



# ABB Technikerschule

Technik. Informatik. Wirtschaft. Management →

# NDS - Web Engineering

Einführung & Grundlagen 1



## Steckbrief Dozent

*Key*      *Value*

---

Name      **Hannes Morgenthaler**

---

Alter      Über **40x** um die 🌟

---

Herkunft      Aus dem Westen der CH  
(Fribourg), Erde 🌎 (*angeblich*)

---

Abschluss      BSc. Informatik FH

---

Job      Lead Software Developer,  
INGTES AG

---

Erfahrung      Webseiten & Frontend: ~20 Jahre

Fullstack Developer: 12+ Jahre

Dozent: ~1 Jahr

---

# Frage: wie schätzt DU dein Vorwissen zu diesem Modul ein?

Skala	Beschreibung	Anzahl Stimmen
1 😳	Webbrowser, was ist das...?	
2 😳 😳	Web-Power-User mit Adblockern und weiteren Extensions	
3 😳 😳 😳	Hab schon mal was von HTML/CSS/JS gehört...	
4 😃 😃 😃 😃	Habe schon selber HTML/CSS/JS geschrieben	
5 😃 😃 😃 😃 😃	Hab schon selber Webseiten und/oder Web-Apps erstellt/gestaltet	

# Welche Erwartungen hast **DU** an dieses Modul?

- Diese **Kompetenz / Fähigkeit** möchte ich gerne erlernen
- Dieses **Thema** oder diese **Technologie** interessiert mich

## Auftrag

1. Überlege dir **2 bis max. 3 Stichworte / Begriffe**
2. Tritt an die **Wandtafel** und **schreibe** diese auf
3. **Existiert** ein Stichwort schon, mach einen **Strich dahinter**

# Zielsetzungen des Dozenten

Ich als Student **kenne die Grundlagen** von

- HTML / CSS / JS
- (sicherer) **Kommunikation** zwischen **Client** und **Server**
- von Responsive Webdesign (*SPA passt sich an Displaygrösse an*)
- von Progressive Web Apps (PWAs) (*SPA kann offline Inhalte speichern und lässt sich installieren*)

Ich als Student **kann selbstständig**

- einen einfachen **WebHost** mit **Ktor** in **Kotlin** programmieren
- eine einfache **Single-Page-App (SPA)** mit **Vue.js** programmieren
- ein **einfaches Design** gemäss einer **Vorlage** in **HTML** / **CSS** umsetzen

# Geplantes Kursprogramm

Reihenfolge und Inhalt von Themen wird möglichst den Bedürfnissen angepasst und kann daher ggf. leicht ändern.

1. Grundlagen
  1. Das **World Wide Web (WWW)**
  2. HTML & CSS
  3. Funktionalität eines **Webbrowsers**
  4. **Document Object Model (DOM)**
  5. **JavaScript & TypeScript**
  6. erste **Dynamische Webseite** von 
  7. **HTTP(S), Sessions & Cookies**
  8. Funktionalität eines **Webservers**
2. **Frontend:** Single Page Applications mit `Vue.js`
3. **Backend:** Webserver mit `Ktor` und `Kotlin`
4. **Responsive Web Design**  
*(Dynamische Anpassung an Bildschirmgrößen)*
5. Weiterarbeit am Projekt **Smart Home**
  1. Programmieren eines **Firefighter-Dashboards**  
*(Front- und Backend)*
  2. Anbindung ans Smart-Quartier
  3. Alarme, Events, Aktionen
6. **Progressive Web Apps**  
*(Offline, Service-Worker, Installation im OS)*

# Allgemeine Anmerkungen zum Kursinhalt

- Das Thema "Web Engineering" ist sehr umfangreich und vielfältig
- Der zeitlich begrenzte Umfang dieses Moduls wird leider bei Weitem nicht reichen, **alle relevanten Aspekte** ausreichend zu beleuchten - es musste (notgedrungen) an diversen Stellen **erhebliche Kürzungen** vorgenommen werden... 😢
- Dieser Kurs fokussiert sich auf die **allerwesentlichsten Aspekte**, die nach Ansicht des Dozenten zur **praktischen Anwendung** notwendig sind.
- In diesem Kurs kommen konkrete **Techniken, Tools und Frameworks** als **eine (!) Möglichkeit des Web Engineering** zum Einsatz, die vom Dozenten und/oder der Schule bevorzugt werden und sich in der Praxis bewährt haben - aus Zeitgründen können kaum Alternativen betrachtet werden. Es sei jedoch angemerkt, dass es für **fast jedes Tool meist dutzende gleichwertige Alternativen** gibt → bei Interesse kann der Dozent gerne Auskunft geben/beraten.

# Allgemeine Regeln und Grundsätze 1

## Ablauf des Unterrichts

- **Qualität vor Quantität:** lieber ein Thema verschieben und dafür den restlichen Stoff richtig verstehen - bitte gebt dem Dozenten Rückmeldung, wenn er es selbst nicht merkt... 🤔
- **Kurstage nach ABB-TS Standard:** jeweils 4 Lektionen à 45 Minuten (jeweils 5' Pause dazwischen)
- In **gemeinsamer** Vereinbarung können situativ Anpassungen vorgenommen werden.

## Eigenverantwortung der Studenten

- **Pünktlichkeit:** unsere Zeit ist (leider) knapp bemessen - rechtzeitig starten = rechtzeitig beenden (gilt auch für Pausen)!
- **Keine Präsenzpflicht:** wer fehlt holt Stoff **selbstständig** nach
- **Wenn etwas nicht verstanden wurde:** fragen! (notfalls auch mehrfach, ich erkläre gerne nochmals 🤝)
- **Hausaufgaben:** dienen der **Vertiefung** der Erkenntnisse und sind **unerlässlich**. Das **selbstständige Lösen der Hausaufgaben** wird beim nächsten Kurstag **vorausgesetzt**.

# Allgemeine Regeln und Grundsätze 2

## Umgang Miteinander

### Gegenseitiger Respekt und Toleranz

- **Wir sind alle Erwachsenen:** Pöbeleien, Provokationen, Mobbing, usw. werden **nicht geduldet** und ggf. geandert.
- **Es gibt keine dummen Fragen:** jede Person bringt ihr eigenes Vorwissen mit, lernt anders und im eigenen Tempo - wir **nehmen aufeinander Rücksicht** und geben einander **wertschätzend Feedback**.

### Unklarheiten, Fragen und Kritik

- **Verständnisprobleme:**
  - wenn möglich direkt im Unterricht ansprechen
  - Schriftliches Kontaktieren des Dozenten via **MS Teams** oder **E-Mail** (*Antwort i.d.R. innerhalb von 24h*)
- **Individuelle Lösungen:** falls die eine oder andere Gruppe **stark benachteiligt** oder **stark über- oder unterfordert** ist, werden individuelle Lösungen gesucht wie z.B. **Zusatzunterricht** oder **Gruppenaufträge zum selbständigen Erarbeiten und Präsentieren eines Themas**.

# Allgemeine Regeln und Grundsätze 3

## Präsenzunterricht ist zum Lernen da

- **Keine Lust oder Zeit:** Wer mal an der Lektion **nicht mitmachen will** oder **nicht kann** z.B. wegen geschäftlicher Verpflichtungen, Stress oder Erschöpfung - kein Problem ( *Eigenverantwortung!*) - bitte einfach den **Unterrichtssaal verlassen** um den Lernfluss der Mitstudenten nicht zu stören!
- **Fragen:** dürfen und sollen jederzeit gestellt werden können - einfach **alles mit Mass:** wenn der Fluss oder Zeitplan des Unterrichts erheblich gestört wird, werden Fragen ggf. auf später vertagt.



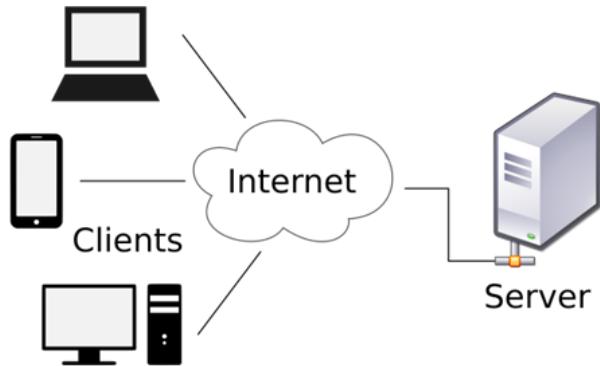
# Das World Wide Web

- ENG für "Weltweites Netz", kurz **Web** oder auch **WWW** genannt.
- Ein **über das Internet** abrufbares System von elektronischen **Hypertext-Dokumenten**, sogenannten **Webseiten**
- Umgangssprachlich wird das **WWW** oft mit dem **Internet** gleichgesetzt; das **WWW** ist jedoch **jünger** und stellt **nur eine** von mehreren möglichen **Nutzungen des Internets** dar. Andere Internetdienste wie z.B. **E-Mail** sind **nicht** in das **WWW** integriert.
- Das **WWW** wurde 1989 von **Tim Berners-Lee** und **Robert Cailliau** am **CERN** in **Genf**

# Das World Wide Web 2

- Tim Berners-Lee entwickelte dazu das **HTTP-Netzwerkprotokoll** und **HTML**. Zudem programmierte er den ersten **Webbrowser** und die erste **Webserver-Software**. Er betrieb auch den **ersten Webserver der Welt** auf seinem Entwicklungsrechner 😎💪!
- Das Gesamtkonzept wurde der Öffentlichkeit 1991 unter **Verzicht** auf jegliche **Patentierung** oder **Lizenzzahlungen** zur **freien Verfügung** gestellt, was **erheblich zur heutigen Bedeutung beitrug**.
- Die weltweit erste Webseite info.cern.ch wurde am 6. August 1991 veröffentlicht.
- Das WWW führte zu umfassenden, oft als revolutionär beschriebenen **Umwälzungen in vielen Lebensbereichen**, zur **Entstehung neuer Wirtschaftszweige** und zu einem grundlegenden **Wandel des Kommunikationsverhaltens** und der Mediennutzung. Es wird in seiner **kulturellen Bedeutung**, zusammen mit anderen Internet-Diensten wie E-Mail, teilweise mit der **Erfindung des Buchdrucks** gleichgesetzt

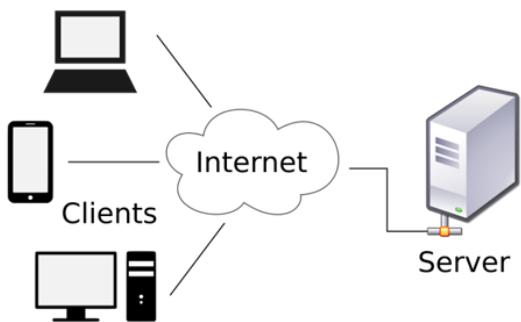
# WWW - Funktionsweise



- *Drei Kernstandards*
  - **HTTP** als **Protokoll**, mit dem der Browser Informationen vom Webserver anfordern kann
  - **HTML** als **Auszeichnungssprache**, die festlegt, wie die Information gegliedert ist und wie die Dokumente verknüpft sind (**Hyperlinks**)
  - **Uniform Resource Identifier** (URI) als eindeutige **Bezeichnung** einer Ressource, die in **Hyperlinks** verwendet wird

# WWW - Funktionsweise 2

- *Erweiterungen*
  - **Cascading Style Sheets** (CSS) beschreiben das **Aussehen** und die **Andordnung** der Elemente einer Webseite, womit der Inhalt von dessen Darstellung separiert wird
  - **Document Object Model** (DOM) als **Programmierschnittstelle** für externe Programme oder Skriptsprachen (wie JavaScript) von Webbrowsern



INSTALLED

- Auto Close Tag** ⌚ 3ms ⚙️  
Automatically add HTML/XML close tag, same as Vis...  
Jun Han
- Auto Complete Tag** ⌚ 3ms ⚙️  
**TAG** Extension Packs to add close tag and rename paired ...  
Jun Han
- Auto Rename Tag** ⌚ 12ms ⚙️  
**<TAG>** **</TAG>** Auto rename paired HTML/XML tag  
Jun Han
- Live Server** ⌚ 15ms ⚙️  
**((•))** Launch a development local Server with live reload f...  
Ritwick Dey
- Prettier - Code formatter** ⌚ 39ms ⚙️  
**prettier** Code formatter using prettier  
Prettier
- vscode-icons** ⌚ 20ms ⚙️  
**VSCode Icons Team** Icons for Visual Studio Code  
VSCode Icons Team

# Auftrag 1 - Minimale Entwicklungsumgebung einrichten

1. Installieren von Visual Studio Code  
(VsCode): <https://code.visualstudio.com>
2. Installieren von Erweiterungen:

