

# 1 Band C

## 1.1 Elemente einer Benutzerschnittstelle kennen und anwenden

### 1.1.1 Controls und Widgets erklären

#### 1.1.1.1 Was sind controls und Widgets

Controls oder Widgets sind grafische Steuerelemente einer Benutzeroberfläche. Sie ermöglichen dem Benutzer, mit einem Programm zu interagieren, Eingaben zu machen und Informationen zu erhalten. Controls und Widgets sind nicht alle Interaktionselemente, da mache nur für die datenübergabe an den Nutzer sind.

#### 1.1.1.2 Häufig verwendete Controls und Widgets

Widget / Control	Beschreibung	Wichtige Eigenschaften
Button (Schaltfläche)	Ein klickbares Element, das eine Aktion auslöst.	Text, Icon, aktiv/deaktiviert
Label (Beschriftung)	Zeigt Text oder Information an, keine Interaktion.	Text, Schriftart, Ausrichtung
Textfeld (Textbox)	Ermöglicht die Eingabe von Text durch den Benutzer.	Inhalt, Länge, Passwortcharakter
Checkbox (Kontrollkästchen)	Ermöglicht das An- oder Abwählen einer Option.	An/Aus-Zustand
Radiobutton (Optionsfeld)	Ermöglicht das Auswählen einer Option aus einer Gruppe.	Gruppenzugehörigkeit, Aktiv
Dropdown (Auswahlliste)	Ermöglicht die Auswahl eines Werts aus einer Liste.	Ausgewählter Wert, Liste
Slider (Schieberegler)	Ermöglicht das Einstellen eines Werts innerhalb eines Bereichs.	Minimalwert, Maximalwert
Progressbar (Fortschrittsanzeige)	Zeigt den Fortschritt eines Vorgangs an.	Minimalwert, Maximalwert
Image (Bild)	Zeigt ein Bild oder Icon an.	Dateipfad, Größe, Skalierung

### 1.1.2 Controls und Widgets sinnvoll verwenden

#### 1.1.2.1 was bedeutet sinnvoll plziert

Sinnvoll plazierte Widgets und Controls sorgen dafür das der Nutzer die Seite möglichst Intuitive und Effizient navigieren kann. Unter anderem sollten einfach erkennbare Symbole verwendet werden und diese sollte an den gewohnten stellen plziert sein. Wenn ein User nach etwas sucht, dann sollte er ohne lange suchen zu müssen fündig werden. Wir haben unsere Symbole und die Plzierung der Element, vor allem in der Nav bar, von der SRF Webseite inspirieren lassen, da diese Seite stark besucht ist und damit unsere Applikation unseren Usern schon bekannt vor kommen sollte.

#### 1.1.2.2 Sinnvolle Umsetzung

In userer Umsetzung haben wir die ERLernte Theorie angewendet. Die Navbar und die Suchfunktion ist wie bei SRF und sollte den meisten Kunden bereits bekannt vorkommen. Der Hauptinhalt der Seite ist die Interaktive Karte. Unsere Footer Bar besteht aus 3 gängigen Symbolen.

- QR-Code
- Map
- Einstellungszahnrad

Das sind alles sehr gängige Symbole welche für fast jeden sofort verständlich sein sollten.

Unsere Search funktion ist auch auf SRF basiert und ist so Intuitive wie möglich gestaltet. Alle möglichen Aktionen sind sofort Zentral ersichtlich. Der hintergrund wird aus dem Fokus genommen damit die neuen Interaktionsmöglichkeiten herausstechen und die zuvor besuchte Seite immer noch so halb ersichtlich ist, dies sollte sicherstellen das die Nutzer wissen das die vorherige Seite nicht verloren gegangen ist.

Die Settingspage ist bekannten überkategorien organisiert mit der Accessibility ganz oben damit man schnell den Screen-reader oder die Schriftgröße zu seinen präferenzen ändern kann.<

### 1.1.3 Controls und Widgets in eine Logische abfolge bringen

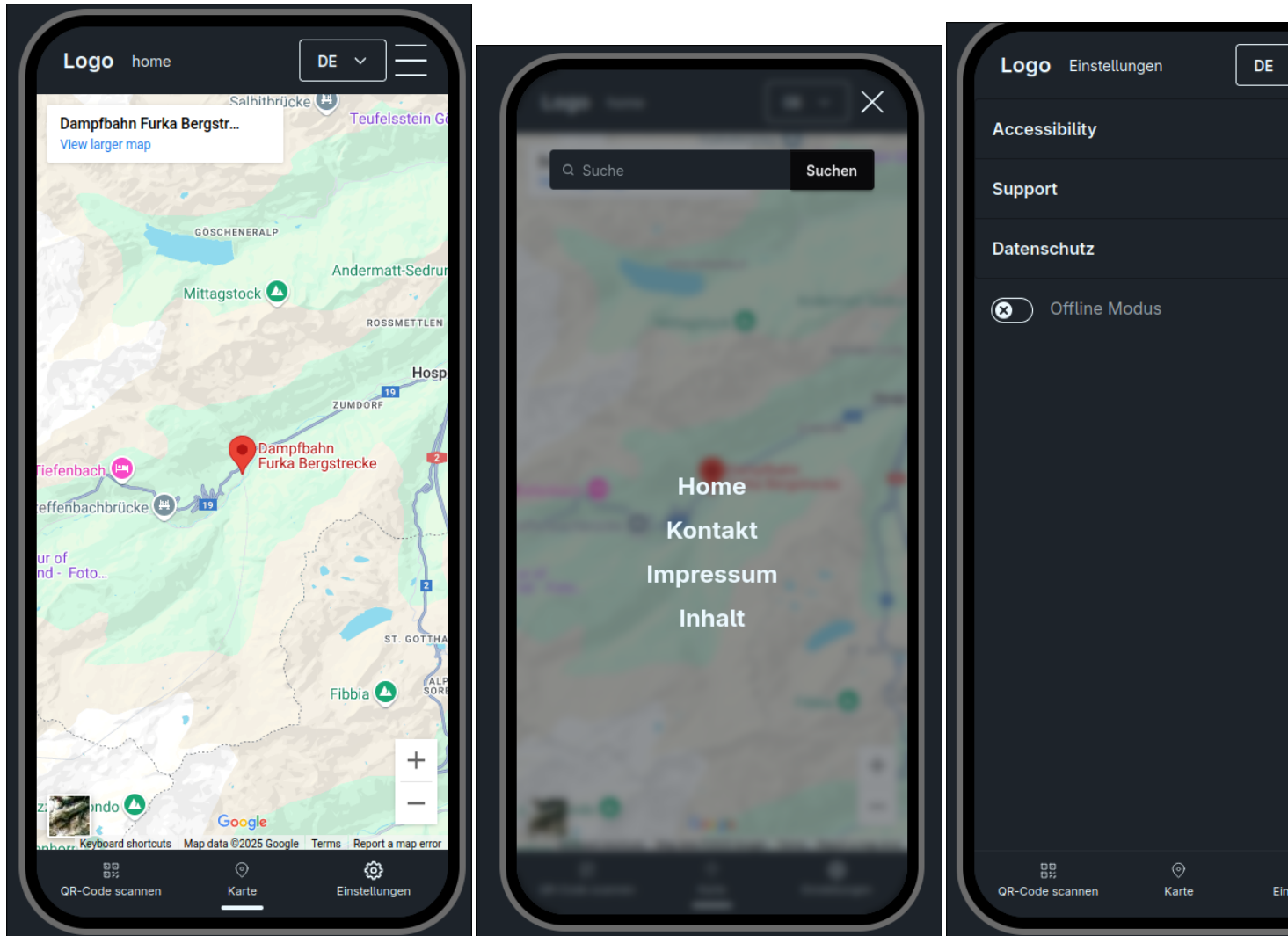
Durch Umfragen mit mehreren Dritten personen und Inspirationen von verschiedenen Webseiten haben wir die Plzierung und die Zusammenareit unserer Controls und Widgets Optimiert. Wir haben unseren Fokus auch auf einen Logischen Aufbau gelegt damit es für jeden, auch für Kinder oder Ältere personen, einfach verstehen können. Unsere Applikation ist jetzt sehr intuitive und sollte für jeden brauchbar sein.

## 1.2 Benutzerschnittstelle erstellen und testen

### 1.2.1 Skizzen als Prototyp umsetzen

Ich habe mich dazu entschieden, den Prototypen als Webseite zu bauen. Bei der Arbeit entwickle ich auch Webseiten. Deshalb geht das für mich einfacher, als ein neues Tool zu lernen. Wenn wir dann die richtige App programmieren würden, könnten wir auch den Code aus dem Prototypen weiter verwenden.

Hier ist der Link zur [Webseite](#)



### 1.2.2 Testen und Probleme identifizieren

- Den Prototypen habe ich nur für die Mobile ansicht optimiert. Im Desktopmodus sieht er nicht so gut aus.
- Wenn man mit dem Menu von einer Seite zu einer anderen wechselt bleibt das Menu offen. Das ist nicht intuitiv.
- Da ich im Browser den Dark mode verwende, sieht sie im light mode nicht so gut aus.

### 1.2.3 Verbesserungen

- Da der Prototyp nur für Mobile optimiert sein muss, könnte ich es auf grossen Bildschirmen in ein Handy packen. So kann man die Webseite trotzdem gut testen.
- Wenn im Menu eine Seite gewählt wird, werde ich machen, dass das Menu automatisch schliesst.
- Ich werde die Webseite im light mode testen.

## 1.3 Benutzerfreundlichkeit verbessern

### 1.3.1 Schwierig unzusetzende Elemente

- Dropdown schließt nicht automatisch bei Klick außerhalb

- 
- Fehlende Keyboard-Navigation (Tab, Pfeiltasten, Enter)
  - Schwierige Positionierung bei unterschiedlichen Bildschirmgrößen
  - Fehlende Barrierefreiheit (ARIA-Rollen und Attribute)
  - Probleme bei responsivem Verhalten auf kleinen Bildschirmen
  - Unklare oder zu kleine Klickflächen
  - Schwierige State-Verwaltung (Öffnen/Schließen, Selektion)

Wir haben uns entschieden spezifisch die folgenden Punkte zu berücksichtigen.

- Schwierige Positionierung bei unterschiedlichen Bildschirmgrößen
- Probleme bei responsivem Verhalten auf kleinen Bildschirmen
- Unklare oder zu kleine Klickflächen

### **1.3.2 Verbesserungen Unterbreiten**

Als Verbesserung werden wir anstatt unsere Buttons Anzuschreiben Symbole verwenden. Dies sollte gut die Grösse des Buttons Klar.

Die Applikation werden wir auch komplett dynamisch gestalten damit diese auf allen möglichen Bildschirmen Funktioniert.

### **1.3.3 Implementation**