

Projets co-eLAB
Année Universitaire 2017 - 2018
Semestre 2

Code projet :2722

COMMANDITAIRE

YNCREA

SUJET

FARMING BY SATELLITE

ELEVES INGENIEURS - DOMAINE

Louis Anthonioz-HEI
Ugo Marchand-HEI
Jean Le-Bellego-ISEN
Tancrede Bernarovat-ISEN
Salah-Eddine Guermat-ISEN

JURY DE SOUTENANCE

Président de jury	Bertrand VANDOORNE
Représentant(s) de l'entreprise	Marie MARION ROLLOY
Professeur(s) superviseur(s)	Nacim IHADDADENE

RESUME

Le projet co-elab Farming By Satellite a commencé en 2017. Celui-ci porte sur l'optimisation et la spatialisation des différentes plantes composant un couvert végétal en fonction des propriétés du sol. Propriété obtenues grâce à l'imagerie satellite. L'étude précédente a porté sur la faisabilité technique de l'optimisation du semis des couverts végétaux grâce à l'imagerie satellite au niveau intra-parcellaire. Les études de veilles concurrentielles précédemment réalisées ont montré qu'il existe de nombreuses entreprises dans le domaine de l'agriculture de précision et de fertilisation azotée et peu d'entreprises dans le domaine des couverts végétaux.

Dans le but d'une future création de start-up (en intrapreneuriat), le projet a continué afin de finaliser la partie informatique.

Les indices du satellite Sentinel 2 ont pu être extraits automatiquement, l'indice DEM du satellite Sentinel 3 a pu être extrait manuellement. De plus le site internet qui fait office d'interface avec l'agriculteur a été modifié. Toutes les données ont été récupéré sur une base de données. Et un site internet qui fait office d'interface administrateur a été créé permettant de récupérer les informations des clients et de traiter sa demande concernant les couverts végétaux.

Table des matières

INTRODUCTION	5
GESTION DE PROJET	6
1. Note de clarification	6
1.1. Présentation	6
1.2. Définition et objectifs du projet.....	7
1.3. Acteurs du projet	8
1.4. Validation	8
2. Bilans de compétences.....	9
Louis Anthonioz-HEI.....	9
Ugo Marchand-HEI	9
Jean Le-Bellego-ISEN.....	10
Tancrede Bernarovat-ISEN	10
Salah-Eddine Guermat-ISEN	10
3. L'équipe	11
3.1. Formation.....	11
3.2. Déroulement.....	11
3.3. Difficultés rencontrées.....	11
3.4. Méthode de travail.....	12
4. Conclusion	12
LE LIVRABLE	13
1. Récupération des indices	13
1.1. Satellite Sentinel 2	13
1.2. Satellite Sentinel 3	14
1.3. Logiciel indispensable pour l'extraction des indices.....	16
2. Les Interfaces web.....	16
2.1. La base de données.....	16
2.2. Modification de l'interface client.....	16
2.3. Interface web administrateur	19
3. Composition des couverts végétaux	24
3.1. Les entrées	24
3.2. Les sorties.....	25
3.3. Les règles.....	25
4. Présentation du livrable	27
CONCLUSION GENERALE	29

BIBLIOGRAPHIE	30
TABLE DES FIGURES	30
TABLE DES TABLEAUX	32
TABLE DES LIENS	32
ANNEXES	33
1. Cahier des charges.....	33
1.1. Extraction des images	33
1.2. Création d'un site interne	34
2. Méthode de détermination des Couverts végétaux	36
2.1. Démarche mise en place	36
2.2. Etudes des espèces à insérer et leur densité	36
2.3. Tableaux des relations	40
3. Le fonctionnement d'une interface web.....	43
3.1. Utilisation de HTML et CSS.....	43
3.2. Utilisation de Javascript	44
3.3. Utilisation de Node.js.....	45
3.4. En Conclusion	49
4. Faire une base de données en Json.....	50
4.1. Présentation	50
4.2. Son fonctionnement	51
4.3. Pour résumer	52
4.4. Requête MongoDB	52
4.5. Les fonctions utilisées	53
5. Le management du projet.....	54
5.1. Les plannings.....	54
5.2. Timesheet.....	56
6. L'interface client : version complète	57
7. Méthode de récupération des indices du satellite Sentinel 2.....	61
7.1. Le fichier bash	61
7.2. Gestion de la récupération des indices.....	62
7.3. Calcul du NDVI par traitement d'images.....	62
8. Méthode de récupération du DEM	64

INTRODUCTION

De nos jours, l'agriculture se trouve dans un nouveau tournant pour concilier une agriculture durable tout en étant productive. Ce tournant s'appuie sur toutes les technologies disponibles comme les satellites, drones ou encore capteurs... Grâce à ces nouveaux grands enjeux mondiaux, des concours sont montés en vue de trouver de nouvelles solutions et concepts. En Janvier 2017, le concours européen Farming by satellite a récompensé l'équipe isaïenne pour son idée sur les couverts végétaux connectés. L'idée proposée est d'adapter le semis de couverts végétaux en fonction des propriétés du sol, renseignées par imagerie satellite. Cette idée permettrait de fournir à l'agriculteur une carte de modulation intra-parcellaire.

Les travaux à effectuer concerne la partie informatique du projet. Un premier groupe avait commencé cette partie. Il avait récupéré l'indice NDVI de façon automatique et créé une interface client sur laquelle les clients pourront renseigner leurs informations. L'objectif de la deuxième partie est de continuer les travaux commencés précédemment. Dans la finalité de permettre à notre client de lancer son projet en intrapreneuriat.

Pour cela, Il faudra modifier l'interface client pour permettre aux clients de renseigner toutes leurs informations et la rendre plus fonctionnelle. Il faudra aussi récupérer tous les indices des satellites sentinel 2 et 3. Ensuite, ces renseignements devront être placés dans une base de données. Puis une interface administrateur devra être créer pour traiter la demande de chaque client afin de déterminer les couverts végétaux les plus adaptés.

GESTION DE PROJET

1. Note de clarification

Farming by Satellite	N° DU PROJET : 2722
	DATE : 12/02/2018

1.1. Présentation

a. Contexte

Issu d'un projet étudiant ISA, le concept des couverts végétaux connectés a remporté en 2017 le concours européen Farming by Satellite. Celui-ci porte sur l'optimisation et la spatialisation des différentes plantes composant un couvert végétal en fonction des propriétés du sol. Propriété obtenues grâce à l'imagerie satellite. L'étude précédente a porté sur la faisabilité technique de l'optimisation du semis des couverts végétaux grâce à l'imagerie satellite au niveau intra-parcellaire. Les études de veilles concurrentielles précédemment réalisées ont montré qu'il existe de nombreuses entreprises dans le domaine de l'agriculture de précision et de fertilisation azotée et peu d'entreprises dans le domaine des couverts végétaux.

Dans le but d'une future création de start-up (en intrapreneuriat), nous continuons le projet enfin de finaliser la partie informatique.

b. Données d'entrée

- Rapports du projet « Farming by Satellite » du semestre 1 (sept-décembre 2017)
- Lien wetransfert du PPT et vidéo de présentation du projet
- Lien Github réalisé par les membre ISEN du projet précédent.

c. Dates (début et fin)

Du 5 février 2018 au 27 avril 2018

1.2. Définition et objectifs du projet

a. Objectif du projet

- Extraire et traiter les images des indices MTVI2 et MCARI provenant le satellite sentinel 2 et les indices VTCI, CTI, DEM, S% provenant du satellite sentinel 3.
- Améliorer l'interface client existante.
- Créer une interface web administrateur permettant de centraliser les saisies extraites des satellites sentinel.

b. Résultats attendus

- Finaliser la partie informatique du projet
- Délai du projet : Du 5 février 2018 au 27 avril 2018

c. Livrables

- Rapport du projet
- Comptes-rendus
- Interface web

d. Budget

Pas de budget défini

e. Contraintes

- Le sujet nécessite un approfondissement des acquis en informatique.
- Nécessite une organisation et une communication importante, compte tenu des différents emplois du temps des acteurs du projet.

f. Conséquences attendues

- Finaliser la partie informatique du projet global de Farming by Satellite.
- Se rapprocher de l'objectif de création d'une Start-up.

1.3. Acteurs du projet

Etudiants :

- Louis Anthonioz-HEI
- Ugo Marchand-HEI
- Jean Le-Bellego-ISEN
- Tancrede Bernarovat-ISEN
- Salah-Eddine Guermat-ISEN

Professeur superviseur :

- Nacim Ihaddadene

Client :

- Marie Marion-Rolloy

1.4. Validation

Dates et visa

Le cahier des charges du projet a été réalisé par notre Client, Marie Marion Rolloy.
Pour avoir davantage d'informations sur son contenu, voir annexe ([Cahier des charges](#)).

2. Bilans de compétences

Louis Anthonioz-HEI

Le projet co-elab m'a permis de travailler différemment que dans mes projets précédents. J'ai travaillé en dehors de mon école, avec des personnes extérieures à HEI. J'ai eu l'occasion de rencontrer des personnes ayant des domaines de professionnalisation très différents. Ceci m'a permis d'enrichir ma culture générale, ce qui me sera bénéfique dans le monde du travail. Avec ce projet j'ai acquis de nombreuses connaissances. En effet, le projet Farming By Satellite concerne une spécialité que je ne connaissais pas, j'ai pu ainsi découvrir le monde agricole. De plus, j'ai travaillé sur la partie informatique du projet. C'était un domaine dans lequel j'avais très peu de connaissance. J'ai appris différents langages (HTML, CSS, Javascript et Node.js). Ce projet m'a montré qu'il est toujours possible de réussir à condition de se donner les moyens. C'est avec cette philosophie de penser que je vais débiter ma vie professionnelle.

Ugo Marchand-HEI

Le projet co-elab fut une nouvelle expérience enrichissante et alimentée d'imprévus. Les projets précédents à HEI faisaient appel à certaines compétences techniques bien particulières en 4^{ème} année, spécifique à notre formation, ou encore à une vision plus large de la mise en place d'un projet, faisant appel à plus de culture générale, en 3^{ème} année. Ce projet de fin d'étude représentait un réel défi technique et managérial, et ceci en fut d'autant plus prenant. Dans un premier, le groupe était constitué de 2 HEI et de 3 ISEN. La première difficulté fut donc d'adapter les calendriers et les tâches à répartir suivant ces contraintes de disponibilités. La plus importante contrainte fut bien entendu l'aspect technique. Après notre premier entretien avec notre cliente et l'élaboration du cahier des charges par cette dernière, il fut clair que les exigences étaient purement informatiques. Tout les membres du projet étant spécialisés en électricité, électronique et systèmes embarqués, il a donc été primordial de se documenter et d'apprendre de manière autonome les langages nécessaires au projet. Ainsi, j'ai pu renforcer mes connaissances en HTML et CSS, et apprendre de nouveaux langages, tels que le Javascript ou le langage python (indispensable pour la partie récupération des indices). J'ai pu de plus étendre ma culture, notamment dû à l'aspect intrinsèque du projet. Je me suis ainsi familiarisé avec l'environnement de l'agriculture qui m'était jusqu'alors inconnu. J'ai donc finalement fait appel à des ressources telles que l'adaptabilité et la persévérance pour mener à bien ce projet. Pour sûr, je serais amené à réutiliser les compétences que j'ai acquises durant ce projet.

Jean Le-Bellego-ISEN

Ce projet a été pour moi une bonne expérience. Le sujet m'a beaucoup plus et le croisement des compétences entre des élèves d'écoles différentes est vraiment intéressant. Un peu perdu au début du projet par le manque de compétences dans le domaine du web et de la programmation, j'ai réussi cependant après une première phase assez fastidieuse à développer de bons acquis dans ces domaines.

Tancrede Bernarovat-ISEN

Ce projet m'a permis de découvrir le monde du web : étant en majeure ingénieur d'affaires en objet connectés je n'avais jamais eu la possibilité de m'y pencher. Je ne pensais pas au début que ce projet serait entièrement de la programmation, cependant il est vrai que j'ai énormément appris durant celui-là et ces connaissances me seront utiles lors mes futurs stages.

J'ai trouvé l'opportunité du projet Co-elab très enrichissante, il était intéressant de travailler avec des personnes ayant des cursus et des habilités ou encore manières de réfléchir totalement différentes de celles que l'on voit à ISEN. Il est cependant dommage que ces particularités propres à chacun n'aient pas pu être utilisées à bon escient car non adaptées au sujet du projet. En effet, aucun de nous n'avait de réelles capacités en programmation, nous avons donc tous dû apprendre par nous-même ce qui nous a fait perdre beaucoup de temps.

Salah-Eddine Guermat-ISEN

Le projet co-elab est très enrichissant car il m'a permis d'acquérir de nouvelles compétences ainsi, de développer un savoir-faire et un savoir-être important. En effet, j'ai eu la chance de travailler avec des étudiants d'une autre école qui suivent des domaines de professionnalisations différents. Ce qui m'a permis d'étendre mes activités, d'optimiser mon organisation et de mieux me structurer pour une meilleure efficacité de travail.

A travers le projet FarmingBySatellite, j'ai eu l'occasion de connaître un domaine que j'ignorais avant. En effet, j'ai pu découvrir le monde d'agriculture. En outre, j'ai eu la chance d'acquérir certaines compétences au niveau de la programmation et du développement web.

Effectivement, ce projet a été pour moi une nouvelle expérience et un challenge important car durant une période courte, on a pu travailler ensemble afin d'atteindre notre objectif. Ce qui m'a permis de comprendre non seulement qu'avec la volonté et la détermination on arrive toujours à notre but.

3. L'équipe

3.1. Formation

Le projet a commencé au début du mois de Février. La première semaine a permis de nous rencontrer et de découvrir le sujet. Nous avons eu l'occasion de rencontrer notre Client et notre tuteur lors de cette semaine. L'équipe est formée de deux étudiants HEI dans le domaine ESEA (Energie, Système Electrique et Automatique) et trois étudiants ISEN dans le domaine Système Embarqué.

3.2. Déroulement

Le projet s'est décomposé en quatre grandes étapes.

- S'imprégner du sujet : Nous avons repris la suite d'un projet ayant commencé un an auparavant. Lors du semestre précédent (Octobre-Décembre 2017) la partie informatique avait été commencée. Nous avons dû reprendre se travail et le comprendre dans le but de modifier certaines parties, en compléter d'autres et en créer des nouvelles.
- Se former : Etant donné qu'aucun des membres de l'équipe avait des qualifications solides en informatique et notamment en développement web, nous avons consacré une grande partie de nos 12 semaines à notre formation.
- Le livrable : Après la formation, nous sommes entrés dans le vif du sujet ; le livrable, la réponse au cahier des charges.
- Les rendus : Nous avons dû réaliser le rapport, créer une vidéo de présentation du projet et préparer une présentation pour la soutenance.

Le planning prévisionnel, réel et le timesheet sont présentés en annexe ([Le management du projet](#)).

3.3. Difficultés rencontrées

Il y a eu deux grandes difficultés :

- Le manque de connaissances en informatique : Nous avons fait une formation accélérée en quelques semaines pour apprendre les différents langages utiles à notre projet (HTML, CSS, Javascript, Node.js, Json, Python). Nous nous sommes répartis au mieux les tâches pour que chacun n'ait pas besoin d'apprendre tous ces langages.
- L'emploi du temps : Les membres HEI étaient à temps complet sur le projet en revanche les membres ISEN travaillaient un jour par semaine pendant deux mois. Ils ont ensuite été à temps complet le dernier mois. De ce fait, l'organisation du groupe à changer à plusieurs reprises. D'où le non suivi du planning prévisionnel.

3.4. Méthode de travail

Nous avons fait des réunions de groupe chaque semaine pour se maintenir au courant de l'avancé et des travaux à réaliser pour la suite du projet. Nous avons travaillé surtout dans les locaux de ISEN pour une question d'organisation ; soit dans le local ADICODE ISEN, soit dans des salles de travail du bâtiment. Nous avons été à Urbawood pour rencontrer notre tuteur et faire des réunions skype avec notre client.

4. Conclusion

Le projet a été une bonne expérience pour l'équipe. Le début n'a pourtant pas été des plus facile ; la demande technique du cahier des charges nous a semblé très lourde. De plus, la gestion d'équipe a été problématique à cause des emplois du temps de chacun. Cependant, nous avons su rebondir et faire face aux enjeux. Nous avons mis en place une organisation de groupe avec une démarche tournée vers l'avant et non statique. Nous nous sommes lancés dans l'apprentissage des différents langages en autonomie. Ce qui nous a permis par la suite de réaliser les tâches du projet. Ce projet a permis à tous ces membres d'acquérir de nombreuses connaissances dans le domaine informatique mais aussi dans le domaine de la gestion de projet et du management.

LE LIVRABLE

1. Récupération des indices

Le but de cette partie était de récupérer de nouveaux indices, dans le prolongement du travail réalisé au semestre précédent. Le premier indice à récupérer était le MCARI/MTVI2 grâce au satellite Sentinel 2. Puis ensuite de réussir à recevoir des indices disponibles exclusivement par le biais du satellite Sentinel 3. Ces travaux se sont entièrement développés grâce à la programmation informatique avec des langages tel que Python et Java Script.

1.1. Satellite Sentinel 2

Notre mission concernant l'exploitation des données du satellite Sentinel 2 était de récupérer les indices permettant d'obtenir la quantité d'azote du sol. Le NDVI ayant été récupéré précédemment, notre objectif était de récupérer le ratio MCARI/MTVI2

Ce ratio s'obtient grâce à une formule mathématique utilisant 4 différentes bandes spectrales (cf. *Rapport du 1^{er} semestre*) récupérées grâce au satellite :

$$\frac{MCARI}{MTVI2} = \frac{((\text{bande 5} - \text{bande 4}) - 0,2 (\text{bande 5} - \text{bande 3})) \frac{\text{bande 5}}{\text{bande 4}}}{(1,5 \frac{1,2 (\text{bande 8} - \text{bande 3}) - 2,5 (\text{bande 4} - \text{bande 3})}{\sqrt{(2 \text{ bande 8} + 1)^2 - (6 \text{ bande 8} - 5 \sqrt{\text{bande 4}})}} - 0,5}$$

Figure 1 : Formule MCARI/MTVI2

Le principe :

Lorsque le client précise la localisation de ces parcelles sur la carte de l'interface client, le server node.js lance le fichier bash qui va récupérer les données du satellite Sentinel 2 correspondant à la localisation indiquée par le client. Les données récupérées sont des bandes au format JP2. Ensuite en utilisant la bibliothèque rasterio et en mettant les formules des indices dans le script python on récupère les images en .tif correspondant aux différents indices.

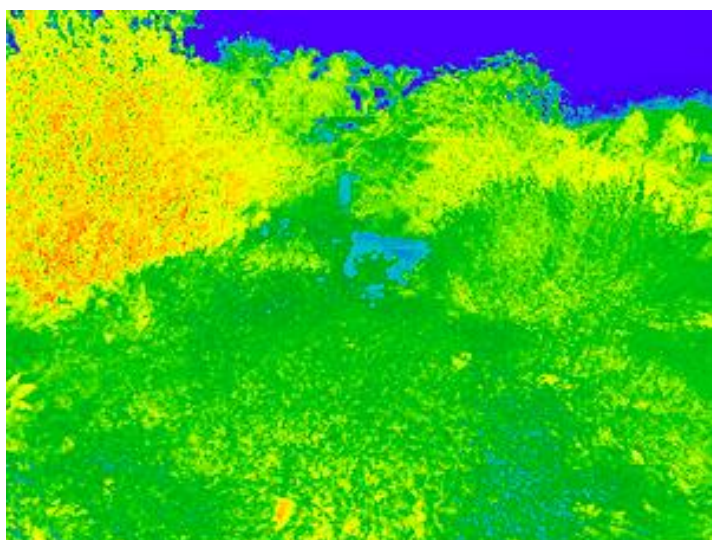


Figure 2 : Image NDVI

Pour plus de précision sur la récupération des indices du satellite Sentinel 2 voir annexe ([Méthode de récupération des indices du satellite Sentinel 2](#)).

Nous avons été confrontés à des difficultés pour exécuter les différents scripts. En premier, le fichier permettant la connexion avec le satellite était un fichier bash. Ce type de fichier ne peut pas être exécuter avec Windows et nécessite un environnement Linux. Une machine virtuelle ne suffit apparemment pas. Pour l'utilisation des modules spécifiques comme Rasterio, cela n'a pas été facile pour nous à mettre en place. La configuration de l'environnement sous linux nous a pris un certain temps.

1.2. Satellite Sentinel 3

Notre travail sur le satellite Sentinel 3 a débuté par une phase de veille et de recherche. Après quelques échanges avec le groupe précédent, il était à priori encore impossible de récupérer les indices automatiquement de la même manière que pour les indices du satellite Sentinel 2. L'API destinée à cette mission n'est effectivement pas encore disponible (toujours en développement à ce jour).

Il est cependant possible d'utiliser la version « pre-ops » (Pré-Opération Data Hub) du Sentinel 3 proposée par le site de copernicus. Celle-ci permet d'avoir accès à des archives de données récupérables en NRT (Near Real Time). Cette pre-ops propose une interface graphique permettant d'affiner les recherche grâce à des critères d'instruments utilisés et de zones géographiques.

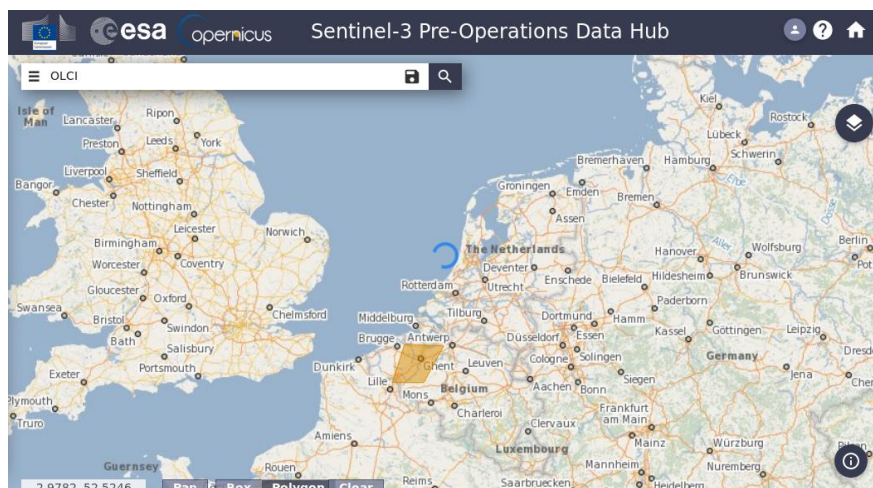


Figure 3 : Sentinel-3 Pre-ops Data Hub

Une fois la zone géographique et l'instrument choisis, l'archive est téléchargeable grâce à un lien direct. Le dossier en question comprend un nombre variable d'images au format NC. Ce format est bien particulier et nécessite des applications comme MatLab, NetCDF ou encore Ncview (Linux) pour être lu. Nous pouvons donc visualiser une parcelle grâce aux instruments de Sentinel 3 manuellement grâce à cette Pre-Ops. Cependant l'exploitation de ces images et l'extraction des indices caractéristiques n'est pas encore possible.

Contrairement au Sentinel 2 qui permettait de récupérer des images en jp2000, et donc d'utiliser le très intéressant module Rasterio de Python (entièrement dédié à l'exploitation des images de Sentinel2), le Sentinel 3 avec ses images au format NC rend le traitement d'image très compliqué. Nous avons exploité plusieurs pistes comme des calcul manuel sur les matrices de pixels mais ces essais n'ont pas été concluant. D'autres idées pour ces calculs grâce à MatLab pourraient peut-être donner suite.

Cependant, il est possible de récupérer l'image de DEM en altitude grâce à des manipulations plutôt simples. *Un tutoriel expliquant la démarche de récupération de l'indice a été réalisé en annexe ([Méthode de récupération du DEM](#)).*

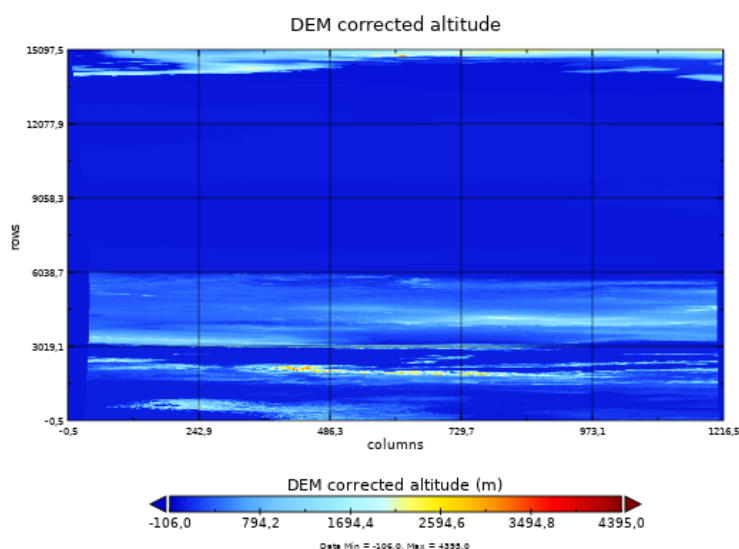


Figure 4 : image DEM

1.3. Logiciel indispensable pour l'extraction des indices

Pour récupérer les indices il est indispensable dans un premier temps, avant tout autre démarche de se configurer un espace de travail adapté :

- Avoir un environnement Linux
- Installer python
- Installer le module python rasterio pour pouvoir traiter les images en jp2.
- Installer node.js pour exécuter le serveur.
- Avoir une bonne connexion internet. Les téléchargements Copernicus sont très lourds, 1.5 Go en moyenne, ce qui peut s'avérer laborieux pour les tests à effectuer.

2. Les Interfaces web

Pour avoir des compléments d'information sur la création d'une interface web, voir annexe ([Le fonctionnement d'une interface web](#)).

2.1. La base de données

Nous avons créé une base de données pour enregistrer dans un dossier « clients » tous les clients, et dans un dossier « administrateurs » tous les administrateurs dans le but de gérer les dossiers de chaque client de façon optimale. Chaque client/administrateur comprendra dans son dossier un certain nombre d'informations qui lui sont propres.

Nous avons choisi pour la base de données d'utiliser MongoDB.

Mongodb est une application servant à la création et l'utilisation de bases de données codées sous le format JSON.

Pour avoir plus de renseignements, voir annexe ([Faire une base de données en Json](#)).

2.2. Modification de l'interface client

L'interface client a été réalisé au premier semestre par une autre équipe. Notre travail a été de faire des modifications.

Rappel sur l'interface client :

Elle est utilisée par les clients (les agriculteurs) pour renseigner les informations nécessaires afin de traiter leurs couverts végétaux.

En premier, l'interface demande à l'agriculteur ses données personnelles (nom, prénom, adresse...), ensuite elle lui demande des renseignements sur son champ. Et ensuite, elle lui demande des informations sur chacune de ses parcelles.

La première étape a été d'ordonner les pages de l'interface ; les regrouper par blocs, mettre des titres. Il a fallu ensuite ajouter des parties et en modifier d'autres.

Figure 5 : Interface client page 1



Figure 7 : interface client page 2

Figure 6 : interface client page 3

Vos couverts végétaux

Parcelle n°1

Informations sur vos cultures

Rotation

Votre culture précédente

Sélectionnez ici

Votre culture suivante

Sélectionnez ici

Date récolte précédente

jour/mois/année

Rendement

vous rendement (Q/hectares)

Cartographie de rendement *

Choisir un fichier | Aucun...oisi

Couvert Végétal

Type de semoirs pour les couverts végétaux

Mode de destruction du couvert

Problèmes de la parcelle

- ☐ Compactions
- ☐ Adventices
- ☐ Maladies
- ☐ Ravageurs

Fertilisation

Plan de fumure

Choisir un fichier | Aucun fichier choisi

Cartographie de Fertilisation

Choisir un fichier | Aucun fichier choisi

Analyses de Sol

Choisir un fichier | Aucun fichier choisi

Reliquats sortie d'hiver *

Sélect. fichiers | Aucun fichier choisi

Autres remarques

remarques

Finir!

* Ce champ n'est pas obligatoire

Figure 8 : interface client page 4

2.3. Interface web administrateur

L'interface administrateur permet à l'utilisateur de récupérer les données des clients ; choisir le client qu'il veut traiter et lui transmettre une réponse concernant les couverts végétaux les plus appropriés. L'utilisateur est une personne travaillant pour farmingBySatellite.

Fonctionnement de l'interface :

Page 1 : Connection

Lorsque l'utilisateur se connecte, il doit s'identifier en entrant son identifiant et son mot de passe. Si les données sont correctes, la deuxième page se chargera. Son identifiant et son mot de passe sont enregistrés dans la base de données. Ainsi il peut y avoir plusieurs utilisateurs.



Figure 9 : Page 1_ Interface administrateur

Page 2 : La recherche clients

Sur la deuxième page, l'utilisateur pourra rechercher un client. Soit en tapant son nom dans le bloc « rechercher un client » Soit en appuyant sur la liste des clients. Pour ce dernier, l'utilisateur verra apparaître la liste complète des clients. La liste fonctionne comme une boucle ; le dernier client à avoir renseigné ses informations sera déposé en bas de la liste. Lorsque le client en haut de la liste a été traité, il sera déposé en bas de la liste automatiquement (chaque client doit être traité une fois par an). Tous les clients sont enregistrés dans la base de données lors de leur inscription. Lorsque la deuxième page est chargée, le serveur demande à la base de données l'ensemble des clients. Si l'utilisateur a sélectionné un client (quel que soit la manière) une troisième page s'ouvre.



Figure 10 : Page 2_ Interface administrateur

Page 3 : Les données du client

Cette page est composée en plusieurs parties. La partie sur les informations personnelles du Client, sur les informations agricoles du client, sur les parcelles du client et sur les cartes du client. Chaque donnée est récupérée de la base de données (au lieu de « *ici on appelle le prenom du client* » on aura simplement son prénom). Cependant la base de données ne récupère pas les fichiers pour l'instant. Il faudra le compléter ultérieurement.

Informations personnelles du client

Nom : du client

Prénom : ici on appelle le prenom du client

Téléphone : ici on appelle le téléphone du client

Adresse mail : ici on appelle l'adresse mail du client

Rue : ici on appelle la rue du client

Code Postal : ici on appelle le code postal du client

Ville : ici on appelle la ville postal du client

Informations agricoles du client

Parcelaire

Nombre d'hectares : ici on appelle le nombre d'hectare du client

Nombre de parcelles : ici on appelle le nombre de parcelles du client

Situation de la parcelle : ici on appelle la situation des parcelles du client

Souscrit

Déclaration PAC client : ici on appelle la déclaration PAC du client

Parcelles

Parcelle 1

Parcelle 2

Parcelle 3

Parcelle 4

Parcelle 5

Cartes

Voir les cartes

Figure 11 : page 3_Interface administrateur

Lorsque l'utilisateur arrive sur cette page, le serveur demande à la base de données d'envoyer toutes les informations concernant le client sélectionné. Pour accéder à une parcelle en particulier il devra cliquer dessus pour la faire apparaître.

Parcelle 1

Rotation

Culture précédente : ici on appelle la culture précédente du client

Culture suivante : ici on appelle la culture suivante du client

Date récolte précédente : ici on appelle la date de récolte du client

Fertilisation

Plan de fumure : ici on appelle le plan de fumure du client

Cartographie de fertilisation : ici on appelle la cartographie de fertilisation du client

Analyse des sols : ici on appelle l'analyse du sol du client

Reliquats sortie d'hiver : ici on appelle les reliquatssorties d'hiver du client

Couvert Végétal

Type de semoir du couvert : ici on appelle le type de semoir du client

Mode de destruction du couvert : ici on appelle le mode de destruction du client

Problème de la parcelle : ici on appelle le probleme de la parcelle du client

Rendement

Le rendement (Q/hectares) : ici on appelle le rendement du client

Cartographie de rendement : ici on appelle la cartographie de rendement du client

Remarque

ici on appelle la remarque sur la parcelle du client

[retour](#) [fiche client](#)

Figure 12 : Page parcelle_Interface administrateur

Pour accéder aux cartes, il devra cliquer sur le bouton «voir les cartes».

Si l'utilisateur veut revenir en arrière il doit appuyer sur le bouton « retour » ; toutes les informations de la page seront supprimées et l'utilisateur pourra sélectionner un autre client.

Grace à cette page, l'utilisateur à toutes les données pour effectuer l'étude sur les couverts végétaux afin de déterminer la meilleure composition.

En bas de la page, après avoir pris connaissance de toutes les données du client, l'utilisateur peut rentrer la quantité d'azote, la quantité de carbone et la compaction du sol.

La méthode est expliquée dans la partie suivante : [Composition des couverts végétaux](#).

Figure 13 : Validation_Interface administrateur

Si l'utilisateur valide, le résultat apparaîtra contenant la composition du couvert végétal.

Figure 14 : Résultat_Interface administrateur

Enfin en appuyant sur le bouton « envoyer un mail », la cinquième page s'ouvre. L'utilisateur peut envoyer ses travaux réalisés au client.

Page 4 : envoie du mail de réponse

Sur cette page l'utilisateur peut envoyer un mail directement sur la page. Il peut écrire un message et joindre des fichiers. Il doit écrire son adresse mail, son mot de passe et l'adresse mail du client (l'adresse mail doit être une adresse mail GMAIL). Il n'a ensuite qu'à appuyer sur envoyer. Si l'utilisateur veut revenir en arrière, il doit juste appuyer sur le bouton « retour ». Les pièces jointes insérées seront automatiquement supprimées.

Les couverts vegetaux

Formulaire d'envoi de mail

Votre adresse mail

Votre mot de passe

Adresse mail du destinataire

L'objet du mail

Le contenu de votre mail

Joindre Envoyer

retour Voir client

Figure 15 : RéponseClient_Interface administrateur

3. Composition des couverts végétaux

Une étude préalable a été effectuée, elle est disponible en annexe ([Méthode de détermination des couverts végétaux](#)).

Nous avons créé un système composé d'entrée, de sortie et de règles reliant les entrées et les sorties. L'objectif est de déterminer les sorties seulement avec la connaissance des entrées.

3.1. Les entrées

Les entrées correspondent aux informations connues, récupérées des satellites. Il y a trois entrées à prendre en compte :

- La quantité d'azote
- La quantité de carbone
- La compaction

La quantité d'azote est comprise dans une échelle entre 3 et 130 KgN/ha. Nous décomposons cette échelle en 5 partie

- Partie 1 : « Très Peu » comprise entre 3 et 27 KgN/ha.
- Partie 2 : « peu » comprise entre 10 et 45 KgN/ha.
- Partie 3 : « normal » comprise entre 40 et 77 KgN/ha.
- Partie 4 : « beaucoup » comprise entre 75 et 115 KgN/ha.
- Partie 5 : « Très beaucoup » comprise entre 100 et 130.

La quantité de carbone est comprise sur une échelle entre 0 et 200 T/ha. Nous décomposons cette échelle en 5 partie

- Partie 1 : « Très Peu » comprise entre 0 et 8 T/ha.
- Partie 2 : « peu » comprise entre 5 et 35 T/ha.
- Partie 3 : « normal » comprise entre 30 et 65 T/ha.
- Partie 4 : « beaucoup » comprise entre 60 et 90 T/ha.
- Partie 5 : « Très beaucoup » comprise entre 80 et 200 T/ha.

On définit arbitrairement la compaction sur une échelle comprise entre 0 et 1. 0 correspond à un sol fort compacté et 1 correspond à un sol décompacté.

Nous décomposons cette échelle en 5 partie

- Partie 1 : « en surface » comprise entre 0 et 0,25.
- Partie 2 : « en profondeur » comprise entre 0,20 et 0,45.
- Partie 3 : « horizon compacté » comprise entre 0,30 et 0,65.
- Partie 4 : « normal » comprise entre 0,60 et 0,80.
- Partie 5 : « décompacté » comprise entre 0,78 et 1.

3.2. Les sorties

Les sorties sont toutes les espèces pouvant être insérer dans les couverts végétaux.

Févérole, Trèfle incarnat, Vesce, Lentille, Tournesol, Colza, Moutarde Blanche, Navette fourragère, Radis Chinois, Radis, Avoine de printemps, Avoine d'hiver.

Dans cette version on va considérer une quantité maximale du mélange pouvant être inséré (à définir) afin de ne pas étouffer le sol. On se limitera à trois espèces par configuration. Le mélange correspond aux différentes espèces formant le couvert végétal

On définit pour chaque espèce une échelle comprise entre 0% et 100% qui correspond à un pourcentage de la quantité maximal du mélange.

La répartition des échelles a été faite de façon arbitraire, l'objectif est de trouver la meilleure répartition possible pour avoir le couvert végétal le plus adapté.

3.3. Les règles

Il faut créer des règles qui seront les relations entre les entrées et les sorties.

Pour chaque cas possible en entrée, il y aura une solution choisie. Le but est de trouver la meilleure

On va séparer l'étude en cinq grands tableaux :

- 1.1. *Le premier tableau regroupera toutes les possibilités s'il y a une compaction dite « en surface ».*
- 1.2. *Le second tableau regroupera toutes les possibilités s'il y a une compaction dite « en profondeur ».*
- 1.3. *Le troisième tableau regroupera toutes les possibilités s'il y a une compaction dite « horizon compactés ».*
- 1.4. *Le quatrième tableau regroupera toutes les possibilités s'il y a une compaction dite « normal ».*
- 1.5. *Le cinquième tableau regroupera toutes les possibilités s'il y a une compaction dite « décompacté »*

3.1. Tableau regroupant toutes les possibilités s'il y a une compaction dite « en surface »

	quantité Azote	Très peu	peu	normal	beaucoup	très beaucoup	
quantité carbone							Compaction
très peu		lentille 50%; Radis 50%	lentille 42%; Radis 57%	lentille 33%; Radis 66%	Moutarde blanche 100%	Moutarde blanche 100%	en profondeur
peu		lentille 57%; Radis 42%	lentille 50%; Radis 50%	lentille 40%; Radis 60%	Moutarde blanche 100%	Moutarde blanche 100%	en profondeur
normal		lentille 60%; Radis 40%	lentille 60%; Radis 40%	lentille 50%; Radis 50%	Navette 60%; Radis 40%	Navette 66%; Radis 33%	en profondeur
beaucoup		lentille 66%; Radis 33%	lentille 60%; Radis 40%	lentille 50%; Radis 50%	Navette 60%; Radis 40%	Navette 66%; Radis 33%	en profondeur
très beaucoup		lentille 80%; Radis 20%	lentille 75%; Radis 25%	lentille 75%; Radis 25%	Navette 75%; Radis 25%	Navette 80%; Radis 20%	en profondeur

Tableau 1 : Tableau regroupant toutes les possibilités s'il y a une compaction dite « en surface »

Explication :

L'avoine d'hiver permet de décompacter le sol et de rendre la terre plus riche en carbone. Il faut réussir à trouver le bon taux pour à la fois décompacter la terre et obtenir le bon taux de carbone dans le sol. Lorsqu'il y a beaucoup de carbone dans le sol (Très Beaucoup) il faut ajouter de l'avoine d'hiver mais pas trop pour ne pas surcharger le sol en carbone.

Les lentilles permettent d'augmenter le taux d'azote dans le sol et le Navette permet de le réduire d'où beaucoup de lentilles lorsque le taux de Navette est bas et beaucoup de Navette lorsque le taux d'azote est élevé.

Tous les tableaux sont présentés en annexe ([Tableaux des relations](#)).

Ceci est la première version, les relations peuvent être amélioré. L'avantage de cette version est qu'on ne dépasse pas la quantité maximale du mélange, il n'y aura pas de phénomène d'étouffement. En revanche les pourcentages choisis ne sont peut-être pas les meilleurs. De plus, il faudrait étudier si les espèces sont les plus adaptées ou s'il faudrait en choisir des autres.

4. Présentation du livrable

Le livrable comprend l'ensemble de notre réalisation. Il y a donc :

- La récupération des indices sentinel 2
- L'interface client
- L'interface administrateur
- La base de données

L'ensemble de nos travaux se trouve dans un dossier sur GitHub (service web d'hébergement et de gestion de développement de logiciels).

Un tutoriel de GitHub existe sur open Classroom pour apprendre à l'utiliser : [tutoriel GitHub](#)

Notre réalisation est accessible par tous sur GitHub : [FarmingBySat2](#)

2e partie du projet Edit

Add topics

40 commits 1 branch 0 releases 3 contributors

Branch: master New pull request Create new file Upload files Find file Clone or download

louisanthonioz Update README.md Latest commit 325b13a 3 minutes ago

Basededonnees	regroupement des deux interface + de la base de données	14 days ago
Récup_indice2	Formule code python pour récupération de l'indice MCARI/MTVI2	3 days ago
infoClient	interface admi page informationclient	8 days ago
interface client	Update README.md	18 minutes ago
interfaceAdministrateur	regroupement des deux interface + de la base de données	14 days ago
interface_admin(real)	Update README.md	7 minutes ago
.gitignore	ajout planning réel	7 days ago
Plan de soutenance.docx	interface admin: ajout résultat client	a day ago
README.md	Update README.md	3 minutes ago
Rapport Projet-Farming by satellite-F...	interface admin: ajout résultat client	a day ago
~\$an de soutenance.docx	interface admin: ajout résultat client	a day ago

README.md

Figure 16 : : Notre dossier GitHub FarmingBySat2

README.md

FarmingBySat2

2e partie du projet

Bienvenue dans farmingBySat2!

Dans ce GitHub vous trouvez tous les documents et les codes pour continuer le projet.br/>

Pour utiliser les documents et les codes :

inscrivez vous sur la page web <https://github.com/>

installer l'application GitHub

cloner le dossier GitHub FarmingBySat2

Vous êtes maintenant en possession du dossier FarmingBySat2 sur votre ordinateur

Dans chaque dossier, un fichier README explique la présence de chacun des sous dossiers et des fichiers et leur fonctionnement.

Figure 17 : README principal du GitHub

```
/*on récupère les éléments du HTML avec document.getElementById('id'); */

/*fonction qui test la compatibilité du mdp écrit par l'admi
la fonction est déclenché lorsque l'admi appuie sur le bouton OK
à coté du block mot de passe
.style.display= "block" permet d'afficher l'élément
.style.display= "none" permet de cacher l'élément
il faut dans un premier temps les afficher
pour cacher un élément par défaut dans le fichier code css:" display:none" */

function goSecondStep() {
    /* on crée une variable test qui est la valeur correcte
    du mot de passe et la variable administrateur la valeur correcte de id*/
    mdp = document.getElementById("password").value;
    id = document.getElementById("identifiant").value;
    test = 'test';
    idadmin = 'administrateur';

    if (mdp == test && id == idadmin) {
        document.getElementById("secondStep").style.display = "block";
        document.getElementById("firstStep").style.display = "none";
        faireLaListe();
    }
    else {
        alert("mot de passe ou identifiant incorrect");
    }
}

/* rechercheNouveau: fonction qui permet de passer sur la 4e page
si on click sur ok lors de la recherche nouveau client
de plus elle fait apparaître le nom et prénom du client
en haut à gauche. Si l'entree est vide la fonction renvoie
un message d'erreur */

function rechercheNouveau(client) {
    nouveauClient = document.getElementById(client).value;

    if (nouveauClient != '') {
```

Chaque code est expliqué pour permettre une lecture facile.

Certaines explications sont sous forme de tutoriel pour comprendre le cheminement du code

```
function lesCartesNouveau() {

    /*la première partie permet de créer un lien qui permettra d'agrandir la carte
    lorsqu'on appuiera sur "afficher en grand" */

    /*Pour créer un lien en javascript:
    1. on crée un élément lien caractérisé par la borne <a> avec document.createElement(element)
    2. on lui donne un id : element.id='nom id'
    3. on lui donne son href (l'adresse du lien) : element.href='direction du lien'
    4. on utilise appendChild pour insérer l'élément enfant (élément a) qu'on vient de créer
    dans son élément parent (l'élément qui le recouvre)
    5. on crée le text avec document.createTextNode('text') ce sera le text sur lequel on appuie pour déclencher le lien
    6. on insère se text dans l'élément <a> grace à la fonction appendChild */

    nouveauLien = document.createElement("a");
    nouveauLien.id = "nouveauLien";
    nouveauLien.href = "afficherCarte";
    document.getElementById("cartes").appendChild(nouveauLien);

    nouveauLienText = document.createTextNode("afficher en grand");
    nouveauLien.appendChild(nouveauLienText);
```

Figure 19 : Tutoriel : créer un lien HTML avec javascript

CONCLUSION GENERALE

Les travaux réalisés au cours de ce projet ont permis à notre client de se rapprocher de son l'objectif de l'intrapreneuriat. Aujourd'hui, une interface client est prête à recevoir les renseignements du client. Une base de données a été créé pour récupérer toutes les données et les transmettre sur l'interface administrateur. Sur cette dernière interface, l'utilisateur pourra faire une étude sur les couverts végétaux grâce aux informations qu'il récupérera dessus. Et il pourra envoyer un bilan au client. Les indices du satellite Sentinel 2 peuvent être récupérés automatiquement et transmis sur l'interface administrateur et la carte du DEM est accessible manuellement. Tous les travaux sont accessibles sur le dossier GitHub [*FarmingBySat2*](#). Cependant, pour que la partie informatique soit entièrement opérationnelle, il faudrait réussir à récupérer les indices du satellite Sentinel 3 automatiquement. De plus, la base de données doit encore être complétée pour permettre la récupération des fichiers.

BIBLIOGRAPHIE

Site internet : *openclassroom*

<https://openclassrooms.com/dashboard>

Site internet Copernicus

<https://scihub.copernicus.eu>

Rapport : Farming By Satellite 2674/Semestre 1/2017-2018

Rapport : Farming By Satellite/Partie informatique/Semestre 1/2017-2018

TABLE DES FIGURES

Figure 1 : Formule MCARI/MTVI2	13
Figure 2 : Image NDVI	14
Figure 3 : Sentinel-3 Pre-ops Data Hub	15
Figure 4 : image DEM	15
Figure 5 : Interface client page 1	17
Figure 7 : interface client page 3.....	17
Figure 6 : interface client page 2.....	17
Figure 8 : interface client page 4.....	18
Figure 9 : Page 1_ Interface administrateur	19
Figure 10 : Page 2_ Interface administrateur	19
Figure 11 : page 3_ Interface administrateur	20
Figure 12 : Page parcelle_ Interface administrateur	21
Figure 13 : Validation_ Interface administrateur.....	22
Figure 14 : Résultat_ Interface administrateur.....	22
Figure 15 : RéponseClient_ Interface administrateur.....	23
Figure 16 : : Notre dossier GitHub FarmingBySat2	27
Figure 17 : README principal du GitHub	28
Figure 18 : Partie du code fomulaire.js	28
Figure 19 : Tutoriel : créer un lien HTML avec javascript.....	28
Figure 20 : Taux de carbone dans le sol	38
Figure 21: logo HTML et CSS	43
Figure 22: traduction du code HTML par le navigateur web	43
Figure 23: logo non officiel	44
Figure 24: schéma de lecture du langage Javascript.....	45
Figure 25: logo officiel.....	45
Figure 26: schéma de lecture du langage Node.js	46
Figure 27: modèle non-bloquant	47
Figure 28: communication asynchrone.....	48
Figure 29: communication synchrone	48
Figure 30 : format d'une base de données sous Json	50
Figure 31 : Interface client_Page d'inscription	51

Figure 32 : Console Mongoddb_résultat 1.....	51
Figure 33 : Schéma d'explication d'une requête Mongoddb	52
Figure 34 : Requête Mongoddb	52
Figure 35 : Console Mongoddb_résultat 2.....	53
Figure 36 : Timesheet.....	56
Figure 37 : Bloc informations générales-avant modification	57
Figure 38 : Bloc information générale- après modification	57
Figure 39 : Bloc informations sur vos cultures-après modification	58
Figure 40 : Bloc informations sur vos cultures-avant modification	58
Figure 41 : Bloc Fertilisation-avant modification	58
Figure 42 : Bloc Fertilisation-après modification	58
Figure 43 : Bloc rendement-après modification	59
Figure 44 : Bloc rendement-avant modification	59
Figure 45 : Bloc Autres remarques- après modification	59
Figure 46 : Bloc couvert Végétal-après modification.....	60
Figure 47 : Block couvert Végétal complet-après modification.....	60
Figure 48 : Schéma de principe du script bash	61
Figure 49 : Coordonnées UTM	62
Figure 50 : Outil Polygon.....	64
Figure 51 : Lancer la recherche	64
Figure 52 : identification	65
Figure 53 : Archive de la recherche	65
Figure 54 : Archive S3A_OL_2_LRR.....	66
Figure 55 : lecture du fichier « DEM altitude » en 2D Y/X	66
Figure 56 : DEM altitude » en 2D Y/X.....	67
Figure 57 : Carte DEM	67

TABLE DES TABLEAUX

Tableau 1 : Tableau regroupant toutes les possibilités s'il y a une compaction dite « en surface »	26
Tableau 2 : Espèces en fonction de la quantité d'azote dans le sol.....	36
Tableau 3 : Espèces en fonction de la quantité d'azote dans le sol (2)	37
Tableau 4 : Espèces en fonction de la quantité de carbone	39
Tableau 5 : Espèces en fonction de la compaction	39
<i>Tableau 6 : Tableau regroupant toutes les possibilités s'il y a une compaction dite « en surface »</i>	<i>40</i>
<i>Tableau 7 : Tableau regroupant toutes les possibilités s'il y a une compaction dite « en profondeur »</i>	<i>41</i>
<i>Tableau 8 : Tableau regroupant toutes les possibilités s'il y a une compaction dite « horizon compacté »</i>	<i>41</i>
<i>Tableau 9 : Tableau regroupant toutes les possibilités s'il y a une compaction dite « normale »</i>	<i>42</i>
<i>Tableau 10 : Tableau regroupant toutes les possibilités s'il y a une compaction dite « décompacté »</i>	<i>42</i>
Tableau 11 : Planning prévisionnel	54
Tableau 12: Planning réel.....	55

TABLE DES LIENS

Lien 1 : [tutoriel GitHub](#)

Lien 2 : [FarmingBySat2](#)

Lien 3 : [tutoriel HTML/CSS](#)

Lien 4 : [tutoriel : apprendre à coder avec javascript](#)

Lien 5 : [tutoriel Node.js](#)

ANNEXES

1. Cahier des charges

Réalisé par Marie Marion

CAHIER DES CHARGES 2018

L'objectif de ce projet CoElab est d'obtenir l'ensemble des paramètres d'imagerie nécessaires, à l'analyse du sol. Ces images devront être traitées, et leur extraction automatisée sur un site internet, qui permettra de centraliser l'information.

Une fois les images obtenues celles-ci serviront à réaliser des modèles de prédiction de certaines propriétés du sol comme le Carbone et l'Azote. Cette étape est donc fondamentale dans l'avancée du projet, car sans les données, la création de modèles prévue pour septembre 2018 n'est pas réalisable.

1.1. Extraction des images

1.1.1. Obtention du MCARI et MTVI2

Ces deux indicateurs sont obtenus à partir d'outils embarqués à bord de Sentinel 2. La combinaison de ces deux données permet d'obtenir le taux de chlorophylle de la plante, qui lui-même donne une information sur la quantité d'azote dans celle-ci et donc indirectement dans le sol.

Il s'agit donc d'extraire ces deux indicateurs et de parvenir comme pour le calcul du NDVI à superposer leurs images en divisant le MCARI/ MTVI2. Le détail du calcul est détaillé p55 du rapport Co-Elab 2017, *Etude de faisabilité du montage d'une startup autour des thèmes de la télédétection et de l'agriculture*.

1.1.2. Obtention d'indicateur du COS

Le COS se calcule à partir d'un certain nombre de valeurs : L'indice de luminosité (Bi), l'indice de végétation (Gi), l'indice d'humidité (Wi), le NDVI, l'indice de végétation et de températures (VTCI), l'altitude (DEM), la pente et les composants topographiques (CTI). Le calcul de ces différents indices sont détaillés de la page 54 à 57, du rapport Co-Elab 2017, *Etude de faisabilité du montage d'une startup autour des thèmes de la télédétection et de l'agriculture*.

Les 4 premiers indicateurs ont été préalablement obtenus à partir de Sentinel 2. L'obtention des autres indicateurs fait justement l'objet de ce projet.

Comme précédemment, il s'agit d'apporter comme rendu final, des grandeurs utilisables lors de l'étape de modélisation qui aura lieu en septembre. L'extraction doit donc être automatisée, facilement accessible, et directement utilisable (superposition des images extraites suivant les calculs du rapport, pour obtenir la valeur d'intérêt finale).

1.2. Création d'un site interne

1.2.1 Interface client :

L'interface Client du site internet a déjà été réalisée, néanmoins celle-ci nécessite des ajustements pour être plus proche des besoins terrains.

- **Informations parcelles > Intrants** : Etre plus précis en demandant à l'agriculteur d'insérer son plan de fumure (document administratif)
- **Informations parcelles > Vos bilans azotés/hydriques** : remplacer par vos reliquats sortie d'hiver.
- **Vos couverts végétaux** : Plus de précision il faut demander à l'agriculteur :
 1. Quelle est sa date prévisionnelle de récolte de la culture précédente,
 2. Son type de semoirs pour les couverts végétaux avec un menu déroulant cf.p.73. (Semoirs à céréales, semoirs à la volée, semoirs de semis direct, etc.),
 3. Mode de destruction du couvert cf.p.62. (Chimique, broyage, labour, gel, déchaumage),
 4. Les problématiques de la parcelle (adventices, maladies, ravageurs, compaction), l'agriculteur peut sélectionner plusieurs choix et pour chacun un autre menu déroulant s'affiche avec le nom des maladies ou des ravageurs à sélectionner par exemple.
- **Autres remarques sur cette parcelle** (à ajouter)

1.2.2. Interface administrateur :

C'est l'interface la plus complexe, car celle-ci doit permettre de visualiser l'ensemble des informations et cartes fournies par l'agriculteur, mais également les données d'imageries obtenues par satellites et ou drones.

- Un recoupement entre les cartes de déclaration PAC et les cartes d'imagerie obtenues doit pouvoir être réalisé. Plus concrètement cela signifie que les cartes obtenues par imagerie doivent pouvoir être découpées en fonction des contours exactes géolocalisés de la déclaration PAC.
- Les différentes cartes doivent être facilement superposables via leurs coordonnées GPS. (Le logiciel de SIG ArcMap, peut être utile dans le traitement de ces différentes cartes, piste à explorer).

In fine il pourrait être intéressant de trouver une relation mathématique capable de faire le lien entre les différentes cartes pour effectuer le calcul et la préconisation de couverts végétaux automatiquement. Le conseil final serait alors bien évidemment légèrement adapté à la réalité du terrain par un agronome. Pour l'instant nous ne pouvons pas encore établir un conseil clair entre les données mesurées par télédétection et un conseil de densité de semis des couverts végétaux. Néanmoins il serait intéressant de développer cette relation reliant les différentes cartes obtenues, pour donner un conseil fictif, dont les paramètres seront simplement ajustés lorsque nous passerons à la phase expérimentale. La méthode Fuzzy Logique sur MatLab a été évoquée et semble très prometteuse. A étudier !!

1.2.3. Planification de rdv

Afin de bien identifier les corrections, améliorations, et problématiques qui surviendront au cours du projet il sera nécessaire de réaliser des points d'avancements réguliers. Pour le site internet par exemple il semble constructif de prévoir 2 rendez-vous (**dates à définir en fonction du rétroplanning d'avancement**). Un premier pour présenter et valider une maquette du site final. L'objectif est de dresser un tableau **rapide** mais exhaustif des fonctionnalités attendues, pour ne pas avoir de surprises après sa réalisation. Un deuxième, après réalisation du site pour le tester ensemble et déterminer les améliorations à apporter.

1.2.4. En bref ...

Il est demandé dans l'ensemble du projet de faire preuve d'esprit critique et d'initiative quand au développement de ces outils. Pour cela il faut bien comprendre le contexte et sujet du projet.

Il s'agit de développer des outils de détection par imagerie satellite des sols et des cultures afin de recueillir des informations, qui seront utilisées pour préconiser un mélange de couverts végétaux. Des contraintes d'ordre technique s'appliquent donc à cette méthode comme la résolution (1km de résolution spatiale n'est pas intéressant pour une parcelle de quelques hectares par exemple), l'acquisition des données (les satellites doivent passer suffisamment régulièrement pour acquérir des images quand les conditions météo le permettent), l'automatisation de l'extraction des images (pour limiter le travail de manutention supplémentaire, lourd pour une jeune startup), etc.

Afin de réaliser ces travaux, il ne faut pas hésiter à prendre appui sur les connaissances des enseignants chercheurs du Groupe Yncréa, qui chacun dans leurs spécialités peuvent apporter leur expertise. Bertrand Vandoorne (bertrand.vandoorne@yncrea.fr), notamment, responsable du pôle agricole de l'ISA, suit le projet depuis son initiation et pourra apporter son expertise d'agronome, et sa connaissance du projet.

Ne pas hésiter également à me faire remonter des questions, je tacherai d'être la plus réactive possible.

Bon courage !!!

Revenir à la [note de clarification](#)

2. Méthode de détermination des Couverts végétaux

2.1. Démarche mise en place

L'objectif est de créer une fonction permettant de déduire les données de sortie (c'est à dire le type d'espèce à mettre et la densité) en utilisant les données d'entrées provenant du satellite (taux de carbone, d'azote et la compaction du sol).

Pour cela, nous avons besoin d'avoir un tableau reliant les entrées et les sorties.

Dans un premier temps, il faut avoir les relations permettant de savoir les espèces à mettre en fonction de la quantité d'azote ; les espèces à mettre en fonction de la quantité de carbone ; les espèces à mettre en fonction de la compaction.

2.2. Etudes des espèces à insérer et leur densité

2.2.1. Fourchette d'azote dans le sol

La mesure par satellite devrait nous permettre d'établir une corrélation avec la quantité d'azote de la culture. Il faudra ensuite réaliser une relation avec la quantité d'azote présente dans le sol, c'est-à-dire la quantité d'azote minérale (NH₄ et NO₃).

Or les reliquats sortis d'hiver permettent d'établir cette quantité d'azote minérale disponible dans le sol et permettront de calibrer le modèle, à terme.

En moyenne celui-ci est donc de 25kgN/ha, avec une valeur basse de 3kgN/ha et une valeur haute de 130kgN/ha.

2.2.2. Légumineuses pour pallier la quantité d'azote

Reprenons la fourchette d'azote minérale obtenue précédemment. Il faut mettre 100% de légumineuse quand on a 3 kgN/ha, inversement il faudra en mettre 0% pour 130kgN/ha.

Espèce	Apport en (KgN /ha)	Qté d'azote minérale dans le sol (kgN/ha)					
		3	25	50	75	100	125
		100%					0%
Féverole	6	200	150	100	50	20	0
Trèfle incarnat	8	15	11.25	7.5	3.75	1.5	0
Vesce	7	60	45	30	15	6	0
Lentille	6	40	30	20	10	4	0

Tableau 2 : Espèces en fonction de la quantité d'azote dans le sol

2.2.3. CIPAN pour limiter le lessivage de l'azote

Cette partie est complémentaire de la précédente puisque les légumineuses (plantes qui apportent l'azote de l'air) et les CIPAN (Culture intermédiaire piège à nitrate = Plante qui capte l'azote en excès dans le sol), agissent sur même paramètre : la quantité d'azote dans le sol.

Ils agissent de manière opposée puisque l'un captera l'azote en excès et l'autre apportera de l'azote dans les cas déficitaires.

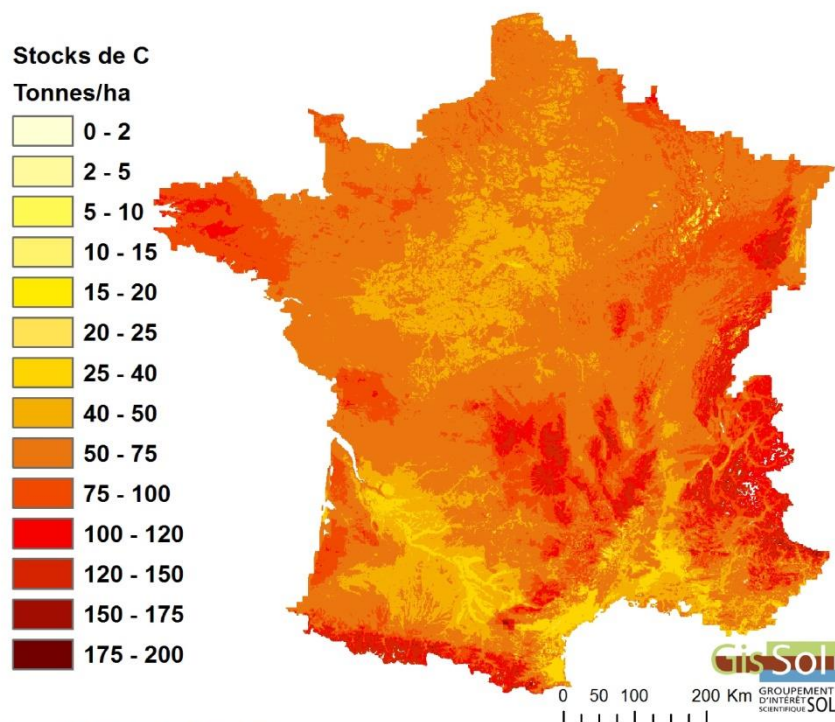
Espèce	Qté d'azote minérale dans le sol (kgN/ha)					
	3	25	50	75	100	125
	0%					100%
Tournesol	0	20	25	30	35	40
Colza	0	3	3.5	4	4.5	6
Moutarde blanche	0	3.5	4.3	5.25	6.1	7
Navette fourragère	0	2.5	3.1	3.75	4.4	5
Radis chinois	0	4.5	5.6	6.75	7.87	9

Tableau 3 : Espèces en fonction de la quantité d'azote dans le sol (2)

2.2.4. Fourchette de carbone

Grace à l'imagerie satellite nous devrions obtenir le COS du sol, c'est-à-dire le carbone organique du sol. Celui-ci a été mesuré par l'INRA, dans la carte ci-dessous. « Les stocks de carbone faibles à moyens (40-50 t/ha) sont caractéristiques des sols des grandes plaines de culture intensive de France ainsi que des sols limoneux comme par exemple le grand Bassin parisien, une partie du Bassin aquitain, le Toulousain et le sillon rhodanien. Les stocks de carbone moyennement élevés (50-70 t/ha) sont caractéristiques des grandes régions forestières ou fourragères de France (Bretagne, Est, Massif central, Normandie) et les stocks de carbone les plus élevés correspondent à des situations climatiques (sols situés en altitude), minéralogiques (sols volcaniques du Massif central) ou hydriques extrêmes (marais de l'Ouest, delta du Rhône). »

Estimation des stocks de carbone organique de 0 à 30 cm de profondeur en France métropolitaine hors Corse



Source: Gis Sol, IGCS-RMQS, Inra 2017.

Figure 20 : Taux de carbone dans le sol

2.2.5. Augmenter la quantité de Carbone grâce aux couverts végétaux

Sélection d'espèces à partir du rapport CoElab 1 sélectionnées grâce à la méthode MERCI. Ce sont les espèces qui apportent le plus de Carbone.

Espèce	T MS/ha de carbone	Qté de carbone organique dans le sol (t/ha)					
		0	25	50	75	100	200
		100%				0%	0%
Radis	30	9	8	7	5	2	2
Trèfle incarnat	29	15	13	11.6	8.33	3.33	3.33
Moutarde blanche	24	7	6.22	5.44	3.8	1.5	1.5
Vesce	22	60	53	46	33	13	13
Avoine de printemps	22	60	53	46	33	13	13
Avoine d'hiver	19	60	53	46	33	13	13

Tableau 4 : Espèces en fonction de la quantité de carbone

2.2.6. Fourchette de compaction du sol

Cette partie nécessiterait de l'expérimentation. Néanmoins la donnée que l'on obtiendra ne sera certainement pas sous forme d'une valeur quantifiable mais plutôt une indication : compacté surface, compacté en profondeur, normal, décompacté.

2.2.7. Les plantes pivots et racines fasciculées.

Ces plantes permettent de décompacter les sols.

	Densité max	Compacté en surface	Compacté en profondeur	Tt Horizons compactés	Normal	Décompacté
Radis	30	0	30	15	5	0
Moutarde blanche	24	0	24	12	3	0
Avoine de printemps	22	22	0	11	2	0
Avoine d'hiver	19	19	0	9.5	1	0

Tableau 5 : Espèces en fonction de la compaction

Il s'agira le plus souvent de prendre une espèce dans chacun de ces 4 tableaux, en fonction de paramètres agronomiques extérieurs. Cependant si la densité globale du couvert est trop élevée, il se produira un phénomène d'étouffement, qui empêchera le développement des plantes. Il y a donc une dose totale maximale sur l'ensemble du mélange. Il faudrait parler en % de cette densité de couverture maximale.

De plus on observe bien que les densités de semis varient énormément en fonction des espèces. Cela s'explique simplement par le fait que celle-ci s'expriment en kg/ha. Ainsi les graines de féverole très grosses et très lourdes comparées à la phacélie qui mesure quelques millimètres. Elle aura donc pour le même nombre de plantes au m², une densité de semis largement supérieure.

Il serait intéressant donc de prendre en compte non pas la densité de semis mais la densité de peuplement après la levée, avec un lien ensuite vers la densité de semis. Ainsi nous pourrions déterminer une densité maximale pour le couvert, et en fonction de chaque propriété cela correspondrait à un % de cette densité.

Autre particularité une plante peut avoir plusieurs propriétés et donc être sélectionnée dans plusieurs des tableaux ci-dessus. Il n'est pas nécessaire alors d'additionner ses densités de semis, car elles se combinent.

2.3. Tableaux des relations

	quantité Azote	Très peu	peu	normal	beaucoup	très beaucoup	
quantité carbone							Compaction
très peu		lentille 50%; Radis 50%	lentille 42%; Radis 57%	lentille 33%; Radis 66%	Moutarde blanche 100%	Moutarde blanche 100%	en profondeur
peu		lentille 57%; Radis 42%	lentille 50%; Radis 50%	lentille 40%; Radis 60%	Moutarde blanche 100%	Moutarde blanche 100%	en profondeur
normal		lentille 60%; Radis 40%	lentille 60%; Radis 40%	lentille 50%; Radis 50%	Navette 60%; Radis 40%	Navette 66%; Radis 33%	en profondeur
beaucoup		lentille 66%; Radis 33%	lentille 60%; Radis 40%	lentille 50%; Radis 50%	Navette 60%; Radis 40%	Navette 66%; Radis 33%	en profondeur
très beaucoup		lentille 80%; Radis 20%	lentille 75%; Radis 25%	lentille 75%; Radis 25%	Navette 75%; Radis 25%	Navette 80%; Radis 20%	en profondeur

Tableau 6 Tableau regroupant toutes les possibilités s'il y a une compaction dite « en surface »

Tableau 7 : Tableau regroupant toutes les possibilités s'il y a une compaction dite « en profondeur »

	quantité Azote	Très peu	peu	normal	beaucoup	très beaucoup	
quantité carbone							Compaction
très peu		lentille 50%; Radis 50%	lentille 42%; Radis 57%	lentille 33%; Radis 66%	Moutarde blanche 100%	Moutarde blanche 100%	en profondeur
peu		lentille 57%; Radis 42%	lentille 50%; Radis 50%	lentille 40%; Radis 60%	Moutarde blanche 100%	Moutarde blanche 100%	en profondeur
normal		lentille 60%; Radis 40%	lentille 60%; Radis 40%	lentille 50%; Radis 50%	Navette 60%; Radis 40%	Navette 60%; Radis 33%	en profondeur
beaucoup		lentille 60%; Radis 33%	lentille 60%; Radis 40%	lentille 50%; Radis 50%	Navette 60%; Radis 40%	Navette 66%; Radis 33%	en profondeur
très beaucoup		lentille 80%; Radis 20%	lentille 75%; Radis 25%	lentille 75%; Radis 25%	Navette 75%; Radis 25%	Navette 80%; Radis 20%	en profondeur

	quantité Azote	Très peu	peu	normal	beaucoup	très beaucoup	
quantité carbone							Compaction
très peu		Radis 33%; Trèfle incarnat 33%; Féverole 33%	Radis 40%; Trèfle incarnat 40%; Féverole 20%	Radis 50%; Trèfle incarnat 50%	Radis 28%; Trèfle incarnat 28%; Navette 42%	Radis 33%; Trèfle incarnat 33%; Navette 66%	horizon compactés
peu		Radis 33%; Trèfle incarnat 16%; Féverole 50%	Radis 40%; Trèfle incarnat 20%; Féverole 40%	Radis 50%; Trèfle incarnat 25%; Féverole 25%	Radis 33%; Trèfle incarnat 16%; Navette 50%	Radis 28%; Trèfle incarnat 14%; Navette 57%	horizon compactés
normal		Radis 33%; Féverole 66%	Radis 40%; Féverole 60%	Radis 50%; Féverole 50%	Radis 50%; Navette 50%	Radis 40%; Navette 60%	horizon compactés
beaucoup		Radis 23%; Féverole 76%	Radis 28%; Féverole 71%	Radis 37%; Féverole 62%	Radis 37%; Navette 62%	Radis 28%; Navette 71%	horizon compactés
très beaucoup		Radis 16%; Féverole 83%	Radis 21%; Féverole 78%	Radis 28%; Féverole 71%	Radis 28%; Navette 71%	Radis 21%; Navette 78%	horizon compactés

Tableau 8 : Tableau regroupant toutes les possibilités s'il y a une compaction dite « horizon compacté »

Tableau 9 : Tableau regroupant toutes les possibilités s'il y a une compaction dite « normale »

	quantité Azote	Très peu	peu	normal	beaucoup	très beaucoup	
quantité carbone							Compaction
très peu		Radis 16%; Trèfle incarnat 83%	Radis 16%; Trèfle incarnat 83%	Radis 11%; Trèfle incarnat 58%; Navette 29%	Radis 10%; Trèfle incarnat 51%; Navette 38%	Radis 9%; Trèfle incarnat 45%; Navette 45%	Nomal
peu		Radis 16%; Trèfle incarnat 62%; lentille 16%	Radis 21%; Trèfle incarnat 78%	Radis 13%; Trèfle incarnat 51%; Navette 34%	Radis 11%; Trèfle incarnat 44%; Navette 44%	Radis 10%; Trèfle incarnat 38%; Navette 51%	Nomal
normal		Radis 16%; Trèfle incarnat 41%; lentille 41%	Radis 21%; Trèfle incarnat 52%; lentille 26%	Radis 28%; Trèfle incarnat 71%	Radis 16%; Trèfle incarnat 41%; Navette 41%	Radis 13%; Trèfle incarnat 34%; Navette 51%	Nomal
beaucoup		Radis 16%; Trèfle incarnat 20%; lentille 62%	Radis 21%; Trèfle incarnat 26%; lentille 52%	Radis 28%; Trèfle incarnat 35%; lentille 35%	Radis 21%; Trèfle incarnat 26%; Navette 52%	Radis 16%; Trèfle incarnat 20%; Navette 62%	Nomal
très beaucoup		Radis 16%; lentille 83%	Radis 21%; lentille 78%	Radis 28%; lentille 71%	Radis 28%; Navette 71%	Radis 21%; navette 78%	Nomal

	quantité Azote	Très peu	peu	normal	beaucoup	très beaucoup	
quantité carbone							Compaction
très peu		Trèfle incarnat 100%	Trèfle incarnat 80%; Navette 20%	Trèfle incarnat 66%; Navette 33%	Trèfle incarnat 57%; Navette 42%	Trèfle incarnat 50%; Navette 50%	Décompact é
peu		Trèfle incarnat 75%; lentille 25%	Trèfle incarnat 100%	Trèfle incarnat 75%; Navette 25%	Trèfle incarnat 60%; Navette 40%	Trèfle incarnat 42%; Navette 57%	Décompact é
normal		Trèfle incarnat 50%; lentille 50%	Trèfle incarnat 66%; lentille 33%	Trèfle incarnat 100%	Trèfle incarnat 50%; Navette 50%	Trèfle incarnat 40%; Navette 60%	Décompact é
beaucoup		Trèfle incarnat 25%; lentille 75%	Trèfle incarnat 33%; lentille 66%	Trèfle incarnat 50%; lentille 50%	Trèfle incarnat 33%; Navette 66%	Trèfle incarnat 25%; Navette 75%	Décompact é
très beaucoup		lentille 100%	lentille 100%	lentille 100%	Navette 100%	Navette 100%	Décompact é

Tableau 10 : Tableau regroupant toutes les possibilités s'il y a une compaction dite « décompacté »

Revenir à la [composition des couverts végétaux](#)

3. Le fonctionnement d'une interface web

3.1. Utilisation de HTML et CSS

Les langages HTML et CSS sont à la base du fonctionnement de tous les sites web. Quand vous consultez un site avec votre navigateur, il faut savoir que, en coulisses, des rouages s'activent pour permettre au site web de s'afficher. L'ordinateur se base sur ce qu'on lui a expliqué en HTML et CSS pour savoir ce qu'il doit afficher, comme le montre la figure suivante.



Figure 21: logo HTML et CSS

HTML et CSS sont deux « langues » qu'il faut savoir parler pour créer des sites web. C'est le navigateur web qui fera la traduction entre ces langages informatiques et ce que vous verrez s'afficher à l'écran.

Pour créer un site web, on doit donner des instructions à l'ordinateur. Il ne suffit pas simplement de taper le texte qui devra figurer dans le site (comme on le ferait dans un traitement de texte Word, par exemple), il faut aussi indiquer où placer ce texte, insérer des images, faire des liens entre les pages, etc.

Langages
HTML et CSS



Traduction par
l'ordinateur

Résultat visible
à l'écran



Du HTML à l'écran

Figure 22: traduction du code HTML par le navigateur web

- HTML (HyperText Markup Language) :

Son rôle est de gérer et organiser le contenu. C'est donc en HTML que vous écrirez ce qui doit être affiché sur la page : du texte, des liens, des images... Vous direz par exemple : « Ceci est mon titre, ceci est mon menu, voici le texte principal de la page, voici une image à afficher, etc. ».

- CSS (Cascading Style Sheets, aussi appelées Feuilles de style) :

Le rôle du CSS est de gérer l'apparence de la page web (agencement, positionnement, décoration, couleurs, taille du texte...). Ce langage est venu compléter le HTML en 1996.

Le HTML définit le contenu (comme vous pouvez le voir, c'est brut de décoffrage !). Le CSS permet, lui, d'arranger le contenu et de définir la présentation : couleurs, image de fond, marges, taille du texte...

Pour plus d'informations, des tutoriels existent. Nous avons réalisé un tutoriel de openclassroom : [tutoriel HTML/CSS](#)

3.2. Utilisation de Javascript

Le JavaScript est un langage dit client-side, c'est-à-dire que les scripts sont exécutés par le navigateur chez l'internaute (le client). Cela diffère des langages de scripts dits server-side qui sont exécutés par le serveur Web. C'est le cas des langages comme le PHP et Node.js.



Figure 23: logo non officiel

C'est important, car la finalité des scripts *client-side* et *server-side* n'est pas la même. Un script server-side va s'occuper de « créer » la page Web qui sera envoyée au navigateur. Ce dernier va alors afficher la page puis exécuter les scripts client-side tel que le JavaScript.

Voici un schéma reprenant ce fonctionnement :

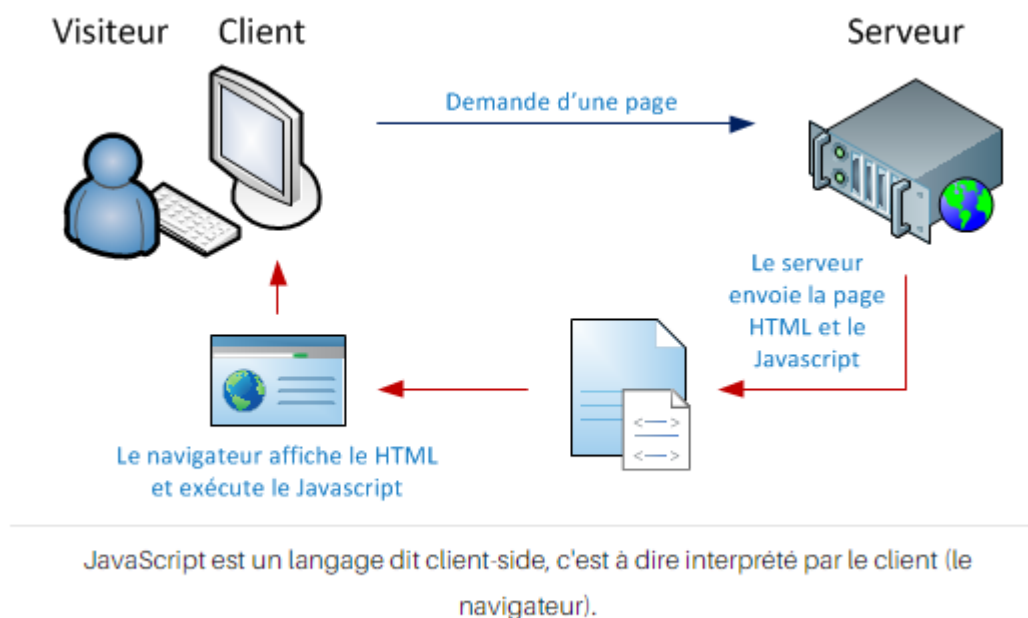


Figure 24: schéma de lecture du langage Javascript

Pour plus d'information, des tutoriels existent sur javascript. Pour commencer avec javascript :
[tutoriel : apprendre à coder avec javascript](#)

3.3. Utilisation de Node.js

Node.js offre un environnement côté serveur qui nous permet aussi d'utiliser le langage JavaScript pour générer des pages web. Il vient en remplacement de langages serveur comme PHP, Java EE, etc.



Figure 25: logo officiel

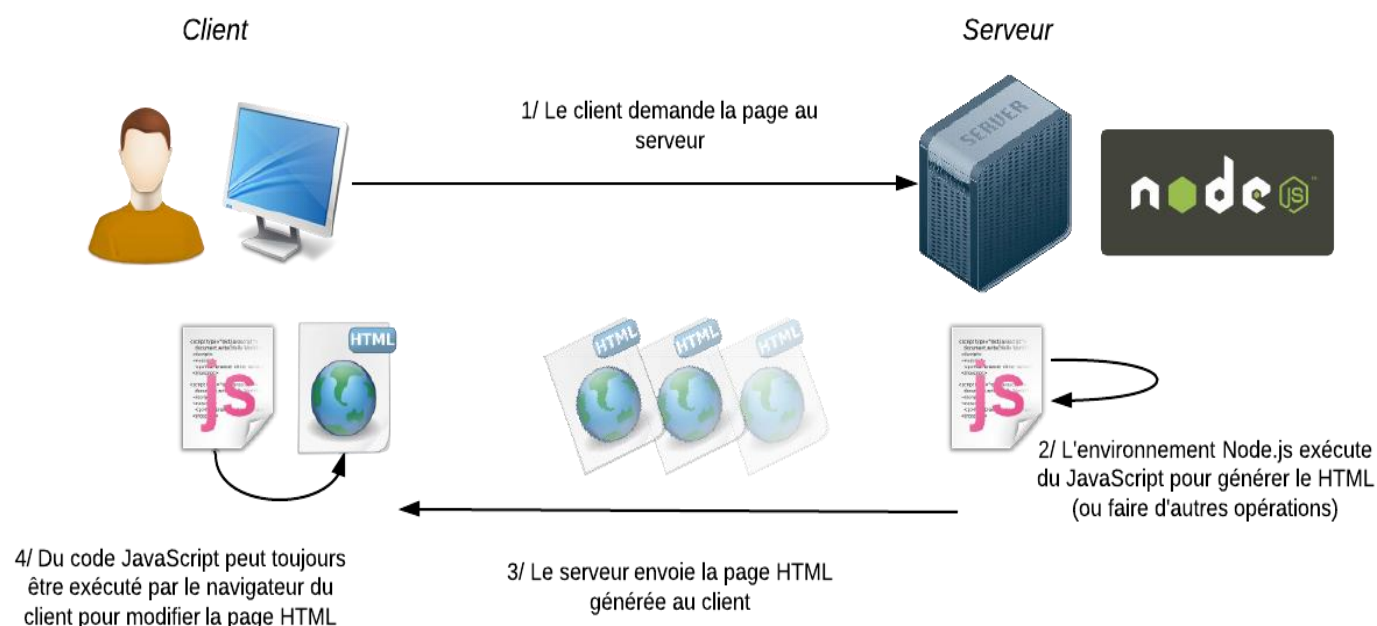


Figure 26: schéma de lecture du langage Node.js

Si Node.js est rapide, cela tient principalement à deux choses : le moteur V8 et son fonctionnement non bloquant.

Le moteur V8

Node.js utilise le moteur d'exécution ultrarapide V8 de Google Chrome. Ce moteur V8 avait fait beaucoup parler de lui à la sortie de Google Chrome, car c'est un outil open source créé par Google qui analyse et exécute du code JavaScript très rapidement.

Le modèle non bloquant

Le programme n'exécute plus les lignes dans l'ordre où elles sont écrites. Il fait ceci :

1. Le programme lance le téléchargement d'un fichier sur Internet
2. Le programme fait d'autres choses (le programme suit son cours)
3. Dès que le téléchargement est terminé, le programme effectue les actions qu'on lui avait demandées : il affiche le fichier

Schématiquement, l'exécution du programme peut donc se représenter comme ça :

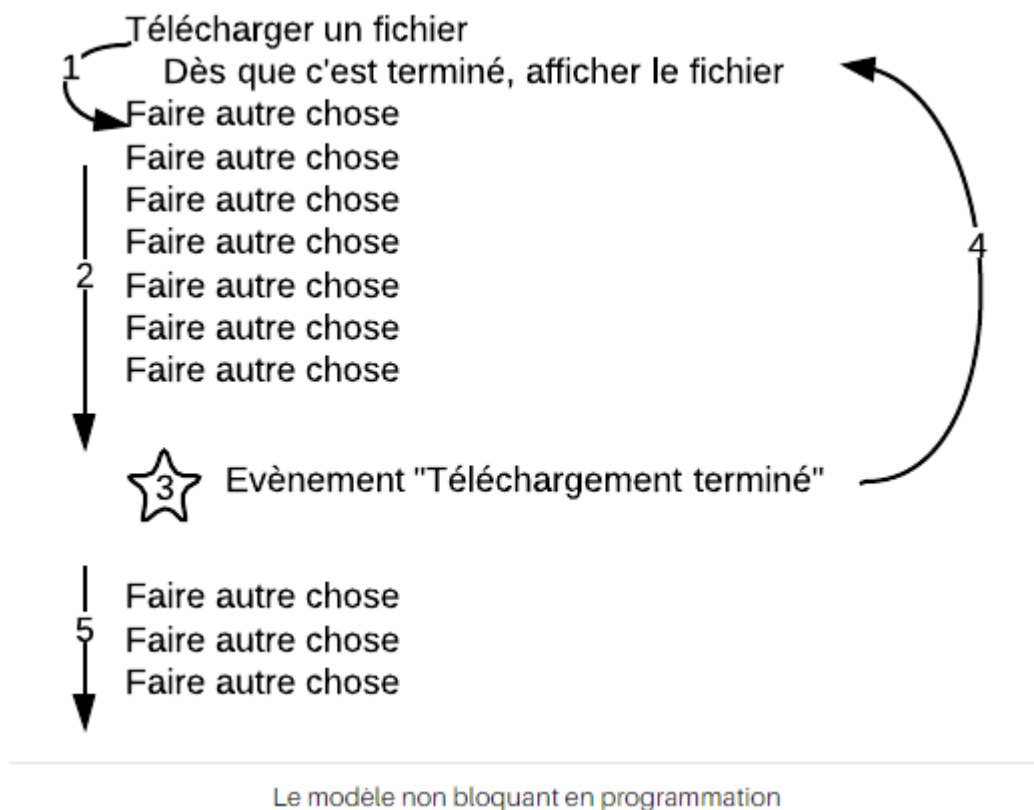


Figure 27: modèle non-bloquant

socket.io : passez au temps réel

C'est l'un des outils indispensables pour réaliser une interface web. WebSocket est une fonctionnalité supportée par l'ensemble des navigateurs récents. Elle permet un échange bilatéral synchrone entre le client et le serveur.

Avant :

Le Web a toujours été conçu comme ça : le client demande et le serveur répond.

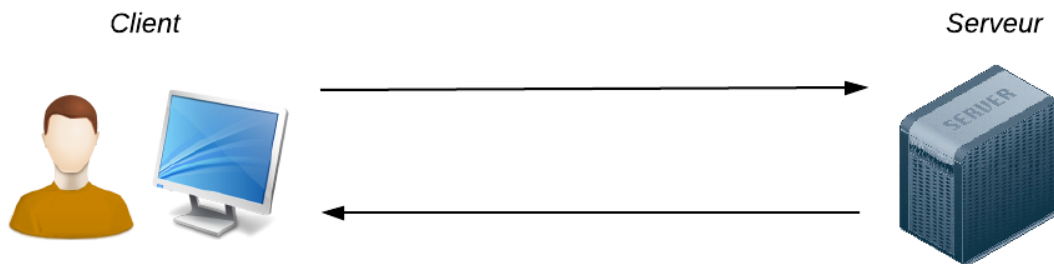


Figure 28: communication asynchrone

Habituellement, la communication est asynchrone : le client demande, le serveur répond

C'était suffisant aux débuts du Web, mais c'est devenu trop limitant ces derniers temps. On a besoin d'une communication plus réactive et immédiate. Dans ce schéma par exemple, le serveur ne peut pas décider de lui-même d'envoyer quelque chose au client (par exemple pour l'avertir "il y a un nouveau message !"). Il faut que le client recharge la page ou fasse une action pour solliciter le serveur, car celui-ci n'a pas le droit de s'adresser au client tout seul.

Maintenant :

WebSocket est une nouveauté du Web qui permet de laisser une sorte de "tuyau" de communication ouvert entre le client et le serveur. Le navigateur et le serveur restent connectés entre eux et peuvent s'échanger des messages dans un sens comme dans l'autre dans ce tuyau. Désormais, le serveur peut donc lui-même décider d'envoyer un message au client.



Figure 29: communication synchrone

Avec les WebSocket, la communication est synchrone : un tuyau de communication reste ouvert entre client et le serveur.

Le serveur veut envoyer un message au client.

Lorsqu'on détecte une connexion, on émet un message au client avec `socket.emit()`. La fonction prend 2 paramètres :

Le type de message qu'on veut transmettre. Ici, mon message est de type "message" (je ne suis pas très original, je sais). Cela vous permettra de distinguer les différents types de message. Par exemple dans un jeu, on pourrait envoyer des messages de type "deplacement_joueur", "attaque_joueur"...

Le contenu du message. Là vous pouvez transmettre ce que vous voulez.

Avec `socket.on()`, on écoute les messages de type "message". Lorsque des messages arrivent, on appelle la fonction de callback qui, ici, affiche simplement une boîte de dialogue.

Pour plus d'informations des tutoriels existes ; Pour apprendre à coder avec node.js on a réalisé le tutoriel suivant : [tutoriel Node.js](#)

3.4. En Conclusion

En maîtrisant les langages HTML , CSS, javascript et Node.js mais aussi des outils issus de ses langages comme socket.io et d'autres qu'on expliquera plus tard ; nous pouvons commencer à élaborer les interfaces web !

Revenir aux [interfaces web](#)

4. Faire une base de données en Json

4.1. Présentation

La base de données se présentera donc sous le format ci-dessous :

```
1 ▼ var clients = [  
2 ▼   "client1" : {  
3 ▼     "clientName" : {  
4     "nom de client1"  
5     };  
6 ▼     "clientFirstName" : {  
7     "prenom de client1"  
8     };  
9 ▼     "clientAdress" : {  
10    "adresse de client1"  
11    };  
12    .  
13    .  
14    .  
15  };  
16 ▼   "client2" : {  
17    .  
18    .  
19    .  
20  };  
21    .  
22    .  
23    .  
24  ];  
25  
26 ▼ var administrateurs = {  
27 ▼   "administrateur1" : {  
28    .  
29    .  
30    .  
31  };  
32 ▼   "administrateur2" : {  
33    .  
34    .  
35    .  
36  };  
37    .  
38    .  
39    .  
40  };
```

La base de données est donc formée de deux collections : l'une nommée « clients » et l'autre « administrateurs ». Dans chacune de ces collections s'enchaîne une succession de dossiers (les différents clients/administrateurs) chacun ayant leurs informations propres sous le même nom de variable (par exemple *Clientname*, *clientFirstname*...). Ce sont des variables communes à tous les clients mais avec une valeur différente pour chaque client. Ceci est très important pour la suite, lorsque l'on utilise des requêtes Mongodb à partir du serveur.

Le but étant :

- De pouvoir piocher facilement dans les clients lorsque l'on effectue une demande au niveau de l'interface administrateur : on peut par exemple rechercher un client en particulier, ou afficher l'ensemble des clients selon un tri spécifique.
- De reconnaître les différents administrateurs via un identifiant et un mot de passe.

Figure 30 : format d'une base de données sous Json

4.2. Son fonctionnement

Figure 31 : Interface client_Page d'inscription

Comment ça marche ?

Le principe est simple, comme expliqué précédemment on veut entrer tous les clients dans un fichier client. Ceci se passe automatiquement au moment de l'inscription du client dans l'interface client : le client entre ses informations, et lorsqu'il envoie le formulaire, une demande de la part du serveur créera un nouveau client associé à un identifiant dans la base de données.

Par exemple, je remplis la première partie du formulaire avec les informations suivantes :

Lorsque dans la console MongoDB je fais une requête demandant les informations correspondantes au client dont le nom est *Clientname* = « Bernaroyat », j'obtiens :

```
> db.clients.find({Clientname: "Bernaroyat"})
{ "_id" : ObjectId("5ac4971d25d138007466bfad"), "Clientname" : "Bernaroyat", "clientFirstname" : "Tancrede", "clientNumber" : "0645499133", "clientAdress" : "4 rue du Faubourg Notre Dame", "clientCity" : "Lille", "clientMail" : "tancrede.bernaroyat@gmail.com" }
>
```

Figure 32 : Console MongoDB_résultat 1

Une fois envoyé, le formulaire a bien enregistré un nouveau client dans la base de données que on peut retrouver aisément grâce à son nom ou son ID.

Et du coté de l'interface administrateur ?

Le principe est assez similaire mais dans le sens inverse : l'administrateur veut rechercher un client. Le serveur envoie une requête à la base de données, qui lance sa recherche et renvoie son résultat au serveur. Le serveur le renvoie en socket à l'interface.

4.3. Pour résumer

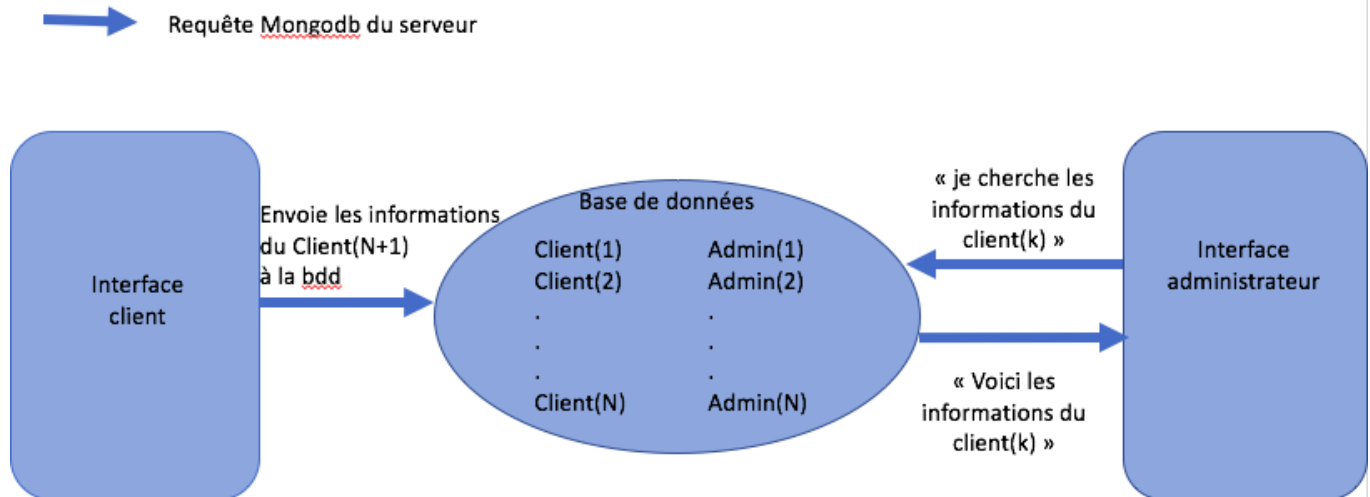


Figure 33 : Schéma d'explication d'une requête Mongodb

4.4. Requête Mongodb

À quoi ressemble concrètement une requête Mongodb du serveur ?

Prenons un exemple dans notre code :

```
1 //Lors des 4 prochaines lignes on se connecte à la base de données
2 var MongoClient = require("mongodb").MongoClient;
3 ▼ MongoClient.connect("mongodb://localhost/bddClient", function(error, db) {
4     if (error) return funcCallback(error);
5     console.log("Connecté à la base de données 'bddClient'");
6
7     var nameToFind = "Bernaroyat";
8     // La bdd contient deux collections
9     // Ici on cherche dans la collection "clients"
10 ▼ db.collection("clients").findOne({Clientname : nameToFind}, function (error, result) {
11     if (error) throw error;
12
13     console.log (
14         //On pointe sur sur le resultat de notre recherche
15         "nom: " + result.Clientname + " " +
16         "prénom: " + result.clientFirstname + " " +
17         "adresse: " + result.clientAddress + " " +
18         "ville: " + result.clientCity + " " +
19         "mail: " + result.clientMail + " " );
20     });
21 });
```

Figure 34 : Requête Mongodb

Ligne 10 : on demande à la base de données de trouver le client dont le nom est *nameToFind*, soit « Bernaroyat ».

On affiche ensuite les différentes informations du client dans la console.

Voici le résultat dans la console :

```
Connecté à la base de données 'bddClient'  
nom: Bernaroyat prénom: Tancrede adresse: 4 rue du Faubourg Notre Dame  
ville: Lille mail: tancrede.bernaroyat@gmail.com
```

Figure 35 : Console Mongobd_ résultat 2

On obtient bien les informations du client.

Cet exemple illustre bien qu'avec une information sur un client on peut retrouver toutes les informations le concernant : les différentes variables de chaque client étant les mêmes, la requête sera la même quel que soit le client.

4.5. Les fonctions utilisées

Nos différentes fonctions incluant des requêtes à MongoDB :

Interface client :

- Envoie des informations d'un client vers la base de données et création d'un nouveau client

Interface administrateur :

- Affichage de tous les clients
- Recherche d'un client en particulier
- Suppression d'un client dans la base de données
- Ajout d'un administrateur
- Suppression d'un administrateur
- Connection d'un administrateur à partir de son nom et de son mot de passe
- Changer le mot de passe d'un administrateur

Bien entendu, toutes ces fonctions du côté serveur sont connectées à une fonction du côté de l'interface (grâce à la fonction socket) car c'est dans l'interface que l'on donne l'ordre d'envoyer la requête.

Revenir aux [Interfaces web](#)

5. Le management du projet

5.1. Les plannings

Tableau 11 : Planning prévisionnel

ACTIVITÉ	DÉBUT DU PLAN	DURÉE DU PLAN	PERSONNES IMPLIQUÉES	Semaines											
				1	2	3	4	5	6	7	8	9	10	11	12
Obtenir les informations des projets précédents	1	3	Tous les membres												
Définir les objectifs et les tâches à réaliser	1	3	Tous les membres												
Modifier interface client	3	5	Jean, Tancrede, Salah												
acquérir connaissance web	3	2	Jean, Tancrede, Salah												
Faire plateforme web administrateur	4	7	Jean, Tancrede, Salah												
acquérir connaissance base de données: langage	3	2	Louis, Ugo												
Faire les bases de données	4	7	Louis, Ugo												
Réaliser programme fuzzy logic	3	5	Louis, Ugo												
recupérer indices sentinél 2	3	3	Louis, Ugo												
Récupérer indices sentinél 3	8	3	Tous les membres												
rédiger le rapport	10	2	Tous les membres												
préparer la soutenance	11	2	Tous les membres												

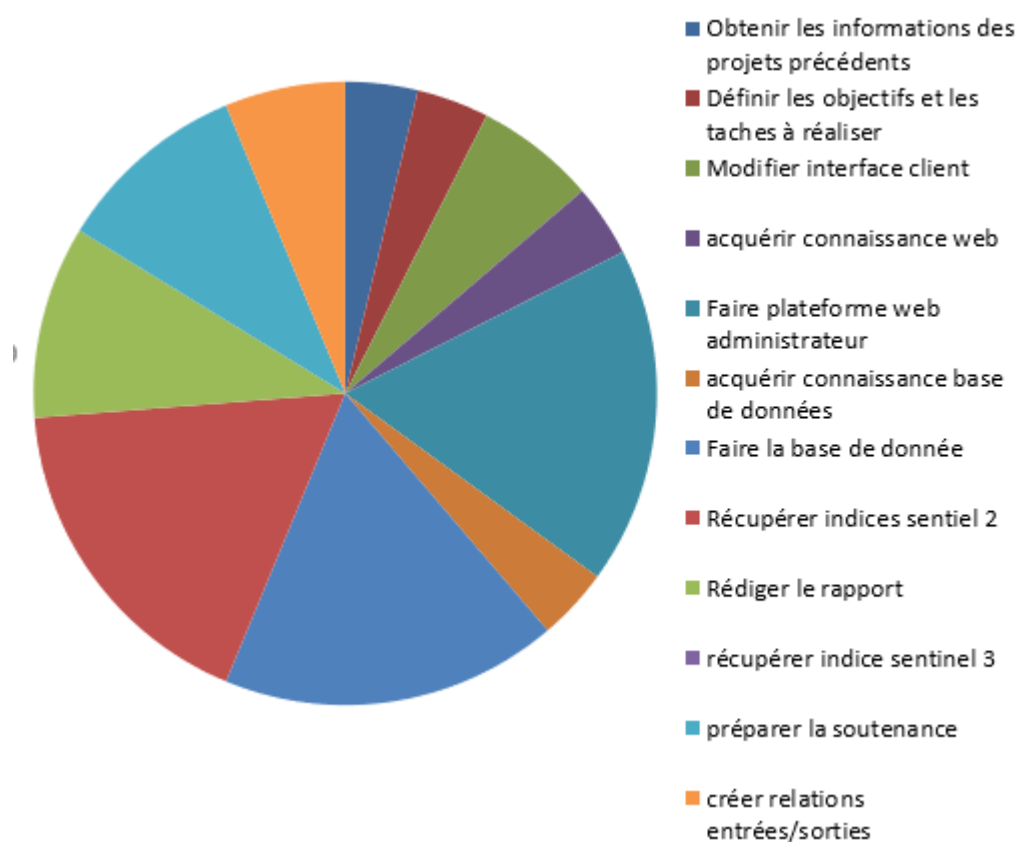
Tableau 12: Planning réel

ACTIVITÉ	DÉBUT RÉEL	DURÉE RÉELLE	PERSONNES IMPLIQUÉES	Semaines											
				1	2	3	4	5	6	7	8	9	10	11	12
Informations des projets précédents	1	10	Tous les membres												
objectifs et les tâches à réaliser	1	3	Tous les membres												
Modifier interface client	5	3	Tous les membres												
connaissance web	2	6	tous les membres												
Faire plateforme web administrateur	6	5	Jean, Louis												
acquérir connaissance base de	2	6	Tancredi												
Faire les bases de données	7	5	Tancredi												
Réaliser relation entrées/sorties	8	3	Louis												
recupérer indices sentinel 2	6	6	Ugo, Jean												
recupérer indices sentinel 3	11	1	Ugo, Jean												
rédiger le rapport	6	6	Tous les membres												
préparer la soutenance	11	12	Tous les membres												

Notre planning a beaucoup évolué au cours des semaines. La première répartition du travail n'avait pas été bien faite. En effet les membres du projet venant de HEI étaient regroupés sur les mêmes tâches ainsi que les membres venant de ISEN. Cela a provoqué un déséquilibre au niveau des tâches à réaliser. En effet, le temps de travail dédié au projet n'était pas le même pour les étudiants de HEI et les étudiants de ISEN. C'est pourquoi nous avons décidé de changer le planning et de se répartir le travail d'une façon plus adéquate.

5.2. Timesheet

Figure 36 : Timesheet



Sur le timesheet, on peut voir qu'un peu plus de 50% des tâches réalisées concernent les demandes du cahier des charges. Le reste concerne la compréhension du sujet (10%), la formation (10%) et les rendues (20%).

Revenir au [déroulement du projet](#).

6. L'interface client : version complète

L'interface client a été réalisée au premier semestre par une autre équipe. Notre travail a été de faire des modifications.

Rappel sur l'interface client :

Elle est utilisée par les clients (les agriculteurs) de renseigner les informations nécessaires pour traiter leurs couverts végétaux.

En premier, l'interface demande à l'agriculteur ses données personnelles (nom, prénom, adresse...), ensuite elle lui demande des renseignements sur son champ. Et ensuite, la partie la plus importante, elle lui demande des informations sur chacune de ses parcelles.

La première étape a été d'ordonner les pages de l'interfaces ; les regrouper par blocs, mettre des titres. Il a fallu ensuite ajouter des parties et en modifier d'autres.

Avant :

Figure 37 : Bloc informations générales-avant modification

Après :

Figure 38 : Bloc information générale- après modification

Avant :

Vos couverts végétaux

Parcelle n°1

Informations sur vos cultures

Votre culture précédente : Sélectionnez ici

Votre culture suivante : Sélectionnez ici

Renseignez vos intrants:

Cartographie de Fertilisation Farmstar *

Choisir un fichier | Aucun fichier choisi

Figure 40 : Bloc informations sur vos cultures-avant modification

Après :

Rotation

Votre culture précédente

Sélectionnez ici

Votre culture suivante

Sélectionnez ici

Date récolte précédente

jour/mois/année

Figure 39 : Bloc informations sur vos cultures-après modification

Analyses de Sol *

Choisir un fichier | Aucun fichier choisi

Vos bilans azotés/hydriques *

Sélect. fichiers | Aucun fichier choisi

Renseignez vos rendements par

Figure 41 : Bloc Fertilisation-avant modification

Fertilisation

Plan de fumure

Choisir un fichier | Aucun fichier choisi

Cartographie de Fertilisation

Choisir un fichier | Aucun fichier choisi

Analyses de Sol

Choisir un fichier | Aucun fichier choisi

Reliquats sortie d'hiver *

Sélect. fichiers | Aucun fichier choisi

Figure 42 : Bloc Fertilisation-après modification

- La case « Renseigner vos intrants » a été remplacé par une case pour insérer un fichier. Ce fichier sera enregistré avec les autres fichiers dans le dossier FarmingData/ parcelle.
- Le titre : « Vos bilans azotés/ hydriques » a été changé par : « Reliquats sortie d'hiver ».
- Le block Fertilisation regroupe les quatre fichiers à insérer.
- La date prévisionnelle de récolte de la culture précédente a été rajouté dans le bloc « Rotation ».

Avant :

Figure 44 : Bloc rendement-avant modification

Après :

Figure 43 : Bloc rendement-après modification

Après :

Figure 45 : Bloc Autres remarques- après modification

- Le bloc « Rendement » et « Autres remarques » ont été ajouté.

Après :

Figure 46 : Bloc couvert Végétal-après modification

Après :

Figure 47 : Block couvert Végétal complet-après modification

- Le boc « Couvert Végétal » a été rajouté.
- Le menu déroulant « type de semoirs pour les couverts végétaux » a été créé. Le client peut choisir dans une liste entre : Semoirs à céréales, semoirs à la volée, semoirs de semis direct.
- Le menu déroulant « Mode de destruction du couvert » a été créé. Le client peut choisir dans une liste entre : Chimique, broyage, labour, gel, déchaumage.
- La partie « Problèmes de la parcelle » a été ajoutée. Les problématiques de la parcelle pouvant être sélectionner sont les suivants : Adventices, Maladies, Ravageurs, Compaction). L'agriculteur peut sélectionner plusieurs cases et pour chacune d'entre elle un autre menu déroulant s'affiche ave une zone de texte pour renseigner son problème.

Revenir à la [modification de l'interface client](#)

7. Méthode de récupération des indices du satellite Sentinel 2

7.1. Le fichier bash

Pour récupérer les indices, il faut utiliser un fichier de commande bash linux. Il est impératif d'être sous linux pour le faire fonctionner. Ce fichier se trouve sur Copernicus. Celui-ci permet d'effectuer de nombreuses opérations de lecture et de tri dans les données des satellites Sentinel. Il a subi quelques adaptations pour notre utilisation : il a été simplifié et épuré au maximum, et le nombre d'arguments a été fortement réduit, pour n'arriver qu'aux coordonnées de la zone qui s'avère intéressante. Par exemple, les arguments concernant les identifiants de connexion au hub de Sentinel, les instruments qui sont intéressants, les satellites concernés, le nombre d'images désirées ont été intégrés au script pour ne pas avoir à les renseigner à chaque appel du script. Le script bash permet donc de faire ceci :

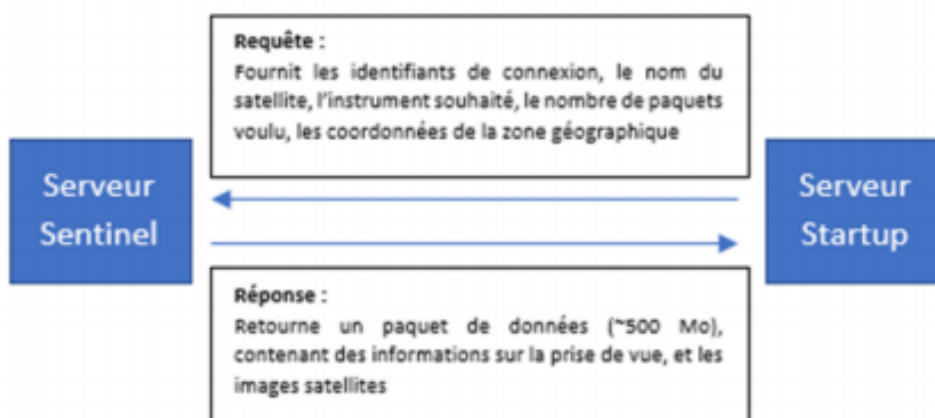


Figure 48 : Schéma de principe du script bash

Il est possible de récupérer des données satellites depuis internet. Il suffit dès lors de renseigner les coordonnées GPS standard (avec latitude / longitude) pour avoir accès à de nombreuses données correspondant à un endroit donné (par exemple, la quantité d'azote, de carbone dans le sol).

7.2. Gestion de la récupération des indices

Node.js permet de gérer le processus et de rendre la récupération automatique. Le serveur permet d'exécuter les actions suivantes :

- Lancement du fichier bash de Copernicus, en lui passant latitudes et longitudes
- Extraction des images du fichier Zip ainsi téléchargé, en supprimant le reste, afin de répondre à la problématique posée par le poids des données de Copernicus.
- Suppression des fichiers de téléchargements de Copernicus, pour indiquer les noms de paquets téléchargés.

Les modules utilisés pour le faire fonctionner sont :

- Http pour lancer le serveur
- Fs et path pour la manipulation de fichiers et de dossiers dans le dossier de stockage du serveur
- Socket io pour permettre les connexions et la communication en temps réel
- Child_process pour lancer les fichiers bash et python depuis le serveur
- Csv-parse pour extraire les noms des archives téléchargées depuis un fichier csv créé par Copernicus
- Adm-zip pour extraire uniquement les dossiers images des archives de Copernicus. Cette version de serveur est capable de télécharger les données nécessaires à la création d'un NDVI pour une zone donnée, de générer ce NDVI et de nettoyer les fichiers restants.

7.3. Calcul du NDVI par traitement d'images

Les images téléchargées par Copernicus étaient au format JP2, un format qui paraissait à l'origine inexploitable. La bibliothèque rasterio est justement utilisée pour le traitement des images téléchargées via Copernicus. Elle permet en effet d'ouvrir un fichier JP2 et de lui appliquer diverses modifications. Le point intéressant de cette bibliothèque est qu'elle permet d'exploiter et de conserver les informations concernant les coordonnées UTM contenues dans le fichier image. Voici comment fonctionne ce système de coordonnées militaire : Chaque endroit de la planète appartient à une zone de 6° de largeur. Il y en a 120 au total (60 hémisphère Nord et 60 hémisphère Sud). Par exemple, voici la zone 31N :



Figure 49 : Coordonnées UTM

Rasterio permet de récupérer ces informations sur une image de type JP2. Celles-ci ont une résolution de 10m et représentent un carré de 100 km de côté. Chaque pixel de cette image peut donc facilement être mis en relation dans ce système de repérage planétaire. Ce sont ces informations qui nous permettent de superposer plusieurs images satellites et de générer le NDVI. Il suffit alors de considérer les photographies infrarouges comme des matrices de pixels, où les 2 premières coordonnées représentent la position du pixel dans l'image, et où la 3ème dimension donne la réponse au rayonnement infrarouge.

La formulation suivante, celle du NDVI, est ensuite appliquée à chacune des entrées de la matrice :

$$\text{ndvi} = ([\text{Near InfraRed}] - [\text{RED}]) / ([\text{Near InfraRed}] + [\text{RED}])$$

Le résultat de ces opérations est stocké dans une matrice suivant le modèle des deux premières. Chaque pixel se voit attribuer une valeur de NDVI comprise entre 0 et 1 (cette valeur pourra être changée d'échelle de façon proportionnelle). Une fois que ce script python a été mis en place, il fallait l'appeler depuis le serveur, comme expliqué précédemment au moyen du module Child Process, en lui envoyant les chemins d'accès aux deux images de bandes B04 et B08. Le script python récupère ces données comme s'il récupère des arguments dans un exemple d'utilisation normal. Une image .tif est alors générée pour le NDVI, et stockée dans le dossier du serveur.

Il est possible d'obtenir les images NDVI automatiquement grâce au Python, Bash, et Node JS.

Retour au [Satellite Sentinel 2](#).

8. Méthode de récupération du DEM

Même si la récupération de ces images n'est pas automatisée, l'accès à cet indice peut être réalisée de façon très efficace en suivant la méthode ci-dessous :

1. Aller sur le lien suivant : <https://scihub.copernicus.eu/s3/#/home>
2. Appuyer sur l'outil Polygon (en bas à droite de l'écran).
3. Créer votre parcelle directement sur la carte (on peut visualiser les coordonnées numériques en bas à gauche de l'écran).



Figure 50 : Outil Polygon

4. Lancer la recherche en appuyant sur la petite loupe présente à côté de la barre de recherche.

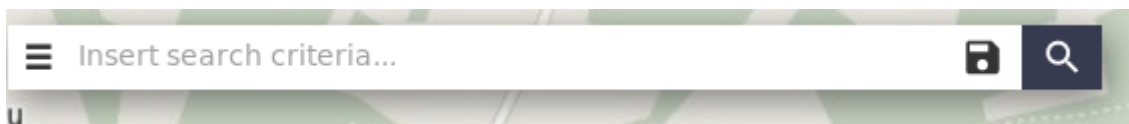


Figure 51 : Lancer la recherche

5. S'identifier (username = s3guest/ Password = s3guest)

Figure 52 : identification

Un chargement se lance, puis les archives apparaissent sur la gauche après quelques instants.

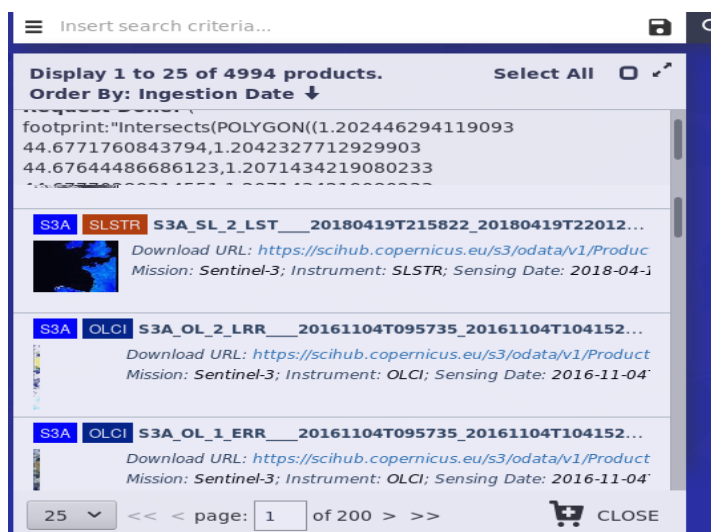


Figure 53 : Archive de la recherche

6. Choisir l'archive S3A_OL_2_LRR pour le DEM, télécharger l'archive, puis la décompresser. Le fichier comprend plusieurs documents sous le format.NC.

ents S3A_OL_2_LRR...NT_002 (3).SEN3

Nom	Taille	Der
geo_coordinates.nc	58,5 Mo	
instrument_data.nc	1,2 Mo	
iww.nc	7,9 Mo	
lqsf.nc	3,8 Mo	
ogvi.nc	2,6 Mo	
otci.nc	2,3 Mo	
rc_ogvi.nc	13,6 Mo	
tie_geo_coordinates.nc	4,7 Mo	
tie_geometries.nc	9,1 Mo	
tie_meteo.nc	78,6 Mo	
time_coordinates.nc	25,2 ko	

Figure 54 : Archive S3A_OL_2_LRR

Pour ouvrir les fichiers il faut installer Java et Panoply

7. Installer [Java](#).
8. Installer [Panoply](#).
9. Ouvrir « geo_coordinates.nc » avec l'application Panoply.

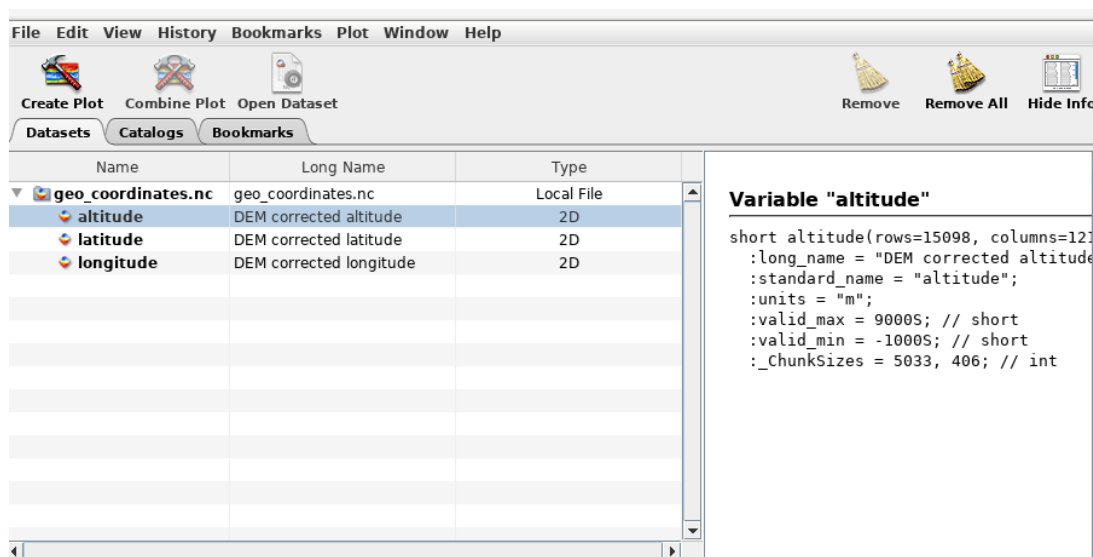


Figure 55 : lecture du fichier « DEM altitude » en 2D Y/X

10. Lancer la lecture du fichier « DEM altitude » en 2D Y/X (voir en fonction de la spécificité du logiciel)

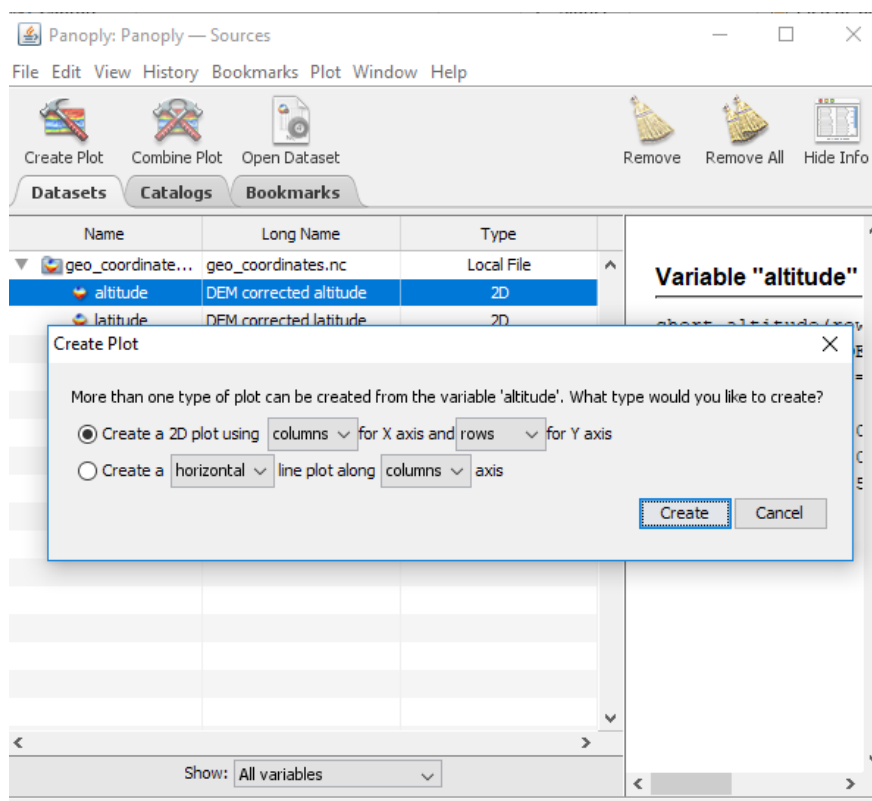


Figure 56 : DEM altitude » en 2D Y/X

Vous avez maintenant accès à la carte du DEM.

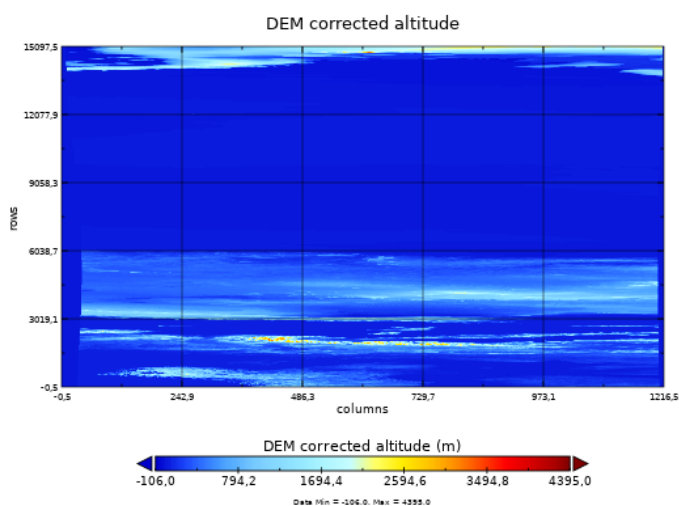


Figure 57 : Carte DEM

Retour au [Satellite Sentinel 3](#)