

Web Mapping : Carnet de laboratoire

2021/2022

Auteurs :

REVENEAU Hugo



Contents

I. Séances dédiées.....	3
Jeu. 01/12/2021 matin.....	3
Lun 06/12/2021 après-midi.....	3
Mar 07/12/2021 matin	4
Mar 07/12/2021 après-midi	4
Lun 13/12/2021 matin.....	5
Lun 13/12/2021 après-midi.....	5
Mar 14/12/2021 matin	5
Mar 14/12/2021 après-midi	5
Ven 17/12/2021 matin	6
II. Avancé du projet sur le temps libre.....	6

I. Séances dédiées

Jeu. 01/12/2021 matin

La première étape est la découverte du sujet, et l'extraction des contraintes majeures : Utilisation du WMTS du Géoportail afin de générer une série d'imagettes carrées qui serviront à la réalisation d'un jeu de taquin. Il est nécessaire de n'en extraire que 15, mais il est possible que la 16^{ème} puisse servir pour les projets connexes. Elle sera donc extraite également.

Cette manipulation est à réaliser au sein d'une interface web complète : nécessité de gérer le Front-end et le Back-end.

Back-end : Il faut stocker les différents jeux d'imagettes dans une base de données. Cette dernière n'a pas besoin d'être géographique car la position dans l'espace des tuiles n'est pas utile à la finalité du projet. **La seule information utile est leur position relative pour détecter la victoire.** Nous utiliserons alors une base de données PostGreSQL standard. Le second volet du Back-end est l'utilisation du WMTS du Géoportail nécessaire à l'import des imagettes. Ce dernier sera requêté directement via les URL des images, comme vu précédemment en cours. **Il faudra alors explorer leur syntaxe pour parvenir à générer un carré de 4x4 images correctes.**

Front-end : Ce projet nécessite une page web permettant la navigation sur une carte du monde, découpée en 4x4 images, permettant alors de choisir le niveau de zoom et l'emprise du jeu d'imagettes à créer. Il faudra afficher une carte, son découpage, implémenter des outils de navigation (Nord, Sud, Est, Ouest, Zoom, Dézoom) et un formulaire de saisie afin d'intégrer un jeu dans la future base de données.

La seconde partie de la matinée est dédiée à une réunion avec mon encadrant, M. Fritsch. Nous avons alors établi un cahier des charges sommaire :

- La carte doit provenir du WMTS Géoportail.
- Les imagettes doivent être séparées par un fin liserai blanc afin de mettre en évidence les séparations entre ces dernières.
- D'un point de vue mise en page, on doit retrouver le navigateur au centre et les outils de navigation sur les côtés. On doit retrouver le formulaire sous cet ensemble.

Lun 06/12/2021 après-midi

Cet après-midi est consacré à l'exploration du système de WMTS du Géoportail. Il m'a alors fallu explorer la page de documentation correspondante ([Cliquer ici](#)). On se base sur l'ortho-image mondiale du Géoportail.

Le premier point abordé est la façon dont le tuilage est réalisé. Les tuiles sont localisées par deux entiers, en partant du coin supérieur gauche, croissant vers la droite et le bas. Le zoom part également de 0 (une seule image), jusqu'à un nombre variable dépendant de la zone observée. A chaque incrémentation du zoom, les tuiles sont divisées en 4 sous-tuiles. Ainsi ce n'est qu'à partir du zoom 2 qu'on peut obtenir 4x4 tuiles. Il conviendra alors de bloquer le zoom à cette valeur.

Après plusieurs essais de rédaction d'URL, voici une URL type du flux WMTS qui nous intéressera :

URL :

<https://wxs.ign.fr/pratique/geoportail/wmts?SERVICE=WMTS&REQUEST=GetTile&VERSION=1.0.0&LAYER=ORTHOIMAGERY.ORTHOPHOTOS&TILEMATRIXSET=PM&TILEMATRIX=6&TILECOL=35&TILEROW=20&STYLE=normal&FORMAT=image/jpeg>

Résultat :



On note qu'on utilise la clé ***pratique*** qui nous permet notamment d'accéder à l'ortho-image. L'URL est constituée d'une suite de paramètres. Le zoom est le ***TILEMATRIXSET***, la position est référencée dans ***TILECOL*** et ***TILEROW***, le type de fond de carte dans ***LAYER***.

Mar 07/12/2021 matin

Il est nécessaire de nous coordonner avec les membres des groupes connexes, qui sont pour ma part Ziad Boukbir et Ilona Baran, notamment sur la façon dont nous allons échanger les imagerie. Afin de leur permettre d'avancer, j'ai généré un jeu d'imagerie primitive, contenant simplement une image de numéros allant de 1 à 16, aux dimension et format des tuiles du WMTS. Le début du projet utilisera donc des images en local afin d'implémenter le Front-end. La base de données sera implémentée par la suite et il « suffira » de changer la méthode d'import.

La suite de la matinée est dédiée à la configuration du Wamp serveur, nécessaire au codage de la page web.

Mar 07/12/2021 après-midi

Cette demi-journée est consacrée au codage du Front-end, notamment la mise en place des différents grid et des différentes div : un en-tête, une zone contenant les outils de navigation et la carte tuilée, une zone pour le formulaire et un pied-de-page.

Cette étape est assez chronophage car elle nécessite d'essayer et de corriger en permanence, afin de trouver un compromis entre le pratique et le visuel.

Lun 13/12/2021 matin

Possédant alors un Front-end pouvant accueillir les images, cette demi-journée est consacrée à la rédaction de requêtes php permettant d'importer dans le navigateur les images stockées en local. Après avoir passé du temps à réfléchir à de telles requêtes, il a semblé évident que je m'égarais et que je perdais inutilement du temps : ces requêtes doivent interroger une base de données et non mes fichiers locaux.

Lun 13/12/2021 après-midi

Il est cette fois-ci temps de commencer l'implémentation du Back-end. Pour cela, une base PostgreSQL est implémentée sous pgadmin4. Elle est constituée d'une unique table contenant un id serial et un nom pour le jeu de données. Un champ contenant un chemin vers les fichiers locaux est dans un premier temps créé, le temps de réfléchir à la meilleure manière de stocker les photos. Cette étape a pris du temps à aboutir car elle nécessitait l'installation sur ma machine personnelle de la bonne version de PostgreSQL, ainsi que la bonne configuration du serveur local Wamp permettant l'échange entre la base de données et la page web créée précédemment.

Mar 14/12/2021 matin

Après réflexion, la base de données est modifiée afin de contenir 16 champs au lieu du chemin précédemment défini, stockant chacun une URL permettant d'accéder à l'image du flux. On stocke ainsi les images par l'intermédiaire du format texte, limitant ainsi l'espace nécessaire. Cela rend possible un éventuel hébergement en ligne.

Ceci fait, il convient alors d'avancer le code JavaScript qui gère l'appel des images. Le choix d'un code orienté objet est destiné à rendre le code plus robuste. La classe **Visualisateur** est alors implémentée. Elle contiendra toutes les méthodes de navigation dans la carte elle-même, mais aussi d'envoi des imagerie dans la base de données.

Mar 14/12/2021 après-midi

Poursuite du code. Le code est alors fonctionnel : il permet de naviguer entre les imagerie, de zoomer et dézoomer, il gère correctement les translations du point de vue du changement des URL, et il permet de stocker ces dernières. Cependant il convient de corriger toute une gamme d'erreurs qui apparaissent logiquement au fur et à mesure du code :

- Erreur lors d'un zoom trop grand, demandant au flux une tuile qui n'existe pas.
- Erreur lors d'un dézoom trop grand, qui demande au flux 16 tuiles là où il n'en possède que 1 ou 4.
- Erreur lorsque l'utilisateur demande un dézoom alors qu'il se trouve sur une bordure : on demande à nouveau une tuile de bord dépassant l'emprise normal de la grille.
- Problème d'insertion dans la base de données puisqu'aucun mécanisme n'interdit d'entrer deux jeux de données avec des noms similaires. Cela ne pose pas de problème maintenant dans l'absolu, mais étant donné que le sujet d'Ikona Baran nécessite de piocher 5 jeux au hasard, il peut être perturbant pour l'utilisateur de voir apparaître plusieurs jeux possédant le même nom.

Ven 17/12/2021 matin

Compte tenu des observations faites le 14/12 et du temps de réflexion qui a été possible entre cette séance et la précédente, les erreurs liées au zoom ont été corrigées.

- En interceptant les erreurs de type 404 renvoyées lors d'un zoom trop fort, il a suffi de générer un dézoom dans ce cas.
- La détection des bords se fait simplement par un calcul de l'intervalle de valeurs de la grille considérée. On peut ainsi, en décalant la vue lors du dézoom, éviter toute sortie de l'emprise, au détriment d'un recul droit de la vue.
- Pour le zoom minimal, il a suffi de limiter le dézoom au zoom 2 pour les raisons expliquées dans partie du 06/12/2021 après-midi.

Cependant, la détection des bords ayant pris beaucoup de temps, l'erreur de nom n'a pas pu être traitée mais le sera par la suite.

II. Avancé du projet sur le temps libre

Cette partie du projet est plus difficile à faire suivre au jour le jour car il progresse petit à petit, parfois sur des temps très courts où parfois sur des soirées entières. Pour pallier cela, le suivi de cette seconde partie se fera sous la forme d'un texte.

Dans un premier temps, la dernière erreur a été corrigée en vérifiant l'intégralité des noms de la base de données, et en bloquant le bouton d'envoi du jeu dans la base en cas de similarité d'un nom avec celui saisi. Cela ne prend cependant pas encore en compte les accents et espaces que l'on préférera éviter.

En parallèle, j'ai initié le transfert de la base de données et du site sur Alwaysdata, afin d'annuler toute démarche d'installation longue, fastidieuse et source d'erreur pour l'utilisateur. Ayant déjà de l'expérience sur Alwaysdata, cette étape n'a nécessité de documentation que pour la création de la base PostgreSQL directement en ligne. Les détails du compte sont :

- Mail : hugo.reveneau@gmail.com
- Mot de passe : disponible sur demande
- Nom d'utilisateur : hugoreveneau2
- url du site : hugoreveneau2.alwaysdata.net/creation

Ce compte pourra également accueillir les pages des sujets connexes dans un premier temps. Dans le cas d'une poursuite du projet, la création d'un compte Alwaysdata dédié au projet sera nécessaire afin de libérer mon adresse mail pour des projets futurs.

Une fois toutes les difficultés liées à la connexion à une nouvelle base de données, cette fois-ci hébergée en ligne, le problème des accents a alors été contré par la normalisation des noms de jeux. Au lieu de contraindre l'utilisateur à écrire d'une manière normée, ce qui nécessite une explication et des rappels en cas d'erreur, il a été choisi de laisser l'input libre, et de le filtrer avant de vérifier son existence dans la base de données ou de l'intégrer dans cette dernière. Ainsi les noms sont normalisés (sans accents, sans majuscules, avec des _ au lieu des espaces) sans que l'utilisateur n'ait à s'en préoccuper, ce qui rend l'utilisation plus fluide et robuste.

Le fond du code étant terminé, un week-end a été dédié à l'élaboration du style CSS du client, aux animations des boutons, à leur positionnement, mais également à la responsivness, ce qui le rend utilisable quelle que soit la plateforme (mobile, PC, laptop, tablette).

Avant la journée porte ouverte j'ai rencontré mon commanditaire (M. Fritsch) afin de lui soumettre ma proposition et d'obtenir des conseils d'amélioration sur mon travail. Un compte rendu est joint à ce corpus de documentation. Nous avons convenu d'une évolution non présente dans le sujet initial mais toutefois intéressante, à savoir la possibilité de choisir entre plusieurs cartes plutôt qu'uniquement l'ortho-image.

Le code JavaScript ayant été orienté objet, il a été très simple de le modifier pour répondre à ce nouvel objectif. Il a suffi d'ajouter un bouton radio à droite de la carte, permettant de choisir entre le plan et l'ortho-image, et d'attribuer à ces radios la modification du paramètre **LAYER** et **FORMAT** en fonction de la couche souhaitée. En ajustant le style des boutons pour qu'ils correspondent à la charte graphique, le résultat est alors satisfaisant. Cette version a été présentée lors des Journées Porte Ouvertes.

La dernière étape est la livraison du résultat, avec toute la documentation nécessaire. Il convient alors de nettoyer le code de tout commentaire superflu et de zones de codes commentées, mais également de rédiger les documentations expert et utilisateur.