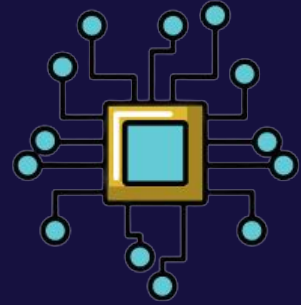




Co-funded by the
Erasmus+ Programme
of the European Union



ROMUAS
KILLS
RDUNO
SING
MACHINES
OUTDATED
RECOVER

C1

Argomento

Expert program toolkit
PR1

Le informazioni e le opinioni contenute in questa pubblicazione sono quelle degli autori e non riflettono necessariamente l'opinione ufficiale dell'Unione Europea. Né le istituzioni e gli organi dell'Uni one europea né chiunque agisca per loro conto possono essere ritenuti responsabili dell'uso che potrà essere fatto delle informazioni ivi contenute.





1. Argomento

L'intera comunità deve affrontare molte sfide nella società globalizzata di oggi, sia a causa della digitalizzazione che del cambiamento climatico. Tutto ciò contribuisce al rapido cambiamento del mercato del lavoro, nonché delle competenze e delle capacità richieste. È necessario rafforzare le competenze, la competitività e l'occupabilità delle risorse umane, dando priorità all'istruzione come motore dello sviluppo della società.

Al giorno d'oggi, intelligenza, innovazione e creatività sono diventate punti di riferimento rilevanti. Rappresentano una garanzia di prosperità futura per il mondo intero. Viviamo in un mondo in cui vengono richiesti non solo prodotti e mercati di alto valore, ma, sempre di più, competenze di alto valore. Per garantire una società della conoscenza, l'eccellenza dovrebbe essere assicurata in tutte le fasi del processo educativo, per aggiornare permanentemente la base di competenze della popolazione in base alle esigenze e per creare un ambiente sociale, economico e normativo in grado di stimolare la ricerca, la creatività e l'innovazione .

Lo scopo del toolkit del programma Expert è quello di creare un materiale formativo che fornisca ai gruppi destinatari del progetto conoscenze e competenze di alta qualità necessarie per modernizzare i macchinari esistenti al fine di renderli "intelligenti" e più adatti alle attuali esigenze della produzione industriale.

Il toolkit è composto da:

1. una parte teorica e una pratica contenente tutorial e video su come recuperare macchine (dispositivi) obsolete utilizzando Arduino
2. una presentazione dell'innovazione sociale che descrive brevemente lo sviluppo della tecnologia e poi si concentra sulle abilità sociali e sull'impatto della tecnologia su tali abilità, per finire con metodi interattivi per svilupparle
3. una presentazione teorica delle competenze imprenditoriali e dei metodi interattivi per svilupparle
4. considerazioni su green skills (competenze "verdi"), economia circolare, vantaggi e benefici del rinnovamento di dispositivi obsoleti e del loro utilizzo e metodi interattivi per sviluppare tali competenze

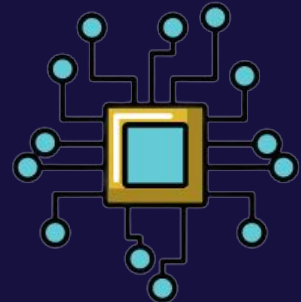
Le persone che seguiranno la prima parte del programma avranno la possibilità di apprendere attraverso esempi rilevanti come recuperare macchine obsolete che altrimenti verrebbero scartate, utilizzando la tecnologia Arduino. Gli esempi suggeriti li renderanno consapevoli di alcune possibilità di utilizzo dei componenti e del software Arduino e susciteranno il loro interesse e la loro creatività poiché l'uso della suddetta tecnologia è illimitato.

Chi seguirà la seconda parte avrà la possibilità di trovare alcune considerazioni sullo sviluppo della tecnologia e rispondendo alle domande dei questionari e interpretando i risultati avrà la possibilità di valutare il livello delle proprie abilità sociali e individuare gli aspetti che necessitano essere migliorati.





Recuperare macchine (dispositivi) aggiornate utilizzando la tecnologia Arduino



ROMUAS
KILLS
RDUINO
SING
MACHINES
OUTDATED
RECOVER

C2

Expert program toolkit
PR1



2. Recuperare macchine (dispositivi) aggiornate utilizzando la tecnologia Arduino

2.1. Parte teorica

Nozioni di base su Arduino e software di programmazione

Che cos'è Arduino?

In generale, Arduino è una piattaforma di prototipazione elettronica, che integra il popolare microcontrollore AVR, e permette lo sviluppo di sistemi di controllo interattivi. È un dispositivo a basso costo e accessibile a tutti. Alcune caratteristiche sono:

- Non richiede un programmatore esterno;
- Interagisce perfettamente con un compilatore dedicato;
- E' compatibile con numerose schede di espansione (es. controller motore, display, moduli, sensori, ecc.).

Tuttavia, il vero potere di Arduino risiede nel suo linguaggio di programmazione basato su C/C++.

La programmazione avviene attraverso le librerie, grazie alle quali anche la creazione di un programma complicato è alla portata di un programmatore principiante.



I vantaggi di Arduino

Il progetto è iniziato nel 2005, in Italia. Da allora ha raccolto una moltitudine di follower e utenti. Fin dall'inizio, Arduino è stato progettato per persone che avevano poca conoscenza della programmazione dei microcontrollori. Il suo software eccellente, la sintassi amichevole e il prezzo basso hanno reso Arduino estremamente popolare.



La comunità costruita attorno a questo progetto è enorme. Questo porta molti vantaggi. Dal punto di vista di un principiante, i più importanti sono i seguenti:

- Un gran numero di soluzioni già pronte. C'è una infinità di progetti Arduino già creati. Se pensi di aver creato qualcosa di "nuovo" e interessante, è probabile che uno degli utenti lo abbia già fatto e lo abbia pubblicato su Internet;

- La popolarità della piattaforma significa che ci sono innumerevoli varietà di schede ed estensioni create da diversi produttori - ne parleremo tra poco;
- Un gran numero di utenti rende facile trovare aiuto quando rimani bloccato da qualche parte nel progetto.

Arduino - La scelta dell'hardware

Arduino è una piattaforma Open Hardware. Ciò significa che sono disponibili tutti i materiali necessari per creare il proprio dispositivo. Per questo motivo, ci sono molte schede compatibili con Arduino.

Attualmente sono disponibili più di 20 modelli originali. In qualsiasi buon negozio di elettronica troverai molti di questi modelli. Ai fini del corso abbiamo scelto la scheda più popolare: Arduino UNO R3.



Com'è fatto Arduino UNO R3

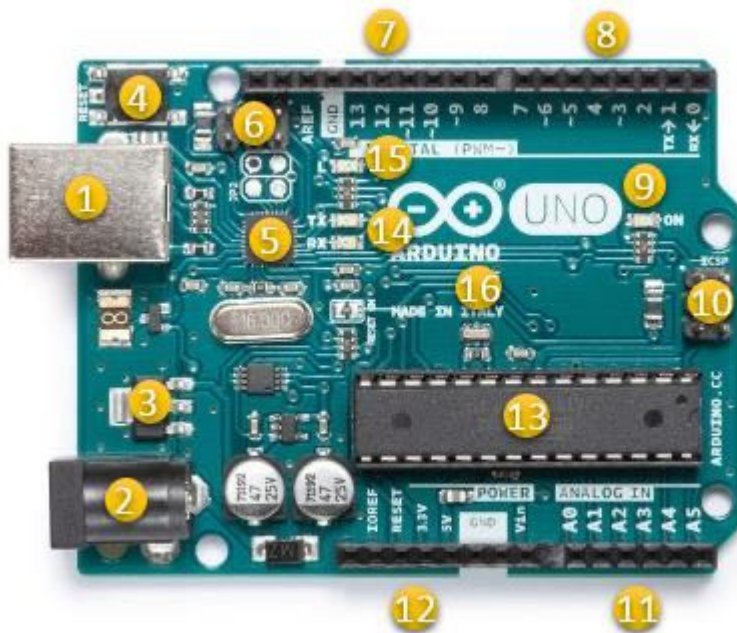
Il cuore di Arduino è il popolare microcontrollore Atmel a 8 bit, l'AVR ATmega328 che opera a 16 MHz. Lavorare a tale frequenza, molto semplicemente, significa che il microcontrollore è in grado di eseguire 16 milioni di operazioni al secondo!

I connettori posti ai lati della scheda sono le uscite dei segnali più importanti. Qui troviamo 14 ingressi/uscite digitali programmabili (I/O). Sei di essi possono essere utilizzati come uscite PWM (ad es. per controllare motori); troviamo poi 6 ingressi analogici. Troviamo anche un pulsante di reset e un connettore per l'alimentazione della scheda.

Arduino può essere alimentato in diversi modi. I metodi più diffusi sono:

- Alimentazione tramite cavo USB;
- Alimentazione tramite fonte di alimentazione esterna (batterie o plug-in).

I componenti più importanti sono contrassegnati nell'immagine seguente:



1. **Connettore USB** - utilizzato per l'alimentazione, la programmazione e la comunicazione con il computer;
2. **Connettore CC** - utilizzato per l'alimentazione (tensione consigliata da 7V a 12V);
3. **Stabilizzatore di tensione** - trasforma la tensione di ingresso in 5V;
4. **Pulsante di ripristino**: ripristina la scheda Arduino;
5. **Microcontrollore** - responsabile della comunicazione con il computer tramite USB;
6. **Terminali di programmazione** per il microcontrollore #5
7. **Pin I/O digitali** (da 8 a 13) e GND digitale (massa);
8. **Pin I/O digitali** (da 0 a 7) e TX e RX (uscita seriale e ingresso seriale rispettivamente);
9. **LED** - indica se la scheda è alimentata;
10. **Pin di programmazione seriale**: consente di programmare il microcontrollore (pin n. 13) tramite comunicazione seriale;
11. **Pin di ingresso analogico** (da 0 a 5);
12. **Pin di alimentazione e reset** della scheda;
13. **Microcontrollore AVR ATmega328** - il cuore della scheda Arduino;
14. **LED** - indicano la trasmissione da/verso il computer;
15. **LED** - collegato al pin #13, è a disposizione dell'utente;
16. **Oscillatore** a 16 MHz: fa funzionare il microcontrollore a una frequenza di 16 MHz.

Cloni e Arduino originale

Come accennato, Arduino è una piattaforma hardware aperta. Ciò significa che ognuno può creare il proprio Arduino o progettare una scheda simile. Le schede identiche ad Arduino sono comunemente chiamate cloni.

I cloni possono essere divisi in due tipi:

- Falsi integrali che imitano l'originale;
- Schede compatibili con Arduino.

Puoi correre il rischio e acquistare una scheda contraffatta a metà prezzo (o meno). Ma, la verità è che sono realizzati con materiali di qualità inferiore, e questo può causare facilmente malfunzionamenti e problemi. Le schede compatibili, invece, possono essere altrettanto efficienti di quelle originali.

Esistono anche altre schede più economiche compatibili con Arduino, che i produttori non pretendono di spacciare come originali. Queste versioni sono prodotte, ad esempio, da DFrobot, che vende le sue schede con il nome DFRduino.

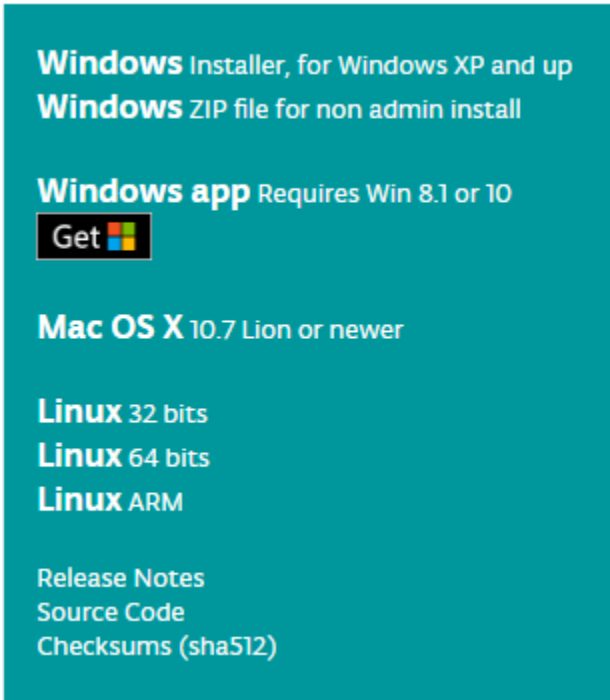


DFRduino è un prodotto affidabile, quindi puoi acquistarlo tranquillamente. I prodotti compatibili con Arduino possono essere facilmente identificati poichè riportano nel nome il suffisso -uino.

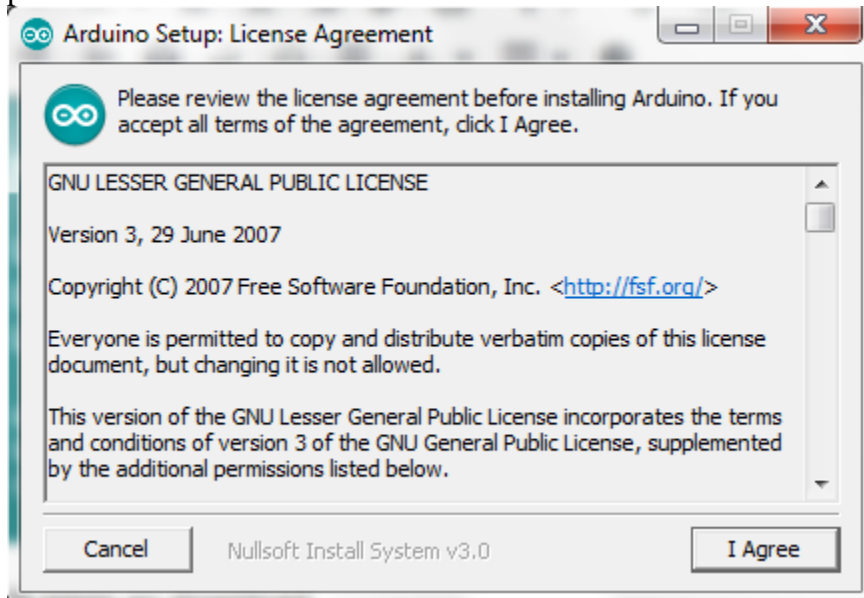
Installazione dell'IDE di Arduino

Prima di iniziare la programmazione, è necessario installare il software opportuno. È possibile scaricare l'ultima IDE di Arduino dal sito Web ufficiale di Arduino. Il software occupa circa 90 MB.

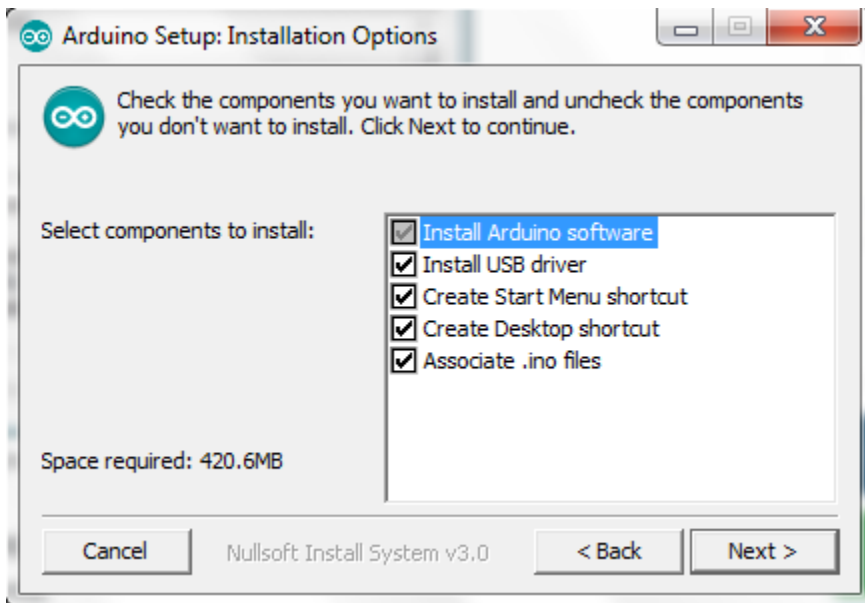
L'installazione è di base. Per iniziare dovrai scegliere il tuo sistema operativo.



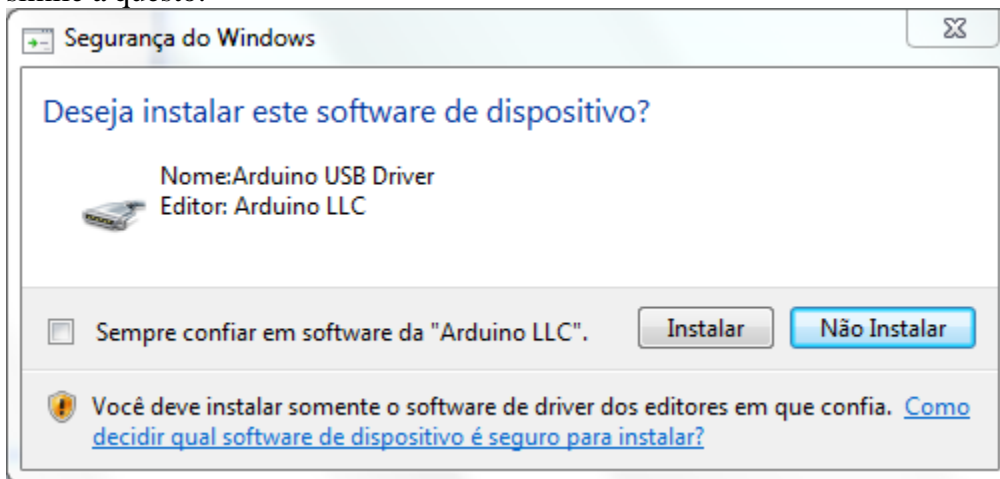
Al termine del download, inizierà l'installazione. Per prima cosa dovrai accettare la licenza del prodotto:



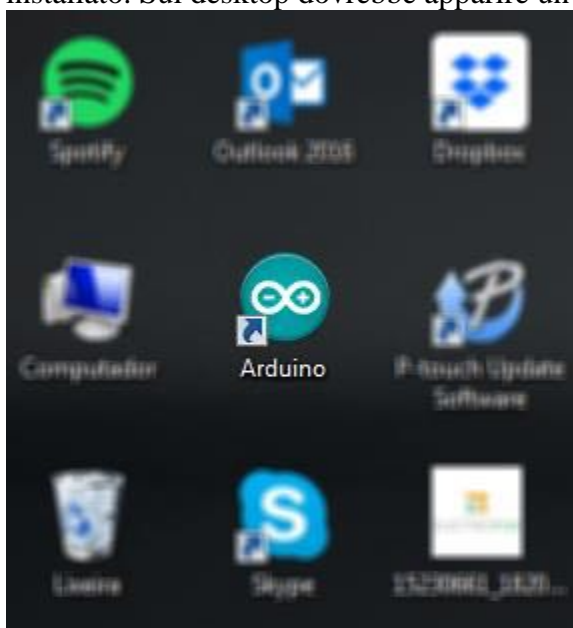
Quindi fare clic su Avanti, prestando attenzione ai componenti da installare:



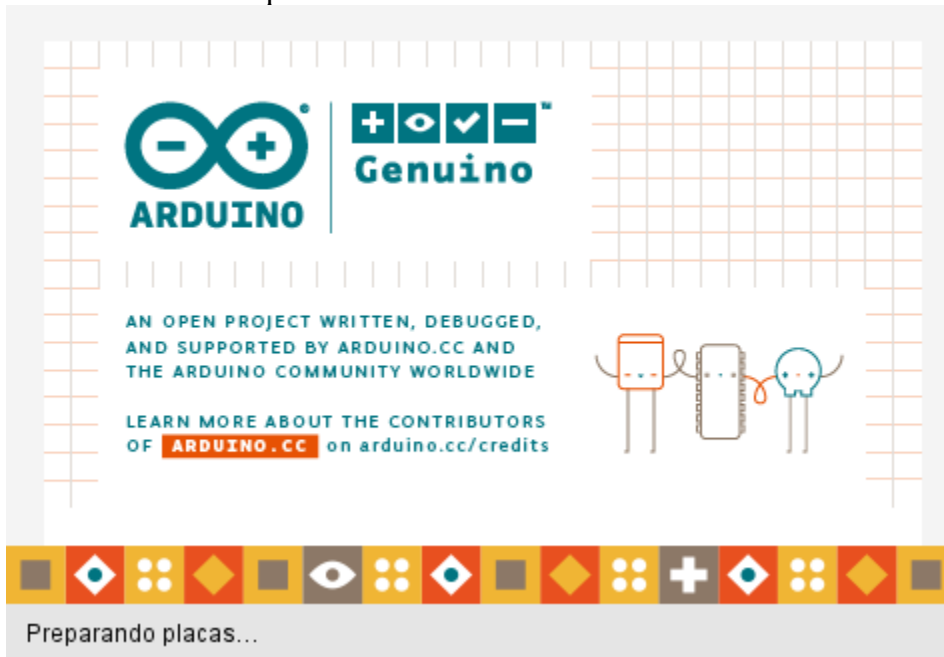
Se scegli di installare il driver USB (consigliato), al termine dell'installazione riceverai un avviso simile a questo:



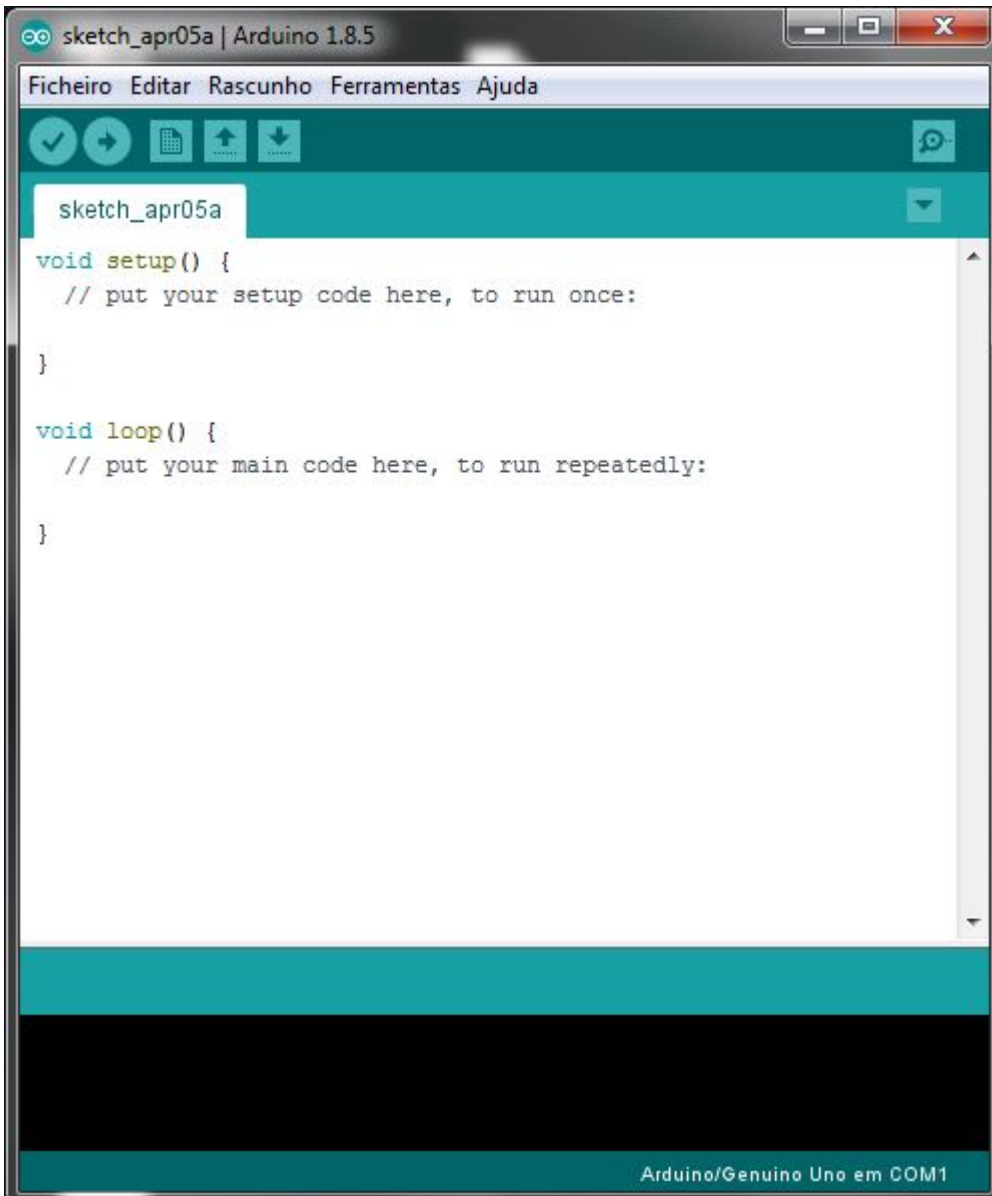
Fai clic su "Installa". Non devi preoccuparti di nulla. Al termine fai clic su Chiudi e il software è installato. Sul desktop dovrebbe apparire un'icona come la seguente:



Se il software è stato installato correttamente, all'apertura dell'IDE dovrebbe apparire un'immagine di benvenuto come questa:



Dopo pochi secondi, si apre l'editor:



Le basi della programmazione Arduino

Nel linguaggio C, tutte le istruzioni che faranno parte del nostro programma, devono essere scritte nella funzione main (maggiori informazioni sulle funzioni di seguito):

```
1 | int main() {  
2 |  
3 | //Conteúdo do Programa  
4 |  
5 | }
```

Il simbolo "//" indica un commento. È un'informazione di una riga che aiuta le persone a capire il programma. Durante la compilazione, tutti i commenti vengono omissi. Se vuoi scrivere un commento più lungo, devi racchiuderlo tra "/*" e "*/".

In Arduino, ci sono aspetti che sono semplificati, rispetto alla normale programmazione.



Ci sono due funzioni: una esegue le istruzioni al suo interno una sola volta, l'altra esegue le istruzioni contenute ciclicamente. Vediamo le due funzioni:

```
1 | void setup() {  
2 | //Instruções que são executadas apenas uma vez  
3 | }  
4 |  
5 | void loop() {  
6 | //Instruções que são executadas em loop  
7 | }
```

In pratica, la prima funzione di solito contiene le impostazioni. Ad esempio: impostare i pin come ingressi o uscite. Questa funzione, dopo aver acceso la scheda, eseguirà le azioni che devono avvenire solo una volta.

Nella seconda funzione scrivi il codice che vuoi eseguire sempre (in loop).

Funzioni: cosa significano?

I codici possono essere scritti da te oppure possono essere "prefabbricati", cioè forniti da programmatori o appassionati che condividono il proprio codice.

Il linguaggio C utilizza le funzioni (function). Una funzione, nei linguaggi di programmazione, è un blocco (elenco) di determinati comandi, la cui esecuzione fornisce un risultato.

Ogni funzione può ricevere diversi argomenti e fornire un risultato. Il programmatore può determinare quali valori saranno il risultato e quali i dati di input. Ogni funzione ha il proprio tipo, che identifica il risultato restituito (il tipo di risultato è indicato con un prefisso, ad esempio int, string, ecc.): può essere un numero, una stringa o qualcos'altro. Esiste anche un tipo specifico di funzione, che non invia alcun valore (è identificata dal tipo void).

Concentriamoci sulle due funzioni principali dei programmi Arduino.

```
1 | void setup() {  
2 | }
```

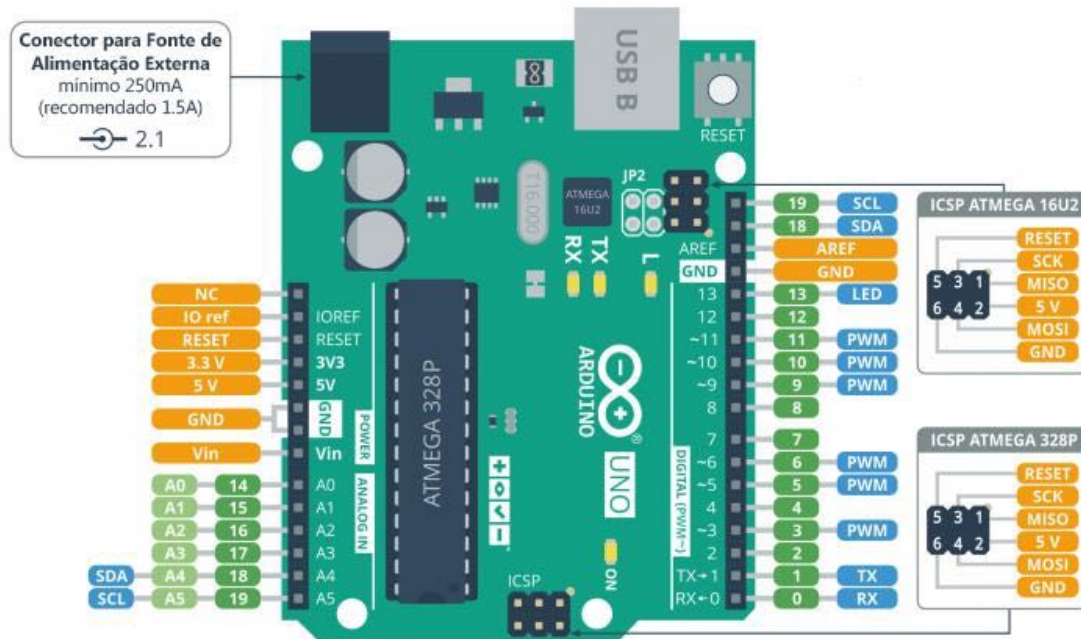
La funzione (o procedura) setup, come abbiamo già visto, prevede l'esecuzione una sola volta delle istruzioni (di set up) che contiene. Come suggerisce il nome, è destinata principalmente alla configurazione generale. Qui viene avviato il processore, vengono configurate le periferiche, ecc.

```
1 | void loop() {  
2 | }
```

La funzione loop (procedura loop) è, come suggerisce il nome, un ciclo infinito. È adatto per istruzioni che devono essere eseguite continuamente.

Arduino UNO - piedinatura

Tramite alcuni connettori è possibile collegare elementi esterni ad Arduino, come LED e pulsanti. Tuttavia, prima di arrivare a quella parte, dobbiamo conoscere la descrizione dei pin/connettori esistenti sulla scheda. Di seguito potete vedere uno schema con i pin principali di Arduino UNO:



In **verde scuro** (da #0 a #19) sono indicati i pin di ingresso/uscita digitale (I/O). Quando vengono utilizzati come uscite, possiamo definirli come 0V (livello logico 0, LOW) o 5V (livello logico 1, HIGH). Quando sono configurati come ingressi, sono in grado di rilevare se il pin in questione ha una tensione di 0V o 5V.

Gli ingressi analogici (A0-A5) sono evidenziati in **verde chiaro**. Questi sono gli unici pin che consentono di misurare la tensione (0-5 V). Come puoi vedere, la numerazione di questi pin coincide con i pin universali (da #14 a #19). Lavorare in modalità analogica è una loro funzione aggiuntiva.

In **blu** sono stati evidenziati i pin con funzioni alternative. Ciò significa che, oltre ad essere normali pin di I/O, possono svolgere funzioni più complesse. Spiegazione di base:

- **SDA, SCL** - Bus I2C utilizzati, ad esempio, per la comunicazione con sensori più avanzati. Ci sono due pin SDA e due pin SCL, nell'angolo in basso a sinistra e in alto a destra della scheda;
- **TX, RX** - Interfaccia UART, utilizzata principalmente per la comunicazione con il computer;
- **PWM** - uscite su cui è possibile generare un segnale ad onda quadra (variabile) PWM. È una funzione molto utile, ad esempio, nei servocomandi;
- **LED** - LED installato in modo permanente nell'Arduino, che è direttamente collegato al pin #13.

In **arancione** sono indicate le uscite non programmabili. Sono legate all'alimentazione del sistema. I connettori indicati nell'immagine come ICSP sono utilizzati per la programmazione diretta di due microcontrollori, che si trovano sulla scheda Arduino UNO. Questi connettori sono utilizzati in casi molto specifici.

Esercitazione pratica Uscite – LED

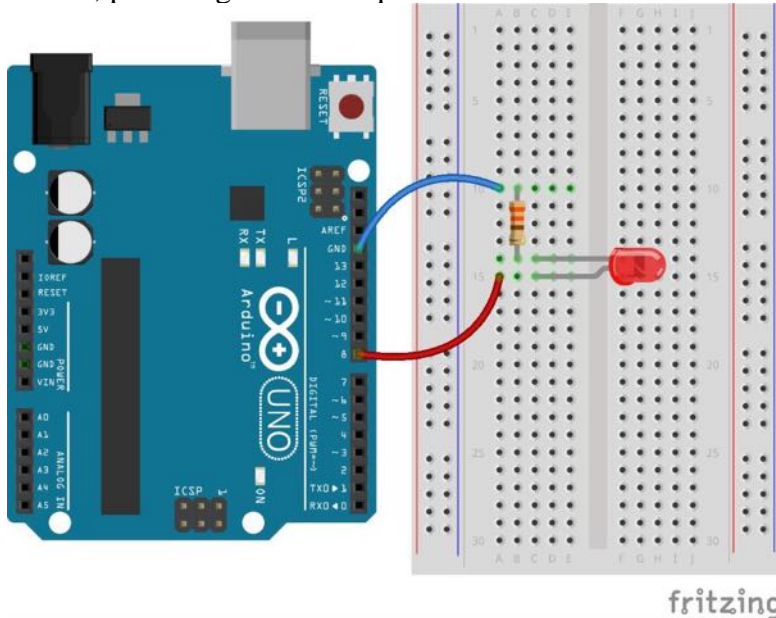
Iniziamo la pratica con qualcosa di molto semplice: collegare un LED. Secondo la descrizione precedente, qualsiasi pin I/O può essere utilizzato per questa connessione. Per ora, scegli l'uscita digitale n. 8. L'uscita digitale è l'uscita che può essere impostata su uno dei due stati (HIGH o LOW). Nel caso di Arduino, questo sarà 5V o 0V.

Materiale richiesto:

- - 1x scheda Arduino UNO e cavo USB;
- - 1x Breadboard;
- - 1x LED;
- - 1x resistenza da 330Ω;

- - 2x cavetti maschio-maschio.

Il sistema deve essere collegato secondo lo schema mostrato. Il LED è collegato in serie con un resistore (330Ω). Successivamente, collegiamo il piedino più lungo del LED (anodo) alla stessa colonna del connettore (rosso nel disegno) collegato al pin n. 8. Il secondo pin dovrebbe trovarsi nella stessa colonna di un piedino del resistore. L'altro piedino dovrebbe trovarsi nella stessa colonna del connettore (verde nella figura) collegato al pin di terra (GND). Ci sono 3 pin GND sulla scheda, puoi sceglierne uno qualsiasi.



La programmazione del circuito con il LED è molto semplice. Collega Arduino al computer con il cavo USB. Apri l'IDE di Arduino e scrivi il codice qui sotto. Quindi caricalo sulla scheda.

```
1 void setup() {  
2   pinMode(8, OUTPUT);  
3   digitalWrite(8, HIGH);  
4 }  
5  
6 void loop() {  
7 }
```

La funzione **pinMode** (Pin, Mode) permette di selezionare se il pin è un ingresso o un'uscita. Il pin è un numero intero compreso tra 0 e 13 e la modalità può essere:

- INPUT
- OUTPUT
- INPUT_PULLUP.

Se vogliamo controllare un'uscita, usiamo la modalità OUTPUT.

Grazie a questa impostazione è possibile impostare lo stato logico dell'uscita e quindi attivare il LED. A tale scopo viene utilizzata la funzione **digitalWrite** (Pin, Status). Lo stato è uno stato logico che può essere ALTO o BASSO.

In questo esempio, il LED è già stato collegato al pin di terra, quindi Arduino deve raggiungere uno stato alto: digitalWrite (8, HIGH).

Dopo aver impostato il pin su uno stato, il suo valore non cambierà fino a quando non si imposta un valore diverso. Pertanto, il programma di cui sopra farà sì che il LED rimanga sempre acceso.



Esercitazione pratica Ritardi - LED lampeggiante

In questo esercizio vogliamo far lampeggiare il LED. Per fare questo, abbiamo bisogno di una nuova funzione per inserire un ritardo. Lo schema elettrico è esattamente lo stesso di sopra. Il codice dovrebbe assomigliare a questo:

```
1 void setup() {
2   pinMode(8, OUTPUT); //Definir o pino 8 como saída
3 }
4
5 void loop() {
6   digitalWrite(8, HIGH); //Ligar o LED
7   delay(1000); //Esperar 1 segundo
8   digitalWrite(8, LOW); //Desligar o LED
9   delay(1000); //Esperar 1 segundo
10 }
```

Nella funzione loop, lo stato viene continuamente cambiato. I ritardi sono stati aggiunti al programma tramite la funzione delay. Questa funzione imposta un certo numero di millisecondi di ritardo.

Se non si introducono i ritardi, il sistema cambierà stato così rapidamente che sarebbe impossibile vedere il led lampeggiare ad occhio nudo. Puoi eseguire questo esperimento impostando 0 ms come ritardo.

Esercitazione pratica – Input – Struttura condizionale (if)

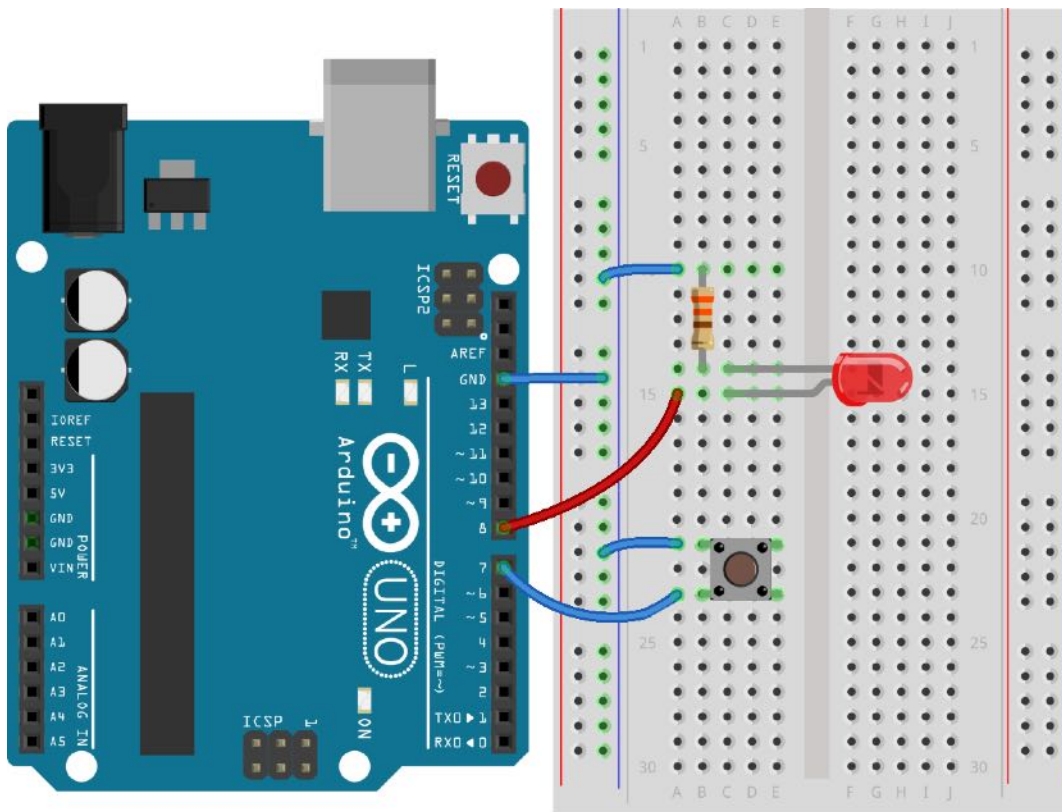
Spesso si desidera che il sistema reagisca a segnali esterni. Questa volta collegheremo ad Arduino un pulsante, oltre al LED.

Materiale richiesto:

- - 1x scheda Arduino UNO e cavo USB;
- - 1x Breadboard;
- - 1x LED;
- - 1x Pulsante;
- - 1x resistenza da 330Ω;
- - 5x connettori.

Il collegamento deve essere effettuato secondo lo schema sottostante. Da un lato il pulsante è stato collegato a terra (meno) e dall'altro al pin 7.





fritzing

L'obiettivo è creare un programma che accenda il LED quando si preme il pulsante. Il compito è molto semplice, ma inseriremo qualcosa di nuovo: le **istruzioni condizionali**.

Vogliamo che il programma sia permanentemente in uno dei due stati: LED acceso o spento. Per prima cosa dobbiamo leggere lo stato logico del pin di ingresso del pulsante.

Ricorda la **modalità INPUT_PULLUP** menzionata in precedenza. La prima parte del nome (input) significa ovviamente input, mentre la seconda (pullup) suggerisce l'inclusione di una resistenza interna che controlla lo stato dell'interruttore. Lo useremo ogni volta che collegheremo un pulsante ad Arduino.

Dobbiamo utilizzare la funzione **digitalRead** (pin), che invia ad Arduino informazioni HIGH o LOW, a seconda dello stato del pulsante. Tuttavia, la sola lettura dello stato dell'input non è sufficiente, dobbiamo far funzionare il programma in base a queste informazioni. Da qui l'istruzione condizionale (if). Grazie a ciò, è possibile eseguire una determinata parte del codice se vengono soddisfatti determinati requisiti o se vengono soddisfatte determinate condizioni. Ad esempio, se premiamo un pulsante.

```
1 | [...]
2 | void loop() {
3 |
4 | if (condição) {
5 |   /* A instrução é executada em loop somente
6 |   quando a condição é cumprida*/
7 | }
8 |
9 | }
```


Questa funzione può essere facilmente estesa con una parte di codice che verrà eseguita solo se la condizione non è verificata. L'istruzione else viene utilizzata per questo.

```
1 | [...]
2 | void loop()
3 | {
4 |
5 | if ( condição) {
6 | /* A instrução é executada em loop somente
7 | quando a condição é cumprida */
8 | } else {
9 | /* A instrução é executada somente
10 | quando a condição não é cumprida*/
11 | }
12 |
13 | }
```

Combinando le conoscenze acquisite, puoi creare un programma che esegua l'attività proposta. Rivedi il codice e caricalo su Arduino.

```
1 | void setup() {
2 | pinMode(8, OUTPUT); //LED como saída
3 | pinMode(7, INPUT_PULLUP); //Botão como entrada
4 | digitalWrite(8, LOW); //Desliga o LED
5 | }
6 |
7 | void loop()
8 | {
9 | if (digitalRead(7) == LOW) { //Se o botão for pressionado
10 | digitalWrite(8, HIGH); //Ligar o LED
11 | } else { //Se o botão não for pressionado
12 | digitalWrite(8, LOW); //Desligar LED
13 | }
14 | }
```

Esempio: interruttore della luce con "timer"

In questo esempio vogliamo, visto l'esempio precedente, che il LED rimanga acceso per 10 secondi dopo aver premuto il pulsante.

Puoi scrivere un programma adatto? Di sicuro! Se hai difficoltà, puoi sempre dare un'occhiata al nostro codice:

```
1 | void setup() {
2 | pinMode(8, OUTPUT); //LED como saída
3 | pinMode(7, INPUT_PULLUP); //Botão como entrada
4 | digitalWrite(8, LOW); //Desliga o LED
5 | }
6 |
7 | void loop()
8 | {
9 | if (digitalRead(7) == LOW) { //Se o botão for pressionado
10 | digitalWrite(8, HIGH); //Ligar o LED
11 | delay(10000); //Esperar 10 segundos
12 | digitalWrite(8, LOW); //Desligar LED
13 | }
14 | }
```

Esempio– Semaforo

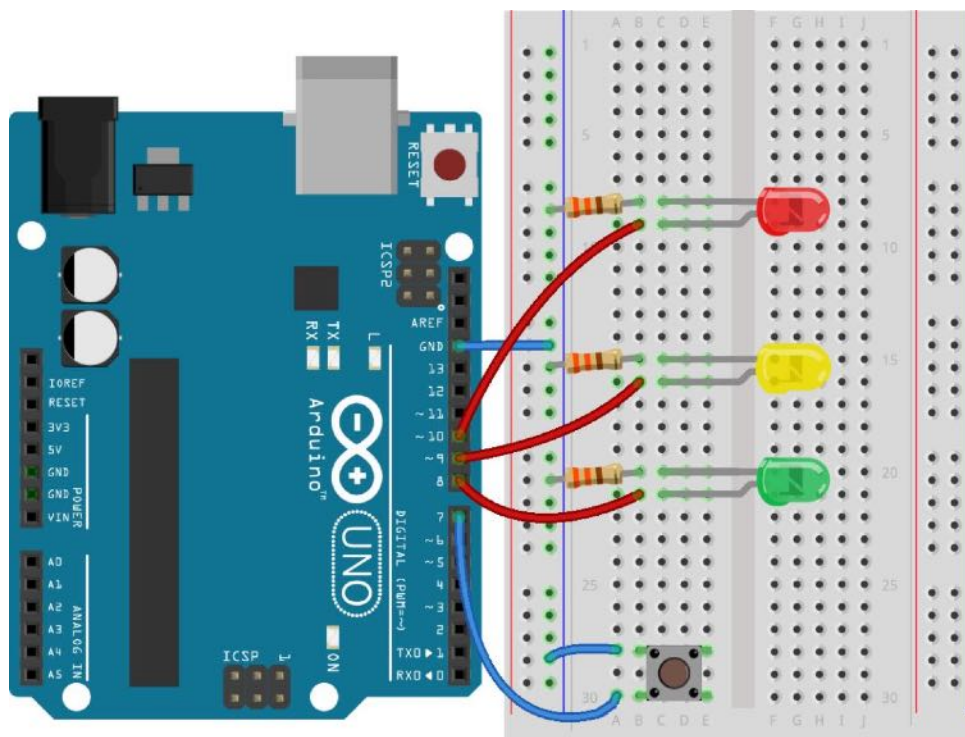
Il prossimo esempio è un sistema semaforico sequenziale. L'obiettivo principale è scrivere un programma che, dopo aver premuto il pulsante, mostri una corretta sequenza di luci. Supponiamo il seguente ciclo:

[...] Verde -> Giallo -> Rosso -> Giallo -> Rosso [...]

Materiale richiesto:

- 1x scheda Arduino UNO e cavo USB;
- 1x Breadboard;
- 1x pulsante;
- 1x LED Rosso;
- 1x LED Giallo;
- 1x LED Verde;
- 3x resistenze da 330Ω;
- 6x cavetti.

Quando si preme il pulsante, il sistema dovrebbe avviare la sequenza. Facciamolo a passi. Innanzitutto, accendi i 3 LED come mostrato di seguito:



fritzing

Scriveremo prima un programma che serve solo a configurare gli ingressi e le uscite.



```
1 void setup() {
2   pinMode(10, OUTPUT); //LED vermelho
3   pinMode(9, OUTPUT); //LED amarelo
4   pinMode(8, OUTPUT); //LED verde
5
6   pinMode(7, INPUT_PULLUP); //Botão
7
8   digitalWrite(10, LOW); //Desligar os LEDs
9   digitalWrite(9, LOW);
10  digitalWrite(8, LOW);
11 }
```

Ora che i LED e il pulsante sono impostati, scriviamo un programma che cambi le luci automaticamente, ogni 1 secondo. Lo sketch completo dovrebbe assomigliare a questo:

```
1 void setup() {
2   pinMode(10, OUTPUT); //LED vermelho
3   pinMode(9, OUTPUT); //LED amarelo
4   pinMode(8, OUTPUT); //LED verde
5
6   pinMode(7, INPUT_PULLUP); //Botão
7
8   digitalWrite(10, LOW); //Desligar os LEDs
9   digitalWrite(9, LOW);
10  digitalWrite(8, LOW);
11 }
12
13 void loop()
14 {
15  digitalWrite(10, LOW); //Vermelho
16  digitalWrite(9, LOW); //Amarelo
17  digitalWrite(8, HIGH); //Verde
18
19  delay(1000); //Esperar 1 segundo
20
21  digitalWrite(10, LOW); //Vermelho
22  digitalWrite(9, HIGH); //Amarelo
23  digitalWrite(8, LOW); //Verde
24
25  delay(1000); //Esperar 1 segundo
26
27  digitalWrite(10, HIGH); //Vermelho
28  digitalWrite(9, LOW); //Amarelo
29  digitalWrite(8, LOW); //Verde
30
31  delay(1000); //Esperar 1 segundo
32
33  digitalWrite(10, HIGH); //Vermelho
34  digitalWrite(9, HIGH); //Amarelo
35  digitalWrite(8, LOW); //Verde
36
37  delay(1000); //Esperar 1 segundo
38 }
```

Carica il programma su Arduino e controlla che tutto funzioni perfettamente.



Esercitazione pratica – La funzione While

Finora abbiamo utilizzato solo la funzione main e la funzione loop obbligatoria. Ora è il momento di conoscere un ciclo che possiamo usare all'interno dei nostri programmi.

Vedremo il ciclo while(), che funziona fintanto che una certa condizione è soddisfatta (true). Il suo funzionamento è mostrato nel seguente codice:

```
1  [...]
2  void loop()
3  {
4
5  while (condição) {
6
7  /* O código é executado em loop enquanto
8  a condição for verdadeira */
9
10 }
11
12 }
```

Per chiarezza, la funzione while esegue solo il codice che si trova tra le sue {} parentesi graffe. Il resto del codice non viene eseguito fino a quando non si esce dal while.

Sfruttiamo il sistema a semaforo che è stato impostato in precedenza, per scrivere un programma che faccia lampeggiare un LED alla volta quando si preme il pulsante. Questo esercizio è un compito più difficile del precedente.

Scopri il codice che ti proponiamo:

```
1  void setup() {
2  pinMode(10, OUTPUT); //LED vermelho
3  pinMode(9, OUTPUT); //LED amarelo
4  pinMode(8, OUTPUT); //LED verde
5
6  pinMode(7, INPUT_PULLUP); //Botão
7
8  digitalWrite(10, LOW); //Desligar LED
9  digitalWrite(9, LOW);
10 digitalWrite(8, LOW);
11 }
12
13 void loop() {
14
15 while (digitalRead(7) == LOW) { //Quando o botão é pressionado
16 digitalWrite(10, LOW); //Vermelho desligado
17 delay(1000);
18 digitalWrite(10, HIGH); //Vermelho ligado
19 delay(1000);
20 }
21
22 }
```

Questa volta, le sequenze dovrebbero essere visualizzate finché non si preme il pulsante.

Supponiamo che il pulsante venga premuto e rilasciato molto rapidamente. Il programma dovrebbe



assomigliare a questo:

```
1  void setup() {
2  pinMode(10, OUTPUT); //LED vermelho
3  pinMode(9, OUTPUT); //LED amarelo
4  pinMode(8, OUTPUT); //LED verde
5
6  pinMode(7, INPUT_PULLUP); //Botão
7
8  digitalWrite(10, LOW); //Desligar LED
9  digitalWrite(9, LOW);
10 digitalWrite(8, LOW);
11 }
12
13 void loop()
14 {
15 digitalWrite(10, LOW); //Vermelho
16 digitalWrite(9, LOW); //Amarelo
17 digitalWrite(8, HIGH); //Verde
18
19 while (digitalRead(7) == HIGH) {} //Quando o botão é pressionado
20
21 digitalWrite(10, LOW); //Vermelho
22 digitalWrite(9, HIGH); //Amarelo
23 digitalWrite(8, LOW); //Verde
24
25 while (digitalRead(7) == HIGH) {} //Quando o botão é pressionado
26
27 digitalWrite(10, HIGH); //Vermelho
28 digitalWrite(9, LOW); //Amarelo
29 digitalWrite(8, LOW); //Verde
30
31 while (digitalRead(7) == HIGH) {} //Quando o botão é pressionado
32
33 digitalWrite(10, HIGH); //Vermelho
34 digitalWrite(9, HIGH); //Amarelo
35 digitalWrite(8, LOW); //Verde
36
37 while (digitalRead(7) == HIGH) {} //Quando o botão é pressionado
38 }
```

In questo caso, il while è stato utilizzato in modo piuttosto strano. Come puoi vedere, non c'è niente dentro le parentesi graffe! Allora come funziona il programma? Il programma utilizza i cicli per fermarsi.

Come funziona?

1. Iniziamo ad accendere i LED secondo una sequenza;
2. Entriamo nella funzione ciclo while(), che è immediatamente sotto;
3. Le parentesi graffe sono vuote, quindi il programma è sempre in loop, senza fare nulla;
4. Solo dopo che il pulsante è stato premuto (la condizione diventa falsa) il programma esce dal ciclo;
5. Viene attivata la sequenza successiva e la situazione si ripete.

Controlliamo ora il programma in pratica!!



Che cosa sta succedendo? Funziona tutto come dovrebbe? Ovviamente no! Anche quando il pulsante viene premuto per un breve periodo, a volte il programma funziona correttamente e talvolta salta alcune posizioni. Perché succede?

Come ricorderete, il processore, in forma semplificata, esegue circa 16 milioni di operazioni al secondo. Pertanto, premendo il pulsante, il processore potrà accedere a tutti gli stati della nostra segnalazione. Detto questo, dopo aver rilasciato il pulsante potrebbe esserci una scelta casuale nella sequenza.

Come risolvere questo problema? Molto semplice! È sufficiente cambiare il programma, in modo che il cambio di luce non avvenga più spesso di, ad esempio, ogni secondo. Per fare ciò possiamo utilizzare la già nota funzione `delay()`.

```
1  void setup() {
2  pinMode(10, OUTPUT); //LED vermelho
3  pinMode(9, OUTPUT); //LED amarelo
4  pinMode(8, OUTPUT); //LED verde
5
6  pinMode(7, INPUT_PULLUP); //Botão
7
8  digitalWrite(10, LOW); //Desligar LED
9  digitalWrite(9, LOW);
10 digitalWrite(8, LOW);
11 }
12
13 void loop()
14 {
15 digitalWrite(10, LOW); //Vermelho
16 digitalWrite(9, LOW); //Amarelo
17 digitalWrite(8, HIGH); //Verde
18
19 delay(1000); //Parar o programa durante 1 segundo
20 while (digitalRead(7) == HIGH) {} //Quando o botão é pressionado
21
22 digitalWrite(10, LOW); //Vermelho
23 digitalWrite(9, HIGH); //Amarelo
24 digitalWrite(8, LOW); //Verde
25
26 delay(1000); //Parar o programa durante 1 segundo
27 while (digitalRead(7) == HIGH) {} //Quando o botão é pressionado
28
29 digitalWrite(10, HIGH); //Vermelho
30 digitalWrite(9, LOW); //Amarelo
31 digitalWrite(8, LOW); //Verde
32
33 delay(1000); //Parar o programa durante 1 segundo
34 while (digitalRead(7) == HIGH) {} //Quando o botão é pressionado
35
36 digitalWrite(10, HIGH); //Vermelho
37 digitalWrite(9, HIGH); //Amarelo
38 digitalWrite(8, LOW); //Verde
39
40 delay(1000); //Parar o programa durante 1 segundo
41 while (digitalRead(7) == HIGH) {} //Quando o botão é pressionado
42 }
```

Ora dovrebbe funzionare perfettamente!

È importante notare che le condizioni nella funzione while () possono essere combinate e molto più complesse.

UART e variabili

Come funziona l'UART?

Il suo principio di funzionamento si basa sull'invio seriale di una sequenza di bit, che vengono trasformati in informazioni. Un set di dati (data frame) viene trasmesso nel modo seguente:



La trasmissione inizia con il bit di start, contrassegnato con BS nella figura. È sempre un bit a zero logico. Poi, a seconda della configurazione, ci sono 7, 8 o 9 bit di dati (contrassegnati da B0-B7) che sono le informazioni da inviare. Il bit di stop (indicato con BK) è un bit logico - termina la trasmissione.

Quando si utilizza l'UART in Arduino, dobbiamo prendere in considerazione due pin:

- Tx – invio dati (pin 1);
- Rx – ricezione dati (pin 0).

Affinché la trasmissione funzioni correttamente, è necessario impostare la stessa velocità di trasferimento dati su entrambi i sistemi, nota come **baud rate**. Questa specifica il numero di bit trasmessi al secondo. I valori più comunemente usati sono: 9.600 e 112.500.

Anche il computer con il quale intendiamo stabilire la comunicazione deve essere dotato dell'apposita interfaccia. Sfortunatamente, i produttori di PC hanno smesso di inserire la porta seriale RS-232, che qualche anno fa faceva parte della dotazione di base della maggior parte dei computer.

Rimane solo la comunicazione USB. Sfortunatamente, questo rende le cose un po' più difficili. Pertanto, vengono solitamente utilizzati convertitori USB-UART, il che semplifica notevolmente il lavoro. La buona notizia è che non devi preoccuparti di questo quando usi Arduino, il convertitore è già integrato sulla scheda.



Pertanto, tutto ciò che devi fare è collegare Arduino al computer tramite il cavo USB (lo stesso utilizzato nella programmazione).

Passiamo agli esempi pratici. I primi due richiedono solo il collegamento di Arduino tramite USB al computer. Aggiungeremo le periferiche al sistema solo in un secondo momento.

Practical Exercise - UART communication

Materiale richiesto:

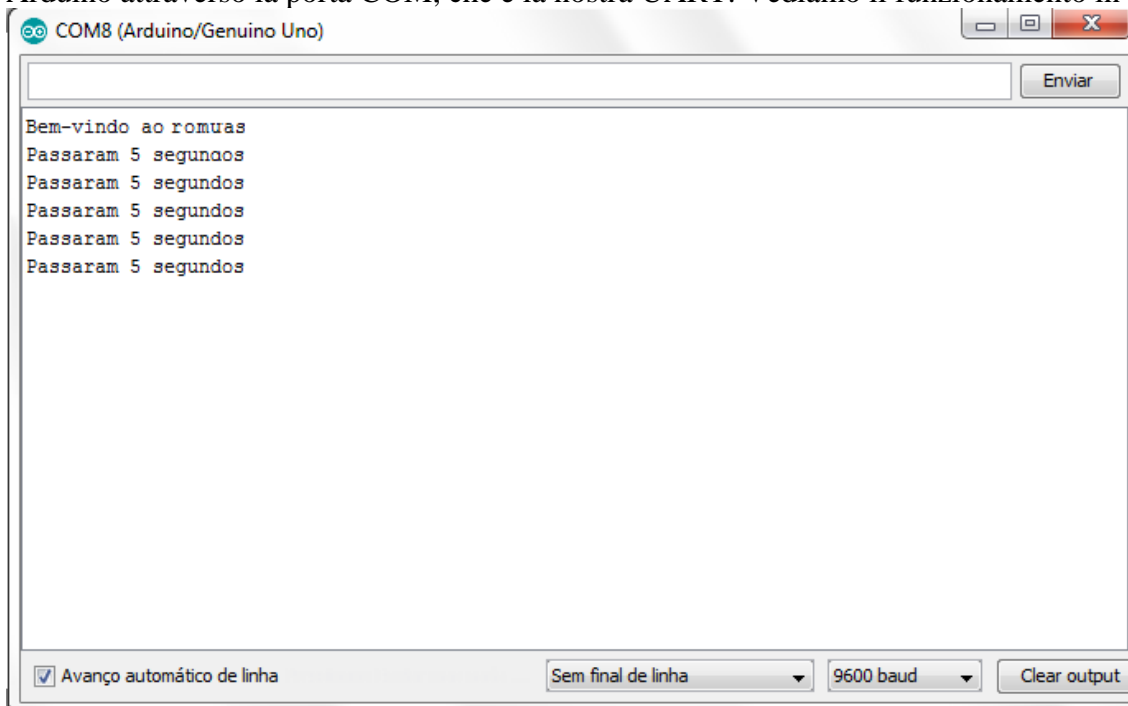
- - 1x scheda Arduino UNO e cavo USB;

Lo scopo del programma qui sotto è molto semplice: inviare un testo al computer:

```
1 void setup(){
2   Serial.begin(9600); //Configuração da velocidade de transmissão
3   Serial.println("Bem-vindo ao ROMUAS!"); //Envio de texto único
4 }
5 void loop() {
6   delay(5000);
7   Serial.println("Passaram 5 segundos"); //Envio de texto em loop
8 }
```

Dopo aver caricato il programma di cui sopra, apparentemente non succede nulla. Per verificarne il funzionamento è necessario selezionare dal menu di Arduino: Tools -> Serial Monitor.

Successivamente, si aprirà una nuova finestra. Qui possiamo osservare cosa viene inviato a/da Arduino attraverso la porta COM, che è la nostra UART. Vediamo il funzionamento in pratica:



Ora analizziamo il programma. La prima cosa da fare è impostare il baud-rate. A questo scopo viene utilizzata la funzione `Serial.begin()`, dove tra parentesi c'è il baud-rate. In questo caso 9600 baud/sec. La funzione `Serial.println()` viene utilizzata per inviare un'informazione (testo o numeri).

Il testo "Benvenuti in ROMUAS!" viene visualizzato una sola volta, perché è inserito nella funzione `setup()`; come ricorderete dal precedente capitolo del corso, le istruzioni inserite in questa funzione vengono eseguite una sola volta.

La trasmissione può essere osservata anche nei LED integrati di Arduino (Tx e Rx)! Si accendono durante il trasferimento dei dati alla/dalla scheda.

Esercitazione pratica - Interazione con il programma

Naturalmente, le informazioni non devono essere inviate continuamente tramite UART; la trasmissione e la ricezione possono anche avvenire una sola volta, quando richiesto. Questo è molto utile, ad esempio, per diagnosticare il funzionamento del sistema o segnalare diverse occorrenze.

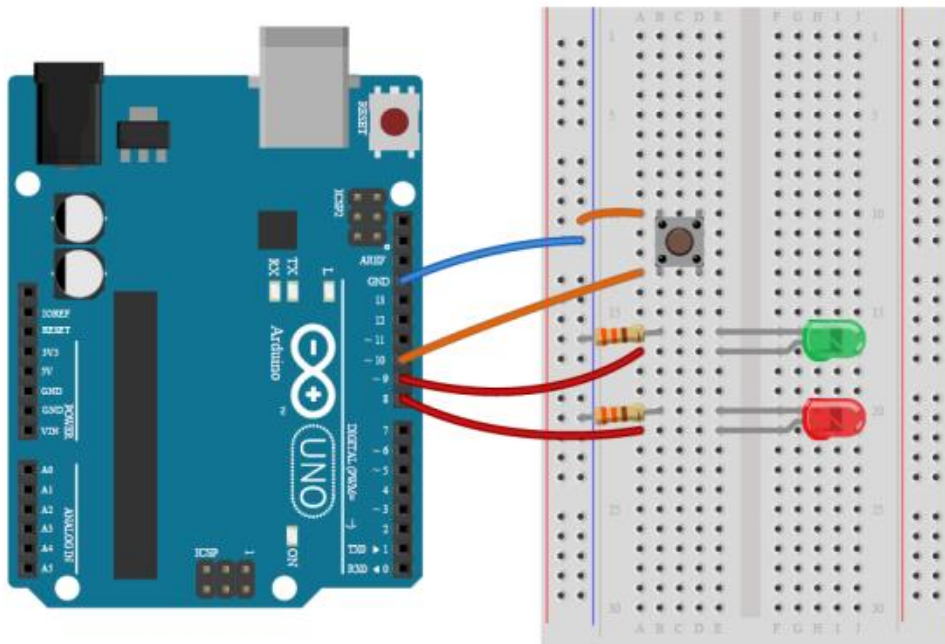
Utilizzando le conoscenze acquisite in precedenza, puoi scrivere uno sketch che attiva un LED quando una finestra è aperta. Ovviamente non utilizzeremo un sensore di apertura finestra, lo

simuleremo utilizzando componenti più semplici. Un pulsante sostituirà il sensore e due LED serviranno per la segnalazione.

Materiale richiesto:

- 1x scheda Arduino UNO e cavo USB;
- 1x Breadboard;
- 1x LED rosso;
- 1x LED verde;
- 1x pulsante;
- 2x resistenze da 330Ω;
- 5x connettori.

Effettuare i collegamenti come segue:



Quando la finestra è chiusa (pulsante premuto), il LED verde è acceso. Quando apriamo il circuito (smettiamo di premere il pulsante) il LED rosso si accende e sul monitor seriale leggeremo il messaggio "Attenzione! Allarme! La finestra è aperta!".



```
1 void setup(){
2   Serial.begin(9600); //Configuração da velocidade de transmissão
3
4   pinMode(8, OUTPUT); //LED vermelho
5   pinMode(9, OUTPUT); //LED verde
6   pinMode(10, INPUT_PULLUP); //Botão
7
8   digitalWrite(8, LOW); //Desligar LEDs
9   digitalWrite(9, LOW);
10  }
11
12  void loop() {
13    if (digitalRead(10) == LOW) { //Se o botão for pressionado
14      digitalWrite(9, HIGH); //Liga o LED verde
15      digitalWrite(8, LOW); //Desliga o LED vermelho
16    } else { //Se o botão não for pressionado
17      digitalWrite(9, LOW); //Desliga o LED verde
18      digitalWrite(8, HIGH); //Liga o LED vermelho
19      Serial.println("Atenção! Alarme! A janela está aberta!");
20    }
21  }
```

Controlliamo come funziona il programma! Sfortunatamente, non funziona molto bene. Le informazioni sugli allarmi vengono inviate continuamente. Preferiamo che l'allarme venga inviato una sola volta. Sai come cambiarlo? Certo, inserendo la funzione while!

```
1 void setup(){
2   Serial.begin(9600); //Configuração da velocidade de transmissão
3
4   pinMode(8, OUTPUT); //LED vermelho
5   pinMode(9, OUTPUT); //LED verde
6   pinMode(10, INPUT_PULLUP); //Botão
7
8   digitalWrite(8, LOW); //Desligar LEDs
9   digitalWrite(9, LOW);
10  }
11
12  void loop() {
13
14    if (digitalRead(10) == LOW) { //Se o botão for pressionado
15      digitalWrite(9, HIGH); //Liga o LED verde
16      digitalWrite(8, LOW); //Desliga o LED vermelho
17    } else { //Se o botão não for pressionado
18      digitalWrite(9, LOW); //Desliga o LED verde
19      digitalWrite(8, HIGH); //Liga o LED vermelho
20      Serial.println("Atenção! Alarme! A janela está aberta!");
21
22      while (digitalRead(10) == HIGH) {
23        //Criação de um loop vazio para a janela voltar a fechar
24        delay(25); //Atraso de 25ms dentro do loop para minimizar interferências
25      }
26    }
27  }
28 }
```

Istruzione #define

Nel tempo, i nostri programmi cresceranno considerevolmente. Cosa succede se dobbiamo modificare la connessione fisica, ad esempio, di un LED o di un pulsante? Cambiare il numero di pin dell'intero codice sarebbe abbastanza difficile.





L'istruzione `#define` aiuta in questo senso. Permette di definire un "simbolo" per un determinato pin, che verrà sostituito dal suo numero prima della compilazione, in qualsiasi punto del codice. Per esempio:

```
1 | #define ledPin 8
2 |
3 | void setup() {
4 |   pinMode(ledPin, OUTPUT); //Configuração do pino 8 como saída
5 | }
6 |
7 | void loop() {
8 |   digitalWrite(ledPin, HIGH); //Liga o LED
9 |   delay(1000); //Espera 1 segundo
10 |  digitalWrite(ledPin, LOW); //Desliga o LED
11 |  delay(1000); //Espera 1 segundo
12 | }
```

Mettendo la riga: `#define ledPin 8` all'inizio del codice, facciamo in modo che, prima della compilazione, qualsiasi parte del programma che ha come pin "ledPin", venga automaticamente trasformata nel numero definito per esso, ovvero 8. Naturalmente il nome del pin può essere diverso, l'importante è stabilire un nome univoco, che vi aiuterà a scrivere programmi lunghi.

IMPORTANTE: dopo l'istruzione `#define` non viene utilizzato il punto e virgola (;).

Di seguito la nuova e migliorata versione del codice sensore apertura finestra:

```
1 | #define LEDvermelho 8
2 | #define LEDverde 9
3 | #define Botão 10
4 |
5 | void setup(){
6 |   Serial.begin(9600); //Configuração da velocidade de transmissão
7 |
8 |   pinMode(LEDvermelho, OUTPUT); //LED vermelho como saída
9 |   pinMode(LEDverde, OUTPUT); //LED verde como saída
10 |  pinMode(Botão, INPUT_PULLUP); //Botão
11 |
12 |  digitalWrite(LEDvermelho, LOW); //Desligar LEDs
13 |  digitalWrite(LEDverde, LOW);
14 | }
15 |
16 | void loop() {
17 |
18 |  if (digitalRead(Botão) == LOW) { //Se o botão for pressionado
19 |    digitalWrite(LEDverde, HIGH); //Liga LED verde
20 |    digitalWrite(LEDvermelho, LOW); //Desliga LED vermelho
21 |  } else { //Se o botão não for pressionado
22 |    digitalWrite(LEDverde, LOW); //Desliga LED verde
23 |    digitalWrite(LEDvermelho, HIGH); //Liga LED vermelho
24 |    Serial.println("Atenção! Alarme! A janela está aberta!");
25 |
26 |    while (digitalRead(Botão) == HIGH) {
27 |      //Criação de um loop vazio para a janela voltar a fechar
28 |      delay(25); //Atraso de 25ms dentro do loop para minimizar interferências
29 |    }
30 |
31 |  }
32 | }
```

Variabili

Prima di passare ad altri programmi (incluso l'invio di informazioni ad Arduino tramite UART), dobbiamo sapere cosa sono le variabili e come funzionano.

Le variabili, in generale, sono dichiarazioni di qualche tipo di informazione richiesta dal codice.

Possono essere caratteri, parole o numeri. La maggior parte delle volte ci impareremo in variabili numeriche.



Quando sono necessarie le variabili? Quando vogliamo memorizzare un valore ed eseguire vari tipi di operazioni con esso. Una variabile, proprio come una funzione, può avere un tipo specifico di informazioni, sul tipo di dati che può memorizzare.

Di seguito è riportato un elenco dei tipi di variabili più importanti:

```
1 | boolean logica = false; //Boolean - verdadeiro (true) ou falso (false)
2 |
3 | int numero = 30000; //Int - números inteiros no intervalo de -32768 a 32767
4 | long numeroGrande = 2000000; //Long - números inteiros no intervalo de
5 | -2147483648 a 2147483647
6 |
7 | float numeroRacional = 6.28; //Float - números racionais que ocupem até 4
8 | bytes de memória
9 |
   | char caractere = 'a'; //Char - caracteres
   | String frase = "Bem-vindo ao blog ElectroFun!"; //String - sequência de
   | caracteres
```

NOTA: I valori massimi che possono essere scritti in una variabile dipendono dalla scheda Arduino utilizzata. I valori di cui sopra sono appropriati per Arduino UNO.

Ecco i tipi di variabili che useremo più spesso:

- **Boolean** - come accennato, viene utilizzato per memorizzare valori veri o falsi. Questo tipo di variabile è generalmente utilizzato per segnalare occorrenze o condizioni di controllo;
- **Int** - è la variabile più comune per la memorizzazione di numeri interi. Può memorizzare informazioni come il numero di pressioni di tasti, quante volte si è verificata una determinata situazione o un valore restituito dal sensore di distanza. E, naturalmente, è possibile eseguire operazioni matematiche sulle variabili;
- **String** - è un insieme di caratteri, cioè, in forma semplificata, possiamo memorizzare una parola, una frase, una didascalia, un messaggio, ecc.

Dichiarazione di variabili

Per usare una variabile, devi dichiararla, cioè indicare al compilatore il suo tipo e il suo nome. Il nome di ogni variabile inizia sempre con una lettera, mai con un numero, e non può contenere spazi. La dichiarazione di una variabile dovrebbe essere fatta come segue:

```
1 | tipo nome = 0;
```

IMPORTANTE: Si noti che il simbolo = viene utilizzato per assegnare un valore ad una variabile e il simbolo == viene utilizzato per confrontare l'uguaglianza tra variabili e valori.

È anche importante notare che se una variabile viene inserita all'interno di una funzione o sottoprogramma, sarà invisibile (non possiamo usarla) in altre funzioni. Vediamo:

```
1 | int variavel = 0; //Variável global - pode ser usada em qualquer parte do
2 | programa
3 |
4 | void setup() {
5 |   int variavel2 = 0; //Variável local - só pode ser utilizada dentro da função
6 |   setup()
7 | }
8 |
9 | void loop() {
   |   int variavel3 = 0; //Variável local - só pode ser utilizada dentro da função
   |   loop()
   | }
```

È normale che al momento tutto sia ancora un po' confuso, ma con il tempo imparerai tutto con la pratica. All'inizio, per facilitare l'apprendimento, utilizzeremo solo variabili globali. Ai programmatori esperti potrebbe non piacere; vedremo più avanti un utilizzo migliore delle variabili. C'è un altro punto importante da sottolineare: i nomi delle variabili. Ricorda di dare nomi alle variabili che ne determinano lo scopo. Ad esempio, invece di:

```
1 | string xx = "Manuel";
```



utilizzare un nome univoco che identifichi la variabile:

```
1 | string nomePessoa = "Manuel";
```

Il problema può sorgere quando lo scopo delle variabili è più complicato da definire. Non aver paura di creare nomi come:

```
1 | int vMotorEsquerdo = 100; //Velocidade do motor esquerdo
```

La cosa più importante è rendere il codice leggibile e comprensibile!

Esercitazione pratica - Variabili

La teoria è finita, ora è il momento di mettere in pratica ciò che abbiamo imparato. Iniziamo con qualcosa di molto semplice. Per iniziare, vediamo un programma che scrive un valore variabile, che aumenta con ogni ciclo.

```
1 | int contador = 0; //Declaração da variável
2 |
3 | void setup() {
4 |   Serial.begin(9600); //Configuração da velocidade de transmissão
5 | }
6 |
7 | void loop() {
8 |   Serial.println(contador); //Enviar o valor da variável
9 |   contador = contador + 1; //Somar 1 ao valor do contador
10 |  delay(100); //Atraso para tornar o programa mais perceptível
11 | }
```

Ovviamente la dichiarazione della variabile è stata posta all'inizio, al di fuori di ogni funzione.

Grazie a questo, possiamo accedere alla variabile ovunque nel programma.

Prima viene impostata e inizializzata la velocità di trasmissione, quindi la funzione loop () esegue 3 azioni:

- 1) Richiama la locazione di memoria dove abbiamo dichiarato una variabile chiamata "counter/contador" e invia il valore trovato tramite UART;
- 2) Incrementa di +1 il valore ricevuto dal contatore;
- 3) Attende 100 ms (per una migliore comprensione) e torna all'inizio del ciclo.

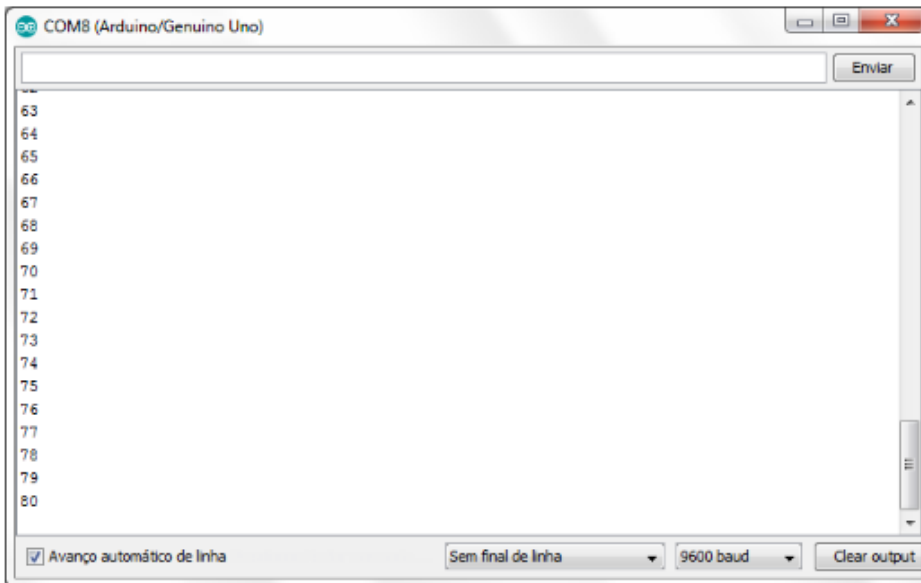
Spiegheremo ora il punto 2, che è l'incremento del valore della variabile. L'istruzione utilizzata è stata la seguente:

```
1 | contador = contador + 1; //Somar 1 ao valor do contador
```

Da un punto di vista matematico, il segno "=" significa uguaglianza, la riga sopra non dovrebbe essere corretta. Tuttavia, in programmazione, il segno "=" significa assegnazione. In pratica, il codice di cui sopra dovrebbe essere inteso come la seguente operazione:

- 1) Recuperare il valore dalla variabile contatore;
- 2) Aggiungere 1 al valore ricevuto;
- 3) Ottenere il risultato dell'operazione e assegnarlo alla variabile stessa.

Scarica il programma su Arduino e controlla che tutto funzioni correttamente. Ovviamente per vedere i risultati dovrai aprire il monitor seriale.



Trasmmissione bidirezionale con Arduino

Naturalmente la comunicazione, per essere utile, deve avvenire in modo bidirezionale. Finora, è stato Arduino a inviarci informazioni. È ora che rispondiamo!

L'obiettivo del primo programma è "ascoltare" il nostro nome. Quando gli inviamo il nome, Arduino dovrebbe rispondere con il seguente messaggio "Ciao, Nome!", dove, ovviamente, il nome sarà quello precedentemente inviato.

```
1  String dadosRecebidos = ""; //Conjunto vazio de dados recebidos
2
3  void setup() {
4  Serial.begin(9600); //Configuração da velocidade de transmissão
5  }
6
7  void loop() {
8  if(Serial.available() > 0) { //Se o Arduino receber dados
9  dadosRecebidos = Serial.readStringUntil('\n'); // Lê os dados recebidos e
10 guarda na própria variável
11 Serial.println("Bem-vindo " + dadosRecebidos + "!"); //Mostrar a mensagem
12 }
}
```

Per prima cosa dichiariamo la variabile `dataReceived`, in cui verrà copiato il set di caratteri ricevuto (nome). Successivamente, come al solito, impostiamo il baud rate. Quindi introduciamo una nuova funzione: **`Serial.available()`**. Questa fornisce il numero di byte che sono stati ricevuti e sono in attesa di supporto da parte di Arduino.

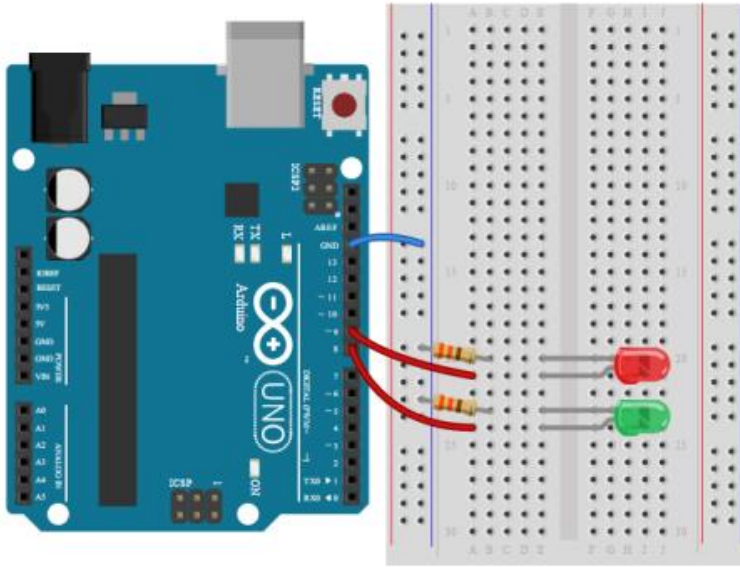
Nel caso in cui il dato sia già disponibile (maggiore di 0), viene inviato alla variabile `dadosRecebidos`. Questa operazione viene eseguita tramite la funzione **`.readStringUntil(carattere finale)`** che copia i dati dal buffer finché non incontra il "carattere finale" (in questo caso, `"\n"`).

Esercitazione pratica sull'interazione con il sistema - Controllo LED tramite UART

Materiale richiesto:

- 1x scheda Arduino UNO e cavo USB;
- 1x Breadboard;
- 1x LED verde;
- 1x LED rosso;
- 2x resistenza da 330Ω;
- 3x cavetti.

In questo esercizio utilizzeremo la possibilità di inviare testo ad Arduino per controllare i LED. Per fare ciò, collegare due LED secondo lo schema seguente (LED sui pin 8 e 9):



L'obiettivo del programma è quello di accendere il LED verde o rosso per 1 secondo quando viene inviato un comando appropriato ad Arduino. Il codice finale è il seguente:

```
1  #define verde 8
2  #define vermelho 9
3
4  String dadosRecebidos = ""; //Conjunto vazio de dados recebidos
5
6  void setup() {
7  Serial.begin(9600); //Configuração da velocidade de transmissão
8  pinMode(verde, OUTPUT); //Configuração dos LEDs como saídas
9  pinMode(vermelho, OUTPUT);
10 |
11 digitalWrite(verde, LOW); //Desligar LEDs
12 digitalWrite(vermelho, LOW);
13 }
14
15 void loop() {
16 if(Serial.available() > 0) { //Se o Arduino receber dados
17 dadosRecebidos = Serial.readStringUntil('\n'); //Lê os dados recebidos e guarda
18 na própria variável
19
20 if (dadosRecebidos == "verde") { //Se escrever a palavra "verde"
21 digitalWrite(verde, HIGH); //Liga o LED verde
22 delay(1000); //Espera 1 segundo
23 digitalWrite(verde, LOW); //Desliga o LED verde
24 }
25
26 if (dadosRecebidos == "vermelho") { //Se escrever a palavra "vermelho"
27 digitalWrite(vermelho, HIGH); //Liga o LED vermelho
28 delay(1000); //Espera 1 segundo
29 digitalWrite(vermelho, LOW); //Desliga o LED vermelho
30 }
31 }
```

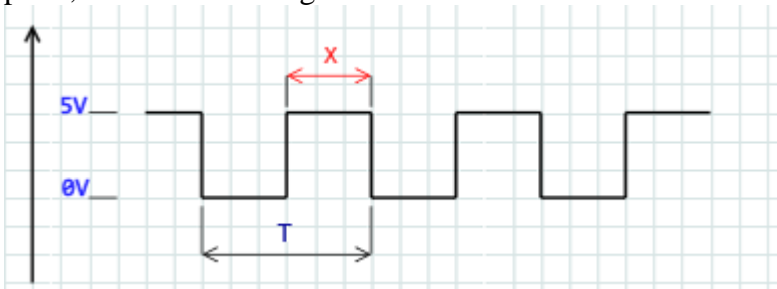
Diamo ora un'occhiata a come funziona il programma. Inizialmente vengono impostati i numeri dei pin con i LED e dichiarata la variabile in cui vengono copiati i dati ricevuti. Nel ciclo viene verificato se Arduino ha ricevuto i dati. In tal caso, si verifica se questo dato corrisponde a uno dei colori. Successivamente, il LED del colore indicato si accende.

Cos'è il segnale PWM?

Supponiamo di collegare un LED al microcontrollore e programmarlo per lampeggiare ciclicamente. Il LED rimane acceso per un secondo, il secondo successivo rimane spento e così via:

```
1 void setup() {
2   pinMode(3, OUTPUT); //Configuração do LED como saída
3 }
4
5 void loop() {
6   digitalWrite(3, HIGH); //Liga o LED
7   delay(1000); //Espera 1 segundo
8   digitalWrite(3, LOW); //Desliga o LED
9   delay(1000); //Espera 1 segundo
10 }
```

Se tracciassimo un grafico per rappresentare la variazione di tensione in funzione del tempo per il pin 3, otterremmo la seguente forma d'onda:



Il valore contrassegnato come x è il tempo in cui il LED è acceso. T è il periodo di tempo tra una accensione e la successiva. A sua volta, il suo inverso, cioè $1/T$, indica la frequenza. Il rapporto tra il tempo in cui il LED è acceso e il tempo in cui il LED è spento è 1:1. In altre parole, il LED è attivo per il 50% del funzionamento del programma. Questo è chiamato ciclo di lavoro (**duty cycle**).

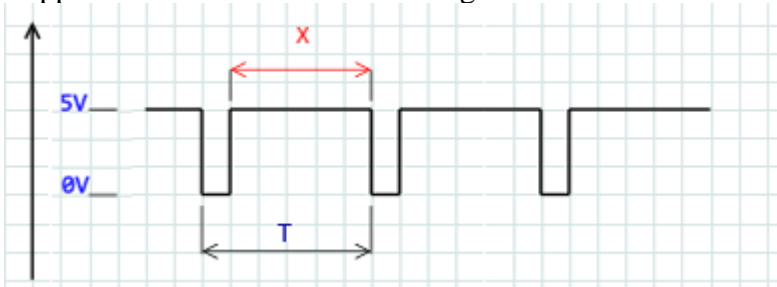
Per riassumere le informazioni sul segnale di cui sopra:

- Ampiezza (valore massimo): 5V;
- Periodo (ciclo): 2 secondi;
- Frequenza: $\frac{1}{2} \text{ s} = 0.5 \text{ Hz}$;
- Duty cycle: 50%.

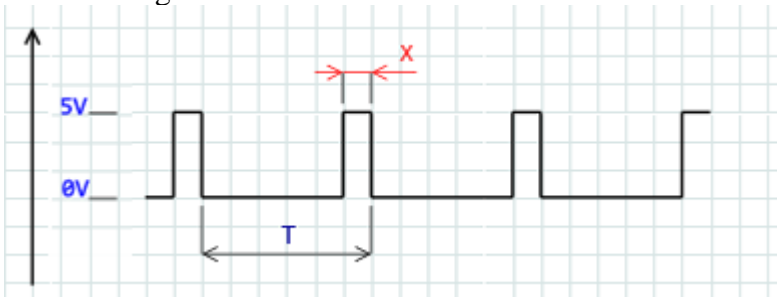
Vediamo ora un altro esercizio simile, con un ciclo di lavoro diverso, pur mantenendo il periodo. Come farlo? Basta estendere il tempo di funzionamento del LED riducendo il tempo in cui è spento. Per esempio:

```
1 void setup() {
2   pinMode(3, OUTPUT); //Configuração do LED como saída
3 }
4
5 void loop() {
6   digitalWrite(3, HIGH); //Ligar o LED
7   delay(1667);
8   digitalWrite(3, LOW); //Desligar o LED
9   delay(333);
10 }
```


Questa volta il LED rimane acceso circa 5/6 del tempo. Quindi il ciclo di lavoro è di circa l'83%. Rappresentando la situazione in un grafico otteniamo:



Al contrario, se invertiamo i due tempi (on e off), il duty cycle del segnale diventa di circa il 17%. Vediamo il grafico:



Dai un'occhiata agli esempi sopra. Quale parametro è cambiato in ciascun esempio? La risposta è semplice: il duty cycle. La frequenza è rimasta la stessa.

Se i ritardi inseriti nei codici di cui sopra sono molto più piccoli, la frequenza del segnale è molto più alta... Congratulazioni! Hai appena compreso il principio del PWM. È un metodo per modulare un segnale rettangolare regolando l'ampiezza dell'impulso.

A cosa serve il segnale PWM?

Questo segnale è usato molto frequentemente. Con esso puoi controllare la luminosità di un LED, la posizione di un servo e la velocità di funzionamento di un motore! Come vedrai, ha innumerevoli applicazioni nella robotica, così come in altri progetti fai-da-te.

Esercizio pratico PWM - Controllo luminosità LED

È il momento del primo esempio pratico di applicazione PWM. Per ora, creiamo un programma molto semplice, il cui obiettivo sarà impostare la luminosità di un LED.

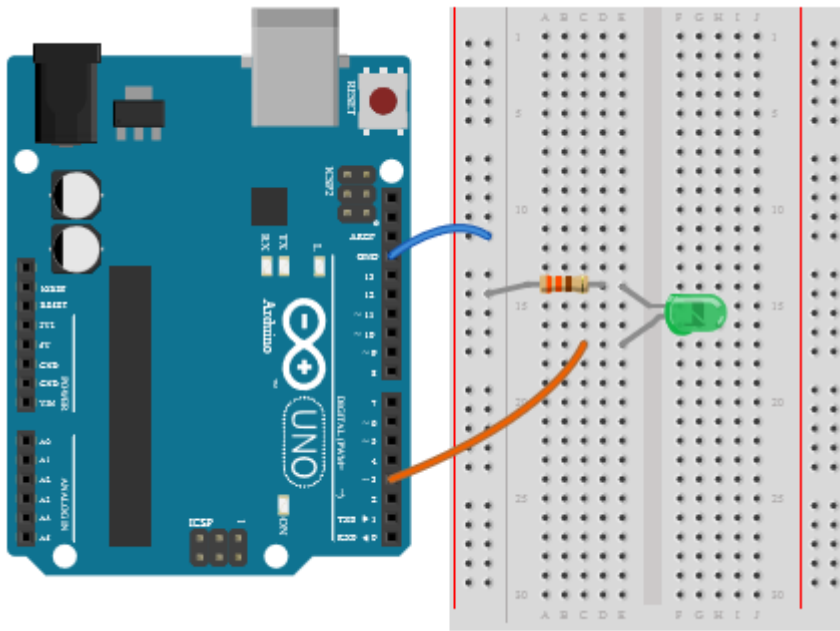
Materiale necessario:

- 1x scheda Arduino UNO e cavo USB;
- 1x Breadboard;
- 1x LED;
- 1x resistenza da 330Ω;
- 2x cavetti.

Arduino è dotato di 6 canali PWM. Ogni uscita, su cui possiamo ottenere il segnale PWM, è stata contrassegnata sulla scheda con una tilde "~".

Ogni canale PWM disponibile in Arduino UNO è lungo 8 bit. Ciò significa che, il segnale che vogliamo ricevere in uscita, può essere definito da un numero da 0 a 255, dove 255 significa 100% duty cycle.

Per eseguire il primo esercizio è necessario collegare il LED al pin 3 come segue:



Ora scriviamo il codice. L'obiettivo è che il LED si illumini lentamente.

```

1 | #define pinoLED 3
2 |
3 | int dutyCycle = 0;
4 | int mudanca = 5;
5 |
6 | void setup() {
7 |   pinMode(pinoLED, OUTPUT); //Configuração do LED como saída
8 | }
9 |
10 | void loop() {
11 |   analogWrite(pinoLED, dutyCycle); //Gerar um sinal com determinado duty cycle
12 |
13 |   if (dutyCycle < 255) { //Se o duty cycle for menor do que 100%
14 |     dutyCycle = dutyCycle + mudanca; //Aumenta o duty cycle
15 |   } else {
16 |     dutyCycle = 0; //Se o duty cycle for igual a 100%, volta ao início
17 |   }
18 |
19 |   delay(50); //Pequeno atraso para tornar o efeito visível
20 | }
  
```

Speriamo di essere stati chiari. Possiamo ora discutere della nuova funzione che è stata inserita: **analogWrite** (pin, duty cycle). Il suo scopo è generare il segnale PWM al pin selezionato con il duty cycle indicato.

Il programma di cui sopra intende aumentare periodicamente il ciclo di lavoro da zero ad un valore immediatamente inferiore a 255 (100%). Quando viene raggiunto il ciclo di lavoro massimo, il LED si spegne e il processo viene ripetuto.

Servomotore

È ora di usare il servomotore! Quello che useremo è l'SG90, un tipo micro, uno dei più piccoli disponibili sul mercato. Tuttavia, le sue dimensioni non influiscono sulla modalità di controllo. Una volta compreso il principio di funzionamento, puoi applicarlo a servomotori più grandi, più potenti e più veloci.



Cos'è un servomotore?

Un servomeccanismo è un motore, un riduttore e un controller in un unico dispositivo. Tuttavia, questi motori non sono progettati per eseguire rotazioni complete. Il più delle volte, i servo hanno un angolo di rotazione di 0-180°. È importante sapere che conoscono la tua posizione attuale, quindi non devi preoccuparti degli errori di posizione.

I principi più importanti dell'utilizzo dei servo:

1. La posizione dell'albero non deve essere girata manualmente se non serve! Potrebbe danneggiare i delicati ingranaggi in plastica;
2. Non alimentare il servo direttamente dall'alimentatore utilizzata nel resto del sistema. Ogni motore riceve una corrente relativamente elevata, soprattutto all'inizio del movimento. Ciò può disturbare il funzionamento degli altri dispositivi e, in casi estremi, danneggiarli.

Come funziona il servo?

Come fa il servomeccanismo a sapere in quale direzione girare? Grazie al driver integrato. È lui che, sulla base del segnale PWM fornito, controlla il motore.

Uno standard accettabile è inviare un segnale con un periodo di 20 ms al servo. Il duty cycle viene interpretato come la posizione in cui il servo deve essere spostato. Il duty cycle del segnale generato dovrebbe essere compreso tra il 5 e il 10%. Questi valori saranno convertiti in due posizioni estreme sul servo (massimo a sinistra e massimo a destra).

Cavi di collegamento del servo:

- **Red** – alimentazione;
- **Yellow** o **orange** – segnale di controllo;
- **Black** o **brown** – GND.

A seconda del produttore, i colori dei fili possono variare. Tuttavia, due saranno sicuramente nero/marrone e rosso. L'altro sarà il cavo del segnale.

Alimentazione del Servo

Come accennato in precedenza, non è consigliato alimentare il servo direttamente dalla stessa fonte che alimenta il microcontrollore. Pertanto, a causa del fatto che il motore consuma una grande corrente, è necessario utilizzare una fonte adeguata per alimentare il sistema.

Purtroppo l'alimentazione dalla porta USB, come avevamo fatto fino ad ora, non è sufficiente. Pertanto, per la prima volta, alimenteremo la scheda con una batteria da 9V!

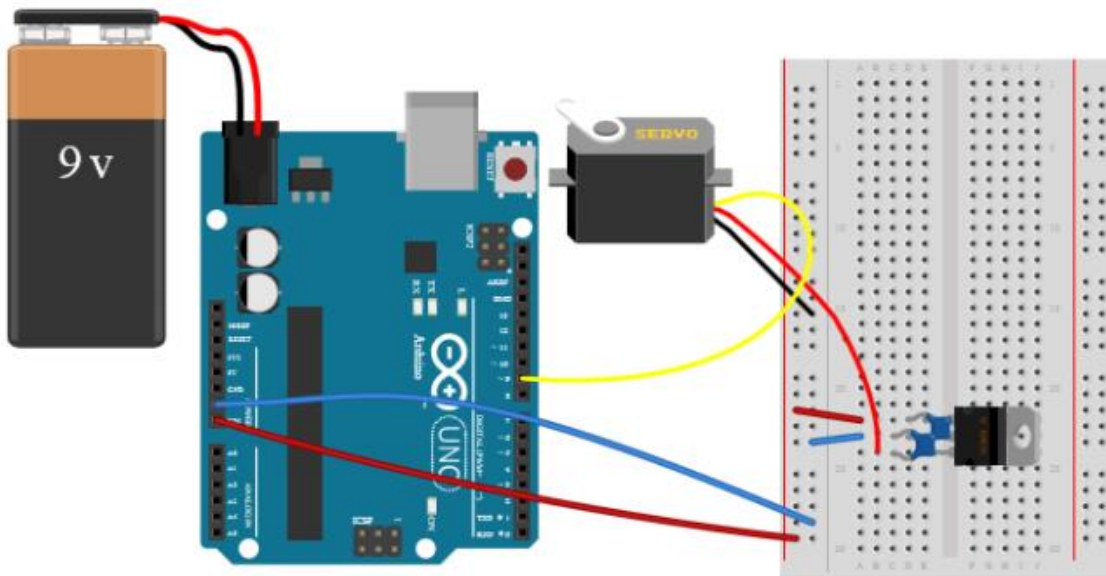
Esercizio pratico - Servomeccanismo

Passiamo al primo programma con il servomotore. Cabliamo il sistema secondo lo schema di montaggio sottostante.

Materiale richiesto:

- 1x scheda Arduino UNO e cavo USB;
- 1x Breadboard;
- 1x ServoMotore SG90;
- 1x Regolatore di tensione LM7805;
- 1x batteria da 9V;
- 1x Connettore per batteria da 9V;
- 7x cavetti.

Innanzitutto, è necessario collegare la batteria. In secondo luogo, è necessario includere un regolatore di tensione LM7805.



Ora il codice che farà muovere gradualmente il servo:

```
1 | #include <Servo.h> //Biblioteca responsável pelo servo motor
2 |
3 | Servo servomecanismo; //Declaração do servo como servomecanismo
4 | int posicao = 0; //Posição atual do servo de 0-180 graus
5 | int mudanca = 6; //Qual deve ser a posição do servo?
6 |
7 | void setup()
8 | {
9 |   servomecanismo.attach(9); //Servo ligado ao pino 9
10 | }
11 |
12 | void loop()
13 | {
14 |   if (posicao < 180) { //Se a posição for inferior a 180 graus
15 |     servomecanismo.write(posicao); //Move-se
16 |   } else { //Caso contrário, volta ao início
17 |     posicao = 0;
18 |   }
19 |
20 |   posicao = posicao + mudanca; //Aumentar a posição atual do servo
21 |   delay(200); //Atraso para melhor efeito
22 | }
```

Questa volta abbiamo aggiunto una libreria che amplierà le capacità del programma con le sue funzioni. Il comando utilizzato è:

```
1 | #include Servo.h
```

In questo caso, aggiungiamo il file Servo.h, che contiene ulteriori istruzioni pre utilizzare il servomotore. Grazie a questo, non abbiamo bisogno di controllare noi stessi il segnale PWM. È sufficiente indicare le posizioni (angolo) verso le quali vogliamo che il servo ruoti.

Per controllare il servo, dobbiamo dichiararlo:

```
1 | Servo servomecanismo;
```

La funzione **attach (pin)** - per l'oggetto Servo - funziona in modo simile a pinMode. Grazie a questa istruzione verrà generato un segnale PWM all'uscita indicata (in questo caso 9).

Dopo aver avviato il programma, il servo dovrebbe muoversi agevolmente da una posizione estrema all'altra in un ciclo. L'istruzione chiave è:

```
1 | servomecanismo.write(posicao);
```

La posizione (posicao) dovrebbe essere un angolo compreso tra 0 e 180°.

Introduzione ai display

Accendere e controllare un display grafico o di testo può sembrare molto difficile. Ci sono molti pixel sullo schermo che devono essere gestiti.

Ogni lettera è composta da diversi pixel, disposti in rettangoli. Il display che useremo è un LCD 16×2. Ciò significa che ha 2 righe, ciascuna con 16 caratteri. Questa è una limitazione di questi piccoli display. Ce ne sono altri con una capacità di testo molto maggiore, dove puoi visualizzare immagini, ecc.

I display LCD sono disponibili in vari colori, ad esempio verde, blu e nero. Puoi usare quello che ti è più comodo.

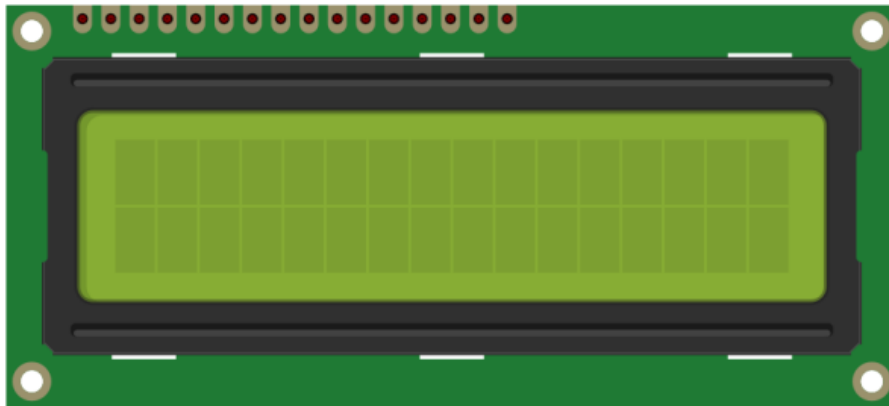
Come funziona un display?

Come accennato in precedenza, le lettere del display sono costituite da pixel. Controllarli individualmente significherebbe creare codice con una moltitudine di righe. Ma, ovviamente, questo non è necessario, il display funziona molto più facilmente grazie ai driver integrati. Il più noto è l'HD44780.

Spesso, nella descrizione di un determinato display, troverai l'indicazione che il display è compatibile con il driver HD44780. In questo caso, sarà facile utilizzare il display!

Come puoi inviare un messaggio al display? È necessario collegare circa 12 fili. Naturalmente, solo alcuni di essi vengono utilizzati per la comunicazione, il resto è necessario per l'alimentazione e altri segnali.

Il più delle volte, il nostro display è dotato di un connettore a 16 pin:



fritzing

Diamo un'occhiata alla legenda dei pin, da sinistra a destra:

1. VSS – terra (GND);
2. VDD – +5V alimentazione;
3. V0 - regolazione del contrasto;
4. RS - selezione del registro;
5. R/W – selezione di write o read;
6. E – segnale di abilitazione;
7. DB0 - Data;
8. DB1 - Data;
9. DB2 - Data;
10. DB3 - Data;
11. DB4 - Data;
12. DB5 - Data;
13. DB6 - Data;
14. DB7 - Data;
15. LED+ - aumento della retroilluminazione;
16. LED- - riduzione della retroilluminazione

I pin da 1 a 3 sono utilizzati per alimentare il display, i pin da 4 a 14 per controllare il display e gli ultimi due (15 e 16) per regolare la retroilluminazione.

I display compatibili con il driver HD44780 possono comunicare tramite 4 e 8 bit. Nella prima modalità sono necessarie 7 connessioni ad Arduino. Nella modalità a 8 bit, deve eseguire fino a 11 connessioni.

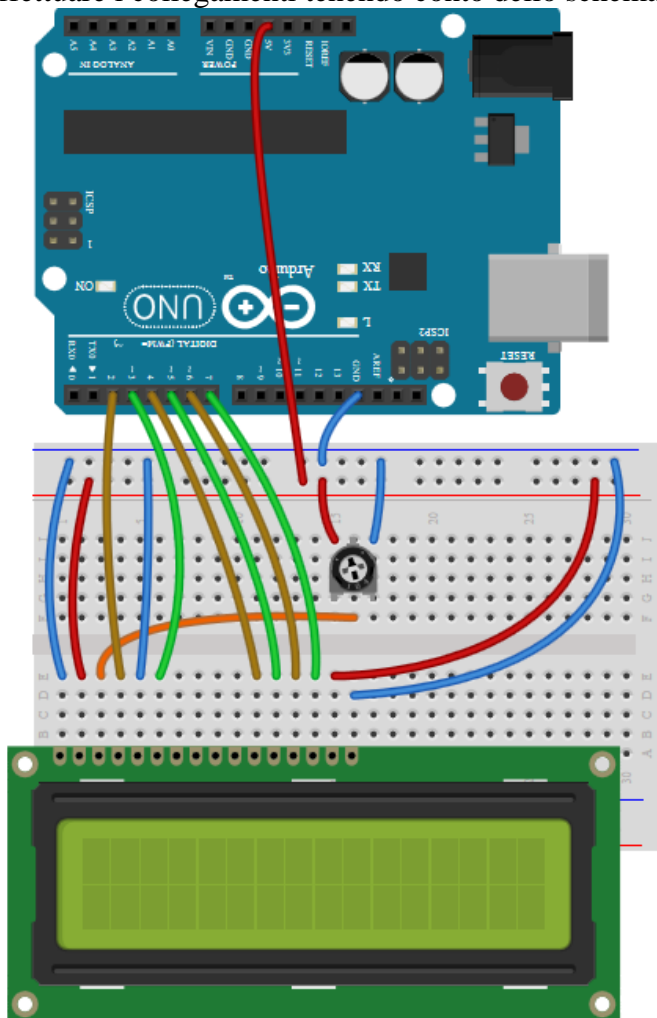
Collegamento del display ad Arduino

Arduino ha una libreria speciale per la visualizzazione di testi. Tuttavia, prima di passare alla programmazione, è necessario effettuare i collegamenti del sistema.

Materiale necessario:

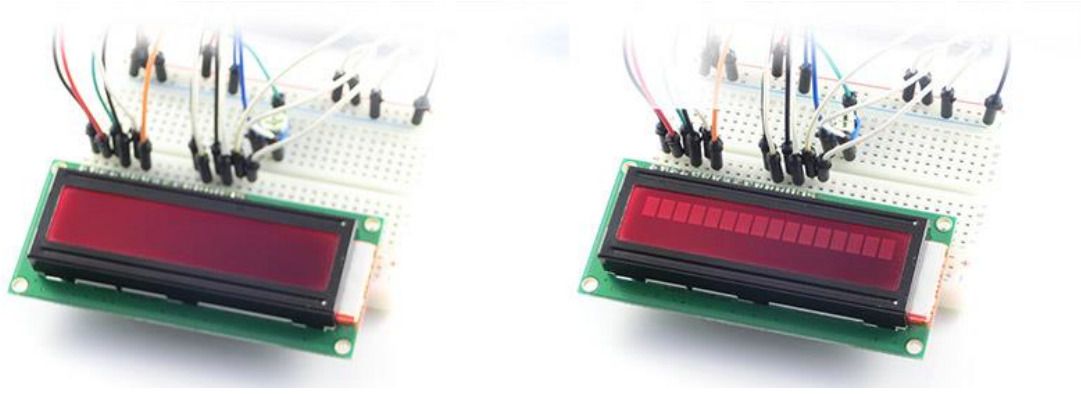
- 1x scheda Arduino UNO e cavo USB;
- 1x Breadboard;
- 1x 16×2 LCD display;
- 1x Potenziometro;
- 16x cavetti

Effettuare i collegamenti tenendo conto dello schema sottostante:



Ma a cosa serve il potenziometro in questo caso? È responsabile della regolazione del contrasto del display. Per quanto riguarda la retroilluminazione, basta collegarla all'alimentatore. Tuttavia, se si desidera regolare la luminosità, è possibile farlo collegando il pin LED+ a un'uscita PWM e programmarlo come si desidera. Ne parleremo dopo.

Dopo aver collegato Arduino all'alimentatore (tramite il cavo USB) il display dovrebbe accendersi, come mostrato di seguito::



Quando giri il potenziometro del contrasto, può succedere che lo schermo o la riga superiore si riempiano di rettangoli. Questo significa che il display è danneggiato? No, è un buon segnale. Un efficiente display collegato alla rete elettrica dovrebbe avere questo aspetto. Regolare quindi il potenziometro in modo tale che i rettangoli siano visibili.

Esercizio pratico - visualizzazione del testo sul display

Analogamente al caso dei servomotori, esiste anche una libreria adatta al funzionamento dei display. Questa volta si chiama **LiquidCrystal**. Inizieremo con un esempio e poi spiegheremo il programma. Non scaricare ancora il codice per il tuo Arduino!

Materiale richiesto:

- 1x scheda Arduino UNO e cavo USB;
- 1x Breadboard;
- 1x 16x2 LCD display;
- 1x Potenziometro;
- 16x cavetti.

```
1 | #include <LiquidCrystal.h> //Biblioteca responsável pelo display
2 | LiquidCrystal lcd(2, 3, 4, 5, 6, 7); //Informações sobre ligação do display
3 |
4 | void setup() {
5 |   lcd.begin(16, 2); //Declaração do tipo de LCD
6 |   lcd.setCursor(0, 0); //Configuração do cursor
7 |   lcd.print("Curso Arduino"); //Exibir texto
8 |   lcd.setCursor(0, 1); //Configuração do cursor
9 |   lcd.print("ROMUAS"); //Exibir texto
10 | }
11 |
12 | void loop() {
13 | }
```

La libreria responsabile della visualizzazione è nel file: **LiquidCrystal.h**. Per iniziare a lavorare sul display, devi dichiararlo. La riga utilizzata per questo è la seguente:

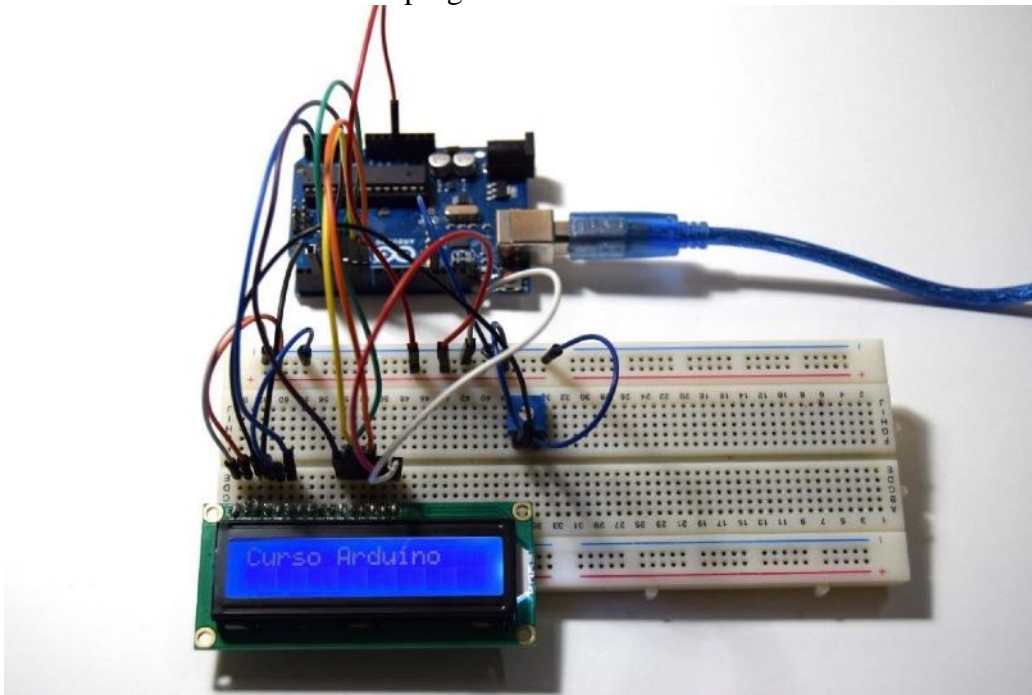
```
1 | LiquidCrystal lcd(2, 3, 4, 5, 6, 7);
```

Tali informazioni indicano che il display LCD è collegato ai pin da 2 a 7 di Arduino. Ma, ovviamente, la selezione dei pin è libera.

La funzione **lcd.begin** (caratteri, righe) definisce il numero di caratteri e righe in cui verrà visualizzato il testo. In questo caso il display permette la visualizzazione di 16 caratteri su ognuna delle due righe.

La funzione **lcd.setCursor** (posizione, linea) imposta il cursore nella posizione data. Ad esempio, (0,0) indica l'inizio, il primo carattere della prima riga, mentre (0,1) indica l'inizio della seconda riga.

In questo caso, la funzione più importante è **lcd.print** ("testo"), perché stampa sullo schermo il testo tra le parentesi. Quando si visualizzano lettere consecutive, il cursore si sposta. Ecco perché la prossima chiamata di `lcd.print` inizia nel punto in cui hai terminato il testo precedente. Ora è il momento di caricare il programma Arduino. Il risultato dovrebbe essere questo:



E se non funziona?

Se non riesci a vedere il testo sul display, controlla i seguenti punti, risolvono il 99,99% dei problemi!

- 1) Alimentazione del display;
- 2) Collegamento dei pin su Arduino;
- 3) Contrasto.

L'ultimo punto è il più facile da controllare e i principianti spesso se ne dimenticano! Si spera che dopo aver controllato i punti precedenti tutto funzioni perfettamente. Ora possiamo discutere le funzioni più importanti relative al display LCD.

Rimuovi contenuto dalla visualizzazione

Durante il test della visualizzazione dei testi sul display, potresti scoprire che a volte ci sono sovrapposizioni. Questo perché il display non cancella automaticamente il suo contenuto (ha i suoi vantaggi). Tuttavia, se si desidera eliminare i contenuti precedenti, utilizzare la funzione `lcd.clear` ().

Esercizio pratico - Regolazione della retroilluminazione

Come accennato in precedenza, la retroilluminazione, essendo gestita da LED, può essere attenuata. Vediamo come si fa!

Materiali richiesti:

- 1x scheda Arduino UNO e cavo USB;
- 1x Breadboard;
- 1x 16x2 LCD display;
- 1x Potenziometro;
- 17x Potenziometro.


```
1 | #include <LiquidCrystal.h> //Biblioteca responsável pelo display
2 | LiquidCrystal lcd(2, 3, 4, 5, 6, 7); //Informações sobre ligação do display
3 |
4 | void setup() {
5 |   lcd.begin(16, 2); //Declaração do tipo de LCD
6 |   lcd.clear();
7 |   lcd.setCursor(0,0);
8 |   lcd.print("Bem-Vindo ao");
9 |   lcd.setCursor(0,1);
10 |  lcd.print("Blog ElectroFun");
11 | }
12 | int brilho = 0;
13 | int mudanca = 5;
14 |
15 | void loop()
16 | {
17 |   analogWrite(11, brilho); //Gerar sinal PWM com determinado duty cycle
18 |   brilho = brilho + mudanca; //Mudar o brilho para valor alterado da variável
19 |   if (brilho == 0 || brilho == 255) { //Se o duty cycle for 0% ou 100%
20 |     mudanca = 0-mudanca; //Alterar brilho na direção oposta
21 |   }
22 |
23 |   delay(30); //Atraso para melhor efeito
24 | }
```

Controllo del motore in CC

Perché aggiungere motori ad Arduino?

Il controllo del senso di rotazione e della velocità del motore apre le porte a molte possibilità. Con queste nuove funzionalità, possiamo costruire un semplice robot mobile che gira per casa ed evita gli ostacoli. Possiamo anche creare un veicolo controllato da remoto tramite il nostro cellulare. Le possibilità sono davvero infinite!

Quali motori useremo?

Diamo un'occhiata al principio del controllo dei popolari motori CC. Questi sono spesso usati nei progetti di robotica.



Esempio di un motore a corrente continua.



È ancora necessario vedere cosa sono i motori DC. In generale, i motori a corrente continua sono dispositivi che convertono l'energia elettrica in energia meccanica. In questo caso specifico parleremo di motori che consumano, in media, meno di 1A se alimentati a 5-9V. Queste restrizioni sono condizionate dal controller del motore utilizzato, ma ne parleremo più avanti.

Il principio del controllo del motore è universale: più potente è il driver, più grande può essere il motore controllato.

Perché non possiamo collegare il motore direttamente all'Arduino?

Arduino, più precisamente il microcontrollore integrato, controlla i segnali. La corrente di ciascuna uscita è relativamente piccola (circa 20 mA). È facile presumere che il 99,999% dei motori che incontri avrà bisogno di molta più corrente. Detto questo, collegando un motore ad Arduino si corre il rischio di danneggiare irreversibilmente la scheda.

Perché le esercitazioni pratiche non includono i motori?

Come in ogni cosa, per iniziare a controllare i motori, dovrai esercitarti (attraverso gli esercizi pratici proposti). Come puoi vedere, negli esercizi proposti, c'è un Arduino, una batteria, un controller per il motore (L293D), condensatori, ma manca qualcosa... niente motori DC.

Ma perché? La verità è che includere un set di motori sarebbe uno spreco di denaro. Dato che sei appena agli inizi in questo mondo, naturalmente non sai quali progetti svilupperai in futuro. Senza saperlo, non sarai in grado di scegliere i motori giusti. Perché investire in qualcosa che probabilmente non userai mai?

Pertanto, nei seguenti esercizi, i motori saranno sostituiti da LED. Potrai comunque osservare il cambio di direzione di rotazione (attraverso il led che si accende) e il cambio di velocità (attraverso la luminosità del led).

Per questo articolo avrai bisogno dei seguenti materiali:

- 1x scheda Arduino UNO e cavo USB;
- 1x Breadboard;
- 1x batteria da 9V battery;
- 1x connettore per la batteria da 9V;
- 1x controller L293D;
- 1x resistenza da 1KΩ;
- 1x condensatore ceramic 10-220uF;
- 1x condensatore elettrolitico 10-220uF;
- 2x LEDs;
- cavetti.

Introduzione agli H-Bridges

Sono necessari componenti intermedi tra Arduino e i motori. Questi sono spesso chiamati H-Bridges. Questi driver possono essere costruiti da diversi transistor oppure puoi semplicemente utilizzare un circuito integrato H-bridge. Dato che sei un principiante, è meglio ricorrere a un chip già pronto.

Lo scopo principale dei ponti H è leggere e convertire i segnali inviati dal microcontrollore per renderli compatibili con il controller del motore. Ad esempio, Arduino, i cui segnali possono funzionare a un massimo di 5V/20mA, con un H-Bridge, può facilmente controllare un motore che richiede 12V/1A per funzionare.

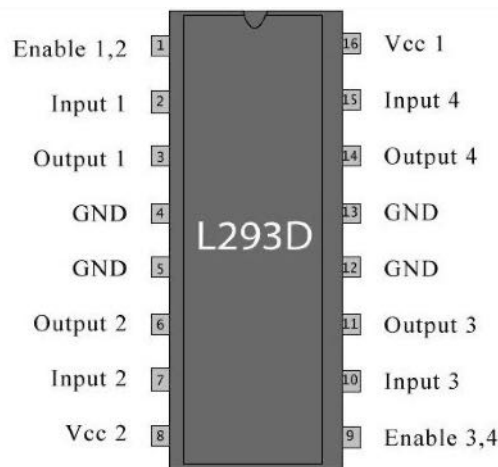
Il tuo primo H-bridge - L293D

Useremo il chip L293D, che nonostante sia vecchio, è economico e molto comune. Il suo principale vantaggio è che ha un assemblaggio THT, il che significa che possiamo collegarlo direttamente alla breadboard.





Questo sistema ha 16 pin. Vediamo la leggenda di ognuno di loro:



Una delle informazioni chiave sull'H-bridge che dovrebbe essere controllata è la sua performance. In particolare, la corrente che il motore sarà in grado di assorbire. L'L293D ha una corrente media per canale di 0,6 A-1,2 A. Cosa significa questo? Significa che, idealmente, i motori possono assorbire fino a 0,6 A, ma se per un attimo la corrente sale a 1,2 A, non succederà niente di male. Purché sia per un breve periodo di tempo!

Vediamo ora ciascuno dei pin. Innanzitutto, i **pin dell'alimentatore**:

- **Terra (GND)** – 4, 5, 12, 13;
- **5V – alimentazione “logica”** (Arduino) - 16;
- **Alimentazione per i motori fino a 36V** - 8.

Ricorda che ogni ponte ha una caduta di tensione. Ciò significa che, ad esempio, se fornisci un'alimentazione a 9V, l'L293D ne consuma una parte e i motori ricevono al massimo 7V. Questo sistema è vecchio, quindi ha una grande caduta di tensione. Altri bridge più recenti (ad esempio TB6612) hanno una caduta di tensione inferiore.

Pin di controllo-motore:

- **Ingressi che definiscono il senso di rotazione del 1° motore** – 2, 7;
- **Ingressi che definiscono il senso di rotazione del 2° motore** – 10, 15;
- **Ingresso che definisce la velocità del 1° motore** – 1;
- **Ingresso che definisce la velocità del 2° motore** – 9.

Per arrestare o cambiare il senso di rotazione del motore è necessario impostare i segnali secondo il seguente diagramma, detto **tavola di verità**:

ENABLE	Input 1	Input 2	Motor
PWM	1	0	Esquerda
PWM	0	1	Direita
PWM	0	0	Parar
PWM	1	1	Parar

Pin Motore:

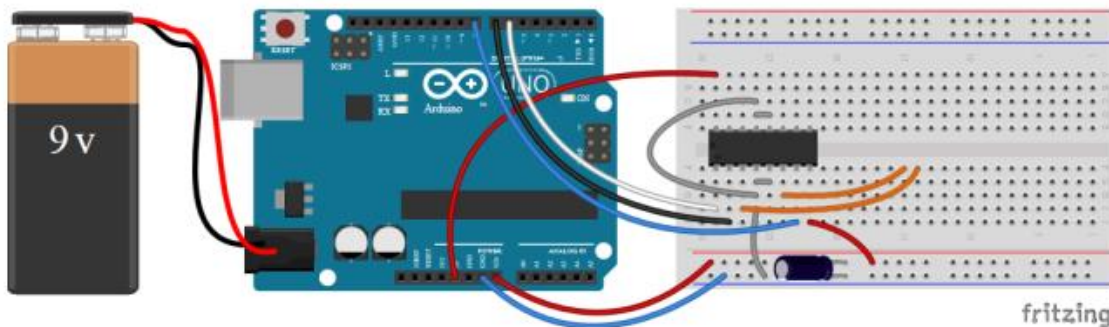
- Uscite relative al primo motore – 3, 6;
- Uscite relative al secondo motore – 11, 14.

Etazione pratica - H Bridge

Parliamo dell'alimentatore. Possiamo alimentare il sistema con una o due fonti di tensione. Se scegliamo la prima opzione, Arduino e i motori saranno alimentati dalla stessa fonte - sconsigliamo questa configurazione, in quanto può portare a frequenti interruzioni e ripristini.

Pertanto, è più sicuro scegliere la seconda opzione, in cui Arduino sarà alimentato da un alimentatore separato. Nel nostro caso utilizzeremo l'alimentazione USB e una batteria da 9V, dalla quale alimenteremo direttamente i motori (anche l'Arduino può essere alimentato in questo modo, ma tramite lo stabilizzatore integrato).

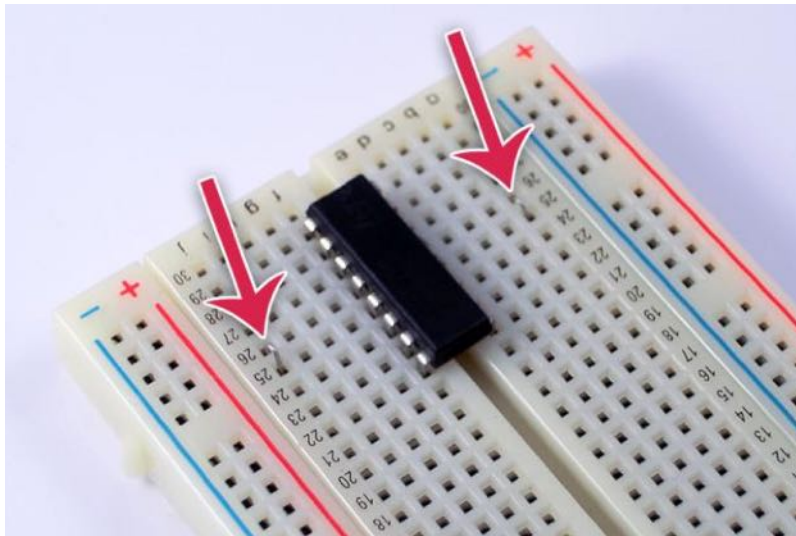
Il sistema deve essere collegato in modo simile allo schema mostrato di seguito. Ricorda che collegheremo 9V alla scheda, non 5V!



I cavetti arancioni saranno collegati al "motore".

Ricordarsi di collegare i condensatori, in quanto sono quelli che filtrano la tensione fornita dalla batteria da 9V.

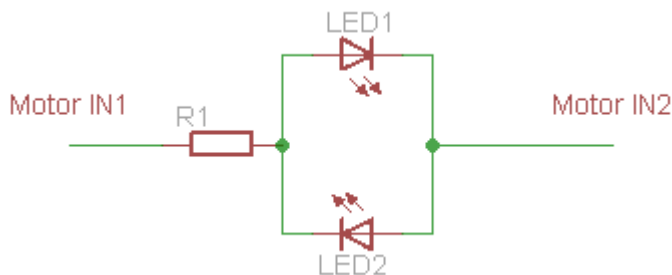
Il sistema è relativamente complicato, quindi per rendere il layout più organizzato e per risparmiare spazio, colleghiamo i due pin GND su entrambi i lati del chip. Questo può essere fatto come segue, utilizzando ad esempio i terminali in eccesso tagliati dai resistori:



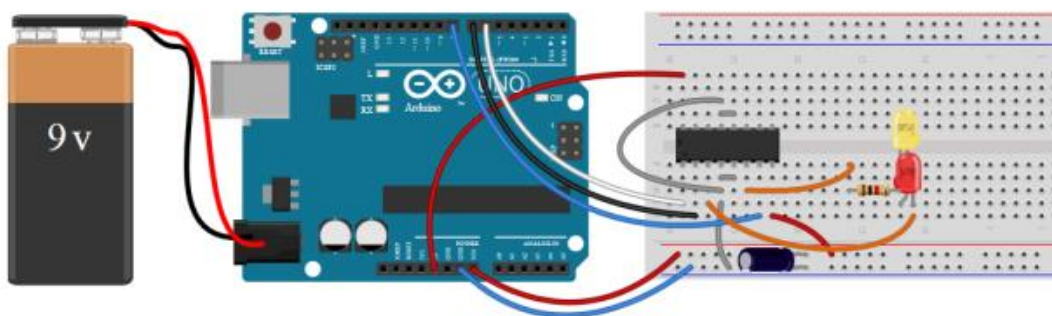
È ora di avviare il motore. Come accennato in precedenza, simuleremo un motore con LED, quindi non è necessario acquistarne uno. Ma come lo facciamo? Molto semplice!

Simulazione del motore

Come sai, un LED è un elemento semiconduttore, che si illumina quando la corrente lo attraversa nella giusta direzione. Pertanto, se colleghiamo due LED in parallelo in modo che siano collegati in direzioni opposte, possiamo verificare in quale direzione scorre la corrente, quando uno dei due LED si accende. A sua volta, la velocità del motore può essere controllata dalla luminosità del LED. Detto questo, al posto del motore, includiamo il seguente circuito:



Pertanto, lo schema elettrico completo è il seguente:



Programmazione - Controllo del senso di rotazione

Passiamo ora alla programmazione, in base allo schema precedente. All'inizio esamineremo il controllo della rotazione del motore. Lasciemo la regolazione della velocità per dopo. Come puoi vedere, i pin Arduino responsabili del motore sono i seguenti:

- 6 (PWM) – regolazione della velocità;
- 7, 8 – controllo senso di rotazione.

Se non vogliamo controllare la velocità del motore, dobbiamo impostare il pin 6 su alto. Potremmo anche collegare il cavo direttamente a 5V. Tuttavia, poiché abbiamo già effettuato le connessioni, utilizzeremo i pin di Arduino:



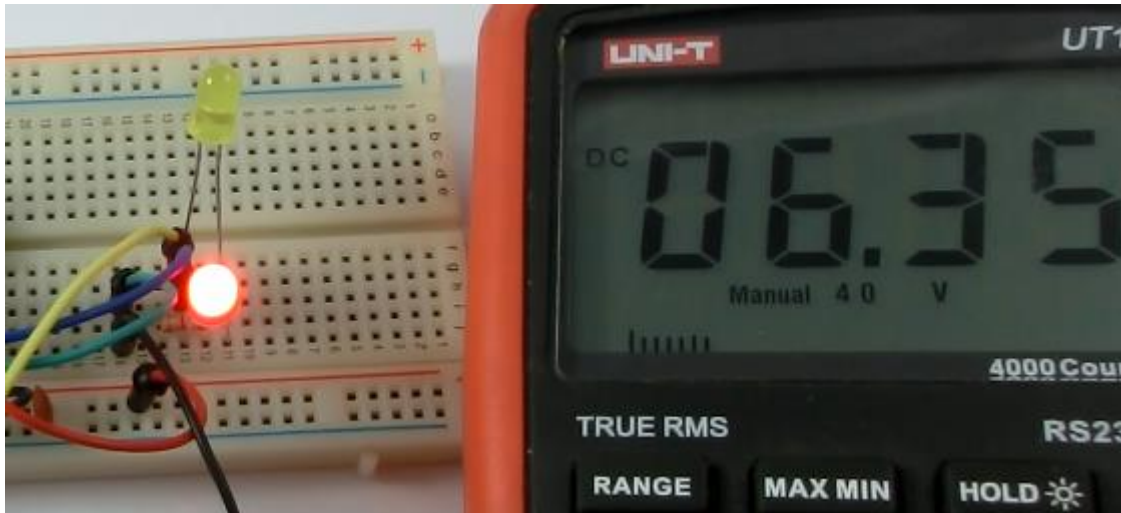
```
1 void setup() {
2   pinMode(6, OUTPUT); //Sinal PWM do motor para controlo da velocidade
3   digitalWrite(6, HIGH); //Definir permanentemente o estado high no pino 6 -
4   velocidade máxima
5
6   pinMode(7, OUTPUT); //Pino que controla a direção da rotação do motor
7   pinMode(8, OUTPUT);
8 }
9
10 void loop() {
11 //Restante programa
}
```

Se il sistema è stato connesso correttamente, dopo aver caricato il programma non dovrebbe succedere nulla! Ora è il momento di aggiungere il codice rimanente alla funzione loop. Supponiamo di voler ruotare il motore 3 secondi in una direzione e 3 secondi nella direzione opposta (velocità massima). A tale scopo, dovremmo aggiungere una parte molto semplice al programma:

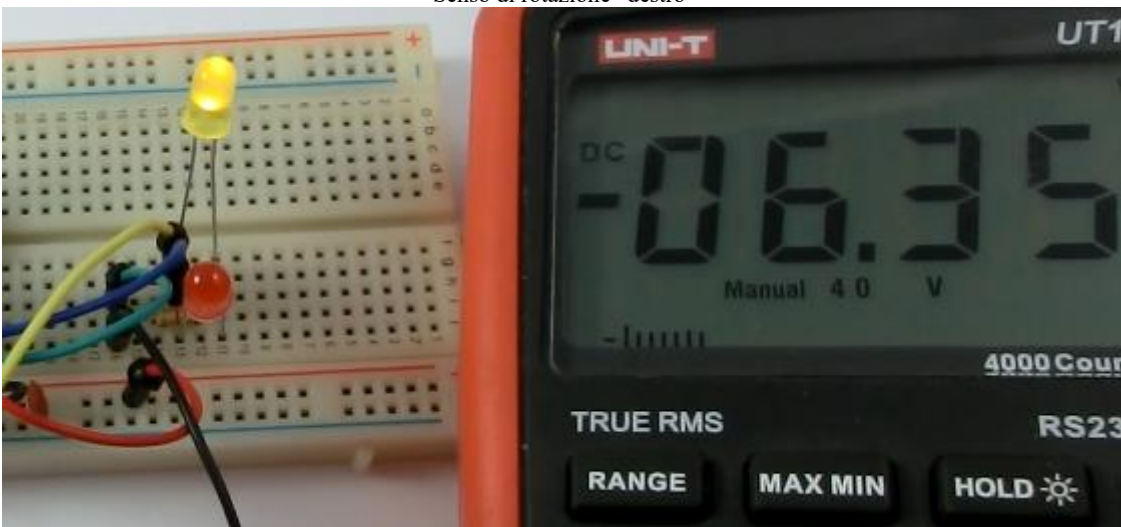
```
1 void setup() {
2   pinMode(6, OUTPUT); //Sinal PWM do motor para controlo da velocidade
3   digitalWrite(6, HIGH); //Definir permanentemente o estado high no pino 6 -
4   velocidade máxima
5
6   pinMode(7, OUTPUT); //Pino que controla a direção da rotação do motor
7   pinMode(8, OUTPUT);
8 }
9
10 void loop() {
11 digitalWrite(7, LOW); //Rodar para a esquerda
12 digitalWrite(8, HIGH);
13 delay(3000); //Durante 3 segundos
14
15 digitalWrite(7, HIGH); //Rodar para a direita
16 digitalWrite(8, LOW);
17 delay(3000); //Durante 3 segundos
}
```

Secondo la **tabella di verità** presentata sopra, sappiamo che il sistema funziona se uno degli ingressi è nello stato "0" basso e l'altro è nello stato "1" alto. Il senso di rotazione dipende dal segnale di ingresso.

Dopo aver caricato il programma su Arduino, i due LED dovrebbero iniziare a lampeggiare alternativamente. Concentrati e capisci perché sta accadendo. I LED si accendono alternativamente perché la corrente scorre dall'uscita 1 all'uscita 2 e viceversa. Questa accensione alternata dei due led è possibile perché questi sono stati inseriti al contrario. Per facilitare la comprensione, guardate le seguenti immagini, dove, oltre ai led, è presente anche un voltmetro:



Senso di rotazione “destro”



Senso di rotazione “sinistro”

Una volta il risultato è positivo e un'altra volta il risultato è negativo. In questo caso, il meno mostra che la corrente scorre nella direzione opposta alla direzione iniziale.

Poiché siamo già riusciti a controllare il senso di rotazione del motore, è ora di affrontare la velocità.

Accelerazione “dolce” del motore

Poiché conosciamo già il ciclo for, possiamo usarlo per accelerare dolcemente il motore. È sufficiente cambiare il segnale PWM in modo regolare (“dolcemente”), ad esempio ogni 25 ms.

Il codice è semplice e la variabile *i* è responsabile del duty cycle del segnale PWM:

```

1  void setup() {
2  pinMode(6, OUTPUT); //Sinal PWM do motor para controlo da velocidade
3
4  pinMode(7, OUTPUT); //Pino que controla a direção da rotação do motor
5  pinMode(8, OUTPUT);
6
7  digitalWrite(7, LOW); //Rodar para a esquerda
8  digitalWrite(8, HIGH);
9  }
10
11 void loop() {
12 for (int i = 0; i <= 255; i++) {
13 analogWrite(6, i); //Aceleração suave do motor
14 delay(25);
15 }
16 }
  
```



Molto spesso i principianti hanno problemi perché il motore non vuole muoversi. Il più delle volte, si scopre che l'alimentatore utilizzato è troppo debole o che il segnale PWM ha un ciclo di lavoro troppo basso. Se mai dovessi avere un problema, ricordati di controllare questi due aspetti!

STAMPA 3D

Cos'è la stampa 3D?

La stampa 3D è il nome dato a una serie di tecniche che riproducono **oggetti in tre dimensioni**. Chiamato anche prototipazione rapida, il processo consente la stampa di modelli attraverso la produzione additiva.

In altre parole, le stampanti formano modelli tridimensionali utilizzando una tecnica che **sovrappone strati sottili** di un certo materiale fino a quando l'oggetto non è pronto.

Con questo processo è possibile stampare parti di macchine, oggetti decorativi, gioielli, organi e persino alimenti.

In genere gli articoli sono realizzati utilizzando come materia prima alcuni tipi di **plastica**, ma è possibile utilizzare anche altri materiali, tra cui il metallo.

La stampa 3D ha rivoluzionato settori come la medicina e l'ingegneria, oltre ad essere molto utile per chi vuole iniziare ad... inventare qualcosa.

Questo perché il processo ha accorciato il percorso per raggiungere una versione di prova di qualsiasi prodotto, chiamata prototipo.

In precedenza, era necessario immaginare, disegnare l'oggetto e sviluppare un progetto in modo che il prototipo potesse essere costruito.

Con una stampante 3D, basta progettare l'oggetto con un **software specifico** e convertire il file in un formato compatibile con la macchina per ottenere un prototipo.

Certo, l'uso del software richiede alcune conoscenze tecniche, ma le dinamiche sono diventate molto più semplici.

A seconda dell'articolo desiderato e del processo di stampa 3D, può essere necessario anche **rifinire** il pezzo.

Come è nata la stampa 3D?

Sebbene l'idea degli oggetti stampati in 3D sembri futuristica, non è così recente.

Nel **1984**, la prima tecnica utilizzata in questo processo, chiamata stereolitografia, fu brevettata su richiesta dell'ingegnere **Charles Hull**.

Hull ha creato una macchina in grado di stampare, grazie all'azione di un laser, lampade speciali, utilizzate nella solidificazione delle resine, così come parti in plastica,.

L'iniziativa ha avuto successo, portando Hull a fondare **3D Systems Corp** e a vendere i suoi servizi innovativi due anni dopo.

Sempre focalizzata su tecnologie innovative, l'azienda è cresciuta e ha aggiunto altri tipi di stampa 3D nel suo portafoglio, come la sinterizzazione laser selettiva (SLS) e i **sistemi a base di polvere** per il 3D.

Attualmente l'azienda è diventata un punto di riferimento nella produzione di particolari tridimensionali personalizzati, in particolare stampi utilizzati negli ambulatori e in altre attività del **settore sanitario**.

In riconoscimento del suo contributo alla società, l'ingegnere Hull è stato insignito dell'European Inventor Award nel 2014.

Oltre a 3D Systems Corp. vale la pena citare la collaborazione di aziende come Stratasys per il campo della prototipazione rapida.

Dagli anni '90 sono state perfezionate metodologie e sviluppato macchine in grado di stampare diverse tipologie di manufatti.

Inizialmente le stampanti commercializzate erano più grosse, costose e destinate alla produzione industriale.



A cosa serve la stampa 3D?

La stampa 3D serve a **costruire vari oggetti personalizzati**, in modo **agile** e relativamente semplice.

Dopo essere stati prototipati, i pezzi diventano reali grazie a questo processo, inclusi dettagli come crepe, forme e colori.

Per il suo **carattere versatile**, la prototipazione rapida serve agli scopi più svariati.

La più comune è la stampa di un singolo oggetto, pensato per essere utilizzato da una sola persona o da un piccolo gruppo.

È il caso di articoli di cancelleria, come portapenne, e di oggetti decorativi, come cornici differenziate, lampade e portachiavi.

Ma il processo sta **guadagnando popolarità nell'industria**, dove aiuta a sostituire le parti, che vengono prodotte su richiesta e a un costo inferiore rispetto alla produzione con dinamiche tradizionali.

A seconda del materiale utilizzato, anche **mobili, tatuaggi e protesi** possono essere stampati e svolgere con successo le funzioni delle strutture che li hanno ispirati.

Come funziona la stampa 3D?

Proprio come una normale stampante, che rilascia getti d'inchiostro in quantità e forme specifiche per formare immagini e lettere, la stampante 3D **inietta materiali diversi** per creare un oggetto.

La differenza è che, invece che su carta, le moderne macchine stampano gli articoli su un vassoio.

Tutto inizia con **la progettazione del pezzo** con un software di modellazione 3D, come **AutoCad e Blender**.

Esistono vari tipi di software, generalmente orientati al settore del prodotto da stampare: edilizia, cancelleria, medicina e così via.

Poi, con il disegno in mano, la **materia prima** viene inserita nella stampante, che si accende e inizia ad espellere il primo strato sul vassoio.

In altre parole, invece di partire da un pezzo grande che verrà tagliato e modellato, la dinamica assembla gli articoli unendo piccole parti e strati di materiale.

Il processo inizia sempre **dal basso verso l'alto**, stampando uno strato alla volta e il completamento può richiedere da pochi minuti a giorni.

Sebbene seguano uno schema simile, esistono diversi tipi di stampa tridimensionale, come descritto di seguito.

Tipologie di stampanti 3D

Esistono **modelli più grandi e più piccoli**, finalizzati alla produzione di piccoli pezzi fino a muri e piccoli edifici.

Usano proprie tecniche di stampa, a seconda delle loro risorse e della materia prima con cui lavorano.

Ecco, di seguito, le metodologie più popolari.

Modellazione a deposizione fusa - Fusion Deposition Modeling - FDM

Chiamata anche fused filament manufacturing (fabbricazione a fusione di filamento - FFF), questa è una delle tecniche più comunemente utilizzate grazie al suo basso costo e alla relativa semplicità.

Questo processo utilizza **fili di plastica** per stampare gli oggetti, da qui il nome della tecnica.

I filamenti vengono riscaldati, mentre una testina che si muove lungo due assi diversi inietta il materiale in un vassoio.

La plastica viene quindi fusa per formare lo stampo richiesto.

Le stampanti che utilizzano la modellazione a deposizione fusa hanno motori che richiedono **meno potenza e raffreddamento** per essere azionati, rispetto ai laser utilizzati in altre modalità di stampa 3D



Pertanto, le macchine che stampano tramite FDM possono essere installate al di fuori degli ambienti industriali, essendo utilizzate principalmente per **scopi accademici** e stampando prototipi che verranno personalizzati.

Del resto la tecnica non consente molte variazioni nella forma dell'oggetto, in quanto ha una finitura più semplice, adatta alla composizione di oggetti meno sofisticati.

Stereolitografia

Nei paragrafi precedenti vi abbiamo detto che la modalità pionieristica è stata la stereolitografia, creata da Chuck Hull negli anni '80.

Questa tecnica impiega **resina liquida** (acrilica, epossidica o vinilica) per comporre ogni dettaglio del pezzo desiderato.

La resina viene indurita da un raggio laser ultravioletto, che viene emesso dalla stampante 3D per formare ogni strato del manufatto.

Infine, la resina liquida in eccesso deve essere rimossa e il pezzo portato in **forno** per ricevere la finitura necessaria.

Questo metodo consente la stampa di oggetti più resistenti e complessi rispetto alla tecnica FDM, però con tempi e costi maggiori.

Sinterizzazione laser selettiva - Selective Laser Sintering - SLS

Questo processo è simile alla stereolitografia, ma utilizza **materia prima in polvere** per formare gli strati del modello tridimensionale.

SLS rappresenta un grande progresso per la prototipazione rapida, in quanto consente l'utilizzo di diversi materiali nella stampa, generando parti in poliammidi, elastomeri, ceramiche e metalli.

Il processo prevede l'utilizzo di una **robusta stampante**, dotata di una camera di stampa, nella quale la polvere viene iniettata, livellata e colpita da un laser ad alta potenza, che la riscalda ed effettua la fusione.

Terminato il primo strato, il rullo della stampante vi passa sopra per ricoprirlo con altra polvere, che verrà **riscaldata e fusa** fino a quando l'oggetto non sarà pronto.

Successivamente, un getto di aria compressa e spazzole rimuovono la polvere in eccesso dall'oggetto.

Sinterizzazione laser diretta dei metalli - Direct Metal Laser Sintering

Come suggerisce il nome, la sinterizzazione laser diretta in metallo funziona **combinando particelle di metallo**.

Come nella SLS, viene utilizzato un laser per riscaldare e unire il materiale, dando vita a prodotti unici.

Uno dei vantaggi di questa tecnica è la produzione di oggetti di forma complessa, che normalmente non verrebbero fabbricati industrialmente.

Polyjet

Polyjet è la metodologia più simile alla stampa a getto d'inchiostro, sovrapponendo gli strati con perfezione e agilità.

Utilizzando un **tipo di plastica** (il fotopolimero) in forma liquida, questo processo si traduce in una riproduzione fedele del disegno.

Quali sono i principali materiali utilizzati nella stampa 3D?

Elenchiamo di seguito i principali materiali per la stampa di modelli tridimensionali:

- Carta
- Materie plastiche
- Metalli
- Gomma



- Vetro
- Resine
- Ceramica.

Le plastiche sintetiche sono la **materia prima più comune**, soprattutto per le stampanti che utilizzano filamenti per creare gli strati degli oggetti.

L'acido polilattico biodegradabile, l'acrilonitrile butadiene stirene e le poliammidi (come il nylon) sono esempi di materie plastiche comunemente utilizzate nella stampa 3D.

Cos'è Tinkercad?



Tinkercad è una raccolta di strumenti software, online e gratuiti, per chi vuole creare modelli 3D in modo molto semplice e intuitivo.

Usando questo strumento, hai accesso a varie forme geometriche per costruire il tuo progetto. Lo slogan di Tinkercad traduce molto bene la sua funzionalità: "dalla mente al progetto in pochi minuti".

Per utilizzare lo strumento è sufficiente creare un account gratuito su <https://www.tinkercad.com/>

Quali sono i principali strumenti di Tinkercad?

Sebbene Tinkercad sia perfetto per i principianti, ciò non significa che coloro che hanno più esperienza con la modellazione 3D non apprezzeranno questo software.

L'idea principale dello strumento è creare modelli più complessi dalla composizione di diverse forme semplici.

Inoltre, il software consente di aggiungere circuiti elettronici ai progetti 3D per creare oggetti con luce e movimento. Il risultato finale può anche essere simulato nel software per verificare come i componenti risponderanno nella vita reale.

Tinkercad fornisce forme come sfere, cilindri, scatole, coni, testi, numeri e connettori che possono comporre il tuo progetto.

Queste forme possono essere raggruppate, duplicate, separate, allineate o riflesse fino a ottenere un risultato soddisfacente per la tua modellazione.

E per stampare il modello costruito in Tinkercad, la buona notizia è che potrai esportare il tuo file in formato **STL** o **OBJ** per poter eseguire le configurazioni di pre stampa con il slicing software di tua scelta.

Devo scaricare TinkerCad?

Uno dei principali vantaggi di Tinkercad è che è totalmente online, ovvero non è necessario scaricarlo per usarlo. Tuttavia, per utilizzarlo è necessario registrarsi. Dopo aver creato l'account gratuitamente, inizia a modellare!

Come muovere i primi passi con Tinkercad?

Per iniziare, vai al sito web di Tinkercad <https://www.tinkercad.com/>. Clicca su "Iscriviti" in alto a destra (solo se non hai già un account, altrimenti accedi semplicemente cliccando su "Accedi").



Nella schermata di registrazione sceglierai il tuo paese e inserirai la tua data di nascita. Quindi fai clic su "Avanti". Inserisci il tuo indirizzo e-mail e la password scelta. Accetta i termini di servizio e la privacy e fai clic su "Crea account".

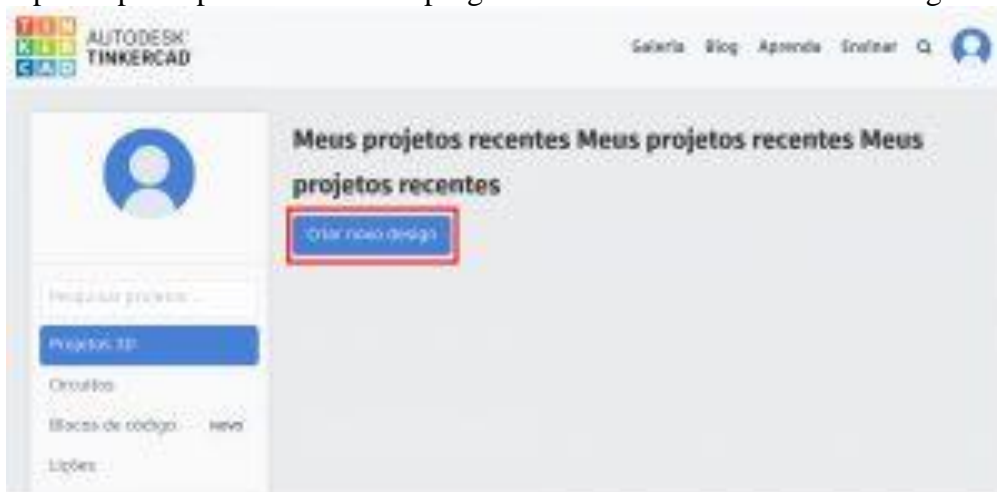


Con l'account creato l'applicazione aprirà automaticamente la schermata iniziale con il tutorial dei primi passi. Se vuoi saltare questo passaggio, fai clic sulla "X" a destra.



Quali sono le caratteristiche del software?

Tinkercad utilizza lo stesso principio di Lego. In esso lavorerai principalmente con forme predefinite e strutture geometriche. Puoi aggiungere o sottrarre pezzi per creare buchi o oggetti cavi. Il primo passo per iniziare il tuo progetto è cliccare su "Crea nuovo design".



Si aprirà la schermata di lavoro come nell'immagine sottostante:



1. Il rettangolo indicato con il numero 1 visualizza le funzioni "Copia" (Ctrl+C), "Incolla" (Ctrl+V), "Duplica" (Ctrl+D), "Cancella", "Annulla" (Ctrl+Z) e "Ripristina" (Ctrl+Y).

2. La feature indicata dal rettangolo rosso numero 2 viene utilizzata per spostare il piano di lavoro. Il piano di lavoro può essere spostato anche selezionandolo con il tasto destro del mouse. Per spostarti in direzioni diverse, tieni premuto il pulsante di selezione.
3. Il rettangolo numero 3 contiene le funzioni più importanti, che sono "Mostra tutto" (Ctrl+Shift+H), "Raggruppa" (Ctrl+G), "Separa" (Ctrl+Shift+G), "Allinea" (L) e "Capovolgi" (M);
4. Il rettangolo rosso indicato con il numero 4 si riferisce agli elenchi di forme, testi e altre risorse che l'applicazione ha da offrire.

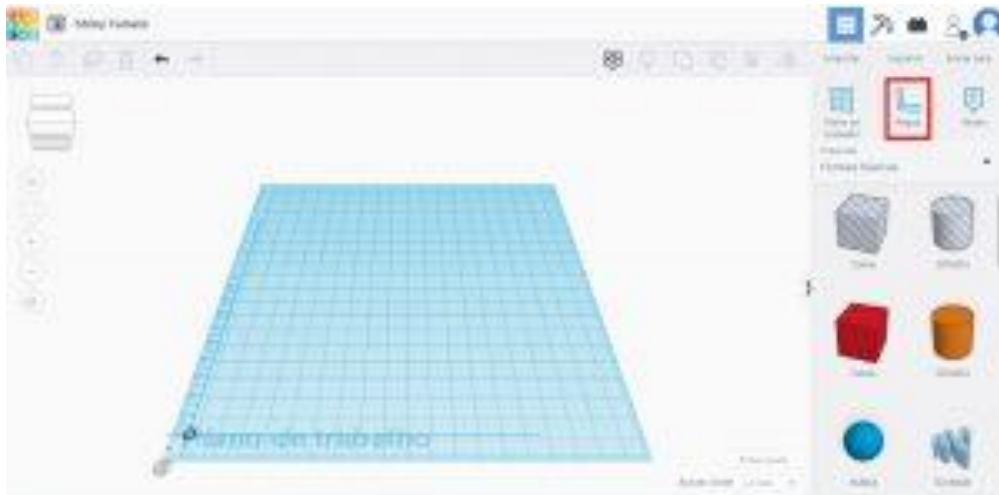
Come iniziare a modellare con Tinkercad?

È ora di iniziare i tuoi progetti in Tinkercad! A titolo di esempio, ti mostreremo un modello di etichetta per il tuo bagaglio. Ma puoi dare sfogo alla tua creatività, poiché l'applicazione ti consente di farlo.

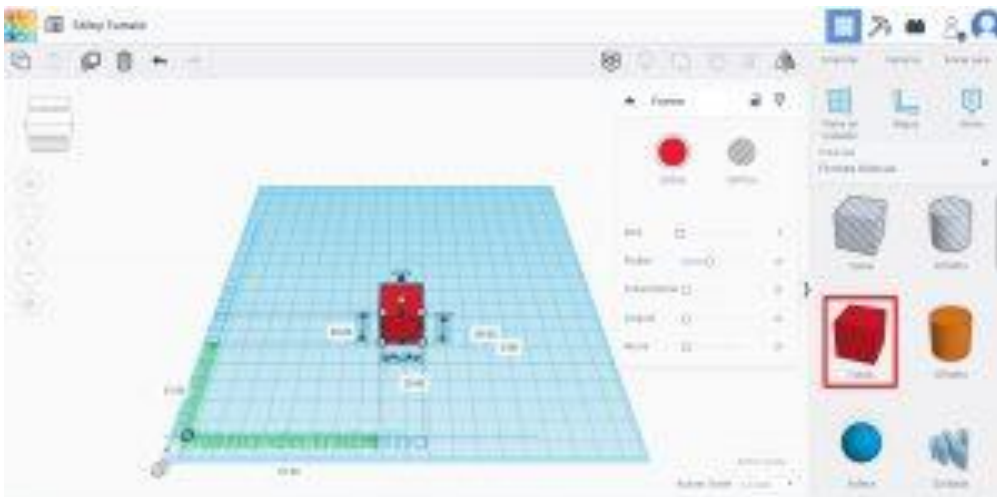


Step 1: utilizzando la forma "Box".

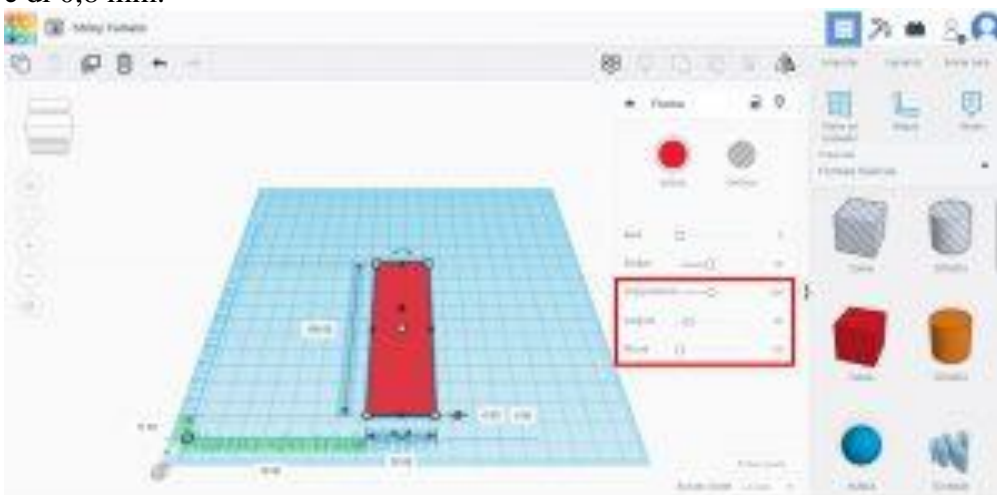
Quando inizi il tuo progetto, il primo (ed essenziale) passo è aggiungere lo strumento "Righello" situato nell'angolo in alto a destra al tuo piano di lavoro. In questo modo avrai sempre un riferimento delle misure.



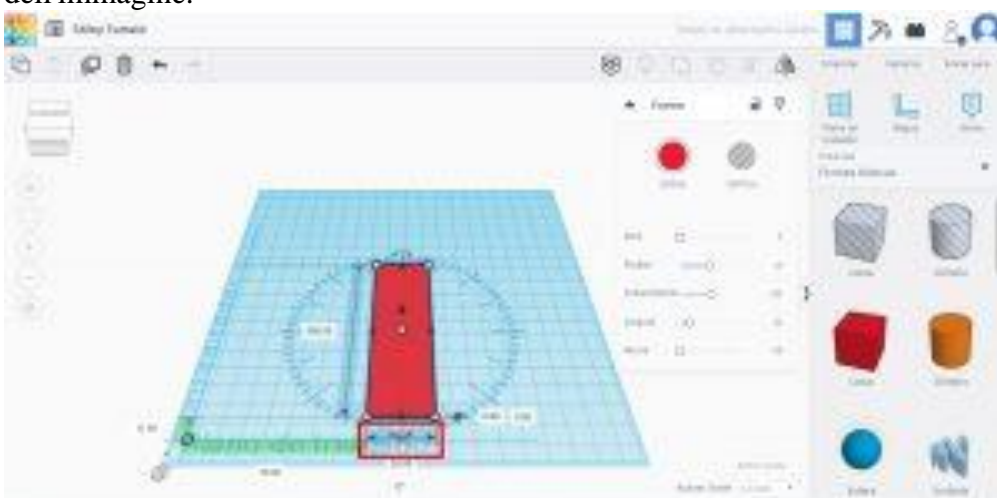
Successivamente, trascina la forma "Box" per iniziare la modellazione.



Fare clic sull'oggetto e modificarlo nel menu della forma. Oppure, se preferisci, modifica le dimensioni desiderate su uno qualsiasi dei quadrati bianchi che appariranno intorno alla figura. Nella nostra etichetta bagaglio abbiamo usato 100 mm di lunghezza e 30 mm di larghezza. L'altezza è di 0,8 mm.

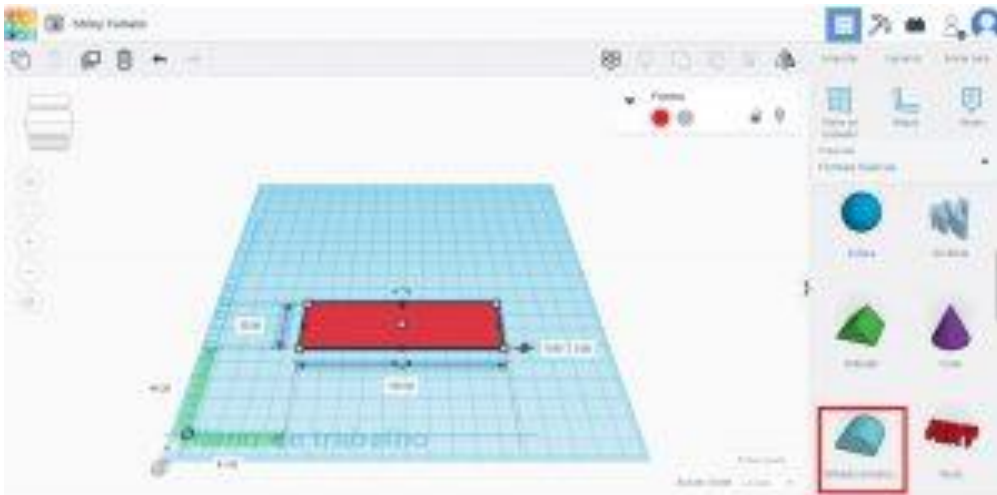


Per ruotare il pezzo sul piano di lavoro basta cliccare sull'immagine e spostare la rotazione dell'immagine.

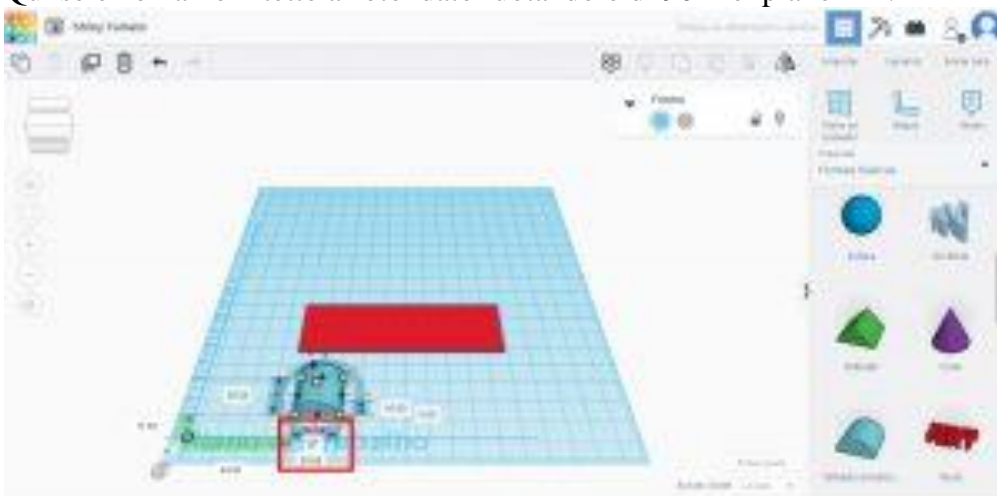


Step 2: utilizziamo la forma del "tetto arrotondato".

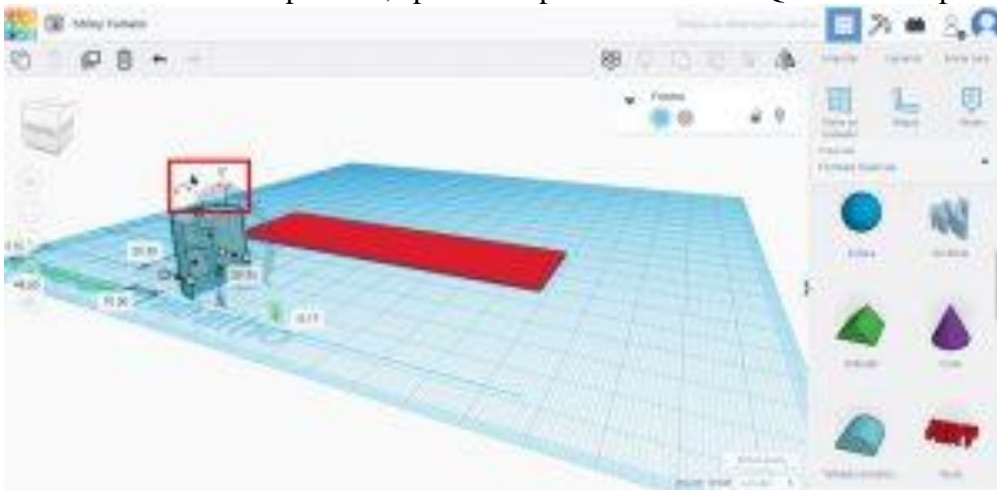
Per aggiungere la parte arrotondata al pezzo puoi usare due forme diverse: il cilindro (mettendone metà nel pezzo rettangolare), o il tetto arrotondato.



Qui selezioniamo il tetto arrotondato ruotandolo di 90° nel piano XY.



Per ruotare di 90° nel piano Z, spostare il piano del tavolo. Questo rende più facile la visione.

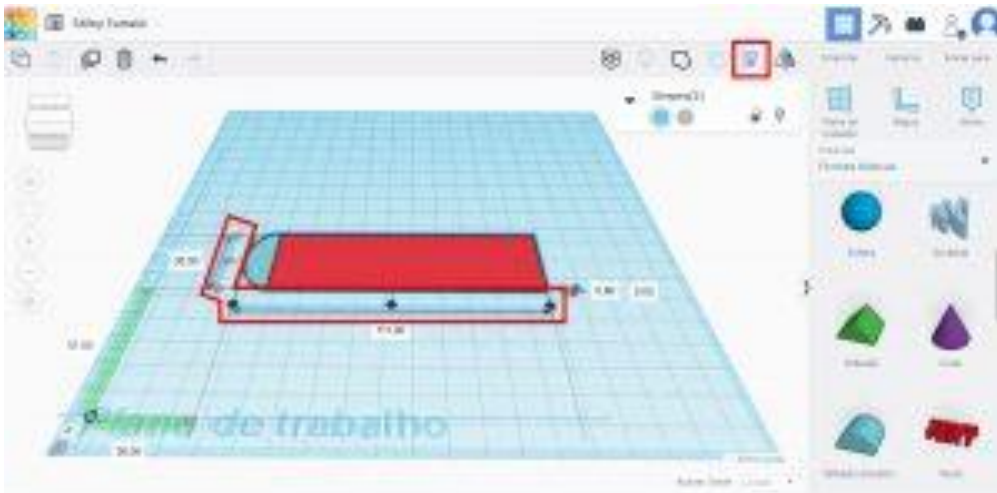


Quindi, modifica le dimensioni del tetto in 10 mm di lunghezza, 30 mm di larghezza e 0,8 mm di altezza. Per fare ciò, fai clic sui quadrati bianchi come mostrato nel passaggio precedente.

Step 3: allineare le parti

Posiziona i due pezzi che hai creato l'uno contro l'altro. Per allinearli, seleziona uno, tieni premuto "Shift" e fai clic sull'altro pezzo. Quindi, nell'angolo in alto a destra, seleziona lo strumento "Allinea".

Appariranno punti neri per scegliere il tipo di allineamento. Nel nostro caso, abbiamo lasciato le parti centrate.



Step 4: raggruppare le parti

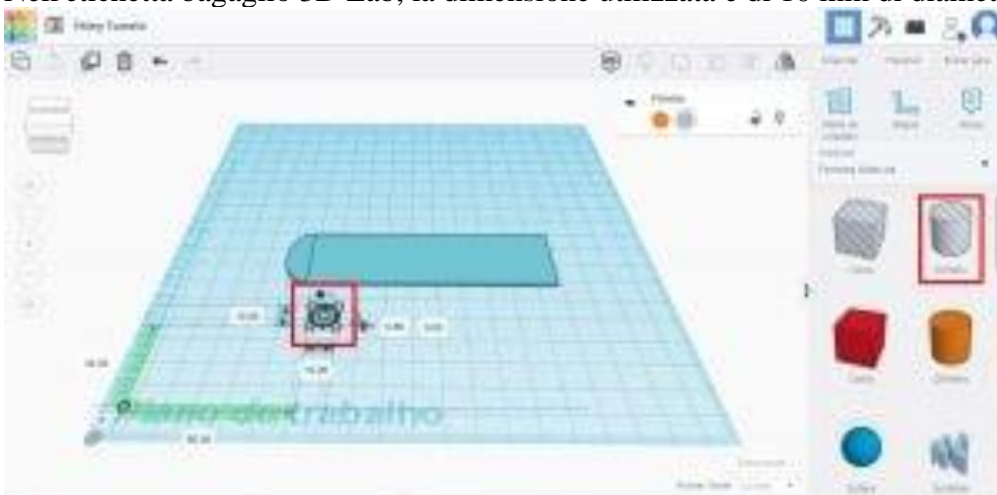
Ora li raggrupperemo! Per fare ciò, mantieni semplicemente le parti selezionate e, nell'angolo in alto a destra, fai clic sull'opzione "Raggruppa". D'ora in poi i due pezzi diventeranno uno solo.



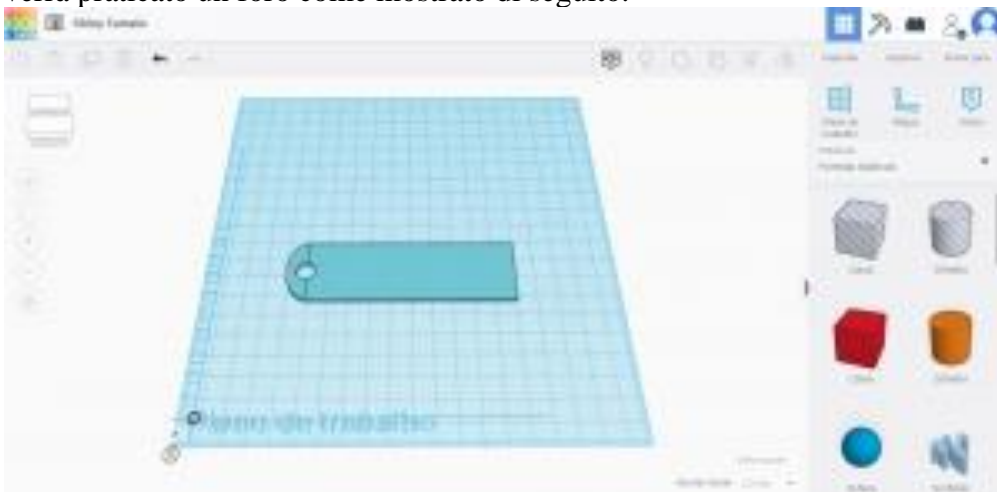
Step 5: Fare un buco nel pezzo

Esistono due modi per praticare un foro cilindrico. Il primo è selezionando il cilindro grigio (che rappresenta anche un buco). La seconda alternativa è selezionare il cilindro solido e trasformarlo in un foro. In entrambi i casi, trascinare il cilindro sopra la parte, modificarne le dimensioni e allineare le due parti.

Nell'etichetta bagaglio 3D Lab, la dimensione utilizzata è di 10 mm di diametro.

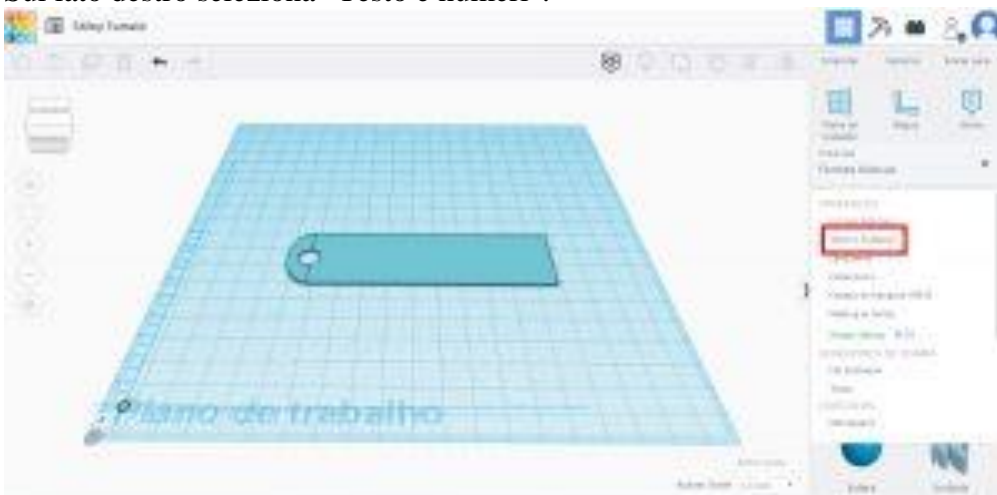


Quindi seleziona le due forme e fai clic su "Raggruppa". Nel punto in cui si trovava il cilindro, verrà praticato un foro come mostrato di seguito.

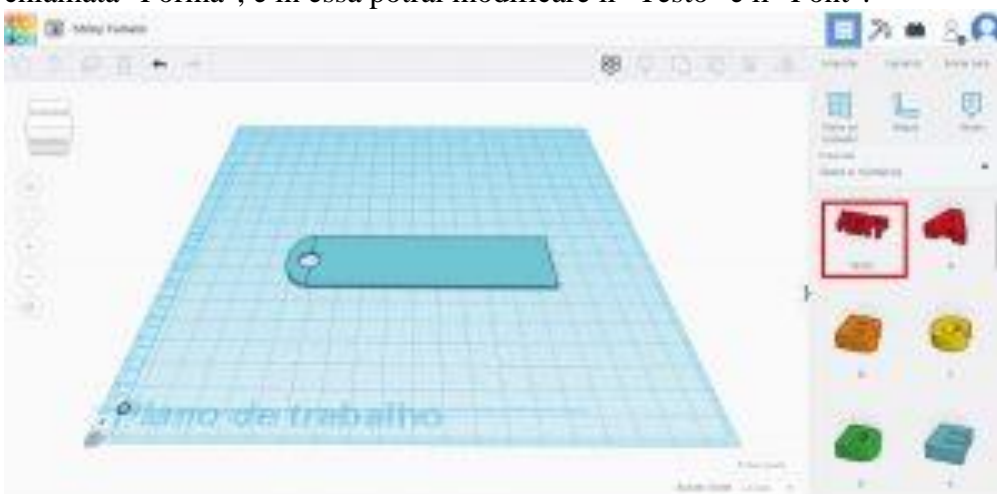


Step 6: aggiungere il testo

Sul lato destro seleziona "Testo e numeri".



Quindi trascina lo strumento "Testo" sul pezzo. A questo punto si aprirà una scheda laterale chiamata "Forma", e in essa potrai modificare il "Testo" e il "Font".

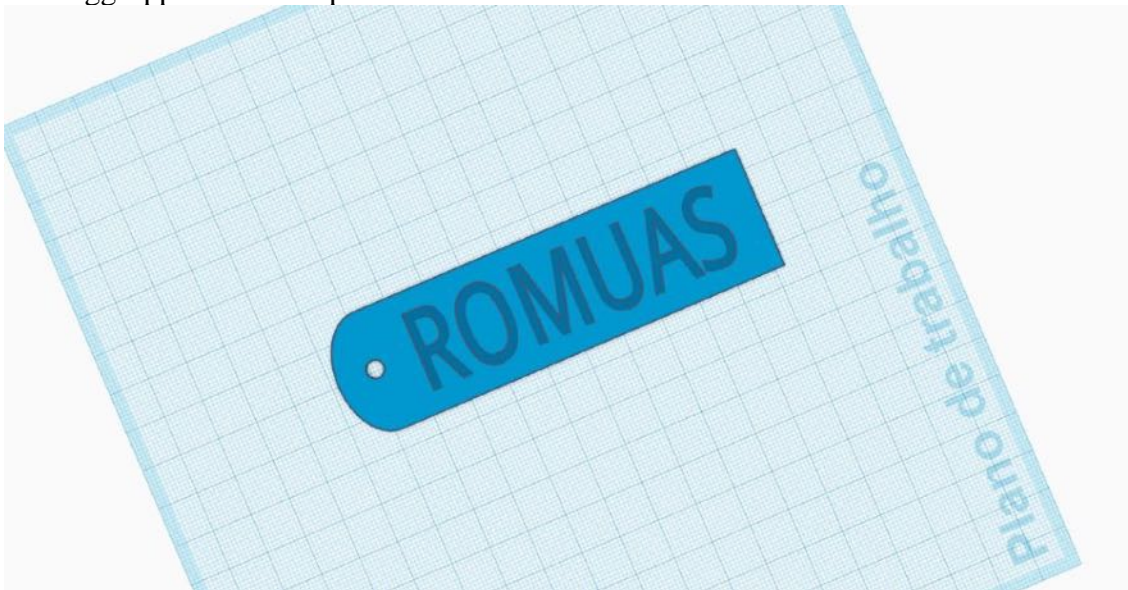


Abbiamo scelto per il nostro esempio il font Sans. Le dimensioni del testo sono 80 mm di lunghezza e 20 mm di larghezza.

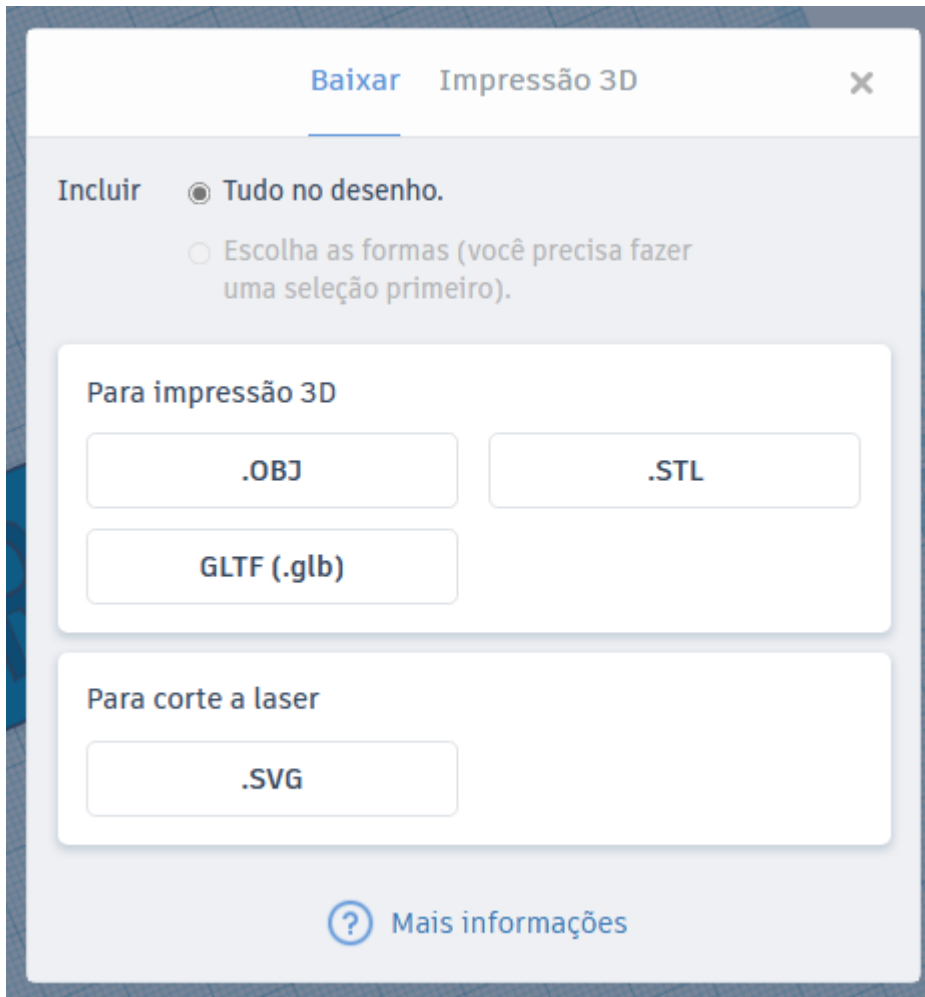
L'altezza utilizzata per il testo è di 2,0 mm dal piano di lavoro, in quanto pezzo e testo sono appoggiati al tavolo.



Ora raggruppa tutti i componenti che hai creato.

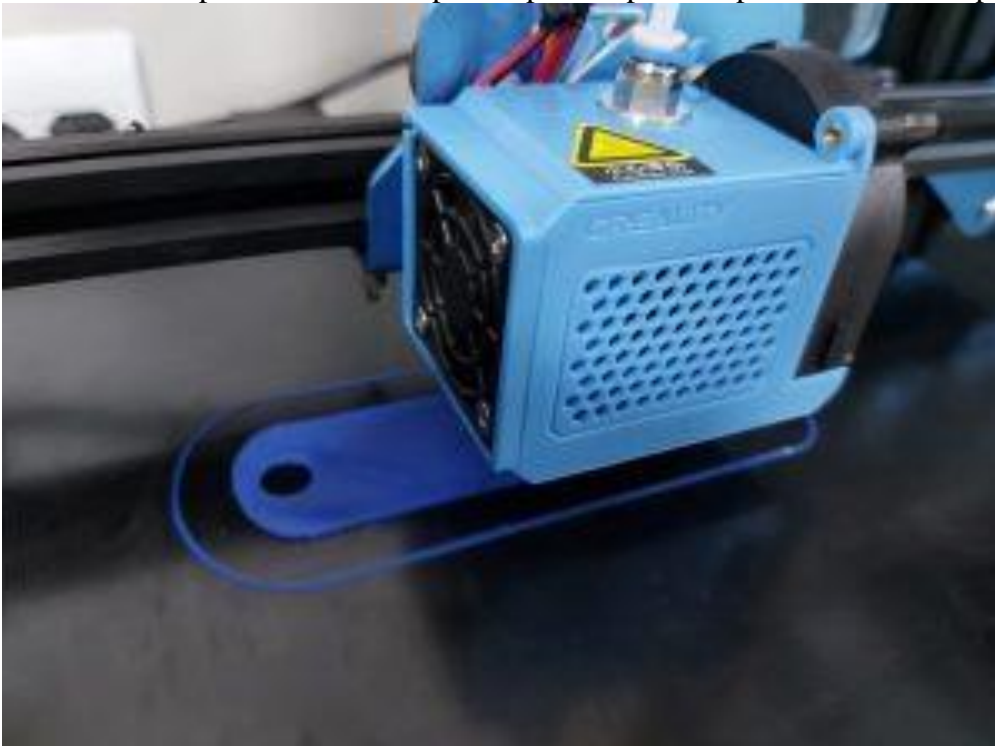


Una volta fatto questo, il tuo modello è finito! Per effettuare lo slicing e stampare, è sufficiente esportare come STL tramite lo strumento "Esporta". Si trova nell'angolo in alto a destra.



Step 7: la stampa

Infine, una volta salvato il disegno in STL, apri il file in un software di slicing a tua scelta. Seleziona le impostazioni di stampa e impostalo per stampare sulla tua stampante 3D.



2.1. Parte pratica (tutorial, video...)

I. Radiocomando analogico

Vecchio radiocomando

Come funziona

Chi non ha un ricevitore radio analogico? Questi dispositivi hanno una ruota per controllare il quadrante. Puoi vedere i numeri su un display analogico chiamato dial.



Esempio di ruota analogica e quadrante

Automatizzeremo la rotazione della ruota per cambiare la frequenza di ricezione. Avremo un'app nel cellulare che invierà la rotazione desiderata dei quadranti memorizzati all'Arduino e farà ruotare la ruota nella posizione esatta di questa frequenza. Eviteremo di aprire o rompere la radio e tutte le nostre modifiche dovrebbero essere esterne.

Appunti:

Ogni ricevitore radio è diverso e devi adattare i materiali e il manuale alla forma e alle dimensioni del tuo ricevitore radio.

Materiali

Abbiamo solo bisogno di Arduino Uno, un motore passo-passo con un driver appropriato e un controller Bluetooth per arduino.

Controllore Bluetooth:



Per comunicare con il controller Arduino con un telefono cellulare abbiamo bisogno di questo dispositivo. Ce ne sono molti e di solito sono economici.



Ci sono molti stepper e non tutti funzionano allo stesso modo. Se dobbiamo acquistarne uno, consigliamo di cercare per motore passo-passo (stepper motor) e Arduino per evitare problemi e trovare tutorial adattati a questo microcontrollore. Inoltre consigliamo di acquistare un motore passo-passo con il driver per non dover regolare tensione e intensità. Nell'esempio utilizzeremo un motore riutilizzato e dovremo utilizzare un altro driver e regolare.

Un motore passo-passo è più difficile da usare e più lento di un motore normale, ma ci dà un controllo preciso sui gradi esatti che vogliamo ruotare. E questo è ciò di cui abbiamo bisogno.

Non è facile collegare lo stepper alla radio. In questo esempio, mostreremo due possibili metodi. Abbiamo questo vecchio Sanyo e un motore passo-passo riutilizzato da una stampante:





Poiché possiamo manipolare l'intera ruota, abbiamo deciso di utilizzare una gomma e utilizzare il motore come puleggia:



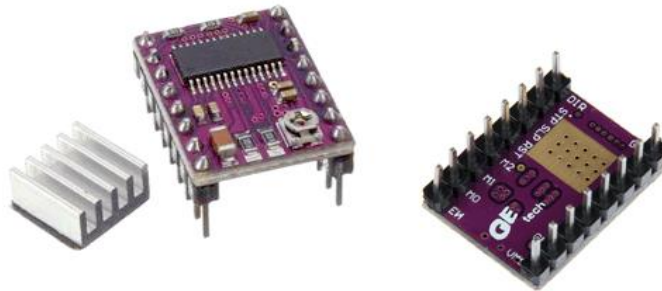
Ma se si tratta di una ruota inserita o è impossibile utilizzare una puleggia, possiamo incollare una ruota Lego o qualcosa di simile e utilizzarla per attrito. In questo caso, più piccola è la ruota, migliore sarà la precisione e la coppia:



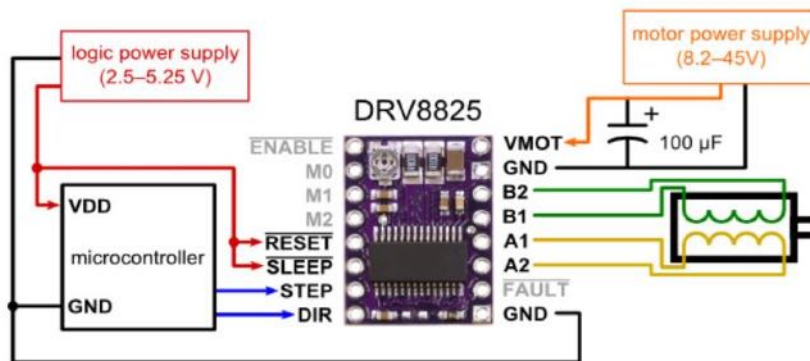
Schema:

Se hai un motore passo-passo con il suo driver, non devi preoccuparti della parte successiva. Nell'altro caso, è necessario un driver:

Controllare un motore passo-passo è molto difficile con Arduino "vanilla". E abbiamo anche un problema con le tensioni. Ma fortunatamente abbiamo driver sul mercato per controllarlo più facilmente. Possiamo cercare A4988 o DRV8825. Raccomandiamo il DRV8825 perché è superiore in alcune specifiche e il prezzo non è molto alto. Possiamo trovare alcuni modelli perché sono utilizzati in stampanti 3D e robot.

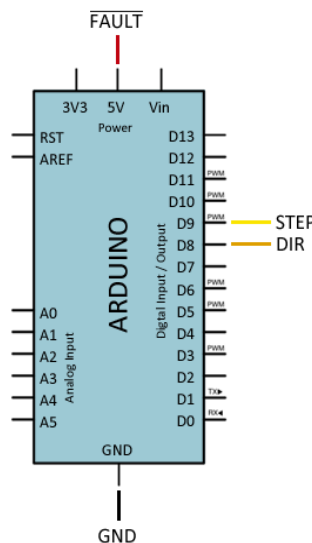


Lo schema dei driver è:

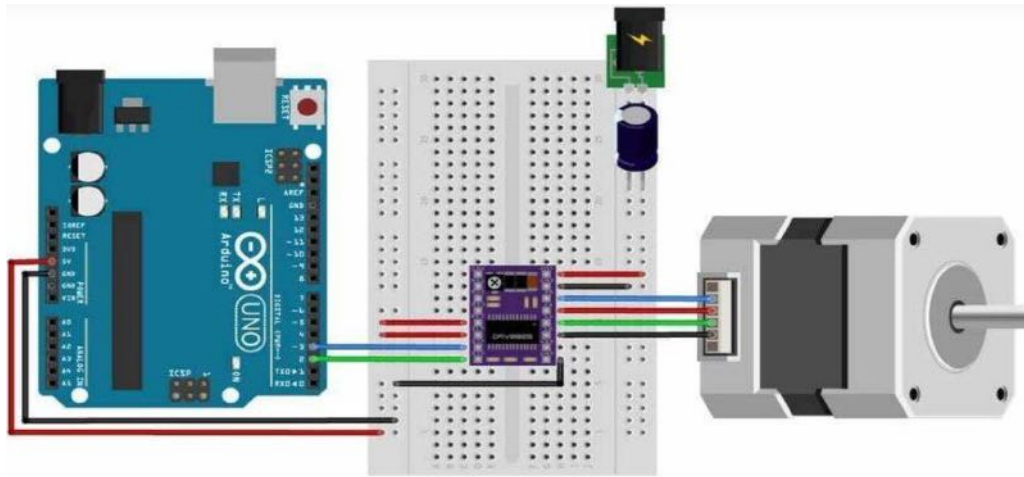


Minimal wiring diagram for connecting a microcontroller to a DRV8825 stepper motor driver carrier (full-step mode).

E in Arduino è:



Come puoi vedere, abbiamo solo bisogno di due perni per i passaggi e la direzione. Questo è lo schema completo:



Processo di assemblaggio

I motori hanno una tensione e un'intensità nominali. Nelle specifiche di ogni motore possiamo trovare una formula per calibrare i driver.

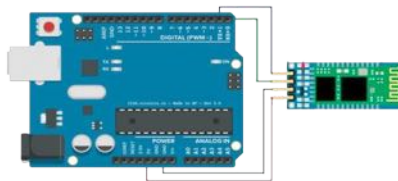
Si consiglia di leggere il manuale del driver e del motore per regolare correttamente la tensione e l'intensità e non romperne nessuno.

In generale i driver sono molto robusti e difficili da rompere durante il lavoro, ma possiamo romperli quando si scollega il motore quando il controller è alimentato. Per questo motivo dobbiamo seguire questi passaggi:

- Collegare il driver senza motore.
- Misurare la tensione tra GND e il potenziometro.
- Regolare il potenziometro fino a raggiungere il valore della formula.
- Spegnerne il driver.
- Collegare il motore e un amperometro al centro.
- Adattarsi all'intensità del motore.
- Spegnerne
- Montare il motore senza amperometro.
- Montaggio su Arduino.

Bluetooth:

È facile assemblare un controller bluetooth. Usa la porta seriale di Arduino e non possiamo usare la seriale per comunicare con il computer, ma possiamo comunicare tramite bluetooth.



Code:

Questo è un semplice esempio per provare il motore:

```
const int dirPin = 8;  
const int stepPin = 9;
```



```
const int steps = 200;
int stepDelay;

void setup() {
  pinMode(dirPin, OUTPUT);
  pinMode(stepPin, OUTPUT);
}

void loop() {
  //Direction and speed
  digitalWrite(dirPin, HIGH);
  stepDelay = 250;
  // 200 steps is a full turn
  for (int x = 0; x < steps * 1; x++) {
    digitalWrite(stepPin, HIGH);
    delayMicroseconds(stepDelay);
    digitalWrite(stepPin, LOW);
    delayMicroseconds(stepDelay);
  }
  delay(1000);
}
```

Per il controllo arduino, qui abbiamo un esempio minimo:

```
const int led = 13;
void setup(){
  Serial.begin(9600);
  pinMode(led, OUTPUT);
}
void loop(){
  if (Serial.available()>0){
    char option = Serial.read();
    if (option >= '1' && option <= '9')
    {
      option -= '0';
      for(int i=0;i<option;i++){
        digitalWrite(led, HIGH);
        delay(100);
        digitalWrite(led, LOW);
        delay(200);
        Serial.println(i);
      }
    }
  }
}
```

Possiamo fare un programma iniziale che conti quanti passi ci servono per fare tutto il quadrante. Questa è la calibrazione del sistema radio analogico Arduino. Mettiamo a mano il quadrante nella prima frequenza e con un'app per fare il telecomando Bluetooth, inviamo una quantità di passi e Arduino ci restituisce il conteggio dei passi:

```
const int dirPin = 8;
const int stepPin = 9;

const int steps = 200;
int stepDelay = 250;
int countSteps = 0;

void setup() {
  pinMode(dirPin, OUTPUT);
  pinMode(stepPin, OUTPUT);
  Serial.begin(9600);
}

void loop() {

  if (Serial.available()>0){
    char option = Serial.read();
    if (option >= '1' && option <= '9')
    {
      option -= '0';
```



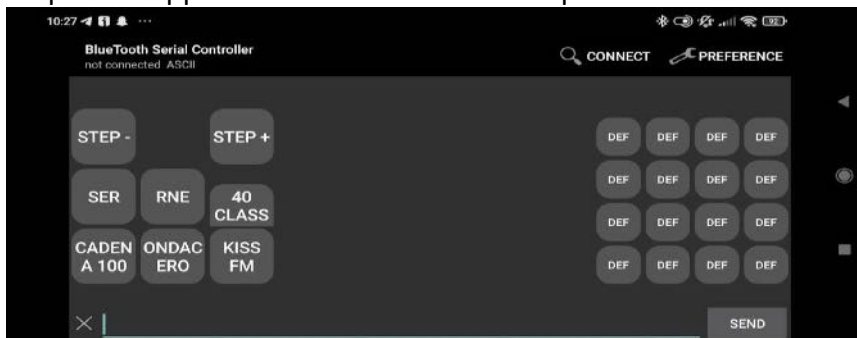


```
digitalWrite(dirPin, HIGH);  
digitalWrite(stepPin, HIGH);  
delayMicroseconds(stepDelay);  
digitalWrite(stepPin, LOW);  
countSteps += option;  
}  
Serial.println(countSteps);  
}  
}
```

Immagina di aver bisogno di 1000 passaggi per completare il quadrante della radio di esempio. Possiamo codificare questo numero in un nuovo codice come numero massimo di passaggi. Il nuovo codice deve conoscere la posizione attuale della ruota e riceverà il numero di passi dall'inizio per una determinata stazione. Possiamo creare un pulsante per le nostre stazioni preferite in un'app per il controllo del bluetooth e mettere la stazione desiderata con un pulsante. Qui abbiamo un codice semplificato:

```
const int dirPin = 8;  
const int stepPin = 9;  
  
const int steps = 200;  
int stepDelay = 250;  
int countSteps = 0;  
  
void setup() {  
  pinMode(dirPin, OUTPUT);  
  pinMode(stepPin, OUTPUT);  
  Serial.begin(9600);  
}  
  
void loop() {  
  if (Serial.available()>0){  
    int dial = Serial.parseInt();  
    int stepsDifference = dial - countSteps;  
    if(stepsDifference < 0){  
      digitalWrite(dirPin, HIGH);  
    }  
    else {  
      digitalWrite(dirPin, LOW);  
    }  
    for(i=0;i<abs(stepsDifference);i++){  
      digitalWrite(stepPin, HIGH);  
      delayMicroseconds(stepDelay);  
      digitalWrite(stepPin, LOW);  
      countSteps += option;  
      Serial.println(countSteps);  
    }  
  }  
}
```

E questo è un esempio dell'app Bluetooth Serial Controller per Android:



Devi solo assegnare il numero di passaggi dall'inizio a ciascun quadrante.

II. Controllo della pompa dell'acqua e della valvola

Irrigazione a goccia per giardino e campi

Come funziona

Il punto di partenza è una valvola di apertura manuale e una pompa dell'acqua elettrica per fornire la pressione necessaria. Questa situazione è inefficiente perché devi aprire la valvola, avviare la pompa, attendere e osservare se il campo è irrigato a sufficienza e poi fermare la pompa e chiudere la valvola. È difficile non usare troppa o poca acqua ed è necessario rimanere durante l'irrigazione. Un miglioramento è installare un programmatore per irrigare alcuni minuti ogni giorno. Il problema con il programmatore è il tempo, perché irrigherà anche in una giornata piovosa e potrebbe non essere acqua sufficiente in una giornata estiva molto calda. C'è un altro problema: le pompe dell'acqua spesso forniscono più pressione e flusso del necessario. Sono sovradimensionati. Provoca una buona irrigazione perché i sistemi di irrigazione a goccia possono autocompensarsi e far scorrere gli stessi litri al minuto. Ma la pompa dell'acqua consuma più energia del necessario. Il nostro sistema è dotato di sensori di umidità sul campo e li utilizza per decidere quando avviare l'irrigazione. Inoltre, ha un sensore di pressione e un driver del motore per controllare la velocità della pompa dell'acqua per raggiungere la pressione minima richiesta.

Note:

Questo documento descrive uno scenario completo. Se è troppo difficile o lungo, puoi scegliere la pompa con sensore di pressione o la valvola con sensore di umidità.

Materiali

Nei test di laboratorio è necessario un set realistico per ricreare uno scenario reale. Abbiamo bisogno di un vaso con piante poste più in alto, di un gocciolatore autocompensato e di una manichetta. Per la parte elettronica abbiamo bisogno di una pompa dell'acqua, un driver del motore, un sensore di umidità e un arduino. Ora andiamo più nello specifico:

Dripper:



I gocciolatori tipo Katif sono i più facili per l'installazione compatta. Puoi scegliere Katif per 1, 2 o 4 litri all'ora. Raccomandiamo meno acqua all'ora per il set da laboratorio. Sono gocciolatori autocompensanti e rilasciano sempre la stessa acqua indipendentemente dalla pressione. Si consiglia di utilizzare un tubo flessibile lungo perché durerà più tempo per il riempimento e sarà più realistico.

Pompa dell'acqua:



Esistono molte pompe dell'acqua a 12 V per il campeggio o gli acquari. Arduino funziona con 3v e 5v e per la protezione elettronica non è consigliabile spostare un motore direttamente da Arduino. Per questo motivo, dobbiamo utilizzare un driver del motore. Trasformerà il nostro segnale PWM 5v in un segnale 12v per diverse velocità nel motore.

Motore

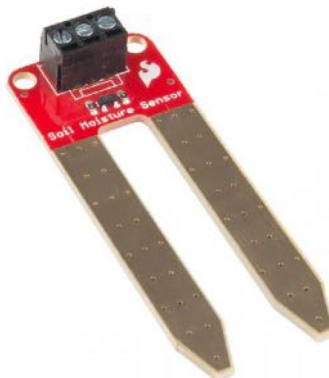
Driver:



L'Arduino Motor Shield ufficiale è quello che consigliamo. Non è molto costoso, compatibile con Arduino Uno e puoi trovare molta documentazione a riguardo.

Uno shield è una scheda che può essere posizionata sopra l'Arduino e i cui connettori si adattano perfettamente all'Arduino Uno. Lo scudo utilizza alcuni dei pin di Arduino ed espone tutti gli altri.

Sensore di umidità:



Questo sensore agisce misurando la conduttività del terreno. Ci sono molti sensori diversi e dobbiamo calibrare ogni volta, perché la sensibilità di ciascuno potrebbe essere diversa. In questo esempio utilizzeremo il modello Sparkfun che necessita di due fili per l'alimentazione e uno per il segnale analogico.

Pressure sensor:



Questo sensore di pressione deve essere installato nel tubo dopo la pompa. Ha tre fili: due per l'alimentazione e uno per il segnale.

Schema:

Innanzitutto, dobbiamo spiegare i pin utilizzati per lo scudo del motore Arduino:

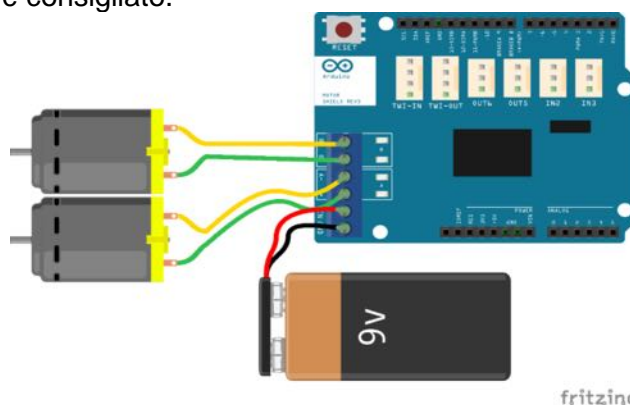
Le specifiche tecniche e le modalità di montaggio sono consultabili qui: <https://store.arduino.cc/arduino-motor-shield-rev3>

Questo scudo ha due canali separati, chiamati A e B. Ognuno utilizza 4 dei pin Arduino per guidare o rilevare il motore. In totale, ci sono 8 pin utilizzati da questo shield. È possibile utilizzare ciascun canale separatamente per pilotare due motori CC o combinarli per pilotare un motore passo-passo bipolare. I pin dello shield, divisi per canale, sono mostrati nella tabella sottostante:

Function	pins per Ch. A	pins per Ch. B
<i>Direction</i>	D12	D13
<i>PWM</i>	D3	D11
<i>Brake</i>	D9	D8
<i>Current Sensing</i>	A0	A1

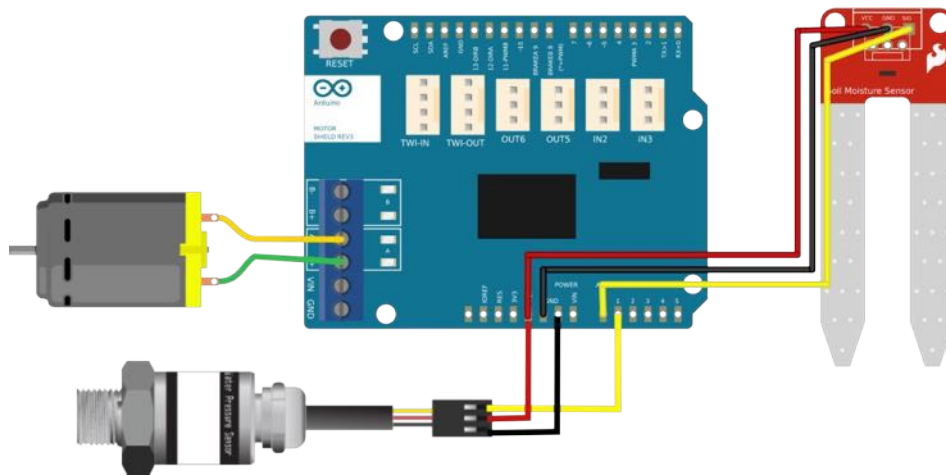
Se non hai bisogno del Brake e del Current Sensing e hai anche bisogno di più pin per la tua applicazione, puoi disabilitare questa funzione tagliando i rispettivi jumper sul lato posteriore dello shield.

Questo schema mostra i collegamenti del motore. Lo schermo del motore può alimentare la scheda Arduino, ma questo non è consigliato.



Questo schema è per 2 motori, ma ne serve solo uno (A), quindi i pin D13, D11, D8, A1 possono essere usati per altre cose.

Questo è lo schema dello scopo. La scheda Arduino è sotto l'Arduino Motor Shield. Alcuni pin possono essere modificati se ne abbiamo bisogno per altri scopi.



Note:

Le uscite digitali possono essere utilizzate per aprire le valvole, se necessario.

Anziché alimentare il sensore tramite i pin da 3,3 V o 5 V, è possibile utilizzare un pin digitale per alimentare il sensore. Ciò impedirà la corrosione del sensore mentre si trova nel terreno.

Questo è un esempio di codice per la lettura del sensore di umidità. Può essere utilizzato anche per leggere il sensore di pressione cambiando il pin analogico:

```
int val = 0; //value for storing moisture value
int soilPin = A0;//Declare a variable for the soil moisture sensor
```

```
void setup()
{
  Serial.begin(9600); // open serial over USB
}
```

```
void loop()
{
  Serial.print("Soil Moisture = ");
  //get soil moisture value from the function below and print it
  Serial.println(readSoil());
}
```

```
//This 1 second timeframe is used so you can test the sensor and see it change in real-time.
//For in-plant applications, you will want to take readings much less frequently.
delay(1000);//take a reading every second
}
```

```
//This is a function used to get the soil moisture content
int readSoil()
{
  delay(10);//wait 10 milliseconds
  val = analogRead(soilPin);//Read the SIG value form sensor
  return val;//send current moisture value
}
```

La calibrazione è difficile perché non è solo l'umidità, ma anche il tipo di terreno perché il sensore si basa sulla conduttività e i materiali del terreno possono modificarla. Si consiglia di leggere l'output seriale e di provare con sempre meno acqua nel terreno. Una volta individuati il minimo e il massimo, puoi utilizzare la funzione `map()` per ottenere un numero normalizzato. (<https://www.arduino.cc/reference/en/language/functions/math/map/>)

Se volessimo accendere il motore quando l'umidità raggiunge un valore inferiore ad una soglia è facile scrivere nei pin D12 e D3 un valore PWM.

**PWM:**

Un'uscita digitale può cambiare lo stato molto rapidamente, così rapidamente che non possiamo vedere il cambiamento. Ma se colleghiamo un LED a questo pin, possiamo vedere meno luce. Questo si chiama "Pulse With Modulation" Arduino semplifica il codice fornendo alcuni pin con PWM e dobbiamo solo chiamare la funzione analogWrite

```
int val = 0; //value for storing moisture value
int soilPin = A0;//Declare a variable for the soil moisture sensor

void setup() {
  Serial.begin(9600); // open serial over USB
  //Setup Channel A
  pinMode(12, OUTPUT); //Initiates Motor Channel A pin
  pinMode(9, OUTPUT); //Initiates Brake Channel A pin
}

void loop(){

Serial.print("Soil Moisture = ");
int moisture = readSoil();
Serial.println(moisture);

if(moisture < 100) {
  //Motor A forward @ full speed
  digitalWrite(12, HIGH); //Establishes forward direction of Channel A
  digitalWrite(9, LOW); //Disengage the Brake for Channel A
  analogWrite(3, 255); //Spins the motor on Channel A at full speed
}
delay(1000);//take a reading every second
}

int readSoil()
{
  delay(10);//wait 10 milliseconds
  val = analogRead(soilPin);//Read the SIG value form sensor
  return val;//send current moisture value
}
```

L'esempio sopra mostra come far funzionare completamente la pompa. Ma la pressione può essere eccessiva e possiamo controllarla con un sensore di pressione. Possiamo aggiungere questo codice per leggere la pressione:

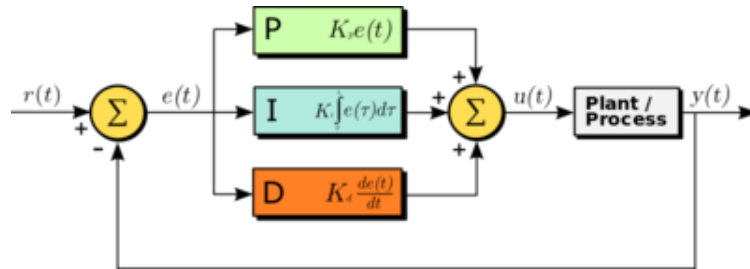
```
int press = 0;
int pressurePin = A1;

...

int readPressure()
{
  delay(10);//wait 10 milliseconds
  press = analogRead(pressurePin);//Read the SIG value form sensor
  return press;
}
```

Dobbiamo scrivere un valore inferiore al pin 3 se la pressione è alta. Il problema è il tempo di reazione della pressione nel tubo. Se la pressione è troppo bassa può inviare un valore elevato al pin PWM e la pressione può essere aumentata troppo velocemente. L'autocalibrazione durante l'irrigazione è difficile per una macchina. Fortunatamente, la soluzione è applicare un algoritmo PID molto ben documentato.

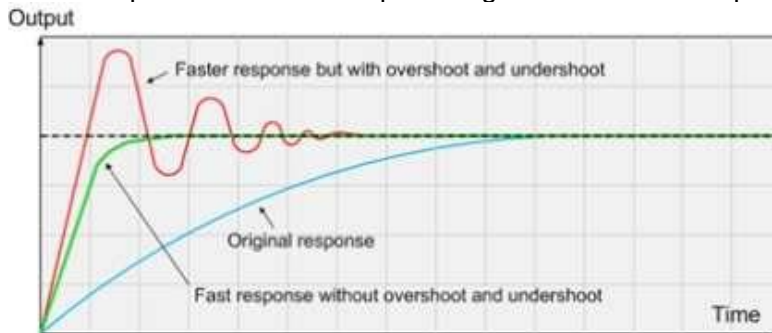




Questo è lo schema del PID. Ha tre componenti:

- Controllo proporzionale: controlla la correzione attuale della velocità.
- Integrativo: accumula gli errori passati per correggere la velocità.
- Derivata: prevede gli errori "futuri" per correggere la velocità.

Dipende dalla calibrazione di questi tre fattori la risposta regolare al sensore di pressione.



Questo è un esempio di come utilizzare il codice per PID:

```

error = (pressure - IDEAL_PRESSURE); // Proportional control
error = error / 10;
if ((long)error*(long)integral)<0
  { integral=0; } // integral Windup.
integral = integral + error; // Integrative
derivative = error - before_error; // Derivative
correction = Kp * error + Ki * integral + Kd * derivative;
pumpSpeed = speed_base + correction;
before_error = error; // for derivative
  
```

K_p , K_i , K_d sono le variabili di calibrazione PID che devono essere inizializzate dopo la sperimentazione. Si consiglia di mettere 1 in tutto e di iniziare la calibrazione con K_p . Quando funziona come la linea rossa, possiamo toccare gli altri per raggiungere la linea verde.

III. Impianto di riscaldamento con controlli di temperatura e tempo



Fig. 1 Prototipo finale

Introduzione

L'obiettivo del progetto è quello di recuperare un vecchio riscaldatore elettrico da laboratorio di chimica e dotarlo di un controllo di temperatura e tempo. Il controllo può essere utilizzato anche in dispositivi industriali.

È stato scelto un controllo ON-OFF per la sua semplicità hardware e per il basso costo e la facilità di reperimento dei componenti.

Tutte le impostazioni vengono eseguite da due encoder rotativi con effetto pulsante che fa clic sulle manopole. Le impostazioni sono visualizzate da un display LCD a 2 righe e 16 caratteri.

Poiché alcune parti di questo circuito coinvolgono la tensione di rete, è assolutamente importante la supervisione di istruttori adeguatamente formati.

Hardware

La misurazione della temperatura viene eseguita da un DS18B20. È un termometro digitale che fornisce letture della temperatura da 9 a 12 bit (configurabili) che indicano la temperatura del dispositivo. Le informazioni vengono inviate al/dal DS18B20 tramite un'interfaccia 1-Wire, in modo che sia necessario collegare solo un filo (e terra) da un microprocessore centrale a un DS18B20. L'alimentazione per la lettura, la scrittura e l'esecuzione delle conversioni di temperatura può essere derivata dalla linea dati stessa senza necessità di una fonte di alimentazione esterna. Ha le seguenti caratteristiche:

- L'esclusiva interfaccia 1-Wire richiede un solo pin di porta per la comunicazione
- La funzionalità multidrop semplifica le applicazioni di rilevamento della temperatura distribuito
- Non richiede componenti esterni
- Può essere alimentato dalla linea dati. L'intervallo di alimentazione è compreso tra 3,0 V e 5,5 V
- Nessuna alimentazione in standby richiesta
- Misura temperature da -55°C a +125°C. L'equivalente Fahrenheit va da -67°F a +257°F
- Precisione $\pm 0,5^\circ\text{C}$ da -10°C a +85°C
- La risoluzione del termometro è programmabile da 9 a 12 bit
- Converte la temperatura a 12 bit in parola digitale in 750 ms (max.)
- Impostazioni di allarme temperatura non volatile definibili dall'utente

- Il comando di ricerca allarmi identifica e indirizza i dispositivi la cui temperatura è al di fuori dei limiti programmati (condizione di allarme temperatura)
- Le applicazioni includono controlli termostatici, sistemi industriali, prodotti di consumo, termometri o qualsiasi sistema termicamente sensibile.

Invece di potenziometri più vecchi e meno flessibili, sono stati scelti encoder rotativi più moderni. Nell'immagine sottostante (fig. 2) è possibile vedere sia la vista dall'alto che quella laterale del modulo encoder da utilizzare. Nella vista dall'alto sono ben visibili tutti i pin di connessione.

La Fig. 3 rappresenta entrambi i segnali A (CLK) e B (DT). Ogni passo di rotazione fa cambiare A e B:

- una rotazione in senso orario fa cambiare A prima di B
- una rotazione in senso antiorario fa cambiare A dopo B.

In sostanza, determinare quale interruttore ha cambiato stato per primo è il modo in cui viene rilevata la direzione di rotazione. Scambiando i perni A e B si invertiranno di conseguenza i sensi di rotazione di salita e discesa.

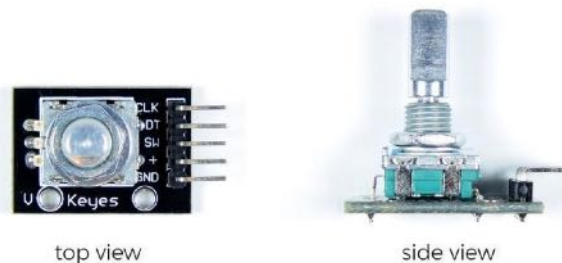


Figura 2 - Encoder view

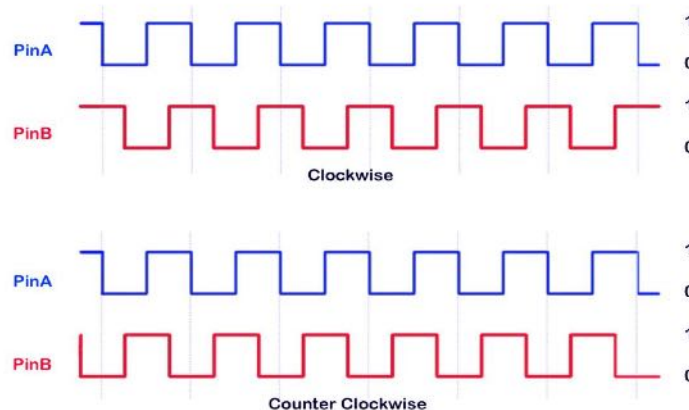


Fig. 3 - Rotary encoder signals

Questi segnali possono essere gestiti direttamente (usando la funzione di interrupt per leggere il pin CLK). Tuttavia, nel nostro caso, questa attività viene eseguita da una libreria per semplificare il programma. A seconda delle connessioni dei pin A e B, possono essere necessarie alcune correzioni. Nel programma (sezione Encoder loop - lettura manopole) il valore letto viene invertito (altrimenti è possibile scambiare i collegamenti dei pin A e B) e diviso per due (la funzione di libreria legge sia i fronti di salita che quelli di discesa).

Riepilogo degli elementi del sistema

- Arduino Uno o compatibile
- Sensore di temperatura digitale DS18B20



- Codificatore di temperatura (con interruttore a scatto) da 20°C a 125°C
- Encoder timer (con interruttore a scatto) da 00h00m a 99h00
- Display LCD 1602 I2C
- Cicalino attivo
- Modulo relè optoisolato 5Vdc – 250Vac, 10A
- Alimentazione CC; ingresso 240Vac, 40ma; uscita 12Vdc, 250mA;
- Fusibile 250V, 10A ritardato
- Presa comandata 230Vac (per collegare la resistenza)

Caratteristiche del Sistema

- La temperatura è controllata in modalità ON-OFF, utilizzando un relè optoisolato come interfaccia di alimentazione.
- Una funzione di reset globale (sia dell'ora che della temperatura) è disponibile cliccando sulla manopola della temperatura.
- impostare il tempo da 1 minuto a 99 ore;
- temperatura impostata da 20 a 125°C;
- range di variazione della temperatura intorno alla temperatura impostata: +/- 0,5°C attorno ad essa;
- un beep acustico conferma il raggiungimento della temperatura;
- La temperatura e l'ora impostate e correnti vengono visualizzate sul display LCD;
- time and temperature reset;
- temperature probe;
- maximum load of 2000W;
- protection fuse (10A);

Come funziona (fare riferimento allo schema elettrico)

Il sistema è in grado di mantenere pressoché costante (temperatura impostata $\pm 0,5^\circ\text{C}$) la temperatura di a

sostanza. Si collega direttamente ad una presa 230Vac e fornisce una presa comandata on-off collegare la spina del dispositivo di riscaldamento (corrente massima=10A)

Per rilevare la temperatura della sostanza viene utilizzato un sensore digitale DS18B20. Ha un acciaio inossidabile

involucro e può essere immerso in sostanze liquide.

Per essere più reattivi, sia gli encoder temporali che quelli di temperatura vengono letti dalle routine di interrupt.

E' possibile impostare sia la temperatura (20÷125°C) che il tempo tramite due switch-encoder come segue:

- Temperatura impostabile agendo sulla manopola sinistra Temperatura (2), e visualizzabile sul display (a)
- Il tempo può essere impostato agendo sulla manopola destra Time (3), e visualizzato sul display (c)
- Cliccando sulla manopola Time, è possibile avviare il riscaldamento; l'attivazione è confermata da un suono "beep" (5)
- Inoltre, il display mostra la temperatura attuale (b), misurata dal sensore (1)
- Anche il tempo rimanente è visibile sul display (d)
- Al raggiungimento della temperatura impostata un doppio "beep" lo conferma (5)
- Quando il timer è attivo sul display viene visualizzato il simbolo "T" (e)
- Quando l'elemento riscaldante è attivo viene visualizzato un simbolo "H" sul display (f), e l'elettrico

presa sul dispositivo forniscono 230V per alimentare l'elemento riscaldante.

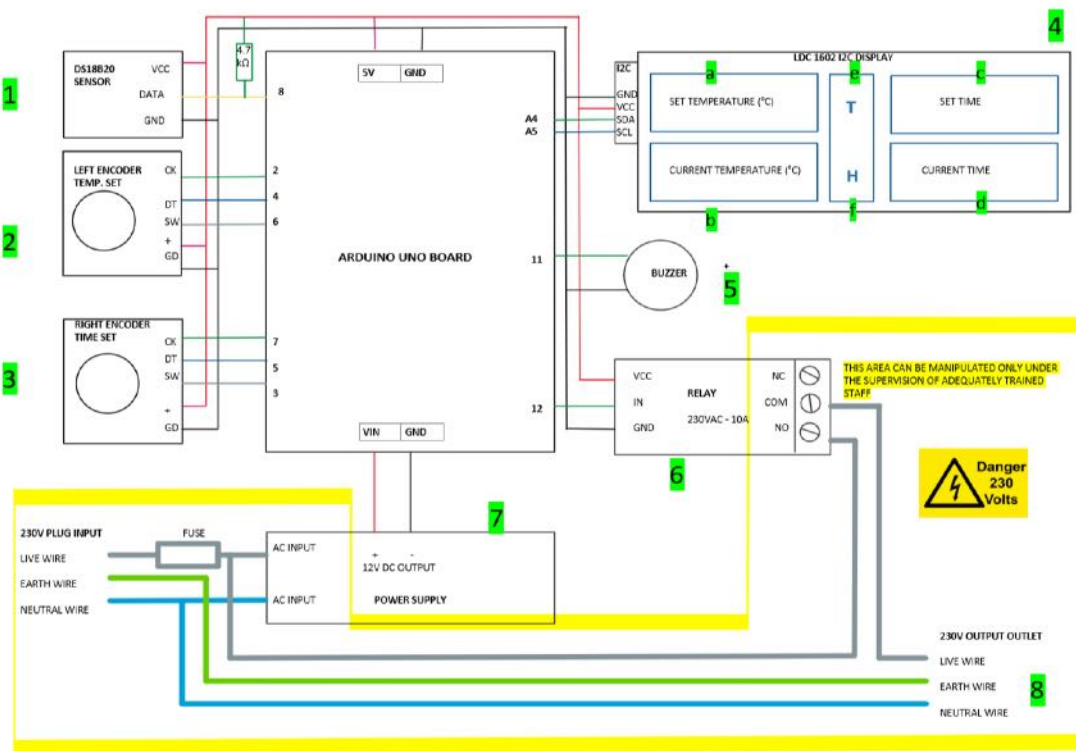





Fig. 4 - Schema elettrico

Dettaglio dei componenti

			Note
1	Arduino Uno		Possono essere utilizzate anche diverse schede Arduino.
2	DS18B20 sonda di temperatura		È necessaria una resistenza di pull-up da 4,7kOhm tra il pin dati (giallo, collegato all'ingresso digitale 8 della scheda Arduino) e Vcc poiché il livello HIGH della scheda è 5V mentre l'uscita del sensore è 3,3V. Per gestire facilmente questo sensore digitale viene utilizzata un'apposita libreria.
3	1602 display con modulo I2C		L'interfaccia I2C è necessaria per semplificare il circuito e ridurre il numero di pin Arduino necessari per gestire il display. Le librerie sono utilizzate sia per il display LCD che per il modulo I2C.

3	Power adapter 230Vac-12Vdc - 250mA (AZ Consegna o equivalente)		È possibile utilizzare anche altri adattatori di alimentazione. Arduino Uno accetta Vin da 7V a 12V.
4	Rotary encoder		Una libreria viene utilizzata per gestire entrambi gli encoder rotativi. A seconda del modulo rotante utilizzato può essere necessario apportare alcune correzioni al codice (Encoder loop - sezione lettura manopole)
5	Opto-isolated relay module 5Vdc – 250Vac, 10A		Verificare se l'ingresso del modulo è attivo livello ALTO o BASSO. Nel programma sottostante è considerato attivo ALTO. Viceversa sarà necessario modificare opportunamente il programma nella sezione "CONTROLLO RISCALDAMENTO".
6	Active buzzer - 5V		Il classico cicalino attivo a 5V spesso incluso nei kit Arduino
7	Fuse 250V - 10A Retarded		Protezione del circuito da sovraccarico. È richiesto anche un portafusibile

Program structure

1. Installazione librerie

- Wire.h>
- <LiquidCrystal_I2C.h>
- <OneWire.h>
- <DallasTemperature.h>
- <Encoder.h>

2. Definizione degli oggetti

3. Definizioni di variabili e costanti

Setup

- I/O configuration
- Serial Monitor configuration
- LCD initialization

Loop

1. Impostazioni e controlli della temperature
2. Impostazioni e controlli dell'ora
3. Encoders loop - lettura manopole
4. Reset del valore delle manopole
5. Controllo del timer
6. Controllo del riscaldamento
7. Controllo degli allarmi
8. Controllo del monitor seriale (solo per il debug)
9. Controllo dell'affissione a cristalli liquidi



- Impostazione temperatura LCD
- LCD Imposta ora
- Tempo rimanente LCD
- Indicatori LCD

Funzioni del cicalino

1. Beep
2. Beep2

Programma

```
//***** HEATER DS18B20 FULL PROGRAM *****

#include <Wire.h>
#include <LiquidCrystal_I2C.h> // I2C LCD library
#include <OneWire.h> // Needed library for DS18B20
#include <DallasTemperature.h> //DS18B20 library
#define ONE_WIRE_BUS 8 // DQ sensor pin
#include <Encoder.h> //Encoder library

Encoder knobLeft(2, 4); //Temperature control encoder
Encoder knobRight(3, 5); //Time control encoder
const int swLeftPin=6; //Reset pin
const int swRightPin=7; //Timer start pin
const int heatPin=12; //Heater relay pin
const int buzzerPin=11; //Buzzer pin

boolean swRightState=1; //Start timer button state
boolean swLeftState=1; //Temperature ad Time reset button state
float setTemp=0.0; //Set temperature
const float stepTemp=1.0; //Set temperature knob step
const int minTemp=20; //Minimum temperature admitted
const int maxTemp=125; //Maximum temperature admitted
const float dT=0.5; //Admitted range (+/-) around set temperature
float sensTemp; //Sensor temperature value

boolean heatFlag=0; //Heating activation flag
boolean knobChangeFlag; //Changing set temperature flag
boolean timerFlag=0; //Activation timer flag

unsigned long maxMin=5940; //Maximum minutes (99 hours)
unsigned long setMillis; //Set milliseconds
unsigned long currentMillis; //Current milliseconds
unsigned long startMillis; //Milliseconds value when the timer starts
unsigned long elapsedMillis; //Milliseconds elapsed from the timer start

unsigned int elapsedMinute=0; // Minutes elapsed from the timer start
unsigned int remainingMinute; //Remaining minutes
unsigned int currentMinute=0; //Current minute displayed on the LCD
unsigned int currentHour=0; // Current hour displayed on the LCD
unsigned int setHour=0; //Set hour
unsigned int setMinute=0; //Set minutes

//Encoder pins and variables
long tempEncValue; //Temperature encoder value (left)
unsigned long timeEncValue; // Time encoder value in minutes (right)
long positionLeft = 0;
long positionRight = 0;

OneWire oneWire(ONE_WIRE_BUS); // Set OneWire connection
DallasTemperature sensor(&oneWire); // Sensor object declaration

LiquidCrystal_I2C lcd(0x27,16,2); // set the LCD address to 0x27for the display

void setup()
{
  pinMode(swRightPin, INPUT_PULLUP); //Input-Output configuration
  pinMode(swLeftPin, INPUT_PULLUP);
  pinMode(heatPin, OUTPUT);
  pinMode(buzzerPin, OUTPUT);
  pinMode(LED_BUILTIN, OUTPUT);

  Serial.begin(9600); // initialize the Serial Monitor
```





```
lcd.init(); // initialize the lcd
lcd.backlight();
sensore.begin(); // initialize the sensor
lcd.setCursor(10, 1); // set the cursor to column 10, line 1
lcd.print("00h00m"); // current time reset on LCD.
}

void loop(){

//Temperature settings and controls
setTemp=minTemp+tempEncValue*stepTemp; //Temperature setting
if(setTemp>maxTemp) knobLeft.write((maxTemp-minTemp)*2); // Max set temperature control
if(setTemp<minTemp) knobLeft.write(0); // Min set temperature control

//Time settings and controls
if(timeEncValue>maxMin) knobRight.write(0); // Max set time control
if(timeEncValue<0) knobRight.write(0); //Zero minutes control

setHour=timeEncValue/60;
setMinute=timeEncValue % 60;

sensore.requestTemperatures(); // Temperature sensor request
sensTemp = sensore.getTempCByIndex(0); // Sensor temperature in °C

//Encoders loop - knobs reading
long newLeft, newRight;
newLeft = knobLeft.read(); //Temperature knob reading
newRight = knobRight.read(); //Time knob reading

if (newLeft != positionLeft || newRight != positionRight) {
tempEncValue=-newLeft/2; //Temp encoder correction *(-
1/2)
timeEncValue=-newRight/2; //Time encoder correction *(-
1/2)
positionLeft = newLeft; //Temp. encoder position update
positionRight = newRight; //Time encoder position update
knobChangeFlag=HIGH; //knobs changing flag activation
}

//Knobs value reset
swLeftState=digitalRead(swLeftPin); //reset button reading
if(swLeftState==LOW){ //if active => reset knobs value
knobRight.write(0);
knobLeft.write(0);
timerFlag=LOW; //Timer flag reset

elapsedMillis=0; //Elapsed time reset
lcd.setCursor(10, 1); // set the cursor to column 10, line 1

lcd.print("00h00m"); // current time reset on LCD
beep();

}

//TIMER CONTROL
swRightState=digitalRead(swRightPin); //Start button reading
if(swRightState==LOW){ //if active the timer is initialized
startMillis=millis(); //start milliseconds
setMillis=timeEncValue*60000; //set milliseconds
timerFlag=HIGH; //timer flag activation
knobChangeFlag=HIGH; //knob changing activation
beep();
}

if(remainingMinute==0&&swRightState==HIGH) timerFlag=LOW; //flag timer testing
digitalWrite(LED_BUILTIN, timerFlag); //timer flag led on
if(timerFlag==HIGH){
elapsedMillis=millis()-startMillis; //Time check
}

elapsedMinute=round(elapsedMillis/60000); //Conversions for the LCD
remainingMinute=timeEncValue-elapsedMinute;
currentHour=remainingMinute / 60;
currentMinute=remainingMinute % 60;

//HEATING CONTROL
if(sensTemp <= (setTemp - dT)&&timerFlag==HIGH)heatFlag=HIGH; //Heater ON
```





```
else if (sensTemp >= (setTemp + dT) || timerFlag == LOW) heatFlag = LOW;           //Heater OFF
digitalWrite(heatPin, heatFlag);                                               //Relay pin is active HIGH
//ALARM CONTROL
if (knobChangeFlag == HIGH && sensTemp > (setTemp - 0.1) && sensTemp < (setTemp + 0.1)) {
    beep2();
    knobChangeFlag = LOW;
}

//Serial monitor control (For debug only)
Serial.print("Inc. temp. = ");
Serial.print(tempEncValue);
Serial.print(", Minuti = ");
Serial.print(timeEncValue);
Serial.print("\t");
Serial.print(setHour);
Serial.print(":");
Serial.print(setMinute);
Serial.print("\t");
Serial.print("swR= ");
Serial.print(digitalRead(swRightPin));

Serial.print("\t Elapsed: ");
Serial.print(elapsedMinute);
Serial.print("\t Rem: ");
Serial.print(remainingMinute);
Serial.print("\t tF: ");
Serial.print(timerFlag);
Serial.print("\t kcF: ");
Serial.print(knobChangeFlag);
Serial.println();

// LCD CONTROL -----
//LCD Set Temperature
lcd.setCursor(0, 0);
if (setTemp < 100) lcd.print(0);
if (setTemp > maxTemp) lcd.clear();
lcd.print(setTemp, 1);
lcd.print("\xDF"C");
//LCD Current temperature
lcd.setCursor(0, 1);
if (sensTemp < 100) lcd.print(0);
lcd.print(sensTemp, 1);
lcd.print("\xDF"C");

// LCD Set Time
lcd.setCursor(10, 0);
if (setHour < 10) lcd.print(0);
lcd.print(setHour);
lcd.print("h");
if (setMinute < 10) lcd.print(0);
lcd.print(setMinute);
lcd.print("m");

//LCD Remaining Time
if (timerFlag == HIGH) { //Remaining time is updated only with timer active
    lcd.setCursor(10, 1);
    if (currentHour < 10) lcd.print(0);
    lcd.print(currentHour);
    lcd.print("h");
    if (currentMinute < 10) lcd.print(0);
    lcd.print(currentMinute);
    lcd.print("m");
}

//LCD Flags
lcd.setCursor(7, 1);
if (heatFlag == HIGH) lcd.print(" H "); //Heater flag "H"
else lcd.print(" ");

lcd.setCursor(7, 0);
if (timerFlag == HIGH) lcd.print(" T "); //Timer flag "T"
else lcd.print(" ");
}

//===== BUZZER FUNCTIONS =====
void beep() { //single beep
```

```
digitalWrite(buzzerPin,HIGH);  
delay(100);  
digitalWrite(buzzerPin,LOW);  
}  
void beep2(){ //double beep  
digitalWrite(buzzerPin,HIGH);  
delay(100);  
digitalWrite(buzzerPin,LOW);  
delay(200);  
digitalWrite(buzzerPin,HIGH);  
delay(100);  
digitalWrite(buzzerPin,LOW);  
}
```

Test



Fig. 5 – Test finale

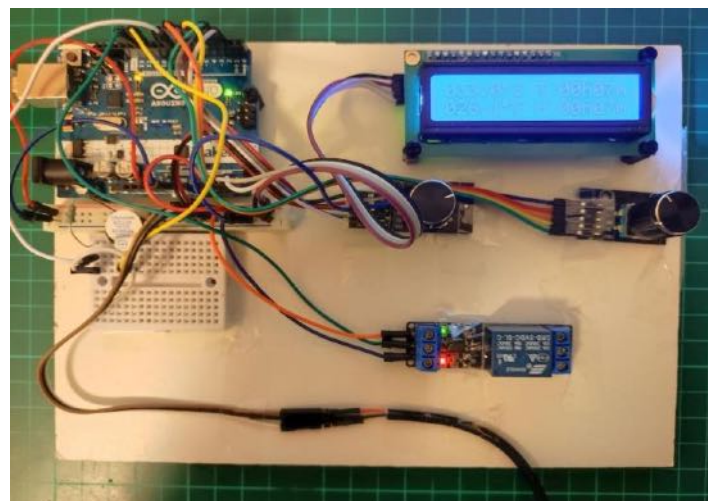


Fig. 6 - Primi test

Durante le prime fasi di sviluppo, il circuito è stato testato sezione per sezione per verificare se ognuna di esse funzionasse correttamente.

Successivamente, il sistema è stato testato nel suo insieme (eccetto le parti a tensione di rete) per verificare possibili interazioni indesiderate.

Alla fine, il prototipo finale è stato assemblato in un case e completamente testato, inclusa la sezione della tensione di rete. Per simulare l'elemento riscaldante è stata utilizzata una lampada a incandescenza.

Dopo tutti questi test, il dispositivo è stato utilizzato in uno scenario di vita reale per fare alcuni esperimenti in un laboratorio di biologia.

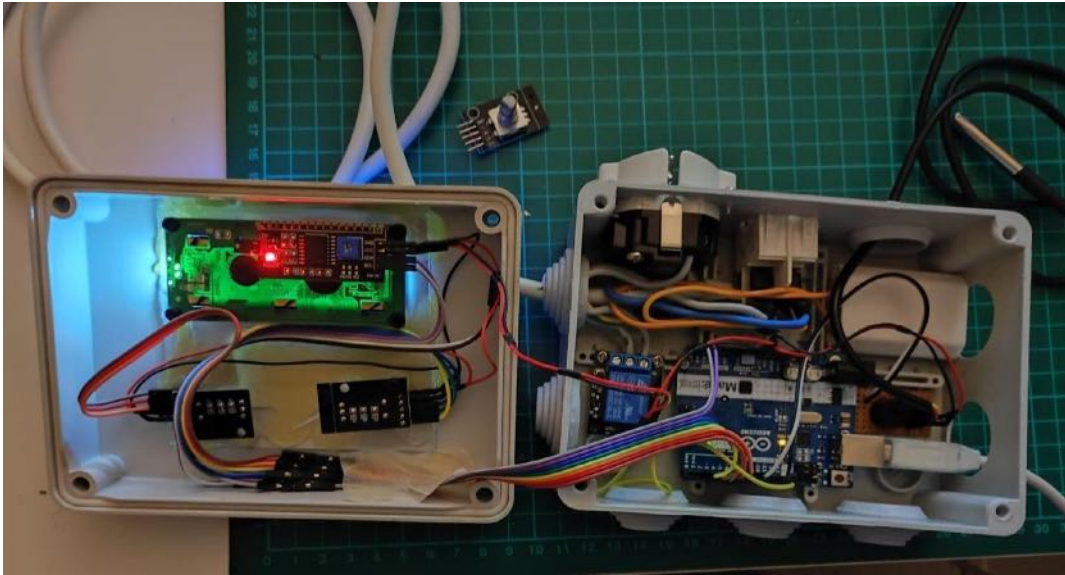


Fig. 7 - Prototipo finale quasi completo

Problemi e possibili miglioramenti

Durante le sperimentazioni in laboratorio con il vero riscaldatore è emerso il problema di un'inerzia termica superiore al previsto. Ciò ha comportato una fluttuazione della temperatura anche di pochi gradi e molto più alta del previsto.

Questo aspetto è un punto di partenza interessante per futuri miglioramenti:

Sostituzione del controllo ON-OFF con un più pronto e preciso controllo PID e conseguente sostituzione del tradizionale relè elettromeccanico con un relè allo stato solido in corrente alternata con rilevamento del passaggio per lo zero.

Sarà ovviamente necessario adattare la parte del programma relativa al controllo del riscaldatore.

Bibliografia

- <https://www.arduino.cc/>
- Dallas Semiconductors DS18B20 datasheet
- <https://www.az-delivery.de/it/collections/kostenlose-e-books>



VALUTAZIONE DELLA QUANTITA' DI FRAZIONE ORGANICA NEI SUOLI.

Il carbonio organico nel suolo rappresenta materiali di origine vegetale, animale e microbica che si trovano in vari stadi di decomposizione e sono associati alla frazione minerale. La frazione organica ha molte proprietà che influenzano il comportamento e la fertilità del suolo. Questo esperimento è condotto per determinare la frazione organica di tre diversi campioni: uno è un terreno fluviale, uno è un terreno da giardino e l'ultimo è un terreno boschivo.

Una bilancia da laboratorio e un riscaldatore vengono utilizzati insieme per condurre l'esperimento.

Ogni campione secco viene pesato prima dell'esperimento e viene riscaldato sul riscaldatore per 10 minuti. Dopo quel tempo è pesato. Questo processo viene ripetuto tre volte (la durata dell'esperimento per ogni campione è di trenta minuti).

Ci sono alcune differenze prima e dopo l'aggiornamento:

- Prima dell'aggiornamento non è possibile controllare la temperatura dei campioni;
- Prima dell'aggiornamento l'operatore deve rimanere in laboratorio e misurare il tempo, dopo l'aggiornamento è possibile l'impostazione e il riscaldatore si spegne automaticamente allo scadere del timer;
- Dopo l'aggiornamento il processo è automatizzato e i dati vengono raccolti e analizzati da Arduino.

[I dati dell'esperimento.](#)

CAMPIONE 1: terreno fluviale.

100 g vengono pesati sulla bilancia da laboratorio e si riscaldano in tre fasi. Non è possibile controllare la temperatura ma viene registrata a intervalli regolari.



Fig. 5-6: terreno fluviale pesato sulla bilancia (a sinistra) e riscaldato sul riscaldatore (a destra)

TEMPO (minuti)	PESO (g)	TEMPERATURA (°C)
10	99,4	110
20	99	130
30	99	174

CAMPIONE 2: terreno del giardino.

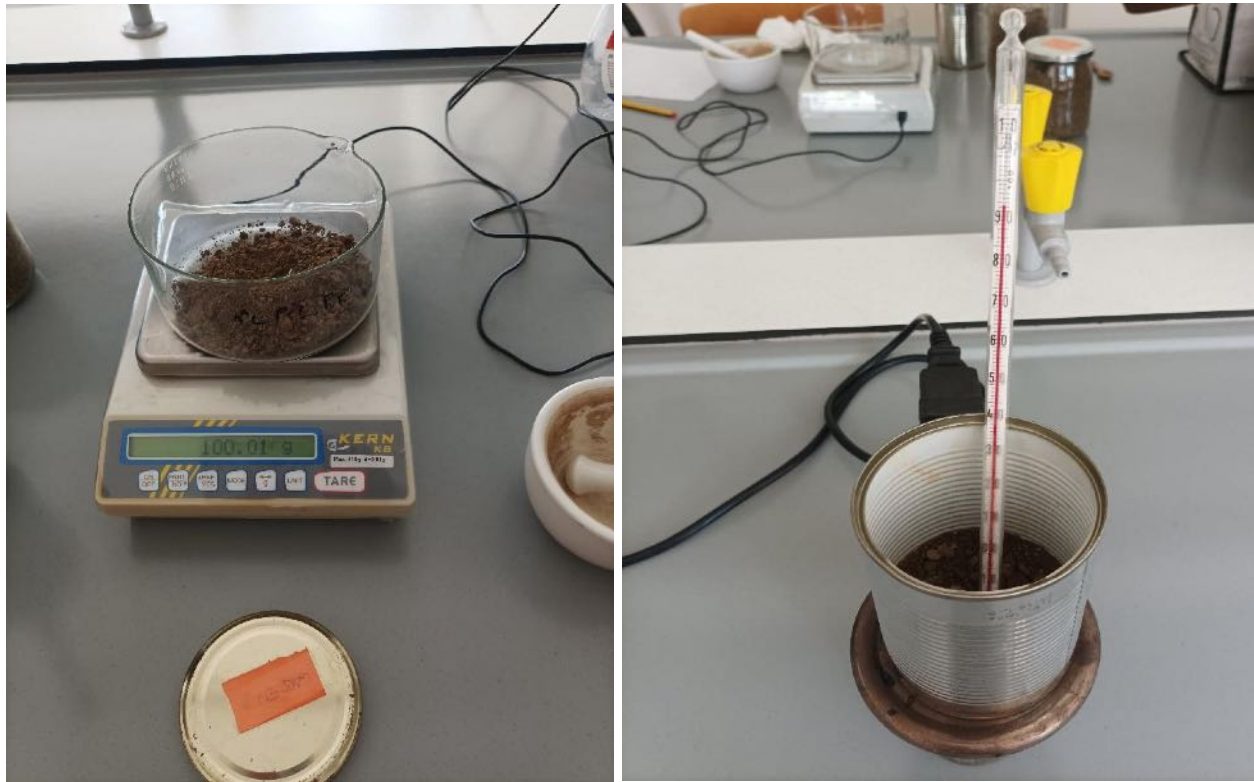


Fig. 7-8: terreno del giardino pesato sulla bilancia (a sinistra) e riscaldato sulla stufa (a destra)

TEMPO (minuti)	PESO (g)	TEMPERATURA (°C)
10	92,44	90
20	86,03	100
30	84,3	161

CAMPIONE 3: terriccio di sottobosco.

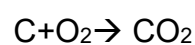


Fig. 9-10: terriccio di sottobosco pesato sulla bilancia (a sinistra) e riscaldato sul riscaldatore (a destra)

TIME (minutes)	WEIGHT (g)	TEMPERATURE (°C)
10	89,31	93
20	83,72	104
30	82,59	125

TEMPO (minuti)	PESO (g)	TEMPERATURA (°C)
10	89,31	93
20	83,72	104
30	82,59	125

La reazione chimica che porta alla perdita di peso è



perché la frazione organica viene bruciata durante il riscaldamento. Il calo ponderale rappresenta il tasso di fertilità del suolo, infatti viene classificato dalla percentuale di frazione organica in molto

scarso (<0,8%), scarso (0,8-1,2%), medio (1,2-2,0%), buono (2,0-4,0), molto buono (4,0-8,0%), molto ricco (>8%) (Arpav, Veneto Agricola, 2007).

I grafici successivi mostrano la perdita di peso nei tre campioni: il terreno fluviale ha perso solo 1 g, significa che ha una bassa frazione organica, infatti è composto principalmente da sabbia. Il terreno più ricco è quello boschivo, che ha perso 17,41 g in 30 minuti. Il terreno dell'orto ha anche un'importante frazione organica, infatti è occupato da un orto adibito alla coltivazione di ortaggi.

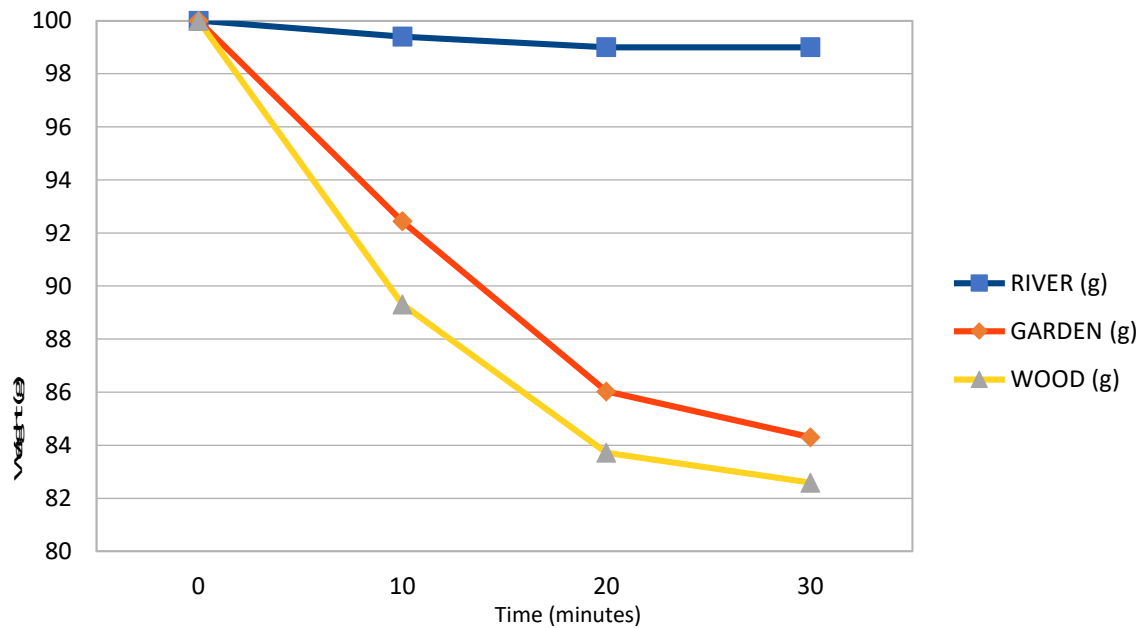


Grafico 1 : perdita di peso in 30 minuti

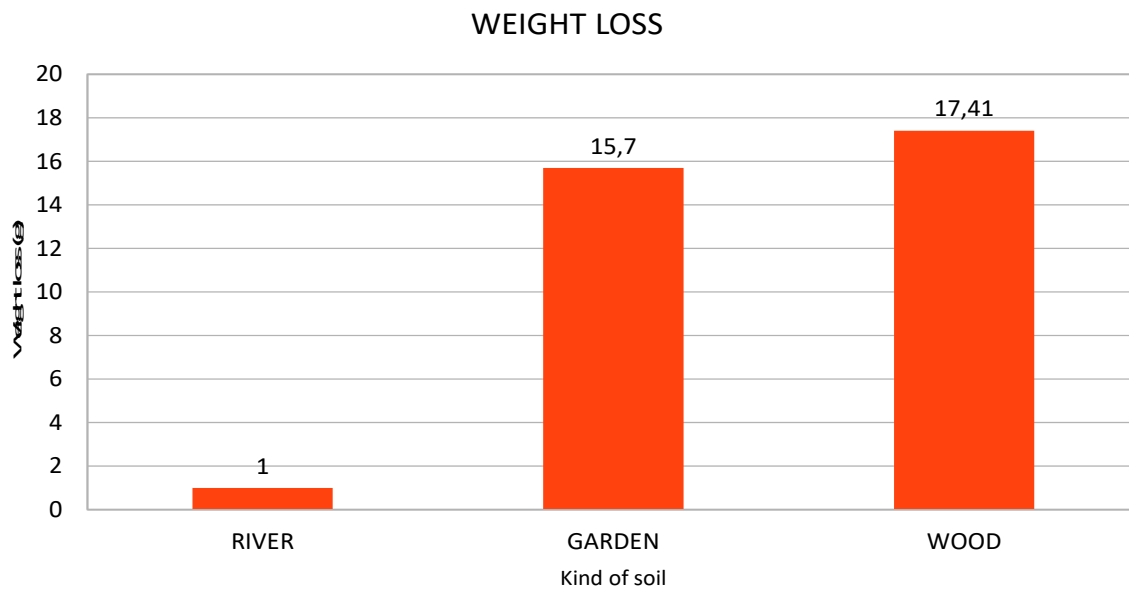


Grafico 2: differenza di perdita di peso

A seguire la valutazione della percentuale di frazione organica e la classificazione dei campioni.

	Peso Iniziale (g)	Perdita di peso (g)	Percentuale di frazione organica (%)
FIUME	100	1	1%
GIARDINO	100	15,7	15,70%
BOSCO	100	17,41	17,41%

Per quanto riguarda la classificazione Arpav e Veneto Agricola, i suoli fluviali sono poveri e molto ricchi sono i suoli di orti e boschi.

Dopo l'aggiornamento, i dati vengono nuovamente analizzati automaticamente con Arduino. L'operatore deve solo regolare la bilancia da laboratorio, mettere i materiali su di essa e impostare il riscaldatore. Il conteggio del tempo dei passi inizia dopo il raggiungimento della temperatura impostata (100°C).

I tre campioni di terreno vengono pesati sulla bilancia del laboratorio, Arduino acquisisce i dati. Quindi, l'operatore imposta la temperatura e il tempo dell'esperimento e i terreni si riscaldano. Il tempo, per il primo campione, è fissato a 12 minuti, per raggiungere la giusta temperatura e iniziare l'esperimento. Per le fasi successive il tempo è impostato a 10 minuti, perché la temperatura è già stata raggiunta. La temperatura è impostata a 100°C, perché per l'inerzia termica aumenta di circa 20°. Il riscaldatore è collegato ad Arduino, che è collegato alla corrente elettrica; un sensore di temperatura è immerso nel campione. Il dispositivo è molto semplice da usare: sulla sinistra del monitor è presente un pulsante per la regolazione della temperatura (l'intervallo è di 1°C). E' possibile vedere, nella parte superiore, la temperatura impostata e, nella



parte inferiore, la temperatura reale della sostanza. Nella parte destra del monitor è possibile vedere il tempo impostato (parte alta) e il tempo rimanente (parte bassa). L'intervallo è di 1 minuto. Dopo l'impostazione, è necessario premere il pulsante a destra per avviare l'esperimento. Quando viene raggiunta la temperatura, il dispositivo emette un suono. Al termine del processo lo spegnimento è automatico.

ESPERIMENTO RIPETUTO DOPO L'AGGIORNAMENTO.

CAMPIONE 1: terreno fluviale.

100 g vengono pesati sulla bilancia da laboratorio e si riscaldano in tre fasi. A differenza dell'esperimento condotto senza Arduino, è possibile controllare la temperatura, che arriva fino al massimo di 121°C. La perdita di peso è maggiore rispetto all'esperimento precedente, forse per una iniziale maggiore umidità del campione.



Fig. 11-12 : campione di terreno fluviale (sinistra) e terreno riscaldato sul riscaldatore collegato al dispositivo (destra)

TEMPO (minuti)	PESO (g)	TEMPERATURA (°C)
10	98,94	95
20	98,14	118
30	98,09	121

CAMPIONE 2: terreno da giardino.

Quell'esperimento mostra all'incirca lo stesso alleggerimento di quello manuale, la differenza è che la temperatura del riscaldatore, dopo l'aggiornamento, non supera mai i 120°C. In particolare,

la prima misurazione mostra una minore diminuzione del peso rispetto all'esperimento condotto senza Arduino, perché la temperatura sale solo fino a 90°C.



Fig. 13-14 : campione di terreno del giardino (a sinistra) e il terreno riscaldato sul riscaldatore collegato al dispositivo (a destra)

TEMPO (minuti)	PESO (g)	TEMPERATURA (°C)
10	97,13	92
20	87,89	112
30	84,07	120

CAMPIONE 3: terreno boschivo.

Anche per quel campione il dimagrimento è più lento rispetto all'esperimento manuale, per gli stessi motivi. Inoltre la diminuzione di peso è maggiore rispetto all'esperimento manuale, probabilmente perché in quella porzione di terreno c'era più frazione organica.



Fig. 15-16 : campione di terra di legno (a sinistra) e terra riscaldata sul riscaldatore collegato al dispositivo (a destra)

TEMPO (minuti)	PESO (g)	TEMPERATURA (°C)
10	89,11	93
20	84,24	104
30	76,64	125

Di seguito, i risultati presentati da Arduino e l'elaborazione dei dati raccolti.

Esperimento	Peso (g)	Sostanza	
Measure 26	100,01	Garden	1st measure
Measure 29	98,94	Sand	2nd Measure
Mesure 38	97,13	Garden	2nd Measure
Measure 41	98,14	Sand	3rd Measure
Measure 44	98,1	Sand	4th Measure
Measure 47	87,89	Garden	3rd Measure
Measure 50	100,07	Wood	1st measure
Measure 53	84,07	Garden	4th Measure
Measure 59	89,11	Wood	2nd Measure
Measure 62	84,24	Wood	4th Measure
Measure 67	76.64	Wood	5th Measure



Grafico 3: perdita di peso in 30 minuti (dopo l'aggiornamento)

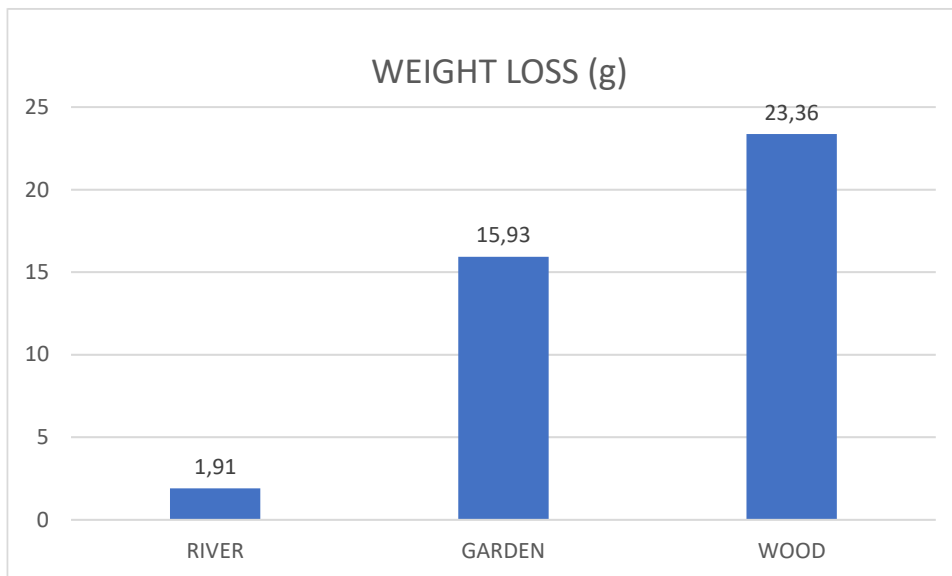


Grafico 4: differenze della perdita di peso (dopo l'aggiornamento)

I dati vengono confrontati per determinare il terreno più fertile, in base alla percentuale di carbonio nel suolo. Nonostante alcune piccole differenze, i risultati sono simili all'esperimento manuale: il terreno boschivo è il più fertile.

	Peso iniziale(g)	Perdita di peso (g)	Percentuale di frazione organica(%)
FIUME	100	1,91	1,91%
GIARDINO	100	15,93	15,93%



BOSCO	100	17,41	23,36%
-------	-----	-------	--------

CRITICITÀ.

Nei materiali granulari è difficile ottenere una temperatura omogenea; l'operatore deve mescolarlo spesso. Per questo motivo, il sensore di temperatura non è preciso in questo esperimento. Inoltre, il riscaldatore è molto invecchiato, quindi l'inerzia termica aumenta la temperatura più del necessario. L'operatore deve regolarlo di circa 20° in meno, perché l'aumento è direttamente proporzionale alla temperatura impostata.

Per concludere, l'esperimento condotto con Arduino è più semplice da condurre, perché dopo le impostazioni l'operatore deve solo pesare i campioni al momento opportuno. Inoltre, Arduino spegne il riscaldatore dopo il tempo impostato ed emette un suono, quindi l'operatore non deve cronometrare l'esperimento e si evitano errori e imprecisioni.

PREPARIAMO DELLO YOGURT...

Questo esperimento non è possibile prima dell'aggiornamento, perché non è possibile regolare la temperatura del riscaldatore. Dopo l'aggiornamento, l'operatore può impostare la giusta temperatura (in questo caso è 42°C), perché un termometro è collegato al riscaldatore e mantiene il giusto valore per il tempo necessario. Il range di temperatura è di 1,0°C.

Questo aggiornamento è molto importante perché amplia il campo di applicazione del dispositivo, ad esempio è possibile applicarlo in un'industria casearia. Per questo utilizzo è stato preparato lo yogurt.

Lo yogurt deriva dalla fermentazione del latte operata dai batteri sul lattosio (C₁₂H₂₂O₁₁), che si suddivide in galattosio e glucosio. Al termine di questo processo si produce acido lattico (C₃H₆O₃) e il pH scende da 6,7 (pH del latte vaccino) a 4,5. Il gusto caratteristico dello yogurt è dato dall'acido lattico. Allo stesso tempo, la caseina (proteina del latte) si lega ad alcune altre sostanze e conferisce allo yogurt la tipica consistenza. Quando il pH è intorno a 4,5 lo yogurt è pronto: è possibile conservarlo in frigorifero per 30 giorni.

DESCRIZIONE DELL'ESPERIMENTO.

I fermenti lattici sono mescolati in un litro di latte intero. Si mette il becker sul fornello e si porta la temperatura a 43°C per sette ore e quindici minuti (i minuti in più servono per il riscaldamento del latte alla giusta temperatura).



Figure 17: fermenti lattici



Figure 18: inizio dell'esperimento- fermenti lattici in un becker grande



Figure 19: I fermenti e il becker utilizzati nel nostro esperimento

Il riscaldatore è collegato ad Arduino, che è collegato alla corrente elettrica; un sensore per la temperatura è immerso nel latte. Lo strumento è molto semplice da utilizzare: sulla sinistra del monitor è presente un pulsante per la regolazione della temperatura (l'intervallo è di 1°C). E' possibile vedere, nella parte superiore, la temperatura impostata e, nella parte inferiore, la temperatura reale della sostanza. Nella parte destra del monitor è possibile vedere il tempo impostato (parte alta) e il tempo rimanente (parte bassa). L'intervallo è di 1 minuto. Dopo l'impostazione, è necessario premere il pulsante giusto per iniziare il lavoro. Quando viene raggiunta la temperatura, lo strumento suona. Al termine del processo lo spegnimento è automatico.



Fig. 20: strumento usato per condurre l'esperimento



Fig. 21: display dello strumento con le relative impostazioni



Ffig. 22: strumentazione al lavoro



CRITICITA'

Quando viene raggiunta la temperatura Arduino spegne il riscaldatore, ma la temperatura aumenta comunque per l'inerzia termica dello stesso. Per risolvere questo problema è necessario impostare la temperatura 5 o 6 gradi più bassa di quella giusta.

Bibliografia.

Tabelle allegate:

<http://www.pasquali.org/dispense/Tabella%20densita%20dei%20materiali.pdf>
Arpav, Veneto Agricoltura (2007), L'interpretazione delle analisi del terreno – Strumento per la sostenibilità ambientale, 74 pp.

IV. Lettore di misure universale

Dal display numerico all'informazione digitale.

Introduzione

Nell'ambito del recupero di vecchi macchinari, potrebbe essere utile leggere le misure fornite dai vecchi dispositivi e rendere disponibili questi dati ad altre piattaforme o sistemi software, ad esempio:

- memorizzazione dei dati nel cloud
- esportazione dei dati in file Excel o CSV
- generazione automatica di grafici o diagrammi

Questo capitolo descrive la progettazione e l'implementazione di un Universal Measure Reader (UMR), ovvero un componente in grado di leggere le misure da diversi dispositivi digitali analizzando le immagini catturate dai display di output e applicando tecniche di riconoscimento ottico dei caratteri (Figura 1).



Per costruire un UMR, sono necessari due componenti:

- Hardware per l'acquisizione delle immagini: può essere una scheda Arduino o Arduino compatibile
- Software per l'analisi dell'immagine e l'estrazione, l'elaborazione, l'archiviazione e la presentazione dei dati
- (Facoltativo) Hardware per fornire feedback:
 - Un led verde acceso se la misura è nel range previsto
 - Un cicalino per la notifica di errori o problem
 - Un interruttore per spegnere o accendere un dispositivo secondo una regola basata sulla misura acquisita..

Ci sono diverse scelte possibili sia per i componenti hardware che per quelli software, in questo capitolo considereremo alcuni dei più adatti.

Struttura del capitolo

Questo capitolo è composto dalle seguenti sezioni:

- Hardware: comprende una breve presentazione dei componenti hardware utilizzati nei progetti proposti.
- Architettura di sistema: include alcuni esempi di architetture di sistema per l'implementazione di un lettore di misure universale.
- Librerie software: include una descrizione delle librerie software utilizzabili per l'OCR.
- Progetti
 - Progetto 1: cattura una misura ogni 30 secondi
 - Progetto 2: acquisire una misura alla pressione di un pulsante



- Progetto 3: Accendere un led verde su misura corretta
- Progetto 4: inviare le misure al cloud

- Bibliografia

Hardware

Tra le numerose schede disponibili sul mercato, che possono catturare un'immagine, abbiamo selezionato per i nostri progetti la scheda ESP32-CAM, perché i seguenti motivi principali:

- È completamente programmabile tramite Arduino IDE, quindi può essere sfruttato gli stessi strumenti e conoscenze utilizzate/acquisite per gli altri progetti descritti in questo documento.
- È molto economico: in genere costa meno di 10 €. È importante in relazione al rinnovamento dei vecchi macchinari mantenere bassi i costi per mantenere il progetto redditizio oltre il rispetto dell'ambiente.
- È abbastanza facile da configurare: nel nostro progetto siamo partiti dagli esempi di base forniti con le librerie Arduino e abbiamo aggiunto ulteriori caratteristiche e funzionalità.

Nelle sezioni seguenti vengono descritte le schede ESP32-CAM, un confronto tra i modelli disponibili sul mercato, la descrizione della nostra scheda preferita e due tutorial sulla configurazione dell'IDE di Arduino e sul setup di base della ESP32-CAM.

ESP32-CAM





L'ESP32-CAM è una scheda che include:

- Chip ESP32-S: è un microcontrollore dual-core che combina funzionalità wireless Bluetooth e Wi-Fi.
- Fotocamera OV2640: è un sensore di immagine rilasciato nel 2005 e sebbene sia una tecnologia vecchia di 15 anni è uno dei sensori più utilizzati.
- Slot per scheda MicroSD: supporta schede microSD fino a 4 GB.
- 10 GPIO accessibili: solo pochi GPIO possono essere effettivamente utilizzati perché la maggior parte viene utilizzata per collegare la fotocamera o il lettore di schede MicroSD. Per questo motivo, in alcuni progetti, abbiamo la necessità di affiancare a questa scheda una scheda Arduino.
- 4 MB di PSRAM: viene utilizzato per il buffering delle immagini dalla telecamera.
- Pulsante Reset: viene utilizzato per riavviare la scheda.
- Flashlight LED: è un LED molto luminoso che può essere utilizzato come torcia.

Alcune delle schede ESP32-CAM non includono un'interfaccia USB-UART, quindi non possono essere collegate direttamente a un PC per essere programmate e richiedono un programmatore FTDI (da TTL a USB) o un programmatore ESP32-CAM MB.

Confronto ESP32-CAM

Sul mercato sono disponibili diverse schede ESP32-CAM, in (1) è riportato un articolo dettagliato con un confronto tra sette diverse schede. Riportiamo di seguito una sintesi del confronto tra quattro diverse schede.

Modello	AI-THINKER	ESP-EYE	T-JOURNAL	TTGO T-CAMERA
				
PSRAM	4 MB	8 MB	X	8MB
SCREEN	X	X	0.91" OLED	0.96" OLED
MicroSD Card	✓	X	X	X
Microphone	X	✓	X	✓
Battery Connector	X	X	✓	✓
Built-in Programmer	X	✓	✓	✓
GPIO	10	X	4	X
Detailed Review	(2)	(3)	(4)	(5)
Cost Range	7 – 14 €	30 – 40 €	12 – 20 €	20 – 30 €

AI-THINKER

Nei progetti riportati in questo capitolo abbiamo utilizzato la scheda AI-THINKER perché ha un ottimo rapporto qualità prezzo:

- È la scheda più economica
- Ha molti più GPIO disponibili rispetto alle altre schede
- Ha un modulo PSRAM

L'unico inconveniente che abbiamo riscontrato è la mancanza di un'interfaccia USB-UART e quindi è necessario un programmatore esterno per caricare il codice.

Suggeriamo di abbinare la scheda AI-Thinker al programmatore ESP32-CAM MB (Figura 2): tramite questo programmatore è facile sia caricare il codice sull'ESP32-CAM sia fornire l'alimentazione. Il programmatore include anche un utile pulsante di reset.

Tipicamente l'ESP32-CAM MB viene venduto in combinazione con l'AI-THIKER, altrimenti è disponibile a meno di 5€.

Una descrizione dettagliata di come caricare il codice su ESP32-CAM con il programmatore ESP32-CAM MB è fornita in (6)



Figure 2 ESP32-CAM MB programmer

PINOUT

Nella Figura 3 è mostrato il PINOUT dei pin disponibili della AI-THINKER BOARD. Esistono altri GPIOI che sono collegati direttamente con la fotocamera e con il lettore di schede SD.

- Se **IO0** è collegato a **GND** l'ESP32 è impostato in modalità lampeggiante ed è possibile caricare il codice..
- **IO1** e **IO3** vengono utilizzati per la trasmissione seriale (TX e RX)
- La scheda supporta una tensione di ingresso di 5 V o 3,3 V e può fornire in uscita la stessa tensione

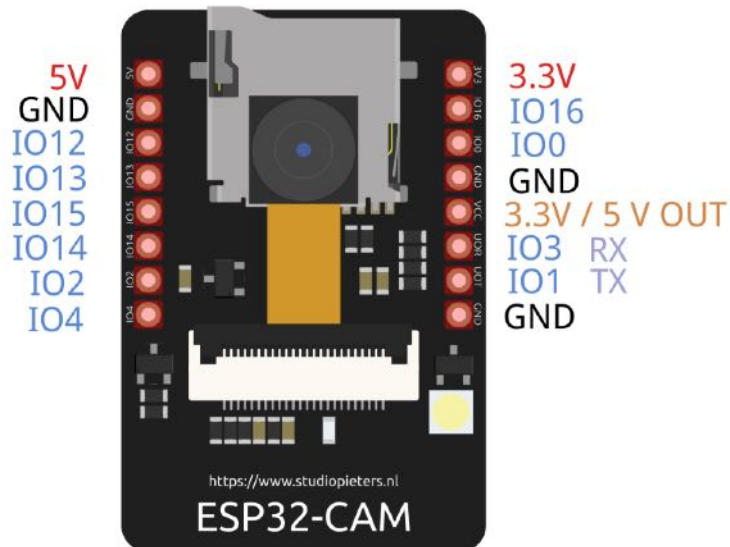




Figure 3 ESP32-CAM pinout

Tutorial 1: Aggiunta della libreria ESP32 all'IDE di Arduino

DIFFICULTY	Easy Medium Hard 
SOFTWARE REQUIREMENTS	Arduino IDE installed on Windows, Linux or macOS
HARDWARE REQUIREMENTS	None: in this phase it is not necessary to connect the ESP32-CAM board
TIME	 about 5 minutes

La scheda ESP32-CAM è completamente compatibile con l'IDE Arduino (7) e per includere le librerie EPS32 e l'esempio è necessario eseguire i seguenti passaggi:

1. Accedi alle preferenze dell'IDE di Arduino tramite il menu: File Preferenze

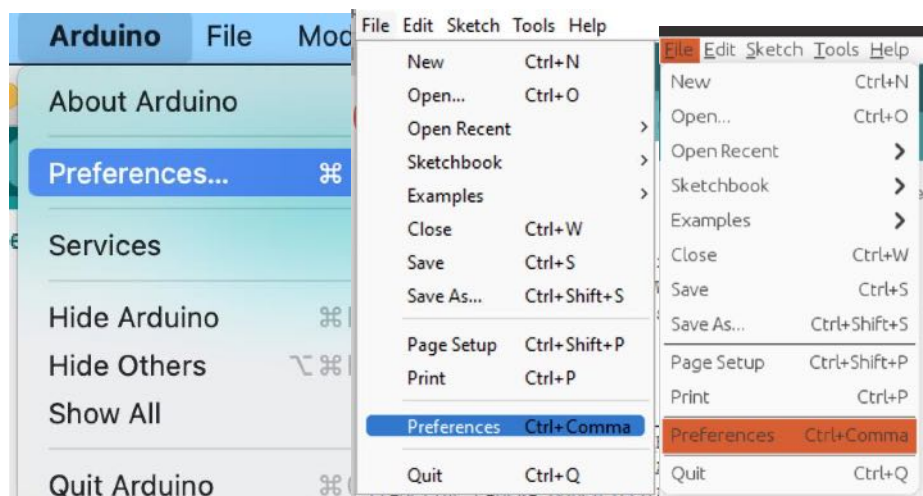


Figure 4 Arduino Preferences on A) macOS B) Windows C) Linux (Ubuntu 20.04)¹

2. Aggiungi l'URL https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_index.json alla sezione URL Gestore schede aggiuntive della finestra di dialogo delle preferenze.

¹ Since the Arduino IDE presents a user interface very similar among the 3 different operative systems, hereafter we include only one screenshot relative to Linux or Windows.

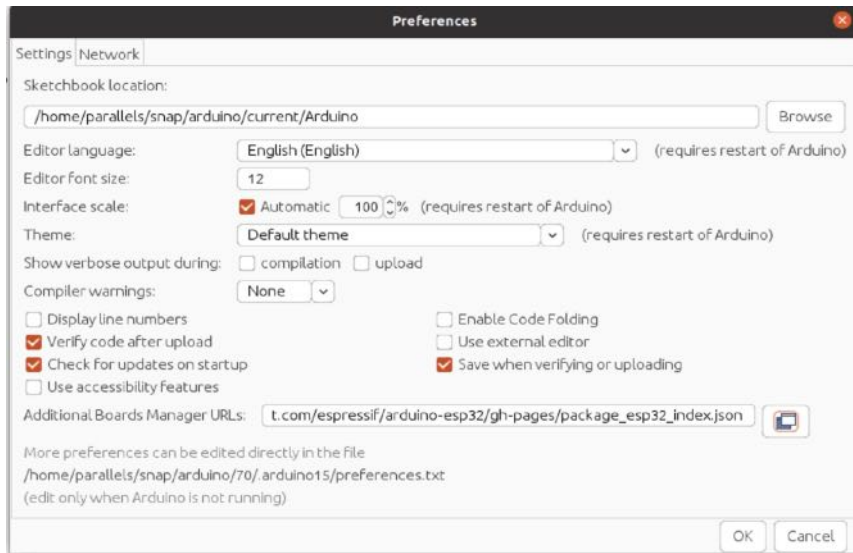


Figure 5 Additional Boards Manager URLs configuration

3. Apri il Gestore delle bacheche tramite il menu Strumenti Bachecca Gestore delle bacheche.

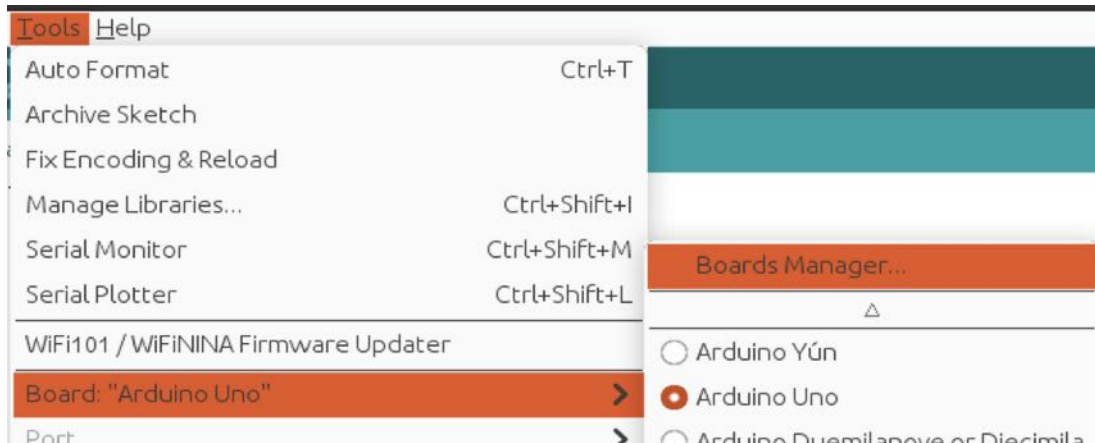


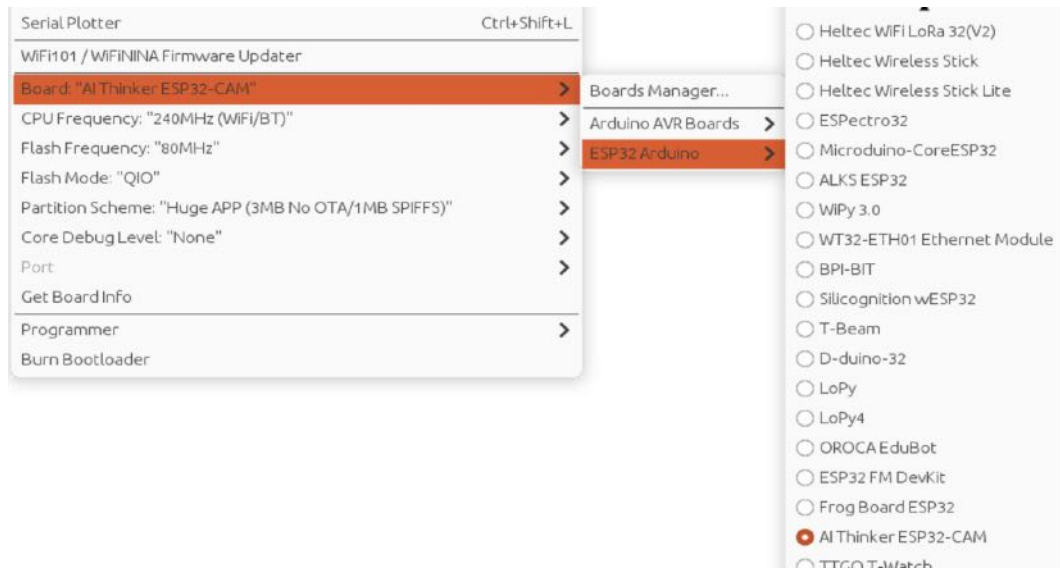
Figure 6 Boards Manager Menu

4. Cerca e installa il pacchetto esp32 di **Espressif Systems**.





Figure 7 esp32 package

5. Seleziona la tua scheda dal menu: Scheda ESP32 Arduino. Nel nostro caso abbiamo selezionato AI Thinker ESP32-CAM

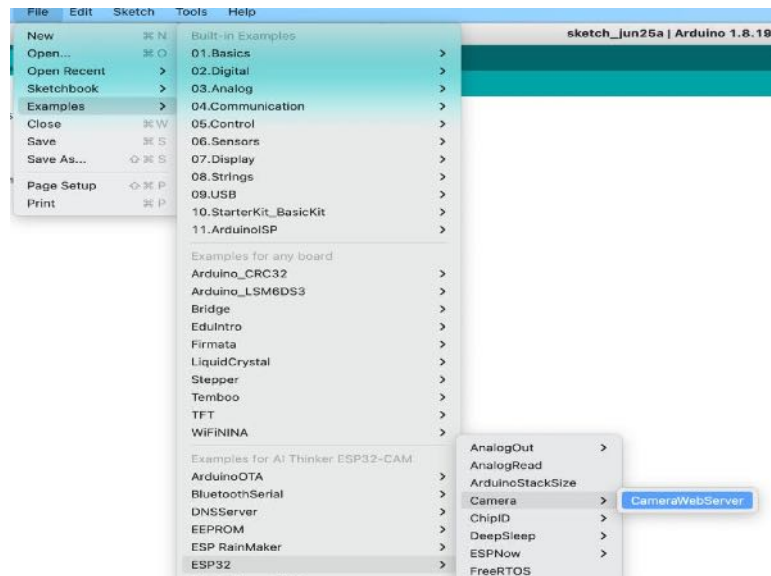


Tutorial 2: Configurazione della scheda ESP32-CAM

DIFFICULTY	Easy Medium Hard 
SOFTWARE REQUIREMENTS	<ul style="list-style-type: none"> • Arduino IDE installed on Windows, Linux or macOS • ESP32 library installed on Arduino IDE (see Tutorial 1: Adding the ESP32 library to Arduino IDE)
HARDWARE REQUIREMENTS	<ul style="list-style-type: none"> • ESP32-CAM board • Programmer e.g., ESP32-CAM MB
TIME	 about 10 minutes

Lo scopo di questo tutorial è mostrare come configurare la scheda ESP32-CAM partendo dall'esempio di lavoro incluso nel pacchetto Espressif (vedi Tutorial 1: Adding the ESP32 library to Arduino IDE)

6. Una volta selezionata la tua ESP32-CAM Board tramite il menu Tools Board Boards Manager, puoi aprire l'esempio CameraWebServer tramite il menu File Examples ESP32 Camera.



7. L'esempio CameraWebServer crea un semplice servizio web che permette di interagire con la telecamera. Lo schizzo è composto da 4 file:
- **CameraWebServer.ino:** è il file principale di Arduino che contiene la configurazione di base per selezionare il modello di scheda corretto e per inizializzare la connessione Wi-Fi. Questo codice avvia un server Web che ospita una semplice pagina Web che consente di acquisire immagini o avviare un flusso video.
 - **camera_pins.h:** definisce in base al modello di scheda selezionato la mappatura del pinout. Questo file è molto utile come riferimento GPIO per altri schizzi di Arduino che utilizzano ESP32-CAM.
 - **camera_index.h:** inizializza la variabile `index_ov2640_html_gz` con i byte della pagina compressa gzip che verrà visualizzata dal servizio web. È possibile caricare una pagina html personalizzata comprimendo il file html con gzip (Dynamic Huffman Coding), suddividendolo in byte e convertendo in hex array. Questo processo è descritto in dettaglio in (8).
 - **app_httpd.cpp:** contiene l'implementazione del web service. Se è necessario aggiungere alcuni gestori al servizio Web, è necessario modificare questo file. Il codice è piuttosto complesso, quindi per comprendere tutte le funzioni è richiesta un'ottima conoscenza della codifica in c o c++. Tuttavia, è possibile utilizzare la maggior parte delle funzioni così come sono e apportare solo lievi modifiche. Seguiremo questo approccio nei progetti descritti nelle prossime sezioni.
8. Per impostare il modello di scheda ESP32-CAM, è necessario rimuovere il commento dalla riga di definizione nel file CameraWebServer.ino. Nel nostro caso, abbiamo decommentato la definizione di `CAEMRA_MODEL_AI_THINKER`. Questa definizione viene utilizzata per impostare correttamente la mappatura dei GPIO e per l'abilitazione.



```
// =====  
// Select camera model  
// =====  
//#define CAMERA_MODEL_WROVER_KIT // Has PSRAM  
//#define CAMERA_MODEL_ESP_EYE // Has PSRAM  
//#define CAMERA_MODEL_ESP32S3_EYE // Has PSRAM  
//#define CAMERA_MODEL_M5STACK_PSRAM // Has PSRAM  
//#define CAMERA_MODEL_M5STACK_V2_PSRAM // M5Camera version B Has PSRAM  
//#define CAMERA_MODEL_M5STACK_WIDE // Has PSRAM  
//#define CAMERA_MODEL_M5STACK_ESP32CAM // No PSRAM  
//#define CAMERA_MODEL_M5STACK_UNITCAM // No PSRAM  
#define CAMERA_MODEL_AI_THINKER // Has PSRAM  
//#define CAMERA_MODEL_TTGO_T_JOURNAL // No PSRAM
```

9. La scheda ESP32-CAM dispone di connettività wireless nativa: sono disponibili sia funzionalità WI-FI che Bluetooth. E' necessario configurare sia l'ssid che la password per connettere la scheda alla rete aziendale/scuola. È necessario modificare le seguenti righe

```
// =====  
// Enter your WiFi credentials  
// =====  
const char* ssid = "*****";  
const char* password = "*****";
```

10. È necessario verificare e caricare il codice modificato sulla scheda ESP32-CAM. Devi selezionare la porta seriale corretta dal menu Strumenti Porta.

11. Se la configurazione è OK, si può leggere sul monitor seriale un'uscita simile alla figura sottostante.

```
rst:0x8 (TG1WDT_SYS_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT)  
configsip: 0, SPIWP:0xee  
clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00  
mode:DIO, clock div:1  
load:0x3fff0030,len:1344  
load:0x40078000,len:13516  
load:0x40080400,len:3604  
entry 0x400805f0
```

```
.  
WiFi connected  
Camera Ready! Use 'http://192.168.1.4' to connect
```

12. Il server web avviato è disponibile all'indirizzo visualizzato nel Monitor seriale solo perché le seguenti righe di codice:

```
Serial.println("");  
Serial.println("WiFi connected");  
  
startCameraServer();  
  
Serial.print("Camera Ready! Use 'http://");  
Serial.print(WiFi.localIP());  
Serial.println("' to connect");
```

13. Verrà visualizzata la pagina web rappresentata in:

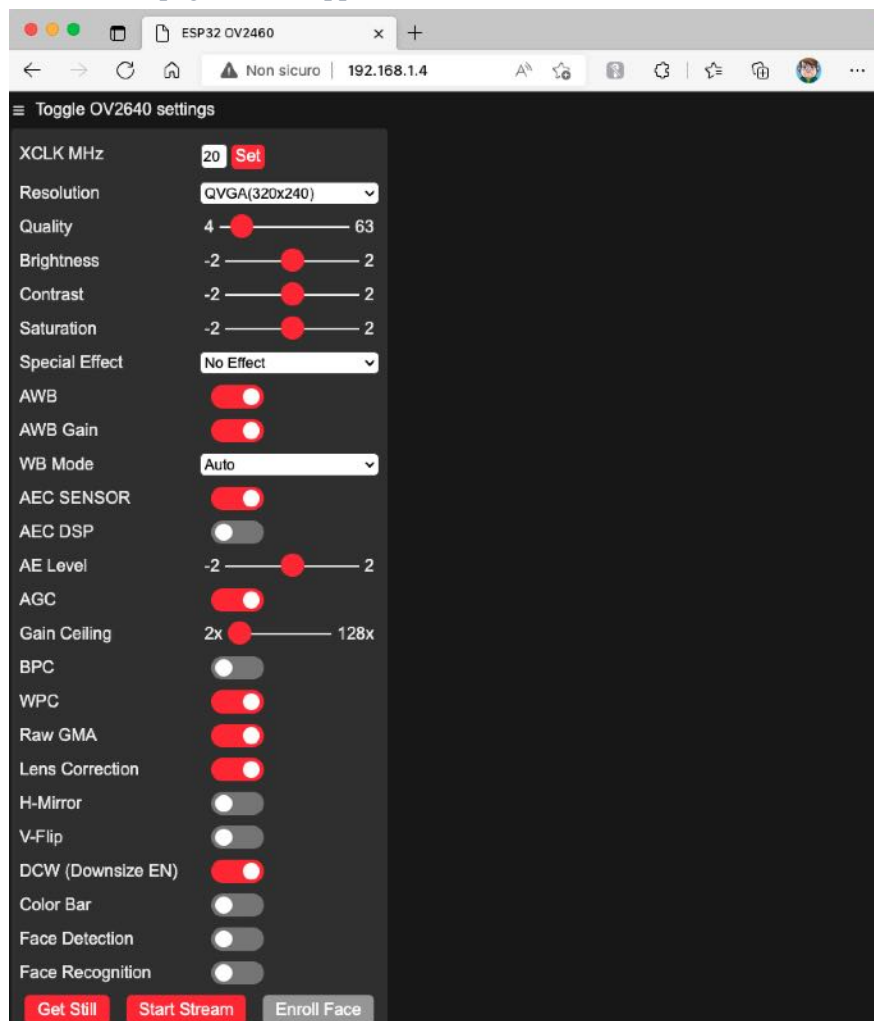


Figure 8 ESP32-CAM configuration web page

La pagina web riassume tutte le funzionalità dell'ESP32-CAM:

- Cattura un'immagine
- Avviare uno streaming video
- Modificare la risoluzione dell'immagine
- Impostare la qualità dell'immagine, la luminosità e il contrasto
- Capovolgi e specchia l'immagine
- Attivare il rilevamento e il riconoscimento del volto

Nella nostra esperienza, anche se le funzionalità di rilevamento e riconoscimento dei volti incorporate sembrano funzionare, non sono affidabili ai fini della sicurezza dell'accesso. In altre parole, non suggeriremo di sfruttare queste capacità per concedere l'accesso a sistemi o ambienti.

Nella Figura 9 è mostrato un esempio di streaming con la massima risoluzione.

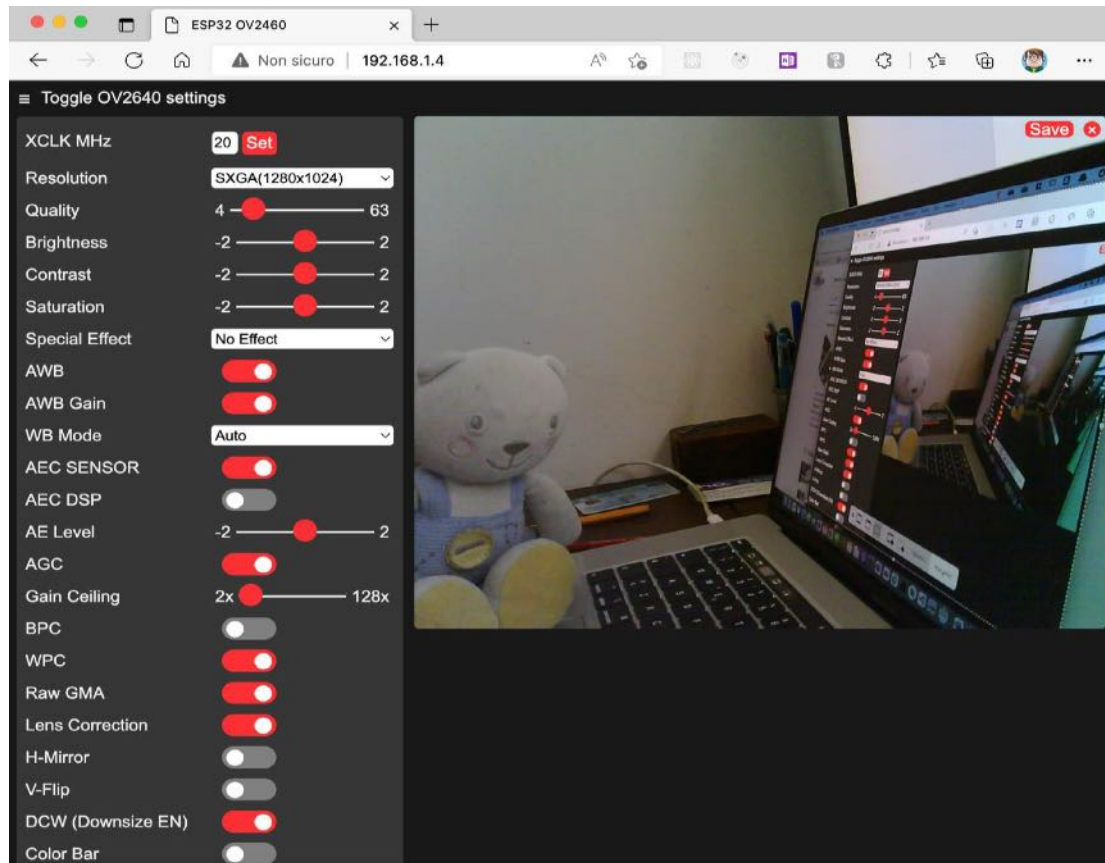


Figure 9 ESP32-CAM Streaming Video con massima risoluzione

Architetture di sistema

L'Universal Measure Reader (UMR) proposto è composto sia da componenti hardware che software, ed è importante progettare quali sovrintendono all'implementazione di specifiche funzionalità.

L'UMR fornisce le seguenti funzionalità principali:

- Scatta una foto del display del dispositivo
- Analizzare l'immagine e "comprendere" la misura tramite tecniche OCR
- Archiviare i dati e fornirli all'utente
 - o Display the date in a web application
 - o Export the data in Excel or CSV file
- Visualizzare la data in un'applicazione web per esportare i dati in file Excel o CSV.

Per raggiungere questi obiettivi esistono diverse possibili architetture di sistema che possono essere scelte caso per caso in relazione al contesto e al tipo di vecchio dispositivo che si vorrebbe modernizzare.

Si noti che le immagini inserite nelle sezioni successive rappresentano un altissimo livello di astrazione, mentre schemi tecnici più dettagliati verranno forniti nella sezione Progetti.



Componenti

L'UMR può essere implementato utilizzando i componenti descritti nella tabella seguente:

Componente	Tipo	Descrizione
ESP32-CAM	Hardware	<p>È descritto in dettaglio nella precedente sezione ESP32-CAM (pagina 27).</p> <p>Connettività senza fili</p> <p>Telecamera a bassa risoluzione</p> <p>GPIO disponibile per input (ad es. pulsanti) e output (ad es. LED)</p> <p>Riconoscimento facciale (non utilizzato)</p>
OCR server	Software	<p>Si tratta di un server web personalizzato che riceve le immagini e fornisce la misura "understood". Sfrutta la libreria software descritta nella sezione SOFTWARE.</p>
OCR cloud service	Software	<p>È un'alternativa al suddetto server OCR. Quando gli strumenti open source sfruttati dal server personalizzato OCR non sono sufficientemente accurati per "capire" la misura, è possibile utilizzare un servizio cloud OCR.</p>
Arduino boards	Hardware	<p>Sè gli ESP32-CAM GPIO non sono sufficienti, è possibile affiancare anche una scheda Arduino, che può:</p> <ul style="list-style-type: none">- guidare output complessi come display lcd, led, buzzers- inviare informazioni all'Arduino Cloud Service- gestire la configurazione ESP32-CAM tramite ingressi: es. combinazione di pulsanti e potenziometri
Arduino Cloud Service	Software	<p>È un servizio cloud per l'archiviazione dei dati del dispositivo Arduino IOT e la visualizzazione di dashboard accattivanti.</p>
Pulsanti, Led, fili, Resistenze, LCD, potenziometri, ecc.	Hardware	<p>Sono utilizzati in combinazione con schede ESP32-CAM o Arduino per fornire ingressi e uscite.</p>



Architettura 1: Software First System (SFS)

In questa architettura il componente principale di Universal Measure Reader è il modulo software che svolge la maggior parte delle funzionalità. Come illustrato nella Figura 10, il modulo software ottiene l'immagine della misura dalle schede ESP32-CAM, analizza l'immagine e fornisce i risultati all'utente attraverso una pagina web.

Questo design riduce al minimo l'implementazione hardware: in pratica, la scheda ESP32-CAM viene utilizzata solo per fornire immagini come se fosse una fotocamera esterna, perché la maggior parte del lavoro viene svolto dal software del server OCR.

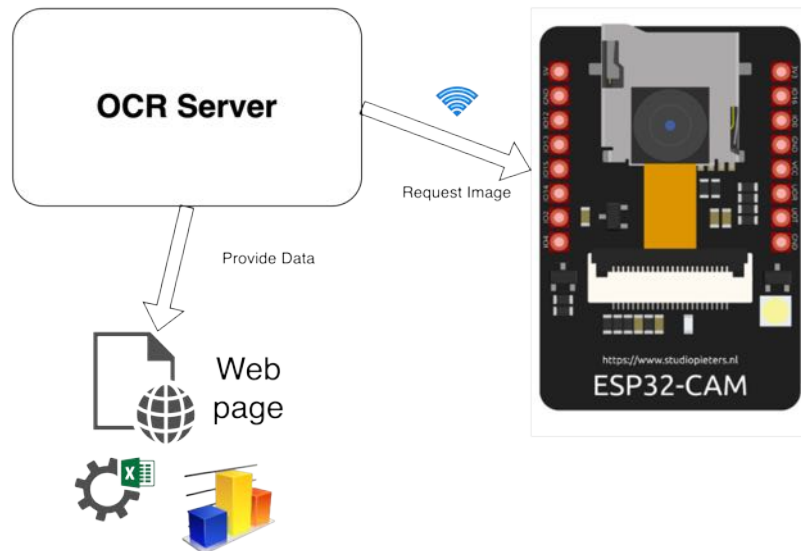


Figura 10 SFS Architecture

Questo tipo di architettura è sfruttato dal Progetto 1: cattura una misura ogni 30 secondi.

Architettura 2: Hardware First System (HFS)

L'Hardware First System Architecture privilegia i moduli hardware invece di quelli software. La Figura 11 HFS Architecture mostra un esempio di architettura HTS, dove:

- 1) L'utente preme un pulsante
- 2) La scheda ESP32-CAM reagisce all'evento di pressione del pulsante, quindi acquisisce un'immagine e la invia al server OCR
- 3) Il Server OCR analizza l'immagine e fornisce la misura riconosciuta ad una scheda Arduino
- 4) La scheda Arduino riceve la misura e la notifica come output (display lcd, led)

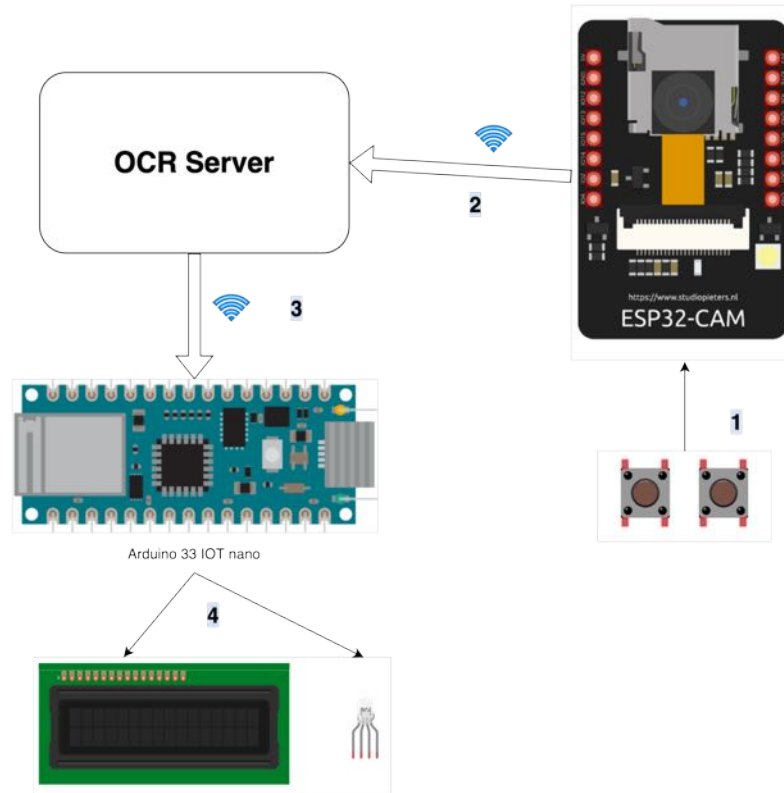


Figura 11 HFS Architecture

In questo schema il Server OCR è più leggero e svolge due funzionalità principali:

- Analisi dell'immagine
- Instradamento dell'output su un'altra scheda

La seconda scheda è necessaria perché sfortunatamente la scheda EPS32-CAM ha pochi GPIO disponibili e quindi non è possibile pilotare l'output direttamente da essa.



Architettura 3: Cloud based System (CLS)

Al giorno d'oggi, la capacità computazionale viene spostata sempre più dalle risorse hardware locali a enormi servizi di provider esterni, comunemente denominati servizi Cloud.

Come descritto nella sezione SOFTWARE, esistono diversi fornitori di servizi cloud e la maggior parte di essi offre servizi AI OCR.

Esistono almeno due possibili architetture di sistema che coinvolgono i servizi cloud in UMR:

- Solo Cloud: il servizio OCR personalizzato è completamente sostituito dai servizi cloud
- Cloud con Middleware: il servizio custom delega alcune funzionalità ai servizi Cloud ma è l'orchestratore del sistema.

Solo Cloud

Come illustrato nella Figura 12, in questa architettura proposta l'ESP32-CAM sfrutta direttamente i Servizi Cloud.

È fattibile ma non facile programmare la ESP32-BOARD per stabilire una connessione con il servizio Cloud selezionato. Dipende da provider a provider e dalle API che offrono, ma in genere la scheda può sfruttare sia API Javascript che servizi REST.

L'API Javascript utilizza la capacità computazionale del client (ovvero il computer connesso al servizio ESP32-BOARD), mentre il servizio REST utilizza la capacità (bassa) della scheda.

Anche l'altra scheda (nel nostro esempio una Arduino 33 IOT) deve essere connessa al servizio Cloud per essere controllata.

Un esempio di interazione è composto dai seguenti passi:

- 1) L'utente preme il pulsante
- 2) La scheda ESP32-CAM
 - a. Cattura un'immagine
 - b. Contatta il servizio Cloud e invia la Foto
- 3) Il servizio cloud
 - a. esegue l'OCR
 - b. notifica il provvedimento alla scheda Arduino
- 4) La scheda Arduino
 - a. Riceve la misura
 - b. Visualizza l'uscita (ad esempio, utilizzando un LCD o un led)



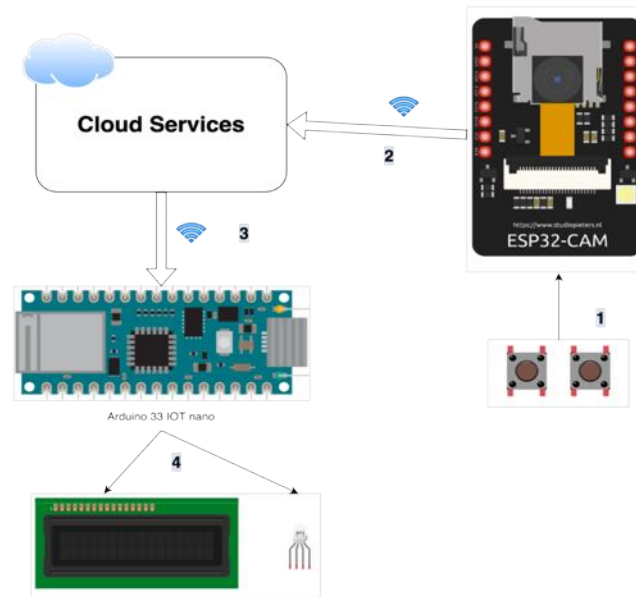


Figure 12 Architettura solo cloud

Cloud con Middleware

La Figura 13 mostra l'architettura che coinvolge il cloud e un middleware software che media le comunicazioni tra la scheda ESP32-CAM e i servizi cloud e altre schede.

In questo caso il middleware è un router "intelligente" di messaggi e notifiche, mentre la principale capacità computazionale è delegata ai servizi cloud.

Un esempio di interazione è composto dai seguenti passi:

- 1) L'utente preme il pulsante
- 2) La scheda ESP32-CAM
 - a. Cattura un'immagine
 - b. Invia l'immagine al middleware
- 3) Il Middleware
 - a. Sfrutta il servizio cloud per eseguire l'OCR
 - b. notifica il provvedimento alla scheda Arduino
- 4) La scheda Arduino
 - a. Riceve la misura
 - b. Visualizza l'uscita (ad esempio, utilizzando un LCD o un led)

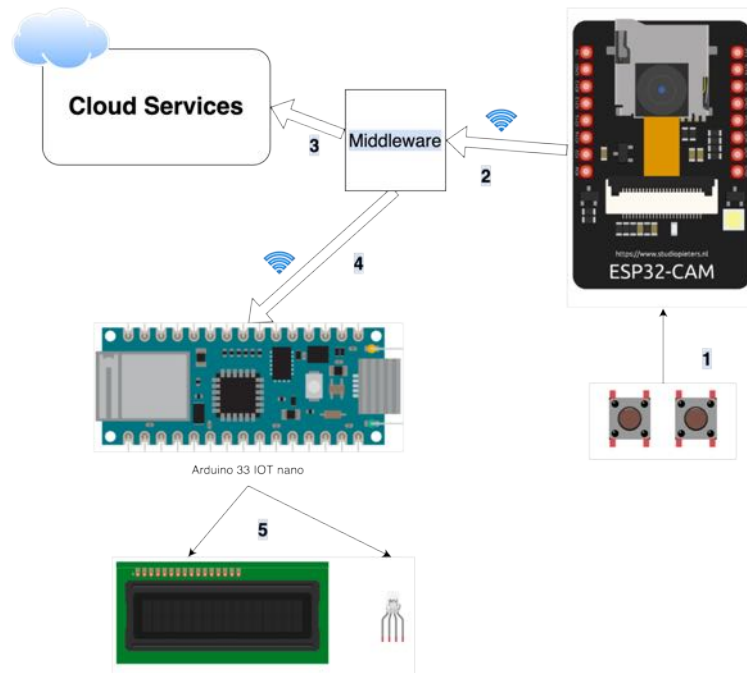


Figure 13 Cloud con Middleware

SOFTWARE

In questa sezione vengono descritte alcune librerie software utili che possono essere sfruttate per eseguire il riconoscimento delle misure e per implementare l'infrastruttura di comunicazione (servizi web) tra i componenti dell'Universal Measure Reader.

Sebbene sia possibile implementare un Universal Measure Reader nella maggior parte dei linguaggi di programmazione, in questa guida abbiamo scelto un linguaggio unico: Python.

Python è stato scelto perché ha una curva di apprendimento fluida e può essere sfruttato anche da programmatori non esperti. Inoltre, la base della programmazione Python viene insegnata sia nelle scuole professionali che nelle scuole superiori.

Optical Character Recognition (OCR)

I sistemi OCR sono applicazioni software in grado di riconoscere caratteri di testo in immagini o filmati. Appartengono ai seguenti campi dell'AI:

- Riconoscimento delle immagini
- Riconoscimento del modello
- Visione artificiale

L'origine dei sistemi OCR non è recente (9): nel 1927 il fisico Emanuel Goldber inventò una macchina in grado di convertire il testo stampato in codice telegrafico.

Negli anni '50 furono sviluppati i primi sistemi OCR commerciali; sono stati utilizzati principalmente per digitalizzare tagliandi e indirizzi postali.

Alla fine degli anni '60 il protagonista della scena del sistema OCR era l'Optical to Tactile CONverter, meglio conosciuto come Optacon (10). Si tratta di un dispositivo elettromeccanico che consente ai non vedenti di leggere materiale stampato. Il dispositivo è composto da due componenti: uno scanner per la lettura del testo e un pad che “traduce” le parole in vibrazioni avvertite sulla punta delle dita.

Dagli anni '70 agli anni 2000 le tecnologie OCR si sono espanse sempre di più: sono state fondate molte aziende e sono stati ampliati i loro campi di applicazione, ad esempio:

- Tecnologie assistive per non vedenti e ipovedenti
- Rilevamento di indirizzi e codici postali per il routing automatico della posta
- Digitalizzazione di vecchi giornali, libri e riviste



- Riconoscimento della forma
- Sicurezza: rilevamento del numero di targa

Dagli anni 2010, molte applicazioni software includono nativamente funzionalità OCR e la maggior parte dei fornitori di servizi cloud fornisce servizi API o SDK che offrono funzionalità OCR avanzate. Nelle sezioni successive viene descritto come sfruttare sia Tesseract, una delle principali applicazioni OCR open source, sia i servizi cognitivi di Microsoft Azure Computer Vision, un servizio cloud OCR molto buono e poco costoso.

Tesseract

Tesseract è senza dubbio il motore OCR **open source** più preciso. È stato sviluppato negli anni '80 da Hewlett-Packard, nel 2005 è stato rilasciato come open source. Dal 2006 lo sviluppo di Tesseract è supportato da Google, che lo integra nei propri servizi cloud (es. google drives, gmail, google cloud).

Installare Tesseract

Nel nostro progetto ci facciamo delle librerie python Tesseract, quindi è necessario installare il motore Tesseract nel server (o PC) dove verrà distribuito l'applicazione.

La documentazione dettagliata per l'installazione di tesseract è disponibile in (11); ne riportiamo di seguito una breve sintesi.

Su Windows

Il programma di installazione di Windows non è fornito direttamente nel repository github ufficiale di Tesseract-OCR, ma è ospitato dalla Mannheim University Library (UB Mannheim) al seguente URL: [Home · UB-Mannheim/tesseract Wiki · GitHub](#)

1. Prima di tutto, è necessario selezionare il programma di installazione per la versione corretta di Windows (32 or 64 bit)²

The latest installers can be downloaded here:

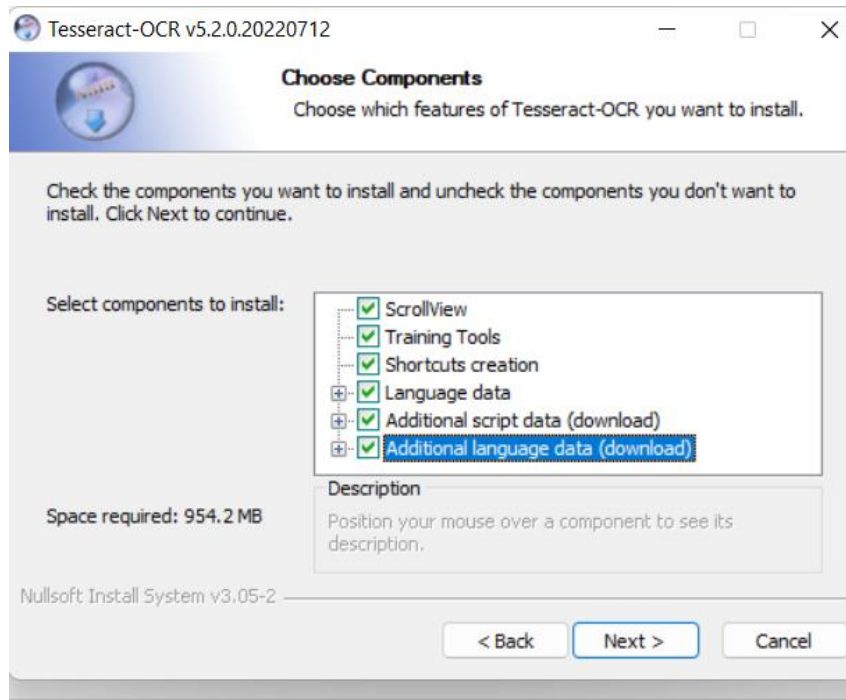
- [tesseract-ocr-w32-setup-v5.2.0.20220712.exe](#) (32 bit) and
- [tesseract-ocr-w64-setup-v5.2.0.20220712.exe](#) (64 bit) resp.

2. Si consiglia di selezionare tutte le funzionalità opzionali durante l'installazione.

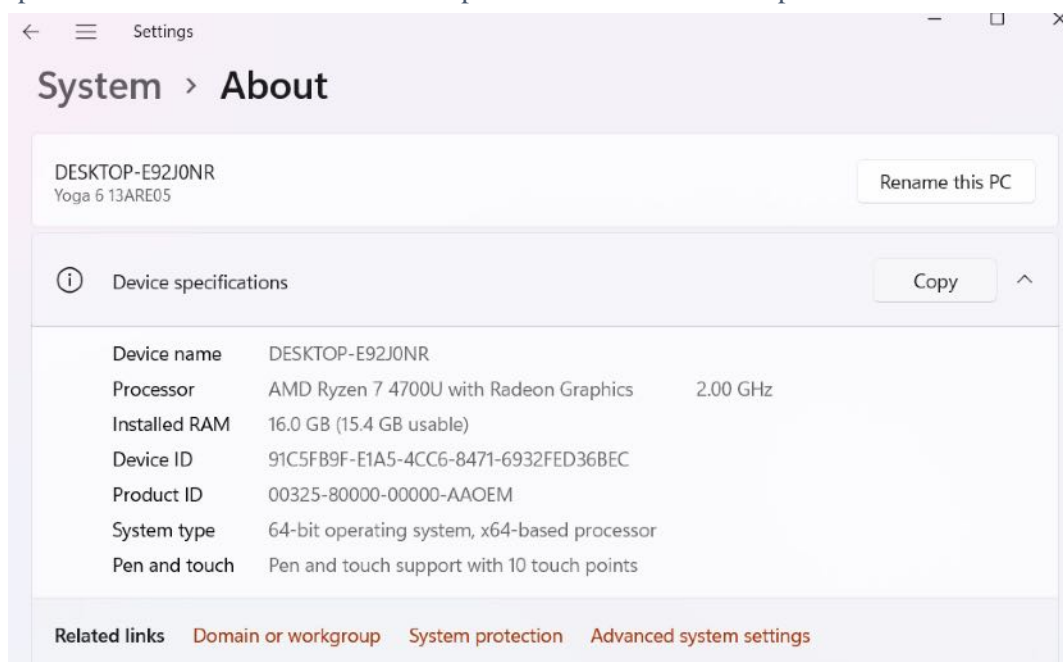
² To understand the system type, you need:

- Open the Start menu, type msinfo32 into the search box, and press Enter.
- In System Summary look for System Type. It can be either a x64-based PC (mean 64 bits) or a x86-based PC (mean 32 bits).

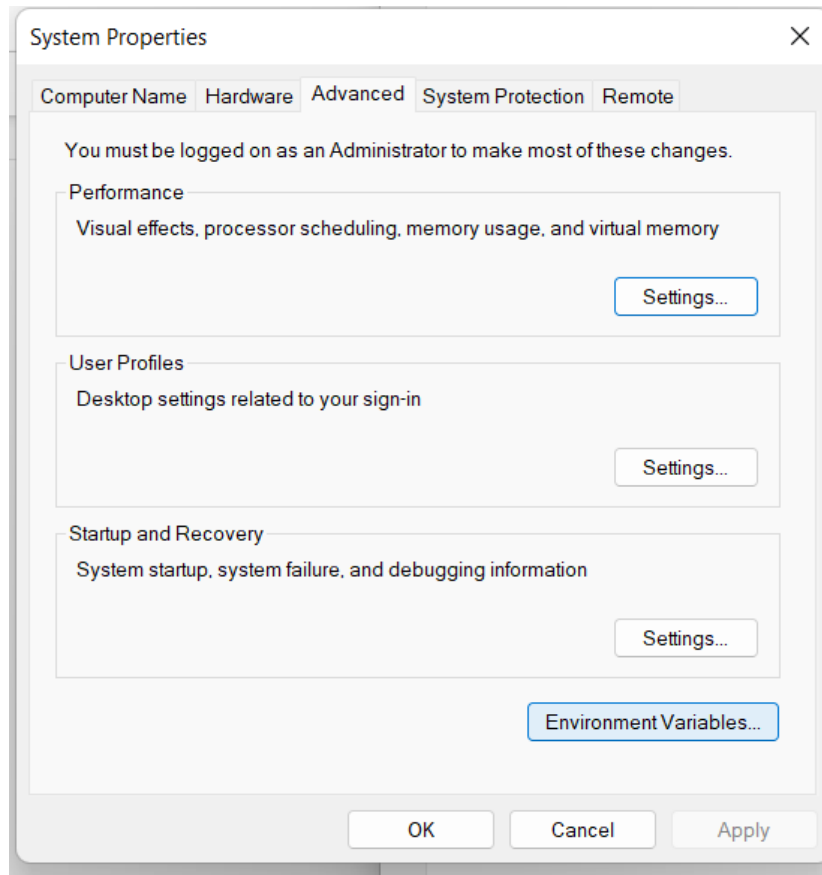




3. Al termine dell'installazione, è necessario aggiungere la cartella di installazione al percorso di sistema.
4. Aprire le informazioni di sistema da Impostazioni e fare clic su Impostazioni avanzate.

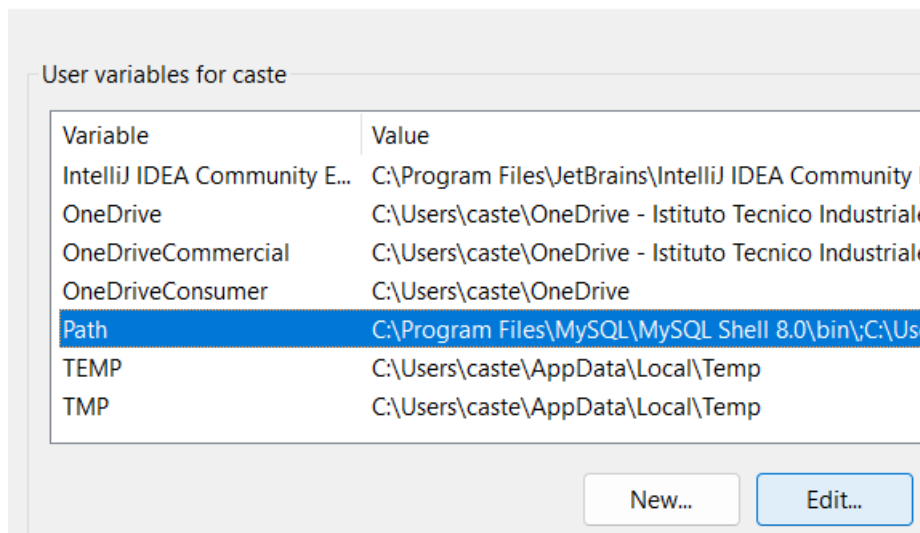


5. Selezionare Variabili d'ambiente

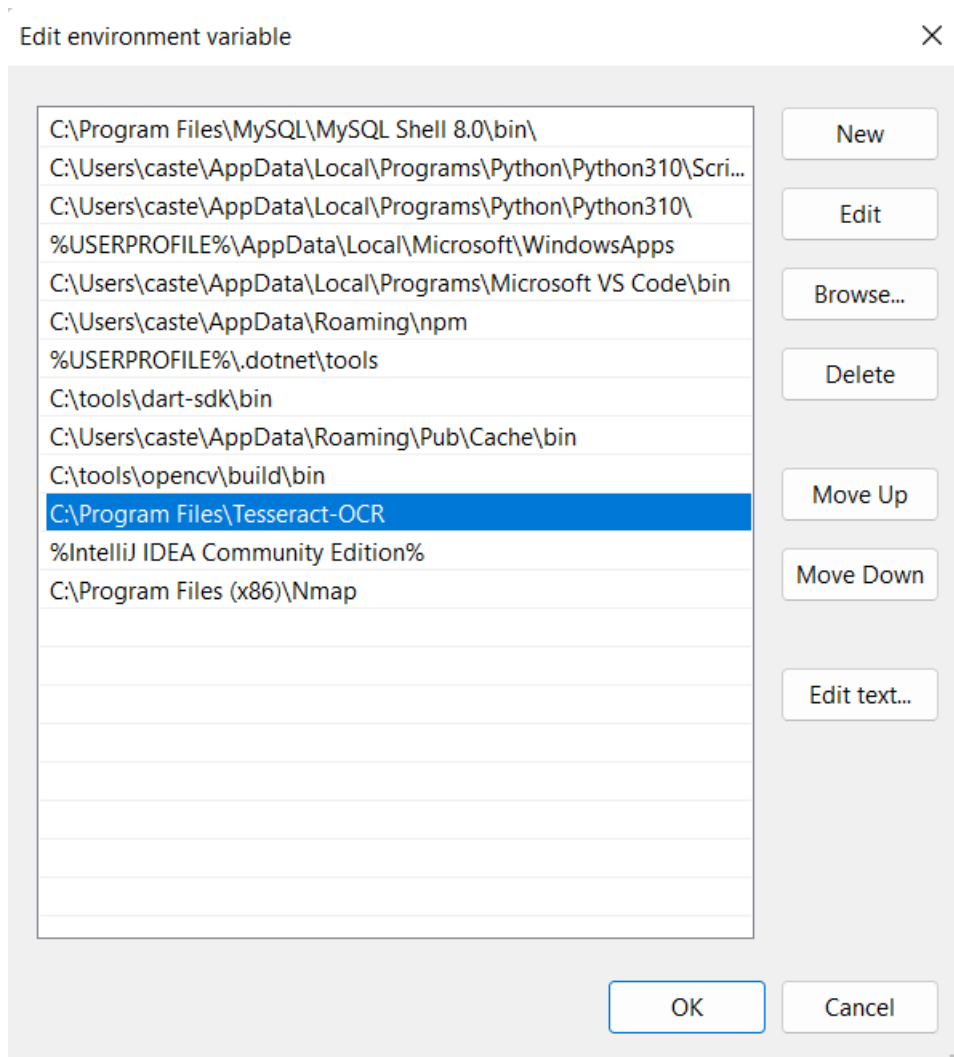


6. Selezionare la riga Percorso e fare clic su modifica.

Environment Variables



7. Scegli il percorso di installazione di Tesseract (tipicamente c:\Program Files\Tesseract-OCR).



Su Mac

Tesseract è disponibile su Mac tramite MacPorts o Homebrew.

Se usi MacPorts, esegui questo comando:

```
sudo port install tesseract
```

Se usi Homebrew, esegui questo comando:

```
brew install tesseract
```

Su Linux

Tesseract è disponibile come pacchetto per la maggior parte della distribuzione Linux, quindi la sua installazione è semplice.

Per esempio, per installare Tesseract in una distribuzione basata su Debian (come Ubuntu), eseguire questo comando:

```
sudo apt install tesseract-ocr  
sudo apt install libtesseract-dev
```



Tesseract Python Library

In python è disponibile un modulo chiamato pytesseract che è un wrapper per il motore Tesseract. Per installarlo nel tuo ambiente, esegui questo comando:

```
pip install pytesseract
```

Tipicamente non è sufficiente utilizzare solo la libreria pytesseract, ma in molti casi le immagini prima di essere inviate al motore OCR devono essere pre-elaborate applicando ridimensionamenti e filtri.

La libreria di elaborazione immagini open source più utilizzata è OpenCV e ovviamente è disponibile un modulo dedicato per Python.

Per installarlo nel tuo ambiente, esegui questo comando:

```
pip install opencv-contrib-python
```

OpenCV viene utilizzato per aprire l'immagine dai file e per applicare i filtri.

Utilizzo di Tesseract in Python.

Un codice Python minimo per sfruttare tesseract è il seguente:

```
1 import cv2
2 from pytesseract import*
3
4 custom_config = r'--oem 3 --psm 8'
5 image = cv2.imread('ocr2.jpeg')
6 gray_image = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
7 text = pytesseract.image_to_string(gray_image,config=custom_config,lang="ssd")
8 print(text)
```

- Le righe 1 e 2 importano i moduli necessari: OpenCv (cv2) e Tesseract (pytesseract).
- La riga 4 imposta una variabile contenente parametri di configurazione personalizzati che verranno spiegati meglio nella sezione successiva
- La riga 5 sfrutta OpenCV per caricare l'immagine ocr2.jpeg
- La riga 6 converte l'immagine in scala di grigi (tipicamente tesseract funziona meglio con le immagini in scala di grigi)
- La riga 7 esegue la funzione image_to_string del modulo tesseract. Questa funzione restituisce il testo letto all'interno dell'immagine passata come parametro. La funzione necessita dei seguenti parametri:
 - Immagine: il più delle volte è necessario applicare dei filtri all'immagine
 - o Parametri di configurazione personalizzati: la prossima sezione li spiegherà
 - o Lingua: è un parametro facoltativo, se non viene fornito verrà selezionata la lingua inglese. È anche possibile selezionare modelli di allenamento personalizzati.





- Alla riga 7 selezioniamo “ssd” come modello di lingua: è uno dei 3 che abbiamo proposto come modello personalizzato per il riconoscimento delle cifre da display lcd.
- Esiste una funzione alternativa a `image_to_string`, denominata `image_to_data` che restituisce un dizionario contenente tutto il testo riconosciuto all'interno delle coordinate in cui sono posizionati e la precisione stimata.
- La riga 8 stampa l'output.

Parametri di configurazione

Il motore Tesseract supporta due tipi di parametri di configurazione: modalità di segmentazione della pagina e modalità OCR.

Esistono 14 possibili modalità di segmentazione della pagina che indicano come si presume che il testo venga visualizzato all'interno dell'immagine fornita come input del motore.

Per impostare questo parametro è necessario utilizzare la parola chiave `--psm` prima del numero selezionato dalla Tabella 1.

Segmentation Mode	Description
1	Solo rilevamento orientamento e script (OSD).
2	Segmentazione automatica delle pagine con OSD.
3	Segmentazione automatica della pagina, ma nessun OSD o OCR. (non implementato)
4	Segmentazione della pagina completamente automatica, ma nessun OSD. (Predefinito)
5	Assumi una singola colonna di testo di dimensioni variabili.
6	Assumi un singolo blocco uniforme di testo allineato verticalmente.
7	Assumi un singolo blocco uniforme di testo.
8	Tratta l'immagine come una singola riga di testo.
9	Tratta l'immagine come una singola parola.
10	Tratta l'immagine come una singola parola in un cerchio.
11	Tratta l'immagine come un singolo carattere.
12	Testo scarno. Trova più testo possibile senza un ordine particolare.
13	Testo sparso con OSD.
14	Linea grezza. Tratta l'immagine come una singola riga di testo, aggirando gli hack specifici di Tesseract.

Table 1 Page Segmentation Modes (PSM)

Esistono 4 modalità del motore OCR che indicano come deve funzionare il motore Tesseract. Per sfruttare questo parametro di configurazione è necessario utilizzare la parola chiave `--oem` prima del numero riportato in Tabella 2.

Modalità del motore	Descrizione
1	Solo motore precedente.
2	Solo motore LSTM per reti neurali.
3	Legacy + Motori LSTM.
4	Predefinito, in base a ciò che è disponibile.

Table 2 OCR Engine Modes



Progetti


Questa sezione comprende 4 progetti, descritti passo passo, che coinvolgono l'Universal Measure Reader.

Si noti che tutti questi progetti sono solo proof of concept (POC), pertanto potrebbero non essere pronti per essere utilizzati in ambienti di lavoro reali. Prima di utilizzare questi progetti in ambienti reali, è necessario considerare la sicurezza dei dispositivi elettronici utilizzati (progettare e sviluppare contenitori conformi agli standard IP65, IP66 o IP67 (12)) e l'ingegnerizzazione del prodotto (ad es. circuiti stampati).

Alcuni esempi di box Arduino per stampante 3D sono riportati in (13).

Progetto 1: cattura una misura ogni 30 secondi

DIFFICULTY	Hardware Easy Medium Hard  Software Easy Medium Hard 
REQUISITI SOFTWARE ARDUINO IDE INSTALLATO SU WINDOWS, LINUX O MACOS	REQUISITI SOFTWARE Arduino IDE installato su Windows, Linux o macOS
REQUISITI HARDWARE SCHEDA ESP32-CAM	REQUISITI HARDWARE Scheda ESP32-CAM

TEMPO	 circa 40 minuti
--------------	---

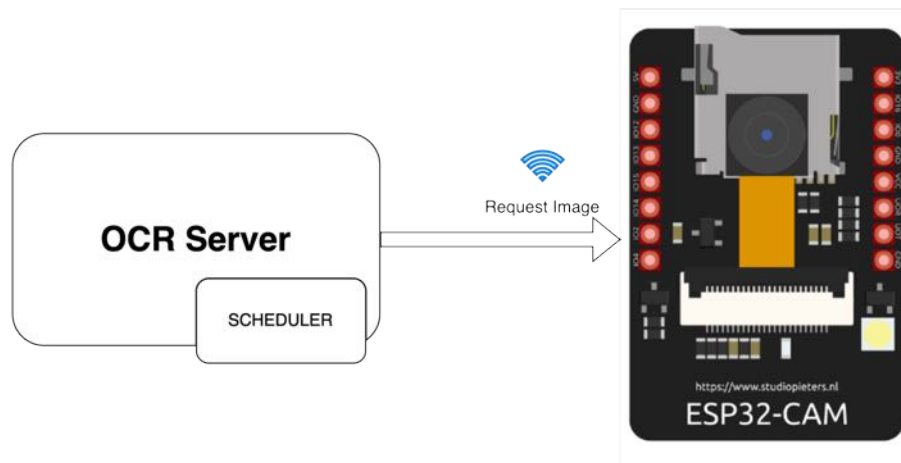


Figure 14 Project 1 overview

1. Come primo passo è necessario configurare l'IDE di Arduino per la scheda ESP32-CAM come descritto nel Tutorial 1: Aggiunta della libreria ESP32 all'IDE di Arduino
2. Il codice di esempio fornito all'interno della libreria espressif è già completamente funzionante e fornisce un servizio web che consente di controllare la fotocamera. Sugeriamo tuttavia di potenziare tale servizio web aggiungendo due modalità (led-on e led-off) per il controllo anche del potente led bianco incluso nella scheda. In accordo con il vecchio display del dispositivo potrebbe essere necessario modificare le condizioni di luce per meglio riconoscere la misura.
3. E' necessario modificare il file `app_httpd.cpp` aggiungendo le seguenti funzioni statiche:

```
347
348 static esp_err_t ledon_handler(httpd_req_t *req)
349 {
350     esp_err_t res = ESP_OK;
351
352     digitalWrite(4, HIGH);
353
354     return res;
355
356 }
357
358 static esp_err_t ledoff_handler(httpd_req_t *req)
359 {
360     esp_err_t res = ESP_OK;
361     digitalWrite(4, LOW);
362
363     return res;
364
365 }
```

Queste funzioni fondamentalmente impostano ALTO o BASSO il valore del LED (GPIO 4); ricevono una richiesta http come parametro anche se non la utilizzano e restituiscono un oggetto `esp_err_t` che può assumere i valori `ESP_OK` o `ESP_FAIL` a seconda che l'operazione venga eseguita correttamente.

4. 4. Definire la registrazione URI delle due funzioni

```
1250 //romuas adding led management
1251 httpd_uri_t ledon_uri = {
1252     .uri = "/led-on",
1253     .method = HTTP_GET,
1254     .handler = ledon_handler,
1255     .user_ctx = NULL};
1256
1257 httpd_uri_t ledoff_uri = {
1258     .uri = "/led-off",
1259     .method = HTTP_GET,
1260     .handler = ledoff_handler,
1261     .user_ctx = NULL};
1262
```

La struttura `httpd_uri_t` permette di definire un nuovo gestore http per il servizio web. È necessario definire l'uri, il metodo http e il gestore della funzione che verrà chiamato dal web service.



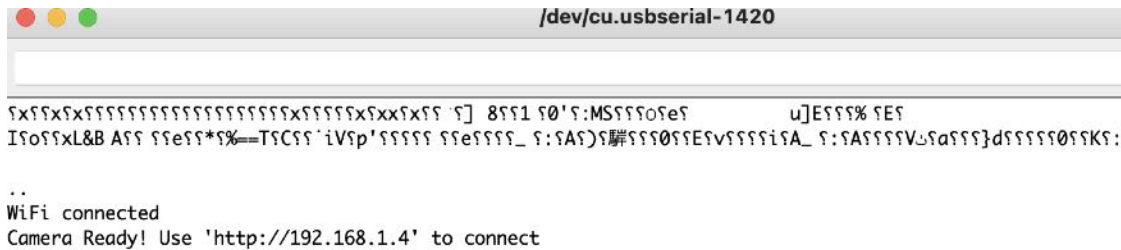
5. Aggiunta della funzione al servizio web

```
1272     if (httpd_start(&camera_httpd, &config) == ESP_OK)
1273     {
1274         httpd_register_uri_handler(camera_httpd, &index_uri);
1275         httpd_register_uri_handler(camera_httpd, &cmd_uri);
1276         httpd_register_uri_handler(camera_httpd, &status_uri);
1277         httpd_register_uri_handler(camera_httpd, &capture_uri);
1278         httpd_register_uri_handler(camera_httpd, &bmp_uri);
1279
1280         httpd_register_uri_handler(camera_httpd, &xclk_uri);
1281         httpd_register_uri_handler(camera_httpd, &reg_uri);
1282         httpd_register_uri_handler(camera_httpd, &greg_uri);
1283         httpd_register_uri_handler(camera_httpd, &pll_uri);
1284         httpd_register_uri_handler(camera_httpd, &win_uri);
1285         httpd_register_uri_handler(camera_httpd, &ledon_uri);
1286         httpd_register_uri_handler(camera_httpd, &ledoff_uri);
1287     }
1288
```

Le righe evidenziate registrano gli oggetti httpd_uri_t sopra definiti nel servizio web.

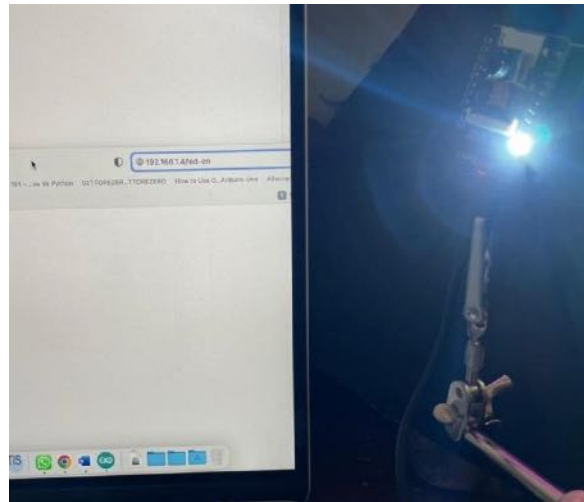
6. 6. Caricare il codice di modifica sulla scheda ESP32-CAM.

Dopo aver caricato lo sketch, puoi vedere nel monitor seriale l'indirizzo ip assegnato dal tuo router all'ESP32-CAM. Nel nostro caso è 192.168.1.4.



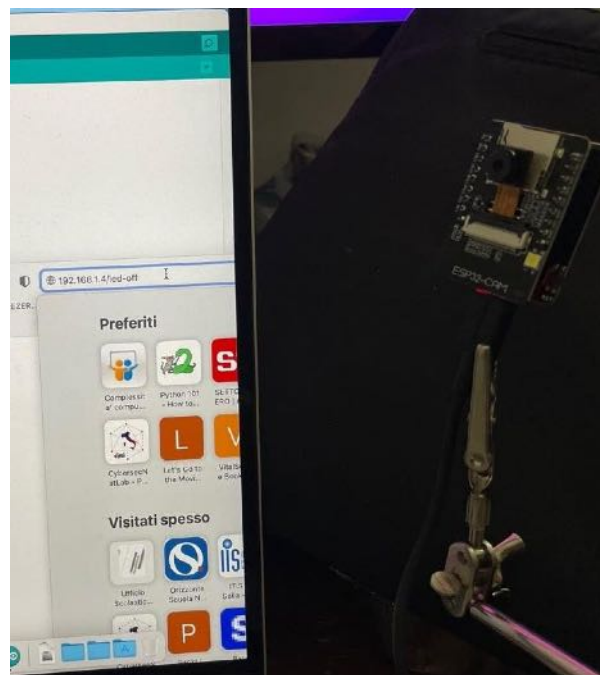
Per accendere il led è necessario aprire un browser ed inserire il seguente indirizzo: <http://{ESP32-CAM IP}/led-on>

Nel nostro caso <http://192.168.1.4/led-on>



Per spegnere il led è necessario aprire un browser ed inserire il seguente indirizzo: <http://{ESP32-CAM IP}/led-off>

Nel nostro caso <http://192.168.1.4/led-off>



7. Dopo aver terminato la codifica Arduino, è necessario passare alla parte del server OCR.
8. Dobbiamo sviluppare un programma che
9. ogni X secondi esegue una richiesta al servizio web ESP32-CAM per il recupero di un'immagine
10. poi sfrutta il motore tesseract per riconoscere la misura ed infine visualizzando la misura
11. Iniziamo a codificare lo scheduler del server OCR in python installando due librerie: schedule e request:

```
pip install requests  
pip install schedule
```



La libreria delle richieste è necessaria per eseguire richieste HTTP verso il server Web ESP32-CAM, mentre la libreria di pianificazione offre funzionalità per la pianificazione dell'esecuzione delle attività.

Come descritto nella sezione Software faremo uso delle librerie OpenCV e Tesseract e anche della nostra libreria ocr che avvolge il motore Tesseract.

```
1  import schedule
2  import requests
3  import time
4  import json
5  import shutil
6  from ocr import ocr_image
7
```

Le righe da 1 a 6 importano i moduli richiesti:

- schedule: permette di programmare le attività
- richieste: permette di eseguire richieste http
- tempo: viene utilizzato per la funzione sleep nel ciclo principale del programma
- json: viene utilizzato per recuperare la configurazione memorizzata in un file json
- shutil: serve per copiare il file di download dall'ESP32-CAM
- ocr: è il nostro wrapper del motore Tesseract

12.12. Caricamento della configurazione

I parametri di configurazione dell'applicazione devono essere memorizzati al di fuori del nostro codice sorgente per consentire di modificarli facilmente. Json è uno dei formati di file più comuni per l'archiviazione delle configurazioni e grazie alla libreria json è molto semplice gestire tale file.

La Figura 15 mostra un esempio di file di configurazione per la nostra applicazione di pianificazione.

Il file `schedule_confing.json` consente di definire:

- l'intervallo di cattura dell'immagine espresso in secondi (`capture_seconds`)
- l'URL del server Web ESP32-CAM (`capture_url`)
- la modalità di segmentazione della pagina Tesseract (`psm`)
- il linguaggio Tesseract (dizionario)
- l'eventuale zona di coltivazione (`crop`, `crop-data`)
- se è necessario mostrare una finestra di debug (`showWindow`)



```
} schedule_config.json > ...
1  {
2      "capture_seconds": 1,
3      "capture_url": "http://192.168.1.4/capture",
4      "psm": 9,
5      "dictionary": "ssd",
6      "crop": true,
7      "crop-data": { "x": 380, "y": 490, "width": 210, "height": 320 },
8      "showWindow": true
9  }
10 }
```

Figure 15 Example of scheduler configuration

Caricare un file json in Python è abbastanza semplice come riportato nel codice sottostante:

```
8  def loadConfiguration():
9      print('Loading configuration')
10     with open('schedule_config.json') as config_file:
11         configuration = json.load(config_file)
12     return configuration
```

This function opens the configuration file and copy its content in a dictionary.

The configuration in Figure 15 is converted in the below python object:

```
{'capture_seconds': 1, 'capture_url': 'http://192.168.1.4/capture', 'psm': 9, 'dictionary': 'ssd', 'crop': True, 'crop-data': {'x': 380, 'y': 490, 'width': 210, 'height': 320}, 'showWindow': True}
```

13. Cattura la misura

La cattura della misura comporta diverse operazioni come riportato nel codice sottostante.

```
14 def captureImage(configuration):
15     global index
16     image_url=configuration['capture_url']
17     filename = image_url.split("/")[-1]
18     r = requests.get(image_url, stream = True)
19
20     if r.status_code == 200:
21         r.raw.decode_content = True
22         filename = str(index)+"-"+filename+".jpeg"
23         with open(filename,'wb') as f:
24             shutil.copyfileobj(r.raw, f)
25
26         index+=1
27         print('Image sucessfully Downloaded: ',filename)
28         text, conf = ocr_image(filename,configuration['psm'],
29                               configuration['dictionary'],
30                               configuration['showWindow'],
31                               crop=configuration["crop-data"].values()
32                               if configuration["crop"] else None )
33         print("Found text", text,conf)
34     else:
35         print('Image Couldn\'t be retrieved')
36
```

- La riga 14 definisce la funzione captureImage che ha come parametro la configurazione letta dal file json
- La riga 15 recupera la variabile globale denominata index e utilizzata per memorizzare l'immagine con nomi diversi. Viene aumentato dopo ogni cattura.
- La riga 16 recupera l'url del servizio web dall'oggetto di configurazione
- La riga 17 esegue una richiesta HTTP GET per il recupero dell'immagine dall'ESP32-CAM
- La riga 20 controlla lo stato delle richieste: 200 significa che la richiesta era OK
- La riga 21 indica che il contenuto grezzo della risposta deve essere decodificato in base all'intestazione http Content-Type
- La riga 22 genera un nome file univoco
- Le righe 23 e 24 salvano il contenuto della risposta in un file immagine
- Le righe dalla 28 alla 32 chiamano la funzione ocr_image passando tutti i parametri di configurazione
- Riga 33 visualizza il testo riconosciuto, mentre Riga 35 riporta gli eventuali errori verificatisi

14. Pianifica l'acquisizione dell'immagine nella funzione principale.

```
37  if __name__ == "__main__":
38      index = 1
39      configuration = loadConfiguration()
40      capture_seconds = configuration['capture_seconds']
41      schedule.every(capture_seconds).seconds.do(captureImage, configuration)
42      captureImage(configuration)
43      while True:
44          schedule.run_pending()
45          time.sleep(1)
```

- La riga 37 controlla se lo script è stato eseguito direttamente e non è stato incluso come modulo
- La riga 38 dichiara una variabile globale utilizzata per generare nomi di immagine univoci
- La riga 39 carica la configurazione dello scheduler
- La riga 40 recupera l'intervallo di tempo di cattura
- La riga 41 configura l'oggetto schedule per l'esecuzione della funzione captureImage in base all'intervallo di tempo
- La riga 42 esegue la funzione captureImage per la prima volta senza attendere l'intervallo di tempo di acquisizione
- Le righe da 43 a 45 rappresentano il ciclo principale dello scheduler. Il metodo run_pending dell'oggetto schedule deve essere eseguito continuamente per consentire allo scheduler di eseguire l'attività nei tempi previsti.
- La funzione time.sleep è necessaria per consentire ad altri thread di assumere il controllo del sistema per un po'.

15. Risultati

Come esempio abbiamo cercato di catturare la misura della temperatura da una stazione meteorologica domestica.

Siamo partiti dall'immagine grezza acquisita (Figura 16) poi attraverso la configurazione indichiamo di ritagliare l'immagine in una zona specifica (Siamo interessati solo alla temperatura interna, che viene mostrata nell'angolo in basso a sinistra del display).

La Figura 17 mostra l'output del programma: viene visualizzata l'immagine ritagliata con il risultato del riconoscimento del motore Tesseract.



Figure 16 Raw Image

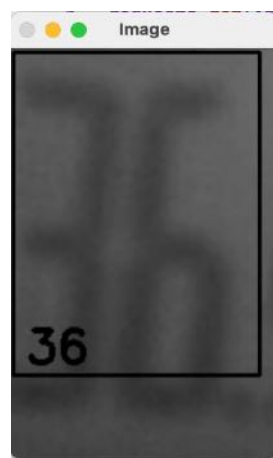






Figure 17 Cropped Image with Tesseract Recognition

Progetto 2: cattura una misura alla pressione di un pulsante

<p style="text-align: center;">DIFFICULTY</p>	<p style="text-align: center;">Hardware</p> <p style="text-align: center;">Easy Medium Hard</p>  <p style="text-align: center;">Software</p> <p style="text-align: center;">Easy Medium Hard</p> 
<p style="text-align: center;">SOFTWARE REQUIREMENTS</p>	<p>Arduino IDE installed on Windows, Linux or macOS, Tesseract Engine</p>
<p style="text-align: center;">HARDWARE REQUIREMENTS</p>	<p>Scheda ESP32-CAM</p> <p>Programmatore (es. ESP32-CAM-MB), solo per il caricamento dello sketch</p> <p>Tagliere</p> <p>Alimentatore per tagliere</p> <p>2 resistenze (10 KΩ)</p> <p>2 pulsanti</p> <p>Fili</p>
<p style="text-align: center;">TEMPO</p>	<p style="text-align: center;">   circa 60 minuti </p>

La scheda ESP32-CAM è programmata per inviare un'immagine al nostro server OCR quando viene premuto un pulsante. Per comodità il led bianco luminoso è comandato da un altro pulsante. Il server OCR mette a disposizione un servizio web per caricare l'immagine e sfruttare Tesseract Engine per il riconoscimento delle misure. Il server OCR fornisce anche una pagina web che mostra le misure recuperate. Una panoramica di alto livello di questo progetto è rappresentata nella Figura 18.

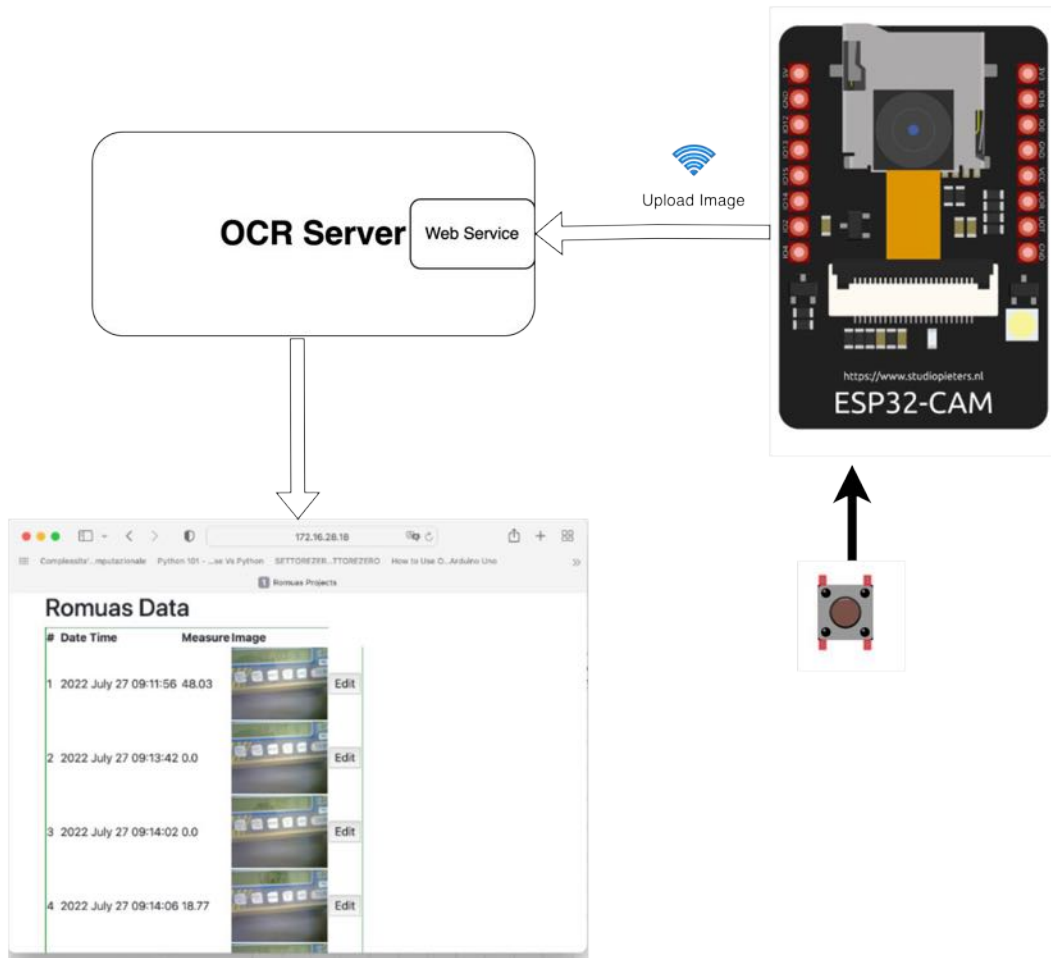


Figure 18 Project 2 high level overview

1. Come prima operazione è necessario cablare il componente come mostrato in Figura 19.
 - Collegare l'alimentatore alla breadboard per fornire 5 V e messa a terra alle guide di alimentazione
 - Collegare il pin 5V dell'ESP32-CM e GND alle rispettive linee Power Trailing
 - Collegare il GPIO 14 al pulsante A e con una resistenza da 10 K Ω a GND
 - Collegare il GPIO 2 al pulsante B e con una resistenza da 10 K Ω a GND
 - Fornire 5 V ai pulsanti

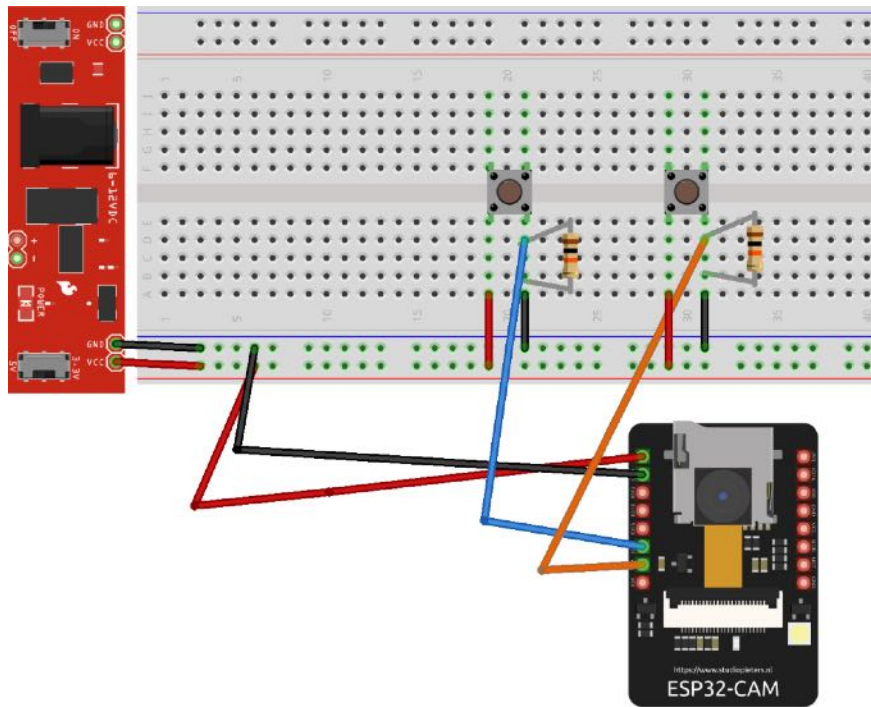


Figure 19 Project 2 Wiring

2. Crea un nuovo schizzo Arduino

Lo sketch proposto si basa principalmente sul lavoro presentato in (14), con la differenza principale che riguarda il lato server della soluzione: noi ci faremo del semplice web server python mentre l'articolo propone PHP.

3. Importare le librerie necessarie.

```

13  #include <Arduino.h>
14  #include <WiFi.h>
15  #include "soc/soc.h"
16  #include "soc/rtc_cntl_reg.h"
17  #include "esp_camera.h"
  
```

- La riga 13 include la libreria Arduino standard, utilizzata per funzioni come digitalWrite, digitalRead, pinMode, ecc.
 - La riga 14 include la libreria WiFi utilizzata per collegare la scheda ESP32-CAM a una rete Wireless e creare un client HTTP per l'invio dell'immagine al server OCR
 - Le righe da 15 a 17 comprendono le librerie specifiche per la scheda ESP32-CAM.
4. Definire le variabili di configurazione.



```
18
19 const char* ssid =
20 const char* password =
21
22 String serverName = "172.16.28.18";
23
24 String serverPath = "/upload-image"; // The server path for uploading the IMAG
25
26 const int serverPort = 5001;
27
28 const int buttonPhotoPin = 2; //change with the PIN you've connected the button
29 const int buttonFlashPin = 14;
30 int buttonPhotoState = 0;
31 int buttonFlashState = LOW;
32 const int ledPin = 4;
33
34 WiFiClient client;
35
```

- Le righe 19 e 20 consentono di impostare i parametri di connessione alla rete Wi-Fi: l'SSID (nome della rete wireless) e la password..
- Le righe dalla 22 alla 26 permettono di impostare l'indirizzo del servizio web dove caricare le immagini. L'indirizzo è composto dal serverName (o indirizzo ip), dalla porta e dal percorso del server. Nel nostro esempio l'indirizzo finale sarà `http://172.16.28.18:5001/upload-image`
- Le righe da 28 a 32 definiscono il pin a cui sono collegati i pulsanti e il loro stato
- La riga 34 dichiara il `WiFiClient` utilizzato per l'invio dell'immagine

5. Definire le configurazioni della telecamera GPIO

```
36 // CAMERA_MODEL_AI_THINKER
37 #define PWDN_GPIO_NUM    32
38 #define RESET_GPIO_NUM  -1
39 #define XCLK_GPIO_NUM    0
40 #define SIOD_GPIO_NUM    26
41 #define SIOC_GPIO_NUM    27
42
43 #define Y9_GPIO_NUM       35
44 #define Y8_GPIO_NUM       34
45 #define Y7_GPIO_NUM       39
46 #define Y6_GPIO_NUM       36
47 #define Y5_GPIO_NUM       21
48 #define Y4_GPIO_NUM       19
49 #define Y3_GPIO_NUM       18
50 #define Y2_GPIO_NUM        5
51 #define VSYNC_GPIO_NUM    25
52 #define HREF_GPIO_NUM     23
53 #define PCLK_GPIO_NUM     22
54
```

6. Definire una funzione per configurare la telecamera



```
56 camera_config_t configCamera(){
57     camera_config_t config;
58     config.ledc_channel = LEDC_CHANNEL_0;
59     config.ledc_timer = LEDC_TIMER_0;
60     config.pin_d0 = Y2_GPIO_NUM;
61     config.pin_d1 = Y3_GPIO_NUM;
62     config.pin_d2 = Y4_GPIO_NUM;
63     config.pin_d3 = Y5_GPIO_NUM;
64     config.pin_d4 = Y6_GPIO_NUM;
65     config.pin_d5 = Y7_GPIO_NUM;
66     config.pin_d6 = Y8_GPIO_NUM;
67     config.pin_d7 = Y9_GPIO_NUM;
68     config.pin_xclk = XCLK_GPIO_NUM;
69     config.pin_pclk = PCLK_GPIO_NUM;
70     config.pin_vsync = VSYNC_GPIO_NUM;
71     config.pin_href = HREF_GPIO_NUM;
72     config.pin_sscb_sda = SIOD_GPIO_NUM;
73     config.pin_sscb_scl = SIOC_GPIO_NUM;
74     config.pin_pwdn = PWDN_GPIO_NUM;
75     config.pin_reset = RESET_GPIO_NUM;
76     config.xclk_freq_hz = 20000000;
77     config.pixel_format = PIXFORMAT_JPEG;
78     // init with high specs to pre-allocate larger buffers
79     if(psramFound()){
80         config.frame_size = FRAMESIZE_SVGA;
81         config.jpeg_quality = 10; //0-63 lower number means higher quality
82         config.fb_count = 2;
83     } else {
84         config.frame_size = FRAMESIZE_CIF;
85         config.jpeg_quality = 12; //0-63 lower number means higher quality
86         config.fb_count = 1;
87     }
88     return config;
89 }
90 }
```

L'oggetto camera_config_t viene inizializzato con la configurazione dei pin

7. Definire una funzione per l'avvio della fotocamera

```
--
92 void startCamera(camera_config_t config){
93     // camera init
94     esp_err_t err = esp_camera_init(&config);
95     if (err != ESP_OK) {
96         Serial.printf("Camera init failed with error 0x%x", err);
97         delay(1000);
98         ESP.restart();
99     }
100
101
102 }
```

8. Definire la funzione per l'invio della foto

```
104 String sendPhoto() {
105     String getAll;
106     String getBody;
107
108     camera_fb_t * fb = NULL;
109     fb = esp_camera_fb_get();
110     if(!fb) {
111         Serial.println("Camera capture failed");
112         delay(1000);
113         ESP.restart();
114     }
115
116     Serial.println("Connecting to server: " + serverName);
117 }
```

```
118 if (client.connect(serverName.c_str(), serverPort)) {
119     Serial.println("Connection successful!");
120     String head = "--Romuas\r\nContent-Disposition: form-data; name=\"file\"; filename=\"esp32-cam.jpg\"\r\nContent-Type: image/jpeg\r\n\r\n";
121     String tail = "\r\n--Romuas--\r\n";
122
123
124     uint32_t imageLen = fb->len;
125     uint32_t extraLen = head.length() + tail.length();
126     uint32_t totalLen = imageLen + extraLen;
127
128     client.println("POST " + serverPath + " HTTP/1.1");
129     client.println("Host: " + serverName);
130     client.println("Content-Length: " + String(totalLen));
131     client.println("Content-Type: multipart/form-data; boundary=Romuas");
132     client.println();
133     client.print(head);
134
135     uint8_t *fbBuf = fb->buf;
136     size_t fbLen = fb->len;
137     for (size_t n=0; n<fbLen; n=n+1024) {
138         if (n+1024 < fbLen) {
139             client.write(fbBuf, 1024);
140             fbBuf += 1024;
141         }
142         else if (fbLen%1024>0) {
143             size_t remainder = fbLen%1024;
144             client.write(fbBuf, remainder);
145         }
146     }
147     client.print(tail);
148
149     esp_camera_fb_return(fb);
150
151     int timeoutTimer = 10000;
152     long startTimer = millis();
153     boolean state = false;
154
155     while ((startTimer + timeoutTimer) > millis()) {
156         Serial.print(".");
157         delay(100);
158         while (client.available()) {
159             char c = client.read();
160             if (c == '\n') {
161                 if (getAll.length()==0) { state=true; }
162                 getAll = "";
163             }
164             else if (c != '\r') { getAll += String(c); }
165             if (state==true) { getBody += String(c); }
166             startTimer = millis();
167         }
168         if (getBody.length()>0) { break; }
169     }
170     Serial.println();
171     client.stop();
172     Serial.println(getBody);
173 }
174 else {
175     getBody = "Connection to " + serverName + " failed.";
176     Serial.println(getBody);
177 }
178 return getBody;
179 }
```

9. Define the setup function that is called once at the start of the program

```
181 void setup() {
182   WRITE_PERI_REG(RTC_CNTL_BROWN_OUT_REG, 0);
183
184   pinMode(buttonPhotoPin, INPUT);
185   pinMode(buttonFlashPin, INPUT);
186   pinMode(ledPin, OUTPUT);
187
188   Serial.begin(115200);
189
190   WiFi.mode(WIFI_STA);
191   Serial.println();
192   Serial.print("Connecting to ");
193   Serial.println(ssid);
194   WiFi.begin(ssid, password);
195   while (WiFi.status() != WL_CONNECTED) {
196     Serial.print(".");
197     delay(500);
198   }
199   Serial.println();
200   Serial.print("ESP32-CAM IP Address: ");
201   Serial.println(WiFi.localIP());
202
203   camera_config_t config =configCamera();
204   startCamera(config);
205
206   //sendPhoto();
207 }
---
```

10. Definire il ciclo principale del programma

```
209 void loop() {
210   buttonPhotoState = digitalRead(buttonPhotoPin);
211   int currentButtonFlashState = digitalRead(buttonFlashPin);
212   if (currentButtonFlashState == HIGH){
213     buttonFlashState = buttonFlashState == HIGH ? LOW : HIGH;
214     digitalWrite(ledPin, buttonFlashState);
215   }
216   Serial.println(buttonPhotoState);
217   if (buttonPhotoState == HIGH) {
218     sendPhoto();
219   }
220 }
```

11. Ora è il turno dell'implementazione del server web OCR.

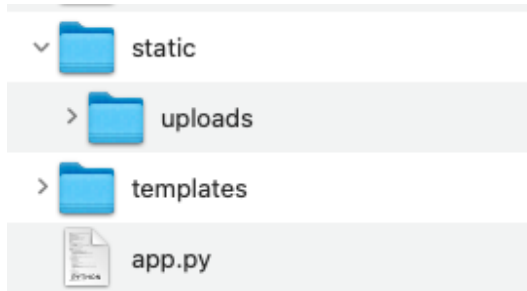
Abbiamo deciso di utilizzare il framework Flask per implementare un web server che fornisca sia un metodo per caricare l'immagine e riconoscere la misura sia una pagina web per visualizzare i risultati.

Per prima cosa dobbiamo installare altre due librerie:

```
pip install flask
pip install xlswriter
```

12. Creare un progetto Flask

Il progetto ha la seguente struttura:



La cartella static/uploads memorizza le immagini caricate, la cartella templates contiene i template delle pagine html, mentre il file app.py contiene il codice dell'applicazione

13. Importare le librerie richieste

```
1 import os
2 from flask import Flask, jsonify, render_template, request, send_file
3 from ocr import romuas_ocr, ocr_core
4 from datetime import datetime
5 import xlswriter
```

La riga 1 importa la libreria del sistema operativo utilizzata per eseguire operazioni sui file di caricamento

La riga 2 importa le funzioni del pallone

La riga 3 importa la nostra libreria ocr

La riga 5 importa la libreria per l'esportazione dei dati in formato Excel

14. Inizializzare l'applicazione Flask

```
8 # define a folder to store and later serve the images
9 UPLOAD_FOLDER = 'static/uploads/'
10
11 # allow files of a specific type
12 ALLOWED_EXTENSIONS = set(['png', 'jpg', 'jpeg'])
13
14 samples = []
15 img_number = 0
16
17 app = Flask(__name__)
18 app.config['UPLOAD_FOLDER'] = UPLOAD_FOLDER
```

Le righe dalla 8 alla 18 dichiarano alcuni parametri di configurazione, ovvero la cartella di upload, l'estensione dell'immagine consentita e creano l'app flask.

La variabile sample memorizzerà le misure, mentre img_number viene utilizzato per generare un nome univoco delle immagini caricate.

15. Definire una funzione per il controllo del file caricato.

```
20 # function to check the file extension
21 def allowed_file(filename):
22     return '.' in filename and \
23         |         | filename.rsplit('.', 1)[1].lower() in ALLOWED_EXTENSIONS
24
```

The function checks if the file extension is in the list of allowed extensions.

16. Create routing for the home page

```
26 @app.route('/')
27 def home_page():
28     return render_template('index.html')
29
```

In Flask il routing (ovvero la mappatura tra l'indirizzo e il metodo eseguito) è definito tramite decorator . Le righe dalla 26 alla 28 assegnano alla route '/' la funzione home_page che visualizza semplicemente la pagina web index.html.

17. Definire una funzione per l'esportazione e file Excel con le misure riconosciute

```
30 @app.route('/download-excel', methods=['GET'])
31 def download_excel():
32     with xlsxwriter.Workbook('data-tmp-2.xlsx') as workbook:
33         worksheet = workbook.add_worksheet()
34         worksheet.write_row(0, 0, ('#', 'Date Time', 'Measure'))
35         for row_num, data in enumerate(map(lambda s: (s[3], s[0], s[1]), samples)):
36             worksheet.write_row(row_num+1, 0, data)
37         workbook.close()
38     try:
39         return send_file('data-tmp-2.xlsx', download_name='excel_export.xlsx')
40     except Exception as e:
41         return str(e)
```

In questa funzione viene sfruttata la libreria xlsxwriter per generare un file Excel contenente le misure raccolte.

18. Definire la funzione per la gestione del caricamento delle immagini

```
43 @app.route('/upload-image', methods=['POST'])
44 def uploadImage():
45     if 'file' not in request.files:
46         return "No file"
47     file = request.files['file']
48     # if no file is selected
49
50
51     if file and allowed_file(file.filename):
52         global img_number
53         filePath = os.path.join(app.config['UPLOAD_FOLDER'], str(img_number)+"-"+file.filename)
54         img_number+=1
55         file.save(filePath)
56         print("Saved", filePath)
57         ocr = romuas_ocr(filePath)
58         text = ocr.show_ocr((560,215,230,120))
59         samples.append((datetime.now().strftime("%Y %B %d %H:%M:%S"), text, filePath, img_number))
60         return "OK"
```

Questa funzione controlla se è presente un file nella richiesta e se ha un'estensione valida, quindi il file viene salvato sul server.

Dopo aver salvato il file, viene eseguito il motore Tesseract (righe 57 e 58) e il risultato viene aggiunto all'oggetto campioni che memorizza la misura riconosciuta, l'immagine e il riferimento temporale (riga 59).

19. Definire il percorso per la pagina dei risultati

```
62 @app.route('/samples', methods=['GET'])
63 def samples_page():
64     return render_template('samples.html', data=samples)
```

Quando l'utente tenta di accedere all'url /samples, verrà visualizzato il template samples.html avente come parametro l'oggetto contenente le misure.

20. Esecuzione dell'applicazione

```
if __name__ == '__main__':
    app.run(host='0.0.0.0', port=5001)
```

Questo è il codice per avviare il server web. Il parametro host '0.0.0.0' consente di pubblicare il server Web sulla rete, mentre il parametro port consente di definire la porta.

21. Definire il modello di pagina di esempio.

Il nucleo del modello di esempio mostra la misura, questo è implementato dalle seguenti righe.

```
<tbody>
  {% for sample in data %}
  <tr>
    <td> {{sample[3]}} </td>
    <td> {{sample[0]}} </td>
    <td> <span id="mes-{{sample[3]}}">{{sample[1]}}</span>
      <input style="display:none" name="edit-{{sample[3]}}" id="edit-{{sample[3]}}" value="{{sample[1]}}" </td>
    <td>  </td>
    <td> <button name="edit" id="editb-{{sample[3]}}"
      value="{{sample[3]}}">Edit </button>
    <button name="save" id="save-{{sample[3]}}" value="{{sample[3]}}" style="display:none">Save </button>
```



```

<button name="cancel" id="cancel-{{sample[3]}}" value="{{sample[3]}}" style="display:none">Cancel </button>
</td>
</tr>
{% endfor %}
</tbody>
</table>
{% endif %}
<a href="/download-excel" download="Export-Excel.xlsx">Export in Excel</a>

```

Fondamentalmente, esegue un'iterazione su tutte le misure (oggetto campioni) e crea una riga in una tabella.

22. Conclusioni

Questo progetto è stato sfruttato in due esperimenti descritti in “RIFERIMENTO Documento Esperimenti” (vedi Figura 20)





Romuas Data


#	Date Time	Measure Image
1	2022 July 27 09:11:56 48.03	 <input type="button" value="Edit"/>

Figura 20 Risultati del progetto: a sinistra un esempio di configurazione, a destra la pagina Web Sample che riporta le misure acquisite.

Progetto 3: Accendere un led verde su misura corretta

<p>DIFFICULTY</p>	<p>Hardware</p> <p>Easy Medium Hard</p>  <p>Software</p> <p>Easy Medium Hard</p> 
--------------------------	---



SOFTWARE REQUIREMENTS	Arduino IDE installed on Windows, Linux or macOS, Tesseract Engine
HARDWARE REQUIREMENTS	Requisiti del progetto 2 e inoltre: <ul style="list-style-type: none">- Arduino Nano 33 IOT³- RGB led- 3 resistors (220Ω)
TEMPO	 circa 30 minuti se il progetto 2 è già stato completato

Questo progetto è un'estensione del Progetto 2: cattura una misura alla pressione di un pulsante. Il servizio web OCR è modificato per gestire una soglia sulle misure riconosciute e se la misura è nell'intervallo valido specificato viene inviato un messaggio ad una scheda IOT Arduino Nano per cambiare il colore in verde di un led RGB, e viceversa quando il la misura è al di fuori del range valido il colore del led diventa rosso. Nella Figura 21 è mostrata una panoramica di alto livello di questo progetto.

³ As alternative, you can use other boards including WI-FI connectivity

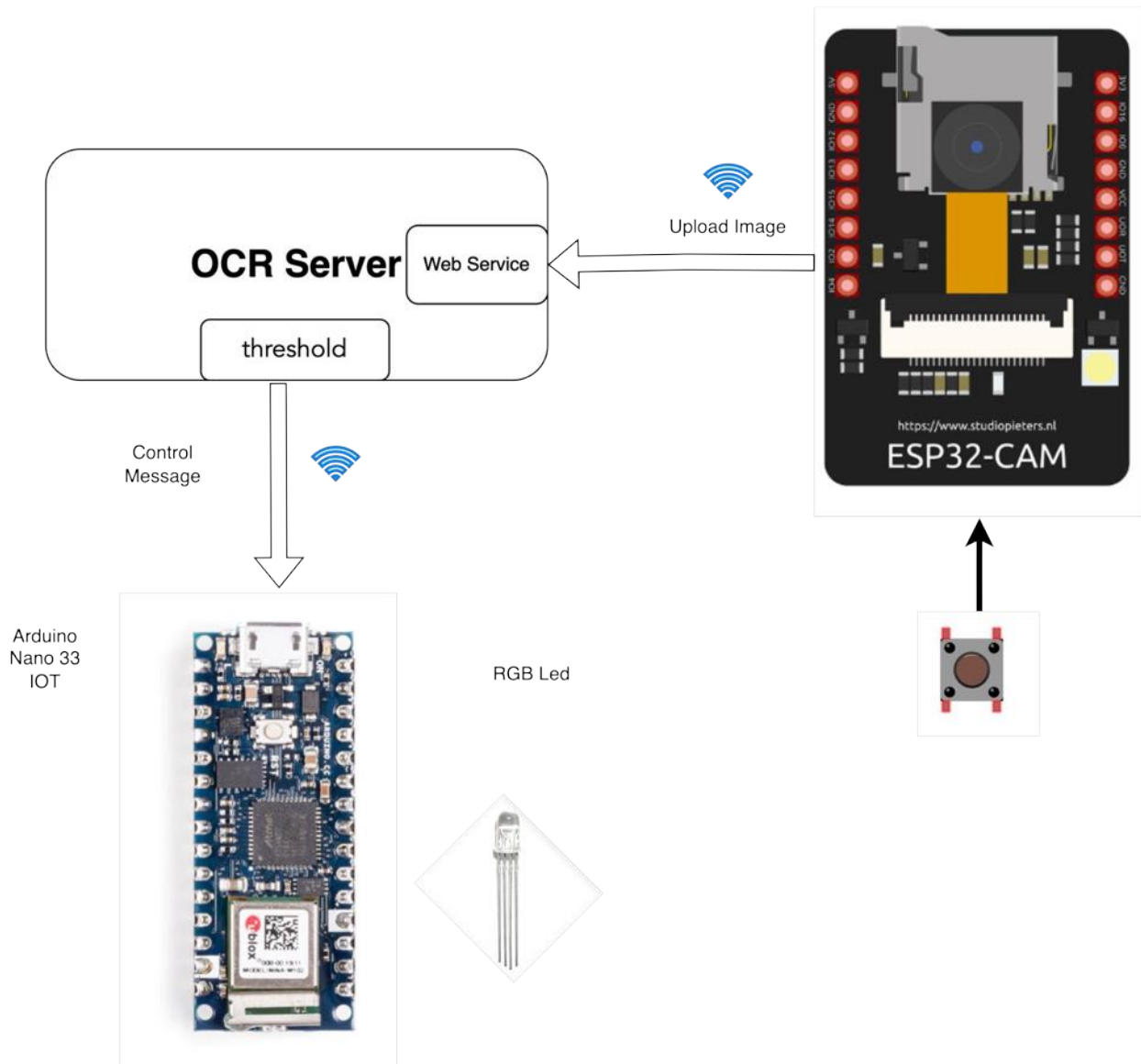


Figure 21 Project 3 high-level overview

1. Come prima fase del progetto è necessario eseguire il cablaggio come illustrato in Figura 22 e Figura 23:
 - Collegando il pin Arduino Nano 33 IOT D2 al primo ramo (ROSSO) del led RGB
 - Collegando il pin D3 di Arduino Nano 33 IOT alla terza gamba (VERDE) del led RGB
 - Collegando il pin Arduino Nano 33 IOT D5 alla quarta gamba (BLU) del led RGB
 - Collegando il pin IOT GND di Arduino Nano 33 al secondo ramo (catodo) del led RGB

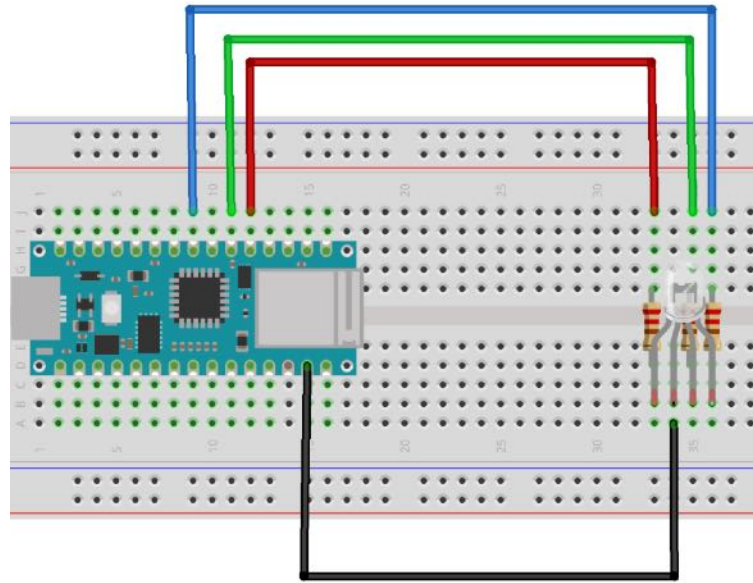


Figura 22 Progetto 3 - Vista breadboard

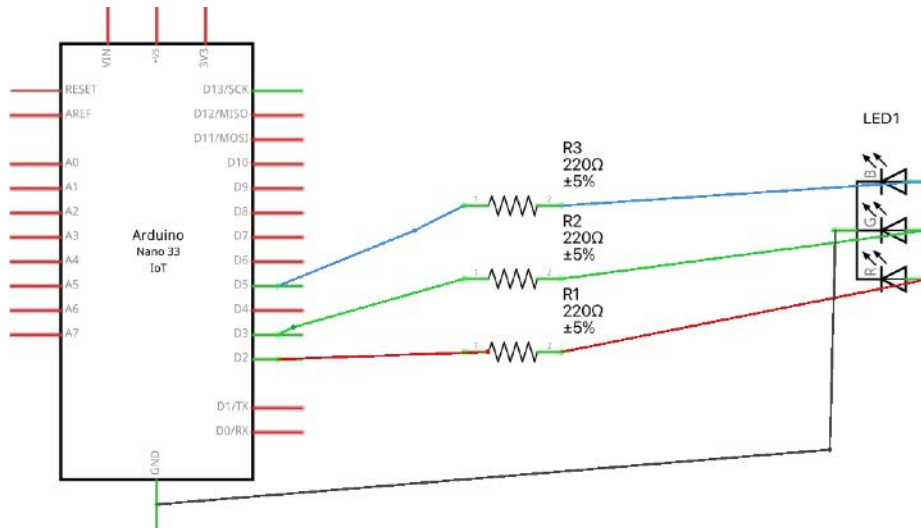
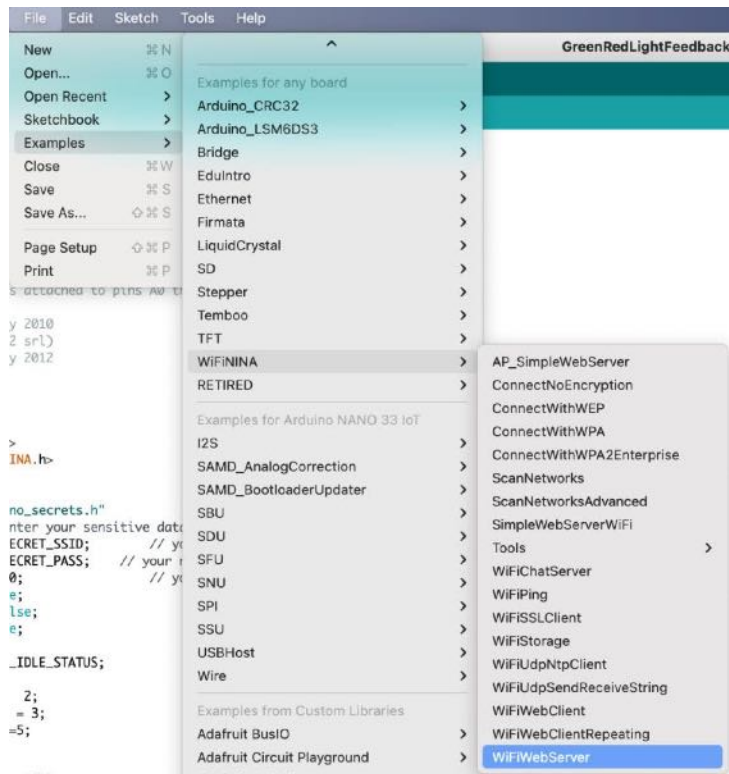


Figura 23 Vista logica del progetto 3

2. Scrivi lo schizzo.

Suggeriamo di partire dall'esempio WiFi NINA □ WiFiWebServer.



3. Modifica arduino_secrets.h con SSID e password della tua connessione Wi-Fi

```
GreenRedLightFeedback arduino_secrets.h
#define SECRET_SSID " "
#define SECRET_PASS " "
```

4. Definire le variabili per memorizzare i parametri di rete, la mappatura dei pin, lo stato della connessione e il server.

```
/*  
  WiFi Web Server  
  
  A simple web server that changes to color of a RGB led connected to D2 (R), D3 (G) and D5 (B)  
  
  */  
  
#include <SPI.h>  
#include <WiFiNINA.h>  
  
#include "arduino_secrets.h"  
////////please enter your sensitive data in the Secret tab/arduino_secrets.h  
char ssid[] = SECRET_SSID;        // your network SSID (name)  
char pass[] = SECRET_PASS;        // your network password (use for WPA, or use as key for WEP)  
int keyIndex = 0;                  // your network key index number (needed only for WEP)  
  
int status = WL_IDLE_STATUS;  
  
const int RED = 2; //pin D2  
const int GREEN = 3; //pin D3  
const int BLUE =5; //pin D5  
  
WiFiServer server(80);
```

5. Definire le funzioni per cambiare il colore del led in Verde, in Rosso o per spegnerlo

```
void lightOff(){  
  digitalWrite(RED, LOW);  
  digitalWrite(GREEN, LOW);  
  digitalWrite(BLUE, LOW);  
  Serial.println("OFF");  
  
}  
  
void lightRed(){  
  digitalWrite(RED, HIGH);  
  digitalWrite(GREEN, LOW);  
  digitalWrite(BLUE, LOW);  
  Serial.println("light Red");  
  
}  
  
void lightGreen(){  
  digitalWrite(RED, LOW);  
  digitalWrite(GREEN, HIGH);  
  digitalWrite(BLUE, LOW);  
  Serial.println("light Green");  
  
}
```

6. Definire la funzione di configurazione, che viene eseguita una volta all'avvio dell'applicazione (o reset)



Questa funzione

- inizializza il pin (righe da 54 a 57) e la connessione seriale (righe da 58 a 61).
- L'utilizzo della connessione seriale è facoltativo, è possibile commentare tali righe. Usiamo la connessione seriale per recuperare informazioni sull'IP assegnato alla scheda Arduino.
- Le righe dalla 64 alla 68 controllano lo stato del modulo WIFI.
- Le righe da 70 a 63 controllano se è disponibile un aggiornamento del firmware
- Le linee dalla 76 alla 85 eseguono la connessione WIFI e attendono che la scheda sia connessa
- Infine nella riga 87 vengono stampati i dettagli della connessione WIFI.

```
53 void setup() {
54   pinMode(RED, OUTPUT);
55   pinMode(GREEN, OUTPUT);
56   pinMode(BLUE, OUTPUT);
57   //Initialize serial and wait for port to open:
58   Serial.begin(9600);
59   while (!Serial) {
60     ; // wait for serial port to connect. Needed for native USB port only
61   }
62
63   // check for the WiFi module:
64   if (WiFi.status() == WL_NO_MODULE) {
65     Serial.println("Communication with WiFi module failed!");
66     // don't continue
67     while (true);
68   }
69
70   String fv = WiFi.firmwareVersion();
71   if (fv < WIFI_FIRMWARE_LATEST_VERSION) {
72     Serial.println("Please upgrade the firmware");
73   }
74
75   // attempt to connect to WiFi network:
76   while (status != WL_CONNECTED) {
77     Serial.print("Attempting to connect to SSID: ");
78     Serial.println(ssid);
79     // Connect to WPA/WPA2 network. Change this line if using open or WEP network:
80     status = WiFi.begin(ssid, pass);
81
82     // wait 10 seconds for connection:
83     delay(10000);
84   }
85   server.begin();
86   // you're connected now, so print out the status:
87   printWifiStatus();
88
89 }
90
```



7. Definire la funzione Stampa stato Wi-Fi.

```
void printWifiStatus() {  
  // print the SSID of the network you're attached to:  
  Serial.print("SSID: ");  
  Serial.println(WiFi.SSID());  
  
  // print your board's IP address:  
  IPAddress ip = WiFi.localIP();  
  Serial.print("IP Address: ");  
  Serial.println(ip);  
  
  // print the received signal strength:  
  long rssi = WiFi.RSSI();  
  Serial.print("signal strength (RSSI):");  
  Serial.print(rssi);  
  Serial.println(" dBm");  
}
```

8. Definire il ciclo principale.

La funzione loop gestisce la connessione del client, ogni volta che un client è connesso la richiesta viene analizzata per capire se richiede di cambiare il colore del led RGB o di spegnere il led.

I client devono comporre l'URL come segue:

http://{ID assegnato alla scheda Arduino}/green □ cambia il colore del led RGB in verde

http://{ID assegnato alla scheda Arduino}/rosso □ cambia il colore del led RGB in rosso

http://{ID assegnato alla scheda Arduino}/off □ spegni il led

L'analisi dell'url viene eseguita dalla riga 153 alla 162.

Il server restituisce comunque una pagina web con testo Romuas Client.

```
--  
94 void loop() {  
95  
96 // listen for incoming clients  
97 WiFiClient client = server.available(); // listen for incoming clients  
98  
99 if (client) { // if you get a client,  
100  
101 Serial.println("new client"); // print a message out the serial port  
102  
103 String currentLine = ""; // make a String to hold incoming data from the client  
104  
105 while (client.connected()) { // loop while the client's connected  
106  
107 if (client.available()) { // if there's bytes to read from the client,  
108  
109 char c = client.read(); // read a byte, then  
110  
111 Serial.write(c); // print it out the serial monitor  
112  
113 if (c == '\n') { // if the byte is a newline character  
114  
115 // if the current line is blank, you got two newline characters in a row.  
116  
117 // that's the end of the client HTTP request, so send a response:  
118  
119 if (currentLine.length() == 0) {  
120  
121 // HTTP headers always start with a response code (e.g. HTTP/1.1 200 OK)  
122  
123 // and a content-type so the client knows what's coming, then a blank line:  
124  
125 client.println("HTTP/1.1 200 OK");  
126  
127 client.println("Content-type:text/html");
```



```
128     client.println();
129
130
131     // the content of the HTTP response follows the header:
132
133     client.print("Romuas Client ");
134
135     // The HTTP response ends with another blank line:
136
137     client.println();
138
139     // break out of the while loop:
140
141     break;
142
143 } else { // if you got a newline, then clear currentLine:
144
145     currentLine = "";
146
147 }
148
149 } else if (c != '\r') { // if you got anything else but a carriage return character,
150
151     currentLine += c; // add it to the end of the currentLine
152
153
154
155
156
157     if (currentLine.endsWith("GET /green")) {
158         lightGreen();
159     }
160
161     if (currentLine.endsWith("GET /red")) {
162         lightRed();
163     }
164
165     if (currentLine.endsWith("GET /off")) {
166         lightOff();
167     }
168
169 }
170
171 // close the connection:
172 delay(1);
173 Serial.println("client disconnected");
174 }
```

9. È necessario inoltre modificare il codice Python (file sorgente app.py) implementato in Progetto 2: cattura una misura alla pressione di un pulsante per
10. Controllare se il valore della misura è in un intervallo valido
11. Notificare alla scheda Arduino la validità della misura
12. È necessario aggiungere una nuova libreria al web server Arduino

7 import requests

13. Aggiungere la dichiarazione delle variabili necessarie per la gestione delle soglie, della tolleranza e del server a cui inviare le notifiche

```
10 enable_external_feedback = True
11 threshold_value= 20
12 threshold_tollerance = 2.0
13 valid_measure_url = "http://192.168.1.80/green"
14 not_valid_measure_url ="http://192.168.1.80/red"
```

14. La linea 10 abilita l'invio della notifica della validità della misura catturata
15. La linea 11 imposta il valore iniziale della soglia
16. La linea 12 imposta la tolleranza iniziale espressa in percentuale

17. Le linee 13 e 14 impostano gli indirizzi a cui inviare rispettivamente le notifiche di misura valida e non valida.
18. E' necessario definire un metodo per la configurazione della soglia e della tolleranza

```
@app.route('/config-threshold', methods=['GET', 'POST'])
def config_threshold():
    global threshold_value
    global threshold_tollerance

    if request.method == 'POST':
        threshold_value = int(request.form.get('value'))
        threshold_tollerance = float(request.form.get('tollerance'))

    return render_template('config-threshold.html',
                           t_value=threshold_value,
                           t_tollerance = threshold_tollerance)
```

Questo metodo se richiamato in POST aggiorna i valori di soglia e tolleranza, e in ogni caso genera una pagina html con il template config-threshold.html

19. Definire il template config-threshold.html che permette di visualizzare e modificare la soglia e la tolleranza

```
17 <form method="post">
18   <p>
19     <label>Threshold</label>
20     <input type="number"
21           name="value"
22           required value="{{t_value}}"
23           min="10" max="200"/>
24   </p>
25
26   <p>
27     <label>Tollerance</label>
28     <input type="number"
29           name="tollerance"
30           required
31           value="{{t_tollerance}}"
32           min="0" max="100" step="0.1"/>%
33   </p>
34   <button type="submit">Update Threshold</button>
35 </form>
```

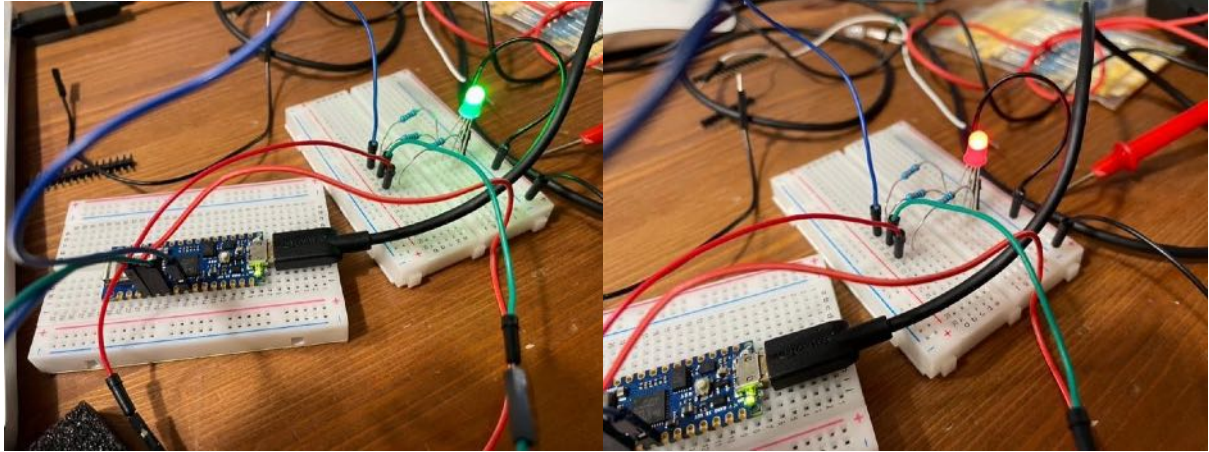
20. Definire la funzione per inviare la notifica della validità della misura catturata

```
125 #send the feedback to the client (ARDUINO NANO)
126 def feedback_measure(measure):
127     error = threshold_value * threshold_tolerance/100
128     feedback_url, valid = valid_measure_url, True if threshold_value - error <= measure <= threshold_value + error else not_valid_measure_url, False
129     r = requests.get(feedback_url)
130
131     if r.status_code == 200:
132
133         print('Feedback sent ',valid)
134     else:
135         print('Error on sending feedback')
136
```




21. Modificare la funzione che gestisce l'upload dell'immagine integrando la gestione della soglia

```
76 ocr = romuas_ocr(filePath)
77 text = ocr.show_ocr((560,215,230,120))
78 samples.append((datetime.now().strftime("%Y %B %d %H:%M:%S"),text,filePath,img_number))
79
80 if enable_external_feedback:
81     feedback_measure(int(text))
82
83 return "OK"
84
```

22. Esempio di funzionamento



Progetto 4 Inviare le misure catturate in Cloud

<p style="text-align: center;">DIFFICULTY</p>	<p style="text-align: center;">Hardware</p> <p style="text-align: center;">Easy Medium Hard</p>  <p style="text-align: center;">Software</p> <p style="text-align: center;">Easy Medium Hard</p> 
<p style="text-align: center;">SOFTWARE REQUIREMENTS</p>	<p>Arduino IDE installed on Windows, Linux or macOS, Tesseract Engine, Arduino Cloud</p>
<p style="text-align: center;">HARDWARE REQUIREMENTS</p>	<p>Project 3 requirements</p>
<p style="text-align: center;">TIME</p>	<p> about 20 minutes if the projects 3 has already been completed</p>

Questo progetto è un'estensione del **Progetto 3: Accendere un led verde su misura corretta** con l'aggiunta dell'integrazione con il servizio Cloud di Arduino per la memorizzazione delle misure catturate e per avere un controllo real-time dello stato dell'Universal Measure Reader.

La Figura 24 mostra un'architettura di alto livello del progetto: in pratica, è la stessa architettura del progetto 3, con la differenza che la notifica della validità della misura oltre essere inviata alla Scheda Arduino Nano è anche inviata nel cloud.

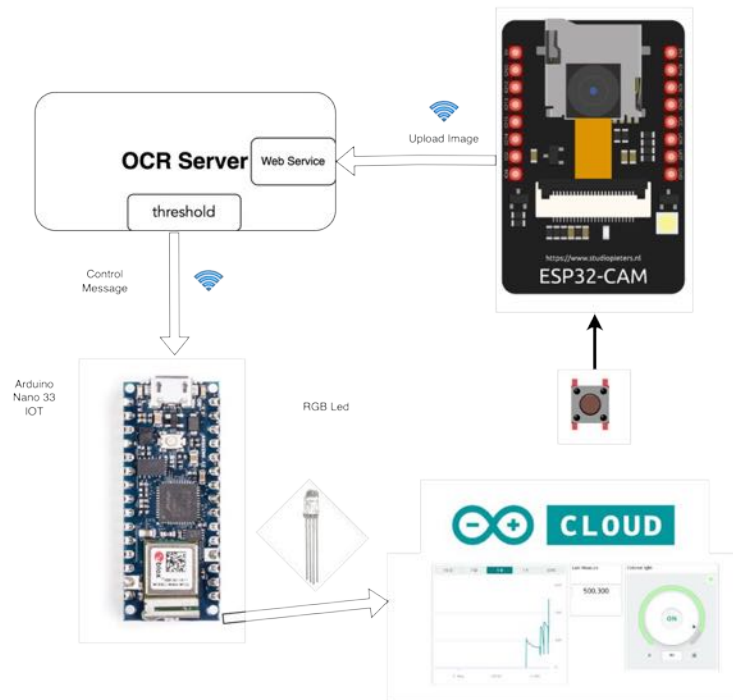
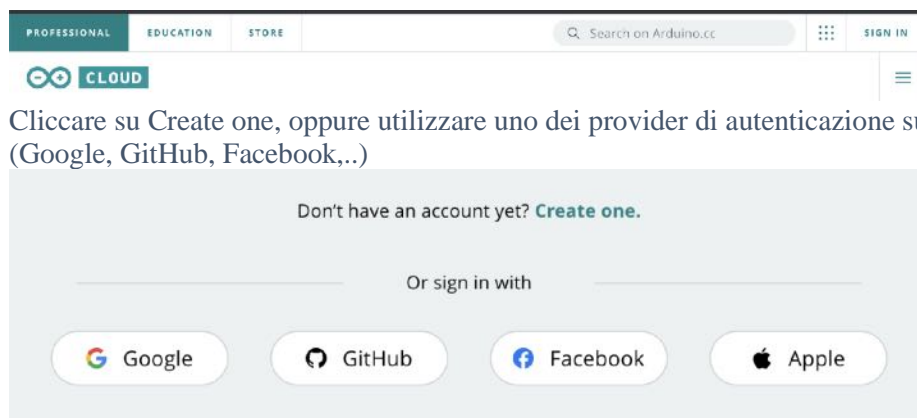


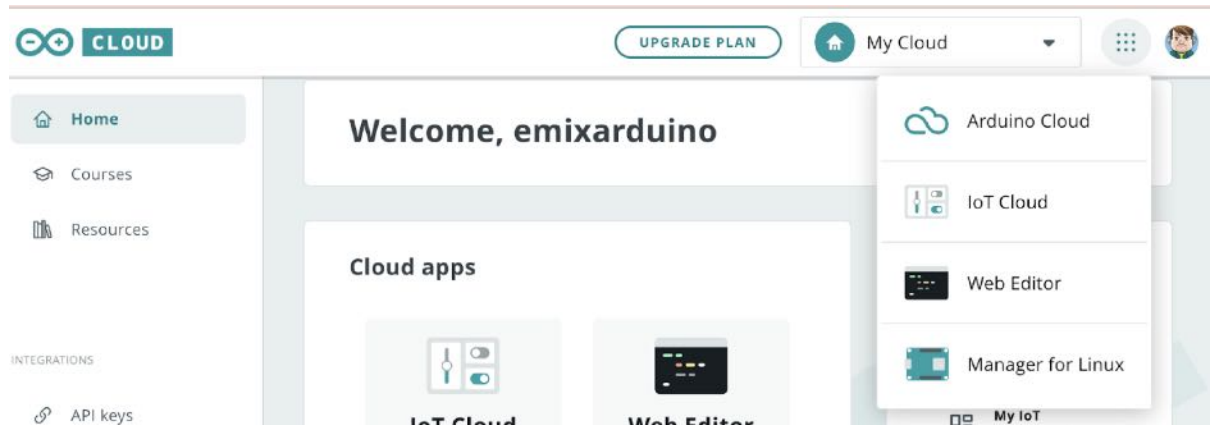
Figura 24 Vista d'insieme del Progetto 4

1. Per prima cosa è necessario completare il cablaggio relativo al progetto 3 (vedi Figura 22 e Figura 23)
2. Effettuare il login o la registrazione al servizio Arduino Cloud
 - o Accedere al sito <https://cloud.arduino.cc/>
 - o Cliccare su Sign In



- o Cliccare su Create one, oppure utilizzare uno dei provider di autenticazione supportato (Google, GitHub, Facebook,..)

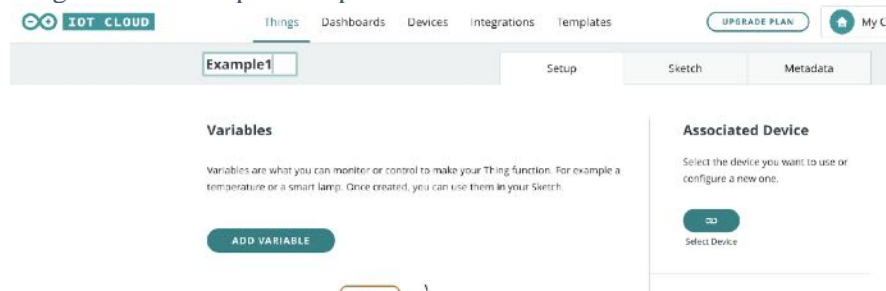
3. Dopo la registrazione ed il login è possibile accedere alla pagina principale del servizio Arduino Cloud e selezionare IoT Cloud del menù in alto a destra



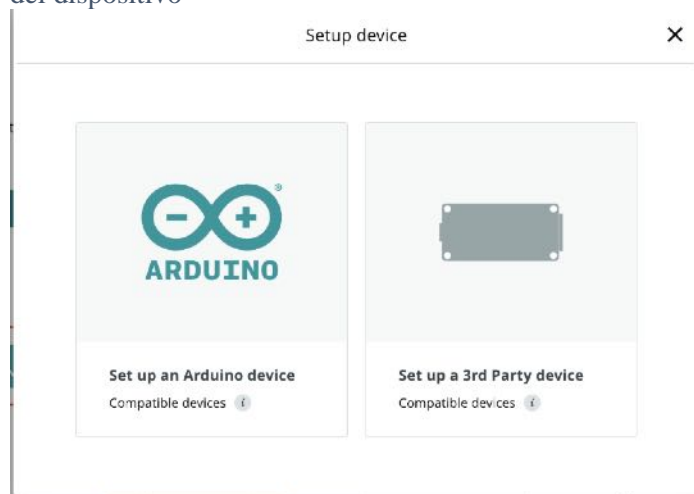
4. E' necessario creare un nuovo dispositivo che sarà il corrispettivo della nostra scheda Arduino reale.
- Cliccare sul menu Things
 - Cliccare il bottone Create



- Scegliere un nome per il dispositivo

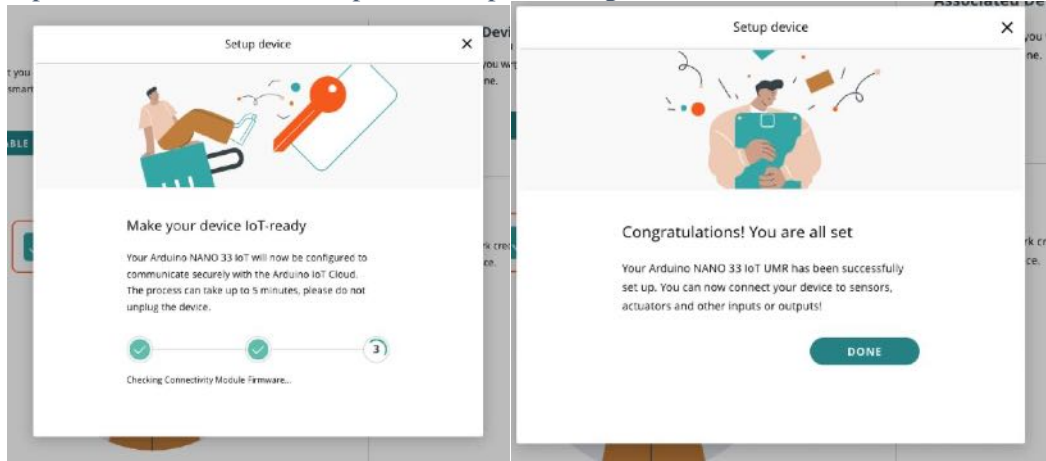


- Cliccare su Select Device nella barra laterale di destra. Verrà avviato il processo di selezione del dispositivo

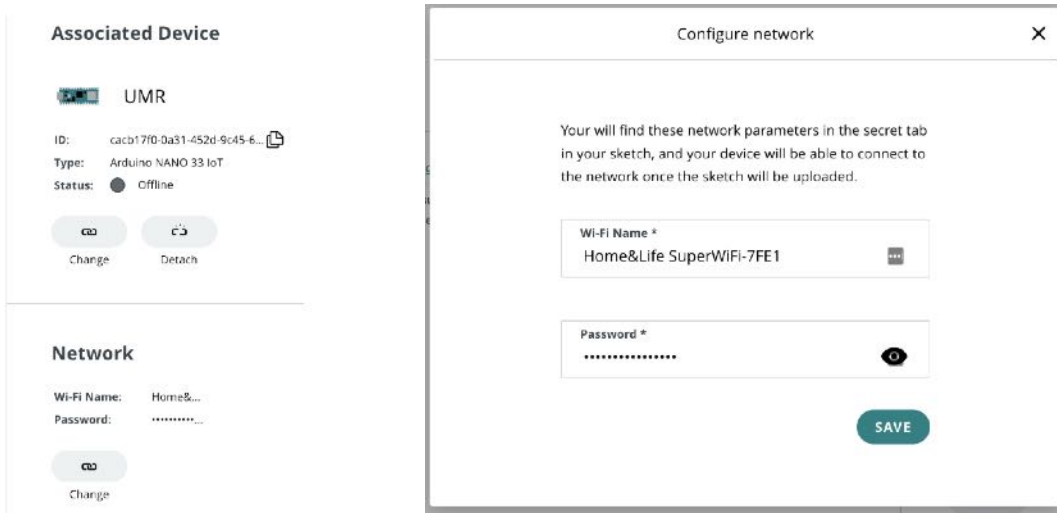


- È necessario connettere al proprio computer la scheda Arduino dopo aver preventivamente installato il software Arduino Agent

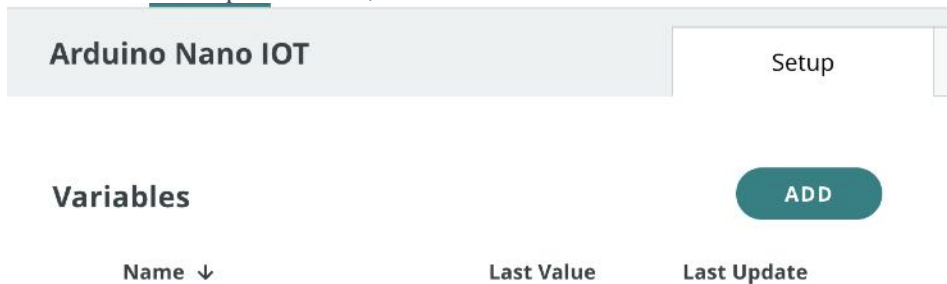
- Il processo di connessione dopo alcune operazioni genererà un file sketch



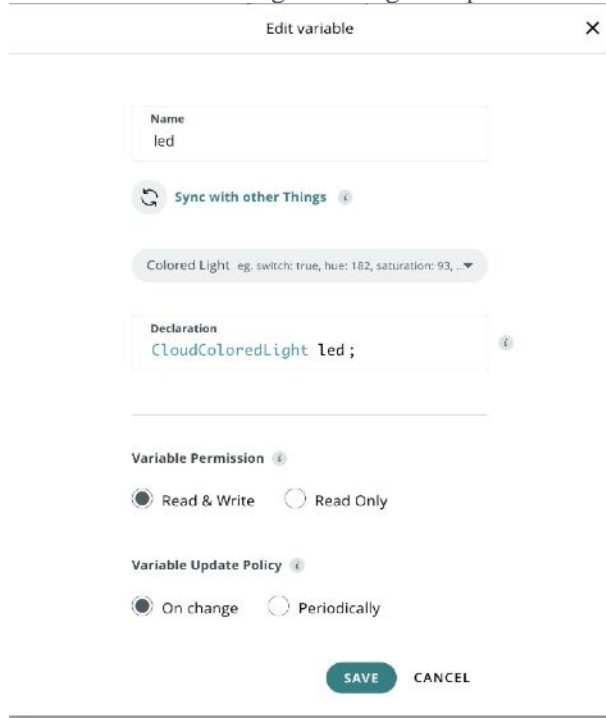
5. Nel tab Setup della pagina del dispositivo, cliccando sul pulsante **Change** , è possibile configurare i parametri di connessione alla propria Wi-Fi



6. È necessario creare due variabili per memorizzare sia lo misura corrente sia la sua validità (colore del led)
- Nella sezione Setup del device, cliccare sul bottone Add

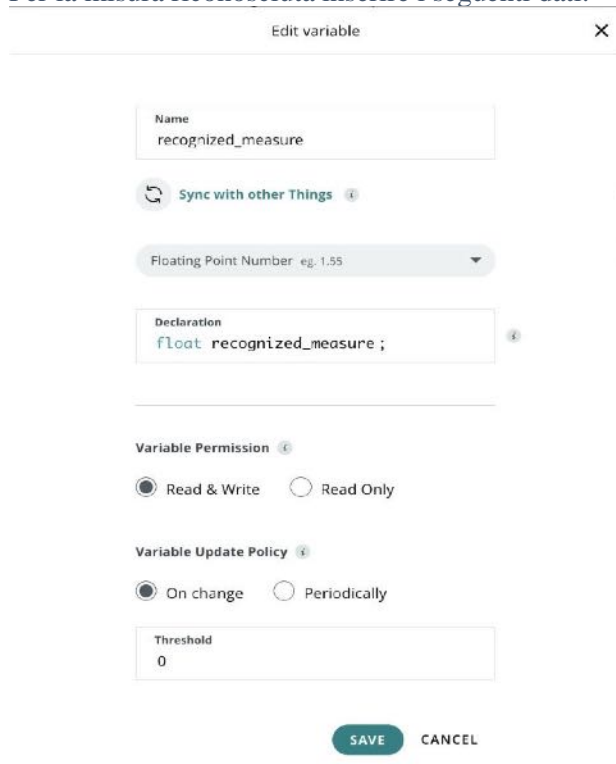


- Per la variabile Led scegliere i seguenti parametri:



The screenshot shows the 'Edit variable' dialog for a variable named 'led'. The 'Name' field contains 'led'. The 'Sync with other Things' option is checked. The 'Colored Light' dropdown is set to 'eg. switch: true, hue: 182, saturation: 93, ...'. The 'Declaration' field contains the code 'CloudColoredLight led;'. The 'Variable Permission' is set to 'Read & Write' (selected) and 'Read Only'. The 'Variable Update Policy' is set to 'On change' (selected) and 'Periodically'. At the bottom, there are 'SAVE' and 'CANCEL' buttons.

- Per la misura riconosciuta inserire i seguenti dati:

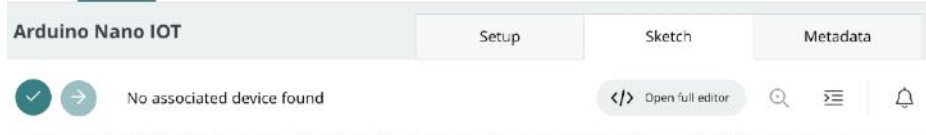


The screenshot shows the 'Edit variable' dialog for a variable named 'recognized_measure'. The 'Name' field contains 'recognized_measure'. The 'Sync with other Things' option is checked. The 'Floating Point Number' dropdown is set to 'eg. 1.55'. The 'Declaration' field contains the code 'float recognized_measure;'. The 'Variable Permission' is set to 'Read & Write' (selected) and 'Read Only'. The 'Variable Update Policy' is set to 'On change' (selected) and 'Periodically'. There is a 'Threshold' field containing the value '0'. At the bottom, there are 'SAVE' and 'CANCEL' buttons.

- Il sistema genera automaticamente un file Sketch
7. È necessario modificare il codice dello Sketch generato integrando il codice scritto per il progetto precedente.
- Poiché il codice è molto simile a quello del progetto precedente, analizzeremo solo le differenze.



- Accedere al tab Sketch e cliccare su Open full editor



- Verranno visualizzati 4 file



- Dovremmo modificare il primo
8. Osservando il file thingProperties.h si può notare che sono state incluse le due variabili che abbiamo creato nel passo 6. Modificare i valori delle due variabili in automatico invierà una notifica al Cloud Arduino.

```
12 float recognized_measure;
13 CloudColoredLight led;
14
15 void initProperties(){
16
17     ArduinoCloud.addProperty(recognized_measure, READWRITE, ON_CHANGE, onRecognizedMeasureChange);
18     ArduinoCloud.addProperty(led, READWRITE, ON_CHANGE, onLedChange);
19
20 }
```

9. Ritornando al file .ino (il nome generato dipende da quando si è generato il file), si devono importare le librerie richieste dal progetto 3 e dichiarare le variabili.

```
16 #include <WiFiNINA.h>
17 #include "thingProperties.h"
18
19
20 const int RED = 2; //pin D2
21 const int GREEN = 3; //pin D3
22 const int BLUE =5; //pin D5
23
24
25 WiFiServer server(80);
26
```

10. È necessario definire una funzione per spegnere del LED RGB

```
27 void lightOff(bool notify=true){
28     analogWrite(RED, LOW);
29     analogWrite(GREEN, LOW);
30     analogWrite(BLUE, LOW);
31     Serial.println("OFF");
32     if(notify)
33         led=ColoredLight(false,100,0,100);
34 }
```

Questa funzione ha un parametro opzionale, di default impostato a true, che permette di indicare se è necessario inviare la notifica di variazione al Cloud.

11. È necessario definire una funzione per modificare il colore del LED RGB

```
void setColor(int r,int g, int b){
    analogWrite(RED,r);
    analogWrite(GREEN,g);
    analogWrite(BLUE,b);
}
```

12. Utilizzando la funzione precedente, è necessario implementare due funzioni specifiche per impostare il colore del led a rosso e a verde



```
void lightRed(bool notify=true){
  analogWrite(RED, HIGH);
  analogWrite(GREEN, LOW);
  analogWrite(BLUE, LOW);
  Serial.println("light Red");
  if(notify)
    led=ColoredLight(true,0,100,100);
}

void lightGreen(bool notify=true){
  analogWrite(RED, LOW);
  analogWrite(GREEN, HIGH);
  analogWrite(BLUE, LOW);
  Serial.println("light Green");
  if(notify)
    led=ColoredLight(true,120,100,100);
}
```

13. Definire la funzione setup che verrà eseguita ogni volta che la scheda verrà avviata o resettata.

```
62 void setup() {
63   // Initialize serial and wait for port to open:
64   Serial.begin(9600);
65   // This delay gives the chance to wait for a Serial Monitor without blocking if none is found
66   delay(1500);
67
68   // Defined in thingProperties.h
69   initProperties();
70
71   // Connect to Arduino IoT Cloud
72   ArduinoCloud.begin(ArduinoIoTPreferredConnection);
73
74   /*
75    * The following function allows you to obtain more information
76    * related to the state of network and IoT Cloud connection and errors
77    * the higher number the more granular information you'll get.
78    * The default is 0 (only errors).
79    * Maximum is 4
80    */
81   setDebugMessageLevel(2);
82   ArduinoCloud.printDebugInfo();
83   // attempt to connect to WiFi network:
84
85   server.begin();
86 }
87
```



14. Definire la funzione per gestire le richieste HTTP (inviate dalla scheda ESP32-CAM)

```
88 void HandleClientRequest(){
89 // listen for incoming clients
90 WiFiClient client = server.available(); // listen for incoming clients
91
92 if (client) { // if you get a client,
93 Serial.println("new client"); // print a message out the serial port
94 String currentLine = ""; // make a String to hold incoming data from the client
95
96 while (client.connected()) { // loop while the client's connected
97
98 if (client.available()) { // if there's bytes to read from the client,
99
100 char c = client.read(); // read a byte, then
101 if (c == '\n') { // if the byte is a newline character
102
103 // if the current line is blank, you got two newline characters in a row.
104 // that's the end of the client HTTP request, so send a response:
105
106 if (currentLine.length() == 0) {
107
108 // HTTP headers always start with a response code (e.g. HTTP/1.1 200 OK)
109 // and a content-type so the client knows what's coming, then a blank line:
110
111 client.println("HTTP/1.1 200 OK");
112 client.println("Content-type:text/html");
113 client.println();
114 // the content of the HTTP response follows the header:
115 client.print("Romuas Client ");
116 // The HTTP response ends with another blank line:
117 client.println();
118 // break out of the while loop:
119
120 break;
121 } else { // if you got a newline, then analyze the line and clear currentLine:
122
123 if (currentLine.indexOf("GET /green")>=0) {
124 lightGreen();
125 }
126
127 if (currentLine.indexOf("GET /red")>=0) {
128 lightRed();
129 }
130
131 if (currentLine.indexOf("GET /off")>=0) {
132 lightOff();
133 }
134
135 if (currentLine.indexOf("GET /?value=")>=0) {
136 int rightIndex = currentLine.lastIndexOf(" ");
137 String measureText = currentLine.substring(12,rightIndex);
138 Serial.println("Measure="+measureText);
139 recognized_measure = measureText.toFloat();
140 }
141 Serial.println(currentLine);
142
143 currentLine = "";
144 }
145
146 } else if (c != '\r') { // if you got anything else but a carriage return character,
147
148 currentLine += c; // add it to the end of the currentLine
149 }
150 }
151 }
```

```
152
153 // close the connection:
154 delay(1);
155 Serial.println("client disconnected");
156 }
157
158
159 }
```

In aggiunta al Progetto 3 è necessario gestire un'ulteriore richiesta HTTP di tipo GET: `http://{ip scheda arduino}/?value`

Questa richiesta consente di inviare alla scheda Arduina il valore della misura riconosciuta. L'aggiornamento della variabile `recognized_measure` (linea 137) permette di notificare automaticamente la nuova lettura al Cloud Arduino.

15. Definizione del loop principale

```
162 void loop() {
163   ArduinoCloud.update();
164   // Your code here
165   HandleClientRequest();
166
167 }
```

16. È necessario definire la funzione `onLedChange` che verrà richiamata quando il valore del colore del led verrà modificato nella pagina web del Cloud Arduino.

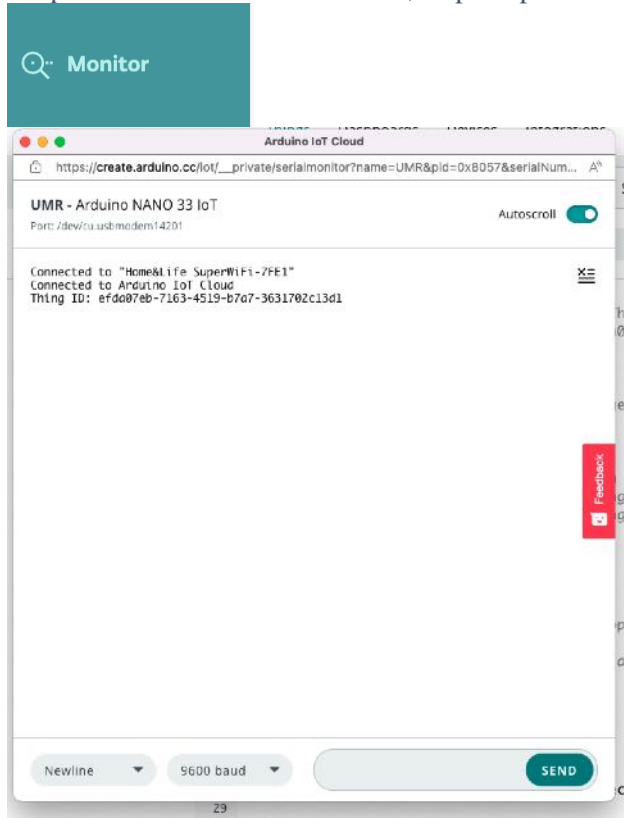
```
200 void onLedChange() {
201   auto colorValue = led.getCloudValue();
202   if(colorValue.swi){
203     uint8_t r,g,b;
204     colorValue.getRGB(r,g,b);
205     setColor(r,g,b);
206
207   }else{
208     lightOff(false);
209
210   }
211 }
```

Onestamente, la modifica del colore del led tramite Cloud non è strettamente necessaria per questo progetto, ma è qui illustrata per mostrare un esempio di integrazione a due vie (da dispositivo a Cloud e da Cloud a dispositivo).

17. Caricamento dello sketch sulla scheda Arduino

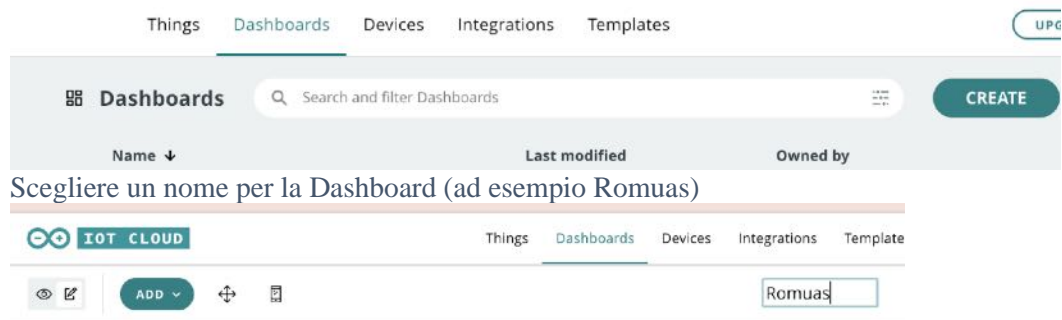
- È necessario premere il bottone con la freccia 

18. Dopo il caricamento dello Sketch, su può aprire la funzionalità di monitor



19. Ritorniamo alla piattaforma Iot cloud e creiamo una nuova Dashboard

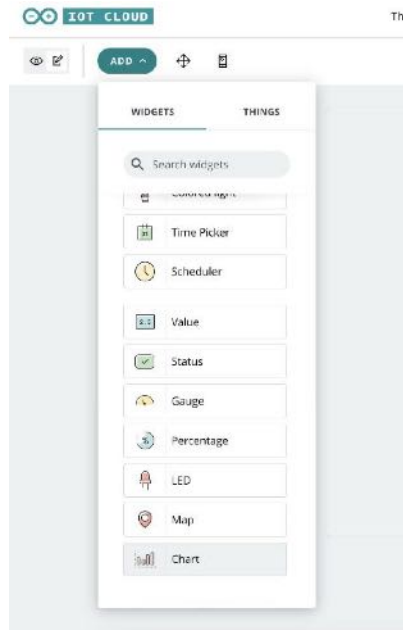
- All'interno del menu Dashboards cliccare sul bottone Create



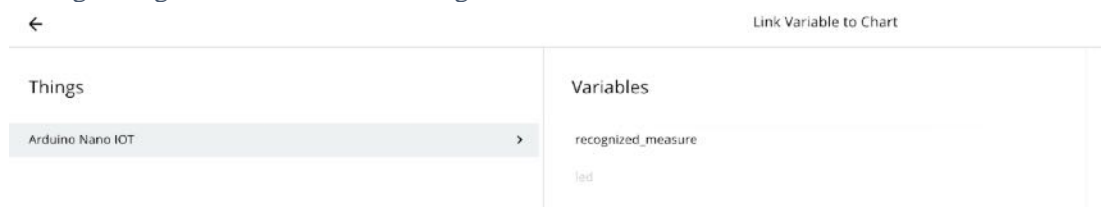
- Scegliere un nome per la Dashboard (ad esempio Romuas)

20. Creare un widget contenente un grafico

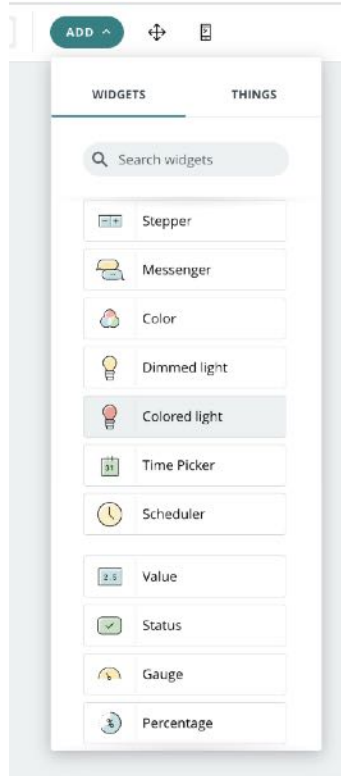
- Cliccare sul bottone Add



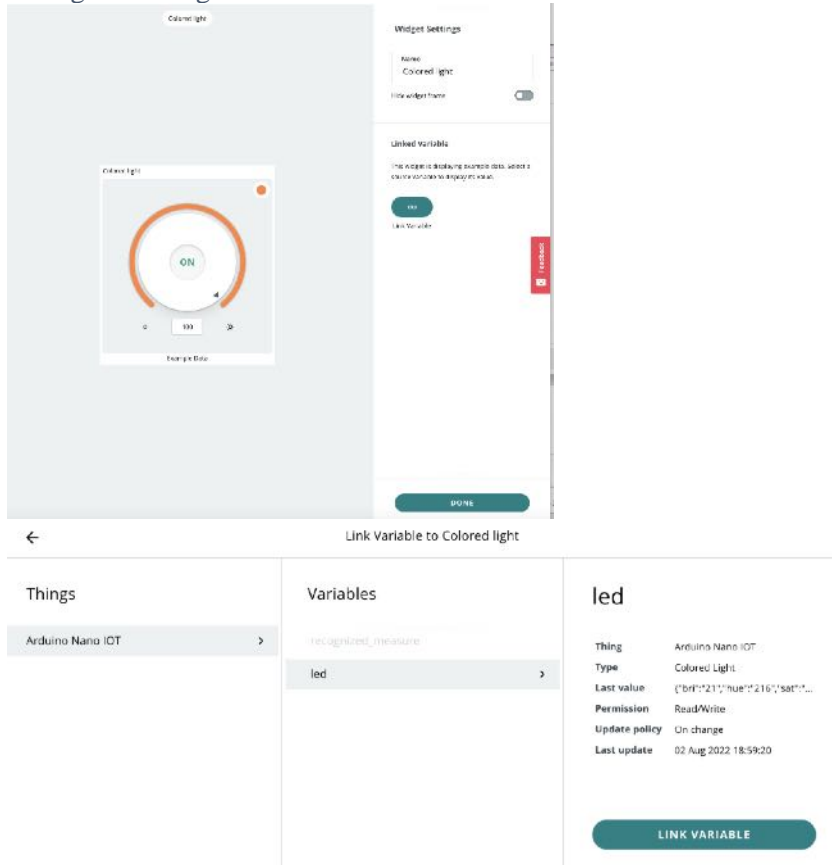
- Collegare il grafico alla variabile recognized_measure



- Il grafico riporterà i dati notificati dalla scheda Arduino al Cloud
21. Aggiungere il widget **Colored Light** che permette di gestire e visualizzare il led RGB
- Cliccare sul bottone Add e selezionare "Colored light"

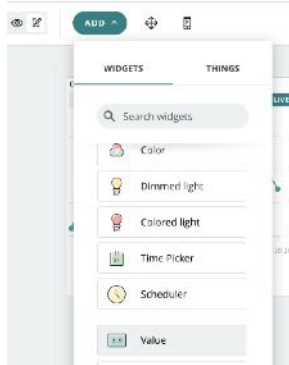


- Collegare il widget alla variabile "led"

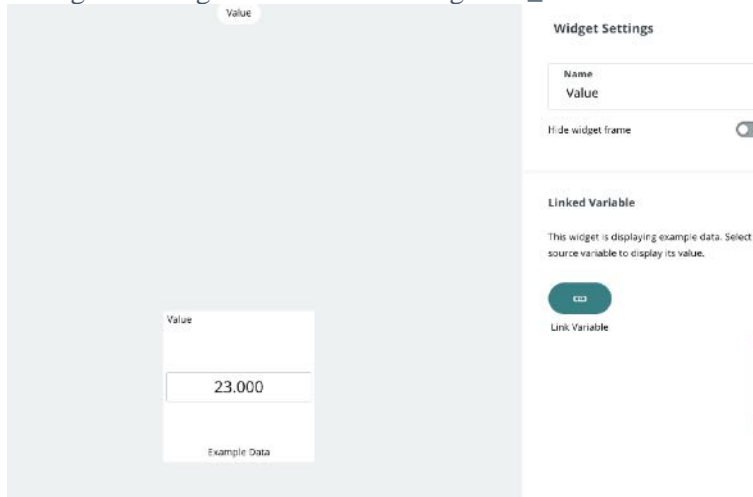


22. Aggiungere un widget per mostrare un valore

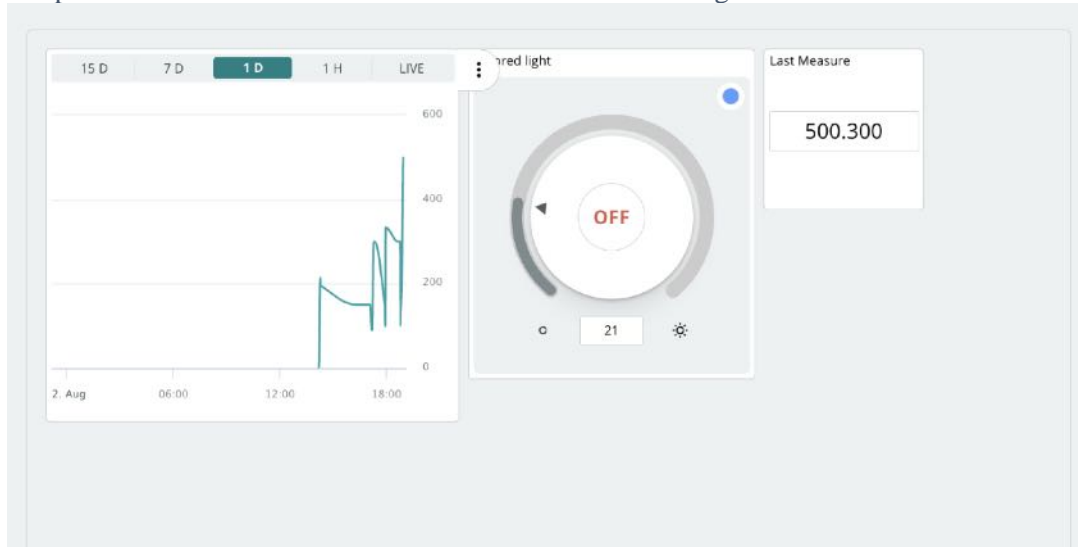
- Cliccare sul menu ADD e selezionare la voce "Value"



- Collegare il widget alla variabile recognized_measure



23. L'aspetto finale della dashboard dovrebbe essere simile al seguente:



I dati sono aggiornati in tempo e reale e la connessione tra la scheda Arduino e il Cloud è a due vie: se si modifica il colore del led nella dashboard il colore del led nella scheda sarà modificato nel dispositivo reale.

24. E' necessario modificare il serve ocr per inviare le misure riconosciute alla scheda Arduino Nano.

- Modificare il file app.py aggiungendo la variabile `measure_url`

```
15  measure_url = "http://192.168.1.80/?value=""
```

- Modificare la funzione `feedback_measure`

```
127 #send the feedback to the client (ARDUINO NANO)
128 def feedback_measure(measure):
129     error = threshold_value * threshold_tolerance/100
130     feedback_url, valid = valid_measure_url, True if threshold_value - error <= measure <= threshold_value + error \
131     else not_valid_measure_url, False
132     r = requests.get(feedback_url)
133
134     if r.status_code == 200:
135
136         print('Feedback sent ',valid)
137     else:
138         print('Error on sending feedback')
139
140     r = requests.get(measure_url+measure)
141
142     if r.status_code == 200:
143
144         print('Feedback sent ',valid)
145     else:
146         print('Error on sending feedback')
```

Le linee da 140 a 146 permettono l'invio della misura alla scheda Arduino Nano che notificherà la misura al Cloud.



ESEMPI DI ESPERIENZE DI LABORATORIO DA EFFETTUARE PRIMA E DOPO L'AGGIORNAMENTO

Sono stati condotti alcuni esperimenti per comprendere l'effetto dell'implementazione di alcuni comuni strumenti utilizzati in un laboratorio di chimica, come una bilancia da laboratorio e un riscaldatore.

I dati vengono raccolti manualmente e analizzati con una calcolatrice e alcuni programmi di calcolo.

La spiegazione di questi esperimenti è delineata di seguito.

AGGIORNAMENTO DI UNA BILANCIA DA LABORATORIO E DI UN RISCALDATORE

Una vecchia bilancia viene utilizzata per condurre due esperimenti, uno finalizzato a valutare la densità di alcuni materiali, il secondo a determinare la quantità di frazione organica di alcuni suoli. Il secondo esperimento viene condotto utilizzando sia la bilancia da laboratorio che il riscaldatore. Dopo la raccolta, i dati vengono analizzati. Un terzo esperimento viene condotto solo con il riscaldatore, dopo l'aggiornamento.

VALUTAZIONE DELLA DENSITÀ DEI DIVERSI MATERIALI.

Per valutare la loro densità apparente, sono stati pesati sei diversi materiali. Inizialmente è stato utilizzato un campione d'acqua per verificare l'accuratezza della bilancia e il volume che il becker può contenere. Un volume noto di materiale diverso è stato pesato sulla bilancia del laboratorio. I campioni erano acqua di rubinetto (H_2O), cloruro di sodio ($NaCl$), saccarosio ($C_{12}H_{22}O_{11}$), carbonato di litio ($LiCO_3$), polvere di rame (Cu), sabbia.

Il contenitore era un becker di vetro che poteva contenere un volume massimo di 50 cm^3 , graduato in 10 cm^3 , 25 cm^3 , 40 cm^3 e 50 cm^3 .

Prima di tutto, sulla bilancia da laboratorio sono stati pesati 40 ml di acqua per verificare l'accuratezza sulla bilancia da laboratorio (centesimo di grammo): il peso era perfettamente di 40 g. Infatti la densità dell'acqua è di circa 1g/cm^3 e la densità apparente non è da considerare, perché l'acqua è allo stato liquido.



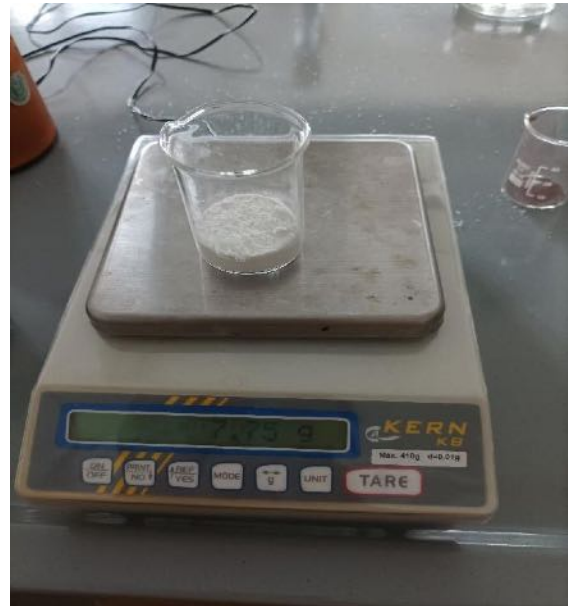


Fig. 1-2: campioni di polvere di rame e carbonato di litio

In un secondo momento i materiali sono stati pesati e la bilancia ha restituito i seguenti dati.

MATERIALE	VOLUME	PESO
Acqua di rubinetto	40 ml	39,96 g
Cloruro di sodio	10 ml	11,91 g
saccarosio	10 ml	9,40 g
Carbonato di litio	10 ml	7,75 g
Polvere di rame	10 ml	59,26 g
sabbia	10 ml	19,41 g

Dopo la raccolta dei dati, questi sono stati analizzati con un software (come Microsoft Excel), inserendo manualmente i valori. La densità è definita come massa divisa per volume o massa per unità di volume. Sono stati considerati valori di densità valutati sperimentalmente per ottenere la porosità dei materiali solidi.

La densità viene calcolata con la seguente formula: $\sigma = M/V$, dove σ è la densità, M è la massa e V è il volume. La densità è comunemente espressa in unità di grammi per centimetro cubo. Ad esempio, la densità dell'acqua è di quasi 1 grammo per centimetro cubo (g/cm^3) e la crosta continentale è di 2,7 grammi per centimetro cubo (g/cm^3).

La porosità è la quantità di spazio vuoto all'interno di un materiale solido secco e dipende dalla forma dei grani, che determina la forma e la dimensione dei pori, e la dimensione dei grani. Per quanto riguarda la valutazione della porosità si deve considerare la densità reale e la densità apparente (misurata con la bilancia). La porosità si calcola con la seguente formula:

$$P = (1 - P_b/P_d) * 100$$



Dove p è la porosità del campione, P_b è la densità apparente e P_d è la densità dei materiali (valore sperimentale).

Successivamente, vengono calcolati i valori di porosità, stimati con dati sperimentali sulla densità delle particelle (vedi allegati).

MATERIALE	PESO	VOLUME	DENSITA' APPARENTE	DENSITA' TABELLARE	POROSITA' %
	g	ml	$\sigma = M/V (g/cm^3)/g/cm^3$		$P = (1 - P_b/P_d) * 100$
tap water	39,96	40	1,00	1	0,1
sodium chloride	11,91	10	1,19	2,16	44,86
saccharose	9,40	10	0,94	1,59	40,86
lithium carbonate	7,75	10	0,78	2,1	63,10
copper powder	59,26	10	5,93	8,9	33,42
sand	19,41	10	1,94	2,7	28,11

ESPERIMENTO RIPETUTO DOPO L'AGGIORNAMENTO.

Dopo l'aggiornamento, i dati vengono nuovamente analizzati automaticamente con Arduino. L'operatore deve solo impostare la bilancia del laboratorio e metterci sopra i materiali.

Di seguito la descrizione, passo passo, del processo.

In primo luogo, l'operatore imposta Arduino per raccogliere i dati (posizionando la telecamera, che acquisisce le immagini). I campioni vengono pesati sulla bilancia da laboratorio, l'operatore clicca tre volte il pulsante a sinistra e il programma associato acquisisce i dati, visibili sul foglio Excel.

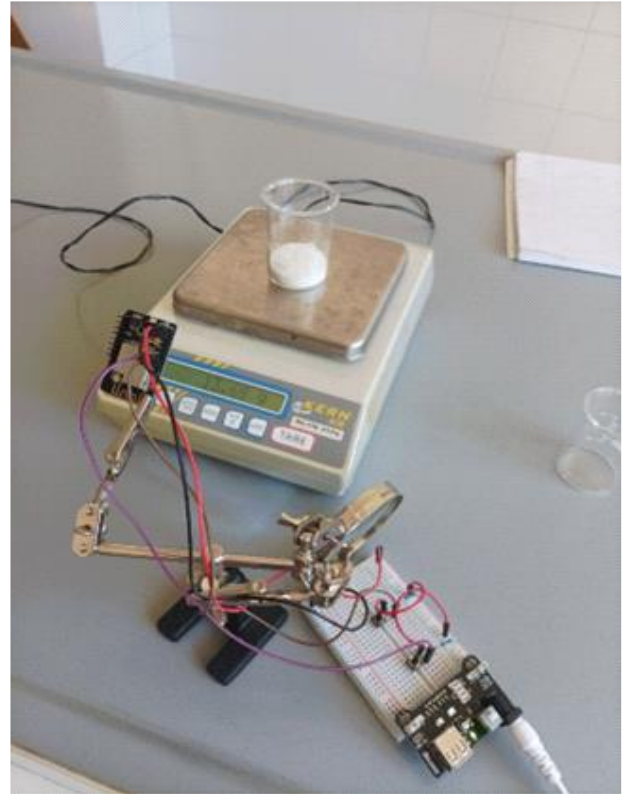
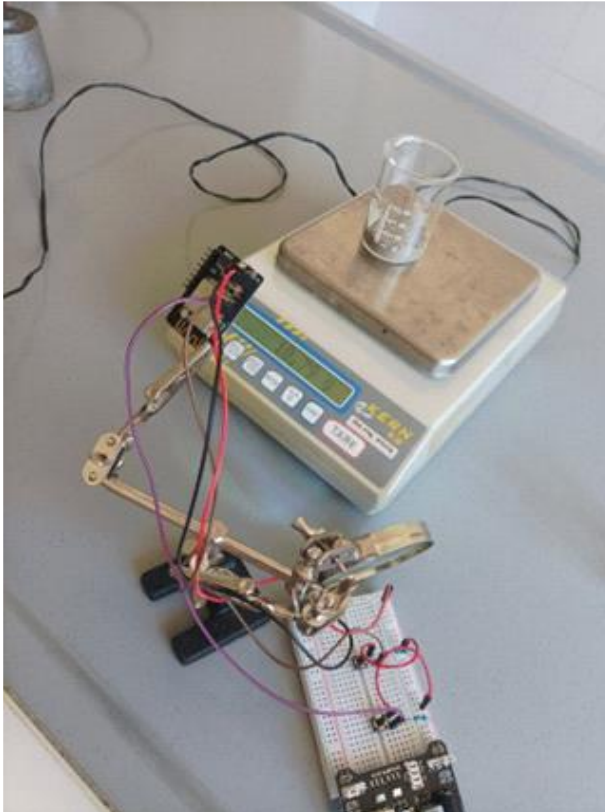


Fig. 3-4: campioni di sabbia e cloruro di sodio sulla bilancia, dopo l'aggiornamento

Elaborazione dei dati:

Esperimento	Peso(g)	Sostanza	
Measure 1	48,03	Acqua di rubinetto	1st measure
Measure 4	18,77	Sabbia	1st measure
Measure 7	13,63	Cloruro di sodio	1st measure
Measure 10	10,24	Saccarosio	1st measure
Measure 13	7,95	Carbonato di litio	1st measure
Measure 16	59,99	Polvere di rame	1st measure

Successivamente, i dati vengono analizzati e confrontati con i valori ottenuti con l'esperimento precedente.

MATERIALE	PESO	VOLUM E	DENSITA' APPARENTE	DENSITA' TABELLARE	POROSITA' %
	g	ml	$\sigma = M/V (g/cm^3)g/cm^3$		$P = (1 - P_b/P_d) * 100$
Acqua di rubinetto	48,04	50	0,96	1	--



Cloruro di sodio	13,63	10	1,36	2,16	36,90
saccarosio	10,24	10	1,02	1,59	35,60
Carbonato di litio	7,95	10	0,80	2,1	62,14
Povere di rame	59,99	10	6,00	8,9	32,60
sabbia	18,77	10	1,88	2,7	30,48

I valori sono leggermente diversi a causa della loro porosità, infatti dipende dalla dimensione dei grani.

Per concludere, la conduzione dell'esperimento con Arduino è più veloce perché l'operatore, dopo averlo impostato, deve solo preparare i campioni nei relativi contenitori. La raccolta dei dati è automatica e l'elaborazione avviene in un secondo momento.

Tale dispositivo è utile in caso di un elevato numero di misure consequenziali, perché l'operatore non deve interrompere il proprio lavoro per prendere nota dei dati ed evita eventuali errori di distrazione.

Bibliografia

1. **Santos, Sara.** ESP32 Camera Dev Boards Review and Comparison. *Maker Advisor*. [Online] 04 28, 2020. <https://makeradvisor.com/esp32-camera-cam-boards-review-comparison/>.
2. **Santos, Rui.** \$7 ESP32-CAM with OV2640 Camera. *Maker Advisor*. [Online] 9 6, 2021. [Cited: 06 23, 2022.] <https://makeradvisor.com/esp32-cam-ov2640-camera/>.
3. **Santos, Sara.** ESP-EYE: ESP32-based board for AI. *Maker Advisor*. [Online] 04 25, 2021. [Cited: 06 23, 2022.] <https://makeradvisor.com/esp-eye-new-esp32-based-board/>.
4. **Santos, Sara.** TTGO T-Journal ESP32 Camera Development Board Review. *Maker Advisor*. [Online] 04 26, 2021. [Cited: 06 23, 2022.] <https://makeradvisor.com/ttgo-t-journal-esp32-camera-board-review/>.
5. **Ayibiowu, Ayo.** TTGO T-CAMERA IS AN ESP32 CAM BOARD WITH OLED AND AI CAPABILITIES. *Electronics-Lab*. [Online] 01 30, 2019. [Cited: 06 23, 2022.] <https://www.electronics-lab.com/ttgo-t-camera-esp32-cam-board-oled-ai-capabilities/>.
6. **Upload Code to ESP32-CAM AI-Thinker using ESP32-CAM-MB USB Programmer.** *Random Nerd Tutorial*. [Online] [Cited: 06 24, 2022.] <https://randomnerdtutorials.com/upload-code-esp32-cam-mb-usb/>.
7. **Arduino Ide Downloads.** *Arduino.cc*. [Online] [Cited: 06 24, 2022.] <https://www.arduino.cc/en/software>.
8. **Editing the ESP32-CAM Camera Web Server HTML.** *Robot Zero One*. [Online] 04 25, 2019. [Cited: 06 25, 2022.] <https://robotzero.one/esp32-cam-custom-html/>.
9. **The technology for converting books and documents into electronic files.** *Software Sources*. [Online] [Cited: July 7, 2022.] <https://www.abbyy.co.il/?categoryId=72050&itemId=168963>.





10. Linvill, John G. *Optacon Final Report*. s.l. : Stanford Electronics Laboratories , 1973. Stanford, California 94305.

11. Installation. *Tesseract OCR*. [Online] 2022. <https://tesseract-ocr.github.io/tessdoc/Installation.html>.

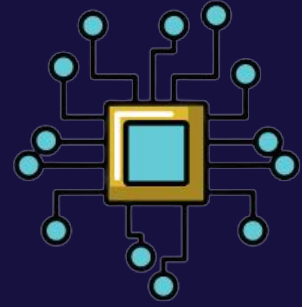
12. IEC. *Degrees of protection provided by enclosures (IP Code)*. 2013. ISBN 978-2-8322-1086-4.

13. Arduino case 3d print. *All3DP*. [Online] <https://all3dp.com/2/arduino-case-3d-print-3d-models/>.

14. Santos, Raul. ESP32-CAM Post Images to Local or Cloud Server using PHP (Photo Manager). *Random Nerd Tutorials*. [Online] <https://randomnerdtutorials.com/esp32-cam-post-image-photo-server/>.



Co-funded by the Erasmus+ Programme of the European Union



ROMUAS
KILLS
RDUINO
SING
MACHINES
OUTDATED
RECOVER

Innovazione sociale



C3

Expert program toolkit
PR1

Le informazioni e le opinioni contenute in questa pubblicazione sono quelle degli autori e non riflettono necessariamente l'opinione ufficiale dell'Unione Europea. Né le istituzioni e gli organi dell'Unione Europea né chiunque agisca per loro conto possono essere ritenuti responsabili dell'uso che potrà essere fatto delle informazioni ivi contenute.





Innovazione sociale

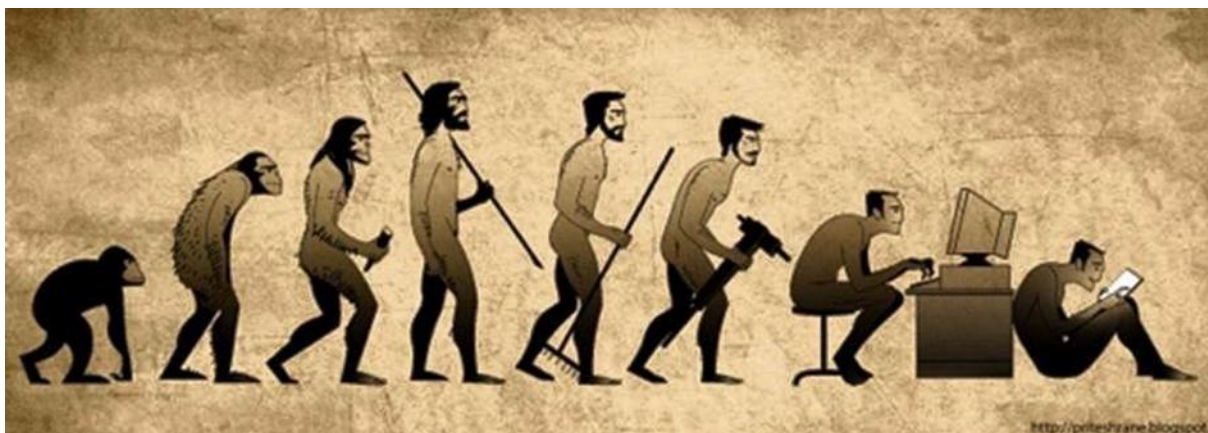
1. Introduzione
2. Breve sviluppo storico della tecnologia
3. Abilità sociali
 - 3.1. Assertività
 - 3.2. Abilità comunicative
 - 3.3. Sviluppo delle abilità sociali
4. Impatto della tecnologia sulle abilità sociali
5. Metodi interattivi per lo sviluppo delle abilità sociali
6. Conclusione
- Bibliografia



"L'aspetto più triste della vita in questo momento è che la scienza raccoglie conoscenza più velocemente di quanto la società raccolga saggezza" Isaac Asimov

1. Introduzione

2,5 milioni di anni fa, le persone hanno iniziato a fabbricare strumenti di pietra per conquistare la natura e sopravvivere in condizioni difficili. Ecco quando è iniziato lo sviluppo della tecnologia. Fin dall'inizio, l'uomo ha sviluppato la tecnologia per adattare la natura ai suoi bisogni e per sopravvivere. Nel tempo, lo sviluppo della tecnologia ha subito un'accelerazione perché l'uomo ha cercato di semplificare, facilitare e velocizzare numerosi processi. Dalla metà del XVIII secolo, quando iniziò la prima rivoluzione industriale, lo sviluppo della tecnologia accelerò, ma si diffuse anche in numerosi settori dell'attività umana. Oggi la tecnologia non è più vista solo come uno "strumento" subordinato all'uomo. L'uomo sviluppa e utilizza la tecnologia e la tecnologia influenza lo sviluppo e le azioni dell'uomo. Di questa relazione beneficiano sia l'uomo che la tecnologia, in una sorta di simbiosi.



Fonte:

Izvor:<https://www.facebook.com/TCHADDN/photos/a.464912620519513/464912600519515/?type=3>

È abbastanza chiaro come l'uomo influisca sulla tecnologia perché noi stessi assistiamo alla continua accelerazione del progresso in vari campi, dall'agricoltura e dall'industria, fino alla medicina, alle comunicazioni e all'industria spaziale. Tuttavia, una domanda che non è stata posta fin dall'inizio dello sviluppo della tecnologia è in che modo la tecnologia influisce sullo sviluppo psicologico e sui processi psicologici dell'uomo. Quali sono le aree dello sviluppo umano e i processi influenzati dalla tecnologia? La sua influenza è esclusivamente positiva o ha anche effetti negativi?

2. Breve storia della tecnologia

Fin dall'inizio, lo scopo della tecnologia era quello di facilitare la sopravvivenza, e successivamente di facilitare e accelerare la produzione. Durante le prime rivoluzioni industriali, a causa della crescente automazione della produzione, molti operai persero il lavoro. Ciò ha causato insoddisfazione tra di loro, ma li ha anche incoraggiati a "riqualificarsi". La riqualificazione ha comportato l'acquisizione di nuove competenze e conoscenze che hanno consentito a questi lavoratori di trovare un nuovo impiego e garantire la sopravvivenza a loro stessi e alle loro famiglie. Una conseguenza diretta dello sviluppo della tecnologia è la crescente complessità dei requisiti di determinate professioni. Allo stesso tempo, le aree di sviluppo tecnologico iniziarono a intrecciarsi sempre di più e lo sviluppo in un'area influenzò notevolmente lo sviluppo in un'altra area. Lo sviluppo della tecnologia cominciò ad essere sempre più influenzato dalle relazioni storiche e geopolitiche. La seconda guerra mondiale influenzò lo sviluppo del primo calcolatore elettronico perché era necessario eseguire numerose operazioni computazionali nella produzione di molti strumenti bellici. Con la fine della II Guerra mondiale, inizia la Guerra Fredda, che ha accelerato lo sviluppo della tecnologia, tutto a causa della reciproca concorrenza tra Occidente e Oriente. Uno dei settori in cui sono stati compiuti i maggiori progressi è quello dei sistemi di comunicazione che hanno consentito lo sviluppo dell'industria militare e spaziale. È stato lo sviluppo dei sistemi di comunicazione, in quel contesto, a gettare le basi per lo sviluppo della tecnologia mobile e di Internet, con il cui sviluppo la tecnologia è stata sempre più integrata nella vita di tutti i giorni.

Social media landscape 2022



Fonte: <https://fredcavazza.net/2022/06/14/panorama-des-medias-sociaux-2022/#more-38456>

Il primo SMS inviato ad un telefono cellulare risale al 1992 e conteneva un semplice augurio natalizio. A quel tempo, i computer desktop venivano sempre più utilizzati nel mondo degli affari e stavano lentamente entrando anche nell'uso domestico. Solo 3 anni dopo, il capo di Microsoft (allora il più grande produttore di computer desktop), annunciò che i loro prodotti si sarebbero rivolti completamente a Internet. Da allora è iniziata la rivoluzione digitale! Internet sta diventando disponibile a un numero crescente di persone e viene utilizzato per cercare informazioni varie, leggere e seguire notizie da tutto il mondo, ascoltare musica, guardare film, contattare amici e



parenti, ecc. Internet sta diventando un "luogo " dove le persone iniziano a trascorrere fino al 30% del proprio tempo libero (secondo Antolović, 2013). Allo stesso tempo, i telefoni cellulari stanno diventando sempre più "intelligenti", hanno sempre più funzioni che consentono loro di agire come minicomputer. Gli SMS vengono utilizzati per inviare saluti, pagare biglietti per parcheggi, partecipare a giochi a premi, ecc. Sono in fase di sviluppo anche servizi come Viber, WhatsApp, Messenger, ecc., che ci consentono di inviare messaggi illimitati a condizione che abbiamo accesso a Internet. Tuttavia, il vero fenomeno della comunicazione globale sono i social network come Facebook, Instagram, TikTok, ecc. Considerando che l'uomo è un essere sociale, non sorprende affatto che sia attratto dalla comunicazione sociale. Sui social cerchiamo di avvicinarci, connetterci e anche fare amicizia con nuove persone, scambiamo informazioni diverse. "La caratteristica principale di tutti i social network è costituita dai mezzi di comunicazione e dalle tecniche universali, nonché da un interesse comune che tiene insieme gruppi di persone e quindi dà popolarità a tutti i social network oggi" (Grbavac, 2014). Entro il 2020, le interazioni sociali tramite Internet, telefoni cellulari e social network riguarderanno principalmente la comunicazione verbale e scritta. Con la dichiarazione della pandemia di coronavirus e le raccomandazioni delle autorità mediche di ridurre al minimo i contatti sociali, il mondo online inizia a funzionare. Pertanto, anche le interazioni sociali vengono trasferite all'ambiente online. Allo stesso tempo, il bisogno di contatto sociale delle persone non è soddisfatto solo dalla comunicazione scritta, ma sono sempre più utilizzate varie piattaforme di comunicazione, come Zoom, MS Teams, Webex, ecc. L'uso di piattaforme di comunicazione consente una comunicazione più rapida con una o più persone, faccia a faccia.

Parallelamente allo sviluppo e al maggiore utilizzo di Internet, smartphone, social network e piattaforme di comunicazione, stanno emergendo sempre più ricerche che si occupano dell'impatto delle tecnologie digitali su vari aspetti dello sviluppo umano, in particolare sulle interazioni sociali e sulle abilità sociali.

3. Abilità sociali

Le abilità sociali sono "un insieme di abilità apprese che consentono a un individuo di interagire in modo competente e appropriato in un dato contesto sociale. Le abilità sociali più identificate nelle culture occidentali includono assertività, capacità di far fronte agli altri, comunicazione e amicizia, risoluzione di problemi interpersonali e capacità di regolare le proprie cognizioni, sentimenti e comportamenti". (APA Dictionary, 2022).





shutterstock.com · 1933402433

Fonte:

<https://www.shutterstock.com/image-vector/social-skills-we-use-everyday-interact-1933402433>

3.1. Assertività

L'assertività è definita come "uno stile di comunicazione adattivo in cui una persona esprime direttamente le proprie emozioni e bisogni, pur mantenendo il rispetto per gli altri. Una mancanza di assertività può contribuire alla depressione e all'ansia, mentre approcci disadattivi all'assertività possono manifestarsi con atteggiamenti aggressivi". (APA Dictionary, 2022).

Come si può vedere nella figura, due dimensioni sono cruciali per diversi stili di comunicazione: l'apertura della comunicazione e il rispetto per gli altri. Tra queste due dimensioni si distribuiscono 4 stili di comunicazione. Uno stile di comunicazione passiva, che mette i desideri e i bisogni degli altri prima dei propri, il che può essere frustrante. La persona non parla mai apertamente e onestamente di ciò che pensa e sente, ma in lui si accumulano sentimenti spiacevoli, come rabbia, collera, sentimento di inferiorità e lo stile di comunicazione passivo-aggressivo si riferisce all'espressione indiretta dei propri desideri e bisogni, ma in modo tale che la persona eviti la responsabilità. Lo stile di comunicazione aggressivo mette al primo posto i propri desideri e bisogni, non rispettando i desideri e i bisogni degli altri. Questo è uno stile di comunicazione inadeguato in cui l'obiettivo è "vincere" gli altri. Una persona con uno stile di comunicazione aggressivo si sente immediatamente potente e allevia la propria tensione, ma questi effetti sono a breve termine. A lungo termine, questo stile di comunicazione può portare a relazioni danneggiate con altre persone che potrebbero sentirsi ferite, arrabbiate e umiliate. Pertanto, è facile concludere che uno stile di comunicazione assertivo porta i maggiori benefici a noi, così come agli altri. Permette la soddisfazione dei bisogni perché esprimiamo apertamente, onestamente e chiaramente i nostri atteggiamenti, opinioni e sentimenti, e allo stesso tempo rispettiamo anche gli atteggiamenti, le opinioni e i sentimenti delle altre persone. Ci esprimiamo più chiaramente, quindi è più probabile che le altre persone ci capiscano meglio. Difendiamo noi stessi e ciò che facciamo, e quindi siamo più soddisfatti di noi stessi. L'assertività ci consente di affrontare le critiche e i conflitti in modo più efficiente, e allo stesso tempo ci rende più facile affrontare situazioni e relazioni difficili. Utilizzando uno stile di comunicazione assertivo, raggiungiamo relazioni amichevoli, aperte e oneste, con partner e affari. Infine, miglioriamo l'immagine di noi stessi.



3.2. Abilità comunicative

Le abilità comunicative sono abilità che consentono di ottenere una comunicazione efficace. Ci sono numerose abilità comunicative che sarebbe auspicabile padroneggiare, ma quelle chiave sono l'ascolto attivo, la parafrasi, il riassunto e il porre domande aperte, il ricevere e fornire feedback di qualità e l'assertività già descritta.

La comunicazione può essere verbale e non verbale. Verbale si riferisce a ogni messaggio che diciamo e/o scriviamo, cioè a quei messaggi che cerchiamo di trasmettere usando le parole. La comunicazione non verbale si riferisce alle espressioni facciali, al tono e al timbro della voce, alla parola e alla posizione del corpo, ecc. La comunicazione non verbale integra la comunicazione verbale in modo tale da darle un tono emotivo, in modo da poter interpretare il messaggio inviato in modo più semplice e preciso. Quando c'è una mancanza di comunicazione non verbale (ad es. utilizzando SMS o social network) allora è possibile interpretare non correttamente il messaggio inviato, il che può portare a conflitti e persino a danneggiare la relazione.

3.3. Sviluppo delle abilità sociali

Le abilità sociali vengono acquisite spontaneamente fin dalla prima infanzia, principalmente in famiglia. I metodi con cui vengono acquisite le prime abilità sociali sono l'imitazione e il metodo per tentativi ed errori. Man mano che il bambino cresce, interagisce con altre persone e gruppi (esempio: all'asilo con coetanei e insegnanti, genitori di altri bambini, ecc.). Attraverso queste interazioni, le abilità sociali vengono stabilite o modificate e il metodo con cui vengono più spesso adottate è l'apprendimento da modello.

Durante l'adolescenza, le interazioni sociali con i coetanei sono più comuni e "il bisogno di appartenere è un forte motivo per entrare a far parte di gruppi di pari" (Aavv, 2010). Le interazioni con i coetanei sono importanti per un sano sviluppo cognitivo, emotivo e sociale. Per ottenere il maggior numero possibile di interazioni sociali, gli adolescenti si iscrivono ai social network.

Nell'età adulta, le interazioni sociali con le persone dell'infanzia e dell'adolescenza vengono mantenute attraverso i social network o si estinguono. Tuttavia, le interazioni sociali che si realizzano continuando la formazione o avviando una carriera professionale sono in aumento. Le abilità sociali acquisite fino a quel momento vengono applicate o adattate e potenziate nel nuovo contesto sociale.

Nella vecchiaia, il numero di interazioni sociali diminuisce a causa della riduzione dello spazio vitale, della comunicazione con un numero inferiore di persone e di un numero inferiore di ruoli sociali. Tuttavia, è importante sottolineare che la ricerca ha dimostrato che la qualità delle relazioni intime è il fattore più importante per un senso di benessere e soddisfazione per la vita.

4. Impatto della tecnologia sulle abilità sociali

Considerando lo sviluppo storico della tecnologia, è evidente che il primo scopo della tecnologia era consentire la sopravvivenza, quindi facilitare e accelerare la produzione. La





tecnologia si è "insinuata" in varie aree dell'attività umana e le stesse abilità sociali hanno iniziato a essere influenzate dallo sviluppo di vari sistemi di comunicazione. L'obiettivo era (ma è tuttora) quello di trasmettere un messaggio chiaro e univoco.

Con lo sviluppo di Internet, della telefonia mobile, dei social network e delle piattaforme di comunicazione, un numero estremamente elevato di persone ha avuto accesso a diverse informazioni da tutto il mondo, ma anche la possibilità di comunicare e mantenere rapporti (stretti) con molte altre persone. I primi studi hanno mostrato che le persone che usano Internet più spesso comunicano meno spesso con i membri della famiglia, hanno una rete sociale più piccola di persone a cui sono vicine e sono più sole e più depresse (Kraut et al., secondo Antolović, 2013). I ricercatori hanno spiegato i risultati affermando che il tempo che le persone dedicano all'utilizzo di Internet è un sostituto del tempo trascorso nelle interazioni sociali in un ambiente offline. Un'altra spiegazione è che le persone usano Internet per sostituire connessioni sociali di qualità superiore con connessioni di qualità inferiore. La ricerca ripetuta ha dato risultati completamente opposti. In altre parole, è stato dimostrato che le persone che usano Internet più spesso hanno una cerchia più ampia di amici intimi e conoscenti, interagiscono più spesso con amici e familiari, sono più coinvolte nelle attività della comunità e sono più abili nell'uso del computer.

Man mano che le persone diventavano più abili con i computer ed esperti nell'uso di Internet, gli impatti negativi dell'utilizzo di Internet sono scomparsi. Tali risultati sono stati probabilmente ottenuti perché è aumentato il numero di utenti Internet, quindi è aumentata anche la rete sociale nell'ambiente online. La ricerca ha anche esaminato la connessione tra estroversione, supporto sociale e uso di Internet. I risultati indicano che le persone più socievoli e coloro che hanno un alto livello di supporto sociale hanno maggiori benefici sociali. Tuttavia, anche gli introversi potrebbero trarne vantaggio perché possono utilizzare nuove opzioni di comunicazione per creare nuove connessioni con le persone e ricevere supporto e informazioni utili.

Tuttavia, gli utenti Internet più comuni sono adolescenti e giovani adulti, soprattutto perché la comunicazione su Internet non avviene in tempo reale, ma dà loro il tempo necessario per pensare correttamente e modificare i messaggi. In altre parole, dà loro il controllo sulle interazioni sociali e consente una migliore auto-presentazione. Tuttavia, va considerato che nella comunicazione su Internet si usa più spesso la comunicazione verbale, cioè manca la comunicazione non verbale che dà il tono emotivo del messaggio, quindi sono possibili fraintendimenti dovuti a messaggi mal interpretati. Questi rischi sono ulteriormente aumentati quando si utilizzano i social network. Un fenomeno molto comune è il cyberbullismo, che si riferisce alla diffusione di informazioni false, spiacevoli o ostili su un'altra persona. Anche il sexting, che si riferisce all'invio o alla pubblicazione di foto o video in cui una persona è nuda o seminuda, è un fenomeno frequente. La "depressione da Facebook" si verifica a causa del confronto tra te stesso e la tua vita con le foto degli altri e le illusioni delle loro vite migliori. Molte persone ancora non capiscono che tutto ciò che pubblicano o fanno nell'ambiente online lascia una "impronta digitale" che può danneggiarli in futuro.

Tuttavia, è importante sottolineare che l'uso di Internet, così come dei social network, ha i suoi vantaggi. Vale a dire, i social network ci offrono la possibilità di connetterci con la famiglia e gli amici, stringere nuove amicizie, ma anche scambiare idee, sviluppare la creatività, partecipare a varie attività umanitarie, diventare attivi nella comunità locale e incontrare altri con cui condividere gli stessi interessi, attività che avremmo difficoltà a svolgere in un ambiente offline.





Consentono anche l'incontro di studenti al di fuori della scuola e lo scambio di idee su compiti diversi e il lavoro cooperativo su un progetto di gruppo.

5. Metodi interattivi per lo sviluppo delle abilità sociali

Le abilità sociali vengono sviluppate fin dalla tenera età con il metodo dell'imitazione (il bambino ripete l'azione ma non sa a cosa serve o quali sono le conseguenze) e con il metodo "prova e sbaglia". Pertanto, a quell'età, l'interazione più importante con il bambino è rappresentata da persone importanti (genitori, nonni, educatori) che dovrebbero essere consapevoli che il bambino li sta osservando e sta ripetendo ciò che fanno. Anche l'interazione con i coetanei è importante per i bambini perché in queste interazioni il bambino impara come affrontare le situazioni di conflitto e come combattere per se stesso. Le prime abilità sociali vengono acquisite in questi modi. A un'età leggermente più avanzata, i bambini iniziano a imparare seguendo il modello che è ancora molto spesso quello dei genitori, ma che può anche essere quello degli insegnanti. Imparando per modello, il bambino considera anche la conseguenza di un determinato comportamento, quindi se è positivo per il modello, è più probabile che lo ripeta, e se è negativo, cercherà di evitarlo. La ricerca ha dimostrato che i modelli che i bambini guardano in un ambiente offline hanno un'influenza maggiore rispetto a quando il modello mostra loro determinati comportamenti nell'ambiente online. Gli adolescenti sono i maggiori utilizzatori delle tecnologie digitali, ma imparano comunque secondo il modello, quindi lo stesso di cui sopra si applicherebbe anche a loro. Tuttavia, proprio a causa della grande quantità di tempo trascorso utilizzando le tecnologie digitali, gli adolescenti potrebbero esercitare alcune abilità sociali. Ad esempio, come scrivere un messaggio univoco e chiaro, come utilizzare i messaggi in prima persona ed essere assertivi, ecc. Queste abilità potrebbero essere acquisite utilizzando il cosiddetto apprendimento programmato. L'apprendimento programmato si riferisce a un sistema che contiene tutte le conoscenze necessarie che devono essere impartite allo studente, ma non vengono offerte immediatamente e tutte, ma lo studente padroneggia il contenuto al proprio ritmo. Allo stesso tempo, al termine di piccole unità, viene verificata la comprensione dello studente. In caso di comprensione, lo studente continua ad apprendere e, in caso di errori, torna indietro. In un tale sistema, è possibile simulare diverse situazioni sociali in cui allo studente è stato chiesto di rispondere come reagir, cosa fare, come rispondere, ecc. Nella tarda adolescenza e nella prima età adulta, la rete sociale cambia: o si continua con gli studi o si inizia lo sviluppo professionale. A quell'età, le nuove abilità sociali potrebbero essere acquisite guardando video, tutorial, ecc. Le persone a quell'età sono abbastanza mature per capire e possono ripetere e mettere in pratica i comportamenti mostrati. Anche loro potrebbero stabilire o modificare le abilità sociali come descritto in precedenza attraverso l'apprendimento programmato.

6. Conclusioni

Non abbiamo bisogno o non possiamo impedire lo sviluppo della tecnologia, ma quello che possiamo fare è adattarci meglio e lavorare sull'educazione, soprattutto dei bambini e dei giovani, per mantenere un rapporto simbiotico. In altre parole, l'uso della tecnologia per la quale non siamo tempestivi e adeguatamente istruiti può portare la tecnologia a prendere il controllo delle nostre vite, il che può avere conseguenze di vasta portata sulla nostra salute mentale, sullo sviluppo educativo e professionale e sulle relazioni interpersonali. Pertanto, è raccomandato, insieme all'educazione e alla moderazione, che la tecnologia (almeno in questa forma) non possa ancora sostituire lo sviluppo delle abilità sociali che si realizzano in un ambiente offline. Tuttavia, l'uso responsabile della tecnologia può contribuire a migliorare la qualità delle relazioni esistenti, stabilire nuove relazioni, sviluppare la creatività, scambiare idee, incoraggiare attività in un ambiente offline, ecc.





Metodi interattivi per sviluppare competenze innovative

1. Scopri il mio elemento

2. Dall'ispirazione all'azione

3. Bibliografia

1. Scopri il mio elemento

KEY WORDS: competenze, scelta professionale, talento, sviluppo personale, riconoscimento dei propri interessi

SARAI IN GRADO DI:

- sviluppare la consapevolezza delle proprie competenze e capacità
- identificare meglio i propri interessi e valori
- sviluppare la consapevolezza di sé
- definire gli obiettivi e le abilità della vita
- responsabilizzarsi e riconoscere l'importanza di responsabilizzare gli altri.

DOMANDE PER LA DISCUSSIONE:

- *Con quanta facilità hai trovato il tuo elemento? Ne hai più di uno?*
- *Qual è la tua conclusione?*
- *Se dovessi scegliere tra ciò in cui sei bravo e ciò che ami, cosa sceglieresti?*
- *Quanti di voi sceglierebbero una carriera solo per un impiego più semplice o uno stipendio migliore?*
- *Seguirai i consigli dei tuoi genitori o i tuoi interessi e perché?*
- *Quali carriere esistono oggi di cui i tuoi genitori non hanno sentito parlare?*
- *Dove potrebbero risiedere le tue capacità e i tuoi talenti? Come raggrupperesti le attività e le carriere delle persone?*

Scrivi alcune cose in ogni colonna (cose che ami, cose in cui sei bravo, cose che ti piace fare e cose per le quali hai supporto) e scopri dove si sovrappongono. Ciò che appare nella maggior parte delle colonne potrebbe essere il tuo elemento!





Mi piace...	Sono brava/o in...	Sono appassionata/o di...	Ho bisogno di aiuto con...	Ho l'opportunità di...
<p>Il concetto di "elemento" si basa sul libro <i>The Element: How Finding Your Passion Changes Everything</i> di Sir Ken Robinson (2010). L'elemento è il punto in cui il talento naturale incontra la passione personale. È qui che le persone si sentono maggiormente se stesse, ispirate e in grado di raggiungere i massimi livelli.</p>				

Tipi professionali di Holland

Realistico: lavorare con strumenti, prodotti, cibo, oggetti meccanici, disegni tecnici, piante, animali, avere abilità fisiche.

Investigativo: comprensione e risoluzione di problemi scientifici e matematici

Artistico – teatro, pittura, danza, musica, scrittura, artigianato

Sociale: aiutare gli altri, insegnare, prendersi cura, fornire informazioni

Intraprendente: gestire, persuadere, vendere idee e prodotti

Convenzionale: lavorare con numeri, organizzazione e direzione

Come possiamo aiutare i giovani a trovare la loro "scintilla" o vocazione nella vita?



Attraverso:

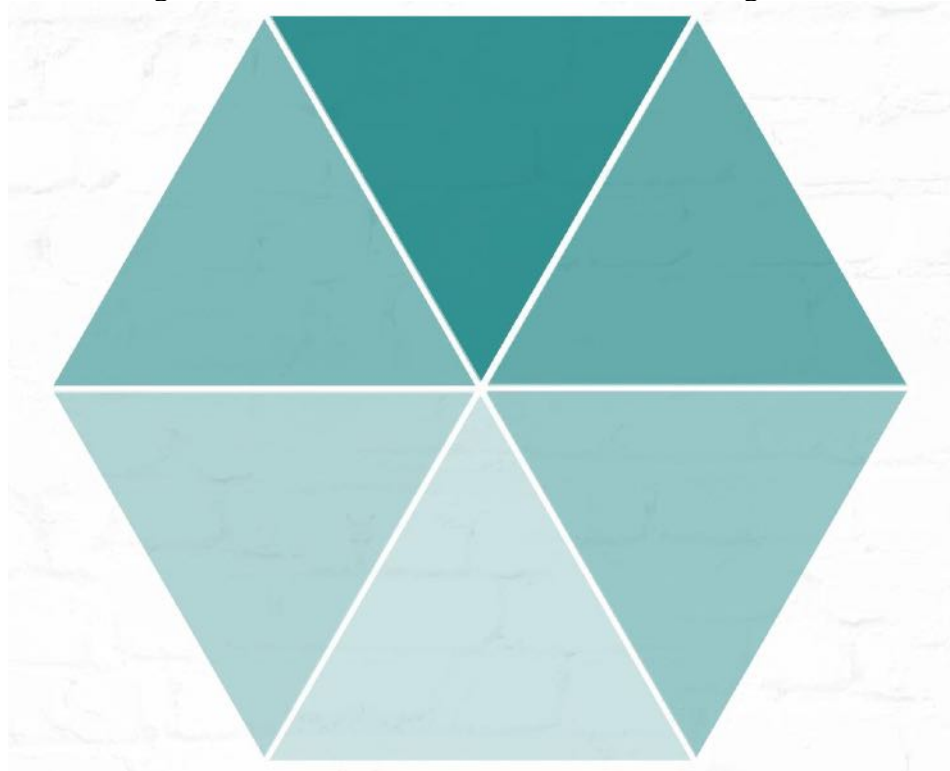
1. riconoscere e apprezzare la loro "scintilla";
2. creare possibilità per provare diverse attività.
3. ascoltare attentamente quando parlano di ciò che li interessa.
4. mostrare supporto.
5. fungere da modello e condividere i nostri interessi e le "scintille" e
6. includere queste attività nel curriculum.

(William Damon: The Path to Purpose: Helping Our Children Find Their Calling in Life. 2008.)

Esagramma di Holland

In ogni parte dell'esagramma scrivi abilità ed esperienze che hai in quell'area:

1. Realistica/creativa: in cosa sei abile?
2. Investigativa/Del pensiero: che tipo di problemi ti piace risolvere?
3. Artistica: cosa ti piace creare e come ti esprimi creativamente?
4. Sociale: chi e come ti piace aiutare?
5. Imprenditorialità: quali cose inizi da solo e altri ti seguono?
6. Convenzionale/Organizzativa: in che modo sei ordinato e ben organizzato?



Prendi in considerazione quanto sia importante per te ciascuna di queste attività e quanto ti piace.

Quindi prova a scegliere fino a tre aree che ritieni siano il tuo punto di forza. Quale carriera ti permetterebbe di mostrare tutte le tue qualità?



L'esagramma di Holland è ampiamente utilizzato per aiutare le persone a capire se stesse e guidare la loro scelta di carriera. È stato sviluppato dallo psicologo americano John L. Holland (1959).

2. Dall'ispirazione all'azione

PAROLE CHIAVE: colloquio, incoraggiamento, sostegno, mentoring

SARAI IN GRADO DI:

- conoscerti meglio
- praticare l'ascolto attivo
- migliorare le capacità di colloquio
- sviluppare la consapevolezza dei propri obiettivi e scegliere le priorità
- imparare a pianificare.

DOMANDE PER LA DISCUSSIONE:

- Come ti è piaciuto lavorare in coppia?
- Com'è stato fare domande e com'è stato rispondere e parlare di te stesso?
- Cosa hai realizzato e imparato?
- Pensi che sia importante pianificare o semplicemente lasciare che le cose seguano il loro corso? Quando è desiderabile fare in un modo e quando nell'altro?
- Cosa puoi fare quando non sei bravo in qualcosa e hai problemi a raggiungere il tuo obiettivo?

QUATTRO DOMANDE:

1. Qual è il tuo obiettivo ora? Scegli quello di cui vorresti parlare oggi.
2. Immagina il tuo futuro ideale! Continua a immaginare! Come ti vedi dopo aver raggiunto il tuo obiettivo? Che aspetto ha esattamente?
3. Come ci arrivi? Qual è il tuo modo di realizzarlo?
4. Cosa farai prima (a partire da oggi o domani) e chi ti aiuterà?

Un altro modo per iniziare questo seminario (a seconda del tempo che hai a disposizione) è l'attività chiamata Il fiume dell'apprendimento. Questa è un'attività creativa in cui disegni e colori su un grande foglio di carta un fiume che rappresenta la tua vita dalla nascita fino ad ora. Quindi, gli studenti usano simboli, illustrazioni o parole per contrassegnare le conoscenze o abilità importanti che hanno acquisito, i momenti cruciali e le persone da cui hanno imparato di più.





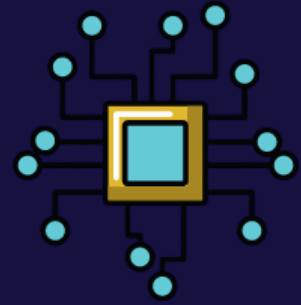
3 Bibliografija

1. <https://dictionary.apa.org/>
2. <https://fredcavazza.net/2022/06/14/panorama-des-medias-sociaux-2022/#more-38456>
3. <https://www.shutterstock.com/image-vector/social-skills-we-use-everyday-interact-1933402433>
4. Antolović, A. (2013): „Kako korištenje Interneta utječe na socijalne interakcije ljudi?“, <http://www.istrazime.com/socijalna-psihologija/kako-koristenje-interneta-utjece-na-socijalne-interakcije-ljudi/>, Portal i udruga Istraži Me, 8.10.2022.
5. Berk, L. E. (2008): „Psihologija cjeloživotnog razvoja“, Jastrebarsko, Naklada Slap
6. Fuller, J. R. (2015): „Them Impact Of Social Media Use On Social Skills“, <https://www.newyorkbehavioralhealth.com/the-impact-of-social-media-use-on-social-skills/>, New York Behavioral Health, 8.10.2022.
7. Grbavac, J. i Grbavac, V. (2014): „Pojava društvenih mreža kao globalnog komunikacijskog fenomena“, Media, culture and public relations, 5, 2014, 2, str. 206-219 (<https://hrcaj.srce.hr/file/188969>)
8. Skupina autora, (2010): „Trebamo li brinuti?“, <https://www.poliklinika-djeca.hr/publikacije/trebamo-li-brinuti/>, Poliklinika za zaštitu djece i mladih Grada Zagreba, 8.10.2022.
9. Župan, B., (2017): „Povijest i tehnologija: ono što ne znamo važnije je od onog što znamo“, <https://gkr.hr/Lab/Citaonica/Povijest-i-tehnologija-ono-sto-ne-znamo-vaznije-je-od-onog-sto-znamo>, Gradska knjižnica Rijeka, 8.10.2022. Dyer, Gregersen, and Christensen - The Innovator's DNA: Mastering the Five Skills of Disruptive Innovators
10. Harkavy Daniel - Cele 7 perspective ale liderilor eficienti Metode testate pentru decizii mai bune si o influenta mai mare, Amaltea, 2021
11. Roxana Carmen Cordoș, Bacali, L. și alții , Antreprenoriat, Ed. Todesco, Cluj-Napoca, 2008
12. Stanier Bungay Mickael, Coaching zi de zi, ed. Publica, 2017
13. Ana Munivrana, Domagoj Moric, Eli Pijaca Plavšić, Mario Bajkuša, Milana Rastovic, Vanja Kožic: workshop collection "start the change!", ACT Printlab, Zagreb 2021.





Co-funded by the
Erasmus+ Programme
of the European Union



ROMUAS
KILLS
RDUINO
SING
MACHINES
OUTDATED
RECOVER

Abilità imprenditoriali



C4

Expert program toolkit
PR1

Le informazioni e le opinioni contenute in questa pubblicazione sono quelle degli autori e non riflettono necessariamente l'opinione ufficiale dell'Unione Europea. Né le istituzioni e gli organi dell'Unione Europea né chiunque agisca per loro conto possono essere ritenuti responsabili dell'uso che potrà essere fatto delle informazioni ivi contenute.





1. Abilità imprenditoriali

4.1. Parte teorica

1. Considerazioni generali
2. Le caratteristiche e le competenze di un imprenditore
 1. Quali sono le capacità imprenditoriali?
 2. Le caratteristiche personali di un imprenditore
 3. Abilità interpersonali imprenditoriali
 4. Abilità critiche e di pensiero creativo per gli imprenditori
 5. Abilità e conoscenze imprenditoriali pratiche
3. Descrizione delle competenze comuni di un imprenditore
 1. Capacità di leadership
 2. Gestione della squadra
 3. Risoluzione dei problemi
 4. Processo decisionale
 5. Abilità comunicative
 6. Creatività e innovazione
 7. Gestione del tempo
 8. Gestione del rischioLeadership skills
4. L'imprenditorialità fa per te?
5. Bibliografia



1. Considerazioni generali

L'imprenditorialità richiede un insieme unico di competenze tra cui l'autodeterminazione, l'automotivazione e la perseveranza. Sia che tu abbia concettualizzato una soluzione a un problema noto o che tu abbia semplicemente deciso di lavorare per te stesso, vorrai avviare un'attività per consolidare la tua presenza online fin dall'inizio.

L'imprenditorialità è l'atto di avviare un'impresa nella speranza di ottenere un profitto. Tuttavia, la nostra moderna percezione dell'imprenditorialità si è evoluta nel riconoscere la sua capacità di risolvere problemi su larga scala e influenzare il cambiamento sociale. Gli imprenditori possono anche diventare leader di pensiero nei loro campi, sebbene questo non sia necessariamente un fattore distintivo, ma può essere un risultato naturale.

L'imprenditorialità svolge anche un ruolo importante nello sviluppo economico del nostro vivace mercato. Porta a migliori standard di vita e genera nuova ricchezza, oltre a un aumento dell'occupazione e del reddito nazionale. Gli imprenditori hanno anche l'opportunità di contribuire a colmare il divario digitale, il divario tra chi ha accesso a Internet e chi ne è sprovvisto.

Negli anni 2000, l'imprenditorialità è stata estesa dalle sue origini nelle imprese a scopo di lucro per includere l'imprenditoria sociale, in cui gli obiettivi aziendali sono ricercati insieme a obiettivi sociali, ambientali o umanitari e persino al concetto di imprenditorialità politica. L'imprenditorialità all'interno di un'impresa esistente o di una grande organizzazione è stata definita intrapreneurship e può includere iniziative societarie in cui le grandi entità scorporano ("spin-off") le organizzazioni sussidiarie.[1]

Gli imprenditori sono leader disposti a correre rischi e ad esercitare l'iniziativa, sfruttando le opportunità di mercato pianificando, organizzando e impiegando risorse,[2] spesso innovando per creare nuovi prodotti o servizi o migliorando quelli esistenti.[3] Negli anni 2000, il termine "imprenditorialità" è stato esteso per includere una mentalità specifica che ha portato a iniziative imprenditoriali, ad es. sotto forma di imprenditoria sociale, imprenditoria politica o imprenditoria della conoscenza.

1. Shane, Scott Andrew (2000). *A General Theory of Entrepreneurship: The Individual-opportunity Nexus*. Edward Elgar Publishing. ISBN 978-1-78100-799-0.
2. [Deakins & Freel 2009](#)
3. Johnson, D. P. M. (2005). "A Glossary of Political Economy Terms, 2005". Auburn University. Archived from [the original](#) on 20 July 2014.



2. Le caratteristiche e le competenze di un imprenditore

Con i progressi tecnologici e la domanda del mercato di nuove idee imprenditoriali e innovazione, la pratica dell'imprenditorialità sta crescendo rapidamente. Oggi, sempre più persone automotivate cercano di diventare completamente indipendenti nella loro vita professionale e aspirano a creare il proprio futuro.

1. Quali sono le capacità imprenditoriali?

Le capacità imprenditoriali sono quelle normalmente associate all'essere un imprenditore, anche se chiunque può svilupparle.

Essere un imprenditore di solito significa avviare e costruire la propria attività di successo, ma le persone con capacità imprenditoriali possono prosperare anche all'interno di organizzazioni più grandi.

Molti ricercatori hanno studiato le capacità imprenditoriali, ma non hanno trovato risposte definitive. Alcuni temi comuni sono:

- Caratteristiche personali.
- Abilità interpersonali.
- Capacità di pensiero critico e creativo.
- Abilità e conoscenze pratiche.

Indipendentemente da come la definisci, l'imprenditorialità non è facile.

Le sezioni seguenti esaminano ciascuna area di competenza in modo più dettagliato e analizzano alcune delle domande che dovrai porti se vuoi diventare un imprenditore di successo.

2. Le caratteristiche personali di un imprenditore

Hai la mentalità per essere un imprenditore di successo? Ad esempio, gli imprenditori tendono ad avere prospettive fortemente innovative e possono assumersi rischi che altri eviterebbero.

Esamina le tue caratteristiche personali, i tuoi valori e le tue convinzioni e poniti queste domande:

- **Ottimismo:** sei un pensatore ottimista? L'ottimismo è una risorsa e ti aiuterà a superare i momenti difficili che molti imprenditori sperimentano quando trovano un modello di business che funziona per loro.
- **Iniziativa:** hai iniziativa e avvii istintivamente progetti di risoluzione dei problemi o di miglioramento aziendale?





- **Guida e tenacia:** sei motivato ed energico? E sei pronto a lavorare sodo, per molto tempo, per realizzare i tuoi obiettivi?
- **Tolleranza al rischio:** sei in grado di assumerti dei rischi e prendere decisioni quando i fatti sono incerti?
- **Resilienza:** sei resiliente, in modo da poterti rialzare quando le cose non vanno come previsto? E impari e cresci dai tuoi errori e fallimenti?

3. Abilità interpersonali imprenditoriali

Come imprenditore, dovrai probabilmente lavorare a stretto contatto con gli altri, quindi è essenziale che tu sia in grado di costruire buoni rapporti con il tuo team, clienti, fornitori, azionisti, investitori e altri stakeholder.

Alcune persone sono più dotate in quest'area rispetto ad altre, ma puoi imparare e migliorare queste abilità.

I tipi di abilità interpersonali di cui avrai bisogno includono:

- **Leadership e motivazione:** puoi guidare e motivare gli altri a seguirti e realizzare la tua visione? E sei in grado di delegare il lavoro ad altre persone? Come imprenditore, dovrai dipendere dagli altri per andare oltre le prime fasi della tua attività: c'è davvero troppo da fare da solo!
- **Abilità comunicative:** sei abile in tutti i tipi di comunicazione? Devi essere in grado di comunicare bene per vendere la tua visione del futuro a un'ampia varietà di pubblico, inclusi investitori, potenziali clienti e membri del team.
- **Ascolto:** ascolti quello che gli altri ti dicono? La tua capacità di ascoltare e assorbire informazioni e opinioni può crearti o distruggerti come imprenditore. Assicurati di essere abile nell'ascolto attivo ed empatico .
- **Relazioni personali:** hai buone "capacità relazionali"? Sei consapevole di te stesso, bravo a regolare le tue emozioni e in grado di rispondere positivamente a feedback o critiche?
- **Negoziazione:** sei un forte negoziatore? Non solo devi negoziare prezzi favorevoli, ma dovrai anche risolvere le divergenze tra le persone in modo positivo e reciprocamente vantaggioso.
- **Etica:** tratti con le persone sulla base di rispetto, integrità, correttezza e fiducia? Puoi condurre eticamente? Troverai difficile costruire un'attività felice e produttiva se tratti con personale, clienti o fornitori in modo squallido.

4. Abilità critiche e di pensiero creativo per gli imprenditori

Come imprenditore, devi trovare nuove idee e prendere buone decisioni su opportunità e potenziali progetti.



Molte persone pensano che o nasci creativo o non lo sei. Ma la creatività è un'abilità che puoi sviluppare e ci sono molti strumenti disponibili per ispirarti.

- **Pensiero creativo:** sei in grado di vedere le situazioni da una varietà di prospettive per generare idee originali?
- **Risoluzione dei problemi:** avrai bisogno di valide strategie per risolvere i problemi aziendali che inevitabilmente si presenteranno.
- **Riconoscere le opportunità:** riconosci le opportunità quando si presentano? Riesci a individuare una tendenza? E sei in grado di creare un piano realizzabile per sfruttare le opportunità che identifichi?

5. Abilità e conoscenze imprenditoriali pratiche

Gli imprenditori hanno anche bisogno di solide capacità e conoscenze pratiche per produrre beni o servizi in modo efficace e per gestire un'azienda.

- **Impostazione degli obiettivi:** l'impostazione di obiettivi SMART (Specifici, Misurabili, Realizzabili, Rilevanti e Limitati nel tempo) concentrerà i tuoi sforzi e ti consentirà di utilizzare il tuo tempo e le tue risorse in modo più efficace.
- **Pianificazione e organizzazione:** hai i talenti, le competenze e le capacità necessarie per raggiungere i tuoi obiettivi? Puoi coordinare le persone per raggiungere questi obiettivi in modo efficiente ed efficace? Le forti capacità di gestione del progetto sono importanti, così come le capacità organizzative di base. E avrai bisogno di un piano aziendale coerente e ben congegnato e delle previsioni finanziarie appropriate.
- **Processo decisionale:** le tue decisioni aziendali dovrebbero basarsi su buone informazioni e prove e soppesare le potenziali conseguenze. Gli strumenti fondamentali per il processo decisionale includono l'analisi dell'albero decisionale, l'analisi della griglia e i sei cappelli per pensare.

Hai bisogno di conoscenze in molte aree diverse quando stai avviando o gestendo un'impresa, quindi preparati per un apprendimento serio!

Assicurati di includere:

- **Conoscenza aziendale:** assicurati di avere una conoscenza pratica delle principali aree funzionali di un'azienda: vendite, marketing, finanza e operazioni. Se non sei in grado di svolgere tutte queste funzioni da solo, dovrai assumere altri che lavorino con te e gestirli con competenza.



- **Conoscenza imprenditoriale:** come finanziarai la tua attività e quanto capitale devi raccogliere? Trovare un modello di business che funzioni per te può richiedere un lungo periodo di sperimentazione e duro lavoro.
- **Conoscenza specifica dell'opportunità:** comprendi il mercato in cui stai tentando di entrare e sai cosa devi fare per portare il tuo prodotto o servizio sul mercato?
- **Conoscenza specifica del rischio:** sai cosa serve per il successo di questo tipo di attività? E capisci le specifiche dell'attività che vuoi avviare?

Puoi anche imparare da altri che hanno lavorato a progetti simili a quelli che stai contemplando o trovare un mentore, qualcun altro che è già stato lì ed è disposto a guidarti.

2. Descrizione delle competenze comuni di un imprenditore

1. Capacità di leadership

Non esiste un modo univoco per descrivere le **grandi capacità di leadership**. Non sorprende che tu possa trovare diversi elenchi che descrivono le capacità di leadership che uno dovrebbe possedere per diventare un grande leader sul posto di lavoro.

In sostanza, le **capacità di leadership** sono le capacità che le persone hanno di guidare e realizzare progetti, incoraggiare iniziative, costruire un senso di scopo comune e responsabilizzare gli altri.

Le **capacità di leadership** includono anche le capacità che le persone hanno di guidare i dipendenti verso il raggiungimento degli obiettivi aziendali, ispirarli, guidare il cambiamento e fornire risultati.



Source: <https://www.thebalancecareers.com/top-leadership-skills-2063782>

2. Gestione della squadra

Sebbene non esista una singola metrica o abilità di leadership che definisca un ottimo team manager, ci sono diversi aspetti che definiscono un team ben funzionante che mostrano una forte leadership. Una gestione di successo è quando tutti hanno:

- Una chiara comprensione delle proprie responsabilità
- Il desiderio di raggiungere obiettivi organizzativi e di gruppo
- Check-in regolari da parte di manager e supervisori

Un manager eccellente trova anche il tempo per ascoltare ed eseguire il feedback dei dipendenti secondo necessità, dando l'esempio - sia mostrando un buon equilibrio tra lavoro e vita private sia portando a termine le attività in tempo - e rimane trasparente sulle loro sfide lavorative e su come le risolveranno



Source: <https://www.marketing91.com/team-management/>

3. Problem solving

Il **Problem solving** è il modo in cui vengono sviluppate soluzioni per rimuovere un ostacolo che impedisce raggiungimento di un obiettivo finale. I problemi che necessitano di soluzioni vanno da problemi personali di base ("come accendo questo elettrodomestico?") ad argomenti più complessi nel campo aziendale e accademico. Il primo scenario rientra nella categoria della risoluzione di problemi semplici (SPS), mentre il secondo è noto come risoluzione di problemi complessi (CPS). I problemi semplici hanno una sola problematica, mentre i problemi complessi hanno più ostacoli che richiedono contemporaneamente soluzioni. ^[1]

1. Frensch, Peter A.; Funke, Joachim, eds. (2014-04-04). *Complex Problem Solving*. doi:[10.4324/9781315806723](https://doi.org/10.4324/9781315806723). ISBN 9781315806723.

<https://www.youtube.com/watch?v=kIAE9ML0XLs>

4. Processo decisionale

Il processo decisionale è **il processo di fare scelte identificando una decisione, raccogliendo informazioni e valutando risoluzioni alternative**. L'utilizzo di un processo decisionale graduale può aiutarti a prendere decisioni più deliberate e ponderate organizzando le informazioni pertinenti e definendo le alternative.

Descriviamo la sequenza delle fasi coinvolte nel processo decisionale.

1. **Classificare del problema**. È generico? È eccezionale e unico? O è la prima manifestazione di un nuovo genere per il quale deve ancora essere sviluppata una regola?
2. **Definire il problema**. Con cosa abbiamo a che fare?
3. **Specificare la risposta al problema**. Quali sono le "condizioni al contorno"?
4. **Decidere cosa è "giusto", piuttosto che cosa è accettabile, al fine di soddisfare le condizioni al contorno**. Cosa soddisferà pienamente le specifiche prima che venga prestata attenzione ai compromessi, agli adattamenti e alle concessioni necessari per rendere accettabile la decisione?
5. **Costruire nella decisione l'azione per realizzarla**. Quale deve essere l'impegno di azione? Chi deve saperlo?
6. **Testare la validità e l'efficacia della decisione rispetto all'effettivo corso degli eventi**. Come viene eseguita la decisione? I presupposti su cui si basa sono appropriati o obsoleti?



Source: https://www.researchgate.net/figure/Decision-making-for-ecosystem-service-management_fig2_313161530

5. Abilità comunicative

Le abilità comunicative implicano **ascoltare, parlare, osservare ed entrare in empatia**. È anche utile comprendere le differenze nel modo di comunicare attraverso interazioni faccia a faccia, conversazioni telefoniche e comunicazioni digitali, come e-mail e social media.

Le sette C della comunicazione sono un elenco di principi per le comunicazioni scritte e orali per garantire che siano efficaci. Le sette C sono: **chiarezza, correttezza, compattezza, cortesia, concretezza, considerazione e completezza**.



Source: <https://www.thebalancecareers.com/communication-skills-list-2063779>

6. Creatività e innovazione

La creatività è la capacità di trascendere i modi tradizionali di pensare o agire e di sviluppare idee, metodi o oggetti nuovi e originali.

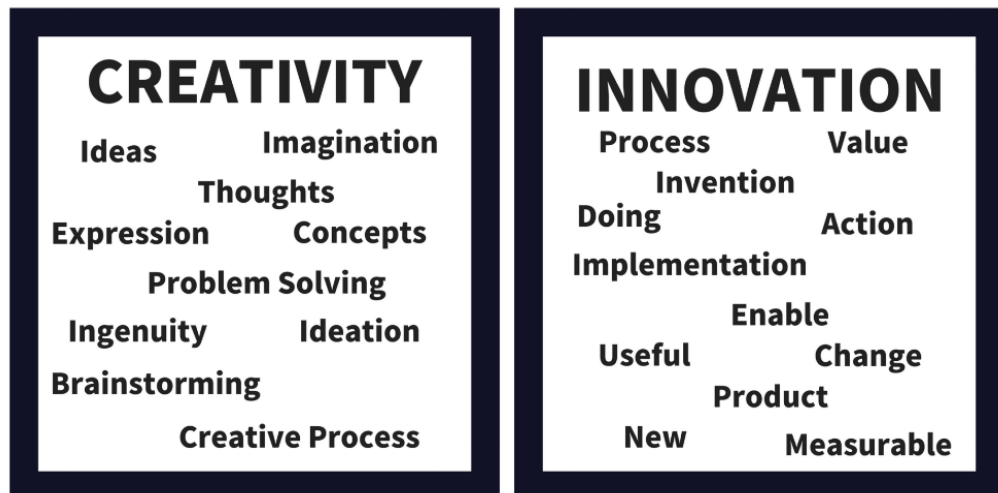
L'innovazione è l'implementazione pratica di idee che si traducono nell'introduzione di nuovi beni o servizi o nel miglioramento dell'offerta di beni o servizi.

La creatività è l'atto di trasformare idee nuove e fantasiose in realtà. La creatività è caratterizzata dalla capacità di percepire il mondo in modi nuovi, di trovare schemi nascosti, di creare connessioni tra fenomeni apparentemente non correlati e di generare soluzioni. La creatività coinvolge due processi: pensare, quindi produrre.

La creatività inizia con una base di conoscenza, l'apprendimento di una disciplina e la padronanza di un modo di pensare. Puoi imparare a essere creativo sperimentando, esplorando, mettendo in discussione le ipotesi, usando l'immaginazione e sintetizzando le informazioni.

Gli studi di Clayton M. Christensen e dei suoi ricercatori hanno scoperto il DNA dell'innovatore: la tua capacità di generare idee innovative non è solo una funzione della mente, ma anche una funzione di cinque comportamenti chiave che ottimizzano il tuo cervello per la scoperta

1. **Associazione:** tracciare connessioni tra domande, problemi o idee provenienti da campi non correlati
2. **Interrogare:** porre domande che sfidano la saggezza comune
3. **Osservare:** scrutare il comportamento di clienti, fornitori e concorrenti per identificare nuovi modi di fare le cose
4. **Networking:** incontrare persone con idee e prospettive diverse
5. **Sperimentare:** costruire esperienze interattive e provocare risposte non ortodosse per vedere quali intuizioni emergono



Source: <https://weareive.org/what-is-the-difference-between-creativity-and-innovation/>

7. Gestione del tempo

La gestione del tempo è **il processo di organizzazione e pianificazione di come dividere il proprio tempo tra diverse attività**. Se lo fai bene, finirai per lavorare in modo più intelligente, non duramente, per fare di più in meno tempo, anche quando il tempo è stretto e le pressioni sono elevate. I migliori gestiscono il loro tempo eccezionalmente bene.

<https://www.youtube.com/watch?v=xgp6eELYY1M>

8. Gestione del rischio

La gestione del rischio è **il processo di identificazione, valutazione e controllo delle minacce al capitale e agli utili di un'organizzazione**. Questi rischi derivano da una varietà di fonti

tra cui incertezze finanziarie, responsabilità legali, problemi tecnologici, errori di gestione strategica, incidenti e disastri naturali.

La gestione del rischio forse non è mai stata così importante come lo è ora. I rischi che le organizzazioni moderne devono affrontare sono diventati più complessi, alimentati dal rapido ritmo della globalizzazione. Emergono costantemente nuovi rischi, spesso legati e generati dall'uso ormai pervasivo della tecnologia digitale. Il cambiamento climatico è stato soprannominato un "moltiplicatore di minacce" dagli esperti di rischio.

<https://www.youtube.com/watch?v=TcKoUe8vRE0>



Source: <https://aipc.ae/risk-management-process/>

3. L'imprenditorialità è giusta per te?

Prima di procedere con il tuo piano per diventare un imprenditore, valuta le tue capacità rispetto a tutte le domande e considerazioni di cui sopra. Usa un'analisi SWOT personale per esaminare i tuoi punti di forza e di debolezza, le tue opportunità e le minacce che potresti incontrare.

Sii onesto con te stesso riguardo alle tue motivazioni e al livello di impegno che sei disposto a dare al tuo progetto. Questo potrebbe impedirti di commettere un errore costoso.

Mentre lavori alla tua analisi, potresti sentirti pronto per tuffarti nella tua nuova entusiasmante avventura. In alternativa, puoi decidere di aspettare e sviluppare ulteriormente le tue abilità. Potresti persino decidere che l'imprenditorialità non fa per te, dopo tutto.



Diventare un imprenditore è un'importante decisione professionale, quindi evita la tentazione di agire in modo impulsivo. Fai i tuoi compiti. Rifletti sulle tue esigenze, sui tuoi obiettivi e sulle tue circostanze finanziarie e personali. L'imprenditorialità può richiedere un'enorme quantità di tempo e dedizione, quindi assicurati che sia giusto.



Metodi interattivi per sviluppare le capacità imprenditoriali

1. Metodi per stimolare la creatività degli imprenditori
2. Metodi interattivi per sviluppare capacità imprenditoriali
 1. Sei pronto per essere un imprenditore?
 2. Inizia la tua attività con una graffetta
 3. Business Model Canvas – Pianifica la tua attività in una pagina!
 4. Gestisci la tua organizzazione con stile
3. Bibliografia

1. Metodi per stimolare la creatività degli imprenditori

I metodi della creatività vengono utilizzati per trovare nuove idee di prodotto. Questi metodi non cercano di trovare soluzioni che soddisfino i bisogni espressi sul mercato, ma cercano di anticipare l'evoluzione dei bisogni dei consumatori.

I metodi della creatività possono essere suddivisi in due categorie: metodi intuitivi e razionali.

Insieme a questi metodi possiamo aggiungere: The Delphi Group Method - il metodo del gruppo Delphi, 6 Thinking Hats Method - il metodo dei 6 cappelli per pensare, The Lotus Technique - la tecnica del loto (fiore di ninfea), The Interdependent Groups Method - il metodo dei gruppi interdipendenti (mosaico), The Pyramid Method - il metodo della piramide, Star bursting - lo scoppio delle stelle (esplosione stellare), ecc.

I metodi intuitivi si basano sull'idea che un gruppo di individui è molto più creativo di ogni individuo che lavora separatamente. L'accento è posto sull'effetto sinergico derivante dall'interazione tra i membri del gruppo. Attraverso questi metodi, vengono sfruttate l'immaginazione e l'intuizione delle persone che partecipano alle sessioni di creatività.

I metodi razionali implicano la ricerca di nuove idee di prodotto, a partire dall'analisi sistematica dei prodotti esistenti. Nel processo di analisi possono essere presi in considerazione diversi aspetti, quali: le caratteristiche dei prodotti esistenti, il loro comportamento nel consumo, le modalità di utilizzo, le situazioni di consumo, ecc.

2. Metodi interattivi per sviluppare le capacità imprenditoriali

2.1 Sei pronto per essere un imprenditore?

Preparazione: 15 minuti

Durata: 40 minuti

Dimensione dei gruppi: 5-30 persone

Età: 16+

Tags: imprenditorialità, autovalutazione

Quando viene utilizzato?

- valutare le qualità/competenze personali necessarie per diventare un imprenditore

Materiali:

- fogli stampati con il test del quoziente imprenditoriale e i risultati
- penne e pennarelli
- le spiegazioni stampate del quiz
- foglio per lavagna a fogli mobile

Come?

Preparazione:

• Stampa il questionario per stabilire il quoziente di imprenditorialità, adattato da Start Your Own Business, Sixth Edition: The Only Startup Book You'll Ever Need; il test è stato originariamente creato dall'assicurazione Northwestern Mutual Life. Avrai bisogno di una copia del test per ogni partecipante.

- Leggi brevemente la presentazione del questionario nella stessa appendice

Presentazione:

1. Chiedi ai partecipanti quali sono, dal loro punto di vista, le qualità più importanti di un imprenditore. Scrivi su una lavagna a fogli mobili le cose che hanno menzionato. Puoi utilizzare

un noto imprenditore nella tua comunità in modo che i partecipanti possano avere in mente una persona reale quando pensano a qualità/abilità specifiche.

2. Una volta terminata la sessione di brainstorming, presentare ai partecipanti l'idea che le capacità imprenditoriali possono essere testate e che durante questo esercizio possono testare il proprio quoziente imprenditoriale.

3. Consegnare a ciascun partecipante un questionario stampato e concedere loro 15-20 minuti per rispondere alle domande.

4. Dopo che tutti hanno finito di compilare il questionario, consegnare a ciascun partecipante un foglio stampato con i risultati del questionario. Aiuta i partecipanti a valutare il loro questionario di conseguenza rispondendo a turno a ciascuna domanda. Allo stesso tempo, spiega per ogni domanda perché alcune risposte sono più "imprenditoriali" di altre

5. Ricapitolando le abilità più importanti che le persone dovrebbero sviluppare per essere imprenditori di successo. Annota le nuove abilità sulla lavagna a fogli mobili che hai usato in origine.

6. Ricorda ai partecipanti quanto segue: il fatto che abbiano queste qualità/capacità, non significa automaticamente che avranno successo nella loro attività; inoltre, qualcuno che al momento non ha queste qualità/capacità, per avere successo può sviluppare queste capacità nel tempo; e in un caso e nell'altro ci vuole molto lavoro per sviluppare un business, le competenze da sole non bastano.

Domande successive:

Ora che sai qual è il tuo quoziente imprenditoriale e se stai pensando di diventare un imprenditore, quali pensi siano le tue capacità su cui puoi già fare affidamento che ti aiuteranno a far crescere la tua attività?

Di quali abilità hai bisogno per sviluppare e come pensi di farlo?

<https://www.youtube.com/watch?v=IjJILQu2xM8>



Source: 21stcenturyeducation.edu.pk/course/entrepreneurship/



Appendice

Metti alla prova il tuo quoziente imprenditoriale!

Adattato da Start Your Own Business, Sixth Edition: The Only Startup Book You'll Ever Need, di Inc the Staff of Entrepreneur Media; il test è stato originariamente sviluppato dalla Northwestern Mutual Life Insurance.

Prima di sostenere questo test, ricorda: anche se hai un alto quoziente imprenditoriale, non significa automaticamente che la tua attività avrà successo, ma che hai le qualità/competenze per avere successo in un'attività per la quale devi lavorare molto.

Non pensare troppo quando rispondi alle domande. Scegli la prima risposta che ti viene in mente quando pensi a te stesso. Inizia la prova!

Rispondi alle seguenti domande con "SI" o "NO" e somma il tuo punteggio, come indicato nella sezione alla fine del test, per scoprire qual è il tuo quoziente di imprenditorialità.

Si No

- ___ ___ 1. Sei uno dei migliori studenti della tua scuola?
- ___ ___ 2. Ti piace partecipare ad attività di gruppo come club o sport di squadra?
- ___ ___ 3. Preferisci stare da solo?
- ___ ___ 4. Ti sei candidato al consiglio studentesco o hai avviato piccole iniziative imprenditoriali, come un chiosco di limonate, un giornale di classe/famiglia o una vendita di carte?
- ___ ___ 5. Sei testardo?
- ___ ___ 6. Sei prudente?
- ___ ___ 7. Sei audace o avventuroso?
- ___ ___ 8. Le opinioni degli altri contano molto per te?
- ___ ___ 9. Avvieresti un'attività in proprio solo per sfuggire alla routine quotidiana?
- ___ ___ 10. Anche se ti piace lavorare, saresti disposto a lavorare di notte?
- ___ ___ 11. Sei disposto a lavorare, non importa quanto tempo, anche se hai dormito poco o niente, per completare un progetto?
- ___ ___ 12. Dopo aver completato con successo un progetto, ne inizi un altro subito dopo?
- ___ ___ 13. Sei disposto a utilizzare i tuoi risparmi per avviare un'impresa?
- ___ ___ 14. Saresti disposto a prendere in prestito denaro da qualcun altro?
- ___ ___ 15. Se la tua attività fallisse, ne avvieresti subito una nuova?
- ___ ___ 16. Oppure inizieresti a cercare un lavoro con uno stipendio regolare?
- ___ ___ 17. Pensi che sia rischioso fare l'imprenditore?
- ___ ___ 18. Scrivi i tuoi obiettivi a breve oa lungo termine?
- ___ ___ 19. Pensi di avere la capacità di gestire bene i tuoi soldi?
- ___ ___ 20. Ti annoi facilmente?
- ___ ___ 21. Sei ottimista?





Come calcolare il tuo punteggio:

1. Se sì, sottrarre 4 punti; in caso contrario, aggiungi 4.
2. Se sì, sottrarre 1 punto; in caso contrario, aggiungi 1.
3. Se sì, raccogli 1 punto; in caso contrario, sottrarre 1 .
4. Se sì, raccogli 2 punti; in caso contrario, sottrarre 2.
5. Se sì, raccogli 1 punto; in caso contrario, sottrarre 1.
6. Se sì, sottrarre 4 punti; in caso contrario, aggiungi 4.
7. Se sì, raccogli 4 punti; in caso contrario, non ottieni punti.
8. Se sì, sottrarre 1 punto; in caso contrario, aggiungi 1.
9. Se sì, aggiungi 2 punti; in caso contrario, sottrarre 2.
10. Se sì, aggiungi 2 punti; in caso contrario, sottrarre 6.
11. Se sì, raccogli 4 punti; in caso contrario, non ottieni punti.
12. Se sì, raccogli 2 punti; in caso contrario, sottrarre 2.
13. Se sì, raccogli 2 punti; in caso contrario, sottrarre 2.
14. Se sì, raccogli 2 punti; in caso contrario, sottrarre 2.
15. Se sì, raccogli 4 punti; in caso contrario, sottrarre 4.
16. Se sì, sottrarre 1 punto; in caso contrario, non ottieni punti.
17. Se sì, sottrarre 2 punti; in caso contrario, aggiungi 2.
18. Se sì, raccogli 1 punto; in caso contrario, sottrarre 1.
19. Se sì, raccogli 2 punti; in caso contrario, sottrarre 2.
20. Se sì, aggiungi 2 punti; in caso contrario, sottrarre 2.
21. Se 2, aggiungi 2 punti; in caso contrario, sottrarre 2.

Risultati del test

- Un punteggio di 35 punti o più: tutto sta andando bene per te. Hai il potenziale per ottenere uno spettacolare successo imprenditoriale.

- Un punteggio compreso tra 15 e 34: il tuo background, le tue abilità e il tuo talento ti danno un'ottima possibilità di avere successo nella tua attività. Dovresti andare lontano.

- Un punteggio compreso tra 0 e 14: hai buone capacità per avviare e/o esperienza per gestire un'impresa e dovresti avere successo nell'avviarne una se impari a gestirla.

- Un punteggio compreso tra -15 e -1: potresti avere successo se ti avventurassi da solo, ma potresti dover lavorare di più per compensare la mancanza di vantaggi e competenze che danno agli imprenditori un vantaggio quando avviano l'attività

- Un punteggio compreso tra -16 e -43: Probabilmente hai altri talenti oltre a questo. Dovresti pensare se costruire un'impresa è ciò che vuoi veramente fare, perché potresti trovarti a nuotare controcorrente. Lavorare in un'azienda o per qualcun altro, sviluppare una carriera in una professione o acquisire esperienza in un campo tecnico sarebbe molto più adatto a te e ti permetterebbe di godere di uno stile di vita più appagante e adatto alle tue capacità e ai tuoi interessi.

Interpretazione:

Se il tuo punteggio è alto, non significa necessariamente che avrai successo nella tua attività, ma che hai le qualità necessarie per avere successo nell'attività che hai avviato. Se il tuo punteggio è basso, non significa che non avrai successo. Ci sono altri fattori da considerare prima di fare questo passo.

2.2. Inizia la tua attività con una graffetta

Preparazione: 30 minuti

Durata: 45 minuti

Dimensione dei gruppi: 5-20 persone





Età: 16+

Tags: imprenditorialità, creatività, uso alternativo, pensiero innovativo, "pensare fuori dagli schemi"

Quando viene utilizzato?

- valutazione del potenziale creativo;
- miglioramento delle capacità di pensiero/pensiero "out of the box";
- capire come la creatività e l'innovazione aiutano a far crescere il tuo business.

Materiali:

- penne, post-it, fogli per lavagna a fogli mobili, pennarelli, graffette

Come?

Preparazione:

- Leggi la breve panoramica del test sugli usi alternativi nel FACTS BOX.

Presentazione:

1. Avviare la discussione chiedendo ai partecipanti la loro definizione di "creatività" e scrivere le loro risposte su una lavagna a fogli mobili.
2. Poi chiedi loro quali sono le principali cose/aspetti che una persona dovrebbe avere per essere considerata creativa. Scrivi anche queste risposte sulla lavagna a fogli mobili.
3. Dai a ogni partecipante una graffetta e molti post-it, quindi dai loro 3 minuti per pensare a quante più alternative possibili all'uso della graffetta. Per ogni idea, dovrebbero usare un post-it diverso.
4. Poi chiedi ai partecipanti di attaccare i post-it sulla lavagna a fogli mobili mentre li presentano. Chiedi loro di attaccare sulla lavagna a fogli mobili solo le nuove idee, quelle che non sono state menzionate prima.
5. Dopo che tutte le idee sono sulla lavagna a fogli mobili, dividere le idee sui post-it con l'aiuto dei partecipanti in tre categorie:
 - Usa la graffetta così com'è;
 - Modificare il design della graffetta;
 - Aggiunge un oggetto alla graffetta, creando un altro oggetto.

Inoltre, analizza le idee insieme ai partecipanti dalle seguenti prospettive:

- Fluidità (quanti usi trovati da ciascun partecipante rispetto al numero totale di usi trovati dal gruppo);
- Originalità (quanto è insolita l'idea);
- Flessibilità (quanti settori diversi vengono coperti con le idee; per esempio, se qualcuno accenna all'idea di utilizzare graffette come orecchini ma anche come collana, allora quella persona ha coperto solo un settore – gioielli/accessori);
- Elaborazione (quanto erano dettagliate le idee).

6. Chiedi ai partecipanti come sono arrivati a queste idee, quindi presenta loro l'idea che la creatività può essere allenata attraverso esercizi quotidiani come questo.

7. Ora chiedi ai partecipanti perché la creatività potrebbe essere importante per un'azienda. Sulla base delle loro risposte, spiega loro che non basta essere creativi, pensare "fuori dagli schemi", e che è importante essere in grado di applicare la creatività per risolvere i problemi della comunità, per soddisfare i bisogni di potenziali clienti o opportunità di business. Pertanto, le idee creative devono essere messe in pratica attraverso l'innovazione.

Domande successive:

Chiedi ai partecipanti di dire cosa è successo una volta che hanno visto le idee di altre persone su come usare una graffetta. Hanno trovato altre idee?

Chiedi loro quanti usi alternativi hanno trovato prima che diventassero davvero creativi usando una graffetta? Cosa dice loro? Aiutali a capire che è molto importante accettare le sfide e che le buone idee vengono dopo piuttosto che all'inizio del processo.

FACTS BOX

Alternative Uses Task - Prova di usi alternativi

Sviluppato da J.P. Guilford nel 1967, l'Alternative Uses Task Test chiede di trovare quanti più usi possibili per un oggetto, come un mattone, una scarpa, una graffetta, ecc., in un breve periodo di tempo (di solito tra 2 e 5 minuti).

Il test misura le capacità di pensiero divergenti di una persona. Ciò contrasta con i test tradizionali che si concentrano sul pensiero convergente, ovvero la capacità di una persona di trovare un'unica soluzione "corretta" a un problema. Questo test incoraggia le persone a trovare più soluzioni a un problema.

In vari studi, il test ha mostrato che i partecipanti che hanno trovato molte idee in un dato momento avevano risposte molto più creative, e le idee offerte all'inizio, tra le prime, erano meno creative rispetto a quelle nella seconda parte dell'elenco delle idee. In altre parole, la creatività migliora con la pratica e la tenacia.

La valutazione del test comprende quattro elementi:

1. Fluidità: quanti usi può trovare una persona;
2. Originalità: quanto sono insolite le idee, una sorta di prova di un pensiero diverso;
3. Flessibilità – la varietà di idee, provenienti da vari settori o categorie;
4. Elaborazione - quanto dettagliate sono descritte le idee.

Questo test può essere utilizzato ogni giorno, partendo da vari oggetti, come strumento per sviluppare la creatività.

<https://www.youtube.com/watch?v=yjYrxcGSWX4>

<https://www.youtube.com/watch?v=tTjETjGDFY>



Image ID: H99MXN
www.alamy.com



<https://www.alamy.com/stock-photo-creative-creativity-innovation-design-vision-concept-126130669.html>

2.3. Business Model Canvas – Pianifica la tua attività in una pagina!

Preparazione: 60 minuti

Durata: 90 minuti

Dimensione dei gruppi: 5-30 persone

Età dei partecipanti: 16+

Tags: imprenditorialità, pianificazione, business plan

Quando viene utilizzato?

- trasformare un'idea imprenditoriale in un vero e proprio progetto, pronto per essere messo in pratica
- pianificazione di progetti che coinvolgono diverse tipologie di committenti e stakeholder

Materiali:

fogli di lavagna a fogli mobili, post-it – di vari colori, pennarelli, videoproiettore e laptop

Preparazione:

Guarda la breve presentazione video sul modello di business canvas

qui: <https://www.youtube.com/watch?v=QoAOzMTLP5s>;

• Preparare un foglio per lavagna a fogli mobili, utilizzando il modello di business canvas disponibile a questo link:

<https://templatelab.com/business-model-templates/>;

• Utilizzando post-it multicolori, prepara la presentazione del progetto/idea imprenditoriale (scegli un'idea semplice, magari anche familiare ai partecipanti), seguendo le nove sezioni del canvas; utilizzare almeno due tipi di clienti in modo che le persone capiscano come i clienti possono diventare partner chiave o fornire risorse chiave al progetto quando viene discusso un altro segmento di clienti.

Presentazione:

1. Chiedere ai partecipanti quali sono gli elementi principali di qualsiasi progetto (obiettivi, scopo, problema, beneficiari, attività, metodologia, risultati stimati, budget, ecc.) o di qualsiasi idea imprenditoriale (prodotti, clienti, canali di distribuzione, prezzi, costi, promozione ecc.).
2. Presenta il business model canvas come uno strumento che racchiude in un'unica pagina tutti i componenti di un progetto/idea imprenditoriale.
3. Esempifica l'uso del modello di business canvas con il progetto/idea imprenditoriale che hai preparato prima del workshop. Mostra come funziona il modello per almeno due diversi segmenti di clientela. Mostra loro come i nove elementi sono correlati tra loro: clienti/segmenti di clienti, esempi di valore del cliente (prodotti/servizi), relazione con il cliente, canali di comunicazione, flussi di entrate, attività chiave, risorse chiave, partner chiave e struttura dei costi.
4. Dividere i partecipanti in gruppi di cinque e dare a ciascuno una lavagna a fogli mobili.
5. Chiedere a ciascun gruppo di preparare un modello di business canvas su un foglio della lavagna a fogli mobili.
6. Chiedere a ciascun gruppo di presentare un'idea di progetto che vorrebbe implementare nella propria comunità/scuola/quartiere e preparare una presentazione del progetto utilizzando il modello di business canvas.



7. Ogni gruppo ha dai cinque ai sette minuti per presentare il proprio progetto.
8. Dai un feedback su ogni idea di progetto. Non concentrarti sull'idea del progetto, ma su come il progetto è stato pianificato utilizzando il modello di business canvas.

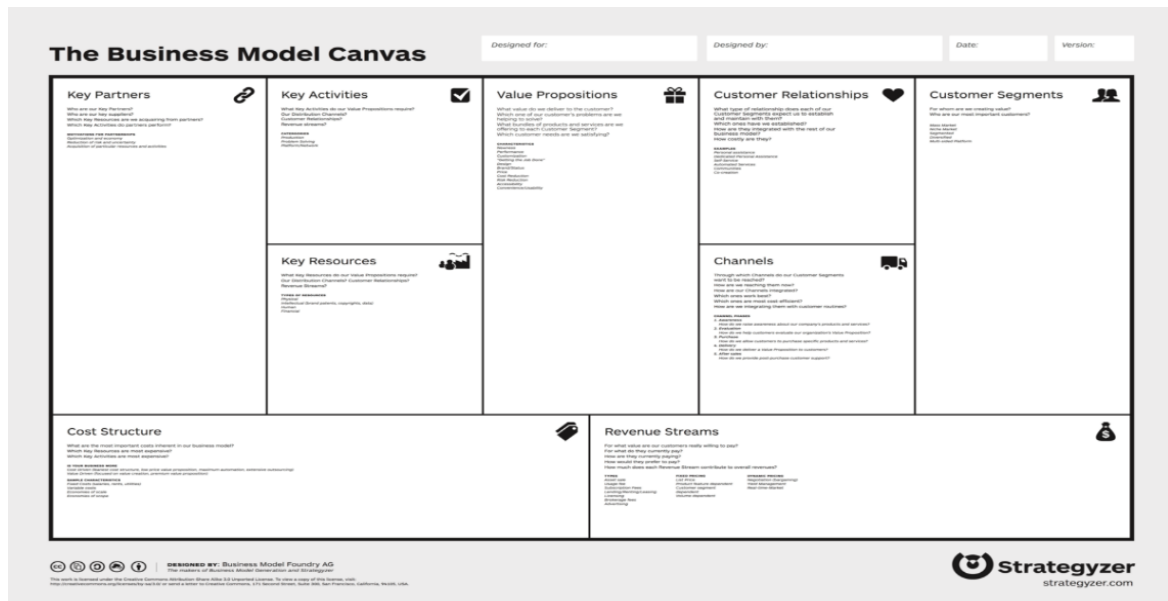
Domande successive:

Come si è sviluppata/cambiata l'idea del progetto durante l'esercizio?

Quali sono i nuovi aspetti/idee o nuove domande relative al progetto, generate con l'utilizzo del business model canvas?

Cosa puoi dire della sostenibilità del tuo progetto, guardando al modello di business che hai sviluppato?

<https://www.youtube.com/watch?v=I8nwNcCfyig>

**2.4. Gestisci la tua organizzazione con stile**

Preparazione: 25 minutes

Durata: 45 minutes

Dimensione dei gruppi: 15 persone

Età: 16+

Tags: imprenditorialità, leadership, stili di leadership

Quando viene utilizzato?

- per apprendere e comprendere diversi stili di leadership
- per comprendere i vantaggi e gli svantaggi di ogni stile di leadership

Materiali: fogli per lavagna a fogli mobili, pennarelli

Come?**Preparazione:**

- Stampa i tre scenari con i tre tipi di leader (leader democratico, leader autocratico e leader laissez-faire) dall'appendice.

Presentazione:

1. Dividi i partecipanti in tre gruppi e selezionare una persona da ciascun gruppo da designare come leader.

2. Assegna a ciascun leader (ma NON ai gruppi) uno dei tre scenari in appendice.
3. Mentre i tre leader leggono il loro copione, consegna a ogni gruppo una lavagna a fogli mobili e dei pennarelli e comunica loro che riceveranno istruzioni dai loro leader.
4. Ora lascia che i gruppi siano guidati dai leader e svolgano il compito in circa 15 minuti.
5. Ogni gruppo deve presentare le proprie decisioni e risultati dell'attività di gruppo.

Domande successive:

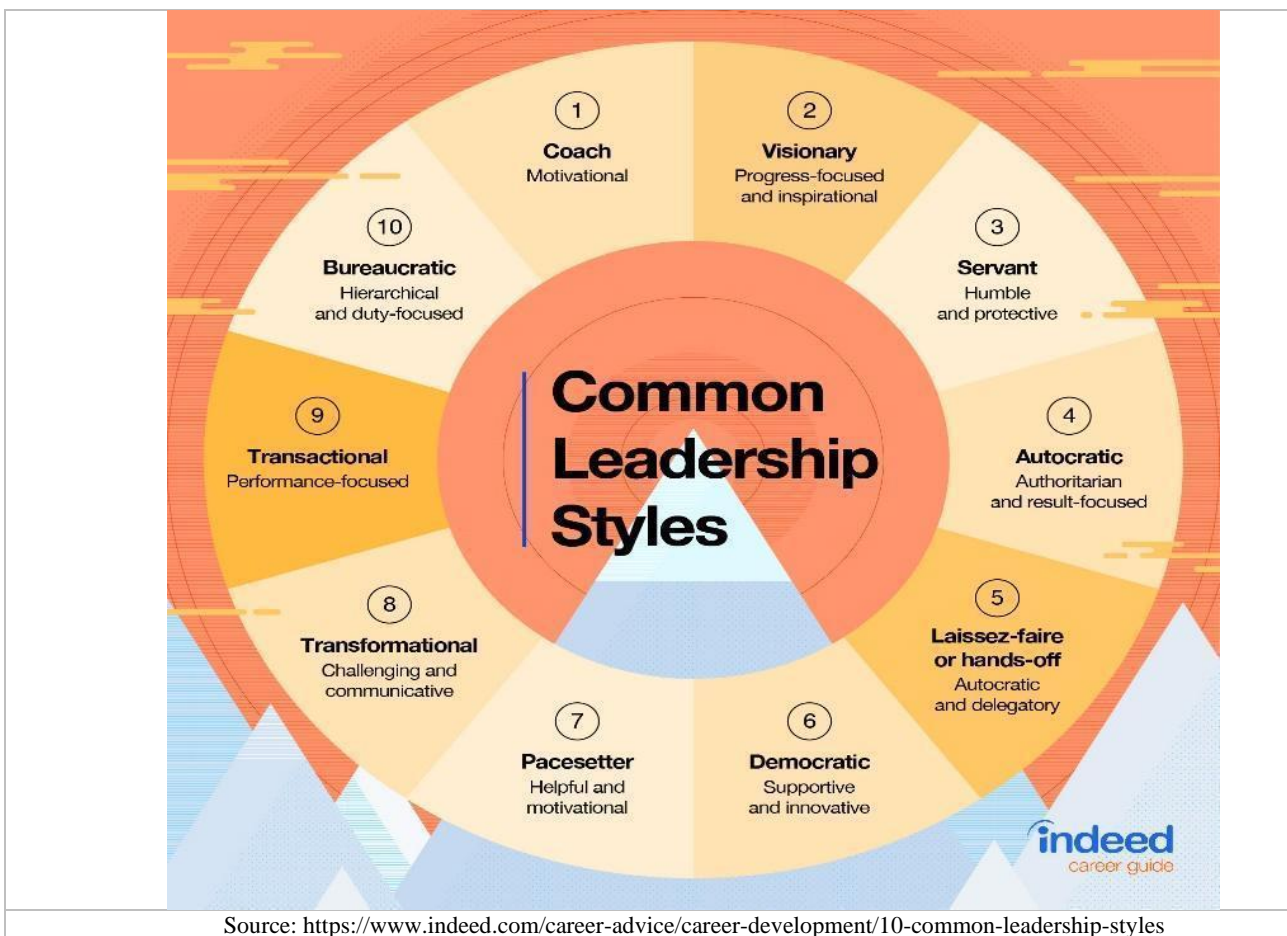
Quali comportamenti hai identificato nei tuoi leader? E tra i "seguaci"?

Come ti sei sentito come leader o membro del team durante il gioco di ruolo?

Quanto è stata efficace ogni riunione guidata dal leader?

Quali sono stati i vantaggi e gli svantaggi di ogni incontro determinati dallo stile di leadership?

<https://www.youtube.com/watch?v=2IEp4TVpxgA>



Appendice

Scenario A



Sei un manager in un'azienda con 250 dipendenti. Il top management ti chiede di dare suggerimenti per l'organizzazione della festa annuale. Lo scopo della festa è celebrare i buoni risultati dell'azienda a cui hanno contribuito tutti i dipendenti, quindi è necessario offrire un'ampia varietà di attività nella proposta. Hai pianificato un breve incontro con il tuo team. Nei prossimi 15 minuti devi raccogliere quante più idee possibili e elaborare un piano per la festa.

Al termine dell'incontro, una persona sarà nominata per presentare i risultati e rispondere alle domande.

Riepilogo:

- Obiettivo dell'incontro: creare un piano di attività per la festa.
- Durata: 15 minuti.
- Logistica: il piano verrà scritto su un foglio di lavagna a fogli mobili.
- Verrà nominata una persona per presentare i risultati.

Cerca di seguire bene i seguenti comportamenti:

- Ti prepari con cura e prendi appunti sulle idee di base che vuoi portare alla riunione.
- Sei l'unico che sa cosa è bene per la squadra.
- Tu conduci la discussione; dà ordini agli altri su ciò che deve essere fatto.
- Assegna immediatamente compiti a ciascun membro del team.
- Influenza e domina la discussione fino all'ultimo dettaglio.
- Parli di più.
- Non devi fare domande e ascoltare; interrompi immediatamente la conversazione se ritieni che non sia necessario.
- Interrompi spesso gli altri.
- Criticare e lodare direttamente, apertamente.
- Non chiedere ai membri del team di presentare le loro idee o suggerimenti.
- Ti aspetti ottimi risultati e giudichi i tuoi subordinati, considerandoti il migliore.
- Dopotutto, hai già il piano per la festa e devi solo spiegarlo ai tuoi subordinati.
- Devi presentare i risultati.

Scenario B

Sei un manager in un'azienda con 250 dipendenti. Il top management ti chiede di dare suggerimenti per l'organizzazione della festa annuale. Lo scopo della festa è celebrare i buoni risultati dell'azienda a cui hanno contribuito tutti i dipendenti, quindi è necessario offrire un'ampia varietà di attività nella proposta. Hai pianificato un breve incontro con il tuo team. Nei prossimi 15 minuti devi raccogliere quante più idee possibili e elaborare un piano per la festa.

Al termine dell'incontro, una persona sarà nominata per presentare i risultati e rispondere alle domande.

Riepilogo:

- Obiettivo dell'incontro: creare un piano di attività per la festa.
- Durata: 15 minuti.
- Logistica: il piano verrà scritto su un foglio di lavagna a fogli mobili.
- Verrà nominata una persona per presentare i risultati.

Cerca di seguire bene i seguenti comportamenti:

- Fin dall'inizio, informi il tuo team sugli obiettivi dell'incontro, su cosa si prevede di raggiungere alla fine e su come discuterne.
- Raggiungere un'intesa sull'organizzazione della riunione e sui compiti individuali.



- Chiedi ai membri del team di dare suggerimenti specifici e incoraggia tutti a essere attivi.
- Fai domande per supporto e incoraggiamento; ascolta attentamente.
- Coinvolgi i tuoi colleghi nel processo decisionale.
- Spieghi loro le tue opinioni e non le imponi.
- Coinvolgi i più tranquilli e incoraggi tutti a contribuire.
- Non forzare nulla, ma fornisci attività di supporto per tutti i dipendenti.
- Controlla la discussione solo quando necessario e supporta le iniziative individuali o la conduzione della riunione da parte dei membri del team.

Scenario C

Sei un manager in un'azienda con 250 dipendenti. Il top management ti chiede di dare suggerimenti per l'organizzazione della festa annuale. Lo scopo della festa è celebrare i buoni risultati dell'azienda a cui hanno contribuito tutti i dipendenti, quindi è necessario offrire un'ampia varietà di attività nella proposta. Hai pianificato un breve incontro con il tuo team. Nei prossimi 15 minuti devi raccogliere quante più idee possibili e elaborare un piano per la festa.

Al termine dell'incontro, una persona sarà nominata per presentare i risultati e rispondere alle domande.

Riepilogo:

- Obiettivo dell'incontro: creare un piano di attività per la festa.
- Durata: 15 minuti.
- Logistica: il piano verrà scritto su un foglio di lavagna a fogli mobili.
- Verrà nominata una persona per presentare i risultati.

Cerca di seguire bene i seguenti comportamenti:

- Inizia informando brevemente i tuoi subordinati sugli obiettivi della riunione; senza dare troppi dettagli; chiedi loro di iniziare a lavorare.
- Sii più un osservatore.
- Lasciali parlare tra loro e non interferire.
- Verifica che siano in grado di gestire la discussione senza che tu venga coinvolto.
- Non esprimere le tue opinioni.
- Non partecipare a discussioni ed evita il contatto visivo.
- Attendi che le domande ti vengano rivolte direttamente.
- Rispondi alle domande con altre domande o chiedi semplicemente: "Ma cosa ne pensi?".
- Non criticare, non lodare.
- Hai un ruolo neutrale; non sei il capo squadra.
- Un'altra persona dovrebbe presentare le risposte.



Bibliography

1. <https://www.investopedia.com/terms/e/entrepreneur.asp#:~:text=Entrepreneurship%20is%20when%20an%20individual,with%20an%20innovative%20new%20idea>
2. <https://en.wikipedia.org/>
3. https://www.mindtools.com/pages/article/newCDV_76.htm
4. Anderson and Adams, “Mastering Leadership,” Wiley, 2015.
5. Bucur Viorel, Curs Antreprenoriat, ClujNapoca 2011
6. Dorel Ursu -Crearea și dezvoltarea afacerilor –planul de afaceri, Modul 4, Programul “Investește în oameni”, 2007-2013
7. Dyer, Gregersen, and Christensen - The Innovator’s DNA: Mastering the Five Skills of Disruptive Innovators
8. Harkavy Daniel - Cele 7 perspective ale liderilor eficienti Metode testate pentru decizii mai bune si o influenta mai mare, Amaltea, 2021
9. Roxana Carmen Cordoș, Bacali, L. și alții , Antreprenoriat, Ed. Todesco, Cluj-Napoca, 2008
10. Stanier Bungay Mickael, Coaching zi de zi, ed. Publica, 2017
11. http://www.tydzienprzedsiebiorczosci.pl/files/best_practices_EN.pdf
12. *Start Your Own Business, Sixth Edition: The Only Startup Book You'll Ever Need*, written by Inc The Staff of Entrepreneur Media. <http://www.amazon.com/Start-Your-Own-Business-Sixth>
13. *How to Improve Your Critical Thinking Skills and Make Better Business Decisions* <http://www.entrepreneur.com/article/226484>
14. Articol despre luarea deciziilor: <https://new.edu/resources/making-rational-decisions>
15. *Dan Ariely: Are we in control of our own decisions?*, TED Talks https://www.ted.com/talks/dan_ariely_asks_are_we_in_control_of_our_own_decisions?language=en
16. Cum să atragi investitorii: http://www.iseeyou-network.eu/files/8514/5579/7112/Business_Angel_Attraction_Plan.pdf
17. Robert Kiyosaki, *Rich Dad’s Cashflow Quadrant. Guide to Financial Freedom* (2012).



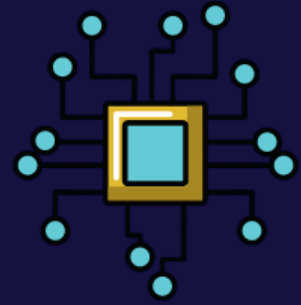


18. *Trump and Kiyosaki on Building Business Wealth*,
<http://www.entrepreneur.com/article/169174>
19. Business Model
Canvas <http://www.businessmodelgeneration.com/> <http://nonprofitcanvaskit.com/>
20. *The 7 habits of highly effective people*, by Stephen Covey: <https://www.stephencovey.com/7habits/7habits-habit1.php>
21. *7 Reasons '7 Habits of Highly Effective People' Lives on 25 Years Later* <http://www.entrepreneur.com/article/236544>
22. *Leadership Styles and Types: Authoritarian, Laissez-Faire & Democratic* <http://study.com/academy/lesson/leadership-styles-and-types-authoritarian-laissez-faire-democratic.html>
23. Leadership situational: <https://situational.com/the-cls-difference/situational-leadership-what-we-do/>
24. Stilurile de leadership: https://www.mindtools.com/pages/article/newLDR_84.html
25. Clutterbuck, David & Chartered Institute of Personnel and Development. *Everyone needs a mentor: fostering talent in your organisation*. (Chartered Institute of Personnel and Development, 2004).
26. Crețu D., Daniliuc F., Vrînceanu C. Ghid practic pentru antreprenori, Evrika Publishing, 2021
27. Goleman David. Leadership. Puterea inteligenței emotionale. Selecție de texte, Curtea veche, București, 2016
28. Humphrey, Ronald H. *Effective leadership: theory, cases, and applications*. (SAGE, 2014).



Co-funded by the Erasmus+ Programme of the European Union

Competenze Green



ROMUAS
KILLS
RDUINO
SING
MACHINES
OUTDATED
RECOVER

C5

Expert program toolkit
PR1

Le informazioni e le opinioni contenute in questa pubblicazione sono quelle degli autori e non riflettono necessariamente l'opinione ufficiale dell'Unione Europea. Né le istituzioni e gli organi dell'Uni one europea né chiunque agisca per loro conto possono essere ritenuti responsabili dell'uso che potrà essere fatto delle informazioni ivi contenute.





1. Competenze Green

INDICE

5.1. Parte teorica

1. Sostenibilità ambientale e cambiamento climatico
2. Verso un'economia più verde
3. Cosa sono le Competenze Green
4. L'economia circolare nel revamping dei macchinari
5. Vantaggi del revamping per un'economia più verde
 - Benefici economici
 - Benefici sociali
 - Benefici ambientali
6. Un modello economico più sostenibile - The Triple Bottom Line (TBL)

5.2. Metodi interattivi per lo sviluppo delle Green Skills

- Esempio di applicazione pratica delle competenze green digitali: Il retrofitting di una vecchia Radio FM (caso studio)
- Promuovere e sviluppare le Competenze Green Skills attraverso il gioco: Penji protegge il pianeta! (gioco online)
- Sviluppare competenze verdi con l'apprendimento attraverso la pratica: il potere rinnovabile delle competenze green per le donne in Zambia (video)

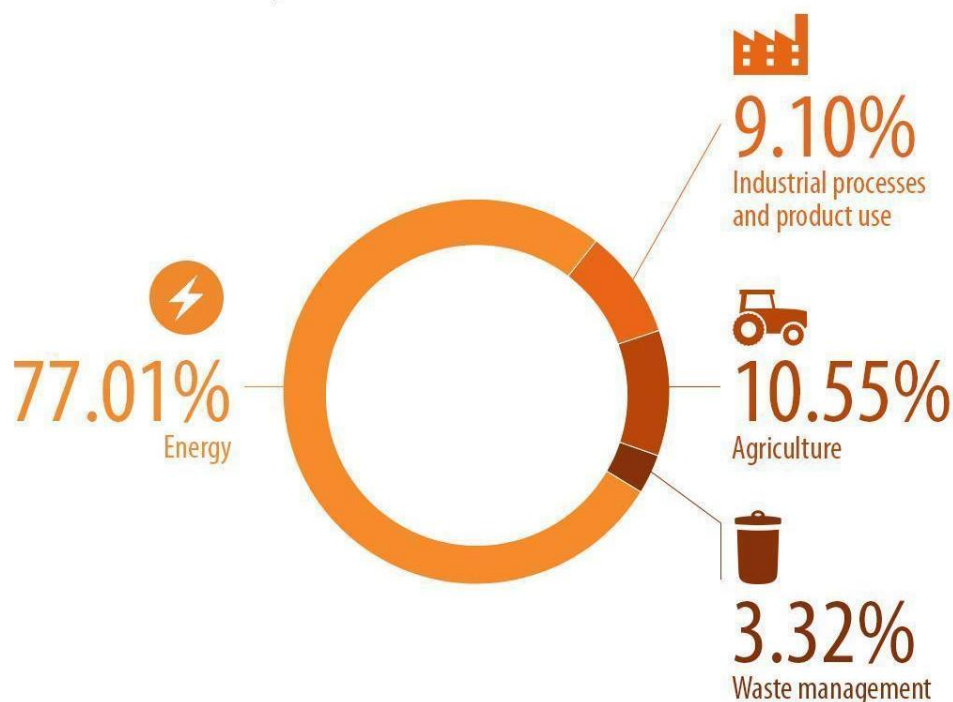
7. Fonti

1. Sostenibilità ambientale e cambiamento climatico

Il cambiamento climatico sta già interessando l'Europa in varie forme, a seconda della regione. Può portare alla perdita di biodiversità, incendi boschivi, diminuzione dei raccolti e temperature più elevate. Può anche influire sulla salute delle persone.

Nel 2021 l'UE ha reso la neutralità climatica e l'obiettivo di zero emissioni nette entro il 2050, giuridicamente vincolante nell'UE. Ha fissato un obiettivo intermedio di riduzione delle emissioni del 55% entro il 2030 con il Green Deal europeo; questo obiettivo di zero emissioni nette è sancito dalla legge sul clima. Il Green Deal europeo costituisce la tabella di marcia da seguire affinché l'UE diventi climaticamente neutra entro il 2050¹.

Greenhouse gas emissions in the EU by sector* in 2019



* All sectors excluding land use, land-use change and forestry (LULUCF)
The percentages do not add up to 100% due to rounded figures being used

Source: European Environment Agency (EEA)



Fonte immagine https://www.europarl.europa.eu/resources/library/images/20211026PHT15838/20211026PHT15838_original.jpg

Il **Patto Climatico**, ovvero il movimento di persone unite che si adoperano per costruire un'Europa più sostenibile, lanciato dalla Commissione europea nell'ambito del Green Deal Europeo,

¹ <https://www.europarl.europa.eu/news/en/headlines/priorities/climate-change/20180703STO07129/eu-responses-to-climate-change>



continuerà il lavoro dell'UE in questo settore e sosterrà attivamente le organizzazioni sindacali, gli enti educativi e le autorità pubbliche per aiutare coloro che cercano lavoro nell'ambito dell'economia sostenibile².

Per affrontare le pressanti sfide ambientali come il cambiamento climatico, l'inquinamento e il crollo della biodiversità, le nazioni e le imprese devono passare a economie e società più verdi, resilienti e climaticamente neutre.

Per questo motivo, l'Organizzazione Internazionale del Lavoro (OIL) ha creato le **Linee guida per una giusta transizione verso economie e società ambientalmente sostenibili per tutti**.

Una giusta transizione implica rendere l'economia più verde nel modo più equo e inclusivo possibile per tutti gli interessati, creando opportunità di lavoro dignitose e senza lasciare indietro nessuno.

Una transizione corretta comporta la massimizzazione delle opportunità sociali ed economiche nelle azioni a favore del clima, riducendo al minimo e gestendo con attenzione eventuali sfide, anche attraverso un dialogo sociale efficace tra tutti i gruppi interessati e il rispetto dei principi e dei diritti fondamentali del lavoro.

Garantire una transizione giusta è importante per tutti i paesi a tutti i livelli di sviluppo. È importante anche per tutti i settori economici – non solo per l'approvvigionamento energetico – e nelle aree urbane e rurali.³

2. Verso un'economia più verde

Un'economia circolare correttamente funzionante non dipende solo dal governo e dall'industria, anche i consumatori hanno un ruolo da svolgere scegliendo prodotti sostenibili, utilizzandoli più a lungo, riparandoli o riciclandoli alla fine del loro ciclo di vita.

L'**economia circolare** è un modello economico che mira a minimizzare il prelievo di risorse, sia quelle biologiche reintegrando nella biosfera, sia quelle tecniche prolungandone l'uso, promuovendone il riuso e rimettendole in ciclo con il riciclo, minimizzando così produzione e smaltimento dei rifiuti. L'economia circolare mira a sostituire il modello operativo lineare dell'economia tradizionale che si basa sul massiccio prelievo di risorse naturali, la loro trasformazione in prodotti che vengono consumati, generando grandi quantità di rifiuti che vengono smaltiti.

L'**economia green** mira al risparmio e all'uso efficiente delle risorse e dell'energia, allo sviluppo delle energie rinnovabili, al riciclo e al rinnovamento dei materiali per avere un benessere inclusivo di migliore qualità, tutelando il capitale naturale e i servizi ecosistemici.

L'economia green è una visione dell'economia nell'era della crisi climatica globale e della scarsità ambientale che considera quindi la questione ecologica un elemento decisivo per le possibilità di

² https://europa.eu/climate-pact/about/priority-topics/green-skills_en

³ https://www.ilo.org/global/topics/green-jobs/WCMS_824102/lang--en/index.htm





sviluppo, miglior benessere e inclusione sociale che tenga conto non solo di una maggiore equa distribuzione dei beni, ma anche dei danni causati al capitale naturale e ai servizi ecosistemici.⁴

L'economia circolare può essere considerata il pilastro di un'economia verde.

La transizione verso un'economia climaticamente neutra innescherà una trasformazione fondamentale in un'ampia gamma di settori. Verranno creati nuovi posti di lavoro, mentre alcuni posti di lavoro saranno sostituiti e altri ridefiniti.

Sta diventando necessario:

- promuovere e sostenere l'occupazione verde
- affrontare la qualificazione e la riqualificazione dei lavoratori
- anticipare i cambiamenti nei luoghi di lavoro del futuro

Per questo le **competenze green** sono ormai un requisito per l'accesso alle professioni più diverse, vista la grande importanza che le tematiche ambientali stanno acquisendo anche all'interno del settore produttivo.

3. Cosa sono le Competenze Green

Secondo la definizione di UNIDO - l'organizzazione delle Nazioni Unite per lo sviluppo industriale, le **Competenze Green** sono le *conoscenze, le capacità, i valori e le attitudini necessarie per sviluppare e sostenere una società sostenibile ed efficiente nell'uso delle risorse*.⁵

Per Competenze Green oggi si intendono tutte quelle competenze che ci permettono di rispondere all'esigenza di riconversione sostenibile della produzione in ogni tipo di realtà, dagli uffici pubblici e privati ai negozi, alle industrie, alle aziende.

Le competenze green possono essere riassunte in due aree principali:

- predisposizione al risparmio energetico
- attitudine alla sostenibilità ambientale⁶

La domanda di competenze green è ormai trasversale e riguarda tutte le professioni. Un numero crescente di aziende è alla ricerca di profili professionali in grado di lavorare con strumenti e prodotti legati all'ecosostenibilità. Ci sono interi settori produttivi come il turismo sostenibile, l'edilizia sostenibile e la mecatronica dove i fondamenti dell' Economia Green hanno un grande impatto.

Le aziende valutano come competenze green per il lavoro tutti quei fattori come l'attitudine al risparmio energetico e alla sostenibilità ambientale, quindi si rivolgono a risorse umane che dimostrano capacità, competenze e attenzione nel rendere le attività aziendali più rispettose

⁴ <https://www.fondazioneviluppосostenibile.org/circular-economy-pilastro-green-economy/>

⁵ <https://www.unido.org/stories/what-are-green-skills>

⁶ <https://jobspa.it/blog/competenze-green-piu-richieste-in-futuro>

dell'ambiente. Tra le nuove tendenze che cambiano il mercato del lavoro, infatti, non c'è solo la creazione e/o l'attivazione di nuovi lavori verdi, ma l'attitudine al risparmio energetico e alla sostenibilità ambientale che è anche la prima competenza richiesta dalle imprese subito dopo le cosiddette competenze trasversali.

Le competenze green sono presentate sia come competenze tecniche specifiche, sia come orientamento personale e propensione culturale. Non riguardano solo la capacità di impostare il rinnovamento tecnologico delle filiere produttive, o una maggiore efficienza energetica o idrica, ma sono valutate anche sulla base della capacità di modificare i comportamenti individuali e organizzativi sugli obiettivi di ecosostenibilità all'interno dei processi produttivi⁷.

Perché i lavori verdi sono essenziali per il futuro (video)

<https://www.youtube.com/embed/NrNQzCjI8A4?feature=oembed>



4. L'economia circolare nel revamping dei macchinari

Le “professioni verdi” comprendono sia professioni specifiche, necessarie per rispondere alle nuove esigenze dell’Economia Green, sia quelle che dovranno affrontare la sfida della riqualificazione delle competenze in chiave green.

⁷ https://excelsior.unioncamere.net/index.php?option=com_content&view=article&id=349:le-competenze-green&



Le professioni in cui queste competenze sono maggiormente richieste sono: ingegneri civili, ingegneri elettronici e delle telecomunicazioni, tecnici della direzione lavori, tecnici della sicurezza sul lavoro, ingegneri energetici e meccanici.

La speranza delle imprese europee è che si sviluppino posizioni legate alla riduzione degli impatti ambientali in ambito tecnologico-digitale.

A titolo di esempio, nelle aziende più strutturate, si può trovare la figura del mobility manager: una persona preposta alla mobilità aziendale, che ha il compito di coordinare in modo più sostenibile gli spostamenti dei dipendenti da casa al lavoro⁸.

Il principio fondamentale su cui si basa il modello economico dell'economia circolare, che mira a minimizzare il prelievo di risorse favorendo il riutilizzo delle risorse biologiche reintegrando nella biosfera e di quelle tecniche prolungandone l'uso⁹, si presta perfettamente al settore del refitting e del revamping di macchinari.

Il concetto di economia circolare è la base per spiegare alcuni dei vantaggi del ricondizionamento di macchinari obsoleti. Ciò riguarda gli aspetti economici, sociali e ambientali.

L'adozione dei sistemi circolari nel settore consiste in riutilizzo, condivisione, riparazione, rinnovamento, ristrutturazione, rigenerazione e riciclaggio per creare un sistema a circuito chiuso, riducendo al minimo l'uso di input di risorse e la creazione di rifiuti, inquinamento ed emissioni di carbonio. Molti di questi aspetti sono inclusi nel revamping dei macchinari.

Il recupero di macchinari obsoleti avrà una forte influenza su tutto questo modello in quanto risponde non solo a programmi ambientali per la gestione dei rifiuti, l'efficienza energetica e la riduzione delle materie prime, ma contribuirà in modo decisivo anche all'economia circolare.

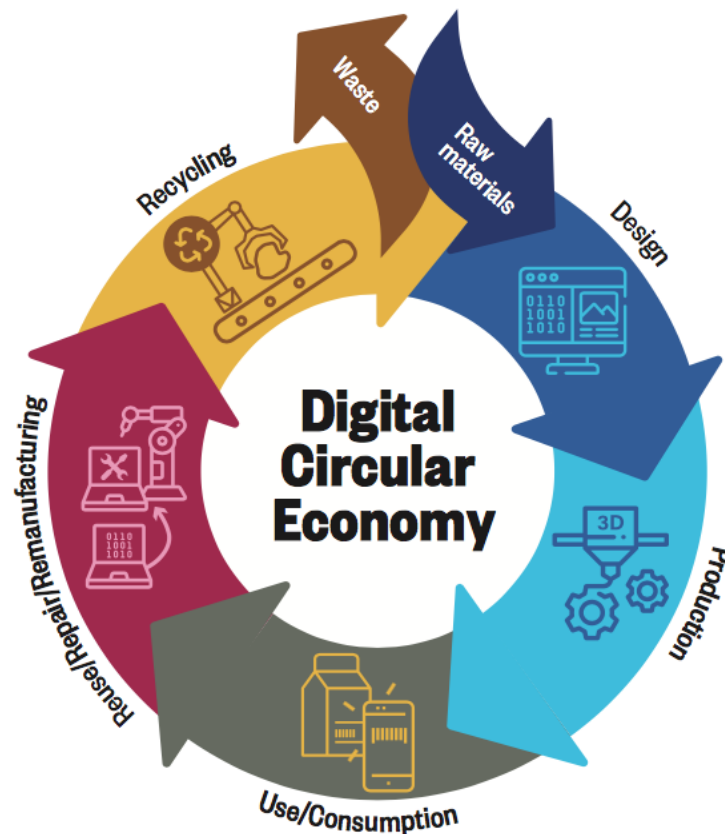
Il revamping dei macchinari porta positivi benefici finanziari ed economici. Aumentando la produttività dell'industria, le macchine modernizzate aumentano anche la produzione complessiva dell'economia; di conseguenza, l'occupazione, il reddito nazionale e il tasso di crescita dell'economia aumentano.

Il rinnovamento è generalmente più economico rispetto all'acquisto di nuovi prodotti o attrezzature. Ad esempio, durante il revamping di un veicolo, vengono riutilizzate parti funzionali (come i sedili, il telaio, gli ingranaggi, i finestrini, ecc.). Ciò non solo aumenta la sicurezza grazie all'utilizzo di soluzioni tecnologiche più avanzate, ma riduce anche i costi operativi.

Inoltre, il revamping o il retrofitting sono pratiche comuni che abbracciano i principi chiave dell'"economia circolare", per cui i macchinari sono progettati pensando alla facilità di manutenzione e con gran parte dei materiali da riciclare alla fine del loro ciclo di vita. I pezzi di ricambio di macchinari vecchi/obsoleti vengono riparati o riprodotti per un uso più efficace, riducendo così la quantità di rifiuti e prolungando la durata del prodotto utilizzando meno risorse. Queste pratiche sono aspetti chiave anche per migliorare le prestazioni ambientali dei macchinari, pur non garantendo le richieste dei clienti in termini di produttività e affidabilità.

⁸ <https://laborability.com/approfondimenti/leuropa-alla-ricerca-di-competenze-green-e-digital>

⁹ <https://blog.ener2crowd.com/circular-economy-e-green-economy/>



Fonte <https://www.climate-kic.org/wp-content/uploads/2019/07/DRCE.pdf>

Il revamping dei macchinari è uno degli elementi chiave di un'economia circolare. Le aziende che utilizzano questo sistema sono in grado di vendere al dettaglio e riciclare i macchinari o le loro parti e modernizzare le proprie attrezzature. In questo modo, le aziende ridurranno i costi, aggiorneranno le macchine in base alle loro esigenze e seguiranno il metodo della produzione circolare digitale e l'ammodernamento delle macchine.

Cosa succede se non acquistiamo prodotti e acquistiamo servizi? Spiegazione dell'economia circolare (Video) https://www.youtube.com/embed/Cd_isKtGaf8?feature=oembed



5. Vantaggi del revamping per un'economia più verde

Benefici economici

- **Aggiornamento del software e/o dell'hardware**

L'installazione di un nuovo software o hardware potrebbe essere molto costosa per le aziende. Aggiornando installazioni software/hardware già presenti si riducono i costi senza sostenere le spese per una nuova versione del software.

- **Aumento della produttività**

La sostituzione di macchinari obsolete può essere piuttosto costosa. Per le aziende, il revamping è la soluzione migliore per aggiornare le proprie macchine e impianti in base alle esigenze al minor costo.

L'uso di macchine dopo il revamping aiuta i produttori ad aumentare il loro reddito perché un macchinario aggiornato funzionerà più velocemente. Gli utenti, azionando un computer o una parte della nuova macchina, potranno completare il lavoro più velocemente rispetto a molti lavoratori impegnati a svolgere lo stesso lavoro manualmente

- **Aumento dell'efficienza del lavoratore**



Aumentando l'efficienza dei lavoratori, questi ultimi possono svolgere i propri compiti in modo migliore rispetto a quanto farebbero manualmente. In questo modo produrranno in modo qualitativamente più preciso e veloce in quantità maggiori, inoltre i lavoratori con più capacità e competenze potranno anche aumentare il loro reddito

- **Creare opportunità di lavoro**

I macchinari creano occupazione e aumentano la produttività, riducendo i costi per l'industria e rendendo più economici beni e prodotti; questo porta ad un aumento della domanda. L'industria ha bisogno di più lavoratori per far fronte alla domanda.

Alcune delle categorie di crescente domanda sono ingegneri meccanici, tecnici di ingegneria e operazioni aerospaziali, tecnici elettromeccanici, ingegneri di vendita, scienziati informatici e di ricerca informatica, programmatori di computer.

Benefici sociali

La necessità di aggiornare o "rinnovare" tali macchine, senza sprecare risorse per l'acquisto di nuove attrezzature, comporta una serie di vantaggi per le imprese e la società stessa. Il revamping o "retrofitting" è infatti diventato un approccio altamente significativo per raggiungere la sostenibilità a tutti i livelli sociali, economici e ambientali e migliorare il tenore di vita delle persone.

- Il riutilizzo dei componenti dei macchinari rende il processo di modernizzazione più economico e attraente, ottimizza anche le prestazioni energetiche e aiuta a prolungare la vita delle macchine
- Il rinnovamento di macchinari crea opportunità educative. Richiede manodopera istruita e qualificata per le operazioni, riparazioni, manutenzione e modernizzazione. Ciò porta alla domanda di istruzione tecnica formale o non formale, che a sua volta crea una domanda di personale docente competente. Le opportunità educative relative al revamping si estendono a un gran numero di campi dell'istruzione, ad esempio ingegneria, apprendimento automatico, progettazione 3D, sviluppo software, ecc.
- La sostituzione o il rinnovo di componenti obsoleti di un macchinario, che si traduce in una migliore efficienza dell'intera macchina, crea anche un ambiente di lavoro migliore. Il revamping garantisce che una macchina funzioni secondo le più recenti esigenze tecnologiche e gli attuali standard di pratica. In questo modo, il lavoro quotidiano dei dipendenti che si occupano di macchine e lavori di costruzione viene migliorato, comprese le loro condizioni di lavoro e gli aspetti di sicurezza.





- Il rinnovamento crea occupazione. Sebbene una nozione generale relativa all'industria delle macchine tenda a sostenere che le macchine sostituiscono la forza lavoro umana, è ancora necessaria molta creatività umana quando si modernizzano i macchinari. Quando si aggiorna una macchina, ci sono molti livelli che devono essere conservati: struttura, dimensioni, sistemi di sicurezza ecc. Questo trasforma il revamping in una tecnica quasi manuale, unica per ogni marchio e contesto. I lavoratori sono sempre necessari per fabbricare nuove macchine o per mantenere, riparare e modernizzare quelle più vecchie.

Benefici ambientali

Oggi la sostenibilità ambientale è una parte importante di qualsiasi attività economica e in particolare nel campo dell'industria.

Il recupero di macchinari obsoleti avrà una forte influenza su tutto questo modello in quanto risponde non solo a programmi ambientali per la gestione dei rifiuti, l'efficienza energetica e la riduzione delle materie prime, ma contribuirà in modo decisivo anche all'economia circolare.

Una delle maggiori sfide che l'industria deve affrontare oggi è la necessità di migliorare ulteriormente le proprie **prestazioni ambientali** per diventare veramente compatibile con lo sviluppo sostenibile. L'industria deve essere un attore attivo nel processo attraverso l'imprenditorialità responsabile e l'eco-efficienza. L'aumento delle prestazioni ambientali significherà ridurre gli impatti ambientali negativi che si verificano in ogni fase del ciclo di vita del prodotto, dall'estrazione delle materie prime attraverso i processi di produzione, il trasporto e la distribuzione dei prodotti fino all'utilizzo e allo smaltimento dei prodotti.

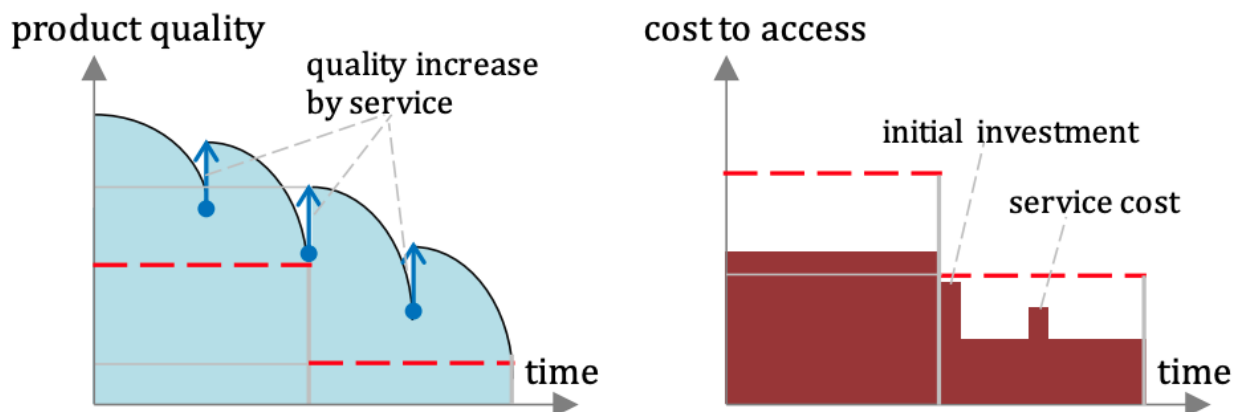
Dobbiamo pensare alla necessità di promuovere un modello di attività economica più sostenibile che risponda alla doppia sfida di ridurre le proprie emissioni e utilizzare le proprie risorse in modo efficiente.

In questo modo, il modello passerà dall'essere parte del problema all'essere parte della soluzione.

Quando si parla di sostenibilità, non ci si riferisce esclusivamente alle questioni ambientali, come l'efficienza energetica o il cambiamento climatico. Il principio della sostenibilità si basa sulle connessioni tra ambiente, società ed economia.

Il revamping dei macchinari assicura che le macchine funzionino senza intoppi secondo le moderne esigenze della tecnologia. Quando le apparecchiature moderne sono aggiornate, incorporando le ultime tecnologie e funzionalità, funzionano in modo più efficiente e hanno maggiori probabilità di eliminare potenziali guasti.

Di conseguenza, i componenti all'avanguardia utilizzati per il revamping si traducono nell'espansione delle capacità prestazionali delle macchine e portano a un posto di lavoro complessivamente più efficiente, che è più sicuro e di facile utilizzo per lavoratori e dipendenti, pur essendo redditizio per i datori di lavoro.



Le macchine modernizzate aiutano ad aumentare la produzione e la durata di beni e prodotti. Incorporando tecnologie all'avanguardia nei macchinari esistenti, le vecchie macchine diventano più produttive e durano più a lungo, consentendo la produzione in serie e automatizzata dei prodotti, portando così a una produzione su larga scala, riducendo i costi e aumentando i profitti.

6. Un modello economico più sostenibile - The Triple Bottom Line (TBL)

Nel 1994, l'autore e imprenditore, John Elkington, si è basato sul concetto di triplice approccio - **triple bottom line (TBL)** nella speranza di trasformare l'attuale sistema aziendale incentrato sulla contabilità finanziaria, per adottare un approccio più completo nella misurazione dell'impatto e del successo. Storicamente, le aziende operavano al servizio esclusivamente dei loro profitti finanziari. Tuttavia, come risultato della teoria e dell'applicazione del triplice approccio, alcune aziende hanno iniziato a realizzare la connessione tra salute ambientale, benessere sociale, successo finanziario e resilienza dell'organizzazione.

La teoria della triple bottom line espande le metriche del successo aziendale per includere contributi alla salute ambientale, al benessere sociale e a un'economia giusta. Queste categorie di fondo sono spesso indicate come le **tre "P"**: **Persone, Pianeta e Prosperità**¹⁰.

La ratio di tale teoria consiste nell'incentivare le imprese ad operare nel contesto economico di riferimento attraverso strategie e scelte capaci di valorizzare contemporaneamente (i) l'ambiente (pianeta), (ii) il contesto sociale (persone) e (iii) il contesto economico -aspetto finanziario (profitto). Tre elementi che, se considerati congiuntamente dall'azienda, avrebbero consentito la creazione di maggior valore della produzione, operando, al contempo, una maggiore attrattività per investitori e consumatori e favorendo un ambiente più consapevolmente orientato alla sostenibilità ecologica e sociale tra dipendenti.¹¹

¹⁰ <https://sustain.wiscorsin.edu/sustainability/triple-bottom-line/>

¹¹ <https://www.previti.it/dal-concetto-di-triple-bottom-line-ai-fattori-esg-le-nuove-politiche-globali>



Fonte immagine https://www.researchgate.net/figure/The-interconnection-of-the-elements-of-the-Triple-Bottom-Line-concept_fig1_329185478

- **Persone**

Questo approccio misura l'impatto delle imprese sul capitale umano. Un'azienda che utilizza la triple bottom line ha una responsabilità non solo nei confronti degli azionisti, ma anche dei dipendenti, dei fornitori, dei clienti, della comunità in cui opera e di chiunque altro sia influenzato dall'organizzazione, direttamente o indirettamente. Riconosce l'interdipendenza di tutte le relazioni e interazioni umane che consentono all'azienda di operare. Ciò può tradursi in azioni come fornire prestazioni sanitarie di qualità e orari di lavoro flessibili ai dipendenti, offrire opportunità di avanzamento professionale o educativo, creare un ambiente di lavoro sicuro e impegnarsi in pratiche di lavoro eque.

- **Pianeta**

Le aziende che seguono il modello TBL lavorano per ridurre la loro impronta ecologica. Riconoscono che minore è l'impatto ambientale di un'azienda, più a lungo può operare. Al suo livello più elementare, ciò implica non produrre prodotti pericolosi o dannosi per il pianeta e le persone che lo abitano, ma include anche la riduzione dei consumi, dei rifiuti e delle emissioni. Prevede azioni specifiche, come l'utilizzo di fonti energetiche rinnovabili, la riduzione del consumo energetico, lo smaltimento sicuro di materiali tossici e l'adozione di una serie di politiche aziendali verdi.

- **Prosperità**



Tutte le aziende sono preoccupate per la loro posizione finanziaria, ma le aziende impegnate nel triplice approccio guardano ai profitti non solo in termini di ciò che possono fare per gli azionisti, ma anche di come possono aiutare la comunità più ampia. In questo modello, un'azienda aiuta a stimolare la crescita economica e creare ricchezza retribuendo equamente i dipendenti, supportando i fornitori locali con la propria attività, generando innovazione e pagando la giusta quota di tasse. Prende anche decisioni finanziariamente prudenti ma guidate eticamente su come e dove ottenere materiali, prodotti o manodopera¹².

Negli ultimi anni, principalmente a causa del crescente interesse per le tematiche ambientali e sociali e al fine di individuare un criterio che fosse - ancor più del Triple Bottom Line - capace di valutare un investimento come socialmente responsabile, sono state elaborate da dottrine economiche più all'avanguardia i cosiddetti fattori ESG (Environmental, Social, Governance). Oggi le istituzioni bancarie e finanziarie, e ancor più organizzazioni specializzate come le agenzie di rating ESG, utilizzano sempre di più il suddetto paradigma come metro di paragone per orientare le scelte di investimento e l'allocazione del capitale. Da ciò ne consegue che un'impresa, ad esempio, per accedere a determinate forme di finanziamento o incentivi pubblici, deve necessariamente effettuare investimenti sostenibili e responsabili, rispettosi degli aspetti ambientali e sociali, nonché finalizzati a generare profitti.

Le componenti ambientali e sociali giocano un ruolo sempre più preponderante anche nelle scelte dei consumatori che prediligono aziende che svolgono la propria attività in modo sostenibile e responsabile. Questo crea un forte legame tra investimenti socialmente responsabili e reputazione aziendale. I fattori ESG come criterio guida per l'individuazione delle strategie e delle politiche aziendali contribuiscono in modo significativo a migliorare e accrescere la reputazione dell'azienda presso investitori e consumatori.

La crescente tutela dell'ambiente e il rispetto del capitale umano sono due aspetti che, oggi in particolare, richiedono un'attenzione specifica in ogni ambito. Grazie al concetto di Triple Bottom Line e ai fattori ESG, le tematiche sociali e ambientali giocano un ruolo di crescente importanza nel settore economico e la loro valutazione rappresenta un passaggio essenziale nelle strategie di investimento sostenibile e responsabile delle aziende¹³.

¹² www.business.com/articles/triple-bottom-line/

¹³ <https://www.previti.it/dal-concetto-di-triple-bottom-line-ai-fattori-esg-le-nuove-politiche-globali>

5.2. Metodi interattivi per lo sviluppo delle Competenze Green

Esempio di applicazione pratica delle competenze green digitali: Il retrofitting di una vecchia Radio FM (caso studio)

Per Green Skills oggi si intendono tutte quelle competenze che ci permettono di rispondere all'esigenza di riconversione sostenibile della produzione in ogni tipo di realtà, dagli uffici pubblici e privati ai negozi, alle industrie, alle aziende.

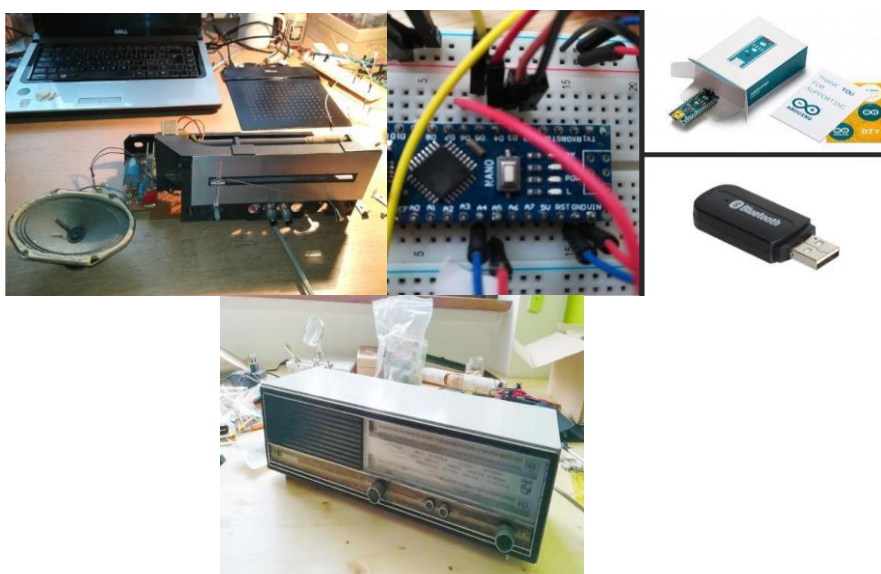
Le competenze green possono essere riassunte in due aree principali:

- predisposizione al risparmio energetico
- attitudine alla sostenibilità ambientale¹⁴

Proprio nell'ambito della sostenibilità ambientale, si colloca la necessità di applicare competenze green al retrofit di vecchi macchinari e oggetti con l'intento di dar loro nuova vita e un nuovo ciclo di utilizzo, evitando l'acquisto di un nuovo prodotto che alimenterebbe il meccanismo intrinseco al classico modello economico lineare.

Un esempio pratico per lo sviluppo di competenze green in ambito digitale, può essere visto nel caso di studio relativo al Retrofitting di una vecchia radio FM che fornisce connettività Bluetooth tramite un microcontrollore ARDUINO.

Il modello utilizzato è l'Arduino Nano che controlla l'accensione della chiavetta Bluetooth e il funzionamento del modulo radio FM. Tramite Arduino si può poi selezionare se far funzionare la radio o il Bluetooth tramite un pulsante per l'interruttore. In questo modo la nostra radio può essere utilizzata sia per ascoltare la radio FM sia per ascoltare la musica dal nostro smartphone.



Autore: Giulio Pons

Fonte: <https://hackaday.io/project/162367-retrofitting-of-an-old-fm-philips-radio>

¹⁴ <https://jobspa.it/blog/competenze-green-piu-richieste-in-futuro>



2021-1-RO01-KA220-VET-000033036

<https://www.youtube.com/watch?v=MuxsjAEGdNI>

Promuovere e sviluppare le Competenze Green Skills attraverso il gioco: Penji protegge il pianeta! (gioco online)

Penjii protegge il pianeta è un gioco per dispositivi mobili che mira a insegnare ai giocatori come salvare il loro pianeta! Questo gioco è stato sviluppato da [Caped Koala Studios](#) come risultato di un progetto [Erasmus+ project "Promoting Green Skills Through Games"](#) con partner provenienti da Austria, Croazia, Irlanda e Spagna che lavorano tutti insieme per promuovere Competenze Green.

In questo gioco il pinguino Penjii corre intorno al pianeta per combattere i problemi che affliggono il nostro pianeta.

Penjii visita 4 diverse parti del mondo (l'Antartide, una bellissima spiaggia, una città caotica, la tua casa) e ogni livello mira a insegnare ai giocatori un problema specifico che affligge il nostro ambiente e il nostro pianeta.



<https://capedkoala.com/penjii-protects-the-planet/>

Sviluppare competenze verdi con l'apprendimento attraverso la pratica: il potere rinnovabile delle competenze green per le donne in Zambia (video)

Le donne del complesso di Kalulushi, nella provincia di Copperbelt, hanno costruito le proprie case con l'ausilio di tecnologie verdi. Con l'aiuto dell'ILO attraverso lo Zambia Green Jobs Program e la Zambia Homeless and Poor People Federation, hanno ottenuto un prestito per acquistare un piccolo appezzamento di terreno e sono state formate sulle competenze e tecnologie verdi, e tramite il metodo "imparando facendo", hanno costruito delle case utilizzando mattoni realizzati da materiali sostenibili.



<https://www.youtube.com/watch?v=4K3AiZaiMoc>

7.Fonti

- <https://www.europarl.europa.eu/news/en/headlines/priorities/climate-change/20180703STO07129/eu-responses-to-climate-change>
- https://europa.eu/climate-pact/about/priority-topics/green-skills_en
- https://www.ilo.org/global/topics/green-jobs/WCMS_824102/lang--en/index.htm
- <https://www.fondazionevilupposostenibile.org/circular-economy-pilastro-green-economy/>
- <https://www.unido.org/stories/what-are-green-skills>
- <https://jobspa.it/blog/competenze-green-piu-richieste-in-futuro>
- https://excelsior.unioncamere.net/index.php?option=com_content&view=article&id=349:le-competenze-green&
- <https://laborability.com/approfondimenti/leuropa-alla-ricerca-di-competenze-green-e-digital>
- <https://blog.ener2crowd.com/circular-economy-e-green-economy/>
- <https://sustain.wisconsin.edu/sustainability/triple-bottom-line/>
- <https://www.previti.it/dal-concetto-di-triple-bottom-line-ai-fattori-esg-le-nuove-politiche-globali>
- www.business.com/articles/triple-bottom-line/
- <https://www.previti.it/dal-concetto-di-triple-bottom-line-ai-fattori-esg-le-nuove-politiche-globali>