



Pädagogische Hochschule Weingarten  
Master Medien- und Bildungsmanagement  
Wintersemester 2025/2026

**Pädagogische  
Hochschule  
Weingarten  
University of  
Education**

## Projektbericht M2



---

### Interaktive Visualisierung von Objekterkennung - wie denkt die Maschine?

---

Prüfer: Müller, Dr.-Ing. Wolfgang; Franke, Stefan

Abgabedatum: 12.02.2026

Vorgelegt von: Bolach, Jan-Ove; Dietenmaier, Felix; Gaugler, Franziska

Matrikelnummer: 7150775, 7208038, 7201437

## 1. Projektmanagement

Die Entwicklung der IMX500-Anwendung erfolgte nach einem agilen Projektmanagement-Ansatz, da dieser besonders geeignet ist, wenn technische und didaktische Anforderungen erst im Projektverlauf konkretisiert werden können (Habermann, 2013; Kusay-Merkle, 2021). Die agile, change-getriebene Arbeitsweise ermöglichte es dem Team, flexibel auf technische Veränderungen, wie den Wechsel des Betriebssystems oder der Kamera, zu reagieren, ohne den Projektfluss wesentlich zu beeinträchtigen (Gerstner, 2024).

Die Arbeit in iterativen Zyklen ermöglichte die kontinuierliche Bereitstellung lauffähiger Software-Inkremeante (GPM Deutsche Gesellschaft für Projektmanagement e. V., 2024). Dies erlaubte ein stetiges Feedback der Stakeholder (Herr Müller und Herrn Franke) wodurch das Endprodukt präzise auf die didaktischen Anforderungen zugeschnitten werden konnte. Gleichzeitig minimierte die regelmäßige Evaluation das Risiko von Fehlentwicklungen und förderte durch transparente Kommunikation die Selbstorganisation im Team. In einem Umfeld, das planbare Prüfungsanforderungen mit komplexen Innovationen wie der IMX500 AI Camera vereint, erwies sich die Verbindung agiler Werte mit strukturierten Rahmenbedingungen als vorteilhaft (Gerstner, 2024; Habermann, 2013; Kusay-Merkle, 2021).

Agiles Projektmanagement beschreibt eine flexible, wertebasierte Denkweise zur Bewältigung von Komplexität und Unsicherheit, während Scrum ein konkretes Rahmenwerk (Framework) zur operativen Umsetzung dieser Prinzipien darstellt (Gerstner, 2024; Schwaber & Sutherland, 2020). Als Framework fiel die Entscheidung auf Scrum. Da jedoch nicht alle Elemente implementiert wurden, darf das Vorgehen laut Schwaber und Sutherland (2020) formal nicht als Scrum bezeichnet werden, da das Rahmenwerk in seiner Gesamtheit unveränderlich ist und das Weglassen von Elementen den Nutzen einschränkt. Das angewandte Framework lässt sich daher eher als „ScrumBut“ klassifizieren. Die Gründe für diese Anpassung liegen in den Rahmenbedingungen eines studentischen Projekts: Da keine Vollzeitverfügbarkeit bestand, waren Daily Scrums nicht realisierbar, und aus Zeitgründen entfiel die Sprint Retrospective. Zudem wurde auf ein Sprint Backlog verzichtet und ausschließlich mit einem Product Backlog (siehe Anhang) gearbeitet. Die aus dem Scrum-Rahmenwerk konsequent übernommenen Elemente sind die Arbeit in Sprints, das Sprint Planning, das Sprint Review sowie das Product Backlog inklusive Produkt-Ziel und Inkrementen (Schwaber & Sutherland, 2020).

Das Projektteam orientierte sich grundsätzlich an den im Scrum Guide definierten Rollen und Verantwortlichkeiten. Aufgrund der geringen Teamgröße wich das Vorgehen jedoch vom klassischen Scrum ab, da die Arbeit stärker interdisziplinär organisiert war. Jan-Ove übernahm schwerpunktmäßig die Rolle des Developers und war insbesondere in der Endphase des Projekts für die Umsetzung der Software und die Erstellung der Inkremeante verantwortlich. Die Rollen des Product Owners und des Scrum Masters wurden von Franziska und Felix gemeinsam wahrgenommen. Zentrale Entscheidungen wurden im Team abgestimmt und gemeinsam getroffen. Als externe Stakeholder fungieren Herr Müller und Herr Franke. Als langfristiges Planungsziel (Product Goal) wurde festgelegt, dass das Lern artefakt Schülerinnen und Schüler (SuS) sowie Studierende dazu befähigt, die einzelnen Schritte einer Objektkennungs-KI anhand eines eigenen Kamerabildes interaktiv nachzuvollziehen und dadurch die Funktionsweise von KI im Bereich der Objektkennung kritisch zu verstehen.

Zur Qualitätssicherung wurden folgende Kriterien festgelegt, die im Sinne einer Definition of Done die Fertigstellung des Inkrementen definieren:

1. Lernzielerreichung: Mindestens 80 % der Nutzenden können spontan 3-4 Prozessschritte (z. B. Input, Inferenz) korrekt benennen.
2. Verständnis: Mindestens 70 % bewerten ihr Verständnis von KI-Objekterkennung nach der Nutzung als „besser“ oder „viel besser“.
3. Zielgruppendifferenzierung: Es existieren zwei Pfade (Einsteiger/Experten) mit spezifisch angepassten Erklärungen.
4. Usability (Walk-up-and-use): Die Erstnutzung eines Pfades ist ohne externe Anleitung innerhalb einer Minute möglich.
5. Stabilität: Das System läuft im Deployment-Setup (Raspberry Pi 4 + IMX500) mindestens eine Stunde fehlerfrei im Dauerbetrieb.
6. Verfügbarkeit: Das Artefakt ist als Git-Repository öffentlich zugänglich und lässt sich im Sinne des Open-Source-Gedankens auf äquivalenter Hardware einfach klonen und installieren.

Da sich das Lernartefakt aktuell noch im Prototypen-Stadium befindet, lag der Schwerpunkt der bisherigen Arbeit auf der technischen Umsetzung und der Sicherstellung der grundlegenden Funktionalität (Kriterien 3 bis 6). Eine Evaluation der Kriterien 1 und 2 konnte im Rahmen dieser Dokumentation daher noch nicht durchgeführt werden, da hierfür eine Erprobung mit der Zielgruppe am fertigen System notwendig ist. Die Überprüfung der Lernwirksamkeit und des Verständnisses bleibt somit ein offener Schritt für die weitere Entwicklung.

## 2. Anforderungsanalyse

Eine genaue Analyse der Zielgruppe ist Grundlage der didaktischen Konzeption, da digitale Lernangebote häufig scheitern, wenn sie nicht an die Voraussetzungen der Lernenden angepasst sind. Im Gegensatz zum Präsenzunterricht sind die Nutzenden in digitalen Formaten meist nicht persönlich bekannt, und nachträgliche Änderungen am fertigen Lernangebot sind nur mit hohem Aufwand möglich (Kerres, 2013). Daher werden zunächst die soziodemografischen Rahmenbedingungen sowie die Lernvoraussetzungen der Zielgruppen betrachtet.

Unterschieden wird zwischen einer Primärzielgruppe (SuS) und einer Sekundärzielgruppe (Studierende), die sich insbesondere im Vorwissen und in ihren Nutzungsmotiven unterscheiden. Zur Primärzielgruppe gehören studieninteressierte SuS der Sekundarstufe II (15-18 Jahre). Das Lernangebot wird in einem informellen Kontext genutzt (z. B. Studieninformationstag) und freiwillig sowie zeitlich begrenzt wahrgenommen. Dies setzt zwar eine hohe intrinsische Motivation voraus, geht jedoch mit einer geringen Frustrationstoleranz einher. Da kaum informatikbezogene Vorkenntnisse vorhanden sind und Objekterkennung oft als „Black Box“ wahrgenommen wird, ist ein spielerisch-entdeckender Zugang mit direktem Feedback notwendig. Eine intuitive Bedienung sowie grundlegende Designprinzipien (z. B. große Klickflächen, hoher Kontrast) sind für die Akzeptanz entscheidend.

Die Sekundärzielgruppe umfasst Studierende der Pädagogischen Hochschule, überwiegend aus Lehramtsstudiengängen. Sie nutzen das Lernangebot mit Blick auf die berufliche Relevanz und didaktische Einsatzmöglichkeiten. Obwohl auch hier meist nur grundlegende technische Kenntnisse vorliegen, erlaubt die höhere Abstraktionsfähigkeit eine vertiefte Auseinandersetzung mit Fachbegriffen und technischen Zusammenhängen. Um Unterförderung zu vermeiden, sollte das System für diese Gruppe fachlich korrekt vertiefende Informationen sowie transparente technische Leistungsdaten bereitstellen und so Anknüpfungspunkte für weiterführendes Interesse bieten.

Um soziodemografische Daten sinnvoll zusammenzuführen und die Zielgruppe für die didaktische Planung greifbarer zu machen, werden Personas eingesetzt. Sie bündeln zentrale Merkmale zu typischen Nutzerprofilen und machen Erwartungen sowie Ziele der Lernenden verständlich (Kerres, 2013). Daher wurden für beide Zielgruppen jeweils zwei Personas entwickelt, um unterschiedliche Voraussetzungen und Nutzungsmotive abzubilden.

Julian (17) repräsentiert den explorativen Entdeckertyp, der das System spielerisch und manipulativ austesten möchte. Sarah (16) hingegen steht für den reflektierenden Lerntyp, der primär an der transparenten Nachvollziehbarkeit der KI-Entscheidungsprozesse („Glass Box“) interessiert ist. Die Sekundärzielgruppe wird durch Lisa (23) abgebildet, die das Exponat im Hinblick auf didaktische und ethische Fragen bewertet, sowie durch Jonas (25), der an technischen Details und der fachlichen Umsetzung interessiert ist.

Auf Basis der detaillierten Zielgruppenanalyse und der Charakterisierung durch Personas wurden konkrete Nutzungsszenarien und User Stories entwickelt. Diese narrativen Methoden dienten dazu, die Interaktion mit dem Exponat im realen Kontext (z. B. am Studieninformationstag) zu simulieren und die spezifischen Bedürfnisse der Nutzergruppen greifbar zu machen. Durch die Durchspielung dieser Szenarien, etwa der spielerische Erstkontakt durch Persona Julian oder die reflexive Suche nach Erklärungen durch Persona Sarah, konnten kritische Erfolgsfaktoren für die User Experience identifiziert werden. Die zentralen Erkenntnisse aus diesen User Stories wurden in die folgenden funktionalen und didaktischen Anforderungen an das Lernartefakt überführt:

- Niedrige Einstiegshürde: Direkter Start im Live-View ohne Menüführung oder Kalibrierung (Niegemann & Heidig, 2020)
- Unmittelbares Feedback: Reaktionszeiten < 100 ms (Tesar et al., 2013)
- Intuitive Bedienbarkeit: Große Bedienelemente, klare Icons (MacKenzie, 2018)
- Multimodale Darstellung: Kombination aus Bild und optionaler Audioausgabe (Kerres, 2013)
- Barrierefreiheit: Hoher Kontrast, Farbcodierung nie alleinstehend (Kerres, 2013)
- Spielerische Motivation: Zähler, Sammel- und Vergleichselemente statt abstrakter Diagramme (Weinert et al., 2021)
- Sprachliche Zugänglichkeit: Alltagsnahe Begriffe statt technischer Fachvariablen (Kerres, 2013)
- Strukturierte Lernunterstützung: Schrittweise Freischaltung von Funktionen (Scaffolding) (Kerres, 2013)

Damit werden sowohl spielerische als auch verstehensorientierte Zugänge der SuS berücksichtigt und ein niedrigschwelliger, motivierender Einstieg in das Thema KI ermöglicht.

Basierend auf der Analyse der Sekundärzielgruppe wurden ebenfalls Nutzungsszenarien entwickelt, die die Interaktion unter dem Aspekt künftiger Professionalität simulieren. Die Erkenntnisse aus den Perspektiven der reflexiven Prüfung (Lisa) und technischen Validierung (Jonas) definierten den Anspruch an fachliche Tiefe und berufliche Relevanz. Dies führte zu folgenden Ableitungen:

- Fachsprachliche Präzision: Aktivierbarer „Studenten-Modus“ mit korrekten Fachtermini (z. B. Inferenz, Output-Tensor) und elaborierten Erklärungstexten (Kerres, 2013)
- Transparenz technischer Daten: Echtzeit-Anzeige von Inference Time, differenzierten Confidence-Werten sowie der RGB-Werte einzelner Bildpunkte zur transparenten Nachvollziehbarkeit der Datenbasis (Niegemann & Heidig, 2020)

- Reflexive Kontextualisierung: Hinweise auf Bias, Trainingsdaten und Fehlklassifikationen (z. B. COCO-Datensatz)
- Professionelles Interface: Reduktion spielerischer Elemente zugunsten eines nüchternen, datenorientierten „Clean UI“ (Niegemann & Heidig, 2020)
- Transferorientierung: Verweise auf GitHub-Repository, Module oder Studienangebote (z. B. QR-Codes)
- Erweiterte Transparenz: Einblendung von Zwischenschritten und exaktere Darstellung des technischen Vorgangs

Diese Ableitungen stellen sicher, dass das Lernartefakt den Erwartungen der Studierenden an fachliche Tiefe, Transparenz und berufliche Anschlussfähigkeit gerecht wird und über eine reine Demonstration hinaus zu reflektiertem Verständnis von KI-Systemen beiträgt.

### 3. Entwicklung

Die Realisierung des Lernartefakts erfolgte auf Basis eines Raspberry Pi 4 in Kombination mit der Raspberry AI Camera (Sony IMX500). Der Entwicklungsprozess gliederte sich dabei in zwei wesentliche Phasen. In der Initialphase (Proof of Concept) wurde eine softwareseitige Objekterkennung mittels TensorFlow Lite (TFLite) und OpenCV auf der CPU implementiert. Dieser Ansatz diente dem grundlegenden Verständnis der Pipeline, stieß jedoch unter älteren Betriebssystemversionen auf massive Kompatibilitätsprobleme zwischen dem libcamera-Stack und den TFLite-Interpretern, was eine stabile Einbindung der AI Camera verhinderte.

Der technologische Durchbruch erfolgte im Januar mit dem Wechsel auf Raspberry Pi OS Bookworm (64-bit) und der Abkehr von CPU-basierter Inferenz. Die Architektur wurde vollständig auf die native Nutzung des IMX500-Hardware-Beschleunigers umgestellt. Anstelle externer TFLite-Modelle nutzt das System nun die Bibliothek picamera2, um direkt auf die On-Chip-Inferenz des Sensors zuzugreifen. Die Kamera liefert hierbei keine Rohdaten, sondern verarbeitete Metadaten (Tensoren, Bounding Boxes, Confidence Scores), was die Systemlast des Raspberry Pi drastisch reduziert.

Die volle Funktionalität des Artefakts können Sie sich [hier](#) ansehen.

### 4. Limitation und Ausblick

Das entwickelte Lernartefakt stellt einen technisch funktionsfähigen Prototyp dar, dessen Entwicklungsprozess noch nicht abgeschlossen ist. Eine zentrale Limitation besteht in der bislang fehlenden empirischen Evaluation sowohl in formativer als auch in summativer Form. Die getroffenen didaktischen Entscheidungen basieren derzeit ausschließlich auf theoretischen Modellen und Experteneinschätzungen und konnten noch nicht durch reale Nutzungsdaten der Primärzielgruppe überprüft werden.

Darüber hinaus besteht Weiterentwicklungsbedarf im Bereich der Interaktivität und Ergonomie. Für explorativ orientierte Lerntypen fehlt bislang die Möglichkeit, zentrale Parameter wie den Confidence Threshold zu verändern und dadurch die Sensitivität der Objekterkennungs-KI unmittelbar erfahrbar zu machen. Zudem sollte die User Experience stärker auf eine intuitive Touch-Bedienung ausgerichtet werden, um den Nutzungserwartungen der „Mobile-First“-Generation gerecht zu werden. Zukünftige Iterationen sollten diese Aspekte im Rahmen eines Feldtests im Naturwissenschaftlichen Zentrum gezielt aufgreifen und weiterentwickeln.

## Literaturverzeichnis

- Gerstner, J. (5. Februar 2024). Klassisch vs. agil: die Projektmanagement–Methoden im Vergleich. *cplace*. <https://www.cplace.com/blog/klassisch-vs-agil-die-projektmanagement-methoden-im-vergleich/>
- GPM Deutsche Gesellschaft für Projektmanagement e. V. (2024). *Agiles vs. Klassisches Projektmanagement: Eine Gegenüberstellung | GPM Deutsche Gesellschaft für Projektmanagement e. V.* <https://www.gpm-ipma.de/ueber-die-gpm/blog/agiles-vs-klassisches-projektmanagement-eine-gegenueberstellung>
- Habermann, F. (2013). Hybrides Projektmanagement — agile und klassische Vorgehensmodelle im Zusammenspiel. *HMD Praxis der Wirtschaftsinformatik*, 50(5), 93–102. <https://doi.org/10.1007/BF03340857>
- Kerres, M. (2013). *Mediendidaktik: Konzeption und Entwicklung mediengestützter Lernangebote* (4., überarb. und aktualisierte Aufl.). Oldenbourg.
- Kusay-Merkle, U. (2021). *Agiles Projektmanagement im Berufsalltag: Für mittlere und kleine Projekte* (2., erweiterte und verbesserte Auflage). Springer Gabler. <https://doi.org/10.1007/978-3-662-62810-2>
- MacKenzie, I. S. (2018). Fitts' Law. In K. L. Norman & J. Kirakowski (Hrsg.), *The Wiley handbook of human computer interaction set* (S. 347–370). Wiley Blackwell. <https://doi.org/10.1002/9781118976005.ch17>
- Niegemann, H. M. & Heidig, S. (2020). Interaktivität und Adaptivität in multimedialen Lernumgebungen. In H. M. Niegemann & A. Weinberger (Hrsg.), *Springer eBook Collection. Handbuch Bildungstechnologie: Konzeption und Einsatz digitaler Lernumgebungen* (Living reference work, continuously updated edition, S. 1–25). Springer. [https://doi.org/10.1007/978-3-662-54373-3\\_33-1](https://doi.org/10.1007/978-3-662-54373-3_33-1)
- Raspberry Pi Ltd. (o. D.). Raspberry Pi AI Camera. Raspberry Pi Documentation. <https://www.raspberrypi.com/documentation/accessories/ai-camera.html>
- Schwaber, K. & Sutherland, J. (2020). *Der Scrum Guide: Der gültige Leitfaden für Scrum: Die Spielregeln*. <https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-German.pdf>
- Tesar, M., Stöckelmayr, K., Pucher, R., Ebner, M., Metscher, J. & Vohle, F. (2013). Multimodale und interaktive Materialien. Gestaltung von Materialien zum Lernen und Lehren. *L3T. Lehrbuch für Lernen und Lehren mit Technologien*. Vorab-Onlinepublikation. <https://doi.org/10.25656/01:8335>
- Weinert, T., Benner, D., Dickhaut, E., Janson, A., Schöbel, S. & Leimeister, J. M. (2021). Unterstützung digitaler Bildungsprozesse durch interaktive gamifizierte Lernvideos – Wie innovative Lernvideos Motivation und Lernerfolg steigern können [Supporting Digital Education Processes Through Interactive Gamified Learning Videos-How Innovative Learning Videos can Increase Motivation and Learning Success]. *HMD Praxis der Wirtschaftsinformatik*, 58(6), 1483–1503. <https://doi.org/10.1365/s40702-021-00798-w>

## KI-basierte Hilfsmittel

KI-basiertes Hilfsmittel	Einsatzform	Betroffene Teile der Arbeit	Link
Adobe. (2025). Adobe Firefly (Firefly Image Model 5) [Generatives KI-Modell]	Generierung visueller Inhalte	Grafiken und Abbildungen im Lernartefakt	<a href="https://www.adobe.com/products/firefly.html">https://www.adobe.com/products/firefly.html</a>
Adobe. (2025). Adobe Firefly Generate Speech [Text-to-Speech-Modell]	Generierung synthetischer Sprachausgabe	Audiospuren im Lernartefakt	<a href="https://www.adobe.com/products/firefly/features/text-to-speech.html">https://www.adobe.com/products/firefly/features/text-to-speech.html</a>
Google. (2025). Gemini (Gemini 2.5 Flash Image with Nano Banana) [Generatives KI-Modell]	Generierung visueller Inhalte	Grafiken und Abbildungen	<a href="https://gemini.google.com">https://gemini.google.com</a>
Google. (2025). Gemini (Gemini 2.5) [Large Language Model]	Unterstützung bei Programmierung und sprachlicher Überarbeitung	Quellcode und Projektbericht	<a href="https://gemini.google.com">https://gemini.google.com</a>
OpenAI. (2026). ChatGPT (Version GPT-5.2) [Large Language Model]	Sprachliche Überarbeitung und Strukturierung von Texten	Projektbericht	<a href="https://chat.openai.com">https://chat.openai.com</a>
Perplexity AI. (2025). Perplexity (Sonar) [KI-gestützte Suchmaschine]	Unterstützung bei Programmierung, Rechercheunterstützung und Quellenfindung	Projektbericht und technische Recherche	<a href="https://www.perplexity.ai">https://www.perplexity.ai</a>

## Anhang

Sprint	Beginn	Ende	Sprint Goal	Sprint Items	Status	Verantwortlich
Product Goal			Das Lernartefakt befähigt SuS und Studierende, die Schritte einer Objekterkennungs-KI an einem eigenen Kamerabild interaktiv nachzuvollziehen und dadurch die Arbeitsweise von KI im Kontext Objekterkennung kritisch zu verstehen.			
Sprint 0	09.10.2025	23.10.2025	Ein gemeinsames Projektverständnis schaffen und die Grundlagen für das Projekt legen.	Rollen verteilen Product Goal skizzieren Technologien festlegen Projektmanagementmethode wählen Erstes Backlog entwerfen	Done Done Done Done Done	Jan-Ove/Franzi/Felix Jan-Ove/Franzi/Felix Jan-Ove/Franzi/Felix Jan-Ove/Franzi/Felix Felix
Sprint 1	24.10.2025	06.11.2025	Ein gemeinsames Projektverständnis schaffen, die Projektskizze fertigstellen und den Raspberry Pi grundlegend in Betrieb nehmen.	Projektskizze erstellen Recherche durchführen Basis-Setup (OS, Python, erste Pakete) RP kennenlernen SSH Verbindung einrichten	Done Done Done Done Done	Franzi Jan-Ove/Franzi/Felix Jan-Ove/Franzi/Felix Jan-Ove/Franzi/Felix Jan-Ove/Franzi/Felix
Sprint 2	07.11.2025	20.11.2025	TensorFlow auf dem Raspberry Pi lauffähig machen, auf die IMX500-Kamera wechseln und Projekt/Zielgruppen schriftlich beschreiben.	TensorFlow auf RPi installieren/laufen lassen Kamera-Wechsel zu IMX500 Zielgruppenanalyse Projektbeschreibung	Done Done Done Done	Jan-Ove/Franzi/Felix Jan-Ove/Franzi/Felix Jan-Ove/Franzi/Felix Jan-Ove/Franzi/Felix
Sprint 3	21.11.2025	04.12.2025	Die TFLite-Umgebung stabilisieren, erste Objekterkennung mit Kamera anstoßen und Grundlagen für Personas sowie GUI-Konzept legen.	TensorFlow auf RPi installieren/laufen lassen Systematische Fehlerbehebung TFLite_detection_webcam.py ausführen (Objekterkennung mit Kamera starten) Kamera-Index finden Personas erstellen GUI Konzept entwickeln	Done Done Done Done Done Done	Jan-Ove/Franzi/Felix Jan-Ove/Franzi/Felix Jan-Ove/Franzi/Felix Jan-Ove/Franzi/Felix Franzi/Felix Franzi/Felix
Sprint 4	05.12.2025	19.12.2025	Die AI-Camera technisch korrekt ansprechen, eine	Kompatible Videoquelle für die AI Camera vorbereiten	Done	Jan-Ove/Franzi/Felix

			kompatible Videoquelle etablieren und das Verständnis der TFLite-Visualisierung vertiefen.	Verstehen, warum cv2.VideoCapture(0) mit libcamera-Stack scheitert	Done	Jan-Ove/Franzi/Felix
				Funktionsfähige Grundlagen-Tests etablieren	Done	Jan-Ove/Franzi/Felix
				Entwicklungsumgebung für TFLite-Objekterkennung stabilisieren	Done	Jan-Ove/Franzi/Felix
				Funktionsweise der Visualisierung und Skripte fachlich verstehen	Done	Jan-Ove/Franzi/Felix
				Didaktisches Grundkonzept für Zielgruppe SuS entwickeln	Done	Franzi/Felix
<b>Weihnachtspause 20.12.2025 - 06.01.2026</b>						
Sprint 5	07.01.2026	14.01.2026	Das Objekterkennungssystem soll von einem einfachen COCO-Proof-of-Concept zu einer skalierbaren Lösung weiterentwickelt werden, die mehr Objektklassen unterstützt und bei der klar entschieden ist, ob die Inferenz langfristig auf der CPU (TFLite) oder direkt auf der IMX500-Kamera laufen soll.	Limitierung des bisherigen Modells verstehen und Alternativmodell auswählen	Done	Jan-Ove/Franzi/Felix
				Architekturentscheidung: CPU-TFLite vs. IMX500-On-Sensor-Inferenz	Done	Jan-Ove/Franzi/Felix
				Didaktisches Grundkonzept für die Zielgruppe Studis entwickeln	Done	Franzi/Felix
				Didaktisch aufbereitete Texte für SuS schreiben	Done	Franzi/Felix
				Wechsel des Betriebssystems von Trixie zu Bookworm	Done	Jan-Ove/Franzi/Felix
				Wireframes GUI erstellen	Done	Felix
Sprint 6	15.01.2026	21.01.2026	Die KI-Kamera (IMX500) liefert stabil Inferenz-Daten, und darauf aufbauend entsteht ein erster grafischer Prototyp mit Startbildschirm und getrennter didaktischer Aufbereitung für Schüler:innen und Studierende.	Hinzufügen der Differenzierungsmöglichkeit für zweite Zielgruppe (Studis)	Done	Jan-Ove
				Anpassen der Texte (höheres Niveau, genauere Darstellung der Schritte)	Done	Felix
				Integration eines Startbildschirms	Done	Jan-Ove
				Umzug des GUIs von OpenCV zu Pygame	Done	Jan-Ove
				Texte für SuS anpassen	Done	Felix
				Texte für Studis schreiben	Done	Felix
				IMX500-Safe-Mode für didaktische Nutzung stabilisieren	Done	Jan-Ove
				Erstellung von Grafiken und Designs	Done	Franzi
				Auslagerung und Pflege des Codes	Done	Jan-Ove
				Erstellung Animationen	Done	Franzi
				Vergleich und Auswahl geeigneter 3D-Druck-Vorlagen	Done	Franzi

Sprint 7	22.01.2026	27.01.2026	Audio- und Sprachintegration sowie GUI-Feinschliff für beide Zielgruppen.	Audiospuren für SuS integrieren	Done	Jan-Ove
				GUI anpassen	Done	Jan-Ove
				Englisch als zweite Sprache integrieren	Done	Jan-Ove
				Integration von Animationen	Done	Jan-Ove
				Texte für beide Zielgruppen anpassen	Done	Franzi/Felix
				Audiospuren generieren	Done	Jan-Ove
				Anpassung des GUIs	Done	Jan-Ove
				Projektbericht schreiben	Done	Felix
				Auslagerung und Pflege des Codes	Done	Jan-Ove
				Vorbereitung und Anpassung für den 3D-Druck	Done	Franzi
Sprint 8	28.01.2026	03.02.2026	Finalisierung und Präsentationsvorbereitung.	Durchführung und Qualitätskontrolle des 3D-Drucks	Done	Franzi
				Überarbeitung des GUI	Done	Jan-Ove
				Anpassung der Animationen	Done	Franzi
				Anpassung der Texte	Done	Franzi/Felix
				Integration der Audiospuren + erneute Generierung	Done	Jan-Ove/Franzi
				Aufbereitung der dokumentierten Inhalte für die Präsentation	Done	Jan-Ove/Franzi/Felix
				Aufteilung der Präsentation	Done	Jan-Ove/Franzi/Felix
				Erstellung des Präsentationsdesigns	Done	Franzi