

nom de l'établissement Lycée International Victor Hugo Boulevard Victor Hugo 31770 COLOMIERS Tél : 0561159494	BTS Systèmes Numériques Option A Informatique et Réseaux	Session 2020
--	---	---------------------

InitCube

<i>Partenaire professionnel :</i>	<i>Étudiants chargés du projet :</i>	<i>Professeurs ou Tuteurs responsables :</i>
Centre National d'Études Spatiales	Noms Prénoms	Noms Prénoms
Centre de Toulouse	-	- BOUYSSONNADE Solange
18, avenue Edouard Belin	-	- COMMENGE Olivier
31401 Toulouse Cedex 4	-	-
	-	-
	-	-

Reprise d'un projet : *Oui / Non*

1 Présentation générale du système supportant le projet :

1.1 PROJET JANUS

L'objectif du CNES est de structurer les activités de développement des cubesats au sein des universités et des écoles d'ingénieurs. Pour cela le CNES a mis en place depuis 2012 un projet qui regroupe le développement et l'exploitation de systèmes orbitaux réalisés par des étudiants de l'enseignement supérieur. Il a été baptisé:

Jeunes en
Apprentissage pour la réalisation de
Nanosatellites au sein des
Universités et des écoles de l'enseignement
Supérieur



Les objectifs majeurs de JANUS sont de :

- Promouvoir les activités spatiales auprès des étudiants de l'enseignement supérieur en leur proposant de développer des systèmes orbitaux composés de :
 - . Nanosatellites de masse comprise entre 1 et 20 kg, en privilégiant la norme « Cubesat ».
 - . Segment sol (station de réception de télémesures et d'émission de télécommandes, centre de contrôle, centre de mission).
- Promouvoir des démonstrateurs en orbite intéressant la communauté scientifique et industrielle. Ceci permettra de valider, en peu de temps et à faible coût, des nouvelles technologies satellites et/ou instrumentales (matériaux, détecteurs, composants, ASIC, propulsion, contrôle d'attitude, calculateurs, architecture électrique, cellules solaires, batteries, moyens de communication,...) et permettra d'améliorer les TRL (Technical Readiness Level). Les résultats de ces démonstrations technologiques pourront être utilisés par le CNES pour les besoins d'autres missions spatiales.

Analyse de l'existant :

Le nom CubeSat (cubesatellite, satellite cube) est associé à un type de nanosatellites défini par la « Cubesat Design Specification » (<http://www.cubesat.org/resources/>), officiellement appelé le standard CubeSat et écrite par l'université de Stanford aux Etats-Unis .

Ce standard définit la puissance, la masse et les dimensions :

- 1U CubeSat mesure 10 cm × 10 cm × 11,35 cm.
- CubeSat 2U: 10 cm × 10 cm × 22,70 cm.
- CubeSat 6U: 20 cm × 10 cm × 34,05 cm.
- CubeSat 12U: 20 cm × 20 cm × 34,05 cm.

Ces dimensions sont suffisantes pour réaliser les fonctions de base du satellite (la communication, l'électronique, la navigation, etc.) et pour porter une « charge utile » permettant au satellite de remplir sa mission scientifique.

Les missions typiques des nanosatellites sont actuellement dédiées soit à l'observation et la mesure de l'environnement terrestre, soit aux tests de nouvelles technologies dans l'environnement spatial. Un nanosatellite peut accompagner un gros satellite lors d'une mise en orbite par une fusée et ainsi permettre aux universités du monde entier de lancer des expériences scientifiques dans l'espace, pour un coût réduit.

En effet, la majorité des cubesats sont fabriqués par des universités et des laboratoires scientifiques.

Au 28 octobre 2018, 878 CubeSats ont déjà été lancés dont 400 nano-satellites en 2018. Plus de 3000 lancements de nano-satellites sont envisagés en 6 ans.

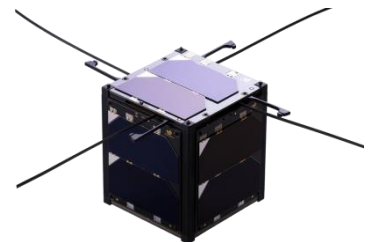
Expression du besoin :

Pour compléter sa mission, JANUS a besoin d'un outil concret permettant d'initier au spatial les étudiants des premières années de l'enseignement supérieur voire même dès la terminale. Cet outil sera un **kit d'initiation aux cubesats (nommé par la suite InitCube)**.

Le kit est prévu pour proposer un système complet comparable à un vrai système spatial avec le segment vol et le segment sol et des outils d'utilisation. **InitCube** sera donc composé :

- d'un **cubesat 1U** constitué :
 - ✓ D'une structure au facteur de forme du standard cubesat
 - ✓ Des fonctions : énergie, radiofréquence, calculateur, logiciel de vol, instrument,
- d'un **segment sol** permettant de communiquer avec le cubesat directement par voie numérique ou par radio fréquence.
- D'un **accompagnement des utilisateurs** principalement des tutoriaux.

Le schéma suivant donne l'organisation d'InitCube :



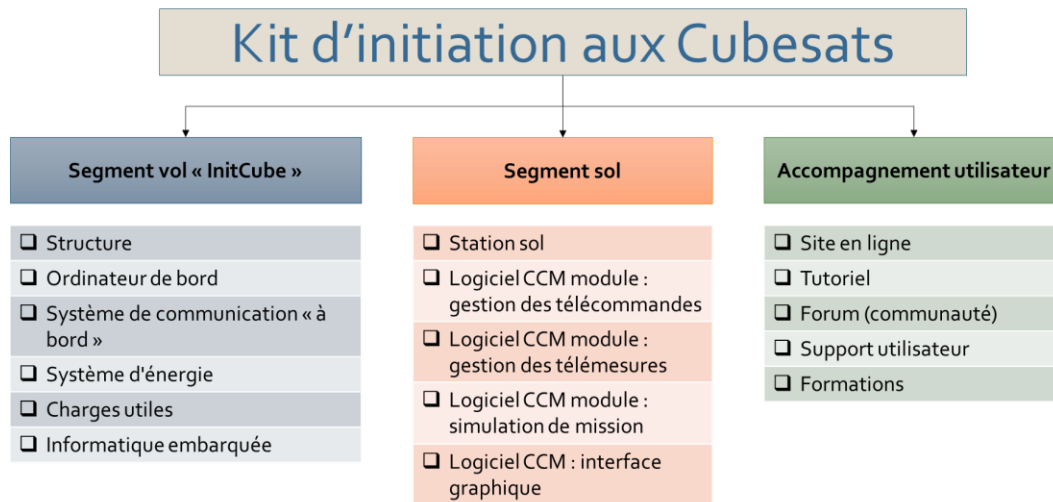


Figure 1 : schéma d'organisation d'InitCube

InitCube sera utilisé par des étudiants et leurs professeurs.

L'utilisateur pourra désassembler et réassembler à volonté InitCube afin de faire ces propres modifications et/ou évolutions. Le Cubesat communiquera avec l'ordinateur de l'utilisateur soit par la station au sol, soit directement par câble avec le PC pour gérer les logiciels embarqués.

Depuis son PC, l'utilisateur pourra :

- Gérer les logiciels embarqués (si le cubesat est directement câblé au PC) ;
- Utiliser le logiciel du CCM ;
- Accéder au site en ligne.

Aussi, le kit serait utilisable soit en mode « développement » lorsque l'utilisateur met en place du nouveau matériel ou des nouveaux logiciel (voir figure 2), soit en mode « opérations » lorsque l'utilisateur se familiarise avec des activités spatiales (voir figure 3).

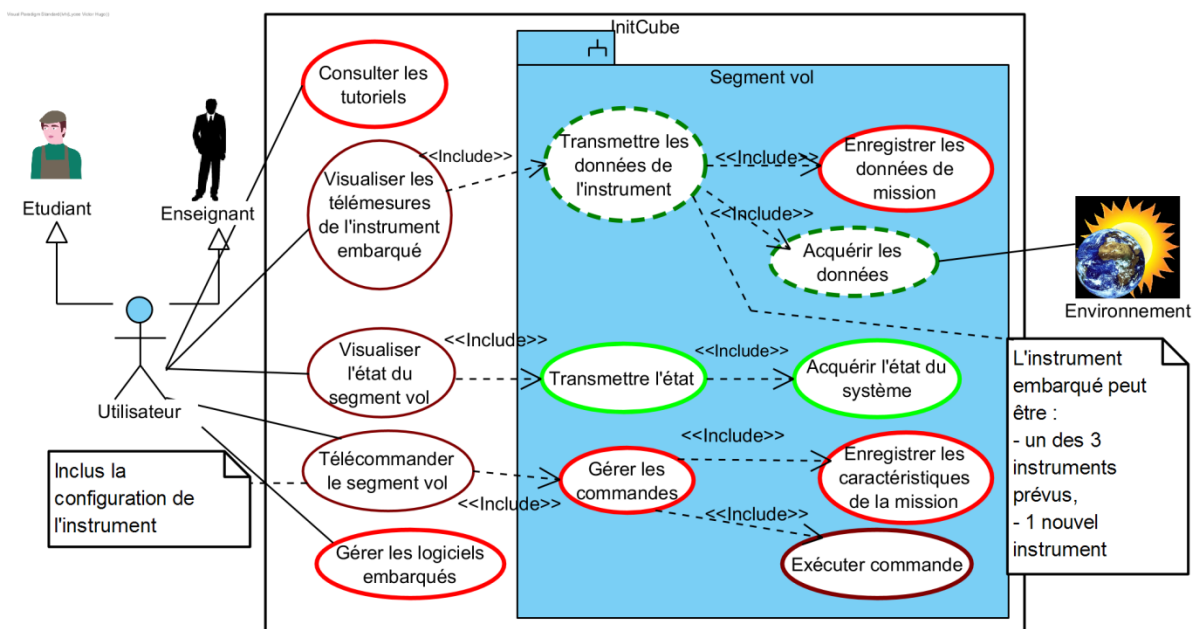
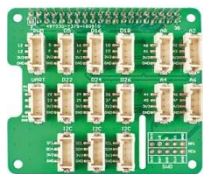


Figure 2 : diagramme des cas d'utilisation

Énoncé des tâches à réaliser par les étudiants (Segment Vol) :

Candidat	Tâche(s)	
Candidat 1	Gérer les commandes	Analyser les classes SegmentSol, Protocole, Commande et Message. Établir comment elles interviendront dans le cas de la réception de commandes. Concevoir et développer le serveur de réception de commandes.
		Analyser la classe Mission et SegmentVol Développer la gestion des commandes de Mission et des commandes directes.
Candidat 2	Enregistrer les données de mission	Analyser les classes SegmentVol, Ordinateur, Stockage et Reboot. Établir comment elles interviendront dans le cas de la sauvegarde de missions avant envoi. Identifier le format des données à enregistrer. Concevoir et réaliser le module de sauvegarde des données de mission.
		Concevoir et réaliser le module de redémarrage du système sur commande ou sur incident.
	Transmettre les télémesures d'instrument	Analyser la classe Protocole et vérifier l'adaptabilité des méthodes de transmission des télémesures aux différents modèles de capteurs. Concevoir et développer les nouvelles méthodes.
Candidat 3	Acquérir les données de l'instrument (Camera/Photo)	Analyser les classes Instrument, I2C, Mode, Status et Mesure. Établir comment elles interviendront dans le cas de la prise de photos par camera.
	Vérifier l'état de l'instrument (Camera/Photo)	Identifier les besoins de généralisation complémentaires. Concevoir et réaliser le module d'acquisition de photos. Analyser les états de fonctionnement possibles de ce module et les gérer.
	Configurer	Établir le format de la configuration de la camera dans le fichier xml.
		Contribuer à la documentation utilisateur.
Candidat 4	Acquérir les données de l'instrument (Magnétomètre)	Analyser les classes Instrument, I2C, Mode, Status et Mesure. Établir comment elles interviendront dans le cas du magnétomètre. Identifier les besoins de généralisation complémentaires. Concevoir et réaliser le module d'acquisition du champ magnétique.
	Vérifier l'état de l'instrument (Magnétomètre)	Analyser les états de fonctionnement possibles de ce module et les gérer.
	Configurer	Établir le format de la configuration du magnétomètre dans le fichier xml.
		Contribuer à la documentation utilisateur.

Description structurelle du système :

Principaux constituants :	Caractéristiques techniques :
Raspberry Pi	
Pi Supply PiJuice HAT	batterie 1820 mAh intégrée mais support de plus grosses batteries Lipo de 5000 mAh. Charge indicateur niveau de charge masse : 57 g Gère la charge et l'alimentation par panneau solaire
Panneaux solaires	fournis par le CNES (Les panneaux Pi Solar ne respectent pas les dimensions d'un Cubesat)
Grove base Hat (I2C) 	<ul style="list-style-type: none"> - ADC : 12-Bit 8-Channel - 6x Digital Grove Port - 4x Analogique - 3x I2C - 1x PWM - 1x UART (modules radio)
Capteurs température, champs magnétique, caméra	I2C SPI etc.
Module radio émetteur-récepteur ISM WT-4432G 433 MHz	
Module radio émetteur-récepteur Zigbee	

Inventaire des matériels et outils logiciels à mettre en œuvre par le candidat :

Désignation :	Caractéristiques techniques :
Machine de développement	Machines virtuelles sur hôte Linux
Environnement Netbeans	Version 8.2
Langage C++	
Systèmes d'exploitation Raspbian avec modules PiJuice et Grove Hat	Stretch Lite
I2C tools	
Visual Paradigm (UML/SysML)	
Time Performance	

Le développement du Segment Sol sera divisé cette année en deux sous-projets distincts de trois étudiants chacun :

- « **Segment Sol – Communication** », qui gèrera la transmission des télécommandes au Segment Vol ainsi que la réception des données de télémessures des instruments et de l'état du Segment Vol. Cette année, les données reçues et émises devront être stockées dans une base de données en vue de conserver un historique et de pouvoir rejouer un scénario de mission.
- « **Segment Sol – IHM** », qui consiste à développer une IHM permettant de :
 - transmettre des commandes ponctuelles au Segment vol ;
 - concevoir un scénario de mission constitué d'une suite de commandes à transmettre au Segment Sol (commencé l'an passé mais non finalisé) ;
 - visualiser l'état du Segment Vol en mode textuel et graphique (réalisé l'an passé, à améliorer) ;

- visualiser les mesures des instruments. L'an passé l'instrument était une caméra infrarouge de 64 pixels dont l'affichage avait été réalisé. Cette année, on ajoute un magnétomètre et une caméra dans le spectre visuel. Les informations concernant les instruments (type et format des grandeurs mesurées, unité, fréquence d'acquisition,...) devront être extraites d'un fichier XML. Cette année le fichier sera fait à la main. Ces fichiers pourront être stockés sur un dépôt Git collaboratif tel que GitHub. La station Sol pourrait alors faire une mise à jour de la liste de ces fichiers à intervalles réguliers ou à la demande de l'utilisateur.
- assurer de nouvelles fonctionnalités telles que :
 - Visualiser l'historique des commandes transmises,
 - Rejouer un scénario de mission.

Énoncé des tâches à réaliser par les étudiants (Segment SOL - Communication) :

Candidat	Tâche(s)	
Candidat 1	Rejouer un scenario de mission	Analyse et compréhension du sujet complet. Conception de la base de données permettant d'assurer l'historique des télécommandes transmises au Segment Vol ainsi que des données reçues de celui-ci (télémessures et état) Conception et réalisation des interfaces avec la BDD.
	Visualiser l'historique des télécommandes	
	Gérer les logiciels embarqués	Gestion des fichiers XML décrivant les instruments (téléchargements via github, création d'un service pour mettre à disposition les données caractéristiques de chaque instrument)
		Installation et configuration de la Raspberry Pi
Candidat 2	Télécommander le Segment Vol	Analyse et compréhension du sujet complet. Concevoir et réaliser un service d'envoi des télécommandes au Segment Vol (réception des télécommandes en provenance du client web, encapsulation suivant le protocole défini l'an passé, transmission via le module radio, stockage des télécommandes transmises dans la BDD pour assurer l'historique)
		Installation et configuration de l'architecture web (serveurs)
Candidat 3	Visualiser les mesures de l'instrument embaqué	Analyse et compréhension du sujet complet. Améliorer et compléter le service de réception des télémessures et de l'état du Segment Vol (réception des trames en provenance du Segment Vol via le module radio, extraction des données de la trame suivant le protocole défini l'an passé, stockage des données reçues dans la BDD pour l'historique, mise en forme des données au format JSON et transmission aux clients.
	Visualiser l'état du Segment Vol	
		Installation et configuration de l'infrastructure réseau autour de Segment Sol

Principaux constituants :	Caractéristiques techniques :
Raspberry Pi	
Module radio émetteur-récepteur ISM WT-4432G 433 MHz	
Module radio émetteur-récepteur Zigbee	

Inventaire des matériels et outils logiciels à mettre en œuvre par le candidat :

Désignation :	Caractéristiques techniques :
Machine de développement	Machines virtuelles sur hôte Linux
Environnement Netbeans	Version 8.2
Langage C++ ou Javascript (Node.js)	
Systèmes d'exploitation Raspbian	Stretch Lite
BDD SqLite, MySQL ou MongoDB (si Node.js)	
Visual Paradigm (UML/SysML)	
Time Performance	
Gestionnaire de version Git/GitHub	

Énoncé des tâches à réaliser par les étudiants (Segment SOL - IHM) :

Candidat	Tâche(s)	
Candidat 1	Télécommander le Segment Vol	Analyse et compréhension du sujet complet. Reconcevoir l'IHM web permettant de télécommander le Segment Vol et de définir un scénario de mission (suite de commandes à transmettre)
		Développer la partie client permettant de se connecter au serveur de commande (en relation avec l'équipe « Segment Sol - Communication »)
Candidat 2	Visualiser les mesures de l'instrument embaqué	Analyse et compréhension du sujet complet. Concevoir et réaliser une IHM web conviviale et modulaire pour suivre en temps réel les mesures du ou des instruments du cubesat (magnétomètre et caméra)
		Améliorer et adapter la partie client permettant de se connecter au serveur de réception des données en provenance du Segment Vol (en relation avec l'équipe « Segment Sol - Communication »).
Candidat 3	Visualiser l'état du Segment Vol	Analyse et compréhension du sujet complet. Améliorer l'IHM de suivi de l'état du Segment Vol
	Rejouer un scénario de mission	Concevoir et réaliser une IHM web conviviale permettant d'accéder à l'historique des missions effectuées
		Concevoir et réaliser une IHM web conviviale pour rejouer un scénario de mission.
		Développer la partie client de la BDD (en relation avec l'équipe « Segment Sol - Communication »).

Inventaire des matériels et outils logiciels à mettre en œuvre par le candidat :

Désignation :	Caractéristiques techniques :
Machine de développement	Machines virtuelles sur hôte Linux
Environnement Netbeans	Version 8.2
Langage C++ (script CGI) ou javascript Node.js	
Systèmes d'exploitation Raspbian	Stretch Lite
BDD SqLite, MySQL (si C++) ou MongoDB (si Node.js)	
Langage HTML5/CSS	
Framework JQueryMobile ou Angular.js	
Visual Paradigm (UML/SysML)	
Time Performance	

Joindre en annexe, les documents explicitant le projet : photos, fiches techniques descriptives, procédé(s) mis en œuvre, cahier des charges simplifié, schémas etc...

Tâches	Revue	Contrats de tâche	Compétences	Candidat_1	Candidat_2	Candidat_3	Candidat_4	Candidat_1	Candidat_2	Candidat_3	Candidat_1	Candidat_2	Candidat_3
		Expression fonctionnelle du besoin											
T1.4	R2	Vérifier la pérennité et mettre à jour les informations.	C2.1										
T2.1	R2	Collecter des informations nécessaires à l'élaboration du cahier des charges préliminaire.	C2.2										
T2.3	R2	Formaliser le cahier des charges.	C2.3 C2.4	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
T3.1	R2	S'approprier le cahier des charges.	C3.1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
T3.3	R2	Élaborer le cahier de recette.	C3.5	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
T3.4	R2	Négocier et rechercher la validation du client.	C2.4	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
		Conception											
T4.2	R3	Traduire les éléments du cahier des charges sous la forme de modèles.	C3.1 C3.3	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
T5.1	R3	Identifier les solutions existantes de l'entreprise.	C3.1 C3.6										
T5.2	R3	Identifier des solutions issues de l'innovation technologique	C3.1 C3.6	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
T4.3	R3	Rédiger le document de recette.	C4.5	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
T6.1	R3	Prendre connaissance des fonctions associées au projet et définir les tâches.	C2.4 C2.5	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
T6.2	R3	Définir et valider un planning (jalons de livrables).	C2.3 C2.4 C2.5	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
T6.3	R3	Assurer le suivi du planning et du budget.	C2.1 C2.3 C2.4 C2.5	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
		Réalisation											
T7.1	R3	Réaliser la conception détaillée du matériel et/ou du logiciel.	C3.1 C3.3 C3.6	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
T7.2	RF	Produire un prototype logiciel et/ou matériel.	C4.1 C4.2 C4.3 C4.4	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
T7.3	RF	Valider le prototype.	C3.5 C4.5 C4.6	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
T7.4	RF	Documenter les dossiers techniques et de maintenance	C2.1 C4.7	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
T9.2	RF	Installer un système ou un service.	C2.5	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
T10.3	RF	Exécuter et/ou planifier les tâches professionnelles de MCO.	C2.5	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
T11.3	RF	Assurer la formation du client.	C2.2 C2.5	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
T12.1	RF	Organiser le travail de l'équipe.	C2.3 C2.4 C2.5	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
T12.2	RF	Animer une équipe.	C2.1 C2.3 C2.5	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
		Vérification des performances attendues											
T9.1	RF	Finaliser le cahier de recette.	C3.1 C3.5 C4.5	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

<i>Avis de la commission</i>

- Les concepts et les outils mis en œuvre par le candidat (1-2-3-4-5)... correspondent au niveau des exigences techniques attendu pour cette formation :

oui / à reprendre pour le candidat (1-2-3-4-5)

- L'énoncé des tâches à réaliser par le candidat (1-2-3-4-5)... est suffisamment complet et précis :

oui / à reprendre pour le candidat 1-2-3-4-5

- Les compétences requises pour la réalisation ou les tâches confiées au candidat (1-2-3-4-5) sont en adéquation avec les savoirs et savoir-faire exigés par le référentiel :

oui / à reprendre pour le candidat (1-2-3-4-5)

- Le nombre d'étudiants est adapté aux tâches énumérées :

oui / trop / insuffisant

Commentaires

Date :

Le président de la commission

Annexe 1 : deux utilisations possibles

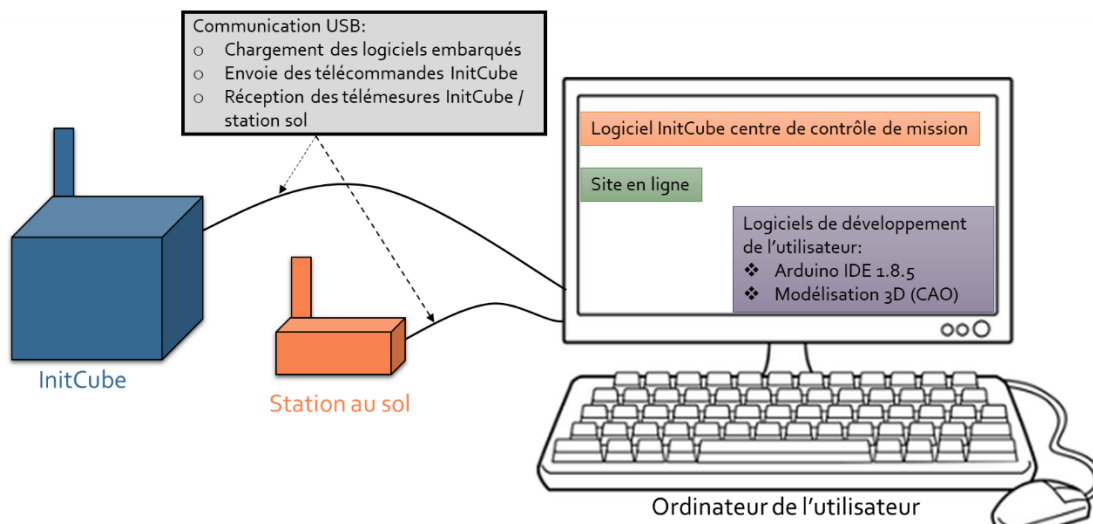


Figure 2: schéma fonctionnel du kit en développement

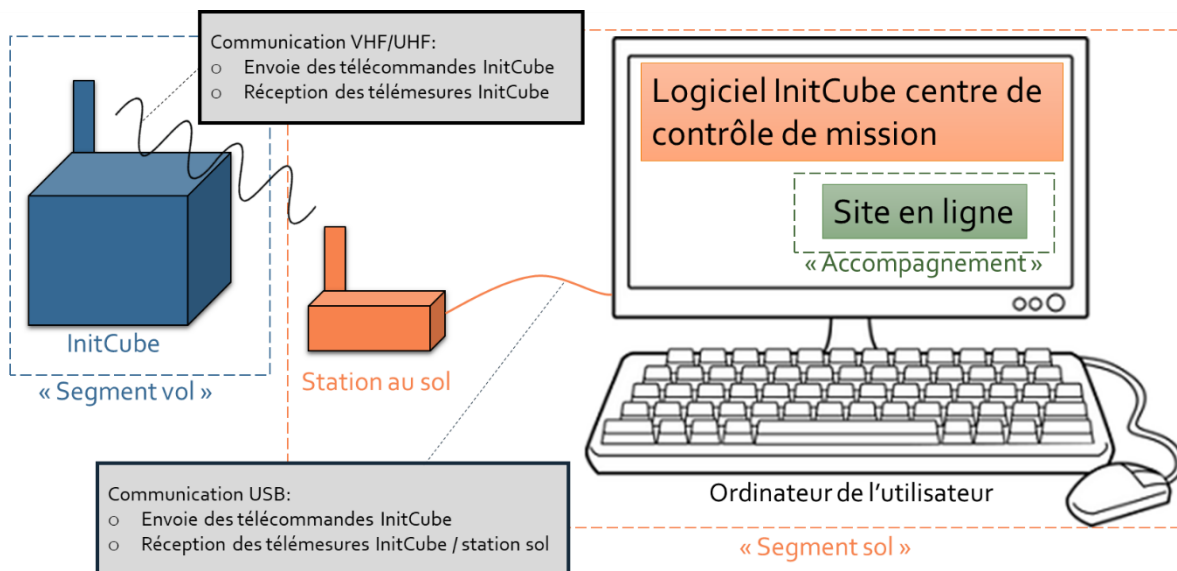


Figure 3: Schéma fonctionnel du kit en opérations

Annexe 2 : le cube existant

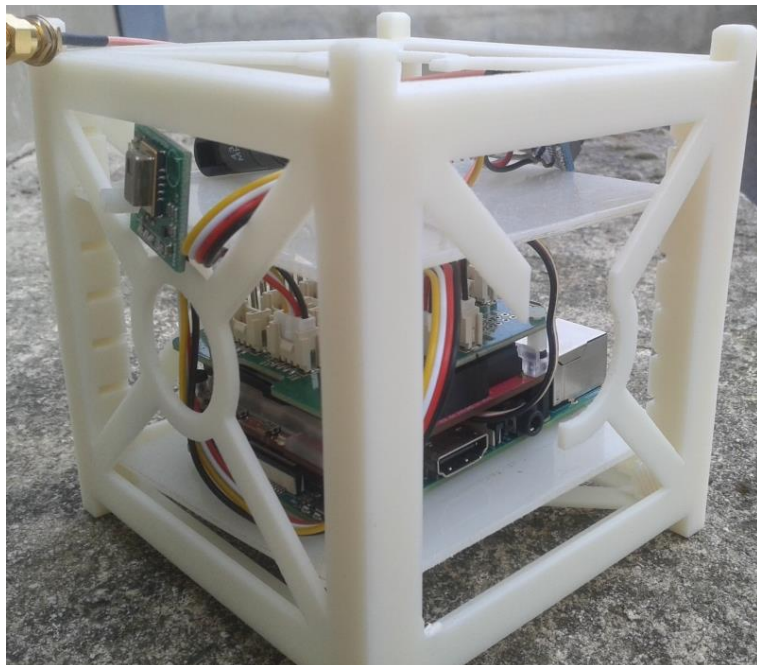


Figure 3 : le cube existant (Version 2019)

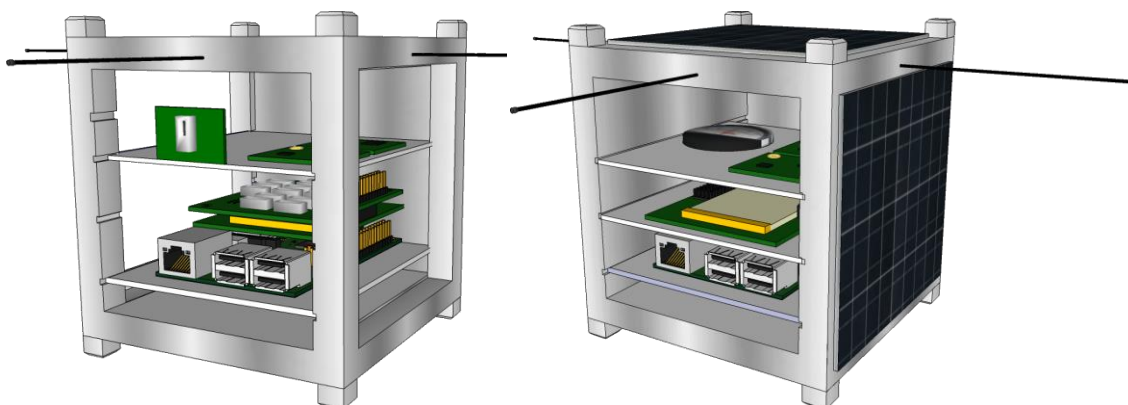


Figure 4 : maquette du cube (ouvert pour montage éléments et fermé pour démonstration)