

REST-API für oneKlick

giropay

Version 1.99.0

Inhalt

Einführung	1
Endpoint	5
REST-Prinzipien	7
Authentifizierung	12
API-Key	12
Signatur	13
Access Token	14
Resource Access	20
Autorisierungen	22
Anlage Käuferautorisierung	22
Verhalten bei nicht angegebenen Rücksprungadressen	27
Abfrage Käuferautorisierung	28
Callback für Statusupdates	33
Anlage Käuferautorisierung während eines Checkouts	35
Duplikate von Käuferautorisierungen	36
Authentifizierung	36
Zurückziehen von Käuferautorisierungen	37
Autorisierungsdialog	37
Anlage Händlerautorisierung	38
Abfrage Händlerautorisierung	38
oneKlick-Checkout	41
Anlage	41
Checkout	67
Anlage	67
Abfrage	91
Callback für Statusupdates	106
Aktualisierung der Versandbedingungen	109
Aktualisierung der Rechnungsreferenz	110
Capture	112
Anlage	113
Abfrage	116
Callback für Statusupdates	119
Order (Secured) schließen	121
Refund	123
Anlage	123
Abfrage	125
Callback für Statusupdates	129
Accountinformationen	132

Abfrage von Accountinformationen	132
Abfrage der Transaktionshistorie	134
SDKs für mobile Apps	139
Allgemeine Datentypen	140
ShippingAddress	140
Item	141
Message	142
Page	143
Zeichensatz	144
SEPA-konforme Zeichen	144
Sandbox Testing	145
Anpassen des Autorisierungsstatus	145
Beispiel	147
Changelog	149

Einführung

giropay ist das Bezahlverfahren der deutschen Privatbanken, Volks- und Raiffeisenbanken und Sparkassen. Die paydirekt GmbH ist Betreiber des giropay-Systems. Im Folgenden wird daher die Anbindung an das Bezahlverfahren giropay erläutert, bei welchem bspw. die Endpunkte die Bezeichnung paydirekt beinhalten.

Dieses Dokument beschreibt die technische Integration des giropay-Bezahlverfahrens in Anwendungen Dritter (oneKlick-Anbieter). Beispiele für solche Anwendungen umfassen mobile Apps, Video-on-Demand-Anwendungen für Smart TVs oder Anwendungen zur persönlichen Finanzverwaltung. Zudem kann das Bezahlverfahren auch direkt in Web-Shops integriert werden, die ihren Kunden das Bezahlen ohne Umweg über das giropay-Checkout-Portal ermöglichen wollen. Die oneKlick-API kann auch für die Erstellung und Verwaltung normaler Checkouts durch einen PSP oder CPSP verwendet werden. Dafür sind die im folgenden Absatz und in den entsprechenden Kapiteln beschriebenen Käuferautorisierung nicht nötig.

Registrierten und freigeschalteten Anbietern steht zu diesem Zweck diese API zur Verfügung. Über sie können oneKlick-Anbieter über eine gesicherte Backend-to-Backend-Verbindung für ihre bei giropay registrierten Benutzer Bezahlvorgänge durchführen oder Account-Informationen wie das vorhandene Guthaben oder die Transaktionshistorie abrufen. Der Kunde autorisiert den oneKlick-Anbieter einmalig und nur für den notwendigen Funktionsumfang durch die sichere Eingabe seiner Zugangsdaten. Dabei werden dem Kunden die einzelnen Funktionen angezeigt, die die Anwendung für ihn durchführen will. Danach kann der Benutzer schnell, bequem und direkt aus der Anwendung heraus Zahlungen tätigen oder Account-Informationen einsehen, ohne seine Zugangsdaten erneut eingeben zu müssen. Das Backend der oneKlick Anwendung erhält dafür vom giropay-System für erteilte Benutzerautorisierungen Token, mit denen sie ihre Berechtigung, Funktionen im Namen des Kunden aufzurufen, ausweisen kann.

Der oneKlick-Anbieter kann für die Durchführung von Bezahlvorgängen selbst bei giropay als Händler registriert sein und in seiner Händlerrolle Zahlungen zugunsten eigener Shops anlegen. Darüber hinaus kann der oneKlick-Anbieter auch für die Shops anderer Händler Zahlungen durchführen. In diesem Fall benötigt er analog zur Kundenautorisierung eine Autorisierung des Händlers für den entsprechenden Shop. Auch CPSPs müssen für die Nutzung dieser API immer eine explizite Händlerautorisierung vorweisen. Allerdings erhalten sie diese für selbst angelegte Händler automatisch.

Die oneKlick-API basiert auf REST, folgt den HATEOAS-Prinzipien, entspricht der HAL-Spezifikation und verwendet JSON als Datenformat.

Nutzung der oneKlick-API mit Käuferautorisierungen

Zu Beginn muss eine oneKlick-Anwendung vom Benutzer eine Autorisierung erbitten. Eine einmal erteilte Autorisierung wird im giropay-System hinterlegt und berechtigt den oneKlick-Anbieter in Zukunft, Funktionen im Namen dieses Benutzers aufzurufen.

Der Anbieter initiiert den Autorisierungsprozess durch Anlage einer Autorisierung als Ressource im giropay-System. Dazu sendet sie einen HTTP-POST-Request mit dem gewünschten Rechtemumfang (*Scope*). Als Antwort erhält er die angelegte Ressource im JSON-Format. Die

Autorisierungsressource enthält einen HAL-Link **web**, auf den der Anbieter den Kunden weiterleitet; beispielsweise über einen HTTP-Redirect oder über eine in einer mobilen App eingebetteten Web-View. Dort kann der Kunde über einen vom giropay-System angebotenen Autorisierungsdialog die Autorisierungsanfrage einsehen, sich mit seinen giropay-Zugangsdaten authentifizieren und die Autorisierung erteilen. Nach erteilter Autorisierung leitet der Autorisierungsdialog den Kunden zurück zur Anwendung. Die Autorisierungsressource enthält nun eine vertrauliche **customerAuthorizationReference**, die nur der authentifizierte oneKlick-Anbieter über ein HTTP-GET auf der Resource abrufen kann. Der Link kann auch jederzeit nach Ablauf der Frist betätigt werden, wodurch die Autorisierung reaktiviert wird, sofern sie noch im Status **APPROVED** ist.



Für die korrekte Ermittlung des Checkout Ergebnisses ist der in Abbildung [Ablauf des Autorisierungsprozesses](#) beschriebene letzte GET-Aufruf zwingend erforderlich.

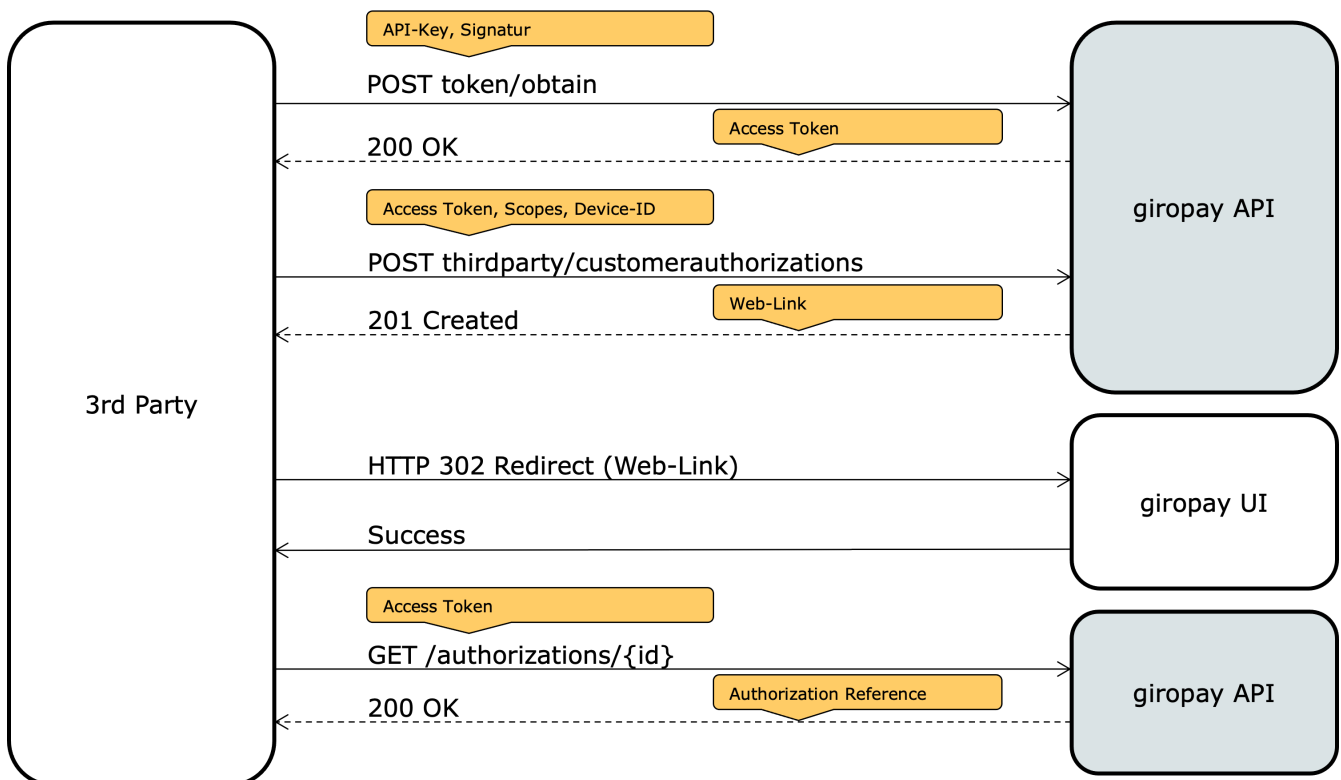


Figure 1. Ablauf des Autorisierungsprozesses

Durch Angabe der **customerAuthorizationReference** kann sich der oneKlick-Anbieter vom giropay-System ein OAuth2-Access-Token mit einer begrenzten Gültigkeitsdauer ausstellen lassen. Das Vorzeigen des Tokens autorisiert den oneKlick-Anbieter zur Nutzung der giropay-API im Rahmen des vom Kunden autorisierten Rechteumfangs. Die **customerAuthorizationReference** ist gültig, bis die Autorisierung vom Kunden, einem paydirekt-Mitarbeiter manuell widerrufen wird oder der oneKlick-Anbieter eine inhaltsgleiche Autorisierung anlegt (siehe [Duplikate von Käuferautorisierungen](#)).

Will der oneKlick-Anbieter Bezahlvorgänge für andere Händler durchführen oder handelt es sich um einen CPSP, muss er zusätzlich eine **merchantAuthorizationReference** für eine entsprechende Autorisierung des Händlers übergeben. Das ausgestellte Access-Token berechtigt den Anbieter dann für den entsprechenden Shop des Händlers Bezahlvorgänge durchzuführen. Die Anlage einer solchen Händlerautorisierung muss über den giropay-Support angefragt werden. Bei CPSP-

Händlern erfolgt die Anlage automatisch und die `merchantAuthorizationReference` kann im Intermediärportal eingesehen werden.

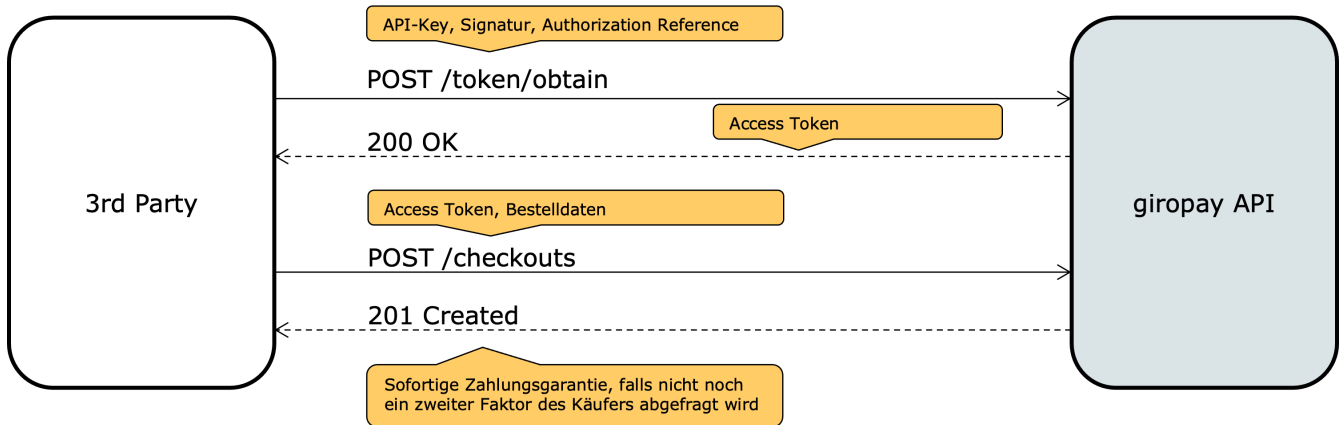


Figure 2. Durchführung eines Bezahlvorganges

Freischaltung der oneKlick-API

Der Zugriff auf die oneKlick-API ist nur nach der individuellen Freischaltung durch die paydirekt GmbH möglich. Bei CPSPs muss die paydirekt GmbH die Funktion grundsätzlich für den gesamten CPSP freischalten, die Aktivierung einzelner CPSP-Händler erfolgt dann in Selbstverwaltung im Intermediärportal. Die Einhaltung der von der paydirekt GmbH geforderten IT-Security Standards ist Voraussetzung für die Freischaltung. Die direkte Interaktion mit der API ist nur über entsprechend gesicherte Backend-to-Backend-Verbindungen möglich. Ein direkter Zugriff auf die API aus einer mobilen App heraus ist beispielsweise nicht erlaubt.



Hinweis für CPSPs, und Händler, die zusätzlich oneKlick anbieten: Die Kapitel Capture und Refund, sowie die allgemeinen Angaben zu Datentypen und Zeichensatz gleichen inhaltlich den entsprechenden Kapiteln in der Sie betreffenden allgemeinen Schnittstellen-Dokumentation. Für CPSPs gilt dies zusätzlich auch für das Kapitel Authentifizierung.

Erfolgsstatus eines Checkouts



Für die korrekte Ermittlung des Checkout-Ergebnisses ist der in Abbildung [Ablauf des Autorisierungsprozesses](#) beschriebene letzte GET-Aufruf zwingend erforderlich.

Für die Sicherstellung einer erfolgreichen und validen Auswertung des Checkout-Ergebnisses wird dringend empfohlen, eine GET-Abfrage auf den vorher angelegten Checkout durchzuführen. Aufrufe der Status-Callback URLs ([siehe Checkout Callback](#)) oder erfolgte Weiterleitung auf eine der beim Create-Request angegebenen Redirect URLs ([siehe Checkout Anlage Request-Fields](#)) können unzureichend sein.

Hintergrund: Der Redirect kann aufgrund von Verbindungsfehlern nach einem erfolgreichen Checkout ausbleiben oder der Käufer bricht die Weiterleitung ab. Die Statusrückmeldung erfolgt asynchron und ist auf Verfügbarkeit des Händlerendpunkts angewiesen. In beiden Fällen kann die paydirekt GmbH somit nicht garantieren, dass hierüber der Ausgang eines Checkouts immer

verlässlich und zeitnah ermittelbar ist.

Zudem sind weder die Redirect URL noch die Status Callback URL Aufrufe gegen Manipulation oder Missbrauchsversuche abgesichert.

Endpoint

Die giropay-Schnittstellen sind über folgende URL erreichbar:

```
https://api.paydirekt.de
```

Zum Test der Schnittstelle steht eine Test-Umgebung (Sandbox) mit folgendem Endpoint zur Verfügung. In der Sandbox werden keine Buchungen auf Bankkonten ausgelöst, auch nicht innerhalb einer Testbank:

```
https://api.sandbox.paydirekt.de
```

Die festen IP-Adressen der Endpoints lauten:

```
Produktiv: 213.95.157.133  
Sandbox: 213.95.255.167
```

Die festen IP-Adressen, von denen Callback URLs aufgerufen werden, lauten:

```
Produktiv: 213.95.26.236  
Sandbox: 213.95.26.164
```

Transport Layer Security

giropay ist ausschließlich über verschlüsselte Endpoints (HTTPS) erreichbar. Der [Payment Card Industry Security Standards Council](#) empfiehlt bereits heute die Verwendung von TLS 1.2 bei API-Anbindungen. Folgende Cipher-Suites werden für TLS 1.2 empfohlen: ECDHE-RSA-AES256-GCM-SHA384

Erklärung:

- Diese Cipher-Suite wird durch das BSI empfohlen und kann noch mindestens bis 2021 und darüber hinaus verwendet werden.
- ECDHE bietet Perfect Forward Secrecy.
- ECDHE benötigt weniger Ressourcen als DHE.
- AES256 ist eine starke Verschlüsselung.
- RSA ist das zugrundeliegende Verfahren für das Private/Public-Key Schlüsselmaterial.
- SHA384 ist eine sichere schnelle Hash-Funktion für Signaturen.
- SHA384 bietet Schutz auch gegen Length-Extension-Attacks (im Gegensatz zu SHA256).
- GCM ist ein moderner Cipher-Block-Mode für AES, der bestimmte Angriffsvektoren, die bei der Alternative CBC möglich sind, verhindert.

- GCM etabliert eine Authenticated Encryption.

Eine Übersicht darüber, welche älteren Versionen von TLS aktuell noch unterstützt werden, finden Sie [auf folgender Seite:](https://www.ssllabs.com/ssltest/analyze.html?d=api.paydirekt.de&hideResults=on)
<https://www.ssllabs.com/ssltest/analyze.html?d=api.paydirekt.de&hideResults=on>



Falls Java verwendet wird, müssen die Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files installiert sein.

REST-Prinzipien

Das giropay-System stellt eine REST-Schnittstelle zur Integration von Zahlungen in Webshops zur Verfügung.

Die API folgt den [HATEOAS-Prinzipien](#), basiert auf der [HAL-Spezifikation](#) und verwendet [JSON](#) als Datenformat.

HTTP-Methoden

Die REST-API definiert Ressourcen, die Geschäftsobjekten entsprechen und die per URL referenziert werden. Zur Erzeugung, Änderung und Datenabfrage dieser Ressourcen werden Standard-HTTP-Methoden entsprechend [RFC 2616](#) verwendet:

Methode	Art	Beschreibung
GET	Query	Liefert die Repräsentation eines oder mehrerer Geschäftsobjekte zurück. Es werden keine Veränderungen auf dieser Ressource durchgeführt.
POST	Command	Legt ein neues Geschäftsobjekt an.

HTTP Header-Felder

Die nachfolgenden Header sind in allen Serviceaufrufen (GET/POST) mit Ausnahme des Access-Token Endpoints mit zu übermitteln.

Header-Feld	Wert	Beispiel
Authorization	Access-Token aus dem OAuth2-Aufruf	Bearer <accessTokenValue>
X-Request-ID	Eindeutige Request-ID, wird zur Protokollierung verwendet. Wenn im Request keine gesetzt ist, wird serverseitig eine erzeugt und in der HTTP Response zurückgegeben. Wenn clientseitig eine Request-ID angegeben wird, sollte das Format <i>UUID Typ 4 Random</i> verwendet werden. Optional.	8eb5d811-ad7f-4e08-94e4-5dcfa03a4b15
Content-Type	Der Internet Media Type des Request Body. Es wird nur <code>application/hal+json</code> unterstützt. Pflichtfeld bei Command-Methoden.	<code>application/hal+json</code>
Accept	Die akzeptierten und erwarteten Internet Media Types der Response. Es wird nur <code>application/hal+json</code> unterstützt. Pflichtfeld bei Aufrufen mit Response-Body.	<code>application/hal+json</code>
Date	Datum und Zeit zum Sendezeitpunkt der Anfrage. Format nach [RFC 7231] IMF-fixdate. Pflichtfeld.	Fri, 30 May 2014 09:46:48 GMT

HTTP Status Codes und Fehlermeldungen

Die folgenden allgemeinen Status Codes können potentiell bei allen Service-Aufrufen auftreten und müssen entsprechend behandelt werden. Besonderheiten für eine Ressource sind jeweils im Abschnitt *Return Codes* erläutert.

HTTP Status Code	Beschreibung
200 (OK)	Der Request wurde erfolgreich ausgeführt.
201 (Created)	Die gewünschte Ressource wurde angelegt. Sie ist unter der im Header hinterlegten <code>Location</code> zu erreichen. Dieser Status Code kommt nach einem bestimmten erfolgreichen POST auf eine Ressourcen-URL.
204 (No Content)	Der Request wurde erfolgreich ausgeführt. In der Response gibt es keinen Body.
400 (Bad Request)	Der Request wurde nicht angenommen. Details stehen in Response Body.
401 (Unauthorized)	Es wurde versucht, auf eine Ressource zuzugreifen, die geschützt ist. Entweder ist der API-Key falsch, das OAuth2 Token fehlt bzw. ist abgelaufen oder der Zugriff auf die Ressource wird verweigert.
403 (Forbidden)	Der Zugriff ist auf diese Ressource mit der verwendeten HTTP-Methode nicht erlaubt.
404 (Not Found)	Unter dieser URL kennt der Server keine Ressource.
410 (Gone)	Die API steht in dieser Version nicht mehr zur Verfügung.
422 (Unprocessable Entity)	Es ist ein fachlicher Fehler aufgetreten. Details stehen im Response Body.
500 (Internal Server Error)	Es ist ein serverseitiger Fehler aufgetreten. Details stehen im Response Body.
503 (Service Unavailable)	Das giropay-System ist vorübergehend nicht erreichbar (z. B. wegen Wartung).

Error Messages

Wenn ein Fehler auftritt, dann werden neben dem HTTP-Status Code Details zum Fehler in einem Array von `Message`-Objekten mit dem Namen `messages` im Response-Body zurückgegeben.

Messages haben einen definierten, eindeutigen Message Code (Feld `code`) und eine Fehlerstufe (`severity`). Außerdem kann ein Attribut `logref` enthalten sein, das den Fehler eindeutig referenziert und die Fehleranalyse vereinfacht. Geben Sie diese `logref` bei Support-Anfragen bitte immer mit an.

Die möglichen Message Codes sind jeweils im Abschnitt *Return Codes* aufgeführt.

Es muss ein Default-Verhalten für unbekannte oder fehlende Message Codes im Falle eines 4xx oder 5xx Status Codes implementiert werden.

Attribut	Typ	Beschreibung
messages	Liste von Messages	Eine Liste aller aufgetretener Fehler.

Sonderfall Validierungsfehler

Validierungsfehler werden ebenfalls in der zuvor beschriebenen Form, zusammen mit dem HTTP-Status Code 400, zurückgeliefert.

Der Pfad eines betroffenen Attributs wird im Feld `path` angegeben. Komplexe Objekte werden durch Punkte getrennt. Arrays werden über `[n]` referenziert.

Zusätzlich kann in einem weiteren optionalen Feld `reasonCode` eine detaillierte Ursache der Validierungsmeldung spezifiziert werden.

Für den Message Code `VALIDATION_ERROR` sind derzeit folgende Reason Codes definiert:

Reason Code	Beschreibung
<code>INVALID_FORMAT</code>	Der Wert entspricht nicht den Formatvorgaben.
<code>MANDATORY_VALUE_MISSING</code>	Ein Pflichtfeld ist nicht angegeben.
<code>INVALID_ENUM_VALUE</code>	Es wurde ein Wert verwendet, der nicht im angegebenen Aufzählungstyp enthalten ist.

Beispiel

POST /api/checkout/v1/checkouts HTTP/1.1
Content-Type: application/json;charset=utf-8
Authorization: Bearer <access_token>

```
{
  "type" : "DIRECT_SALE",
  "totalAmount" : 100.0,
  "shippingAmount" : 5.999,
  "orderAmount" : 96.5,
  "refundLimit" : 200,
  "items" : [ {
    "quantity" : 3,
    "name" : "Bobbycar",
    "ean" : "800001303",
    "price" : 25.99
  }, {
    "quantity" : 1,
    "name" : "Helm",
    "price" : 18.53
  } ],
  "shoppingCartType" : "MIXED",
  "currency" : "EUR",
  "overcapture" : false,
  "shippingAddress" : {
    "addresseeGivenName" : "Marie",
    "addresseeLastName" : "Mustermann",
    "company" : "Musterbau GmbH & Co KG",
    "street" : "Kastanienallee",
    "streetNr" : "999",
    "additionalAddressInformation" : "Haus 4",
    "zip" : "90402",
    "city" : "Schwaig",
    "countryCode" : "DE",
    "state" : "Bayern"
  },
  "merchantCustomerNumber" : "cust-732477",
  "merchantOrderReferenceNumber" : "order-A12223412",
  "merchantReconciliationReferenceNumber" : "recon-A12223412",
  "merchantInvoiceReferenceNumber" : "20150112334345",
  "note" : "Ihr Einkauf bei Spielauto-Versand.",
  "sha256hashedEmailAddress" : "fxP4R-IxH1Eaxpb0f_i5Shc8-FrYrtmx5lx35f9Xzgg",
  "expiryTime" : 1800,
  "redirectUrlAfterSuccess" : "https://spielauto-versand.de/order/123/success",
  "redirectUrlAfterCancellation" : "https://spielauto-versand.de/order/123/cancellation",
  "redirectUrlAfterAgeVerificationFailure" : "https://spielauto-versand.de/order/123/underAge",
  "redirectUrlAfterRejection" : "https://spielauto-versand.de/order/123/rejection",
  "callbackUrlStatusUpdates" : "https://spielauto-versand.de/callback/status"
}
```

```
HTTP/1.1 400 Bad Request
Content-Type: application/json;charset=utf-8
```

```
{
  "messages" : [ {
    "severity" : "ERROR",
    "code" : "VALIDATION_ERROR",
    "path" : "shippingAmount",
    "reasonCode" : "INVALID_FORMAT",
    "logref" : "7a9ff422-aabd-4d06-8a81-e4328741389a:3keQiD5j4Kq"
  } ]
}
```

Links

In den Response-Daten können Links zu anderen Ressourcen enthalten sein. Die Links sind abhängig vom Kontext und Status der Ressource und geben einen Rückschluss auf mögliche Aktionen, die für eine Ressource derzeit möglich sind. Es kann sein, dass ein Link, zum Beispiel nach Ablauf einer gewissen Frist, nicht mehr enthalten ist, weil die Aktion nicht mehr möglich ist.

Die Rückgabedaten folgen dem [Hypertext Application Language \(HAL\)](#) Format.

Es wird dringend empfohlen, immer Links zu verwenden, um Aktionen durchzuführen.

Zusätzliche Felder

Zusätzliche Felder, die bei Anfragen oder Antworten übertragen werden und nicht in diesem Dokument beschrieben sind, werden vom giropay-System ignoriert. Ein Fehler oder eine Warnung wird dabei nicht zurückgegeben.

Authentifizierung

Das giropay-System ermöglicht es oneKlick-Anbietern, Requests gegen die giropay-API im Namen eines Shops (Anlage eines Checkouts) und im Namen eines Kunden (Bestätigung eines Checkouts) zu tätigen. Den Nachweis, dass sie die notwendigen Rechte besitzt, erbringt der oneKlick-Anbieter durch Vorlage eines OAuth2-Access-Tokens, in dem diese Rechte kryptographisch enkodiert sind.

Ein Access-Token kann sich der oneKlick-Anbieter vom giropay-System ausstellen lassen, indem sie sich per API-Key-Verfahren authentifiziert und die vertraulichen Autorisierungsreferenzen auf erteilte Kunden- und Händlerautorisierungen übergibt.

API-Key

API-Key

Der API-Key dient der Identifikation eines bestimmten oneKlick-Anbieters. Händler, die die oneKlick-API für einen ihrer eigenen Shops nutzen wollen, können die API-Keys ihrer Shops weiterverwenden. oneKlick-Anbieter, die kein Händler sind, können sich einen API-Key für ihre Anwendung im Portal für oneKlick-Anbieter generieren lassen. Der Anbieter muss dafür vorher durch den giropay-Support registriert werden. Es können mehrere API-Keys gleichzeitig gültig sein. Auf der Sandbox sind die ersten 8 Stellen neu erstellter API-Keys immer "00000000".

Ein API-Key wird vom giropay-System generiert und hat folgendes Format:

Länge: 36 Zeichen

Kodierung: version-4 UUID (random) [\[RFC 4122\]](#)

Beispiel: `87dbc6cd-91d2-4574-bcb5-2aaaf924386d`

Beispiel auf der Sandbox: `00000000-91d2-4574-bcb5-2aaaf924386d`

API-Secret

Das API-Secret dient der Authentifizierung des oneKlick-Anbieters. Es wird verwendet, um eine [Signatur](#) der Authentifizierungsanfrage zu erzeugen. Das API-Secret selbst wird in der Kommunikation so niemals übertragen, wodurch die Sicherheit deutlich erhöht wird. Ein API-Secret ist genau einem API-Key zugeordnet und wird zusammen mit diesem verwaltet.

Ein API-Secret wird vom giropay-System generiert und hat folgendes Format:

Länge: 44 Zeichen

Kodierung: Base64 URL Encoding [\[RFC 4648 Section 5\]](#) mit Base64 Padding

Beispiel: `9Tth0qty_9zplTy0d_QbHYvKM4iSngjoipW06VxAo=`

Access Token

Für alle Requests gegen die giropay-API ist ein OAuth2 Access Token [\[RFC 6749 1.4\]](#) erforderlich. In diesem sind die Authentifizierungsinformationen des Aufrufers gespeichert. Ein Access Token wird durch Aufruf des [\[Access-Token\]](#)-Endpoints durch Angabe des Grant Type `api_key` erzeugt und wird bei allen folgenden Requests im Request Header `Authorization` als `Bearer` angegeben:

```
Authorization: Bearer <accessToken>
```

Der Access Token hat eine begrenzte Gültigkeitsdauer. Nach Ablauf der Gültigkeit kann einfach ein neuer Access Token bezogen werden. Die Gültigkeitsdauer wird im Response des Access Token zurückgeliefert.

Bei Bedarf kann das aufrufende System mehrere Access Tokens gleichzeitig verwenden. Besteht das aufrufende System aus mehreren Knoten, so kann jeder Knoten für seine Anfragen einen eigenen Access Token verwenden. Es ist aus Performance-Gründen jedoch nicht gestattet, für jeden einzelnen Request einen neuen Access Token anzufordern.

Grant Type

Der *Grant Type* definiert die Methode, wie sich ein Benutzer oder ein System gegenüber dem giropay-System authentifiziert. Für die REST-API wird der Grant Type `api_key` verwendet.

Signatur

Die Signatur (Auth-Code) bestätigt, dass der oneKlick-Anbieter im Besitz des API-Secrets ist und gültige Anfragen an das giropay-System senden darf.

Die Signatur wird aus einem *String-to-Sign* und einem *API-Secret* erzeugt. In ersteres fließen folgende Werte in der hier angegebenen Reihenfolge ein:

Nr	Wert	Quelle	Format	Beispiel
1.	Request-ID	Request Header <code>X-Request-ID</code>	version-4 UUID (random) [RFC 4122]	ec85749b-aa36-412a-a397-2b40200c119c
2.	Datum	Request Header <code>X-Date</code> . Wichtig: Die Zeit muss der Zeitzone UTC entsprechen!	yyyyMMddHHmmss	20141231235859
3.	API-Key	Request Header <code>X-Auth-Key-TP</code>	version-4 UUID (random) [RFC 4122]	4c15310a-7936-4a19-8d80-f2b7bd95dc9b
4.	Random Nonce	Request Body-Feld <code>randomNonce</code>	Base64 URL Encoding [RFC 4648 Section 5]	Ns0WLCI3qA2AMH98wUYyhqALyVlzS0q7nHh7NdeMyvUgKJoyF0rNXkgq5fq2VNJs

Die *Random Nonce* muss pro Anfrage durch das aufrufende System per Zufallsgenerator neu generiert werden; sie fließt in die Signatur mit ein und wird im Request Body-Feld `randomNonce` an das giropay-System übertragen.

Aus den oben angegebenen Werten wird zunächst der sogenannte *String-to-Sign* gebildet. Dazu werden die Werte in der angegebenen Reihenfolge unter Trennung mittels des Trennzeichens `:` zu einer Zeichenkette verbunden (String Konkatenation). Das Trennzeichen wird nur eingefügt, sofern ein weiterer Wert folgt. Keiner der Werte darf fehlen oder als *null* kodiert werden.

Format-Beispiel für einen String-to-Sign *Z*:

```
Z = ec85749b-aa36-412a-a397-2b40200c119c:20141231235859:4c15310a-7936-4a19-8d80-f2b7bd95dc9b:Ns0WLCI3qA2AMH98wUYyhqALyVlzS0q7nHh7NdeMyvUgKJoyF0rNXkgq5fq2VNJs
```

Auf Basis eines *String-to-Sign Z* zu einem Request *R* sowie des API-Secrets *S* wird die Signatur der Anfrage mittels der Funktion HMAC-SHA-256 [RFC 2104] gebildet:

```
Signatur(Request) = new String(encodeBase64URLCompatible(HMAC-SHA-256(getBytes(Z, "UTF-8"), decodeBase64URLCompatible(getBytes(S, "UTF-8")))), "UTF-8")
```



Die Signatur wird per Base64 *URL Encoding* [RFC 4648 Section 5] kodiert. Die problematischen Zeichen **+** und **/** werden durch **-** (Minus) und **_** (Unterstrich) ersetzt.

Diese Signatur wird im Request Header *X-Auth-Code-TP* angegeben.

Im Vergleich zu Basic Auth oder einer reinen Hash-Signatur bietet das API-Key Verfahren auf Basis von HMAC einen wesentlich höheren Grad der Sicherheit, da die Signatur nicht gefälscht oder wiederverwendet werden kann und das Geheimnis in der Kommunikation niemals übertragen wird.

Access Token

```
POST /api/merchantintegration/v1/token/obtain
```

Über diesen Endpoint wird, wie oben beschrieben, ein Access Token ausgestellt. Für Requests von oneKlick-Anbietern sind die auf *-TP* endenden Headerfelder für API-Key und Signatur zu verwenden. Die Angabe weiterer API-Keys und Signaturen in anderen Headern, wie denen der Händler-API, führt immer zur Ablehnung der Anfrage, selbst wenn sie korrekt sind.

Request

Feld	Typ	Eigenschaften	Beschreibung
Header-Felder			
<i>X-Auth-Key-TP</i>	String	Pflichtfeld. 36 Zeichen.	API-Key des oneKlick-Anbieters oder API-Key eines Shops, dessen Händler für oneKlick freigeschaltet ist Zweck: Identifizierung Format: UUID Typ 4 Random [RFC 4122] Beispiel: 4c15310a-7936-4a19-8d80-f2b7bd95dc9b

Feld	Typ	Eigenschaften	Beschreibung
X-Auth-Code-TP	String	Pflichtfeld. Maximal 44 Zeichen.	Signatur der Anfrage Zweck: Authentifizierung Kodierung: Base64 URL Encoding [RFC 4648 Section 5] Beispiel: Im6MZB4ht8Dp8mXIBzPjqdgaQSZVl5C zoVzmoCnQAfk=
X-Auth-Customer-Ref	String	Optional. 44 Zeichen.	customerAuthorizationReference einer gültigen Käuferautorisierung Zweck: Rechteerweiterung des ausgestellten Tokens um die Möglichkeit, Aktionen im Namen dieses Käufers innerhalb des in der Autorisierung enthaltenen Scopes durchzuführen. Wird benötigt für oneKlick-Checkouts.
X-Auth-Merchant-Ref	String	Optional. 44 Zeichen. Wenn für diesen Aufruf ein Shop-API-Key verwendet wird, ist die Angabe der Referenz für diesen Shop nicht nötig.	merchantAuthorizationReference einer gültigen Händlerautorisierung Zweck: Rechteerweiterung des ausgestellten Tokens um die Möglichkeit, Aktionen im Namen dieses Händlers/Shops innerhalb des in der Autorisierung enthaltenen Scopes durchzuführen. Wird benötigt für Checkouts, Captures und Refunds.
X-Request-ID	String	Pflichtfeld. 36 Zeichen.	Zufällig erzeugte UUID der Anfrage Zweck: Protokollierung, Log-Statement Korrelation und Debugging Format: UUID Typ 4 Random Beispiel: ec85749b-aa36-412a-a397-2b40200c119c
X-Date	String	Pflichtfeld.	Datum und Zeit zum Sendzeitpunkt der Anfrage Format: [RFC 7231] IMF-fixdate Beispiel: Fri, 30 May 2014 09:46:48 GMT
Content-Type	String	Pflichtfeld.	Immer application/hal+json; charset=utf-8
Body-Felder			
grantType	String	Pflichtfeld.	Immer api_key

Feld	Typ	Eigenschaften	Beschreibung
<code>randomNonce</code>	String	Pflichtfeld. 64 Zeichen.	Zufällige Zeichenkette zur Diversifizierung der Request Body Inhalte Zweck: Erhöhung der Sicherheit des Verfahrens Kodierung: Base64 URL Encoding [RFC 4648 Section 5] Beispiel: Ns0WLCI3qA2AMH98wUYyhqALyVlzS0q7nHh7NdeMyvUgKJoyF0rNXkgq5fq2VNJs

Response

Feld	Typ	Eigenschaften	Beschreibung
<code>access_token</code>	String	Immer vorhanden.	OAuth2 Access Token für Anfragen gegen die giropay-API. Die Länge des Tokens ist abhängig von den in ihm codierten Datenfeldern und eine Maximallänge ist nicht vorgesehen. Sie soll deshalb bei der Implementierung des Clients nicht limitiert werden.
<code>token_type</code>	String	Immer vorhanden.	Immer <code>bearer</code> .
<code>expires_in</code>	Number	Immer vorhanden.	Gültigkeitsdauer des Access Tokens in Sekunden zur Information.
<code>scope</code>	String	Immer vorhanden.	Gültigkeitsbereich des Access Tokens zur Information.
<code>aid</code>	String	Immer vorhanden.	UUID aus 36 Zeichen zur eindeutigen Identifikation des Tokens.
<code>jti</code>	String	Immer vorhanden.	JSON Web Token ID des Access Tokens zu Debugging Zwecken.

Return Codes

Status Code	Message Code	Beschreibung
200 (Ok)		Access Token wurde erfolgreich erstellt.
400 (Bad Request)	<code>INVALID_GRANT</code>	Der <code>X-Auth-Key</code> bzw. <code>X-Auth-Code</code> fehlt oder ist ungültig.

Status Code	Message Code	Beschreibung
401 (Unauthorized)	API_KEY_IN_REQUEST_UNKNOWN	Der API-Key ist im giropay-System nicht hinterlegt.
401 (Unauthorized)	API_KEY_REQUEST_HEADER_INVALID	Die Anfrage enthält mehr als einen X-Auth-Key.
401 (Unauthorized)	API_KEY_IN_REQUEST_INACTIVE	Der API-Key ist deaktiviert.
401 (Unauthorized)	API_KEY_REQUEST_SIGNATURE_INVALID	Der X-Auth-Code ist falsch oder leer.
403 (Forbidden)	THIRD_PARTY_API_LOCKED	Der Zugang zur API ist für Sie gesperrt, bitte wenden Sie sich an den giropay-Händlersupport
403 (Forbidden)	AUTHORIZATION_INVALID	Die übergebene <code>customerAuthorizationReference</code> oder die <code>merchantAuthorizationReference</code> verweist auf eine nicht (mehr) gültige Autorisierung. Bitte prüfen Sie den Status der Autorisierungsressource (Käuferautorisierung , Händlerautorisierung) und veranlassen ggf. eine erneute Ausstellung.
403 (Forbidden)	AUTHORIZATION_NOT_STRONGLY_AUTHENTICATED	Der Käufer hat die mit der <code>customerAuthorizationReference</code> referenzierte Autorisierung zwar mit seinen Zugangsdaten bestätigt, aber noch nicht mit dem per E-Mail versandten Link final aktiviert.
403 (Forbidden)	BLOCKED_FROM_CUSTOMER_AUTHORIZATIONS	Anlage und Nutzung von Käuferautorisierungen ist für diese Anwendung blockiert.
404 (Not found)	THIRD_PARTY_AUTHORIZATION_NOT_FOUND	Eine der Autorisierungsreferenzen konnte nicht aufgelöst werden. Bitte prüfen Sie die Einträge <code>customerAuthorizationReference</code> und <code>merchantAuthorizationReference</code> .

Beispiel

```
POST /api/merchantintegration/v1/token/obtain HTTP/1.1
Content-Type: application/hal+json;charset=utf-8
Date: Mon Sep 26 14:15:06 CEST 2022
X-Request-ID: 06313db4-c36d-4969-a758-035484a1649d
Accept: application/hal+json
Content-Length: 115
X-Auth-Key-TP: 75ec4d41-d3e7-4707-b402-d833ef336f1e
X-Auth-Code-TP: __BgyabZNIyEdQp-80knUCwTNldHdW9CnchtDeDjoSM=
Host: api.paydirekt.de
```

```
{
  "grantType" : "api_key",
  "randomNonce" : "-Zxa-814cExydrWS6JP7LTqfKrZiL7s_koa7fvAGQdymp5LTALbZ8rRS7XWG7E-h"
}
```

HTTP/1.1 200 OK

Content-Type: application/hal+json;charset=utf-8

Content-Length: 2132

```
{
  "access_token" : "2ch_YAx1LXweBdkTC1Dp-
hLhMyLmUWFD4XQNYczDzQ03TYenSPyglmzKCwhtq1xNxQyUbIkNKLIQqbQxFkgea7FBNpaWiRM9gEks3bikJS
ahi2X_LZt80efgvZ5rqrN0hZj7whr6tyAmbot6ih-
rj3Tr0xQeDein6N5s7EqPPbXXhYlhCeF9L7LFNIHKsyjCanxMqCNHrZ9txwEhp8I3y7L1qltVTpozzlhp3e84V
WEiwlaskXqa5ZAEfYes-2RRScGXSSr9-
OZcoaTslsn8eSmbBi_t7p7IcT7xvtgQppzN_DOgd8DSTb4PDqEIajoMQ5i2VyMwmDHWLmtiHXPIfAvoemxuMTY
X0hj_m6jVz0TlRr15lqBIp0u96hzjz0mXd8h9fNu0RLgR5ZLSVx0e51VAKSKiPzDuw7CCCjA5T0PRPL7GMHL5m
kaVlgwLNRlhn1JQIP8E5EMCSv9ctbHpnecvISK-HIZiByDhswXj-LQ_LwA9vsC7gecAkHBJKvrOXDB8oIIICe-
qhjEI6xipkzDZbEWhv0fB0FA0ablSmXOb0zhRaD4YG02uQghsB4MCLmc-zIoarVXqQbZn3Zx_Jv-
XyoNe4vKtHQGvKY4ITew_Quiz3FK_Rf_Xs61eBSS1SpQTTQPX8Ro302XftISyCyozFKBLQ0LoScyN8vT9yl6Qxb
NG740dHslrg-8H1AniU-z-ApWJVtoe1SH6JG761xZI2Y0kWHfunI-
x15kSGqULjjMkUDacyvHpfZDew3CJ7WjyKwyHb2MeamSFfn8AwJ1zhUHwtD0qiuxYsram4PzMhsMwK8nxZerp
KJUiu1zJhdfCG74ojBlz1yVoLUVGwsqqYJu_KrZ0nokgD1vBaKztvRzsTGjm7a5vDsYZ-
DfsbY9Xbh0EZEejd0N03wLiByfINIdBwoUdUrHeEiS5xIba8ukLxkoQNmY2ihGjk4SvyCjOP3htH_t0aLV0rLf
4RMPBb4cnNqsQ43BsJVCs15XV70jm-
3J2Dudj_lIPEG7CXgnD1afCRP7B_VWLNqy8ftpwToYKmaqq_WcbHLdbzHGS9C-
SoIdqdZXYlq1qtAcCVpoYY5pu7GQ5H12ihFuRfYaG9AL9gzs5aMuMNaqWDbs1nzdwrta3FzL228U-
j0XehwusDHPd4RPep8_cCqcd5qHXrCbLxfugv601wDyUsCorhgiRI28pHmK70AtRG6LIpEsHlgzb0QmTtUykHf
NMgqCVuQmIVfdns36ZCp10DysbJtH0gDv9g1lFuALXALJ7WECBiLXe93Dms4b3SIBnz8Uj6qa8Z3m18Br05dR
xKJCqPXoxJD9Xvmya8Xiku4t68usplJP74L3xhfur5Br3QgD0qohEl4wqOamLImSIhaWc1S-
CRn1RrLjM34USHiIV0BKfx-
00KsM90UnUI6rPrmoN6mG2ECefIT3HnhFu0s9FtM38wBrBKK2k6nGem7bbA0e1s89y5AxMo_9Sfiyb_3JhwMs
EpK1MJx1V9Xw6aJDvvhx4YfFB9X-
k3LvL2GhvwXON32vQsYa0MC6MyjVd16FXpjPRXKBZDMgIrcHLBP_cwGbY7Q1F9nbPRBW12brAYXcSRtg0c9FD1
-yiRsUNvp68ijlfnNsZY2ZBvsDk00mi0rwwg_6_trCaXXyQesaTU3I5E7ks6UVhpySxlp_3dCEaWVXRtaTwpYG3
AYTgKR1tm01T27Cd1YZqwz6rcF61GAIwRab-4LZVVUA91J0AyQJNfZEBfe8En4gKOSn1xjpo6sB05z0o-
RCDmY",
  "token_type" : "bearer",
  "expires_in" : 3599,
  "scope" : "account accountsummary checkout credit merchant intermediary reporting
thirdparty thirdpartycustomerauthorization thirdpartymerchantauthorization
transaction",
  "aid" : "d1855d4a-2cff-414a-889c-b2852d75849c",
  "jti" : "ywthkqf8ksHsV7StNp9jJcB7tN8"
}
```

Beispiel (Falscher Auth-Code)

```
POST /api/merchantintegration/v1/token/obtain HTTP/1.1
Content-Type: application/json;charset=utf-8
X-Request-ID: 653a8c6f-c6d3-4279-889a-3a59cc1f81be
Content-Length: 115
X-Auth-Key-TP: 14338b5b-d856-46f0-bf8c-a8331d8d9578
X-Auth-Code-TP: hyeZravq_mh_2H305nLkwe0awd-ycWEVwICW0pxk_io=
Host: api.paydirekt.de
X-Date: Mon, 26 Sep 2022 12:15:08 GMT
```

```
{
  "grantType" : "api_key",
  "randomNonce" : "dBHmMKPGZNxG6HfzLsAB0puB0-NCw_a1E2TrjS0sLg19cMIrz268Q5nHFps_86Gw"
}
```

```
HTTP/1.1 401 Unauthorized
Content-Type: application/json;charset=utf-8
Content-Length: 170
```

```
{
  "messages" : [ {
    "severity" : "ERROR",
    "code" : "API_KEY_REQUEST_SIGNATURE_INVALID",
    "logref" : "653a8c6f-c6d3-4279-889a-3a59cc1f81be:8ZngSTG0h3p"
  } ]
}
```

Resource Access

Mit dem gültigen Access Token kann nun auf die API zugegriffen werden, z. B. um einen neuen Checkout anzulegen oder abzufragen. Das giropay-System identifiziert den Aufrufer anhand dieses Tokens.

Der Access Token wird im Header-Feld **Authorization** mit dem Prefix **Bearer** angegeben (vgl. [HTTP Header-Felder](#)).

Der Access Token ist befristet gültig (vergleiche **expires_in**) und sollte aus Performance-Gründen für alle Anfragen im Rahmen dieser Gültigkeit genutzt werden.

Beispiel (Abgelaufener Access Token)

Kurz vor Ablauf dieser Zeit sollte ein neuer Access Token **angefordert** werden. Für den Fall, dass die Gültigkeit eines Access Token aufgrund der zeitlichen Begrenzung abgelaufen ist, liefert das giropay-System auf eine fachliche Anfrage hin den HTTP Status Code **401** mit dem Message Code **ACCESS_TOKEN_EXPIRED**. Das aufrufende System muss an dieser Stelle einen neuen Access Token beziehen und den Aufruf wiederholen.

Der Body der HTTP Response enthält neben der **Message**-Struktur auch die in [\[RFC 6749\]](#)

definierten Felder `error` und `error_description`.

```
GET /api/merchantintegration/v1/some-endpoint HTTP/1.1
Date: Mon Sep 26 14:14:43 CEST 2022
Authorization: Bearer 1zvGFQY3JS4sWk4Z39eQZTX-E5cNqRuXMCdaIHxgfBPnvF-
qnfakYjlsFx0Cm45tFBfelx_400271EhWhi9h-
JUJMJ7rEs_hwFxlqzhpLlqVwPf5p8SZ7NC4MF28mq65gXmpJ4-
2EFokeK3rW0rjAa_E3seh5sVD18LYzVUS0KInPdN0bCVYe0iAYRt4lNa18VuLp4a4wmr if1G6mKhSBLysYLLGf
OSL0nDKS9Q0rxEfddkJwtfLJhdVXGE0pYw_NuG_oP26tJ9DYSsLGBTBG4sXKFuQD9BCyC-JR2sI-
BRc_GjlnZmlb9AAnPHBHT0hhvbsrhj9keY6zUbmUrmK9drzqGDnBY-wyXY0fkIa05VlnM-
ROSi7zU6MLsQS56L0rPbqxMaM8AiFy3P1fe98T07HVtS-
i6bXIWjHM78SrBQ0ZABmueHGzwS_J2L2E0TrKqFdW-
wLqcHYc6DKr9_v8GfqPwJn1XYj1hvBNG0UBqxxZ372lsuiYlaa9YeTg3vw9vpwwKY0nyfrW6Mp1T-
BKK7UIa0m1n1bByLrQpvULXbdxHuWuGJ8BjAsXHz2eqiUtr8bdTvhSWnssLIZei0HM_-
pUUmYmq4KK9D93Bg07u4vSg4v_WJnCnqml_EhcryOK4z9n5iv-
LwiFkQvgF6pSJJQ2AXMqHuFXuhsdJ7Tu94apC1_SYXQuF0Ty3Y8QJcuEMIE4mZKFsmIsaZqxbk4fqEa0kj-
Ngm6htox6ZG2eH3-9H1bX2ehCsUAL4GEVDteawM2rOBbC2FbDTcEQyf1L-
LtxF5s2PME4iq1oXrcjM92RWU3Lvs3kNKGd9L76ZxEdD8D05zfj5v_nsGcqzV01Hr0JdX6gjfVEGAfTMIkisG
-m4PFNJ0qj0A3qwwEx0MFiY70gT1N19jB4VyWTcL49cldUn1wvfxoJI27SOQTRGKD1gy15d31SNAStKHeuzbeY
x6DMdrwYGT2T77p6FHiR0zttnDjUGNWEPR1JA9XwKNrMNSC2biJ7rTfbLMYrH1CMV3F8BeHjJ70AW0yHZz0J9s
pepC358kqvFrjfNTV-ae-SM-GLED279v3WNm66lu_mBuzvTpjKTXylzE9xjIvaJ10GySbUWLMZG-
LWGj39hujsaA4Y9f0XJpoCBXnPgm0xJvtjJ_BfTs2Fkh9UHecpT885B7cbtrXxoLumaeKJWGrvYRsqsvs_U__Eh
-T1zcXcqEaRv0RglZDpci4MyABde01DLmdebHNXxtYPu0_mLa4NWG7Ag7a5h7h6Q-6ivs1R
-BTF4hJY0992Ip1KjL-z78vSfMyUpmARPFZ_S2ky5tGb1fUuig-PNkcerL1DoLGV
-W9o_1C8g0mCQZVOPNfu43vxMLMV3P3wmXUgh-SWwR4CjxiIMK-K2
-aMH160KKe_8NdveZ0fdjIrEJHki6irFOxEh17W0VQh
-fIA8yles2DieaKlKKMgbE9R67QTtelvLceHIqAVVpALuoNlnymV4bLpbjx19FC0gXL2tuq01k5nC1yJ3M__KP
GMooi08ACFEeAzmhOBnUGR-
gOFypHNBMPKDB4K1SJJmj5kBRuQ1jf7AQpHBno99XJtBvbzvIGEeXuDtLcC904y3RKRK_0FmQC6M7sCuZ19Uw
H6EDVKvZgJJ0Y8e_YeREK_9UG3NxmPS3jsCQezz7zMwIKkRvBPpvIN0kTn000EJL953EHhXLaf1ttKKiVb-
U3M13GNQa3_Nrv-0-Ex70Y_Pwmp-P-
je0F5Dzm0lzJYWJifzCtsnEKPQvVhB16BccV4_y82bi9rIWwmCfRrmOMl1lzHtR7gMEdJau0WA80YBlEKI
Host: api.paydirekt.de
```

```
HTTP/1.1 401 Unauthorized
Content-Type: application/hal+json;charset=utf-8
Content-Length: 197
```

```
{
  "messages" : [ {
    "severity" : "ERROR",
    "code" : "ACCESS_TOKEN_EXPIRED",
    "logref" : "4UyStHHp19o"
  } ],
  "error" : "invalid_token",
  "error_description" : "Access token expired"
}
```


Autorisierungen

Eine *Autorisierung (Authorization)* repräsentiert eine Vereinbarung zwischen einer *Drittanwendung* auf der einen Seite und einem Käufer oder Händler auf der anderen Seite. Bei einer Drittanwendung handelt es sich entweder um einen Shop eines bei giropay aktiven und für oneKlick freigeschalteten Händler oder er wurde von paydirekt-Mitarbeitern individuell als Drittanwendung hinterlegt. Eine Autorisierung definiert eine Menge von Rechten (*Scopes*), die eine Drittanwendung berechtigen, entsprechende Funktionen der giropay-API im Namen des Käufers oder Händlers auszuführen. In den folgenden Kapiteln wird beschrieben, wie Autorisierungen angelegt und abgefragt werden können.

Anlage Käuferautorisierung

```
POST /api/thirdpartycustomerauthorization/v1/authorizations
```

Bei der Anlage von Käuferautorisierungen muss ein Access Token verwendet werden, welches ohne eine Referenz auf eine Käufer- oder Händlerautorisierung erzeugt wurde. Ausnahme ist der CPSP. Dieser muss bei der Authentifizierung zwingend eine `merchantAuthorizationReference` angeben haben um zu identifizieren, welchem seiner Händler die neue Käuferautorisierung zuzuordnen ist.

Request

Feld	Typ	Eigenschaften	Beschreibung
<code>scopes</code>	Array	Pflichtfeld. Derzeit implementiert: <code>PERFORM_CHECKOUT</code> , <code>PERFORM_RECURRING_PAYMENT</code> , <code>READ_ACCOUNT_STATUSES</code> , <code>READ_CREDIT</code> , <code>READ_TRANSACTIONS</code> , <code>READ_USERNAME</code>	Umfang der vom Käufer zu erteilenden Berechtigungen.
<code>deviceId</code>	String	Entweder der Device-Fingerprint oder die Device-ID muss angegeben werden. Maximal 50 Zeichen.	Eindeutige Device-ID zur Identifikation des Endgeräts des Käufers.
<code>deviceName</code>	String	Pflichtfeld. Maximal 50 Zeichen.	Name des Endgeräts des Käufers. Wird dem Käufer in seinen Portalen angezeigt.

Feld	Typ	Eigenschaften	Beschreibung
<code>redirectUrlAfterSuccess</code>	String	Optionales Feld. Muss gesetzt sein, wenn <code>redirectUrlAfterCancellation</code> und <code>redirectUrlAfterFailure</code> gesetzt sind.	Die Rücksprungadresse, die aufgerufen wird, nachdem die Autorisierung erfolgreich durch den Kunden bestätigt worden ist.
<code>redirectUrlAfterCancellation</code>	String	Optionales Feld. Muss gesetzt sein, wenn <code>redirectUrlAfterSuccess</code> und <code>redirectUrlAfterFailure</code> gesetzt sind.	Die Rücksprungadresse, die aufgerufen wird, nachdem der Kunde die Bestätigung der Autorisierung abgebrochen hat.
<code>redirectUrlAfterFailure</code>	String	Optionales Feld. Muss gesetzt sein, wenn <code>redirectUrlAfterSuccess</code> und <code>redirectUrlAfterCancellation</code> gesetzt sind.	Die Rücksprungadresse, die aufgerufen wird, nachdem die Bestätigung der Autorisierung durch einen technischen Fehler abgebrochen worden ist.
<code>contextInformation.deviceIpAddress</code>	String	Pflichtfeld.	IP-Adresse des Endgeräts des Käufers zum Zeitpunkt der Autorisierungsanlage.
<code>contextInformation.deviceFingerprint</code>	String	Entweder der Fingerprint oder die Device-ID muss angegeben werden.	Fingerprint des Endgeräts des Käufers zum Zeitpunkt der Autorisierungsanlage.
<code>usageAgreement</code>	Object	Optional.	Wenn die Autorisierung für regelmäßig wiederkehrende Bezahlvorgänge verwendet wird, ist die Art und Häufigkeit in dieser Struktur zu beschreiben.
<code>usageAgreement.amount</code>	Number	Pflichtfeld, wenn <code>usageAgreement</code> angegeben wird.	Der regelmäßig abgerufene Betrag.
<code>usageAgreement.currency</code>	String	Pflichtfeld, wenn <code>usageAgreement</code> angegeben wird.	Die Währung - immer "EUR".
<code>usageAgreement.startDate</code>	String	Pflichtfeld, wenn <code>usageAgreement</code> angegeben wird.	Das Datum der erstmaligen Bezahlung im Format "YYYY-MM-DD" - z.B. "2017-12-30".

Feld	Typ	Eigenschaften	Beschreibung
<code>usageAgreement.recurrenceInterval.intervalUnit</code>	String	Pflichtfeld, wenn <code>usageAgreement</code> angegeben wird.	Die Einheit des Bezahl-Intervalls. Mögliche Werte sind: "DAY", "WEEK", "MONTH", "QUARTER", "YEAR"
<code>usageAgreement.recurrenceInterval.interval</code>	Number	Pflichtfeld, wenn <code>usageAgreement</code> angegeben wird.	Die Länge des Bezahl-Intervalls.
<code>callbackUrlStatusUpdates</code>	String	Optional.	URL des Endpoints, der Mitteilungen zu Statusänderungen der Autorisierung empfangen soll.
<code>checkoutType</code>	String	Optional	Die Art der Bestellung: <code>DIRECT_SALE</code> , <code>ORDER</code> oder <code>ORDER_SECURED</code> . Das Feld kann mitgegeben werden, wenn durch den oneKlick-Anbieter immer nur eine Art des Checkouts verwendet wird. Dies empfiehlt sich vor allem bei der gesicherten Vorbestellung (<code>ORDER_SECURED</code>). Dadurch kann bereits zum Zeitpunkt der Autorisierungserteilung festgestellt werden, ob die Bank des Kunden diese Art von Checkouts unterstützt und betroffenen Kunden so schon in diesem Schritt mitgeteilt werden, dass dieses Feature von ihrer Bank nicht unterstützt wird.

Response

In der Response werden Informationen zur angelegten Authorization zurückgegeben. Die Struktur ist identisch mit dem Authorization Endpoint [GET](#)-Aufruf.

Links

Relation	Description
<code>self</code>	Link zu dieser Autorisierung. Es wird empfohlen, ausschließlich diesen Link zur Abfrage der Autorisierung zu verwenden.
<code>web</code>	Link zum Autorisierungsportal für diese Autorisierung. Der Käufer kann auf diesen Link weitergeleitet werden, um die Autorisierung zu bestätigen. Der Link ist an Autorisierungen im Status <code>OPEN</code> immer vorhanden.

Return Codes

Status Code	Message Code	Beschreibung
201 (Created)		Die Authorization wurde angelegt. Anschließend muss ein Redirect auf den in der Resource hinterlegten Link web durchgeführt werden.
400 (Bad Request)	VALIDATION_ERROR	Der Request ist syntaktisch ungültig.
400 (Bad Request)	CONVERSION_ERROR	Der Request enthält ungültige Zeichen. Häufigste Ursache hierfür ist, dass der Request nicht korrekt UTF-8 codiert ist. Falls die ungültigen Zeichen einem Attribut im Request zugeordnet werden können, wird dieses im Feld path genannt und der ungültige Wert im Feld content übermittelt.
401 (Unauthorized)	ACCESS_TOKEN_EXPIRED	Siehe Resource Access .
401 (Unauthorized)	UNAUTHORIZED	Der Zugriff wurde verweigert. Bitte wenden Sie sich an den giropay-Support.
403 (Forbidden)	BLOCKED_FROM_CUSTOMER_AUTHORIZATIONS	Anlage und Nutzung von Käuferauthorisierungen ist für diese Anwendung blockiert.
404 (Not Found)	THIRD_PARTY_NOT_FOUND	Der angefragte oneKlick-Anbieter existiert nicht oder ist nicht zugreifbar. Wenn dies trotz gültigen Credentials auftritt, wenden Sie sich bitte an den giropay-Support.
422 (Unprocessable Entity)	THIRD_PARTY_NOT_ACTIVE	Der oneKlick-Anbieter ist (noch) gesperrt, es können keine Authorizations angelegt werden.
422 (Unprocessable Entity)	SCOPE_NOT_SUPPORTED	Der oneKlick-Anbieter hat nicht das Recht, das angeforderte Funktionalität anzufordern.

Beispiel

POST /api/thirdpartycustomerauthorization/v1/authorizations HTTP/1.1

Content-Type: application/json;charset=utf-8

User-Agent: mobile

Authorization: Bearer <access_token>

```
{
  "scopes" : [ "PERFORM_CHECKOUT" ],
  "deviceId" : "SomeDevice",
  "deviceName" : "SomeDeviceName",
  "redirectUrlAfterSuccess" : "https://one.klick/success?id=123-456-789",
  "redirectUrlAfterCancellation" : "https://one.klick/cancelled?id=123-456-789",
  "redirectUrlAfterFailure" : "https://one.klick/failed?id=123-456-789",
  "contextInformation" : {
    "deviceIpAddress" : "127.0.0.1",
    "deviceFingerprint" : "ae8faa7f67709ebab82f579945c7e6b5"
  },
  "usageAgreement" : {
    "amount" : {
      "amount" : 10,
      "currency" : "EUR"
    },
    "startDate" : "2017-12-30",
    "recurrenceInterval" : {
      "intervalUnit" : "WEEK",
      "interval" : 1
    }
  },
  "callbackUrlStatusUpdates" : "https://one.klick/callbackUrlStatusUpdates?id=123-456-789",
  "checkoutType" : "ORDER_SECURED"
}
```

HTTP/1.1 201 Created

Content-Type: application/hal+json;charset=utf-8

```
{
  "customerAuthorizationId" : "3d712d42-e409-4d76-9ef8-4c46dafd5321",
  "thirdPartyId" : "639cd159-721c-4a62-9d0b-e50da9958425",
  "clientId" : "a5605dd5-a864-4d96-b5be-02b1d0e36597",
  "creationTimestamp" : "2022-10-24T08:31:51.290Z",
  "scopes" : [ "PERFORM_CHECKOUT" ],
  "status" : "OPEN",
  "token" : "5440a24b-0b56-4d41-9615-04422e7f5c8a",
  "deviceId" : "SomeDevice",
  "deviceName" : "SomeDeviceName",
  "redirectUrlAfterSuccess" : "https://one.klick/success?id=123-456-789",
  "redirectUrlAfterCancellation" : "https://one.klick/cancelled?id=123-456-789",
  "redirectUrlAfterFailure" : "https://one.klick/failed?id=123-456-789",
  "thirdPartyName" : "AwesomeCompany",
  "clientName" : "someClientName",
  "usageAgreement" : {
    "amount" : {
      "amount" : 10,
      "currency" : "EUR"
    },
    "startDate" : "2017-12-30",
    "recurrenceInterval" : {
      "intervalUnit" : "WEEK",
      "interval" : 1
    }
  },
  "callbackUrlStatusUpdates" : "https://one.klick/callbackUrlStatusUpdates?id=123-456-789",
  "checkoutType" : "ORDER_SECURED",
  "_links" : {
    "self" : {
      "href" :
"https://api.paydirekt.de/api/thirdpartycustomerauthorization/v1/authorizations/3d712d42-e409-4d76-9ef8-4c46dafd5321"
    },
    "web" : {
      "href" : "http://localhost/authorization/#/5440a24b-0b56-4d41-9615-04422e7f5c8a"
    }
  }
}
```

Verhalten bei nicht angegebenen Rücksprungadressen

Wenn bei der Anlage einer Käuferautorisierung die Felder `redirectUrlAfterSuccess`, `redirectUrlAfterCancellation` und `redirectUrlAfterFailure` nicht übergeben werden, gibt es zwei mögliche Optionen für das Rücksprungverhalten.

Angabe des URL-Query-Parameters appId

Wenn der **web**-HAL-Link der Autorisierung um den URL-Query-Parameter **appId** ergänzt wird, wie beispielsweise `https://www.paydirekt.de/authorization-redesign/5b9fbc55-7041-4bf3-989a-3fe2613bf293?appId=2613bf2`, dann wird automatisch eine Rücksprungadresse der Form `paydirekt-authorization://authorization?result=${result}&appId=${appId}` verwendet. Für den Erfolgsfall wird hierbei **result** mit **approved** befüllt, für den Abbruchfall mit **cancelled** und dem Fehlerfall mit **error**. Für das Beispiel `https://www.paydirekt.de/authorization-redesign/5b9fbc55-7041-4bf3-989a-3fe2613bf293?appId=2613bf2` ergibt sich somit die Rücksprungadresse `paydirekt-authorization://authorization?result=approved&appId=2613bf2` für den Erfolgsfall.

Diese alternativen Rücksprungadressen können so beispielsweise unter Android Apps verwendet werden um eine Webview oder einen Chrome Custom Tab zu schließen. Dafür muss die App sich für das Scheme `paydirekt-authorization` registrieren und Aufrufe solcher URLs geeignet interpretieren. Es wird empfohlen, für jeden Autorisierungsvorgang den Query-Parameter **appId** mit einem zufällig generierten String zu belegen. Dieser String wird der App über die Rücksprungadresse wieder zurückgegeben und kann von dieser überprüft werden, um sicherzustellen, dass der Aufruf der Adresse aus dem Autorisierungsportal stammt, und nicht von einer dritten App ausgelöst worden ist.

Keine Angabe des URL-Query-Parameters appId

Falls an den **web**-Link von der Autorisierungsanlage kein URL-Query-Parameter **appId** angehängt wird und auch keine Rücksprungadressen gesetzt werden, dann wird dem Käufer im Autorisierungsportal von giropay keine Möglichkeit in der Weboberfläche geboten zum Händler zurückzuspringen. Beispielsweise fehlt dann der Button **Abbrechen und zurück zu ...** in der Weboberfläche.

Abfrage Käuferautorisierung

```
GET /api/thirdpartycustomerauthorization/v1/authorizations/{authorizationId}
```

Eine zuvor angelegte Autorisierung kann über die GET-Methode aufgerufen werden, um ihren Status und andere Eigenschaften abzufragen sowie den für die spätere Nutzung nötigen **customerAuthorizationReference** zu erhalten. Diese Resource sollte immer über den bei der **Anlage** erhaltenen Link aufgerufen werden.

Für die Abfrage von Käuferautorisierungen muss wie bei der Anlage ein Access Token verwendet werden, welches ohne eine Referenz auf eine Käufer- oder Händlerautorisierung erzeugt wurde. Ausnahme ist der CPSP. Dieser muss bei der Authentifizierung zwingend eine **merchantAuthorizationReference** angegeben haben.

Response

Feld	Typ	Eigenschaften	Beschreibung
customerAuthorizationId	String	Immer vorhanden.	ID der Autorisierungsresource.
customerAuthorizationReference	String	Vorhanden, sobald die Autorisierung den Status APPROVED erreicht hat.	Vertrauliche Referenz für die Autorisierung. Wird benötigt, wenn der oneKlick-Anbieter Aktionen unter Verwendung der Autorisierung ausführen möchte.
thirdPartyId	String	Immer vorhanden.	ID des oneKlick-Anbieters.
thirdPartyName	String	Optional.	Name des oneKlick-Anbieters.
clientId	String	Immer vorhanden.	ID der oneKlick-Anwendung.
clientName	String	Optional.	Name der oneKlick-Anwendung.
accountId	String	Vorhanden, sobald die Autorisierung den Status APPROVED erreicht hat.	ID des Käuferaccounts, der die Autorisierung erteilt hat.
deviceId	String	Mindestens eines der Felder Fingerprint und Device-ID ist vorhanden.	Eindeutige Device-ID zur Identifikation des Endgeräts des Käufers.
deviceName	String	Vorhanden, falls er bei der Anlage übergeben wurde.	Name des Endgeräts des Käufers. Wird dem Käufer in seinen Portalen angezeigt.
stronglyAuthenticated	Boolean	Vorhanden, nachdem die Autorisierung durch den Kunden durch einen ersten Faktor bestätigt worden ist.	Gibt an, ob der Benutzer sich für die Bestätigung der Autorisierung bereits über einen zweiten Faktor stark authentifiziert hat. Autorisierungen ohne starke Authentifizierung sind nur für eine kurze Zeit nach ihrer Bestätigung durch den ersten Faktor gültig.
creationTimestamp	String	Immer vorhanden.	Zeitpunkt, zu dem die Autorisierung erstellt wurde. Formatierung entsprechend ISO-8601 durch den Standard Java DateTimeFormatter . Abweichend davon werden immer drei Nachkommastellen für die Millisekunden ausgegeben.

Feld	Typ	Eigenschaften	Beschreibung
<code>approvalTimestamp</code>	String	Vorhanden, sobald die Autorisierung den Status <code>APPROVED</code> erreicht hat.	Zeitpunkt, zu dem die Autorisierung vom Käufer erteilt wurde. Formatierung wie <code>creationTimestamp</code> .
<code>revokeTimestamp</code>	String	Vorhanden, sobald die Autorisierung den Status <code>REVOKED</code> erreicht hat.	Zeitpunkt, zu dem die Autorisierung zurückgezogen wurde. Formatierung wie <code>creationTimestamp</code> .
<code>scopes</code>	Array	Immer vorhanden.	Umfang der vom Käufer zu erteilenden Berechtigungen.
<code>status</code>	String	Immer vorhanden.	Status der Autorisierung. Mögliche Werte sind: <code>OPEN</code> : Angelegt, aber noch nicht vom Käufer bestätigt. <code>APPROVED</code> : Vom Käufer bestätigt. <code>EXPIRED</code> : Wurde nicht innerhalb von 5 Minuten erteilt oder durch eine neue, inhaltlich gleiche Autorisierung ersetzt. <code>REVOKED</code> : Wurde zurückgezogen. Dies kann durch den Kunden selbst oder einen giro-pay-Mitarbeiter geschehen sein. <code>CHECKOUT_TYPE_NOT_SUPPORTED_BY_BANK</code> : Bei der Autorisierungsanlage wurde der Checkout-Typ <code>ORDER_SECURED</code> übergeben. Dieser wird von der Bank des Kunden nicht unterstützt.
<code>redirectUrlAfterSuccess</code>	String	Vorhanden, falls bei der Anlage übergeben.	Die Rücksprungadresse, die aufgerufen wird, nachdem die Autorisierung erfolgreich durch den Kunden bestätigt worden ist.
<code>redirectUrlAfterCancellation</code>	String	Vorhanden, falls bei der Anlage übergeben.	Die Rücksprungadresse, die aufgerufen wird, nachdem der Kunde die Bestätigung der Autorisierung abgebrochen hat.
<code>redirectUrlAfterFailure</code>	String	Vorhanden, falls bei der Anlage übergeben.	Die Rücksprungadresse, die aufgerufen wird, nachdem die Bestätigung der Autorisierung durch einen technischen Fehler abgebrochen worden ist.

Feld	Typ	Eigenschaften	Beschreibung
<code>usageAgreement</code>	Object	Nur bei Verwendung der Autorisierung für wiederkehrende Zahlungen.	Wenn die Autorisierung für regelmäßig wiederkehrende Bezahlvorgänge verwendet wird, ist die Art und Häufigkeit in dieser Struktur zu beschreiben.
<code>usageAgreement.amount</code>	Number	Vorhanden, wenn <code>usageAgreement</code> vorhanden ist.	Der regelmäßig abgerufene Betrag.
<code>usageAgreement.amount.currency</code>	String	Vorhanden, wenn <code>usageAgreement</code> vorhanden ist.	Die Währung - immer "EUR".
<code>usageAgreement.startDate</code>	String	Vorhanden, wenn <code>usageAgreement</code> vorhanden ist.	Das Datum der erstmaligen Bezahlung im Format "YYYY-MM-DD" - z.B. "2017-12-30".
<code>usageAgreement.recurrenceInterval.intervalUnit</code>	String	Vorhanden, wenn <code>usageAgreement</code> vorhanden ist.	Die Einheit des Bezahl-Intervalls. Mögliche Werte sind: "DAY", "WEEK", "MONTH", "QUARTER", "YEAR"
<code>usageAgreement.recurrenceInterval.interval</code>	Number	Vorhanden, wenn <code>usageAgreement</code> vorhanden ist.	Die Länge des Bezahl-Intervalls.
<code>callbackUrlStatusUpdates</code>	String	Vorhanden, falls bei der Anlage übergeben.	URL des Endpoints, der Mitteilungen zu Statusänderungen der Autorisierung empfangen soll.
<code>checkoutType</code>	String	Vorhanden, falls bei der Anlage übergeben.	Die Art der Bestellung: DIRECT_SALE, ORDER oder ORDER_SECURED. Das Feld kann mitgegeben werden, wenn durch den oneKlick-Anbieter immer nur eine Art des Checkouts verwendet wird. Dies empfiehlt sich vor allem bei der gesicherten Vorbestellung (ORDER_SECURED). Dadurch kann bereits zum Zeitpunkt der Autorisierungserteilung festgestellt werden, ob die Bank des Kunden diese Art von Checkouts unterstützt und betroffenen Kunden so schon in diesem Schritt mitgeteilt werden, dass dieses Feature von ihrer Bank nicht unterstützt wird.

Links

Relation	Description
<code>self</code>	Link zu dieser Autorisierung. Es wird empfohlen, ausschließlich diesen Link zur Abfrage der Autorisierung zu verwenden.

Return Codes

Status Code	Message Code	Beschreibung
200 (Ok)		Die Authorization wurde gefunden.
401 (Unauthorized)	<code>ACCESS_TOKEN_EXPIRED</code>	Siehe Resource Access .
404 (Not Found)	<code>THIRD_PARTY_AUTHORIZATION_NOT_FOUND</code>	Die angefragte Authorization existiert nicht oder ist nicht zugreifbar.
404 (Not Found)	<code>THIRD_PARTY_NOT_FOUND</code>	Der angefragte oneKlick-Anbieter existiert nicht oder ist nicht zugreifbar. Wenn dies trotz gültigen Credentials auftritt, wenden Sie sich bitte an den giro-pay-Support.

Beispiel (Authorization nach Approval durch Kunden)

```
GET /api/thirdpartycustomerauthorization/v1/authorizations/e4c3e78a-10b6-4099-a5cd-59ee3c7a6fc8 HTTP/1.1
Content-Type: application/json;charset=utf-8
Authorization: Bearer <access_token>
```

HTTP/1.1 200 OK

Content-Type: application/hal+json;charset=utf-8

```
{
  "customerAuthorizationId" : "e4c3e78a-10b6-4099-a5cd-59ee3c7a6fc8",
  "customerAuthorizationReference" : "3mmH_XWNzay1TSR0UhDHN2n8_HN0BzBcbMxco_BOHqI=",
  "thirdPartyId" : "639cd159-721c-4a62-9d0b-e50da9958425",
  "clientId" : "a5605dd5-a864-4d96-b5be-02b1d0e36597",
  "creationTimestamp" : "2022-10-24T08:31:49.974Z",
  "approvalTimestamp" : "2022-10-24T08:31:49.974Z",
  "revokeTimestamp" : "2022-10-24T08:31:49.974Z",
  "scopes" : [ "PERFORM_CHECKOUT" ],
  "status" : "REVOKED",
  "token" : "dd1fedb6-b8a8-4e24-8105-02bbc0f2b573",
  "accountId" : "78a27b6d-a668-4563-bda3-ab3b0c9bde55",
  "deviceId" : "SomeDevice",
  "deviceName" : "SomeDeviceName",
  "redirectUrlAfterSuccess" : "https://one.klick/success?id=123-456-789",
  "redirectUrlAfterCancellation" : "https://one.klick/cancelled?id=123-456-789",
  "redirectUrlAfterFailure" : "https://one.klick/failed?id=123-456-789",
  "thirdPartyName" : "AwesomeCompany",
  "clientName" : "someClientName",
  "stronglyAuthenticated" : false,
  "usageAgreement" : {
    "amount" : {
      "amount" : 10,
      "currency" : "EUR"
    },
    "startDate" : "2017-12-30",
    "recurrenceInterval" : {
      "intervalUnit" : "WEEK",
      "interval" : 1
    }
  },
  "callbackUrlStatusUpdates" : "https://one.klick/callbackUrlStatusUpdates?id=123-456-789",
  "checkoutType" : "ORDER_SECURED",
  "_links" : {
    "self" : {
      "href" :
        "https://api.paydirekt.de/api/thirdpartycustomerauthorization/v1/authorizations/e4c3e78a-10b6-4099-a5cd-59ee3c7a6fc8"
    }
  }
}
```

Callback für Statusupdates

Wenn bei der Anlage einer Käuferautorisierung eine Callback URL `callbackUrlStatusUpdates`

übergeben wird, sendet das giropay-System bei Statusänderungen der Autorisierung an diese URL einen HTTP-Request. Bei Verbindungsfehlern oder Responses mit einem HTTP-Statuscode von 5xx wird das giropay-System die Übermittlung der Statusänderung in einem Zeitfenster von 24 Stunden bis zu 5 Mal wiederholen. Statusänderungen werden dabei immer in der Reihenfolge übermittelt, in der sie an der Autorisierung aufgetreten sind. Kann eine Statusänderung nicht übermittelt werden, so werden auch alle nachfolgenden Statusänderungen nicht übermittelt, bis entweder die fehlgeschlagene Übermittlung nachgeholt werden konnte oder alle wiederholten Versuche fehlgeschlagen sind.

POST callbackUrlStatusUpdates

Request

customerAuthorization/callback/status-update

Feld	Typ	Eigenschaften	Beschreibung
customerAuthorizationId	String	Immer vorhanden.	Eindeutige UUID aus 36 Zeichen für diese Käuferautorisierung. Wird vom giropay-System vergeben.
status	String	Immer vorhanden.	Status der Autorisierung: OPEN , APPROVED , EXPIRED oder REVOKED . OPEN : Angelegt, aber noch nicht vom Käufer bestätigt. APPROVED : Vom Käufer bestätigt. EXPIRED : Wurde nicht innerhalb von 5 Minuten erteilt oder durch eine neue, inhaltlich gleiche Autorisierung ersetzt. REVOKED : Wurde zurückgezogen. Dies kann durch den Kunden selbst oder einen paydirekt Mitarbeiter geschehen sein.
stronglyAuthenticated	Boolean	Immer vorhanden.	Gibt an, ob der Benutzer sich für die Bestätigung der Autorisierung bereits über einen zweiten Faktor stark authentifiziert hat. Autorisierungen ohne starke Authentifizierung sind nur für eine kurze Zeit nach ihrer Bestätigung durch den ersten Faktor gültig.
statusUpdateTimeamp	String	Immer vorhanden.	Zeitstempel der Statusänderung.

Feld	Typ	Eigenschaften	Beschreibung
sequenceNumber	Number	Immer vorhanden.	Sequenznummer der Statusänderung. Anhand der Sequenznummer kann ein Shop die Reihenfolge der Statusänderungen nachvollziehen und gegebenenfalls fehlende Statusänderungen erkennen.

Beispiel

```
POST /webhooktest/v1/customerauthorization/status HTTP/1.1
Content-Type: application/json;charset=utf-8
```

```
{
  "customerAuthorizationId" : "737a8b3d-09f3-4275-96f6-023e477fd609",
  "status" : "APPROVED",
  "stronglyAuthenticated" : false,
  "statusUpdateTimestamp" : "2020-01-01T12:00:00.000Z",
  "sequenceNumber" : 2
}
```

Response

Die Response enthält keinen Body und wird vom giropay-System nicht weiter verarbeitet. Der HTTP-Statuscode sollte dennoch für etwaige Serviceanfragen Rückschlüsse auf die Verarbeitung der Status-Benachrichtigung zulassen.

Beispiel

```
HTTP/1.1 200 OK
```

Anlage Käuferautorisierung während eines Checkouts

Eine Käuferautorisierung kann alternativ zur ihrer [direkten Anlage](#) auch während eines [Checkouts](#) angelegt werden. Das giropay-System legt in diesem Fall zu Gunsten derselben oneKlick-Anwendung, die den Checkout angelegt hat, eine neue Käuferautorisierung an und bietet dem Käufer im Checkout-Prozess an, diese zu bestätigen. Die oneKlick-Anwendung kann die derart angelegte Autorisierung im Anschluss [abrufen](#) und mit ihr so weiter verfahren, als würde sie direkt angelegt und vom Käufer bestätigt.

Um eine Käuferautorisierung während eines Checkouts anlegen zu lassen, muss die oneKlick-Anwendung den Request zur Anlage des Checkouts um Attribute über die anzulegende Autorisierung erweitern. Diese Attribute sind im Kapitel [Checkout Anlage](#) in der Feldtabelle und den Beispielen ersichtlich. Für die erfolgreiche Anlage der Autorisierung muss die oneKlick-Anwendung dieselben Voraussetzungen wie für die [direkte Anlage](#) erfüllen. Ist dies nicht der Fall,

wird der Checkout trotzdem vom giropay-System akzeptiert, aber keine Autorisierung angelegt.

Wenn der Käufer sich dazu entscheidet, oneKlick einzurichten, wird an dem Checkout ein HAL-Link auf die dafür angelegte Käuferautorisierung hinterlegt. Der Link ist im entsprechenden Abschnitt des Kapitels [Checkout Abfrage](#) beschrieben. Über diesen Link kann die oneKlick-Anwendung die Autorisierung selbstständig (z.B. per Polling oder nach einem gewissen Zeitraum) abrufen und prüfen. Der Käufer wird am Ende des Gesamtprozesses auf die bei der Checkout-Anlage angegebene Redirect-URL für einen erfolgreich abgeschlossenen Checkout zurückgeleitet, auch wenn er keine Autorisierung erteilt hat, den Checkout selbst aber erfolgreich abgeschlossen hat.

Duplikate von Käuferautorisierungen

Wenn ein Käufer eine Käuferautorisierung bestätigt, werden automatisch alle eventuell vorhandenen alten Duplikate der neuen Autorisierung in den Zustand **EXPIRED** überführt und können danach nicht mehr verwendet werden. Eine Käuferautorisierung gilt als Duplikat einer anderen Käuferautorisierung, wenn **thirdPartyId**, **clientId**, **deviceId** und **scopes** identisch sind. Dies soll sicherstellen, dass der Käufer in seinem Käuferportal keine redundanten Autorisierungen sieht. oneKlick-Anwendungen können somit immer nur die zuletzt angelegte und bestätigte Käuferautorisierung verwenden.

Authentifizierung

Zur Sicherheit des Käufers müssen Käuferautorisierungen von ihm über einen zweiten Faktor bestätigt werden. Zu diesem Zweck wird der Käufer über die Website des Autorisierungsdialogs in die giropay-App geführt. In der giropay-App kann der Käufer die Details der Autorisierung einsehen und diese bestätigen.



Bei der Einbindung dieses Prozesses in eine Android oder iOS App gibt es einige Fallstricke zu beachten, besonders bei der Verwendung von WebViews. Zudem wird jeweils eine Test-App für Android und iOS angeboten, damit die Integration gegen eine Testumgebung getestet werden kann. Weitere Informationen und Zugang über: haendler@giropay.de

Bei einer erfolgreichen Bestätigung ändert sich das Feld **stronglyAuthenticated** zu **true**, sowie das Feld **status** auf **APPROVED**. Eine Autorisierung kann erst benutzt werden, wenn sie erfolgreich bestätigt wurde.



Das Feld **stronglyAuthenticated** sollte nicht mehr ausgewertet werden, stattdessen sollte stets das Feld **status** zur Auswertung des Zustandes der Autorisierung herangezogen werden.

Bestätigte Autorisierungen, sowie Autorisierungen welche noch auf eine Antwort des Käufers warten, können jederzeit in der giropay-App eingesehen und nachträglich bestätigt oder abgelehnt werden. Nach 30 Minuten ohne Bestätigung oder Ablehnung des Käufers geht eine Autorisierung in den Status **EXPIRED** über.

Zurückziehen von Käuferautorisierungen

```
DELETE /api/thirdpartycustomerauthorization/v1/authorizations/{authorizationId}
```

Eine zuvor angelegte Autorisierung kann über die DELETE-Methode zurückgezogen werden, dadurch wird ihr Status auf **REVOKED** gesetzt.

Für das Zurückziehen von Käuferautorisierungen muss wie bei der Anlage oder Abruf ein gültiges Access Token verwendet werden.

Beispiel (Authorization nach Approval durch Kunden)

```
DELETE /api/thirdpartycustomerauthorization/v1/authorizations/d5bc3a01-1f6d-4ad3-99ff-c465291ecf0d HTTP/1.1
Content-Type: text/plain;charset=utf-8
Authorization: Bearer <access_token>
```

```
HTTP/1.1 204 No Content
```

Return Codes

Status Code	Message Code	Beschreibung
204 (Ok)		Die Autorisierung wurde zurückgezogen.
401 (Unauthorized)	ACCESS_TOKEN_EXPIRED	Siehe Resource Access .
404 (Not Found)	THIRD_PARTY_AUTHORIZATION_NOT_FOUND	Die angefragte Autorisierung existiert nicht oder ist nicht zugreifbar.
404 (Not Found)	THIRD_PARTY_NOT_FOUND	Der angefragte oneKlick-Anbieter existiert nicht oder ist nicht zugreifbar. Wenn dies trotz gültigen Credentials auftritt, wenden Sie sich bitte an den giropay-Support.

Autorisierungsdialog

Der Benutzer erteilt eine Autorisierung über einen vom giropay-System bereitgestellten Autorisierungsdialog. Der Autorisierungsdialog ist eine Website, die von einer oneKlick-Anwendung direkt aufgerufen werden kann.

Der Link zum Autorisierungsdialog für eine Autorisierung ist als HAL-Link in dieser enthalten.

Wenn bei der Anlage einer Autorisierung Rücksprungadressen übergeben worden sind, dann leitet der Autorisierungsdialog den Benutzer nach dem Abschluss des Autorisierungsprozess auf die dem Ergebnis des Prozesses entsprechende Rücksprungadresse weiter. Dabei kann der Benutzer sowohl auf eine Website, als auch auf eine mobile App zurückgeleitet werden (siehe [Verhalten bei nicht](#)

angegebenen Rücksprungadressen).



Der Aufruf einer Rücksprungadresse ist keine zuverlässige Informationsquelle für die Statusänderung einer Autorisierung. Webbrowseroptimierungen können diese Rücksprungadressen aufrufen, ohne dass der Käufer jemals aktiv den Rücksprung durchgeführt hat. Zum Nachverfolgen von Statusänderungen siehe [Callback für Statusupdates](#).

Anlage Händlerautorisierung



Die Verwaltung von Händlerautorisierungen ist nur für individuell angelegte oneKlick-Anwendungen relevant. Händler, die für ihre eigenen Shops oneKlick durchführen, benötigen niemals eine `merchantAuthorizationReference`. CPSPs finden sie in der Händlerverwaltung im Intermediärportal.

Die Anlage von Händlerautorisierungen wird von einem paydirekt-Mitarbeiter durchgeführt. Der oneKlick-Anbieter kann diese in seinem Portal einsehen. Dort erfährt er u.a. deren `merchantAuthorizationReference` und die ID, mit der ein Aufruf des Endpunkts [Abfrage Händlerautorisierung](#) ausgeführt werden kann.

Abfrage Händlerautorisierung

```
GET /api/thirdpartymerchantauthorization/v1/authorizations/{authorizationId}
```

Eine bereits vorhandene Händlerautorisierung kann über die GET-Methode abgefragt werden, um ihre Eigenschaften sowie die für die spätere Nutzung nötige `merchantAuthorizationReference` zu erhalten.

Response

Feld	Typ	Eigenschaften	Beschreibung
<code>merchantAuthorizationId</code>	String	Immer vorhanden.	ID der Autorisierungsresource.
<code>merchantAuthorizationReference</code>	String	Vorhanden, sobald die Autorisierung den Status <code>APPROVED</code> erreicht hat.	Vertrauliche Referenz für die Autorisierung. Wird benötigt, wenn der oneKlick-Anbieter unter Verwendung der Autorisierung ausführen möchte.
<code>thirdPartyId</code>	String	Immer vorhanden.	ID des oneKlick-Anbieters.
<code>clientId</code>	String	Immer vorhanden.	ID des Clients.

Feld	Typ	Eigenschaften	Beschreibung
<code>merchantId</code>	String	Immer vorhanden.	ID des Händlers, der die Autorisierung erteilt hat.
<code>shopId</code>	String	Immer vorhanden.	ID des Shops, der die Autorisierung erteilt hat.
<code>creationTimestamp</code>	String	Immer vorhanden.	Zeitpunkt, zu dem die Autorisierung erstellt wurde. Formatierung entsprechend ISO-8601 durch den Standard Java DateTimeFormatter . Abweichend davon werden immer drei Nachkommastellen für die Millisekunden ausgegeben.
<code>approvalTimestamp</code>	String	Vorhanden, sobald die Autorisierung den Status <code>APPROVED</code> erreicht hat.	Zeitpunkt, zu dem die Autorisierung erteilt wurde. Formatierung wie <code>creationTimestamp</code> .
<code>revokeTimestamp</code>	String	Vorhanden, sobald die Autorisierung den Status <code>REVOKED</code> erreicht hat.	Zeitpunkt, zu dem die Autorisierung zurückgezogen wurde. Formatierung wie <code>creationTimestamp</code> .
<code>scopes</code>	Array	Immer vorhanden.	Menge der vom Shop erteilten Berechtigungen. Mögliche Werte sind: <code>PERFORM_CHECKOUT</code> : Zahlung auslösen. <code>CREATE_REFUND</code> : Rückzahlung auslösen.
<code>status</code>	String	Immer vorhanden.	Status der Autorisierung. Mögliche Werte sind: <code>APPROVED</code> : Bestätigt. <code>REVOKED</code> : Zurückgezogen. Dies kann durch den Händler, den oneKlick-Anbieter selbst oder einen paydirekt-Mitarbeiter geschehen sein.
<code>thirdPartyName</code>	String	Optional.	Name des oneKlick-Anbieters.
<code>clientName</code>	String	Optional.	Name des Clients.
<code>merchantName</code>	String	Optional.	Name des Händlers, der die Autorisierung erteilt hat.
<code>shopName</code>	String	Optional.	Name des Shops, der die Autorisierung erteilt hat.

Links

Relation	Description
self	Link zu dieser Autorisierungs-Ressource.

Return Codes

Status Code	Message Code	Beschreibung
200 (Ok)		Die Autorisierung wurde gefunden.
401 (Unauthorized)	ACCESS_TOKEN_EXPIRED	Siehe Resource Access .
404 (Not Found)	MERCHANT_AUTHORIZATION_NOT_FOUND	Die angefragte Autorisierung existiert nicht oder ist nicht zugreifbar.

Beispiel

```
GET /api/thirdpartymerchantauthorization/v1/authorizations/517cd636-63ce-43a9-954e-8cf84d475b13 HTTP/1.1
Content-Type: application/json;charset=utf-8
Authorization: Bearer <access_token>
```

```
HTTP/1.1 200 OK
Content-Type: application/hal+json;charset=utf-8
```

```
{
  "merchantAuthorizationId" : "517cd636-63ce-43a9-954e-8cf84d475b13",
  "merchantAuthorizationReference" : "secretReference",
  "thirdPartyId" : "a3dc71de-e82b-4442-a3c6-7d12332baa18",
  "clientId" : "43c5713e-2a49-452b-967f-e1b33d99874d",
  "merchantId" : "14b0fee1-a4b2-4e71-affd-f5dc3ad0d424",
  "shopId" : "525f825f-64a9-4dc3-b221-26faa212f0a2",
  "creationTimestamp" : "2022-09-26T12:23:30.914Z",
  "approvalTimestamp" : "2022-09-26T12:24:30.914Z",
  "scopes" : [ "PERFORM_CHECKOUT", "CREATE_REFUND" ],
  "status" : "APPROVED",
  "thirdPartyName" : "thirdPartyName",
  "clientName" : "clientName",
  "merchantName" : "Nicht verfügbar",
  "shopName" : "Nicht verfügbar",
  "_links" : {
    "self" : {
      "href" :
        "https://api.paydirekt.de/api/thirdpartymerchantauthorization/v1/authorizations/517cd636-63ce-43a9-954e-8cf84d475b13"
    }
  }
}
```

oneKlick-Checkout

Ein Checkout repräsentiert den gesamten Zahlungsvorgang. Es wird zwischen den Typen *Einmalzahlung* und (*gesicherte*) *Vorbestellung* unterschieden. In diesem Kapitel wird beschrieben, wie oneKlick-Checkouts angelegt werden. Diese werden ohne Käuferinteraktion durchgeführt. Durch die vorab erteilte Käuferautorisierung gelten sie sofort als vom Käufer bestätigt. Nach der Anlage stehen alle im [folgenden Kapitel](#) beschriebenen Funktionen auch für oneKlick-Checkouts zur Verfügung.

Für die Nutzung aller Endpunkte in diesem und dem [folgenden Kapitel](#) ist immer ein Access Token mit Händlerrechten notwendig (siehe [Access-Token](#)). Das heißt alle oneKlick-Anbieter außer Händler, die für eigene Shops tätig sind, müssen bei der Authentifizierung eine `merchantAuthorizationReference` angeben haben. Für die Durchführung von oneKlick-Checkouts muss darüber hinaus auch eine `customerAuthorizationReference` übergeben worden sein.

Anlage

```
POST /api/checkout/v1/checkouts
```

Neben dem Authentifizierungsendpoint ist dies der einzige Endpoint, der im System des Anbieters fest hinterlegt werden muss. Alle weiteren Ressourcen können über entsprechende Links angesprochen werden.



Die Features `overcapture` und `express` sind im Rahmen des giroipay oneKlick Verfahrens nicht zulässig. Diese Funktionen sind ausschließlich für normale Checkouts nutzbar.



Für Shops welche im giroipay-System als Point of Sale(POS)-Shop geführt werden gibt es bei der Anlage von Checkouts besondere Anforderungen/Pflichtfelder. Bitte hier die Angaben zum Feld `contextInformation` besonders beachten.

Request

Feld	Typ	Eigenschaften	Beschreibung
<code>type</code>	String	Pflichtfeld.	<p>Die Art der Bestellung: <code>DIRECT_SALE</code>, <code>ORDER</code> oder <code>ORDER_SECURED</code>.</p> <p>Im Falle eines <code>DIRECT_SALE</code> (Direktbestellung mit Einmalzahlung) wird automatisch eine Zahlungsautorisierung erzeugt. Der Händler erhält sofort eine Zahlungsgarantie.</p> <p>Zu einer <code>ORDER</code> (Vorbestellung oder Bestellung mit Teilzahlungen) können direkt im Anschluss oder später ein oder mehrere Captures angestoßen werden. Für eine Order wird keine Zahlungsgarantie an den Händler ausgesprochen. Die Garantie wird erst mit dem Capture gegeben.</p> <p>Zu einer <code>ORDER_SECURED</code> (gesicherte Vorbestellung oder Bestellung mit Teilzahlungen) können direkt im Anschluss oder später ein oder mehrere Captures angestoßen werden. Für eine gesicherte Vorbestellung wird eine Zahlungsgarantie für den gewählten Zeitraum (maximal 15 Kalendertage) an den Händler ausgesprochen. Captures (Teilzahlungen) werden innerhalb des Garantienzeitraums immer ausgeführt. Weitere Informationen zu Order und Capture finden Sie im Kapitel Capture.</p>
<code>oneClick</code>	Boolean	Pflichtfeld für oneClick-Checkouts.	<p>Flag, gibt an, dass es sich um einen oneClick-Checkout handelt. Muss hier <code>true</code> sein.</p>
<code>totalAmount</code>	Number	<p>Pflichtfeld.</p> <p>Zwischen 0.01 und 50000.</p> <p>Maximal zwei Nachkommastellen.</p>	<p>Der Gesamtbetrag der Bestellung, inkl. aller Lieferkosten. Es werden maximal 2 Nachkommastellen unterstützt.</p> <p>Im Falle eines <code>DIRECT_SALE</code> wird eine Zahlung für diesen Betrag initiiert.</p> <p>Im Falle einer <code>ORDER</code> können Captures bis maximal zu diesem Betrag angestoßen werden.</p>

Feld	Typ	Eigenschaften	Beschreibung
shippingAmount	Number	Optional. Mindestens 0.00. Maximal zwei Nachkommastelle n.	Die Versandkosten der Bestellung. Dieser Wert wird nicht für Berechnungen verwendet, sondern dient nur zur Information.
orderAmount	Number	Optional. Zwischen 0.01 und 50000. Maximal zwei Nachkommastelle n.	Der Warenwert der Bestellung, ohne Versandkosten. Dieser Wert wird nicht für Berechnungen verwendet, sondern dient nur zur Information.
refundLimit	Number	Optional. Zwischen 100 und 200. Wird bei Nichtangabe auf 200 gesetzt, das heißt maximal das Zweifache des Bestellwertes kann zurückgezahlt werden.	Maximal zurückzahlbarer Betrag als Prozentwert des Gesamtwertes der Bestellung.
currency	String	Pflichtfeld.	Die Währung des Gesamtbetrags. Derzeit wird nur EUR unterstützt.
items	Array von Items	Optional.	Die einzelnen Positionen des Warenkorbs. Es wird empfohlen, diese Werte zu übergeben. Dies verbessert die Erkennung von fraudulenten Vorgängen und hilft, Disputes zu vermeiden.

Feld	Typ	Eigenschaften	Beschreibung
shoppingCartType	String	Optional.	Kategorisiert den Warenkorb eines Checkouts anhand der Eigenschaften der enthaltenen Güter: Der Standardwert ist MIXED . PHYSICAL : Der Warenkorb enthält ausschließlich physische Güter. DIGITAL : Der Warenkorb enthält ausschließlich digitale Güter. MIXED : Der Warenkorb enthält sowohl physische als auch digitale Güter. ANONYMOUS_DONATION : Beim Warenkorb handelt es sich ausschließlich um eine anonyme Spende. AUTHORITIES_PAYMENT : Beim Warenkorb handelt es sich ausschließlich um Behördenzahlungen.
deliveryType	String	Optional.	Der Bestimmungsort einer Lieferung: STANDARD , PACKSTATION oder STORE_PICKUP . Der Standardwert ist STANDARD . STANDARD : Die Güter werden an eine gewöhnliche Postadresse geliefert. PACKSTATION : Die Güter werden an eine Packstation geliefert. STORE_PICKUP : Die Güter werden in der Filiale abgeholt. Dieses Feld enthält bei Express-Checkouts immer den Wert STANDARD und wird nicht anhand der gewählten Lieferoption aktualisiert.
shippingAddress	ShippingAddress	Optional für oneKlick-Checkouts.	Die Lieferanschrift des Empfängers.
merchantCustomerNumber	String	Optional. Maximal 50 Zeichen.	Händler-interne Kundennummer des Käufers. Wird dem Kunden in der Transaktionsübersicht angezeigt. Wird dem Händler in der Transaktionsübersicht angezeigt.

Feld	Typ	Eigenschaften	Beschreibung
<code>merchantOrderReferenceNumber</code>	String	Pflichtfeld. Maximal 20 Zeichen. Nur SEPA-konforme Zeichen .	Händler-interne, eindeutige Bestellnummer für diesen Kaufvorgang. Wird dem Kunden in der Transaktionsübersicht angezeigt. Wird dem Händler in der Transaktionsübersicht angezeigt. Die Bestellnummer wird in der Händler-Lastschrift als <code>instructionId</code> und im Verwendungszweck bei Händler und Käufer verwendet.
<code>merchantInvoiceReferenceNumber</code>	String	Optional. Maximal 100 Zeichen.	Händler-interne, eindeutige Rechnungsnummer für diesen Kauf bzw. Zahlvorgang. Wird dem Händler in der Transaktionsübersicht angezeigt.
<code>merchantReconciliationReferenceNumber</code>	String	Optional. Maximal 30 Zeichen.	Reconciliation-Nummer. Diese wird bei Direct-Sales in den Verwendungszweck der Händlerzahlung eingefügt. Bei Vorbestellungen wird stattdessen die bei Anlage der Captures gesetzte <code>merchantReconciliationReferenceNumber</code> verwendet.
<code>sha256hashedEmailAddress</code>	String	Optional. Maximal 64 Zeichen.	Die E-Mail Adresse des Käufers als Base-64 encodierter SHA-256 Hash-Wert ohne Padding, sofern vorhanden. Pseudo-Code: <code>sha256hashedEmailAddress = toStringUTF8(base64(sha256(toBytesUTF8(emailAddress))))</code> . Beispiel: Aus der E-Mail-Adresse <code>max@muster.de</code> muss sich exakt der folgende Hashwert ergeben (ohne Anführungszeichen): "6JL4VUgVxkq2m+a9I6ScfW2ofJP5y6wsvSaHIsX+iLs=". Diese Information wird zur Fraud Prevention verwendet.
<code>callbackUrlStatusUpdates</code>	String	Optional. Maximal 2000 Zeichen.	URL des Endpoints, der Mitteilungen zu Statusänderungen des Checkouts empfangen soll.

Feld	Typ	Eigenschaften	Beschreibung
<code>minimumAge</code>	Number	Optional.	Das Mindestalter (in Jahren), das der Käufer erreicht haben muss, um die Bestellung ausführen zu dürfen, beispielsweise, weil sie Artikel enthält, die einer Altersbeschränkung unterliegen (Filme, Computerspiele, ...). Die Prüfung erfolgt direkt nach dem Login des Käufers in das giropay-System. Bei erfolgreicher Verifikation wird der Ablauf ohne weitere Meldung wie gewohnt fortgesetzt. Bei nicht-erfolgreicher Verifikation, d. h. der Kunde hat das erforderliche Mindestalter noch nicht erreicht, erfolgt eine Umleitung auf den Link <code>redirectUrlAfterAgeVerificationFailure</code> (siehe unten). Bei Nicht-Angabe dieses Wertes erfolgt keine Altersverifikation.
<code>note</code>	String	Optional. Maximal 37 Zeichen.	Freitextfeld, das auf dem Kontoauszug im Feld Verwendungszweck erscheint (nur <code>DIRECT_SALE</code>).
<code>deliveryInformation</code>	Object	Optional.	Enthält Informationen über die Lieferung.
<code>deliveryInformation.expectedShippingDate</code>	String	Optional. Zeitstempel im <code>ISO-8601</code> -Format.	Erwartetes Versanddatum.
<code>deliveryInformation.logisticsProvider</code>	String	Optional.	Paket-Dienstleister.
<code>deliveryInformation.trackingNumber</code>	String	Optional.	Sendungsnummer.
<code>requestedPreauthorizationValidity</code>	String	Optional. Datum im Format <code>yyyy-mm-dd</code> . Maximal 15 Kalendertage in der Zukunft (entspricht dem Standardwert, sofern nicht befüllt).	Gewünschter Garantiezeitraum (maximal 15 Kalendertage) für eine gesicherte Vorbestellung (<code>ORDER_SECURED</code>).
<code>contextInformation</code>	Object		Informationen zum Point of Sale und zum Endgerät des Käufers.

Feld	Typ	Eigenschaften	Beschreibung
<code>contextInformation.deviceId</code>	String	Entweder der Device-Fingerprint oder die Device-ID muss für oneKlick-Checkouts angegeben werden. Maximal 50 Zeichen.	Eindeutige ID zur Identifikation des Endgeräts des Käufers.
<code>contextInformation.deviceFingerprint</code>	String	Entweder der Device-Fingerprint oder die Device-ID muss für oneKlick-Checkouts angegeben werden. Maximal 50 Zeichen.	Fingerprint des Endgeräts des Käufers.
<code>contextInformation.deviceIpAddress</code>	String	Pflichtfeld für oneKlick-Checkouts von 'Point of Sales' Shops. Maximal 40 Zeichen.	IP-Adresse des Endgeräts des Käufers zum Zeitpunkt der Checkoutanlage.
<code>contextInformation.terminalId</code>	String	Pflichtfeld für oneKlick-Checkouts von 'Point of Sales' Shops. Maximal 50 Zeichen.	Eindeutige ID zur Identifikation des Point of Sales, an dem dem Käufer die bezahlte Ware oder Dienstleistung zur Verfügung gestellt wird. Z.B. ein Snackautomat oder Zapfsäule.
<code>contextInformation.terminalAddress</code>	Object	Pflichtfeld für oneKlick-Checkouts von 'Point of Sales' Shops.	Adresse des Point of Sales. Inhaltliche Vorgaben entsprechen der ShippingAddress , wobei Vor- und Nachname durch einen <code>name</code> ersetzt sind und es kein Feld "Emailadresse" gibt.

Response

In der Response wird der angelegte Checkout zurückgegeben.

Im HTTP Header-Feld `Location` ist die URL der angelegten Checkout-Ressource, inkl. Checkout-ID enthalten. Die Checkout-ID wird im Response Body ebenfalls zurückgegeben.

Feld	Typ	Eigenschaften	Beschreibung
<code>checkoutId</code>	String	Immer vorhanden.	Eindeutige UUID aus 36 Zeichen für diesen Checkout-Vorgang. Wird vom giropay-System vergeben.
<code>type</code>	String	Immer vorhanden.	<p>Die Art der Bestellung: <code>DIRECT_SALE</code>, <code>ORDER</code> oder <code>ORDER_SECURED</code>.</p> <p>Im Falle eines <code>DIRECT_SALE</code> (Direktbestellung mit Einmalzahlung) wird automatisch eine Zahlungsautorisierung erzeugt. Der Händler erhält sofort eine Zahlungsgarantie.</p> <p>Zu einer <code>ORDER</code> (Vorbestellung oder Bestellung mit Teilzahlungen) können direkt im Anschluss oder später ein oder mehrere Captures angestoßen werden. Für eine Order wird keine Zahlungsgarantie an den Händler ausgesprochen. Die Garantie wird erst mit dem Capture gegeben.</p> <p>Zu einer <code>ORDER_SECURED</code> (gesicherte Vorbestellung oder Bestellung mit Teilzahlungen) können direkt im Anschluss oder später ein oder mehrere Captures angestoßen werden. Für eine gesicherte Vorbestellung wird eine Zahlungsgarantie für den gewählten Zeitraum (maximal 15 Kalendertage) an den Händler ausgesprochen. Captures (Teilzahlungen) werden innerhalb des Garantienzeitraums immer ausgeführt. Weitere Informationen zu Order und Capture finden Sie im Kapitel Capture.</p>
<code>oneClick</code>	Boolean	Immer vorhanden bei oneClick-Checkouts.	Flag, gibt an, dass es sich um einen oneClick-Checkout handelt. Muss hier <code>true</code> sein.

Feld	Typ	Eigenschaften	Beschreibung
status	String	Immer vorhanden.	<p>Status des Checkouts: OPEN, PENDING, APPROVED, REJECTED, CANCELED, CLOSED oder EXPIRED.</p> <p>OPEN: Der Checkout wurde neu angelegt und vom Benutzer noch nicht bestätigt.</p> <p>PENDING: Der Checkout wurde vom Kunden bestätigt, aber noch nicht per <i>execute</i> aktiviert (derzeit nur <i>giropay Express</i>)</p> <p>APPROVED: Der Checkout wurde vom Kunden bestätigt und vom System genehmigt. Im Falle einer ORDER können jetzt Captures angelegt werden.</p> <p>REJECTED: Der Checkout wurde vom System abgelehnt. Die Zahlung war nicht erfolgreich. Es ist nicht zu erwarten, dass die Zahlung bei erneutem Versuch mit diesem Benutzer erfolgreich sein wird.</p> <p>CANCELED: Der Checkout wurde vom Kunden abgebrochen oder es kam zu einem technischen Fehler. Der Checkout kann nicht weiter bearbeitet werden.</p> <p>CLOSED: Nur ORDER: Die Bestellung ist geschlossen, es können keine weiteren Captures durchgeführt werden. Refunds sind weiter möglich.</p> <p>EXPIRED: Der Checkout ist abgelaufen und kann nicht weiter verwendet werden. Der Benutzer hat den Checkout nicht innerhalb von 30 Minuten bestätigt.</p>
correlationId	String	Vorhanden, sobald sich ein Kunde eingeloggt hat. UUID aus 36 Zeichen.	Die eindeutige Identifikation des Checkouts und aller dazugehöriger Transaktionen.

Feld	Typ	Eigenschaften	Beschreibung
<code>creationTimestamp</code>	String	Immer vorhanden.	Der Erzeugungszeitpunkt dieses Checkouts. Wird vom giropay-System vergeben. Formatierung entsprechend ISO-8601 durch den Standard Java DateTimeFormatter . Abweichend davon werden immer drei Nachkommastellen für die Millisekunden ausgegeben.
<code>expiryTimestamp</code>	String	Immer vorhanden.	Der Ablaufzeitpunkt dieses Checkouts. Kann über das Feld <code>expiryTime</code> bei Anlage angepasst werden, ansonsten wird vom giropay-System als Standardwert 30 Minuten ab Erzeugung vergeben. Formatierung entsprechend ISO-8601 durch den Standard Java DateTimeFormatter . Abweichend davon werden immer drei Nachkommastellen für die Millisekunden ausgegeben.
<code>totalAmount</code>	Number	Immer vorhanden.	Der Gesamtbetrag der Bestellung, inkl. aller Lieferkosten. Es werden maximal 2 Nachkommastellen unterstützt. Im Falle eines <code>DIRECT_SALE</code> wird eine Zahlung für diesen Betrag initiiert. Im Falle einer <code>ORDER</code> können Captures bis maximal zu diesem Betrag angestoßen werden.
<code>shippingAmount</code>	Number	Vorhanden, falls bei Anlage übergeben.	Die Versandkosten der Bestellung. Dieser Wert wird nicht für Berechnungen verwendet, sondern dient nur zur Information.
<code>orderAmount</code>	Number	Vorhanden, falls bei Anlage übergeben.	Der Warenwert der Bestellung, ohne Versandkosten. Dieser Wert wird nicht für Berechnungen verwendet, sondern dient nur zur Information.
<code>refundLimit</code>	Number	Vorhanden, falls bei Anlage übergeben.	Maximal zurückzahlbarer Betrag als Prozentwert des Gesamtwertes der Bestellung.
<code>currency</code>	String	Immer vorhanden.	Die Währung des Gesamtbetrags. Derzeit wird nur EUR unterstützt.

Feld	Typ	Eigenschaften	Beschreibung
<code>items</code>	Array von <code>Items</code>	Vorhanden, falls bei Anlage übergeben.	Die einzelnen Positionen des Warenkorbs. Es wird empfohlen, diese Werte zu übergeben. Dies verbessert die Erkennung von fraudulenter Vorgängen und hilft, Disputes zu vermeiden.
<code>shoppingCartType</code>	String	Vorhanden, falls bei Anlage übergeben.	Kategorisiert den Warenkorb eines Checkouts anhand der Eigenschaften der enthaltenen Güter: Der Standardwert ist <code>MIXED</code> . <code>PHYSICAL</code> : Der Warenkorb enthält ausschließlich physische Güter. <code>DIGITAL</code> : Der Warenkorb enthält ausschließlich digitale Güter. <code>MIXED</code> : Der Warenkorb enthält sowohl physische als auch digitale Güter. <code>ANONYMOUS_DONATION</code> : Beim Warenkorb handelt es sich ausschließlich um eine anonyme Spende. <code>AUTHORITIES_PAYMENT</code> : Beim Warenkorb handelt es sich ausschließlich um Behördenzahlungen.
<code>deliveryType</code>	String	Vorhanden, falls bei Anlage übergeben.	Der Bestimmungsort einer Lieferung: <code>STANDARD</code> , <code>PACKSTATION</code> oder <code>STORE_PICKUP</code> . Der Standardwert ist <code>STANDARD</code> . <code>STANDARD</code> : Die Güter werden an eine gewöhnliche Postadresse geliefert. <code>PACKSTATION</code> : Die Güter werden an eine Packstation geliefert. <code>STORE_PICKUP</code> : Die Güter werden in der Filiale abgeholt. Dieses Feld enthält bei Express-Checkouts immer den Wert <code>STANDARD</code> und wird nicht anhand der gewählten Lieferoption aktualisiert.
<code>shippingAddress</code>	<code>ShippingAddress</code>	Vorhanden, falls bei Anlage übergeben.	Die <code>Lieferanschrift</code> des Empfängers.

Feld	Typ	Eigenschaften	Beschreibung
<code>merchantCustomerNumber</code>	String	Vorhanden, falls bei Anlage übergeben.	Händler-interne Kundennummer des Käufers. Wird dem Kunden in der Transaktionsübersicht angezeigt. Wird dem Händler in der Transaktionsübersicht angezeigt.
<code>merchantOrderReferenceNumber</code>	String	Immer vorhanden.	Händler-interne, eindeutige Bestellnummer für diesen Kaufvorgang. Wird dem Kunden in der Transaktionsübersicht angezeigt. Wird dem Händler in der Transaktionsübersicht angezeigt. Die Bestellnummer wird in der Händler-Lastschrift als <code>instructionId</code> und im Verwendungszweck bei Händler und Käufer verwendet.
<code>merchantInvoiceReferenceNumber</code>	String	Vorhanden, falls bei Anlage übergeben.	Händler-interne, eindeutige Rechnungsnummer für diesen Kauf- bzw. Zahlvorgang. Wird dem Händler in der Transaktionsübersicht angezeigt.
<code>merchantReconciliationReferenceNumber</code>	String	Vorhanden, falls bei Anlage übergeben.	Reconciliation-Nummer. Diese wird bei Direct-Sales in den Verwendungszweck der Händlerzahlung eingefügt. Bei Vorbestellungen wird stattdessen die bei Anlage der Captures gesetzte <code>merchantReconciliationReferenceNumber</code> verwendet.
<code>callbackUrlStatusUpdates</code>	String	Vorhanden, falls bei Anlage übergeben.	URL des Endpoints, der Mitteilungen zu Statusänderungen des Checkouts empfangen soll.

Feld	Typ	Eigenschaften	Beschreibung
<code>minimumAge</code>	Number	Vorhanden, falls bei Anlage übergeben.	Das Mindestalter (in Jahren), das der Käufer erreicht haben muss, um die Bestellung ausführen zu dürfen, beispielsweise, weil sie Artikel enthält, die einer Altersbeschränkung unterliegen (Filme, Computerspiele, ...). Die Prüfung erfolgt direkt nach dem Login des Käufers in das giropay-System. Bei erfolgreicher Verifikation wird der Ablauf ohne weitere Meldung wie gewohnt fortgesetzt. Bei nicht-erfolgreicher Verifikation, d. h. der Kunde hat das erforderliche Mindestalter noch nicht erreicht, erfolgt eine Umleitung auf den Link <code>redirectUrlAfterAgeVerificationFailure</code> (siehe unten). Bei Nicht-Angabe dieses Wertes erfolgt keine Altersverifikation.
<code>note</code>	String	Vorhanden, falls bei Anlage übergeben.	Freitextfeld, das auf dem Kontoauszug im Feld Verwendungszweck erscheint (nur <code>DIRECT_SALE</code>).
<code>deliveryInformation</code>	Object	Optional.	Enthält Informationen über die Lieferung.
<code>deliveryInformation.expectedShippingDate</code>	String	Optional.	Erwartetes Versanddatum.
<code>deliveryInformation.logisticsProvider</code>	String	Optional.	Paket-Dienstleister.
<code>deliveryInformation.trackingNumber</code>	String	Optional.	Sendungsnummer.
<code>preauthorizationValidity</code>	String	Optional.	Garantiezeitraum für die gesicherte Vorbestellung (<code>ORDER_SECURED</code>). Captures innerhalb dieses Zeitraums sind für den Händler garantiert.
<code>contextInformation</code>	Object	Immer vorhanden bei oneKlick-Checkouts.	Informationen zum Point of Sale und zum Endgerät des Käufers.

Feld	Typ	Eigenschaften	Beschreibung
<code>contextInformation.deviceId</code>	String	Mindestens eines der Felder Fingerprint und Device-ID ist bei oneKlick-Checkouts vorhanden.	Eindeutige ID zur Identifikation des Endgeräts des Käufers.
<code>contextInformation.deviceFingerprint</code>	String	Mindestens eines der Felder Fingerprint und Device-ID ist bei oneKlick-Checkouts vorhanden.	Fingerprint des Endgeräts des Käufers.
<code>contextInformation.deviceIpAddress</code>	String	Immer vorhanden bei oneKlick-Checkouts von 'Point of Sales' Shops.	IP-Adresse des Endgeräts des Käufers zum Zeitpunkt der Checkoutanlage.
<code>contextInformation.terminalId</code>	String	Immer vorhanden bei oneKlick-Checkouts von 'Point of Sales' Shops.	Eindeutige ID zur Identifikation des Point of Sales, an dem dem Käufer die bezahlte Ware oder Dienstleistung zur Verfügung gestellt wird. Z.B. ein Snackautomat oder Zapfsäule.
<code>contextInformation.terminalAddress</code>	Object	Immer vorhanden bei oneKlick-Checkouts von 'Point of Sales' Shops.	Adresse des Point of Sales. Inhaltliche Vorgaben entsprechen der ShippingAddress , wobei Vor- und Nachname durch einen <code>name</code> ersetzt sind und es kein Feld "Emailadresse" gibt.
<code>thirdPartyId</code>	String	Optional. UUID aus 36 Zeichen.	ID des oneKlick-Anbieters, der diesen Checkout technisch angelegt hat.
<code>thirdPartyClientId</code>	String	Optional. UUID aus 36 Zeichen.	ID der oneKlick-Anwendung.
<code>thirdPartyClientName</code>	String	Optional. UUID aus 36 Zeichen.	Name der oneKlick-Anwendung.
<code>thirdPartyCustomerAuthorizationId</code>	String	Optional. UUID aus 36 Zeichen.	ID der Kundenautorisierung.

Feld	Typ	Eigenschaften	Beschreibung
<code>thirdPartyMerchantAuthorizationId</code>	String	Optional. UUID aus 36 Zeichen.	ID der Händlerautorisierung.
<code>_links</code>	Object		Links zu Ressourcen und verfügbaren Aktionen des Checkouts.
<code>_embedded</code>	Optional Captures und/oder Refunds, falls vorhanden		Embedded Ressourcen

Links

Relation	Description
<code>self</code>	Link zu dieser Ressource.
<code>close</code>	Link zum Endpoint, über den die Order geschlossen werden kann. Dieser Link ist nur vorhanden, wenn es sich um einen Checkout vom Typ <code>order</code> handelt und er im Status <code>APPROVED</code> ist.
<code>captures</code>	Link zu <code>Captures</code> . Dieser Link ist nur vorhanden, wenn für diese Bestellung ein <code>Capture</code> möglich ist.
<code>refunds</code>	Link zu <code>Refunds</code> . Dieser Link ist nur vorhanden, wenn für diese Bestellung eine Rückzahlung möglich ist, d. h. wenn mindestens ein <code>Capture</code> im Status <code>SUCCESSFUL</code> vorhanden ist. Wenn der Link nicht vorhanden ist, sind keine Rückzahlungen möglich.
<code>updateDeliveryInformation</code>	Link zum Update der Lieferinformation. Dieser Link ist nur vorhanden, wenn die Update Frist für diesen Checkout noch nicht überschritten ist.
<code>updateMerchantInvoiceReferenceNumber</code>	Link zum Update der Rechnungsreferenznummer. Dieser Link ist nur vorhanden, wenn die Update Frist für diesen Checkout noch nicht überschritten ist.

Return Codes

Status Code	Message Code	Beschreibung
201 (Created)		Der OneKlick Checkout wurde angelegt und in der Response zurückgeliefert. Falls er im Status APPROVED ist, wird im Falle eines DirectSales nun automatisch die Zahlung ausgelöst; bei einer Order können nun Captures abgerufen werden.
400 (Bad Request)	VALIDATION_ERROR	Der Request ist syntaktisch ungültig.
400 (Bad Request)	CONVERSION_ERROR	Der Request enthält ungültige Zeichen. Häufigste Ursache hierfür ist, dass der Request nicht korrekt UTF-8 codiert ist. Falls die ungültigen Zeichen einem Attribut im Request zugeordnet werden können, wird dieses im Feld path genannt und der ungültige Wert im Feld content übermittelt.
401 (Unauthorized)	ACCESS_TOKEN_EXPIRED	Siehe Resource Access .
401 (Unauthorized)	UNAUTHORIZED	Der Zugriff wurde verweigert. Bitte wenden Sie sich an den giropay-Support.
403 (Forbidden)	FORBIDDEN	Sie haben keine ausreichenden Berechtigungen zur Anlage von oneKlick-Checkouts. Möglicherweise fehlt ein entsprechender Scope in der referenzierten Käufer- oder Händlerautorisierung.
422 (Unprocessable Entity)	MERCHANT_BANKACCOUNT_LOCKED	Das Bankkonto des Händlers wurde gesperrt und es können keine neue Checkouts oder Captures angelegt werden. Der Händler sollte sich an den giropay-Support wenden.
422 (Unprocessable Entity)	CHECKOUT_REJECTED	Der Checkout wurde vom System abgelehnt.
422 (Unprocessable Entity)	BUYER_NOT_ACTIVE	Der giropay-Account des Käufers ist nicht mehr oder noch nicht aktiv.
422 (Unprocessable Entity)	MERCHANT_NOT_ACTIVE	Das Händler-Profil, zu dessen Gunsten der Checkout ausgeführt werden soll, ist nicht mehr oder noch nicht aktiv.

Beispiel (Direct Sale, inkl. Altersverifikation)

```
POST /api/checkout/v1/checkouts HTTP/1.1
Content-Type: application/json;charset=utf-8
Authorization: Bearer <access_token>
```

```
{
  "type" : "DIRECT_SALE",
  "oneClick" : true,
```

```
"contextInformation" : {
  "deviceId" : "8adb80f93",
  "deviceFingerprint" : "ae8faa7f67709ebab82f579945c7e6b5",
  "deviceIpAddress" : "123.123.123.123",
  "terminalId" : "b9f1c1b7",
  "terminalAddress" : {
    "name" : "Pumpe 3",
    "company" : "Mustermann Tankstellen GmbH",
    "street" : "Hamburger Allee",
    "streetNr" : "123A",
    "zip" : "60626",
    "city" : "Frankfurt a.M.",
    "countryCode" : "DE",
    "state" : "Hessen"
  }
},
"totalAmount" : 100.0,
"shippingAmount" : 3.5,
"orderAmount" : 96.5,
"refundLimit" : 200,
"items" : [ {
  "quantity" : 3,
  "name" : "Bobbycar",
  "ean" : "800001303",
  "price" : 25.99
}, {
  "quantity" : 1,
  "name" : "Helm",
  "price" : 18.53
} ],
"shoppingCartType" : "MIXED",
"currency" : "EUR",
"shippingAddress" : {
  "addresseeGivenName" : "Marie",
  "addresseeLastName" : "Mustermann",
  "company" : "Musterbau GmbH & Co KG",
  "street" : "Kastanienallee",
  "streetNr" : "999",
  "additionalAddressInformation" : "Haus 4",
  "zip" : "90402",
  "city" : "Schwaig",
  "countryCode" : "DE",
  "state" : "Bayern"
},
"deliveryInformation" : {
  "expectedShippingDate" : "2016-10-19T12:00:00.000Z",
  "logisticsProvider" : "DHL",
  "trackingNumber" : "1234567890"
},
"merchantCustomerNumber" : "cust-732477",
"merchantOrderReferenceNumber" : "order-A12223412",
```

```
"merchantReconciliationReferenceNumber" : "recon-A12223412",
"merchantInvoiceReferenceNumber" : "20150112334345",
"note" : "Ihr Einkauf bei Spielauto-Versand.",
"sha256hashedEmailAddress" : "fxP4R-IxH1Eaxpb0f_i5Shc8-FrYrtmx5lx35f9Xzgg",
"minimumAge" : 18,
"callbackUrlStatusUpdates" : "https://spielauto-versand.de/callback/status"
}
```

HTTP/1.1 201 Created

Location: <https://api.paydirekt.de/api/checkout/v1/checkouts/7c9c5f49-68aa-470c-804c-70da9e28eb75>

Content-Type: application/hal+json;charset=utf-8

```
{
  "checkoutId" : "7c9c5f49-68aa-470c-804c-70da9e28eb75",
  "type" : "DIRECT_SALE",
  "oneClick" : true,
  "status" : "APPROVED",
  "correlationId" : "ext0815-00000261",
  "creationTimestamp" : "2022-10-25T12:35:18.008Z",
  "totalAmount" : 100.0,
  "shippingAmount" : 3.5,
  "orderAmount" : 96.5,
  "refundLimit" : 200,
  "currency" : "EUR",
  "items" : [ {
    "quantity" : 3,
    "name" : "Bobbycar",
    "ean" : "800001303",
    "price" : 25.99
  }, {
    "quantity" : 1,
    "name" : "Helm",
    "price" : 18.53
  } ],
  "shoppingCartType" : "MIXED",
  "shippingAddress" : {
    "addresseeGivenName" : "Marie",
    "addresseeLastName" : "Mustermann",
    "company" : "Musterbau GmbH & Co KG",
    "street" : "Kastanienallee",
    "streetNr" : "999",
    "additionalAddressInformation" : "Haus 4",
    "zip" : "90402",
    "city" : "Schwaig",
    "countryCode" : "DE",
    "state" : "Bayern"
  },
  "merchantOrderReferenceNumber" : "order-A12223412",
  "merchantCustomerNumber" : "cust-732477",
}
```

```
"merchantInvoiceReferenceNumber" : "20150112334345",
"merchantReconciliationReferenceNumber" : "recon-A12223412",
"note" : "Ihr Einkauf bei Spielauto-Versand.",
"minimumAge" : 18,
"callbackUrlStatusUpdates" : "https://spielauto-versand.de/callback/status",
"contextInformation" : {
  "deviceId" : "8adb80f93",
  "deviceFingerprint" : "ae8faa7f67709ebab82f579945c7e6b5",
  "deviceIpAddress" : "123.123.123.123",
  "terminalId" : "b9f1c1b7",
  "terminalAddress" : {
    "name" : "Pumpe 3",
    "company" : "Mustermann Tankstellen GmbH",
    "street" : "Hamburger Allee",
    "streetNr" : "123A",
    "zip" : "60626",
    "city" : "Frankfurt a.M.",
    "countryCode" : "DE",
    "state" : "Hessen"
  }
},
"thirdPartyId" : "4d054c51-c90c-4338-8063-8c0a2e74ccef",
"thirdPartyClientId" : "3d60b3bd-e846-4f72-b90d-1d8bfe19afa4",
"thirdPartyClientName" : "14696318-3c6f-40f0-9117-9f4358b7d643",
"thirdPartyCustomerAuthorizationId" : "222c039c-8329-4830-9585-4e5a9128f723",
"thirdPartyMerchantAuthorizationId" : "a6cb7e14-c522-45c9-94e3-46fd2d666453",
"deliveryInformation" : {
  "expectedShippingDate" : "2016-10-19T12:00:00.000Z",
  "logisticsProvider" : "DHL",
  "trackingNumber" : "1234567890"
},
"expiryTimestamp" : "2022-10-25T13:05:18.008Z",
"_links" : {
  "refunds" : {
    "href" : "https://api.paydirekt.de/api/checkout/v1/checkouts/7c9c5f49-68aa-470c-804c-70da9e28eb75/refunds"
  },
  "updateDeliveryInformation" : {
    "href" : "https://api.paydirekt.de/api/checkout/v1/checkouts/7c9c5f49-68aa-470c-804c-70da9e28eb75/deliveryInformation"
  },
  "updateMerchantInvoiceReferenceNumber" : {
    "href" : "https://api.paydirekt.de/api/checkout/v1/checkouts/7c9c5f49-68aa-470c-804c-70da9e28eb75/merchantInvoiceReferenceNumber"
  },
  "self" : {
    "href" : "https://api.paydirekt.de/api/checkout/v1/checkouts/7c9c5f49-68aa-470c-804c-70da9e28eb75"
  }
},
"_embedded" : {
```

```

"captures" : [ {
  "type" : "CAPTURE_DIRECT_SALE",
  "transactionId" : "8731cef8-ac1a-46e4-8713-d0cb48ef0710",
  "amount" : 100.0,
  "finalCapture" : true,
  "callbackUrlStatusUpdates" : "https://spielauto-versand.de/callback/status",
  "deliveryInformation" : {
    "expectedShippingDate" : "2016-10-19T12:00:00.000Z",
    "logisticsProvider" : "DHL",
    "trackingNumber" : "1234567890"
  },
  "status" : "SUCCESSFUL",
  "_links" : {
    "self" : {
      "href" : "https://api.paydirekt.de/api/checkout/v1/checkouts/7c9c5f49-68aa-470c-804c-70da9e28eb75/captures/8731cef8-ac1a-46e4-8713-d0cb48ef0710"
    }
  }
} ]
}
}

```

Beispiel (Order)

```

POST /api/checkout/v1/checkouts HTTP/1.1
Content-Type: application/json;charset=utf-8
Authorization: Bearer <access_token>

```

```

{
  "type" : "ORDER",
  "oneClick" : true,
  "contextInformation" : {
    "deviceId" : "8adb80f93",
    "deviceFingerprint" : "ae8faa7f67709ebab82f579945c7e6b5",
    "deviceIpAddress" : "123.123.123.123",
    "terminalId" : "b9f1c1b7",
    "terminalAddress" : {
      "name" : "Pumpe 3",
      "company" : "Mustermann Tankstellen GmbH",
      "street" : "Hamburger Allee",
      "streetNr" : "123A",
      "zip" : "60626",
      "city" : "Frankfurt a.M.",
      "countryCode" : "DE",
      "state" : "Hessen"
    }
  },
  "totalAmount" : 100.0,
  "shippingAmount" : 3.5,
  "orderAmount" : 96.5,

```

```
"refundLimit" : 200,
"items" : [ {
  "quantity" : 3,
  "name" : "Bobbycar",
  "ean" : "800001303",
  "price" : 25.99
}, {
  "quantity" : 1,
  "name" : "Helm",
  "price" : 18.53
} ],
"shoppingCartType" : "MIXED",
"currency" : "EUR",
"deliveryInformation" : {
  "expectedShippingDate" : "2016-10-19T12:00:00.000Z",
  "logisticsProvider" : "DHL",
  "trackingNumber" : "1234567890"
},
"merchantCustomerNumber" : "cust-732477",
"merchantOrderReferenceNumber" : "order-A12223412",
"merchantReconciliationReferenceNumber" : "recon-A12223412",
"merchantInvoiceReferenceNumber" : "20150112334345",
"note" : "Ihr Einkauf bei Spielauto-Versand.",
"sha256hashedEmailAddress" : "fxP4R-IxH1Eaxpb0f_i5Shc8-FrYrtmx5lx35f9Xzgg",
"minimumAge" : 18,
"callbackUrlStatusUpdates" : "https://spielauto-versand.de/callback/status"
}
```

HTTP/1.1 201 Created

Location: <https://api.paydirekt.de/api/checkout/v1/checkouts/a81bd75d-f412-4306-84f5-7fdc0d31e399>

Content-Type: application/hal+json;charset=utf-8

```
{
  "checkoutId" : "a81bd75d-f412-4306-84f5-7fdc0d31e399",
  "type" : "ORDER",
  "oneClick" : true,
  "status" : "APPROVED",
  "correlationId" : "ext0815-00000254",
  "creationTimestamp" : "2022-10-25T12:35:12.613Z",
  "totalAmount" : 100.0,
  "shippingAmount" : 3.5,
  "orderAmount" : 96.5,
  "refundLimit" : 200,
  "currency" : "EUR",
  "items" : [ {
    "quantity" : 3,
    "name" : "Bobbycar",
    "ean" : "800001303",
    "price" : 25.99
  }
```



```

}, {
  "quantity" : 1,
  "name" : "Helm",
  "price" : 18.53
} ],
"shoppingCartType" : "MIXED",
"merchantOrderReferenceNumber" : "order-A12223412",
"merchantCustomerNumber" : "cust-732477",
"merchantInvoiceReferenceNumber" : "20150112334345",
"merchantReconciliationReferenceNumber" : "recon-A12223412",
"note" : "Ihr Einkauf bei Spielauto-Versand.",
"minimumAge" : 18,
"callbackUrlStatusUpdates" : "https://spielauto-versand.de/callback/status",
"contextInformation" : {
  "deviceId" : "8adb80f93",
  "deviceFingerprint" : "ae8faa7f67709ebab82f579945c7e6b5",
  "deviceIpAddress" : "123.123.123.123",
  "terminalId" : "b9f1c1b7",
  "terminalAddress" : {
    "name" : "Pumpe 3",
    "company" : "Mustermann Tankstellen GmbH",
    "street" : "Hamburger Allee",
    "streetNr" : "123A",
    "zip" : "60626",
    "city" : "Frankfurt a.M.",
    "countryCode" : "DE",
    "state" : "Hessen"
  }
},
"thirdPartyId" : "6979cddf-b05e-43da-b809-0522bea7202c",
"thirdPartyClientId" : "783bf429-aab3-429b-be94-145433322131",
"thirdPartyClientName" : "ec706559-b428-4988-aa3c-63288c46eba9",
"thirdPartyCustomerAuthorizationId" : "8688f9ff-2a12-44d8-b6b0-8a0ffa91e1cf",
"thirdPartyMerchantAuthorizationId" : "49af39a7-aff0-4f22-9b11-f724ac5925af",
"deliveryInformation" : {
  "expectedShippingDate" : "2016-10-19T12:00:00.000Z",
  "logisticsProvider" : "DHL",
  "trackingNumber" : "1234567890"
},
"expiryTimestamp" : "2022-10-25T13:05:12.613Z",
"_links" : {
  "captures" : {
    "href" : "https://api.paydirekt.de/api/checkout/v1/checkouts/a81bd75d-f412-4306-84f5-7fdc0d31e399/captures"
  },
  "close" : {
    "href" : "https://api.paydirekt.de/api/checkout/v1/checkouts/a81bd75d-f412-4306-84f5-7fdc0d31e399/close"
  },
  "updateDeliveryInformation" : {
    "href" : "https://api.paydirekt.de/api/checkout/v1/checkouts/a81bd75d-f412-4306-

```

```

84f5-7fdc0d31e399/deliveryInformation"
  },
  "updateMerchantInvoiceReferenceNumber" : {
    "href" : "https://api.paydirekt.de/api/checkout/v1/checkouts/a81bd75d-f412-4306-
84f5-7fdc0d31e399/merchantInvoiceReferenceNumber"
  },
  "self" : {
    "href" : "https://api.paydirekt.de/api/checkout/v1/checkouts/a81bd75d-f412-4306-
84f5-7fdc0d31e399"
  }
}
}
}

```

Beispiel (Order secured)

```

POST /api/checkout/v1/checkouts HTTP/1.1
Content-Type: application/json;charset=utf-8
Authorization: Bearer <access_token>

```

```

{
  "type" : "ORDER_SECURED",
  "oneClick" : true,
  "contextInformation" : {
    "deviceId" : "8adb80f93",
    "deviceFingerprint" : "ae8faa7f67709ebab82f579945c7e6b5",
    "deviceIpAddress" : "123.123.123.123",
    "terminalId" : "b9f1c1b7",
    "terminalAddress" : {
      "name" : "Pumpe 3",
      "company" : "Mustermann Tankstellen GmbH",
      "street" : "Hamburger Allee",
      "streetNr" : "123A",
      "zip" : "60626",
      "city" : "Frankfurt a.M.",
      "countryCode" : "DE",
      "state" : "Hessen"
    }
  },
  "totalAmount" : 100.0,
  "shippingAmount" : 3.5,
  "orderAmount" : 96.5,
  "refundLimit" : 200,
  "items" : [ {
    "quantity" : 3,
    "name" : "Bobbycar",
    "ean" : "800001303",
    "price" : 25.99
  }, {
    "quantity" : 1,
    "name" : "Helm",

```

```
"price" : 18.53
} ],
"shoppingCartType" : "MIXED",
"currency" : "EUR",
"deliveryInformation" : {
  "expectedShippingDate" : "2016-10-19T12:00:00.000Z",
  "logisticsProvider" : "DHL",
  "trackingNumber" : "1234567890"
},
"merchantCustomerNumber" : "cust-732477",
"merchantOrderReferenceNumber" : "order-A12223412",
"merchantReconciliationReferenceNumber" : "recon-A12223412",
"merchantInvoiceReferenceNumber" : "20150112334345",
"note" : "Ihr Einkauf bei Spielauto-Versand.",
"sha256hashedEmailAddress" : "fxP4R-IxH1Eaxpb0f_i5Shc8-FrYrtmx5lx35f9Xzgg",
"minimumAge" : 18,
"callbackUrlStatusUpdates" : "https://spielauto-versand.de/callback/status",
"requestedPreauthorizationValidity" : "2022-11-04"
}
```

HTTP/1.1 201 Created

Location: <https://api.paydirekt.de/api/checkout/v1/checkouts/39d1f7e5-3cce-4ff3-b34d-e0c875d72c65>

Content-Type: application/hal+json;charset=utf-8

```
{
  "checkoutId" : "39d1f7e5-3cce-4ff3-b34d-e0c875d72c65",
  "type" : "ORDER_SECURED",
  "oneClick" : true,
  "status" : "APPROVED",
  "correlationId" : "ext0815-00000255",
  "creationTimestamp" : "2022-10-25T12:35:12.780Z",
  "totalAmount" : 100.0,
  "shippingAmount" : 3.5,
  "orderAmount" : 96.5,
  "refundLimit" : 200,
  "currency" : "EUR",
  "items" : [ {
    "quantity" : 3,
    "name" : "Bobbycar",
    "ean" : "800001303",
    "price" : 25.99
  }, {
    "quantity" : 1,
    "name" : "Helm",
    "price" : 18.53
  } ],
  "shoppingCartType" : "MIXED",
  "merchantOrderReferenceNumber" : "order-A12223412",
  "merchantCustomerNumber" : "cust-732477",
```

```
"merchantInvoiceReferenceNumber" : "20150112334345",
"merchantReconciliationReferenceNumber" : "recon-A12223412",
"note" : "Ihr Einkauf bei Spielauto-Versand.",
"minimumAge" : 18,
"callbackUrlStatusUpdates" : "https://spielauto-versand.de/callback/status",
"contextInformation" : {
  "deviceId" : "8adb80f93",
  "deviceFingerprint" : "ae8faa7f67709ebab82f579945c7e6b5",
  "deviceIpAddress" : "123.123.123.123",
  "terminalId" : "b9f1c1b7",
  "terminalAddress" : {
    "name" : "Pumpe 3",
    "company" : "Mustermann Tankstellen GmbH",
    "street" : "Hamburger Allee",
    "streetNr" : "123A",
    "zip" : "60626",
    "city" : "Frankfurt a.M.",
    "countryCode" : "DE",
    "state" : "Hessen"
  }
},
"thirdPartyId" : "4087160b-1422-48a8-91a9-3f0ab0d5e0d6",
"thirdPartyClientId" : "b7240009-a6b2-4b83-8708-99beb249b0ad",
"thirdPartyClientName" : "e31f1bc6-eef8-497a-b408-4235223678be",
"thirdPartyCustomerAuthorizationId" : "87438cfb-8c0b-4960-bc93-c6a336ed2144",
"thirdPartyMerchantAuthorizationId" : "3d268342-004e-4eae-b6f4-5f0e186635b3",
"deliveryInformation" : {
  "expectedShippingDate" : "2016-10-19T12:00:00.000Z",
  "logisticsProvider" : "DHL",
  "trackingNumber" : "1234567890"
},
"expiryTimestamp" : "2022-10-25T13:05:12.780Z",
"preauthorizationValidity" : "2022-11-04",
"_links" : {
  "captures" : {
    "href" : "https://api.paydirekt.de/api/checkout/v1/checkouts/39d1f7e5-3cce-4ff3-b34d-e0c875d72c65/captures"
  },
  "close" : {
    "href" : "https://api.paydirekt.de/api/checkout/v1/checkouts/39d1f7e5-3cce-4ff3-b34d-e0c875d72c65/close"
  },
  "updateDeliveryInformation" : {
    "href" : "https://api.paydirekt.de/api/checkout/v1/checkouts/39d1f7e5-3cce-4ff3-b34d-e0c875d72c65/deliveryInformation"
  },
  "updateMerchantInvoiceReferenceNumber" : {
    "href" : "https://api.paydirekt.de/api/checkout/v1/checkouts/39d1f7e5-3cce-4ff3-b34d-e0c875d72c65/merchantInvoiceReferenceNumber"
  },
  "self" : {
```

```
"href" : "https://api.paydirekt.de/api/checkout/v1/checkouts/39d1f7e5-3cce-4ff3-  
b34d-e0c875d72c65"  
  }  
}
```

Checkout

Ein Checkout repräsentiert den gesamten Zahlungsvorgang. In den folgenden Kapiteln wird beschrieben, wie Checkouts angelegt werden und wie die zu einem Zahlungsvorgang hinterlegten Daten abgefragt werden können. Hier wird zwischen den Typen *Einmalzahlung* und *Vorbestellung* unterschieden.

Anlage

```
POST /api/checkout/v1/checkouts
```

Neben dem Authentifizierungsendpoint ist dies der einzige Endpoint, der im Webshop fest hinterlegt werden muss. Alle weiteren Ressourcen können über entsprechende Links angesprochen werden.

Request

Feld	Typ	Eigenschaften	Beschreibung
<code>type</code>	String	Pflichtfeld.	<p>Die Art der Bestellung: DIRECT_SALE, ORDER oder ORDER_SECURED.</p> <p>Im Falle eines DIRECT_SALE (Direktbestellung mit Einmalzahlung) wird automatisch eine Zahlungsautorisierung erzeugt. Der Händler erhält sofort eine Zahlungsgarantie.</p> <p>Zu einer ORDER (Vorbestellung oder Bestellung mit Teilzahlungen) können direkt im Anschluss oder später ein oder mehrere Captures angestoßen werden. Für eine Order wird keine Zahlungsgarantie an den Händler ausgesprochen. Die Garantie wird erst mit dem Capture gegeben.</p> <p>Zu einer ORDER_SECURED (gesicherte Vorbestellung oder Bestellung mit Teilzahlungen) können direkt im Anschluss oder später ein oder mehrere Captures angestoßen werden. Für eine gesicherte Vorbestellung wird eine Zahlungsgarantie für den gewählten Zeitraum (maximal 15 Kalendertage) an den Händler ausgesprochen. Captures (Teilzahlungen) werden innerhalb des Garantienzeitraums immer ausgeführt. Weitere Informationen zu Order und Capture finden Sie im Kapitel Capture.</p>
<code>totalAmount</code>	Number	<p>Pflichtfeld.</p> <p>Zwischen 0.01 und 50000.</p> <p>Maximal zwei Nachkommastellen.</p>	<p>Der Gesamtbetrag der Bestellung, inkl. aller Lieferkosten. Es werden maximal 2 Nachkommastellen unterstützt.</p> <p>Im Falle eines DIRECT_SALE wird eine Zahlung für diesen Betrag initiiert.</p> <p>Im Falle einer ORDER können Captures bis maximal zu diesem Betrag angestoßen werden.</p>
<code>shippingAmount</code>	Number	<p>Optional.</p> <p>Mindestens 0.00.</p> <p>Maximal zwei Nachkommastellen.</p>	<p>Die Versandkosten der Bestellung. Dieser Wert wird nicht für Berechnungen verwendet, sondern dient nur zur Information.</p>

Feld	Typ	Eigenschaften	Beschreibung
<code>orderAmount</code>	Number	Optional. Zwischen 0.01 und 50000. Maximal zwei Nachkommastellen.	Der Warenwert der Bestellung, ohne Versandkosten. Dieser Wert wird nicht für Berechnungen verwendet, sondern dient nur zur Information.
<code>refundLimit</code>	Number	Optional. Zwischen 100 und 200. Wird bei Nichtangabe auf 200 gesetzt, das heißt maximal das Zweifache des Bestellwertes kann zurückgezahlt werden.	Maximal zurückzahlbarer Betrag als Prozentwert des Gesamtwertes der Bestellung.
<code>overcapture</code>	Boolean	Optional. (Standard: <code>false</code>)	Flag für Overcapture-Checkouts. Bei einem Overcapture-Checkout darf die Summe der <code>Captures</code> den Warenwert der Bestellung um bis zu 10% übersteigen. Bei aktiviertem Flag muss das <code>finalCapture</code> -Feld des letzten <code>Captures</code> auf <code>true</code> gesetzt sein. <code>overcapture</code> darf nur aktiviert sein, wenn es sich um einen Checkout des Typs <code>ORDER</code> handelt. Kann nur von Händlern verwendet werden, die für dieses Feature freigeschaltet wurden.
<code>currency</code>	String	Pflichtfeld.	Die Währung des Gesamtbetrags. Derzeit wird nur EUR unterstützt.
<code>items</code>	Array von <code>Items</code>	Optional.	Die einzelnen Positionen des Warenkorbs. Es wird empfohlen, diese Werte zu übergeben. Dies verbessert die Erkennung von fraudulenter Vorgängen und hilft, Disputes zu vermeiden.

Feld	Typ	Eigenschaften	Beschreibung
shoppingCartType	String	Optional.	Kategorisiert den Warenkorb eines Checkouts anhand der Eigenschaften der enthaltenen Güter: Der Standardwert ist MIXED . PHYSICAL : Der Warenkorb enthält ausschließlich physische Güter. DIGITAL : Der Warenkorb enthält ausschließlich digitale Güter. MIXED : Der Warenkorb enthält sowohl physische als auch digitale Güter. ANONYMOUS_DONATION : Beim Warenkorb handelt es sich ausschließlich um eine anonyme Spende. AUTHORITIES_PAYMENT : Beim Warenkorb handelt es sich ausschließlich um Behördenzahlungen.
deliveryType	String	Optional.	Der Bestimmungsort einer Lieferung: STANDARD , PACKSTATION oder STORE_PICKUP . Der Standardwert ist STANDARD . STANDARD : Die Güter werden an eine gewöhnliche Postadresse geliefert. PACKSTATION : Die Güter werden an eine Packstation geliefert. STORE_PICKUP : Die Güter werden in der Filiale abgeholt. Dieses Feld enthält bei Express-Checkouts immer den Wert STANDARD und wird nicht anhand der gewählten Lieferoption aktualisiert.
shippingAddress	ShippingAddress	Optional bei AUTHORITIES_PAYMENT und ANONYMOUS_DONATION Warenkörben. Sonst Pflichtfeld. Darf bei Expresscheckoutanlage nicht enthalten sein.	Die Lieferanschrift des Empfängers. Diese Adresse wird dem Kunden nach dem Login im giropay-System zur Kontrolle angezeigt.

Feld	Typ	Eigenschaften	Beschreibung
<code>merchantCustomerNumber</code>	String	Optional. Maximal 50 Zeichen.	Händler-interne Kundennummer des Käufers. Wird dem Kunden in der Transaktionsübersicht angezeigt. Wird dem Händler in der Transaktionsübersicht angezeigt.
<code>merchantOrderReferenceNumber</code>	String	Pflichtfeld. Maximal 20 Zeichen. Nur SEPA-konforme Zeichen .	Händler-interne, eindeutige Bestellnummer für diesen Kaufvorgang. Wird dem Kunden in der Transaktionsübersicht angezeigt. Wird dem Händler in der Transaktionsübersicht angezeigt. Die Bestellnummer wird in der Händler-Lastschrift als <code>instructionId</code> und im Verwendungszweck bei Händler und Käufer verwendet.
<code>merchantInvoiceReferenceNumber</code>	String	Optional. Maximal 100 Zeichen.	Händler-interne, eindeutige Rechnungsnummer für diesen Kauf- bzw. Zahlvorgang. Wird dem Händler in der Transaktionsübersicht angezeigt.
<code>merchantReconciliationReferenceNumber</code>	String	Optional. Maximal 30 Zeichen.	Reconciliation-Nummer. Diese wird bei Direct-Sales in den Verwendungszweck der Händlerzahlung eingefügt. Bei Vorbestellungen wird stattdessen die bei Anlage der Captures gesetzte <code>merchantReconciliationReferenceNumber</code> verwendet.
<code>sha256hashedEmailAddress</code>	String	Optional. Maximal 64 Zeichen.	Die E-Mail Adresse des Käufers als Base-64 encodierter SHA-256 Hash-Wert ohne Padding, sofern vorhanden. Pseudo-Code: <code>sha256hashedEmailAddress = toStringUTF8(base64(sha256(toBytesUTF8(emailAddress))))</code> . Beispiel: Aus der E-Mail-Adresse <code>max@muster.de</code> muss sich exakt der folgende HashWert ergeben (ohne Anführungszeichen): "6JL4VUgVxkq2m+a9I6ScfW2ofJP5y6wsvSaHIsX+iLs=". Diese Information wird zur Fraud Prevention verwendet.

Feld	Typ	Eigenschaften	Beschreibung
<code>redirectUrlAfterSuccess</code>	String	Pflichtfeld, außer bei oneKlick-Checkouts. Maximal 2000 Zeichen.	Die Rücksprung-Adresse des Webshops (inkl. Referenz auf die Bestellung), die nach erfolgreicher Bezahlung aufgerufen wird. Siehe Erfolgsstatus eines Checkouts .
<code>redirectUrlAfterCancellation</code>	String	Pflichtfeld, außer bei oneKlick-Checkouts. Maximal 2000 Zeichen.	Die Rücksprung-Adresse des Webshops (inkl. Referenz auf die Bestellung), die im Falle eines Abbruchs oder technischen Fehlers aufgerufen wird. Damit wird signalisiert, dass der Kaufvorgang grundsätzlich fortgeführt werden kann und im Anschluss ein weiterer, neuer Checkout initiiert werden kann, beispielsweise, weil der Kunde noch einen Artikel in der Bestellung hinzufügen möchte. Siehe Erfolgsstatus eines Checkouts .
<code>redirectUrlAfterRejection</code>	String	Pflichtfeld, außer bei oneKlick-Checkouts. Maximal 2000 Zeichen.	Die Rücksprung-Adresse des Webshops (inkl. Referenz auf die Bestellung), die im Falle einer Abweisung der Zahlung aufgerufen wird. Damit wird signalisiert, dass keine Zahlung autorisiert wurde (z. B. aufgrund falscher TAN-Eingabe, fehlender Bank-Autorisierung oder Betrugsverdacht). Falls das Webshop-System keine Unterscheidung zwischen <code>redirectUrlAfterCancellation</code> und <code>redirectUrlAfterRejection</code> unterstützt, kann in beiden Feldern die gleiche URL angegeben werden. Siehe Erfolgsstatus eines Checkouts .
<code>callbackUrlStatusUpdates</code>	String	Optional. Maximal 2000 Zeichen.	URL des Endpoints, der Mitteilungen zu Statusänderungen des Checkouts empfangen soll.

Feld	Typ	Eigenschaften	Beschreibung
<code>minimumAge</code>	Number	Optional.	Das Mindestalter (in Jahren), das der Käufer erreicht haben muss, um die Bestellung ausführen zu dürfen, beispielsweise, weil sie Artikel enthält, die einer Altersbeschränkung unterliegen (Filme, Computerspiele, ...). Die Prüfung erfolgt direkt nach dem Login des Käufers in das giropay-System. Bei erfolgreicher Verifikation wird der Ablauf ohne weitere Meldung wie gewohnt fortgesetzt. Bei nicht-erfolgreicher Verifikation, d. h. der Kunde hat das erforderliche Mindestalter noch nicht erreicht, erfolgt eine Umleitung auf den Link <code>redirectUrlAfterAgeVerificationFailure</code> (siehe unten). Bei Nicht-Angabe dieses Wertes erfolgt keine Altersverifikation.
<code>redirectUrlAfterAgeVerificationFailure</code>	String	Pflichtfeld, wenn <code>minimumAge</code> gesetzt ist. Maximal 2000 Zeichen.	Die Rücksprung-Adresse des Webshops (inkl. Referenz auf die Bestellung), die im Falle einer nicht erfolgreichen Altersverifikation aufgerufen wird. Damit wird signalisiert, dass die Bestellung abgebrochen wurde, da der Käufer das erforderliche Mindestalter für mindestens einen Artikel in der Bestellung noch nicht erreicht hat. Dieses Feld wird im Falle einer geforderten Altersverifikation, d. h. bei Angabe des Feldes <code>minimumAge</code> , zum Pflichtfeld. Ist keine Altersverifikation erforderlich, ist dieses Feld wegzulassen.
<code>note</code>	String	Optional. Maximal 37 Zeichen.	Freitextfeld, das auf dem Kontoauszug im Feld Verwendungszweck erscheint (nur <code>DIRECT_SALE</code>).
<code>expiryTime</code>	Number	Optional. Minimal 120. Maximal 1800.	Zahl, die die Gültigkeit des Checkouts ab Anlage in Sekunden angibt. Bei Nicht-Angabe werden 1800 Sekunden angenommen.
<code>deliveryInformation</code>	Object	Optional.	Enthält Informationen über die Lieferung.

Feld	Typ	Eigenschaften	Beschreibung
<code>deliveryInformation.expectedShippingDate</code>	String	Optional. Zeitstempel im ISO-8601 -Format.	Erwartetes Versanddatum.
<code>deliveryInformation.logisticsProvider</code>	String	Optional.	Paket-Dienstleister.
<code>deliveryInformation.trackingNumber</code>	String	Optional.	Sendungsnummer.
<code>customerAuthorizationRequest</code>	Object	Optional. Nur wirksam, wenn der Checkout durch eine oneKlick-Anwendung, einen für oneKlick freigeschalteten Händler selbst oder dessen CPSP angelegt wird.	Gibt an, dass dem Käufer im Rahmen des Checkout-Vorgangs die Erteilung einer oneKlick-Autorisierung zugunsten der aktuellen oneKlick-Anwendung bzw. des aktuellen Händlershops angeboten werden soll. Informationen hierzu finden Sie im Kapitel Käuferautorisierung während eines Checkouts .
<code>customerAuthorizationRequest.scopes</code>	Array	Pflichtfeld, falls <code>customerAuthorizationRequest</code> übergeben wurde. Darf nicht leer sein.	Die vom Käufer zu gewährenden Rechte. Es wird momentan ausschließlich <code>PERFORM_CHECKOUT</code> unterstützt.
<code>customerAuthorizationRequest.callbackUrlStatusUpdates</code>	String	Optional.	URL des Endpoints, der Mitteilungen zu Statusänderungen der oneKlick-Autorisierung empfangen soll.
<code>requestedPreauthorizationValidity</code>	String	Optional. Datum im Format yyyy-mm-dd. Maximal 15 Kalendertage in der Zukunft (entspricht dem Standardwert, sofern nicht befüllt).	Gewünschter Garantiezeitraum (maximal 15 Kalendertage) für eine gesicherte Vorbestellung (<code>ORDER_SECURED</code>).

Response

In der Response wird der angelegte Checkout zurückgegeben. Die Struktur ist identisch mit dem Checkout Endpoint [GET](#)-Aufruf.

Im HTTP Header-Feld `Location` ist die URL der angelegten Checkout-Ressource, inkl. Checkout-ID

enthalten. Die Checkout-ID wird im Response Body ebenfalls zurückgegeben.

Links

Relation	Description
self	Link zu dieser Ressource.
approve	Link zur giropay-Webseite, zu dem der Käufer per HTTP Code 302 weitergeleitet werden muss, um die Zahlung zu bestätigen. Die Weiterleitung darf nicht über ein HTML Formular (unter Nutzung des approve Links im <code>action</code> Attribute des <code>form</code> Tags) erfolgen. Die Bestätigung durch den Kunden muss innerhalb von 30 Minuten erfolgen. Dieser Link ist nur vorhanden, wenn der Checkout vom Kunden noch nicht autorisiert wurde. Der Aufbau bzw. die Parameter des Links können sich aufgrund von internen Weiterentwicklungen verändern. Verwenden Sie daher ausschließlich den von unserem System zur Laufzeit zurückgelieferten Link.
updateDeliveryInformation	Link zum Update der Lieferinformation. Dieser Link ist nur vorhanden, wenn die Update Frist für diesen Checkout noch nicht überschritten ist.
updateMerchantInvoiceReferenceNumber	Link zum Update der Rechnungsreferenznummer. Dieser Link ist nur vorhanden, wenn die Update Frist für diesen Checkout noch nicht überschritten ist.

Der [approve](#)-Link gibt an, zu welcher Seite der Benutzer via HTTP Redirect (302) weitergeleitet werden muss, um die Zahlung zu bestätigen.

Return Codes

Status Code	Message Code	Beschreibung
201 (Created)		Der Checkout wurde angelegt. Anschließend muss ein Redirect auf den in der Resource hinterlegten Link approve durchgeführt werden.
400 (Bad Request)	VALIDATION_ERROR	Der Request ist syntaktisch ungültig.

Status Code	Message Code	Beschreibung
400 (Bad Request)	CONVERSION_ERROR	Der Request enthält ungültige Zeichen. Häufigste Ursache hierfür ist, dass der Request nicht korrekt UTF-8 codiert ist. Falls die ungültigen Zeichen einem Attribut im Request zugeordnet werden können, wird dieses im Feld path genannt und der ungültige Wert im Feld content übermittelt.
401 (Unauthorized)	ACCESS_TOKEN_EXPIRED	Siehe Resource Access .
401 (Unauthorized)	UNAUTHORIZED	Der Zugriff wurde verweigert. Bitte wenden Sie sich an den giropay-Support.
422 (Unprocessable Entity)	MERCHANT_BANKACCOUNT_LOCKED	Das Bankkonto des Händlers wurde gesperrt und es können keine neue Checkouts oder Captures angelegt werden. Der Händler sollte sich an den giropay-Support wenden.
422 (Unprocessable Entity)	PSP_LOCKED	Der Payment Service Provider (PSP) ist gesperrt, es können keine neuen Checkouts oder Captures angelegt werden.

Beispiel (Direct Sale, inkl. Altersverifikation)

```
POST /api/checkout/v1/checkouts HTTP/1.1
Content-Type: application/json;charset=utf-8
Authorization: Bearer <access_token>
```

```
{
  "type" : "DIRECT_SALE",
  "totalAmount" : 100.0,
  "shippingAmount" : 3.5,
  "orderAmount" : 96.5,
  "refundLimit" : 200,
  "items" : [ {
    "quantity" : 3,
    "name" : "Bobbycar",
    "ean" : "800001303",
    "price" : 25.99
  }, {
    "quantity" : 1,
    "name" : "Helm",
    "price" : 18.53
  } ],
  "shoppingCartType" : "MIXED",
  "currency" : "EUR",
  "overcapture" : false,
  "shippingAddress" : {
    "addresseeGivenName" : "Marie",
```

```
"addresseeLastName" : "Mustermann",
"company" : "Musterbau GmbH & Co KG",
"street" : "Kastanienallee",
"streetNr" : "999",
"additionalAddressInformation" : "Haus 4",
"zip" : "90402",
"city" : "Schwaig",
"countryCode" : "DE",
"state" : "Bayern"
},
"deliveryInformation" : {
  "expectedShippingDate" : "2016-10-19T12:00:00.000Z",
  "logisticsProvider" : "DHL",
  "trackingNumber" : "1234567890"
},
"merchantCustomerNumber" : "cust-732477",
"merchantOrderReferenceNumber" : "order-A12223412",
"merchantReconciliationReferenceNumber" : "recon-A12223412",
"merchantInvoiceReferenceNumber" : "20150112334345",
"note" : "Ihr Einkauf bei Spielauto-Versand.",
"sha256hashedEmailAddress" : "fxP4R-IxH1Eaxpb0f_i5Shc8-FrYrtmx5lx35f9Xzgg",
"minimumAge" : 18,
"expiryTime" : 1800,
"redirectUrlAfterSuccess" : "https://spielauto-versand.de/order/123/success",
"redirectUrlAfterCancellation" : "https://spielauto-versand.de/order/123/cancellation",
"redirectUrlAfterAgeVerificationFailure" : "https://spielauto-versand.de/order/123/ageverificationfailed",
"redirectUrlAfterRejection" : "https://spielauto-versand.de/order/123/rejection",
"callbackUrlStatusUpdates" : "https://spielauto-versand.de/callback/status",
"customerAuthorizationRequest" : {
  "scopes" : [ "PERFORM_CHECKOUT" ],
  "callbackUrlStatusUpdates" : "http://one.klick/callbackUrlStatusUpdates"
}
}
```

HTTP/1.1 201 Created

Location: <https://api.paydirekt.de/api/checkout/v1/checkouts/666dbf07-69b1-4d0c-94c1-df69e77fd609>

Content-Type: application/hal+json;charset=utf-8

```
{
  "checkoutId" : "666dbf07-69b1-4d0c-94c1-df69e77fd609",
  "type" : "DIRECT_SALE",
  "status" : "OPEN",
  "creationTimestamp" : "2022-10-25T12:30:43.886Z",
  "totalAmount" : 100.0,
  "shippingAmount" : 3.5,
  "orderAmount" : 96.5,
  "refundLimit" : 200,
}
```



```
"currency" : "EUR",
"items" : [ {
  "quantity" : 3,
  "name" : "Bobbycar",
  "ean" : "800001303",
  "price" : 25.99
}, {
  "quantity" : 1,
  "name" : "Helm",
  "price" : 18.53
} ],
"shoppingCartType" : "MIXED",
"deliveryType" : "STANDARD",
"shippingAddress" : {
  "addresseeGivenName" : "Marie",
  "addresseeLastName" : "Mustermann",
  "company" : "Musterbau GmbH & Co KG",
  "street" : "Kastanienallee",
  "streetNr" : "999",
  "additionalAddressInformation" : "Haus 4",
  "zip" : "90402",
  "city" : "Schwaig",
  "countryCode" : "DE",
  "state" : "Bayern"
},
"merchantOrderReferenceNumber" : "order-A12223412",
"merchantCustomerNumber" : "cust-732477",
"merchantInvoiceReferenceNumber" : "20150112334345",
"merchantReconciliationReferenceNumber" : "recon-A12223412",
"note" : "Ihr Einkauf bei Spielauto-Versand.",
"minimumAge" : 18,
"redirectUrlAfterSuccess" : "https://spielauto-versand.de/order/123/success",
"redirectUrlAfterCancellation" : "https://spielauto-
versand.de/order/123/cancellation",
"redirectUrlAfterAgeVerificationFailure" : "https://spielauto-
versand.de/order/123/ageverificationfailed",
"redirectUrlAfterRejection" : "https://spielauto-versand.de/order/123/rejection",
"callbackUrlStatusUpdates" : "https://spielauto-versand.de/callback/status",
"thirdPartyId" : "08ec0853-6316-4fb9-914c-e94cd5ed627b",
"thirdPartyClientId" : "7281895b-9bd0-4dfa-8a33-0b30e988ef17",
"thirdPartyClientName" : "Managed client of CPSP shop",
"thirdPartyMerchantAuthorizationId" : "0b828267-73a3-4936-a9f5-e541834de236",
"deliveryInformation" : {
  "expectedShippingDate" : "2016-10-19T12:00:00.000Z",
  "logisticsProvider" : "DHL",
  "trackingNumber" : "1234567890"
},
"expiryTimestamp" : "2022-10-25T13:00:43.886Z",
"customerAuthorizationDemand" : {
  "thirdPartyId" : "08ec0853-6316-4fb9-914c-e94cd5ed627b",
  "thirdPartyClientId" : "7281895b-9bd0-4dfa-8a33-0b30e988ef17",
```

```

"thirdPartyClientName" : "Managed client of CPSP shop",
"scopes" : [ "PERFORM_CHECKOUT" ],
"authorizationPreapproved" : false,
"callbackUrlStatusUpdates" : "http://one.klick/callbackUrlStatusUpdates"
},
"_links" : {
  "approve" : {
    "href" : "https://localhost/checkout/?p=666dbf07-69b1-4d0c-94c1-
df69e77fd609#/checkout/666dbf07-69b1-4d0c-94c1-df69e77fd609"
  },
  "updateDeliveryInformation" : {
    "href" : "https://api.paydirekt.de/api/checkout/v1/checkouts/666dbf07-69b1-4d0c-
94c1-df69e77fd609/deliveryInformation"
  },
  "updateMerchantInvoiceReferenceNumber" : {
    "href" : "https://api.paydirekt.de/api/checkout/v1/checkouts/666dbf07-69b1-4d0c-
94c1-df69e77fd609/merchantInvoiceReferenceNumber"
  },
  "self" : {
    "href" : "https://api.paydirekt.de/api/checkout/v1/checkouts/666dbf07-69b1-4d0c-
94c1-df69e77fd609"
  }
}
}

```

Beispiel (Order)

POST /api/checkout/v1/checkouts HTTP/1.1
Content-Type: application/json;charset=utf-8
Authorization: Bearer <access_token>

```
{
  "type" : "ORDER",
  "totalAmount" : 100.0,
  "shippingAmount" : 3.5,
  "orderAmount" : 96.5,
  "refundLimit" : 200,
  "items" : [ {
    "quantity" : 3,
    "name" : "Bobbycar",
    "ean" : "800001303",
    "price" : 25.99
  }, {
    "quantity" : 1,
    "name" : "Helm",
    "price" : 18.53
  } ],
  "shoppingCartType" : "PHYSICAL",
  "deliveryType" : "STANDARD",
  "currency" : "EUR",
  "shippingAddress" : {
    "addresseeGivenName" : "Marie",
    "addresseeLastName" : "Mustermann",
    "company" : "Musterbau GmbH & Co KG",
    "street" : "Kastanienallee",
    "streetNr" : "999",
    "additionalAddressInformation" : "Haus 4",
    "zip" : "90402",
    "city" : "Schwaig",
    "countryCode" : "DE",
    "state" : "Bayern"
  },
  "merchantCustomerNumber" : "cust-732477",
  "merchantOrderReferenceNumber" : "order-A12223412",
  "merchantInvoiceReferenceNumber" : "20150112334345",
  "expiryTime" : 1800,
  "redirectUrlAfterSuccess" : "https://spielauto-versand.de/order/123/success",
  "redirectUrlAfterCancellation" : "https://spielauto-versand.de/order/123/cancellation",
  "redirectUrlAfterRejection" : "https://spielauto-versand.de/order/123/rejection",
  "callbackUrlStatusUpdates" : "https://spielauto-versand.de/callback/status"
}
```

HTTP/1.1 201 Created
Location: <https://api.paydirekt.de/api/checkout/v1/checkouts/bd3378ba-325c-47a9-9139-aaf39848ec3e>

Content-Type: application/hal+json;charset=utf-8

```
{
  "checkoutId" : "bd3378ba-325c-47a9-9139-aaf39848ec3e",
  "type" : "ORDER",
  "status" : "OPEN",
  "creationTimestamp" : "2022-10-25T12:31:18.078Z",
  "totalAmount" : 100.0,
  "shippingAmount" : 3.5,
  "orderAmount" : 96.5,
  "refundLimit" : 200,
  "currency" : "EUR",
  "items" : [ {
    "quantity" : 3,
    "name" : "Bobbycar",
    "ean" : "800001303",
    "price" : 25.99
  }, {
    "quantity" : 1,
    "name" : "Helm",
    "price" : 18.53
  } ],
  "shoppingCartType" : "PHYSICAL",
  "deliveryType" : "STANDARD",
  "shippingAddress" : {
    "addresseeGivenName" : "Marie",
    "addresseeLastName" : "Mustermann",
    "company" : "Musterbau GmbH & Co KG",
    "street" : "Kastanienallee",
    "streetNr" : "999",
    "additionalAddressInformation" : "Haus 4",
    "zip" : "90402",
    "city" : "Schwaig",
    "countryCode" : "DE",
    "state" : "Bayern"
  },
  "merchantOrderReferenceNumber" : "order-A12223412",
  "merchantCustomerNumber" : "cust-732477",
  "merchantInvoiceReferenceNumber" : "20150112334345",
  "redirectUrlAfterSuccess" : "https://spielauto-versand.de/order/123/success",
  "redirectUrlAfterCancellation" : "https://spielauto-versand.de/order/123/cancellation",
  "redirectUrlAfterRejection" : "https://spielauto-versand.de/order/123/rejection",
  "callbackUrlStatusUpdates" : "https://spielauto-versand.de/callback/status",
  "expiryTimestamp" : "2022-10-25T13:01:18.078Z",
  "_links" : {
    "approve" : {
      "href" : "https://localhost/checkout/?p=bd3378ba-325c-47a9-9139-aaf39848ec3e#/checkout/bd3378ba-325c-47a9-9139-aaf39848ec3e"
    },
    "updateDeliveryInformation" : {
```

```
"href" : "https://api.paydirekt.de/api/checkout/v1/checkouts/bd3378ba-325c-47a9-9139-aaf39848ec3e/deliveryInformation"
},
"updateMerchantInvoiceReferenceNumber" : {
  "href" : "https://api.paydirekt.de/api/checkout/v1/checkouts/bd3378ba-325c-47a9-9139-aaf39848ec3e/merchantInvoiceReferenceNumber"
},
"self" : {
  "href" : "https://api.paydirekt.de/api/checkout/v1/checkouts/bd3378ba-325c-47a9-9139-aaf39848ec3e"
}
}
```

Beispiel (Order Secured)

POST /api/checkout/v1/checkouts HTTP/1.1
Content-Type: application/json;charset=utf-8
Authorization: Bearer <access_token>

```
{
  "type" : "ORDER_SECURED",
  "totalAmount" : 100.0,
  "shippingAmount" : 3.5,
  "orderAmount" : 96.5,
  "refundLimit" : 200,
  "items" : [ {
    "quantity" : 3,
    "name" : "Bobbycar",
    "ean" : "800001303",
    "price" : 25.99
  }, {
    "quantity" : 1,
    "name" : "Helm",
    "price" : 18.53
  } ],
  "shoppingCartType" : "PHYSICAL",
  "deliveryType" : "STANDARD",
  "currency" : "EUR",
  "shippingAddress" : {
    "addresseeGivenName" : "Marie",
    "addresseeLastName" : "Mustermann",
    "company" : "Musterbau GmbH & Co KG",
    "street" : "Kastanienallee",
    "streetNr" : "999",
    "additionalAddressInformation" : "Haus 4",
    "zip" : "90402",
    "city" : "Schwaig",
    "countryCode" : "DE",
    "state" : "Bayern"
  },
  "merchantCustomerNumber" : "cust-732477",
  "merchantOrderReferenceNumber" : "order-A12223412",
  "merchantInvoiceReferenceNumber" : "20150112334345",
  "expiryTime" : 1800,
  "redirectUrlAfterSuccess" : "https://spielauto-versand.de/order/123/success",
  "redirectUrlAfterCancellation" : "https://spielauto-
versand.de/order/123/cancellation",
  "redirectUrlAfterRejection" : "https://spielauto-versand.de/order/123/rejection",
  "callbackUrlStatusUpdates" : "https://spielauto-versand.de/callback/status",
  "requestedPreauthorizationValidity" : "2022-10-28"
}
```

HTTP/1.1 201 Created

Location: <https://api.paydirekt.de/api/checkout/v1/checkouts/a20e18a4-9e99-4988-b156->

486ce3948014

Content-Type: application/hal+json;charset=utf-8

```
{
  "checkoutId" : "a20e18a4-9e99-4988-b156-486ce3948014",
  "type" : "ORDER_SECURED",
  "status" : "OPEN",
  "creationTimestamp" : "2022-10-25T12:32:15.791Z",
  "totalAmount" : 100.0,
  "shippingAmount" : 3.5,
  "orderAmount" : 96.5,
  "refundLimit" : 200,
  "currency" : "EUR",
  "items" : [ {
    "quantity" : 3,
    "name" : "Bobbycar",
    "ean" : "800001303",
    "price" : 25.99
  }, {
    "quantity" : 1,
    "name" : "Helm",
    "price" : 18.53
  } ],
  "shoppingCartType" : "PHYSICAL",
  "deliveryType" : "STANDARD",
  "shippingAddress" : {
    "addresseeGivenName" : "Marie",
    "addresseeLastName" : "Mustermann",
    "company" : "Musterbau GmbH & Co KG",
    "street" : "Kastanienallee",
    "streetNr" : "999",
    "additionalAddressInformation" : "Haus 4",
    "zip" : "90402",
    "city" : "Schwaig",
    "countryCode" : "DE",
    "state" : "Bayern"
  },
  "merchantOrderReferenceNumber" : "order-A12223412",
  "merchantCustomerNumber" : "cust-732477",
  "merchantInvoiceReferenceNumber" : "20150112334345",
  "redirectUrlAfterSuccess" : "https://spielauto-versand.de/order/123/success",
  "redirectUrlAfterCancellation" : "https://spielauto-versand.de/order/123/cancellation",
  "redirectUrlAfterRejection" : "https://spielauto-versand.de/order/123/rejection",
  "callbackUrlStatusUpdates" : "https://spielauto-versand.de/callback/status",
  "expiryTimestamp" : "2022-10-25T13:02:15.791Z",
  "_links" : {
    "approve" : {
      "href" : "https://localhost/checkout/?p=a20e18a4-9e99-4988-b156-486ce3948014#/checkout/a20e18a4-9e99-4988-b156-486ce3948014"
    }
  },
}
```

```
"updateDeliveryInformation" : {
  "href" : "https://api.paydirekt.de/api/checkout/v1/checkouts/a20e18a4-9e99-4988-
b156-486ce3948014/deliveryInformation"
},
"updateMerchantInvoiceReferenceNumber" : {
  "href" : "https://api.paydirekt.de/api/checkout/v1/checkouts/a20e18a4-9e99-4988-
b156-486ce3948014/merchantInvoiceReferenceNumber"
},
"self" : {
  "href" : "https://api.paydirekt.de/api/checkout/v1/checkouts/a20e18a4-9e99-4988-
b156-486ce3948014"
}
}
}
```

Beispiel (Validierungsfehler)

POST /api/checkout/v1/checkout HTTP/1.1
Content-Type: application/json;charset=utf-8
Authorization: Bearer <access_token>

```
{
  "type" : "DIRECT_SALE",
  "totalAmount" : 100.0,
  "shippingAmount" : 5.999,
  "orderAmount" : 96.5,
  "refundLimit" : 200,
  "items" : [ {
    "quantity" : 3,
    "name" : "Bobbycar",
    "ean" : "800001303",
    "price" : 25.99
  }, {
    "quantity" : 1,
    "name" : "Helm",
    "price" : 18.53
  } ],
  "shoppingCartType" : "MIXED",
  "currency" : "EUR",
  "overcapture" : false,
  "shippingAddress" : {
    "addresseeGivenName" : "Marie",
    "addresseeLastName" : "Mustermann",
    "company" : "Musterbau GmbH & Co KG",
    "street" : "Kastanienallee",
    "streetNr" : "999",
    "additionalAddressInformation" : "Haus 4",
    "zip" : "90402",
    "city" : "Schwaig",
    "countryCode" : "DE",
    "state" : "Bayern"
  },
  "merchantCustomerNumber" : "cust-732477",
  "merchantOrderReferenceNumber" : "order-A12223412",
  "merchantReconciliationReferenceNumber" : "recon-A12223412",
  "merchantInvoiceReferenceNumber" : "20150112334345",
  "note" : "Ihr Einkauf bei Spielauto-Versand.",
  "sha256hashedEmailAddress" : "fxP4R-IxH1Eaxpb0f_i5Shc8-FrYrtmx5lx35f9Xzgg",
  "expiryTime" : 1800,
  "redirectUrlAfterSuccess" : "https://spielauto-versand.de/order/123/success",
  "redirectUrlAfterCancellation" : "https://spielauto-versand.de/order/123/cancellation",
  "redirectUrlAfterAgeVerificationFailure" : "https://spielauto-versand.de/order/123/underAge",
  "redirectUrlAfterRejection" : "https://spielauto-versand.de/order/123/rejection",
  "callbackUrlStatusUpdates" : "https://spielauto-versand.de/callback/status"
}
```

HTTP/1.1 400 Bad Request

Content-Type: application/json;charset=utf-8

```
{
  "messages" : [ {
    "severity" : "ERROR",
    "code" : "VALIDATION_ERROR",
    "path" : "shippingAmount",
    "reasonCode" : "INVALID_FORMAT",
    "logref" : "7a9ff422-aabd-4d06-8a81-e4328741389a:3keQiD5j4Kq"
  } ]
}
```

Beispiel (Fachlicher Fehler)

POST /api/checkout/v1/checkouts HTTP/1.1
Content-Type: application/json;charset=utf-8
Authorization: Bearer <access_token>

```
{
  "type" : "DIRECT_SALE",
  "totalAmount" : 100.0,
  "shippingAmount" : 3.5,
  "orderAmount" : 96.5,
  "refundLimit" : 200,
  "items" : [ {
    "quantity" : 3,
    "name" : "Bobbycar",
    "ean" : "800001303",
    "price" : 25.99
  }, {
    "quantity" : 1,
    "name" : "Helm",
    "price" : 18.53
  } ],
  "shoppingCartType" : "MIXED",
  "currency" : "EUR",
  "overcapture" : false,
  "shippingAddress" : {
    "addresseeGivenName" : "Marie",
    "addresseeLastName" : "Mustermann",
    "company" : "Musterbau GmbH & Co KG",
    "street" : "Kastanienallee",
    "streetNr" : "999",
    "additionalAddressInformation" : "Haus 4",
    "zip" : "90402",
    "city" : "Schwaig",
    "countryCode" : "DE",
    "state" : "Bayern"
  },
  "merchantCustomerNumber" : "cust-732477",
  "merchantOrderReferenceNumber" : "order-A12223412",
  "merchantReconciliationReferenceNumber" : "recon-A12223412",
  "merchantInvoiceReferenceNumber" : "20150112334345",
  "note" : "Ihr Einkauf bei Spielauto-Versand.",
  "sha256hashedEmailAddress" : "fxP4R-IxH1Eaxpb0f_i5Shc8-FrYrtmx5lx35f9Xzgg",
  "expiryTime" : 1800,
  "redirectUrlAfterSuccess" : "https://spielauto-versand.de/order/123/success",
  "redirectUrlAfterCancellation" : "https://spielauto-versand.de/order/123/cancellation",
  "redirectUrlAfterAgeVerificationFailure" : "https://spielauto-versand.de/order/123/underAge",
  "redirectUrlAfterRejection" : "https://spielauto-versand.de/order/123/rejection",
  "callbackUrlStatusUpdates" : "https://spielauto-versand.de/callback/status"
}
```

HTTP/1.1 422 Unprocessable Entity
Content-Type: application/json;charset=utf-8

```
{  
  "messages" : [ {  
    "severity" : "ERROR",  
    "code" : "MERCHANT_BANKACCOUNT_LOCKED",  
    "logref" : "57ef578c-3ae4-4a20-810f-bf373deef500:2Xto6fjzm2R"  
  } ]  
}
```

Beispiel (Konvertierungsfehler)

POST /api/checkout/v1/checkout HTTP/1.1
Content-Type: application/json;charset=utf-8
Authorization: Bearer <access_token>

```
{
  "type" : "DIRECT_SALE",
  "totalAmount" : 100.0,
  "shippingAmount" : 3.5,
  "orderAmount" : 96.5,
  "refundLimit" : 200,
  "items" : [ {
    "quantity" : 3,
    "name" : "Bobbycar",
    "ean" : "800001303",
    "price" : 25.99
  }, {
    "quantity" : 1,
    "name" : "Helm",
    "price" : 18.53
  } ],
  "shoppingCartType" : "MIXED",
  "currency" : "EUR",
  "overcapture" : false,
  "shippingAddress" : {
    "addresseeGivenName" : "Marie",
    "addresseeLastName" : "Mustermann",
    "company" : "ungültige Zeichen sind z.B.: €¥£\0\0\0",
    "street" : "Kastanienallee",
    "streetNr" : "999",
    "additionalAddressInformation" : "Haus 4",
    "zip" : "90402",
    "city" : "Schwaig",
    "countryCode" : "DE",
    "state" : "Bayern"
  },
  "merchantCustomerNumber" : "cust-732477",
  "merchantOrderReferenceNumber" : "order-A12223412",
  "merchantReconciliationReferenceNumber" : "recon-A12223412",
  "merchantInvoiceReferenceNumber" : "20150112334345",
  "note" : "Ihr Einkauf bei Spielauto-Versand.",
  "sha256hashedEmailAddress" : "fxP4R-IxH1Eaxpb0f_i5Shc8-FrYrtmx5lx35f9Xzgg",
  "expiryTime" : 1800,
  "redirectUrlAfterSuccess" : "https://spielauto-versand.de/order/123/success",
  "redirectUrlAfterCancellation" : "https://spielauto-versand.de/order/123/cancellation",
  "redirectUrlAfterAgeVerificationFailure" : "https://spielauto-versand.de/order/123/underAge",
  "redirectUrlAfterRejection" : "https://spielauto-versand.de/order/123/rejection",
  "callbackUrlStatusUpdates" : "https://spielauto-versand.de/callback/status"
}
```

```
HTTP/1.1 400 Bad Request
Content-Type: application/json;charset=utf-8
```

```
{
  "messages" : [ {
    "severity" : "ERROR",
    "code" : "CONVERSION_ERROR",
    "path" : "shippingAddress.company",
    "reasonCode" : "HTTP_MESSAGE_NOT_READABLE",
    "logref" : "8f9830d2-9f5f-4b3e-a91b-b58f6eacccf7:a49SurA7TJR",
    "content" : "ungültige Zeichen sind z.B.: €¥£\ \ \ \ "
  } ]
}
```

Abfrage

```
GET /api/checkout/v1/checkouts/{checkoutId}
```

Ein zuvor angelegter Checkout kann über die GET-Methode aufgerufen werden, um den Status der Zahlung und andere Eigenschaften abzufragen, sowie um die Links zu weiteren Ressourcen, z. B. Captures und Refunds, zu ermitteln.

Response

Feld	Typ	Eigenschaften	Beschreibung
<code>checkoutId</code>	String	Immer vorhanden.	Eindeutige UUID aus 36 Zeichen für diesen Checkout-Vorgang. Wird vom giropay-System vergeben.

Feld	Typ	Eigenschaften	Beschreibung
type	String	Immer vorhanden.	<p>Die Art der Bestellung: DIRECT_SALE, ORDER oder ORDER_SECURED.</p> <p>Im Falle eines DIRECT_SALE (Direktbestellung mit Einmalzahlung) wird automatisch eine Zahlungsautorisierung erzeugt. Der Händler erhält sofort eine Zahlungsgarantie.</p> <p>Zu einer ORDER (Vorbestellung oder Bestellung mit Teilzahlungen) können direkt im Anschluss oder später ein oder mehrere Captures angestoßen werden. Für eine Order wird keine Zahlungsgarantie an den Händler ausgesprochen. Die Garantie wird erst mit dem Capture gegeben.</p> <p>Zu einer ORDER_SECURED (gesicherte Vorbestellung oder Bestellung mit Teilzahlungen) können direkt im Anschluss oder später ein oder mehrere Captures angestoßen werden. Für eine gesicherte Vorbestellung wird eine Zahlungsgarantie für den gewählten Zeitraum (maximal 15 Kalendertage) an den Händler ausgesprochen. Captures (Teilzahlungen) werden innerhalb des Garantienzeitraums immer ausgeführt. Weitere Informationen zu Order und Capture finden Sie im Kapitel Capture.</p>

Feld	Typ	Eigenschaften	Beschreibung
<code>status</code>	String	Immer vorhanden.	<p>Status des Checkouts: OPEN, PENDING, APPROVED, REJECTED, CANCELED, CLOSED oder EXPIRED.</p> <p>OPEN: Der Checkout wurde neu angelegt und vom Benutzer noch nicht bestätigt.</p> <p>PENDING: Der Checkout wurde vom Kunden bestätigt, aber noch nicht per <i>execute</i> aktiviert (derzeit nur <i>giropay Express</i>)</p> <p>APPROVED: Der Checkout wurde vom Kunden bestätigt und vom System genehmigt. Im Falle einer ORDER können jetzt Captures angelegt werden.</p> <p>REJECTED: Der Checkout wurde vom System abgelehnt. Die Zahlung war nicht erfolgreich. Es ist nicht zu erwarten, dass die Zahlung bei erneutem Versuch mit diesem Benutzer erfolgreich sein wird.</p> <p>CANCELED: Der Checkout wurde vom Kunden abgebrochen oder es kam zu einem technischen Fehler. Der Checkout kann nicht weiter bearbeitet werden.</p> <p>CLOSED: Nur ORDER: Die Bestellung ist geschlossen, es können keine weiteren Captures durchgeführt werden. Refunds sind weiter möglich.</p> <p>EXPIRED: Der Checkout ist abgelaufen und kann nicht weiter verwendet werden. Der Benutzer hat den Checkout nicht innerhalb von 30 Minuten bestätigt.</p>
<code>correlationId</code>	String	Vorhanden, sobald sich ein Kunde eingeloggt hat. UUID aus 36 Zeichen.	Die eindeutige Identifikation des Checkouts und aller dazugehöriger Transaktionen.

Feld	Typ	Eigenschaften	Beschreibung
<code>creationTimestamp</code>	String	Immer vorhanden.	Der Erzeugungszeitpunkt dieses Checkouts. Wird vom giropay-System vergeben. Formatierung entsprechend ISO-8601 durch den Standard Java DateTimeFormatter . Abweichend davon werden immer drei Nachkommastellen für die Millisekunden ausgegeben.
<code>expiryTimestamp</code>	String	Immer vorhanden.	Der Ablaufzeitpunkt dieses Checkouts. Kann über das Feld <code>expiryTime</code> bei Anlage angepasst werden, ansonsten wird vom giropay-System als Standardwert 30 Minuten ab Erzeugung vergeben. Formatierung entsprechend ISO-8601 durch den Standard Java DateTimeFormatter . Abweichend davon werden immer drei Nachkommastellen für die Millisekunden ausgegeben.
<code>totalAmount</code>	Number	Immer vorhanden.	Der Gesamtbetrag der Bestellung, inkl. aller Lieferkosten. Es werden maximal 2 Nachkommastellen unterstützt. Im Falle eines <code>DIRECT_SALE</code> wird eine Zahlung für diesen Betrag initiiert. Im Falle einer <code>ORDER</code> können Captures bis maximal zu diesem Betrag angestoßen werden.
<code>shippingAmount</code>	Number	Vorhanden, falls bei Anlage übergeben.	Die Versandkosten der Bestellung. Dieser Wert wird nicht für Berechnungen verwendet, sondern dient nur zur Information.
<code>orderAmount</code>	Number	Vorhanden, falls bei Anlage übergeben.	Der Warenwert der Bestellung, ohne Versandkosten. Dieser Wert wird nicht für Berechnungen verwendet, sondern dient nur zur Information.
<code>refundLimit</code>	Number	Vorhanden, falls bei Anlage übergeben.	Maximal zurückzahlbarer Betrag als Prozentwert des Gesamtwertes der Bestellung.

Feld	Typ	Eigenschaften	Beschreibung
<code>overcapture</code>	Boolean	Vorhanden, falls bei Anlage übergeben.	Flag für Overcapture-Checkouts. Bei einem Overcapture-Checkout darf die Summe der <code>Captures</code> den Warenwert der Bestellung um bis zu 10% übersteigen. Bei aktiviertem Flag muss das <code>finalCapture</code> -Feld des letzten <code>Captures</code> auf <code>true</code> gesetzt sein. <code>overcapture</code> darf nur aktiviert sein, wenn es sich um einen Checkout des Typs <code>ORDER</code> handelt. Kann nur von Händlern verwendet werden, die für dieses Feature freigeschaltet wurden.
<code>maxCapturableAmount</code>	Number	Vorhanden, falls Overcapture-Flag gesetzt.	Bei gesetztem Overcapture-Flag enthält dieses Feld den maximalen Betrag, der insgesamt abgerufen werden kann.
<code>maxOvercaptureDifference</code>	Number	Vorhanden, falls Overcapture-Flag vorhanden.	Bei gesetztem Overcapture-Flag enthält dieses Feld den zum Gesamtbetrag zusätzlichen Betrag, der in der Summe abgerufen werden kann.
<code>currency</code>	String	Immer vorhanden.	Die Währung des Gesamtbetrags. Derzeit wird nur EUR unterstützt.
<code>items</code>	Array von <code>Items</code>	Vorhanden, falls bei Anlage übergeben.	Die einzelnen Positionen des Warenkorbs. Es wird empfohlen, diese Werte zu übergeben. Dies verbessert die Erkennung von fraudulenter Vorgängen und hilft, Disputes zu vermeiden.

Feld	Typ	Eigenschaften	Beschreibung
shoppingCartType	String	Vorhanden, falls bei Anlage übergeben.	Kategorisiert den Warenkorb eines Checkouts anhand der Eigenschaften der enthaltenen Güter: Der Standardwert ist MIXED . PHYSICAL : Der Warenkorb enthält ausschließlich physische Güter. DIGITAL : Der Warenkorb enthält ausschließlich digitale Güter. MIXED : Der Warenkorb enthält sowohl physische als auch digitale Güter. ANONYMOUS_DONATION : Beim Warenkorb handelt es sich ausschließlich um eine anonyme Spende. AUTHORITIES_PAYMENT : Beim Warenkorb handelt es sich ausschließlich um Behördenzahlungen.
deliveryType	String	Vorhanden, falls bei Anlage übergeben.	Der Bestimmungsort einer Lieferung: STANDARD , PACKSTATION oder STORE_PICKUP . Der Standardwert ist STANDARD . STANDARD : Die Güter werden an eine gewöhnliche Postadresse geliefert. PACKSTATION : Die Güter werden an eine Packstation geliefert. STORE_PICKUP : Die Güter werden in der Filiale abgeholt. Dieses Feld enthält bei Express-Checkouts immer den Wert STANDARD und wird nicht anhand der gewählten Lieferoption aktualisiert.
shippingAddress	ShippingAddress	Vorhanden, falls bei Anlage übergeben.	Die Lieferanschrift des Empfängers. Diese Adresse wird dem Kunden nach dem Login im giropay-System zur Kontrolle angezeigt.
merchantCustomerNumber	String	Vorhanden, falls bei Anlage übergeben.	Händler-interne Kundennummer des Käufers. Wird dem Kunden in der Transaktionsübersicht angezeigt. Wird dem Händler in der Transaktionsübersicht angezeigt.

Feld	Typ	Eigenschaften	Beschreibung
<code>merchantOrderReferenceNumber</code>	String	Immer vorhanden.	Händler-interne, eindeutige Bestellnummer für diesen Kaufvorgang. Wird dem Kunden in der Transaktionsübersicht angezeigt. Wird dem Händler in der Transaktionsübersicht angezeigt. Die Bestellnummer wird in der Händler-Lastschrift als <code>instructionId</code> und im Verwendungszweck bei Händler und Käufer verwendet.
<code>merchantInvoiceReferenceNumber</code>	String	Vorhanden, falls bei Anlage übergeben.	Händler-interne, eindeutige Rechnungsnummer für diesen Kauf bzw. Zahlvorgang. Wird dem Händler in der Transaktionsübersicht angezeigt.
<code>merchantReconciliationReferenceNumber</code>	String	Vorhanden, falls bei Anlage übergeben.	Reconciliation-Nummer. Diese wird bei Direct-Sales in den Verwendungszweck der Händlerzahlung eingefügt. Bei Vorbestellungen wird stattdessen die bei Anlage der Captures gesetzte <code>merchantReconciliationReferenceNumber</code> verwendet.
<code>redirectUrlAfterSuccess</code>	String	Immer vorhanden, außer bei oneKlick-Checkouts.	Die Rücksprung-Adresse des Webshops (inkl. Referenz auf die Bestellung), die nach erfolgreicher Bezahlung aufgerufen wird. Siehe Erfolgsstatus eines Checkouts .
<code>redirectUrlAfterCancellation</code>	String	Immer vorhanden, außer bei oneKlick-Checkouts.	Die Rücksprung-Adresse des Webshops (inkl. Referenz auf die Bestellung), die im Falle eines Abbruchs oder technischen Fehlers aufgerufen wird. Damit wird signalisiert, dass der Kaufvorgang grundsätzlich fortgeführt werden kann und im Anschluss ein weiterer, neuer Checkout initiiert werden kann, beispielsweise, weil der Kunde noch einen Artikel in der Bestellung hinzufügen möchte. Siehe Erfolgsstatus eines Checkouts .

Feld	Typ	Eigenschaften	Beschreibung
<code>redirectUrlAfterRejection</code>	String	Immer vorhanden, außer bei oneKlick-Checkouts.	Die Rücksprung-Adresse des Webshops (inkl. Referenz auf die Bestellung), die im Falle einer Abweisung der Zahlung aufgerufen wird. Damit wird signalisiert, dass keine Zahlung autorisiert wurde (z. B. aufgrund falscher TAN-Eingabe, fehlender Bank-Autorisierung oder Betrugsverdacht). Falls das Webshop-System keine Unterscheidung zwischen <code>redirectUrlAfterCancellation</code> und <code>redirectUrlAfterRejection</code> unterstützt, kann in beiden Feldern die gleiche URL angegeben werden. Siehe Erfolgsstatus eines Checkouts .
<code>callbackUrlStatusUpdates</code>	String	Vorhanden, falls bei Anlage übergeben.	URL des Endpoints, der Mitteilungen zu Statusänderungen des Checkouts empfangen soll.
<code>minimumAge</code>	Number	Vorhanden, falls bei Anlage übergeben.	Das Mindestalter (in Jahren), das der Käufer erreicht haben muss, um die Bestellung ausführen zu dürfen, beispielsweise, weil sie Artikel enthält, die einer Altersbeschränkung unterliegen (Filme, Computerspiele, ...). Die Prüfung erfolgt direkt nach dem Login des Käufers in das giropay-System. Bei erfolgreicher Verifikation wird der Ablauf ohne weitere Meldung wie gewohnt fortgesetzt. Bei nicht-erfolgreicher Verifikation, d. h. der Kunde hat das erforderliche Mindestalter noch nicht erreicht, erfolgt eine Umleitung auf den Link <code>redirectUrlAfterAgeVerificationFailure</code> (siehe unten). Bei Nicht-Angabe dieses Wertes erfolgt keine Altersverifikation.

Feld	Typ	Eigenschaften	Beschreibung
<code>redirectUrlAfterAgeVerificationFailure</code>	String	Vorhanden, falls bei Anlage übergeben.	Die Rücksprung-Adresse des Webshops (inkl. Referenz auf die Bestellung), die im Falle einer nicht erfolgreichen Altersverifikation aufgerufen wird. Damit wird signalisiert, dass die Bestellung abgebrochen wurde, da der Käufer das erforderliche Mindestalter für mindestens einen Artikel in der Bestellung noch nicht erreicht hat. Dieses Feld wird im Falle einer geforderten Altersverifikation, d. h. bei Angabe des Feldes <code>minimumAge</code> , zum Pflichtfeld. Ist keine Altersverifikation erforderlich, ist dieses Feld wegzulassen.
<code>note</code>	String	Vorhanden, falls bei Anlage übergeben.	Freitextfeld, das auf dem Kontoauszug im Feld Verwendungszweck erscheint (nur <code>DIRECT_SALE</code>).
<code>deliveryInformation</code>	Object	Optional.	Enthält Informationen über die Lieferung.
<code>deliveryInformation.expectedShippingDate</code>	String	Optional.	Erwartetes Versanddatum.
<code>deliveryInformation.logisticsProvider</code>	String	Optional.	Paket-Dienstleister.
<code>deliveryInformation.trackingNumber</code>	String	Optional.	Sendungsnummer.
<code>customerAuthorizationDemand</code>	Object	Optional.	Gibt an, dass dem Käufer im Rahmen des Checkout-Vorgangs die Erteilung einer oneKlick-Autorisierung angeboten wurde.
<code>customerAuthorizationDemand.scopes</code>	Array	Immer vorhanden.	Die vom Käufer gewährten Rechte.
<code>customerAuthorizationDemand.thirdPartyId</code>	String	Immer vorhanden.	Id des begünstigten oneKlick-Anbieters. Wenn Händler oneKlick selbst integriert haben, entspricht sie der Händler-Id.
<code>customerAuthorizationDemand.thirdPartyClientId</code>	String	Immer vorhanden.	Id der begünstigten oneKlick-Anwendung. Wenn Händler oneKlick selbst integriert haben, entspricht sie der Händlershop-Id.

Feld	Typ	Eigenschaften	Beschreibung
<code>customerAuthorizationDemand.thirdPartyClientName</code>	String	Immer vorhanden.	Name der begünstigten oneKlick-Anwendung. Wenn Händler oneKlick selbst integriert haben, entspricht er dem Händlershop-Namen.
<code>customerAuthorizationDemand.authorizationPreapproved</code>	Boolean	Immer vorhanden.	Ist <code>true</code> , wenn der Käufer bereits vor dem Einloggen der oneKlick-Autorisierung zugestimmt hat.
<code>customerAuthorizationDemand.callbackUrlStatusUpdates</code>	Boolean	Vorhanden, falls im <code>customerAuthorizationRequest</code> übergeben wurde.	URL des Endpoints, der Mitteilungen zu Statusänderungen der oneKlick-Autorisierung empfangen soll.
<code>preauthorizationValidity</code>	String	Optional.	Garantiezeitraum für die gesicherte Vorbestellung (ORDER_SECURED). Captures innerhalb dieses Zeitraums sind für den Händler garantiert.
<code>thirdPartyId</code>	String	Optional. UUID aus 36 Zeichen.	ID des oneKlick-Anbieters, der diesen Checkout technisch angelegt hat.
<code>thirdPartyClientId</code>	String	Optional. UUID aus 36 Zeichen.	ID der oneKlick-Anwendung.
<code>thirdPartyClientName</code>	String	Optional. UUID aus 36 Zeichen.	Name der oneKlick-Anwendung.
<code>thirdPartyMerchantAuthorizationId</code>	String	Optional. UUID aus 36 Zeichen.	ID der Händlerautorisierung.
<code>_links</code>	Object		Links zu Ressourcen und verfügbaren Aktionen des Checkouts.
<code>_embedded</code>	Object		Embedded Ressourcen

Links

Relation	Description
<code>self</code>	Link zu dieser Ressource.

Relation	Description
approve	Link zur giropay-Webseite, zu dem der Käufer per HTTP Code 302 weitergeleitet werden muss, um die Zahlung zu bestätigen. Die Weiterleitung darf nicht über ein HTML Formular (unter Nutzung des <code>approve</code> Links im <code>action</code> Attribute des <code>form</code> Tags) erfolgen. Die Bestätigung durch den Kunden muss innerhalb von 30 Minuten erfolgen. Dieser Link ist nur vorhanden, wenn der Checkout vom Kunden noch nicht autorisiert wurde. Der Aufbau bzw. die Parameter des Links können sich aufgrund von internen Weiterentwicklungen verändern. Verwenden Sie daher ausschließlich den von unserem System zur Laufzeit zurückgelieferten Link.
close	Link zum Endpoint, über den die Order geschlossen werden kann. Dieser Link ist nur vorhanden, wenn es sich um einen Checkout vom Typ <code>order</code> handelt und er im Status <code>APPROVED</code> ist.
captures	Link zu Captures . Dieser Link ist nur vorhanden, wenn für diese Bestellung ein Capture möglich ist.
refunds	Link zu Refunds . Dieser Link ist nur vorhanden, wenn für diese Bestellung eine Rückzahlung möglich ist, d. h. wenn mindestens ein Capture im Status <code>SUCCESSFUL</code> vorhanden ist. Wenn der Link nicht vorhanden ist, sind keine Rückzahlungen möglich.
updateDeliveryInformation	Link zum Update der Lieferinformation. Dieser Link ist nur vorhanden, wenn die Update Frist für diesen Checkout noch nicht überschritten ist.
updateMerchantInvoiceReferenceNumber	Link zum Update der Rechnungsreferenznummer. Dieser Link ist nur vorhanden, wenn die Update Frist für diesen Checkout noch nicht überschritten ist.
customerAuthorization	Link zur im Rahmen des Checkouts erteilten Käuferautorisierung. Verweist auf den in Abfrage Käuferautorisierung beschriebenen Endpunkt. Dieser Link ist nur vorhanden, wenn die angeforderte Autorisierung vom Käufer erteilt wurde.

Embedded Resources

Resource	Beschreibung
<code>captures</code>	Array von Captures , sofern Captures zu diesem Checkout vorhanden sind.
<code>refunds</code>	Array von Refunds , sofern Refunds zu diesem Checkout vorhanden sind.

Return Codes

Status Code	Message Code	Beschreibung
200 (Ok)		Der Checkout wurde gefunden.
401 (Unauthorized)	<code>ACCESS_TOKEN_EXPIRED</code>	Siehe Resource Access .
404 (Not Found)	<code>CHECKOUT_NOT_FOUND</code>	Der angefragte Checkout existiert nicht

Beispiel (Direct Sale nach der Anlage)

```
GET /api/checkout/v1/checkouts/666dbf07-69b1-4d0c-94c1-df69e77fd609 HTTP/1.1
Content-Type: application/json;charset=utf-8
Authorization: Bearer <access_token>
```

```
HTTP/1.1 200 OK
Content-Type: application/hal+json;charset=utf-8

{
  "checkoutId" : "666dbf07-69b1-4d0c-94c1-df69e77fd609",
  "type" : "DIRECT_SALE",
  "status" : "OPEN",
  "creationTimestamp" : "2022-10-25T12:30:43.886Z",
  "totalAmount" : 100.0,
  "shippingAmount" : 3.5,
  "orderAmount" : 96.5,
  "refundLimit" : 200,
  "currency" : "EUR",
  "items" : [ {
    "quantity" : 3,
    "name" : "Bobbycar",
    "ean" : "800001303",
    "price" : 25.99
  }, {
    "quantity" : 1,
    "name" : "Helm",
    "price" : 18.53
  } ],
  "shoppingCartType" : "MIXED",
```

```

"deliveryType" : "STANDARD",
"shippingAddress" : {
  "addresseeGivenName" : "Marie",
  "addresseeLastName" : "Mustermann",
  "company" : "Musterbau GmbH & Co KG",
  "street" : "Kastanienallee",
  "streetNr" : "999",
  "additionalAddressInformation" : "Haus 4",
  "zip" : "90402",
  "city" : "Schwaig",
  "countryCode" : "DE",
  "state" : "Bayern"
},
"merchantOrderReferenceNumber" : "order-A12223412",
"merchantCustomerNumber" : "cust-732477",
"merchantInvoiceReferenceNumber" : "20150112334345",
"merchantReconciliationReferenceNumber" : "recon-A12223412",
"note" : "Ihr Einkauf bei Spielauto-Versand.",
"minimumAge" : 18,
"redirectUrlAfterSuccess" : "https://spielauto-versand.de/order/123/success",
"redirectUrlAfterCancellation" : "https://spielauto-versand.de/order/123/cancellation",
"redirectUrlAfterAgeVerificationFailure" : "https://spielauto-versand.de/order/123/ageverificationfailed",
"redirectUrlAfterRejection" : "https://spielauto-versand.de/order/123/rejection",
"callbackUrlStatusUpdates" : "https://spielauto-versand.de/callback/status",
"thirdPartyId" : "08ec0853-6316-4fb9-914c-e94cd5ed627b",
"thirdPartyClientId" : "7281895b-9bd0-4dfa-8a33-0b30e988ef17",
"thirdPartyClientName" : "Managed client of CPSP shop",
"thirdPartyMerchantAuthorizationId" : "0b828267-73a3-4936-a9f5-e541834de236",
"deliveryInformation" : {
  "expectedShippingDate" : "2016-10-19T12:00:00.000Z",
  "logisticsProvider" : "DHL",
  "trackingNumber" : "1234567890"
},
"expiryTimestamp" : "2022-10-25T13:00:43.886Z",
"customerAuthorizationDemand" : {
  "thirdPartyId" : "08ec0853-6316-4fb9-914c-e94cd5ed627b",
  "thirdPartyClientId" : "7281895b-9bd0-4dfa-8a33-0b30e988ef17",
  "thirdPartyClientName" : "Managed client of CPSP shop",
  "scopes" : [ "PERFORM_CHECKOUT" ],
  "authorizationPreapproved" : false,
  "callbackUrlStatusUpdates" : "http://one.klick/callbackUrlStatusUpdates"
},
"_links" : {
  "approve" : {
    "href" : "https://localhost/checkout/?p=666dbf07-69b1-4d0c-94c1-df69e77fd609#/checkout/666dbf07-69b1-4d0c-94c1-df69e77fd609"
  },
  "updateDeliveryInformation" : {
    "href" : "https://api.paydirekt.de/api/checkout/v1/checkouts/666dbf07-69b1-4d0c-

```

```

94c1-df69e77fd609/deliveryInformation"
  },
  "updateMerchantInvoiceReferenceNumber" : {
    "href" : "https://api.paydirekt.de/api/checkout/v1/checkouts/666dbf07-69b1-4d0c-
94c1-df69e77fd609/merchantInvoiceReferenceNumber"
  },
  "self" : {
    "href" : "https://api.paydirekt.de/api/checkout/v1/checkouts/666dbf07-69b1-4d0c-
94c1-df69e77fd609"
  }
}
}
}

```

Beispiel (Direct Sale nach Approval durch Kunden)

```

GET /api/checkout/v1/checkouts/666dbf07-69b1-4d0c-94c1-df69e77fd609 HTTP/1.1
Content-Type: application/json;charset=utf-8
Authorization: Bearer <access_token>

```

```

HTTP/1.1 200 OK
Content-Type: application/hal+json;charset=utf-8

```

```

{
  "checkoutId" : "666dbf07-69b1-4d0c-94c1-df69e77fd609",
  "type" : "DIRECT_SALE",
  "status" : "APPROVED",
  "correlationId" : "ext0815-00000152",
  "creationTimestamp" : "2022-10-25T12:30:43.886Z",
  "totalAmount" : 100.0,
  "shippingAmount" : 3.5,
  "orderAmount" : 96.5,
  "refundLimit" : 200,
  "currency" : "EUR",
  "items" : [ {
    "quantity" : 3,
    "name" : "Bobbycar",
    "ean" : "800001303",
    "price" : 25.99
  }, {
    "quantity" : 1,
    "name" : "Helm",
    "price" : 18.53
  } ],
  "shoppingCartType" : "MIXED",
  "deliveryType" : "STANDARD",
  "shippingAddress" : {
    "addresseeGivenName" : "Marie",
    "addresseeLastName" : "Mustermann",

```

```

"company" : "Musterbau GmbH & Co KG",
"street" : "Kastanienallee",
"streetNr" : "999",
"additionalAddressInformation" : "Haus 4",
"zip" : "90402",
"city" : "Schwaig",
"countryCode" : "DE",
"state" : "Bayern"
},
"merchantOrderReferenceNumber" : "order-A12223412",
"merchantCustomerNumber" : "cust-732477",
"merchantInvoiceReferenceNumber" : "20150112334345",
"merchantReconciliationReferenceNumber" : "recon-A12223412",
"note" : "Ihr Einkauf bei Spielauto-Versand.",
"minimumAge" : 18,
"redirectUrlAfterSuccess" : "https://spielauto-versand.de/order/123/success",
"redirectUrlAfterCancellation" : "https://spielauto-
versand.de/order/123/cancellation",
"redirectUrlAfterAgeVerificationFailure" : "https://spielauto-
versand.de/order/123/ageverificationfailed",
"redirectUrlAfterRejection" : "https://spielauto-versand.de/order/123/rejection",
"callbackUrlStatusUpdates" : "https://spielauto-versand.de/callback/status",
"thirdPartyId" : "08ec0853-6316-4fb9-914c-e94cd5ed627b",
"thirdPartyClientId" : "7281895b-9bd0-4dfa-8a33-0b30e988ef17",
"thirdPartyClientName" : "Managed client of CPSP shop",
"thirdPartyMerchantAuthorizationId" : "0b828267-73a3-4936-a9f5-e541834de236",
"deliveryInformation" : {
  "expectedShippingDate" : "2016-10-19T12:00:00.000Z",
  "logisticsProvider" : "DHL",
  "trackingNumber" : "1234567890"
},
"expiryTimestamp" : "2022-10-25T13:00:43.886Z",
"customerAuthorizationDemand" : {
  "thirdPartyId" : "08ec0853-6316-4fb9-914c-e94cd5ed627b",
  "thirdPartyClientId" : "7281895b-9bd0-4dfa-8a33-0b30e988ef17",
  "thirdPartyClientName" : "Managed client of CPSP shop",
  "scopes" : [ "PERFORM_CHECKOUT" ],
  "authorizationPreapproved" : false,
  "callbackUrlStatusUpdates" : "http://one.klick/callbackUrlStatusUpdates"
},
"_links" : {
  "refunds" : {
    "href" : "https://api.paydirekt.de/api/checkout/v1/checkouts/666dbf07-69b1-4d0c-
94c1-df69e77fd609/refunds"
  },
  "customerAuthorization" : {
    "href" :
"https://api.paydirekt.de/api/thirdpartycustomerauthorization/v1/authorizations/15b13f
58-d712-4d42-a894-b640b32eea81"
  },
  "updateDeliveryInformation" : {

```

```

    "href" : "https://api.paydirekt.de/api/checkout/v1/checkouts/666dbf07-69b1-4d0c-94c1-df69e77fd609/deliveryInformation"
  },
  "updateMerchantInvoiceReferenceNumber" : {
    "href" : "https://api.paydirekt.de/api/checkout/v1/checkouts/666dbf07-69b1-4d0c-94c1-df69e77fd609/merchantInvoiceReferenceNumber"
  },
  "self" : {
    "href" : "https://api.paydirekt.de/api/checkout/v1/checkouts/666dbf07-69b1-4d0c-94c1-df69e77fd609"
  }
},
"_embedded" : {
  "captures" : [ {
    "type" : "CAPTURE_DIRECT_SALE",
    "transactionId" : "607d1caf-c516-4501-b42f-59695572138d",
    "amount" : 100.0,
    "callbackUrlStatusUpdates" : "https://spielauto-versand.de/callback/status",
    "deliveryInformation" : {
      "expectedShippingDate" : "2016-10-19T12:00:00.000Z",
      "logisticsProvider" : "DHL",
      "trackingNumber" : "1234567890"
    },
    "status" : "SUCCESSFUL",
    "_links" : {
      "self" : {
        "href" : "https://api.paydirekt.de/api/checkout/v1/checkouts/666dbf07-69b1-4d0c-94c1-df69e77fd609/captures/607d1caf-c516-4501-b42f-59695572138d"
      }
    }
  } ]
}
}

```

Callback für Statusupdates

POST callbackUrlStatusUpdates

Wenn Händler/PSPs bei Anlage von Checkouts eine Callback URL `callbackUrlStatusUpdates` übergeben, sendet das giropay-System bei Statusänderungen des Checkouts an diese URL einen HTTP-Request. Bei `DIRECT_SALE` werden für den eingebetteten Capture ebenfalls `Statusupdates` versendet. Bei Verbindungsfehlern oder Responses mit einem HTTP-Statuscode von 5xx wird das giropay-System die Übermittlung der Statusänderung in einem Zeitfenster von 24 Stunden bis zu 5 Mal wiederholen. Statusänderungen werden dabei immer in der Reihenfolge übermittelt, in der sie am Checkout aufgetreten sind. Kann eine Statusänderung nicht übermittelt werden, so werden auch alle nachfolgenden Statusänderungen nicht übermittelt, bis entweder die fehlgeschlagene Übermittlung nachgeholt werden konnte oder alle wiederholten Versuche fehlgeschlagen sind.



Für die korrekte Ermittlung des Checkout Ergebnisses ist der letzte GET-Aufruf zwingend erforderlich. Siehe auch [Erfolgsstatus eines Checkouts](#).

Request

Feld	Typ	Eigenschaften	Beschreibung
<code>checkoutId</code>	String	Immer vorhanden.	Eindeutige UUID aus 36 Zeichen für diesen Checkout-Vorgang. Wird vom giropay-System vergeben.
<code>merchantOrderReferenceNumber</code>	String	Pflichtfeld. Maximal 20 Zeichen. Nur SEPA-konforme Zeichen .	Händler-interne, eindeutige Bestellnummer für diesen Kaufvorgang. Wird dem Kunden in der Transaktionsübersicht angezeigt. Wird dem Händler in der Transaktionsübersicht angezeigt. Die Bestellnummer wird in der Händler-Lastschrift als <code>instructionId</code> und im Verwendungszweck bei Händler und Käufer verwendet.

Feld	Typ	Eigenschaften	Beschreibung
checkoutStatus	String	Immer vorhanden.	<p>Status des Checkouts: OPEN, PENDING, APPROVED, REJECTED, CANCELED, CLOSED oder EXPIRED.</p> <p>OPEN: Der Checkout wurde neu angelegt und vom Benutzer noch nicht bestätigt.</p> <p>PENDING: Der Checkout wurde vom Kunden bestätigt, aber noch nicht per <i>execute</i> aktiviert (derzeit nur <i>giropay Express</i>)</p> <p>APPROVED: Der Checkout wurde vom Kunden bestätigt und vom System genehmigt. Im Falle einer ORDER können jetzt Captures angelegt werden.</p> <p>REJECTED: Der Checkout wurde vom System abgelehnt. Die Zahlung war nicht erfolgreich. Es ist nicht zu erwarten, dass die Zahlung bei erneutem Versuch mit diesem Benutzer erfolgreich sein wird.</p> <p>CANCELED: Der Checkout wurde vom Kunden abgebrochen oder es kam zu einem technischen Fehler. Der Checkout kann nicht weiter bearbeitet werden.</p> <p>CLOSED: Nur ORDER: Die Bestellung ist geschlossen, es können keine weiteren Captures durchgeführt werden. Refunds sind weiter möglich.</p> <p>EXPIRED: Der Checkout ist abgelaufen und kann nicht weiter verwendet werden. Der Benutzer hat den Checkout nicht innerhalb von 30 Minuten bestätigt.</p>
statusUpdateTimeStamp	String	Immer vorhanden.	Zeitstempel der Statusänderung.
sequenceNumber	Number	Immer vorhanden.	Sequenznummer der Statusänderung. Anhand der Sequenznummer kann ein Shop die Reihenfolge der Statusänderungen nachvollziehen und gegebenenfalls fehlende Statusänderungen erkennen.

Beispiel

```
POST /webhooktest/v1/checkout/status HTTP/1.1
Content-Type: application/json;charset=utf-8

{
  "checkoutId" : "a5847179-cc74-4190-bc09-80e7bdc5976e",
  "checkoutStatus" : "APPROVED",
  "merchantOrderReferenceNumber" : "order-A12223412",
  "statusUpdateTimestamp" : "2020-01-01T12:00:00.000Z",
  "sequenceNumber" : 4
}
```

Response

Die Response enthält keinen Body und wird vom giropay-System nicht weiter verarbeitet. Der HTTP-Statuscode sollte dennoch für etwaige Serviceanfragen Rückschlüsse auf die Verarbeitung der Status-Benachrichtigung zulassen.

Beispiel

```
HTTP/1.1 200 OK
```

Aktualisierung der Versandbedingungen

```
PUT /api/checkout/v1/checkouts/{checkoutId}/deliveryInformation
```

Über diesen Endpoint können die Versandbedingungen eines bereits angelegten Checkouts aktualisiert werden. Alle im Request angegebenen Felder werden an der Checkout-Ressource überschrieben. Bei allen nicht im Request enthaltenen Feldern bleibt der Wert in der Ressource unangetastet. Die aktuellen Versandbedingungen des Checkouts sind für den Käufer sichtbar. Aktualisierungen sind nur bis 25 Tage nach dem letzten Capture auf den Checkout gestattet.

Request

Feld	Typ	Eigenschaften	Beschreibung
<code>expectedShippingDate</code>	String	Optional. Zeitstempel im ISO-8601 -Format.	Erwartetes Versanddatum.
<code>logisticsProvider</code>	String	Optional.	Paket-Dienstleister.
<code>trackingNumber</code>	String	Optional.	Sendungsnummer.

Response

In der Response wird der aktualisierte Checkout zurückgegeben. Die Struktur ist identisch mit dem Checkout Endpoint [GET](#)-Aufruf.

Return Codes

Status Code	Message Code	Beschreibung
200 (Ok)		Der Checkout wurde gefunden.
401 (Unauthorized)	ACCESS_TOKEN_EXPIRED	Siehe Resource Access .
404 (Not Found)	CHECKOUT_NOT_FOUND	Der zu aktualisierende Checkout existiert nicht.
422 (Unprocessable Entity)	CHECKOUT_UPDATE_TIMEFRAME_EXPIRED	Der Checkout kann nicht mehr aktualisiert werden, da mehr als 25 Tage seit dem letzten Capture verstrichen sind.

Beispielrequest

```
PUT /api/checkout/v1/checkouts/c9f472cc-9739-4356-89b3-cc8e83639571/deliveryInformation HTTP/1.1
Content-Type: application/json;charset=utf-8
Authorization: Bearer <access_token>
```

```
{
  "expectedShippingDate" : "2016-10-19T12:00:00Z",
  "logisticsProvider" : "DHL",
  "trackingNumber" : "1234567890"
}
```

Aktualisierung der Rechnungsreferenz

```
PUT /api/checkout/v1/checkouts/{checkoutId}/merchantInvoiceReferenceNumber
```

Über diesen Endpoint kann die Rechnungsreferenz eines bereits angelegten Checkouts aktualisiert werden. Die aktuelle Rechnungsreferenz des Checkouts ist für Käufer nicht einsehbar. Aktualisierungen sind nur bis 25 Tage nach dem letzten Capture auf den Checkout gestattet.

Request

Feld	Typ	Eigenschaften	Beschreibung
merchantInvoiceReferenceNumber	String	Optional. Maximal 100 Zeichen.	Händler-interne, eindeutige Rechnungsnummer für diesen Kauf- bzw. Zahlvorgang. Wird dem Händler in der Transaktionsübersicht angezeigt.

Response

In der Response wird der aktualisierte Checkout zurückgegeben. Die Struktur ist identisch mit dem Checkout Endpoint [GET](#)-Aufruf.

Return Codes

Status Code	Message Code	Beschreibung
200 (Ok)		Der Checkout wurde gefunden.
401 (Unauthorized)	ACCESS_TOKEN_EXPIRED	Siehe Resource Access .
404 (Not Found)	CHECKOUT_NOT_FOUND	Der zu aktualisierende Checkout existiert nicht.
422 (Unprocessable Entity)	CHECKOUT_UPDATE_TIMEFRAME_EXPIRED	Der Checkout kann nicht mehr aktualisiert werden, da mehr als 25 Tage seit dem letzten Capture verstrichen sind.

Beispielrequest

```
PUT /api/checkout/v1/checkouts/65a626ab-5e3b-4582-b6b4-6379434850c2/merchantInvoiceReferenceNumber HTTP/1.1
Content-Type: application/json;charset=utf-8
Authorization: Bearer <access_token>
```

```
{
  "merchantInvoiceReferenceNumber" : "12345678"
}
```

Capture

Durch ein Capture wird das Konto des Käufers um einen bestimmten Betrag belastet und dem Händler eine Zahlungsgarantie ausgesprochen.

Jedes Capture wird von der Käuferbank autorisiert. Eine erfolgreiche Bank-Autorisierung führt zu einer Zahlungsgarantie über den angefragten Betrag und einer entsprechenden Buchung auf dem Konto. Dies führt je nach Ergebnis zu einem finalen Status **SUCCESSFUL** oder **REJECTED**.

Es gibt folgende Typen von Captures:

CAPTURE_DIRECT_SALE

Wird automatisch erzeugt, wenn der Käufer einen Checkout vom Typ **DIRECT_SALE** bestätigt und die Autorisierung durch die Käuferbank erfolgreich war.

CAPTURE_ORDER

Nach erfolgreicher Anlage eines Checkouts vom Typ **ORDER** kann der Händler **bis maximal 182 Tage nach Erstellung der Order** einen oder mehrere Capture(s) vom Typ **CAPTURE_ORDER** erzeugen. Währenddessen ist die **Anzahl der Captures unbegrenzt**. Die maximale Summe aller Captures ist **auf den Checkout-Betrag (totalAmount) begrenzt**. Der letzte Capture sollte mit einem Flag **finalCapture** versehen werden, um die Bestellung für den Kunden zu schließen.

CAPTURE_ORDER_SECURED

Zu einer **ORDER_SECURED** (gesicherte Vorbestellung oder Bestellung mit Teilzahlungen) können direkt im Anschluss oder später ein oder mehrere Captures angestoßen werden. Für eine gesicherte Vorbestellung wird eine Zahlungsgarantie für den gewählten Zeitraum (maximal 15 Kalendertage) an den Händler ausgesprochen. Captures (Teilzahlungen) werden innerhalb des Garantiezeitraums immer ausgeführt.

Eine Bestellung vom Typ **ORDER** oder **ORDER_SECURED** wird dem Kunden in der Transaktionsübersicht so lange als offen angezeigt, bis

- der volle Betrag der Order abgerufen wurde und die Order dadurch automatisch geschlossen wird, oder
- ein Capture mit **finalCapture** markiert wurde, oder
- die Order explizit geschlossen wurde, oder
- 182 Tage abgelaufen sind.

Falls die genaue Höhe des Bestellbetrags bei Anlage der Order noch nicht feststeht, kann

- die Overcapture-Funktionalität verwendet werden (siehe entsprechende Feldbeschreibung in [Checkout - Anlage](#)), oder
- eine entsprechende Warenkorbposition verwendet werden, um den Checkout-Betrag zu erhöhen, oder
- ein höherer Checkout-Betrag (**totalAmount**) angegeben werden (ohne separate Warenkorbposition).

In jedem Fall soll der Händler den Kunden auf diesen Sachverhalt bereits im Shop hinweisen. Der Käufer autorisiert immer den gesamthaften Checkout-Betrag inklusive einer möglichen Erhöhung.

Anlage

POST /api/checkout/v1/checkouts/{checkoutId}/captures



Der Endpoint zur Anlage eines Captures ist als **captures**-Link in der **Checkout**-Ressource enthalten, sofern es sich um eine **ORDER** handelt und die Anlage weiterer Captures noch möglich ist. Es sollte immer der Link aus der **Checkout**-Ressource verwendet werden.

Es wird ein neuer Capture angestoßen, um einen Betrag abzurufen. Der Capture ist nach der Generierung zunächst im Status **PENDING** bis eine Prüfung durch die Käuferbank erfolgt ist, was in der Regel weniger als eine Sekunde dauert.

Request

Feld	Typ	Eigenschaften	Beschreibung
amount	Number	Pflichtfeld. Zwischen 0.01 und 50000. Maximal zwei Nachkommastellen.	Der Betrag der Zahlung in Währung der ursprünglichen Order. Der Maximalbetrag liegt bei 50 000,00 EUR.
finalCapture	Boolean	Optional. Flag muss bei dem letzten Capture eines Checkouts mit aktiviertem overcapture -Flag true sein.	Gibt an, ob durch diesen Capture der letzte Betrag abgerufen wurde, insbesondere dann, wenn der Betrag dieser Zahlung geringer ist als der Gesamtbetrag der zugrunde liegenden Order. Die Order gilt damit als vollständig bezahlt und wird dem Kunden entsprechend als geschlossen angezeigt. Weitere Captures sind für diese Order nicht mehr möglich.
note	String	Optional. Maximal 37 Zeichen.	Ein Freitext für den Kunden. Der Freitext wird dem Kunden für diese Zahlung im Verwendungszweck angezeigt.
merchantCaptureReferenceNumber	String	Optional. Maximal 30 Zeichen.	Händler-interne Referenznummer für diesen Capture-Vorgang.

Feld	Typ	Eigenschaften	Beschreibung
<code>merchantReconciliationReferenceNumber</code>	String	Optional. Maximal 30 Zeichen.	Reconciliation-Nummer. Diese wird in den Verwendungszweck der Händlerzahlung eingefügt.
<code>captureInvoiceReferenceNumber</code>	String	Optional. Maximal 100 Zeichen.	Händler-interne, eindeutige Rechnungsnummer für diesen Capture-Vorgang.
<code>callbackUrlStatusUpdates</code>	String	Optional. Maximal 2000 Zeichen.	URL des Endpoints, der Mitteilungen zu Statusänderungen des Captures empfangen soll.
<code>deliveryInformation</code>	Object	Optional.	Enthält Informationen über die Lieferung.
<code>deliveryInformation.expectedShippingDate</code>	String	Optional.	Erwartetes Versanddatum.
<code>deliveryInformation.logisticsProvider</code>	String	Optional.	Paket-Dienstleister.
<code>deliveryInformation.trackingNumber</code>	String	Optional.	Paketnummer.

Response

In der Response wird der angelegte Capture zurückgegeben. Die Struktur ist identisch mit dem Capture Endpoint [GET](#)-Aufruf.

Im HTTP Header Feld `Location` ist die URL der angelegten Capture-Ressource, inkl. Transaction-ID enthalten. Die Transaction-ID wird im Response Body ebenfalls zurückgegeben.

Return Codes

Status Code	Message Code	Beschreibung
201 (Created)		Der Capture wurde erfolgreich angelegt.
400 (Bad Request)	<code>VALIDATION_ERROR</code>	Der Request ist syntaktisch ungültig.
401 (Unauthorized)	<code>ACCESS_TOKEN_EXPIRED</code>	Siehe Resource Access .
401 (Unauthorized)	<code>UNAUTHORIZED</code>	Der Zugriff wurde verweigert. Bitte wenden Sie sich an den giropay-Support.
404 (Not Found)	<code>CHECKOUT_NOT_FOUND</code>	Der Checkout wurde nicht gefunden.
422 (Unprocessable Entity)	<code>CAPTURE_AMOUNT_EXCEEDED</code>	Die Summe aller Captures übersteigt den Gesamtbetrag des Checkouts.
422 (Unprocessable Entity)	<code>CAPTURE_CHECKOUT_WRONG_TYPE</code>	Für diesen Checkout-Typ kann kein Capture angelegt werden.

Status Code	Message Code	Beschreibung
422 (Unprocessable Entity)	CAPTURE_ORDER_CLOSED	Es wurde versucht, ein Capture für einen bereits geschlossenen Checkout anzulegen.
422 (Unprocessable Entity)	CAPTURE_ORDER_NOT_APPROVED	Es wurde versucht, einen Capture für eine Order anzulegen, die vom Kunden nicht bestätigt wurde.
422 (Unprocessable Entity)	CAPTURE_NOT_AUTHORIZED	Der Capture-Betrag wurde von der Käuferbank nicht autorisiert.
422 (Unprocessable Entity)	MERCHANT_BANKACCOUNT_LOCKED	Ihr Händlerkonto ist gesperrt. Bitte wenden Sie sich an den giropay-Support.
422 (Unprocessable Entity)	CHECKOUT_REJECTED	Der Checkout wurde abgelehnt. Es kann kein Capture angelegt werden.
422 (Unprocessable Entity)	ACCOUNT_DEBOARDED	Es wurde versucht, einen Capture für einen Kunden anzulegen, welcher nicht mehr bei giropay teilnimmt.

Beispiel

```
POST /api/checkout/v1/checkouts/a20e18a4-9e99-4988-b156-486ce3948014/captures HTTP/1.1
Content-Type: application/json;charset=utf-8
Authorization: Bearer <access_token>
```

```
{
  "amount" : 10,
  "finalCapture" : false,
  "note" : "Thanks for shopping.",
  "merchantCaptureReferenceNumber" : "capture-21323",
  "merchantReconciliationReferenceNumber" : "recon-1234",
  "captureInvoiceReferenceNumber" : "invoice-1234",
  "callbackUrlStatusUpdates" : "https://spielauto-versand.de/callback/status",
  "deliveryInformation" : {
    "expectedShippingDate" : "2016-10-19T12:00:00.000Z",
    "logisticsProvider" : "DHL",
    "trackingNumber" : "1234567890"
  }
}
```

HTTP/1.1 201 Created

Location: <https://api.paydirekt.de/api/checkout/v1/checkouts/a20e18a4-9e99-4988-b156-486ce3948014/captures/3aef8297-d754-42ee-92f7-ec17258e23ed>

Content-Type: application/hal+json;charset=utf-8

```
{
  "type" : "CAPTURE_ORDER",
  "transactionId" : "3aef8297-d754-42ee-92f7-ec17258e23ed",
  "amount" : 10,
  "merchantReconciliationReferenceNumber" : "recon-1234",
  "finalCapture" : false,
  "merchantCaptureReferenceNumber" : "capture-21323",
  "captureInvoiceReferenceNumber" : "invoice-1234",
  "callbackUrlStatusUpdates" : "https://spielauto-versand.de/callback/status",
  "deliveryInformation" : {
    "expectedShippingDate" : "2016-10-19T12:00:00.000Z",
    "logisticsProvider" : "DHL",
    "trackingNumber" : "1234567890"
  },
  "status" : "SUCCESSFUL",
  "_links" : {
    "self" : {
      "href" : "https://api.paydirekt.de/api/checkout/v1/checkouts/a20e18a4-9e99-4988-b156-486ce3948014/captures/3aef8297-d754-42ee-92f7-ec17258e23ed"
    }
  }
}
```

Abfrage

```
GET /api/checkout/v1/checkouts/{checkoutId}/captures/{captureId}
```

Die Abfrage liefert die Daten eines spezifischen Captures. Die Capture-Transaktionen sind in der Checkout-Resource eingebettet. Eine explizite Abfrage einer Capture-Transaktion ist im Allgemeinen nicht notwendig.

Response

Feld	Typ	Eigenschaften	Beschreibung
<code>type</code>	String	Immer vorhanden.	Typ der Capture-Transaktion. Entweder <code>CAPTURE_DIRECT_SALE</code> oder <code>CAPTURE_ORDER</code> . Der Wert wird automatisch durch den Typ des Checkouts ermittelt.

Feld	Typ	Eigenschaften	Beschreibung
<code>transactionId</code>	String	Immer vorhanden.	Eindeutige Transaktions-ID dieses Captures (UUID aus 36 Zeichen). Der Wert wird durch das giro-pay-System vergeben.
<code>amount</code>	Number	Immer vorhanden.	Der Betrag der Zahlung in Wahrung der ursprunglichen Order. Der Maximalbetrag liegt bei 50 000,00 EUR.
<code>finalCapture</code>	Boolean	Vorhanden, falls bei Anlage ubergeben.	Gibt an, ob durch diesen Capture der letzte Betrag abgerufen wurde, insbesondere dann, wenn der Betrag dieser Zahlung geringer ist als der Gesamtbetrag der zugrunde liegenden Order. Die Order gilt damit als vollstandig bezahlt und wird dem Kunden entsprechend als geschlossen angezeigt. Weitere Captures sind fur diese Order nicht mehr moglich.
<code>merchantCaptureReferenceNumber</code>	String	Vorhanden, falls bei Anlage ubergeben.	Handler-interne Referenznummer fur diesen Capture-Vorgang.
<code>merchantReconciliationReferenceNumber</code>	String	Vorhanden, falls bei Anlage ubergeben.	Reconciliation-Nummer. Diese wird in den Verwendungszweck der Handlerzahlung eingefugt.
<code>captureInvoiceReferenceNumber</code>	String	Vorhanden, falls bei Anlage ubergeben.	Handler-interne, eindeutige Rechnungsnummer fur diesen Capture-Vorgang.
<code>callbackUrlStatusUpdates</code>	String	Vorhanden, falls bei Anlage ubergeben.	URL des Endpoints, der Mitteilungen zu Statusanderungen des Checkouts empfangen soll.
<code>paymentInformationId</code>	String	Ist erst vorhanden, wenn die Transaktion im Zahlungsverkehr verarbeitet wurde.	PaymentInformation-Id, die zu dieser Transaktion gehort. Sie dient der Zuordnung einzelner Transaktionen zu einer Sammelbuchung. UUID aus 36 Zeichen.
<code>status</code>	String	Immer vorhanden.	Gibt an, ob die Autorisierung durch die Bank noch aussteht (PENDING), erfolgreich war (SUCCESSFUL) oder abgelehnt wurde (REJECTED). Im Falle von SUCCESSFUL ist die Zahlungsgarantie fur diese Transaktion gegeben.

Feld	Typ	Eigenschaften	Beschreibung
<code>deliveryInformation</code>	Object	Vorhanden, falls bei Anlage übergeben.	Enthält Informationen über die Lieferung.
<code>deliveryInformation.expectedShippingDate</code>	String	Vorhanden, falls bei Anlage übergeben.	Erwartetes Versanddatum.
<code>deliveryInformation.logisticsProvider</code>	String	Vorhanden, falls bei Anlage übergeben.	Paket-Dienstleister.
<code>deliveryInformation.trackingNumber</code>	String	Vorhanden, falls bei Anlage übergeben.	Paketnummer.
<code>_links</code>	Object		Links zu Ressourcen und verfügbaren Aktionen.

Return Codes

Status Code	Message Code	Beschreibung
200 (Ok)		Der Capture wurde erfolgreich gesendet.
401 (Unauthorized)	<code>ACCESS_TOKEN_EXPIRED</code>	Siehe Resource Access .
404 (Not Found)	<code>CHECKOUT_NOT_FOUND</code>	Der Checkout wurde nicht gefunden.
404 (Not Found)	<code>TRANSACTION_NOT_FOUND</code>	Die Transaktion wurde nicht gefunden.

Beispiel

```
GET /api/checkout/v1/checkouts/a20e18a4-9e99-4988-b156-486ce3948014/captures/3aef8297-
d754-42ee-92f7-ec17258e23ed HTTP/1.1
Content-Type: application/json;charset=utf-8
Authorization: Bearer <access_token>
```

HTTP/1.1 200 OK

Content-Type: application/hal+json;charset=utf-8

```
{
  "type" : "CAPTURE_ORDER",
  "transactionId" : "3aef8297-d754-42ee-92f7-ec17258e23ed",
  "amount" : 10,
  "merchantReconciliationReferenceNumber" : "recon-1234",
  "finalCapture" : false,
  "merchantCaptureReferenceNumber" : "capture-21323",
  "captureInvoiceReferenceNumber" : "invoice-1234",
  "callbackUrlStatusUpdates" : "https://spielauto-versand.de/callback/status",
  "deliveryInformation" : {
    "expectedShippingDate" : "2016-10-19T12:00:00.000Z",
    "logisticsProvider" : "DHL",
    "trackingNumber" : "1234567890"
  },
  "status" : "SUCCESSFUL",
  "paymentInformationId" : "0000000-1111-2222-3333-444444444444",
  "_links" : {
    "self" : {
      "href" : "https://api.paydirekt.de/api/checkout/v1/checkouts/a20e18a4-9e99-4988-b156-486ce3948014/captures/3aef8297-d754-42ee-92f7-ec17258e23ed"
    }
  }
}
```

Callback für Statusupdates

Händler/PSPs können bei Anlage des Captures eine Callback URL `callbackUrlStatusUpdates` übergeben. Das giropay-System sendet bei Statusänderungen des Captures an diese einen HTTP-Request. Bei Verbindungsfehlern oder Responses mit einem HTTP-Statuscode von 5xx wird das giropay-System die Übermittlung der Statusänderung in einem Zeitfenster von 24 Stunden bis zu 5 Mal wiederholen. Statusänderungen werden dabei immer in der Reihenfolge übermittelt, in der sie am Checkout aufgetreten sind. Kann eine Statusänderung nicht übermittelt werden, so werden auch alle nachfolgenden Statusänderungen nicht übermittelt, bis entweder die fehlgeschlagene Übermittlung nachgeholt werden konnte oder alle wiederholten Versuche fehlgeschlagen sind.

POST `callbackUrlStatusUpdates`



Für die korrekte Ermittlung des Checkout Ergebnisses ist der letzte GET-Aufruf zwingend erforderlich. Siehe auch [Erfolgsstatus eines Checkouts](#).

Request

Feld	Typ	Eigenschaften	Beschreibung
<code>checkoutId</code>	String	Immer vorhanden.	Eindeutige UUID aus 36 Zeichen für diesen Checkout-Vorgang. Wird vom giropay-System vergeben.
<code>merchantOrderReferenceNumber</code>	String	Pflichtfeld. Maximal 20 Zeichen. Nur SEPA-konforme Zeichen .	Händler-interne, eindeutige Bestellnummer für diesen Kaufvorgang. Wird dem Kunden in der Transaktionsübersicht angezeigt. Wird dem Händler in der Transaktionsübersicht angezeigt. Die Bestellnummer wird in der Händler-Lastschrift als <code>instructionId</code> und im Verwendungszweck bei Händler und Käufer verwendet.
<code>transactionId</code>	String	Immer vorhanden.	Eindeutige Transaktions-ID dieses Captures (UUID aus 36 Zeichen). Der Wert wird durch das giropay-System vergeben.
<code>merchantCaptureReferenceNumber</code>	String	Optional. Maximal 30 Zeichen.	Händler-interne Referenznummer für diesen Capture-Vorgang.
<code>captureStatus</code>	String	Immer vorhanden.	Gibt an, ob die Autorisierung durch die Bank noch aussteht (<code>PENDING</code>), erfolgreich war (<code>SUCCESSFUL</code>) oder abgelehnt wurde (<code>REJECTED</code>). Im Falle von <code>SUCCESSFUL</code> ist die Zahlungsgarantie für diese Transaktion gegeben.
<code>statusUpdateTimeStamp</code>	String	Immer vorhanden.	Zeitstempel der Statusänderung.
<code>sequenceNumber</code>	Number	Immer vorhanden.	Sequenznummer der Statusänderung. Anhand der Sequenznummer kann ein Shop die Reihenfolge der Statusänderungen nachvollziehen und gegebenenfalls fehlende Statusänderungen erkennen.

Beispiel

```
POST /webhooktest/v1/capture/status HTTP/1.1
Content-Type: application/json;charset=utf-8
```

```
{
  "checkoutId" : "9a5c5bfd-26a4-46d8-826e-7756344368cc",
  "merchantOrderReferenceNumber" : "order-A12223412",
  "transactionId" : "94f958b5-a1e9-4e39-9215-faeab3f7bd49",
  "captureStatus" : "SUCCESSFUL",
  "merchantCaptureReferenceNumber" : "9557ea4a-ff13-4631-b9f8-6e4e5770a7d1",
  "statusUpdateTimestamp" : "2020-01-01T12:00:00.000Z",
  "sequenceNumber" : 2
}
```

Response

Die Response enthält keinen Body und wird vom giropay-System nicht weiter verarbeitet. Der HTTP-Statuscode sollte dennoch für etwaige Serviceanfragen Rückschlüsse auf die Verarbeitung der Status-Benachrichtigung zulassen.

Beispiel

```
HTTP/1.1 200 OK
```

Order (Secured) schließen

```
POST /api/checkout/v1/checkouts/{checkoutId}/close
```

Bestellungen vom Typ **ORDER** als auch **ORDER_SECURED** können und sollen im giropay-System geschlossen werden, wenn für die Bestellung kein weiterer **CAPTURE** durchgeführt werden wird. Damit wird die Bestellung dem Kunden nicht weiter als *offen* angezeigt.



Bestellungen sollten sobald wie möglich geschlossen werden, damit bspw. im Fall von **ORDER_SECURED** reservierte Restbeträge freigegeben werden können.

Bestellungen können nur geschlossen werden, wenn sie vorher vom Kunden durch seinen Login bestätigt worden sind.

Request

Die Anfrage benötigt keinen Request Body.

Response

In der Response wird der angelegte Checkout zurückgegeben. Die Struktur ist identisch mit dem Checkout Endpoint [GET-Aufruf des Checkouts](#).

Return Codes

Status Code	Message Code	Beschreibung
200 (OK)		Die Bestellung wurde erfolgreich geschlossen.
401 (Unauthorized)	ACCESS_TOKEN_EXPIRED	Siehe Resource Access .
422 (Unprocessable Entity)	ORDER_NOT_APPROVED	Die Bestellung ist vom Kunden nicht bestätigt worden. Unbestätigte Bestellungen können nicht geschlossen werden.
422 (Unprocessable Entity)	ORDER_ALREADY_CLOSED	Die Bestellung ist bereits geschlossen.
422 (Unprocessable Entity)	NOT_AN_ORDER	Dieser Checkout ist nicht vom Typ ORDER .

Refund

Für Bestellungen können Rückzahlungen (Refunds) an den Kunden angestoßen werden.

Die Anlage eines Refunds bezieht sich immer auf einen gesamten Checkout und nicht auf einen konkreten Capture.

Ein Refund ist nur nach erfolgreichen Direct Sales und Orders mit mindestens einem erfolgreichen Capture möglich. In diesem Fall ist an der [Checkout](#)-Resource ein [refunds](#)-Link vorhanden. Der Rückzahlungsbetrag ist auf maximal 200% der Summe der bisher getätigten Zahlungen limitiert.

Refunds sind zeitlich und in der Anzahl unbeschränkt möglich, sofern der maximale Rückzahlungsbetrag nicht überschritten wird.

Anlage

```
POST /api/checkout/v1/checkouts/{checkoutId}/refunds
```



Der Endpoint zur Anlage eines Refunds ist als [refunds](#)-Link in der [Checkout](#)-Ressource enthalten, sofern der Checkout eine Rückzahlung ermöglicht. Der Endpoint sollte nicht direkt aufgerufen werden. Es sollte immer der Link aus der [Checkout](#)-Ressource verwendet werden.

Mit der Erzeugung einer Refund-Ressource wird eine Rückzahlung an den Kunden initiiert.

Request

Feld	Typ	Eigenschaften	Beschreibung
amount	Number	Pflichtfeld. Zwischen 0.01 und 100000. Maximal zwei Nachkommastellen.	Der Betrag der Rückzahlung in Währung der ursprünglichen Bestellung. Die Summe der Rückzahlungen für eine Bestellung ist begrenzt durch den im Feld refundLimit gegebenen Prozentwert des Gesamtwertes der Bestellung.
note	String	Optional. Maximal 37 Zeichen.	Ein Freitext für den Kunden. Der Freitext wird dem Kunden im Verwendungszweck angezeigt.

Feld	Typ	Eigenschaften	Beschreibung
<code>reason</code>	String	Optional.	Grund des Refunds MERCHANT_TECHNICAL_PROBLEM: Technisches Problem bei Abwicklung. MERCHANT_CAN_NOT_DELIVER_GOODS: Händler kann Ware nicht liefern. REFUND_OBLIGINGNESS: Rücküberweisung wegen Kulanz. CUSTOMER_RETURN_GOODS: Rücküberweisung wegen Warenrücksendung.
<code>merchantRefundReferenceNumber</code>	String	Optional. Maximal 30 Zeichen.	Händler-interne Referenznummer für diesen Refund-Vorgang.
<code>merchantReconciliationReferenceNumber</code>	String	Optional. Maximal 30 Zeichen.	Reconciliation-Nummer. Diese wird in den Verwendungszweck der Händlerzahlung eingefügt.
<code>callbackUrlStatusUpdates</code>	String	Optional. Maximal 2000 Zeichen.	URL des Endpoints, der Mitteilungen zu Statusänderungen des Refunds empfangen soll.

Response

In der Response werden die Daten des angelegten Refunds zurückgegeben. Die Struktur ist identisch mit dem Refund Endpoint [GET](#)-Aufruf.

Im HTTP Header Feld `Location` ist die URL der angelegten Refund-Ressource, inkl. Transaction-ID enthalten. Die Transaction-ID wird im Response Body ebenfalls zurückgegeben.

Return Codes

Status Code	Message Code	Beschreibung
201 (Refund created)		Der Refund wurde erfolgreich angelegt. Er befindet sich im Status PENDING .
400 (Bad Request)	VALIDATION_ERROR	Der Request ist syntaktisch ungültig.
401 (Unauthorized)	ACCESS_TOKEN_EXPIRED	Siehe Resource Access .
401 (Unauthorized)	UNAUTHORIZED	Der Zugriff wurde verweigert. Bitte wenden Sie sich an den giropay-Support.
404 (Not Found)	CHECKOUT_NOT_FOUND	Der Checkout wurde nicht gefunden, oder Sie haben keine Zugriff auf diesen Checkout.
422 (Unprocessable Entity)	REFUND_AMOUNT_EXCEEDED	Die Summe aller Rückzahlungen übersteigt die maximale Rückzahlungssumme.

Status Code	Message Code	Beschreibung
422 (Unprocessable Entity)	USER_DEBOARDED	Der Käuferaccount wurde endgültig geschlossen.

Beispiel

```
POST /api/checkout/v1/checkouts/bd3378ba-325c-47a9-9139-aaf39848ec3e/refunds HTTP/1.1
Content-Type: application/json;charset=utf-8
Authorization: Bearer <access_token>
```

```
{
  "amount" : 10,
  "note" : "Ihre Bestellung vom 31.03.2015",
  "merchantRefundReferenceNumber" : "refund-99989",
  "merchantReconciliationReferenceNumber" : "recon-1234",
  "reason" : "MERCHANT_CAN_NOT_DELIVER_GOODS",
  "callbackUrlStatusUpdates" : "https://spielauto-versand.de/callback/status"
}
```

```
HTTP/1.1 201 Created
Location: https://api.paydirekt.de/api/checkout/v1/checkouts/bd3378ba-325c-47a9-9139-aaf39848ec3e/refunds/765aa0ab-4ad8-4a71-97fb-a94d5eaab549
Content-Type: application/hal+json;charset=utf-8
```

```
{
  "type" : "REFUND",
  "transactionId" : "765aa0ab-4ad8-4a71-97fb-a94d5eaab549",
  "amount" : 10,
  "merchantReconciliationReferenceNumber" : "recon-1234",
  "note" : "Ihre Bestellung vom 31.03.2015",
  "merchantRefundReferenceNumber" : "refund-99989",
  "status" : "PENDING",
  "reason" : "MERCHANT_CAN_NOT_DELIVER_GOODS",
  "callbackUrlStatusUpdates" : "https://spielauto-versand.de/callback/status",
  "merchantRefundSettled" : false,
  "_links" : {
    "self" : {
      "href" : "https://api.paydirekt.de/api/checkout/v1/checkouts/bd3378ba-325c-47a9-9139-aaf39848ec3e/refunds/765aa0ab-4ad8-4a71-97fb-a94d5eaab549"
    }
  }
}
```

Abfrage

GET /api/checkout/v1/checkouts/{checkoutId}/refunds/{refundId}

Liefert die Daten zu einer Rückzahlung. Nach einer Initiierung eines Refunds befindet sich der Refund mindestens einen Tag im Status **PENDING**. Es wird empfohlen, den Status 1-2 mal pro Tag abzufragen, bis ein finaler Status (**SUCCESS** oder **FAILED**) erreicht ist.



Die Refund-Transaktionen sind in der Checkout-Ressource embedded. Refundinformationen können also auch durch Abfrage des zugehörigen Checkouts abgerufen werden.

Response

Feld	Typ	Eigenschaften	Beschreibung
type	String	Immer vorhanden.	Typ der Transaktion. Hier immer REFUND .
transactionId	String	Immer vorhanden.	Eindeutige Transaktions-ID dieses Refunds (UUID aus 36 Zeichen). Der Wert wird durch das giropay-System vergeben.
amount	Number	Immer vorhanden.	Der Betrag der Rückzahlung in Währung der ursprünglichen Bestellung. Die Summe der Rückzahlungen für eine Bestellung ist begrenzt durch den im Feld refundLimit gegebenen Prozentwert des Gesamtwertes der Bestellung.
note	String	Vorhanden, falls bei Anlage übergeben.	Ein Freitext für den Kunden. Der Freitext wird dem Kunden im Verwendungszweck angezeigt.

Feld	Typ	Eigenschaften	Beschreibung
<code>reason</code>	String	Optional.	Grund des Refunds MERCHANT_TECHNICAL_PROBLEM: Technisches Problem bei Abwicklung. MERCHANT_CAN_NOT_DELIVER_GOODS: Händler kann Ware nicht liefern. REFUND_OBLIGINGNESS: Rücküberweisung wegen Kulanz. CUSTOMER_RETURN_GOODS: Rücküberweisung wegen Warenrücksendung. TECHNICAL_PROBLEM: Technisches Problem bei Abwicklung/Rückabwicklung. MERCHANT_CAN_NOT_PROOF_SHIPMENT: Händler kann Versandbeleg nicht vorweisen (Dispute). REFUND_ON_BEHALF_OF_MERCHANT: Rückabwicklung im Auftrag des Händlers. REFUND_NOT_ON_BEHALF_OF_MERCHANT: Rückabwicklung ohne Auftrag des Händlers. REFUND_MERCHANT_FRAUD: Rückabwicklung wegen Händlerfraud.
<code>merchantRefundReferenceNumber</code>	String	Vorhanden, falls bei Anlage übergeben.	Händler-interne Referenznummer für diesen Refund-Vorgang.
<code>merchantReconciliationReferenceNumber</code>	String	Vorhanden, falls bei Anlage übergeben.	Reconciliation-Nummer. Diese wird in den Verwendungszweck der Händlerzahlung eingefügt.
<code>paymentInformationId</code>	String	Dieses Feld ist erst vorhanden, wenn die Transaktion im Zahlungsverkehr verarbeitet wurde.	PaymentInformation-Id, die zu dieser Transaktion gehört. Sie dient der Zuordnung einzelner Transaktionen zu einer Sammelbuchung. UUID aus 36 Zeichen.
<code>callbackUrlStatusUpdates</code>	String	Vorhanden, falls bei Anlage übergeben.	URL des Endpoints, der Mitteilungen zu Statusänderungen des Refunds empfangen soll.

Feld	Typ	Eigenschaften	Beschreibung
<code>status</code>	String	Immer vorhanden.	Status des Refunds PENDING : Initialer Status. Der Refund wird im Zahlungsverkehr bearbeitet. ERROR : Der Rückzahlungsvorgang ist aus technischen Gründen temporär unterbrochen. SUCCESSFUL : Der Refund war erfolgreich. Der Betrag ist bereits auf dem Käuferkonto verbucht oder wird gerade von der Käuferbank verarbeitet. FAILED : Der Refund konnte im Zahlungsverkehr endgültig nicht verarbeitet werden und muss neu initiiert werden.
<code>_links</code>	Object		Links zu Ressourcen und verfügbaren Aktionen.

Return Codes

Status Code	Message Code	Beschreibung
200 (Ok)		Die Anfrage wurde korrekt verarbeitet.
401 (Unauthorized)	ACCESS_TOKEN_EXPIRED	Siehe Resource Access .
404 (Not Found)	CHECKOUT_NOT_FOUND	Der Checkout wurde nicht gefunden, oder Sie haben keine Zugriff auf diesen Checkout.
404 (Not Found)	TRANSACTION_NOT_FOUND	Die Rückzahlung mit der angegebenen <code>refundId</code> wurde nicht gefunden.

Beispiel

```
GET /api/checkout/v1/checkouts/bd3378ba-325c-47a9-9139-aaf39848ec3e/refunds/765aa0ab-4ad8-4a71-97fb-a94d5eaab549 HTTP/1.1
Content-Type: application/json;charset=utf-8
Authorization: Bearer <access_token>
```

HTTP/1.1 200 OK

Content-Type: application/hal+json;charset=utf-8

```
{
  "type" : "REFUND",
  "transactionId" : "765aa0ab-4ad8-4a71-97fb-a94d5eaab549",
  "amount" : 10,
  "merchantReconciliationReferenceNumber" : "recon-1234",
  "note" : "Ihre Bestellung vom 31.03.2015",
  "merchantRefundReferenceNumber" : "refund-99989",
  "status" : "PENDING",
  "reason" : "MERCHANT_CAN_NOT_DELIVER_GOODS",
  "callbackUrlStatusUpdates" : "https://spielauto-versand.de/callback/status",
  "merchantRefundSettled" : false,
  "paymentInformationId" : "0000000-1111-2222-3333-444444444444",
  "_links" : {
    "self" : {
      "href" : "https://api.paydirekt.de/api/checkout/v1/checkouts/bd3378ba-325c-47a9-9139-aaf39848ec3e/refunds/765aa0ab-4ad8-4a71-97fb-a94d5eaab549"
    }
  }
}
```

Callback für Statusupdates

Händler/PSPs können bei Anlage des Refunds eine Callback URL `callbackUrlStatusUpdates` übergeben. Das giropay-System sendet bei Statusänderungen des Refunds an diese einen HTTP-Request. Bei Verbindungsfehlern oder Responses mit einem HTTP-Statuscode von 5xx wird das giropay-System die Übermittlung der Statusänderung in einem Zeitfenster von 24 Stunden bis zu 5 Mal wiederholen. Statusänderungen werden dabei immer in der Reihenfolge übermittelt, in der sie am Checkout aufgetreten sind. Kann eine Statusänderung nicht übermittelt werden, so werden auch alle nachfolgenden Statusänderungen nicht übermittelt, bis entweder die fehlgeschlagene Übermittlung nachgeholt werden konnte oder alle wiederholten Versuche fehlgeschlagen sind.

POST `callbackUrlStatusUpdates`



Für die korrekte Ermittlung des Checkout Ergebnisses ist der letzte GET-Aufruf zwingend erforderlich. Siehe auch [Erfolgsstatus eines Checkouts](#).

Request

Feld	Typ	Eigenschaften	Beschreibung
<code>checkoutId</code>	String	Immer vorhanden.	Eindeutige UUID aus 36 Zeichen für diesen Checkout-Vorgang. Wird vom giropay-System vergeben.

Feld	Typ	Eigenschaften	Beschreibung
transactionId	String	Immer vorhanden.	Eindeutige Transaktions-ID dieses Refunds (UUID aus 36 Zeichen). Der Wert wird durch das giro-pay-System vergeben.
merchantRefundReferenceNumber	String	Optional. Maximal 30 Zeichen.	Händler-interne Referenznummer für diesen Refund-Vorgang.
merchantReconciliationReferenceNumber	String	Optional. Maximal 30 Zeichen.	Reconciliation-Nummer. Diese wird in den Verwendungszweck der Händlerzahlung eingefügt.
refundStatus	String		Status des Refunds PENDING: Initialer Status. Der Refund wird im Zahlungsverkehr bearbeitet. ERROR: Der Rückzahlungsvorgang ist aus technischen Gründen temporär unterbrochen. SUCCESSFUL: Der Refund war erfolgreich. Der Betrag ist bereits auf dem Käuferkonto verbucht oder wird gerade von der Käuferbank verarbeitet. FAILED: Der Refund konnte im Zahlungsverkehr endgültig nicht verarbeitet werden und muss neu initiiert werden.
statusUpdateTimeStamp	String	Immer vorhanden.	Zeitstempel der Statusänderung.
sequenceNumber	Number	Immer vorhanden.	Sequenznummer der Statusänderung. Anhand der Sequenznummer kann ein Shop die Reihenfolge der Statusänderungen nachvollziehen und gegebenenfalls fehlende Statusänderungen erkennen.

Beispiel

```
POST /webhooktest/v1/refund/status HTTP/1.1
Content-Type: application/json;charset=utf-8
```

```
{
  "checkoutId" : "6dc67733-9387-463b-b2bd-d3619f63d945",
  "transactionId" : "aa03d5ab-afc3-4d1e-ab47-3bc718ad015a",
  "merchantRefundReferenceNumber" : "refund-12345",
  "merchantReconciliationReferenceNumber" : "reconciliation-12345",
  "refundStatus" : "SUCCESSFUL",
  "statusUpdateTimestamp" : "2020-01-01T12:00:00.000Z",
  "sequenceNumber" : 2
}
```

Response

Die Response enthält keinen Body und wird vom giropay-System nicht weiter verarbeitet. Der HTTP-Statuscode sollte dennoch für etwaige Serviceanfragen Rückschlüsse auf die Verarbeitung der Status-Benachrichtigung zulassen.

Beispiel

```
HTTP/1.1 200 OK
```

Accountinformationen

Die oneKlick API erlaubt den lesenden Zugriff auf ausgewählte Käuferaccountdaten, sofern der entsprechende Scope in der beim [Token Obtain](#) verwendeten Käuferautorisierung enthalten ist.



Die Möglichkeit, [Autorisierungen](#) mit den für diese Funktionen nötigen Scopes erstellen zu können, muss individuell für oneKlick-Anbieter freigeschaltet werden. Bitte wenden Sie sich hierzu an haendler@giropay.de.

Abfrage von Accountinformationen

```
GET /api/accountsummary/v1/accountsummaries/{accountId}
```

Über diesen Endpunkt können Informationen eines giropay-Accounts abgerufen werden. Die zur Verfügung gestellten Informationen hängen von den Scopes der verwendeten [Autorisierung](#) ab.

Folgende Informationen können abgerufen werden:

- Der Benutzername eines Accounts, wenn die verwendete Autorisierung den Scope **READ_USERNAME** besitzt.
- Der Status eines Accounts, wenn die verwendete Autorisierung den Scope **READ_ACCOUNT_STATUS** besitzt.
- Das Guthaben eines Accounts, wenn die verwendete Autorisierung den Scope **READ_CREDIT** besitzt.

Mögliche Ausprägungen des Status eines Accounts sind hierbei:

Status	Beschreibung
ALIAS_RESERVED	Für dieses Konto wurde bereits der Benutzername aktiviert, es ist aber noch nicht initialisiert.
BATCH_ACCOUNT_COMPLETION_REQUIRED	Das Konto wurde über die Batch-Schnittstelle erstellt, der Kunde hat es aber noch nicht durch Vergabe eines Passworts aktiviert.
NOT_LINKED_TO_BANK	Das Konto wurde durch den Kunden im Rahmen der Expressregistrierung erzeugt, aber noch nicht im Online-Banking verknüpft.
EMAIL_ACTIVATION_REQUIRED	Der Kunde hat seine E-Mail-Adresse noch nicht bestätigt.
ACTIVE	Das Konto existiert und ist nutzbar.
LOCKED	Das Konto unterliegt derzeit einer Bezahl- oder Login-Sperre.
NOT_ACTIVE	Das Konto wurde aufgelöst.
REMOVED	Das per Batch-Schnittstelle angelegte Konto wurde aufgelöst, bevor der Kunde es aktiviert hat.

Response

Feld	Typ	Eigenschaften	Beschreibung
<code>accountId</code>	String	Immer vorhanden.	Die ID des Accounts.
<code>username</code>	String	Immer vorhanden, wenn die verwendete Kundenautorisierung das Recht <code>READ_USERNAME</code> besitzt.	Der Benutzername des Accounts.
<code>status</code>	String	Immer vorhanden, wenn die verwendete Kundenautorisierung das Recht <code>READ_ACCOUNT_STATUS</code> besitzt.	Der Status des Accounts.
<code>credit.amount</code>	Number	Immer vorhanden, wenn die verwendete Kundenautorisierung das Recht <code>READ_CREDIT</code> besitzt.	Die Summe verfügbaren Guthabens des Accounts.
<code>credit.currency</code>	String	Immer vorhanden, wenn die verwendete Kundenautorisierung das Recht <code>READ_CREDIT</code> besitzt.	Die Währung des verfügbaren Guthabens des Accounts.

Return Codes

Status Code	Message Code	Beschreibung
200 (Ok)		Der Status wurde erfolgreich abgerufen.
400 (Bad Request)	<code>VALIDATION_ERROR</code>	Der Request ist syntaktisch ungültig.
400 (Bad Request)	<code>CONVERSION_ERROR</code>	Der Request enthält ungültige Zeichen.
401 (Unauthorized)	<code>ACCESS_TOKEN_EXPIRED</code>	Siehe Resource Access .

Status Code	Message Code	Beschreibung
401 (Unauthorized) und 403 (Forbidden)	UNAUTHORIZED	Der Zugriff wurde verweigert. Bitte prüfen Sie, ob die genutzte Käuferautorisierung den o.g. Scope enthält.
404 (Not Found)		Der Account wurde nicht gefunden oder darf vom oneKlick-Anbieter nicht abgerufen werden.

Beispiel

```
GET /api/accountssummary/v1/accountssummaries/b2dce62f-065c-4aea-838e-95c34e242978
HTTP/1.1
Content-Type: application/json;charset=utf-8
Authorization: Bearer <access_token>
```

```
HTTP/1.1 200 OK
Content-Type: application/hal+json;charset=utf-8
```

```
{
  "accountId" : "b2dce62f-065c-4aea-838e-95c34e242978",
  "username" : "MaxMuster1999",
  "status" : "ACTIVE",
  "credit" : {
    "amount" : 19.29,
    "currency" : "EUR"
  },
  "_links" : {
    "self" : {
      "href" :
        "https://api.paydirekt.de/api/accountssummary/v1/accountssummaries/b2dce62f-065c-4aea-838e-95c34e242978"
    }
  }
}
```

Abfrage der Transaktionshistorie

```
GET
/api/transaction/v1/activities/history/accounts/{accountId}/transactions?from={fromDate}&to={toDate}&page={page}&size={size}
```

Über diesen Endpunkt können die Zahlungen des giropay-Kontos abgerufen werden. Diese werden hierbei auf die wesentlichen Informationen beschränkt. Der relevante Scope für diese Funktion ist **READ_TRANSACTIONS**. Die Transaktionen werden **paginiert** zurückgeben.

Query-Parameter

Feld	Typ	Eigenschaften	Beschreibung
<code>fromDate</code>	Datum	Optional	Grenzt die abgerufenen Transaktionen auf solche ein, die nach dem angegebenen Datum durchgeführt worden sind.
<code>toDate</code>	Datum	Optional	Grenzt die abgerufenen Transaktionen auf solche ein, die vor dem angegebenen Datum durchgeführt worden sind.
<code>size</code>	Datum	Optional	Bestimmt die abzurufende Page für diese Abfrage.
<code>size</code>	Datum	Optional	Bestimmt die Größe der Pages für diese Abfrage.

Response

Feld	Typ	Eigenschaften	Beschreibung
<code>_embedded.transactions</code>	Array	Immer vorhanden.	Stellt die Liste der gesuchten Aktivitäten dar.
<code>_embedded.transactions[].type</code>	String	Immer vorhanden.	Gibt den Typ der Aktivität an. Mögliche Werte sind: ORDER_OPEN: Bestellung, ORDER_PARTIALLY_CAPTURED: Bestellung mit Teilzahlung, ORDER_FULLY_CAPTURED: Abgeschlossene Bestellung, ORDER_CANCELED: Stornierte Bestellung, CAPTURE_DIRECT_SALE: Zahlung, CAPTURE_ORDER: Zahlung, REFUND: Rückzahlung, OVERCAPTURE: Zahlung, MONEY_TRANSFER: Geld Senden, MONEY_REQUEST: Geld Empfangen.
<code>_embedded.transactions[].amount</code>	Number	Immer vorhanden. Zwei Nachkommastellen.	Gibt absoluten Betrag der Aktivität in Euro an an.
<code>_embedded.transactions[].usedCreditAmount</code>	Number	Immer vorhanden. Zwei Nachkommastellen.	Gibt den für die Aktivität verwendeten Gutschein-Betrag an.

Feld	Typ	Eigenschaften	Beschreibung
<code>_embedded.transactions[].usedPromotionalCreditAmount</code>	Number	Immer vorhanden. Zwei Nachkommastellen.	Gibt den für die Aktivität verwendeten Promotions-Betrag an.
<code>_embedded.transactions[].currency</code>	String	Immer vorhanden.	Gibt die Währung des Betrags der Aktivität an.
<code>_embedded.transactions[].debtorName</code>	String	Immer vorhanden.	Gibt Namen des Schuldners für die Aktivität an.
<code>_embedded.transactions[].creditorName</code>	String	Immer vorhanden.	Gibt Namen des Gläubigers für die Aktivität an.
<code>_embedded.transactions[].moneyTransferParticipantRole</code>	String	Bei Transaktionen vom Typ <code>MONEY_TRANSFER</code> immer vorhanden. Bei Transaktionen eines anderen Typs nie vorhanden.	Gibt die Rolle des Kunden bei einem Money Transfer an: <code>SENDER</code> , wenn der Kunde die Zahlung gesendet hat; <code>RECIPIENT</code> wenn der Kunde die Zahlung empfangen hat.
<code>_embedded.transactions[].merchantOrderReferenceNumber</code>	String	Optional.	Gibt die Bestellnummer des Vorgangs, die vom Händler vergeben wurde, an.
<code>_embedded.transactions[].merchantCustomerNumber</code>	String	Optional.	Gibt die Kundennummer des Käufers beim Händler an.
<code>_embedded.transactions[].endToEndReferenceNumber</code>	String	Immer vorhanden.	Gibt die eindeutige Referenz des Vorgangs im System an.
<code>_embedded.transactions[].creationTimestamp</code>	String	Immer vorhanden.	Gibt den initialen Zeitpunkt der Aktivität an.
<code>_embedded.transactions[].updateTimestamp</code>	String	Immer vorhanden.	Gibt den Zeitpunkt der letzten Änderung an der Aktivität an.
<code>_embedded.transactions[].shippingAddress</code>	ShippingAddress	Optional.	Enthält die für die Transaktion verwendete Lieferadresse.
<code>_embedded.transactions[].shoppingCartItems</code>	Liste von Items	Optional.	Enthält den für die Transaktion angegebenen Warenkorbinhalt.
<code>page</code>	Page	Immer vorhanden.	Enthält Pagination-Informationen für die zurückgegebene Ressourcen-Liste.

Return Codes

Status Code	Message Code	Beschreibung
200 (Ok)		Die Transaktionen wurden erfolgreich abgerufen.
400 (Bad Request)	VALIDATION_ERROR	Der Request ist syntaktisch ungültig.
400 (Bad Request)	CONVERSION_ERROR	Der Request enthält ungültige Zeichen.
401 (Unauthorized)	ACCESS_TOKEN_EXPIRED	Siehe Resource Access .
401 (Unauthorized) und 403 (Forbidden)	UNAUTHORIZED	Der Zugriff wurde verweigert. Bitte prüfen Sie, ob die genutzte Käuferautorisierung den o.g. Scope enthält.

Beispiel

```
GET /api/transaction/v1/activities/history/accounts/1f322aed-baae-4007-8c31-adea2c1050d2/transactions?from=2015-01-01T00:00:00.000Z&to=2018-01-01T00:00:00.000Z&page=0&size=50 HTTP/1.1
Accept: application/hal+json
Authorization: Bearer <access_token>
```

```
HTTP/1.1 200 OK
Content-Type: application/hal+json;charset=utf-8
```

```
{
  "_embedded" : {
    "transactions" : [ {
      "type" : "CAPTURE_DIRECT_SALE",
      "amount" : 10,
      "usedCreditAmount" : 2,
      "usedPromotionalCreditAmount" : 3,
      "currency" : "EUR",
      "shoppingCartItems" : [ {
        "ean" : "EAN 1",
        "name" : "Item 1",
        "quantity" : 1,
        "price" : 1
      } ],
      "debitorName" : "",
      "creditorName" : "SomeCreditor_CO",
      "shippingAddress" : {
        "addresseeGivenName" : "Hans",
        "addresseeLastName" : "Wurst",
        "company" : "Senacor Technologies AG",
        "street" : "Wieseneckstrasse",
        "streetNr" : "26",
        "additionalAddressInformation" : "3. Stock",
```

```

    "zip" : "90571",
    "city" : "Schwaig b. Nürnberg",
    "countryCode" : "DE",
    "state" : "Frankonia"
  },
  "merchantOrderReferenceNumber" : "order-0815",
  "merchantCustomerNumber" : "customer-0815",
  "endToEndReferenceNumber" : "e2e-0815",
  "creationTimestamp" : "2016-07-13T12:34:56.789Z",
  "updateTimestamp" : "2016-07-14T12:34:56.789Z"
}, {
  "type" : "MONEY_TRANSFER",
  "amount" : 20,
  "currency" : "EUR",
  "debtorName" : "debtorFirstName debtorLastName",
  "creditorName" : "creditorFirstName creditorLastName",
  "moneyTransferParticipantRole" : "RECIPIENT",
  "merchantOrderReferenceNumber" : "merchantOrderReferenceNumber",
  "merchantCustomerNumber" : "merchantCustomerNumber",
  "endToEndReferenceNumber" : "e2e-0816",
  "creationTimestamp" : "2016-07-15T12:34:56.789Z",
  "updateTimestamp" : "2016-07-16T12:34:56.789Z"
} ]
},
"_links" : {
  "self" : {
    "href" :
      "https://api.paydirekt.de/api/transaction/v1/activities/history/accounts/1f322aed-
      baae-4007-8c31-adea2c1050d2/transactions?from=2015-01-01T00%3A00%3A00Z&to=2018-01-
      01T00%3A00%3A00Z"
  }
},
"page" : {
  "size" : 50,
  "totalElements" : 2,
  "totalPages" : 1,
  "number" : 0
}
}

```

SDKs für mobile Apps

Die Entwicklung und der Support für die iOS und Android SDKs wurden eingestellt. Stattdessen sollten die API-Endpunkte direkt angesprochen werden.

Allgemeine Datentypen

Neben den Standard JSON-Datentypen gibt es komplexe Datentypen, die für einen ordentlichen Request benötigt werden.

ShippingAddress

Datentyp für Liefer- und Rechnungsadressen.

Feld	Typ	Eigenschaften	Beschreibung
<code>addresseeGivenName</code>	String	Optional bei als <code>AUTHORITIES_PAYMENT</code> oder <code>ANONYMOUS_DONATION</code> kategorisierten Warenkörben, sonst Pflichtfeld. Maximal 100 Zeichen.	Vorname.
<code>addresseeLastName</code>	String	Optional bei als <code>AUTHORITIES_PAYMENT</code> oder <code>ANONYMOUS_DONATION</code> kategorisierten Warenkörben, sonst Pflichtfeld. Maximal 100 Zeichen.	Nachname.
<code>company</code>	String	Optional. Maximal 100 Zeichen.	Firmenname.
<code>additionalAddressInformation</code>	String	Optional. Maximal 100 Zeichen.	Adresszusatz.
<code>street</code>	String	Optional. Maximal 100 Zeichen.	Der Name der Straße, ohne Hausnummer.
<code>streetNr</code>	String	Optional. Maximal 10 Zeichen.	Die Hausnummer.
<code>zip</code>	String	Optional bei als <code>DIGITAL</code> , <code>AUTHORITIES_PAYMENT</code> oder <code>ANONYMOUS_DONATION</code> kategorisierten Warenkörben, sonst Pflichtfeld. Maximal 10 Zeichen.	Die Postleitzahl.
<code>city</code>	String	Optional bei als <code>DIGITAL</code> , <code>AUTHORITIES_PAYMENT</code> oder <code>ANONYMOUS_DONATION</code> kategorisierten Warenkörben, sonst Pflichtfeld. Maximal 100 Zeichen.	Die Stadt.

Feld	Typ	Eigenschaften	Beschreibung
countryCode	String	Optional bei als DIGITAL , AUTHORITIES_PAYMENT oder ANONYMOUS_DONATION kategorisierten Warenkörben, sonst Pflichtfeld. 2 Zeichen.	Der Ländercode im ISO 3166-1 Format.
state	String	Optional 100 Zeichen.	Bundesland oder Bundesstaat. Kann zum Beispiel bei Auslandsadressen verwendet werden.
emailAddress	String	Pflichtfeld bei als DIGITAL kategorisierten Warenkörben, sonst optional. Muss eine gültige Email-Adresse sein.	Die Email-Adresse des Käufers.

```
{
  "addresseeGivenName" : "Hermann",
  "addresseeLastName" : "Meyer",
  "street" : "Wieseneckstraße",
  "streetNr" : "26",
  "zip" : "90571",
  "city" : "Schwaig bei Nürnberg",
  "countryCode" : "DE",
}
```

Beispiel mit internationaler Adresse mit Bundesland

```
{
  "addresseeGivenName" : "John",
  "addresseeLastName" : "Smith",
  "street" : "Main St",
  "streetNr" : "799 E",
  "zip" : "85705",
  "city" : "Tucson",
  "countryCode" : "US",
  "state" : " AZ
}
```

Item

Ein *Item* ist eine Warenkorbposition.

Feld	Typ	Eigenschaften	Beschreibung
<code>quantity</code>	Number	Pflichtfeld. Ganzzahlig. Mindestens 1.	Die Anzahl der Positionen für diesen Artikel.
<code>name</code>	String	Pflichtfeld. Maximal 100 Zeichen.	Die Bezeichnung des Artikel.
<code>ean</code>	String	Optional. Maximal 100 Zeichen.	Die International Article Number (EAN bzw. GTIN) des Artikels.
<code>price</code>	Number	Pflichtfeld. Positiv oder Negativ. Maximal 2 Nachkommastellen.	Der Einzelpreis eines Artikels, inkl. Steuern. Bei Gutscheinen (z. B. in Form vom Gutscheinen) kann der Betrag auch negativ sein.

```
{
  "quantity" : 3,
  "name" : "Bobbycar",
  "ean" : "800001303",
  "price" : 25.99
}
```

Message

Meldungen und insbesondere Fehlermeldungen. Details zur Fehlerbehandlung siehe [Error Messages](#).

Feld	Typ	Eigenschaften	Beschreibung
<code>code</code>	String	Pflichtfeld. Maximal 100 Zeichen.	Der eindeutige Fehlercode zu diesem Fehler.
<code>severity</code>	String	Pflichtfeld.	Die Fehlerstufe. Die möglichen Werte sind: ERROR, WARN und INFO.
<code>path</code>	String	Optional. Maximal 1000 Zeichen.	Das Feld zu dem dieser Fehler erzeugt wurde. Dieses Feld wird bei einem <code>VALIDATION_ERROR</code> (Response Code 400) befüllt.

Feld	Typ	Eigenschaften	Beschreibung
<code>reasonCode</code>	String	Optional. Maximal 1000 Zeichen.	Der Grund für diesen Fehler als eindeutiger Code, sofern dies aus dem Feld <code>code</code> nicht eindeutig hervorgeht. Der <code>reasonCode</code> dient vor allem zu Debugging-Zwecken, es sollte hierauf keine Logik implementiert werden.
<code>logref</code>	String	Optional. Maximal 1000 Zeichen.	Eine giro-pay-interne Referenznummer zu dieser Fehlermeldung. Bitte geben Sie immer diese Referenznummer bei Support-Anfragen an.
<code>content</code>	String	Optional.	Der ungültige Inhalt bei einem <code>CONVERSION_ERROR</code> .

```
{
  "code" : "VALIDATION_ERROR",
  "severity" : "ERROR",
  "path" : "currency",
  "reasonCode" : "INVALID_FORMAT",
  "logref" : "934180ed-ef93-4af1-835a-3071714fdce5:8t1ph9gFha1"
}
```

Page

Beschreibt die Segmentierung großer Ressourcen in Abschnitte, die einzeln abgerufen und angezeigt werden können.

Feld	Typ	Eigenschaften	Beschreibung
<code>size</code>	Integer	Immer vorhanden.	Die maximale Anzahl der Elemente in dieser Page.
<code>totalElements</code>	Integer	Immer vorhanden.	Die Gesamtanzahl der Elemente über alle Pages.
<code>totalPages</code>	Integer	Immer vorhanden.	Die Gesamtanzahl der Pages.
<code>number</code>	Integer	Immer vorhanden.	Der Index der aktuellen Page.

Zeichensatz

Grundsätzlich gilt, dass die per HTTP-Request übergebenen Daten UTF-8 encodiert sein müssen. Sofern nicht anders vermerkt, werden nur folgende Zeichen in Feldern vom Typ `String` akzeptiert:

- Alle Buchstaben in der Unicode-Kategorie `Letter`
- Alle Zahlen in der Unicode-Kategorie `Number`
- Folgende Sonderzeichen: `.-!#$%&'*+/?^_`´{|}~"(),;:<>@[]`
- Leerzeichen und geschütztes Leerzeichen
- Linebreak und Linefeed

Auch giropay liefert sämtliche Responses in UTF-8. Allgemein ist dabei zu beachten, dass nicht garantiert ist, dass nicht-SEPA-konforme Zeichen in allen Systemen (insbesondere auf Bankenseite, z. B. auf Kontoauszügen) korrekt, bzw. überhaupt, angezeigt werden.

SEPA-konforme Zeichen

In SEPA-Zahlungen sind gemäß [Anlage 3 des DFÜ-Abkommens](#) nur Zeichen des eingeschränkten SWIFT Latin Character Set zugelassen:

- `a b c d e f g h i j k l m n o p q r s t u v w x y z`
- `A B C D E F G H I J K L M N O P Q R S T U V W X Y Z`
- `0 1 2 3 4 5 6 7 8 9`
- `' : ? , - (+ .) /`

In dem eingeschränkten Zeichensatz sind somit insbesondere nicht enthalten: Umlaute, das €-Symbol, bestimmte Sonderzeichen wie der Unterstrich, das Leerzeichen oder eckige Klammern.

Zudem dürfen Referenzen, Kennzeichnungen und Kennungen weder mit einem Schrägstrich `„/“` beginnen oder enden, noch zwei aufeinanderfolgende Schrägstriche `„//“` beinhalten.

Damit es im Zahlungsverkehr zu keinen Störungen kommt, dürfen für entsprechend gekennzeichnete Attribute nur die zugelassenen Zeichen verwendet werden.

Bei Verwendung von nicht zugelassenen Zeichen oder unzulässiger Verwendung von zugelassenen Zeichen kann eine Zahlung oder auch ein gesamter Zahlungscontainer zurückgewiesen werden.

Sandbox Testing

Um automatisiertes Testen zu ermöglichen, gibt es auf der Sandbox-Umgebung unter `/testsupport/*` zusätzliche Schnittstellen um zwingend erforderliche Nutzerinteraktionen zu überspringen oder das Erzeugen von Testfällen zu vereinfachen.



Die Features steht nur auf der Sandbox-Umgebung zur Verfügung.



Weiterführende Informationen auf: <https://integration.giropay.de/oneclick/sandboxsystem/index.html>

Anpassen des Autorisierungsstatus

```
PATCH /testsupport/v1/customerauthorizations/{authorizationId}
```

Eine zuvor angelegte Autorisierung kann über die PATCH-Methode modifiziert werden, um ihren Status und andere Eigenschaften anzupassen. Dieser Endpunkt kann bspw. verwendet werden um eine Autorisierungsbestätigung durch einen Käufer in der giropay App nachzustellen.

Request

Feld	Typ	Eigenschaften	Beschreibung
<code>newStatus</code>	String	Pflichtfeld.	Status der Autorisierung. Mögliche Werte sind: OPEN : Angelegt, aber noch nicht vom Käufer bestätigt. APPROVED : Vom Käufer bestätigt. REJECTED : Vom System endgültig abgelehnt EXPIRED : Wurde nicht innerhalb von 5 Minuten erteilt oder durch eine neue, inhaltlich gleiche Autorisierung ersetzt. REVOKED : Wurde zurückgezogen. Dies kann durch den Kunden selbst oder einen paydirekt-Mitarbeiter geschehen sein.
<code>accountId</code>	String	Pflichtfeld.	Gewünschte Id des Accounts, für den die Autorisierung bestätigt bzw. abgelehnt werden soll.
<code>apiKey</code>	String	Pflichtfeld.	Siehe api-key .
<code>apiSecret</code>	String	Pflichtfeld.	Siehe api-key .

Feld	Typ	Eigenschaften	Beschreibung
<code>merchantAuthorizationReference</code>	String	Optionales Feld.	<code>merchantAuthorizationReference</code> einer gültigen Händlerautorisierung.
<code>customerAuthorizationRef</code>	String	Optionales Feld.	<code>customerAuthorizationReference</code> einer gültigen Käuferautorisierung.

Response

Feld	Typ	Eigenschaften	Beschreibung
<code>customerAuthorizationId</code>	String	Immer vorhanden.	ID der Autorisierungsresource.
<code>thirdPartyId</code>	String	Immer vorhanden.	ID des oneKlick-Anbieters.
<code>clientId</code>	String	Immer vorhanden.	ID der oneKlick-Anwendung.
<code>stronglyAuthenticated</code>	Boolean	Vorhanden, nachdem die Autorisierung durch den Kunden durch einen ersten Faktor bestätigt worden ist.	Gibt an, ob der Benutzer sich für die Bestätigung der Autorisierung bereits über einen zweiten Faktor stark authentifiziert hat. Autorisierungen ohne starke Authentifizierung sind nur für eine kurze Zeit nach ihrer Bestätigung durch den ersten Faktor gültig.
<code>creationTimestamp</code>	String	Immer vorhanden.	Zeitpunkt, zu dem die Autorisierung erstellt wurde. Formatierung entsprechend ISO-8601 durch den Standard Java DateTimeFormatter . Abweichend davon werden immer drei Nachkommastellen für die Millisekunden ausgegeben.
<code>approvalTimestamp</code>	String	Vorhanden, sobald die Autorisierung den Status <code>APPROVED</code> erreicht hat.	Zeitpunkt, zu dem die Autorisierung vom Käufer erteilt wurde. Formatierung wie <code>creationTimestamp</code> .
<code>revokeTimestamp</code>	String	Vorhanden, sobald die Autorisierung den Status <code>REVOKED</code> erreicht hat.	Zeitpunkt, zu dem die Autorisierung zurückgezogen wurde. Formatierung wie <code>creationTimestamp</code> .
<code>scopes</code>	Array	Immer vorhanden.	Umfang der vom Käufer zu erteilenden Berechtigungen.

Feld	Typ	Eigenschaften	Beschreibung
<code>status</code>	String	Immer vorhanden.	Status der Autorisierung. Mögliche Werte sind: OPEN : Angelegt, aber noch nicht vom Käufer bestätigt. APPROVED : Vom Käufer bestätigt. REJECTED : Vom System endgültig abgelehnt EXPIRED : Wurde nicht innerhalb von 5 Minuten erteilt oder durch eine neue, inhaltlich gleiche Autorisierung ersetzt. REVOKED : Wurde zurückgezogen. Dies kann durch den Kunden selbst oder einen paydirekt-Mitarbeiter geschehen sein.
<code>redirectUrlAfterSuccess</code>	String	Vorhanden, falls bei der Anlage übergeben.	Die Rücksprungadresse, die aufgerufen wird, nachdem die Autorisierung erfolgreich durch den Kunden bestätigt worden ist.
<code>redirectUrlAfterCancellation</code>	String	Vorhanden, falls bei der Anlage übergeben.	Die Rücksprungadresse, die aufgerufen wird, nachdem der Kunde die Bestätigung der Autorisierung abgebrochen hat.
<code>redirectUrlAfterFailure</code>	String	Vorhanden, falls bei der Anlage übergeben.	Die Rücksprungadresse, die aufgerufen wird, nachdem die Bestätigung der Autorisierung durch einen technischen Fehler abgebrochen worden ist.

Beispiel

Request

```
{
  "newStatus" : "APPROVED",
  "accountId" : "someAccountId",
  "apiKey" : "someApiKey",
  "apiSecret" : "someSecret"
}
```

Response

```

{
  "customerAuthorizationId" : "310ccd0b-54f3-4177-bb28-91a7b816c2f6",
  "thirdPartyId" : "0fd94c3d-c4a3-430e-ab50-00ed326bb6e2",
  "clientId" : "7a9f88f1-8068-495b-b1b2-202b7c7fb315",
  "status" : "APPROVED",
  "scopes" : [ ],
  "creationTimestamp" : "2021-05-11T09:11:32.171Z",
  "approvalTimestamp" : "2021-05-11T09:11:32.888Z",
  "stronglyAuthenticated" : true,
  "_links" : {
    "self" : {
      "href" :
"https://api.paydirekt.de/api/test/thirdpartycustomerauthorization/v2/authorizations/89a3d981-44f3-4622-bb2f-4a6e8b6df53c"
    },
    "appConfirmationStatus" : {
      "href" :
"https://api.paydirekt.de/api/test/thirdpartycustomerauthorization/v1/authorizations/appconfirmationstatus/89a3d981-44f3-4622-bb2f-4a6e8b6df53c"
    },
    "revoke" : {
      "href" :
"https://api.paydirekt.de/api/test/thirdpartycustomerauthorization/v2/authorizations/89a3d981-44f3-4622-bb2f-4a6e8b6df53c"
    }
  }
}

```

Changelog

Die folgende Tabelle fasst alle Änderungen an diesem Dokument seit April 2016 zusammen.

Datum	Art	Beschreibung	Abschnitt
24.10.2022	NEW FEATURE	Neuer Status <code>CHECKOUT_TYPE_NOT_SUPPORTED_BY_BANK</code> bei Autorisierung mit CheckoutType <code>ORDER_SECURED</code> , wenn die Bank dieses Feature noch nicht unterstützt	Abfrage Käuferautorisierung
07.09.2022	CHANGE	Kürzung Felddescription	Requestfelder Checkoutanlage, Responsefelder Checkoutanlage
30.08.2022	CHANGE	Neues optionales Feld <code>checkoutType</code> bei Anlage von Autorisierungen	Anlage Käuferautorisierung
08.06.2022	CHANGE	Umformulierungen im Rahmen der Markenmigration von paydirekt zu giro pay	
17.05.2022	DELETE	Entfernen aller GitHub-Verweise und -Links	Access Token
09.05.2022	NEW INFO	Expliziter Hinweis zur Ergebnisermittlung eines Checkouts	Erfolgsstatus eines Checkouts
09.11.2021	NEW INFO	Zurückziehen von Autorisierungen	Zurückziehen von Autorisierungen
09.11.2021	CHANGE	Anpassung des Bereichs für Autorisierungen	Autorisierungen
09.11.2021	NEW INFO	Ergänzung der möglichen Werte für Transaktionen bei Abfrage der Transaktionshistorie	Abfrage der Transaktionshistorie
12.08.2021	CHANGE	SEPA-konforme Zeichen um Einschränkungen für "/" erweitert	SEPA-konforme Zeichen
06.07.2021	CHANGE	Die Angabe der Felder <code>contextInformation.deviceIpAddress</code> , <code>contextInformation.terminalId</code> und <code>contextInformation.terminalAddress</code> bei der Anlage von Checkouts sind nur noch für 'Point of Sales' Shops verpflichtend.	Checkout - Anlage
28.05.2021	NEW FEATURE	Auf der Sandbox-Umgebung kann der Status einer Autorisierung verändert werden.	Sandbox Testing
06.05.2021	NEW INFO	Informationen über die Verwendung von Rücksprungadressen ergänzt.	Autorisierungsdialog

Datum	Art	Beschreibung	Abschnitt
06.05.20 21	NEW INFO	Informationen über die Einstellung der mobilen SDKS ergänzt.	SDKs für mobile Apps
05.05.20 21	NEW INFO	Informationen über das Verhalten bei nicht angegebenen Rücksprungadressen ergänzt.	Verhalten bei nicht angegebenen Rücksprungadressen
15.06.20 20	NEW FEATURE	Aktive Benachrichtigungen über Statusupdates von Käuferautorisierungen über Callback-URL	Callback für Statusupdates
08.04.20 20	NEW INFO	Informationen über den Standardwert von <code>requestedPreauthorizationValidity</code> ergänzt.	Checkout - Anlage
30.08.20 19	BUGFIX	Anlage von Käuferautorisierungen während eines Checkouts kann jetzt von allen oneKlick-Anwendungen genutzt werden	Anlage Käuferautorisierung während eines Checkouts
18.06.20 19	NEW FEATURE	Anlage von Käuferautorisierungen während eines Checkouts	Anlage Käuferautorisierung während eines Checkouts
13.05.20 19	NEW INFO	Beschreibung der Anlage von normalen Checkouts	
16.04.20 19	NEW FEATURE	Neuer Scope für wiederkehrende Zahlungen	Anlage Käuferautorisierung
05.04.20 19	NEW INFO	Beschreibung der Nutzung dieser API durch CPSPs	
04.04.20 19	FIX	Die Beschreibung der Felder beim Abruf einer Käuferautorisierung ist jetzt vollständig.	Authorization - Abfrage
02.01.20 19	CHANGE	Händler bekommen eine explizite Fehlermeldung, falls der Kunde zwischenzeitlich nicht mehr an paydirekt teilnimmt.	Capture - Anlage
12.11.20 18	NEW INFO	Dokumentation der in String-Feldern akzeptierten Zeichen	Zeichensatz
03.09.20 18	NEW FEATURE	Neue API für den Abruf von Accountinformationen	Abruf von Accountinformationen
30.08.20 18	NEW FEATURE	Ausgabe der Rolle (Sender/Empfänger) bei Money Transfers	Abruf von Transaktionsdaten
21.08.20 18	NEW FEATURE	Ausgabe von Lieferadresse und Warenkorb in den Käufer-Transaktionsdaten	Abruf von Transaktionsdaten
28.05.20 18	NEW INFO	Dokumentation des Fehlercodes <code>ORDER_NOT_APPROVED</code>	Capture - Order schließen
04.05.20 18	CHANGE	Weitere Ausprägungen des <code>accountStatus</code> bei Abfrage desselbigen	Abfrage des Accountstatus

Datum	Art	Beschreibung	Abschnitt
30.04.2018	NEW FEATURE	Auch der Benutzername kann nun abgerufen werden, wenn ein Käufer die Berechtigung dazu erteilt hat.	Abfrage des Benutzernamens (Kapitel entfernt)
12.04.2018	NEW FEATURE	Ausgewählte Informationen zu Käufer-Accounts können abgerufen werden, wenn ein Käufer die Berechtigung dazu erteilt hat.	Abruf von Accountinformationen, Authorization - Anlage
09.04.2018	FIX	Das Feld <code>stronglyAuthenticated</code> ist nicht Teil der Autorisierungsanlage.	Authorization - Anlage
29.03.2018	NEW INFO	Der Request zur Anlage einer Käufer-Autorisierung kann die Details für wiederkehrende regelmäßige Zahlungen enthalten.	Authorization
21.03.2018	NEW INFO	Erweiterte Dokumentation der starken Authentifizierung von Käuferautorisierungen und des Umgangs mit Duplikaten.	Starke Authentifizierung von Käuferautorisierungen Duplikate von Käuferautorisierungen
16.03.2018	CHANGE	Umbenennung von Feldern an Käuferautorisierungen: <code>thirdPartyAuthorizationId</code> und <code>thirdPartyAuthorizationReference</code> heißen nun <code>customerAuthorizationId</code> und <code>customerAuthorizationReference</code> . Die alten Feldernamen werden während einer Übergangszeit noch zusätzlich ausgeliefert.	Authorization - Abfrage
22.02.2018	NEW FEATURE	Neues Feld <code>stronglyAuthenticated</code> beim Abruf von Käuferautorisierungen	Authorization - Abfrage
22.02.2018	NEW INFO	Aktualisierte Dokumentation der mobile SDKs	SDKs für mobile Apps
21.02.2018	NEW FEATURE	Zusätzliches Attribut <code>deviceFingerprint</code> bei der Anlage von Käuferautorisierungen und oneKlick-Checkouts	Authorization - Anlage, Checkout - Anlage
15.02.2018	NEW FEATURE	Unterstützung von Rücksprungadressen bei der Anlage von Käuferautorisierungen	Authorization - Anlage
10.01.2018	NEW FEATURE	Neuer Prozessschritt bei der Autorisierungserteilung	
22.12.2017	NEW INFO	Angaben zum Vorhandensein aller Response-Felder	

Datum	Art	Beschreibung	Abschnitt
29.11.2017	NEW FEATURE	Versandbedingungen und Rechnungsreferenz können nachträglich geändert werden.	Checkout - Aktualisierung Versandbedingungen Checkout - Aktualisierung Rechnungsreferenz
18.10.2017	FIX	Beschreibung möglicher Fehlerfälle beim Abruf des Access Tokens	Authentifizierung - Access Token
06.09.2017	NEW FEATURE	Bei der Checkoutanlage kann jetzt das für diesen geltende maximale Refundlimit gesetzt werden.	Checkout - Anlage
24.08.2017	NEW FEATURE	Beschreibung des Abrufs der Händlerautorisierung	Händlerauthorization - Abfrage
09.08.2017	NEW FEATURE	Beschreibung der gesicherten Vorbestellung	Checkout - Anlage
28.06.2017	FIX	Beschreibung möglicher Fehlerfälle bei der oneKlick-Checkoutanlage	Checkout - Anlage
14.06.2017	NEW FEATURE	Mobile SDKs	SDKs für mobile Apps
14.06.2017	CHANGE	Die Lieferadresse ist nun für oneKlick-Checkouts optional.	Checkout - Anlage
22.03.2017	CHANGE	Neue URLs der Endpunkte für Autorisierungen	Authorization - Anlage Authorization - Abfrage
22.02.2017	NEW FEATURE	Statusupdates werden bei Fehlschlägen erneut gesendet.	Checkout - Callback
08.02.2017	NEW FEATURE	Bei der Anlage von OneKlick Checkouts und Authorizations müssen nun zusätzliche Informationen zur Fraudprevention übermittelt werden.	Checkout - Anlage Authorization - Anlage
08.02.2017	NEW FEATURE	OneKlick Checkoutanlage	Checkout - Anlage
25.01.2017	NEW FEATURE	Autorisierungsanlage und -abfrage	Authorization - Anlage Authorization - Abfrage
25.01.2017	NEW FEATURE	Einführung der neuen API für oneKlick	