



WORKSHOP: COLOR-THEMING

Ron Lucke – elan e.V.

Aktuelle Herausforderungen

- Inkonsistente Farbverwendung erschwert Anpassung
- Keine einfache Definition oder Änderung von Themes
- Anforderungen an Barrierefreiheit nicht konsequent abgesichert
- Unterschiedliche Hochschul-CI schwer umsetzbar
- Was ist mit den Icons?

Erwartungen und Anforderungen

- Einheitliches Farbmanagement für das gesamte System
- Flexible, aber kontrollierte Anpassung an Hochschul-CI
- Sicherstellung der Barrierefreiheit ohne separate High-Contrast-Modi
- Anpassbare UI-Elemente bei gleichzeitiger Konsistenz

Technische Möglichkeiten

Design Tokens für Farben mit semantischer Benennung

- Einheitliche Benennung für eine klare Struktur
- Farbbezeichnungen sind sprechend
- Gefahr zu vieler Variablen

Kontrast-Checker in der Admin-Oberfläche

- Admin wird unterstützt die richtige Farbkombination zu wählen
- Eine Vorschau für die Einstellungen schafft Klarheit

Icons und ihre Farbgebung

- Welche Icons sollen CI-Farben übernehmen?
- Wann ist eine neutrale Farbgebung besser?
- Herausforderung: Icons in Dark- und Light-Mode
- Verschiedene Möglichkeiten mit Icons in einem System umzugehen

Geplante Schritte

Einführung standardisierter Design Tokens für Farben

- Fortsetzung der bisherigen Strategie die mit 6.0 eingeführt wurde
- Allgemeingültige Bezeichnungen für Farbwerte
- Spezifischere Bezeichnungen in den einzelnen SCSS Dateien

Anpassung der Icon Einbindung in Stud.IP

- Icons nur noch ohne fill bereitstellen → entspricht nur noch **black**
- Per Webpack aus dem Icon Verzeichnis ein Spritesheet generieren
- Vue und PHP laden svg direkt und nicht als img, mit fill="currentColor" lässt sich dann per CSS die Farbe setzen
- background-image nutzt als Fallback die SVG Dateien, Form per mask, Farbe per background-color

Verbesserung der Hochschulspezifischen Anpassungen

- brand-color ist nicht mehr direkt alles was blau ist
- Klar definierte Bereiche deren Farben man anpassen kann

Trennung zwischen CI-, funktionalen, highlight und System Farben

- CI-Farben → Hochschulspezifische Anpassungen
- Funktionale Farben → Einheitlich für alle (z. B. Fehlermeldungen, Buttons, Navigation)
- Highlights → besondere Hervorhebungen (z.B. Call-To-Action, Links etc.)
- System → Grundgerüst (Fieldset, Contentbox etc.)

Sicherstellung von Mindestkontrasten

Ziel ist es in allen Bereichen WCAG
AAA zu erreichen

Theme-Preview

- Visuelle Prüfung der Kontrastwerte → kann mit DevTools geprüft werden
- Vorschau für die Anpassungen

Vorbereitung für weitere Modi

Dark-Mode

- Ersetzt helle Hintergründe durch dunkle, um Augenbelastung zu reduzieren
- Passt Kontraste und Farben an, um gute Lesbarkeit zu gewährleisten
- Unterstützt Nutzerpräferenzen und Systemeinstellungen automatisch

High-Contrast

- Maximiert den Farbkontrast für bessere Sichtbarkeit und Lesbarkeit
- Reduziert visuelle Ablenkungen und verbessert Barrierefreiheit
- Ermöglicht Nutzung in Kombination mit Betriebssystem-High-Contrast-Einstellungen



Wo Licht ist, ist auch Schatten

Brauchen wir einen High-Contrast-Mode?

Guter Kontrast ist nicht optional

- Bessere UX für alle Nutzenden
- High-Contrast für Betriebssysteme unterstützen
- Windows Einstellungen überschreiben



Weiterführende Entwicklungen

- Ergänzung weiterer Design Tokens (z. B. Schriftarten)
- Nutzergruppenspezifische UI-Anpassungen
- Automatisierte Prüfmechanismen für Barrierefreiheit

Zukunftsperspektiven

- Welche Mechanismen braucht Stud.IP für eine flexible CI-Anpassung?
- Wollen wir mehr als nur Farben anpassbar machen?
- Was sind eure Ideen und Wünsche?



**VIELEN DANK FÜR DIE
AUFMERKSAMKEIT**