



Totalmobile

Connect Web Services Overview

V1.34

Version Control

Document Version	Version Date	Changed By	Reason
1.25	21/03/2019	Stuart Watt	Added section for coding examples
1.26	20/05/2019	Stuart Watt	New method added for job code creation
1.26	15/11/2019	Stuart Watt	Added OrderContact Object and ContactPreferences
1.26	20/01/2020	Stuart Watt	Added ResourceExternalId to Booking and Availability
1.26	02/03/2020	Stuart Watt	Added AvailabilityByTrade API Method
1.3	29/06/2020	Ryan Morris	Added draft PurchaseInvoice method
1.31	21/09/2020	Aaron Bloom	Added new customer details field to sales order method
1.32	24/02/2021	Stuart Watt	Added Customer method
1.33	03/03/2021	Aaron Bloom	Changes to Customer method
1.34	31/03/2021	Aaron Bloom	Updated field requirements on Customer, Contact and Address objects

Contents

Version Control	2
Contents	3
1. Summary	6
1.1 Overview	6
1.2 Jump URL	7
1.3 Embedded booking screen	9
1.3.1 Embedded booking screen: detailed tenant portal example	10
2.0 Web API	14
Address Object	14
Appointment Object	14
Attribute (GET)	15
Attribute (POST)	16
Attribute Object	17
AttributeItem Object	18
Authorisation Token	18
Availability (GET)	20
Availability Object	21
AvailabilityStatus Object	22
AvailabilityDays Object	23
AvailabilityPlanningWindow Object	23
AvailabilityResponse Object	23
AvailabilityByTrade (POST)	24
AvailabilityByTrade Object	25
AvailabilityByTradeResponse Object	26
Booking (POST)	26
Booking Object	27
BookingResponse Object	29
BookSubContractorOrder (POST)	30
BookSubContractorOrder (PATCH)	31
BookSubContractorOrder Object	32
BookSubContractorOrderResponse Object	32
BookSubContractorOrderUpdate Object	33
BookSubContractorOrderUpdateCode Object	33
BookingSummary (GET)	34

BookingSummary Object.....	35
BookingSummaryResponse Object	35
Code (POST)	35
Code (PUT)	37
Code (DELETE).....	38
Code (DELETE) by LocationExternalId	39
Code (GET) All	40
Code (GET) Single	41
Code (GET) Single By LocationExternalId	42
Code Object.....	43
Contact Object.....	44
ContactPreference - Enumerated Type.....	44
ContractorType - Enumerated Type	45
Customer (POST)	45
CustomerModel Object.....	46
CustomerAddress Object	48
CustomerContact Object	49
Customer Object.....	51
Error Object	51
IdLookup (GET)	52
Invoice Address	53
JobCode (GET)	54
JobCode (POST)	55
JobCode (PUT).....	55
JobCode Object.....	57
MessagingPreference - Enumerated Type.....	58
NameValuePair	59
OrderContact Object.....	59
OrderContractorType - Enumerated Type.....	60
OrderSource - Enumerated Type	60
OrderStatus - Enumerated Type	60
PlanningWindow (GET)	60
PlanningWindow Object	61
PurchaseInvoice (POST).....	62
PurchaseInvoice Object.....	63
PurchaseInvoiceLine object.....	66
PurchaseInvoiceCategory – Enumerated Type.....	66
PaymentTerms – Enumerated Type	66
RateBook (GET) All	67
RateBook (GET) Single	68

RateBook (PUT)	69
RateBook Object	70
ResponseCallback (POST)	70
SalesOrder (GET)	71
SalesOrder (PATCH)	73
SalesOrder (POST)	74
SalesOrder Object	75
SalesOrderUpdate Object	78
UserAccount Object	79
UserAccount (POST)	80
VisitResultType - Enumerated Type	81
VisitStatus - Enumerated Type	81
3.0 Code Examples	81
3.1 Appointment Booking	82

1. Summary

1.1 Overview

This document describes the core Connect web services integration.

This integration is focused on a CRM or Housing management system integrating with the Connect repairs and maintenance system.

The document also describes the use of the web services for tenant self service repairs bookings and subcontract portal updates.

There are two primary approaches to the integration:

Approach	Summary	Advantages / Disadvantages
Jump URL	Enter order in external system and call the order create webservice to create the order in Connect. Execute a jump URL (via a secure token) which brings up the Connect booking screen to book the appointment. Note: Any updates to the order and appointments directly in Connect can be asynchronously sent to the external system.	+ Technically easier and quicker to get working + User has all the facilities of the Connect booking options - Appears to user as two different systems
Embedded booking screen	Enter order in external system and call the order create webservice to create the order in Connect. Call the check availability web service and display the results in the external system.	+ User only aware of external system + Better interface for tenant self service

	User selects the appointment slot and the appointment is booked by calling the create appointment webservice.	-Technically harder to get working - User does not have all the facilities of the Connect booking options
--	---	--

1.2 Jump URL

This section describes the business flow for a tenant booking repairs via the jump URL.

The basic process is as follows:

- Authentication process for the web service sign on (OAUTH2) *see [Authorisation Token](#)*
- Obtain URL token
- Order raised in external system
- The repair is diagnosed
 - Obtain any descriptive information
 - Identify work required (Diagnosis code / SOR codes)
- Order created in Connect *see [SalesOrder](#)*
- Execute jump URL ideally in an iframe or the user can move to a separate tab

The standard Connect booking screen is displayed.

Appointment Slots							
	Monday 07 May 2018	Tuesday 08 May 2018	Wednesday 09 May 2018	Thursday 10 May 2018	Friday 11 May 2018	Saturday 12 May 2018	Sunday 13 May 2018
ALL DAY 08:00-16:59				29	29		
MORNING 08:00-12:59				29	29		
AFTERNOON 12:00-16:59				29	29		
SCHOOL RUN 09:30-14:29				29	29		
OUT OF HOURS 17:00-23:59							

- User selects preferred appointment slot in Connect screen
- Appointment booked and confirmed
- Results passed back asynchronously by calling a web service supplied by the external system. Exactly what is required will vary.

The following events are currently returned if required:

- Order completion
- New appointments
- Re-appointments
- Follow on appointments
- Completed visits
- Deleted visits
- Shelved visits

The business flow also needs to support cancelling orders which can be done via the Connect order cancel web service. See [SalesOrder](#)

Notes:

There are 3 URLs available

- Booking screen - to book a new appointment
- List of orders - to review other jobs at the property
- List of appointments for an order - to re-arrange or cancel an appointment

1.3 Embedded booking screen

This section describes the business flow for the embedded booking screen.

The basic process is as follows:

- The contact details are verified (if required)
- The repair is diagnosed
 - Obtain any descriptive information
 - Identify work required (Diagnosis code / SOR codes)
- Authentication process for the web service sign on (OAUTH2) *see [Authorisation Token](#)*
- Order created in Connect *see [SalesOrder](#)*

- Check if the order is fulfilled internally (DLO) or externally (contractor) from the Diagnosis tree
- If INTERNAL
 - Check appointment availability (using Diagnosis code / SOR codes) *see [Availability](#)*
 - User selects preferred appointment
 - Appointment booked and confirmed *see [Booking](#)*
- If EXTERNAL
 - Book a Contractor Order - *See [BookSubcontractorOrder](#)*

The business flow also needs to support:

- Emailing for work not available via diagnosis
- Reviewing appointments already booked *see [BookingSummary](#)*
- Changing appointment dates
- Cancelling orders *see [SalesOrder](#)*

1.3.1 Embedded booking screen: detailed tenant portal example

User log on

User logs on to tenant portal

External system verifies the contact details and confirms whether text messages are required:

Customer Details

Name

TEST CUSTOMER

Mobile Phone

123

Home Phone

123

Home Email

Home Email

Repairs is diagnosed (example only)

Diagnosis Search	Q Search...
Diagnosis Category	Other
Code Template	Front door
Out Of Hours	☐

In the example above there is a simple two level diagnosis structure:

Diagnosis	Sub Area	Code
Baths	Bath leak	BATH001
Baths	Bath loose	BATH002
Baths	Bath panel loose or broken	BATH003
Baths	Grab bar loose	BATH004
Baths	Grab rail required in bathroom	BATH005

The external system creates the works order in Connect by calling the Connect Web Service method [SalesOrder](#).

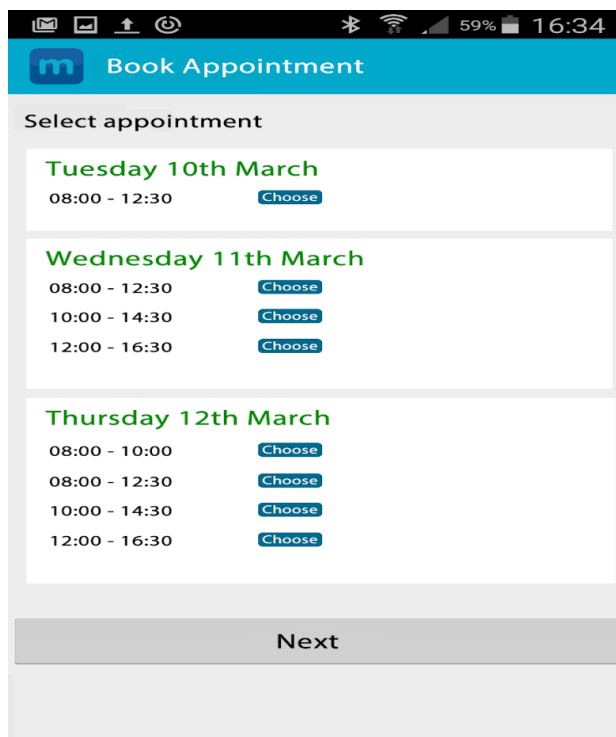
The works order is created for the customer as if it had been entered by the Order Entry screen in Connect.

Note: Use of SOR codes instead of Diagnosis code is also supported. The Diagnosis route allows the same booking logic as is supported in Connect.

Booking an appointment

The external system then calls the Connect Web Service method [Availability](#) and a list of available slots is returned

These slots can be displayed as required in the external system. The screenshot below is just an example.



The screenshot shows a mobile application interface for booking an appointment. At the top, there is a status bar with icons for mail, calendar, upload, and camera, along with system icons for Bluetooth, Wi-Fi, signal strength, 59% battery, and the time 16:34. Below the status bar is a blue header with a white 'm' logo and the text 'Book Appointment'. The main content area is titled 'Select appointment' and displays three sections for different days: 'Tuesday 10th March', 'Wednesday 11th March', and 'Thursday 12th March'. Each section lists available time slots with a 'Choose' button next to each. For Tuesday, the slot is 08:00 - 12:30. For Wednesday, the slots are 08:00 - 12:30, 10:00 - 14:30, and 12:00 - 16:30. For Thursday, the slots are 08:00 - 10:00, 08:00 - 12:30, 10:00 - 14:30, and 12:00 - 16:30. At the bottom of the form is a grey button labeled 'Next'.

Day	Time Slot	Action
Tuesday 10th March	08:00 - 12:30	Choose
Wednesday 11th March	08:00 - 12:30	Choose
	10:00 - 14:30	Choose
	12:00 - 16:30	Choose
Thursday 12th March	08:00 - 10:00	Choose
	08:00 - 12:30	Choose
	10:00 - 14:30	Choose
	12:00 - 16:30	Choose

Next

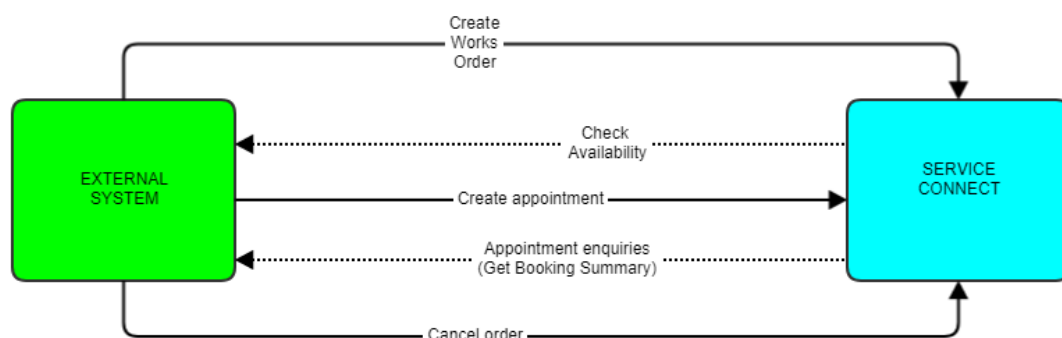
Creating the appointment

The user then selects an appointment slot.

The external system creates the appointment in Connect by calling the Connect Web Service method [Booking](#).

The external system displays a confirmation message to the user

The diagram below shows the basic integration flow for booking an appointment:



Emailing for work not available via diagnosis - this would be handled outside of Connect.

Reviewing appointments - a list of appointments is displayed in the external system. This can be retrieved by calling the Connect Web Service method [BookingSummary](#).

Any appointments at a status of booked not due within the next 'n' days can be re-appointed

Re-arranging an appointment - an appointment can be re-arranged by calling the Connect Web Service method [Availability](#) and a revised appointment slot booked and then calling the Connect Web Service method [Booking](#).

Cancelling an order / appointment - an order can be cancelled by calling the Connect Web Service method [SalesOrder](#).

2.0 Web API

The Web API help pages can be found at the following URL with the username and password provided separately:

<https://api-ga-customer.clouddialogs.com/WS-API>

Address Object

Name	Type	Requirement for POST	Description	Version
ExternalId	String (64)		The external id of the address	1
Line1	String (128)		The first line of the address.	1
Line2	String (128)		The second line of the address.	1
Line3	String (128)		The third line of the address	1
City	String (64)		The city of the address	1
PostCode	String (48)		The postcode of the address	1
Region	String (64)		The region of the address	1

Appointment Object

Name	Type	Description	Version
PrimaryOrderNumber	String	A string which identifies the order.	1

OrderRaisedDate	Date / Time yyyy-MM-dd	The order raised date.	1
VisitExternalId	String	A string which identifies the visit.	1
VisitResultType	VisitResultType	The visit result type.	1
VisitResultTypeName	String	The visit result type name.	1
VisitStatus	VisitStatus	The visit status.	1
VisitStatusName	String	The visit status name.	1
Resource	String	The name of the resource.	1
VisitInstructions	String	The visit instructions.	1
OperativeComments	String	The operative comments.	1
PlanningWindowStart	Date / Time yyyy-MM-dd	The planning window start.	1
PlanningWindowEnd	Date / Time yyyy-MM-dd	The planning window end.	1

Attribute (GET)

GET api/SalesOrder/{orderId}/Attribute

Method :

Name	Method	Description	Version
Attribute	GET	This method returns a the array of attributes for an order.	3

Parameters:

Name		Description	From
OrderId	Integer	An internal integer which identifies the order.	Uri

Status Codes:

Code	Status
200	OK
400	Bad Request

A successful response will be in the following format:

[Attribute](#)

An unsuccessful response will be in the following format:

[Error](#)

Attribute (POST)

POST api/SalesOrder/{orderId}/Attribute

Method :

Name	Method	Description	Version
Attribute	POST	This method allows scheduling attributes to be set on an order.	3

Parameters:

Name	Type	Description	From	Version
OrderId	Integer	An internal integer which identifies the order.	Uri	3
Attributes	Array of AttributeItem	An array of attributes for the order.	Body	3

Status Codes:

Code	Status
200	OK
400	Bad Request

A successful response will be in the following format:

```
{}
```

An unsuccessful response will be in the following format:

[Error](#)

Attribute Object

Name	Type	Requirement for POST	Description	Version
OrderId	Integer		An internal integer which identifies the order.	3
Attribute	Array of AttributeItem		An array of attributes for the order.	3

AttributeItem Object

Name	Type	Requirement for POST	Description	Version
AttributeName	String		A string which identifies the attribute.	3
Value	String		A string identifying the value.	3

Authorisation Token

Method :

Name	Method	Description
Token	POST	This method allows for the retrieval of an authorisation token which is used when making subsequent API calls.

Header:

Name	Value
Content-Type	application/x-www-form-urlencoded

Body:

Name	Description
grant_type	The grant type to use, usually set to 'password'
username	The user name provided
password	The password provided

Response:

Name	Description
access_token	The token which is used in subsequent API calls.

token_type	The token type, usually 'bearer'
expires_in	When the token will expire in seconds
userName	The user name for the token
.issued	When the token was issued
.expires	When the token will expire

Availability (GET)

Method:

Name	Method	Description	Version
Availability	GET	This method will check the availability for booking appointments.	1

Parameters:

[Availability](#)

Status Codes:

Code	Status
200	OK
400	Bad Request

A successful response will be in the following format:

[AvailabilityResponse](#)

An unsuccessful response will be in the following format:

[Error](#)

Availability Object

Name	Type	Requirement for POST	Description	Version
PrimaryOrderNumber	String	Required	A string which identifies the order.	1
Date	Date / Time yyyy-MM-dd		The current date. The response will include the whole week that the date belongs to.	1
IgnoreWeekends	Boolean		Allows bookings at weekends.	1
Duration	Integer		A custom duration - reserved for future use	1
VisitExternalId	String		A string which identifies the visit. This is used when reappointing a visit.	1
TwoToAttend	Boolean		Specifies that paired working is required - reserved for future use	1

LimitByPreferredSkills	Boolean		Limit the selection by preferred skill - reserved for future use	1
LimitByPreferredZones	Boolean		Limit the selection by preferred zone - reserved for future use	1
LimitByValidZoneDays	Boolean		Limit the selection by valid zone days - reserved for future use	1
OverrideTargetDate	Boolean		Override target date - reserved for future use	1
OverrideTargetDateReason	String		Override target date reason - reserved for future use	1
NumberOfWeeks	Integer		Set the number of weeks to return, 1-5 weeks, defaults to 1 week.	1
ResourceExternalId	String		The Resource External Id	3

AvailabilityStatus Object

Name	Value	Version
NotAvailable	0	1
Available	1	1

Preferred	2	1
-----------	---	---

AvailabilityDays Object

Name	Type	Description	Version
Date	Date / Time yyyy-MM-dd	The date.	1
AvailabilityStatus	AvailabilityStatus	The availability status.	1
TravelTime	Integer	The travel time in minutes.	1

AvailabilityPlanningWindow Object

Name	Type	Description	Version
PlanningWindowId	Integer	The id of the planning window.	1
AvailabilityDays	Array< AvailabilityDay >	This holds availability information for the 7 days of the current week.	1

AvailabilityResponse Object

Name	Type	Description	Version
Availability	Availability	The availability request information.	1
PlanningWindows	Array< PlanningWindo >	An array of planning	1

	w>	windows.	
AvailabilityPlanningWindows	Array< AvailabilityPlanningWindow >	An array of objects which hold availabilities by planning window.	1

AvailabilityByTrade (POST)

Method:

Name	Method	Description	Version
AvailabilityByTrade	POST	This method will check the availability for booking appointments by trade.	3

Parameters:

[AvailabilityByTrade](#)

Status Codes:

Code	Status
200	OK
400	Bad Request

A successful response will be in the following format:

[AvailabilityByTradeResponse](#)

An unsuccessful response will be in the following format:

[Error](#)

AvailabilityByTrade Object

Name	Type	Description	Version
Date	Date / Time yyyy-MM-dd	The current date. The response will include the whole week that the date belongs to.	3
NumberOfWeeks	Integer	Set the number of weeks to return, 1-5 weeks, defaults to 1 week.	3
Skills	Array<String>	A list of skill names the resource must have.	3
Postcode		The postcode of the visit	3
SchedulingAttributes	Array< AttributeItem >	An array of attributes for the order.	3
TargetDate	Date / Time yyyy-MM-dd	The target date for the job	3
TwoToAttend	Boolean	Specifies that paired working is required	3
Duration	Integer	The duration of the visit	3

AvailabilityByTradeResponse Object

Name	Type	Description	Version
Availability	AvailabilityByTrade	The availability by trade request information.	3
PlanningWindows	Array< PlanningWindow >	An array of planning windows.	3
AvailabilityPlanningWindows	Array< AvailabilityPlanningWindow >	An array of objects which hold availabilities by planning window.	3

Booking (POST)

Method:

Name	Method	Description	Version
Booking	POST	This method will allow for the booking of an appointment.	1

Body:

[Booking](#)

Status Codes:

Code	Status
200	OK
400	Bad Request

A successful response will be in the following format:

[BookingResponse](#)

An unsuccessful response will be in the following format:

[Error](#)

Booking Object

Name	Type	Requirement for POST	Description	Version
PrimaryOrderNumber	String	Required	A string which identifies the order.	1
Date	Date / Time yyyy-MM-dd	Required	The current date. The response will include the whole week that the date belongs to.	1
PlanningWindowId	Integer	Required	The id of the planning window.	1
IgnoreWeekends	Boolean		Allows bookings at weekends.	1

Duration	Integer		Set a custom duration for the appointment.	1
TwoToAttend	Boolean		Specifies that paired working is required.	1
VisitExternalId	String		A string which identifies the visit. This is used when reappointing a visit.	1
VisitInstructions	String		Visit instructions.	1
MessagingPreference	MessagingPreference		The messaging preference.	1
ShelveAppointment	bool		Determines whether to create a shelved appointment.	1
LimitByPreferredSkills	Boolean		Limit the selection by preferred skill - reserved for future use	1
LimitByPreferredZones	Boolean		Limit the selection by preferred zone - reserved for future use	1
LimitByValidZoneDays	Boolean		Limit the selection by valid zone days - reserved for future use	1
OverrideTargetDate	Boolean		Override target date - reserved for future use	1

OverrideTargetDateReason	Date / Time yyyy-MM-dd		Override target date reason - reserved for future use	1
ResourceExternalId	String		The Resource External Id	3

BookingResponse Object

Name	Type	Description	Version
PrimaryOrderNumber	String	A string which identifies the order.	1
Date	Date / Time yyyy-MM-dd	The date of the booking.	1
PlanningWindow	PlanningWindow	The planning window.	1
Duration	Integer	The duration of the appointment in minutes.	1
IgnoreWeekends	Boolean	Allows bookings at weekends.	1
TwoToAttend	Boolean	Specifies that paired working is required.	1
VisitExternalId	String	A string which identifies the visit. This is used when reappointing a visit.	1
VisitInstructions	String	Visit instructions.	1

MessagingPreference	MessagingPreference	The messaging preference.	1
LimitByPreferredSkills	Boolean	Limit the selection by preferred skill.	1
LimitByPreferredZones	Boolean	Limit the selection by preferred zone.	1
LimitByValidZoneDays	Boolean	Limit the selection by valid zone days.	1
OverrideTargetDate	Boolean	Override target date.	1
OverrideTargetDateReason	String	Override target date reason.	1

BookSubContractorOrder (POST)

Method:

Name	Method	Description	Version
BookSubContractorOrder	POST	This method allows the booking of a sub contractor order.	1

Parameters:

[BookSubContractorOrder](#)

Status Codes:

Code	Status
200	OK

400	Bad Request
-----	-------------

A successful response will be in the following format:

[BookSubContractorOrderResponse](#)

An unsuccessful response will be in the following format:

[Error](#)

BookSubContractorOrder (PATCH)

Method:

Name	Method	Description	Version
BookSubContractorOrder	PATCH	This method updates a sub contractor order.	3

Parameters:

[BookSubContractorOrderUpdate](#)

Status Codes:

Code	Status
------	--------

200	OK
400	Bad Request

A successful response will be in the following format:

[BookSubContractorOrderUpdate](#)

An unsuccessful response will be in the following format:

[Error](#)

BookSubContractorOrder Object

Name	Type	Required	Description	Version
PrimaryOrderNumber	String	Required	A string which identifies the order.	1
SupplierExternalId	String	Required	An string which identifies the supplier.	1

BookSubContractorOrderResponse Object

Name	Type	Description	Version
PrimaryOrderNumber	String	A string which identifies the order.	1

SupplierExternalId	String	An string which identifies the supplier.	1
SupplierName	String	The supplier name.	1

BookSubContractorOrderUpdate Object

Name	Type	Description	Version
PrimaryOrderNumber	String	A string which identifies the order.	3
SubcontractOrderId	Integer	An integer which identifies the subcontract order.	3
CustomerExternalId	String	A string which identifies the customer.	3
CategoryExternalId	String	A string which identifies the category.	3
Comments	String	A comment which describes the reason for the update.	3
Codes	Array of BookSubContractorOrderUpdateCode	The order codes associated with this update.	3

BookSubContractorOrderUpdateCode Object

Name	Type	Description	Version
CodeExternalId	String	A string which identifies the order code.	3

LocationExternalId	String	A string which identifies the location.	3
Quantity	Decimal	The quantity.	3

BookingSummary (GET)

Method:

Name	Method	Description	Version
BookingSummary	GET	This method will retrieve appointments for a given property.	1

Parameters:

[BookingSummary](#)

Status Codes:

Code	Status
200	OK
400	Bad Request

A successful response will be in the following format:

[BookingSummaryResponse](#)

An unsuccessful response will be in the following format:

[Error](#)

BookingSummary Object

Name	Type	Requirement for POST	Description	Version
PropertyId	String	Required if PrimaryOrderNumber not set	A string which identifies the property.	1
PrimaryOrderNumber	String	Required if PropertyId not set	A string which identifies the order.	1
WindowStart	Date / Time yyyy-MM-dd HH:mm:ss		The window start date.	1
WindowEnd	Date / Time yyyy-MM-dd HH:mm:ss		The window end date.	1

BookingSummaryResponse Object

Name	Type	Description	Version
Customer	Customer	The customer details.	1
Appointments	Array< Appointment >	An array of appointments.	1

Code (POST)

POST api/SalesOrder/{orderId}/Code

Method:

Name	Method	Description	Version
Code	POST	This method allows a code to be added to a Sales Order.	3

Parameters:

Name	Type	From
OrderId	Integer	Uri
Code	Code	Body

Status Codes:

Code	Status
200	OK
400	Bad Request

A successful response will be in the following format:

{}

An unsuccessful response will be in the following format:

[Error](#)

Code (PUT)

PUT api/SalesOrder/{orderId}/Code

Method:

Name	Method	Description	Version
Code	PUT	This method allows a code to be updated for a Sales Order.	3

Parameters:

Name	Type	From
OrderId	Integer	Uri
Code	Code	Body

Status Codes:

Code	Status
200	OK
400	Bad Request

A successful response will be in the following format:

```
{}
```

An unsuccessful response will be in the following format:

[Error](#)

Code (DELETE)

DELETE api/SalesOrder/{orderId}/Code/{id}

Method:

Name	Method	Description	Version
Code	DELETE	This method allows a code to be removed from a Sales Order.	3

Parameters:

Name	Type	From
OrderId	Integer	Uri
Id	String	Uri

Status Codes:

Code	Status
200	OK
400	Bad Request

A successful response will be in the following format:

}

An unsuccessful response will be in the following format:

[Error](#)

Code (DELETE) by LocationExternalId

DELETE api/SalesOrder/{orderId}/Code/{id}/Location/{locationExternalId}

Method:

Name	Method	Description	Version
Code	DELETE	This method allows a code to be removed from a Sales Order.	3

Parameters:

Name	Type	From
OrderId	Integer	Uri
Id	String	Uri
LocationExternalId	String	Uri

Status Codes:

Code	Status
------	--------

200	OK
400	Bad Request

A successful response will be in the following format:

```
{}
```

An unsuccessful response will be in the following format:

[Error](#)

Code (GET) All

GET api/SalesOrder/{orderId}/Code

Method:

Name	Method	Description	Version
Code	GET	This method retrieved all codes for a Sales Order.	3

Parameters:

Name	Type	From
OrderId	Integer	Uri

Status Codes:

Code	Status
200	OK
400	Bad Request

A successful response will be in the following format:

Array of [Code](#)

An unsuccessful response will be in the following format:

[Error](#)

Code (GET) Single

GET api/SalesOrder/{orderId}/Code/{id}

Method:

Name	Method	Description	Version
Code	GET	This method retrieves a code for a Sales Order.	3

Parameters:

Name	Type	Description	From
OrderId	Integer	The internal id identifying an order.	Uri
Id	String	The external id identifying the code.	Uri

Status Codes:

Code	Status
200	OK
400	Bad Request

A successful response will be in the following format:

[Code](#)

An unsuccessful response will be in the following format:

[Error](#)

Code (GET) Single By LocationExternalId

GET api/SalesOrder/{orderId}/Code/{id}/Location/{locationExternalId}

Method:

Name	Method	Description	Version
Code	GET	This method retrieves a code for a Sales Order.	3

Parameters:

Name	Type	Description	From
OrderId	Integer	The internal id identifying an order.	Uri
Id	String	The external id identifying the code.	Uri
LocationExternalId	String	A string identifying the location.	Uri

Status Codes:

Code	Status
200	OK
400	Bad Request

A successful response will be in the following format:

[Code](#)

An unsuccessful response will be in the following format:

[Error](#)

Code Object

Name	Type	Description	Requirement	Version
CodeExternalId	String	A string which identifies the code.	Required	3
LocationExternalId	String	A string which identifies the location.	Required if Sales Order is Split By Job Code Locations	3

Quantity	Decimal	The quantity	Required For POST, PUT	3
----------	---------	--------------	---------------------------	---

Contact Object

Name	Type	Requirement for POST	Description	Version
ExternalId	String (64)		A string which identifies a contact in the system.	1
Address	Address			1
FirstName	String (64)		The first name.	1
LastName	String (64)		The last name.	1
HomePhone	String (32)		The home phone.	1
WorkPhone	String (32)		The work phone.	1
MobilePhone	String (32)		The mobile phone.	1
EmailAddress	String (64)		The email address.	1

ContactPreference - Enumerated Type

Name	Value	Version
None	0	3
EmailOnly	1	3
MobileTextOnly	2	3

HomeTextOnly	3	3
EmailAndMobileText	4	3
EmailAndHomeText	5	3

ContractorType - Enumerated Type

Name	Value	Version
Internal	0	1
External	1	1

Customer (POST)

Method:

Name	Method	Description	Version
Customer	POST	This method allows for the creation of a customer record	4

Header:

Name	Value	Description
Content-Type	application/json	
Authorization	Bearer <access_token>	The access token retrieved using the Token method.
Accept	application/json; version=4	

Body:

CustomerModel

Status Codes:

Code	Status
201	Created
400	Bad Request

A successful response for Version 4 will return:

CustomerModel

An unsuccessful response will be in the following format:

Error

CustomerModel Object

Name	Type	Requirement for POST	Description	Version
ExternalId	String (64)	Required	A string which identifies the property.	4
Name	String (64)	Required	The name of the customer	4
ParentExternalId	String (64) Must be an existing Customer External ID			4

CustomerAddress	<u>CustomerAddress</u>	Required	The property address details	4
CustomerContact	<u>CustomerContact</u>	Required	The contact details	4
OrganisationExternalId	String (64) Must be valid Organisation in Connect	Required	The external id of the organisation	4
TenancyCommencementDate	Date / Time yyyy-MM-dd		The tenancy commencement date	4
TenancyType	String (64) Must be valid Tenancy Type in Connect		The tenancy type of the property.	4
PairedWorkingRequired	Boolean		Paired working required flag	4
ContractArea	String (128) Must be valid Contract Area in Connect		The contract area	4
ExcludeForDecency	Boolean		Exclude for decency flag	4
ExcludeForDecencyReasonExternalId	String (64)		The exclude for decency reason external id	4
RepairingResponsibilityExternalId	String (64)		The repairing responsibility external id	4

LinkedPropertyForDecencyExternalId	String (64)		The linked property for decency	4
Active	Boolean		The active flag for the property	4
CustomFields	<u>Array<NameValuePair></u> Must be valid custom fields in Connect		Custom fields for the property	4

CustomerAddress Object

Name	Type	Requirement for POST	Description	Version
ExternalId	String (64)	Required	The external id of the address	4
Address1	String (128)	Required	The first line of the address.	4
Address2	String (128)		The second line of the address.	4
Address3	String (128)		The third line of the address	4
City	String (64)	Required	The city of the address	4
PostCode	String (48)	Required	The postcode of the address	4
Region	String (64)		The region of the address	4
PropertyType	String (64)		The property type	4

	Must be valid Property Type in Connect			
OwnershipType	String (64) Must be valid Ownership Type in Connect		The ownership type	4
PropertyScheme	String (64) Must be valid Property Scheme in Connect		The property scheme	4
PropertyWarningDescription	String (4000)		The property warning description	4
AccessNotes	String (2048)		The access notes	4
BuildDate	String (64)		The build date	4
HousingAreaExternalId	String (128) Must be valid Housing Area in Connect		The housing area external id	4
NumberOfBedrooms	String (8)		The number of bedrooms in the property	4

CustomerContact Object

Name	Type	Requirement for POST	Description	Version
------	------	----------------------	-------------	---------

ExternalId	String (64)	Required	The external id of the contact.	4
FirstName	String (64)	Required	The first name of the contact.	4
LastName	String (64)	Required	The last name of the contact.	4
Salutation	String (20)		The salutation of the contact.	4
HomePhone	String (32)		The home phone number of the contact.	4
WorkPhone	String (32)		The work phone number of the contact.	4
MobilePhone	String (32)		The mobile phone number of the contact.	4
HomeEmail	String(64)		The home email address of the contact.	4
WorkEmail	String(64)		The work email address of the contact.	4
ContactPreference	<u>ContactPreference</u>		The contact preference of the contact.	4

MessagingPreference	MessagingPreference		The send message preference of the contact.	4
ContactWarningDescription	String (4000)		The contact warning description.	4

Customer Object

Name	Type	Requirement for POST	Description	Version
ExternalId	String [64]	Required	A string which identifies a customer in the system.	1
OrganisationExternalId	String [64]	Required	A string which identifies an organisation in the system.	1
Name	String [64]		The customer name.	1
LandlordReference	String [128]		The landlord reference.	1
PrimaryContact	Contact	Required	The primary contact.	1
PrimaryAddress	Address	Required	The primary address.	1
InvoiceAddress	InvoiceAddress		The invoice address.	1

Error Object

Name	Description	Version
Message	The error message.	1
ModelState	Information about the error.	1

IdLookup (GET)

Method :

Name	Method	Description	Version
IdLookup	GET	This method returns a internal integer identifier.	3

Parameters:

Lookup	Parameter	Type	Parameter	Type
OrderId	PrimaryOrderNumber	String	ContractExternalId	String

Status Codes:

Code	Status
200	OK
400	Bad Request

A successful response will be in the following format:

Integer

An unsuccessful response will be in the following format:

[Error](#)

Invoice Address

Name	Type	Requirement for POST	Description	Version
ContactName	String		The contact name.	1
ContactMobileNumber	String		The mobile number.	1
ContactLandlineNumber	String		The landline number.	1
ContactEmailAddress	String		The email address.	1
Line1	String [128]	Required	The first line of the address.	1
Line2	String [128]		The second line of the address.	1
Line3	String [128]		The third line of the address.	1
City	String [64]		The city.	1
Region	String [64]		The region.	1
PostCode	String [48]		The postcode.	1
Country	String [48]		The country.	1
Zone	String [48]		The zone.	1

JobCode (GET)

GET api/JobCode/{id}

Method:

Name	Method	Description	Version
JobCode	GET	This method retrieves a JobCode by CodeExternalId.	3

Parameters:

Name	Type	From
Id	Integer	Uri

Status Codes:

Code	Status
200	OK
400	Bad Request

A successful response will be in the following format:

[JobCode](#)

An unsuccessful response will be in the following format:

[Error](#)

JobCode (POST)

POST api/JobCode/

Method:

Name	Method	Description	Version
JobCode	POST	This method creates a new JobCode.	3

Body:

[JobCode](#)

Status Codes:

Code	Status
200	OK
400	Bad Request

An unsuccessful response will be in the following format:

[Error](#)

JobCode (PUT)

PUT api/JobCode/{id}

Method:

Name	Method	Description	Version
JobCode	PUT	This method updates a JobCode by CodeExternalId.	3

Parameters:

Name	Type	From
Id	Integer	Uri
JobCode	JobCode	Body

Status Codes:

Code	Status
200	OK
400	Bad Request

A successful response will be in the following format:

[JobCode](#)

An unsuccessful response will be in the following format:

[Error](#)

JobCode Object

Name	Type	Requirement	Description	Version
CodeExternalId	String	Required	A string which identifies the job code (SOR code).	1
Description	String		A long description of the job code.	1
ShortDescription	String		A short description of the job code.	1
ContractExternalId	String		A string which identifies the contract.	1
WorkTypeExternalId	String		The work type external id.	1
WorkCategoryExternalId	String		The work category external id.	1
SortCode	String		The sort code.	1
UnitOfMeasure	String		The unit of measure.	1
StandardMinuteValue	Integer		The standard minute value.	1
ResourceCategoryExternalId	String		The resource category external id.	1
ResourceSubCategoryExternalId	String		The resource sub category external id.	1

SkillExternalId	String		The skill external id.	1
Quantity	Decimal		The quantity.	1
StandardPrice	Decimal		The standard price.	1
StandardCost	Decimal		The standard cost.	1
Location	String		The location.	1
Planned	Boolean		The planned flag.	1
CostCentre	String		The cost centre.	2
AccountCode	String		The account code.	2
SorSequenceNumber	String		The SOR sequence number.	2
TradeCode	String		The trade code.	3
IsPricingOnly	Boolean		This flags the code as pricing item only.	3
Capital	Boolean		The capital flag.	3
DiagnosticCode	Boolean		The diagnostic code flag.	3

MessagingPreference - Enumerated Type

Name	Value	Version
0	Message	1

1	DontMessage	1
2	NeverMessage	1

NameValuePair

Name	Type	Requirement for POST	Description	Version
Name	String (64)		The name for the pair	4
Value	String (Max)		The value for the pair	4

OrderContact Object

Name	Type	Requirement for POST	Description	Version
ContactName	String		A string which identifies a contact in the system.	3
HomePhone	String		The home phone.	3
WorkPhone	String		The work phone.	3
MobilePhone	String		The mobile phone.	3
EmailAddress	String		The email address.	3
MessagingPreference	MessagingPreference		The messaging preference	3
ContactPreference	ContactPreference		The contact	3

ce	ce		preference	
----	--------------------	--	------------	--

OrderContractorType - Enumerated Type

Name	Value	Version
0	Internal	1
1	External	1

OrderSource - Enumerated Type

Name	Value	Version
Default	0	2
TenantPortal	1	2

OrderStatus - Enumerated Type

Name	Value	Version
None	0	1
Cancel	11	1

PlanningWindow (GET)

Method:

Name	Method	Description	Version
PlanningWindow	GET	This method will retrieve all planning windows.	1

Status Codes:

Code	Status
200	OK
400	Bad Request

A successful response will be in the following format:

Array<[PlanningWindow](#)>

An unsuccessful response will be in the following format:

[Error](#)

PlanningWindow Object

Name	Type	Description	Version
Id	Integer	The id of the planning window.	1
Name	String	The name of the planning window.	1
StartHour	Integer	The start hour.	1
StartMinute	Integer	The start minute.	1
EndHour	Integer	The end hour.	1
EndMinute	Integer	The end minute.	1

PurchaseInvoice (POST)

Method:

Name	Method	Description	Version
PurchaseInvoice	POST	This method allows for the creation of a matched purchase invoice	3

Header:

Name	Value	Description
Content-Type	application/json	
Authorization	Bearer <access_token>	The access token retrieved using the Token method.
Accept	application/json; version=3	

Body:

PurchaseInvoice

Status Codes:

Code	Status
201	Created
400	Bad Request

A successful response for Version 3 will return:

PurchaseInvoice

An unsuccessful response will be in the following format:

Error

PurchaseInvoice Object

Name	Type	Requirement for POST	Description	Version
SupplierExternalId	String	Required	A string which identifies the supplier.	3
PurchaseInvoiceCategory	PurchaseInvoiceCategory	Required	Category of the purchase invoice	3
SupplierInvoiceReference	String [64]	Required	A string to identify	3

			the supplier invoice	
InvoiceDateUTC	Date / Time yyyy-MM-dd	Required	The date of the invoice	3
PaymentTerms	PaymentTerms	Required	Payment terms of the purchase invoice	3
EarlySettlementDiscount	Boolean			3
EarlySettlementDiscountPercent	String		Percentage of discount for an early settlement on this purchase invoice	3
EarlySettlementDiscountValue	Decimal		Value of discount for an early settlement on this purchase invoice	3
EarlySettlementDate	Date / Time yyyy-MM-dd		Date settlement must be received for discount on this purchase invoice	3
GoodsValue	Decimal	Required	The value of goods	3

			for this purchase invoice	
DeliveryCost	Decimal		The cost of delivering the goods on this purchase invoice	3
OtherCosts	Decimal		Any other costs associated with this purchase invoice	3
VAT	Decimal		Amount of VAT paid on this purchase invoice	3
LabourCost	Decimal		Labour costs for this purchase invoice	3
Comments	String			3
PurchaseInvoiceLines	Array<PurchaseInvoiceLine>			3

PurchaseInvoiceLine object

Name	Type	Requirement for POST	Description	Version
CodeExternalId	String	Required	External ID of the code item	3
UnitCost	Decimal		Cost of the item per unit	3
LineCost	Decimal		Total cost of the purchase invoice line	3
Description	String			3
PrimaryOrderNumber	String[32]		A string which identifies an order in the system.	3
PurchasingCard Number	String[64]			3
ResourceExternalId	String		A string that identifies the resource that required the code item	3
Quantity	Integer			3
UOM	String		The unit of measure for the code item	3
DeliveryDate	Date / Time yyyy-MM-dd		Date the code item was delivered	3
PurchaseOrder Number	String	Required	A string that identifies the purchase order for the purchase invoice	3

PurchaseInvoiceCategory – Enumerated Type

Name	Value	Version
PurchaseOrder	0	3
SubcontractorOrder	1	3

PaymentTerms – Enumerated Type

Name	Value	Version
------	-------	---------

N/A	0	3
NetMonthly	1	3
NoDays	2	3

RateBook (GET) All

Method:

Name	Method	Description	Version
RateBook	GET	This method retrieves all RateBooks for a Job Code.	3

Parameters:

Name	Type
CodeExternalId	String

Status Codes:

Code	Status
200	OK
400	Bad Request

A successful response will be in the following format:

Array of [RateBook](#)

An unsuccessful response will be in the following format:

[Error](#)

RateBook (GET) Single

Method:

Name	Method	Description	Version
RateBook	GET	This method retrieves a single RateBooks for a Job Code and Contract	3

Parameters:

Name	Type
CodeExternalId	String
ContractExternalId	String

Status Codes:

Code	Status
200	OK
400	Bad Request

A successful response will be in the following format:

[RateBook](#)

An unsuccessful response will be in the following format:

[Error](#)

RateBook (PUT)

Method:

Name	Method	Description	Version
RateBook	PUT	This method updates a RateBooks for a Job Code and Contract	3

Parameters:

Name	Type	From
CodeExternalId	String	Uri
ContractExternalId	String	Uri
RateBook	RateBook	Body

Status Codes:

Code	Status
200	OK
400	Bad Request

A successful response will be in the following format:

[RateBook](#)

An unsuccessful response will be in the following format:

[Error](#)

RateBook Object

Name	Type	Required	Description	Version
CodeExternalId	String	Required	A string identifying the code.	3
ContractExternalId	String	Required	A string identifying the contract.	3
ContractName	String	Not Required	The contract name.	3
Price	Decimal	Required for Post	The price.	3

ResponseCallback (POST)

Method:

Name	Method	Description	Version
ResponseCallback	POST	This method accepts a response from an asynchronous method call to a third-party Web API.	2

Body:

Name	Type	Requirement for POST	Description	Version
CloudExternalId	String	Required	A string which identifies the original method call.	2
StatusCode	Integer	Required	The status code.	2
Message	String		The response message.	2

Status Codes:

Code	Status
200	OK
400	Bad Request

SalesOrder (GET)

Method:

Name	Method	Description	Version
SalesOrder	GET	This method allows for the retrieval of a sales order.	1

Header:

Name	Value	Description
Content-Type	application/json	

Authorization	Bearer <access_token>	The access token retrieved using the Token method.
Accept	application/json; version=2	

Parameters:

Name	Description
PrimaryOrderNumber	A string which identifies an order in the system.
ContractExternalId	A string which identifies a contract in the system.

Status Codes:

Code	Status
200	Success
404	Not Found

A successful response will be in the following format:

[SalesOrder](#)

An unsuccessful response will be NULL.

SalesOrder (PATCH)

Method:

Name	Method	Description	Version
SalesOrder	PATCH	This method allows for the update or cancellation of a sales order.	2

Header:

Name	Value	Description
Content-Type	application/json	
Authorization	Bearer <access_token>	The access token retrieved using the Token method.
Accept	application/json; version=2	

Body:

[SalesOrderUpdate](#)

Status Codes:

Code	Status
200	OK

400	Bad Request
-----	-------------

A successful response will be in the following format:

```
{}
```

An unsuccessful response will be in the following format:

[Error](#)

SalesOrder (POST)

Method:

Name	Method	Description	Version
SalesOrder	POST	This method allows for the creation of a sales order.	1

Header:

Name	Value	Description
Content-Type	application/json	
Authorization	Bearer <access_token>	The access token retrieved using the Token method.
Accept	application/json; version=2	

Body:

[SalesOrder](#)

Status Codes:

Code	Status
201	Created
400	Bad Request

A successful response for Version 2 will return:

```
{
```

A successful response for Version 3 will return:

[SalesOrder](#)

An unsuccessful response will be in the following format:

[Error](#)

SalesOrder Object

Name	Type	Requirement for POST	Description	Version
OrderCloudId	String		An external	1

			identifier.	
PrimaryOrderNumber	String [32]	Required	A string which identifies an order in the system.	1
SecondaryOrderNumber	String [32]		An optional string identifying an order.	1
UseExistingCustomerDetails	Boolean		When set to true, only an external ID and Organisation external ID can be provided for a customer and the rest is derived from the existing record	3
Customer	Customer	Required	The customer record.	1
OrderContact	OrderContact		The order contact	3
ContractExternalId	String [64]	Required	A string which identifies a contract in the system.	1
TargetDateUTC	Date / Time yyyy-MM-dd	Required	The target date for the order in Universal Time Coordinated format.	1
PriorityExternalId	String	Required	A string which identifies a priority in the system	1
OrderStatus	OrderStatus		The order status of the order.	1

LastVisitResultType	VisitResultType		The result type of the last visit	1
JobDescription	String	Required		1
JobCodes	Array< JobCode >			1
CancelReason	String			1
CancelDescription	String			1
NeverMessage	Boolean			2
TwoToAttend	Boolean			2
IsInspectionJob	Boolean			2
InvokeDiagnosis	Boolean		Determines whether to create order from diagnosis	2
PropertyWarningDescription	String [4000]			2
ContactWarningDescription	String [4000]			2
OrderSource	OrderSource		The order source.	2
CrmlId	String		The external CRM identifier for the order.	3
CodeTemplateId	Integer		The code template id - for diagnostics	3
ContractorType	ContractorType		The contractor type - for diagnostics	3

OutOfHours	Boolean		Out of hours - for diagnostics	3
WorkStream	String		The external workstream identifier.	3

SalesOrderUpdate Object

Name	Type	Requirement for POST	Description	Version
PrimaryOrderNumber	String [64]	Required	A string which identifies the order.	1
ContractExternalId	String [64]	Required	A string which identifies the contract.	1
OrderStatus	OrderStatus		The order status, set to 11 for cancellation.	1
JobDescription	String		The job description.	1
PriorityExternalId	String		A string which identifies the priority.	1
TargetDateUTC	Date / Time yyyy-MM-dd	Required (If PriorityExternalId is set)	The target date for the order in Universal Time Coordinated format.	1

CancelReason	String	Required (If OrderStatus = 11)	The reason for cancellation.	1
CancelDescription	String		A detailed description for cancellation.	1
JobCodes	Array< JobCodes >		The job codes (SORs) Not currently implemented	1

UserAccount Object

Name	Type	Requirement for POST	Description	Version
UserName	String	Required	The user name.	2
FirstName	String		The first name.	2
LastName	String		The last name.	2
Email	String		The email address.	2
ProfileName	String		The required profile name in the system.	2
Token	String		The authorisation token. Returned only.	2
Url	String		The redirect URL.	2

UserAccount (POST)

Method:

Name	Method	Description	Version
UserAccount	POST	This method allows the creation of a user account in the system from a third-party, and will generate a token for single sign-in.	2

Body:

[UserAccount](#)

Status Codes:

Code	Status
200	OK
400	Bad Request

A successful response will be in the following format:

[UserAccount](#)

An unsuccessful response will be in the following format:

[Error](#)

To use the token and log in to the account, the following URL format should be used, replacing the XXXXX with the token:

<https://s-url-test.clouddialogs.com/XXXXXXXXXXXXXXXXXXXXXXX>

VisitResultType - Enumerated Type

Name	Value	Version
Completed	0	2
NoAccess	1	2
FollowOnRequired	2	2
Abandon	3	2

VisitStatus - Enumerated Type

Name	Value	Version
Planned	0	2
Accepted	1	2
Started	2	2
Completed	3	2
PendingDeletion	4	2
Booked	5	2
Shelved	6	2

3.0 Code Examples

3.1 Appointment Booking

Based on setting up a QA environment where this can be tested

Authorisation

<https://api-ga-customer.clouddialogs.com/WS-API/token>

method: POST

ContentType: application/x-www-form-urlencoded

grant_type: password

username: <username>

password: <password>

username - This is usually in the format <account_name>@[clouddialogs.com](https://api-ga-customer.clouddialogs.com)

password - This will be supplied to the end user

Availability

<https://api-ga-customer.clouddialogs.com/WS-API/api/availability?PrimaryOrderNumber=<primaryordernumber>&Date=<date>>

method: GET

ContentType: application/json

Accept: application/json;version=3

Authorization: Bearer <token>

primaryordernumber - The Primary Order Number from Service Desk

date - in the following format yyyy-MM-dd

token - retrieved from the authorisation call above

Book

<https://api-ga-customer.clouddialogs.com/WS-API/api/booking>

method: POST

ContentType: application/json

Accept: application/json;version=3

Authorization: Bearer <token>

```
{  
  "PrimaryOrderNumber": "<primaryordernumber>",  
  "Date": "<date>",  
  "PlanningWindowId": <planningwindowid>  
}
```

primaryordernumber - The Primary Order Number from Service Desk

date - in the following format yyyy-MM-dd

token - retrieved from the authorisation call above

planningwindowid - The from the Availability response