

Phoenix API

Navodila za razvijalce

Različica dokumenta 1.2 • API v1 • 23. oktober 2026

Ta priročnik opisuje povezavo zunanjih aplikacij s Phoenixom za branje poslovnih partnerjev, zaposlenih in šifrantov ter za ustvarjanje, pregled, urejanje in brisanje prevozov oseb in potnih nalogov. Vključuje vse trenutno izvedene javne zahteve, primere podatkov, odzive, omejitve in ravnanje ob napakah.

Za prvo povezavo pridobite API ključ podjetja in se z upravljavcem Phoenixa dogovorite za testne podatke. V testnem okolju preberite šifrante in opravite poskusni vnos z enolično oznako zahtevka. Pred vključitvijo rednega pošiljanja preverite rezultat tudi v uporabniškem vmesniku Phoenixa.

Vsi ID-ji, GUID-i, nazivi, davčne številke, številke kartic in zneski v primerih so ponazoritve. Nadomestite jih z veljavnimi vrednostmi iz svojega podjetja in poslovnega leta. Noben primer ne vsebuje delujočega API ključa.

Vsebina

- 1 Povezava in prvi zahtevki
- 2 Skupna pravila in odgovori
- 3 Ponavljanje zahtevkov in delo brez povezave
- 4 Branje podatkov in šifrantov
- 5 Prevozi oseb
- 6 Potni nalogi
- 7 Omejitve in obravnava napak
- 8 Primeri odjemalcev
- 9 Pregled vseh zahtevkov

Kazalo je povezano s poglavji. Naslovi so urejeni s slogi za uporabo podokna Krmar v LibreOffice Writerju.

1 Povezava in prvi zahtevek

Testno in produkcijsko okolje

Produkcijski osnovni naslov je:

`https://atena.poslovni.si/lib/api/v1/`

Testni osnovni naslov API-ja je:

`https://had.poslovni.si/phoenix/lib/api/v1/`

Testno okolje Had je na voljo na `https://had.poslovni.si/phoenix/`. Produkcijska aplikacija je na `https://atena.poslovni.si/`. Za API ključ in ustrezne testne podatke se obrnite na upravljavca Phoenixa.

V nadaljevanju oznaka `BASE_URL` pomeni osnovni naslov API-ja izbranega okolja, vključno s končno poševnico. Pred klicem izberite ustrezni naslov, API ključ, podjetje in poslovno leto. Za testiranje uporabite dogovorjene testne podatke. Testnih podatkov ne pošiljajte v produkcijsko podjetje; identifikatorjev zapisov in veljavnosti ključev ne prenašajte samodejno med okoljema.

Uporaba API-ja je brezplačna. Za pridobitev API ključa, dostop do testnega okolja in dogovor o integraciji se obrnite na upravljavca Phoenixa.

API uporablja HTTPS in JSON v UTF-8. Certifikat strežnika vedno preverite. Odjemalec se povezuje samo s Phoenixom; preverjanje ključa pri Pandori izvede strežnik pri vsaki zahtevi. Neposreden dostop do Pandore ali podatkovnih baz ni potreben.

Prijava in izbira podjetja

Vsak zahtevek vsebuje naslednji glavi:

```
Authorization: Bearer TV0J_API_KLJUC
Accept: application/json
```

API ključ je vezan na eno podjetje. Ima ločeni dovoljenji Branje in Pisanje, lahko ima datum veljavnosti ter je lahko preklican. GET zahteva Branje, POST zahteva Pisanje. OAuth in prijava z uporabniškim geslom v tem API-ju nista izvedena.

Podjetja in imena baze ne pošiljajte v zahtevku. Podjetje določi ključ, poslovno leto pa obvezni parameter leto. Phoenix ne izbere samodejno najnovejšega leta. Isti ID ali GUID v drugem podjetju oziroma letu ne predstavlja dovoljenja za dostop do tistega zapisa.

Ključ hranite kot poverilnico posamezne integracije oziroma podjetja. Ne vključite ga v javno izvirno kodo, URL, dnevnike ali skupni namestitveni paket mobilne aplikacije. Ob menjavi ključa upoštevajte pravila ponavljanja iz poglavja 3.

Prvi klic

```
export PHOENIX_API_KEY='TV0J_API_KLJUC'
export PHOENIX_API_URL='https://had.poslovni.si/phoenix/lib/api/v1/'

curl --get "${PHOENIX_API_URL}partnerji.php" \
  --data-urlencode 'leto=2026' \
  --data-urlencode 'na_stran=10' \
  --header "Authorization: Bearer ${PHOENIX_API_KEY}" \
  --header 'Accept: application/json'
```

Primer prvega klica uporablja testno okolje Had. Uporabite API ključ in podatke, dogovorjene za testiranje. Za produkcijo spremenite `PHOENIX_API_URL` na `https://atena.poslovni.si/lib/api/v1/` in uporabite ustrezne produkcijske podatke.

Najprej preverite HTTP status. Pri uspešnem običajnem odgovoru preverite tudi podjetje.davcna_stevilka in leto. Ob praznem seznamu je odgovor še vedno uspešen, podatki je prazen seznam in skupaj je 0.

Priporočen potek prve integracije

1. Pridobite partnerje, zaposlene, vrste prevoza, države in valute iz izbranega leta.
2. Za izbranega zaposlenega preberite načine plačila. Ohranite tako njegov ID kot GUID.
3. V svoji aplikaciji pripravite potrjeno vožnjo in jo razvrstite kot prevoz oseb ali potni nalog. Zasebnih voženj ne pošiljajte samodejno v poslovno evidenco.
4. Pred pošiljanjem trajno shranite URL operacije, leto, vsebino in Idempotency-Key.
5. Po uspehu shranite vrnjeni GUID dokumenta in ID-je postavk. Za nadaljnje spremembe uporabite te identifikatorje.
6. Preverite ponovitev iste zahteve, napačne podatke, izgubljen odgovor in zavrnitev spremembe poknjženega dokumenta.

2 Skupna pravila in odgovori

Metode in oblike podatkov

Branje uporablja GET. Ustvarjanje, urejanje in brisanje uporabljajo POST na različnih naslovih. Metodi PUT in DELETE nista javni operaciji te različice.

Vsak POST zahteva še naslednji glavi:

```
Content-Type: application/json
Idempotency-Key: povezovalac-voznja-2026-000001-ustvari
```

Telo je objekt JSON. Datoteke se pošiljajo v JSON kot Base64, ne kot multipart/form-data. Neznani parametri URL in nedovoljena polja telesa se zavrnejo. Pri POST je v URL dovoljen samo leto; filtre in identifikatorje operacije pošljite v telesu.

Podatek	Oblika in pravilo
Poslovno leto	Parameter leto, štiri številke, od 1900 do 2199; dejanska baza mora obstajati
Datum	Niz YYYY-MM-DD
Čas prevoza oseb	HH:MM ali HH:MM:SS
Odhod in vrnitev potnega naloga	YYYY-MM-DD HH:MM ali YYYY-MM-DDTHH:MM; vhod brez sekund
Lokalni čas	Konvencija Phoenixa Europe/Ljubljana; brez samodejnega pošiljanja UTC ali časovnega odmika
Decimalna števila	Decimalna pika, brez ločil tisočic; v vhodnem JSON priporočeno število
GUID	Niz oblike 11111111-2222-3333-4444-555555555555
timestamp_lastedit	Natančen 20-mestni niz; ohranite kot niz
Logična vrednost	JSON true ali false, kadar jo polje zahteva

Pri potnih nalogih veljajo tudi omejitve obstoječih časovnih polj; datumi odhoda in vrnitve izven podprtega obsega TIMESTAMP se zavrnejo. Neveljaven lokalni čas ob spomladanskem premiku ure se zavrne. Čase iz GPS pred pošiljanjem pretvorite v zahtevani lokalni zapis.

Odzivi niso povsem poenoteni glede decimalnih tipov. Ceniki dnevnic in kilometrina v seznamu prevozov so decimalni nizi, poslovni zneski potnega naloga pa JSON števila. Za denarne izračune v odjemalcu uporabljajte decimalno aritmetiko in prikaz ločite od prenosa podatkov.

Skupni parametri seznamov

Parametri spodaj veljajo za vse javne sezname. Pregled posameznega potnega naloga `pn_nalog.php` je izjema in ne uporablja strani, iskanja ali štetja.

Parameter	Obvezen	Pomen
leto	Da	Poslovno leto
stran	Ne	Pozitivno celo število, privzeto 1
na_stran	Ne	Od 1 do 100, privzeto 50
iskanje	Ne	Največ 150 znakov; področje iskanja določa seznam
samo_stevilo	Ne	0 privzeto; 1 vrne samo število zadetkov

Posamezni seznamski imajo dodatne filtre, opisane v svojih poglavjih. Iskanje ne omogoča poljubnih SQL pogojev. Znakoma `%` in `_` v podatkovnih iskanjih ni pripisan pomen nadomestnega znaka.

Uspešen seznam

Naslednji primer vsebuje enega poslovnega partnerja in običajno ovojnico odgovora:

```
{
  "stran": 1,
  "na_stran": 50,
  "skupaj": 1,
  "podatki": [
    {
      "id": 10,
      "guid": "11111111-2222-3333-4444-555555555555",
      "sifra": "P001",
      "naziv": "Primer d.o.o.",
      "naslov": "Primerna ulica 1",
      "postna_stevilka": "1000",
      "kraj": "Ljubljana",
      "drzava": "Slovenija",
      "davcna_oznaka": "SI",
      "davcna_stevilka": "12345678",
      "ddv_zavezanec": true,
      "email": "kontakt@example.com",
      "telefon": "",
      "timestamp_lastedit": "20261022083000123450"
    }
  ],
  "uspeh": true,
  "zahteva_id": "0123456789abcdef0123456789abcdef",
  "podjetje": {"davcna_stevilka": "12345678"},
  "leto": 2026
}
```

skupaj šteje vse zadetke filtra, ne samo trenutne strani. Oznaka `zahteva_id` je tudi v odzivni glavi `X-Request-ID` in je namenjena sledenju tehničnemu klicu.

Samo število zadetkov

```
GET partnerji.php?leto=2026&samo_stevilo=1
GET zaposleni.php?leto=2026&samo_stevilo=1
```

Uspešno telo vsebuje samo:

```
{"skupaj": 103}
```

V tem načinu ni ovojnice s podjetjem, letom ali uspeh; X-Request-ID ostaja v glavi. Filtri še vedno veljajo, strani pa ne omejujejo štetja. Odgovori napak ostajajo v standardni obliki. Štetje samo po sebi ne pove, ali so se podatki partnerjev spremenili.

Napaka

```
{
  "uspeh": false,
  "zahteva_id": "0123456789abcdef0123456789abcdef",
  "napaka": {
    "oznaka": "NALOG_JE_POKNJIZEN",
    "sporocilo": "Knjiženega naloga ni dovoljeno spreminjati ali izbrisati."
  }
}
```

Za obravnavo uporabite HTTP status in napaka.oznaka. Besedilo sporočila je namenjeno prikazu oziroma razlagi in se lahko spremeni. Posredni strežnik lahko pri prevelikem telesu ali nedosegljivi storitvi vrne tudi odgovor, ki ni JSON; odjemalec mora obravnavati tudi ta primer.

3 Ponavljanje zahtevkov in delo brez povezave

Pomen oznake Idempotency Key

Vsak nov poslovni dogodek dobi svojo oznako Idempotency-Key. Oznaka ima od 1 do 128 znakov in sme vsebovati črke ASCII, številke, piko, vezaj, podčrtaj in dvopičje. Lastni ID vožnje lahko vključite v oznako, na primer mobilna-voznja-482-ustvari. Za naslednjo spremembo te vožnje uporabite novo oznako, na primer mobilna-voznja-482-uredi-1.

Če isti zahtevek pošljete ponovno na isti naslov z istim API ključem, letom, oznako in vsebino, Phoenix vrne prvotni poslovni rezultat s ponovitev=true. Uspešnega vnosa ne izvede še enkrat. Nova zahteva ima svoj tehnični zahteva_id, poslovni rezultat pa se nanaša na prvotni dogodek.

Druga vsebina ali druga operacija ob isti oznaki vrne HTTP 409

PONOVLJENA_OZNAKA_Z_DRUGO_VSEBINO. Pri ponovitvi ohranite tudi JSON tipe, zaporedje seznamov in vsebino datotek. Najzanesljiveje je ponovno poslati že shranjeno telo zahteve.

Oznaka je vezana na identiteto API ključa v izbranem podjetju in poslovnem letu. Pri menjavi ključa se ne zanašajte na deduplikacijo stare oznake: najprej preverite že ustvarjeni dokument. Idempotency-Key ni isto kot zahteva_id, GUID dokumenta ali ID postavke. Posebno polje zunanji_id in iskanje dokumenta po njem trenutno nista izvedena.

Čakalna vrsta mobilne aplikacije

1. Potrjeno vožnjo shranite lokalno skupaj s podatki, datotekami in oznako operacije.
2. Ko je omrežje na voljo, pošljite shranjeno zahtevo.
3. Ob uspehu trajno zapišite vrnjeni GUID dokumenta, ID-je postavk in potrditev dogodka.
4. Ob prekinitvi ali izgubljenem odgovoru ponovite isto zahtevo z isto oznako. Nova oznaka lahko povzroči nov dokument.
5. Med urejanjem ustvarite nov poslovni dogodek in novo oznako; stare potrjene vsebine ne spreminjajte za nazaj.
6. Zahtevke za isti dokument obdelujte zaporedno. Pri potnih nalogih pred zamenjavo seznamov pridobite svež GET, da z zastarelim seznamom ne prepišete vmesne spremembe.

Ob začasnih napakah uporabite postopno podaljševanje presledka med poskusi. Ob 400, 401, 403, 404, 409 ali 422 najprej preučite napako; nespremenjenega zahtevka ne ponavljajte v neskončnost.

ZAHTEVEK_ZA_PREVERJANJE ali NEPOPOLN_ZAHTEVEK pomenita, da rezultat prejšnjega poskusa zahteva pregled pri upravljavcu. Ne obidite ju z novo oznako. Ob izpadu lahko zaradi obstoječega načina shranjevanja ostane delno stanje; API ne obljublja, da je vsak neuspešen klic pustil vse podatke nedotaknjene.

Uspešna ponovitev vrne prvotni rezultat tudi, če je bil dokument pozneje zaključen, poknjižen ali izbrisan. Tak odgovor potrjuje prvotni dogodek. Za trenutno stanje izvedite GET; ponovitev ustvarjanja izbrisanega dokumenta ne obnovi.

4 Branje podatkov in šifrantov

Vsi spodnji naslovi so relativni glede na BASE_URL. Veljajo skupni parametri seznamov in dovoljenje Branje.

Poslovni partnerji

```
GET partnerji.php?leto=2026
GET partnerji.php?leto=2026&iskanje=Primer
GET partnerji.php?leto=2026&spremenjeno_od=2026-10-22
```

Dodatni parameter spremenjeno_od omeji rezultat po timestamp_lastedit. Iskanje deluje po nazivu, davčni številki in šifri. Vrnjeni so neizbrisani partnerji, razvrščeni po id naraščajoče.

Polja elementa podatki	Tip oziroma pomen
id	Celo število, ID partnerja
guid	Niz, identifikator partnerja
sifra, naziv	Niza, šifra in naziv
naslov, postna_stevilka, kraj, drzava	Naslovni podatki
davnca_oznaka, davnca_stevilka	Davčni podatki partnerja
ddv_zavezanec	Logična vrednost
email, telefon	Kontaktna podatka
timestamp_lastedit	Natančen niz iz evidence; pri starejših zapisih je lahko prazen ali null

Besedilna polja se vračajo iz obstoječe evidence in lahko vsebujejo prazno vrednost ali null. Celoten primer je v poglavju 2.

Spremembe partnerjev od določenega datuma

Filter uporablja izključno timestamp_lastedit. Sprejme naslednje oblike:

spremenjeno_od	Pomen
2026-10-22	Od začetka dneva
2026-10-22T08:30:00	Od začetka navedene sekunde
2026-10-22 08:30:00	Enako; presledek v URL kodirajte kot %20
20261022083000123450	Od natančne 20-mestne oznake

Velja primerjava večje ali enako. Tudi partner točno na meji se vrne. Oznaka ni Unix čas; ne pretvarjajte je v število s plavajočo vejico in ne pošiljajte pripone Z ali odmika časovnega pasu.

```
GET partnerji.php?leto=2026&spremenjeno_od=20261022083000123450
GET partnerji.php?leto=2026&spremenjeno_od=2026-10-22&samo_stevilo=1
```

Ob prvi sinhronizaciji preberite vse strani brez filtra. Pri naslednjih branjih uporabite časovni prekrivni interval in združujte po GUID-u. Med branjem vseh strani ohranite isti filter; meje ne premikajte po vsaki strani. Zadnji element strani ni nujno najnovejša sprememba, ker je vrstni red po ID.

Filter bere trenutno stanje in ni dnevnik sprememb ali zamrznjen posnetek več strani. Izbrisani partnerji se ne vrnejo, prav tako pri uporabi filtra niso vključeni zapisi s praznim ali nepravilnim timestampom. Sprememba, pri kateri obstoječi program ne posodobi timestamp_lastedit, ni zaznavna. Za uskladitev izbrisov in takšnih zapisov občasno ponovno preberite celoten seznam.

Zaposleni

```
GET zaposleni.php?leto=2026
GET zaposleni.php?leto=2026&aktiven=vsi
GET zaposleni.php?leto=2026&iskanje=Ana
```

Dodatni parameter aktiven je 1 za aktivne privzeto, 0 za neaktivne ali vsi za vse. Iskanje deluje po imenu in priimku v obeh vrstnih redih, šifri, e-pošti in delovnem mestu. Vrstni red je priimek, ime in ID.

Polja elementa podatki	Tip oziroma pomen
id	Celo število; uporabite za voznika, sovoznika in delavca pri vnosu
guid	Niz; uporabite pri branju načinov plačila
sifra, ime, priimek	Osnovni podatki zaposlenega
email, email2, telefon, telefon2	Kontaktni podatki
delovno_mesto	Besedilni opis
datum_zaposlitve, datum_preinitve	Datum ali null za prazen oziroma ničelni datum
aktiven	Logična vrednost iz evidence aktivnosti

API ne vrača EMŠO, davčne številke zaposlenega, plače ali gesel. Davčna številka v ovojnici označuje podjetje. Numeričnega ID-ja in GUID-a zaposlenega ne zamenjajte.

Države z dnevnicami

```
GET drzave.php?leto=2026
GET drzave.php?leto=2026&iskanje=Avstrija
GET drzave.php?leto=2026&drzava_id=10
```

Dodatni filter drzava_id je pozitiven številčni ID države. Vrne vse države s slovenskimi nazivi, razvrščene po nazivu in ID-ju. Iskanje deluje po nazivu in oznaki. Dnevnice so že vključene, zato ločen klic cenika ni nujen.

Polja elementa podatki	Tip oziroma pomen
id, naziv, oznaka	Številčni ID, naziv in kratica države
eu	Logična vrednost

Polja elementa podatki	Tip oziroma pomen
edavki_koda	Oznaka iz evidence
valuta	Oznaka valute dnevnice; lahko prazna
dnevnic_6_8	Decimalni niz cenika
dnevnic_8_12	Decimalni niz cenika
dnevnic_12_24	Decimalni niz cenika

ID 10 je samo primer; državo poiščite v dejanskem seznamu. Zneski so shranjene vrednosti cenika z decimalno piko, na primer "12.30". API s tem branjem ne potrjuje aktualnosti cenika in ne izračunava konkretnega potovanja. Poimenovanje intervalov v tem seznamu ne spreminja obstoječih pravil avtomatskega obračuna iz poglavja 6.

Ločen seznam dnevnic

GET pn_dnevnice.php?leto=2026&drzava_id=10

Podpira iste dodatne filtre in iskanje kot države. Polja elementa so drzava_id, drzava, oznaka_drzave, valuta, dnevnic_6_8, dnevnic_8_12 in dnevnic_12_24. Decimalni zneski so nizi. Vir cenika je isti kot pri državah; ta klic ostaja na voljo za odjemalce, ki želijo samo dnevnice.

Vrste prevoza

GET pn_prevoz.php?leto=2026

Element vsebuje id kot celo število in naziv kot niz. Iskanje deluje po nazivu, razvrstitev po nazivu in ID-ju. Ta ID uporabite kot vozilo pri prevozu oseb oziroma sredstva pri potnem nalogu. Ne pošiljajte poljubnega naziva namesto ID-ja.

Vrste prilog

GET pn_vrste_prilog.php?leto=2026

Seznam ima fiksne vrednosti v spodnjem vrstnem redu. Podpira tudi iskanje, strani in štetje. Prazna možnost Izberi ni vključena.

sifra	naziv
drugo	Drugo
prevoz	Prevoz
spanje	Prenočišče

Načini plačila za zaposlenega

GET pn_nacini_placila.php?leto=2026&zaposleni_guid=GUID_ZAPOSLENEGA

Dodatni parameter zaposleni_guid je obvezen in mora biti GUID iz GET zaposleni. Vrneta se Gotovina in Nakazilo TRR ter vse kartice, ki pripadajo temu zaposlenemu. Tudi zaposleni brez kartic ima prvi dve možnosti. Ob iskanju lahko filter izloči tudi ti fiksni možnosti.

Polje elementa	Pomen
----------------	-------

Primeri polja vrednost so 1/////Gotovina, 2/////Nakazilo TRR in 3///GUID_KARTICE///IME_KARTICE. Za zapis stroška uporabite ločeni polji tip in kartica_guid; razčlenjevanje vrednost ni potrebno. Pri 1 in 2 v POST pošljite nacin_placila_guid kot prazen niz, ne null.

Kartice sledijo po svojem ID-ju. Iskanje deluje po nazivih in številki kartice. Neznani zaposleni vrne 404, neveljaven ali manjkajoč GUID pa 400. Odgovor vsebuje celotno številko kartice, zato ga ne vključujte v javne dnevnike.

Valute

```
GET valute.php?leto=2026
GET valute.php?leto=2026&iskanje=USD
```

Element vsebuje id kot celo število ter oznaka, koda, naziv in država iz evidence valut. Iskanje deluje po oznaki, nazivu in državi; razvrstitev po oznaki in ID-ju. Primer elementa:

```
{
  "id": 2,
  "oznaka": "USD",
  "koda": "840",
  "naziv": "Ameriški dolar",
  "drzava": "ZDA"
}
```

Pri vnosu stroška uporabite valuta="USD", torej oznaka, ne id ali koda. Samostojni javni seznam tečajev trenutno ni izveden; uporabljeni tečaj se vrne pri shranjenem strošku.

5 Prevozi oseb

Prevoz oseb ima skupno glavo, eno ali več vrstic voženj in opsijske datoteke. Nov prevoz mora imeti vsaj eno vrstico. Kasneje mu lahko dodajate vrstice in priloge, dokler ni poknjizen. Glavni identifikator je prevoz_guid.

Pregled glav prevozov

```
GET po_prevozi.php?leto=2026
GET po_prevozi.php?leto=2026&voznik=5&mesec=2026-10&poknjizen=0
GET po_prevozi.php?leto=2026&guid=11111111-2222-3333-4444-555555555555
```

Dodatni filter	Pomen
voznik	Pozitiven številčni ID zaposlenega
mesec	YYYY-MM iz zahtevanega leta; mesec polja obracun
guid	GUID prevoza; v tem GET je ime parametra guid
poknjizen	0 neknjiženi, 1 knjiženi, vsi privzeto

Iskanje deluje po imenu voznika in smeri, vrstni red je st in id naraščajoče. Podprti so skupni parametri seznamov in samo_stevilo. Pridobivanje po GUID-u še vedno vrne seznam; če zadetka ni, je seznam prazen.

Element podatki vsebuje id, guid, st, datum, izdan_v, obracun, voznik, voznik_ime, sovoznik, sovoznik_ime, vozilo, vozilo_ime, reg, smer, kilometrina, poknjizen, vrsta, posel, priloge in vrsta_po. ID-ji, st, vrsta in poknjizen so števila, sovoznik je lahko null, kilometrina je decimalni niz. Polje priloge v glavi je besedilo; datoteke se berejo posebej.

Ustvarjanje prevoza

POST po_ustvari.php?leto=2026 vrne HTTP 201. Telo vsebuje obvezni glava in vrstice ter opsijski priloge za datoteke.

```
{
  "glava": {
    "datum": "2026-10-22",
    "izdan_v": "Ljubljana",
    "obracun": "2026-10-31",
    "voznik": 5,
    "vozilo": 12
  },
  "vrstice": [
    {
      "datum": "2026-10-22",
      "cas_odhoda": "08:05",
      "cas_prihoda": "11:15",
      "kraj_odhoda": "Ljubljana",
      "kraj_prihoda": "Maribor",
      "km": 130,
      "stevec_start": 15000,
      "stevec_konec": 15130,
      "oseb": 2
    }
  ]
}
```

Polje glava	Obvezno	Pravilo
datum	Da	Datum izdaje v izbranem letu
izdan_v	Da	Kraj izdaje, neprazno besedilo, največ 100 znakov
obracun	Da	Datum obračuna v izbranem letu; ne pred datumom izdaje
voznik	Da	Številčni ID zaposlenega
vozilo	Da	Številčni ID iz pn_prevoz
sovoznik	Ne	Številčni ID zaposlenega; brez njega privzeto prazno
vrsta	Ne	Celo število 0–127, privzeto 0
smer, posel, priloge	Ne	Besedilo do 255 znakov; privzeto prazno
reg	Ne	Besedilo do 50 znakov; privzeto prazno
vrsta_po	Ne	Besedilo do 77 znakov; privzeto prazno

Phoenix sam ustvari GUID in naslednjo številko st, dopolni imena voznika, sovoznika in sredstva ter prebere kilometrino iz nastavitvev. Podatka user in poknjizen ne pošiljajte; nov prevoz je neknjižen. Kilometrine ne nastavljate sami.

Polje vrstice	Obvezno	Pravilo
datum	Da	Datum izbranega leta med datumom glave in obračunom
cas_odhoda, cas_prihoda	Da	HH:MM ali HH:MM:SS
kraj_odhoda, kraj_prihoda	Da	Neprazno besedilo, največ 255 znakov

Polje vrstice	Obvezno	Pravilo
km	Da	Pozitivno število
stevec_start, stevec_konec	Ne	Nenegativni stanji števca, privzeto 0
oseb	Ne	Celo število 0–99999, privzeto 0

Posamezen zahtevek vsebuje od 1 do 500 vrstic. Km in stanji števca so največ 1000000000. Če sta obe stanji števca pozitivni, končno ne sme biti manjše od začetnega. Števca ne nadomestita km; podatek km pošljite izrecno.

Glava in vrstice imajo ločena datumska pravila. Za vožnje, ki prečkajo polnoč, pred vključitvijo uskladite razdelitev vrstic, saj ima vrstica samo en datum in dva časa; ne dodajajte izmišljenega polja za datum prihoda.

Rezultat ustvarjanja in dodajanja

```
{
  "prevoz": {
    "id": 3,
    "guid": "11111111-2222-3333-4444-555555555555",
    "st": 3,
    "kilometrini": "0.43"
  },
  "dodano_vrstic": 1,
  "vrstice_ids": [7],
  "ponovitev": false,
  "uspeh": true,
  "zahteva_id": "0123456789abcdef0123456789abcdef",
  "podjetje": {"davcna_stevilka": "12345678"},
  "leto": 2026
}
```

Če zahtevek vsebuje datoteke, rezultat doda dodano_prilog in priloge z metapodatki novih datotek. Shranite prevoz.guid in vrstice_ids. Končni zapis preverite s po_prevozi.php in po_vrstice.php. Ne sklepajte o identiteti dokumenta samo iz njegove zaporedne številke.

Dodajanje vrstic obstoječemu prevozu

POST po_dodaj_vrstice.php?leto=2026 vrne HTTP 200.

```
{
  "prevoz_guid": "11111111-2222-3333-4444-555555555555",
  "vrstice": [
    {
      "datum": "2026-10-24",
      "cas_odhoda": "09:00",
      "cas_prihoda": "10:00",
      "kraj_odhoda": "Ljubljana",
      "kraj_prihoda": "Kranj",
      "km": 30
    }
  ]
}
```

Namesto prevoz_guid lahko uporabite izbor po vozniku in mesecu obračuna:

```
{
  "izbor": {
    "voznik": 5,
    "mesec": "2026-10",
    "vozilo": 12,
    "prvi": false
  },
  "vrstice": [
    {
      "datum": "2026-10-24",
      "cas_odhoda": "09:00",
      "cas_prihoda": "10:00",
      "kraj_odhoda": "Ljubljana",
      "kraj_prihoda": "Kranj",
      "km": 30
    }
  ]
}
```

voznik in mesec sta v izbor obvezna, vozilo je opcijski filter. Izbira išče neknjižene prevoze. Če jih je več, vrne 409 VEC_PREVOZOV, razen če je prvi=true; tedaj izbere prvega po st in ID naraščajoče. Če ni nobenega, vrne 404 in je treba prevoz posebej ustvariti. prevoz_guid in izbor se izključujeta. Številka in že shranjena kilometrina prevoza ostaneta isti.

Hkrati lahko dodate vrhnji seznam priloge za nove datoteke. Zahtevke ne zamenjuje starih vrstic, ampak jim doda nove. Knjiženemu prevozu ni dovoljeno dodajati vrstic ali datotek.

Pregled vrstic

```
GET po_vrstice.php?leto=2026&prevoz_guid=GUID_PREVOZA
GET po_vrstice.php?leto=2026&prevoz_guid=GUID_PREVOZA&vrstica_id=7
```

Obvezni dodatni parameter je prevoz_guid, opcijski vrstica_id omeji rezultat na postavko tega prevoza. Veljajo skupni parametri seznamov. Iskanje deluje po krajih odhoda in prihoda. Vrstni red je datum, čas odhoda in ID.

Odgovor vsebuje še prevoz z id, guid in st ter elemente podatki s polji id, header, datum, cas_odhoda, cas_prihoda, kraj_odhoda, kraj_prihoda, stevec_start, stevec_konec, km in oseb. header je GUID prevoza. ID in oseb sta celi števili, km in stanji števca so števila. Neobstoječ prevoz vrne 404, odsotna postavka obstoječega prevoza pa prazen seznam. Branje je dovoljeno tudi po knjiženju.

Urejanje glave

POST po_uredi.php?leto=2026 vrne HTTP 200.

```
{
  "prevoz_guid": "11111111-2222-3333-4444-555555555555",
  "glava": {
    "reg": "LJ-AB-123",
    "izdan_v": "Kranj",
    "sovoznik": null
  }
}
```

Pošljite vsaj eno dovoljeno polje glave iz ustvarjanja. Neposlana polja ostanejo. sovoznik=null ali prazen niz ga odstrani; neobvezno besedilo počistite s praznim nizom. Ob spremembi zaposlenega ali sredstva se ime dopolni iz evidence.

GUID, številka, kilometrina in knjiženost se ne urejajo. Po spremembi datumov morajo vse obstoječe vrstice še vedno ustrezati obdobju glave. Knjižen prevoz vrne 409 PREVOZ_JE_POKNJIZEN. Uspeh vsebuje prevoz, operacija="uredi_glavo", ponovitev in običajno ovojnico; sveže podatke preberite z GET.

Urejanje in brisanje vrstice

POST po_uredi_vrstico.php?leto=2026:

```
{
  "prevoz_guid": "11111111-2222-3333-4444-555555555555",
  "vrstica_id": 7,
  "vrstica": {"cas_prihoda": "12:30", "km": 140, "oseb": 3}
}
```

Pošljite vsaj eno dovoljeno polje vrstice iz ustvarjanja. Veljajo ista pravila za končno združeno vrstico. Postavke ni mogoče prestaviti na drug prevoz.

POST po_izbrisi_vrstico.php?leto=2026:

```
{
  "prevoz_guid": "11111111-2222-3333-4444-555555555555",
  "vrstica_id": 7
}
```

Obe operaciji vrneta HTTP 200 z identiteto prevoz, vrstica_id, ponovitev in operacija="uredi_vrstico" oziroma "izbrisi_vrstico". ID postavke mora pripadati navedenemu prevozu, sicer je odgovor 404 VRSTICA_NI_NAJDENA. Knjiženost prepreči obe operaciji. Izbris zadnje vrstice je dovoljen in pusti prazno glavo. Prevozu za izbris postavke ni treba biti zadnji po številki.

Datoteke ob ustvarjanju in naknadno

Prevozu lahko dodate priloge ob po_ustvari.php, ob po_dodaj_vrstice.php ali samostojno s POST po_dodaj_priloge.php?leto=2026. Vrhnji seznam priloge vsebuje datoteke v naslednji obliki:

```
{
  "prevoz_guid": "11111111-2222-3333-4444-555555555555",
  "priloge": [
    {
      "date": "2026-10-22",
      "description": "Parkirnina",
      "ime_datoteke": "Parkirni racun.jpg",
      "vsebina_base64": "TUKAJ_DEJANSKA_BASE64_VSEBINA"
    }
  ]
}
```

Vsa štiri polja priloge so obvezna. description in ime_datoteke imata največ 255 znakov. Ime vsebuje le osnovo in končnico, brez poti. Pri samostojnem dodajanju je obvezna vsaj ena priloga; vrstic ne pošiljate. Pri ustvarjanju in dodajanju vrstic je seznam opsijski. glava.priloge je ločeno staro besedilno polje in ni seznam datotek.

Podprti so PDF, JPG/JPEG, PNG in WEBP. Vsebina mora ustrezati končnici, slike morajo biti prepoznavne. Base64 pošljite brez predpone data: in z dejansko vsebino datoteke. Omejitve velikosti so v poglavju 7.

Odgovor ustvarjanja ali dodajanja vsebuje še dodano_prilog in seznam priloge z id, guid kot novim imenom datoteke, date, description, mime in velikost v bajtih. Pri samostojnem dodajanju je dodano_vrstic=0 in vrstice_ids=[]. Nova datoteka dobi samodejno ime in je vidna v obstoječem pregledu Phoenixa.

Pregled prilog

```
GET po_priloge.php?leto=2026&prevoz_guid=GUID_PREVOZA
GET po_priloge.php?leto=2026&prevoz_guid=GUID_PREVOZA&samo_stevilo=1
```

Obvezni parameter je prevoz_guid. Veljajo skupni parametri seznamov. Iskanje deluje po opisu in imenu datoteke, razvrstitev je po ID. Odgovor vsebuje prevoz ter podatki z id, guid kot imenom datoteke, description in date.

Ta GET vrača metapodatke, ne bajtov datotek ali samostojne povezave za prenos. Datoteke odprete v Phoenixu; namenski javni prenos datotek skozi API trenutno ni izveden. Pregled je dovoljen tudi po knjiženju. Samostojna API operacija za brisanje posamezne priloge prevoza oseb trenutno ni izvedena.

Brisanje celotnega prevoza

POST po_izbrisi.php?leto=2026:

```
{"prevoz_guid": "11111111-2222-3333-4444-555555555555"}
```

Izbris je dovoljen samo za zadnji prevoz po številki st, če ni poknjižen. Višja številka katerega koli drugega prevoza prepreči izbris starejšega. Gre za dejanski izbris glave, vrstic in pripadajočih prilog; lastne nedeljene datoteke se odstranijo. Deljene datoteke ali datoteke drugega izvirnega leta se lahko ohranijo.

Odgovor HTTP 200 vsebuje prevoz z id, guid in st, operacija="izbrisi_prevoz", ponovitev, izbrisano_vrstic, izbrisano_prilog, ohranjeno_deljenih_datotek in ciscenje_prilog. Za trenutno stanje datotek velja ciscenje_prilog="zakljuceno" ali "potreben_pregled"; pri slednjem obvestite upravljavca in ne ustvarjajte dvojnika zahtevka.

Sproščena številka se lahko ponovno uporabi, novi dokument pa dobi nov GUID. Za dolgoročno povezavo uporabite GUID, ne številke ali domnevno naslednjega ID-ja.

6 Potni nalogi

Potni nalog vsebuje glavo, odseke poti, samodejno izračunane dnevnic in stroške. Vsak strošek lahko vsebuje eno datoteko. Dnevnice ali končnih vsot odjemalec ne vpisuje sam. Glavni identifikator je pn_guid.

Ustvarjanje naloga

POST pn_ustvari.php?leto=2026 vrne HTTP 201. Primer vsebuje izrecno povratno pot, zato dodatna samodejna vrnitev ni potrebna:

```
{
  "glava": {
    "delavec": 5,
    "datum": "2026-10-22 08:30",
    "datum_konca": "2026-10-22 17:00",
    "sredstva": 12,
    "obracun": "2026-10-22",
    "drzava": 159,
    "izstava": "2026-10-22"
  },
  "kraji": [
    {
      "start": "Ljubljana",
      "end": "Maribor",
      "km": 130,
      "dest": "Obisk partnerja Primer d.o.o."
    },
    {
      "start": "Maribor",
      "end": "Ljubljana",
      "km": 130,
      "dest": "Vrnitev"
    }
  ],
  "priloge": [
    {
      "priloga": "Parkirnina",
      "value": 5,
      "nacin_placila": 1,
      "nacin_placila_guid": "",
      "valuta": "EUR",
      "vrsta_priloge": "prevoz"
    }
  ]
}
```

Podatki glave

Polje	Obvezno	Pravilo
delavec	Da	Številčni ID iz GET zaposleni
datum	Da	Odhod kot datum in čas brez sekund
datum_konca	Da	Vrnitev kot datum in čas brez sekund; ne pred odhodom
sredstva	Da	Številčni ID iz GET pn_prevoz
obracun	Da	Datum v izbranem letu; ne pred dnem vrnitve
drzava	Da	Številčni ID iz GET drzave
izstava	Da	Datum izdaje
predujm	Ne	Nenegativen znesek v EUR, največ 999999.99, do dve decimalki; privzeto 0
predujm_datum	Pogojno	Datum; obvezen pri pozitivnem predujmu
dodatki	Ne	Besedilo do 255 znakov, privzeto prazno

Leto zahtevka določa datum obračuna, ne nujno leto začetka potovanja. Potovanje se lahko začne v predhodnem letu. Za datum in datum_konca API shrani sekunde 00 in jih v odzivu tudi vrne.

Phoenix iz zaposlenega dopolni ime, naslov, delovno mesto, vozilo in registracijo. Iz nastavitve dopolni nalog, naloga in kilometrino st_km_po. Ustvari guid, naslednji num in st_naloga, na primer 18/2026. Največja podprta številka je 99999. odhod je enak datum, prihod pa datum_konca. Nov nalog ima obracunan=1, zaključen=0 in poknjizen=0. Obračunan ne pomeni poknjen.

Partnerja izberite v svoji aplikaciji iz GET partnerji in njegov naziv vključite v dest. Neposrednega vhodnega polja za ID partnerja ni. Besedili nalog in naloga izhajata iz nastavitve in nista vhodni polji. Namen posameznega odseka navedite v dest; dodatne opombe lahko pošljete v dodatki.

Poti in samodejna vrnitev

kraji je seznam od 1 do 500 odsekov. Vsak odsek zahteva spodnja polja:

Polje	Pomen
start	Kraj odhoda, neprazno besedilo do 255 znakov
end	Kraj prihoda, neprazno besedilo do 255 znakov
km	Nenegativno število z največ dvema decimalkama, največ 999999.99
dest	Cilj oziroma namen odseka, neprazno besedilo do 255 znakov

Konec prejšnjega odseka se mora ujemati z začetkom naslednjega; primerjava ne razlikuje velikosti črk. Če se pot ne konča v prvem začetnem kraju, Phoenix sam doda odsek od zadnjega kraja do prvega, z dest="Vrnitev". Razdaljo poišče v obstoječi evidenci, najprej v tej smeri in nato v obratni. Če je ni, uporabi 0 km. Če vrnitev pošljete sami, se ne doda še enkrat.

Vsota st_km vključuje vse odseke in morebitno samodejno vrnitev. Če želite obračunati natančno GPS razdaljo v obe smeri, pošljite obe smeri izrecno. API ne izračunava poti prek zemljevidov ali GPS. Posamezni km in skupna razdalja morajo ostati v obsegu 999999.99.

Samodejne dnevnic

API uporablja obstoječo logiko Phoenixa za slovenski obračun. Odjemalec ne pošilja seznama dnevnic ali njihovega skupnega zneska. Izračun temelji na odhodu, vrnitvi, državi in shranjenem ceniku.

Minute se pri izračunu odštevanja ur odrežejo, enako kot v sedanjem obrazcu. Primer 08:59–16:00 se obravnava kot 8 ur. Pri večdnevni poti se upoštevajo vmesni polni koledarski dnevi ter posebej ure prvega in zadnjega dne.

Za Slovenijo, ID 159, se preostali interval nad 12 ur spremeni v polno dnevnic, za tujino nad 14 ur. Ostali intervali od 8 ur dobijo srednjo in od 6 ur kratko dnevnic. To so dejanska pravila trenutnega obračuna, tudi če imajo javna polja cenika drugačna imena intervalov.

Če valuta izbranega cenika ni nastavljena, se uporabi nadomestni cenik ID 211; izbrana država ostane na vrstici. Zajtrk je pri samodejnem vnosu 0. Pri tuji valuti API uporabi zadnji dodani tečaj in izračuna EUR z deljenjem s tečajem ter množenjem s količino. To ni samodejno pridobivanje dnevnega tečaja iz zunanje storitve.

V rezultatu dnevnic vrsta_dnevnice uporablja obstoječe identifikatorje dnevnic_6_10, dnevnic_10_14 in dnevnic_14_24. GET cenika uporablja imena dnevnic_6_8, dnevnic_8_12 in dnevnic_12_24. Teh dveh oblik ne zamenjajte in polj dnevnic ne pošiljajte nazaj kot vhod.

Stroški in tečaji

Vrhnji priloge je seznam največ 100 stroškov. Pri ustvarjanju ga lahko izpustite ali pošljete []. Vsak strošek zahteva:

Polje	Pomen
priloga	Neprazen opis stroška, do 255 znakov
value	Znesek izvirne valute, nenegativen, največ 999999.99, do dve decimali
nacin_placila	1 gotovina, 2 TRR, 3 kartica
nacin_placila_guid	Pri 1 in 2 obvezen prazen niz; pri 3 GUID kartice zaposlenega
valuta	Oznaka iz GET valute, na primer EUR ali USD
vrsta_priloge	prevoz, spanje ali drugo
tečajnica	Opcijski pozitiven tečaj
datoteka	Opcijski objekt datoteke

Za kartico se preveri, da pripada zaposlenemu tega naloge. Opis načina plačila dopolni Phoenix: Gotovina, TRR nakazilo ali naziv kartice. ID-ja zaposlenega ali kartice ne uporabite namesto GUID-a kartice.

Če tečajnica ni podana, API za strošek najprej poišče tečaj na datum obračun, pri več vnosih zadnjega po ID. Če na ta dan ni vnosa, uporabi zadnji dodani tečaj za valuto. Ta rezervni tečaj je lahko tudi novejši od datuma obračuna. Za EUR je tečaj vedno 1; druga izrecna vrednost se zavrne.

Pretvorba je skupaj_znesek = value / tečajnica, zaokroženo na dva decimalna mesta. Primer: 112.34 USD / 1.1234 = 100 EUR. Uporabljeni tečaj se vrne na strošku. Če ni veljavnega tečaja ali valuta ni

dovoljena, se zahtevek zavrne. Pri dnevnicah se za izbiro tečaja uporablja zadnji dodani vnos, kot opisuje prejšnje poglavje.

Slika ali PDF pri strošku

Vsak strošek lahko vsebuje največ eno datoteko v naslednjem polju:

```
"datoteka": {  
  "ime_datoteke": "Cestnina.pdf",  
  "vsebina_base64": "TUKAJ_DEJANSKA_BASE64_VSEBINA"  
}
```

To je izsek objekta stroška. Ne pošiljajte date ali description v objektu datoteka; opis in datum izhajata iz stroška in obračuna. Datoteko je mogoče poslati ob ustvarjanju ali naknadno z urejanjem naloga. Celotna vsebina je Base64 brez predpone data:. Podprti tipi in omejitve so enaki kot pri prevozi oseb.

Datoteka dobi novo strežniško ime. Polje datoteka v shranjenem strošku vsebuje to ime; ne vsebuje Base64 vsebine in ni samodejno javni URL. GET naloga ne prenaša bajtov datotek.

Rezultat in pregled naloga

```
GET pn_nalog.php?leto=2026&pn_guid=GUID_NALOGA
```

GET zahteva samo leto in pn_guid. Ne uporablja iskanja, strani ali samo_stevilo=1. Če nalog ni najden v podjetju in letu ključa, vrne 404 NALOG_NI_NAJDEN. Bere tudi zaključene in poknjžene naloge. Splošen seznam oziroma iskalnik potnih nalogov v tej različici ni izveden; vrnjeni GUID po ustvarjanju shranite.

Ustvarjanje in GET vrneta glava, kraji, dnevnice in priloge ter običajno ovojnico. Ustvarjanje doda še ponovitev in vrnitev_dodana. To so dejanski shranjeni podatki z ID-ji in zneski. Datotek v Base64 ni v odzivu. Naslednji izsek prikazuje pomembna polja glave, ne celotnega odgovora:

```
{  
  "glava": {  
    "id": 28,  
    "guid": "66666666-7777-4888-8999-AAAAAAAAAAAA",  
    "num": 18,  
    "st_naloga": "18/2026",  
    "delavec": 5,  
    "datum": "2026-10-22 08:30:00",  
    "datum_konca": "2026-10-22 17:00:00",  
    "st_km": 260,  
    "st_km_po": 0.43,  
    "obracunan": 1,  
    "zaključen": 0,  
    "poknjizen": 0  
  }  
}
```

Odziv poleg vhodnih polj vsebuje naslednje pomembne skupine:

Del odgovora	Dodatna polja in pomen
glava	id, guid, num, st_naloga; delavec_ime, delavec_naslov, delavec_del_mesto; nalog, naloga, sredstva_ime, vozilo, reg; odhod, prihod, st_km_po; statusi in izračunane vsote
kraji	id, pn kot ID naloga, pn_guid, user_guid ter start, end, km, dest
dnevnice	id, guid, pn_guid, user_guid, drzava, vrsta_dnevnice, zajtrk, valuta, valuta_znesek, znesek, tecajnica, kolicina, skupaj_znesek
priloge	id, pn, pn_guid, user_guid, priloga, value, valuta, valuta_value, tecajnica, skupaj_znesek, nacin_placila, nacin_placila_guid, nacin_placila_opis, vrsta_priloge, datoteka

Poslovna številčna polja naloga in podrobnosti so JSON števila. Datum odhoda in vrnitve v odzivu vsebuje sekunde. user_guid pri novih API vnosih ostane prazen. Glava lahko vrne tudi obstoječa polja za druge načine obračuna; ta niso dovoljena vhodna polja tega API-ja. Odjemalec naj pri branju dopušča dodatna polja in za urejanje sestavi samo dovoljeni vhod.

Izračun skupnih zneskov

Polje glave	Izračun
st_km	Vsota km vseh odsekov, tudi samodejne vrnitve
znesek_prevoz	$st_km \times st_km_po$ in stroški vrste prevoz
znesek_prehrana	Vsota dnevnic
znesek_spanje	Vsota stroškov vrste spanje
znesek_drugo	Vsota stroškov vrste drugo
znesek_skupaj	Vsota štirih zneskovnih kategorij
znesek_izplacilo	Skupaj minus stroški TRR in kartic ter minus predujem

Posamezni izračuni v EUR se zaokrožijo na dva decimalna mesta, vsote se računajo v centih. Izplačilo je lahko negativno, kadar odbitki presegajo stroške. Končnih zneskov ne pošiljajte v glava, saj jih izračuna Phoenix.

Urejanje naloga in naknadne priloge

POST pn_uredi.php?leto=2026 vrne HTTP 200. Vsaj eno izmed glava, kraji ali priloge mora biti prisotno.

```
{
  "pn_guid": "66666666-7777-4888-8999-AAAAAAAAAAAA",
  "glava": {
    "datum_konca": "2026-10-22 18:00",
    "dodatki": "Dopolnjen opis poti"
  }
}
```

glava je delna sprememba dovoljenih vhodnih polj iz ustvarjanja. Izpuščena polja ostanejo. ID, GUID, num in st_naloga se ohranijo, prav tako shranjena kilometrina ter besedili nalog in naloga. Pri istem zaposlenem ostanejo njegova zgodovinska imena, naslov, delovno mesto, vozilo in registracija; pri spremembi zaposlenega se dopolnijo iz novega.

Urejanje je prepovedano, če je zaključen > 0 ali poknjizen > 0. To velja tudi za spremembo poti, stroškov in datotek. Stanja ni mogoče ponastaviti skozi API.

Poslani kraji zamenjajo vse obstoječe odseke. Poslani priloge zamenjajo vse obstoječe stroške. Izpuščen seznam se ohrani. priloge=[] odstrani vse stroške, kraji pa morajo vsebovati vsaj en odsek. Za naknadno dodajanje stroška najprej preberite nalog in sestavite celoten želeni seznam, vključno z obstoječimi stroški, ki jih želite ohraniti.

Za ohranitev datoteke pri zamenjavi stroškov pošljite obstojeca_priloga_id iz svežega GET:

```
{
  "pn_guid": "66666666-7777-4888-8999-AAAAAAAAAAAA",
  "priloge": [
    {
      "priloga": "Popravljen parkirnina",
      "value": 6,
      "nacin_placila": 1,
      "nacin_placila_guid": "",
      "valuta": "EUR",
      "vrsta_priloge": "prevoz",
      "obstojeca_priloga_id": 75
    },
    {
      "priloga": "Nova cestnina",
      "value": 12.5,
      "nacin_placila": 1,
      "nacin_placila_guid": "",
      "valuta": "EUR",
      "vrsta_priloge": "prevoz",
      "datoteka": {
        "ime_datoteke": "Cestnina.pdf",
        "vsebina_base64": "TUKAJ_DEJANSKA_BASE64_VSEBINA"
      }
    }
  ]
}
```

Izsek predpostavlja, da želite ohraniti le ta dva stroška; druge bi tak zahtevke odstranil. Obstoječa ID 75 mora pripadati temu nalogu. Nova datoteka in obstojeca_priloga_id na istem strošku se izključujeta. Če pri poslanem strošku izpustite oboje, novi strošek nima datoteke.

Pri ohranjanju stroškov po potrebi pošljite tudi njihov vrnjeni tecajnica, da ob ponovnem izračunu ne izberete novega. Če celoten seznam priloge izpustite, se ohranijo obstoječi stroški, datoteke in shranjeni tečaji. Dnevnice se pri urejanju ponovno izračunajo po pravilih cenika.

Urejanje ponovno shrani podrobnosti in jim lahko dodeli nove ID-je. Za naslednjo spremembo uporabite ID-je iz zadnjega odgovora ali GET. Zastarelih ID-jev stroškov ne uporabljajte. Uspeh vsebuje novi celotni nalog, vrnitev_dodana, operacija="uredi", ponovitev in ciscenje_datotek. Datoteke, ki niso več povezane in niso deljene z drugim nalogom, se odstranijo.

Brisanje naloga

POST pn_izbrisi.php?leto=2026:

```
{"pn_guid": "66666666-7777-4888-8999-AAAAAAAAAAAA"}
```

Dovoljeno je izbrisati samo zadnji nalog po num v izbranem podjetju in letu, če poknjizen ni večji od 0. Zaključitev sama ne prepreči brisanja zadnjega nepoknjiženega naloga. Obstoj višje številke katerega koli drugega naloga prepreči izbris starejšega.

Izbris odstrani glavo, poti, dnevnice, stroške in lastne nedeljene datoteke. Naslednji nalog lahko ponovno uporabi sproščeno številko, vendar dobi nov GUID. Ponovitev starega zahtevka za izbris ne izbriše novega naloga z isto številko.

Odgovor HTTP 200 vsebuje glava z identiteto izbrisanega naloga, operacija="izbrisi", ponovitev, izbrisano_krajev, izbrisano_dnevnice, izbrisano_prilog in ciscenje_datotek. ciscenje_datotek="potreben_pregled" zahteva pregled pri upravljavcu. API ne ponuja samostojnega brisanja odseka poti ali stroška; želeni seznam spremenite skozi pn_uredi.php.

7 Omejitve in obravnava napak

Omejitve zahtevkov

Omejitev	Vrednost
Velikost strani seznamov	Največ 100 zapisov
Iskanje	Največ 150 znakov
Idempotency-Key	1–128 dovoljenih znakov ASCII
JSON brez novih datotek	Največ 1 MiB oziroma 1048576 bajtov
JSON z novimi datotekami	Največ 22 MiB oziroma 23068672 bajtov
Posamezna datoteka	Privzeto 3,5 MB; nastavljivo pri administratorju
Število novih datotek na zahtevek	Največ 5
Skupaj izvirne datoteke na zahtevek	Največ 15 MiB oziroma 15728640 bajtov
Vrste datotek	PDF, JPG, JPEG, PNG, WEBP
Vrstice prevoza oseb	1–500 na dodajanje oziroma ustvarjanje
Odseki potnega naloga	1–500 v poslanem seznamu
Stroški potnega naloga	Največ 100 v seznamu

Omejitev posamezne datoteke določa nastavitve v Phoenix-u za vsako posamezno podjetje in poslovnega leto. Nastavitve spremeni administrator. Če manjka ali ni veljavna, API uporabi privzeto 3,5 MB. V tej nastavitvi 1 MB pomeni 1048576 bajtov; 3,5 MB je 3670016 bajtov izvirne datoteke pred kodiranjem Base64. Datoteka točno na meji je dovoljena.

Nastavitve velja za vsako novo datoteko posebej, skupna omejitev 15 MiB izvirnih datotek in 22 MiB JSON-a pa velja tudi po spremembi nastavitve. Zvišanje meje posamezne datoteke ne poveča skupne omejitve zahtevka. Obstojećih prilog pri urejanju brez novih datotek ne preverjamo ponovno glede na znižano mejo; tudi uspešne ponovitve že potrjenih zahtevkov se ne zavrnejo zaradi nove nižje meje.

Zvišanje omejitve posamezne datoteke pri administratorju ne poveča skupne omejitve 15 MiB izvirnih datotek na zahtevek. Omejitev petih datotek se nanaša na nove datoteke posameznega zahtevka, ne na skupno število prilog celotnega dokumenta. Obnova zelo velikega obstoječega dokumenta lahko zahteva tudi zavrnitev s 413. Zunanji spletni strežnik lahko uveljavlja dodatno omejitev telesa.

Statusi HTTP

HTTP	Pomen in priporočeno ravnanje
200	Uspešno branje, dodajanje, urejanje ali brisanje
201	Uspešno ustvarjanje dokumenta; tudi ponovitev potrjenega ustvarjanja
400	Neveljaven parameter, leto, JSON ali Idempotency-Key; popravite zahtevek

HTTP	Pomen in priporočeno ravnanje
401	Ključ manjka, je neveljaven, pretečen ali preklican; uredite poverilnico
403	Ključ nima dovoljenja za branje ali pisanje
404	Podjetje, leto, dokument ali povezana postavka ni najdena
405	Napačna metoda; preverite odzivno glavo Allow
409	Poslovna omejitev, konflikt ali zahtevek, ki potrebuje pregled
413	Preveliko telo, datoteke ali obnovitveni podatki
415	Napačen Content-Type; uporabite application/json
422	Napačna polja ali poslovni podatki; preverite telo zahteve
426	API zahteva HTTPS
503	Storitev, baza, preverjanje dostopa ali potrebna konfiguracija trenutno ni na voljo

Pomembne oznake napak

Oznaka	Pomen
NEVELJAVEN_KLJUC	Ključ ni ustrezen ali ni veljaven
BRANJE_NI_DOVOLJENO	Manjka dovoljenje Branje
PISANJE_NI_DOVOLJENO	Manjka dovoljenje Pisanje
NAPACNO_LETO	Manjkajoče ali napačno leto
LETO_NI_NAJDENO	Za podjetje ni baze zahtevanega leta
NEZNAN_PARAMETER	Parameter URL ni podprt
NAPACEN_DATUM_SPREMEMBE	Napačen filter spremenjeno_od
MANJKA_IDEMPOTENCY_KEY	Oznaka ponovitve manjka ali ima napačno obliko
NAPACEN_JSON	Telo ni veljaven objekt JSON
NAPACNI_PODATKI_PREVOZA	Napačen podatek ali nedovoljeno polje prevoza
NAPACNI_PODATKI_NALOGA	Napačen podatek potnega naloga
PREVOZ_JE_POKNJIZEN	Sprememba ali izbris knjiženega prevoza ni dovoljen
PREVOZ_NI_ZADNJI	Obstaja prevoz z višjo številko
VEC_PREVOZOV	Izbor za dodajanje ni enoličen
VRSTICA_NI_NAJDENA	ID vrstice ne pripada izbranemu prevozu

Oznaka	Pomen
NALOG_JE_ZAKLJUCEN	Zaključenega naloga ni dovoljeno urejati
NALOG_JE_POKNJIZEN	Poknjizenega naloga ni dovoljeno urejati ali brisati
NALOG_NI_ZADNJI	Obstaja potni nalog z višjo številko
NALOG_NI_NAJDEN	Nalog ni najden v izbranem podjetju in letu
PONOVLJENA_OZNAKA_Z_DRUGO_VSEBINO	Ista oznaka je bila uporabljena za drugačno vsebino ali operacijo
ZAHTEVAK_ZA_PREVERJANJE	Prejšnji poskus zahteva pregled; ne uporabite slepo nove oznake
NEPOPOLN_ZAHTEVAK	Nedokončan zahtevek zahteva pregled
PANDORA_NI_DOSEGLJIVA	Preverjanje dostopa ni uspelo
PODATKI_NISO_NA_VOLJO	Splošna interna napaka ali nedosegljiv vir

Seznam oznak pokriva glavne primere, ni zaprt nabor vseh možnih internih preverjanj. Vedno obravnavajte tudi neznano oznako glede na HTTP status. Ob sporočilu o manjkajočem tečaju, napačni povezavi podjetja ali nejasnem zaposlenem se obrnite na upravljavca.

Podatki za podporo

Pri prijavi težave posredujte čas klica, okolje, naslov operacije, leto, HTTP status, X-Request-ID oziroma zahteva_id, napaka.oznaka, oznako Idempotency-Key in GUID dokumenta, če ga že imate. API ključa, slik računov in celotnih števil kartic ne pošiljajte v običajnih dnevnikih. Vsebinske zahteve posredujte po dogovorjenem varnem postopku.

Uporaba API-ja je brezplačna. Brezplačna uporaba ne pomeni neomejenega obsega zahtevkov ali zagotovljene ravni storitve (SLA). Pogoje dostopa, morebitne kvote in način podpore pred redno uporabo uskladite z upravljavcem Phoenixa.

8 Primeri odjemalcev

Primeri uporabljajo enak HTTP API ne glede na programski jezik. Za Android je primeren katerikoli odjemalec HTTPS, ki podpira zahtevane glave in JSON. Naslednji primeri ne izklopljajo preverjanja certifikata.

POST s curl

V datoteko prevoz.json shranite telo iz poglavja 5. Pred pošiljanjem določite in trajno shranite oznako operacije.

```
export PHOENIX_IDEMPOTENCY_KEY='mobilna-voznja-482-ustvari'

curl --request POST \
  "${PHOENIX_API_URL}po_ustvari.php?leto=2026" \
  --header "Authorization: Bearer ${PHOENIX_API_KEY}" \
  --header 'Accept: application/json' \
  --header 'Content-Type: application/json' \
  --header "Idempotency-Key: ${PHOENIX_IDEMPOTENCY_KEY}" \
  --data-binary @prevoz.json
```

Ob ponovitvi ohranite isti naslov, datoteko in oznako. Za nov poslovni dogodek uporabite novo oznako.

Python brez dodatnih paketov

Primer pošlje shranjeni prevoz.json. Spremenljivke okolja vsebujejo osnovni naslov, ključ in že shranjeno oznako operacije.

```
import json
import os
import urllib.error
import urllib.request
from pathlib import Path

url = os.environ['PHOENIX_API_URL'] + 'po_ustvari.php?leto=2026'
headers = {
    'Authorization': 'Bearer ' + os.environ['PHOENIX_API_KEY'],
    'Accept': 'application/json',
    'Content-Type': 'application/json',
    'Idempotency-Key': os.environ['PHOENIX_IDEMPOTENCY_KEY'],
}
request = urllib.request.Request(
    url, data=Path('prevoz.json').read_bytes(),
    headers=headers, method='POST'
)
try:
    with urllib.request.urlopen(request, timeout=30) as response:
        status = response.status
        body = response.read()
except urllib.error.HTTPError as error:
    status = error.code
    body = error.read()

try:
    result = json.loads(body)
except (ValueError, UnicodeDecodeError):
    raise RuntimeError(f'HTTP {status}: odziv ni JSON')

if status not in (200, 201):
    raise RuntimeError(f'HTTP {status}: {result.get("napaka")}')

print('GUID prevoza:', result['prevoz']['guid'])
```

Omrežno prekinitev in iztek časa obravnavajte v čakalni vrsti ter ponovno pošljite isto shranjeno zahtevo. Dodatno si shranite status, oznako dogodka in vrnjeno identiteto, ne samo izpisa na zaslon.

PHP s curl

```
<?php
$url = getenv('PHOENIX_API_URL') . 'po_ustvari.php?leto=2026';
$curl = curl_init($url);
curl_setopt_array($curl, array(
    CURLOPT_POST => true,
    CURLOPT_POSTFIELDS => file_get_contents('prevoz.json'),
    CURLOPT_RETURNTRANSFER => true,
    CURLOPT_CONNECTTIMEOUT => 10,
    CURLOPT_TIMEOUT => 30,
    CURLOPT_HTTPHEADER => array(
        'Authorization: Bearer ' . getenv('PHOENIX_API_KEY'),
        'Accept: application/json',
        'Content-Type: application/json',
        'Idempotency-Key: ' . getenv('PHOENIX_IDEMPOTENCY_KEY')
    )
));
$body = curl_exec($curl);
$status = curl_getinfo($curl, CURLINFO_RESPONSE_CODE);
$error = curl_error($curl);
curl_close($curl);
if($body === false){
    throw new RuntimeException($error);
}
$result = json_decode($body, true, 512, JSON_THROW_ON_ERROR);
if(!in_array($status, array(200, 201), true)){
    throw new RuntimeException('HTTP ' . $status);
}
echo $result['prevoz']['guid'];
```

Priprava Base64 datoteke

```
import base64
from pathlib import Path

datoteka = {
    'ime_datoteke': 'racun.pdf',
    'vsebina_base64': base64.b64encode(
        Path('racun.pdf').read_bytes()
    ).decode('ascii'),
}
```

Za strošek potnega naloga uporabite ta objekt kot datoteka. Pri prevozu oseb mu dodajte date in description ter ga vstavite v vrhnji seznam priloge. Pri ponavljanju ne prekodirajte slike v drugo vsebino in ne generirajte novega imena datoteke.

9 Pregled vseh zahtevkov

Vsi naslovi spodaj so glede na BASE_URL. Pri vsakem je leto obvezen v URL. Vsi GET zahtevajo Branje, vsi POST Pisanje in Idempotency-Key.

Metoda	Naslov	Namen
GET	partnerji.php	Partnerji in spremembe po timestamp_lastedit
GET	zaposleni.php	Osnovni podatki zaposlenih
GET	drzave.php	Države z vrednostmi dnevnic
GET	pn_dnevnice.php	Ločen cenik dnevnic
GET	pn_prevoz.php	Vrste prevoza
GET	pn_vrste_prilog.php	Vrste stroškovnih prilog
GET	pn_nacini_placila.php	Gotovina, TRR in kartice zaposlenega
GET	valute.php	Obstoječe valute
GET	po_prevozi.php	Glave prevozov oseb
GET	po_vrstice.php	Vrstice izbranega prevoza
GET	po_priloge.php	Metapodatki prilog prevoza
GET	pn_nalog.php	Celoten potni nalog po GUID-u
POST	po_ustvari.php	Nov prevoz oseb
POST	po_dodaj_vrstice.php	Nove vrstice na obstoječi prevoz
POST	po_dodaj_priloge.php	Nove priloge na obstoječi prevoz
POST	po_uredi.php	Urejanje glave prevoza
POST	po_uredi_vrstico.php	Urejanje postavke prevoza
POST	po_izbrisi_vrstico.php	Brisanje postavke prevoza
POST	po_izbrisi.php	Brisanje zadnjega nepoknjiženega prevoza

Metoda	Naslov	Namen
POST	pn_ustvari.php	Nov potni nalog
POST	pn_uredi.php	Urejanje naloga ter zamenjava poti ali stroškov
POST	pn_izbrisi.php	Brisanje zadnjega nepoknjiženega naloga

Kaj ostaja v odjemalcu

Zaznavanje vožnje z GPS ali Bluetooth, glasovni vnos, izbira zasebne vožnje, priprava fotografij, potrjevanje vožnje, lokalna hramba in pošiljanje ob povratku povezave so naloge zunanje aplikacije. Phoenix sprejme že pripravljene poslovne podatke.

Zaključevanje in knjiženje dokumentov se izvedeta v Phoenixu. API v tej različici nima operacij za spremembo teh statusov, samostojnega iskanja potnih nalogov po mesecu ali zaposlenem, prenosa vsebine datotek, branja izbranih partnerjev ali iskanja po lastnem zunanjem ID-ju. Izbrani partner se na pot zapiše kot besedilo cilja.

Preverjanje pred redno uporabo

Preizkusite prvo branje, strani in prazne sezname, shranjevanje v pravilno podjetje in leto, vnos s fotografijo, iste podatke v Phoenixovem prikazu ter ponovitev istega zahtevka po izgubljenem odgovoru. Preverite tudi lastno obravnavo napačnega ključa, pomanjkanja dovoljenj, zaključenega in poknjiženega dokumenta, neveljavne kartice ter popravkov poti in stroškov.

Dogovorite se, ali boste za vsako vožnjo ustvarili ločen prevoz ali dodajali vrstice mesečnemu prevozu. Pri potnih nalogih uporabljajte trenutno potrjeno vsebino, ohranite zadnji GUID in ID-je ter ob ponovnem urejanju izhajajte iz svežega GET. Pred predajo dejanskim uporabnikom uskladite testni in produkcijski naslov, ključe ter pogoje uporabe.