

Описание апи для интеграции внешних сервисов для оплаты парковки.

Єдиний паркувальний простір

Version 1 20.01.2022

Вступление	1
Описание методов	2
1. Оплата парковки	2
Порядок отправки данных	4
2. Информация о статусе парковочной сессии	5
3. Информация о всех парковках оператора	7
4. Информация об измененных парковках оператора	10
5. Расчет стоимости парковочной сессии	12
6. Шаблоны для покупки новых абонементов	15
7. Сохранение оплаты за абонемент	18

Вступление

В данном документе описан механизму взаимодействия и обмена данными между внешним сервером получения оплат за выполненные парковочные сессии и сервисом Єдиного паркувального простору (ЄПП). Сервис ЄПП позволяет интеграцию с внешними системами при помощи REST API для получения и агрегирования информации относительно парковочных площадок, совершенных парковок и выполненных оплат, предоставления информации о стоимости парковки на выбранном парковочном пространстве. Используемый формат обмена данными – JSON.

Описание методов

Базовая ссылка

- тестовое апи - <https://test.parkovka.app/partners>
- продуктивное апи - <https://parkovka.app/partners>

Для авторизации используется ключ апи(API_KEY)

Таблица 1. Список методов апи

№	Описание	Относительная ссылка
1	Оплата парковки	POST /api/v2/parkingmeter/park POST /api/v2/privat/park
2	Информация о статусе парковочной сессии	POST /api/v2/parkingmeter/transactionstate
3	Информация о всех парковках оператора	POST /api/v2/parkingmeter/allparkings
4	Информация об измененных парковках оператора	POST /api/v2/parkingmeter/modifiedparkings
5	Расчет стоимости парковочной сессии	POST /api/v2/parkingsessions/calculatecost
6	Шаблоны для покупки новых абонементов	POST /api/v2/seasontickets/sptemplates
7	Сохранение оплаты за абонемент	POST /api/v2/privat/seasonticket

1. Оплата парковки

Интерфейс предназначен для передачи информации относительно выполненной оплаты за паркование через стороннее приложение. После получение данных об оплате от приложения сторонний платежный сервер передает информацию о сессии в систему ЕПП путем отправки POST – запроса на адресов:

- при использовании паркомата: [/api/v2/parkingmeter/park](#)
- при оплате не через паркомат: [/api/v2/privat/park](#)

Таблица 2. Структура запроса для оплаты парковки

Название поля	Тип	Описание
timestamp *	String	Текущее время в UTC в формате yyyy-dd-MM HH:mm:ss
data *	String	Base64(Json(DataModel))
sign *	String	MD5 hex значение строки timestamp+API_KEY+data
lang	String	Локализация: en/ru/uk. Необязательное к заполнению

* - обязательное поле

Таблица 3. Структура модели данных(DataModel) для этой операции

Название поля	Тип	Обязательное	Описание
parking	String	Required if parkingMeterId null	Код парковки (например, A1000, G1001) из реестра парковочных площадок оператора паркования
parkingMeterId	String	Required if parking null	ID паркомата через который была осуществлена оплата
carlicense	String	Required	Гос. номер автомобиля
amount	Decimal	Required	Сумма, уплаченная за парковочную сессию. До 2х знаков после запятой
paymenttype	Int	Optional	Код формы оплаты через паркомат: 0 = Наличные 1 = Безналичные
duration	Int	Required	Длительность сессии в минутах. Целое число.
currency	String	Optional	Код валюты. “UAH” всегда по умолчанию для Украины. Может быть null .
cardType	String	Optional	Код метода оплаты. Возможные значения: Mastercard Visa Other
cardNumber	String	Optional	Зашифрованный номер карты, по которому можно определить тип карты. Одно из двух полей (cardType, cardNumber) должно быть заполнено.
time	String	Required	Время старта сессии для запросов с опозданием. Строка – время UTC в формате уууу-dd-MM HH:mm:ss
Idsea	String	Optional	Уникальный идентификатор транзакции на стороне поставщика сессии

Пример объекта DataModel:

```
{
  "parkometer": "38383933333210",
  "carlicense": "AA7825HH",
  "amount": 10.0,
  "currency": "UAH",
  "duration": 60
}
```

Таблица 4. Ответ

Название поля	Тип	Описание
errorCode	Int	Код ошибки. Значение 0 – успешно выполненная операция. Значение, отличное от 0, сигнализирует о возникновении ошибки при обработке данных, значение зависит от каждого запроса.
errorDescription	String	Детализация ошибки: строка с описанием деталей ошибки в понимаемом человеку виде. NULL, если errorCode = 0
additionalInformation	String	Опциональное поле. Дополнительная информация об ошибке. Индивидуальное значение в разных операциях
data	Object	Опционально. Возвращаемый на клиент объект. Используется преимущественно в GET-запросах. Индивидуальное значение в разных операциях
requestId	String	Опционально. Уникальный идентификатор запроса для отладки.

Пример ответа для запроса, в котором неправильно высчитан Sign:

```
{
  "errorCode": 1,
  "errorDescription": "Sign is not equal to MD5(TimeStamp+API_KEY+Data)",
  "additionalInformation": null,
  "data": null
}
```

Таблица 5. Возможные коды ошибок (errorCode):

Код	Описание
0	Операция прошла успешно.
1	Ошибка при проверке подписи. Детали ошибки в поле errorDescription
2	Ошибка в DataModel. Детали ошибки в поле errorDescription
4080	Ошибка сервера (Exception) при выполнении операции

Порядок отправки данных

При получении новой оплаты за парковку Оборудование отправляет данные о Транзакции на Сервер. Сервер подготавливает эти данные в такой способ:

1. Отмечается время создания транзакции, которое будет использоваться во всех дальнейших шагах. Это строка в формате **YYYY-dd-mm HH:MM:SS** , далее будет упоминаться как timestamp (например, **2022-15-01 12:00:00**).
2. Подготавливается сам объект транзакции:

```
{
  "parkingMeterId": "000001", // идентификатор Оборудования
  "carlicense": "AA0000AA", // гос. номер авто
  "amount": 10.0, // оплаченная сумма в гривнах
  "duration": 60, // оплаченное время в минутах
  "paymenttype": 0, // тип оплаты (0 - наличные, 1 - карта)
  "cardType": "Other", // тип карты, указывается только если оплата проведена картой
  "time": "2022-15-01 12:00:00", // timestamp
}
```

3. Формируется base64-строка в кодировке UTF-8 на основе JSON-представления объекта транзакции, далее будет упоминаться как `b64_string`. На примере объекта транзакции выше она будет такой:
- `eyJwYXJraW5nTWV0ZXJJZCI6IClwMDAwMDEiLCAiY2FybG1jZW5zZSI6ICJBQTAwMDEBQSI6ICJhbW91bnQiOiAxMC4wLCAiZHVyYXRpb24iOiA2MCwgInBheWw1bnR0eXB1IjogMCwgInRpbWUiOiAiMjAyMi0xNS0wMSAxMjowMDowMCJ9`
4. Формируется подпись данных (далее упоминается как `sign`) в 2 шага: `timestamp`, `token` и `b64_string` конкатенируются в одну строку, на примере данных выше и `API_KEY=PARTNER` результат будет такой: `2022-15-01 12:00:00PARTNEReyJwYXJraW5nTWV0ZXJJZCI6IClwMDAwMDEiLCAiY2FybG1jZW5zZSI6ICJBQTAwMDEBQSI6ICJhbW91bnQiOiAxMC4wLCAiZHVyYXRpb24`

Конкатенация строк кодируется алгоритмом `md5` и отдается в hex-формате (на примере данных выше результат: `04fc42e5a99c0ac915ac0b66ec3206a6`)

5. Формируется итоговый JSON вида:

```
{
  "TimeStamp": timestamp, // timestamp с шага 1
  "data": b64_string, // b64_string с шага 3
  "sign": sign, // sign с шага 4
  "lang": "en", // отметка о языке запроса, не меняется
}
```

2. Информация о статусе парковочной сессии

Интерфейс предназначен для проверки статуса обработки переданной парковочной сессии, если при передаче самой сессии не было получено ответа от сервера ЕПП и необходимо получить информацию требуется ли повторная передача данных.

Таблица 6. Параметры запроса данных

Название	Тип		Описание
Sign	String	Required	значение API_KEY
TransactionId	String	Required	Идентификатор транзакции, сформированный внешней платежной системой при оплате парковочной сессии

Пример запроса:

```
{
  "TransactionId": "31252e21-fff8-483e-be00-29ff815307a4",
  "Sign": "8JMGLVI3YEV2GUCXZJ43LFN086V0AKJHVDJPJ267AAMQXNJXIMGPZB626VIPAR305NGQ"
}
```

Таблица 7. Параметры ответа:

Название поля	Тип	Описание
ErrorCode	Int	Код ошибки. Значение 0 – успешно выполненная операция. Значение, отличающееся от 0, сигнализирует о возникновении ошибки при обработке данных.
ErrorDescription	String	Детализация ошибки: строка с описанием деталей ошибки. NULL, если errorCode = 0
Data	Object	Объект данных с информацией о транзакции
RequestId	String	Идентификатор запроса

Модель данных по каждой парковке для этой операции должна иметь следующую структуру:

Название	Тип		Описание
transactionState	Int	Required	Статус обработки парковочной сессии по данной транзакции: 0 – сессия валидна, успешно принята и обработана 1 – сессия получена, но содержит ошибки, требует повторной передачи данных 2 – сессия не найдена, информация об оплате не была получена
errorDescription	String	Optional	Описание ошибки, возникшей при получении парковочной сессии, передается только в случае если transactionState = 1

Пример ответа для transactionState = 0

```
{
  "ErrorCode":0,
  "ErrorDescription":null,
  "AdditionalInformation":null,
  "Data": {
    "transactionState":0,
    "errorDescription":null
  },
  "RequestId":"942a8025-c418-412f-b6bc-cd54d4a26387"
}
```

Пример ответа для transactionState = 1

```
{
  "ErrorCode":0,
  "ErrorDescription":null,
  "AdditionalInformation":null,
  "Data": {
    "transactionState":1,
    "errorDescription":"Паркомат з таким Id не існує в системі"},
  "RequestId":"7796477f-e484-4107-907a-d840dae8c8e4"
}
```

Пример ответа для transactionState = 2

```
{
  "ErrorCode":0,
  "ErrorDescription":null,
  "AdditionalInformation":null,
  "Data": {
    "transactionState":2,
    "errorDescription":null
  },
  "RequestId":"37cc4b83-301f-488b-bcea-da63909f046d"
}
```

3. Информация о всех парковках оператора

/api/v2/parkingmeter/allparkings

Интерфейс предназначен для синхронизации идентификаторов парковок между ПЗ СЭА и ЕПП и возвращает полный перечень парковочных площадок оператора.

Таблица 8. Модель запроса

Название поля	Тип		Описание
Sign	String	Required	значение API_KEY
ServiceProviderId	Int	Required	Идентификатор оператора паркования
ParkingPlaceKind	Int	Required	Вид парковки. Значения: • 0 - All types • 1 – Paid / Платная • 2 – Free / Бесплатная • 3 – Info / Информационная
Id	GUID	Optional	Идентификатор запроса
Language	String	Optional	Код локали: en/ru/uk. Служебный параметр для совместимости. Необязательное к заполнению
UserAgent	String	Optional	Код устройства. Служебный параметр для совместимости. Необязательное к заполнению.
CultureName	String	Optional	Служебный параметр для совместимости. Необязательное к заполнению.

Ответ

Название поля	Тип	Описание
ErrorCode	Int	Код ошибки. Значение 0 – успешно выполненная операция. Значение, отличающееся от 0, сигнализирует о возникновении ошибки при обработке данных, значение зависит от каждого запроса.
ErrorDescription	String	Детализация ошибки: строка с описанием деталей ошибки в понимаемом человеку виде. NULL, если errorCode = 0
AdditionalInformation	String	Опциональное поле. Дополнительная информация об ошибке. Индивидуальное значение в разных операциях
Data	Object	Опционально. Возвращаемый на клиент объект. Используется преимущественно в GET-запросах. Индивидуальное значение в разных операциях
RequestId	GUID	Опционально. Идентификатор запроса

Модель данных по каждой парковки для этой операции должна иметь следующую структуру:

Название поля	Тип		Описание
ParkingPlaceId	Int	Required	Идентификатор парковки
ParkingmeterId	List	Required	ID паркомата/паркоматов, привязанных к парковке. Может быть пустым.
ParkingPlaceKind	Int	Required	Вид парковки. Значения: • 1 – Paid / Платная • 2 – Free / Бесплатная • 3 – Info / Информационная
Addresses	String	Optional	Адрес парковки
ZoneName	String	Optional	Название зоны, в которую включена парковка. Если значение NULL – парковка не включена в зону.

Пример успешного ответа

```
{
  "ErrorCode":0,
  "ErrorDescription":null,
  "AdditionalInformation":null,
  "Data": [
    {
      "ParkingPlaceId": "1022"
      "ParkingmeterId": []
      "Address": "address 1"
      "ZoneName": null
    },
    {
      "ParkingPlaceId": "1023"
      "ParkingmeterId": []
      "Address": "address 2"
      "ZoneName": null
    },
  ],
  "RequestId":"37cc4b83-301f-488b-bcea-da63909f046d"
}
```

Таблица 9. Возможные коды ошибок (errorCode)

Код	Описание
0	Success
1	NoData
2	NoSuchServiceProvider
3	ApiKeyIncorrectOrNotFound

4. Информация об измененных парковках оператора

/api/v2/parkingmeter/modifiedparkings

Возвращает только те данные, по которым были изменения за последние 24 часа.

Таблица 10. Запрос

Название поля	Тип		Описание
Sign	String	Required	значение API_KEY
ServiceProviderId	Int	Required	Идентификатор оператора паркования
ParkingPlaceKind	Int	Required	Вид парковки. Значения: • 0 - All types • 1 – Paid / Платная • 2 – Free / Бесплатная • 3 – Info / Информационная
Id	GUID	Optional	Идентификатор запроса
Language	String	Optional	Код локали: en/ru/uk. Служебный параметр для совместимости. Необязательное к заполнению
UserAgent	String	Optional	Код устройства. Служебный параметр для совместимости. Необязательное к заполнению.
CultureName	String	Optional	Служебный параметр для совместимости. Необязательное к заполнению.

Ответ:

Название поля	Тип	Описание
ErrorCode	Int	Код ошибки. Значение 0 – успешно выполненная операция. Значение, отличающееся от 0, сигнализирует о возникновении ошибки при обработке данных, значение зависит от каждого запроса.
ErrorDescription	String	Детализация ошибки: строка с описанием деталей ошибки в понимаемом человеку виде. NULL, если errorCode = 0
AdditionalInformation	String	Опциональное поле. Дополнительная информация об ошибке. Индивидуальное значение в разных операциях
Data	Object	Опционально. Возвращаемый на клиент объект. Используется преимущественно в GET-запросах. Индивидуальное значение в разных операциях
RequestId	GUID	Опционально. Идентификатор запроса

Модель данных по каждой парковке для этой операции должна иметь следующую структуру:

Название поля	Тип		Описание
ParkingPlaceId	Int	Required	Идентификатор парковки
ParkingmeterId	List	Required	ID паркомата/паркоматов, привязанных к парковке. Может быть пустым.
ParkingPlaceKind	Int	Required	Вид парковки. Значения: • 1 – Paid / Платная • 2 – Free / Бесплатная • 3 – Info / Информационная
Addresses	String	Optional	Адрес парковки
ZoneName	String	Optional	Название зоны, в которую включена парковка. Если значение NULL – парковка не включена в зону.

```
{
  "ErrorCode":0,
  "ErrorDescription":null,
  "AdditionalInformation":null,
  "Data": [
    {
      "ParkingPlaceId": "1022"
      "ParkingmeterId": []
      "Address": "address 1"
      "ZoneName": null
    },
    {
      "ParkingPlaceId": "1023"
      "ParkingmeterId": []
      "Address": "address 2"
      "ZoneName": null
    },
  ],
  "RequestId":"37cc4b83-301f-488b-bcea-da63909f046d"
}
```

Таблица 11. Возможные коды ошибок (errorCode)

Код	Описание
0	Success
1	NoData
2	NoSuchServiceProvider
3	ApiKeyIncorrectOrNotFound

5. Расчет стоимости парковочной сессии

Интерфейс предназначен для передачи информации относительно стоимости паркования на выбранном парковочном пространстве. Для получения информации о стоимости парковки внешний сервер направляет запрос в систему ЕПП путем отправки POST – запроса на адрес

<api/v2/parkingsessions/calculatecost>

Таблица 12. Модель запроса

Название поля	Тип		Описание
parkingMeterId	String	Optional	ID паркомата, через который будет осуществлена оплата

Название поля	Тип		Описание
ParkingPlaceId	String	Required	Код парковки (например, A1000, G1001) из реестра парковочных площадок оператора паркования.
ParkingDurationMinutes	Int	Required	Продолжительность парковки в минутах
StartTimeUtc	Int	Required	Текущее клиентское время (в формате UTC Unix TimeStamp)
PaymentModel	Int	Required	Код используемой модели оплаты
VehicleType	Int	Required	Тип транспортного средства
Id	String	Optional	Идентификатор запроса
Language	String	Optional	Язык приложения

Пример запроса

```
{
  "ParkingPlaceId":27,
  "ParkingDurationMinutes":1200,
  "StartTimeUtc":1503051411,
  "PaymentModel": 0, /*0 or null - use balance model, 1 - use transactional model*/
  "VehicleType":0,
  "Id":null,
  "Language":null
}
```

В ответ на данный запрос сервер ЕПП вернет следующие данные:

Название поля	Тип	Описание
errorCode	Int	Код ошибки. Значение 0 – успешно выполненная операция. Значение, отличающееся от 0, сигнализирует о возникновении ошибки при обработке данных, значение зависит от каждого запроса.
errorDescription	String	Детализация ошибки: строка с описанием деталей ошибки. NULL, если errorCode = 0

Название поля	Тип	Описание
additionalInformation	String	Опциональное поле. Дополнительная информация об ошибке. Индивидуальное значение в разных операциях
data	Object	Информация о стоимости.
requestId	String	Опционально. Уникальный идентификатор запроса для отладки.

Модель данных для этой операции должна иметь следующую структуру:

Название поля	Тип		Описание
time	Int	Required	Текущее время (в формате UTC Unix TimeStamp)
parking	String	Required	Код парковки (например, A1000, G1001) из реестра парковочных площадок оператора паркования.
parkingPlaceId	Int	Required	Идентификатор парковки
duration	Int	Required	Продолжительность парковки
amount	Decimal	Required	Стоимость парковки
currencyCode	Int	Optional	Код валюты, в которой рассчитана стоимость
currencyName	String	Required	Название валюты, в которой рассчитана стоимость
serviceProviderId	String	Required	Идентификатор оператора парковки
isVirtual	String	Optional	Не используется

Пример ответа

```
{
  "errorCode":0,
  "errorDescription":null,
  "additionalInformation":null,
  "data":{
    "time":1492181664,
    "parkingPlace":"A1027",
    "parkingPlaceId":27,
    "duration":60,
    "amount":0.0,
    "currencyCode":840,
    "serviceProviderId":5,
    "currencyName":"USD",
    "isVirtual":false
  }
  "requestId":"392e1958-3335-4b68-ba1d-633510cb80f7"
}
```

6. Шаблоны для покупки новых абонементов

Интерфейс предназначен для получения условий продажи абонементов.

</api/v2/seasontickets/sptemplates>

Таблица 13. Запрос

Название поля	Тип		Описание
Id	GUID	R	Идентификатор запроса
ServiceProviderId	Int	R	Идентификатор оператора паркования, информацию о доступных условиях продажи абонементов (шаблонах абонементов) необходимо получить
Language	String	O	Доступные коды локали: en/ru/uk. Служебный параметр для совместимости. Необязательное к заполнению
UserAgent	String	O	Код устройства. Служебный параметр для совместимости. Необязательное к заполнению.

Пример запроса:

```
{
  "ServiceProviderId":1,
  "Id":null,
  "Language":"uk",
  "UserAgent":null,
  "CultureName":null
}
```

Таблица 14. Ответ

Название поля	Тип	Описание
ErrorCode	Int	Код ошибки. Значение 0 – успешно выполненная операция. Значение, отличающееся от 0, сигнализирует о возникновении ошибки при обработке данных.
ErrorDescription	String	Детализация ошибки: строка с описанием деталей ошибки. NULL, если errorCode = 0
Templates	Object	Массив данных для поиска шаблона для покупки абонемента
RequestId	GUID	Опционально. Идентификатор запроса

Модель данных по каждому типу условий продажи абонемента (шаблону) должна иметь следующую структуру

Название поля	Тип	Описание
TemplateId	Int	Уникальный идентификатор условий продажи абонементов.
Name	String	Название условий продажи абонементов, локализовано
Description	String	Описание условий продажи абонементов, локализовано
Zones	Object	Массив зон паркования, заданных в шаблоне (для которых будет действовать абонемент)
ParkingPlaces	Object	Массив площадок для паркования, заданных в шаблоне явно

Название поля	Тип	Описание
Period	Int	Тип периода действия абонемента. <i>Возможные значения</i> • Дневной = 1, • Месяц с пользовательской даты = 2, • Календарный месяц = 3, • Календарный квартал = 4, • Квартал с пользовательской даты = 5, • Календарные полгода = 6, • Полгода с пользовательской даты = 7, • Календарный год = 8, • Год с пользовательской даты = 9
Periods	Object	Дополнительные массивы, уточняющие период действия абонемента. Только для period = 8 или 6.
Validity Type	Int	Тип ограничений действия абонемента по времени суток: <i>Доступные значения</i> • без ограничений, круглосуточный = 0 • ночной = 1 • дневной = 2 • периоды дня = 3
Validity Days	Object	Дополнительный массив, уточняющий в какие дни недели действует абонемент. Только для ValidityType = 0
Validity Time	Object	Дополнительный массив, уточняющий в какое время суток действует абонемент. Только для ValidityType = 1, 2, 3
Parking Condition	Int	Условия действия абонемента. Доступные значения • Бесплатная парковка = 1, • Дисконт на парковку (фиксированный процент от стандартного тарифа) = 2.
CostOr Discount	Int	Процент скидки на парковку, который будет действовать при покупке абонемента. Актуально для parkingCondition = 2
TicketCost	Decimal	Стоимость абонемента по данным условиям. Без учета акций

Название поля	Тип	Описание
Currency	String	Валюта, в которой должен быть оплачен абонемент

zones – массив парковочных тарифных зон, для которых возможно приобрести абонемент

Название поля	Тип	Описание
id	Int	Уникальный идентификатор тарифной зоны.
name	String	Название тарифной зоны, не локализовано

parkingPlaces – массив площадок для паркования, для которых возможно приобрести абонемент, только те парковки, которые заданы в шаблоне. .

Название поля	Тип	Описание
id	Int	Уникальный идентификатор площадки для паркования.
name	String	Код площадки для паркования, не локализовано

Periods – уточнение периода, на который можно приобрести полугодовой или годовой абонемент

Название поля	Тип	Описание
year	String	Порядковый номер города (например, 2020, 2021).
halfYear	Int	Порядковый номер полугодия (1, 2)

ValidityTime – уточнение временного диапазона действия абонемента в течении суток

Название поля	Тип	Описание
Start	String	Время начала действия абонемента
End	String	Время окончания действия абонемента

7. Сохранение оплаты за абонемент

Интерфейс предназначен для передачи в систему ЕПП информации относительно приобретенных через внешние сервисы абонементов на паркование. Данные передаются путем отправки POST – запроса на адрес:

</api/v2/privat/seasonticket>

Таблица 15. Структура запроса

Название поля	Тип		Описание
timestamp	String	Required	Текущее время в UTC в формате yyyy-dd-MM HH:mm:ss

Название поля	Тип		Описание
data	String	Required	Base64(Json(DataModel))
sign	String	Required	MD5(timestamp+API_KEY+data)
lang	String	Optional	Код локализации: en/ru/uk. Необязательное к заполнению

Таблица 16. Структура модели данных(DataModel)

Название поля	Тип		Описание
templateId	Int	Required	Идентификатор условий продажи абонеента
name	String	Optional	Название условий продажи абонеента
description	String	Optional	Описание условий продажи абонеента
ticketNumber	String	Required	Номер (серийный) абонеента, уникальное значение в системе
vehicleType	Int	Optional	Тип транспортного средства. Для легковых автомобилей: 0
vehicleLicensePlate	String	Required	Номер транспортного средства, для которого приобретен абонемент
PhoneNumber	String	Required	Номер телефона пользователя, который совершил покупку абонеента
ClientName	String	Optional	Имя, фамилия водителя
dateStart	Int	Required	Дата начала действия абонеента (в формате UTC Unix TimeStamp)
dateEnd	Int	Required	Дата окончания действия абонеента (в формате UTC Unix TimeStamp)
ticketCost	Decimal	Required	Стоимость абонеента, оплаченная водителем
TransactionId	String	Required	Идентификатор финансовой транзакции по оплате абонеента
CardType	String	Optional	Тип карты, который был оплачен абонемент: Visa, MasterPass
CardNumber	String	Optional	Номер карты в экранированном виде

Таблица 17. Структура ответа

Название поля	Тип	Описание
errorCode	Int	Код ошибки. 0 – успешно выполненная операция. Значение, отличающееся от 0, сигнализирует о возникновении ошибки при обработке данных.
errorDescription	String	Детализация ошибки: строка с описанием деталей ошибки. NULL, если errorCode = 0

Название поля	Тип	Описание
additionalInformation	String	Опциональное поле. Дополнительная информация об ошибке. Индивидуальное значение в разных операциях
requestId	String	Опционально. Уникальный идентификатор запроса для отладки.