

Corso di Laurea Triennale in Ingegneria e Scienze Informatiche

# Prototipo di irrigazione prescrittivo

Tesi di laurea in:  
SUPERVISOR'S COURSE NAME

*Relatore*

**Prof. Supervisor Here**

*Candidato*

**Davide Speciali**

*Correlatori*

**Dott. CoSupervisor 1**

**Dott. CoSupervisor 2**

---

---

# Abstract

TODO

---

---

*Optional. Max a few lines.*

---

---

# Indice

<b>Abstract</b>	<b>iii</b>
<b>Introduzione</b>	<b>1</b>
<b>1 Precision Watering</b>	<b>3</b>
1.1 Necessità e obiettivi . . . . .	4
1.2 Approcci . . . . .	4
1.3 Limiti incontrati . . . . .	4
<b>2 Tecnologie utilizzate</b>	<b>5</b>
2.1 Hardware . . . . .	5
2.2 Software . . . . .	6
<b>3 Un prototipo di irrigazione prescrittivo</b>	<b>11</b>
<b>4 Conclusioni e sviluppi futuri</b>	<b>13</b>
	<b>15</b>
<b>Bibliografia</b>	<b>15</b>





---

# Introduzione

## Structure of the Thesis

**Davide Speziali:** At the end, describe the structure of the paper



---

# Capitolo 1

## Precision Watering

La risposta al cambiamento climatico sta diventando uno dei temi più importanti per il futuro delle popolazioni del pianeta (una bella fontazza cringe qua ci sta). Per questo motivo a partire dal 2015 le United Nations (UN) hanno adottato l'agenda per lo sviluppo sostenibile 2030 (capire come è corretto citarlo) e i suoi 17 Sustainable Development Goals (SDG). Tra i 17 obiettivi compare anche SDG 6 "Acqua pulita e sanificazione", l'obiettivo è quello di garantire un accesso sicuro all'acqua a tutta la popolazione del mondo [UN15] . Il target 6.4, in particolare punta ad un miglioramento dell'efficienza del l'uso di acqua in tutti i settori [UN-21a, UN-21c] , tra cui soprattutto quello agricolo che utilizza circa il 75% dell'acqua dolce ottenibile in natura [Wal00]. Per questo motivo si è sviluppato l'interesse per lo sviluppo di tecnologie di monitoraggio dei terreni e lo sviluppo di modelli per l'irrigazione efficiente ed automatica. (cerca citazione di un aumento del precisio farming)

La sezione 1.1 illustra qual è l'impatto climatico dell'agricoltura per quel che concerne lo sfruttamento di acqua, la sezione 1.2 elenca quali possono essere i modelli di simulazione del terreno e la loro importanza e infine la sezione 1.3 ragiona su quali sono i limiti degli approcci precedentemente elencati.

## 1.1 Necessità e obiettivi

La quantità di acqua usata per l'agricoltura quasi raddoppiata dal 1970 ed oggi l'agricoltura rimane l'area dove l'utilizzo di acqua è il maggiore, circa tre volte rispetto quello delle altre industrie. [FAO20] L'utilizzo eccessivo di acqua ha portato ad un aumento della siccità(fonte), sono presenti diversi studi che dimostrano una diminuzione del 30% del raccolto di frutta e verdura negli stati uniti a causa della siccità mentre in altri casi effetti combinati di siccità e aumento di temperature ha portato ad una diminuzione sia della dimensione del raccolto che della quantità di fotosintesi per la pianta di grano [CMDAMM19].

In alcuni casi però, secondo i parametri delle UN, sono state impletate delle soluzioni funzionanti che hanno permesso un drastico miglioramento dell'efficienza nell'uso dell'acqua nell'area agricola, un caso di esempio importante riguarda la Giordania dove la scelta di culture più resistenti alla scarsità di acqua è stato affiancata all'utilizzo di precision fertigation, ossia l'utilizzo del numero ottimo di fertilizzanti e pesticidi insieme con l'acqua.

Spiegazione precision watering

Attualmente sono tanti i paesi che stanno impiegando le proprie risorse alla ricerca di modi più efficienti per l'utilizzo dell'acqua, in particolar modo Lussemburgo e Svizzera. [UN-21b]

## 1.2 Approcci

Sia csm che capitolo 2

## 1.3 Limiti incontrati

Non so se è separabile dagli approcci

---

# Capitolo 2

## Tecnologie utilizzate

### 2.1 Hardware

- **Sensori**, TODO numero sensore e citazione alle specifiche
- **Arduino UNO R3** è una scheda microcontrollore basata sull'ATmega328P. Permette di caricare i programmi scritti in linguaggio C++ usando un'architettura SuperLoop, dove un unico pezzo di codice viene ripetuto fin tanto che il dispositivo è riceve alimentazione. Il superloop è interrompibile momentaneamente solo tramite degli interrupt di sistema, che vengono gestiti con routine non interrompibili, ciò implica che durante una routine, non possono avvenire altri interrupt. È quindi consigliabile che le routine di gestione siano particolarmente brevi, sia per evitare perdita di eventi, sia perchè molte librerie di sistema sfruttano internamente gli interrupt, per esempio `delay` (equivalentemente a `sleep` di python). Offre l'utilizzo di un totale di 22 pin General Purpose Input/Output (GPIO): 16 dei quali digitali, di cui 6 utilizzabili come output analogici tramite pulse-width modulation (PWM), e altri 6 pin input analogico. Può inoltre gestire una comunicazione seriale tramite USB con la quale è possibile comunicare con altri dispositivi.
- **Raspberry Pi 4 model B** è un Single Board Computer (SBC) in grado di utilizzare sistemi operativi veri e propri basati sul kernel Linux o su RISC

OS. Presenta un processore quad-core ARM a 64bit ad una velocità di clock di 1.80GHz. La configurazione utilizzata presenta

**Davide Speziali: QUANTITARAM**

Gb di RAM LPDDR4-3200. Compatibile con Bluetooth 5.0, BLE, connessioni wireless IEEE 802.11ac a 2.4 GHz e 5.0 GHz, oltre che Gigabit Ethernet. Permette anche l'utilizzo di 40 pin GPIO digitali. Compatibile con periferiche USB tramite 2 porte USB2.0 e 2 porte USB3.0. L'uscita video è permessa da tramite 2 porte Micro-HDMI, che supportano dual-screen 4k. Come storage viene utilizzata una singola scheda microSD. Include un'interfaccia MIPI DSI per display e MIPI CSI per telecamere, ed è alimentato tramite un connettore USB-C da 5V/3A. Prodotto dalla Raspberry Pi Foundation, che ha realizzato diversi modelli per varie esigenze, è compatibile con vari sistemi operativi, tra cui il più noto è Raspberry Pi OS (precedentemente chiamato Raspbian), una versione di Debian sviluppata dalla stessa fondazione.

## 2.2 Software

- **JavaScript (JS)** è un linguaggio di programmazione interpretato, usato principalmente per la programmazione web sia a livello client che server, per esempio con l'utilizzo di Node.js. È un linguaggio dinamico, basato sui prototipi, multi-paradigma, single-threaded, che supporta stili di programmazione orientati agli oggetti, imperativi e dichiarativi. Standardizzato come ECMAScript, JS è alla base di un vasto ecosistema di librerie e framework che ne ampliano le capacità e l'applicabilità in diversi contesti.
  - **Chart.js** è una libreria JS semplice e flessibile per la generazione di grafici lato frontend. Si tratta di un progetto Open Source che sfrutta i canvas di HTML5. Di default permette l'utilizzo di 8 diversi tipi di grafici, è compatibile con plug-in che permettono l'aggiunta di nuove funzionalità.
    - \* **chartjs-chart-matrix** è un plug-in che permette la creazione di grafici a matrice, un esempio è in fig. 2.1

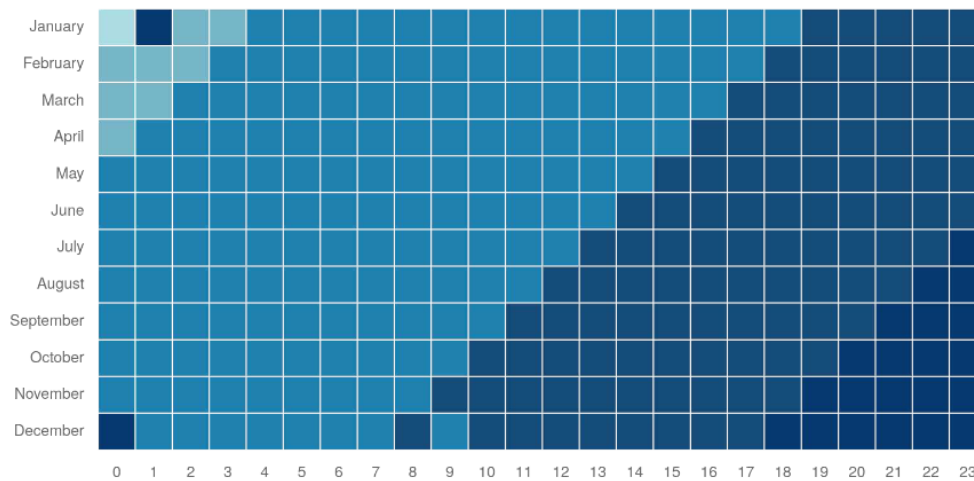


Figura 2.1: Esempio di una mappa delle temperature medie giornaliere per mese, l'intensità del colore è proporzionale alla temperatura.

- \* **chartjs-plugin-streaming** è un plug-in che permette la creazione di grafici con valori in tempo reale. Prevede lo scorrimento verso sinistra del grafico in base al tempo. È possibile configurarlo per modificare la granularità, la frequenza di aggiornamento ed un eventuale delay in modo da avere una visualizzazione "fluida" dei nuovi dati. Necessita di un adapter per una libreria per la gestione delle date e ricezione del tempo attuale, come Luxon.
- \* **chartjs-plugin-datalabels** è un plug-in per la visualizzazione dei dati su un grafico, un esempio su un grafico a linea è visibile in fig. 2.2
- \* **chartjs-adapter-luxon** è un plug-in che si occupa di fornire a chartjs-plugin-streaming le informazioni sulle date da Luxon.
- **Luxon** è un wrapper di date e tempi per JS. Utilizza un'API per gestire le date, che consente il supporto agli intervalli, alle durate (14 days, 5 minutes, 33 seconds), la loro conversione e parsing. Offre la localizzazione delle stringhe con una gestione interna delle time-zones.
- **bootstrap** è un toolkit frontend potente ed estendibile. Permette la creazione di interfacce web utilizzando unicamente classi HTML, senza la necessità di scrivere il proprio foglio di stile CSS.

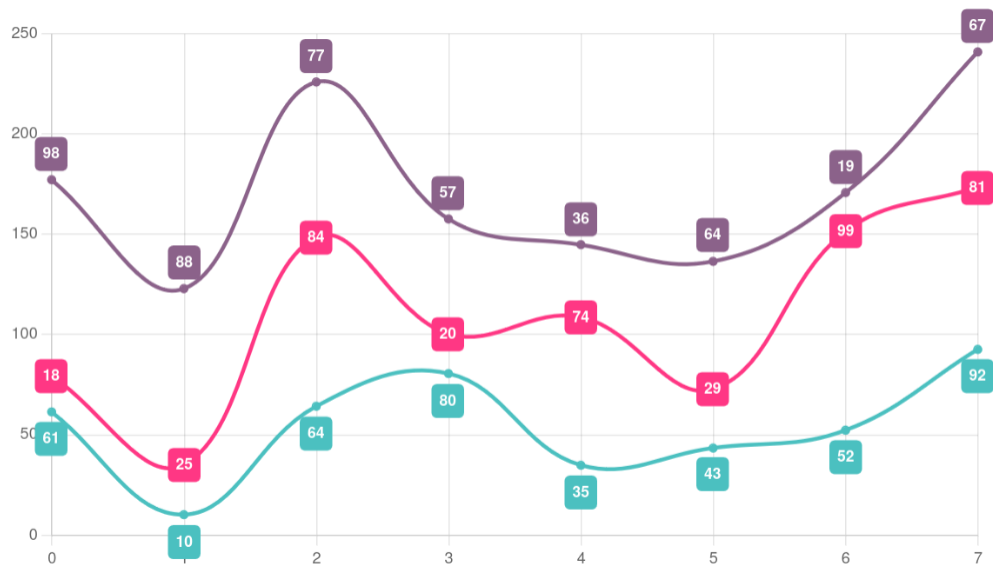


Figura 2.2: Esempio utilizzo di datalabels dove in ogni linea la posizione del label è differente.

- **jQuery** è una famosa libreria per la manipolazione e navigazione di documenti HTML, la gestione degli eventi, le animazioni e richieste HTTP.
- **Socket.io** permette l’apertura di una comunicazione bidirezionale sempre attiva tra client e server utilizzando WebSocket.
- **Python** è un linguaggio di programmazione ad alto livello che supporta il paradigma object-oriented, la programmazione strutturata e molte caratteristiche di programmazione funzionale. È un linguaggio debolmente tipizzato, che utilizza unicamente l’indentazione per la definizione dei blocchi di codice, al contrario di molti linguaggi, non prevede quindi l’utilizzo di parentesi graffe. Supporta l’overloading di operatori e funzioni tramite delegati, oltre che sintassi avanzate quali slicing (l’utilizzo di un certo range di elementi in una lista) e list comprehension (la creazione di nuove liste partendo da una originale). Il codice python viene interpretato (con delle compilazioni a bytecode alla prima esecuzione per migliorare le performance) che supporta anche un uso interattivo.



- **Flask** è un framework web per la creazione di backend per Python.
- **Flask-SocketIO** è una libreria Python per la gestione di websocket tramite Socket.io.
- **NumPy** è la libreria principale per il calcolo scientifico Python. Permette la gestione di array  $n$ -dimensionali ed è ampiamente utilizzato per operazioni di algebra lineare, trasformate di Fourier, manipolazioni di dati e altre applicazioni numeriche. È uno strumento essenziale per chi lavora con analisi numeriche, machine learning e data science.
- **pySerial** è una libreria per l'accesso a porte seriali. Permette la stessa gestione delle porte seriali su ogni piattaforma. Supporta diversi byte size, bit di parità e di fine messaggio, oltre che controllo del flusso. La porta è configurata per la trasmissione binaria.
- **dotenv** è un modulo per la gestione delle variabili d'ambiente.
- **pytz** è una libreria per il calcolo del fuso orario in python.
- **SciPy** è una libreria di Python per il calcolo scientifico e tecnico, espande NumPy. Offre un'ampia quantità di funzionalità per l'ottimizzazione, la statistica, l'integrazione, l'algebra lineare, l'elaborazione dei segnali, la gestione di immagini e altre operazioni matematiche avanzate. È molto utilizzata per esempio nei campi della fisica, dell'ingegneria, della biologia computazionale.
- **C++** è un linguaggio di programmazione nato come espansione del linguaggio C, presenta caratteristiche di programmazione funzionale, generica e orientata agli oggetti, permettendo però una gestione a basso livello della memoria. Per quest'ultima caratteristica è tipicamente utilizzato in contesti dove le prestazioni sono particolarmente importanti come lo sviluppo di sistemi operativi, software embedded, motori di gioco e applicazioni real-time. In particolar modo è utilizzato anche per lo sviluppo di software per dispositivi con risorse limitate, come microcontrollori e sistemi embedded, dove la gestione diretta della memoria e delle risorse hardware è essenziale.
  - **Arduino API** è una libreria in C/C++ per la scrittura di codice per dispositivi Arduino. Contiene funzioni riguardanti l'hardware, per esem-

pio, contiene funzioni per la lettura/scrittura di segnali digitali e analogici sui pin, per la gestione degli interrupt e della porta seriale. Oltre alle funzioni hardware, l'API include, ad esempio, strumenti per la gestione dei timer, funzioni matematiche, funzioni per la gestione di bit e byte e per la gestione degli stream. Contiene inoltre i tipi di variabile 'enum', 'String' e funzioni per la conversione.

- **TimerInterrupt** è una libreria che permette l'utilizzo del timer fisico di una board Arduino come interrupt di sistema, permettendo lo svolgimento ininterrotto di routine periodicamente.
- **ArduinoJson** è una semplice libreria per la gestione di stringhe JSON su piattaforma Arduino. Permette serializzazione e deserializzazione, ed è quindi molto utile per mandare dati strutturati sul seriale in modo che venga che lo scambio dei dati sia standardizzato.

---

## Capitolo 3

# Un prototipo di irrigazione prescrittivo

Descrive nel dettaglio il tuo lavoro, quindi come le tecnologie presentate nella precedente sezione sono state utilizzate nel progetto. Puoi prenderla larga descrivendo il nostro sistema (quello che ti ho mostrato sul mio pc) per poi descrivere la necessità di averne una rappresentazione su piccola scala e come è stata realizzata, così come descrivere il ciclo di vita del dato (collection, processing, exploitation). Ora come ora è normale tu non abbia chiare le idee su questo capitolo, andando avanti integreremo le tue conoscenze con quelle pregresse nostre sul dominio applicativo e sul nostro sistema. Questo capitolo deve essere il core della tesi, lunghezza 20 pagine.

---

---

## Capitolo 4

# Conclusioni e sviluppi futuri

Breve capitolo che trae le conclusioni sul lavoro svolto, il risultato ottenuto rispetto a quello atteso e lo spazio dedicato a migliorie future.

---

---

# Bibliografia

- [CMDAMM19] Alessia Cogato, Franco Meggio, Massimiliano De Antoni Migliorati, and Francesco Marinello. Extreme weather events in agriculture: A systematic review. *Sustainability*, 11(9), 2019.
- [FAO20] FAO. Aquastat dissemination system. <https://data.apps.fao.org/aquastat/>, 2020. [Accessed 22-08-2024].
- [UN-21a] UN-Water. Indicator 6.4.1 “change in water use efficiency over time”. <https://www.unwater.org/our-work/integrated-monitoring-initiative-sdg-6/indicator-641-change-water-use-efficiency-over-time>, 2021. [Accessed 22-08-2024].
- [UN-21b] UN-Water. Progress on water-use efficiency (sdg target 6.4). <https://sdg6data.org/en/indicator/6.4.1>, 2021. [Accessed 22-08-2024].
- [UN-21c] UN-Water. UN-Water analytical brief - Water-use efficiency, 2021. UN-Water publication.
- [UN15] UN. Goal 6: Ensure access to water and sanitation for all. <https://www.un.org/sustainabledevelopment/water-and-sanitation/>, 2015. [Accessed 22-08-2024].
- [Wal00] J.S. Wallace. Increasing agricultural water use efficiency to meet future food production. *Agriculture, Ecosystems & Environment*, 82(1):105–119, 2000.





---

# Ringraziamenti

Optional. Max 1 page.