




This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

API Reference

[Start Service](#)

Start Service

	POST /services/start
	Available since: 8.0.200.00

Contents

- [1 Start Service](#)
 - [1.1 Description](#)
 - [1.2 Operation](#)
 - [1.3 Response](#)
 - [1.4 Example](#)

Description

Starts a service based on:

- The service type passed in the body.
- The contact information available (customer ID or contact key for later lookup).


If your application does not assign a customer ID, your service is anonymous. In that case, your application must assign a contact key. For further information, read the following pages:

- [Documentation:CS:Developer:Anonymous Service](#)
- [Service Start Event](#)

 Your application is not responsible for the service ID creation. UCS automatically assigns 64-bit integer IDs to the created services.

Operation

Start Service

ID	CV.WS.SRV.1		
Method	POST		
URL	/services/start		
Field Name	Type	Mandatory	Description
Body: Service Start Event<ref>The body of this operation is mandatory and makes use of the Service Start Event .</ref>			
customer_id	string	yes if no <i>contact_key</i>	The unique ID of the customer associated with the given service.  Services with no <i>customer_id</i> are anonymous . In this case, the event should include a <i>contact_key</i> .
contact_key	string	yes if no <i>customer_id</i>	Key for later lookup of the service. This <i>contact_key</i> is mandatory if no customer ID is specified. For example, the application might store a PIN or ANI as the contact key when the service is first started. Later, if the customer is

ID	CV.WS.SRV.1		
			identified then the contact key is used to lookup the existing service record and associate it with the customer.
service_type	long or string	yes	<p>The unique ID associated with the type of service, typically the DB ID of a value in the Service Type Business Attribute<ref name="business"></p> <p>Refer to Configuration Options for more details on Business Attribute mapping.</ref>.</p>
session_id	string	no	The ID of the related session, for instance, the orchestration session, or any other business session.
interaction_id	string	no	The ID of the related Genesys interaction. This ID can be used by other Genesys reporting products such as Stat Server, URS, Composer, and GVP.
application_type	long or string	no	The unique ID associated with the type or class<ref name="business"/>application issuing the service event. May be used to group related applications, potentially across resource types.
application_id	integer	no	The unique ID (Genesys DB ID) for the application issuing the service event, such as a GVP VoiceXML application or an Orchestration SCXML application.
resource_type	long or string	no	The unique ID associated with the type or class<ref name="business"/> of resource which provides the service (such as GVP, Agent Desktop, or

ID	CV.WS.SRV.1		
			Orchestration).
resource_id	integer	no	<p>The unique DB ID for the specific resource which provides the service. For instance:</p> <ul style="list-style-type: none"> the Genesys DB ID of a specific GVP or orchestration platform. the DB ID of a given agent, according to the context.
media_type	long or string	no	The media type<ref name="business"/> applicable to the given service, for instance, e-mail, voice, chat, etc.
est_duration	integer	no	The estimated service duration in seconds.
timestamp	date/time	no	<p>The UTC time at which the event was raised, with a precision of milliseconds, using the ISO 8601<ref name="ISO 8601">http://en.wikipedia.org/wiki/ISO_8601</ref> representation : [YYYY] - [MM] - [DD]T[HH]:[mm]:[ss].[SSS]Z. If the application does not specify this timestamp, the server does it when the event is processed.</p>
<extension n> Supported since 8.0.2	Extension (single-valued) or Extension[] (multi-valued)	no	Service extensions, where <extension n> corresponds to the unique name of the service extension. Your application can specify as many extensions as needed as long as they were previously defined with the Create Service Extension Schema operation.

<references />

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response	
HTTP code	201
HTTP message	Created
Header	Location: \${base_uri}/service/\${service_id} where: <ul style="list-style-type: none">• \${base_uri} is the URI of the created service.• \${service_id} is the service ID.
Body	{"service_id": "\${service_id}"} where: <ul style="list-style-type: none">• \${service_id} is the service ID.

Example

Operation

```
POST /services/start
{
  "timestamp": "2009-05-12T12:05:12.145Z",
  "interaction_id": "123ABCAADFJ1259ACF",
  "application_type": 400,
  "application_id": 40,
  "est_duration": 60,
  "contact_key": "42",
  "service_type": 100,
  "media_type": 1,
  "resource_id": 5005,
  "resource_type": 2,
  "disposition": 10,
  "coupon": {
    "coupon_name": "DISCOUNTCODE15"
  },
  "satisfaction": {
    "score": 85,
    "agentID": 2025
  },
  "relatedOffers": [
    {
      "offer_name": "VIP credit card black ed.",
      "type": 9,
      "comments": "proposed to all client"
    },
  ],
}
```

Start Service

```
{
  {
    "offer_name": "3 times payment GOLD",
    "type": 4,
    "comments": "limited offer"
  },
  {
    "offer_name": "life insurance",
    "type": 3,
    "comments": "healt check to be done before approval"
  }
}
```

Result

```
HTTP 201 Created
location: /services/12349857
{"service_id": 12349857}
```