

# **PROGETTO Drone Control System |**

## **Diario di lavoro - 13.02.2019**

Luca Di Bello, Fadil Smajilbasic, Andrea Rauso, Jari Näser

**Canobbio, 13.02.2019**

### **Lavori svolti**

Nella prima giornata del progetto abbiamo letto e analizzato il quaderno dei compiti, in seguito abbiamo stilato il Gantt e la struttura del progetto.

Abbiamo cominciato la documentazione del progetto partendo dalla stesura dell'analisi. Abbiamo fatto delle ricerche su Leap Motion e Tello.

### **Problemi riscontrati e soluzioni adottate**

### **Punto della situazione rispetto alla pianificazione**

In orario

### **Programma di massima per la prossima giornata di lavoro**

Continuazione dell'analisi del progetto

# PROGETTO Drone Control System |

## Diario di lavoro - 15.02.2019

Luca Di Bello, Fadil Smajilbasic, Andrea Rauso, Jari Näser

**Canobbio, 15.02.2019**

### Lavori svolti

Durante la giornata odierna il gruppo si é diviso i compiti, Luca e Fadil gestiscono la parte client del progetto, ovvero la gestione dei valori del sensore Leap Motion e i comandi del drone DJI tello, mentre Andrea e Jari gestiscono la parte server che riceve i comandi del drone, i cui dati verranno poi visualizzati in un frame.

Per la visualizzazione dei dati forniti abbiamo deciso di usare la libreria JFreeChart.

### Problemi riscontrati e soluzioni adottate

- 

### Punto della situazione rispetto alla pianificazione

In orario

# PROGETTO Drone Control System |

## Diario di lavoro - 20.02.2019

Luca Di Bello, Fadil Smajilbasic, Andrea Rauso, Jari Näser

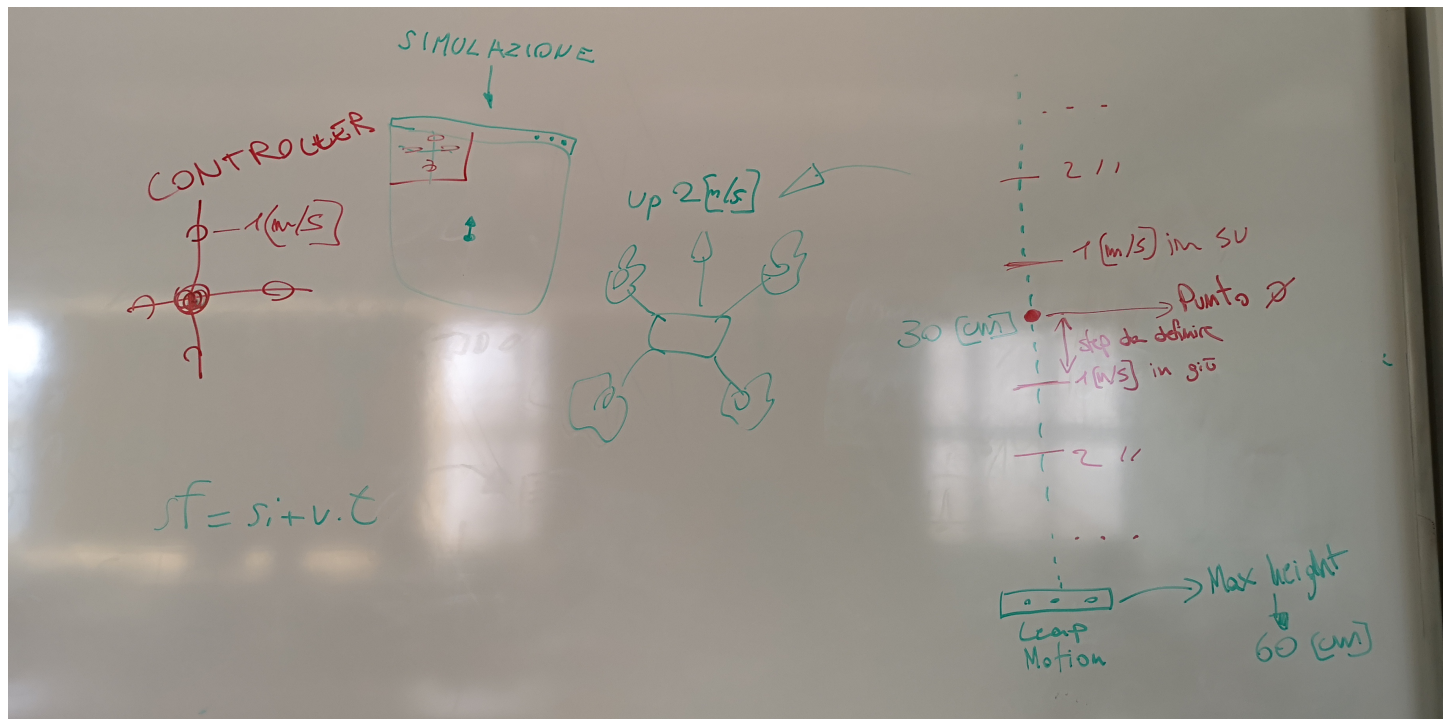
Canobbio, 20.02.2019

### Lavori svolti

Durante la giornata odierna il gruppo ha deciso come verranno letti i dati dal sensore Leap Motion e come verranno visualizzati.

Il sensore leggerà l'altezza e/o l'inclinazione della mano che indicherà la velocità di spostamento su un determinato asse (nell'immagine sottostante a destra), la conversione di questi valori in velocità dovrà ancora essere deciso. La velocità verrà mostrata in un frame su un diagramma a tre assi (nell'immagine sottostante a sinistra).

Qui di seguito una rappresentazione:



Fadil ha creato la classe LeapMotionReader che permette di leggere i dati dal sensore Leap Motion quali la posizione e la rotazione (beccheggio, rollio e imbardata) sui 3 assi di movimento della mano.

Luca e Jari si sono dedicati alla gestione delle classi per l'invio e della ricezione dei comandi

Andrea ha continuato l'analisi della libreria per creare i grafici per il drone

## **Problemi riscontrati e soluzioni adottate**

Nella gestione dei pacchetti il controller riesce a inviare e ricevere correttamente i pacchetti UDP mentre il simulatore riesce a ricevere ma non a inviare i pacchetti verso il controller

## **Punto della situazione rispetto alla pianificazione**

In orario

# PROGETTO Drone Control System | Diario di lavoro - 22.02.2019

Luca Di Bello, Fadil Smajilbasic, Andrea Rauso, Jari Näser

**Canobbio, 22.02.2019**

## Lavori svolti

Andrea:

Io ho finito l'analisi della libreria JFreeChart per la creazione dei grafici per il simulatore del drone.

Luca:

Io ho iniziato a sviluppare una classe che sfrutta le funzionalità implementate dalla classe "CommandManager" e "LeapMotionReader". Essa è chiamata "DroneController" e si occupa di leggere i dati dal Leap Motion ed inviarli tramite protocollo UDP al drone (simulatore) tramite la classe CommandManager.

Jari:

Io ho creato tutta la struttura dei metodi come quella fornita dalla SDK di tello nella classe CommandReader.java, inoltre sta lavorando sulla corretta interpretazione delle richieste dei vari metodi mandati via pacchetto UDP dalla classe Controller.java alla Simulator.java.

## Problemi riscontrati e soluzioni adottate

Riguardo al problema dell'invio dei pacchetti dal simulatore verso il controller è stato risolto impostando una porta di ascolto nel controller che in precedenza non è stata impostata.

Luca durante lo sviluppo della classe "DroneController" si è reso conto che il suo ambiente di sviluppo (NetBeans) non era impostato correttamente per lo sviluppo utilizzando la libreria "LeapJava" (libreria per l'utilizzo di Leap Motion tramite il linguaggio Java). Quando Luca cercava di compilare il programma, il compilatore mostrava un warning con questa dicitura:

```
"Native code library failed to load. java.lang.UnsatisfiedLinkError: no LeapJava in java.library.path".
```

Grazie all'aiuto di Fadil Smajilbasic Luca è riuscito a trovare una soluzione, questi sono i vari step:

1. Scaricare nuovamente la libreria dal sito ufficiale
2. Aggiungere la libreria al progetto
3. Aggiungere una stringa di configurazione della VM la quale serve a specificare la posizione dei file .dll necessari per il corretto funzionamento della libreria "LeapJava":  

```
-Djava.library.path="path fino ai file dll della libreria"
```

Stringa utilizzata da Luca:  

```
-Djava.library.path="C:\Users\luca6\Desktop\libs\x64\"
```
4. Compilato (build) il progetto
5. Il programma adesso si avvia correttamente

## Punto della situazione rispetto alla pianificazione

Siamo al passo con il gantt preventivo.

## Programma di massima per la prossima giornata di lavoro

Luca cercherà di creare una versione beta del controller del drone (DroneController), la quale permetterà di controllare solo l'altitudine del drone.

Jari invece continuerà lo sviluppo della classe CommandReader.java nella quale cercherà di simulare l'esatto comportamento del drone tello attraverso varie variabili con i vari rispettivi metodi.

# PROGETTO Drone Control System |

## Diario di lavoro - 27.02.2019

Luca Di Bello, Fadil Smajilbasic, Andrea Rauso, Jari Näser

**Canobbio, 27.02.2019**

### Lavori svolti

Andrea:

Ho messo i file della libreria JFreeChart all'interno del progetto e ho creato una classe per mostrare i valori della rotazione del drone.

Fadil:

Sono riuscito a far leggere al DroneController i dati dal LeapMotion, inoltre ho migliorato l'integrazione tra la classe Dronecontroller e LeapMotionReader

Jari:

Ho continuato con lo sviluppo della classe CommandReader nella quale ho ristrutturato la struttura di vari metodi e ho continuato ad implementare le simulazioni dei vari metodi di movimento del drone. Inoltre ho anche creato la struttura delle variabili assieme ad Andrea per poter lavorare su di esse e per semplificare la simulazione disegnata del drone attraverso il loro utilizzo.

Luca:

Aiutato fadil modificando la classe DroneController

### Problemi riscontrati e soluzioni adottate

### Punto della situazione rispetto alla pianificazione

Siamo al passo con il gantt preventivo.

### Programma di massima per la prossima giornata di lavoro

Jari continuerà con lo sviluppo dei metodi di movimento.

# PROGETTO Drone Control System |

## Diario di lavoro - 01.03.2019

Luca Di Bello, Fadil Smajilbasic, Andrea Rauso, Jari Näser

**Canobbio, 01.03.2019**

### Lavori svolti

Jari:

Non presente

Fadil:

Io ho creato il diagramma delle classi sia per il DroneController che per il DroneSimulator. Ho apporato alcune modifiche alle classi DroneController e LeapMotionReader. Ho anche eliminato i file di configurazione del progetto dalla repo poichè avevamo dei problemi di compatibilità tra netbeans di windows, macos e ubuntu.

Luca:

Non presente

Andrea:

Non presente

### Problemi riscontrati e soluzioni adottate

### Punto della situazione rispetto alla pianificazione

Siamo al passo con il gantt preventivo.

### Programma di massima per la prossima giornata di lavoro



# PROGETTO Drone Control System |

## Diario di lavoro - 13.03.2019

Luca Di Bello, Fadil Smajilbasic, Andrea Rauso, Jari Näser

**Canobbio, 13.03.2019**

### Lavori svolti

Andrea:

Ho creato il mockup delle interfacce che andremo ad utilizzare per mostrare i dati di posizione e rotazione del drone e ho cominciato lo sviluppo dei grafici in cui verranno visualizzati la posizione del drone dall'alto e di profilo

Jari:

Ho continuato con lo sviluppo della classe CommandReader nella quale sto finendo di simulare il movimento del drone al ricevimento dei vari comandi della SDK di Tello.

Fadil:

Ho modificato la classe LeapMotionReader rinominandola FrameHelper, poichè conteneva solo metodi utili a ricavare delle informazioni da un frame, in più ho aggiunte altre funzionalità a quella classe. Ho integrato le funzionalità della classe FrameHelper nella classe DroneController. Devo ancora correggere la classe CommandManager di Luca per farla funzionare con la struttura che abbiamo adesso.

Luca:

Ho completato la classe Commands, implementando tutti i comandi supportati dal drone DJI Tello. Il comando "curve" ed il comando "rc" non sono ancora stati implementati.

### Problemi riscontrati e soluzioni adottate

Non abbiamo riscontrato nessun problema.

### Punto della situazione rispetto alla pianificazione

Siamo al passo con il gantt preventivo.

### Programma di massima per la prossima giornata di lavoro

Jari: Se possibile finire la classe CommandReader e fare dei primi test unendola a Simulator.

# PROGETTO Drone Control System |

## Diario di lavoro - 15.03.2019

Luca Di Bello, Fadil Smajilbasic, Andrea Rauso, Jari Näser

**Canobbio, 15.03.2019**

### Lavori svolti

Jari:

Ho concluso lo sviluppo ed implementazione della maggior parte dei metodi di movimento del drone.

Fadil:

Io integrato la mia classe DroneController con le classi CommandManager e FrameHelper. Adesso possiamo leggere la posizione della mano tradurla in comandi del drone e mandare il comando al drone.

Luca:

Non era presente alla lezione.

Andrea:

Non era presente alla lezione.

### Problemi riscontrati e soluzioni adottate

Non abbiamo riscontrato nessun problema.

### Punto della situazione rispetto alla pianificazione

Siamo al passo con il gantt preventivo.

### Programma di massima per la prossima giornata di lavoro

Jari: Testare il funzionamento dei vari metodi già fatti collegandola alla classe CommandManager e continuare a sviluppare i metodi non ancora conclusi.

# PROGETTO Drone Control System |

## Diario di lavoro - 20.03.2019

Luca Di Bello, Fadil Smajilbasic, Andrea Rauso, Jari Näser

**Canobbio, 20.03.2019**

### Lavori svolti

Jari:

Ho continuato lo sviluppo della classe CommandReader e assieme a Luca e Fadil abbiamo iniziato a testare la comunicazione fra le svariate classi correggendo i primi errori.

Fadil:

Ho corretto vari errori nella classe DroneController e ho migliorato il modo in cui scopo il movimento della mano.

Luca:

Svolto delle piccole modifiche alla classe Commands ed alla classe DroneController

Andrea:

Ho finito di creare le interfacce in cui verranno mostrati i dati di posizione e di rotazione del drone.

### Problemi riscontrati e soluzioni adottate

Non abbiamo riscontrato nessun problema.

### Punto della situazione rispetto alla pianificazione

Siamo al passo con il gantt preventivo.

### Programma di massima per la prossima giornata di lavoro

Jari: Se possibile integrare le classi fatte da Andrea e fare una prima simulazione grafica.

# PROGETTO Drone Control System |

## Diario di lavoro - 22.03.2019

Luca Di Bello, Fadil Smajilbasic, Andrea Rauso, Jari Näser

**Canobbio, 22.03.2019**

### Lavori svolti

Jari:

Assieme a Fadil e Luca abbiamo cercato di capire meglio il funzionamento del pitch, roll e yaw sul quale mi sono anche informato in modo dettagliato dovendo anche usarli per calcolare attitude e acceleration, e tof.

Inoltre abbiamo reso la connessione fra le classi Simulator e CommandManager molto stabile e funzionante anche su svariate piattaforme come pc e sistemi operativi diversi.

Fadil:

Ho cambiato ancora modo per scoprire il movimento della mano, per il regolamento della altezza, adesso uso la velocità della mano per scoprire il movimento. Ho aggiunto i metodi utili, per rilevare il movimento della mano destra, alla classe FrameHelper che userò nella classe Dronecontroller per mandare i comandi relativi allo spostamento di rollio, beccheggio e imbardata.

Luca:

Finito la classe SettingManager utile per gestire i file di config.

Andrea:

Ho collegato le classi di visualizzazione di dati al simulatore in modo che si possa vedere i valori mandati dal drone. Inoltre all'interno di queste classi ho cambiato il tipo di valore da float a int in modo da essere in regola con la struttura dei dati del simulatore.

### Problemi riscontrati e soluzioni adottate

Non abbiamo riscontrato nessun problema.

### Punto della situazione rispetto alla pianificazione

Siamo al passo con il gantt preventivo.

### Programma di massima per la prossima giornata di lavoro

-

# PROGETTO Drone Control System |

## Diario di lavoro - 27.03.2019

Luca Di Bello, Fadil Smajilbasic, Andrea Rauso, Jari Näser

**Canobbio, 27.03.2019**

### Lavori svolti

Jari:

Assieme a Fadil e Andrea abbiamo messo assieme tutte le classi e abbiamo ottenuto una prima simulazione grafica funzionante del movimento del drone sull'asse Y.

Inoltre abbiamo ragionato su come trattare il pitch, roll e yaw; Sia per la rappresentazione grafica che per il passaggio dei dati tramite i pacchetti.

Questa cosa sarà da stabilire e da metterci d'accordo con il resto del gruppo.

Fadil:

Io e Jari abbiamo lavorato ancora al collegamento tra il controller e il simulatore. Ho provato a leggere i valori del rollio beccheggio e imbardata dai vettori creati alle punta delle dita al posto di prendere quelli letti dal centro della mano. Devo ancora valutare quale approccio scegliere.

Luca:

Non presente.

Andrea:

Ho cominciato a unificare tutte le classi di visualizzazione dei dati sotto un unico frame in modo da ottenere un'unica finestra contenente tutti i dati e non 3 finestre separate. Ho modificato il piano cartesiano dove viene mostrata la posizione dall'alto e di profilo, i valori sui 2 assi in entrambi i piani vanno da -500 a +500.

### Problemi riscontrati e soluzioni adottate

Non abbiamo riscontrato nessun problema.

### Punto della situazione rispetto alla pianificazione

Siamo al passo con il gantt preventivo.

### Programma di massima per la prossima giornata di lavoro

-



# PROGETTO Drone Control System |

## Diario di lavoro - 29.03.2019

Luca Di Bello, Fadil Smajilbasic, Andrea Rauso, Jari Näser

**Canobbio, 29.03.2019**

### Lavori svolti

Jari:

Io ho continuato e terminato lo sviluppo ed implementazione dei metodi della SDK di tello nella classe CommandReader.

Inoltre ho anche implementato la risposta via pacchetto UDP ai metodi getter che finiscono per '?' ad esempio: 'battery?'.

Infine ho contribuito a migliorare la comunicazione e il funzionamento fra le varie classi una volta messe assieme.

Fadil:

Ho implementato il calcolo dell'angolo di rollio tra il pollice e il mignolo della mano destra e assieme a Luca abbiamo aggiunto la possibilità di assegnare dei parametri al DroneController grazie a un file di config

Luca:

Ho commentato e terminato l'implementazione della classe 'SettingsManager' (la classe che gestisce il file di config). Ho sviluppato anche una classe chiamata 'ControllerSettings', la quale si appoggia alla classe 'SettingsManager', che permette di accedere facilmente ai dati letti dal file di config tramite delle variabili in sola lettura.


Andrea:

Ho collegato la classe TelloChartFrame che mostra tutti i dati del drone al simulatore e ho risolto alcuni errori di visualizzazione dei dati per cui veniva fatto il refresh di diversi dati sullo stesso grafico

### Problemi riscontrati e soluzioni adottate

Fadil:

Avevo dei problemi mentre cercavo di leggere l'angolo di rollio beccheggio e imbardata, poiché l'angolo letto non era quello corretto.

Ho trovato e implementato questa soluzione tramite StackOverflow, questo  il link:

<https://stackoverflow.com/a/9970297>

## Punto della situazione rispetto alla pianificazione


Siamo al passo con il gantt preventivo.

### Programma di massima per la prossima giornata di lavoro

Jari:

Continuare a sviluppare le classi CommandReader e Simulator con le features scritte sotto forma di commento nella classe CommandReader.

Luca:

- Una GUI per il controller che permetterà  di cambiare delle impostazioni sul volo "On Air" (quindi senza dover far atterrare e ripartire il drone)
- Implementare nella classe 'SettingsManager' dei metodi utili per la modifica delle impostazioni nel file di config (senza dover modificarlo a mano)

# PROGETTO Drone Control System |

## Diario di lavoro - 03.04.2019

Luca Di Bello, Fadil Smajilbasic, Andrea Rauso, Jari Näser

**Canobbio, 03.04.2019**

### Lavori svolti

Jari:

Ho continuato con il raffinamento della comunicazione fra le varie classi dando feedback e modificando quello necessario e mi sono informato in modo approfondito sul drone tello per poter simulare al meglio tutte le sue funzionalità e movimenti.

Fadil:

Ho implementato il controllo completo del drone usando le estremità della mano per il calcolo dei angoli di rollio, beccheggio e imbardata della mano destra. Usando le estremità e un paio di calcoli matematici l'angolo letto risulta molto più preciso che usando i metodi forniti dall'SDK del LeapMotion.

Ho fatto un paio di test insieme a Jari per controllare il giusto funzionamento della classe DroneController avendo implementato questi nuovi metodi.

Luca:

Ho fatto delle piccole modifiche alla classe SettingManager ed ho implementato il pannello per gestire i config dalla GUI.

Andrea:

Ho lavorato sulla classe TelloChartFrame riducendo il codice necessario per mostrare i grafici di posizione e rotazione.

### Problemi riscontrati e soluzioni adottate

Un calcolo preciso dei angoli della mano destra

Soluzione trovata su due siti:

- <https://stackoverflow.com/questions/2676719/Calculating-the-angle-between-the-line-defined-by-two-points>
- <https://math.stackexchange.com/questions/1201337/finding-the-angle-between-two-points>

Codice:

```
Math.toDegrees(Math.atan2(Y1 - Y2, X1 - X2));
```

## Punto della situazione rispetto alla pianificazione

Siamo al passo con il gantt preventivo.

### Programma di massima per la prossima giornata di lavoro

-

# PROGETTO Drone Control System |

## Diario di lavoro - 05.04.2019

Luca Di Bello, Fadil Smajilbasic, Andrea Rauso, Jari Näser

**Canobbio, 05.04.2019**

### Lavori svolti

Jari:

Ho continuato a raffinare e rendere il più reale possibile il funzionamento di vari metodi del CommandReader come ad esempio il getBattery che si basa sul tempo di volo del drone e sulla capacità reale della batteria.

Inoltre ho cambiato la struttura di CommadReader e Simulator e sto testando i vari metodi di CommandReader uno per uno.

Fadil:

Ho iniziato a scrivere i commenti javadoc di tutti i metodi nella classe DroneController e ho commentato tutti i metdoi nella classe FrameHelper.

Luca:

Ho finito l'implementazione del pannello per la modifica delle impostazioni nella GUI.

Andrea:

Ho lavorato sulla classe TelloChartFrame modificando la struttura dei metodi e commentando i metodi che non avevano ancora una descrizione del loro funzionamento. Ho modificato completamente l'UML relativo alla simulazione del drone aggiornandolo con l'attuale struttura delle classi.

### Problemi riscontrati e soluzioni adottate

### Punto della situazione rispetto alla pianificazione

Siamo al passo con il gantt preventivo.

### Programma di massima per la prossima giornata di lavoro

Luca:

Implementare log nella GUI



# PROGETTO Drone Control System |

## Diario di lavoro - 10.04.2019

Luca Di Bello, Fadil Smajilbasic, Andrea Rauso, Jari Näser

**Canobbio, 10.04.2019**

### Lavori svolti

Jari:

Ho fatto una parte del diagramma UML di DroneController con Fadil e ho continuato a testare i vari metodi della classe CommandReader.

Fadil:

Ho aggiornato l'UML della parte del DroneController e ho fatto refactor del codice scritto sia da me che da luca.

Luca:

Problemi con le librerie di Leap Motion. Risolto con l'aiuto di Fadil.

Andrea:

Ho corretto l'UML del DroneSimulator aggiungendo classi, metodi e/o attributi mancanti.

### Problemi riscontrati e soluzioni adottate

### Punto della situazione rispetto alla pianificazione

Siamo al passo con il gantt preventivo.

### Programma di massima per la prossima giornata di lavoro

-

# PROGETTO Drone Control System |

## Diario di lavoro - 12.04.2019

Luca Di Bello, Fadil Smajilbasic, Andrea Rauso, Jari Näser

**Canobbio, 12.04.2019**

### Lavori svolti

Jari:

Ho continuato a testare i vari metodi della classe CommandReader e a sistemare lì dove necessario, inoltre ho anche fatto la JavaDoc per le seguenti classi: Simulator, CommandReader, BatteryThread e PacketReceivingCheckerThread.

Fadil:

Ho messo assieme la classe DroneController e DroneControllerMonitor, ho inserito dei listener per far comunicare le due classi, ho fatto del testing

Luca:

Oggi ho testato il controller dell'applicazione e mi sono accorto che a lungo andare inizia ad utilizzare troppe risorse (il computer scalda e rallenta)

Andrea:

Ho aggiornato l'UML della parte DroneSimulator e ho rimosso classi non più in uso

### Problemi riscontrati e soluzioni adottate

- 

### Punto della situazione rispetto alla pianificazione

Siamo al passo con il gantt preventivo.

### Programma di massima per la prossima giornata di lavoro

Fadil:

Aggiungere l'autoscroll alla pagina di log nell'interfaccia grafica



# PROGETTO Drone Control System |

## Diario di lavoro - 17.04.2019

Luca Di Bello, Fadil Smajilbasic, Andrea Rauso, Jari Näser

**Canobbio, 17.04.2019**

### Lavori svolti

Jari:

Assieme al resto del team ho testato per la prima volta il drone Tello e abbiamo raffinato e corretto in parte i comandi.

Fadil:

Avendo ricevuto il drone ho iniziato a lavorare sul fine-tuning dei comandi per una esperienza migliore durante l'uso del LeapMotion e ho fatto vari test per verificare il giusto funzionamento del drone.

Luca:

Ho continuato a fare la GUI del controller del drone utile per poterlo controllare senza l'utilizzo del LeapMotion.

Andrea:

Ho creato il gantt consuntivo del progetto e ho aggiornato la javadoc della classe TelloChartFrame per rispettare le convenzioni

### Problemi riscontrati e soluzioni adottate

.

### Punto della situazione rispetto alla pianificazione

Siamo al passo con il gantt preventivo.

### Programma di massima per la prossima giornata di lavoro

Fadil:

Aggiungere l'autoscroll alla pagina di log nell'interfaccia grafica

# PROGETTO Drone Control System | Diario di lavoro - 03.05.2019

Luca Di Bello, Fadil Smajilbasic, Andrea Rauso, Jari Näser

## Canobbio, 03.05.2019

### Lavori svolti

Jari:

Ho creato il diagramma delle classi del progetto DroneSimulator e ho continuato con la documentazione.

Fadil:

Ho modificato il modo in quale il controller calcola l'intervallo di tempo tra un comando e l'altro, inoltre ho fatto altri test con il drone tutti con successo. L'integrazione tra il leapMotion e il drone è più che ottimale.

Luca:

Ho continuato la documentazione terminando il capitolo 'analisi e specifica dei requisiti'. Ho anche creato le classi 'FlightRecorder' e 'FlightBuffer'.

Andrea:

Ho continuato la documentazione aggiungendo l'implementazione delle classi di DroneSimulator e l'aggiunta dei mockup delle interfacce.

### Problemi riscontrati e soluzioni adottate

Problema con l'aggiunta dell'autoscroll al pannello che segna i comandi mandati:

Codice usato per testare:

```
DefaultCaret caret = (DefaultCaret) logTextArea.getCaret(); caret.setUpdatePolicy(DefaultCaret.ALWAYS_UPDATE);
```

Non ha funzionato ma cercherò delle altre soluzioni

### Punto della situazione rispetto alla pianificazione

Siamo al passo con il gantt preventivo.

### Programma di massima per la prossima giornata di lavoro

# **PROGETTO Drone Control System |**

## **Diario di lavoro - 08.05.2019**

**Luca Di Bello, Fadil Smajilbasic, Andrea Rauso, Jari Näser**

**Canobbio, 08.05.2019**

### **Lavori svolti**

Jari:

Ho continuato con lo sviluppo e correzione della documentazione.

Fadil:

Ho lavorato con luca sull'implementazione del flight recording e ho aggiunto il riconoscimento delle gesture per il controllo del yaw del drone.

Luca:

Ho continuato l'implementazione delle classi relative al recording dei voli.

Andrea: Ho continuato con la documentazione modificando il sommario trasformandolo in una lista numerata, ho aggiunto e aggiornato le immagini degli UML e i siti della libreria JFreeChart nella sitografia.

### **Problemi riscontrati e soluzioni adottate**

### **Punto della situazione rispetto alla pianificazione**

Siamo al passo con il gantt preventivo.

### **Programma di massima per la prossima giornata di lavoro**

Chiedere se nei componenti dell'Hardware bisogna anche mettere i componenti specifici dei vari PC.

# **PROGETTO Drone Control System |**

## **Diario di lavoro - 10.05.2019**

**Luca Di Bello, Fadil Smajilbasic, Andrea Rauso, Jari Näser**

**Canobbio, 10.05.2019**

### **Lavori svolti**

Jari:

Ho continuato con lo sviluppo e correzione della documentazione.

Inoltre come dovuto ho richiesto il parere del docente Luca Muggiasca sulle specifiche dell'hardware il quale ha detto che non si necessita dell'elenco dei componenti presenti nei vari PC.

Fadil:

Ho aggiunto commenti alla classe DroneController e ho messo apposto il gantt consuntivo.

Luca:

Aggiunto un menù nella GUI utile per la registrazione e l'esecuzione dei voli già registrati.

Andrea:

Ho continuato a lavorare sulla documentazione

### **Problemi riscontrati e soluzioni adottate**

### **Punto della situazione rispetto alla pianificazione**

Siamo al passo con il gantt preventivo.

### **Programma di massima per la prossima giornata di lavoro**