

Modulüberblick M291: Web-Frontend

Worum geht es in diesem Modul?

Sie lernen, wie moderne Weboberflächen aufgebaut werden: von **UI-Design verstehen** über **HTML/CSS/JavaScript** bis zu **Vue, APIs, Animationen** und **Deployment**.

Wir arbeiten projektbasiert: Sie bauen mehrere kleine Frontend-Projekte, die Schritt für Schritt anspruchsvoller werden.

Ablauf in Themenblöcken

- **Block 1: UI-Designs interpretieren & Projektstart (HTML/CSS)**
 - Kursorganisation und Arbeitsweise (Repo, Ordner, Abgaben)
 - UI-Designs lesen: Bereiche erkennen (Header, Navigation, Hero, Sections)
 - Projektsetup: „**Landingpage Hero Alvarado**“
 - HTML-Grundstruktur: header, nav, main, section
 - CSS-Grundlagen: Container, Layout-Wrapper, einfache Flexbox-Anordnung
- **Block 2: Refresher CSS (Media Queries, Fonts, Variablen)**
 - CSS-Variablen (':root') für Farben/Abstände/Typografie
 - Google Fonts einbinden
 - Media Queries: Desktop → Mobile Layout
 - Weiterarbeit am Projekt: „**Landingpage Hero Alvarado**“ (Layout & Responsive)
- **Sportferien**
 - Kein Unterricht
- **Block 3: UI-Designs umsetzen mit JavaScript (HTML/CSS/JS)**
 - JavaScript-Refresher: Variablen, Funktionen, Events (kurz & praxisnah)
 - DOM: Elemente auswählen und verändern (querySelector, getElementById)
 - Klassen setzen/togglen (classList)
 - Projekt: **Dark/Light Mode Switch** für die Landingpage
- **Block 4: DOM-Manipulation, Events, State & Accessibility (FAQ Accordion)**
 - DOM-Events & Zustand (state): „offen/zu“ als Logik verstehen
 - Projektstart: „**FAQ Accordion**“
 - Interaktion: Klick, Tastatur, Fokus
 - Accessibility-Grundlagen: aria-expanded, sinnvolle Buttons, Focus Styles
- **Block 5: Transitions & Accessibility vertiefen (FAQ Accordion)**
 - CSS Transitions (smooth auf/zu)
 - Barrierefreiheit prüfen und verbessern (Keyboard Navigation, Kontraste)
 - Projektabschluss: FAQ Accordion „ready“
- **Block 6: Formulare & Validierungen**
 - Input-Typen (Text, Email, Password, Checkbox, Radio)
 - Fehlermeldungen anzeigen (visuell + verständlich)
 - Validierung (Basis): required, pattern, min/max

- Mini-Übungen: Fehlerzustände und User Feedback
- **Block 7: Debugging & Refactoring**
 - Debugging mit DevTools (Console, Elements, Sources)
 - Code sauber machen: Namen, Struktur, Wiederverwendung
 - JavaScript: Import/Export (Grundidee modularer Code)
 - Externe Stylesheets / Skripte sauber einbinden
 - Kurzer Quiz-/Checkpoint zur Standortbestimmung
- **Block 8: Frameworks - Einführung Vue**
 - Warum Frameworks? (Komponenten, State, Wiederverwendung)
 - Vue Setup und Grundprinzipien
 - Erste Bindings und Events (v-bind, v-on, v-model)
- **Block 9: Vue anwenden - Projekt in Vue umsetzen**
 - Vue Repetition: Bindings, Events, Conditionals
 - Projekt: „**FAQ Accordion**“ in Vue (State im Framework)
 - Ziel: gleiche Funktion wie vorher – aber als Komponenten gedacht
- **Frühlingsferien**
 - Kein Unterricht
- **Block 10: Vue Komponenten & Props**
 - Komponenten erstellen und verschachteln
 - Props/Events (Kommunikation zwischen Komponenten)
 - Projekt: FAQ Accordion mit sauberen Komponenten (AccordionItem)
- **Block 11: Vue + fetch (Daten aus API laden)**
 - Daten von externer API laden (fetch, async/await)
 - Loading-/Error-States anzeigen
 - Daten als Liste rendern (v-for)
 - Verbindung zu Backend/Modul 290 (falls vorhanden): „echte“ Daten nutzen
- **Block 12: Animationen & Transitions**
 - CSS Transitions wiederholen und erweitern
 - CSS Animationen: Keyframes, Timing, einfache Effekte
 - Kleine UI-Microinteractions (Hover, Buttons, Panels)
- **Block 13: Canvas / SVG**
 - SVGs erstellen und manipulieren (Icons, kleine Grafiken)
 - Canvas Einstieg: einfache Zeichnungen / kleine Charts
 - Ziel: eine visuelle Darstellung, die im Browser live gerendert wird
- **Ferien (QV-Woche)**
 - Kein Unterricht
- **Block 14: Testing, Deploy & Performance**
 - Testing-Grundlagen (manuell, Checklisten, Cross-Browser)
 - Performance: Bildgrößen, Laden, „nicht unnötig schwer“
 - Minification & Bundling (Grundprinzip)
 - Deploy einer Web-App (z.B. GitHub Pages/Netlify)
- **Block 15: Leistungsbeurteilung 02**

- Praktische Umsetzung eines Features / Mini-Projekts unter Vorgaben
- Bewertung: Funktion, Struktur, Sauberkeit, Responsive/A11y-Basics
- **Block 16: Neue CSS/JS Features (Auswahl)**
 - Beispielthemen: Scroll-Animationen, moderne CSS-Features, kleine Browser-APIs
- **Block 17: Abschluss Modul**
 - Reflexion: Was kann ich jetzt?
 - Aufräumen & Portfolio (README, Screenshots, Links)

From:

<https://wiki.bzz.ch/> - **BZZ - Modulwiki**

Permanent link:

https://wiki.bzz.ch/de/modul/m291/learningunits/lu01/theorie/0_modulaufbau

Last update: **2026/01/31 14:11**

