
ESAME DI STATO 2020.2021

ELABORATO DELLE DISCIPLINE D'INDIRIZZO

(individuate come oggetto della seconda prova scritta – Articolo 18, O. M. n. 52)

COGNOME E NOME DEL CANDIDATO

Stangherlin Enrico

Classe

5AI

ARGOMENTO DELL'ELABORATO

Applicazione con droni

TRATTAZIONE

1. INTRODUZIONE.....	3
2. OBIETTIVI	4
3. DESCRIZIONE DI DETTAGLIO	4
3.1 DRONI	5
3.1.1 Connessione	5
3.1.2 Categorizzazione.....	5
3.1.2.1 Numero di eliche – A rotore	6
3.1.2.2 Drone ad ali fisse.....	6
3.1.3 Quadcopter.....	6
3.2 PROGRAMMAZIONE	7
3.2.1 Python	7
Utilizzo nel progetto.....	7
Test di volo.....	8
tello_test.py – Main.....	8
tello.py – stats.py	8
Videocamera e movimento da tastiera	9
3.2.2 C++.....	10
3.2.2.1 CMake	10
3.2.2.2 Utilizzo nel progetto.....	10
Codifica H.264	11
3.2.3 MySQL	12
Utilizzo nel progetto.....	12
Progettazione Concettuale - Modello Entità/Relazione	12
Progettazione Logica – Ristrutturazione E/R.....	12
3.2.4 PHP.....	13
3.2.4.1 HTML e CSS	13
3.2.4.2 Utilizzo nel progetto.....	14
Login	14
Upload.....	15
Caratteristiche comuni delle pagine	16
4. SVILUPPI FUTURI.....	16
5. CONCLUSIONI.....	18
6. FONTI.....	19

1. Introduzione

Al giorno d'oggi siamo circondati dalla tecnologia: dagli assistenti vocali alle macchine in grado di guidare in maniera autonoma, dalle intelligenze artificiali anti-frode a quelle utilizzate per programmare i videogiochi. Ogni giorno il mondo continua a progredire con le tecnologie e non solo in ambito informatico, ma anche in quello meccanico o d'automazione. Ma è con l'unione di tutte queste discipline che la tecnologia progredisce davvero e aiuta le persone a vivere meglio nella loro quotidianità.

Videocamere di sorveglianza, utili per mantenere l'ordine. Le possiamo trovare nei negozi e nelle banche, ma non solo. Sarà capitato a tutti di camminare per le strade di qualche grande città e vedere l'occhio del grande fratello osservarci dall'alto. Ogni movimento è *registrato*; ogni passo che facciamo viene salvato, almeno *finché siamo nel raggio d'azione*. Certo, al giorno d'oggi la tecnologia ci permette di avere videocamere con angolo di visione di 360° e una capacità di movimento multidirezionale, ma restano pur sempre **oggetti statici nello spazio**.

Con questo progetto si vuole introdurre un'implementazione ai sistemi di sorveglianza. Un dispositivo in grado di vedere dall'alto, ma stando al passo con le persone: i **droni**.

Introdotti dai tedeschi a scopo militare nella seconda guerra mondiale (1943 – "*Fritz X*"), sono oggi molto utilizzati anche da giovani e appassionati, seppur in maniera più ridotta. Ma non sono solo utilizzati nel mondo militare o dello svago, anzi dal 2014 iniziarono a rivoluzionare le forze dell'ordine, che cominciarono ad utilizzarli sempre più, fino ad oggi.

La loro utilità sta nel fatto che possono fornire **viste senza precedenti** di una scena o un incidente, aiutano a combattere il crimine tenendo al sicuro gli agenti con l'inseguimento di un criminale o l'ispezione di un edificio, o ancora possono perlustrare le strade.

2. Obiettivi

Il progetto riguarda quindi un drone, che incorpora in sé sia l'ambito **informatico** sia quello **d'automazione**. L'obiettivo *teorico* è di poter far muovere il drone in maniera autonoma, seguendo un percorso tra due punti evitando gli ostacoli. Il percorso dovrebbe poi essere salvato in un server, insieme con il video relativo ad esso. Questo andrebbe a simulare dei "droni cittadini", il quale compito è quello di girare per le strade, seguendo percorsi specifici e registrando audio e video per garantire la sicurezza. Per mancanza di reperibilità dei materiali la prima parte verrà analizzata solo in maniera teorica, quindi l'unico punto che verrà propriamente sviluppato sarà il secondo, tramite il quale si andrà a creare una sorta di "Scatola Nera" del drone per poi salvare tutti i video effettuati in una cartella locale.

3. Descrizione di dettaglio

In quanto segue non si andrà a spiegare come funziona la parte meccanica del drone, ma verrà solo fatta un'introduzione sulle diverse tipologie che sono presenti per poi passare al lato di programmazione che comprende quella del drone e della pagina web.

Nell'immagine sottostante (vedi **Figura 1**) è presente *l'infrastruttura* di quanto si andrà a creare. I punti **A** (vedi **3.1.1 Connessione**) e **B** (vedi **Upload**) sono spiegati successivamente. Per quanto riguarda il drone si andrà a utilizzare *Tello EDU*: la maggior parte del progetto si riferisce a questo modello piuttosto che altri.

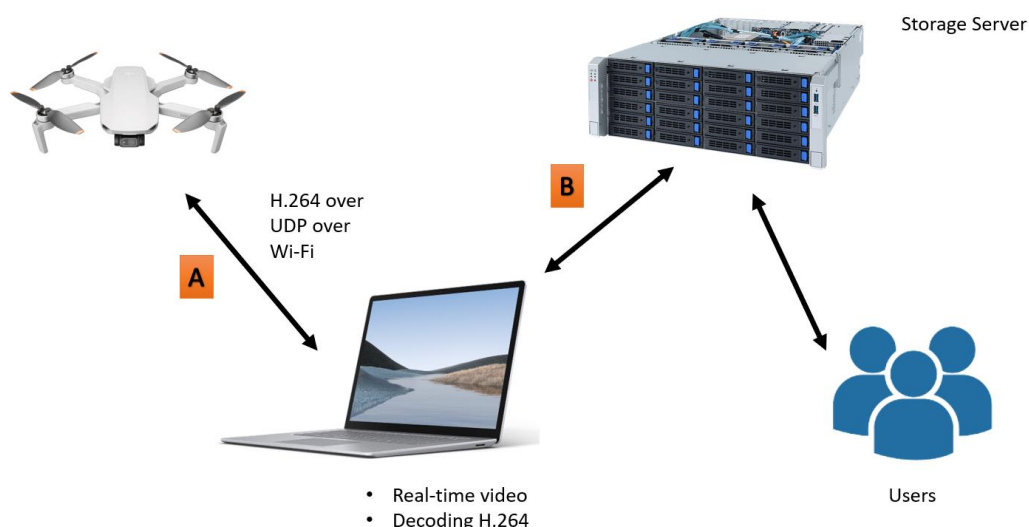


Figura 1

3.1 Droni

Innanzitutto, prima di parlare di droni, bisogna capire come sono fatti, come funzionano e come utilizzarli.

3.1.1 Connessione

Ci sono vari tipi di collegamento tra drone e altro dispositivo, ma quello che viene analizzato in questo elaborato è il collegamento tramite una porta Wi-Fi UDP al Tello EDU.

Si premette che il drone Tello EDU ha indirizzo IP 192.168.10.1 e la porta UDP è la 8889.

Per collegarsi bisognerà innanzitutto impostare un *Client UDP* sul computer o mobile per poter inviare, e quindi ricevere risposta, dalla stessa porta del drone. Basterà questo per essere collegati, ma se si vogliono inviare comandi al drone bisognerà trasmettere il comando "command", per inizializzare la modalità SDK (vedi **Figura 2**).

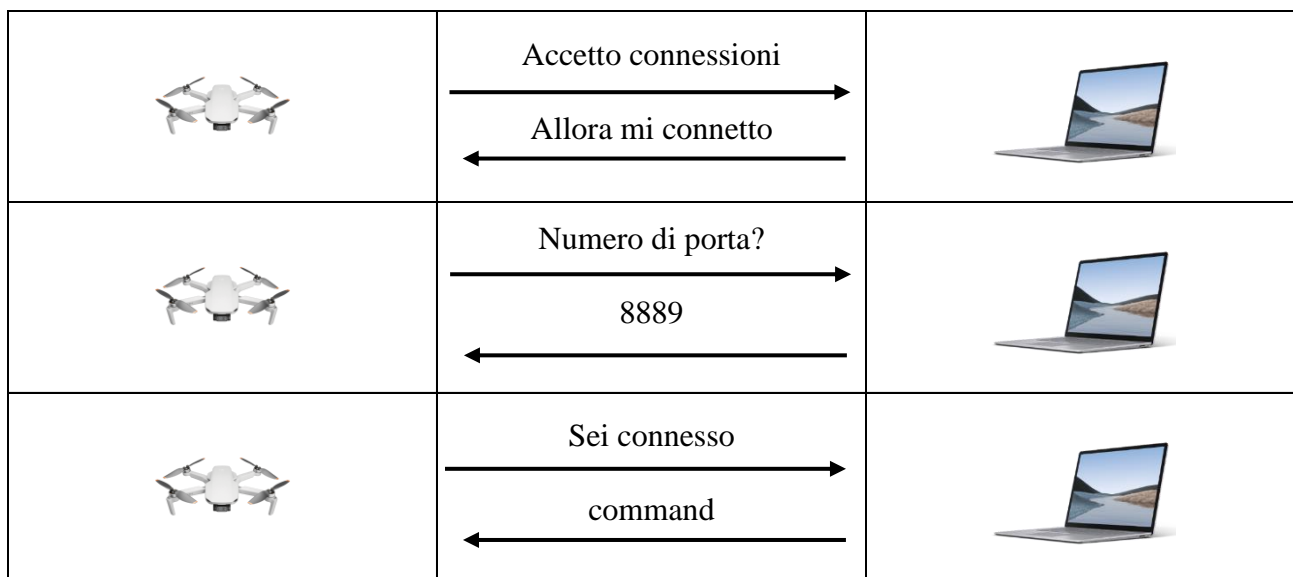


Figura 2

3.1.2 Categorizzazione

I droni principalmente si caratterizzano tra loro per:

- numero di eliche;
- se a rotore o ad ala fissa.

3.1.2.1 Numero di eliche – A rotore

All'interno di questa categoria esiste a sua volta una suddivisione in droni a multi-rotore o rotore singolo.

Per quanto riguarda il rotore singolo si ha solo un modello, costituito da un rotore responsabile dell'elica incaricata del decollo. È definito a rotore singolo, ma in realtà ne possiede un altro situato alla coda che funge da "timone" per il direzionamento nello spazio orizzontale. Un esempio di drone a singolo rotore, anche se molto più in grande, sono gli elicotteri.

Per quanto riguarda i droni a multi-rotore, ognuno di essi varia in base al *numero di eliche* (3, 4, 6, 8) ma, in quanto per il progetto è stato utilizzato un drone Quadcopter, verrà approfondito solo lo stesso.

3.1.2.2 Drone ad ali fisse

Questa è una categoria completamente diversa dalle altre. Ci sono design piuttosto unici rispetto ai tipi di droni a multi-rotore usualmente utilizzati. Su di questi si trova un'unica ala e sembrano degli aerei tradizionali. Questi droni **non sono in grado di stare stabili nell'aria** in quanto non sono molto potenti per combattere la forza gravitazionale. Essi trovano le loro applicazioni nella *registrazione dei movimenti*, dove possono andare avanti anche grazie al loro sistema di batteria incorporato.

3.1.3 Quadcopter

Il Quadcopter è composto da 4 motori a corrente continua di tipo brushless, appositamente progettati per lo scopo. Due di questi motori di muovono in senso orario mentre gli altri due in senso antiorario, sono opposti a parità di rotazione e la rotazione deve sempre essere direzionata verso la parte di struttura perpendicolare all'asse di movimento in avanti. Tramite questa modalità di rotazione il drone è in grado di muoversi in una sfera mentre si trova in aria in quanto tutte le eliche riescono ad applicare una forza uguale o contraria, maggiore o minore, in base se si voglia alzarsi da terra, muoversi orizzontalmente, ruotare su sé stessi o discendere al suolo (vedi **Figura 3** per un esempio di sollevamento da terra).

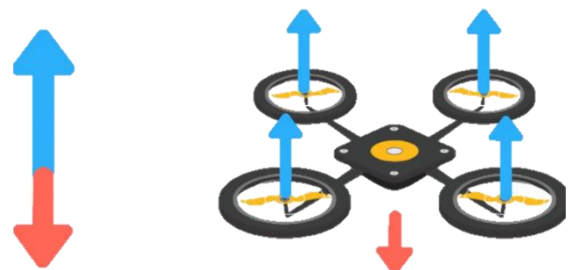


Figura 3

3.2 Programmazione

Per programmare il drone Tello EDU si hanno a disposizione 3 linguaggi: *Scratch*, *Python* e *Swift*. Il primo è un linguaggio di programmazione a blocchi, ovvero un metodo di programmazione visuale, dove non serve conoscere un vero e proprio codice, ma basta trascinare l'azione che si vuole eseguire in una coda di azioni. Gli altri due sono linguaggi a tutti gli effetti, dei quali Swift pensato e basato sulla tecnologia Apple.

3.2.1 Python

Per il progetto si è scelto di utilizzare il linguaggio **Python**, considerandolo il linguaggio più appropriato per questo lavoro in quanto uno dei suoi scopi è il **System Testing**.

Python è un linguaggio di programmazione di più *"alto livello"* rispetto alla maggior parte degli altri linguaggi, è multi-paradigma e ha tra i principali obiettivi la **dinamicità**, la **semplicità** e la **flessibilità**. Le caratteristiche distintive di questo linguaggio sono le variabili non tipizzate e l'uso dell'indentazione per la sintassi delle specifiche, al posto delle comuni parentesi.

Questo linguaggio è stato visionato durante il percorso scolastico come mezzo per lo sviluppo di sistemi anti-frode e analisi di dati.

Utilizzo nel progetto

Ryze mette a disposizione le librerie per la programmazione del drone in tutti e tre i linguaggi. Il loro scopo non è quello di dare un lavoro completo per la gestione del drone, ma fornire gli elementi di base che fungono da intermediario tra il linguaggio che si va a scegliere e gli Script del drone stesso.

Tramite queste librerie è possibile utilizzare delle parole chiave per far eseguire delle azioni al drone.

Test di volo

In un primo momento si è andati ad eseguire dei **test** sul drone *senza attivare la videocamera*, in questa maniera si poteva testare la parte fondamentale del progetto: **il volo**.

Per poter rendere più pulito il tutto è stato creato un sistema gerarchico (vedi **Figura 4**) composto da due file contenenti funzioni e librerie e un file che funge da “*Main*” il quale importa gli altri. Facendo ciò il codice risulta molto più chiaro, ma lascia comunque la possibilità di tutti i file eseguendo il programma principale. [Questa tecnica sarà poi usata anche per le pagine PHP].

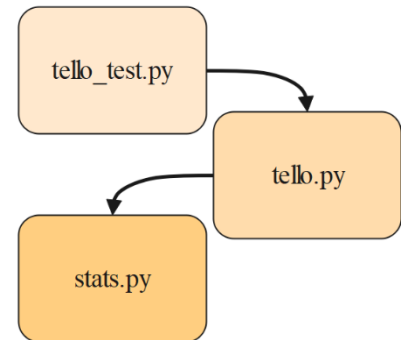


Figura 4

Oltre ai file Python è stato anche inserito un **file di testo** contenente tutte le *azioni* che dovrà eseguire il drone una volta avviato l'applicativo.

tello_test.py – Main

In questo primo file si esegue la lettura del file di testo contenente i vari comandi (vedi **Figura 5**). Inoltre il programma stampa in una **scheda di Debug** dell'ambiente di sviluppo ogni *azione* che viene *eseguita* o l'*errore* relativo ad un'azione non andata a buon fine.

```

tello = Tello()
file_comandi = open("comandi.txt", "r")
comandi = file_comandi.readlines()
for comando_corrente in comandi:
    if comando_corrente != '' and comando_corrente != '\n':
        comando_corrente = comando_corrente.rstrip()
        tello.send_command(comando_corrente)
  
```

Figura 5

tello.py – stats.py

Questi due file contengono “*comandi obbligati*” che servono per il **collegamento** e la **trasmissione** di comandi al drone e per la **stampa** delle *informazioni* relative ad ogni azione.

Di seguito un esempio di tello.py: l'invio di risposta da parte del drone alla ricezione del comando (vedi **Figura 6**).


```
def _receive_thread(self):
    while True:
        try:
            self.response, ip = self.socket.recvfrom(1024)
            print('from %s: %s' % (ip, self.response))

            self.log[-1].add_response(self.response)
        except socket.error or exc:
            print("Caught exception socket.error : %s" % exc)
```

Figura 6

Videocamera e movimento da tastiera

Una volta appurato il funzionamento delle librerie di Tello ed eseguiti i vari test si è passati all'implementazione della *videocamera e del controllo da tastiera*, per fare in modo che il drone si **muova in tempo reale** e non segua dei comandi predeterminati.

Anche questa volta sono stati creati 3 programmi distinti.

Il primo, main.py (vedi **Figura 7**), importa gli altri, esegue il collegamento al drone ed avvia il programma effettivo.

Il secondo, tello_control_ui.py, crea un'interfaccia con l'utente e gestisce i metodi alla pressione dei vari pulsanti.

L'ultimo, tello.py (vedi **Figura 8**), implementa al suo interno ciò che veniva eseguito nel precedente programma "stats.py", ma con l'aggiunta dei metodi di movimento e la ricezione e il decoding del video tramite una libreria sviluppata in C++ (vedi **Codifica H.264**).

Ogni ScreenShot effettuato verrà poi salvato all'interno di una cartella "img" contenuta nel repository del file "main.py".

```
import tello
from tello_control_ui import TelloUI

def main():
    drone = tello.Tello('', 8889)
    vplayer = TelloUI(drone, "./img/")
    vplayer.root.mainloop()

if __name__ == "__main__":
    main()
```

Figura 7

```
def _h264_decode(self, packet_data):
    res_frame_list = []
    frames = self.decoder.decode(packet_data)
    for framedata in frames:
        (frame, w, h, ls) = framedata
        if frame is not None:
            frame = np.fromstring(frame, dtype=np.ubyte, count=len(frame), sep='')
            frame = (frame.reshape((h, ls / 3, 3)))
            frame = frame[:, :w, :]
            res_frame_list.append(frame)
    return res_frame_list
```

Figura 8

3.2.2 C++

C++ è un linguaggio di programmazione *general purpose* sviluppato in origine come evoluzione del linguaggio C inserendo la programmazione orientata agli oggetti.

Il linguaggio è costituito da due parti: *il nucleo e la libreria standard*. Il primo è il codice stesso del programma, la seconda include invece la maggior parte della Standard Template Library (STL), che definisce strutture dati generiche, iteratori e algoritmi generici.

3.2.2.1 CMake

CMake è un software libero per *l'automazione dello sviluppo*. Il suo scopo è rimpiazzare Automake nella creazione dei Makefile, ovvero quei file che gestiscono le dipendenze, risolvendole e invocando programmi esterni per il lavoro necessario.

In breve, lo scopo di CMake è verificare che all'interno del computer siano presenti le librerie necessarie per il funzionamento di un programma e compilare un file con l'estensione desiderata per fare in modo che vengano fornite delle librerie funzionanti ad altri linguaggi.

3.2.2.2 Utilizzo nel progetto

Il drone Tello EDU per l'invio di immagini sfrutta uno *standard industriale* per la **compressione video**, in maniera tale da utilizzare *meno capacità nella memorizzazione e/o trasmissione*. Il suddetto standard è H.264 ed è considerato essenziale nel mondo moderno in quanto rende possibile l'interoperabilità tra prodotti diversi (quali televisore, videocamere, tablet, cellulari, ecc...). Esso esegue processi di predizione, trasformazione e codifica per produrre un flusso di bit H.264 compresso.

Non essendo però Python pensato per la codifica e decodifica del video si è dovuti appoggiarsi a C++ per la creazione della libreria necessaria per il funzionamento della ricezione video.

Visual Studio è fornito di librerie per lo sviluppo di programmi per Linux in C++ che vanno a implementare un compilatore CMake, quindi si è scelto di utilizzare l'IDE in questione per la programmazione del *decoder del video*, in quanto l'encoding è già eseguito dal drone e non resta che decodificare per poter visualizzare l'immagine.

Qui sarà esposto solo il procedimento di codifica da parte del drone, in quanto il processo di decodifica è l'inverso dello stesso.

Codifica H.264

Per poter successivamente ritornare allo stato originale bisognerà conoscere le dimensioni ipotetiche del frame compresso da trasmettere, quindi il codificatore crea un frame video in Macroblock (un componente di compressione) di dimensioni 16x16pixel. Tramite questo genera una "predizione" basata su dati precedentemente codificati, sia che essi siano dello stesso frame (Intra Prediction) sia di frame già codificati e trasmessi (Inter Prediction).

La predizione di cui sopra restituisce un insieme di coefficienti, ciascuno dei quali va a definire la ponderazione per il passaggio successivo. Questi, se combinati, vanno a ricreare l'intera immagine.

Ora ogni output di coefficienti viene quantizzato, ovvero viene ridotta la precisione dei coefficienti. Questo fa sì che, all'interno della codifica, la maggior parte dei valori sia 0. Per definire la precisione si utilizza un parametro denominato QP (Quantization Parameter). Più il QP è alto più sarà alto il rapporto di compressione, ma a risentirne sarà la risoluzione dell'immagine. Viceversa, più il QP diminuisce, più l'immagine sarà di risoluzione elevata, ma con essa il rapporto di compressione diminuirà drasticamente.

Dopo questi processi il risultato deve essere codificato tramite dei valori per formare il bitstream compresso. Questi valori sono chiamati elementi di sintassi e comprendono:

- i coefficienti QP;
- informazioni che permetteranno al decodificatore di ricreare la previsione iniziale;
- informazioni sulla struttura dei dati e sugli strumenti di compressione;
- informazioni sul video completo.

Una volta identificati i vari elementi di sintassi, questi sono convertiti in codice binario utilizzando una codifica a lunghezza variabile o una codifica aritmetica.

Di seguito qualche riga di codice di esempio per la decodifica di H.264, in particolare l'inversione di predizione delle dimensioni del frame (vedi **Figura 9**).

```
class ConverterRGB24
{
    SwsContext *context;
    AVFrame *framergb;

public:
    ConverterRGB24();
    ~ConverterRGB24();
    int predict_size(int w, int h);
    const AVFrame& convert(const AVFrame &frame, unsigned char* out_rgb);
};
```

Figura 9

3.2.3 MySQL

MySQL è un RDBMS (*Relational Database Management System*) composto da un Client e riga di comando e un Server.

È quindi un database basato sul modello relazionale, dove tutti i dati sono rappresentati come relazioni e sono manipolati con gli operatori logici dell'algebra o del calcolo relazionale. Questo modello consente al progettista di creare una rappresentazione consistente e logica dell'informazione.

Utilizzo nel progetto

Si è utilizzato un database nel progetto per avere la possibilità di salvare i video registrati dal drone, per far sì di poterli visionare anche in seguito.

Progettazione Concettuale - Modello Entità/Relazione

Il modello E/R è un linguaggio visuale per la schematizzazione del modello concettuale di un database. Questo offre al progettista un insieme di costrutti finalizzati alla rappresentazione astratta dei concetti presenti nel dominio applicativo che si intende informatizzare.

Progettazione Logica – Ristrutturazione E/R

Una volta messo in pratica il modello E/R non resta che normalizzarlo. Per fare ciò bisogna eliminare la ridondanza e gestire le relazioni tra le varie entità.

Un cruciale punto del progetto è il fatto che i database non possono contenere al loro interno dei video. Per ovviare a ciò si è deciso di creare una pagina PHP in grado di inserire all'interno di una cartella specifica i vari video registrati, per poi andare a specificare all'interno di una tabella il percorso del file.

Il database è composto da 4 tabelle:

- utenti: contiene tutti gli utenti che, tramite un Form di Login (vedi **Login**), possono accedere ai video;
- log: tiene conto di tutti i login effettuati con successo, in maniera tale da poter ovviare a problemi di sicurezza;
- posizione: contiene tutti i comuni d'Italia;
- video: al suo interno si trovano vari dati di un video, tra cui il nome dello stesso file.

Le tabelle del database di seguito (vedi **Figura 10**).



Figura 10

3.2.4 PHP

PHP (*PHP: Hypertext Preprocessor*) è un linguaggio di scripting interpretato, originariamente concepito per la programmazione di pagine web dinamiche.

È principalmente utilizzato per sviluppare applicazioni web lato server, ma può essere usato anche per scrivere script a riga di comando o applicazioni stand-alone con interfaccia grafica.

3.2.4.1 HTML e CSS

HTML (*HyperText Markup Language*) è un linguaggio di markup di pubblico dominio, nato per la formattazione e impaginazione di documenti ipertestuali disponibili nel web 1.0. Oggi è utilizzato principalmente per il disaccoppiamento della struttura logica di una pagina web e la sua rappresentazione, gestita tramite gli stili CSS.

CSS (*Cascading Style Sheets*) è un linguaggio usato per definire la formattazione di documenti HTML, XHTML e XML, ad esempio i siti web e relative pagine. L'introduzione del CSS si è resa necessaria per separare i contenuti delle pagine GRML dalla loro formattazione o layout e permettere

una programmazione più chiara e facile da utilizzare, sia per gli autori delle pagine stesse sia per gli utenti.

3.2.4.2 Utilizzo nel progetto

Per la visualizzazione dei video, volendo simulare un drone della forza dell'ordine, si sono andate a creare tre pagine web:

- pagina di Login;
- pagina Video (contiene semplicemente tutti i video. Da qui è inoltre possibile rimuoverli);
- pagina Upload.

Le pagine web sono state sviluppate sfruttando XAMPP come appoggio per il sito in LocalHost.

Login

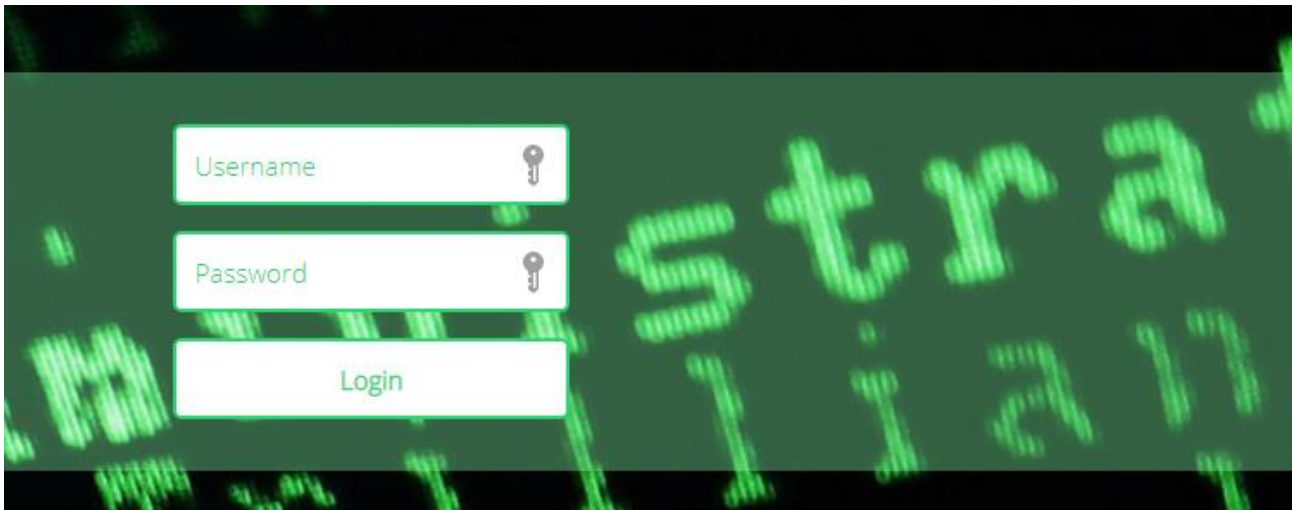


Figura 11

La pagina di **Login** è stata sviluppata per la gestione degli accessi, per fare in modo che a poter visualizzare tutti i video siano solo operatori predestinati e non malintenzionati o terzi i quali non possono avere possibilità di vedere immagini che di per sé violerebbero le leggi sulla Privacy.

Nella pagina sono contenuti *due campi di testo e un pulsante* (vedi **Figura 11**). Alla pressione di questo viene fatta una richiesta al database di un eventuale utente che ha Username e Password specificati e, nel caso non sia presente, nega l'accesso. Sotto, l'esempio di codice per il controllo di ciò (vedi **Figura 12**).

```
if(isset($_POST['login'])){
    $user = $_POST['username'];
    $pw = $_POST['password'];
    $query = "SELECT id FROM utenti WHERE user='$user' AND pw='$pw'";
    $result = $db->query($query);
    if ($result->num_rows == 1) {
        $row = $result->fetch_assoc();
        $id = $row['id'];
        $_SESSION['id'] = $id;
        $data = date('d-m-Y H:i:s');
        $query = "INSERT INTO log(userId, dataLogin) VALUES ('$id', '$data')";
        $result = $db->query($query);
        header("Location: video.php");
    }
}
```

Figura 12

Upload

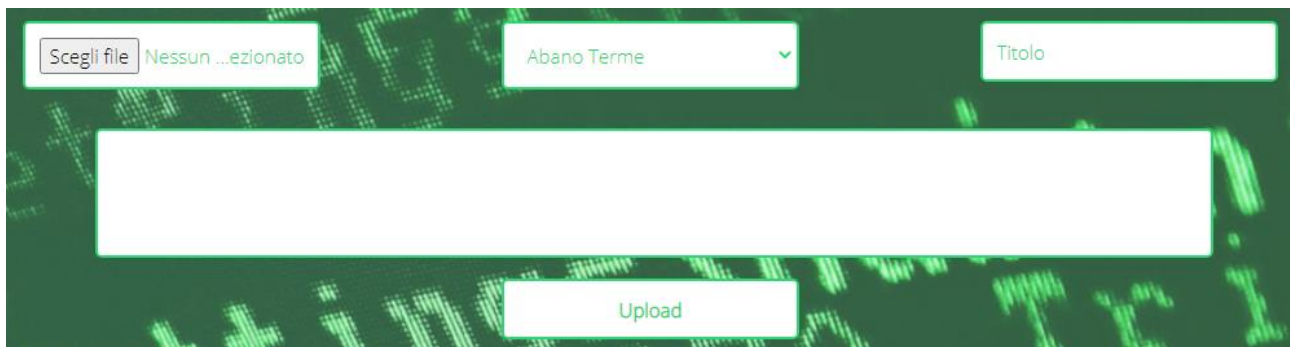


Figura 13

La pagina **Upload** funge da intermediario per il caricamento del video. I campi da compilare (vedi **Figura 13**) sono tutti obbligatori, o verrà restituito un errore in caso non contengano nulla.

Una volta pigiato sul testo dell'Upload verrà salvato in una cartella specifica il video, rinominato con un nome generato casualmente in base all'ora di inserimento per poter aggirare il sistema di nomi dell'OS cosicché più video possano avere lo stesso titolo all'interno del database. Inoltre verrà salvato nel database il percorso del file, con titolo, descrizione, luogo e data di caricamento. Di seguito un segmento di codice incaricato di caricare i dati nella tabella e del salvataggio del video all'interno della cartella (vedi **Figura 14**).

```
if($peso <= 201 && $peso != 0){
    if(move_uploaded_file($_FILES["filevideo"]["tmp_name"], './docs/' . $newfilename)){
        $stmt = $db->query("INSERT INTO video(percorso, titolo, descrizione, inseri-
mento, luogo) VALUES ('$newfilename', '$titolo', '$desc', '$data', $luogo)");
        header("Location: video.php");
    }else{
        header("Location: ?e1");          // Errore di Upload
    }
}else{
    header("Location: ?e2".$peso);      // Errore di Upload
}
```

Figura 14

Caratteristiche comuni delle pagine

Ogni pagina presenta inoltre degli elementi in comune:

- una barra di navigazione che, in base a dove ci si trova e se si è eseguito l'accesso, rimanda o alla pagina di Login o a quella dei Video;
- il file **'funzione.php'**, che contiene funzioni comuni tra tutte le pagine e che è incluso in ogni file PHP del progetto.

4. Sviluppi Futuri

Inutile dire però che non possiamo far eseguire tutti i comandi, anche quelli più semplici, da un operatore, il quale dovrebbe stare ore intere di fronte a uno schermo o indossando occhiali specifici per “vedere con gli occhi del drone”. È appunto qui che entra veramente in gioco l'evoluzione dell'informatica e dei sistemi che la regolano.

Questa tecnologia è dotata dei più vari strumenti di controllo o scansione, ma non sono in grado di muoversi da soli per le varie zone senza rischiare di andare a urtare un albero o andare fuori traiettoria per colpa del vento o delle correnti d'aria. Per esempio specifici droni della protezione civile in Italia sono dotati di telecamere con sensori ad infrarossi a lunga gittata che permettono loro di poter trovare feriti tra macerie di una casa o dispersi tra i monti italiani, ma non hanno la possibilità di farlo 24/24h per motivi innanzitutto di batteria ma anche perché un operatore dovrebbe stare a fissare il monitor per tutto il tempo. Inoltre le gittate raggiungono al massimo una ventina di chilometri e quindi dovrebbe esserci una centralina di controllo ogni 40 per poter avere una copertura completa e questa cosa è impensabile.

Ed è proprio a questo punto che entrano in gioco le intelligenze artificiali (IA), adocchiate sin da subito dai più e meno esperti, scettici e non. Ma cos'è l'Intelligenza Artificiale? In modo semplicistico è l'abilità di un sistema tecnologico di risolvere problemi o svolgere compiti e attività tipici della mente e dell'abilità umane. Guardando al settore informatico, potremmo identificare l'IA come la disciplina che si occupa di realizzare macchine in grado di "agire" autonomamente. Ed è proprio questa loro funzione ad essere interessante: quella di saper decidere cosa fare.

Se chiedessimo a uno dei reparti disciolti del reggimento alpino di pattugliare le Alpi e riferire al quartier generale, senza dare ulteriori istruzioni, questo seguirebbe il proprio caporale e per suoi ordini si dividerebbe in base alla griglia della zona per poi riferire ogni informazione su posizione e immagini al caporale, che lo inoltrerebbe al QG.

Ora immaginiamo che al posto di quei soldati ci siano dei droni, dei piccoli dispositivi volanti, facilmente occultabili, veloci e precisi. Tramite ordini umani vengono definiti un generale e vengono dati ordini.

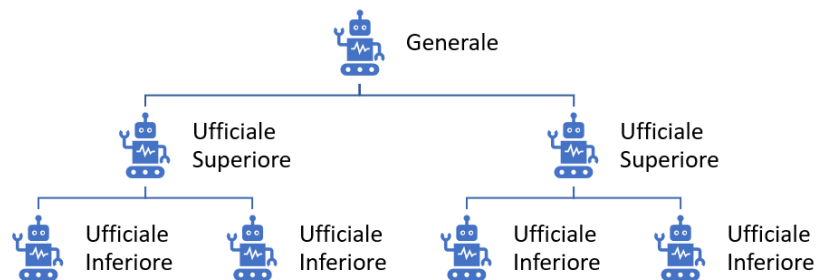


Figura 15

Ora tocca a lui scegliere i propri ufficiali superiori, che a loro volta sceglieranno quelli inferiori e così via in base alla dimensione da perlustrare (vedi **Figura 15**). Ora tutti saranno al servizio del proprio superiore, dovendo obbedire ciecamente agli ordini, ma potendone dare ai propri sottoposti. Inoltre ad ogni azione "sospetta" verrebbe trasmessa un'immagine al superiore, che a sua volta la inoltrerebbe ancora al proprio superiore, e ancora fino ad arrivare al generale che deciderà se sia rilevante o meno e agirà in base a questa scelta.

Quanto scritto sopra è possibile tramite le blockchain che permettono di creare uno sciame di droni, ovvero un agglomerato di droni che si muovono seguendo un "master" che riferisce agli altri la sua posizione e ordina cosa fare in base ad essa o a delle azioni specifiche.

L'implementazione di questi metodi è però ancora in fase di sviluppo, in quanto bisogna tenere conto di molte variabili di vario genere, tra le quali: la durata della batteria, il raggio d'azione dei droni, quali e quante porte dedite alla trasmissione di dati o per il collegamento tra droni avere e aprire, la capacità di calcolo del sistema di elaborazione di scelte.

5. Conclusioni

In conclusione i droni sono, a parere mio, uno dei risultati più riusciti dell'unione tra automazione e informatica. Sono in grado di far vedere il mondo da un'altra prospettiva, letteralmente. Un hobby, uno strumento militare, un semplice gioco: i droni hanno mille scopi in base a come li si voglia utilizzare.

Se devo essere sincero da un lato ho trovato limitativo il fatto di dover per forza utilizzare un drone d'aria, mentre dall'altro mi ritengo fortunato a non aver dovuto programmare droni anfibi o simili in quanto non ce l'avrei mai fatta.

Devo ammettere che è stata una vera sfida: sono stato catapultato in un mondo che conoscevo solo di nome, ho dovuto studiare quasi da zero due linguaggi diversi, aprire la mente e dare libero sfogo alla fantasia per cercare di ottimizzare al meglio il progetto con ciò che avevo. Ho applicato le conoscenze e le esperienze di tre anni, dall'alternanza ai molteplici programmi alla gestione del tempo per riuscire a portare avanti lo studio, la programmazione e la scrittura dell'elaborato.

Il progetto presenta ampi margini di miglioramento, ma sono orgoglioso di essere riuscito a programmare una base di ciò che in futuro potrebbe essere non solo uno svago, ma a seconda dei casi uno strumento che potrebbe salvare delle vite.

6. FONTI

An Overview of H.264 Advanced Video Coding. (s.d.). Tratto da Vcodex:
<https://www.vcodex.com/an-overview-of-h264-advanced-video-coding/#:~:text=264%20%3F-,H.,it%20is%20stored%20or%20transmitted.&text=An%20encoder%20converts%20video%20into,back%20into%20an%20uncompressed%20format.>

Biasiotti, A. (2019, Novembre 27). *Una nuova proposta di identificazione distanza dei droni.* Tratto da Punto Sicuro: <https://www.puntosicuro.it/security-C-124/security-C-125/una-nuova-proposta-di-identificazione-distanza-dei-droni-AR-19602/>

Boldrini, N. (2020, Febbraio 28). *Intelligenza Artificiale (AI): cos'è, come funziona e applicazioni 2020.* Tratto da AI4BUSINESS: <https://www.ai4business.it/intelligenza-artificiale/intelligenza-artificiale-cose/>

Build fast, responsive sites with Bootstrap. (s.d.). Tratto da Bootstrap: <https://getbootstrap.com/>

C++. (s.d.). Tratto da Wikipedia: <https://it.wikipedia.org/wiki/C%2B%2B>

CMake. (s.d.). Tratto da Wikipedia: <https://it.wikipedia.org/wiki/CMake>

Codifica aritmetica. (s.d.). Tratto da Wikipedia: https://it.wikipedia.org/wiki/Codifica_aritmetica

CSS. (s.d.). Tratto da Wikipedia: <https://it.wikipedia.org/wiki/CSS>

Ferraro, M. (2021, Marzo 23). *Google lancia un'applicazione che permette a due dispositivi di connettersi senza Internet.* Tratto da Il Mattino: https://www.ilmattino.it/tecnologia/app_e_giochi/google_lancia_un_applicazione_permette_dispositivi_di_connettersi_loro_senza_la_connessione_internet-5850749.html#:~:text=Google%20ha%20lanciato%20una%20nuova,quanto%20riferito%20dal%20portale%209to5Google.

Gribaudo, M. (s.d.). *Sistemi Multimediali - Codifiche a lunghezza variabile.*

How to store videos in database. (2011, Settembre 28). Tratto da CODE PROJECT: <https://www.codeproject.com/Questions/261542/how-to-store-videos-in-database>

HTML. (s.d.). Tratto da Wikipedia: <https://it.wikipedia.org/wiki/HTML>

La storia dei droni in 10 tappe. (2018, Ottobre 10). Tratto da PROFESSIONAL AVIATION:
<https://www.professionalaviation.it/la-storia-dei-droni-in-10-tappe/#:~:text=Nel%201917%20l'Aerial%20Target,essere%20pilotato%20verso%20i%20nemici.>

Macroblock. (s.d.). Tratto da Wikipedia: <https://en.wikipedia.org/wiki/Macroblock>

make. (s.d.). Tratto da Wikipedia: <https://it.wikipedia.org/wiki/Make>

Massimo, N. (2021, Maggio 19). *Tipologie di Droni: Analizziamo le diverse tipologie dei droni.*
Tratto da wondershare: <https://filmora.wondershare.it/drones/types-of-drones.html>

Modello Relazionale. (s.d.). Tratto da Wikipedia: https://it.wikipedia.org/wiki/Modello_relazionale

MySQL. (s.d.). Tratto da Wikipedia: <https://it.wikipedia.org/wiki/MySQL>

Normativa Droni. (s.d.). Tratto da ENAC - Ente Nazionale per l'Aviazione Civile:
<https://www.enac.gov.it/sicurezza-aerea/droni/normativa-droni>

PHP. (s.d.). Tratto da Wikipedia: <https://it.wikipedia.org/wiki/PHP>

Python. (s.d.). Tratto da Wikipedia: <https://it.wikipedia.org/wiki/Python>

Relational database management system. (s.d.). Tratto da Wikipedia:
https://it.wikipedia.org/wiki/Relational_database_management_system

Standard Template Library. (s.d.). Tratto da Wikipedia:
https://it.wikipedia.org/wiki/Standard_Template_Library

STORIA ED EVOLUZIONE DEI DRONI: GLI APPARECCHI SAPR DALLA NASCITA AL FUTURO. (2019, Novembre 7). Tratto da EGM96 - Drone solution Provider:
https://www.egm96.it/storia-ed-evoluzione-dei-droni-gli-apparecchi-sapr-dalla-nascita-al-futuro/#_ngl1ce75vzht

Tello EDU. (s.d.). Tratto da Ryze Robotics: <https://www.ryzerobotics.com/tello-edu>

IL DIRIGENTE SCOLASTICO

Francesco Daniele Laterza