

# InquiNO

## Guida introduttiva alla realizzazione del progetto InquiNO

Rivolta sia a liberi appassionati, sia a professori, per essere di supporto alla realizzazione di un prototipo,

### NOTA DI UTILIZZO

La guida è suddivisa in alcuni step, più qualche appendice di approfondimento. I più esperti possono limitarsi a leggere le sezioni “in breve” e “per esperti” all’inizio di ogni capitolo, mentre chi vuole più supporto e una spiegazione più completa ed estesa, può farsi un’idea introduttiva con le sezioni “in breve”, saltare le sezioni “per esperti” e continuare la lettura seguendo il percorso proposto.

## INTRODUZIONE

### IN BREVE:

Il progetto si propone di realizzare un sistema open per la condivisione di dati geolocalizzati relativi all’inquinamento ambientale.

Questo progetto nasce con lo scopo di gettare le basi per realizzare un sistema centralizzato per il monitoraggio dello stato di inquinamento del territorio, basi che possono poi essere riproposte alle scuole della zona che possono contribuire attraverso nuovi progetti, per la realizzazione nuove centraline per il monitoraggio o per il potenziamento di quanto costruito.

Percorso guidato di realizzazione di una semplice centralina per il controllo della qualità dell’aria, basata su una scheda Arduino e sensori facilmente reperibili.

Nello specifico, si vuole guidare i ragazzi nella creazione del circuito elettronico e nella scrittura del codice necessario, supportati da librerie di codice già create per facilitare il lavoro e stimolare l’apprendimento del concetto e dell’approccio, più che della tecnica.

Questo progetto propone inoltre un concetto particolarmente interessante: un dispositivo che prende dei dati dal mondo reale, li digitalizza e interpreta, e tramite Wi-Fi li trasmette ad un server centrale che li memorizza, elabora e mostra.

La centralina può connettersi tramite Wi-Fi o altre tecnologie, ad un server appositamente dedicato alla ricezione dei dati, che agglomera il tutto e mette a disposizione una pagina web con una mappa del territorio su cui visualizzare i dati. In futuro, saranno disponibili anche ulteriori modalità di visualizzazione e di studio dei dati.

Non sono necessarie conoscenze o capacità pregresse né lato studenti/appassionati né lato docenza (in caso si tratti di una iniziativa o attività scolastica), poiché si tratta di realizzare un oggetto già documentato e concreto, con oltretutto il supporto di librerie e materiale adatto a semplificare il lavoro.

Ovviamente, anche il supporto dei Tutor del FablabVrea è assicurato, sentitevi liberi di scriverci a [tutors.fablabivrea@gmail.com](mailto:tutors.fablabivrea@gmail.com) .

Tutta la parte online di server e piattaforma di visualizzazione dei dati è fornita dall'Accademia, presentando un livello di difficoltà maggiore ed essendo unica e condivisa tra tutti gli aderenti al progetto.

## INDICE

### IN BREVE:

Nella sua prima iterazione, il progetto si compone da quattro moduli essenziali:

- Modulo di acquisizione dei dati
- Modulo di trasmissione dei dati
- Server per il trattamento dei dati
- Interfaccia web per la fruizione dei dati

I quattro moduli sono studiati per essere sviluppati in modo del tutto indipendente, con interfacce di comunicazione preconcordate. In questo modo sarà possibile sostituire un solo modulo per volta per effettuare eventuali miglioramenti/potenziamenti. Le sezioni di circuito e codice svilupperanno contemporaneamente i primi due moduli, mentre il terzo e il quarto saranno affrontati nella sezione di funzionamento del sito.

La guida è suddivisa in alcuni step, più qualche appendice di approfondimento.

I più esperti possono limitarsi a leggere la sezione “per esperti” all’inizio di ogni capitolo, mentre chi vuole più supporto e una spiegazione più completa ed estesa, può continuare la lettura e seguire il percorso proposto.

### Punti principali:

- Introduzione
  - descrizione del progetto e indice della guida e modalità di utilizzo
- Get Started
  - Lista materiali consigliati e quick setup per esperti
- Basi di Arduino
  - Basi di elettronica e di programmazione scollegate dal progetto ma essenziali
- Impostazione ambiente InquiNO
  - Installazione librerie necessarie e check-up del materiale
- Realizzare il circuito
  - Schema del circuito da realizzare e guida passo passo alla realizzazione
- Realizzare il codice
  - Quale codice caricare e guida alla sua realizzazione
- Funzionamento del sito
  - Intro al sito web e alle sue API (funzionalità)
- Next steps
  - Linee guida per installare la centralina, e aiutarci a diffondere l’iniziativa

### Appendici:

- Sensori questi sconosciuti
  - descrizione dei sensori utilizzati

## GET STARTED

### PREMESSA

get started rapidissimo, PER ESPERTI che vogliono solo il necessario per realizzare un dispositivo, se sei qui per imparare, salta questa sezione, ti verrà tutto spiegato con calma. Tutta la sezione è IN BREVE

1. Riassunto dei 4 moduli che compongono il progetto
2. Elenco dei materiali e strumenti utili alla realizzazione
3. Circuito e codice

### Acquisizione dei dati

L'acquisizione dei dati viene effettuata attraverso un dispositivo Arduino, munito di sensori adeguati e programmato attraverso l'utilizzo di librerie dei sensori. I dati recuperati da questo software verranno inviati al modulo successivo tramite la relativa interfaccia, descritta nel prossimo paragrafo.

### Trasmissione dei dati

Per poter trasmettere i valori dei sensori al server si presuppone di avere a disposizione una connessione WiFi, a cui la scheda Arduino potrà collegarsi tramite un modulo wifi connesso alla scheda tramite porta seriale. Verrà usata una libreria apposita per l'utilizzo di questo modulo, che si propone di ricevere i valori dei sensori dal sistema di acquisizione per inviarli al server centrale tramite una chiamata alle API REST disponibili sul server

### Stoccaggio e trattamento dei dati

E' stato sviluppato un server centrale per poter raccogliere tutti i dati trasmessi dai vari dispositivi e poi metterli a disposizione dai software di fruizione (in questo momento, solamente l'interfaccia web, ma in futuro i software potrebbero essere molteplici). Le API dovranno occuparsi di tutte le funzioni di comunicazione tra utenti e server centrale.

### Interfaccia web per la fruizione dei dati

Per poter fruire i dati si propone di utilizzare un'interfaccia web in grado di scaricare i dati dal server centrale e mostrarli, possibilmente abbinati ad una mappa del territorio. Questo e' il modulo che ha il maggior numero di possibili interazioni: e' possibile partire da una semplice cartina con indicati i valori di inquinamento posizionati geograficamente nelle posizioni piu' corrette per arrivare a veri e propri strumenti di data warehouse per l'analisi andamentale dei valori registrati. Ci sono molti strumenti di analisi open collegabili a questa interfaccia, che andranno valutati in futuro.

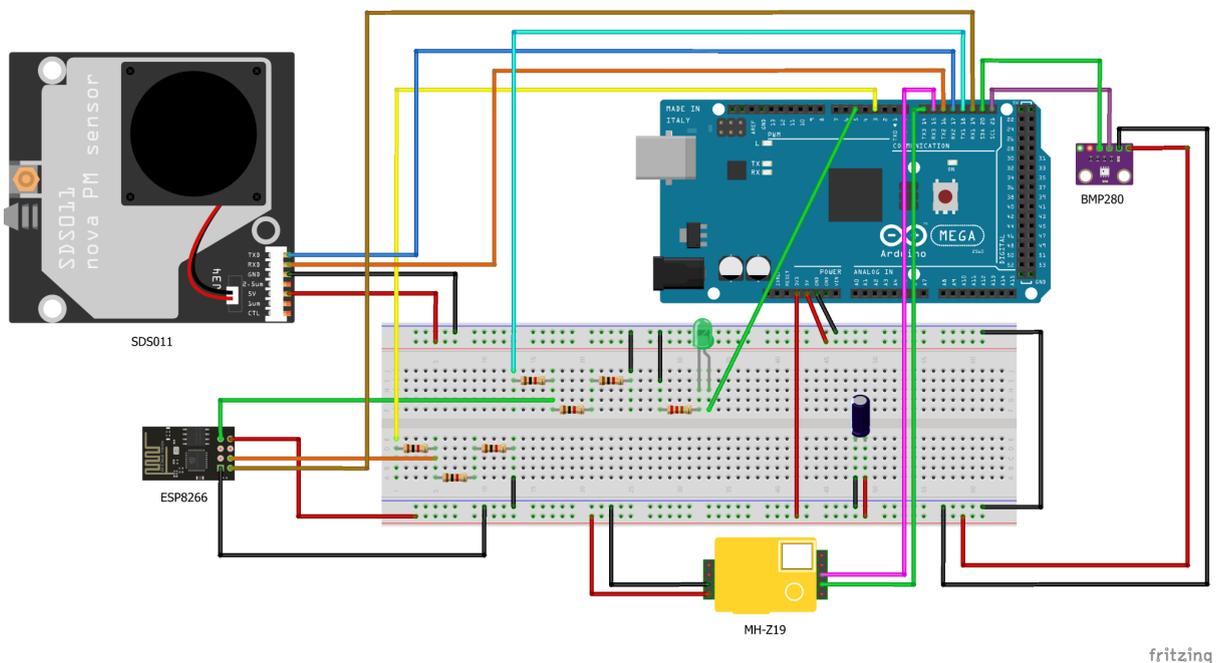
## Lista materiali:

- Arduino MEGA (o equivalente, ma mega, attenzione, deve avere 3 porte seriali hardware)
- Breadboard
- Cavetti
- LED (x1)
- Resistenze
  - 1kΩ (x6)
  - 220Ω (x1)
- Condensatori (x1) → INSERIRE CAPACITÀ CONDENSATORE
- ESP8266 (per connessione Wi-Fi)
- Sensore di temperatura e pressione BMP280
- Sensore di CO<sub>2</sub> MH-Z19 o MH-Z19C
- Sensore di polveri sottili SDS011

In appendice ci sono alcuni link a pagine Amazon o a negozi di elettronica online dove poter acquistare i componenti elettronici di base e i sensori elencati.

Detto in estrema sintesi, tutta l'attività può essere riassunta in: Scarica l'Arduino IDE, prendi un'Arduino, breadboard, sensori e cavo, scarica queste librerie e il codice dalla nostra repo di Github e adattalo per collegarsi al tuo WiFi e funzionare autonomamente.

## Circuito



## Codice

Come anticipato, consigliamo di scaricare dalla nostra repo il materiale. Tutte le librerie necessarie ed il codice per il funzionamento si trovano nella cartella "device".

Consigliamo di importare le librerie dai file da noi proposti nella cartella device/lib in quanto in alcuni casi sono stati corretti errori.

## BASI DI ARDUINO

PREMESSA: sezione introduttiva ma propedeutica, se sai già cos'è un'Arduino e come si programma, salta pure questa sezione.

Prima basi di elettronica, gioca con breadboard, led, resistenza

Poi basi di programmazione, accendi e spegni il led, funzioni

BAM! LIBRERIE! ecco lo strumentopolo giusto!

TODO

## IMPOSTAZIONE AMBIENTE InquiNO

### IN BREVE:

In questo capitolo scaricheremo il materiale da noi preparato usando i comandi git dal terminale, e poi installeremo le librerie necessarie.

### PER ESPERTI:

Clonare la repo <https://github.com/Fablab-lvrea-DIT/progetto-ambiente> e installare le librerie presenti nella cartella device/lib, consigliamo di scaricare quelle perchè sono stati corretti alcuni errori.

LEGGI IL GET STARTED SE TI FA PIACERE! DOPO AVER IMPARATO LE BASI DI ARDUINO, IL PROF OAK DICE CHE E' IL MOMENTO GIUSTO PER FARE UN BEL CHECK DEL MATERIALE E FARSÌ UNA IDEA DEL PROGETTO!

Innanzitutto, bisogna scaricare il materiale.

Andare sulla pagina del progetto su github, al seguente link

<https://github.com/Fablab-lvrea-DIT/progetto-ambiente>

Le seguenti istruzioni sono per Windows, ma potete contattarci per istruzioni specifiche di altri sistemi operativi.

Innanzitutto, per copiare il nostro codice dalla nostra "cartella virtuale" su github, serve uno strumento chiamato Git. Bisogna installarlo sul proprio pc, è solo uno strumento tecnico molto piccolo e innocuo, ma necessario. In caso fosse un problema, è possibile scaricarne una versione da questo link, ma invito ad andare sulla pagina ufficiale per scaricare l'ultima versione.

<https://github.com/git-for-windows/git/releases/download/v2.35.1.windows.2/Git-2.35.1.2-32-bit.exe>

Dopo averlo scaricato, e lanciato, avremo questo nostro comando speciale.

Poi, bisogna creare dove più comodo sul proprio pc una cartella, aprirla, cliccare col tasto destro e scegliere “aprire finestra di terminale qui”.

Si aprirà una schermata nera in cui potrete scrivere il seguente comando. Attenzione, il copia incolla non funziona come al solito in questa schermata, se volete incollare il comando che trovate qui sotto invece di copiarlo a mano, dopo averlo copiato potete incollarlo cliccando semplicemente col tasto destro al posto di fare “incolla”.

```
git clone https://github.com/Fablab-Ivrea-DIT/progetto-ambiente.git
```

Complimenti, avete ora tutto il nostro materiale sul vostro computer!  
Adesso, bisogna installare le librerie necessarie.

Aprirete Arduino IDE (scaricabile dalla pagina ufficiale di Arduino oppure, in una versione, da questo link <https://downloads.arduino.cc/arduino-1.8.19-windows.exe> )

Installatelo e apritelo. Ora, per aggiungere al vostro sistema le librerie utilizzate bisogna cliccare su Sketch -> #include libreria -> aggiungi libreria da file .ZIP e selezionare uno dei file nella cartella device/lib. Ripetere questa operazione per tutte le librerie.

Il lavoro sarà supportato da un'ulteriore libreria. Negli step successivi ne parleremo più approfonditamente, ma aprendo il codice da noi messo a disposizione, si potrà vedere un file principale e due file di librerie. Si tratta di codice messo a disposizione da noi per facilitare il vostro lavoro: ci sono funzioni per gestire tutti i sensori e le funzioni per interagire con il WiFi, già tutto pronto!

Questa libreria però è già pronta: non va installata, basta mettere nella cartella del progetto i 4 file delle 2 librerie da noi predisposte, così come potrete vedere già fatto nel codice che avete appena scaricato, che guarderemo a breve.

Complimenti, l'ambiente è pronto!!!

## LA NOSTRA LIBRERIA

### IN BREVE:

Abbiamo predisposto due librerie, una per i sensori ed una per il WiFi.

### PER ESPERTI:

Abbiamo predisposto due librerie, di cui trovate i file .h e .cpp nella cartella device/progetto\_ambiente, apribili anche come tab aggiuntivi sull'ide. Dateci un'occhiata, nei .h sono elencate tante funzioni utili, soprattutto per verificare il funzionamento dei sensori!

Come potrete vedere aprendo il file principale (.ino) nella cartella device/progetto\_ambiente abbiamo creato due nostre librerie, a supporto del progetto.

Le librerie sono “pacchetti” di funzionalità già predisposte da altri e pronte all’utilizzo. Per capirci, sono un vocabolario potenziato. E’ come se ad una interrogazione di Storia, invece di studiare la vita di Cesare, poteste caricare nel vostro telefono una app “storia di cesare” e, al momento dell’interrogazione, dire “racconta storia di cesare” alla app, e basta, sarà la app ad occuparsi di raccontare la storia di cesare alla professoressa.

Ogni libreria è composta da tante sotto librerie, ma le nostre, essendo semplici e “didattiche”, si sviluppano su due file. Un file .h riassume tutte le “parole” che la libreria mette a disposizione, cioè le funzioni. Si tratta di un semplice elenco in cui è possibile vedere tutto ciò che è possibile fare con quella libreria. Poi, nel file .cpp, associato a ognuno di quei nomi, c’è un blocco di codice che “fa le cose”.

Per capirci, basta importare nel codice quelle librerie e poi è possibile usare funzionalità semplificate. Per inizializzare un sensore per esempio, anzichè usare tutte le righe di codice necessarie, sarà possibile scrivere `sensors.inizializza sensore`.

In un esempio pratico, se vogliamo testare il sensore SDS di pm, ci basterà mettere nel setup la funzione di inizializzazione `sensors.SDSsetup()` e poi nel loop la funzione `sensors.SDStest()`. Ricordate di mettere un bel `delay(5000)` dopo!!

## REALIZZARE IL CIRCUITO

### IN BREVE:

In questa sezione presentiamo le connessioni necessarie a creare il circuito elettronico della centralina. Prima verrà presentato il circuito completo, ma consigliamo di proseguire e montare il circuito un sensore alla volta! Per ogni sensore, ci sarà anche un breve paragrafo su un rapidissimo codice per poterlo testare “singolarmente”.

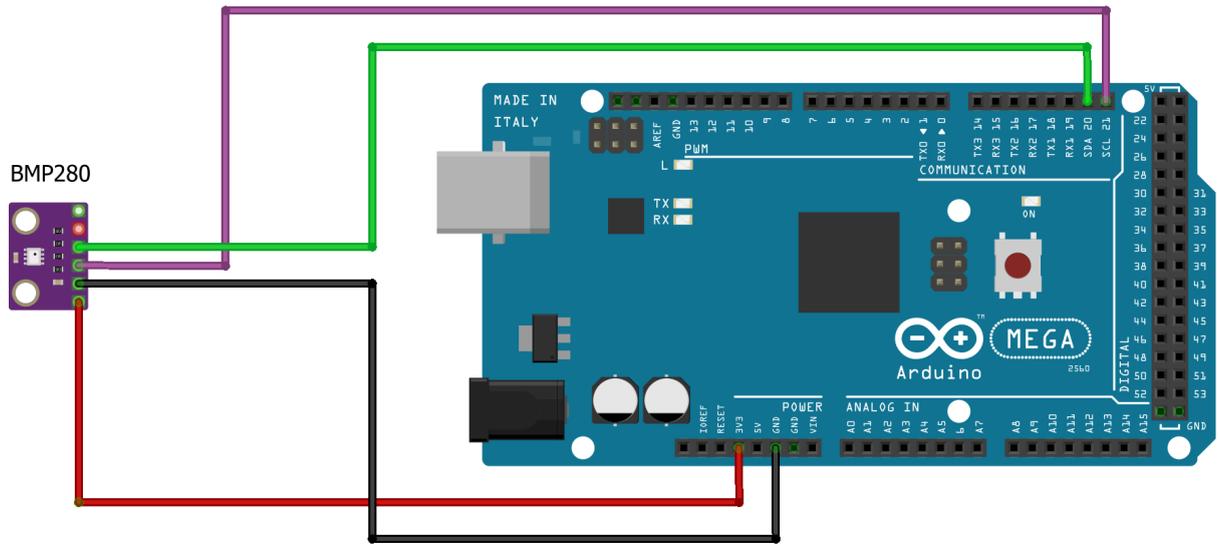
### PER ESPERTI:

A seguire c’è il circuito completo, ma consigliamo di collegare un sensore alla volta usando le funzioni di test della libreria per testarne il corretto funzionamento.

Di seguito tutti i circuiti e qualche indicazione per poterli testare singolarmente. Il consiglio è di collegare un sensore alla volta. Nello specifico, consigliamo di lasciare per ultimo l’ESP essendo il Wifi l’ultimo step. Ogni sensore va collegato e poi testato, cancellando il contenuto dello script principale all’interno del setup e del loop (un piccolo consiglio è quello di commentare i due blocchi di codice con i caratteri `/* codice che si vuole commentare */`

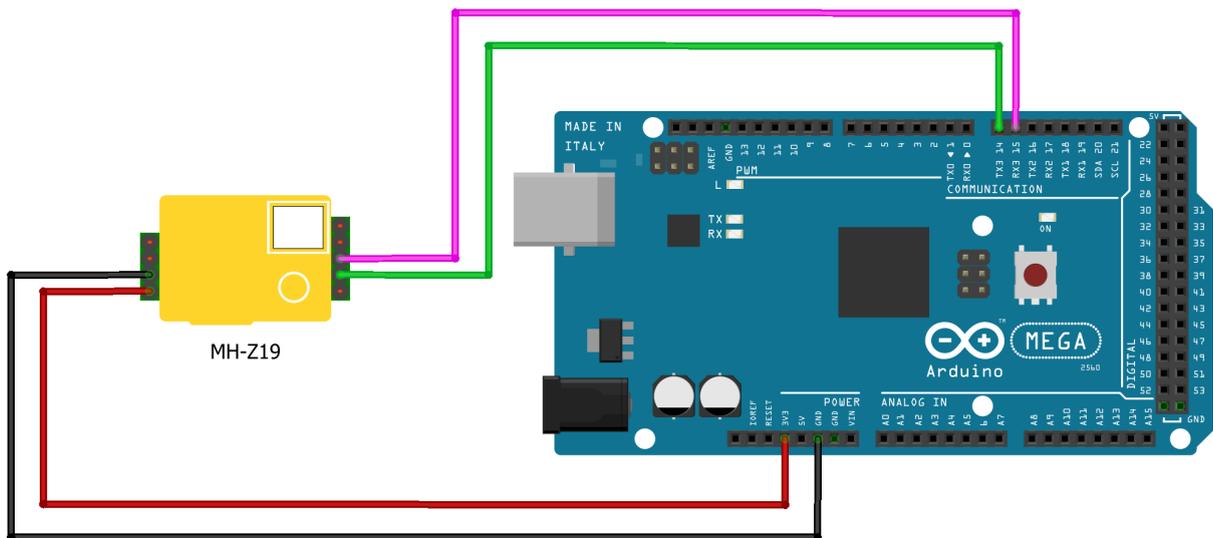


## BMP280 (temperatura e pressione)



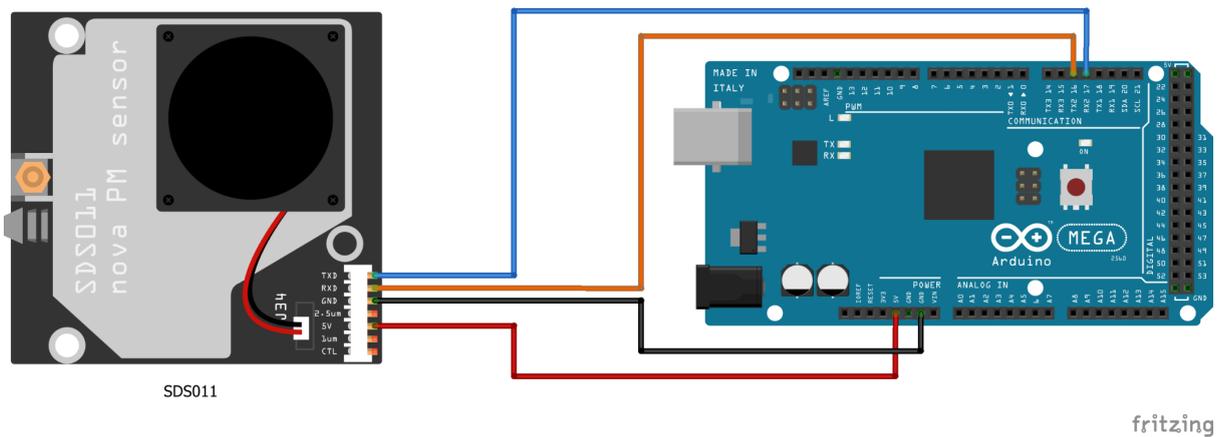
fritzing

## MH-Z19 (concentrazione di CO<sub>2</sub>)



fritzing

## SDS011 (polveri sottili PM2.5 e PM10)



## REALIZZARE IL CODICE

Codice BAM, poi realizzazione codice principale un pezzo alla volta... per esempio, prima includere sensori, provare un sensore in demo, poi includere wifi, lanciare il ping, poi integrarli, e poi realizzazione codice completo

## FUNZIONAMENTO DEL SITO

Questo è il sito, c'è un database, ogni dispositivo è identificato univocamente e trasmette dati per la sua zona, registrazione sul sito e funzionamento delle varie api.  
Integrazione del token di identificazione

## NEXT STEPS

Come installare la centralina (a che altezza, cose varie, menate e ammenicoli), e come farsi figo con le altre scuole e diffondere l'iniziativa