

KI-Workshop: Praxisnahes Lernen für Jugendliche

6. März 2025

Inhaltsverzeichnis

1	Warum KI? Die Schlüsseltechnologie der nächsten Generation	1
2	Warum dieser Workshop wertvoll ist	3
2.1	Technische Grundlagen	3
2.2	Grundlagen in Netzwerktechnik	3
2.3	Social Media, Datenschutz & Sicherheit	3
2.4	Linux & Web-Technologien	3
2.5	Einführung in Künstliche Intelligenz	3
2.6	Hands-on: Das KI-Projekt (Praktische Umsetzung)	4
2.7	Eigenständige Problemlösung mit KI	4
3	Warum dieser Workshop für Jugendliche wertvoll ist	5
4	Wichtige Punkte für den Workshop	7
5	Warum eine feste Gruppe für alle sechs Miniworkshops sinnvoll ist	9
6	KI-generierte Inhalte: Risiken und Beispiele	11
7	Die verwendeten Tools im Überblick	12
8	Liste der verwendeten KI-Modelle	14
9	Struktur des Workshops	16
9.1	Tag 1: Einführung & Technische Grundlagen	16
9.2	Tag 2: Datenschutz, Social Media & Scratch (Logik lernen)	16
9.3	Tag 3: Web-Technologien, Python & KI-Grundlagen	17
9.4	Tag 4: Hands-on mit KI – Erste Modelle testen	17
9.5	Tag 5: Eigenes KI-Projekt entwickeln (Gruppenarbeit)	17
9.6	Tag 6 (Halber Tag): Projektvorstellung & Reflexion	18
10	Diskussionsfragen für den Workshop	19

1 Warum KI? Die Schlüsseltechnologie der nächsten Generation

Künstliche Intelligenz (KI) ist nicht nur eine Technologie, sie ist eine der größten Revolutionen seit der Entstehung des Internets. Während das Internet die Welt miteinander vernetzt hat, verändert KI, wie wir Probleme angehen, Erkenntnisse gewinnen und Innovationen schaffen. Was früher zeitaufwändige, komplexe Berechnungen und Analysen erforderte, wird heute von KI in kürzester Zeit erledigt. Sie erkennt Muster schneller als der Mensch, ermöglicht tiefere Einblicke und optimiert Prozesse, die uns zuvor unerreichbar schienen. In meinem Workshop liegt der Fokus bewusst auf dieser revolutionären Technologie, da sie eine Schlüsselrolle in der Zukunft der Forschung und darüber hinaus spielt. KI hat das Potenzial, wissenschaftliche Durchbrüche zu beschleunigen und uns dabei zu helfen, die Herausforderungen der Zukunft effizienter zu meistern.

Exoplanetenentdeckung

- **Modell:** Kepler's K2 (und KI-basierte Modelle wie Machine Learning for Exoplanet Discovery).
- **Beispiel:** Forscher verwenden maschinelles Lernen, um riesige Datenmengen von Welt- raumteleskopen zu analysieren. DeepMind hat beispielsweise KI-Algorithmen entwickelt, die die Entdeckung von Exoplaneten aus Lichtkurven verbessern. Ein weiteres Modell, das in dieser Forschung verwendet wird, ist das ExoMiner-Modell, das auf maschinellem Lernen basiert und von der NASA entwickelt wurde, um Exoplaneten mit höherer Präzision zu identifizieren.

Proteinstrukturforschung

- **Modell:** AlphaFold von DeepMind.
- **Beispiel:** AlphaFold revolutionierte die Proteinstrukturvoraussage, indem es eine KI entwickelte, die die Faltung von Proteinen vorhersagt. Das Modell ist ein Deep-Learning-Modell, das auf der Kombination von neuronalen Netzen und biologischem Wissen basiert, um die 3D-Strukturen von Proteinen aus ihrer Aminosäuresequenz zu ermitteln.

MRT-Auswertung

- **Modell:** DeepMind's Health AI und U-Net.
- **Beispiel:** DeepMind hat KI-Modelle entwickelt, die MRT-Scans und andere medizinische Bilddaten analysieren, um Krankheiten zu diagnostizieren. U-Net ist ein weiteres häufig verwendetes Modell in der medizinischen Bildverarbeitung, insbesondere in der Segmentierung von medizinischen Bildern, z.B. bei der Identifikation von Tumoren oder Organschäden.

Entzifferung verbrannter Schriftrollen

- **Modell:** Deep Learning-based Text Recognition und CNNs (Convolutional Neural Networks).
- **Beispiel:** KI-Modelle wie Convolutional Neural Networks (CNNs) haben bei der Entzifferung verbrannter und stark beschädigter Schriftrollen, wie denen aus Herculaneum, große

Fortschritte erzielt. Diese Schriftrollen sind oft stark zusammengerollt und aufgrund von Feuer und physischem Schaden kaum lesbar. Die KI hilft dabei, die Textfragmente zu rekonstruieren, die durch den Schaden noch erkennbar sind. Sie analysiert die beschädigten Stellen und erkennt Muster, die für das menschliche Auge unsichtbar wären. Auf diese Weise ermöglicht es die KI, die Inhalte dieser antiken Texte zu entschlüsseln, die ansonsten möglicherweise für immer verloren gegangen wären.

Medizinische Diagnosen

- **Modell:** PathAI, CheXNet, und DeepMind AI for Healthcare.
- **Beispiel:** PathAI verwendet maschinelles Lernen, um die Diagnose von Krebs zu unterstützen, indem es Gewebeproben analysiert. CheXNet, ein Convolutional Neural Network, wurde entwickelt, um Lungenentzündungen und andere Krankheiten anhand von Röntgenbildern zu diagnostizieren. DeepMind hat ebenfalls KI-Modelle entwickelt, die die Analyse von Augenkrankheiten und anderen medizinischen Tests effizienter und präziser machen.

Klimamodelle und Wettervorhersagen

- **Modell:** DeepMind's Energy Forecasting AI und Weather Prediction with Machine Learning.
- **Beispiel:** DeepMind hat ein KI-Modell entwickelt, das dabei hilft, den Energieverbrauch und die Wetterbedingungen besser vorherzusagen, um die Nutzung erneuerbarer Energiequellen zu optimieren. Machine Learning for Weather Forecasting nutzt historische Wetterdaten, um präzisere Vorhersagen zu treffen und klimatische Veränderungen schneller zu erkennen.

Arzneimittelentwicklung

- **Modell:** AlphaFold (auch hier), ChemBERTa und Graph Neural Networks.
- **Beispiel:** Bei der Entdeckung neuer Medikamente wird maschinelles Lernen eingesetzt, um Wirkstoffe zu identifizieren, die gegen spezifische Krankheiten wirken könnten. Ein Beispiel ist das ChemBERTa-Modell von BERT, das für die Verarbeitung chemischer Daten entwickelt wurde. Graph Neural Networks werden auch verwendet, um die Beziehungen zwischen Molekülen und ihren biologischen Wirkungen zu verstehen und neue Arzneimittel zu entwickeln.

Diese KI-Modelle und -Technologien zeigen, wie vielseitig und leistungsfähig maschinelles Lernen und künstliche Intelligenz in der modernen Forschung sind. Sie ermöglichen es, komplexe wissenschaftliche und medizinische Herausforderungen viel schneller und effizienter zu lösen, als es mit traditionellen Methoden möglich wäre.

2 Warum dieser Workshop wertvoll ist

2.1 Technische Grundlagen

- Hardware verstehen
- Betriebssysteme kennenlernen: Linux, Windows, macOS
- Virtualisierung & Container:
 - Unterschied zwischen VMs und Docker
 - Debian in einer VM installieren
- Paketmanagement unter Linux:
 - APT, Paketinstallation & Verwaltung
 - PIP & Git: Pakete beziehen & kompilieren (z. B. purple-facebook für Pidgin)

2.2 Grundlagen in Netzwerktechnik

- Grundlagen von IP-Adressen (IPv4 und IPv6)
- NAT vs. Bridge – Was ist besser für Virtualisierung?
- Ports und Firewall – Warum sind sie wichtig?
- Grundlagen von Protokollen: HTTP, HTTPS, SSH, TCP/IP
- Warum HTTPS wichtig ist (und warum man es nicht aus Bequemlichkeit weglassen sollte)

2.3 Social Media, Datenschutz & Sicherheit

- Datenkraken vs. Alternativen: Google vs. DuckDuckGo/Ecosia
- Fake News und Medienkompetenz
- Alternativen zu Big Tech: NewPipe, Fediverse
- Messenger absichern: Facebook Messenger mit Pidgin, OTR und Tor
- Passwortsicherheit und Management: KeePass, sicheres Passwort erstellen

2.4 Linux & Web-Technologien

- Webserver einrichten (Nginx)
- Grundlagen von HTML und CSS (mit Unterstützung durch Duck.AI)
- Python und Flask einrichten
- Python-Umgebungen verwalten
- Installation und Nutzung von PyTorch (CPU-Version)

2.5 Einführung in Künstliche Intelligenz

- Was ist KI?
- Wo wird KI eingesetzt?

- Welche Vorteile und Herausforderungen gibt es?

2.6 Hands-on: Das KI-Projekt (Praktische Umsetzung)

- Jede Gruppe muss 3 von 5 KI-Modellen auswählen:
 - BLIP – Bilderkennung
 - DistilBERT – Textverarbeitung und Sentimentanalyse
 - GPT-2 – Textgenerierung
 - EasyOCR – Texterkennung
 - Real-ESRGAN – Bildverbesserung
- Einrichten eines Webinterfaces mit Flask zur Steuerung der Modelle

2.7 Eigenständige Problemlösung mit KI

- Code nicht einfach kopieren, sondern selbst erarbeiten
- Nutzen von Duck.AI zur Unterstützung:
 - Auswahl zwischen 5 verschiedenen Sprachmodellen
 - Erlernen, dass es mehr gibt als nur ChatGPT

◎ Warum ist dieser Workshop wertvoll?

- ✓ Praxisorientiert statt nur Theorie
- ✓ Eigenständiges Arbeiten & Problemlösung lernen
- ✓ Echte Programmier- & KI-Skills statt oberflächlicher Mini-Workshops
- ✓ Abwechslungsreiche Gruppenprojekte

3 Warum dieser Workshop für Jugendliche wertvoll ist

1. Bessere Lernstrategien durch KI

- Die Jugendlichen lernen, präzise Fragen zu stellen und bessere Antworten zu bekommen.
- Sie verstehen, wie man KI-Tools sinnvoll fürs Lernen einsetzt (z. B. Recherche, Zusammenfassungen).

2. Praktische IT-Skills, die über den Schulstoff hinausgehen

- Installation eines Betriebssystems (Debian in einer VM) und Einführung in Virtualisierung (z. B. Docker).
- Einrichten eines Webservers (NGINX) und einer Python-Umgebung für KI-Anwendungen.
- Paketmanagement & Git, also Software effizient verwalten und Versionierung verstehen.

3. Grundverständnis für KI und ihre Einsatzgebiete

- Die Jugendlichen erfahren, wie KI-Modelle funktionieren und welche es gibt.
- Sie setzen drei von fünf KI-Modellen selbst ein:
 - Bilderkennung (BLIP)
 - Texterkennung (EasyOCR)
 - Textanalyse (DistilBERT)
 - Textgenerierung (GPT-2)
 - Bildverbesserung (Real-ESRGAN)

4. Eigenständiges Arbeiten & Problemlösung

- Statt nur Anleitungen zu befolgen, müssen sie selbst Lösungen finden.
- Sie nutzen Duck.AI zur eigenständigen Erarbeitung von Lösungen – KI als Werkzeug richtig einsetzen.

5. Datenschutz & digitale Selbstverteidigung

- Sie lernen, welche Daten Google & Co. sammeln und wie man sich schützt.
- Praktische Tools für Datenschutz & Sicherheit:
 - Alternativen zu Datenkraken (z. B. DuckDuckGo, NewPipe, Fediverse).
 - Facebook Messenger sicher machen mit Pidgin, OTR und Tor.
 - Richtiges Passwortmanagement mit KeePass.

6. Social Media: Gefahren & Alternativen

- Die Wahrheit über TikTok & Co. – Was mit Nutzerdaten passiert und welche psychologischen Tricks Plattformen nutzen.
- Influencer & ihr schädlicher Einfluss – Manipulation, Konsumverhalten, falsche Körperbilder.

- Fake News & Propaganda erkennen – Wie gezielte Kampagnen funktionieren (z. B. Trumps Wahlkampf-Strategien auf Facebook/Twitter).
- Faktencheck lernen – Nutzung von Mimikama (DACH-Region) und anderen Faktencheckern.

7. Kreativität & Teamarbeit

- Jede Gruppe entwickelt ihr eigenes KI-Projekt – keine 0815-Lösungen.
- Sie müssen zusammenarbeiten, um eine funktionierende Web-App zu erstellen.

8. Crashkurs in Webentwicklung & Programmierung

- HTML & CSS für das Webinterface.
- Scratch als Einstieg, um logisches Denken zu fördern, bevor es mit Python weitergeht.

9. Hands-on statt Theorie

- Statt nur über KI zu reden, bauen sie echte KI-Anwendungen!
- Sie erleben den Unterschied zwischen Theorie & Praxis.

10. Open-Source & Eigeninitiative statt teurer Software

- Die Jugendlichen lernen, dass freie Software oft genauso gut oder besser ist als kommerzielle Lösungen.
- Sie verstehen, dass man mit Python & Open-Source-KI-Tools viele Probleme selbst lösen kann.
- Kein Vendor Lock-in – sie lernen, unabhängig zu bleiben und nicht auf teure Software angewiesen zu sein.

Bonus für die Verwendung als Schulprojekt:

- Fördert Medienkompetenz & kritisches Denken, was immer wichtiger wird.
- Kann als offizielles Schulprojekt oder Wettbewerbsteilnahme genutzt werden.
- Ist als Projektwoche ideal – intensiv, praxisnah und spannend!

4 Wichtige Punkte für den Workshop

1. Lücken in der schulischen Ausbildung (IT) ausgleichen

1. Wenig bis keine Informatik- und Programmierkenntnisse in der Schule, besonders bei jüngeren Jugendlichen.
2. Die Schule vermittelt hauptsächlich grundlegende, wenig praxisrelevante Themen, während komplexere, zukunftsorientierte Technologien fehlen.

2. Vorbereitung auf die digitale Zukunft

1. Jugendliche sind nicht ausreichend auf die Anforderungen der digitalisierten Welt vorbereitet.
2. Informatik und technologische Fähigkeiten sind heute in fast allen Berufsfeldern unverzichtbar – aber in vielen Schulen bekommen sie wenig bis keine Bedeutung.

3. Wachsende Bedeutung von Technologie

1. Ein starkes Fundament in IT ist entscheidend, um im digitalen Zeitalter zu bestehen. KI, Programmierung und maschinelles Lernen sind essentielle Fähigkeiten für die Zukunft.
2. Der Workshop vermittelt den Jugendlichen praktische Kenntnisse und zeigt ihnen, wie sie Technologien wie Open-Source-Software, Python und KI-basierte Tools selbstständig nutzen können.

4. Open-Source-Denken und Eigeninitiative

1. Der Workshop fördert den Open-Source-Gedanken, der den Jugendlichen zeigt, dass sie nicht auf teure Software angewiesen sind, sondern Lösungen selbst entwickeln können.
2. Sie lernen, wie sie durch einfache Programmierung in Python und die Nutzung von freien Tools wie KI-Modelle und Automatisierungen selbst erstellen können.

5. Praktische IT-Fähigkeiten, die in der Schule oft zu kurz kommen

1. Das Lernen von praktischen IT-Anwendungen wie dem Erstellen von Webseiten, der Nutzung von GitHub und dem Arbeiten mit verschiedenen Python-Bibliotheken hilft, den Jugendlichen Fähigkeiten zu vermitteln, die sie direkt anwenden können.
2. Programme wie Stable Diffusion und die Nutzung von Flask-Apps bieten den Jugendlichen einen Einstieg in moderne Technologien und machen komplexe Themen greifbar.

6. Verknüpfung von Theorie und Praxis im IT-Bereich

1. Die Jugendlichen setzen ihr Wissen in praktischen Projekten um, was das theoretische Verständnis fördert und gleichzeitig ihre Problemlösungsfähigkeiten stärkt.
2. Sie lernen nicht nur, wie man Code schreibt, sondern auch, wie man echte Probleme mit Technologie löst – z.B. mit KI oder durch Programmierung.

7. Förderung von Kreativität und Problemlösungsfähigkeiten

1. Der Workshop zeigt den Jugendlichen, dass IT und Programmierung nicht nur für technische Berufe sind, sondern auch ihre Kreativität fördern können (z.B. durch das Erstellen von KI-generierten Bildern).
2. Sie entwickeln ein Verständnis dafür, wie sie Technologie nutzen können, um Probleme zu lösen und neue Ideen umzusetzen.

8. IT und digitale Medienkompetenz als Schlüsselkompetenzen

1. Da Jugendliche immer mehr Zeit mit digitalen Medien verbringen, ist es umso wichtiger, dass sie diese nicht nur passiv konsumieren, sondern auch verstehen, wie diese Technologien funktionieren und wie sie selbst davon profitieren können.
2. Der Workshop stärkt ihre digitale Medienkompetenz und hilft ihnen, verantwortungsvoll mit den Technologien umzugehen (z.B. im Umgang mit Social Media und Datenschutz).

Zusammenfassung

Diese Punkte verdeutlichen, wie wichtig es ist, den Jugendlichen praktische IT-Fähigkeiten zu vermitteln, um sie nicht nur auf die Zukunft vorzubereiten, sondern auch ihre Problemlösungsfähigkeiten und Kreativität zu fördern!

5 Warum eine feste Gruppe für alle sechs Miniworkshops sinnvoll ist

Die folgenden Punkte verdeutlichen die Vorteile einer festen Gruppe für die Miniworkshops:

Kontinuierliches Lernen statt wiederholtem Neustart

1. Durch eine feste Gruppe kann auf bereits erworbenem Wissen aufgebaut werden und das Verständnis schrittweise vertieft werden.
2. Eine feste Gruppe reduziert die Notwendigkeit, Grundlagen immer wieder zu erklären.

Tieferes Verständnis statt oberflächlichem Einblick

1. Künstliche Intelligenz und Programmierung sind komplexe Themen, die Zeit zur Erarbeitung benötigen.
2. Durch eine fortlaufende Betreuung können die Teilnehmenden echte Fortschritte machen, anstatt nur theoretische Einblicke zu erhalten.

Effizientere Nutzung der Workshop-Zeit

1. Durch eine feste Gruppe entfällt die wiederholte Einführung neuer Teilnehmer.
2. Die gewonnene Zeit kann für vertiefende Inhalte und praktische Anwendungen genutzt werden.

Mehr Eigenverantwortung & Teamarbeit

1. Eine feste Gruppe ermöglicht es den Teilnehmenden, gemeinsam an einem fortlaufenden Projekt zu arbeiten.
2. Dies fördert Verantwortungsbewusstsein, Zusammenarbeit und den kreativen Umgang mit Herausforderungen.

Höhere Erfolgsschancen für die Teilnehmenden

1. KI- und Programmierkenntnisse lassen sich nicht in einer einzigen Session nachhaltig vermitteln.
2. Mehrere aufeinander aufbauende Einheiten erhöhen die Chancen, dass das erlernte Wissen langfristig verstanden und angewendet werden kann.

Geringerer Betreuungsaufwand durch effiziente Wissensvermittlung

1. Eine feste Gruppe reduziert die Notwendigkeit, Grundlagen immer wieder zu erklären.
2. Dadurch bleibt mehr Zeit für individuelle Unterstützung und weiterführende Inhalte.

Höherer Lerneffekt durch praktische Umsetzung

1. Durch eine kontinuierliche Betreuung können die Teilnehmenden ein eigenes Projekt von Grund auf entwickeln und über mehrere Sessions hinweg verbessern.

2. Dies führt zu einem tieferen Verständnis und einer besseren Verknüpfung von Theorie und Praxis.

Erhöhte Motivation der Teilnehmenden

1. Die Aussicht, ein eigenes KI-Projekt zu realisieren, steigert das Engagement der Jugendlichen.
2. Ein Zertifikat am Ende des Workshops hätte größeren Wert, wenn es auf einer praktischen Leistung basiert, anstatt auf der bloßen Teilnahme an einzelnen Sessions.

Mehr Praxis statt Theorie

1. Ein durchgehendes Konzept ermöglicht eine stärkere Fokussierung auf praktische Anwendungen.
2. Statt in jeder Einheit nur einen theoretischen Überblick zu geben, können die Teilnehmenden direkt mit Programmierung und KI-Modellen arbeiten.

Bessere Vorbereitung auf reale Anforderungen

1. In der Berufswelt sind technologische Projekte oft langfristig und erfordern kontinuierliches Lernen.
2. Eine feste Gruppe ermöglicht eine realitätsnahe Herangehensweise an IT- und KI-Projekte.

6 KI-generierte Inhalte: Risiken und Beispiele

Die folgenden Punkte zeigen, wie KI generierte Inhalte zu Risiken führen können:

Deepfake-Pornografie

1. Ein Beispiel für eine Deepfake-App ist die "DeepNudeApp, die 2019 veröffentlicht und später wieder entfernt wurde.
2. Dennoch kursierten modifizierte Versionen weiter im Internet und es gab zahlreiche Fälle von gefälschten Deepfake-Videos, bei denen Prominente wie Emma Watson oder Scarlett Johansson Opfer wurden.

Politische Manipulation durch Deepfakes

1. Ein Beispiel für eine politische Manipulation ist das Fake-Video von Wolodymyr Selenskyj (2022), bei dem russische Propaganda ein Deepfake-Video erstellte, in dem der ukrainische Präsident zur Kapitulation aufrief.
2. Es gab auch ein Deepfake-Video von Joe Biden (2023), das genutzt wurde, um Wähler in die Irre zu führen und politische Wahlen zu beeinflussen.

KI und Betrug (Scam-Anrufe & Fake-Stimmen)

1. Kriminelle nutzen KI, um die Stimmen von Angehörigen nachzuahmen und Fake-Anrufe zu tätigen, wie im Enkeltrick mit KI-generierten Stimmen.
2. Es gab Berichte, dass ältere Menschen durch solche Betrugsmaschen ihr Geld verloren haben.

KI-generierte Drogenanleitungen & Waffendesigns

1. Einige Open-Source-KI-Modelle wurden ohne Sicherheitsfilter veröffentlicht, wie LLaMA oder GPT-J.
2. Forscher haben gezeigt, dass sie mit wenigen Anpassungen Anleitungen zur Herstellung von Drogen oder Waffen liefern konnten.

3D-gedruckte Waffen mit KI-Designs

1. Einige Gruppen nutzen KI-Modelle, um bessere 3D-druckbare Waffen zu entwerfen und die Verbreitung von illegalen Waffen zu erleichtern.

Fazit für den Workshop:

- KI ist ein mächtiges Werkzeug, aber sie kann auch gefährlich missbraucht werden.
- Es ist wichtig, sich bewusst zu sein, dass nicht alles im Internet echt ist.
- Man muss ethisch verantwortlich mit KI umgehen – nur weil etwas geht, heißt es nicht, dass man es tun sollte.

7 Die verwendeten Tools im Überblick

Scratch (Einstieg in die Programmierung)

- Kein Account nötig – Scratch kann direkt im Browser genutzt werden.
- Läuft überall – Webversion für alle Plattformen, offline als Scratch Desktop für Windows/macOS/Linux.
- Einfacher Einstieg – Keine Programmiererfahrung nötig, visuelle Blöcke statt Code.
- Fördert logisches Denken – Ideal, um Programmierkonzepte zu verstehen, bevor es mit Python weitergeht.
- Offline nutzbar – Projekte können lokal gespeichert und weiterbearbeitet werden.
- Ideal für Gruppenarbeit – Die Jugendlichen können zusammen an Projekten arbeiten.
- Exportierbar – Scratch-Projekte können als HTML5-Apps exportiert werden.
- Gamification – Spielerisches Lernen durch interaktive Animationen und kleine Spiele.

Duck.AI (Coding-Unterstützung & Webentwicklung)

- Kein Account nötig – Einfach aufrufen und benutzen.
- Läuft überall – Webbrowser-basiert, also unabhängig vom Betriebssystem.
- Hilft bei HTML & CSS – Perfekt für Webentwicklung und Styling von Projekten.
- Intelligente Code-Unterstützung – Gibt Tipps und generiert Vorschläge, falls jemand nicht weiterkommt.
- Ersetzt Google-Suche – Die Jugendlichen müssen nicht selbst nach Code-Snippets suchen.
- Schnelle Hilfe bei Problemen – Duck.AI kann Bugs erklären und Lösungen vorschlagen.
- Kein Tracking & Datenschutzfreundlich – Kein Login, keine unnötige Datensammlung.
- 5 Sprachmodelle zur Auswahl - GPT-4o mini, Llama 3.3, Claude 3, o3-mini, Mistral

VirtualBox (Unabhängige Entwicklungsumgebung mit Linux)

- Läuft überall – VirtualBox gibt's für Windows, macOS und Linux.
- Kein Account nötig – Open-Source-Software, einfach herunterladen und nutzen.
- Unabhängig von der Hardware – Die VM stellt eine einheitliche Umgebung bereit, egal ob die Jugendlichen einen alten oder neuen PC haben.
- Projekt kann mitgenommen werden – Die VM-Datei kann einfach auf einen USB-Stick kopiert werden.
- Hausaufgaben & Weiterarbeit möglich – Die Jugendlichen können das Projekt zuhause weiterentwickeln.
- Schützt das Host-System – Falls mal was schiefläuft, bleibt das Hauptbetriebssystem unangetastet.
- Einheitliche Entwicklungsumgebung – Jeder hat genau die gleiche Software- und Library-Version.

- Perfekt für Linux & Terminal-Kompetenz – Die Jugendlichen lernen, mit Linux zu arbeiten, ohne ihr eigenes System umzubauen.

Debian (Das Linux-Betriebssystem in der VM)

- Kein Account nötig – Open-Source, frei nutzbar, keine Registrierung.
- Läuft überall – Durch die VM unabhängig von der Host-Hardware.
- Stabile & sichere Umgebung – Perfekt für Software-Entwicklung und Server-Anwendungen.
- Lernen mit echter Linux-Umgebung – Jugendliche bekommen Praxis mit Terminal-Befehlen, Paketmanagement & Server-Konfiguration.
- Paketmanagement mit APT – Einfaches Installieren von Software.
- Open-Source-Denken fördern – Zeigt den Jugendlichen, dass es Alternativen zu Windows & Mac gibt.
- Geringe Systemanforderungen – Läuft problemlos in VirtualBox, auch auf älteren Rechnern.

Python (Programmiersprache für KI & Webentwicklung)

- Kein Account nötig – Open-Source, einfach installieren und loslegen.
- Läuft überall – Windows, macOS, Linux – perfekt für alle Plattformen.
- Einfache Syntax & Einsteigerfreundlich – Gut verständlich, ideal als erste Programmiersprache.
- Für KI & Webentwicklung geeignet – Flask für Web-Apps, PyTorch für KI-Modelle.
- Starke Community & viele Ressourcen – Viele Tutorials, freie Libraries, Stack Overflow-Hilfe.
- Direkt in Debian nutzbar – Keine extra Installation in der VM nötig.
- Verknüpfung mit Scratch – Jugendliche lernen erst visuelle Programmierung und steigen dann auf echten Code um.

8 Liste der verwendeten KI-Modelle

1. BLIP (Bilderkennung & Bildbeschreibung)

BLIP ist ein KI-Modell, das Bilder in Text umwandelt und kontextuelle Informationen erkennt. Es ist ideal für Anwendungen wie Barrierefreiheit oder automatische Bildunterschriften.

- Kann Bilder in Text umwandeln – Perfekt für Barrierefreiheit oder automatische Bildunterschriften.
- Versteht Kontext in Bildern – Kann komplexe Szenen gut beschreiben.
- Funktioniert offline – Kein Cloud-Zugriff nötig.
- Modifizierbar – Man kann eigene Modelle trainieren oder Finetuning betreiben.

2. DistilBERT (Textanalyse & Sentiment-Erkennung)

DistilBERT ist ein KI-Modell, das Texte verstehen und bewerten kann. Es ist ideal für Anwendungen wie Stimmungsanalysen oder Fake-News-Erkennung.

- Schneller & effizienter als BERT – Halb so groß, aber fast genauso gut.
- Kann Texte verstehen & bewerten – Z. B. für Stimmungsanalysen oder Fake-News-Erkennung.
- Läuft auf schwacher Hardware – Ideal für den Einsatz in der VM.
- Gut für NLP-Aufgaben – Kann Fragen beantworten, Texte zusammenfassen usw.

3. GPT-2 (Textgenerierung, ungefiltert)

GPT-2 ist ein KI-Modell, das flüssigen Text generieren kann. Es ist ideal für kreative Texte oder Auto vervollständigungen.

- Generiert flüssigen Text – Super für kreative Texte oder Auto vervollständigungen.
- Keine Ethik-Filter – Gibt genau das aus, was der Prompt hergibt.
- Einfach zu nutzen & trainieren – Open-Source & flexibel anpassbar.
- Läuft auch ohne GPU (in kleinerer Form) – Perfekt für ressourcenschwache Geräte.

4. EasyOCR (Texterkennung aus Bildern & Screenshots)

EasyOCR ist ein KI-Modell, das Text in Bildern und Scans erkennt. Es ist ideal für automatische Textextraktion.

- Erkennt Text in Bildern & Scans – Ideal für automatische Textextraktion.
- Unterstützt viele Sprachen – Funktioniert nicht nur mit Englisch.
- Offline nutzbar – Läuft lokal ohne Internetverbindung.
- Schnell & einfach einsetzbar – Keine komplizierte Einrichtung nötig.

5. Real-ESRGAN (Bildverbesserung & Upscaling)

Real-ESRGAN ist ein KI-Modell, das die Bildqualität verbessert und Bilder skaliert. Es ist ideal für alte oder verpixelte Bilder.

- Verbessert Bildqualität mit KI – Ideal für alte oder verpixelte Bilder.
- Funktioniert offline – Kein Cloud-Rendering nötig.
- Skaliert Bilder ohne Detailverlust – Perfekt für Vergrößerungen.
- Gut für Spiele & alte Fotos – Kann Texturen und alte Bilder schärfer machen.

9 Struktur des Workshops

9.1 Tag 1: Einführung & Technische Grundlagen

Ziel: Die Teilnehmer verstehen, wie Computer & Virtualisierung funktionieren, und richten ihre Arbeitsumgebung ein.

Themen:

- Einführung in den Workshop: Vorstellung, Ziele, Ablauf, Gruppenaufteilung
- Hardware verstehen: CPU, GPU, RAM, Speicher, Netzwerke
- Betriebssysteme: Unterschiede zwischen Linux, Unix, Windows
- Virtualisierung: VMs vs. Docker
- Debian in einer VM installieren & einrichten
- Paketmanagement unter Linux: APT, grundlegende Befehle

Netzwerktechnik:

- Grundlagen von IP-Adressen (IPv4 & IPv6)
- NAT vs. Bridge – Warum NAT für die VM genutzt wird
- Ports & Firewall – Welche Ports für Flask, SSH & Webserver geöffnet werden müssen
- Grundlagen von Protokollen – HTTP, HTTPS, SSH, TCP/IP

Praxis:

- Debian-Installation in einer VM
- Erste Schritte mit Linux-Befehlen

9.2 Tag 2: Datenschutz, Social Media & Scratch (Logik lernen)

Ziel: Die Jugendlichen verstehen Datenschutzrisiken & beginnen mit Scratch, um logisches Denken zu fördern.

Themen:

- Datenkraken vs. Alternativen: Google vs. DuckDuckGo/Ecosia
- Fake News & Medienkompetenz: Propaganda & Manipulation (Beispiele: Wahlkampf, Deepfakes)
- Alternativen zu Big Tech: NewPipe, Fediverse
- Messenger absichern: Facebook Messenger mit Pidgin, OTR & Tor
- Passwortsicherheit & KeePass: Sichere Passwörter erstellen
- Erste Schritte mit Scratch: Grundlagen der Programmierlogik

Praxis:

- KeePass einrichten & sichere Passwörter testen
- Erste Scratch-Projekte: Bedingte Anweisungen, Schleifen, Events

9.3 Tag 3: Web-Technologien, Python & KI-Grundlagen

Ziel: Grundlagen für Webentwicklung & KI-Programmierung schaffen.

Themen:

- Grundlagen von HTML & CSS (mit Unterstützung durch Duck.AI)
- Einrichten eines Webservers (NGINX)
- Einführung in Python: Syntax, Variablen, Funktionen
- Python-Umgebungen verwalten: PIP, virtuelle Umgebungen
- Was ist KI? (Einsatzgebiete, Vorteile, Herausforderungen)

Praxis:

- Eigene erste HTML-Seite mit CSS stylen
- Flask-Server einrichten
- Erste Python-Programme schreiben

9.4 Tag 4: Hands-on mit KI – Erste Modelle testen

Ziel: Erste KI-Modelle zum Laufen bringen und verstehen, wie sie funktionieren.

Themen:

- Installation & Nutzung von PyTorch (CPU-Version)
- KI-Modelle & ihre Funktionsweise
- Erste Schritte mit:
 - BLIP – Bilderkennung
 - DistilBERT – Textverarbeitung
 - GPT-2 – Textgenerierung
 - EasyOCR – Texterkennung
 - Real-ESRGAN – Bildverbesserung

Praxis:

- KI-Modelle installieren & einfache Tests durchführen
- Flask-Apps für KI-Modelle starten

9.5 Tag 5: Eigenes KI-Projekt entwickeln (Gruppenarbeit)

Ziel: Die Teilnehmer setzen ihr eigenes KI-Projekt um.

Themen:

- Projektplanung: Welche KI-Modelle wählen die Gruppen?
- Problemlösung mit KI: Duck.AI zur Unterstützung nutzen
- Code nicht nur kopieren, sondern verstehen & anpassen

- Feinschliff am Webinterface & Präsentation vorbereiten

Praxis:

- Jede Gruppe setzt ihr eigenes KI-Projekt um (z. B. Bilderkennungstool, Fake-News-Detektor, Textgenerator, Bildverbesserungstool)
- Testen, Debuggen & Vorbereiten auf die Präsentation

9.6 Tag 6 (Halber Tag): Projektvorstellung & Reflexion

Ziel: Ergebnisse präsentieren & Erfahrungen reflektieren.

Themen:

- Jede Gruppe stellt ihr Projekt vor
- Feedbackrunde: Was war einfach, was war schwierig?
- Missbrauch von KI: Diskussion über Deepfakes, politische Manipulation & Betrug
- Fazit & mögliche Weiterentwicklung nach dem Workshop

Praxis:

- Präsentation der Gruppenprojekte
- Diskussion über ethische Fragen & KI-Zukunft

10 Diskussionsfragen für den Workshop

IT-Grundlagen & Open Source

1. Warum ist Open-Source-Software wichtig? Welche Vorteile hat sie gegenüber proprietärer Software?
2. Kannst du Beispiele für Open-Source-Projekte nennen, die du selbst nutzen würdest?
3. Welche Rolle spielt Open Source in der heutigen IT-Welt?
4. Warum setzen große Unternehmen wie Google, Microsoft oder Tesla trotz eigener Softwareentwicklungen auf Open-Source-Technologien?
5. Sollte mehr Open-Source-Software in Schulen unterrichtet werden? Warum oder warum nicht?

Datenschutz & Social Media

1. Wie viel Wert legst du persönlich auf Datenschutz? Ändert sich deine Meinung, wenn du über Überwachung und Tracking nachdenkst?
2. Welche Daten gibst du bewusst oder unbewusst auf Social Media preis?
3. Findest du, dass Social-Media-Plattformen genug tun, um deine Daten zu schützen? Falls nicht, was könnte verbessert werden?
4. Sollte es strengere Gesetze für den Umgang mit Nutzerdaten geben? Warum (nicht)?
5. Was hältst du von dem Argument: „Ich habe nichts zu verbergen, also ist Datenschutz für mich egal“?

Programmierung & Automatisierung

1. Warum ist es heute so wichtig, grundlegende Programmierkenntnisse zu haben, selbst wenn man nicht Informatiker werden will?
2. Welche Bereiche in deinem Alltag könnten durch Automatisierung verbessert werden?
3. Denkst du, dass jeder Mensch programmieren lernen sollte, ähnlich wie Lesen und Schreiben? Warum oder warum nicht?
4. Welche Programmiersprachen werden deiner Meinung nach in der Zukunft besonders relevant sein?
5. Glaubst du, dass Künstliche Intelligenz in Zukunft Programmierer ersetzen wird?

Künstliche Intelligenz & Ethik

1. Wo begegnet dir KI im Alltag, ohne dass du es vielleicht bewusst wahrnimmst?
2. Welche KI-Anwendungen findest du besonders hilfreich? Gibt es auch welche, die du kritisch siehst?
3. Sollte KI reguliert werden? Falls ja, wer sollte entscheiden, was erlaubt ist und was nicht?
4. Ist es ethisch vertretbar, dass KI kreative Berufe (z. B. Künstler, Musiker) beeinflusst?

5. Glaubst du, dass KI eine größere Gefahr oder eine größere Chance für die Gesellschaft darstellt?

Deepfakes, Manipulation & Fake News

1. Wie kann man als normaler Internetnutzer Fake News und Deepfakes erkennen?
2. Welche Konsequenzen können Deepfake-Videos für Politik und Gesellschaft haben?
3. Sollte es Gesetze gegen den Missbrauch von KI-generierten Inhalten geben? Wo zieht man die Grenze zwischen Kunst und Manipulation?
4. Glaubst du, dass Menschen in der Zukunft noch zwischen echten und gefälschten Inhalten unterscheiden können?
5. Was wäre eine sinnvolle Methode, um gegen die Verbreitung von Fake News vorzugehen?

Zukunft der Arbeit & KI in Berufen

1. Welche Berufe könnten durch KI in den nächsten Jahren stark verändert oder sogar ersetzt werden?
2. Glaubst du, dass KI eher Arbeitsplätze vernichtet oder neue schafft?
3. In welchen Berufen könnte der Mensch nie durch eine KI ersetzt werden?
4. Sollten Schüler und Studenten mehr über KI lernen, um auf zukünftige Berufe vorbereitet zu sein?
5. Wird KI uns das Leben erleichtern oder uns abhängiger machen?

Kreativität & KI

1. Kann KI wirklich kreativ sein, oder imitiert sie nur menschliche Kreativität?
2. Was hältst du von KI-generierter Kunst oder Musik? Sollte sie als „echte“ Kunst anerkannt werden?
3. Wie würdest du dich fühlen, wenn dein Lieblingsbuch oder -song von einer KI geschrieben wurde?
4. Sollte KI-generierte Kunst gekennzeichnet werden, damit man sie von menschlichen Werken unterscheiden kann?
5. Kann KI als Werkzeug dienen, um Menschen kreativer zu machen, oder nimmt sie ihnen die kreative Arbeit weg?

Cybersicherheit & Hacking

1. Was bedeutet „ethisches Hacking“ und wo liegt die Grenze zwischen gutem und bösem Hacken?
2. Was sind die größten Bedrohungen für die Cybersicherheit? Hast du selbst schon mal einen Angriff (z. B. Phishing) erlebt?
3. Warum sind sichere Passwörter und Zwei-Faktor-Authentifizierung so wichtig?
4. Sollte Hacking als Schulfach unterrichtet werden, um Cyberangriffe besser zu verstehen?

5. Wie könnte man sich besser vor Cyberangriffen schützen?