

Bootcamp Web Digi5

-- 2025 – 2026 --

Objectifs pédagogiques

- Comprendre les enjeux d'un projet e-commerce
- Maîtriser les concepts essentiels (CRUD, rôles utilisateurs, RGPD)
- Planifier efficacement un projet en équipe

A. Les composants essentiels

Frontend (Interface utilisateur)

- Pages produits avec filtres et recherche
- Système de panier et checkout
- Interface d'administration
- Responsive design obligatoire

Backend (Logique métier)

- Gestion des utilisateurs et authentification
- API REST pour les opérations CRUD
- Gestion des stocks et commandes
- Système de paiement (fictif)

Base de données

- Modèle relationnel optimisé
- Index pour les performances
- Sauvegarde et migration
- Données scrapées et nettoyées

✗ Systèmes EXCLUS pour ce projet

SQLite

Problème principal : Concurrency limitée (une seule écriture à la fois)

Impact : Scripts de scraping + utilisateurs simultanés = goulot d'étranglement

Limitation : Pas de vrais types de données (tout stocké comme TEXT)

Verdict : Inadapté pour un projet avec scraping intensif

Bases NoSQL pures (MongoDB, DynamoDB)

Problème : Pas de relations natives (JOIN coûteux)

Impact : Structure produit/variante/commande devient complexe

Bases orientées graphes (Neo4j)

Overkill pour ce use case

Complexité : Courbe d'apprentissage trop élevée pour un bootcamp

Coût : Solutions payantes pour la production

Base de données

- Modèle relationnel optimisé
- Index pour les performances
- Sauvegarde et migration
- Données scrapées et nettoyées

⚠️ Systèmes DÉCONSEILLÉS (mais possibles)

MySQL (version < 8.0)

Limitations : Support JSON limité, pas de CTE récursives

Performance : Index moins optimisés que PostgreSQL

Scraping : Gestion des types de données moins flexible

MariaDB

Acceptable mais PostgreSQL reste supérieur

Avantage : Compatible MySQL, léger

Inconvénient : Écosystème moins riche

Base de données

- Modèle relationnel optimisé
- Index pour les performances
- Sauvegarde et migration
- Données scrapées et nettoyées

✓ Systèmes RECOMMANDÉS

🏆 **PostgreSQL** (FORTEMENT RECOMMANDÉ)

Pourquoi parfait pour ce projet :

Avantages spécifiques :

JSONB : Parfait pour stocker données scraping variables

Index GIN/GiST : Recherche textuelle ultra-rapide

Contraintes : Validation des données au niveau BDD

Extensions : pg_trgm pour recherche floue, uuid-oss pour identifiants

🥈 **MySQL 8.0+** (Alternative solide)

Avantages :

Popularité : Plus d'hébergeurs compatibles

Performance : Très bon sur les lectures

Outils : phpMyAdmin, MySQL Workbench



Justification technique

Volume de données estimé :

500 références × 5 variantes moyennes =
2,500 variantes

500 produits × 3 images moyennes = 1,500
images

50 utilisateurs × 10 commandes = 500
commandes

Total : ~5,000 enregistrements principaux

Performance attendue avec PostgreSQL :

Recherche : < 50ms avec index appropriés
Import scraping : 1000+ produits/minute
Concurrent users : 50+ simultanés sans
problème

Verdict final : PostgreSQL 15 + Redis pour
cache = configuration idéale pour ce
bootcamp 🚀

Sécurité E-commerce - Points Clés



Authentification & Autorisation

Autorisation

- **Principe du moindre privilège :**
Rôles avec permissions minimales nécessaires
- **Validation côté serveur :** Jamais de confiance au frontend seul
- **Middleware d'autorisation :**
Vérification systématique des droits
- **Tokens à durée limitée :**
Renouvellement automatique requis

Autorisation

- **Principe du moindre privilège :**
Rôles avec permissions minimales nécessaires
- **Validation côté serveur :** Jamais de confiance au frontend seul
- **Middleware d'autorisation :**
Vérification systématique des droits
- **Tokens à durée limitée :**
Renouvellement automatique requis

Sécurité E-commerce - Points Clés



Protection contre les Injections

SQL Injection

- **ORM obligatoire** : Prisma, TypeORM, ou requêtes préparées uniquement
- **Validation d'entrée** : express-validator ou équivalent
- **Sanitization** : Nettoyage systématique des inputs utilisateur
- **Principe de liste blanche** : Validation positive plutôt que blacklist

XSS (Cross-Site Scripting)

- **Échappement automatique** : React JSX par défaut
- **Content Security Policy** : Headers stricts pour les sources autorisées
- **DOMPurify** : Sanitization du HTML si nécessaire
- **Validation frontend ET backend** : Double vérification obligatoire

Sécurité E-commerce - Points Clés



Sécurisation des Sessions & Cookies

HTTPS obligatoire : Certificats SSL/TLS en production

Secure cookies : Flag secure + httpOnly + sameSite

CSRF Protection : Tokens CSRF sur formulaires sensibles

Session timeout : Expiration automatique après inactivité

Invalidation : Suppression sessions côté serveur à la déconnexion

Sécurité E-commerce - Points Clés



Validation & Sanitization

Côté Backend

- **Schema validation** : Zod, Joi ou express-validator
- **Type checking** : TypeScript fortement recommandé
- **Limites de taille** : Payload, upload files, string length
- **Format validation** : Email, URL, numéros, dates

Côté Frontend

- **Validation temps réel** : react-hook-form + resolvers
- **Messages d'erreur sécurisés** : Pas d'exposition d'infos sensibles
- **Désactivation submit** : Pendant validation/soumission

Sécurité E-commerce - Points Clés



Headers de Sécurité

Helmet.js : Configuration automatique des headers

HSTS : Force HTTPS avec includeSubDomains

X-Frame-Options : Protection contre clickjacking

X-Content-Type-Options : Prévention MIME sniffing

Referrer-Policy : Contrôle des informations de référence

Sécurité E-commerce - Points Clés

 Bénéfices Sécurité : React/Angular + Node.js Express

React - Sécurité Frontend

- Protection XSS native
- Gestion d'état sécurisée
- Validation côté client intégrée

Angular - Sécurité Renforcée


- Protection XSS native
- Gestion d'état sécurisée
- Validation côté client intégrée
- Sanitization automatique
- Guards de sécurité intégrés
- HttpInterceptors pour sécurité

Node.js Express - Backend Sécurisé

- Écosystème de sécurité mature
- Validation et sanitization centralisées
- Authentification robuste


Sécurité E-commerce - Points Clés

 Bénéfices Sécurité : React/Angular + Node.js Express

-  Verdict sécurité : Angular légèrement avantage
- **Pourquoi Angular est plus sécurisé :**
- TypeScript obligatoire = Moins d'erreurs de types
- Sanitization plus stricte par défaut
- Architecture plus structurée = Moins d'erreurs architecture
- Guards natifs pour protection des routes
- HttpInterceptors pour sécurité centralisée
-
- **Pourquoi React reste excellent :**
- Simplicité = Moins de complexité = Moins d'erreurs
- Écosystème mature pour authentification
- Flexibilité pour solutions sur mesure
- Performance = UX sécurisée (pas de freeze)

Sécurité E-commerce - Points Clés

 Bénéfices Sécurité : React/Angular + Node.js Express

-  Angular gagne légèrement en sécurité pure, React gagne en simplicité d'implémentation sécurisée.
- Pour le bootcamp : React recommandé (plus simple à sécuriser correctement en 6 jours) !

Sécurité E-commerce - Points Clés

SSR vs CSR avec React - Guide Complet

- CSR (Client-Side Rendering)

- Navigateur reçoit → HTML vide + JS bundle → JS s'exécute → Page s'affiche

- SSR (Server-Side Rendering)

- Serveur génère → HTML complet → Navigateur affiche → JS hydrate

- SSG (Static Site Generation)

- Build time → HTML pré-généré → CDN serve → JS hydrate si besoin

Avantages CSR :

- ✓ Navigation ultra-rapide après le premier chargement
- ✓ Interactions riches (SPA fluide)
- ✓ Simple à développer et héberger
- ✓ Moins de charge serveur (statique)
- ✓ Perfect pour applications (dashboard, admin)

Inconvénients CSR :

- ✗ SEO catastrophique (Google voit une page vide)
- ✗ Premier chargement lent (gros bundle JS)
- ✗ UX loading (écrans blancs)
- ✗ Performance mobile (JS lourd)
- ✗ Pas d'aperçu réseaux sociaux (Open Graph vide)

Sécurité E-commerce - Points Clés

SSR vs CSR avec React - Guide Complet

- CSR (Client-Side Rendering)

- Navigateur reçoit → HTML vide + JS bundle → JS s'exécute → Page s'affiche

- SSR (Server-Side Rendering)

- Serveur génère → HTML complet → Navigateur affiche → JS hydrate

- SSG (Static Site Generation)

- Build time → HTML pré-généré → CDN serve → JS hydrate si besoin

Avantages SSR :

- ✓ **SEO parfait** (Google voit tout le contenu)
- ✓ **Performance initiale** (contenu visible immédiatement)
- ✓ **Partage réseaux sociaux** (Open Graph fonctionnel)
- ✓ **Accessibilité** (contenu sans JS)
- ✓ **Core Web Vitals** optimisés

Inconvénients SSR :

- ✗ **Complexité architecture** (serveur Node.js)
- ✗ **Charge serveur** (rendu à chaque requête)
- ✗ **Coût hébergement** plus élevé
- ✗ **Hydratation** peut être lente
- ✗ **Debugging** plus complexe



Avantages :

- **Une seule langue** : JavaScript partout
- **Apprentissage simplifié** : Pas de context switching
- **Écosystème cohérent** : npm pour tout
- **Déploiement unifié** : Vercel/Netlify + Railway

Next.js App (Frontend + API Routes)



Database (PostgreSQL + Prisma)



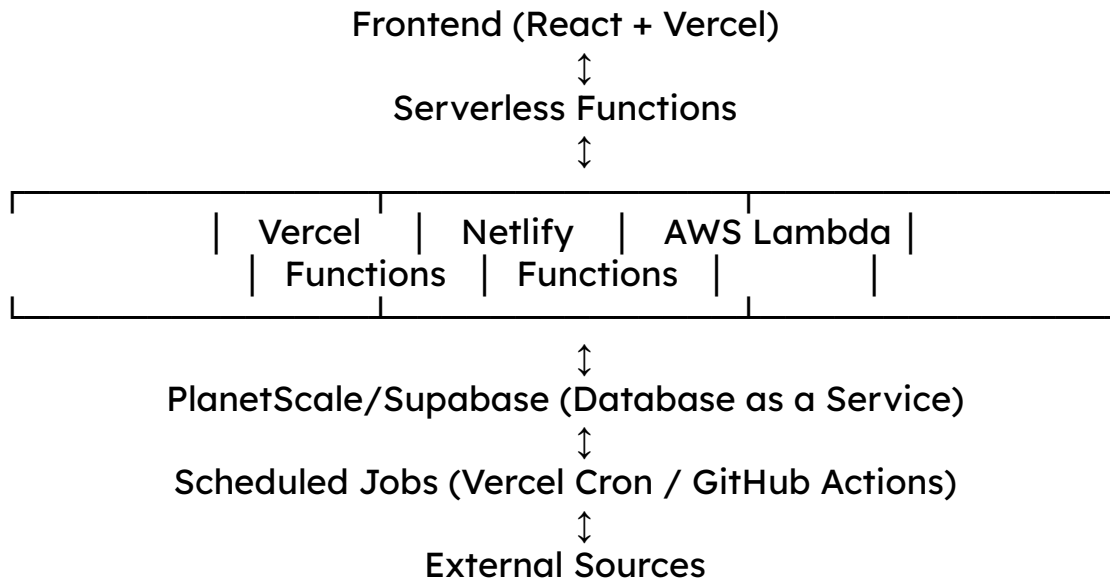
Scraping Service (API Routes + Queue)



External Sources

Avantages :

- **Performance** : SSR/SSG natif, optimisations automatiques
- **DX** : Hot reload full-stack, TypeScript natif
- **Déploiement** : Vercel one-click
- **SEO** : Rendu serveur pour le référencement



Avantages :

- **Performance** : SSR/SSG natif, optimisations automatiques
- **DX** : Hot reload full-stack, TypeScript natif
- **Déploiement** : Vercel one-click
- **SEO** : Rendu serveur pour le référencement



Comparatif des Variantes

Variante	Difficulté	Performance	Scalabilité	Coût	Idéal pour
Full JS	★ ★	★ ★ ★	★ ★	Gratuit	Bootcamp débutants
Next.js	★ ★ ★	★ ★ ★ ★ ★ ★	★ ★ ★ ★	Gratuit	Projets modernes
Python Backend	★ ★ ★	★ ★ ★ ★	★ ★ ★	Faible	Data-heavy
Microservices	★ ★ ★ ★ ★	★ ★ ★ ★ ★ ★	★ ★ ★ ★ ★	Élevé	Entreprise
Serverless	★ ★ ★ ★	★ ★ ★ ★	★ ★ ★ ★ ★	Variable	Startup
JAMstack	★ ★ ★	★ ★ ★ ★ ★ ★	★ ★ ★	Faible	Sites vitrine

```
-- Table des marques
CREATE TABLE brands (
  id SERIAL PRIMARY KEY,
  name VARCHAR(100) NOT NULL,
  logo_url VARCHAR(255),
  created_at TIMESTAMP DEFAULT NOW()
);
```

```
CREATE TABLE products (
  id SERIAL PRIMARY KEY,
  name VARCHAR(255) NOT NULL,
  brand_id INTEGER REFERENCES brands(id),
  category_id INTEGER REFERENCES
categories(id),
  description TEXT,
  base_price DECIMAL(10,2),
  sku VARCHAR(50) UNIQUE,
  is_active BOOLEAN DEFAULT true,
  scraped_from VARCHAR(255),
  created_at TIMESTAMP DEFAULT NOW(),
  updated_at TIMESTAMP DEFAULT NOW()
);
```

```
-- Table des catégories
CREATE TABLE categories (
  id SERIAL PRIMARY KEY,
  name VARCHAR(100) NOT NULL,
  slug VARCHAR(100) UNIQUE,
  parent_id INTEGER REFERENCES
categories(id)
);
```

```
-- Table des variantes (tailles/couleurs)
CREATE TABLE product_variants (
  id SERIAL PRIMARY KEY,
  product_id INTEGER REFERENCES
products(id),
  size DECIMAL(4,1),
  color VARCHAR(50),
  stock_quantity INTEGER DEFAULT 0,
  price DECIMAL(10,2),
  sku VARCHAR(50) UNIQUE,
  created_at TIMESTAMP DEFAULT NOW()
);
```

```
-- Table des images
CREATE TABLE product_images (
  id SERIAL PRIMARY KEY,
  product_id INTEGER REFERENCES
products(id),
  image_url VARCHAR(500) NOT NULL,
  alt_text VARCHAR(255),
  is_primary BOOLEAN DEFAULT false,
  display_order INTEGER DEFAULT 0
);
```

Exercice pratique (15 min)

Dessiner le schéma de base de données complet sur papier

Module 2 : Web Scraping et constitution de la BDD (75 min)

A. Concepts du web scraping



Qu'est-ce que le scraping ?

- Extraction automatisée de données depuis des sites web
- Transformation des données HTML en données structurées
- Respect des robots.txt et des conditions d'utilisation
- Gestion de la charge sur les serveurs cibles

Défis techniques du scraping :

- JavaScript dynamique (SPA)
- Rate limiting et anti-bot
- Structure HTML changeante
- Données manquantes ou incohérentes

Module 2 : Web Scraping et constitution de la BDD (75 min)

- B. Outils de scraping recommandés
-  Python (Le plus populaire)
 - a. **Beautiful Soup + Requests** - Simple et efficace
 - b. **Selenium** - Pour sites JavaScript
-  Node.js (Alternative)
 - a. **Puppeteer** - Chrome headless
 - b. **Cheerio** - jQuery pour Node.js

Module 2 : Web Scraping et constitution de la BDD (75 min)

C. Sources de données recommandées

Sites français populaires :

- **Courir** (courir.com) - Large catalogue, structure claire
- **JD Sports** (jdsports.fr) - Interface moderne
- **Size?** (size.fr) - Sneakers premium
- **Footlocker** (footlocker.fr) - Classique et complet
- **Sarenza** (sarenza.com) - Données structurées

Sites internationaux :

- **StockX** (stockx.com) - Prix du marché, API disponible
- **GOAT** (goat.com) - Authentification, données riches
- **Grailed** (grailed.com) - Streetwear et vintage
- **Flight Club** (flightclub.com) - Sneakers rares

Module 2 : Web Scraping et constitution de la BDD (75 min)

C. Sources de données recommandées

Sites français populaires :

- **Courir** (courir.com) - Large catalogue, structure claire
- **JD Sports** (jdsports.fr) - Interface moderne
- **Size?** (size.fr) - Sneakers premium
- **Footlocker** (footlocker.fr) - Classique et complet
- **Sarenza** (sarenza.com) - Données structurées


Sites internationaux :

- **StockX** (stockx.com) - Prix du marché, API disponible
- **GOAT** (goat.com) - Authentification, données riches
- **Grailed** (grailed.com) - Streetwear et vintage
- **Flight Club** (flightclub.com) - Sneakers rares

APIs alternatives (plus faciles) :

- **StockX API** (accès limité mais officiel)
- **SneakerDB API** (base de données complète)
- **Foot Locker API** (partenariat possible)
- **Adidas/Nike Developer** (produits officiels)

Module 2 : Web Scraping et constitution de la BDD (75 min)

-  **Exercice pratique (30 min)**
 - a. Scraper 50 produits depuis un site au choix avec Python/BeautifulSoup



Module 3 : Gestion des rôles et BDD (45 min)

A. Modèle utilisateurs avec BDD

```
-- Table des utilisateurs
CREATE TABLE users (
  id SERIAL PRIMARY KEY,
  email VARCHAR(255) UNIQUE NOT NULL,
  password_hash VARCHAR(255) NOT NULL,
  first_name VARCHAR(100),
  last_name VARCHAR(100),
  role ENUM('customer', 'moderator', 'admin')
  DEFAULT 'customer',
  is_active BOOLEAN DEFAULT true,
  email_verified BOOLEAN DEFAULT false,
  created_at TIMESTAMP DEFAULT NOW(),
  updated_at TIMESTAMP DEFAULT NOW()
);
```

```
-- Table des sessions/tokens
CREATE TABLE user_sessions (
  id SERIAL PRIMARY KEY,
  user_id INTEGER REFERENCES
  users(id),
  token_hash VARCHAR(255),
  expires_at TIMESTAMP,
  created_at TIMESTAMP DEFAULT
  NOW()
);
```

```
-- Table des permissions
(optionnel, pour plus de
flexibilité)
CREATE TABLE permissions (
  id SERIAL PRIMARY KEY,
  name VARCHAR(100) UNIQUE,
  description TEXT
);
```

```
CREATE TABLE role_permissions (
  role VARCHAR(50),
  permission_id INTEGER
  REFERENCES permissions(id),
  PRIMARY KEY (role,
  permission_id)
);
```



Module 4 : Processus d'achat avec BDD (40 min)

```
CREATE TABLE orders (  
  id SERIAL PRIMARY KEY,  
  user_id INTEGER REFERENCES users(id),  
  order_number VARCHAR(50) UNIQUE,  
  status ENUM('pending', 'confirmed',  
  'shipped', 'delivered', 'cancelled') DEFAULT  
  'pending',  
  total_amount DECIMAL(10,2),  
  shipping_address TEXT,  
  billing_address TEXT,  
  payment_method VARCHAR(50),  
  payment_status ENUM('pending', 'paid',  
  'failed', 'refunded') DEFAULT 'pending',  
  created_at TIMESTAMP DEFAULT NOW(),  
  updated_at TIMESTAMP DEFAULT NOW()  
);
```

```
CREATE TABLE order_items (  
  id SERIAL PRIMARY KEY,  
  order_id INTEGER REFERENCES orders(id),  
  product_variant_id INTEGER REFERENCES  
  product_variants(id),  
  quantity INTEGER NOT NULL,  
  unit_price DECIMAL(10,2),  
  total_price DECIMAL(10,2),  
  created_at TIMESTAMP DEFAULT NOW()  
);
```

```
CREATE TABLE shopping_carts (  
  id SERIAL PRIMARY KEY,  
  user_id INTEGER REFERENCES  
  users(id),  
  product_variant_id INTEGER  
  REFERENCES product_variants(id),  
  quantity INTEGER NOT NULL,  
  created_at TIMESTAMP DEFAULT  
  NOW(),  
  updated_at TIMESTAMP DEFAULT  
  NOW(),  
  UNIQUE(user_id,  
  product_variant_id)  
);
```



Module 4 : Processus d'achat avec BDD (40 min)

```
CREATE TABLE orders (  
  id SERIAL PRIMARY KEY,  
  user_id INTEGER REFERENCES users(id),  
  order_number VARCHAR(50) UNIQUE,  
  status ENUM('pending', 'confirmed',  
  'shipped', 'delivered', 'cancelled') DEFAULT  
  'pending',  
  total_amount DECIMAL(10,2),  
  shipping_address TEXT,  
  billing_address TEXT,  
  payment_method VARCHAR(50),  
  payment_status ENUM('pending', 'paid',  
  'failed', 'refunded') DEFAULT 'pending',  
  created_at TIMESTAMP DEFAULT NOW(),  
  updated_at TIMESTAMP DEFAULT NOW()  
);
```

```
CREATE TABLE order_items (  
  id SERIAL PRIMARY KEY,  
  order_id INTEGER REFERENCES orders(id),  
  product_variant_id INTEGER REFERENCES  
  product_variants(id),  
  quantity INTEGER NOT NULL,  
  unit_price DECIMAL(10,2),  
  total_price DECIMAL(10,2),  
  created_at TIMESTAMP DEFAULT NOW()  
);
```

```
CREATE TABLE shopping_carts (  
  id SERIAL PRIMARY KEY,  
  user_id INTEGER REFERENCES  
  users(id),  
  product_variant_id INTEGER  
  REFERENCES product_variants(id),  
  quantity INTEGER NOT NULL,  
  created_at TIMESTAMP DEFAULT  
  NOW(),  
  updated_at TIMESTAMP DEFAULT  
  NOW(),  
  UNIQUE(user_id,  
  product_variant_id)  
);
```




Module 5: RGPD

Obligations légales obligatoires

- Consentement explicite : cases à cocher NON pré-cochées
- Information claire sur le but de chaque donnée collectée
- Base légale définie : contrat, intérêt légitime ou consentement
- Minimisation des données : collecter uniquement le nécessaire
- Politique de confidentialité accessible en 2 clics maximum
- Mentions légales complètes (SIRET, adresse, contact DPO)
- Information au moment de la collecte sur les formulaires
- Durée de conservation clairement indiquée



Module 5: RGPD

Droits des utilisateurs à implémenter

- Droit d'accès : export des données personnelles
- Droit de rectification : modification des informations
- Droit à l'effacement : suppression définitive du compte
- Droit d'opposition : refus du traitement
- Droit à la portabilité : export en format exploitable
- Droit de limitation : gel temporaire des données
- Droit d'opposition marketing : opt-out publicité
- Formulaire de contact DPO obligatoire
- Endpoints API pour exercer les droits
- Délai de réponse : 1 mois maximum



Module 5: RGPD

Implémentation technique obligatoire

- Bannière de consentement granulaire par finalité
- Consentement préalable pour cookies non essentiels
- Révocation facile du consentement
- Durée des cookies : 13 mois maximum
- Chiffrement des données sensibles (mots de passe bcrypt)
- Accès limité avec authentification forte
- Sauvegarde sécurisée avec chiffrement • Logs d'accès et de modifications
- Registre des traitements documenté



Module 5: RGPD

Cookies e-commerce

- Essentiels (panier, session, sécurité) : PAS de consentement requis
- Analytiques (Google Analytics) : consentement REQUIS
- Marketing (Facebook Pixel, AdWords) : consentement REQUIS
- Personnalisation (recommandations) : consentement REQUIS



Module 5: RGPD

Documents obligatoires

- Politique de confidentialité avec identité du responsable
- Mentions légales avec SIRET et adresse hébergeur
- CGV/CGU avec clauses protection des données
- Bannière cookies avec choix granulaires

Sanctions CNIL

- Amendes jusqu'à 4% du CA annuel mondial
- Maximum 20 millions d'euros
- Mise en demeure publique • Impact sur l'image de marque



Module 5: RGPD

Actions immédiates projet

- Rédiger politique de confidentialité complète
- Implémenter bannière cookies conforme
- Créer page "Mes données personnelles"
- Hacher mots de passe avec bcrypt
- Ajouter mentions légales SIRET + hébergeur
- Formulaire contact DPO
- Export données utilisateur
- Fonction suppression compte
- Logs de sécurité



Module 5: RGPD

Erreurs à éviter absolument

- Cases pré-cochées pour le consentement
- Consentement global pour tous les cookies
- Conservation illimitée des données
- Pas de politique de confidentialité
- Transferts hors UE sans protection
- Absence de contact DPO
- Mots de passe en clair



Module 6: Hébergement

Architecture complète gratuite :

