# CuBi: A Decluttering Robot

## Conceptual Design Review

MRSD Team D

Stakeholders:

Customer – Cyert Center for Early Education

Mentor – Dr. David Held

Professors:

Dr. Dimitrios (Dimi) Apostolopoulos

Dr. John M. Dolan


The Team:

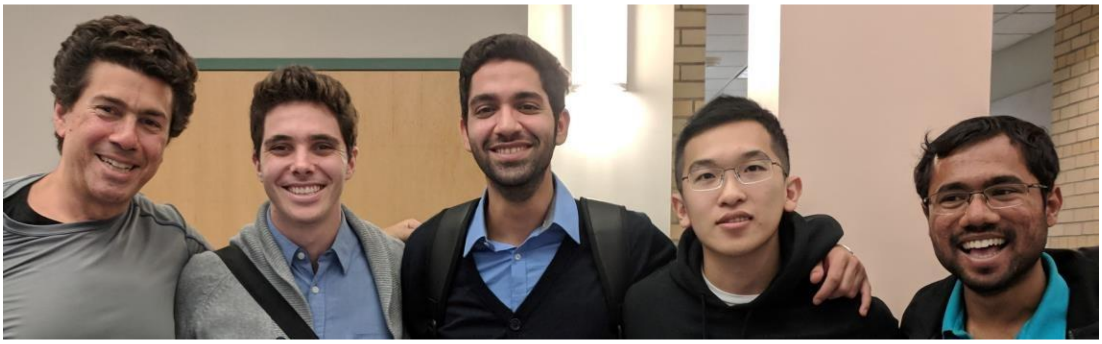
Jorge Anton

Laavanye Bahl

Paulo Camasmie

Changsheng Shen (Bobby)

Nithin Subbiah Meganathan


Carnegie Mellon University

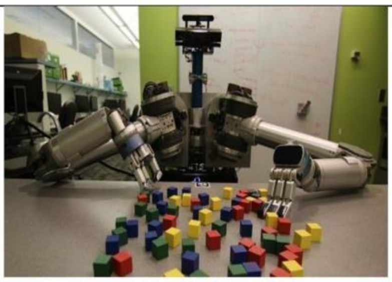

December 2018

# Table of Contents

# 1. Project Description

There is a need for a task-specific domain robot, that will help people with daily chores, such as the "Roomba" [1], a robotic vacuum cleaner. However, to our knowledge, there is no consumer-level robot that will declutter the room, a task which is needed prior to a vacuum operation. A mobile base with a robotic arm on it is not a novel concept. There are examples from toys to research-grade robots. However, by optimizing the design of a robot, its mechanisms, perception and algorithms to a specific task, it should be possible to develop an efficient, affordable, and commercial version of it [2][3]:

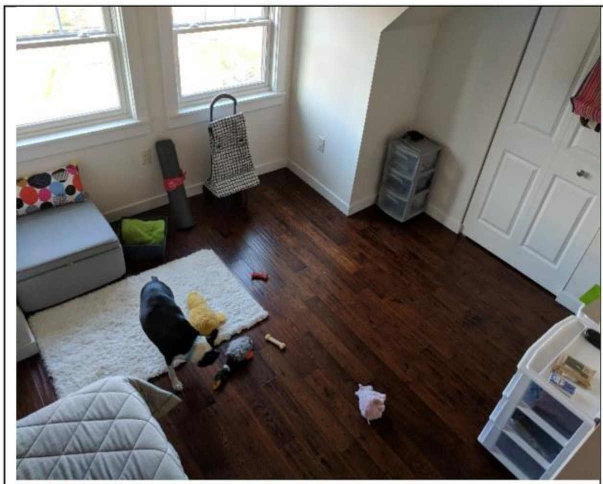
*Roboscooper – Wireless remote control*



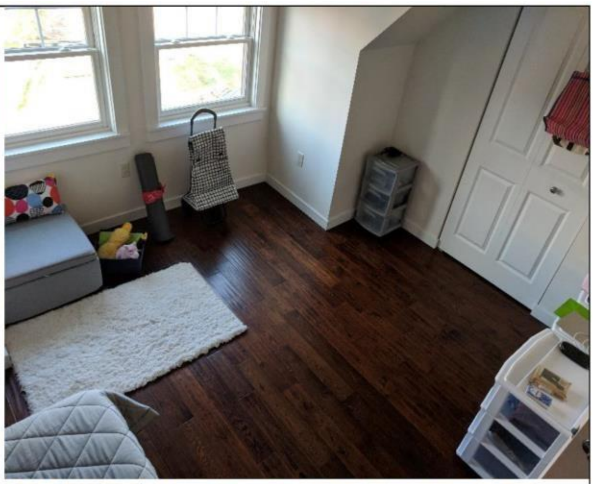
*Herb – application of AI for decluttering workspace Carnegie Mellon University*

*ATRV2 – Mobile base with robotic arm Carnegie Mellon University*

The main goal of this project is to automate the task of picking up clutter to improve the daily lives of parents, pet owners, and daycare workers. By the end of the project, our robot should be able to encounter a room in an initial state, such as the one on the left, and work autonomously, avoiding people, pets and obstacles along the way, to achieve the state on the right, in an optimized matter, with all objects picked up from the floor and placed at a desired destination.



*Initial State (pet avoidance is desirable)*



*Final State (Goal) = Learning State, which will also be used to teach what the goal is*

## 2. Use Case

Zachary is an early childhood educator in an infant toddler classroom. Most of the children are under one year of age. Ellen, who is under his care, suddenly started crying after dropping a rattle from her hand. As Zachary comforted her, he looked around the room. Over the last hour, the room had become cluttered with a dozen toys. As the children explored the space, the educators were busy with changing diapers, giving bottles, and offering children snacks. Now that all of that is finished, it was time to get the room cleaned up and organized so the morning activities can begin.

**CuBi finds and picks clutter**

**CuBi drops clutter at the bin**

Once Ellen stopped crying and was playing peek-a-boo with an educator in the other playroom, Zachary turned on CuBi to perform its work while he gathered the materials needed for the light and shadow exploration. CuBi went around the room, when no babies were around, picking up tennis ball-sized toys from the floor and placed them in the bin at the corner of the room. For 30 minutes, CuBi performed its work without crashing into anything and placed almost all the toys in the bin. Then CuBi went back to its dock to self-recharge for a few hours. Now that the floor was decluttered, Zachary was ready to set up for the light and shadow activity. He was happy to see the clean floor and quickly set up the next experience.

**The room is decluttered after CuBi's operation**

## 3. System-Level Requirements

The system-level requirements are divided into two categories: mandatory requirements and desirable requirements. Under each category, requirements are further classified as performance requirements that are functional requirements with qualitative measures, and non-functional requirements, based on their essence. The requirements originate from the project goal, derived from the use case, and validated through preliminary calculations and stakeholders' feedback.

### 3.1. Mandatory Performance Requirements

The system will:

**M.P.1.** Explore, scan and map 90% of the reachable area in a room.

**M.P.2.** Clean up a 20m² room with a dozen tennis-ball-sized objects within 30 minutes.

**M.P.3.** Navigate to a designated reachable location in a room with pose error < 10%.

**M.P.4.** Go over carpets and rugs with thickness less than 12mm.

**M.P.5.** Detect and avoid 95% of the obstacles with a clearing distance of 20cm.

**M.P.6.** Localize indoors with accumulated error < 10% per 30 minutes of operation.

**M.P.7.** Classify all tennis ball-sized clutter objects with classification error < 20%.

**M.P.8.** Pick up and collect the classified clutter within 5 attempts.

**M.P.9.** Carry at least 2 tennis ball-sized object to the drop-off location.

**M.P.10.** Drop the clutter in a designated container with success rate > 90%.

### 3.2. Mandatory Non-Functional Requirements

The system shall:

**M.N.1.** Operate autonomously.

**M.N.2.** Be mechanically safe (i.e. no sharp edges).

### 3.3. Desirable Performance Requirements

The system will:

**D.P.1.** Continuously operate for at least 2 hours once fully charged.

**D.P.2.** Auto-recharge 70% of its battery within 1.5 hours.

**D.P.3.** Have a sensing range of 15 cm to 4 m.

**D.P.4.** Have a physical dimension limit of 0.5 x 0.5 x 0.5 m.

**D.P.5.** Be affordable with a maximum cost of $5000 USD.

## 3.4. Desirable Non-Functional Requirements

The system shall:

**D.N.1.** Be easy to use by pressing buttons or through a GUI.

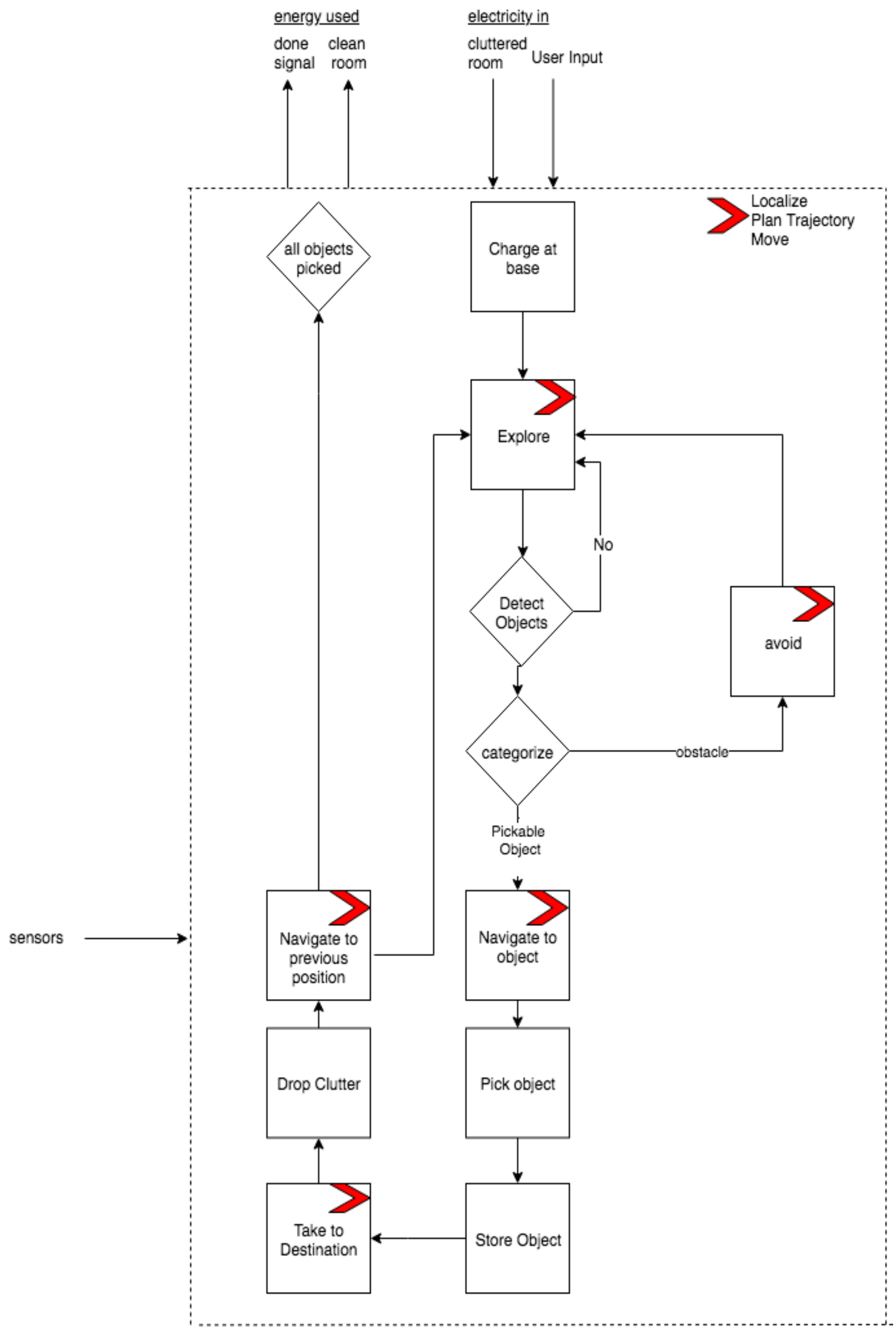
**D.N.2.** Have an inconspicuous, seamless appearance.

**D.N.3.** Be reliable and not get stuck or malfunction frequently.

## 4. Functional Architecture

The functional architecture aligns heavily with our use case. After CuBi is turned on, it is constantly looping through four major functions: exploring, categorizing objects, picking up objects, and dropping them off at a predefined location. It will continue repeating these major tasks until the room has successfully been decluttered. At this point, it will return back to its initial location and start re-charging.

One thing to note about the architecture is that the boxes marked with a red arrow require localizing, trajectory planning and moving. The boxes shaped like diamonds show functions which output decisions that determine what the robot will do next.




**Functional Architecture**

# 5. System-Level Trade Study

## 5.1. Manual vs Autonomous Operation

A system-level trade study determines the best solution for picking up clutter from the floor and meeting system requirements. It was conducted comparing the following scenarios: human performing the task, human using a manual reach and grabbing tool [4], human-robot collaboration, and a fully autonomous robot [5]. In the first case, people cannot afford to spend time off their busy schedule to clean the clutter. The cost of hiring someone to do this task is also high in the long-term. An example of a human-robot collaboration would be in the area of imitation learning, where a user indicates the area from where clutter needs to be collected. This could be accomplished in different ways, such as the robot following the human who walks around the clutter until a closed boundary is formed or a human could drop a colored rope around the cluttered area. A human would also need to indicate the destination for the picked- up clutter. In a fully autonomous scenario, the robot will autonomously detect the clutter, pick it up, and take it to a pre-specified destination.

The chosen criteria for the system-level trade studies are selected based on the highest goals and requirements of the project, with each grade determined based on the importance and subtleties listed above. It is hard to assign cost to a human since a human can be productive and happier performing more relevant tasks. Therefore, saving time and effort is very important to the project. Being safe, reliable, and seamless are also critical to properly deploy the robot at households. Finally, affordability is key if the robot will be adopted by many people. Based on the criteria and grade selected, the trade study shows that the fully autonomous option scored the highest.

**Trade Study: Manual vs Autonomous**

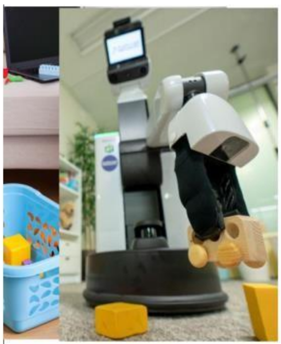
| | |  |  |  |  |
|---|---|---|---|---|---|
| **Value Ratings *** | **Specs** | Person | Person using Manual Tool | Semi-Autonomous Robot | Fully Autonomous Robot |
| 1: Inadequate | **Price (USD)** | N/A | $20 | $2,000 - $20,000 | $2,000 - $20,000 |
| 2: Tolerable | **Cost of Operation** | $20/hour to hire someone | $20/hour to hire someone | $10/hr | $1/hr |
| 3: Adequate | | | | | |
| 4: Good | | | | | |
| 5: Excellent | | | | | |
| **Criteria** | **Weight Factor (100%)** | **Value (1 - 5) *** | | | |
| **Save Time & Effort** | 20 | 1 | 2 | 3 | 5 |
| **Operates Autonomously** | 10 | 1 | 1 | 4 | 5 |
| **Operates Efficiently** | 5 | 1 | 2 | 4 | 5 |
| **Easy to Use** | 10 | 1 | 2 | 4 | 5 |
| **Seamless** | 10 | 5 | 4 | 4 | 2 |
| **Safe** | 10 | 4 | 4 | 3 | 3 |
| **Reliable** | 10 | 5 | 5 | 3 | 3 |
| **High Coverage Areas** | 10 | 2 | 3 | 4 | 5 |
| **Affordable** | 15 | 5 | 5 | 3 | 2 |
| **Weighted Average** | 5 | **2.8** | **3.15** | **3.45** | **3.85** |

*** Subjective Value Method**

## 5.2. Robot Level Trade Study

Going down one level in our trade-study, different types of autonomous robots are compared to find the most viable one. The trade studies include: a fully general-purpose research-level robot[5], able to pick up clutter in the room and perform many other tasks, a purpose-specific designed robot dedicated to the task of picking up small objects from the floor, and finally a generic multi-purpose mobile base with an integrated robotic arm on top of it[6], which is ROS-compatible and available off-the-shelf. The main question to consider is if any hardware solution should be acquired and the focus of the project should be on software development and AI only, or if the system should include the full design, development and built of purposely built hardware. Based on the trade-study above, the option of task-specific robot scored the highest.

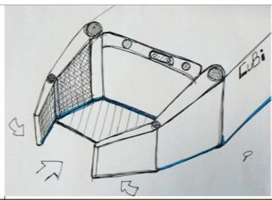
**Trade Study: Robot Level**

| | | | | |
|---|---|---|---|---|
| | |  |  |  |
| | **Specs** | General Research Robot | Task Specific Robot | Interbotix Turtlebot 2i Mobile ROS Platform |
| **Value Ratings *** | **Price (USD)** | $11,750 | $2,500 | $2,695 |
| 1: Inadequate | **Dimensions** | 135cm tall | 50cm x 50cm x 50cm | Flexible |
| 2: Tolerable | **Weight** | 37 Kg | 1.8 kg | 5.0 kg (estimated) |
| 3: Adequate | **Payload** | 20 kg | 2 Kg | 1 Kg |
| 4: Good | **Maximum Speed** | 50 cm/s | 70 cm/s | 70 cm/s |
| 5: Excellent | **Number Picking Objects** | 1 | 2 | 1 |
| **Criteria** | **Weight Factor (100%)** | | | |
| **Save Time & Effort** | 20 | 5 | 5 | 5 |
| **Operates Autonomously** | 10 | 5 | 5 | 5 |
| **Operates Efficiently** | 5 | 5 | 5 | 3 |
| **Easy to Use** | 10 | 3 | 5 | 3 |
| **Seamless** | 5 | 1 | 5 | 2 |
| **Safe** | 20 | 4 | 5 | 2 |
| **Reliable** | 5 | 5 | 5 | 3 |
| **Durable** | 5 | 5 | 5 | 3 |
| **High Coverage Areas** | 5 | 5 | 4 | 2 |
| **Affordable** | 15 | 2 | 5 | 5 |
| **Weighted Average** | 5 | **3.95** | **4.95** | **3.6** |

## 5.3. Manipulator Trade Study

This trade study compares the option of using an off-the shelf gripper and manipulator [7][8], versus the option of building a task-specific gripper and manipulator. The study below shows the specific technical specs and features of each option. In this case, functional requirements weighted heavily, since performing the task adequately and in a holistic manner, inside households is the focus of the project. The robot acceptance by the way it looks, operates and completes a task is more important to users than pure speed or performance. Intangibles such as seamlessness, and low noise levels are important for the system. It does not meet the reach or speed of a traditional arm, but it is mounted on a mobile base, and the speed will be more than adequate to meet the requirements of the system.

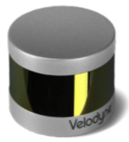
**Manipulator Trade Study**

| | Concept |  |  |  |
|---|---|---|---|---|
| | **Specs** | Industrial Robotic Arm | Hobby Robotic Arm | Domain Specific Manipulator |
| | **Description** | UR-3 Collaborative table-top | uArm Swift Pro | CuBi Custom (estimate specs) |
| | **DOF** | 6 DOF | 4 DOF | 3 DOF + 2 under actuated |
| | **Repeatability** | Repeatability +/-0.1mm | Rep +/- 0.2mm | Rep +/- 0.2mm |
| **Value Ratings *** | **Footprint** | Dia 128mm | 150 x 140mm | 150mm x 5mm each |
| 1: Inadequate | **Weight** | 11.2Kg | 2.2Kg | 1.5Kg including motors |
| 2: Tolerable | **Payload** | 3Kg | 500g | 1Kg |
| 3: Adequate | **Max Speed** | 1m/s | 100mm/s | 100mm/s |
| 4: Good | **Reach** | 500mm | 320mm | 150mm |
| 5: Excellent | **Cost** | $23,000 | $800 | $1,000 |
| **Criteria** | **WeightFactor (100%)** | | **Value (1 - 5) *** | |
| **Speed** | 5 | 5 | 4 | 3 |
| **Reach** | 5 | 5 | 3 | 3 |
| **Payload** | 5 | 5 | 3 | 4 |
| **Affordability** | 10 | 1 | 5 | 5 |
| **Safety** | 20 | 5 | 4 | 5 |
| **Compact Size** | 15 | 3 | 3 | 5 |
| **Dexterity** | 5 | 5 | 4 | 3 |
| **Accuracy** | 5 | 5 | 3 | 3 |
| **Development Time** | 5 | 5 | 5 | 3 |
| **Seamless Design** | 5 | 1 | 3 | 5 |
| **Weight** | 10 | 1 | 3 | 5 |
| **Durability** | 10 | 5 | 3 | 4 |
| **Weighted Average** | 5 | **3.2** | **3.3** | **3.95** |

## 5.4. Sensor Trade Study

A sensor trade study was done taking into consideration multiple things like potential use of a single or fewer sensors for multiple applications of object detection and classification; pose and size estimation; localization and mapping. The active structured stereo camera satisfies all the above applications. It gives us RGB, point cloud and depth information together. It works well in indoor environments and low light conditions, satisfies the range requirements, is affordable, has a decent field of view, is ROS-compatible and is not computationally demanding.

**Sensor Trade Study**

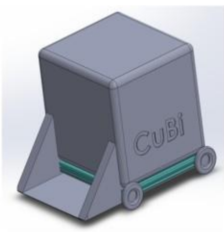
| Trade Studies | Sensors | | | | |
|---|---|---|---|---|---|
| | |  |  |  |  |
| | **Specs** | Active Structured Stereo Camera (E.g. IR Projection) | Passive Stereo Camera | Laser Scanner (LiDAR) (2D) Single Beam | Laser Scanner (LiDAR) (3D) 16 Beam |
| | **Reference Model** | Orbbec Astra | Stereo Labs ZED | RP LiDAR A2 | Velodyne Puck |
| | **Output** | RGB + RGBD + Point Cloud | RGB + RGBD + Point Cloud | Point Cloud | Point Cloud |
| | **Computation Cost for depth** | Medium | Very High (Require GPU) | Low | Low |
| | **Low light performance** | Medium | Poor | High | High |
| | **Rich Visual Data for CV** | Yes | Yes | No | No |
| | **Range** | 0.6 – 5.5m | 0.5 - 10 m | 0.15 - 12 m | 80 m |
| | **Resolution** | High and Dense | Very High and Dense | Low | High |
| **Value Ratings *** | **Footprint** | 165 x 30 x 40 mm | 175 x 30 x 33 mm | 73 x 73 x 40 mm | 103 x 72 x 72 mm |
| 1: Inadequate | **Weight** | 300 g | 159 g | 190g | 830g |
| 2: Tolerable | **FoV** | 60° (H) x 49.5° (V) 73° (Diag) | 90° (H) x 60° (V) x 110° (Diag) | 360° (H) | 360° (H), ± 15° (V) |
| 3: Adequate | **Data Interface** | USB | USB | USB | USB |
| 4: Good | **FPS / Scan Rate (LiDAR)** | 30 @720p | 60 @ 720p | Scan rate - 10Hz | Scan rate - 10Hz |
| 5: Excellent | **Cost** | $150 | $500 | $320 | $4,000 |
| | **Reflectance Issues** | Adversely affected | Not greatly affected | Adversely affected | Adversely affected |
| **Criteria** | **Weight Factor (100%)** | **Value (1 - 5) *** | | | |
| **Applications** | | | | | |
| **Object Detection** | 12 | 5 | 5 | 2 | 4 |
| **Object Identification** | 9 | 5 | 5 | 1 | 2 |
| **Localization** | 14 | 3 | 3 | 4 | 5 |
| **Performance Factors and Non-functional requirements** | | | | | |
| **Output information** | 12 | 5 | 5 | 2 | 3 |
| **Range** | 8 | 4 | 4 | 4 | 5 |
| **Affordability** | 8 | 5 | 4 | 4 | 1 |
| **Low light performance** | 7 | 4 | 3 | 5 | 5 |
| **Computation requirement** | 7 | 4 | 2 | 5 | 4 |
| **Reflectance issues** | 4 | 3 | 5 | 3 | 3 |
| **Resolution** | 5 | 4 | 5 | 2 | 5 |
| **FoV** | 5 | 3 | 4 | 3 | 5 |
| **Accuracy** | 5 | 4 | 3 | 3 | 4 |
| **Size** | 2 | 4 | 4 | 5 | 3 |
| **Weight** | 2 | 5 | 5 | 5 | 3 |
| **Weighted Average** | 5 | **4.2** | **4.04** | **3.19** | **3.77** |

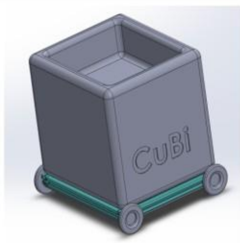
*** Subjective Value Method**

## 5.5. Storage Trade Study

The first option is to have the actual tray between the grippers. In this case, two or three objects could be picked and taken to a destination at the same time. The second option is to have a trunk built into the body of the robot. This option gives an all-in-one solution but adds substantially to the mechanical and actuation complexity of the robot, since the grippers would have to place the objects inside the trunk. In this case, the robot would keep the objects in it until a user manually removes them, which would defeat the purpose of the project. Adding a manipulator to remove the objects from its trunk to take it to a destination, would again increase complexity level and cost. The final option would be for the robot to haul a trailer where it would place the objects collected. Multiple trailers could exist, each containing a set of picked-up objects. In that case the advantage would be that the robot would not have to take the object to another destination, but simply leave the trailer automatically where it belongs. This option also adds considerable mechanical complexity since the robot would have to rotate its body 180 degrees to place the objects picked behind the trailer. A conveyor to deliver the objects behind the robot is also an option. But in both cases, cost would be higher. The trade-study below shows that the simplicity, ease of use and proper capacity of the front tray system, scored higher than the other two options.

**Storage Trade Study**



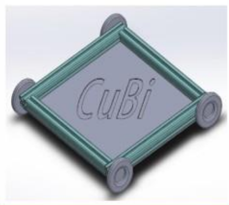



| | Specs | Front Tray Storage | Body Trunk Storage | Rear Trailer Storage |
|---|---|---|---|---|
| **Value Ratings *** | **Price (USD)** | $100 | $100 | $300 |
| 1: Inadequate | **Dimensions (cm)** | 20 x 40 x 10 | 30 x 30 x 10 | 30 x 30 x 30 |
| 2: Tolerable | **Cubic Capacity cm^3** | 8000 | 13500 | 27000 |
| 3: Adequate | **Weight (g)** | 500 | 200 | 1500 |
| 4: Good | **Payload** | 1 Kg | 2 Kg | 5 Kg |
| 5: Excellent | **Number Picking Objects** | 3 | 5 | 10 |
| **Criteria** | **Weight Factor (100%)** | | | |
| **Multiple Toy Storage** | 10 | 3 | 4 | 5 |
| **Operation Speed** | 10 | 3 | 4 | 5 |
| **Mechanical Complexity** | 20 | 5 | 3 | 2 |
| **Go over carpets** | 10 | 5 | 5 | 3 |
| **Seamless** | 5 | 5 | 5 | 3 |
| **Safe** | 20 | 5 | 5 | 3 |
| **Ease of Use** | 5 | 5 | 3 | 4 |
| **Durable** | 5 | 5 | 5 | 3 |
| **High Coverage Areas** | 5 | 3 | 4 | 5 |
| **Affordable** | 10 | 5 | 5 | 3 |
| **Weighted Average** | 5 | **4.5** | **4.25** | **3.35** |

## 5.6. Mobile Base Trade Study

There are various off-the-shelf mobile base platforms available that are specifically designed for robotics research and development. These mobile bases vary in many aspects of specifications, including physical dimension, payload, duration, traversability, etc. Several criteria were chosen, as listed below, to evaluate three common off-the-shelf robotic mobile bases, based on the project goal. A free Create 2 mobile base was received [9] from iRobot as a gift. However, the specifications are not a good fit for our project, according to the trade study conducted. The TurtleBot 3 Waffle [11] scores the highest and will likely be adopted as the testbed platform. However, if it turns out that the base cannot satisfy our specific needs in the future, as development and testing proceed, a customized mobile base would have to be designed and built.

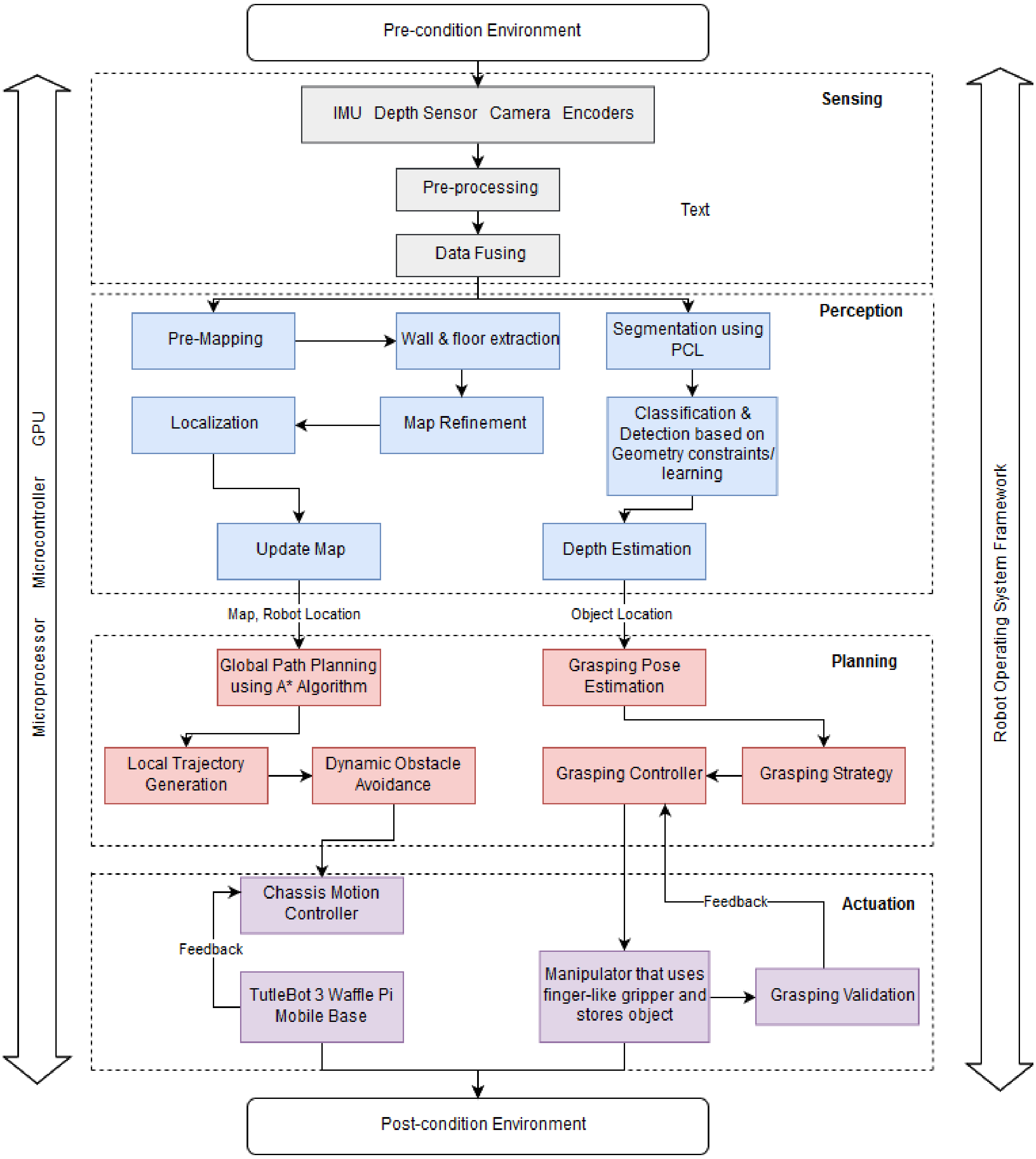
**Mobile Base Trade Study**

| | Mobile Bases | | | | |
|---|---|---|---|---|---|
| | |  |  |  |  |
| | Specs | iRobot Create® 2 | Clearpath Robotics Jackal | TurtleBot 3 Waffle | Self-designed Mobile Base |
| | Cost (USD) | $199 | $12,750 | $1,399 | $1000 (estimated) |
| Value Ratings * | Dimension (L x W x H) | 350 x 350 x 93 mm | 508 x 430 x 250 mm | 281 x 306 x 141 mm | Flexible |
| 1: Inadequate | Weight | 3.5 kg | 17 kg | 1.8 kg | 8.0 kg (estimated) |
| 2: Tolerable | Maximum Payload | 10 kg | 20 kg | 15 kg | 15 kg (estimated) |
| 3: Adequate | Duration | 1 hour | 6 hours | 2 hours | 2 hours (estimated) |
| 4: Good | Charging Time | 2 hours | 4 hours | 2.5 hours | 2 hours (estimated) |
| 5: Excellent | Expandability | Medium | Medium | Medium | High |
| Criteria | Weight Factor (100%) | Value (1 - 5) * | | | |
| Cost | 12 | 5 | 1 | 3 | 4 |
| Payload | 12 | 3 | 5 | 4 | 4 |
| Expandability | 15 | 3 | 5 | 4 | 5 |
| Charging Time | 5 | 4 | 3 | 2 | 3 |
| Noise Level | 10 | 3 | 2 | 4 | 4 |
| Durability | 10 | 3 | 5 | 4 | 4 |
| Traversability | 6 | 3 | 5 | 3 | 4 |
| Development Time | 20 | 4 | 4 | 4 | 2 |
| Seamless Design | 5 | 5 | 2 | 5 | 5 |
| Weight | 5 | 5 | 2 | 5 | 4 |
| Weighted Average | | 3.69 | 3.62 | 3.82 | 3.75 |
| * Subjective Value Method | | | | | |

## 6. Cyber Physical Architecture

The Cyber physical architecture shows the interactions between the hardware and software components of the system. It is divided according to the basic functionalities of a robot: Sensing, Perception, Planning, and Actuation.

Cyber physical connects each function in our architecture to a physical and information aspect of our system. For example, the 'Avoid Obstacle' function requires CuBi to sense, identify, and plan in order to accomplish it. Thus, our architecture has been designed in such a way that the subfunctions of each function relate to individual components of our system.




**Cyber physical Architecture**

# 7. Subsystem Descriptions

## 7.1. Mobile Base

The mobile base of the system will be a wheeled-based chassis with several actuated motors and caster wheels. In order to operate in a complex, cluttered indoor household environment efficiently, the mobile base will have the capability to go over mattresses and rugs, be relatively small and agile, and be energy-efficient to operate continuously for at least two hours. The mobile base will have internal batteries, motor controllers and wheel encoders, and it will provide communication and power source for the payload, such as the manipulator, onboard computer, peripheral sensors, etc. After conducting the trade studies, the TurtleBot 3 Waffle Pi has been chosen as the mobile base platform. However, if the base cannot meet our requirements in future development and testing, we might have to design our own mobile base platform for our specific needs.

## 7.2. Gripper

The gripper consists of two actuated arms and two under-actuated fingers. The left and right arm might work synchronously to grab a specific object, particularly a larger one, or asynchronously, alternating its movement, depending on the object size and shape, to pull smaller objects onto the tray. If the gripper developed does not work, an off-the-shelf manipulator might have to be used.

## 7.3. Storage

Storage consists of a tray where the gripper will place the objects. The tray serves as a wedge which, in conjunction with the gripper will scoop objects up from the floor. The tray will be large enough to hold three tennis ball-sized objects, acting as a buffer. This interim storage capacity allows the robot to optimize picking sequence and trajectory, saving traveling time and meeting the performance requirements. The tray will be emptied when the robot takes the objects to its destination. The tray might also be mounted on a prismatic joint and be retractable to facilitate the discharge of the objects into a storage bin. If we are not able to place the objects inside a container because of the additional complexity of the mechanism required, we will instead determine a location on the floor, marked with a square tape, and allow the robot to deliver parts inside the marked area.

## 7.4. Perception

Perception will be used to sense the environment and to identify different objects. A stereo RGBD camera will be used for obtaining rich visual RGB data for classification, as well as depth and point cloud for size and distance estimation. Combined information will be used to estimate the size, pose and type of object. A combination of techniques such as geometric vision, learning-based and probabilistic methods will then be used to classify objects into two categories: objects to pick and obstacles to avoid. For example, a 3D bounding box would be fitted around each detected object to classify them according to threshold criteria of shape, size and pose. A list of estimated object data and relative locations will be maintained and utilized by the planner for navigation. If geometric features prove to be an insufficient criterion, then object classification through learning with additional labeled or synthesized data will be used.

## 7.5. Localization

The robot will be able to localize itself in the room. The localization can be achieved either with a pre-built map, or through real-time simultaneous localization and mapping (SLAM) techniques. The robot will be equipped with multiple sensors, including wheel encoders, stereo

RGBD cameras, or 2D laser scanners. With the real-time sensor readings and a map of the environment, probabilistic algorithms, such as Particle Filter, can be utilized for localization. If no map is provided, the robot will also be able to localize itself by propagating its state, performing visual SLAM or laser SLAM algorithms.

## 7.6. Planner

The path planner for the system will consist of two parts: the global path planner and the local trajectory generator. The former is a high-level planner which takes in our pre-built map as an input, generates a cost-map based on some constraints, and finds a traversable, collision-free path from designated point A to point B. The local trajectory generator takes this generated path from the global planner, performs interpolation, and calculates a set of waypoints with desired position, velocity and acceleration values based on sensor readings and vehicle dynamics within a local area centered around the robot itself. When dynamic obstacles are detected along the generated path, the local trajectory generator will also perform an online update of the waypoints in order to avoid and keep a clearing distance from the obstacles.

# 8. Project Management

## 8.1. Work Plan & Tasks

**1. Visual sensing and perception**

1.1. Sensor identification, calibration, setup and installation

1.2. Data collection, pre-processing, augmentation. synthesis, labelling and fusion

1.3. Object detection algorithm to estimate depth, size and pose

1.3.1. Geometry based algorithm

1.3.2. Segmentation algorithm

1.4. Object classification algorithm

1.5. Destination/ base identification algorithm

1.6. Localization and mapping

1.6.1. Initial mapping and map refinement (removal of walls and floor)

1.6.2. Localization - odometry method selection, feature extraction, loop closure

**2. Planning and navigation**

2.1. Planning and navigation algorithm

2.1.1. Global path planning

2.1.2. Trajectory generation

2.1.3. Local planner

2.1.4. Dynamic obstacle avoidance

2.2. Self-charging algorithm (desired)

2.3. Path optimization algorithm (research)

2.4. Learning based planning algorithms (research)

**3. Grasping**

3.1. Manipulator

3.1.1. Grippers R&D: Trade studies, prototype drawing, and fabrication

3.1.2. Actuation mechanism (arms and fingers)

3.1.2.1. Selection and set up of motors and controller after trade studies

3.1.2.2. Validation of assembly on actual toys

3.2. Storage R&D: Trade studies, prototype drawing and fabrication

3.3. Chassis and body R&D: Trade studies, prototype drawing and fabrication

3.4. Microcontroller selection and programming

3.5. Control algorithms for manipulator

3.5.1. Grasping pose estimation

3.5.2. Grasping strategy and trajectory generation

3.5.3. Feedback mechanism and validation

3.5.4. Storage and dropping

3.6. Learning for manipulation (research)

**4. Mobility**

- 4.1. Mobile base trade studies (e.g. payload, torque, turning radius)
- 4.2. Modification of motors and actuators
- 4.3. Microcontroller selection and programming
- 4.4. Encoder interfacing
- 4.5. Controller for mobility: algorithm and feedback mechanism

**5. Robot integration**

- 5.1. Electric and mechanical components integration
  - 5.1.1. Attachment and assembly of sensors, encoders, motors and actuators, battery, manipulator, storage, processors and microcontrollers
  - 5.1.2. Power management and electrical design and assembly
    - 5.1.2.1. Calculate power consumptions for each component
    - 5.1.2.2. Trade studies for suitable power source
    - 5.1.2.3. Electronic circuit design
    - 5.1.2.4. Wiring of all components
- 5.2. Software Integration
  - 5.2.1. ROS
    - 5.2.1.1. Sub-systems urdf files
    - 5.2.1.2. Sub-Systems ROS implementation
    - 5.2.1.3. Robot urdf files
    - 5.2.1.4. System ROS Integration
  - 5.2.2. Sub-system level pipelines and system-level pipeline
  - 5.2.3. Communication protocols
  - 5.2.4. Parallel processing

**6. Validation, testing and benchmarking**

- 6.1. Subsystem level
  - 6.1.1. Object detection and classification
  - 6.1.2. Localization and mapping
  - 6.1.3. Planning and navigation
  - 6.1.4. Manipulation and storage
  - 6.1.5. Mobility
  - 6.1.6. Sub-systems ROS validation
- 6.2. System-level
- 6.3. Non-functional / desired (e.g. mechanical and electrical safety, speed, noise)
- 6.4. Battery consumption and duration

**7. Project management**

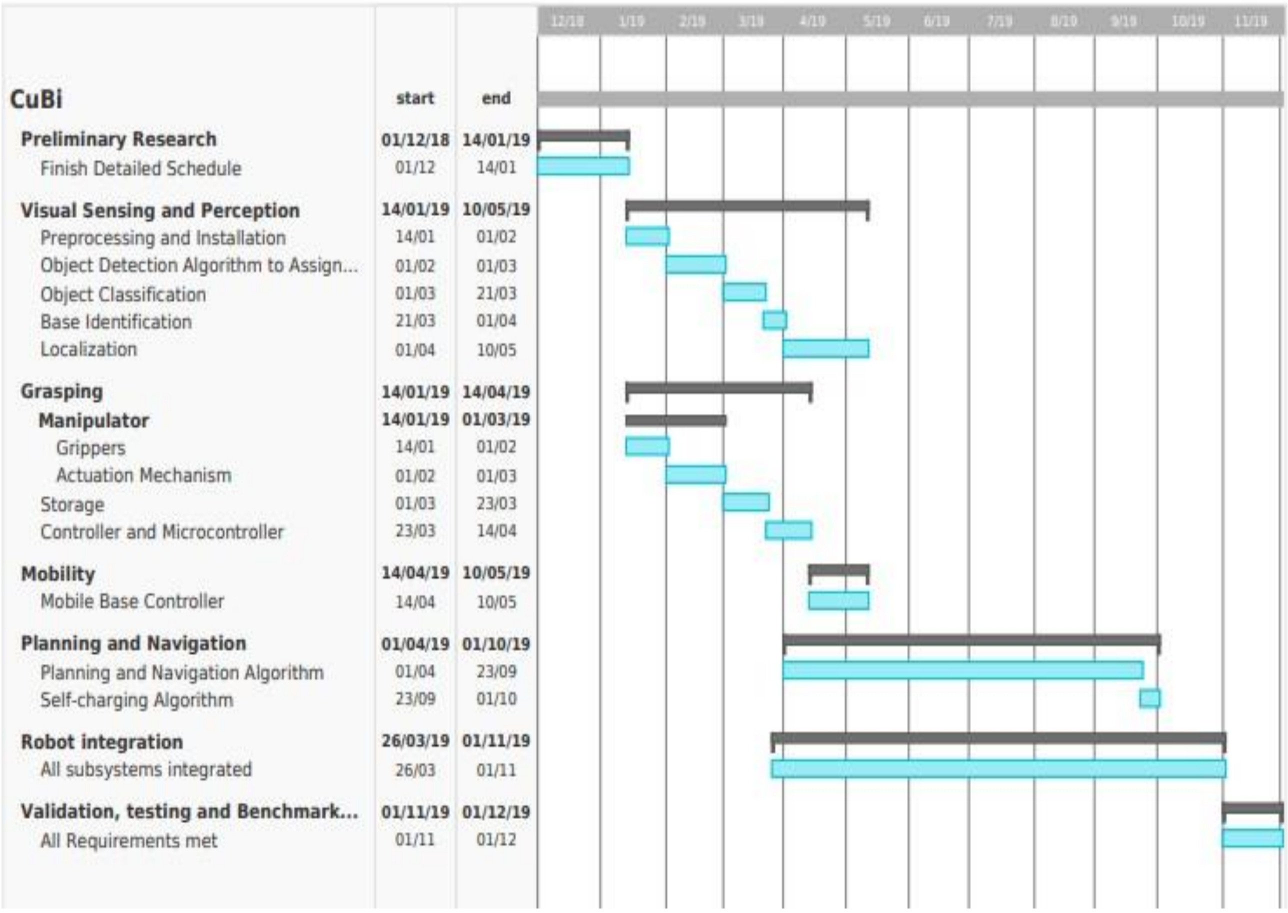
- 7.1. Team and work management
- 7.2. Schedule management
- 7.3. Cost management
- 7.4. Risk management
- 7.5. Resource management

## 8.2. Schedule & Progress Reviews

Assumption: 5 people working 15 hours a week for 30 weeks.

Management and logistics will take 10 hours a week. Logistics include planning meetings, preparing external reports or presentations, etc. Management includes risk mitigation, updating schedule, and debriefing about what is going well and poorly.

With the help of the WBS, a list of tasks was formulated. Each of the lowest-level tasks takes approximately 40 to 50 hours to complete. However, to be able to effectively and more accurately create a semester plan and mitigate any risks, lower-level trade studies and further research on subsystems and components needs to be conducted. Only at that point can the team create granular tasks, ideally around 5 to 15 hours in length, from which weekly schedules can be generated. As can be seen in the Gantt chart below, this will be done over the winter break so that there is a clear idea of what need to be done at the beginning of the spring semester.




**Initial Gantt chart representing high-level timeline of the project**

In spring the focus will be on developing the subsystems and during fall, trajectory planning, integration of all subsystems and validation would be performed. This schedule gives a 14-day buffer to the end of the semester. This buffer can be incremented if any work is done during the 2019 summer.

For the first progress review, all the main trade studies would have been finished and detailed weekly plans and 2-4-week plans for the spring semester will be created.

For the second progress review, the team will demo the concept validation experiment shown below. To do so, a version of the 3D-printed gripper will be created.

**An example of Concept Validation from our Grasping & Storage Subsystem:**

One of the highest risks of the project is to choose a gripper design that is mechanically safe and simple. This will be tested by placing one example toy on the ground and attaching a prototype gripper onto a cardboard box and seeing if we can pick up toys just by moving the cardboard box and grippers manually of with a remote controller. This will also help give us a sense of what the major difficulties will be when creating the automated gripper.

### 8.5. System Validation Experiments

The V-model will be followed for the validation experiments. Therefore, there will be continuous iterations and testing at the component, subsystem and system level. The team has slightly modified this model to include a concept validation which will be performed during the design stages of each subsystem. With this, there will be a certainty that the trade studies will lead to reasonable decisions and hence begin to mitigate future risks. These concept validations will focus on testing the highest-risk assumptions of each subsystem and the integration with other subsystems. An example of this can be found in the appendix.

### Spring 2019

By the spring validation test day, the goal would be to have finished designing and building all subsystems and demonstrate them by passing all subsystem validation tests. These experiments consist mostly of performance requirement tests. Future integration risks will be eliminated with the concept validation tests. Specifically, on test day, two tests will be performed that will help validate the key performance requirements related to the two main subsystems: perception and grasping.

*** All tests can be performed in a conference room and require CuBi, its charging base, 4 markers used for landmarks, and three toys.*

**M.P.3.** Will navigate to a designated reachable location in a room with pose error < 10%.

- Place 4 markers on the ground. Each of them will be 2 meters away from the starting location placed around CuBi. The 4 markers will form a square around CuBi.

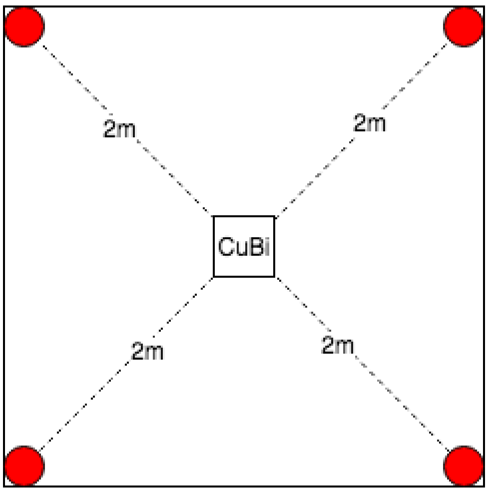



**Diagram of where CuBi and the markers will be placed. Markers are the red circles.**

- CuBi will be placed facing one of the markers. Turn CuBi on and have CuBi navigate to each of the 4 markers in clockwise order and output the estimated distance between each marker. Since trajectory planning subsystem will not be finished, the path will be pre-specified, and no obstacles will be placed in it.
- Check if distances are within 10% of the ground truth.
- The ground truth will be determined using a ruler.

**M.P.7. :** Will pick up and collect the classified clutter within 5 attempts.
- Place a toy flat on the ground and put CuBi 0.1m in front of it. The toy will be an example of what exists in the day care.
- Turn CuBi on and record how many attempts it takes CuBi to pick up the toy.
- Repeat these 3 times.

## Fall 2019

The focus during Fall semester will be on optimizing, making the system more robust, and especially integrating all subsystems. The tests performed on the test day are system-level and require all subsystems to be completed.

System-level:

1. **M.P.2.** Will clean up a 20m² room with a dozen clutter objects within 30 minutes.
    a. Independent party will place 15 toys in a pre-defined reachable area. The toys will consist of toys that can be found in the day care. The toys must be flat on the ground and not laying on obstacles.
    b. Turn CuBi on and CuBi will navigate the room, search for the toys and place them in the designated location.
    c. All the toys which end up in the designated location will be considered picked up. For this test to be considered a success, CuBi needs to correctly return 12 toys to the correct location.
2. **M.P.4.** Will detect and avoid 95% of the obstacles with a clearing distance of 20cm.
    a. Put CuBi in a room full of chairs and tables. These will be placed by a third party. All obstacles must be placed 0.9m away from each other (so that CuBi with a size of 0.5m and can have a clearance of 0.2m on each side).
    b. Place 4 markers on the ground. Each of them will be 2 meters away from the starting location placed around CuBi. The 4 markers will form a square around CuBi.
    c. Turn CuBi on. CuBi needs to reach all four waypoints without crashing.

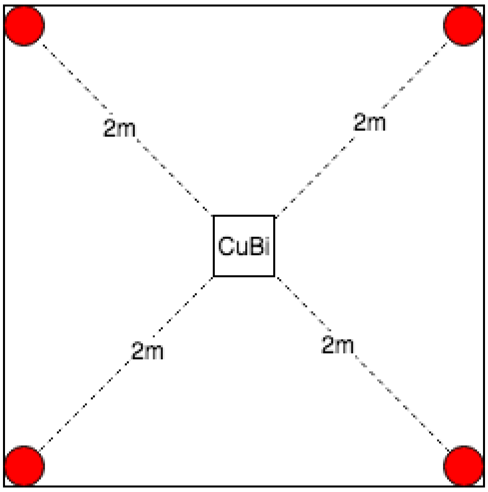



**Diagram of where CuBi and the markers will be placed. Markers are the red circles.**

## 8.6. Team Member Responsibilities

**Table. Team Member Responsibilities**

| **Tasks** | **Bobby** | **Jorge** | **Laavanye** | **Nithin** | **Paulo** |
|---|---|---|---|---|---|
| Object detection and classification | | 2 | 1 | 3 | |
| Manipulator Designing | | 3 | | 2 | 1 |
| Manipulator Controls | | | 3 | 1 | 2 |
| Mobility Controls | 3 | 2 | | 1 | |
| Planning | 1 | 3 | 2 | | |
| SLAM | 2 | | 1 | | 3 |
| Robot integration | 1 | 3 | | | 2 |
| Software integration | 1 | 3 | 2 | | |
| Project Management | | 1 | | 2 | 3 |

1. Primary responsibility 2. Secondary responsibility 3. Tertiary responsibility

## 8.7. Provisional BOM

| Component | Manufacturer | Part # | Unit Price (USD) | Quantity | Total Cost |
|---|---|---|---|---|---|
| Mobile Base | TurtleBot | TurtleBot 3 Waffle Pi | 1399 | 1 | 1399 |
| RGBD Camera | Intel | RealSense D435i | 199 | 2 | 398 |
| Computing Platform | Intel | NUC8i7 | 449 | 1 | 449 |
| Computing Platform | Nvidia | Jetson TX2 | 599 | 1 | 599 |
| Laser Scanner | Hokuyo | URG-04LX-UG01 | 1115 | 2(Inventory) | 0 |
| Microcontroller | Teensy | Teensy 3.1 | 19.95 | 2 | 39.9 |
| Servo Motor | Dynamixel | XM430-W350-T | 229.9 | 4 | 919.6 |
| Connector | Dynamixel | ROBOTIS FR12-H101K Set | 33.9 | 2 | 67.8 |
| Connector | Dynamixel | ROBOTIS FR12-S101K Set | 23.9 | 2 | 47.8 |
| Chassis Motor | DJI | RM M3508 | 115 | 4 | 460 |
| Motor Controller | DJI | RM C620 | 89 | 4 | 356 |
| Connector | DJI | RM M3508 Accessories Kit | 89 | 1 | 89 |
| T Slot Aluminum Extrusion | Zyltech | EXT-2020-REG-1000-10X | 7.99 | 10 | 79.9 |
| Aluminum Profile Connector (20 Set) | PZRT | 2020 Series | 25.99 | 1 | 25.99 |
| | | | | | **4930.99** |

## 8.8. Risk Management

| S. No | Risk | Type | Owner | Likelihood | Conse-quence | Risk Mitigation |
|---|---|---|---|---|---|---|
| 1 | TurtleBot 3 Waffle Pi might not be suitable for CuBi | Technical | Bobby | 3 | 5 | • Provide enough time for validation<br>• Identify off-the-shelf alternatives or design a customized base<br>• Research beforehand for system compatibility and reliability |
| 2 | Geometry-based classification might not be accurate | Technical | Laavanye | 4 | 3 | • Use labelled data for learning-based classification or synthesize data. |
| 3 | Lack of time for validation | Scheduling | Nithin | 3 | 4 | • Start building physical and software systems parallelly<br>• Test software on existing platforms |
| 4 | Privacy issue might prevent us to use camera | Technical | Jorge | 4 | 4 | • Blur the faces of children on image<br>• Obtain approval from the children's parents<br>• Not saving any images in the memory |
| 5 | Single sensor for all applications might be insufficient. | Technical/ Scheduling | Laavanye | 4 | 3 | • Procure LiDAR from the inventory |
| 6 | Manipulator might not be able to scoop objects | Technical | Paulo | 3 | 4 | • Validate the idea using a prototype<br>• Limit scope of objects to be picked up<br>• Identify off-the-shelf alternatives |

| | | | | | | |
|---|---|---|---|---|---|---|
| 7 | Budget may not afford expensive sensors | Cost | Bobby | 3 | 2 | • Use sensors from the MRSD inventory<br>• Ask for sensors from the sponsor |
| 8 | Manipulator design and integration might take time | Technical | Nithin | 2 | 4 | • Finalize the design and outline integration over this winter<br>• Find off-the-shelf alternatives |
| 9 | Parts and accessories might not be available when needed | Scheduling | Jorge | 4 | 3 | • Prepare a schedule to finalize and order parts<br>• Prepare a list of alternatives for each part |
| 10 | Discontent among the team due to personal work | Personnel | Laavanye | 5 | 4 | • Team members cover for critical occasions<br>• Provide buffer time to each task |
| 11 | Low/ no light conditions might be a problem for the vision system | Technical | Laavanye | 5 | 5 | • Impose a precondition to have light in the room before operation.<br>• Research in the area of low/ no light conditions (special sensors, methods). |

# 9. References


[1] https://www.irobot.com/

[2] http://www.theoldrobots.com/Roboscooper.html

[3] https://www.cmu.edu/news/stories/archives/2016/may/robots-clutter.html

[4] https://www.amazon.com/Ettore-49036-32-Inch-Gripn-Inches/dp/B001B13PC2

[5] https://www.toyota-global.com/innovation/partner_robot/

[6] https://www.trossenrobotics.com/interbotix-turtlebot-2i-mobile-ros-platform.aspx

[7] https://www.ufactory.cc/#/en/uarmswift

[8] https://www.universal-robots.com/

[9] https://cdn-shop.adafruit.com/datasheets/create_2_Open_Interface_Spec.pdf

[10] https://www.clearpathrobotics.com/jackal-small-unmanned-ground-vehicle/

[11] http://www.robotis.us/turtlebot-3/