NATIONAL UNIVERSITY OF SINGAPORE

AY 2021/2022

EG3301R Final Report

Team: **IS-333 AI Laundromat**

Team Members:

1. Anurag Chatterjee
2. Zhang Huiwen
3. Ong Jia Yuan

Introduction

Singapore has one of the highest proportion of aged workers in ASEAN and worldwide, with a median working age of 42.2 years[1[1]] with approximately 1 in 4 people aged 50 and above in 2020 [2]. Based on the latest data, the proportion of working elderly has increased by 4% and is projected to be higher in the years to come [3][4]][5][6].

We have identified that the laundry industry in particular has the highest proportion of elderly workers in Singapore, where close to 70% of people employed are above 60 years old[7]. There has also been an acute manpower crunch in this industry due to a tight labour market [8], slower population growth [9][10] and falling birth rates[11]. A higher average working age brings in mobility problems [12] which is a problem as laundry needs to move fast to meet tight delivery schedules of customers. This led to higher operational inefficiencies in the industry, as pointed out by factory managers as one of the leading problems in many laundry businesses in Singapore.

Upon a deep dive into this problem, we've figured out that time delays for laundry movement from point to point on the factory floor is the main culprit behind operational inefficiency after an on-site factory visit at a few laundry companies in Singapore. This is an important issue to tackle as the laundry industry provides essential services to numerous sectors such as airlines, hospitals and hotels which often have tight laundry delivery schedules.

Problem Exploration

According to our latest field research conducted at Systemic Holdings - Singapore's largest laundry company, the greatest inefficiency during operation is the disorganized movement of trolleys within the factory due to a lack of clear, demarcated routes for workers to move them within the floor which wastes their precious time and effort.

The need for manual labor to push the trolley is a waste of precious human capital as workers can boost their productivity if they can focus on a certain production process only. This rationale is based on the theory of specialization - where workers perform better if they are allocated only one task and religiously do it over and over again resulting in higher quality work.

Based on our study, the total time spent on trolley movement is on average **3h18 min**[Appendix]. As the trolley is moved by humans it moves at an average speed of **1.2m/s**[13] based on calculations which creates a small delay as some stations might be idle while waiting for the arrival of the trolley.

---

[1][1]https://www.statista.com/topics/5821/ageing-population-of-singapore/#dossierKeyfigures
[2]https://stats.mom.gov.sg/Pages/Growing_Alongside_Our_Ageing_Workforce.aspx
[3]https://www.singstat.gov.sg/find-data/search-by-theme/population/elderly-youth-and-gender-profile/latest-data
[4]https://www.statista.com/statistics/1112950/singapore-elderly-share-of-employed-residents/
[5]https://www.cnbc.com/2019/08/26/singapores-ong-ye-kung-on-re-training-workers-skillsfuture.html
[6]https://stats.mom.gov.sg/Pages/Singapore-Yearbook-Of-Manpower-Statistics-2021-Labour-Force.aspx
[7]https://www.mom.gov.sg/newsroom/speeches/2016/1109-speech-by-minister-at-texcare-forum-singapore-2016
[8]https://www.channelnewsasia.com/singapore/construction-firms-manpower-shortage-covid-19-support-measures-1356411
[9]https://www.businesstimes.com.sg/government-economy/singapores-population-growth-slowed-to-11-per-year-in-the-last-decade-census
[10]https://www.aseantoday.com/2019/02/seniors-at-work-the-new-norm-for-ageing-in-singapore/
[11]https://www.singstat.gov.sg/modules/infographics/population
[12]https://www.straitstimes.com/singapore/100000-singapore-residents-face-difficulties-performing-basic-activities-population-census
[13]https://www.sciencedirect.com/topics/medicine-and-dentistry/walking-speed

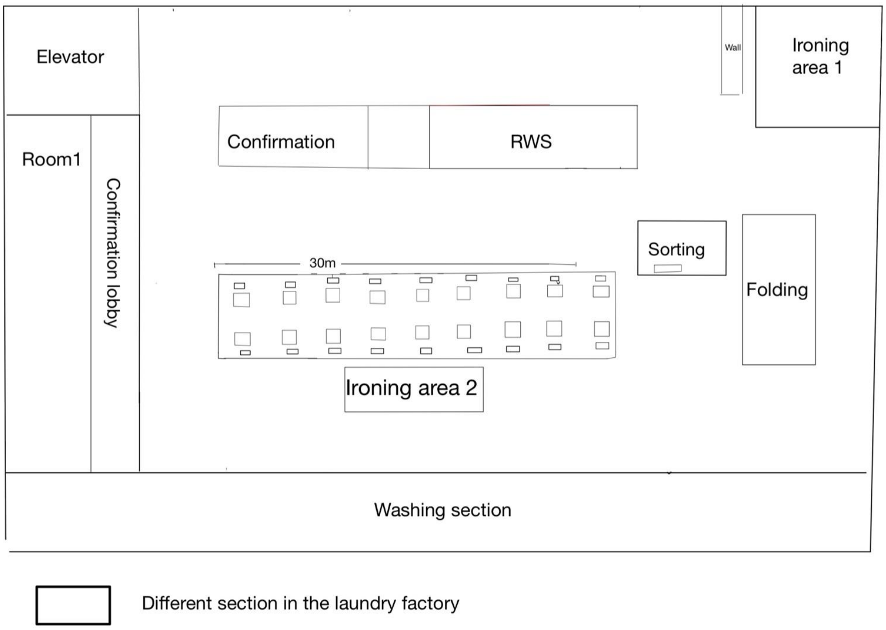


Figure 1: Laundry factory overview

Figure 1 shows the overview of the factory floor. The factory floor consists of different sections, e.g. ironing, sorting separated from each other. In order for the clothes to be delivered to each section, they require a trolley to push the clothes over to each station - from washing to folding for the final laundry to be delivered to the customer.

Problem Statement

1. Laundry throughput is reduced due to time spent pushing trolleys instead of employees focusing on their respective station only where they are specialized in.
2. Bottlenecks caused by lack of optimum trolley allocation leading to disorganized movements within the factory floor, causing unnecessary congestion and time delays.
3. Human walking speeds, given their age factor, limits the speed of trolley movement which increases the time gap in receiving the laundry from the previous station which reduces efficiency.

Design Statement

How might we design a trolley system that is able to enhance the operational efficiency to increase the laundry throughput at the factory that is easy to use and at low manufacturing cost?

Design Specifications

Customer statement	Customer need	Metric	Metric Unit	Range
The walking speed of human limits the speed of trolley movement which increase time gap and reduces efficiency	Increase the speed of the trolley by automating the trolley system	Speed of the trolley	m/s	>1m/s
Bottlenecks caused by disorganized movement of trolleys within the factory floor The need for optimum allocation of trolleys in each station which would be able to optimize the movement of trolleys and increase operational efficiency.	An organised trolley system that is able to optimize the trolley allocation and flow movement that is able to save precious time and effort for the aged employees and focus on lighter tasks, given their age	Time reduction after the trolley system had employed and the percentage of efficiency increase	Time/ percentage of efficiency increase	3-4h/ 10-15% increase in efficiency
Laundry throughput is reduced due to time wasted on pushing trolley instead of people continuously working at their respective station to produce good work	Automating the trolley system to increase throughput	The percentage of output increase per hour	The percentage of output increase per hour	15% - 20% increase in output

Trolleys were improperly used due to lack of proper allocation to respective stations, resulting in delays in different sections of the laundry cleaning process and causing extra fatigue for the already aged workers.

Concept Selection

Robot able to move at the correct direction	**A board that is able to receive input and output of the designed algorithm to ensure maximum efficiency and flow**	**Can be easily detached and attached to different trolleys to ensure greater flexibility of the robot in terms of movement across the factory floor to increase the overall efficiency**	**Able to carry heavy loads so that it is able to operate in a fully autonomous manner without the need for human to push the trolley**	**Ensure workplace safety in the factory as loads tend to be heavy and poses health risks for the elderly majority workforce within the factory**
Line following array which follow the factory floor coloured lines to move	Raspberry Pi as the main motherboard to receive input and move based on input to create the operational flow	Connect the robot and the clamp and use the clamp to stabilize the robot onto the trolley	Use a motor of sufficient torque to move the trolley forwards and backwards, while overcoming basic drag and friction	Usage of a strong braking system with very little braking distance to stop in time in case of an obstruction along the path
Lidar sensor to detect the surroundings. Helps learn familiar places using machine learning and move with extremely high accuracy	Arduino as the main motherboard to receive input and move based on input to create the flow	Tow the trolley	Use a motor of sufficient torque to lift and move	Usage of a HCSRO4 sensor to detect any obstruction and people moving in front to ensure the bot can stop
Pi camera V2 using machine learning to learn and find familiar places which would help the robot move optimally	Linux subsystem computer controlling multiple raspberry pi and Node MCU to move on time , priority and other factors to create the flow	Usage of a velcro to attach itself to a trolley		Usage of lidar similar to tesla to make the bot stop whenever it detects some foreign objects

- The line following array is selected because it is more adaptable to unfamiliar environments compared to others which use machine learning with longer time taken for operation.
- Raspberry Pi is to ensure effective communication between interrelated computing devices which controls the entire trolley movement flow across the factory.
- HCSRO4 sensor is used for obstruction detection purposes as it enables the robot to stop in time when people are moving in front at a certain distance to avoid collisions and possible damage to the bot.

Detail Design

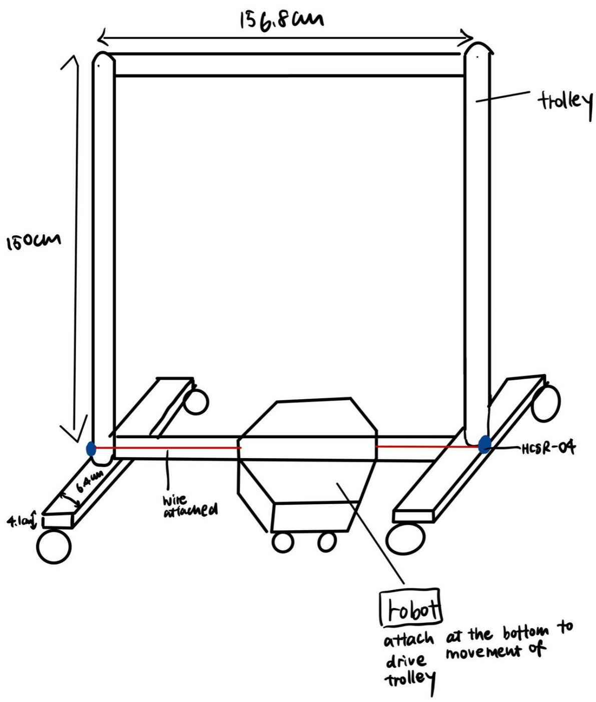


Figure 2: Overview of trolley with bot attached

As shown in figure 2, the body will be attached at the bottom of the trolley to facilitate the movement of the trolley. The bot is placed below the trolley to increase stability.

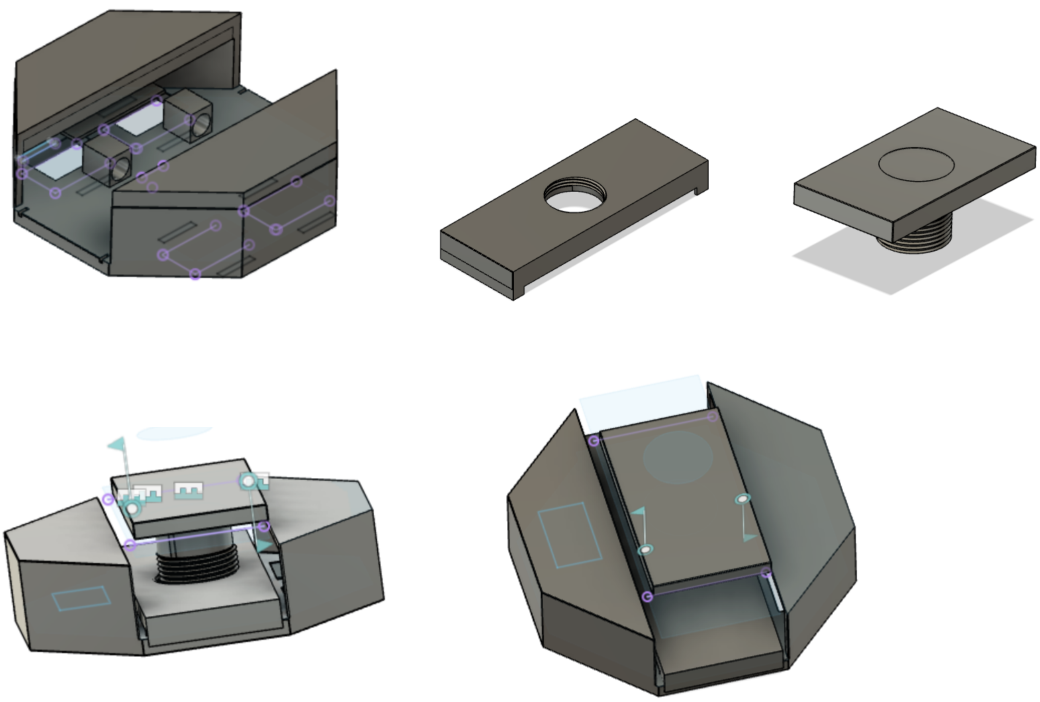


Figure 3: CAD design for the body

Figure 3 shows the detailed CAD design for the body and the cap, the cap will be placed at the top of the body. The function of the cap is to ensure the height is adjustable when attached to the trolley. The cap will also have an electromagnet capable of turning on and off which will attach itself onto the trolley base.

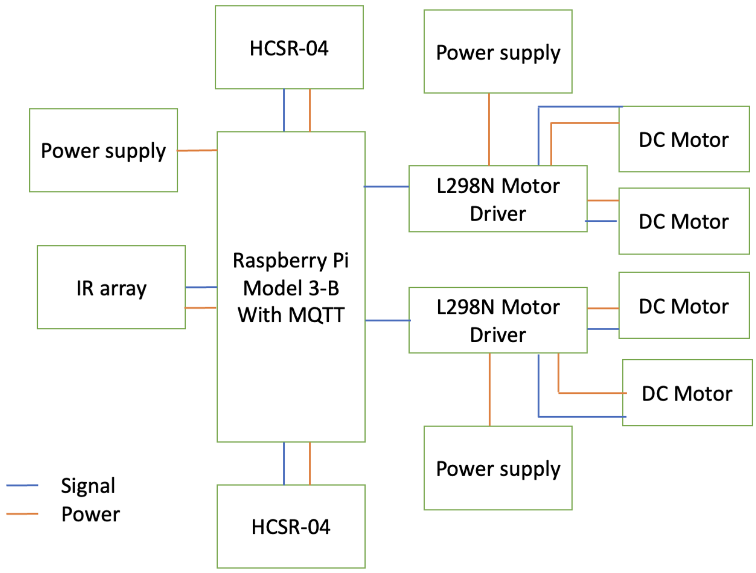


Figure 4: Block diagram of the prototype

Figure 4 shows the main flow of signals that will control the movement of the robot - the blue lines represent signals and red for power i.e voltage/current to control the various electrical components i.e. from sensors to motor drivers. The Raspberry Pi functions as the main microcontroller which will send and receive signals and power to all components which are actuators.

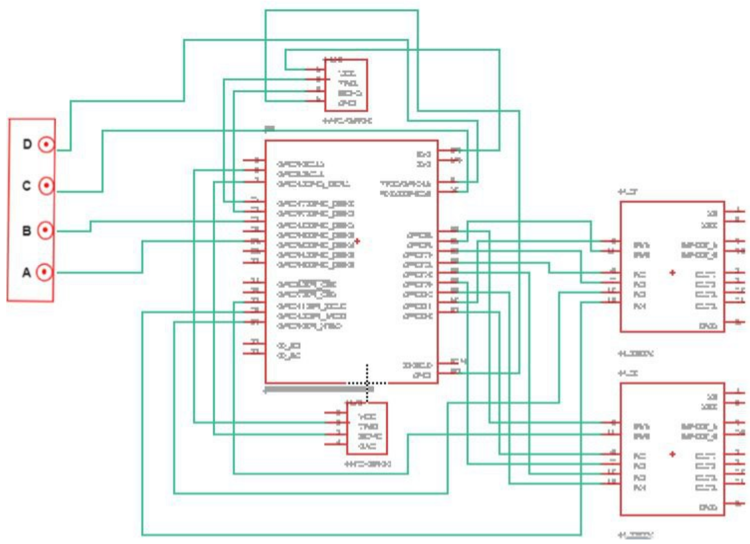


Figure 5: Detailed schematic diagram

Figure 5 shows a detailed diagram of all electrical and power connections to all actuators from the GPIO pins of the Raspberry Pi which controls their functionalities. The centre is the

Pi, while the HCSR04 is at the top and bottom respectively, and the right is the L298N motor driver while the left is the IR follower line array.

Prototype Components

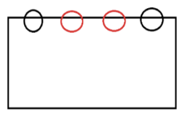
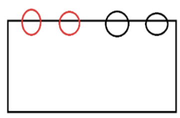
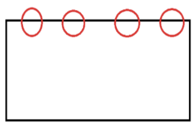
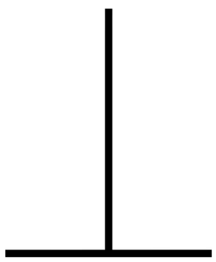
<table>
<tr><td>HCSR04 Ultrasonic Distance Sensor</td><td>The HCSR04 sensor has been used as a safety mechanism to help detect nearby obstacles while the robot is moving with respect to the line. This is important to take note of as obstruction to the path of the robot can result in damage to the prototype and maybe the object or person.</td></tr>
<tr><td>Line Following IR Array</td><td>The line follower array is an important component of the entire system, as it helps the robot to detect the line which it has to follow and hence, guides the robot to move in a particular direction. It has 4 different IR sensors in an horizontal array which will make it more accurate.
<table>
<tr><td>Straight</td><td>Turn</td><td>Stop</td></tr>
<tr><td>The array will keep vertical black line in the 2 middle IR sensors (red circle indicated)</td><td>2 left IR sensor or 2 right IR sensor will sense the line if it encounters turning

Turn will occur if the left IR sensor (2 of the left or right IR sensor senses the black line on the left or right.

The turning intensity will depend on the number of left sensors sensing a black line.</td><td>The array will stop if all 4 IR sensor senses a horizontal black line</td></tr>
<tr><td>IR array shown below each circle representing different individual IR

Red circle indicate IR involved in the path described above</td><td>IR array shown below each circle representing different individual IR

Red circle indicate IR involved in the path described above</td><td>IR array shown below each circle representing different individual IR

Red circle indicate IR involved in the path described above</td></tr>
<tr><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td></tr>
</table>
</td></tr>
<tr><td>L298N Motor Driver</td><td>The L298N motor driver is used to control the differential drive, speed and direction of the motors by switching the current to the front and back motors when Raspberry Pi sends a signal to it. The driver supplies 12V to the motors and the direction of current is switched from forward to reverse direction using the h-bridge circuit embedded within the driver which can control the direction and speed of the 2 dc motors simultaneously.</td></tr>
</table>

Raspberry Pi Model 3 B+	Raspberry Pi Model 3 B+ has been used due to its networking and multi-tasking (MQTT e.t.c) compared to Arduino. It also has a faster clock speed and has a diverse range of pins which different components can connect to. Raspberry pi is a microcontroller which is able to send signals to various components. Moreover, its wireless function enables it to link to the server which is an important part of how it works.
Node MCU ESP8266 With MQTT Configuration	The node MCU was chosen as the wireless controller to send packages to the server via MQTT from input from the remote control to ensure maximum operational flow.
Remote Control & Receiver	The numbers on the remote control allows us to have different functions tied to each number on the controls to better assist the operators.
RS Pro Brushed DC Motor	We have used 4 brushed DC motors which have sufficient torque to move the prototype, based on our detail design calculations[Appendix]. There are 2 of these motors in the sides of the robot where each pair would be controlled by the L298N motor driver which would supply 12-14V to all 4 motors for proper operation. The motors would be powered by TATTU R-Line 4S1P LIPO battery pack with XT30 plug, of 650mAh capacity, 14.8V 95C which would be used for testing purposes. The intended power source is supposed to be higher for the motors to overcome additional resistance forces such as static and dynamic friction upon commencing and maintaining movement. A brushless DC motor with ESC was initially wanted over the brushed dc motor with the motor driver however we were unable to find a small enough brushless DC motor to suit our needs hence we used a Brushed DC motor. It's torque and RPM allows it to achieve a speed of 5km/hr which is the walking speed of a human at its maximum PWM which allows us to validate and test the capability of this machine in increasing efficiency of the factory.

Automated Trolley System

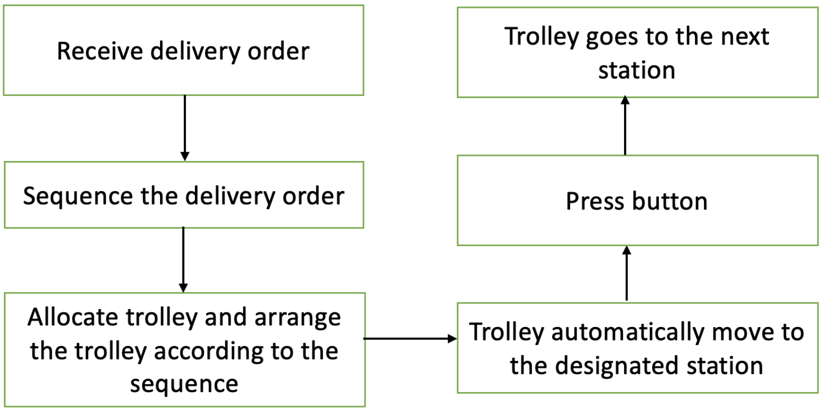
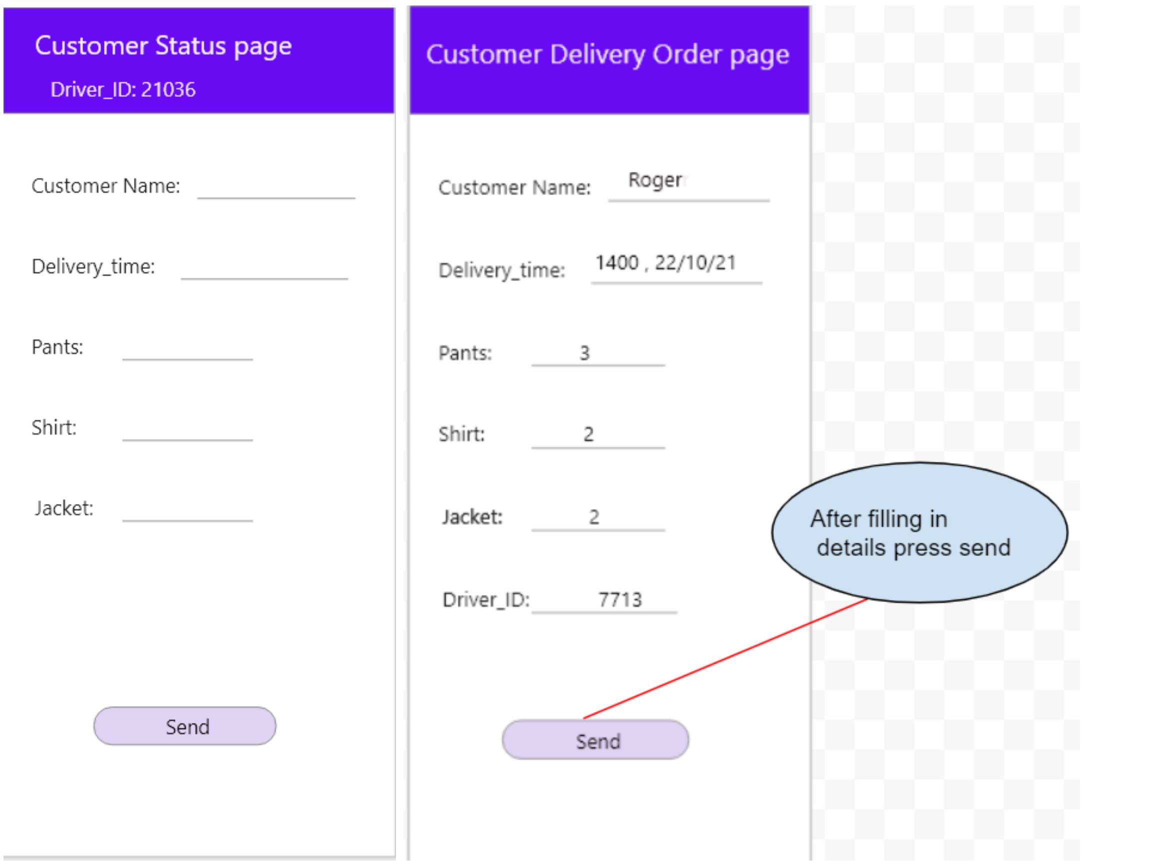


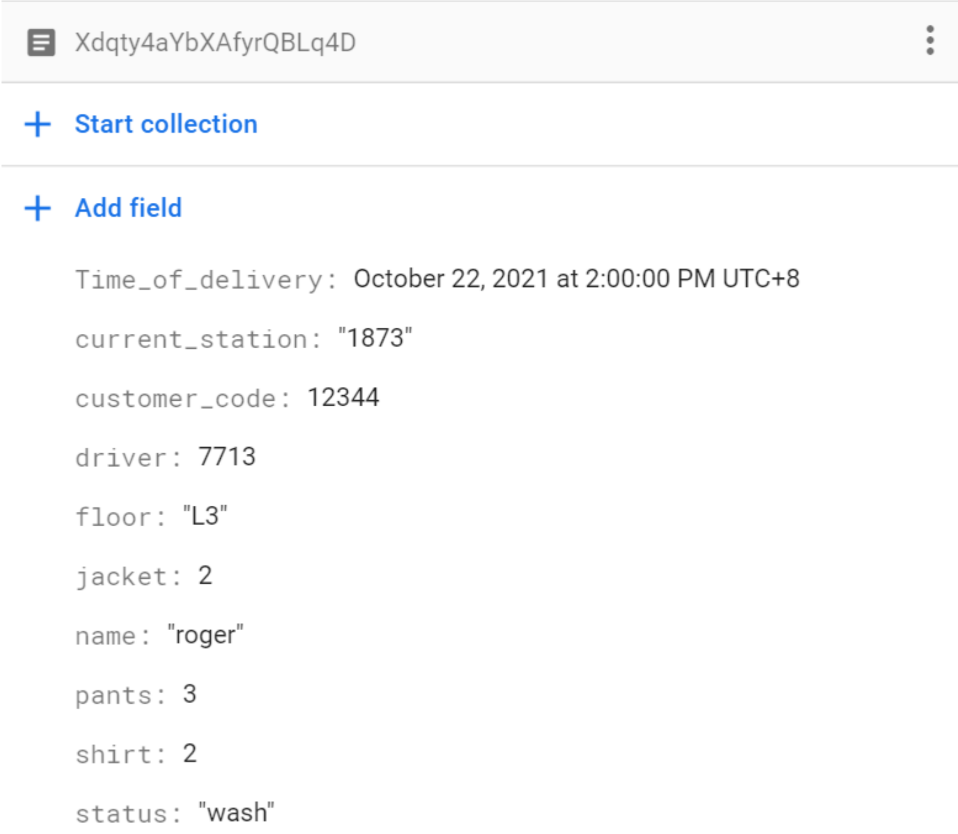
Figure 6: Overview of the automatic trolley system

Every customer will be issued a particular delivery order for the day where they will be sequenced and the order of sequence will be sent to the server. The delivery order will show the delivery timing,the number of clothes or towels and the location where it will be done at. This information will determine the number of trolleys allocated to each station and the order sequence. These steps are shown below in greater detail:

1.Information logging page to get information from the customer

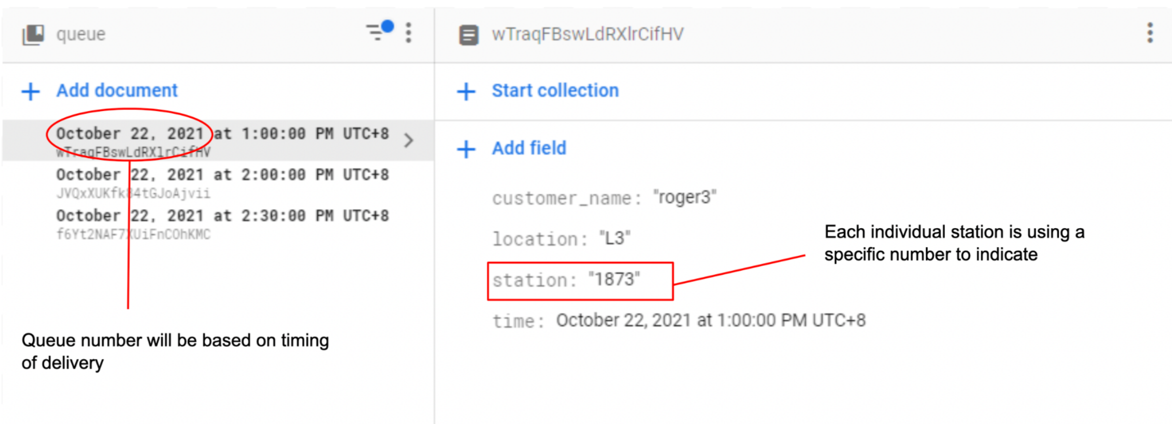


2.Server receives information.

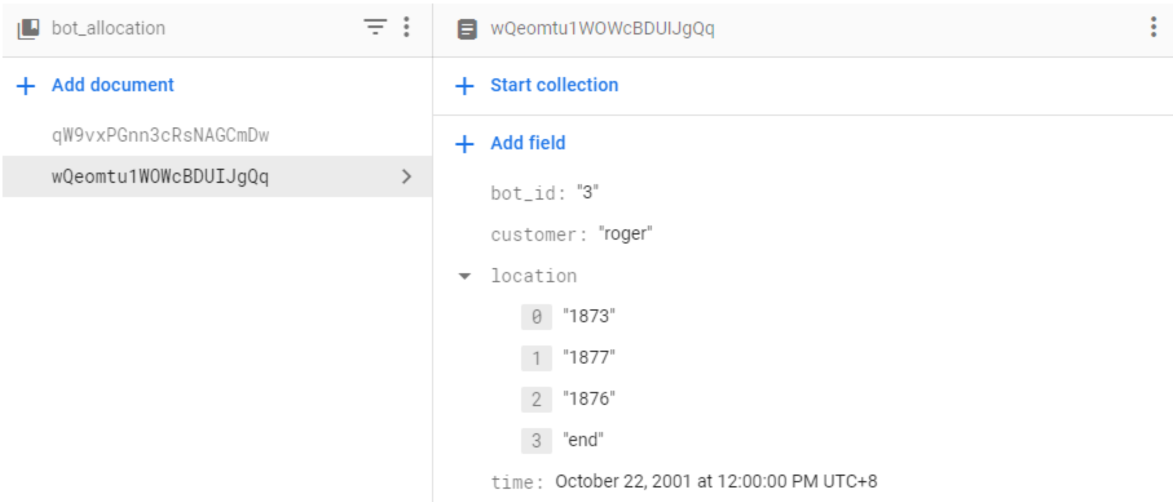


3. Using the information received, Server queues the delivery order by location (floor) and time of delivery using an algorithm. Trolley movement will be based on the queue number.

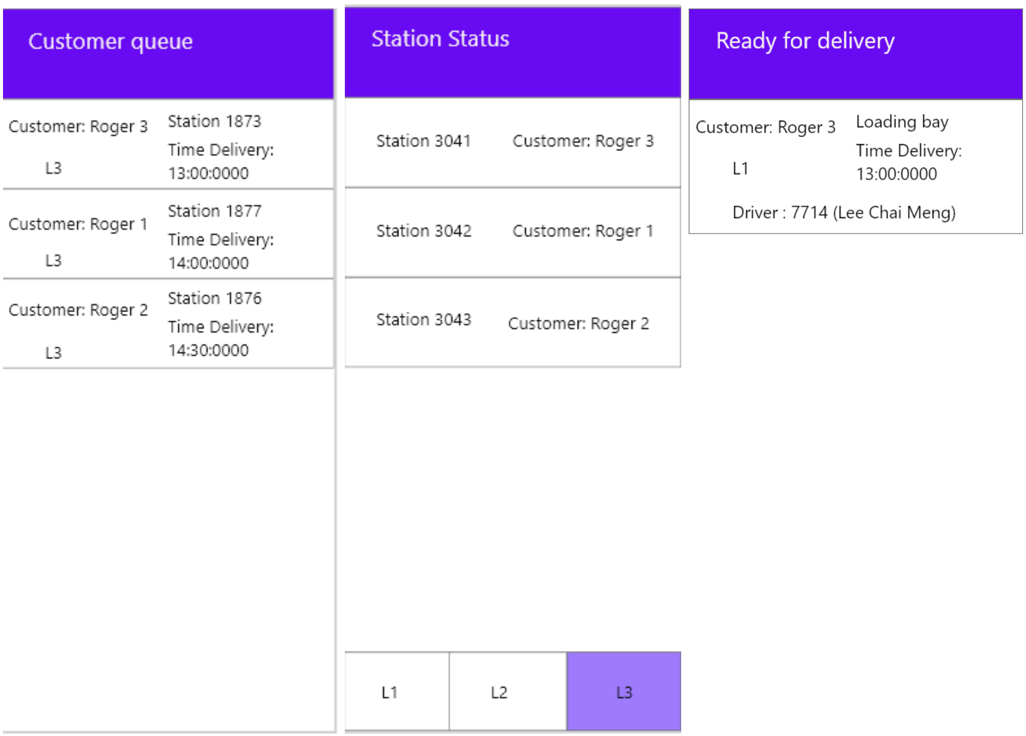
The figure below shows a firebase server indicating the current queue for customers and the location of the trolley at a particular time.

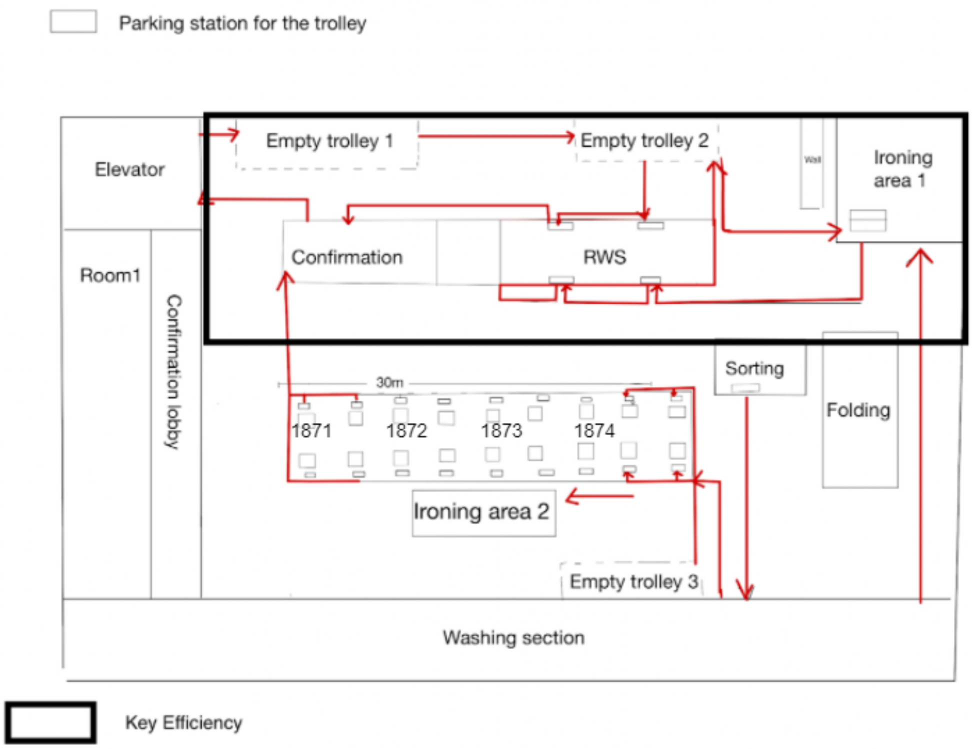


4. Based on the queue, the server will select bots that are ‘free’ and assign them a queue number based on customer and time of movement to a particular station. The trolley will move based on the assigned route, from station “1873” to “1876” and will move to the end location where it will be loaded and delivered”.



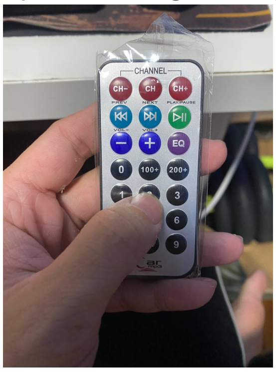
5. Managers can check the status of each customer with the app and they can check each station by level as well to see which customer they are serving to see whether everything is flowing well.



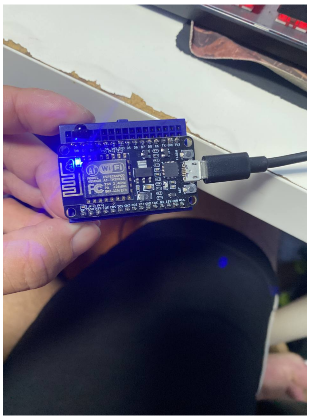


A single trolley parking lot (empty trolley 1,2,3) belongs to 2 to 3 stations, during operation, the server will send a message to the Pi and the trolley will then move to the designated station automatically with the sequence tape on the trolley to notify the worker on the current laundry sequence. Based on the sequence, the worker will be able to know which laundry they are supposed to put on the trolley. When the worker is done with their respective task, they will press on the “Done” button on the remote which will notify the server via the ESP8266 using MQTT that the trolley is moving to the next station.Below shows the basic working principle of this technology:

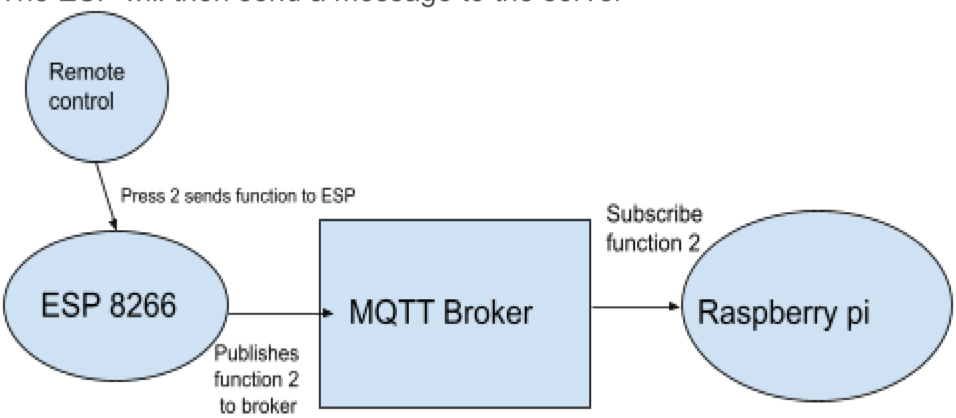
1. Upon finishing laundry press '2' on the remote



2. The remote will send a signal to the ESP8266



3. The ESP will then send a message to the server



Example of a message sent

```
void loop() {
 if (!client.connected())  // Reconnect if connection is lost
 {
  reconnect();
 }
 client.loop();
```

```
// if the remote is being pressed
if(digitalRead(SWITCH_PIN) == 0)
{
  state = !state; //toggle state
  if(state == 1) // ON
  {
    client.publish(Function, "on");
    Serial.println((String)Function + " => on");
  }
  else // OFF
  {
    client.publish(Function, "off");
    Serial.println((String)Function + " => off");
  }

  while(digitalRead(SWITCH_PIN) == 0) // Wait for switch to be released
  {
    // Let the ESP handle some behind the scenes stuff if it needs to
    yield();
    delay(20);
  }
}
}
```

Message sent to server:
Function2, ‘on’

Example of a message received:

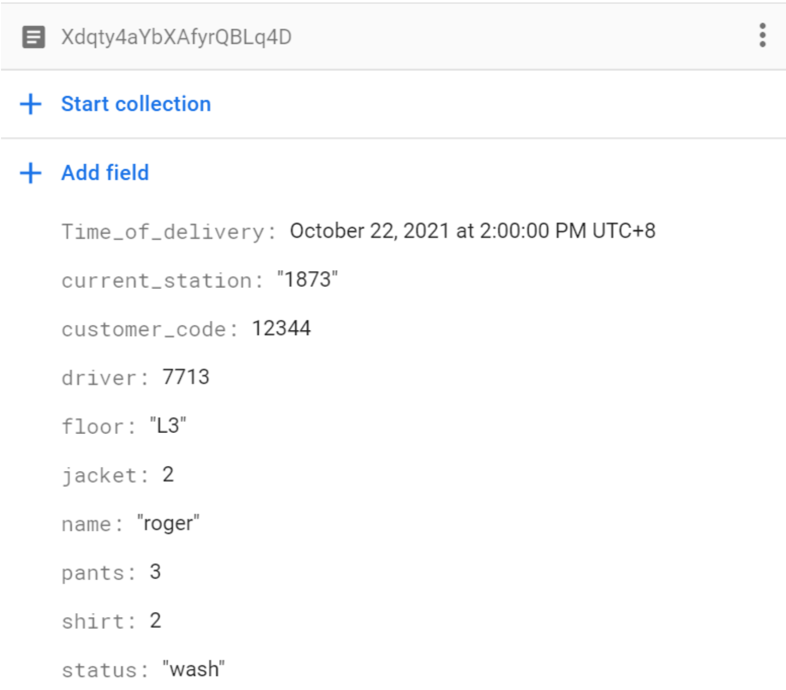
```
while (!client.connected()) {
  Serial.print("Attempting MQTT connection...");
  // Attempt to connect
  if(client.connect(ID)) {
    client.subscribe(Function);
    Serial.println("connected");
    Serial.print("Subcribed to: ");
    Serial.println(Function);
    Serial.println('\n');

  } else {
    Serial.println(" try again in 5 seconds");
    // Wait 5 seconds before retrying
    delay(5000);
  }
 }
}

Serial.print("Message arrived [");
 Serial.print(topic);
 Serial.print("] ");
 Serial.println(response);
 if(response == "on")  //Move bot to next location in array
 {
   AIOclient.publish("bot"+pi.id, Changelocation)
   Stationno = firestore.collection(‘bot_allocation’).doc(customer_id.uid).snapshot.data(locations)[n+1]
   firestore.collection(‘laundry’).doc(customer_id.uid).update({station:stationno})}
   client.publish(STATE_Function,"on");

 }
}
```

4. The pi receives the message and moves to next location and updates the server



5.

After one trolley is completed, another trolley will come to the station to ensure a continuous, smooth flow of laundry operations within the factory floor with threefold advantages:

- ➔ Aged workers would no longer need to push the heavy trolleys which saves up time and effort for them to focus on their tasks assigned to them for greater productivity
- ➔ Less workplace hazards for the workers as injuries might occur if they carry heavy laundry
- ➔ Lower number of “idle” trolleys hanging around the factory floor which results in smoother factory operations and faster delivery of laundry to targeted customers

At the final station, when the worker presses the button, the server will be notified that the order is done and the delivery man will come to pick it up. After pick up, the trolley will move back to the parking lots.

The system uses a priority queue to allocate the trolley which would address the disorganised trolley flow movement. It helps to provide an optimum trolley allocation to each station to increase efficiency for better laundry operations. To better implement our automated trolley solution, we designed a customized route where the trolley will travel within the factory floor shown in figure 7.

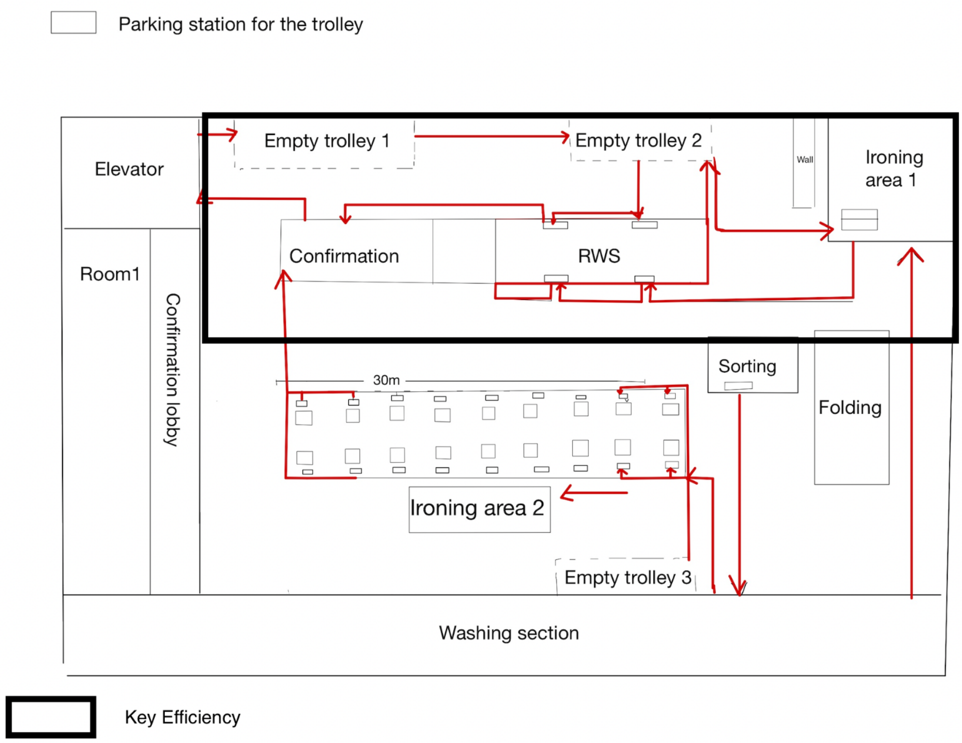


Figure 7: The design route for the moving trolley with the attached robot

Note: Key efficiency for the above diagram refers to how the trolley routes are designed to be more efficient in the confined stations, the design only includes conformation, RWS and ironing area 1.

This strategy will help increase the efficiency of laundry operations flow and hence, save precious time. This method has been implemented as it was discovered that there was no existing route design within the factory which would govern the movement of the trolley, and no demarcated lanes would mean that workers would not know which "best" quick path to take the laundry to each respective station.

The customized design could also help avoid collisions using the Kaizen principle. The red line as shown in Figure 7 on the left indicates the future movement of the trolley using our implemented system to ensure a smoother operations flow for laundry businesses. Our prototype will follow the same route as shown above using line tracking methods.

The design of the route will mainly focus on the RWS, confirmation and the ironing sections as shown in Figure 7. Firstly, the trolley will move from the ironing area 1 to the RWS as shown in the map on Figure 7. In RWS there are two trolley parking stations because we observed that the worker will often park at these two locations i.e. location 1 and 2 to sort the clothes. Initially, to push the trolley from location 1 to location 2 in RWS, the worker would need to travel back and forth multiple times to sort the clothes which wastes time.

The new route will allow the trolley to move from ironing section 1 to RWS location 1 directly to distribute clothes belonging to location 1, and it will move to location 2 for workers to sort the remaining clothes. This allows the trolley to travel one time in one direction which saves time and increases efficiency. After the clothes are distributed at the RWS station, the trolley will be emptied and moved to the Empty trolley 2 parking area. The empty trolley 2 parking site is catered to both RWS and ironing area 1.

Empty trolley 1 is the parking area for the trolleys that have been returned, and it will be distributed and rejoin the trolley system - it would connect to empty trolley 2 parking station since empty trolley 2 will be used very frequently by the ironing and RWS station.

Testing Requirements

For the testing requirements, we would need to measure the following which would aid us in the construction of our prototype:

1. Movement and flow of the factory - how our new movement and floor plan can better help the factory reduce total walking distance compared to current
2. Load test and movement test with empty trolley and full trolley
3. Speed of trolley
4. Safety when moving - stopping distance etc.
5. Algorithm test of whether MQTT works within this isolated network

Figure 8: Bot prototype

Figure 9: Factory testing

Test results

Load Test

Load test	Load	Does the robot move?	Can robot follow the line and turn well?
1	Empty	yes	yes
2	Half full (30kg)	yes	yes , ~85% of the time
3	Fully filled trolley(60kg)	yes	Yes, ~70% of the time

Speed of trolley

A) For Empty Load:

Trolley	Speed (m/s)	Path	Distance(m)	Comments
Empty	2m/1.56s = 1.28m/s	Straight	0-2m (PWM 40)	Very straight but a little slower than human walking speed at 1.2m/s
Empty	3m/2.1s = 1.43m/s	Straight	2-5m (PWM60)	Quite fast relatively straight
Empty	5m/2.46s =	Straight	5-10m	Very very fast and very

	2.03m/s		(PWN 100)	straight line to follow otherwise will overshoot a little and will have to backtrack a little
Empty	2m/2.4s = 0.833m/s	Left Turn	0-2m (PWM 30)	Turns well but tape must be placed at a consistent angle throughout the track to turn smoothly(we used 30 degree))
Empty	2m/2.5s = 0.8m/s	Right Turn	0-2m (PWM 30)	Turns well but tape must be placed at a consistent angle throughout the track to turn smoothly(we used 30 degree)

<u>B) For Full Load:</u>

Trolley	Speed	Path	Distance
Full	2m/3.89s = 0.514m/s	0-2m (PWM 40)	Very straight but a little slower than human walking speed at 1.2m/s
Full	3m/4.2s = 0.714m/s	2-5m (PWM60)	Quite fast relatively straight
Full	5m/4.7s = 1.06 m/s	5-10m (PWN 100)	Trolley
Full	2m/4.2s = 0.476m/s	0-2m (PWM 30)	Turns well but can be improved as wheels of external trolley might hinder movement while turning and delay the bot a little at times
Full	2m/4.2 = 0.472m/s	0-2m (PWM 30)	Turns well but can be improved as wheels of external trolley might hinder movement while turning and delay the bot a little at times

<u>Movement Analysis:</u>

Timing above are best timing for movement of path but this is a analysis of the multiple test conducted

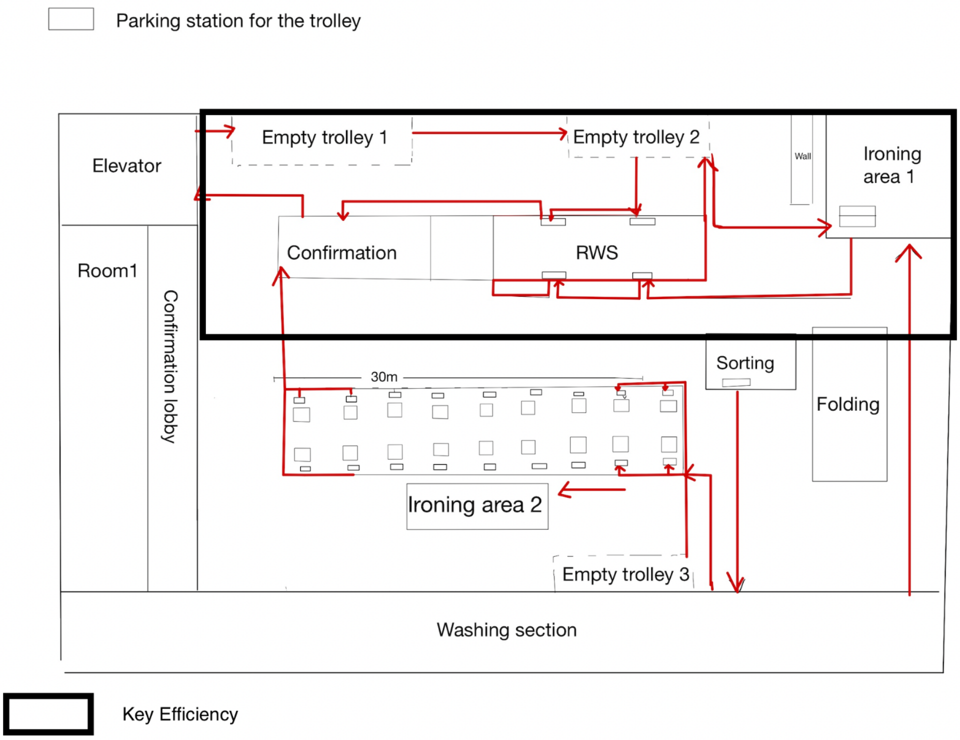
<u>Straight</u>	<u>Directional Turn</u>
PWM 40- Movement of trolley is steady and on par with human walking speed for this application it will be good to be used at short distances and tendencies to overshoot is very low However there are some instances that the bot	PWM 55:15(left or right motor with 55 PWN and the other 15 PWN e.g. turn right left PWN 55 right PWN 15)- Turn is at a close to 30 degree angle and has to be gradual and not sharp Sharp angles might make the array lose sight of line

might veer off track as the trolley wheels of external trolleys are not aligned at the start which means there will be a additional device needed to set the direction of external trolley wheels as some wheels are very loose due to prolonged usage For trolleys with damaged wheels the bot will need dampaners to reduce damage to wheel axis on the bot but it can guide the trolley properly but might have to veer left and right at times	and will have to back track again to find the line again Misses lines at times but at a lower rate compared to higher speed turns similar to a car having better control when turning slower Damaged wheels trolley makes the bot lose sight of line ~40 percent of the time as the wheels are not aligned well and might veer or bounce the robot off track Full trolley turns have a risk of toppling (experimented but hasn't fallen yet but looks like it might fall) when executing sharp turns therefore a gradual turn over 30 degrees is recommended over sharp turns
PWM 60 - Trolley moves faster than human pace when empty and slightly faster when full Tends to overshoot <1m if the path is short and turns next Stopping distance < 0.1m(max) Great for longer distance movement but turns has to be gradual turns and a little lead has to be given if in this setting But if reaches end line a straight black line it stops very well and within the line	PWM 60:20 This speed is not bad for turning however sometimes the array will mistaken the turn to be stop line as it drift a little due to the external trolley losing grip and dragging the bot along It also overshoots at some tests around ~10 percent Not as optimal and smooth as 55:5 PWM and
PWM 100- Really fast and very optimal for long distances which can be applied for RWS path, washing to ironing path and other straight line path with more than 20 m of travelling However drawbacks are it misses and overshoot quite frequently it's braking distance is higher than the rest but not dangering anyone with the help of HCSR-04 but it might scare some of the workers as seen during testing therefore a lowering to around 70 pwm is recommended as they are less scared of a running trolley around that speed But it's a good test of max speed of trolley and can be experimented further for level one operation with a empty ground where lesser workers are present for now 100 PWM is restricted at lvl3	PWM 70:40 Bad turn as clothes looks like it might topple but it hasn't toppled once Turn is overshot frequently ~60 percent Might put a little too much stress on the wheels as it drifts a little and might stop as it might see it as a horizontal line Definitely not recommended

Evaluation

After going through multiple tests of different PWM in the factory we have a lot of improvement to be done as there are some things which can be added to improve the design ,however it validates the load it can carry, the speed it can travel, the turns that it can execute and validates the messaging algorithm which is discussed further down.
When going on high PWM we should make it slow down nearing stop line or turn using FSM or Encoders and mapping however due to time constraint these will be implemented in future

Based on speed this is amount of time reduced in movement when using this prototype:



Path	Time taken to move along the path-Human	Time taken move along the path-Robot
RWS path	1 min 45 second	1 min 29 sec
Ironing section path	1 min 15 sec	1 min 31 sec (require more turning)
Washing section path	1 min to RWS 30 seconds to Ironing	40 seconds to RWS (Straight path) 41 seconds to ironing

Safety Analysis:

<u>**PWM of motor**</u>	<u>**Path**</u>	<u>**Braking distance**</u>	<u>**Range when detected**</u>
25	Straight	0.3cm	4.98cm
25	Left	0.1cm	4.99cm
25	Right	0.2cm	4.98cm
50	Straight	0.5cm	4.80cm
100	Straight	0.7cm	4.93cm

System & Battery Analysis:

Our 4 motors, RS Pro DC Brushed Gear type has an individual rating of 41.3W and operates from 12-15V with current rating of 0.99A. 1 battery will be connected to 2 motors, so we have 2 batteries connected to 4 motors.

By calculation 4 motors will consume a maximum of 4A. Based on analysis done in the iDP lab, our entire system consumed 14.8V and 3.2A of current for 4 motors, which meets the max current threshold and voltage requirements.

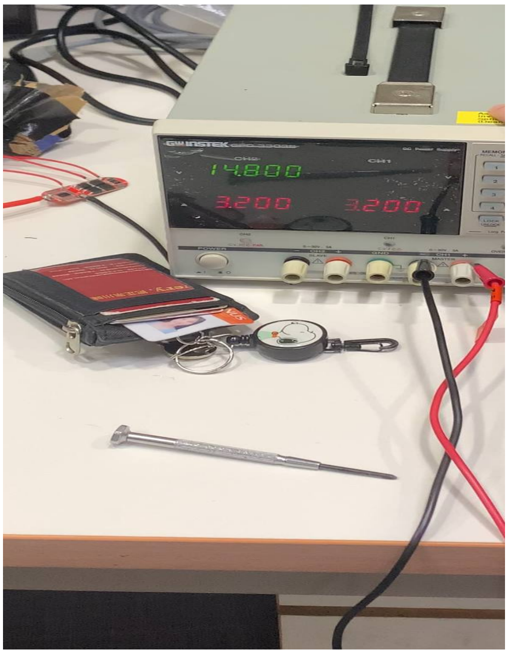


Figure 10: Power testing

The motors would be powered by TATTU R-Line 4S1P LIPO battery pack with XT30 plug, of 650mAh capacity, 14.8V 95C. As we have 2 batteries connected in parallel, our total battery capacity would be 650 x 2 which will be 1.3Ah which will power our 4 motors. This power source is just for testing the actual power source will be one with a higher capacity.

Total Energy Supplied	Capacity (amp/hr) x Voltage = 14.8V x 190C = **2812J**
Total Power Consumed By 4 Motors	Power rating of motor x 4 = 41.3W x 4 = **165.2W**
Time Taken For Battery To Last	Energy/Power = 2812J/165.2W=**17h**

Algorithm Test:

The test demonstrates Raspberry Pi and ESP8266 Node MCU are connected.

Raspberry Pi	**Message sent**	**Receive status**
Message when pi reaches station	`AIOclient.publish("bot"+pi.id, updateLocation)`	200
Message when pi leaves station to next ESP8266	`AIOclient.publish("bot"+pi.id, Changelocation)`	200

ESP8266 Node MCU	**Message U sent**	**Receive Status**
Message when pi arrives	AIOclient.publish("bot"+esp.id, updateLocation)	200
Message when laundry ready for next station	AIOclient.publish("bot"+esp.id, Changelocation)	200

Results:
Algorithm tests for MQTT successful using a personal hotspot environment where the network is isolated need a few firewalls and permission settings to ensure no one can tap into the network and use the ip address and id of each bot as it is easily exploitable.

API TEST Simple App(Beta Version):

From DO to app to server	**One station to another after pressing done to check status change**	**To server when last station from raspberry pi to inform driver**

Laundry{
Customer_code:12455
Customer_name:'roger towel'
Time of delivery:1450
Driver_code:5556
Number_of_shirt:2
Number_of_pants:3
Number_of_jacket:4
Status:Washing
Floor_location:L3
Location_of_payload: #8173
}
	Laundry{
	Customer_code:12455
	Customer_name:'roger towel'
	Time of delivery:1450
	Driver_code:5556
	Number_of_shirt:2
	Number_of_pants:3
	Number_of_jacket:4
	Status:Ironing
	Floor_location:L3
	Location_of_payload:
	#7123
	}
		Laundry{
		Customer_code:12455
		Customer_name:'roger towel'
		Time of delivery:1450
		Driver_code:5556
		Number_of_shirt:2
		Number_of_pants:3
		Number_of_jacket:4
		Status:'At #6266 ready for delivery'
		Floor_location:L1
		Location_of_payload:#6266
		}

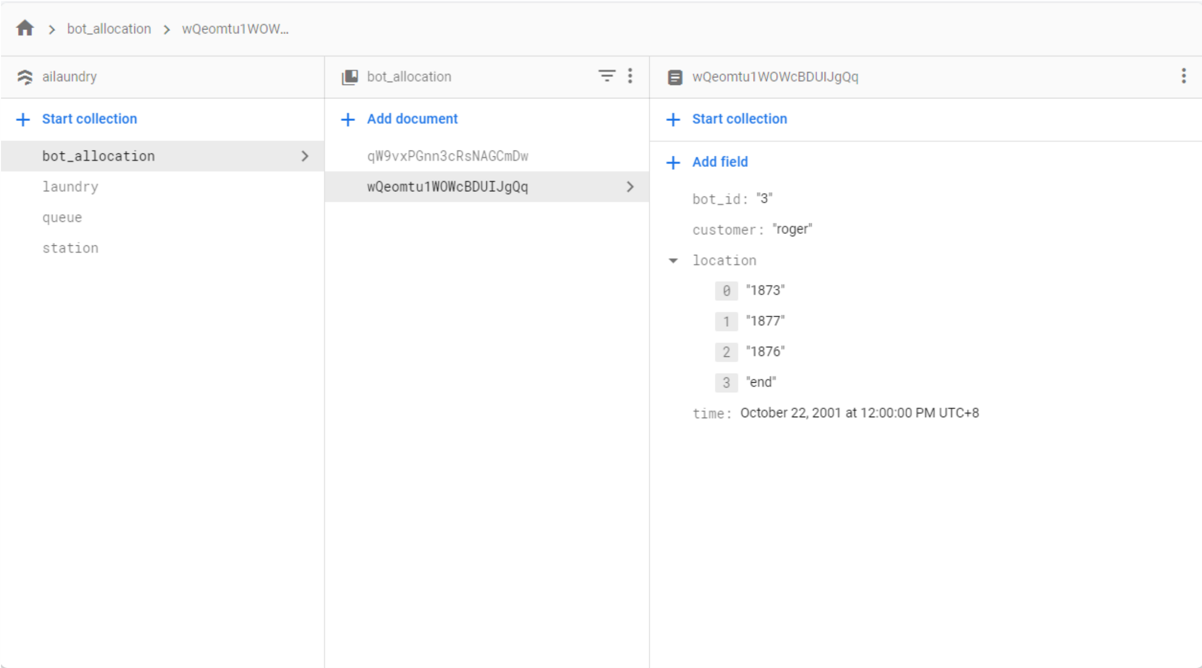


Figure 11: Sample of Delivery order added to API test

Results:

API test to test on some random DO it has been successful in registering itself in an isolated test on firebase in a personal server within personal hotspot

It has also successfully filtered and arrange the order in delivery sequence(`.collection("queue")`
`.orderBy("time", "asc")`
)

It can also allocate bots which are free (location empty array means free or at end means free)

It can update location of bot in the server well on press from remote control

Potential:
More parameters needed from API to better assist the factory in other operations as specified by the chief engineer.
Machine learning can be applied to DO for estimating future output
Machine learning can also be applied to use it to translate a physical DO to an electronic one as now it has to be manually typed into a computer from a driver's picture.

Validation

The key efficiency indicator here would be that if we were able to remove 120 trips (1 trip per trolley) per day, we can have an 10 - 15% increase in sorting levels of the laundry and in order to validate this, we would need to check reduction in delivery timing or reduction in preparation timing and number of output increase due to the timing saved.

Based on the average best timings and 100 PWM on straight path

Path	AVG. Time taken to move along the path-Human	AVG. best Time taken move along the path-Robot
RWS + Ironing + conformation + elevator	1 min 45 second	1 min 29 sec

Based on isolated testing result

- Avg. output per hour RWS: ***60 pieces of clothes/h***

(min. Output:20 max output:120)

- Avg. output per hour Ironing: ***60 pieces of clothes/h***
- ***Avg. output per hour conformation: 60 pieces of clothes/h***

- Number of trolleys in transit from RWS to conformation = ***8 per hour***
- Amount of time saved for using a robot instead of human per trip= ***16s per trolley***
- Amount of time saved = 16s * 8 = 128s = ***2min 8 sec***
- Overall percentage of time saved each trip: 128/840 = ***15.2%***
- ***Throughput increase from time saving = 60/3600 * 128 = 2 clothes per hour***

- Time saved for worker to moving trolley in one trip: 1min 45s =***105s***
- Number of trip per hour = ***8 trips***
- T***otal amount of time saved for worker to move trolley in 1 hour = 105*8 =840s***

- Number of trip increase due to higher throughput, output increase in RWS per hour= 60/3600*840= ***14***
- Output in Ironing per hour = 60/3600*840 = ***14***
- Output in conformation per hour = 60/3600*840 = ***14***
- Percentage increase in output = 14/60*100= ***23%***
- Number of trips increase due to higher throughput produce = ***14/15=1***
- Time taken for one trip = 1min 29s=***89s***

- Net amount of time save = Amount of time saved for worker to move trolley in 1 hour=Time taken for one trip = 840-89 =***751s***
- New output after taken into account into increase number of trip:
- New Output increase in RWS per hour = 60/3600*751=***12***
- New Output in Ironing per hour = 60/3600*751=***12***
- New Output in conformation per hour = 60/3600*751=***12***

- Percentage increase in throughput by reducing manpower in pushing trolley= 12/60*100=***20%***
- Overall efficiency increase = Percentage of throughput increase = 12/60*100=***20% increase in overall throughput***

This is the optimum condition if workers don't get tired and their speed of doing work would not deteriorate.

<u>Conclusion:</u>

Based on our testing results, we proved that our system is capable of increasing the laundry throughput significantly through factory movement automation. This is what the factory needs as this reduces time spent for movement and increases efficiency which satisfies our problem statement identified.

In our journey, we have designed a new map for trolley movement and storage which solves the bottleneck caused by disorganized trolley movement mentioned in our problem statement. This has been done through continuous studies of trolley movement in the factory and based on that, we crafted a design path for the trolley using design thinking techniques which has proven useful for the factory manager as the design path allows trolleys to move in a more organized manner..

We have successfully validated our bot requirements i.e line following capabilities, able to move loaded trolley, and is fully autonomous.The validations show that bot is capable of substituting the mundane task of trolley movement which in turns allows employees who are currently pushing the trolleys around to be more focused in their own work.

Furthermore, we have addressed our aims by validating that our robot is indeed faster than human in certain paths. With that speed, it decreases time in motion for the laundry which in turn, decreases the time for the laundry process. Even though the time saved is only 16 seconds for the RWS path, we are confident that we can improve this efficiency by using a faster motor and better design.

Lastly, the server controlling the bot to streamline around the factory smoothly according to the delivery order is a good way to avoid delays, make the trolley system more organised and smooth. The validation has shown that the server is capable of communicating with one another to allow us to control the bots moving around, and the queue system is also able to work using the algorithm.

All the above-mentioned descriptions are the components that make the system fully autonomous. This concept has proven that it can increase throughput of RWS by 20 percent just by removing the mundane task of movement in the factory and has the potential to increase throughput throughout the factory.

Thus, to conclude, we have demonstrated results in the usage of AGV with IoT in the laundromat industry to automate a mundane and tiring tasks i.e the movement of laundry and we believe this project has the capabilities to be one of the key that allows factories throughout singapore to be more efficient and increase throughput since most of them uses trolley to transit their load in the factory.

Word Count: 2948

Bibliography

[1]“Walking Speed - an overview | ScienceDirect Topics,” *www.sciencedirect.com*. https://www.sciencedirect.com/topics/medicine-and-dentistry/walking-speed.

[2]“Laundry Care - Singapore | Statista Market Forecast,” *Statista*. https://www.statista.com/outlook/cmo/home-laundry-care/laundry-care/singapore (accessed Nov. 06, 2021).

[3]“Laundry Care in Singapore,” *Euromonitor*. https://www.euromonitor.com/laundry-care-in-singapore/report (accessed Nov. 06, 2021).

[4]R. Hirschmann, “Singapore population by age | Statista,” *Statista*, 2018. https://www.statista.com/statistics/624913/singapore-population-by-age-group/.

[5]“$name,” *Base*, 2014. https://www.singstat.gov.sg/modules/infographics/population.

[6]“Elderly, Youth and Gender Profile - Latest Data,” *Base*. https://www.singstat.gov.sg/find-data/search-by-theme/population/elderly-youth-and-gender-profile/latest-data.

[7]*Business Times.com.sg*, 2021. https://www.businesstimes.com.sg/government-economy/singapores-population-growth-slowed-to-11-per-year-in-the-last-decade-census (accessed Nov. 06, 2021).

[8]M. Menon, “100,000 Singapore residents face difficulties performing basic activities: Population census,” *The Straits Times*, Singapore, Jun. 18, 2021.

[9]T. text provides general information S. assumes no liability for the information given being complete or correct D. to varying update cycles and S. C. D. M. up-to-Date D. T. R. in the Text, “Topic: Aging population of Singapore,” *www.statista.com*. https://www.statista.com/topics/5821/ageing-population-of-singapore/.

[10]“Singapore reviewing if more can be done to help firms hit by labour crunch, entry curbs: Tan See Leng,” *CNA*. https://www.channelnewsasia.com/singapore/construction-firms-manpower-shortage-covid-19-support-measures-1356411 (accessed Nov. 06, 2021).

[11]Y. T. Choo, “Construction firms to get help adopting robotics, automation amid manpower crunch,” *The Straits Times*, Singapore, Sep. 09, 2021.

[12]*Business Times.com.sg*, 2021. https://www.businesstimes.com.sg/real-estate/developers-contractors-battle-labour-crunch-supply-shortage-to-meet-deadlines (accessed Nov. 06, 2021).

[13]“0312 Jobs Situation Report - 19th Edition,” *Ministry of Manpower Singapore*. https://www.mom.gov.sg/newsroom/press-releases/2021/0312-jobs-situation-report---19th-edition (accessed Nov. 06, 2021).

[14]“Article: Growing Alongside Our Ageing Workforce,” *stats.mom.gov.sg*. https://stats.mom.gov.sg/Pages/Growing_Alongside_Our_Ageing_Workforce.aspx.

[15]"Topic: Aging population of Singapore," *Statista*. https://www.statista.com/topics/5821/ageing-population-of-singapore/#dossierKeyfigures (accessed Nov. 06, 2021).

[16]"Singapore Yearbook Of Manpower Statistics 2021: Labour Force Table(s)," *stats.mom.gov.sg*. https://stats.mom.gov.sg/Pages/Singapore-Yearbook-Of-Manpower-Statistics-2021-Labour-Force.aspx (accessed Nov. 06, 2021).

[17]"Singapore: elderly share of employed residents 1970-2020," *Statista*. https://www.statista.com/statistics/1112950/singapore-elderly-share-of-employed-residents/.

[18]M. Liew, "Seniors at work: The new norm for ageing in Singapore | ASEAN Today," *Aseantoday.com*, Feb. 19, 2019. https://www.aseantoday.com/2019/02/seniors-at-work-the-new-norm-for-ageing-in-singapore/.

[19]B. Lim, "Singapore braces for challenges in the job market," *CNBC*, Aug. 26, 2019. https://www.cnbc.com/2019/08/26/singapores-ong-ye-kung-on-re-training-workers-skillsfuture.html.

[20]S. C. in Singapore, "Singapore's 'silver tsunami': how the city-state depends on its elderly workforce," *the Guardian*, Dec. 06, 2016. https://www.theguardian.com/cities/2016/dec/06/singapore-silver-tsunami-city-state-elderly-workforce-ageing-cities (accessed Nov. 06, 2021).

[21]"Singapore: Case Study for Asia's Aging Workforce," *Brink – The Edge of Risk*, Oct. 24, 2017. https://www.brinknews.com/singapore-aging-workforce/.

[22]T. Aw, "Opinion | Aging, but Still Working, in Singapore," *The New York Times*, Jun. 07, 2016.

[23]"1109 Speech by Minister at Texcare Forum Singapore 2016," *Ministry of Manpower Singapore*. https://www.mom.gov.sg/newsroom/speeches/2016/1109-speech-by-minister-at-texcare-forum-singapore-2016 (accessed Nov. 06, 2021).

Appendix A

Manual movement of trolley

The average time taken for 1 trip for the trolley from station to station ranges from **45sec - 1min 30 sec**.The exact number of trolleys observed transitioning within the factory floor **from 2pm - 4pm** is 74 based on a study in the factory.
We take the human walking speed as 1 minute to work across the stations, the total time taken during the afternoon is 74 * 1 = **74 min (1hr 14min).**
The factory operates 8 hours, therefore when we extrapolate, 8/3* 1h14min = 3h18 min which is the time spent on the manual movement of the trolley.

Motor calculation

Average load of trolley + laundry = 60 kg

Robot Body weight = 5kg

We take frictional coefficient of the floor as n=0.2

Radius of wheel = 0.04m

Total Frictional force= normal force *frictional coefficient = 0.2x9.81*65=127.53

Total Torque needed for movement= 127.53x0.04=5.1012Nm

Motor Specification:

Voltage=12V

speed =431 rmp

Speed of the motor = $4.31/60 \times 2\pi \times 0.04$=1.8m/s

Speed of the motor is higher than avg. human walking speed @1.2m/s

Max torque of motor = 2.4Nm

70% efficiency @rated tolerance torque = 1.7Nm (From spec sheet)

No. of motor needed to run = 5.1012/1.77 = **4 motors approximately**