# AI SPV (Atama Masası) API

Mopuan uygulaması ve sunucusu için entegrasyon kılavuzu

opsenseapi · Sürüm 1.0 · 9 Kasım 2026

## 1. Bu servisler ne için kullanılır

**AI SPV** (Atama Masası), bir ekibin işlerini planlar. Henüz kimsenin almadığı açık iş emirlerini alır ve her birini hangi çalışanın yapması gerektiğini önerir. Bir **hedef** seçersiniz (örneğin kalite veya düşük maliyet) ve AI SPV o hedef için en iyi planı hesaplar.

Bir öneri yalnızca **taslaktır**. Bir atamayı **onaylamadan** (commit) OpSense'te hiçbir şey değişmez. Kullanıcı planı inceleyebilir, bir kişiyi bir işe kilitleyebilir, bir işi atlayabilir veya ekibi değiştirebilir, sonra yeniden hesaplatabilir.

Sekiz servis vardır:

| Servis | Ne için kullanılır |
|---|---|
| `SRVGetAiSpvDesk` | Masayı yüklemek: bir tarih aralığı için ekip, açık iş emirleri, mevcut işler, hizmetler ve yerler. |
| `SRVCalculateAiSpv` | Bir hedef için önerilen planı hesaplamak. Hiçbir şeyi değiştirmez. |
| `SRVAiSpvCommitAssignment` | Bir iş emrini bir çalışana atamak. |
| `SRVAiSpvCreateWorkOrder` | Atanmamış yeni bir iş emri oluşturmak. |
| `SRVAiSpvTogglePool` | Bir çalışanı planlama ekibinden çıkarmak veya geri almak. |
| `SRVAiSpvSetQualityOverride` | Bir çalışan için elle kalite puanı girmek. |
| `SRVAiSpvClearQualityOverride` | Bir elle kalite puanını veya hepsini kaldırmak. |
| `SRVAiSpvSaveStaffCost` | Bir çalışanın saatlik maliyetini kaydetmek. |

## 2. Hedefler

Hedefi `goal` alanında gönderin. Göndermezseniz `balanced` kullanılır. AI SPV her çalışanı her iş için on ölçüte göre (0 ile 100 arası) puanlar ve ağırlıklarla toplar. Her hedef farklı ağırlıklar kullanır.

| `goal` | Neyi öne çıkarır |
|---|---|
| `balanced` | Her şeyin karışımı. Varsayılandır. Kalite ve beceri en çok sayılır. |
| `quality` | İş için en becerikli ve en iyi puanlı çalışan; işin önceliği ile çalışanın seviyesi arasında iyi uyum. Maliyeti dikkate almaz. |

| | |
|---|---|
| `efficiency` | Kısa yol, hızlı çalışanlar, yakında boşalan kişiler ve aynı binada kalmak. |
| `fairness` | Dengeli iş yükü. En çok boş zamanı olan çalışan sıradaki işi alır. |
| `sla` | İşi zamanında bitirmek: şu an boş, hızlı, yakın ve güvenilir kişiler. |
| `cost` | En düşük saatlik ücret. Yalnızca ekipte en az iki çalışanın maliyeti kayıtlıysa çalışır; aksi halde maliyet dikkate alınmaz. |

Bir işi bitiş saatinden sonra bitirecek seçim, hedef ne olursa olsun 25 puan kaybeder.

## 3. Kimlik doğrulama

Her çağrı iki alan ister:

| Alan | Zorunlu mu | Notlar |
|---|---|---|
| `api_key` | **Zorunlu** | Şirketin anahtarı. Çağrının hangi şirket için olduğunu belirler. |
| `actor_mopuan_user_id` | **Zorunlu** | Masayı kullanan kişinin Mopuan çalışan id'si. |

**Bu servisleri yalnızca kendi sunucunuzdan çağırın.** Sunucunuz `actor_mopuan_user_id` alanını, oturumu açık olan çalışanla doldurmalıdır. Bu değeri asla telefondan veya uygulamanın isteğinden okumayın. Uygulama bu değeri seçebilirse, her kullanıcı başka bir çalışan gibi işlem yapabilir. `api_key` hiçbir zaman mobil uygulamada bulunmamalıdır.

İşlem yapan kişi iki şeyi belirler: çağrının neye izin verildiği (bölüm 4) ve değişikliği kimin yaptığı olarak kimin kaydedileceği (örneğin bir iş emrini kimin atadığı).

## 4. Yetkiler

AI SPV, işlem yapan kişinin rolündeki **Work Orders** (İş Emirleri) modülü yetkisini kullanır.

| Servisler | Gereken yetki |
|---|---|
| `SRVGetAiSpvDesk`, `SRVCalculateAiSpv` | Work Orders: görüntüleme |
| Diğer tüm servisler | Work Orders: düzenleme |

`SRVAiSpvSaveStaffCost` ayrıca **Payments** modülünde düzenleme yetkisi ister ve Payments modülü şirket için kapatılmamış olmalıdır.

Bir rolün neye yetkili olduğunu `SRVGetUserRoles` ile okuyabilirsiniz (Mopuan Roller ve Yetkiler kılavuzuna bakın). Masa verisinde ayrıca `canEdit` vardır; işlem yapan kişi düzenleyebiliyorsa `true` olur. `false` ise işlem düğmelerini gizleyin.

## 5. İstekler ve yanıtlar

```
POST https://<api-host>/v1/<service>
Content-Type: application/x-www-form-urlencoded
```

Tüm parametreleri **form alanı** olarak gönderin. JSON istek gövdesi okunmaz.

Bazı alanlar liste veya nesne taşır. Bunları **tek bir form alanı içinde JSON metni** olarak gönderin: `SRVCalculateAiSpv` üzerinde `locks`, `skips` ve `pool`. Bu alanların her biri için sınırlar:

- En fazla **64 KB** metin.
- En fazla **5000** kayıt.

Sınırı aşan veya geçerli JSON olmayan bir alan `error_code` 10051 ile reddedilir.

**Bu servisler, diğer OpSense servislerinin kullandığı `{"result": ...}` sarmalayıcısı olmadan yanıt verir.** Başarılı yanıt:

```
{"result": "success", "data": { ... }}
```

Hata yanıtı:

```
{"error_code": "10059", "error_text": "Work order is no longer open for this assignment."}
```

`error_code` bir metindir (string). Önce her zaman `error_code` var mı diye bakın. Kodunuzda `error_text` değil `error_code` kullanın.

## 6. Tipik akış

1. **Masayı yükleyin.** Tarih aralığı için `SRVGetAiSpvDesk` çağırın. Ekibi, açık iş emirlerini ve zaten atanmış işleri gösterin.
2. **Hesaplayın.** Aynı aralık ve kullanıcının seçtiği hedefle `SRVCalculateAiSpv` çağırın. Planı gösterin: hangi iş kime düşüyor ve atanamayan iş emirleri.
3. **İnceleyin.** Kullanıcı bir kişiyi bir işe kilitleyebilir (`locks`), bir işi atlayabilir (`skips`) veya bu hesap için bazı kişileri ekipten çıkarabilir (`pool`). Bu seçimlerle `SRVCalculateAiSpv` işlemini yeniden çağırın.
4. **Onaylayın.** Kullanıcının kabul ettiği her atama için `SRVAiSpvCommitAssignment` çağırın; her çağrıda bir iş emri.
5. **Yeniden hesaplayın.** Her onaydan sonra `SRVCalculateAiSpv` işlemini tekrar çağırın. Plan, kimde ne kadar iş olduğuna bağlıdır.

## 7. SRVGetAiSpvDesk

Masayı çizmek için gereken her şeyi döndürür. Yalnızca okuma yapar.

| Alan | Zorunlu mu | Notlar |
|---|---|---|
| `api_key` | **Zorunlu** | Bkz. Kimlik doğrulama. |

| | | |
|---|---|---|
| `actor_mopuan_user_id` | **Zorunlu** | Bkz. Kimlik doğrulama. |
| `date` | İsteğe bağlı | İlk gün, `Y-m-d`. Varsayılan: bugün. |
| `date2` | İsteğe bağlı | Son gün, `Y-m-d`. Varsayılan: bugün artı 7 gün. |

**Tarih kuralları.** `SRVGetAiSpvDesk` ve `SRVCalculateAiSpv` için geçerlidir.

- Tarih, `Y-m-d` biçiminde gerçek bir takvim tarihi olmalıdır. Başka her şey (`2026-02-30`, `09/11/2026`, boş değer) varsayılanıyla değiştirilir. Diğer tarih yine kullanılır.
- `date`, `date2` değerinden sonraysa ikisi yer değiştirir.
- Aralık en fazla **31 gündür**. Daha uzun aralık, `date` tarihinden başlayarak 31 güne kısaltılır.
- Yanıttaki `days`, aralıktaki her günü listeler.

```
curl -X POST "https://<api-host>/v1/SRVGetAiSpvDesk" \
 -d "api_key=<api_key>" \
 -d "actor_mopuan_user_id=345" \
 -d "date=2026-11-10" \
 -d "date2=2026-11-16"
```

Kısaltılmış yanıt:

```
{
 "result": "success",
 "data": {
   "lang": "en",
   "canEdit": true,
   "from": "2026-11-10",
   "to": "2026-11-16",
   "days": ["2026-11-10", "2026-11-11", "..."],
   "today": "2026-11-09",
   "historyDays": 90,
   "services": [{"id": "7", "name": "Flat Cleaning", "dur": 75, "samples": 41}],
   "locs": [{"id": "12", "name": "Harbour House", "lat": 51.5072, "lng": -0.1276}],
   "staff": [{
     "id": "U20", "dbId": 20, "name": "Jane Doe", "role": "Housekeeper",
     "photo": "https://.../uploads/users/jane.jpg", "color": "#6B63E6",
     "inPool": true, "kpi": 82, "quality": 90, "qualityOverride": true,
     "cost": 14.5, "rel": 96, "jobs": 120, "done": 115, "home": "12",
     "homeGeo": {"lat": 51.507, "lng": -0.128},
     "svc": {"7": {"jobs": 30, "done": 29, "skill": 91, "speed": 110, "avg_minutes": 68}},
     "rota": {"2026-11-10": {"off": false, "label": "06:00 - 15:00", "sh": [360, 900]}}
   }],
   "tasks": [{
     "id": "WO4521", "dbId": 4521, "title": "Flat 3B", "branchName": "Harbour House",
     "service": "7", "pri": 2, "loc": "12", "date": "2026-11-10",
     "rel": 600, "due": 780, "roomId": 88
   }],
   "existing": [{
     "id": "WO4400", "dbId": 4400, "title": "Flat 1A", "branchName": "Harbour House",
     "service": "7", "pri": 1, "loc": "12", "date": "2026-11-10",
     "rel": 540, "due": 780, "roomId": 80, "assignedTo": 20
   }]
 }
}
```

## Masa alanları

| Alan | Açıklama |
|---|---|
| `lang` | İşlem yapan kişinin dili: `en`, `tr`, `el` veya `vi`. AI SPV'nin ürettiği metinler (bölüm 8'deki `why` metinleri) bu dili kullanır. |
| `canEdit` | İşlem yapan kişinin Work Orders düzenleme yetkisi varsa `true`. |
| `from`, `to` | Tarih kurallarından sonra kullanılan aralık. |
| `days` | Aralıktaki her gün, `Y-m-d`. |
| `today` | Bugünün tarihi. |
| `historyDays` | Puanların kaç günlük geçmişe dayandığı (90). |
| `services` | Aktif hizmetler: `id`, `name`, `dur` (olağan süre, dakika; henüz kimse süre ölçmediyse 60), `samples` (`dur` değerinin kaç ölçülmüş işe dayandığı). |
| `locs` | Yerler (binalar): `id`, `name`, `lat`, `lng`. Bilinmiyorsa `lat` ve `lng` `null` olur. |
| `staff` | Ekip. Aşağıya bakın. |
| `tasks` | Aralıktaki açık, atanmamış iş emirleri; en fazla 2000. AI SPV bunları planlar. Aşağıya bakın. |
| `existing` | Aralıkta zaten atanmış iş emirleri; en fazla 5000. Yük olarak sayılır ama değiştirilmez. `tasks` ile aynı alanlar ve ek olarak `assignedTo`. |

### Çalışan (staff) alanları

Ekip, Supervisor seviyesinin altındaki aktif çalışanlardır.

| Alan | Açıklama |
|---|---|
| `id` | Çalışan anahtarı, örneğin `U20`. `locks` ve `pool` içinde kullanın. |
| `dbId` | OpSense kullanıcı id'si, örneğin `20`. `user_id` alanlarında ve `assignedTo` için kullanın. |
| `name`, `role` | Ad soyad ve rol adı. |
| `photo` | Fotoğraf URL'si veya boş metin. |
| `color` | Bu çalışan için hex renk kodu. |
| `inPool` | Çalışan planlama ekibindeyse `true`. Yalnızca ekipteki kişilere iş verilir. |
| `kpi` | Çalışanın bu ayki KPI puanı (0 ile 99 arası); veri yoksa `null`. |
| `quality` | Planın kullandığı kalite puanı (0 ile 99 arası). Elle girilen değer `kpi` değerinden önce gelir. İkisi de yoksa `null`. |

| | |
|---|---|
| `qualityOverride` | `quality` elle girilmiş bir değerse `true`. |
| `cost` | Saatlik maliyet veya `null`. |
| `rel` | Güvenilirlik: atanan işlerin bitirilen oranı (0 ile 100 arası); 3'ten az iş varsa `null`. |
| `jobs`, `done` | Geçmiş döneminde atanan ve bitirilen iş sayısı. |
| `home` | Çalışanın genelde çalıştığı bina (bir `locs` id'si) veya `null`. |
| `homeGeo` | Çalışanın gününün başladığı yer, yaklaşık 100 m'ye yuvarlanmış `{lat, lng}`; yoksa `null`. |
| `svc` | Hizmet id'sine göre: `jobs`, `done`, `skill` (0 ile 100 arası), `speed` (60 ile 140 arası, 100 şirket ortalaması, yoksa `null`), `avg_minutes`. |
| `rota` | Güne göre: `off` (izin günüyse `true`), `label` (vardiya metni), `sh` (vardiya, gece yarısından sonraki dakika olarak `[başlangıç, bitiş]`, yoksa `null`). Listede olmayan gün normal çalışma günüdür. |

### İş (task) alanları

| Alan | Açıklama |
|---|---|
| `id` | İş anahtarı, örneğin `WO4521`. `locks` ve `skips` içinde kullanın. |
| `dbId` | İş emri id'si. Onaylarken `work_order_id` olarak kullanın. |
| `title` | Oda adı, yoksa bina adı, yoksa `#id`. |
| `branchName` | Bina adı. |
| `service` | Hizmet id'si (bir `services` id'si). |
| `pri` | Öncelik, 1 (normal) ile 4 (acil) arası. |
| `loc` | Bina id'si (bir `locs` id'si). |
| `date` | Randevu günü, `Y-m-d`. |
| `rel` | En erken başlangıç, gece yarısından sonraki dakika olarak (randevu saati). |
| `due` | En geç bitiş, gece yarısından sonraki dakika olarak: öncelik 1 ile 4 için `rel` artı 240, 180, 120 veya 60. |
| `roomId` | Oda id'si veya 0. |

## 8. SRVCalculateAiSpv

Önerilen planı hesaplar. Hiçbir şeyi değiştirmez.

| Alan | Zorunlu mu | Notlar |
|---|---|---|
| `api_key` | **Zorunlu** | Bkz. Kimlik doğrulama. |
| `actor_mopuan_user_id` | **Zorunlu** | Bkz. Kimlik doğrulama. |
| `date`, `date2` | İsteğe bağlı | Aralık. Kurallar bölüm 7'deki gibidir. |
| `goal` | İsteğe bağlı | `balanced`, `quality`, `efficiency`, `fairness`, `sla`, `cost` değerlerinden biri. Varsayılan `balanced`. |
| `locks` | İsteğe bağlı | JSON nesne: iş anahtarı ve çalışan anahtarı, örneğin `{"WO4521":"U20"}`. O kişiyi o işe zorlar. |
| `skips` | İsteğe bağlı | İş anahtarlarının JSON dizisi, örneğin `["WO4530"]`. Bu işler atanmadan bırakılır. |
| `pool` | İsteğe bağlı | Çalışan anahtarlarının JSON dizisi, örneğin `["U20","U31"]`. Kayıtlı ekip yerine yalnızca bu kişilerle planlar. Hiçbir şey kaydedilmez. Göndermezseniz kayıtlı ekip kullanılır. |

İş anahtarları `WO` ve bir sayıdan oluşmalıdır. Çalışan anahtarları `U` ve bir sayıdan oluşmalıdır. Başka her şey `error_code` 10051 ile reddedilir. Bilinmeyen bir anahtar hata değildir: yok sayılır.

Çalışan o iş için normalde engelli olsa bile (örneğin izin günü) kilit uygulanır. Atama o zaman `manual` olarak işaretlenir. `locks` ve `skips` yalnızca `smart` için geçerlidir.

```
curl -X POST "https://<api-host>/v1/SRVCalculateAiSpv" \
 -d "api_key=<api_key>" \
 -d "actor_mopuan_user_id=345" \
 -d "date=2026-11-10" \
 -d "date2=2026-11-16" \
 -d "goal=quality" \
 --data-urlencode 'locks={"WO4521":"U20"}' \
 --data-urlencode 'skips=["WO4530"]' \
 --data-urlencode 'pool=["U20","U31","U44"]'
```

## Yanıt

`data` hedefi ve iki sonucu içerir:

| Alan | Açıklama |
|---|---|
| `goal` | Kullanılan hedef. |
| `smart` | Hedef için AI SPV planı. `mode` değeri `smart`. |
| `baseline` | Karşılaştırma için basit plan: en erken boşalan çalışan işi alır. `mode` değeri `simple`. `locks` ve `skips` dikkate alınmaz. AI SPV'nin neyi iyileştirdiğini göstermek için kullanın. |

Her sonuçta aynı alanlar vardır:

| Alan | Açıklama |
|---|---|
| `mode` | `smart` veya `simple`. |
| `assignments` | Çalışan bulunan işler. |
| `unassigned` | Atanamayan veya sizin atladığınız işler. |
| `days` | Ekibin günü, tarihe göre (aşağıya bakın). |
| `kpi` | Tüm plan için özet rakamlar. |


```
{
 "result": "success",
 "data": {
   "goal": "quality",
   "smart": {
     "mode": "smart",
     "assignments": [{
       "t": {"id": "WO4521", "dbId": 4521, "title": "Flat 3B", "service": "7", "pri": 2, "loc": "12", "date": "2
       "day": "2026-11-10",
       "staffId": "U20",
       "manual": true,
       "score": 88.4, "base": 88.4, "pen": 0,
       "c": {"skill": 91, "quality": 90, "speed": 62.5, "proximity": 100, "avail": 100,
             "balance": 87.5, "reliability": 96, "continuity": 100, "fit": 86, "cost": 70},
       "reasons": {"skill": 91, "quality": 90, "...": "..."},
       "tr": 0, "trInfo": {"m": 0, "km": 0, "mode": "same"},
       "start": 600, "finish": 675, "dur": 75, "late": false,
       "start_time": "10:00", "end_time": "11:15",
       "cands": [{"id": "U20", "score": 88.4}, {"id": "U31", "score": 80.1}],
       "blocked": [{"id": "U44", "why": "Fully booked that day"}],
       "loadPct": 16
     }],
     "unassigned": [{
       "t": {"id": "WO4530", "dbId": 4530, "title": "Flat 9C", "...": "..."},
       "day": "2026-11-10",
       "blocked": [{"staffId": "U31", "why": "Fully booked that day", "load": 480, "cap": 480, "freeAt": 1020, "
       "blockedTotal": 12,
       "whyCounts": {"Fully booked that day": 9, "Off: Annual leave": 3},
       "manual": true
     }],
     "days": {
       "2026-11-10": [{"staffId": "U20", "load": 75, "cap": 540, "travel": 0, "jobs": 1,
                       "steps": [{"t": {"id": "WO4521", "...": "..."}, "start": 600, "finish": 675, "tr": 0, "l
     },
     "kpi": {"assigned": 52, "total": 58, "ontime": 98, "travel": 310, "match": 81.2,
             "quality": 86, "balance": 74, "cost": 612.5}
   },
   "baseline": { "mode": "simple", "...": "..." }
 }
}
```


## Atama (assignment) alanları

| Alan | Açıklama |
|---|---|

| | |
|---|---|
| `t` | İş (masadaki alanların aynısı). |
| `day` | İşin günü. |
| `staffId` | Önerilen çalışan (bir `staff` `id`'si). Kişiyi masa verisinden bulun. |
| `manual` | Kişi bir kilitle zorlandıysa `true`. |
| `score` | Son puan: `base` eksi `pen`. |
| `base` | On ölçütün ağırlıklı puanı (0 ile 100 arası). |
| `pen` | Ceza: iş `due` değerinden sonra bitecekse 25, aksi halde 0. |
| `c` | On ölçüt, her biri 0 ile 100 arası: `skill`, `quality`, `speed`, `proximity`, `avail`, `balance`, `reliability`, `continuity`, `fit`, `cost`. Maliyet kullanılmıyorsa `cost` `null` olur. `reasons` aynı değerleri tekrarlar. |
| `tr` | İşten önceki yol süresi, dakika. |
| `trInfo` | Yol ayrıntısı: `m` (dakika), `km`, `mode` (`same`, `walk`, `transit` veya `unknown`); başlangıç yeri bilinmiyorsa `null`. |
| `start`, `finish` | Başlangıç ve bitiş, gece yarısından sonraki dakika olarak. |
| `start_time`, `end_time` | Aynı saatler `HH:MM` olarak. |
| `dur` | İş süresi, dakika. Hizmete ve çalışanın hızına bağlıdır. |
| `late` | `finish`, işin `due` değerinden sonraysa `true`. |
| `cands` | İşi alabilecek çalışanlar ve `score` değerleri; planın sıraladığı sırayla. |
| `blocked` | İşi alamayan çalışanlar ve `why` nedeni. |
| `loadPct` | Bu işten sonra çalışanın o günkü yükü, vardiyasının yüzdesi olarak. |

## Atanamayan (unassigned) alanları

| Alan | Açıklama |
|---|---|
| `t`, `day` | İş ve günü. |
| `manual` | Siz atladıysanız `true`. Yalnızca o zaman bulunur. |
| `blocked` | İşi alamayan **en fazla 5** kişi: `staffId`, `why`, `load`, `cap` (dakika), `freeAt` (gece yarısından sonraki dakika), `off` (izin metni veya `null`). Gösterilen durum, o günün sonundaki durumdur. |
| `blockedTotal` | Toplam kaç kişinin engelli olduğu. |

| | |
|---|---|
| `whyCounts` | Tüm engelli kişiler üzerinden neden metni ve sayısı. "9 kişi dolu, 3 kişi izinli" gibi bir özet için kullanın. |

Yanıtı küçük tutmak için listeler iş başına 5 ile sınırlıdır. Tüm tablo için `blockedTotal` ve `whyCounts` kullanın. Atlanan işte `blocked` listesi boştur ve `blockedTotal` 0'dır.

`why` şunlardan biridir: `Off: <label>`, `Cannot finish within the shift`, `Fully booked that day`, `Usually does not finish this service`. Metin, işlem yapan kişinin dilindedir (`lang`).

### Gün ve KPI alanları

`days[<tarih>]` ekipteki her kişi için bir kayıt içerir: `staffId`, `load` ve `cap` (dakika), `travel` (dakika), `jobs` (bu planın eklediği iş sayısı) ve `steps` (işler sırayla: `t`, `start`, `finish`, `tr`, `late`).

`kpi`:

| Alan | Açıklama |
|---|---|
| `assigned`, `total` | Atanan iş sayısı ve aralıktaki iş sayısı. |
| `ontime` | Zamanında biten atanmış işlerin yüzdesi. |
| `travel` | Toplam yol süresi, dakika. |
| `match` | Ortalama `base` puanı. |
| `quality` | Önceliğe göre ağırlıklandırılmış eşleşme kalitesi. |
| `balance` | Yükün ne kadar dengeli olduğu (0 ile 100 arası, yüksek daha dengeli). |
| `cost` | Tahmini işçilik maliyeti; hiçbir çalışanın maliyeti yoksa `null`. |

## 9. SRVAiSpvCommitAssignment

Bir iş emrini bir çalışana atar. Kabul edilen her öneri için kullanın.

| Alan | Zorunlu mu | Notlar |
|---|---|---|
| `api_key` | **Zorunlu** | Bkz. Kimlik doğrulama. |
| `actor_mopuan_user_id` | **Zorunlu** | Bkz. Kimlik doğrulama. |
| `work_order_id` | **Zorunlu** | İşin `dbId` değeri. |
| `user_id` | **Zorunlu** | Çalışanın `dbId` değeri. Şirketin aktif bir çalışanı olmalıdır. |

```
curl -X POST "https://<api-host>/v1/SRVAiSpvCommitAssignment" \
 -d "api_key=<api_key>" \
```

```
 -d "actor_mopuan_user_id=345" \
 -d "work_order_id=4521" \
 -d "user_id=20"
```

```
{"result": "success", "data": {"work_order_id": 4521, "user_id": 20}}
```

Onay şunları yapar:

1. İş emrini çalışana atar. İşlem yapan kişi, atamayı yapan olarak kaydedilir.
2. Atanan kişi yeniyse ve e-posta adresi varsa ona bildirim gönderir: bir e-posta, OpSense içinde bir bildirim ve çalışan Mopuan'a bağlıysa bir Mopuan push bildirimi.
3. Şirketin "atanınca başlat" kuralı gerektiriyorsa iş emrini başlatır.
4. Bağlı paket işlerini günceller: aşama, teslimat siparişi durumu ve teslimat iş emri; bildirimleriyle birlikte.

2 ile 4. adımlar başarısız olursa atama yine de kaydedilir ve yanıt yine başarılıdır.

**Çakışma reddi (10059).** Onay yalnızca iş emri hâlâ **açıksa** (durum 1) ve **atanmamışsa veya zaten aynı çalışana atanmışsa** kabul edilir. Başkası aldıysa ya da bu arada başlatıldı, bitirildi veya kapatıldıysa çağrı `error_code` 10059 ile başarısız olur ve hiçbir şey değişmez. Aynı çalışanı tekrar onaylamak kabul edilir ve yeniden bildirim göndermez.

10059 aldığınızda: yeniden denemeyin. Masayı yükleyin ve yeniden hesaplayın.

## 10. SRVAiSpvCreateWorkOrder

Atanan kişisi olmayan yeni bir iş emri oluşturur. Tarihi aralıktaysa `tasks` içinde görünür.

| Alan | Zorunlu mu | Notlar |
|---|---|---|
| `api_key` | **Zorunlu** | Bkz. Kimlik doğrulama. |
| `actor_mopuan_user_id` | **Zorunlu** | Bkz. Kimlik doğrulama. |
| `branch_id` | **Zorunlu** | Şirketin bir binası (`locs` `id`). |
| `service_id` | **Zorunlu** | Şirketin bir hizmeti (`services` `id`). |
| `room_id` | İsteğe bağlı | O binanın bir odası. Binaya ait değilse yok sayılır. |
| `priority` | İsteğe bağlı | 1 ile 4 arası. Varsayılan ve geçersizse: 1. |
| `appoinment_date` | İsteğe bağlı | Randevu tarihi ve saati, örneğin `2026-11-12 09:30:00`. Varsayılan (ve okunamazsa): şu an. Alan adı `appoinment_date` olarak yazılır; aynen böyle gönderin. |

```
curl -X POST "https://<api-host>/v1/SRVAiSpvCreateWorkOrder" \
 -d "api_key=<api_key>" \
 -d "actor_mopuan_user_id=345" \
```

```
 -d "branch_id=12" \
 -d "service_id=7" \
 -d "room_id=88" \
 -d "priority=2" \
 -d "appoinment_date=2026-11-12 09:30:00"
```

```
{"result": "success", "data": {"id": 4600}}
```

`id` yeni iş emrinin id'sidir. Hatalar: 10052 (bilinmeyen bina veya hizmet), 10055 (iş emri oluşturulamadı; hiçbir şey kaydedilmedi).

## 11. SRVAiSpvTogglePool

Bir çalışanı planlama ekibinden çıkarır veya geri alır. Seçim şirket için kaydedilir ve siz değiştirene kadar sonraki tüm masa ve hesaplamalarda kullanılır. Birini yalnızca tek bir hesaplama için dışarıda bırakmak için `SRVCalculateAiSpv` üzerindeki `pool` alanını kullanın.

| Alan | Zorunlu mu | Notlar |
|---|---|---|
| `api_key` | **Zorunlu** | Bkz. Kimlik doğrulama. |
| `actor_mopuan_user_id` | **Zorunlu** | Bkz. Kimlik doğrulama. |
| `user_id` | **Zorunlu** | Çalışanın `dbId` değeri. Şirketin aktif bir çalışanı olmalıdır. |
| `excluded` | **Zorunlu** | `1` çalışanı ekipten çıkarır. `0` geri alır. |

```
curl -X POST "https://<api-host>/v1/SRVAiSpvTogglePool" \
 -d "api_key=<api_key>" \
 -d "actor_mopuan_user_id=345" \
 -d "user_id=20" \
 -d "excluded=1"
```

```
{"result": "success", "data": {"user_id": 20, "excluded": true}}
```

## 12. Kalite değeri (quality override)

Elle girilen kalite puanı (0 ile 99 arası), planda o çalışan için KPI puanının yerine geçer. Masa bunu `qualityOverride: true` olarak gösterir.

### SRVAiSpvSetQualityOverride

| Alan | Zorunlu mu | Notlar |
|---|---|---|
| `api_key` | **Zorunlu** | Bkz. Kimlik doğrulama. |

| | | |
|---|---|---|
| `actor_mopuan_user_id` | **Zorunlu** | Bkz. Kimlik doğrulama. |
| `user_id` | **Zorunlu** | Çalışanın `dbId` değeri. Şirketin aktif bir çalışanı olmalıdır. |
| `quality` | **Zorunlu** | 0 ile 99 arası bir tam sayı. |

```
curl -X POST "https://<api-host>/v1/SRVAiSpvSetQualityOverride" \
 -d "api_key=<api_key>" \
 -d "actor_mopuan_user_id=345" \
 -d "user_id=20" \
 -d "quality=90"
```

```
{"result": "success", "data": {"user_id": 20, "quality": 90}}
```

### SRVAiSpvClearQualityOverride

| Alan | Zorunlu mu | Notlar |
|---|---|---|
| `api_key` | **Zorunlu** | Bkz. Kimlik doğrulama. |
| `actor_mopuan_user_id` | **Zorunlu** | Bkz. Kimlik doğrulama. |
| `user_id` | İsteğe bağlı | Çalışanın `dbId` değeri. **Göndermezseniz veya 0 gönderirseniz şirketin tüm elle kalite puanları kaldırılır.** |

```
{"result": "success", "data": {"user_id": 20}}
```

Tüm puanlar kaldırıldığında `data` boş bir listedir.

## 13. SRVAiSpvSaveStaffCost

Bir çalışanın saatlik maliyetini kaydeder. Maliyet `cost` hedefinde ve `kpi.cost` rakamında kullanılır. Payments modülünün kullandığı saatlik ücretin aynısıdır.

| Alan | Zorunlu mu | Notlar |
|---|---|---|
| `api_key` | **Zorunlu** | Bkz. Kimlik doğrulama. |
| `actor_mopuan_user_id` | **Zorunlu** | Bkz. Kimlik doğrulama. |
| `user_id` | **Zorunlu** | Çalışanın `dbId` değeri. Şirketin bir çalışanı olmalıdır. |
| `value` | **Zorunlu** | Saatlik maliyet, 0 ile 9999 arası, 2 ondalığa yuvarlanır. Boş değer maliyeti kaldırır. |

```
curl -X POST "https://<api-host>/v1/SRVAiSpvSaveStaffCost" \
 -d "api_key=<api_key>" \
 -d "actor_mopuan_user_id=345" \
 -d "user_id=20" \
 -d "value=14.50"
```

```
{"result": "success", "data": {"user_id": 20, "value": 14.5}}
```

Maliyet kaldırıldıysa `value` `null` olur. Payments düzenleme yetkisi gerekir (bölüm 4). 10056 hatası, şirketin Payments ücretlerinin kullanılamadığı, bu yüzden maliyetin kaydedilemediği anlamına gelir.

## 14. Hatalar

| error_code | Anlamı | Yeniden denensin mi? |
|---|---|---|
| 10015 | `api_key` eksik | Hayır. İsteği düzeltin. |
| 10016 | `api_key` geçersiz veya pasif | Hayır. Anahtarı kontrol edin. |
| 10030 | `actor_mopuan_user_id` eksik, sayı değil veya şirketin aktif bir çalışanı değil | Hayır. Sunucunuzun gönderdiği değeri düzeltin. |
| 10031 | İşlem yapan kişinin rolünün bu çağrı için yetkisi yok (bölüm 4) | Hayır. |
| 10050 | `goal` altı hedeften biri değil | Hayır. |
| 10051 | `locks`, `skips` veya `pool` geçerli JSON değil, 64 KB veya 5000 kaydı aşıyor ya da hatalı bir anahtar içeriyor | Hayır. |
| 10052 | Bilinmeyen veya başka şirkete ait çalışan, iş emri, bina veya hizmet; ya da geçersiz `work_order_id` veya maliyet alanı | Hayır. |
| 10053 | `excluded` geçerli bir değer değil | Hayır. |
| 10054 | `quality` 0 ile 99 arası bir tam sayı değil | Hayır. |
| 10055 | İş emri oluşturulamadı | Evet, bir kez. Tekrar ederse bizimle iletişime geçin. |
| 10056 | Bu şirket için Payments ücretleri kullanılamıyor | Hayır. |
| 10057 | `value` 0 ile 9999 arası bir sayı değil | Hayır. |
| 10059 | İş emri artık bu atama için açık değil: açık değil veya başka birine atanmış | Hayır. Masayı yenileyin. |
| 10099 | İstek bizim tarafımızda tamamlanamadı | Evet |

## 15. Öneriler

- **Her onaydan sonra yeniden hesaplayın.** Her onay kimde iş olduğunu değiştirir, bu yüzden planın geri kalanı değişir. Sıradaki öneriyi göstermeden önce `SRVCalculateAiSpv` işlemini tekrar çağırın.
- **Her istekte bir iş emri onaylayın.** Çok sayıda onayı toplu göndermeyin. Her sonucu kullanıcıya gösterin.
- **10059'u yenileyerek karşılayın.** İş emrini başkası değiştirmiş olabilir. Kullanıcıya söyleyin, masayı yükleyin ve yeniden hesaplayın. Aynı onayı yeniden denemeyin.
- **Masayı uzun süre saklamayın.** Çalışanlar, vardiyalar ve iş emirleri portalda değişir. Kullanıcı masayı açtığında ve her onaydan sonra yükleyin.
- **Aralıkları kısa tutun.** Genelde bir hafta yeterlidir. En fazla 31 gündür.
- **Planı taslak sayın.** Bir onay başarılı olana kadar hiçbir şey kaydedilmez.
- **`actor_mopuan_user_id` alanını sunucunuzda doldurun,** oturumu açık kullanıcıdan. Telefondan asla.