

Guide utilisateur

Web service REST presenceRegistration

Check In and Out at Work

Version	Date	Nature des modification
1.0	01/03/2023	Première version
1.1	03/06/2024	Réécriture des explications concernant OAuth
1.2	09/08/2024	Intégration des modification de la release 1.4
1.3	19/09/2025	Ajout d'une explication concernant l' "erreur 400", avec une référence à la page https://www.rest-documentation.socialsecurity.be/ pour l'accès au WS presenceregistration

Contenu

Enregistrement des présences dans le secteur du nettoyage	3
Service SOAP et service REST.....	4
Préparation.....	5
Swagger	6
Principe de fonctionnement.....	7
Environnements	7
Editeurs de software	7
Asynchronisme	8
Déroulement classique.....	8
Création	8
Consultation.....	10
Délais de traitement	10
Documentation technique.....	14
RegisterInBulk.....	14
Description.....	14
Requête.....	14
Réponse	15
Exemple	16
Read by id.....	19
Description.....	19
Requête.....	19
Réponse	20
Exemple	22
Search.....	24
Description.....	24
Requête.....	24
Autres paramètres	26
Réponse	26
Exemple	27

Enregistrement des présences dans le secteur du nettoyage

Conformément à la loi programme du 26/12/2022, les travailleurs effectuant des activités spécifiques de nettoyage doivent obligatoirement enregistrer le début (IN) et la fin (OUT) de leurs prestations, ainsi que les périodes de repos.

Un enregistrement de présence représente l'entrée ou la sortie d'un employé sur son lieu de travail. Il existe donc 2 types de présence :

- **IN** : l'entrée du travailleur (mais également la fin d'une pause)
- **OUT** : la sortie du travailleur (mais également le début d'une pause)

Un enregistrement de présence typique regroupe les informations suivantes :

- **QUI** ? Qui est entré ou sorti du bâtiment ? Cette information est représentée par le numéro d'identification à la sécurité sociale (NISS) du travailleur.
- **OÙ** ? Sur quel lieu de travail le travailleur est-il entré ou d'où est-il sorti ? Cette information est représentée soit par les coordonnées de géolocalisation du travailleur au moment du 'check in' (entrée) ou du 'check out' (sortie), ou d'une description de l'adresse au format texte.
- **QUAND** ? À quelle heure le travailleur est-il entré ou sorti ? Cette information est représentée par une date et une heure, auxquelles le travailleur a effectué son check in ou son check out.

Check In and Out at Work (parfois abrégé 'ClaO') est le service en ligne de suivi des prestations des travailleurs des différents secteurs concernés. Elle permet la création et la consultation de 'presence registrations' (enregistrements de présences).

Deux méthodes permettent l'enregistrement des présences :

- enregistrement à l'aide d'un smartphone, et
- appel à un web service REST (couplé à votre éventuel système de badge existant), appelé aussi **PresenceRegistration**.

Ce guide utilisateur décrit les aspects fonctionnels du web service REST.

Service SOAP et service REST

Le portail de la sécurité sociale propose deux services avec le nom **PresenceRegistration**:

- l'un utilise la technologie SOAP, et
- l'autre utilise la technologie REST.

Il s'agit de deux web services différents, destinés à des secteurs particuliers :

- Les secteurs de la **construction**, de la **viande** et du **gardiennage** doivent utiliser le service **SOAP**. L'enregistrement de présence se fait une fois par jour, par personne et par déclaration de travail.
- Le secteur du **nettoyage** doit utiliser le service **REST**. L'enregistrement de présence se fait à chaque entrée et sortie du travailleur, et également à chaque pause pour chaque lieu de travail.

Au sein d'un même endroit, il peut y avoir des travailleurs qui doivent utiliser le service REST plusieurs fois par jour et d'autres qui doivent utiliser le service SOAP une fois par jour. Par exemple, dans le cadre d'un chantier de construction qui inclut des prestations de nettoyage.

D'autres secteurs seront ajoutés au cours de l'année 2024 et 2025. Ils devront également utiliser le service REST.

Comme mentionné ci-dessus, ce guide utilisateur décrit uniquement le service REST.

Pour le service SOAP, rendez-vous sur la page ['Comment enregistrer les présences ?' de Checkinetwork sur le portail de la sécurité sociale](#).

Préparation

Avant d'accéder au service REST **PresenceRegistration**, vous devez effectuer quelques étapes préalables. Ces étapes sont décrites en détail sur le portail de la sécurité sociale et sont accessibles via le lien suivant:

<https://www.rest-documentation.socialsecurity.be/>

Si vous avez déjà utilisé un service web de la sécurité sociale, il est possible que certaines de ces étapes aient déjà été complétées.

Concrètement, il s'agit de :

- s'identifier en tant qu'employeur ;
- demander ou créer un certificat ;
- créer un compte de service web dans « Channel Management ».

Pour la création d'un compte de service web dans « Channel Management », suivez les étapes décrites dans l'URL suivante : [Activating and managing your REST channel in Chaman | REST Documentation](#).

Pour ClaO, vous devez :

- créer un compte de service web de type 'REST' ;
- sélectionner l'autorisation "Check in And Out at Work" (voir capture d'écran ci-dessous).

Ajouter un compte Webservice

Type*
REST

Nom du compte*
My ClaO account

Permissions sécurisées

☐ CareerPro - Federal Learning Account - Déclaration	☒ Check In And Out @ Work
☐ Consultation de la déclaration Dimona	☐ Effectuer des déclarations Dimona
☐ WITA Amateur - Entreprise	

Gestion du certificat

Certificat* Nom du certificat*

Valider Annuler

Swagger

Le Swagger est un fichier au format YAML, un standard de représentation de données. Le Swagger de PresenceRegistration décrit les méthodes et les zones. Vous le trouverez sur le portail de la sécurité sociale, en bas de la page suivante :

https://www.socialsecurity.be/site_fr/employer/applics/check-in-and-out-at-work/general/how-to-register.htm

Principe de fonctionnement

L'ONSS met à disposition des employeurs un web service, leur permettant :

- d'enregistrer les présences de leurs travailleurs dans ClaO.
- de consulter les présences de leur travailleurs, et de consulter les éventuelles remarques.

Environnements

L'ONSS met à disposition 2 environnements dont voici les endpoints :

Environnement	Adresse
Simulation	https://services-sim.socialsecurity.be/REST/presenceRegistration/v1
Production	https://services.socialsecurity.be/REST/presenceRegistration/v1

Vos tests doivent être réalisés sur l'environnement de Simulation. Concernant Chaman et OAuth, l'environnement de Production est utilisé à la fois pour la Simulation et la Production du service REST PresenceRegistration.

L'environnement de Production de PresenceRegistration ne doit contenir que des données de prestations réellement effectuées.

Editeurs de software

Cette partie s'adresse aux éditeurs de software qui souhaitent proposer leurs services aux entreprises concernées par le Check In & Out at Work.

Votre service consiste-t-il uniquement à envoyer des enregistrements vers la Sécurité Sociale (par exemple, vous gérez une badgeuse) ?

Dans ce cas, vous pouvez utiliser votre propre certificat.

Souhaitez-vous offrir des fonctionnalités avancées impliquant la consultation des enregistrements et de leurs anomalies ?

Dans ce cas, vous devrez utiliser le certificat de votre client.

Vous devrez assister votre client pour la création d'un certificat self-signed, et l'envoi de celui-ci sur Chaman.

Ensuite, vous pourrez installer votre logiciel sur une machine appartenant au client.

Asynchronisme

Il convient de souligner que l'ONSS ne peut pas assurer le traitement synchrone des remarques liées aux enregistrements de présences. En effet, le traitement des présences implique des appels à d'autres web services, rendant dès lors le traitement des remarques asynchrones.

Par conséquent, les web services ClaO décrivent deux opérations distinctes :

- la création de présences, et
- la consultation de présences.

Déroulement classique

Création

- L'utilisateur doit utiliser l'endpoint de création afin de soumettre ses présences.
([/presenceRegistrations/registerInBulk](#))
- Il reçoit ensuite l'un des 2 codes suivants :
 - **200** : dans ce cas, la réponse comporte autant d'objets que d'enregistrements de présence créés par l'utilisateur. Pour chaque enregistrement, un objet associé présente deux propriétés mutuellement exclusives :
 - **createdPresenceRegistration** : cette propriété décrit l'enregistrement de présence qui a été créé. Il est à noter que la propriété 'remarks' est vide. En effet, comme mentionné précédemment, le traitement des remarques est asynchrone. Par conséquent, lors de la création de l'enregistrement de présence dans ClaO, les remarques ne sont pas encore calculées. Si cette propriété est nulle, cela indique qu'une erreur s'est produite lors de l'enregistrement de la présence. Cette erreur est décrite dans la propriété '[notCreatedPresenceRegistration](#)'.
 - **notCreatedPresenceRegistration** : si cette propriété n'est pas nulle, cela indique qu'une erreur s'est produite lors de l'enregistrement de la présence. Le problème est décrit dans les propriétés de cet objet :
 - '[presenceRegistrationSubmitted](#)' : comme son nom l'indique, cette propriété reprend l'enregistrement de présence telle qu'il a été soumis par l'utilisateur. Cette présence contient donc une (ou des) erreur(s), décrite(s) dans la propriété '[errorList](#)'
 - '[errorList](#)' : contient une liste d'erreurs décrivant les raisons pour lesquelles l'enregistrement n'a pas pu être créé. Il convient dès lors de corriger les erreurs mentionnées, et de soumettre à nouveau **uniquement les présences en erreur**.

o **400, 401, 403 ...** : Le serveur ne peut pas traiter la requête. La cause provient du **client**.

Deux raisons possibles:

- **Non-conformité avec l'openAPI.** Quelques exemples:
 - Le numéro NISS n'a pas le format correct.
 - Exemple de code d'erreur lorsqu'un « 9 » est ajouté à la fin d'un NISS:
`"[Path '/items/0/ssin'] ECMA 262 regex \"^\\d{11}$\" does not match input string \"904101963209\""`
 - Signification : un numéro NISS doit toujours comporter exactement 11 caractères (NB : dans cet exemple, il y en a 12).
 - Le numéro DDT n'a pas le format correct.
 - Exemple de code d'erreur lorsqu'on ajoute deux tirets au numéro DDT:
`"[Path '/items/0/contractualRelationshipReference'] ECMA 262 regex \"^[A-HJ-NP-Z0-9]{13}$\" does not match input string \"1Y1-002W0ZVMG-Z\""`
 - Signification : un numéro DDT doit toujours comporter exactement 13 caractères (NB : dans cet exemple, il y en a 15).
 - Le numéro d'entreprise n'a pas le bon format.
 - Exemple de numéro d'entreprise où il manque un « 0 » au début:
`"[Path '/items/0/employer/enterpriseNumber'] ECMA 262 regex \"^[0|1]\\d{9}$\" does not match input string \"406798006\""`
 - Signification : un numéro d'entreprise commence toujours par un « 0 » ou un « 1 », suivi de 9 chiffres.
 - Le type d'enregistrement (in ou out) est manquant.
 - Exemple de code d'erreur:
`"[Path '/items/0'] Object has missing required properties ([\"type\"])"`
- **Les règles de sécurité ne sont pas respectées** (problème d'authentification/autorisation).
 - Quelques exemples :
 - "401 unauthorized"
 - "403 forbidden"
 - Plus d'informations disponibles ici: [Additional guidelines for calling REST APIs | REST Documentation](#)
- **500** : une erreur non gérée a eu lieu, les enregistrements de présence n'ont pas été créés.

Consultation

L'utilisateur dispose de deux méthodes afin de consulter les enregistrements de présences :

- Sur la base d'un **ID**, afin d'obtenir les informations d'une présence en particulier (</presenceRegistrations/{id}>)
- Sur la base de **critères de recherche**, afin de lister la ou les présences qui correspondent aux critères mentionnés (</presenceRegistration/search>)

Un utilisateur utilisant les web services sera toujours associé à l'employeur propriétaire du certificat. Dès lors, la consultation des enregistrements de présences sera restreinte aux enregistrements pour ce dit employeur, ou sa chaîne de sous-traitance.

Comme mentionné précédemment, lors de la création des présences, le champ **remarks** sera vide. Il est donc nécessaire d'attendre que ClaO traite les remarques avant d'essayer de les consulter.

Délais de traitement

Afin d'éviter de surcharger inutilement le web service de PresenceRegistration, il est essentiel de prendre en considération les informations suivantes :

- Chaque enregistrement de présence possède une propriété '**validity**'. Cette propriété permet de suivre le traitement de l'enregistrement. Les différentes valeurs suivantes sont possibles :
 - **PENDING** : la présence a été créée, mais elle n'a pas encore été traitée, ce qui signifie que la liste des remarques est vide mais susceptible d'évoluer.
 - **VALIDATED** : la présence a été traitée, et aucune remarque n'a été générée. La liste des remarques est donc vide, et les remarques ne seront plus calculées.
 - **FAILED** : la présence a été traitée et des remarques ont été relevées. Cette liste de remarques est susceptible d'évoluer dans le futur (voir point 'batch'). Attention : l'enregistrement de présence est tout de même enregistré. Vous devez dans ce cas examiner les remarques et examiner si il est possible de les corriger.

Que signifie exactement la validité (validity) 'Failed' ?

La validité 'Failed' signifie que l'enregistrement de présence a correctement été reçu par l'ONSS, et qu'il a bien été enregistré dans notre base de données.

Cependant, l'enregistrement a été contrôlé et des remarques ont été constatées qui doivent attirer votre attention. Les remarques se trouvent dans la propriété '**remarks**'.

- Il peut s'agir d'un problème avec la déclaration Dimona. Dans ce cas, le problème doit être réglé dans le service en ligne 'Dimona'. Lorsque ce problème est résolu dans Dimona, il suffit d'attendre pour que cela se répercute dans ClaO (voir le point suivant, 'Batch').
- Il peut s'agir d'un problème avec la déclaration de travaux. Cela se résout dans le service en ligne 'Déclaration de travaux'. Par exemple, la remarque '12. Pas de DDT pour le BCE'. Si vous êtes entrepreneur-déclarant, cela signifie qu'un travailleur d'une entreprise non déclarée dans Déclaration de travaux (DDT) a travaillé pour votre déclaration. Vous devez dans ce cas, corriger la déclaration de travaux au plus vite, ou avertir l'ONSS.
- Il peut aussi s'agir de problème d'utilisation de Check In and Out at Work par vos travailleurs. Par exemple, la remarque '21. Enregistrement OUT manquant', signifie que nous avons reçu deux pointages 'IN' d'affilé pour ce travailleur.

Les remarques

Voici les différentes remarques possibles. Pour connaître les actions à entreprendre, veuillez vous référer au 3^{ème} point du [document contenant les procédures en cas de problème technique, négligence et erreur](#).

Code de la remarque	Libellé
CAW_1	Il n'existe pas de relation entre le travailleur et l'entreprise
CAW_2	Il existe une relation mais elle n'est pas active
CAW_4	L'entreprise n'existe pas
CAW_6	La déclaration Limosa n'existe pas pour cet identifiant
CAW_7	La déclaration Limosa existe mais n'est pas active
CAW_8	Il n'existe pas de Limosa contenant ce NISS-BIS
CAW_9	Il n'existe pas de Limosa active contenant ce NISS-BIS
CAW_10	La déclaration de travaux n'existe pas pour cet identifiant
CAW_11	La déclaration de travaux existe mais n'est pas active
CAW_12	L'entreprise n'a pas de contrat dans la déclaration de travaux
CAW_14	Un enregistrement similaire existe déjà
CAW_15	Il n'existe personne avec ce NISS

Code de la remarque	Libellé
CAW_17	Le contrat est inactif dans la déclaration de travaux
CIAO_21	Deux ou plusieurs IN d'affilée
CIAO_22	Deux ou plusieurs OUT d'affilée
CIAO_24	OUT sans IN dans les 24 heures précédentes
CIAO_25	Pas de coordonnées GPS remontées
CIAO_32	Délai de réception des enregistrements dépassé
CIAO_33	Il n'existe pas de Limosa contenant ce NISS-BIS
CIAO_34	Enregistrement refusé par CAW
CIAO_35	Service geocoding indisponible
CIAO_36	Impossible de déterminer l'adresse sur base des coordonnées GPS
CIAO_37	Adresse non vérifiable en raison d'informations manquantes
CIAO_38	Impossible de déterminer l'adresse sur base de l'adresse encodée

Batch

Un système de traitement par lots (**batch**) s'exécute quotidiennement et recalcule les remarques des présences ayant la validity **FAILED**. Chaque jour, le batch recalculera toutes les remarques des présences pour les dates suivantes :

Soit D étant la date à laquelle le batch tourne :

- **D - 1** (les remarques de la veille)
- **D - 7** (les remarques créées il y a 1 semaine)
- **M - 1** (les remarques créées il y a 1 mois)
- **M - 3** (les remarques créées il y a 3 mois)

Il est donc inutile d'effectuer des appels chaque minute pour un enregistrement de présence créé il y a, par exemple, 5 jours. En effet, ces remarques ne seront pas traitées à nouveau avant le prochain batch, qui sera exécuté 2 jours plus tard : le batch 'D - 7'.

Bien que 95 % des enregistrement de présences soient traités dans un délai de 10 secondes (en fonction du trafic), nous recommandons les maxima suivants en termes d'appels pour obtenir les remarques des présences nouvellement créées :

Temps après la création de la présence	Fréquence d'appel pour récupérer les remarques
de 0 à 1 minute	Un seul appel toutes les 5 secondes tant que la validité est à PENDING. En effet, si sa validité (validity) est à : • FAILED : les remarques n'évolueront plus avant J + 1, car elles ont été traitées. • VALIDATED : les remarques n'évolueront plus car la présence a été considérée comme valide.
à jour J + 1	Un appel unique. Les remarques ne seront plus traitées avant le prochain batch (à jour J + 7), et uniquement si la validité de la présence est FAILED .
à jour J + 7	Un appel unique. Les remarques ne seront plus traitées avant le prochain batch (à jour J + 30), et uniquement si la validité de la présence est FAILED .
à jour M + 1	Un appel unique. Les remarques ne seront plus traitées avant le prochain batch (à jour M + 1), et uniquement si la validité de la présence est FAILED .
à jour M + 3	Un appel unique. Les remarques ne seront plus traitées, et uniquement si la validité de la présence est FAILED .

Documentation technique

RegisterInBulk

Description

La méthode registerInBulk permet de créer une ou plusieurs présences (jusqu'à 200 par requête) à la condition suivante : les enregistrements doivent avoir lieu en temps réel ; les enregistrements avec un retard de plus de 10 minutes ne sont pas acceptés. Ces présences sont ensuite traitées de manière asynchrone par ClaO.

Requête

HTTP POST /presenceRegistrations/registerInBulk

Body

Le body de la requête doit contenir une liste d'objets 'presence', semblable à celui-ci :

```
{
  "items": [
    {...},
    {...}
  ]
}
```

Chaque objet doit correspondre à la description ci-dessous :

Nom de la propriété	Type de valeur	Obligatoire	Description
registrationDate	string <date-time>	Oui	La date à laquelle le IN ou le OUT a eu lieu, au format ISO 8601 standard : YYYY-MM-DDTHH:MM:SSZ • YYYY-MM-DD pour la date • HH:MM:SS pour l'heure • Z pour représenter la zone ou le fuseau horaire dans lequel la date est donnée
ssin	string <^\d{11}\$>	Oui	Le NISS du travailleur qui a fait le IN ou le OUT
type	string ["IN", "OUT"]	Oui	Le type de présence : • IN : début de travail ou fin de pause • OUT : début de pause ou fin de travail

Nom de la propriété	Type de valeur	Obligatoire	Description
employer	Objet	Oui	L'employeur pour lequel la présence a été enregistrée. Cet objet doit obligatoirement contenir une et une seule de ces 2 propriétés: • enterpriseNumber : string <code><^[0 1]\d{9}\$></code> : le numéro BCE de l'employeur (employeur belge) • foreignVatNumber : string <code><longueur maximale 255></code> : le numéro de TVA de l'employeur (employeur étranger)
placeOfWork	Objet	Oui	Le lieu de travail où la présence a été enregistrée. Cet objet doit obligatoirement contenir une et une seule de ces 2 propriétés : • coordinates : objet : les coordonnées du lieu de prestation sur lequel le travailleur enregistre sa présence (au format WGS84) ○ longitude : number : la longitude de la coordonnées (X) ○ latitude : number : la latitude de la coordonnées (Y) • address : objet : l'adresse du lieu de prestation. N'utilisez pas d'abréviation. ○ postcode : string : code postal ○ municipalityName : string : ville ○ streetName : string : nom de la rue ○ houseNumber : string : numéro ○ boxNumber : string : numéro de boîte
contractualRelationshipReference	string <code><^[A-HJ-NP-Z0-9]{13}\$></code>	Oui	La référence DDT du contrat entre le client/donneur d'ordre et le déclarant

Réponse

Pour chaque présence envoyée dans la requête, un objet associé est présent dans la réponse :

```
{
  "items": [
    {...},
    {...}
  ]
}
```

Chaque objet correspond à la description ci dessous :

Nom de la propriété	Type de valeur	Obligatoire	Description
createdPresenceRegistration	Objet	Non	• Si une erreur s'est produite, cette propriété est nulle • Si l'enregistrement de présence a bien été enregistré, cette propriété contient la présence enregistrée (avec, donc, son <i>id</i> unique)
notCreatedPresenceRegistration	Objet	Non	• Si une erreur s'est produite, cet objet possède 2 propriétés : ○ presenceRegistrationSubmitted : qui décrit l'enregistrement de présence telle que soumis par l'utilisateur (qu'il convient donc de corriger) ○ errorList : une liste d'erreur, expliquant pourquoi l'enregistrement de présence n'a pas été enregistré. Chaque erreur possède 2 propriétés: ▪ errorCode : un code technique, unique à chaque erreur ▪ errorDescription : une description textuelle de l'erreur

Une présence ainsi créée possède l'ensemble des propriétés décrite dans le endpoint '**Read by id**'.

Exemple

Requête

Voici un exemple de requête contenant deux enregistrements de présence. Le premier est valide, le second ne l'est pas.

```
HTTP POST /presenceRegistrations/registerInBulk

{
  "items": [
    {
      "registrationDate": "2019-08-28T14:15:22Z",
      "ssin": "22343312345",
      "type": "in",
      "employer": {
        "enterpriseNumber": "450905686"
      },
      "placeOfWork": {
        "coordinates": {
          "longitude": 25.485606,
          "latitude": 20.673302
        }
      },
      "contractualRelationshipReference": "1Y1003SQ5VSSZ"
    },
    {
      "registrationDate": "2019-08-28T14:15:22Z",
      "ssin": "2299z7777000",
      "type": "in",
      "employer": {
        "enterpriseNumber": "45090568666666"
      },
      "placeOfWork": {
        "coordinates": {
          "longitude": 25.485606,
          "latitude": 20.673302
        }
      },
      "contractualRelationshipReference": "1Y1ZZZZZZZZZZ"
    }
  ]
}
```

Réponse

```
[
  {
    "createdPresenceRegistration": {
      "id": 17611,
      "registrationDate": "2019-08-28T15:15:22+02:00",
      "ssin": "22663312345",
      "type": "in",
      "employer": {
        "enterpriseNumber": "450905686",
        "foreignVatNumber": null
      },
      "placeOfWork": {
```

```

        "coordinates": {
            "longitude": 25.485606,
            "latitude": 20.673302
        },
        "description": null
    },
    "contractualRelationshipReference": "1Y1003SQ5VSSZ",
    "activity": "cleaning",
    "channel": "ws",
    "customReference": null,
    "status": {
        "code": "registered",
        "date": "2024-02-27T07:00:33.460412229+01:00"
    },
    "validity": "pending",
    "remarks": []
},
"notCreatedPresenceRegistration": null
},
{
    "createdPresenceRegistration": null,
    "notCreatedPresenceRegistration": {
        "presenceRegistrationSubmitted": {
            "id": null,
            "registrationDate": "2019-08-28T14:15:22Z",
            "ssin": "22663312345",
            "type": "in",
            "employer": {
                "enterpriseNumber": "4509056866666",
                "foreignVatNumber": null
            },
        },
        "placeOfWork": {
            "coordinates": {
                "longitude": 25.485606,
                "latitude": 20.673302
            },
            "address": {
                "postCode": null,
                "municipalityName": null,
                "streetName": null,
                "houseNumber": null,
                "boxNumber": null
            }
        }
    },
    "contractualRelationshipReference": "1Y1ZZZZZZZZZZ",
    "activity": null,
    "channel": null,
    "customReference": null,
    "status": null,
    "remarks": []
},
"errorList": [
    {
        "errorCode": "error.presence-registration.creation.enterprise-
number",
        "errorDescription": "enterprise number is not valid"
    },
    {

```

```

        "errorCode": "error.presence-registration.creation.contractual-
relationship-reference",
        "errorDescription": "contractual relationship reference is not valid"
    }
  ]
}
]

```

Explications

- Le premier enregistrement de présence est valide (il s'agit d'un exemple pour illustrer les deux cas ; en réalité le NISS utilisé n'est pas correct).
 - Dans la réponse, l'objet correspondant possède donc bien une propriété **createdPresenceRegistration** qui décrit la présence créée.
 - Dans la réponse, la propriété **notCreatedPresenceRegistration** est nulle, car il n'y a pas eu d'erreur.
- Le second enregistrement de présence n'est pas valide.
 - Le numéro de d'entreprise de l'employeur n'est pas valide.
 - Le numéro de déclaration de travaux est aussi invalide
 - Ainsi, l'objet correspondant dans la réponse possède une propriété **createdPresenceRegistration** nulle.
 - L'objet possède également une propriété **notCreatedPresenceRegistration**, qui décrit:
 - **presenceRegistrationSubmitted** : la présence soumise, possédant donc une (ou des) erreur(s).
 - **errorList** : la liste des erreurs en question.

Read by id

Description

Cette méthode permet d'obtenir les informations relatives à une présence particulière, sur base de son ID technique.

Requête

Où **id** est l'ID de la présence dont vous voulez obtenir les informations :

```
HTTP GET /presenceRegistrations/{id}
```

Réponse

Cette méthode peut répondre de différentes façons :

Code de réponse	Interprétation
200	La présence a bien été trouvée.
404	- l'id mentionné dans la requête ne correspond à aucune présence. OU - la présence existe, mais elle n'est pas liée à l'employeur (ou sa chaîne de sous-traitance) lié au token utilisé lors de la requête.

Dans le cas d'un code 200, la réponse contient donc un objet comme tel :

Nom de la propriété	Type de valeur	Description
id	nombre	ID technique unique de l'enregistrement de présence (à utiliser dans le endpoint de consultation)
registrationDate	string <date-time>	La date à laquelle le IN ou le OUT a eu lieu
ssin	string <^\d{11}\$>	Le NISS du travailleur qui a fait le IN ou le OUT
worker	Object	Il s'agit du travailleur. Il contient ces 2 propriétés : - givenName : string : le prénom - familyName : string : le nom de famille
type	string ["IN", "OUT"]	Le type de présence : - IN - OUT

Nom de la propriété	Type de valeur	Description
employer	Objet	L'employeur pour lequel l'enregistrement de présence a été enregistré. Cet objet doit obligatoirement contenir une et une seule de ces 2 propriétés : enterpriseNumber : string <^[0 1]\d{9}\$> : le numéro BCE de l'employeur (employeur belge) foreignVatNumber : string <longueur maximale 255> : le numéro de TVA de l'employeur (employeur étranger)
placeOfWork	Objet	L'endroit où la présence a été enregistrée. Cet objet doit obligatoirement contenir une et une seule de ces 2 propriétés : coordinates : objet : les coordonnées du lieu de prestation sur lequel le travailleur enregistre sa présence (au format WGS84) longitude : number : la longitude de la coordonnées (X) latitude : number : la latitude de la coordonnées (Y) address : objet : l'adresse du lieu de prestation. N'utilisez pas d'abréviation. postcode : string : code postal municipalityName : string : ville streetName : string : nom de la rue houseNumber : string : numéro boxNumber : string : numéro de boîte
contractualRelationshipReference	string <^[A-HJ-NP-Z0-9]{13}\$>	La référence DDT du contrat entre le client/donneur d'ordre et le déclarant
activity	string	Décrit le type d'activité réalisée par le travailleur
channel	string	Le canal via lequel la présence a été enregistrée : MOBILE_URL MOBILE_MANUAL WS WEB_APP

Nom de la propriété	Type de valeur	Description
customReference	string	Référence personnalisable pour éventuellement faciliter l'intégration (cela permet de garder un lien entre l'ID technique de l'application source par exemple)
status	Objet	Le statut de la présence. Cet objet possède 2 propriétés: • Code : le code du statut actuel de la présence: ○ REGISTERED ○ EDITED ○ CANCELLED • Date : la date à laquelle le statut a été changé la dernière fois
validity	Object	La validité de la présence: • pending • failed • validated
remarks	Liste d'objet 'remarque'	Détient une liste de toutes les remarques calculées pour cette présence. Un objet remarque possède les propriétés suivantes: • Code : le code unique de la remarque • Labels : ○ nl : le nom de la remarque en néerlandais ○ fr : le nom de la remarque en français ○ de : le nom de la remarque en allemand ○ en : le nom de la remarque en anglais

Exemple

Requête

```
HTTP GET /presenceRegistrations/17053
```

Réponse

```
{
  "id": 17053,
  "registrationDate": "2024-01-30T13:58:53.774+01:00",
  "ssin": "22663312345",
  "worker": {
    "givenName": "John",
    "givenName": "Doe"
  },
  "type": "in",
  "employer": {
    "enterpriseNumber": "70010100188",
    "foreignVatNumber": null
  },
  "placeOfWork": {
    "coordinates": {
      "longitude": 4.348314,
      "latitude": 50.839552
    },
    "address": {
      "postCode": null,
      "municipalityName": null,
      "streetName": null,
      "houseNumber": null,
      "boxNumber": null
    }
  },
  "contractualRelationshipReference": "1Y1ZZZZZZZZZZ",
  "activity": "cleaning",
  "channel": "mobile_manual",
  "customReference": null,
  "status": {
    "code": "failed",
    "date": "2024-01-30T13:58:59.930499+01:00"
  },
  "validity": "failed",
  "remarks": [
    {
      "code": "ciao_26",
      "labels": {
        "nl": "Gps-coördinaten niet gevonden in AVW",
        "fr": "Coordonnées GPS introuvables pour la DDT",
        "de": "GPS-Koordinaten für Arbeitsmeldung nicht gefunden",
        "en": "GPS coordinates not found for DOW"
      }
    }
  ]
}
```

Search

Description

Cette méthode permet de rechercher des présences correspondant à une liste de critères. En réponse, elle fournit une liste paginée des enregistrements de présences qui satisfont ces critères.

Une entreprise peut faire des recherches pour ses propres travailleurs ou ses sous-traitants. Lorsque vous recherchez les enregistrements d'un sous-traitant, vous devez remplir la zone de déclaration de travaux (contractualRelationshipReference).

Vous êtes éditeur de software ? Dans ce cas, vous devrez utiliser le certificat de votre client pour récupérer les enregistrements de présence. En effet, votre certificat permet la création d'enregistrements, mais pas la lecture de ceux-ci.

Requête

```
HTTP POST /presenceRegistrations/search
```

Body

Le **body** de la requête doit contenir un objet contenant une propriété **criteria** :

```
{
  "criteria": {
    ...
  }
}
```


Le corps de la requête doit répondre à la description ci-dessous :

Nom de la propriété	Type de valeur	Obligatoire	Valeur par défaut	Description
criteria	objet	Oui		Il s'agit des critères de recherche. Ces critères possèdent exactement les mêmes propriétés que l'objet présence décrit au point Read by id auxquelles s'ajoutent la propriété obligatoire suivante: • registrationDate : ○ startDate : string <date-time>: la date de début sur base de laquelle filtrer les enregistrements de présences ○ endDate : string <date-time>: la date de fin sur base de laquelle filtrer les enregistrements de présences Les enregistrements de présences qui résulteront de recherche posséderont donc une date d'enregistrement comprise entre registrationDate.startDate et registrationDate.endDate
sort	objet	Non	<pre>{ "direction": "desc", "ignoreCase": false, "property": "registrationDate" }</pre>	Un objet permettant de décrire la façon dont les résultats sont triés: • direction : ["ASC", "DESC"] pour ascendant ou descendant • ignoreCase : pour préciser si la recherche doit tenir compte de la casse ou pas • property : le nom de la propriété sur laquelle faire le tri

Autres paramètres

La requête peut contenir les paramètres de requête suivants :

Nom de la propriété	Type de valeur	Obligatoire	Valeur par défaut	Description
page	nombre	Non	1	La page à laquelle vous voulez accéder
pageSize	nombre	Non	50	Le nombre d'éléments par page

Réponse

Code de réponse	Interprétation
200	La recherche s'est bien déroulée, les résultats sont présents dans la réponse
500	Une erreur est survenue, vérifiez que les critères de recherche sont bien correctement formatés

Dans le cas d'un code 200, la recherche renvoie un objet correspondant à la description suivante :

Nom de la propriété	Type de valeur	Description
items	Liste d'objets	La liste des présences répondant aux critères de recherche. Les présences sont au format décrit au point Read by ID
first	string	Le lien vers la première page des résultats de la recherche
last	string	Le lien vers la dernière page des résultats de la recherche
prev	string	Le lien vers la page précédente des résultats de la recherche
next	string	Le lien vers la page suivante des résultats de la recherche
page	nombre	Le numéro de la page actuelle des résultats de recherche
pageSize	nombre	La taille de la page actuelle des résultats de recherche

Nom de la propriété	Type de valeur	Description
sort	objet	Un objet permettant de décrire la façon dont les résultats sont triés: direction : ["ASC", "DESC"] pour ascendant ou descendant ignoreCase : pour préciser si la recherche doit tenir compte de la casse ou pas property : le nom de la propriété sur laquelle faire le tri
total	nombre	Le nombre total d'éléments résultant de cette recherche
totalPages	nombre	Le nombre total de pages résultant de cette recherche

Exemple

Requête

HTTP POST /presenceRegistrations/search

```
{
  "criteria": {
    "registrationDate": {
      "startDate": "2024-01-30T10:12:52+01:00",
      "endDate": "2024-02-15T10:12:54+01:00"
    },
    "type": "in"
  }
}
```

Réponse

```
{
  "items": [
    {
      "id": 17053,
      ...
    },
    {
      "id": 17054,
      ...
    }
  ],
  "first":
  "/REST/presenceRegistration/v1/presenceRegistrations/search?page=1&pageSize=50"
,
```

```
    "last":
"/REST/presenceRegistration/v1/presenceRegistrations/search?page=2&pageSize=50"
  ,
    "prev": null,
    "next":
"/REST/presenceRegistration/v1/presenceRegistrations/search?page=2&pageSize=50"
  ,
    "page": 1,
    "pageSize": 50,
    "sort": {
      "direction": "desc",
      "ignoreCase": false,
      "property": "registrationDate"
    },
    "total": 52,
    "totalPages": 2
  }
```

Explications

Cette requête liste l'ensemble des enregistrements de présences, qui :

- possèdent une date d'enregistrement comprise entre le 30/01/2024 à 10:12 et le 15/02/2024 à 12:54 (champ '**registrationDate**' dans la requête)
- sont de type 'in' (champ '**type**' dans la requête)

Un critère est ajouté de manière transparente à la requête. Ces présences ont toutes été enregistrées pour l'employeur lié au token utilisé dans la requête.