

# Cahier des Charges

## WebGestion

Application web de facturation et de gestion commerciale

Initiative personnelle — BTS SIO SLAM — E6

Complément au projet entreprise WinGestion

### 1. Présentation générale

**Nom du projet** : WebGestion

**Type** : Application web de facturation et de gestion commerciale

**Architecture** : Client léger (Laravel, PHP/MySQL)

**Public cible** : Artisans, indépendants, TPE/PME

#### Contexte et démarche

WebGestion est un projet personnel développé en dehors du cadre de l'entreprise, par initiative propre, dans le but de compléter les compétences acquises lors du développement de WinGestion.

En effet, le projet confié en entreprise — WinGestion — étant exclusivement une application cliente lourde développée en WinDev, il ne couvrait pas la compétence de développement web (client léger). Afin de démontrer une maîtrise complète des architectures logicielles au programme du BTS SIO SLAM, j'ai choisi de concevoir WebGestion de façon autonome.

*WebGestion est la transposition web de WinGestion : même logique métier, même public cible, même contexte réglementaire — mais développé avec une stack web moderne (Laravel, PHP, MySQL), accessible depuis n'importe quel navigateur sans installation. Cette démarche proactive illustre une capacité à s'autoformer, à prendre des initiatives et à porter une solution sur deux architectures distinctes.*

La réforme de la facturation électronique obligatoire (2026-2027) renforce la pertinence d'une solution web accessible en ligne, compatible avec les futurs standards numériques.

### 2. Objectifs du projet

- Proposer une solution de facturation en ligne, accessible sur PC, tablette et smartphone.
- Gérer les clients, produits, devis, factures et paiements via une interface web.
- Faciliter la collaboration multi-utilisateurs grâce à une architecture centralisée.
- Sécuriser les données (authentification, rôles, sauvegardes, HTTPS).

- Acquérir et démontrer une maîtrise du développement web fullstack (Laravel, PHP, MySQL).
- Exposer les données via une API REST pour permettre des intégrations futures.
- Préparer l'intégration future avec le Portail Public de Facturation (PPF).

## 3. Fonctionnalités principales

---

### 3.1 Gestion des utilisateurs

- Authentification sécurisée (login / mot de passe hashé via bcrypt).
- Rôles et permissions : administrateur, utilisateur simple.
- Middleware d'authentification Laravel (protection des routes).

### 3.2 Gestion des clients

- Création, modification, suppression, consultation des fiches clients.
- **Distinction du type de client : Particulier / Professionnel / Entreprise, conditionnant l'accès à certaines fonctionnalités.**
- Recherche et filtrage.
- Export PDF de la fiche client.

### 3.3 Produits & services

- Fiche produit : désignation, prix HT, TVA, stock simplifié.
- Organisation par catégories.
- Alerte de stock faible visible depuis le tableau de bord.

### 3.4 Facturation

- Création de devis, factures et avoirs.
- Numérotation automatique (ex. F-2026-001, D-2026-001...).
- Calculs automatiques HT / TVA / TTC.
- Export PDF personnalisé (logo, mentions légales).
- Suivi des statuts : Brouillon, Envoyé, Payé.

### 3.5 Suivi des paiements

- Enregistrement des règlements (CB, virement, espèces, chèque).
- Multi-règlements possibles sur une même facture.

### 3.6 Export comptable

Module d'export simplifié, destiné à être utilisé en complément d'un logiciel comptable tiers.

- **Export au format CSV des factures et avoirs.**
- **Export au format CSV des règlements.**

### 3.7 Tableau de bord (Page d'accueil)

La page d'accueil constitue le point d'entrée de l'application. Elle présente une synthèse visuelle de l'activité en cours, sans être un module statistique à part entière.

Indicateurs affichés en temps réel :

- Chiffre d'affaires du mois en cours.
- Nombre de devis en attente.
- Total des impayés.
- Nombre de nouveaux clients du mois.

Raccourcis vers les modules principaux (avec compteurs) :

- Clients, Produits, Devis, Factures, Avoirs, Règlements, Paramètres.

Sections complémentaires :

- Tableau des dernières factures créées (code, client, date, montant TTC, statut).
- Alertes et rappels : produits en stock faible, avec lien vers la liste complète.

### 3.8 API REST

WebGestion expose une API REST permettant l'accès aux données de facturation et aux statistiques. Cette API est destinée à des intégrations futures (application mobile, synchronisation avec WinGestion, e-commerce, etc.).

Authentification :

**POST** /api/register → Création d'un compte utilisateur

**POST** /api/login → Authentification — retourne un token Sanctum

**POST** /api/logout → Déconnexion (token révoqué) — nécessite auth:sanctum

Documents (factures, devis, avoirs) :

**GET** /api/documents → Liste des documents filtrables (societe\_id, client\_id, type, search)

**GET** /api/documents/{id} → Détail d'un document avec ses lignes

*Note : l'authentification Sanctum sur ces routes est temporairement désactivée en phase de développement/test. La sécurisation sera rétablie avant la mise en production.*

Statistiques de ventes (protégées par auth:sanctum) :

**GET** /api/stat-vente/produit/{id} → Ventes globales d'un produit (toutes périodes)

**GET** /api/stat-vente/periode/{annee}/{mois} → Ventes globales sur une période donnée

**GET** /api/stat-vente/produit/{id}/{annee}/{mois} → Ventes d'un produit sur un mois précis

### 3.9 Paramètres

- Informations de l'entreprise (raison sociale, SIRET, adresse, TVA).
- Personnalisation : logo, mentions légales.

- Gestion multi-sociétés (optionnelle).

## 4. Contraintes techniques

---

**Framework back-end** : Laravel (PHP)

**Base de données** : MySQL / MariaDB

**Serveur web** : Apache / Nginx

**Front-end** : Blade + Bootstrap / Tailwind CSS

**Authentification API** : Laravel Sanctum (tokens)

**Compatibilité navigateurs** : Chrome, Firefox, Edge

### Comparaison architecturale avec WinGestion

WebGestion et WinGestion partagent la même logique métier, mais diffèrent fondamentalement dans leur architecture :

**WinGestion** → Client lourd (WinDev / HyperFileSQL / Windows uniquement)

**WebGestion** → Client léger (Laravel / MySQL / tout navigateur, tout OS)

Cette dualité démontre la capacité à adapter une solution métier à des contraintes techniques différentes.

## 5. Sécurité

---

- Hashage des mots de passe (bcrypt via Laravel).
- Middleware d'authentification et gestion des permissions par rôle.
- Authentification API via tokens Laravel Sanctum (register / login / logout).
- Protection CSRF intégrée (Laravel).
- HTTPS obligatoire en production.
- Sauvegardes automatiques de la base de données.
- Conformité RGPD : données personnelles stockées sur serveur privé.

## 6. Livrables attendus

---

- Application WebGestion (code source Laravel + base de données MySQL).
- Documentation utilisateur.
- Données de test (jeu d'essai).
- Schémas techniques (MCD/MLD, routes Laravel).

## 7. Planning prévisionnel

---

Durée estimée : 20 semaines — projet développé de façon autonome en parallèle du BTS.

**Phase 1** — Analyse des besoins → **3 semaines**

**Phase 2** — Conception (MCD, interfaces, routes Laravel) → **4 semaines**

**Phase 3** — Développement (dont API REST) → **9 semaines**

**Phase 4** — Tests et corrections → **3 semaines**

**Phase 5** — Livraison et documentation → **1 semaine**

**Durée totale estimée : 20 semaines**

## 8. Perspectives d'évolution

---

- Connexion avec le Portail Public de Facturation (PPF) — dans le cadre de la réforme 2026-2027.
- Export Factur-X BASIC (PDF/A-3 + XML) pour les clients de type Entreprise — en cours de développement.
- Autocomplétion d'adresse via l'API Adresse (adresse.data.gouv.fr) — cohérence avec WinGestion.
- Sécurisation complète de l'API REST (activation de Sanctum sur toutes les routes).
- Application mobile (PWA ou application native) s'appuyant sur l'API REST existante.
- Synchronisation bidirectionnelle avec WinGestion (partage de la base clients/produits).
- API REST étendue (e-commerce, comptabilité tierce).