

1. Architecture Générale.....	2
2. Sécurisation des flux et des données.....	2
3. Enregistrements de commandes.....	2
Structure d'une commande.....	2
Création et validation des commandes.....	3
Sécurisation des justificatifs.....	7
Signature des transactions.....	7
4. Gestion des sauvegardes et des archives.....	9
Sauvegarde par l'utilisateur du logiciel d'encaissement.....	9
Sauvegardes réalisées en interne par l'éditeur du logiciel.....	10
5. Conservation des données.....	10
6. Mode hors ligne.....	10
7. Système centralisateur.....	11
8. Accès de l'administration fiscale aux données.....	11
9. Gestion des incidents.....	12
10. Purges.....	12
11. Versions de l'application.....	12
12. Périmètre fiscal.....	13
Conformité au Référentiel de Certification v1.7.....	14

1. Architecture Générale

Le logiciel caisse.enregistreuse.fr repose sur une architecture SaaS multi-instance hébergée sur des serveurs Linux (Ubuntu 22.04) chez Digital Ocean, en data centers européens.

Application Web : Développée en PHP 8.2.

Base de données : MySQL 8, configurée en mode transactionnel répliquée de base de données sur un serveur de répliquon et procédure de sauvegarde automatisée réalisant une backup complète de la base de données dans un fichier zip au format SQL.

Serveurs : Accès sécurisé uniquement par les administrateurs Net-Assembly via SSH.

2. Sécurisation des flux et des données

- Tous les échanges entre client et serveur se font en HTTPS/TLS.
- Les bases de données sont protégées par firewall applicatif.
- Les signatures de transactions sont calculées et stockées immédiatement après enregistrement.

3. Enregistrements de commandes

Structure d'une commande

La base de données contient différents champs pour une commande.

La date de création de la commande, la date de validation de la commande, la date de valeur de la commande, l'identifiant de l'établissement d'utilisation du système d'encaissement, l'identifiant de la caisse, l'identifiant de l'utilisateur, l'identifiant du mode de paiement utilisé en tant que premier mode de paiement, le prix initial, le montant total des réductions, le montant total des TVA, le prix final total toute taxe comprise, un titre court pour la commande résumant les principaux articles de la commande, le nombre d'articles de la commande, le montant total payer pour cette commande, le numéro interne du compte client attaché à la commande, un champ indiquant si la commande est terminée ou non, une note privée sur la commande, note publique pour la commande, une pièce d'identité du client ayant effectuer cette commande, un champ indiquant s'il s'agit d'un remboursement ou d'un achat, un identifiant interne, le numéro de commande, la méthode de livraison, la table sur lequel le client consomme, l'état de préparation de la commande, l'état de livraison de la commande, le nombre de couverts que constitue cette commande, le nombre de points de fidélité gagné avec cette commande, l'état de d'expédition de la commande l'état de

Page 2

Dossier d'architecture technique

facture envoyé de la commande, le sceau cryptographique de la commande, un numéro de bipper dans le cadre d'une gestion des bipeurs pour la préparation des commandes...

Chaque commande est constituée d'une liste d'articles chaque article disposant d'un prix initial, d'un prix final HT, d'un taux TVA, d'un montant de TVA, d'un taux de réduction si une réduction est appliquée ou d'un montant de réduction de réduction est appliquée en montant, d'un prix final, d'une quantité, d'un nom (libellé), d'une éventuelle image, d'un état de préparation d'un identifiant d'article.

A commande correspond une liste de règlement, la commande pouvant être payée en une seule ou plusieurs fois avec un seul mode de paiement ou plusieurs modes de paiement chaque paiement est enregistré en base de données avec à chaque fois la date du règlement le mode du règlement. Dès l'instant où le dernier règlement est enregistré pour la commande il devient impossible alors d'enregistrer d'autres règlements pour la commande ni d'en modifier.

Création et validation des commandes

Chaque commande doit passer par deux filtres afin d'être créée ou validée.

Les filtres pour ces deux opérations sont définis dans des fichiers dédiés :

Fichier "includes/article_add_check.php"

Contient les fonctions de vérification des conditions de création d'une nouvelle commande

```
if ($caisse[etat]=="N" || $caisse["id"]==0)
{
    ?>
    showErreurCaisse('<?=showTradJS("Veuillez ouvrir la caisse.")?>');
    showNotifInfoBox('<?=showTradJS("Veuillez ouvrir la caisse.")?>');
    getSystemPage('caisse_ouvre');
    <?
    die("");
}
if
(date("Y-m-d",strtotime($caisse[datedernierchangementetat])-60*60*$boutique[heureOuverture])<date("Y-m-d",
getBoutiqueTime()-60*60*$boutique[heureOuverture]) && $boutique["obligeFermeCaisseDaily"]==1 )
{
    ?>
    showErreurCaisse('<?=formatStr(showTradJS("La caisse n a pas ete fermee hier"),"Javascript")?>');
    showNotifInfoBox('<?=formatStr(showTradJS("La caisse n a pas ete fermee hier"),"Javascript")?>');
    getSystemPage('caisse_ferme');
    <?
    die("");
}
if
(date("Y-m",strtotime($caisse[datedernierchangementetat])-60*60*$boutique[heureOuverture])<date("Y-m",getB
outiqueTime()-60*60*$boutique[heureOuverture]) && $boutique["obligeFermeCaisseDaily"]==2 )
{
    ?>
    showErreurCaisse('<?=formatStr(showTradJS("La caisse n a pas ete fermee hier"),"Javascript")?>');
    showNotifInfoBox('<?=formatStr(showTradJS("La caisse n a pas ete fermee hier"),"Javascript")?>');
    getSystemPage('caisse_ferme');
    <?
    die("");
}
if (strtotime('+1 year', strtotime($caisse[datedernierchangementetat]))<getBoutiqueTime() &&
$boutique["obligeFermeCaisseDaily"]==3 )
{
    ?>
```

Dossier d'architecture technique

```
showErreurCaisse('<?=formatStr(showTradJS("La caisse n a pas ete fermee hier"),"Javascript")?>');
showNotifInfoBox('<?=formatStr(showTradJS("La caisse n a pas ete fermee hier"),"Javascript")?>');
getSystemPage('caisse_ferme');
<?
die("");
}
```

Deux vérifications sont faites : vérification de la bonne ouverture d'une caisse, et vérification de l'obligation de fermeture de caisse quotidienne/mensuelle/annuelle

Fichier "includes/shouldStartNewOrder.php"

Contient les fonctions de vérification des conditions de modification d'une commande

```
<?php
if (intval($idcommande)==0 || mystrtotime ($commande[datevalidation])>0 )
{
    if ($boutique[choixlivraisonobligatoire] && $boutique[eatinspecialtva] )
    {
        ?>
        askDeliveryMethod();
        showNotifInfoBox('<?=showTradJS("Préciser le choix livraison/à emporter/sur place.")?>');
        <?
        die("");
    }
    if ($idplu==-1) die(" ");
    $idcaisse = $caisse["id"];
    $defDateValeur = 0;

    $query = "INSERT INTO commandes (datecreation, datevaleur, idboutique, idcaisse, idutilisateur,
    idmodepaiement, prixinitial, totalreduction, totaltva, prixfinal, typetva, nbarticles,
    montantpaye,choixlivraison) VALUES (FROM_UNIXTIME(?),0,?,?,?,null,?,?,?,?,?,? )";

    mysql_q($query,array(getBoutiqueTime()),$idboutique,$idcaisse,$idutilisateur,$prixinitial,$totalreduction,$
    totaltva, $prixfinal,$typetva, $nbarticles, 0, $defaultChoixlivraison) );

    $idcommande = getLastID();
    $query = "UPDATE utilisateurs SET idcommande=? WHERE id=? ";
    mysql_q($query,array($idcommande,$idutilisateur) );
    $_SESSION['util'][$idcommande] = $idcommande;
    unset($commande);
    newOrderAction();
    echo "try {sIDc(\".$idcommande.\"); } catch (e) {}";
}
else
{
    $shallUpdatePrixFinal = true;
}
```

Si la commande a déjà été validée, créer une nouvelle commande vierge pour y appliquer l'opération demandée.

Ceci rend donc impossible l'édition d'une commande validée.

Si des corrections modifications ou annulation sont apportées à des transactions, par quelque moyens que ce soit, ces corrections s'effectueront alors par un enregistrement de données d'encaissement corrective avec des opérations de plus ou de moins et non par modification directe des données d'encaissement enregistrées.

Dossier d'architecture technique

Fichier "includes/coreCert.php"

Contient toutes les fonctions liées aux signatures. Fonction de création du sceau cryptographique ; Fonction de validation d'une commande ; Fonction de génération de la clé cryptographique associée à la boutique

```
function validateOrderIntoInvoice($idcommande_i) {
    global $boutique,$idcommande;

    $idcommande = $idcommande_i;
    $validationOfOrderOK = true;
    updatePrixCommande ();

    $commande = mysql_one("SELECT * FROM commandes WHERE id=? ",array($idcommande));

    if (mystrtotime ($commande["datevalidation"])==0) {
        mysql_q("UPDATE commandes SET
datevalidation=FROM_UNIXTIME(".(getBoutiqueTime()).") WHERE id=? " , array($idcommande)
);
        $reload = true;
    }
    if ( mystrtotime ($commande[datevaleur])==0) {
        mysql_q("UPDATE commandes SET datevaleur=FROM_UNIXTIME(".(getBoutiqueTime()).")
WHERE id=? " , array($idcommande) );
        $reload = true;
    }
    if ($reload) {
        $commande = mysql_one("SELECT * FROM commandes WHERE id=?
",array($idcommande));
    }

    $nextID = getCommandeNextID();

    //$query .= "UPDATE commandes SET
datevalidation=FROM_UNIXTIME(".(getBoutiqueTime()).")";
    $commande["idinterne"] = $nextID["maxID"];
    $query = "UPDATE commandes SET idinterne=".$commande["idinterne"];

    if ($nextID["thehash"] || $nextID["maxID"]==1) {
        $newHash =
makeHashOrder($boutique["id"],$commande,$nextID["thehash"],$nextID["maxID"]);

        $commande["thehash"] = $newHash;
        $query .= ",thehash='".$commande["thehash"].'";
    }
    $query .= " WHERE idinterne=0 AND id=".$idcommande;
    /*echo $query;
    die();*/
    mysql_query_perso($query);

    mysql_q("UPDATE boutiques SET statNbSales=statNbSales+1 WHERE id=? ",
array($boutique[id] ) );

    return $commande;
}
function makeHashOrderAddB($idboutique,$hash) {
    return $hash."_".sha1($idboutique.$softwareSecretKey);
}
function makeHashOrder($idboutique,$c,$hash,$id) {
    $stringRes =
makeHashOrderAddB($idboutique,$c["datevalidation"]."_". $c["datevaleur"]."_". $c["idcaiss
e"]."_". $c["prixinitial"]."_". $c["totalreduction"]."_". $c["totaltva"]."_". $c["prixfinal
"]."_". $c["typetva"]."_". $c["nbarticles"]."_". $c["choixlivraison"]."_". $c["id"]."_". $c[
"idinterne"]."_". $hash."_". $id);
    return hash('sha256', $stringRes);
}
```



Dossier d'architecture technique

```
}
function getCommandeNextID() {
    global $boutique;
    my_sem_get("lockBoutique".$boutique["id"]);
    $i = mysql_q("SELECT max(idinterne) as maxid FROM commandes WHERE
idboutique=?",array($boutique["id"]));
    $maxid = $i[0][maxid];

    $c = mysql_one("SELECT thehash FROM commandes WHERE idboutique=? AND idinterne=?
LIMIT 1",array($boutique["id"],$maxid));
    return array("maxID"=>$maxid+1,"theshash"=>$c["theshash"]);
}

function updatePrixCommande () {
    global
    $utilisateur,$boutique,$idcommande,$recalcTitre,$erreurReduction,$validationOfOrderOK;
    $comptabiliseCA = true;
    require "includes/export/calculePrixCommande.php";
    if ($validationOfOrderOK) {
        foreach ($pileArticlesRendus as $listedArt) {
            if ($listedArt["totalTTC"]) {
                mysql_q("UPDATE articles SET prixFinalHT=? WHERE
id=?",array($listedArt["totalTTC"]-$listedArt["totalTVA"],$listedArt["id"]));
            }
        }
    }
    //die();
    // un hack car sinon il faut recalculer les valeurs de prixfinal lors de la
suppression d'un article
    $montantTotalTVA = 0;
    foreach ($totauxTVA as $taux => $total)
        $montantTotalTVA += $total;
    $commande["prixfinal"] = priceImport($totalPrixFinal);
    $commande["totaltva"] = $montantTotalTVA;

    if (!$erreurReduction) {
        $qP = "UPDATE commandes SET
narticles=".$nbTotalArticles.",prixfinal=".$totalPrixFinal.",totaltva=".$montantTotalT
VA." WHERE idinterne=0 AND id=".$idcommande."";
        mysql_query_perso($qP);
    }
    return $commande;
}
```

On trouve dans ce fichier les principales fonctions :

- validateOrderIntoInvoice : permet de transformer un devis en commande, donc lui affecter un numéro de commande, une date de valeur, un hash SHA 256
- makeHashOrder : fonction de création du hash SHA256
- getCommandeNextID : permet d'obtenir le prochain identifiant de commande
- makeHashOrderAddB : génère la clé privée d'un établissement à partir d'un sha1, et d'une clé privée du logiciel.
- updatePrixCommande : lors de la validation de la commande, calcule la valeur définitive des prix HT des articles de la commande, et les enregistre en base de donnée

Sécurisation des justificatifs

Le système d'encaissement permet de distinguer et d'identifier sans ambiguïté un justificatif émis avant paiement d'un justificatif émis après paiement. Tout justificatif réimprimé porte ainsi la mention duplicata. Le système assure la traçabilité des impressions et des réimpressions de justificatif de manière sécurisée. L'opération est enregistrée en base de données, côté serveur sans accès possible de la part du client. En cas de téléchargement, d'envoi par email ou d'envoi par SMS de la facture, le champ en base de données duplicata est positionné. Chacune des méthodes d'envoi des factures est soumise aux mêmes contraintes concernant l'émission et la rémission.

Signature des transactions

Chaque transaction est signée selon l'algorithme SHA256, en chaînant chaque signature à celle de la transaction précédente, assurant l'intégrité de l'historique.

La signature d'une transaction se base sur les informations suivantes :

- prix final de la commande
- prix initial de la commande
- Date de valeur
- date de validation
- identifiant interne de la caisse ayant procédé à l'enregistrement
- montant total des réductions appliquées à la commande
- montant total de la TVA collectée
- nombre d'articles dans la commande
- choix de la méthode de livraison
- identifiant interne de la commande
- identifiant public de la commande
- hash cryptographique de la commande précédente portant le numéro interne inférieur
- identifiant interne de la boutique

```
function validateOrderIntoInvoice($idcommande_i) {
    global $boutique,$sidcommande;

    $idcommande = $idcommande_i;
    updatePrixCommande ();

    $commande = mysql_one("SELECT * FROM commandes WHERE id=? ",array($idcommande));

    if (mystrtotime ($commande["datevalidation"])==0) {
        mysql_q("UPDATE commandes SET datevalidation=FROM_UNIXTIME('.".getBoutiqueTime().")' WHERE id=? ",
        array($idcommande) );
        $reload = true;
    }
    if ( mystrtotime ($commande[datevaleur])==0) {
        mysql_q("UPDATE commandes SET datevaleur=FROM_UNIXTIME('.".getBoutiqueTime().")' WHERE id=? ",
        array($idcommande) );
        $reload = true;
    }
    if ($reload) {
        $commande = mysql_one("SELECT * FROM commandes WHERE id=? ",array($idcommande));
    }
}
```

Dossier d'architecture technique

```
$nextID = getCommandeNextID();

$commande["idinterne"] = $nextID["maxID"];
$query = "UPDATE commandes SET idinterne=".$commande["idinterne"];

$newHash = makeHashOrder($boutique["id"],$commande,$nextID["theshash"],$nextID["maxID"]);

$commande["theshash"] = $newHash;
$query .= ",theshash='". $commande["theshash"]."'";

$query .= " WHERE idinterne=0 AND id=".$sidcommande;

mysql_query_perso($query);

mysql_q("UPDATE boutiques SET statNbSales=statNbSales+1 WHERE id=? ", array($boutique[id] ) );

return $commande;
}

function makeHashOrderAddB($idboutique,$hash) {
    return $hash."_".sha1($idboutique."myCustomSoftwareKey5d9RuGgugéd_tçdjkdz");
}

function makeHashOrder($idboutique,$c,$hash,$id) {
    $srcHash =
    $stringRes =
    makeHashOrderAddB($idboutique,$c["datevalidation"]."_". $c["datevaleur"]."_". $c["idcaisse"]."_". $c["prixinitial"]."_". $c
["totalreduction"]."_". $c["totaltva"]."_". $c["prixfinal"]."_". $c["typetva"]."_". $c["nbarticles"]."_". $c["choixlivraison
"]."_". $c["id"]."_". $c["idinterne"]."_". $hash."_". $id);
    return hash('sha256', $stringRes);
}

function getCommandeNextID() {
    global $boutique;
    my_sem_get("lockBoutique".$boutique["id"]);
    $i = mysql_q("SELECT max(idinterne) as maxid FROM commandes WHERE idboutique=?",array($boutique["id"]));
    $maxid = $i[0][maxid];

    $c = mysql_one("SELECT thehash FROM commandes WHERE idboutique=? AND idinterne=? LIMIT
1",array($boutique["id"],$maxid));
    return array("maxID"=>$maxid+1,"theshash"=>$c["theshash"]);
}
```

4. Gestion des sauvegardes et des archives

Sauvegarde par l'utilisateur du logiciel d'encaissement

Le logiciel permet de télécharger les données dans différents formats, ceux-ci peuvent être Format HTML, PDF, FEC, SAF-T, CSV, Excel.

Dans ce logiciel les sauvegardes et les archives sont intitulés des rapports et ils peuvent être téléchargés en se rendant dans la page rapport du logiciel.

Indépendamment du format d'export choisi, le rapport présentera à minima les informations suivantes, adossées aux commandes : l'identifiant interne de la commande, sa date de création, sa date de validation, sa date de valeur, le nom de la caisse ayant effectué la vente, le dernier mode de paiement utilisé, la table du client, le client en question (son identifiant son nom et son prénom), les différents montants de TVA ventilés par taux de TVA, le nombre d'articles, le montant payé, le prix final, le prix d'achat, le détail de la commande, et sceau cryptographique de la commande, et le vendeur qui a procédé à l'enregistrement de la commande.

Il sera possible de télécharger uniquement les données qui sont associées à une caisse en particulier ou à toutes les caisses, un utilisateur en particulier ou à tous les utilisateurs.

Différentes plages de dates pour l'export de données seront disponibles : l'export quotidien, hebdomadaire, mensuel, annuel mais il sera également possible de télécharger des rapports portant sur une plage de date personnalisée.

Deux formats de rapport se distinguent des autres rapports parce qu'ils présentent un panel plus élargi de donner pour le rapport. Ces rapports sont les rapports au format PDF et HTML qui permettent d'afficher différentes tables de données au sein d'un même fichier. On retrouvera dans ses rapports : un tableau récapitulatif qui présente la date de création du rapport, le chiffre d'affaires global, le montant total des ventes et des remboursements, le montant total de la TVA collectée, le nombre d'opérations enregistrées, la valeur moyenne d'une opération. Sous forme de tableau distincts on retrouvera ensuite : la liste complète des commandes, le rapport par taux de TVA, le rapport par table de consommation, le rapport quotidien, le rapport des sorties de caisse, le rapport des fermetures de caisse, le graphique des ventes au cours du temps, le graphique du nombre de ventes au cours du temps, le rapport des fermetures de caisse, le rapport par caisse, par client, par utilisateur, par rayon, par groupe de rayon, par article, par méthode de paiement, par méthode de livraison, le rapport des réductions, le rapport sur les heures d'affluence, le rapport par chapitre comptable, le rapport des livraisons externes, le rapport sur les DLC.

Les rapports peuvent être également configurés pour être automatiquement envoyés quotidiennement ou mensuellement.

Archivage en utilisant la fonctionnalité de rapport permet d'obtenir des fichiers non compressés.

Dossier d'architecture technique

La fonctionnalité de rapport ou d'archivage permet d'exporter les données portant sur n'importe quelle plage de date. La période couverte par une archive ne peut pas être supérieure à un an ou un exercice fiscal.

Les archives téléchargées par l'utilisateur sont également enregistrées dans notre base de données et sont conservées pendant une durée qui dépend de la période sur laquelle porte l'archive. Si la période est une année la durée de conservation du rapport sera de 7 ans, si la période est d'un mois, la durée de conservation sera d'un an, pour une période d'une semaine, 30 jours de conservation, pour un rapport quotidien 10 secondes de conservation, pour un rapport sur une plage de date personnalisée 7 ans de conservation.

Sauvegardes réalisées en interne par l'éditeur du logiciel

- Serveur de réplication de base de donnée en temps réel
- Sauvegardes automatiques des bases de données.
- Sauvegarde sur support externe tous les 3 mois

5. Conservation des données

Les données d'encaissement, de traçabilité, ainsi que les preuves de l'inaltérabilité, sont conservées de manière illimitée dans le temps en base de données. Les données cumulatives ainsi que les données de traçabilité sont également conservées dans le système sans limite de durée. Les données d'encaissement peuvent être conservées soit dans le système lui-même, soit dans le système lui-même ainsi que dans les archives téléchargées par l'utilisateur du système de caisse.

6. Mode hors ligne

Le mode hors ligne existe dans le logiciel mais celui-ci ne permet pas d'enregistrer des opérations de caisse mais seulement de préparer des brouillons de commandes qui ne pourront pas être validés (cf chapitre sur la validation des commandes) tant que l'accès à Internet ne sera pas rétabli.

Cela signifie qu'il n'est pas possible d'imprimer un reçu de caisse, de télécharger une facture, d'enregistrer une facture, d'obtenir un numéro de commande, etc.

7. Système centralisateur

La conservation des données étant assurée par un système centralisateur, le système d'encaissement a prévu un mécanisme fiable de transfert des données pour assurer que l'exhaustivité du flux de données soit transférée y compris en cas de déconnexion en cours.

Lorsque l'utilisateur du système de caisse ajoute un article à une commande, celui-ci s'affiche temporairement sur la commande et une requête est envoyée au serveur afin d'enregistrer la modification. Lorsque la requête aboutit, le ticket de caisse est mis à jour afin de présenter la commande telle qu'elle est enregistrée dans le système centralisateur. L'opération de validation permettant de transformer le devis en facture ne pourra être réalisée que lorsque toutes les opérations de modification de la commande ont abouti et que l'utilisateur dispose à l'écran de la version enregistrée dans le système de centralisation de la commande.

8. Accès de l'administration fiscale aux données

Le système d'encaissement prévoit un accès pour l'administration fiscale à l'ensemble des données d'encaissement enregistrées. Nous fournissons un moyen automatisé pour que l'administration fiscale puisse vérifier l'intégrité des données d'encaissement. Un manuel de l'utilisateur à destination de l'administration fiscale, en français, détaille la procédure permettant d'accéder aux données, ainsi qu'un descriptif clair du fonctionnement des outils utilisés pour accéder aux données et en vérifier l'intégrité.

L'administration fiscale a la possibilité de soit s'adresser à l'utilisateur du système de caisse ou de soit s'adresser directement à l'éditeur du logiciel de caisse, chacun de ces interlocuteurs sera en mesure de communiquer à l'administration fiscale un accès qui lui sera destiné. La procédure à réaliser pour accorder un tel accès à l'administration fiscale est très simple. Il suffit de se rendre en page Configuration, Utilisateurs, puis d'ajouter un nouveau compte utilisateur en saisissant son email, un mot de passe qui lui sera attribué et le rôle "Comptable". Le rôle comptable est un rôle commun à l'administration fiscale et au comptable de l'utilisateur du système d'encaissement.

Une fois connecté avec un tel compte l'administration fiscale aura accès au logiciel avec les onglets suivants :

Historique : cette page permet d'accéder à la liste complète des commandes du compte, il est possible d'effectuer une recherche par numéro de commande par date par identifiant client par nom de clients par caisse d'encaissement

Rapport : cette page permet d'accéder au rapport comptable et aux données consolidées intitulées dans ce document archives. Il sera possible dans cette page de procéder aux exports de données au format CSV, XLS, FEC, SAFT, HTML, PDF

Vérification des sceaux cryptographiques

Dossier d'architecture technique

Cette page permet d'accéder à deux outils : le premier effectue une vérification de la chaîne de sceaux cryptographiques de l'intégralité des commandes de l'utilisateur du système d'encaissement et affiche le résultat, le deuxième outil permet de manuellement vérifier l'exactitude de la valeur du code de hachage enregistré dans la base de données courant l'authenticité l'intangibilité des données enregistrées.

Aide : cette page permet d'accéder à l'aide utilisateur du logiciel. Il est possible dans cette page de consulter la procédure permettant d'accéder aux données de caisse ainsi qu'un descriptif clair du fonctionnement des outils utilisés pour accéder aux données et en vérifier l'intégrité.

L'accès destiné à l'administration fiscale ne permettrait pas de modifier d'ajouter des données d'encaissement, cet accès sera limité à une consultation.

9. Gestion des incidents

- Surveillance 24/7 des serveurs via monitoring interne (munin, outils internes digital ocean, outils de notification en cas de crash)

10. Purges

Ce logiciel ne réalise pas de purge de données, les données sont conservées de manière illimitée dans le temps en base de données.

11. Versions de l'application

Le système d'encaissement identifie clairement ces versions par un numéro de version majeur, un numéro de version mineur, et un numéro de révision. Ces numéros de version sont inextricablement liés au système d'encaissement. Ils sont aisément accessibles depuis l'interface utilisateur standard du système d'encaissement en vous rendant en page aide rubrique mention légale du logiciel. Toute modification de code dans le périmètre fiscal ou dans le paramétrage impactant le respect des exigences du référentiel entraîne une incrémentation du numéro de version majeure. une empreinte unique pour chaque version majeure et générer en fonction de la concaténation de la liste des fichiers qui constituent le périmètre fiscal. Cette empreinte est réalisée en effectuant un SHA 256 sur le résultat de la concaténation du contenu des fichiers qui constituent le périmètre fiscal et du numéro de version majeure. Cette empreinte peut être consultée en page aide du logiciel.

12. Périmètre fiscal

Le système d'encaissement maintient la liste des fichiers du code source impliqués dans les exigences techniques applicables au système d'encaissement. Cette liste de fichiers constitue le périmètre fiscal du logiciel.

Le tableau suivant récapitule pour chaque exigence le périmètre fiscal du code source afférent à l'exigence en question.

Exigence	Titre	Périmètre fiscal code source
Exigence 1	Documentation règlementaire	/documentation/*
Exigence 2	Documentation complémentaire	/documentation/*
Exigence 3	Enregistrement des données	/includes/coreCert.php
Exigence 4	Corrections	/includes/shouldStartNewOrder.php
Exigence 5	Mode école	/includes/coreCert.php
Exigence 6	Clôtures	/includes/article_add_check.php
Exigence 7	Données cumulatives	/includes/coreCert.php
Exigence 8	Inaltérabilité	/includes/coreCert.php
Exigence 9	Sécurisation	Sys admin
Exigence 10	Archivage	/includes/coreCert.php
Exigence 11	Périodicité d'archivage	/includes/coreCert.php
Exigence 12	Intégrité des archives	/includes/coreCert.php
Exigence 13	Purge	Non applicable
Exigence 14	Purge partielle	Non applicable

Exigence 15	Journalisation	Bin logs
Exigence 16	Conservation des données en base	MySQL
Exigence 17	Conservation des archives	Documentation
Exigence 18	Centralisation ou détection des interruptions	Non applicable
Exigence 19	Accès administration fiscale	Documentation
Exigence 20	Identification du périmètre fiscal	/documentation/*
Exigence 21	Gestion des versions	/includes/coreCert.php
Exigence 22	Intégrité lors de la mise à jour	SVN

Conformité au Référentiel de Certification v1.7

Documentation technique (Exigence IV.1)

La structure du système et les processus sont documentés. L'organisation et les technologies sont précisément décrites.

Inaltérabilité et sécurisation des données (Exigence IV.4 - Exigences 8 et 9)

Le chaînage SHA256 assure l'inaltérabilité. Les transmissions et les données sont sécurisées via HTTPS/TLS et accès SSH restreint.

Archivage (Exigence IV.5 - Exigence 10)

Les données sont archivées périodiquement et sécurisées par compression et signature.

Conservation des archives (Exigence IV.8 - Exigences 16 et 17)

Les archives sont conservées sur des serveurs protégés pour une durée minimale de 6 ans.

Système centralisateur et mode hors-ligne (Exigence IV.8 - Exigence 18)

Le logiciel ne permet pas la réalisation de facture en mode déconnecté. Le mode déconnecté permet uniquement la préparation de devis / brouillons et ne permet pas l'enregistrement de commandes, ni l'émission de documents justificatifs. Il sera ensuite nécessaire de se

Dossier d'architecture technique

reconnecter à Internet puis d'importer les brouillons dans le logiciel, afin de les transformer en commande, afin de pouvoir obtenir une facture, un ticket de caisse, ou un justificatif. Le mode déconnecté n'est prévu que pour subvenir à des courtes interruptions de service de l'ordre de quelques minutes.

Accès administration fiscale (Exigence IV.9 - Exigence 19)

Un module d'export CSV est disponible pour répondre aux demandes de l'administration fiscale.