

3I013 Soutenance 6 Mai 2019

Nicolas CASTANET
Maël FRANCESCHETTI
Daoud KADOCH
Fabien MANSON

Sommaire

La Demande du Client

Scénario d'Utilisation

Architecture Matérielle

Architecture Logicielle

Test Effectués

Déploiement

Problèmes rencontrés

Répartition du travail

Conclusion

La Demande du Client

- ▶ Le client souhaite effectuer des rondes avec un drone Bebop 2
- ▶ Le drone doit voler de manière autonome en suivant un plan de vol prédéfini
- ▶ Le retour vidéo du drone doit être redirigé à un iPod touch qui sera placé dans un masque FPV pour permettre à l'utilisateur de voir comme s'il était à la place du drone

Scénario d'utilisation

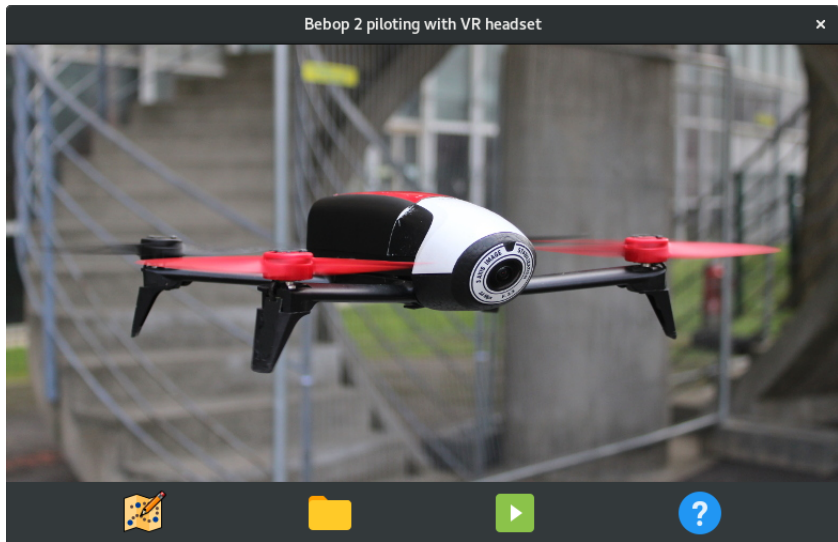
1. Démarrage du drone
2. Lancement de l'application
3. Saisie du plan de vol
4. Préparation du vol
5. Sélection du plan de vol
6. Décollage du drone
7. Fin du vol

Scénario : 1 - Démarrage du drone

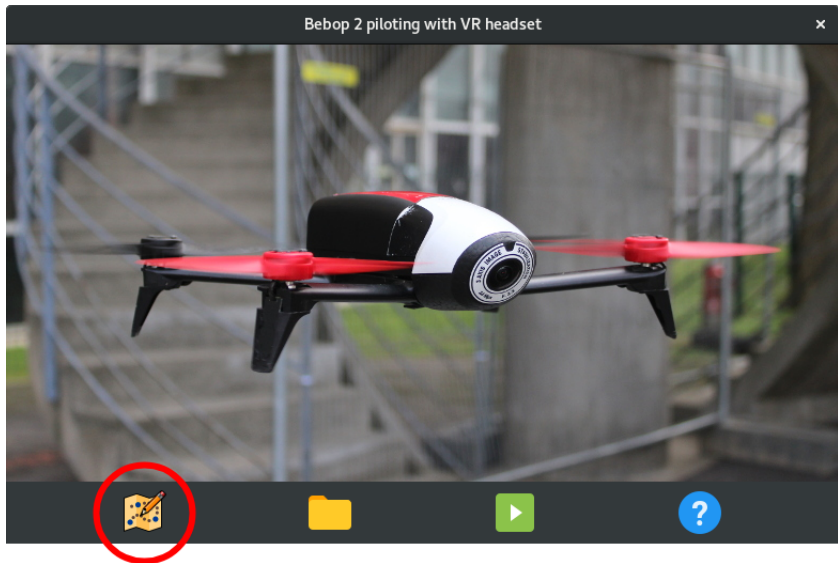
Le drone doit être allumé et posé sur un endroit stable et adapté à un décollage.

Il faut patienter quelques instants avant que le réseau wifi du drone ne soit actif.

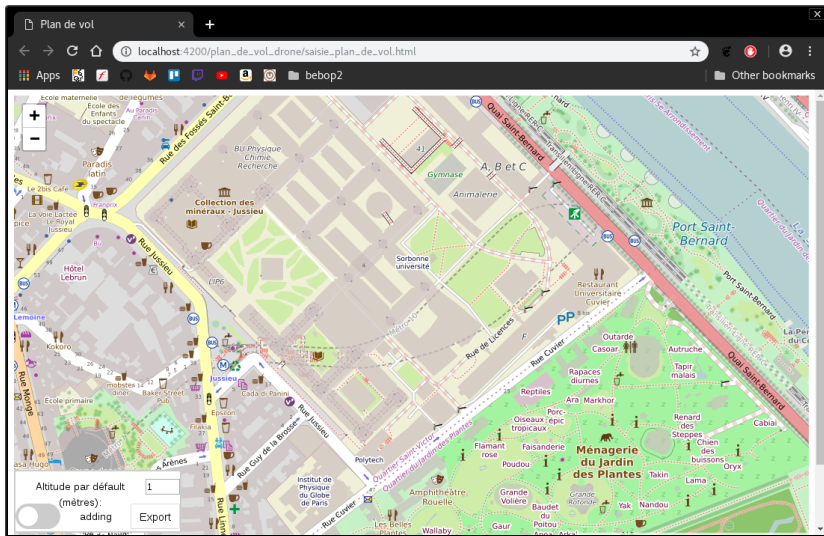
Scénario : 2 - lancement de l'application



Scénario : 3 - Saisie d'un plan de vol



Scénario : 3 - Saisie d'un plan de vol



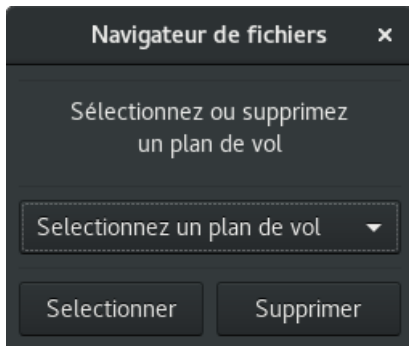
Scénario : 4 - Préparation du vol

1. Connexion du PC au wifi du drone
2. Connexion de l'iPod au réseau local et démarrage de l'application sur l'iPod
3. Mise en place de l'iPod dans le masque FPV

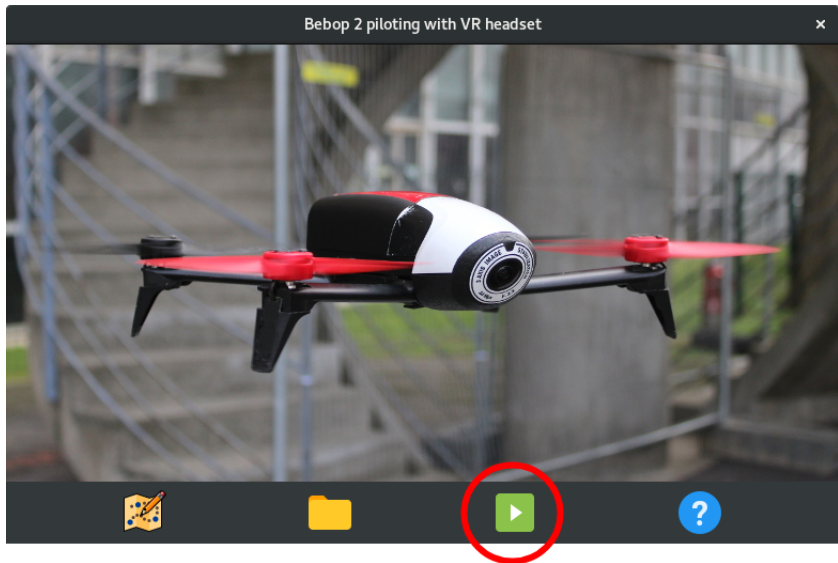
Scénario : 5 - Choix du plan de vol



Scénario : 5 - Choix du plan de vol



Scénario : 6 - Décollage du drone



Scénario : 6 - Décollage du drone

L'utilisateur peut enfiler le masque FPV et ensuite secouer la tête pour faire décoller le drone.



Scénario : 7 - Fin du vol

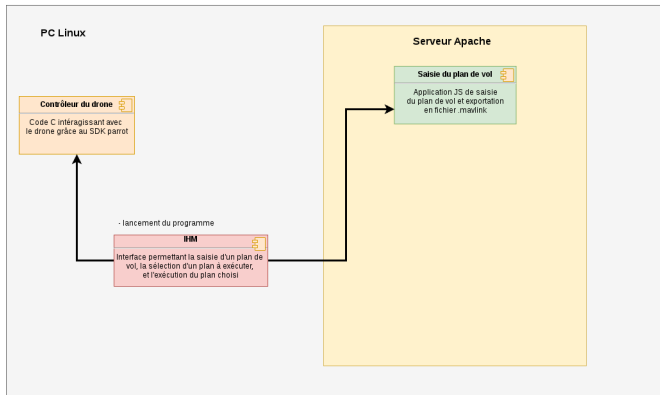
L'utilisateur peut enclencher un arrêt d'urgence en secouant la tête ou bien attendre que le drone ait fini le vol et atterrisse de lui-même.



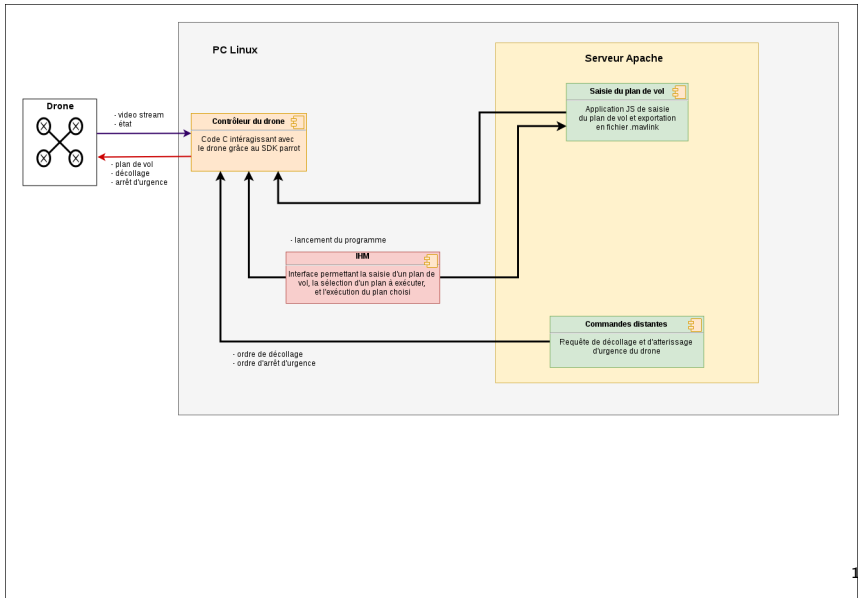
Architecture Matérielle



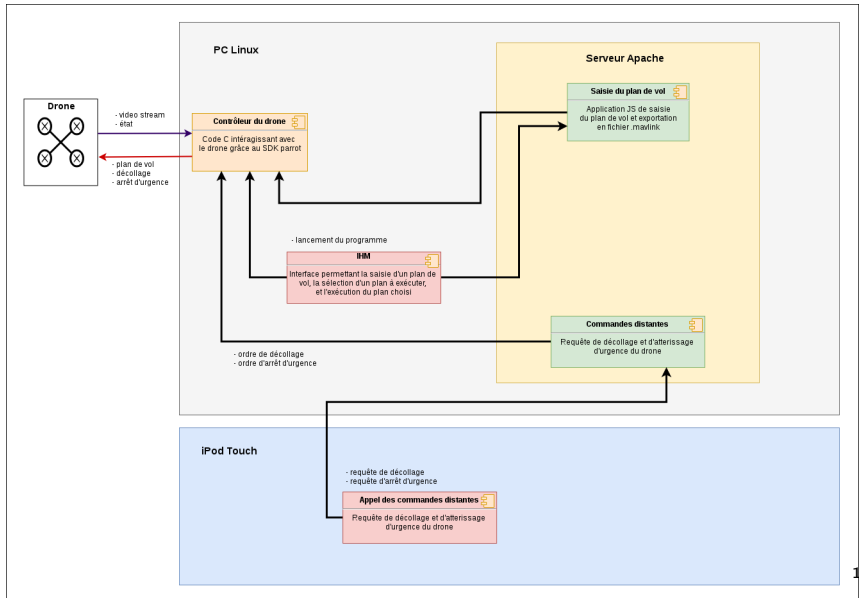
Architecture Logicielle : interface utilisateur



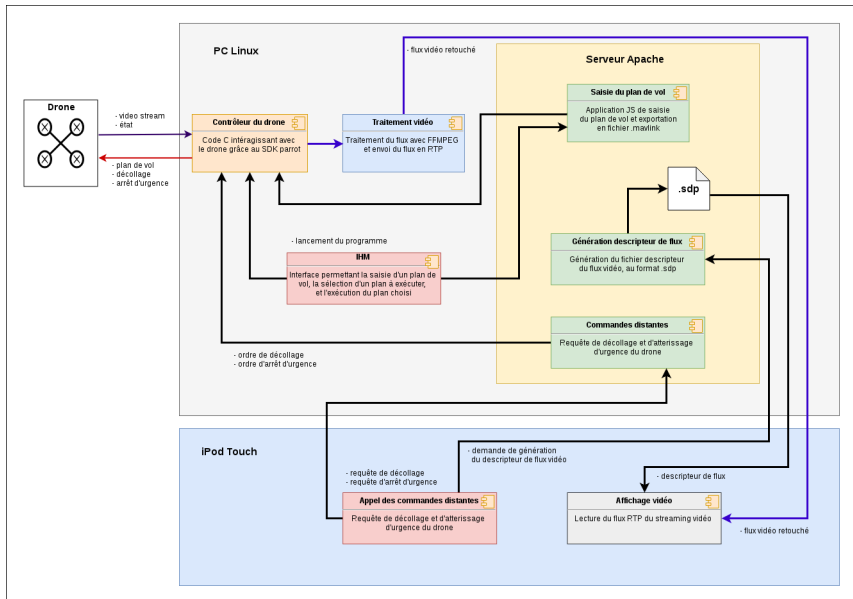
Architecture Logicielle : controle du drone



Architecture Logicielle : contrôle avec l'iPod



Architecture Logicielle : retour vidéo sur l'iPod



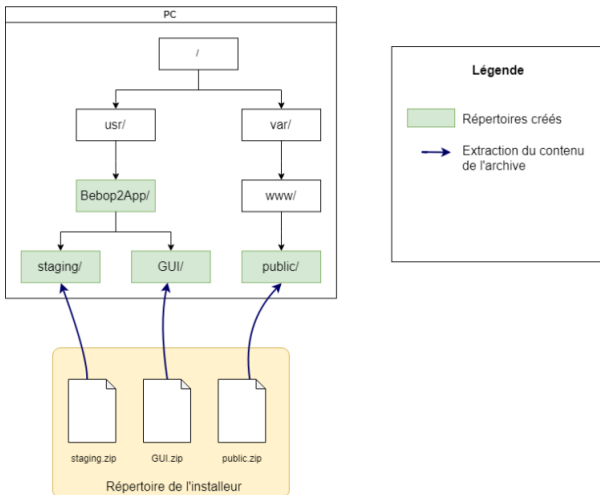
Tests effectués

- ▶ Saisie d'un plan de vol avec la fonctionnalité dédiée
- ▶ Démarrage du drone par un mouvement de tête avec le masque FPV contenant l'iPod
- ▶ Exécution du plan de vol choisi en totalité
- ▶ Arrêt d'urgence par un mouvement de tête avec le masque FPV contenant l'iPod
- ▶ Affichage de la vidéo du drone sur l'iPod en temps réel, avec un traitement pour la vue FPV (une image par oeil)

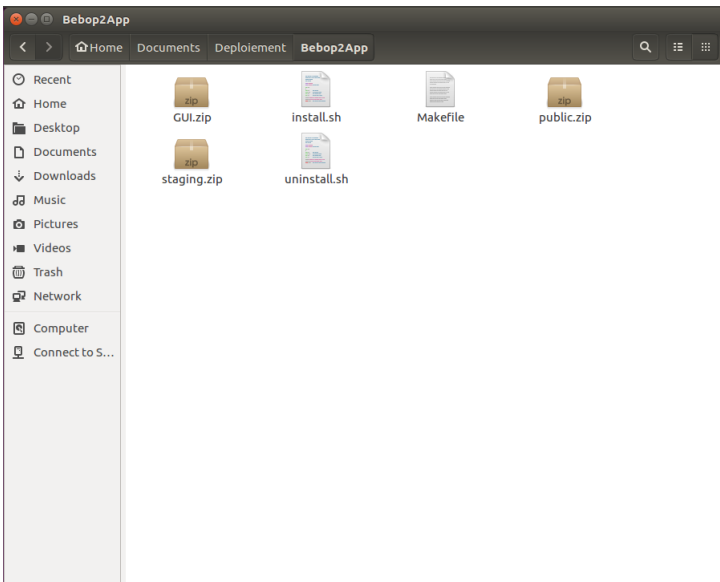
Déploiement

- ▶ L'archive contient un script Makefile permettant d'extraire au bon endroit les différents composants de l'application.
- ▶ Les outils requis tels que GTK+ 3.0, un serveur Lamp et FFMPEG devront être installés par l'utilisateur au préalable.
- ▶ Le serveur devra être paramétré pour donner accès au répertoire "public".

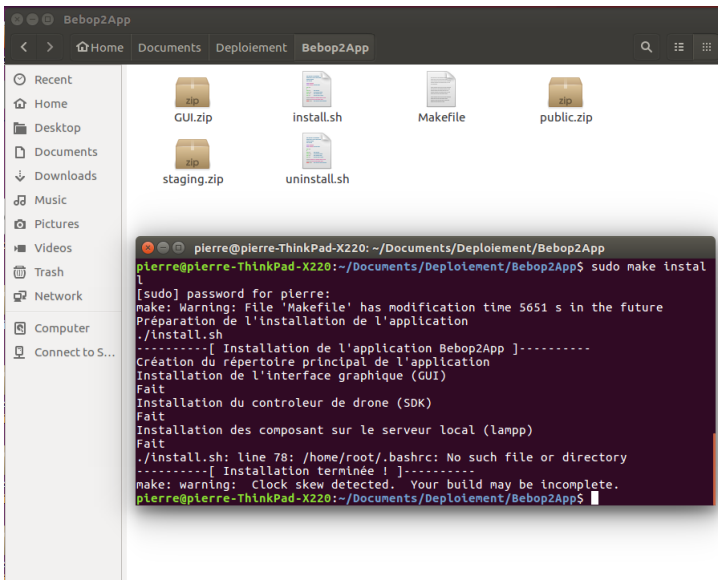
Déploiement : action du Makefile



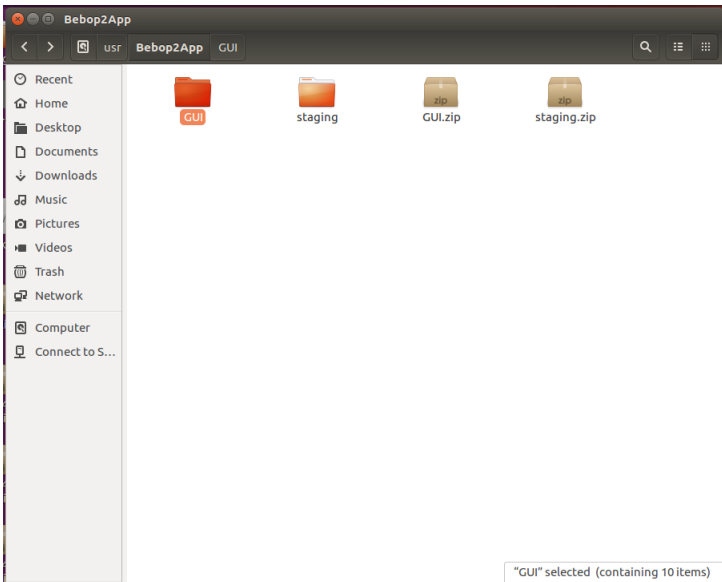
Déploiement : contenu de l'archive



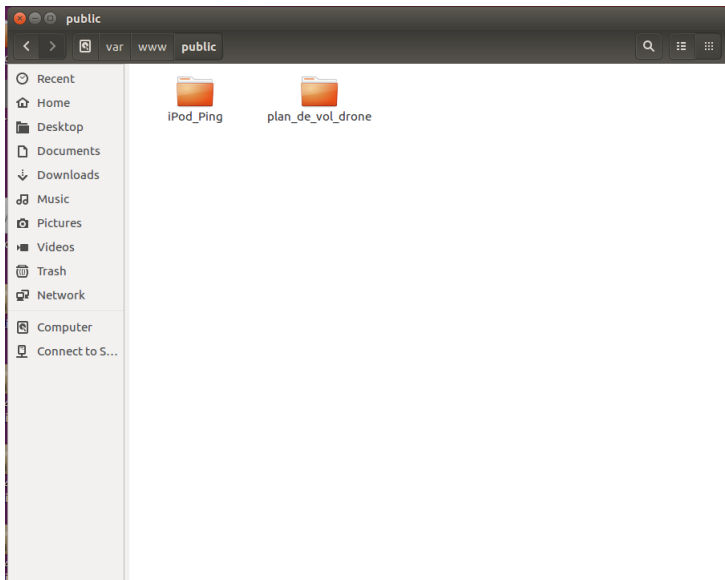
Déploiement : exécution du Make



Déploiement : résultat côté controle du drone et IHM



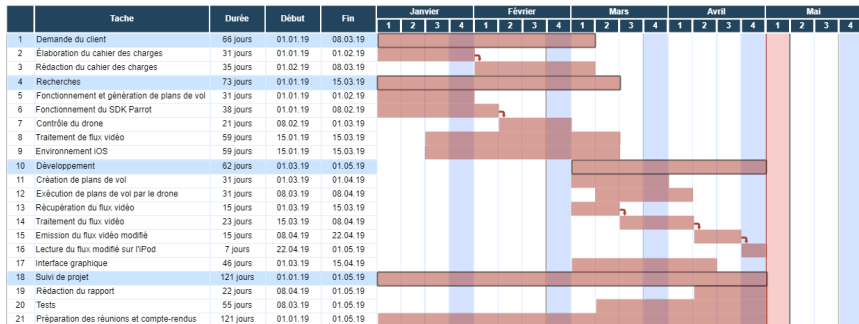
Déploiement : résultat côté serveur



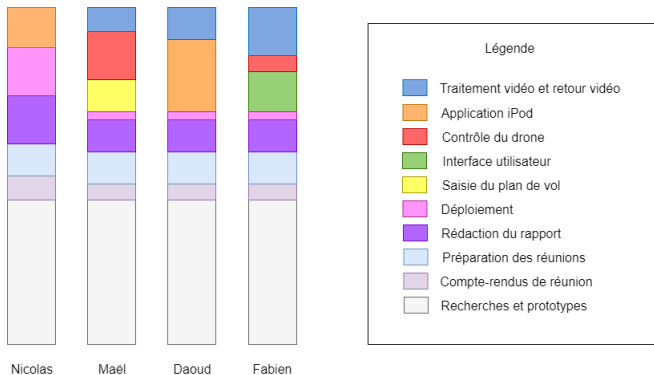
Problèmes rencontrés

1. Calibration du drone après chaque arrêt d'urgence (choc ou inclinaison trop forte du drone)
2. Transmission du flux vidéo à l'iPod
3. Perte du signal GPS sur campus de l'UPMC

Diagramme de Gantt



Répartition des tâches



Conclusion