



ExtremeLocation API User Guide

Release v2.1.0



Copyright © 2018 Extreme Networks, Inc. All rights reserved.

Legal Notice

Extreme Networks, Inc. reserves the right to make changes in specifications and other information contained in this document and its website without prior notice. The reader should in all cases consult representatives of Extreme Networks to determine whether any such changes have been made.

The hardware, firmware, software or any specifications described or referred to in this document are subject to change without notice.

Trademarks

Extreme Networks and the Extreme Networks logo are trademarks or registered trademarks of Extreme Networks, Inc. in the United States and/or other countries.

All other names (including any product names) mentioned in this document are the property of their respective owners and may be trademarks or registered trademarks of their respective companies/owners.

For additional information on Extreme Networks trademarks, please see:

www.extremenetworks.com/company/legal/trademarks

Open Source Declarations

Some software files have been licensed under certain open source or third-party licenses. End-user license agreements and open source declarations can be found at:

www.extremenetworks.com/support/policies/software-licensing

Table of Contents

- Preface..... 4**
 - Conventions..... 4
 - Providing Feedback to Us..... 5
 - Getting Help..... 5
 - Documentation and Training..... 6

- Chapter 1: ExtremeLocation APIs..... 7**

- Chapter 2: Outbound APIs..... 8**
 - Wi-Fi Outbound API Events..... 9
 - BLE Outbound API Events..... 16
 - Creating the Sample Web Application..... 22
 - Setting Up the Remote Web Server..... 23
 - Configuring Subscriber Settings..... 29

- Chapter 3: REST API..... 31**
 - API Request Components..... 31
 - API Response Codes..... 32
 - Accessing the REST API..... 33
 - Sites Resource..... 33
 - Floor Resource..... 39
 - Device Resource..... 46



Preface

This section discusses the conventions used in this guide, ways to provide feedback, additional help, and other Extreme Networks® publications.

Conventions

This section discusses the conventions used in this guide.

Text Conventions

The following tables list text conventions that are used throughout this guide.

Table 1: Notice Icons





Icon	Notice Type	Alerts you to...
	General Notice	Helpful tips and notices for using the product.
	Note	Important features or instructions.
	Caution	Risk of personal injury, system damage, or loss of data.
	Warning	Risk of severe personal injury.
<i>New!</i>	New Content	Displayed next to new content. This is searchable text within the PDF.

Table 2: Text Conventions

Convention	Description
Screen displays	This typeface indicates command syntax, or represents information as it appears on the screen.
The words enter and type	When you see the word “enter” in this guide, you must type something, and then press the Return or Enter key. Do not press the Return or Enter key when an instruction simply says “type.”
[Key] names	Key names are written with brackets, such as [Return] or [Esc] . If you must press two or more keys simultaneously, the key names are linked with a plus sign (+). Example: Press [Ctrl]+[Alt]+[Del]
<i>Words in italicized type</i>	Italics emphasize a point or denote new terms at the place where they are defined in the text. Italics are also used when referring to publication titles.

Terminology

When features, functionality, or operation is specific to a switch family, such as ExtremeSwitching, the family name is used. Explanations about features and operations that are the same across all product families simply refer to the product as the switch.

Providing Feedback to Us

Quality is our first concern at Extreme Networks, and we have made every effort to ensure the accuracy and completeness of this document. We are always striving to improve our documentation and help you work better, so we want to hear from you! We welcome all feedback but especially want to know about:

- Content errors or confusing or conflicting information.
- Ideas for improvements to our documentation so you can find the information you need faster.
- Broken links or usability issues.

If you would like to provide feedback to the Extreme Networks Information Development team, you can do so in two ways:

- Use our short online feedback form at <https://www.extremenetworks.com/documentation-feedback/>.
- Email us at documentation@extremenetworks.com.

Please provide the publication title, part number, and as much detail as possible, including the topic heading and page number if applicable, as well as your suggestions for improvement.

Getting Help

If you require assistance, contact Extreme Networks using one of the following methods:

Extreme Portal	Search the GTAC (Global Technical Assistance Center) knowledge base, manage support cases and service contracts, download software, and obtain product licensing, training, and certifications.
The Hub	A forum for Extreme Networks customers to connect with one another, answer questions, and share ideas and feedback. This community is monitored by Extreme Networks employees, but is not intended to replace specific guidance from GTAC.
Call GTAC	For immediate support: 1-800-998-2408 (toll-free in U.S. and Canada) or +1 408-579-2826. For the support phone number in your country, visit: www.extremenetworks.com/support/contact

Before contacting Extreme Networks for technical support, have the following information ready:

- Your Extreme Networks service contract number and/or serial numbers for all involved Extreme Networks products
- A description of the failure
- A description of any action(s) already taken to resolve the problem
- A description of your network environment (such as layout, cable type, other relevant environmental information)
- Network load at the time of trouble (if known)

- The device history (for example, if you have returned the device before, or if this is a recurring problem)
- Any related RMA (Return Material Authorization) numbers

Subscribing to Service Notifications

You can subscribe to email notifications for product and software release announcements, Vulnerability Notices, and Service Notifications.

- 1 Go to www.extremenetworks.com/support/service-notification-form.
- 2 Complete the form with your information (all fields are required).
- 3 Select the products for which you would like to receive notifications.



Note

You can modify your product selections or unsubscribe at any time.

- 4 Click **Submit**.

Documentation and Training

To find Extreme Networks product guides, visit our documentation pages at:

Current Product Documentation	www.extremenetworks.com/documentation/
Archived Documentation (for earlier versions and legacy products)	www.extremenetworks.com/support/documentation-archives/
Release Notes	www.extremenetworks.com/support/release-notes
Hardware/Software Compatibility Matrices	https://www.extremenetworks.com/support/compatibility-matrices/
White papers, data sheets, case studies, and other product resources	https://www.extremenetworks.com/resources/

Training

Extreme Networks offers product training courses, both online and in person, as well as specialized certifications. For more information, visit www.extremenetworks.com/education/.

1 ExtremeLocation APIs

ExtremeLocation provides the following two types of APIs:

- **Outbound APIs:** A set of APIs to push real-time location data and events for wireless clients that are detected in a venue.
- **REST APIs:** A set of APIs to query and retrieve information like site names, floor plans, and region information from the ExtremeLocation server.

The Outbound and RESTful APIs are part of ExtremeLocation's subscriber service.

Related Links

[Outbound APIs](#) on page 8

[REST API](#) on page 31

2 Outbound APIs

Wi-Fi Outbound API Events
BLE Outbound API Events
Creating the Sample Web Application
Setting Up the Remote Web Server
Configuring Subscriber Settings

ExtremeLocation supports real-time reporting of events and notifications to an external server (also called a Subscriber). This feature uses either Wi-Fi or Bluetooth Low-Energy (BLE) beacon positioning and is available to all ExtremeLocation customers by default.

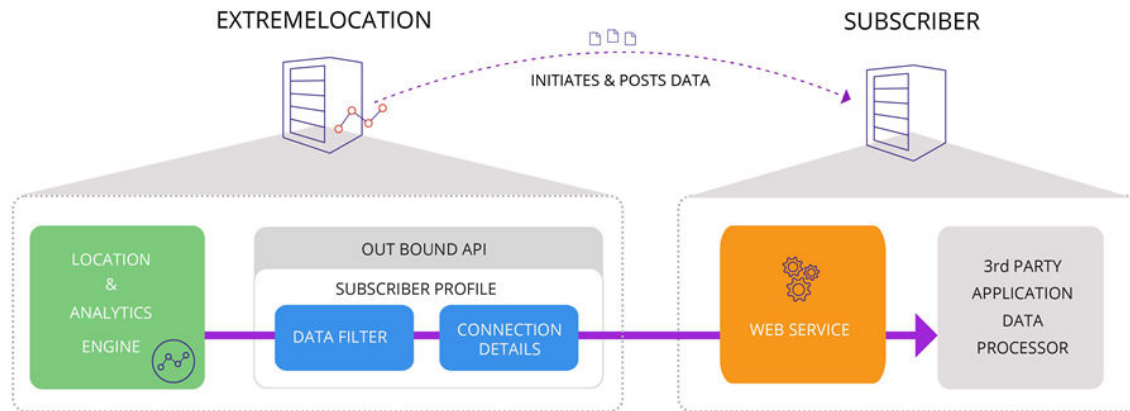
Wi-Fi positioning uses existing WLAN infrastructure for location tracking. ExtremeLocation estimates the location of the client device based on the signal strength of 802.11 packets reported by the access points or sensors deployed on the site. With BLE technology, the SDK in the loyalty app on BLE-enabled mobile devices receives the nearest beacon(s) and communicates the beacon IDs to ExtremeLocation to calculate the client location.

These micro-proximity engagement solutions help you identify and locate visitors in the venue, and engage with them by delivering highly customized and contextual messages directly on their mobile devices. You can use the location data and events and integrate them with your CRM and Big Data systems to build custom applications that use real-time location data.

To receive real-time data from ExtremeLocation, you must have a web server in your network capable of receiving and processing the real-time data. You must configure the credentials required for accessing your server through the **Subscriber Settings** page in the ExtremeLocation GUI. To prevent transmission of data in clear text, Extreme Networks suggests enabling HTTPS support on the your server. Data is presented in JSON format and is encrypted before transmission to your server. For more information, see [Setting Up the Remote Web Server](#) on page 23.

When an event is generated for any site in the ExtremeLocation account, it is sent to the outbound API handler in real-time. The outbound API handler checks if any filters are to be applied to the event. Filters are used to control the event types to pass to the external web server. It then uses the subscriber connection information to send the filtered events to your web server.

For more information on configuring subscriber settings, see [Configuring Subscriber Settings](#) on page 29.



Note

This is an unidirectional connection from ExtremeLocation to the subscriber server. The subscriber server cannot send requests to ExtremeLocation.

Related Links

[Wi-Fi Outbound API Events](#) on page 9

[BLE Outbound API Events](#) on page 16

Wi-Fi Outbound API Events

The events generated by the ExtremeLocation Outbound APIs using Wi-Fi positioning can be classified as follows:

- [Location Events](#) on page 9
- [Presence Events](#) on page 10
- [Category Events](#) on page 13
- [Crowding Events](#) on page 15

Location Events

A Location event is generated when multiple sensors deployed at a site detect:

- A change in the location of the device (position tracking),
- Or a RSSI value above a specified threshold (zone tracking).

Sample JSON Content for Wi-Fi Location Event

```
{
  "eventType": "LOCATION_EVENT",
  "clientMac": "34:f3:9a:2d:a7:b2",
  "siteName": "SanJose_Office",
  "floorName": "4thFloor",
  "x": 51.913044445216656,
  "y": 20.260869942605495,
  "deviceType": "VISITOR",
```

```

    "confidence": "0.1",
    "timestamp": 1503902860334
  }

```

Parameter	Description
eventType	The type of event. For a location event this value is LOCATION_EVENT.
clientMac	The MAC address of the device for which the location event was generated.
siteName	The name of the site in which the client is located.
floorName	The name of the floor in which the client is located.
x	The "X" coordinate of the location where the client is located. This is the positional coordinate of the location relative to the floor map's origin (0,0) point.
y	The "Y" coordinate of the location where the client is located. This is the positional coordinate of the location relative to the floor map's origin (0,0) point.
deviceType	The device classification type of the client. This classification is based on rules configured by the user. The valid values are: <ul style="list-style-type: none"> VISITOR - A device that is present at a site for a limited duration and then leaves the site. ASSET - A device that belongs to a site. An asset could be a wireless POS, an access point deployed at a site, or other similar devices. ASSOCIATE - A device that is carried by an associate and used for work related activities. An associate device could be a hand-held device like a wireless bar code scanner, a mobile/tablet used for inventory purposes, or other similar devices.
confidence	The confidence level of the location engine for locating this client at the x and y coordinates. The value of this field is between 0 and 1. Multiplying this value by 100 indicates the confidence value in percentage (%). A high value indicates that the location engine is confident that the client device is located at those particular x and y coordinates.
timestamp	The time (in epoch milliseconds) when the event was triggered.

Presence Events

A Presence event is generated when a client device enters or exits a site or is in the vicinity of a site. A Presence event can be further classified as a Device Enter or a Device Exit event.

Device Enter Event

A Device Enter event is generated when a device is first seen by the sensors deployed at a site. The sensors report information such as the time of entry, the signal strength value, the visitor type (first time visitor or repeat visitor), etc.

Sample JSON Content for Wi-Fi Device Enter Event

```

{
  "eventType": "DEVICE_ENTERED",
  "clientMac": "00:6f:64:31:59:5f",
  "sensorMac": "74:67:f7:9a:15:ba",

```

```

    "siteId": "SITE_203a7140654a4db5a1c82ef12264e64b",
    "siteName": "SanJose_Office",
    "lastSeen": "1503902650494",
    "timestamp": 1503902398255,
    "rssiValue": -87.0,
    "loyalty": "REPEAT"
  }

```

Parameter	Description
eventType	The type of event. For a Device Presence event this value is DEVICE_ENTERED.
clientMac	The MAC address of the device for which the Device Enter event was generated.
sensorMac	The MAC address of the sensor which saw the client device first.
siteId	The unique ID assigned to the site by ExtremeLocation. This value is used internally and is not for external third party subscribers.
siteName	The name of the site in which the client has been seen.
lastSeen	The time (in epoch milliseconds) when the client was last seen in this site.
timestamp	The time (in epoch milliseconds) when the event was triggered.
rssiValue	The signal strength measured by the sensor for this client device.
loyalty	The calculated loyalty value of the client device. This value can be one of: <ul style="list-style-type: none"> UNKNOWN - Possibly a passerby device with no loyalty information. NEW - A client device which is seen for the first time at this site. REPEAT - A client device which was seen in this site in the last 30 days and is detected again. This device is also known as a repeat visitor.

Device Exit Event

A Device Exit event is generated when the device exits a site and can no longer be seen by the location's sensors. The sensors report information such as the time the client device spent at the site, the type of engagement, the visitor type (first time or repeat visitor), etc.

Sample JSON Content for Wi-Fi Device Exit Event

```

{
  "eventType": "DEVICE_EXIT",
  "clientMac": "80:58:f8:d0:29:29",
  "siteId": "SITE_203a7140654a4db5a1c82ef12264e64b",
  "siteName": "SanJose_Office",
  "firstSeen": "1503898669390",
  "lastSeen": "1503901358434",
  "dwell": 2689044,
  "timestamp": 1503902510362,
  "visitType": "VISITOR",
  "loyalty": "REPEAT",
  "engagement": "TYPE2",
  "bestSensorMac": "74:67:f7:06:fb:96",
  "bestRSSI": -67.0
}

```

Parameter	Description
eventType	The type of event. For a Device Presence event this value is DEVICE_EXIT.
client Mac	The MAC address of the device for which the Device Exit event was generated.
siteld	The unique ID assigned to the site by ExtremeLocation. This value is used internally and is not for external third party subscribers.
siteName	The name of the site in which the client is located.
firstSeen	The time (in epoch milliseconds) when the client was first seen in this site and when the DEVICE_ENTERED event was triggered.
lastSeen	The time (in epoch milliseconds) when the client was last seen in this site.
dwel	The time duration in seconds that the client spent in this site. This is the difference in time (epoch time) between when the client was first seen and last seen in the site.
timestamp	The time (in epoch milliseconds) when the event was triggered.
visitType	The type of client visit. This is classified according to the duration spent in the site: <ul style="list-style-type: none"> • PASSERBY - This classification is for devices that are seen by the location's sensors but are not located inside the site's perimeter. • BOUNCED_VISITOR - This classification is for devices that are seen inside the site but have spent time that is below a configured duration within the site. • VISITOR - This classification is for devices that are seen inside a site and have spent time that is above a configured duration within the site.
loyalty	The calculated loyalty value of the client device. This value can be one of: <ul style="list-style-type: none"> • UNKNOWN - Possibly a passerby device with no loyalty information. • NEW - A client device which is seen for the first time at this site. • REPEAT - A client device which has already been seen in this site before. This device is also known as a repeat visitor.
engagement	The engagement type classification of the client device. This classification is based on the time the client device spends in the site: <ul style="list-style-type: none"> • TYPE0 - This classification is for clients that have spent up to five minutes inside the site. • TYPE1 - This classification is for clients that have spent up to 20 minutes inside the site. • TYPE2 - This classification is for clients that have spent up to one hour (60 minutes) inside the site. • TYPE3 - This classification is for clients that have spent up to six hours (360 minutes) inside the site. • TYPE4 - This classification is for clients that have spent more than six hours (360 minutes) inside the site.
bestSensorMac	The MAC address of the sensor which detected the client device with the highest RSSI value.
bestRSSI	The highest RSSI value recorded by the sensor with the MAC address as in the "bestSensorMac" field.

Category Events

A Category event is generated when a client device enters or exits a category. The Category event can be further classified as a Category Enter event or a Category Exit event.

Category Enter Event

A Category Enter event is generated when a client device enters a region (if configured) for which a category has been assigned.

This event is also triggered when the client device is seen by a sensor (the sensor that reports the highest RSSI value) in a region. If the region has a category assigned to it, then that category has precedence over the category assigned to an access point in that region.

Sample JSON Content for Wi-Fi Category Enter Event

```
{
  "eventType": "CATEGORY_ENTERED",
  "clientMac": "00:6f:64:31:59:5f",
  "sensorMac": "74:67:f7:9a:15:ba",
  "siteName": "SanJose_Office",
  "floorName": "4thFloor",
  "x": 51.913044445216656,
  "y": 20.260869942605495,
  "deviceType": "VISITOR",
  "previousCategories": "[Cali_AP1]",
  "categories": "[Cali_AP2]",
  "firstseen": "1503901450747",
  "lastSeen": "1503902650494",
  "timestamp": 1503902398255,
}
```

Parameter	Description
eventType	The type of event. For a Device Presence event this value is CATEGORY_ENTERED.
clientMac	The MAC address of the device for which the Device Enter event was generated.
sensorMac	The MAC address of the sensor which saw the client device first.
siteName	The name of the site in which the client has been seen.
floorName	The name of the floor in which the client is located.
x	The "X" coordinate of the location where the client is located. This is the geographical coordinate of the location.
y	The "Y" coordinate of the location where the client is located. This is the geographical coordinate of the location.

Parameter	Description
deviceType	The type of client device. This classification is based on rules configured by the user. The valid values are: <ul style="list-style-type: none"> VISITOR - A device that is present at a site for a limited duration and then leaves the site. ASSET - A device that belongs to a site. An asset could be a wireless POS, an access point deployed at a site, or such similar devices. ASSOCIATE - A device that is carried by an associate and used for work related activities. An associate device could be a hand-held device like a wireless bar code scanner, a mobile/tablet used for inventory purposes, or other similar devices.
previousCategories	The name of the category (if any) that the client device is previously identified in. Value is empty if no previous category is found.
categories	The name of the category in which the client device is currently located.
firstSeen	The time (in epoch milliseconds) when the client was first seen in the current category and when the CATEGORY_ENTERED event was triggered.
lastSeen	The time (in epoch milliseconds) when the client was last seen in the current category.
timestamp	The time (in epoch milliseconds) when the event was triggered.

Category Exit Event

A Category Exit event is generated when a client device exits a region (if configured) for which a category has been assigned, or is no longer visible to any sensor that has the same category. The sensor reports information such as the previously assigned category, the time when the device was last seen, the amount of time spent in the category, etc.

Sample JSON Content for Wi-Fi Category Exit Event

```
{
  "eventType": "CATEGORY_EXIT",
  "clientMac": "34:f3:9a:2d:a7:b2",
  "siteName": "SanJose_Office",
  "floorName": "4thFloor",
  "x": 51.913044445216656,
  "y": 20.260869942605495,
  "deviceType": "VISITOR",
  "categories": ["Cali_AP2"],
  "firstSeen": "1503898669390",
  "lastSeen": "1503901358434",
  "dwell": 2689044,
  "timestamp": 1503902510362,
  "engagement": "TYPE2"
}
```

Parameter	Description
eventType	The type of event. For a Category Exit event this value is CATEGORY_EXIT.
clientMac	The MAC address of the device for which the Category Exit event was generated.
siteName	The name of the site in which the client was seen in.

Parameter	Description
floorName	The name of the floor in which the client was seen in.
x	The "X" coordinate of the location where the client was last located. This is the geographical coordinate of the location.
y	The "Y" coordinate of the location where the client was last located. This is the geographical coordinate of the location.
deviceType	The type of client device. This classification is based on rules configured by the user. The valid values are: <ul style="list-style-type: none"> VISITOR - A device that is present at a site for a limited duration and then leaves the site. ASSET - A device that belongs to a site. An asset could be a wireless POS, an access point deployed at a site, or such similar devices. ASSOCIATE - A device that is carried by an associate and used for work related activities. An associate device could be a hand-held device like a wireless bar code scanner, a mobile/tablet used for inventory purposes, or other similar devices.
categories	The name of the category the client device was last identified in.
firstSeen	The time (in epoch milliseconds) when the client was first seen in the current category and when the CATEGORY_ENTERED event was triggered.
lastSeen	The time (in epoch milliseconds) when the client was last seen in the current category.
dwel	The time duration in seconds that the client spent in this category. This is the difference in time (epoch time) between when the client was first seen in the category and last seen.
timestamp	The time (in epoch milliseconds) when the event was triggered.
engagement	The engagement type of the client device. This classification is based on the time the client device spends in the site: <ul style="list-style-type: none"> TYPE0 - This classification is for clients that have spent up to five minutes inside the site. TYPE1 - This classification is for clients that have spent up to 20 minutes inside the site. TYPE2 - This classification is for clients that have spent up to one hour (60 minutes) inside the site. TYPE3 - This classification is for clients that have spent up to six hours (360 minutes) inside the site. TYPE4 - This classification is for clients that have spent more than six hours (360 minutes) inside the site.

Crowding Events

A Crowding event is generated when the ratio of the number of visitor devices to the number of associate devices exceeds the configured threshold. The Crowding event threshold can be set either globally or for a category.

Sample JSON Content for Wi-Fi Device Crowding Event

```
{
  "siteName": "SanJose_Office",
  "category": "Cali_AP2",
  "eventType": "CROWDING_EVENT",
  "timestamp": 1513801441993
}
```

Parameter	Description
siteName	The name of the site for which the Crowding event is generated.
category	The name of the category for which the Crowding event is generated.
eventType	The type of event. For a Crowding event this value is CROWDING_EVENT.
timestamp	The time (in epoch milliseconds) when the event was triggered.

BLE Outbound API Events

The events generated by the ExtremeLocation Outbound APIs using BLE positioning can be classified as follows:

- [BLE Location Events](#) on page 16
- [BLE Presence Events](#) on page 17
- [BLE Category Events](#) on page 19
- [BLE Crowding Events](#) on page 21

BLE Location Events

A BLE Location event is generated when a Bluetooth-enabled mobile device that is running the Extreme loyalty app, is in close proximity to any beacon(s) deployed at the site.

Sample JSON Content for BLE Location Event

```
{
  "eventType": "BLE_LOCATION_EVENT",
  "siteName": "Site1",
  "floorName": "2nd Floor",
  "deviceUID": "290C5D1D-2AE7-4241-8BC2-A81AFEF5517A",
  "deviceType": "AppUser",
  "x": 57.72102355957031,
  "y": 39.44526290893555,
  "timestamp": 1535603077213,
  "app": "5b710b822ef18c2ab8e180d7",
  "isIDFA": false,
  "isMasked": false,
  "user": "",
  "userId": "5b7119003fd99341403f0a32"
}
```


Parameter	Description
eventType	The type of event. For a location event this value is BLE_LOCATION_EVENT.
siteName	The name of the site in which the client is located.
floorName	The name of the floor in which the client is located.
deviceUID	Identifier for Advertisers - IDFA (or fallback Device ID), which is a unique ID for a user.
deviceType	The type of device. For BLE, this will always be AppUser.
x	The "X" coordinate of the location where the client is located. This is the positional coordinate of the location relative to the floor map's origin (0,0) point.
y	The "Y" coordinate of the location where the client is located. This is the positional coordinate of the location relative to the floor map's origin (0,0) point.
timestamp	The time (in epoch milliseconds) when the event was triggered.
app	The ID generated by the system to identify the app.
isIDFA	A flag to indicate if the DeviceUID is stored as an IDFA.
isMasked	A flag to indicate if the DeviceUID is masked or in its original format.
user	Optional app field that distinguishes a user from a third-party system.
userId	The ID generated by the system to identify the user.

BLE Presence Events

A BLE Presence event is generated when a client device enters or exits a site deployed with BLE beacons. A BLE Presence event can be further classified as a BLE Device Enter or a BLE Device Exit event.

BLE Device Enter Event

A BLE Device Enter event is generated when a device is first seen by the beacons deployed at a site.

Sample JSON Content for BLE Device Enter Event

```
{
  "eventType": "BLE_DEVICE_ENTERED",
  "siteName": "Site1",
  "timestamp": 1535457043212,
  "firstSeen": "1535603077213",
  "deviceType": "AppUser",
  "loyalty": "REPEAT",
  "user": "",
  "userId": "5b7119003fd99341403f0a32",
  "deviceUID": "290C5D1D-2AE7-4241-8BC2-A81AFEF5517A",
  "isMasked": false,
  "isIDFA": false
}
```

Parameter	Description
eventType	The type of event. For a Device Presence event this value is BLE_DEVICE_ENTERED.
siteName	The name of the site in which the client is located.
timestamp	The time (in epoch milliseconds) when the event was triggered.
firstSeen	The time (in epoch milliseconds) when the client device is first seen at the site.
deviceType	The type of device. For BLE, this will always be AppUser.
loyalty	The calculated loyalty value of the client device. This value can be one of: <ul style="list-style-type: none"> UNKNOWN - Possibly a passerby device with no loyalty information. NEW - A client device which is seen for the first time at this site. REPEAT - A client device, if ever seen previously in this site, is known as a repeat visitor.
user	Optional app field that distinguishes a user from a third-party system.
userId	The ID generated by the system to identify the user.
deviceUID	Identifier for Advertisers - IDFA (or fallback Device ID), which is a unique ID for a user.
app	The ID generated by the system to identify the app.
isMasked	A flag to indicate if the DeviceUID is masked or in its original format.
isIDFA	A flag to indicate if the DeviceUID is stored as an IDFA.

BLE Device Exit Event

A BLE Device Exit event is generated when the device stays in a site deployed with beacons for more than the configured visit duration. A BLE Device Enter event will be triggered for the next visit.

Sample JSON Content for BLE Device Exit Event

```
{
  "eventType": "BLE_DEVICE_EXIT",
  "siteName": "Site1",
  "lastSeen": "1535603077213",
  "timestamp": 1535546239106,
  "firstSeen": "2018-08-28T17:20:43.204Z",
  "dwell": 69770495,
  "deviceType": "AppUser",
  "loyalty": "REPEAT",
  "user": "",
  "userId": "5b7119003fd99341403f0a32",
  "deviceUID": "290C5D1D-2AE7-4241-8BC2-A81AFEF5517A",
  "isMasked": false,
  "isIDFA": false
}
```

Parameter	Description
eventType	The event type. For a Device Exit event this value is BLE_DEVICE_EXIT.
siteName	The name of the site in which the client is located.

Parameter	Description
lastSeen	The time (in epoch milliseconds) when the client was last seen in the site for current visit.
timestamp	The time (in epoch milliseconds) when the event was triggered.
firstSeen	The time (in epoch milliseconds) when the client was first seen in the current site and when the BLE_DEVICE_ENTERED event was triggered for the visit.
dwll	The difference in milliseconds between firstSeen and lastSeen time values.
deviceType	The type of device. For BLE, this will always be AppUser.
loyalty	The calculated loyalty value of the client device. This value can be one of: <ul style="list-style-type: none"> UNKNOWN - Possibly a passerby device with no loyalty information. NEW - A client device which is seen for the first time at this site. REPEAT - A client device which was seen in this site in the last 30 days and is detected again. This device is also known as a repeat visitor.
user	Optional app field that distinguishes a user from a third-party system.
userId	The ID generated by the system to identify the user.
deviceUID	Identifier for Advertisers - IDFA (or fallback Device ID), which is a unique ID for a user.
isMasked	A flag to indicate if the DeviceUID is masked or in its original format.
isIDFA	A flag to indicate if the DeviceUID is stored as an IDFA.

BLE Category Events

A BLE Category event is generated when a client device enters or exits a category. The Category event can be further classified as a BLE Category Enter event or a BLE Category Exit event.

BLE Category Enter Event

A BLE Category Enter event is generated when a client device is seen by a beacon which is mapped to a category. There can be multiple category enter events for a single client.

Sample JSON Content for BLE Category Enter Event

```
{
  "eventType": "BLE_CATEGORY_ENTERED",
  "siteName": "Site1",
  "user": "",
  "userId": "5b7119003fd99341403f0a32",
  "deviceUID": "290C5D1D-2AE7-4241-8BC2-A81AFEF5517A",
  "app": "5b710b822ef18c2ab8e180d7",
  "isIDFA": false,
  "isMasked": false,
  "deviceType": "AppUser",
  "categories": [
    "Belli's Cude"
  ],
  "firstSeen": "1538642512000",
  "lastSeen": "1538642512000",
}
```

```

    "timestamp":1535578567761
}

```

Parameter	Description
eventType	The event type. For a BLE Category Enter event this value is BLE_CATEGORY_ENTERED.
siteName	The name of the site in which the client is located.
user	Optional app field that distinguishes a user from a third-party system.
userId	The ID generated by the system to identify the user.
deviceUID	Identifier for Advertisers - IDFA (or fallback Device ID), which is a unique ID for a user.
app	The ID generated by the system to identify the app.
isIDFA	A flag to indicate if the DeviceUID is stored as an IDFA.
isMasked	A flag to indicate if the DeviceUID is masked or in its original format.
deviceType	The type of device. For BLE, this will always be AppUser.
categories	The name of the category in which the client device is located.
firstSeen	The time (in epoch milliseconds) when the client was first seen in the current category.
lastSeen	The time (in epoch milliseconds) when the client was last seen the current category.
timestamp	The time (in epoch milliseconds) when the event was triggered.

BLE Category Exit Event

A BLE Category Exit event is generated when a client device moves away from beacon(s) for which category has been assigned.

Sample JSON Content for BLE Category Exit Event

```

{
  "eventType":"BLE_CATEGORY_EXIT",
  "siteName": "Site1",
  "user": "",
  "userId":"5b7119003fd99341403f0a32",
  "deviceUID":"290C5D1D-2AE7-4241-8BC2-A81AFEF5517A",
  "app":"5b710b822ef18c2ab8e180d7",
  "isIDFA": false,
  "isMasked": false,
  "deviceType":"AppUser",
  "categories": [
    "Belli's Cude"
  ],
  "firstSeen":"1538642512000",
  "lastSeen":"1538642512000",
  "dwell": 2342082,
  "timestamp":1535578609908
}

```

Parameter	Description
eventType	The event type. For a BLE Category Exit event this value is BLE_CATEGORY_EXIT.
siteName	The name of the site in which the client is located.
user	Optional app field that distinguishes a user from a third-party system.
userId	The ID generated by the system to identify the user.
deviceUID	Identifier for Advertisers - IDFA (or fallback Device ID), which is a unique ID for a user.
app	The ID generated by the system to identify the app.
isIDFA	A flag to indicate if the DeviceUID is stored as an IDFA.
isMasked	A flag to indicate if the DeviceUID is masked or in its original format.
deviceType	The type of device. For BLE, this will always be AppUser.
categories	The name of the category in which the client device is located.
firstSeen	The time (in epoch milliseconds) when the client was first seen in the current category.
lastSeen	The time (in epoch milliseconds) when the client was last seen the current category.
dwel	The time duration in milliseconds that the client spent in this category. This is the difference in time between when the client was first seen and last seen in the category.
timestamp	The time (in epoch milliseconds) when the event was triggered.

BLE Crowding Events

A BLE Crowding event is generated when the ratio of the number of visitor devices to the number of associate devices (as classified and seen by Wi-Fi location) exceeds the configured threshold. The Crowding event threshold can be set either globally or for a category. If there is no visitor count, the crowding event is generated if the number of visitors exceeds the minimum visitor count configured in the **Settings** dialog in the ExtremeLocation GUI.


Sample JSON Content for BLE Device Crowding Event

```
{
  "eventType": "BLE_CROWDING_EVENT",
  "timestamp": 1535562069801,
  "category": "Roshan\u0027s cube"
}
```

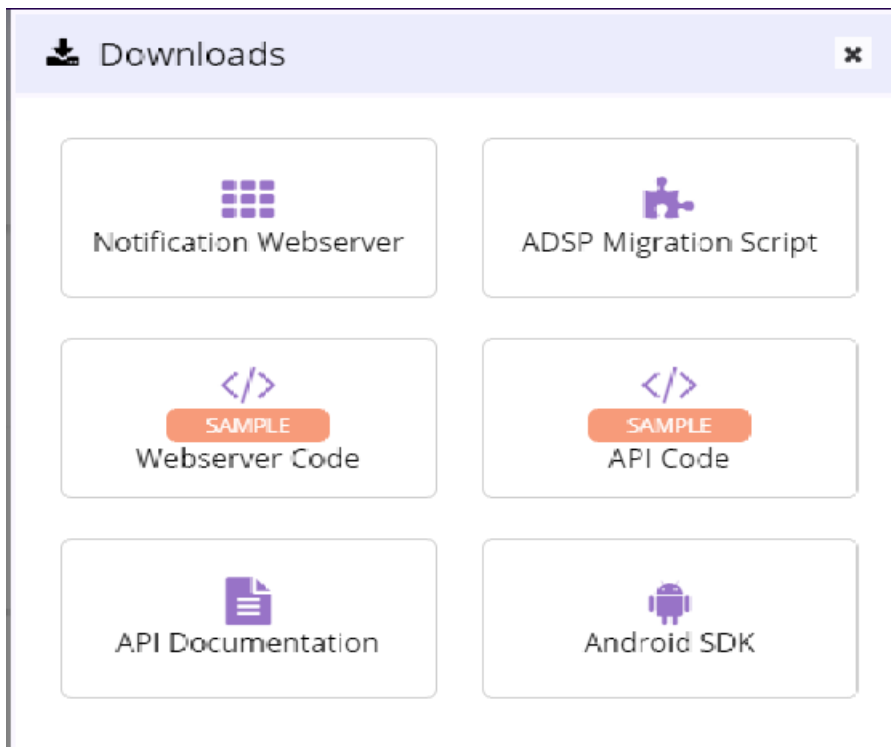
Parameter	Description
eventType	The event type. For a BLE Crowding event this value is BLE_CROWDING_EVENT.
timestamp	The time (in epoch milliseconds) when the event was triggered.
category	The name of the category for which the Crowding event is generated.

Creating the Sample Web Application

You can use these sample JAVA classes and Java Server Pages (JSP) pages to create a sample web server application for handling notification events from ExtremeLocation. These resources are available for download from the ExtremeLocation user interface. To download these resources:

- 1 In the ExtremeLocation GUI, click  on the top right of the screen.
- 2 Click **Downloads**.

The **Downloads** screen displays.



- 3 Click on the appropriate link to download the required software.

Class Definitions

Class Name	Description
<code>LocationingEvent.java</code>	This class is a Plain Old Java Object (POJO) class with fields and 'get' and 'set' methods to access these fields. This class is meant for storing location events. You can use this POJO to add this event type to the list of events object and then use it in the JSP to display the location events.
<code>BLELocationingEvent.java</code>	This POJO represents BLE location events. You can use it to add this event type to the list of events object and then use it in the JSP to display location events.
<code>PresenceEvent.java</code>	This POJO represents the 'DEVICE_ENTERED' & 'DEVICE_EXIT' events. You can use it to add these event types to the list of events object and then use it in the JSP to display presence events.
<code>BLEPresenceEvent.java</code>	This POJO represents the 'BLE_DEVICE_ENTERED' & 'BLE_DEVICE_EXIT' events. You can use it to add these event types to the list of events object and then use it in the JSP to display presence events.
<code>NotificationStore.java</code>	This is a utility class that stores multiple events as list objects of different types.
<code>NotificationSubscriberEx.java</code>	This servlet performs POST operations and retrieves the request header from the request object. It then parses the request body to get the JSON sent from the server. It parses the JSON object to identify the event type and the specific event. The event data is stored in their respective JAVA classes and saved into the list object that is used for displaying the events.
<code>ZoneNotificationData.java</code>	This POJO represents category events such as 'CATEGORY_ENTERED' & 'CATEGORY_EXIT'. You can use this object to add these event types to the list of events object and then use it in the JSP to display category events.
<code>BLECategoryEvent.java</code>	This POJO represents BLE category events such as 'BLE_CATEGORY_ENTERED' & 'BLE_CATEGORY_EXIT'. You can use this object to add these event types to the list of events object and then use it in the JSP to display category events.
<code>ZoneNotification.jsp</code>	This Java Server Page (JSP) is responsible for displaying event data and their numbers.
<code>BLENotification.jsp</code>	This Java Server Page (JSP) is responsible for displaying event data and their numbers for BLE Events.

Setting Up the Remote Web Server

The sample web server requires the following components:

- Apache Tomcat Server
- An IDE that supports JAVA development (Eclipse)

- Java Development Kit version 1.8 and above

The server also requires the following JAVA components.

- common-codec-1.6.jar
- commons-logging-1.1.1.jar
- gson-2.3.jar
- httpclient-4.2.5.jar
- httpclient-cache-4.2.5.jar
- httpcore-4.2.4.jar
- httpmime-4.2.5.jar
- jackson-annotations-2.2.0.jar
- jackson-core-2.2.0.jar
- jackson-databind-2.2.0.jar
- javax.servlet-api-3.1.0.jar
- log4j-1.2.16.jar

Logging requires the following properties file:

- log4j.properties

The class files required for Apache Tomcat support are included in the Eclipse project file.

**Note**

Ensure that you have a working installation of the Eclipse IDE able to compile `java` code. When using *Windows*, ensure you have set the `JAVA_HOME` environment variable to the JDK directory where the `javac.exe` file is installed.

To create your sample web server application:

- 1 From the open Eclipse IDE instance, go to **File > Import**.
The **Import** dialog displays.

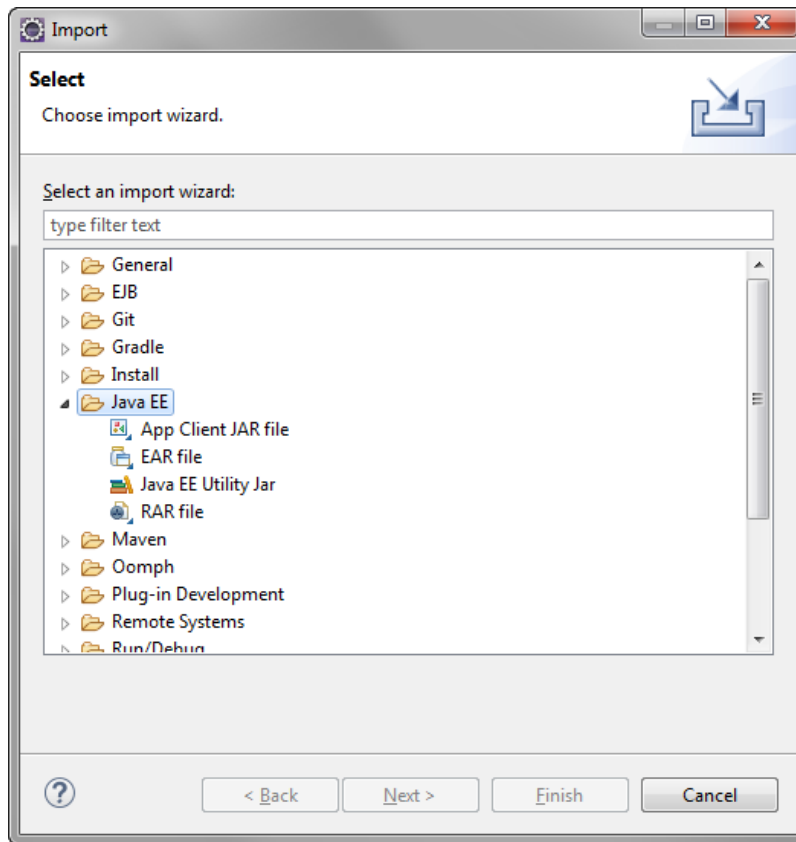


Figure 1: Eclipse IDE - Import Dialog

- 2 Select **General > Existing Projects into Workspace**.

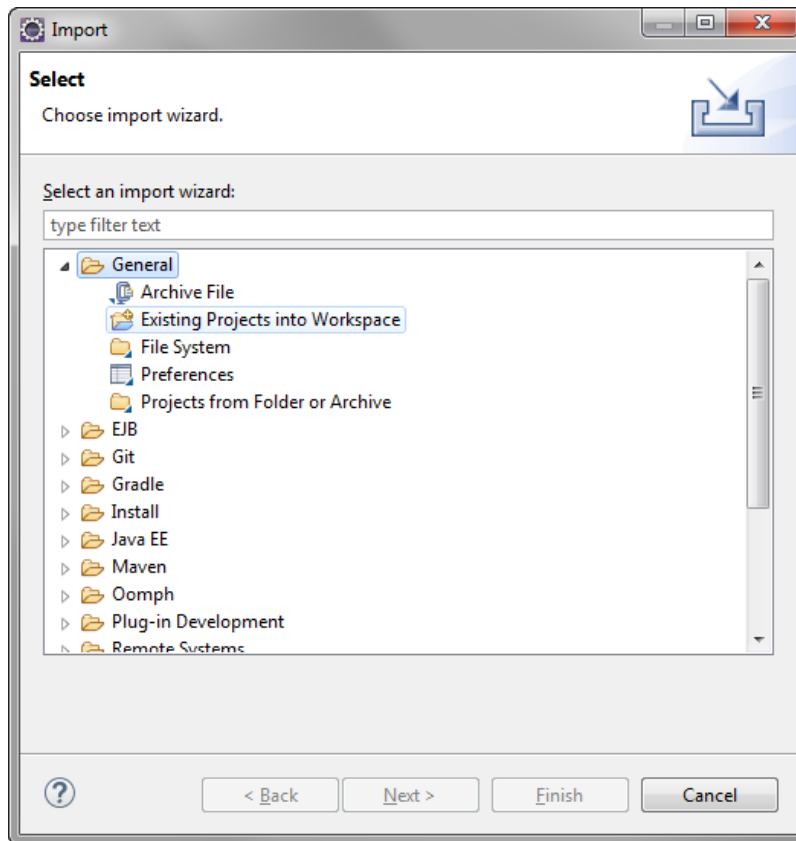


Figure 2: Eclipse IDE - Import Dialog - Select project import type

- 3 Click **Next**.

The **Import Projects** screen displays.

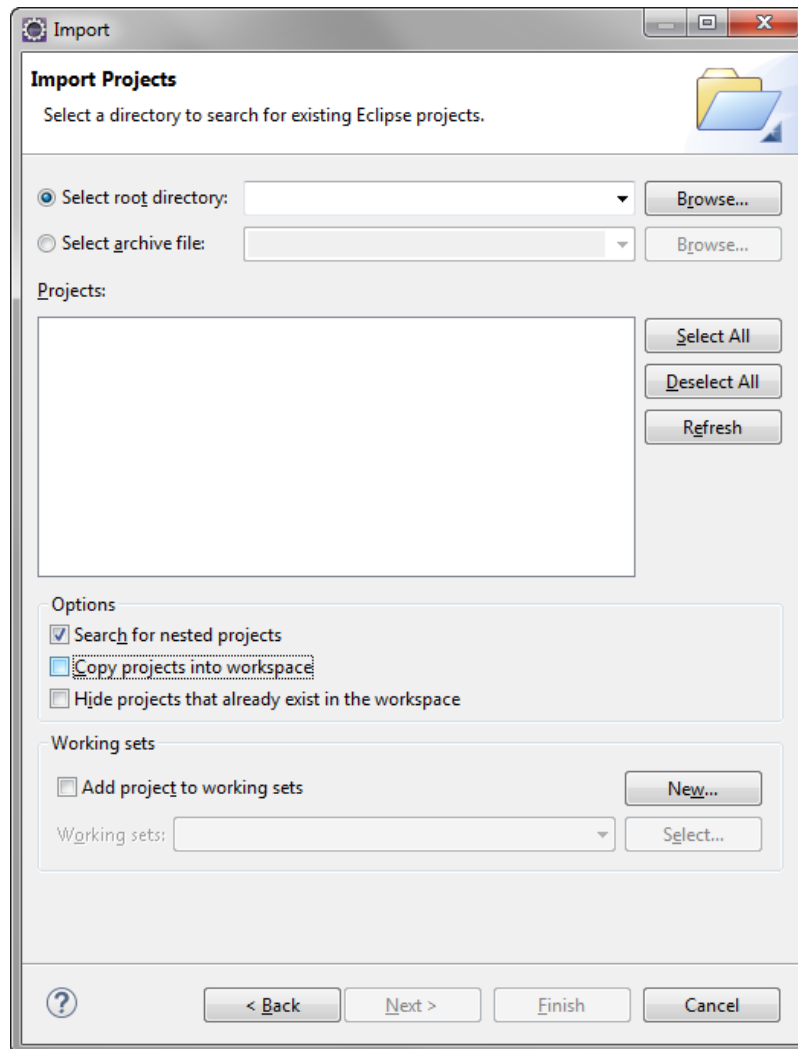


Figure 3: Eclipse IDE - Import Dialog - Import Projects

- 4 Click **Select archive file** and then click **Browse** next to this field.

- 5 Navigate to the directory where you have downloaded the Eclipse Sample Web Application project's archive file and then select it. Click **Finish** to import the archive file into the Eclipse IDE.

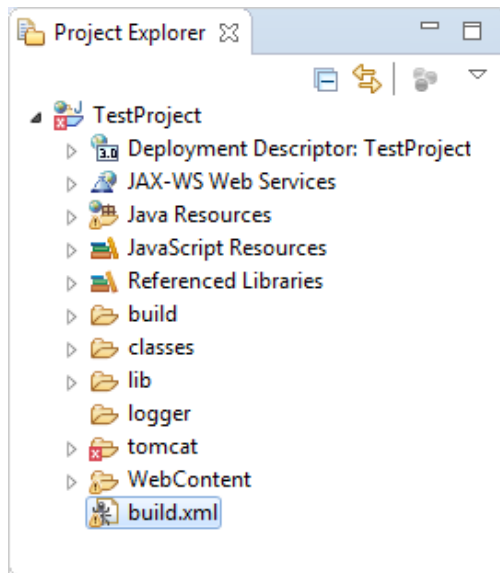


Figure 4: Eclipse IDE - Directory Structure After Project Import

- 6 In the **Project Explorer** window, right click on the `build.xml` file.
- 7 Select **Run As > Ant Build**

The project is compiled. If the project builds successfully, a new folder named **war** is created under the project's main directory.

- 8 Expand the **war** folder.

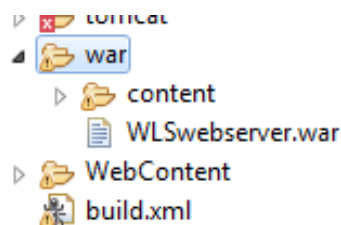


Figure 5: Eclipse IDE - Directory Contents for the .war Folder

- 9 Copy the `.war` file located in this folder to your installation of the Apache Tomcat server.

- 10 Use the following example URL to launch the web application for Wi-Fi events notifications:
<http://localhost:9090/WLSwebserver/ZoneNotification.jsp?presencedata=on&locationdata=on&zonedata=on>

No.	Event Type	Client MAC	Sensor MAC	Site Name	Category Name	X	Y	Last Seen Time	Event Time
249	LOCATION_EVENT	00:01:30:00:00:66		Site_SIM_3	NA	17.0	2.0	03:51:34 09/Oct/2017	03:51:34 09/Oct/2017
248	LOCATION_EVENT	00:01:30:00:00:82		Site_SIM_3	NA	30.0	2.0	03:51:29 09/Oct/2017	03:51:29 09/Oct/2017
247	CATEGORY_ENTERED	00:01:30:00:00:82	22:22:00:00:00:16	Site_SIM_3	Gaming	NA	NA	03:51:29 09/Oct/2017	03:51:29 09/Oct/2017
246	LOCATION_EVENT	00:01:30:00:00:8a		Site_SIM_3	NA	29.0	4.0	03:51:29 09/Oct/2017	03:51:29 09/Oct/2017
245	LOCATION_EVENT	00:01:30:00:00:6f		Site_SIM_3	NA	30.0	9.0	03:51:28 09/Oct/2017	03:51:28 09/Oct/2017
244	CATEGORY_ENTERED	00:01:30:00:00:6f	22:22:00:00:00:17	Site_SIM_3	Gaming	NA	NA	03:51:28 09/Oct/2017	03:51:28 09/Oct/2017
243	LOCATION_EVENT	00:01:30:00:00:91		Site_SIM_3	NA	17.0	2.0	03:51:28 09/Oct/2017	03:51:28 09/Oct/2017
242	LOCATION_EVENT	00:01:30:00:00:80		Site_SIM_3	NA	17.0	2.0	03:51:28 09/Oct/2017	03:51:28 09/Oct/2017
241	LOCATION_EVENT	00:01:30:00:00:90		Site_SIM_3	NA	30.0	2.0	03:51:28 09/Oct/2017	03:51:28 09/Oct/2017
240	CATEGORY_ENTERED	00:01:30:00:00:90	22:22:00:00:00:16	Site_SIM_3	Gaming	NA	NA	03:51:28 09/Oct/2017	03:51:28 09/Oct/2017
239	LOCATION_EVENT	00:01:30:00:00:89		Site_SIM_3	NA	29.0	4.0	03:51:28 09/Oct/2017	03:51:28 09/Oct/2017
238	LOCATION_EVENT	00:01:30:00:00:8d		Site_SIM_3	NA	17.0	2.0	03:51:28 09/Oct/2017	03:51:28 09/Oct/2017
237	LOCATION_EVENT	00:01:30:00:00:77		Site_SIM_3	NA	30.0	9.0	03:51:28 09/Oct/2017	03:51:28 09/Oct/2017

- 11 Use the following URL to launch the web application for BLE events notifications:
<http://13.232.102.170:9090/WLSwebserver/BLENotification.jsp?BLEPresenceData=on&BLELocationData=on&BLECategoryData=on&BLECrowdingData=on>

No.	Event Type	USER	USER ID	SiteName	DEVICE UID	CATEGORY NAME	X	Y	DWELL	FIRST SEEN	LAST SEEN	TIMESTAMP
19255	BLE_DEVICE_EXIT	susil	00:01:30:00:cf:8f	Site_SIM_48	50:22:00:00:13:09	NA	NA	NA	1027000	04:53:51	05:10:58	10:41:01
19254	BLE_DEVICE_EXIT	susil	00:01:30:00:d1:3a	Site_SIM_48	50:22:00:00:12:b5	NA	NA	NA	1036000	04:53:42	05:10:58	10:41:01
19253	BLE_DEVICE_EXIT	susil	00:01:30:00:cf:71	Site_SIM_48	50:22:00:00:13:1b	NA	NA	NA	1026000	04:53:52	05:10:58	10:41:01
19252	BLE_DEVICE_EXIT	susil	00:01:30:00:d1:80	Site_SIM_48	50:22:00:00:13:06	NA	NA	NA	1028000	04:53:50	05:10:58	10:41:01
19251	BLE_DEVICE_EXIT	susil	00:01:30:00:ca:e6	Site_SIM_48	50:22:00:00:11:e5	NA	NA	NA	1033000	04:53:45	05:10:58	10:41:01
19250	BLE_DEVICE_EXIT	susil	00:01:30:00:ca:c5	Site_SIM_48	50:22:00:00:11:be	NA	NA	NA	1035000	04:53:43	05:10:58	10:41:01
19249	BLE_DEVICE_EXIT	susil	00:01:30:00:d1:6e	Site_SIM_48	50:22:00:00:12:d1	NA	NA	NA	1033000	04:53:45	05:10:58	10:41:01
19248	BLE_DEVICE_EXIT	susil	00:01:30:00:cf:6f	Site_SIM_48	50:22:00:00:12:f9	NA	NA	NA	1029000	04:53:49	05:10:58	10:41:01
19247	BLE_DEVICE_EXIT	susil	00:01:30:00:d1:8e	Site_SIM_48	50:22:00:00:12:b4	NA	NA	NA	1012000	04:54:06	05:10:58	10:41:01
19246	BLE_DEVICE_EXIT	susil	00:01:30:00:cf:1d	Site_SIM_48	50:22:00:00:12:eb	NA	NA	NA	1030000	04:53:48	05:10:58	10:41:01
19245	BLE_DEVICE_EXIT	susil	00:01:30:00:cf:41	Site_SIM_48	50:22:00:00:12:12	NA	NA	NA	1028000	04:53:50	05:10:58	10:41:01
19244	BLE_DEVICE_EXIT	susil	00:01:30:00:e9:b9	Site_SIM_48	50:22:00:00:13:28	NA	NA	NA	1024000	04:53:54	05:10:58	10:41:01
19243	BLE_DEVICE_EXIT	susil	00:01:30:00:cf:2c	Site_SIM_48	50:22:00:00:12:9c	NA	NA	NA	1014000	04:54:04	05:10:58	10:41:01

You can now configure the ExtremeLocation subscriber settings to send Location, Presence, Category, and Crowding events to this web server.

Configuring Subscriber Settings

When an event is generated for any site in your ExtremeLocation account, it is sent to your remote web server in real-time. Filters, if any, are applied to the events before being transmitted. Filters are used to control the event types that are passed to your web server. At your end, a web server or service must be configured to receive the data. This web server can then pass on the received data to third party application data processors for further action or it can store the data locally.

Use the Subscriber Settings window to configure the information required for ExtremeLocation to connect to your server. Also use this screen to filter the event types of interest for your deployment.

To configure ExtremeLocation to connect to your remote web server:

- 1 Select **Settings > System Settings > Subscriber Settings**.

The **Subscriber Classification and Management** window opens.

The screenshot shows the 'Subscriber Configuration and Management' settings page. The left sidebar contains a navigation menu with the following items: Settings, User Management, System Settings (with sub-items: Device Classification, Thresholds, and Subscriber Settings), License, and Category. The main content area is titled 'Subscriber Configuration and Management' and contains the following fields and options:

- Subscriber Push URL:** A text input field containing 'http://' and '/WLSwebserver/notification'.
- Username:** A text input field containing 'admin'.
- Password:** A password input field with masked characters (dots).
- Events:** A section with four checked checkboxes: 'Presence', 'Category', 'Location', and 'Crowding'.

At the bottom right of the form, there are two buttons: 'Reset' and 'Save'.

- 2 Enter the **Subscriber Push URL**.

This is the URL of the remote web server where all event types selected in the **Events** field are transferred in real-time.



Note

The web server must listen on one of the standard ports, 443 (https) or 80 (http). ExtremeLocation does not support web servers listening on other ports.

- 3 Enter the valid user name for accessing the remote web server in the **Username** field.
- 4 Enter the password associated with the above username in the **Password** field.
- 5 Select the type of events (Presence, Category, Location, Crowding) that will be pushed to the remote web server in real-time.
- 6 Click **Save** to save these subscriber settings.

Click **Reset** to revert the changes made to the subscriber settings parameters.

3 REST API

API Request Components
API Response Codes
Accessing the REST API
Sites Resource
Floor Resource
Device Resource

The ExtremeLocation API provides a programmatic interface to retrieve and configure a wide range of authentication, site, floor, and device information. It is based on RESTful principles and is organized around the main resources and features from the ExtremeLocation platform's graphical user interface.

The ExtremeLocation API uses standard HTTP methods for requests and responses. API request and response bodies are formatted in JavaScript Object Notation (JSON). To submit API calls, your RESTful API consuming program needs to have logged in using credentials granting full Admin privileges and generated an API key.

This chapter provides information about accessing the API, structure of the API request and response bodies, error codes, and examples.



Note

You cannot run the sample requests in this chapter as-is. Replace call-specific parameters such as API keys and IDs with your own values.

API Request Components

To construct a REST API request, combine the following components:

Component	Description
The HTTP method	<ul style="list-style-type: none">• GET: Return data from the server• POST: Create a new resource on the server• PUT: Update a resource on the server
The base URL of the API	<code>http://<ip>:<port>/</code>
The URI to the resource	The resource to create, update, or query. For example, <code>WiFiHelperServices/surveyfloorplanservice/list/floors</code> .

Component	Description
Path parameters	These variables are part of the full URL path and are used to point to a specific resource within a collection. For example, / surveyfloorplanservice/get/image/ floorplan/{parent}, where {parent} is the path parameter and is substituted with an actual value when making the API call.
Query Parameters	These parameters, if any, appear after a question mark (?) in the API request. The ? followed by the parameters and their values is referred to as the "query string". In the query string, each parameter is listed right after the other with an ampersand (&) separating them. The order of the query string parameters does not matter. For example, / surveyfloorplanservice/list/floors? nodeType=FLOOR&parent=SITE_203a7140 654a4db5a1c82ef12264e64b, where nodeType and parent are query parameters.
HTTP request headers	The following HTTP headers are supported: <ul style="list-style-type: none"> • Accept: Required for operations with a response body, syntax is <code>Accept: application/json</code>. • Content-Type: Required for operations with a request body, syntax is <code>Content-Type: application/json</code>. • Authorization: Required to pass the API key when making API calls.
JSON request body	Required for most POST and PUT requests.

API Response Codes

The ExtremeLocation REST API returns standard as well as some custom HTTP status codes. Some API calls also return JSON response bodies that include information about the resource.

Table 3: HTTP Response Status Codes

Code	Description
200 OK	The request was successful
201 Created	The resource was created successfully
204 No Content	Success with no response body
400 Bad Request	The operation failed because the request is syntactically incorrect or violated schema.
401 Unauthorized	The authentication credentials are invalid or the user is not authorized to use the API
404 Not Found	The server did not find the specified resource that matches the request URI

Table 3: HTTP Response Status Codes (continued)

Code	Description
405 Method Not Allowed	The API does not support the requested HTTP method, for example, PATCH.
500 Internal Server Error	An exception occurred in the server which lead to this error.
599 Session Expired	User session has expired and needs to reauthenticate.

Accessing the REST API

You can use any language or library that can submit REST API requests and process JSON to query the ExtremeLocation REST API. Examples of languages and libraries that can build REST API clients include:

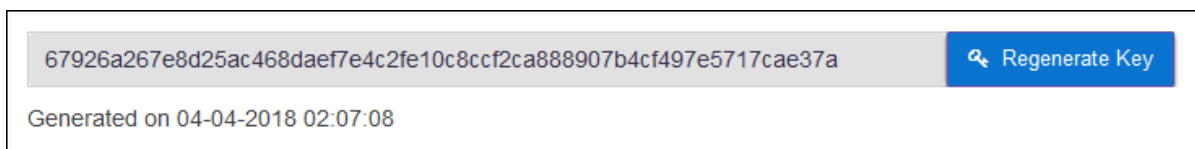
- For Java, the Jersey library provides the reference implementation of JAX-RS, a Java standard for RESTful web services. The implementation includes a client library that can run directly on the JVM.
- For Python, the Requests and JSON libraries facilitate REST API applications.
- For .Net, the core language provides facilities for submitting HTTP requests, and .Net libraries include a serializer for JSON.
- For the Linux shell, Wget and cuRL can execute REST API calls. Linux shell utilities, like awk and grep, can parse and process JSON.

You can also use tools like Postman, an easy-to-use [Chrome extension](#) for making HTTP requests.

To access the ExtremeLocation REST API server and make API calls:

- 1 Log in to the ExtremeLocation GUI using full Admin credentials.
- 2 Get your API key.

The API key for your account can be found on the **Settings > System Settings > API Keys** screen.



- 3 Forward the API key as part of the authorization header every time you make REST API calls.

Sites Resource

The Sites resource provide methods to retrieve the following site specific information:

- Complete list of sites that the user has created
- Complete list of site groups that the user has created
- Complete list of categories that the user has created across all the sites

Get Site List

GET `http:// <ip>:<port>/eloc-api/v1/sites`

Retrieve list of sites managed by this user account.



Note

Get your **API key** from the **System Settings** menu in the ExtremeLocation GUI, and forward it within the Authorization header for each API call for this resource.

Sample Response (200 OK)

The following is a sample response from the `/sites` endpoint.

```
[{
  "id": "SITE_a7cf634551ce4e8aa213ff522afac133",
  "nodeType": "SITE",
  "ancestors": [ "12344321" ],
  "parent": "12344321",
  "bleConfig": {
    "id": "5b5e9b97c44d4914e40f35fc",
    "name": "SiteTest",
    "description": "TestT",
    "city": "Kulhui",
    "state": "Madhya Pradesh",
    "postalCode": "486886",
    "street": "Parsauna Road",
    "country": "India",
    "hasGeofence": true,
    "lat": 47.87214396888731,
    "lng": 1.7578125,
    "timezone": "Asia/Kolkata",
    "visitDuration": 60,
    "captureDuration": 1800,
    "bounceDuration": 3,
    "walkByDuration": 1,
    "captureMinZoneCount": 1,
    "physicalWebUrl": "",
    "zoneCount": 1,
    "beaconCount": 0,
    "created": "2018-07-30T05:01:11.101Z",
    "updated": "2018-08-14T09:50:44.627Z",
    "isActive": true,
  },
  "name": "SiteTest",
  "geoLocations": [ {
    "lat": "47.87214396888731",
    "lon": "1.7578125"
  } ],
  "geoAddress": {
    "street": "Parsauna Road",
    "state": "Madhya Pradesh",
    "postalCode": "486886",
    "country": "India",
    "city": "Kulhui"
  },
  "timezone": "Asia/Kolkata",
  "today": 0,
  "yesterday": 0,
  "yesterdayRelative": 0,
  "trend": 0,
  "todayString": "Today's visits (till Aug 16 13:00:37 IST)",
  "yesterdayRelativeString": "Yesterday's visits (till Aug 15 13:00:37 IST)",
  "apCount": 0,
  "beaconCount": 0,
  "license": 0,
}
```

```

"geofence": {
  "id": "5b5e9b98da441d132c049989",
  "name": "SiteTest_geofenc",
  "radius": 150,
  "vertex": {
    "x": 47.87214396888731,
    "y": 1.7578125
  },
  "area": 70650,
  "created": "2018-07-30T05:01:12.941Z",
  "updated": "2018-07-30T05:02:47.533Z",
  "isActive": true,
}
}
}

```

The following table describes each item in the response:

Table 4: Response Definitions

Parameter	Description	Data Type
id	Unique ID of the site generated by ExtremeLocation system.	String
nodeType	The type of node. For site it is "SITE".	String
ancestors	Ancestors of the site. The ancestor for site is Tenant.	String Array
parent	Parent of the site. Site's parent is Tenant.	String
bleConfig	Details related to only BLE site configuration.	Object
id	Unique ID of the site created for BLE Site Config.	String
name	Name of the Site in BLE Config.	String
description	Description of the Site in BLE Config.	String
city	Name of the city where the site is located in BLE Config.	String
state	Name of the state where the site is located in BLE Config.	String
street	Name of the street where the site is located in BLE Config.	String
country	Name of the country where the site is located in BLE Config.	String
postalCode	Postal Code of the city where the site is located in BLE Config.	String
hasGeofence	A flag to indicate if geofence for this site exists in BLE Config.	Boolean
lat	Latitude of the location of the site in BLE Config.	Number
lng	Longitude of the location of the site in BLE Config.	String
timezone	Timezone of the site in BLE Config.	String

Table 4: Response Definitions (continued)

Parameter	Description	Data Type
visitDuration	Time duration, in minutes, being a visit is considered inactive in BLE Config.	Number
captureDuration	The duration, in minutes, required before a user is considered a capture in BLE Config.	Number
bounceDuration	If the visit time, in minutes, is less than this value the user is considered a bounce in BLE Config.	Number
walkByDuration	If the visit time, in minutes, is less than this value the user is considered a walkby in BLE Config.	Number
captureMinZoneCount	The number of zones that a user must hit before being considered a capture in BLE Config.	Number
physicalWebUrl	The physical web address for this site in BLE Config.	String
zoneCount	Number of categories associated with the site.	Number
beaconCount	Number of beacons associated with the site.	Number
created	Date/Time when the site was created in BLE Config.	String
updated	Date/Time when the site was last updated in BLE Config.	String
isActive	Represents if the site is active in BLE Config.	Boolean
name	Name of the site configured by the user.	String
geoLocations	Geo-coordinates of the site.	Object
lat	Latitude of the site location configured by the user.	String
lon	Longitude of the site location configured by the user.	String
geoAddress	Address of the site based on geo-coordinates.	Object
street	Name of the street for site configured by the user.	String
state	Name of the state for site configured by the user.	String
country	Name of the country for site configured by the user.	String

Table 4: Response Definitions (continued)

Parameter	Description	Data Type
city	Name of the city for site configured by the user.	String
postalCode	Postal Code of the site configured by the user.	String
timezone	Timezone of the site.	String
today	Total visits for present day.	Number
yesterday	Total visits for previous day.	Number
yesterdayRelative	Total visits for previous day till current time.	Number
trend	Percentage increase in visits for previous and present day relative to current time.	Number
todayString	Represents the current date/time.	String
yesterdayRelativeString	Represents the date of previous day with current time.	String
apCount	Number of APs in the site.	Number
beaconCount	Number of beacons in the site.	Number
license	Number of AP licenses allotted to the site.	Number
geofence	Geofence details of the site.	Object
id	Unique ID of the geofence.	String
name	Name of the geofence.	String
radius	Radius in meters of the geofence.	Number
vertex	Vertex of the geofence.	Number
area	Area in square meters of the geofence.	Number
created	Creation data/time of the geofence.	String
updated	Last Updated date/time of the geofence.	String
isActive	Active status of the geofence.	Boolean

Possible Response Codes:

- Success (200 OK)
- Bad Request (400)
- Unauthorized (401)

Get Site Group List

GET `http:// <ip>:<port>/eloc-api/v1/siteGroup`

Retrieve list of Site Groups created by this user account. Site Groups are logical grouping of multiple sites that can be managed collectively.



Note

Get your API key from the **System Settings** menu in the ExtremeLocation GUI, and forward it within the Authorization header for each API call for this resource.

Sample Response (200 OK)

The following is a sample response from the /siteGroup endpoint.

```
[{
  "name": "ELoc_SiteGrpTest_1",
  "children": [],
  "sites": [{
    "id": "SITE_ee967d8007a94840bff7eb15343bd77b",
    "name": "ELoc_TestSite_1"
  }]
}]
```

The following table describes each item in the response:

Table 5: Response Definitions

Parameter	Description	Data Type
name	Name of the site group configured by the user.	String
children	Site groups listed under the site group.	String Array
sites	Child sites for the site group.	Object Array
id	Unique ID of the site.	String
name	Name of the site.	String

Possible Response Codes:

- Success (200 OK)
- Bad Request (400)
- Unauthorized (401)

Get Site Categories

GET `http:// <ip>:<port>/eloc-api/v1/categories`

Retrieve a list of site categories created and managed by this user account.



Note

Get your API key from the **System Settings** menu in the ExtremeLocation GUI, and forward it within the Authorization header for each API call for this resource.

Sample Response (200 OK)

The following is a sample response from the `/db/generic/lsense_category_$tid` endpoint.

```
[{
  "name": "Category1",
  "crThrd": 11,
  "beaconCount": 0,
  "apCount": 0,
  "siteCount": 0,
  "experienceCount": 0
}]
```

The following table describes each item in the response:

Table 6: Response Definitions

Parameter	Description	Data Type
name	Name of the category configured by the user.	String
crThrd	Crowding threshold set for Wi-Fi visitors for the category.	Number
beaconCount	Number of beacons associated with the category.	Number
apCount	Number of access points associated with the category.	Number
siteCount	Number of sites associated with the category.	Number
experienceCount	Number of experiences associated with the category.	Number

Possible Response Codes:

- Success (200 OK)
- Bad Request (400)
- Unauthorized (401)

Floor Resource

The Floor resource provide methods to retrieve the following floor specific information:

- Complete list of floors for a specific site
- Floor plan of a specific floor in a specific site
- Complete list of regions configured on a specific floor of a specific site

Get Floors List

POST `http:// <ip>:<port>/WifiHelperServices/surveyfloorplanservice/list/floors`

Retrieve list of floors for a specific site.



Note

Get your **API key** from the **System Settings** menu in the ExtremeLocation GUI, and forward it within the Authorization header for each API call for this resource.

Sample JSON Request Body

```
{
  "parent": "SITE_156990ae940e47dbbe979d02535616a1",
  "nodeType": "FLOOR"
}
```

Table 7: Request Parameters

Parameter	Description	Required/Optional	Data Type
nodeType	The type of node for which this request is made. For this request, the node type should always be "FLOOR"	Required	String
parent	The unique site ID for the site.	Required	String

Sample Response (200 OK)

The following is a sample response from the `/surveyfloorplanservice/list/floors` endpoint.

```
{
  "total": 3,
  "floors": [
    {
      "floorNumber": 1,
      "floorName": "4th Floor",
      "ceilingHeight": 3.048,
      "backgroundImages": [
        { "fileName": "FLOOR_d2b6ddcbde2441619e263ad4bd4a41d0" }
      ],
      "regions": [
        {
          "regionname": "Bakery",
          "categories": ["Calibration_Reg1"],
          "description": "The Bakery",
          "bgcolor": "ffff00",
          "x1": 37.23,
          "y1": 24.41,
          "x2": 42.65,
          "y2": 30.4
        },
        {
          "regionname": "Toys",
          "categories": ["Calibration_Reg2"],
          "description": "The Toys Section",
          "bgcolor": "00ffa2",

```



```

        "x1":37.32,
        "y1":18.66,
        "x2":42.4,
        "y2":23.74
    },
    {
        "regionname":"Male Clothing",
        "categories": ["Calibration_Reg3"],
        "description":"The Male clothing section",
        "bgcolor":"ff1dd7",
        "x1":63.39,
        "y1":18.33,
        "x2":69.05,
        "y2":30.07
    },
    {
        "regionname":"FreshProduce",
        "categories": ["Calibration_Reg3"],
        "description":"Fresh Fruits & Vegetables",
        "bgcolor":"ff1800",
        "x1":14.17,
        "y1":22.46,
        "x2":23.19,
        "y2":29.08
    }
],
"scale":14.41,
"bounds":
{
    "bottomLeft":{"x":0.0,"y":0.0},
    "topRight":
    {"x":71.0617626648161,
    "y":56.14156835530881
    }
},
"gridMeta":
{
    "cols":8,"rows":6},
    "id":"FLOOR_d2b6ddcbde2441619e263ad4bd4a41d0",
    "nodeType":"FLOOR",
    "ancestors":
    [
        "dd59b67dca9b4686810adc46d3cd530a",
        "SITE_203a7140654a4db5a1c82ef12264e64b"
    ],
    "parent":"SITE_203a7140654a4db5a1c82ef12264e64b"
}
],
"success":true
}

```

The following table describes each item in the response:

Table 8: Response Definitions

Parameter	Description	Data Type
total	The total number of floors in the site.	Integer
floors	The list of floors in the site.	Top level object
success	A flag to indicate if the request was successful or failed.	Boolean

Table 9: Floors object: Represents a floor in the list

Parameter	Description	Data Type
floorNumber	Number of the floor in the Site.	Integer
floorName	Name of the floor.	String
ceilingHeight	Ceiling height of the floor in meters.	Integer
backgroundImages	List of names of the background images of the floor.	Object Array
fileName	Name of the background images of the floor.	String
regions	List of regions configured by the user on the floor.	Object Array
regionname	Name of the region on the floor.	String
categories	List of categories assigned to the region.	String Array
description	Description of the region.	String
bgcolor	Color assigned to the region to represent on the floor.	String
x1	X coordinate of the lower-left corner of the region.	Integer
y1	Y coordinate of the lower-left corner of the region.	Integer
x2	X coordinate of the top-right corner of the region.	Integer
y2	Y coordinate of the top-right corner of the region.	Integer
scale	Scale value configured by user. This value represents the number of pixels per meter.	Float
floorDescription	Description of the floor.	String
bounds	Bottom-left and top-right boundaries of the floor.	Object
gridMeta	The total number of rows and columns in the floor per 10 square meters.	Object
cols	The number of columns in the floor.	Integer

Table 9: Floors object: Represents a floor in the list (continued)

Parameter	Description	Data Type
rows	The number of rows in the floor.	Integer
id	The unique ID of the floor generated by ExtremeLocation.	String
nodeType	The node type of the floor.	String
ancestors	The IDs of the ancestors of the floor. Ancestors of the floor are Site and Tenant.	String Array
parent	The ID of the parent of the floor. Parent of the floor is Site.	String

Possible Response Codes:

- Success (200 OK)
- Bad Request (400)
- Unauthorized (401)

Get Floor Plan

POST `http:// <ip>:<port>/WifiHelperServices/surveyfloorplanservice/get/image/floorplan`

Retrieve the floor plan of a floor in a specific site.

**Note**

Get your **API key** from the **System Settings** menu in the ExtremeLocation GUI, and forward it within the Authorization header for each API call for this resource.

Sample JSON Request Body

```
{
  "parent": "SITE_4b968d6f42ae40a2bff1c49b445571b2",
  "id": "FLOOR_b41b5ef27e4d4ef8b3b31eda96a5fd24",
  "nodeType": "FLOOR"
}
```

Table 10: Request Parameters

Parameter	Description	Required/Optional	Data Type
id	The unique ID of the floor for which you require the floor plan	Required	String
parent	The unique site ID for the site	Required	String
nodeType	The type of node of the floor	Required	String

Sample Response (200 OK)

Returns the image of the floor plan.

```
{
  "code": 200,
  "status": "OK",
  "success": true
}
```

Possible Response Codes:

- Success (200 OK)
- Bad Request (400)
- Unauthorized (401)

Get Regions List

POST `http:// <ip>:<port>/WifiHelperServices/get/floor/regions`

Retrieve a list of regions assigned to the floors in a site.



Note

Get your API key from the **System Settings** menu in the ExtremeLocation GUI, and forward it within the Authorization header for each API call for this resource.

Sample JSON Request Body

```
{
  "parent": "SITE_290b49393de14c4a89681769879737a9",
  "nodeType": "FLOOR",
  "id": "FLOOR_9934a6ca181e457a99dbf01937091043"
}
```

Table 11: Request Parameters

Parameter	Description	Required/Optional	Data Type
nodeType	The type of node. For this request, the node type should always be "FLOOR".	Required	String
id	The unique ID of the floor for which you require the regions.	Required	String
parent	The unique site ID for the site.	Required	String

Sample Response (200 OK)

The following is a sample response from the `/get/floor/regions` endpoint.

```
{
  "total": 2,
  "regions": [
    {
      "regionname": "FoodCourt Region",
      "categories": [ "FoodCourt" ],
      "description": "Regions",
    }
  ]
}
```

```

        "bgcolor": "cyan",
        "x1": 2,
        "y1": 2,
        "x2": 32,
        "y2": 6
    },
    {
        "regionname": "Gaming Region",
        "categories": [ "Gaming" ],
        "description": "Regions",
        "bgcolor": "cyan",
        "x1": 2,
        "y1": 8,
        "x2": 32,
        "y2": 12
    }
],
"regClientsMap": {
    "FoodCourt Region": 0,
    "Gaming Region": 0
},
"success": true
}

```

The following table describes each item in the response:

Table 12: Response Definitions

Parameter	Description	Data Type
total	Total number of regions on the floor.	Integer
regions	List of regions configured by the user on the floor.	Object Array
success	A flag to indicate if the request was successful or failed.	Boolean

Table 13: Regions object: Represents a region in the list

Parameter	Description	Data Type
regionname	Name of the region on the floor.	String
categories	List of categories assigned to the region.	String Array
description	Description of the region.	String
bgcolor	Color assigned to the region to represent on the floor.	String
x1	X coordinate of the lower-left corner of the region.	Integer
y1	Y coordinate of the lower-left corner of the region.	Integer
x2	X coordinate of the top-right corner of the region.	Integer

Table 13: Regions object: Represents a region in the list (continued)

Parameter	Description	Data Type
y2	Y coordinate of the top-right corner of the region.	Integer
regClientsMap	List of mapping of region to clients count in the region.	Object

Possible Response Codes:

- Success (200 OK)
- Bad Request (400)
- Unauthorized (401)

Device Resource

The Device resource provides a method to get the list of all devices deployed on a floor. The resource endpoint also returns the exact physical location of the device, the categories assigned to it, and its RSSI threshold.

Get Device List

POST `http:// <ip>:<port>/WifiHelperServices/get/devices`

Retrieve list of devices deployed on a floor in a specific site.

**Note**

Get your **API key** from the **System Settings** menu in the ExtremeLocation GUI, and forward it within the Authorization header for each API call for this resource.

Sample JSON Request Body

```
{
  "parent": "SITE_a77fc2614c90445295b1a543a4d03f53",
  "nodeType": "DEVICE",
  "floorId": "FLOOR_b05c057eb514478d8a5ac61b76ec487f"
}
```

Table 14: Request Parameters

Parameter	Description	Required/Optional	Data Type
parent	The ID of the parent of the device. The parent of the device is the site.	Required	String
nodeType	The node type of the device, which in this case is DEVICE.	Required	String
floorId	The unique ID of the floor for which you require the list of devices.	Required	String

Sample Response (200 OK)

The following is a sample response from the `/get/devices` endpoint.

```
{
  "total":1,
  "devices":[
    {
      "mac":"74:67:f7:9a:12:db",
      "floorId":"FLOOR_d2b6ddcbde2441619e263ad4bd4a41d0",
      "location":{"x":50.527412,"y":28.66415,"z":0.0},
      "categories":["Cali_AP3"],
      "zoneRssiThreshold":-55,
      "id":"7467f79a12db",
      "nodeType":"DEVICE",
      "ancestors":["dd59b67dca9b4686810adc46d3cd530a",
        "SITE_203a7140654a4db5a1c82ef12264e64b"],
      "parent":"SITE_203a7140654a4db5a1c82ef12264e64b"},
      "success":true
    }
  ]
}
```

The following table describes each item in the response:

Table 15: Response Definitions

Parameter	Description	Data Type
total	The total number of devices on the floor.	Integer
devices	A list of devices on the floor.	Top Level Object
success	A flag to indicate if the request was successful or failed.	Boolean

Table 16: Devices object: Represents a device in the list

Parameter	Description	Data Type
mac	The MAC address of the device.	String
floorId	The unique identifier of the floor on which the device is located.	String
location	The location (x,y,z coordinates) of the device on the floor.	Object Array
categories	List of categories assigned to the device.	String Array
zoneRssiThreshold	The RSSI threshold used for SENSOR_ZONE tracking to locate clients.	Integer
id	The unique identifier of the device.	String
nodeType	The node type of the device.	String

Table 16: Devices object: Represents a device in the list (continued)

Parameter	Description	Data Type
ancestors	The IDs of the ancestors of the device. Ancestors of the device are Site and Tenant.	String Array
parent	The ID of the parent of the device. Parent of the device is Site.	String

Possible Response Codes:

- Success (200 OK)
- Bad Request (400)
- Unauthorized (401)