

Drone programmer Documentation :

0 Prérequis :

Afin de travailler sur cette application il est nécessaire d'être équipé comme suit :

- Un ordinateur avec MacOS
- xCode
- Un drone Parrot bebop 2
- Cloner le repository github
- Vérifier que [ces champs](#) soient remplis correctement

1 But de l'application :

Le but de l'application droneProgrammer est de pouvoir piloter un drone bebop2 de la marque Parrot.

Il y'a plusieurs moyens de piloter le drone :

- 1) Un pilotage *live* où le drone réagit au divers bouton pressé. Ce mode sera appelé « Free flight » dans le reste de cette documentation ainsi que dans le code de l'application.
- 2) Un pilotage programmé, où il est possible de programmer le plan de vol du drone. Dans ce mode, il faut entrer une suite de commande ainsi que la position des obstacles et des objectifs. Il est ensuite possible de simuler le vol afin de vérifier si le drone touche tous les objectifs et aucun obstacle avant de le lancer.

2 Fonctionnement globale de l'application :

- 1) Photo storyboard
- 2) Le storyboard de l'application est le suivant : On commence par une vue initiale, c'est un menu où il y'a trois options.
 - a. « Free flight » Le pilotage live avec la visualisation de la caméra du drone et la possibilité de prendre des photos.
 - b. « Gestionnaire » On y trouve une liste de plan de vol sauvegardée depuis la planification de vol. Lorsqu'on choisit un plan de vol on arrive directement au mode créateur avec ce plan de vol.
 - c. « Plan de Vol » qui correspond à un mode création de plan de vol. Il possible d'ajouter les commandes suivantes :
 - Décoller
 - Atterrir
 - Aller à droite
 - Aller à Gauche
 - Avancer
 - Reculer
 - Monter
 - Descendre

Il est également possible d'ajouter des obstacles et des objectifs qui sont représentés en coordonnées (x,y,z). Une fois que le plan de vol est créé il est possible de le simuler. Si la simulation réussie, (Tous les objectifs touchés et aucun obstacles percutés) il est alors possible de se connecter au drone et de le lancer.

3 Les technologies utilisées :

- Swift 4.0 -> Langage de programmation de l'application
- Json -> Encodage des plans de vols
- SceneKit -> Librairie 3D pour la simulation
- Objective-C -> Langage de programmation pour les *samples* du drone.

Il est conseillé d'avoir de bonnes connaissances dans les trois premières technologies, la dernière n'est que peu utilisée. Cependant, pour aller plus loin, il est possible d'utiliser directement l'API du drone, qui est écrite en objective-C et qui permet d'aller plus loin que les *samples*.

4 Liens divers :

- [Documentation librairie bebop 2](#)
- [Utilisation des samples](#)
- [Json documentation](#)
- [Swift documentation](#)
- [SceneKit documentation](#)

5 Contacts :

Professeur : Didier Buchs, didier.buchs@unige.ch

Assistants : Damien Morard, damien.morard@unige.ch

Dimitri Raccordon, dimitri.raccordon@unige.ch

Étudiants : Mathias Tonini, mathias.tonini@etu.unige.ch, mathias.tonini@gmail.com

Deniz Sungurtekin, deniz.sungurtekin@etu.unige.ch, denizsungurtekin@gmail.com