

TURING BUS

SAMMLUNG

www.turing-bus.de

OFFENER LEHRMATERIALIEN



TURING-BUS

Sammlung offener Lehrmaterialien

Dezember 2019

Das Projekt „Mit dem Turing-Bus auf den Spuren Künstlicher Intelligenz“ war ein Verbundprojekt der Gesellschaft für Informatik e.V. (GI) und der Open Knowledge Foundation Deutschland e.V. im Wissenschaftsjahr 2019 – Künstliche Intelligenz. Das Projekt wurde durch das Bundesministerium für Bildung und Forschung gefördert und durch das Weizenbaum Institut für die Vernetzte Gesellschaft unterstützt.

In dieser Veröffentlichung sind die im Rahmen der Turing-Challenge entstandenen Workshop-Beschreibungen und Arbeitsblätter zusammengefasst.

Autor*innen: Andrea Knaut, Bela Seeger, Kathinka Richter, Maximilian Voigt, Nikolas Becker, Stefan Ullrich, Frithjof Nagel, Constanze Kurz, Juliane Krüger, Rainer Rehak

2. aktualisierte Auflage Juli 2020



Unterstützt durch:
weizenbaum
institut

Inhaltsverzeichnis

1	ÜBER DEN TURING-BUS	1
2	ÜBERSICHT WORKSHOPS	1
3	WORKSHOP: WIR PROGRAMMIEREN EINEN MENSCHLICHEN ROBOTER.....	2
3.1	KOPIERVORLAGE ROBOTER-BEFEHLE	4
4	WORKSHOP: STATIONENLERNEN	5
4.1	ARBEITSBLATT INTELLIGENZTEST	6
5	WORKSHOP: DAS DIGITALE FLUGBLATT	7
5.1	HINTERGRUND.....	8
5.2	ZIELGRUPPE/VORKENNTNISSE.....	8
5.3	LERNZIELE.....	9
5.4	INHALTE	9
5.5	ABLAUF UND METHODEN	9
5.5.1	<i>Kernmodul zum Einstieg „Flugblattinhalte“ (15-30 min)</i>	<i>10</i>
5.5.2	<i>Kernmodul „Digitales Flugblatt“ (60-75 min)</i>	<i>12</i>
5.5.3	<i>Erweiterungsmodul „Netzwerkgrundlagen“ (20 min).....</i>	<i>14</i>
5.5.4	<i>Quellen und weiterführende Literatur</i>	<i>14</i>
5.6	DAS DIGITALE FLUGBLATT – INSTALLATIONSANLEITUNG	16
5.6.1	<i>Benötigtes Technikzubehör.....</i>	<i>16</i>
5.6.2	<i>Installationsanleitung</i>	<i>16</i>
5.6.3	<i>Windows – Installationsschritte zum Aufsetzen des Raspberry Pis.....</i>	<i>17</i>
5.6.4	<i>Abgekürzte Kommandozeilenbefehle für die weitere Netzwerkkonfiguration des Raspberry Pi als Hotspot aus dem Heise-Artikel.....</i>	<i>18</i>
5.6.5	<i>MacOSX – Installationsschritte zum Aufsetzen des Raspberry Pis</i>	<i>19</i>
5.7	MATERIALIEN	20
5.7.1	<i>Materialien für das Kernmodul „Digitales Flugblatt“</i>	<i>20</i>
5.7.1.1	<i>Präsentationsfolien für den Einstieg</i>	<i>20</i>
5.7.1.2	<i>Verschlüsselungstabelle.....</i>	<i>20</i>
5.7.1.3	<i>Arbeitsblatt Netzwerkcodes.....</i>	<i>21</i>
5.7.1.4	<i>Flugblattbeispiel.....</i>	<i>22</i>
5.7.1.5	<i>Arbeitsblatt: Das Digitale Flugblatt Variante (mit Windows Farbpalette).....</i>	<i>23</i>
5.7.2	<i>Materialien für das Einführungsmodul Netzwerk- begriffe</i>	<i>31</i>
5.7.2.1	<i>Filme für das Netzwerkgrundlagen-Erweiterungsmodul</i>	<i>31</i>
5.7.2.2	<i>Vorlage: Netzwerk-Grundbegriffe-Tafelbild.....</i>	<i>32</i>
6	WORKSHOP: WIE ENTSTEHT EIN DATENSATZ?	35
6.1	EIN BLICK HINTER DIE FASSADE SOGENANNTER KÜNSTLICHER INTELLIGENZ	35
6.1.1	<i>Im Fokus: Daten & Sensoren</i>	<i>35</i>
6.1.2	<i>Ablauf.....</i>	<i>37</i>
6.1.2.1	<i>Basis-Workshop: 3 Stunden</i>	<i>37</i>
6.1.2.2	<i>Vertiefung und Entwicklung des eigenen Zugangs: 3 Stunden</i>	<i>38</i>
6.2	LERNZIELE DES WORKSHOPS.....	42
6.3	TIPPS.....	42
6.4	WEITERFÜHRENDE LITERATUR	42
7	WORKSHOP: MEINE CHATBOTFREUNDIN.....	43
7.1	ABLAUF.....	43
7.2	MATERIALIEN	48
7.2.1	<i>Turing-Test ¹.....</i>	<i>48</i>
7.2.1.1	<i>Arbeitsblatt „Ach wie gut, dass niemand weiß, dass ich ein Computer bin“</i>	<i>48</i>

7.2.1.2	Arbeitsblatt Fragen und Antworten	52
7.2.2	<i>Chatbot</i>	53
7.2.2.1	Arbeitsblatt für die Gruppe "Chatbot: ELIZA"	53
7.2.2.2	Arbeitsblatt für die Gruppe "Chatbot: Scratch & Watson"	55
7.2.2.3	Arbeitsblatt für das "Chatbot-Rechercheteam"	63
7.2.2.4	Zusatzblatt Chatbot-Rechercheteam: Wer war Alan Turing?.....	65
7.2.2.5	Zusatzblatt Chatbot-Rechercheteam: Weizenbaum und Eliza	67
8	WORKSHOP: GEWISSENSBITS – ETHISCHE FALLBEISPIELE DER KÜNSTLICHEN INTELLIGENZ.....	69
8.1	ARBEITSBLATT 1: DIE ÜBLICHEN VERDÄCHTIGEN.....	70
8.2	ARBEITSBLATT 2: SMARTE ARMBÄNDER.....	72
8.3	ARBEITSBLATT 3: DAS SELBSTFAHRENDE AUTO	75
9	WORKSHOP: DAS ROBOTERGERICHT	77
10	IMPRESSUM	79

1 Über den Turing-Bus

Der Turing-Bus ist ein mobiles Bildungsangebot der Open Knowledge Foundation Deutschland e.V. und der Gesellschaft für Informatik e.V. (GI) im Wissenschaftsjahr 2019 – Künstliche Intelligenz.

Benannt nach Computerpionier Alan Turing, möchte der Turing-Bus im ländlichen Raum und in Gegenden jenseits der üblichen Metropolen Grundlagen der Informatik und gesellschaftliche Konsequenzen der Digitalisierung vermitteln und diskutieren. Die Zielgruppe des Projektes sind Jugendliche und junge Erwachsene im Alter zwischen 15 und 19 Jahren. Das Bundesministerium für Bildung und Forschung fördert den Turing-Bus im Rahmen des Wissenschaftsjahres 2019 – Künstliche Intelligenz.

Weitere Informationen unter www.turing-bus.de

2 Übersicht Workshops

	Dauer	Betreuungsschlüssel
Wir programmieren einen menschlichen Roboter	30 Minuten	1:50
Stationenlernen	30-60 Minuten	1:20
Das digitale Flugblatt	2-3 Stunden	1:10
Wie entsteht ein Datensatz?	2-3 Stunden	1:12
Meine Chatbotfreundin	2-3 Stunden	1:10
Gewissensbits	2 Stunden	1:20
Das Robotergericht	2 Stunden	1:20

Kopiervorlage

Seiten mit diesem Symbol sind zum Kopieren für den Unterricht gedacht!



3 Workshop: Wir programmieren einen menschlichen Roboter

Betreuungsschlüssel: 1:50

Dauer: 30 Minuten

Vorkenntnisse: keine

Verwendete Technik: Wasserflasche, Glas, Papier, Stift, Klebeband

Der folgende kurze Workshop bietet sich an, um die Teilnehmenden in den Begriff des Algorithmus einzuführen. Er eignet sich besonders als Einführung in einen längeren Workshop-Tag.

Der Workshop bringt uns den Algorithmus-Begriff nahe, indem ein menschlicher „Roboter“ vom Publikum „programmiert“ werden soll, um Wasser aus einer Flasche in ein Glas zu gießen. Dazu werden Befehle (hoch, links, runter, Hand drehen usw.) bereitgestellt. Die TN sollen dadurch spielhaft verstehen, dass Algorithmen exakte Befehle in einer definierten Reihenfolge ausführen.



Aufgabe: Programmiere den Roboter so, dass er ein Glas Wasser eingießt.

Es gibt eine Person, die den Roboter spielt. Am besten befestigt man ein Schild „ROBOTER“ an ihrer Stirn, so dass sie auch erstmal nichts sieht, sondern muss sich auf die Steuerung durch das Publikum verlassen. Diese Person sitzt an einem Tisch. Vor ihr steht eine offene, gefüllte Wasserflasche, daneben ein leeres Plastik-Glas. Sie lässt beide Arme hängen. Die Befehle, die der Roboter versteht, sind an die Wand projiziert oder werden auf andere Weise allen bekannt gegeben. Das Publikum soll nun die Befehle rufen, die den Roboter dazu bringen, das Glas zu befüllen. Im besten Fall klappt das, in den schlechtesten Fällen gießt der Roboter daneben, lässt die Flasche fallen oder haut das Glas um – daher Wasser und Plastik verwenden. Ist das Glas befüllt oder geht alles zu lange schief, wird abgebrochen.

Danach wird mit der ganzen Gruppe die Programmierung ausgewertet. Zum einen kann man erklären, wie die Befehle in einer Computer Programmierung üblicherweise heißen würden:

- „Hand“ oder „Arm“ stellen **Variablen** dar,
- „AUSGANGSPOSITION“, „oben“, „unten“, „rechts“, „links“ sind **konstante Werte** und
- 40 cm, -90 bis 90 Grad sind **Werte**, die den Variablen zugewiesen werden,

- „bewege“, „drehe“, „greife zu“, „lass los“ sind schließlich **Methoden**.

Zum anderen fehlen bestimmte Teile, die einer Vollautomatisierung im Wege stehen. Hier kann man das Publikum also befragen, was fehlt. Schlussfolgerungen sollten sein:

- **Sensoren**, die bestimmte Bedingungen erfassen können, die bis jetzt das Publikum durch das Hinsehen erfasst hat. Dafür kann man beispielsweise folgendes erfragen: Wie kann ein Roboter ohne Hilfe feststellen, dass das Glas voll ist?

3.1 Kopiervorlage Roboter-Befehle



Zutaten (Variablen):

- Glas = 0; (leer)
- Wasserflasche = 100; (voll)
- Armposition = 0,0; (Ursprungsposition)

Handlungsanweisungen (Befehle):

- Bewege nach: unten | oben | vorn | hinten
- Greife zu
- Lass los
- Drehe Hand

Kontrollstrukturen:

- Tue (Befehl) solange bis (Bedingung)
- Wenn (Bedingung), dann (Befehl), sonst (andere Befehl)

4 Workshop: Stationenlernen

Betreuungsschlüssel: 1:20

Dauer: 30-60 Minuten

Vorkenntnisse: keine

Verwendete Technik: Ein Laptop je zwei Teilnehmer*innen

In einem zweiten Einführungsmodul haben wir Beispiele für unterschiedliche KI- und Maschinenlernverfahren über vorher selektierte Webseiten auf Laptops bereitgestellt. Je nach Gruppengröße waren bis zu 8 dieser Stationen parallel verfügbar um im Wechsel von den TN ausprobiert zu werden (ca. 2 Min. / Station). Mit einem einfachen Arbeitsblatt vergeben die TN dabei Schulnoten um zu bewerten, für wie 'intelligent' sie die jeweiligen Tools halten. Die Ergebnisse werden im Anschluss auf einer auf dem Boden aufgeklebten Skala räumlich visualisiert. Anhand dieser Visualisierung werden die Meinungen diskutiert. Dadurch entsteht eine Auseinandersetzung mit dem Intelligenzbegriff, der eine der zentralen Herausforderungen im Verständnis des KI Begriffs darstellt.

Aufgabe für die Lernenden: Für wie intelligent hältst du folgendes System? Ordne es einer Zahl zwischen 1 (doof wie Brot) und 10 (schlau wie Mai Thi Nguyen-Kim) zu.

Nr.	Station	Benötigte Materialien	Max. TN	Material
1	Emotionstracking	1 Notebook pro 2 TN (integrierte Webcam, Lautsprecher)	10	noahlevenson.github.io/stealing-ur-feelings/tech-demo/
2	Robotik	Cozmo und Zubehör (Würfel, Ladestation, Ladegerät) Tablet / Smartphone	5	
3	Sprachassistenten	Alexa, Google etc.	5	
4	Intelligenz im Web	1 Notebook pro 2 TN (integrierte Webcam, Lautsprecher)	10	clickclickclick.click
5	Gaming	1 Notebook pro 2 TN (integrierte Webcam, Lautsprecher)	10	tenso.rs/demos/rock-paper-scissors/
6	Trainieren eines neuronalen Netzes	1 Notebook pro 2 TN (integrierte Webcam, Lautsprecher)	10	quickdraw.with-google.com
7	Kommunizieren mit Chat Bots	1 Notebook pro 2 TN (integrierte Webcam, Lautsprecher)	10	www.med-ai.com/models/sina.html.de

4.1 Arbeitsblatt Intelligenztest



Für wie intelligent hältst du folgende Dinge? Ordne die Stationen einer Zahl zwischen 1 (doof wie Brot) und 10 (schlau wie Mai Thi Nguyen-Kim) zu.

Station 1

Station 2

Station 3

Station 4

Station 5

Station 6

Station 7

5 Workshop: Das digitale Flugblatt

Autor*innen: Andrea Knaut, Kathinka Richter, Stefan Ullrich, Frithjof Nagel

Betreuungsschlüssel: 1:10

Dauer: 2 – 3 h

Vorbereitungsdauer: 2-8 Stunden (je nach Zahl der Teilnehmenden, eigenen Vorkenntnissen, vorhandener Rechnerinfrastruktur/Vorinstallationen)

Das digitale Flugblatt umfasst Anleitungen, Ablaufbeschreibung und Arbeitsblätter zur Vorbereitung und Durchführung eines Hands-On-Workshops. Hier könnt ihr lernen, wie ihr in 90 bis 180 Minuten einen Raspberry Pi Zero so konfiguriert, dass er als WLAN-Hotspot eine von euch gestaltete Webseite in Flugblattform in die nähere Umgebung sendet. Alle, die sich mit eurem Access Point verbinden, sehen sofort den Inhalt dieses digitalen Flugblatts.

Ihr lernt hierbei etwas über grundlegende Netzwerktechniken und Webdesign. Außerdem könnt ihr lernen, wie ihr andere auf euer WLAN aufmerksam machen könnt. Der Workshop ist vor allem inspiriert durch Daniel Coopers c't-Artikel „In die Freiheit entlassen“

(<https://www.heise.de/select/ct/2017/22/1508780300482172>).

Mit dem digitalen Flugblatt lassen sich Inhalte ganz verschiedener Themengebiete verbinden, die mit Bildern und einfachen Gestaltungselementen aufbereitet werden. Das Flugblatt lässt sich außerdem als demokratische Aktionsform einsetzen.

Einsatzbereiche:

- in der Schule in der Oberstufe im Informatik-Grundkurs oder -Wahlpflichtkurs, auch erweiterbar für den Leistungskurs,
- gut geeignet für fächerübergreifende Formate (Informatik + Politik/Philosophie/Deutsch/Geschichte/Kunst o.ä.),
- im außerschulischen Bereich z.B. als Halbtags-Aktionsworkshop, als Workshop in einem Freizeitprogramm, in Makerspaces, Jugendclubs, auf Camps

Themen:

- Flugblätter als Medium der politischen Meinungsäußerung und der Werbung
- (optional) Zusammenfassen von Argumenten in Kurztexten, Bildern und Rolle von Farben und Layout
- Netzwerktechnik (Webserver, IP-Adressen, Namenssysteme, WiFi, FTP, SSH, HTTP)
- Webseitengestaltung mit HTML und CSS
- universale Rolle von Werkzeugen wie Texteditoren, Kommandozeile
- (optional) IT-Sicherheit: Honeypots, Serverlogs, Rolle von Kryptographie, Tor Hidden Service

Anzahl Mentor*innen/Teilnehmer*innen (TN): bei geringen informatischen Vorkenntnissen der TN ist ein*e Lehrer*in für 6 bis 8 TN sinnvoll; bei guten informatischen Vorkenntnissen der meisten Teilnehmenden kann eine Person auch 16 oder mehr Personen betreuen.

Materialaufwand/räumliche Bedingungen/Kosten:

- Raspberry Pi Zeros und Zubehör (20,- bis 50,- Euro pro Raspberry Pi);
- PCs oder Laptops zur Konfiguration und Inbetriebnahme der Raspberry Pies müssen vorhanden sein (2 TN pro Gerät + ggf. Präsentations-PC/-Laptop);

- Akkuladung der Laptops sollte für 1-2 Stunden halten, sonst ausreichend Steckdosen/Verlängerungskabel und Verteilerstecker einplanen
- Beamer, Projektionsfläche, flexibel anordnungsbar Tische und Stühle, Whiteboard/Stifte oder Tafel/Kreide;
- (optional) Internetzugang, aber nicht zwingend
- (optional, bei Durchführung des Erweiterungsmoduls Kommunikationsguerilla): buntes Papier, Buntstifte, A2-Papierblätter, Eddings, Scheren, Post-Its in verschiedenen Farben, Sprühkreide u.Ä.
- siehe Materialien: Arbeitsblätter, Netzwerkkomponenten-Schema aus Papier, Entschlüsselungstabellen

Kurzbeschreibung:

Im Kernmodul des Workshops nehmen die Teilnehmenden einen günstigen Minirechner (Raspberry Pi Zero) als WLAN-Hotspot mit einer selbsterstellten Webseite in Betrieb: das digitale Flugblatt. Andere Teilnehmende können sich mit ihren Smartphones mit dem Hotspot verbinden und bekommen das Flugblatt sofort angezeigt.

5.1 Hintergrund

Der Workshop ist inspiriert durch den Artikel „In die Freiheit entlassen. Digitales Flugblatt: Raspberry Pi mit Batterie als anonymer WLAN-Hotspot und Webserver“ von Daniel Cooper in der c't 22/2017. Informatik-Studierende und -Interessierte auf dem DevCamp der GI in München 2018 haben die dort vorgeschlagene Konfiguration getestet und Vorschläge für eine verbesserte Umsetzung im Rahmen eines Workshops für Schülerinnen und Schüler gemacht. Der Workshop wurde auf den meisten Stationen der vom Bundesministerium für Bildung und Forschung geförderten Turing-Bus-Tour 2018 im Rahmen des Wissenschaftsjahres 2018 – Arbeitswelten der Zukunft erfolgreich durchgeführt und verbessert. Die Arbeitswelt der Zukunft ist eine vernetzte: die Basistechnik des Internets zu verstehen ist ein Schlüssel zur demokratischen Teilhabe in dieser Zukunft – erst recht, wenn Demokratie in Gefahr ist. Hier setzt unser Workshop an.

5.2 Zielgruppe/Vorkenntnisse

Der Workshop richtet sich an **Jugendliche ab der 8. Klasse mit geringen bis guten Informatik-Vorkenntnissen**. Geringe bis gute Informatikvorkenntnisse bedeutet, dass erste Erfahrungen mit dem Editieren von Quellcode-Dateien, der Benutzung von Kommandozeilen und Dateimanagern von Vorteil sind, aber nicht absolut notwendig.

Wenn die Teilnehmer*innen des Workshops nur selten Laptops oder Desktoprechner nutzen und **kaum Vorkenntnisse in Informatik** haben, ist allerdings ein erhöhter Unterstützungsaufwand nötig – Teams sollten entsprechend zusammengestellt werden.

Bei gutem Vorwissen in den Gebieten Netzwerktechnik und Webprogrammierung können sehr **Erfahrene** die Webseitengestaltung aufwändiger vornehmen oder Details der Netzwerkkonfiguration verbessern. Ebenfalls hilfreich sind Kenntnisse im Zusammenfassen, Argumentieren und der Mediengestaltung, da dann in der Regel das Finden und Zusammenstellen von Inhalten und der Einsatz geeigneter Bilder den Teilnehmer*innen leichter fällt. Auch für ältere Erwachsene ist der Workshop gut geeignet.

5.3 Lernziele

1. Die Workshop-Teilnehmerinnen und -Teilnehmer (TN) können zentrale Begriffe der Netzwerktechnik erkennen, kurz in ihrer Bedeutung für das Internet einordnen und erklären (z.B.: Webserver, DNS, Client, HTTP, URL, HTML, CSS u.Ä.).
2. Die TN können grundlegende HTML- und CSS-Elemente so verändern, dass sie die Gestaltung einer vorgegebenen Webseite anpassen.
3. Die TN erkennen, dass ein Webserver aus Programmen und Konfigurationsdateien besteht, die sie leicht editieren können, um die Betriebsweise (zum Beispiel den Hostname) nach ihren Bedürfnissen zu verändern.
4. Die TN gestalten eine Webseite zu einem interessanten schulrelevanten, gesellschaftlichen oder politischen Thema und wissen, wie sie diese in Betrieb nehmen können.

5.4 Inhalte

Im Kernmodul des Workshops nehmen die Teilnehmenden einen günstigen Minirechner (Raspberry Pi Zero) als WLAN-Hotspot mit einer selbsterstellten Webseite in Betrieb: das digitale Flugblatt. Andere Teilnehmende können sich mit ihren Smartphones mit dem Hotspot verbinden und bekommen das Flugblatt sofort angezeigt.

Es kann als Ausdrucksmittel kreativen digitalen Protests auf einer Versammlung, einer Demonstration oder als Teil einer Werbeaktion genutzt werden. Bei der Erstellung der Inhalte des Flugblatts bietet es sich an, entweder ein Thema eines anderen Schulfachs, aus der Lebenswelt der Teilnehmenden oder eine politische Diskussion aufzugreifen. Das Medium wird zudem selbst als technischer Baustein der allgegenwärtigen Netzwelt zentraler Gegenstand des Workshops: Grundlagen der Netzwerktechnik, Erstellung und Inbetriebnahme einer Webseite werden „hinter den Kulissen“, also auf Code-Ebene, erarbeitet. Universale etablierte Profi-Werkzeuge zur selbstbestimmten Kontrolle über Netzwerktechniken werden genutzt. So erschließen sich die Teilnehmenden zudem die Gestaltungspotentiale kleiner, billiger Bastelcomputer. Bei größeren Vorkenntnissen in Webprogrammierung und Netzwerkkonfiguration lassen sich eindrucksvollere Webseiten gestalten und verschiedene erweiterte Netzwerkkonfigurationen des Raspberry Pies umsetzen.

5.5 Ablauf und Methoden

Der Workshop besteht aus den Kernmodulen „Flugblattinhalte“ und „Digitales Flugblatt“, die etwa 90 Minuten dauern. Die Erweiterungsmodule „Kommunikationsguerilla“ (45 min) „Netzwerkgrundlagen“ (20 min), „Krypto-Warm-Up“ (15 min) und „Flyer-Ranking“ (10 min) tauchen als kurze Elemente teilweise bereits im Kernmodul auf. Es lohnt sich allerdings, sie deutlich länger zu gestalten. Insbesondere das Zusatzmodul „Kommunikationsguerilla“ stellt eine Anregung dar, wie der Workshop in fächerübergreifende und aktionistische Formate eingebettet werden kann.

5.5.1 Kernmodul zum Einstieg „Flugblattinhalte“ (15-30 min)

Sinn: Keine Webseite ohne Inhalte. Die schönsten Webprogrammier-Fähigkeiten nutzen nichts, wenn es nichts zu gestalten gibt. Ein Flugblatt ist ein politisches Medium, mit dem Menschen in öffentliche Debatten Forderungen in Kurzform und übersichtlicher Weise mit einprägsamen Bildern einzubringen suchen. Durch Gesprächsrunden, Schreiben und Zeichnen in Einzelarbeit auf dem Papier erarbeiten wir uns zunächst erste Inhalte für das digitale Flugblatt – die Auseinandersetzung mit der Technik soll hier noch nicht ablenken.

Bei fächerübergreifendem Unterricht kann dieses Modul verlängert und intensiviert oder durch eine Inhaltserarbeitung aus dem entsprechenden Nicht-Informatik-Unterrichtsfach/angrenzenden Workshops ersetzt werden. Das Zusatzmodul „Kommunikationsguerilla“ ist ebenfalls eine intensivere und erweiterte Variante dieses Moduls.

Zeit	Phase	Inhalt	Materialien	Sozialform/ Methode
0:05	Begrü- bung	Vorstellung der Infonaut*innen (I, das sind die Workshopleiter*innen) und des Kontextes	ggf. selbst zu erstel- lende Folien für die Prä- sentation der I und des Projekt Kontextes	
Es bietet sich bei ausreichend Zeit an, zur Auflockerung das Erweiterungsmodul „Krypto- Warm-Up“ bereits ganz am Anfang zu spielen.				
0:05- 0:15	Einstieg	Vorstellungsrunde der Teil- nehmenden (TN) mit Rück- fragen, z. B. „Kannst du dir unter dem Thema „Digita- les Flugblatt“ etwas vor- stellen? Wenn ja, was?“, „Wie und wo nutzt du freie WLANs oder Hotspots?“, nach der Vorstellung be- ginnt ein lockerer Aus- tausch zur Frage: „Welche Forderungen zur Verbesse- rung des Schulalltags/Le- bens in deinem Dorf/deiner Kleinstadt/Stadt würdest du auf einer Versammlung oder Demonstration ein- bringen?“	Tafel, Whiteboard	lockere Gesprächs- runde, in der alle TN einmal zu Wort kom- men und L Themen- stichworte an der Tafel notiert
falls mehr Zeit für den Einstieg vorhanden				
0:10- 0:25	Erarbei- tung	I wählt ein passendes Oberthema (Bsp.: Hatespe- ech, Schulalltag, Soziales Leben, Filme, Spiele, Um- weltpolitik, ...) und bittet	TN: Stift, 1- 2 leere Blätter L: Stoppuhr	Fokussprint in Stillar- beit: In 5 Minuten sollen die TN ihre Ideen zum Oberthema zu Papier

Zeit	Phase	Inhalt	Materialien	Sozialform/ Methode
		die TN um die ganz rechts erläuterte Schreibübung.		zu bringen als Text, als Bild oder in Stichpunkten – sie können einfach „drauflos schreiben“, ohne sich dabei zu zensieren; nach 5 Minuten bricht I das Schreiben ab; nun bekommen die TN eine Minute, um ihre Gedanken nochmal zu lesen und wichtiges hervorzuheben; dann sammelt I die Blätter ein, mischt sie und verteilt sie an andere TN, diese erhalten nun 3 Minuten, um das Blatt eines anderen TN zu lesen, zu kommentieren und darin hervorzuheben, im Anschluss legen alle die Blätter auf einem großen Tisch aus; jeder nimmt sich sein/ihr Ursprungsblatt und erhält 2 Minuten, um die Kommentare zu lesen und Kerngedanken zu formulieren.
0:25-0:30	Sicherung	I fragt nach, welche der Gedanken und Themen sich aus Sicht der TN für ein Flugblatt eignen könnten		Gesprächsrunde

5.5.2 Kernmodul „Digitales Flugblatt“ (60-75 min)

Sinn: Wie werden die Flugblatt-Inhalte nun zu einer Webseite? Was müssen wir konfigurieren, so dass sie mit einem selbst eingerichteten Hotspot in die Welt hinausgesendet werden kann und andere sie auf ihren Smartphones lesen können? Die Beantwortung dieser Frage erarbeiten sich die Teilnehmenden, indem sie es direkt in Teamarbeit und mit einiger Hilfestellung und einigen Vorgaben ausprobieren – sie nutzen hierbei eine klassische heuristische Form der Problemlösung kennen, die in der Softwareentwicklung und im dazu gehörigen Testen eine wichtige Rolle spielt: Versuch und Irrtum, Trial & Error. Die Teilnehmenden erkunden und nutzen universale Webentwicklungswerkzeuge, lernen Server und Internetzugangspunkte als IT-System von Software-Diensten, die Konfigurationsdateien verarbeiten, kennen. Diese müssen nicht allein auf fernen Serverfarmen in abgeschotteten Hochsicherheitstrakten auf Spezialhardware laufen, sondern können auch auf jedem billigen Einplatinenrechner von jedem Menschen einfach aufgesetzt, konfiguriert und in Betrieb genommen werden. Das Internet als dezentrales Netz von Peers, die selbst zu Zulieferern von Inhalten werden, wird so ganz unmittelbar greifbar.

Vorbereitung: Dieses Modul bedarf einer umfangreichen Vorbereitung, die je nach Erfahrung im Umgang mit Rechnern und Anzahl der genutzten Rechner zwei bis acht Stunden dauern kann. Auf den genutzten PCs/Laptops und den Raspberry Pi Zeros müssen im Vorfeld Programme bzw. das System installiert und noch einige vorbereitende Konfigurationen vorgenommen werden. Die Anleitung hierfür findet sich in „Vorbereitung und Installation“.

Zeit	Phase	Inhalt	Materialien	Sozialform
0:10	Einstieg	Vorstellung des Moduls und des Raspberry Pies	Präsentation (siehe Power-Point-Folien)	frontaler Vortrag
0:10-0:15	Einstiegsvariante 2	Vorbereitete Netzwerkbe-griiffe oder Informatiker müssen mithilfe einer Code-tabelle entschlüsselt wer-den	Codetabelle (siehe Materia-lien)	Schnitzeljagd
Bei ausreichend Zeit und/oder geringen Vorkenntnissen der TN im Bereich Netzwerktech-nik kann an dieser Stelle das Erweiterungsmodul „Netzwerkgrundlagen“ eingefügt wer-den. Es erweitert den in der nächsten Zeile folgenden Abschnitt auf 20 Minuten.				
0:10-0:15	Einführung In-ternet	Es werden zwei Erklärvideos gezeigt. Darin wird der Auf-bau und die Funktionsweise des Internets erklärt. Da-nach sollen die Schüler aus ausgedruckten Netzwerkbe-griffen, Bildern und Verbin-dungspfeilen ein Schaubild legen (oder ein Tafelbild mit Kreide erstellen). Damit werden zuvor vermittelte In-halte aus den Videos abge-fragt, optisch in Zusammen-hang gebracht und offene Fragen und Verständnis-	Videos, Com-puter, und Beamer, inter-aktive Tafel o-der A3 Ausdru-cke für Schaubild In-terne/ Netz-werkbegriffe (siehe Materia-lien „Schaubil-der Internet“ und PDF Doku-ment „A3-Druckvorlage-	L-TN-Gespräch

Zeit	Phase	Inhalt	Materialien	Sozialform
		probleme können gemeinsam in der Gruppe geklärt werden. Dies ermöglicht eine Wiederholung und Vertiefung des Lernstoffes.	Schaubild-Internet.pdf")	
Bei ausreichend Zeit und/oder geringen Vorkenntnissen der TN im Bereich Webprogrammierung sollte der erste Teil, Gestalten und Programmieren, sowie der letzte Teil (Konfigurieren, Testen, Anpassen) ausgedehnt werden. Schnelle TN können hier Unerfahrenen helfen, finden Zusatzaufgaben auf dem Arbeitsblatt oder können das Flugblatt ausgefallener gestalten.				
0:15-0:35 (0:15-0:40)	Gestalten/ Programmieren	Das Flugblatt gestalten.	pro 2er-Team: vorkonfigurierter Laptop oder PC, vorkonfigurierter Raspberry Pi Zero (1 pro 2 Personen, inkl. USB-Kabel und Powerbank) Arbeitsblatt	Teamarbeit
0:35-0:40 (0:40-0:50)	Inbetriebnahme	Das Flugblatt in Betrieb nehmen.	Turing-Bus, Bilder, Farbtabelle (Arbeitsblätter siehe „Materialien“, Vorinstallation der Rechner und Raspberry Pies siehe „Vorbereitung und Installation“)	Teamarbeit
0:40-0:50 (0:50-0:60)	Konfigurieren, Testen, Anpassen	Dem Flugblatt einen Namen geben.	Arbeitsblatt Turing-Bus, Bilder, Farbtabelle (Arbeitsblätter siehe „Materialien“, Vorinstallation der Rechner und Raspberry Pies siehe „Vorbereitung und Installation“)	Teamarbeit
Alternativ zum letzten Teil können die Erweiterungsmodule „Flyer-Ranking“ und/oder „Krypto-Modul“ durchgeführt werden.				
0:50-0:60 (0:60-0:75)	Auswertung und Sicherung	Die erstellten Flugblätter werden von den jeweiligen Teams kurz für alle anderen präsentiert. Hierzu geht L mit dem an den Beamer angeschlossenen Präsentationsgerät in die jeweiligen Hotspots und lädt die entsprechenden Seiten, so dass alle sie sehen. Die TN stellen diese kurz den anderen vor. Applaus, Anerkennung/Kritik, ggf. kurze Besprechung der Rückmeldungen. Im Anschluss der	Beamer, Präsentations-PC/Laptop	Gesprächsrunde

Zeit	Phase	Inhalt	Materialien	Sozialform
		Präsentationen kann bei verbleibender Zeit über Anwendungsmöglichkeiten des digitalen Flugblatts im öffentlichen Raum und mögliche Gefahren gesprochen werden.		

5.5.3 Erweiterungsmodul „Netzwerkgrundlagen“ (20 min)

Sinn: Diese ganzen Abkürzungen, IP, HTML, ISP, URL, WWW, DNS – was soll das alles? Sie stehen für sogenannte Protokolle und Sprachen, die das Internet konstituieren, und es gibt noch viel mehr. Außerdem sind da Server, Clients, Hosts und so vieles mehr. In diesem Modul wird dieser Buchstabensalat in eine Ordnung gebracht und erläutert, wie all diese Komponenten zusammenspielen, damit das Internet funktioniert.

Vorbereitung: Wird dieses Modul erstmalig durchgeführt, müssen im Vorfeld noch die einzelnen Piktogramme und Bildchen für das Netzwerk-Schaubild, das die TN zusammenlegen, auf A3 ausgedruckt und ausgeschnitten werden.

Zeit	Phase	Inhalt	Materialien	Sozialform/Methoden
0:10	Einstieg	In kurzen Erklärvideos wird der grundlegende Aufbau und die Funktionsweise des Internets erläutert und die grundlegenden Netzbegriffe erklärt.	Filme (siehe Materialien), ggf. Internetzugang erforderlich zum direkten Abspielen bei YouTube	frontale Vorführung, TN machen sich Notizen
0:10-0:20	Erarbeitung	Die Schüler sollen aus Bildern, Verbindungen und Netzbegriffen ein Netzwerkschaubild entwerfen. Dabei werden die zuvor eingeführten Begriffe, aus den Videos, werden wiederholt und in eine bildhafte Verbindung gebracht.	Ausgedruckte Netzwerkkomponenten (siehe Materialien „Netzwerk-Grundbegriffe-Schaubild“) oder Kreide um ein eigenes Tafelbild zu erstellen	L-TN-Gespräch

5.5.4 Quellen und weiterführende Literatur

Cooper, Daniel: In die Freiheit entlassen. Digitales Flugblatt: Raspberry Pi mit Batterie als anonymes WLAN-Hotspot und Webserver. In: c't 22/2017. URL: <https://www.heise.de/select/ct/2017/22/1508780300482172>

Eine andere Variante der Realisierung des Flugblatts bietet die Online-Webseitenerstellung mit WordPress. In diesem Material ist die Vorgehensweise erläutert:

Raspberry Pi Foundation: Build a LAMP Web Server with WordPress.
<https://projects.raspberrypi.org/en/projects/lamp-web-server-with-wordpress>

Die genutzte Software ist in der Installationsanleitung verlinkt.

5.6 Das digitale Flugblatt – Installationsanleitung

5.6.1 Benötigtes Technikzubehör

1. Raspberry Pi Zero WH (hat eine bereits angelötete GPIO-Stiftleiste; praktisch, falls das Gerät auch für andere Projekte bspw. mit Stekkabeln genutzt werden soll), oder Raspberry Pi Zero W (ohne angelötete GPIO-Stiftleiste), ca. 16,00 Euro
2. Powerbank mit microUSB-Kabel zum transportablen Betrieb der Raspberry Pies (z.B. 10000mAh), ca. 15,00 Euro
3. MicroSD-Karte mind. 8 GB, ca. 4,00 Euro
4. (optional, empfiehlt sich aber, wenn man den Pi nicht völlig "blind" aufsetzen will) Micro-USB-Datenkabel, ca. 7,00 Euro
5. (optional) Mini-HDMI-zu-HDMI-Adapter für Anschluss Raspi an Monitor, 1,00 Euro
6. (optional) micro-B-USB-zu-2-Port-USB-A-female-Kabel für Anschluss Raspi an Tastatur und/oder Maus, 1,- bis 6,- Euro (je nach Port-Anzahl)
7. (optional) Gehäuse für den Raspberry Pi, ca. 6,00 Euro

Hinweis: Artikel 1. + 5.-7. gibt es bei Elektronikversandhäusern wie watterott, Pimoroni, buyzero, ThePiHut, ModMyPi oft in Kombi-Kits.

5.6.2 Installationsanleitung

Die folgenden Schrittfolgen basieren auf den Empfehlungen folgender Artikel und beziehen sich zum einen auf Windows 10 und zum anderen auf MacOSX – insbesondere 1. bezieht sich auch auf Linux.

1. <https://www.heise.de/select/ct/2017/22/1508780300482172>
2. <http://www.circuitbasics.com/raspberry-pi-zero-ethernet-gadget/>
3. <https://www.raspberrypi.org/forums/viewtopic.php?t=171109>
4. <https://www.factoryforward.com/pi-zero-w-headless-setup-windows10-rndis-driver-issue-resolved>

Erforderliche Software zum Installieren und Bespielen des Raspberry Pi

- Raspbian (aktuelles Debian-Linux-Betriebssystem-Image für den Raspberry Pi)
https://downloads.raspberrypi.org/raspbian_latest
- Win32 Disk Imager Software (zum Aufspielen des Image auf die SD-Karte des Raspberry Pi)
<https://sourceforge.net/projects/win32diskimager/>
(auf dem Mac: Etcher <https://www.balena.io/etcher/> oder auf der Kommandozeile dd)
- für Windows 10: RNDIS-Treiber (damit Windows 10 den per USB-Kabel angeschlossenen Pi erkennt)
<http://web1.moddevices.com/shared/mod-duo-rndis.zip>
- für Windows: Bonjour-Service von Apple (falls nicht schon iTunes installiert ist; ist auch für die Erkennung des Raspberry Pi am USB-Kabel nötig)
https://support.apple.com/kb/DL999?locale=en_US
- für Windows: SSH-Client PuTTY (zur Konfiguration des Raspberry Pi via Kommandozeile)
<https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html>
- FileZilla (FTP-Client zum Aufspielen der Webseite auf den Raspberry Pi)
<https://filezilla-project.org/download.php>

- Notepad++ (zum Editieren von Konfigurationsdateien vom Raspberry Pi (beim Mac geht zum Beispiel Textwrangler: <https://textwrangler.de.upto-down.com/mac>) <https://notepad-plus-plus.org/download/>)
- (optional) VNCViewer (zum Bedienen des Pi via GUI-Desktop) <https://www.realvnc.com/de/connect/download/viewer/>

5.6.3 Windows – Installationsschritte zum Aufsetzen des Raspberry Pies

1. Formatiere die SD-Karte über Datenträgerverwaltung von Windows.
2. Schreibe das heruntergeladene aktuelle Raspbian-Stretch-Image mit Win32 Disk Imager auf die Mikro-SD-Karte des Raspberry Pi.
3. Gehe nach dem Brennen mit dem Explorer in das Dateisystem der SD-Karte und öffne folgende Dateien im Editor notepad++: **config.txt** und **cmdline.txt**
--> in der Datei config.txt füge diese Zeile als neue Zeile ans Ende der Datei:
`dtoverlay=dwc2`
4. --> in der Datei cmdline.txt füge folgendes Kommando ein, das lediglich durch ein Leerzeichen getrennt hinter dem Kommando `rootwait` eingefügt wird:
`modules-load=dwc2,g_ether`
Alle hiernach möglicherweise noch folgenden Kommandos folgen nach nur einem Leerzeichen hinter diesem neu eingefügten.
5. mit notepad++ die Datei `wpa_supplicant.conf` mit korrekten SSID-/PWD-/KEY-Daten für ein funktionierendes WLAN auf dem Pi erstellen – Beispielinhalt ab nächster Zeile kopieren:
`ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev`
`update_config=1`
`country=DE`

`network={`
 `ssid="the_ssid_of_your_wlan"`
 `psk="the_wpa2_key_of_your_wlan"`
 `key_mgmt=WPA-PSK`
`}`
6. SSH in Raspbian anschalten, indem man ein leere Datei mit dem Namen `ssh` und ohne Dateinamenerweiterung ebenfalls in den BOOT-Ordner des Dateisystems der SD-Karte kopiert.
7. SD-Karte auswerfen, in den Raspberry Pi stecken und das USB-Kabel zum Daten-USB-Port des Pies verbinden (das ist der innen liegende, nicht der äußere PWR-USB-Port)
8. Im besten Falle installiert sich der RNDIS-Treiber beim Anschluss des Raspberry Pi von selbst als USB Serial Device (im Gerätemanager überprüfbar). In Windows 10 installiert sich der RNDIS-Treiber als COM-Port. Um Ethernet over USB nutzen zu können, muss der RNDIS-Treiber per Hand installiert werden. BITTE HIER DIE LINKS IN DER ANLEITUNG BEACHTEN. <https://www.factoryforward.com/pi-zero-w-headless-setup-windows10-rndis-driver-issue-resolved>
9. Öffne Putty. In SSH connect schreibe `raspberrypi.local`. Klicke Open. Eine SSH-Verbindung sollte jetzt aufgebaut werden (beim ersten Mal muss noch der Key akzeptiert werden.)
10. Login mit Default-Username: `pi`
Default-Passwort: `raspberrypi`

11. Eingabe der folgenden Befehle am Kommando-Prompt:

```
pi@raspberrypi:~$ sudo raspi-config
```

--> Interfacing option, enable VNC

--> change user password to something unique (turingbus)

Nun kann der Raspberry Pi ggf. auch via VNCViewer und Desktop weiter konfiguriert/erreicht werden, wenn der vncserver gestartet wird.

5.6.4 Abgekürzte Kommandozeilenbefehle für die weitere Netzwerkkonfiguration des Raspberry Pi als Hotspot aus dem Heise-Artikel

```
pi@raspberrypi:~$ sudo apt-get install hostapd dnsmasq
```

Webserver

```
pi@raspberrypi:~$ sudo apt-get install nginx
```

SITE liegt unter /var/www/html (am besten mit FileZilla befüllen)

Konfigurationsdateien von heise laden/per Filezilla

```
pi@raspberrypi:~$ wget ftp://ftp.heise.de/pub/ct/listings/1722-144.zip
```

(stattdessen kann sie auch per FileZilla in den pi-Home-Ordner kopieren)

```
pi@raspberrypi:~$ unzip 1722-144.zip
```

```
pi@raspberrypi:~$ sudo cp 1722-144/default/* /etc/default
```

```
pi@raspberrypi:~$ sudo cp -a 1722-144/hostapd /etc
```

```
pi@raspberrypi:~$ sudo cp 1722-144/dnsmasq.conf /etc
```

```
pi@raspberrypi:~$ cd 1722-144
```

```
pi@raspberrypi:~$ sudo cp fakedns/*.py /usr/local/bin
```

```
pi@raspberrypi:~$ sudo cp fakedns/*.service /etc/systemd/system
```

```
pi@raspberrypi:~$ sudo systemctl enable fakedns.service
```

#Umleitungen bei HTTP-Errorcodes

```
pi@raspberrypi:~$ sudo pico /etc/nginx/sites-available/default
```

in diese Datei vor "Location" die Zeile einfügen:

```
error_page 404 =302 http://192.168.255.1/;
```

```
pi@raspberrypi:~$ service nginx restart
```

#use DNS-Masq als DNS-Umleiter

```
pi@raspberrypi:~$ sudo pico /etc/dnsmasq.conf
```

hier einfügen:

```
address=/#/192.168.255.1
```

Netzwerk

```
pi@raspberrypi:~$ sudo pico /etc/network/interfaces
```

hier einfügen:

```
auto wlan0
```

```
iface wlan0 inet static
```

```
address 192.168.255.1
```

```
netmask 255.255.255.0
```

```
wireless-mode Master
```

```
wireless-power off
```

```
auto usb0
iface usb0 inet static
address 192.168.3.11
netmask 255.255.255.0
```

--> Netzwerk-Interfaces neu starten (Achtung, immer Gefahr der Disconnection)

```
pi@raspberrypi:~$ ifup wlan0
pi@raspberrypi:~$ sudo /etc/init.d/networking restart
ODER
pi@raspberrypi:~$ sudo systemctl restart networking
```

Umbenennen des Hotspots (siehe Arbeitsblatt):

```
pi@raspberrypi:~$ sudo service hostapd restart
```

5.6.5 MacOSX – Installationsschritte zum Aufsetzen des Raspberry Pies

1. Formatiere die SD-Karte über das Festplattendienstprogramm von MacOSX mit FAT32
2. Schreibe das heruntergeladene aktuelle Raspbian-Stretch-Image mit Etcher auf die Mikro-SD-Karte des Raspberry Pi.
3. Gehe danach mit dem Finder in das Dateisystem der SD-Karte und öffne folgende Dateien im Editor Textwrangler: config.txt und cmdline.txt
--> in der Datei config.txt füge diese Zeile als neue Zeile ans Ende der Datei:
dtoverlay=dwc2
4. --> in der Datei cmdline.txt füge folgendes Kommando ein, das lediglich durch ein Leerzeichen getrennt hinter dem Kommando rootwait eingefügt wird:
modules-load=dwc2,g_ether
Alle hiernach möglicherweise noch folgenden Kommandos folgen nach nur einem Leerzeichen hinter diesem neu eingefügten.
5. mit Textwrangler die Datei wpa_supplicant.conf mit korrekten SSID-/PWD-/KEY-Daten für ein funktionierendes WLAN auf dem Pi erstellen – Beispielinhalt ab nächster Zeile kopieren:
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
update_config=1
country=DE

network={
 ssid="the_ssid_of_your_wlan"
 psk="the_wpa2_key_of_your_wlan"
 key_mgmt=WPA-PSK
}
6. SSH in Raspbian anschalten, indem man ein leere Datei mit dem Namen ssh und ohne Dateinamenerweiterung ebenfalls in den BOOT-Ordner des Dateisystems der SD-Karte kopiert.
7. SD-Karte auswerfen, in den Raspberry Pi stecken und das USB-Kabel zum Daten-USB-Port des Pies verbinden (das ist der innen liegende, nicht der äußere PWR-USB-Port)
8. Öffne das Programm Terminal und verbinde dich via ssh mit dem Raspberry Pi. Hierzu schreibe folgendes auf die Kommandozeile:

```
ssh pi@raspberrypi.localraspberrypi.local
```

Beim ersten Mal wird noch abgefragt, ob du dich wirklich verbinden willst, weil der Key des neuen Hosts noch unbekannt ist. Hier `yes` eingeben.

- Bei der Passwortabfrage

```
pi@raspberrypi.local's password:
```

 ist das Default-Passwort: `raspberrypi`

Eingabe der folgenden Befehle am Kommando-Prompt:

```
pi@raspberrypi:~$ sudo raspi-config
```

--> Interfacing option, enable VNC

--> change user password to something unique (turingbus)

Nun kann der Raspberry Pi ggf. auch via VNCViewer und Desktop weiter konfiguriert/erreicht werden, wenn der vncserver gestartet wird.

5.7 Materialien

5.7.1 Materialien für das Kernmodul „Digitales Flugblatt“

5.7.1.1 Präsentationsfolien für den Einstieg

Im Ordner Materialien finden Sie eine PowerPoint Präsentation mit Folien zu folgenden Inhalten:

- „Wie funktioniert das Abrufen einer Webseite?“ und Funktionsweise „DNS System“ (2 Varianten: V1 alles zusammen, V2 getrennte Erklärung DNS)
- Die Schritte zum Workshop „Digitales Flugblatt“

5.7.1.2 Verschlüsselungstabelle

Anhand dieser Tabelle können Lehrkräfte die gewünschten verschlüsselten Wörter auswählen und bspw. in Form einer Schnitzeljagd, oder aber für die Flugblattseite für ihre Schüler*innen vorbereiten.

	BEGRIFF	VERSCHLÜSSELUNG
1	IP-Adresse	963263-187282397833352352833
2	Netzwerkprotokoll	534833911219319833397159263397514911514159514376376
3	HTML-Dokument	160911661376-282514159399661833534911
4	Internet Protokoll	963534911833397534833911 263397514911514159514376376
5	Binaercode	899963534187833397599514282833
6	Server Provider	352833397329833397 263397514329963282833397
7	Client Browser	599376963833534911 899397514319352833397
8	World Wide Web	319514397376282 319963282833 319833899
9	Glasfaser	453376187352640187352833397

10	Uniform source Locator	Re- 399534963640514397661 376514599187911514397	397833352514399397599833
11	Hypertext	160444263833397911833958911	
12	Router	397514399911833397	
13	E-Mail	833-661187963376	

5.7.1.3 Arbeitsblatt Netzwerkcodes

- Sucht den WLAN-Hotspot.
- Verbindet euch mit dem Hotspot und schaut euch das hinterlegte Digitale Flugblatt an.
- Im Text versteckt sich ein verschlüsselter Begriff.
- Nutzt die folgende Code-Tabelle, um die Verschlüsselung zu knacken!



BUCHSTABE	CODE
A	187
B	899
C	599
D	282
E	833
F	640
G	453
H	160
I	963
J	507
K	159
L	376
M	661
N	534
O	514
P	263
Q	146
R	397
S	352
T	911
U	399

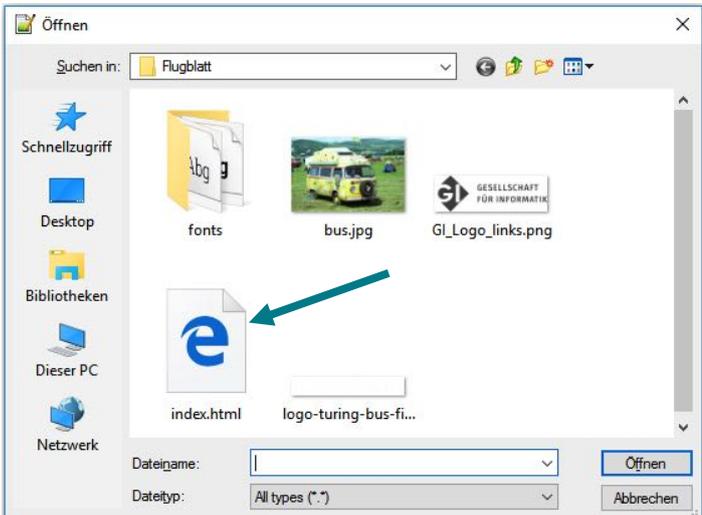
V	329
W	319
X	958
Y	444
Z	219

5.7.1.4 Flugblattbeispiel

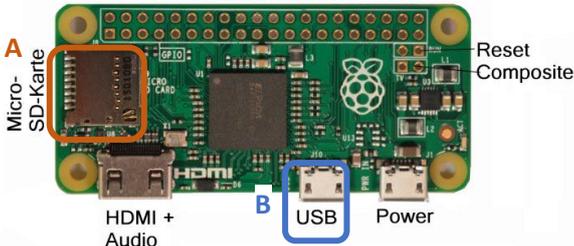
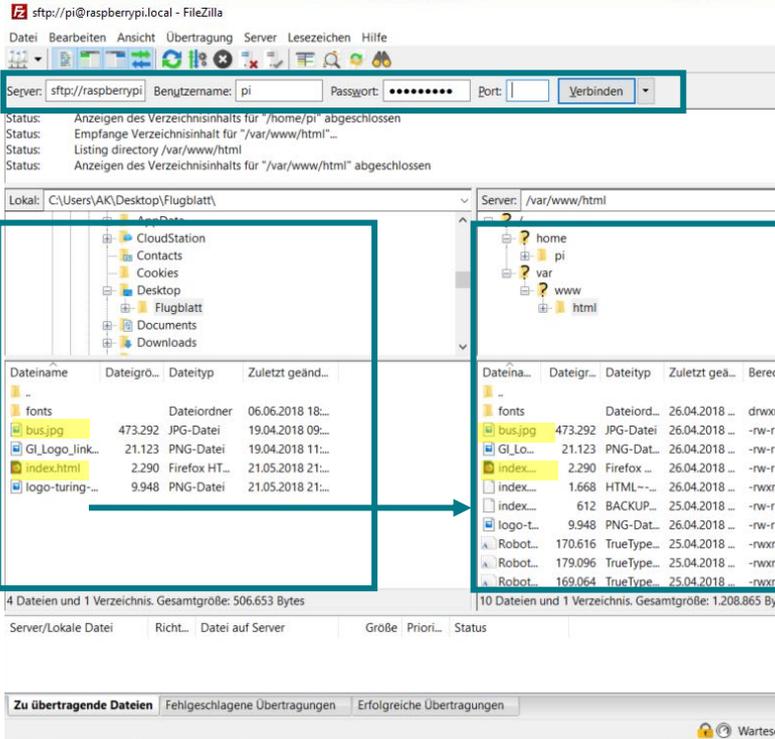


5.7.1.5 Arbeitsblatt: Das Digitale Flugblatt Variante (mit Windows Farbpalette)



TEIL 1 – Die HTML-Seite des Flugblatts neu gestalten.		
1.	Laptop hochfahren und anmelden	
2.	Zur Bearbeitung des Flugblatts brauchst du einen Editor, zum Beispiel Notepad++ . Start durch Doppelklick auf das Icon auf dem Desktop:	
3.	Notepad++ öffnet sich mit einer leeren Datei. Öffne mit Hilfe der Tastenkombination Strg-O die Datei index.html , die im Ordner Flugblatt auf dem Desktop des Computers liegt.	
Aufgabe 1 – Ändern des Texts auf dem Flugblatt		
4.	Finde in der Datei index.html die Stellen, an denen der im Browser sichtbare Text steht.	Zum Beispiel zwischen den HTML-Tags <code><p class="zitat"></code> und <code></p></code> oder <code><p class="text"></code> und <code></p></code>
5.	Ersetze diesen Text mit deinem eigenen. Achte dabei darauf, dass du die HTML-Tags nicht löschst.	
6.	Speichere die fertig bearbeitete Datei mit Strg-S ab. Überprüfe das Ergebnis, indem du durch Doppelklick auf die eben gespeicherte index.html im Ordner Flugblatt auf dem Desktop die Datei in einem Browser aufrufst. Du siehst nun, dass das Flugblatt jetzt deinen Text enthält.	

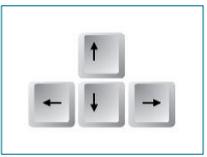
Aufgabe 2 – Ändern des Hintergrundbilds	
7.	Finde in der Datei index.html das Hintergrundbild des digitalen Flugblatts. Das Bild ist im CSS-Code der Datei in der Klasse .background hinterlegt. In welcher Zeile steht es?
8.	<div style="display: flex; border-bottom: 1px solid black;"> <div style="flex: 1;"> <p>Ersetze das Bild mit einem anderen Bild, das zu deinem Text aus Aufgabe 1 passt.</p> <p>Suche hierfür im Ordner Bilder auf dem Desktop nach einem geeigneten Hintergrundbild. Kopiere dieses in den Ordner Flugblatt (Desktop) und trage den Dateinamen an Stelle des Namens der in Schritt 7 gefundenen Bilddatei ein.</p> <p>Achte darauf, dass du die einfachen Anführungszeichen dabei nicht löschst und die Dateiendung (jpg/png) übereinstimmt.</p> <p>Speichere wie oben ab und überprüfe die Anpassung wieder im Browser. Dort musst du zur Aktualisierung der Seite nur Strg-R drücken.</p> </div> <div style="flex: 1; border-left: 1px solid black; padding-left: 10px;"> <pre>.background { background:url('DateinameDeinesBilder.Dateiendung') no-repeat center center fixed; -webkit-background-size: cover; -moz-background-size: cover; -o-background-size: cover; background-size: cover; hyphens:auto; }</pre> </div> </div>
Aufgabe 3 – Ändern der Farbe des Texthintergrunds	
9.	<div style="display: flex; border-bottom: 1px solid black;"> <div style="flex: 1;"> <p>Finde in der Datei index.html die CSS-Klasse .layer und dort die Eigenschaft background-color.</p> <p>Die Farbe ist als RGB-Code hinterlegt. Der erste Wert steht für ROT, der zweite für GRÜN, der dritte für BLAU. Der vierte Wert ist eine Prozentangabe für den Grad der Durchsichtigkeit. Im Beispiel rechts steht 0.8 für 80%.</p> </div> <div style="flex: 1; border-left: 1px solid black; padding-left: 10px;"> <pre>.layer { background-color: rgba(250, 142, 14, 0.8); margin: 5%; padding: 10%; top: 0; left: 0; }</pre> </div> </div>
10.	Benutze die Tabelle Farbtabelle_RGB-HEX.pdf auf dem Desktop , um eine neue Farbe festzulegen. Verändere den Grad der Durchsichtigkeit.
11.	Speichere (mit Strg-S) wie oben ab und überprüfe die Anpassung wieder im Browser. Dort musst du zur Aktualisierung der Seite nur Strg-R drücken.

TEIL 2 – Das neue Flugblatt in Betrieb nehmen.	
Aufgabe 4 – Hochladen des angepassten Flugblatts auf den Raspberry Pi Zero.	
<p>12. A Nimm die SD-Karte aus dem Adapter und stecke sie in den SD-Karten-Slot des Raspberry Pi's.</p> <p>Aufbau des Raspberry Pi Zero:</p>	
<p>B Stecke das weiße USB-Kabel in den mittleren Mini-USB-Anschluss und verbinde den Raspberry Pi nun mit dem USB-Anschluss des Laptops (siehe Bild rechts).</p> <p>Im schwarzen Gehäuse sollte jetzt ein Lämpchen leuchten.</p>	
<p>13. Zur Übertragung des geänderten Flugblatts auf den Raspberry Pi brauchst du ein FTP-Programm, z.B. FileZilla.</p> <p>Start durch Doppelklick auf das Icon auf dem Desktop:</p>	
<p>14. 1. Gib in die obere Zeile ein: Server: raspberrypi.local Benutzername: pi Passwort: turingbus Port: 22</p> <p>2. Wähle im linken Fenster unter „Lokal“ den Ordnerpfad aus, in dem die von dir bearbeitete Datei index.html liegt (Bsp. rechts).</p> <p>3. Trage im rechten Fenster folgenden Ordnerpfad für die Webseite ein: /var/www/html (anderen löschen).</p> <p>4. Ziehe die Datei index.html & deine Bilder mit dem Mauszeiger in das rechte Fenster.</p> <p>5. Bei der Frage, ob Dateien überschrieben werden sollen, wähle „ja“ aus.</p>	

15.	Teste, ob du dein Flugblatt nun auf dem Hotspot sehen kannst, indem du dich mit dem WLAN deiner Gruppe verbindest. Dies kannst du über dein Smartphone oder den Laptop machen.	
	Problemlösung: Gerät funktioniert nicht richtig, gelbes Dreieck wird im Gerätemanager (Windows) angezeigt	<ol style="list-style-type: none"> 1. <i>Gerätemanager</i> öffnen (unter Windows) 2. Das "USB Ethernet / RNDIS Gadget" –Gerät öffnen und auf <i>Eigenschaften</i> gehen 3. Unter dem Reiter <i>Erweitert</i> muss eine statischen IP-Adresse:192.168.43.1 eingegeben werden

Aufgabe 5 (Zusatz) – Ändern der Schriftart und -größe		
16.	Suche das CSS-Attribut font-family . Füge eine neue Schriftart mit Komma getrennt vor den anderen (Roboto, Verdana) ein. Wähle eine der folgenden: <i>Georgia, Palatino, Linotype, Times, Times New Roman, Arial, Geneva, Helvetica, Impact, Lucida Grande, Lucida Sans, Tahoma, Trebuchet MS, Verdana, Monaco, Lucida Console, Courier, New Courier</i> Achtung: Schriftarten aus mehreren Worten, wie bspw. die "Times New Roman", müssen in Anführungszeichen eingefügt werden. Speichere die Änderungen und schau dir die Resultate im Browser an. Wähle eine Schriftart, die dir gefällt.	font-family: Roboto, Verdana;  "Times New Roman",
17.	Suche das CSS-Attribut font-size und ändere dessen Wert zum Beispiel von 2em auf 2.5em oder von 28pt auf 16 pt . Speichere die Änderungen und schau dir die Resultate im Browser an. Wähle eine Schriftgröße, die dir gefällt.	font-size:28pt; → font-size: 16pt; font-size:2em; → font-size: 2.5em;

Aufgabe 6 auf der nächsten Seite

TEIL 3 – Dem Flugblatt einen anderen Namen geben		
Aufgabe 6 (Zusatz) – Umbenennen des WLAN-Hotspots		
18.	Verbindung zum Raspberry Pi über PUTTY aufbauen	<ol style="list-style-type: none"> Öffne das Programm PUTTY (Desktop)  <ol style="list-style-type: none"> Gib in die obere Zeile ein: Server: raspberrypi.local Username: pi Passwort: turingbus Enter-Taste drücken ()
19.	<p>Editiere nun den Namen des Hotspots in der Datei hostapd.conf auf folgende Weise:</p> <p><i>Bewegung in den Kommandozeilen über die Cursertasten! (siehe Bild)</i></p> 	<ol style="list-style-type: none"> In Kommandozeile schreiben: sudo nano /etc/hostapd/hostapd.conf  <ol style="list-style-type: none"> Dort den Hashtag „#“ entfernen vor der Zeile #ssid=GruppeX GruppeX ersetzen durch einen Namen deiner Wahl Um die Änderung zu speichern, drücke Strg-O. ENTER  Verlasse den Editor nano mit Strg-X.
20.	Starte den hostapd-Service neu.	<ol style="list-style-type: none"> In Kommandozeile schreiben: sudo service hostapd restart 
21.	Starte den Raspberry Pi neu und warte 35 Sekunden. Euer WLAN sollte dann angezeigt werden als Hotspot.	<ol style="list-style-type: none"> In Kommandozeile schreiben: sudo shutdown -r now
22.	Teste, ob du mit deinem Smartphone oder Laptop nun einen neuen Hotspot mit deinem eben gewählten Namen sehen kannst und verbinde dich.	

Farbpalette RGB-HEX Ausgewählte Web-Farben

HTML name	R G B		
	Hex	Decimal	
Pink colors			
Pink	FF C0 CB	255	192 203
LightPink	FF B6 C1	255	182 193
HotPink	FF 69 B4	255	105 180
DeepPink	FF 14 93	255	20 147
PaleVioletRed	DB 70 93	219	112 147
MediumVioletRed	C7 15 85	199	21 133
Red colors			
LightSalmon	FF A0 7A	255	160 122
Salmon	FA 80 72	250	128 114
DarkSalmon	E9 96 7A	233	150 122
LightCoral	F0 80 80	240	128 128
IndianRed	CD 5C 5C	205	92 92
Crimson	DC 14 3C	220	20 60
Firebrick	B2 22 22	178	34 34
DarkRed	8B 00 00	139	0 0
Red	FF 00 00	255	0 0

Orange colors			
OrangeRed	FF 45 00	255	69 0
Tomato	FF 63 47	255	99 71
Coral	FF 7F 50	255	127 80
DarkOrange	FF 8C 00	255	140 0
Orange	FF A5 00	255	165 0

Yellow colors			
Yellow	FF FF 00	255	255 0
LightYellow	FF FF E0	255	255 224
LemonChiffon	FF FA CD	255	250 205
LightGoldenrodYellow	FA FA D2	250	250 210
PapayaWhip	FF EF D5	255	239 213
Moccasin	FF E4 B5	255	228 181
PeachPuff	FF DA B9	255	218 185
PaleGoldenrod	EE E8 AA	238	232 170
Khaki	F0 E6 8C	240	230 140
DarkKhaki	BD B7 6B	189	183 107
Gold	FF D7 00	255	215 0

HTML name	R G B		
	Hex	Decimal	
Purple, violet, and magenta colors			
Lavender	E6 E6 FA	230	230 250
Thistle	D8 BF D8	216	191 216
Plum	DD A0 DD	221	160 221
Violet	EE 82 EE	238	130 238
Orchid	DA 70 D6	218	112 214
Fuchsia	FF 00 FF	255	0 255
Magenta	FF 00 FF	255	0 255
MediumOrchid	BA 55 D3	186	85 211
MediumPurple	93 70 DB	147	112 219
BlueViolet	8A 2B E2	138	43 226
DarkViolet	94 00 D3	148	0 211
DarkOrchid	99 32 CC	153	50 204
DarkMagenta	8B 00 8B	139	0 139
Purple	80 00 80	128	0 128
Indigo	4B 00 82	75	0 130
DarkSlateBlue	48 3D 8B	72	61 139
SlateBlue	6A 5A CD	106	90 205
MediumSlateBlue	7B 68 EE	123	104 238

Cyan colors			
Aqua	00 FF FF	0	255 255
Cyan	00 FF FF	0	255 255
LightCyan	E0 FF FF	224	255 255
PaleTurquoise	AF EE EE	175	238 238
Aquamarine	7F FF D4	127	255 212
Turquoise	40 E0 D0	64	224 208
MediumTurquoise	48 D1 CC	72	209 204
DarkTurquoise	00 CE D1	0	206 209
LightSeaGreen	20 B2 AA	32	178 170
CadetBlue	5F 9E A0	95	158 160
DarkCyan	00 8B 8B	0	139 139
Teal	00 80 80	0	128 128

Blue colors

LightSteelBlue	B0 C4 DE	176 196 222
PowderBlue	B0 E0 E6	176 224 230
LightBlue	AD D8 E6	173 216 230
SkyBlue	87 CE EB	135 206 235
LightSkyBlue	87 CE FA	135 206 250
DeepSkyBlue	00 BF FF	0 191 255
DodgerBlue	1E 90 FF	30 144 255
CornflowerBlue	64 95 ED	100 149 237
SteelBlue	46 82 B4	70 130 180
RoyalBlue	41 69 E1	65 105 225
Blue	00 00 FF	0 0 255
MediumBlue	00 00 CD	0 0 205
DarkBlue	00 00 8B	0 0 139
Navy	00 00 80	0 0 128
MidnightBlue	19 19 70	25 25 112

HTML name	R G B	
	Hex	Decimal
Blue colors		
LightSteelBlue	B0 C4 DE	176 196 222
PowderBlue	B0 E0 E6	176 224 230
LightBlue	AD D8 E6	173 216 230
SkyBlue	87 CE EB	135 206 235
LightSkyBlue	87 CE FA	135 206 250
DeepSkyBlue	00 BF FF	0 191 255
DodgerBlue	1E 90 FF	30 144 255
CornflowerBlue	64 95 ED	100 149 237
SteelBlue	46 82 B4	70 130 180
RoyalBlue	41 69 E1	65 105 225
Blue	00 00 FF	0 0 255
MediumBlue	00 00 CD	0 0 205
DarkBlue	00 00 8B	0 0 139
Navy	00 00 80	0 0 128
MidnightBlue	19 19 70	25 25 112

Gray and black colors

Gainsboro	DC DC DC	220 220 220
LightGray	D3 D3 D3	211 211 211
Silver	C0 C0 C0	192 192 192
DarkGray	A9 A9 A9	169 169 169
Gray	80 80 80	128 128 128
DimGray	69 69 69	105 105 105
LightSlateGray	77 88 99	119 136 153
SlateGray	70 80 90	112 128 144
DarkSlateGray	2F 4F 4F	47 79 79
Black	00 00 00	0 0 0

Brown colors		
Cornsilk	FF F8 DC	255 248 220
BlanchedAlmond	FF EB CD	255 235 205
Bisque	FF E4 C4	255 228 196
NavajoWhite	FF DE AD	255 222 173
Wheat	F5 DE B3	245 222 179
Burlywood	DE B8 87	222 184 135
Tan	D2 B4 8C	210 180 140
RosyBrown	BC 8F 8F	188 143 143
SandyBrown	F4 A4 60	244 164 96
Goldenrod	DA A5 20	218 165 32
DarkGoldenrod	B8 86 0B	184 134 11
Peru	CD 85 3F	205 133 63
Chocolate	D2 69 1E	210 105 30
SaddleBrown	8B 45 13	139 69 19
Sienna	A0 52 2D	160 82 45
Brown	A5 2A 2A	165 42 42
Maroon	80 00 00	128 0 0

HTML name	R G B		
	Hex	Decimal	
White colors			
White	FF FF FF	255	255 255
Snow	FF FA FA	255	250 250
Honeydew	F0 FF F0	240	255 240
MintCream	F5 FF FA	245	255 250
Azure	F0 FF FF	240	255 255
AliceBlue	F0 F8 FF	240	248 255
GhostWhite	F8 F8 FF	248	248 255
WhiteSmoke	F5 F5 F5	245	245 245
Seashell	FF F5 EE	255	245 238
Beige	F5 F5 DC	245	245 220
OldLace	FD F5 E6	253	245 230
FloralWhite	FF FA F0	255	250 240
Ivory	FF FF F0	255	255 240
AntiqueWhite	FA EB D7	250	235 215
Linen	FA F0 E6	250	240 230
LavenderBlush	FF F0 F5	255	240 245
MistyRose	FF E4 E1	255	228 225

5.7.2 Materialien für das Einführungsmodul Netzwerkbegriffe

5.7.2.1 Filme für das Netzwerkgrundlagen-Erweiterungsmodul

Nummer 1: Dieses Video, hauptsächlich der Teil zwischen 1:03 und 2:19, eignet sich als Einstieg in die Thematik und kann vor dem Film von Aaron Titus gezeigt werden, um weitere Begriffe rund um das WWW zu erklären. Es werden jedoch nur einige Begriffe daraus für den Workshop Digitales Flugblatt benötigt. Die anderen gezeigten „Teile des Internets“ sind zum Teil heute nur noch wenig geläufig.
<https://www.youtube.com/watch?v=BHxLo5ETPV8&t=4s> (Autor: Maira Kedrowitsch)

Nummer 2: Deutsch synchronisierter Film „Das Internet in 5 Minuten erklärt“ (ab 0:26 bis Ende), https://youtu.be/9EcCs4E0_0Y Die Filmdatei finden Sie auch im *Ordner Materialien* („Erklärvideo.mp4“). *Originalquelle Englisch:* Aaron Titus: „How the Internet Works In 5 Minutes“, https://youtu.be/7_LPdttKXPc

alternativ:

Filme aus Khanacademy-Kurs „How the Internet Works“ (alle CC-BY-NC-SA und können daher vorher heruntergeladen werden)

„IP addresses and DNS“ URL: <https://www.khanacademy.org/computing/computer-science/internet-intro/internet-works-intro/v/the-internet-ip-addresses-and-dns>

„HTTP and HTML“ URL: <https://www.khanacademy.org/computing/computer-science/internet-intro/internet-works-intro/v/the-internet-http-and-html>

„Wires, Cables, and WiFi“ URL: <https://www.khanacademy.org/computing/computer-science/internet-intro/internet-works-intro/v/the-internet-wires-cables-and-wifi>

5.7.2.2 Vorlage: Netzwerk-Grundbegriffe-Tafelbild

Die Teilnehmer*innen legen die vorher ausgedruckten und ausgeschnittenen Bilder („A3-Druckvorlage-Schaubild-Internet.pdf“ finden Sie bei den Materialien) zu einem „Netzwerk-schaubild“ zusammen, ordnen die korrekten Begriffe den Bildern zu und erklären ihre Funktion im Internet.

Dabei können drei Schaubilder entstehen bzw. können Nummer 1 und 2 auch als ausfüllbares Arbeitsplatt genutzt werden:

1. Die Bestandteile des Internets



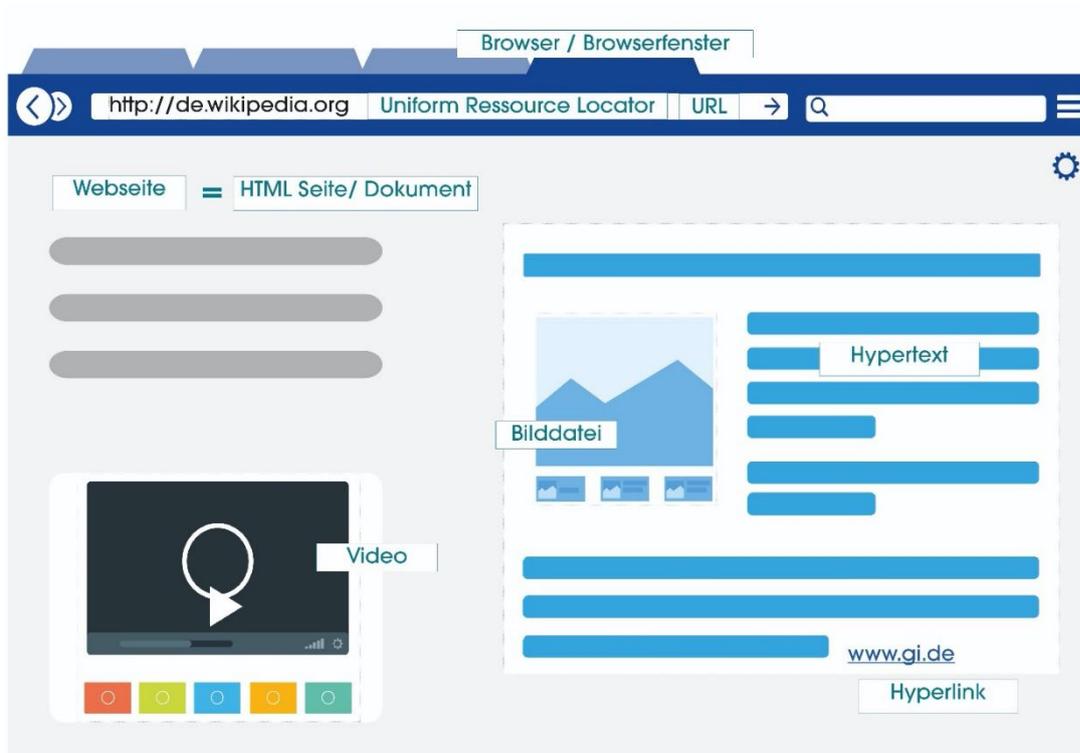
Bestandteile des Internets

Usenet (Diskussionsforen, Newsgruppen)	Telenet (Teletype network, Fernschreiber Netzwerk)	FTP (File Transfer Protocol,	www (world wide web)	E-Mail (electronic Mail)
--	--	--	--------------------------------	------------------------------------

Usenet <ul style="list-style-type: none"> nur Textform ohne Bilder veraltet entstand weit vor dem www 	Telenet <ul style="list-style-type: none"> ist ein Netz(werk)-protokoll: Kommunikationsprotokoll für den Austausch von Daten zwischen Rechnern bzw. Prozessoren, die in einem Netzwerk miteinander verbunden sind es kontrolliert und speichert welche Daten, von wem und an wem gesendet werden spezialisiert auf zeichenorientierten Datenaustausch (Textnachrichten) 	FTP <ul style="list-style-type: none"> Netzwerkprotokoll zur Kontrolle der Übertragung von Daten über IP-Netzwerke IP = Internet Protocol, stellt die Grundlage des gesamten Internets dar
--	---	---

2. Netzbegriffe (Browser & Website)

Eine unausgefüllte Vorlage finden Sie im *Ordner Materialien* (Dokumentenname: „schaubild_Browser_unausgefüllt.jpg“).

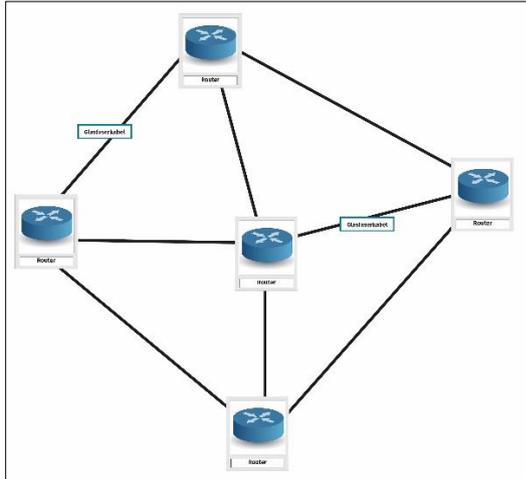


3. Schaubild Internet (Netzbegriffe)

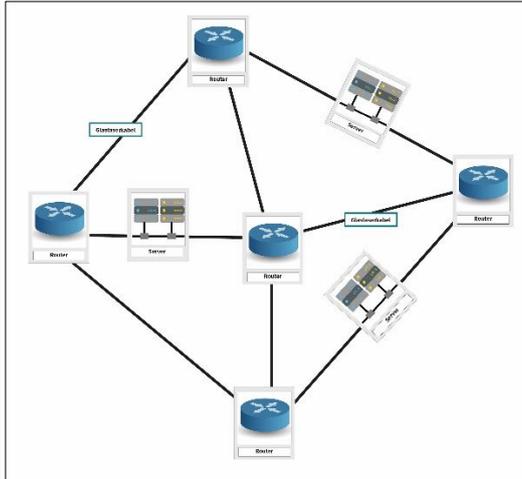
Die Schüler sollen nach den Videos die gelernten Begriffe in einem Netz einordnen und in Verbindung bringen. Dabei können die unterschiedlichen Übertragungsmedien einbezogen werden sowie auch die Vergabe von IP-Adressen (Thematik IP-Netzwerke und Internet der Dinge, IOT)

Je nach Vorkenntnissen und Alter der Schüler kann das Schaubild auch vereinfacht dargestellt werden. Im Folgenden wird exemplarisch dargestellt wie die Schritte zum Aufbau eines Netzwerkes aussehen könnten. Im Ordner Materialien finden Sie Schaubilder für die einzelnen Internetverbindungen mit und ohne IP-Adressen sowie eine schrittweise Erweiterung des Schaubildes.

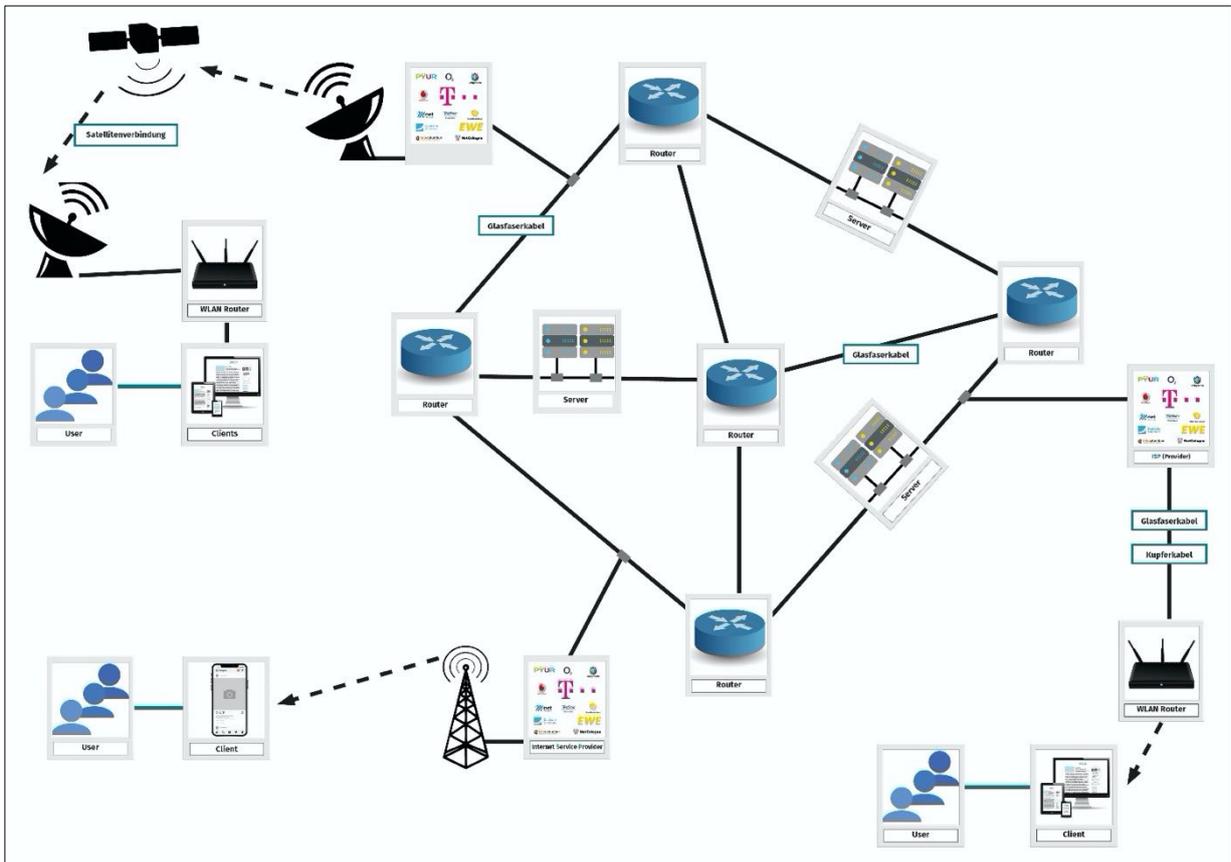
(Im Ordner Materialien: „A3-Druckvorlage-Schaubild-Internet.pdf“)



2) Netz mit Routern als Richtungsgeber



3) Netz mit Routern und Servern



1) Schaubild mit den unterschiedlichen Verbindungsarten (kabelgebundener Anschluss, Funk und Satellit)

Weitere Darstellungen und die Druckvorlage finden Sie im *Ordner Materialien*.

6 Workshop: Wie entsteht ein Datensatz?

Autor: Maximilian Voigt, Bela Seeger

Betreuungsschlüssel: 1:12

Dauer: 2-3 h

Vorkenntnisse: Informatische und physikalische Grundlagen sind von Vorteil

Verwendete Technik: Notebook mit Arduino IDE und Internetzugang, ein Arduino Hardware-Set¹, eine Auswahl verschiedener Sensoren, bei Bedarf: Bastelmaterialien wie Pappe, Farbe, Kleber etc.

Die Diskussion um Künstliche Intelligenz dreht sich oft vorrangig um Anwendungsmöglichkeiten und Zukunftspotentiale. Seltener wird über ihren Treibstoff gesprochen: Daten. Wie ein Datensatz

entsteht, wie man mithilfe von Sensoren selbst Daten erstellen kann und welche schwerwiegenden Folgen ein Datensatz von niedriger Qualität haben kann, wird hier anhand von Hands-On-Formaten nachvollziehbar. In diesem Modul erstellen wir Datensätze mithilfe von Mikrocontrollern (Arduinos).

6.1 Ein Blick hinter die Fassade sogenannter Künstlicher Intelligenz

6.1.1 Im Fokus: Daten & Sensoren

Der Workshop schaut hinter die Fassade moderner Systeme und bricht die Vorstellungen von vermeintlich intelligenten Objekten. Er behandelt die Grundstruktur von Computern, zeigt wie sie durch Sensoren ihre Umgebung vermessen, wie die erhobenen Daten in „intelligenten“ Programmen verarbeitet werden und beleuchtet ethische Dimensionen.

Computerprogramme, die komplexe Sachverhalte klassifizieren sollen - eine Aufgabe, die in der Regel „Künstlicher Intelligenz“ (KI) zugeordnet wird -, sind auf hochwertige Daten angewiesen. Denn bei der Entwicklung neuronaler Netzwerke werden sie als Trainingsgrundlage benötigt. Ein selbstlernendes Computerprogramm kann also immer nur so gut sein wie die Daten, auf denen es basiert. Wenn es um das Verständnis von „KI“ geht, spielt deswegen die Entstehung und Verarbeitung von Daten eine wichtige Rolle. Der hier skizzierte Workshop soll diese Themen behandeln. Im Fokus steht die Entstehung von Daten durch die Verwendung von Sensoren sowie die anschließende Verarbeitung. Besonders behandelt werden auch die ethischen Dimensionen sogenannter „Künstlicher Intelligenz“. Außerdem geraten die Teilnehmenden in die Rolle, eigene „intelligente“ Objekte zu entwerfen und so zu der Frage, was Intelligenz eigentlich bedeutet. Dadurch fokussiert das Konzept nicht die detaillierte Funktionsweise „intelligenter“ Computerprogramme, sondern die Vorstellungen, die mit ihnen einhergehen. Was macht Objekte intelligent? Wie gut und autark können vermeintlich intelligente Systeme arbeiten?

6.1.2 Ablauf

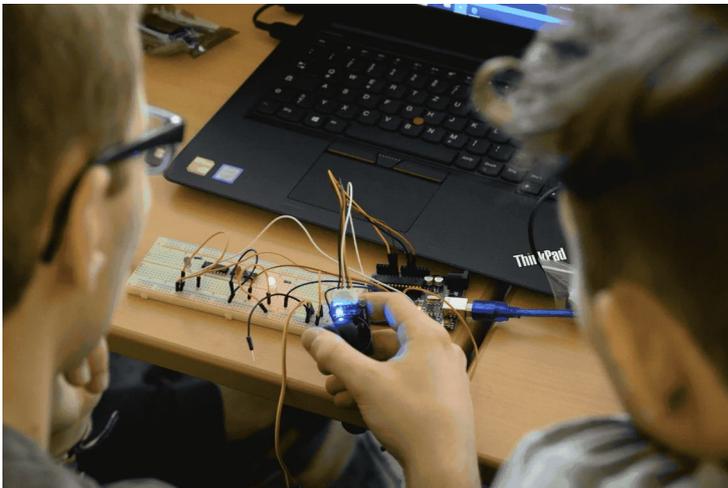
Der Workshop ist in folgende Einheiten aufgeteilt - der detaillierte Ablauf befindet sich nach den Variantenvorstellungen.

6.1.2.1 Basis-Workshop: 3 Stunden

- Kennenlernen und Auseinandersetzung mit dem Thema: Wo treffen wir im Alltag auf „KI“?



- [Videodiskussion](#): Was sind die ethischen Aspekte von „KI“?
- Den Arduino kennenlernen: Wir programmieren das Hello World der Mikrocontroller-Programmierung: eine blinkende LED

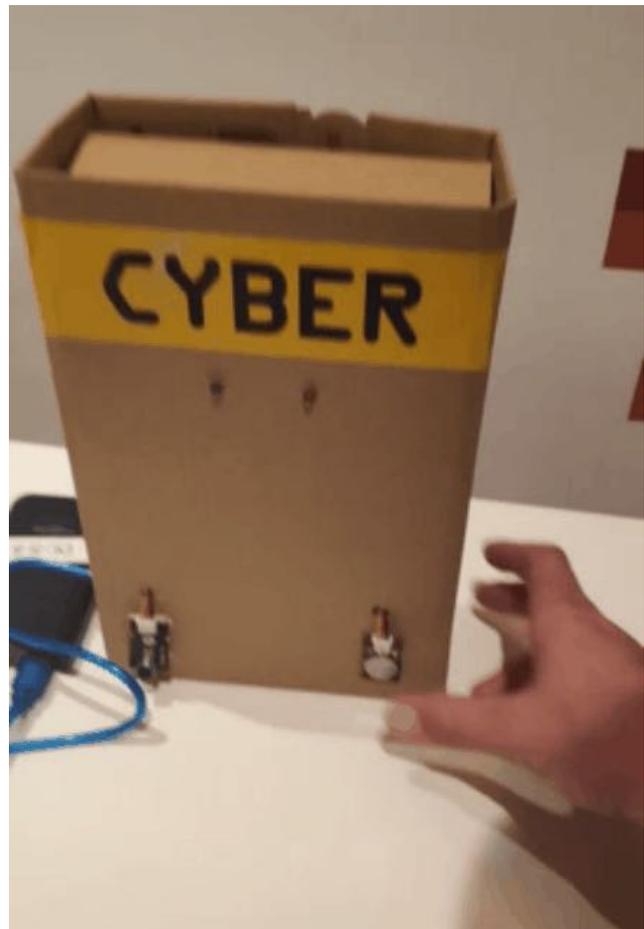


- Sensoren entdecken: Welche Sensoren gibt es und welchen menschlichen Sensoren entsprechen sie? [Arbeitsblatt](#)
- Sensoren testen und auswerten: Wie genau messen verschiedenen Sensoren, welche Tücken haben sie?
- Optionaler Abschluss / Pause

6.1.2.2 Vertiefung und Entwicklung des eigenen Zugangs: 3 Stunden

**Freies Arbeiten mit der Aufgabe:
Baut ein Objekt, das mehr oder weniger
intelligent wirkt.**

Hier entwickelten die Teilnehmenden z.B. einen "intelligenten Lügendetektor". Zwei Sensoren erfassen den Probanden: ein Mikrofon sowie ein kapazitiver Sensor (Touch-Schalter). Durch das Auflegen der Hand und das Sprechen einer Aussage wird der Eindruck erweckt, dass im Innern eine komplexe Berechnung stattfindet. Die LED zeigt das Ergebnis. Wir der Code betrachtet, ist das Verfahren rein zufällig. Die Jugendlichen haben also das Gehäuse genutzt, um die Funktion und die Einfachheit des Gegenstandes zu verschleiern. Die Intelligenz, die wir dem Gegenstand zuweisen, erscheint als rein äußerlich:



Was?	min	Inhalt	Methode	Lernziel	Material / Aufbau
Einführung	15	<p>Begrüßung, Vorstellung der Teilnehmenden, Kenntnisstand (Wo ist überall KI drin?)</p> <p>Es wird eine Art Reise nach Jerusalem gespielt. Personen, die keinen Platz bekommen haben, müssen ihren Namen und etwas mit KI-Bezug in ihrem Leben nennen (Vermutung). Alle, die sich dadurch auch angesprochen fühlen, müssen aufspringen und sich einen neuen Sitzplatz suchen. Beispiel: „Ich heiße Max und ich nutze Instagram“. Anschließend beginnt das Spiel von vorne.</p>	Der/die Workshopleiter*in spricht zwischen- durch die genannten Technologien an und versucht herauszufinden, was genau in ihnen über "künstliche Intelligenz" verfügt.	Die TN haben sich einander vorgestellt und dem Thema angenähert	Sitzkreis mit n-1 Sitzplätzen
	15	<p>Den TN wird das Video schrittweise gezeigt. Da es in englischer Sprache ist, wird zwischendurch nach einer kurzen Zusammenfassung gefragt. Die drei Fragen lauten: Welches Problem soll diese Technologie lösen? Wie arbeitet sie? Wenn ihr es selbst anwenden könntet, was würdet ihr damit machen, welche Vor- bzw. Nachteile seht ihr? Das klassische Stichwort ist hier das der Überwachung. Wenn wir unseren eigenen Haushalt überwachen können, gibt es bestimmte Vorteile. Haben fremde Menschen Zugriff, wird es problematisch. In der Regel sind sich alle TN einig, dass es eine problembehaftete Technologie ist. Zum Abschluss wird kurz das Smartphone angesprochen, das die TN in ihrer Tasche tragen und mit beinahe allen Sensoren des Sensorboards ausgestattet ist.</p>	Videodiskussion (Plenum)	Die TN können bewerten, welche moralisch-ethischen Dimensionen das Thema KI hat und wie KI im Bereich der Überwachung eingesetzt werden kann.	https://www.youtube.com/watch?v=aqbKrrru2co
	15	<p>Die TN sollen sich in 2er Gruppen aufteilen und sich die zur Verfügung gestellte Hardware genau angucken. Was kommt ihnen bekannt vor? Jede 2er Gruppe soll einen Gegenstand vorstellen.</p>	2er-Gruppenarbeit / Vorstellung im Plenum	Die TN kennen die Hardware, die ihnen zur Verfügung gestellt wurde	Hardware-Sortiment inkl. Arduino

Arduino & elektro-technische Einführung	45	Die TN sollen einmal gemeinsam das Hello World der Mikrocontroller-Programmierung umsetzen: eine blinkende LED. Dazu sollen sie nach Anleitung im 1. Schritt den LED-Stromkreis aufbauen und testen sowie im 2. das Computer-Programm schreiben.	2er-Gruppenarbeit	Die TN können die Grundfunktionen des Arduinos einsetzen und den Grundaufbau des Arduino-Controllers realisieren	Hardware-Sortiment inkl. Arduino und Laptop
Sensoren testen	15	Es werden Blätter mit darauf abgebildeten menschlichen Sensoren ausgeteilt. Außerdem bekommt jede Gruppe zwei Sensoren. Anschließend sollen sie die elektrotechnischen Sensoren denen der abgebildeten menschlichen Sensoren zuordnen. Im Plenum soll dann jede Gruppe ihre Vermutung äußern. Der*Die Workshopleiter*in ordnet das Gesagte dann ein.	Match the Thing: 4er-Gruppenarbeit	Die TN kennen eine Auswahl von Sensoren und können sie menschlichen Sensoren zuordnen	https://github.com/okfde/edu-labs/blob/master/sets/extern/turing-bus-arbeitsblatt-menschliche-sensoren.pdf
	45	Anschließend darf sich jede Gruppe einen Sensor aussuchen (es darf auch einer der anderen sein) und ihn mit Hilfe des Internets programmieren. Das Ziel ist es, Messergebnisse zu Gesicht zu bekommen. Sind die Gruppen schnell fertig, dann sollen sie sich gegenseitig ihren Sensor erklären bzw. dabei helfen, diesen ebenfalls aufzubauen.	2er-Gruppenarbeit	Die TN wissen, wie sie mindestens einen Sensor an einen Mikrocontroller anschließen und programmieren müssen, um Messergebnisse zu erhalten.	
	10	Was haben die Messungen abgebildet? Wie gut waren die Messergebnisse? Waren sie verlässlich? Wie gut würde eine "KI" funktionieren, die auf dieser Datengrundlage arbeitet?	Diskussion im Plenum	Die TN können die Genauigkeit ihres Sensors einschätzen und bewerten, wie gut oder schlecht eine "KI" auf der Grundlage der Daten arbeiten würde.	
PAUSE	30 - 60	Aufgabe für die Pause: Ideen entwickeln, ggf. schon Gruppen bilden			
Idee entwickeln	15	Die TN sollen sich in 4er Gruppen aufteilen und gemeinsam eine Idee entwickeln. Das Ziel ist es, ein Objekt zu bauen, welches mehr oder weniger intelligent wirkt. Es darf also auch Intelligenz vortäuschen. Idealerweise hat jede Person in der Gruppe einen der folgenden Aufgabenbereiche: Hardware, Software, Design, Kommunikation.		Die TN lernen ihr Projekt in einem engen Zeitraum zu planen und Rollen zu verteilen	https://docs.google.com/document/d/1rZJFjFy50iRog60ArG4BhSCHe6XHBunlniJkCYTZ-Hoc/edit#head=h.pefufao9fzxd
Freies Arbeiten	60	Die TN entwickeln ihre Prototypen. Dazu können sie das Internet nutzen. Außerdem haben sie diverse Bastelmaterialien wie Pappe, Stifte und Klebeband, um	Gruppenarbeit in ca. 4er-Gruppen	Die TN können selbstorganisiert ein Projekt umsetzen, sie erkennen, dass die	Papier, Schere, Klebeband, Heißkleber, Lötstation,

		eigene Gehäuse zu bauen.		Zuweisung von Intelligenz viel mit einer gut gebauten Fassade zu tun hat.	Kabel, Farbe, Stifte etc.
Ab-schluss / Vorbereitung Präsentation	30	Die letzten 30 Minuten sollen für die Fertigstellung sowie die Vorbereitung der Präsentation genutzt werden. Es ist wichtig immer wieder den Ablauf der Zeit zu thematisieren, damit alle pünktlich fertig werden.	Gruppenarbeit in ca. 4er-Gruppen	Die TN können selbstorganisiert ein Projekt umsetzen, sie erkennen, dass die Zuweisung von Intelligenz viel mit einer gut gebauten Fassade zu tun hat	Papier, Schere, Klebeband, Heißkleber, Lötstation, Kabel, Farbe, Stifte etc.
	30	Abschlussrunde: jede Gruppe präsentiert kurz ihre Ergebnisse. Bei der Präsentation soll versucht werden das Publikum davon zu überzeugen, dass es sich bei dem Objekt um eines mit einer gewissen Intelligenz handelt.	Präsentation im Plenum (frei gesprochen, bei Bedarf Hilfsmittel)	Die TN lernen was ein Gehäuse und Präsentationsmethoden dazu beitragen wie sehr wir Gegenständen Intelligenz zuweisen	

6.2 Lernziele des Workshops

- Die TN verstehen die Verbindung von Daten und KI
- Die TN wissen wie z.B. ein Smart Home oder Phone und eine nachgeschaltete KI an Informationen kommt
- Die TN erkennen, dass eine KI auch nur ein Computerprogramm ist
- Die TN kennen ein Beispiel, in dem KI im Bereich Smart Home zum Einsatz kommt
- Die TN können 3 technische und menschliche Sensoren einordnen
- Die TN können selbstständig Daten erfassen, indem sie einen Mikrocontroller nutzen
- Die TN erkennen, dass ein Mikrocontroller die Basis vieler Computer in ihrem Alltag ist
- Die TN verstehen, dass KI-Systeme nur so zuverlässig arbeiten können wie die Daten sind, die sie bekommen.
- Die TN erkennen, dass KI-Systeme anfällig für Fehler sind
- Die TN können ethische Aspekte von KI bewerten/nennen

6.3 Tipps

- Wenn die Vorkenntnisse stark auseinandergehen, sollte die Zusammenstellung der Teams entsprechend moderiert und besonders schnelle / vorgebildete Teilnehmende dazu angehalten werden, Langsamere zu helfen.
- Da der Workshop sehr hardwarelastig ist und dieser oft mit Respekt oder einer gewissen Zurückhaltung begegnet wird, ist es wichtig, den Rahmen des Workshops möglichst locker und humorvoll zu gestalten. Außerdem sollten immer wieder alltagstaugliche Metaphern verwendet werden, wenn einzelne Technologien eingeleitet werden: Die Arduino IDE ist ein Übersetzer zwischen menschlicher und maschineller Sprache; Der Stromfluss ist wie Wasser, Widerstände wie Rohrverengungen, Sensoren wie die Augen und Ohren der Computer etc.
- Weil viele Sensoren zum Einsatz kommen, die ganz unterschiedlich anzusteuern sind, kann der Workshopleiter/die Workshopleiterin nicht alle kennen. Damit sollte ganz offen und souverän umgegangen werden. Denn letztendlich entspricht das ja auch der Alltagspraxis eines Programmierers/einer Programmiererin, Bibliotheken und Schnittstellen zu recherchieren.

6.4 Weiterführende Literatur

- [Computing Machinery and Intelligence](#)
- [Material, auf das dieser Workshop aufbaut](#)

7 Workshop: Meine Chatbotfreundin

Autorinnen: Dr. Andrea Knaut, Katinka Richter

Betreuungsschlüssel: 1:10

Dauer: 2-3h

Verwendete Technik: Chatbot Software (ELIZA, Scratch) Laptops

„Hallo, ich bin dein virtueller Assistent. Wie kann ich dir behilflich sein?“

Wer ist ihnen noch nicht begegnet, den freundlichen Chatbots auf Bank-Webseiten, in Messengern oder als Teil der Sprachassistenzsysteme von Alexa bis Siri?

Das Imitationsspiel Turings, in dem ein Chatbot eine Frau nachspielt, gilt als der historische Intelligenztest für Künstliche Intelligenz. Wir schauen uns an, wie so ein Chatbot funktioniert, ab wann er als intelligent gilt, und erproben, wie kinderleicht wir uns selbst einen programmieren können ...

7.1 Ablauf

Der Ablauf des Workshops ist zweigliedrig. In einem ersten Schritt spielen wir mit den Teilnehmenden den Turing-Test (imitation game) durch, um sie an das Konzept von „Künstlicher Intelligenz“ (KI)

heranzuführen und zum Nachdenken über den Intelligenz-Begriff anzuregen.

In einem zweiten Schritt programmieren wir mit den Teilnehmenden eigene Chatbots, um auch die technische Umsetzung einer KI kennen zu lernen. Anschließend diskutieren wir im Plenum über das Gelernte und, wie sich die Erfahrungen mit der Chatbotprogrammierung auf die Ergebnisse des Turing-Tests anwenden lassen.

LERNZIELE

1	Die Schülerinnen und Schüler erschließen sich selbstständig neue Anwendungen und Informatiksysteme – in diesem Falle Chatbots.
2	Die Schülerinnen und Schüler modifizieren und ergänzen Quelltexte von Programmen nach Vorgaben. Sie ergänzen AIML-Skripte und Python-Code fertiger Chatbots.
3	Die Schülerinnen und Schüler kooperieren in Projektarbeit bei der Bearbeitung eines informatischen Problems. Hier versuchen sie gemeinsam, die Funktionsweise eines Chatbots zu erschließen, zu modifizieren und zu testen.
4	Die Schülerinnen und Schüler erfahren die extreme Begrenztheit der „Verständnisfähigkeit“ der Maschine (hier: Chatbots) beim Entwurf und Test eigener Problemlösungen.
5	Die Schülerinnen und Schüler verfügen über grundlegende Kenntnisse zur historischen Entwicklung der Informatik, hier in Bezug auf die Rolle der Chatbots für die Teildisziplin Künstliche Intelligenz.

6	Die Schülerinnen und Schüler formulieren angemessene Kriterien zur Einschätzung informatischer Sachverhalte und wenden diese an.
---	--

BEI DIESEM WORKSHOP GIBT ES ZWEI VARIANTEN:

A: 180-Min-Variante

B: 90-Min-Variante

Beide Workshops beinhalten Teil B, der Unterschied ist, dass bei Variante A ein längeres Einstiegsmodul hat.

Zeit/Minuten	Phase (Ziel)	Inhalt	Methode/Sozialform	Materialien
A/B: 0-10	Einstieg	kurze Vorstellungsrunde mit Kommentar zur Frage, wo schon mal einem Chatbot begegnet wurde	Stuhlkreis / Sitzen rund um einen mittleren Tisch: Gespräch	Namenszettel, Stifte, Whiteboard/Flipchart für Stichworte
A: 10-30 (lang)	Einstieg/ Aufwärmen	<p>Turing-Test-Spiel aus der AI-Unplugged-Reihe</p> <p>Es werden vier S bestimmt, von denen zwei eine der Spiel-Karteikarten erhalten mit ihrer Rolle, die sie nicht verraten dürfen. Zwei weitere sind Läufer*innen und dürfen ebenfalls nie verraten, welche Rolle diejenigen haben, deren Antworten sie bringen. "Computer" und "Mensch" verlassen den Raum und gehen beispielsweise in den Flur. Der "Mensch" bekommt die Aufgabe, auf alle Fragen, die ihm der*die Läufer*in bringt, kurz und ehrlich zu antworten. Der "Computer" muss die Antworten geben, die auf dem Arbeitsblatt stehen; kursiv gedruckte Anweisungen muss er gesondert befolgen. Die Läufer*innen tragen die Fragen und die Antworten schriftlich hin und her und lassen sich nicht anmerken, was sie über die Rolle der verborgenen Personen wissen.</p> <p>Die restlichen S erhalten ein Arbeitsblatt mit vorgegebenen Fragen. Sie diskutieren einerseits Kriterien die menschliche von künstlicher Intelligenz unterscheiden anhand eigener Erlebnisse wie z.B. Alexa. Daraufhin diskutieren sie welche Frage sie wählen, anhand der vorher entwickelten Kriterien, [...] ob und wie sie mit der zu erwartenden Antwort herausfinden können, welche der verborgenen S Computer und welcher Mensch ist. Wenn sie die Antwort dann real zurückerhalten, werten sie aus,</p>	2 Räume (bzw. 1 und Flur), im Hauptraum sitzen alle anderen Schüler*innen im Stuhlkreis/um einen Tisch: Diskussion und Mensch. Maschine + ihre Läufer*innen verlassen den Raum	Karteikarte für die Spieler*innen-Rollen und die Läufer*innen, Arbeitsblätter mit den Fragen und Antworten

		ob es für das Mensch- oder Maschinesein der entsprechenden Person spricht. Sie können so viele Fragen stellen, wie auf dem Arbeitsblatt enthalten sind, um die Lösung zu finden. Sie können sich aber auch schon früher entscheiden.		
B: 10-15	Vorbereitung	Vorstellung von drei Schlüsselforschungsaufgaben für den Workshop: 1. Chatbot-Recherche-Team (grüne Karten) 2. Chatbot mit Machine Learning (gelbe Karten) 3. Chatbot mit Regelwerk (blaue Karten)	Bilden von 5 Teams zu je 2-3 Leuten	TEAMSCHILDER (einfach Arbeitsblätter nehmen) ALS PUZZLE-TEILE A: Chatbot-Recherche-Team B: ELIZA-Team C: AIML-Team D: RiveScript-Team E: SCRATCH-Chatbot-Team N.N: F: tensorflow-Team
B: 15-45	Erarbeitung (1, 3, 6)		selbstständige Arbeit in Gruppen, Unterstützung durch Mentor*innen	
		GRUPPE A: Chatbot-Recherche-Team		
		GRUPPE B: ELIZA-Team		
		Wir testen je zwei verschiedene Chatbots und versuchen, wie mit normalen Menschen mit ihnen zu sprechen. Wo liegen die Probleme? Wie erkennen wir, dass sie keine normalen Menschen sind? Stoppt die Zeit. Welche der beiden Chatbots erscheinen geeigneter für ein normales Gespräch? Was, wenn wir ein Spezialgespräch führen?	je zwei TN pro Rechner	ausgewählte Chatbots
A/B: 10-20	Auswertung (1, 6)	Wir werten aus, wie lange das Hintersicht-Führen jeweils dauerte und woran wir es erkannt haben.		
A: 20-40 (lang)	Vertiefung und Diskussion	Denkanstoß: Ist ein Chatbot nur ein Chinesisches Zimmer? Welche Rolle spielen Chatbots in Spielen?		Searle, Hirn-im-Tank-Film; Chatbots in Spielen

B: 20-30	Ausblick	Ich stelle vier verschiedene Arbeitsaufgaben vor, einen Chatbot zu erstellen, und erläutere, dass wir diese alle selbst ausprobieren werden, um ein wenig mehr dahinter zu kommen.			
B: 30-55	Erarbeitung	selbständige Gruppenarbeit mit der Bot-Erstellung, jeder bekommt ein Geheimnis einprogrammiert	<p>Gruppenbildung: (eine Gruppe besteht aus 1 oder mehreren 2er/3er-Teams)</p> <p><u>GRUPPE A</u>: Wir testen und verändern ein ELIZA-Skript.</p> <p><u>GRUPPE B</u>: Wir testen und verändern ein AIML-Skript.</p> <p><u>GRUPPE C</u>: Wir testen und verändern CHATTERBOT-Code.</p> <p><u>GRUPPE D</u>: Wie testen und verändern ein tensorflow-Chatbot. NEURAL NETWORK</p> <p><u>GRUPPE E</u>: Historische Chatbots-Recherche-Und-Text-Tour. (inkl. Filmchen, diverse Tests von Chatbots, freie Recherche, zentrale Begriffe und Funktionsweisen)</p>		
B: 55-70	Präsentation	Der*die Lehrer*in moderiert die Vorstellung der Chatbots der einzelnen Gruppen. Das aktuelle Gespräch wird an die Wand projiziert. Ein*e Schüler*in übernimmt die Eingabe. Die Chatbots werden der Reihe nach „befragt“. Die SuS machen Vorschläge für den Dialog mit dem Bot.			
B: 70-80	Auswertung und Diskussion	Die SuS äußern sich zu den Problemen, die bei der Entwicklung des Bots aufgetreten sind und überlegen, wie sie die Kommunikation mit dem Bot verfeinern könnten. Anschließend projiziert der*die Lehrer*in die Ergebnisse des Fragebogens (siehe Stunden 1 & 2) an die Wand. Die SuS äußern ihre Meinung über die Fähigkeiten aktueller KI und spekulieren über zukünftige Entwicklungen in der KI. Sie SuS äußern, ob sie bei einem erneuten Ausfüllen des Fragebogens anders agieren würden und wenn ja, warum.			
B: 80-90	Präsentation und Abschluss	<u>Gruppe E</u> gibt letzte Background-Infos: Was ist der Turing-Test, Was ist der Loebner-Prize, Wer war Eliza?			

7.2 Materialien

7.2.1 Turing-Test ¹

7.2.1.1 Arbeitsblatt „Ach wie gut, dass niemand weiß, dass ich ein Computer bin“



Zielgruppe: Sekundarstufe I, Sekundarstufe II

Darum geht's: Wie muss sich eine Maschine verhalten, um als intelligent zu gelten? Was bedeutet künstliche Intelligenz genau? Diese Fragen beschäftigen Forscher seit den Anfängen der künstlichen Intelligenz. Alan Turing hat 1950 mit dem Turing-Test eine Idee entwickelt, wie man feststellen könnte, ob eine Maschine intelligent ist. Diese Aktivität stellt den Turing-Test mit Schülerinnen und Schülern nach und soll zur Diskussion anregen, ob Computer tatsächlich so etwas wie menschliche Intelligenz zeigen können. Sie deckt außerdem auf, wie leicht man durch sorgfältig gewählte Beispiele von „Intelligenz“ von einer Maschine in die Irre geführt werden kann.

Diese Ideen stecken dahinter:

- Intelligente Systeme nutzen bestimmte Strategien, um menschliches Verhalten zu imitieren.
- Es braucht spezielle Verfahren, um die Intelligenz von Maschinen zu bewerten.
- Die Definition von (künstlicher) Intelligenz ist nicht eindeutig.

Das wird benötigt:

- Arbeitsblätter/Folie mit vorgegebenen Turing-Test Fragen für die ganze Klasse
- eine Kopie der Antworten auf die Turing-Test-Fragen (Für Rolle Computer)
- 4 freiwillige Schülerinnen oder Schüler in den Rollen Computer (1x), Mensch (1x) und Läufer (2x)

¹ Der Turing-Test-Workshop wurde von Tim Bell, Ian H. Witten und Mike Fellows entwickelt und von Annabel Lindner und Stefan Seegerer übersetzt und adaptiert. Er steht unter Creative

Commons CC-BY-SA Lizenz. Die Illustrationen stammen von Annabel Lindner und Stefan Seegerer und stehen unter CC-BY-NC 3.0 Lizenz. <http://unplugged.canterbury.ac.nz> und <https://aiunplugged.org>

So funktioniert's: Die Schülerinnen und Schüler spielen in dieser Aktivität ein Frage-Antwort-Spiel, bei dem sie allein durch das Stellen von Fragen und die Analyse der zugehörigen Antworten versuchen müssen, einen Computer von einem Menschen zu unterscheiden. Dazu nehmen je eine Schülerin bzw. ein Schüler die Rolle eines Computers und eine menschliche Rolle ein. Sie werden von ihren Mitschülerinnen und Mitschülern befragt und anhand der Antworten muss die Klasse bestimmen, wer welche Rolle repräsentiert.

Kontext: Seit Jahrhunderten streiten Philosophen, ob eine Maschine über menschliche Intelligenz verfügen kann bzw., ob das menschliche Gehirn nicht vielleicht auch nur eine sehr gute Maschine ist. Einige halten künstliche Intelligenz für eine absurde Idee, andere glauben, dass wir irgendwann Maschinen entwickeln werden, die genauso intelligent sind wie wir. Künstliche Intelligenz hat viel Potenzial, andererseits schürt die Vorstellung von intelligenten Maschinen auch Ängste.

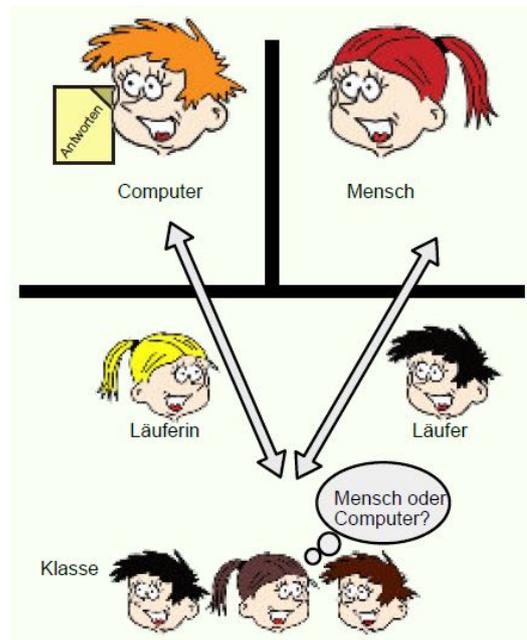
Ablauf

Thematisieren Sie mit den Schülerinnen und Schülern vor dem Spiel, ob sie Computer für intelligent halten oder annehmen, dass Computer jemals intelligent sein werden. Fragen Sie nach, wie man entscheiden könnte, ob ein Computer intelligent ist und stellen Sie kurz den Turing-Test vor, welcher mit der Aktivität nachgestellt wird.

Vor Beginn der eigentlichen Aktivität werden vier freiwillige Schülerinnen bzw. Schüler bestimmt, sie übernehmen die Rollen des Computers und des Menschen (siehe Abb. 15). Außerdem gibt es zwei Läufer, die den fairen Verlauf des Spiels absichern und mit Zettel und Stift ausgestattet werden, um die Antworten zu notieren. Die Rollen von Mensch und Computer werden von der Lehrkraft im Geheimen zugeteilt, bevor diese beiden Schülerinnen bzw. Schüler das Klassenzimmer in zwei getrennte Räume verlassen (alternativ können Sie auch eine Trennwand nutzen, es muss jedoch sichergestellt werden, dass sich die Schülerinnen und Schüler nicht gegenseitig sehen). Die Schülerin bzw. der Schüler in der Rolle des Computers erhält dort eine Kopie der Antworten auf die Fragen des Turing-Tests. Jeder der Läufer ist für eine Rolle zuständig, für welche wird ebenfalls geheim gehalten.

Die Klasse muss nun herausfinden, welche Schülerin bzw. welcher Schüler die Rolle des Computers eingenommen hat. Dazu wählt sie vom ausgeteilten Arbeitsblatt pro Runde eine Frage, die an Computer und Mensch gestellt werden soll. Nachdem eine Frage gewählt wurde, sollen die Schülerinnen und Schüler zunächst erklären, warum sie diese Frage für geeignet halten, um den Computer vom Menschen zu unterscheiden. Diese Argumentation ist das zentrale Element dieser Aufgabe, da die Klasse auf diese Weise darüber nachdenkt, wie sich die Antworten einer Person und eines „intelligenten“ Computers unterscheiden.

Dann wird die Frage von den Läufern an ihre Mitschülerinnen bzw. Mitschüler in den anderen Räumen übermittelt, ebenso werden die Antworten von dort von den Läufern zurück in die Klasse gebracht. Dabei muss der Mensch kurz und ehrlich auf die gestellte Frage antworten – also eine menschliche Antwort geben. Der Computer wählt hingegen die entsprechende Antwort vom Arbeitsblatt aus. Wenn die Anweisungen kursiv geschrieben sind, muss der Computer selbst eine Antwort ausarbeiten (z.B. die aktuelle Uhrzeit angeben). Bei der Übermittlung



der gegebenen Antworten müssen die Läufer besonders darauf achten, nicht zu offenbaren, mit wem sie interagieren.

Die Klasse diskutiert nun, welche Antwort wahrscheinlich von einem Computer stammt. Wiederholen Sie den Vorgang mit einigen weiteren Fragen, möglichst so lange, bis die Klasse eine klare Entscheidung treffen kann, wer der Computer ist. Kann die Klasse Mensch und Computer nicht verlässlich unterscheiden, hat der Computer den Turing-Test bestanden.

Hintergrund

Obwohl noch kein aktuelles Computerprogramm über so etwas wie allgemeine Intelligenz verfügt, ist die Frage, ob Computer dazu grundsätzlich in der Lage sind, noch unbeantwortet. Dies hängt maßgeblich damit zusammen, dass bereits die Definition von Intelligenz kontrovers diskutiert wird.

Vor diesem Hintergrund schlug der britische Mathematiker Alan Turing 1950 ein Verfahren vor, mit dem die Intelligenz einer Maschine bestimmt werden kann ohne eine genaue Definition von Intelligenz zu benötigen. Dieser sogenannte Turing-Test lässt den Computer seine „Intelligenz“ demonstrieren. Das Szenario des Tests ähnelt der oben beschriebenen Aktivität: Eine Fragestellerin bzw. ein Fragesteller interagiert sowohl mit einer Person als auch mit einem Computer via Chat. Kann sie oder er die beiden nicht zuverlässig voneinander unterscheiden, hat der Computer den Turing-Test bestanden. Da die Kommunikation über Chat erfolgt, kann sich der Computer nicht durch physische Eigenschaften, etwa die Stimmlage, verraten. Ein bekanntes Beispiel für ein solches Interaktionssystem ist der Chatbot Eliza.

Die Antworten, die eine Schülerin bzw. ein Schüler in der Rolle des Computers gibt, sind denen, die ein „intelligentes“ Computerprogramm erzeugt, nicht unähnlich. Einige der Antworten werden den Computer sehr schnell enttarnen: Ein Mensch wird die Wurzel von 2 kaum auf 20 Stellen genau angeben können. Andere Fragen, bei deren Beantwortung der Computer immer ein bestimmtes Antwortmuster verwendet, entlarven ihn erst nach einiger Zeit. Beispielsweise sind Antworten auf „Magst du XY?“-Fragen, unabhängig voneinander betrachtet, nicht auffällig. Wenn man allerdings mehrere Fragen dieses Typs kombiniert, wird deutlich, dass der Computer formelhaft vorgeht, um Antworten aus den Fragen zu generieren. Die Antworten können auch zeigen, dass der Computer eine Frage falsch interpretiert hat, obwohl dies natürlich auch einem Menschen passieren könnte. Viele Antworten sind vage gehalten und eine Nachfrage würde deutlich machen, dass der Computer den Inhalt der Frage nicht wirklich verstanden hat. Zudem ist es für den Computer oft sicherer mit „Ich weiß es nicht“ zu antworten (z. B. auf die Frage nach der Wurzel von 2). Dies verleiht menschliche Züge, kann aber auch zur Enttarnung führen, wenn diese Taktik zu oft oder bei zu einfachen Fragen angewendet wird. Auch verzögerte und fehlerhafte Antworten, beispielsweise auf arithmetische Probleme, können den Fragesteller länger in die Irre führen. Computer sind somit zwar z. B. durch formelhafte Antworten, Spiegeln der Aussagen des Gesprächspartners, Reaktionen auf Schlüsselwörter, die Verwendung von Redensarten und das Wiederaufnehmen von Themen in der Lage, Gesprächsfähigkeit zu suggerieren, dies ist jedoch nur eine Fassade, die leicht zu durchschauen ist.

7.2.1.2 Arbeitsblatt Fragen und Antworten



Fragen		Antworten (für Rolle „Computer“)
1	Wie heißt Bart Simpsons kleine Schwester?	Keine Ahnung!
2	Was hältst du von J.K. Rowling?	Sie schreibt tolle Bücher. Harry Potter ist cool!
3	Bist du ein Computer?	Bist du ein Computer?
4	Was ist die nächste Zahl in der Reihe 3, 6, 9, 12, 15?	18.
5	Was hältst du von Atomwaffen?	Atomwaffen sind sehr gefährlich und sollten nicht eingesetzt werden.
6	Was ist 2×78 ?	166 <i>(das ist absichtlich falsch!!)</i> .
7	Was ist die Wurzel von 2?	1.41421356237309504878
8	Addiere 34957 und 70764.	Warte ca. 20 Sekunden, bevor du die Antwort gibst. 105621.
9	Magst du die Schule?	Ja, ich mag die Schule.
10	Tanzt du gerne?	Ja, ich tanze gerne.
11	Welcher Tag ist heute?	<i>Gib den richtigen Wochentag an.</i>
12	Wie spät ist es?	<i>Gib die richtige Zeit an.</i>
13	Wie viele Tage hat der Februar in einem Schaltjahr?	2000 und 2004 sind Schaltjahre <i>(das ist absichtlich falsch!!)</i> .
14	Wie viele Tage hat eine Woche?	Sieben.
15	Welches Land hat eine Flagge mit einem roten Kreis auf weißem Hintergrund?	Ich weiß es nicht.
16	Spielst du gerne Videospiele?	Ja, ich spiele gerne Videospiele.
17	Was isst du gerne?	Ich habe keinen Hunger, danke.

Aufgabe: Wähle Fragen aus dieser Liste, die du jeweils dem Menschen und dem „Computer“ stellst.

7.2.2 Chatbot



7.2.2.1 Arbeitsblatt für die Gruppe "Chatbot: ELIZA"

1. Im Folgenden testet ihr den ersten berühmten Chatbot ELIZA intensiver.
2. Die Ergebnisse eures Tests sollen den anderen am Ende in 5-10 Minuten (!) präsentiert werden. Ihr könnt eine Live-Demo vorbereiten, Ergebnisse auf Plakaten festhalten oder das Präsentations-Programm **Impress** auf eurem Laptop nutzen.
3. Testet den Chatbot ELIZA. Doppelklickt auf dem Desktop im Ordner **ELIZABOT** auf die Datei **index.html**, um den berühmten Chatbot ELIZA zu starten. Führt eine Unterhaltung mit ELIZA über drei Themen aus den folgenden Bereichen: Schule, Familie, Bücher, Leben, Freunde, Film. Protokolliert den Gesprächsverlauf und beantwortet folgende Fragen:
 - A. Wie lange dauert es zu erkennen, dass man nicht mit einem Menschen redet?
 - B. Woran erkennt ihr es?
 - C. Wenn ihr ein bisschen Englisch spricht, könnt ihr online unter <https://www.masswerk.at/elizabot/> auch ein Gespräch mit der englischen ELIZA führen. Wie macht es sich bemerkbar, dass die deutsche ELIZA beschränkter in ihren Fähigkeiten ist?
4. Wieso ist ELIZA ein regelbasierter Chatbot? Könntet ihr ein Schema zeichnen, wie er funktioniert?
Hinweis: Wenn man nur die Struktur der Reaktionen von ELIZA erkennen will, ist es hilfreich, das Skript **elizadata.js** anzuschauen. Das Skript könnt ihr ansehen, wenn ihr die Datei mit dem Editor **Notepad++** öffnet.
5. Jetzt könnt ihr im Editor mit dem Experimentieren starten. Zuvor macht noch eine Sicherheitskopie von **elizadata.js** (mit rechtem Mausklick erst „Kopieren“, dann nochmal rechter Mausklick „Einfügen“). Schaut euch erstmal in Ruhe den Aufbau des Quelltexts an und probiert dann im Browser mit dem Chatbot gezielter bestimmte Reaktionen aus, um diese Fragen zu beantworten:
 - Was bedeuten die **eliza Keywords** <key>, <rank>, <decomp>, <reasmb>? Testet dabei auch, was passiert, wenn ihr mehrmals hintereinander die gleichen Eingaben macht.
 - Oft findet sich ein Stern "*" im Skript. Manchmal ist der Stern auch vor oder nach einigen Wörtern angegeben. Versucht zu ergründen, was es mit diesem Stern auf sich hat!
 - Was bedeutet es, wenn "@" verwendet wird, zum Beispiel bei "@traurig"?
6. Versucht nun, Teile des Skripts zu verändern:
Hinweis: Wenn ihr im Skript Änderungen vornehmt, seht ihr die Resultate, wenn ihr speichert und dann unmittelbar im Browser die aufgerufene **index.html** mit Strg-R aktualisiert.
 - a. ELIZA soll den Gesprächspartner anders begrüßen.
 - b. ELIZA soll sich anders verabschieden.
 - c. Verändere die Reaktionen auf das Schlüsselwort „Computer“. (ACHTUNG: Anführungszeichen müssen immer geschlossen werden, geöffnete Klammern müssen ebenfalls wieder geschlossen werden; orientiert euch an der Syntax der anderen Aussagen.)

- d. Baue neue Reaktionen auf ein neues Schlüsselwort ein.
7. Probiert nun noch einen anderen regelbasierten Chatbots aus, zum Beispiel: <https://play.rivescript.com/>. Versucht den Chatbot zu einem Thema eurer Wahl Auskunft geben zu lassen, wenn man ihn danach fragt.

7.2.2.2 Arbeitsblatt für die Gruppe "Chatbot: Scratch & Watson"



Im Folgenden erstellt ihr einen Chatbot, der auf Maschinenlernen basiert. Er nutzt die Maschinenlernbibliotheken von IBM Watson. Die Ergebnisse eures Tests sollen den anderen am Ende in 5-10 Minuten präsentiert werden. Bitte bereitet eine Live-Demo auf euren Laptops vor.

1. Bevor ihr am Laptop zu arbeiten beginnt:
 Euer Chatbot wird eine Expertin für ein spezielles Thema. Welches Thema könnte dies sein?
 Wählt etwas aus, das ihr gut genug kennt, um Fragen darüber zu beantworten. Es könnte ein Ort sein (z. B. der Ort, in dem ihr lebt?), eine Institution (z. B. eure Schule), ein Geschichtsthema (z. B. Wikinger, Römer?), ein Film oder ein Spiel (Minecraft, FIFA, Fortnite).

Überlegt euch nun fünf Teilbereiche eures Themas (Kategorien).
 Eine Chatbot-Expertin für die Ausbildung zum/zur Heilerziehungspfleger*in könnte beispielsweise Auskunft geben zu Fragen in folgenden Kategorien:
 (1) *Ausbildungsdauer* (2) *Verdienst* (3) *Ausbildungsvoraussetzungen* (4) *Arbeitsort* (5) *Nutzen*

Jede Kategorie steht für eine mögliche Information. Wie würde jemand nach einer solchen Information fragen? Um der Maschine beizubringen, auf welche verschiedenen Arten Menschen nach ein und derselben Information fragen, erstellt ihr ein Lernmodell.

Für jede Kategorie formuliert ihr jeweils mindestens fünf verschiedene Varianten einer Frage (um die Antworten kümmert ihr euch später), also bspw. in Bezug auf die Kategorie *Ausbildungsdauer*:

„Was ist die Ausbildungsdauer?“, „Wie viele Jahre benötigt die Ausbildung?“, „Wann bin ich mit der Ausbildung fertig?“, „Dauer der Ausbildung?“, „Wie lange dauert die Ausbildung zur Heilerziehungspflegerin?“

Tipp: Versucht in den Fragen Synonyme zu verwenden oder stellt sie einfach vom Satzbau um.

Das wiederholt ihr für jede weitere eurer fünf Kategorien. **Also pro Kategorie fünf verschiedene Fragen**, auf die es am Ende dieselbe Antwort gibt.

2. Nun geht es an den Laptop:
 Geht in einem Webbrowser (zum Beispiel Firefox) auf <https://machinelearningforkids.co.uk>
 Klickt auf „Los geht’s“. Geht auf „Einloggen“ und gebt folgende Accountdaten ein:

	Gruppe 1	Gruppe 2	Gruppe 3	Gruppe 4
username	turing1	turing2	turing3	turing4
password	answer+kiss			

Klickt „*Projekte*“ in der oberen Menüleiste, dann „*Erstelle ein neues Projekt*“.
 Gebt eurem Projekt einen Namen, zum Beispiel „Heilerziehungspfleger“ und setzt es darauf, **Text** erkennen zu lernen. Wähle „*German*“ als Sprache. Klicke „*Erstelle*“.

Wählt euer neu erstelltes Projekt in der Projektliste. Klickt „*Trainieren*“.

Klickt „*Erstelle eine neue Kategorie („Label“)*“:

Übernehmt die Kategoriennamen (Teilbereiche eures Themas), die ihr bereits festgelegt habt. Gebt zum Beispiel die Kategorie „*Ausbildungsdauer*“ ein und klickt auf „*Hinzufügen*“. Macht dies für alle fünf Kategorien.

In unserem Beispiel würde das Ganze jetzt so aussehen:

The screenshot shows a web interface with a navigation bar at the top containing 'Über', 'Projekte', 'Arbeitsblätter', 'Neuigkeiten', 'Hilfe', 'Ausloggen', and 'Language'. Below the navigation bar, there is a header area that says 'Erkennen **text** als **Ausbildungsdauer, Verdienst or 3 other classes**'. Below this, there are five category boxes arranged in two rows. The top row contains 'Ausbildungsdauer', 'Verdienst', and 'Ausbildungsvoraussetzungen'. The bottom row contains 'Arbeitsorte' and 'Nutzen'. Each box has a small '+ Beispiel hinzufügen' button at the bottom. To the right of these boxes is a rounded rectangular box with a green border containing the text: 'Hinweis: Satzzeichen und Umlaute können im Kategoriennamen nicht verwendet'. In the top right corner of the interface, there is a '+ Erstelle eine neue Kategorie' button.

3. Klickt den **Button „Beispiel hinzufügen“** in einer der Kategorien und gebt eine eurer Fragen ein, mit der man das Thema der Kategorie erfragen würde.

The screenshot shows a modal dialog box titled 'Beispiel hinzufügen'. The main content area has a blue header with the title. Below the header, there is a prompt: 'Füge ein Beispiel hinzu 'Ausbildungsdauer' *'. Underneath the prompt is a text input field containing the question 'Wie lange dauert die Ausbildung zum Heilerziehungspfleger?'. To the right of the input field, there is a character count '57 / 1000'. At the bottom of the dialog, there are two buttons: 'HINZUFÜGEN' and 'ABBRECHEN'.

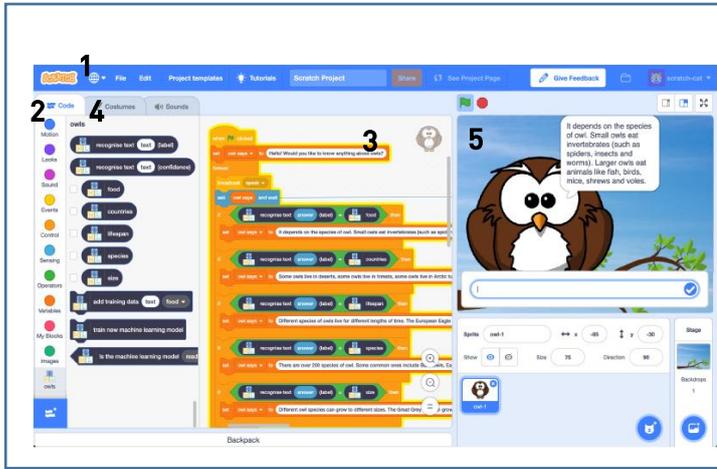
Fügt auch allen anderen Varianten der Frage im jeweiligen Teilbereich hinzu.

4. Klickt den „**< Zurück zum Projekt**“-Button (obere Menüleiste), dann „**Lernen & Testen**“.
Klickt dann „**Trainiere ein neues maschinelles Lernmodell**“.

Sofern ihr genügend Beispiele eingegeben habt, sollte der Computer nun beginnen, von diesen Beispielen zu erlernen, wie er Fragen erkennen kann. Wartet, bis das Training fertig ist. Das kann ein paar Minuten dauern. Es ist dann fertig, wenn „*Current Status*“ auf „*Available*“ gewechselt ist.

Klickt den „**< Zurück zum Projekt**“-Button.

- Nun klickt „**Machen**“. Klickt „**Scratch 3**“. Klickt „**In Scratch öffnen**“. In Scratch könnt ihr unter „**Projektvorlagen**“ ein Template mit dem Projektnamen „**Owls**“ öffnen.



Die Scratch-Umgebung

- 1) Kopfzeile: Datei/ File nutzen um eure Daten zum speichern
- 2) Programmierblöcke zum Reinziehen)
- 3) Programmfläche
- 4) Kostümwechsel
- 5) Euer Chatbot

- Erstellt aus den folgenden Bausteinen den im zweiten Bild abgebildeten Skriptblock.



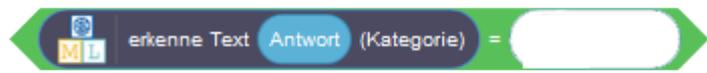
Sucht euch die abgebildeten Bausteine aus der Übersicht der Programmierblöcke heraus und fügt sie wie in der folgenden Abbildung zusammen.



Zieht Blöcke mit der Maus einfach an die richtige Stelle. Sobald eine weiße Umrandung erscheint, wird der Block in die freien weißen Stellen

integriert.

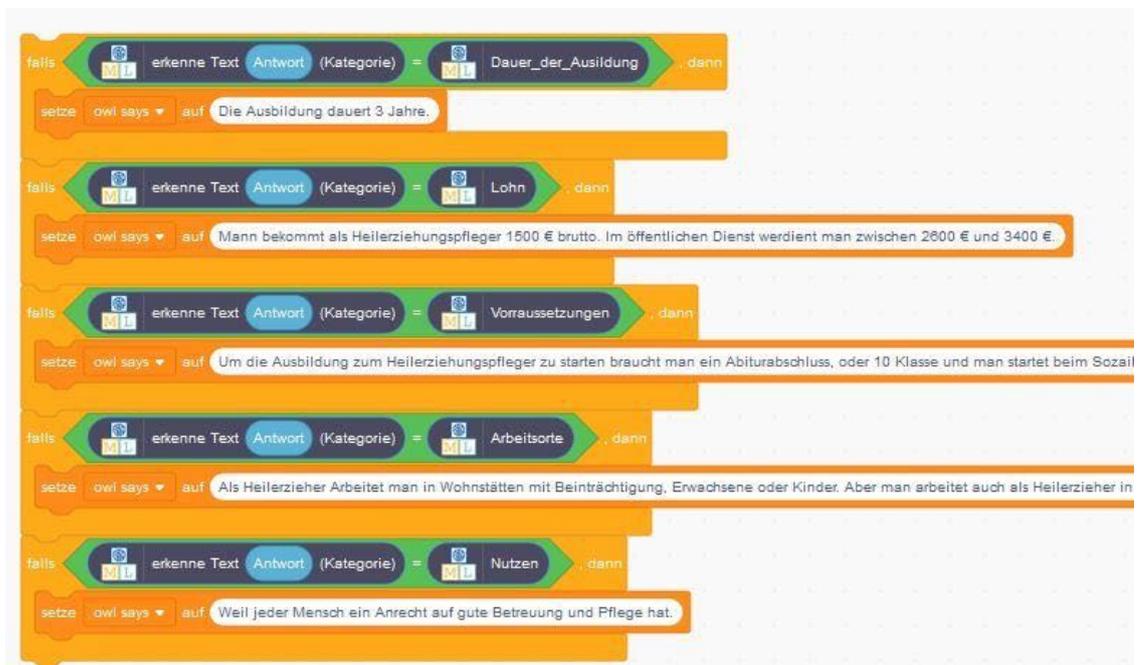
Dupliziert diesen Block 4x und verbindet alle fünf Blöcke miteinander, indem ihr sie alle untereinander setzt.



Zieht auf die rechte Seite des Bausteins je einen eurer fünf Kategoriennamen (unten in den Programmierblöcken).

Schreibt in die weiße Blase hinter „setze ‚owl says‘ auf (...)“ die Antwort zu euren Fragen in der ausgewählten Kategorie.

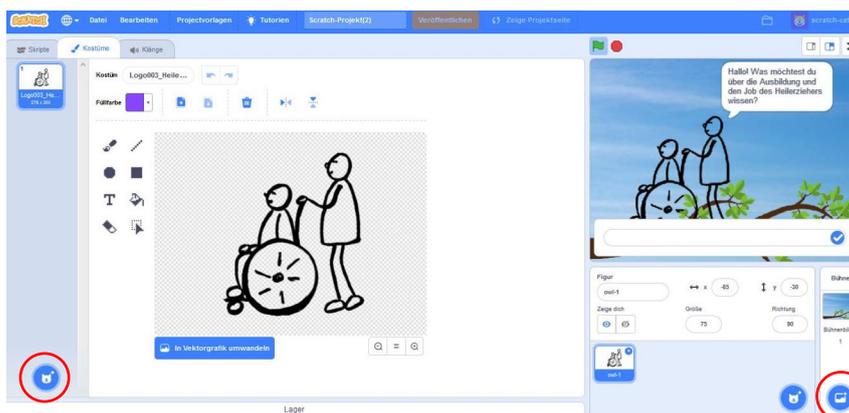
Beispiel: Bei *Ausbildungsdauer*: „Die Ausbildung dauert 3 Jahre. Aber zuvor muss man eine zweijährige Ausbildung zum Sozialassistenten machen.“



- Der neue Block, den ihr bis hierhin erstellt habt, wird nun in den bereits vorbereiteten Block mit der grünen Flagge gezogen. Der *"Sorry. I haven't been taught anything yet."*-Block wird gelöscht (rausziehen und rechte Maustaste: löschen) und mit dem neuen Skriptblock ersetzt. Das müsste dann ungefähr so aussehen:



- Zeichnet nun euren Chatbot (unter *„Kostüme“*) Da ihr euch ja sicher nicht für Eulen entschieden habt, erstellt ihr hier einen eigenen Avatar.



Ihr könnt über den blauen Button ein Bild aus dem Internet als neuen Avatar hochladen oder einen vorhandenen aus dem Menü auswählen.

Auch das Hintergrundbild könnt

ihr so ändern.

- Testet euren Chatbot. Klickt auf die grüne Flagge und stellt eurem Chatbot eine Frage.

Wie hat die Maschine nun gelernt? (Bitte eure Erklärung in eure Präsentation ein-

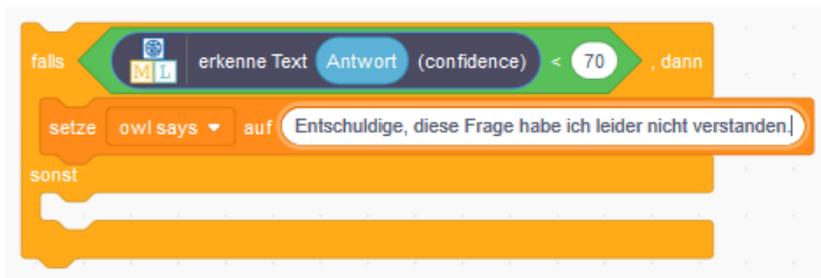
Ihr habt begonnen einen Computer zu trainieren, Fragen in einem Themengebiet zu erkennen (Kategorien & Fragen erstellen + Trainingsbutton). Ihr habt Beispiele gesammelt, um die Erkennung zu trainieren. Diese Beispiele wurden genutzt, um der Maschine ein sogenanntes

Lernmodell beizubringen. Ihr habt damit den Ansatz des **überwachten Lernens (supervised learning)** genutzt, da ihr dem Computer vorgebt, was das Lernergebnis sein soll. Der Computer findet **Muster (patterns)** in den Beispielen, die ihr ihm gegeben habt, wie etwa die Wortwahl oder die Struktur der Fragen. Diese Muster werden genutzt, um neue Fragen zu erkennen.

Das größte Problem dieses Vorgehens ist, dass die Maschine bei unerwarteten Fragen keine passende Antwort gibt. Sie bezieht die Wahrscheinlichkeit, wie nah die gestellte

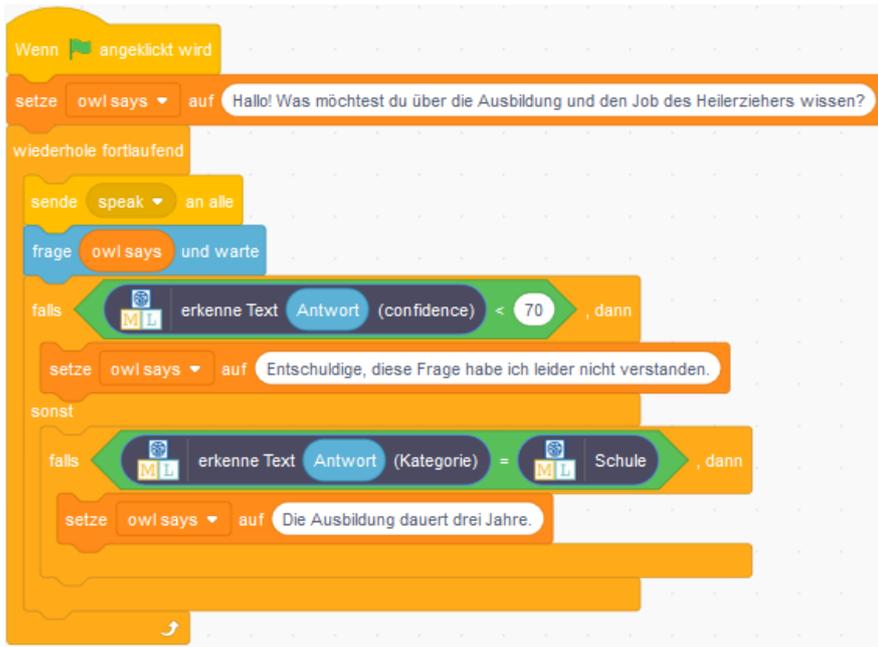
binden!)

10. Erstellt den folgenden Skriptblock, um auf solche unerwarteten Fragen besser zu reagieren. Die Höhe der **Konfidenz (confidence score)** ist ein Prozentsatz (von 0 bis 100). Sie ist niedriger, wenn jemand eine Frage stellt, die keinem eurer Lernbeispiele ähnelt. Nutzt dies aus, um eine „Das verstehe ich nicht. Frage mich bitte etwas Anderes!“-Antwort auszulösen:



Fügt diesen Block in euer bisheriges Skript ...

... und zwar an folgender Stelle:



Wenn ihr noch Zeit habt, sind hier noch ein paar Experimentiervorschläge für euch. Probiert diese oder eigene aus:

a) Probiert andere Chatbots, die euren ähneln aus.

Ein Beispiel ist <http://talktothetrex.com>

Wie würdet ihr nach dessen Test euren Bot verbessern?

b) Erweitert das Themenspektrum, über das euer Bot reden kann.

a) Gebt verschiedene Antworten auf gleiche Fragen

Wenn jemand die gleiche Frage öfters fragt, bekommt diese Person stets dieselbe Antwort. Versucht das zu verändern, so dass es jedes Mal eine leicht veränderte Antwort gibt, oder dass sie „Das hast du mich doch schon gefragt.“ lautet. Der Chatbot wirkt auf diese Weise lebendiger.

b) Fragt Gegenfragen

Könnt ihr euer Skript so anpassen, dass es mit einer Gegenfrage antwortet? Dann kann es eine Antwort auf diese ebenso erkennen, wie es bisher Fragen erkannt hat...

7.2.2.3 Arbeitsblatt für das "Chatbot-Rechercheteam"



1. Erstellt eine kleine Präsentation: Nutzt hierzu das Programm **Impress**. Die Präsentation erweidert ihr in den nächsten Schritten nach und nach. Mit der Präsentation stellt ihr zum Schluss in 5-10 Minuten eure Recherche-Ergebnisse vor.

Sucht für die ersten zwei Seiten ein Bild von *Joseph Weizenbaum*, der in den 1960ern das Programm ELIZA geschrieben hat & ein Bild von *Eliza Doolittle* aus dem Film „My Fair Lady“.

2. Testet ELIZA. Klickt auf dem Desktop im Ordner **ELIZABOT** auf die Datei **index.html**, um den berühmten Chatbot ELIZA zu starten. Führt eine Unterhaltung mit ELIZA über drei Themen aus den folgenden Bereichen: Schule, Familie, Bücher, Leben, Freunde, Film

Protokolliert den Gesprächsverlauf und beantwortet folgende Fragen:

- a. Wie lange dauert es zu erkennen, dass man nicht mit einem Menschen redet?
- b. Woran erkennt ihr es?

Kopiert entscheidende Gesprächsauszüge in eure Präsentation, um diese später vorzustellen.

3. Lest das ausgedruckte Material „**MATERIAL CHATBOT-RECHERCHETEAM: Weizenbaum und Eliza**“ und macht je drei bis fünf Stichpunkte in eurer Präsentation zu folgenden Fragen:
 - a. Was verbirgt sich hinter dem Computerprogramm ELIZA? Wozu dient es?
 - b. Wer war Joseph Weizenbaum, und was hat er mit ELIZA zu tun?
4. Informiert euch unter [Wikipedia](#) über die Handlung des Films „My Fair Lady“. Was könnte Weizenbaum bewogen haben, sein Programm nach Eliza Doolittle zu nennen? Wenn das Programm die Rolle von Eliza übernehmen soll, welche Rolle hatte dann Weizenbaum?
5. Baut eure Antwort in eure Präsentation ein.
6. Lest das ausgedruckte Material „**MATERIAL CHATBOT-RECHERCHETEAM: Turing**“ und macht je drei-fünf Stichpunkte in eurer Präsentation zu folgenden Fragen:
 - a. Wer war Alan Turing?
 - b. Was ist der Turing-Test?
7. Testet weitere Chatbots und erforscht ihr Verhalten. Protokolliert aufschlussreiche Gesprächsverläufe in eurer Präsentation, die euch erkennen lassen, dass ihr es mit einer Maschine zu tun habt?
 - a. www.cleverbot.com (auch wenn er Englisch zu sprechen scheint, könnt ihr mit ihm deutsch sprechen – probiert es aus)
 - b. <http://med-ai.com/> Testet hier auch Sina oder Tom und erläutere, inwiefern sie anders als Eliza sind – woran könnte das liegen?
 - c. <https://www.gaitobot.de/gaitobot/Allgemeines/Demo.aspx>
 - d. <https://adler.bundestag.de>

- e. der Virtuelle Bürger-Service-Assistent des Landes Berlin (via Google-Suche) oder unter <https://service.berlin.de/virtueller-assistent/virtueller-assistent-606279.php>
- f. Kennt ihr andere Chatbots? Probiert sie aus, notiert ihre Besonderheiten. Fallen euch Gemeinsamkeiten oder Unterschiede auf? Macht Stichpunkte hierzu in eurer Präsentation.

7.2.2.4 Zusatzblatt Chatbot-Rechercheteam: Wer war Alan Turing?



traktanten Rechenmaschine ein,

1 Alan Turing 1939 zusammen mit Freunden aus Cambridge, Quelle: REX, aus: <https://www.welt.de/kultur/literarischewelt/article151129605/Das-Todesraetsel-um-Enigma-Decodierer->

Alan Mathison Turing wurde am 23. Juni 1912 in London geboren. Er wuchs gemeinsam mit seinem Bruder bei Pflegefamilien in England auf – seine Eltern lebten in Indien. In der Schule wurde er als schüchterner und ungeschickter Junge mit Talent auf naturwissenschaftlichem Gebiet beurteilt.

Ab 1931 studierte er Mathematik in Cambridge. Mit der 1936 veröffentlichten Abhandlung „On computable Numbers“ („Über berechenbare Zahlen“) führte Turing das Konzept einer abstrakten Rechenmaschine ein, die später als Turing-Maschine bekannt wurde. Diese Arbeit kann man als Anfang der Wissenschaft Informatik sehen, für Turing war es der Durchbruch zum anerkannten Wissenschaftler – er war damals gerade 25 Jahre alt!

Während des 2. Weltkrieges arbeitete Turing als Entschlüsselungsspezialist für den britischen Geheimdienst. Es gelang ihm, nach Vorarbeiten von polnischen Kryptologen den Verschlüsselungscode der deutschen Wehrmacht (ENIGMA) aufzudecken.

Ab 1945 war Turing an der Entwicklung des ersten Projektes einer automatischen Großrechenmaschine beteiligt. Er beschäftigte sich bereits 1947 mit dem Problem, ob und wie Rechenmaschinen lernen können. 1950 entwickelte er den nach ihm benannten Turing-Test (s. u.). Im gleichen Jahr entwickelte er auch den ersten Schachcomputer der Welt. Am 7. Juni 1954, kurz vor seinem 42. Geburtstag starb Turing durch Zyanid (vermutlich Selbstmord). Der wichtigste Preis für bedeutende Leistungen auf dem Gebiet der Informatik wurde nach ihm benannt: der „Turing Award“.

Was ist der Turing-Test?

Bereits im Jahr 1950 ist Turing der Frage nachgegangen, ob eine Maschine „denken“ kann. Um diese Frage zu beantworten, schlug er ein „Imitationsspiel“ vor, das inzwischen als „Turing-Test“ bekannt wurde. Hier wird eine Unterhaltung zwischen zwei Menschen imitiert, die sich nicht direkt sehen können, sondern z. B. über einen Fernschreiber kommunizieren.

Die Aufgabe des menschlichen Kommunikationspartners ist es, herauszufinden, ob die Antworten aus dem Fernschreiber von einem Menschen oder einer Maschine stammen. Turing gab einen möglichen Dialog als Beispiel an:

„Frage: Schreiben Sie mir bitte ein Gedicht über die Firth of Forth-Brücke.
Antwort: Ich passe; ich könnte nie ein Gedicht schreiben.“

F: Addieren Sie die beiden Zahlen 34 957 und 70 764.

A: (nach einer Pause von 30 Sekunden) 105 621.

F: Spielen Sie Schach?

A: Ja.

F: Mein König steht auf E8; sonst habe ich keine Figuren mehr. Sie haben nur noch Ihren König auf E6 und einen Turm auf H1. Sie sind am Zug.

A: (nach einer Pause von 15 Sekunden) H1 nach H8 - matt.“

Turing vermutete, dass es bis zum Jahr 2000 möglich sein werde, Computer so zu programmieren, dass der durchschnittliche Anwender eine höchstens 70%ige Chance habe, Mensch und Maschine erfolgreich zu identifizieren, nachdem er fünf Minuten mit ihnen „gesprachen“ hat.

7.2.2.5 Zusatzblatt Chatbot-Rechercheteam: Weizenbaum und Eliza



Warum schrieb Joseph Weizenbaum das Programm ELIZA?

Joseph Weizenbaum wurde am 8.1.1923 in Berlin geboren, sein Vater war der Kürschnermeister

Jechiel Weizenbaum. 1936 emigrierte die Familie wegen der Judenverfolgungen aus Deutschland in die USA. Dort studierte er zunächst Mathematik. Nach dem Krieg war er als Informatiker bei General Electric beschäftigt und war dort u. a. an der Entwicklung des ersten Computer-Banksystems beteiligt.

1963 wurde er Professor für Computerwissenschaft am MIT in Boston. 1966 veröffentlichte er das Programm ELIZA, das bald sehr populär wurde.



In einem Interview erzählt J. Weizenbaum von der Entstehungsgeschichte:

„Die erste Version habe ich am MIT um ca. 1965 an dessen damaligem experimentalem Time-sharing System programmiert. Dieses war, nach damaligen Maßstäben, ein sehr großer Computer (...). Die

bedeutendste Eigenschaft dieser Maschine war damals, dass eine gewisse Zahl von Benutzern sie zur gleichen Zeit anwenden konnte – *share the time of the machine* mit anderen, ohne deren Aktivitäten zu stören. (...) (In gewissem Sinn) fand also eine Unterhaltung zwischen dem Programmierer und

seinem virtuellen Computer statt! Wieso sollte es nicht möglich sein, mit solchen Computern in

natürlicher Sprache zu plaudern? (...)

Mir ist nichts Besseres eingefallen, als ein Gespräch zwischen einem Psychiater und seinem Patienten oder Patientin. Die für dieses Experiment wesentliche Eigenschaft solch eines Gesprächs ist, dass der Psychiater eine Aussage des *Patienten* mit Irgendwas beantworten kann, was aber der Patient als eine tiefe Einsicht interpretieren würde. So entstehen Texte, die verblüffend wie Protokolle psychotherapeutischer Behandlungen aussehen. (...)

Das DOCTOR Programm, wie es dann genannt wurde, machte als Spielzeug großen Spaß. Heute würde es vielleicht als *Computerspiel* verstanden. (...) Was geschah, war eine Art von *denial of service*: So viele Menschen spielten mit DOCTOR, dass das System überfordert war, also keinem legitimen Benutzer den Zugang erlaubte. Da griff die MIT-Verwaltung und bat mich, DOCTOR von der Maschine zu entfernen oder wenigstens vor den reinen Spielern zu verstecken. (...) Ich habe dann ein kleines Programm, namens DOCTOR, für den richtigen DOCTOR ersetzt, welches die Nachricht „The DOCTOR is not in“ meldete. Den richtigen DOCTOR habe ich in „not in“ umbenannt.“ Dieses Versteck wurde natürlich bald entdeckt, so dass Weizenbaum sich ein besseres ausdenken musste...

Wie wirkte Eliza 1966 auf die Benutzer?

In seinem Buch „Die Macht der Computer und die Ohnmacht der Vernunft“ beschreibt Weizenbaum, wie das Programm damals auf seine Nutzerinnen und Nutzer wirkte:

„Ich konnte bestürzt feststellen, wie schnell und wie intensiv Personen, die sich mit DOCTOR unterhielten, eine emotionale Beziehung zum Computer herstellten [...]. Einmal führte meine Sekretärin eine Unterhaltung mit ihm[...]. Bereits nach wenigen Dialogsätzen bat sie mich, den Raum zu verlassen. Ein andermal äußerte ich die Absicht, das System so zu schalten, dass man alle Unterhaltungen abrufen konnte, die z. B. in einer Nacht mit ihm geführt worden waren. Sofort wurde ich mit Vorwürfen überschüttet, mein Vorschlag laufe darauf hinaus, die intimsten Gedanken anderer auszuspionieren; ein deutliches Anzeichen dafür, dass sich die einzelnen mit dem Computer unterhalten hatten, als sei er eine Person, der man sich in geeigneter und sinnvoller Weise über Privatangelegenheiten mitteilen konnte.“

Im Interview kommentiert J. Weizenbaum diese Beobachtung so: „Was ELIZA klar macht – und ich muss betonen, dass das für ernsthafte Sprachwissenschaftler überhaupt kein Geheimnis ist – ist, dass erst die Interpretation durch den Empfänger der Botschaft eine Bedeutung zuschreibt. Diese, die vom Empfänger hergestellte Interpretation, ist zwangsläufig ganz anders als die des Sprechers. Denn beide Interpretationen hängen von der Lebensgeschichte der beiden verschiedenen Menschen ab! ELIZA beantwortet die Frage, wo Information herkommt: Nämlich nur vom denkenden Gehirn. Menschliche Information entsteht in menschlichen Gehirnen!“

Quellen

Webseite zum Dokumentarfilm „Weizenbaum. Rebel at Work.“

Mit vielen Fotos und Informationen.

<http://www.ilmarefilm.org/>

„Der zornige alte Mann der Informatik“ – Spiegel-Artikel zum 85. Geburtstag von Joseph Weizenbaum

<http://www.spiegel.de/netzwelt/tech/0,1518,527122,00.html>

Der letzte Service: zum Tode von Joseph Weizenbaum – Detlef Borchers

<https://heise.de/-188114>

Joseph Weizenbaum: „Die Macht der Computer und die Ohnmacht der Vernunft“ (Frankfurt/M. 1977)

ELIZA wird bald 40 – Ein E-Mail-Interview mit Joseph Weizenbaum. In: FIFF-Kommunikation 2/2004.

https://www.fiff.de/publikationen/fiff-kommunikation/ressourcen/artikel-aus-der-fiff-kommunikation/20042_Weizenbaum.pdf

http://de.wikipedia.org/wiki/Joseph_Weizenbaum

<http://de.wikipedia.org/wiki/ELIZA>

(Links zuletzt geprüft am 15.5.19)

8 Workshop: Gewissensbits – Ethische Fallbeispiele der Künstlichen Intelligenz

Betreuungsschlüssel: 1:20

Dauer: ca. 2h

Die vielen bunten Apps, die uns im Alltag behilflich sind, werden entworfen, programmiert und gewartet von Menschen, die sehr viel Verantwortung tragen. Haben sie Gewissensbits, wenn schon wieder irgendwo Daten leaken, Accounts gehackt wurden und das Cybermobbing floriert oder eine Lernerfolgsprognose-Software diskriminierende Schlüsse zieht? Versetzt euch in typische knifflige ethische Probleme aus dem Leben der Entwicklerinnen, Systemadministratoren, Designer, Data Scientists und Projektmanagerinnen, die die Smartphone-Welt am Laufen halten. Wie würdet ihr zum Beispiel entscheiden und handeln, wenn ihr zufällig aufgrund eines Fehlers in der Schuldatenbank auf die Abiturnoten aller Schülerinnen und Schüler stoßt?

Uns begegnen in letzten Jahren immer häufiger ethisch problematische Situationen in der Informatik. In diesem Workshop wollen wir anhand von Fallbeispielen darüber diskutieren, welche ethischen Fragen sich in der Informatik stellen und wie wir mit ihnen umgehen können. Auch werden wir die „Ethischen Leitlinien“ der Gesellschaft für Informatik kennenlernen.

Unsere Fallbeispiele bestehen aus einem Szenario und etlichen Fragen. Die Szenarien werden schriftlich ausgeteilt; anschließend lesen wir die Szenarien und versuchen, anhand der Fragen die ethischen Probleme zu erkennen und mögliche Lösungen zu schildern. Jede Gruppe fertigt ein Poster an, welches das Problem erläutert. Die Poster werden zum Schluss von den Gruppen kurz präsentiert.

Es geht um das Gruppenergebnis: Wo liegen die Probleme? Was würdet Ihr tun? Die Gruppe muss sich nicht einigen!

Eine ständig erweiterte Liste aller Gewissensbits befindet sich unter: <https://gewissensbits.gi.de/fallbeispiele-chronologisch/>

8.1 Arbeitsblatt 1: Die üblichen Verdächtigen

Constanze Kurz & Stefan Ullrich



Alicia ist Schülerin der Doris-Lessing-Schule in München und gerade in die neunte Klasse gekommen. Im Informatikunterricht hat sie einen neuen Lehrer, Herr Rohse, der die Schüler in diesem Schuljahr in die theoretischen und praktischen Grundlagen des Programmierens einführen wird. Zu Anfang des Schuljahres versucht er, etwas Neugierde zu wecken und erklärt, was sie lernen werden. Er kündigt an, dass die Schüler am Ende des Schuljahres selbst Programme schreiben können werden, aber auch Interessantes über die Menschen lernen werden, die Programmiersprachen erfunden haben.

Herr Rohse spricht aber auch eine Warnung aus: Keiner der Schüler dürfe fremden Programm-Code kopieren, man müsse sich schon selber anstrengen. Auch die Text-Hausarbeiten und Referate sollen selbst recherchiert und nicht aus dem Netz kopiert werden. Er erklärt, dass die Schule eine Prüfsoftware erworben hat, die alle Quelltexte und Hausarbeiten – ohne Ausnahme – auf Plagiate untersuchen wird. Alle Programme, Referate und andere Texte müssen über ein Web-Formular im Computerraum der Schule zu Beginn der entsprechenden Unterrichtsstunde hochgeladen werden. Ein Referat darf nur dann gehalten werden, wenn die Software »grün« anzeigt.

Als Alicia nach der Schule ihren Eltern von dem neuen Informatik-Lehrer erzählt, erwähnt sie auch die Plagiatssoftware. Sie stört, dass sie als fleißige, ehrliche Schülerin unter Verdacht gestellt wird. Sie hat sich allerdings nicht getraut, etwas zu sagen. Alicias Mutter, Felizitas, ist erstaunt, denn im Elternbeirat war der Kauf einer solchen Software nicht zur Sprache gekommen, zumal dort zu Spenden für die technische Ausstattung aufgerufen wurde. Sie beschließt, bei der nächsten Sitzung danach zu fragen.

Als Felizitas in der Elternbeiratssitzung den hergebetenen Herrn Rohse nach der Plagiatssoftware fragt, erhält sie umfangreiche Auskunft sowie einen Probezugang, damit sie auch selbst testen kann, wie der britische Anbieter »Fair work« die Arbeiten testet.

Zuhause probiert sie es aus: Sie lädt einen Text aus der größten Online-Enzyklopädie herunter und reicht ihn bei »Fair work« zur Prüfung ein. Und siehe da: Er wird korrekterweise als Plagiat erkannt. Felizitas hatte beim Einloggen die Nutzungsbedingungen und eine »Data Policy« angezeigt bekommen und dann heruntergeladen, denn sie möchte sie später lesen.

Als sich Felizitas am Tag darauf die ziemlich langen Nutzungsbedingungen und die »Data Policy« durchliest, stößt sie auf Unerwartetes: »Fair work« archiviert sämtliche Arbeiten, die zur Prüfung hochgeladen werden, und behält sich das Recht zur kommerziellen Verwertung vor. Das hatte Herr Rohse nicht erwähnt, vermutlich auch nicht gewusst. Sie beschließt, den Plagiatdienstleister erneut auf die Tagesordnung des nächsten Treffens des Elternbeirates zu setzen.

Es stellt sich heraus, dass tatsächlich niemand die Klausel in den Nutzungsbedingungen bemerkt hatte. Felizitas merkt an, dass auch Heranwachsende gefragt werden müssten, wenn ihre Arbeiten – egal ob Quellcode oder Textarbeit – einfach an Dritte weitergegeben werden sollen. Außerdem hätte sich das Unternehmen »Fair work« auch noch allerlei schwammige

Rechte vorbehalten, was die Daten der Autoren und deren Weitergabe betreffe. Felizitas erklärt süffisant, sie sei keine »Juristin mit Spezialgebiet internationale Datenmafia«, um einschätzen zu können, was das eigentlich für die hochgeladenen Informationen über ihre Tochter bedeute. Und gehöre die Vermittlung der vielbeschworenen »Digitalen Mündigkeit« nicht auch zum Informatikunterricht, fragt sie.

Die Stimmung wird gereizt, weitere Eltern äußern sich kritisch. Allerdings sieht die Mehrheit der Elternschaft kein Problem in dem Vorgehen des Lehrers. Er habe schließlich die Aufgabe gestellt und dürfe auch die Bedingungen diktieren. Es sei zwar nicht schön, dass man alle Kinder unter eine Art Generalverdacht stellen würde, aber dass viele Arbeiten aus dem Netz geladene Kopien sind, bestreiten die Eltern nicht. Und am Ende seien es doch nur Arbeiten von Neuntklässlern, die Kirche solle man bitteschön im Dorf lassen.

Fragen:

1. Kann eine solche Plagiatssoftware dabei helfen, aus unehrlichen Schülern ehrliche zu machen? Ist dieser Ansatz für eine Schule angemessen?
2. Sollen schulische Arbeiten einer kommerziellen Verarbeitung zugeführt werden dürfen, wenn man dafür im Gegenzug einige Plagiate aufdecken kann?
3. Wäre es in Ordnung, wenn man die Schüler und Eltern vorab fragt, ob sie einverstanden sind? Müssten dann die Schülerarbeiten, deren Autoren nicht zugestimmt haben, einzeln und per Hand auf Plagiate getestet werden?
4. Macht es einen Unterschied, wenn die Plagiatserkennungssoftware auf dem Schulserver betrieben würde und die Daten lediglich für die Lehrerschaft einsehbar sind?
5. Wem gehören schulische Auftragsarbeiten? Hat der Lehrer nicht ein ebenso großes Bestimmungsrecht wie die Schüler? Schließlich hätten die Schüler ohne ihn den Text ja gar nicht verfasst.
6. Wäre die Schule verpflichtet gewesen, dafür zu sorgen, dass die Nutzungsbedingungen eines Dienstleisters vor dem Einsatz geprüft werden?
7. Sehen Sie einen Unterschied in der moralischen Bewertung, wenn es um Quelltexte statt um Textarbeiten geht?

Erschienen in *Informatik-Spektrum* 37 (1), 2014, S. 59-60

8.2 Arbeitsblatt 2: Smarte Armbänder

Constanze Kurz & Stefan Ullrich



Vicky ist in der Technik-Abteilung einer Firma für Gesundheitsbedarf angestellt. Die früher nur aus zehn Leuten bestehende Truppe wurde in den letzten beiden Jahren enorm aufgestockt und das Personal verdreifacht. Dabei ist Vicky Chefin ihres eigenen kleinen Teams aus sechs Mitarbeitern geworden. Sie ist Informatikerin und beschäftigt sich in der Firma zumeist mit der ganz praktischen Umsetzung von Hardware-Lösungen.

Im letzten Jahr hatte ihr Team für eine große Reha-Kette in Armbändern integrierte Chips entwickelt und geliefert, die nun sowohl als Zugangslösung zu Klinik-Räumlichkeiten dienen als auch durch Vibration an die Medikamenteneinnahme erinnern. Vickys Team hatte die Hardware entworfen, die Software und die Integration in die Kliniken der Kette übernahmen andere Firmen-Abteilungen. Die Zusammenarbeit hatte gut funktioniert, das Zeitbudget wurde nur mäßig überschritten.

Ihrem neuen Projekt sieht Vicky gelassen entgegen, da dieselbe Technik weiterverwendet werden kann, die zuvor für die Klinik-Kette entwickelt worden war. Zufälligerweise hatte sie den Deal über einen persönlichen Kontakt mit Ferdinand, einem Pfarrer, sogar selbst an Land gezogen. Der neue Kunde CARE-ful betreibt mehrere kirchliche Schul- und Betreuungseinrichtungen. Die Hardware kann zwar nicht als Armband umgesetzt werden, da sich der Auftraggeber dagegen entschieden hatte. Vertreter von CARE-ful und die kirchlichen Kontaktpersonen hatten sich die Klinik-Armbänder zeigen lassen und erbeten, eine andere Lösung vorzuschlagen. Die Kinder würden die Armbänder zu schnell zerstören, hatte Vicky als Begründung gehört.

Das Team, das die Lesetechnik für die Chips in den Gebäuden anbringen und die Software installieren soll, hatte nach einem Besuch von zwei Einrichtungen vorgeschlagen, einfach die Schuluniform der Kinder zu nutzen. Der Leiter des Software-Teams Enrico hatte mit Vicky besprochen, ob man die Chips in den Kragen integrieren könne. Das hatte sie bejaht und die Planungen begonnen.

Vickys Firma war begeistert von der Idee, denn man konnte CARE-ful nun ein Komplettsystem anbieten: neue Schul- und Hortklassen bekämen die standardisierte Kleidung inklusive waserdicht verschweißtem Chip geliefert, in die bereits getragenen Uniformen würde der Chip im Nachhinein unter dem Kragen angebracht.

Eine einfache Idee mit einem Klettverschluss wurde dazu umgesetzt, damit die Sachen waschbar bleiben. Wenn die Eltern den Chip doch mitwaschen sollten, war der Ersatz natürlich lieferbar – gegen Entgelt, versteht sich.

Nicht nur das, mit wenigen Handgriffen sei auch die Barrierefreiheit gewährleistet, was prompt einen weit größeren Auftrag bringt: In den kommenden Jahren würden in allen Gebäuden von CARE-ful barrierefreie Türsysteme installiert, die sich selbständig beim Herannahen eines berechtigten Chips in der Kleidung öffnen. Auch Fahrstühle könnten für Schülerinnen und Schüler mit eingeschränkter Mobilität freigeschaltet werden. Vickys Chefs waren höchst erfreut.

Alles lief recht gut im Plan, einzig die Kommunikation mit der Software-Abteilung von Enrico fand nur punktuell statt. Vicky hatte zuvor noch nie mit Enrico gearbeitet und war irritiert darüber, dass nicht nur er, sondern eigentlich sein ganzes Team sehr abgekapselt arbeiteten. Es wurde wenig kommuniziert, manche E-Mail und manches Ticket im firmeneigenen System blieb einfach unbearbeitet. Andererseits hatte Vicky keine Anhaltspunkte, dass Enricos Team nicht im Zeitplan war. Im Grunde waren ihr weniger Kommunikation und seltene Meetings sogar lieber als zu viel.

Nur sechs Wochen, bevor das neue System an der ersten Schule gestartet werden sollte, erhält Vicky eine merkwürdige Nachricht auf ihrer privaten E-Mailadresse. Sie ist gerade auf dem Sprung nach Hause und hat zuvor noch die ersten der neuen Schuluniformen mit integrierten Chips abgeholt und inspiziert. Gutgelaunt macht sie sich auf den Weg und sieht auf dem Mobiltelefon die Nachricht von Pfarrer Ferdinand. Sie hatten ein privates Treffen für nächste Woche vereinbart, welches er nun mit harschen Worten absagt.

Bestürzt ruft Vicky Ferdinand an und fragt ihn, was los sei. Mit distanzierter Stimme sagt er zu ihr: „Dein Vertrauensmissbrauch enttäuscht mich. Ich verweigere meine Zustimmung! Du hättest mir sagen müssen, wenn ihr hinter meinem Rücken ein solches Überwachungsvorhaben baut und den Eltern Zusatzdienste verkauft.“

Vicky ist entgeistert, was meint er? Sie beruft ein Meeting ein, das Enrico aber kurzerhand absagt. Man sei im Stress und hätte wenig Zeit für Meetings. Sein Team sei im Zeitplan, alle technischen Schnittstellen sind umgesetzt, Enrico sehe keinen Sinn, jetzt alle grundlos aus der Arbeit zu reißen. Enricos Team muss tatsächlich für Meetings immer durch die halbe Stadt reisen, weil die Abteilungen auf verschiedene Standorte verteilt sind.

Vicky kennt Roswita, die jetzt in Enricos Team arbeitet, seit mehreren Jahren und versucht, sie zu kontaktieren. Sie erreicht Roswita erst abends und erfährt zu ihrem Entsetzen erstmals, dass Enricos Team an CARE-ful eine Software verkauft hat, die alle uniformierten Kinder durch die gesamten Gebäude dauerhaft trackt. Es ist sogar eine Zusatzfunktion verkauft worden, die den Eltern oder Lehrern der Kinder Benachrichtigungen aufs Mobiltelefon sendet, wo sich Schüler wann aufhalten. Davon war niemals die Rede gewesen in Vickys Team, an sowas wollte sie nie mitwirken! Was soll sie jetzt machen?

Fragen:

1. Darf man ein Zugangssystem, das ursprünglich für die Nutzung von erwachsenen Patienten entwickelt wurde, einfach zu einem Trackingsystem für Kinder umbauen?
2. Hätte sich Vicky neben der Hardware stärker für die Softwareumsetzung interessieren müssen?
3. Ändert es etwas, wenn die Eltern der Kinder diesem Tracking zustimmten? Aber was, wenn nicht alle Eltern und Lehrer zustimmten?
4. Die hier erwähnte Monitoring-Technik dient auch der Erinnerung an die Medikamenteneinnahme. Wie berücksichtigt man diese positiven Effekte in einer ethischen Beurteilung der Überwachungstechnik?
5. Ist es vertretbar, dass Vicky sich quasi auf informellem Wege die Informationen dazu besorgte, was das Software-Team erarbeitet hat?
6. Muss Vicky Rücksicht nehmen auf Pfarrer Ferdinand, der das Projekt vermittelte und erbost ist?

7. Reicht es, wenn Vicky darauf besteht, dass prozedurale Maßnahmen getroffen werden, die die Überwachung nur in begründeten Ausnahmefällen zulässt? Was wären solche begründbaren Ausnahmefälle?
8. An wen sollte sich Vicky wenden, um über das Problem zu reden?

Erschienen im *Informatik Spektrum* 41(4), 2018, S. 285-287.

8.3 Arbeitsblatt 3: Das selbstfahrende Auto

Christina Class & Debora Weber-Wulff



Sie haben sich seit Jahren darauf vorbereitet. Visionen der 50er Jahre eines selbstfahrenden Autos werden wirklich wahr. Galene haben sie ihre Entwicklung, das selbstfahrende Auto, getauft, und es ist auf dem Testgelände fantastisch gefahren. Auch bei Testfahrten in Amerika lief alles prima. In den USA gibt es nicht so viele Vorschriften, und es gibt unendlich weite Straßen, auf denen man gute Sicht hat und viel ausprobieren kann.

In Deutschland war alles komplizierter und es hat länger gedauert, um die Genehmigung für Testfahrten im Straßenverkehr zu erhalten, aber diese lag nun vor. Morgen war die Presse für die breit angekündigte „Jungfernfahrt“ eingeladen. Jürgen, einer der stolzen „Eltern“ von Galene, will sie nach Absprache mit dem Teamleiter erst einmal so ohne Pressetrubel auf der geplanten Strecke fahren lassen, um sicher zu sein, dass alles klappt. Als guter Ingenieur hat er die Strecke mit Bedacht ausgewählt. Es ist Sonntagnachmittag, da ist auf diesen Straßen wenig Verkehr. Und er selber sitzt ja auch hinterm Steuer und kann notfalls eingreifen, wenn etwas sein sollte. Er ist sich sicher, dass er nicht auffallen und evtl. andere Autofahrer oder Passanten irritieren wird.

Er gibt das Ziel im Sprachcomputer ein. Galene berechnet den Weg unter Berücksichtigung der aktuellen Verkehrsnachrichten und den bekannten Baustellen. Alles ist ruhig, keine Baustellen im Weg, kein Regen oder Nebel aber etwas Wind: ein sonniger Herbsttag und der perfekte Tag für eine erste Testfahrt!

Jürgen genießt die Fahrt. Es ist schon ein tolles Gefühl, sich fahren zu lassen, obwohl es immer noch sehr ungewohnt ist, kein Gas zu geben, nicht zu bremsen und nicht zu steuern. Galene fädelt auf der Stadtautobahn perfekt ein, überholt einen Oldtimer, und fährt an der nächsten Ausfahrt wieder raus, sanft an der Ampel anhaltend. Galene hält dabei immer genügend Abstand zum Vordermann ein. Die Steuerung ist so präzise, dass sie bis auf einen Zentimeter heranfahren könnte, um hinter dem Vordermann anzuhalten. Aber das macht andere Verkehrsteilnehmer unnötig nervös, und so haben sie Galene auf 40 cm Abstand programmiert.

Jürgen würde gerne mit seinem Handy filmen, wie er die grüne Welle bekommt und dann an der dritten Ampel links abbiegt – er war sich nicht ganz sicher, ob Galene das alles korrekt berechnen würde, aber es hat perfekt geklappt. Da er für die anderen Autofahrer den Anschein aufrecht halten muss als fahre er das Auto, kann er schlecht mit dem Handy filmen. Sie kommen in ein neueres Wohnviertel und Galene bremst auf die vorgeschriebenen 30 km/h runter. Links ist eine Schule, auf beiden Seiten befinden sich Bushaltestellen für die Schulbusse. Sie haben viel Zeit darauf verwendet, Galene auf solche Verkehrssituationen vorzubereiten.

Aber zum Glück sind ja gerade Herbstferien. Rechts kommen sie an einem Park mit großen Rasenflächen vorbei. Er hört Kindergeschrei und blickt nach rechts. Jürgen sieht Hunde toben, bunte Bälle auf der Wiese und noch buntere Drachen in der Luft fliegen. Sie bewegen sich mit dem Wind in seine Richtung. Jürgen greift instinktiv zum Steuer, er ist sich bewusst, dass Kinder beim Spielen nicht auf den Verkehr achten.

Gerade zu Beginn hat er das öfter erlebt: er wurde nervös, griff zum Steuer und drückte den Knopf, um die Kontrolle zu übernehmen. Aber es war nie notwendig und so lernte er, entspannt zu bleiben und Galene die Kontrolle zu überlassen. Und doch passiert es: Ein Kind

rennt mit seinem Drachen zwischen zwei parkenden Autos auf die Straße und wird von Galene erfasst. Es stürzt bewusstlos zu Boden.

Galene bremst sofort, da die Sensoren den Aufprall erfasst haben. Jürgen zieht gleichzeitig den Nothaltknopf. Galene steht, der Warnblinker ist eingeschaltet. Jürgen steigt aus und rennt zum Kind, die Mutter kommt dazu und schreit Jürgen an. Eine junge Frau steigt aus dem Auto, das hinter Jürgen gefahren ist und leistet Erste Hilfe. Sie sagt, sie sei Krankenschwester.

Ein Hundebesitzer aus dem Park hat schon den Notruf abgesetzt, der Krankenwagen ist schnell da und fährt das Kind mit seiner Mutter in das nächstgelegene Krankenhaus. Auch die Polizei ist da, um den Unfall aufzunehmen. Jürgen steht ziemlich unter Schock. Die junge Frau, die Erste Hilfe geleistet hat, wendet sich sofort an die Polizei, noch bevor diese Jürgen befragen kann. Ihr Name sei Sabine und sie sei hinter dem Unfallauto gefahren. Ihrer Meinung nach sei es zu schnell gefahren. Sie selber sei viel langsamer als 30 gefahren, bei all dem Kindergeschrei aus dem Park, den Hunden, Bällen und Drachen musste man ja mit sowas rechnen!

Die Polizisten bitten Jürgen, sich auszuweisen. Er gibt ihnen den Personalausweis, Führerschein, sowie die Betriebserlaubnis für Testfahrten. Die Polizisten sind erstaunt und stellen einige Fragen zum Auto, sie sind neugierig. Da es sich um Testfahrten handelt und das Auto nicht generell zugelassen ist, wollen sie Galene in jedem Fall abschleppen lassen, um genauere Untersuchungen anzustellen, insbesondere auch um Daten für die Auswertung auszulesen. Jürgen weiß, dass Galene korrekt gefahren ist, aber der Vorwurf der Zeugin Sabine macht ihn trotzdem nachdenklich. Die Probefahrt und Pressekonferenz am nächsten Nachmittag sind gefährdet. Ein PR Desaster, vor allen Dingen nach diesem Unfall!

Fragen:

1. Das Auto hatte eine offizielle Betriebserlaubnis für Testfahrten. War es in Ordnung, vor der offiziellen Probefahrt, eine Testfahrt durchzuführen?
2. Flugzeugpiloten müssen immer wieder ein Training absolvieren, um im Notfall schnell genug zu reagieren und die Kontrolle vom Autopiloten zu übernehmen. Wird ein solches Training auch bei selbstfahrenden Autos notwendig sein?
3. Als Jürgen die Kinder im Park sah, griff er instinktiv zum Lenkrad. Müsste er in einer solchen Situation, in der er mit auf die Straße rennenden Kindern rechnete, die Kontrolle über das Fahrzeug übernehmen?
4. Galene fuhr die vorgeschriebene Geschwindigkeit von 30 km/h. Die Zeugin machte den Vorwurf, das sei bei den vielen Kindern im Park zu schnell gewesen. Inwiefern kann und soll eine Anpassung der Geschwindigkeit an Gegebenheiten, die neben der Straße stattfinden, in Algorithmen abgebildet werden?
5. Es wird kaum möglich sein, in Tests alle möglichen Fälle abzudecken. Daher kann es vorkommen, dass die Software von selbstfahrenden Autos falsch reagiert. Wer ist dann zur Verantwortung zu ziehen?
6. Wie und wann werden Software Updates bei selbstfahrenden Autos durchgeführt? Nur in der Werkstatt, oder wenn das Auto gerade steht? Wer kontrolliert, ob und wann ein Update durchgeführt werden soll? Und wie sieht die Situation aus, wenn ein Unfall durch ein nicht durchgeführtes Update wahrscheinlich hätte verhindert werden können? Wer übernimmt dann die Verantwortung?

Erschienen in *Informatik Spektrum* 38(6), 2015, S. 575–577.

9 Workshop: Das Robotergericht

Autor: Rainer Rehak

Betreuungsschlüssel: 1:20

Dauer: ca. 2h

Ein Roboter steht vor Gericht. Er soll einem Menschen den Job weggenommen haben und muss sich nun rechtfertigen. Denn schließlich wurde ja lediglich ein Arbeitsablauf automatisiert, um schneller, sicherer und zuverlässiger zu funktionieren. Andererseits klagt die Arbeitnehmervertretung, dass eine weitere Person nun arbeitslos ist.

In den Rollen von Anklage, Verteidigung und Gericht diskutieren wir über die zunehmende Automatisierung der Gesellschaft und wie wir mit ihr umgehen sollten.



4Plakat vom Robotergericht

Ziel dieses Workshops ist es, jenseits der extremen Idealisierung oder Verteufelung über die gesellschaftlichen Implikationen von Automatisierung nachdenken zu können. Welche Aufgaben könnten warum automatisiert werden, welche Berufe sind in Gefahr, welche Folgen hätte das für Wirtschaft bzw. Gesellschaft und mit welchen begleitenden Maßnahmen kann Automatisierung politisch sinnvoll gestaltet werden. All diese Themen können zwar nicht abschließend geklärt werden, doch eine sinnvolle Strukturierung nützlicher Fragen und Ansätze steht am Ende eines erfolgreichen Workshops.

Ablauf

Zunächst können sich die TeilnehmerInnen zwischen den Teams "Anklage", "Verteidigung" und "RichterIn" entscheiden. Die ersten beiden Teams sollten gleich groß sein und größer als das letzte. Nun steht eine Kurzrecherche und Gruppendiskussion in den jeweiligen Teams an, welche Berufe oder Tätigkeiten besonders gut oder schlecht zu automatisieren sind. Kernbegriffe wie "Kreativität", "Menschenbezug", "Gefährlichkeit" oder "Routinearbeiten" können dabei auf Merktzetteln festgehalten werden. Danach sollten sich die Teams "Anklage" und "Verteidigung" daranmachen, Argumente für ihre jeweilige Position zu finden und zu formulieren. Team "RichterIn"

macht sich derweil daran, Fragen an die anderen Teams vorzubereiten sowie ggf. schon einmal Abwägungen und "vorzuschreibende" Maßnahmen vorzudiskutieren. Letztendlich muss Team "RichterIn" jedoch ein wenig spontan sein und auf die Aussagen und Argumente der anderen Teams eingehen können. Nachdem es einige Runden hin und her gegangen ist schließt Team "RichterIn" die Verhandlung und berät einige Minuten intern. Am Ende wird die Automatisierung vermutlich unter besonderen Auflagen und einzuhaltenden Regeln freigesprochen werden.

Tipps zur Durchführung des Workshops:

Bei den Gruppendiskussionen muss vermutlich anfangs etwas nachgeholfen werden, insbesondere in Bezug auf das Entmystifizieren der üblicherweise zirkulierenden Studien, wonach die Hälfte aller Jobs verloren gehen werden. Zudem sollte totale Ablehnung oder Befürwortung von Automatisierung durch Argumente belegt werden, damit die Workshop-TeilnehmerInnen ihre eigenen Ansichten hinterfragen bzw. begründen lernen. Bei diesem Thema ist es auch relevant, die gesellschaftlichen Rahmenbedingungen

mitzudenken: Wer profitiert von Vollautomatisierung? Was passiert mit Menschen bei einem Arbeitsplatzverlust? Wer sollte Umschulungen zahlen? Hilft ein Grundeinkommen? Ist Vollautomatisierung überhaupt realistisch?

Literaturempfehlung:

Nuss, Sabine & Butollo, Florian (Hg.): Marx und die Roboter – Vernetzte Produktion, Künstliche Intelligenz und lebendige Arbeit, 2019, <https://kritisch-lesen.de/rezension/der-alb-traum-von-der-menschenleeren-fabrik>

10 Impressum

2. aktualisierte Auflage Juli 2020

Turing Bus – Sammlung offener Lehrmaterialien

Herausgeberin

Gesellschaft für Informatik e.V.
Anna-Louisa-Karsch-Str. 2
10178 Berlin

E-Mail

info@turing-bus.de

Internet

www.turing-bus.de



Dieses Heft und alle in ihm veröffentlichten Texte und Bilder stehen, soweit nicht anders angegeben, unter der **Creative Commons Lizenz Namensnennung - Weitergabe unter gleichen Bedingungen 4.0 International (CC BY-SA 4.0)**.

<https://creativecommons.org/licenses/by-sa/4.0/>

Gerne dürfen Sie unsere Materialien

- *Teilen, Vervielfältigen und Weiterverbreiten* in jedwedem Format oder Medium
- *Bearbeiten, Remixen und darauf Aufbauen!*

Kontakt zu uns

Mail: info@turing-bus.de
Telefon: +49 (0)30-24009-865

Gesellschaft für Informatik e.V. (GI)
Anna-Louisa-Karsch-Str. 2
10178 Berlin

Open Knowledge Foundation Deutschland e.V.
Singerstraße 109
10179 Berlin



www.turing-bus.de



www.twitter.com/TuringBus



www.instagram.com/turingbus

Im Wissenschaftsjahr 2019 – Künstliche Intelligenz

www.turing-bus.de



GESELLSCHAFT
FÜR INFORMATIK



weizenbaum
institut

