WebOS Theme Implementation Guide

Overview

This guide documents the comprehensive webOS theme implementation for loomOS. The theme is inspired by the classic Palm/HP webOS interface, featuring a clean, minimalist design with a light gray color palette.

Table of Contents

Design Philosophy

The webOS theme follows these core principles:

- **Minimalism**: Clean, uncluttered interface with focus on content
- **Neutral Palette**: Light grays with minimal use of color
- **Card-Based Design**: Content organized in distinct, elevated cards
- **Subtle Effects**: Soft shadows and gentle transitions
- **Touch-Friendly**: Generous spacing and clear tap targets
- **Accessibility**: High contrast and clear visual hierarchy

Color Palette

Background Colors

```
--webos-bg-primary: #e8e8e8       /* Main background */
--webos-bg-secondary: #ebebeb     /* Secondary surfaces */
--webos-bg-tertiary: #eeeeee      /* Elevated surfaces */
```

Surface Colors

```
--webos-surface: #ffffff          /* Card backgrounds */
--webos-surface-hover: #fafafa    /* Hover state */
--webos-surface-active: #f5f5f5   /* Active state */
```

Text Colors

```
--webos-text-primary: #000000     /* Primary text - Pure black */
--webos-text-secondary: #666666   /* Secondary text */
--webos-text-tertiary: #999999    /* Tertiary text */
--webos-text-disabled: #cccccc    /* Disabled text */
```

Border Colors

```
--webos-border-light: #e0e0e0     /* Light borders */
--webos-border-medium: #d4d4d4    /* Medium borders */
--webos-border-dark: #b0b0b0      /* Dark borders */
```

Accent Colors (Minimal Usage)

```
--webos-accent-blue: #4a90e2      /* Links, selections */
--webos-accent-green: #5cb85c     /* Success states */
--webos-accent-red: #d9534f       /* Error states */
--webos-accent-orange: #f0ad4e    /* Warning states */
```

Components

1. WebOS Just Type Search

The iconic "JUST TYPE" search interface.

```
import { WebOSJustTypeSearch } from '@/components/webos';

<WebOSJustTypeSearch
  onSearch={(query) => console.log(query)}
  placeholder="Just type to search..."
  showLabel={true}
/>
```

Features:
- Clean, minimalist design
- "JUST TYPE" label
- Rounded corners
- Subtle shadow effects
- Auto-focus on mount

2. WebOS Card Carousel

Horizontal scrolling card layout.

```
import { WebOSCardCarousel } from '@/components/webos';

const cards = [
  {
    id: '1',
    title: 'Card Title',
    content: <div>Card content here</div>,
    onClick: () => console.log('Clicked')
  }
];

<WebOSCardCarousel cards={cards} cardWidth={320} gap={24} />
```

Features:

- Smooth horizontal scrolling
- Snap scrolling
- Navigation arrows
- Touch/swipe support
- Responsive design

3. WebOS Card

Individual card component.

```
import { WebOSCard } from '@/components/webos';

<WebOSCard title="Card Title" image="/path/to/image.jpg">
  <p>Card content</p>
</WebOSCard>
```

Features:

- Subtle shadow
- Rounded corners
- Hover effects
- Optional image header
- Clean typography

Usage Guide

Applying the WebOS Theme

The webOS theme is applied globally through CSS variables. To use it:

1. **Global Application** (Already applied in root layout):

```
<body className="webos-theme" data-theme="webos">
```

1. **Component-Level Application**:

```
<div className="webos-theme">
  {/* Your components */}
</div>
```

Using WebOS CSS Classes

Layout Classes

```
// Container
<div className="webos-container">

// Flex layouts
<div className="webos-flex">
<div className="webos-flex-center">
<div className="webos-flex-between">

// Grid layouts
<div className="webos-grid webos-grid-3">
```

Card Classes

```
// Basic card
<div className="webos-card">
  <div className="webos-card-header">Header</div>
  <div className="webos-card-body">Content</div>
  <div className="webos-card-footer">Footer</div>
</div>

// Card carousel
<div className="webos-card-carousel">
  <div className="webos-card-carousel-item">
    <div className="webos-card">...</div>
  </div>
</div>
```

Button Classes

```
<button className="webos-button">Button</button>
<button className="webos-button webos-button-primary">Primary</button>
```

Typography Classes

```
<h1 className="webos-heading-1">Heading 1</h1>
<h2 className="webos-heading-2">Heading 2</h2>
<h3 className="webos-heading-3">Heading 3</h3>
<p className="webos-body">Body text</p>
<p className="webos-caption">Caption text</p>
```

Search Classes

```
<div className="webos-search-container">
  <div className="webos-search-label">JUST TYPE</div>
  <input className="webos-search-bar" />
</div>
```

Navigation Classes

```
// Top navbar
<nav className="webos-navbar">
  <div className="webos-navbar-content">...</div>
</nav>

// Bottom dock
<div className="webos-dock">
  <div className="webos-dock-icon">
    <Icon />
  </div>
</div>
```

Utility Classes

```
// Blur effects
<div className="webos-blur-surface">
<div className="webos-glass">

// Backgrounds
<div className="webos-gradient-bg">

// Scrolling
<div className="webos-smooth-scroll">
<div className="webos-hide-scrollbar">

// Animations
<div className="webos-animate-fade-in">
<div className="webos-animate-slide-up">
<div className="webos-animate-scale-in">
```

Migration Guide

Converting Existing Components to WebOS Theme

Step 1: Update Background Colors

```
// Before
<div style={{ background: '#f0f2f5' }}>

// After
<div style={{ background: 'var(--semantic-bg-base)' }}>
// or
<div className="webos-gradient-bg">
```

Step 2: Update Card Styling

```
// Before
<div className="bg-white rounded-lg shadow-md p-6">

// After
<div className="webos-card">
  <div className="webos-card-body">
```

Step 3: Update Typography

```
// Before
<h1 className="text-3xl font-light">

// After
<h1 className="webos-heading-1">
```

Step 4: Update Buttons

```
// Before
<button className="px-4 py-2 bg-blue-500 rounded">

// After
<button className="webos-button webos-button-primary">
```

Step 5: Update Search Components

```
// Before
<input type="text" className="rounded-full px-4 py-2" />

// After
<WebOSJustTypeSearch onSearch={handleSearch} />
```

Best Practices

1. Use Semantic Variables

Always use semantic CSS variables instead of hardcoded colors:

```
/* Good */
color: var(--semantic-text-primary);

/* Avoid */
color: #000000;
```

2. Maintain Consistency

Use webOS components and classes consistently across the app:

```
// Good
<div className="webos-card">
  <div className="webos-card-body">
    <h3 className="webos-heading-3">Title</h3>
    <p className="webos-body">Content</p>
  </div>
</div>

// Avoid mixing styles
<div className="bg-white rounded-lg p-4">
  <h3 className="webos-heading-3">Title</h3>
</div>
```

3. Respect the Minimalist Philosophy

Keep designs clean and uncluttered:

- Use white space generously
- Minimize color usage
- Prefer subtle shadows over heavy ones
- Keep borders light and subtle

4. Maintain Accessibility

- Ensure sufficient color contrast
- Provide focus indicators
- Use semantic HTML
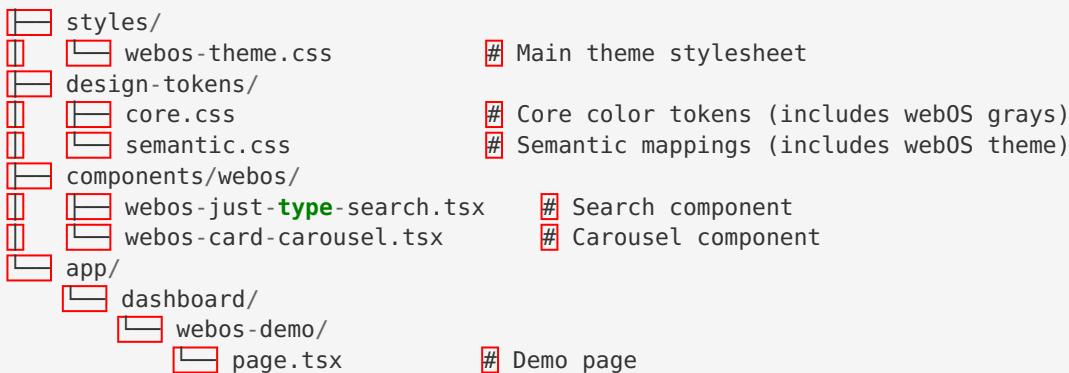- Support keyboard navigation

Demo Page

A comprehensive demo showcasing all webOS theme features is available at:

```
/dashboard/webos-demo
```

This page demonstrates:
- "JUST TYPE" search functionality
- Card carousel layout
- WebOS cards and buttons
- Bottom dock navigation
- Proper spacing and typography

File Structure

```
styles/
    webos-theme.css             # Main theme stylesheet
design-tokens/
    core.css                    # Core color tokens (includes webOS grays)
    semantic.css                # Semantic mappings (includes webOS theme)
components/webos/
    webos-just-type-search.tsx   # Search component
    webos-card-carousel.tsx      # Carousel component
app/
    dashboard/
        webos-demo/
            page.tsx        # Demo page
```

CSS Variables Reference

All webOS CSS variables are defined in `styles/webos-theme.css`:
- Colors: `--webos-*`
- Spacing: `--webos-space-*`
- Radius: `--webos-radius-*`
- Shadows: `--webos-shadow-*`
- Typography: `--webos-text-*`, `--webos-font-*`

- Transitions: `--webos-transition-*`
- Z-index: `--webos-z-*`

Semantic mappings in `design-tokens/semantic.css` :
- Applied via `[data-theme="webos"]` or `.webos-theme`
- Maps to core tokens from `design-tokens/core.css`

Browser Support

The webOS theme supports:
- Chrome/Edge 90+
- Firefox 88+
- Safari 14+
- Mobile browsers (iOS Safari, Chrome Mobile)

Features used:
- CSS Custom Properties
- CSS Grid and Flexbox
- Backdrop Filter (with fallbacks)
- Smooth Scrolling
- CSS Animations

Future Enhancements

Planned improvements:
1. Dark mode support for webOS theme
2. Additional card layouts (grid, masonry)
3. More animation presets
4. Theme customization panel
5. Additional webOS-inspired components

Support

For questions or issues related to the webOS theme:
1. Check this guide first
2. Review the demo page at `/dashboard/webos-demo`
3. Examine the CSS files in `styles/` and `design-tokens/`
4. Create an issue on GitHub

Version: 1.0
Last Updated: November 14, 2025
Author: loomOS Development Team