**Projekt Mülltrennung**

*Mülltrennung für jedermann*

[Personenliste 3](#_Toc373194628)

[Abstract 3](#_Toc219688570)

[Sections 3](#_Toc1030739823)

[Definition von Zielen 4](#_Toc1077937889)

[State of the Art 4](#_Toc882865887)

[Methodology 4](#_Toc10096755)

[Impact 4](#_Toc1823413961)

[KPI 5](#_Toc767976133)

[Workplan 6](#_Toc2035405851)

[Deliverables 6](#_Toc1212323412)

[Milestones 7](#_Toc1801840388)

[Projektteilnehmer und Aufgabenverteilung 7](#_Toc1631459127)

## Personenliste

* Marius Mattes
* André Schambach
* Sinan Licina
* Felix Lang

## Abstract

Das Projekt zielt darauf ab, eine automatisierte Mülltrennungseinrichtung zu entwickeln, die Mülltypen anhand von Sensoren oder Bildverarbeitung erkennt und in entsprechende Behälter leitet. Ein Raspberry Pi steuert den gesamten Prozess, inklusive der Mülltrennung und Benachrichtigungen bei fast vollen Behältern. Die Benutzeroberfläche für die Steuerung wird webbasiert sein, um eine einfache Bedienung zu ermöglichen.

## Sections

* **Warum das Projekt umgesetzt wird**: Die Mülltrennung in Haushalten kann durch Automatisierung vereinfacht werden. Das Projekt zielt darauf ab, Müll effizienter zu trennen und so die Recyclingrate zu verbessern.
* **Relevanz**: Steigerung der Recyclingquote, Reduzierung von Falschzuordnungen und eine saubere, benutzerfreundliche Lösung für die Haushalte.

## Definition von Zielen

* **Primärziel**: Entwicklung eines Prototyps, der Mülltypen erkennt und trennt.
* **Teilziele**:
  + Sensorische Materialerkennung
  + Optional: Funktionierende Bildverarbeitung zur Objekterkennung
  + Steuerung eines mechanischen Sortiersystems
  + Integration der Hardware und Software

## State of the Art

* **Vorhandene Systeme**: Industrielle Mülltrenner sind teuer und auf große Mengen ausgerichtet. Kleine Systeme für den Haushalt fehlen weitgehend.
* **Alternative Ansätze**: Sensoren zur Materialerkennung (z. B. Metall-Sensoren) oder einfache manuelle Trennhilfen.

## Methodology

* Materialerkennung:
  + Kameramodul oder Materialsensoren (z. B. NIR-Sensoren) am Fließband.
* Sortierung:
  + Steuerung von drei mechanischen Schienen durch Servomotoren.
* Füllstandsanzeige:
  + Ultraschall-Sensoren zur Messung des Behälterfüllstandes.
  + Anzeige von Warnungen über LEDs oder ein Display.
* Software:
  + Programmierung in Python zur Steuerung der Sensorik und Motoren, alle gesteuert durch den Raspberry-Pi

## Impact

* **Umwelt**: Verbesserung der Recyclingquote durch fehlerfreie Trennung.
* **Problemlösung**: Entlastung der Benutzer und Reduktion von Müll im Restabfall.
* **Gesellschaft**: Förderung eines nachhaltigen Lebensstils.

## KPI

* Messbare Ziele:
* Genauigkeit der Müllerkennung: Zielt auf eine Erkennungsgenauigkeit von mindestens 90%.
* Funktionsfähigkeit des Weichensystems: Es soll zuverlässig den Müll in die richtigen Behälter leiten.
* Benutzerfreundlichkeit der Weboberfläche: Mindestens 80% positive Rückmeldungen in den Tests.
* Anzahl der erfolgreichen Benachrichtigungen bei fast vollen Behältern.

## Workplan

* **Phase 1: Planung & Vorbereitung** (1 Woche)
  + Festlegung der Komponenten (Sensoren, Raspberry Pi, Motoren)
  + Definition der Anforderungen für die Weboberfläche
* **Phase 2: Hardware-Implementierung** (2 Wochen)
  + Aufbau des Laufbands und Integration der Sensoren
  + Entwicklung des Weichensystems zur Mülltrennung
* **Phase 3: Software-Entwicklung** (3 Wochen)
  + Implementierung der Müllerkennungssoftware (Bildverarbeitung oder Sensorintegration)
  + Entwicklung der Weboberfläche
  + Integration der Benachrichtigungsfunktion bei vollen Behältern
* **Phase 4: Testphase** (1 Woche)
  + Testen der Hardware- und Softwarekomponenten
  + Fehlerbehebung und Optimierung
* **Phase 5: Abschluss & Dokumentation** (1 Woche)
  + Erstellung der Dokumentation
  + Abschlusspräsentation

## Deliverables

* **Phase 1**: Anforderungsdokument und Hardware-Spezifikationen
* **Phase 2**: Prototyp der Hardware
* **Phase 3**: Funktionsfähige Software, Weboberfläche und Benachrichtigungsfunktionen
* **Phase 4**: Testberichte und Optimierungen
* **Phase 5**: Endbericht und Präsentation

## Milestones

* **Milestone 1**: Fertigstellung der Hardware-Installation
* **Milestone 2**: Abschluss der Softwareentwicklung
* **Milestone 3**: Erfolgreicher Abschluss der Testphase
* **Milestone 4**: Präsentation der fertigen Lösung

## Projektteilnehmer und Aufgabenverteilung

1. Elektronik & Hardware-Entwicklung
   1. Kompetenzen: Erfahrung im Umgang mit Mikrocontrollern (z. B. Raspberry Pi), Motorsteuerungen, Sensoren (z. B. Infrarot, Ultraschall, Kameras), und mechanischen Komponenten für die Mülltrennung (z. B. Weichensysteme).
   2. Verantwortung: Aufbau und Inbetriebnahme der Hardware, Integration der Sensoren und Aktuatoren (Motoren, Servos).
2. Softwareengineering (Bildverarbeitung und Sensorintegration)
   1. Kompetenzen: Programmierung von Bildverarbeitungssystemen, Softwareentwicklung auf dem Raspberry Pi, sowie die Ansteuerung von Sensoren und Motoren.
   2. Verantwortung: Implementierung der Müllerkennungsalgorithmen (Bildverarbeitung oder Sensorintegration), Entwicklung der Steuerungssoftware für das Weichensystem, Benachrichtigungslogik.
3. Webentwicklung
   1. Kompetenzen: Erfahrung in der Erstellung von Weboberflächen (z. B. mit HTML, CSS, JavaScript, Django), Backend-Entwicklung zur Integration der Mülltrennsystemsteuerung in eine benutzerfreundliche Oberfläche.
   2. Verantwortung: Entwicklung und Design der Weboberfläche für die Steuerung des Systems sowie der Benachrichtigungen.
4. Projektmanagement
   1. Kompetenzen: Planung, Organisation und Koordination der einzelnen Projektphasen, insbesondere die Koordination zwischen Hardware, Software und Testteams.
   2. Verantwortung: Sicherstellen, dass das Projekt rechtzeitig und im Rahmen des Budgets abgeschlossen wird, Überwachung des Fortschritts und der Qualität.
5. Benutzerfreundlichkeit und Design
   1. Kompetenzen: Design von benutzerfreundlichen Interfaces, z. B. für das Web-Interface. Erfahrung im UX/UI-Design.
   2. Verantwortung: Sicherstellen, dass die Benutzeroberfläche intuitiv und einfach zu bedienen ist.

