

12/3/24, 11:17 утра



Ссылка OCC

Сгенерировано: 2024-12-03 11:17:04 GMT+0000

SAP Коммерция | 2205

Публичный

Оригинальное содержание:https://help.sap.com/docs/SAP_COMMERCE/e5d7cec9064f453b84235dc582b886da?locale=en-US&state=PRODUCTION&version=2205

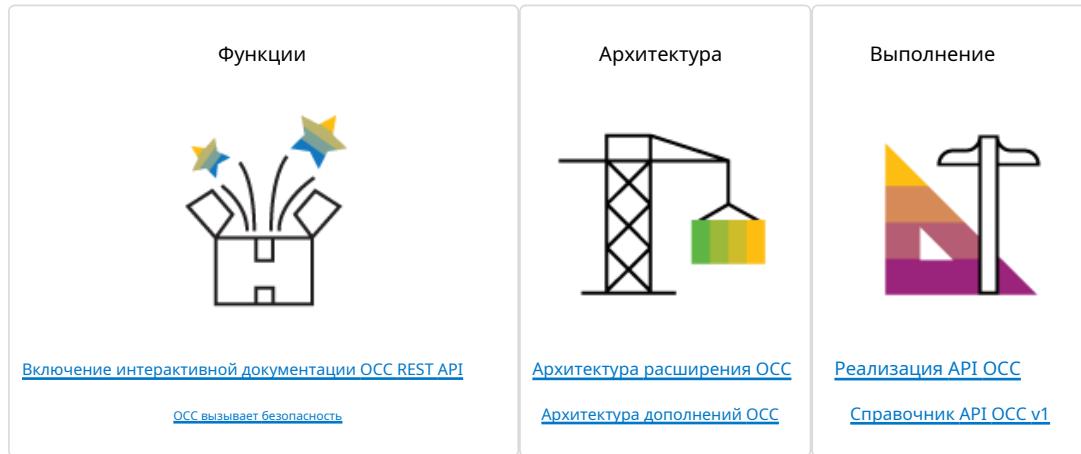
Предупреждение

Этот документ был создан на основе SAP Help Portal и является неполной версией официальной документации по продукту SAP. Информация, включенная в пользовательскую документацию, может не отражать расположение тем на SAP Help Portal и может не иметь важных аспектов и/или корреляций с другими темами. По этой причине он не предназначен для продуктивного использования.

Для получения более подробной информации посетите сайт<https://help.sap.com/docs/ отказ от ответственности>.

Omni Commerce Connect

API Omni Commerce Connect (OCC) предоставляет широкий набор услуг коммерции и данных. Он позволяет интегрировать функциональность SAP Commerce в любое место вашего прикладного ландшафта.



[Архитектура расширения OCC](#)

Расширения OCC позволяют расширить функциональность OCC без установки дополнений.

[Архитектура дополнений OCC](#)

OCC — это набор расширений, предоставляющих веб-сервисы RESTful, ориентированные на коммерцию. Дополнения OCC расширяют OCC новыми функциями. [OCC вызывает безопасность](#)

Безопасность вызовов OCC обеспечивается гибко настраиваемыми механизмами безопасности

[Spring. Реализация API OCC](#)

Получите общее представление о том, как реализован API SAP Commerce OCC, и узнайте, что вам нужно знать, чтобы расширить его с помощью собственной реализации API.

[Справочник API OCC v1](#)

Первая версия API OCC обеспечивает взаимодействие с функциональностью SAP Commerce с сохранением состояния.

Устаревший API OCC v1 продолжает поддерживаться, но не рекомендуется для новых разработок. Вместо этого используйте текущую и более полнофункциональную реализацию v2. Для получения более подробной информации см.[Реализация API OCC](#).

-Осторожность

Эта страница относится к программному обеспечению, которое было устарело как часть Accelerator UI и устаревших расширений шаблонов OCC. Для получения дополнительной информации см.
[Прекращение поддержки пользовательских интерфейсов Accelerator и старых расширений шаблонов OCC](#).

[Сравнение OCC v1 с OCC v2](#)

SAP Commerce OCC API предлагает как устаревшую реализацию v1 stateful, так и более новую, default stateless REST API. Получите полный обзор различий.

[Вызовы OCC v1 REST](#)

Получите обзор вызовов REST, доступных в версии 1 SAP Commerce OCC API.

[Образцы потоков OCC v1](#)

Примеры вызовов потоков для API SAP Commerce OCC в v1

[Ответы об ошибках от OCC v1](#)

Ошибкачные ответы, возвращаемые commerce сервисом расширение.

[Совместное использование ресурсов из разных источников](#)

Cross-Origin Resource Sharing (CORS) — это попытка W3C внедрить стандартный механизм для поддержки кросс-доменных запросов в веб-браузерах и на участвующих серверах.

[Обновление до расширений OCC](#)

Руководство по обновлению с версии v1 до реализации API OCC на основе расширений.

Сравнение OCC v1 с OCC v2

SAP Commerce OCC API предлагает как устаревшую реализацию v1 stateful, так и более новую, default stateless REST API. Получите полный обзор различий.

Обзор

SAP Commerce OCC API имеет две доступные реализации. OCC v2 обеспечивает настраиваемую и не сохраняющую состояния реализацию веб-сервисов RESTful, поддерживая будущий рост в среде headless commerce. Однако устаревшая реализация v1 по-прежнему поддерживается для существующих клиентов, использующих эту версию. Документацию по версии два по умолчанию см.[Реализация API OCC](#).

Версия 2 (по умолчанию)

Основные особенности версии 2 следующие:

| Особенность | Описание |
|--------------------|--|
| Без гражданства | <ul style="list-style-type: none"> Вызовы теперь не имеют состояния, поэтому данные о клиентах больше не сохраняются между последующими запросами. Каждый раз клиенту необходимо предоставить все необходимые данные, такие как идентификатор клиента или идентификатор корзины. |
| Реализация RESTful | <ul style="list-style-type: none"> URL-ресурсы были переработаны и теперь более соответствуют принципам REST. Заказчику необходимо указать тип ресурса и идентификатор ресурса, с которым он хотел бы работать. |
| Создание данных | <ul style="list-style-type: none"> Выполняется с использованием списка параметров URL или RequestBody. |

Версия 1

Основные особенности версии 1 следующие:

| Особенность | Описание |
|-------------------------|--|
| С сохранением состояния | <ul style="list-style-type: none"> Способ связи с коммерческим уровнем с учетом состояния Потоки, в некоторой степени схожие с потоками в магазинах, основанные на услугах ускорителя расширение (например, процесс оформления заказа). Корзина клиента и другие данные о клиенте хранятся в сеансе и сохраняются между последующими запросами (исходя из предположения, что JSESSIONID сохранен). Клиент может больше сосредоточиться на реальных действиях, которые он хочет выполнить, а не на хранении и предоставлении всех контекстных данных. |
| Формат ответа | <ul style="list-style-type: none"> Ответы предоставляются в формате XML или JSON, в зависимости от запроса. |

Разделение ресурсов версий

Отдельные сервлеты

Для каждой версии существует отдельный сервлеt, определенный в web.xml. Эти сервлеты имеют разные конфигурации на основе Java. Также есть сервлеt, определенный для авторизации OAuth - в данном случае это общая функция для обеих версий.

Определение сервлета v1 выглядит следующим образом:

```
...
<сервлет>
<имя-сервлета>springmvc-v1</имя-сервлета> <класс-
сервлета>org.springframework.web.servlet.DispatcherServlet</класс-сервлета> <параметр-init>

<имя-параметра>contextClass</имя-параметра>
<значение-параметра>
    org.springframework.web.context.support.AnnotationConfigWebApplicationContext </param-value>

</init-param>
<init-param>
    <имя-параметра>contextConfigLocation</имя-параметра>
    <значение-параметра>
        de.hybris.platform.ycommercewebservices.v1.config.WebConfig </param-value>

</init-param>
<load-on-startup>1</load-on-startup> </сервлет>

<servlet-mapping>
<имя-сервлета>springmvc-v1</имя-сервлета> <url-
шаблон>/v1/*</url-шаблон>
</servlet-mapping>
...
```

Определение сервлета v2 выглядит следующим образом:

```
...
<сервлет>
<имя-сервлета>springmvc-v2</имя-сервлета> <класс-
сервлета>org.springframework.web.servlet.DispatcherServlet</класс-сервлета> <параметр-init>

<имя-параметра>contextClass</имя-параметра>
<значение-параметра>
    org.springframework.web.context.support.AnnotationConfigWebApplicationContext </param-value>

</init-param>
<init-param>
    <имя-параметра>contextConfigLocation</имя-параметра>
    <значение-параметра>
        de.hybris.platform.ycommercewebservices.v2.config.WebConfig </param-value>

</init-param>
<load-on-startup>1</load-on-startup> </сервлет>

<servlet-mapping>
<имя-сервлета>springmvc-v2</имя-сервлета> <url-
шаблон>/v2/*</url-шаблон>
</servlet-mapping>
...
```

Весенняя конфигурация

Отдельный веб-контекст для сервлетов

Каждая версия имеет свой собственный веб-контекст, определенный на основе конфигурации Java. Конфигурация Java определяет обработчик отображения запросов, преобразователи сообщений и обработчики исключений. Она также импортирует стандартную конфигурацию XML для соответствующей версии.

WebConfig.java конфигурация для v1:

```
/*
 * Конфигурация Spring, которая заменяет тег <mvc:annotation-driven>. Позволяет переопределить default
 * RequestMappingHandlerMapping с нашим собственным обработчиком сопоставления
 *
 */
@Configuration
@ImportResource(
{ "WEB-INF/config/v1/springmvc-v1-servlet.xml" }) открытый класс WebConfig
расширяет WebMvcConfigurationSupport {

    @Pесурс
    частный список<HttpMessageConverter<?>> messageConvertersV1;

    @Pесурс
    частный список<HandlerExceptionResolver> exceptionResolversV1;

    частный ApplicationContext applicationContext;

    @Переопределить
    @Бин
    public RequestMappingHandlerMapping {      requestMappingHandlerMapping()

        окончательный CommerceHandlerMapping handlerMapping = new CommerceHandlerMapping("v1");
        handlerMapping.setOrder(0);
        handlerMapping.setDetectHandlerMethodsInAncestorContexts(true);

        handlerMapping.setContentNegotiationManager(mvcContentNegotiationManager());

    }

    @Переопределить
    protected void configureMessageConverters(final List<HttpMessageConverter<?>> конвертеры) {

        converters.addAll(messageConvertersV1);
    }

    @Переопределить
    protected void configureHandlerExceptionResolvers(final List<HandlerExceptionResolver> exceptionResolvers) {

        окончательный ExceptionHandlerExceptionResolver exceptionHandlerExceptionResolver = new ExceptionHandlerExceptionRes
        exceptionHandlerExceptionResolver.setApplicationContext(applicationContext);
        exceptionHandlerExceptionResolver.setContentNegotiationManager(mvcContentNegotiationManager());
        exceptionHandlerExceptionResolver.setMessageConverters(getMessageConverters());
        exceptionHandlerExceptionResolver.afterPropertiesSet();

        exceptionResolvers.add(exceptionHandlerExceptionResolver);
        exceptionResolvers.addAll(exceptionResolversV1); exceptionResolvers.add(new
            ResponseStatusExceptionResolver());
        exceptionResolvers.add(новый      DefaultHandlerExceptionResolver());
    }

    @Переопределить
    public void setApplicationContext(final ApplicationContext applicationContext) выдает BeansException {

        super.setApplicationContext(applicationContext);
        this.applicationContext = applicationContext;
    }
}


```

WebConfig.java конфигурация для v2:

```

/**
 * Конфигурация Spring, которая заменяет тег <mvc:annotation-driven>. Позволяет переопределить default
 * RequestMappingHandlerMapping с нашим собственным обработчиком сопоставления
 *
 */
@Configuration
@ImportResource(
{ "WEB-INF/config/v2/springmvc-v2-servlet.xml" }) открытый класс WebConfig
расширяет WebMvcConfigurationSupport {

    @Pесурс
    частный список<HttpMessageConverter<?>> messageConvertersV2;

    @Pесурс
    частный список<HandlerExceptionResolver> exceptionResolversV2;

    частный ApplicationContext applicationContext;

    @Переопределить
    @Бин
    public RequestMappingHandlerMapping {      requestMappingHandlerMapping()

        окончательный CommerceHandlerMapping handlerMapping = new CommerceHandlerMapping("v2");
        handlerMapping.setOrder(0);
        handlerMapping.setDetectHandlerMethodsInAncestorContexts(true);

        handlerMapping.setContentNegotiationManager(mvcContentNegotiationManager());

    }

    @Переопределить
    protected void configureMessageConverters(final List<HttpMessageConverter<?>> конвертеры) {

        converters.addAll(messageConvertersV2);
    }

    @Переопределить
    protected void configureHandlerExceptionResolvers(final List<HandlerExceptionResolver> exceptionResolvers) {

        окончательный ExceptionHandlerExceptionResolver exceptionHandlerExceptionResolver = new ExceptionHandlerExceptionRes
        exceptionHandlerExceptionResolver.setApplicationContext(applicationContext);
        exceptionHandlerExceptionResolver.setContentNegotiationManager(mvcContentNegotiationManager());
        exceptionHandlerExceptionResolver.setMessageConverters(getMessageConverters());
        exceptionHandlerExceptionResolver.afterPropertiesSet();

        exceptionResolvers.add(exceptionHandlerExceptionResolver);
        exceptionResolvers.addAll(exceptionResolversV2); exceptionResolvers.add(new
            ResponseStatusExceptionResolver());
        exceptionResolvers.add(новый      DefaultHandlerExceptionResolver());
    }

    @Переопределить
    public void setApplicationContext(final ApplicationContext applicationContext) выдает BeansException {

        super.setApplicationContext(applicationContext);
        this.applicationContext = applicationContext;
    }
}

```

Общий контекст

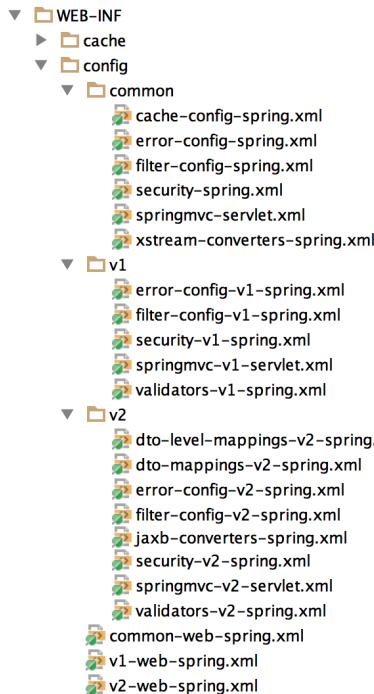
В web.xml есть общий контекст, определенный. Beans, определенные в этом контексте, доступны для обоих сервлетов. Этот контекст должен содержать конфигурацию безопасности и bean-компоненты, используемые в фильтрах.

```
...
<контекст-параметр>
<param-name>contextConfigLocation</param-name> <param-value>WEB-INF/
ycommercewebservices-web-spring.xml</param-value> </context-param>
...

```

Локализация файлов конфигурации

Файлы конфигурации пружин можно найти в WEB-INF/config. Файлы конфигурации, специфичные для версий, находятся в подкаталогах v1, v2.



Включение версии 1

Старая версия v1 по умолчанию отключена. Чтобы включить ее, раскомментируйте фильтры и определение сервлета в web.xml, как показано в примере ниже.

```
<!-- Раскомментируйте, чтобы сделать версию v1 доступной -->
<!--
<фильтр>
  <описание>
    Пружинная конфигурированная цепочка пружинных фильтрующих
    бобов </description>
  <filter-name>commerceWebServicesFilterChainV1</filter-name> <filter-
  class>org.springframework.web.filter.DelegatingFilterProxy</filter-class> </filter>

  <фильтр-отображение>
    <имя-фильтра>commerceWebServicesFilterChainV1</имя-фильтра> <имя-
    сервлета>springmvc-v1</имя-сервлета>
  </фильтр-отображение>
  <фильтр-отображение>
    <имя-фильтра>httpPutFormFilter</имя-фильтра> <имя-
    сервлета>springmvc-v1</имя-сервлета> </соответствие-
    фильтра>
  </фильтр-отображение>
<сервлет>
  <имя-сервлета>springmvc-v1</имя-сервлета> <класс-
  сервлета>org.springframework.web.servlet.DispatcherServlet</класс-сервлета> <параметр-init>
    <имя-параметра>contextClass</имя-параметра>
    <значение-параметра>
      org.springframework.web.context.support.AnnotationConfigWebApplicationContext </param-value>
  </параметр-init>
  <параметр>
    <имя-параметра>contextConfigLocation</имя-параметра>
    <значение-параметра>
      de.hybris.platform.ycommercewebservices.v1.config.WebConfig </param-value>
  </параметр>
</сервлет>

```

```

</init-param>
<load-on-startup>1</load-on-startup> </servlet>

<servlet-mapping>
    <имя-сервлета>springmvc-v1</имя-сервлета> <url-
        шаблон>/v1/*</url-шаблон>
    </servlet-mapping>
    - - -
    <!-- КОНЕЦ Раскомментируйте, чтобы сделать версию v1 доступной -->

```

Кроме того, включите конфигурацию пружины v1 в `commercewebservices-web-spring.xml`:

```

<!-- Раскомментируйте, чтобы включить версию v1 --> <!--<import
resource="config/v1-web-spring.xml"/>-->

```

Сопутствующая информация

[Реализация RESTful](#)

[Обновление до расширений OCC](#)

Вызовы OCC v1 REST

Получите обзор вызовов REST, доступных в версии 1 SAP Commerce OCC API.

-Осторожность

Эта страница относится к программному обеспечению, которое было устарело как часть Accelerator UI и устаревших расширений шаблонов OCC. Для получения дополнительной информации см.

[Прекращение поддержки пользовательских интерфейсов Accelerator и старых расширений шаблонов OCC](#).

Обзор

В настоящее время возможно получить список всех продуктов, удовлетворяющих заданным критериям, и сведения о продукте, указанном по его коду. Дочерние страницы этого документа содержат информацию о доступных в настоящее время вызовах, сгруппированных по ресурсу. Каждый документ содержит таблицы, представляющие примеры URL-запросов вместе с доступными параметрами.

Параметры запроса локализации

Каждый из вызовов может содержать дополнительные параметры URL, которые изменяют локализацию возвращаемых объектов. Общие параметры:

- `<язык>`: Изменяет язык локализованных значений в ответе. Укажите код языка ISO в качестве значения. Если нет `<язык>` Если указан параметр, то ответ локализуется на языке по умолчанию вашего базового магазина.
- `<curr>`: Изменяет валюту вашего вызова веб-сервиса. Это означает, что все вычисления выполняются в запрошенной валюте, и все значения цен представлены с использованием запрошенной валюты. Укажите код валюты ISO в качестве значения. Если нет `<curr>` Если указан параметр, то используется валюта по умолчанию вашего базового магазина.

Вы можете использовать эти параметры с каждым запрошенным ресурсом. Параметры обрабатываются отдельно в специализированном фильтре HTTP-запросов. Проверьте следующие примеры:

- `https://localhost:9001/rest/v1/mysite/cart?curr=USD&lang=en`: Для запроса используйте английский язык и доллар США.
- `https://localhost:9001/rest/v1/mysite/cart/entry?curr=USD`: Использовать доллар США и язык магазина mysite по умолчанию.
- `https://localhost:9002/rest/v1/mysite/customers/current?lang=de`: Использовать немецкий язык и валюту по умолчанию в магазине mysite.

Параметры сайта

Во многих URL-адресах ресурсов есть параметр пути к сайту, например `https://localhost:9001/rest/v1/electronics/catalogs` где `<электроника>` это `<BaseSite.uid>` свойство веб-сайта. В вашей пользовательской реализации вам необходимо добавить свой собственный базовый сайт:

- Шаблон URL, например (?i) `^https://localhost(:[0-9]+)?/rest/.*$`

Конфигурировать `productUrlPattern`, например `{путь-к-категории}/{название-продукта}/p/{код-продукта}`

Вы можете использовать следующие параметры:

- <{baseSite-uid}>
- <{category-path}>
- <{название-продукта}>
- <{код-продукта}>
- Конфигурировать категорияUrlPattern, например /{путь-категории}/с/{код-категории}

Вы можете использовать следующие параметры:

- <{baseSite-uid}>
- <{category-path}>
- <{код-категории}>
- <{catalog-id}>
- <{catalogVersion}>

Вот пример ответа:

<https://localhost:9001/rest/v1/electronics/catalogs/electronicsProductCatalog/Online/categories/brands?options=PRODUCTS>

```
<категория>
  <id>бренды</id>
  <lastModified>2012-08-13T09:11:00+02:00</lastModified>
  <name>Бренды</name>
  <url>/Бренды/с/бренды</url>

<подкатегории/>
<продукты>
  <продукт>
    <averageRating>3.5</averageRating>
    <purchasable>верно</purchasable>
    <code>478828</code>
    <url>/Open-Catalogue/Камеры/Цифровые-камеры/Цифровые-зеркальные-камеры/10-2-мегапиксельные-D-зеркальные-камеры-со-стандартным-зум-объективом/p/478828
    <производитель>Sony</производитель>
    <name>10,2-мегапиксельная цифровая зеркальная фотокамера со стандартным зум-объективом</name>
  <baseOptions/>
```

Если вы не укажете параметр пути к сайту или он не существует, вы получите следующее исключение.

`de.hybris.platform.ycommercewebservices.exceptions.InvalidResourceException: Базовый сайт electronics не существует`

Вы можете легко настроить свойства сайта, используя панель администрирования Backoffice в разделе **Веб-сайт WCM**.

Использование API с сохранением и без сохранения состояния

Хотя большинство ресурсов API можно использовать в режиме без сохранения состояния, есть некоторые ресурсы, которые требуют, чтобы клиент API отслеживал файлы cookie для управления сеансом. Как правило, мы советуем клиентам API отслеживать все файлы cookie, которые они получают от сервера, и автоматически перезаписывать существующие файлы cookie в случае, если сервер отправляет новое значение файла cookie для существующего имени файла cookie.

Использование файлов cookie для управления сеансом особенно важно для всех ресурсов корзины. POST в /{сайт}/корзина/запись будет создавать новую корзину в памяти каждый раз, когда вы выполняете запрос, если вы пропустите куки, которые вы получаете от сервера. Если вы пропустите передачу куки, вы не сможете ни сохранить корзину клиента, ни добавить несколько продуктов в отдельных запросах таким образом.

Новые файлы cookie отправляются с сервера с помощью HTTP-ответов и заголовка Set-Cookie. Вашему API-клиенту необходимо проверить, доступен ли заголовок Set-Cookie. Если он доступен, извлеките информацию о файлах cookie. Смотрите пример на основе кода Groovy.

```
//con — это HttpURLConnection илиHttpsURLConnection
def cookie = con.getHeaderField('Set-Cookie') //например JSESSIONID=C69059FD51C68E610321A818B0019DB2; Path=/rest; def cookieNoPath = cookie.split(';')[0] //теперь только значение cookie
```

Как только ваш API отправляет HTTP-запрос на вызов одного из ресурсов, файл cookie должен быть частью заголовков запроса.

```
con.setRequestProperty("Cookie", cookie)
```

Корзина ресурсов

Обзор доступных ресурсов корзины.

-Примечание

Примеры URL используются только для презентационных целей. В настроенной среде необходимо заменить адрес сервера на свой собственный.

Последовательность вызова методов API корзины

Последовательность вызова методов из ресурса Cart имеет важное значение. В следующей таблице представлен пример простого рабочего процесса.

| Метод | URL | Текст запроса |
|----------|---|---------------|
| получать | https://localhost:9002/rest/v1/electronics/customers | |
| получать | https://localhost:9002/authorizationserver/oauth/token | |
| получать | https://localhost:9002/rest/v1/customers/current/addresses | |
| получать | https://localhost:9002/rest/v1/electronics/cart | |
| получать | https://localhost:9002/rest/v1/electronics/cart/entry | |
| получать | https://localhost:9002/rest/v1/electronics/cart/entry | |
| получать | https://localhost:9002/rest/v1/electronics/cart | |
| помещать | https://localhost:9002/rest/v1/electronics/cart/entry/0 | |
| удалить | https://localhost:9002/rest/v1/electronics/cart/entry/0 | |
| получать | https://localhost:9002/rest/v1/electronics/cart | |
| помещать | https://localhost:9002/rest/v1/electronics/cart/address/delivery/\$id\$ | |

| Метод | URL | Текст запроса |
|----------|---|--|
| | | |
| УДАЛИТЬ | https://localhost:9002/rest/v1/electronics/cart/address/delivery | |
| ПОЛУЧАТЬ | https://localhost:9002/rest/v1/electronics/cart/deliverymodes | |
| ПОМЕЩАТЬ | https://localhost:9002/rest/v1/electronics/cart/address/delivery/\$id\$ | |
| ПОЛУЧАТЬ | https://localhost:9002/rest/v1/electronics/cart/deliverymodes | |
| ПОМЕЩАТЬ | https://localhost:9002/rest/v1/electronics/cart/deliverymodes/premium-gross | |
| УДАЛИТЬ | https://localhost:9002/rest/v1/electronics/cart/deliverymodes | |
| ПОМЕЩАТЬ | https://localhost:9002/rest/v1/electronics/cart/deliverymodes/standardgross | |
| ПОЛУЧАТЬ | https://localhost:9002/rest/v1/electronics/customers/current/paymentinfos | |
| ПОМЕЩАТЬ | https://localhost:9002/rest/v1/electronics/cart/paymentinfo/\$paymentInfoId\$ | |
| ПОЧТА | https://localhost:9002/rest/v1/electronics/cart/authorize | |
| ПОЧТА | https://localhost:9002/rest/v1/electronics/cart/placeorder | |
| ПОЛУЧАТЬ | https://localhost:9002/rest/v1/orders | |
| ПОЛУЧАТЬ | https://localhost:9002/rest/v1/orders/\$orderNumber\$ | |
| ПОЧТА | https://localhost:9002/rest/v1/customers/current/logout | |
| ПОЧТА | https://localhost:9002/rest/v1/electronics/cart/paymentinfo | Необходимо установить заголовок Content-Type: на application/x-www-form-urlencoded |

| Метод | URL | Текст запроса |
|-------|-----|---|
| | | <p>Имя владельца счета: "Anja Hertz" Номер карты: "4111111111111111" Тип карты: "visa"</p> <p>Месяц истечения срока: "01"</p> <p>Год окончания: "2013"</p> <p>сохранено: "истина"</p> <p>defaultPaymentInfo: "true"</p> <p>billingAddress.titleCode: "Мистер"</p> <p>billingAddress.firstName: "Аня"</p> <p>billingAddress.lastName: "Герц"</p> <p>billingAddress.line1: "тест1"</p> <p>billingAddress строка2: "тест2"</p> <p>billingAddress.postalCode: "12345"</p> <p>billingAddress.town: "somecity"</p> <p>billingAddress.country.isocode: "</p> |

Ресурс: /{site}/cart

| | |
|-------------------|---|
| Метод | получать |
| Описание | <p>Возвращает корзину сеанса, если такая корзина существует. Если в текущем сеансе корзины не было:</p> <ul style="list-style-type: none"> Для вошедшего в систему пользователя корзина восстанавливается, если она существует, и возвращается Для анонимного пользователя или при отсутствии сохраненной корзины создается и возвращается пустой объект данных mockcart. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/cart |
| Аутентификация | Никто |
| Параметры запроса | Непригодный |
| Заголовки | <ul style="list-style-type: none"> Принимать <p>(Стиль: заголовок, Обязательно: true):</p> <p>Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json</p> |
| Представления | приложение/xml, приложение/json |
| Файлы cookie | Для этого требуется информация о сеансе, отправленная в cookie или через :jsessionid=... в URL. В противном случае создается новый сеанс и, следовательно, новая корзина. |

Ресурс: /{site}/cart/entry

| | |
|-------------------|---|
| Метод | ПОЧТА |
| Описание | Добавляет продукт в корзину сеанса. Если корзина сеанса еще не существует, она создается заранее. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/cart/entry |
| Аутентификация | Никто |
| Параметры запроса | <p>Параметры необходимо указать в теле POST.</p> <ul style="list-style-type: none"> <код> <p>(Стиль: код, Обязательно: true):</p> <p>Код продукта, который необходимо добавить в корзину. Поиск продукта выполняется для текущей версии каталога продуктов сеанса.</p> |

| | |
|---------------|---|
| | <ul style="list-style-type: none"> • <КОЛ-ВО> <p>(Стиль: qty, Обязательно: false, По умолчанию: 1);</p> <p>Количество нового товара в корзине.</p> |
| Заголовки | <ul style="list-style-type: none"> • Принимать <p>(Стиль: заголовок, Обязательно: true);</p> <p>Заголовок Accept необходимо отправлять с каждым запросом. Он может быть приложение/xml или приложение/json</p> |
| Представления | приложение/xml, приложение/json |
| Файлы cookie | Требует информацию о сеансе, отправленную в куки или через ;jsessionid=... в URL. В противном случае создается новый сеанс и, следовательно, новая корзина. Чтобы корзина была доступна в следующих запросах, добавьте куки JSESSIONID к следующим вызовам. |

Ресурс: /{site}/cart/entry/{entryNumber}

| | |
|-------------------|--|
| Метод | ПОМЕЩАТЬ |
| Описание | Обновляет количество записей в корзине. Если нет записи в корзине с запрошенным номером, CommerceCartModificationException возвращается в ответ. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/cart/entry/0?qty=5 |
| Аутентификация | Никто |
| Параметры запроса | <p>Параметры необходимо указать в теле PUT. HttpPutFormContentFilter используется для чтения тела PUT.</p> <ul style="list-style-type: none"> • <КОЛ-ВО> <p>(Стиль: qty, Обязательно: true);</p> <p>Новое количество для ввода в корзину.</p> |
| Заголовки | <ul style="list-style-type: none"> • Тип контента <p>application/x-www-form-urlencoded (Стиль: заголовок, Обязательно: true);</p> <p>Тип контента application/x-www-form-urlencoded требуется для HttpPutFormContentFilter</p> <ul style="list-style-type: none"> • Принимать <p>(Стиль: заголовок, Обязательно: true);</p> <p>Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json</p> |
| Представления | приложение/xml, приложение/json |
| Файлы cookie | Требует информацию о сеансе, отправленную в куки или через ;jsessionid=... в URL. В противном случае создается новый сеанс и, следовательно, новая корзина. Чтобы корзина была доступна в следующих запросах, добавьте куки JSESSIONID к следующим вызовам. |

Ресурс: /{site}/cart/entry/{entryNumber}

| | |
|----------|---|
| Метод | УДАЛИТЬ |
| Описание | Удаляет запись корзины. Если записи корзины с запрошенным номером нет, в ответе возвращается CommerceCartModificationException. |

| | |
|-------------------|--|
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/cart/entry/0 |
| Аутентификация | Никто |
| Параметры запроса | Непригодный. |
| Заголовки | <ul style="list-style-type: none"> ● Принимать (Стиль: заголовок, Обязательно: true): Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json |
| Представления | приложение/xml, приложение/json |
| Файлы cookie | Он требует информацию о сеансе, отправленную в куки или через ;jsessionid=... в URL. В противном случае создается новый сеанс и, следовательно, новая корзина. Чтобы корзина была доступна в следующих запросах, добавьте куки JSESSIONID к следующим вызовам. |

Ресурс: /{site}/cart/address/delivery/{id}

| | |
|---|---|
| Метод | ПОМЕЩАТЬ |
| Описание | Устанавливает адрес доставки для корзины сеанса. Идентификатор адреса должен быть РК текущего адреса. Страна адреса должна быть среди стран доставки текущего базового магазина. Метод возвращает данные корзины, которые должны отражать изменение в корзине. |
| Предварительные условия состояния оформления заказа | Должна быть создана корзина сессии. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/cart/address/delivery/123 |
| Аутентификация | Доступ ограничен ROLE_CUSTOMERGROUP. Этот метод ограничен каналом HTTPS. |
| Параметры запроса | Непригодный. |
| Заголовки | <ul style="list-style-type: none"> ● Принимать (Стиль: заголовок, Обязательно: true): Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json |
| Представления | приложение/xml, приложение/json |
| Файлы cookie | Для этого требуется информация о сеансе, отправленная в куки или через ;jsessionid=... в URL. В противном случае создается новый сеанс и, следовательно, новая корзина. Чтобы корзина была доступна в следующих запросах, добавьте куки JSESSIONID к следующим вызовам. |

Ресурс: /{site}/cart/address/delivery/

| | |
|-------------------|---|
| Метод | УДАЛИТЬ |
| Описание | Удаляет адрес доставки из корзины сеанса. Ответ включает данные корзины, которые должны отражать изменение в корзине. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/cart/address/delivery |
| Аутентификация | Доступ ограничен ROLE_CUSTOMERGROUP. Этот метод ограничен каналом HTTPS. |
| Параметры запроса | Непригодный. |
| Заголовки | <ul style="list-style-type: none"> ● Принимать (Стиль: заголовок, Обязательно: true): |

Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json

| | |
|---------------|---|
| Представления | приложение/xml, приложение/json |
| Файлы cookie | Для этого требуется информация о сеансе, отправленная в куки или через ;jsessionid=... в URL. В противном случае создается новый сеанс и, следовательно, новая корзина. Чтобы корзина была доступна в следующих запросах, добавьте куки JSESSIONID к следующим вызовам. |

Ресурс: /{site}/cart/deliverymodes/

| | |
|---|--|
| Метод | получать |
| Описание | Возвращает все поддерживаемые способы доставки для текущего базового магазина и адреса доставки корзины. |
| Предварительные условия состояния оформления заказа | В корзине должен быть указан адрес доставки, в противном случае будет возвращен пустой список. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/cart/deliverymodes |
| Аутентификация | Доступ ограничен ROLE_CUSTOMERGROUP. Этот метод ограничен каналом HTTPS. |
| Параметры запроса | Непригодный. |
| Заголовки | <ul style="list-style-type: none"> ● Принимать (Стиль: заголовок, Обязательно: true): Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json |
| Представления | приложение/xml, приложение/json |
| Файлы cookie | Требует информацию о сеансе, отправленную в куки или через ;jsessionid=... в URL. В противном случае создается новый сеанс и, следовательно, новая корзина. Чтобы корзина была доступна в следующих запросах, добавьте куки JSESSIONID к следующим вызовам |

Ресурс: /{site}/cart/deliverymodes/{code}

| | |
|-------------------|---|
| Метод | помещать |
| Описание | Устанавливает режим доставки в соответствии с кодом для текущей корзины. Метод возвращает данные корзины, которые должны отражать изменение в корзине. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/cart/deliverymodes/standardgross |
| Аутентификация | Доступ ограничен ROLE_CUSTOMERGROUP. Этот метод требует базовой аутентификации HTTP и ограничен каналом HTTPS. |
| Параметры запроса | Непригодный. |
| Заголовки | <ul style="list-style-type: none"> ● Принимать (Стиль: заголовок, Обязательно: true): Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json. |
| Представления | приложение/xml, приложение/json |
| Файлы cookie | Для этого требуется информация о сеансе, отправленная в cookie или через ;jsessionid=... в URL. В противном случае создается новый сеанс и, следовательно, новая корзина. Чтобы корзина была доступна в следующих запросах, добавьте cookie JSESSIONID к следующим вызовам. |

Ресурс: /{site}/cart/deliverymodes/

| | |
|-------------------|--|
| Метод | УДАЛИТЬ |
| Описание | Удаляет режим доставки из текущей корзины. Ответ включает данные корзины, которые должны отражать изменение в корзине. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/cart/deliverymodes |
| Аутентификация | Доступ ограничен ROLE_CUSTOMERGROUP. Этот метод требует базовой аутентификации HTTP и ограничен каналом HTTPS. |
| Параметры запроса | Непригодный. |
| Заголовки | <ul style="list-style-type: none"> • Принимать (Стиль: заголовок, Обязательно: true): Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json. |
| Представления | приложение/xml, приложение/json |
| Файлы cookie | Для этого требуется информация о сеансе, отправленная в куки или через jsessionid=... в URL. В противном случае создается новый сеанс и, следовательно, новая корзина. Чтобы корзина была доступна в следующих запросах, добавьте куки JSESSIONID к следующим вызовам. |

Ресурс: /{site}/cart/paymentinfo/

| | |
|---|---|
| Метод | ПОЧТА |
| Описание | Создает новую платежную информацию по кредитной карте для текущего пользователя. Ответ включает данные платежной информации. |
| Предварительные условия состояния оформления заказа | Должна быть тележка для сеансов. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/cart/paymentinfo |
| Аутентификация | Доступ ограничен ROLE_CUSTOMERGROUP. Этот метод требует базовой аутентификации HTTP и ограничен каналом HTTPS. |
| Текст запроса | <p>Необходимо предоставить как URL-кодированное тело сообщения:</p> <ul style="list-style-type: none"> • <имя владельца аккаунта> (Требуется: правда); • <номер_карты> (Требуется: правда); • <тип карты> (Требуется: правда); <p>Вызов ПОЛУЧИТЬ /{site}/cardtypes заранее узнать, какие типы карт поддерживаются</p> <ul style="list-style-type: none"> • <expiryMonth> (Требуется: правда); • <год_исчечения> (Требуется: правда); • <сохранено> (Требуется: ложь); <p>Если платежная информация должна быть сохранена для клиента, ее можно будет повторно использовать для будущих заказов.</p> |

| | |
|---------------|---|
| | <ul style="list-style-type: none"> • <<i>defaultPaymentInfo</i>> <p>(Требуется: false, по умолчанию: false);</p> <ul style="list-style-type: none"> • <<i>billingAddress.titleCode</i>> <p>(Требуется: правда);</p> <ul style="list-style-type: none"> • <адрес выставления счетов.<i>rstName</i>> <p>(Требуется: правда);</p> <ul style="list-style-type: none"> • <адрес выставления счета.фамилия> <p>(Требуется: правда);</p> <ul style="list-style-type: none"> • <<i>billingAddress.line1</i>> <p>(Требуется: правда);</p> <ul style="list-style-type: none"> • <адрес_билинга.строка2> <p>(Требуется: ложь);</p> <ul style="list-style-type: none"> • <адрес выставления счета.почтовый индекс> <p>(Требуется: правда);</p> <ul style="list-style-type: none"> • <адрес_билинга.город> <p>(Требуется: правда);</p> <ul style="list-style-type: none"> • <адрес_билинга.страна.<i>isocode</i>> <p>(Требуется: правда);</p> |
| Заголовки | <ul style="list-style-type: none"> • Принимать <p>(Стиль: заголовок, Обязательно: true);</p> <p>Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json.</p> |
| Представления | приложение/xml, приложение/json |
| Файлы cookie | Для этого требуется информация о сеансе, отправленная в куки или через ;jsessionid=... в URL. В противном случае создается новый сеанс и, следовательно, новая корзина. Чтобы корзина была доступна в следующих запросах, добавьте куки JSESSIONID к следующим вызовам. |

Ресурс: /{site}/cart/paymentinfo/{id}

| | |
|---|---|
| Метод | помещать |
| Описание | Добавляет данные для оплаты кредитной картой в соответствии с идентификатором корзины текущего пользователя. |
| Предварительные условия состояния оформления заказа | Должна быть сессия корзины. Ответ включает данные корзины, которые должны отражать изменение в корзине. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/cart/paymentinfo/1234 |
| Аутентификация | Доступ ограничен ROLE_CUSTOMERGROUP. Этот метод требует базовой аутентификации HTTP и ограничен каналом HTTPS. |
| Заголовки | <ul style="list-style-type: none"> • Принимать <p>(Стиль: заголовок, Обязательно: true);</p> <p>Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json.</p> |
| Представления | приложение/xml, приложение/json |
| Файлы cookie | Он требует информацию о сеансе, отправленную в cookie или через ;jsessionid=... в URL. В противном случае создается новый сеанс и, следовательно, новая корзина. Чтобы корзина была доступна в |

Ресурс: /{site}/cart/authorize

| | |
|---|---|
| Метод | ПОЧТА |
| Описание | Авторизует платеж по кредитной карте с помощью кода безопасности CCV. Ответ содержит данные корзины. |
| Предварительные условия состояния оформления заказа | Должна быть сеансовая корзина с ассоциированной оплатой кредитной картой. Ответ содержит данные корзины, которые должны отражать изменение в корзине. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/cart/authorize |
| Аутентификация | Доступ ограничен ROLE_CUSTOMERGROUP. Этот метод требует базовой аутентификации HTTP и ограничен каналом HTTPS. |
| Параметры запроса | <p>Необходимо предоставить как URL-кодированное тело сообщения:</p> <ul style="list-style-type: none"> • <securityCode> <p>(Требуется: правда):</p> |
| Заголовки | <ul style="list-style-type: none"> • Принимать <p>(Стиль: заголовок, Обязательно: true):</p> <p>Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json</p> |
| Представления | приложение/xml, приложение/json |
| Файлы cookie | Для этого требуется информация о сеансе, отправленная в куки или через ;jsessionid=... в URL. В противном случае создается новый сеанс и, следовательно, новая корзина. Чтобы корзина была доступна в следующих запросах, добавьте куки JSESSIONID к следующим вызовам. |

Ресурс: /{site}/cart/placeorder

| | |
|---|---|
| Метод | ПОЧТА |
| Описание | Размещает заказ на основе корзины сеанса. Ответ содержит новые данные заказа. |
| Предварительные условия состояния оформления заказа | Сеанс пользователя должен иметь действительную корзину. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/cart/placeorder |
| Аутентификация | Доступ ограничен ROLE_CUSTOMERGROUP. Этот метод требует базовой аутентификации HTTP и ограничен каналом HTTPS. |
| Заголовки | <ul style="list-style-type: none"> • Принимать <p>(Стиль: заголовок, Обязательно: true):</p> <p>Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json.</p> |
| Представления | приложение/xml, приложение/json |
| Файлы cookie | Требует информацию о сеансе, отправленную в куки или через ;jsessionid=... в URL. Используйте куки JSESSIONID для следующих вызовов. |

Ресурс: /{site}/cart/restore

| | |
|-------------------|--|
| Метод | получать |
| Описание | Восстанавливает анонимную корзину по guid. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/cart/restore?guid=xxxxx |
| Аутентификация | Доступ ограничен ROLE_CLIENT и ROLE_TRUSTED_CLIENT. Этот метод ограничен каналом HTTPS. |
| Заголовки | <ul style="list-style-type: none"> ● Принимать (Стиль: заголовок, Обязательно: true); Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json. |
| Представления | приложение/xml, приложение/json |

Ресурс: /{site}/cart/promotion/{promotionCode}

| | |
|-------------------|--|
| Метод | ПОЧТА |
| Описание | Включает продвижение заказа, заданное promotionCode для текущей корзины. Если продвижение применено успешно, то измененная корзина будет возвращена. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/cart/OrderThreshold15Скидка |
| Аутентификация | Доступ ограничен ROLE_CLIENT, ROLE_CUSTOMERGROUP и ROLE_TRUSTED_CLIENT. Этот метод ограничен каналом HTTPS. |
| Заголовки | <ul style="list-style-type: none"> ● Принимать (Стиль: заголовок, Обязательно: true); Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json. |
| Представления | приложение/xml, приложение/json |
| Файлы cookie | Требует информацию о сеансе, отправленную в куки или через ;jsessionid=... в URL. Используйте куки JSESSIONID для следующих вызовов. |

Ресурс: /{site}/cart/promotion/{promotionCode}

| | |
|-------------------|--|
| Метод | УДАЛИТЬ |
| Описание | Отключает продвижение заказа, заданное promotionCode для текущей корзины. Если продвижение удалено успешно, то измененная корзина будет возвращена. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/cart/promotion/OrderThreshold15Скидка |
| Аутентификация | Доступ ограничен ROLE_CLIENT, ROLE_CUSTOMERGROUP и TRUSTED_CLIENT. Этот метод требует HTTP-аутентификации клиента или заказчика и ограничен каналом HTTPS. |
| Заголовки | <ul style="list-style-type: none"> ● Принимать (Стиль: заголовок, Обязательно: true); Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json. |
| Представления | приложение/xml, приложение/json |
| Файлы cookie | Требует информацию о сеансе, отправленную в куки или через ;jsessionid=... в URL. Используйте куки JSESSIONID для следующих вызовов. |

Ресурс: /{site}/cart/voucher/{voucherCode}

| Метод | ПОЧТА |
|-------------------|--|
| Описание | Применяет ваучер, предоставленный voucherCode для текущей корзины. Если код ваучера применен успешно, то будет возвращена измененная корзина. Ответ JSON должен содержать атрибут appliedVouchers с информацией о ваучере. Ответ XML должен содержать тег appliedVouchers с информацией о ваучере. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/cart/voucher/xyz-MHE2-B8L5-LPHE |
| Аутентификация | Доступ ограничен ROLE_CLIENT, ROLE_CUSTOMERGROUP и ROLE_TRUSTED_CLIENT. Этот метод ограничен каналом HTTPS. |
| Параметры запроса | Непригодный. |
| Заголовки | <ul style="list-style-type: none"> • Принимать (Стиль: заголовок, Обязательно: true): Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json. |
| Представления | приложение/xml, приложение/json |
| Файлы cookie | Требует информацию о сеансе, отправленную в куки или через jsessionid=... в URL. Используйте куки JSESSIONID для следующих вызовов. |

Ресурс: /{site}/cart/voucher/{voucherCode}

| Метод | УДАЛИТЬ |
|-------------------|--|
| Описание | <p>Метод удаляет ваучер, заданный voucherCode из текущей корзины. Если код ваучера удален успешно, то будет возвращена измененная корзина.</p> <ul style="list-style-type: none"> • В ответе JSON атрибут appliedVouchers больше не должен содержать выпущенный ваучер. "appliedVouchers": [] • В XML-ответе тег appliedVouchers больше не должен содержать выпущенный ваучер. <примененные ваучеры /> |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/cart/voucher/xyz-MHE2-B8L5-LPHE |
| Аутентификация | Доступ ограничен ROLE_CLIENT, ROLE_CUSTOMERGROUP и TRUSTED_CLIENT. Этот метод ограничен каналом HTTPS. |
| Параметры запроса | Непригодный. |
| Заголовки | <ul style="list-style-type: none"> • Принимать (Стиль: заголовок, Обязательно: true): Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json. |
| Представления | приложение/xml, приложение/json |
| Файлы cookie | Для этого требуется информация о сеансе, отправленная в файле cookie или через <jsessionid=...> в URL. Используйте куки JSESSIONID для следующих вызовов. |

Каталоги ресурсов

обзор доступных ресурсов каталогов

-Примечание

Примеры URL используются только для презентационных целей. В настроенной среде необходимо заменить адрес сервера на свой собственный.

Ресурс: /{site}/catalogs

| | |
|-------------------|--|
| Метод | получать |
| Описание | Возвращает все каталоги с версиями, которые определены для базового магазина. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/catalogs |
| Параметры запроса | <ul style="list-style-type: none"> <опции>(Стиль: запрос, Обязательно: false): <p>Он определяет уровень запрашиваемой детализации. Этот параметр может иметь комбинацию следующих значений, разделенных ,: BASIC, CATEGORIES, PRODUCTS, SUBCATEGORIES. Помните, что значение должно быть правильно закодировано в URL.</p> <p>Параметры зависят друг от друга: например, параметр PRODUCTS имеет смысл только в том случае, если также используется параметр CATEGORIES, поскольку элементы products встроены в категорию в объекте ответа.</p> <p>Пример:<опции>=CATEGORIES,PRODUCTS запрашивает корневые категории с продуктами.</p> |
| Заголовки | <ul style="list-style-type: none"> Принять (Стиль: заголовок, Обязательно: true): <p>Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json.</p> |
| Представления | приложение/xml, приложение/json |

Ресурс: /{site}/catalogs/{id}

| | |
|-------------------|--|
| Метод | получать |
| Описание | Возвращает каталог по идентификатору с версиями, определенными для текущего базового хранилища. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/catalogs/electronics Каталог продукции |
| Параметры запроса | <ul style="list-style-type: none"> <опции>(Стиль: запрос, Обязательно: false): <p>Он определяет уровень запрашиваемой детализации. Этот параметр может иметь комбинацию следующих значений, разделенных ,: BASIC,CATEGORIES,PRODUCTS,SUBCATEGORIES. Помните, что значение должно быть правильно закодировано в URL.</p> <p>Параметры зависят друг от друга: параметр PRODUCTS имеет смысл только в том случае, если также используется параметр CATEGORIES, поскольку элементы products встроены в категорию в объекте ответа.</p> <p>Пример:<опции>=CATEGORIES, PRODUCTS запрашивает корневые категории с продуктами.</p> |
| Заголовки | <ul style="list-style-type: none"> Принять (Стиль: заголовок, Обязательно: true): |

Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json.

Представления

приложение/xml, приложение/json

Ресурс: /{site}/catalogs/{catalogId}/{catalogVersionId}

| | |
|-------------------|--|
| Метод | получать |
| Описание | Возвращает информацию о версии каталога, существующей для текущего базового магазина. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/catalogs/electronics Каталог продукции/Онлайн |
| Параметры запроса | <ul style="list-style-type: none"> <опции>(Стиль: запрос, Обязательно: false): <p>Он определяет уровень запрашиваемой детализации. Этот параметр может иметь комбинацию следующих значений, разделенных по,:,BASIC,CATEGORIES,PRODUCTS,SUBCATEGORIES. Помните, что значение должно быть правильно закодировано в URL.</p> <p>Параметры зависят друг от друга: например, параметр PRODUCTS имеет смысл только в том случае, если также используется параметр CATEGORIES, поскольку элементы products встроены в категорию в объекте ответа.</p> <p>Пример: <опции>=CATEGORIES,PRODUCTS запрашивает корневые категории с продуктами.</p> |
| Заголовки | <ul style="list-style-type: none"> Принять (Стиль: заголовок, Обязательно: true): <p>Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json.</p> |
| Представления | приложение/xml, приложение/json |

Ресурс: /{site}/catalogs/{catalogId}/{catalogVersionId}/categories/{categoryCode}

| | |
|-------------------|---|
| Метод | получать |
| Описание | Возвращает информацию о категориях, которые существуют в версии каталога базового магазина. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/catalogs/electronics Каталог Продуктов/Онлайн/категории/1 |
| Параметры запроса | <ul style="list-style-type: none"> <опции>(Стиль: запрос, Обязательно: false): <p>Он определяет уровень запрашиваемой детализации. Этот параметр может иметь комбинацию следующих значений, разделенных на:BASIC,PRODUCTS,SUBCATEGORIES. Помните, что значение должно быть правильно закодировано в URL.</p> <p>Параметры зависят друг от друга: параметр PRODUCTS имеет смысл только в том случае, если также используется параметр CATEGORIES, поскольку элементы products встроены в категорию в объекте ответа.</p> <p>Пример: <опции>=CATEGORIES,PRODUCTS запрашивает корневые категории с продуктами.</p> |
| Заголовки | <ul style="list-style-type: none"> Принять(Стиль: заголовок, Обязательно: true): <p>Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json.</p> |
| Представления | приложение/xml, приложение/json |

Группа клиентов ресурсов

Обзор доступных ресурсов группы клиентов.

-Примечание

Примеры URL используются только для презентационных целей. В настроенной среде необходимо заменить адрес сервера на свой собственный.

Ресурс: /{site}/customergroups

| | |
|-------------------|--|
| Метод | ПОЧТА |
| Описание | Создает новую группу клиентов - прямую подгруппу группы клиентов. Метод требует ROLE_CUSTOMERMANAGERGROUP и защищенный канал HTTPS. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/customergroups |
| Параметры запроса | <ul style="list-style-type: none"> • <uid>(Стиль: body, Обязательно: true): Uid для новой группы клиентов. • <localizedName>(Стиль: body, Обязательно: false): Имя в текущей локали. |
| Заголовки | <ul style="list-style-type: none"> • Принять (Стиль: заголовок, Обязательно: true): Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json. |
| Представления | приложение/xml, приложение/json |

Ресурс: /{site}/customergroups/{uid}/members

| | |
|-------------------|--|
| Метод | ПОМЕЩАТЬ |
| Описание | Назначить пользователей в группу клиентов с заданными <uid>. Требуется роль CUSTOMERMANAGERGROUP и защищенный канал HTTPS. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/customergroups/customergroup/members |
| Параметры запроса | <ul style="list-style-type: none"> • <участник>(Стиль: body, Обязательно: true): Список идентификаторов пользователей для назначения группе клиентов. Список должен быть в форме:<участник>=<uid1>&<участник>=<uid2>... |
| Заголовки | <ul style="list-style-type: none"> • Принять (Стиль: заголовок, Обязательно: true): Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json. |
| Представления | приложение/xml, приложение/json |

Ресурс: /{site}/customergroups

| | |
|----------|---|
| Метод | ПОЛУЧАТЬ |
| Описание | Возвращает все группы клиентов, которые являются прямыми подгруппами customergroup. Требует роль CUSTOMERMANAGERGROUP и защищенный канал HTTPS. |

| | |
|-------------------|--|
| Пример URL-адреса | <code>https://localhost:9002/rest/v1/electronics/customergroups?currentPage=0&pageSize=20</code> |
| Параметры запроса | <ul style="list-style-type: none"> <текущая страница>(Стиль: параметр URL, Обязательно: false, По умолчанию: 0): номер текущей страницы (начинается с 0) <Размер страницы>(Стиль: параметр URL, Обязательно: false, По умолчанию: 2147483647): количество групп клиентов, возвращаемых на одной странице |
| Заголовки | <ul style="list-style-type: none"> Принять (Стиль: заголовок, Обязательно: true): <p>Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json.</p> |
| Представления | приложение/xml, приложение/json |

Ресурс: /{site}/customergroups/{uid}

| | |
|-------------------|--|
| Метод | ПОЛУЧАТЬ |
| Действительность | 5.0 |
| Описание | <p>Возвращает группу клиентов с указанным uid. Требуется роль CUSTOMERMANAGERGROUP.</p> <p>Запрошенная группа должна быть подгруппой customergroup, в противном случае выдается исключение InvalidCustomerGroupException. Требуется безопасный канал.</p> |
| Пример URL-адреса | <p>https://localhost:9002/rest/v1/electronics/customergroups/customergroup/{myGroup}?options=УЧАСТНИКИ,ПОДГРУППЫ</p> |
| Параметры запроса | <ul style="list-style-type: none"> <опции>(Стиль: параметр url, Обязательно: false): <ul style="list-style-type: none"> BASIC - значение по умолчанию. Ответ содержит только <uid>, <локализованное имя> и количество членов, УЧАСТНИКИ - при использовании этой опции ответ будет содержать информацию о пользователях-участниках (клиентах), ПОДГРУППА - при использовании этой опции ответ будет содержать информацию о подгруппах этой группы. <p>Вы можете комбинировать варианты.</p> |
| Заголовки | <ul style="list-style-type: none"> Принять (Стиль: заголовок, Обязательно: true): <p>Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json.</p> |
| Представления | приложение/xml, приложение/json |

Ресурс: /{site}/customergroups/{uid}/members/{userId}

| | |
|-------------------|--|
| Метод | УДАЛИТЬ |
| Описание | Удалить пользователя с указанным <userId> из группы клиентов с заданным <uid>. Требуется роль CUSTOMERMANAGERGROUP. |
| Пример URL-адреса | <code>https://localhost:9002/rest/v1/electronics/customergroups/customergroup/members/dab61415-7832-4daf-ae1d-c48988e5bfef</code> |
| Заголовки | <ul style="list-style-type: none"> Принять (Стиль: заголовок, Обязательно: true): <p>Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json.</p> |

| | |
|---------------|---------------------------------|
| Представления | приложение/xml, приложение/json |
|---------------|---------------------------------|

Клиенты ресурсов

Обзор доступных Клиентам ресурсов.

-Примечание

Примеры URL используются только для презентационных целей. В настроенной среде необходимо заменить адрес сервера на свой собственный.

Ресурс: /customers/current/logout

| | |
|-------------------|--|
| Метод | ПОЧТА |
| Описание | Веб-сервис для выхода клиента из системы. Завершает сеанс. Включает фильтр выхода Spring Security. |
| Пример URL-адреса | https://localhost:9002/rest/v1/customers/current/logout |
| Заголовки | <ul style="list-style-type: none"> • Принять (Стиль: заголовок, Обязательно: true); <p>Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json.</p> |

Ресурс: /{site}/customers

| | |
|-------------------|--|
| Метод | ПОЧТА |
| Описание | Зарегистрируйте клиента. |
| Пример URL-адреса | <ul style="list-style-type: none"> • Доступ ограничен ROLE_CLIENT, поэтому его могут получить только зарегистрированные клиенты API. Токен доступа клиента должен быть отправлен с запросом. Чтобы получить токен для заданного <идентификатор_клиента> сначала необходимо отправить запрос на авторизацию: <ul style="list-style-type: none"> ◦ curl --location 'https://localhost:9002/authorizationserver/oauth' \ -- заголовок 'Content-Type: application/x-www-form-urlencoded' \ -- данные-urlencode 'client_secret={clientSecret}' \ -- данные-urlencode 'grant_type=client_credentials' \ -- данные-urlencode 'client_id={client_id}'' • Теперь вы можете отправить регистрационный звонок: <ul style="list-style-type: none"> ◦ https://localhost:9002/rest/v1/electronics/customers |
| Параметры запроса | <ul style="list-style-type: none"> • <логин>(Стиль: body, Обязательно: true): <p>Логин клиента. Логин клиента нечувствителен к регистру.</p> • <пароль>(Стиль: body, Обязательно: true): <p>Пароль клиента.</p> • <ПервоеИмя>(Стиль: body, Обязательно: true): <p>Имя клиента.</p> • <фамилия>(Стиль: body, Обязательно: true): <p>Фамилия клиента.</p> • <titleCode>(Стиль: body, Обязательно: false): <p>Код заголовка клиента. Список кодов см. в StoreController</p> |

| | |
|---------------|---|
| | <ul style="list-style-type: none"> • <токен_доступа>(Стиль: body, Обязательно: false); <p>Токен доступа предоставлен данному клиенту.</p> |
| Заголовки | <ul style="list-style-type: none"> • Принять(Стиль: заголовок, Обязательно: true); <p>Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json.</p> |
| Представления | приложение/xml, приложение/json |

Ресурс: /{site}/customers/current/addresses/default/{id}

| | |
|-------------------|---|
| Метод | ПОМЕШАТЬ |
| Описание | Установить адрес как адрес клиента по умолчанию. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/customers/current/addresses/default/12345 |
| Параметры запроса | <ul style="list-style-type: none"> • <идентификатор>(Стиль: body, Обязательно: true); <p>Идентификатор адреса.</p> |
| Аутентификация | <ul style="list-style-type: none"> • Доступ ограничен ROLE_CUSTOMERGROUP, поэтому токен доступа клиента должен быть отправлен с запросом. Чтобы получить токен для данного клиента, сначала необходимо отправить вызов авторизации: <ul style="list-style-type: none"> ○ curl --location 'https://localhost:9002/authorizationserver/oauth' \ - - заголовок 'Content-Type: application/x-www-form-urlencoded' \ - - данные-urlencode 'имя пользователя={uid}' \ - - данные-urlencode 'пароль={pwd}' \ - - данные-urlencode 'client_secret={clientSecret}' \ - - данные-urlencode 'grant_type=пароль' \ - - данные-urlencode 'client_id={client_id}' • Требуется HTTPS-канал |
| Заголовки | <ul style="list-style-type: none"> • Принять (Стиль: заголовок, Обязательно: true); <p>Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json.</p> |
| Представления | приложение/xml, приложение/json |

Ресурс: /{site}/customers/current/prole

| | |
|-------------------|--|
| Метод | ПОЧТА |
| Описание | Обновите профиль клиента. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/customers/current/profile |
| Параметры запроса | Необходимо предоставить как URL-кодированное тело поста. Будут обновлены только ненулевые значения. <ul style="list-style-type: none"> • <titleCode>(Требуется: ложь) • <ПервоеИмя>(Требуется: ложь) • <фамилия>(Требуется: ложь) • <язык>(Требуется: ложь) • <валюта>(Требуется: ложь) |
| Аутентификация | <ul style="list-style-type: none"> • Доступ ограничен ROLE_CUSTOMERGROUP, поэтому токен доступа клиента должен быть отправлен с запросом. Чтобы получить токен для данного клиента, сначала необходимо отправить вызов авторизации: |

| | |
|---------------|---|
| | <ul style="list-style-type: none"> curl --location 'https://localhost:9002/authorizationserver/oauth' \ -- заголовок 'Content-Type: application/x-www-form-urlencoded' \ -- данные-urlencode 'имя пользователя={uid}' \ -- данные-urlencode 'пароль={pwd}' \ -- данные-urlencode 'client_secret={clientSecret}' \ -- данные-urlencode 'grant_type=пароль' \ -- данные-urlencode 'client_id={client_id}' <p>• Требуется HTTPS-канал</p> |
| Заголовки | <ul style="list-style-type: none"> Принять (Стиль: заголовок, Обязательно: true); <p>Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json.</p> |
| Представления | приложение/xml, приложение/json |

Ресурс: /{site}/customers/current/addresses

| | |
|-------------------|--|
| Метод | получать |
| Описание | Получите адреса клиентов. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/customers/current/addresses |
| Аутентификация | <ul style="list-style-type: none"> Доступ ограничен ROLE_CUSTOMERGROUP, поэтому токен доступа клиента должен быть отправлен с запросом. <p>Чтобы получить токен для данного клиента, сначала необходимо отправить вызов авторизации:</p> <pre>curl --location 'https://localhost:9002/authorizationserver/oauth' \ -- заголовок 'Content-Type: application/x-www-form-urlencoded' \ -- данные-urlencode 'имя пользователя={uid}' \ -- данные-urlencode 'пароль={pwd}' \ -- данные-urlencode 'client_secret={clientSecret}' \ -- данные-urlencode 'grant_type=пароль' \ -- данные-urlencode 'client_id={client_id}'</pre> <p>• Требуется HTTPS-канал</p> |
| Заголовки | <ul style="list-style-type: none"> Принять (Стиль: заголовок, Обязательно: true); <p>Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json.</p> |
| Представления | приложение/xml, приложение/json |

Ресурс: /{site}/customers/current/addresses

| | |
|-------------------|--|
| Метод | ПОЧТА |
| Описание | Создать новый адрес. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/customers/current/addresses |
| Параметры запроса | <p>Видеть HttpRequestAddressDataPopulator класс. Необходимо предоставить как URL-кодированное тело сообщения.</p> <ul style="list-style-type: none"> <titleCode>(Требуется: правда) <ПервоеИмя>(Требуется: правда) <фамилия>(Требуется: правда) <строка1>(Требуется: правда) <строка2>(Требуется: ложь) <город>(Требуется: правда) <почтовый индекс>(Требуется: правда) |

| | |
|----------------|--|
| | <ul style="list-style-type: none"> • <страна.isocode>(Требуется: правда) • <регион.isocode>(Требуется: ложь) |
| Аутентификация | <ul style="list-style-type: none"> • Доступ ограничен ROLE_CUSTOMERGROUP, поэтому токен доступа клиента должен быть отправлен с запросом. <p>Чтобы получить токен для данного клиента, сначала необходимо отправить вызов авторизации:</p> <ul style="list-style-type: none"> ◦ curl --location 'https://localhost:9002/authorizationserver/oauth' \ -- заголовок 'Content-Type: application/x-www-form-urlencoded' \ -- данные-urlencode 'имя пользователя={uid}' \ -- данные-urlencode 'пароль={pwd}' \ -- данные-urlencode 'client_secret={clientSecret}' \ -- данные-urlencode 'grant_type=пароль' \ -- данные-urlencode 'client_id={client_id}' <ul style="list-style-type: none"> • Требуется HTTPS-канал |
| Заголовки | <ul style="list-style-type: none"> • Принять (Стиль: заголовок, Обязательно: true): <p>Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json.</p> |
| Представления | приложение/xml, приложение/json |

Ресурс: /{site}/customers/current/addresses/{id}

| | |
|-------------------|--|
| Метод | ПОМЕЩАТЬ |
| Описание | Обновите адрес. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/customers/current/addresses/12345 |
| Параметры запроса | <p>Видеть HttpRequestAddressDataPopulator класс. Необходимо предоставить как URL-кодированное тело поста. Будут обновлены только ненулевые значения.</p> <ul style="list-style-type: none"> • <titleCode>(Требуется: ложь) • <ПервоеИмя>(Требуется: ложь) • <фамилия>(Требуется: ложь) • <строка1>(Требуется: ложь) • <строка2>(Требуется: ложь) • <город>(Требуется: ложь) • <почтовый индекс>(Требуется: ложь) • <страна.isocode>(Требуется: ложь) • <регион.isocode>(Требуется: ложь) |
| Аутентификация | <ul style="list-style-type: none"> • Доступ ограничен ROLE_CUSTOMERGROUP, поэтому токен доступа клиента должен быть отправлен с запросом. <p>Чтобы получить токен для данного клиента, сначала необходимо отправить вызов авторизации:</p> <ul style="list-style-type: none"> ◦ curl --location 'https://localhost:9002/authorizationserver/oauth' \ -- заголовок 'Content-Type: application/x-www-form-urlencoded' \ -- данные-urlencode 'имя пользователя={uid}' \ -- данные-urlencode 'пароль={pwd}' \ -- данные-urlencode 'client_secret={clientSecret}' \ -- данные-urlencode 'grant_type=пароль' \ -- данные-urlencode 'client_id={client_id}' <ul style="list-style-type: none"> • Требуется HTTPS-канал |
| Заголовки | <ul style="list-style-type: none"> • Принять (Стиль: заголовок, Обязательно: true): <p>Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json.</p> |

- Content-Type application/x-www-form-urlencoded (Стиль: заголовок, Обязательно: true) Тип контента application/xwww-form-urlencoded требуетсяHttpPutFormContentFilter

Представления

приложение/xml, приложение/json

Ресурс: /{site}/customers/current/addresses/{id}

| | |
|-------------------|---|
| Метод | УДАЛИТЬ |
| Описание | Удалить адрес. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/customers/current/addresses/12345 |
| Параметры запроса | <ul style="list-style-type: none"> <идентификатор>(Стиль: body, Обязательно: true): Идентификатор адреса. |
| Аутентификация | <ul style="list-style-type: none"> Доступ ограничен ROLE_CUSTOMERGROUP, поэтому токен доступа клиента должен быть отправлен с запросом. Чтобы получить токен для данного клиента, сначала необходимо отправить вызов авторизации: <ul style="list-style-type: none"> curl --location 'https://localhost:9002/authorizationserver/oauth' \ -- заголовок 'Content-Type: application/x-www-form-urlencoded' \ -- данные-urlencode 'имя пользователя={uid}' \ -- данные-urlencode 'пароль={pwd}' \ -- данные-urlencode 'client_secret={clientSecret}' \ -- данные-urlencode 'grant_type=пароль' \ -- данные-urlencode 'client_id={client_id}' Требуется HTTPS-канал |
| Заголовки | <ul style="list-style-type: none"> Принять (Стиль: заголовок, Обязательно: true): Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json. |
| Представления | приложение/xml, приложение/json |

Ресурс: /{site}/customers/current

| | |
|-------------------|---|
| Метод | ПОЛУЧАТЬ |
| Описание | Возвращает профиль клиента. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/customers/current |
| Заголовки | <ul style="list-style-type: none"> Принять (Стиль: заголовок, Обязательно: true): Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или приложение/json. |
| Аутентификация | <ul style="list-style-type: none"> Доступ ограничен ROLE_CUSTOMERGROUP, поэтому токен доступа клиента должен быть отправлен с запросом. Чтобы получить токен для данного клиента, сначала необходимо отправить вызов авторизации: <ul style="list-style-type: none"> curl --location 'https://localhost:9002/authorizationserver/oauth' \ -- заголовок 'Content-Type: application/x-www-form-urlencoded' \ -- данные-urlencode 'имя пользователя={uid}' \ -- данные-urlencode 'пароль={pwd}' \ -- данные-urlencode 'client_secret={clientSecret}' \ -- данные-urlencode 'grant_type=пароль' \ -- данные-urlencode 'client_id={client_id}' Требуется HTTPS-канал |

| | |
|---------------|---------------------------------|
| Представления | приложение/xml, приложение/json |
|---------------|---------------------------------|

Ресурс: /{site}/customers/current/password

| | |
|-------------------|--|
| Метод | ПОЧТА |
| Описание | Изменить пароль клиента. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/customers/current/password |
| Параметры запроса | <ul style="list-style-type: none"> <старый>(Стиль: body, Обязательно: true): <p>Старый пароль.</p> <новый>(Стиль: body, Обязательно: true): <p>Новый пароль.</p> |
| Аутентификация | <ul style="list-style-type: none"> Доступ ограничен ROLE_CUSTOMERGROUP, поэтому токен доступа клиента должен быть отправлен с запросом. <p>Чтобы получить токен для данного клиента, сначала необходимо отправить вызов авторизации:</p> <ul style="list-style-type: none"> curl --location 'https://localhost:9002/authorizationserver/oauth' \ -- заголовок 'Content-Type: application/x-www-form-urlencoded' \ -- данные-urlencode 'имя пользователя={uid}' \ -- данные-urlencode 'пароль={pwd}' \ -- данные-urlencode 'client_secret={clientSecret}' \ -- данные-urlencode 'grant_type=пароль' \ -- данные-urlencode 'client_id={client_id}' <ul style="list-style-type: none"> Требуется HTTPS-канал |
| Заголовки | <ul style="list-style-type: none"> Принять (Стиль: заголовок, Обязательно: true): <p>Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json.</p> |
| Представления | приложение/xml, приложение/json |

Ресурс: /{site}/customers/current/forgottenpassword

| | |
|-------------------|--|
| Метод | ПОЧТА |
| Описание | Восстановите забытый пароль клиента. |
| Пример URL-адреса | <ul style="list-style-type: none"> Доступ ограничен ROLE_CLIENT, поэтому его могут получить только зарегистрированные клиенты API. Токен доступа клиента должен быть отправлен с запросом. Чтобы получить токен для заданного <идентификатор_клиента>сначала необходимо отправить запрос на авторизацию: <ul style="list-style-type: none"> curl --location 'https://localhost:9002/authorizationserver/oauth' \ -- заголовок 'Content-Type: application/x-www-form-urlencoded' \ -- данные-urlencode 'client_secret={clientSecret}' \ -- данные-urlencode 'grant_type=client_credentials' \ -- данные-urlencode 'client_id={client_id}' <ul style="list-style-type: none"> Теперь вы можете позвонить: <ul style="list-style-type: none"> https://localhost:9002/rest/v1/electronics/customers/current/forgottenpassword |
| Параметры запроса | <ul style="list-style-type: none"> <логин>(Стиль: body, Обязательно: true): <p>Логин клиента. Логин клиента нечувствителен к регистру.</p> |
| Заголовки | <ul style="list-style-type: none"> Принять (Стиль: заголовок, Обязательно: true): |

Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json.

Представления

приложение/xml, приложение/json

Ресурс: /{site}/customers/current/paymentinfos

| | |
|-------------------|--|
| Метод | получать |
| Описание | Возвращает платежную информацию о кредитной карте текущего клиента, которая была ранее сохранена при оформлении заказа. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/customers/current/paymentinfos |
| Заголовки | <ul style="list-style-type: none"> Принять (Стиль: заголовок, Обязательно: true); <p>Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json.</p> |
| Аутентификация | <ul style="list-style-type: none"> Доступ ограничен ROLE_CUSTOMERGROUP, поэтому токен доступа клиента должен быть отправлен с запросом. Чтобы получить токен для данного клиента, сначала необходимо отправить вызов авторизации: <ul style="list-style-type: none"> curl --location 'https://localhost:9002/authorizationserver/oauth/token' \ - - заголовок 'Content-Type: application/x-www-form-urlencoded' \ - - данные-urlencode 'имя пользователя={uid}' \ - - данные-urlencode 'пароль={pwd}' \ - - данные-urlencode 'client_secret={clientSecret}' \ - - данные-urlencode 'grant_type=пароль' \ - - данные-urlencode 'client_id={client_id}'' Требуется HTTPS-канал |
| Представления | приложение/xml, приложение/json |

Ресурс: /{site}/customers/current/paymentinfos/{id}

| | |
|-------------------|--|
| Метод | получать |
| Описание | Возвращает платежную информацию о кредитной карте клиента по идентификатору платежной информации. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/customers/current/paymentinfos/1234 |
| Заголовки | <ul style="list-style-type: none"> Принять (Стиль: заголовок, Обязательно: true); <p>Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json.</p> |
| Аутентификация | <ul style="list-style-type: none"> Доступ ограничен ROLE_CUSTOMERGROUP, поэтому токен доступа клиента должен быть отправлен с запросом. Чтобы получить токен для данного клиента, сначала необходимо отправить вызов авторизации: <ul style="list-style-type: none"> curl --location 'https://localhost:9002/authorizationserver/oauth/token' \ - - заголовок 'Content-Type: application/x-www-form-urlencoded' \ - - данные-urlencode 'имя пользователя={uid}' \ - - данные-urlencode 'пароль={pwd}' \ - - данные-urlencode 'client_secret={clientSecret}' \ - - данные-urlencode 'grant_type=пароль' \ - - данные-urlencode 'client_id={client_id}'' Требуется HTTPS-канал |
| Представления | приложение/xml, приложение/json |

Ресурс: /{site}/customers/current/paymentinfos/{id}

| | |
|-------------------|--|
| Метод | УДАЛИТЬ |
| Описание | Удаляет платежную информацию о кредитной карте клиента по идентификатору платежной информации. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/customers/current/paymentinfos/1234 |
| Заголовки | <ul style="list-style-type: none"> Принять (Стиль: заголовок, Обязательно: true); <p>Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json.</p> |
| Аутентификация | <ul style="list-style-type: none"> Доступ ограничен ROLE_CUSTOMERGROUP, поэтому токен доступа клиента должен быть отправлен с запросом. Чтобы получить токен для данного клиента, сначала необходимо отправить вызов авторизации: <pre>curl --location 'https://localhost:9002/authorizationserver/oauth/token' \ --заголовок 'Content-Type: application/x-www-form-urlencoded' \ --данные-urlencode 'имя пользователя={uid}' \ --данные-urlencode 'пароль={pwd}' \ --данные-urlencode 'client_secret={clientSecret}' \ --данные-urlencode 'grant_type=пароль' \ --данные-urlencode 'client_id={client_id}'</pre> <ul style="list-style-type: none"> Требуется HTTPS-канал |
| Представления | приложение/xml, приложение/json |

Ресурс: /{site}/customers/current/paymentinfos/{id}

| | |
|-------------------|--|
| Метод | ПОМЕЩАТЬ |
| Описание | Обновляет платежную информацию по кредитной карте существующего клиента по идентификатору платежной информации. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/customers/current/paymentinfos/1234 |
| Заголовки | <ul style="list-style-type: none"> Принять (Стиль: заголовок, Обязательно: true); <p>Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json.</p> |
| Параметры запроса | <p>Необходимо предоставить как URL-кодированное тело сообщения:</p> <ul style="list-style-type: none"> <имя владельца аккаунта>(Требуется: правда); <номер_карты>(Требуется: правда); <тип карты>(Требуется: правда); <p>Заранее вызовите GET /{site}/cardtypes, чтобы узнать, какие типы карт поддерживаются.</p> <ul style="list-style-type: none"> <expiryMonth>(Требуется: правда); <год_истечения>(Требуется: правда); <сохранено>(Требуется: ложь); <p>Сообщает, сохранена ли платежная информация для клиента и может ли она быть повторно использована для будущих заказов.</p> <ul style="list-style-type: none"> <defaultPaymentInfo>(Требуется: ложь); |
| Аутентификация | <ul style="list-style-type: none"> Доступ ограничен ROLE_CUSTOMERGROUP, поэтому токен доступа клиента должен быть отправлен с запросом. Чтобы получить токен для данного клиента, сначала необходимо отправить вызов авторизации: <pre>curl --location 'https://localhost:9002/authorizationserver/oauth/token' \ --заголовок 'Content-Type: application/x-www-form-urlencoded' \ --данные-urlencode 'имя пользователя={uid}' \ --данные-urlencode 'пароль={pwd}' \ --данные-urlencode 'client_secret={clientSecret}' \ --данные-urlencode 'grant_type=пароль' \ --данные-urlencode 'client_id={client_id}'</pre> |

- Требуется HTTPS-канал

Представления

приложение/xml, приложение/json

Ресурс: /{site}/customers/current/paymentinfos/{id}/address

| | |
|-------------------|---|
| Метод | ПОЧТА |
| Описание | Обновляет платежный адрес существующей платежной информации кредитной карты клиента по идентификатору платежной информации. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/customers/current/paymentinfos/1234/address |
| Заголовки | <ul style="list-style-type: none"> • Принять (Стиль: заголовок, Обязательно: true): <p>Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json.</p> |
| Параметры запроса | <p>Необходимо предоставить как URL-кодированное тело сообщения:</p> <ul style="list-style-type: none"> • <titleCode>(Требуется: правда): • <ПервоеИмя>(Требуется: правда): • <фамилия>(Требуется: правда): • <строка1>(Требуется: правда): • <строка2>(Требуется: правда): • <почтовый индекс>(Требуется: правда): • <город>(Требуется: правда): • <страна.iso code>(Требуется: правда): <p>Если платежная информация должна быть сохранена для клиента, ее можно будет повторно использовать для будущих заказов.</p> <ul style="list-style-type: none"> • <defaultPaymentInfo>(Требуется: ложь): |
| Аутентификация | <ul style="list-style-type: none"> • Доступ ограничен ROLE_CUSTOMERGROUP, поэтому токен доступа клиента должен быть отправлен с запросом. Чтобы получить токен для данного клиента, сначала необходимо отправить вызов авторизации: <pre>curl --location 'https://localhost:9002/authorizationserver/oauth/token' \ --заголовок 'Content-Type: application/x-www-form-urlencoded' \ --данные-urlencode 'имя пользователя={uid}' \ --данные-urlencode 'пароль={pwd}' \ --данные-urlencode 'client_secret={clientSecret}' \ --данные-urlencode 'grant_type=пароль' \ --данные-urlencode 'client_id={client_id}'</pre> <ul style="list-style-type: none"> • Требуется HTTPS-канал |
| Представления | приложение/xml, приложение/json |

Ресурс: /{site}/customers/current/customergroups

| | |
|-------------------|--|
| Метод | ПОЛУЧАТЬ |
| Описание | Возвращает все группы клиентов текущего клиента. Требуется роль доступа к группе клиентов и защищенный канал HTTPS. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/customers/current/customergroups |
| Заголовки | <ul style="list-style-type: none"> • Принять (Стиль: заголовок, Обязательно: true): <p>Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json.</p> |

| | |
|----------------|--|
| Аутентификация | <ul style="list-style-type: none"> Доступ ограничен ROLE_CUSTOMERGROUP, поэтому токен доступа клиента должен быть отправлен с запросом. <p>Чтобы получить токен для данного клиента, сначала необходимо отправить вызов авторизации:</p> <ul style="list-style-type: none"> curl --location 'https://localhost:9002/authorizationserver/oauth' \ -- заголовок 'Content-Type: application/x-www-form-urlencoded' \ -- данные-urlencode 'имя пользователя={uid}' \ -- данные-urlencode 'пароль={pwd}' \ -- данные-urlencode 'client_secret={clientSecret}' \ -- данные-urlencode 'grant_type=пароль' \ -- данные-urlencode 'client_id={client_id}' <ul style="list-style-type: none"> Требуется HTTPS-канал |
| Представления | приложение/xml, приложение/json |

Ресурс: /{site}/customers/current/login

| | |
|-------------------|---|
| Метод | ПОЧТА |
| Описание | Изменить логин клиента. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/customers/current/login |
| Параметры запроса | <ul style="list-style-type: none"> <новыйВход>(Стиль: body, Обязательно: true): Новый логин клиента. Логин клиента нечувствителен к регистру. <пароль>(Стиль: body, Обязательно: true): Текущий пароль клиента. |
| Заголовки | <ul style="list-style-type: none"> Принять (Стиль: заголовок, Обязательно: true): Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json. |
| Представления | приложение/xml, приложение/json |

Ресурс: /{site}/customers/{uid}/customergroups

| | |
|-------------------|--|
| Метод | ПОЛУЧАТЬ |
| Описание | Возвращает все группы клиентов указанного клиента. Требуется роль доступа admingroup и защищенный канал HTTPS. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/customers/demo/customergroups |
| Заголовки | <ul style="list-style-type: none"> Принять (Стиль: заголовок, Обязательно: true): Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json. |
| Аутентификация | <ul style="list-style-type: none"> Доступ ограничен ROLE_CUSTOMERGROUP, поэтому токен доступа клиента должен быть отправлен с запросом. <p>Чтобы получить токен для данного клиента, сначала необходимо отправить вызов авторизации:</p> <ul style="list-style-type: none"> curl --location 'https://localhost:9002/authorizationserver/oauth' \ -- заголовок 'Content-Type: application/x-www-form-urlencoded' \ -- данные-urlencode 'имя пользователя={uid}' \ -- данные-urlencode 'пароль={pwd}' \ -- данные-urlencode 'client_secret={clientSecret}' \ -- данные-urlencode 'grant_type=пароль' \ -- данные-urlencode 'client_id={client_id}' <ul style="list-style-type: none"> Требуется HTTPS-канал |

| | |
|---------------|---------------------------------|
| Представления | приложение/xml, приложение/json |
|---------------|---------------------------------|

Заказы на ресурсы

Список доступных ресурсов Orders.

-Примечание

Примеры URL используются только для презентационных целей. В настроенной среде необходимо заменить адрес сервера на свой собственный.

Ресурс: /{site}/orders

| | |
|-------------------|---|
| Метод | получать |
| Описание | Возвращает данные истории заказов для всех заказов, размещенных текущим пользователем для текущего базового магазина. Ответ содержит результат поиска заказов, отображаемый на нескольких страницах, если это необходимо. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/orders |
| Аутентификация | Этот метод требует базовой HTTP-аутентификации и ограничен каналом HTTPS. |
| Параметры запроса | <p>В качестве параметров URL-запроса:</p> <ul style="list-style-type: none"> • <статусы>(Требуется: ложь) Фильтрует только определенные статусы заказов. Это означает: <статусы> =CANCELLED, CHECKED_VALID вернет только заказы со статусом CANCELLED или CHECKED_VALID. • <текущая страница>(Требуется: ложь) Атрибут пагинации, указывающий, какая страница запрашивается • <размер страницы>(Требуется: ложь) Атрибут пагинации, указывающий запрошенный размер страницы • <сортировать>(Требуется: ложь) Атрибут пагинации, указывающий предпочтение сортировки |
| Заголовки | <ul style="list-style-type: none"> • Принять(Стиль: заголовок, Обязательно: true); Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json. |
| Представления | приложение/xml, приложение/json |

Ресурс: /{site}/orders/{code}

| | |
|-------------------|---|
| Метод | получать |
| Описание | Возвращает конкретные детали заказа. Ответ содержит объект подробной информации о заказе. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/orders/000001 |
| Аутентификация | Этот метод требует базовой HTTP-аутентификации и ограничен каналом HTTPS. |
| Заголовки | <ul style="list-style-type: none"> • Принять (Стиль: заголовок, Обязательно: true); |

| | |
|--|--|
| | Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json. |
|--|--|

| | |
|---------------|---------------------------------|
| Представления | приложение/xml, приложение/json |
|---------------|---------------------------------|

Ресурсные продукты

Список доступных ресурсов «Продукты».

-Примечание

Примеры URL используются только для презентационных целей. В настроенной среде необходимо заменить адрес сервера на свой собственный.

Ресурс: /{site}/products

| Метод | получать |
|-------------------|--|
| Описание | <p>Возвращает список продуктов и дополнительных данных, таких как доступные фасеты, доступные параметры сортировки и параметры пагинации. Он может включать предложения по написанию, например:</p> <pre><предложение по написанию> <suggestion>sony</suggestion> <query>sony:relevance</query> </ spellingSuggestion></pre> <p>Чтобы это заработало, вам необходимо:</p> <ul style="list-style-type: none"> Убедиться, что <code><enableSpellCheck>на</code>Положение установлено значение true. По умолчанию оно должно быть уже установлено значение true. Настройте индексированные свойства для использования при проверке орфографии. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/products? query=a&pageSize=40 |
| Аутентификация | Никто |
| Параметры запроса | <ul style="list-style-type: none"> <code><запрос></code>(Стиль: запрос, Обязательно: true): <p>Сериализованный запрос, свободный текстовый поиск, фасеты.</p> <p>Формат сериализованного запроса:</p> <pre>freeTextSearch:сортировка:facetKey1:facetValue1:facetKey2:facetValue2</pre> <p>Строка запроса должна быть закодирована в URL, и клиент должен убедиться, что элементы : не являются частью бесплатный текстовый поиск. Значение запроса также должно быть правильно закодировано в URL.</p> <ul style="list-style-type: none"> <code><текущая страница></code>(Стиль: запрос, Обязательно: false): <p>Текущая запрошенная страница результатов. Необязательно. Значение по умолчанию: 0.</p> <ul style="list-style-type: none"> <code><размер страницы></code>(Стиль: запрос, Обязательно: false): <p>Количество результатов, возвращаемых на страницу. Необязательно. Значение по умолчанию: 20.</p> <ul style="list-style-type: none"> <code><сортировать></code>(Стиль: запрос, Обязательно: false): <p>Метод сортировки, применяемый к отображаемым результатам поиска.</p> |
| Заголовки | <ul style="list-style-type: none"> Принять (Стиль: заголовок, Обязательно: true): <p>Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json.</p> |
| Представления | приложение/xml, приложение/json |

Ресурс: /{site}/products/{product_code}

| | |
|-------------------|---|
| Метод | получать |
| Описание | Возвращает сведения об одном продукте, указанном по коду продукта. Дополнительные параметры могут быть выражены с помощью параметра options. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/products/553637?option=ОПИСАНИЕ |
| Аутентификация | Никто |
| Параметры запроса | <ul style="list-style-type: none"> <опции>(Стиль: запрос, Обязательно: false): <p>Определяет уровень запрашиваемых данных. Может иметь комбинацию следующих значений, разделенных запятой: BASIC, DESCRIPTION, GALLERY, CATEGORIES, PROMOTIONS, STOCK, REVIEW, CLASSIFICATION, REFERENCES, PRICE.</p> <p>Помните, что значение должно быть правильно закодировано в URL.</p> <p>Пример: параметры=ОСНОВНОЙ,ОБЗОРзапрашивает базовый уровень детализации, а также отзывы о продукте.</p> |
| Заголовки | <ul style="list-style-type: none"> Принять (Стиль: заголовок, Обязательно: true): <p>Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json.</p> |
| Представления | приложение/xml, приложение/json |

Ресурс: /{site}/products/export/incremental

| | |
|-------------------|--|
| Метод | получать |
| Описание | Возвращает все продукты с примененной пагинацией. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/products/export/incremental?timestamp=2012-03-28T07:50:49%2B00:00&tPage=0&pageSize=10&options=BASIC |
| Аутентификация | <ul style="list-style-type: none"> Доступ ограничен ролью ROLE_TRUSTED_CLIENT, поэтому его могут получить только зарегистрированные и доверенные клиенты API. Требуется HTTPS-канал |
| Параметры запроса | <ul style="list-style-type: none"> <текущая страница>(Стиль: запрос, Обязательно: false): <p>Текущая запрошенная страница результатов, необязательно. Значение по умолчанию: 0.</p> < отметка времени > в формате RFC-8601 <размер страницы>(Стиль: запрос, Обязательно: false): <p>Количество результатов, возвращаемых на страницу. Необязательно. Значение по умолчанию: неограниченно.</p> <опции>(Стиль: запрос, Обязательно: false): <p>Он определяет уровень запрашиваемой детализации. Этот параметр может иметь комбинацию следующих значений, разделенных запятой: BASIC, DESCRIPTION, GALLERY, CATEGORIES, PROMOTIONS, STOCK, REVIEW, CLASSIFICATION, REFERENCES.</p> <p>Помните, что значение должно быть правильно закодировано в URL.</p> <p>Пример: параметры=ОСНОВНОЙ ОБЗОРзапрашивает базовый уровень детализации плюс обзоры продукта. </p> |
| Заголовки | <ul style="list-style-type: none"> Принять (Стиль: заголовок, Обязательно: true): |

Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json.

| | |
|---------------|---------------------------------|
| Представления | приложение/xml, приложение/json |
|---------------|---------------------------------|

Ресурс: /{site}/products/export/full

| | |
|-------------------|--|
| Метод | получать |
| Описание | Возвращает все продукты с примененной пагинацией. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/products/export/full?currentPage=0&pageSize=10&options=BASIC |
| Аутентификация | <ul style="list-style-type: none"> Доступ ограничен ролью ROLE_TRUSTED_CLIENT, поэтому его могут получить только зарегистрированные и доверенные клиенты API. Требуется HTTPS-канал |
| Параметры запроса | <ul style="list-style-type: none"> <текущая страница>(Стиль: запрос, Обязательно: false); Текущая запрошенная страница результатов, необязательно. Значение по умолчанию: 0. <размер страницы>(Стиль: запрос, Обязательно: false); Количество результатов, возвращаемых на страницу. Необязательно. Значение по умолчанию: неограниченно. <опции>(Стиль: запрос, Обязательно: false); Определяет уровень запрашиваемой детализации. Может иметь комбинацию следующих значений, разделенных запятой: BASIC, DESCRIPTION, GALLERY, CATEGORIES, PROMOTIONS, STOCK, REVIEW, CLASSIFICATION, REFERENCES. Имейте в виду, что значение должно быть правильно закодировано в URL. Пример: параметры=ОСНОВНОЙ ОБЗОРзапрашивает базовый уровень детализации, а также отзывы о продукте. |
| Заголовки | <ul style="list-style-type: none"> Принять (Стиль: заголовок, Обязательно: true); Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json. |
| Представления | приложение/xml, приложение/json |

Ресурс: /{site}/products/suggest

| | |
|-------------------|--|
| Метод | получать |
| Описание | <p>Относится к теме исправления проверки орфографии. Это список всех доступных предложений, связанных с данным <термин> и ограничивает результаты заданным значением < макс > параметр. Возвращает результат в формате XML.</p> <pre>< предложения > < предложение > < value > штатив </ value > </ предложение > < предложение > < value > лента </ value > </ предложение > </ предложения ></pre> |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/products/suggest?term=t&max=3 |

| | |
|-------------------|---|
| Параметры запроса | <ul style="list-style-type: none"> • <термин>(Стиль: body, Обязательно: true): Определенный термин • <макс>(Стиль: body, Обязательно: false): Задает предел результатов. Значение по умолчанию: 10. |
| Заголовки | <ul style="list-style-type: none"> • Принять (Стиль: заголовок, Обязательно: true): Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json. |

Ресурс: /{site}/products/{code}/reviews

| | |
|-------------------|--|
| Метод | ПОЧТА |
| Описание | Метод создает новый отзыв клиента как анонимного пользователя. Метод отвечает данными отзыва. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/product/816780/reviews |
| Параметры запроса | <p>Необходимо предоставить как URL-кодированное тело сообщения.</p> <ul style="list-style-type: none"> • <рейтинг>Обязательно. Значение должно быть от 1 до 5. • <псевдоним>Необязательный • <заголовок>Необходимый • <комментарий>Необходимый |
| Заголовки | <ul style="list-style-type: none"> • Принять (Стиль: заголовок, Обязательно: true): Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json. |
| Представления | приложение/xml, приложение/json |

Ресурс: /{site}/products/expressUpdate

| | |
|-------------------|--|
| Метод | получать |
| Описание | Метод возвращает продукты, добавленные в экспресс-канал обновлений. Очередь очищается с помощью определенного cronjob. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/products/expressUpdate?timestamp=2013-06-21T11%3A30%3A21%2B01%3A00 |
| Аутентификация | <ul style="list-style-type: none"> • Доступ ограничен ролью ROLE_TRUSTED_CLIENT, поэтому его могут получить только зарегистрированные и доверенные клиенты API. • Требуется HTTPS-канал |
| Параметры запроса | <ul style="list-style-type: none"> • <отметка времени>: Требуется в формате RFC-8601. Из очереди извлекаются только элементы, более новые, чем заданный параметр. • <каталог>: Необязательно. Возвращаются только продукты из этого каталога. Формат: CatalogId:ВерсияКаталога |
| Заголовки | <ul style="list-style-type: none"> • Принять (Стиль: заголовок, Обязательно: true): |

| | |
|---------------|--|
| | Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json. |
| Представления | приложение/xml, приложение/json |

Ресурс: /{site}/products/{code}/nearLocation

| | |
|-------------------|--|
| Метод | получать |
| Описание | Метод возвращает уровень запасов товара, отсортированный по расстоянию от указанного местоположения, переданного параметром freetext. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/products/816780/nearLocation?location=Tokio&pageSize=5&tPage=2 |
| Параметры запроса | <p>Необходимо предоставить как URL-кодированное тело сообщения.</p> <ul style="list-style-type: none"> ● <местоположение>: Обязательно. Свободный текстовый параметр местоположения. ● <размер страницы>: Необязательный ● <текущая страница>: Необязательный |
| Заголовки | <ul style="list-style-type: none"> ● Принять (Стиль: заголовок, Обязательно: true); <p>Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json.</p> |
| Представления | приложение/xml, приложение/json |

Ресурс: /{site}/products/{code}/nearLatLong

| | |
|-------------------|--|
| Метод | получать |
| Описание | Метод возвращает уровень запасов товара, отсортированный по расстоянию от определенного местоположения, переданного с помощью параметров долготы и широты. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/products/816780/nearLatLong?широта=35.6816951&долгота=139.7650482&pageSize=5&tPage=2 |
| Параметры запроса | <p>Необходимо предоставить как URL-кодированное тело сообщения.</p> <ul style="list-style-type: none"> ● <долгота>: Обязательно. Параметр местоположения по долготе. ● <широта>: Обязательно. Параметр местоположения широты. ● <размер страницы>: Необязательный ● <текущая страница>: Необязательный |
| Заголовки | <ul style="list-style-type: none"> ● Принять (Стиль: заголовок, Обязательно: true); <p>Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json.</p> |
| Представления | приложение/xml, приложение/json |

Разные ресурсы

Список доступных ресурсов категории «Разное», не входящих в другие группы.

-Примечание

Примеры URL используются только для презентационных целей. В настроенной среде необходимо заменить адрес сервера на свой собственный.

Ресурс: /{site}/languages

| | |
|-------------------|--|
| Метод | получать |
| Описание | Список всех доступных языков (все языки базового магазина). В случае пустого списка языков для базового магазина, возвращается список всех языков в системе. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/languages |
| Заголовки | <ul style="list-style-type: none"> Принять (Стиль: заголовок, Обязательно: true); <p>Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json.</p> |

Ресурс: /{site}/currencies

| | |
|-------------------|--|
| Метод | получать |
| Описание | Список всех доступных валют (все используемые валюты для текущего магазина). В случае пустого списка валют для магазинов, возвращает список всех валют в системе. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/currencies |
| Заголовки | <ul style="list-style-type: none"> Принять (Стиль: заголовок, Обязательно: true); <p>Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json.</p> |

Ресурс: /{site}/deliverycountries

| | |
|-------------------|--|
| Метод | получать |
| Описание | Список всех поддерживаемых стран доставки для текущего магазина. Список отсортирован в алфавитном порядке. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/deliverycountries |
| Заголовки | <ul style="list-style-type: none"> Принять (Стиль: заголовок, Обязательно: true); <p>Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json.</p> |

Ресурс: /{site}/cardtypes

| | |
|-------------------|---|
| Метод | получать |
| Описание | Перечислите поддерживаемые типы платежных карт. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/cardtypes |

| | |
|-----------|--|
| Заголовки | <ul style="list-style-type: none"> Принять (Стиль: заголовок, Обязательно: true); <p>Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json.</p> |
|-----------|--|

Ресурс: /{site}/titles

| | |
|-------------------|--|
| Метод | получать |
| Описание | Перечислите все локализованные названия. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/titles |
| Заголовки | <ul style="list-style-type: none"> Принять (Стиль: заголовок, Обязательно: true); <p>Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json.</p> |

Магазины ресурсов

Список доступных ресурсов Магазина.

-Примечание

Примеры URL используются только для презентационных целей. В настроенной среде необходимо заменить адрес сервера на свой собственный.

Ресурс: /{site}/stores

| | |
|-------------------|---|
| Метод | получать |
| Описание | <p>Перечисляет все местоположения магазинов, которые находятся рядом с местоположением, указанным в запросе или по широте и долготе.</p> <pre><searchResult> <результаты /> <paginacija> <pageSize>10</pageSize> <currentPage>0</currentPage> <totalResults>49</totalResults> <totalPages>5</totalPages> </paginacija> <boundEastLongitude>135.2738964</boundEastLongitude> <boundSouthLatitude>33.5829778</boundSouthLatitude> <sourceLatitude>48.1366069</sourceLatitude> <boundNorthLatitude>62.6902359999999</boundNorthLatitude> <sourceLongitude>11.5770851</sourceLongitude> <locationText>мюнхен</locationText> <boundWestLongitude>-112.1197262</boundWestLongitude> </ searchResult></pre> |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/stores? query=munich https://localhost:9002/rest/v1/electronics/stores? query=munich |
| Заголовки | <ul style="list-style-type: none"> Принять (Стиль: заголовок, Обязательно: true); <p>Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json.</p> |

| | |
|-------------------|---------------------------------|
| Параметры запроса | Непригодный. |
| Представления | приложение/xml, приложение/json |

Ресурс: /{site}/stores/{name}

| | |
|-------------------|---|
| Метод | получать |
| Описание | Возвращает местоположение магазина по его уникальному названию. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/stores/Чиба |
| Заголовки | <ul style="list-style-type: none"> Принять (Стиль: заголовок, Обязательно: true): Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json. |
| Параметры запроса | Непригодный. |
| Представления | приложение/xml, приложение/json |

Продвижение ресурсов

Список доступных ресурсов для акций.

-Примечание

Примеры URL используются только для презентационных целей. В настроенной среде необходимо заменить адрес сервера на свой собственный.

Ресурс: /{site}/promotions

| | |
|-------------------|---|
| Метод | получать |
| Описание | Метод возвращает список акций |
| Пример URL-адреса | <ul style="list-style-type: none"> https://localhost:9002/rest/v1/electronics/promotions?type=all https://localhost:9002/rest/v1/electronics/promotions?type=all&promotionGroup=electronicsPromoGrp |
| Аутентификация | <ul style="list-style-type: none"> Доступ ограничен ROLE_CLIENT и ROLE_TRUSTED_CLIENT. Только зарегистрированные клиенты API могут получить его. Токен доступа клиента должен быть отправлен с запросом. Чтобы получить токен для заданного <идентификатор_клиента>, вам необходимо сначала отправить авторизационный вызов: <ul style="list-style-type: none"> ПОЧТА -https://localhost:9002/rest/oauth/token Заголовок: Тип содержимого: application/x-www-form-urlencoded Текст запроса: <code>client_id=\${clientId}&client_secret=\${clientSecret}&grant_type=client_credentials</code> Требуется HTTPS-канал |
| Параметры запроса | <ul style="list-style-type: none"> <тип>(Требуется: правда): Определяет, какой тип акций должен быть возвращен. Поддерживаемые значения для этого параметра: <ul style="list-style-type: none"> <все> Все доступные акции возвращаются <продукт>: Возвращаются только рекламные акции на продукты |

| | |
|---------------|---|
| | <ul style="list-style-type: none"> ○ <заказ>: Возвращаются только акции по заказу ● <promotionGroup>(Требуется: ложь): Возвращаются только акции из этой группы |
| Заголовки | <ul style="list-style-type: none"> ● Принять (Стиль: заголовок, Обязательно: true): Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json. |
| Представления | приложение/xml, приложение/json |

Ресурс: /{site}/promotions/{code}

| | |
|-------------------|--|
| Метод | получать |
| Описание | Возвращает сведения об одной акции, указанной кодом акции. Дополнительные параметры могут быть выражены с помощью <опции> параметр. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/promotions/MultiBuy?options=EXTENDED |
| Аутентификация | Доступ ограничен ROLE_CLIENT и ROLE_TRUSTED_CLIENT. Этот метод требует HTTP-аутентификации клиента и ограничен каналом HTTPS. |
| Параметры запроса | <ul style="list-style-type: none"> ● <опции>(Стиль: запрос, Обязательно: false): Определяет уровень запрашиваемых данных. Может иметь комбинацию следующих значений, разделенных запятой: BASIC, EXTENDED. The <БАЗОВЫЙ> значение включено по умолчанию. Значение должно быть правильно URL-кодировано. Пример: параметры=БАЗОВЫЙ,РАСШИРЕННЫЙ запрашивает базовый уровень детализации плюс расширенный. |
| Заголовки | <ul style="list-style-type: none"> ● Принять (Стиль: заголовок, Обязательно: true): Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json. |
| Представления | приложение/xml, приложение/json |

Ресурсные ваучеры

Список доступных ресурсов ваучеров.

-Примечание

Примеры URL используются только для презентационных целей. В настроенной среде необходимо заменить адрес сервера на свой собственный.

Ресурс: /{site}/vouchers/{code}

| | |
|-------------------|--|
| Метод | получать |
| Описание | Возвращает сведения об одном ваучере, указанном по коду ваучера. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/vouchers/abc-9PSW-EDH2-RXKA |
| Аутентификация | <ul style="list-style-type: none"> ● Доступ ограничен ROLE_CLIENT, поэтому его могут получить только зарегистрированные клиенты API. Токен доступа клиента должен быть отправлен с запросом. Чтобы получить токен для заданного <идентификатор_клиента>, вам необходимо отправить авторизационный вызов |

первый: ПОСТ -<https://localhost:9002/authorizationserver/oauth/token>

Заголовок:

Тип контента: приложение/x-www-form-urlencoded

Текст запроса:

client_id=\${clientId}&client_secret=\${clientSecret}&grant_type=client_credentials

- Требуется HTTPS-канал

Заголовки

- Принять (Стиль: заголовок, Обязательно: true);

Заголовок Accept необходимо отправлять с каждым запросом. Это может быть application/xml или application/json.

Представления

приложение/xml, приложение/json

Компоненты ресурсов WCMS

Список доступных ресурсов компонентов WCMS.

-Примечание

Примеры URL используются только для презентационных целей. В настроенной среде необходимо заменить адрес сервера на свой собственный.

Ресурс: /{site}/wcms/images/{code}

| | |
|-------------------|--|
| Метод | получать |
| действительность | 5.0.1 |
| Описание | Возвращает сведения об одном CMSImageComponent, указанном кодом. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/wcms/images/код_изображения |
| Параметры запроса | <ul style="list-style-type: none"> • Принять (Стиль: заголовок, Обязательно: true); <p>Это может быть application/xml,application/json, image/*</p> <p>Если заголовок Accept установлен на image/* и компонент не виден, то сервер возвращает ошибку 404 (Не найдено).</p> <p>Если присутствует поле заголовка Accept и сервер не может отправить ответ, который является приемлемым в соответствии с combinedAccept</p> <p>значение поля, то сервер возвращает ошибку 406 (неприемлемо).</p> <p>Если поле заголовка Accept отсутствует, возвращается представление JSON.</p> |
| Представления | <ul style="list-style-type: none"> • Принять (Стиль: заголовок, Обязательно: true); <p>Это может быть application/xml, application/json, image/*</p> |

Ресурс: /{site}/wcms/paragraphs/{code}

| | |
|-------------------|---|
| Метод | получать |
| действительность | 5.0.1 |
| Описание | Возвращает сведения об одном CMSParagraphComponentуказанный код. |
| Пример URL-адреса | https://localhost:9002/rest/v1/electronics/wcms/paragraphs/paragraph_code |
| Параметры запроса | <ul style="list-style-type: none"> • Принимать |

Заголовок должен быть отправлен с каждым запросом. Это может быть application/xml, application/json, text/html

Если заголовок Accept установлен на text/html и компонент не виден, то сервер возвращает ошибку 404 (Not Found).

Если присутствует поле заголовка Accept и если сервер не может отправить ответ, который является приемлемым в соответствии с объединенным Accept

значение поля, то сервер возвращает ошибку 406 (неприемлемо).

Если поле заголовка Accept отсутствует, возвращается представление JSON.

- Принять (Стиль: заголовок, Обязательно: true);

Это может быть application/xml, application/json, text/html

Образцы потоков OCC v1

Примеры вызовов потоков для API SAP Commerce OCC в v1

Обзор

-Примечание

Примеры URL используются только для презентационных целей. В настроенной среде необходимо заменить адрес сервера на свой собственный.

В общем случае запросы REST не имеют состояния, но `commerceweb-service` расширение использует стандартный сеанс HTTP для упрощения вызовов при работе с корзиной клиента. Это тот же подход, что и сеанс конечного пользователя на веб-фронт-энде.

Несмотря на то, что сеанс нужен только для нескольких вызовов, рекомендуется всегда его использовать. Сервер создает новый сеанс и отправляет идентификатор сеанса в ответ на первый запрос. Хорошей практикой является сохранение идентификатора сеанса и использование его в следующих вызовах каждый раз, когда сервер его генерирует. Использование идентификатора сеанса с самого начала также упрощает приложение, поскольку нет необходимости анализировать, следует ли его запоминать или нет.

HTTP-сессия

Сеанс — это последовательность операций (запрос, ответ), выполняемых одним пользователем. Обычно эти операции совместно используют некоторые общие данные, например, корзину клиента или информацию о вошедшем в систему пользователе. Сеанс начинается во время первого запроса и завершается, когда пользователь выходит из системы или после длительного периода бездействия.

Протокол HTTP не учитывает сеансы, и конкретные запросы привязываются к сеансу на основе идентификатора сеанса, который отправляется в виде cookie-файла с каждым запросом.

Каждый раз, когда сервер отправляет новый идентификатор сеанса (cookie JSESSIONID в приложениях Java), клиентское приложение должно запомнить его и использовать в следующих вызовах, в противном случае каждый раз будет создаваться новый сеанс.

Пример:

Конечный пользователь хочет добавить продукт в корзину. Клиентское приложение отправляет запрос:

Запрос №1

ПОСТ `https://localhost:9002/rest/v1/electronics/cart`

Данные POST:

код=3429337

Данные cookie-файлов:

Это первый запрос, поэтому клиентское приложение не имеет никаких куки. Сервер не получает куки JSESSIONID и предполагает, что это новый сеанс. Он создает сеанс и отправляет его идентификатор клиентскому приложению в заголовке Set-Cookie:

Установить-Cookie: JSESSIONID=911CFAE8289BFB29AB347EC95B874EF7; Путь=/rest/; HttpOnly

Клиентское приложение должно запомнить этот файл cookie и повторно отправлять его с последующими запросами на этот сервер. Так, например, если конечный пользователь хочет получить содержимое своей корзины, приложение отправляет запрос:

Запрос №2

ПОЛУЧИТЬ https://localhost:9002/rest/v1/electronics/cart

Данные файлов cookie:

JSESSIONID=911CFAE8289BFB29AB347EC95B874EF7

Сервер получает JSESSIONID и находит соответствующую сессию, содержащую содержимое корзины текущего пользователя, и возвращает ее в качестве ответа.

Атрибуты файлов cookie

Файлы cookie определяются как пара «ключ-значение» с некоторыми дополнительными атрибутами:

Установить-Cookie: JSESSIONID=911CFAE8289BFB29AB347EC95B874EF7; Путь=/rest/; Безопасный; Только HTTP

- <Путь>Атрибут устанавливает 'диапазон видимости' куки. Таким образом, в этом случае куки JSESSIONID будет использоваться только с запросами к URL, начинающимся с /rest.
- <Безопасно>аг указывает, что этот файл cookie может передаваться только по защищенному каналу связи.

Изменение идентификатора сеанса

Идентификатор сеанса может изменяться при изменении нового уровня безопасности, в частности, когда:

- Пользователь входит в систему

Это защита от атаки Session fixation. Протокол

- меняется с HTTPS на HTTP

Это защита от атаки Man-in-the-middle. Когда cookie-файл генерируется при использовании протокола HTTPS, сервер устанавливает в этом cookie-файле флаг Secure.

Затем его нельзя использовать с протоколом HTTP, как описано выше, поэтому в этом случае генерируется новый JSESSIONID. Но в целом, когда приложение переключается на протокол HTTPS, оно не должно переключаться обратно на HTTP до завершения сеанса (выход пользователя и т. д.). Самое простое правило — всегда использовать канал HTTPS.

Постоянное соединение HTTP

По соображениям производительности важно повторно использовать существующие соединения, особенно когда ожидается последующее общение. Это очень важно при использовании протокола HTTPS, поскольку создание безопасного соединения требует больше времени и больше сетевого общения.

При использовании клиента HTTP 1.0, поддерживающего постоянное соединение, отправьте с запросом следующий заголовок:

Соединение: Keep-Alive

Если сервер помещает тот же заголовок в ответ, клиент может использовать это соединение со следующими вызовами.

В HTTP 1.1 соединения по умолчанию считаются постоянными, если не указано иное.

Несмотря на то, что соединение является «постоянным», это не означает, что оно сохраняется вечно. Оно должно быть закрыто клиентом или сервером после периода бездействия.

Множественные проблемы с сайтом

Возможно иметь несколько сайтов, поддерживаемых одним экземпляром сервера. В каждом вызове REST конкретный сайт указывается в URL. Например:

получать https://localhost:9002/rest/v1/electronics/cart

получать https://localhost:9002/rest/v1/apparel-uk/cart

Важно, что некоторые сущности являются общими, а другие разделены в зависимости от местоположения.

Учетные записи пользователей являются общими, поэтому невозможно иметь две учетные записи пользователей с одинаковым логином, привязанные к другим сайтам. Кроме того, пользователь, созданный на одном сайте, сможет использовать ту же учетную запись на других сайтах.

Токен доступа OAuth также связан с пользователем и не зависит от сайта. Вот почему токены доступа получаются с помощью URL, который не содержит идентификатор сайта:

ПОЛУЧАТЬ`https://localhost:9002/authorizationserver/oauth/token`

После получения добавьте токен доступа в заголовок, например:

заголовок «Авторизация: Предъявитель abc738b9-b930-4956-b2f0-f81665f78785»

Затем токен доступа можно использовать с вызовами REST для различных сайтов:

ПОЛУЧАТЬ`https://localhost:9002/rest/v1/electronics/customers/current`

ПОЛУЧАТЬ`https://localhost:9002/rest/v1/apparel-uk/customers/current`

Заказы

История заказов разделена в зависимости от магазина. Например, когда пользователь звонит:

ПОЛУЧАТЬ`https://localhost:9002/rest/v1/electronics/orders?access_token=abc738b9-b930-4956-b2f0-f81665f78785`

Возвращает заказы, размещенные в магазине электроники (связанном с запрошенным сайтом электроники).

Тележки

Корзина пользователя связана с текущим сеансом пользователя. Чтобы это работало правильно на нескольких сайтах, очень важно использовать отдельные сеансы HTTP для каждого сайта.

Чтобы узнать больше об отдельных шагах для сценариев покупки, смотрите: [Отдельные шаги процесса покупки в v1](#).

Чтобы ознакомиться с полными сценариями, посетите: [Сценарии процесса покупки клиента в v1](#).

Отдельные шаги процесса покупки в v1

Примеры отдельных шагов сценариев процесса покупки для Omni Commerce Connect (OCC) в v1.

Обзор

В разделах ниже представлены наборы доступных вызовов, которые могут быть выполнены с использованием API v1. Вызовы были сгруппированы в темы, отражающие их цель, однако их следует рассматривать как отдельные примеры, а не как полные сценарии. Для полных бизнес-сценариев см.: [Сценарии процесса покупки клиента в V1](#).

Зарегистрировать нового клиента

| Метод | URL | Тело (в кодировке URL) |
|-------|---|--|
| почта | <code>https://localhost:9002/authorizationserver/oauth/token</code> | client_id=\$CLIENT_ID\$&client_secret=\$CLIENT_SECRET\$&grant_type=c |
| почта | <code>https://localhost:9002/rest/v1/electronics/customers</code> | логин=john.doe@mail.com &пароль=ваш_пароль &rstName=Джон&lastName=Дой&titleCode=г-н |

Получить токен для клиента

| Метод | URL | Тело (в кодировке URL) |
|-------|--|---|
| ПОЧТА | https://localhost:9002/authorizationserver/oauth/token | client_id=\$CLIENT_ID\$&client_secret=\$CLIENT_SECRET\$&grant_type=пароль&имя пользователя=john.doe@mail.com &пароль=ваш_ |

Управление профилем клиентов

Шаги сценария:

- Получить токен для клиента
- Управление адресами
- Управление платежной информацией
- Обновление профиля клиента
- Изменить пароль клиента

| Метод | URL | Тело (в кодировке URL) |
|----------|--|---------------------------------------|
| ПОЧТА | https://localhost:9002/authorizationserver/oauth/token | client_id=\$КЛИЕНТ_&grant_type=пароль |
| ПОЛУЧАТЬ | https://localhost:9002/rest/v1/electronics/customers/current/addresses | |
| ПОЧТА | https://localhost:9002/rest/v1/electronics/customers/current/addresses | rstName=Джон&las19&город=Осака&по27 |
| ПОМЕЩАТЬ | https://localhost:9002/rest/v1/electronics/customers/current/addresses/default/8796158590999 | |
| ПОЛУЧАТЬ | https://localhost:9002/rest/v1/electronics/customers/current/paymentinfos | сохранено=истина |
| ПОЛУЧАТЬ | https://localhost:9002/rest/v1/electronics/customers/current/paymentinfos/8796191359018 | |
| УДАЛИТЬ | https://localhost:9002/rest/v1/electronics/customers/current/paymentinfos/8796191359018 | |
| ПОМЕЩАТЬ | https://localhost:9002/rest/v1/electronics/customers/current/profile | titleCode=dr&rstNa |

| Метод | URL | Тело (в кодировке URL) |
|----------|---|------------------------|
| помещать | https://localhost:9002/rest/v1/electronics/customers/current/password | старый=ваш_пароль& |

Поиск продукта

| Метод | URL | Параметры | Описание |
|----------|---|--|---|
| получать | https://localhost:9002/rest/v1/electronics/products | запрос=:цена-возрастание:категория:575 | Возвращаться проду от категория с удостоверением личности 575 так к восходить цена. |
| получать | https://localhost:9002/rest/v1/electronics/products | запрос=:имя-описание:бренд:бренд_10 | Возвращаться проду из б с удостоверением личности бренд_ сортirованный спускаться имя. |
| получать | https://localhost:9002/rest/v1/electronics/products | запрос=камера&pageSize=40 | Поиск для 'са проду Мы можем грани the ответ узкий поиск результаты |
| получать | https://localhost:9002/rest/v1/electronics/products | запрос=камера:релевантность:категория:575 | Поиск для 'са проду категория с удостоверением личности 575. |
| получать | https://localhost:9002/rest/v1/electronics/products | запрос=камера:релевантность:категория:575:бренд:бренд_10 | Поиск для 'са проду от категория с удостоверением личности 575 год бренд Идентификатор бренд_ |
| получать | https://localhost:9002/rest/v1/electronics/products | запрос= камера:релевантность:категория:575:бренд:бренд_10:цена:\$500-\$999.99 | Поиск для 'са проду категория с удостоверением личности 575 год бренд Идентификатор бренд_ и ж цена |

| Метод | URL | Параметры | Описание |
|-------|-----|-----------|---------------------------|
| | | | между 500 ан 999.99 |

Поиск ближайшего магазина, в котором есть определенный товар

Шаги сценария:

- Поиск продукта
- Найти ближайший магазин, в котором есть этот товар

| Метод | URL | Параметры | Описание |
|----------|---|----------------------|---|
| получать | https://localhost:9002/rest/v1/electronics/products | запрос=EOS | Поисковые запросы продукты содержащий Строка «EOS» в имя. |
| получать | https://localhost:9002/rest/v1/electronics/products/1382080 | | Получает подробности о «Каноне» EOS 450D'. |
| получать | https://localhost:9002/rest/v1/electronics/products/1382080/nearLocation | местоположение=Токио | Поисковые запросы магазины, имеющие «Canon EOS» 450D' в наличии. Результатом является сортировано по расстояние от указанное местоположение. |

Добавление товара в корзину

-Осторожность

Не забудьте установить JSESSIONID в куки. Он необходим для запросов, которые изменяют корзину. Вы получите его в заголовке Set-Cookie в ответе.

| Метод | URL | Параметры | Описание |
|----------|---|----------------------|---|
| получать | https://localhost:9002/rest/v1/electronics/cart | | Получает пустая корзина. |
| почта | https://localhost:9002/rest/v1/electronics/cart/entry | код=3429337 | Добавляет продукт с код 3429337 к тележка. |
| почта | https://localhost:9002/rest/v1/electronics/cart/entry | код=1934795&кол-во=2 | Добавляет два продукты с кодом 1934795 к тележка. |

| Метод | URL | Параметры | Описание |
|----------|---|-------------------------------------|--|
| ПОЧТА | https://localhost:9002/rest/v1/electronics/cart/entry | code=1934795&qty=1&storeName=Накано | Добавляет один продукт с код 1934795 к тележка. Продукт будет быть забранным в Накано магазин. |
| ПОМЕЩАТЬ | https://localhost:9002/rest/v1/electronics/cart/entry/0 | кол-во=3 | Изменения количество для первый продукт в корзине от 1 до 3. |
| ПОМЕЩАТЬ | https://localhost:9002/rest/v1/electronics/cart/entry/2/store | storeName=Токио | Изменения магазин где продукт будет быть поднятым. |
| УДАЛИТЬ | https://localhost:9002/rest/v1/electronics/cart/entry/2/store | | Сбрасывает магазин где запись должно быть взял. Вход есть доставлено выбраннй доставка метод. |
| УДАЛИТЬ | https://localhost:9002/rest/v1/electronics/cart/entry/0 | | Удаляет первый продукт из тележка. |
| ПОЛУЧАТЬ | https://localhost:9002/rest/v1/electronics/cart | | Получает обновленная корзина. |

Процесс оформления заказа

Оформление заказа с созданием нового адреса и платежной информации для клиента

Предположение: клиент зарегистрирован и имеет корзину с товарами для текущего сеанса.

-Осторожность

Не забудьте установить JSESSIONID в куки. Он необходим для запросов, которые вносят изменения в корзину. Вы получите его в заголовке Set-Cookie в ответе.

Шаги сценария:

- Создайте адрес и установите его в качестве адреса доставки
- Установите режим доставки
- Создать платежную информацию
- Авторизовать корзину
- Разместить заказ

| Метод | URL | Текст запроса |
|----------|---|--|
| ПОЧТА | https://localhost:9002/rest/v1/electronics/customers/current/addresses | |
| ПОМЕЩАТЬ | https://localhost:9002/rest/v1/electronics/cart/address/delivery/8796158787607 | |
| ПОЛУЧАТЬ | https://localhost:9002/rest/v1/electronics/cart/deliverymodes | |
| ПОМЕЩАТЬ | https://localhost:9002/rest/v1/electronics/cart/deliverymodes/premium-gross | |
| ПОЧТА | https://localhost:9002/rest/v1/electronics/cart/authorize | |
| ПОЧТА | https://localhost:9002/rest/v1/electronics/cart/placeorder | |
| ПОЧТА | https://localhost:9002/rest/v1/electronics/cart/paymentinfo | <p>Необходимо установить заголовок Content-Type на: приложение/x-www-form-urlencoded</p> <p>Имя владельца счета: "John Doe" Номер карты: "4111111111111111" Тип карты: "visa" Месяц истечения срока: "01" Год окончания: "2017" сохранено: "истина" defaultPaymentInfo: "true" billingAddress.titleCode: "Мистер" billingAddress.firstName: "Джох" billingAddress.lastName: "Лань" billingAddress.line1: "Тойосак" billingAddress строка2: "16.03.19" billingAddress.postalCode: "53" billingAddress.town: "Osaka" billingAddress.country.isoCode billingAddress.region.isoCode:</p> |

Оформление заказа с указанием текущего адреса и платежной информации для клиента

Предположение: клиент зарегистрирован и имеет корзину с товарами для текущего сеанса.

-Осторожность

Не забудьте установить JSESSIONID в куки. Он необходим для запросов, которые вносят изменения в корзину. Вы получите его в заголовке Set-Cookie в ответе.

Шаги сценария:

- Попробуйте получить способы доставки
- Укажите адрес доставки
- Установить режим доставки
- Установить платежную информацию
- Авторизовать корзину

| Метод | URL | Параметры | Авторизация | Действие |
|----------|---|------------------|-------------------------|---|
| получать | https://localhost:9002/rest/v1/electronics/cart/delivermodes | | OAUTH-токен для клиента | Генерация Повторное введение нет объявление |
| получать | https://localhost:9002/rest/v1/electronics/customers/current/addresses | | OAUTH-токен для клиента | Генерация объявление |
| помещать | https://localhost:9002/rest/v1/electronics/cart/address/delivery/8796158754839 | | OAUTH-токен для клиента | Создание 87 как объявление |
| получать | https://localhost:9002/rest/v1/electronics/cart/delivermodes | | OAUTH-токен для клиента | Генерация Чтение средний |
| помещать | https://localhost:9002/rest/v1/electronics/cart/delivermodes/premium-gross | | OAUTH-токен для клиента | Создание группы мо |
| получать | https://localhost:9002/rest/v1/electronics/customers/current/paymentinfos | сохранено=истина | OAUTH-токен для клиента | Генерация информации взаимодействия |
| помещать | https://localhost:9002/rest/v1/electronics/cart/paymentinfo/8796158591018 | | OAUTH-токен для клиента | Капитализация информации 87 (повторно) пр |
| ПОЧТА | https://localhost:9002/rest/v1/electronics/cart/authorize | securityCode=123 | OAUTH-токен для клиента | Ауторизация |
| ПОЧТА | https://localhost:9002/rest/v1/electronics/cart/placeorder | | OAUTH-токен для клиента | Платеж |

Получайте заказы клиентов

| Метод | URL | Тело (в кодировке URL) |
|----------|---|---|
| ПОЧТА | https://localhost:9002/authorizationserver/oauth/token | client_id=\$CLIENT_ID\$&client_secret=\$CLIENT_SECRET\$ &grant_type=пароль&имя пользователя=john.doe@mail.com &пароль= |
| получать | https://localhost:9002/rest/v1/electronics/orders | |

| Метод | URL | Тело (в кодировке URL) |
|----------|---|------------------------|
| получать | https://localhost:9002/rest/v1/electronics/orders/00001002 | |

Поиск магазина, отображение сведений о магазине

Шаги сценария:

- Поиск магазина
- Показать данные магазина

| Метод | URL | Параметры | Описание |
|----------|---|--------------|--|
| получать | https://localhost:9002/rest/v1/electronics/stores | запрос=Токио | Поиск магазинов, которые находятся недалеко от указанного местоположения параметры (в данном случае рядом с городом Токио). Возвращает список магазинов. |
| получать | https://localhost:9002/rest/v1/electronics/stores/Накано | | Получает сведения о магазине, идентифицированном по названию (в данном случае «Nakano»). |

Восстановление корзины

Шаги сценария:

- Поиск товаров, сеанс «1» создается автоматически
- Получить подробную информацию о продукте
- Добавьте товар(ы) в корзину, корзина «1» будет создана автоматически
- Получить корзину, чтобы показать ее и запомнить руководство по корзине
- Сеанс 1 истекает после длительного бездействия
- Получить корзину, чтобы отобразить ее, сеанс «2» и корзина «2» создаются автоматически
- Попробуйте восстановить корзину «1» до текущего сеанса «2»
- Получить корзину "1"

| Метод | URL | Параметры | Файлы cookie |
|----------|---|---------------------------|--------------|
| получать | https://localhost:9002/rest/v1/electronics/products | запрос=камера&pageSize=40 | |

| Метод | URL | Параметры | Файлы cookie |
|----------|---|------------|--------------------------------|
| получать | https://localhost:9002/rest/v1/electronics/products/489702 | | JSESSIONID получено в шаг 1 |
| почта | https://localhost:9002/rest/v1/electronics/cart/entry | код=489702 | JSESSIONID получено в шаг 1 |
| получать | https://localhost:9002/rest/v1/electronics/cart | | JSESSIONID получено в шаг 1 |
| - | - | - | - |
| получать | https://localhost:9002/rest/v1/electronics/cart | | JSESSIONID получено в шаг 1 |
| получать | https://localhost:9002/rest/v1/electronics/cart/restoreguid=02017b6dad8834ba6f3fd26eb9ab4f270e9b1858 | | JSESSIONID получено в шаг 6 |

| Метод | URL | Параметры | Файлы cookie |
|----------|---|-----------|-----------------------------------|
| получать | https://localhost:9002/rest/v1/electronics/cart | | JSESSIONID получено в шаг 7 |

Сценарии процесса покупки клиента в v1

Ниже приведены полные сценарии процесса покупки для клиентов для Omni Commerce Connect (OCC) в версии 1.

Обзор

Приведенные ниже сценарии представляют собой полное пошаговое руководство по использованию заданной последовательности вызовов.

Сценарий зарегистрированного клиента

Предварительные условия: покупатель уже зарегистрирован в магазине и указал всю информацию, необходимую для оформления заказа (например, адрес и платежную информацию).

-Осторожность

Не забудьте установить JSESSIONID в куки. Он необходим для запросов, которые вносят изменения в корзину. Вы получаете его в заголовке Set-Cookie в ответе.

Шаги сценария:

- Поиск продукта
 - Получить подробную информацию о продукте
- Добавить товар в корзину Получить
- токен доступа для покупателя
- Оформить заказ
- Показать размещенный заказ для клиента
- Выйти

-Примечание

Вызовы должны быть выполнены в порядке, определенном в таблице request flow. Прокрутите вправо, чтобы увидеть полное содержимое таблицы.

| Метод | URL | Тело (в кодировке URL) |
|----------|---|---------------------------|
| получать | https://localhost:9002/rest/v1/electronics/products | запрос=камера&pageSize=40 |
| получать | https://localhost:9002/rest/v1/electronics/products/489702 | |
| Почта | https://localhost:9002/rest/v1/electronics/cart/entry | код=489702 |

| Метод | URL | Тело (в кодировке URL) |
|----------|---|--|
| | | |
| ПОЧТА | https://localhost:9002/authorizationserver/oauth/token | client_id=\$CLIENT_ID\$&client_secret=\$ &grant_type=пароль&имя пользователя=джон |
| ПОЛУЧАТЬ | https://localhost:9002/rest/v1/electronics/cart | |
| ПОЛУЧАТЬ | https://localhost:9002/rest/v1/electronics/customers/current/addresses | |
| ПОМЕЩАТЬ | https://localhost:9002/rest/v1/electronics/cart/address/delivery/8796158754839 | |
| ПОЛУЧАТЬ | https://localhost:9002/rest/v1/electronics/cart/delivermodes | |
| ПОМЕЩАТЬ | https://localhost:9002/rest/v1/electronics/cart/delivermodes/premium-gross | |
| ПОЛУЧАТЬ | https://localhost:9002/rest/v1/electronics/customers/current/paymentinfos | сохранено=истина |
| ПОМЕЩАТЬ | https://localhost:9002/rest/v1/electronics/cart/paymentinfo/8796158591018 | |
| ПОЧТА | https://localhost:9002/rest/v1/electronics/cart/authorize | securityCode=123 |
| ПОЧТА | https://localhost:9002/rest/v1/electronics/cart/placeorder | |
| ПОЛУЧАТЬ | https://localhost:9002/rest/v1/electronics/orders | |
| ПОЛУЧАТЬ | https://localhost:9002/rest/v1/electronics/orders/00001002 | |
| ПОЧТА | https://localhost:9002/rest/v1/customers/current/logout | |

Сценарий нового клиента

Предварительные условия: покупатель не зарегистрирован в магазине.

-Осторожность

Не забудьте установить JSESSIONID в куки. Он необходим для запросов, которые вносят изменения в корзину. Вы получите его в заголовке Set-Cookie в ответе.

Шаги сценария:

- Зарегистрируйте нового клиента и получите токен доступа
- Создать новый адрес для клиента
- Поиск определенного продукта
 - Получить подробную информацию о продукте
- Добавить товары в корзину
- Оформление заказа: установка адреса доставки, установка способа доставки, установка данных кредитной карты (paymentinfo), авторизация карты
- Разместить заказ
- Отобразить размещенный заказ для клиента
- Выйти

-Примечание

Вызовы должны быть выполнены в порядке, определенном в таблице request low. Прокрутите вправо, чтобы увидеть полное содержимое таблицы.

| Метод | URL | Текст запроса |
|----------|---|---------------|
| ПОЧТА | https://localhost:9002/authorizationserver/oauth/token | |
| ПОЧТА | https://localhost:9002/rest/v1/electronics/customers | |
| ПОЧТА | https://localhost:9002/authorizationserver/oauth/token | |
| ПОЧТА | https://localhost:9002/rest/v1/electronics/customers/current/addresses | |
| ПОЛУЧАТЬ | https://localhost:9002/rest/v1/electronics/products | |
| ПОЛУЧАТЬ | https://localhost:9002/rest/v1/electronics/products /3429337 | |
| ПОЧТА | https://localhost:9002/rest/v1/electronics/cart/entry | |

| Метод | URL | Текст запроса |
|----------|---|---|
| помещать | https://localhost:9002/rest/v1/electronics/cart/address/delivery/8796158754839 | |
| получать | https://localhost:9002/rest/v1/electronics/cart/deliverymodes | |
| помещать | https://localhost:9002/rest/v1/electronics/cart/deliverymodes/premium-gross | |
| почта | https://localhost:9002/rest/v1/electronics/cart/authorize | |
| почта | https://localhost:9002/rest/v1/electronics/cart/placeorder | |
| получать | https://localhost:9002/rest/v1/electronics/orders | |
| получать | https://localhost:9002/rest/v1/electronics/orders/00001002 | |
| почта | https://localhost:9002/rest/v1/customers/current/logout | |
| почта | https://localhost:9002/rest/v1/electronics/cart/paymentinfo | <p>Необходимо установить Head Content-Type на: приложение/x-www-form-urlencoded</p> <p>Имя владельца счета: "John Doe" Номер карты: "4111111111111111" Тип карты: "visa" Месяц истечения срока: "01" Год окончания: "2017" сохранено: "истина" defaultPaymentInfo: "true" billingAddress.titleCode: "Мистер" billingAddress.firstName: "Джох" billingAddress.lastName: "Лань" billingAddress.line1: "Тойосак" billingAddress строка2: "16.03.19" billingAddress.postalCode: "53" billingAddress.town: "Osaka" billingAddress.country.isocode billingAddress.region.isocode:</p> |

Сценарий получения в магазине

Предварительные условия: Клиент уже зарегистрирован в магазине и указал всю информацию, необходимую для оформления заказа (например, платежную информацию).

-Осторожность

Не забудьте установить JSESSIONID в куки. Он необходим для запросов, которые вносят изменения в корзину. Вы получите его в заголовке Set-Cookie в ответе.

Шаги сценария:

- Получите токен доступа для клиента
- Добавьте товары, которые будут забраны в магазине. Оформите заказ и установите «самовывоз» в качестве способа доставки.
- Отобразить размещенный заказ для клиента
- Выйти

-Примечание

Вызовы должны быть выполнены в порядке, определенном в таблице request flow. Прокрутите вправо, чтобы увидеть полное содержимое таблицы.

| Метод | URL | Параметры тела |
|----------|---|---|
| | | Тип содержимого: application/x-www-form-urlencoded |
| ПОЧТА | https://localhost:9002/authorizationserver/oauth/token | client_id=\$CLIENT_ID\$&client_secret=\$CLIENT_SECRET\$ |
| ПОЧТА | https://localhost:9002/rest/v1/electronics/cart/entry | code=1382080 |
| ПОМЕЩАТЬ | https://localhost:9002/rest/v1/electronics/cart/entry/0/store | storeName=Накано |
| ПОЧТА | https://localhost:9002/rest/v1/electronics/cart/entry | code=1934795&qty=1&storeName=Накано |
| ПОЛУЧАТЬ | https://localhost:9002/rest/v1/electronics/cart | |
| ПОЛУЧАТЬ | https://localhost:9002/rest/v1/electronics/cart/deliverymodes | |
| ПОМЕЩАТЬ | https://localhost:9002/rest/v1/electronics/cart/deliverymodes/pickup | |
| ПОЛУЧАТЬ | https://localhost:9002/rest/v1/electronics/customers/current/paymentinfos | сохранено=истина |
| ПОМЕЩАТЬ | https://localhost:9002/rest/v1/electronics/cart/paymentinfo/8796158591018 | |
| ПОЧТА | https://localhost:9002/rest/v1/electronics/cart/authorize | securityCode=123 |
| ПОЧТА | https://localhost:9002/rest/v1/electronics/cart/placeorder | |

| Метод | URL | Параметры тела |
|----------|---|--|
| получать | https://localhost:9002/rest/v1/electronics/orders | Тип содержимого: application/x-www-form-urlencoded |
| получать | https://localhost:9002/rest/v1/electronics/orders/00002036 | |
| почта | https://localhost:9002/rest/v1/customers/current/logout | |

Самовывоз из магазина с объединением магазинов - Сценарий

Предпосылки:

- Клиент уже зарегистрирован в магазине и указал всю информацию, необходимую для оформления заказа (например, адрес и платежную информацию).
- установлен acceleratorwebservicesaddon

-Осторожность

Не забудьте установить JSESSIONID в куки. Он необходим для запросов, которые вносят изменения в корзину. Вы получите его в заголовке Set-Cookie в ответе.

Шаги сценария:

- Получить токен доступа для клиента
- Добавьте товар, который можно будет забрать в магазине
- Добавьте товар, который будет забран в другом магазине.
- Используйте функцию консолидации магазина.
- Проверить
- Выйти

-Примечание

Вызовы должны быть выполнены в порядке, определенном в таблице request low. Прокрутите вправо, чтобы увидеть полное содержимое таблицы.

| Метод | URL | Параметры тела |
|----------|---|---|
| получать | https://localhost:9002/authorizationserver/oauth/token | client_id=\$CLIENT_ID\$&client_secret=\$CLIENT_SECRET\$ |
| получать | https://localhost:9002/rest/v1/electronics/cart/entry | code=2053226&qty=1&storeName=Shinbashi |
| получать | https://localhost:9002/rest/v1/electronics/cart/entry | code=1934793&qty=1&storeName=Накано |
| получать | https://localhost:9002/rest/v1/electronics/cart | |

| Метод | URL | Параметры тела |
|----------|---|--|
| | | Тип содержимого: application/x-www-form-urlencoded |
| получать | https://localhost:9002/rest/v1/electronics/cart/consolidate | |
| почта | https://localhost:9002/rest/v1/electronics/cart/consolidate | storeName=Шинбashi |
| получать | https://localhost:9002/rest/v1/electronics/cart/deliverymodes | |
| помещать | https://localhost:9002/rest/v1/electronics/cart/deliverymodes/pickup | |
| получать | https://localhost:9002/rest/v1/electronics/customers/current/paymentinfos | сохранено=истина |
| помещать | https://localhost:9002/rest/v1/electronics/cart/paymentinfo/8796158591018 | |
| почта | https://localhost:9002/rest/v1/electronics/cart/authorize | securityCode=123 |
| почта | https://localhost:9002/rest/v1/electronics/cart/placeorder | |
| почта | https://localhost:9002/rest/v1/customers/current/logout | |

Совмещение способа доставки с самовывозом из магазина — сценарий

Предварительные условия: Клиент уже зарегистрирован в магазине и указал всю информацию, необходимую для оформления заказа (например, адрес и платежную информацию).

-Осторожность

Не забудьте установить JSESSIONID в куки. Он необходим для запросов, которые вносят изменения в корзину. Вы получите его в заголовке Set-Cookie в ответе.

Шаги сценария:

- Получите токен доступа для клиента
- Добавьте продукт, который будет доставлен
- Добавьте товар, который можно будет забрать в магазине
- Проверить

- Отображение размещенного заказа для клиента
- Выйти

-Примечание

Вызовы должны быть выполнены в порядке, определенном в таблице request flow. Прокрутите вправо, чтобы увидеть полное содержимое таблицы.

| Метод | URL | Параметры тела |
|----------|---|---|
| | | Тип содержимого: application/x-www-form-urlencoded |
| получать | https://localhost:9002/authorizationserver/oauth/token | client_id=\$CLIENT_ID\$&client_secret=\$CLIENT_SECRET\$ |
| получать | https://localhost:9002/rest/v1/electronics/cart/entry | code=1382080 |
| получать | https://localhost:9002/rest/v1/electronics/cart/entry | code=1934795&qty=1&storeName=Nak |
| получать | https://localhost:9002/rest/v1/electronics/cart | |
| получать | https://localhost:9002/rest/v1/electronics/customers/current/addresses | |
| помещать | https://localhost:9002/rest/v1/electronics/cart/address/delivery/8796192014359 | |
| получать | https://localhost:9002/rest/v1/electronics/cart/deliverymodes | |
| помещать | https://localhost:9002/rest/v1/electronics/cart/deliverymodes/premium-gross | |
| получать | https://localhost:9002/rest/v1/electronics/customers/current/paymentinfos | сохранено=истина |
| помещать | https://localhost:9002/rest/v1/electronics/cart/paymentinfo/8796158591018 | |
| получать | https://localhost:9002/rest/v1/electronics/cart/authorize | securityCode=123 |
| получать | https://localhost:9002/rest/v1/electronics/cart/placeorder | |
| получать | https://localhost:9002/rest/v1/electronics/orders | |

| Метод | URL | Параметры тела |
|----------|---|---|
| получать | https://localhost:9002/rest/v1/electronics/orders/00002036 | Тип содержимого: application/x-www-form |
| почта | https://localhost:9002/rest/v1/customers/current/logout | |

Сценарий оформления заказа гостем с созданием полной учетной записи

-Осторожность

Не забудьте установить JSESSIONID в куки. Он необходим для запросов, которые вносят изменения в корзину. Вы получите его в заголовке Set-Cookie в ответе.

Шаги сценария:

- Поиск продукта
 - Получить подробную информацию о продукте
- Добавить товар в корзину
- Получить токен доступа клиента
- Представьтесь как гость
- Проверить
 - Отображение размещенного заказа
- Преобразовать гостевую учетную запись в полноценную учетную запись клиента (по желанию)
- Войти как клиент
 - Получить список заказов

-Примечание

Вызовы должны быть выполнены в порядке, определенном в таблице request low. Прокрутите вправо, чтобы увидеть полное содержимое таблицы.

| Метод | URL | Параметры тела |
|----------|---|--|
| получать | https://localhost:9002/rest/v1/electronics/products | Тип содержимого: application/x-www-form |
| получать | https://localhost:9002/rest/v1/electronics/products /489702 | |
| почта | https://localhost:9002/rest/v1/electronics/cart/entry | |
| почта | https://localhost:9002/authorizationserver/oauth/token | client_id=\$CLIENT_ID\$&client_secret=\$ |
| почта | https://localhost:9002/rest/v1/electronics/customers/current/guestlogin | |

| Метод | URL | Параметры тела |
|----------|---|--|
| | | Тип содержимого: application/x-www-form |
| ПОЧТА | https://localhost:9002/rest/v1/electronics/customers/current/addresses | |
| ПОМЕЩАТЬ | https://localhost:9002/rest/v1/electronics/cart/address/delivery/8796158754839 | |
| ПОЛУЧАТЬ | https://localhost:9002/rest/v1/electronics/cart/delivermodes | |
| ПОМЕЩАТЬ | https://localhost:9002/rest/v1/electronics/cart/delivermodes/premium-gross | |
| ПОЧТА | https://localhost:9002/rest/v1/electronics/cart/authorize | securityCode=123 |
| ПОЧТА | https://localhost:9002/rest/v1/electronics/cart/placeorder | |
| ПОЧТА | https://localhost:9002/rest/v1/electronics/customers/current/convert | пароль=ваш_пароль |
| ПОЧТА | https://localhost:9002/rest/v1/customers/current/logout | |
| | ... | |
| ПОЧТА | https://localhost:9002/authorizationserver/oauth/token | client_id=\$CLIENT_ID\$&client_secret=\$ &grant_type=пароль&имя пользователя=джон |
| ПОЛУЧАТЬ | https://localhost:9002/rest/v1/electronics/orders | |
| ПОЛУЧАТЬ | https://localhost:9002/rest/v1/electronics/orders/00002009 | |
| ПОЧТА | https://localhost:9002/rest/v1/electronics/cart/paymentinfo | Имя владельца счета: "John Doe" Номер карты: "4111111111111111" Тип карты: "visa" Месяц истечения срока: "01" Год окончания: "2017" сохранено: "истина" defaultPaymentInfo: "true" billingAddress.titleCode: "Мистер" billingAddress.firstName: "Джо" billingAddress.lastName: "Doe" billingAddress.line1: "Toyosak" |

| Метод | URL | Параметры тела |
|-------|-----|--|
| | | Тип содержимого: application/x-www-form billingAddress.строка2: "3-16-19" billingAddress.почтовый индекс: "53" billingAddress.город: "Осака" billingAddress.страна.isocode billingAddress.регион.isocode: |

Сценарий оформления заказа гостем с проверкой статуса заказа

-Осторожность

Не забудьте установить JSESSIONID в куки. Он необходим для запросов, которые вносят изменения в корзину. Вы получите его в заголовке Set-Cookie в ответе.

Шаги сценария:

- Поиск продукта
- Получить сведения о продукте
- Добавить продукт в корзину
- Получить токен доступа клиента
- Представьтесь гостем
- Проверить
- Выйти
- Получить информацию о заказе

-Примечание

Вызовы должны быть выполнены в порядке, определенном в таблице request low. Прокрутите вправо, чтобы увидеть полное содержимое таблицы.

| Метод | URL | Параметры тела | Тип контента |
|----------|---|----------------|----------------|
| получать | https://localhost:9002/rest/v1/electronics/products | | |
| получать | https://localhost:9002/rest/v1/electronics/products/489702 | | |
| ПОЧТА | https://localhost:9002/rest/v1/electronics/cart/entry | | |
| ПОЧТА | https://localhost:9002/authorizationserver/oauth/token | | клиент_id=\$CL |
| ПОЧТА | https://localhost:9002/rest/v1/electronics/customers/current/guestlogin | | |
| ПОЧТА | https://localhost:9002/rest/v1/electronics/customers/current/addresses | | |

| Метод | URL | Параметры тела |
|----------|---|---|
| | | Тип контента |
| | | |
| помещать | https://localhost:9002/rest/v1/electronics/cart/address/delivery/8796158754839 | |
| получать | https://localhost:9002/rest/v1/electronics/cart/deliverymodes | |
| помещать | https://localhost:9002/rest/v1/electronics/cart/deliverymodes/premium-gross | |
| почта | https://localhost:9002/rest/v1/electronics/cart/authorize | securityCode= |
| почта | https://localhost:9002/rest/v1/electronics/cart/placeorder | |
| почта | https://localhost:9002/rest/v1/customers/current/logout | |
| | ... | |
| получать | https://localhost:9002/rest/v1/electronics/orders/byGuid/00aa0ba88c9c7c93b886423e354556dc21c430e0 | |
| почта | https://localhost:9002/rest/v1/electronics/cart/paymentinfo | аккаунтНо Номер карты Тип карты: срок действияMont срок действияДа сохранено: "т defaultPay биллингДобавить биллингДобавить биллингДобавить биллингДобавить биллингДобавить биллингДобавить биллингДобавить биллингДобавить биллингДобавить биллингДобавить |

Сценарий использования ваучера

Предварительные условия: В системе определен ваучер с кодом HY2008.

-Осторожность

Не забудьте установить JSESSIONID в куки. Он необходим для запросов, которые вносят изменения в корзину. Вы получите его в заголовке Set-Cookie в ответе.

Шаги сценария:

- Добавить товары в корзину
- Применить ваучер
- Получить корзину
- Зарегистрируйте нового клиента и получите для него токен доступа
- Создайте новый адрес для клиента
- Оформление заказа: установка адреса доставки, установка способа доставки, установка данных кредитной карты (платежная информация), авторизация карты
- Разместить заказ
- Отображение размещенных заказов для клиента
- Выйти

-Примечание

Вызовы должны быть выполнены в порядке, определенном в таблице request flow. Прокрутите вправо, чтобы увидеть полное содержимое таблицы.

| Метод | URL | Параметры тела |
|----------|---|---|
| | | Тип контента: приложение/x-www- |
| ПОЧТА | https://localhost:9002/rest/v1/electronics/cart/entry | |
| ПОЛУЧАТЬ | https://localhost:9002/rest/v1/electronics/cart | |
| ПОЧТА | https://localhost:9002/authorizationserver/oauth/token | client_id=\$CLIENT_ID\$&client_secret=\$ |
| ПОЧТА | https://localhost:9002/rest/v1/electronics/cart/voucher/HY2008 | |
| ПОЛУЧАТЬ | https://localhost:9002/rest/v1/electronics/cart | |
| ПОЧТА | https://localhost:9002/rest/v1/electronics/customers | логин=john.doe%40mail.com&пароль &rstName=Джон&lastName=Дой&titleC |
| ПОЧТА | https://localhost:9002/authorizationserver/oauth/token | client_id=\$CLIENT_ID\$&client_secret=\$ |
| ПОЧТА | https://localhost:9002/rest/v1/electronics/customers/current/addresses | |

| Метод | URL | Параметры тела |
|----------|---|--------------------------------|
| | | Тип контента:приложение/x-www- |
| помещать | https://localhost:9002/rest/v1/electronics/cart/address/delivery/8796192014359 | |
| получать | https://localhost:9002/rest/v1/electronics/cart/deliverymodes | |
| помещать | https://localhost:9002/rest/v1/electronics/cart/deliverymodes/premium-gross | |
| получать | https://localhost:9002/rest/v1/electronics/customers/current/paymentinfos | |
| помещать | https://localhost:9002/rest/v1/electronics/cart/paymentinfo/8796158591018 | |
| ПОЧТА | https://localhost:9002/rest/v1/electronics/cart/authorize | securityCode=123 |
| ПОЧТА | https://localhost:9002/rest/v1/electronics/cart/placeorder | |
| получать | https://localhost:9002/rest/v1/electronics/orders | |
| получать | https://localhost:9002/rest/v1/electronics/orders/00002036 | |
| ПОЧТА | https://localhost:9002/rest/v1/customers/current/logout | |

| Метод | URL | Параметры тела |
|-------|---|---|
| | | Тип контента: приложение/x-www- |
| почта | https://localhost:9002/rest/v1/electronics/cart/paymentinfo | Имя владельца счета: "John Doe" Номер карты: "4111111111111111" Тип карты: "visa" Месяц истечения срока: "01" Год окончания: "2017" сохранено: "истина" defaultPaymentInfo: "true" billingAddress.titleCode: "Мистер" billingAddress.firstName: "Джох" billingAddress.lastName: "Лань" billingAddress.line1: "Тойосак" billingAddress строка2: "16.03.19" billingAddress.postalCode: "53" billingAddress.town: "Osaka" billingAddress.country.isocode billingAddress.region.isocode: |

Ответы об ошибках от OCC v1

Ошибочные ответы, возвращаемые коммерческими сервисами расширение.

Форматы ошибок

Ответы об ошибках, используемые в веб-службах OCC, имеют унифицированный формат, который может описывать отдельную ошибку или список ошибок.

| Атрибут ошибки | Обязательный? | Описание |
|----------------|---------------|---|
| <тип> | Да | Тип ошибки, например: AccessDeniedError, Ошибка проверки, и т. д. |
| <сообщение> | Да | Текстовое сообщение с описанием ошибки. |
| <subjectType> | Нет | Тип субъекта, например: параметр, сайт |
| <тема> | Нет | Идентификатор субъекта, например: <фамилия>, <электроника> |
| <причина> | Нет | Код причины ошибки, например: недействительный, отсутствует, <неизвестныйИдентификатор> |

Форматы ответа

Формат ошибки XML

Пример ошибки в формате XML:

```

<ошибки>
  <ошибка>
    <сообщение>...</сообщение>
    <причина>...</причина>
    <субъект>...</субъект>
    <subjectType>...</subjectType> <type>...</type>
  </ошибка>
  <ошибка>
  ...

```

</ошибки>

...

</ошибки>

Формат ошибки JSON

Пример ошибки в формате JSON:

```
{
  "ошибки": [
    {
      "сообщение": "...",
      "причина": "...",
      "предмет": "...",
      "ТипТемы": "...",
      "тип": "..."
    },
    {
      "сообщение": "...",
      "причина": "...",
      "предмет": "...",
      "ТипТемы": "...",
      "тип": "..."
    }
  ]
}
```

Стандартные ошибки

Ошибки аутентификации и авторизации

| Ответ Статус | Случай | Тело ответа |
|-------------------------------------|--|--|
| HTTP/1.1 401 Несанкционированный | <p>Когда запрос требует аутентификации пользователя или у пользователя нет прав на ресурс.</p> <p>Примеры:</p> <ul style="list-style-type: none"> ПОЛУЧАТЬ -https://localhost:9002/rest/v1/wsTest/customers/current/addresses без токена авторизации ПОЛУЧАТЬ -https://localhost:9002/rest/v1/wsTest/customers/current/addresses с токеном авторизации клиента | <?xml версия="1.0" ru <ошибки> <ошибка> <сообщение>Доступ <type>ДоступDe </ошибка> </ошибки> |
| HTTP/1.1 401 Несанкционированный | <p>Отсутствует параметр client_id в запросе токена.</p> <p>Пример: ПОСТ -https://localhost:9002/authorizationserver/oauth/token</p> <p>Заголовок:</p> <p>Тип содержимого: application/x-www-form-urlencoded</p> <p>Текст запроса:</p> <p>client_secret=секрет&grant_type=учетные_данные_клиента</p> | <?xml версия="1.0" ru <ошибки> <ошибка> <сообщение> Ау <type>Отменить авторство </ошибка> </ошибки> |
| HTTP/1.1 401 Несанкционированный | <p>Неверный client_id в запросе токена.</p> <p>Пример: ПОСТ -https://localhost:9002/authorizationserver/oauth/token</p> <p>Заголовок:</p> <p>Тип содержимого: application/x-www-form-urlencoded</p> <p>Текст запроса:</p> <p>client_id=неправильный_идентификатор_клиента&client_secret=секрет&grant_type=учетные_данные_клиента</p> | <?xml версия="1.0" ru <ошибки> <ошибка> <сообщение>Нет КЛ <type>Отменить авторство </ошибка> </ошибки> |
| HTTP/1.1 401 Несанкционированный | <p>Неправильный client_secret в запросе токена.</p> <p>Пример: ПОСТ -https://localhost:9002/authorizationserver/oauth/token</p> <p>Заголовок:</p> <p>Тип содержимого: application/x-www-form-urlencoded</p> | <?xml версия="1.0" ru <ошибки> <ошибка> <сообщение>Плохо С <тип>BadClien |

| Ответ | Статус | Тело ответа |
|-------------------------------------|--|--|
| | Текст запроса: client_id=mobile_android&client_secret=wrong_secret&grant_type=client_credentials | </ошибка> </ошибки> |
| HTTP/1.1 401 Несанкционированный | Когда токен авторизации неверен. | <?xml версия="1.0" ru <ошибки> <ошибка> <сообщение>Недействительно <тип>Недействительный </ошибка> </ошибки> |
| HTTP/1.1 401 Несанкционированный | Когда срок действия токена истек. | <?xml версия="1.0" ru <ошибки> <ошибка> <сообщение>Доступ <тип>Недействительный </ошибка> </ошибки> |
| HTTP/1.1 400 Плохо Запрос | Неправильный тип гранта в запросе токена. Пример: ПОСТ - https://localhost:9002/authorizationserver/oauth/token Заголовок: Тип содержимого: application/x-www-form-urlencoded Текст запроса: client_id=mobile_android&client_secret=secret&grant_type=неверные_учетные_данные | <?xml версия="1.0" ru <ошибки> <ошибка> <сообщение>Отказ <тип>Отменить поддержку </ошибка> </ошибки> |
| HTTP/1.1 400 Плохо Запрос | Отсутствует тип гранта в запросе токена. Пример: ПОСТ - https://localhost:9002/authorizationserver/oauth/token Заголовок: Тип содержимого: application/x-www-form-urlencoded Текст запроса: client_id=mobile_android&client_secret=секрет | <?xml версия="1.0" ru <ошибки> <ошибка> <сообщение>Мисси <тип>НедействительныйР </ошибка> </ошибки> |
| HTTP/1.1 400 Плохо Запрос | Отсутствующие или неверные учетные данные клиента в запросе токена клиента. Пример: ПОСТ - https://localhost:9002/authorizationserver/oauth/token Заголовок: Тип содержимого: application/x-www-form-urlencoded Текст запроса: client_id=mobile_android&client_secret=secret&grant_type=пароль | <?xml версия="1.0" ru <ошибки> <ошибка> <сообщение>Плохо с <тип>НеверныйG </ошибка> </ошибки> |

Ошибки неправильного запроса

| Ответ | Статус | Тело ответа |
|-------|--|--|
| | Когда в запросе отсутствует обязательный параметр. Пример: ПОЛУЧИТЬ - https://localhost:9002/rest/v1/wsTest/products/export/incremental без обязательного параметра временной метки | <?xml версия="1.0" кодировка="UTF-16"?> <ошибки> <ошибка> <message>Обязательный строковый параметр <тип>MissingServletRequestParam </error> </ошибка> |

| Ответ Статус | Случай | Тело ответа |
|---|--|---|
| HTTP/1.1 404 Нет Найденный | <p>Когда ресурс не существует.</p> <p>Пример: ПОЛУЧИТЬ - https://localhost:9002/rest/v1/electronics/methodDoesNotExist</p> | <pre><?xml версия='1.0' кодировка='UTF-8'?> <ошибки> <ошибка> <message>Нет ресурса для ра</message> <type>UnknownResourceError</type> </ошибка></pre> |
| HTTP/1.1 405 Метод Нет Допустимый | <p>Когда тип метода HTTP недопустим.</p> <p>Пример: ПУТ -https://localhost:9002/rest/v1/electronics/products</p> | <pre><?xml версия="1.0" кодировка="UTF-16"?> <ошибки> <ошибка> <message>Метод запроса 'PUT' не</message> <type>HttpRequestMethodNotSupported</type> </ошибка></pre> |

Конкретные ошибки hybris

Общие ошибки

| Ответ Статус | Ошибка | Случай |
|---------------------------------|-----------------|--|
| HTTP/1.1 400 Плохо Запрос | Ошибка проверки | <p>Общий случай:</p> <p>Когда нет обязательного параметра или параметр имеет неверное значение. Примеры ошибок приведены ниже.</p> <hr/> <p>Если в запросе на добавление адреса отсутствует параметр или указан неверный параметр.</p> <p>Пример: ПОСТ -https://localhost:9002/rest/v1/electronics/customers/current/address</p> <p>Заголовок:</p> <p>Тип содержимого: application/x-www-form-urlencoded</p> <p>Текст запроса:</p> <pre>line1=Тойосаки&line2=3-16-19&town=Osaka&postalCode=531-0072&country.isoCode=JP</pre> <hr/> <p>Если в запросе на добавление платежной информации отсутствует параметр или указан неверный параметр.</p> <p>Пример: ПОСТ -https://localhost:9002/rest/v1/electronics/cart/paymentinfo</p> <p>Заголовок:</p> <p>Тип содержимого: application/x-www-form-urlencoded</p> <p>Текст запроса:</p> |

| Ответ Статус | Ошибка | Случай |
|-----------------|--------|--|
| | | <p>accountHolderName=John+Doe&cardNumber=4111111111111111&expiryMonth=01&billingAddress=16-19&billingAddress.postalCode=531-0072&billingAddress.town=Osaka&billingAddress</p> |
| | | <p>Если в запросе на добавление отзыва отсутствует параметр или указан неверный параметр.</p> <p>Пример: ПОСТ -https://localhost:9002/rest/v1/electronics/products/872912/reviews</p> |
| | | <p>Когда параметр storeName неверен (например, магазин не существует) для записи о самовывозе из магазина. Пример:</p> <ul style="list-style-type: none"> • ПОЧТА -https://localhost:9002/rest/v1/electronics/cart/entry?code=489702&store • ПОМЕЩАТЬ -https://localhost:9002/rest/v1/electronics/cart/entry/0/store?storeName= |
| | | <p>Когда в базе данных нет объекта страны с указанным кодом.</p> <p>Пример: ПОСТ -https://localhost:9002/rest/v1/electronics/customers/current/address</p> <p>Заголовок:</p> <p>Тип содержимого: application/x-www-form-urlencoded</p> <p>Текст запроса:</p> <p>firstName=Джон&lastName=Доу&titleCode=г-н&line1=Тойосаки&line2=3-16-19&town=Осака&p</p> |
| | | <p>Когда параметр входа имеет неверное значение.</p> <p>Пример: ПОСТ -https://localhost:9002/rest/v1/electronics/customers</p> <p>Заголовок:</p> <p>Тип содержимого: application/x-www-form-urlencoded</p> <p>Текст запроса:</p> |

| Ответ Статус | Ошибка | Случай |
|---------------------------------|----------------------------------|--|
| | | <p>login=НЕПРАВИЛЬНЫЙ_ЛОГИН&password=ваш_пароль&firstName=Джон&lastName=Доу&titleCode=г-н</p> <hr/> <p>Если параметр типа акции указан неверно.</p> <p>Пример: ПОЛУЧИТЬ -https://localhost:9002/rest/v1/electronics/promotions?type=unknownType</p> <hr/> <p>Когда параметр временной метки имеет неверный формат.</p> <p>Пример: GET - https://localhost:9002/rest/v1/electronics/products/expressUpdate?timestamp=2013-06-21</p> |
| HTTP/1.1 400 Плохо Запрос | НедопустимыйResourceError | <p>Когда в запросе указан неверный идентификатор базового сайта.</p> <p>Пример: ПОЛУЧИТЬ -https://localhost:9002/rest/v1/wrongBaseSite/titles</p> |
| HTTP/1.1 400 Плохо Запрос | НеоднозначныйИдентификаторОшибка | <p>Когда имеется более одного ресурса с указанным идентификатором.</p> <p>Пример: ПОЛУЧИТЬ -https://localhost:9002/rest/v1/electronics/products/ 12345</p> |
| HTTP/1.1 400 Плохо Запрос | НеизвестныйИдентификаторОшибка | <p>Общий случай: когда ресурс с указанным идентификатором не существует.</p> <hr/> <p>Если товара с указанным кодом не существует.</p> <p>Пример: ПОЛУЧИТЬ -https://localhost:9002/rest/v1/electronics/products/13820867876 и пр</p> <hr/> <p>Если код заголовка неверен.</p> <p>Пример: ПОСТ -https://localhost:9002/rest/v1/electronics/customers/current/addresse</p> <p>Заголовок:</p> <p>Тип содержимого: application/x-www-form-urlencoded</p> <p>Текст запроса:</p> |

| Ответ Статус | Ошибка | Случай |
|---------------------------------|------------------------|---|
| | | <p>firstName=Джон&lastName=Доу&titleCode=НЕВЕРНЫЙ_ID&line1=Тойосаки&line2=3-16-19&town=О</p> <p>Когда магазин с указанным идентификатором не существует.</p> <p>Пример: ПОЛУЧИТЬ -https://localhost:9002/rest/v1/electronics/stores/НЕВЕРНЫЙ_ИДЕНТИФИКАТОР</p> |
| HTTP/1.1 400 Плохо Запрос | МодельСохранениеОшибка | <p>При возникновении ошибки при сохранении ресурса в базе данных.</p> |

Ошибки клиентов ресурсов

| Ответ Статус | Ошибка | Случай |
|---------------------------------|------------------------------|--|
| HTTP/1.1 400 Плохо Запрос | DuplicateUidError | <p>Когда невозможно создать клиента, поскольку указанный логин уже существует.</p> <p>Пример: ПОСТ -https://localhost:9002/rest/v1/electronics/customers</p> <p>Заголовок:</p> <p>Тип содержимого: application/x-www-form-urlencoded</p> <p>Текст запроса:</p> <p>login=test1@test.com &password=your_password&firstName=John&lastName=Doe&titleCode=mr</p> <p>когда клиент с логином test1@test.com уже существует</p> |
| HTTP/1.1 400 Плохо Запрос | Ошибка несоответствия пароля | <p>Если указан неверный пароль.</p> <p>Пример: ПУТ -https://localhost:9002/rest/v1/electronics/customers/current/login</p> <p>Заголовок:</p> <p>Тип содержимого: application/x-www-form-urlencoded</p> <p>Текст запроса:</p> <p>newLogin=aa@bb.com &password=ваш_пароль</p> <p>и пароль неверный</p> |

Ошибки корзины ресурсов

| Ответ Статус | Ошибка | Случай |
|---------------------------------|-------------------------------|--|
| HTTP/1.1 400 Плохо Запрос | CommerceCartModificationError | <p>Общий случай:</p> <p>Это исключение возникает при добавлении, обновлении или удалении товаров из корзины. Примеры ошибок: р</p> <p>Когда запись корзины с запрошенным номером отсутствует в текущей корзине.</p> <p>Пример:</p> <ul style="list-style-type: none"> • ПОМЕЩАТЬ -https://localhost:9002/rest/v1/electronics/cart/entry/3?qty=5 • УДАЛИТЬ -https://localhost:9002/rest/v1/electronics/cart/entry/3 |

| Ответ Статус | Ошибка | Случай |
|---------------------------------|--------------------------------------|--|
| | | <p>Когда указанное количество неверно</p> <p>Пример: ПОСТ -https://localhost:9002/rest/v1/electronics/cart/entry?code=4</p> |
| HTTP/1.1 400 Плохо Запрос | Ошибка проверки | <p>Когда корзина недействительна в запросе на размещение заказа. Это означает, что нет необходимой информации для</p> <p>Пример: ПОСТ -https://localhost:9002/rest/v1/electronics/cart/placeorder</p> |
| HTTP/1.1 400 Плохо Запрос | НеподдерживаемыйОшибкаАдресаДоставки | <p>Если адрес, указанный в качестве адреса доставки, неверен; например: адрес не существует,</p> <p>Пример: ПУТ -https://localhost:9002/rest/v1/electronics/cart/address/deliv</p> |
| HTTP/1.1 400 Плохо Запрос | НеподдерживаемыйрежимдоставкиОшибка | <p>Когда способ доставки не существует или адрес доставки не указан для корзины.</p> <p>Пример: ПУТ -https://localhost:9002/rest/v1/electronics/cart/deliverymodes</p> |
| HTTP/1.1 400 Плохо Запрос | НевернаяОшибкаИнформацииОплаты | <p>Когда возникла проблема с настройкой платежной информации для корзины; например, не было сеанса с</p> <p>Пример: ПУТ -https://localhost:9002/rest/v1/electronics/cart/paymentinfo/8</p> |
| HTTP/1.1 400 Плохо Запрос | Ошибка авторизации платежа | <p>При возникновении проблем с авторизацией платежа, например: отсутствует сессионная корзина или нет р</p> <p>Пример: ПОСТ -https://localhost:9002/rest/v1/electronics/cart/authorize?se</p> |
| HTTP/1.1 400 Плохо Запрос | НетОшибкаОформленияКорзины | <p>Общий случай:</p> <p>Это исключение выдается, когда нет сессионной корзины во время шага процесса оформления заказа. Перед вызовом</p> |

| Ответ Статус | Ошибка | Случай |
|---------------------------------|------------------------------|---|
| | | <p>Если в запросе на установку адреса доставки не указано ни одной корзины в сеансе.</p> <p>Пример: ПУТ -https://localhost:9002/rest/v1/electronics/cart/address/deliv</p> <p>и вы забыли добавить куки JSESSIONID из предыдущего запроса</p> |
| | | <p>Если при применении или выдаче ваучера в сеансе нет корзины.</p> <p>Пример: ПОСТ -https://localhost:9002/rest/v1/electronics/cart/voucher/xyz</p> |
| | | <p>Если при добавлении платежной информации в сеансе нет корзины.</p> <p>Пример: ПОСТ -https://localhost:9002/rest/v1/electronics/cart/paymentinfo</p> <p>Заголовок:</p> <p>Тип содержимого: application/x-www-form-urlencoded</p> <p>Текст запроса:</p> <p>accountHolderName=John+Doe&cardNumber=4111111111111111&cardType=visa&expi 16-19&billingAddress.postalCode=531-0072&billingAddress.town=Osaka&billin</p> |
| | | <p>Когда в сеансе размещения заказа нет корзины.</p> <p>Пример: ПОСТ -https://localhost:9002/rest/v1/electronics/cart/placeorder</p> |
| HTTP/1.1 400 Плохо Запрос | CommerceCartRestorationError | <p>Когда корзина, которую вы хотите восстановить, не существует.</p> <p>Пример: ПОЛУЧИТЬ -https://localhost:9002/rest/v1/electronics/cart/restore?guid=</p> |
| HTTP/1.1 400 Плохо Запрос | НедостаточноеоОшибкаЗапаса | <p>Общий случай:</p> <p>Когда товара нет в наличии или его мало на складе. Примеры ошибок приведены ниже.</p> <p>Если при добавлении в корзину товара нет в наличии.</p> <p>Пример: ПОСТhttps://localhost:9002/rest/v1/electronics/cart/entry?code=8</p> |
| | | <p>Если товара нет в наличии в магазине, где его необходимо забрать.</p> <p>Пример: ПУТhttps://localhost:9002/rest/v1/electronics /cart/entry/0/store</p> |

| Ответ Статус | Ошибка | Случай |
|---------------------------------|------------------------------------|--|
| | | <p>Если при попытке разместить заказ товара нет в наличии.</p> <p>Пример: ПОСТ <code>https://localhost:9002/rest/v1/electronics/cart/placeorder</code></p> |
| HTTP/1.1 400 Плохо Запрос | StockSystemError | <p>Когда система учета запасов не включена и информация о наличии товара на складе отсутствует.</p> <p>Пример:</p> <ul style="list-style-type: none"> ПОЧТА <code>https://localhost:9002/rest/v1/electronics/cart/entry?code=489</code> ПОМЕЩАТЬ <code>https://localhost:9002/rest/v1/electronics/cart/entry/store?sto</code> |
| HTTP/1.1 400 Плохо Запрос | CommercePromotionОграничениеОшибка | <p>Когда для акции нет PromotionOrderRestriction, когда мы пытаемся применить акцию к</p> <p>Пример: ПОСТ <code>-https://localhost:9002/rest/v1/electronics/cart/promotion/Или</code></p> |
| HTTP/1.1 400 Плохо Запрос | ОшибкаОперацииВаучера | <p>При возникновении ошибки во время операции с ваучером. Примеры ошибок приведены ниже.</p> <p>При попытке применить несуществующий ваучер.</p> <p>Пример: ПОСТ <code>-https://localhost:9002/rest/v1/electronics/cart/voucher/notE</code></p> <p>При возникновении ошибки в процессе подачи заявки на ваучер.</p> <p>Пример: ПОСТ <code>-https://localhost:9002/rest/v1/electronics/cart/voucher/xyz-</code></p> <p>При возникновении ошибки в процессе выдачи ваучера.</p> <p>Пример: УДАЛИТЬ <code>-https://localhost:9002/rest/v1/electronics/cart/voucher/xyz</code></p> |

Ошибки группы клиентов ресурса

| Ответ Статус | Ошибка | Случай | P |
|---------------------------------|---------------------------|---|---|
| HTTP/1.1 400 Плохо Запрос | яnvalidCustomerGroupError | Когда запрашиваемая группа не является подгруппой группы клиентов. Пример: ПОЛУЧИТЬ - https://localhost:9002/rest/v1/electronics/customergroups/customermanagergroup | |

Ошибки ресурса ваучера

| Ответ Статус | Исключение | Случай | Тело ответа |
|---------------------------------|-----------------------|---|--|
| HTTP/1.1 400 Плохо Запрос | ОшибкаОперацииВаучера | Если ваучер с указанным кодом не существует. Пример: ПОЛУЧИТЬ - https://localhost:9002/rest/v1/electronics/vouchers/abc-9PSW-EDH2 | <?xml version="1.0" кодировка <errors> <ошибка> <message>Номер ваучера <type>VoucherOperat </ошибка> </ошибки> |

Совместное использование ресурсов из разных источников

Cross-Origin Resource Sharing (CORS) — это попытка W3C внедрить стандартный механизм для поддержки кросс-доменных запросов в веб-браузерах и на существующих серверах.

Обзор

Программное обеспечение SAP Commerce использует фильтр CORS с открытым исходным кодом, который является стандартным фильтром Java EE.

Cross-Origin Resource Sharing позволяет веб-браузерам обходить политику одного и того же источника, которая обычно запрещает межсайтовые HTTP-запросы. CORS поддерживается большинством современных веб-браузеров, включая Internet Explorer, начиная с версии 10.

Конфигурация по умолчанию для расширения uscommercewebservices

Файл `web.xml` для `uscommercewebservices` поставляется с конфигурацией по умолчанию, которая позволяет браузерным клиентам JavaScript использовать RESTful `uscommercewebservices` API.

```

<фильтр>
    <имя-фильтра>CORS</имя-фильтра> <класс-фильтра>com.thetransactioncompany.cors.CORSFilter</класс-фильтра> <параметр-init>
        <param-name>cors.supportedMethods</param-name> <param-value>GET, POST, HEAD,
        PUT, DELETE, OPTIONS</param-value> </init-param>
    </фильтр>
    <фильтр-отображение>
        <имя-фильтра>CORS</имя-фильтра> <url-шаблон>/*</url-шаблон> </
        сопоставление-фильтра>

```

Как только браузер из другого источника делает HTTP-запрос, этот фильтр автоматически добавляет заголовки CORS по умолчанию к ответу. Инициирующий браузер должен понимать заголовки CORS, чтобы разрешить межсайтовый запрос.

Образец запроса

ПОЛУЧИТЬ <http://earlybird.hybris.com:9001/rest/v1/electronics/products?query=camera> Источник:
<http://nfcbiergarten.appspot.com> ...

Примеры ответов

200 OK
Access-Control-Allow-Credentials: правда

Access-Control-Allow-Origin: <http://nfcbiergarten.appspot.com> . . .

Отключение CORS

Если у вас нет браузерного JavaScript-клиента, который обращается к нашему API, отключите CORS. Для этого удалите фильтр и сопоставление фильтра из web.xml. Кроме того, удалите файл CORS jar из \ext-hybris\ycommercewebservices\web\webroot\WEB-INF\lib\каталог.

Обновление до расширений OCC

Руководство по обновлению с версии v1 до реализации API OCC на основе расширений.

Измененные URL-адреса

Вам необходимо изменить URL-адреса, чтобы они указывали на новые ресурсы. Главное изменение — это номер версии в URL-адресе, который теперь v2. Все пути URL-адресов ресурсов также должны быть адаптированы к новой конвенции. Конкретное сопоставление ресурсов описано в [Сопоставление ресурсов API](#) ниже.

Новые типы возврата

В расширениях OCC все возвращаемые объекты были изменены на объекты WsDTO, поэтому необходимо изменить всю обработку ответа на новую структуру ответа. Все объекты WsDTO, возвращаемые методами API, описаны в [Концепция WsDTO](#) страница.

Параметры нового запроса

В расширениях OCC методы имеют новый параметр запроса, который определяет форму возвращаемого объекта. Этот параметр запроса предоставляется по умолчанию. Однако, чтобы гарантировать, что объекты ответа предоставляют все требуемые поля, вы можете настроить параметр запроса.

Сопоставление ресурсов API

В следующей таблице показано сопоставление ресурсов API.

| B1 | | Расширения OCC |
|---------------------------------------|-----------------------|--|
| /сайт/клиенты | ПОЧТА | /сайт/пользователи |
| /сайт/клиенты/текущий | получать | /сайт/пользователи/{идентификатор_пользователя} |
| /сайт/клиенты/текущий/логин | помещать | /сайт/пользователи/{идентификатор_пользователя}/логин |
| /сайт/клиенты/текущий/профиль | помещать | /сайт/пользователи/{идентификатор_пользователя} |
| /сайт/клиенты/текущий/пароль | РАЗМЕСТИТЬ/РАЗМЕСТИТЬ | /сайт/пользователи/{идентификатор_пользователя}/пароль |
| /site/customers/current/забытыйпароль | ПОЧТА | /site/забытыепаролитокены |
| /сайт/клиенты/текущие/адреса | получать | /сайт/пользователи/{идентификатор_пользователя}/адреса |

| | | | |
|---|--------------------|--|--|
| B1 | | | Расширения ОСС |
| /{сайт}/клиенты/текущие/адреса | ПОЧТА | | /{сайт}/пользователи/{идентификатор_пользователя}/адреса |
| /{сайт}/клиенты/текущие/адреса/{id} | ПОМЕЩАТЬ | | /{сайт}/пользователи/{идентификатор_пользователя}/адреса/{a} |
| /{сайт}/клиенты/текущие/адреса/{id} | УДАЛИТЬ | | /{сайт}/пользователи/{идентификатор_пользователя}/адреса/{a} |
| /{сайт}/клиенты/текущие/адреса/по умолчанию/{id} | ПОМЕЩАТЬ | | удалено - используйте /{site}/users/{userId} |
| /{site}/customers/current/paymentinfos | ПОЛУЧАТЬ | | {site}/users/{userId}/paymentdetail |
| /{сайт}/клиенты/текущие/платежныеинформации/{id} | ПОЛУЧАТЬ | | {site}/users/{userId}/paymentdetai |
| /{сайт}/клиенты/текущие/платежныеинформации/{id} | УДАЛИТЬ | | {site}/users/{userId}/paymentdetai |
| /{сайт}/клиенты/текущие/платежныеинформации/{id} | ПОМЕЩАТЬ | | {site}/users/{userId}/paymentdetai |
| /{site}/customers/current/paymentinfos/{id}/address | ПОМЕЩАТЬ | | {site}/users/{userId}/paymentdetai |
| /{сайт}/клиенты/текущие/группыклиентов | ПОЛУЧАТЬ | | {site}/users/{userId}/customergrou |
| /{site}/customers/{uid}/customergroups | ПОЛУЧАТЬ | | {site}/users/{userId}/customergrou |
| /{site}/customers/current/locationLatLong | ПОМЕЩАТЬ | | удалено - потому что нет сесси |
| /{сайт}/клиенты/текущее/местоположение | ПОМЕЩАТЬ | | удалено - потому что нет сесси |
| /{сайт}/клиенты/текущее/местоположение | ПОЛУЧАТЬ | | удалено - потому что нет сесси |
| /{site}/customer/current/гостевой_вход | ПОЧТА | | {site}/users/{userId}/carts/{cartI |
| /{сайт}/клиент/текущий/конвертировать | ПОЧТА | | /{сайт}/пользователи |
| /{сайт}/заказы | ПОЛУЧАТЬ | | /{сайт}/пользователи/{идентификатор_пользователя}/заказы |
| /{сайт}/заказы/{код} | ПОЛУЧАТЬ | | /{сайт}/пользователи/{идентификатор_пользователя}/заказы/{код} |
| ТЕЛЕЖКИ | | | |
| /{сайт}/корзина | ПОЛУЧАТЬ | | {site}/users/{userId}/carts/{cartI |
| /{сайт}/корзина/запись | ПОЧТА | | {site}/users/{userId}/carts/{cartI |
| /{сайт}/корзина/запись/{номер записи} | РАЗМЕСТИТЬ,УДАЛИТЬ | | {site}/users/{userId}/carts/{cartI |
| /{сайт}/корзина/адрес/доставка/{id} | ПОМЕЩАТЬ | | {site}/users/{userId}/carts/{cartI |
| /{сайт}/корзина/адрес/доставка | УДАЛИТЬ | | {site}/users/{userId}/carts/{cartI |
| /{сайт}/корзина/режимы доставки/{код} | ПОМЕЩАТЬ | | {site}/users/{userId}/carts/{cartI |
| /{сайт}/корзина/режимы доставки | ПОЛУЧИТЬ,УДАЛИТЬ | | {site}/users/{userId}/carts/{cartI |
| /{сайт}/корзина/разместитьзаказ | ПОЧТА | | {site}/users/{userId}/orders?cartI <securityCode> |
| /{site}/cart/paymentinfo | ПОЧТА | | {site}/users/{userId}/carts/{cartI |
| /{сайт}/корзина/платежная информация/{id} | ПОМЕЩАТЬ | | {site}/users/{userId}/carts/{cartI <id>} |
| /{сайт}/корзина/авторизация | ПОЧТА | | удаленный |
| /{сайт}/корзина/восстановление | ПОЛУЧАТЬ | | удаленный |
| /{site}/cart/promotion/{promotionCode} | ПОЧТА | | {site}/users/{userId}/carts/{cartI |
| /{site}/cart/promotion/{promotionCode} | УДАЛИТЬ | | {site}/users/{userId}/carts/{cartI |
| /{сайт}/корзина/ваучер/{код ваучера} | ПОЧТА | | {site}/users/{userId}/carts/{cartI |

| | | | |
|---|---------------------|-------------------------------------|----------------|
| B1 | | | Расширения ОСС |
| /{сайт}/корзина/ваучер/{код ваучера} | удалить | /site/users/{userId}/carts/{cartId} | |
| /{сайт}/корзина/оформление заказа | ПОЧТА | | |
| КАТАЛОГИ | | | |
| /{сайт}/каталоги | получать | /сайт}/каталоги | |
| /{сайт}/каталоги/{id} | получать | /сайт}/каталоги/{каталогId} | |
| /{site}/catalogs/{catalogId}/{catalogVersionId} | получать | /site/catalogs/{catalogId}/{catal | |
| /{site}/catalogs/{catalogId}/{catalogVersionId}/categories/{category} | получать | /site/catalogs/{catalogId}/{catal | |
| ГРУППЫ КЛИЕНТОВ | | | |
| /{сайт}/группыклиентов | получить,разместить | /сайт}/группыклиентов | |
| /{site}/customergroups/{uid}/members | помещать | /site/customergroups/{groupId}/me | |
| /{site}/customergroups/{uid}/members/{userId} | удалить | /site/customergroups/{groupId}/me | |
| /{сайт}/customergroups/{uid} | получать | /site/customergroups/{groupId} | |
| ЗАКАЗЫ | | | |
| /{site}/orders/byGuid/{guid} | получать | /сайт}/заказы/{код} | |
| /{site}/orders/statusFeed | получать | /site/feeds/orders/statusfeed | |
| АКЦИИ | | | |
| /{сайт}/акции | получать | /сайт}/акции | |
| /{сайт}/акции/{код} | получать | /сайт}/акции/{код} | |
| РАЗНОЕ | | | |
| /{сайт}/языки | получать | /сайт}/языки | |
| /{сайт}/валюты | получать | /сайт}/валюты | |
| /{site}/страныдоставки | получать | /site/страныдоставки | |
| /{сайт}/названия | получать | /сайт}/названия | |

| | | |
|---|----------|--|
| B1 | | Расширения ОСС |
| /сайт/типы карт | получать | /сайт/типы карт |
| МАГАЗИНЫ | | |
| /сайт/магазины | получать | /сайт/магазины |
| /сайт/магазины/{имя} | получать | /сайт/stores/{storeId} |
| ВАУЧЕРЫ | | |
| /сайт/ваучеры/{код} | получать | |
| ПРОДУКЦИЯ | | |
| /сайт/продукты | получать | /сайт/продукты/поиск |
| /site/products{productCode}/stock?storeName=<имя> | получать | /site/products/{productCode}/stoc |
| /сайт/продукция/экспорт/полный | получать | /сайт/продукция/экспорт/полный |
| /site/products/export/incremental | получать | /site/products/export/incremental |
| /сайт/продукция/экспорт/ссылки/{код} | получать | /site/products/{productCode}/refe |
| /site/products/expressОбновление | получать | /site/products/expressОбновление |
| /сайт/продукты/{код} | получать | /сайт/продукты/{productCode} |
| /сайт/продукты/предложить | получать | /сайт/продукты/предложения |
| /сайт/продукты/{код}/отзывы | почта | /site/products/{productCode}/revi |
| /сайт/продукты/{код}/рядом сРасположением | получать | /site/products/{productCode}/stoc |
| /сайт/продукты/{код}/nearLatLong | получать | /site/products/{productCode}/stoc <latitude> |