

# Frontend und Backend – Was ist der Unterschied?

## Inhaltsverzeichnis

### Frontend und Backend – Was ist der Unterschied?

#### ► Was bedeuten Frontend und Backend?

#### Was ist ein Full-Stack Entwickler?

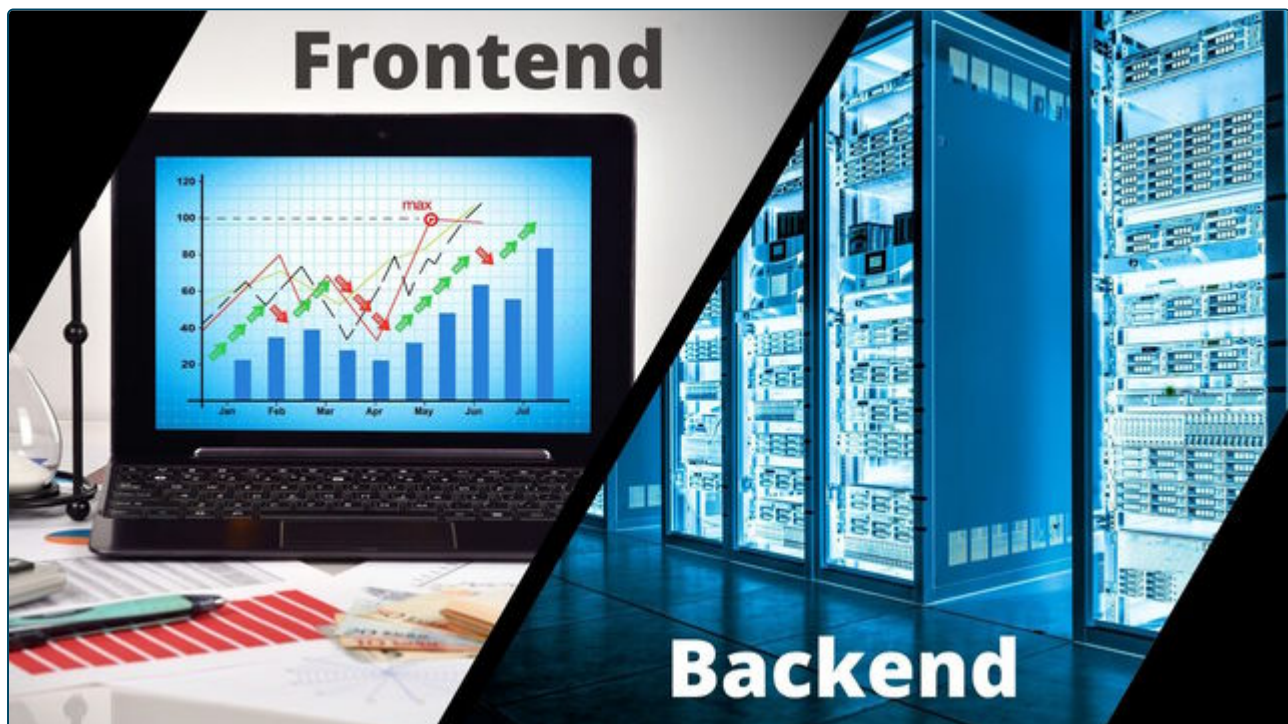
#### ► Frontend und Backend im Digital Signage

#### Warum die Trennung in Frontend und Backend?

#### Produktberatung und Kontakt

#### Unsere Kontaktdaten

In diesem Beitrag dringen wir etwas tiefer in den Tekkie-Bereich ein. Wir beschäftigen uns mit zwei Fachbezeichnungen, die IT-Menschen gerne von sich geben. Da diese Begriffe in verschiedenen Umfeldern auftreten, tragen sie einiges zur Verwirrung bei. Lernen Sie also den Backend-Frontendunterschied anhand konkreter Beispiele aus der [Digital Signage Branche](#) kennen.



Kostenfreie Beratung sichern

## Was bedeuten Frontend und Backend?

Komplexe Softwarelösungen, besonders webbasierte, lassen sich in zwei Bereiche unterteilen. Der Begriff **Frontend** beschreibt das für den Anwender sichtbare und interaktive Benutzerinterface. Das **Backend** arbeitet unsichtbar im Hintergrund und besteht aus dem Teil, der Daten und Nutzeraktionen verarbeitet.

Die Begriffe werden gerne auch abseits der Websoftware angewendet. Allerdings führt das bisweilen zu merkwürdigen, eher verwirrenden Auswüchsen. Das bekannte Typo3 CMS nennt den Administrationsbereich ebenfalls Backend und den öffentlich sichtbaren Teil, Frontend. Ich halte das für historisch gewachsenen Unfug, da es die Begriffe verzerrt und Verwirrung stiftet.

### Frontend – Der visuelle Teil Ihrer Arbeit

Wenn Sie mit einer Software arbeiten, befinden Sie im Frontend-Bereich. Der beinhaltet drei Komponenten:

- Das sogenannte User-Interface (UI) mit seinen Formularen, Eingabemasken und Menüs
- Der Ansichtsbereich, mit Datenvisualisierung (Listen, Tabellen, Grafiken) und Bildern sowie Videos
- Das Design der Anwendung oder Webseite

Frontendentwicklung hat viel mit Grafik, Layout und Nutzererfahrung, der sogenannten User Experience (UX) zu tun. Frontend-Entwickler bestehen in der Regel aus Designer\*innen oder Grafiker\*innen mit Programmierkenntnissen. Sie sind dafür ausgebildet, Benutzeroberflächen möglichst effizient und benutzerfreundlich zu gestalten. Ihre Programmierkenntnisse beschränken sich deshalb in der Regel auf Frontendsprachen und deren Frameworks.

### Gute Design- und Auszeichnungssprachen für die Frontendentwicklung

Als typische Programmiersprache für Frontendentwicklung, nicht nur für Webseite, gilt JavaScript. Sie machte in den letzten 20 Jahre enorme Fortschritte hinsichtlich Standardisierung, Performance und Leistungsfähigkeit. Inzwischen existieren mächtige Bibliotheken sowie Frameworks wie jQuery, React, Angular, Vue usw. Sogenannte Frameworks sind vorprogrammierte Gerüste, welche Programmierern die Arbeit an komplexen Projekten erleichtern.

Kostenfreie Beratung sichern

deklarative QML. Die Oberfläche meines [SMIL Players](#) für Android basiert beispielsweise auf QML.

Für das Layout und die Datenvisualisierung in Webprojekten empfiehlt sich HTML in Kombination mit CSS. HTML zeichnet die Inhalte aus und CSS bringt sie optisch zur Geltung.

### HTML ist keine Programmiersprache

Werden Sie unbedingt hellhörig, wenn jemand von HTML-Programmierung spricht. So etwas geben nur Personen mit Halbwissen von sich. HTML ist eine Auszeichnungssprache, mit der Sie Inhalte kennzeichnen und besitzt keine Programmierbefehle.

## Backend – Die Datenzugriffsebene

Im Backend werden die vom Benutzer eingegebenen Daten verarbeitet, berechnet, gespeichert und mit anderen Anwendungen ausgetauscht. Typische Backendsoftware sind Datenbanken, Webserver oder sogenannte Microservices. Der Benutzer bekommt davon in der Regel nichts mit. Die Komponenten arbeiten im Hintergrund.

Die Aufgaben der Backend-Entwickler bestehen darin, die sogenannte Businesslogik zu programmieren und [Schnittstellen](#) zu definieren. Sie sind ausgebildete Softwareingenieure, beherrschen in der Regel mehrere Programmierkonzepte und kennen sich mit Datenbanken sowie mit Softwaredesign aus.

Grafik, Design oder Kenntnisse von Benutzerführung besitzen diese Programmierer eher weniger.

### Gute Backendsprachen

Für die Backendentwicklung sind folgende Sprachen üblich:

- Python
- Java
- PHP
- C#
- Ruby
- Perl
- SQL
- JavaScript

Anmerkungen:

Kostenfreie Beratung sichern

## SQL ist keine richtige Programmiersprache

SQL gilt als Abfragesprache und ebenfalls nicht als Programmiersprache. Mit SQL lassen sich keine beliebigen Programme schreiben, da sie nicht [Turing-vollständig](#) (Wikipedia) ↗ ist.

## JavaScript für Backends

Dank mächtiger Frameworks wie Node.js und Deno nutzen seit einigen Jahren viele Entwickler JavaScript auch als Backend-Sprache. Die Meinungen dazu sind unterschiedlich. Als Vorteil gilt, dass alle im Projekt die gleiche Sprache nutzen und Node.js im Netzwerk außerordentlich performant ist.

Ich halte allerdings nichts davon eine limitierte Frontendsprache mit einem notorisch instabilen Node.js in ein Backend zu pressen, nur um auf Gedeih und Verderb eine einheitliche Sprache zu nutzen. Gute Software setzt für jede Aufgabe die jeweils passende Programmiersprache ein.

Oft läuft es auch nur darauf hinaus, Kosten für die in der Regel etwas teureren Backendentwickler einzusparen.

## Was ist ein Full-Stack Entwickler?

Sogenannte Full-Stack Entwickler arbeiten sowohl in der Frontend- als auch in der Backendentwicklung. Sie beherrschen mehrere Programmiersprachen und dienen als Bindeglied, da sie Zusammenhänge besser verstehen.

## Frontend und Backend im Digital Signage



DIGITAL SIGNAGE  
— SOFTWARE —



EN | DE

ten. Dabei lässt sich klar erkennen, wie komplex das Thema wird, wenn wir den Bereich der Webanwendungen verlassen und die [Digital Signage Lösung](#) ganzheitlich betrachten.

Es gibt nämlich Software, die ausschließlich Frontendaufgaben erfüllt und welche, die nur als Backend fungiert.

## Frontend & Digital Signage

Kostenfreie Beratung sichern

Als typische Frontend-Komponenten können wir in unserer Branche die sogenannten **Digital Signage Widgets** bezeichnen. Die Daten kommen in der Regel von einem Microservice-Backend und das Widget zeigt die Informationen grafisch aufbereitet auf Ihren Bildschirmen.

Ein **Digital Signage Player** lässt sich auch als Frontend-Anwendung betrachten. Er holt sich die Inhalte von einem Webserver und zeigt sie auf dem Bildschirm. Der Betrachter sieht nur die Inhalte oder berührt einen Touchscreen, um bestimmte Informationen zu erhalten.

## Backend & Digital Signage

Als reine Backend-Software können wir einen **Digital Signage Proxy** bezeichnen. Der sorgt dafür, dass die Bildschirme die zur Verfügung stehende Internetleitung optimal ausnutzen und keine Inhalte doppelt herunterladen.

Das **Digital Signage CMS** ist als typische Webanwendung ist der Definition nach beides. Der Benutzer, der damit seine Playlisten erstellt, nutzt es als Frontend. Der Player oder der Proxy nutzen es als Backend. Wobei letzterer, wie gerade angemerkt, als Backend für den Player dient.

## Warum die Trennung in Frontend und Backend?

Um Projektentwicklung besser zu skalieren, unterteilt man Software in Front- und Backend. Während das Backend Softwareingenieure und Datenbankprogrammierer entwickeln, arbeiten am Frontend Designer und UX-Programmierer. So programmieren mehrere Spezialisten effizient und unabhängig voneinander an unterschiedlichen Bereichen. Kommunikation und Datenfluss erfolgen über festgelegte Schnittstellen.

Es gibt übrigens noch mehr Unterteilungen, um Software konzeptionell aufzuteilen. Unter anderem das sogenannte MVC-Pattern (Model-View-Controller), welches Websoftware in Daten-, Ansicht- und Kontrollkomponente unterteilt. Ich möchte das hier nicht weiter ausführen, weil es den Rahmen sprengen wird. Nur soviel:

Die Kunst in der Softwareentwicklung besteht darin, große Anforderungen in kleine Einzelanforderungen zu unterteilen und die nacheinander zu lösen. Dafür ist Organisation, Kommunikationsschnittstellen und geschickte Aufteilung der Aufgabengebiete notwendig.

Kostenfreie Beratung sichern

rer, der im Keller die Welt mit seinen Fingerspitzen kontrolliert und innerhalb von Sekunden in die abgeschottesten Systeme eindringt, existiert heutzutage eher als Hollywood-Klischee. **Programmieren ist Teamarbeit!**



**Autor: Niko Sagiadinos**

Entwickler & Co-Founder [SmilControl – Digital Signage](#)

Besuchen Sie mich auf: [LinkedIn](#), [Xing](#) oder [GitHub](#)

## Produktberatung und Kontakt

Sie haben weitere Fragen oder wünschen eine kostenfreie Beratung?

Setzen Sie sich einfach mit uns in Verbindung.

Ihr Name (Bitte ausfüllen)

Ihre E-Mail-Adresse (Bitte ausfüllen)

Ihre Telefonnummer (optional, falls wir Sie zurückrufen sollen)

Ihre Nachricht an uns

☐ Mit der Nutzung dieses Formulars erklären Sie sich mit der Speicherung und Verarbeitung Ihrer Daten durch SmilControl einverstanden.

Abschicken

## Unsere Kontaktdaten

Kostenfreie Beratung sichern

D-30419 Hannover

☎ +49 (0) 511 – 96 499 560

Amtsgericht Hannover

HRB 221981

USt-Id: DE 281 780 194

Vertretungsberechtigter Geschäftsführer:

Nikolaos Sagiadinos



Webkatalog

[Über uns](#) [Datenschutzerklärung](#) [Login CMS](#)

---

© 2024 SmilControl

Kostenfreie Beratung sichern