

2019

# Software & Scripts

LUKAS WEGMÜLLER  
INF2016G

## Inhalt

Software .....	2
Backend .....	2
Frontend .....	2
Browsererweiterungen (Plugins) .....	3
Scripts.....	3
Für unser Projekt verwendete Software & Scripts .....	3

## Software

Die Software ist nötig, um Webseiten dynamisch abzufüllen und für unser Auge schön zu gestalten. Natürlich kann man auch ein gutes Resultat erreichen, wenn man selber das Layout für die Seite schreibt, um aber eine dynamische Seite zu erhalten, ist die Anbindung an ein Backend unmöglich.

## Backend

Das Backend verwaltet die Abfragen des Benutzers und gibt ihm eine zur Abfrage passende Antwort. Es kann sich um eine Weiterleitung auf eine andere Seite, einen Fehler oder eine Seite mit dem aus dem Backend geholten Daten handeln. Werden die Daten für den Benutzer auch noch in späteren Situationen benötigt, speichert das Backend alles in eine Datenbank und steuert auch die Abfragen darüber.

## PHP

Mit etwa 80% ist PHP die meistverwendete Backend-Sprache für dynamische Webseiten. Da sie eher beschränkt ist, gibt es eine Unmenge an Librarys, welche man verwenden kann, um gewünschte Erweiterungen nicht komplett neu schreiben zu müssen.

Vorteile	Nachteile
Einfach erlernbar	Veraltete Programmiersprache
Weit verbreitet	Viele Sicherheitsaspekte, welche spezifisch beachtet werden müssen
Viele Tutorials im Internet vorhanden	

Es gibt zwar viele gute Frameworks und Librarys für PHP, jedoch eignet es sich doch eher für kleinere Projekte, da man es sehr einfach und schnell in das HTML einbinden kann. Für grössere Projekte ist es eher schwer, Argumente für PHP zu finden, welche die anderen Technologien in den Schatten stellen würde.

## ASP.NET

Nach PHP ist ASP.NET das am zweithäufigsten verwendete Backend-Framework. Als Sprache wird in der allermeisten Fällen C# oder Visual Basic verwendet.

Vorteile	Nachteile
Einfache Anbindung an MSSQL Datenbanken	Benötigt einen Windows-Webserver (IIS)
Moderne und effiziente Technologie	Nicht für kleine Projekte geeignet
Kompatibilität mit Linux und macOS kann mit .NET Core erreicht werden	

Sehr viele sogenannte «Nuget-Packages» können dem Projekt hinzugefügt und als Erweiterung genutzt werden. Eines der bekanntesten und praktischen ist das Entity Framework, welches die Anbindung, Erstellung, Migration usw. der Datenbank vornimmt und ebenfalls schon einige Sicherheitsrisiken der Datenbank eliminiert.

## Frontend

Unabhängig ob das Backend super läuft, will der Besucher der Webseite zuerst mal vom Layout und dem Design beeindruckt sein. Andernfalls wird er die Seite schnellstmöglich wieder verlassen. Da Firmen wie Google Unmengen an Zeit in wunderschöne Designs stecken, erwartet der Benutzer dies dann auch von kleineren Seiten und es ist praktisch unmöglich, dem nachzukommen.

## CSS Frameworks

Praktisch jeder Web-Entwickler kennt einige CSS-Frameworks, welche das Design der Webseite sehr stark vereinfachen. Weltweit am bekanntesten ist wohl Bootstrap. Das Ziel von solchen Frameworks ist es, nicht besonders unterschiedlich als andere auszusehen, aber eben doch das gewisse Etwas haben, was den Benutzern gefällt. Sehr viele Frameworks sind in den Grundzügen fast identisch, aber unterscheiden sich dann in kleineren Details sehr deutlich.

Vorteile	Nachteile
Vereinfach das Design der Seite	Teilweise sehr grosse Frameworks
Benutzer ist sich das Design von anderen Seiten gewohnt	

## Browsererweiterungen (Plugins)

Ein Browser kann heutzutage schon extrem viel und reicht in der Grundfunktionalität für einen grossen Teil der Benutzer bereits aus. Allerdings ist es trotzdem praktisch, wenn man diese mühsamen Werbungen auf Youtube (oder anderen Seiten) zu blockieren und einfach nur den Inhalt konsumieren kann, ohne gestört zu werden. Allerdings gibt es auch Designtechnische Erweiterungen wie einen «Dark-Mode» in welchen der Browser statt in weissen eher in schwarzgrauen Tönen erscheint und für gewisse Situationen das Auge eher schont, da es nicht zu hell auf dem Monitor ist.

Vorteile	Nachteile
Individuell einstellbar	Mögliches Sicherheitsrisiko
Viele verschiedene Plugins existieren	Erhöhte Auslastung des Browsers

## Scripts

Falls man nur eine kleine Funktionalität will, dafür aber nicht ein ganzes Backend-Framework aufsetzen will, oder es sogar nur etwas im Frontend ist, was man aber nicht mit CSS umsetzen kann, verwendet man Scripts. Am meisten wird dafür JavaScript / JQuery verwendet. Scripts können auch als Schnittstelle zwischen dem Frontend und dem Backend stehen. Ein bekanntes Beispiel dafür sind zum Beispiel «Ajax»-Requests, welche im script-Tag im HTML auf das Backend zugreifen und asynchron die Daten liefern und danach kann man mit diese in der View anzeigen.

Vorteile	Nachteile
Einfache Implementierung	Eingeschränkt in der Funktionalität
Asynchrone Abfrage von Daten möglich	

## Für unser Projekt verwendete Software & Scripts

Da unsere Webseite natürlich möglichst gut aussehen soll und wir so oder so nur Bootstrap verwenden dürfen, haben wir das auch entsprechend eingebunden. Ansonsten verwenden wir bisher keine anderen Erweiterungen und haben es voraussichtlich auch nicht geplant.