

# Cahier des charges - Projet Banc de thons 3D

---

## 1) Contexte & objectifs

Contexte : Vous êtes développeur·euse front-end junior dans une agence spécialisée en datavisualisation interactive. Sujet : visualiser des données poissons récifaux (Nouvelle-Calédonie) via une scène 3D.

Objectifs : Créer une application Vue 3 + Three.js affichant un banc de thons animés, reliés à une API publique data.gouv.nc.

## 2) Périmètre

- Fonctionnel : chargement API, scène 3D, survol/clic, filtres, responsive.
- Hors périmètre : back-office, authentification, i18n.

## 3) Parties prenantes

- Client : valide livrable (pédagogie).
- Dev Front : conception, intégration, tests.
- Utilisateurs : étudiants/curieux.

## 4) Livrables

- Lien GitHub (code source propre).
- Lien Netlify (site en ligne).
- Documents préparatoires (wireframes, planning).

## 5) Exigences techniques

Stack : Vue 3 (JS), Pinia, Router, Three.js, SCSS, Vite, Netlify.

API : data.gouv.nc /rorc\_poissons\_recifaux, 20 résultats max, filtres par campagne/site/station/transect/type.

## 6) Spécifications fonctionnelles

- 1 thon = 1 enregistrement API.
- Scène 3D avec flocking léger.
- Tooltip au survol, panneau d'info au clic.

- Filtrage via sidebar.
- Recentrage caméra.

## 7) UI/UX

- Thème sombre, responsive.
- TopBar (recherche + recentrer), Sidebar (filtres), InfoPanel (détails).
- Tooltip interactif.
- Accessibilité basique (contraste, labels, focus).

## 8) Planning

J1 Setup, J2 API, J3 Environnement 3D, J4 UI/UX, J5 Tests/perf, J6 Déploiement.

## 9) Tests & critères d'acceptation

- Chargement 20 résultats sans erreurs.
- Filtrage OK.
- Tooltip + clic OK.
- Aucun poisson ne traverse le sol.
- Responsive mobile/desktop.

## 10) Risques

- CORS (contourné via proxy/redirects).
- GLB lourd (fallback capsule).
- API change (normalisation).

11) Wireframe

