

Dinge aus der Richtung der Informationstechnologie

## Inhaltsverzeichnis

### Einleitung

- 1 Warum wieder HTML
- 2 Einrichtung einer HTML 5 und Javascript Entwicklungsumgebung unter ChromeOS
- 3 Warum ChromeOS nicht so gut ist?
- 4 Zu KI
- 5 Twitter (X) Tipps
- 6 Aktueller Nextcloud Client auf den Raspberry PI
- 7 Einrichtung von Dokuwiki unter CentOS 8.x
- 8 Gedanken zum smarten Haus und Anderes in dem Zusammenhang
- 9

# **Einleitung**

Eine Sammlung von Texten aus dem Bereich der Informationstechnologie zusammengetragen als Buch.

Inwieweit alles dem aktuellen Stand entspricht kann nicht so gesagt werden. Aber wir wissen, dass selbst ein Wort eine Erkenntnisbereicherung darstellen kann.

## Warum wieder HTML?

Vor einiger Zeit habe ich meine Webseite von Dokuwiki<sup>1</sup> auf Wordpress umgestellt. Beides ist zwar bequem in der Erstellung der Webseiten, aber damit verbunden sind ganz andere Dinge.

Wordpress<sup>2</sup> ist so aufgebläht das ich als Einzelperson gar nicht weiß, was da so vor sich geht. Zudem dann noch die Plug-ins aufgrund diverser Aspekte die PHP und MySQL so mit sich bringen, wo Haufen Dritthersteller auch noch rummischen, wo ich denke das sie auch nicht alles über Wordpress wissen und daher weiter Fehleranfälligkeit anfallen.

Zudem ist eine reine HTML-Webseite grundsätzlich ressourcenärmer als Wordpress. Es wird da nur ein Webserver benötigt. PHP, MySQL sind nicht notwendig. Außerdem ist die Fehleranfälligkeit was Sicherheitsaspekte oder Programmierfehler angeht geringer<sup>3</sup> und es ist länger verfügbar. Auch Cookiesgedanken schweben nicht zu primär rum.

Barrierefreiheit wird durch den Browser und Screenreader gewährleistet. Smartphone bzw. Tablets sind eh nicht so angesagt, diese Geräte sind hauptsächlich für den Konsum und die Kommunikation konzipiert. Außerdem werden die Anforderungen minimiert für die Archivierung und Wiederherstellung. Zudem gibt es Arbeitsschutz. Smartphone oder Tablet ist für die Büroarbeit nicht ausgelegt.

Außerdem bin ich eine private Person und da sind viele gesetzliche Ansprüche geringer, da hier geringe Ressourcen zu Realisierung vorhanden sind.

Das Ganze lässt sich auch über die Git-Repo<sup>4</sup> gut realisieren.

Stand: 21.02.2025.

---

<sup>1</sup> <https://www.dokuwiki.org/dokuwiki>, abgerufen am 21.02.2025.

<sup>2</sup> <https://wordpress.com/de/>, abgerufen am 21.02.2025.

<sup>3</sup> sie schreiben die Beschreibung der Seite selber und haben das so im Blick.

<sup>4</sup> <https://docs.codeberg.org/codeberg-pages/>, abgerufen am 21.02.2025.

## 2 Einrichtung einer HTML 5 und Javascript Entwicklungsumgebung unter Chrome OS

### **1 Vorwort**

Dieser kurze Artikel betrachtet die Einrichtung einer Entwicklungsumgebung für HTML 5 und Javascript unter dem Betriebssystem Chrome OS5.

Im Laufe der Covid-19 Pandemie Homeoffice Situation ist auch die Verbreitung von Chrome OS Geräten angestiegen<sup>6</sup>.

Für die Entwicklung von HTML 5 und Javascript Programmen sind im wenigstens drei Komponenten notwendig: Editor, Webbrower und ein Webserver. Auf diese drei Aspekte wird dieser Text sein Merkmal legen. Die genannten Anwendungen sind allesamt lokal (also offline) ausführbar. Zudem sind diese auch auf andere Systeme lauffähig, vorausgesetzt der Browser Chrome läuft darauf.

### **2 Webbrower**

Für die Entwicklung ist ein Javascript und HTML 5 fähiger Webbrower Voraussetzung.

Chrome OS besteht, wie der Name schon vermuten lässt, aus der Hauptanwendung Chrome<sup>7</sup>. Somit ist der Webbrower bereits vorhanden. Eine Installation ist nicht notwendig. Browserupdates werden zudem über das System selbst mit erledigt.

Nachteil wäre, dass die Entwicklung auf einen einzigen Browser erfolgt. Allerdings verwenden mittlerweile viele Browser die selbe Engine, so das dies kein übermäßiger Nachteil mehr darstellt.

### **3 Editor**

Um den Quellcode zu erstellen, ist ein Editor nützlich. Nachfolgend werden drei Editoren vorgestellt.

#### **3.1 Text**

Chrome OS liefert bereits einen Editor namens Text<sup>8</sup> mit. Dieser Editor ist recht

---

<sup>5</sup> [https://de.wikipedia.org/wiki/Google\\_Chrome\\_OS](https://de.wikipedia.org/wiki/Google_Chrome_OS), abgerufen am 20.12.2020.

<sup>6</sup> <https://www.googlewatchblog.de/2020/05/google-chrome-os-verkaufszahlen>, abgerufen am 20.12.2020.

<sup>7</sup> <https://www.google.de/intl/de/chrome/>, abgerufen am 20.12.2020.

<sup>8</sup> <https://chrome.google.com/webstore/detail/text/mmfbcljfglbokpmkimbfgdhdkjmjhdbgb?hl=en>, abgerufen am 20.12.2020.

minimal, bietet aber grundlegende Funktionen wie Syntax Syntax Highlighting<sup>9</sup>, Tabs, Zeilenummern und ein dunkles Design. Für kleine Projekte völlig ausreichend oder für Entwickler denen eine Minimalumgebung ausreicht.

## 3.2 Caret

Wer es etwas komplexer möchte, kann sich den Editor Caret<sup>10</sup> ansehen. Diese Anwendung ist nicht bei Chrome OS standardmäßig installiert, sondern muss über den Webstore<sup>11</sup> bezogen werden. Dies sollte aber keine große Hürde darstellen. Caret bietet sich für größere Projekte oder für Entwickler die mehr Möglichkeiten suchen an. Caret bietet neben den üblichen Dingen, z. B. eine Projektverwaltung und Makros.

Für Caret existiert ein auch Fork<sup>12</sup> mit Tern Support.

## 3.3 Code Pad Text Editor

Den Code Pad Text Editor<sup>13</sup> kann als Zwischending von Text und Caret in seinem Funktionsumfang sehen. Dieser Editor muss ebenfalls über dem Webstore installiert werden. Dieser Editor steht unter der GPL und der Code befindet sich auf GitHub<sup>14</sup>.

## 4 Webserver

Im Webstore ist auch ein Webserver für Chrome OS verfügbar. Der „Webserver for Chrome“<sup>15</sup> läuft lokal im puren Chrome OS ohne den Developer Mode<sup>16</sup>. Die möglichen Einstellungen sind recht übersichtlich, aber für den Zweck völlig ausreichend. Einfach den Ordner mit den HTML 5 bzw. Javascript Dateien hinzufügen, dann den Webserver startet und per `http://localhost:8887` im Chrome aufrufen.

## 5 Nachwort

Mit den hier gezeigten Chrome Anwendungen ist der Aufbau einer minimalen HTML 5 und Javascript Entwicklungsumgebung einfach möglich.

---

<sup>9</sup> <https://de.wikipedia.org/wiki/Syntaxhervorhebung>, abgerufen am 20.12.2020.

<sup>10</sup> <https://chrome.google.com/webstore/detail/caret/fjialecfjciodhpclledpamjacpmelml?hl=en>, abgerufen am 20.12.2020.

<sup>11</sup> <https://chrome.google.com/webstore/category/apps>, abgerufen am 20.12.2020.

<sup>12</sup> <https://chrome.google.com/webstore/detail/caret-t/agiednhnlghobdgpgfdnbafnnngmoij>, abgerufen am 20.12.2020.

<sup>13</sup> <https://chrome.google.com/webstore/detail/code-pad-text-editor/adaepfiocmagdimjec-pifghcgfjlfmkh?hl=en-GB>, abgerufen am 20.12.2020.

<sup>14</sup> <https://github.com/andrewbrg/codepad-chrome-app>, abgerufen am 20.12.2020.

<sup>15</sup> <https://chrome.google.com/webstore/detail/web-server-for-chrome/ofhbbkphbkjhfoeikjpcbhemloc-giib?hl=en-GB>, abrufen am 20.12.2020.

<sup>16</sup> [https://chromium.googlesource.com/chromiumos/docs/+/master/developer\\_mode.md](https://chromium.googlesource.com/chromiumos/docs/+/master/developer_mode.md), abgerufen am 20.12.2020.

Mit der Linuxumgebung<sup>17</sup> unter Chrome OS kann dieses Vorhaben auch mit Linuxanwendungen realisiert werden. Auf dies wurde aber bewusst verzichtet, um zu zeigen, dass das Ganze auch mit reinen Chrome OS Programmen durchführbar ist.

Andere Informationen bzgl. Softwareentwicklung unter Chrome OS ist unter <https://chromeos.dev/en> zu finden.

Stand: 20.12.2020.

---

<sup>17</sup> <https://chromeos.dev/en/linux>, abgerufen am 20.12.2020.

# Warum ChromeOS nicht so gut ist

<https://www.n-tv.de/technik/Acer-Chromebook-Plus-515-eine-flotte-Windows-Alternative-article24578893.html>, abgerufen am 05.12.2023

Es ist keine Alternative zu Windows. Es ist noch schlechter nutzbar als Windows. Sie sind absolut netzabhängig, außer bei Mail (Offlinevorhaltung bis zu 90 Tage maximal) und paar gecachten Dateien, die sie vorher mühselig ausgewählt haben müssen. Kalender fast nicht offline nutzbar auf diese Weise. Ansonsten müssen sie per Android-Anwendungen und Linux aufrüsten [zusätzliche Ressourcen (Festspeicher, Einrichtung, Verknüpfungen, Arbeitsspeicher der virtuellen Maschine zusätzlich + Linuxsystem, Androidemulation belegt zusätzlich) → Geldmittel], da können sie auch gleich ein Tablet (Konsum) oder Linux nutzen. Also eine Infrastruktur, die so völlig verwuschelt ist, dass sich der Sinn nicht mehr so erschließt. Also auf der Schiene dann eher gleich Android und mehr die Multitaskingbarrieren weg.

ChromeOS ist maximal für Webentwickler und nebenläufige Berufe interessant. Google ist eine Internetfirma, also da fast nur www<sup>18</sup>. Dann wissen sie auch, wie ChromeOS ist (Denk- bzw. Lebensweise). Zu mehr ist es nicht nützlich. Aber auch hier gabs es auf grober Übersicht nicht mal ein gutes Grafikprogramm (Autor im Bereich Spieleentwicklung aktiv), um Spieleentwicklung auf dieser Basis durchzuführen.

Zudem hat Google die Angewohnheit ihre Apis „ständig“ nach Best Practice<sup>19</sup> zu ändern. Entweder werden Standards nie so eingehalten das nachgelegt werden muss. Oder Standards werden ständig anders interpretiert, da Standards eigentlich allgemeingültig dauerhafter irgendwie festgelegt sind.

Selbst lokales Backup (gesteuerte Datensicherung, nicht mühselig alles per Hand) geht kaum. Bei 64GB Festspeicher können sie schon mal die Hälfte wegdenken, da vom System belegt. Kaum Konvertierungen von Dokumenten aus den Google Anwendungen möglich (Notizen z. B. basieren nur auf Json, Konvertierungen<sup>20</sup> notwendig). Also für Unternehmen überhaupt oder Naturwissenschaftler nicht brauchbar. Höchstens minimal Tabellenkalkulation und Texte. Also Kleinstnischen.

Die im Artikel aufgezeigte Hardware ist für das System auch völlig überdimensioniert. Kaufen sie lieber was Robustes<sup>21</sup>. Spieldownload (ist auch nur ein Videosstream + zusätzlich wird nur ihre Eingabe mitgeschickt) geht auch mit weniger.

---

<sup>18</sup> <https://wirtschaftslexikon.gabler.de/definition/world-wide-web-www-49260>, abgerufen am 26.12.2024.

<sup>19</sup> <https://www.bwl-lexikon.de/wiki/best-practice>, abgerufen am 26.12.2024.

<sup>20</sup> <https://github.com/thethales/GoogleTasksJSONToTXT>, abgerufen am 26.12.2024.

<sup>21</sup> Lebenszyklen sind im IT Bereich üblicherweise 10 Jahre, im Server Bereich sogar noch höher.

Dies ging mit Chromecast sogar (Stadia<sup>22</sup> war der beste Anbieter in diesem Bereich). Zudem hat sich immer wieder gezeigt, dass die Kosten (Ressourcen, nicht nur in Geld denken) am Ende höher sind, als die auftretenden Unzulänglichkeiten [Zeitbegrenzung, Ausfall des Netzes, Latenzen (E-Sport) Stromkosten (sind bei 120W-Systemen bei ca. 5 € die Woche<sup>23</sup>), also Eingriffe irgendwie in ihre Freiheit]. Google Kosten zusätzlich, Abos etc. pp.

Weiterer Punkt, der eher negativ ist. Nach der Supportlaufzeit können sie faktisch mit dem Gerät nichts mehr machen<sup>24</sup>. Nur als Experte mit Glück (anderes BIOS, UEFI, dann Linux, Tastaturprobleme, dann wieder Lösung über externe Tastatur, also zusätzliche Ressourcen. Um das BIOS zu ersetzen, müssen sie auch noch an die Hardware ran, weil ein Löschschütz davor liegt. Wenn das Gerät zu dünn ist, wird es noch schwieriger. Rückspielen des vorher gesicherten BIOS verlief eher negativ auf dem Testgerät. Nun positiv: bei meinen vorhandenen Geräten wurde jetzt der Support bis 2030 laut interner Systemanzeige sozusagen verlängert (vorher 2027?).

Das Ganze wurde persönlich mit ChromeBox 3 in zwei Konfigurationen und einigen Chromebooks über ein Jahr lang untersucht.

Stand: 26.12.2024.

---

<sup>22</sup> <https://stadia.google.com/gg>, abgerufen am 05.12.2023.

<sup>23</sup> <https://rechneronline.de/steckdose/stromkosten.php>, abgerufen am 26.12.2024, berechnet mit 18h und 120W (volle Leistung, meist geringer), <https://www.nvidia.com/de-de/geforce-now/memberships>, abgerufen am 26.12.2024, Zeitbegrenzung (es ist ihre Lebenszeit).

<sup>24</sup> Also vorsicht beim Gebrauchtkauf. Gucken sie vorher ob der Support noch existiert.

