

BACHELORTHESIS
Bastian Krammer

Konstruktion eines Artificial Pets in einer Smart Home Umgebung

FAKULTÄT TECHNIK UND INFORMATIK
Department Informatik

Faculty of Computer Science and Engineering
Department Computer Science

Bastian Krammer

Konstruktion eines Artificial Pets in einer Smart Home Umgebung

Bachelorarbeit eingereicht im Rahmen der Bachelorprüfung
im Studiengang Bachelor of Science Technische Informatik
am Department Informatik
der Fakultät Technik und Informatik
der Hochschule für Angewandte Wissenschaften Hamburg

Betreuender Prüfer: Prof. Dr. Thomas Lehmann
Zweitgutachter: Prof. Dr. Kai von Luck

Eingereicht am: 16. Juni 2021

Bastian Krammer

Thema der Arbeit

Konstruktion eines Artificial Pets in einer Smart Home Umgebung

Stichworte

Agenten, künstliche Tiere, soziale Robotik, smarte Objekte, smarte Wohnung

Kurzzusammenfassung

Diese Thesis befasst sich mit einer Pflanze als künstliches Tier, die in einer smarten Umgebung ein sozialeres und aktiveres Umfeld schaffen soll. Somit muss die Einordnung der Pflanze in die Bereiche der sozialen Robotik, der künstlichen Tiere und von smarten Objekten stattfinden. Darauf aufbauend wird die Pflanze technisch modifiziert, dass sie die Möglichkeiten hat sich auszudrücken, ihre eigenen Bedürfnisse wahrzunehmen und sich selbständig innerhalb der Umgebung zu bewegen. Mithilfe dieses technischen Aufbaus kann sie selbständig ihre Bedürfnisse befriedigen und Emotionen in Form von Körperbewegung, Töne und Farben ausdrücken.

Bastian Krammer

Title of Thesis

Construction of an artificial pet in a smart home environment

Keywords

Agents, artificial pets, social robots, smart objects, smart home

Abstract

This thesis deals with a plant as an artificial animal that is intended to create a more social and active environment in a smart environment. Thus, the classification of the plant must take place in the areas of social robotics, artificial animals and smart objects. Based on this, the plant will be technically modified so that it has the ability to express itself, perceive its own needs and move independently within the environment. With the help of this technical construction, it can independently satisfy its needs and express emotions in the form of body movement, sounds and colors.

Inhaltsverzeichnis

Abbildungsverzeichnis	vii
Tabellenverzeichnis	viii
1 Einleitung	1
1.1 Motivation	1
1.2 Ziele	2
1.3 Aufbau der Arbeit	2
2 Analyse	4
2.1 Agenten	4
2.1.1 Umgebung	5
2.2 Soziale Robotik	6
2.3 Künstliche Tiere	7
2.4 Smarte Objekte	9
2.5 Ausdrücken von Affekten	9
2.6 Zusammenfassung der Analyse	13
3 Umsetzung	14
3.1 Konzept	14
3.1.1 Umgebung	14
3.1.2 Einordnung Plantbot	16
3.1.3 Ziele und Verhalten des Plantbots	16
3.2 Physischer Aufbau	20
3.2.1 Roboter Plattform	21
3.2.2 Sensorik und Aktorik	22
3.3 Software Design	25
3.3.1 Navigator	27
3.3.2 Emotion Engine	32

3.3.3	ROS-Pakete	33
3.3.4	Externe ROS-Pakete	39
3.3.5	Kommunikation	39
3.4	Evaluation	43
3.4.1	Physischer Aufbau	43
3.4.2	Verhalten und Ziele des Plantbots	44
4	Fazit und Ausblick	46
4.1	Fazit	46
4.2	Ausblick	48
	Literaturverzeichnis	50
A	Anhang	54
A.0.1	Schaltplan Lochrasterplatine	54
	Selbstständigkeitserklärung	55

Abbildungsverzeichnis

2.1	Die Baby Robbe Paro (Quelle: [1])	8
2.2	Proxemik Zonen	10
2.3	Links ist das <i>Lauflicht</i> und rechts das <i>geteilte Lauflicht</i> Farbmuster zu sehen.	12
3.1	Turtlebot3 Waffle Pi	21
3.2	Lochrasterplatine für den Arduino und der zusätzlichen Sensorik und Aktorik	23
3.3	Die verwendete Sensorik und Aktorik mit dessen Verarbeitungseinheiten.	25
3.4	Der Plantbot Roboter.	26
3.5	Informationsfluss zwischen den ROS-Komponenten des Plantbots und der Umwelt.	28
3.6	Subsumption-Architektur für die Navigator-Komponente. Die Priorisierung der Aktionen ist von unten nach oben gestaffelt.	30
3.7	FSM für die Implementierung der Emotionen FSM	34
3.8	Karte der Living Place Wohnung aufgenommen mit dem LDS des Turtlebot Aufbaus.	36
3.9	ROS-nodes in den unterteilten Pakete	38
3.10	Die Komponenten mit dessen Nachrichtenaustausch.	42
3.11	Nachrichtenaustausch über den I ² C-Bus.	43
A.1	Schaltplan Lochrasterplatine	54

Tabellenverzeichnis

3.1	Ziel 01 des Plantbots	17
3.2	Ziel 02 des Plantbots	18
3.3	Ziel 03 des Plantbots	18
3.4	Ziel 04 des Plantbots	18
3.5	Ziel 05 des Plantbots	18
3.6	Ziel 06 des Plantbots	18
3.7	Ziel 07 des Plantbots	19
3.8	Ziel 08 des Plantbots	19
3.9	Sensor Werte für die Übergänge der Emotionen	32
3.10	Die verwendeten Topics mit den zugehörigen messages	40

1 Einleitung

Das Einleitungskapitel beschreibt die Motivation 1.1 hinter dieser Thesis und was die Ziele 1.2 davon sind. Des Weiteren wird darauf eingegangen, wie die Arbeit aufgebaut ist und welche Inhalte wo zu finden sind 1.3.

1.1 Motivation

Die psychische Gesundheit von Menschen ist stark von Interaktionen im sozialen Umfeld abhängig [3]. In vielen Bereichen der heutigen Gesellschaft ist es allerdings schwierig, diese Interaktionen für bedürftige Personen ausreichend aufrechtzuerhalten. So fehlt es in Bereichen wie der Altenpflege oder Institutionen für behinderte Kinder oft an Personal, um ein ausgiebiges Programm für soziale Interaktionen anbieten zu können [16]. Auch Angehörige können durch ihr privates Leben nicht immer die notwendige Zeit aufbringen, um das auszugleichen. Ebenfalls, wenn Personen alleine in einem Haushalt leben, insbesondere in Zeiten wie einer Pandemie, kann es zu einem Mangel an Interaktionen führen. In solchen Bereichen können Roboter für ein sozialeres Umfeld sorgen und kleinere Interaktionen mit den Personen durchführen.

Der Begriff Roboter wird durch die Filmindustrie oft mit intelligenten, menschenähnlichen Sci-Fi Maschinen assoziiert, doch in der heutigen Zeit werden Roboter meist in der Industrie genutzt, um bestimmte physische Aufgaben, mit unterschiedlicher Komplexität, zu erledigen [19]. Diese Roboter benötigen Computer, um mit anderen Robotern kommunizieren zu können, ihre Umwelt über Sensorik wahrnehmen und verarbeiten zu können und letztendlich ihre Aufgaben bewältigen zu können. Da diese Computer früher sehr viel Platz in Anspruch genommen haben, mussten die dazugehörigen Maschinen dementsprechend groß sein. Doch durch die fortschreitende Technik ist es heutzutage möglich, die zugehörigen Hardware-Bausteine wesentlich kleiner und trotzdem leistungsstärker und energieeffizienter zu bauen. Des Weiteren wird die Hardware durch die Massenproduktionen preislich für die Breite Masse zugänglicher [12].

Durch die kleiner und billiger werdende Technik ergibt sich somit die Möglichkeit, alltägliche Gegenstände mit der notwendigen Rechenleistung, Sensorik und Aktorik auszustatten, um sie als Roboter agieren zu lassen. Diese Roboter können somit dafür sorgen, für Menschen ein aktiveres und sozialeres Umfeld zu gestalten.

1.2 Ziele

Ziel dieser Thesis ist die Ausstattung eines alltäglichen Gegenstandes mit der notwendigen Technik, dass er als Roboter seine Bedürfnisse selbständig stillen kann und ein aktiveres und sozialeres Umfeld schafft. Konkret wird für diese Thesis eine Zimmerpflanze dafür verwendet. Diese wird umfunktioniert, dass sie als eigenständiges Individuum in einer Wohnung lebt und sich selbständig darin versorgen kann. Somit soll sich die Pflanze im Rahmen einer Smart Home Umgebung frei bewegen können und sie soll in der Lage sein, mit der Wohnung und mit Personen innerhalb dieser Wohnung interagieren zu können. In dieser Thesis wird dafür die sogenannte „Living Place“ [2] Wohnung der HAW Hamburg als Probestfläche genutzt. Außerdem soll die Pflanze als eigenständiges Individuum persönliche Eigenschaften wie Emotionen aufweisen und ausdrücken können.

1.3 Aufbau der Arbeit

In Kapitel 2 wird analysiert, in welchen Bereichen die Roboter Pflanze in der Wissenschaft einzuordnen ist und welche Gebiete dafür besonders relevant sind. Dazu werden die Gebiete der sozialen Robotik 2.2, der künstlichen Tiere 2.3 und von smarten Objekten 2.4 genauer betrachtet. Daneben wird auf die Ausdrucksmöglichkeiten von eingeschränkten Robotern eingegangen 2.5. Zuletzt werden die wichtigsten Punkte des Analysekapitels zusammengefasst dargestellt 2.6.

Das Umsetzungskapitel 3 beinhaltet den kompletten Aufbau des Pflanzenroboters. Dafür wird ein Konzept erarbeitet 3.1, dass die konkreten Ziele des Aufbaus aufgrund des Analysekapitels beschreibt und daraufhin wird der physische Aufbau 3.2, der Software Entwurf 3.3 und die technische Kommunikation 3.3.5 des Roboters erläutert. Diese Umsetzung wird abschließend für das Kapitel evaluiert 3.4.

Zum Schluss wird in dem Kapitel 4 das gewonnene Fazit 4.1 und der Zukunftsausblick 4.2 für diese Thesis aufgeführt.

2 Analyse

Um eine Pflanze als sozialen Mitbewohner agieren zu lassen, müssen zunächst mehrere Aspekte genauer betrachtet werden. Darunter fällt die Betrachtung, warum die Agenten Metapher in diesem Kontext hilfreich ist und was ihre wichtigsten Merkmale sind. Mithilfe von diesem Wissen werden anschließend die grundlegenden Eigenschaften von sozialen Robotern, künstlichen Tieren und smarten Objekten betrachtet. Diese Themen werden benötigt, um die Pflanze wissenschaftlich einordnen zu können und um zu verstehen, welche sozialen Aspekte die Pflanze in einer sozialen Umgebung benötigt und wie sie sich darin Verhalten soll. Außerdem werden wegen des eingeschränkten Erscheinungsbildes der Pflanze die möglichen Ausdrucksarten der Roboterpflanze behandelt, um die Akzeptanz bei Menschen möglichst zu begünstigen.

2.1 Agenten

In der Wissenschaft und insbesondere der Informatik werden eigenständige Einheiten als Agenten bezeichnet. Der Begriff Agent hat keine allgemeingültige Definition, denn in unterschiedlichen Bereichen der Wissenschaft wird auf andere Eigenschaften eines Agenten Wert gelegt. In der Artificial Intelligence (AI) ist es zum Beispiel essenziell, dass ein Agent aus seinen Erfahrungen lernt, in anderen Bereichen ist dies unerwünscht [30]. In der zugehörigen Fachliteratur ist die Grundlage eines Agenten ein Computersystem, das sich in einer physischen oder virtuellen Umgebung befindet und durch autonome Aktionen in dieser Umgebung seinen Zielen nachgehen kann [30, 11, 14]. Dabei spielt die Umgebung, in der sich ein Agent befindet, eine zentrale Rolle. Denn dieser muss in der Lage sein, seine Umgebung zu erkennen, sich in ihr zu bewegen und mit ihr zu interagieren, um seine eigenen Ziele autonom erreichen zu können.

2.1.1 Umgebung

Agenten haben meist eine partielle Kontrolle über ihre Umgebung. Das bedeutet, sie können ihre Umgebung beeinflussen und sie können von der Umgebung beeinflusst werden. Welche Aktion von einem Agenten in einer bestimmten Situation ausgeführt wird, um sein Ziel zu erreichen, hängt somit davon ab, in welchem Zustand sich die Umgebung gerade befindet. Wie komplex dieser Entscheidungsprozess ist, wird durch die Eigenschaften einer Umgebung bestimmt. Russell schlägt für die Klassifizierung einer Umgebung die folgenden Umgebungseigenschaften vor [25]:

- **Zugänglichkeit vs. Unzugänglichkeit**
Eine Umgebung ist zugänglich, wenn es einem Agenten möglich ist, alle Informationen über diese Umgebung erhalten zu können. Die meisten Umgebungen in der realen Welt entsprechen nicht dieser Definition und werden somit als unzugänglich eingestuft.
- **Deterministisch vs. nicht-deterministisch**
In einer deterministischen Umgebung hat jede Aktion, die darin ausgeführt wird, genau ein daraus resultierendes Ergebnis. Das bedeutet, dass Aktionen eines Agenten immer gleich ablaufen und somit immer erfolgreich abgeschlossen werden. Aber auch hier gilt, dass reale Umgebungen, aus einer Agentenperspektive, nie als deterministisch betrachtet werden können.
- **Episodisch vs. nicht Episodisch**
In einer episodischen Umgebung ist die Erfahrung des Agenten in „Episoden“ unterteilt. Jede Episode besteht daraus, dass der Agent wahrnimmt und dann handelt. Die Qualität seiner Handlung hängt nur von der Episode selbst ab, da nachfolgende Episoden nicht davon abhängen, welche Handlungen in vorherigen Episoden stattfinden.
- **Statisch vs. dynamisch**
Eine statische Umgebung kann als solche betrachtet werden, wenn sie nur durch die Aktionen von einem einzigen Agenten beeinträchtigt und verändert werden kann. Im Gegenzug dazu, befinden sich in einer dynamischen Umgebung mehrere Akteure, die die Umgebung parallel verändern können. Dies wird in der Fachliteratur als Multi-Agenten System bezeichnet [14, 30]. Sowohl in der realen Welt als auch in einer virtuellen Umgebung ist vorwiegend eine dynamische Umgebung zu finden.

- Diskret vs. kontinuierlich

Ob eine Umgebung diskret oder kontinuierlich ist, hängt von der Abstraktion dieser Umgebung ab. Generell kann sich eine diskrete Umgebung nur in endlich vielen Zuständen befinden, eine kontinuierliche hingegen in unendlich vielen. Betrachtet man ein häufiges Beispiel Taxifahren, ist die Umgebung in der Realität kontinuierlich. In dem richtigen Modell kann Taxifahren als diskrete Umgebung definiert werden. Ein weiteres Beispiel für diskrete Umgebungen ist das Spiel Schach.

Durch die genannten Umgebungseigenschaften wird maßgeblich festgelegt, welche Möglichkeiten der Agent hat, die Interaktionen innerhalb dieser Umgebung auszuführen bzw. was er bei der Ausführung einer Aktion zu beachten hat.

2.2 Soziale Robotik

Die beschriebene Agenten-Metapher lässt sich auf verschiedene Bereiche der Wissenschaft anwenden. In dieser Thesis wird der Fokus auf die Umsetzung eines Agenten in Form eines Roboters gesetzt, der sich in der physischen Welt befindet. Dieser Roboter befindet sich außerdem innerhalb eines sozialen Umfelds und muss dabei mit Menschen interagieren und seinen eigenen Bedürfnissen nachgehen. In wissenschaftlichen Arbeiten ist bei der Interaktion zwischen Menschen und Robotern aufgefallen, dass soziale Fähigkeiten auf der Seite des Roboters wichtig sind, um von Menschen akzeptiert zu werden [13, 26]. Diese sogenannten sozialen Roboter führen dabei nicht zwingend routinierte Aufgaben aus, um die Menschen bei einer Arbeit zu unterstützen, sondern sie können eine soziale Beziehung aufbauen, welche für Bildungsunterstützung, therapeutische Zwecke oder die generelle soziale Gesundheit einer Person genutzt werden kann [9]. Laut Fong u. a. sollte ein sozialer Roboter dabei folgende Eigenschaften aufweisen [15]:

- Emotionen ausdrücken und wahrnehmen können.
- Kommunikationsfähig sein.
- Soziale Beziehungen zu Menschen erkennen und aufbauen können.
- Ähnliche Verhaltensweise wie ein Lebewesen aufweisen, dass als Vorbild dient.
- Eigene Persönlichkeitsmerkmale zeigen.

Soziale Roboter haben das Ziel, dass soziale und emotionale Verhalten eines Lebewesens nachzuahmen. Um zum Beispiel das Verhalten eines Menschen nachzuahmen, können anthropomorphe Züge, wie das menschliche Gesicht, Arme, Beine und Sprache genutzt werden. Einer der ersten Realisierungen von sozialen Robotern, die Menschen nachahmen, ist zum Beispiel der Roboter *Kismet* [10]. Dieser Roboter ahmt einen menschlichen Kopf nach und ist mit Ohren, Augen, Augenlider und einem Mund ausgestattet. Ein fortgeschritteneres Beispiel ist der „humanoide“ Roboter *Sophia*, welcher als „first citizen robot of the world“ bezeichnet wird [24].

Trotz fortgeschrittener Technik, sind Forscher noch weit von einem echten humanoiden Roboter entfernt. Das liegt unter anderem an dem Forschungsstand für soziale Interaktionen zwischen Individuen. Für dieses Forschungsgebiet kann die Erforschung von Tier-Mensch Beziehungen eine große Bereicherung sein, welche im Rahmen von künstlichen Tieren genauer betrachtet wird [20].

2.3 Künstliche Tiere

Künstliche Tiere, in der Wissenschaft als *artificial pets* bezeichnet, stellen eine Untergruppe von sozialen Robotern dar. Diese haben das Ziel, ein Tier als Roboter nachzuahmen und eine soziale Mensch-Tierbeziehung aufzubauen. Dabei taucht der Begriff *Companion* oder *Kumpen* häufig in der sozialen Robotik auf, der dem Tier nicht nur eine soziale Eigenschaft zuschreibt, sondern das Tier als eine Art Begleiter beschreibt. Die Beziehung zwischen Menschen und Tieren, insbesondere der Beziehung zwischen Menschen und ihren Haustieren, sind somit soziale Interaktionen, die uns viele Auskünfte über soziales Verhalten und Beziehungen bieten kann. Besonders Haustiere haben eine lange Tradition in der Geschichte der Menschheit und waren die ersten sozial fortgeschrittenen „Spielzeuge“ für Kinder, mit denen sie ihre Pflege- und Betreuungsfähigkeiten verbessern konnten [20]. Tiere und vor allem Hunde werden heutzutage für soziale Interaktionen für bedürftige Menschen, wie zum Beispiel in einer tiergestützten Therapie, benutzt, um den Menschen zu helfen und eine soziale Umgebung zu kreieren [22]. Unter diesen Gesichtspunkten haben Forscher an künstlichen Tieren gearbeitet, um diese Mensch-Tierbeziehung künstlich nachzuahmen. Denn auch wenn Tiere vielversprechende Helfer sind, sind künstliche Tiere im Vorteil. Sie sind robuster, sie können bei kranken Menschen keine zusätzlichen Infektionen auslösen und das Wohlbefinden des Tieres kann nicht gefährdet werden [20].

In der Vergangenheit gab es bereits viele Projekte künstliche Tiere zu konstruieren. Die



Abbildung 2.1: Die Baby Robbe Paro (Quelle: [1])

Spanne reicht von dem sehr simplen „Tamagotchi“, bis hin zu komplexeren Robotern wie „Aibo“, „Tama“ oder „NeCoRo“, die von Katzen und Hunden inspiriert wurden. Ein weiteres Projekt ist die Babyrobbe „Paro“ (siehe Abb. 2.1), mit der Forscher ein künstliches Tier erschaffen haben, welches durch ihre Eigenschaften die Interaktionen und Bindungen bei Personen mit schweren sensorischen und kognitiven Beeinträchtigungen stimulieren soll [19].

Dafür besteht die Robbe aus einem komplexen Sensornetzwerk, welches ihr erlaubt auf ihre Umgebung zu reagieren. Dieses beinhaltet die vier primären Sinne *Sehen* (Licht Sensor), *Hören* (Richtungs- und Spracherkennung), *Gleichgewicht* und einen *Tastsinn*, der zwischen dem Skelett und dem Fell liegt, um Berührungen wahr zu nehmen. Paro kann außerdem seinen Nacken horizontal und vertikal bewegen, seine Flosse nach vorne und hinten bewegen und beide Augenlider unabhängig voneinander bewegen. Mit dieser Sensorik und Aktorik ist es möglich, die Robbe mit ihrer eigenen Persönlichkeit und Emotionen auszustatten.

Um die sozialen Eigenschaften der Robbe zu erforschen, wurde sie unter anderem in einer therapeutischen Studie mit drei behinderten Personen eingesetzt. Zwei davon sind betroffen von der Erkrankung Down-Syndrom und die dritte an dem Hanhart und Möbius Syndrom.

Durch das *Spielen* mit der Robbe in den Therapiesitzungen, fing eine Probandin an öfter über Gefühle von Paro zu sprechen, obwohl sie sonst kaum über Emotionen bzw. generell kaum gesprochen hat, auch wenn sie dazu ermutigt wurde. Alle Patienten wünschten sich eine weitere Arbeit mit Paro nach Beendigung des drei monatigen Experimentes, was zeigt, dass sie eine gewisse Bindung zu der Robbe aufgebaut haben und das ursprüngliche Ziel der Robbe erfüllt worden ist [19].

2.4 Smarte Objekte

Für die in dieser Thesis behandelte Roboter-Pflanze ist neben der Betrachtung von physischen Robotern, die ein Lebewesen darstellen sollen, auch die Betrachtung von smarten Objekten relevant. Denn bei der Pflanze handelt es sich nicht um ein aktives Lebewesen, wie es bei künstlichen Tieren oder sozialen Robotern der Fall ist, sondern um ein Objekt aus dem Alltag.

Zunächst ist die Definition von smarten Objekten relevant. Smarte Objekte sind physische, alltägliche, Gegenstände, die mit Sensorik, Aktorik, Rechenleistung und der Technik für eine Netzwerkkommunikation ausgestattet sind. Mithilfe dieser technischen Ausstattung ist das Objekt in der Lage, seine Umgebungsdaten zu erfassen, diese zu Verarbeiten und die Informationen mit Personen in ihrer Nähe oder über das Netzwerk mit anderen smarten Objekten austauschen zu können [17].

Durch die schnelle Weiterentwicklung von Technologien und die immer kleiner werdenden Sensoren und Aktoren, bieten sich mehr Möglichkeiten in der Ausstattung dieser Objekte. Smarte Objekte finden dadurch immer öfter Einzug in moderne Haushalte [18]. Beispiele für smarte Objekte sind Gegenstände wie smarte Glühbirnen, smarte Thermostate und smarte Tassen.

2.5 Ausdrücken von Affekten

Bei künstlichen Tieren und sozialen Robotern ist es naheliegend, dass sie sich wie ihre lebenden Äquivalente über ihre Gestik, Mimik und Akustik ausdrücken können. Doch bei Robotern, die, aufgrund ihres Aufgabengebiets oder anderweitiger Einschränkungen, keine anthropomorphen Eigenschaften, wie Gesichtsausdrücke oder Gestiken mit ihren Extremitäten, aufweisen können, ist es schwieriger ihre Affekte auszudrücken. Darunter fallen zum Beispiel Roboter im Einsatzgebiet der Menschenrettung oder dem Haushalt, wie Reinigungsroboter [7]. Außerdem siedelt sich der in dieser Arbeit entwickelte Pflanzenroboter in diesem Gebiet ein. Dennoch ist es bei dieser Art von Robotern wichtig, ein Erscheinungsbild zu schaffen, das den Menschen vertraut ist, um soziale Interaktionen zuzulassen. So bezeichnet zum Beispiel ein Opfer eines simulierten Unfalls einen Rettungsroboter, der keinerlei Ausdrücke aufweist, als „creepy“ und als „nicht beruhigend“ [8].

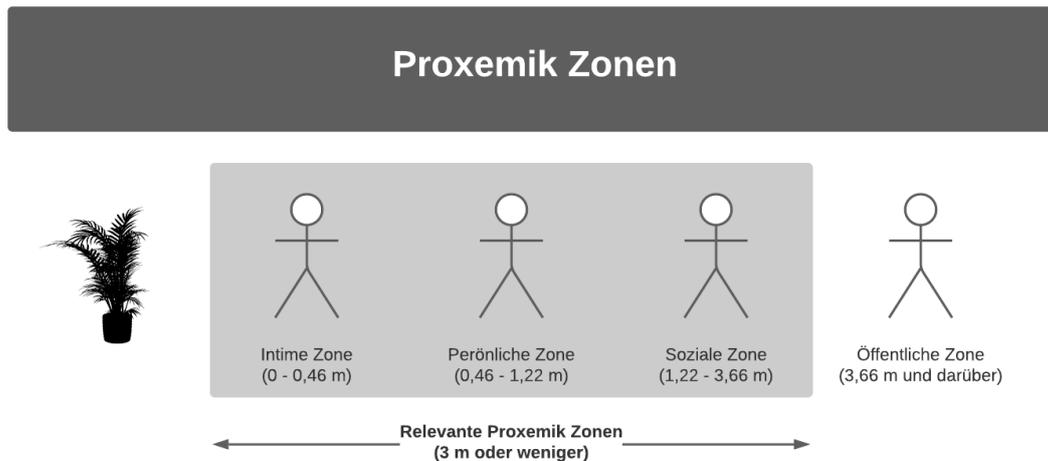


Abbildung 2.2: Proxemik Zonen

Außerdem spielt bei der Betrachtung sozialer Interaktionen zwischen zwei Individuen der Abstand der Individuen eine Rolle. Argyle teilt den Abstand von Individuen, Proxemik genannt, in vier Hauptkategorien ein [4]. Von 0 bis 0,46 m ist die *intime Zone*, von 0,46 bis 1,22 m die *persönliche Zone*, von 1,22 bis 3,66 m die *soziale Zone* und 3,66 m und darüber die *öffentliche Zone* (siehe Abb. 2.2). Da die meisten sozialen Interaktionen innerhalb von 3 m passieren, wird in dieser Thesis nur die intime, persönliche und soziale Zone betrachtet. Die persönliche Zone spielt bei der Betrachtung eine besondere Rolle, da sich die Person innerhalb dieses Bereiches wohl, sicher und geschützt fühlt. Falls andere Individuen oder Objekte in diese Zone eindringen, kann das Unbehagen bei der Person auslösen. Das zeigt, dass die Distanz zwischen zwei agierenden Menschen einen entscheidenden Einfluss auf die Qualität der Interaktion und das Komfortlevel haben, wobei die soziale Zone die präferierte Zone für soziale Interaktionen ist. Diese Erkenntnisse aus der menschlichen Interaktion werden von der Robotik Gemeinschaft auf Mensch-Roboter-Interaktionen übertragen.

Die Benutzung eines künstlichen Gesichtes ist die häufigste Herangehensweise an das Ausdrücken von Affekten eines Roboters. Darüber hinaus gibt es weitere Forschungen, um Affekte mittels *Körperbewegung*, *Körperhaltung*, *Orientierung*, *Farben* und *Geräusche* auszudrücken. Diese Ausdrucksmöglichkeiten können entweder als redundante oder als primäre Quelle genutzt werden [7, 28, 8].

Körperhaltung, Körperbewegung und Orientierung

Beim Betrachten der Körperhaltung, der Körperbewegung und der Orientierung des Körpers können Schlüsse über das Wohlfühlen der Individuen getroffen werden. So agieren Menschen zum Beispiel lieber mit Robotern, die sich in Bewegung befinden anstatt nur starr herumstehen. Die Orientierung des Roboters spielt bei einer Interaktion ebenfalls eine entscheidende Rolle. So bekommt eine Person ein Gefühl der Fürsorge, wenn sich ein Roboter bei einer Interaktion zu der Person orientiert. Im Fall des Rettungsroboters wird außerdem bemerkt, dass es hilfreich ist, wenn der Roboter in die Richtung eines menschlichen Helfers gerichtet ist. Das gibt den Anschein, dass der Roboter den Anweisungen des Helfers zuhört und Folge leistet und der Patient somit mehr Vertrauen in die Handlungen des Roboters gewinnt [7].

Farben

Farben haben vielseitige Bedeutungen, was es schwer macht mit ihnen zu arbeiten. Nijdam untersucht zu diesem Thema mehrere Arbeiten, die über das Mapping von Farben auf Emotionen handeln. Aus diesen Arbeiten zieht er den Schluss, dass das meist wahrgenommene Mapping von Farbe und Ausdrücken die Farbe Rot auf die Emotion wütend, die Farbe Blau auf die Emotion traurig und die Farbe Weiß auf die Emotion entspannt legt [21, 28]. Dieses Mapping wird für diese Thesis übernommen.

Zusätzlich zu dem Ausdrucksmittel der Farbe können Farbmuster eingesetzt werden, um die Signalstärke der Farbe zu verbessern. Hierfür bezieht Song und Yamada Farben, in Verbindung mit Mustern, auf die positive und negative Wahrnehmung eines Roboters [29]. Die höchste Attraktivität konnte dabei die Kombination zwischen der Farbe Blau und dem Muster *langsames geteiltes Lauflicht* hervorrufen. Die feindlichste Wahrnehmung hat der Roboter mit der Kombination zwischen der Farbe Rot und dem Muster *schnell blinkend* erzielt. Diese zwei Kombinationen können so für die Emotionen wütend und traurig übernommen werden. Für die Emotion entspannt und somit der Farbe Weiß, wird die positivste Weiß-Muster Kombination genutzt, welche die Farbe Weiß mit dem Muster *langsames Lauflicht* verbindet. Die beiden Muster *Lauflicht* und *geteiltes Lauflicht* sind zur Veranschaulichung in Abb. 2.3 zu sehen.



Abbildung 2.3: Links ist das *Lauflicht* und rechts das *geteilte Lauflicht* Farbmuster zu sehen.

Geräusche

In der Robotik werden *Geräusche* als zusätzliche Möglichkeit eingesetzt, um Ausdrücke zu verstärken. Dabei werden in dieser Thesis nur Geräusche untersucht die Folgen von Tönen sind. So beschreibt Bartneck die Kombination von Emotionen mit den dazugehörigen Tönen wie folgt [6]:

- Fröhlichkeit: schnelles Tempo, mäßige Variationen im Timing, mäßiger bis lauter Klangpegel, meist Staccato-Artikulation, schnelle Tonanschläge, helles Timbre, leichtes oder kein Vibrato.
- Traurigkeit: langsames Tempo, niedriger Schallpegel, Legato-Artikulation, langsame Tonanschläge, langsames und tiefes Vibrato, weiche Klangfarbe, Schlussrhythmus, flache Intonation.
- Wut: schnelles Tempo, hoher Schallpegel, kein Endrhythmus, meist nicht legato artikuliert, sehr scharfe Tonanschläge, scharfes Timbre, verzerrte Töne;
- Angst: große Tempovariationen, große Abweichung im Timing, sehr niedriger Tonpegel, große dynamische Variation, meist Staccato-Artikulation, schnelles und unregelmäßiges Vibrato, Pausen zwischen Phrasen und weiches Spektrum.
- Zartheit: langsames Tempo, niedriger bis mittlerer Schallpegel, Legato-Artikulation, langsame Tonanschläge, weiches Timbre und intensives Vibrato.

Der Fokus von Bartneck liegt auf Musik Parametern, trotzdem stimmen die akustischen Hinweise, die mit jeder Emotion verbunden sind, mit anderen Forschungen aus diesem Feld überein [4, 5, 27].

Mithilfe der genannten Ausdrucksarten ist es einem eingeschränkten Roboter trotzdem möglich, seine Affekte auszudrücken, um im Kontakt mit Menschen als sozialer Partner akzeptiert zu werden.

2.6 Zusammenfassung der Analyse

Die Betrachtung dieser verschiedenen Arten von smarten und sozialen Objekten hilft dabei, den Pflanzenroboter als eigenständiges Individuum im aktuellen Forschungsstand einzuordnen. Somit zeigt sich die Wichtigkeit der Einordnung der Umgebung in der sich ein eigenständiges Individuum bzw. ein physischer Agent befindet und was für Auswirkungen diese auf das Verhalten und das Softwaredesign des Agenten haben. Außerdem wird über die Thematik der sozialen Robotik die Relevanz der sozialen Aspekte eines Roboters ersichtlich. Diese sozialen Aspekte übertragen auf einen Roboter und in dieser Thesis speziell betrachtet auf künstliche Tiere, lassen zudem Schlüsse über den Einsatz bei hilfsbedürftigen Menschen zu, um deren sozialen Kontakte über sie zu erweitern und ein aktiveres und sozialeres Umfeld zu schaffen.

Mit diesen Aspekten können die Eigenschaften der Pflanze eingegrenzt werden, dass sie soziale Eigenschaften wie ein sozialer Roboter, ein eigenes Leben mit Bedürfnissen und Emotionen wie ein künstliches Tier und die technische Ausstattung zur Kommunikation und Wahrnehmung wie ein smartes Objekt aufweist.

Welche Eigenschaften und welches Verhalten der Pflanzenroboter konkret in dieser Thesis annehmen wird, wird in Kapitel 3.1 behandelt.

3 Umsetzung

Dieses Kapitel umfasst die gesamte Umsetzung des Roboters. Diesbezüglich wird ein Konzept erstellt, das auf den Aspekten der Analyse beruht und eine genaue Betrachtung auf die Ziele des umzusetzenden Roboters liefert. Darauf basierend wird die Umsetzung des physischen Aufbaus (siehe 3.2) des Roboters, als auch das Softwaredesign (siehe 3.3) erstellt. In welcher Form die Ziele des Konzepts in physischer und Software Form umgesetzt werden konnten, wird in Kapitel 3.4 evaluiert.

3.1 Konzept

In diesem Kapitel wird das Konzept für die Konstruktion der Pflanze anhand des Analyse-Kapitels angefertigt. Die Roboterpflanze wird in der weiteren Arbeit als *Plantbot* bezeichnet.

Das Ziel dieser Thesis ist es, eine Roboterpflanze, die als Mitbewohner für ein Smart Home fungiert, aufzubauen und zu entwickeln um ein aktiveres und sozialeres Umfeld in der Wohnung zu schaffen. Dabei verhält sich die Pflanze wie ein eigenständiges Lebewesen, dass sich um ihre eigenen Bedürfnisse kümmert und vorgibt Emotionen zu empfinden. Folglich wird die Pflanze nach dem Konzept eines physischen Agenten aufgebaut, der sich innerhalb einer Umgebung befindet, diese wahrnehmen und in ihr autonom seine Ziele verfolgen kann. Um diese Zielsetzung definieren zu können, muss zunächst die Umgebung, in der sich der Plantbot befindet, klar definiert werden.

3.1.1 Umgebung

Für diese Thesis wird sich die Pflanze in der Smart Home Wohnung *Living Place*¹ von der HAW Hamburg befinden. Diese Wohnung erstreckt sich über ein Stockwerk und besitzt eine Fläche von 140 m². Dort befinden sich die für die Pflanze frei zugänglichen

¹Living Place, <https://livingplace.haw-hamburg.de/>. [07.06.2021]

Bereiche Esszimmer, Schlafzimmer, Küche und Wohnzimmer. Lediglich das Badezimmer ist begrenzt durch eine Tür, die die Pflanze nicht ohne Hilfe bewegen kann. Innerhalb des Schlafzimmers befindet sich ein festgelegter Punkt, der als Basisaufenthaltort des Plantbots festgelegt ist. Dies bietet die Möglichkeit den Akku des Plantbots aufzuladen und über die Nacht ruhen zu lassen.

Zudem ist die Wohnung mit Smart Home Technologien ausgestattet, die über einen Server im Living Place angesteuert werden können. Darunter befinden sich Kameras, Mikrofone, Rollläden, Lampen und Heizungen. Die Fenster lassen sich über eine mechanische Erweiterung kippen. Objekte innerhalb des Netzwerkes können über eine API mit dem Server kommunizieren und Informationen mit der in der Wohnung verbauten Sensorik und Aktorik austauschen.

Durch diese zusätzlichen Informationen für einen Agenten können die Eigenschaften der Umgebung wie folgt festgelegt werden:

- Die Wohnung ist zugänglich, da durch die verbaute Sensorik in der Wohnung und durch die verbaute Sensorik der Pflanze alle für die Pflanze relevanten Umgebungsdaten bereitgestellt werden können.
- Die Wohnung ist nicht-deterministisch, da mehrere Aktionen durch mehrere Agenten parallel ausgeführt werden können und sich somit der Agent nicht auf einen immer gleichen Ablauf einer Aktion verlassen kann.
- Die Wohnung ist Episodisch, da die Pflanze während einer Aktion die Umgebung wahrnimmt und daraufhin handelt. Informationen dieser Aktion werden nicht gespeichert und jede Aktion bezieht sich nur auf die aktuell gemessenen Sensor Daten.
- Die Wohnung ist dynamisch, da sich mehrere Akteure gleichzeitig in der Wohnung befinden und Aktionen ausführen können.
- Die Wohnung ist kontinuierlich, da es viele, auch unbekannte, Datenquellen gibt, die sich ständig ändern.

Diese Umgebungseigenschaften haben für den Plantbot zur Konsequenz, dass er kontinuierlich die Informationen seiner Umwelt messen und abfragen muss, um darauf reagieren zu können. Diese Aspekte müssen in dem Softwaredesign (siehe Abschnitt 3.3) berücksichtigt werden.

3.1.2 Einordnung Plantbot

Wie in Abschnitt 2.2 beschrieben, ist für die Akzeptanz eines Roboters als Mitbewohner wichtig, dass der Roboter soziale Fähigkeiten aufweist. Da in dieser Thesis der Plantbot allerdings nicht als vollständiger sozialer Roboter gilt, müssen nicht alle in Abschnitt 2.2 aufgezählten sozialen Eigenschaften umgesetzt werden, da der Plantbot eine Mischung aus einem smarten Objekt, einem künstlichen Tier und einem sozialen Roboter ist. Die Pflanze wird durch die technische Ausstattung zu einem smarten Objekt, in dem sie mit Personen in ihrem Umfeld und mit anderen Objekten über das Netzwerk kommunizieren und die notwendige Umgebung wahrnehmen kann. Darüber hinaus ist der Plantbot in der Lage sich zu bewegen, Emotionen darzustellen und somit seinen eigenen Willen zu äußern, wie vergleichbare künstliche Tiere. Der Plantbot ist trotzdem eingeschränkt in seinen Möglichkeiten sich auszudrücken, da er keine Extremitäten o. ä. besitzt. Zudem repräsentiert er kein lebendes Lebewesen, was ihn zu keinem vollständigen künstlichen Tier macht. Die Mischung aus einem sozialen Roboter, einem künstlichen Tier und einem smarten Objekt, ermöglicht es, das Verhalten, das Erscheinungsbild und die Bedürfnisse der Pflanze nach Vorbildern aus den jeweiligen Bereichen zu gestalten.

3.1.3 Ziele und Verhalten des Plantbots

Der Plantbot soll ein Verhalten aufweisen, was dem eines eigenständigen Mitbewohners entspricht. Darunter zählt die Versorgung seiner eigenen Bedürfnisse und die sozialen Interaktionen mit Personen in der Wohnung. Außerdem soll er eine Nachtruhe einhalten, um die Bewohner der Wohnung nicht beim Schlafen zu stören. Zusätzlich soll der Plantbot, in Abhängigkeit von seinen Bedürfnissen, Emotionen simulieren und diese an seine Umwelt kommunizieren.

Die Bedürfnisse des Plantbots setzen sich dabei aus den Bedürfnissen der Pflanze und des Roboters zusammen. Eine Pflanze benötigt Wasser, die richtige Zimmertemperatur und die Einstrahlung von Sonnenlicht, um wachsen und Überleben zu können. Zusätzlich benötigt ein Roboter Energie in Form von Strom, um seine elektrischen Bauteile weiterhin bedienen zu können. Folglich sind die Grundbedürfnisse des Plantbots die Versorgung mit Wasser, Licht, Strom und die geeignete Zimmertemperatur in der Wohnung.

Um seinen Wasserhaushalt aufrecht halten zu können, soll der Plantbot imstande sein,

seine aktuelle Wassersituation messen zu können und bei Bedarf in der Dusche der Wohnung seinen Wasserhaushalt auffüllen zu können. Die Versorgung mit Sonnenlicht geschieht darüber, dass sich der Plantbot über den Tag verteilt an geeigneten Fenstern aufhält, um den Prozess der Fotosynthese am Laufen zu erhalten. Des Weiteren soll der Plantbot den Strom zum Aufladen des Akkus an seinem Basisaufenthaltort im Schlafzimmer beziehen. Zur Kontrolle der Temperatur innerhalb der Wohnung soll der Plantbot mit der Wohnung kommunizieren und Fenster öffnen bzw. schließen und die Heizung regulieren können.

Für die Akzeptanz des Plantbots sind soziale Interaktionen hilfreich, weshalb der Plantbot Personen beim Betreten der Wohnung begrüßen kann. Dazu fährt der Plantbot zu einer neu eingetretenen Person und begrüßt sie. Dies soll sich über ein Schütteln der Pflanze äußern. Außerdem soll die Pflanze zu der eingetretenen Person gerichtet sein, wie in Kapitel 2.5 erläutert wird.

Wenn der Plantbot alle seine Grundbedürfnisse gedeckt hat, stellt sich der Plantbot an einen beliebigen Ort in der Wohnung und wartet, bis ein neues Event ausgelöst wird. Ist die gewählte Position des Plantbots durch andere Akteure nicht erwünscht, können ihm diese signalisieren sich eine neue Position zu suchen.

In der Nachtruhe soll sich der Plantbot an seinem Basisaufenthaltort befinden. In diesem Zeitraum soll er keine Töne oder sonstiges Verhalten von sich geben.

Die durch den Plantbot simulierten Emotionen ergeben sich aus den Erkenntnissen aus Abschnitt 2.5. Sie umfassen: entspannt, traurig und wütend. Die angezeigte Emotion leitet sich aus dem Zustand der Bedürfnisse der Pflanze ab. Außerdem sind sie gestaffelt und je nach Abweichung der Bedürfnisse fällt die Emotion zuerst von der neutralen Emotion entspannt auf die schlechtere Emotion traurig und dann auf die Emotion wütend.

Die Ziele, die in der Umsetzung mit dem Plantbot erreicht werden sollen, sind in den Tabellen 3.1 bis 3.8 zu sehen.

Tabelle 3.1: Ziel 01 des Plantbots

Nr.	01
Ziel	Versorgung mit Wasser
Beschreibung	Fällt die Erde des Plantbots unter 85 % Feuchtigkeit, soll er sich mit Wasser versorgen.

Tabelle 3.2: Ziel 02 des Plantbots

Nr.	02
Ziel	Versorgung mit Licht
Beschreibung	Jeden Tag soll der Plantbot vier Stunden in der Sonne verbringen. Dazu wählt er das Fenster mit den besten Lichtverhältnissen.

Tabelle 3.3: Ziel 03 des Plantbots

Nr.	03
Ziel	Temperatur in einem angenehmen Bereich halten
Beschreibung	Bei Temperaturen niedriger als 18°C sollen Fenster geschlossen und Heizungen angemacht werden, bei Temperaturen höher 24°C sollen Fenster geöffnet und Heizungen abgestellt werden.

Tabelle 3.4: Ziel 04 des Plantbots

Nr.	04
Ziel	Versorgung mit Strom
Beschreibung	Bei einem Akkustand von unter 15 % soll der Plantbot zu seinem Basisaufenthaltort fahren und warten, bis er aufgeladen wird.

Tabelle 3.5: Ziel 05 des Plantbots

Nr.	05
Ziel	Begrüßen von neuen Personen
Beschreibung	Beim betreten der Wohnung soll diese Person von dem Plantbot mit einem Schütteln begrüßt werden.

Tabelle 3.6: Ziel 06 des Plantbots

Nr.	06
Ziel	Ausdrücken von Emotionen
Beschreibung	Der Plantbot soll in der Lage sein die Emotionen entspannt, traurig und wütend auszudrücken. Als Basis dafür werden die Bedürfnisse des Plantbots genutzt.

Tabelle 3.7: Ziel 07 des Plantbots

Nr.	07
Ziel	Nachtruhe einhalten
Beschreibung	Zwischen 22:00 Uhr und 8:00 Uhr soll sich die Pflanze an ihrem Basisaufenthaltort aufhalten und keine Töne von sich geben.

Tabelle 3.8: Ziel 08 des Plantbots

Nr.	08
Ziel	Wenn er an einem Ort entspannt, soll man ihn wegschicken können
Beschreibung	Wenn der Plantbot nichts zu tun hat entspannt er an einem beliebigen Ort in der Wohnung. Dabei soll man ihm signalisieren können, das er wo anders hinfahren soll.

Ausdrücken der Emotionen

Die Emotionen umfassen entspannt, traurig und wütend. Diese Emotionen werden für diese Thesis festgelegt, da die Arbeit von Song und Yamada als Grundlage für diese Thematik angesehen wird und daraus hervorgeht, dass diese drei Emotionen am besten über die Ausdrucksarten der Farbe, des Sounds und der Körperbewegung dargestellt werden können [28]. Die Körperbewegung für den Ausdruck der Emotionen wütend und traurig ist in der Arbeit von Song und Yamada als Vibration des Körpers dargestellt, was in dieser Thesis als ein Schütteln der Pflanze dargestellt werden soll. Diese Vibration ist in schnelles und in langsames vibrieren eingeteilt, folglich soll das Schütteln in schnell und langsam umgesetzt werden. Die Geräusche des Plantbots ergeben sich aus der Auflistung in Abschnitt 2.5. Diese Auswahl deckt sich mit den Erkenntnissen aus [28].

Das Ausdrücken einer Emotion soll auf Dauer über eine RGB-Leiste geschehen. Lediglich beim Wechsel einer Emotion sollen die Ausdrucksarten über die Körperbewegung und über den Sound genutzt werden.

Die Kombinationen der Farben und der Farbmuster für das dauerhafte Ausdrücken einer Emotion wird im Folgenden erläutert.

- Entspannt: Weiß mit dem Farbmuster *langsames Lauflicht*.
- Traurig: Blau mit dem Farbmuster *langsames geteiltes Lauflicht*.
- Wütend: Rot mit dem Farbmuster *schnelles geteiltes Lauflicht*.

Wenn der Plantbot zwischen den Emotionen wechselt, werden die folgenden Ausdrucksarten verwendet, um die jeweilige Emotion einzuleiten:

- Entspannt
 - Farbe: Weiß mit dem Farbmuster *langsames geteiltes Lauflicht*.
 - Körperbewegung: Auf der Stelle im Kreis drehen.
 - Sound: Ruhiger, flacher Piepton.
- Traurig
 - Farbe: Blau mit dem Farbmuster *langsam atmend*.
 - Körperbewegung: Langsames schütteln.
 - Sound: Fallender Piepton.
- Wütend
 - Farbe: Rot mit dem Farbmuster *schnell blinkend*.
 - Körperbewegung: Schnelles schütteln.
 - Sound: Steigender Piepton.

Mithilfe dieser Ausdrucksarten ist der Plantbot imstande, seine Emotionen an seine Umwelt zu kommunizieren.

3.2 Physischer Aufbau

In diesem Kapitel wird der physische Aufbau rund um die Pflanze behandelt. Dieser wird benötigt, um die Pflanze als Plantbot in der Wohnung agieren zu lassen. Der Aufbau besteht dabei aus einer Plattform, die mit einem Sensor- und Aktor-Netzwerk ausgestattet ist, damit sich die Pflanze bewegen, Eindrücke sammeln und sich mitteilen kann.

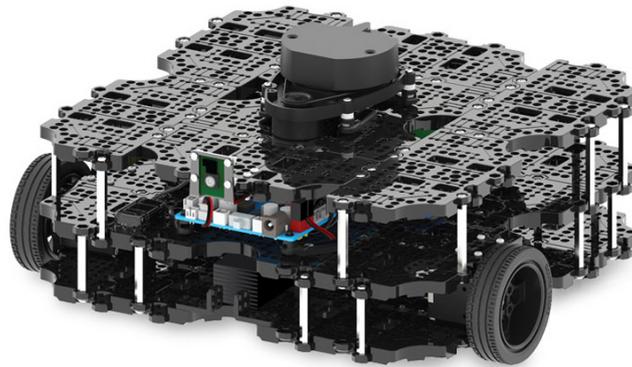


Abbildung 3.1: Turtlebot3 Waffle Pi

3.2.1 Roboter Plattform

Damit es der Pflanze möglich ist, sich frei in ihrer Umgebung zu bewegen, benötigt sie eine Plattform. Dies bietet der Pflanze die Möglichkeit, darauf zu stehen und sich mithilfe von ausgestatteten Motoren fortzubewegen. Dafür wird in dieser Thesis der Bausatz Turtlebot3 Waffle Pi (siehe Abb. 3.1) verwendet². Der Grundaufbau dieser Roboterplattform beinhaltet zwei Motoren, einen Akku, einen Laser Distanz Sensor (LDS)³, eine Kamera, einen Raspberry Pi 3b+ und eine OpenCR Platine.

Die OpenCR Platine⁴ ist für Robot Operating System (ROS) embedded Systeme entwickelt worden, um Open-Source-Hardware sowie Software für den Turtlebot bereitstellen zu können. Sie übernimmt dabei die Funktion, den Akku zu überwachen und die Motoren sowie das Soundmodul auf der Platine anzusteuern.

Neben der OpenCR Platine befindet sich ebenfalls ein Raspberry Pi auf dem Turtlebot. Dieser ist mit der Kamera, dem 2D 360° LDS und der OpenCR Platine verbunden und ist für die Ausführung der ROS Pakete zuständig (siehe 3.3.3). Der LDS ermöglicht es, die gesamte Umgebung des Plantbots kontinuierlich zu erfassen. Mithilfe dieser Daten können Anwendungen, wie Simultaneous Localization and Mapping (SLAM) und die Navigation des Plantbots, umgesetzt werden, welche für die Selbständigkeit der Pflanze benötigt werden. Über das Netzwerk kann der Raspberry Pi somit alle relevanten Turtlebot Daten mit dem Server der Living Place Wohnung teilen, damit sie dort mit weiteren

²Turtlebot, Website der Turtlebot Plattformen, <https://www.turtlebot.com>. [07.06.2021]

³LDS, Manual für den Laser Distanz Sensor, https://emanual.robotis.com/docs/en/platform/turtlebot3/appendix_lds_01/#appendix-lds01. [07.06.2021]

⁴OpenCR, Manual der OpenCR Platine, <https://emanual.robotis.com/docs/en/parts/controller/opencr10>. [07.06.2021]

ROS-Paketen verarbeitet werden können.

Für die Arbeit mit der Pflanze muss die Plattform erweitert werden. Wie in Abb. 3.1 zu sehen, besteht die Plattform aus zwei Ebenen. Auf der obersten Ebene befindet sich der LDS. Demnach muss für die Pflanze eine dritte Ebene hinzugefügt werden, damit der Pflanze ein sicherer Stand gewährleistet werden kann (siehe Abb. 3.4).

Mithilfe dieses technischen Aufbaus ist die Pflanze in der Lage, sich in der Wohnung frei zu bewegen, Objekte in ihrer Umgebung zu erkennen und sich über das Soundmodul auszudrücken. Die weiteren benötigten Ein- und Ausdrucksmöglichkeiten werden über zusätzliche Sensorik und Aktorik realisiert.

3.2.2 Sensorik und Aktorik

Neben der verwendeten Technik des Turtlebot Bausatzes werden zusätzlich Sensoren und Aktoren benötigt, um die Anforderungen aus dem Konzept umsetzen zu können. Für die im Konzept festgelegten Verhaltensmuster werden ein Feuchtigkeitssensor, ein Helligkeitssensor und ein Temperatursensor benötigt. Die gemessenen Werte müssen an den Raspberry Pi übertragen werden, der die Informationen an das ROS-Netzwerk weiterleitet. Der Raspberry Pi besitzt keine analogen Eingänge, weshalb in dieser Thesis ein zusätzlicher Arduino Nano eingesetzt wird. Dieser verfügt über die benötigten analogen Eingänge und kann damit die Messwerte einlesen. Anschließend überträgt er die gemessenen Sensorwerte über einen I²C-Bus an den Raspberry Pi. Zusätzlich wird über den Arduino die Aktorik angesteuert, um die Emotionen des Plantbots auszudrücken. Die Befehle zur Ansteuerung der Aktorik empfängt der Arduino ebenfalls über den I²C-Bus des Raspberry Pis. Ausführlich ist die Kommunikation über den I²C-Bus in 3.3.5 beschrieben.

Für den Arduino und die Anschlüsse der Sensorik und Aktorik wird eine Lochrasterplatine angefertigt, um den Aufbau der Technik so wartbar wie möglich zu gestalten (siehe Abb. 3.2). Im Anhang ist der Schaltplan der Platine zu finden und im digitalen Anhang sind die Datenblätter zu den Sensoren zu finden.

Feuchtigkeitssensor

Bei dem verwendeten Feuchtigkeitssensor handelt es sich um einen Sensor der Firma Seeed Studio. Er liefert ein analoges Signal, das einen Wert zwischen 0 und 950 annimmt.

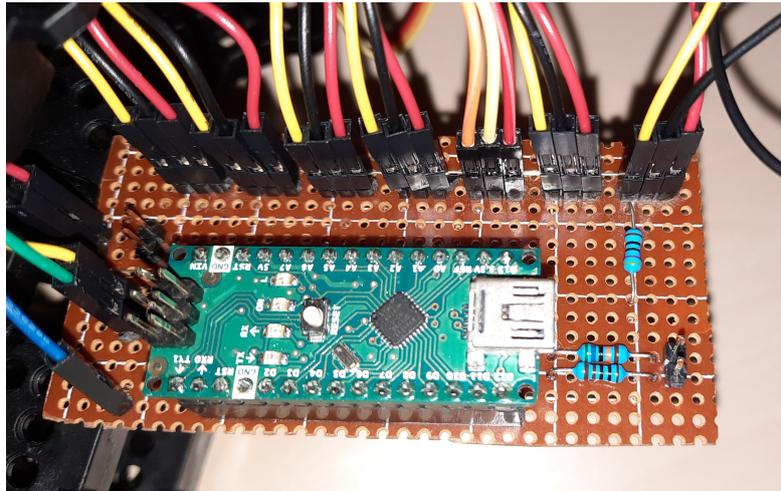


Abbildung 3.2: Lochrasterplatine für den Arduino und der zusätzlichen Sensorik und Aktorik

Laut Angaben des Herstellers befindet sich der Wert bei trockener Erde zwischen 0 und 300, bei feuchter Erde zwischen 300 und 700 und wenn er von Wasser umgeben ist zwischen 700 und 950. Experimentelle Messungen für die verwendete Pflanze und Erde in dieser Arbeit haben gezeigt, dass der gemessene Wert bei sehr trockener Erde bei ca. 450 liegt und bei frisch gegossener Erde bei ca. 700. Da die Feuchtigkeit mit diesem Sensor punktuell in der Erde gemessen wird, kommen zur Erhöhung der Genauigkeit zwei dieser Sensoren zum Einsatz. Diese werden gegenüber voneinander in der Erde des Plantbots platziert. Die gemessenen Werte, der beiden Sensoren, werden miteinander gemittelt und anschließend über den I²C-Bus an den Raspberry Pi übertragen.

Helligkeitssensor

Die Lichtqualität stellt für die Pflanze einen wichtigen Rohstoff dar. Ohne die Einwirkung von Licht findet keine Photosynthese und somit kein Wachstum statt. Aus diesem Grund wird die Einwirkung des Lichts als ein weiterer Faktor angesehen, der die Emotionen der Pflanze beeinflusst. Um die Lichteinwirkung digital verarbeiten zu können, wird in dieser Thesis ein Helligkeitssensor verwendet, der die direkte Einstrahlung des Lichts misst. Unter Verwendung der Fotodiode „GL5528“ wird ein analoger Wert zwischen 0 und 1024 gemessen und anschließend über den I²C-Bus an den Raspberry Pi übertragen.

Temperatursensor

Für das Wohlbefinden der Pflanze spielt außerdem die Umgebungstemperatur eine Rolle [23]. Diese wird mit einem „DHT22“ Sensor gemessen und über einen „1-Wire-Bus“, mit einer Auflösung von 0,1 °C, an den Arduino übertragen. Hierfür wird eine Bibliothek der Firma *Adafruit* verwendet⁵. Neben der Temperatur misst dieser Sensor zusätzlich die Luftfeuchtigkeit der Umgebung. Auf den Luftfeuchtigkeitswert wird in dieser Thesis nicht weiter eingegangen.

Abstandssensoren

Damit die Pflanze während der Fahrt problemlos durch die Wohnung gelangt und nicht an Hindernisse stößt, kommen, als zusätzliche Schutzmaßnahmen zum LDS, zwei Abstandssensoren zum Einsatz. Welche Objekte vor der Pflanze erkennen und somit ein sofortiges Stoppen der Pflanze einleiten können. Des Weiteren kann mit der Pflanze über die Distanzsensoren kommuniziert werden, indem man sich vor einen der Sensoren stellt. Die Abstandssensoren sind in den rechten und in den linken Distanzsensor eingeteilt (in Fahrtrichtung betrachtet). Die Reichweite der verwendeten Sharp (GP2Y0A41SK0F) Sensoren beträgt 4 bis 30 cm.

Aktorik

Neben der Sensorik werden Aktoren zum Ausdrücken der Affekte benötigt. Die zwei Ausdrucksarten über die Körperbewegung und über Sound werden bereits über die Motoren und das Soundmodul der Turtlebot Plattform abgedeckt. Lediglich die dritte Ausdrucksart, über Farben, wird ebenfalls über den Arduino und mit der Lochrasterplatine angesteuert. Um die Farben darzustellen, werden RGBs verwendet, die aus einer roten, einer grünen und einer blauen LED bestehen, die separat angesteuert werden können. Das macht es möglich, jeder RGB eine Farbe aus dem RGB-Farbspektrum zuzuweisen. Durch den Einsatz einer RGB-Leiste mit 32 RGBs können die Farben und die Farbmuster dargestellt werden.

Zur Ansteuerung der RGB-Leiste über den Arduino stellt *Adafruit* eine Bibliothek zur

⁵DHT-sensor-library, GitHub-Seite der Adafruit DHT Bibliothek, <https://github.com/adafruit/DHT-sensor-library>. [07.06.2021]

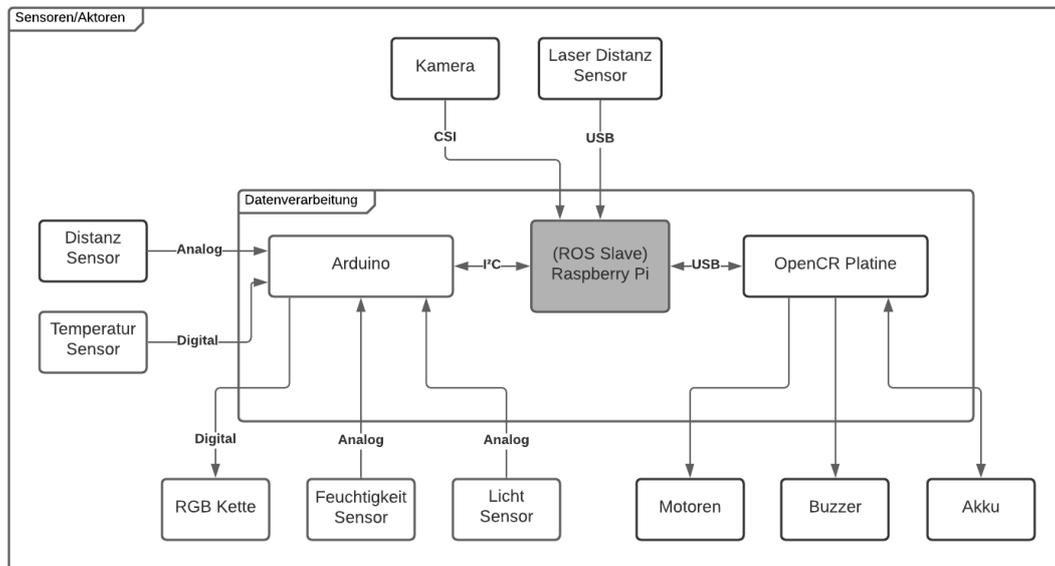


Abbildung 3.3: Die verwendete Sensorik und Aktorik mit dessen Verarbeitungseinheiten.

Verfügung⁶. Diese abstrahiert den Datenbus, um eine einfache Ansteuerung zu ermöglichen.

Der fertige Aufbau mit der Turtlebot Plattform, der zusätzlichen Sensorik und Aktorik und der Pflanze ist in Abb. 3.4 zu sehen. In Abb. 3.3 sind alle verwendeten Sensoren und Aktoren mit ihren Verarbeitungseinheiten dargestellt, um Klarheit über die Zugehörigkeit der Bauteile zu geben.

3.3 Software Design

Durch den Aufbau können Daten aus der realen Welt digital gemessen werden und die Pflanze erhält die Möglichkeit, sich in der Welt zu bewegen und auszudrücken. Doch um die gemessenen Werte verarbeiten und die Pflanze anschließend darauf reagieren lassen zu können, wird das nachfolgende Softwarekonzept benötigt.

Als Grundlage für das Umsetzen der Software wird in dieser Thesis ROS verwendet. Dabei handelt es sich nicht um ein richtiges Betriebssystem, wie der Name suggeriert, sondern es

⁶Adafruit_NeoPixel, GitHub Seite der Adafruit NeoPixel Bibliothek, https://github.com/adafruit/Adafruit_NeoPixel. [07.06.2021]



Abbildung 3.4: Der Plantbot Roboter.

bietet Dienste an, die man von einem Betriebssystem kennt. Die Hauptbestandteile sind Hardware Abstraktionen, Bereitstellung von Gerätetreibern, Nachrichtenaustausch zwischen Prozessen, Programmbibliotheken bereitstellen und eine Paketverwaltung⁷. ROS bietet somit die notwendige Basisstruktur an, um sich auf die Entwicklung der Prozesse des Plantbots fokussieren zu können.

In dieser Thesis wird die Version *ROS Melodic Morenia* genutzt, welches auf ROS 1 basiert. Dies wird auf einem Ubuntu 18.04 Server Abbild auf dem Raspberry Pi und auf dem Living Place Server installiert.

Das Softwaredesign für diese Thesis besteht aus zwei Hauptkomponenten, dem Navigator, der dafür zuständig ist die Sensorwerte zu verarbeiten und daraufhin Aktionen auszuführen, und der Emotion Engine, die den Status über das Wohlempfinden der Pflanze über eine Emotion darstellt und diesen an die Umwelt mitteilt. Neben diesen zwei Hauptkomponenten werden weitere Komponenten zum Einlesen und verarbeiten der Sensorwerte und zum ansteuern der Aktoren benötigt. Alle benötigten Software-Komponenten mit dargestelltem Informationsfluss werden in Abb. 3.5 dargestellt und in den nachfolgenden Kapiteln erläutert.

3.3.1 Navigator

Der Navigator ist dafür zuständig, die Sensorwerte des Plantbots zu verarbeiten und daraufhin die entsprechenden Aktionen auszuführen. Durch die Living Place Umgebung muss der Plantbot in der Lage sein, auf Veränderungen der Umwelt reagieren zu können, weil er nicht der einzige Akteur in dieser Umgebung ist. Aus diesem Grund wird das Softwaredesign des Navigators an der Subsumption-Architektur angelehnt, welche eine reaktive Roboterarchitektur darstellt. Denn reaktive Architekturen sind dazu konzipiert, kontinuierlich die Umgebungsdaten zu verarbeiten und dementsprechend reagieren zu können. Die Subsumption-Architektur wurde durch Rodney Brooks populär und stellt sich gegen die klassische Good Old-Fashioned Artificial Intelligence (GOF AI), die mit einer symbolischen Repräsentation der Welt versucht ein intelligentes Verhalten zu erreichen [30]. Dies macht er, indem er den verfügbaren Aktionen bestimmte Eingabewerte zuweist und sie mit einer Priorisierung versieht. Folglich wird immer nur die Aktion ausgeführt, welche von einer Veränderung eines Eingabewertes ausgelöst wird und darunter die höchste Priorisierung besitzt.

⁷ROS, Webseite des Robert Operating Systems, <http://ros.org>. [07.06.2021]

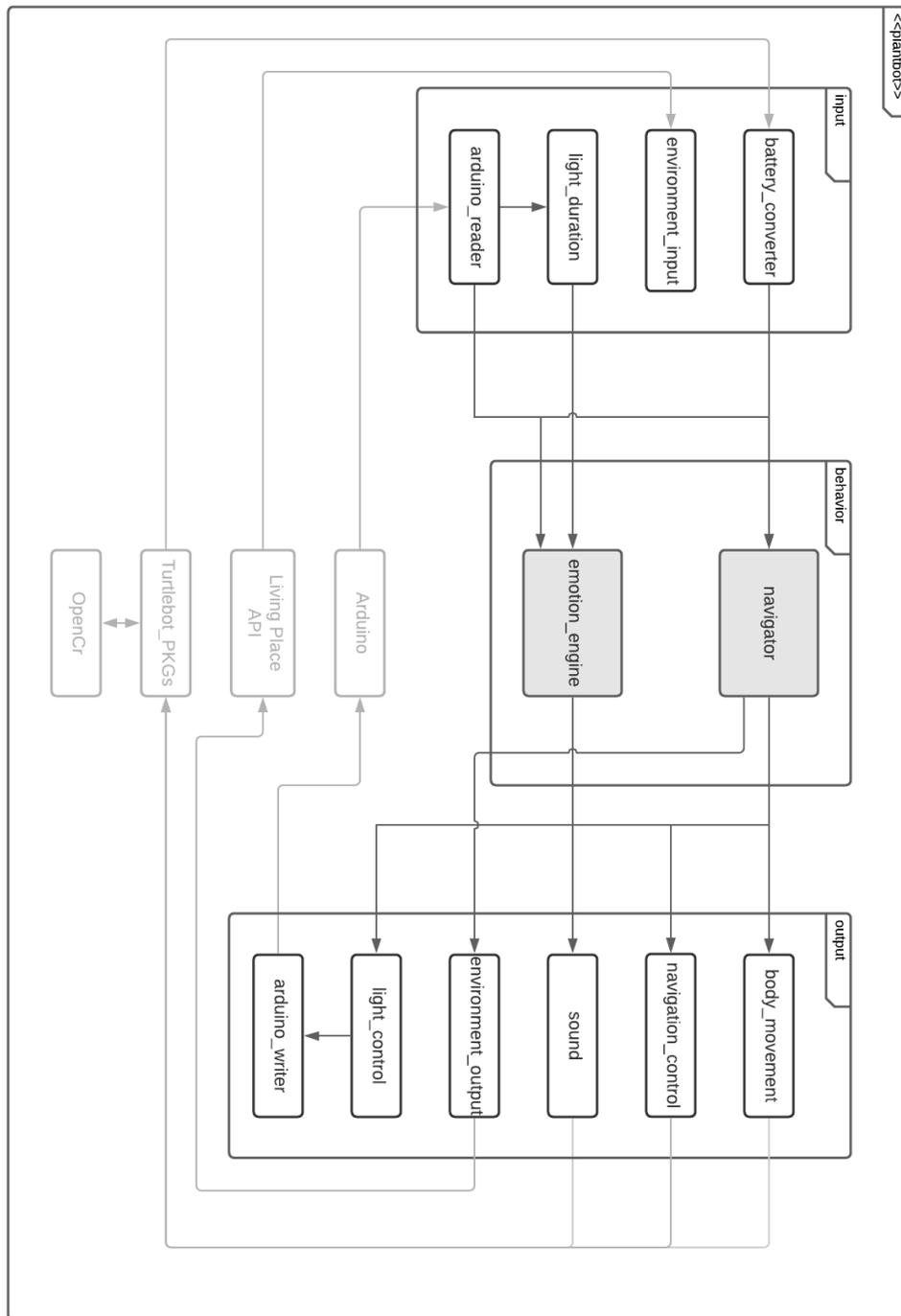


Abbildung 3.5: Informationsfluss zwischen den ROS-Komponenten des Plantbots und der Umwelt.

Die Aktionen, die der Plantbot umsetzen soll, sind in Abb. 3.6 dargestellt und sind aus den Zielen des Konzeptes (siehe 1.2) abgeleitet. In dem Diagramm ist die Priorisierung der Aktionen von unten nach oben gestaffelt. Die Aktion *schlafen* ist somit die höchst priorisierte Aktion und wird mit dem richtigen Eingabewert allen anderen Aktionen vorgezogen. Höher priorisierte Aktionen unterbrechen außerdem niedriger priorisierte Aktionen. Das bedeutet, dass wenn der Navigator einen Eingabewert bekommt, der einer höher priorisierten Aktion zugewiesen ist als gerade ausgeführt wird, dann unterbricht der Navigator die aktuelle Aktion und führt die höher priorisierte Aktion aus. Folglich muss der Navigator gewährleisten, dass immer nur eine Aktion zu einem Zeitpunkt ausgeführt wird.

Die Aktionen aus Abb. 3.6 werden im Folgenden erläutert.

Sleep

Die höchst priorisierte Aktion ist die Aktion schlafen. Diese Aktion setzt Ziel Nummer 07 (siehe 3.7) des Konzeptes um. Folglich muss der Plantbot in der Zeit zwischen 22:00 und 8:00 Uhr die Nachtruhe einhalten. Somit ist die Uhrzeit der benötigte Eingabewert um die Aktion auszulösen. Dies bedeutet, dass er seine aktuelle Aktion abbrechen und sich zu seinem Basisaufenthaltsort im Schlafzimmer begeben muss. Während der Nachtruhe darf er sich nicht bewegen, keine Töne von sich geben und die RGB-Leiste wird ausgeschaltet. Erst am nächsten Morgen um 8:00 Uhr kann er seinen anderen Aktionen nachgehen.

Charge Battery

Die nächst priorisierte Aktion ist *aufladen des Akkus*, welche Ziel Nummer 04 (siehe 3.4) umsetzt. Ohne den Akku ist der Plantbot nicht in der Lage seine elektrischen Funktionen zu nutzen, weshalb das Aufladen des Akkus die zweithöchste Priorität besitzt. Die Aktion wird ausgelöst, wenn der Akkustand des Plantbots unter 15 % fällt. Ist dies der Fall, fährt der Plantbot zu einer Person, falls sich eine in der Wohnung aufhält, macht auf sich aufmerksam, und fährt anschließend zu seinem Basisaufenthaltsort im Schlafzimmer. Dort wartet er, bis eine Person das Ladegerät ansteckt und sein Akku aufgeladen wird.

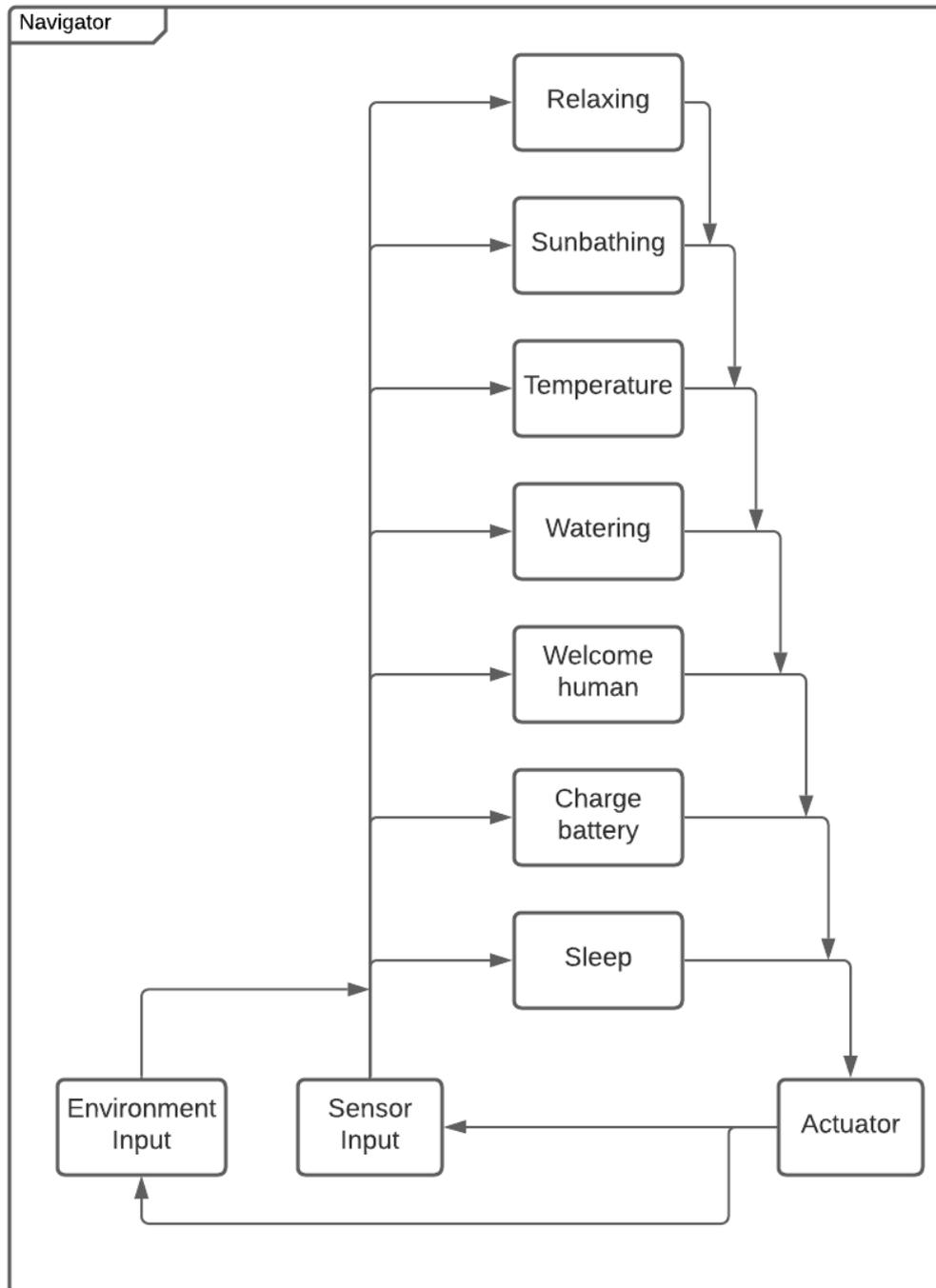


Abbildung 3.6: Subsumption-Architektur für die Navigator-Komponente. Die Priorisierung der Aktionen ist von unten nach oben gestaffelt.

Welcome Human

Die Aktion mit der dritten Priorität besitzt die Aktion *begrüßen von neuen Personen*, welche Ziel Nummer 05 (siehe 3.5) umsetzt. Wenn eine Person die Wohnung betritt, erfährt der Plantbot dies über die Kommunikation mit der Wohnung. Dann fährt die Pflanze zu dieser Person und schüttelt sich, um die Person willkommen zu heißen.

Watering

Als nächste Aktion wird die *Wässerung der Pflanze* festgelegt, was dem Ziel Nummer 01 (siehe 3.1) aus dem Konzept entspricht. Sobald der Mittelwert der beiden (Erd-) Feuchtigkeitsensoren unter 85 % fällt (siehe Tabelle 3.9), versucht der Plantbot ebenfalls eine Person auf sich aufmerksam zu machen und fährt daraufhin in die Dusche um sich gießen zu lassen. Dort wartet er bis er gegossen wird oder eine höher priorisierte Aktion ausgeführt werden soll.

Temperature

Der Aktion *Überwachen der Temperatur* wird die fünfte Priorisierung zugeteilt und ist auf Ziel Nummer 03 (siehe 3.3) zurückzuführen. Hierbei möchte der Plantbot eine Zimmertemperatur aufrechterhalten, die zwischen 18°C und 24°C liegt. Um die Temperatur innerhalb dieses Bereiches zu halten, hat der Plantbot die Möglichkeit, über die Komponente `environment_output` (siehe 3.3.3) mit der Wohnung zu kommunizieren und so Fenster zu schließen oder zu öffnen und Heizungen in der Wohnung zu regulieren.

Sunbathing

Zusätzlich zu der Bewässerung ist *Fotosynthese* essenziell für das Überleben der Pflanze. Deswegen muss der Plantbot täglich eine bestimmte Zeit in der Sonne verbringen, was dem Ziel Nummer 02 (siehe 3.2) entspricht. Für die Umsetzung dieser Aktion fährt der Plantbot alle Fenster ab, misst die Helligkeit an diesen Fenstern und stellt sich an das Fenster mit dem besten Helligkeitswert. Sie bleibt so lange an diesem Fenster stehen, bis die Sonnenstunden pro Tag erreicht sind oder eine Person über die Distanzsensoren mitteilt, dass der Plantbot an ein anderes Fenster fahren soll.

Tabelle 3.9: Sensor Werte für die Übergänge der Emotionen

	Feuchtigkeit	Temperatur in °C	Sonnenstunden/Tag	Batterie
Entspannt	$x > 85 \%$	$18 < x < 24$	80-100 %	30-100 %
Traurig	$x > 75 \%$	$16 < x \leq 18 \parallel$ $24 \leq x < 26$	50-80%	15-30 %
Wütend	$x \leq 75 \%$	$x \leq 16 \parallel x \geq 26$	0-50 %	0-15 %

Relaxing

Die Aktion mit der niedrigsten Priorisierung ist die *Relax* Aktion, welche dem Ziel Nummer 08 (siehe 3.8) entspricht. Sobald alle Lebenswichtigen und sozialen Aktionen abgeschlossen sind, stellt sich der Plantbot an einen beliebigen Ort in der Wohnung bis eine der höher priorisierten Aktionen ausgeführt wird oder eine Person über die Distanzsensoren mitteilt, dass der Plantbot an eine andere Stelle fahren soll.

Jede der aufgeführten Aktionen wird in einer separaten FSM-Klasse implementiert und über die Navigator-Klasse verwaltet.

3.3.2 Emotion Engine

Die Emotion Engine ist für das Verarbeiten der Emotionen des Plantbots zuständig. Dabei werden die gemessenen Sensorwerte auf die Emotionen entspannt, traurig und wütend gelegt. Folglich ist diese Komponente für das Ziel Nummer 06 (siehe 3.6) zuständig. Die relevanten Sensorwerte für diese Emotionen sind der Akkustand, der Feuchtigkeitswert der Erde, die Umgebungstemperatur und die Lichteinstrahlung. Der Wert für die Lichteinstrahlung wirkt sich allerdings nicht direkt auf die Emotion der Pflanze aus, sondern diese wird über die Komponente `light_duration` (siehe Abb. 3.5) gemessen. Diese misst, wie lange der Plantbot an einem Tag an einer hellen Stelle in der Wohnung stand und diese Information wird ebenfalls auf die Emotionen des Plantbots übertragen. Das Mapping der Sensorwerte auf die jeweiligen Emotionen ist in Tabelle 3.9 zu sehen.

Die drei Sensorwerte für den Akkustand, die Feuchtigkeit und für die Temperatur wirken sich dabei anders auf die Emotionen aus wie die Lichteinstrahlung. Sobald einer der Werte die Grenze zu einer negativeren Emotion überschreitet, wird ein zweistündiger Timer gestartet. Erst wenn dieser Timer abgelaufen ist, verändert sich die Emotion des Roboters. Falls während dieses Timers einer den anderen Sensoren die Grenze überschreitet,

wird der Timer um eine Stunde herabgesetzt und die Emotion wird somit schneller angenommen. Für die Verbesserung der Emotion müssen wieder alle Sensorwerte im Bereich darüber liegen, dann wird die verbesserte Emotion sofort übernommen.

Die Dauer, wie lange sich der Plantbot an einem Tag in der Sonne befunden hat, wirkt sich anders auf die Emotion der Pflanze aus. Der Plantbot überprüft um 0:00 Uhr wie viele Sonnenstunden in den vergangenen 24 Stunden registriert wurden. Davon abgeleitet ergibt sich die beste Emotion, die der Plantbot an dem darauf folgenden Tag erreichen kann. Sein Wohlempfinden kann nur wieder gesteigert werden, wenn der Plantbot genug Sonnenstunden an diesem Tag sammelt. Sobald er genug Sonnenstunden für die bessere Emotion erreicht hat, steigt seine Emotion auf diese an, wenn die Sensorwerte der anderen Sensoren mit dieser Emotion übereinstimmen.

Dieses Verhalten wird nach der FSM in Abb. 3.7 in der `emotion_engine`-Komponente umgesetzt.

Die Emotion der Pflanze soll dauerhaft über die angebrachte RGB-Leiste dargestellt werden und tritt eine Änderung der Emotion auf, wird das über die RGBs, über die Körperbewegung und über den Sound signalisiert. Wie sich das konkret äußert, ist in dem Kapitel 3.1.3 beschrieben.

Die Kommunikation zwischen der Emotion Engine Komponente und den Komponenten, die für die Ausdrücke zuständig sind, wird in 3.3.5 beschrieben.

3.3.3 ROS-Pakete

Durch die Benutzung von ROS wird die genutzte Prozessstruktur vorgegeben. Prozesse werden in ROS als „nodes“ bezeichnet und diese nodes werden in Paketen organisiert. Pakete, die zusammen eine Funktionalität darstellen, können wiederum in „Metapakete“ organisiert werden. Neben nodes muss ein Paket Konfigurationsdateien⁸ bereitstellen, die dem integrierten `CMake`⁹ Build System die nötigen Informationen über dieses Paket zur Verfügung stellt. Außerdem kann ein Paket Bibliotheken enthalten, die nicht als eigenständiger Prozess in ROS laufen, sondern als Dienste für andere nodes bereitgestellt werden. Damit es den nodes möglich ist untereinander zu kommunizieren, müssen zusätzlich Nachrichten definiert werden, die über ein integriertes publish-subscriber System ausgetauscht werden können (siehe Kapitel 3.3.5).

⁸`CMakeLists.txt`, ROS-Wiki für `CMakeLists.txt`, <https://wiki.ros.org/catkin/CMakeLists.txt>. [07.06.2021]

⁹`CMake`, `CMake` Webseite, <https://cmake.org/>. [07.06.2021]

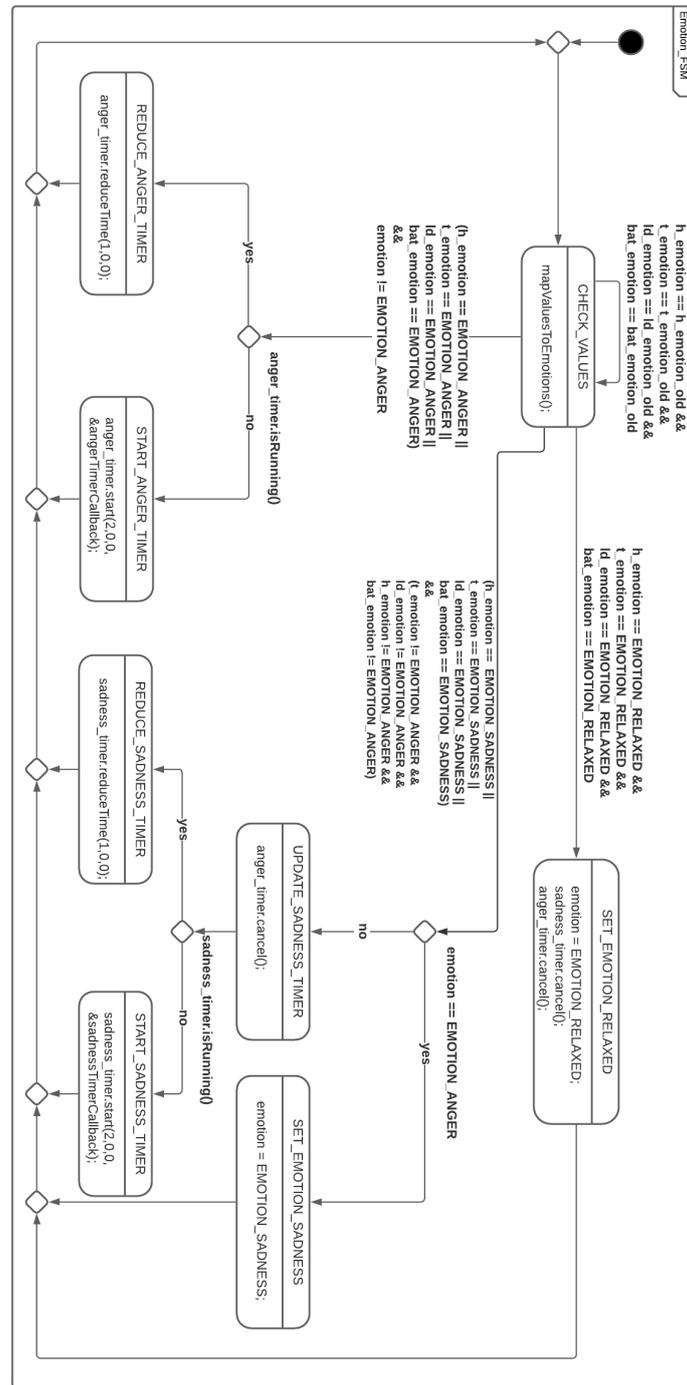


Abbildung 3.7: FSM für die Implementierung der Emotionen FSM

In dieser Thesis wird die Software in zwei Metapakete unterteilt, zum einen in das Metapaket, das auf dem Server vorzufinden ist (`plantbot_master`) und zum anderen das Metapaket, das auf dem Raspberry Pi des Plantbots vorzufinden ist (`plantbot_slave`). Die Metapakete enthalten beide die identischen Pakete `plantbot_msgs`, `common_information` und `plantbot_bringup`. Das Paket für die Nachrichten wird in beiden Umgebungen benötigt, da alle nodes alle Arten von Nachrichten miteinander austauschen können. Das Paket `common_information` wird als Bibliothek für alle nodes angeboten und enthält die Definition von Konstanten, die für unterschiedliche Komponenten relevant sind. In dem Paket `plantbot_bringup` sind die notwendigen `launch` Dateien zum Starten der nodes in dieser Umgebung abgelegt. Folglich werden diese Bibliotheken auf allen Umgebungen benötigt, auf denen die Plantbot nodes laufen werden.

Metapaket `plantbot_master`

Der Server besitzt mehr Rechenleistung und anderweitige Ressourcen als der Raspberry Pi, wodurch alle nodes, die nicht zwingend auf dem Raspberry Pi laufen müssen, auf dem Server ausgeführt werden. Alle benötigten nodes ergeben sich aus den in Abb. 3.5 dargestellten Komponenten. Jede Komponente wird durch eine node umgesetzt. Bis auf die Komponenten `arduino_reader` und `arduino_writer`, die über den I²C-Bus mit dem Arduino kommunizieren und deshalb auf dem Raspberry Pi laufen müssen, können alle Komponenten auf dem Server ausgeführt werden.

Im Folgenden werden die nodes auf die Pakete aufgeteilt und ihre Aufgaben beschrieben, beginnend mit den Paketen für das `plantbot_master` Metapaket.

Die Pakete `plantbot_navigator` und `plantbot_emotion_engine` sind jeweils für die Hauptkomponenten der Softwarearchitektur zuständig. Ihre Funktionsweisen sind in den Kapiteln 3.3.2 und 3.3.1 beschrieben.

Das Paket `plantbot_expressions` ist für die Verarbeitung der Ausdrücke der Pflanze zuständig. Dazu zählen die Komponenten `sound`, `body_movement` und `light_control`. Diese Komponenten bekommen jeweils die aktuelle Emotion des Plantbots und die Art des nächsten Ausdrucks mitgeteilt und reagieren entsprechend darauf (siehe 3.1.3).

Das Paket `plantbot_environment` enthält die nodes `environment_input` und `environment_output`. Diese nodes sind für die Kommunikation mit der Wohnung zu-

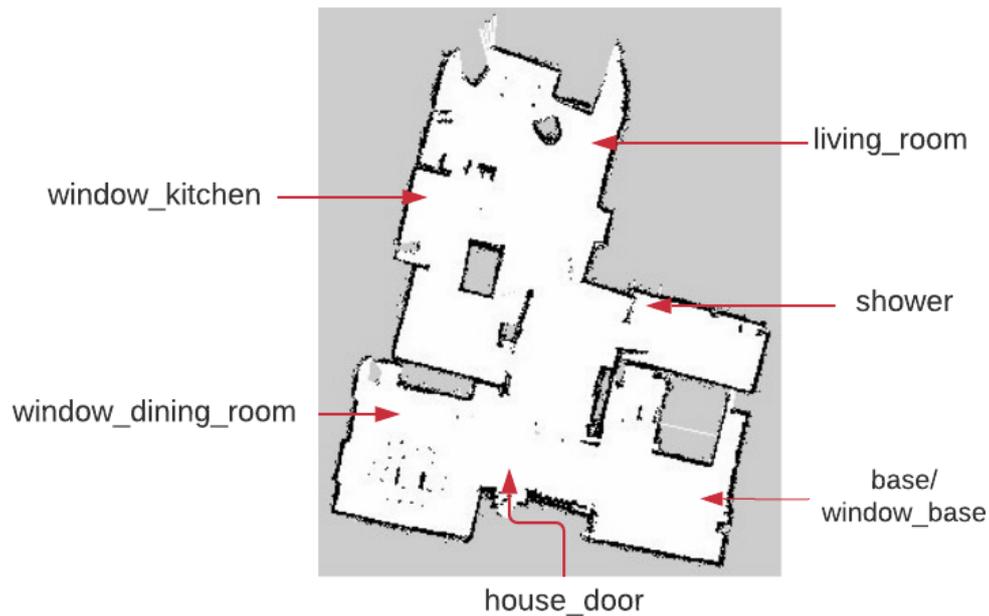


Abbildung 3.8: Karte der Living Place Wohnung aufgenommen mit dem LDS des Turtlebot Aufbaus.

ständig. Dafür nutzen sie die von der Living Place angebotene API, um mit den Fenstern und mit der Heizung in der Wohnung zu kommunizieren.

Für die Bewegungen der Pflanze ist das Paket **plantbot_movement** zuständig. Dieses beinhaltet die Komponente `navigation_control`, welche für die Navigation in der Wohnung verantwortlich ist. Die Koordinaten, die für die Navigation erforderlich sind, sind für die benötigten Punkte in der Wohnung abgespeichert. Diese stehen über den sogenannten `goal_manager`, eine Bibliothek in **plantbot_libs**, zur Verfügung. Um die Navigation gewährleisten zu können, benötigt der Plantbot immer die gleiche Position beim Starten des Systems. Diese wird auf den Basisaufenthaltsort im Schlafzimmer der Wohnung festgelegt. Die aufgezeichnete Karte des Living Places, mithilfe von SLAM Algorithmen, ist mit den benötigten Punkten in Abb. 3.8 dargestellt.

Die Konvertierung der Sensorwerte findet in dem Paket **plantbot_sensors** statt. Hierfür werden die Komponenten `light_duration` und `battery_converter` implementiert. Die `light_duration` Komponente übernimmt die Messung, wie lange der Plantbot an einem Tag in einem hellen Bereich in der Wohnung verbracht hat. Dieser Wert

setzt sich täglich um 00:00 Uhr in der Nacht zurück. Die `battery_converter` Komponente ist für die Konvertierung des Batteriewertes zuständig. Das ist notwendig, da die bereitgestellten Turtlebot nodes den Akkustand in dem Format 1.00 bis 2.00, entspricht 0 % - 100 %, an das System übertragen. Um einfacher damit arbeiten zu können, wird dieses Format in Werte zwischen 0 und 100 übertragen. Außerdem teilt er dem ROS-System mit, ob sich der Akku gerade entlädt oder er gerade aufgeladen wird.

Das Metapaket **plantbot_libs** beinhaltet die Bibliotheken, die von den Plantbot nodes benötigt werden. Diese bestehen aus den Bibliotheken `common_information`, `goal_manager`, `navigator_fsm`, `locking_queue`, `midnight_checker`, `rapidjson` und `timer`.

Nachfolgend werden die Aufgaben der Bibliotheken erläutert:

- `common_information`: Enthält Definitionen von Konstanten die von mehreren nodes benötigt werden.
- `goal_manager`: Verwaltet die Koordinaten der wichtigsten Punkte der Wohnung. Die Koordinaten werden in einer JSON-Datei abgespeichert.
- `navigator_fsm`: Beinhaltet alle benötigten Aktionen des Navigators als FSM implementiert.
- `locking_queue`: C++ Implementierung einer Queue, die bei einem Zugriff auf eine leere Queue so lange blockiert, bis ein anderer Prozess etwas auf der Queue ablegt.
- `midnight_checker`: Ruft jeden Tag bei Mitternacht eine gegebene *Callback*-Methode auf.
- `rapidjson`: Ein JSON parser/generator für C++¹⁰.
- `timer`: Nach Ablauf des Timers wird eine *Callback*-Methode aufgerufen.

Metapaket **plantbot_slave**

Auf dem `plantbot_slave` werden lediglich die Pakete **plantbot_arduino** und **plantbot_libs** benötigt. Das `plantbot_arduino` Paket beinhaltet die zwei Komponenten `arduino_writer` und `arduino_reader`. Beide greifen auf den I²C-Bus zwischen dem

¹⁰RapidJson, RapidJSON Webseite, <https://rapidjson.org/>. [07.06.2021]

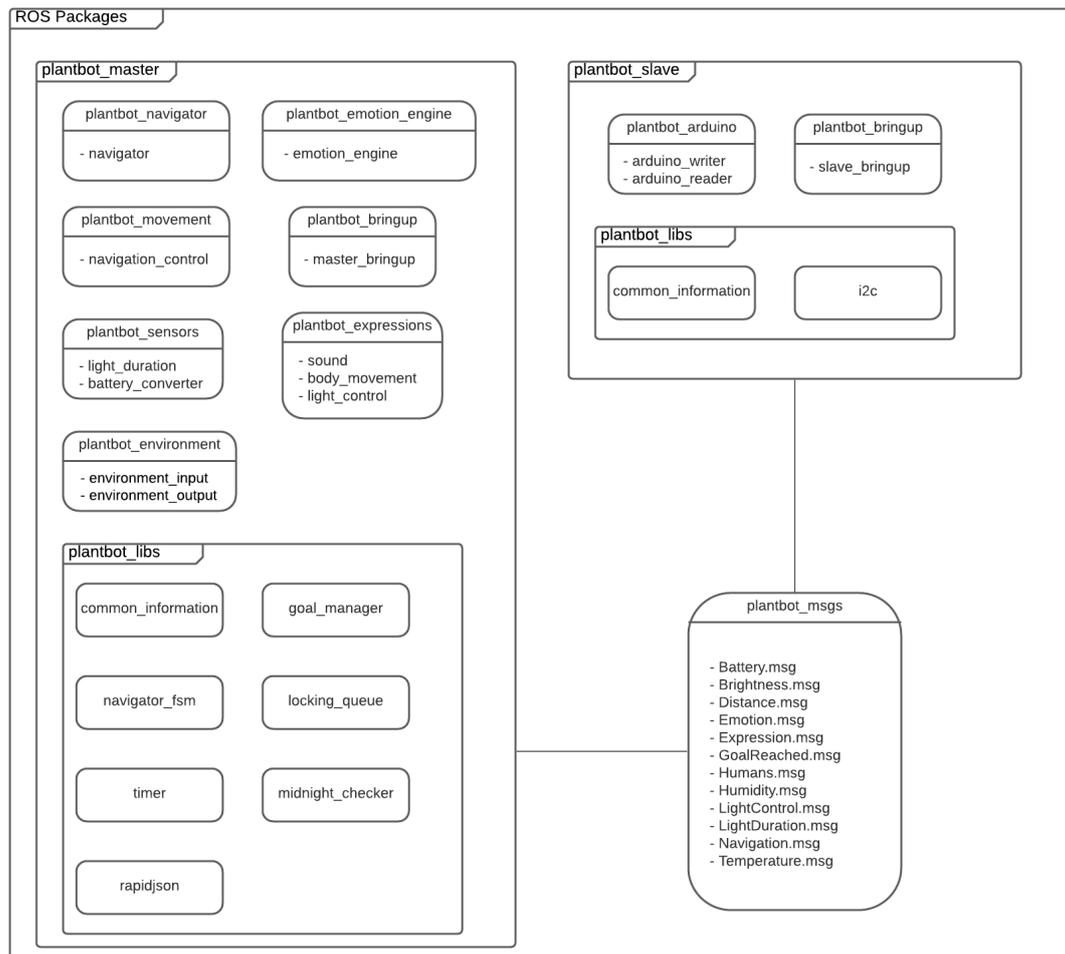


Abbildung 3.9: ROS-nodes in den unterteilten Pakete

Arduino und dem Raspberry Pi zu und tauschen so die Daten zu den gemessenen Sensor Werten und die Befehle für die RGB-Leiste aus (siehe 3.3.5).

Das Metapaket `plantbot_libs` enthält ebenfalls die Bibliothek `common_information` und darüber hinaus eine I²C-Bibliothek, die den Datenaustausch über den Bus abstrahiert.

Die verwendeten Nachrichten, welche für den Datenaustausch zwischen den nodes benötigt werden, befinden sich in dem Paket **plantbot_msgs**. Dieses Paket ist sowohl in dem `plantbot_master`, als auch in dem `plantbot_slave` Metapaket identisch vorhanden.

Die aufgezählten Pakete werden mit ihren nodes in Abb. 3.9 dargestellt.

3.3.4 Externe ROS-Pakete

Zusätzlich zu den selber erstellten Paketen werden externe Pakete genutzt, die für den Turtlebot bereitgestellt werden und somit nützliche Funktionen bereits mitbringen¹¹.

Die Firma ROBOTIS stellt für den Turtlebot ein Metapaket zur Verfügung, das die Grundlegende Ansteuerung zwischen dem Raspberry Pi, dem LDS und der OpenCR Platine übernimmt. Hierfür werden folgende Pakete genutzt.

Das **turtlebot3_slam** Paket bietet verschiedene Implementierungen des SLAM Ansatzes an [31]. Mithilfe dieser Algorithmen lässt sich eine Karte der Umgebung erstellen, die später benötigt wird, um den Turtlebot Roboter selbständig von Punkt zu Punkt fahren zu lassen.

Für die Bereitstellung der gemessenen Daten des Turtlebots wird das Paket **turtlebot3_bringup** genutzt. Beim Starten der *turtlebot3_robot.launch* Datei innerhalb dieses Paketes, werden die relevanten nodes ausgeführt und somit die Turtlebot Daten über die topics an das ROS-System bereitgestellt.

3.3.5 Kommunikation

Die Kommunikation zwischen den einzelnen Komponenten lässt sich in die Kommunikation zwischen den ROS-nodes und die Kommunikation mit Elementen außerhalb des ROS-Systems untergliedern. Beim Kommunizieren innerhalb des ROS-Systems werden Nachrichten über topics übertragen. Welche topics für welche Nachrichten in dieser Thesis benötigt werden und welche nodes dabei miteinander Kommunizieren wird im nachfolgenden Kapitel behandelt. Die Kommunikation mit Elementen außerhalb des ROS-Systems wird in dem darauf folgenden Kapitel behandelt.

Kommunikation der ROS-nodes untereinander

In ROS ist ein publish-subscriber System implementiert, das genutzt wird, um Nachrichten zwischen den verschiedenen Prozessen auszutauschen. Durch den Einsatz einer Middleware ist es dem ROS-System möglich, diese Nachrichten über das Netzwerk zu verschicken. So müssen keine weiteren Vorkehrungen getroffen werden, um den Raspberry Pi und den Server des Living Places miteinander kommunizieren zu lassen. Damit das

¹¹ROBOTIS, GitHub repository für das turtlebot3 Metapaket, <https://github.com/ROBOTIS-GIT/turtlebot3>. [07.06.2021]

Tabelle 3.10: Die verwendeten Topics mit den zugehörigen messages

Topic	Message
battery	Battery.msg
brightness	Brightness.msg
distance	Distance.msg
emotion	Emotion.msg
expression	Expression.msg
goal_reached	GoalReached.msg
humans	Humans.msg
humidity	Humidity.msg
light_control	LightControl.msg
light_duration	LightDuration.msg
navigation	Navigation.msg
temperature	Temperature.msg
ROS topics	
battery_state	BatteryState.msg
initialpose	PoseWithCovarianceStamped.msg
cmd_vel	Twist.msg

funktioniert existiert in der ROS-Umgebung eine Master node, bei der sich alle anderen nodes beim Starten anmelden müssen. An diese Master node teilt jeder Teilnehmer mit, an welchen topics er interessiert ist und über welche topics er Nachrichten verschickt. In Tabelle 3.10 sind alle verwendeten topics mit den dazugehörigen Nachrichten aufgeführt. Neben den selber erstellten topics und Nachrichten werden topics, die für den Turtlebot angeboten werden, benötigt, um alle Funktionalitäten zu implementieren.

Nachfolgend werden die Aufgaben von jedem verwendeten topic erläutert:

- `battery`: Verbreitet den aktuellen Akkustand des Plantbots.
- `brightness`: Verbreitet den vom Arduino gemessenen Helligkeitswert.
- `distance`: Verbreitet die Werte der beiden Distanzsensoren des Plantbots.
- `emotion`: Verbreitet die aktuelle Emotion des Plantbots.
- `expression`: Verbreitet, wie sich der Plantbot als nächstes ausdrücken soll.
- `goal_reached`: Verbreitet, ob der Plantbot ein Navigationsziel erreicht hat oder nicht.

- `humans`: Verbreitet, welche Menschen sich wo aufhalten.
- `humidity`: Verbreitet den gemittelten Feuchtigkeitswert des Arduinos.
- `light_duration`: Verbreitet die Dauer in Minuten, wie lange der Plantbot sich an diesem Tag bereits in der Sonne befand.
- `navigation`: Verbreitet das nächste Navigationsziel des Plantbots.
- `temperature`: Verbreitet die Umgebungstemperatur um den Plantbot.

Neben diesen selber erstellten topics werden folgende bereitgestellte topics verwendet. Über das topic `battery_state` werden aktuelle Informationen in Bezug auf den Akku des Plantbots geteilt. Wichtig für diese Thesis ist dabei der Akkustand. Das topic `initialpose` wird benötigt, um der Navigationskomponente des Turtlebots mitzuteilen, wo sich der Turtlebot beim Starten befindet. Und das topic `cmd_vel` wird benötigt, um die Motoren des Plantbots unabhängig von einer Navigation anzusteuern und somit die Ausdrücke über die Körperbewegung realisieren zu können.

Mithilfe dieser topics und Nachrichten ist es den nodes möglich, Nachrichten und somit alle aktuellen Informationen auszutauschen. Die Verbindungen zwischen den Komponenten mit Betrachtung auf die topics ist in Abb. 3.10 dargestellt.

Kommunikation mit Elementen außerhalb des ROS-Systems

Die Kommunikation mit Elementen außerhalb des ROS-Systems beschränkt sich auf die Kommunikation zwischen dem Arduino und dem Raspberry Pi. Diese beiden Komponenten kommunizieren über einen I²C-Bus miteinander, wobei der Raspberry Pi den Master und der Arduino den Slave darstellt. Die Kommunikation ist aufgeteilt in Sensorwerte von dem Arduino auslesen und in Befehle für die RGB-Leiste an den Arduino schicken. Um die Sensorwerte des Arduinos auszulesen, muss der Raspberry Pi, da er der Master des Busses ist, die Werte konkret anfordern. Dafür wird beim Schreiben ein Header benutzt, der dem Arduino signalisiert, ob ein Wert ausgelesen werden soll oder ob Werte für die RGB-Kette geschrieben werden. Falls ein Sensorwert ausgelesen werden soll, schickt der Raspberry Pi zuerst das Headerbyte und dann das Sensorbyte, welches den Sensor, der ausgelesen werden soll, beinhaltet. Daraufhin antwortet der Arduino mit dem aktuellen Wert des Sensors, der in zwei Bytes verschickt wird.

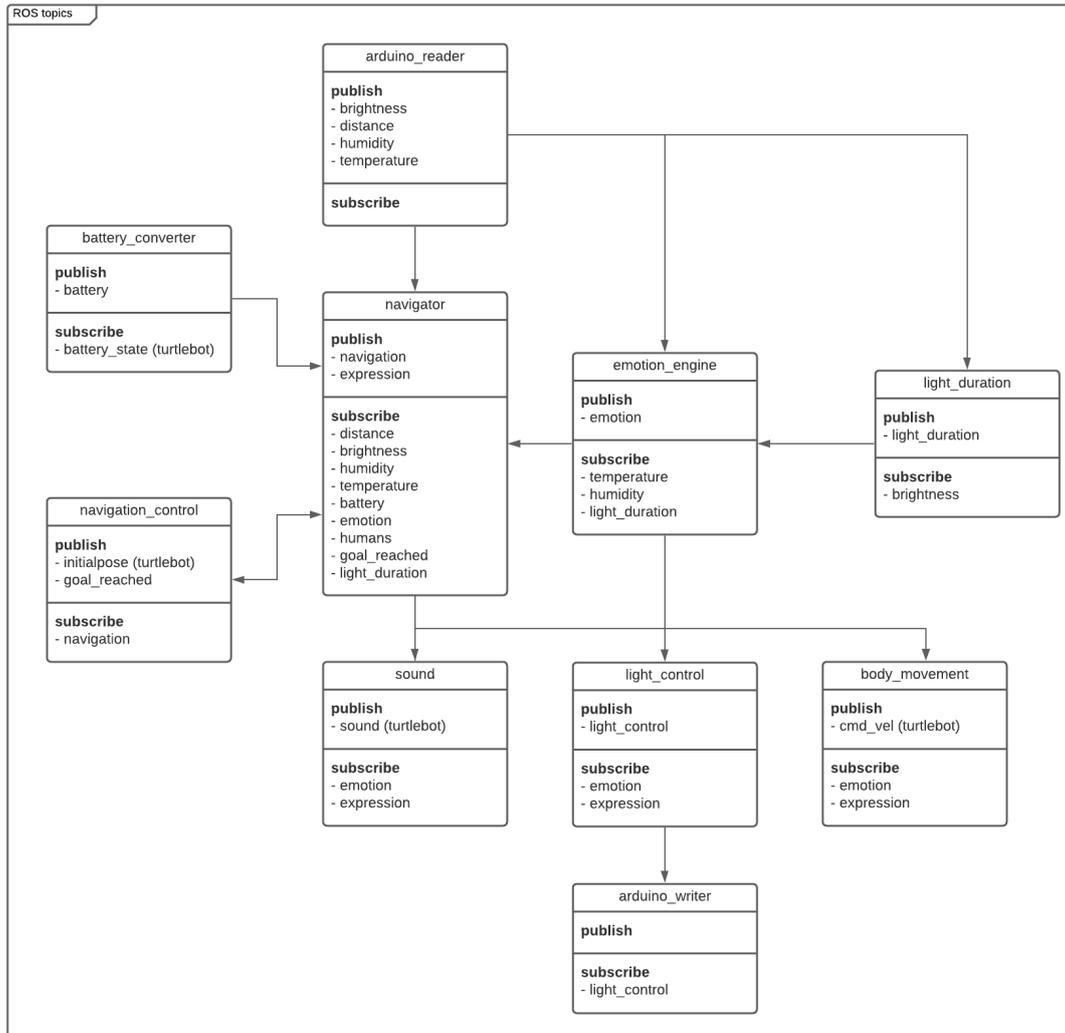


Abbildung 3.10: Die Komponenten mit dessen Nachrichtenaustausch.

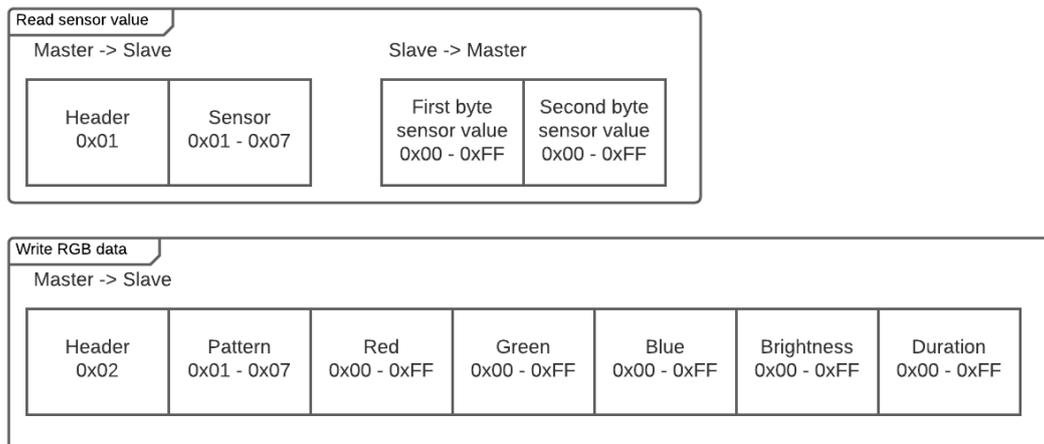


Abbildung 3.11: Nachrichtenaustausch über den I²C-Bus.

Wenn die Werte für die RGB-Kette geschrieben werden sollen, wird wieder das zugehörige Headerbyte verschickt, dann das Byte, das das Muster definiert, dann die drei Bytes für die Farben Rot, Grün und Blau, dann das Byte für die Helligkeit der RGBs und am Schluss das Byte für die Geschwindigkeit des Musters. Die Anordnung und die Werte, die die Bytes annehmen können, sind zusätzlich in Abb. 3.11 dargestellt.

3.4 Evaluation

Dieses Kapitel soll die Ziele aus dem Konzept Kapitel 3.1 wieder aufgreifen und betrachten, inwieweit diese umgesetzt werden konnten. Dazu wird zunächst auf die Ziele des physischen Aufbaus eingegangen und wie diese umgesetzt werden konnten und daraufhin wird das Verhalten und die Ziele die damit verbunden sind betrachtet.

3.4.1 Physischer Aufbau

Für den physischen Aufbau ist die Zielsetzung dieser Thesis, dass sich der Plantbot frei in der Wohnung bewegen kann und somit seine Umgebung wahrnehmen kann, sich um seine Bedürfnisse selbständig kümmern kann und seine Emotionen über Aktoren ausdrücken kann. Diese Ziele sind aufseiten des physischen Aufbaus komplett umgesetzt. Denn durch den Turtlebot3 Waffle Pi Bausatz ist der Plantbot in der Lage, sich mithilfe der darin

integrierten Motoren und des LDS in der Wohnung frei zu bewegen und seine Umgebung wahrzunehmen. Zusätzlich zu dem LDS wurden zur Wahrnehmung der Umgebung zwei Distanzsensoren angebracht, um kein Objekt zu übersehen und um eine Möglichkeit zu schaffen, dem Plantbot Signale zu geben.

Um seine Bedürfnisse zu stillen, ist der Plantbot mit der notwendigen Sensorik ausgestattet worden. Dazu zählen die zwei Feuchtigkeitssensoren, der Helligkeitssensor und der Temperatursensor. So kann er wahrnehmen, wenn er etwas benötigt und ob das Verlangen nach einem Bedürfnis gestillt wurde.

Die Emotionen kann der Plantbot über die Motoren mit einer Körperbewegung, über die RGB-Leiste mit Farben und über das verbaute Soundmodul über Ton ausdrücken.

Lediglich bei der Navigation über den LDS und beim Ausdrücken der Emotion über die Körperbewegung und über Sound kam es zu Problemen. Über den LDS musste eine zusätzliche Plattform für die Pflanze angebracht werden. Durch die zusätzlichen Halterungen der dritten Plattform und die Kabel, die von der Sensorik und Aktorik zur Ebene eins geleitet werden mussten, wurden die Ergebnisse des LDS verfälscht und die Objekterkennung lief nicht reibungslos. Dies führte zur Erkennung von nicht vorhandenen Objekten und dementsprechend konnte die Navigation nicht problemlos ausgeführt werden. Durch Verwendung des Open-Source Codes des LDS Treibers¹², konnte das Problem behoben werden, indem die minimale Distanz zum Erkennen von Objekten des LDS, in der Datei `src/hlds_laser_publisher.cpp` von „0.12“ auf „0.3“ erhöht wurde. Mit dieser Modifikation funktioniert die Navigation durch die Wohnung reibungslos.

Beim Ausdrücken der Emotionen über die Körperbewegung und über den Sound kam es auch zu Problemen. Die Motoren und das Soundmodul werden über die OpenCR Platine des Plantbots angesteuert. Diese Platine ist mit einem Einkernprozessor ausgestattet, weshalb es nicht möglich ist, echt parallele Aktionen auszuführen. Durch diese fehlende Parallelität ist es nicht möglich, die Körperbewegung und das Abspielen einer Tonfolge parallel ablaufen zu lassen. Das Problem wurde gelöst, indem die Tonfolge vor der Körperbewegung abgespielt wird, um eine Überschneidung zu verhindern.

3.4.2 Verhalten und Ziele des Plantbots

Bei der Evaluation zum Verhalten und der Ziele des Plantbots sind neben den realen Sensordaten simulierte Eingabewerte zum Einsatz gekommen, um die Vollständigkeit

¹²LDS Treiber, Repository des LDS Treibers, https://github.com/ROBOTIS-GIT/hls_lfcd_lds_drivers.de/. [07.06.2021]

der genutzten Navigator FSMs und der Emotion Engine zu testen.

Der Einsatz des Navigators mit der Subsumption-Architektur und den verschiedenen Aktionen der Pflanze ist umgesetzt worden und ist funktionsfähig. Auch die ROS-Komponenten (siehe 3.3.3) sind mit ihren Aufgaben funktionsfähig umgesetzt. Die Versorgung mit Wasser und mit Strom kann allerdings nicht komplett durch den Plantbot übernommen werden. Für diese Aktionen sind Personen in der Wohnung notwendig, die den Plantbot in der Dusche gießen und die den Plantbot an seinem Basisaufenthaltort an das Ladekabel anschließen.

Die Kommunikation mit der Wohnung ist außerdem nur innerhalb des ROS-Systems realisiert. Dies liegt an der fehlenden Umsetzung der Funktion Personen in der Wohnung lokalisieren zu können und der Möglichkeit, Fenster zu steuern. Dies wurde für diese Thesis als gegeben von der Wohnung angesehen. Lediglich die Steuerung der Heizungen wäre über einen MQTT-Broker möglich.

Dadurch ist das Ziel der Einhaltung einer angenehmen Temperatur und das Begrüßen von neuen Personen nur theoretisch möglich und wird mit simulierten Daten realisiert.

4 Fazit und Ausblick

In diesem Kapitel wird das Fazit und der mögliche Ausblick aufbauend auf dieser Thesis behandelt.

4.1 Fazit

Das Ziel dieser Thesis war es, eine Pflanze zu modifizieren, dass sie in der Lage ist ein Umfeld belebter und sozialer zu gestalten.

Um dieses Ziel zu erreichen, mussten zunächst die wissenschaftlichen Rahmenbedingungen untersucht werden, um den Plantbot wissenschaftlich einordnen zu können. Hierzu wurde sich mit eigenständigen Einheiten, sogenannten Agenten, beschäftigt und wichtige Umgebungsmerkmale für das Verhalten eines Agenten aufgezeigt. Über diese Agenten-Metapher wurden daraufhin die Themengebiete der sozialen Robotik, von künstlichen Tieren und von smarten Objekten untersucht. Durch diese Betrachtungen ergaben sich die wesentlichen Merkmale der drei Themengebiete und der Plantbot konnte zwischen diesen Themen eingeordnet werden.

Eine weitere wichtige Untersuchung war das Ausdrücken von Affekten bei Robotern, die ein eingeschränktes Erscheinungsbild besitzen. Wie sich herausgestellt hat, werden Roboter eher als Interaktionspartner akzeptiert, wenn sie ein Erscheinungsbild besitzen, das uns Menschen vertraut ist. Weist ein Roboter keine Möglichkeiten auf, Affekte zu kommunizieren. z. B. wegen eines fehlenden Gesichts oder Gliedmaßen, sollten sie mittels anderer Funktionen die Möglichkeit haben, diese auszudrücken. Hierfür bieten sich Bewegungen, Tonfolgen oder Farben an. Mit diesen Erkenntnissen konnte ein Konzept für den Plantbot erarbeitet werden, das es ihm ermöglicht, sich frei in einer Wohnung zu bewegen, seine Bedürfnisse zu erkennen, sie zu stillen und soziale Interaktionen durchzuführen.

Für die Umsetzung des Konzepts wurde die Pflanze mit einer Roboterplattform ausgestattet, mit der sie in der Lage ist, sich selbständig durch die Wohnung zu bewegen.

Anschließend wurde die Pflanze und die Roboterplattform mit zusätzlicher Sensorik und Aktorik ausgestattet, um die Bedürfnisse der Pflanze messen und Ausdrücken zu können. Hierfür wurden zwei Feuchtigkeitssensoren, ein Temperatursensor, ein Helligkeitssensor und zwei Distanzsensoren verbaut. Damit ist der Plantbot in der Lage, die Feuchtigkeit der Erde, seine Umgebungstemperatur und seine Umgebungshelligkeit wahrzunehmen. Über die Distanzsensoren kann zusätzliche Sicherheit bei der Navigation gewährleistet werden. Zusätzlich dienen die Distanzsensoren als Möglichkeit der Kommunikation, wenn man sich z. B. vor den Plantbot stellt und ihm so ein Signal geben kann.

Neben der Roboterplattform wurde an Aktorik zusätzlich eine RGB-Leiste verbaut, um die Affekte des Plantbots ausdrücken zu können. Also ist der Plantbot in der Lage, sich über die Roboterplattform mit Tonfolgen und mit Körperbewegungen und über die RGB-Leiste mit Farben und auch Farbmustern auszudrücken.

Nach dem technischen Aufbau konnte das definierte Verhalten des Plantbots in Form von Software umgesetzt werden. Als Basis für die Implementierung kam dabei das *Robot Operating System* zum Einsatz. Dieses ermöglichte es, die notwendigen Prozesse für die Umsetzung des Verhaltens auf mehreren Geräten laufen zu lassen und durch die Bereitstellung von Gerätetreibern konnte Hardware, wie der Laser Distanz Sensor, sofort genutzt werden.

Für die Implementierung wurden zwei Hauptkomponenten definiert, nämlich der Navigator und die Emotion Engine. Der Navigator enthält das komplette Verhalten des Plantbots, indem er jede Aktion die der Plantbot tätigen kann in einer FSM vorliegen hat und über Priorisierung entscheiden kann, welche Aktion zu einem Zeitpunkt ausgeführt werden soll.

Mit der Emotion Engine wird das Wohlbefinden des Roboters simuliert. Das Wohlempfinden wird dabei aus den Bedürfnissen der Wasserversorgung, der Versorgung mit Licht, einer angenehmen Umgebungstemperatur und über den Akkustand des Roboters zusammengesetzt. Diese Faktoren entscheiden, ob der Plantbot die Emotion entspannt, traurig oder wütend annimmt. Bei einer Änderung der Emotion drückt der Roboter das über die Körperbewegung, einer Tonfolge und mit einer zugehörigen Farbe und einem Farbmuster aus. Nach Abschluss dieser Sequenz wird die Emotion dauerhaft nur über eine Farbe und ein Farbmuster ausgedrückt.

Auf technischer Seite konnte nur die Parallelität der Körperbewegung und der Tonfolge während eines Emotionswechsels nicht gewährleistet werden. Ansonsten ist die technische Ausstattung des Plantbots voll funktionsfähig.

Bei der Software Implementierung konnte die Umsetzung der Einhaltung einer ange-

nehmen Temperatur nicht eingehalten werden, da die Umsetzung einer funktionierenden Fenstersteuerung in der Smart Home Wohnung fehlt. Auch das Begrüßen einer neu eingetretenen Person konnte nur simuliert werden, da die Wohnung zu dem jetzigen Zeitpunkt keine Personen in der Wohnung erkennen und lokalisieren kann.

Nichtsdestotrotz ist die Umsetzung des Plantbots gelungen und die Funktionen zum Erkennen und Versorgen der Bedürfnisse konnten erfolgreich umgesetzt werden.

4.2 Ausblick

Aufbauend auf diese Thesis können noch Funktionalitäten verbessert oder hinzugefügt werden. Dieses Kapitel liefert einen Ausblick darauf, was noch möglich ist.

Die fehlende Kommunikation mit der Wohnung und damit einhergehend die fehlenden Funktionalitäten der Wohnung, die Fenster zu öffnen und Personen in der Wohnung zu erkennen, könnten noch umgesetzt werden. Auch die selbständige Versorgung von Wasser und Strom, durch eine Bewässerungsanlage in der Dusche und einer Aufladestation, würde die bestehende Arbeit abrunden.

Neben diesen physischen Funktionalitäten kann die soziale Betrachtung des Plantbots eine größere Rolle einnehmen. Bei der Versorgung mit Wasser zum Beispiel benötigt der Plantbot Zugang zum Badezimmer. In diesem Kontext kann die Streitkultur zwischen einer Person, die ins Bad möchte und dem Plantbot, der ins Bad möchte, untersucht werden. Folglich müsste die Kommunikation mit der Pflanze verbessert werden. Dies könnte über Berührungen der Blätter, besprühen der Blätter mit Wasser oder über Spracherkennung geschehen. Auf Basis dieser Kommunikationsmöglichkeiten könnte der Plantbot Persönlichkeitsprofile von seinen Interaktionspartnern erstellen. Diese würden dann Auskunft darüber geben, welche Personen ein gutes Verhältnis zu der Pflanze besitzen und wer nicht so gut mit ihr auskommt. Dementsprechend könnte der Plantbot seine Umgangsform mit den Personen ins Positive oder ins Negative anpassen.

Auf der technischen Seite des Plantbots können ebenfalls Verbesserungen vorgenommen werden. Die Sensorik zur Wahrnehmung der Feuchtigkeit und der Helligkeit funktionieren zwar, sind aber nicht besonders genau. Hier könnten genauere Sensoren verbaut werden und bei der Helligkeit könnte außerdem näher auf die Bedürfnisse einer Pflanze eingegangen werden. Konkret gemeint ist das Lichtspektrum, welches die Pflanze benötigt, um eine ideale Fotosynthese durchführen zu können. Somit könnten das Lichtspektrum und

die Lichtintensität genauer gemessen werden, um dem Bedürfnis der Pflanze gerechter zu werden. In diesem Kontext sollte ebenfalls die Konfigurationsmöglichkeiten des Plantbots verbessert werden, um andere Pflanzenarten, die unterschiedlich viel Sonne und Wasser benötigen, zu unterstützen.

Die Motivation zum Aufbauen dieses Roboters war, eine sozialere und aktivere Umgebung zu schaffen. Dies soll bedürftigen Personen, in Pflegeheimen oder anderen Umgebungen, in denen soziale Aktivitäten mangelhaft vorhanden sind, helfen. Um dies wissenschaftlich zu belegen, kann in einer Studie, die auf dieser Arbeit aufbaut, der Einfluss auf die Umgebung durch den Plantbot untersucht werden und was für Auswirkungen er auf die soziale Intelligenz von Personen innerhalb dieser Umgebung haben kann.

Literaturverzeichnis

- [1] Robbe Paro. Nov 2009. – URL <http://www.parorobots.com>. – [Online; accessed 8. Mar. 2021]
- [2] *Living Place* |. Dec 2020. – URL <https://livingplace.haw-hamburg.de>. – [Online; accessed 30. Dec. 2020]
- [3] ARGYLE, M.: *Social Interaction*. Aldine Transaction, 2007. – ISBN 9780202309125
- [4] ARGYLE, M.: *Bodily Communication*. Taylor & Francis, 2013. – ISBN 9781134964253
- [5] BALKWILL, Laura-Lee ; THOMPSON, William F. ; MATSUNAGA, RIE: Recognition of emotion in Japanese, Western, and Hindustani music by Japanese listeners 1. In: *Japanese Psychological Research* 46 (2004), Nr. 4, S. 337–349
- [6] BARTNECK, Christoph: Affective Expressions of Machines. In: *CHI '01 Extended Abstracts on Human Factors in Computing Systems*. New York, NY, USA : Association for Computing Machinery, 2001 (CHI EA '01), S. 189–190. – URL <https://doi.org/10.1145/634067.634181>. – ISBN 1581133405
- [7] BETHEL, Cindy ; MURPHY, Robin: Survey of Non-facial/Non-verbal Affective Expressions for Appearance-Constrained Robots. In: *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on* 38 (2008), 02, S. 83 – 92
- [8] BETHEL, Cindy L. ; MURPHY, Robin R.: Affective expression in appearance constrained robots. In: *Proceedings of the 1st ACM SIGCHI/SIGART conference on Human-robot interaction*, 2006, S. 327–328
- [9] BREAZEAL, C.: Socially intelligent robots: research, development, and applications. In: *2001 IEEE International Conference on Systems, Man and Cybernetics. e-Systems and e-Man for Cybernetics in Cyberspace (Cat.No.01CH37236)* Bd. 4, 2001, S. 2121–2126 vol.4

- [10] BREAZEAL, Cynthia ; BROOKS, Rodney: Robot emotion: A functional perspective. In: *Who needs emotions* (2005), S. 271–310
- [11] BREWKA, Gerd: Artificial intelligence—a modern approach by Stuart Russell and Peter Norvig, Prentice Hall. Series in Artificial Intelligence, Englewood Cliffs, NJ. In: *The Knowledge Engineering Review* 11 (1996), Nr. 1, S. 78–79
- [12] CHWELOS, Paul D. ; BERNDT, Ernst R. ; COCKBURN, Iain M.: Faster, smaller, cheaper: an hedonic price analysis of PDAs. In: *Applied Economics* 40 (2008), Nr. 22, S. 2839–2856
- [13] DAUTENHAHN, Kerstin: Socially intelligent robots: dimensions of human-robot interaction. In: *Philosophical Transactions of the Royal Society B: Biological Sciences* 362 (2007), Nr. 1480, S. 679–704. – URL <https://royalsocietypublishing.org/doi/abs/10.1098/rstb.2006.2004>
- [14] FERBER, J.: *Multi-agent Systems: An Introduction to Distributed Artificial Intelligence*. Addison-Wesley, 1999. – ISBN 9780201360486
- [15] FONG, Terrence ; NOURBAKHSI, Illah ; DAUTENHAHN, Kerstin: A survey of socially interactive robots. In: *Robotics and Autonomous Systems* 42 (2003), Nr. 3, S. 143–166. – URL <https://www.sciencedirect.com/science/article/pii/S092188900200372X>. – Socially Interactive Robots. – ISSN 0921-8890
- [16] KLÖPPNER, Matthias ; KUCHENBUCH, Max ; SCHUMACHER, Lutz: *Fachkräftemangel im Pflegesektor*. Springer-Verlag, 2017
- [17] KORTUEM, Gerd ; KAWSAR, Fahim ; SUNDRAMOORTHY, Vasughi ; FITTON, Daniel: Smart objects as building blocks for the Internet of things. In: *IEEE Internet Computing* 14 (2010), Nr. 1, S. 44–51
- [18] LÓPEZ, Tomás S. ; RANASINGHE, Damith C. ; PATKAI, Bela ; MCFARLANE, Duncan: Taxonomy, technology and applications of smart objects. In: *Information Systems Frontiers* 13 (2011), Nr. 2, S. 281–300
- [19] MARTI, P. ; POLLINI, A. ; RULLO, A. ; SHIBATA, T.: Engaging with Artificial Pets. In: *Proceedings of the 2005 Annual Conference on European Association of Cognitive Ergonomics*, University of Athens, 2005 (EACE '05), S. 99–106. – ISBN 9602546565
- [20] MIKLÓSI, Ádám ; GÁCSI, Márta: On the Utilization of Social Animals as a Model for Social Robotics. In: *Frontiers in Psychology* 3 (2012), S. 75. – URL <https://>

- [//www.frontiersin.org/article/10.3389/fpsyg.2012.00075](http://www.frontiersin.org/article/10.3389/fpsyg.2012.00075). – ISSN 1664-1078
- [21] NIJDAM, Niels A.: Mapping emotion to color. In: *Book Mapping emotion to color* (2009)
- [22] ODENDAAL, J.S.J: Animal-assisted therapy — magic or medicine? In: *Journal of Psychosomatic Research* 49 (2000), Nr. 4, S. 275–280. – URL <https://www.sciencedirect.com/science/article/pii/S0022399900001835>. – ISSN 0022-3999
- [23] POLLOCK, C. J.: The response of plants to temperature change. In: *The Journal of Agricultural Science* 115 (1990), Nr. 1
- [24] RETTO, Jesús: Sophia, first citizen robot of the world. In: *ResearchGate*, URL: <https://www.researchgate.net> (2017)
- [25] RUSSELL, Stuart J.: *Artificial Intelligence: A Modern Approach*. Prentice Hall, feb 1995. – URL <https://www.xarg.org/ref/a/0131038052/>. – ISBN 0131038052
- [26] SALICHS, M. A. ; BARBER, R. ; KHAMIS, A. M. ; MALFAZ, M. ; GOROSTIZA, J. F. ; PACHECO, R. ; RIVAS, R. ; CORRALES, A. ; DELGADO, E. ; GARCIA, D.: Maggie: A Robotic Platform for Human-Robot Social Interaction. In: *2006 IEEE Conference on Robotics, Automation and Mechatronics*, 2006, S. 1–7
- [27] SCHERER, Klaus R.: Nonlinguistic vocal indicators of emotion and psychopathology. In: *Emotions in personality and psychopathology*. Springer, 1979, S. 493–529
- [28] SONG, Sichao ; YAMADA, Seiji: Expressing Emotions through Color, Sound, and Vibration with an Appearance-Constrained Social Robot. In: *Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction*. New York, NY, USA : Association for Computing Machinery, 2017 (HRI '17), S. 2–11. – URL <https://doi.org/10.1145/2909824.3020239>. – ISBN 9781450343367
- [29] SONG, Sichao ; YAMADA, Seiji: Bioluminescence-Inspired Human-Robot Interaction: Designing Expressive Lights That Affect Human’s Willingness to Interact with a Robot. In: *Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*. New York, NY, USA : Association for Computing Machinery, 2018 (HRI '18), S. 224–232. – URL <https://doi.org/10.1145/3171221.3171249>. – ISBN 9781450349536

- [30] WOOLDRIDGE, M.: *An Introduction to MultiAgent Systems*. Wiley, 2008. – ISBN 9780470353479
- [31] XUEXI, Z. ; GUOKUN, L. ; GENPING, F. ; DONGLIANG, X. ; SHILIU, L.: SLAM Algorithm Analysis of Mobile Robot Based on Lidar. In: *2019 Chinese Control Conference (CCC)*, 2019, S. 4739–4745

A Anhang

A.0.1 Schaltplan Lochrasterplatine

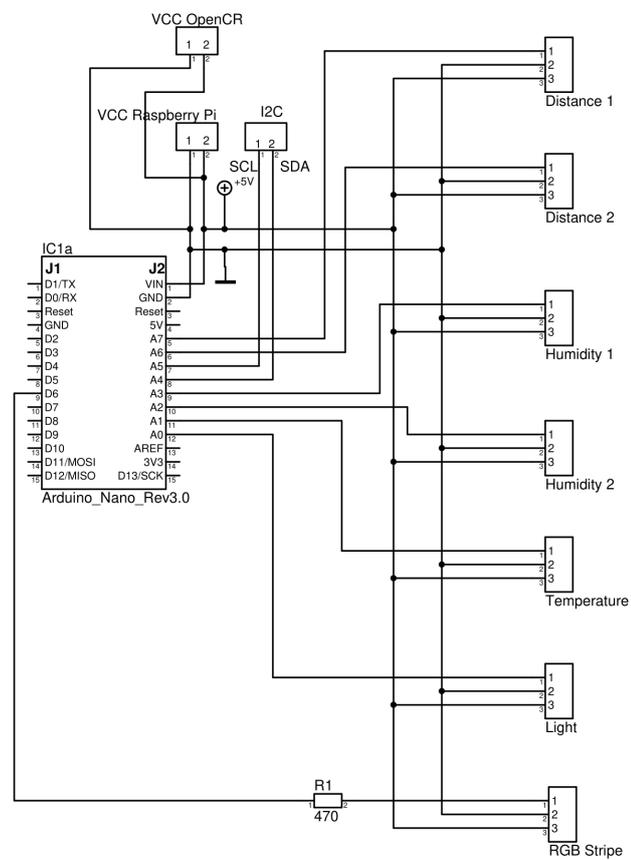


Abbildung A.1: Schaltplan Lochrasterplatine

Erklärung zur selbstständigen Bearbeitung einer Abschlussarbeit

Gemäß der Allgemeinen Prüfungs- und Studienordnung ist zusammen mit der Abschlussarbeit eine schriftliche Erklärung abzugeben, in der der Studierende bestätigt, dass die Abschlussarbeit „– bei einer Gruppenarbeit die entsprechend gekennzeichneten Teile der Arbeit [(§ 18 Abs. 1 APSO-TI-BM bzw. § 21 Abs. 1 APSO-INGI)] – ohne fremde Hilfe selbständig verfasst und nur die angegebenen Quellen und Hilfsmittel benutzt wurden. Wörtlich oder dem Sinn nach aus anderen Werken entnommene Stellen sind unter Angabe der Quellen kenntlich zu machen.“

Quelle: § 16 Abs. 5 APSO-TI-BM bzw. § 15 Abs. 6 APSO-INGI

Erklärung zur selbstständigen Bearbeitung der Arbeit

Hiermit versichere ich,

Name: _____

Vorname: _____

dass ich die vorliegende Bachelorarbeit – bzw. bei einer Gruppenarbeit die entsprechend gekennzeichneten Teile der Arbeit – mit dem Thema:

Konstruktion eines Artificial Pets in einer Smart Home Umgebung

ohne fremde Hilfe selbständig verfasst und nur die angegebenen Quellen und Hilfsmittel benutzt habe. Wörtlich oder dem Sinn nach aus anderen Werken entnommene Stellen sind unter Angabe der Quellen kenntlich gemacht.

Ort Datum Unterschrift im Original