

loomOS Styling Playground - Quick Start

Get Started in 3 Steps

1. Navigate to the Playground

```
cd /home/ubuntu/code_artifacts/loomOS-styling-playground
```

2. Start the Development Server

```
npm run dev
```

3. Open in Browser

```
http://localhost:3000/dashboard
```

Key Files for Styling

Most Important Files

1. `app/dashboard/page.tsx` - Main dashboard with cards, carousel, dock
2. `app/dashboard/layout.tsx` - Top bar, app launcher, global search
3. `app/globals.css` - Global styles and design tokens
4. `tailwind.config.ts` - Tailwind configuration

Design System

- `design-tokens/core.css` - Colors, spacing, typography
- `design-tokens/semantic.css` - Component-specific tokens
- `design-tokens/motion.css` - Animation timings

Quick Styling Examples

Change Card Color

File: `app/dashboard/page.tsx`

```
// Line ~13-30: Find cardData array
const cardData = [
  {
    id: 'work-orders',
    color: '#YOUR_COLOR_HERE', // Change this!
  }
];
```

Change Background

File: `app/dashboard/page.tsx`

```
// Line ~600+: Find main div
style={{
  background: 'linear-gradient(135deg, #f0f0f0 0%, #e0e0e0 50%, #f0f0f0 100%)',
}}
```

Change Top Bar Color

File: `app/dashboard/layout.tsx`

```
// Line ~90+: Find SystemStatusBar nav element
style={{
  backgroundColor: '#YOUR_COLOR', // Change this!
}}
```

Adjust Card Size

File: `app/dashboard/page.tsx`

```
// Line ~430+: Find card style
style={{
  width: '450px', // Change from 400px
  height: '350px', // Change from 300px
}}
```

Common Tasks

Task	File	What to Change
Card colors	<code>app/dashboard/page.tsx</code>	<code>cardData</code> array, <code>color</code> property
Dashboard background	<code>app/dashboard/page.tsx</code>	Main div background gradient
Top bar styling	<code>app/dashboard/layout.tsx</code>	<code>SystemStatusBar</code> nav element
Dock appearance	<code>app/dashboard/page.tsx</code>	Bottom fixed dock div
Border radius	<code>tailwind.config.ts</code>	<code>borderRadius</code> section
Shadows	<code>tailwind.config.ts</code> or inline	<code>boxShadow</code> values
Animations	<code>app/dashboard/page.tsx</code>	<code>transition-all</code> <code>duration-XXX</code> classes

Useful Commands

```
# Start dev server
npm run dev

# Start on different port
PORT=3001 npm run dev

# Clear cache and restart
rm -rf .next && npm run dev

# Check for errors
npm run lint

# Type check
npm run type-check
```

Documentation

- `PLAYGROUND_GUIDE.md` - Complete styling guide (READ THIS FIRST!)
- `DASHBOARD_STRUCTURE.md` - Component structure reference
- `QUICK_START.md` - This file
- `README.md` - Original project README

Pro Tips

1. **Use Browser DevTools:** Right-click → Inspect to see live styles
2. **Make Small Changes:** Test one change at a time
3. **Hard Refresh:** `Ctrl+Shift+R` to clear browser cache
4. **Use CSS Variables:** Prefer `var(--semantic-bg-muted)` over hex colors
5. **Test Responsive:** Use DevTools responsive mode (`Ctrl+Shift+M`)

Design Token Shortcuts

Colors

```
// Instead of: backgroundColor: '#f0f0f0'
// Use: backgroundColor: 'var(--bg-surface)'

// Common tokens:
var(--semantic-text-primary)
var(--semantic-bg-muted)
var(--chrome-darker)
var(--glass-white-60)
var(--accent-blue)
```

Spacing

```
// Instead of: padding: '16px'
// Use: className="p-base"

// Or: padding: 'var(--space-base)'
```

Shadows

```
// Instead of: boxShadow: '0 4px 8px rgba(0,0,0,0.1)'
// Use: boxShadow: 'var(--shadow-card)'
```

Quick Animation Changes

Speed up/slow down

```
// Find: className="transition-all duration-500"
// Change to: duration-300 (faster) or duration-700 (slower)
```

Change easing

```
// Find: ease-out
// Try: ease-in-out, ease-in, or cubic-bezier(0.4, 0, 0.2, 1)
```

Testing Checklist

Before committing changes:

- [] Looks good on desktop (1440px)
- [] Looks good on tablet (768px)
- [] Looks good on mobile (375px)
- [] Animations are smooth
- [] Text is readable (good contrast)
- [] No console errors
- [] Hover states work
- [] Focus states visible (tab navigation)

Common Issues

Changes not showing?

```
# Hard refresh browser
Ctrl+Shift+R (Windows/Linux)
Cmd+Shift+R (Mac)

# Clear Next.js cache
rm -rf .next
npm run dev
```

Server won't start?

```
# Kill existing process
pkill -f "next dev"

# Clear and reinstall
rm -rf node_modules
npm install
npm run dev
```

TypeScript errors?

```
// Add at top of file
// @ts-nocheck

// Or above specific line
// @ts-ignore
```

Next Steps

1. **Read the full guide:** Open `PLAYGROUND_GUIDE.md`
2. **Explore the code:** Browse `app/dashboard/` files
3. **Make a small change:** Try changing a card color
4. **Test it:** Start the server and view in browser
5. **Iterate:** Make changes, refresh, repeat
6. **Document:** Keep notes on what works well
7. **Commit:** Use git to save your progress
8. **Create PR:** When ready, push to a branch and open PR

Need Help?

1. Check `PLAYGROUND_GUIDE.md` for detailed instructions
2. Check `DASHBOARD_STRUCTURE.md` for component locations
3. Use browser DevTools to inspect elements
4. Search the codebase with `grep -r "search-term" app/`

Color Palette Quick Reference

Current Dashboard Colors

Backgrounds:

- Main BG: Gradient from `--semantic-bg-muted` to `--semantic-bg-subtle`
- Cards: `#f0f8e8` (green), `#f8f8f8` (gray), `var(--semantic-surface-base)`
- Top Bar: `var(--chrome-darker)` (very dark)

Icons/Accents:

- Blue: `#7a9eb5`
- Brown: `#b58a7a`
- Green: `#8ab57a`
- Tan: `#b5a07a`

Text:

- Primary: `var(--semantic-text-primary)`
- Secondary: `var(--semantic-text-secondary)`
- On Chrome: `var(--chrome-text)`

Ready to Style!

You're all set! Start exploring and experimenting with the dashboard design.

Remember: This is your playground - feel free to try bold ideas!

Quick Command Reference:

```
cd /home/ubuntu/code_artifacts/loomOS-styling-playground
npm run dev
# Then open: http://localhost:3000/dashboard
```