

Documentation technique

Pro à Pro - Consultation des fichiers priceIQ - 2024



Table des matières

I. Introduction.....	2
A. Entreprise.....	2
B. Objectif.....	2
C. Vue d'ensemble.....	2
II. Architecture Global.....	4
A. Positionnement de la fonctionnalité.....	4
B. UML.....	5
III. Backend (Côté Serveur).....	6
A. Utilisation de fonction.....	6
1. Vérification du token.....	6
2. Vérification de la tâche.....	6
B. Nouveaux Web Services.....	7
1. Tarifs PricelQ.....	7
2. Corridors.....	7
3. Client Segment.....	7
4. Corridors de prix.....	7
5. Table de pricelQ.....	8
6. Valeurs par défaut.....	8
IV. Frontend (Côté Client).....	9
A. Langages et technologies.....	9
B. AJAX.....	9
1. Tarifs PricelQ.....	9
2. Corridors.....	10
3. Segment client.....	10
4. Corridors de prix.....	11
5. Table de pricelQ.....	12
6. Valeurs par défaut.....	13
V. Annexes.....	14
A. Tarifs PricelQ.....	14
1. Contrôleur.....	14
B. Corridors.....	15
1. Contrôleur.....	15
C. Client segment.....	16
1. Contrôleur.....	16
D. Corridors de prix.....	17
1. Contrôleur.....	17
E. Table de PricelQ.....	17
1. Contrôleur.....	17
F. Valeurs par défaut.....	18
1. Contrôleur.....	18

I. Introduction

A. Entreprise

Pro à Pro est une entreprise française d'environ 1 000 salariés. Elle est spécialisée dans la distribution de produits alimentaires frais et de qualité. Avec son réseau de logistique avancé, Pro à Pro livre ses clients en 24h.

B. Objectif

L'objectif de la fonctionnalité est de pouvoir consulter les corridors de prix des différents fichiers venant de priceIQ. La fonctionnalité permet de trier les éléments de chaque fichier.

C. Vue d'ensemble

The image displays three screenshots of a web application interface, each showing a different view with a table and filters.

1. Tarifs PriceIQ

- Header: Afficher 10 éléments, Rechercher: [input], Limite de résultat: 10
- Table columns: Agence, Article, Indice Prix, Quantité mini, Date début, Date fin, Prix net
- Table content: Aucune donnée disponible dans le tableau
- Footer: Affichage de 0 à 0 sur 0 éléments, Précédent, Suiv
- Filters: Code article (1), Code article [input], Indice tarifaire (1), Sélectionner... [input], Effacer Tout, Filtrer

2. Consultation des corridors

- Header: Afficher 10 éléments, Rechercher: [input], Limite de résultat: 100
- Table columns: ACTION, CUSTOMER_TYPE, ACTIVITE_CLIENT, REGION, CUSTOMER_SPEND_BIN, CUSTOMER_DELIVERY_FREQUENCY_BIN, CLIENT_NATIONAL
- Table content: Aucune donnée disponible dans le tableau
- Footer: Affichage de 0 à 0 sur 0 éléments, Précédent, Suiv
- Filters: Region (1), Sélectionner... [input], Catégorie de client, Actifs client, Sélectionner... [input], Paramètres avancés: customer-spend-bin, Sélectionner... [input], customer-delivery-frequency-bin, Sélectionner... [input], client-national, Sélectionner... [input], Effacer Tout, Filtrer

3. Consultation des clients affectés à un segment

- Header: Afficher 10 éléments, Rechercher: [input], Limite de résultat: 100
- Table columns: ACTION, CODE_CLIENT, CLINOM
- Table content: Aucune donnée disponible dans le tableau
- Footer: Affichage de 0 à 0 sur 0 éléments, Précédent, Suiv
- Filters: ID Client, Sélectionner... [input], Effacer Tout, Filtrer

Consultation des corridors de prix

Afficher 10 éléments

Rechercher:

ACTION	PRODUCT_ID	START_PRICE	TARGET_PRICE	FLOOR_PRICE
Aucune donnée disponible dans le tableau				

Affichage de 0 à 0 sur 0 éléments

Précédent Suiv

Filtres

Limite de résultat : 100

ID Produit

Effacer tout Filter

Tables issues de PriceIQ

Table : Aucune table sélectionnée...

Filtres

Limite de résultat : 100

Tables*

Selectionner...

Effacer tout Filter

Valeurs par défaut

Afficher 10 éléments

Rechercher:

Element	Valeur par défaut	Date et Heure Modif
Aucune donnée disponible dans le tableau		

Affichage de 0 à 0 sur 0 éléments

Précédent Suiv

Filtres

Limite de résultat : 10

Element

Selectionner...

Effacer tout Filter

Consultation des erreurs

Table : Aucune table sélectionnée...

Filtres

Limite de résultat : 100

Tables*

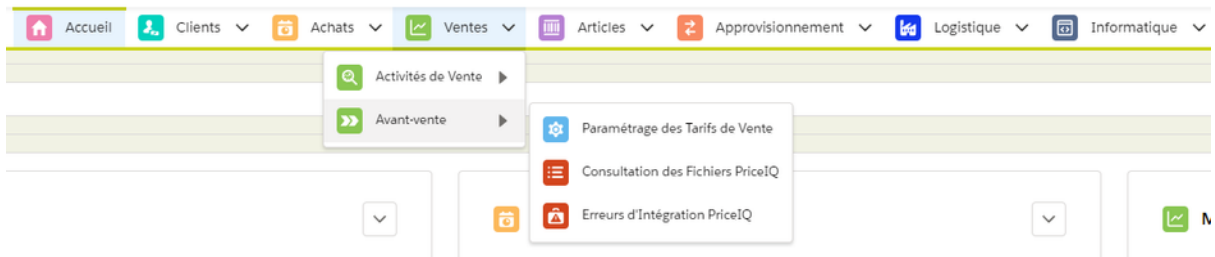
Selectionner...

Effacer tout Filter

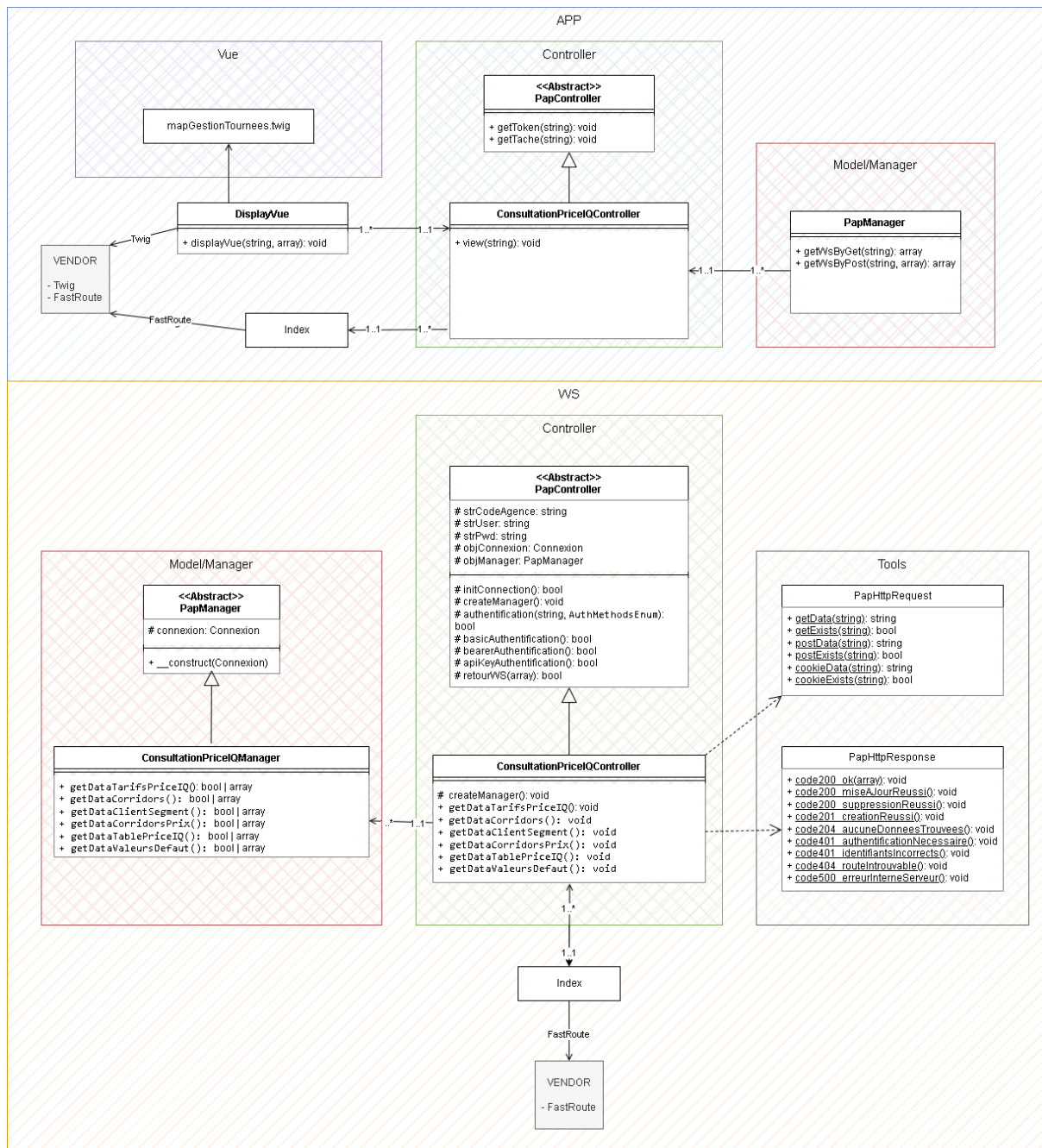
II. Architecture Global

A. Positionnement de la fonctionnalité

La fonctionnalité se trouve dans l'application web E-Gescom. Dans la catégorie "Ventes > Avant-vente > **Consultation des Fichiers PriceIQ**". On peut y accéder grâce à la barre de navigation ou par la page d'accueil. Pour pouvoir accéder à cette fonctionnalité il faut avoir la tâche "GESWPRIQ".



B. UML



III. Backend (Côté Serveur)

A. Utilisation de fonction

1. Vérification du token

- **Nom** : *checkToken*
- **Description** : Vérifier si l'utilisateur est bien connecté avant de lui afficher la page.
- **Classe** : *PapController*
- **Paramètre** : *Aucun*
- **Retour** : *Aucun*
- **Exemple** : `$this->checkToken();`
- **Code** :

```
public function checkToken(): void
{
    if (empty($_SESSION["token"])) {
        header("Location: /app/");
    }
}
```

2. Vérification de la tâche

- **Nom** : *checkTache*
- **Description** : Vérifier si l'utilisateur a les droits pour pouvoir accéder à la page.
- **Classe** : *PapController*
- **Paramètre** :
 - Nom de la tâche → **Obligatoire**
- **Retour** : *Aucun*
- **Exemple** : `$this->checkTache("GESWPLREC");`
- **Code** :

```
public function checkTache(string $tache): void
{
    $tache = strtoupper($tache);
    $agence = $_GET["agence"];
    if (array_key_exists($tache, $_SESSION["Tache_Menu_Dashboard"]) &&
        ($_SESSION["Tache_Menu_Dashboard"][$tache]["ACCES"] != "0")) {
        header("Location: /app/dashboard?agence=" . $agence);
        exit();
    }
}
```

B. Nouveaux Web Services

1. Tarifs PriceIQ

- **Description** : Permet de récupérer les tarifs de priceIQ.
- **Route** :
 - **API** : POST
 - **Paramètres** :
 - Code article
 - Indice tarifaire
 - **URL** :
`/ws/ventes/avant-ventes/consultation-fichiers-priceiq/{agence}/tarifs-priceiq`
- **Code** :
 - **Contrôleur** : [ConsultationPriceIQController](#)

2. Corridors

- **Description** : Permet de récupérer les corridors de priceIQ.
- **Route** :
 - **API** : POST
 - **Paramètre** :
 - Région → **Obligatoire**
 - Catégorie
 - Customer Spend Bin
 - Customer Delivery Frequency Bin
 - Client National
 - **URL** :
`/ws/ventes/avant-ventes/consultation-fichiers-priceiq/{agence}/corridors`
- **Code** :
 - **Contrôleur** : [ConsultationPriceIQController](#)

3. Client Segment

- **Description** : Permet de récupérer les segments clients.
- **Route** :
 - **API** : POST
 - **Paramètres** :
 - Code client
 - **URL** :
`/ws/ventes/avant-ventes/consultation-fichiers-priceiq/{agence}/client-segment`
- **Code** :
 - **Contrôleur** : [ConsultationPriceIQController](#)

4. Corridors de prix

- **Description** : Permet de récupérer les corridors de prix.
- **Route** :
 - **API** : POST
 - **Paramètres** :

- Code article
- **URL :**
/ws/ventes/avant-ventes/consultation-fichiers-priceiq/{agence}/corridors-prix
- **Code :**
 - **Contrôleur :** [*ConsultationPriceIQController*](#)

5. Table de priceIQ

- **Description :** *Permet de récupérer données des différentes tables de priceIQ.*
- **Route :**
 - **API :** *POST*
 - **Paramètres :**
 - Table → **Obligatoire**
 - **URL :**
/ws/ventes/avant-ventes/consultation-fichiers-priceiq/{agence}/table-priceiq
- **Code :**
 - **Contrôleur :** [*ConsultationPriceIQController*](#)

6. Valeurs par défaut

- **Description :** *Permet de récupérer les valeurs par défaut.*
- **Route :**
 - **API :** *POST*
 - **Paramètres :**
 - Element → **Obligatoire**
 - **URL :**
/ws/ventes/avant-ventes/consultation-fichiers-priceiq/{agence}/valeurs-defaut
- **Code :**
 - **Contrôleur :** [*ConsultationPriceIQController*](#)

IV. Frontend (Côté Client)

A. Langages et technologies

- **TWIG** : Moteur de rendu permettant de construire une page avec des templates.
- **Javascript** : Langage de programmation côté client.
 - **JQuery** : Faciliter le développement javascript.
 - **Datatable** : Construction de tableau dynamique.
- **HTML** : Langage de balisage permettant de construire une page web.
- **CSS** : Langage permettant d'ajouter du style à la page.

B. AJAX

1. Tarifs PriceIQ

- **Nom** : `getDataTarifsPriceIQ`
- **Paramètres** :
 - **Article** : *String*
 - **Indice** : *String*
 - **Limite** : *Integer*
- **Méthode** : *POST*
- **Type** : *JSON*
- **Code** :

```
function getDataTarifsPriceIQ(article, indice, limite) {
    let data;
    $.ajax({
        async: false,
        type: "POST",
        url:
            "/ws/ventes/avant-ventes/consultation-fichiers-priceiq/{ { agence
            }}/tarifs-priceiq",
        data: {
            article: article,
            indice: indice,
            limite: limite
        },
        dataType: "json",
        success: function (result) {
            data = result;
        },
        error: function (req, err) {
            console.log("Error : " + err);
        }
    });
    return data;
}
```

2. Corridors

- **Nom** : *getDataCorridors*
- **Paramètres** :
 - **Region** : *String*
 - **Categorie** : *String*
 - **Customer Spend Bin** : *String*
 - **Customer Delivery Frequency Bin** : *String*
 - **Client National** : *String*
- **Méthode** : *POST*
- **Type** : *JSON*
- **Code** :

```
function getDataCorridors(region, categorie, customerSpendBin,
CustomerDeliveryFrequencyBin, clientNational) {
    let data;
    $.ajax({
        async: false,
        type: "POST",
        url:
"/ws/ventes/avant-ventes/consultation-fichiers-priceiq/{ { agence
}}/corridors",
        data: {
            region: region,
            categorie: categorie,
            customerSpendBin: customerSpendBin,
            CustomerDeliveryFrequencyBin:
CustomerDeliveryFrequencyBin,
            clientNational: clientNational
        },
        dataType: "json",
        success: function (result) {
            data = result;
        },
        error: function (req, err) {
            console.log("Error : " + err);
        }
    });
    return data;
}
```

3. Segment client

- **Nom** : *getDataClientSegment*
- **Paramètres** :
 - **Client** : *String*
- **Méthode** : *POST*

- **Type :** *JSON*
- **Code :**

```
function getDataClientSegment(client) {
    let data;
    $.ajax({
        async: false,
        type: "POST",
        url:
"/ws/ventes/avant-ventes/consultation-fichiers-priceiq/{ { agence
}}/client-segment",
        data: {
            client: client
        },
        dataType: "json",
        success: function (result) {
            data = result;
        },
        error: function (req, err) {
            console.log("Error : " + err);
        }
    });
    return data;
}
```

4. Corridors de prix

- **Nom :** *getDataCorridorsPrix*
- **Paramètres :**
 - **Article :** *String*
- **Méthode :** *POST*
- **Type :** *JSON*
- **Code :**

```
function getDataCorridorsPrix(article) {
    let data;
    $.ajax({
        async: false,
        type: "POST",
        url:
"/ws/ventes/avant-ventes/consultation-fichiers-priceiq/{ { agence
}}/corridors-prix",
        data: {
            article: article
        },
    },
```

```

        dataType: "json",
        success: function (result) {
            data = result;
        },
        error: function (req, err) {
            console.log("Error : " + err);
        }
    });
    return data;
}

```

5. Table de priceIQ

- **Nom** : *getDataTablePriceIQ*
- **Paramètres** :
 - **Table** : *String*
- **Méthode** : *POST*
- **Type** : *JSON*
- **Code** :

```

function getDataTablePriceIQ(table) {
    let data;
    $.ajax({
        async: false,
        type: "POST",
        url:
"/ws/ventes/avant-ventes/consultation-fichiers-priceiq/{ { agence
}}/table-priceiq",
        data: {
            table: table
        },
        dataType: "json",
        success: function (result) {
            data = result;
        },
        error: function (req, err) {
            console.log("Error : " + err);
        }
    });
    return data;
}

```

6. Valeurs par défaut

- **Nom** : *getDataValeursDefaut*
- **Paramètres** :
 - **Element** : *String*
- **Méthode** : *POST*
- **Type** : *JSON*
- **Code** :

```
function getDataValeursDefaut(element) {
    let data;
    $.ajax({
        async: false,
        type: "POST",
        url:
"/ws/ventes/avant-ventes/consultation-fichiers-priceiq/{ { agence
}}/valeurs-defaut",
        data: {
            element: element
        },
        dataType: "json",
        success: function (result) {
            data = result;
        },
        error: function (req, err) {
            console.log("Error : " + err);
        }
    });
    return data;
}
```

V. Annexes

A. Tarifs PriceIQ

1. Contrôleur

```
public function getDataTarifsPriceIQ(String $agence): void
{
    $connexionReussi = $this->authentification($agence,
AuthMethodsEnum::BEARER);
    if (!$connexionReussi) {
        return;
    }

    $limite = PapHttpRequest::getData("limite");
    $article = PapHttpRequest::getData("article");
    $indice = PapHttpRequest::getData("indice");

    $params = [];

    if (!empty($article)) {
        $params["article"] = $article;
    }

    if (!empty($indice)) {
        $params["indice"] = $indice;
    }

    $params["limite"] = $limite;

    $this->createManager();
    $data = $this->_objManager->getDataTarifsPriceIQ($params)

    if ($data === false) {
        PapHttpResponse::code500_erreurInterneServeur();
    } elseif (!empty($data)) {
        PapHttpResponse::code200_ok($data);
    } else {
        PapHttpResponse::code204_aucuneDonneesTrouvees();
    }
}
```

B. Corridors

1. Contrôleur

```
public function getDataCorridors(String $agence): void
{
    $connexionReussi = $this->authentification($agence,
AuthMethodsEnum::BEARER);
    if (!$connexionReussi) {
        return;
    }

    $limite = PapHttpRequest::getData("limite");
    $region = PapHttpRequest::getData("region");
    $categorie = PapHttpRequest::getData("categorie");
    $customerSpendBin = PapHttpRequest::getData("customerSpendBin");
    $customerDeliveryFrequencyBin =
PapHttpRequest::getData("customerDeliveryFrequencyBin");
    $clientNational = PapHttpRequest::getData("clientNational");

    $params = [
        "region" => $region;
    ];

    if (!empty($categorie)) {
        $params["categorie"] = $categorie;
    }

    if (!empty($customerSpendBin)) {
        $params["customerSpendBin"] = $customerSpendBin;
    }

    if (!empty($customerDeliveryFrequencyBin)) {
        $params["customerDeliveryFrequencyBin"] =
$customerDeliveryFrequencyBin;
    }

    if (!empty($clientNational)) {
        $params["clientNational"] = $clientNational;
    }

    $params["limite"] = $limite;

    $this->createManager();
    $data = $this->_objManager->getDataCorridors($params)
```



```

    if ($data === false) {
        PapHttpResponse::code500_erreurInterneServeur();
    } elseif (!empty($data)) {
        PapHttpResponse::code200_ok($data);
    } else {
        PapHttpResponse::code204_aucuneDonneesTrouvees();
    }
}

```

C. Client segment

1. Contrôleur

```

public function getDataClientSegment(String $agence): void
{
    $connexionReussi = $this->authentication($agence,
AuthMethodsEnum::BEARER);
    if (!$connexionReussi) {
        return;
    }

    $limite = PapHttpRequest::getData("limite");
    $client = PapHttpRequest::getData("client");

    $params = [];

    if (!empty($client)) {
        $params["client"] = $client;
    }

    $params["limite"] = $limite;

    $this->createManager();
    $data = $this->_objManager->getDataClientSegment($params)

    if ($data === false) {
        PapHttpResponse::code500_erreurInterneServeur();
    } elseif (!empty($data)) {
        PapHttpResponse::code200_ok($data);
    } else {
        PapHttpResponse::code204_aucuneDonneesTrouvees();
    }
}

```

D. Corridors de prix

1. Contrôleur

```
public function getDataCorridorsPrix(String $agence): void
{
    $connexionReussi = $this->authentification($agence,
AuthMethodsEnum::BEARER);
    if (!$connexionReussi) {
        return;
    }

    $limite = PapHttpRequest::getData("limite");
    $article = PapHttpRequest::getData("article");

    $params = [];

    if (!empty($article)) {
        $params["article"] = $article;
    }

    $params["limite"] = $limite;

    $this->createManager();
    $data = $this->_objManager->getDataCorridorsPrix($params)

    if ($data === false) {
        PapHttpResponse::code500_erreurInterneServeur();
    } elseif (!empty($data)) {
        PapHttpResponse::code200_ok($data);
    } else {
        PapHttpResponse::code204_aucuneDonneesTrouvees();
    }
}
```

E. Table de PriceIQ

1. Contrôleur

```
public function getDataTablePriceIQ(String $agence): void
{
    $connexionReussi = $this->authentification($agence,
AuthMethodsEnum::BEARER);
    if (!$connexionReussi) {
```

```

        return;
    }

    $limite = PapHttpRequest::getData("limite");
    $table = PapHttpRequest::getData("table");

    $params = [
        "table" => $table
    ];

    $params["limite"] = $limite;

    $this->createManager();
    $data = $this->_objManager->getDataTablePriceIQ($params)

    if ($data === false) {
        PapHttpResponse::code500_erreurInterneServeur();
    } elseif (!empty($data)) {
        PapHttpResponse::code200_ok($data);
    } else {
        PapHttpResponse::code204_aucuneDonneesTrouvees();
    }
}

```

F. Valeurs par défaut

1. Contrôleur

```

public function getDataValeursDefaut(String $agence): void
{
    $connexionReussi = $this->authentification($agence,
    AuthMethodsEnum::BEARER);
    if (!$connexionReussi) {
        return;
    }

    $limite = PapHttpRequest::getData("limite");
    $element = PapHttpRequest::getData("element");

    $params = [
        "element" => $element
    ];

    $params["limite"] = $limite;
}

```

```
$this->createManager();
$data = $this->_objManager->getDataValeursDefaut($params)

if ($data === false) {
    PapHttpResponse::code500_erreurInterneServeur();
} elseif (!empty($data)) {
    PapHttpResponse::code200_ok($data);
} else {
    PapHttpResponse::code204_aucuneDonneesTrouvees();
}

}
```