



teeone.golf Online Booking Engine Web API

Integration Manual

Table of contents

TABLE OF CONTENTS.....	2
1. PREFACE	3
• PURPOSE	3
• AUDIENCE.....	3
• DOCUMENT HISTORY	3
2. API LOCATION.....	4
3. ACCESS AND AUTHENTICATION FOR API USAGE	4
4. API CALLS.....	5
• API/EXTERNAL/ACCESS/TOKEN	5
• API/EXTERNAL/VENDORS/PROVIDERS.....	6
• API/EXTERNAL/VENDORS/PROVIDERCOURSES.....	8
• API/EXTERNAL/AVAILABILITY/DAYAVAILABILITY.....	10
• API/EXTERNAL/PREBOOKING/PREBOOKINGREQUEST	13
• API/EXTERNAL/BOOKING/BOOKINGCONFIRMATIONREQUEST	15
• API/EXTERNAL/BOOKING/BOOKINGDETAILS.....	17
• API/EXTERNAL/PREANCELLATION/PREANCELLATIONREQUEST	19
• API/EXTERNAL/CANCELLATION/CANCELLATIONCONFIRMATIONREQUEST	21
APPENDIX.....	23
A. RESPONSE CODES.....	23
B. DATA PARAMETER FORMAT.....	23
C. EXAMPLE OF USE	24

1. Preface

- **Purpose**

This document provides a guide to integrate third party software with teeone.golf booking engine using the Web API available for this purpose.

- **Audience**

This technical document is intended for software developers.

- **Document history**

Version	Date	Comments
1.0	21/12/2015	First version of this document
1.1	12/01/2016	- Added tourOperatorID param to Providers call. - Changed ProviderCourses call URL. - Added description of players param in DayAvailability
1.2	17/02/2016	- Added BookingDetails call. - Added PreCancellationRequest call. - Added CancellationConfirmationRequest call.
1.3	02/03/2016	- Added cancelDateTime param to BookingDetails call - Deleted players param from PreCancellationRequest call
1.4	04/03/2016	- Added optional personal details params to BookingConfirmationRequest call
1.5	17/05/2016	- Added tourOperatorName param to Providers call. - Added more possible values for rateTypeID param in DayAvailability. - Added rateTypeID param to PreBookingRequest call.
1.6	22/11/2016	- Corrected success response from PreCancellationRequest
1.7	06/04/2017	- Added package rate "1GF + 1 Buggy"
1.8	14/07/2017	- Added course coordinates (latitude and longitude) in ProviderCourses call.
1.9	22/08/2018	- Changed API location in development environment
1.10	23/08/2021	- Added explanation and examples of how to consume the API using golf club prices.
1.11	01/06/2022	- Added support for promo codes and yield discounts
1.12	02/04/2025	- Brand change

2. API location

teeone.golf online booking API can be accessed in both development and production environment.

In both cases, all API are called through a secured connection via HTTPS.

For development usage, the URL for the API is:

<https://devapi.teeone.golf/TOBookingEngine/v1>

3. Access and authentication for API usage

All teeone.golf API are token-based securized.

To access to those API calls, it is necessary to be previously authenticated. Teeone.golf will provide a test user to access to development environment. Once all tests are successful, another user and password will be provided to access to the real environment.

If the credentials given are correct, a valid token will be sent along with other data necessary for further API calls.

4. API calls

- **api/External/Access/Token**

Returns an access token and the session and vendor ID that have to be used in further API calls. It also returns the session expiration date.

- *Methods Allowed*
POST
- *Content Type*
application/x-www-form-urlencoded
- *Data Params*

```
username=[string]&password=[string]&lifeSpanSeconds=[integer]&culture=[string]
```

Param	Description
username	Username provided by teeone.golf to log in
password	Password provided by teeone.golf to log in
lifeSpanSeconds	Time, in seconds, in which the session will expire
culture	Culture used in the API (by default en-GB)

- *Success Response*
HTTP Status code: 200 - OK

Content:

```
{
  "code": [integer],
  "msg": [string],
  "data": {
    "sessionID": [integer],
    "vendorID": [integer],
    "accessToken": [string],
    "expiration": [date]
  }
}
```

Param	Description
code	Response code (See error code Appendix)
msg	Response message
sessionID	Session id
vendorID	Vendor id
accessToken	Security token generated for further API calls
expiration	Date and time when the session will expire

- Error Response*

HTTP Status codes: 415 - **Unsupported Media Type**, 500 - **Internal Server Error**

Content:

```
{
  "code": [integer],
  "msg": [string]
}
```

Param	Description
code	Response code (See error code Appendix)
msg	Response message

- api/External/Vendors/Providers**

Returns a list of tee time providers of a specific vendor.

- Methods Allowed*

POST

- Content Type*

application/json

- Data Params*

```
{
  "sessionID": [integer],
  "vendorID": [integer],
  "accessToken": [string],
  "culture": [string]
}
```

Param	Description
sessionID	Session id provided in Token API call
vendorID	Vendor id provided in Token API call
accessToken	Session token provided in Token API call
culture	Culture used in the API (by default en-GB)

- *Success Response*
HTTP Status code: 200 - OK

Content:

```
{
  "code": [integer],
  "msg": [string],
  "data": {
    "providersList": [{
      "providerID": [integer],
      "providerName": [string],
      "isClub": [Boolean],
      "tourOperatorID": [integer],
      "tourOperatorName": [string]
    },...],
    "providersCount": [integer]
  }
}
```

Param	Description
code	Response code (See error code Appendix)
msg	Response message
providersCount	Number of providers included in the list
providersList	List of providers of a vendor
Each provider of the list contains	
providerID	Provider id
providerName	Name of the provider
isClub	Indicates whether a provider is a club (true) or a tour operator (false)
tourOperatorID	Tour Operator ID (-1 if the provider is a club)
tourOperatorName	Tour Operator name (empty if the provider is a club)

ONLINE BOOKING ENGINE WEB API

- Error Response*

HTTP Status codes: 401 - Unauthorized, 500 - Internal Server Error

Content:

```
{
  "code": [integer],
  "msg": [string]
}
```

Param	Description
code	Response code (See error code Appendix)
msg	Response message

- api/External/Vendors/ProviderCourses**

Returns the list of courses available for a specific provider.

- Methods Allowed*

POST

- Content Type*

application/json

- Data Params*

```
{
  "vendorID": [integer],
  "providerID": [integer],
  "tourOperatorID": [integer],
  "sessionID": [integer],
  "accessToken": [string],
  "culture": [string]
}
```

Param	Description
sessionID	Session id provided in Token API call
vendorID	Vendor id provided in Token API call
providerID	Provider id given in Token API call
tourOperatorID	Tour operator id (only resellers)
accessToken	Session token provided in Token API call
culture	Culture used in the API (by default en-GB)

ONLINE BOOKING ENGINE WEB API

- *Success Response*

HTTP Status code: 200 - OK

Content:

```
{
  "code": [integer],
  "msg": [string],
  "data": {
    "coursesList": [{
      "courseID": [string],
      "courseName": [string],
      "holes": [integer],
      "latitude": [string],
      "longitude": [string]
    },... ],
    "coursesCount": [integer]
  }
}
```

Param	Description
code	Response code (See error code Appendix)
msg	Response message
coursesCount	Number of courses included in the list
coursesList	List of courses available of a provider
Each course of the list contains	
courseID	Course id
courseName	Name of the course
holes	Number of holes of the course
latitude	Latitude of the course (signed degrees format)
longitude	Longitude of the course (signed degrees format)

- *Error Response*

HTTP Status codes: 401 - Unauthorized, 500 - Internal Server Error

Content:

```
{
  "code": [integer],
  "msg": [string]
}
```

Param	Description
code	Response code (See error code Appendix)
msg	Response message

ONLINE BOOKING ENGINE WEB API

- **api/External/Availability/DayAvailability**

Returns the tee times availability of a given date and course for a specific vendor.

- *Methods Allowed*
POST
- *Content Type*
application/json
- *Data Params*

```
{
  "vendorID": [integer],
  "providerID": [integer],
  "sessionID": [integer],
  "tourOperatorID": [integer],
  "accessToken": [string],
  "culture": [string],
  "courseID": [integer],
  "playDate": [date],
  "fromTime": [date],
  "toTime": [date],
  "players": [integer],
  "fromPrice": [integer],
  "promoCode": [string],
  "toPrice": [integer],
  "pageSize": [integer],
  "pageNum": [integer]
}
```

Param	Description
sessionID	Session id provided in Token API call
vendorID	Vendor id provided in Token API call
providerID	Provider id given in Token API call
tourOperatorID	Tour operator id (only resellers)
accessToken	Session token provided in Token API call
culture	Culture used in the API (by default en-GB)
courseID	Course id
playDate	Play date
fromTime	Beginning play time
toTime	Ending play time
Players	Number of players wanted (-1 for any)
fromPrice	Minimum price allowed (Decimal represented as integer) (See Appendix)
promoCode	Promotional code
toPrice	Maximum price allowed (Decimal represented as integer) (See Appendix)
pageSize	Number of results per page
pageNum	Number of results' page to be returned

- *Success Response*
HTTP Status code: 200 - OK

Content:

```
{
  "code": [integer],
  "msg": [string],
  "data": {
    "promoCodeMsg": [string],
    "promoCodeValid": [Boolean],
    "fee": [integer],
    "totalCharge": [Boolean],
    "teeTimesAvailable": [{
      "time": [date],
      "courseID": [integer],
      "playersAvailable": [integer],
      "ratesList": [{
        "rateID": [integer],
        "rateTimeID": [integer],
        "rateTypeID": [integer],
        "packagePriceMultiplier": [integer],
        "promoCodeId": [integer],
        "rateName": [string],
        "rateComments": [string],
        "sellPrice": [integer],
        "rackPrice": [integer],
        "discountPercentage": [integer],
        "bookablePlayers": [
          [integer],...
        ],
        "bookablePlayersCount": [integer],
        "servicesIncluded": [
          [string],...
        ],
        "servicesIncludedCount": [integer],
        "yieldDiscountId": [integer],
      }, ...}],
    "ratesCount": [integer]
  },...],
  "teeTimesAvailableCount": [integer]
}
```

Param	Description
code	Response code (See error code Appendix)
msg	Response message
promoCodeValid	Indicates whether the promo code sent is valid or not (not available yet)
promoCodeMsg	Result of the promo code check
fee	Fee per green fee charged to the user (Decimal represented as integer) (See Appendix)
totalCharge	Indicates whether the total amount of the booking or only the fee will be charged to the user (Decimal represented as integer) (See Appendix)
teeTimesAvailable	List of available tee times for the filters sent
Each tee time of the list contains	
time	Tee time
coursed	Course id
playersAvailable	Number of players available to be booked for the current tee time
ratesCount	Number of rates in the rates list
ratesList	List of rates available for the specific tee time
Each rate of the list contains	
rateID	Rate id
rateTimeID	Rate time id
rateTypeID	Rate type id. Valid values: 1 – 18 holes 5 – 18 holes for hotel guests 2 – 9 holes 6 – 9 holes for hotel guests 3 – 2 GF + Buggy 7 - 2 GF + Buggy for hotel guests 4 – 4 GF + 2 Buggies 8 - 4 GF + 2 Buggies for h. guests 9 – 1 GF + 1 Buggy 10 - 1 GF + 1 Buggy for h. guests
packagePriceMultiplier	Price multiplier for packages. Indicates the multiplier to apply to the price to get the price per round. (Decimal represented as integer) (See Appendix)
rateName	Name of the rate
rateComments	Brief description of the rate
sellPrice	Retail price (Decimal represented as integer) (See Appendix)
rackPrice	Rack price (Decimal represented as integer) (See Appendix)
discountPercentage	Discount percentage of the retail price compared to rack price (Decimal represented as integer) (See Appendix)
bookablePlayers	List of number of players allowed for this rate
bookablePlayersCount	Number of elements in bookablePlayers
servicesIncluded	List of services included in the rate (cart, buggy,...)
servicesIncludedCount	Number of elements in servicesIncluded
promoCodeId	ID of the promo code requested (if valid). Must be send in PreBookingRequest to apply promo code
yieldDiscountId	ID of yield discount applied to the price. Must be send in PreBookingRequest to apply the discounted price

ONLINE BOOKING ENGINE WEB API

- *Error Response*

HTTP Status codes: 401 - Unauthorized, 500 - Internal Server Error

Content:

```
{
  "code": [integer],
  "msg": [string]
}
```

Param	Description
code	Response code (See error code Appendix)
msg	Response message

- **api/External/PreBooking/PreBookingRequest**

Makes a prebooking operation (pending to confirm) of the tee times sent.

- *Methods Allowed*

POST

- *Content Type*

application/json

- *Data Params*

```
{
  "vendorID": [integer],
  "sessionID": [integer],
  "accessToken": [string],
  "culture": [string],
  "userFirstName": [string],
  "userLastName": [string],
  "userEmail": [string],
  "teeTimesList": [{
    "tourOperatorID": [integer],
    "providerID": [integer],
    "playDateTime": [date],
    "courseID": [integer],
    "ratesList": [{
      "rateID": [integer],
      "rateTimeID": [integer],
      "rateTypeID": [integer],
      "players": [integer],
      "price": [integer],
      "promoCodeID": [integer],
      "rateName": [string],
      "yieldDiscountID": [integer]
    },...]
  },...]
}
```

Param	Description
sessionID	Session id provided in Token API call
vendorID	Vendor id provided in Token API call
accessToken	Session token provided in Token API call
culture	Culture used in the API (by default en-GB)
userFirstName	Customer's first name
userLastName	Customer's last name
userEmail	Customer's email
teeTimesList	List of tee times to prebook
Each tee time of the list contains	
tourOperatorID	Tour operator id (only resellers)
providerID	Provider id
playDateTime	Date and time of the tee time to prebook
courseID	Course id
ratesList	List of green fees to prebook per rate
Each rate of the list contains	
rateID	Rate id
rateTimeID	Rate time id
rateTypeID	Rate type id (see DayAvailability for valid value)
players	Number of players to prebook for this rate
price	Price of this rate (Decimal represented as integer) (See Appendix)
promoCodeId	Promo code ID obtained in availability search
rateName	Rate name
yieldDiscountId	Yield discount ID applied to price obtained in availability search

- *Success Response*
HTTP Status code: 200 - OK

Content:

```
{
  "code": [integer],
  "msg": [string],
  "data": {
    "orderReference": [string]
  }
}
```

Param	Description
code	Response code (See error code Appendix)
msg	Response message
orderReference	Order reference code

ONLINE BOOKING ENGINE WEB API

- *Error Response*
HTTP Status codes: 401 - Unauthorized, 500 - Internal Server Error

Content:

```
{
  "code": [integer],
  "msg": [string]
}
```

Param	Description
code	Response code (See error code Appendix)
msg	Response message

• **api/External/Booking/BookingConfirmationRequest**

Makes a booking confirmation of a prebooking request. Optional customer details can be sent, that will rewrite the details sent in PreBookingRequest

- *Methods Allowed*
POST
- *Content Type*
application/json
- *Data Params*

```
{
  "vendorID": [integer],
  "sessionID": [integer],
  "accessToken": [string],
  "culture": [string],
  "userFirstName": [string],
  "userLastName": [string],
  "userEmail": [string],
  "orderRef": [string]
}
```

Param	Description
sessionID	Session id provided in Token API call
vendorID	Vendor id provided in Token API call
accessToken	Session token provided in Token API call
culture	Culture used in the API (by default en-GB)
orderRef	Order reference code
userFirstName	Customer's first name (optional)
userLastName	Customer's last name (optional)
userEmail	Customer's email (optional)

ONLINE BOOKING ENGINE WEB API

- *Success Response*

HTTP Status code: 200 - OK

Content:

```
{
  "code": [integer],
  "msg": [string],
}
```

Param	Description
code	Response code (See error code Appendix)
msg	Response message

- *Error Response*

HTTP Status codes: 401 - Unauthorized, 500 - Internal Server Error

Content:

```
{
  "code": [integer],
  "msg": [string]
}
```

Param	Description
code	Response code (See error code Appendix)
msg	Response message

- **api/External/Booking/BookingDetails**

Gets the details of an already confirmed booking.

- *Methods Allowed*
POST
- *Content Type*
application/json
- *Data Params*

```
{
  "vendorID": [integer],
  "sessionID": [integer],
  "accessToken": [string],
  "culture": [string],
  "orderRef": [string]
}
```

Param	Description
sessionID	Session id provided in Token API call
vendorID	Vendor id provided in Token API call
accessToken	Session token provided in Token API call
Culture	Culture used in the API (by default en-GB)
orderRef	Order reference code

- *Success Response*
HTTP Status code: 200 - OK

Content:

```
{
  "code": [integer],
  "msg": [string],
  "data": {
    "userFirstName": [string]
    "userLastName": [string],
    "bookingDetails": [{
      "providerID": [integer],
      "tourOperatorID": [integer],
      "providerName": [string],
      "courseID": [integer],
      "courseName": [string],
      "playDateTime": [date],
      "rateName": [string],
      "cancelDateTime": [date],
      "players": [integer],
      "bookingDetailsID": [integer]
    }...]
  }
}
```

Param	Description
code	Response code (See error code Appendix)
msg	Response message
userFirstName	User first name
userLastName	User last name
bookingDetails	List of bookings associated with the order reference
Each element of the list contains	
tourOperatorID	Tour operator id (only resellers)
providerID	Provider id
providerName	Provider name
courseID	Course id
courseName	Name of the course
playDateTime	Play date and time
rateName	Rate name
cancelDateTime	Deadline for cancellation
players	Number of players
bookindDetailsID	Unique identifier for that booking

- *Error Response*

HTTP Status codes: 401 - Unauthorized, 500 - Internal Server Error

Content:

```
{
  "code": [integer],
  "msg": [string]
}
```

Param	Description
code	Response code (See error code Appendix)
msg	Response message

- **api/External/PreCancellation/PreCancellationRequest**

Makes a pre-cancellation operation (pending to confirm) of the tee times sent.

- *Methods Allowed*

POST

- *Content Type*

application/json

- *Data Params*

```
{
  "vendorID": [integer],
  "sessionID": [integer],
  "accessToken": [string],
  "culture": [string],
  "orderRef": [string],
  "teeTimesList": [{
    "providerID": [integer],
    "playDateTime": [date],
    "bookingDetailsID": [integer]
  },...]
}
```

Param	Description
sessionID	Session id provided in Token API call
vendorID	Vendor id provided in Token API call
accessToken	Session token provided in Token API call
culture	Culture used in the API (by default en-GB)
orderRef	Booking order reference code
teeTimesList	List of tee times to pre-cancel
Each tee time of the list contains	
bookingDetailsID	Details id provided in BookingDetails for tee time
providerID	Provider id
playDateTime	Date and time of the tee time to prebook

- *Success Response*
HTTP Status code: 200 - OK

Content:

```
{
  "code": [integer],
  "msg": [string],
  "data": {
    "orderReference": [string]
  }
}
```

Param	Description
code	Response code (See error code Appendix)
msg	Response message
orderReference	Cancellation order reference

- *Error Response*
HTTP Status codes: 401 - Unauthorized, 500 - Internal Server Error

Content:

```
{
  "code": [integer],
  "msg": [string]
}
```

Param	Description
code	Response code (See error code Appendix)
msg	Response message

- **api/External/Cancellation/CancellationConfirmationRequest**

Makes a cancellation confirmation of a pre-cancellation request.

- *Methods Allowed*
POST
- *Content Type*
application/json
- *Data Params*

```
{
  "vendorID": [integer],
  "sessionID": [integer],
  "accessToken": [string],
  "culture": [string],
  "orderRef": [string],
  "cancellationOrderRef": [string]
}
```

Param	Description
sessionID	Session id provided in Token API call
vendorID	Vendor id provided in Token API call
accessToken	Session token provided in Token API call
Culture	Culture used in the API (by default en-GB)
orderRef	Order reference code
cancellationOrderRef	Cancellation order reference code

- *Success Response*
HTTP Status code: 200 - OK

Content:

```
{
  "code": [integer],
  "msg": [string]
}
```

ONLINE BOOKING ENGINE WEB API

Param	Description
code	Response code (See error code Appendix)
msg	Response message

- Error Response*

HTTP Status codes: **401 - Unauthorized**, **500 - Internal Server Error**

Content:

```
{
  "code": [integer],
  "msg": [string]
}
```

Param	Description
code	Response code (See error code Appendix)
msg	Response message

APPENDIX

A. Response Codes

API calls have two different response codes: HTTP status codes and the content response code. The following table shows the way they are related and the different values they may have:

HTTP Status Code	Content response code	Description
200 - OK	1	Correct (no error detected)
	2	Error processing the request (bad ID, incorrect data parameter, etc.)
415 - Unsupported Media Type	4	The media type used to call the API is not correct. (Only appears in token API)
401 - Unauthorized	4	Security check failed (bad token, bad ID,...)
500 - Internal Server Error	3	Unexpected error. Check error message in order to get the error details

B. Data parameter format

Some parameters that represent date, time and decimal numbers use the following format:

- *Dates and times.* Uses the ANSI standard format: yyyy-MM-ddTHH:mm
- *Decimal numbers.* They are represented as integer numbers. The last two digits represent the decimal part, and the rest of digits represent the whole part. If the number has only two digits, it represents only the decimal part. Example: 50 means 0.50.

C. Example of use

To use the teeone.golf Online Booking API, you previously need to have the credentials to get a valid token to use the API. Here's an example of the full cycle API calls in order to book tee times.

Step 1

Call to *api/External/Access/Token* to get the access token.

Request:

```
username=tour@golfspain.com&password=H37$mAS&lifeSpanSeconds=3600&culture=en-GB
```

Response:

```
{
  "code": 1,
  "msg": "",
  "data": {
    "sessionID": 1053,
    "vendorID": 15,
    "accessToken":
    "BM6SX4Lgv7gGTI1UKwwe4jNubxNJRjOVFgv1fa51rT9WHLTxIZFKkknE1i+S+SX08Dytp
    6lS/2kBSERY10BhT3x4F+k/+V2BQZ2gH5vF1DY38HLVJRcxbEazUAUELJ4A",
    "expiration": "2021-12-21T14:51:55.8145299+01:00"
  }
}
```


Step 2

Call to *api/External/Vendors/Providers* to get the providers.

Request:

```
{
  "sessionID": 1053,
  "vendorID": 15,
  "accessToken":
  "BM6SX4Lgv7gGTiLUKwwe4jNubxNJRjOVFgv1fa51rT9WHLTxIZFKkknE1i+S+SX08Dytp61S/2kBSE
  RY10BhT3x4F+k/+V2BQZ2gH5vF1DY38HLVJRcxBEazUAUELJ4A",
  "culture": "en-GB"
}
```

Response:

```
{
  "code": 1,
  "msg": "",
  "data": {
    "providersList": [{
      "providerID": 1,
      "providerName": "Club Demo",
      "isClub": true
    }],
    "providersCount": 1
  }
}
```

Step 3

Call to *api/External/Vendors/ProviderCourses* to get the courses of one provider.

Request:

```
{
  "vendorID": 15,
  "providerID": 1,
  "sessionID": 1053,
  "tourOperatorID": -1,
  "accessToken":
  "BM6SX4Lgv7gGTiLUKwwe4jNubxNJRjOVFgv1fa51rT9WHLTxIZFKkknE1i+S+SX08Dytp6lS/2kBSE
  RY10BhT3x4F+k/+V2BQZ2gH5vF1DY38HLVJRcxbEazUAUELJ4A",
  "culture": "en-GB"
}
```

Response:

```
{
  "code": 1,
  "msg": "",
  "data": {
    "coursesList": [{
      "courseID": "1",
      "courseName": "TEE 1 - 18 holes",
      "holes": 18,
      "latitude": "40.423232",
      "longitude": "-3.687890"
    }, {
      "courseID": "2",
      "courseName": "TEE 1 - 9 holes",
      "holes": 9,
      "latitude": "40.423232",
      "longitude": "-3.687890"
    }, {
      "courseID": "3",
      "courseName": "TEE 10",
      "holes": 9,
      "latitude": "40.423232",
      "longitude": "-3.687890"
    }
  ],
  "coursesCount": 3
}
```

Step 4

Call to *api/External/Availability/DayAvailability* to get the availability of a specific day.

Request:

```
{
  "vendorID": 15,
  "providerID": 1,
  "sessionID": 1053,
  "tourOperatorID": -1,
  "accessToken":
  "BM6SX4Lgv7gGTiLUKwwe4jNubxNJRjOVFgv1fa51rT9WHLTxIZFKkknE1i+S+SX08Dytp6lS/2kBSE
  RY10BhT5SakKvTeLaTJ9MS1z4lkxyNXn084JoentDhKhfhaZ6H",
  "culture": "en-GB",
  "courseID": 1,
  "playDate": "2021-12-25",
  "fromTime": "08:00",
  "toTime": "11:00",
  "players": -1,
  "fromPrice": 100,
  "toPrice": 50000,
  "pageSize": -1,
  "pageNum": -1,
  "promoCode": ""
}
```

ONLINE BOOKING ENGINE WEB API

Response:

```
{
  "code": 1,
  "msg": "",
  "data": {
    "promoCodeMsg": "",
    "promoCodeValid": false,
    "fee": 0,
    "totalCharge": false,
    "teeTimesAvailable": [{
      "time": "1900-01-01T10:00:00",
      "courseID": "1",
      "playersAvailable": 4,
      "rates": [{
        "rateID": 1017,
        "rateTimeID": 1016,
        "rateTypeID": 1,
        "packagePriceMultiplier": 100,
        "rateName": "Adult rate",
        "rateComments": "",
        "sellPrice": 12010,
        "rackPrice": 14010,
        "discountPercentage": 1400,
        "bookablePlayers": [1, 2, 3, 4],
        "bookablePlayersCount": 4,
        "servicesIncluded": [],
        "servicesIncludedCount": 0,
        "promoCodeId": -1,
        "yieldDiscountId": 513
      }, {
        "rateID": 1018,
        "rateTimeID": 1017,
        "rateTypeID": 3,
        "packagePriceMultiplier": 50,
        "rateName": "2GF+Buggy",
        "rateComments": "No cancellations allowed",
        "sellPrice": 16020,
        "rackPrice": 18020,
        "discountPercentage": 2500,
        "bookablePlayers": [2, 4],
        "bookablePlayersCount": 4,
        "servicesIncluded": ["GPS", "Lunch"],
        "servicesIncludedCount": 2,
        "promoCodeId": -1,
        "yieldDiscountId": -1
      }
    ]},
    "ratesCount": 2
  },
}
```

(continue...)

(...continue)

```
{
  "time": "1900-01-01T10:10:00",
  "courseID": "1",
  "playersAvailable": 4,
  "rates": [{
    "rateID": 1017,
    "rateTimeID": 1016,
    "rateTypeID": 1,
    "packagePriceMultiplier": 100,
    "rateName": "Senior rate",
    "rateComments": "",
    "sellPrice": 10010,
    "rackPrice": 12010,
    "discountPercentage": 1400,
    "bookablePlayers": [1, 2, 3, 4],
    "bookablePlayersCount": 4,
    "servicesIncluded": [],
    "servicesIncludedCount": 0,
    "promoCodeId": -1,
    "yieldDiscountId": -1
  }],
  "ratesCount": 1
}],
"teeTimesAvailableCount": 2
}
```

Step 5

Call to *api/External/PreBooking/PreBookingRequest* to make a prebooking based on the availability.

Request:

```
{
  "vendorID": 15,
  "sessionID": 1053,
  "accessToken":
  "BM6SX4Lgv7gGTiLUKwwe4jNubxNJRjOVFgv1fa51rT9WHLTxIZFKkknE1i+S+SX08Dytp61S/2kBSE
  RY10BhT5SakKvTeLAtJ9MS1z4lkxyNXn084JoentDhKhfhaz6H ",
  "culture": "en-GB",
  "userFirstName": "John",
  "userLastName": "Smith",
  "userEmail": "john.smith@customeremail.com",
  "teeTimesList": [{
    "tourOperatorID": -1,
    "providerID": 1,
    "playDateTime": "2021-12-25T10:00:00",
    "courseID": 1,
    "ratesList": [{
      "rateID": 1017,
      "rateTimeID": 1016,
      "players": 2,
      "price": 28020,
      "rateName": "Adult rate",
      "rateTypeID": 1,
      "promoCodeId": -1,
      "yieldDiscountId": 513
    }, {
      "rateID": 1018,
      "rateTimeID": 1017,
      "players": 2,
      "price": 16020,
      "rateName": "2G+Buggy",
      "rateTypeID": 3,
      "promoCodeId": -1,
      "yieldDiscountId": -1
    }
  ]
}]
}
```

Response:

```
{
  "code": 1,
  "msg": "",
  "data": {
    "orderReference": "6GH5D3MA"
  }
}
```

Step 6

Call to *api/External/Booking/BookingConfirmationRequest* to confirm a previous prebooking request.

Request:

```
{
  "vendorID": 15,
  "sessionID": 1053,
  "accessToken":
  "BM6SX4Lgv7gGTiLUKwwe4jNubxNJRjOVFgv1fa51rT9WHLTxIZFKkknE1i+S+SX08Dytp6lS/2kBSE
  RY10BhT5SakKvTeLAtJ9MS1z4lkxyNXn084JoentDhKhfhaz6H",
  "culture": "en-GB",
  "orderRef": "6GH5D3MA"
}
```

Response:

```
{
  "code": 1,
  "msg": ""
}
```

Step 7

Call to *api/External/Booking/BookingDetails* to show the booking info and to get data for a possible cancellation request.

ONLINE BOOKING ENGINE WEB API

Request:

```
{
  "vendorID": 15,
  "sessionID": 1053,
  "accessToken":
  "BM6SX4Lgv7gGTILUKwwe4jNubxNJRjOVFgv1fa51rT9WHLTxIZFKkknE1i+S+SX08Dytp6lS/2kBSE
  RY10BhT5SakKvTeLAtJ9MS1z4lkxyNXn084JoenTdhKhfhaZ6H",
  "culture": "en-GB",
  "orderRef": "6GH5D3MA"
}
```

Response:

```
{
  "code": 1,
  "msg": "",
  "data": {
    "userFirstName": "John",
    "userLastName": "Smith",
    "bookingDetails": [{
      "providerID": 1,
      "tourOperatorID": -1,
      "providerName": "Club Demo",
      "courseID": 1,
      "courseName": "TEE 1 - 18 holes",
      "playDateTime": "2021-12-25T10:00:00",
      "cancelDateTime": "2021-12-15T10:00:00",
      "rateName": "Adult rate",
      "players": 2,
      "bookingDetailsID": 134
    }, {
      "providerID": 1,
      "tourOperatorID": -1,
      "providerName": "Club Demo",
      "courseID": 1,
      "courseName": "TEE 1 - 18 holes",
      "playDateTime": "2021-12-25T10:00:00",
      "cancelDateTime": "2021-12-15T10:00:00",
      "rateName": "2GF+Buggy"
      "players": 2,
      "bookingDetailsID": 259
    }
  ]
}
```


Step 8

Call to *api/External/PreCancellation/PreCancellationRequest* in order to start a cancellation request.

Request:

```
{
  "vendorID": 15,
  "sessionID": 1053,
  "accessToken":
  "BM6SX4Lgv7gGTiLUKwwe4jNubxNJRjOVFgv1fa51rT9WHLTxIZFKkknE1i+S+SX08Dytp6lS/2kBSE
  RY10BhT5SakKvTeLAtJ9MS1z4lkxyNXn084JoenTdhKhfhaZ6H",
  "culture": "en-GB",
  "orderRef": "6GH5D3MA",
  "teeTimesList": [{
    "providerID": 1,
    "playDateTime": "2021-12-25T10:00:00",
    "bookingDetailsID": 134
  }]
}
```

Response:

```
{
  "code": 1,
  "msg": "",
  "data": {
    "orderReference": "H74GG36X"
  }
}
```

Step 9

Call to *api/External/Cancellation/CancellationConfirmationRequest* to confirm a previous pre-cancellation request.

Request:

```
{
  "vendorID": 15,
  "sessionID": 1053,
  "accessToken":
  "BM6SX4Lgv7gGTI1UKwwe4jNubxNJRjOVFgv1fa51rT9WHLTxIZFKkknE1i+S+SX08Dytp61S/2kBSE
  RY10BhT5SakKvTeLAtJ9MS1z41kxyNXn084JoentdhKhfhaZ6H",
  "culture": "en-GB",
  "cancellationOrderRef": "H74GG36X"
}
```

Response:

```
{
  "code": 1,
  "msg": ""
}
```