

Projet Développement : Easy GNSS

Développement d'un logiciel pour GNSS low-cost



Nassim CHEBBAH (Chef de projet)
Arthur FAVREAU
Edgar LENHOF
Sanam MONANY

SOMMAIRE

- ▶ I : Contexte du projet
- ▶ II : Objectifs & Besoins
- ▶ III : Etude technique
- ▶ IV : Réalisation & Suivi
- ▶ V : Bilan du projet



I – Contexte du projet

- ▶ L'apparition de la technologie Raspberry rend possible la création d'un GNSS précis et à moindre coût en comparaison aux appareils actuellement sur le marché.
- ▶ Une vaste communauté voit un grand intérêt dans ce GNSS low-cost, et s'implique donc dans sa création. L'utilisation de données Open Source est privilégiée.
- ▶ Nos commanditaires sont impliqués dans le projet depuis plus de 3 ans, et font régulièrement participer des élèves de l'ENSG aux différentes étapes du projet.

- ▶ Jean-Yves Perrin :
Gendarmerie Nationale, passionné de drones.
- ▶ Francklin N'Guyen Van :
Intermittent à France Télévisions, passionné de drones et de programmation informatique.
- ▶ Jacques Beilin :
Enseignant-chercheur en géodésie au DPTS.
Notre professeur référent à l'ENSG.

► Acteurs :

- Commanditaires
- ENSG (Elèves, anciens élèves et enseignants)
- Collaborateurs extérieurs
- Communauté intéressée par le projet

► Besoins :

- Utilisation étendue à un public large, si possible à l'international
- Conception avec un budget limité et du contenu Open Source
- Précision se rapprochant le plus possible d'un GNSS du marché

- ▶ Librairie open-source proposant un ensemble de programmes pour le positionnement standard et précis par GPS. Elle est développée et mise à disposition gratuitement par l'université de Tokyo
- ▶ Avantages :
 - > Facilité d'utilisation
 - > Paramétrisation assez complète
 - > Nombreux champs d'application
- ▶ Nous utilisons ici la version 2.4.3 du logiciel, qui ne possède pas encore de manuel d'utilisation. Nous utilisons donc celui de la version la version 2.4.2

- ▶ On dispose de deux softwares permettant de faire du GNSS en modes Base et Rover :
 - RTKBase : Réalisé en C par d'anciens élèves de l'ENSG. RTKLib est compilé directement dans le code
 - TouchRTKStation : Réalisé en Python par M. Taro Suzuki, un universitaire japonais. RTKLib est compilé en dehors du code
- ▶ On dispose également de 4 hardware compatibles avec les 2 softwares





II – Objectifs & Besoins

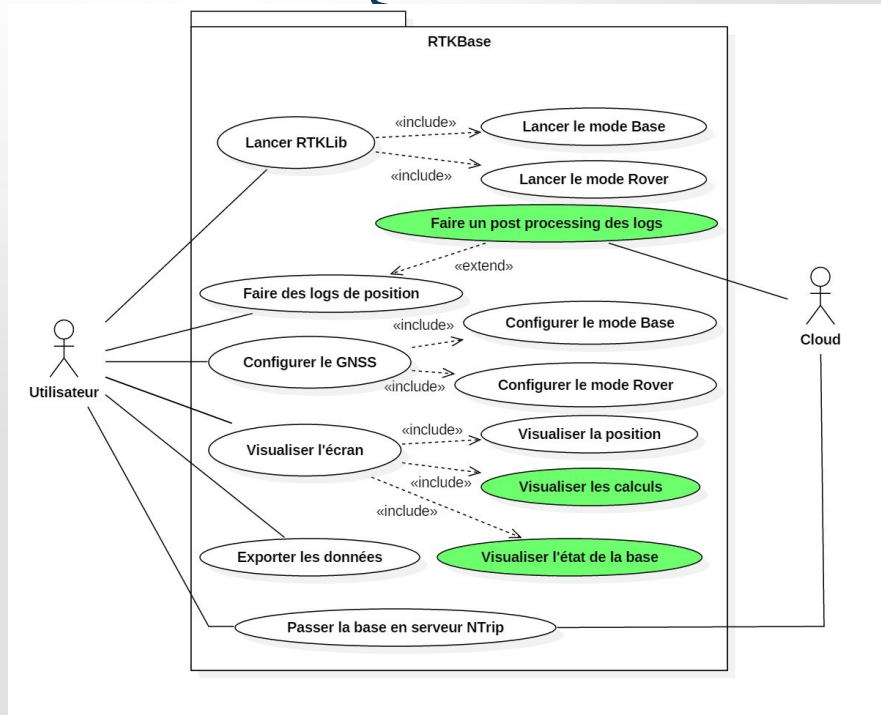
- ▶ Réécrire le code de RTKBase en Python
- ▶ Ajouter des fonctionnalités à TouchRTKStation
- ▶ Rendre le code simple à comprendre et à reprendre
- ▶ Fournir une documentation adaptée

- ▶ Ajouter des fonctions de post-processing (**Priorité 0**)
- ▶ Ajouter tous les modes de calcul de RTKBase (**Priorité 0**)
- ▶ Ajouter de nouveaux paramètres dans RTKBase (**Priorité 0**)
- ▶ Rendre le code compatible avec l'écran 5 pouces de RTKBase et le 4 pouces de TouchRTKStation (**Priorité 1**)
- ▶ Sur le Rover : Configurer automatiquement la position de la base (**Priorité 1**)
- ▶ Rendre l'affichage de l'interface plus ergonomique (**Priorité 2**)

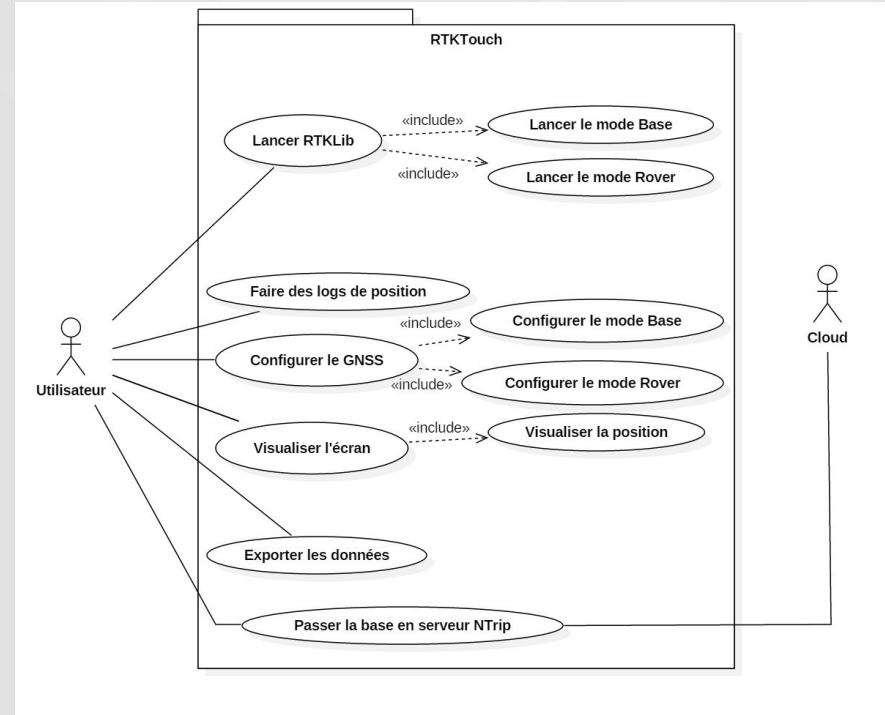
OBJECTIFS & BESOINS

Diagrammes de cas d'utilisation

EasyGNS



TouchRTKStation



- ▶ Faire un fork du code de TouchRTKStation sur GitHub
- ▶ Utiliser exclusivement le langage Python et l'anglais
- ▶ Travailler en priorité avec des outils Open Source
- ▶ Faire un code clair et bien commenté
- ▶ Utiliser des noms de variables adaptés

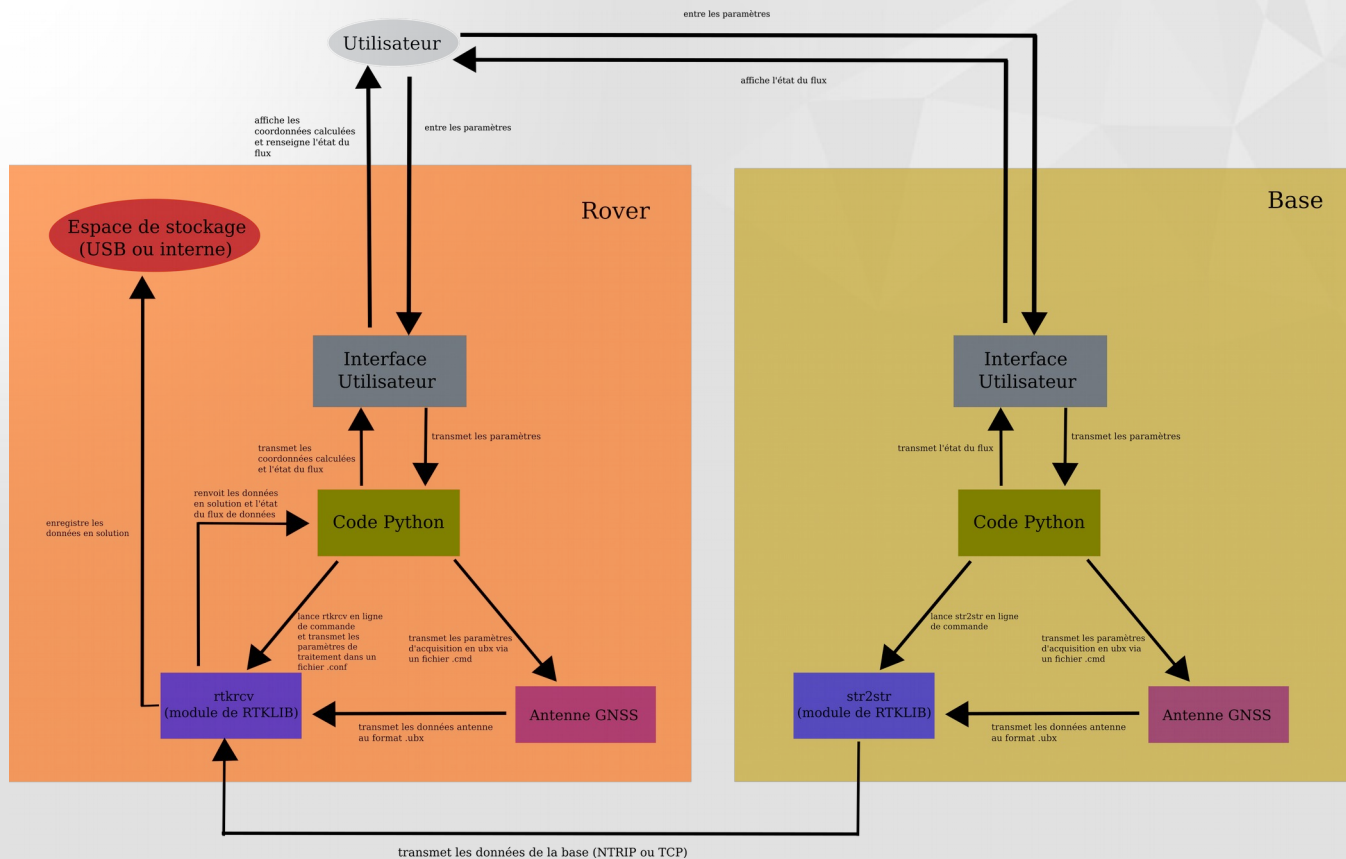
- ▶ Faire vivre le projet démarré il y a déjà 3 ans
- ▶ Fournir un outil prêt à l'emploi à la communauté
- ▶ Montrer l'intérêt de l'école pour ce projet
- ▶ Permettre aux ING1 de tester notre GNSS low-cost pendant le stage à Forcalquier
- ▶ Maintenir un partenariat durable entre l'ENSG et les commanditaires



III – Etude technique

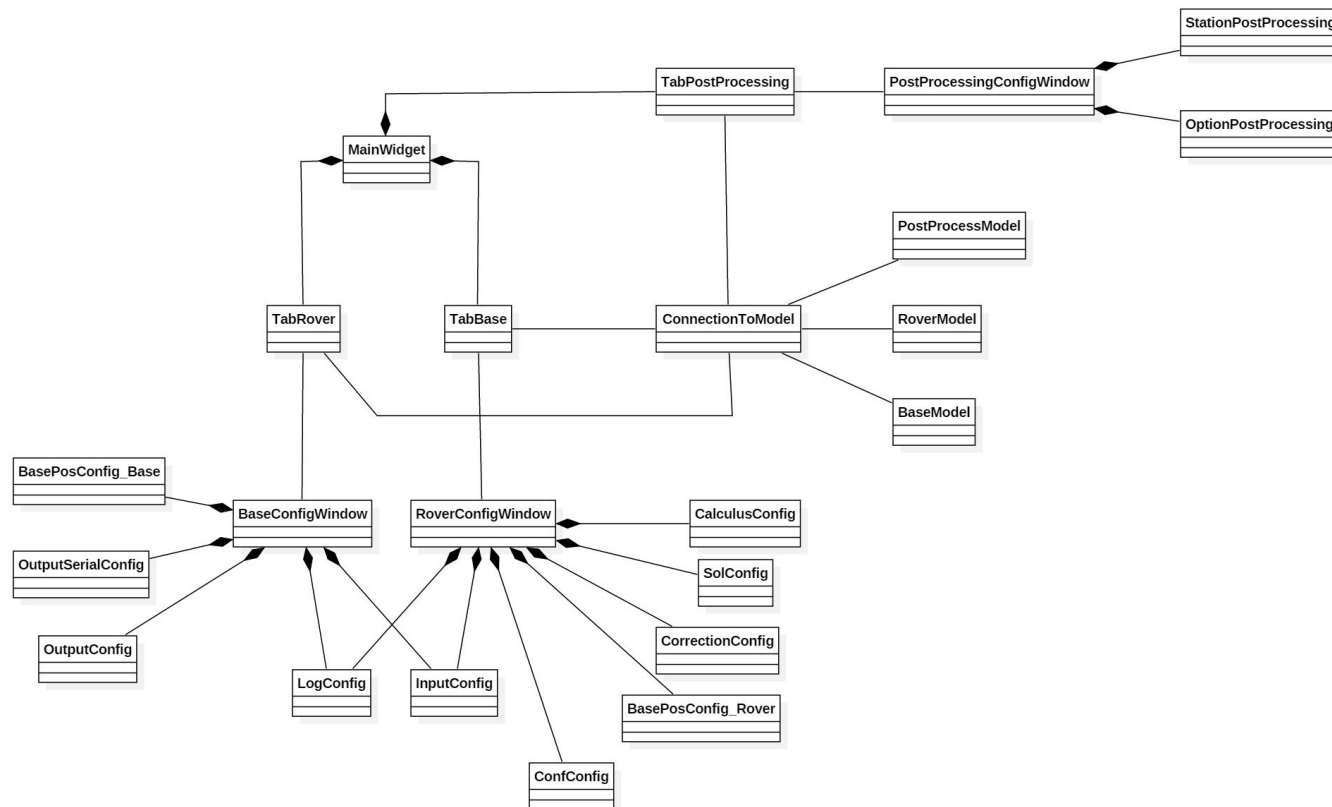
ETUDE TECHNIQUE

Diagramme de déploiement



ETUDE TECHNIQUE

Diagramme de classes





IV – Réalisation & Suivi

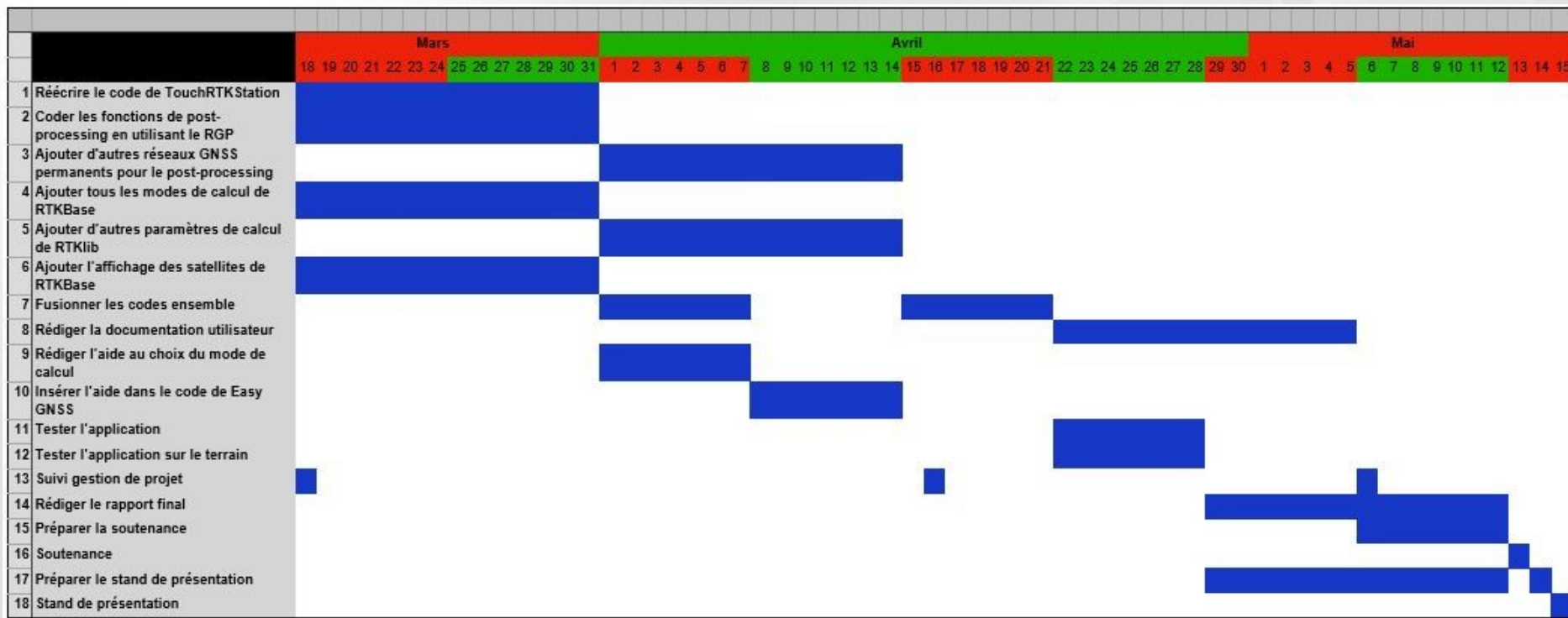
- ▶ Perte ou dégradation de matériel
(Gravité : **Majeure** ; Probabilité : **Faible**)
- ▶ Perte de données
(Gravité : **Modérée** ; Probabilité : **Faible**)
- ▶ Problèmes de communication avec les commanditaires
(Gravité : **Mineure** ; Probabilité : **Moyenne**)
- ▶ Conflits avec les commanditaires
(Gravité : **Modérée** ; Probabilité : **Faible**)
- ▶ Conflits dans l'équipe (Gravité : **Mineure** ; Probabilité : **Faible**)

- ▶ Perte ou dégradation de matériel
(Gravité : **Majeure** ; Probabilité : **Faible**)
- ▶ Perte de données
(Gravité : **Modérée** ; Probabilité : **Faible**)
- ▶ Problèmes de communication avec les commanditaires
(Gravité : **Mineure** ; Probabilité : **Moyenne**)
- ▶ Conflits avec les commanditaires
(Gravité : **Modérée** ; Probabilité : **Faible**)
- ▶ Conflits dans l'équipe (Gravité : **Mineure** ; Probabilité : **Faible**)

- ▶ Perte ou dégradation de matériel
(Gravité : **Majeure** ; Probabilité : **Faible**)
Problème : GNSS qui ne démarre pas
Solution : Reformatage de la Raspberry Pi
- ▶ Problèmes de communication avec les commanditaires
(Gravité : **Mineure** ; Probabilité : **Moyenne**)
Problème : Commanditaires pas toujours disponibles
Solution : Entretiens planifiés à l'avance
- ▶ Conflits dans l'équipe (Gravité : **Mineure** ; Probabilité : **Faible**)
Solution : Le chef de projet a le dernier mot

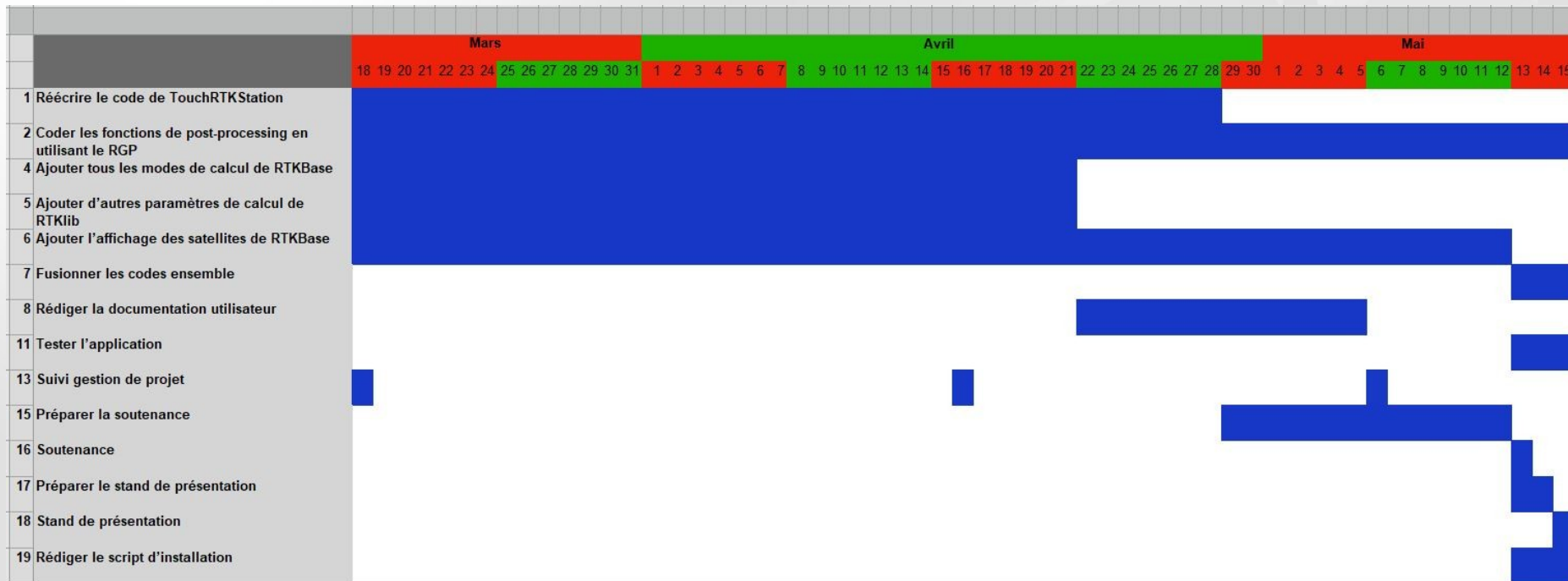
REALISATION & SUIVI

Diagramme de GANTT initial



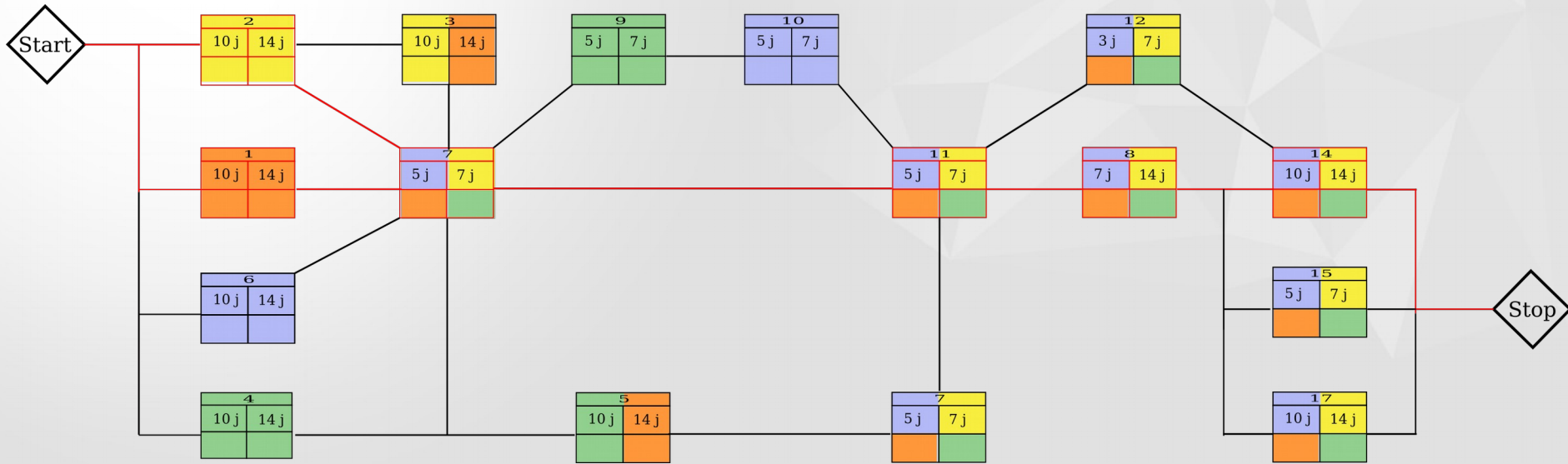
REALISATION & SUIVI

Diagramme de GANTT final



REALISATION & SUIVI

Diagramme de PERT initial



► Nassim : Jaune

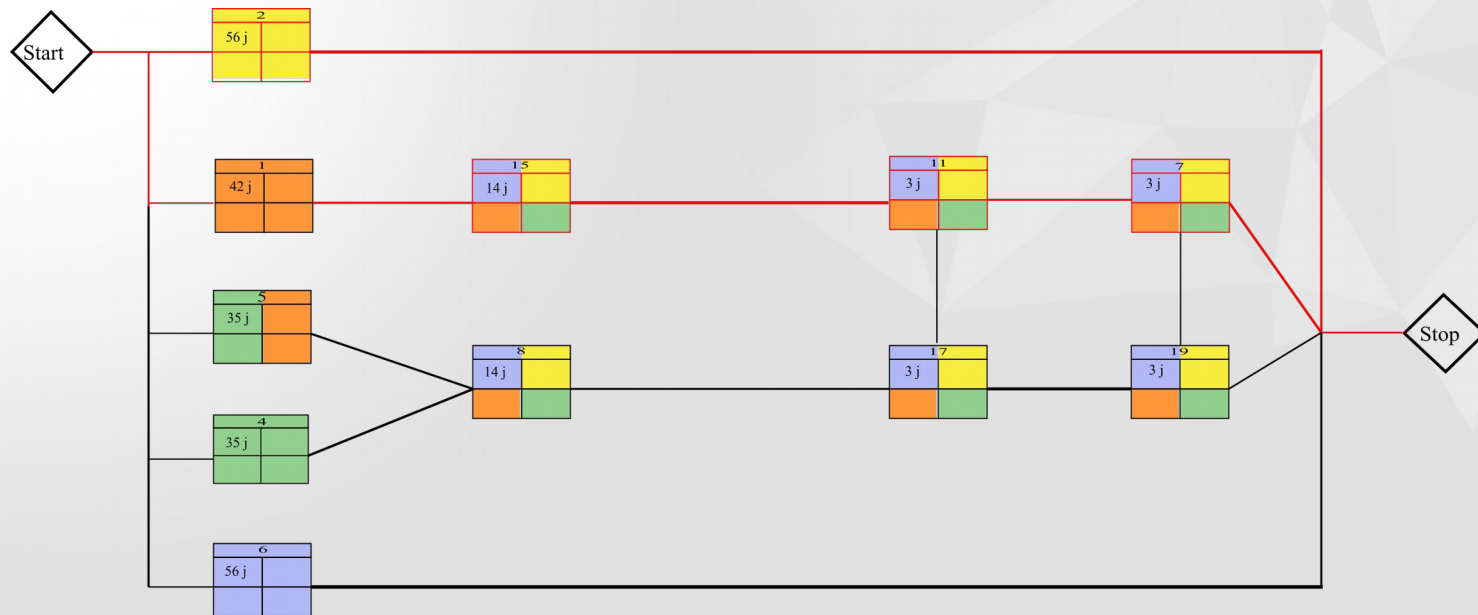
► Edgar : Orange

► Arthur : Vert

► Sanam : Violet

REALISATION & SUIVI

Diagramme de PERT final



► Nassim : Jaune

► Edgar : Orange

► Arthur : Vert

► Sanam : Violet

- ▶ Il n'y a pas de manuel d'utilisation de la version 2.4.3 de RTKLib
- ▶ Le refactoring du code a été compliqué en raison de la difficulté de reprise du code
- ▶ Nous avons eu d'autres projets en parallèle, ainsi que les préparatifs pour le stage à réaliser

- ▶ Meilleure organisation
 - > Optimisation de la répartition des tâches
 - > Planification de la durée de chaque tâche
- ▶ Meilleure appréciation du projet
 - > Vision détaillée du projet au moyen de diagrammes
 - > Connaissance des risques pour mieux les anticiper

- ▶ Entretiens téléphoniques et mails
- ▶ Discussions régulières avec le professeur référent
- ▶ Planification des entretiens téléphoniques à l'avance
- ▶ Compte-rendu réalisé après chaque entretien

► Besoin (Phase d'analyse) :

- > Rencontre avec M. Perrin et récupération du matériel
- > Entretiens téléphoniques réguliers (2 ou 3 par semaine)
avec M. Nguyen pour poser des questions
- > Oral blanc d'analyse avec M. Beilin

► Suivi

- > Un entretien téléphonique par semaine, pour informer de l'avancement du projet

► Recette

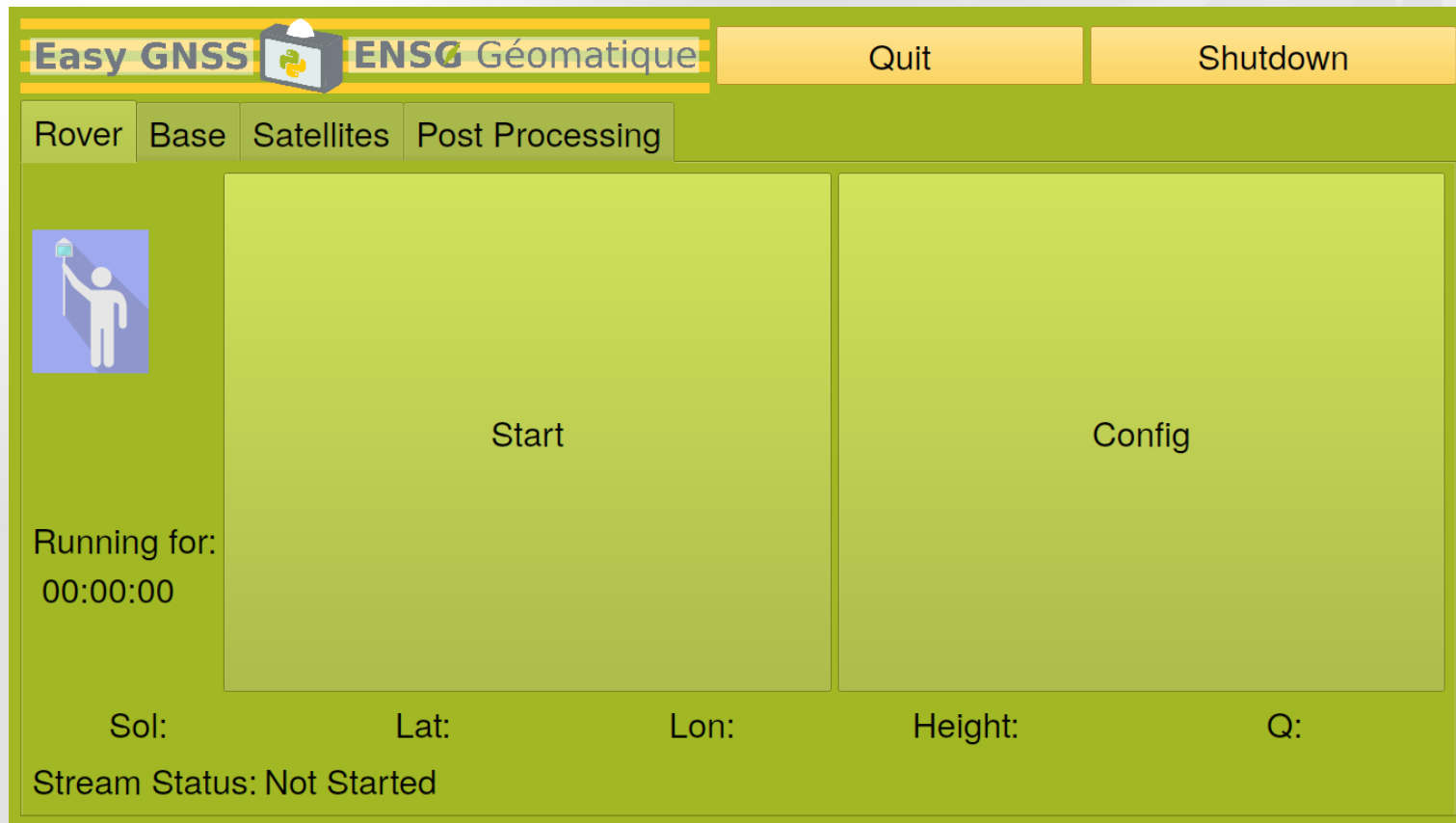
- > Entretiens téléphoniques réguliers
- > Présentation aux commanditaires dans le hall



V – Bilan du projet


BILAN DU PROJET

Interface graphique



BILAN DU PROJET

Interface graphique

Easy GNSS  **ENSG Géomatique**

Quit Shutdown

Rover Base Satellites Post Processing

Conf file Calculus BasePos Solution Log Input 1 Input 2

Use existing conf file ? ☐ Enable

Select the conf file to be used

Save created conf file ? ☒ Enable

Filename

Apply Close

Q:

- ▶ Un fork GitHub de TouchRTKStation
- ▶ Le code Python d'EasyGNSS, avec les fonctionnalités exigées
- ▶ Une documentation utilisateur complète
- ▶ Divers autres documents relatifs au projet

- ▶ Le GNSS low-cost sera testé par les ING1 pendant le stage à Forcalquier 2019
- ▶ Améliorations possibles :
 - > Rendre l'affichage de l'interface plus ergonomique
 - > Ajouter une documentation intégrée, pour permettre à l'utilisateur de choisir le mode de calcul adapté à ses besoins
 - > Faire du post-processing avec d'autres constellations que celles actuellement implémentées
- ▶ Au terme du projet, le GNSS sera utilisable sur des drones



Merci de votre
attention !

Avez-vous des
questions ?