



Events External API

Business Summary

Contents

- Purpose of this document 3
- Functional requirements 3
 - Limitations 4
- Functional specification 4
 - API Information 4
- Expected usage 5
- Integration 5
- Example booking flow 5
 - Assumptions 6

Purpose of this document

This document provides a business level overview of the Events External API which will enable external third-party applications to create their own booking widgets and booking journey utilising the functionality of the Zonal booking platform.

The full technical specification can now be found on swagger.

Functional requirements

1. It must be possible to search for sites by geo-location
 - a. By region
 - b. By county
 - c. By city
2. It must be possible for third-party applications to create their own booking journeys
 - a. All queries must support the same list of optional parameters
 - b. Third-party applications can choose which parameters to include based on information already captured in their booking journey e.g. choose site, then occasion, then date/time; or choose occasion, then date/time, then site
 - c. All responses are scoped to the API key that is provided
3. It must be possible for third-party applications to create a new event on behalf of a guest
 - a. The event must target a specific site
 - b. The guest must specify the occasion
 - c. The guest may select an area for the site
 - d. The guest must specify the date and time of the event
 - e. The guest must specify the number of adults and children attending the event
 - f. The guest may select a menu
 - g. The guest may select one or more upsells to be included in the event and be informed of any cost of selecting the upsell(s)
 - h. The guest must be informed of any payment and pre-order requirements based on their selection
 - i. The guest may opt to personalise the event with a name e.g. Dad's birthday
 - j. The guest must specify contact details including their name, email address and telephone number
 - k. The guest must accept the terms and conditions of use
 - l. The guest may opt in to marketing via different methods e.g. telephone, email etc.
4. It must be possible for third-party applications to create a new enquiry on behalf of a guest
 - a. The enquiry must target a specific site
 - b. The guest must specify the occasion
 - c. The guest must specify the date and time they wish to book for
 - d. The guest must specify contact details including their name, email address and telephone number
 - e. The guest must accept the terms and conditions of use
 - f. The guest may specify a menu, area and upsells
 - g. The guest may opt in to marketing via different methods e.g. telephone, email etc.
5. The API must be versioned to prevent breaking changes
6. The API key must be scoped to all sites within the estate, all sites within a company within the estate or a specific site

- a. All end points must filter results to only match the sites that the request's API key has access to regardless of any parameter

Limitations

1. Geo-location search is limited to sites within the UK only.
2. Prices are limited to UK pounds only
3. The API caters for the booking journey only. For guests to manage their booking, they will receive a link in their confirmation email that will direct them to a guest portal. From there, the guest will be able to amend or cancel their booking, and to make any pre-order selections that may be required for their booking.

Functional specification

API Information

Component	Notes
Scheme	HTTPS All requests must be made using secure HTTP.
Host	api.liveres.co.uk The domain where the API will be available in production. Other environments (staging) will use a different domain.
Base path	/events/v1 All endpoints must be prefixed with the base path e.g. GET /events/v1/outlets
Produces	application/json All responses will be provided in the JSON interchange format as defined by RFC 7159. Therefore, all requests must specify the following HTTP header: Accept: application/json Any request that does not provide the above header will result in a "406 Not Acceptable" response.
Consumes	application/json All POST and PUT requests must specify the following HTTP header and provide a body formatted using the JSON interchange format as defined in RFC 7159. Content-Type: application/json Any POST request that does not provide the above header and/or does not provide a body in the JSON format will result in a "415 Unsupported Media Type" response.
Security	Basic auth API key and password will be provided for use with basic authentication. These credentials may be scoped to a subset of sites within the estate. Must be provided for all API requests.
Type	Types mentioned in this document conform to the OpenAPI 3.0 specification unless otherwise stated. https://www.openapis.org/

Format	time	Schema entries and parameters marked as using the “time” format must provide the time representation in 24-hour format e.g. 16:30 or 07:00
	price	Some schema entries will specify prices. These will be strings containing the currency symbol and a positive decimal value to 2 places e.g. £10.95 or £5.00 Prices will be limited to pound sterling

Expected usage

The use of this API should be within the acceptable usage limit set upon integrators. The usage limits vary depending on client, type of integration and intended use.

Integration

All integrators must use a specified non-production environment to integrate against. This environment will have the same application running but may not have the same hardware.

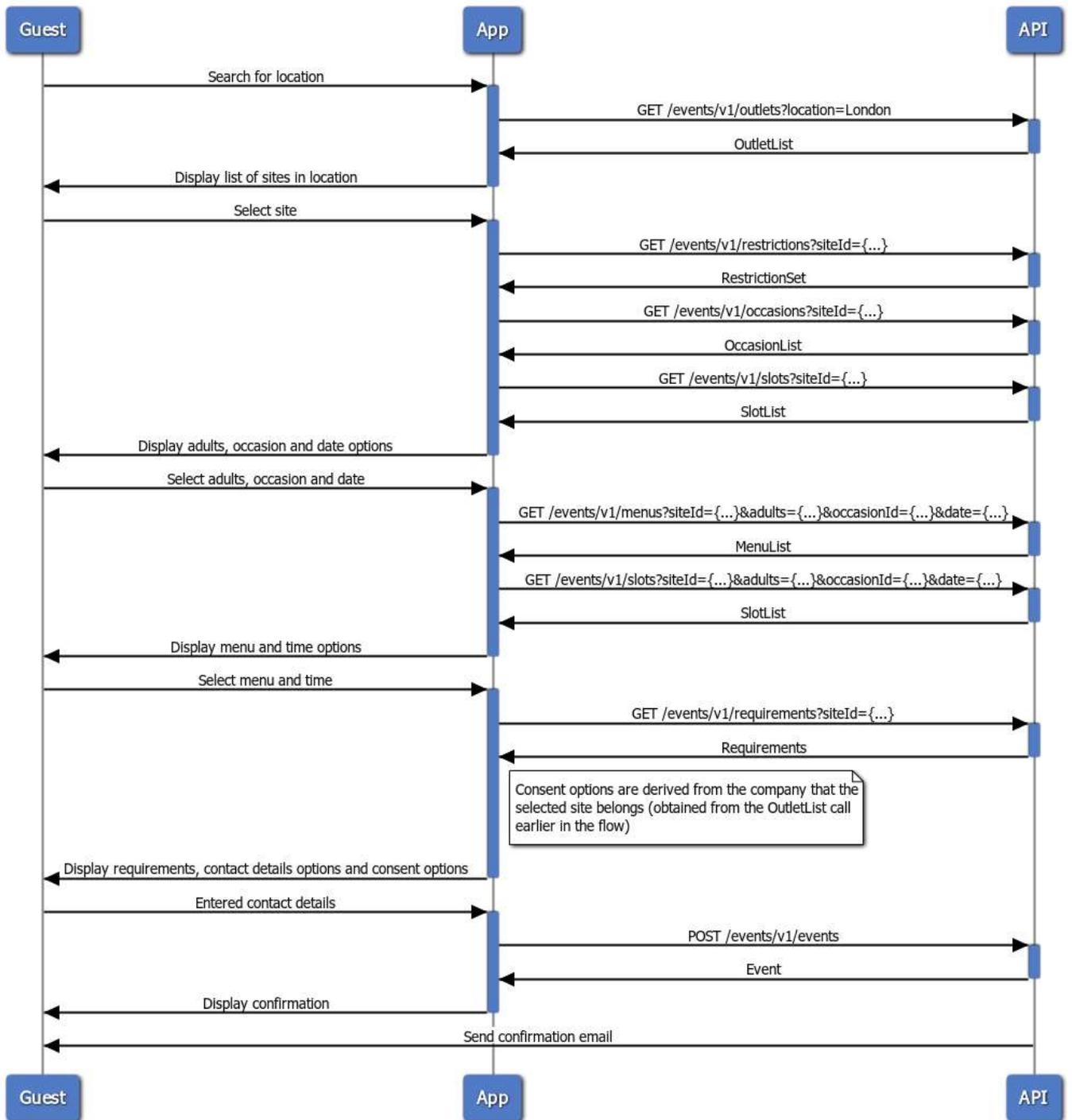
Only approved integrations will be allowed access to our production environment.

Advice can be given for the use of Zonal APIs. Please discuss with your Zonal representative.

Example booking flow

The below sequence diagram details the calls necessary to perform the following example booking flow steps:

1. Search for site by geo-location
2. Select adults, occasion and date
3. Select menu and time
4. Display booking requirements (deposit, pre-order, pre-payment, etc)
5. Enter contact details, special requests and consent
6. Submit booking



Assumptions

1. The application must be able to cache information from calls within the booking flow to use at later stages