# Executive Summary – Solution Evaluation

The following sub-functions were rated against each criteria by the ‘*average’* solutions; I just put an estimate of how the common solution will perform, and then we can actually decide what specific solution we want and/or conduct further research to find the best tool for our team.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Specification** | **Image Processing** | | **IoT + App/Website** | | **Voice Commands** | | **Gesture Control** | | **Microprocessor** | |
| Sorting accuracy | >=75% | 10 | N/A | 10 | N/A | 10 | N/A | 10 | N/A | 10 |
| Composition | N/A | 10 | N/A | 10 | N/A | 10 | Raw material | 0 | Raw material | 0 |
| Safety Standards | ? | ? | ? | ? | ? | ? | Met (Jetson Nano) | 10 | Met | 10 |
| Lifespan | N/A | 10 | N/A | 10 | N/A (depending on input) | 10 | ?? | 10 | ~5 yrs | 5 |
| Cost | $0 | 10 | $35-$110 | 5 | $0-$7 | 9 | $100 | 2 | $25-$100 | 5 |
| Sorting | ~2-5 seconds | 5 | ~2-5 seconds (TCP) | 5 | N/A | 10 | N/A | 10 | ~2-5 seconds (TCP + BT/Wifi) | 5 |
| Power consumption | 7-12V | 7 | N/A (cloud computing) | 10 | N/A | 10 | 5V | 10 | 7-12V | 7 |
| System Complexity | Requires Cam | 8 | Requires BT/wifi + cloud | 7 | Easy implementation | 9 | Good | 9 | Required for most functions | 8 |

\*Note: photos for image detection (Trashnet) were taken by an iPhone 5, iPHone 7 and iPhone SE. If we choose this solution, we need to either use common phone, or use a camera with similar specifications if we are assuming the same efficacy rate. If not, we need to train the model with an entirely new set to get an efficacy rate.

\*Note: Microprocessor can be used for LCD+Buttons, Touch Screen, loT+App/Website, general logic

\*Note: I’m assuming a cost of $0 for phone usage (assuming user already has phone)

\*\*Need to find lifespan of Jetson nano

# Sub-Function Solutions Research

## Identification – Image Processing

Image processing for waste exists online for free. The following are public Github repos with brief descriptions.

### Waste Sorter

Capabilities:Sorts images of into cardboard, glass, metal, paper, plastic, trash at a success rate of 92%. Uses fatsia deep learning framework.

Requirements:Given source files, corresponding dataset (available on the Github README), and Anaconda IDE (free python IDE)

Repository:<https://github.com/collindching/Waste-Sorter>

### Trash Recycling Classifier

Capabilities:Sorts images of into cardboard, glass, metal, paper, plastic, trash. Uses pytorch deep learning framework. Efficiency *unknown.*

Requirements:Given source files, corresponding dataset (available on the Github README, cloud environment for training (AWS or Google colab).

Repository:<https://github.com/winkhai/trash-recycling-classifier>

### Recycle

Capabilities:Sorts images of glass, metal, and plastic at a success rate of 85%. Uses

Requirements:Given source files, corresponding dataset (available on the Github README), and a python IDE

Repository:<https://github.com/ZAFERSHAMIM/recycle>

### Trashnet

Capabilities:Sorts images of glass, paper, cardboard, plastic metal and trash at a success rate of 75%. Devices used to take dataset photos were common phones (iPhone 7, iPhone 5, iPHone SE).

Requirements:Given source files, corresponding dataset (available on the Github README), and a python IDE

Repository:<https://github.com/garythung/trashnet>

## IoT + App / Website

We can make an app for free using third party tools or manually (XCode, VS Code)

### Android App Builder – MIT App Inventor / App Builder / CodeUp Start

Capabilities: Minimal, low code environments with networking, scalable, compatible with most cloud environments

Cost: $25 (to publish to Google Play)

Source:

<https://www.producthunt.com/alternatives/google-app-maker>

<https://appinventiv.com/blog/how-to-submit-app-to-google-play-store/#:~:text=There%20is%20only%20a%20one,you%20publish%20are%20cost%2Dfree>

### Apple App Builder – MIT App Inventor / App Builder / CodeUp Start

Capabilities: Minimal, low code environments with networking, scalable, compatible with most cloud environments

Cost: $99 (to publish to App Store)

Source: <https://codewithchris.com/submit-your-app-to-the-app-store/#:~:text=In%20order%20to%20be%20able,in%20the%20Apple%20Developer%20Program.&text=It%20costs%20%2499%2Fyear%20but,Stores%20on%20all%20Apple%20platforms>

### Manual App Builder – XCode / VS Code

Capabilities: High code environments with networking, scalable, compatible with most cloud environments (but a lot of boiler plate code)

Cost: $0

Source: Me

### Cloud Environments – Google / AWS

Capabilities: App engine + Cloud DB servers + Cloud SQL

Cost: $11 / month (24/7 access)

Source: <https://cloud.google.com/products/calculator#id=89a14829-8f1e-4f0e-846f-7fc8f00ab297>

## Voice Commands

These can be implemented using free APIs either on from an app or on the microprocessor directly. We need to make that decision, and cost will be decided accordingly. Note some microprocessors already have audio input.

### Web API – React API (for apps or microprocessor)

Capabilities: Flexible / programmable voice commands, relatively easy to install with npm

Cost: $0 (apart from either phone or microphone on microprocessor)

Source: <https://medium.com/better-programming/how-to-implement-voice-command-in-your-react-app-f226edb2f202>

### Microprocessor – Microphrone

Capabilities: continuous input, 40dB or 50dB , 1.25V bias

Cost: ~$7

Source: <https://www.adafruit.com/product/1713>

## Gesture Control

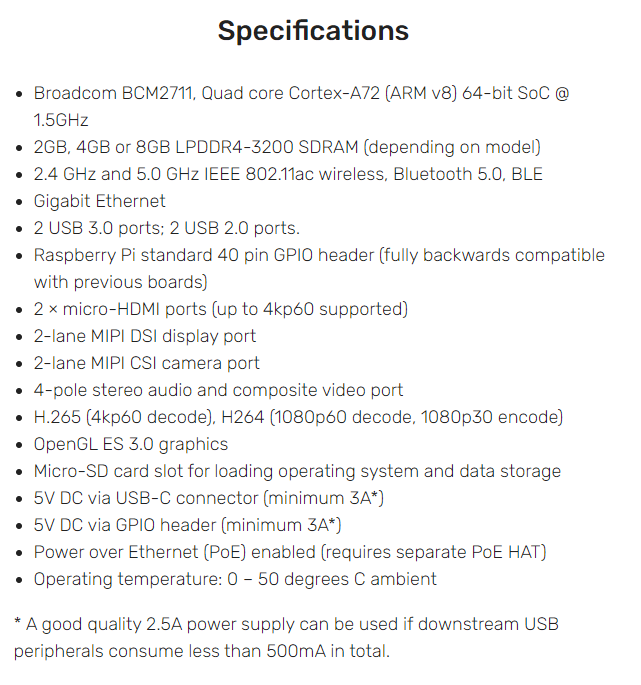
We would need a processor that already has gesture control capabilities (it’s too much boiler plate / overhead to implement within 2 months). Most notable microprocessor capable of this is the NVIDIA Jetson Nano. See the appropriate section for all the info.

# Extraneous

## Logic – Microprocessors

### Raspberry Pi

Specifications: made from raw materials, average 5 year lifespan



Cost:~$35

Source:<https://raspberrypi.stackexchange.com/questions/38321/raspberry-pi-lifespan-reliability#:~:text=The%20system%20will%20run%20all,autoreboot%2C%20and%20good%20SD%20cards>.

<http://www.designlife-cycle.com/raspberry-pi> <https://www.raspberrypi.org/products/raspberry-pi-4-model-b/specifications/>

### Arduino

Specifications:

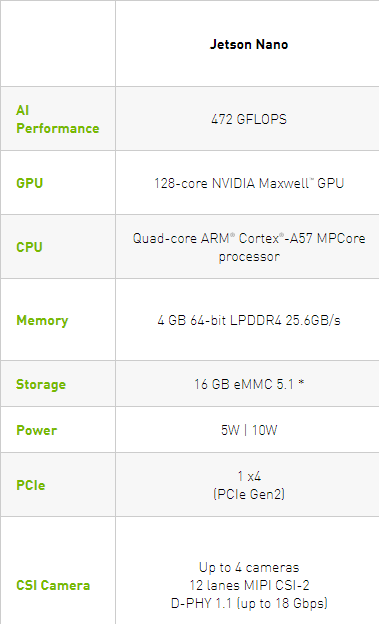


Cost:~$20

Source:<https://store.arduino.cc/usa/arduino-uno-rev3>

### Jetson Nano

Specifications: 5V consumption



Cost:~$100

Source:<https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/jetson-nano/>

<https://www.ximea.com/support/wiki/apis/Jetson_Nano_Benchmarks#:~:text=The%20carrier%20board%20consumes%20between,offer%20(5V%20and%202A)>.

## UDP vs TCP Connections

I’m assuming TCP here because we don’t need the images to be sent super quickly and I’d imagine we don’t want to deal with lost packages/images and we also don’t want to compress images before sending.

\*\*do we want to encrypt data ?? probably out of scope