

**Pflichtenheft**

**HTW Berlin**

**ZWL-Roboter**

Autor: Gruppe ZWL

Letzte Änderung: 24. Nov 2022

Dateiname: ZWL-Roboter\_Pflichtenheft.docx

Version: 1.0

***Copyright***

© ZWL-Roboter Gruppe

Die Weitergabe, Vervielfältigung oder anderweitige Nutzung dieses Dokumentes oder Teile davon ist unabhängig vom Zweck oder in welcher Form untersagt, es sei denn, die Rechteinhaber/In hat ihre ausdrückliche schriftliche Genehmigung erteilt.

***Version Historie***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Version | | Datum | Verantwortlich | Änderung |
| 0.1 | 28.10.2022 | | Markus | Überblick + Hauptziele |
| 0.2 | 29.10.2022 | | Elian | Workflow |
| 0.3 | 01.11.2022 | | Maged | Annahmen |
| 0.4 | 03.11.2022 | | Markus | Überblick Funktionalität |
| 0.5 | 05.11.2022 | | Maged | Abgrenzungen |
| 0.6 | 07.11.2022 | | Elian | Funktionalität |
| 0.7 | 10.11.2022 | | Rayen | Erweiterungen Funktionalität |
| 0.8 | 15.11.2022 | | Markus | Erweiterungen Überblick |
| 0.9 | 24.11.2022 | | Alle | Allgemeine Erweiterungen |
| 1.0 | 24.11.2022 | | Elian | Abgabe |

**Inhaltsverzeichnis**

[Verzeichnis vorhandener Dokumente II](#_Toc102464365)

[1 Überblick 1](#_Toc102464366)

[2 Hauptziele 2](#_Toc102464367)

[3 Annahmen und Abgrenzungen 2](#_Toc102464368)

[4 Workflow 3](#_Toc102464369)

[5 Funktionalität 4](#_Toc102464370)

[5.1 Überblick 4](#_Toc102464371)

[5.2 Zauberwürfel Seiten fotografieren …… **Fehler! Textmarke nicht definiert.**](#_Toc102464372)

[5.3 Lösungsstrategie berechnen …… 7](#_Toc102464373)

[5.4 Zauberwürfel lösen …… 8](#_Toc102464373)

[6 Offene Fragen 9](#_Toc102464374)

[7 Modulabhängigkeiten 9](#_Toc102464375)

[8 Wer hat was gemacht 9](#_Toc102464376)

#### 

#### Abbildungsverzeichnis

Abbildung 1: Systemmodell ..............................................................................................................................1

Abbildung 2: Workflow……................................................................................................................................3

Abbildung 3: Möglicher aufbau..........................................................................................................................4

Abbildung 4: Schaltplanung…...........................................................................................................................5

Abbildung 5: Use Case Diagra...........................................................................................................................5

Abbildung 6: Mögliche Würfelseiten...................................................................................................................6

**Tabellenverzeichnis**

Tabelle 1: Hauptziele……………………………………………………………………….………………...…………2

Tabelle 2: Annahmen……………………………………………………………………………………………………2

Tabelle 3: Abgrenzungen……………………………………………………………………………….………………2

Tabelle 4: Zauberwürfel Seiten fotografieren…………………………………………………………………………6

Tabelle 5: Lösungsstrategie berechnen……………………………………………………………….………………7

Tabelle 6: Zauberwürfel lösen………………………………………………………………………….………………8

Tabelle 7: offene Frage………………………………………………...……………………………….………………9

Tabelle 8: wer hat was gemacht……………………………………………………………………….………………9

#### Verzeichnis vorhandener Dokumente

Alle für die vorliegende Spezifikation ergänzenden Unterlagen müssen hier aufgeführt werden

|  |  |  |
| --- | --- | --- |
| **Dokument** | **Autor** | **Datum** |
| ZWL-Roboter\_Workflow.graphml | Elian | 17.11.2022 |
| ZWL-Roboter\_USD.webp | Rayen | 20.11.2022 |
| ZWL-Roboter\_Plan(1).mpp | Maged | 22.11.2022 |

# Überblick

Im Rahmen des vorliegenden Projekts wird ein System aufgebaut, das automatisiert einen beliebig verdrehten Zauberwürfel in die Ausgangsposition zurückdreht. Entsprechend dem Lastenheft soll das zu entwickelnde System folgende Merkmale verfügen:

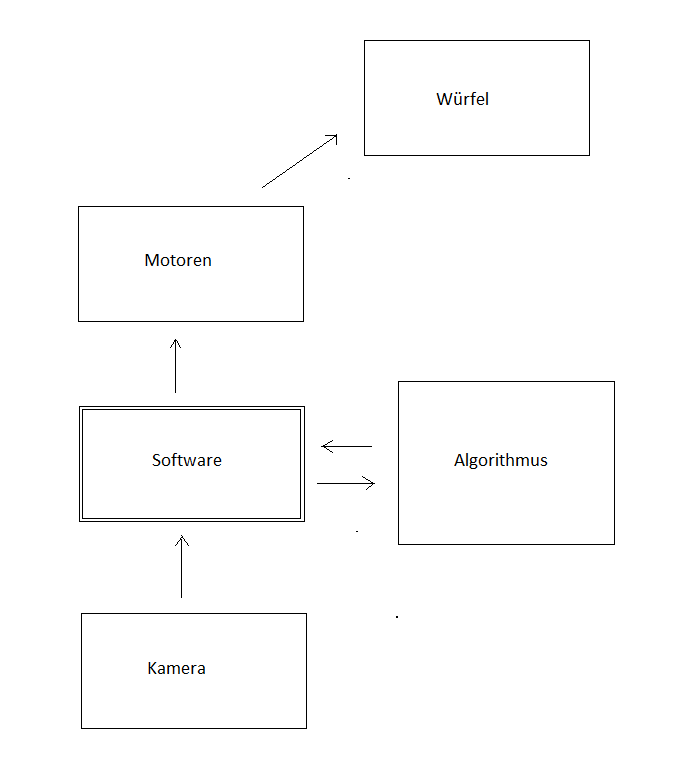
Die Ausgangskonfiguration des Würfels wird durch eine Kamera detektiert. Durch die Kamera wird stets die aktuelle Position mit Hilfe der Farben ermittelt, um den Lösungsweg anschließend berechnen zu können.

Die Kamera befindet sich über dem Würfel in einer Halterung. Die Kamerahalterung wird fest an der Bodenplatte verschraubt.

Der Lösungsweg wird durch einen ausgewählten Algorithmus ermittelt.

Das System verfügt über eine Drehmechanik mit insgesamt 3 beweglichen Achsen. Die einzelnen Achsen werden jeweils von einem Schrittmotor angetrieben. Die Motorsteuerung erfolgt über Software.

Der aktuelle Status bzw. eventuell entstandene Probleme werden über Ton bzw. über eine Anzeige (LED-Indikator) dem Benutzer mitgeteilt. Für die Steuerung des Roboters sind zwei Buttons vorgesehen (Start/Aus & Not-Aus).



**Abbildung 1: Systemmodell**

# Hauptziele

|  |  |  |
| --- | --- | --- |
| # | Ziel | Beschreibung der Implementation |
| 1 | Lösen des Würfels. | Es soll mit der Wahrscheinlichkeit von 95% den Würfel lösen können. |
| 2 | Zeit | Löst den Würfel in maximal 5min. |
| 3 | Status | Zeigt den aktuellen Lösungsvorschritt |
| 4 | Fehler Reduzierung. | Fehler, welche durch Hardware entstehen sollen, vorgebeugt werden und nicht häufiger als 10% vorkommen. |
| 5 | Vereinfachung des Systems. | Kein Vorwissen / Einarbeitung notwendig. |

**Tabelle 1: Hauptziele**

# Annahmen und Abgrenzungen

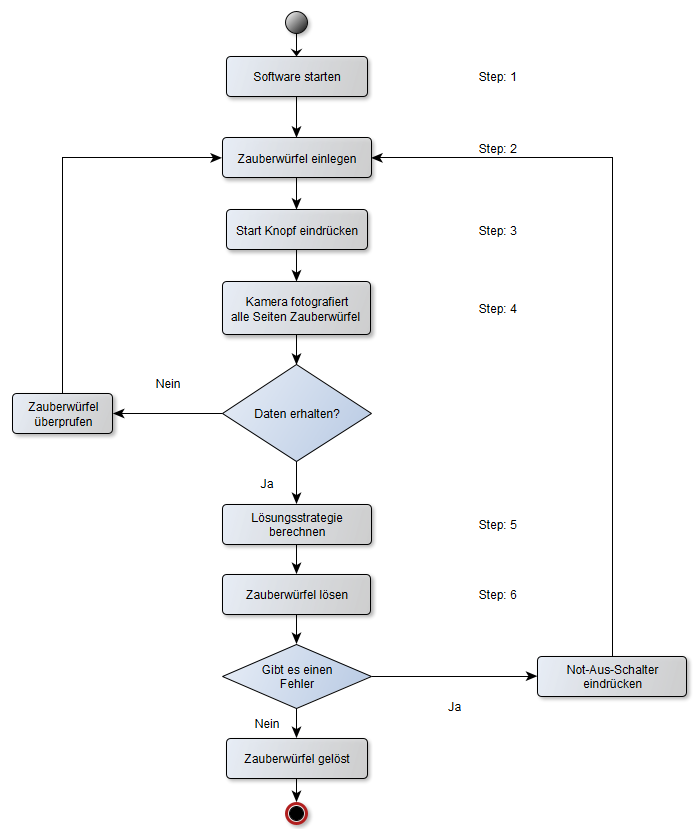
|  |  |
| --- | --- |
| # | Annahmen (fachliche und technische Annahmen) |
| 1 | Arduino (Microkontroller) |
| 2 | Laptop |
| 3 | 3d Teile |
| 4 | Schrittmotoren |
| 5 | Kamera |
| 6 | DC-Stromversorgung 12V |

**Tabelle 2: Annahmen**

|  |  |
| --- | --- |
| # | Abgrenzungen(Was ist in dieser Lösung nicht enthalten bzw. abgedeckt) |
| 1 | Sprachsteuerung |
| 2 | Handy-Applikation |
| 3 | Drahtlose Kommunikation |
| 4 | Steuerung per Internet |

**Tabelle 3: Abgrenzungen**

# Workflow



**Abbildung 2: Workflow**

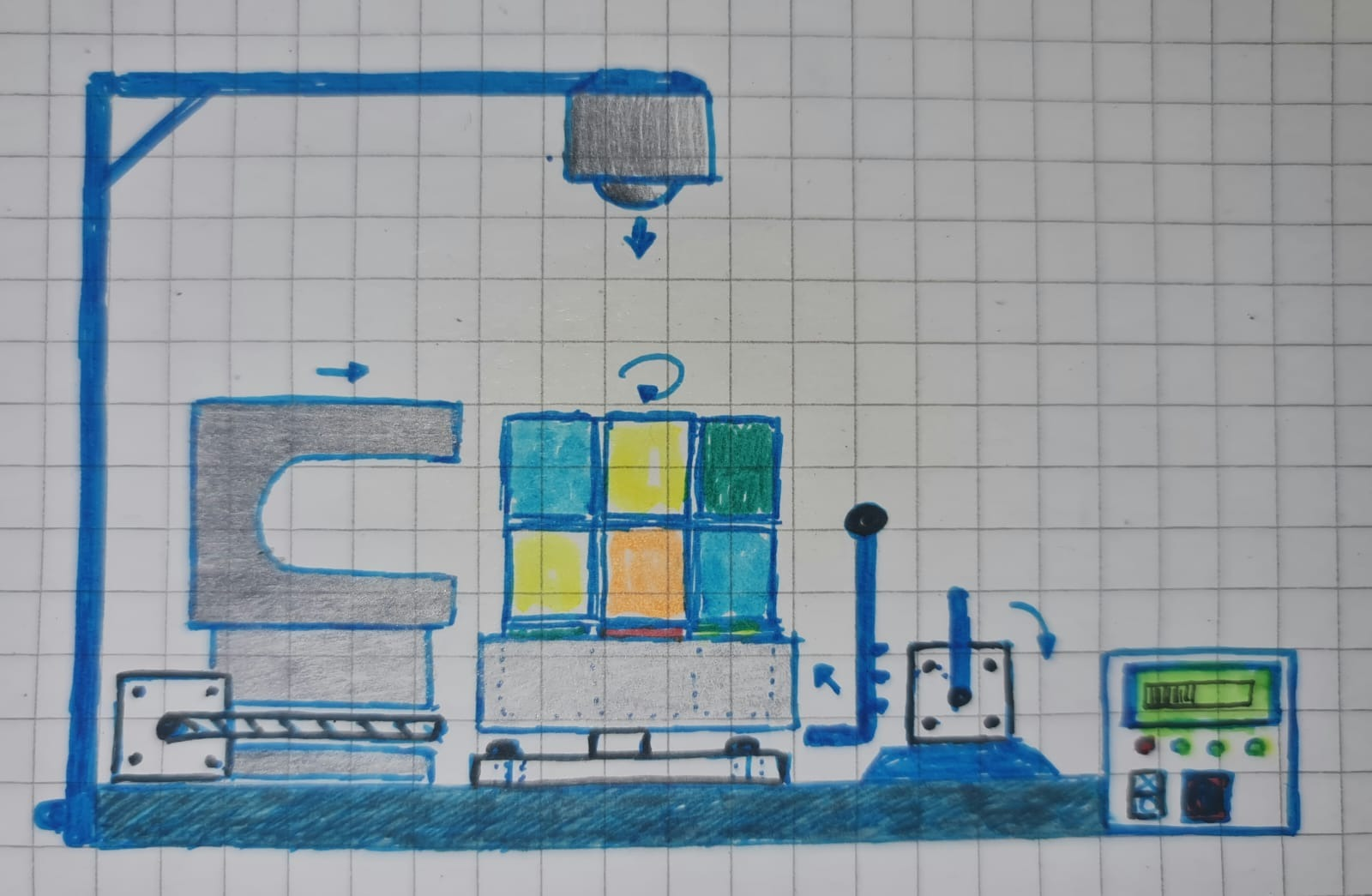
# Funktionalität

## Überblick

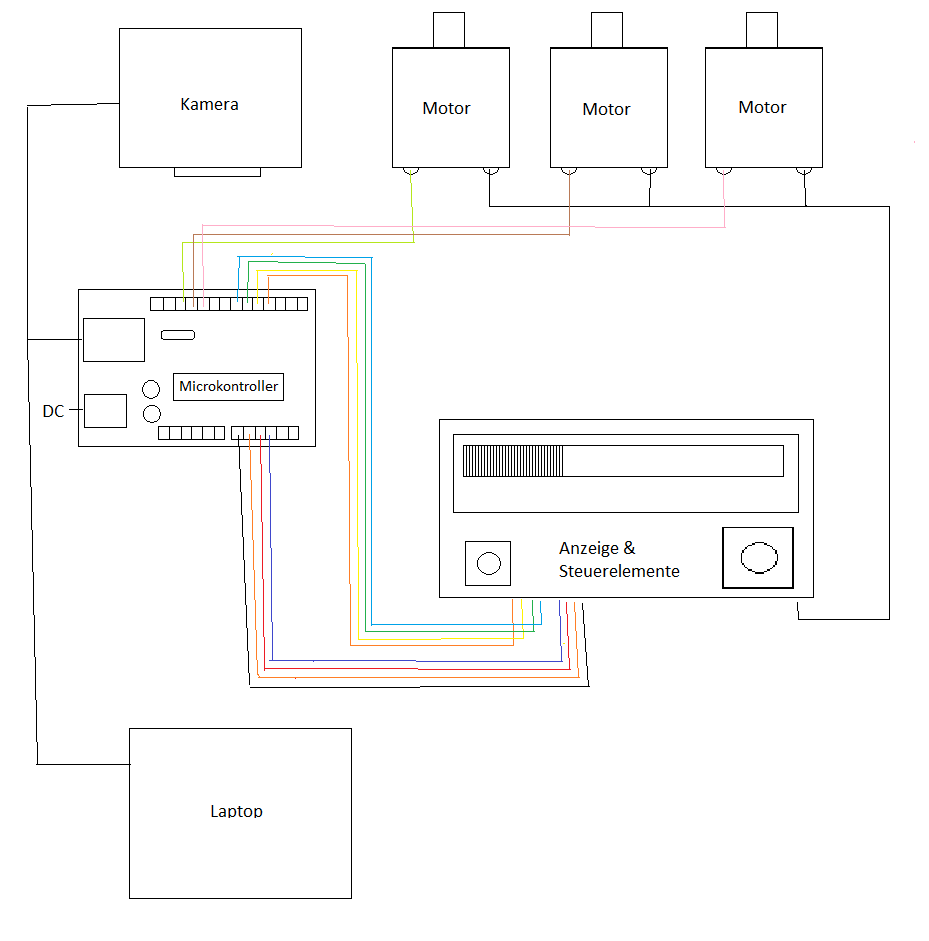
Der Roboter muss an eine konstante Stromquelle (12 V) angeschlossen werden. Würfel in den vorgesehenen Platz einlegen und den Start-Knopf betätigen. Der Roboter erkennt, ob ein Würfel eingelegt ist oder nicht und fängt erstmal mit dem Fotografieren (Scanner) der einzelnen Seiten an.

Intern wird eine Karte der Seitenteile inkl. Der einzelnen Farben gebildet. Der Algorithmus berechnet danach die notwendigen Lösungsschritte. Die Motoren werden entsprechend der Lösungsschema angesteuert. Sollte etwas nicht funktionieren oder der Würfel rausfallen stoppt er den Vorgang und meldet es dem Benutzer.

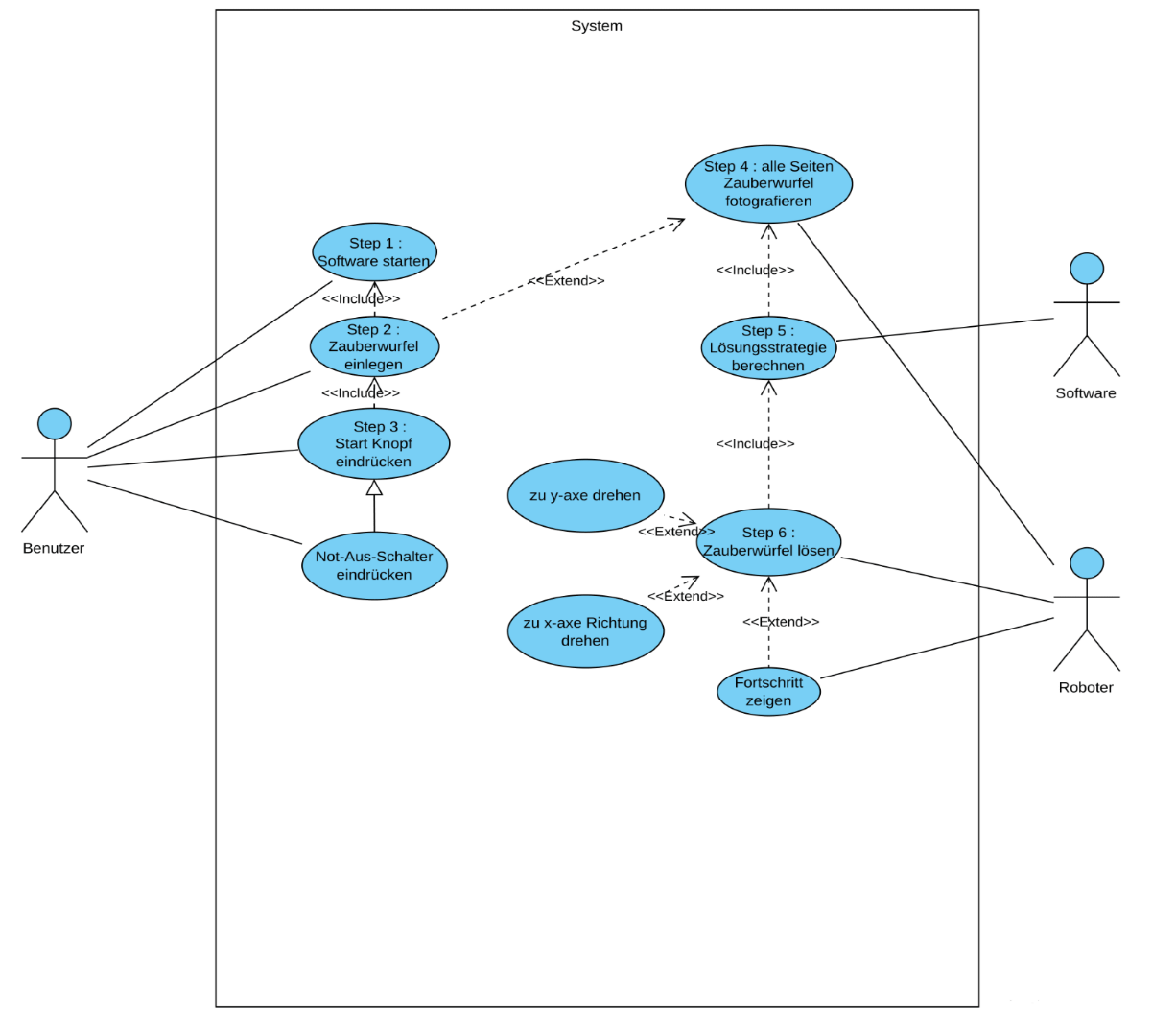
Der Vorschritt, also der ungefährer Prozentstatus, wird auf dem Display angezeigt. Dadurch weiß man ungefähr, wann er fertig ist.



**Abbildung 3: Möglicher Aufbau**



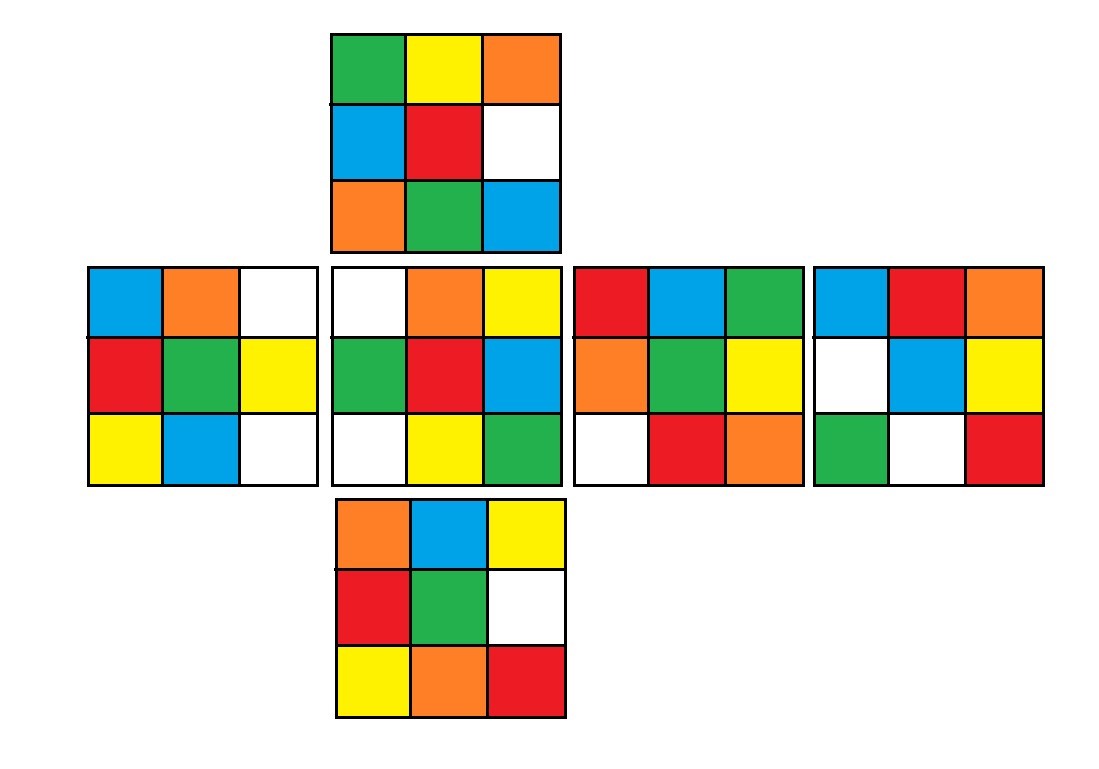
**Abbildung 4: Schaltplanung**

  
  
**Abbildung 5: Use Case Diagramm**

## Zauberwürfel Seiten fotografieren:

|  |  |
| --- | --- |
| Zweck/Ziel | Der Roboter fotografiert alle Seiten des Zauberwürfels mit einer Kamera, um diese dann inder Software auszuwerfen |
| Akteur/Auslöser | Benutzer, Start-Button |
| Berechtigung | Jeder ist in der Lage das System zu bedienen |
| WF-Rererenz | Kamera fotografiert alle Seiten Zauberwürfel Step: 4 |
| Vorbedingung | * Zauberwürfel muss im Position sein. * Zauberwürfel muss vom Typ 3x3x3 sein. * Zauberwürfel soll vorgegebene Größe haben. |
| Verarbeitungs-schritte | Um ein Foto von allen Seiten des Zauberwürfels zu machen:    Fotos von den Seiten des Zauberwürfels werden mit einer Kamera aufgenommen, die direkt über dem Würfel positioniert wird.  Wenn das Foto aufgenommen  wird, dreht der Roboter den Würfel, um die nächste Seite aufzunehmen. |
| Ergebnis | 6 Fotos aufgenommen und jedes Foto zeigt eine Seite des Zauberwürfels |
| Plausibilitäten | Es müssen 6 verschiedene Dateien vorhanden sein. |
| Fehlerhandling | Der gesamte Prozess wird vom Benutzer überwacht.    Falls der Würfel seine Position nicht erfolgreich ändert, wird der Benutzer den Würfel richtig positionieren.  Kamera nicht korrekt angeschlossen: es gibt eine Fehlermeldung |
| Folgeprozess | Step 5: Lösungsstrategie berechnen |
| Anforderung | Bilder der Einzelseiten müssen vorhanden sein  Zauberwürfel muss richtig eingestellt werden  Start-Button muss gedruckt werden |
| Test Cases | Start-button drucken, ohne den Zauberwürfel einzustellen  Kamera nicht angeschlossen. |

**Tabelle 4: Zauberwürfel Seiten fotografieren**

**

**Abbildung 6: Mögliche Würfelseiten Fotografie der Kamera**

## Lösungsstrategie berechnen

|  |  |
| --- | --- |
| Zweck/Ziel | Die Lösungsstrategie für den Zauberwürfel berechnen. |
| Akteur/Auslöser | Roboter, System |
| Berechtigung | Daten von dem System (Kamera) erhalten. |
| WF-Rererenz | Lösungsstrategie berechnen Step 5. |
| Vorbedingung | * Würfel befindet sich in seiner Position. * Kamera fotografiert den Zauberwürfel. * Fotos als PNG speichern. |
| Daten-Input | Daten von fotografierte Zauberwürfel (Step: 4)  Die Daten sollen passende zu Algorithmus sein (Datenformat) |
| Verarbeitungs-schritte | Um ein Lösungsstrategie des Zauberwürfels zu berechnen:   * Fotos werden als PNG gespeichert (z.B). * Die Zusammensetzung des Würfels wird anhand der Farben Ermitteln, die in jedem Foto mit Hilfe von Bildverarbeitungsalgorithmen vorhanden ermittelt. * Dann basierend auf den Ergebnissen und einem spezialisierten Algorithmus, um einen Zauberwürfel zu lösen, werden wir eine Lösungsstrategie berechnen. |
| Ergebnis | Eine Reihe von Befehlen, um den Zauberwürfel entweder in y-Richtung oder in x-Richtung zu drehen |
| Plausibilitäten | Es müssen Reihe von verschiedenen Befehlen vorhanden sein |
| Fehlerhandling | Was passiert im Fehlerfall? z.B.   * Falscher Würfel eingelegt, dann gibt es eine Fehlermeldung. Daten unvollständig, dann wird es erneut Fotografiert bzw. es gibt eine Fehlermeldung. |
| Folgeprozess | Der Roboter fängt den Zauberwürfel zu lösen(Step: 6) |
| Test Cases | Einen gelösten Würfel einlegen.  Würfel hat auf jeder Seite gleiche Farben. |

**Tabelle 5: Lösungsstrategie berechnen**

## Zauberwürfel lösen

|  |  |
| --- | --- |
| Zweck/Ziel | Der Roboter löst den Zauberwürfel |
| Akteur/Auslöser | Roboter |
| Berechtigung | Software, Roboter |
| WF-Rererenz | Zauberwürfel lösen Step 6 |
| Vorbedingung | * Lösungsstrategie berechnen. * Die Lösungsstrategie für der Zauberwürfel erhalten. |
| Daten-Input | * Die berechnete Lösungsstrategie. * Lösungsschema für Motor Ansteuerung. |
| Verarbeitungs-schritte | Beschreibung der Verarbeitungslogik gemäß Workflow Definition.     * Entsprechend der Lösung steuere die Schrittmotoren. * Steuerung erfolgt nach dem vorgegebenen Schema. * Entsteht ein Fehler wird es dem User gemeldet und der Vorgang gestoppt. |
| Ergebnis | Ein gelöster Zauberwürfel |
| Plausibilitäten | Ein erneutes Fotografieren kann das erfolgreiche Lösen des Würfels bestätigen bzw. Korrekturen starten. |
| Fehlerhandling | Was passiert im Fehlerfall? z.B.   * Lösen nicht erfolgreich: Meldung an dem Benutzer und stoppen. Mechanischer Fehler: der Benutzer greift ein und stoppt den Vorgang. Gelöster Würfel ist fehlerhaft: ein erneutes Starten des Vorgangs erforderlich. |
| Folgeprozess | Ende |
| Test Cases | Einen gelösten Würfel einlegen.  Würfel hat auf jeder Seite gleiche Farben. |

**Tabelle 6: Zauberwürfel lösen**

# 6 Offene Fragen

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **#** | **Issue** | **Status** | **Owner** | **Deadline** |
| 1 | Keine Offene Fragen an Auftraggeber. |  |  |  |

**Tabelle 7: Offene Fragen**

# 7 Modulabhängigkeiten

Das Projekt beinhaltet Kenntnisse, die aus den folgenden Modulen erworbenen wurden: Konstruktion, Software-Entwicklungsprojekt, Programmierung, Mechatronik und Projektmanagement. Das sind die vorausgesetzten Module, um das Projekt erfolgreich abzuschließen.

**8 Wer hat was gemacht**

|  |  |  |
| --- | --- | --- |
| Autor | Aufgabe/Kapitel | Anteil |
| Elian | Workflow | 100% |
| Markus | Überblick | 100% |
| maged | Annahmen und Abgrenzungen | 100% |
| Rayen | Funktionalität | 65% |
| Elian | Funktionalität | 35% |
|  |  |  |
|  |  |  |

**Tabelle 8: wer hat was gemacht**