Ausarbeitung – Interaktives Lern-Tool für neuronale Netze

# Arbeitsaufteilung

|  |  |  |
| --- | --- | --- |
| Phase | Patrick | Lukas |
| Analysieren | Ausarbeitung Idee | Ausarbeitung Idee |
|  | Erstellung Personas |
|  | Erstellung Szenario |
| Interpretieren | Recherche nach Tools | Erstellung Figma Kontext |
|  | Testen NN (neuronales Netz) from Scratch |  |
| Kreieren | Tests mit ThreeJS (Kreise und Linien | Erstellung Wheel und Slider Interaktion mit ThreeJS |
|  | Tests Tensorflow |  |
|  | Ergänzen von Infos bei Zwischenpräsentation | Umsetzung Zwischenpräsentation |
|  | Dynamisches Erzeugen von Layers + Tensorflow | Erzeugung TikTakToe und Studentenwohnheim Datensatz |
|  | Tests-NN TikTakToe Problem | Tests-NN TikTakToe Problem + Tests Studentenwohnheim |
|  | Gestaltung von Prototypen für Experteninterview | Umsetzung eines Prototyps für das Experteninterview (Interaktion + Studenten NN) |
|  | Debuggen Prototyp bei Generierung von Ergebnissen in Output-Layer | Erzeugung von Ergebnissen, nachdem neuer Input ausgewählt wurde |
| Bewerten | Vorbereitung + Durchführung von Experteninterview | Erstellung Fragebogen für Interview + Anfragen |
| Analysieren | Auswertung Interview |  |
| Interpretieren | Planung aus dem Ergebnis des Interviews |  |
| Kreieren | Vorbereiten des gemeinsamen Standes für die Einarbeitung der Ergebnisse des Interaktionsprototypen | Einarbeitung der Funktionen des Interatkionsprototypen |
| Bewerten | Durführung des AB-Tests | Durchführung des AB-Tests |

# Idee: Patrick und Lukas

Wir haben zusammen eine Idee evaluiert, in welcher es darum geht ein Tool zu erstellen, mit welchem durch einen konstruktivistischen Lernansatz, Lernende ein neuronales Netz durch Interaktion kennenlernen. Es gibt bereits ähnliche Ansätze Tools ([Interaktiv: Neuronales Netz – Hart und Trocken](https://www.hartundtrocken.de/my-product/interaktiv-neuronales-netz/)), jedoch noch keins bei welchem die internen Parameter des Netzes angepasst werden können.

# Arbeitsbereich Lukas Willmann

## Analysieren 1

Für den ersten Abschnitt habe ich vier Personas erstellt. ([ID\_XAI\_TS/Documents/Personas at main · Z3r0cks/ID\_XAI\_TS (github.com)](https://github.com/Z3r0cks/ID_XAI_TS/tree/main/Documents/Personas)) Die Personas bilden lernwillige Personen ab, welche sich fortbilden wollen. Anhand dieser Personas erstellte ich ein Szenario für eine der Personas in einem möglichen Studenten-Alltag. ([ID\_XAI\_TS/Documents/Szenario at main · Z3r0cks/ID\_XAI\_TS (github.com)](https://github.com/Z3r0cks/ID_XAI_TS/tree/main/Documents/Szenario))

## Interpretieren 1

Mithilfe der Personas konnte ein „klickbarer“-Figma Prototyp erstellt werden, welcher für uns den Use-Case verdeutlichte. ([NN\_Explainer – Figma](https://www.figma.com/file/IPAPgOgXqxD9lLfVlNVUVv/NN_Explainer?type=design&node-id=0-1&mode=design)) Lernende würden an einem Desktop-PC auf einer Website Informationen über Neuronale Netze in verschiedenen Kapitel abrufen können. Auf dieser Website wäre das interaktive Tool als eigenes Kapitel oder am Ende eines passenden Kapitels eingebettet.

In unserem Projekt konzentrierten wir uns auf die Umsetzung des Tools. Nicht auf die komplette Umsetzung der Website mit dem Tool, da dies den Rahmen für uns gesprengt hätte. Dieser Figma-Kontext half uns aber das Tool auf bestimmte Weise zu entwerfen, wie z.B. die Beachtung von verschiedenen Desktop-Größen (Kamera-Handling in der Szene), jedoch nicht die Auslegung für Mobile, da wir davon ausgehen, dass die Nutzer für eine konzentrierte Lernatmosphäre eher an Desktop-PCs sitzen mit einer Scroll-Möglichkeit (Maus oder Touchpad eines Laptops).

## Kreieren 1

Damit die Kamera zentral auf die Layers gerichtet ist in Abhängigkeit von der Anzahl der Neuronen wie viele in einem Layer sind und wie viele Layer es gibt habe ich eine Kamera-Steuerungsklasse erstellt, mit welcher das neuronale Netz zentriert im Bild bleibt. ([ID\_XAI\_TS/src/classes/CameraHandler.ts at main · Z3r0cks/ID\_XAI\_TS (github.com)](https://github.com/Z3r0cks/ID_XAI_TS/blob/main/src/classes/CameraHandler.ts))

Um die Kommunikation zwischen unseren Programmierteilen zu verbessern habe ich ein Klassendiagramm erzeugt. ([ID\_XAI\_TS/Documents/interactUML.drawio at main · Z3r0cks/ID\_XAI\_TS (github.com)](https://github.com/Z3r0cks/ID_XAI_TS/blob/main/Documents/interactUML.drawio))

Die Interaktionen des Nutzers werden über die Klassen in der Interaktions-Lib gesteuert. ([ID\_XAI\_TS/src/classes/InteractionLib at main · Z3r0cks/ID\_XAI\_TS (github.com)](https://github.com/Z3r0cks/ID_XAI_TS/tree/main/src/classes/InteractionLib)) Durch einen Parameter in der „InteractionFactory“ ([ID\_XAI\_TS/src/classes/InteractionLib/InteractionFactory.ts at main · Z3r0cks/ID\_XAI\_TS (github.com)](https://github.com/Z3r0cks/ID_XAI_TS/blob/main/src/classes/InteractionLib/InteractionFactory.ts)) kann gesteuert werden, ob deine „Wheel“ oder eine „Slider“ Interaktion erzeugt werden soll. Diese Klassen wurden zuerst in einem JavaScript Projekt erstellt. Bei der Umstellung auf TypeScript wurden die Klassen aber angepasst.

Um Datensätze zu haben mit welchen wir NNs trainieren konnten erzeugte ich einmal einen TikTakToe-Datensatz und einen Wohnungen-Datensatz. Diese wurden durch Brute-Force-Python Scripte erzeugt. Bei dem TikTakToe Datensatz wurde viel mit Zufalls-Zahlen gearbeitet und in dem Datensatz der Wohnungen wurden logische Regeln und ebenfalls Zufallszahlen verwendet um die Daten zu generieren und zu labeln. ([ID\_XAI\_TS/DataSets at main · Z3r0cks/ID\_XAI\_TS (github.com)](https://github.com/Z3r0cks/ID_XAI_TS/tree/main/DataSets))

Um die NNs zu testen welche aus den Datensätzen entstehen können habe ich mit Jupyter-Notebooks gearbeitet. ([ID\_XAI\_TS/DataSets/JupyterTests at main · Z3r0cks/ID\_XAI\_TS (github.com)](https://github.com/Z3r0cks/ID_XAI_TS/tree/main/DataSets/JupyterTests))

Da die Kombination zwischen unserem Hauptprojekt und dem Projekt an welchem ich mit den Interaktion gearbeitet habe zu komplex war um in der Zeit bis zur Endpräsentation zu kombinieren, habe ich einen ersten testbaren Prototyp in meinem Projekt erzeugt. Das Modell ist in seinen Layern, Neuronen und Output Möglichkeiten statisch. Das Modell wird beim Start 200 mal mit dem Wohnungs-Datensatz trainiert. Danach kann der Tester die beiden Interaktionen unter Anleitung testen. [LksWllmnn/NeuroInteract: Interaction with a small Neuronal Network (github.com)](https://github.com/LksWllmnn/NeuroInteract)

## Bewerten 1

Um unsere wissenschaftliche Fragen für das Projekt zu bearbeiten habe ich einen einfachen Fragebogen erstellt. ([ID\_XAI\_TS/Documents/Fragebogen Interaktion der Anwendung KI.pdf at main · Z3r0cks/ID\_XAI\_TS (github.com)](https://github.com/Z3r0cks/ID_XAI_TS/blob/main/Documents/Fragebogen%20%20Interaktion%20der%20Anwendung%20KI.pdf)) Im ersten Teil des Fragebogens gibt es qualitative Fragen zur Nutzung des Prototypen. Hierbei war die Intuition, dass wir ein genaueres Verständnis bekommen welche Interaktion für NNs angenehmer ist und nicht nur ob eine Testperson das Mausrad oder einen Slider lieber mag. Im zweiten Teil haben wir den wissenschaftlichen „SUS“-Fragebogen verwendet um einen wissenschaftlich Bewährten Fragebogen für unsere Evaluation heran ziehen zu können. ([(PDF) SUS: A quick and dirty usability scale (researchgate.net)](https://www.researchgate.net/publication/228593520_SUS_A_quick_and_dirty_usability_scale))