

# Attendance Analytics API

**Author:** Josh S. Sakweli, Backend Lead Team **Last Updated:** 2026-05-23 **Version:** v2.0

**Base URL:** `https://api.nexgate.com/api/v1`

**Short Description:** The Attendance Management API provides comprehensive attendee tracking and analytics for event organizers. Organizers can view overall attendance statistics with per-day and per-ticket-type breakdowns, browse a unified attendance list (present, absent, and partially attended in one view) with rich filtering, and drill into the full check-in history for any individual ticket. The system supports both single-day and multi-day events.

## Hints:

- **Organizer Only:** All endpoints restricted to the event organizer
- **Unified List:** Present and absent tickets are in one list — use `?status=PRESENT|ABSENT|PARTIALLY_ATTENDED` to filter
- **Ticket-Based Rows:** Each row in the list is one ticket, not one person — one buyer can appear multiple times
- **Attendee Number:** Unique human-readable ticket ID = `bookingReference + "-" + ticketSeries` (e.g. `EVT-A3F4B21C-VIP-0042`)
- **Buyer vs Attendee:** `buyer` = who paid, `attendee` = who holds the ticket — can be different people
- **Buyer Types:** `SYSTEM_USER` = online purchase with account, `AT_D00R` = walk-in name only (no account)
- **Day Filtering:** `?dayNumber` is optional — omit for overall view, provide for per-day status
- **Partial Attendance:** `PARTIALLY_ATTENDED` only applies when no `dayNumber` filter is set on multi-day events
- **Pagination:** Default 20 per page
- **Form Response:** `formResponseId` on each attendee/ticket row links to the buyer's applicant form submission — `null` if the event had no form or the form was not yet submitted

---

## Standard Response Format

All API responses follow a consistent structure using our Globe Response Builder pattern:

## Success Response Structure

```
{
  "success": true,
  "httpStatus": "OK",
  "message": "Operation completed successfully",
  "action_time": "2026-05-23T10:30:45",
  "data": {
  }
}
```

## Error Response Structure

```
{
  "success": false,
  "httpStatus": "BAD_REQUEST",
  "message": "Error description",
  "action_time": "2026-05-23T10:30:45",
  "data": "Error description"
}
```

## Standard Response Fields

Field	Type	Description
<code>success</code>	boolean	<code>true</code> for successful operations, <code>false</code> for errors
<code>httpStatus</code>	string	HTTP status name (OK, NOT_FOUND, FORBIDDEN, etc.)
<code>message</code>	string	Human-readable message describing the result
<code>action_time</code>	string	ISO 8601 timestamp of when the response was generated
<code>data</code>	object	Response payload for success, error string for failures

## HTTP Method Badge Standards

- **GET** - `GET` - Green (Safe, read-only operations)
- **POST** - `POST` - Blue (Create new resources)

- **PUT** - **PUT** - Yellow (Update/replace entire resource)
- **PATCH** - **PATCH** - Orange (Partial updates)
- **DELETE** - **DELETE** - Red (Remove resources)

# Endpoints

## 1. Get Attendance Summary

**Purpose:** Returns overall attendance statistics for an event including per-day and per-ticket-type breakdowns. Always covers the full event — no filtering applied.

**Endpoint:** **GET** `/e-events/attendance/{eventId}/summary`

**Access Level:**  Protected — Event Organizer Only

**Authentication:** Bearer Token

**Request Headers:**

Header	Type	Required	Description
<code>Authorization</code>	string	Yes	Bearer <code>&lt;token&gt;</code>

**Path Parameters:**

Parameter	Type	Required	Description	Validation
<code>eventId</code>	UUID	Yes	Event identifier	Must be a valid UUID of an existing event owned by the authenticated organizer

**Success Response JSON Sample:**

```
{
  "success": true,
  "httpStatus": "OK",
  "message": "Attendance summary retrieved",
  "action_time": "2026-05-23T10:30:45",
  "data": {
    "eventId": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
```

```
"eventTitle": "East African Tech Summit 2026",
"totalDays": 3,
"eventSchedule": [
  { "dayNumber": 1, "dayName": "Day 1 - Opening", "date": "2026-12-15" },
  { "dayNumber": 2, "dayName": "Day 2 - Conference", "date": "2026-12-16" },
  { "dayNumber": 3, "dayName": "Day 3 - Closing", "date": "2026-12-17" }
],
"overallStats": {
  "totalTickets": 500,
  "present": 310,
  "partiallyAttended": 45,
  "absent": 145,
  "attendanceRate": 71.0,
  "byDay": [
    {
      "dayNumber": 1,
      "dayName": "Day 1 - Opening",
      "date": "2026-12-15",
      "totalTickets": 500,
      "checkedIn": 400,
      "absent": 100,
      "attendanceRate": 80.0,
      "status": "COMPLETED"
    },
    {
      "dayNumber": 2,
      "dayName": "Day 2 - Conference",
      "date": "2026-12-16",
      "totalTickets": 500,
      "checkedIn": 355,
      "absent": 145,
      "attendanceRate": 71.0,
      "status": "ONGOING"
    },
    {
      "dayNumber": 3,
      "dayName": "Day 3 - Closing",
      "date": "2026-12-17",
      "totalTickets": 500,
      "checkedIn": 0,
```



Field	Description
<code>data.overallStats.absent</code>	Tickets never checked in
<code>data.overallStats.attendanceRate</code>	$((\text{present} + \text{partiallyAttended}) / \text{total}) \times 100$ , 2 dp
<code>data.overallStats.byDay[].status</code>	COMPLETED, ONGOING, or UPCOMING
<code>data.byTicketType</code>	Same stats broken down per ticket type

### Error Response JSON Sample:

```
{
  "success": false,
  "httpStatus": "NOT_FOUND",
  "message": "Event not found",
  "action_time": "2026-05-23T10:30:45",
  "data": "Event not found"
}
```

### Standard Error Types:

- `401 UNAUTHORIZED`: Missing or invalid token
- `403 FORBIDDEN`: Authenticated user is not the event organizer
- `404 NOT_FOUND`: Event does not exist

## 2. Get Attendance List

**Purpose:** Returns a paginated, filterable list of all tickets for an event. Each row is one ticket. Present and absent tickets are unified in one list — use the `status` filter to narrow down.

**Endpoint:** `GET` `/e-events/attendance/{eventId}/list`

**Access Level:** `🔒` Protected — Event Organizer Only

**Authentication:** Bearer Token

### Request Headers:

Header	Type	Required	Description
<code>Authorization</code>	string	Yes	Bearer <token>

### Path Parameters:

Parameter	Type	Required	Description	Validation
eventId	UUID	Yes	Event identifier	Must be a valid UUID of an existing event owned by the authenticated organizer

### Query Parameters:

Parameter	Type	Required	Description	Validation	Default
status	enum	No	Filter by attendance status	PRESENT, ABSENT, PARTIALLY_ATTENDED	All
dayNumber	integer	No	Filter for a specific event day	Must be between 1 and totalDays	None (overall)
ticketTypeId	UUID	No	Filter by ticket type	Must belong to this event	All types
buyerType	enum	No	Filter by how the ticket was purchased	SYSTEM_USER, AT_DOOR	All
search	string	No	Search by attendee name, email, or phone	Partial match, case-insensitive	None
page	integer	No	Page number (0-indexed)	Min: 0	0
size	integer	No	Items per page	Min: 1	20

### Success Response JSON Sample:

```
{
  "success": true,
  "httpStatus": "OK",
  "message": "Attendance list retrieved",
  "action_time": "2026-05-23T10:30:45",
  "data": {
    "eventId": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
    "eventTitle": "East African Tech Summit 2026",
    "appliedFilters": {
      "ticketTypeId": "3fa85f64-5717-4562-b3fc-2c963f66afa7",
      "ticketTypeName": "VIP Pass",
      "status": "ALL",
    }
  }
}
```

```
"dayNumber": 1,
"dayName": "Day 1 - Opening",
"buyerType": "ALL",
"search": null
},
"summary": {
  "totalTickets": 100,
  "present": 90,
  "absent": 10,
  "partiallyAttended": null,
  "attendanceRate": 90.0
},
"attendees": [
  {
    "ticketInstanceId": "3fa85f64-5717-4562-b3fc-2c963f66afa8",
    "formResponseId": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
    "attendeeNumber": "EVT-A3F4B21C-VIP-0042",
    "ticketTypeId": "3fa85f64-5717-4562-b3fc-2c963f66afa7",
    "ticketType": "VIP Pass",
    "ticketSeries": "VIP-0042",
    "attendeeName": "John Doe",
    "attendeeEmail": "john@example.com",
    "attendeePhone": "+255712345678",
    "buyer": {
      "buyerName": "Jane Doe",
      "buyerEmail": "jane@example.com",
      "buyerId": "3fa85f64-5717-4562-b3fc-2c963f66afa9",
      "buyerType": "SYSTEM_USER"
    },
    "bookingReference": "EVT-A3F4B21C",
    "pricePaid": 150.00,
    "attendanceStatus": "PRESENT",
    "checkInTime": "2026-12-15T09:15:00+03:00",
    "checkInLocation": "Main Gate",
    "checkedInBy": "Scanner Operator 1",
    "scannerId": "3fa85f64-5717-4562-b3fc-2c963f66afb0"
  },
  {
    "ticketInstanceId": "3fa85f64-5717-4562-b3fc-2c963f66afb1",
    "formResponseId": null,
```

```

    "attendeeNumber": "EVT-B5D2E12F-GENE-0101",
    "ticketType": "General Admission",
    "ticketSeries": "GENE-0101",
    "attendeeName": "Ali Hassan",
    "attendeeEmail": null,
    "attendeePhone": null,
    "buyer": {
      "buyerName": "Ali Hassan",
      "buyerEmail": null,
      "buyerId": null,
      "buyerType": "AT_D00R"
    },
    "bookingReference": "EVT-B5D2E12F",
    "pricePaid": 50.00,
    "attendanceStatus": "ABSENT",
    "checkInTime": null,
    "checkInLocation": null,
    "checkedInBy": null,
    "scannerId": null
  }
],
"pagination": {
  "currentPage": 0,
  "pageSize": 20,
  "totalPages": 5,
  "totalElements": 100,
  "hasNext": true,
  "hasPrevious": false
}
}
}

```

### Success Response Fields:

Field	Description
<code>data.appliedFilters</code>	Echo of all active filters so the client knows what was applied
<code>data.summary.totalTickets</code>	Total tickets matching the ticket type filter (unaffected by status/search)
<code>data.summary.present</code>	Tickets with <code>PRESENT</code> status under current filters

Field	Description
<code>data.summary.absent</code>	Tickets with <code>ABSENT</code> status under current filters
<code>data.summary.partiallyAttended</code>	<code>null</code> when <code>dayNumber</code> is active; count when viewing overall multi-day
<code>data.summary.attendanceRate</code>	$((\text{present} + \text{partiallyAttended}) / \text{total}) \times 100$ , 2 dp
<code>data.attendees[].ticketInstanceId</code>	Unique ticket instance UUID
<code>data.attendees[].formResponseId</code>	UUID of the buyer's applicant form response — <code>null</code> if the event had no form
<code>data.attendees[].attendeeNumber</code>	Unique readable ID = <code>bookingReference + "-" + ticketSeries</code>
<code>data.attendees[].attendanceStatus</code>	<code>PRESENT</code> , <code>ABSENT</code> , or <code>PARTIALLY_ATTENDED</code>
<code>data.attendees[].buyer.buyerType</code>	<code>SYSTEM_USER</code> (online) or <code>AT_DOOR</code> (walk-in)
<code>data.attendees[].buyer.buyerId</code>	Populated for <code>SYSTEM_USER</code> , <code>null</code> for <code>AT_DOOR</code>
<code>data.attendees[].checkInTime</code>	<code>null</code> when <code>attendanceStatus</code> is <code>ABSENT</code>
<code>data.pagination</code>	Standard pagination envelope

### Notes:

- When `dayNumber` is provided: `attendanceStatus` is `PRESENT` or `ABSENT` for that specific day only
- When `dayNumber` is omitted: `attendanceStatus` reflects the overall across all days (`PARTIALLY_ATTENDED` possible)
- `summary` reflects the current filter scope, not the whole event
- `partiallyAttended` in `summary` is `null` when a `dayNumber` filter is active

### Error Response JSON Sample:

```
{
  "success": false,
  "httpStatus": "NOT_FOUND",
  "message": "Invalid day 5. Valid: 1-3",
  "action_time": "2026-05-23T10:30:45",
  "data": "Invalid day 5. Valid: 1-3"
}
```

### Standard Error Types:

- `401 UNAUTHORIZED`: Missing or invalid token
- `403 FORBIDDEN`: Authenticated user is not the event organizer
- `404 NOT_FOUND`: Event not found / day number out of range / ticket type not found

# 3. Get Attendee Detail

**Purpose:** Returns the full check-in history for a single ticket across all event days. Shows per-day status including upcoming days.

**Endpoint:** `GET /e-events/attendance/{eventId}/attendees/{ticketInstanceId}`

**Access Level:**  Protected — Event Organizer Only

**Authentication:** Bearer Token

**Request Headers:**

Header	Type	Required	Description
Authorization	string	Yes	Bearer <token>

**Path Parameters:**

Parameter	Type	Required	Description	Validation
eventId	UUID	Yes	Event identifier	Must be owned by the authenticated organizer
ticketInstanceId	UUID	Yes	Individual ticket instance identifier	Must belong to a booking under this event

**Success Response JSON Sample:**

```
{
  "success": true,
  "httpStatus": "OK",
  "message": "Attendee detail retrieved",
  "action_time": "2026-05-23T10:30:45",
  "data": {
    "ticketInstanceId": "3fa85f64-5717-4562-b3fc-2c963f66afa8",
    "formResponseId": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
    "attendeeNumber": "EVT-A3F4B21C-VIP-0042",
    "attendeeName": "John Doe",
    "attendeeEmail": "john@example.com",
    "attendeePhone": "+255712345678",
    "ticketType": "VIP Pass",
```

```
"ticketSeries": "VIP-0042",
"bookingReference": "EVT-A3F4B21C",
"pricePaid": 150.00,
"overallStatus": "PARTIALLY_ATTENDED",
"daysAttended": 2,
"daysTotal": 3,
"checkInsByDay": [
  {
    "dayNumber": 1,
    "dayName": "Day 1 - Opening",
    "dayDate": "2026-12-15",
    "status": "CHECKED_IN",
    "checkInTime": "2026-12-15T09:15:00+03:00",
    "checkInLocation": "Main Gate",
    "checkedInBy": "Scanner Operator 1",
    "scannerId": "3fa85f64-5717-4562-b3fc-2c963f66afb0"
  },
  {
    "dayNumber": 2,
    "dayName": "Day 2 - Conference",
    "dayDate": "2026-12-16",
    "status": "NOT_CHECKED_IN",
    "checkInTime": null,
    "checkInLocation": null,
    "checkedInBy": null,
    "scannerId": null
  },
  {
    "dayNumber": 3,
    "dayName": "Day 3 - Closing",
    "dayDate": "2026-12-17",
    "status": "UPCOMING",
    "checkInTime": null,
    "checkInLocation": null,
    "checkedInBy": null,
    "scannerId": null
  }
]
}
```

## Success Response Fields:

Field	Description
<code>data.ticketInstanceId</code>	Unique ticket instance UUID
<code>data.formResponseId</code>	UUID of the buyer's applicant form response — <code>null</code> if the event had no form
<code>data.attendeeNumber</code>	Unique readable ID = <code>bookingReference + "-" + ticketSeries</code>
<code>data.overallStatus</code>	<code>PRESENT</code> (all days), <code>PARTIALLY_ATTENDED</code> (some), <code>ABSENT</code> (none)
<code>data.daysAttended</code>	Count of days this ticket was checked in
<code>data.daysTotal</code>	Total number of event days
<code>data.checkInsByDay[].status</code>	<code>CHECKED_IN</code> , <code>NOT_CHECKED_IN</code> , or <code>UPCOMING</code>
<code>data.checkInsByDay[].checkInTime</code>	<code>null</code> when not checked in or day is upcoming
<code>data.checkInsByDay[].checkedInBy</code>	Name of the scanner operator who processed the check-in

## Error Response JSON Sample:

```
{
  "success": false,
  "httpStatus": "NOT_FOUND",
  "message": "Ticket not found",
  "action_time": "2026-05-23T10:30:45",
  "data": "Ticket not found"
}
```

## Standard Error Types:

- `401 UNAUTHORIZED`: Missing or invalid token
- `403 FORBIDDEN`: Authenticated user is not the event organizer
- `404 NOT_FOUND`: Event not found or ticket instance does not belong to this event

# Quick Reference Guide

## Attendance Status Values

Value	Meaning	Available when
-------	---------	----------------

PRESENT	Checked in (all days if no day filter)	Always
ABSENT	Never checked in	Always
PARTIALLY_ATTENDED	Checked in some days but not all	No <code>dayNumber</code> filter, multi-day events only

## Day Status Values

Value	Meaning
CHECKED_IN	Ticket was scanned on that day
NOT_CHECKED_IN	Day passed, ticket not scanned
UPCOMING	Day has not started yet

## Buyer Type Values

Value	<code>buyerEmail</code>	<code>buyerId</code>	Meaning
SYSTEM_USER	Populated	Populated	Bought online, has platform account
AT_DOOR	null	null	Walk-in sale, name only captured at door

## Common HTTP Status Codes

- `200 OK`: Successful request
- `401 UNAUTHORIZED`: Authentication required or token invalid
- `403 FORBIDDEN`: Not the event organizer
- `404 NOT_FOUND`: Event, ticket, or day number not found
- `500 INTERNAL_SERVER_ERROR`: Unexpected server error

Revision #8

Created 11 December 2025 10:23:18 by Admin Qbit

Updated 24 May 2026 07:30:50 by Admin Qbit