

[ГХ] 7.2 Поиск рейсов через Яндекс.Расписания с указанием фильтров и отображением результатов поиска

Оглавление

- Общее описание
  - Известные ошибки, особенности и ограничения
  - Макеты UI
- Сценарий работы
  - Роли пользователей
  - Системы
  - Внешние системы
  - Входные данные
  - Предусловия
  - Алгоритм работы
  - Результат
  - Диаграмма UML
  - Альтернативные сценарии и ошибки
- Маппинг данных UI + API TWM
- Логирование

Общее описание

На экране “Поиск билетов” пользователь вводит параметры поиска, нажимает кнопку “Искать” и получает список рейсов, соответствующих заданным критериям.

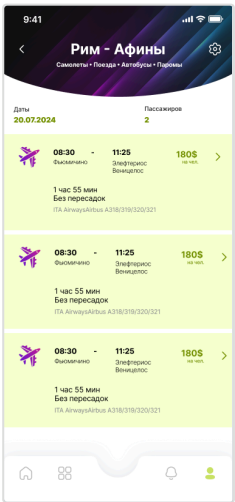
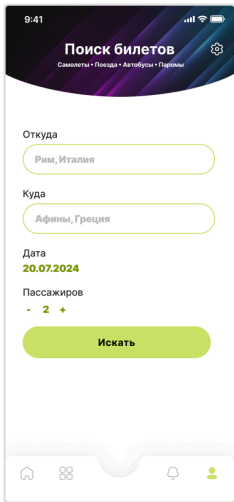
Известные ошибки, особенности и ограничения

Дата поездки в формате ДД.ММ.ГГГГ, выбирается через встроенный календарь. Минимальная дата поиска - текущий день. Максимальная дата поиска - +1 год от текущей даты.

Количество пассажиров по умолчанию - 1.

Максимальное количество пассажиров - 10.

Макеты UI



## Сценарий работы

### Роли пользователей

- Пользователь соц. сети TWM

### Системы

- iOS TWM / Android TWM
- Сервер-приложение TWM (Backend)
- БД

### Внешние системы

- Яндекс.Расписания


### Входные данные

- Откуда (пункт отправления)
- Куда (пункт назначения)
- Дата поездки
- Количество пассажиров

### Предусловия

- Пользователь соц. сети авторизован в приложении TWM
- Пользователь соц. сети находится в разделе “Поиск билетов”

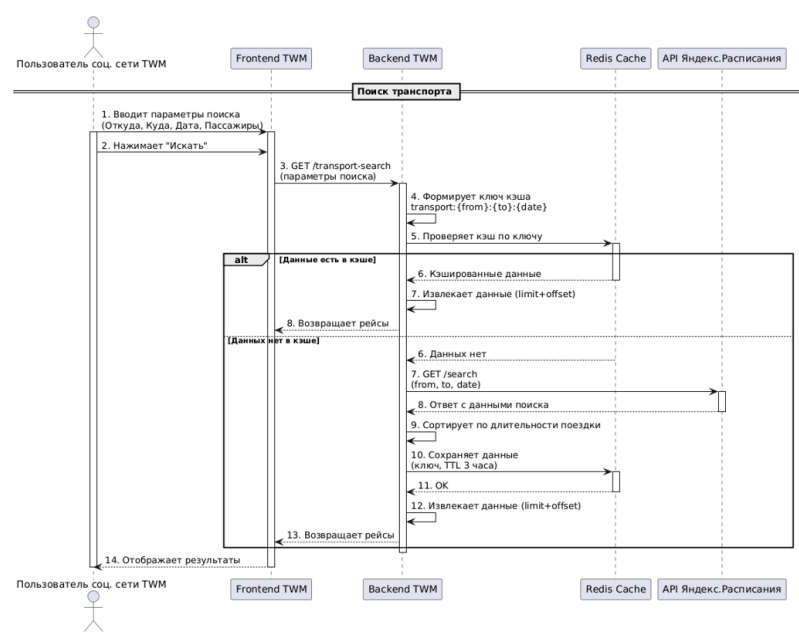
### Алгоритм работы

1. **Пользователь соц. сети** вводит параметры поиска:
  - Откуда
  - Куда
  - Дата поездки
  - Количество пассажиров
2. **Пользователь соц. сети** нажимает кнопку “Искать”
3. **Frontend TWM** отправляет запрос на поиск рейсов GET /transport-search на **Backend TWM**  
[GX] 7.2.1 GET /transport-search - Поиск рейсов на транспорт в Яндекс.Расписания
4. **Frontend TWM** отображает информацию о рейсах на экране

### Результат

Пользователь получает список рейсов на транспорт, соответствующий заданным критериям.

Диаграмма UML



Альтернативные сценарии и ошибки

2A. Если ошибка соединения с интернет:

- Отобразить текст предупреждения наверху экрана “Нет соединения. Проверьте подключение к интернету и попробуйте снова.”
- Запрос на поиск рейсов не отправляется

3A. Если ошибка подключения к серверу:

- Отобразить текст предупреждения наверху экрана “Сервер временно недоступен. Повторите попытку позже.”
- Запрос на поиск рейсов не отправляется

Маппинг данных UI + API TWM

Экран “Поиск билетов”

На UI	В API TWM (URL/JSON)	Комментарий
Поле “Откуда”	from	Пункт отправления
Поле “Куда”	to	Пункт назначения
Параметр “Дата”	date	Дата поездки
Параметр “Пассажиров”	passengers	Количество пассажиров
-	limit	Максимальное количество результатов поиска в ответе
-	offset	Смещение относительно

	первого результата поиска
--	------------------------------

## Логирование

Отдельные требования не предъявляются. Использовать общие по системе.

# [ГХ] 7.2.1 GET /transport-search - Поиск рейсов на транспорт в Яндекс.Расписания

## Оглавление

- Общее описание
  - Документация внешней системы
- Сценарий работы (Use Case)
  - Системы
  - Входные данные
  - Предусловия
  - Алгоритм работы
  - Результат
  - Альтернативные сценарии и обработка ошибок
- Техническая реализация метода
  - Запрос
  - Ответ
    - Успешный ответ
    - Обработка ошибок
- Маппинг данных
- Логирование

## Общее описание


Метод позволяет получить список рейсов, следующих от пункта отправления к пункту прибытия, и информацию по каждому рейсу на выбранную дату из Яндекс.Расписания.

## Документация внешней системы

 [Расписание рейсов между станциями](#) |

## Особенности и ограничения

Количество запросов в сутки зависит от тарифа подключения к API Яндекс.Расписания. На бесплатном тарифе - до 500 запросов в сутки.

Пункты отправления и прибытия в запросе должны быть указаны в системе кодирования Яндекс.Расписания: 

Параметр limit можно не передавать, значение по умолчанию - 100.

Параметр offset можно не передавать, значение по умолчанию - 0.

## Сценарий работы (Use Case)

### Системы

- Backend TWM
- API Яндекс.Расписания
- БД
- Redis

### Входные данные

- from
- to
- date
- passengers
- limit
- offset

### Предусловия

- Пользователь соц. сети вводит параметры поиска на странице “Поиск билетов” и нажимает кнопку “Искать”
- Frontend TWM отправляет запрос на поиск рейсов GET /transport-search на Backend TWM

### Алгоритм работы

1. **Backend TWM** получает запрос на поиск рейсов GET /transport-search
2. Backend TWM формирует уникальный ключ кэша Redis `transport:{from}:{to}:{date}`, например, `transport:Москва:Санкт-Петербург:2025-06-15`
3. **Backend TWM** проверяет кэш:
  - a. Если данные есть, то извлекает запрошенное количество рейсов (limit), начиная с указанного смещения (offset), или только доступные, если данных меньше, чем `offset + limit` и возвращает результат на **Frontend TWM**
  - b. Если данных нет, то переходит к следующему шагу
4. **Backend TWM** вызывает метод **Яндекс.Расписания** для поиска рейсов GET /search

Пример запроса:

```
1 curl 'https://api.rasp.yandex.net/v3.0/search/?from=c50&to=c43&date=2025-06-30&transfers=true&
```

## 5. Яндекс.Расписания возвращает ответ с данными поиска

Пример ответа:

✓ Пример тела ответа

```
1 {
2   "search": {
3     "from": {
4       "type": "settlement",
5       "title": "Пермь",
6       "short_title": "Пермь",
7       "popular_title": "Пермь",
8       "code": "c50"
9     },
10    "to": {
11      "type": "settlement",
12      "title": "Казань",
13      "short_title": "Казань",
14      "popular_title": "Казань",
15      "code": "c43"
16    },
17    "date": "2025-06-30"
18  },
19  "segments": [
20    {
21      "thread": {
22        "number": "",
23        "title": "Пермь, автовокзал — Чебоксары",
24        "short_title": "Пермь, автовокзал — Чебоксары",
25        "express_type": null,
26        "transport_type": "bus",
27        "carrier": {
28          "code": 61126,
29          "title": "000 \"Транспортная компания \"Якорь\"",
30          "codes": {
31            "sirena": null,
32            "iata": null,
33            "icao": null
34          },
35          "address": "г.Чебоксары, проезд Машиностроителей, 1 \"к\"",
36          "url": "http://tkyakor.ru/",
37          "email": "transport_yakor@mail.ru",
38          "contacts": "+7 (8352) 26-66-66, +7 (902) 663-10-49, +7 (919) 972-45-",
39          "phone": "+7 (8352) 26-66-66, +7 (902) 663-10-49, +7 (919) 972-45-35",
40          "logo": null,
41          "logo_svg": null
42        },
43        "uid": "empty_1_f9635690t9636024_48",
44        "vehicle": "МАН 48/0",
45        "transport_subtype": {
46          "title": null,
47          "code": null,
48          "color": null
49        },
50        "thread_method_link": "api.rasp.yandex.net/v3/thread/?date=2025-06-30&uid"
```

```

51     },
52     "stops": "",
53     "from": {
54         "type": "station",
55         "title": "Пермь, автовокзал",
56         "short_title": null,
57         "popular_title": null,
58         "code": "s9635690",
59         "station_type": "bus_station",
60         "station_type_name": "автовокзал",
61         "transport_type": "bus"
62     },
63     "to": {
64         "type": "station",
65         "title": "Казань, Центральный автовокзал",
66         "short_title": null,
67         "popular_title": "Центральный автовокзал",
68         "code": "s9623344",
69         "station_type": "bus_station",
70         "station_type_name": "автовокзал",
71         "transport_type": "bus"
72     },
73     "departure_platform": "",
74     "arrival_platform": "",
75     "departure_terminal": null,
76     "arrival_terminal": null,
77     "duration": 46140.0,
78     "has_transfers": false,
79     "tickets_info": {
80         "et_marker": false,
81         "places": [
82             {
83                 "name": null,
84                 "price": {
85                     "whole": 2500,
86                     "cents": 0
87                 },
88                 "currency": "RUB"
89             }
90         ]
91     },
92     "departure": "2025-06-30T16:20:00+05:00",
93     "arrival": "2025-07-01T03:09:00+03:00",
94     "start_date": "2025-06-30"
95 },
96 {
97     "thread": {
98         "number": "325E",
99         "title": "Пермь — Новороссийск",
100         "short_title": "Пермь — Новороссийск",
101         "express_type": null,
102         "transport_type": "train",
103         "carrier": {
104             "code": 112,
105             "title": "РЖД/ФПК",
106             "codes": {
107                 "sirena": null,
108                 "iata": null,

```



```

109         "icao": null
110     },
111     "address": "Москва, ул. Новая Басманная , д. 2",
112     "url": "http://www.rzd.ru/",
113     "email": "info@rzd.ru",
114     "contacts": "Единая телефонная линия: +7 (800) 775-00-00 (звонок беспл
115     "phone": "+7 (800) 775-00-00",
116     "logo": "https://yastat.net/s3/rasp/media/data/company/logo/logo.gif"
117     "logo_svg": null
118 },
119 "uid": "325E_3_2",
120 "vehicle": null,
121 "transport_subtype": {
122     "title": null,
123     "code": null,
124     "color": null
125 },
126 "thread_method_link": "api.rasp.yandex.net/v3/thread/?date=2025-06-30&uid
127 },
128 "stops": "",
129 "from": {
130     "type": "station",
131     "title": "Пермь-2",
132     "short_title": "",
133     "popular_title": "",
134     "code": "s9607774",
135     "station_type": "station",
136     "station_type_name": "станция",
137     "transport_type": "train"
138 },
139 "to": {
140     "type": "station",
141     "title": "Казань-2 (Восстание-Пасс.)",
142     "short_title": "",
143     "popular_title": "",
144     "code": "s9612620",
145     "station_type": "train_station",
146     "station_type_name": "вокзал",
147     "transport_type": "train"
148 },
149 "departure_platform": "",
150 "arrival_platform": "",
151 "departure_terminal": null,
152 "arrival_terminal": null,
153 "duration": 53760.0,
154 "has_transfers": false,
155 "tickets_info": {
156     "et_marker": false,
157     "places": []
158 },
159 "departure": "2025-06-30T21:56:00+05:00",
160 "arrival": "2025-07-01T10:52:00+03:00",
161 "start_date": "2025-06-30"
162 },
163 {
164     "departure_from": {
165         "type": "station",
166         "title": "Большое Савино",

```

```

167         "short_title": null,
168         "popular_title": null,
169         "code": "s9600381",
170         "station_type": "airport",
171         "station_type_name": "аэропорт",
172         "transport_type": "plane"
173     },
174     "arrival_to": {
175         "type": "station",
176         "title": "Казань",
177         "short_title": null,
178         "popular_title": null,
179         "code": "s9600379",
180         "station_type": "airport",
181         "station_type_name": "аэропорт",
182         "transport_type": "plane"
183     },
184     "transport_types": [
185         "plane",
186         "plane"
187     ],
188     "departure": "2025-06-30T06:25:00+05:00",
189     "arrival": "2025-06-30T10:50:00+03:00",
190     "transfers": [
191         {
192             "type": "settlement",
193             "title": "Москва",
194             "short_title": "Москва",
195             "popular_title": "Москва",
196             "code": "c213"
197         }
198     ],
199     "has_transfers": true,
200     "details": [
201         {
202             "thread": {
203                 "number": "SU 1395",
204                 "title": "Пермь – Москва",
205                 "short_title": "Пермь – Москва",
206                 "carrier": {
207                     "code": 26,
208                     "title": "Аэрофлот",
209                     "codes": {
210                         "sirena": "CY",
211                         "iata": "SU",
212                         "icao": "AFL"
213                     }
214                 },
215                 "vehicle": "Airbus A320",
216                 "express_type": null,
217                 "transport_type": "plane",
218                 "transport_subtype": {
219                     "title": null,
220                     "code": null,
221                     "color": null
222                 },
223                 "uid": "SU-1395_250630_c26_12"
224             },

```

```

225         "from": {
226             "type": "settlement",
227             "title": "Пермь",
228             "short_title": "Пермь",
229             "popular_title": "Пермь",
230             "code": "c50"
231         },
232         "to": {
233             "type": "settlement",
234             "title": "Москва",
235             "short_title": "Москва",
236             "popular_title": "Москва",
237             "code": "c213"
238         },
239         "departure_platform": "",
240         "arrival_platform": "",
241         "departure_terminal": "",
242         "arrival_terminal": "B",
243         "stops": "",
244         "duration": 8100.0,
245         "start_date": "2025-06-30",
246         "departure": "2025-06-30T06:25:00+05:00",
247         "arrival": "2025-06-30T06:40:00+03:00"
248     },
249     {
250         "is_transfer": true,
251         "duration": 9000.0,
252         "transfer_point": {
253             "type": "settlement",
254             "title": "Москва",
255             "short_title": "Москва",
256             "popular_title": "Москва",
257             "code": "c213"
258         },
259         "transfer_from": {
260             "type": "station",
261             "title": "Шереметьево",
262             "short_title": null,
263             "popular_title": null,
264             "code": "s9600213",
265             "station_type": "airport",
266             "station_type_name": "аэропорт",
267             "transport_type": "plane"
268         },
269         "transfer_to": {
270             "type": "station",
271             "title": "Шереметьево",
272             "short_title": null,
273             "popular_title": null,
274             "code": "s9600213",
275             "station_type": "airport",
276             "station_type_name": "аэропорт",
277             "transport_type": "plane"
278         }
279     },
280     {
281         "thread": {
282             "number": "FV 6199",

```

```

283         "title": "Москва — Казань",
284         "short_title": "Москва — Казань",
285         "carrier": {
286             "code": 8565,
287             "title": "Россия",
288             "codes": {
289                 "sirena": "ФВ",
290                 "iata": "FV",
291                 "icao": null
292             }
293         },
294         "vehicle": "Сухой Суперджет 100",
295         "express_type": null,
296         "transport_type": "plane",
297         "transport_subtype": {
298             "title": null,
299             "code": null,
300             "color": null
301         },
302         "uid": "FV-6199_250630_c8565_12"
303     },
304     "from": {
305         "type": "settlement",
306         "title": "Москва",
307         "short_title": "Москва",
308         "popular_title": "Москва",
309         "code": "c213"
310     },
311     "to": {
312         "type": "settlement",
313         "title": "Казань",
314         "short_title": "Казань",
315         "popular_title": "Казань",
316         "code": "c43"
317     },
318     "departure_platform": "",
319     "arrival_platform": "",
320     "departure_terminal": "B",
321     "arrival_terminal": "1",
322     "stops": "",
323     "duration": 6000.0,
324     "start_date": "2025-06-30",
325     "departure": "2025-06-30T09:10:00+03:00",
326     "arrival": "2025-06-30T10:50:00+03:00"
327 }
328 ]
329 }
330 ],
331 "pagination": {
332     "total": 12,
333     "limit": 100,
334     "offset": 0
335 }
336 }

```

6. **Backend TWM** сохраняет данные в кэше Redis:

- ключ `transport:{from}:{to}:{date}` , например,  
`transport:Москва:Санкт-Петербург:2025-06-15`
- TTL: 3 часа
- сортировка данных по длительности поездки (от меньшей к большей)

✓ Пример сохранённых данных в кэше (JSON)

```
1  {
2    "search": {
3      "fromTitle": "Пермь",
4      "toTitle": "Казань",
5      "date": "2025-06-30"
6    },
7    "transports": [
8      {
9        "transportType": "bus",
10       "carrierTitle": "000 \\"Транспортная компания \\"Якорь\\"",
11       "vehicle": "МАН 48/0",
12       "fromTitle": "Пермь, автовокзал",
13       "toTitle": "Казань, Центральный автовокзал",
14       "duration": 46140.0,
15       "hasTransfers": false,
16       "ticketPrice": {
17         "whole": 2500,
18         "cents": 0,
19         "currency": "RUB"
20       },
21       "departure": "2025-06-30T16:20:00+05:00",
22       "arrival": "2025-07-01T03:09:00+03:00"
23     },
24     {
25       "transportType": "train",
26       "carrierTitle": "РЖД/ФПК",
27       "vehicle": "null",
28       "fromTitle": "Пермь-2",
29       "toTitle": "Казань-2 (Восстание-Пасс.)",
30       "duration": 53760.0,
31       "hasTransfers": false,
32       "ticketPrice": {
33         "whole": null,
34         "cents": null,
35         "currency": "null"
36       },
37       "departure": "2025-06-30T21:56:00+05:00",
38       "arrival": "2025-07-01T10:52:00+03:00"
39     },
40     {
41       "transportType": "plane",
42       "carrierTitle": "Аэрофлот",
43       "vehicle": "Airbus A320",
44       "fromTitle": "Пермь",
45       "toTitle": "Москва",
46       "duration": 8100.0,
47       "hasTransfers": true,
48       "ticketPrice": {
49         "whole": null,
```

```

50         "cents": null,
51         "currency": "null"
52     },
53     "departure": "2025-06-30T06:25:00+05:00",
54     "arrival": "2025-06-30T06:40:00+03:00"
55 }
56 ]
57 }

```

7. **Backend TWM** возвращает на **Frontend TWM** запрошенное количество рейсов (limit), начиная с указанного смещения (offset), или только доступные, если данных меньше, чем offset + limit

## Результат

Frontend TWM получает информацию о рейсах от метода GET /transport-search

## Альтернативные сценарии и обработка ошибок

1A. Если запрос не содержит обязательных параметров (from, to, date):

- Вернуть ошибку “Не указаны обязательные параметры: [список отсутствующих параметров].” (HTTP 400)
- Зафиксировать событие в журнале ошибок

2A. Если соединение с Redis недоступно:

- Повторить попытку соединения (макс. 3 попытки)
- При повторной ошибке:
  - Пропустить шаг и перейти к вызову API Яндекс.Расписания
  - Зафиксировать событие в журнале ошибок
  - Отправить уведомление в систему мониторинга

3A. Если при проверке в Redis возникает ошибка:

- Повторить операцию (макс. 3 попытки)
- При повторной ошибке:
  - Пропустить шаг и перейти к вызову API Яндекс.Расписания
  - Зафиксировать событие в журнале ошибок
  - Отправить уведомление в систему мониторинга

4A. Если не удалось вызвать метод Яндекс.Расписания:

- Повторить попытку вызова метода (макс. 3 попытки)
- При повторной ошибке:

- Вернуть ошибку “Внутренняя ошибка сервера.” (HTTP 500)
- Зафиксировать событие в журнале ошибок
- Отправить уведомление в систему мониторинга

5A. Если API Яндекс.Расписания возвращает ошибку:

- Повторить запрос (макс. 3 попытки)
- При повторной ошибке:
  - Вернуть существующие данные из кэша или сообщение: “Не удалось получить актуальные рейсы на транспорт. Повторите попытку позже.” (502)
  - Зафиксировать событие в журнале ошибок с деталями запроса/ответа
  - Отправить уведомление в систему мониторинга

5B. Если API Яндекс.Расписания не возвращает данные:

- Вернуть пустой список рейсов (HTTP 200)
- Зафиксировать событие в журнале предупреждений

6A. Если не удалось отсортировать данные по длительности:

- Вернуть данные без сортировки (HTTP 200)
- Зафиксировать событие в журнале ошибок
- Отправить уведомление в систему мониторинга

7A. Если при возврате результата на Frontend возникает ошибка:

- Повторить попытку (макс. 3 попытки)
- При повторной ошибке:
  - Вернуть ошибку “Внутренняя ошибка сервера.” (HTTP 500)
  - Зафиксировать событие в журнале ошибок
  - Отправить уведомление в систему мониторинга

## Техническая реализация метода

### Запрос

**Тип метода:** GET

**URL:** <https://api.tripwithme.com/public/v1.0/transport-search>

**Авторизация:** Basic Auth

**Headers:**

- Authorization

### Query params:

- from
- to
- date
- passengers
- limit
- offset

**Body:** -

### Пример запроса:

```
1 curl 'https://api.tripwithme.com/public/v1.0/transport-search?from=Кострома&to=Вологда&date=2025-06-30'
2 -H 'Authorization: Basic XXXXX' \
3 -H 'Accept: application/json'
```

### Ответ

Успешный ответ

**HTTP 200**

### Пример тела ответа:

✓ Пример тела ответа

```
1 {
2   "search": {
3     "fromTitle": "Пермь",
4     "toTitle": "Казань",
5     "date": "2025-06-30"
6   },
7   "trips": [
8     {
9       "transportType": "bus",
10      "carrierTitle": "000 \\"Транспортная компания \\"Якорь\\"",
11      "vehicle": "МАН 48/0",
12      "fromTitle": "Пермь, автовокзал",
13      "toTitle": "Казань, Центральный автовокзал",
14      "duration": 46140.0,
15      "hasTransfers": false,
16      "ticketPrice": {
17        "whole": 2500,
18        "cents": 0,
19        "currency": "RUB"
20      },
21      "departure": "2025-06-30T16:20:00+05:00",
22      "arrival": "2025-07-01T03:09:00+03:00"
23    },
24    {
```



```

25     "transportType": "train",
26     "carrierTitle": "РЖД/ФПК",
27     "vehicle": "null",
28     "fromTitle": "Пермь-2",
29     "toTitle": "Казань-2 (Восстание-Пасс.)",
30     "duration": 53760.0,
31     "hasTransfers": false,
32     "ticketPrice": {
33         "whole": null,
34         "cents": null,
35         "currency": "null"
36     },
37     "departure": "2025-06-30T21:56:00+05:00",
38     "arrival": "2025-07-01T10:52:00+03:00"
39 },
40 {
41     "transportType": "plane",
42     "carrierTitle": "Аэрофлот",
43     "vehicle": "Airbus A320",
44     "fromTitle": "Пермь",
45     "toTitle": "Москва",
46     "duration": 8100.0,
47     "hasTransfers": true,
48     "ticketPrice": {
49         "whole": null,
50         "cents": null,
51         "currency": "null"
52     },
53     "departure": "2025-06-30T06:25:00+05:00",
54     "arrival": "2025-06-30T06:40:00+03:00"
55 }
56 ]
57 "pagination": {
58     "total": 10,
59     "limit": 10,
60     "offset": 0
61 }
62 }

```

## Обработка ошибок

Причина возникновения ошибки указана в теле ответа, в параметре `error.description`.

## HTTP 400

Возвращается, если:

- в запросе не переданы обязательные параметры данные
- в запросе переданы некорректные данные (неверный формат, недопустимое значение)

## Пример тела ответа:

```

1 {
2     "error": {

```

```
3     "id": "2f9f3767-0016-5000-b000-17ef8394c2cb",
4     "description": "Не указан параметр from."
5   }
6 }
```

## HTTP 404

Возвращается, если пункта, по которому производится поиск, не существует

### Пример тела ответа:

```
1 {
2   "error": {
3     "id": "2f9f3767-0016-5000-b000-17ef8394c2cb",
4     "description": "Пункт не найден."
5   }
6 }
```

## HTTP 500

Возвращается, если возникла ошибка в работе Backend TWM

### Пример тела ответа:

```
1 {
2   "error": {
3     "id": "2f9f3767-0016-5000-b000-17ef8394c2cb",
4     "description": "Внутренняя ошибка сервера."
5   }
6 }
```

## HTTP 502

Возвращается, если Яндекс.Расписания возвращает ошибку

### Пример тела ответа:

```
1 {
2   "error": {
3     "id": "2f9f3767-0016-5000-b000-17ef8394c2cb",
4     "description": "Извините, произошла непредвиденная ошибка. Повторите попытку позже."
5   }
6 }
```

## Маппинг данных

Для запроса:

В URL TWM	В URL Яндекс.Расписания	Комментарий
from	from	Пункт отправления

to	to	Пункт назначения
date	date	Дата поездки
passengers	-	Количество пассажиров
-	transfers	Признак, разрешающий добавить к результатам поиска рейсы с пересадками
limit	limit	Максимальное количество результатов поиска в ответе
offset	offset	Смещение относительно первого результата поиска

**Для ответа:**

<b>В JSON Яндекс.Расписания</b>	<b>В JSON TWM</b>	<b>Комментарий</b>
search.from.title	search.fromTitle	Название пункта отправления поиска
search.to.title	search.toTitle	Название пункта назначения поиска
search.date	search.date	Дата, на которую получен список рейсов

segments[].thread.transport_type	transport[].transportType	Тип транспорта
segments[].thread.carrier.title	transport[].carrierTitle	Название перевозчика
segments[].thread.vehicle	transport[].vehicle	Название транспортного средства
segments[].from.title	transport[].fromTitle	Название пункта отправления рейса
segments[].to.title	transport[].toTitle	Название пункта назначения рейса
segments[].duration	transport[].duration	Длительность поездки в секундах
segments[].has_transfers	transport[].hasTransfers	Признак наличия пересадок по ходу рейса
segments[].tickets_info.places[].price.whole	transport[].ticketPrice.whole	Цена (рубли)
segments[].tickets_info.places[].price.cents	transport[].ticketPrice.cents	Цена (копейки)
segments[].tickets_info.places[].currency	transport[].ticketPrice.currency	Валюта
segments[].departure	transport[].departure	Время отправления
segments[].arrival	transport[].arrival	Время прибытия

## Логирование

Отдельные требования не предъявляются. Использовать общие по системе.