Glossar

|  |  |  |
| --- | --- | --- |
| Nr | Name | Beschreibung |
| 1 | **Informatisch Begriffe** | |
| 1.1 | Html | Html oder lang HyperText Markup Language, ist dazu gemacht Inhalte auf Webseiten anzuzeigen. Ein Html Projekt kann man erstellen, in dem man ein File mit der Endung Html kreiert. Dieses File kann man danach in einem Browser öffnen. In Html wird alles in sogenannten Tags Hineingeschriebenen. |
| 1.2 | Tag | Ein Tag ist ein Schlüsselwort welches in kleiner/grösser als Zeichen geschrieben wird. Dies kann danach ein Browser lesen und verstehen. Ein Beispiel ist: <h1> Hallo </h1>, was zu einem Titel wird, in welchem «Hallo» steht. (vgl Hayes 2020) |
| 1.3 | Css | Css oder lang Cascading Style Sheets wird dafür benutzt, um die Inhalte von HTML schön zu gestalten. Dies kann auf viele Weise getan werden. Zum Beispiel könnte man dem «h1» tag eine Hintergrundfarbe geben. |
| 1.4 | JavaScript | JavaScript ist neben Html und Css die dritte Grundkomponente für das Web. JavaScript ist eine Programmiersprach, welche für die Logik in einer Webseite zuständig ist. 90% der Webseiten wären ohne JavaScript statisch und mit einem normalen PDF vergleichbar. (vgl Megida 2021). Für JavaScript gibt es sehr viele Libraries und Frameworks, welche die Grundfunktionen von JavaScript erweitern. |
| 1.5 | Canvas | Canvas ist ein tag, welches man in Html definieren kann. Dieser Tag kann dazu verwendet werden Figuren zu zeichnen. Um Diese Figuren aber wirklich zeichnen zu können, brauch es JavaScript. |
| 1.6 | Library | Eine Library oder auch Bibliothek sind einfach gesagt eine Ansammlung von Funktionen. Sie helfen dabei Probleme einfacher zu lösen. Man kann in einem Programm frei entscheiden, wo und wann man die Funktionen einer Library benutzten kann. Eine Library kann man sich also wie eine Büchersammlung ansehen. Einige Titel dieser Bücher kann man benutzten, man ist jedoch nicht gezwungen. (vgl Wozniewicz 2019) |
| 1.7 | Framework | Ein Framework ist wie auch schon eine Library eine Ansammlung von Funktionen, mit dem Unterschied, dass man bei einem Framework nicht entscheiden kann, wo und wann man das Framework benutzt, da das ganze Programm in diesem Framework sein muss. Wie der Name dies bereits erzählt, ist es ein Rahmen, in welchem man Arbeiten kann. (vgl Wozniewicz 2019) |
| 1.8 | API | API steht für Application Programming interface und ist dafür gedacht, dass zwei Applikationen miteinander reden können. |
| 1.9 | Funktionale Programmierung | Eine Funktionale Programmierung besteht hauptsächlich aus einfachen Funktionen. Der Code ist einfach zu lesen und zu testen. Darum beginnen viele Programmierer mit der Funktionalen Programmierung. |
| 1.10 | Objekt Orientiere Programmierung | Objekt Orientiere Programmierung ist ein Klassenbasierte Programmierung. Alles besteht aus Objekten, welche man übergibt und verarbeitet. Die Daten in den Objekten sind gekapselt was heisst, dass sie nicht wissen, was in der «Aussenwelt» also im Programm sonst passiert. Damit ist Zugriff sehr kontrolliert. |
| 1.11 | Klasse | Eine Klasse kann man sich als einen Bauplan vorstellen. Als Beispiel: Wenn man ein Haus bauen möchte, hätte man die Klasse Haus, was die Bauanleitung des Hauses widerspiegelt. In dieser Bauanleitungen ist dann z.B. beschrieben, wie man eine Türe einbaut. Solche Instruktionen, werden als Methoden abgebildet als hätte man somit in der Klasse Haus die Methode Türe\_einbauen. |
| 1.12 | Abstrakte Klasse | Eine Abstrakte klasse kann man nicht wie eine normale Klasse instanziieren. Diese Klasse ist dafür gedacht, um anderen Klassen vorgaben zu geben. In diesem Beispiel könnte eine Abstrakte klasse Gebäude heissen. Jedes Gebäude braucht eine Türe und darum auch eine Methode, welche die Türe implementiert. |
| 1.13 | Funktion | Eine Funktion ist ein Stück code, welcher benannt wurde. Diese Funktionen kann man einfach mit dem Namen aufrufen. Eine Funktion ist dazu da, das Programm in Blöcke zu strukturieren. Ausserdem kann man so Redundanzen verhindern. |
| 1.14 | Methode | Eine Methode ist sehr ähnlich wie eine Funktion. Der Unterschied ist, dass eine Methode immer ein Objekt braucht, um aufgerufen zu werden. In unserem Beispiel kann man keine Tür einbauen, ohne ein Haus zu haben. Also muss die Methode Türe\_einbauen auf ein Haus aufgerufen werden. Anderst wie in diesem Beispiel, sind Methoden meistens auf Englisch und werden darum kein geschrieben. Eine korrekte Namensgebung wäre also: «implementDoor» |
| 1.15 | Instanziieren | Sobald man dieses Haus jetzt bauen wollte, könne man diese Klasse instanziieren. Dadurch bekommt man ein Objekt mit dem Typ Haus und schon hat man ein Haus. Um Eine Tür einzubauen, könnte man jetzt einfach die Methode Türe\_einbauen aufrufen. |
| 1.16 | Deploy | Um ein Produkt online zu stellen, muss man sie Deployen. Auf Deutsch heisst Deploy einfach einsetzen oder aufstellen. |
| 2 | **Three.js Begriffe** | |
| 2.1 | Scene | Eine Scene in wie eine Szene in einem Film. Nur was in der Szene ist, kann gerendert und gesehen werden. Jedes Objekt, welches man sehen will, muss in die scene kommen. Das gilt auch für jedes Licht und die Kamera. |
| 2.2 | Rendering | Rendering zu Deutsch Wiedergabe oder Übertragung, ist ein Prozess um Bilder oder Modelle zu einem realistischem Bild zu übertragen. Dieses Geschehen passiert bei Three.js, viele Male in der Sekunde. |
| 2.5 | Shaders | Ein Shader ist ein kleines Programm, welche die Scenes während dem renderen manipuliert.  «Sie werden verwendet, um jeden Scheitelpunkt einer Geometrie zu positionieren und jeder sichtbare Pixel dieser Geometrie einzufärben.» (Simon 2021d). |
| 2.6 | Orbit Controls | Orbit Controls ist eine Library, die man verwenden kann, um seine Kamera zu Kontrollieren. Damit kann man mit der Maus die Kamera bewegen. |
| 2.7 | Draco Loader | Mit einem Draco loader, kann man files laden, welche im Draco Format sind. Dieses Format bekommen man, wenn man eine Draco compression macht. Dabei werden die Dateien wie z.B von Blender ein wenig kleiner. |
| 3 | **Generelle Begriffe** | |
| 3.1 | Browser | Ein Browser ist ein Tool wie Chrome oder Firefox, mit welchen man das Internet durchsuchen kann. «Ein Browser - das ist Englisch und bedeutet so viel wie "Stöberer" - ist ein Programm, mit dem Websites korrekt angezeigt werden.» (Predan-Hallabrin 2019) |
| 3.3 | Frame | Ein Frame ist einfach gesagt einfach ein Bild. Die berühmte Bezeichnung FPS (Frames per Second) kommt davon, wie viele Bilder oder Frames in einer Sekunde gezeigt werden. Wenn diese Frame rate genug hoch ist, sieht eine Person keine einzelne Bilder mehr sondern ein Video. |
| 3.4 | Third Person | Thrid Person ist eine Sicht, welche ist, als würde eine andere Person dem gespielten Charakter von einer Distanz folgen und ihn dabei Filmen. Das Pendant dazu wäre First Person. |

# Quellenverzeichnis

Hayes Adam (13. November 2020) HyperText Markup Language URL: <https://www.investopedia.com/terms/h/html.asp> (besucht am 24.11.2021)

Dillion Megida (29. March 2021) What is JavaScript? A Definition of the JS Programming Language URL: <https://www.freecodecamp.org/news/what-is-javascript-definition-of-js/> (besucht am 24.11.2021)

Wozniewicz Brandon (1. Februar 2019) The Difference Between a Framework and a Library URL: <https://www.freecodecamp.org/news/the-difference-between-a-framework-and-a-library-bd133054023f/> (besucht am 24.11.2021)

Domainssaubillig (2021) Objektorientierte vs. funktionale Programmierung im Web URL: <https://www.domainssaubillig.de/blog/blog-artikel/items/objektorientierte-vs-funktionale-programmierung-im-web.html> (zuletzt besuch 24.11.2021)

Simon Bruno (2021c) <https://threejs-journey.com/lessons/27> (zuletzt besuch 24.11.2021)

Predan-Hallabrin Beatrice (31.08.2019) Was ist ein Browser? Einfach erklkärt. URL: <https://praxistipps.chip.de/was-ist-ein-browser-einfach-erklaert_41369>