

StartUp

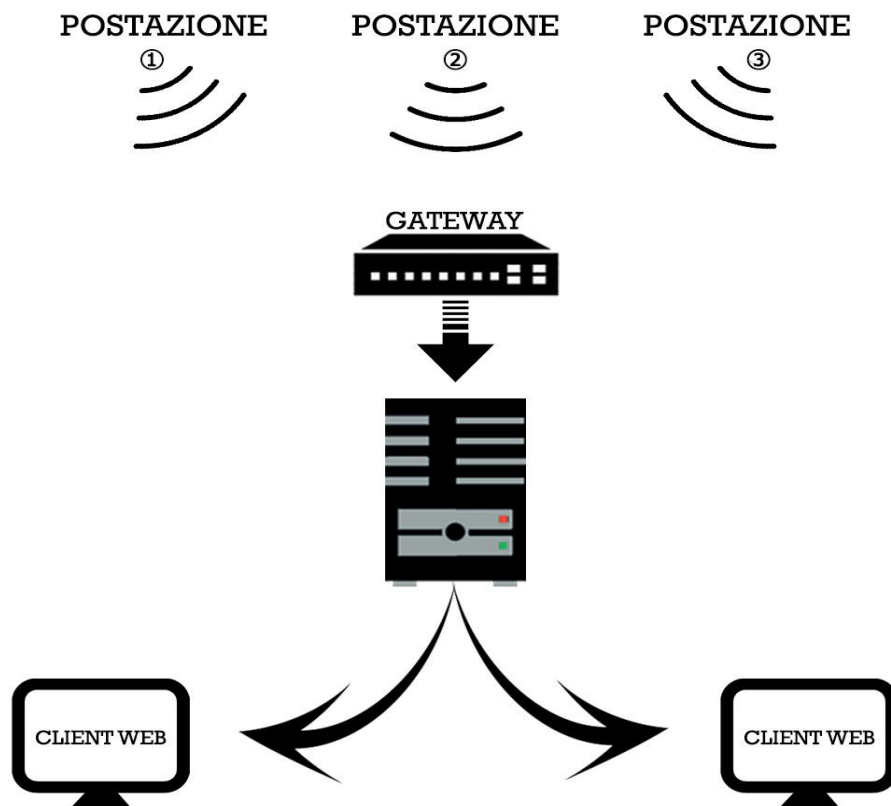
Documentazione progetto



Introduzione

L'istituto superiore Morano ha dato delle direttive per la realizzazione di una StartUp, con l'obiettivo di riuscire a creare un sistema che raccolga dati attraverso tre sensori: umidità dell'aria, umidità del terreno, temperatura dell'aria. L'azienda è formata da due divisioni: la prima divisione è stata incaricata di eseguire la parte di elettronica del progetto, mentre la seconda quella di informatica. Queste due macroattività si dividono a loro volta in microattività, formate da addetti selezionati per compiti precisi. In particolar modo, all'interno della seconda macroattività, sono presenti 3 gruppi di lavoro.

Descrizione del sistema



Il sistema è basato su tre postazioni per l'utilizzo di tre particolari sensori per: temperatura dell'aria, umidità dell'aria e umidità del terreno; questi sensori hanno un gateway d'appoggio che si collega, tramite wi-fi, ad un broker MQTT. Viene effettuato un publish

(per la pubblicazione di informazioni) su un topic appositamente scelto; il formato del pacchetto inviato è: <ID>,<temperatura>,<umidità>,<igrometria>,<dataora>.

Il server centrale riceve i diversi dati dal gateway; nel momento in cui li riceve, li rilancia sul socket sul quale è in ascolto il client, mediante la pagina scaricata durante la prima connessione al server.

Viene utilizzata la programmazione pilotata da eventi, o event-driven (asincrona, l'istruzione non viene bloccata) e, dato che si tratta di un'applicazione non basata su tempi reali, si usufruisce ugualmente di Node.js per l'estrapolazione di informazioni da un'applicazione già esistente, per via di esigenze dovute al tempo a disposizione.

DIPARTIMENTO DI INFORMATICA

La divisione informatica è organizzata in sottogruppi, divisi in modo da poter assicurare la creazione del server node (per il quale è possibile la memorizzazione delle rilevazioni) e la creazione della pagina HTML dove saranno rappresentati i grafici con le relative rilevazioni.

Organigramma

- CORE: gruppo CAVIALE

RESPONSABILE: Vittorio Auriemma;

PARTECIPANTI: Giuseppe Garotti, Alessio Guerra, Alessandro Castelli;

- GRAFICA: gruppo DJICKAR

RESPONSABILE: Emanuele Franzese;

PARTECIPANTI: Giuseppe Andretta, Vincenzo Longobardo, Salvatore Belardo, Angelo Mugione;

- DOCUMENTAZIONE: gruppo DocGrimaldiPedata

RESPONSABILE: Raffaele Grimaldi Capitello;

PARTECIPANTI: Luca Pedata;

Core

Il gruppo CAVIALE, per raggiungere il proprio obiettivo, utilizzerà diversi software, tra cui: Git (software di controllo distribuito, utilizzabile da interfaccia a riga di comando), Node.js (piattaforma event-driven, con moduli base scritti in Javascript, dove gli sviluppatori possono scrivere nuovi moduli in Javascript), MongoDB (DBMS non relazionale, orientato ai documenti, classificatosi come database di tipo NoSQL). Inoltre sarà necessario rendere il sistema compatibile con la comunicazione su protocollo MQTT per poter ricevere aggiornamenti dei dati dal gateway implementato dalla divisione di elettronica.

Grafica

Il gruppo DJICKAR (grafica) si divide a sua volta in due sottogruppi:

- Il primo è composto da Vincenzo Longobardo ed Angelo Mugione, i quali andranno a realizzare, grazie all'ausilio di Bootstrap (framework front-end gratuito per uno sviluppo web più rapido e semplice), la struttura delle pagine nelle quali saranno inseriti i diversi grafici e valori: index.html (inglobare grafici con tutti i valori), report.html (in funzione di un arco temporale), admin.html (si impostano i valori di default);
- il secondo, con i restanti partecipanti, che si occupa, tramite chart.js (libreria di visualizzazione dei dati), dello sviluppo della parte di rappresentazione del grafico.

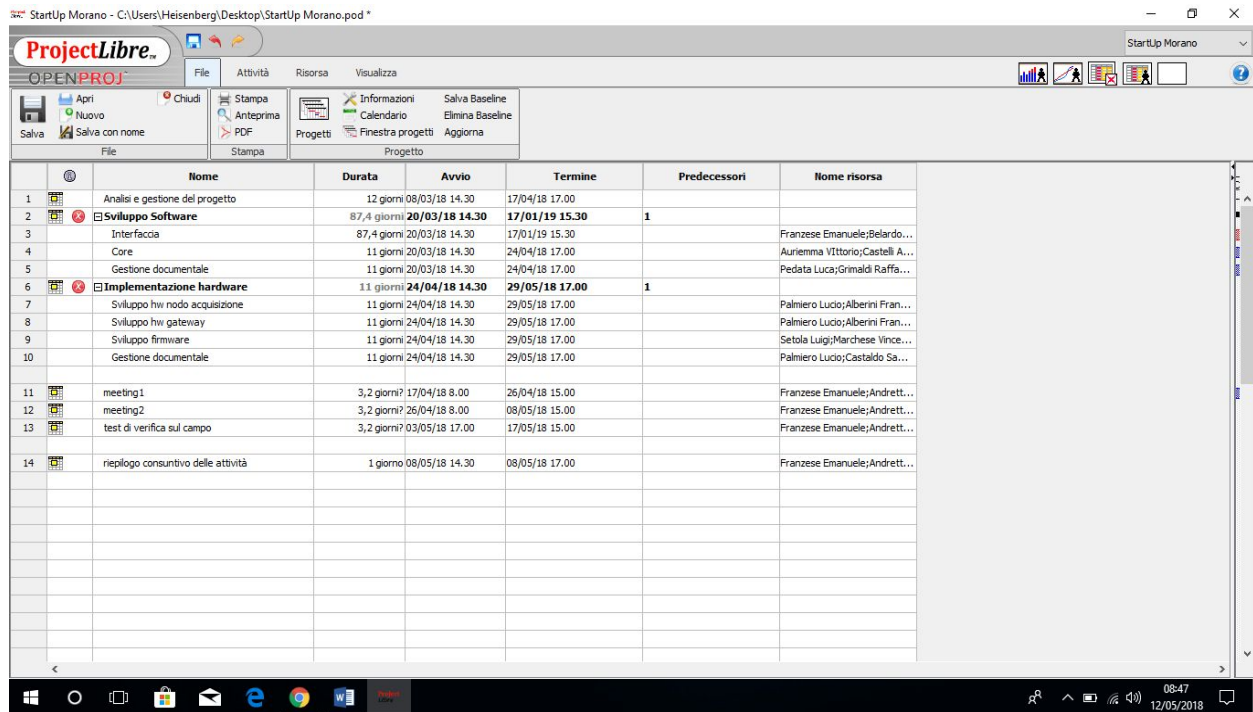
I due gruppi, prima di pubblicare le pagine definitive dovranno interagire in modo da integrare le due realizzazioni.

Documentazione

Il gruppo DocGrimaldiPedata si occupa della descrizione dettagliata di ogni singola fase del progetto. Dunque, è necessario offrire delle spiegazioni, soddisfacenti quanto sintetiche, sul lavoro svolto da ogni membro lavorativo. Inoltre, di fondamentale importanza è il supporto al lavoro dell'equipe in generale, per la quale sarà messo a disposizione un

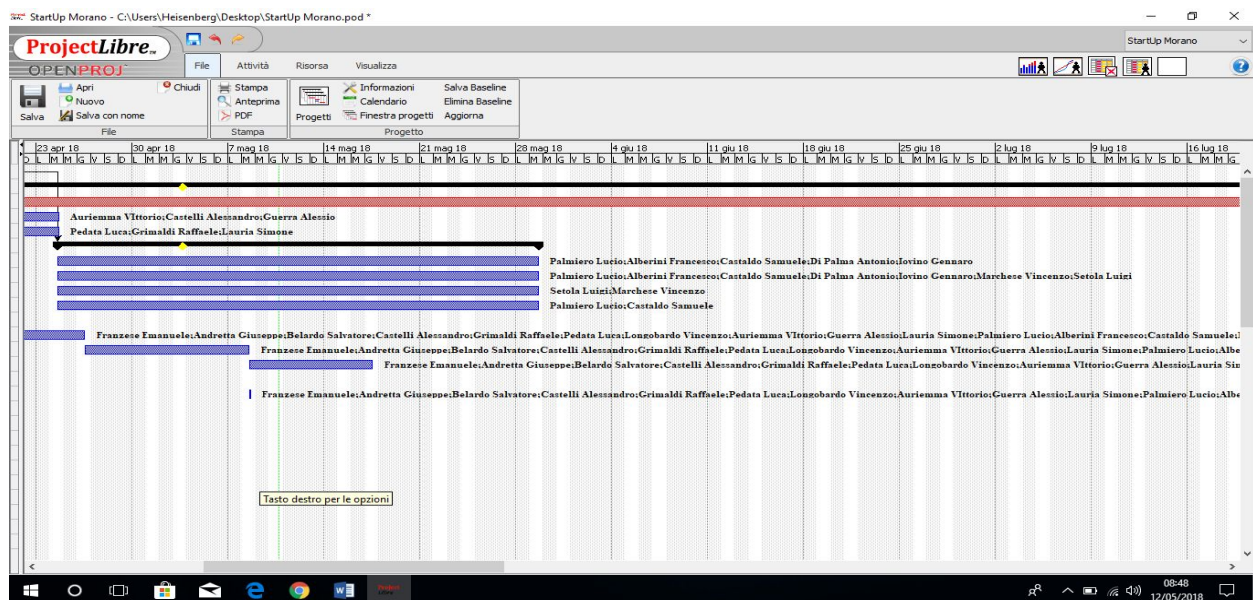
manuale per l'uso di Git, descritto nel modo più minuzioso possibile.

Infine, si avrà a disposizione un diagramma con lo scopo di definire le attività di ciascun gruppo, con i relativi tempi di progettazione (diagramma di Gantt); ciò è possibile grazie ad un ottimo software di supporto: "ProjectLibre".



The screenshot shows the ProjectLibre application window. The main view displays a Gantt chart with tasks listed on the left. The tasks are organized into a hierarchy, with 'Sviluppo Software' and 'Implementazione hardware' being the primary categories. Each task has a duration, start date, end date, and a list of predecessors. The interface includes a menu bar (File, Attività, Risorsa, Visualizza) and a toolbar with various icons for file operations, printing, and project management.

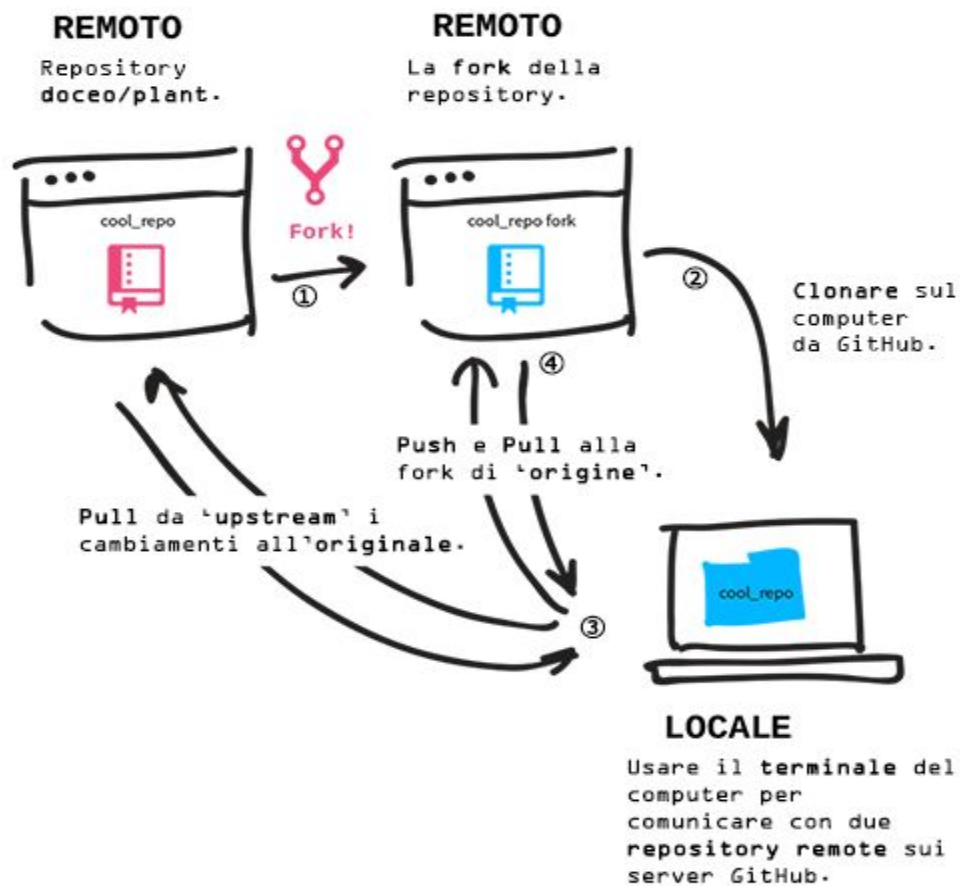
| ID | Nome | Durata | Avvio | Termine | Predecessori | Nome risorsa |
|----|-------------------------------------|-------------|----------------|----------------|--------------|---------------------------------|
| 1 | Analisi e gestione del progetto | 12 giorni | 08/03/18 14.30 | 17/04/18 17.00 | | |
| 2 | Sviluppo Software | 87,4 giorni | 20/03/18 14.30 | 17/01/19 15.30 | 1 | |
| 3 | Interfaccia | 11 giorni | 20/03/18 14.30 | 17/01/19 15.30 | | Franzese Emanuele;Belardo... |
| 4 | Core | 11 giorni | 20/03/18 14.30 | 24/04/18 17.00 | | Auriemma Vittorio;Castelli A... |
| 5 | Gestione documentale | 11 giorni | 20/03/18 14.30 | 24/04/18 17.00 | | Pedata Luca;Grimaldi Raffa... |
| 6 | Implementazione hardware | 11 giorni | 24/04/18 14.30 | 29/05/18 17.00 | 1 | |
| 7 | Sviluppo hw nodo acquisizione | 11 giorni | 24/04/18 14.30 | 29/05/18 17.00 | | Palmiero Lucio;Alberini Fran... |
| 8 | Sviluppo hw gateway | 11 giorni | 24/04/18 14.30 | 29/05/18 17.00 | | Palmiero Lucio;Alberini Fran... |
| 9 | Sviluppo firmware | 11 giorni | 24/04/18 14.30 | 29/05/18 17.00 | | Setola Luigi;Marchese Vince... |
| 10 | Gestione documentale | 11 giorni | 24/04/18 14.30 | 29/05/18 17.00 | | Palmiero Lucio;Castaldo Sa... |
| 11 | meeting1 | 3,2 giorni? | 17/04/18 8.00 | 26/04/18 15.00 | | Franzese Emanuele;Andrett... |
| 12 | meeting2 | 3,2 giorni? | 26/04/18 8.00 | 08/05/18 15.00 | | Franzese Emanuele;Andrett... |
| 13 | test di verifica sul campo | 3,2 giorni? | 03/05/18 17.00 | 17/05/18 15.00 | | Franzese Emanuele;Andrett... |
| 14 | riepilogo consuntivo delle attività | 1 giorno | 08/05/18 14.30 | 08/05/18 17.00 | | Franzese Emanuele;Andrett... |



Metodo di sviluppo

Il software, scelto per essere adattato, è un software già esistente (Plant), pubblicato al repository `github.com/doceo/plant`. La scelta della tecnologia e del software è stata decisa in base al tempo limitato da poter destinare allo sviluppo, si precisa quindi che le funzionalità saranno limitate allo stretto necessario per realizzare l'acquisizione dei dati e alla loro rappresentazione grafica.

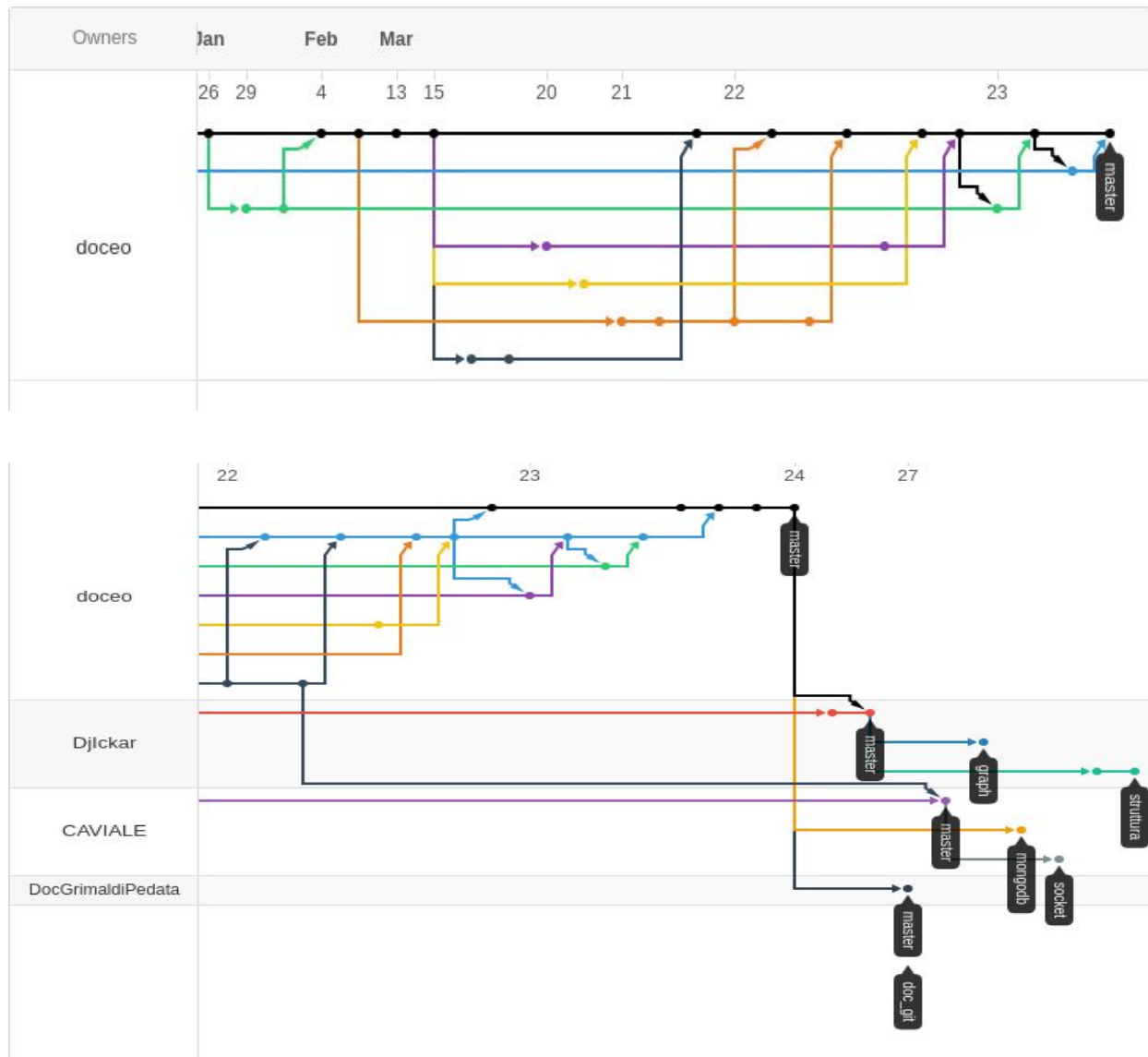
I gruppi di lavoro hanno realizzato una fork del progetto iniziale e, coordinati attraverso github, hanno provveduto a realizzare le varie features creando di volta in volta branch diversi a partire dalle issue segnalate sul branch master dell'account `github.com/doceo`. Le modalità di sviluppo hanno quindi seguito lo schema di seguito riportato:



In allegato il manuale d'uso redatto dal gruppo della documentazione e messo a disposizione degli sviluppatori.

https://github.com/doceo/plant/blob/master/doc/Git_metodo_di_lavoro.pdf

Di seguito sono riportati i vari branch creati partendo dal branch master:



Dipartimento Elettronica

La divisione elettronica è organizzata in sottogruppi, divisi in modo da poter garantire, oltre alla realizzazione delle postazioni di rilevazione, anche la gestione della conversione dei dati da flussi analogici che viaggiano su onde radio a pacchetti TCP su protocollo TCP/IP. Verrà, inoltre, utilizzato il protocollo di comunicazione MQTT, interfacciandosi al broker Mosquitto, che nasce proprio per scambio di dati IOT.

Piedinatura RF



CE: L'input che, quando è vero, consente l'operazione attiva incluso l'input e / o l'output dei dati e, se falso, impedisce il funzionamento attivo e fa sì che la memoria si trovi in modalità standby ridotta con le uscite mobili.

CSN: è il nome di una linea di controllo nell'elettronica digitale utilizzata per selezionare uno (o un insieme) di circuiti integrati (comunemente chiamati "chip") tra più collegati allo stesso bus del computer, solitamente utilizzando la logica a tre stati.

CSK: L'orologio pulsa sincronizzando la trasmissione dei dati generata dal master.

MOSI: La linea Slave per l'invio di dati al master.

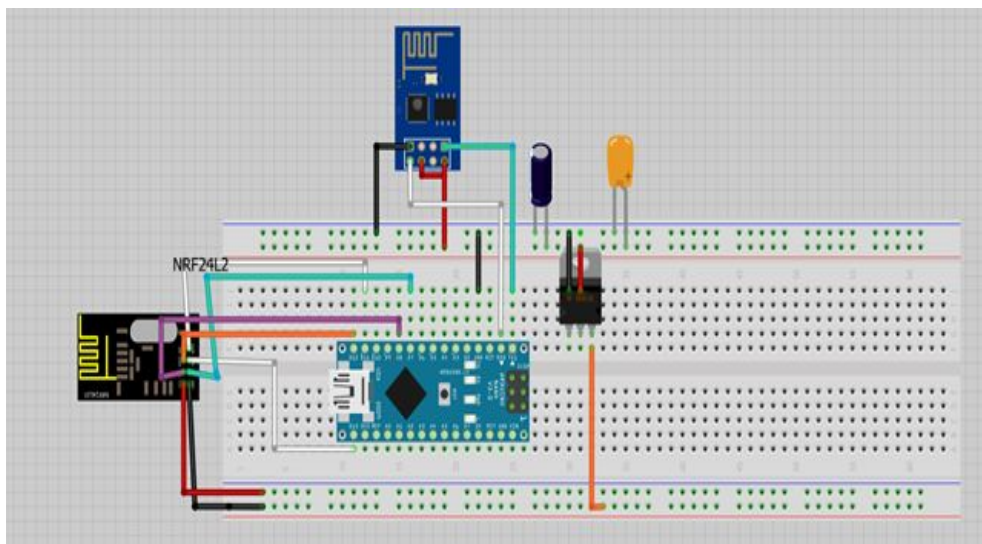
MISO: La linea Master per l'invio dei dati alle periferiche.

IRQ: un IRQ è una *richiesta* da parte dell'hardware collegato alla CPU, eseguita mandando opportuni segnali ai pin fisici di collegamento della CPU stessa. Tale richiesta, a seconda dello stato in cui si trova la CPU, può anche essere accantonata per essere eseguita in un secondo tempo. Gli IRQ sono spesso organizzati in gerarchie di priorità, dove un IRQ di basso livello può essere a sua volta interrotto da un IRQ di livello più alto. I diversi tipi di computer in genere standardizzano i loro IRQ in modo che ad un certo IRQ corrisponda sempre un certo tipo di evento esterno: un tasto premuto sulla tastiera, una comunicazione in arrivo dalla porta seriale e così via.

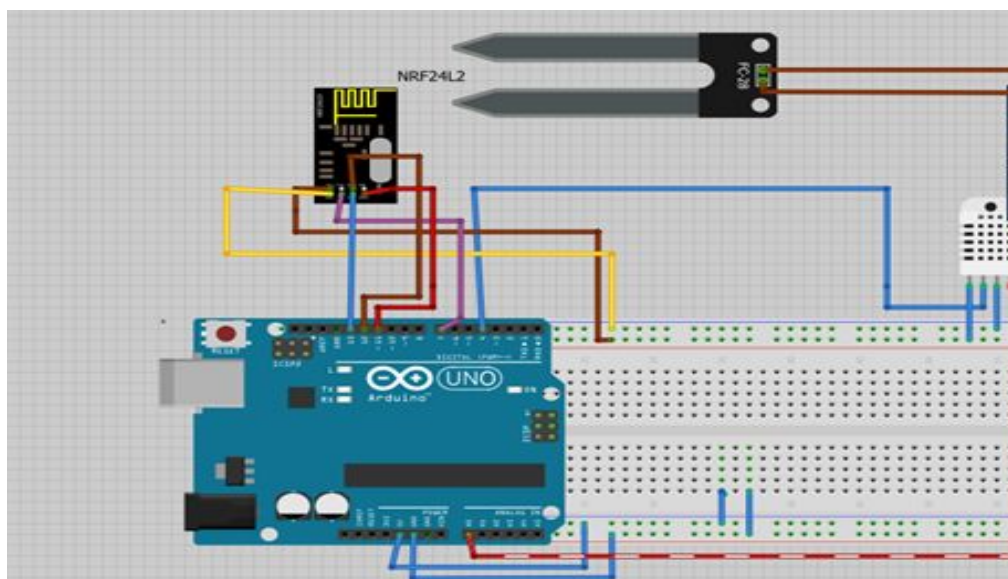
SPI: SPI è un'interfaccia single-master – multi-slave, seriale, sincrona, full duplex. Si noti che questo tipo di comunicazione è adatto a circuiti posti sulla medesima scheda o in stretta vicinanza e non è adatto alla comunicazione a distanza.

| | RF | arduino |
|------|----|---------|
| CE | 3 | 7 |
| MISO | 7 | 11 |
| MOSI | 6 | 12 |
| SCK | 5 | 13 |
| GND | 1 | 3.3V |
| VCC | 2 | |
| CSN | 4 | 8 |

SCHEMA TECNICO DEL GATEWAY



SCHEMA TECNICO DEL NODO



Organigramma divisione Elettronica

- SVILUPPO HW NODO ACQUISIZIONE

RESPONSABILE: Francesco Alberini;

PARTECIPANTI: Lucio Palmiero;

- SVILUPPO HW GATEWAY

RESPONSABILE: Samuele Castaldo;

PARTECIPANTI: Lucio Palmiero, Francesco Alberini, Antonio Di Palma, Gennaro Iovine;

- SVILUPPO FIRMWARE

RESPONSABILE: Luigi Setola;

PARTECIPANTI: Vincenzo Marchese;

- GESTIONE DOCUMENTALE

RESPONSABILE: Lucio Palmiero;

PARTECIPANTI: Samuele Castaldo;

Riferimenti web

www.GitHub.com

FONTE:

<http://www.mcmajan.com/mcmajanwpr/blog/2014/06/12/arduino-comunicazione-wireless-2-4ghz-con-nrf24l01-e-comunicazione-spi-it/>

Dht 11:

Link: <https://www.progettiarduino.com/11-temperatura-e-umidita-grave-dht-11.html>

Link dht 11 library: <https://github.com/adidax/dht11>

Nrf24 library: <http://www.airspayce.com/mikem/arduino/RadioHead/>