

Présentation d’Idée d’Application avec Shiny

GARREL Julian, GADJEU Kevine, NDUMCHOP Glory, BLANCHET Tibault

13 février 2026

Contents

Introduction	1
Mardi 10 février matin	1
Mardi 10 février Après-midi	2
Mercredi 11 février	2
Jeudi 12 février	3
Liens	3
Bibliographie	3

Introduction

Nous avons eut plusieurs idées d’application utilisant le module Shiny, nos trois idées se base sur une ou plusieurs cartes. Notre première idée est la création d’une carte à l’échelle française ou régional des différentes AOP ou label avec la possibilité d’obtenir plus amples informations en cliquant dessus. Notre seconde idée est la représentation sur une carte de la france, d’une région ou d’un département d’information issu de l’insee avec ou sans traitement d’analyse. Notre troisième idée est la création d’une carte des évènements d’une ville ou d’un département avec une barre de recherche par date ou type d’évènement ou lieux et une gestion de la temporalité.

Mardi 10 février matin

Nous avons choisi de nous pencher sur notre première idée qui est la présentation d’une carte interactive des différentes AOPs françaises. Pour ce faire nous avons choisi d’utilisé un jeu de donnée .shp comme base de carte afin de n’avoir que les départements et comunnes françaises et non l’entièrete du monde avec openstreet map. Ce jeu de donnée issu de notre travail de 3ème année sous QGIS, nous a fait rencontrer quelques problèmes: Les fichiers étant trop lourds pour gitHub, tibault s’est retrouvé avec un commit qui ne pouvait être push sur le répertoire commun ce qui entraîne un blocage total de la fonction push depuis Rstudio et la suppression de ces documents demandés eux aussi un commit et un push. Pour régler ce problème, il a esseyé d’utiliser l’interface Git Gui pour télécharger le commit, cependant celà recharger les documents précédement supprimer dans le dossier. Cette situation avait créer un embranchement dans

le workflow de travail qui s'était rejoint sur le push suivant de Julian. Tibault a donc recherché le noeud d'origine de l'embranchement afin de faire revenir sa version à cette base saine avant de pull les modifications de Julian. Pour ce faire, Tibault a effectué des recherches en particulier sur le site de (Bréhin, 2025) mais n'ayant pas pu résoudre mon problème en étant certain de la manipulation, Tibault a utilisé l'IA Claude afin d'obtenir plus ample explication et utiliser la commande `git reset --hard HEAD~1` pour revenir à la base saine.

Julian a quand a lui travailler sur les données des AOP, il a trouvé ces données sur le site (Data Gouv, 2025), le fichier étant très volumineux et dense en information, nous avons fait le choix d'utiliser l'IA pour la partie suivante. Nous avons rajouter un filtre pour catégoriser les AOP en 4 catégories : `Produit_carné`, `Produits_laitiers`, `Boissons`, `Autres` ce qui permettra à l'utilisateur final de trier les AOP selon ces catégories. Julian a ensuite utiliser les données .shp issus de notre travail d'e 3 ème année'admin express pour s'en servir de base de carte, cependant elle était trop lourde et comportaient des informations non utiles pour notre projet. Nous avons donc épurer nos données et pour lier notre carte avec nos AOP, nous avons ajouté le code INSEE dans nos deux fichiers en vue d'une futur jointure.

Mardi 10 février Après-midi

Nous nous sommes séparé les tâches : Glory sur l'UI pour la page d'accueil et l'onglet de la carte, Kevine sur l'UI de la page About, Julian sur le traitement des données et Tibault en support d'explication pour l'UI et la rédaction/synthétisation du traitement des données. Concernant le traitement des données, nous sommes partie de nos données .shp, nous avons dû nous les transformer géométriquement afin de les passer en WGS 84. Ensuite Julian a fait une jointure gauche entre notre catégorisation des AOPs et notre fichier `commune.shp`. Afin de faciliter la lecture de la carte et de ne pas la surcharger la carte avec les zones géographiques des AOPs, Julian a calculer les centroides de chaque AOPs afin de ne faire apparaître qu'un unique point par AOP. Suite à cela nous avons enregistré le fruit de nos calculs sur différents fichiers afin ne pas refaire faire les calculs à l'ouverture de l'application mais uniquement de la lecture des résultats.

Glory: sur la partie Ui, j'ai fait une barre déroulante des produits regroupés en 4 catégories: boissons, produits laitiers, produits carnés et autres. J'ai aussi fait une déroulante afin de permettre à l'utilisateur de trier par région. J'ai également fait des recherches afin apporter plus de couleurs dans l'interface, mais je n'ai pas encore obtenu des résultats qui marchent.

Mercredi 11 février

Julian avait créé une application en local pour faire les premiers essais sur la création et la gestion d'une carte interactive et de ses paramètres. Cette application test a servi de base pour l'onglet "Carte AOP" en reprenant le code test et en l'implémentant dans `ui.R` et `server.R`. Lors de cette transition, plusieurs problèmes sont apparus. Au démarrage, la carte restait vide car le `checkboxGroupInput` dans `ui.R` avait `choices = NULL` au lieu de charger directement les catégories comme dans la version test. Aucune catégorie n'apparaissait donc dans l'interface. Ce problème a été résolu en ajoutant une initialisation côté serveur avec `updateCheckboxGroupInput` pour charger dynamiquement les catégories et les pré-sélectionner toutes par défaut. En tentant de corriger ce premier problème, des `req(input$categories)` et des vérifications `is.null()` ont été ajoutés dans les `observe()` pour éviter les erreurs au démarrage. Ces ajouts, qui n'existaient pas dans la version test, ont créé un nouveau bug, l'impossibilité de désélectionner une AOP une fois sélectionnée, car ces vérifications bloquaient la réactivité des `observe()`. La solution a été de revenir à la logique simple de la version test en retirant tous ces `req()` et vérifications superflues. Une erreur "l'argument 'expr' est manquant" est apparue à cause d'une mauvaise syntaxe `observeEvent(once = TRUE, ...)` qui a été corrigée en utilisant `observe() %>% bindEvent(TRUE, once = TRUE)`. Cette journée peut se résumer à il ne faut pas sur-complicquer le code en ajoutant des sécurités inutiles qui créent plus de problèmes qu'elles n'en résolvent.

Jeudi 12 février

Julian s’est concentré sur la refonte complète de l’interface graphique de l’application pour lui donner un aspect plus professionnel et avec un style “premium”. La première étape a été la création d’une page d’accueil moderne avec quatre colonnes d’images en pleine hauteur représentant les catégories d’AOP (Boissons, Produits Laitiers, Produits Carnés, Autres), chacune accompagnée d’un bouton “Découvrir” qui renvoie vers la carte avec la catégorie en pré-sélection. Avant même de commencer le travail, un problème existait avec un bouton “Aller à la Carte” sur la page d’accueil était invisible et n’apparaissait qu’au survol de la souris. Ce problème a été résolu en ajoutant des styles CSS inline forçant les couleurs de fond et de texte du bouton. La deuxième étape a été l’intégration du logo AOP dans l’en-tête et la création d’une barre de titre premium avec un fond bordeaux profond (#2D1B1F) et du texte beige clair. Cette modification a cependant créé un nouveau problème, le fond bordeaux s’étendait sur toute la page, rendant illisibles les éléments de la page “Carte AOP” et de la page “About”. Un ensemble complet de règles CSS a dû être ajouté pour forcer le fond blanc sur le contenu des onglets, styliser les onglets pour qu’ils s’intègrent harmonieusement dans la barre bordeaux, et ajuster les couleurs de tous les composants (labels, dropdowns, checkboxes, textes) pour assurer leur lisibilité. Le problème le plus critique est apparu lors des tests fonctionnels où les boutons “Découvrir” pour Produits Laitiers et Produits Carnés de la page d’accueil ne filtraient pas correctement les données sur la carte. Une incohérence de nomenclature en était à l’origine. Les catégories dans les données étaient au singulier (“Produit laitier”, “Produit carné”) tandis que les boutons utilisaient le pluriel. Pour corriger ce problème à la source, le script `preparation_donnees_pour_app.R` a été modifié en ajoutant une transformation qui convertit automatiquement toutes les catégories au pluriel lors du chargement des données, garantissant ainsi une cohérence parfaite et évitant tout problème futur. Le résultat final est une application avec une identité visuelle cohérente, professionnelle et entièrement fonctionnelle.

Liens

ISARA : <https://isaranet.fr/> consulté le ...

Bibliographie

Liste des publications :

BRÉHIN, Maxime, 2025. *Supprimer un commit • Comprendre Git* [en ligne]. janvier 2025. [Consulté le 10 février 2026]. Disponible à l'adresse : <https://comprendre-git.com/fr/annuler-defaire-corriger/supprimer-un-commit/>

DATA GOUV, 2025. *data.gouv.fr* [en ligne]. octobre 2025. [Consulté le 10 février 2026]. Disponible à l'adresse : <https://www.data.gouv.fr/datasets/aires-geographiques-des-aoc-aop/>