

MEDICAID

Smart Solutions for Safer Healthcare Waste

Alen Daniel, Won Lee, Anishvaran Manohar, and Luciano Zarzuela



Problem Statement (Ref: Pages 5-10)

According to the World Health Organization, about 5.9 million tons of medical waste are generated in the U.S. each year. About 15% of the amount produced is hazardous. Each category of waste requires careful separation and handling, which can be difficult, as in a survey conducted by Covanta and One, 62% of Americans have a lack of knowledge regarding the proper ways to recycle and dispose of products. Improper organization results in having to incinerate waste, which results in emissions harmful to both the human body and the environment.



Table 3: KAP score regarding BMW amongst nursing professionals

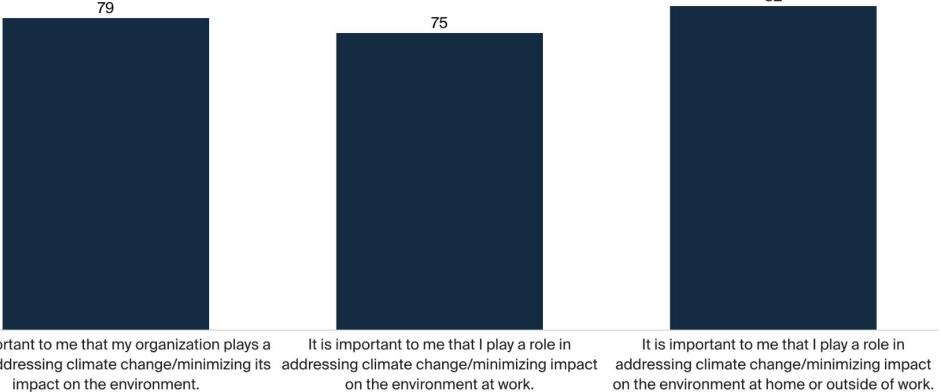
	Knowledge	Attitude	Practice
Excellent (>90%)	13(19.7)	28(42.4)	38(57.5)
Good (80% to 90%)	27(40.9)	1(1.5)	8(12.1)
Average (50% to 80%)	5(7.6)	8(12.1)	8(12.1)
Poor (<50%)	21(31.8)	29(43.9)	12(18.2)

...

EXHIBIT 1

A majority of clinicians surveyed feel it's important that the health system they work for plays a role in addressing climate change.

Percentage of surveyed clinicians who "agreed" or "strongly agreed" with the following statement



A health system or hospital's actions to reduce its environmental impact can play a role in a prospective employee's decision to join the organization.

Percentage of surveyed clinicians who said an organization's or prospective employer's policies and actions on reducing its environmental impact played or could play a role in their decision to ...



MEDICAL WASTE MANAGEMENT MARKET

Global Market Insights

GLOBAL STATISTICS

>\$12.8 BN

Market size
(2021)

>\$23.6 BN

Market size
(2030)

6.9%

CAGR
(2022-30)

MARKET SHARE (2021)



32.3%

Recycling
services
segment



>13.8%

Nursing
home
segment

Ref: Pages 6-9



Current Solutions (Ref: Pages 12-22)



Decision Matrix (Ref: Pages 23-28)

1: Lowest 5: Highest	Sorting	Sensing	Auto	Speed	Accuracy	Cost
SterilWave	1	2	2	2	5	1
Daniels Health Products	3	3	1	3	3	3
Trashbot 	5	4	4	5	4	2
Incineration	1	1	2	3	4	4

Consumer Needs (Ref: Pages 29-31)

Organization Needs

- Simplified organization process
- Manual input minimization

Disposal Needs

- Efficiency of Materials
 - Electric surgical instruments disposed after minimal use
- Curbing Environmental Harm
 - Incineration pollution, exorbitant landfill pollution

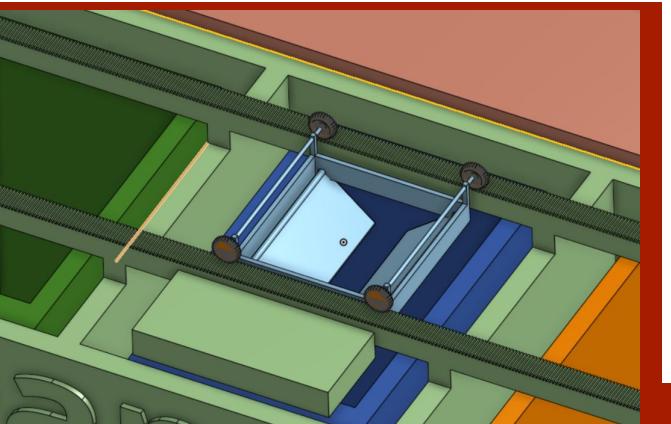
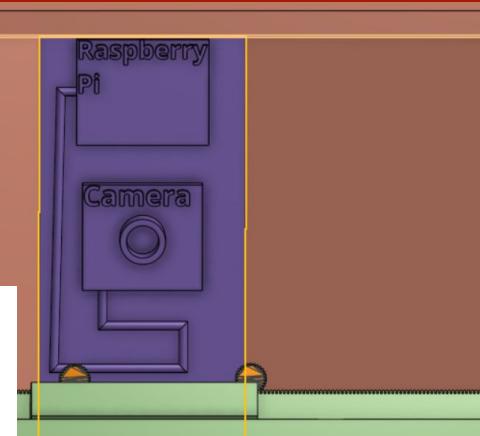
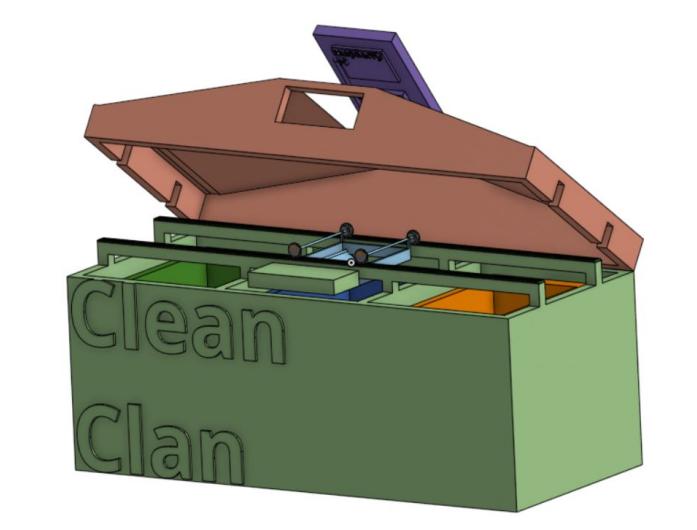
Design Constraints (Ref: Pages 33-35)

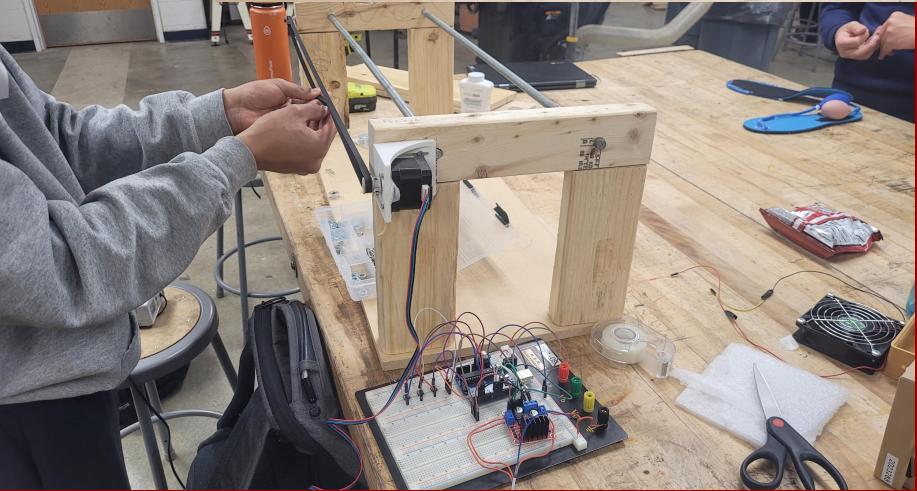
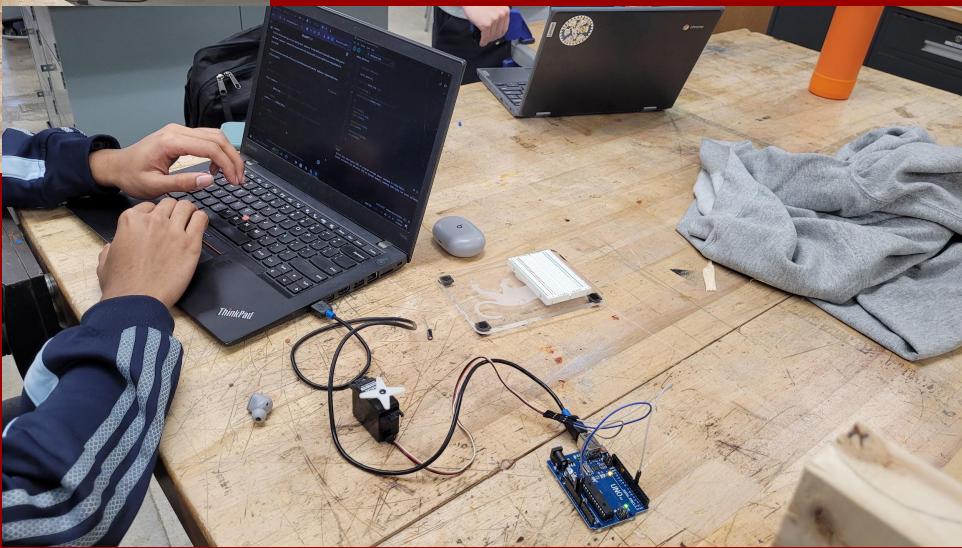
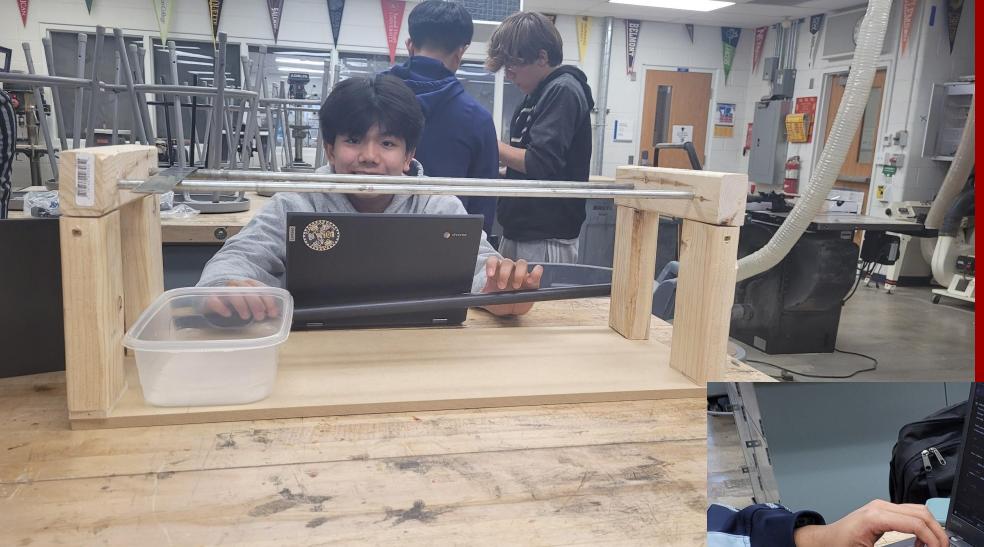
4 Core Goals

- IMPROVE **SAFETY**
- INCREASE **COMPLIANCE**
- REDUCE **COSTS**
- MITIGATE **ENVIRONMENTAL HARMs**



Prototype Concept – Sorter Machine (Ref: Element G p.59)





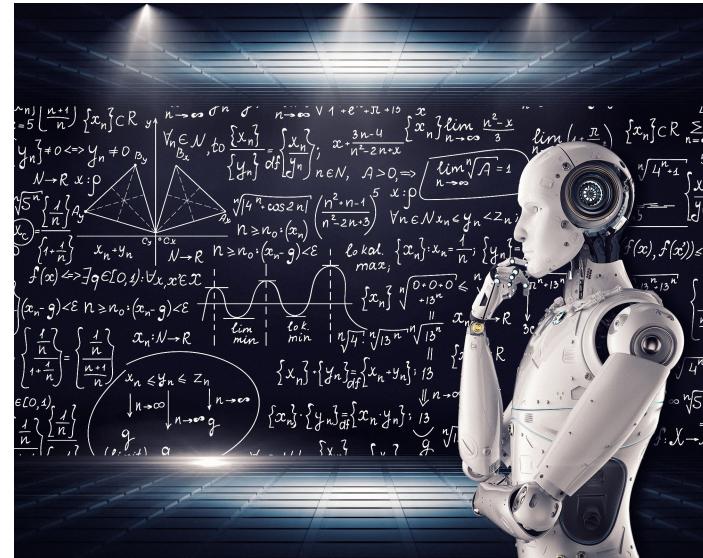




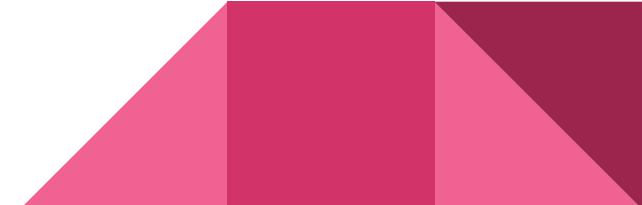
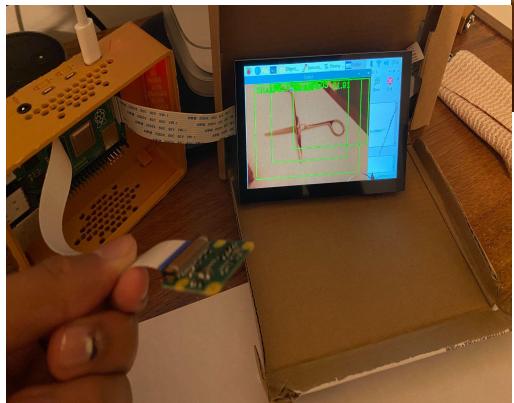
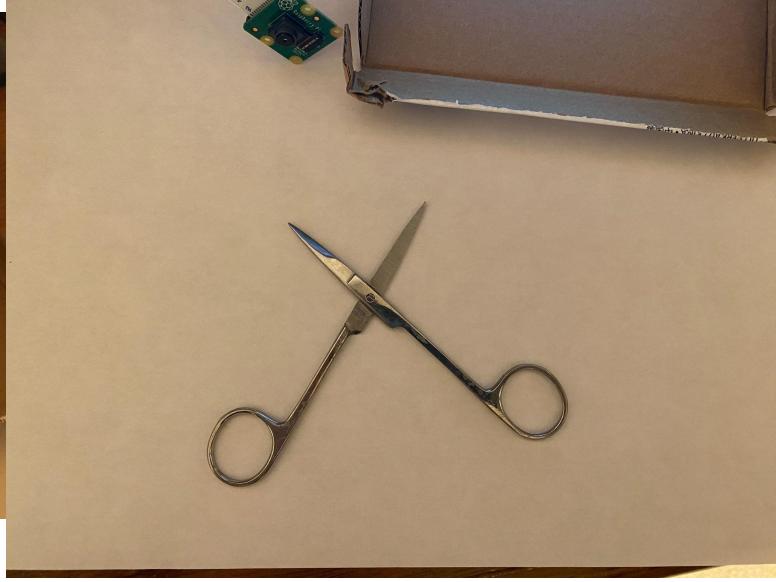
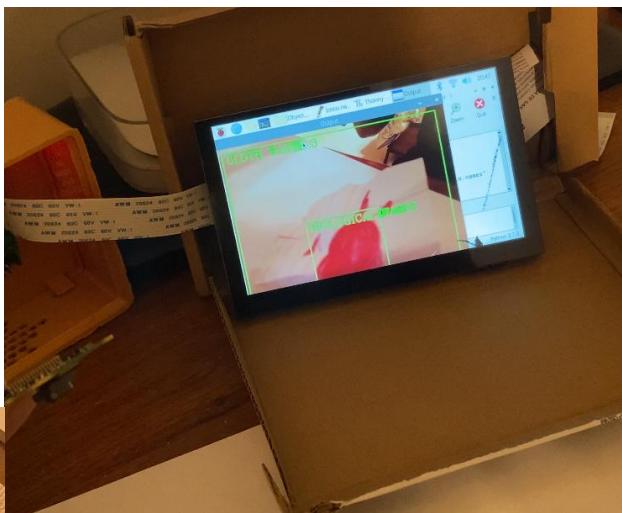
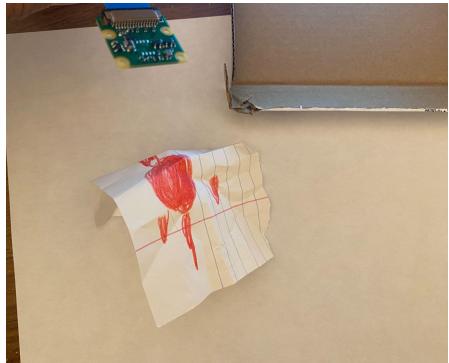
```
object-ident.py ● object-ident-2.py ● object-ident-3.py ● frozen_inference_graph.pb ● coco.names
C: > Users > alenn > Downloads > Object_Detection_Files > Object_Detection_Files > object-ident.py > ...
1 import cv2
2
3
4 classNames = []
5 configFile = "/home/pi/Desktop/Object_Detection_Files/coco.names"
6 with open(configFile,"rt") as f:
7     classNames = f.read().rstrip("\n").split("\n")
8
9 configPath = "/home/pi/Desktop/Object_Detection_Files/ssd_mobilenet_v3_large_coco_2020_01_14.pbtxt"
10 weightsPath = "/home/pi/Desktop/Object_Detection_Files/frozen_inference_graph.pb"
11
12 net = cv2.dnn_DetectionModel(weightsPath,configPath)
13 net.setInputSize(320,320)
14 net.setInputScale(1.0/ 127.5)
15 net.setInputMean((127.5, 127.5, 127.5))
16 net.setInputSwapRB(True)
17
18
19 def getObjects(img, thres, nms, draw=True, objects=[]):
20     classIds, confs, bbox = net.detect(img,confThreshold=thres,nmsThreshold=nms)
21     #print(classIds,bbox)
22     if len(objects) == 0: objects = classNames
23     objectInfo =[]
24     if len(classIds) != 0:
25         for classId, confidence,box in zip(classIds.flatten(),confs.flatten(),bbox):
26             className = classNames[classId - 1]
27             if className in objects:
28                 objectInfo.append([box,className])
29             if (draw):
30                 cv2.rectangle(img,box,color=(0,255,0),thickness=2)
31                 cv2.putText(img,classNames[classId-1].upper(),(box[0]+10,box[1]+30),
32                             cv2.FONT_HERSHEY_COMPLEX,1,(0,255,0),2)
33                 cv2.putText(img,str(round(confidence*100,2)),(box[0]+200,box[1]+30),
34                             cv2.FONT_HERSHEY_COMPLEX,1,(0,255,0),2)
35
36     return img,objectInfo
37
38
```

Model Creation and Training ([Elements H and I] p.73)

- Expert Consultation: Ciby Daniel (Senior Level Cloud/Network Engineer)
 - Model Training Resources, Python, Product Optimization
- Jupyter Notebook IDE
- LabelIMG: image tagging
- Tensorflow Object Detection
- Evaluation and Retesting



Model Training and Testing



Evaluation and Refinement

Summary of Results

- What did we do well?

Refinement

- More categories
- Faster processing
- More sterilization/sanitation
- Machine Learning

