 0;
}

.error-content {
  flex: 1;
}

.error-title {
  font-weight: 600;
  font-size: 14px;
  margin-bottom: 4px;
}

.error-description {
```

```
font-size: 13px;
opacity: 0.9;
}
```

How to Request Changes

 Copy

Instead of saying: "The error message is too alarming"

Say: "Soften the error message visual impact by reducing the backgrou

Instead of saying: "Add some animation when the error appears"

Say: "Implement a subtle entrance animation for the error message - a

Error Component Types

- **Inline Error:** Small message below a specific input field
- **Alert Banner:** Full-width notification at top of viewport
- **Error Card:** Self-contained error message (as shown)
- **Error Dialog:** Modal presentation for blocking errors
- **Field Indicator:** Visual marking on the input itself



Email verified successfully

You can now continue with the registration process.

Success State Feedback Element

Technical Description

 Copy

Success confirmation component with positive visual styling. This comp

CSS Properties



```
.success-message {
  display: flex;
  align-items: flex-start;
  padding: 16px;
  border-radius: 8px;
  background-color: rgba(52, 199, 89, 0.08); /* iOS green with low opacity */
  border: 1px solid rgba(52, 199, 89, 0.2);
  color: #108041; /* Accessible green */
  margin-bottom: 16px;
}

.success-icon {
  width: 20px;
  height: 20px;
  margin-right: 12px;
  flex-shrink: 0;
}

.success-content {
  flex: 1;
}

.success-title {
  font-weight: 600;
  font-size: 14px;
  margin-bottom: 4px;
}

.success-description {
  font-size: 13px;
  opacity: 0.9;
}
```

How to Request Changes



Instead of saying: "Make the success message more celebratory"

Say: "Enhance the success message with a subtle checkmark animation (s

Instead of saying: "Success message is too similar to error"

Say: "Increase differentiation between success and error states by usi

When to Use

- After completing form submissions or multi-step processes
- When confirming changes have been saved
- When validating input as correct during form completion
- After successful authentication or verification
- When changes or updates have been successfully applied



Loading Indicator [Feedback Element](#)

Technical Description

 Copy

Loading state indicators with animated visual feedback. These componer

CSS Properties

 Copy

```
/* Spinner Loader */
@keyframes spin {
  to { transform: rotate(360deg); }
}

.spinner {
```

```

width: 24px;
height: 24px;
border: 3px solid rgba(0, 112, 243, 0.3);
border-top-color: #0070F3;
border-radius: 50%;
animation: spin 0.8s linear infinite;
}

/* Progress Bar */
.progress-container {
width: 100%;
height: 4px;
background-color: rgba(0, 0, 0, 0.1);
border-radius: 2px;
overflow: hidden;
}

.progress-bar {
height: 100%;
background-color: #0070F3;
border-radius: 2px;
transition: width 0.3s ease;
}

/* Indeterminate Progress Bar */
@keyframes indeterminate {
0% { transform: translateX(-100%); }
100% { transform: translateX(400%); }
}

.progress-indeterminate {
width: 30%;
height: 100%;
background-color: #0070F3;
border-radius: 2px;
animation: indeterminate 1.5s infinite;
}

```

How to Request Changes

Instead of saying: "The loading spinner is boring"

Say: "Replace the standard circular spinner with a branded loading ani

Instead of saying: "Loading bar needs to look smoother"

Say: "Enhance the progress bar's perceived smoothness by implementing

Loading Indicator Types

- **Spinner:** Rotating circular indicator for indeterminate loading
- **Progress Bar:** Linear indicator showing completion percentage
- **Skeleton UI:** Placeholder elements mimicking content structure
- **Pulse/Wave:** Subtle animation on content containers
- **Text Indicator:** Ellipsis or changing text alongside static elements

Layout Components

Card Title

This is a basic card component with content area and optional footer actions.

Cancel Confirm

Card Layout Element

Technical Description

 Copy

Surface container component with content grouping and visual separatic

CSS Properties


```
.card {
  background-color: white;
  border-radius: 12px;
  overflow: hidden;
  box-shadow: 0 2px 8px rgba(0, 0, 0, 0.08);
  transition: box-shadow 0.3s ease, transform 0.3s ease;
}

.card:hover {
  box-shadow: 0 4px 12px rgba(0, 0, 0, 0.12);
}

.card-content {
  padding: 20px;
}

.card-title {
  font-size: 18px;
  font-weight: 600;
  color: #333333;
  margin-bottom: 8px;
}

.card-text {
  font-size: 14px;
  color: #666666;
  line-height: 1.5;
}

.card-footer {
  padding: 12px 20px;
  background-color: #F9F9FB;
  border-top: 1px solid #EEEEEE;
  display: flex;
  justify-content: flex-end;
  gap: 8px;
}
```

 Copy

Instead of saying: "Make the card pop more"

Say: "Increase the card's perceived elevation by adjusting its shadow"

Instead of saying: "The card corners could be rounder"

Say: "Refine the card's visual softness by increasing the border radius"

Card Anatomy

- **Container:** Main surface with elevation and boundary
- **Header (optional):** Title section, often with actions or metadata
- **Media (optional):** Image or rich media content area
- **Content:** Primary information area with text and UI elements
- **Footer (optional):** Action buttons or supplementary information

Confirm Action



Are you sure you want to perform this action? This cannot be undone.

Cancel

Confirm

Modal [Layout Element](#)

Technical Description

 Copy

Dialog overlay component with focus-capturing interaction pattern. This

CSS Properties

```
/* Modal Backdrop */
.modal-backdrop {
  position: fixed;
  top: 0;
  left: 0;
  right: 0;
  bottom: 0;
  background-color: rgba(0, 0, 0, 0.5);
  display: flex;
  align-items: center;
  justify-content: center;
  z-index: 50;
  opacity: 0;
  transition: opacity 0.3s ease;
}

.modal-backdrop.active {
  opacity: 1;
}

/* Modal Container */
.modal {
  background-color: white;
  border-radius: 12px;
  box-shadow: 0 4px 20px rgba(0, 0, 0, 0.15);
  width: 100%;
  max-width: 400px;
  max-height: 90vh;
  overflow-y: auto;
  transform: scale(0.95);
  opacity: 0;
  transition: transform 0.3s ease, opacity 0.3s ease;
}

.modal.active {
  transform: scale(1);
  opacity: 1;
}

/* Modal Header */
.modal-header {
```

```
padding: 16px 20px;
display: flex;
justify-content: space-between;
align-items: center;
border-bottom: 1px solid #EEEEEE;
}

/* Modal Body */
.modal-body {
padding: 20px;
}

/* Modal Footer */
.modal-footer {
padding: 16px 20px;
display: flex;
justify-content: flex-end;
gap: 8px;
border-top: 1px solid #EEEEEE;
}
```

How to Request Changes



Instead of saying: "The popup feels too abrupt"

Say: "Refine the modal entrance animation by using a 400ms cubic-bezier"

Instead of saying: "Make the background darker when the modal is open"

Say: "Increase modal backdrop opacity to 70% and add a subtle blur effect"

Implementation Best Practices

- Trap keyboard focus within the modal when open
- Provide multiple dismissal methods (close button, backdrop click, ESC key)
- Return focus to the triggering element when modal closes
- Use appropriate ARIA roles (dialog, alertdialog)
- Ensure the modal is properly centered on all screen sizes

- Prevent background content scrolling while modal is active

Email Address

you@example.com

Password

Enter your password

☐ Remember me

Sign In

Form Layout [Layout Element](#)

Technical Description

 Copy

Input group layout pattern with consistent vertical rhythm. This struc

CSS Properties

 Copy

```
.form-container {
  display: flex;
  flex-direction: column;
  gap: 20px; /* Consistent spacing between form groups */
  width: 100%;
}

.form-group {
  display: flex;
  flex-direction: column;
  gap: 6px; /* Space between label and input */
}
```

```
.form-label {
  font-size: 14px;
  font-weight: 500;
  color: #333333;
}

.form-input {
  width: 100%;
  padding: 10px 12px;
  font-size: 16px;
  border: 1px solid rgba(0, 0, 0, 0.15);
  border-radius: 8px;
  transition: border-color 0.2s, box-shadow 0.2s;
}

.form-input:focus {
  border-color: #0070F3;
  box-shadow: 0 0 0 2px rgba(0, 112, 243, 0.2);
  outline: none;
}

.form-actions {
  margin-top: 24px;
  display: flex;
  justify-content: flex-end;
  gap: 12px;
}
```

How to Request Changes



Instead of saying: "The form fields are too cramped"

Say: "Increase the vertical spacing between form groups to 24px and ac

Instead of saying: "Make the form look more organized"

Say: "Improve form visual structure by implementing a subtle group cor

Form Layout Patterns

- **Single Column:** Most mobile-friendly, best completion rates
- **Two Column:** Balanced between space efficiency and usability
- **Grouped Fields:** Related inputs visually clustered together
- **Inline Form:** Horizontal layout for very simple forms (search, subscribe)
- **Progressive Disclosure:** Multi-step approach revealing fields gradually

Design Systems

Heading 1

Heading 2

Heading 3

Body text that forms the majority of content. It should be highly readable with proper line height and good contrast.

Secondary text provides supporting information with slightly reduced visual prominence.

Caption text for auxiliary details like timestamps or metadata.


Typography Hierarchy System Element

Technical Description

 Copy

Typographic scale system with defined hierarchy levels. This foundatic

CSS Properties

 Copy

```
/* Font Family */
```

```
body {
  font-family: 'Inter', -apple-system, BlinkMacSystemFont, 'Segoe UI',
  color: #333333;
  line-height: 1.5;
}

/* Typographic Scale */
.display {
  font-size: 36px;
  font-weight: 700;
  line-height: 1.2;
  letter-spacing: -0.02em;
}

.heading-1 {
  font-size: 28px;
  font-weight: 700;
  line-height: 1.3;
  letter-spacing: -0.015em;
}

.heading-2 {
  font-size: 24px;
  font-weight: 600;
  line-height: 1.3;
  letter-spacing: -0.01em;
}

.heading-3 {
  font-size: 20px;
  font-weight: 600;
  line-height: 1.4;
  letter-spacing: -0.005em;
}

.subtitle {
  font-size: 16px;
  font-weight: 500;
  line-height: 1.5;
}

.body-text {
```



```
font-size: 16px;
font-weight: 400;
line-height: 1.6;
}

.caption {
font-size: 12px;
font-weight: 400;
line-height: 1.5;
color: #666666;
}

.overline {
font-size: 12px;
font-weight: 500;
line-height: 1.4;
letter-spacing: 0.05em;
text-transform: uppercase;
color: #666666;
}
```

How to Request Changes



Instead of saying: "The headings need to stand out more"

Say: "Increase contrast in the typographic hierarchy by adjusting head

Instead of saying: "The text feels too crowded"

Say: "Enhance typographic breathing room by increasing the line height

Typography Considerations

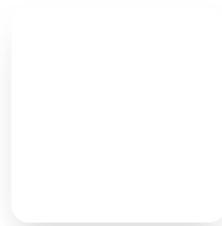
- **Scale Ratio:** Consistent mathematical relationship between sizes
- **Weight Hierarchy:** Using variation in font weight to establish importance
- **Color Contrast:** Ensuring all text meets WCAG AA (minimum 4.5:1) standards
- **Line Length:** Optimizing for 45-75 characters per line for readability
- **Responsive Adjustments:** How type scales on different viewport sizes



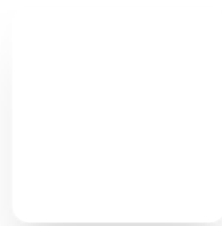
Level 1



Level 2



Level 3



Level 4

Elevation/Shadow System System Element

Technical Description

 Copy

Graduated elevation system with defined shadow levels. This systematic

CSS Properties

 Copy

```
/* Base Shadow Properties */
.shadow-level-0 {
  box-shadow: none;
  /* Flat elements, no elevation */
}

.shadow-level-1 {
  box-shadow: 0 1px 2px rgba(0, 0, 0, 0.05), 0 1px 3px rgba(0, 0, 0, 0.05);
  /* Subtle elevation for cards, buttons in rest state */
}

.shadow-level-2 {
  box-shadow: 0 2px 4px rgba(0, 0, 0, 0.05), 0 3px 6px rgba(0, 0, 0, 0.05);
  /* Medium elevation for hover states, dropdown menus */
}

.shadow-level-3 {
  box-shadow: 0 4px 8px rgba(0, 0, 0, 0.05), 0 8px 16px rgba(0, 0, 0, 0.05);
  /* Higher elevation for modals, dialogs, popovers */
}
```

```
.shadow-level-4 {
  box-shadow: 0 8px 16px rgba(0, 0, 0, 0.05), 0 16px 32px rgba(0, 0, 0, 0.05);
  /* Maximum elevation for important dialogs, onboarding elements */
}

/* Inner Shadow for Pressed States */
.shadow-inset {
  box-shadow: inset 0 2px 4px rgba(0, 0, 0, 0.06);
  /* For pressed buttons, active states */
}
```

How to Request Changes



Instead of saying: "Make the shadows more dramatic"

Say: "Increase shadow contrast across all elevation levels by adjusting the opacity and blur radius"

Instead of saying: "The shadow looks weird on dark mode"

Say: "Adapt the shadow system for dark mode by inverting the shadow color and adjusting the opacity"

Shadow Variables

- **Y-offset:** Distance shadow is cast vertically (suggests height)
- **Blur radius:** How diffuse/soft the shadow appears
- **Spread radius:** How far the shadow extends beyond element
- **Opacity:** Shadow transparency affects perceived depth
- **Color:** Usually black, but can be adapted for different backgrounds
- **Multiple layers:** Combining shadows for more realistic effects

State Terminology System Element

Technical Description

Consistent interface state system for interactive elements. This stanc

State Definitions & Usage

Default/Rest

The normal, initial appearance of an element before any user interaction.

 Copy

```
/* Example for button */  
.button {  
  background-color: #F8F9FA;  
  border: 1px solid #E2E8F0;  
  color: #334155;  
}
```

Hover

Visual feedback when a pointer is positioned over an interactive element.

 Copy

```
/* Example for button */  
.button:hover {  
  background-color: #F1F5F9;  
  border-color: #CBD5E1;  
  box-shadow: 0 2px 4px rgba(0, 0, 0, 0.05);  
}
```

Focus

Visual indicator showing an element has keyboard focus, critical for accessibility.

 Copy

```
/* Example for input */
```

```
.input:focus {  
  border-color: #3B82F6;  
  box-shadow: 0 0 0 3px rgba(59, 130, 246, 0.3);  
  outline: none;  
}
```

Active/Pressed

Momentary state during the action of clicking/tapping an interactive element.

 Copy

```
/* Example for button */  
.button:active {  
  background-color: #E2E8F0;  
  border-color: #CBD5E1;  
  transform: translateY(1px);  
  box-shadow: inset 0 2px 4px rgba(0, 0, 0, 0.05);  
}
```

Selected

Persistent state indicating an element has been chosen from a group.

 Copy

```
/* Example for tab */  
.tab.selected {  
  color: #1E40AF;  
  font-weight: 500;  
  border-bottom: 2px solid #1E40AF;  
}
```

Disabled

Non-interactive state indicating an element cannot be used currently.

 Copy

```
/* Example for button */  
.button:disabled {  
  background-color: #F1F5F9;  
  color: #94A3B8;  
  border-color: #E2E8F0;  
  cursor: not-allowed;  
  box-shadow: none;  
}
```

Error

Indicates validation failure or problem with element's current value.

 Copy

```
/* Example for input */  
.input.error {  
  border-color: #DC2626;  
  background-color: #FEF2F2;  
}  
  
.error-message {  
  color: #DC2626;  
  font-size: 12px;  
  margin-top: 4px;  
}
```

Loading

Indicates an operation is in progress and the element is waiting for completion.

 Copy

```
/* Example for button */  
.button.loading {  
  position: relative;
```

```
color: transparent; /* Hide text */
}

.button.loading::after {
  content: "";
  position: absolute;
  width: 16px;
  height: 16px;
  top: calc(50% - 8px);
  left: calc(50% - 8px);
  border: 2px solid rgba(255, 255, 255, 0.3);
  border-radius: 50%;
  border-top-color: white;
  animation: spin 0.8s linear infinite;
}
```

How to Request Changes



Instead of saying: "The button doesn't seem clickable"

Say: "Enhance the interactive affordance by implementing a more pronounced button state"

Instead of saying: "It's hard to tell which tab is selected"

Say: "Strengthen the selected state visual treatment by combining multiple visual cues"

State System Guidelines

- Maintain consistent state treatments across similar element types
- Use multiple visual cues (color, shape, shadow) for important states
- Ensure state changes have appropriate transition animations
- Don't rely solely on color to communicate state (accessibility)
- Consider mobile touch vs desktop click interactions in state design
- Test states across different viewport sizes and devices

