LUMINOUSBEES

# Progetto ArduBee

User Requirements Specification Document

DIBRIS – Università di Genova. Scuola Politecnica, Corso di Transactional systems & data warehouse

**DATA : 16/08/2020**

**VERSION : 1.0**

**Autori**

Matteo Cardano

Stefano Lavaggi

Andrea Palmiere

**REVISION HISTORY**

|  |  |  |  |
| --- | --- | --- | --- |
| Versione | Data | Autori | Note |
| 1.0 | 16/08/2020 | Cardano, Lavaggi,Palmiere | Prima stesura |
|  |  |  |  |

**Indice dei Contenuti**

1. [Introduzione 3](#_gjdgxs)
   1. [Scopo del Documento 3](#_30j0zll)
   2. Overview del Documento 3
   3. [Definizioni e Acronimi 3](#_1fob9te)
2. [Descrizione Generale del Sistema 4](#_tyjcwt)
   1. [Contesto 4](#_3dy6vkm)
   2. Motivazioni 4
   3. [Obiettivo del Progetto](#_1t3h5sf) 5
   4. Stakeholders 6
3. [User Requirement](#_2s8eyo1) 6

# Introduzione

## Scopo del Documento

Analisi della problematica cliente. Il documento è necessario affinché le componenti richieste del cliente (LuminousBee) vengano rispettate dai progettisti.

## Overview del Documento

Nel documento è descritto il problema, le sue caratteristiche e alcune considerazioni tecniche. Viene fornita inoltre una descrizione della struttura generale.

## Definizioni e Acronimi

|  |  |
| --- | --- |
| **Acronimo-Nome** | **Definizione** |
| AP | ArduPilot |
| MV | Mavlink |
| PM | pyMavlink |
|  |  |

# Descrizione Generale del Sistema

Il progetto ha l’obiettivo di sviluppare un’interfaccia di programmazione a blocchi per permettere ad un utente inesperto la possibilità di effettuare le operazioni di base con il drone, fornendo un layer di astrazione superiore, in particolare tramite l’interconnessione dei vari blocchi sarà possibile effettuare un vero e proprio piano di volo scrivendo un programma attraverso un facile interfaccia grafica a blocchi

## Contesto

Il progetto è legato ad un contesto operativo. Attraverso il particolare hardware sviluppato dall’azienda e il software in modo da garantire un ottima integrazione e un facile utilizzo tramite un’interfaccia grafica che permette l’utilizzo della programmazione a blocchi

## Motivazioni

Attualmente è disponibile solamente una ground station che permette l’interazione da riga di comando.

## Obiettivo del Progetto

In seguito ai colloqui introduttivi abbiamo valutato quali istruzioni è implementare attraverso PM, e il livello delle istruzioni (medio-alto) mentre blockly viene utilizzato per gestire l’interfaccia ( nel progetto si fa riferimento all’utilizzo del codice python , javaScript e al protocollo MV).

## Stakeholders

* Programmatore piano di volo

# User Requirement

|  |  |
| --- | --- |
| **PRIORITA’** | **SIGNIFICATO** |
| **M** | Mandatory. Requisito obbligatorio. |
| **D** | Desiderabile. Requisito che dovrebbe essere inserito nel sistema, a meno che il costo per implementarla non sia troppo alto. |
| **O** | Optional. Una funzionalità marcata con O può essere inserita nel sistema, a discrezione del manager del progetto. Ad esempio se il tempo di sviluppo è minore di quello previsto oppure se il costo per implementarla non è troppo alto. |
| **E** | future Enhancement. Questo requisito viene lasciato per la prossima release. |

|  |  |  |
| --- | --- | --- |
| **ID** | **DESCRIZIONE** | **PRIORITA’** |
| 1 | Avviamento motori | M |
| 2 | Spegnimento motori | M |
| 3 | Take-off | M |
| 4 | Return to land | M |
| 5 | Spostamento attraverso coordinate gps | M |
| 6 | Spostamento con riferimento NED | D |
| 9 | Rotazioni | D |
| 10 | Cambio modalità di guida | M |