

Rapport du projet de fin d'études

*Pour l'obtention du diplôme Ingénieur d'Etat en Ingénierie des
Systèmes d'Information et Big-data*

La conception et la mise en œuvre d'un
l'enchaînement de traitement des flux donnés
géographiques



Réalisé par :

Ayoub TIGOUDERN

Encadré par :

Mr. Jean-Marie Favreau (LIOMS)

Encadré par :

Pr. Lahcen Moumoun

Ecole Nationale des sciences Appliquées de Berrechid B.P 218,

Département Informatique, mathématique

Site web : <http://www.ensab.ac.ma/>

Année universitaire : 2021/2020

REMERCIEMENTS

Je remercie Mr **Jean-Marie Favreau** pour m'avoir accueilli durant cinq mois afin de réaliser ce projet et pour la confiance et la liberté qu'il m'a donné, leurs conseils ainsi que leurs éclairages sur l'ensemble du projet.

Je tiens à remercier Professeur **Lahcen MOUMOUN**, mon encadrant, pour tout le soutien, l'aide et l'orientation qu'il m'a apporté tout au long de ce travail ainsi que pour la patience et le temps inconditionnel qu'il m'a consacré.

Je remercie également **Jérémy Kalsron et Samuel Braikheh** et le reste de l'équipe de LIMOS pour leur disponibilité et pour m'avoir ainsi permis d'effectuer mon stage dans de très bonnes conditions.

Mes vifs remerciements s'adressent à Mr **Boujemaa ACHCHAB**, le responsable de la filière, pour avoir toujours été à l'écoute des étudiants tout au long de notre formation.

J'adresse également mes remerciements les plus chaleureux au directeur de l'école nationale des sciences appliquées de Berrechid et l'ensemble des professeurs de l'ENSAB pour la qualité de leur enseignement.

J'exprime ma profonde reconnaissance envers le staff pédagogique du Filière Ingénierie des systèmes d'information et Big-Data, et tout le personnel administratif de l'école nationale des sciences appliquées de Berrechid, qui nous ont aidés que ce soit de près ou de loin.

Tous mes remerciements vont finalement aux membres du jury, qui ont accepté de dédier leurs connaissances et expériences professionnelles pour juger mon travail.

RÉSUMÉ

L'objectif de ce projet est de faire l'étude de faisabilité technique des besoins ainsi que d'élaborer l'état d'art technique en se basant sur les tests des différentes technologies et outils de problématique de recherche étudiée, également la mise en place d'un enchaînement de traitement des données géographiques dans le cadre de projet ACTIVmap, ce pipeline composé de Postgis pour stocker les données, de serveur Geoserver pour l'administration des données et des couches, d'une application web réalisé en utilisant Geodjango et OpenLayers et un serveur PyWPS pour construire et publier des processus et des scripts python pour appliquer des traitements au flux des données. Aussi la liaison des différents blocs via des protocoles de OGC comme (WPS, WFS, WMS etc) et d'autres de WEB, ainsi que ce projet exploite les données collaboratives d'OpenStreetMap.

Mots-clés : PyWPS, WPS, Geoserver, WFS, PostGis, QGIS.

ABSTRACT

The objective of this project is to carry out the technical feasibility study of the needs as well as to develop the state of the art based on the tests of the various technologies and tools, also the establishment of geographic data processing sequence as part of the ACTIVmap, project, this pipeline consists of Postgis to store the data, Geoserver for administration of and layers, web application using Geodjango and OpenLayers and a PyWPS server for build and publish python processes and scripts to apply processing to data flow. Also, the connection of the different blocks via OGC protocols like (WPS, WFS, WMS etc) and others of WEB, as well as this project exploits the OpenStreetMap collaborative data.

Keywords: Pywp, WPS, Geoserver, WFS, PostGis, QGIS.

LISTE DES FIGURES

Figure 1: Organigramme de LIMOS	12
Figure 2: représentation graphique de l'idée de projet.	13
Figure 3:Diagramme des missions durant la période de stage.....	16
Figure 4: Processus de méthode Agile	18
Figure 5: Première diagrammes de Gantt	20
Figure 6: Deuxième diagrammes de Gantt	21
Figure 7: Troisième diagramme de Gantt.....	21
Figure 8:Deux objets représentés en mode raster et en mode vecteur	25
Figure 9: Organisation des données en couches	28
Figure 10: Architecture micro services.	41
Figure 11:L'architecture générale de projet	42
Figure 12: Les protocoles web assurant la circulation de flux des données.....	43
Figure 13: Les couches entrées pour exécuter les processuss	50
Figure 14: Workspace ACTIVmap	51
Figure 15: les couches publiées sous Geoserver	52
Figure 16: Design Pattern Classification,.....	54
Figure 17:L'authentification.....	54
Figure 18: l'espace administration.....	54
Figure 19: L'ajout d'un utilisateur	55
Figure 20: Gestion des permissions des groupes des utilisateurs	55
Figure 21: L'affichage des couches dans une carte.....	56
Figure 22: Exemple d'un Algorithme construit en modeleur graphique de QGIS.....	57
Figure 23: Exemple de requête d'exécution de processus Buffer	58

SOMMAIRE

Remerciements

Résumé et Abstarct

Liste des figures

Sommaire

INTRODUCTION GÉNÉRALE8

Chapitre I : Contexte général du projet

Introduction11

1. Présentation de l'organisme d'accueil.....11

1.1 L'organigramme de laboratoire..... 12

2. Le projet ACTIVmap.....12

2.1 Les équipes de projet ACTIVmap..... 13

2.1.1 L'équipe de GeoVIS..... 13

2.1.2 L'équipe de LASTIG (IGN) 14

2.1.3 L'équipe ELIPSE..... 14

2.1.4 L'entreprise FeelObject..... 14

2.1.6 Le diagramme des équipes participant au projet ACTIVmap 15

3. Objectifs et description des besoins15

3.1 Problématique..... 15

3.2 Les objectifs du projet..... 16

3.3 Missions 16

4. Planning et déroulement du projet18

4.1 L'approche agile..... 18

4.2 Planification des tâches 19

4.4 Les membres de l'équipe 19

4.5 Les diagrammes de Gantt 20

Conclusion22

Chapitre II : Les fondements théoriques

Introduction24

1. Revue des concepts.....24

1.1	L'information géographique.....	24
1.2	Système d'information géographique.....	26
1.3	Fonctions d'un SIG	27
1.4	Organisation des données en couche	28
1.5	Les données géographiques dans le contexte web	29
1.6	Utilisation d'OpenStreetMap	29
1.7	Les protocoles web de l'OGC adoptés	29
2.	Etat de l'art technique.....	33
2.1	Technologies et outils.....	33
2.1.1	GeoServer	33
2.1.2	GeoNode	34
2.1.3	Lizmap.....	34
2.1.4	Zoo Project	34
2.1.5	Deegree Project.....	35
2.1.6	PyWPS.....	35
2.1.7	Py-QGIS-WPS.....	35
2.2	Étude comparative de l'état de l'art technique.....	36
2.2.1	La méthodologie suivie	37
3.	Discussion autour des extensions de Web Processing Service de Geoserver.....	38
3.1	Geoserver wps-remote	38
3.2	Scripting de Geoserver.....	39
	Conclusion	39

Chapitre III : La mise en œuvre et la solution retenue

	Introduction	41
1.	Solution proposée	41
1.1	Description du solution proposée	42
1.2	Architecture générale	42
1.3	Protocoles utilisées.....	43
2.	Technologies et outils de développement.....	44
2.1	Technologies et langages de programmation	44
2.1.1	Python.....	44
2.1.2	GeoDjango	44
2.1.3	OpenLayers	45
2.1.4	La base des données PostGIS	45
2.2	Outils et environnements de travail	45
2.2.1	QGIS.....	45

2.2.2	Docker.....	46
2.2.3	Le serveur d'applications Tomcat.....	46
2.2.4	Postman.....	46
2.2.5	Visual Studio.....	47
2.2.6	Geojson.io.....	47
2.2.7	OpenStreetMap.....	47
Conclusion		48
 Chapitre IV : Étude de Cas		
Introduction		50
1.	Description des données de l'étude de cas	50
1.1	Administration de Geoserver	51
2.	Application Web.....	52
2.2	Diagramme de classe	52
2.3	Design Pattern Classification utilisé dans l'application web (DDP).....	53
2.4	Espace administrateur	54
2.5	Processus de traitement des données.....	56
3.	Difficultés et défis de déploiement de cette solution	57
3.1	Au niveau de serveur PyWPS.....	57
3.2	Contraintes de la requête xml	58
3.3	Contraintes du type des données.....	58
3.4	Contraintes fonctionnelles	59
Conclusion		59
 Conclusion générale		60
 Bibliographie & Webographie.....		61
 Annexe		62

INTRODUCTION GÉNÉRALE

La déficience visuelle correspond à la perte de tout ou partie de la vision. Elle peut être de naissance ou survenir au cours de la vie. Selon les estimations des nations unies en 2019¹, 253 millions de personnes présentent une déficience visuelle : 36 millions d'entre elles sont aveugles et 217 millions présentent une déficience visuelle modérée à sévère dans le monde. Le déplacement des personnes en situation de déficience visuelle est un enjeu majeur de leur autonomie. Pour apprendre à se déplacer au sein d'un environnement urbain, celles-ci peuvent apprendre auprès d'un instructeur de locomotion son rôle est d'accompagner ces personnes pour lui permettre d'utiliser les sens à sa disposition. Pour cela, l'instructeur peut réaliser des cartes tactiles représentant une zone d'étude de manière photoréaliste ou schématique compréhensible par ces personnes pour faciliter leurs itinéraires.

Le projet ACTIVmap vise à produire des outils d'aide à la conception de ces cartes pour les instructeurs de locomotion. En outre, un des objectifs du projet est d'ajouter à celles-ci des informations sonores pour proposer une carte multisensorielle. ACTIVmap est issu du projet COMPAS (Cartographie et Outils Multisensoriels Pour l'Accessibilité Spatiale) qui explore plus largement l'accessibilité spatiale, et partage des thématiques avec le projet OD4M (Open Data for Mobility).

En effet, les réalisations dans le cadre du projet aspirent à reposer sur des données ouvertes, et particulièrement la base de données libre OpenStreetMap, accessible en lecture et écriture à tout un chacun et évolutive. Ces données représentent une source importante qui peut aider à faciliter la mobilité des personnes en déficience visuelle.

La manipulation des données OpenStreetMap fait partie des travaux transversaux des projets. Cette contribution peut concerner l'intégration de nouvelles données, la proposition de formalismes pour intégrer de nouvelles thématiques liées à l'accessibilité, ou encore s'intéresser aux outils de traitement automatique et proposition des nouvelles approches de consolidation et d'enrichissement de ces sources.

Mon stage s'inscrit dans le projet en se concentrant sur la conception et la réalisation d'un enchaînement de traitement des données géographiques dans le contexte web afin d'automatiser

¹ <https://www.who.int/fr/news-room/fact-sheets/detail/blindness-and-visual-impairment>

et de concrétiser les nouvelles approches qui en relation avec la mobilité des personnes en situation de déficience visuelle. Ce rapport de stage de fin d'étude d'ingénierie explore les problématiques inhérentes aux technologies de traitement des données spatiales afin de mise en place une infrastructure, puis propose un traitement automatique des données et aussi qui permet de développer une application web distribué. Le présent rapport se compose de cinq chapitres :

- ✓ Le premier chapitre intitulé « Contexte général du projet » présente l'organisme d'accueil et la direction où nous avons effectué notre stage, ainsi que le cadre général du projet avec le planning et le déroulement du projet.
- ✓ Dans le deuxième chapitre, nous abordons les fondements théoriques et les concepts de base liés au traitement de l'information géographique. Une étude théorique approfondie concernant les notions de traitement automatique des données spatiales dans le contexte web. Aussi ce chapitre présente un état de l'art technique de la problématique étudiée et par rapport auquel nous situons notre apport.
- ✓ Dans le troisième chapitre, nous définissons notre approche et nous décrivons notre solution proposée ainsi que l'ensemble des technologies et outils adoptés afin de réaliser nos besoins.
- ✓ Le quatrième chapitre consacré à une étude de cas la description des différentes étapes de test de la solution retenue ainsi que les défis rencontrés.

Chapitre I : Contexte général du projet

Introduction

Dans la première partie de ce chapitre, nous présentons l'organisme au sein duquel s'est déroulé ce stage. Pour mettre en exergue l'objectif de ce travail, nous présentons les motivations qui sont à l'origine de nos investigations. Et pour assurer la bonne conduite du projet, un plan d'action est présenté avec la méthodologie adoptée et le processus de développement durant toute la période de ce stage.

1. Présentation de l'organisme d'accueil

Le Laboratoire d'Informatique, de Modélisation et d'Optimisation des Systèmes (LIMOS)² est une Unité Mixte de Recherche en informatique, et plus généralement en Sciences et Technologies de l'Information et de la Communication (STIC). Le LIMOS est principalement rattaché à l'Institut des Sciences de l'Information et de leurs Interactions et de façon secondaire à l'Institut des Sciences de l'Ingénierie et des Systèmes (INSIS). Il a pour tutelles académiques l'Université Clermont Auvergne et l'École Nationale Supérieure des Mines de Saint-Etienne (EMSE), et comme établissement partenaire l'école d'ingénieur SIGMA. Le LIMOS est membre des labex IMOBS et ClercVolc et de la fédération de recherche en Environnement (qui regroupe 17 laboratoires UCA et INRA du site de Clermont-Ferrand). Il est membre associé de la fédération MODMAD portée par l'Université Jean Monnet de Saint-Etienne. Le positionnement scientifique du LIMOS est centré autour de l'Informatique, la Modélisation et l'Optimisation des Systèmes Organisationnels et Vivants.



L'effectif :

- 95 enseignants-chercheurs et chercheurs
- 77 doctorants
- 8 postdocs
- 9 chercheurs associés
- 14 Ingénieurs, Techniciens et Administratifs

² <https://limos.fr/>

1.1 L'organigramme de laboratoire LIMOS

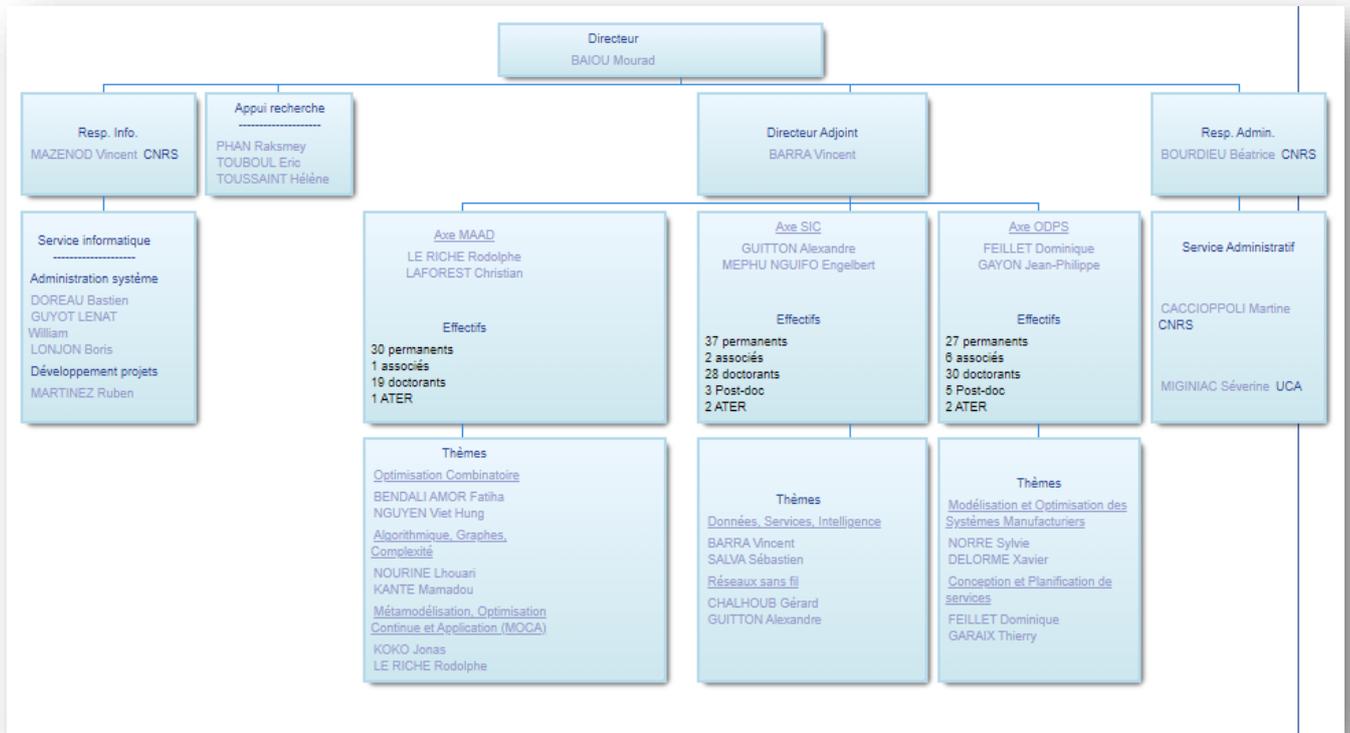


Figure 1: Organigramme de LIMOS

2. Le projet ACTIVmap



ACTIVmap

ACTIVmap³ est un Projet de Recherche Collaborative - Entreprise (PRCE) mené par le Laboratoire d'Informatique, de Modélisation et d'Optimisation des Systèmes (LIMOS), conjointement avec l'équipe GeoVIS de l'UMR LASTIG (IGN), l'équipe ELIPSE de l'Institut de Recherche en Informatique de Toulouse, et l'entreprise FeelObject. Le projet est financé par l'ANR pour une durée de 42 mois (mars 2020 - février 2024), et a obtenu le soutien d'Aerospace Valley.

³ <https://activmap.limos.fr/>

La production de cartes accessibles aux déficients visuels (DV) est aujourd'hui une pratique majoritairement artisanale. Les avancées scientifiques en géomatique, combinées à la disponibilité de données géographiques collaboratives (type OpenStreetMap) et à des approches interactives offrent des possibilités originales de conception de telles cartes.

Dans ce projet, les porteurs développeront un ensemble d'outils interconnectés, proposant une aide à la conception de cartes, et permettant des modes d'accès adaptés à la diversité des handicaps et des utilisations de ces représentations géographiques. La conception et la validation des solutions envisagée seront élaborées en collaboration avec des représentants de la communauté DV, et les résultats seront intégrés aux produits développés par le partenaire industriel du projet.⁴



Figure 2: représentation graphique de l'idée de projet.

2.1 Les équipes de projet ACTIVmap

Le projet ACTIVmap est puisqu'un projet de recherche compose de plusieurs équipes et collaborateurs qui travaillent sur des axes de recherche différents avec des objectifs variés, cependant que tous ces travaux réalisés par ces équipes partagent même visons et même perspectives.

2.1.1 L'équipe de GeoVIS5

L'équipe GeoVIS fournit des connaissances, des méthodes et des outils de géo visualisation permettant à différents utilisateurs de concevoir des représentations graphiques de phénomènes spatio-temporels, et d'inférer des connaissances spatio-temporelles à partir de l'interaction avec certaines dimensions d'un phénomène, à partir de données spatiales (cartes, imagerie, 3D modèles, MNT, nuages de points, etc.) et des données externes (textes, photographies, données web, données thématiques, etc.).



⁴ <https://activmap.limos.fr/>

⁵ <https://www.umr-lastig.fr/geovis/>

2.1.2 L'équipe de LASTIG (IGN)⁶

L'équipe de LASTIG travaille sur le domaine de la science de l'information géographique. La recherche de LASTIG couvre le cycle de vie complet des données spatiales, de la capture à la visualisation, en passant par la modélisation, l'intégration et l'analyse des données spatiales. Pour faire suite au précédent projet de recherche du laboratoire, le LASTIG s'intéresse particulièrement aux données de référence spatio-temporelles.



2.1.3 L'équipe ELIPSE

Le projet de l'équipe Etude de L'Interaction Personne Système (ELIPSE) s'inscrit dans le domaine de l'Interaction Homme Machine (IHM).



Sa pluridisciplinarité (sciences cognitives, IHM, informatique) permet de traiter des Techniques d'Interaction Avancées (TIA) comme objets de recherche dans le but de développer des modèles, des nouvelles techniques d'interaction ou dispositifs, et comme outils de recherche pour explorer leur impact sur la perception et la cognition humaine.⁷

2.1.4 L'entreprise FeelObject⁸

FeelObject est une entreprise ayant une expertise en fabrication additive ou impression 3D professionnelle et empreinte d'une forte culture de l'innovation, l'équipe de FeelObject s'appuie sur son propre laboratoire d'ingénierie (Studio) pour imaginer, designer, tester et fabriquer les objets de demain.



⁶ <https://www.ign.fr/>

⁷ <https://www.irit.fr/departement/intelligence-collective-interaction/elipse/>

⁸ <https://www.feelobject.fr/en>

2.1.5 Le diagramme des équipes participant au projet ACTIVmap



Institut de Recherche
en Informatique de Toulouse
CNRS - INP - UT3 - UT1 - UT2J



L'entreprise partenaire et le
réalisateur industriel

Première équipe de recherche

L'entreprise partenaire et le
réalisateur industriel



ACTIVmap



Deuxième équipe de
recherche

Source de financement du
projet (2020-2024)

Troisième équipe de recherche

3. Objectifs et description des besoins

3.1 Problématique

Notre projet consiste à concevoir et à réaliser un enchaînement de traitement des données géographiques dans le contexte web qui permet d'exécuter les processus et des scripts établis par les chercheurs, aussi une application web pour l'objectif d'enchaîner des outils qui stockent des données, publier et administrer les couches et gérer les droits d'accès via les protocoles web, exécuter les algorithmes, afficher et modifier les résultats dans les pages web.

Autrement dit, ce projet il sert à implémenter un pipeline web qui a pour l'objectif de tester une architecture qui compose de plusieurs briques de traitement des données géographiques

chaque bloc de traitement on prépare des exemples d'application pour tester le fonctionnement et interopérabilité avec le reste de pipeline. Notre but c'est de donner une combinaison des outils qui sert à manipuler la base des données Openstreetmap via la modification et création des couches et des cartes et orienter ces données à impression en relief.

3.2 Les objectifs du projet

L'objectif de ce projet est l'élaboration d'une base théorique autour des besoins cités dans le cahier des charges, installer et comparer les différentes technologies de l'état de l'art puis valider de la solution proposée via une étude de cas technique aussi d'autres d'objectifs comme le déploiement et la configuration d'un serveur qui peut exécuter des scripts de traitement et de modification de flux des données, élaboration d'un schéma de stockage des couches données géographiques. Aussi le déploiement et la configuration d'un serveur de publication et personnalisation des couches des données géographiques. Également le développement d'une application web qu'interagir avec les différentes parties de l'architecture.

3.3 Missions



Figure 3: Diagramme des missions durant la période de stage

3.3.1 L'étude de faisabilité technique

Il s'agit des suggestions et des identifications des besoins fonctionnels ainsi que les contraintes à respecter avec la description des scénarios possibles pour chaque bloc de traitement de projet avec ses acteurs et ses entrées/sorties.

Également, la stabilisation et la suggestion des besoins non fonctionnels (les caractéristiques techniques), le choix des technologies qu'on peut adapter pour chaque bloc de pipeline à titre d'exemples des exigences (Open source, Rapidité, sécurité, la tolérance en panne, le traitement en temps réel etc.).

3.3.2 Conception et documentation

Participation à la modélisation de l'architecture générale de pipeline avec la structure globale de projet ACTIVmap et ces parties pour l'objectif de faire la conception détaillée, garder l'homogénéité, faciliter l'intégration et concevoir une comparabilité efficace avec d'autres parties.

La construction d'une conception technique détaillée de data-flow et work-flow de projet pour le but de déterminer la façon dont les logiciels fournissent les différentes fonctionnalités, ainsi d'avoir une organisation logique des composants de projet via la modélisation des diagrammes.

3.3.3 Implémentation et développement

L'une de mes missions est d'implémenter les tâches demandées et développer les fonctionnalités en utilisant les différents langages de programmation principalement Python et ces Framework comme Geodjango.

3.3.4 Validation des tâches

Aussi la validation des tâches réalisées avec d'autres participants/collaborateurs de projet et la suggestion des adaptations afin de se progresser en garde la comparabilité de différentes parties.

3.3.5 Intégration et déploiement

J'ai été amené à déplorer des solutions que ça soit pour tester dans le cadre d'état de l'art ou d'intégrer les nouvelles fonctionnalités réalisées dans l'enchaînement global de projet, apporter des modifications s'il est nécessaire, coordonner avec les autres équipes.

3.3.6 Support

J'ai été chargé d'effectuer des modifications d'amélioration en termes de performance en ajouteront des nouvelles fonctionnalités, aussi la correction et la réparation des blocs de projets techniquement et fonctionnellement sous la collaboration et suggestions des porteurs de projet.

4. Planning et déroulement du projet

La gestion du projet vise à organiser le bon déroulement de projet et de s'assurer que les étapes du processus de développement suivent le bon chemin. L'organisation de travail est un facteur essentiel de productivité et d'efficacité, elle permet de planifier les tâches, de mobiliser les ressources et de limiter les imprévus afin de rester dans la limite des objectifs et du périmètre. Pour cela nous nous sommes basés sur l'approche agile.

4.1 L'approche agile

Dans le contexte de notre projet où les dimensions de notre produit n'est pas fixe dès le début et où nous avons besoin de collaborer et dialoguer avec le reste des membres de l'équipe en quotidien pour pouvoir réussir toutes les étapes (fixation des besoins, et automatisation de la chaîne de déploiement), l'utilisation d'une approche agile est une priorité pour nous afin de réussir la mission dans les meilleures conditions. Cette approche repose sur la collaboration et l'autonomie des équipes pluridisciplinaires.

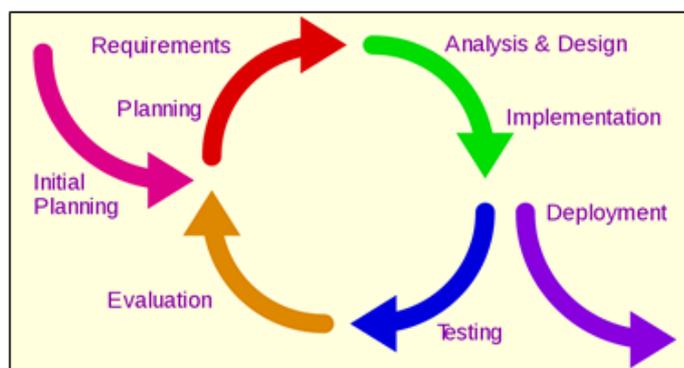


Figure 4: Processus de méthode Agile

Nous avons choisi la méthode agile pour les raisons suivantes :

- Les besoins sont inconnus à l'avance.
- Les technologies à utiliser sont inconnues à l'avance.
- Avoir une idée claire sur les différentes étapes du projet
- Entièrement développée et testée pour de courtes itérations
- Simplicité des processus
- Il s'appuie sur : la transparence, l'inspection et l'adaptation.

4.2 Planification des tâches

En suivant cette méthode le projet est organisé autour de « semestres » d'une durée allant généralement de deux à quatre semaines. Il débute souvent par le « semestre 0 », dédié à la réalisation de tous les travaux de préparation et de mise en place : conception et architecture, environnements de développement, outils de suivi et d'intégration.... Au début et la fin de chaque semestre l'équipe (développeurs et les chercheurs) est réunie afin de déterminer les user stories vont être traitées. Durant toute la durée du semestre, des discussions quotidiennes « daily meetings » sont organisées via Mattermost. Elles ont généralement lieu le matin et réunissent l'ensemble de l'équipe et ne durent que 15 minutes en discussion instantanée. L'objectif de ces réunions est de synchroniser l'équipe de façon à ce que chacun ait le même niveau d'information. A la fin du semestre, une démonstration du produit dans son état actuel est faite aux membres de l'équipe et une réunion avec les membres de IGN. C'est le moment pour toute l'équipe de rappeler les objectifs qui avaient été fixés et les fonctionnalités qui ont été réalisées et livrées. Enfin, l'équipe projet se réunit une dernière fois durant le semestre pour la rétrospective. C'est l'occasion de faire la liste des processus qui ont bien fonctionné durant le semestre et de ceux qui nécessitent d'être améliorés.

4.3 Les membres de l'équipe

Jean-Marie Favreau: Maître de conférences en informatique, Université Clermont Auvergne, chercheur à LIMOS également le porteur du projet ACTIVmap et le chef d'équipe s'intéressent à la problématique de mobilité des personnes en situation de la déficience visuelle que ce soit dans le cadre d'activités de recherche, dans le cadre associatif, personnel, ou par l'organisation d'événements de sensibilisation.

Guillaume TOUYA : Ingénieur chercheur des Travaux géographiques et Cartographiques de l'état et directeur du laboratoire COGIT, IGN.

Jérémy Kalsron : doctorant qui travaille à la construction des algorithmes génération textuelle à partir des données géographiques vient d'OpenStreetMap.

Samuel Braikeh : Ingénieur Technicien Administratif, il travaille aussi sur le projet OD4M.

Ridley campbell : étudiant en M2, stagiaire en PFE à IGN.

4.4 Les diagrammes de Gantt

4.4.1 Première diagramme de Gantt

Le premier diagramme sous le thème de familiarisation au domaine GIS et au contexte global de projet, les tâches cités dans ce premier diagramme intègrent principalement dans le cadre familiarisation avec le domaine d'application de projet qu'est la géolocalisation et la géomatique aussi la prise en main de certains logiciels de GIS.

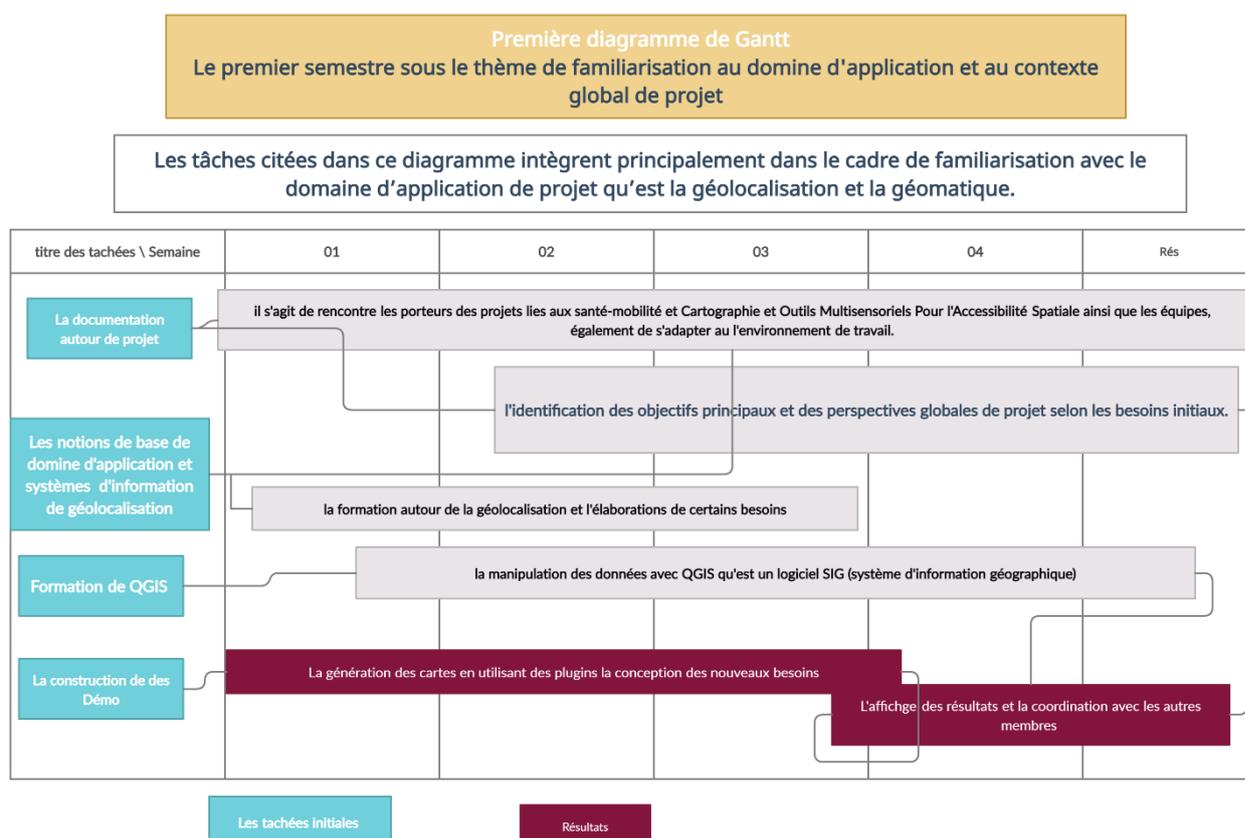


Figure 5: Première diagrammes de Gantt

4.4.2 Deuxième diagramme de Gantt

Les activités de ce semestre s'inscriront dans le cadre de l'état de l'art autour des solutions techniques déjà établies dans le même périmètre de projet, aussi la veille technologique et l'étude faisabilité technique des besoins initiaux, également la conception de première architecture. L'implémentation et tests des solutions proposés ainsi que la redéfinition de certains besoins et l'adaptation des autres. L'élaboration des nouvelles fonctionnalités.

Deuxième Diagramme de Gantt la réadaptation des besoins

Les activités de ce semestre s'inscriront dans le cadre de l'état d'art autour des solutions techniques déjà établies dans même périmètre projet, aussi la velle technologique et l'étude faisabilité technique des besoins initiaux, également la conception de première architecture.

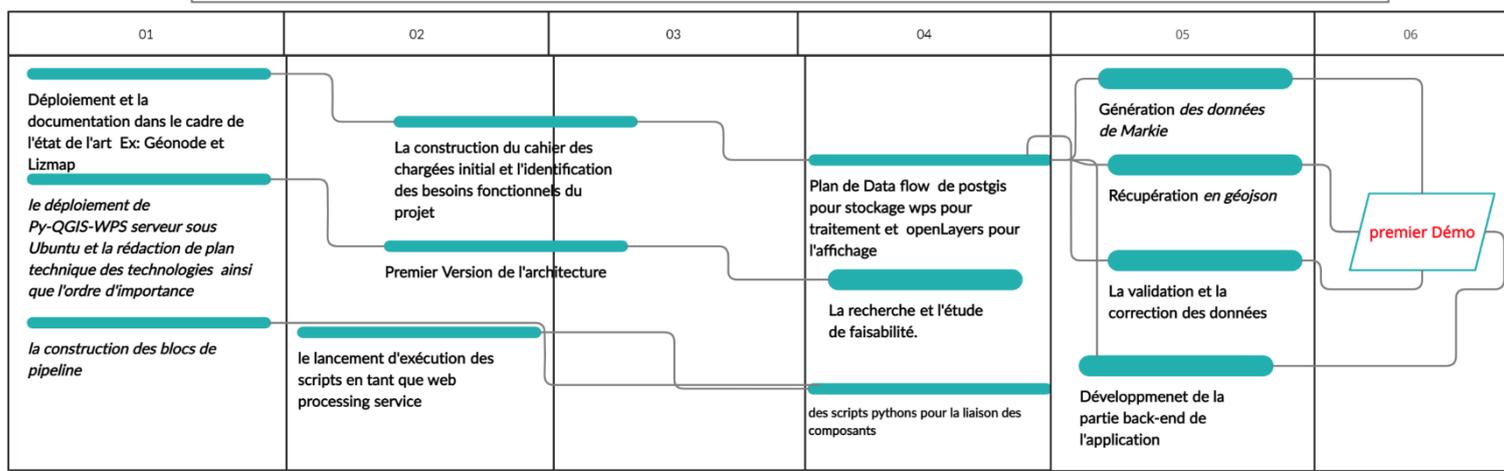


Figure 6: Deuxième diagrammes de Gantt

4.4.3 Troisième diagramme de Gantt

Ce troisième diagramme reflète les tâches effectuées dans le dernier semestre de mon stage duquel on a élaboré l'architecture de projet et validé ces composants via l'étude de cas, ainsi qu'on a proposé la solution retenue.

Le troisième diagramme de Gantt sous le thème de la mise en œuvre de l'architecture et de la validation de la solution proposée

Troisième diagramme concerne l'élaboration de la solution retenue et la vérification de son efficacité

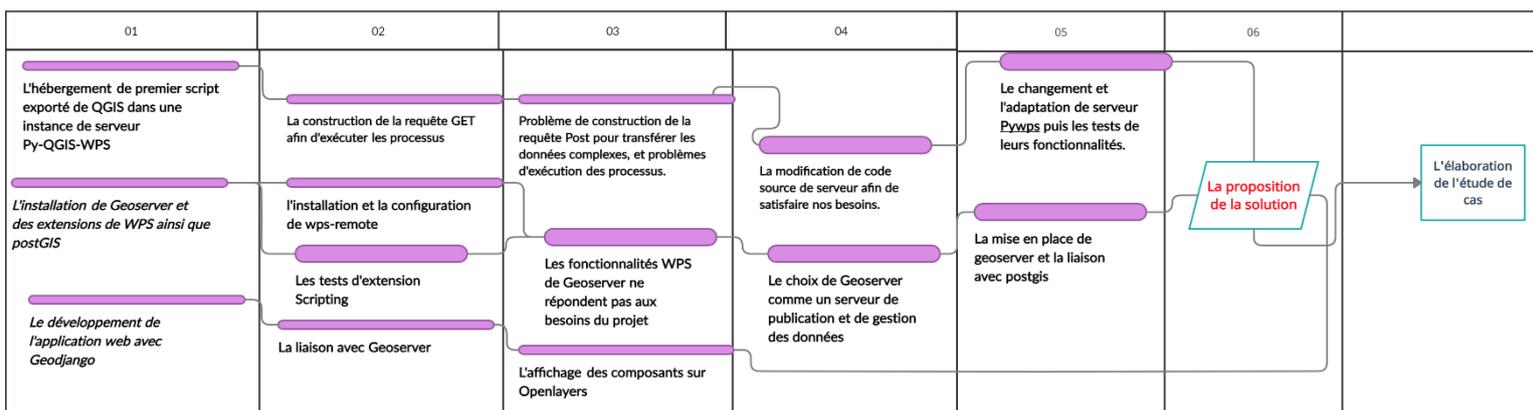


Figure 7: Troisième diagramme de Gantt

Conclusion

Ce premier chapitre représente le contexte général du projet, notamment un bref descriptif de l'organisme d'accueil et les objectifs du projet ainsi que la démarche adoptée pour la conduite de construction des besoins et son planning prévisionnel de réalisation. Le chapitre suivant exposera la présentation des fondements théoriques et l'état de l'art technique que nous utiliserons et les outils qui répondront aux centaines besoins.

Chapitre II : Les fondements théoriques

Introduction

Dans ce premier chapitre on va présenter revue des concepts liés à l'information géographique, sa description et sa manipulation au cœur des systèmes d'informations géographiques. Nous nous sommes focalisés sur les concepts liés à la problématique traitée ; l'enchaînement automatique de traitement des données géographiques dans le contexte web. Aussi ce chapitre consacré à l'état de l'art technique, nous présentons une synthèse des travaux techniques réalisés autour de la problématique de recherche traité ainsi que les différents outils explorés toute au long de cette période.

1. Revue des concepts

L'outil SIG reste l'environnement d'exploitation de l'information géographique organisée en base de données géographiques ou via l'utilisation des données de source libre comme OpenStreetMap, dans nos jours il est indispensable d'exploiter ces sources et de les consolider afin de satisfaire les différents besoins et exigeants des utilisateurs, tels que ; le niveau de détail, préférences, et de contribuer plus efficacement dans ces sources. La nouvelle aire de recherche consiste à satisfaire cette communauté qui utilise l'information géographique diversifiée en proposant des nouveaux outils ou bien en enrichissant cette base des données collaborative comme celui de OpenStreetMap. Via la modélisation des processus de la généralisation ou de la modification de données géographiques.

Dans ce chapitre, nous introduisons des concepts classiques liées à l'information géographique et des concepts spécifiques à la problématique traitée tels que, les données géographiques et les protocoles web.

1.1 L'information géographique

Nous allons présenter brièvement quelques notions essentielles relatives aux bases de données géographiques.

L'information géographique est l'information recueillie dans une base de données géographiques qui comprend une information relative à un phénomène du monde représenté par son aspect et ses caractéristiques, et sa localisation décrite dans un système de coordonnées de référence (**Denègre & Salgé, 1996**).⁹

⁹ https://fr.wikipedia.org/wiki/Information_g%C3%A9ographique

1.1.1 Représentation de l'information géographique

Il existe deux grands modes de représentation des données géographiques sous forme numérique : les représentations vecteur et raster (voir la **Figure I-1**).

- **Le mode raster** (ou maillé) se base sur le partitionnement de l'espace en un maillage constitué de cellules rectangulaires (et généralement carrées) appelées *pixels*. Chaque pixel porte une ou plusieurs informations descriptives caractérisant son contenu. La géométrie d'un objet géographique correspond à un ensemble de pixels. Ce mode de représentation est utilisé pour les photographies numériques.
- **Le mode vecteur** consiste à représenter la géométrie des objets géographiques sous la forme d'une primitive géométrique, qui peut être, en deux dimensions, un point (décrit par ses coordonnées dans un repère), une ligne (décrite par une suite de points), ou une surface (décrite par son contour qui est une ligne fermée). Les points constitutifs des lignes peuvent être reliés entre eux par des segments ou par des courbes plus complexes telles que des splines ou des courbes de Bézier.¹⁰

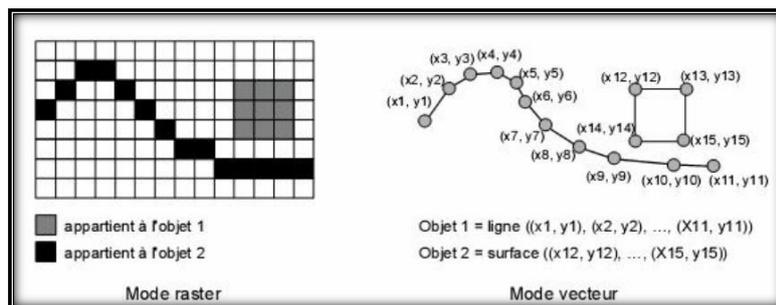


Figure 8: Deux objets représentés en mode raster et en mode vecteur

Une base de données géographiques est définie par (Ruas & Libourel, 2002) comme : "Une base de données géoréférencées c'est-à-dire de données qui possèdent des coordonnées permettant de les localiser, de connaître leur position sur un espace limité (la terre, un pays, une ville, un océan, mars, ...etc.)".¹¹

Une base de données géographiques est constituée d'un ensemble d'objets organisés en classes d'objets de même nature. Chaque objet de la base de données est décrit par :

¹⁰ Source : Intégration des Méthodes Multicritères et des Ensembles Flous dans un SIG pour Analyser l'Adaptabilité des Terres Agricoles : Application au Maïs Grain en Languedoc—Roussillon

¹¹ Action de recherche BDMUL

- Une géométrie : elle est une primitive géométrique vecteur : point, ligne (ruban) ou surface, c'est-à-dire sous la forme d'une suite structurée de coordonnées de points. Par exemple, une surface peut être représentée par la suite des points de son contour.
- Des données attributaires (champs, ou données sémantiques) : ces données attributaires permettent de compléter la description de l'objet, similaires à ceux que l'on trouve classiquement dans toute base de données. Par exemple, un tronçon routier peut porter un attribut donnant des informations sur le type de la route qu'il représente (départementale, nationale, autoroute, ...etc.).

Il existe principalement deux types de Base de Données Géographiques (BDG) selon la destination de création de telle base de données. On parle de Base de Données Cartographique (BDC) si l'objectif est de cartographier cette Base de données, en anglais c'est "Digital Cartographic Model". Par opposition, le terme "Digital Landscape Model" désigne une base de données géographique non destinée à être cartographiée. Notons qu'il est pertinent de parler de "l'échelle" d'une base de données cartographique, puisqu'une telle base est destinée à être affichée à une échelle donnée.

1.2 Système d'information géographique

L'homme a voulu, depuis longtemps, découvrir son environnement. Pour satisfaire ce besoin, il utilise des techniques afin de représenter et manipuler le flux d'informations géographiques. Différents supports d'informations sont utilisés : cartes topographique, images satellitaires, ...etc. Avec l'avènement des données numériques, l'information géographique a été intégrée dans des bases de données géographiques (BDG) et dans des outils spécifiques appelés systèmes d'information géographique (SIG). Ces deux outils se sont développés conjointement.

Des nombreuses définitions ont été données au système d'information géographique (SIG) Plus précisément deux définitions, semblant très importantes à citer. La première purement logicielle où un SIG est vu en tant qu'un outil informatique permettant de manipuler des données géographiques. Il dispose un système de gestion de base de données (SGBD), et un ensemble d'outils spécifiques dédiés à la composante spatiale des bases de données géographiques. Il permet de visualiser leur contenu en attribuant des symboles cartographiques aux différents objets de la base. Il permet ainsi d'effectuer des calculs et des requêtes complexes s'appuyant sur les géométries des objets mais aussi d'appliquer des transformations sur ces dernières. La plupart des SIG proposent un environnement de programmation permettant de coder et intégrer de nouvelles fonctions, par exemple, le ARCGIS dispose le langage VBA, et MapInfo utilise le

mpabasic (**EdiGéo,1992**). La seconde plus générale où le SIG est vu en tant que système d'information. Il est alors appelé SIRS (Système d'Information à Référence Spatiale). Il est défini comme un "ensemble organisé globalement comprenant des éléments (données, équipements, procédures, ressources humaines) qui se coordonnent, à partir d'une référence spatiale commune, pour concourir à un résultat" (**Bédard,1982**). Un SIG permet d'analyser les données grâce à des outils d'analyse spatiale. Les calculs peuvent être simples (longueur d'une ligne, distance euclidienne entre deux objets), ou plus complexes (calcul d'intersections entre objets, recherche du plus court chemin dans un réseau, etc.). Il permet également de transformer les données en modifiant leur géométrie. Là aussi, les outils de transformation peuvent être simples (édition d'une ligne point à point), ou plus complexes (algorithme de simplification du contour, etc.). La plupart des SIG disposent d'un langage de programmation qui permet de coder de nouvelles fonctions d'analyse ou de transformation. Aussi, les SIG ouvrent plusieurs possibilités qui satisfont les besoins croissants des utilisateurs comme l'aide à la décision, à la gestion du territoire, à la planification dans divers domaines¹² :

- Urbanisme : aménagement, gestion des plans d'occupation du sol (POS), gestion territoriale et municipale....
- Etudes économiques et socio-économiques : population, logement, géomarketing... Réseaux de communication et de transport : électricité, eau, routes, fleuves, réseaux souterrains, suivi et calcul d'itinéraires ... Applications militaires. Science de la terre : géologie, géotechnique....

1.3 Fonctions d'un SIG¹³

Un SIG intègre généralement cinq fonctionnalités de base dites :

- L'Acquisition pour la collecte des données grâce à des fonctions de saisie des données sous forme numérique.
- L'Archivage grâce à un système de gestion de bases de données (SGBD)
- L'Analyse par des fonctions de manipulation, croisement et transformation des données spatiales au moyen de requêtes dans le SGBD.
- L'affichage pour la restitution des résultats par des fonctions de mise en forme et de visualisation.

¹² Livre de la conception de SIG : méthode et formalisme / Dimos N. Pantazis, Jean-Paul Donnay.

¹³ <https://www.esrifrance.fr/sig4.aspx>

- L'Abstraction par des fonctions rendant compte de la modélisation de la réalité.

L'apport des SIG par rapport à la cartographie traditionnelle est d'une part le stockage d'une quantité importante de données et d'autre part une capacité de traitement de l'information accrue.

1.4 Organisation des données en couche¹⁴

Il est fortement nécessaire d'organiser ou de structurer l'information géographique sous forme de couches. On définit une couche dans un SIG comme étant un ensemble d'un ou plusieurs thèmes associés au même territoire. On peut donc avoir des couches mono-thèmes ou multi-thèmes selon les applications des SIG considérés. Par exemple, les thèmes : hydrographie, occupation du sol, orographie, agglomérations et géologie partagent le même territoire d'une commune. Bien qu'ils soient différents par leurs modes de saisie (méthode,

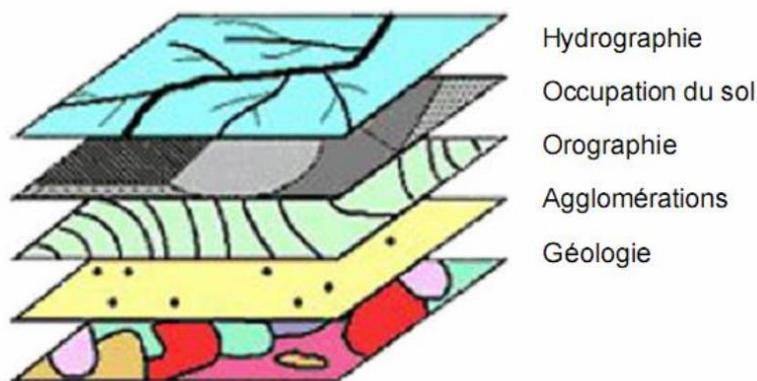


Figure 9: Organisation des données en couches

précision, échelle...etc.) et par la fréquence de leur mise à jour (parfois de façon séparée), pour des raisons d'analyse spatiale, ces thèmes peuvent être associés, non seulement par voie de leur affichage à l'écran, mais également à l'intérieur de la base de données.¹⁵

Le choix du nombre de couches ainsi que celui des thèmes à associer à une couche sera souvent un compromis entre efficacité lors de l'interrogation, difficulté de la construction et évolution de la couche. Il est clair qu'il ne faut pas mettre ensemble tous les thèmes dans une seule couche. Des thèmes consultés ensemble doivent constituer une couche. Mais une

¹⁴ Livre Couches fonctionnelles et Rôles : nouveaux concepts pour une modélisation cohérente et intégrale des informations géographiques

¹⁵ https://www.emse.fr/tice/uved/SIG/Glossaire/co/Organisation_donnees_1.html

couche comportant trop de thèmes est illisible à l'écran et prend trop de temps à créer et à mettre à jour, voire à consulter.

1.5 Les données géographiques dans le contexte web

OpenStreetMap (OSM) : est un projet collaboratif de cartographie en ligne qui vise à constituer une base de données géographiques libre du monde (permettant par exemple de créer des cartes sous licence libre), en utilisant le système GPS et d'autres données libres. Il a été mis en route en juillet 2004 par Steve Coast à l'University College de Londres. Par l'utilisation de moyens informatiques reposant sur Internet qui permettent l'intervention et la collaboration de tout utilisateur bénévole, OpenStreetMap relève de la géomatique 2.0, de l'information géographique bénévole et de la néo géographie, dont les outils composent le GeoWeb.¹⁶

1.6 Utilisation d'OpenStreetMap

Et puisque OpenStreetMap est principalement une carte du monde entier publiée gratuitement sous une licence de contenu libre naturellement été embarquées dans des solutions web. Construite par des contributeurs bénévoles, elle est éditable par tous et en constante évolution. Et aussi, la communauté fournit un ensemble d'outils très puissants qui permet de récupérer les données, de les publier facilement. Tous ces avantages, nous pourrions de considérer comme une source des données pour notre solution. On peut citer aussi d'autres qualités comme la complétude, la qualité spatiale, temporelle et sémantique ainsi que la cohérence. Pour bien traiter et exploiter cette source, il est nécessaire d'utiliser les services et les protocoles de L'OGC.

1.7 Les protocoles web de l'OGC adoptés

1.7.1 L'open geospatial consortium

L'Open Geospatial Consortium,¹⁷ ou OGC : est un consortium international pour développer et promouvoir des standards ouverts, les spécifications OpenGIS, afin de garantir l'interopérabilité des contenus, des services et des échanges dans les domaines de la géomatique et de l'information géographique. OGC est une organisation internationale à but non lucratif fondée en 1994.

¹⁶ <http://igm.univ-mlv.fr/~dr/XPOSE2012/OpenStreetMap/whatIsOsm.html>

¹⁷ http://igm.univ-mlv.fr/~dr/XPOSE2009/Services_Cartographiques/webservice.html

1.7.2 Les services et protocoles web de OGC

Avec les services Web de l'OGC (Open Geospatial Consortium), vous pouvez diffuser vos cartes et vos données sur le Web, dans un format ouvert, reconnu dans le monde entier. L'OGC émet des spécifications pour rendre les cartes et données disponibles sur le Web à quiconque possède une application cliente prise en charge.¹⁸

1.7.2.1 Web Map Service (WMS)

Les services WMS (Web Map Service) selon la documentation officielle de OGC il sert à diffuser des collections de couches sous forme de cartes de type image.

Les services WMS s'avèrent utiles si vous souhaitez que vos cartes soient disponibles en ligne dans un format ouvert et reconnu sur divers clients et plateformes. Les clients compatibles avec la spécification WMS peuvent afficher et utiliser votre service. Quatre versions de la spécification WMS ont été publiées jusqu'à présent : 1.0.0, 1.1.0, 1.1.1 et 1.3.0 (la plus récente).

Les applications clientes utilisent un service WMS en ajoutant des paramètres à l'URL du service. Les services WMS publiés sur Geoserver par exemple permettent d'effectuer les opérations suivantes :

1. Requête de métadonnées à propos du service (GetCapabilities),
2. Requête d'une carte de type image (GetMap),
3. Requête d'informations relatives aux entités de la carte (GetFeatureInfo [facultatif]),
4. Requête de styles définis par l'utilisateur (GetStyles),
5. Requête de symboles de légende (GetLegendGraphic).

Un service WMS ne doit pas forcément prendre en charge toutes ces opérations. Il doit toutefois prendre en charge au moins les opérations GetMap et GetCapabilities pour être un service WMS de base, et prendre en charge l'opération facultative GetFeatureInfo pour être un service WMS interrogeable. Les opérations GetLegendGraphic et GetStyles s'appliquent uniquement dans un service WMS.¹⁹

¹⁸ <https://infoterre.brgm.fr/page/geoservices-ogc>

¹⁹ <https://www.rncan.gc.ca/cartes-outils-et-publications/outils/normes-politiques-operationnelles-geospaciales/service-des-paves-cartographiques-web-wmts/service-de-cartes-web-wms/8939>

1.7.2.2 Web Feature Service (WFS)

Selon la documentation de OGC les services WFS (Web Feature Service) sert à afficher des données sous forme d'entités vectorielles, La diffusion de vos données par l'intermédiaire d'un service WFS permet aux applications compatibles avec les services Web d'accéder aux entités géographiques de votre carte ou géo-database. Contrairement au service WMS, qui renvoie une image de carte, le service WFS renvoie des entités avec la géométrie et les attributs que les clients peuvent utiliser dans n'importe quelle analyse géospatiale. Les services WFS prennent également en charge des filtres qui vous permettent d'exécuter des requêtes spatiales et attributaires sur les données.

- Permet d'obtenir ou demander des données basées sur des contraintes spatiales et non-spatiales.
- Fournit les fonctionnalités de bases que la plupart des applications SIG demandent.
- Permet de mettre à jour une instance des données. Cela permet à un utilisateur de changer un attribut dans un ensemble de données sélectionné par exemple : Si on a besoin de montrer qu'une route est fermée.
- Permet d'effacer un instant de donnée si elle n'est plus valide ou nécessaire.
- Créer de nouvelles données. Cela permet à un utilisateur d'ajouter des données à une carte.

1.7.2.3 Les trois requêtes fournies par WFS

- GetCapabilities Requête très similaire à celle présente dans WMS, Permet au client de découvrir quels services et données le WMS supporte.
- DescribeFeatureType Permet au client de déterminer le schéma d'un ensemble de donnée, Il est ainsi possible de savoir ce que on va recevoir par la suite ou de savoir comment insérer de nouvelles données.
- GetFeature : Retourne les données demandées par le client, Le client peut définir sa requête en utilisant des filtres spatiaux et non spatiaux afin de limiter le nombre de données retournées.²⁰

²⁰ Livre : Modèle pour un serveur de données géographiques Nissrine Souissi , Michel Mainguenaud

1.7.2.4 Web processing service (WPS)

C'est un service pour le traitement géospatial, La spécification WPS (Web Processing Service) de l'OGC est une spécification internationale de diffusion et d'exécution de traitement géospatial sur le Web. Vous pouvez créer un service WPS en activant la fonctionnalité WPS lors de la publication d'un service de géo-traitement.

Les services WPS sont utiles si vous souhaitez que vos services de géo-traitement soient disponibles dans un format ouvert et reconnu sur divers clients et plateformes. Les clients compatibles avec la spécification WPS 1.0.0 peuvent afficher et utiliser votre service.²¹

1.7.2.5 Types de données prenant en charge par les services WPS

Seuls certains types de données peuvent être utilisés en tant que paramètres en entrée et en sortie pour les services WPS. Ils sont dérivés des types de données pris en charge pour la géo traitement. Le tableau ci-dessous répertorie tous les types de données en entrée et en sortie pour les services WPS, ainsi que leurs sous-types équivalents.

Tableau 1: les types des données de WPS

Type de données	Sous-type
Standard	Longue
	Double
	Chaîne
	Date
	Booléen
	Unité linéaire
Jeu d'entités	GML 3-SF
Jeu d'enregistrements	GML 3-SF
Fichier	.zip
	Binaire
	XML
	Texte brut
Raster	Image/TIFF
	Binaire (en entrée uniquement)

²¹ <https://enterprise.arcgis.com/fr/server/latest/publish-services/windows/wps-services.html>

2. Etat de l'art technique

2.1 Technologies et outils

Dans cette section, on va traiter les différents logiciels/technologies opens sources que on a testés durant cette période de stage. Certains d'entre eux sont utilisés pour la mise en œuvre des solutions qui supporte WPS sur l'environnement Web dans le coté serveurs ou le coté client, tandis que d'autres des projets déjà construits au moment d'élaboration des besoins. Les produits logiciels utilisés pour La mise en œuvre de WPS côté serveur est illustrée dans le tableau ainsi que leur description détaillée est donnée ci-dessous :

S. No.	Software/ Project	Langage de Programmation	Platforms
1.	GeoServer WPS	Java	Windows, Linux, Mac OS,
2.	520 North	Java	Windows, Linux, Mac OS
3.	ZOO-Project	Java, Python, C, C++, PHP, JavaScript	Windows, Mac OS, Unix
4.	Deegree WPS	Java	Windows, Linux, Mac OS
5.	PyWPS	Python	Platform Independent
6.	Py-Wps-qgis	Python	Lunix

Tableau 2:les projets open source qui implémente WPS coté serveur, source les sites officiels des projets

2.1.1 GeoServer

GeoServer est un serveur logiciel open source écrit en Java qui permet aux utilisateurs de partager et de modifier des données géospatiales. Conçu pour l'interopérabilité, il publie des données à partir de n'importe quelle source de données spatiales majeure en utilisant des normes ouvertes.



En tant que projet communautaire, GeoServer est développé, testé et soutenu par un groupe diversifié d'individus et d'organisations du monde entier. GeoServer est l'implémentation de référence des normes de service d'entités Web (WFS) et de service de couverture Web (WCS) de l'Open Geospatial Consortium (OGC), ainsi qu'un service de carte Web (WMS) certifié conforme à hautes performances. GeoServer constitue un composant essentiel du Web géospatial.²²

²²<http://geoserver.org/>

2.1.2 GeoNode

GeoNode est un système de gestion de contenu pour les données géospatiales qui fournit la création, le partage et l'utilisation collaborative des données géospatiales. Les jeux de données peuvent être téléchargés dans différents formats, les cartes peuvent être modifiées, stylisées et agrégées grâce à des outils basés sur le navigateur, cartes et métadonnées peuvent être publiées et cherchées, et les révisions, évaluations et commentaires peuvent être renseignés par les utilisateurs. GeoNode est construit sur : GeoServer, GeoExplorer, pycsw, Django et GeoExt.²³



2.1.3 Lizmap

Lizmap est Un Web SIG libre pour des projets QGIS, Créé par la société 3liz en 2011, il comporte Un plugin QGIS et un client Web et un QGIS Serveur comme serveur cartographique.



2.1.4 Zoo Project

ZOO-Project fournit un Framework pour le Web Processing Service (WPS) adapté pour les développeurs afin de créer et chaîner plusieurs Web Processing Services. Un serveur WPS fournit un accès web à des fonctions qui lancent des algorithmes spatiaux. ZOO-Project supporte de nombreux langages de programmation et contient deux applications de démos utilisant des fournisseurs de services simples d'outils spatiaux (un module avec bibliothèque C partagée et un avec Python) basés sur les bibliothèques GEOS et OGR.



ZOO est composé de trois parties :

Kernel ZOO : Un noyau côté serveur en C qui rend possible le fait de gérer et chaîner des webservices codés dans différents langages de programmation.

Services ZOO : Une suite d'exemples de webservices en évolution basée sur diverses bibliothèques Open Source.

API ZOO : Une API JavaScript côté serveur capable d'appeler et de chaîner les services ZOO, qui rend le développement et le chaînage de processus plus simple.

²³ https://live.osgeo.org/fr/overview/geonode_overview.html

ZOO est basé sur un ‘noyau de service WPS’ qui constitue le cœur du système de ZOO (c’est à dire le “ZOO Kernel”). Ce dernier est capable de charger des bibliothèques dynamiques et de les manipuler au gré des demandes des services web. Le noyau ZOO est écrit en langage C, mais supporte de nombreux autres langages de programmation dans le but de se connecter à de nombreuses bibliothèques et par dessus tout de simplifier le travail du développeur final de services web.

- Serveur WPS Multi langages.
- Création de services web simple et procédure de déploiement.
- Chaînage simple de WPS avec l’API de ZOO.

2.1.5 Deegree Project

The Deegree est un logiciel open source pour les infrastructures de données spatiales et le web géospatial. deegree est une application Java pour servir des cartes (et des données vectorielles) que d’autres clients peuvent afficher. Les composants disponibles sont : Services Web : WFS, WMS, WMTS, CSW, WPS.²⁴



2.1.6 PyWPS

PyWPS fournit un accès web a des opérations spatiales personnalisées (aussi appelés Processes) via le standard Web Processing Service (WPS). Les process sont écrits dans le langage de programmation Python et peuvent intégrer des outils tels que GRASS GIS, R, GDAL/OGR, proj et d’autres bibliothèques avec des liens Python. Les données sources peuvent être fournies avec des requêtes WPS ou accédées à partir de données sur le serveur WPS. Ces caractéristiques²⁵ sont simple à entretenir aussi rapide à conduire ainsi que capable de transporter Py-QGIS-wps



2.1.7 Py-QGIS-WPS

Py-QGIS-WPS est une implémentation de la norme Web Processing Service de l’Open Geospatial Consortium basée sur l’API de traitement QGIS. Cette implémentation permet d’exposer et d’exécuter sur un serveur : les algorithmes de traitement QGIS disponibles sur Desktop, les modèles et scripts de traitement QGIS version bureautique.²⁶



²⁴ <https://www.deegree.org/>

²⁵ <https://pywps.readthedocs.io/en/latest/>

2.2 Étude comparative de l'état de l'art technique

Caractéristique\ Solution	WPS Geoserver	Py-QGIS-wps	PyWPS	Deegree Project	Zoo Project	52°North
Supporte des modèles exportés de QGIS	Non	Oui	Non	Non	Non	Non
Documentation	Complete et suffisante	Pas suffisante	Suffisante	Suffisante	Existe	Existe
Communauté	Existe	Très nombreux	Existe	Très nombreux	Existe	Existe
L'interface d'administration	Couvert toutes les parties de la gestion	N'existe pas	Une interface web pour afficher les processus hébergés	Existe	Pour certains composants et API.	N'existe pas
Le constructeur de requêtes Post « XML »	Existe	N'existe pas	N'existe pas	N'existe pas.	Oui via WPS client	Oui via WPS client
Stable	Stable	En cours de développement	Stable	En cours de développement	Stable.	Stable
Open source	Oui	Oui	Oui	Oui	Oui	Oui
Les langages supportés	Java& Maven	Python	Oui en python	Java	Supporte multi-langages (C, Python et JavaScript)	Java, python
L'intégration avec d'autres parties de projet	Wfs, wms et les couches existe en même serveur	L'intégration via les protocoles web comme http, https	Via le protocole FTP pour la gestion des fichiers, http pour le flux des données	Via les requêtes http	Propose un package des APIs serviront à déployer, exécuter les processus.	Propose JavaScript API pour WPS et WPS clients.

²⁶ <https://www.3liz.com/en/opensource.html>

2.2.1 La méthodologie suivie

Le marché de la géométrie et surtout qui compose des fonctionnalités de web service est dominé en grande partie par les solutions propriétaires. Ces derniers sont certes efficaces, mais leur code est jalousement gardé par leurs concepteurs. La solution alternative réside dans les logiciels libres et open source. Ceux-ci prônent une philosophie axée sur le partage et le soutien de la communauté.

Un grand Nombre d'entre eux sont malheureusement inconnus du grand public. Dans cette étude, nous vous présentons une large sélection des projets libres qui remplaceront aussi bien certaines solutions propriétaires déjà commercialisées.

On a basé pour construire cette étude sur trois éléments principaux :

- ✓ Les projets supportent les standards d'OGC pour développer et promouvoir des standards ouverts, afin de garantir l'interopérabilité des contenus, des services et des échanges, favoriser la coopération entre développeurs, fournisseurs et utilisateurs, basé sur le principe du consensus volontaire.
- ✓ Le financement de projet ainsi que l'aide initiale, la nature de développement que ça soit par le personnel de l'entreprise, communautés de développement indépendantes collaborateurs, communautés d'utilisateurs, partage de ressources.
- ✓ Compatibilité et longévité des projets puisque les données sont changeantes il indispensable de prise en conscience le désir de compatibilité comme
 - Indépendance à l'OS (Windows, Mac, Linux).
 - Import / export de fichiers ESRI (Shape, GML, Geojson etc.).
 - Compatibilité avec d'autres logiciels (PostGIS, etc.) et d'autres SIG libres.
- ✓ Un autre facteur très important c'est la longévité du programme à titre d'exemple :
 - Développement d'une large communauté d'utilisateurs/programmeurs.
 - Généralisation des fonctionnalités (sortir du spécifique).
 - Passerelles avec d'autres SIG libres complémentaires

Pour les ressources qu'on a utilisées toute au long de cette étude se divise en deux parties, une partie consacrée au forum, l'autre est consacré à l'accès aux tutoriels, pour la partie des forums on peut citer ces facteurs :

- ✓ Des forums parce qu'ils représentent une base des témoignages considérables qui se divise en forum des systèmes de questions/réponses au sein des communautés, résolutions de problèmes d'utilisation, d'installation et l'échange de compétences.
- ✓ Discussion en temps réel avec des développeurs parfois possibles.
- ✓ Vérification que le problème n'a pas déjà été posté et la discussion des solutions.

Dans l'autre côté, on a l'accès aux tutoriels et à la documentation officielle, car ce type des ressources joue un rôle primordial, il doit comporter différents types de guides comme le didacticiel d'utilisation, les courantes pas à pas et workshops, des démonstrations vidéo ainsi que des tutoriels de développement, on a aussi la disponibilité des guides d'utilisation via les sites internet officiels des logiciels, des forums ou via les moteurs de recherches.

Aussi, on tester ces projets via les installations techniques, en adéquation avec nos besoins et l'infrastructure locale de laboratoire LIMOS.

3. Discussion autour des extensions de Web Processing Service de Geoserver

3.1 Geoserver wps-remote

GeoServer WPS permet de découvrir, d'exécuter et de surveiller les processus s'exécutant sur une ou plusieurs machines distantes, en les exposant via le protocole WPS et en permettant des exécutions synchrones et asynchrones de ceux-ci, avec un suivi éventuel de la progression.

Le processus distant peut être n'importe quoi, à partir d'un script Python ou d'un exécutable en ligne de commande. La seule contrainte est d'avoir un composant distant capable de gérer quelques RPC, comme run, Progress, Complete (ce qui signifie collecter et envoyer le résultat à la machine GeoServer), toutes les communications et la commande s'effectuent via le protocole XMPP, en tant que système de communication multilingue approprié.

3.1.1 Les contraintes techniques envisageables

Pour déployer cette solution, il est indispensable de déployer une version de serveur qui comporte le plugin WPS-remote dans le système d'exploitation Centos, mettre en place Serveur Openfire XMPP, un système de fichiers partagé NFS, une base des données PostgreSQL, Framework WPS Python Wrapper à distance puis la lisons avec des processus « Gdal Contour » à distance.

Les inconvénients majeurs de cette solution sont la complexité très élevée de l'architecture ainsi que la communauté qui ne supporte plus cette architecture, également les contraintes de l'interopérabilité et la mise en place des composants.

3.2 Scripting de Geoserver

L'extension de script GeoServer permet aux utilisateurs d'étendre GeoServer de manière dynamique en écrivant des scripts dans des langages autres que Java.

Le seul langage de script distribué sous forme de package à télécharger est Python. Cette extension est une extension communautaire.²⁷

3.2.1 Les contraintes techniques envisageables

L'obstacle majeur concerne l'absence de l'extension `gs-script-web.jar` dans les ressources officielles de geoserver qui empêche l'affichage de page Script dans l'espace d'administration de serveur. Le seul conseil de la communauté est de le télécharger depuis le dépôt de Maven, mais cette solution ne marche plus car le package est obsolète. Deuxième contrainte porte sur l'absence de la possibilité d'installation des bibliothèques fondamentales de traitement des données géographiques pour exécuter les processus en tant que web service. La troisième contrainte en relation avec l'intégrité envers d'autres composants de pipeline, et l'ordre d'exécution.

Conclusion

Cette exploration des différents outils techniques menés par différentes communautés de traitement des données géographiques à travers le monde, nous a permis d'identifier les différentes approches et solutions opens sources adoptées dans le domaine de la géomatique que ça soit pour la création des processus de traitement, le stockage, la publication et l'affichage des données.

L'idée qui nous a envahi en étudiant de près ces travaux, est de trouver une architecture optimale qui va nous servir à réaliser notre cahier des charges.

Dans le prochain chapitre, nous abordons la solution retenue et les outils supportés qui satisfont les besoins techniques et théoriques du projet.

²⁷ <https://docs.geoserver.org/latest/en/user/community/scripting/py/index.html>

Chapitre III : La mise en œuvre et la solution retenue

Introduction

Comme déjà spécifié, pour mettre en pratique notre application, nous avons besoin d'un certain outil de développement ainsi qu'un ensemble des technologies. Cette section a pour but de présenter la solution proposée et les moyens nécessaires pour réaliser notre étude de cas afin de vérifier les concepts théoriques.

1. Solution proposée

Pour remédier aux problèmes précédemment cités, la solution qu'on a établie porte principalement sur une application web distribuée compose de plusieurs blocs chacun à son rôle bien précis, on a décomposé ce principe applicatif en services indépendants, afin de simplifier les modifications dans le futur et de rendre l'application plus puissante. L'application devient alors une suite de services (des micro-services). Ces services autonomes sont amenés à communiquer entre eux par le biais d'un protocole simple tel que HTTP et celui de OGC. La complexité de l'application est alors décomposée en un ensemble de modules faciles à maintenir et à comprendre.²⁸

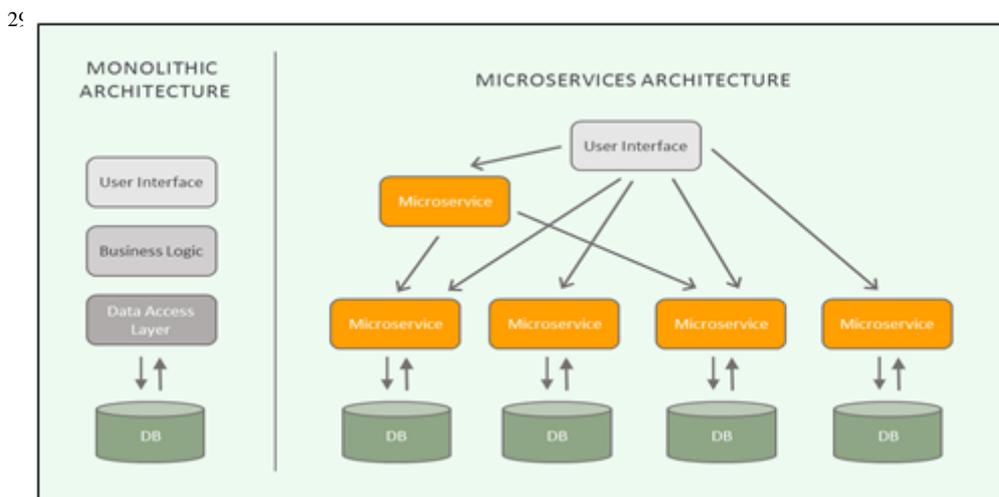


Figure 10: Architecture micro services.

²⁸ <https://aymax.fr/pourquoi-une-architecture-de-microservices/>

1.1 Description du solution proposée

L'un des critères importants c'est de trouver un serveur web offriront certaines fonctionnalités techniques et qui peut héberger des scripts ayant la possibilité d'appliquer des traitements aux flux des données d'origine Openstreetmap via l'exploitation des librairies de python comme Geopandas qui gérant des fonctions pour intégrer et analyser de la donnée dans des chaînes de traitement ainsi que de faire la manipulation des objets spatiaux avec des fonctionnalités statistiques pour ces raisons on a choisi le serveur PyWPS, puis d'avoir une base des données qui va stocker les couches des données qui vont être des entrées des processus, aussi un serveur qui nous donne la main de publier les couches via les protocoles de OGC, et la partie web qui va relier les différentes parties d'une part et d'afficher les résultats d'autre part. les différents tests qu'on a réalisés durant le stage nous a amenée à proposer la architecture suivante.

1.2 Architecture générale

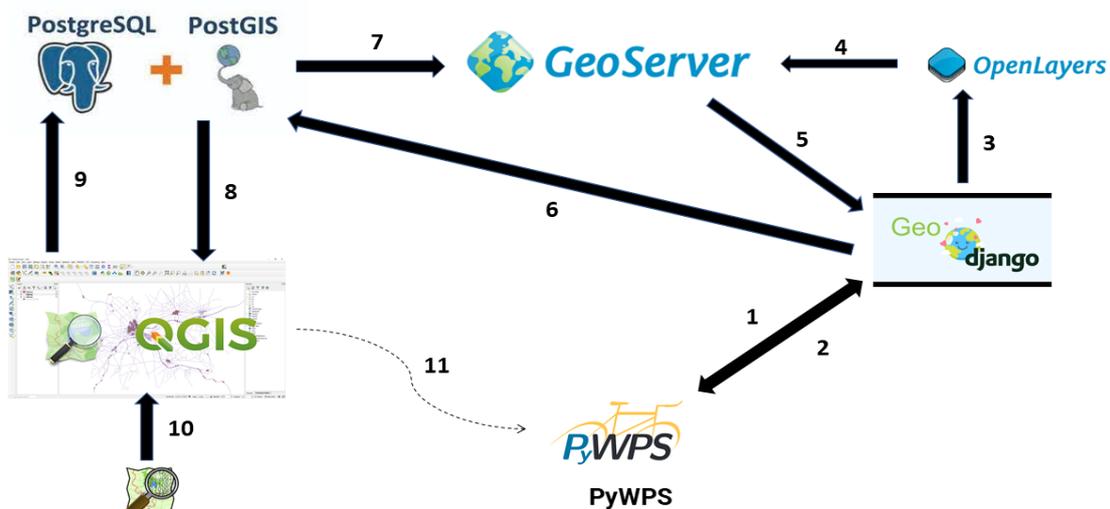


Figure 11: L'architecture générale de projet

Pour mettre en exergue notre solution proposée nous avons élaboré ce schéma qui englobe les différents blocs de traitement on peut distinguer trois blocs principaux, première partie correspond à l'axe des données depuis Openstreetmap vers la base des données Postgis, puis les deux serveurs web (GeoServer et Py Wps) et enfin l'application backend pour plus de clartés on cite la signification des nombres mentionnés dans le schéma :

- 1) La transmission de la requête en XML d'exécution de script/processus WPS.
- 2) La transmission de résultat d'exécution via le protocole WFS.
- 3) L'affichage des résultats au format d'une carte et des couches.
- 4) La requête de sélection des couches des données publiées depuis Geoserver via WPS ou WFS protocole.

- 5) Geoserver retourne le résultat de la sélection qui peut être comme input des processus ou ce flux des données sera affiché au OpenLayers.
- 6) L'enregistrement des données dans la base des données, ou l'envoi des requêtes de la création, modification ou suppression des données.
- 7) La publication des couches dans geoserver et la spécification des droits d'accès.
- 8) Une piste facultative qui sert à récupérer des données sous QGIS Desktop pour faire des traitements supplémentaires.
- 9) La possibilité d'enregistrer les résultats dans la base données.
- 10) L'exportation des données nécessaire depuis OpenstreetMap pour quel soit accessible depuis d'autre briques d pipeline.
- 11) QGIS peut avoir un plugin WPS Client interagit avec les processus.

1.3 Protocoles utilisées

Ce diagramme représente les protocoles utilisés afin de circuler le flux des données dans l'application et enchaîner les différentes briques de pipeline, la plupart des protocoles orientent le traitement des données géographiques et appartiennent à OGC, pour WFS sert à envoyer les données vectorielles, pour WMS sert à envoyer les données en images et pour WPS c'est un protocole pour l'exécution des processus en tant que web service.

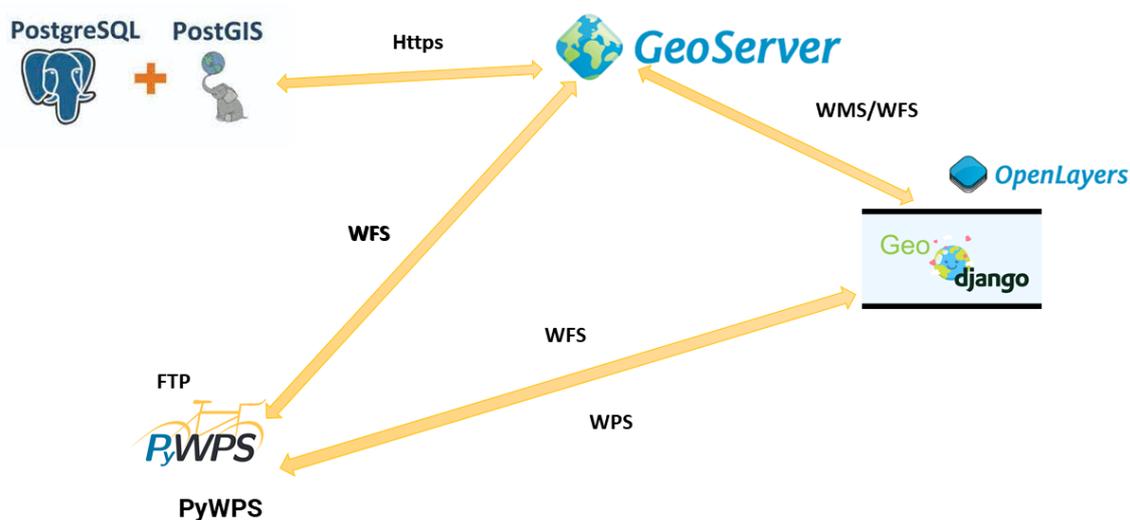


Figure 12: Les protocoles web assurant la circulation de flux des données

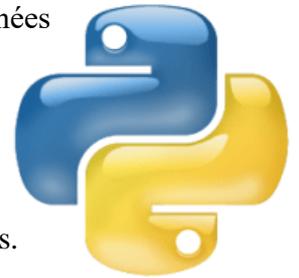
2. Technologies et outils de développement

Ces outils présentés dans ce chapitre sont utilisés pour satisfaire les besoins de projet et ils sont utilisés pour la construction de l'étude de cas, certains d'entre eux participent directement dans le projet, d'autres d'une manière indirecte.

2.1 Technologies et langages de programmation

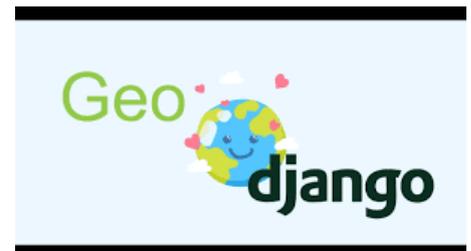
2.1.1 Python

Python est le langage de programmation open source le plus employé par les informaticiens. Ce langage s'est propulsé en tête de la gestion d'infrastructure, d'analyse de données ou dans le domaine du développement de logiciels. En effet, parmi ses qualités, Python permet notamment aux développeurs de se concentrer sur ce qu'ils font plutôt que sur la manière dont ils le font. Il a libéré les développeurs des contraintes de formes qui occupaient leur temps avec les langages plus anciens. Ainsi, développer du code avec Python est plus rapide qu'avec d'autres langages. Il reste aussi accessible pour les débutants, à condition de lui consacrer un peu de temps pour la prise en main. De nombreux tutoriels sont d'ailleurs disponibles pour l'étudier sur des sites Internet spécialisés ou sur des comptes YouTube. Sur les forums d'informatique, il est toujours possible de trouver des réponses à ses questions, puisque beaucoup de professionnels l'utilisent.³⁰



2.1.2 GeoDjango

GeoDjango est un module contribué de Django qui le transforme en un système géographique Web de grande envergure. L'objectif de GeoDjango est de faciliter le plus possible la création d'applications Web géographiques telles que des services basés sur la géolocalisation. Ses fonctionnalités comprennent : Des champs de modèle Django pour les objets géométriques OGC les données matricielles. Des extensions de l'ORM de Django pour les requêtes et les manipulations des données spatiales.³¹



³⁰<https://www.journaldunet.fr/web-tech/dictionnaire-du-webmastering/1445304-python-definition-et-utilisation-de-ce-langage-informatique>

³¹ <https://docs.djangoproject.com/fr/3.2/ref/contrib/gis/tutorial/>

Des interfaces Python faiblement liées et de haut niveau pour les opérations géométriques SIG et matricielles et la manipulation de données dans divers formats. L'édition de champs géométriques dans l'interface d'administration.

2.1.3 OpenLayers

OpenLayers est un logiciel libre, publié sous licence BSD. Il constitue une bibliothèque de fonctions JavaScript assurant un noyau de fonctionnalités orienté vers la mise en place d'applications clientes Web cartographiques fluides. OpenLayers permet d'afficher des fonds cartographiques tuilés ainsi que des marqueurs provenant d'une grande variété de sources de données. Une partie de cette bibliothèque permet aussi de gérer l'ergonomie proposée à l'utilisateur, mais ce n'est pas directement son rôle.³²



2.1.4 La base des données PostGIS

PostGIS est une extension (plugin) du SGBD PostgreSQL, qui active la manipulation d'informations géographiques (spatiales) sous forme de géométries (points, lignes, polygones), conformément aux standards établis par l'Open Geospatial Consortium. Il permet à PostgreSQL d'être un SGBD spatial (SGBDs) pour pouvoir être utilisé par les systèmes d'informations géographiques. Le nom provient de la contraction de PostgreSQL et de GIS (acronyme anglais de SIG). Concrètement, PostGIS permet le traitement d'objets spatiaux dans PostgreSQL, autorisant le stockage des objets graphiques en base de données pour les SIG.³³



2.2 Outils et environnements de travail

2.2.1 QGIS

Lancé en 2002, QGIS est un Système d'Information Géographique open source, librement accessible et donc gratuit, distribué sous licence publique générale GNU. Ce



³² <https://openlayers.org/>

³³ <http://www.postgis.fr/chrome/site/docs/workshop-foss4g/doc/PostGISIntro.pdf>

programme est reconnu pour sa vaste palette d'outils, la multitude de formats gérés et la simplicité de son interface. Parmi ses qualités, on soulignera l'accès aux fonctionnalités d'autres logiciels libres inclus lors de l'installation (p. ex. GRASS et SAGA) ou ajoutés indépendamment (p. ex. PostgreSQL). QGIS est compatible avec Linux, Unix, Mac OS X et Windows.³⁴

2.2.2 Docker

Docker est un logiciel libre permettant de lancer des applications dans des conteneurs logiciels. Docker permet la mise en œuvre de conteneurs s'exécutant en isolation, via une API de haut-niveau. Construit sur des capacités du noyau Linux (surtout les cgroups et espaces de nommage), un conteneur Docker, à l'opposé de machines virtuelles traditionnelles, ne requiert aucun système d'exploitation séparé et n'en fournit aucun. Il s'appuie plutôt sur les fonctionnalités du noyau et utilise l'isolation de ressources (comme le processeur, la mémoire, les entrées et sorties et les connexions réseau) ainsi que des espaces de noms séparés pour isoler le système d'exploitation tel que vu par l'application. Docker accède aux capacités de virtualisation du noyau Linux, soit directement à travers la bibliothèque runc (disponible depuis Docker 0.9), soit indirectement via libvirt, LXC (Linux Containers) ou systemd-nspawn.



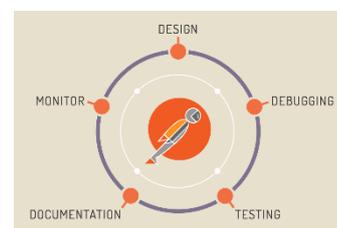
2.2.3 Le serveur d'applications Tomcat

Pour faire fonctionner geoserver, nous avons besoin de mettre en place un serveur d'applications, Nous allons utiliser la version Tomcat 9.



2.2.4 Postman

Postman permet de construire et d'exécuter des requêtes HTTP, de les stocker dans un historique afin de pouvoir les rejouer, mais surtout de les organiser en Collections. Cette classification permet notamment de regrouper des requêtes de façon « fonctionnelle » (par exemple enchaînement d'ajout d'item au panier, ou bien un processus d'identification).

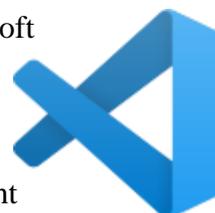


³⁴ https://live.osgeo.org/archive/6.0/fr/overview/qgis_overview.html

Postman assure également la gestion des Environnements, qui permet de contextualiser des variables et d'exécuter des requêtes ou des séries de requêtes dans différentes configurations (typiquement : dev, recette, prod).

2.2.5 Visual Studio

Visual Studio Code est un éditeur de code extensible développé par Microsoft pour Windows, Linux et macOS². Les fonctionnalités incluent la prise en charge du débogage, la mise en évidence de la syntaxe, la complétion intelligente du code, les snippets, la refactorisation du code et Git intégré. Les utilisateurs peuvent modifier le thème, les raccourcis clavier, les préférences et installer des extensions qui ajoutent des fonctionnalités supplémentaires. Le code source de Visual Studio Code provient du projet logiciel libre et open source VSCode de Microsoft publié sous la licence MIT permissive, mais les binaires compilés sont des logiciels gratuits pour toute utilisation. Dans le Stack Overflow 2019.



2.2.6 Geojson.io

geojson.io est un outil simple et rapide pour créer, afficher et partager des cartes. geojson.io est nommé d'après GeoJSON, un format de données open source, et il prend en charge GeoJSON - et d'autres types aussi comme KML, GPX, CSV, GTFS, TopoJSON.



2.2.7 OpenStreetMap

OpenStreetMap (OSM) est un projet collaboratif de cartographie en ligne qui vise à constituer une base de données géographiques libre du monde (permettant par exemple de créer des cartes sous licence libre), en utilisant le système GPS et d'autres données libres. Il est mis en route en juillet 2004 par Steve Coast à l'University College de Londres.



Par l'utilisation de moyens informatiques reposant sur Internet qui permettent l'intervention et la collaboration de tout utilisateur bénévole, OpenStreetMap relève de la géomatique 2.0, de

l'information géographique bénévole et de la néogéographie, dont les outils composent le GeoWeb.³⁵

Conclusion

Au cours de ce chapitre, nous avons présenté la solution retenue et les outils utilisée pour réaliser notre projet. Nous passerons dans le chapitre suivant l'étape d'étude de cas de pipeline.

³⁵ <https://www.openstreetmap.org/>

Chapitre IV : Étude de Cas

Introduction

Dans ce chapitre on va présenter l'étape de la réalisation et de concrétisation des concepts théoriques des besoins de projet. C'est la phase de développement, celle où il faut produire le code nécessaire afin de valider l'architecture et la solution proposées.

Dans cette partie, nous allons présenter l'ensemble des composants de l'étude de cas qu'a pour l'objectif de monter et présenter le principe de fonctionnement de pipeline ainsi de circuler les flux donnés dans l'enchaînement, pour la source des données c'est les couches de OpenStreetMap sous format de GeoJson.

1. Description des données de l'étude de cas

Les données utilisées appartiennent à la zone géographique de Clermont Ferrand en format de Geojson, qui sont stockés dans Postgis, publié en geoserver, ils sont des entrées des processus construits dans le modeleur de QGIS. Ces couches correspondent à des tables dans Postgis. Ces données sont fournies par Markie une chercheuse qui travaille dans l'équipe ELIPSE de même projet ACTIVmap, ces données sont à la base des entres des processus de QGIS pendant la majorité de la période de stage on essaie d'exécuter les processus exportés

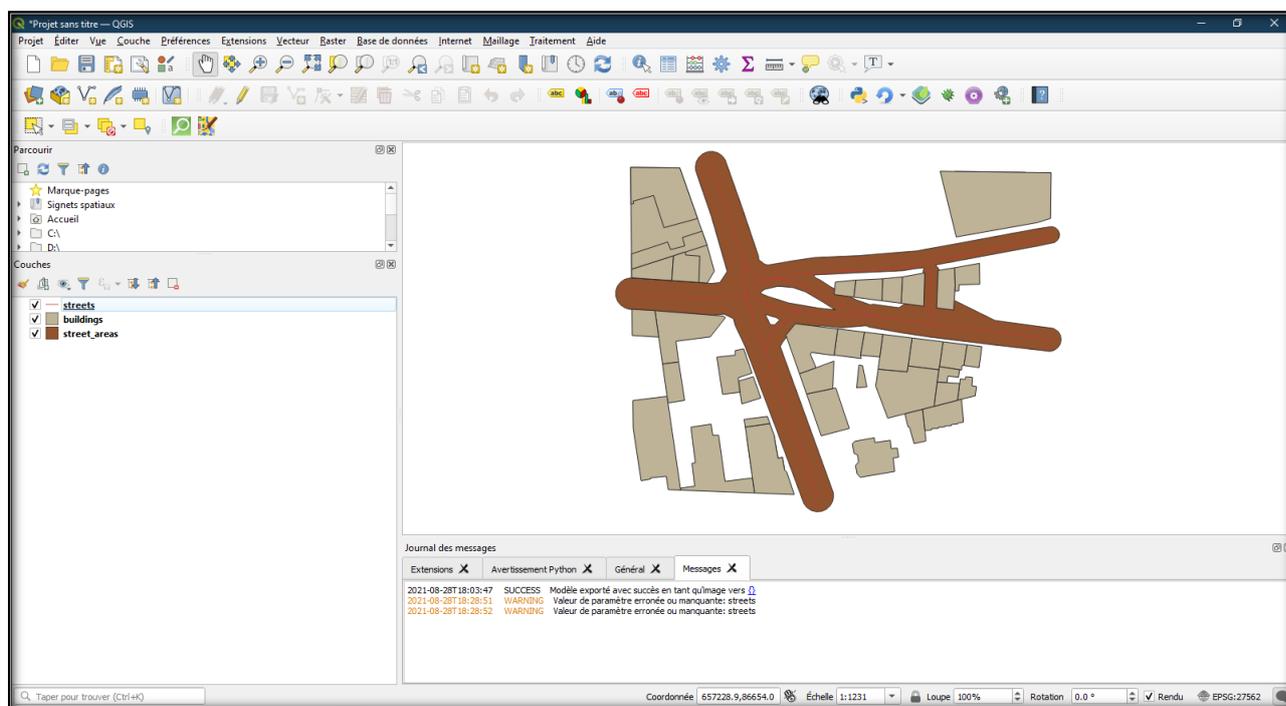


Figure 13: Les couches entrées pour exécuter les processus

via ces données, après les changements des besoins on a les utilisés comme d'input des processus de la nouvelle architecture pour les données lustrées dans la figure de l'interface de QGIS on trouve les couches suivantes :

- ✓ Streets une couche correspond aux boulevards et aux carrefours de la zone étude ce format de d'une line centrée de la zone boulevard ou tag steert de openstreetmap.
- ✓ La couche Buildings correspond aux blocs de bâtis (les battements, les maisons, etc).
- ✓ Street_areas une couche qui décrit un regroupement de l'espace occupé par la sole des carrefours.

1.1 Administration de Geoserver

L'administration des couches et les droits d'accès seront mis en place par l'administrateur de serveur Geoserver qu'est reliées avec PostGis. On a créé un Workspace qui s'appelle ACTIVmap qui souvent regroupe des couches et les éléments similaires de même zone.

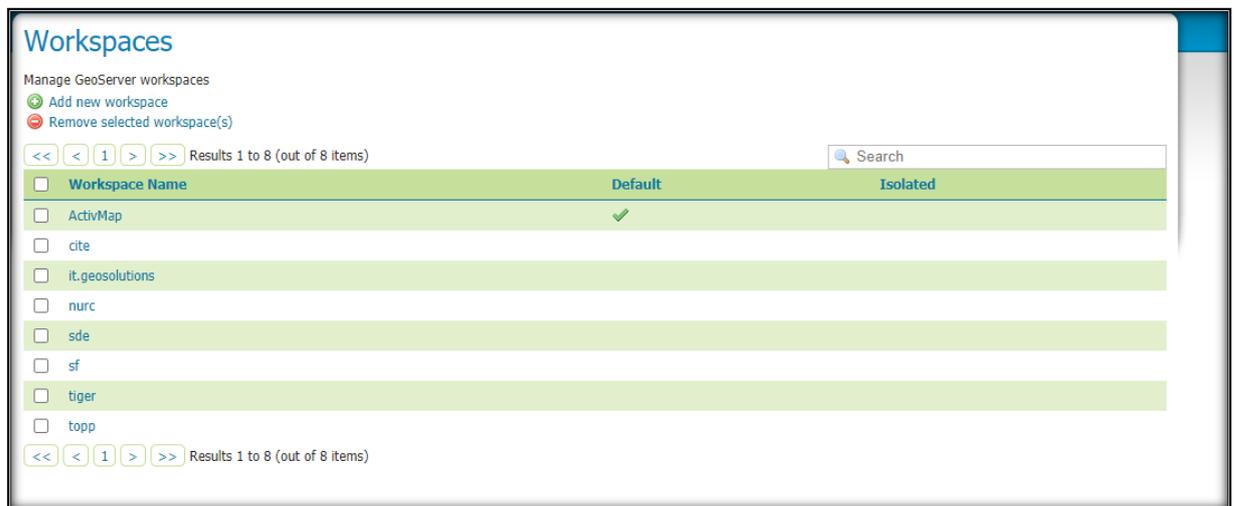


Figure 14: Workspace ACTIVmap

Aussi l'activation des protocoles WPS, WFS et WMS afin d'interroger avec des processus en envoyant des données sous format Raster ou vecteur.

L'interface de la configuration des couches publiées, et la visualisation des résultats.

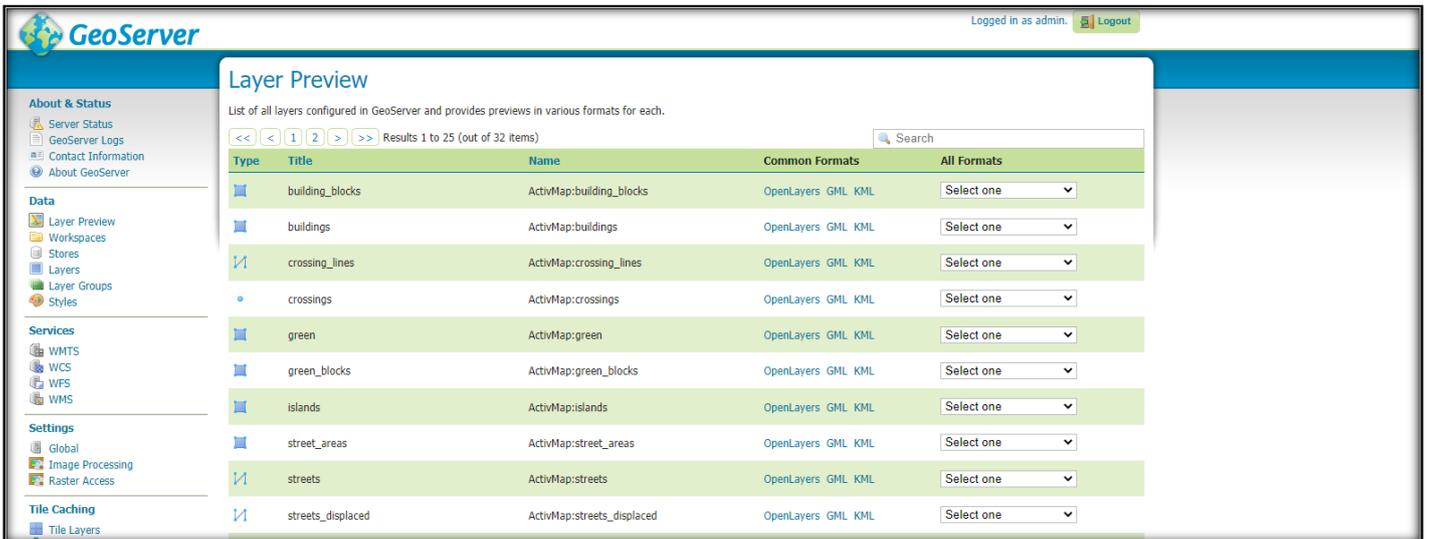


Figure 15: les couches publiées sous Geoserver

2. Application Web

2.1 Diagramme de classe

Ce diagramme de classe schématise la structure interne de la partie backend développée en Geodjango permet de fournir une représentation abstraite des objets du système qui vont

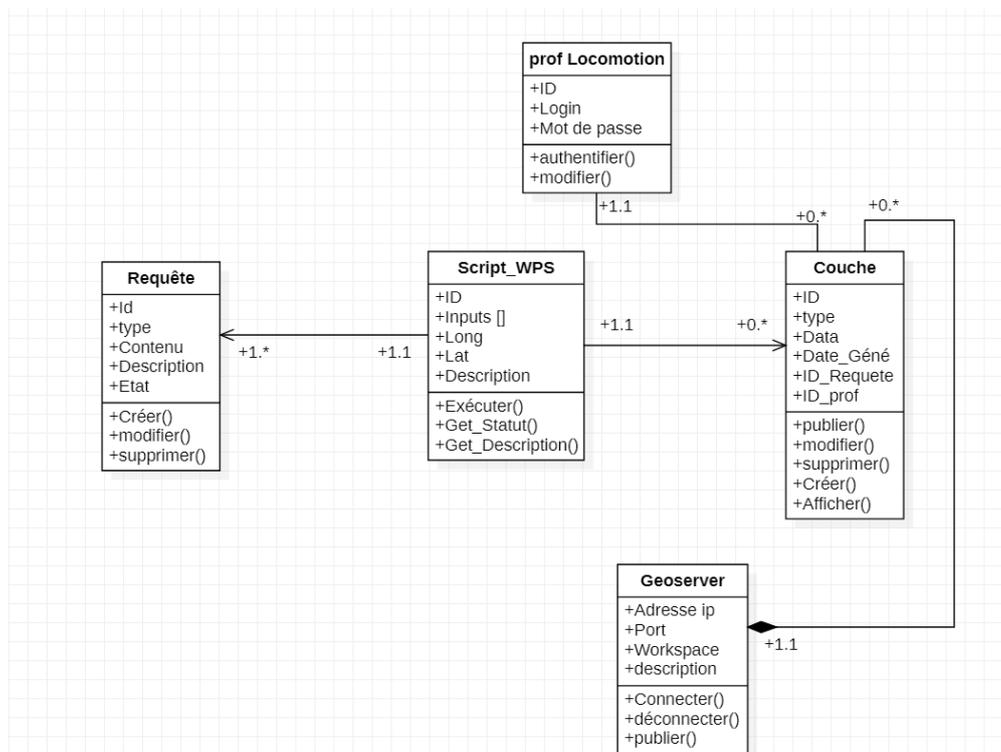


Tableau 3: Diagramme de classe

interagir avec les différentes parties. Il comporte les différents objets (classes) modélisant le système, et dont on aura besoin lors de développement.

- Professeur de locomotion peut générer plusieurs couches et les droits d'accès ainsi de faire la correspondance avec les autres parties de l'architecture.
- Les couches seront publiées dans le serveur de Geoserver puis stocker dans la base des données PostGis..
- Les requêtes il va être construites préalable pour distingues les inputs et outputs et acheminer les résultats.
-

2.2 Design Pattern Classification utilisé dans l'application web (DDP)

Notre application GeoDjango de base en quatre composants essentiels.

- ✓ **Modèles** (models.py): ensemble des classes de communiquer avec la base de données et préserver les données de manière POO.
- ✓ **Formulaires** (forms.py) : Pour collecter les données des utilisateurs.
- ✓ **Vues** (views.py) : Contrôleur central pour gérer la demande de l'utilisateur et communiquer avec les objets du modèle.
- ✓ **Modèles** (dossier de modèles) : affichez la réponse aux utilisateurs dans une structure généralisée avec des informations personnalisées.

On a utilisé ce design pattern pour construire notre application car il facilite l'implémentation des composants de l'architecture distribuée.³⁶

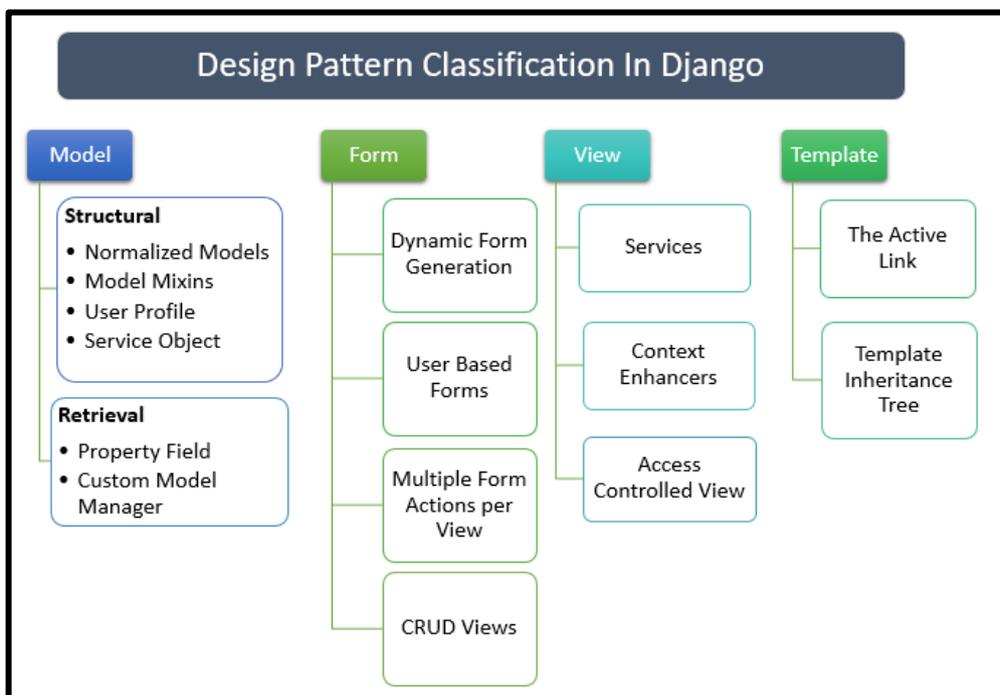


Figure 16: Design Pattern Classification

2.3 Espace administrateur

Cette interface permet d'accéder à l'espace administrateur via un email et mot passe.

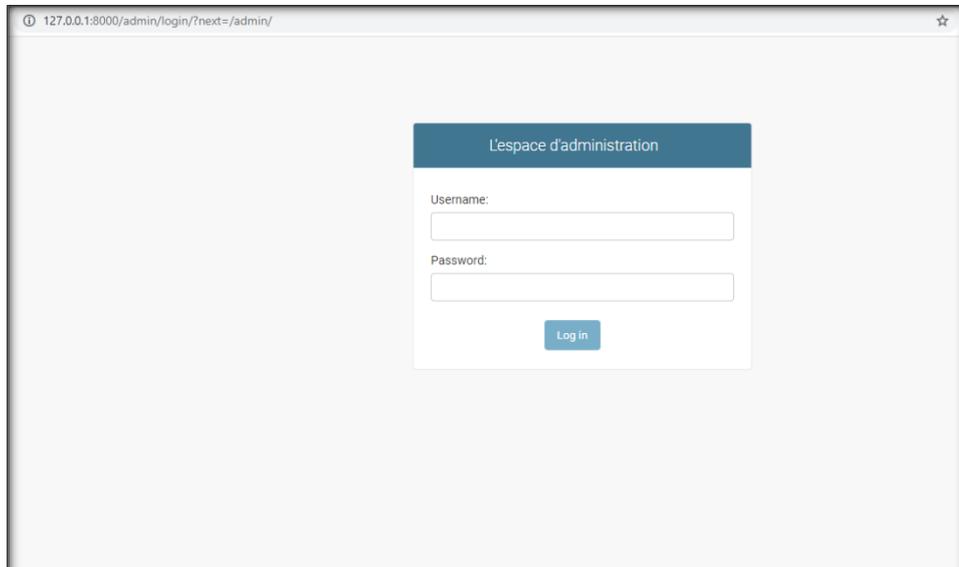


Figure 17: L'authentification

Dans l'espace administrateur donne la permission de modification, l'ajout et la suppression des utilisateurs, et des informations de géolocalisation des zones traitées.

Ce formulaire donne la possibilité d'ajouter un utilisateur de l'application.

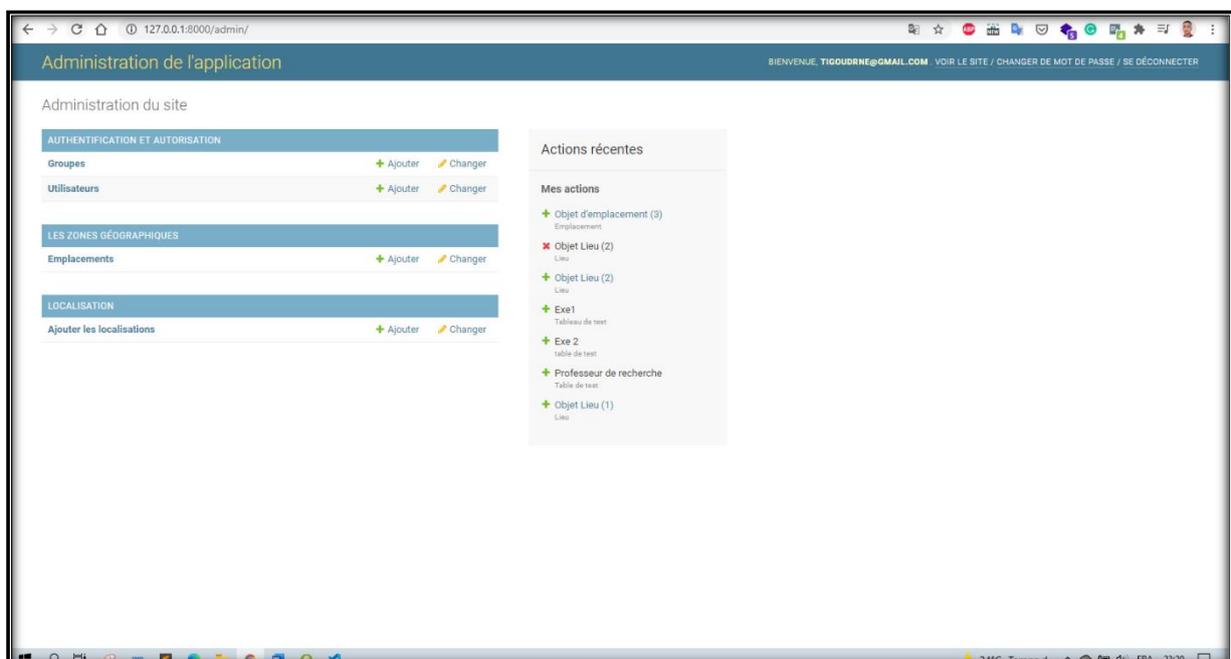


Figure 18: l'espace administration

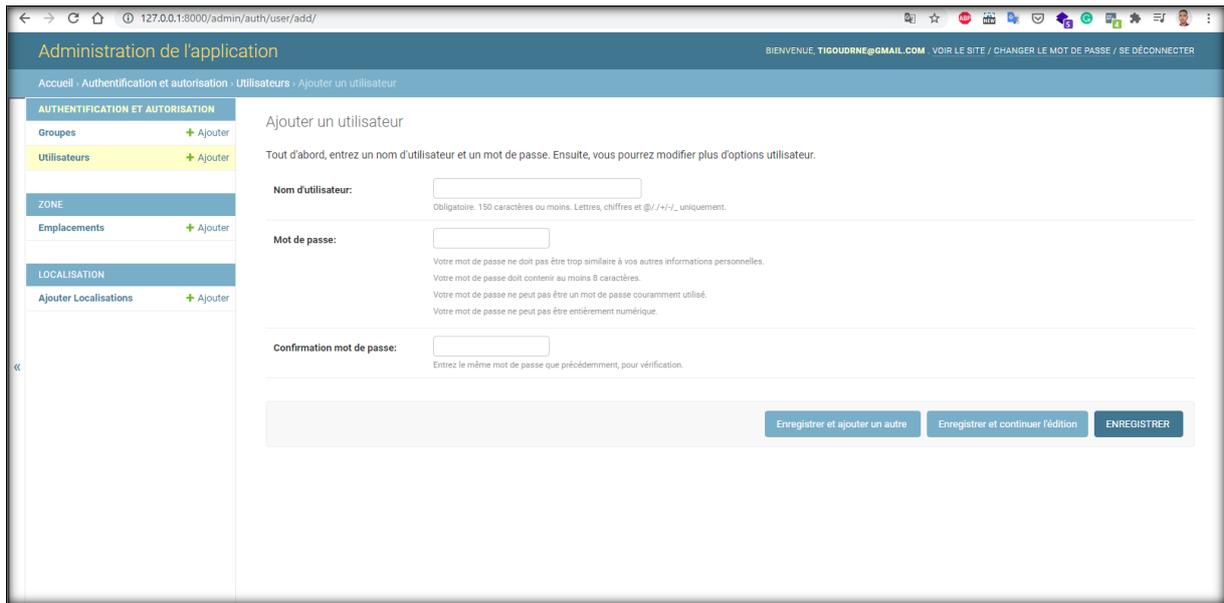


Figure 19: L'ajout d'un utilisateur

La gestion des utilisateurs et l'affectation des permissions et des rôles à des groupes utilisateurs.

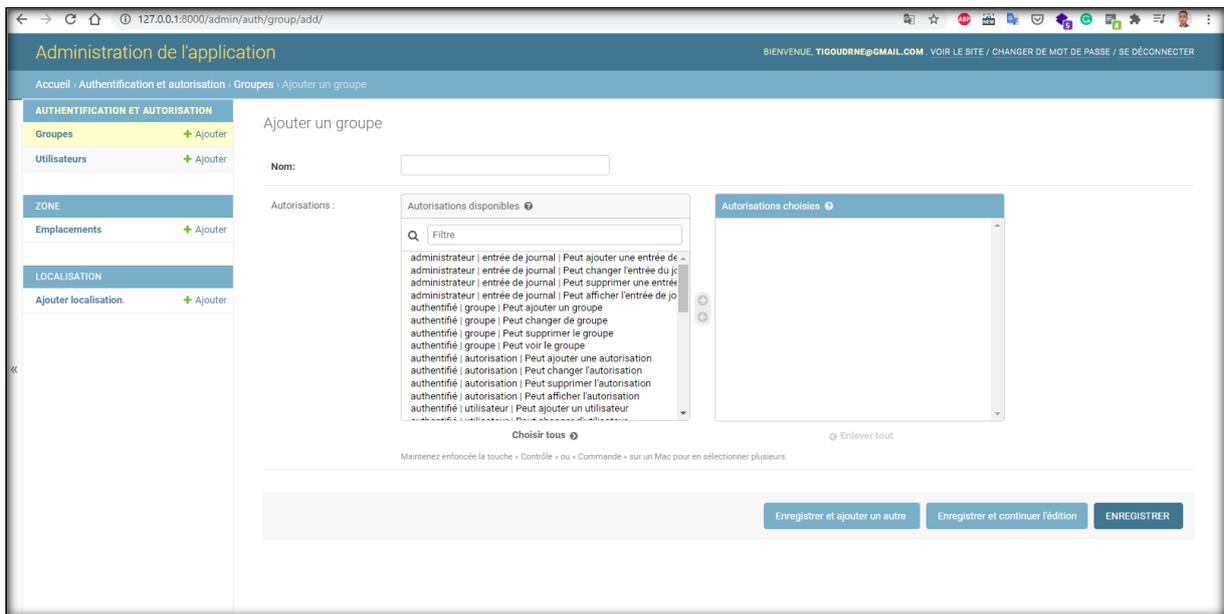


Figure 20: Gestion des permissions des groupes des utilisateurs

Cette interface nous montre la récupération des couches depuis Geoserver et l'affichage dans Openlayers dans la carte de OpenStreetMap.

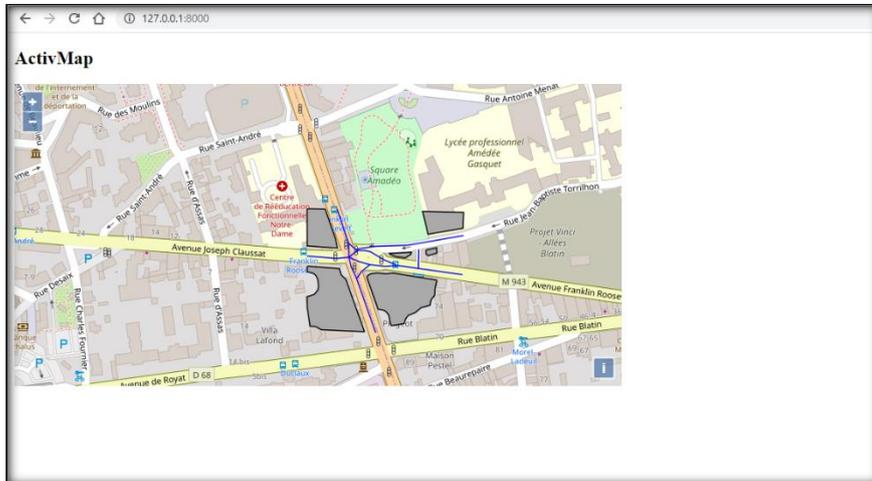


Figure 21: L'affichage des couches dans une carte.

2.4 Processus de traitement des données

La première interaction avec les processus de traitement de flux des données géographique viens de modeler graphique de QGIS qui permet de créer des modèles complexes en utilisant une interface simple et facile. "Dans un SIG, la plupart des opérations d'analyses ne sont pas simples mais font parties d'une chaîne d'opérations. En utilisant le modeler graphique, cette chaîne de traitements peut être regroupée dans une tâche, qui est plus simple à exécuter et peut être réutilisée sur d'autres jeux de données. Peu importe le nombre d'étapes et d'algorithmes impliqués, un modèle est exécuté comme un seul algorithme, permettant ainsi d'économiser temps et effort, notamment sur de plus grands modèles. Aussi la plupart des algorithmes utilisées existe dans python mais la manière la plus rapide de créer un script Python est de créer un modèle puis de l'exporter en Python. Puis de l'utiliser dans un serveur PyWPS."³⁷

Après l'hébergement de script python dans le serveur PyWPS il est indispensable de construire une requête xml qui va interroger le service web, cette requête il va être passer en Post.

³⁷ https://docs.qgis.org/2.14/fr/docs/user_manual/processing/modeler.html

On a utilisé pour exécuter et envoyer la requête via Https le logiciel Postman.

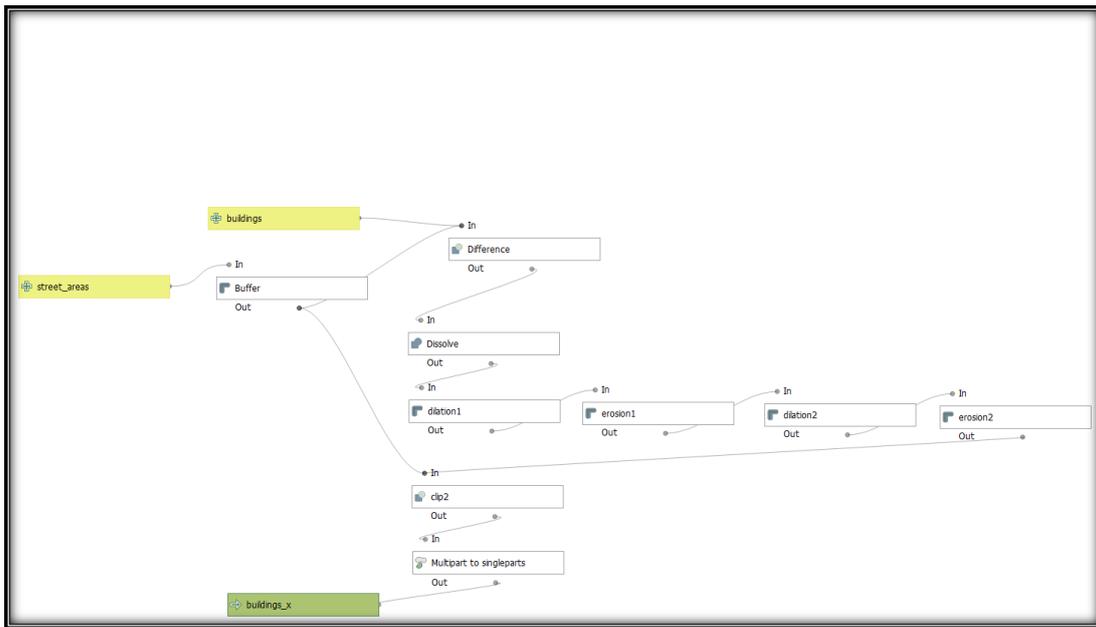


Figure 22: Exemple d'un Algorithme construit en modeleur graphique de QGIS

3. Difficultés et défis de déploiement de cette solution

Cette partie correspond aux inconvénients majeurs de cette architecture ainsi que certaines règles à respecter.

3.1 Au niveau de serveur PyWPS

C'est vrai que l'exportation des processus depuis QGIS en script python facilite énormément des choses mais PyWPS supporte une structure rigide qui doit être respectée comme la fonction « `_handler` » et la fonction « `__init` » pour initier certains paramètres, aussi l'unicité de type des données (les versions de gml, les données vectorielles etc.), également le contenu des données comme l'échelle et la projection et les normes utilisées. Aussi la version de serveur doit être déployée sous linux car ce dernier qui supporte le multi-Processing. Ainsi que l'ordonnancement d'exécution et le nombre des requêtes envoyé. Et parfois le serveur ne supporte pas un traitement lourd avec une complexité élevé.

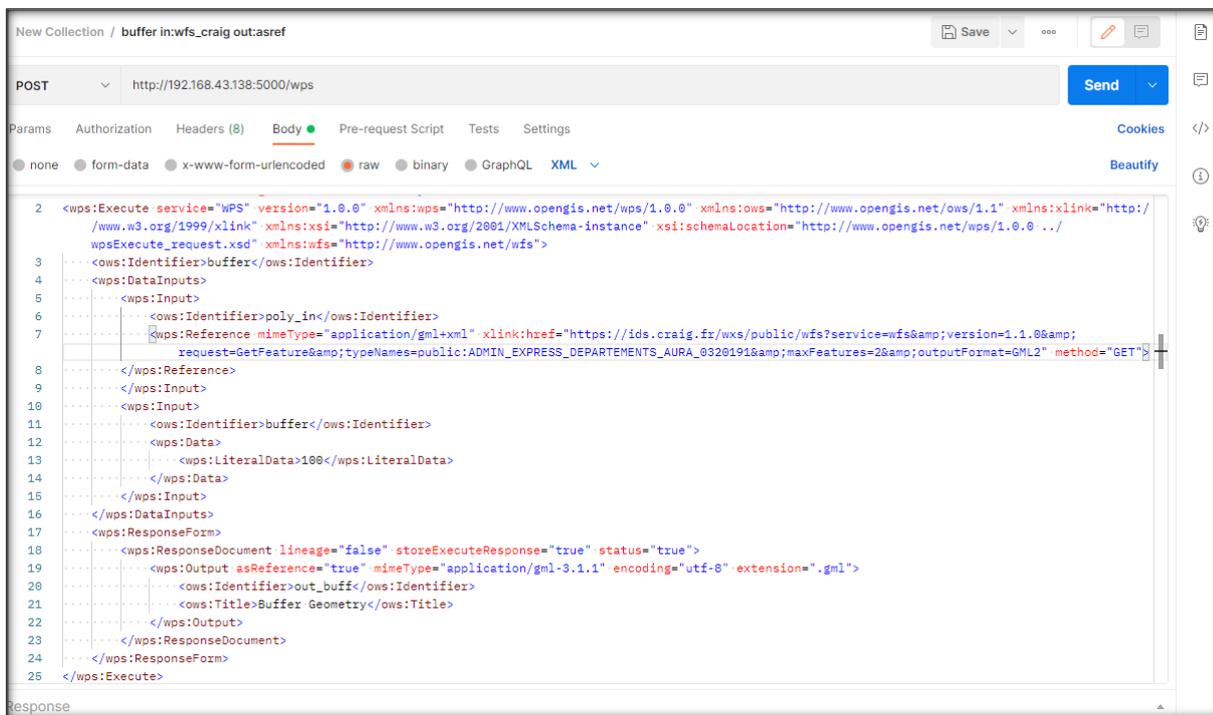


Figure 23: Exemple de requête d'exécution de processus Buffer

3.2 Contraintes de la requête xml

La requête XML qui va être envoyée depuis l'application web peut comporter des requêtes imbriquées (requête WFS pour récupérer les entrées ou des données directement comme celui de gml2). Également les communautés qui supportent l'utilisation des requêtes WPS dans le domaine de la géométrie sont très nombreux, la rareté de la documentation d'une part, et le serveur qui ne supporte pas des fonctionnalités assez développées d'autre part représentent des défis majeurs à surmonter. Aussi chaque changement de processus ou des données la requête doit être à redéfinir.

3.3 Contraintes du type des données

Durant l'implémentation il faut faire attention aux types des données circulés (GML, KML, Raster etc.) et aux versions comme GML on trouve versions 1,2 et 3, et sont-ils supportés ou non ? Aussi la projection et l'échelle doivent être respectées afin de bien visualiser les résultats.

3.4 Contraintes fonctionnelles

Certaines fonctionnalités doivent être prises en considération dans le moyen terme comme la séparation des données traitées, et la substitution des couches pour éviter la redondance et la surcharge de la base des données et d'unifier les normes de géolocalisation.

Conclusion

Dans ce chapitre nous avons détaillé la partie de réalisation de l'étude de cas de projet en illustrant et détaillant les interfaces de la plateforme réalisée et les outils qu'on a utilisés pour montrer l'efficacité de la solution proposée.

Conclusion générale

Notre projet de fin d'études mené au sein de laboratoire d'informatique, de modélisation et d'optimisation des systèmes (LIMOS), a été pour nous non seulement une étape à franchir pour achever mon parcours académique, mais aussi une opportunité de prendre part à un projet innovant qui s'inscrit dans le domaine de géolocalisation et le traitement des données spatiales ainsi que de découvrir le milieu de la recherche scientifique.

Notre mission consistait à concevoir et à mettre en œuvre une solution capable de monter un enchaînement de traitement des données géographiques ainsi que d'exécuter des processus et d'construire aux chercheurs qui travaillent dans le projet ACTIVmap une architecture efficace qui leurs serviront d'avancer dans leurs travaux de recherche et opérationnelles.

Malgré les difficultés que nous avons rencontrées durant ce projet, qui résident essentiellement dans la nouveauté de domaine d'application et les outils avec lesquels nous avons travaillés, nous sommes finalement parvenus à nous familiariser avec ces nouvelles technologies, mais malheureusement la contrainte de temps, de ce stage, ne nous a pas donné la possibilité de finaliser le projet, et que certains de ces travaux restent en cours.

Ce projet a été pour nous l'occasion de consolider et de mettre en œuvre l'ensemble des connaissances que nous avons acquis à l'école nationale des sciences appliquées de Berrechid (ENSAB), d'être confrontés au monde professionnel, de développer des relations humaines avec le personnel et de découvrir les enjeux et les problématiques relatifs à la mise en place de projets.

Bibliographie & Webographie

<https://www.openstreetmap.org/#map=12/45.8113/3.1991>

https://docs.qgis.org/3.16/fr/docs/pyqgis_developer_cookbook/plugins/ide_debugging.html

<https://www.coursera.org/learn/introduction-gis-mapping>

<https://docs.geoserver.org/latest/en/user/>

<https://docs.qgis.org/3.16/en/docs/index.html>

<https://pywps.readthedocs.io/en/latest/>

<https://py-qgis-wps.readthedocs.io/en/latest/index.html>

<https://docs.djangoproject.com/fr/3.2/ref/contrib/gis/tutorial/>

<https://gis.stackexchange.com/>

<https://www.ogc.org/standards/wps>

<https://www.ogc.org/standards/geopackage>

<http://postgis.fr/chrome/site/docs/workshopfoss4g/doc/PostGISIntro.pdf>

<https://enterprise.arcgis.com/fr/server/10.3/create-webapps/linux/about-creating-web-gis-applications.htm>

<https://www.geoportail.gouv.fr/>

<https://geoportail.wallonie.be/home.html>

<https://docs.geonode.org/en/master/start/index.html>

<https://github.com/52North/wps-js-client>

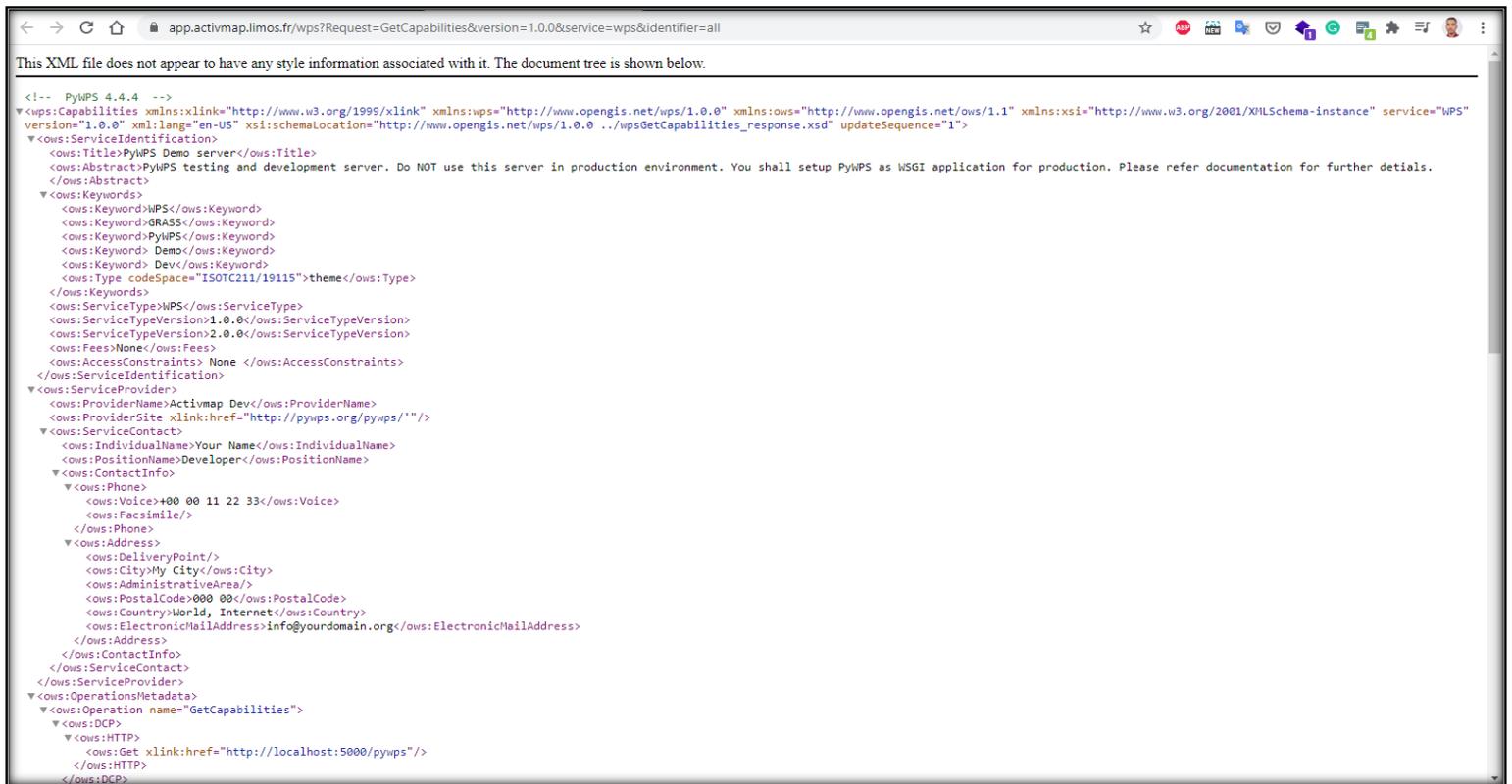
Annexe

Nous présentons les screens de scripts Buffer qu'on a utilisés durant l'étude de cas, ce script écrit en python et hébergé en serveur PyWPS, il comporte deux fonctions principales le constructeur d'initialisation de certains paramètres de script tels que le type des données supportées, le nom, la nature des données ainsi de suite, comporte aussi une autre fonction `_handler` dans laquelle en spécifiant le traitement appliqué au flux des données.

```
grassbuffer.py 3
C: > PFE > Tpywps > pywps-flask > processes > grassbuffer.py > ...
1
2 from pywps import Process, LiteralInput, ComplexInput, ComplexOutput, Format
3
4
5 from pywps.validator.mode import MODE
6
7
8
9
10 class GrassBuffer(Process):
11
12     def __init__(self):
13         inputs = [ComplexInput('poly_in', 'Input1',
14                               supported_formats=[Format('application/gml+xml')],
15                               mode=MODE.SIMPLE),
16                  LiteralInput('buffer', 'Buffer', data_type='float',
17                               allowed_values=(0, 1, 10, (10, 10, 100), (100, 100, 1000)))]
18         outputs = [ComplexOutput('buff_out', 'Buffered',
19                                 supported_formats=[
20                                     Format('application/gml+xml')
21                                 ])]
22
23         super(GrassBuffer, self).__init__(
24             self._handler,
25             identifier='grassbuffer',
26             version='0.1',
27             title="GRASS v.buffer",
28             abstract="The process uses the GRASS GIS \
29             v.buffer module to generate buffers around inputs ",
30             profile='',
31             inputs=inputs,
32             outputs=outputs,
33             store_supported=True,
34             status_supported=True,
35             # grass_location="/tmp/outputs/pywps_process_GMkyxP/pywps_location"
36             grass_location="epsg:3857"
37         )
38
```

```
37
38
39     def _handler(self, request, response):
40
41         from grass.pygrass.modules import Module
42         Module('v.import',
43               input=request.inputs['poly_in'][0].file,
44               epsg=3857, output='poly', extent='input')
45         Module('v.buffer',
46               input='poly',
47               distance=request.inputs['buffer'][0].data,
48               output='buffer')
49         Module('v.out.ogr', input='buffer', output='buffer.gml', format='GML')
50
51         response.outputs['buff_out'].file = 'buffer.gml'
52
53         return response
54
```

Un screen d'une requête Wps pour récupérer les protocoles héberger dans le serveur ainsi que les informations autour de la façon d'exécution de chaque scripte.



This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<!-- PyWPS 4.4.4 -->
<wps:Capabilities xmlns:xlink="http://www.w3.org/1999/xlink" xmlns:wps="http://www.opengis.net/wps/1.0.0" xmlns:ows="http://www.opengis.net/ows/1.1" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" service="WPS"
version="1.0.0" xml:lang="en-US" xsi:schemaLocation="http://www.opengis.net/wps/1.0.0 ../wpsGetCapabilities_response.xsd" updateSequence="1">
  <ows:ServiceIdentification>
    <ows:Title>PyWPS Demo server</ows:Title>
    <ows:Abstract>PyWPS testing and development server. Do NOT use this server in production environment. You shall setup PyWPS as WSGI application for production. Please refer documentation for further details.
    </ows:Abstract>
    <ows:Keywords>
      <ows:Keyword>WPS</ows:Keyword>
      <ows:Keyword>GRASS</ows:Keyword>
      <ows:Keyword>PyWPS</ows:Keyword>
      <ows:Keyword> Demo</ows:Keyword>
      <ows:Keyword> Dev</ows:Keyword>
      <ows:Type codeSpace="ISOTC211/19115">theme</ows:Type>
    </ows:Keywords>
    <ows:ServiceType>WPS</ows:ServiceType>
    <ows:ServiceTypeVersion>1.0.0</ows:ServiceTypeVersion>
    <ows:ServiceTypeVersion>2.0.0</ows:ServiceTypeVersion>
    <ows:Fees>None</ows:Fees>
    <ows:AccessConstraints> None </ows:AccessConstraints>
  </ows:ServiceIdentification>
  <ows:ServiceProvider>
    <ows:ProviderName>Activmap Dev</ows:ProviderName>
    <ows:ProviderSite xlink:href="http://pywps.org/pywps/" />
  </ows:ServiceProvider>
  <ows:ServiceContact>
    <ows:IndividualName>Your Name</ows:IndividualName>
    <ows:PositionName>Developer</ows:PositionName>
  </ows:ServiceContact>
  <ows:ContactInfo>
    <ows:Phone>
      <ows:Voice>+00 00 11 22 33</ows:Voice>
      <ows:Facsimile>
      </ows:Phone>
    </ows:Phone>
    <ows:Address>
      <ows:DeliveryPoint>
      <ows:City>My City</ows:City>
      <ows:AdministrativeArea>
      <ows:PostalCode>000 00</ows:PostalCode>
      <ows:Country>World, Internet</ows:Country>
      <ows:ElectronicMailAddress>info@yourdomain.org</ows:ElectronicMailAddress>
    </ows:Address>
    </ows:ContactInfo>
  </ows:ContactInfo>
  </ows:ServiceContact>
  </ows:ServiceProvider>
  <ows:OperationsMetadata>
    <ows:Operation name="GetCapabilities">
      <ows:DCP>
        <ows:HTTP>
          <ows:Get xlink:href="http://localhost:5000/pywps/" />
          </ows:HTTP>
        </ows:DCP>
      </ows:DCP>
    </ows:Operation>
  </ows:OperationsMetadata>
</wps:Capabilities>
```