

Réservation d'un créneau

La réservation d'un créneau s'effectue par un appel à l'adresse
`/api/agenda/SLUG-DE-LAGENDA/fillslot/IDENTIFIANT-DU-CRENEAU/` via la méthode HTTP POST.

Cette procédure est commune aux agendas de type rendez-vous et événement, seul le format de l'identifiant du créneau diffère.

Cette adresse est mentionnée dans le champ `fillslot_url` obtenu en demandant la [liste des créneaux disponibles](#), ce qui évite la plupart du temps d'avoir à la construire manuellement.

Paramètres JSON, corps de la requête

La plupart des paramètres sont communs aux deux types d'agenda.

Nom	Description	Exemple
<code>backoffice_url</code>	URL de la démarche effectuant la réservation	<code>{{form_url_backoffice}}</code>
<code>form_url</code>	URL de la demande	<code>{{form_url}}</code>
<code>cancel_callback_url</code>	URL à appeler lors d'une annulation	
<code>cancel_booking_id</code>	Identifiant de réservation à annuler	45
<code>count</code>	Nombre de places à réserver	2
<code>label</code>	Texte arbitraire pour l'affichage en backoffice	<code>{{form_name safe}}</code>
<code>user_display_label</code>	Texte pour affichage dans export ics	<code>{{form_name safe}}</code>
<code>user_name</code>	Nom complet l'utilisateur (peut être décomposé et remplacé par <code>user_first_name</code> et <code>user_last_name</code>)	<code>{{form_user_display_name}}</code>
<code>user_first_name</code>	Prénom de l'utilisateur	<code>{{form_user_var_first_name}}</code>
<code>user_last_name</code>	Nom de l'utilisateur	<code>{{form_user_var_last_name}}</code>
<code>user_email</code>	Adresse de courriel de l'utilisateur	<code>{{form_user_email}}</code>
<code>user_phone_number</code>	Numéro de téléphone de l'utilisateur	<code>{{form_user_var_phone}}</code>
<code>user_external_id</code>	Identifiant unique de l'utilisateur	<code>{{form_user_nameid}}</code>
<code>exclude_user</code>	Interdire à l'utilisateur d'avoir plusieurs réservations en même temps	true (false par défaut)
<code>bypass_delays</code>	Ne pas prendre en compte les délais de réservation	true (false par défaut)
<code>extra_emails</code>	Adresses de courriel supplémentaires où envoyer un rappel	<code>{{form_var_emails_raw}}</code> ou " a@example.com,b@example.com "
<code>extra_phone_numbers</code>	Numéros de téléphone supplémentaires où envoyer un rappel	<code>{{form_var_numbers_raw}}</code> ou "+33122334455,+33122334466"
<code>absence_callback_url</code>	URL à appeler lorsque l'événement est marqué comme pointé, en cas d'absence	
<code>presence_callback_url</code>	URL à appeler lorsque l'événement est marqué comme pointé, en cas de présence	

Il est possible de passer n'importe quel paramètre en plus de ceux ci-dessus : ils seront enregistrés et apparaîtront dans la clé `extra_data` de l'API qui retourne les [informations d'une réservation](#).

Les appels à `absence_callback_url` et `presence_callback_url` se feront en POSTant un objet avec les clés `user_was_present` (booléen), `user_check_type_slug` (identifiant du motif d'absence), `user_check_type_label` (libellé du motif d'absence).

Le cancel_booking_id permet de combiner l'annulation d'un créneau et la réservation d'un nouveau (par exemple sur un agenda différent).

Agenda évènement

Certains paramètres supplémentaires peuvent être inclus :

Nom	Description	Exemple
events	Restreindre la réservation aux évènements passés ou futurs ("future" par défaut)	"future", "past", "all"
force_waiting_list	Forcer le passage en liste d'attente	true

Exemple

POST /api/agenda/foo-bar/fillslot/event-slug/

```
{
  "agenda": {
    "label": "Foo bar",
    "slug": "foo-bar"
  },
  "api": {
    "accept_url": "http://chrono.dev.publik.love/api/booking/8/accept/",
    "anonymize_url": "http://chrono.dev.publik.love/api/booking/8/anonymize/",
    "booking_url": "http://chrono.dev.publik.love/api/booking/8/",
    "cancel_url": "http://chrono.dev.publik.love/api/booking/8/cancel/",
    "ics_url": "http://chrono.dev.publik.love/api/booking/8/ics/",
    "suspend_url": "http://chrono.dev.publik.love/api/booking/8/suspend/"
  },
  "booking_id": 8,
  "datetime": "2017-05-21 17:00:00",
  "end_datetime": null,
  "err": 0,
  "in_waiting_list": false,
  "places": {
    "available": 19,
    "full": false,
    "has_waiting_list": false,
    "reserved": 1,
    "total": 20
  }
}
```

Agenda rendez-vous

Certains paramètres supplémentaires peuvent être inclus :

Nom	Description	Exemple
use_color_for	Libellé auquel associer la couleur d'un rendez-vous dans le backoffice	

Exemple

POST /api/agenda/rdv-pref/fillslot/18:2020-06-16-1000/

```
{
  "agenda": {
    "label": "rdv pref",
    "slug": "rdv-pref"
  },
}
```

```

"api": {
  "anonymize_url": "https://chrono.dev.publik.love/api/booking/50/anonymize/",
  "cancel_url": "https://chrono.dev.publik.love/api/booking/50/cancel/",
  "ics_url": "https://chrono.dev.publik.love/api/booking/50/ics/",
  "suspend_url": "https://chrono.dev.publik.love/api/booking/50/suspend/"
},
"booking_id": 50,
"datetime": "2020-06-16 10:00:00",
"desk": {
  "label": "ds",
  "slug": "ds"
},
"duration": 30,
"end_datetime": "2020-06-16 10:30:00",
"err": 0,
"in_waiting_list": false,
"resources": []
}

```

Agenda évènement : poser plusieurs réservations

Il est possible de réserver plusieurs évènements en une seule fois via un appel POST à l'adresse `/api/agenda/SLUG-DE-LAGENDA/events/fillslots/`.

Paramètres JSON, corps de la requête

Par rapport à l'appel classique de réservation d'un évènement, deux paramètres doivent obligatoirement être présents.

Nom	Description	Exemple
<code>user_external_id</code>	Identifiant unique de l'utilisateur. Grâce à cet identifiant, chaque appel aura pour effet de mettre à jour les réservations.	<code>{{form_user_nameid}}</code>
<code>slots</code>	Slugs des évènements à réserver. Une valeur vide aura pour effet d'annuler les réservations de l'utilisateur.	<code>{{form_var_event_raw join:";"}}</code>

Exemple

POST `/api/agenda/foo-bar/events/fillslots/`

```

{
  "err": 0,
  "booking_count": 3,
  "waiting_list_events": [
    {
      "id": "event-slug",
      "slug": "event-slug",
      ...
    }
  ],
  "cancelled_booking_count": 0
}

```

Agenda évènement : poser plusieurs réservations sur plusieurs agendas

Il est possible de réserver plusieurs évènements qui se trouvent sur des agendas différents via un appel HTTP POST à l'adresse `/api/agendas/events/fillslots/`.

Paramètres de l'URL

Nom	Description	Exemple
<code>agendas</code>	Slugs des agendas concernés par l'appel.	<code>agenda-slug,agenda2-slug,agenda3-slug</code>

	Ils permettent de savoir quelles réservations mettre à jour.	
--	--	--

Paramètres JSON, corps de la requête

Par rapport à l'appel classique de réservation d'un évènement, deux paramètres doivent obligatoirement être présents.

Nom	Description	Exemple
user_external_id	Identifiant unique de l'utilisateur. Grâce à cet identifiant, chaque appel aura pour effet de mettre à jour les réservations.	{{form_user_nameid}}
slots	Slugs des évènements à réserver. Une valeur vide aura pour effet d'annuler les réservations de l'utilisateur.	agenda-slug@event-slug,agenda2-slug@event-slug

Exemple

POST /api/agendas/events/fillslots/?agendas=agenda-slug

```
{
  "err": 0,
  "booking_count": 3,
  "waiting_list_events": [
    {
      "id": "event-slug",
      "slug": "event-slug",
      ...
    }
  ],
  "cancelled_booking_count": 0
}
```

Agenda évènement : réserver toutes les occurrences d'évènements récurrents

Il est possible de réserver un ou plusieurs évènements récurrents, en totalité ou sur une plage temporelle définie, via un appel POST à l'adresse /api/agendas/recurring-events/fillslots/.

Paramètres de l'URL

Nom	Description	Exemple
agendas	Slugs des agendas concernés par l'appel.	agenda-slug,agenda2-slug,agenda3-slug
action	Action à réaliser sur les évènements.	update, book ou unbook
date_start	Réserver seulement les évènements commençant après cette date.	2021-09-04
date_end	Réserver seulement les évènements commençant avant cette date.	2022-06-21

Paramètres JSON, corps de la requête

Par rapport à l'appel classique de réservation d'un évènement, deux paramètres doivent obligatoirement être présents.

Nom	Description	Exemple
user_external_id	Identifiant unique de l'utilisateur. Grâce à cet identifiant, chaque appel aura pour effet de mettre à jour les réservations.	{{form_user_nameid}}
slots	Slugs et jours des évènements à réserver.	agenda-slug@event-slug:0,agenda-slug@event2-slug:1 pour réserver le premier évènement tous les lundis et le second

Exemple

POST /api/agendas/recurring-events/fillslots/?agendas=agenda-slug

```
{
  "err": 0,
  "booking_count": 142,
  "full_events": [
    {
      "id": "event-slug",
      "slug": "event-slug",
      ...
    }
  ],
  "cancelled_booking_count": 0
}
```