

OBJEDNÁVKA

číslo : 440

ODBĚRATEL

IČO: 60437359 DIČ: CZ60437359

Regionální organizátor Pražské
integrované dopravy

Rytířská 406/10

110 00 Praha 1

Bankovní spojení PPF Banka a.s.

Číslo účtu 2000930004/6000

DODAVATEL

IČO: 02795281 DIČ: CZ02795281

Operátor ICT, a.s.

Dělnická 213/12

170 00 Praha

Bankovní spojení

Číslo účtu 5920172/0800

Objednáváme: BE IPT č. 1 Zpracování uživatelské zpětné vazby

Objednáváme u Vás,

na základě Smlouvy o vývoji, implementaci a provozu intermodálního
plánovače trasy ze dne 26. 7. 2019, ve znění pozdějších dodatků
objednáváme práce na BE IPT (změnový požadavek č. 1) Zpracování
uživatelské zpětné vazby dle přiložené nabídky ze dne 20. 7. 2023

Termín do 15 týdnů od nabytí účinnosti objednávky.

Předpokládaná cena: 1 060 800 Kč + DPH

Objednává: Vysloužil

V Praze

Dne:

Vyřizuje:

Telefon:

E-mail:

Regionální organizátor pražské integrované dopravy

UPOZORNĚNÍ DODAVATELI: I NA FAKTUŘE UVEĎTE ČÍSLO TÉTO OBJEDNÁVKY !

Číslo nabídky: 202307119OSC

Nabídka: „Podpora generování reportů z BO na samostatném DB serveru “

K rukám:

oict.

Regionální organizátor Pražské integrované dopravy
Rytířská 406/10, 110 00, Praha 1

Nabízející: Operátor ICT, a.s.

Operátor ICT, a.s. (dále jen „OICT“), je městskou společností, která pro Hlavní město Prahu zajišťuje odborné služby ICT a realizaci ICT projektů pro městské části, další městské společnosti, případně další subjekty.

Předmět nabídky

Předmětem nabídky, připravené na základě poptávky objednatele, je plnění dle čl. 2 odst. 2.3 Smlouvy o vývoji, implementaci a provozu intermodálního plánovače trasy ze dne 26. 7. 2019, ve znění Dodatku č. 1 ze dne 12. 5. 2020, Dodatku č. 2 ze dne 8. 6. 2021 a Dodatku č. 3 ze dne 14. 7. 2023 (dále jen „smlouva“). Konkrétně jde o Rozvojový požadavek BE IPT č1_Zpracování uživatelské zpětné vazby.

ZNĚNÍ ROZVOJOVÉHO POŽADAVKU:

1. Zvýšení frekvence stahování všech real time dat z minut na sekundy

- Zrušit dnešní enum typu "EVERY_MINUTE" a nahradit ho Integer, který bude znamenat interval v sekundách
- Dopsat interval v sekundách i do "i" ček v Admin app u všech proměnných v sekci Dynamic data
- Nastavit defaultně tyto hodnoty:
 - ptVehiclePositionsDownloadFrequency 30 (změna)
 - ptTripUpdatesDownloadFrequency 30 (změna)
 - carAnomaliesDownloadFrequency 300 (jako dnes)
 - carFcdDownloadFrequency 300 (jako dnes)
 - Ostatní atributy 60 (jako dnes)

V rámci zpracování se nestanoví minimální hodnoty updatu dat (což je nutné vzhledem k času zpracování samotných dat). Předpoklad minimální hodnoty updatu dat je

- gbfs i gtfs-realtime data: 15 vteřin
- car fcd a ndic data: 120 vteřin
- data funkčnosti výtahů: 60 vteřin
- Parking Measurements data: N/A

v rámci zpracování se hodnota ověří a nastaví empiricky

Současně změnit systém zpracování dat, tak aby aktualizace dat z externích zdrojů vyvolala po aktualizaci i aktualizaci dat v rámci díla. Požadované chování je, aby se při změně hodnoty frekvence stahování dat v Admin app, upravila i frekvence aktualizace dat v rámci komponent díla (navázala na ní).

Předpokladem tedy je, že parametry aktualizaci dat v rámci díla:

- IRE: RT_DATA_UPDATE_INTERVAL_SECONDS
- SO: dynamicRoadDataUpdate.everyMilliseconds
- SO: ptStairsFreeStatusCheck.everyMilliseconds

budou nahrazeny, častou rychlou kontrolou změny dat, tak aby se dílo vnitřně doptávalo na změnu dat, a ta by byla událostí, která spouští zasílání aktualizovaných dat do dalších částí díla.

Poznámka: vychází z issue: <https://gitlab.com/operator-ict/ipt/umotional/-/issues/526>

2. Přidání informace pro walk a bike/walk with bike a bike segmenty ohledně povrchovosti a překážek (eskalátor, výtah, schody, rampa, travelátor) z pathway i OSM a přidání informace o exitu z metra

Datová struktura response

pathwaySegments

- nově se budou v rámci stávajícího elementu `pathwaySegments` zaznamenávat všechny prvky, které jsou v podzemí, zakryté či nejsou pod širým nebem
- objekt `pathwaySegments` bude mít následující strukturu: `[{ startIndex:int endIndex:int }]`
- bude platit pro walk, walk with bike, bike a bikesharingové segmenty

pathwayStructures

- dále bude vytvořen nový element `pathwayStructures` zahrnující informace o všech "překážkách" - schodech, elevátorech,... - a exitech po cestě
- objekt `pathwayStructures` bude mít následující strukturu: `[{ startIndex:int endIndex:int type: one of enum("stairs", "travelator", "escalator", "elevator", "exit") description: string (nepovinné) vertical: one of enum("up","down") endLevel: integer (nepovinné) }]`
- description je obsahuje volitelný název prvku (nejčastěji exitu)
- liniové objekty (schody, eskalátory, travelátory) budou zaznamenány liniově (`startIndex != endIndex`), bodové (výtah, exit) budou zaznamenány bodově (`startIndex == endIndex`)
- bude platit pro walk, walk with bike, bike a bikesharingové segmenty
- v testing app se zobrazí odlišením dané liniové části segmentu (např. odlišný styl čáry) a bodem (ikonkou) na začátku segmentu.

Chování po cestě skrz GTFS pathways

pathwaySegments

- všechny GTFS pathways se zapíší do `pathwaySegments` stejně jako doposud

pathwayStructures

- do `pathwayStructures` se zapíše všechny GTFS pathway jiné než `walkway`, a to liniově nebo bodově v závislosti na charakteru
 - bude v mobilní aplikaci v budoucnu reprezentována pravděpodobně ikonkou na začátku segmentu, případně (?) jinou linií v mapě
1. Schody
 - celá pathway, kde `pathway_mode = 2`; liniově (`startIndex < endIndex`),
 - { `startIndex:1 endIndex:3 type: "stairs" vertical: "up" endLevel: 0` }
 2. Travelátor
 - celá pathway, kde `pathway_mode = 3`; liniově (`startIndex < endIndex`),
 - { `startIndex:1 endIndex:3 type: "travelator"` }
 3. Eskalátor
 - celá pathway, kde `pathway_mode = 4`; liniově (`startIndex < endIndex`),
 - { `startIndex:1 endIndex:3 type: "escalator" vertical: "up" endLevel: 1` }
 4. Výtah
 - pathway, kde `pathway_mode = 5`; liniově (`startIndex < endIndex`) NEBO bodově (`startIndex = endIndex`), existují šikmé výtahy
 - { `startIndex:7 endIndex:7 type: "elevator" vertical: "down" endLevel: -1` }
 5. Exit
 - konec poslední pathwaye, tj. stop definovaný jako exit (`location_type = 2`); bodově (`startIndex = endIndex`)
 - { `startIndex:9 endIndex:9 type: "exit" description: "E4"` }

Atribut `vertical`:

- realizován bude pouze u schodů, travelátorů, eskalátorů a výtahů
- se získává z `level_indexu` `levelu` z `from_stop_id` a `to_stop_id` definující pathway (propojeno skrz `stops.txt-> levels.txt`)
- pokud `level_index` z `to_stop_id` > `level_index` z `from_stop_id`, pak `vertical=up`
- pokud `level_index` z `to_stop_id` < `level_index` z `from_stop_id`, pak `vertical=down`
- jinak atribut `vertical` není přítomen
- zobrazíme v aplikaci šipečkou

Atribut `endLevel`:

- získává z `level_indexu` `levelu` `to_stop_id` definující pathway (propojeno skrz `stops.txt-> levels.txt`)
- pokud atribut není v datech, tak není přítomen
- zobrazíme v aplikaci (např. u výtahů - do kterého patra jet)

Atribut `description`:

- realizován bude u exitů
- u exitů označuje název exitu (=stop_name)

Cesta z metra, kde jdu:

- 0-1 po nástupišti
- 1-2 po eskalátoru do vestibulu (level -1)
- 2-4 chodbou vestibulem
- 4-5 po schodech na povrch do exitu E5

- bude popsána v `pathwaySegments` takto: `[{ startIndex:0 endIndex:5}]`
- bude popsána v `pathwayStructures` takto: `[{ startIndex:1 endIndex:2 type: "escalator" vertical: "up" endLevel: -1 }, { startIndex:4 endIndex:5 type: "stairs" vertical: "up" endLevel: 0 }, { startIndex:5 endIndex:5 type: "exit" description: "E5" }]`

Chování po cestě skrz pěší síť OSM

- obdobné chování budeme implementovat i po cestě přes běžnou pěší síť OSM, čili mimo GTFS pathways
- do `pathwaySegments` se zapíše:
 - liniově - úseky, které jsou zakryté
- do `pathwayStructures` se zapíše:
 - liniově/bodově - schody, travelátory, eskalátory, výtahy

pathwaySegments

- všechny úseky, které nejsou na povrchu
- OSM query: `covered = yes || layer < 0 || tunnel = *` (kromě no)
- `[{ startIndex:30 endIndex:39}]`

pathwayStructures

1. Schody
 - celá `highway=steps`
 - `{ startIndex:1 endIndex:2 type: "stairs" }`
2. Travelátor
 - `(highway=footway && conveying=yes/forward/backward/reversible)`
 - `{ startIndex:3 endIndex:4 type: "travelator" }`
3. Eskalátor
 - začátek `highway=steps && conveying=yes/forward/backward/reversible`
 - `{ startIndex:4 endIndex:5 type: "escalator" }`
4. Výtah
 - `highway=elevator`
 - `{ startIndex:7 endIndex:7 type: "elevator" }`

Atribut vertical:

- budeme implementovat pouze u schodů, travelátorů, eskalátorů
- získává se z atributu `incline` v závislosti na směru cesty po překážce (schody, eskalátor, travelátor)
- pokud jdu po směru digitalizace a `incline = up`, pak `vertical = up`, pokud jdu proti směru digitalizace a `incline=up`, pak `vertical = down`
- pro výtahy zatím vyplněnost OSM nedává moc možností, abychom je mohli používat, tedy datová struktura bude o to méně penetrována daty.

Atribut endLevel:

- vyplněnost OSM zatím nedává moc možností pro to, abychom je mohli používat, tedy datová struktura bude o to méně penetrována daty.

Pěší cesta, která vede:

- 0-5 po povrchu
- 5-6 po povrchových schodech dolů
- 6-8 po povrchu

- 8-9 skrz podchod rovně
- 9-10 eskalátorem na povrch
- 10-12 po povrchu
- bude popsána v pathwaySegments takto: `[{ startIndex:8 endIndex:9}]`
- bude popsána v pathwayStructures takto: `[{ startIndex:5 endIndex:6 type: "stairs" vertical: "down" }, { startIndex:9 endIndex:10 type: "escalator" vertical: "up"}]`

Zdroje dat: stávající OSM a GTFS

Poznámka: vychází z issue: <https://gitlab.com/operator-ict/ipt/umotional/-/issues/545>

3. Do Testing app přidat offline našeptávač pro zastávky a obce

Zastávky:

- Nově napojit na soubor GTFS stops (automatické hromadné zadávání id zastávek) <https://data.pid.cz/stops/json/stops.json>
- Současně se přidá parametr do Admin app - static data a změna IDM.

Obce:

- Napojení na soubor GTFS-municipalities.json
- Pro vývoj dočasný zdroj: https://deviptoict.blob.core.windows.net/storage/FE-whisperer/GTFS-municipalities/GTFS-municipalities_new.json
- Pro produkci trvalý zdroj: <https://iptoict.blob.core.windows.net/storage/FE-whisperer/GTFS-municipalities/GTFS-municipalities.json>

Požadovaný stav

1. Do offline našeptávače se propisují obce ze souboru GTFS-municipalities.json.
2. Pro volání API IPT se využije obsahu pole v atributu proper-ties.gtfsIds, ve které se nachází seznam zastávek. Pokud se v datech nebude atribut nacházet, použije se atribut geometry.coordinates.
3. Název je závislý na atributu properties.icon, která může nabývat 2 hodnot:
 - o municipality – zobrazí se název "Obec" (např. „Praha 1")
 - o center – zobrazí se název "Obec – Centrum" (např. „Praha – Centrum“)
- Prezentační logika zobrazovaných výsledků z našeptávače pro obce, zobrazí se:
 - Kopeč "name", Úžice "municipality", okres Mělník "district"

Poznámka: vychází z issue: <https://gitlab.com/operator-ict/ipt/umotional/-/issues/383>

4. Sjednocení datového zdroje GTFS RT (vehiclePosition a tripUpdate)

V admin app sjednotit stávající dva zdroje dat:

- ptVehiclePositionsExternalUrl = https://api.golemio.cz/v2/vehiclepositions/gtfsrt/vehicle_positions.pb
- ptTripUpdatesExternalUrl = https://api.golemio.cz/v2/vehiclepositions/gtfsrt/trip_updates.pb

do jednoho

- ptVehiclePositionsAndPtTripUpdatesExternalUrl = https://api.golemio.cz/v2/vehiclepositions/gtfsrt/pid_feed.pb

Z důvodu, že hrozí časová nekonzistence, pokud se soubory stahují samostatně.

Poznámka: vychází z issue: <https://gitlab.com/operator-ict/ipt/umotional/-/issues/553>

5. Přidání stavů jízdenky: "Přeprava zdarma" a "Pro Vaši tarifní kategorii zdarma"

Stávající stav:

BE v RoutingResponse posílá v objektu PriceAndPayment.PaymentTiming enum:

```
paymentTiming:
  type: string
  description: Specifies when a payment is made
  enum:
    - PRE_RIDE
    - POST_RIDE
    - COUPON_PURCHASED
    - PAID_IN_FIRST_SEGMENT
    - FREE_RIDE_FOR_ALL
    - FREE_RIDE_FOR_YOU
```

V případě že pro PT

- COUPON_PURCHASED
- FREE_RIDE_FOR_ALL
- FREE_RIDE_FOR_YOU

a) neplatí pro celý segment, ale jen první jeho část pošle se paymentTiming: PRE_RIDE také v objektu PtTicket údaj lastStopIdWithoutTicket: obsahující id zastávky pro kterou jako poslední není potřeba zakoupit jízdenku.

b) neplatí pro celý segment, ale jen poslední část pošle se paymentTiming: PRE_RIDE údaj od které zastávky je již jízda pokryta kupónem nebo CP se neposílá.

Cílový stav

Pozn: pro udržení zpětné kompatibility s funkčností FE je potřeba zachovat výše popsany stav a pouze ho rozšířit. Z toho důvodu není možné rozšířit enum paymentTiming: protože by nové hodnoty nahradily stávající PRE_RIDE, který chodí v segmentech částečně pokrytých kupónem nebo CP. FE by to pravděpodobně neuměl vyhodnotit.

Do ptDetails se přidá nový nepovinný objekt stopPaymentInfo, který bude obsahovat následující sadu nepovinných atributů:

preRide - pro tyto zastávky je v rámci dané trasy potřeba jízdenka uvedená v ptTicket daného segmentu

postRide - aktuálně nemá využití pro PT ale ve specifikaci bychom s ním měli počítat

paidInFirstSegment - pro tyto zastávky v rámci dané trasy platí jízdenka zakoupená v předchozím segmentu

couponPurchased - pro tyto zastávky v rámci dané trasy platí zakoupený kupón

freeRideForYou - pro tyto zastávky není potřeba v rámci dané trasy zakoupit jízdenku na základě daného CP

freeRideForAll - pro tyto zastávky není potřeba v rámci dané trasy zakoupit jízdenku na základě daného CP

`ticketPurchased` - - v rámci dané trasy platí zakoupená jízdenka - v návaznosti na <https://gitlab.com/operator-ict/ipt/umotional/-/issues/517>

Každý výše uvedený atribut bude obsahovat pole 2 hodnot. Zde jako příklad uvádíme "couponPurchased", stejná struktura bude u všech.

```
{
```

```
"couponPurchased":[
  {
    "startStopIndex":1, // index první zastávky z ptDetails.stopTimes pro
    kterou platí "couponPurchased"

    "endStopIndex":5 // index poslední zastávky pro kterou platí
    "couponPurchased"
  },
  {
    "startStopIndex":10,
    "endStopIndex":20
  }
]
}
```

Údaj o tom, že je daná zastávka pokryta kupónem "couponPurchased" nebo pro ni není potřeba jízdenka (freeRide...) má přednost před pokrytím zastávky jízdenkou z ptTicket. Důvodem je možnost zobrazit uživateli informaci, od které zastávky by již mohl jet na základě svého CP nebo předem zakoupeného kupónu

Údaj o tom že daná zastávka je zdarma na základě CP nebo pro všechny má přednost před údajem o tom, že daná zastávka je pokryta kupónem.

Poznámka: vychází z issue: <https://gitlab.com/operator-ict/ipt/umotional/-/issues/533>

6. Chybová hláška, kterou pošle BE, když FE zašle neexistující STOP ID

1. BE akceptuje jako lokaci seznam stopId + gps souřadnice najednou (což je v současnosti považováno za chybu). Pokud by taková kombinace dorazila tak BE by nejdřív hledal podle stopID, a pokud by žádné stopId se seznamu nenašel použil by gps souřadnice.
2. Pokud dojde k vyhledávání pomocí gps souřadnic (t.j ani jedno stopID nebylo nalezeno) tak BE vrací v routingStatus stav *REQUEST_CHANGED a v routingRequestExceptions zapíše na příklad:

```
{
  "routingRequestExceptions":
  [
    {
      "exceptionCode":"STOP_ID_NOT_FOUND",
      "objectName":"stopIds:",
      "listOfValues":[ // seznam nenalezených zastávek
        "U12345",
        "U23698"
      ]
    }
  ]
}
```

jinak se k vyhledávání pomocí gps souřadnic co se týče vyhodnocení chyb staví stejně jako v <https://gitlab.com/operator-ict/ipt/umotional/-/issues/486>. T.j v případě, jakkoliv nesprávného určení lokace vrátí routingStatus LOCATION_NOT_FOUND a v routingRequestExceptions kromě "exceptionCode": "STOP_ID_NOT_FOUND" vrátí taky příslušné info o nenalezené lokaci.

3. Pokud nějaké stopID v seznamu není nalezeno, ale alespoň jedno stopID nalezeno je, BE vyhledává stále podle nalezených stopID vrací ale také *REQUEST_CHANGED a do `routingRequestExceptions` zapíše výjimku stejně jako pro bod 2.

Poznámka: vychází z issue: <https://gitlab.com/operator-ict/ipt/umotional/-/issues/470>

7. Rozdělit volání jednotlivých endpointů GBFS v Admin app

Každý z 10 endpointů bude mít samostatnou proměnnou v Admin app včetně samostatné frekvence volání. Defaultně bude nastavena následující frekvence:

1x 60 vteřin pro:

- `/sharedbikes/gbfs/{system_id}/free_bike_status`
- `/sharedbikes/gbfs/{system_id}/station_status`

1x 3600 vteřin:

- `/sharedbikes/gbfs/systems_list`
- `/sharedbikes/gbfs/{system_id}/gbfs`
- `/sharedbikes/gbfs/{system_id}/gbfs_versions`
- `/sharedbikes/gbfs/{system_id}/system_information`
- `/sharedbikes/gbfs/{system_id}/vehicle_types`
- `/sharedbikes/gbfs/{system_id}/station_information`
- `/sharedbikes/gbfs/{system_id}/system_pricing_plans`
- `/sharedbikes/gbfs/{system_id}/geofencing_zones`

Záznamy z dat s frekvencí 1x 60 vteřin které mají vazbu na neexistující záznam z dat s frekvencí 1 x 3600 vteřin, zahazovat, ale logovat.

Poznámka: vychází z issue: <https://gitlab.com/operator-ict/ipt/umotional/-/issues/539>

8. Přidat data o klimatizaci spojů MHD

- V routing response (včetně `intermodalrouting/ptOnly` endpointu) přidat data o klimatizaci spojů MHD.
- V routing response se předpokládá přidání nového nepovinného atributu do objektu `rideDetails/PtDetails/Trip`
- Parametr Klimatizace nebude filtrem pro vyhledávání, poslouží pouze pro zobrazení ikony na FE.
- Pro zabezpečení zpětné kompatibility s FE se bude jednat extenzi – tedy nový atribut.

Zdroj dat: GTFS RT/ `vehicle->.transit_realtime.ovapiVehicleDescriptor->vehicleType` atribut `"airConditioned":true`

Poznámka: atribut je přítomen jak v trip updates, tak vehicle positions protobuffer souborech.

Poznámka: vychází z issue: <https://gitlab.com/operator-ict/ipt/umotional/-/issues/481>

9. Rozšířit endpoint `vehiclelocations/publictransport`

- Rozšířit vehiclelocations/publictransport Request i Response o Typ dopravního prostředku PT o nepovinný parametr ptVehicle (rozšířit jako pole parametrů): type: string description: Allowed public transport modes enum: - BUS – TRAM – METRO – TRAIN – FERRY – FUNICULAR – TROLLEYBUS
- Rozšířit vehiclelocations/publictransport Request o nepovinné parametry Trip ID a Route ID (rozšířit jako pole parametrů).
- Současný filtr na boundingbox zůstává beze změny (jedná se o extenzi).
- Nové parametry budou fungovat pro filtrování
- Tedy přidat v Testing app i checkbox pro routeld a tripld.
- Testing app se rozšíří v případě endpointu vehiclelocations/publictransport o rozlišení jednotlivých typů dopravních prostředků v public transport (enum PtVehicle - BUS - TRAM - METRO - TRAIN - FERRY - FUNICULAR - TROLLEYBUS), včetně možnosti je na mapě vypnout/zapnout separátně (checkbox).

Poznámka: vychází z issue: <https://gitlab.com/operator-ict/ipt/umotional/-/issues/529>

10. Zabezpečit konzistentní logiku, kdy a co se vrací v priceAndPayment, Upravit logiku, kdy se vrací a co obsahuje objekt priceAndPayment napříč dopravními módy

Aktuální stav:

- Chůze: priceAndPayment:{ priceInCzk:0 priceAccuracy:"EXACT"
- Soukromé kolo: priceAndPayment:{ priceInCzk:0 priceAccuracy:"ESTIMATE" paymentMethods:["UNKNOWN"] paymentTiming:"PRE_RIDE"
- Parkování soukromé kolo stojan: nic

Požadovaný stav:

- Chůze: nic
- Soukromé kolo: nic
- Parkování soukromé kolo stojan: nic (v budoucnu možná změna v souvislosti z implem. B+R)

Poznámka: vychází z issue: <https://gitlab.com/operator-ict/ipt/umotional/-/issues/431>

11. Omezení hledání na zájmový region

1. Hledání trasy bude omezeno na zájmový region, přičemž:

- Do admin app bude přidán zájmový region jako další zdroj statických dat.

Zdroje dat pro určení zájmového regionu jsou následující:

- TEST: https://deviptoict.blob.core.windows.net/storage/Region/zajmova_oblast.geojson
- PROD: https://iptoict.blob.core.windows.net/storage/Region/zajmova_oblast.geojson

2. routingStatus enum ve stávající podobě zůstane zachován, t.j

- OK
- PLAN_NOT_FOUND
- INVALID_REQUEST
- SERVER_ERROR
- LOCATION_NOT_FOUND
- OK_REQUEST_CHANGED
- PLAN_NOT_FOUND_REQUEST_CHANGED

3. do objektu `routingRequestExceptions` se doplní dva chybové stavy:

```
{
"routingRequestExceptions":
[
  {
    "exceptionCode": "WRONG_LOCATION", // pro chybu v souřadnicích nebo v nesprávné
    kombinaci parametrů určení lokace, v routingStatus se nadále vrací INVALID_REQUEST
    "objectName": "origin", // dle toho, pro kterou lokaci došlo k chybě, pokud dojde
    k chybě ve vícero lokacích pošle se samostatná chyba za každou
  },
  {
    "exceptionCode": "LOCATION_OUTSIDE_OF_REGION", // pokud jsou souřadnice správné ale
    mimo zájmový region. V routingStatus se vrací LOCATION_NOT_FOUND
    "objectName": "origin", // dle toho, pro kterou lokaci došlo k chybě, pokud dojde
    k chybě ve vícero lokacích pošle se samostatná chyba za každou
  },
]
}
```

Poznámka: vychází z issue: <https://gitlab.com/operator-ict/ipt/umotional/-/issues/486>

12. Rozdělení SHARED_VEHICLE_TIME_TO_PICKUP a SHARED_VEHICLE_TIME_TO_PARK dle módů

Atributy SHARED_VEHICLE_TIME_TO_PICKUP a SHARED_VEHICLE_TIME_TO_PARK budou v Admin App a end pointu system settings nahrazeny atributy stejné byznys logiky, ale pro jednotlivé dopravní módy sharingu (s předdefinovanými hodnotami PICKUP == 180, PARK == 120):

- SHARED_BIKE_TIME_TO_PICKUP 180
- SHARED_BIKE_TIME_TO_PARK 120
- SHARED_SCOOTER_TIME_TO_PICKUP 180
- SHARED_SCOOTER_TIME_TO_PARK 120
- SHARED_MOPED_TIME_TO_PICKUP 180
- SHARED_MOPED_TIME_TO_PARK 120
- SHARED_CAR_TIME_TO_PICKUP 180
- SHARED_CAR_TIME_TO_PARK 120

V UX Admin app zanikne sekce Shared vehicle a jednotlivé proměnné budou v sekcích daného dopravního módu, kterého se týkají.

Poznámka: vychází z issue: <https://gitlab.com/operator-ict/ipt/umotional/-/issues/459>

13. Testing app – rozlišení jednotlivých operátorů sharingu, přidání end pointu mobilityoperators a rozlišení jednotlivých typů dopravních prostředků v public transport

1. Testing app se rozšíří v případě endpointů stations/bikesharing a vehiclelocations/freefloating:
 - O rozlišení jednotlivých operátorů sharingu v rámci svých dopravních módů, včetně možnosti je na mapě vypnout/zapnout separátně (checkbox).
 - UX se bude přidržovat barvy operátora vycházející z jeho barevné identity (dle kódu barvy v MO.json). Zpřístupněno na

<https://deviptoict.blob.core.windows.net/storage/MobilityOperator/MobilityOperator.json>

proměnná "color".

2. Testing app se rozšíří o end point mobilityoperators

- O zobrazení geofencing zones pro jednotlivé operátory, včetně možnosti je na mapě vypnout/zapnout separátně (checkbox) - Tedy nově připojit end point mobilityoperators.
- Pokud možno v UX polygonů se přidržet barvy operátora vycházející s jeho barevné identity - totožnou s bodem 1 (dle kódu barvy v MO.json).
- Realizovatelnost v návaznosti na na rozvojový požadavek https://gitlab.com/operator-ict/ipt/umotional/-/issues/529#note_1412479571

Poznámka: vychází z issue: <https://gitlab.com/operator-ict/ipt/umotional/-/issues/382>

14. Úprava ošetření výjimek parametrů hledání

1. V současnosti BE IPT vysílá pouze velmi obecnou informaci o zpracování požadavku na vyhledávání v parametru routingStatus:

aktuální stav routing status je:

routingStatus:

type: string

description: Routing response status

enum:

- OK
- PLAN_NOT_FOUND
- INVALID_REQUEST
- SERVER_ERROR
- LOCATION_NOT_FOUND
- OK_REQUEST_CHANGED
- PLAN_NOT_FOUND_REQUEST_CHANGED

Z důvodu zpětné kompatibility nelze seznam zásadně měnit, pouze doplňovat.

2. Cílový stav s ohledem na zachování zpětné kompatibility

Na stejné úrovni jako routingStatus bude přidán nový objekt routingRequestExceptions.

Pokud vyhledávání skončí dle stávajících pravidel stavem *REQUEST_CHANGED budou detaily proč k tomu došlo přepsány do tohoto objektu. Například pokud v parametrech přijde operátor, který není "povolený":

```
{
"routingRequestExceptions":
[
  {
    "exceptionCode": "MAX_VALUE_EXCEEDED",
    "systemSetting": "OPERATORS_ALLOWED_SHARED_BIKE",
    "objectName": "providerIds",
    "listOfValues": [
      "rekola",
      "nextbike_tq"
    ]
  }
]
```

}

- Obdobně se do `routingRequestExceptions` propíšíou i všechny další výjimky validace requestů a parametrů uvnitř requestu nezávisle na tom jaký byl `routingStatus`. Kromě v objednávce explicitně uvedených hodnot `exceptionCode` určí další enum hodnoty reportovaných výjimek dodavatel a tyto patřičně popíše v dokumentaci.
- Všechny parametry tohoto objektu kromě `exceptionCode` budou nepovinné.
- Informace o `routingRequestExceptions` se zobrazí rovněž ve webové testovací aplikaci

Poznámka: vychází z issue: <https://gitlab.com/operator-ict/ipt/umotional/-/issues/559>

15. Maximální doba vyčkávání u garantovaných přestupů

Aktuální stav:

- Informace o maximální době čekání navazujícího spoje `max_waiting_time` (v sekundách) a z jakého na jaký spoj se čeká `from_trip_id` a `to_trip_id` jsou nyní uvedeny v GTFS datech `transfers.txt`
- IPT zohledňuje garantované přestupy a dopočítává zpoždění navazujícího spoje který čeká na zpožděný spoj na který navazuje, nevyužívá ale informaci z `max_waiting_time` ale konstantu 5 minut.

Cílový stav:

1. Vysílání informace o maximální době čekání

V `RoutingResponse` se do objektu `RouteSegment` přidá nepovinný parametr `maxWaitingTime` typu `integer`. Ten bude vyplněn hodnotou `max_waiting_time` z `transfers.txt` v případě, že se jedná o přestupní segment který má `transferType: GUARANTEED_TRANSFER` a tento údaj je k dispozici v datech.

2. Započítání garantovaného přestupu a maximální doby čekání při vyhledávání spojení PT

V IPT bude ponechána stávající logika dopočítávání zpoždění garantovaného spoje, včetně jeho propagace v `routingResponse`, ale místo stávající konstanty 5 minut bude IPT využívat hodnotu z `max_waiting_time` pokud tato bude k dispozici v datech

3. Vypnutí dopočítávání zpoždění garantovaného spoje na straně IPT

V admin aplikaci se do sekce Public transport přidá Bool proměnná `PT_GUARANTEED_TRANSFER_DELAY_PROPAGATION`. Defaultně bude nastavená na `Yes`. V případě nastavení na `No`, se vypne logika dopočítávání zpoždění garantovaného spoje na straně IPT a bude se brát neupravené zpoždění z GTFS real time data feedu

Poznámka: vychází z issue: <https://gitlab.com/operator-ict/ipt/umotional/-/issues/494>

16. Započítávání zakoupených jízdenek

Cílový stav

- MA bude zasílat v requestu jak platné kupóny, tak i platné jízdenky (aktivované a ty které mají určený čas aktivace).
 - Routing request z MA do BE se nebude upravovat.
 - Jedno a vícedenní jízdenky (do 168 hodin včetně) se nebudou vyhodnocovat jako COUPON_PURCHASED, ale jako nová hodnota TICKET_PURCHASED, tedy v routing response tedy bude nově vracet i TICKET_PURCHASED v paymentTiming enumu.
 - FE bude posílat všechny jízdenky, ale BE bude jízdenky kratší než 1 den prozatím ignorovat. (Toto řešení bude tedy lépe připravené na to, až se v budoucnu rozhodneme započítávat i krátkodobější jízdenky).
 - Tedy zakoupené doklady bude vyhodnocovat následovně:
 - pokud je platnost trvání delší, než 168h je to kupon (COUPON_PURCHASED)
 - pokud 24-168h je to jízdenka (TICKET_PURCHASED - nová hodnota)
 - pokud méně než 24h, nezahrnujeme
 - Pokud je část trasy pokrytá platným kupónem i platnou aktivovanou jízdenkou, tak jízdenka má při hlášce přednost, tedy vrací se hláška TICKET_PURCHASED.
 - Logika stávajícího výpočtu ceny se nemění, pouze se bude rozdílně zpracovávat kupón a jízdenka.
- Poznámka: vychází z issue: <https://gitlab.com/operator-ict/ipt/umotional/-/issues/517>

Cenová nabídka

Název položky	Sazba / J bez DPH v Kč	Jednotka (J)	Počet J	Sazba celkem bez DPH v Kč
<i>tester</i>	3.520,- Kč	MD	14	49 280,- Kč
<i>senior analytik</i>	5.440,- Kč	MD	25	136 000,- Kč
<i>projektový manažer</i>	5.440,- Kč	MD	15	81 600,- Kč
<i>konzultant</i>	6.160,- Kč	MD	14	86 240,- Kč
<i>vývojář</i>	6.160,- Kč	MD	73	449 680,- Kč
<i>UX designer</i>	6.160,- Kč	MD	4	24 640,- Kč
<i>databázový specialista</i>	6.160,- Kč	MD	12	73 920,- Kč
<i>senior konzultant</i>	6.720,- Kč	MD	10	67 200,- Kč
<i>manažer</i>	8.160,- Kč	MD	4	32 640,- Kč
<i>SW architekt</i>	11.920,- Kč	MD	5	59 600,- Kč
Celková cena v Kč bez DPH			176	1 060.800,- Kč

Platební podmínky

Faktura za plnění bude vystavena nejdříve po dokončení poskytnutého plnění, které bude potvrzeno objednatelem.

Doba plnění a místo plnění

Termín dodání: 15 týdnů od nabytí účinnosti objednávky.

Kontaktní a další informace

Tato nabídka má pouze informativní charakter a není závazným návrhem k uzavření smlouvy.

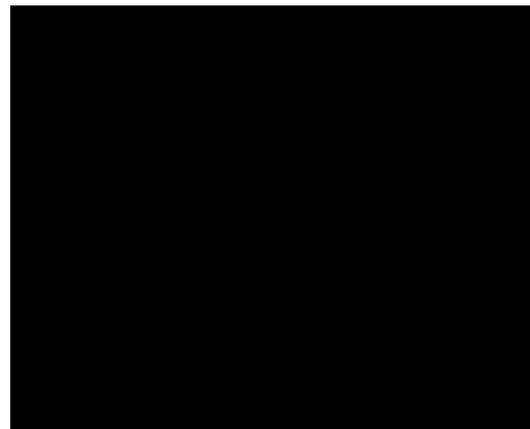
Operátor ICT, a.s.

www.operatorict.cz

IČO: 02795281

DIČ: CZ02795281

V Praze dne 20.7.2023



PŘÍLOHY

Bez přílohy



Akceptační doložka

Poskytovatel Objednávku č. 440 ze dne 28. 7. 2023 akceptuje, a to za podmínek uvedených v nabídce Poskytovatele ze dne 20. 7. 2023.

Za Poskytovatele spol. Operátor ICT, a.s.

V Praze dne (dle el. podpisu)

