

[AO] Задача на REST API: Получение информации о заказе на курс [GET /v1/orders/{orderId}]




Проект платформы для дополнительного образования от GetAnalyst


Автор: Антонина Охтина

antoninaohtina@gmail.com

+7 (909) 950-15-96

 Ссылка на Jira

 Связанные документы (внутренние)

 [A Free Database Designer for Developers and Analysts](#)

- БД U-Verse

 [\[AO\] Таблица БД - Заказ_](#)

[В РАБОТЕ](#) - описание

таблицы БД `order`

Оглавление:

- Общее описание
 - Доступы и ограничения
- Запрос
 - Пример запроса
 - Тело запроса (Body)
 - Пример ответа - Успех
 - Требования к обработке ошибок
- Требования к реализации
- Маппинг данных
- Логирование и мониторинг

Общее описание

Метод предназначен для получения полной информации о конкретном заказе на курс на платформе U-Verse по его уникальному идентификатору (id).

Метод предназначен для использования в приложениях U-Verse:

- для пользователей-Студентов:
 - отображение подробной информации о сделанном заказе на курс;
 - первичное заполнение формы заявки на отказ от прохождения курса;
- для Администратора системы:
 - просмотр информации о заказе на курс.

Доступы и ограничения

Метод должен быть доступен для авторизованных пользователей:

- пользователя-Студента (если заказ принадлежит пользователю)
- Администратора

Запрос

Method type: GET

BaseURL: <https://api.uverse.com>

API name:

- public - для пользователей-Студентов
- admin - для Администраторов

API version: v1

Method name: /orders/{orderId}

Authorization:

- Тип: Bearer Token
- Требуется всегда
- Роли: Студент (которому принадлежит заказ), Администратор

Headers:

- Используются общие для системы.

Query-parameters:

- Нет.

Пример запроса

Запрос:

```
1 GET https://api.uverse.com/public/v1/orders/{orderId}
```

```
1 GET https://api.uverse.com/admin/v1/orders/{orderId}
```

Пример:

```
1 GET https://api.uverse.com/public/v1/orders/737f33c7-cd10-4f0f-93e7-7b93
```

Тело запроса (Body)

Нет.

Пример ответа - Успех

Код HTTP: 200 OK

Тело ответа (Body):

```
1 {
2   "id": "60fba13a-fd7e-4f14-a056-c4618b04ba4d",
3   "number": "16273-25",
4   "status": "Оплачен",
5   "createdAt": "2025-08-24T11:36:15.137Z",
6   "updatedAt": "2025-08-30T11:54:48.004Z",
7   "accessDate": "2025-09-15T11:54:48.004Z",
8   "student": {
9     "id": "97cac9ef-1b23-4532-861f-e7b8655d1601",
10    "name": "Максим",
11    "patronymic": "Александрович",
12    "surname": "Галкин",
13    "phone": "79097771234",
14    "email": "email12345@gmail.com"
15  },
16  "flow": {
17    "id": "ec32e393-5af1-4941-98c2-b363a45d2fcb",
18    "title": "2025-09",
19    "type": "По расписанию",
20    "startDate": "2025-09-15",
21    "endDate": "2026-06-15",
22    "course": {
23      "id": "0a61cc69-dbb6-4704-9941-0d4d80708717",
24      "title": "Системный анализ",
25      "accessTerminationDate": "2027-06-15",
26      "coverUrl": "https://files.u-verse.com/covers/c32e69e0-2525-4064-",
27      "lessonsCount": 90,
28      "completedLessons": 18
29    }
30  },
31  "terms": {
32    "id": "2e659286-8637-4f0e-8dd3-710b0d5b7d06",
33    "type": "Общие",
34    "tariff": "Базовый (с проверкой домашних заданий)",
35    "basePrice": 166500.00,
36    "currency": "RUB",
37    "discountPrice": 144000.00,
38    "installment": false,
39    "installmentDuration": 0,
40    "monthlyPayment": 0.00
41  },
42  "payment": {
43    "paymentsCount": 1,
44    "paymentsAmount": 144000.00,
45    "currency": "RUB",
46    "firstPaymentDate": "2025-08-30",
47    "paymentType": "bankCard",
48    "costs": 5360.00
49  },
50  "admin": {
51    "id": "5c95df11-1e65-4856-b366-334f4bd3ed7b",
52    "name": "Ольга",
53    "surname": "Владимирова"
54  },
55  "refusal": {
56    "id": "fe908133-1e43-485f-b45d-aba92d476760",
57    "status": {
58      "id": "85c520aa-fdac-4fc7-97fc-907f690080db",
59      "name": "На согласовании"
60    }
61  }
62 }
```

Требования к обработке ошибок

Каждый ответ - ошибка возвращает:

```
1 {  
2   "code": "2001",  
3   "message": "Пользователь не авторизован"  
4 }
```

Тело ответа (Body):

Описание возможных ошибок - см. [\[АО\] Коды ошибок_В РАБОТЕ](#)



Проект платформы для дополнительного
образования от GetAnalyst

Автор: Антонина Охтина

antoninaohtina@gmail.com

+7 (909) 950-15-96

Требования к реализации

Основной сценарий:

1. Аутентификация и авторизация.

Проверить заголовок `Authorization: Bearer <Token>`.

- Если заголовок отсутствует: вернуть ошибку **HTTP-401** + код 2001.
- Если токен имеет неверный формат: вернуть ошибку **HTTP-401** + код 2002.
- Если токен не соответствует ни одному из пользователей или устарел: вернуть ошибку **HTTP-401** + код 2003.
- Иначе - далее.

2. Получить на вход {orderId}.

3. Проверить в БД, что есть соответствующий заказ с указанным {orderId}:

- Если заказ не найден, то вернуть ошибку с кодом **HTTP-404** + код 3301.
- Иначе - далее.

4. Для пользователя-Студента (API name - `public`) проверить принадлежность запрашиваемого заказа пользователю:

- Если пользователь запрашивает данные по своему заказу (`order.student_id` соответствует ID пользователя, сделавшего запрос), то далее.
- Иначе - вернуть ошибку **HTTP-403** + код 2004.

5. Сформировать данные о заказе на основании данных таблицы `order` и связанных с ней.

Вернуть с кодом **HTTP-200** JSON.

Требования к соответствию JSON и БД приведены в разделе Маппинг данных.

6. Любая ошибка при обращении к БД - вернуть ошибку **HTTP-500** + код 5012.

Маппинг данных

Для запроса:

Параметр в пути запроса (path-параметр)	Описание / комментарий, требования к валидации при приеме на обработку	Обязательность	Тип данных	В БД
{orderId}	Уникальный идентификатор заказа, который запрашивает пользователь.	Да	string (uuid)	order.id

Для тела ответа:

Параметр в JSON	Тип данных в JSON	Обязательность	Описание / комментарий, требования к преобразованию после извлечения из БД или обработки	Поле(я) в БД
id	string (uuid)	Да	UUID заказа	order.id
number	string	Да	Номер заказа	order.number
status	string	Да	Статус заказа Определять через связку order.status_id → status.id	Данные из связанной таблицы status status.name
createdAt	string (timestamp)	Да	Дата создания заказа	order.created_at
updatedAt	string (timestamp)	Да	Дата обновления информации о заказе	order.updated_at
accessDate	string (timestamp)	Да	Дата предоставления доступа к материалам курса	order.access_date

<code>student</code>	<object>	Да	Данные Студента, оформившего заказ. Определять через связку <code>order.user_id → student.id</code>	Данные из связанной таблицы <code>student</code>
<code>student.id</code>	string (uuid)	Да	Идентификатор Студента, которому принадлежит заказ	<code>order.user_id → student.id</code>
<code>student.name</code>	string	Да	Имя Студента	<code>student.first_name</code>
<code>student.patronymic</code>	string	Да	Отчество Студента	<code>student.middle_name</code>
<code>student.surname</code>	string	Да	Фамилия Студента	<code>student.last_name</code>
<code>student.phone</code>	string	Да	Номер телефона Студента	<code>student.phone</code>
<code>student.email</code>	string	Да	Email Студента	<code>student.email</code>
<code>flow</code>	<object>	Да	Данные потока, на который оформлен заказ Определять через связку <code>order.flow_id → flow.id</code>	Данные из связанной таблицы <code>flow</code>
<code>flow.id</code>	string (uuid)	Да	Идентификатор потока, на который оформлен заказ	<code>order.flow_id → flow.id</code>
<code>flow.title</code>	string	Да	Название потока	<code>flow.title</code>
<code>flow.type</code>	string	Да	Тип потока	<code>flow.type</code>
<code>flow.startDate</code>	string (timestamp)	Нет	Дата старта потока Передавать, если поток проводится по расписанию, т.е. <code>flow.type = "По расписанию"</code>	<code>flow.start_date</code>
<code>flow.endDate</code>	string (timestamp)	Нет	Дата завершения потока Передавать, если поток проводится по расписанию,	<code>flow.end_date</code>

			т.е. <code>flow.type = "По расписанию"</code>	
<code>flow.course</code>	<object>	Да	Данные курса, к которому относится поток Определять через связку <code>order.flow_id → flow.id</code> , <code>flow.course_id → course.id</code>	Данные из связанных таблиц <code>flow</code> , <code>course</code> , <code>lesson</code> , <code>student_lesson</code>
<code>flow.course_id</code>	string (uuid)	Да	Идентификатор курса, на который оформлен заказ	<code>flow.course_id → course.id</code>
<code>flow.course.title</code>	string	Да	Название курса	<code>course.title</code>
<code>flow.course.accessTerminationDate</code>	string (timestamp)	Да	Дата прекращения доступа к курсу Расчетное значение: - если поток проводится по расписанию (<code>flow.type = 'По расписанию'</code>), то дату окончания потока увеличивать на период доступа к материалам курса; - иначе (т.е. <code>flow.type = 'Свободный'</code>), дату предоставления доступа к материалам курса увеличивать на срок обучения и период доступа к материалам курса	<code>flow.endDate + course.access_period</code> или <code>order.access_date + course.duration + course.access_period</code>
<code>flow.course.coverUrl</code>	string (url)	Нет	URL обложки курса Передавать, если в БД есть значение	<code>course.cover_url</code>
<code>flow.course.lessonsCount</code>	integer	Да	Количество всех уроков в курсе Определять через связку <code>order.flow_id =</code>	Количество <code>lesson.id</code>

			<pre>flow.id → flow.course_id = course.id → course.id = module.course_id → module.id = lesson.module_id</pre>	
<pre>flow.course.completeLessons</pre>	integer	Да	<p>Количество всех уроков в курсе, пройденных Студентом (т.е. <code>student_lesson.state = 'Просмотрен' </code>)</p> <p>Определять через связку</p> <pre>order.flow_id = flow.id → flow.course_id = course.id → course.id = module.course_id → module.id = lesson.module_id → lesson.id = student_lesson.lesson_id</pre> <p>и</p> <pre>order.student_id = student_id → student.id = student_lesson.student_id</pre>	Количество <code>student_lesson.id</code>
<pre>terms</pre>	<object>	Да	<p>Условия, на которых оформлен заказ</p> <p>Определять через связку</p> <pre>order.terms_id → terms.id</pre>	Данные из связанной таблицы <code>terms</code>
<pre>terms.id</pre>	string (uuid)	Да	Идентификатор условий оплаты, по которым оформлен заказ	<pre>order.terms_id → terms.id</pre>

<code>terms.type</code>	string	Да	Тип условий оплаты (общие или индивидуальные)	<code>terms.type</code>
<code>terms.tariff</code>	string	Да	Название тарифа курса, на который оформлен заказ	<code>terms.tariff</code>
<code>terms.basePrice</code>	integer	Да	Базовая цена курса в соответствии с условиями оплаты Значение перевести из копеек в рубли (в случае введения новой валюты, при переводе учитывать самое маленькое ее значение)	<code>terms.price</code>
<code>terms.currency</code>	string	Да	Валюта, в которой производится расчет за курс Трехбуквенный код в формате ISO 4217	<code>terms.currency</code>
<code>terms.discountPrice</code>	integer	Нет	Цена курса с учетом скидки Расчетное значение Выводить, если предоставлена скидка (т.е. <code>terms.discount_value</code> <> 0) Значение перевести из копеек в рубли (в случае введения новой валюты, при переводе учитывать самое маленькое ее значение)	<code>terms.price</code> - <code>terms.discount_value</code>
<code>terms.installment</code>	boolean	Да	Признак оплаты в рассрочку	<code>terms.installment</code>
<code>terms.installmentDuration</code>	integer	Нет	Период рассрочки, месяцы Выводить, если предусмотрена оплата в рассрочку (т.е. <code>terms.installment = true</code>)	<code>terms.duration</code>
<code>terms.monthlyPayment</code>	integer	Нет	Размер ежемесячного платежа при оплате в рассрочку Выводить, если предусмотрена оплата в рассрочку (т.е.	<code>terms.payment</code>

			<code>terms.installment = true)</code> Значение перевести из копеек в рубли (в случае введения новой валюты, при переводе учитывать самое маленькое ее значение)	
<code>payment</code>	<object>	Нет	Информация о фактических платежах Выводить, если по заказу существует хотя бы один успешный платеж (т.е. <code>payment.status_id = '235ed000-9ef8-4fd8-9ae7-4cbd8142d840'</code> (статус “succeeded”)) Определять через связку <code>order.id → payment.order_id</code>	Данные из связанных таблиц <code>payment</code>
<code>payment.paymentsCount</code>	integer	Да	Количество успешных платежей по заказу - со статусом “succeeded” (т.е. <code>payment.status_id = '235ed000-9ef8-4fd8-9ae7-4cbd8142d840'</code>)	Количество <code>payment.id</code>
<code>payment.paymentsAmount</code>	integer	Да	Сумма всех успешных платежей, осуществленных по заказу Значение перевести из копеек в рубли (в случае введения новой валюты, при переводе учитывать самое маленькое ее значение)	Сумма <code>payment.amount</code>

<code>payment.currency</code>	string	Да	Валюта платежа (одинакова для всех платежей по заказу, определяем по первому платежу) Трехбуквенный код в формате ISO 4217	<code>payment.currency</code>
<code>payment.firstPaymentDate</code>	string (timestamp)	Да	Дата первого (или единственного) успешного платежа по заказу	<code>payment.captured_at</code>
<code>payment.paymentType</code>	string	Да	Способ оплаты по заказу Должен быть одинаковым у всех успешных платежей. Если имеют место разные способы оплаты, выводить фиксированное значение <code>'not_defined'</code>	<code>payment.payment_method</code>
<code>payment.costs</code>	integer	Да	Расходы платформы, связанные с платежами по заказу Расчетное значение: сумма комиссии за платеж (= сумма платежа - сумма фактически поступившая) и комиссии за чек Значение перевести из копеек в рубли (в случае введения новой валюты, при переводе учитывать самое маленькое ее значение)	Сумма для всех платежей по заказу <code>(payment.amount - payment.income_amount) + payment.receipt_fee</code>
<code>admin</code>	<object>	Нет	Данные об Администраторе, создавшем заказ Передавать, если в заказе указан идентификатор Администратора Определять через связку <code>order.admin_id → employee.id</code>	Данные из связанной таблицы <code>employee</code>
<code>admin.id</code>	string (uuid)	Да	Идентификатор Администратора	<code>order.admin_id → employee.id</code>

<code>admin.name</code>	string	Да	Имя Администратора	<code>employee.first_name</code>
<code>admin.surname</code>	string	Да	Фамилия Администратора	<code>employee.last_name</code>
<code>refusal</code>	<object>	Нет	<p>Данные о заявке на отказ от прохождения курса, созданной к заказу</p> <p>Передавать, если существует заявка в статусе, отличном от “Отозвана” или “Отклонена” (т.е. <code>refund.status_id</code> <> '8aa04d59-a48e-480b-b4a3-06d45a0d842e' (“Отозвана”) или <code>refund.status_id</code> <> '7e92dcca-61aa-4cef-add7-282e7be30679' (“Отклонена”))</p> <p>Определять через связку <code>order.id</code> → <code>refund.order_id</code></p>	Данные из связанной таблицы <code>refund</code>
<code>refusal.id</code>	string (uuid)	Да	Идентификатор заявки на отказ	<code>order.id</code> → <code>refund.order_id</code>
<code>refusal.status</code>	<object>	Да	<p>Данные о статусе заявки на отказ</p> <p>Определять через связку <code>order.id</code> → <code>refund.order_id</code>, <code>refund.status_id</code> → <code>status.id</code></p>	Данные из связанной таблицы <code>status</code>
<code>refusal.status.id</code>	string (uuid)	Да	Идентификатор статуса	<code>refund.status_id</code> → <code>status.id</code>
<code>refusal.status.name</code>	string	Да	Название статуса	<code>status.name</code>

Логирование и мониторинг

Общие требования к логированию запросов REST API системы. Особенности нет.



Проект платформы для дополнительного
образования от GetAnalyst

Автор: Антонина Охтина

antoninaohtina@gmail.com

+7 (909) 950-15-96