

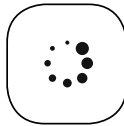
Wahrnehmung und Konsum von Information beeinflusst von Technologien des Internets

Muss eine Maschine, ein System, beim Zuliefern von Information wirklich so viel mehr „Intelligenz“ und Wissen nutzen, als der Mensch, der beim Konsum am Screen scheinbar einen Großteil seines Verstandes auflöst?

www.reallysimplesatisfaction.com



Wahrnehmung und Konsum von Information beeinflusst von Technologien des Internets



Stefan Endres

Kunsthochschule Kassel

2019



- 1. Einführung**
- 2. Technologie**
 - 2.1. RSS, Feeds, APIs & AGBs**
 - 2.2. Eine Seite im Netz**
 - 2.3. Eine Synthese im Browser**
 - 2.4. Scraping, SOP, CORS & Proxy**
- 3. Konsum & Wahrnehmung**
 - 3.1. Informationsaufnahme & Interaktion am Bildschirm**
 - 3.2. Anwendung & Intention**



1. Einführung

Wie wahrscheinlich sehr vielen meiner Generation, als Kind der 90er Jahre, wurde der Zugang zum Internet mir wortwörtlich in die Wiege gelegt. Kaum jemand in meinem Umfeld hatte damals ein Bild davon, was das Internet denn nun ist, was es seinen Nutzern gibt, was es von ihnen verlangt – und besonders was es in Zukunft sein wird und kann. So konnte ich recht uneingeschränkt, ohne Aufsicht und frei Seite für Seite besuchen und Stück für Stück das damalige (noch-) Web 1.0 erfahren.

Seine starre Kataloghaftigkeit, die rohe Optik und Bedienbarkeit wurde für mich schon bald durch Systeme wie eBay und Napster auf eine neue Ebene (Web 2.0) gehoben. Es war nun möglich erste eigene Inhalte ohne Programmierkenntnisse ins Netz zu bringen, mit anderen Menschen in Kontakt zu treten, Austausch zu betreiben, aber vor allem an Informationen und Dinge zu kommen, an die man bisher nicht so leicht, oder nicht in diesem Umfang kam. Zudem war das Internet zu dieser Zeit noch eine rechtliche Grauzone (noch mehr als heute...). Als Jugendlicher besserte ich so mein Taschengeld mit Online-Auktionen auf, wechselte mehrmals im Jahr mein Handy-Modell, blockierte die Festplatte des Familien-Rechners mit „Massen“ an Musik-Dateien – und das ist nur ein Auszug.

Als einer meiner Schulfreunde ein eigenes Forum für unseren Freundeskreis ins Leben rief, wurden mir besonders zwei Aspekte des Online-Lebens bewusst: Exklusivität und Anonymität. Nicht jeder in meinen Kreisen



bekam die Erlaubnis dort einen Account anzulegen und beizutreten (manch einer hatte noch nicht einmal Zugang zum Internet), wieder vereinzelte Mitglieder erhielten Administratoren-Rollen (konnten so sämtliche Beiträge ändern oder entfernen und auf einen exklusiven Bereich zugreifen). Bei Missbrauch dieser wurde die Befugnis vom Haupt-Administrator wieder aberkannt, man wurde degradiert und ausgeschlossen. Wenn man wusste wie, konnte man sich einen weiteren Account anlegen, unter einer neuen Identität Namen und mit irreführendem Avatar. Man konnte Dinge von sich geben, wie man es im „echten Leben“ nie tun konnte/ wollte, fremde Rollen einnehmen, Gerüchte verbreiten ohne Einfluss auf die eigene Person. Nach Shitstorms, Mobbing-Vorhaben, jugendlichem Schwachsinn und Starrsinn, war letztendlich jeder in der Rolle eines Administrators und das Forum starb nach und nach aus.

Zum Teil lag das sicher auch am Aufkommen, bzw. in-Reichweite-Kommen von Social-Media-Plattformen wie SchülerVZ und StudiVZ. Diese beschreiben für mich den Übergang von 15-Pubertierende-in-einem-Netzwerk (mit kleiner Exkursion über MySpace) zum großen Vorbild Facebook. Das schon in recht frühen Stadium enorme Ausmaß an dort einsehbaren persönlichen Daten, deren Verknüpfung spürbarer denn je ins Offline-Leben reicht, sowie die Vorstellung der Möglichkeiten in der (automatischen) Verarbeitung dieser Daten (damals noch-zukünftiges-Web 3.0), führte schon nach wenigen Monaten bei Facebook (und nun vor fast 10 Jahren) zu meinem Austritt. Bis heute habe ich auf keiner Social-Media-Plattform einen Beitrag geteilt.

Schon zu Beginn des langersehnten Studiums an einer Kunsthochschule entwickelte sich mein Fokus vom analogen-statischen zum digitalen-dynamischen Produzieren und durch weniger als eine handvoll von Studenten, erlangte ich meine ersten Grundlagen des Programmierens. Der graphische Output spielte für mich schon immer eine größere Rolle als die Beständigkeit oder Nachvollziehbarkeit des Codes. Von generativen Processing-



Experimenten mit Ton, Bild und Physik kam ich über kleine Eingriffe ins Netz, Auseinandersetzungen mit Hacking und Netz-Kunst zum Web-Design und -Development (wahrscheinlich auch motiviert durch die lukrativ anmutende Nebenwirkung dieses Feldes). Mir fiel es deutlich leichter bei wirklichen Aufträgen mein Handwerk zu lernen, es ergaben sich nach und nach kleine Jobs und ich gestaltete und realisierte erste Portfolios, Ankündigungs-Webseiten, gefolgt von Archiven und Online-Shops – allein und mit anderen. Nach Jahren von Abstinenz begann ich wieder dem Internet beizutragen, allerdings nicht als User.

Eine digitale Infrastruktur aufzubauen und instandzuhalten, diese mit einer Ästhetik anzureichern und aus dem Produkt eine Erfahrung (für die Besucher sowie die Betreiber) entstehen zu lassen, war nach Abschluss und Veröffentlichung, unabhängig von der Branche, der Komplexität und dem Publikum immer sehr stark auf ihren Inhalt und dessen Verwaltung angewiesen. Für meine Position bedeutet das von einer gewichtigen Verantwortung gelöst zu sein, die Autorschaft abzugeben. Dies kombiniert mit einer Faszination und Neugierde für das Zusammenspiel von diversen fluiden und größtenteils unsichtbaren Technologien mit den (unter anderem daraus resultierenden) Konsumgewohnheiten und Verhaltensweisen der User, brachte in mir die Motivation herauf das Projekt Really Simple Satisfaction zu kreieren und in Frage zu stellen.



2. Technologie

Really Simple Satisfaction bedient sich als Website automatisiert an einem externen Dienst (RSS-Feed), der strukturierte Daten von tagesaktuellen, globalen Such-Trends des Internets bereitstellt. Anhand dieser Daten wird Inhalt gesammelt, aufbereitet, kanalisiert und zum Zeitpunkt des Aufrufes in eine interaktive, graphische Hülle gespeist, welche dem User letztendlich ausgeliefert und von ihm (im besten Fall) konsumiert wird.

Im Folgenden werden die wichtigsten technischen Bestandteile aufgefächert, die hervorgetretenen Hürden und daraus resultierenden Strategien erläutert, sowie die Auswirkungen auf das Endprodukt reflektiert. Anhand einzelner Abhandlungen sollen die Dinge sichtbar gemacht werden, die sonst verborgen hinter einer leuchtenden Oberfläche, aufkommen, bestehen und wieder verschwinden.



2.1. RSS, Feeds, APIs & AGBs

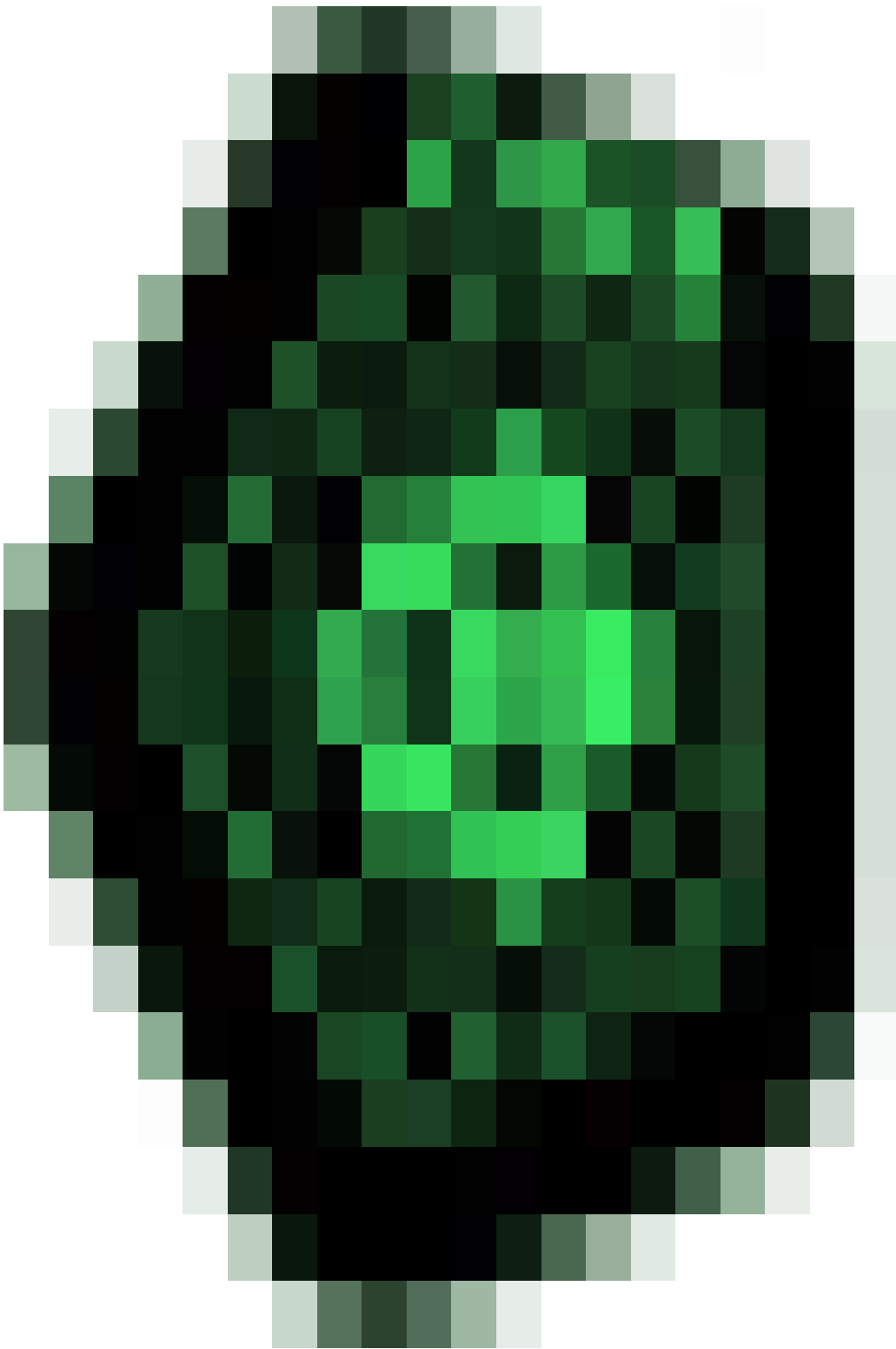
RSS (früher „Rich Site Summary“, aktuell geläufig unter dessen Backronym „Really Simple Syndication“) ist ein Dateiformat für Web-Feeds, das schon seit dem Jahr 1999 besteht. Websites die RSS zur Verfügung stellen, halten ihre Abonnenten über eine RSS-Datei so immer auf dem Laufenden, wenn sich Inhalte ändern oder neue hinzugefügt werden. Die Daten werden in einer standardisierten Formatierung übermittelt (Vorgang der Aggregation), als artikel-ähnliche Informationsblöcke und bestehen aus Überschrift, Textauszug, Schlagwörtern, sowie der Website-Adresse als Quelle, allerdings ohne Elemente für Layout oder Navigation – sie sind also in erster Linie für das Lesen durch Maschinen vorgesehen. Der User benötigt ein sogenanntes Aggregator-Programm, einen Feed-Reader, der auf die Adresse der RSS-Datei des abonnierten Feeds zeigt, dadurch automatisch Aktualisierungen laden kann und den Inhalt graphisch darstellt (Vorgang der Syndikation). Feed-Reader gibt es als native Applikationen, Web-Apps oder Erweiterungen für den Browser (Feedly, The Old Reader oder Inoreader als populäre, aktive – Google Reader und Digg Reader als bereits stillgelegte Exemplare), auch manche E-Mail-Programme besitzen RSS-Lesefunktionen. Eine weitere Art von Aggregator-Programmen sind RSS-Parser, die als Teil von eigenständigen Websites, den bezogenen Inhalt aufbereiten und deren Oberfläche bereitstellen.

Als User habe ich RSS immer nur, wenn überhaupt, am Rande wahrgenommen und zwar meistens rechts oben auf verschiedenen Websites, als kleines Icon – welches dem Symbol für W-LAN sehr ähnlich sieht. Mehr Aufmerksamkeit konnte es aber nicht auf sich ziehen, womöglich weil seine Hochphase schnell von Social-Media-Feeds eingedämmt wurde und RSS-Feeds von den Bildflächen vieler verschwanden.



Unter anderem im angewandten Kontext des Design-Büros, mit dem ich zusammenarbeite, kam es in den letzten ein bis zwei Jahren häufiger zu der (An-)Frage, ob man denn auch Inhalt des eigenen Instagram-Profiles auf seine Website einbinden kann, ohne die vom werbefinanzierten Online-Dienst mitgelieferte Optik, natürlich automatisiert. Instagram bietet dafür eine Programmierschnittstelle (Instagram API Platform) an, die dies ermöglicht und die Website so deutlich bereichert, ohne zusätzlichen Aufwand in der Inhaltsverwaltung durch den Betreiber. Der Haken daran: Seit 2018 wird die Schnittstelle immer weiter abgebaut und durch den letzten Schritt, Anfang 2020, wird es nicht mehr erlaubt sein vom eigenen Benutzer-Profil Informationen oder Medien zu beziehen. (Ersetzt wird diese Schnittstelle heute schon teilweise mit der Instagram Graph API, die sich allerdings auf große und mittelgroße Business-Konten beschränkt. Ob es mit dieser genauso möglich sein wird, Daten extern zu verarbeiten und nach aussen, an die Öffentlichkeit zu leiten ist im Moment noch unklar.) Zu den angegebenen Gründen des Datenschutzes, erscheint es mir auch als logische Schlussfolgerung, dass der Konzern es lieber hat, wenn die Nutzer Instagram-Inhalte ausschließlich auf der Instagram-Plattform konsumieren, wo es ständig nachvollziehbar ist, wie sie sich verhalten, was sie von sich und ihrem Umfeld preisgeben und ihnen jederzeit Werbung in den Weg gelegt werden kann.

Es wirkt so, als würde dieses Vorgehen der Online-Ökonomie die gänzliche Abschirmung von der digitalen Außenwelt bezwecken. Instagrams Beiträge können nur innerhalb der App geteilt und weitergeleitet werden. Besucht man auf Instagram einen Link zu einer Website, öffnet sich diese in einem hauseigenen Browser. Auch in die andere Richtung fließt nichts: Es gibt bisher noch keine offene Web-Anwendung um extern publizierte Inhalte auf Instagram zu leiten (wie es z. B. der Tweet-Button von Twitter oder Facebooks „Teilen“-Button möglich macht) und keines der Bilder von Instagram ist im Moment von einer Suchmaschine (direkt) indiziert worden. Man wendet sich



also ab, vom eigentlichen Konzept des Internets, wo alles miteinander vernetzt ist. Das Gewicht der Auswirkungen dieser Separation ist erst auf den zweiten Bild zu erahnen. Das früher (im Vergleich zum DIN-Format, 16:9/ HD, Panavision/ Kino, dem Format sämtlicher Geldscheine, etc.) eher unbeachtete 1:1-/ Square-Format ist einer der Grundsteine des Erscheinungsbildes von Instagram und erweckt mittlerweile umgehend und unangefochten eine unterbewusste Verknüpfung mit dem Konzern. So schlägt einem heute wahrscheinlich jede Smartphone-Kamera-App dieses Format als eines der ersten Auswahlmöglichkeiten vor. Durch ständig erneuerte Nutzungsbedingungen und AGBs (und viele User schenken diesen absolut keine Beachtung) wird ein Raum außerhalb von nationalem oder internationalem Recht geschaffen. Man bewegt sich dort, latent getrieben von Narzissmus, FOMO („Fear of missing out“, die Angst etwas zu verpassen) oder einfach um Zeit totzuschlagen und ist nie komplett im Bilde, was mit einem geschieht, oder geschehen kann. Die Ungewissheit, ob das eigene Handeln auch genau das Handeln ist wofür man es hält, oder ob man parallel noch ein paar andere Fäden zieht, wird in diesem Raum schon im Keim erstickt.

Von den Verschwörungstheorien abgesehen, war der Effekt, welcher eine Art von Aktualität und Lebendigkeit auf manche mager bestückte Websites brachte, enorm. Für eine Ankündigungs-Website einer Ausstellung meiner Kunsthochschul-Klasse über Erregung von Aufmerksamkeit und Informationsaufnahme am Bildschirm, wollte ich genau diesen Effekt um die eigentlich relevanten Auskünfte stützen. Nach kurzer Recherche erwies sich das schon beinahe antik erscheinende RSS als mehr als nur eine Alternative zu Social-Media: Keine Anmeldung oder Accounts waren erforderlich, RSS-Dateien sind frei zugänglich, Daten fließen von dort ausschließlich in Richtung des RSS-Readers (in meinem Fall ein RSS-Parser) und nicht woanders hin – die Privatsphäre des Abonnenten bleibt also unangetastet. Folglich war das Ergebnis nur von der Auswahl der Feeds und deren Inhalt, hinsichtlich Qualität,



Quantität und Aktualisierungsfrequenz, abhängig. Durch meine erste wirkliche Auseinandersetzung mit RSS entstand eine experimentelle Website, deren Hintergrund mit Videos des Tages von Wikimedia bespielt wird, darauf die aktuellen Schlagzeilen von BBC-News laufen, überlagert von zufälligen Bildern eines Tumblr-Blogs (und wo, wann und mit wem die Ausstellung stattgefunden hat, konnte man auch noch einsehen).

Die Flut an Daten, die Möglichkeiten und Leistungskraft in der Verarbeitung durch eine Website und daher auch der befreite gestalterische Umgang der sich mir bot, stellten für mich ein Abbild dessen dar, worauf die heutzutage scheinbar höchsten Güter des Internet beruhen mögen. So sollte dieses Projekt nur ein erster Schritt in Richtung einer Arbeitsweise sein, die Really Simple Satisfaction für mich vorstellbar und realisierbar gemacht hat.

2.2. Eine Seite im Netz

Die W3C („World Wide Web Consortium“, Gründer und Vorsitzender ist Tim Berners-Lee, der als Erfinder des World Wide Web gilt) bildet als internationale Mitgliederorganisation das Gremium zur Standardisierung, Spezifikationen, Ausrichtung und Entwicklung der Technologie im World Wide Web. Ihre 451 Mitglieder (am 14.02.2019) setzen sich großteils aus Universitäten und Bildungseinrichtungen, Banken, Elektronik-Hersteller, allen üblichen Online- und Technologie-Giganten oder einzelnen Regierungsapparaten zusammen. Sie finanziert sich durch Beiträge dieser Mitglieder und kann so Technologien wie HTML, XML, CSS, SVG und PNG standardisieren.



Ihr Ziel ist es „das Netz zu seiner vollen Entfaltung zu führen“, hinsichtlich z. B. dessen Universalität, Dezentralisierung oder dem Aufbauen von Vertrauen in die Interaktion mit ihm.

Der Prozess der Realisierung einer Website, ist stark vom status quo der Technologien des Internets abhängig. Konzepte und Möglichkeiten scheinen schier unendlich und ändern sich jeden Augenblick – schreibt man eine Zeile Code, ist dieser quasi schon veraltet. Es entsteht ein lebendiges Geflecht, verschiedenartiger Programmiersprachen, Abwandlungen von Programmiersprachen, die miteinander kommunizieren, wiederum selbst Code erzeugen können, auf Daten zugreifen und diese für die Umgebung und Wahrnehmung der User aufbereiten. Ladezeiten, Dateigrößen, Gestaltungs-Konzepte, Frameworks, Zugriffsrechte und Datenbanken verschmelzen zu einer gigantischen Infrastruktur, die für den Menschen aber nur soweit sichtbar ist wie die Spitze eines Eisbergs in Zeiten der Klimaerwärmung.

In diesem Unterkapitel möchte ich den Begriff der Website auf das wesentliche herunterbrechen und so greifbar wie möglich zu machen, sowie die für Really Simple Satisfaction (und viele andere von mir entwickelte Websites) verwendeten Programmiersprachen, deren Funktionsweisen sowie -orte und Verknüpfungen vorstellen. Es soll als keinerlei technische Anleitung, Lösung oder Wertung verstanden werden, sondern nur die Umgebung darstellen, in der ich mich, in meiner Position bewege.



Eine Website ist ein Verzeichnis (ein Ordner), das mindestens ein von einem Browser (dem Client) interpretierbares Textdokument enthält. Es kann weitere Verzeichnisse (oder Verzeichnisse in Verzeichnissen,...), Dokumente und Dateien aller Art (Bild, Ton, Video, Anwendungen, etc.) beinhalten, liegt auf einem Rechner (dem Server oder Host), der mit einem Netzwerk (dem Internet) verbunden ist und durch eine URL („Uniform Resource Locator“, Domain, Pfad, oder Adresse) die Website lokalisierbar macht.

Eine Website wird im Allgemeinen in Frontend und Backend aufgeteilt. Das Frontend ist näher am Benutzer und das Backend näher am System. Das Backend ist der Teil der Website, der durch deren Aufruf auf dem Server ausgeführt wird. Dort werden Daten bezogen und systematisiert an das Frontend weitergegeben. Das Frontend bereitet die Daten im Browser graphisch auf, ermöglicht es der Oberfläche sich zu verändern und mit dem User zu interagieren. Vor kurzem bin ich auf den Ausdruck „middle-end“ gestoßen. Für sich sprechend, manifestiert er die – durch das Aufkommen von Web-Apps (Websites die auch nach Seitenaufruf, ohne neu zu laden, mit dem Server kommunizieren) – mehr und mehr bedeutende Schnittstelle von Frontend und Backend.

PHP („PHP: Hypertext Preprocessor“): Wird eine URL durch einen Browser abgerufen, werden Inhalte und Daten u.a. von Datenbanken, Text-Dateien, anderen Websites oder Services durch PHP-Dateien auf dem Server abgefragt und ausgelesen. Auch hat PHP-Code Zugriff auf die URL und kann so durch dort definierte Parameter (z. B. „...?parameter=Nudelsuppe&...“) seine Aufgaben in der Inhaltsverwaltung ausrichten. Die formatierten Daten macht es für den Browser als generierten HTML-Code interpretierbar und übergibt ihn dem Frontend. Werden Websites durch PHP erzeugt, spricht man von dynamischen Websites (statische beruhen auf einzelnen HTML-Dokumenten in denen der Inhalt fix eingeschrieben ist).



HTML („Hyper Text Markup Language“): Ist im besuchten Verzeichnis die Datei namens „index.html“ (oder „index.php“, die HTML-Code generiert) gefunden, stellt diese dem Browser, in verschiedene Typen strukturierte Daten zu Verfügung. So erlangt der Inhalt eine erste grobe Gliederung und Hierarchisierung, lädt durch Quellenangaben („src=...“) Dateien (z. B. weitere Programm-Skripte, Medien, Bilder, aber auch ganze Websites) hinzu und kann durch Links („href=...“) andere Verzeichnisse oder Websites adressieren und dorthin weiterleiten.

CSS („Cascading Style Sheet“): Meist über einen Link im Kopfbereich des HTML-Dokuments wird (mindestens) eine CSS-Datei referenziert. Von ihr erhalten HTML-Inhaltsblöcke Gestaltungsanweisungen, werden „gestylt“ – Farben, Größen, Positionen, Layout-Verhalten, Schriftbild, teils auch Animationen werden festgelegt. Sie können in Abhängigkeit zur Größe, bzw. des Formates des Browserfensters gesetzt werden und so ihre Erscheinung der Umgebung anpassen (Responsive Design). Die gestufte Struktur des CSS-Codes bietet darüber hinaus nicht viel mehr Möglichkeiten und wird bei größeren Mengen schnell unübersichtlich und unflexibel. CSS-Präprozessoren wie SCSS, Sass oder Less bieten zum Glück eine vereinfachte Schreibweise und mehr Funktionsumfang, machen den Vorgang des Stylings so deutlich dynamischer und effizienter. Die Dateien mit dem „Pseudo-CSS-Code“ werden unabhängig von der Website durch ein Compiler-Programm, in reinen, auf Wunsch reduzierten, CSS-Code umgewandelt, welcher letztendlich in die Webseite eingebunden werden kann.

JS („JavaScript“): Es hat sich etabliert, dass JavaScript-Dateien im unteren Bereich einer Website eingebunden werden, denn der Quell-Code einer Webseite wird von oben nach unten gelesen. Das JavaScript wird über ein Source-Attribut („src=...“) im HTML-Code erreichbar gemacht und erst nach dem Auslesen des oberhalb liegenden Inhaltes in die Website geladen.



So werden Lade- und Darstellungskonflikte vermieden. Die HTML-Struktur wird nun als DOM („Document Object Model“), in einer Art verschachtelter Baumstruktur dargestellt, wo jedes HTML-Element als Objekt für das JavaScript sichtbar ist. Etliche Informationen, z. B. zu Typ, Inhalt, Attributen, Darstellung im Browser (Maße, Position, Style, etc.) und Verknüpfungen zu anderen Elementen (Eltern-, Kind-, Geschwister-Elemente) oder Funktionen, sind nun erreichbar und können jeder Zeit, in endlosen Logik-Schleifen und nach jedem gewünschten Entscheidungsverfahren manipuliert werden. JavaScript dient auch der Ereignisbehandlung. Ob der User klickt, tippt, scrollt, lediglich den Cursor bewegt oder in einen anderen Tab wechselt (u.v.m.), kann abgegriffen und darauf reagiert werden. Auch können Systeminformationen (je nach Gerät, Betriebssystem und Browser) wie Batteriezustand, Browser-Typ und Zeit ausgelesen werden. Eine Unterkunfts-Buchungsseite erhöhte so die Preise, die den Nutzern angezeigt wurden, die mit einem Smartphone und niedriger Batterieladung surfen. In Bezug zum oben erwähnten Begriff „middle-end“ spielt JavaScript im Werdegang des Web-Designs eine sehr große Rolle: Es kann mit dem Server kommunizieren und so Daten (nach-)laden – ohne dass die komplette Seite erneut aufgerufen werden muss. Websites die Medien erst laden, wenn diese ins Browser-Fenster gescrollt wurden, bzw. am Ende einer Seite weiteren Inhalte hinzufügen („endless scrolling“, „infinite scrolling“) bauen darauf auf. Man kann heutzutage sowieso schon fast nicht mehr von Web-„Seiten“ (der User hangelt sich von Seite zu Seite um Inhalt zu konsumieren) sprechen, wohl besser von Streams (der User muss immer weniger [kompliziert] interagieren um Inhalt zu konsumieren), die den Großteil ihrer Leistungskraft, Reaktivität und Flexibilität aus der Anwendung von JavaScript schöpfen.



2.3. Eine Synthese im Browser

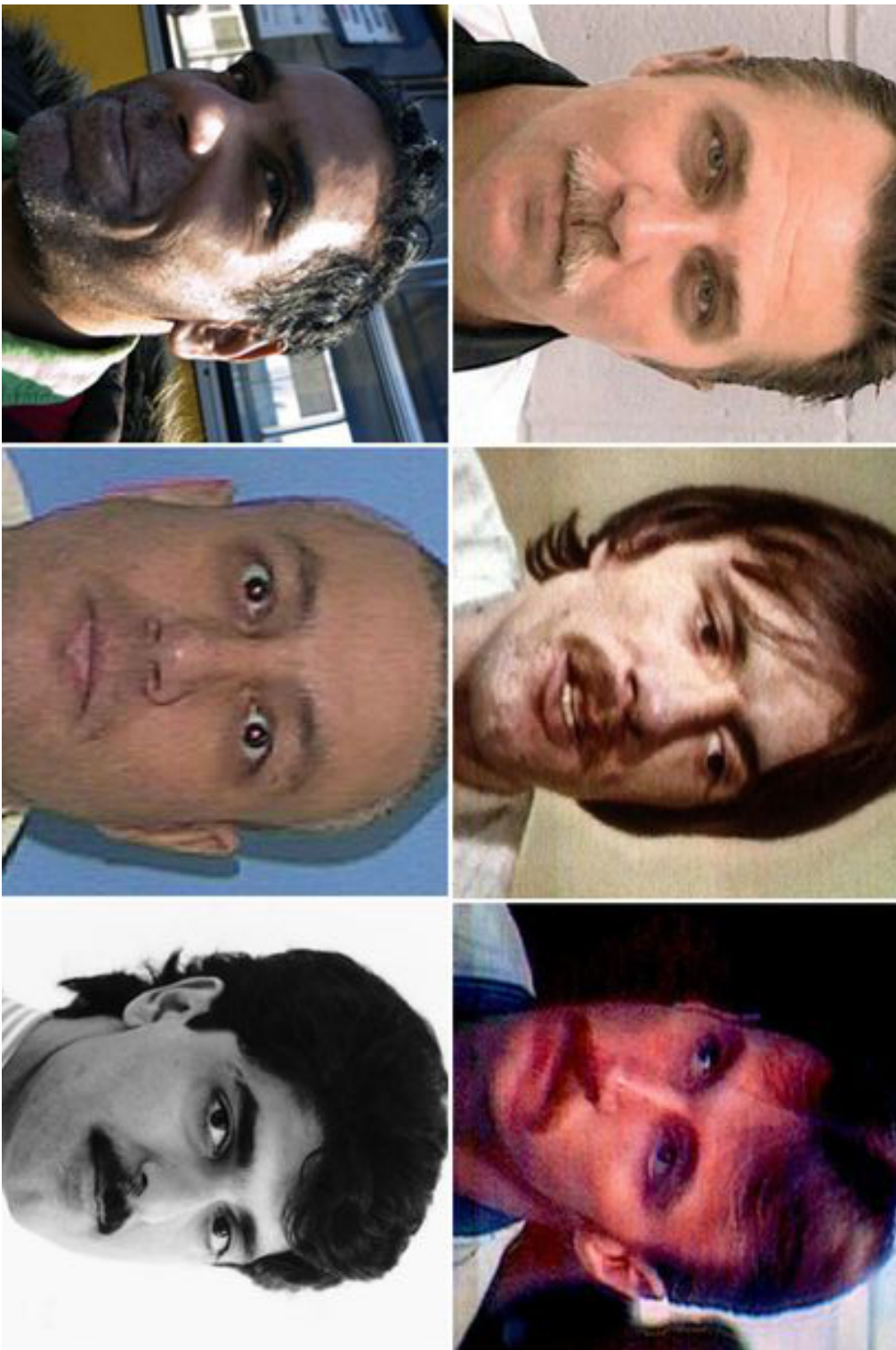
Der Code von Really Simple Satisfaction setzt sich, auf dem Server, aus 7,09% PHP, 1,73% HTML, 7,64% CSS und 79,78% JavaScript zusammen (sowie 3,75% der Bilddatei des Loading-Spinners).

Dem Aufruf der URL von Really Simple Satisfaction folgend, wird dem Browser ein statisches HTML-Dokument ausgeliefert, welches die Struktur des Interfaces mit vorgegebenem Inhalt bereitstellt. Die wichtigsten Elemente dieser Struktur sind:

- Hauptüberschrift am oberen Rand,
- drei Textfelder am unteren Rand,
- Fortschritts-Balken oben und unten,
- ein Feld für den Loading-Spinner
bzw. später umgewandelt zum Fortfahren-Button,
- der (noch leere) Haupt-Inhalts-Container,
- das Options-Menü,
- sowie die Playlist-Anzeige.

Die verlinkte CSS-Datei gibt dem Interface seine optische Erscheinung, setzt Schriftbild und Layout, ein erstes Farbschema, sowie Abwandlungen für kleinere Browser-Fenster und Hover-Zustände (Maus befindet sich über Element) fest.

Mit einer extern generierten JavaScript-Datei („modernizr.js“), die im Seiten-Kopf eingebunden ist, lasse ich lediglich überprüfen, ob die Seite durch ein touch-fähiges Gerät aufgerufen wurde („Feature-Detection“) und den HTML-Code diesbezüglich kennzeichnen.



Ist sie geladen, beginnt die kompilierte Haupt-JavaScript-Datei („bundle.js“) ihr Werk.

Ein Web-Worker wird deklariert und startet eine Anfrage an einen externen Server, bekommt dadurch die IP-Adresse des Users zurückgeschickt und übersetzt sie in einen RGB-Farbwert, welchen er an das Haupt-JavaScript weiterleitet. Ein Web-Worker ist lediglich eine eigene JavaScript-Datei die außerhalb der Haupt-Datei arbeitet und mit dieser kommuniziert – es kann so Rechenleistung aus dem Browser-Fenster ausgelagert und mehr in die Darstellung der Seite investiert werden. Die IP-Adresse ist die zugewiesene Adresse des eigenen Gerätes im Internet, bestehend aus mehreren Zahlen- und/ oder Buchstabenkombinationen. Der Web-Worker lässt so bestimmte Elemente des Interfaces eine Farbe erhalten. Ist dies erledigt, wird er außer Kraft gesetzt und eine Lade-Sequenz gestartet. Diese Choreographie aktualisiert die mittlere Textbox, mit von mir ausgewählten Inhalten und erhöht die Breite der Fortschritts-Balken in vorgegebenen Rhythmus. Am Ende wird der Fortfahren-Button sichtbar.

Zeitgleich zum oben beschriebenen, startet ein weiterer Web-Worker, dessen Aufgabe es durchgehend ist, eine Liste von URLs (sie unterscheiden sich nur anhand des Länder-Codes) anzufragen und deren Rückgabewerte an das Haupt-JavaScript zu übermitteln.

Die angefragten URLs setzen sich hier jeweils aus zwei einzelnen URLs zusammen: Eine URL, die zu einem PHP-Programm auf dem beheimateten Server führt und daran angehängt, als URL-Parameter, eine URL die auf eine RSS-Datei von Google-Trends (auf einem Server von Google), dem eigentlichen Ziel, zeigt („Adresse-der-Website“/“Pfad-zum-PHP-Programm“/?url=„Adresse-der-Google-Trends-RSS-Datei“).



Das PHP-Programm dient der Verwaltung der abgefragten Daten (als Parser), sowie als Proxy-Server (als Vermittler). Auf die Bedeutsamkeit dieses Vermittlers wird in einem späteren Kapitel noch ausführlicher eingegangen.

Über den Proxy-Server gelangt der Web-Worker an die von Google-Trends zu Verfügung gestellten Daten und schickt diese an das Haupt-JavaScript zurück. Sprich: Datensätze für 48 Länder, mit jeweils den 20 meist gesuchten Begriffen – sowie zusätzlich zu jedem Begriff, ein bis fünf URLs von Websites, die dem Trend entsprechende Themen behandeln. Für jede Trend-URL startet der Web-Worker erneut eine Anfrage über den Proxy-Server und bekommt von ihm sortierten Inhalt jeder einzelnen Website ausgeliefert (Inhalte aller Text- und Überschriftfelder, URLs aller Video-, Audio-, Bild- und eingebundenen Elemente [z. B. YouTube-Player, Google-Maps, etc.]). Pro Website wird ein Datensatz (im JSON-Format [“JavaScript Object Notation”]: Datei-Format zum Datenaustausch, das in praktisch allen verbreiteten Programmiersprachen interpretierbar ist) an die Haupt-JavaScript-Datei zurückgeschickt und in entsprechenden Inhaltslisten („Arrays“) zwischengespeichert. Der Prozess beansprucht einiges an Netzwerkleistung (es wird versucht bis zu über 4.000 Webseiten anzusprechen) und läuft daher, abhängig von der Qualität der Internetverbindung, noch einige Zeit nach Ende der Lade-Sequenz im Hintergrund weiter.

Um der Website Interaktivität zu verleihen werden zu Beginn sogenannte Event-Listeners für die Ereignisbehandlung registriert.

Es wird festgelegt was passiert, wenn der User scrollt:

- Die Hauptüberschrift und die Textboxen bewegen sich aus dem Sichtfeld, ändern dann ihren Inhalt und bewegen sich nach einer jeweils zufällig bestimmten Zeit wieder an ihre Ausgangsposition.



이 페이지에서 콘텐츠를 로드하는 데 시간이 걸릴 수 있습니다.

콘텐츠가 로드되는 동안 페이지 콘텐츠를 숨기거나 새로고침을 시도하지 않습니다.
Beta

- Das Hintergrundbild der Seite wird ausgetauscht.
- Der Seitentitel und das Favicon (Benennung der Seite dargestellt im Browser-Tab und das voran stehende Symbol) wird ausgetauscht.
- Die Meta-Daten im Kopfbereich der Seite werden ausgetauscht. Sie sind nicht sichtbar, sondern stellen Informationen (URL, Titel, Beschreibung, Pfad zu einem Bild) bezüglich der Seite, für das Teilen auf externen Plattformen („Sharing“, z. B. über Facebook) zur Verfügung.
- Die Breite der Fortschritts-Balken wird aktualisiert, anhand der Relation der momentanen Scroll-Position und Gesamthöhe der Seite.
- Alle Video- und Audio-Elemente die geladen sind und noch nicht abspielen, werden gestartet, auf Endlosschleife gestellt und eine bestimmte Lautstärke wird ihnen zugewiesen. Ihr Dateiname wird in die Playlist-Anzeige eingefügt und diese wird für kurze Zeit sichtbar.
- Sobald der User vier Sichtfeld-Höhen vom unteren Ende der Seite entfernt ist, wird zufällig gewählter Inhalt nachgeladen und aus der Inhaltsliste entfernt. Ist kein Inhalt mehr vorrätig, kann man bis ans Ende der Seite gelangen und wird wieder an den Anfang zurückgeworfen.

Klickt der User auf den Fortfahren-Button, wird/ werden:

- Dieser für kurze Zeit erneut durch einen Loading-Spinner ausgetauscht, dann komplett ausgeblendet.
- Die Textboxen ein Mal (wie oben schon beschrieben) aktualisiert.
- Ein Hintergrundbild geladen.
- Ein erster Schwung an zufällig gewählttem Inhalt in den Haupt-Inhalts-Container geladen (und aus der Inhaltsliste entfernt).
- Der Haupt-Inhalts-Container eingeblendet.



Klickt (Links- oder Rechtsklick) der User nach erfolgreichem Fortfahren auf den Haupt-Inhalts-Container:

- Wird das Optionsmenü eingeblendet.

Klickt der User auf eine Option des Menüs wird diese ausgeführt:

- Auto-Scrolling wird (de-)aktiviert. Die Seite scrollt nun automatisch eine Sichtfeld-Höhe nach unten, pausiert für einen Moment und wiederholt dies endlos. Die Art von Animation wird ebenso durch JavaScript erzeugt. Dem User wird es zu dieser Zeit nicht mehr erlaubt die Seite von selbst zu scrollen.
- Alle Video- und Audio-Elemente werden stumm oder wieder laut geschaltet.
- Anhand der Meta-Daten wird zur Sharing-Funktion von Facebook weitergeleitet.
- Die Seite wird erneut geladen.
- Das Optionsmenü wird ausgeblendet.

2.4. Scraping, SOP, CORS & Proxy

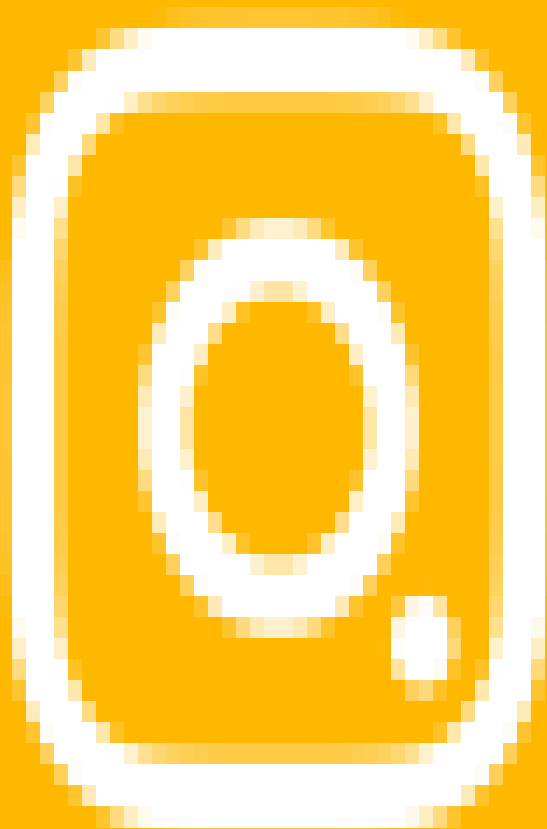
Die Infrastruktur von Really Simple Satisfaction fragt im Moment des Aufrufes bis zu über 4.000 fremde Websites automatisiert an, um deren Inhalt zu beziehen („Web-Scraping“). Ein Bild oder einen Text einer Website durch eigene Hand zu kopieren, weiterzuverarbeiten oder sich in irgendeiner Art und Weise



anzueignen, hat sich mittlerweile ganz natürlich in den Alltag vieler integriert. Geschieht dies nicht durch einen User und in tausendfacher Ausführung greifen mehrere Sicherheitsmechanismen ein, denn grundsätzlich besteht jede Website auf ihren Inhalt und dessen Herkunft.

Die „Same-Origin-Policy (SOP)“ besteht bereits seit 1995 als in allen modernen Browsern implementiertes Sicherheitskonzept, was es den im Frontend einer Website, bzw. letztendlich im Browser arbeitenden Programmiersprachen, wie JavaScript, verbietet auf Objekte (ganze Seiten, Skripte oder auch einzelne Dateien) anderer Websites oder anderer Herkunft (Speicherort, also Server) zuzugreifen. Attacken wie Datendiebstahl von Cookies oder das Einbinden von schädlichen Skripten in Websites sollen dadurch bereits durch den Browser blockiert werden. Die Herkunftsprüfung bezieht sich auf die Zusammensetzung der URL aus Netzwerkprotokoll („http://“ oder „https://“), der Domain und des Server-Ports.

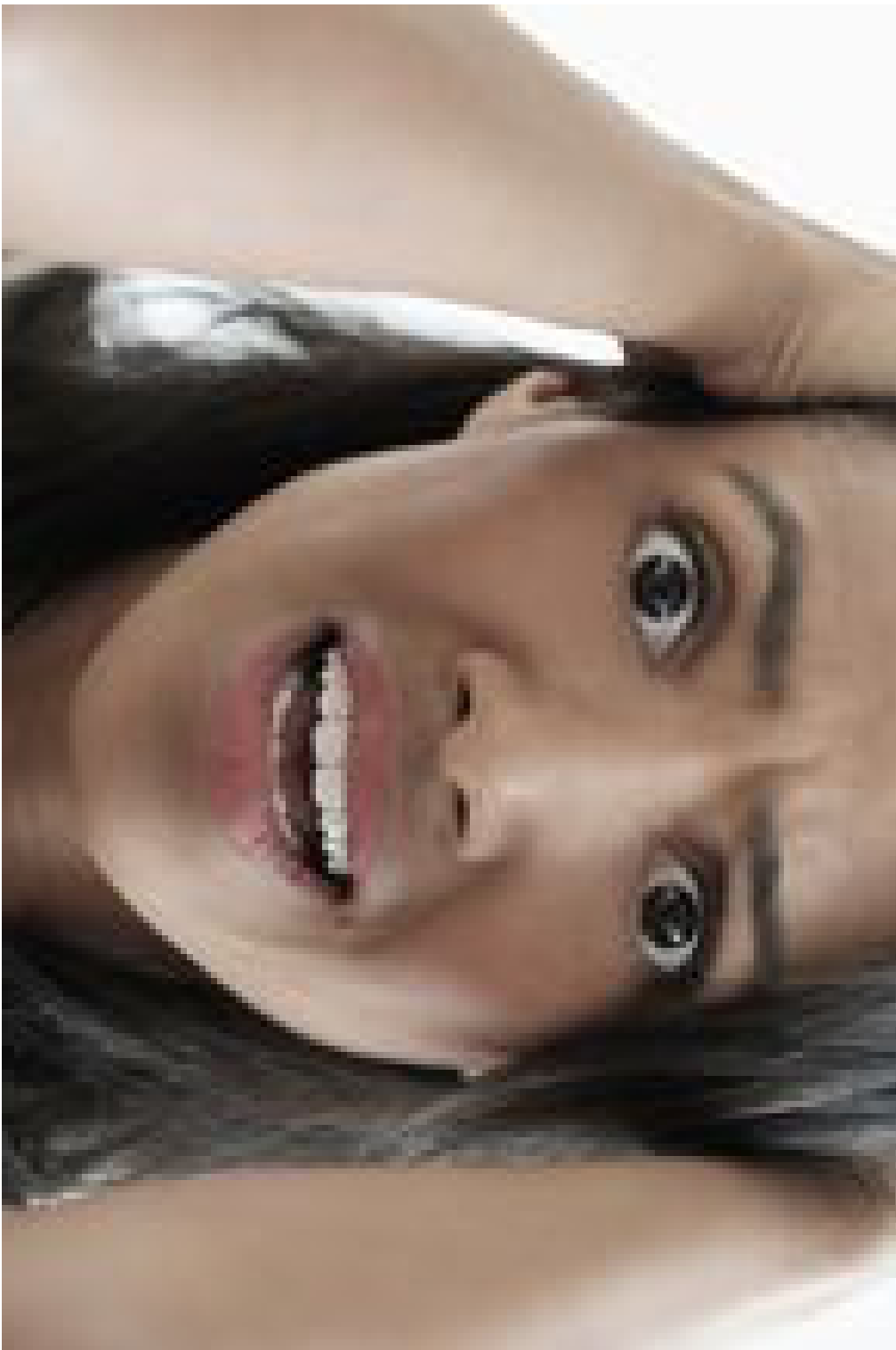
Websites, WebApps und deren Entwicklung werden immer umfangreicher und komplexer, deshalb werden Teile davon auf verschiedene Server oder Subdomains ausgelagert. Es kommt also häufig vor, dass sich eine Seite „https://service.beispiel.xyz/“ an Daten oder Mechanismen der Seite „https://www.beispiel.xyz/“ bedienen möchte, oder dass in einer Entwicklungsumgebung das Frontend und das Backend einer Seite auf verschiedenen Server(-Ports) laufen (z. B. „http://entwicklungsumgebung:8080“ möchte die Daten von „http://entwicklungsumgebung:8888“ darstellen). Freiheit, Flexibilität und Funktionalität wird hier von SOP deutlich eingeschränkt und es wurde ein Mechanismus namens „Cross-Origin Resource Sharing (CORS)“ eingeführt, der mittlerweile von fast allen Browsern unterstützt wird. Eine Möglichkeit CORS zu aktivieren, ist das Einfügen einer einzigen Zeile Code („header(„Access-Control-Allow-Origin: https://service.beispiel.xyz/“);“) in eine serverseitige PHP-Datei.



Um diese Datei bearbeiten zu können braucht man selbstverständlich die Zugriffsrechte des Servers, so ist weiterhin ein hoher Sicherheitsstandard gewährt.

Besitzt man diese Zugriffsrechte nicht, möchte aber dennoch automatisiert fremde Seiten besuchen, schlagen einem die Suchmaschinen etliche CORS-Proxy-Services vor. Solch ein Proxy ist eine Website deren Server als Vermittler zwischen der anfragenden (Quelle) und der angefragten (Ziel) Website steht. Die Anfrage an die fremde Seite wird so also nicht mehr direkt vom Browser vorgenommen, sondern an den Proxy-Server weitergereicht. Da diese externen CORS-Proxy-Services frei verfügbar sind, werden sie selbstverständlich auch von unzähligen anderen Entwicklern und Betreibern verwendet, dadurch mehr oder minder schnell von vielen Websites identifiziert und blockiert. Zudem werden sie dem Zweck der Anonymisierung nicht gänzlich gerecht: Sie übermitteln teilweise die Herkunft der Anfrage, welche bei erhöhtem Aufkommen von der SOP-Technik im Browser unterbunden wird. So kam ich während der Entwicklung von Really Simple Satisfaction schnell und einfach an die gewünschten Daten, konnte mich teilweise ganze Wochen auf diesem Erfolg ausruhen, aber wurde letztendlich immer und mit jedem dieser Services wieder auf 0 zurückgeworfen.

Der einzige Ausweg der sich aus dem Nebel meiner Verzweiflung auftat, war es einen Proxy-Server zu schreiben, der maßgeschneidert auf dem eigenen Server arbeitet. Als alleinige PHP-Datei von Really Simple Satisfaction, aktiviert er CORS, startet von sich aus die Anfrage (bekommt diese also nicht nur weitergereicht) und macht die Herkunft dieser unersichtlich. Die Adresse des Ziels erhält er über den Wert des URL-Parameters seiner Anfrage („?url=...“; siehe oben: „Adresse-der-Website“/“Pfad-zum-PHP-Programm“/?url=„Adresse-der-Google-Trends-RSS-Datei“ oder „Adresse-der-Website“/“Pfad-zum-PHP-Programm“/?url=„Adresse-der-Ziel-Website“).



Neben der Anonymisierung dient der Proxy auch als RSS-Parser, bzw. DOM-Parser: Er bereitet den abgefragten Quell-Code der einzelnen Seiten auf, extrahiert nach festgelegtem Schema die benötigten Daten und wandelt sie in JSON-Formatierung um. Im nächsten Schritt speichert er jeden Datensatz einer Webseite in eine eigene Textdatei, in einem Verzeichnis auf dem Server. So fungiert er auch als Zwischenspeicher („Cache“) und prüft bei Aufruf, ob es die Textdatei für die parametrisierte RSS-/ Website-URL bereits gibt und ob sie älter als 12 Stunden ist (die Trends des Tages aktualisieren sich alle 24 Stunden). Nur wenn diese veraltet oder noch nicht vorhanden ist, spricht er die dementsprechende Website an und verarbeitet deren Inhalt – andernfalls entnimmt er den Inhalt der noch aktuellen Textdatei. Es müssen nicht mehr ständig, durch jeden Besuch eines einzelnen Users, 4.000 Websites erreicht werden, was der Netzentlastung genauso wie der Browser-Leistungsfähigkeit erheblich zugutekommt – und je weniger automatisierte Anfragen von einer Website an andere gehen, desto unauffälliger ist deren Handeln im Netz.



3. Konsum & Wahrnehmung

Laut der App Screen Time verbringe ich am Tag durchschnittlich 9 Stunden und 13 Minuten vor den Bildschirmen meines Laptops und meines Smartphones. Im Durchschnitt bekommt eine Website für 15 Sekunden die Aufmerksamkeit eines Users, bevor er sich dazu entscheidet diese zu verlassen. Die maximale Länge eines Videos einer Story bei Facebook beträgt 20 Sekunden, bei Instagram 15 Sekunden und eines Snaps bei Snapchat 10 Sekunden. Die Formen des Konsums von Printmedien und Fernsehen, selbst die eines Lagerfeuers, wurden längst vom Internet absorbiert und multipliziert. Die Online-Ökonomie bringt immer frequenter Neuerungen hervor um sein Publikum zu paralisieren und zu analysieren – um alle Aufmerksamkeit zu gewinnen, sie in Daten zu transformieren und dank dieser noch effizienter vorzugehen. Die Vermarktung von Videorekordern, VHS-Kassetten und Camcordern erlaubte es erstmals der breiten Masse in das Medium des Bildschirms einzugreifen und es zu manipulieren, sich der wirtschaftlichen Kontrolle zu entziehen und diese in Frage zu stellen. Kulturelle Bewegungen von Videokunst, über Medienkunst, Digitaler Kunst, Digitaler Netzkunst, net.art bis hin zur Post Internet Art verarbeiten seit dem ästhetische, soziologische, politische und psychologische Aspekte, sowie Prozesse des Lebens am Draht, vor dem Screen und dahinter.

Die Lesezeit dieses Kapitels beträgt 9 Minuten und 38 Sekunden.



3.1. Informationsaufnahme & Interaktion am Bildschirm

Der Bildschirm ist wohl die präsenteste Oberfläche der heutigen Zeit. Er hat die Erfahrung unseres Alltags, unsere Art zu sehen, zu denken und zu handeln immens verändert. Wir wachen auf und gehen unsere Mitteilungen am Smartphone durch. In öffentlichen Verkehrsmitteln sind mehr als die Hälfte der Menschen im Screen vertieft. Er wird genutzt, zu einem großen Teil in vielen Berufsfeldern, in Sachen der Bildung, um Zahlungen abzuwickeln, um zu kommunizieren, um uns unterhalten und zerstreuen zu lassen, manchmal auf der Couch zusätzlich mit einem zweiten Bildschirm auf dem Schoß. Dann gehen wir wieder schlafen, mit ihm auf dem Nachttisch neben uns liegend.

Als Objekt entsprang der Bildschirm der Malerei: Seine Abmessung, Form und Konstruktion sind mit der von einer Leinwand vergleichbar. Seit dem befindet er sich in einer scheinbar endlosen Metamorphose – von der Kinoleinwand zum Fernsehgerät, vom Computer-Monitor zum Laptop-Screen, vom Smartphone-Display zur Smartwatch, vom Virtual-Reality-Headset bis hin zur Netzhaut des Auges, auf die direkt projiziert wird. Es geht heute nicht mehr ausschließlich darum möglichst große, oder möglichst kleine Bildschirme zu entwickeln, sondern um eine Varianz und Vielfalt verschiedener Größen, die jeweils auf einen bestimmten Zweck ausgelegt sind.

Die Bildschirme des Kinos oder des (analogen) Fernsehens sind nicht in der Lage die Art von Geschwindigkeit, Sensibilität und Reichhaltigkeit spürbar zu machen, wie es die von Smartphones, Laptops und Computern durch ein Interface tun können. Das Wort „Interface“ beschreibt eine Art Schnittstelle, eine Transaktion und einen Vermittler – es sieht nach innen, ist aber nur nach außen hin sichtbar. Es beeinflusst, wie wir den Bildschirm betrachten, die

Informationen aufnehmen und erst durch die Wahrnehmung von ihm, wird die Grundlage für die Interaktion mit der Maschine geschaffen. Unser Konsum und unser Handeln am Bildschirm erscheint uns als Aktivität, dabei sind wir deutlich passiver als wir denken. Unsere Möglichkeiten und Tätigkeiten werden vom Interface, anhand von festgelegten Gestaltungskonzepten und einer reaktiven technologischen Architektur, limitiert und kontrolliert, obwohl man doch eigentlich immer das Gefühl hat, dass man selbst die Kontrolle über das Gerät hat.

Auch wenn wir uns in der Rolle des Kontrollierenden pudelwohl fühlen, bevorzugen wir es oft, Entscheidungen oder Auswahlverfahren aus unseren Händen zu geben und der Passivität ihren Lauf zu lassen. Seitdem YouTube die Autoplay-Option eingeführt hat, spielt nach dem Ende eines Videos automatisch ein neu vorgeschlagenes ab. Man muss sich nicht mehr dazu entscheiden weiter zu machen oder aufzuhören, die Wahl welches Video man als nächstes ansehen möchte fällt weg und es ist auch nicht mal mehr nötig einen Finger zu heben um es zu starten. Lässt man sich auf dieses treiben-Lassen ein, kann man feststellen wie man sich nach und nach immer mehr vom Kontext des Ausgangsvideos entfernt und man in eine Art "Main-Stream" gelangt, der so erscheint als würde er letztendlich alle passiv Konsumierenden in sich hineinziehen. (Die Plattform lässt vor jedem Video einen Werbeclip laufen, so kann sie durch Autoplay deutlich mehr Einnahmen erzielen.) Da viele User Autoplay hier als lästig empfinden, eventuell auch weil die Vorschläge selten attraktiv und oft aus dem Kontext gerissen sind (YouTube besteht aus sehr unterschiedlichen Inhalt – von Amateur- bis High-End-Produktionen, von 10-Stunden-Entspannungs-Videos bis 10-Sekunden-Spass-Videos), ist es auch nur bei einer Option geblieben, die schnell ersichtlich deaktiviert werden kann. Anders wird es bei der Video-on-Demand-Plattform Netflix gehandhabt. Der Inhalt ist homogener, zusammenhängender, bzw. baut aufeinander auf. Eine Folge einer Serie reiht sich nahtlos an die andere, aus dem Vorhaben ein

MELIÄ

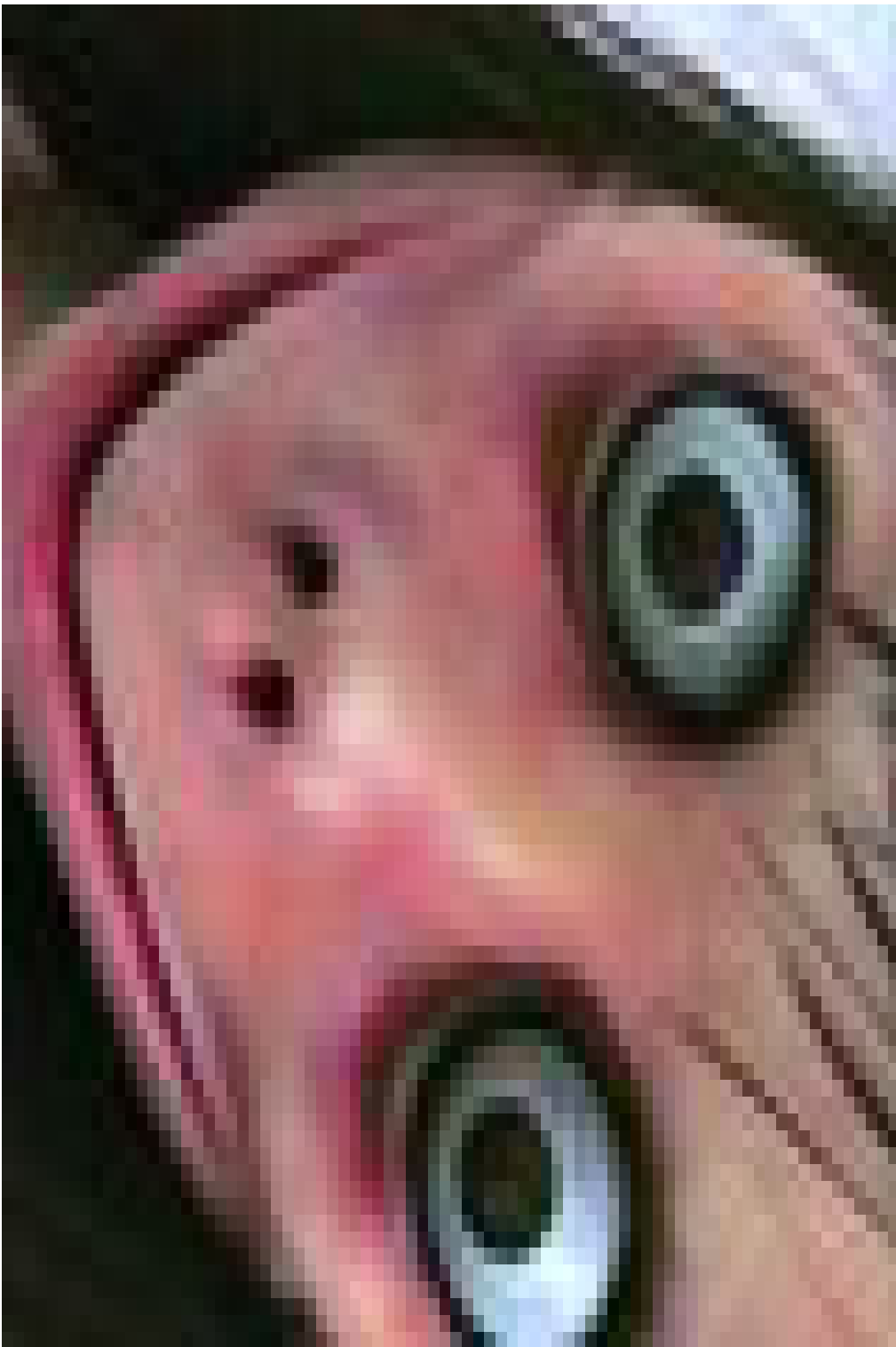
HANOI

VIETNAM

Dear Valued Guests,

Thank you for choosing our hotel for your stay in Hanoi.

We would like to inform you that the security scanner will be installed at our hotel lobby by the diplomatic protocol of Vietnam due to the visit of a Head of State staying at our hotel. Please also note that the security will be re-inforced in all areas of the hotel.



bis zwei Episoden zu schauen, wird eine ganze Staffel, das Binge-Watching hat begonnen. Autoplay kann auch hier deaktiviert werden, man muss das Einstellungs-Menü erst aber überhaupt finden (wollen). Anhand persönlicher Vorlieben setzt Netflix die am meisten wünschenswerten Genres weit oben in die Vorschlagsübersicht, angeführt von der „Top-Auswahl für Stefan“ und überall dem startet der Trailer des Haupt-Features, in voller Fensterbreite, drei Sekunden nach Aufruf der Seite, automatisch. Dieses Entgegenkommen des Interfaces ist nicht nur Teil des Repertoires von Bewegtbildanbietern: Selbst die Suchmaschine Google besitzt einen „Auf gut Glück!“-Button. Im Falle des Falles, wenn man einmal nicht weiß was man denn überhaupt suchen soll, führt er auf eine Seite die aktuell begehrte Beiträge und Themen, gemischt mit der Historie von Google bereitstellt. Ein „Auf gut Glück!“-Button ist sinngemäß auch an fast jedem Spielautomaten eines Casinos vorhanden um dauerhaft automatische Walzendrehungen durchzuführen. Man muss also erst wieder eine Taste betätigen, wenn alles Geld in den Automaten hinein- oder herausgerieselt ist.

Die Geschwindigkeit der Interaktion und die Dichte von Information bei Spielautomaten wirkt anfangs überfordernd, aber wird dann schnell angenommen. Man gewöhnt sich daran und es entsteht eine beruhigende Wirkung, die den Spieler davon ablenkt bzw. ihn besänftigt, dass er eigentlich durchgehend mit einer hohen Wahrscheinlichkeit Verlust macht. Ähnliche beruhigende Wirkungen findet man in unzähligen psychedelisch-bunten Smartphone-Games, Eltern setzen ihre Kinder vor den Fernseher um sich nicht mit Erziehung oder Sorgen auseinandersetzen zu müssen, Kindes Augen machen dort keinen Unterschied mehr zwischen Werbung und dem eigentlichen Programm – und immer häufiger sieht man Tablets als Bordcomputer in Kinderwägen mitfahren. Das Scrollen durch Social-Media-Feeds ist Hypnose auf der selben Ebene. Man scrollt und scrollt, mehrmals in der Stunde und am Ende des Tages kann man sich wahrscheinlich nur an maximal ein Like



erinnern was man vergeben hat und der Feed nimmt kein Ende. Macht sich irgendwann mal ein kleines Stück Zeit auf, z. B. an der Haltestelle oder in einer Warteschlange, zünde ich mir eine Zigarette an und scrolle, ist es in einem geschlossenen Raum, scrolle ich lediglich. Die Zusammensetzung der Beiträge die ich sehe folgt keinem fassbaren Strang mehr, sie wird von Algorithmen generiert, bei jedem Aufruf neu, was der Zerstreuung und Fesselung des Geistes natürlich zugutekommt.

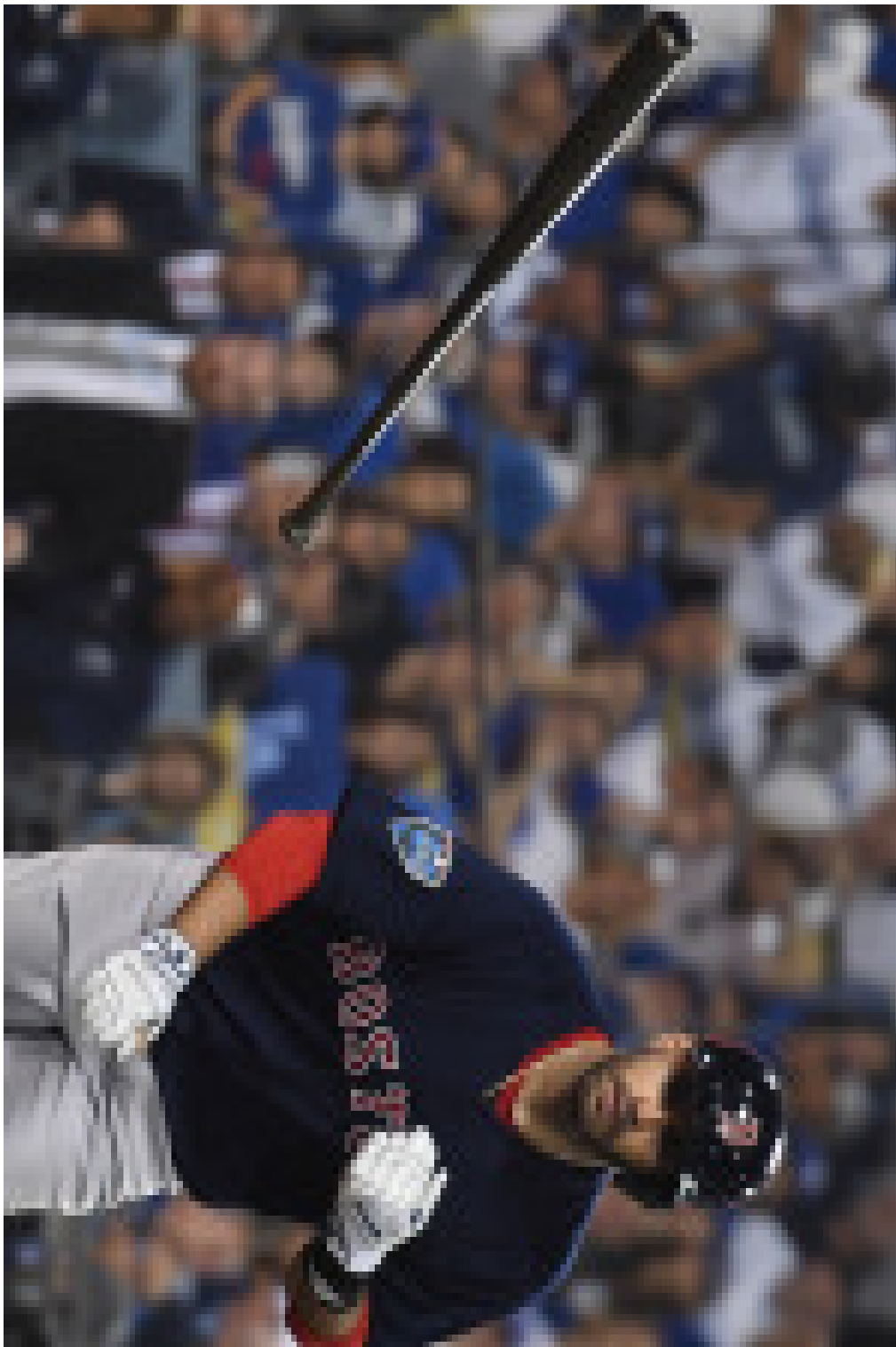
Das Aufkommen und Dasein der Feed-Algorithmen von Facebook, Instagram und Twitter steht in einem ständigen Diskurs bei den Nutzern. Instagram verabschiedete sich vor einigen Jahren vom chronologischen Feed, es scheint so als hätte die Mehrheit dies akzeptiert und längst vergessen, Twitter hat ihn allerdings seit kurzem als Option wieder eingeführt – Instagram möchte mitziehen und ihn auch gerne bald wieder zur Verfügung stellen... Die Algorithmen basieren auf den ersten Blick auf ähnlichen Prinzipien und orientieren sich nach Attraktivität und Beliebtheit, Beziehungen und Verknüpfungen, Frequenz und Aktualität. „Machine Learning“ trainiert sie dabei kontinuierlich um immer bessere Ergebnisse zu erzielen, was sie in den Status der Künstlichen Intelligenz hebt. Alles was auf dem Bildschirm zu sehen ist, ist eine mystische Version von Prozessen hinter der Oberfläche und wirtschaftsorientierte Geheimhaltung verstärkt den Effekt um weiteres: Die generischen Feed-Algorithmen können nicht frei eingesehen werden, sie werden von den Konzernen gehütet wie all ihre Augäpfel. Künstliche Intelligenzen kann man als Konsument nicht eindeutig feststellen und handeln sie in einem gigantischen, sich ständig umformenden Organismus von Followers, Posts, Re-Posts, Likes, Stories und Kommentaren, könnte ihr Auswurf doch eigentlich auch auf reinem Zufall basieren.



Die Fragestellung, ob eine Maschine beim Zuliefern von Inhalt wirklich so viel mehr „Intelligenz“ und Wissen nutzen muss, als der Mensch, der beim Konsum scheinbar einen Großteil seines Verstandes auflöst, spielt bei Really Simple Satisfaction eine zentrale Rolle.

3.2. Anwendung & Intention

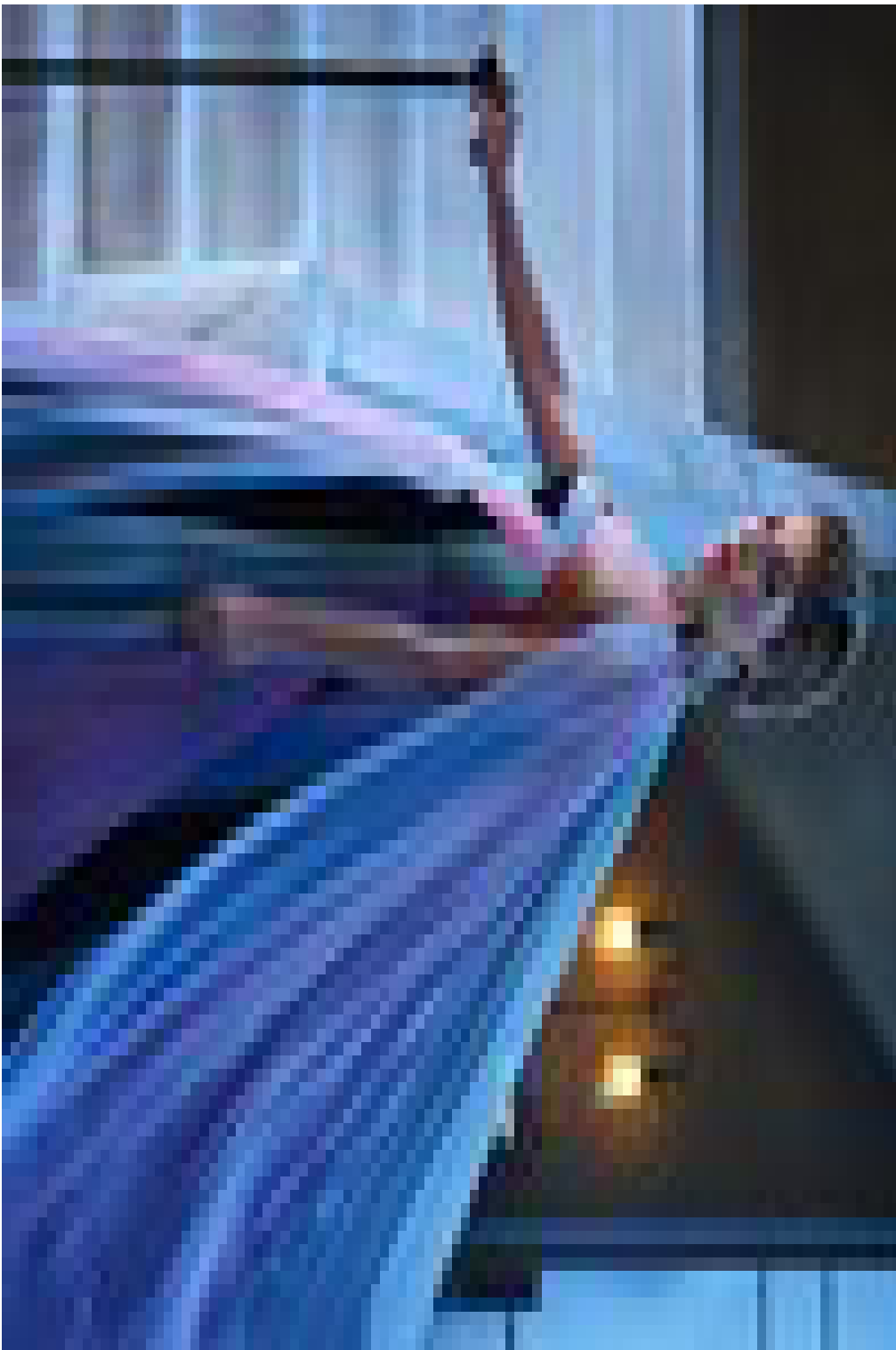
Die wichtigste Interaktion von Really Simple Satisfaction, wie bei den Social-Media-Feeds auch, ist das Scrollen. Durch die vertikale Bewegung hangelt man sich von Eindruck zu Eindruck, parallel wird weiterer Inhalt nachgeladen und kaum bemerkbar an das Ende des Feeds angehängt. Das benutzerfreundliche Interface von Social-Media-Feeds wird hier erweitert und befreit: Der einzelne Beitrag ist nicht mehr in einem vordefinierten Format, wird nie abgeschnitten – seine Größe entspricht den natürlichen Pixelmaßen der jeweiligen Datei. Dadurch wird es ermöglicht nicht, wie gewohnt, nur einen einzelnen Beitrag auf der vollen Breite der Feed-Fläche darzustellen (Instagram „Explore-Feed“ zeigt zumindest zwei bis drei in einer Reihe), sondern deutlich mehr, in einer vielfältigeren graphischen Zusammenstellung. Auch seine vertikale Ausrichtung (ob bündig oben, mittig zentriert, oder bündig unten in der Reihe) wird durch den Zufall bestimmt. Es ergibt sich ein Gestaltungsraster, das intelligent ausgeklügelt wirkt aber nicht nachvollziehbar ist, organisch anmutende Ansammlungen von verschiedenen Inhalten und ein abwechslungsreicher Lesefluss werden erzeugt – ebenso aber auch feine Lücken und ganze Freiflächen. Eine Ebene dahinter, bei jeder noch so kurzen Scroll-Interaktion, poppt ein vollflächig fixiertes Bild nach dem anderen, zufällig gewählten, auf.



Es füllt also die Lücken, ist aber nie ganz einsehbar und oft zwangsmäßig auf ein n-faches vergrößert. Wiederum eine Ebene über allem, bewegen sich Text-Boxen am Rand des Sichtfeldes aus ihm hinaus und wieder herein, mit immer neuem Inhalt. Die einzige im Gestaltungskonzept vorhandene Zusatzfarbe wird durch die IP-Adresse des Nutzers, die je nach Gerät und Ort eine andere ist, errechnet. Sie ist ständig wahrnehmbar und stellt unbedingte Bezüge zu Inhalten her, die ein ähnliches Farbschema aufweisen. Es fällt nur sehr dezent auf, aber manche Bilder gewinnen dadurch passiv an Bedeutung, egal welche Auflösung oder Art von Motiv sie aufweisen.

Der Inhalt von Really Simple Satisfaction wird wo es nur geht aus seinem Kontext gerissen. Er entspringt den 20 populärsten Suchtrends des Tages für jeweils 48 Länder. Diese Ergebnisse stehen in Relation zum totalen Suchaufkommen von Google, was einer Liste von mehreren Milliarden Suchanfragen pro Tag entspricht. Würde man sie in der Schriftart Arial Regular und einer Schriftgröße von 11 Punkt mit doppeltem Zeilenabstand ausdrucken, könnte man sie mindestens 1,5 Mal um die Erdkugel wickeln. Schon allein diese Dimension und die Diversität die die Trends bedingen, lässt sie von Bestimmung durch reine Willkür kaum mehr trennen. Die Prozesse hinter Really Simple Satisfaction lösen sie zusätzlich von ihrem regionalen und chronologischen Ursprung, atomisieren einzelne verwandte Beiträge in viele einzelne Textstücke, Bilder, Interface-Symboliken und eingebettete Medien und liefern sie dem User in Form eines gut geschüttelten audiovisuellen Cocktails aus, der sich immer wieder nachfüllt, sobald man ihn fast leeresogen hat.

Im Vergleich zu den schlichten, eingängigen Interfaces der Social-Media-Feeds, wirkt die Oberfläche von Really Simple Satisfaction noch sehr ungewohnt, überladen und irritierend (stellt den Inhalt aber immer noch in den Mittelpunkt). Dennoch wird früher oder später eine Gewöhnung an diese Umgebung spürbar, in der man immer wieder einen Eindruck gewinnt mit dem



man bereits vertraut ist, der den temporären persönlichen Interessen und dem eigenen Horizont entspricht. Instinktiv treibt uns Neugierde dazu diesen kontinuierlich zu erweitern. Durch die stetige Versorgung an Neuigkeiten lässt sich der vorgegebene Weg schwer verlassen und hält uns davon ab die simulierte Jagd nach Information frühzeitig niederzulegen. Der Fokus verschwimmt und wahllos eingestreute Icons und Navigations-Elemente ohne Funktion werden kaum mehr wahrgenommen. Die im Laufe der Benutzung immer weniger differenzierbare Sound-Ebene verschmilzt nebenbei zu einem Rauschen, was womöglich alles in der Umgebung übertönt. Man steht neben sich und scrollt. Scrollen ist die einzige körperliche Leistung die man im Tunnel von Really Simple Satisfaction aufbringen kann. Die Auto-Scroll-Option ermöglicht aber noch einen weiteren Schritt und Website übernimmt die Interaktion mit ihr selbst. Man kann sich zurücklehnen und abdriften. Während der Entwicklungsphase, auch schon in frühem Stadium des Projekts, wurde so meine Konzentration und Arbeitszeit zwangsmäßig immer wieder von kleineren und größeren Entertainment-Ausflügen durchdrungen (womöglich war das ja auch gar nicht so verkehrt).

Als Schöpfer und Entwickler von Really Simple Satisfaction befreie ich mich davon, so weit es geht, mich für den Inhalt, dessen Herkunft, Thematik und Zusammensetzung verantworten zu müssen, minimiere meine Position als Erzähler und Autor, lasse sie verschwimmen und der Geschichte ihren freien Lauf.

Wie attraktiv ist visueller Inhalt (ein Bild, ein Video) noch, ohne Angaben von Likes, Autor, Veröffentlichungs-Alter, ohne zusätzliche Phrasen und Hashtags?

Wie wird er wahrgenommen und interpretiert, neben mickrig kleinen, verpixelten, völlig fremden Nachbarn, auf einer Bildfläche, die sich mehrmals



in der Sekunde aktualisiert und von einem polarisierenden Farbverlauf überlagert ist?

Was hört man, wenn Podcasts, Nachrichten, Radio-Streams, Amateur-Videos, computergenerierte Stimmen oder Werbungen gleichzeitig spielen, sich immer weiter aufschichten und man nur deren Dateinamen kennt?

Welche Bezüge werden geschaffen, egal ob der Inhalt einem thematisch vertraut ist oder nicht, wenn er fremden Überschriften, Slogans, Schlagworten, Codes, oder ganzen Paragraphen, in diversen Sprachen und Alphabeten gegenübergestellt wird?

Was und wie viel bleibt davon übrig, wenn es oft nicht gelingen kann Informationen rechtzeitig zu erfassen?

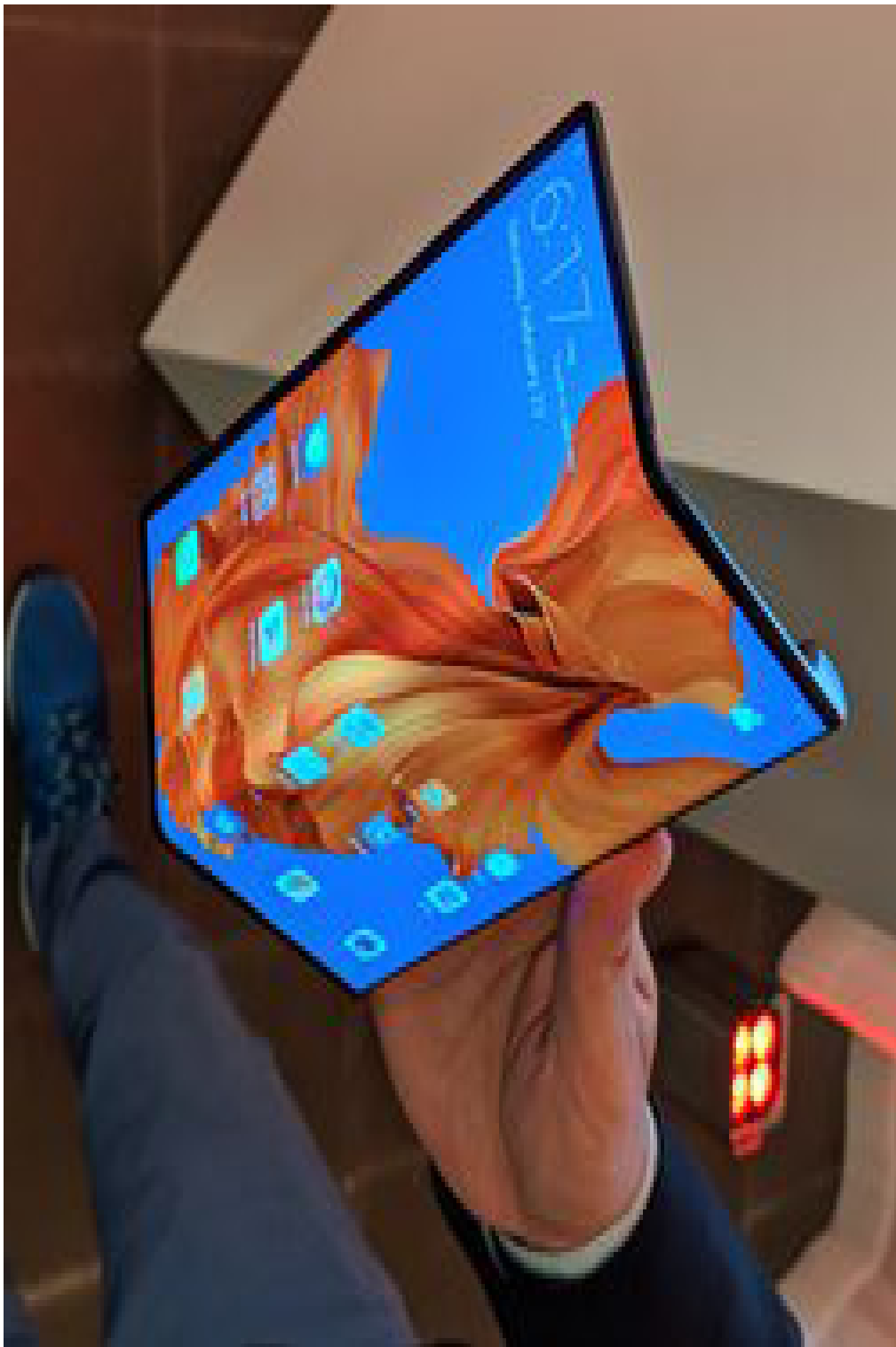
Wie viel sind sie wert, wenn man noch nicht einmal nach ihnen gesucht hat?

à,§à,±à, à, à,µà¹ 21 à, à,,à,jà, à,²à, à,±à, à, à¹ 2562 - 14:48 à, .

Venezuela

Sie sind hier:

Vandaag presenteert Kim Putters zijn boek Veenbrand. Hierin analyseert hij de 'smeulende kwesties' in onze samenleving die hij ziet als directeur van het SCP en die ook uit SCP onderzoek naar voren komen. Wie profiteert van welvaart en vooruitgang? Welke gevolgen heeft toenemende sociale ongelijkheid? Zorgt de duurzaamheidstransitie voor nieuwe sociale scheidslijnen?



Hoe krijgen burgers meer invloed op politici en bestuurders?

Il mondo della moda in lutto per Karl Lagerfeld:
i saluti social sfoglia la gallery

Mobil

Khloé Kardashian Has „Basically“ Forgiven Tristan

GOAL! Arsenal 1-0 BATE (Volkov 4 og); agg 1-1

“차단기 늦게 열었다”...아파트 경비원 폭행

Innovative Cities

Merci d'avoir partagé cet article avec vos amis

Bale's ‚belligerence‘ draws attention to his Real struggles

이미지 캡션

1 Mayıs 1948'de okuyucusuyla buluşan Hürriyet, ilk günden bu yana „Tarafsız ve kaynağından doğru haber“ ilkesinden ödün vermeksizin yayın yaşamını sürdürmekte, sektörünün öncü ve örnek gazetesi konumunu daha da öteye götürerek küresel bir medya markası olma iddiasıyla geleceğe uzanmaktadır.

Lady Gaga's Intimate Performance With Bradley Cooper at Oscars is Now a...



lifestyle trends

[Back to top](#)

A quick moment for Rami and Lucy, the woman who captured his heart.

Вы сейчас просматриваете новость „Спаситель Майдана или изменник Родины: кто такой генерал Замана“. Другие Последние новости Украины смотрите в блоке „Последние новости“

[About Us](#)

8:48pm

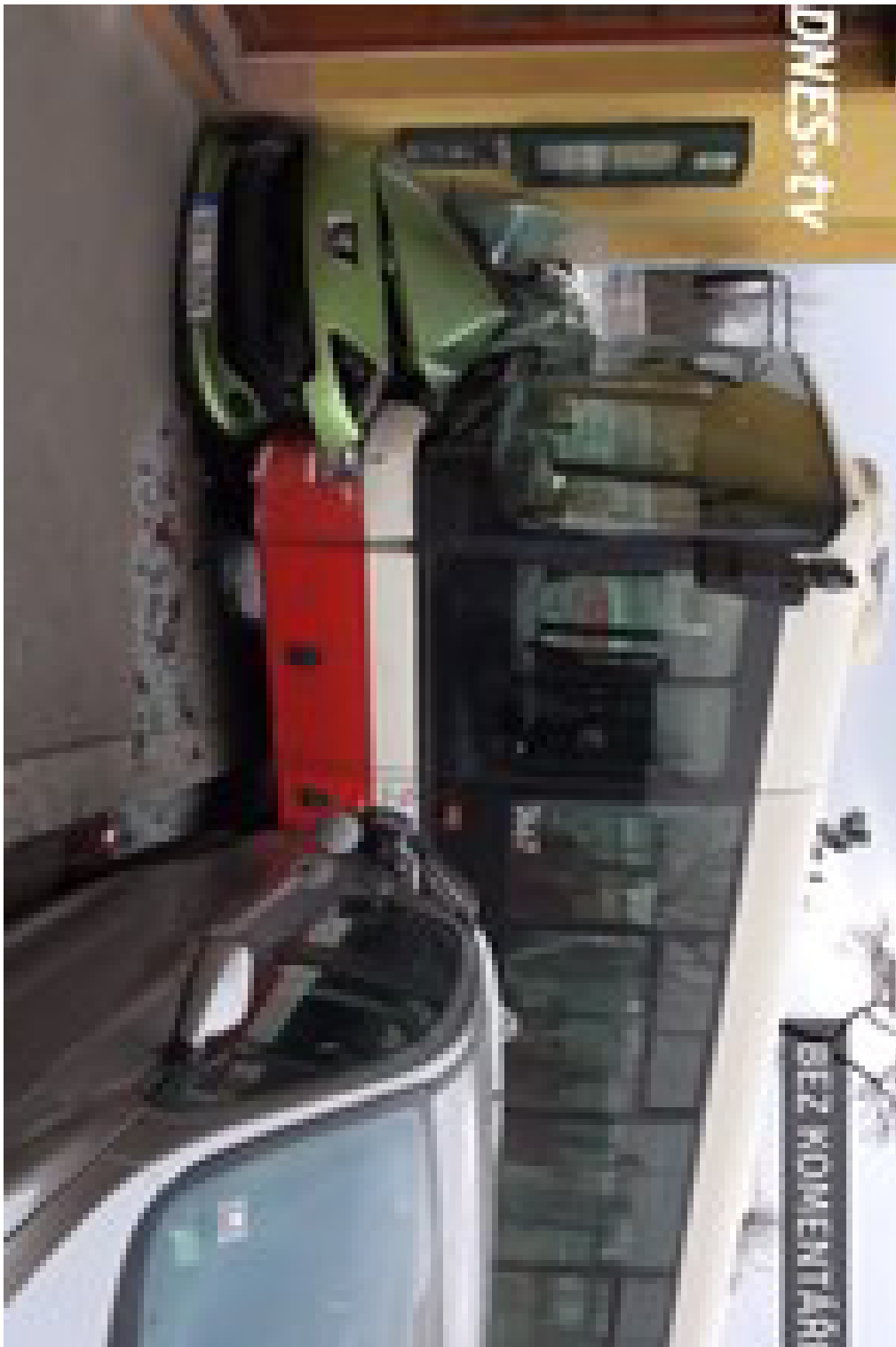
Sau tất cả, 2 công ty đại diện Song Song chính thức lên tiếng về tin đồn ly hôn và cương quyết ...

[Vegan food](#)

In the past, the market's waited for Apple to popularize new product categories, as it did with the iPad for tablets, but now the rest of the tech industry is forging ahead. Patience is low, enthusiasm is high, and, judging by the Huawei Mate X unveiled yesterday, product design and development are already at an advanced stage.

데뷔 40년 정태춘 10년간 음악 놓은 이유

高須クリニック・高須克弥院長がウーマンラッシュアワー・村本大輔に『最後通告



Snap up all the ingredients you need to make the perfect pancakes for free with TopCashback.co.uk. Sign up and get £5 worth of pancake ingredients from any store completely free. Just take a picture of your receipt and add it to the Top-Cashback app before Sunday 10 March.

ICSI

Haberturk.com ekibi olarak Türkiye’de ve dünyada yaşanan ve haber değeri taşıyan her türlü gelişmeyi sizlere en hızlı, en objektif ve en doyurucu şekilde ulaştırmak için çalışıyoruz. Yoğun gündem içerisinde sunduğumuz haberlerimizle ve olaylarla ilgili eleştiri, görüş, yorumlarınız bizler için çok önemli. Fakat karşılıklı saygı ve yasalara uygunluk çerçevesinde oluşturduğumuz yorum platformlarında daha sağlıklı bir tartışma ortamını temin etmek amacıyla ortaya koyduğumuz bazı yorum ve moderasyon kurallarımıza dikkatinizi çekmek istiyoruz.

NEXT ON CHANNEL

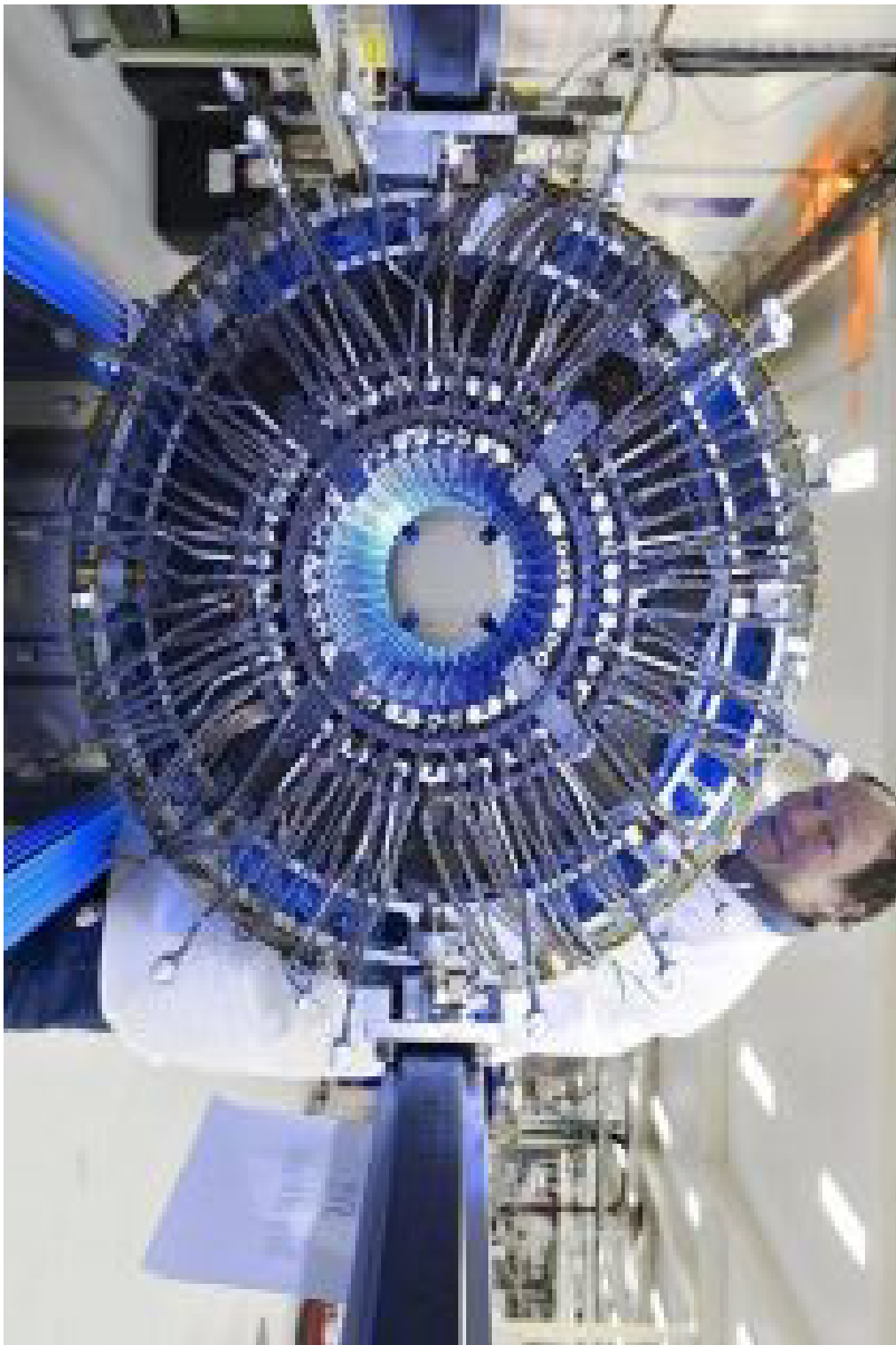
The Registry of Sarees is generating new interest in the garment through documentation, combining traditions

R 3Â 515Â 000

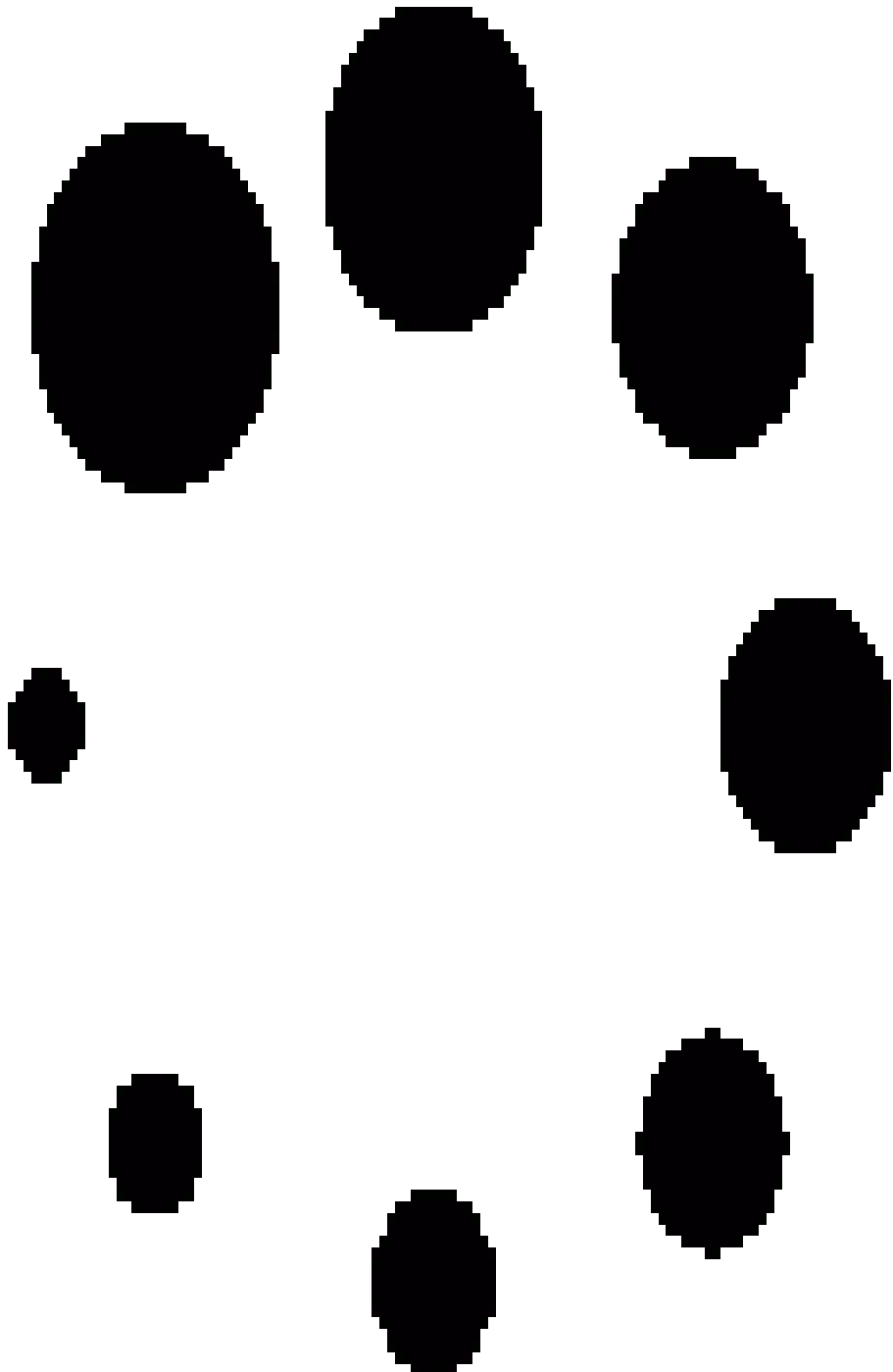
{{sender.name}}

ĐŃŃĐµĐ¼ ĐĐ¾ŃĐµĐ»ĐµĐ½Đ°Đ¾

Please wait....



Man stellt immer einen Anspruch an Information, man möchte sie verstehen, einen Gewinn oder eine Belohnung durch sie erzielen können. Auch wenn sie im Ungewissen und ohne sinnvolle Verknüpfung bleibt – besonders wenn man sich nicht darum bemühen muss – ist alles was davon bleibt und einen nicht aufhören lässt, das Gefühl sehr einfach Zufriedenheit erlangt zu haben.



www.reallysimplesatisfaction.com

„Wahrnehmung und Konsum von Information beeinflusst von Technologien des Internets“ ist der schriftliche Teil meiner Abschlussarbeit „Really Simple Satisfaction“ an der Kunsthochschule Kassel, im Studienschwerpunkt Neue Medien des Studiengangs Visuelle Kommunikation (Wintersemester 2018/ 2019), bei den Prüfern Prof. Joel Baumann und VProf. Hans Bernhard.

Danke

- @hansbernhard
- @joelbaumann
- @michael__alfred
- @simon_niedermeier
- @wolfgangschoeffel
- @__anna__hofmann
- <http://joernroeder.de/>
- @unfun_studio
- @kunsthochschulekassel

!