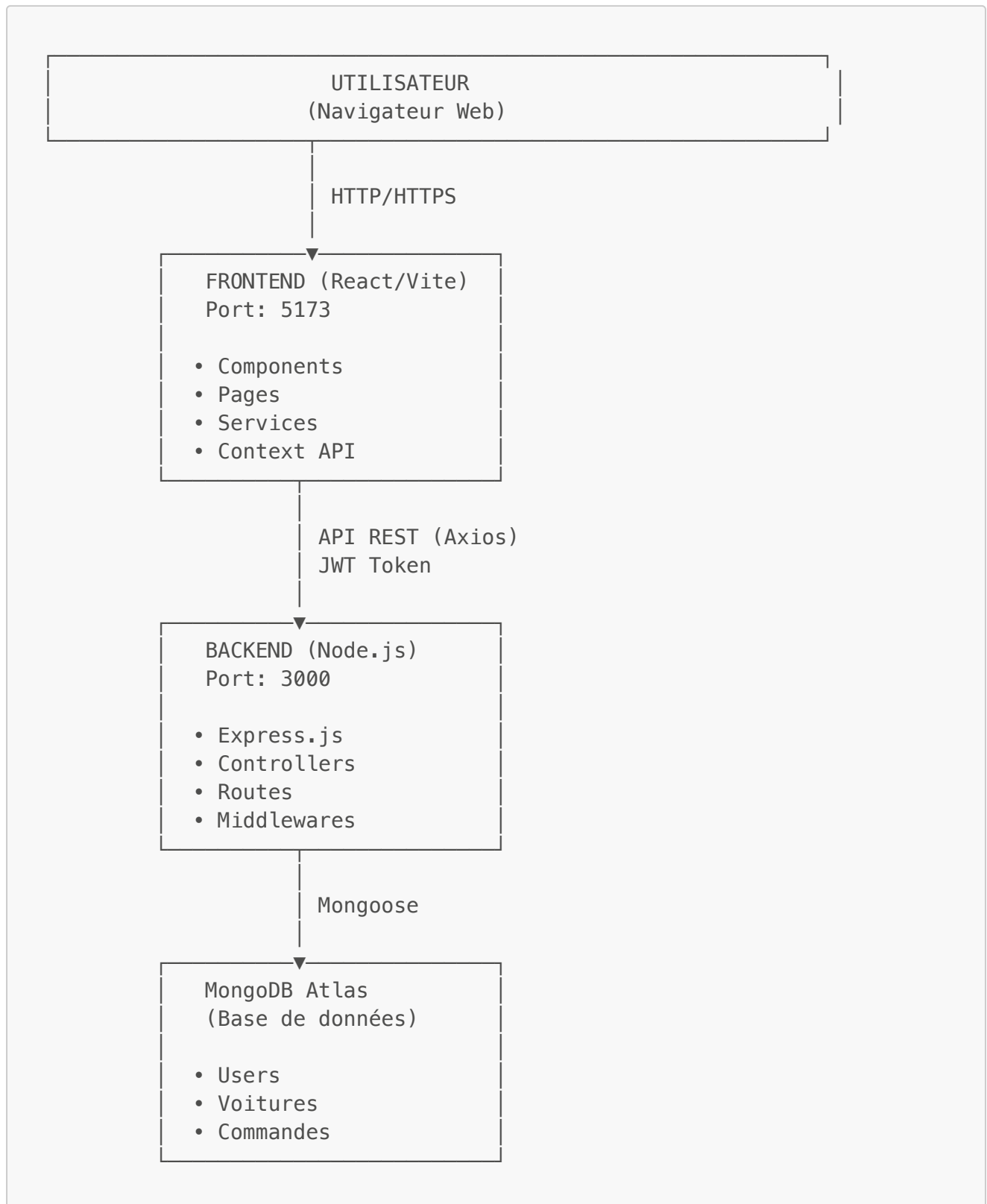


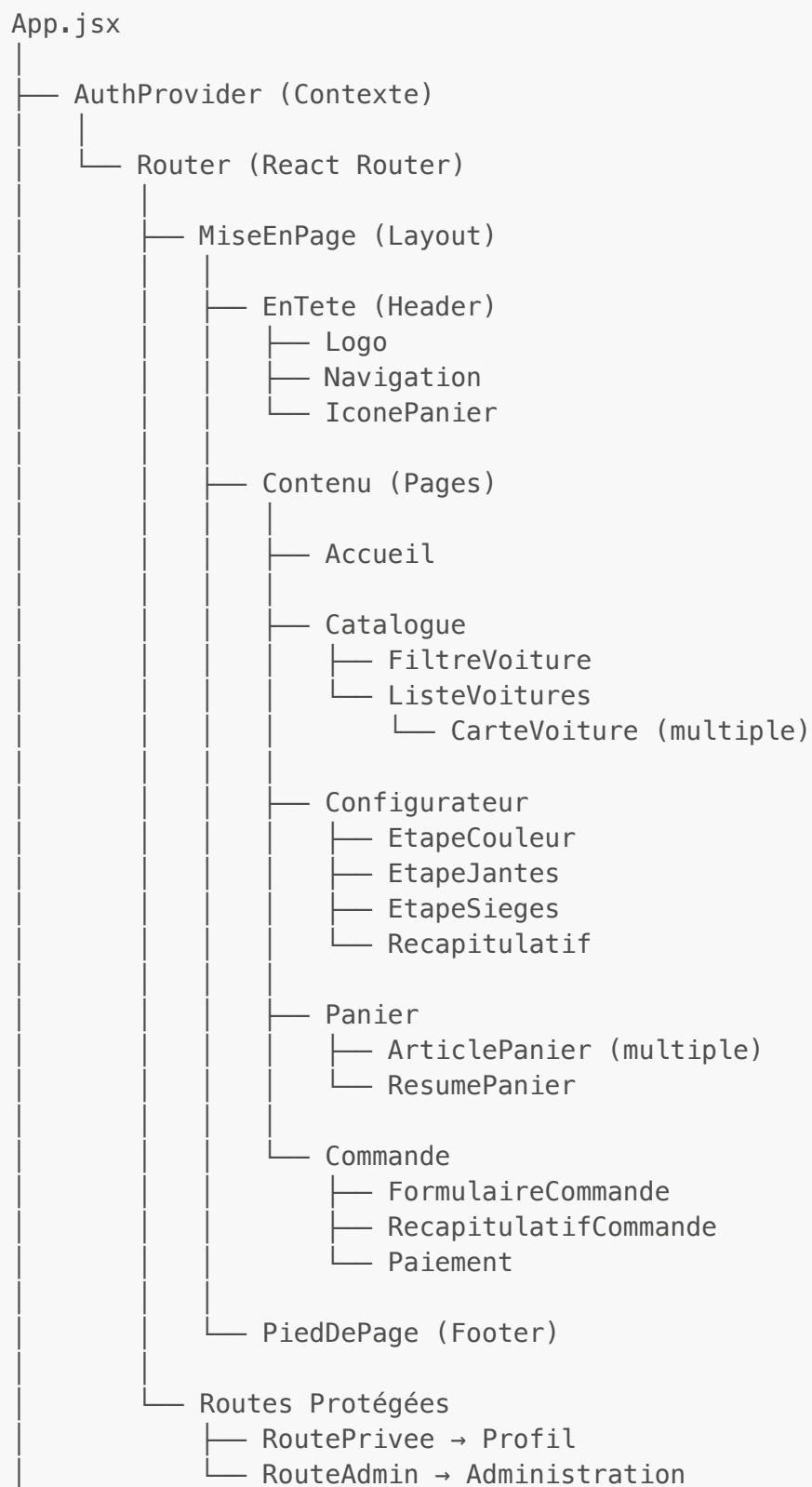
Diagrammes d'Architecture - Plateforme Porsche

Vue d'ensemble du système

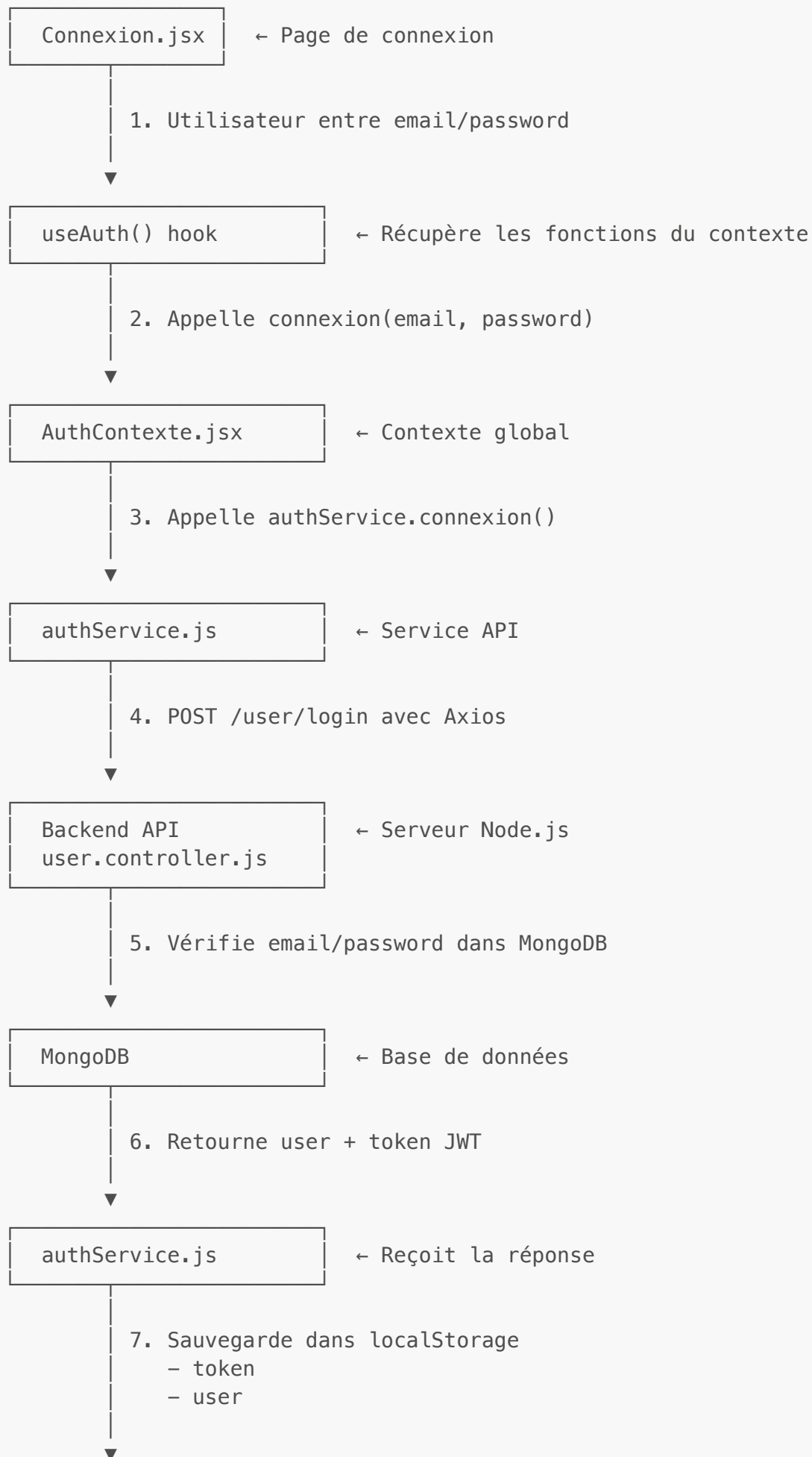


Architecture Frontend (React)

Hiérarchie des composants

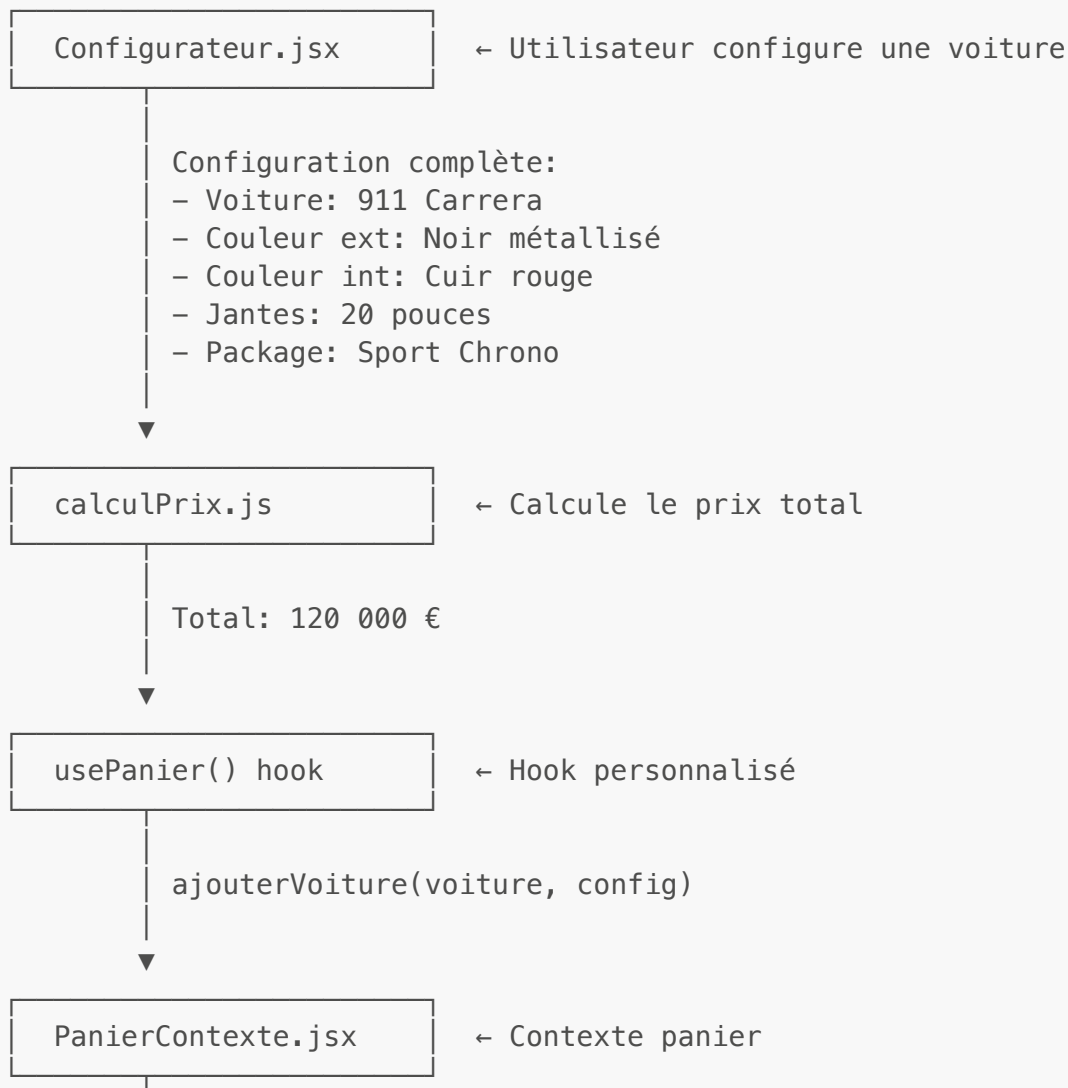


Flow de données : Authentification





Flow de données : Ajout au panier



```
ajouterArticle({  
  type: 'voiture',  
  voiture,  
  configuration,  
  prix: 120000  
})
```

localStorage

← Sauvegarde automatique

Panier sauvegardé

IconePanier.jsx
(dans EnTete)

← Badge mis à jour
Affiche: 1 article

🎨 Architecture par couches

COUCHE PRÉSENTATION
(Ce que l'utilisateur voit)

Pages/
.jsx

Components
.jsx

Layout
.jsx

Styles
.css

Props, Events

COUCHE LOGIQUE
(Gestion de l'état et des données)

Hooks/
.js

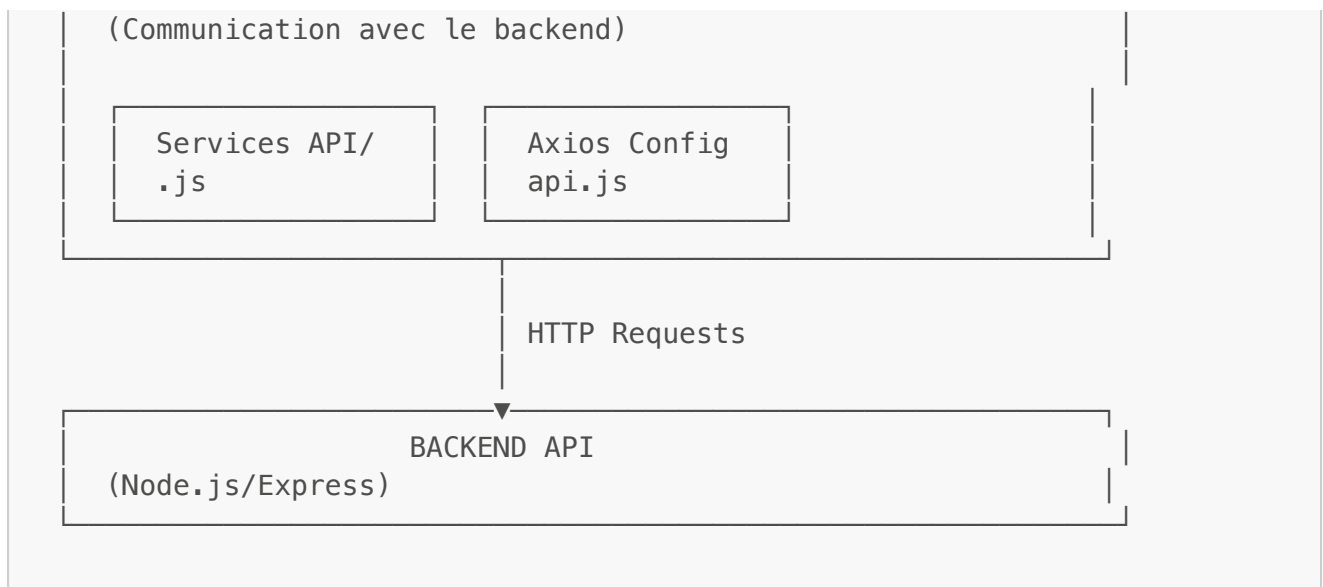
Context
.jsx

Utils/
.js

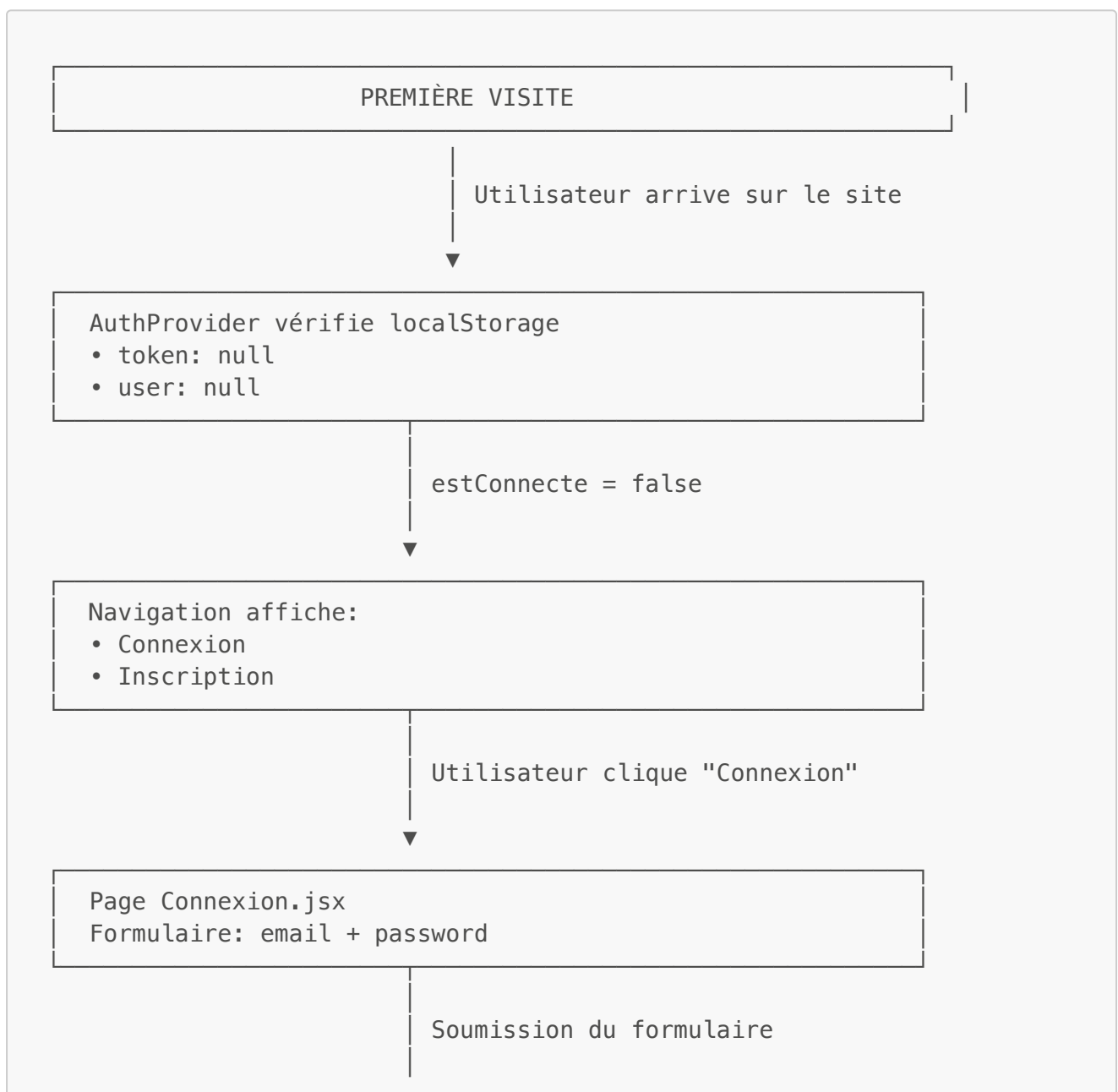
Helpers
.js

API Calls

COUCHE SERVICES



Système d'authentification



authService.connexion(email, password)
→ POST /user/login

Réponse: { token, user }

Sauvegarde dans localStorage:
• localStorage.setItem('token', token)
• localStorage.setItem('utilisateur', JSON.stringify(user))

AuthContexte met à jour

État global mis à jour:
• utilisateur: { email, nom, prenom, role }
• estConnecte: true

Tous les composants sont notifiés

Navigation affiche maintenant:
• Mon profil
• Mes commandes
• Déconnexion

Toutes les requêtes API incluent:
Authorization: Bearer <token>

Utilisateur peut accéder aux routes protégées

VISITE SUIVANTE

Utilisateur revient sur le site

AuthProvider vérifie localStorage
• token: "eyJhbGciOiJIUzI1..."

• user: { email, nom, prenom, role }

Token trouvé !

AuthContexte charge l'utilisateur

- setUser(user)
- estConnecte: true

Utilisateur automatiquement connecté

L'utilisateur voit son profil sans se reconnecter



Protection des routes

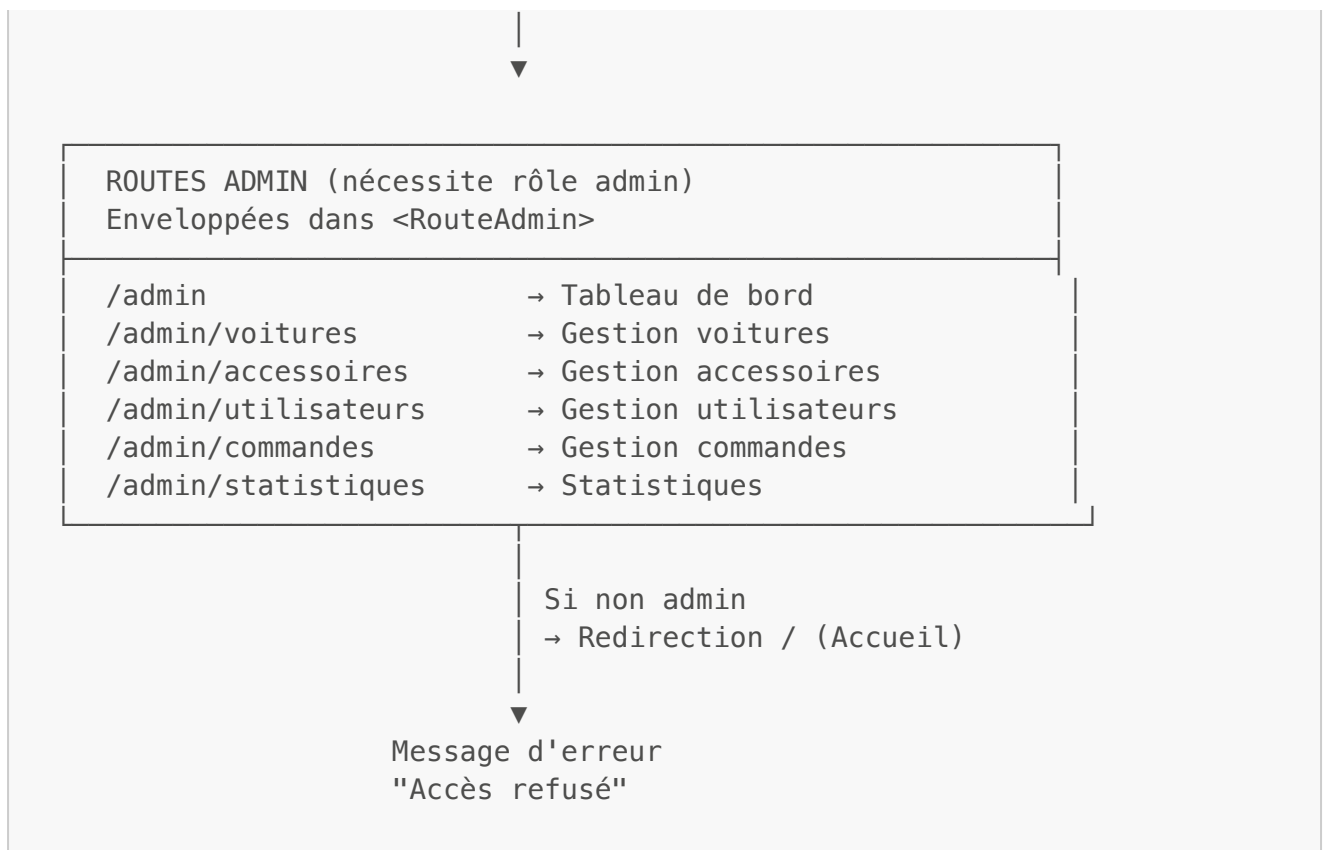
ROUTES PUBLIQUES (accessible à tous)

/	→ Accueil
/connexion	→ Connexion
/inscription	→ Inscription
/catalogue	→ Catalogue voitures
/catalogue/:id	→ Détails voiture
/accessoires	→ Catalogue accessoires

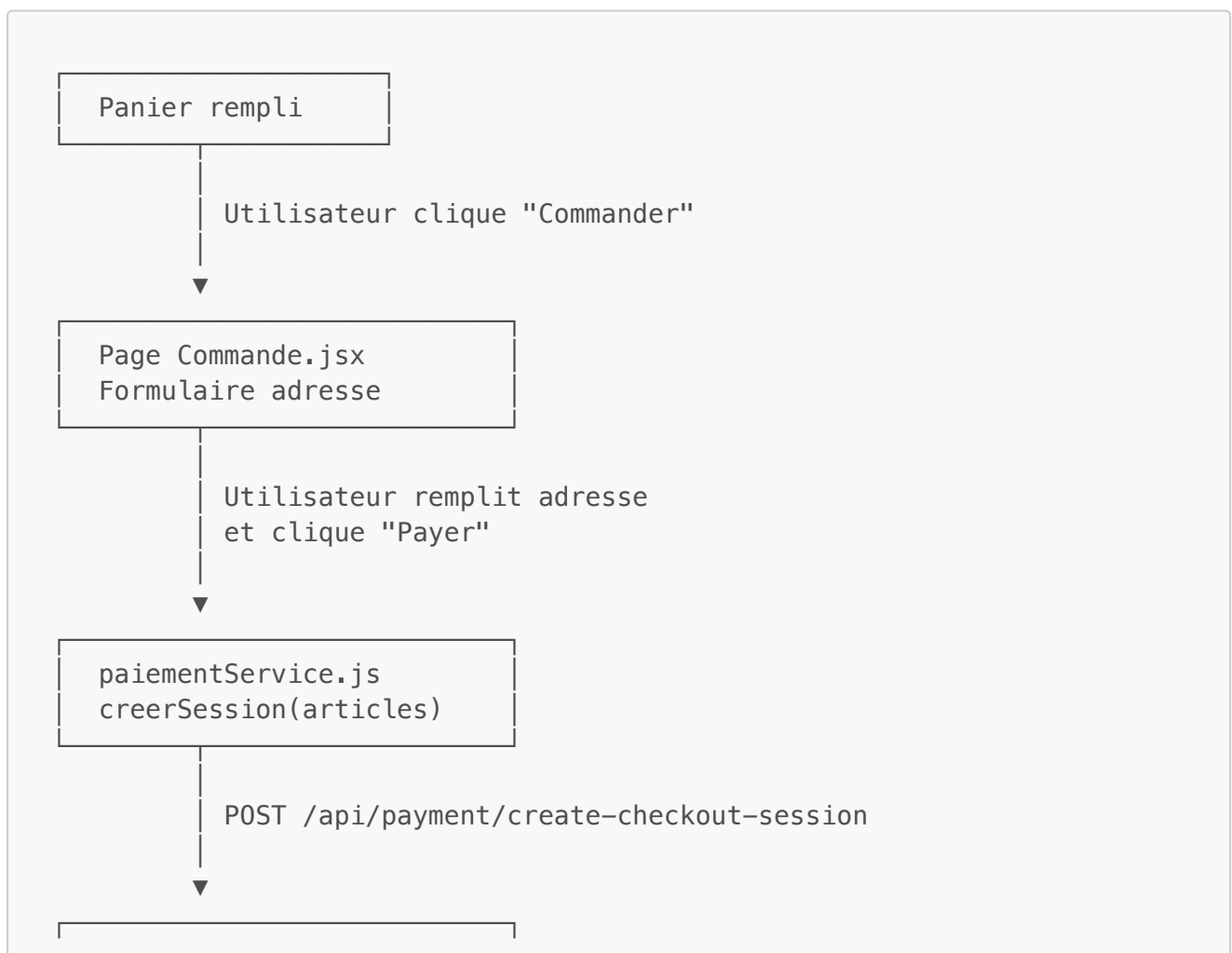
ROUTES PRIVÉES (nécessite authentification) Enveloppées dans <RoutePrivee>

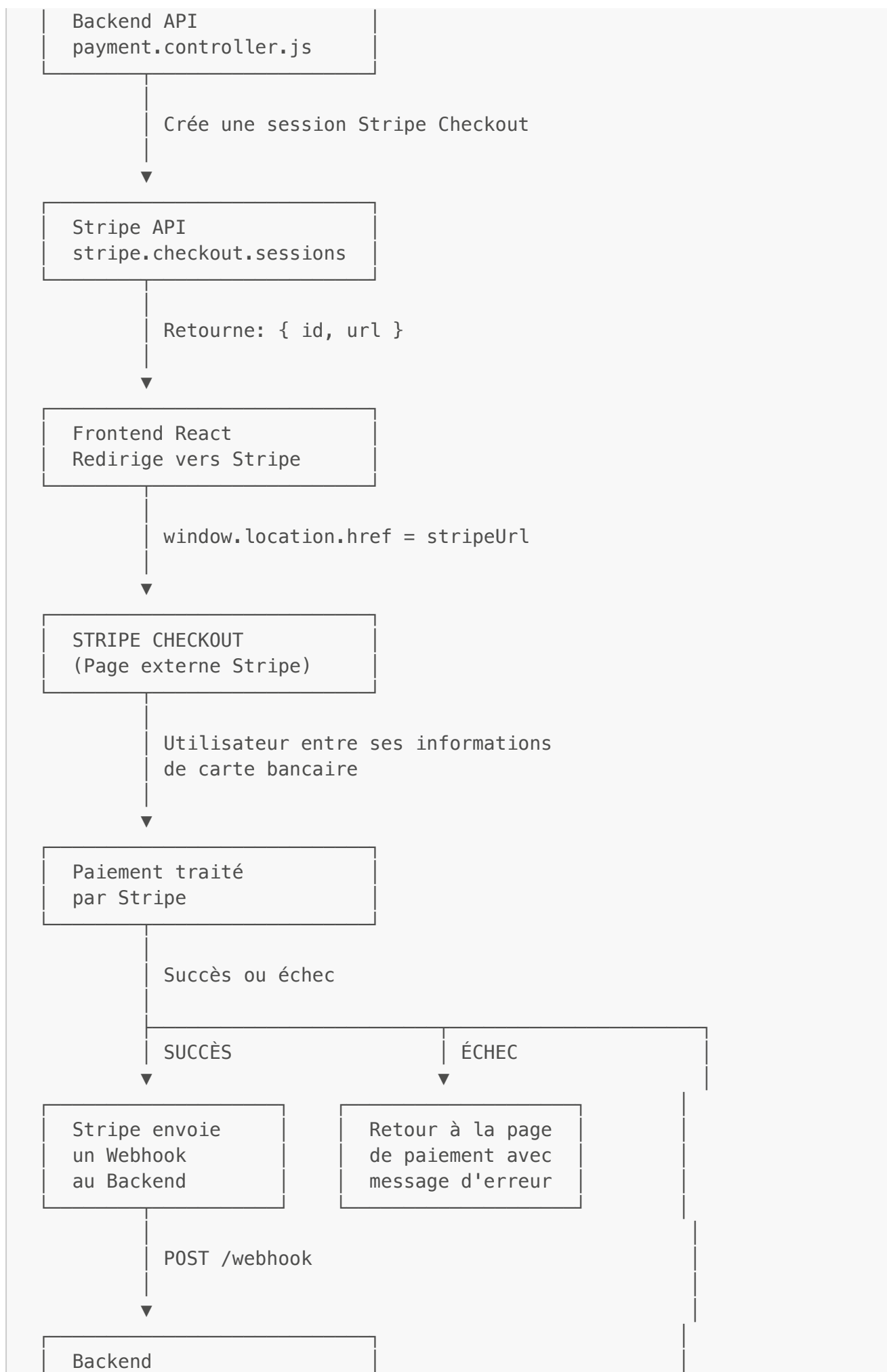
/configureur/:id	→ Configureur voiture
/panier	→ Panier
/commande	→ Formulaire commande
/paiement	→ Paiement
/profil	→ Mon profil
/mes-commandes	→ Historique commandes
/mes-reservations	→ Mes réservations

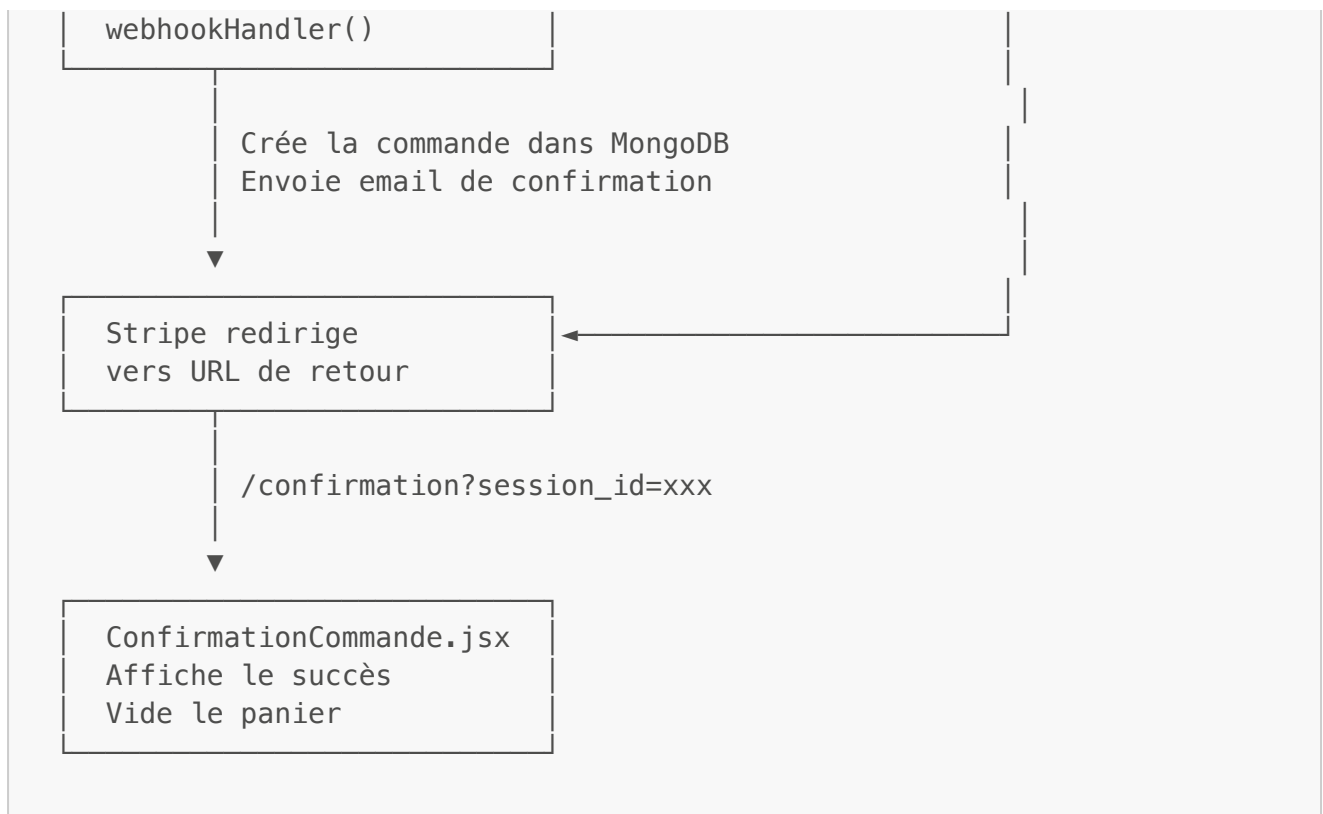
Si non connecté
→ Redirection /connexion



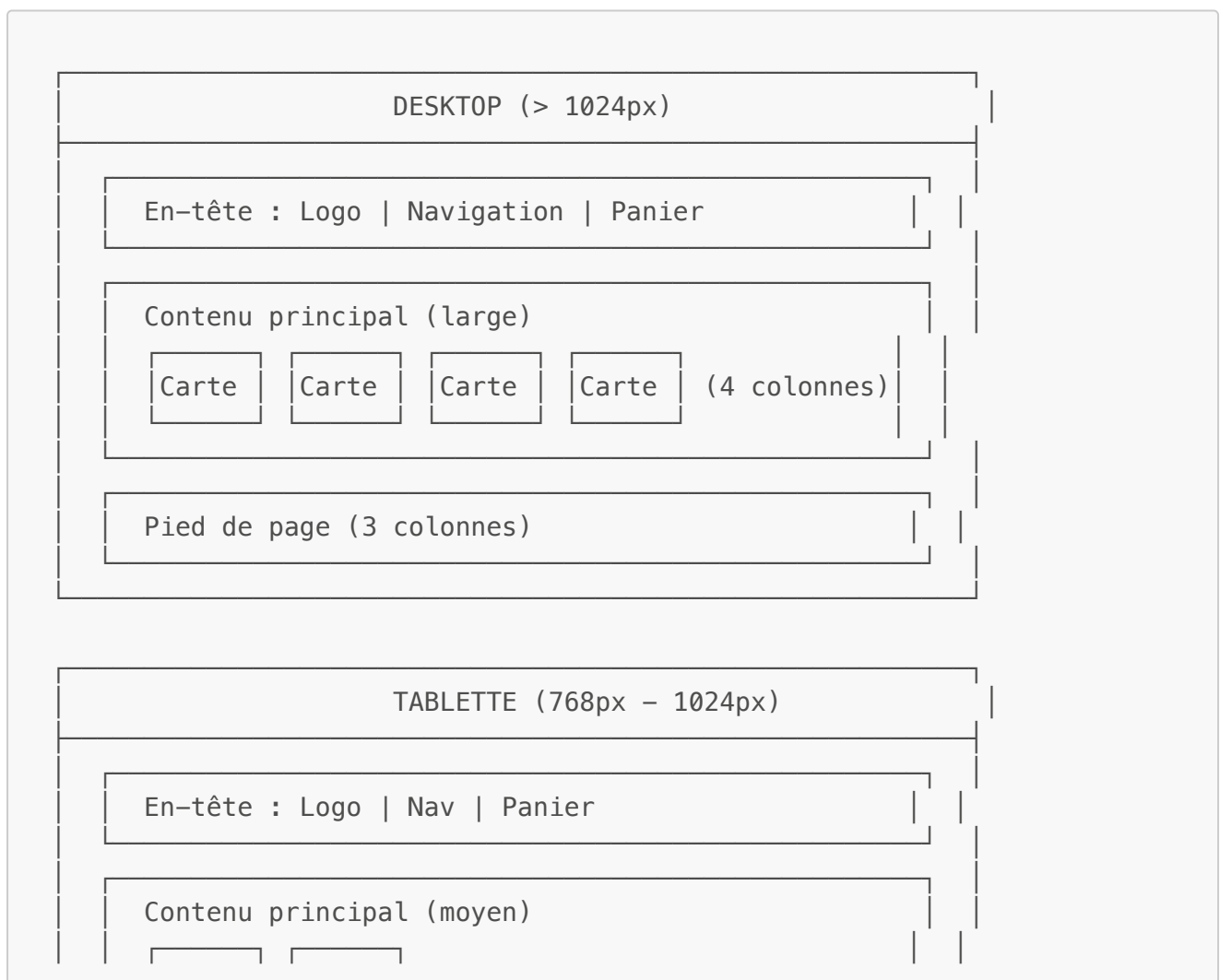
Flow de paiement Stripe

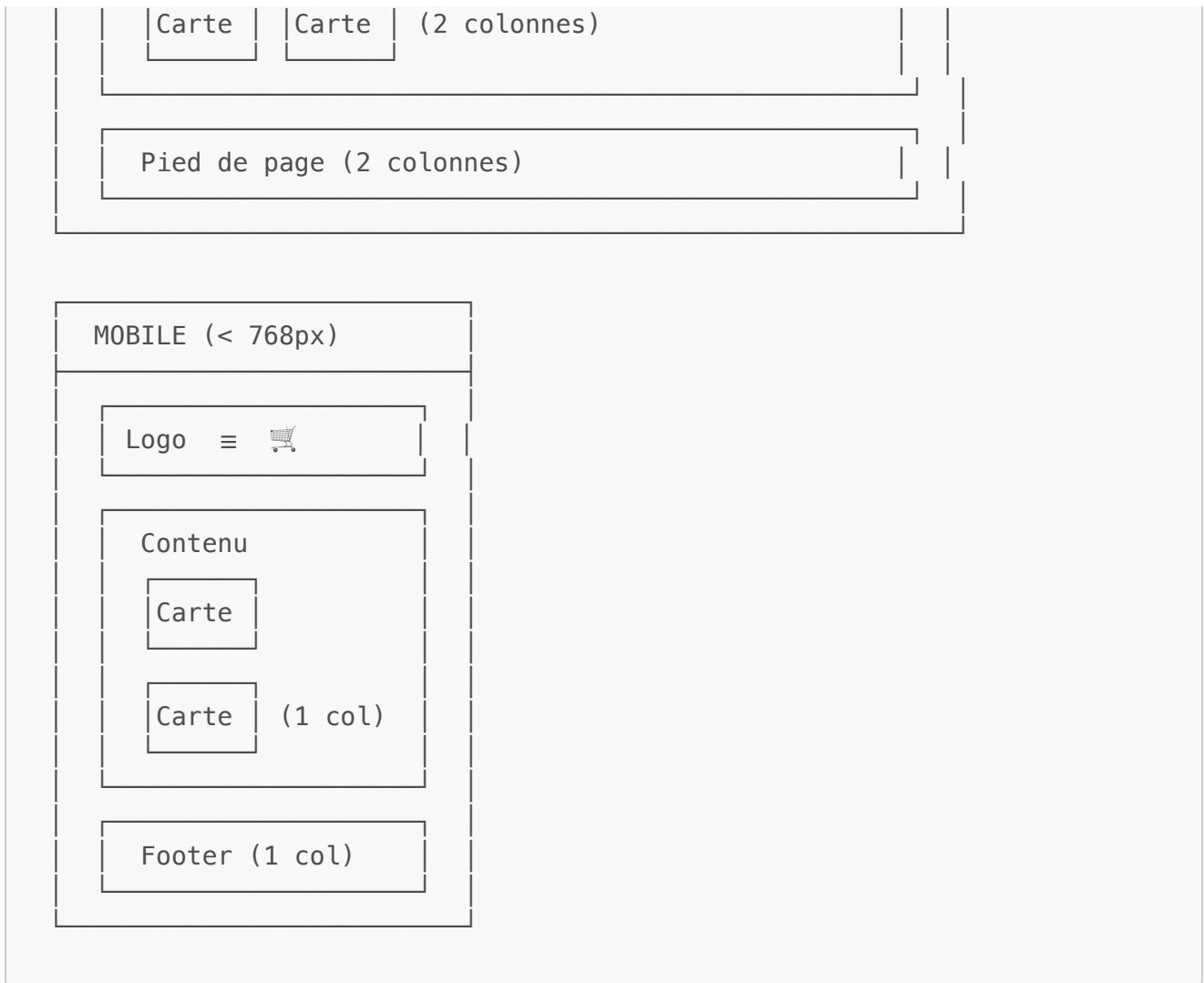






📱 Responsive Design





Points clés à retenir

1. Séparation des responsabilités

Services → Récupèrent les données
Hooks → Gèrent la logique
Contextes → Partagent l'état
Composants → Affichent l'UI

2. Flux unidirectionnel des données

API → Service → Hook → Contexte → Composant → UI

3. Protection en couches

Routes protégées → Vérification token → Vérification rôle → Accès autorisé

4. État global vs État local

État global (Context) :

- Authentification
- Panier
- Thème

État local (useState) :

- Formulaires
- Modales ouvertes/fermées
- Sélections temporaires

Ces diagrammes vous aident à visualiser l'architecture. Utilisez-les comme référence ! 