# **Documentation Technique - Système de Gestion des Notes de Frais des Bénévoles**

## **Table des matières**

1. Introduction
2. Structure des fichiers
3. Documentation du fichier HTML (index.html)
4. Documentation du fichier CSS (styles.css)
5. Documentation du fichier JavaScript (script.js)
6. Documentation du fichier PHP de traitement (traitement.php)
7. Documentation de la page de confirmation (confirmation.php)
8. Installation et déploiement
9. Maintenance et évolution

## **Introduction**

Le système de gestion des notes de frais des bénévoles est une application web permettant aux bénévoles d'une association sportive de remplir, calculer et soumettre leurs notes de frais en ligne. Ce système remplace le processus papier tout en conservant la structure et les informations du bordereau original. L'application permet également de générer un fichier JSON des données saisies pour un éventuel traitement ultérieur.

## **Structure des fichiers**

L'application se compose de cinq fichiers principaux :

* index.html : Page principale contenant le formulaire de saisie
* styles.css : Fichier de styles CSS pour la mise en page et l'apparence
* script.js : Fichier JavaScript gérant les calculs dynamiques et les interactions
* traitement.php : Script PHP pour le traitement et l'enregistrement des données
* confirmation.php : Page de confirmation après soumission du formulaire

## **Documentation du fichier HTML**

### **Aperçu général**

Le fichier index.html constitue l'interface principale de l'application. Il s'agit d'un formulaire structuré qui reprend les champs du bordereau papier original, avec des fonctionnalités améliorées pour faciliter la saisie.

### **Structure**

* **En-tête** : Titre du formulaire et informations de l'association
* **Informations personnelles** : Identification du bénévole et adresse
* **Représentant légal** : Section pour ajouter des adhérents représentés
* **Tableau des frais** : Saisie détaillée des frais avec calculs automatiques
* **Section signature** : Champs pour la signature et la date
* **Partie administrative** : Zone réservée à l'association
* **Boutons d'action** : Soumettre, Imprimer, Réinitialiser

### **Éléments interactifs clés**

1. **Gestion des adhérents** : Interface pour ajouter/supprimer des adhérents représentés
2. **Tableau de frais dynamique** : Ajout/suppression de lignes et calculs automatiques
3. **Tarif kilométrique** : Définition du tarif appliqué avec répercussion automatique sur les calculs
4. **Bouton d'impression** : Fonctionnalité d'impression directe du formulaire

### **Validation**

Le formulaire intègre une validation native HTML5 avec des attributs required pour les champs obligatoires et des restrictions de type (number, date) pour garantir la cohérence des données.

## **Documentation du fichier CSS**

### **Objectifs**

Le fichier styles.css définit l'apparence visuelle de l'application, avec un focus sur la lisibilité, l'ergonomie et la responsivité. Il reproduit visuellement l'aspect d'un document administratif tout en offrant une expérience utilisateur moderne.

### **Organisation**

* **Reset et base** : Normalisation des styles par défaut
* **Structure du conteneur** : Mise en page générale et positionnement
* **Éléments de formulaire** : Personnalisation des champs et contrôles
* **Tableau de frais** : Mise en forme détaillée du tableau des lignes de frais
* **Sections spécifiques** : Styles particuliers pour chaque section du formulaire
* **Impression** : Styles adaptés pour l'impression du document
* **Responsivité** : Adaptation aux différentes tailles d'écran

### **Fonctionnalités CSS notables**

1. **Media queries** : Adaptation pour l'impression et les écrans mobiles
2. **Flexbox** : Utilisé pour l'alignement des champs et le positionnement
3. **Mise en évidence des zones** : Différenciation visuelle entre les sections utilisateur et administratives
4. **États des éléments** : Styles spécifiques pour les survols, focus et erreurs

## **Documentation du fichier JavaScript**

### **Fonctionnalités principales**

Le fichier script.js gère toutes les interactions dynamiques du formulaire, principalement les calculs automatiques et la gestion des lignes de frais.

### **Structure et initialisation**

Le code est encapsulé dans un événement DOMContentLoaded pour s'assurer que tous les éléments du DOM sont chargés avant l'exécution. Il initialise les références aux éléments clés du formulaire et configure l'année en cours par défaut.

### **Fonctions principales**

1. **calculateTrajetCost(km, tarif)** : Calcule le coût d'un trajet en fonction des kilomètres et du tarif
2. **calculateRowTotal(row)** : Calcule le total d'une ligne (trajet + frais annexes)
3. **updateTotals()** : Met à jour les totaux généraux du formulaire
4. **addNewRow()** : Ajoute une nouvelle ligne au tableau des frais
5. **attachRowEventListeners(row)** : Configure les écouteurs d'événements pour une ligne
6. **addNewAdherent()** : Ajoute un champ pour un nouvel adhérent représenté

### **Gestion des événements**

* Écoute des changements sur les champs numériques pour recalculer les totaux
* Gestion des clics sur les boutons d'ajout/suppression de lignes
* Impression du document via le bouton dédié
* Validation avant soumission pour s'assurer que tous les champs requis sont remplis

### **Particularités techniques**

* Utilisation de l'API standard du DOM pour les manipulations d'éléments
* Calculs en temps réel avec formatage des valeurs monétaires
* Gestion de la suppression conditionnelle (empêche la suppression de la dernière ligne)

## **Documentation du fichier PHP de traitement**

### **Rôle principal**

Le fichier traitement.php gère la récupération, la validation et l'enregistrement des données soumises par le formulaire. Il génère également une référence unique pour chaque note de frais.

### **Processus de traitement**

1. **Initialisation** : Démarrage de la session et définition des fonctions utilitaires
2. **Validation des données** : Nettoyage et validation des entrées utilisateur
3. **Structuration des données** : Organisation des données en structure hiérarchique
4. **Génération de référence** : Création d'un identifiant unique pour la note de frais
5. **Enregistrement** : Sauvegarde des données au format JSON
6. **Redirection** : Transfert vers la page de confirmation avec message de succès

### **Sécurité**

* Nettoyage des entrées via la fonction nettoyer() pour prévenir les injections
* Utilisation de htmlspecialchars() pour échapper les caractères spéciaux
* Validation des types de données (conversion explicite pour les valeurs numériques)

### **Stockage des données**

Les données sont enregistrées au format JSON dans un dossier dédié (notes\_frais), avec un nom de fichier basé sur la date/heure et le nom du bénévole.

### **Notes d'implémentation**

Le code inclut des commentaires pour une fonction de génération PDF, désactivée par défaut car nécessitant une bibliothèque externe (FPDF ou TCPDF).

## **Documentation de la page de confirmation**

### **Objectif**

Le fichier confirmation.php affiche un message de confirmation après le traitement réussi du formulaire, fournissant à l'utilisateur un retour visuel et la référence de sa note de frais.

### **Fonctionnement**

1. Récupération du message de confirmation depuis la session
2. Affichage formaté du message avec la référence unique
3. Proposition de liens pour revenir à l'accueil ou imprimer le document

### **Structure visuelle**

* Icône de succès
* Titre de confirmation
* Message détaillé avec la référence mise en évidence
* Boutons d'action pour les prochaines étapes

## **Installation et déploiement**

### **Prérequis**

* Serveur web avec support PHP 7.0+
* Droits d'écriture sur le dossier notes\_frais

### **Étapes d'installation**

1. Copier tous les fichiers dans un répertoire accessible depuis le serveur web
2. Créer le dossier notes\_frais avec les permissions d'écriture appropriées
3. Vérifier que le serveur est configuré pour exécuter les scripts PHP
4. Tester l'application en accédant à index.html via un navigateur

### **Bibliothèques optionnelles**

Pour activer la génération PDF, installer FPDF ou TCPDF et décommenter la section correspondante dans traitement.php.

## **Maintenance et évolution**

### **Maintenance régulière**

* Vérifier régulièrement l'espace disque utilisé par les fichiers JSON
* Mettre à jour le tarif kilométrique par défaut selon les changements réglementaires
* Sauvegarder régulièrement le dossier notes\_frais

### **Évolutions possibles**

1. **Authentification utilisateur** : Ajout d'un système de connexion pour les bénévoles
2. **Tableau de bord administrateur** : Interface pour visualiser et gérer les notes soumises
3. **Export multi-formats** : Ajout d'options d'export PDF, Excel, etc.
4. **Intégration BDD** : Passage du stockage JSON à une base de données
5. **Validation automatique** : Ajout d'un workflow de validation/approbation des notes