

Piattaforma Digitale Nazionale Dati

Documentazione tecnica e-service di Regione Lombardia

Orari e Percorsi del trasporto pubblico locale

Versione 1

Indice

| | |
|--|----|
| 1. Introduzione..... | 3 |
| 2. Autenticazione | 3 |
| 3. Descrizione degli ambienti..... | 3 |
| 4. Metodi di chiamata..... | 3 |
| 4.1 Servizio “search” | 3 |
| 4.1.1 Metodo GET | 3 |
| 4.1.2 Metodo POST | 5 |
| 4.2 Servizio “solution” | 8 |
| 4.2.1 Metodo GET | 8 |
| 4.2.2 Metodo POST | 15 |
| 4.3 Servizio “timetables” | 23 |
| 4.3.1 Metodo GET | 23 |
| 4.3.2 Metodo POST | 26 |
| 4.4 Servizio “timetablesValidity” | 30 |
| 4.4.1 Metodo GET Il metodo viene utilizzato per ricercare le validità degli orari: | 30 |
| 5. Uso dei codici di stato http | 32 |

1. Introduzione

La API permette di cercare orari e percorsi per muoversi in Lombardia con treni, autobus, metropolitane, tram e altri servizi di trasporto pubblico.

Per qualsiasi esigenza relativa all'e-service è possibile richiedere assistenza alla seguente mail:
assistenza_api@regione.lombardia.it

2. Autenticazione

L'autenticazione sarà gestita attraverso la piattaforma PDND che genera un voucher di 30 minuti e di tipo *Bearer JWT*. Tale e-service non prevede ulteriori pattern di interoperabilità MODI.

3. Descrizione degli ambienti

Gli URL Base rispettivamente dell'ambiente di collaudo e produzione in PDND sono i seguenti:

- URL Base di Collaudo: <https://api.integrazione.lispa.it/c/pdndbasic/erogazione/tplapi>
- URL Base di Produzione: <https://api.servizirl.it/c/pdndbasic/erogazione/tplapi>

4. Metodi di chiamata

Di seguito sono descritti i metodi messi a disposizione del servizio con i relativi parametri da valorizzare.

4.1 Servizio "search"

Questo servizio può essere utilizzato per ricercare dei punti a partire da una descrizione. Sulla base del testo inserito per effettuare la ricerca, il servizio restituisce una lista di punti, ognuno dei quali corredato da una descrizione, una tipologia e la coppia di coordinate, che ne individua la posizione. A tale scopo sono messi a disposizione i metodi GET e POST.

4.1.1 Metodo GET

Il metodo viene utilizzato per ricercare punti utilizzando il query params per indicare i criteri di ricerca.

- ENDPOINT: Url Base + `"/tplapi/v1.0.0/search?param=<valore>&maxResult=<valore>"`
- METODO: GET
- HEADER:
 - Content-Type: application/Json
 - Authorization: Bearer Token_Di_Accesso.

4.1.1.1 Parametri di richiesta

Nel caso di GET, non è previsto body e viene utilizzato il query params per indicare i criteri di ricerca.

I parametri richiesti sono i seguenti:

- "param": Stringa di ricerca del punto di partenza/arrivo. Stringa con lunghezza tra 2 e 100 caratteri. Obbligatorio.
- "maxResult": Numero massimo di risultati restituiti dal servizio. Numero intero da 1 a 200. Se =0 o null, si assume il valore di default =200. Se si inserisce un numero superiore a 200, si assume il limite massimo (=200).

4.1.2.2 Campi della risposta

Il JSON di risposta conterrà, i seguenti elementi:

| Nome campo | Descrizione | Tipo di dato | Mandatorietà | Valori Ammissibili |
|------------|---|------------------|---|---|
| stato | Codice di stato della risposta | Numero intero | Obbligatorio | <ul style="list-style-type: none"> • -1=fault (valorizzato oggetto "errori") • 0=risolutivo (valorizzato oggetto "points") |
| points | Oggetto contenente la lista dei punti trovati | Oggetto | Nel caso in cui non vengano trovati punti, nel Json l'array dei punti sarà vuoto | |
| label | Nome del punto | Stringa 100 Char | | |
| type | Tipologia del punto trovato | Stringa | | <ul style="list-style-type: none"> • "areadifermata": fermata/stazione del trasporto pubblico locale; • "civico": indirizzo comprensivo di numero civico; • "comune": comune, di norma identificato dal municipio; • "indirizzo": indirizzo senza numero civico; • "poi": (punto di interesse) |
| x | Coordinata X del punto espressa nel sistema di riferimento WGS84. | Numero | | Numero max. 6 decimali, separatore ".". |
| y | Coordinata Y del punto espressa nel sistema di riferimento WGS84. | Numero | | Numero max. 6 decimali, separatore ".". |
| errori | Oggetto contenente la lista di eventuali errori | Oggetto | Nel caso in cui non vengano trovati errori, nel Json l'array degli errori sarà vuoto. | |
| codice | Codice dell'errore | Stringa | | |

| descrizione | Descrizione dell'errore | Stringa | | |
|-------------|-------------------------|---------|--|--|
|-------------|-------------------------|---------|--|--|

4.1.1.3 Esempio di risposta

Si fornisce un esempio di JSON di risposta restituito in caso di successo:

```
{
  "stato": "0",
  "points": [
    {
      "label": "Milano, Cadorna M1 M2, Staz. Ferroviaria,triennale",
      "type": "areadifermata",
      "x": "9.176402",
      "y": "45.468428"
    },
    {
      "label": "Milano, Porta Venezia M1, Stazione Ferroviaria",
      "type": "areadifermata",
      "x": "9.206242",
      "y": "45.475253"
    },
    {
      "label": "Saronno, P.za Cadorna, Stazione Ferroviaria",
      "type": "areadifermata",
      "x": "9.030598",
      "y": "45.625635"
    }
  ],
  "errori": []
}
```

4.1.1.4 Codici d'errore

Il servizio "search" può ritornare i seguenti errori:

| Codice | Descrizione |
|--------|---|
| ER001 | Formato della stringa del campo param non valida: il numero di caratteri deve essere compreso tra 2 e 100 |
| ER002 | Formato del campo maxResult non valido: inserire un numero intero da 1 a 200 |
| ER900 | Request Not Valid |
| ER901 | Messaggio di errore generato dal servizio |

4.1.2 Metodo POST

Il metodo viene utilizzato per ricercare punti utilizzando il body per indicare i criteri di ricerca.

- ENDPOINT: Url Base + tplapi/v1.0.0/search
- METODO: POST
- HEADER:

- Content-Type: application/Json
- Authorization: Bearer Token_Di_Accesso

4.1.2.1 Parametri di richiesta

Nel caso di POST, del campo Body si utilizzerà la tipologia "raw". Il valore da inserire nel Body è un JSON formato come nell'esempio che segue:

```
{
  "text": "Milano Bovisa",
  "maxResult": 10
}
```

Il significato dei parametri è il seguente:

- "param": Stringa di ricerca del punto di partenza/arrivo. Stringa con lunghezza tra 2 e 100 caratteri. Obbligatorio.
- "maxResult": Numero massimo di risultati restituiti dal servizio. Numero intero da 1 a 200. Se =0 o null, si assume il valore di default =200. Se si inserisce un numero superiore a 200, si assume il limite massimo (=200).

4.1.2.2 Campi della risposta

Il JSON di risposta conterrà, i seguenti elementi:

| Nome campo | Descrizione | Tipo di dato | Mandatorietà | Valori Ammissibili |
|------------|---|------------------|--|--|
| stato | Codice di stato della risposta | Numero intero | Obbligatorio | <ul style="list-style-type: none"> • -1=fault (valorizzato oggetto "errori") • 0=risolutivo (valorizzato oggetto "points") |
| points | Oggetto contenente la lista dei punti trovati | Oggetto | Nel caso in cui non vengano trovati punti, nel Json l'array dei punti sarà vuoto | |
| label | Nome del punto | Stringa 100 Char | | |
| type | Tipologia del punto trovato | Stringa | | <ul style="list-style-type: none"> • "areadifermata": fermata/stazione del trasporto pubblico locale; • "civico": indirizzo comprensivo di numero civico; • "comune": comune, di norma identificato dal municipio; • "indirizzo": indirizzo senza numero civico; |

| | | | | |
|-------------|---|---------|---|---|
| | | | | <ul style="list-style-type: none"> • “poi”: (punto di interesse) |
| x | Coordinata X del punto espressa nel sistema di riferimento WGS84. | Numero | | Numero max. 6 decimali, separatore “.”. |
| y | Coordinata Y del punto espressa nel sistema di riferimento WGS84. | Numero | | Numero max. 6 decimali, separatore “.”. |
| errori | Oggetto contenente la lista di eventuali errori | Oggetto | Nel caso in cui non vengano trovati errori, nel Json l'array degli errori sarà vuoto. | |
| codice | Codice dell'errore | Stringa | | |
| descrizione | Descrizione dell'errore | Stringa | | |

4.1.2.3 Esempio di risposta

Si fornisce un esempio di JSON di risposta restituito in caso di successo:

```
{
  "stato":
    "0",
  "points":
    [
      {
        "label": "Milano, Affori M3, Stazione Ferroviaria",
        "type": "areadifermata",
        "x": "9.167727",
        "y": "45.521221"
      },
      {
        "label": "Milano, Affori Centro M3, V. P. Rossi",
        "type": "areadifermata",
        "x": "9.173685",
        "y": "45.513921"
      }
    ],
  "errori": []
}
```

4.1.2.4 Codici d'errore

Il servizio “search” può ritornare i seguenti errori:

| Codice | Descrizione |
|--------|---|
| ER001 | Formato della stringa del campo param non valida: il numero di caratteri deve essere compreso tra 2 e 100 |
| ER002 | Formato del campo maxResult non valido: inserire un numero intero da 1 a 200 |
| ER900 | Request Not Valid |
| ER901 | Messaggio di errore generato dal servizio |

4.2 Servizio “solution”

Questo servizio può essere utilizzato per ricercare le soluzioni di viaggio (percorsi e tratte elementari che lo compongono) con il trasporto pubblico locale tra un punto di partenza ad un punto di arrivo in un certo momento. Esso restituisce le informazioni sui percorsi trovati in base ai parametri di ricerca utilizzati.

Il formato e la tipologia dei punti di partenza e di arrivo sono gli stessi descritti dal servizio precedente (Search).

Si noti che i percorsi proposti dal servizio sono ordinati (attributo idPercorso) secondo una logica, interna al motore di travel planning, che considera l’orario di partenza e la durata complessiva del viaggio.

4.2.1 Metodo GET

Il metodo viene utilizzato per ricercare soluzioni di viaggio utilizzando il query params per indicare i criteri di ricerca;

- METODO: GET
- ENDPOINT: Url Base + “/tplapi/v1.0.0/solution/?param={...}”
- HEADER:
 - Content-Type: application/json
 - Authorization: Bearer Token_Di_Accesso

4.2.1.1 Parametri di richiesta

Nel caso di GET, non è previsto body e viene utilizzato il query params per indicare i criteri di ricerca.

In caso di GET, il nome del parametro da fornire per una ricerca valida dei percorsi di viaggio è “param” ed avrà come formato un JSON strutturato in questo modo:

```
param={"richiesta":
  {
    "from": "valore",
    "fromX": "valore",
    "fromY": "valore",
    "to": "valore",
    "toX": "valore",
    "toY": "valore",
    "date": "valore",
    "when": "valore",
    "options": ["valore","valore"],
    "changeNumber": "valore",
    "durationChange": "valore",
    "tipoData": "valore"
  },
  "lang": "valore"
}
```

I valori dei parametri devono essere passati tra apici “”.

Il JSON conterrà i seguenti attributi:

- “from”: Descrizione del punto di partenza. Stringa con lunghezza massima di 100 caratteri. Opzionale.
- “fromX”: Coordinata x del punto di partenza espressa nel sistema di riferimento WGS84. Numero compreso tra -180 +180. Separatore “.”. Obbligatorio.
- “fromY”: Coordinata y del punto di partenza espressa nel sistema di riferimento WGS84. Numero compreso tra -90 +90. Separatore “.”. Obbligatorio.
- “to”: Descrizione del punto di arrivo. Stringa con lunghezza massima di 100 caratteri. Opzionale.
- “toX”: Coordinata x del punto di partenza espressa nel sistema di riferimento WGS84. Numero compreso tra -180 +180. Separatore “.”. Obbligatorio.
- “toY”: Coordinata y del punto di partenza espressa nel sistema di riferimento WGS84. Numero compreso tra -90 +90. Separatore “.”. Obbligatorio.
- “date”: Data a cui è riferita la ricerca delle soluzioni di viaggio. Data in formato dd/mm/yyyy. Obbligatorio.
- “when”: Ora a cui è riferita la ricerca delle soluzioni di viaggio. Orario in formato hh:mm. Obbligatorio.
- “options”: Mezzi coinvolti nella ricerca delle soluzioni di viaggio. Array di numeri. Obbligatorio. Valori ammissibili:
 - 1=Treno
 - 2=Metro
 - 3=Bus
 - 4=Tram
 - 5=Battello
 - 6=Altro (funivia)
- “changeNumber”: Numero massimo di cambi desiderati. Intero. Opzionale. Se null, si assume il valore di default. Valori ammissibili:
 - -1=qualsiasi numero di cambi (default)
 - 0=nessun cambio o 1=1 cambio o 2=2 cambi o 3=3 cambi
- “durationChange”: Durata minima del cambio. Intero. Opzionale. Se null, si assume il valore di default. Valori ammissibili:
 - -1=qualsiasi durata (default) o 0=maggiore di 4 minuti o 200=maggiore di 10 minuti o 400=maggiore di 20 minuti
- “tipoData”: Indica se i parametri “date” e “when” si riferiscono alla partenza o all’arrivo. Intero. Opzionale. Se null, si assume il valore di default. Valori ammissibili:
 - 0=data e orario si riferiscono alla partenza (default) o 1=data e orario si riferiscono all’arrivo
- “lang”: Lingua di riferimento per la restituzione dei messaggi di errore eventualmente generati. Stringa. Opzionale. Se null, si assume il valore di default. Valori ammissibili:
 - it=italiano (default)
 - en=inglese o fr=francese o de=tedesco

4.2.1.2 Campi della risposta

Il JSON di risposta conterrà, i seguenti elementi:

| Nome campo | Descrizione | Tipo di dato | Mandatorietà | Valori ammissibili |
|-----------------|---|----------------------|---|--|
| stato | Codice di stato della risposta | Numero intero | Obbligatorio | <ul style="list-style-type: none"> -1=fault (valorizzato oggetto "errori") 0=risolutivo (valorizzato oggetto "lista Percorsi") |
| lang | Lingua di riferimento per la restituzione dei messaggi di errore eventualmente generati | Stringa | Obbligatorio | <ul style="list-style-type: none"> it=italiano (default) en=inglese fr=francese de=tedesco |
| richiesta | <i>Oggetto contenente l'elenco dei parametri di ricerca della request</i> | Oggetto | | <i>I campi opzionali della request sono sempre restituiti, eventualmente vuoti.</i> |
| listaPercorsi | <i>Oggetto contenente l'elenco dei percorsi che compongono la soluzione</i> | Oggetto | <i>Nel caso in cui non vengano trovate soluzioni di viaggio, nel Json l'array dei percorsi sarà vuoto</i> | <i>La risposta può contenere un numero variabile di entità percorso (massimo 6), tratta e fermata.</i> |
| idPercorso | Ordine progressivo delle soluzioni di viaggio trovate | Intero | Obbligatorio | |
| durata | Durata complessiva del percorso in minuti. | Ora in formato hh:mm | Obbligatorio | |
| numeroCambi | Numero di cambi necessari per effettuare il viaggio | Intero | Obbligatorio | |
| mezziPercorso[] | Elenco delle tipologie di mezzo coinvolte nella soluzione. | Array di numeri | Obbligatorio | <ul style="list-style-type: none"> 1=Treno 2=Metro 3=Bus 4=Tram 5=Battello 6=Altro (funivia) |
| partenza | Descrizione del punto di partenza della soluzione | Stringa 100 Char | | |
| oraPartenza | Orario di partenza della soluzione | Ora in formato hh:mm | Obbligatorio | |
| xPartenza | Coordinata X del punto espressa nel sistema di riferimento WGS84. | Numero | Obbligatorio | Numero compreso tra -180 +180 con massimo 6 cifre decimali. Separatore ".". |
| yPartenza | Coordinata Y del punto espressa nel sistema di riferimento WGS84. | Numero | Obbligatorio | Numero compreso tra -90 +90 con massimo 6 cifre decimali. Separatore ".". |
| arrivo | Descrizione del punto di arrivo della soluzione | Stringa 100 Char | | |
| oraArrivo | Orario di arrivo della soluzione | Ora in formato hh:mm | Obbligatorio | |

| | | | | |
|-------------|--|----------------------|--------------|---|
| xArrivo | Coordinata X del punto espressa nel sistema di riferimento WGS84. | Numero | Obbligatorio | Numero compreso tra -180 +180 con massimo 6 cifre decimali. Separatore “.”. |
| yArrivo | Coordinata Y del punto espressa nel sistema di riferimento WGS84. | Numero | Obbligatorio | Numero compreso tra -90 +90 con massimo 6 cifre decimali. Separatore “.”. |
| listaTratte | <i>Oggetto contenente l'elenco delle tratte che compongono il percorso della soluzione</i> | <i>Oggetto</i> | | |
| idTratta | Ordine progressivo delle tratte che compongono la soluzione | Intero | Obbligatorio | |
| durata | Durata della tratta in minuti. | Ora in formato hh:mm | Obbligatorio | |
| linea | Codice della linea contenente la tratta. | Stringa 6 Char | | È restituito vuoto se è un tratto a piedi: ""=Tratto a piedi |
| gestore | Codice del gestore che effettua il servizio. | Stringa | | È restituito vuoto se è un tratto a piedi: ""=Tratto a piedi |
| mezzo | Identificativo del tipo di mezzo che percorre il tratto. È restituito vuoto se è un tratto a piedi. | Intero | | <ul style="list-style-type: none"> • 1=Treno • 2=Metro • 3=Bus • 4=Tram • 5=Battello • 6=Altro (funivia) • ""=Tratto a piedi |
| partenza | Descrizione del punto di partenza della tratta | Stringa 100 Char | | |
| oraPartenza | Orario di partenza. | Ora in formato hh:mm | Obbligatorio | |
| xPartenza | Coordinata X del punto di partenza espressa nel sistema di riferimento WGS84. | Numero | Obbligatorio | Numero compreso tra -180 +180 con massimo 6 cifre decimali. Separatore “.”. |
| yPartenza | Coordinata Y del punto di partenza espressa nel sistema di riferimento WGS84. | Numero | Obbligatorio | Numero compreso tra -90 +90 con massimo 6 cifre decimali. Separatore “.”. |
| arrivo | Descrizione del punto di partenza della tratta | Stringa 100 Char | | |
| oraArrivo | Orario di arrivo | Ora in formato hh:mm | Obbligatorio | |
| xArrivo | Coordinata X del punto di arrivo espressa nel sistema | Numero | Obbligatorio | Numero compreso tra -180 +180 con massimo 6 cifre |

| | | | | |
|--------------|--|----------------------|--|---|
| | di riferimento WGS84. | | | decimali. Separatore " ". |
| yArrivo | Coordinata Y del punto di arrivo espressa nel sistema di riferimento WGS84. | Numero | Obbligatorio | Numero compreso tra -90 +90 con massimo 6 cifre decimali. Separatore " ". |
| note | Note sulla tratta | Stringa | | |
| listaFermate | <i>Oggetto contenente l'elenco delle fermate che compongono la tratta del percorso della soluzione</i> | <i>Oggetto</i> | | |
| idFermata | Ordine progressivo delle fermate nella tratta | Intero | Obbligatorio | |
| nome | Nome della fermata | Stringa | Obbligatorio | |
| arrivo | Orario di arrivo nella fermata. | Ora in formato hh:mm | | |
| partenza | Orario di partenza dalla fermata. | Ora in formato hh:mm | | |
| x | Coordinata X della fermata espressa nel sistema di riferimento WGS84. | Numero | Obbligatorio | Numero compreso tra -180 +180 con massimo 6 cifre decimali. Separatore " ". |
| y | Coordinata Y della fermata espressa nel sistema di riferimento WGS84. | Numero | Obbligatorio | Numero compreso tra -90 +90 con massimo 6 cifre decimali. Separatore " ". |
| errori | <i>Oggetto contenente la lista di eventuali errori</i> | <i>Oggetto</i> | <i>Nel caso in cui non vengano trovati errori, nel Json l'array degli errori sarà vuoto.</i> | |
| codice | Codice dell'errore | Stringa | | |
| descrizione | Descrizione dell'errore | Stringa | | |

4.2.1.3 Esempio di risposta

Si fornisce un esempio di JSON di risposta restituito in caso di successo:

```
{
  "stato": "0",
  "lang": "it",
  "richiesta": {
    ...
  },
  "listaPercorsi": [
    {
      "idPercorso": "1",
      "partenza": "Milano, Via Torquato Taramelli, 24",
      "arrivo": "Busto Arsizio Fs, Stazione",
    }
  ]
}
```

```

"oraPartenza": "11:21",
"oraArrivo": "12:15",
"durata": "00:54",
"numeroCambi": "1",
"listaTratte": [
  {
    ...,
    "listaFermate": [
      {
        ...
      }
    ]
  }
],
"mezziPercorso": [ "2",
  "1"
],
"xPartenza": "9.193987",
"yPartenza": "45.491215",
"xArrivo": "8.864712",
"yArrivo": "45.615779"
}
],
"mezziPercorso": [ "3",
  "1",
  "1"
],
"xPartenza": "9.193987",
"yPartenza": "45.491215",
"xArrivo": "8.864712",
"yArrivo": "45.615779"
}
],
"listaErrori": []
}

```

4.2.1.4 Codici d'errore

Il servizio "solution" può ritornare i seguenti errori:

| Codice | Descrizione |
|--------|---|
| ER003 | Formato della stringa del campo from non valida: la stringa non deve superare i 100 caratteri |
| ER004 | Formato coordinate del campo fromX non valido: numero compreso tra -180 +180 con massimo 6 cifre decimali |
| ER005 | Formato coordinate del campo fromY non valido: numero compreso tra -90 +90 con massimo 6 cifre decimali |
| ER007 | Formato della stringa del campo to non valida: la stringa non deve superare i 100 caratteri |
| ER008 | Formato coordinate del campo toX non valido: numero compreso tra -180 +180 con massimo 6 cifre decimali |
| ER009 | Formato coordinate del campo toY non valido: numero compreso tra -90 +90 con massimo 6 cifre decimali |
| ER011 | Formato campo date non valido: inserire una data nel formato dd/mm/aaaa |

| | |
|--------|--|
| ER012 | Formato campo when non valido: inserire un orario nel formato hh:mm |
| ER013 | Formato campo options non valido. Valori ammissibili: 1, 2, 3, 4, 5, 6 separati da virgola |
| ER014 | Formato campo changeNumber non valido. Valori ammissibili: -1, 0, 1, 2, 3 |
| ER015 | Formato campo durationChange non valido. Valori ammissibili: -1, 0, 200, 400 |
| ER016 | Formato campo tipoData non valido. Valori ammissibili: 0, 1 |
| ER900 | Request Not Valid |
| ER901 | Messaggio di errore generato dal servizio |
| K94002 | Stazione non presente. |
| K9380 | Stazione definita più di una volta. |
| K9360 | Data al di fuori del periodo di validità dell'orario. |
| K9320 | Dati non corretti o incompleti. |
| K9300 | Stazione d'arrivo inesistente. |
| K9280 | Stazione intermedia inesistente. |
| K9260 | Stazione di partenza inesistente. |
| K9250 | Richiesta interrotta. |
| K9240 | Errore interno. |
| K9230 | Overflow del sistema. |
| K9220 | Nessuna stazione trovata nelle vicinanze dell'indirizzo indicato. |
| K901 | Nessun collegamento trovato. |
| K900 | Nessun collegamento trovato. Verificare i campi obbligatori. |
| K895 | Nessun collegamento trovato. Verificare che partenza e arrivo non siano troppo vicini. |

| | |
|------|--|
| K892 | Nessun collegamento trovato. |
| K891 | Nessun percorso trovato. Provare ad inserire una fermata intermedia. |
| K890 | Nessun collegamento trovato. |

4.2.2 Metodo POST

Il metodo viene utilizzato per ricercare soluzioni di viaggio utilizzando il body per indicare i criteri di ricerca. È possibile impostare il numero di soluzioni di viaggio desiderate e ottenere la lista completa delle fermate della corsa della singola tratta.

- METODO: POST
- ENDPOINT: Url Base + /tplapi/v1.0.0/solution
- HEADER:
 - Content-Type: application/json
 - Authorization: Bearer Token_Di_Accesso

4.2.2.1 Parametri di richiesta

Il JSON conterrà, i seguenti attributi:

- “from”: Descrizione del punto di partenza. Stringa con lunghezza massima di 100 caratteri. Opzionale.
- “fromX”: Coordinata x del punto di partenza espressa nel sistema di riferimento WGS84. Numero compreso tra -180 +180. Separatore “.”. Obbligatorio.
- “fromY”: Coordinata y del punto di partenza espressa nel sistema di riferimento WGS84. Numero compreso tra -90 +90. Separatore “.”. Obbligatorio.
- “to”: Descrizione del punto di arrivo. Stringa con lunghezza massima di 100 caratteri. Opzionale.
- “toX”: Coordinata x del punto di partenza espressa nel sistema di riferimento WGS84. Numero compreso tra -180 +180. Separatore “.”. Obbligatorio.
- “toY”: Coordinata y del punto di partenza espressa nel sistema di riferimento WGS84. Numero compreso tra -90 +90. Separatore “.”. Obbligatorio.
- “date”: Data a cui è riferita la ricerca delle soluzioni di viaggio. Data in formato dd/mm/yyyy. Obbligatorio.
- “when”: Ora a cui è riferita la ricerca delle soluzioni di viaggio. Orario in formato hh:mm. Obbligatorio.
- “options”: Mezzi coinvolti nella ricerca delle soluzioni di viaggio. Array di numeri. Obbligatorio.
Valori ammissibili:
 - 1=Treno
 - 2=Metro
 - 3=Bus
 - 4=Tram

- 5=Battello
- 6=Altro (funivia)
- “changeNumber”: Numero massimo di cambi desiderati. Intero. Opzionale. Se null, si assume il valore di default. Valori ammissibili:
 - -1=qualsiasi numero di cambi (default)
 - 0=nessun cambio
 - 1=1 cambio
 - 2=2 cambi
 - 3=3 cambi
- “durationChange”: Durata minima del cambio. Intero. Opzionale. Se null, si assume il valore di default. Valori ammissibili:
 - -1=qualsiasi durata (default)
 - 0=maggiore di 4 minuti
 - 200=maggiore di 10 minuti
 - 400=maggiore di 20 minuti
- “tipoData”: Indica se i parametri “date” e “when” si riferiscono alla partenza o all’arrivo. Intero. Opzionale. Se null, si assume il valore di default. Valori ammissibili:
 - 0=data e orario si riferiscono alla partenza (default)
 - 1=data e orario si riferiscono all’arrivo
- “nSolutions”: Numero massimo di soluzioni restituite dal servizio. Numero intero da 1 a 6. Opzionale. Se null, si assume il valore di default.
- “lang”: Lingua di riferimento per la restituzione dei messaggi di errore eventualmente generati. Stringa. Opzionale. Se null, si assume il valore di default. Valori ammissibili:
 - it=italiano (default)
 - en=inglese
 - fr=francese
 - de=tedesco

4.2.2.2 Campi della risposta

Il JSON conterrà i seguenti elementi:

| Nome campo | Descrizione | Tipo di dato | Mandatorietà | Valori ammissibili |
|------------|---|---------------|--------------|--|
| stato | Codice di stato della risposta | Numero intero | Obbligatorio | <ul style="list-style-type: none"> • -1=fault (valorizzato oggetto “errori”) • 0=risolutivo (valorizzato oggetto “lista Percorsi”) |
| lang | Lingua di riferimento per la restituzione dei messaggi di errore eventualmente generati | Stringa | Obbligatorio | <ul style="list-style-type: none"> • it=italiano (default) • en=inglese • fr=francese • de=tedesco |
| richiesta | Oggetto contenente l'elenco dei parametri | Oggetto | | I campi opzionali della request sono |

| | | | | |
|----------------------|--|----------------------|---|--|
| | <i>di ricerca della request</i> | | | <i>sempre restituiti, eventualmente vuoti.</i> |
| <i>listaPercorsi</i> | <i>Oggetto contenente l'elenco dei percorsi che compongono la soluzione</i> | <i>Oggetto</i> | <i>Nel caso in cui non vengano trovate soluzioni di viaggio, nel Json l'array dei percorsi sarà vuoto</i> | <i>La risposta può contenere un numero variabile di entità percorso (massimo 6), tratta e fermata.</i> |
| idPercorso | Ordine progressivo delle soluzioni di viaggio trovate | Intero | Obbligatorio | |
| durata | Durata complessiva del percorso in minuti. | Ora in formato hh:mm | Obbligatorio | |
| numeroCambi | Numero di cambi necessari per effettuare il viaggio | Intero | Obbligatorio | |
| mezziPercorso[] | Elenco delle tipologie di mezzo coinvolte nella soluzione. | Array di numeri | Obbligatorio | <ul style="list-style-type: none"> • 1=Treno • 2=Metro • 3=Bus • 4=Tram • 5=Battello • 6=Altro (funivia) |
| partenza | Descrizione del punto di partenza della soluzione | Stringa 100 Char | | |
| oraPartenza | Orario di partenza della soluzione | Ora in formato hh:mm | Obbligatorio | |
| xPartenza | Coordinata X del punto espressa nel sistema di riferimento WGS84. | Numero | Obbligatorio | Numero compreso tra -180 +180 con massimo 6 cifre decimali. Separatore ".". |
| yPartenza | Coordinata Y del punto espressa nel sistema di riferimento WGS84. | Numero | Obbligatorio | Numero compreso tra -90 +90 con massimo 6 cifre decimali. Separatore ".". |
| arrivo | Descrizione del punto di arrivo della soluzione | Stringa 100 Char | | |
| oraArrivo | Orario di arrivo della soluzione | Ora in formato hh:mm | Obbligatorio | |
| xArrivo | Coordinata X del punto espressa nel sistema di riferimento WGS84. | Numero | Obbligatorio | Numero compreso tra -180 +180 con massimo 6 cifre decimali. Separatore ".". |
| yArrivo | Coordinata Y del punto espressa nel sistema di riferimento WGS84. | Numero | Obbligatorio | Numero compreso tra -90 +90 con massimo 6 cifre decimali. Separatore ".". |
| <i>listaTratte</i> | <i>Oggetto contenente l'elenco delle tratte che compongono il percorso della soluzione</i> | <i>Oggetto</i> | | |
| idTratta | Ordine progressivo delle tratte che | Intero | Obbligatorio | |

| | | | | |
|--------------|--|----------------------|--------------|---|
| | compongono la soluzione | | | |
| durata | Durata della tratta in minuti. | Ora in formato hh:mm | Obbligatorio | |
| linea | Codice della linea contenente il tratto. | Stringa 6 Char | | È restituito vuoto se è un tratto a piedi: ""=Tratto a piedi |
| gestore | Codice del gestore che effettua il servizio. | Stringa | | È restituito vuoto se è un tratto a piedi: ""=Tratto a piedi |
| mezzo | Identificativo del tipo di mezzo che percorre il tratto. È restituito vuoto se è un tratto a piedi. | Intero | | <ul style="list-style-type: none"> • 1=Treno • 2=Metro • 3=Bus • 4=Tram • 5=Battello • 6=Altro (funivia) • ""=Tratto a piedi |
| partenza | Descrizione del punto di partenza della tratta | Stringa 100 Char | | |
| oraPartenza | Orario di partenza. | Ora in formato hh:mm | Obbligatorio | |
| xPartenza | Coordinata X del punto di partenza espressa nel sistema di riferimento WGS84. | Numero | Obbligatorio | Numero compreso tra -180 +180 con massimo 6 cifre decimali. Separatore ".". |
| yPartenza | Coordinata Y del punto di partenza espressa nel sistema di riferimento WGS84. | Numero | Obbligatorio | Numero compreso tra -90 +90 con massimo 6 cifre decimali. Separatore ".". |
| arrivo | Descrizione del punto di arrivo della tratta | Stringa 100 Char | | |
| oraArrivo | Orario di arrivo | Ora in formato hh:mm | Obbligatorio | |
| xArrivo | Coordinata X del punto di arrivo espressa nel sistema di riferimento WGS84. | Numero | Obbligatorio | Numero compreso tra -180 +180 con massimo 6 cifre decimali. Separatore ".". |
| yArrivo | Coordinata Y del punto di arrivo espressa nel sistema di riferimento WGS84. | Numero | Obbligatorio | Numero compreso tra -90 +90 con massimo 6 cifre decimali. Separatore ".". |
| note | Note sulla tratta | Stringa | | |
| listaFermate | <i>Oggetto contenente l'elenco delle fermate che compongono la tratta del percorso della soluzione</i> | <i>Oggetto</i> | | |
| idFermata | Ordine progressivo delle fermate nella tratta | Intero | Obbligatorio | |

| | | | | |
|-----------------------|---|----------------------|---|---|
| nome | Nome della fermata | Stringa | Obbligatorio | |
| arrivo | Orario di arrivo nella fermata. | Ora in formato hh:mm | | |
| partenza | Orario di partenza dalla fermata. | Ora in formato hh:mm | | |
| x | Coordinata X della fermata espressa nel sistema di riferimento WGS84. | Numero | Obbligatorio | Numero compreso tra -180 +180 con massimo 6 cifre decimali. Separatore ".". |
| y | Coordinata Y della fermata espressa nel sistema di riferimento WGS84. | Numero | Obbligatorio | Numero compreso tra -90 +90 con massimo 6 cifre decimali. Separatore ".". |
| listaPercorsoCompleto | Oggetto contenente l'elenco delle fermate che compongono l'intera corsa della soluzione (quindi anche fermate precedenti o successive della tratta) | Oggetto | | |
| idFermata | Ordine progressivo delle fermate nella tratta | Intero | Obbligatorio | |
| nome | Nome della fermata | Stringa | Obbligatorio | |
| arrivo | Orario di arrivo nella fermata. | Ora in formato hh:mm | | |
| partenza | Orario di partenza dalla fermata. | Ora in formato hh:mm | | |
| x | Coordinata X della fermata espressa nel sistema di riferimento WGS84. | Numero | Obbligatorio | Numero compreso tra -180 +180 con massimo 6 cifre decimali. Separatore ".". |
| y | Coordinata Y della fermata espressa nel sistema di riferimento WGS84. | Numero | Obbligatorio | Numero compreso tra -90 +90 con massimo 6 cifre decimali. Separatore ".". |
| errori | Oggetto contenente la lista di eventuali errori | Oggetto | Nel caso in cui non vengano trovati errori, nel Json l'array degli errori sarà vuoto. | |
| codice | Codice dell'errore | Stringa | | |
| descrizione | Descrizione dell'errore | Stringa | | |

4.2.2.3 Esempio di risposta

Si fornisce un esempio di JSON di risposta restituito in caso di successo:

```
{
  "stato": "0",
```

```
"lang": "it", "richiesta": {  
  ...  
},  
"listaPercorsi": [  
  {  
    "idPercorso": "1",  
    "partenza": "Milano, Via Torquato Taramelli, 24",  
    "arrivo": "Busto Arsizio Fs, Stazione",  
    "oraPartenza": "11:21",  
    "oraArrivo": "12:15",  
    "durata": "00:54",  
    "numeroCambi": "1",  
    "listaTratte": [  
      {  
        ...  
        "listaFermate": [  
          {  
            ...  
          }  
        ],  
        "listaPercorsoCompleto": [  
          {  
            ...  
          }  
        ]  
      }  
    ],  
    "mezziPercorso": [ "2",  
      "1"  
    ],  
    "xPartenza": "9.193987",  
    "yPartenza": "45.491215",  
    "xArrivo": "8.864712",  
    "yArrivo": "45.615779"  
  }  
]
```

```

],
"mezziPercorso": [ "3",
  "1",
  "1"
],
"xPartenza": "9.193987",
"yPartenza": "45.491215",
"xArrivo": "8.864712",
"yArrivo": "45.615779"
}
],
"listaErrori": []
}

```

4.2.2.4 Codici d'errore

Il servizio “solution” può ritornare i seguenti errori:

| Codice | Descrizione |
|--------|---|
| ER003 | Formato della stringa del campo from non valida: la stringa non deve superare i 100 caratteri |
| ER004 | Formato coordinate del campo fromX non valido: numero compreso tra -180 +180 con massimo 6 cifre decimali |
| ER005 | Formato coordinate del campo fromY non valido: numero compreso tra -90 +90 con massimo 6 cifre decimali |
| ER007 | Formato della stringa del campo to non valida: la stringa non deve superare i 100 caratteri |
| ER008 | Formato coordinate del campo toX non valido: numero compreso tra -180 +180 con massimo 6 cifre decimali |
| ER009 | Formato coordinate del campo toY non valido: numero compreso tra -90 +90 con massimo 6 cifre decimali |
| ER011 | Formato campo date non valido: inserire una data nel formato dd/mm/aaaa |
| ER012 | Formato campo when non valido: inserire un orario nel formato hh:mm |
| ER013 | Formato campo options non valido. Valori ammissibili: 1, 2, 3, 4, 5, 6 separati da virgola |
| ER014 | Formato campo changeNumber non valido. Valori ammissibili: -1, 0, 1, 2, 3 |
| ER015 | Formato campo durationChange non valido. Valori ammissibili: -1, 0, 200, 400 |

| | |
|--------|--|
| ER016 | Formato campo tipoData non valido. Valori ammissibili: 0, 1 |
| ER900 | Request Not Valid |
| ER901 | Messaggio di errore generato dal servizio |
| K94002 | Stazione non presente. |
| K9380 | Stazione definita più di una volta. |
| K9360 | Data al di fuori del periodo di validità dell'orario. |
| K9320 | Dati non corretti o incompleti. |
| K9300 | Stazione d'arrivo inesistente. |
| K9280 | Stazione intermedia inesistente. |
| K9260 | Stazione di partenza inesistente. |
| K9250 | Richiesta interrotta. |
| K9240 | Errore interno. |
| K9230 | Overflow del sistema. |
| K9220 | Nessuna stazione trovata nelle vicinanze dell'indirizzo indicato. |
| K901 | Nessun collegamento trovato. |
| K900 | Nessun collegamento trovato. Verificare i campi obbligatori. |
| K895 | Nessun collegamento trovato. Verificare che partenza e arrivo non siano troppo vicini. |
| K892 | Nessun collegamento trovato. |
| K891 | Nessun percorso trovato. Provare ad inserire una fermata intermedia. |
| K890 | Nessun collegamento trovato. |

4.3 Servizio “timetables”

Questo servizio può essere utilizzato per ricercare gli orari ad una fermata.

Esso restituisce le informazioni sugli orari di passaggio delle corse alla fermata, trovati in base ai parametri di ricerca utilizzati.

A tale scopo sono messi a disposizione i metodi seguenti:

- GET sugli endpoint /timetables per ricercare gli orari utilizzando il query params per indicare i criteri di ricerca;
- POST sull'endpoint /timetables, per ricercare gli orari utilizzando il body per indicare i criteri di ricerca.

4.3.1 Metodo GET

Il metodo viene utilizzato per ricercare gli orari utilizzando il query params per indicare i criteri di ricerca;

- METODO: GET
- ENDPOINT: Url Base + “/tplapi/v1.0.0/timetables/?param={...}”
- HEADER:
 - Content-Type: application/json
 - Authorization: Bearer Token_Di_Accesso

4.3.1.1 Parametri di richiesta

Nel caso di GET, non è previsto body e viene utilizzato il query params per indicare i criteri di ricerca.

Il nome del parametro da fornire per una ricerca valida degli orari di passaggio alla fermata è “param” ed avrà come formato un Json strutturato in questo modo:

```
param={"lang": "valore",  
      "richiesta": {  
        "stop": "valore",  
        "date": "valore",  
        "when": "valore",  
        "options": ["valore"],  
        "tipoData": "valore"  
      }  
}
```

I valori dei parametri devono essere passati tra apici “”.

Il JSON conterrà i seguenti attributi:

- “lang”: Lingua di riferimento per la restituzione dei messaggi di errore eventualmente generati. Stringa. Opzionale. Se null, si assume il valore di default. Valori ammissibili:
 - it=italiano (default)
 - en=inglese
 - fr=francese
 - de=tedesco

- “stop”: Descrizione della fermata di cui si ricercano gli orari. Stringa con lunghezza massima di 100 caratteri. Obbligatorio.
- “date”: Data a cui è riferita la ricerca. Data in formato dd/mm/yyyy. Obbligatorio.
- “when”: Ora a cui è riferita la ricerca. Orario in formato hh:mm. Obbligatorio.
- “options”: Mezzi coinvolti nella ricerca. Array di numeri. Obbligatorio. Valori ammissibili:
 - 1=Treno
 - 2=Metro
 - 3=Bus
 - 4=Tram
 - 5=Battello
 - 6=Altro (funivia)
- “tipoData”: Indica se i parametri “date” e “when” si riferiscono alla partenza o all’arrivo. Intero. Opzionale. Se null, si assume il valore di default. Valori ammissibili:
 - 0=data e orario si riferiscono alla partenza (default)
 - 1=data e orario si riferiscono all’arrivo

4.3.1.2 Campi della risposta

Il JSON di risposta conterrà, i seguenti elementi:

| Nome campo | Descrizione | Tipo di dato | Mandatorietà | Valori ammissibili |
|------------|---|-------------------------|--------------|--|
| stato | Codice di stato della risposta | Intero | Obbligatorio | <ul style="list-style-type: none"> • -1=fault (valorizzato oggetto “errori”) • 0=risolutivo (valorizzato oggetto “lista Percorsi”) |
| lang | Lingua di riferimento per la restituzione dei messaggi di errore eventualmente generati | Stringa | Obbligatorio | <ul style="list-style-type: none"> • it=italiano (default) • en=inglese • fr=francese • de=tedesco |
| richiesta | <i>Oggetto contenente l’elenco dei parametri di ricerca della request</i> | <i>Oggetto</i> | | <i>I campi opzionali della request sono sempre restituiti, eventualmente vuoti.</i> |
| listaOrari | <i>Oggetto contenente l’elenco degli orari delle corse in passaggio alla fermata</i> | <i>Oggetto</i> | | |
| time | Orario di partenza della corsa | Orario in formato hh:mm | Obbligatorio | |
| run | Descrizione del capolinea della corsa | Stringa 100 Char | Obbligatorio | <ul style="list-style-type: none"> • capolinea di arrivo della corsa nel caso in cui si sia scelto tipoData=0 (ricerca di corse per data e ora di |

| | | | | |
|--------------|---|------------------|--|--|
| | | | | partenza dalla fermata) <ul style="list-style-type: none"> capolinea di partenza della corsa nel caso in cui si sia scelto tipoData=1 (ricerca di corse per data e ora di arrivo alla fermata) |
| line | Codice della linea a cui si riferisce l'orario | Stringa 6 Char | Obbligatorio | |
| transport | Identificativo del tipo di mezzo che effettua la corsa | Array di numeri | Obbligatorio | <ul style="list-style-type: none"> 1=Treno 2=Metro 3=Bus 4=Tram 5=Battello 6=Altro (funivia) |
| note | Eventuali note | Stringa | Opzionale | |
| listaFermate | <i>Oggetto contenente Informazioni relative alla fermata impostata nella ricerca oppure elenco delle possibili fermate nel caso in cui la descrizione della fermata impostata nella ricerca non abbia prodotto un risultato univoco</i> | <i>Oggetto</i> | | |
| Id | Identificativo progressivo della fermata | Intero | Obbligatorio | |
| Stop | Descrizione della fermata di cui si ricercano gli orari | Stringa 100 Char | Obbligatorio | |
| X | Coordinata x della fermata espressa nel sistema di riferimento nel sistema WGS84 | Numero | Obbligatorio | Numero compreso tra -180 e +180, separatore "." |
| y | Coordinata y della fermata espressa nel sistema di riferimento nel sistema WGS84 | Numero | Obbligatorio | Numero compreso tra -90 e +90, separatore "." |
| errori | <i>Oggetto contenente la lista di eventuali errori</i> | <i>Oggetto</i> | <i>Nel caso in cui non vengano trovati errori, nel Json l'array degli errori sarà vuoto.</i> | |
| codice | Codice dell'errore | Stringa | | |
| descrizione | Descrizione dell'errore | Stringa | | |

4.3.1.3 Esempio di risposta

Si fornisce un esempio di JSON di risposta restituito in caso di successo:

```
{
  "stato": "0",
  "lang": "it", "richiesta": {
    "date": "24/03/2021",
    "stop": "Cormano, V. Zara",
    "when": "10:37",
    "tipoData": "0",
    "options": [
      "1",
      "2"
    ]
  },
  "listaOrari": [
    {
      "time": "10:45",
      "run": "MILANO, Comasina M3",
      "line": "705",
      "transport": "3"
    }
  ],
  "listaFermate": [
    {
      "id": "1",
      "stop": "Cormano, V. Zara",
      "x": "9.153879",
      "y": "45.549848"
    }
  ],
  "listaErrori": []
}
```

4.3.1.4 Codici d'errore

Il servizio “timetables” può ritornare i seguenti errori:

| Codice | Descrizione |
|--------|--|
| ER011 | Formato campo date non valido: inserire una data nel formato dd/mm/aaaa |
| ER012 | Formato campo when non valido: inserire un orario nel formato hh:mm |
| ER016 | Formato campo tipoData non valido. Valori ammissibili: 0, 1 |
| ER018 | Nessuna fermata trovata |
| ER019 | Fermata ambigua |
| ER020 | Formato della stringa del campo stop non valida: il numero di caratteri deve essere compreso tra 2 e 100 |

4.3.2 Metodo POST

Il metodo viene utilizzato per ricercare gli orari utilizzando il body per indicare i criteri di ricerca.

- METODO: POST
- ENDPOINT: Url Base + /tplapi/v1.0.0/timetables
- HEADER:
 - Content-Type: application/json
 - Authorization: Bearer Token_Di_Accesso

4.3.2.1 Parametri di richiesta

Il JSON conterrà i seguenti attributi:

- “lang”: Lingua di riferimento per la restituzione dei messaggi di errore eventualmente generati. Stringa. Opzionale. Se null, si assume il valore di default. Valori ammissibili:
 - it=italiano (default)
 - en=inglese
 - fr=francese
 - de=tedesco
- “stop”: Descrizione della fermata di cui si ricercano gli orari. Stringa con lunghezza massima di 100 caratteri. Obbligatorio.
- “date”: Data a cui è riferita la ricerca. Data in formato dd/mm/yyyy. Obbligatorio.
- “when”: Ora a cui è riferita la ricerca. Orario in formato hh:mm. Obbligatorio.
- “options”: Mezzi coinvolti nella ricerca. Array di numeri. Obbligatorio. Valori ammissibili:
 - 1=Treno
 - 2=Metro
 - 3=Bus
 - 4=Tram
 - 5=Battello
 - 6=Altro (funivia)
- “tipoData”: Indica se i parametri “date” e “when” si riferiscono alla partenza o all’arrivo. Intero. Opzionale. Se null, si assume il valore di default. Valori ammissibili:
 - 0=data e orario si riferiscono alla partenza (default)
 - 1=data e orario si riferiscono all’arrivo

4.3.2.2 Campi della risposta

Il JSON di risposta conterrà, i seguenti elementi:

| Nome campo | Descrizione | Tipo di dato | Mandatorietà | Valori ammissibili |
|------------|--|---------------|--------------|--|
| stato | codice di stato della risposta | Numero intero | Obbligatorio | <ul style="list-style-type: none"> • -1=fault (valorizzato oggetto “errori”) • 0=risolutivo (valorizzato oggetto “lista Percorsi”) |
| lang | Lingua di riferimento per la restituzione dei messaggi di errore | Stringa | Obbligatorio | <ul style="list-style-type: none"> • it=italiano (default) • en=inglese • fr=francese |

| | | | | |
|---------------------|---|-------------------------|--------------|---|
| | eventualmente generati | | | <ul style="list-style-type: none"> de=tedesco |
| <i>richiesta</i> | <i>Oggetto contenente l'elenco dei parametri di ricerca della request</i> | <i>Oggetto</i> | | <i>I campi opzionali della request sono sempre restituiti, eventualmente vuoti.</i> |
| <i>listaOrari</i> | <i>Oggetto contenente l'elenco degli orari delle corse in passaggio alla fermata</i> | <i>Oggetto</i> | | |
| time | Orario di partenza della corsa | Orario in formato hh:mm | Obbligatorio | |
| run | Descrizione del capolinea della corsa | Stringa 100 Char | Obbligatorio | <ul style="list-style-type: none"> capolinea di arrivo della corsa nel caso in cui si sia scelto tipoData=0 (ricerca di corse per data e ora di partenza dalla fermata) capolinea di partenza della corsa nel caso in cui si sia scelto tipoData=1 (ricerca di corse per data e ora di arrivo alla fermata) |
| line | Codice della linea a cui si riferisce l'orario | Stringa 6 Char | Obbligatorio | |
| transport | Identificativo del tipo di mezzo che effettua la corsa | Array di numeri | Obbligatorio | <ul style="list-style-type: none"> 1=Treno 2=Metro 3=Bus 4=Tram 5=Battello 6=Altro (funivia) |
| <i>listaFermate</i> | <i>Oggetto contenente Informazioni relative alla fermata impostata nella ricerca oppure elenco delle possibili fermate nel caso in cui la descrizione della fermata impostata nella ricerca non abbia prodotto un risultato univoco</i> | <i>Oggetto</i> | | |
| Id | Identificativo progressivo della fermata | Intero | Obbligatorio | |
| Stop | Descrizione della fermata di cui si ricercano gli orari | Stringa (100) | Obbligatorio | |

| | | | | |
|-------------|--|---|---|--|
| x | Coordinata x della fermata espressa nel sistema di riferimento nel sistema WGS84 | Numero compreso tra -180 e +180, separatore “.” | Obbligatorio | |
| y | Coordinata y della fermata espressa nel sistema di riferimento nel sistema WGS84 | Numero compreso tra -90 e +90, separatore “.” | Obbligatorio | |
| errori | Oggetto contenente la lista di eventuali errori | Oggetto | Nel caso in cui non vengano trovati errori, nel Json l’array degli errori sarà vuoto. | |
| codice | Codice dell’errore | Stringa | | |
| descrizione | Descrizione dell’errore | Stringa | | |

4.3.2.3 Esempio di risposta

Si fornisce un esempio di JSON di risposta restituito in caso di successo:

```
{
  "stato": "0",
  "lang": "it",
  "richiesta": {
    "date": "25/03/2021",
    "stop": "Cormano, V. Zara",
    "when": "10:37",
    "tipoData": "0",
    "options": [
      "1",
      "2",
      "3",
      "4",
      "5",
      "6"
    ]
  },
  "listaOrari": [
    {
      "time": "10:45",
      "run": "MILANO, Comasina M3",
      "line": "705",
      "transport": "3"
    }
  ],
  "listaFermate": [
    {
      "id": "1",
      "stop": "Cormano, V. Zara",
      "x": "9.153879",
      "y": "45.549848"
    }
  ],
  "listaErrori": []
}
```

4.3.2.4 Codici d'errore

Il servizio “timetables” può ritornare i seguenti errori:

| Codice | Descrizione |
|--------|--|
| ER011 | Formato campo date non valido: inserire una data nel formato dd/mm/aaaa |
| ER012 | Formato campo when non valido: inserire un orario nel formato hh:mm |
| ER016 | Formato campo tipoData non valido. Valori ammissibili: 0, 1 |
| ER018 | Nessuna fermata trovata |
| ER019 | Fermata ambigua |
| ER020 | Formato della stringa del campo stop non valida: il numero di caratteri deve essere compreso tra 2 e 100 |

4.4 Servizio “timetablesValidity”

Questo servizio può essere utilizzato per ricercare la validità degli orari delle aziende che forniscono i dati a “Muoversi in Lombardia”.

A tale scopo è stato messo a disposizione il metodo GET sull’endpoint “/timetablesValidity” con l’obiettivo di garantire il recupero della lista di tutte le validità.

4.4.1 Metodo GET

Il metodo viene utilizzato per ricercare le validità degli orari:

- METODO: GET
- ENDPOINT: Url Base + “/tplapi/v1.0.0/timetablesValidity”
- HEADER:
 - Content-Type: application/json
 - Authorization: Bearer Token_Di_Accesso

4.4.1.2 Campi della risposta

Il JSON di risposta conterrà, i seguenti elementi:

| Nome campo | Descrizione | Tipo di dato | Mandatorietà | Valori ammissibili |
|------------|--|----------------------------|--------------|---|
| company | Denominazione dell’azienda | Stringa 500 Char | Obbligatorio | <i>Nel caso in cui non vengano trovati orari validi, nel Json l’array sarà vuoto.</i> |
| startDate | Data di inizio del periodo di validità degli orari | Data in formato dd/mm/yyyy | Obbligatorio | |
| endDate | Data di fine del periodo di validità degli orari | Data in formato dd/mm/yyyy | Obbligatorio | |
| lastUpdate | Data di ultimo aggiornamento degli orari | Data in formato dd/mm/yyyy | Obbligatorio | |

4.4.1.3 Esempio di risposta

Si fornisce un esempio di JSON di risposta restituito in caso di successo:

```
[
  {
    "company": "Trenitalia",
    "startDate": "01/09/2020",
    "endDate": "30/06/2021", "lastUpdate": "09/09/2020"
  }
]
```

5. Uso dei codici di stato http

È importante utilizzare i codici di stato http, in quanto questi possono essere letti anche da utenti non umani.

| Code (codice) | Code Summary | Descrizione |
|------------------------|------------------------|---|
| Successful code | | |
| 200 | OK | La richiesta è andata a buon fine. |
| 201 | Created | Indica che la richiesta HTTP ha portato alla creazione di una risorsa. |
| Error code | | |
| 400 | Bad request | Il server non può elaborare la richiesta a causa di un errore del client (e.g. sintassi della richiesta non corretta: formato JSON non valido). |
| 401 | Unauthorized | Possibili due significati: a) L'autenticazione è stata richiesta e non è andata a buon fine; b) L'autenticazione non è ancora stata fornita. |
| 403 | Forbidden | Nessuna risorsa corrispondente trovata nell'API per la richiesta specificata. |
| 415 | Unsupported Media Type | L'entità richiesta ha un tipo di supporto che il server o la risorsa non supporta. Di seguito sono riportati i tipi di contenuto supportati: application/json, application/x-www-form-urlencoded e multipart/form-data. Se si invia un tipo di contenuto diverso dal tipo di contenuto supportato, viene visualizzato questo codice di stato. |
| 429 | Too many requests | Soglie di limitazione superate. L'intestazione http Retry-After specifica il numero di secondi di attesa prima di riprendere le chiamate API. |
| 500 | Internal server error | Si è verificato un errore del server. |
| 501 | Not implemented | Quando il server non riconosce il metodo di richiesta o non è in grado di soddisfare la richiesta. |
| 503 | Server unavailable | Risorse del server superate, servizio temporaneamente non disponibile. |