

# Smart Factory del Riciclo

**Realizzato da:** Cortinovis Melissa, Gherardi Viola, Palazzi Chiara, Pesenti Isabel della classe 3E della scuola secondaria di primo grado (IC Valle Serina San Pellegrino Terme)

**Nome del gruppo:** Fuori circuito 4.0

**Materia e docente coinvolto:** Tecnologia, prof.ssa Pesenti Rachele.

[Link al video realizzato](#)

## Obiettivi



La nostra docente di Tecnologia ci ha proposto a inizio anno di realizzare un progetto robotico con gli strumenti presenti presso la nostra scuola. Noi abbiamo subito accolto con entusiasmo la proposta anche se inizialmente non avevamo idea di cosa realizzare.

Poi ci siamo focalizzate sull'ecologia e su come la tecnologia del giorno d'oggi possa aiutarci, infatti questa piattaforma trasforma il concetto di

"discarica" in una Smart Factory del Riciclo, dove la tecnologia non solo separa i materiali, ma ottimizza l'intera catena logistica urbana, rendendo il processo più silenzioso, pulito e redditizio.

L'obiettivo del sistema è automatizzare la gestione dei flussi all'interno della piattaforma ecologica per massimizzare l'efficienza e la qualità del riciclo. Smistando infatti i rifiuti "alla fonte" con estrema densità, si riduce drasticamente la necessità di spostare mezzi pesanti e carrelli elevatori all'interno dell'area. Meno chilometri percorsi dai camion significano minori emissioni di CO<sub>2</sub>. Inoltre, si riduce l'esposizione degli operatori a materiali potenzialmente pericolosi.

## Pianificazione e Risorse

I materiali che abbiamo avuto a disposizione a scuola per realizzare il progetto sono stati:

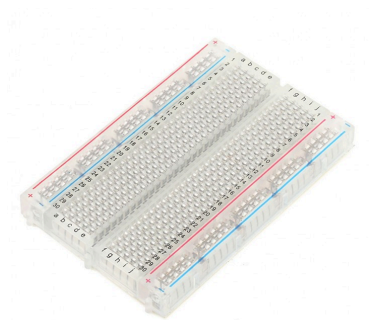
Scheda Arduino Uno



Braccio Robotico TinkerKit



BreadBoard per testare i collegamenti



Fotocamera intelligente Husky lens



Resistenze varie



Bottoni elettronici



### Cavi di collegamento Dupont di vario tipo



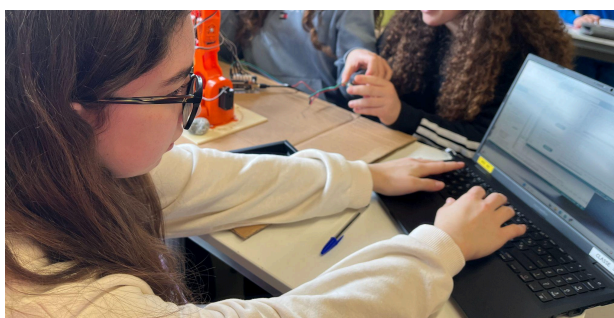
Le ore di lezione dedicate a scuola alle varie fasi di lavoro per montare, testare, assemblare e programmare il tutto sono riportati nella seguente tabella:

COSTRUZIONE BRACCIO	6 ore
PROGRAMMAZIONE ARDUINO UNO	10 ore
PROGRAMMAZIONE HUSKY LENS	4 ore
ASSEMBLAGGIO DI TUTTO IL PROGETTO	5 ore
COLLAUDO	4 ore

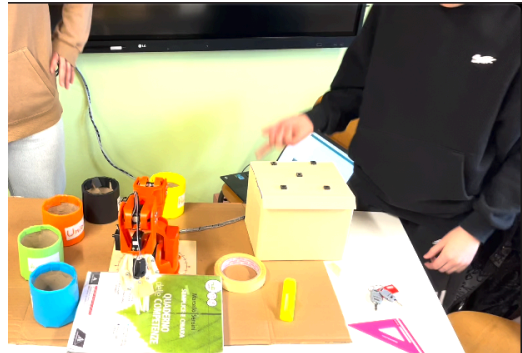
Come si vede dalla tabella l'attività che ha richiesto più ore di lezione è stata quella di programmazione, che ha richiesto anche una parte di studio da parte nostra in quanto non avevamo mai programmato in C++.

## Realizzazione del progetto

Abbiamo inizialmente montato un braccio robotico Tinkerkit dotato di sei motori e programmabile con una scheda Arduino UNO e una scheda di espansione. Dopodiché abbiamo assemblato tutte le sue componenti e collegato i motori alla scheda di espansione agganciata alla scheda Arduino UNO. A questo punto avremmo dovuto procedere con la programmazione di linguaggio C++. Per noi è stato il passaggio più complesso perché fino a quel momento avevamo programmato solo con SCRATCH, quindi abbiamo approfondito e studiato tutte quelle istruzioni ed espressioni che componevano



questo codice. Abbiamo dapprima imparato ad usare le istruzioni base e poi abbiamo installato la libreria “Braccio” nell’IDE Arduino che seguendo le indicazioni riportate nel manuale del braccio robotico Tinkerkit, siamo riuscite a scrivere le prime istruzioni per far muovere il robot. Questo è stato solo l’inizio di quello che sarebbe stato un lungo lavoro. Infatti abbiamo dovuto testare varie istruzioni che messe una di seguito all’altra davano al robot la sequenza di movimenti da fare per spostare i vari rifiuti nella posizione corretta. Dapprima abbiamo azionato il robot con l’aiuto di bottoni, uno per ogni tipo di rifiuto.



Giunte a questo punto però abbiamo deciso di rendere il processo ancora più Smart grazie all’utilizzo della fotocamera Husky Lens dotata di AI in grado di riconoscere gli oggetti che avevamo già inserito precedentemente, integrando alla programmazione tutti i comandi necessari per far muovere il braccio quando è presente un certo tipo di rifiuto. Abbiamo quindi eliminato i bottoni. La fotocamera intelligente, dopo aver riconosciuto il tipo di rifiuto, manda un messaggio alla scheda Arduino UNO la quale a sua volta invia al braccio robotico la sequenza di istruzioni corretta per gettare il rifiuto nel cestino

corretto.

Infine, siamo passate alla collaudo del sistema e all’assemblaggio finale, ma come spesso succede, abbiamo riscontrato alcuni problemi: la telecamera Husky Lens non era in grado di distinguere una pallina di stagnola da un sacchetto accartocciato di plastica a causa della forte similitudine. Abbiamo quindi dovuto utilizzare oggetti con forme diverse per facilitarne il riconoscimento.

## Analisi dei Risultati

Al termine di questo lavoro siamo riuscite a realizzare un circuito basato sulla programmazione di una scheda Arduino UNO che ha risposto correttamente ai comandi che volevamo fargli eseguire. La fotocamera con intelligenza artificiale alla fine è riuscita a identificare correttamente gli oggetti che avevamo preparato.



Una parte importante del lavoro è stata dedicata alla programmazione dei movimenti da far fare al braccio robotico. Ogni istruzione infatti doveva prevedere la movimentazione dei vari motori secondo una sequenza e con tempi che permettessero di raccogliere l'oggetto e posizionarlo nel contenitore corretto per lo smaltimento.

Il risultato finale corrisponde pienamente agli obiettivi che ci eravamo prefissate, ovvero riuscire a far comunicare eseguire il programma scritto in C++ a seconda dei segnali inviati dalla fotocamera intelligente.

## Conclusioni e Autovalutazione

Grazie a questo lavoro abbiamo imparato a usare strumenti come Arduino UNO, a programmare in C++, a lavorare con sensori e attuatori e a programmare una fotocamera con intelligenza artificiale. Inoltre, abbiamo capito meglio come progettare e realizzare un piccolo impianto tecnologico. Abbiamo imparato anche a gestire e organizzare al meglio il lavoro di squadra, migliorando la nostra capacità di risolvere problemi, dato che durante il progetto ci siamo trovate di fronte ad alcune difficoltà e abbiamo dovuto trovare soluzioni. Infine, abbiamo imparato a organizzare il lavoro e a gestire il tempo, rispettando le scadenze e portando avanti il progetto in modo efficace.

### SVILUPPI FUTURI:

Un'idea che ci è venuta per un possibile miglioramento del nostro progetto è quello di aggiungere delle luci o suoni a seconda del tipo di rifiuto che viene

riconosciuto, il cestino potrebbe avere a sua volta una luce che si accende o fare un suono per indicare all'operatore della discarica dove sta andando a finire il rifiuto ricevuto..

Un'altra idea sarebbe quella di integrare modelli di AI più avanzati per riconoscere anche rifiuti complessi che devono essere separati per smaltire al meglio tutti i materiali di cui è composto il rifiuto.

Il sistema potrebbe essere implementato anche con questi dispositivi aggiuntivi:

- REGISTRO DEI DATI: Aggiungere un file che registra i dati dei rifiuti riconosciuti;
- SCHERMO INFORMATIVO = Aggiungere un piccolo display che mostra che tipo di rifiuto è stato riconosciuto;
- MESSAGGI EDUCATIVI= Dopo aver riconosciuto il rifiuto, il sistema potrebbe mostrare brevi messaggi per insegnare a fare meglio la raccolta differenziata;
- PANNELLI SOLARI = Aggiungere dei pannelli solari in modo che al posto di utilizzare la corrente il robot funzioni tramite la luce solare;

Siamo molto soddisfatte del progetto che abbiamo realizzato, nonostante ci siano stati alcuni problemi che però siamo riuscite a risolvere. Le ore dedicate, sia quelle a scuola che quelle di studio a casa, per la realizzazione del progetto sono state tante ma quando abbiamo visto che tutto funzionava come volevamo noi siamo state contente e le fatiche e i sacrifici fatti non pesavano più. Siamo per questo orgogliose di presentare la nostra idea realizzata e funzionante.

## **Bibliografia e Sitografia**

Abbiamo utilizzato dispense e spiegazioni fornite dalla nostra professoressa.

Guida cartacea e Online del braccio TinkerKit:

<https://docs.arduino.cc/retired/getting-started-guides/Braccio/>

Guida sulla programmazione e utilizzo di Huskylens e libreria per IDE Arduino:

<https://dronebotworkshop.com/huskylens>

[https://github.com/DFRobot/DFRobot\\_HuskylensV2](https://github.com/DFRobot/DFRobot_HuskylensV2)

Pagina di progetti e tutorial Arduino come spunti iniziali per imparare la programmazione di Arduino Uno:

<https://www.progettiarduino.com/progetti-e-tutorial.html>

## Codice di programmazione realizzato:

```
#include <Servo.h>
#include <Braccio.h>
#include "HUSKYLENS.h"
#include "Wire.h"

HUSKYLENS huskylens; //nome della fotocamera

int ID =1; // numero identificativo dell'oggetto identificato

//Servomotori del braccio robotico
Servo base;
Servo shoulder;
Servo elbow;
Servo wrist_ver;
Servo wrist_rot;
Servo gripper;

void setup() {

  Serial.begin(9600);
  Wire.begin();
  while (!huskylens.begin(Wire)) {
    Serial.println("HuskyLens non connessa!");
    delay(1000);
  }
  Serial.println("HuskyLens pronta!");
  Braccio.begin(); //inizializzazione del Braccio robotico
}

void loop() {

  if (!huskylens.request()) {
    Serial.println("Errore richiesta dati");
  }
  else if (!huskylens.available()) {
    Serial.println("Nessun oggetto rilevato");
  }
  else {
//questa sequenza restituisce l'informazione dell'ID dell'oggetto riconosciuto da
HuskyLens
    while (huskylens.available()) {
      HUSKYLENSResult result = huskylens.read();
```

```

    Serial.print("ID: ");
    Serial.print(result.ID);
    ID = result.ID;
    Serial.print("\n");
}
}
delay(500);

//Se non premo nulla il robot si mette nella posizione di partenza stabilita
//posizione di partenza
if (ID==1){
    // movimnenti mot-.aspetta m1 m2 m3 m4 m5 m6
    Braccio.ServoMovement (10, 0, 90, 90, 90, 90, 80);
}

//Se identifica il primo oggetto
    if (ID==2){
        // movimnenti mot-. aspetta m1 m2 m3 m4 m5 m6
        Braccio.ServoMovement (10, 0, 90, 90, 90, 90, 80);
        Braccio.ServoMovement (10, 0, 90, 20, 90, 90, 5);
        Braccio.ServoMovement (20, 0, 90, 20, 0, 90, 5);
        Braccio.ServoMovement (20, 0, 90, 20, 0, 90, 5);
        Braccio.ServoMovement (20, 0, 90, 20, 0, 90, 80);
        delay(2000);
        Braccio.ServoMovement (20, 0, 90, 90, 0, 90, 100);
        Braccio.ServoMovement (20, 150, 90, 90, 0, 90, 100);
        Braccio.ServoMovement (20, 150, 90, 50, 0, 90, 100);
        Braccio.ServoMovement (20, 150, 90, 50, 0, 90, 10);
    }

//Se identifica il secondo oggetto
    if (ID==3){
        // movimnenti mot-. aspetta m1 m2 m3 m4 m5 m6
        Braccio.ServoMovement (10, 0, 90, 90, 90, 90, 80);
        Braccio.ServoMovement (10, 0, 90, 20, 90, 90, 5);
        Braccio.ServoMovement (20, 0, 90, 20, 0, 90, 5);
        Braccio.ServoMovement (20, 0, 90, 20, 0, 90, 5);
        Braccio.ServoMovement (20, 0, 90, 20, 0, 90, 80);
        delay(2000);
        Braccio.ServoMovement (20, 0, 90, 90, 0, 90, 100);
        Braccio.ServoMovement (20, 180, 90, 90, 0, 90, 100);
        Braccio.ServoMovement (20, 180, 90, 50, 0, 90, 100);
        Braccio.ServoMovement (20, 180, 90, 50, 0, 90, 10);
    }

//Se identifica il terzo oggetto

```

```

if (ID==4){
// movimnenti mot-. aspetta  m1 m2 m3 m4 m5 m6
Braccio.ServoMovement (10, 0, 90, 90, 90, 90, 80);
Braccio.ServoMovement (10, 0, 90, 20, 90, 90, 5);
Braccio.ServoMovement (20, 0, 90, 20, 0, 90, 5);
Braccio.ServoMovement (20, 0, 90, 20, 0, 90, 5);
Braccio.ServoMovement (20, 0, 90, 20, 0, 90, 80);
delay(2000);
Braccio.ServoMovement (20, 0, 90, 90, 0, 90, 100);
Braccio.ServoMovement (20, 50, 90, 90, 0, 90, 100);
Braccio.ServoMovement (20, 50, 90, 50, 0, 90, 100);
Braccio.ServoMovement (20, 50, 90, 50, 0, 90, 10);
}

//Se identifica il quarto oggetto
if (ID==5){
// movimnenti mot-. aspetta  m1 m2 m3 m4 m5 m6
Braccio.ServoMovement (10, 0, 90, 90, 90, 90, 80);
Braccio.ServoMovement (10, 0, 90, 20, 90, 90, 5);
Braccio.ServoMovement (20, 0, 90, 20, 0, 90, 5);
Braccio.ServoMovement (20, 0, 90, 20, 0, 90, 5);
Braccio.ServoMovement (20, 0, 90, 20, 0, 90, 80);
delay(2000);
Braccio.ServoMovement (20, 0, 90, 90, 0, 90, 100);
Braccio.ServoMovement (20, 120, 90, 90, 0, 90, 100);
Braccio.ServoMovement (20, 120, 90, 50, 0, 90, 100);
Braccio.ServoMovement (20, 120, 90, 50, 0, 90, 10);
}

//Se identifica il quinto oggetto
if (ID==6){
// movimnenti mot-. aspetta  m1 m2 m3 m4 m5 m6
Braccio.ServoMovement (10, 0, 90, 90, 90, 90, 80);
Braccio.ServoMovement (10, 0, 90, 20, 90, 90, 5);
Braccio.ServoMovement (20, 0, 90, 20, 0, 90, 5);
Braccio.ServoMovement (20, 0, 90, 20, 0, 90, 5);
Braccio.ServoMovement (20, 0, 90, 20, 0, 90, 80);
delay(2000);
Braccio.ServoMovement (20, 0, 90, 90, 0, 90, 100);
Braccio.ServoMovement (20, 70, 90, 90, 0, 90, 100);
Braccio.ServoMovement (20, 70, 90, 50, 0, 90, 100);
Braccio.ServoMovement (20, 70, 90, 50, 0, 90, 10);
}
} //chiusura del void loop

```

