

# Introduction au projet

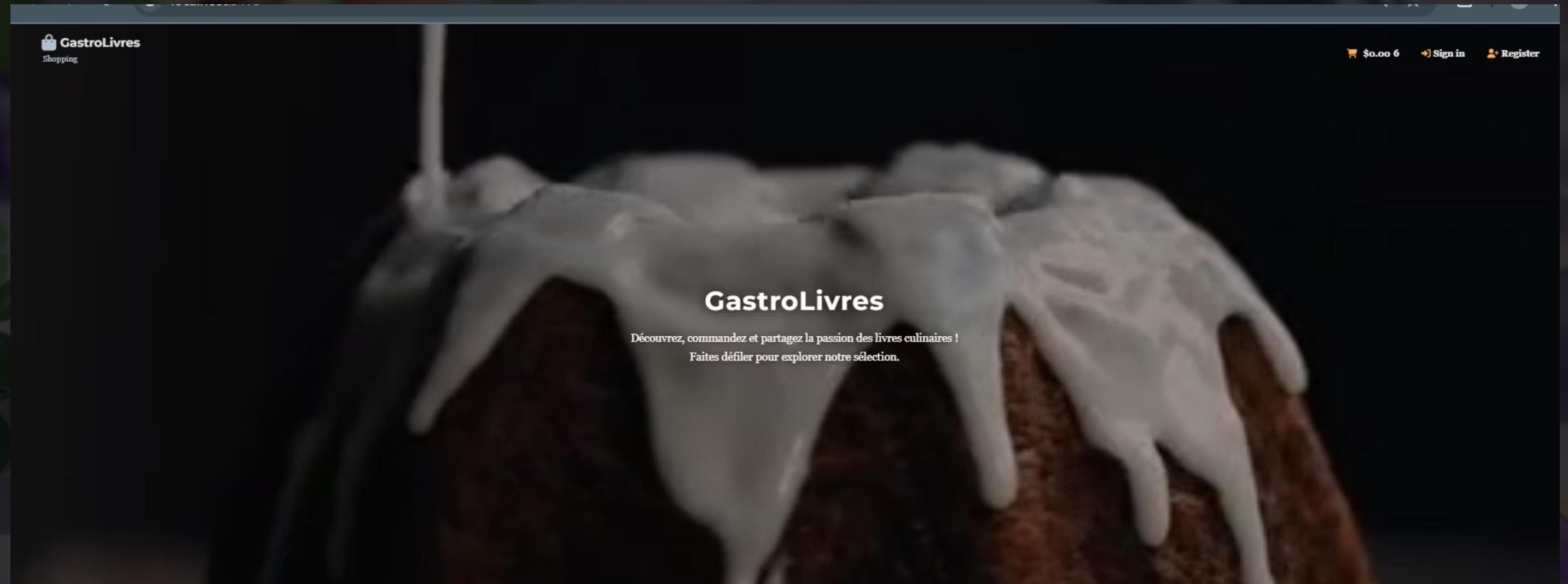
Le projet **GastroLivres** s'inscrit dans le cadre du cours *Programmation Web Avancée* et consiste à concevoir une plateforme de commerce électronique spécialisée dans la vente de livres de cuisine. À la croisée d'un site vitrine et d'une boutique en ligne, l'application vise à offrir une expérience riche à différents profils d'utilisateurs : amateurs qui débutent en pâtisserie, familles en quête de menus équilibrés, chefs confirmés ou passionnés de gastronomie.

L'objectif principal est de couvrir l'intégralité du cycle d'achat autour du livre culinaire : découverte du catalogue, consultation de fiches détaillées, gestion du panier, passage de commande, paiement sécurisé via un prestataire externe, consultation des avis et interactions sociales (listes cadeaux, recommandations). Au-delà de la partie front-office pour le visiteur, le site prévoit également des interfaces destinées aux acteurs métiers (éditeurs, gestionnaires, administrateurs) afin de piloter le catalogue et les contenus.

Ce travail est réalisé par :

Hicham El mallouki

Imad Abahmane



# Objectifs et Architecture Technique

L'objectif principal est de couvrir l'intégralité du cycle d'achat autour du livre culinaire : découverte du catalogue, consultation de fiches détaillées, gestion du panier, passage de commande, paiement sécurisé via un prestataire externe, consultation des avis et interactions sociales (listes cadeaux, recommandations). Au-delà de la partie front-office pour le visiteur, le site prévoit également des interfaces destinées aux acteurs métiers (éditeurs, gestionnaires, administrateurs) afin de piloter le catalogue et les contenus.

Techniquement, le projet repose sur une architecture moderne composée d'un frontend en React/Vite, d'une API backend en Node.js/Express, d'une base de données MySQL et d'une intégration Stripe pour le paiement. Cette stack permet de mettre en œuvre les notions avancées abordées dans le cours : conception d'API REST sécurisée, gestion des rôles, manipulation d'un store global côté front, transactions SQL, intégration d'un service externe, ainsi que la mise en place de bonnes pratiques d'expérience utilisateur (UX) et d'accessibilité.

# Présentation de l'entreprise et du projet

## 2.1 Présentation de l'entreprise

**GastroLivres Inc.** est une jeune entreprise québécoise qui souhaite démocratiser la cuisine gourmande à travers un catalogue de livres spécialisés. L'entreprise s'adresse autant aux passionnés qu'aux familles en quête d'idées, et compte sur une présence en ligne forte pour faire connaître ses auteurs partenaires (chefs, nutritionnistes, boulanger). Sa vision : devenir la référence francophone des ouvrages culinaires et proposer un accompagnement personnalisé aux lecteurs.

## 2.2 Présentation du projet de site web

Le site **livresgourmands.net** doit offrir une boutique en ligne complète autour du livre de cuisine :

- Parcourir le catalogue par catégories, popularité ou nouveauté.
- Consulter une fiche complète (visuel, résumé, auteur, prix, stock, avis).
- Gérer un panier, confirmer une commande, effectuer un paiement sécurisé.
- Permettre aux utilisateurs authentifiés de suivre leurs commandes, déposer un avis, créer des listes cadeaux.
- Fournir aux éditeurs/gestionnaires un accès pour mettre à jour le catalogue, gérer les stocks et modérer les commentaires.

# Spécifications Techniques Détaillées

## Frontend

React 18 + Vite, React Router, Context API pour le panier, Axios pour consommer l'API.

## Backend

Node.js 22 / Express, authentification JWT, bcrypt, gestion de la configuration via [.env](#).

## Base de données

MySQL (tables utilisateurs, ouvrages, commandes, avis, listes cadeaux), initialisation automatique à partir d'un jeu de données mock.

## Paiement

Intégration Stripe (clé publique/privée), endpoint [/api/checkout](#) pour décrémenter le stock et préparer la session de paiement.

## Sécurité

Validation des entrées, rôles utilisateurs (client, éditeur, gestionnaire, admin), mots de passe hachés, aucune donnée bancaire stockée, HTTPS prévu en production.

## Outils & CI/CD

npm, Nodemon pour le dev, scripts npm run dev/build, déploiement ciblé sur un hébergeur PaaS (Vercel/Netlify pour le front, Render/Railway pour l'API).

# Périmètre Fonctionnel et Contraintes

## Périmètre fonctionnel

Consultation du catalogue, fiches détaillées, ajout/retrait d'articles au panier, simulation de commande, suivi via notifications contextuelles.

## Contraintes techniques

Front-end React 18 (Vite), back-end Node/Express avec MySQL, API REST autohébergée, proxy Vite vers <http://localhost:3000>, compatibilité Chrome/Edge, prise en charge responsive.

## Contraintes organisationnelles

Équipe étudiante (2 personnes), planning maximum 6 semaines, livrables : dépôt Git, documentation de conception, notice d'installation, revue finale.

## Budget

Aucun budget financier engagé (0 €) : ressources matérielles et logicielles issues d'outils gratuits ou déjà disponibles.

## Indicateurs de réussite

Application déployable localement sans erreur, cohérence catalogue/BD, UX fluide (temps de chargement < 2 s sur catalogue), documentation complète et à jour.

# Arborescence du Site

Accueil

Page d'entrée et hub principal

Catalogues

Nouveaux livres et tous les livres

Vente & Paiement

Promotions, panier et paiement

Support & Compte

Pages d'info et espace utilisateur



# Arborescence du Site

Accueil |— Nouveaux livres |— Tous les livres | — Fiche produit (dynamique, accessible depuis tout lien "Voir le détail") |— Promotions & Coups de cœur (Carousel) |— Simulation de paiement | — Panier | — Informations de livraison | — Confirmation (récapitulatif + statut) |— Pages de contexte | — À propos / Bienvenue | — Contact | — FAQ | — Mentions légales | — Conditions générales | — Politique de livraison |— Espace utilisateur (non libre-service, accès via boutons dédiés) |— Inscription |— Connexion

# Wireframe

- Accueil : organisation des zones (header fixe avec navigation, hero visuel, carousel central, grille produits, bandeau newsletter, footer).
- Fiche produit : layout en deux colonnes (visuel, infos détaillées), section avis, CTA panier.
- Panier & paiement : étapes séquencées (cart → livraison → récap), mise en avant du total et des actions.
- Mobile : adaptation (menu burger, carousel tactile, panneaux pliables pour infos).
- Composants communs : header responsive, cartes produit, modals (confirmation panier, toast).

# Planche de tendance

- Palette gourmande : rouges framboise, ocres miel, neutres crème, touches vert basilic.
- Textures & visuels : photographies macro de recettes, motifs nappe, icônes illustrés.
- Ambiance : style convivial, artisanal chic, rappel d'un univers pâtisserie/épicerie fine.
- Inspirations citées : références de sites culinaires premium, magazines gastronomiques, packaging de tables gourmandes.
- Moodboard numérique : captures, codes couleur, typographies, textures.

# Charte graphique

- **6.1 Logotype**
  - Nom “GastroLivres” ou “Livres Gourmands”, version horizontale + version icône.
  - Construction : pictogramme sac + livre stylisé, marges de protection, déclinaisons monochromes.
  - Usages : tailles minimales, arrière-plans autorisés, interdits (distorsion, superposition).

# Couleurs

- Palette primaire :
  - Rouge Framboise #C0392B (actions),
  - Crème Vanille #FAF3E5 (fond).
- Palette secondaire :
  - Vert Sauge #4E6C50,
  - Brun Pain d'épices #8B5E3C.
- Neutres : #111927, #E3E6ED.
- Ratios d'utilisation, contrastes AA; exemples (boutons, fond, textes).

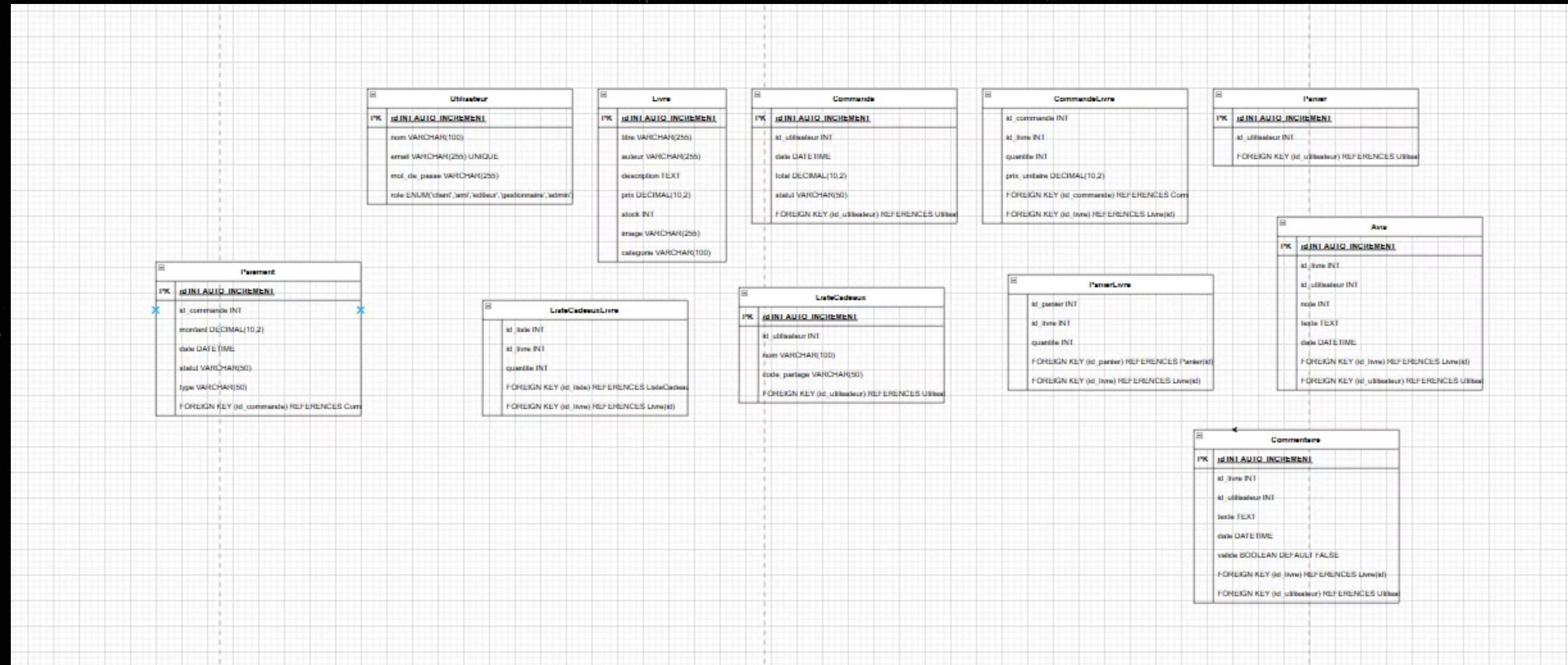
# Boutons et pictos

- Boutons primaires : fond rouge, texte crème, arrondi 8 px, hover plus sombre, focus outline accessible.
- Boutons secondaires : contour vert/texte vert, fond transparent.
- Iconographie : pack Font Awesome, usage cohérent (panier, usager, livraison), taille 18–24 px.
- États : normal, survol, désactivé avec changement d'opacité; guidelines pour pictos (alignement, marges)

# Typographies

- Titres : “Playfair Display” (sérif gourmet), tailles H1 48 px → H3 24 px.
- Corps : “Lato” ou “Nunito”, corps 16 px, interlignage 1.6.
- Titres secondaires / boutons : uppercase léger avec tracking.
- Bonnes pratiques d’usage (hiérarchie, italique, accentuation).

# Modèle de données / Schéma relationnel



# Intégration MySQL



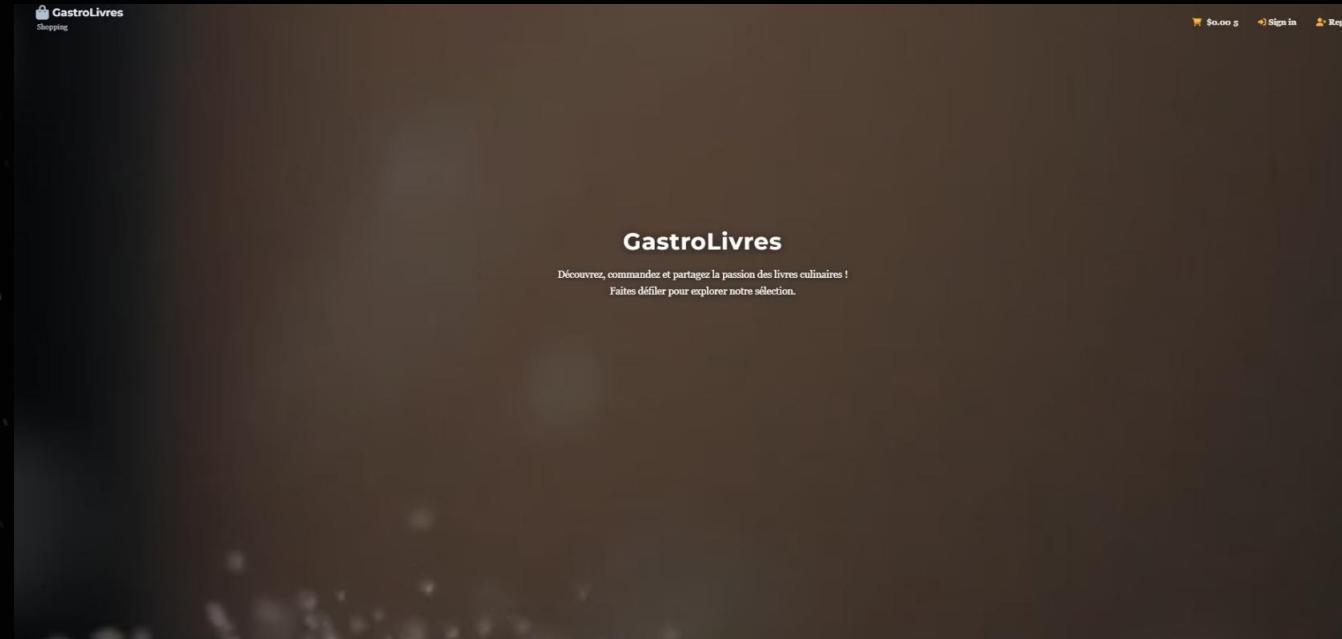
- **Rôle :** MySQL sert de socle persistant pour le catalogue, les utilisateurs, les commandes et les stocks.
- **Accès applicatif :** l'API Node/Express (backend/src/db.js) utilise `mysql2/promise` pour établir un pool de connexions via les variables d'environnement et exécuter requêtes paramétrées.
- **Workflow front → back → base :**
  - Le front (React/Vite) consomme l'API REST via `frontend/src/services/api.js`.
  - L'API interroge MySQL (liste des ouvrages, lecture fiche, décrément du stock pendant le checkout).
  - Les modifications (ex. PUT `/api/ouvrages/:id` ou POST `/api/checkout`) s'écrivent directement dans la base.
- **Schéma & relations :** rappelle le modèle de données joint (tables `Livre`, `Utilisateur`, `Commande`, etc.) pour prouver la cohérence avec les fonctionnalités.
- **Contraintes & bonnes pratiques :**
  - Connexions sécurisées par `.env` (`DB_HOST`, `DB_USER`, `DB_PASSWORD`, `DB_NAME`).
  - Transactions utilisées pour les opérations critiques (`/api/checkout`) afin de maintenir l'intégrité du stock.
  - Seed initial depuis `frontend/mock/db.json` ou script SQL mentionné dans les annexes.
  - Sauvegardes recommandées (dump régulier) et distinction environnement dev/prod.

# Parcours Utilisateurs : Consultation et Ajout au Panier

1

## 1. Consultation catalogue

Arrivée sur Accueil via navigation principale.



Sign in :



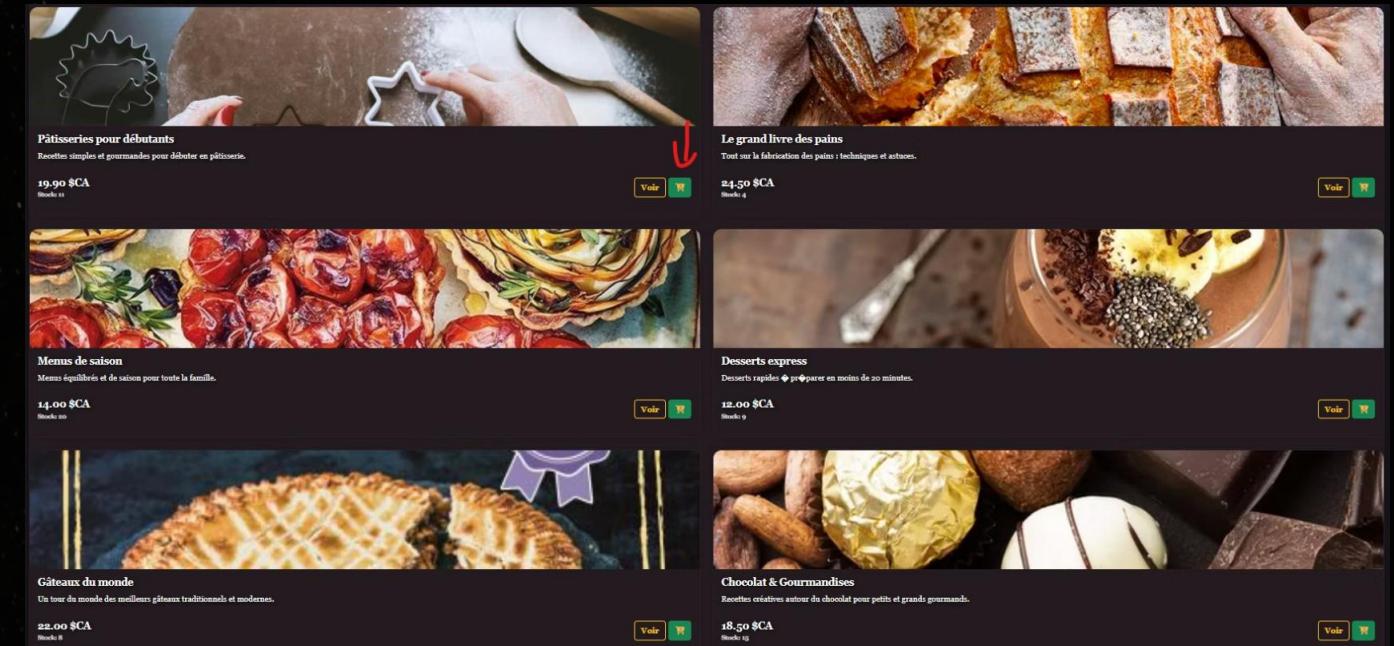
2

## 2. Ajout au panier

Depuis la fiche ou la carte produit, clic sur "Ajouter au panier".

Toast de confirmation + mise à jour de l'icône panier dans l'en-tête.

Consultation du panier via l'entête, ajustement des quantités ou suppression d'un article.



## Panier

14.00 \$CA - 3 + Menus de saison

19.90 \$CA - 2 + Pâtisseries pour débutants

Vider le panier

Sous-total

Résumé

81.80 \$CA

Carte bancaire  
Visa / Mastercard

Apple Pay / Google Pay  
Ma Banque

PayPal  
Bénéfice disponible

Payer (81.80 \$CA)

# Parcours Utilisateurs : Simulation de Paiement

01

Depuis le panier

Bouton “Passer au paiement”.

02

Saisie des informations

Coordonnées, choix de livraison dans  
`CarteCredit.jsx`.

03

Validation

API `/api/checkout` simule le débit, décrémente  
le stock en base, affiche page de confirmation  
avec récapitulatif.

## VI. Contraintes non fonctionnelles

- **Performance** : temps de chargement initial < 2 s (desktop) ; bundle < 500 KB ; API répond < 500 ms ; pagination si > 30 ouvrages.
- **Sécurité** : aucune donnée sensible stockée côté front ; accès API limité aux routes nécessaires ; mots de passe chiffrés si fonctionnalité activée ; flux Stripe isolé avec clés d'essai.
- **Accessibilité** : contraste vérifié (WCAG AA), navigation clavier sur menu/panier, alt text obligatoire, messages d'erreur lisibles.
- **Compatibilité** : Chrome/Edge/Firefox récents, responsive ≥ 375 px ; dégradé pour Safari (test manuel).
- **Maintenabilité** : code React modulaire, ESLint + Prettier, conventions de commits, documentation API dans README backend.

# Plan de Tests et Validation & Planification

## VII. Plan de tests & validation

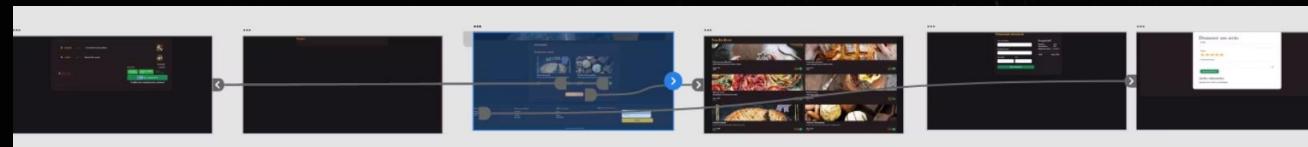
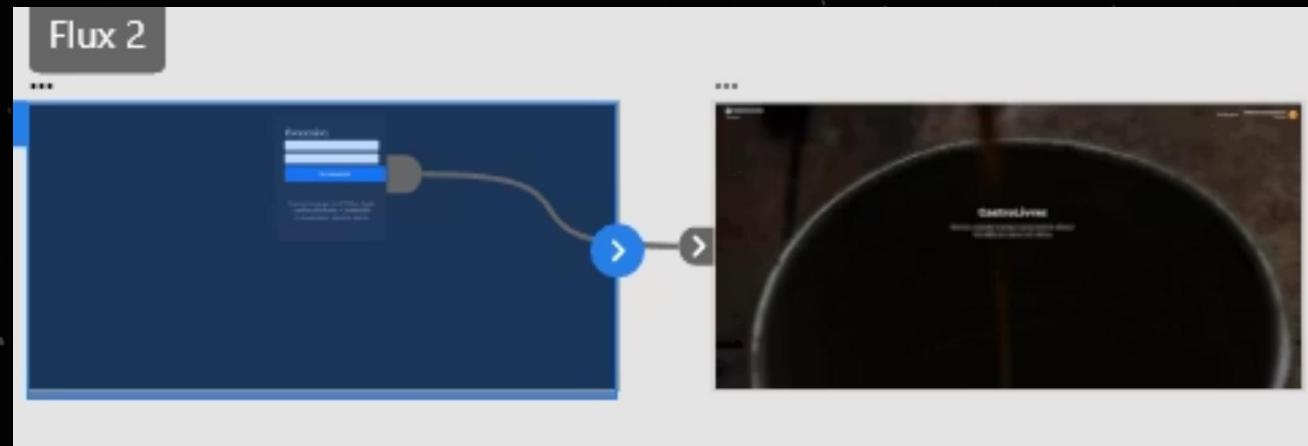
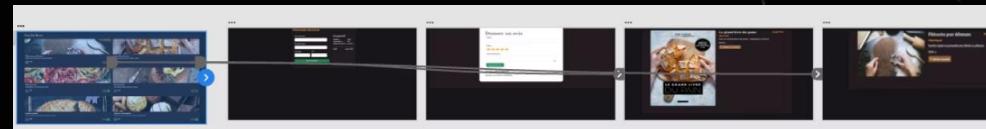
- **Objectifs** : vérifier fonctionnalités cœur (catalogue, panier, checkout factice), robustesse des toasts, cohérence stock.
- **Organisation** : tests unitaires (React hooks, services API), tests d'intégration (checkout API), tests manuels scénarisés (parcours utilisateur).
- **Scénarios principaux** : ajout panier, modification quantité, checkout avec stock limite, reprise après rafraîchissement, navigation mobile.

## VIII. Planification & ressources

- **Planning** : Semaine 1 cadrage + maquettes ; Semaine 2-3 développement front ; Semaine 3-4 backend + intégration ; Semaine 5 tests & ajustements ; Semaine 6 documentation finale.
- **Ressources** : 2 développeurs polyvalents, 1 référent qualité (tests/docs).
- **Jalons** : Prototype interactif (fin S2), première démo intégrée (fin S4), revue finale (fin S6).

# Annexes et Maquettes Figma

Maquettes ADOBE(XD):



X. Maquettes Figma  
(desktop/mobile)

X.1 Lien d'accès :

<https://www.figma.com/design/tfheh24BKDfGdatBlmPbS5/Projet-web?node-id=0-1&p=f&t=CliFsULUQFFK1KSV-0>

## IX. Annexes

- Document ADOBE XD
- lien Figma

Lien github :

