

DOCUMENTATION API

SPARKWEB – V25.41



POUR
SPARKWEB

SAS au capital de 30 000 € – RCS Mâcon 519 816 292 – NAF 6202A



SAS Greenspark
16 Rue Nationale 71420 G  nelard FRANCE



+33 9 72 55 39 22



contact@greenspark.fr



www.greenspark.fr

SOMMAIRE

1	DESCRIPTIF DU DOCUMENT	3
2	GESTION DES TOKENS	4
2.1	Token de type (Agent VRP)	5
2.2	Guide pour l'intégration avec un logiciel de gestion d'agent / VRP	6
3	INPUT / OUTPUT	7
3.1	Paramètre Entrée	7
3.2	Paramètre en sortie	7
4	ARTICLES	8
4.1	Récupération d'un article (GetArticle) — Obsolète	8
4.2	Récupération de plusieurs articles (GetArticles)	9
4.3	Mise à jour d'un article (UpdateArticle)	11
4.4	Récupération des tarifs (GetTarifs)	12
5	CLIENTS	13
5.1	Vérification qu'un client existe (DoesClientExist)	16
5.2	Création d'un client (CreateClient)	17
5.3	Création d'un client (UpdateClient)	18
5.4	Récupération d'un client - Obsolète	19
5.5	Récupération de plusieurs clients (GetClients)	20
5.6	Récupération de plusieurs Contacts (GetContacts)	23
5.7	Modification mail client (UpdateEmail)	24
5.8	Lier le client web avec le client sparkwine (LinkClient)	25
6	COMMANDES	26
6.1	Schéma d'un flux complet classique (Nouvelle Commande)	27
6.2	Création d'une commande (CreateOrder)	28
6.3	Création d'une ligne de commande (CreateOrderLine)	30
6.4	Validation de la commande (ValidateOrder)	33
6.5	Récupération (GetOrder)	34
6.6	DashBoard – Commande (SparkWine)	35
7	GESTION DES ERREURS	36

1 DESCRIPTIF DU DOCUMENT

Ce document représente la documentation concernant l'API de communication entre un site externe et le logiciel SparkWine.

Cette interface (API) va permettre l'échange de données entre le site WEB et l'ERP grâce à différents modules :

- Partie Article :
 - ~~GetArticle~~ : Récupération d'un article
 - **Obsolète on utilise plutôt GetArticles**
 - **GetArticles** : Récupération de plusieurs articles
 - **UpdateArticle** : Mise à jour de l'id web d'un article
 - **GetTarifs** : Récupération de plusieurs tarifs associés aux articles
- Partie Commande :
 - Création de nouvelles commandes
 - En 3 étapes :
 - **CreateOrder** :
 - Création de l'entête
 - **CreateOrderLine** :
 - Création des lignes (On appel autant de fois qu'il y a de lignes)
 - **ValidateOrder** :
 - Indique que l'on a terminé, on passe ici certains champs qui sont utilisé pour un contrôle d'intégrité + Infos de paiement
 - **GetOrder** : Récupération des informations d'une commande
- Partie Client :
 - **CreateClient** : Création d'un client
 - **DoesClientExist** : Retour le code client SparkWine en fonction d'un eMail
 - **UpdateClient** : Mise à jour d'un client
 - ~~GetClient~~ : Récupération d'un client
 - **Obsolète on utilise plutôt GetClients**
 - **GetClients** : Récupération de plusieurs clients
 - **LinkClient** : Permet de lier l'id web avec l'id sparkwine
 - **UpdateMail** : Met à jour l'email sur SparkWine

Nous allons dans la suite de ce document faire une description de chaque méthode d'api.

2 GESTION DES TOKENS

Dans chaque appel d'api il va falloir générer un système de clés afin de pouvoir communiquer en sécurité. Ainsi, deux paramètres seront toujours présents : la clé publique et la clé hash.

Pour chaque appel d'api :

- Création d'une clé publique
 - Type MD5
 - Doit être unique à chaque appel
- Création d'une clé hash
 - Concaténation de plusieurs données :
 - Token privé
 - Fourni au préalable par GreenSpark
 - Clé publique
 - Date au format « Ymd » : 20200908

Dans SparkWine il est possible d'avoir plusieurs Token, pour chaque Token on peut définir un comportement et des valeurs par défaut différentes dans SparkWine :

The screenshot displays the 'Général' (General) tab of the SparkWine API configuration interface. The interface is divided into several sections:

- Infos générales**: Includes a 'Token' field with the value 'InmCQRXISQK8kCx6cZI4Kcl8GcCB9uj7r33b4pskR9hpBxUtzE' and a 'PersSign' button labeled 'RDE'.
- Articles**: A dropdown menu for 'Code dépôt' set to '01 - DEPOT PRINCIPAL'.
- Clients**: A series of dropdown menus for 'Type n° séquence' (UNIQUE), 'Tarif' (01 - TARIF GENERAL), 'Devise' (EUR), 'Conditions paiement' (15 Jours), 'Conditions Expédition' (CHEZ LE CLIENT (FRANCO) TEST), 'Mode de paiement' (VIREMENT), 'Mode d'expédition' (COLISSIMO), 'Langue' (FRANÇAIS), 'Catégorie Client' (CLIENT AUTRE), 'TVA' (FRANCE 20%), 'Commercial' (zNon Défini), 'Exprimé en TTC ?' (checked), and 'Reste pmt/règle inv. int.' (PAS DE RELANCE).
- Comportement de l'API**: Includes a 'Type Token' dropdown (Logiciel VRP / Agent), a checked 'Journal règlement / code', an unchecked 'Accès encours comptable', an unchecked 'Qté WEB = Qté US ?', and a 'Stock sécu (Déduit du dis 5)' field.
- A Trier**: A series of dropdown menus for 'Statut initial' (ADRESSE LIVRAISON), 'Lancement Projet' (AUTRE), 'Zone Commerciale' (TEST), 'Union Economique' (AELE), 'Libellé Mode Com.' (COURRIER), 'Régime fiscal' (DROITS ACQUITTES), 'Firme' (Général), 'Secteur Com.' (zNon Défini), 'Devise' (EUR), 'Classe rem Client' (CHR), '% Remise Client' (0,00), 'Dernière récupération' (23/12/2019 12:19), 'Cumul stock', 'Agent' (WINE MERCHANTS), 'Transporteur', and 'Activité'.
- API Externe**: Includes fields for 'Url site web' and 'Token site web'.
- Commande**: Includes a 'Type Commande' dropdown (01 - VENTE NORMALE), a 'Campagne Marketing' dropdown, and an 'Update OH' button.
- Dépôts web**: A table with columns 'Dépôt' and 'Dépôt'.

2.1 Token de type (Agent VRP)

Un **token distinct doit être généré pour chaque agent**, afin de segmenter leurs accès et de limiter leur visibilité aux seuls clients et commandes qui les concernent.

Paramétrage du token

Les paramètres suivants doivent être définis :

- **Type de token** : Logiciel VRP / Agent
- **Qte Web = Qte US** : Le logiciel de l'agent doit envoyer une quantité de 12 pour 12 bouteilles, exprimée en unité de stock
- **Agent** : Sélectionner l'agent concerné

Il est également nécessaire de compléter les autres champs avec les **valeurs par défaut souhaitées**.

The screenshot shows a configuration window with the 'Général' tab selected. The 'Token' field contains the string 'InnCQRXISQK3KcX6cZi4Kc8GcCB9uj7r33b4pskR9hpBxUtzE'. In the 'Comportement de l'API' section, the 'Type Token' dropdown is set to 'Logiciel VRP / Agent', the 'Qte WEB = Qte US ?' checkbox is checked, and the 'Agent' dropdown is set to 'WINE MERCHANTS'. The 'PersSign' section has a 'RDE' button.

Transmission du token

Une fois le token généré, **il doit être transmis à l'agent concerné**, qui devra l'enregistrer dans la fiche du **Fournisseur / Domaine / Cave** directement depuis son logiciel, afin d'établir le lien de connexion.

2.2 Guide pour l'intégration avec un logiciel de gestion d'agent / VRP

Pour l'intégration avec un logiciel d'agent, il faut savoir que **SparkWine fournira un Token distinct pour chaque agent et pour chaque société** (domaine, cave, etc.).

L'objectif est que le logiciel de VRP associe chaque *fournisseur* lié à un agent à un Token unique.

Ce Token permettra ensuite d'accéder aux fonctionnalités suivantes :

Articles

- Récupération des disponibilités, descriptions et tarifs des articles
- Seuls les articles présents dans le tarif propre à l'agent sont accessibles

Clients

- Consultation des clients liés à l'agent
- Création et modification de fiches clients
- Seuls les clients rattachés à l'agent sont visibles

Commandes

- Consultation des statuts et des informations des commandes
- Création de nouvelles commandes

Certaines fonctions ou critères ne sont peut-être pas disponible dans l'API cas échéant n'hésitez pas à nous demander pour que nous les rajoutions. (Nous le ferons volontiers si évidemment nous en avons la possibilité)

3 INPUT / OUTPUT

3.1 Paramètre Entrée

Dans chacune des API il y a donc toujours 2 paramètres minimum en entrée :

- HashKey [string]
- PublicKey [string]

*NB : Dans la suite de l'API nous n'allons pas les préciser pour chaque fonction nous allons simplement écrire : **TokenManagement***

3.2 Paramètre en sortie

Les sorties de chaque fonction peuvent avoir 2 formes :

- Chaque méthode retourne un objet avec le nom de la méthode + Responses (Il ne retourne toujours qu'un seul élément Responses)
- Si la méthode gère un retour multiple elle finit par un « s » et elle contient donc un Tableau avec les éléments dans l'objet de réponse
- Exemple :
 - Méthode a retour unique : GetArticle
 - Structure de GetArticleResponse
 - IsError : STRING
 - ErrorCode : INTEGER
 - ErrorMessage : STRING
 - Code : STRING
 - WebCode : STRING
 - Méthode a retour multiple : GetArticles
 - GetArticleResponse :
 - IsError : STRING
 - ErrorCode : INTEGER
 - ErrorMessage : STRING
 - Articles : Articles [] (Tableau avec les articles)
 - Code : STRING
 - WebCode : STRING

Il y a aussi toujours au minimum 3 paramètres pour la gestion des erreurs dans les objets XXXResponses

- IsError [string] → 0 ou 1
 - Erreur lors de l'appel
- ErrorCode [int]
 - Code d'erreur
- ErrorMessage [string]
 - Message de l'erreur

*NB : Dans la suite de l'API nous n'allons pas les préciser pour chaque fonction nous allons simplement écrire : **ErrorManagement***

Attention : Il faut donc à chaque appel regarder si le champ « IsError » de la Response est à 1 (Si c'est le cas il faut raise une erreur)

4.1 Récupération d'un article (GetArticle) – Obsolète

Remplacée par l'utilisation de GetArticles en spécifiant le code article

Permet de récupérer l'articles en fonction du code Web / SparkWine et s'il est coché « Disponible en ligne »

Procédure : `q_gs_sWeb_p_get_article`

- Paramètres :

- **TokenManagement**

- Code [string]

- Code Sparkwine (Unique dans SparkWine)

- WebCode [string]

- Code Web (Identifiant dans le site Web)

- GetArticleResponse :

- **ErrorManagement**

- Code [string]

- Code Sparkwine

- WebCode [string]

- Code Web

- Title [string]

- Nom de l'article

- Millesime

- Millesime de l'article

- Brand [string]

- Marque de l'article

- Color [string]

- Couleur de l'article

- Conditioning [integer]

- Conditionnement de l'article (nombre de bouteilles)

- Packaging [string]

- Format du carton

- Price [float]

- Prix tarifs de l'article (HT ou TTC en fonction du paramétrage du Tarifs dans SparkWine)

- ConditioningPrice [float]

- Prix de l'article en fonction du conditionnement ($\text{Conditioning} * \text{Price}$)

- IsSalable [string] → 0 ou 1

- L'article est-il vendable ?

- GrossWeight [float]

- Poids de l'article

- Stock [float]

- Stock de l'article

4.2 Récupération de plusieurs articles (GetArticles)

Permet de récupérer tous les articles qui sont cochés « Disponible en ligne »

Possibilité de filtrer par date dans un intervalle pour récupérer seulement les articles modifiés durant la période saisie.

Un filtre technique peut être paramétrer dans Sparkwine pour ne récupérer que des articles avec un certain paramétrage

Procédure : q_gs_sWeb_p_get_articles

- Paramètres :
 - **TokenManagement**
 - **WebArticleId [string]**
 - **Code Web ou Sparkwine ou Vide pour tout récupérer**
 - **StartDate [DateTime]**
 - **Date début pour l'intervalle de modification**
 - **EndDate [Datetime]**
 - **Date de fin pour l'intervalle de modification**
- GetArticleResponse :
 - **ErrorManagement**
 - Articles [] → Tableau des articles :
 - Code [string]
 - Code Sparkwine
 - WebCode [string]
 - Code Web
 - Title [string]
 - Nom de l'article
 - Millesime
 - Millesime de l'article
 - Brand [string]
 - Marque de l'article
 - Color [string]
 - Couleur de l'article
 - Conditioning [integer]
 - Conditionnement de l'article (nombre de bouteilles)
 - Packaging [string]
 - Format du carton
 - **Valeur possible : Dépend de l'environnement (Voir avec le client dans SparkWine)**
 - IsSalable [string] → 0 ou 1
 - L'article est-il vendable ?
 - GrossWeight [float]
 - Poids de l'article
 - RegionCode [string]
 - Code région
 - RegionTitle [string]
 - Libellé région
 - CuveeCode [integer]
 - Code cuvée (ou sous-région)
 - CuveeTitle [string]

- Libellé cuvée (ou sous-région)
- AppellationCode [string]
 - Code Appellation
- AppellationTitle [string]
 - Libellé appellation
- GammeCode [string]
 - Code domaine
- GammeTitle [string]
 - Libellé domaine
- NcCode [string]
 - Nomenclature douanière
- Degre [float]
 - Degré
- Custom1Code [string]
 - Code champs personnalisé 1
- Custom1Title [string]
 - Libellé champs personnalité 1
- Custom3Code [string]
 - Code champs personnalisé 3
- Custom3Title [string]
 - Libellé champs personnalité 3
- Stock [float]
 - Stock de l'article
 - En fonction du cumul de stock choisis dans les paramètres du Token
- CepageCode [string]
 - Code cépage
- CepageTitle [string]
 - Libellé cépage
- PaysCode [string]
 - Code pays
- PaysTitle [string]
 - Libellé pays
- EANCode [string]
 - Code EAN
- GestionRegieCode [string]
 - Code Gestion régie
- GestionRegieTitle [string]
 - Libellé gestion régie
- TVACode [integer]
 - Code TVA
- TVATaux [float]
 - Taux de TVA
- TVATitle [string]
 - Libellé TVA

4.3 Mise à jour d'un article (UpdateArticle)

Permet d'associer un article web avec un article SparkWine

Procédure : q_gs_sWeb_p_update_article

- Paramètres :
 - **TokenManagement**
 - Code [string]
 - Code Sparkwine
 - WebCode [string]
 - Code Web
- UpdateArticleResponse
 - **ErrorManagement**
 - Code [string]
 - Code Sparkwine
 - WebCode [string]
 - Code Web

4.4 Récupération des tarifs (GetTarifs)

Permet de récupérer tous les prix d'un article pour un tarif donné

Dans le cas d'un paramétrage VINIVRP, on ne voit que les tarifs de l'agent

Procédure : q_gs_sWeb_p_get_tarifs

- Paramètres :
 - **TokenManagement**
 - CodeTarif [integer]
 - Code Tarif
 - StartDate [DateTime]
 - Date début pour l'intervalle de modification
 - EndDate [Datetime]
 - Date de fin pour l'intervalle de modification
- GetTarifsResponse
 - **ErrorManagement**
 - Code [string]
 - Code Sparkwine
 - WebCode [string]
 - Code web
 - Price [float]
 - Prix du tarif
 - LowerLimit [float] [nullable]
 - Quantité seuil du tarif
 - IsTtc [boolean]
 - Le tarif est-il géré en TTC ?
 - AlternativeUnit [string] [nullable]
 - Unité alternative
 - AlternativeUnitFactor [float]
 - Facteur de conversion de l'unité (1 par défaut)
 - BaremeCode [integer]
 - Code du Barème associé au tarif
 - TypeValeurCode [integer]
 - Code Type de valeur
 - TypeValeurLib [string]
 - Nom Type de valeur
 - TypeSeuilCode [integer]
 - Code type de seuil
 - TypeSeuilLib [string]
 - Nom Type de seuil
 - BaseCalcCode [integer]
 - Code base de calcul
 - BaseCalcLib [string]
 - Nom Base de calcul
 - Devise [string]
 - La devise
 - PrixHorsPort [float]
 - Prix Hors port

4.5 Recherche dynamique d'articles (GetArticlesSearch)

Permet de réaliser une recherche dynamique d'articles en fonction de multiples critères. Les résultats sont retournés avec un score de pertinence pour permettre un tri par ordre de correspondance. Cette fonction est particulièrement utile pour implémenter une barre de recherche intelligente.

La pertinence des résultats est calculée via un système de score qui pondère l'importance de chaque critère. Par exemple, une correspondance sur le CodeArticle ou le NomArticle aura plus de poids dans le score final.

Procédure : q_gs_sWeb_p_article_search

- **Paramètres :**
 - TokenManagement
 - **Critères de recherche (tous optionnels) :**
 - CodeSociete [string][nullable]
 - Code société de l'article
 - NomArticle [string][nullable]
 - Nom ou partie du nom de l'article.
 - CodeArticle [string][nullable]
 - Code Sparkwine ou partie du code de l'article.
 - Appellation [string][nullable]
 - Libellé de l'appellation.
 - Millesime [string][nullable]
 - Le millésime de l'article.
 - Couleur [string][nullable]
 - Couleur du vin.
 - Region [string][nullable]
 - Région de production.
 - Domaine [string][nullable]
 - Nom du domaine ou de la gamme.
 - Cepage [string][nullable]
 - Cépage principal.
 - Marque [string][nullable]
 - Marque de l'article.
 - FormatBouteille [string][nullable]
 - Format de la bouteille (ex: "75cl").
 - Degre [float][nullable]
 - Degré d'alcool.
 - IsVin [char(1)][nullable]
 - Indique si l'article est un vin ('1' pour Oui, '0' pour Non).

- IsVinSansAlcool [char(1)][nullable]
 - Indique si l'article est un vin sans alcool ('1' pour Oui, '0' pour Non).
- TypeConditionnement [string][nullable]
 - Code du type de conditionnement (ex: 'BT' pour bouteille, 'BB' pour bib).
- AvecCRD [char(1)][nullable]
 - Gestion de la régie ('1' = Avec CRD, '2' = Sans CRD, '3' = Autre).
- **Paramètres de résultat :**
 - TopN [integer][nullable]
 - Nombre maximum de résultats à retourner (Défaut : 100).
 - MinScore [float][nullable]
 - Score de pertinence minimum pour qu'un article soit inclus dans les résultats (Défaut : 0.0).
- **ArticleSearchResponse :**
 - ErrorManagement
 - Articles [] \ Tableau des articles correspondants :
 - CodeArticle [string]
 - NomArticle [string]
 - ScoreGlobal [float]
 - Score de pertinence de 0 à 100. **Les résultats sont triés sur ce champ par ordre décroissant.**
 - Appellation [string]
 - Millesime [string]
 - Couleur [string]
 - Region [string]
 - Domaine [string]
 - Cepage [string]
 - Marque [string]
 - FormatBouteille [string]
 - Degre [float]
 - TypeRegie [string]
 - NbBouteillesCarton [integer]
 - CriteresCorrespondants [integer]
 - Nombre de critères de recherche qui correspondent à cet article.
 - CriteresTotaux [integer]
 - Nombre total de critères envoyés dans la requête.

Explication du système de score

Le ScoreGlobal est une note de pertinence calculée pour chaque article retourné. Il est basé sur la somme des scores de chaque critère fourni, avec une pondération appliquée pour donner plus d'importance à certains champs :

- **Très forte pondération** : CodeArticle
- **Forte pondération** : NomArticle
- **Moyenne pondération** : Appellation

Pour les champs textuels, la recherche utilise un algorithme de similarité de chaîne. Cela signifie qu'une correspondance exacte n'est pas nécessaire ; un nom proche ou contenant les termes recherchés obtiendra un bon score.

Pour les champs numériques comme Millesime ou Degre, le score est plus élevé lorsque la valeur est identique, et diminue progressivement à mesure que la valeur s'en éloigne.

Exemples concrets d'utilisation

1. Recherche simple sur la marque et le cépage

Recherche de tous les vins de la marque "AURORA" faits à partir du cépage "MERLOT".

- **Paramètres envoyés :**
 - Marque: "AURORA"
 - Cepage: "MERLOT"
- **Résultat attendu :**

Une liste d'articles de la marque "AURORA" et du cépage "MERLOT". Les articles qui correspondent aux deux critères auront un ScoreGlobal très élevé. Les articles ne correspondant qu'à un seul des deux critères pourraient également apparaître mais avec un score plus faible.

2. Recherche multi-critères pour un vin rouge précis

Recherche d'un vin rouge de l'appellation "IGP PAYS OC", millésime 2023, avec un degré d'alcool proche de 14.

- **Paramètres envoyés :**
 - Appellation: "IGP PAYS OC"
 - Couleur: "Rouge"
 - Millesime: "2023"
 - Degre: 14.0
- **Résultat attendu :**

Le système retournera en premier les vins rouges de l'appellation "IGP Pays d'Oc" du millésime 2023 avec un degré de 14. Ensuite, les résultats avec des millésimes ou degrés proches (ex: 2022 ou 13.5°) apparaîtront avec un score légèrement inférieur.

5 CLIENTS

5.1 Vérification qu'un client existe (DoesClientExist)

Recherche si un client existe dans SparkWine à l'aide de l'adresse mail

Procédure : q_gs_sWeb_p_check_client_exist

- Paramètres :
 - **TokenManagement**
 - Email [string]
 - Email du client
- DoesClientExistResponse :
 - **ErrorManagement**
 - ClientId [string]
 - Id SparkWine du client
 - Name [string]
 - Nom du client

5.2 Création d'un client (CreateClient)

Création d'un client

Procedure: q_gs_sWeb_p_create_client

- Paramètres :
 - **TokenManagement**
 - WebClientId [string]
 - Id web du client
 - Email [string]
 - Email du client
 - Civility [integer]
 - Civilité
 - Valeur possible : Dépend de l'environnement (Voir avec le client dans SparkWine)
 - FirstName [string]
 - Prénom du client
 - LastName [string]
 - Nom du client
 - BirthDate [date]
 - Date d'anniversaire
 - LastUpdate [date]
 - Date de mise à jour
 - WebCreationDate
 - Date de création sur le WEB
 - SiretNumber [string]
 - Numéro de Siret
 - IntraComVat [string]
 - Numéro tva intracom
 - Gestion Adresse :
 - AddressLine1 [string]
 - Ligne 1 de l'adresse
 - AddressLine2 [string]
 - Ligne 2 de l'adresse
 - Building [string]
 - Infos bâtiment
 - PostalCode [string]
 - Code Postal
 - City [string]
 - Ville
 - Country [string]
 - Pays
 - Valeur possible : Code Pays ISO
 - PhoneNumber [string]
 - Numéro de téléphone (De préférence Portable)
 - IsAdresse [string]
 - 1 si c'est une adresse de facturation sinon 0
- CreateClientResponse :
 - **ErrorManagement**
 - ClientId [string]
 - Id SparkWine du client
 - WebClientId [string]
 - Id web du client

5.3 Création d'un client (UpdateClient)

Procédure: q_gs_sWeb_p_update_client

Mise à jour d'un client

- Paramètres :
 - **TokenManagement**
 - WebClientId [string]
 - Id web du client
 - Email [string]
 - Email du client
 - Civility [integer]
 - Civilité
 - Valeur possible : Dépend de l'environnement (Voir avec le client dans SparkWine)
 - FirstName [string]
 - Prénom du client
 - LastName [string]
 - Nom du client
 - BirthDate [date]
 - Date d'anniversaire
 - LastUpdate [date]
 - Date de mise à jour
 - WebCreationDate
 - Date de création sur le WEB
 - SiretNumber [string]
 - Numéro de Siret
 - IntraComVat [string]
 - Numéro tva intracom
 - Gestion Adresse :
 - AddressLine1 [string]
 - Ligne 1 de l'adresse
 - AddressLine2 [string]
 - Ligne 2 de l'adresse
 - Building [string]
 - Infos bâtiment
 - PostalCode [string]
 - Code Postal
 - City [string]
 - Ville
 - Country [string]
 - Pays
 - Valeur possible : Code Pays ISO
 - PhoneNumber [string]
 - Numéro de téléphone (De préférence Portable)
- UpdateClientResponse :
 - **ErrorManagement**
 - ClientId [string]
 - Id SparkWine du client
 - WebClientId [string]
 - Id web du client

5.4 Récupération d'un client - **Obsolète**

Remplacée par l'utilisation de `GetClients` en spécifiant le code client

Procédure : `q_gs_sweb_p_get_client`

Récupération d'un client :

- Paramètres :
 - **TokenManagement**
 - `WebClientId` [string]
 - Id web du client
 - Type : String
- GetClientResponse :
 - **ErrorManagement**
 - `ClientId` [string]
 - Id SparkWine du client
 - `WebClientId` [string]
 - Id web du client
 - `Name` [string]
 - Nom du client
 - `Email` [string]
 - Email du client
 - `BirthDate` [date]
 - Date d'anniversaire
 - `SiretNumber` [string]
 - Numéro de Siret
 - `IntraComVat` [string]
 - Numéro tva intracom
 - `IsInactive` [string] ➔ 0 ou 1
 - Le client est-il inactif

5.5 Récupération de plusieurs clients (GetClients)

Procédure : q_gs_sweb_p_get_clients

Récupération de plusieurs clients :

- Paramètres :
 - **TokenManagement**
 - WebClientId [string]
 - Id web du client **ou** id **sparkwine**
 - Type : String
 - Ou null pour tout récupérer
- GetClientsResponse :
 - **ErrorManagement**
 - ClientId [string]
 - Id SparkWine du client
 - WebClientId [string]
 - Id web du client
 - Name [string]
 - Nom du client
 - Email [string]
 - Email du client
 - BirthDate [date]
 - Date d'anniversaire
 - SiretNumber [string]
 - Numéro de Siret
 - IntraComVat [string]
 - Numéro tva intracom
 - IsInactive [string] ➔ 0 ou 1
 - Le client est-il inactif
 - WebGroupClient [integer]
 - Groupe Web
 - Adresse [string]
 - Adresse
 - CpltAdresse[string]
 - Complément d'adresse
 - InfosBat [string]
 - Information bâtiment
 - IdRelais [string]
 - ID point relais
 - CP [string]
 - Code postal
 - Commune [string]
 - Nom commune
 - Ville [string]
 - Nom ville
 - PaysCode [string]
 - Code pays
 - PaysTitle [string]
 - Nom pays
 - TVACode [integer]
 - Code TVA
 - TVATaux [float]

- Taux TVA
- TVATitle [string]
 - Nom TVA
- ModeDePaiementCode [string]
 - Code du mode de paiement
- ModeDePaiementTitle [string]
 - Nom du mode de paiement
- CondPaiementCode [string]
 - Code des conditions de paiement
- CondPaiementTitle [string]
 - Nom des conditions de paiement
- CategorieCode [integer]
 - Code catégorie client
- CategorieTitle [string]
 - Nom catégorie client
- EncoursCommande [float]
 - Montant Encours commande
- EncoursComptable [float]
 - Montant Encours comptable
- EncoursTotal [float]
 - Montant encours totaux
- SecteurComCode [string]
 - Code secteur com.
- SecteurComTitle [string]
 - Nom secteur com.
- MarcheComCode [string]
 - Code marché com.
- MarcheComTitle [string]
 - Nom marché com.
- RegionComLib [string]
 - Code région Com.
- RegionComTitle [string]
 - Nom région com
- ModeComCode [integer]
 - Code Mode Com.
- ModeComTitle [string]
 - Nom mode com.
- TelMobile [string]
 - Numéro de téléphone mobile
- TelFixe [string]
 - Numéro de téléphone fixe
- Email [string]
 - Adresse Mail
- CommercialCode [string]
 - Code commercial
- CommercialTitle [string]
 - Nom commercial
- PerRemise [float]
 - Pourcentage remise
- TarifCode [integer]
 - Code Tarif
- TarifTitle [string]

- Nom tarif
- LangueCode [integer]
 - Code langue
- LangueTitle [string]
 - Nom Langue

5.6 Récupération de plusieurs Contacts (GetContacts)

Procédure : q_gs_sweb_p_get_contacts

Récupération de plusieurs Contacts :

- Paramètres :
 - **TokenManagement**
 - WebClientId [string]
 - Id web du client
 - Type : String
 - Ou null pour tout récupérer
- GetContactsResponse :
 - **ErrorManagement**
- ContactCode [Integer]
 - Numéro de contact
- LastName [String]
 - Nom
- FirstName [String]
 - Prenom
- Name [String]
 - Nom complet
- CiviliteCode [Integer]
 - Code Civilité
- CiviliteTitle [String]
 - Nom civilité
- Email [String]
 - Adresse mail
- Mobile [String]
 - Numéro téléphone mobile
- Fixe [String]
 - Numéro téléphone Fixe
- ServiceCode [String]
 - Code service
- ServiceLib [String]
 - Nom service

5.7 Modification mail client (UpdateEmail)

Procédure : q_gs_sweb_p_update_mail_client

- Paramètres :
 - **TokenManagement**
 - ClientId [string]
 - Id SparkWine du client
 - WebClientId [string]
 - Id web du client
 - Email [string]
 - Nouveau email du client
- UpdateEmail Response :
 - **ErrorManagement**
 - ClientId [string]
 - Id SparkWine du client
 - WebClientId [string]
 - Id web du client

5.8 Lier le client web avec le client sparkwine (LinkClient)

Procédure : q_gs_sWeb_p_link_client

- Paramètres :
 - **TokenManagement**
 - ClientId [string]
 - Id SparkWine du client
 - WebClientId [string]
 - Id web du client
- LinkClientResponse :
 - **ErrorManagement**
 - ClientId [string]
 - Id SparkWine du client
 - WebClientId [string]
 - Id web du client

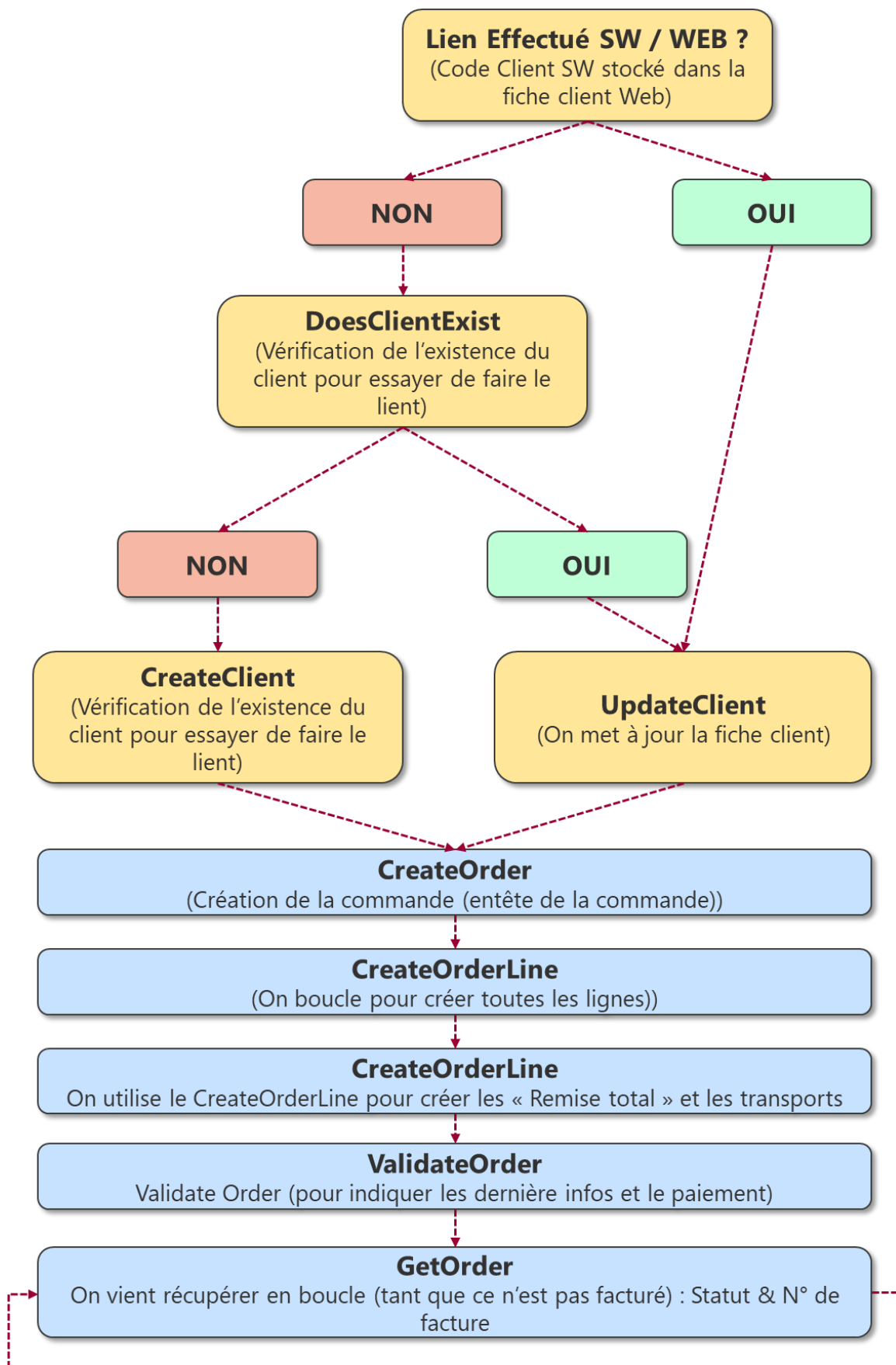
Important :

Le système de création de commande repose sur 3 fonctions à appeler dans l'ordre

1. CreateOrder : Création de l'entête de commande (Une fois par commande)
2. CreateOrderLine : Création ligne de commande
 - a. Autant de fois que nécessaire
 - b. Pareil pour générer un article de remise ou un article de transport
3. ValidateOrder : Quand on a terminé d'insérer des lignes (Une fois par commande)

Les commandes passent par un SAS et donc ne sont pas réellement générées dans l'ERP au moment du Validate Order, la validation peut soit être automatique soit manuelle, mais si elle bloque l'API n'est pas impacté.

6.1 Schéma d'un flux complet classique (Nouvelle Commande)



6.2 Création d'une commande (CreateOrder)

Procédure : q_gs_sweb_p_create_order

Création d'une commande entête de commande

- Paramètres :
 - **TokenManagement**
 - WebOrderId [string]
 - Id web de la commande
 - Type : String
 - WebClientId [string]
 - Id web du client
 - Type : String
 - OrderDate [date]
 - Date de la commande
 - Gestion Adresse :
 - AddressAlias [string]
 - Nom de l'adresse
 - AddressLine1 [string]
 - Ligne 1 de l'adresse
 - AddressLine2 [string]
 - Ligne 2 de l'adresse
 - Building [string]
 - Infos batiment
 - PostalCode [string]
 - Code Postal
 - City [string]
 - Ville
 - Country [string]
 - Pays
 - Valeur possible : Code Pays ISO
 - PhoneNumber [string]
 - Numéro de téléphone (De préférence Portable)
 - DeliveryComment [string]
 - Commentaire de livraison
 - BesoinControle [string]
 - Besoin d'un contrôle pour la commande
 - TexteGeneral [string]
 - Commentaire sur la commande
 - TexteFactProforma [string]
 - Commentaire sur la facture
 - TexteBonPrepa [string]
 - Commentaire sur bon de préparation
 - CodeClientFact [string]
 - Code client facturé
 - CodeClientLivre [string]
 - Code client livré [string]
 - CodeClientPayeur
 - Code client payeur [string]
 - DateLivrSouhai [date]
 - Date de livraison souhaité [date]
 - ComLivraison

- Commentaire livraison
- CritExt_ModeEcheancePaieement
 - Critère Mode Echéance Paiement
- CritExt_ModeReglement
 - Critère Mode Règlement
- CritExt_LivraisonAEffectuer
 - Critère Livraison à effectué
- MontantTotalComissionPrevu
 - Montant total de commission Prévu
- CreateOrderResponse :
 - **ErrorManagement**
 - WebOrderId [string]
 - Id web de la commande
 - OrderNumber [string]
 - Id commande (N° temporaire de la commande, car les commandes

Information complémentaire :

- L'adresse modifiable dans le CreateOrder sert d'adresse de livraison
- Pour l'adresse de facturation / payeur on passe par la création d'un client et on peut ensuite utiliser les champs : CodeClientFact, CodeClientLivre, CodeClientPayeur
- Les remise en valeur global commande, les montant de port, etc... sont créer via une ligne a part avec un article a part qui doit être paramétré dans le système.

6.3 Création d'une ligne de commande (CreateOrderLine)

Procédure : q_gs_sweb_p_create_order_line

Création d'une ligne de commande

- Paramètres :
 - **TokenManagement**
 - WebOrderId [string]
 - Id web de la commande
 - WebArticleId [string]
 - Id de l'article SparkWine
 - UnitPriceET [float]
 - Prix Unitaire HT (avant remise)
 - Complément : Prix brut (Avant remise) → Prix tarif appliqué la ligne
 - UnitPriceIT [float]
 - Prix Unitaire TTC
 - DiscountAmount [float]
 - Montant des remises
 - DiscountPercent [float]
 - Pourcentage de remise
 - Quantity [float]
 - Quantité
 - **Complément** : C'est la quantité total expédié (Gratuit inclus)
 - AmountET [float]
 - Montant HT
 - **Complément** : Montant HT après remise, permet de contrôler que notre calcul de montant a bien la même logique que le système web externe
 - AmountIT [float]
 - Montant TTC
 - PromotionCode [string]
 - Code du coupon
 - PromotionDescription [string]
 - Descriptif du coupon
 - PromotionValidityDate [date]
 - Date de validité
 - UnitAlternative [string]
 - Unité alternative
 - QteGratuit [integer]
 - Quantité gratuite
 - CodeGratuit [integer]
 - Code de gratuité
 - Remise2 [float]
 - Remise en %
 - Remise3 [float]
 - Remise en %
 - RemiseManuelle [float]
 - Remise en valeur
 - TexteExterne [string]
 - Commentaire Externe
 - TexteInterne [string]
 - Commentaire interne

- PrixHorsPort [float]
 - Prix hors port (indicatif pour le moment)
- TauxCommissionPrevu
 - Taux de commission prévu
- MontantTotalComissionPrevu
 - Montant total de commission prévu
- CritExt_Emballage [string]
 - Critère Emballage
- CritExt_Couleurs [string]
 - Critère Couleur
- CritExt_LibArticle [string]
 - Critère Libellé Article

Type de gratuit (Option standard) : Attention chaque environnement peut changer son paramétrage :

- 1 ➔ FIDELITE
- 2 ➔ ECHANTILLON
- 3 ➔ COOP
- 5 ➔ PROMOTION
- 6 ➔ ECHANGE

- CreateOrderLineResponse :
 - **ErrorManagement**
 - WebOrderId [string]
 - Id web de la commande
 - OrderNumber [string]
 - Id commande

6.4 Validation de la commande (ValidateOrder)

Procédure : q_gs_sweb_p_validate_order

Validation de la commande :

- Paramètres :
 - WebOrderId [string]
 - Id web de la commande
 - Type : String
 - AmountET [float]
 - Montant Total HT
 - AmountIT [float]
 - Montant Total TTC
 - PaymentMethod [string]
 - Méthode de paiement
 - **Valeur possible : Dépend de l'environnement (Voir avec le client dans SparkWine)**
 - PaymentId [string]
 - Id de la transaction
 - Carrier [string]
 - Transporteur
 - Valeur Possible :
 - WebAddressId [string]
 - Id de l'adresse de livraison à utiliser
 - ShippingMethod [integer]
 - Mode d'expédition
 - **Valeur possible : Dépend de l'environnement (Voir avec le client dans SparkWine)**
 - ShippingCondition [integer]
 - Condition d'expédition
 - **Valeur possible : Dépend de l'environnement (Voir avec le client dans SparkWine)**
 - Reference [string]
 - Référence de la commande
 - IsMontantPaye [string]
 - 0 ou 1
 - Montant payé différent du total commande
 - MontantPaye
 - Montants payés
- ValidateOrderResponse :
 - WebOrderId [string]
 - Id web de la commande
 - OrderNumber [string]
 - Id commande

6.5 Récupération (GetOrder)

Procédure : q_gs_sweb_p_get_order

Récupération d'informations sur la commande

- Paramètres :
 - **TokenManagement**
 - WebOrderId [string]
 - Id web de la commande
- GetOrderResponse :
 - **ErrorManagement**
 - WebOrderId [string]
 - Id web de la commande
 - OrderNumber [string]
 - Id commande
 - OrderStatus [integer]
 - Statut « ADV » de la commande
 - **Valeur possible : Dépend de l'environnement (Voir avec le client dans SparkWine)**
 - Par Défaut :
 - 10 - ATTENTE VALIDATION
 - 13 - VALIDÉ
 - 20 - EXPEDITION PARTIELLE
 - 40 - EN COURS D'EXPÉDITION
 - 50 - PRET A FACTURER
 - 70 - FACTURÉ
 - 90 - ANNULÉ
 - PackageStatus [string]
 - Statut de préparation
 - **Valeur possible : Dépend de l'environnement (Voir avec le client dans SparkWine)**
 - PackageCount [integer]
 - Nombre de colis / cartons
 - InvoiceNumber [string]
 - Numéro de facture
 - InvoiceDate [date]
 - Date de facture
 - InvoiceDueDate [date]
 - Date d'échéance
 - DateExpedPrev [date]
 - Date d'expédition planifiée dans SparkWine
 - Transporteur [string]
 - Transporteur
 - **Valeur possible : Dépend de l'environnement (Voir avec le client dans SparkWine)**
 - TrackingNumber [string]
 - Tracking number de SparkWine (Pour l'instant le champ est vide)

6.6 DashBoard – Commande (SparkWine)

Le DashBoard est un SAS par lequel les commandes passent avant d'entrer comme des commandes réelles.

Il se compose d'une liste avec les commandes en attente de validation :

Commande [5]	Code Web	N° Cde	Etat Commande	Code tiers	Lib Tiers	Con	Montant HT	Montant TTC	Lib Tiers	Client livré	Création client	Web client id
▶ WEB-ORDER-7757			VALIDÉ	10083267		☐	758,33	910,00	DE SCHENKER		20/11/2021	WEB-CLIENT-5508
WEB-ORDER-7815			VALIDÉ	10083337		☐	75,83	80,00	DAUMAS PRODUCTEUR		25/11/2021	WEB-CLIENT-5529
WEB-ORDER-7820			VALIDÉ	10062563		☐	75,83	80,00			20/09/2018	WEB-CLIENT-5534
WEB-ORDER-7823			VALIDÉ	10036928		☐	151,66	160,00	DE SCHENKER		19/07/2018	WEB-CLIENT-904
WEB-ORDER-7825			VALIDÉ	10023385		☐	75,83	80,00	DE SCHENKER		19/07/2018	WEB-CLIENT-4785

Ensuite pour chaque ligne on va donc pouvoir consulter et valider les commandes : (La notion de validation va donc créer une commande SparkWine Réel)

Général | Synthèse | Tous les champs

Infos générales

Code web
WEB-ORDER-7815

N° pré-commande
25/11/2021 22:00:03:637206492

Commande validée ?
☐

Infos commande

N° Commande
[]

Date Commande
25/11/2021

Type Commande
VENTE PAR INTERNET

Mode de paiement
CARTE INTERNET CA

Conditions paiement
COMPTANT

N° campagne
PRIVE 2021

% Remise Commande
0,00

Remise en Valeur
0,00

Montant HT
75,83

Montant TTC
80,00

Date expéd. prévue
25/11/2021

Infos client

Entreprise
10083337

Libellé Tiers
[]

Création client
25/11/2021 [Changer client](#)

Web client id
WEB-CLIENT-5529

Langue
FRANÇAIS

Infos livraison

Nom destinataire
[]

Adresse livraison 1
[]

Adresse livraison 2
[]

Informations bâtiment
[]

CP/VILLE
[]

Pays de livraison
[]

Tél. mobile
[]

Em@il
[]

Transport

Produit Transporteur
[] MANQUANT [] : 01

Mode d'expédition
ROUTE

Conditions Expédition
PORT PAYE

Transporteur
DAUMAS PRODUCTEUR

Notes

Erreurs

Retraitement Fait

Actions

[Valider](#)

Ligne de commande [2]

Code article	Lib. Article	Quantité	Prix HT	Prix TTC	% Rem.	Monta
▶ PREST-009	SAUMON FUME BIO	1	75,83	80,00	0,00	
PORT	FRAIS DE PORT	1	16,67	20,00	0,00	

Mise à jour client

DteCréa	Type mise	Validé?	Code tiers	Lib Tiers
▶		☐		

7.1 References (GetReferences)

Procédure : q_gs_sWeb_p_get_references

Récupération d'informations sur la commande

- Paramètres :
 - **TokenManagement**
 - Reference [string]
 - Si Null ou "" on renvoie toutes les références
 - Sinon on peut filtrer avec Transporteur, ModePaiement, ModeExpedition
- GetOrderResponse :
 - **ErrorManagement**
 - Type [string]
 - Type de référence
 - Code [string]
 - Code de la reference
 - Libelle [integer]
 - Libellé de la référence
 - DateCreation [string]
 - Date de création
 - DateModification [integer]
 - Date de Modification

8 GESTION DES ERREURS

Code	Message
400	Token non reconnu
500	Erreur Interne
501	L'article n'existe pas
510	Le client existe déjà
511	Le client n'existe pas
512	L'adresse n'existe pas
513	L'adresse n'existe pas
520	La commande existe déjà
521	La commande n'existe pas