**📑 Cahier des Charges – WebGestion**

**1. Présentation générale**

**Nom du projet** : WebGestion  
**Type d’application** : Application web de facturation et de gestion commerciale  
**Architecture** : Client léger (développé en Laravel)  
**Public cible** : Artisans, indépendants, petites entreprises (TPE/PME)

**Contexte** :  
La réforme de la **facturation électronique obligatoire** (2026-2027) incite les entreprises à adopter des solutions numériques accessibles partout.  
WebGestion vise à offrir une alternative en ligne au logiciel WinGestion (client lourd), afin de permettre aux utilisateurs de gérer leurs factures, devis, paiements, stocks et documents directement depuis un navigateur, sans installation locale.

**2. Objectifs du projet**

* Proposer une solution de facturation **accessible en ligne** depuis tout appareil (PC, tablette, smartphone).
* Offrir les mêmes fonctionnalités que WinGestion (facturation, suivi des paiements, gestion des stocks, statistiques, comptabilité simplifiée).
* Faciliter la collaboration (multi-utilisateurs, gestion des rôles).
* Garantir la **sécurité des données** (authentification, sauvegardes).
* Préparer l’intégration avec le **Portail Public de Facturation (PPF)**.

**3. Fonctionnalités principales**

**3.1 Gestion des utilisateurs**

* Authentification sécurisée (login/mot de passe).
* Gestion des rôles et droits d’accès (administrateur, utilisateur).

**3.2 Gestion des clients**

* CRUD (ajout, modification, suppression, consultation).
* Recherche et filtres (nom, société, email).
* Import de documents propres à chaque client (GED).
* Export CSV / PDF.

**3.3 Produits & services**

* Fiches produit (désignation, prix HT, TVA, stock, code-barres).
* Organisation par catégories.

**3.4 Gestion des stocks**

* Approvisionnements et retraits avec contrôle.
* Historique des mouvements.
* Inventaires.
* Récapitulatif imprimable (valeur du stock, ventes par catégorie).

**3.5 Facturation**

* Création de devis, commandes, factures, factures d’acompte, avoirs, bons de livraison.
* Transformation automatique (devis → commande → facture).
* Numérotation automatique ou personnalisée.
* Calculs HT / TVA / TTC automatiques.
* Gestion des tranches (bâtiment).
* Suivi des statuts (brouillon, validée, refusée, payée).
* Export PDF (logo, mentions légales).
* Export vers la comptabilité.

**3.6 Suivi des paiements**

* Enregistrement des règlements (espèces, virement, CB, chèque).
* Multi-règlements possibles sur un même document (tranches).

**3.7 Comptabilité**

* Export des factures en Excel.
* Suivi débit/crédit.

**3.8 Statistiques**

* Indicateurs clés : chiffre d’affaires, factures payées/impayées.
* Suivi des ventes par produit.
* Suivi des devis acceptés/refusés.

**3.9 Paramètres**

* Informations de l’entreprise (raison sociale, SIRET, adresse, TVA).
* Logo et mentions légales personnalisables.
* Gestion multi-sociétés.
* Paramètres email (envoi automatique des documents).
* Activation/désactivation de fonctionnalités (stock, tranches, bons de livraison, multi-sociétés…).
* Gestion des utilisateurs par l’administrateur.
* Ajout/Modification/Suppression sur les taux de TVA, modes de règlements, répertoire pour l’export des document (comptabilité), civilités/formes juridique des clients.

**4. Contraintes techniques**

* **Framework** : Laravel (PHP)
* **Base de données** : MySQL / MariaDB
* **Serveur web** : Apache ou Nginx
* **Front-end** : Blade + Bootstrap/Tailwind
* **Hébergement** : serveur Linux (mutualisé ou dédié)
* **Compatibilité** : navigateurs modernes (Chrome, Edge, Firefox)

**5. Sécurité**

* Hashage des mots de passe (bcrypt).
* Middleware d’authentification et gestion des sessions.
* Rôles et permissions par utilisateur.
* Sauvegardes automatiques de la base de données.
* Connexion sécurisée en HTTPS.

**6. Livrables attendus**

* Application WebGestion (code source + base de données).
* Manuel utilisateur.
* Jeu d’essai (données fictives).
* Cahier de tests.
* Documentation technique (MCD/MLD, schéma applicatif).

**7. Planning prévisionnel**

1. Analyse des besoins – 2 semaines
2. Conception (MCD, interfaces, routes Laravel) – 3 semaines
3. Développement – 7 semaines
4. Tests et corrections – 2 semaines
5. Livraison & Documentation – 1 semaine

**Durée totale estimée : 15 semaines**

**8. Budget prévisionnel**

* Développement interne.
* Hébergement serveur (mutualisé ou cloud).
* Coûts liés au temps de développement + maintenance.

**9. Perspectives d’évolution**

* Intégration avec le **Portail Public de Facturation (PPF)**.
* Application mobile (consultation et création rapide de devis/factures).
* API REST pour interconnexion avec d’autres systèmes (ex. comptabilité externe, e-commerce).
* Ajout d’un module de gestion avancée des stocks (alertes sur seuils bas).