

PARTIE CÉLIAN :

- **GET php/get_bateaux.php**
→ Cette interface permet de **récupérer la liste complète des bateaux enregistrés** dans la base. C'est ce qui alimente notamment la carte interactive sur la page de visualisation.
- **GET php/predict-cluster-type.php?mmsi=...**
→ Cette interface appelle un script Python pour **prédire automatiquement le type de bateau** à partir de ses caractéristiques. Le résultat est retourné en JSON.
- **GET php/predictCluster.php**
→ Cette interface appelle un autre script Python qui **regroupe les navires en clusters** selon leur comportement de navigation (vitesse, cap, position...). Le résultat est affiché sous forme de carte avec des couleurs différentes par cluster.

PARTIE NOLAN :

Partie 5: **POST php/predict_traj.php?mmsi=...&horizon=...**

→ Cette interface appelle un script Python pour prédire la trajectoire future d'un navire identifié par son MMSI.

Elle accepte les données via POST, GET, ou en brut (JSON ou x-www-form-urlencoded) — ce qui la rend compatible avec des appels fetch, curl ou formulaires.

Elle prend deux paramètres :

- mmsi (obligatoire) : identifiant unique du navire
- horizon (optionnel, valeurs possibles : 5, 10, 15) : durée de prédiction en minutes (défaut : 5)

Le script Python renvoie la position actuelle et la position prédite, encapsulées dans un objet JSON. En cas d'erreur (paramètre manquant, problème d'exécution Python, réponse invalide), un message structuré avec success: false et des informations de débogage est retourné.

partie3

- **GET php/get_bateaux2.php** → Cette interface permet de **récupérer la liste complète des bateaux avec leurs dernières positions** depuis la base de données. Elle groupe les données par MMSI et retourne la position la plus

récente de chaque bateau, formatée avec les coordonnées, vitesses et caps. C'est utilisé pour alimenter le tableau principal et la carte générale.

- **GET `php/get_trajetoire.php?mmsi=...`** → Cette route permet de **recupérer l'historique complet des positions d'un bateau spécifique** identifié par son MMSI. Elle retourne toutes les positions ordonnées chronologiquement pour tracer la trajectoire complète du navire sur la carte interactive.

Partie5 :type

- **POST `php/predict_type.php`** → Cette interface permet de **prédire le type de navire** à partir de ses caractéristiques physiques (statut, longueur, largeur, tirant d'eau). Elle reçoit les données au format JSON, les valide, puis appelle un script Python de machine learning pour effectuer la prédiction. Le résultat contient les prédictions de différents modèles et est retourné au format JSON pour alimenter l'interface utilisateur de classification automatique des navires.
-

Partie 2

- **POST `php/traitement.php`** → Cette interface permet d'**ajouter un nouveau bateau** dans la base de données avec toutes ses caractéristiques et sa position actuelle. Elle reçoit les données via un formulaire POST, effectue une validation complète des champs obligatoires (MMSI, nom, statut, dimensions, coordonnées, etc.), puis insère le navire dans la table Bateaux. Cette route est utilisée par le formulaire d'ajout manuel de navires sur l'interface utilisateur.
- **GET `php/get_etats.php`** → Cette interface permet de **recupérer la liste complète des statuts de navires** disponibles dans la base de données. Elle retourne tous les types de statuts avec leurs IDs et noms, triés par ordre croissant. Cette route alimente notamment les listes déroulantes du formulaire d'ajout de bateau pour permettre à l'utilisateur de sélectionner le bon statut du navire.