

Event Group Tickets, Product Group Buys — Frontend Change Log

Author: Josh S. Sakweli, Backend Lead Team **Last Updated:** 2026-10-31 **Version:** v1.0 **Backend version / branch:** `fet/location_service` (+ notification server `d17b028`) **Deployed to:** staging, already live since the 2026-10-25 deploy (`6434a415`) **Compared with:** staging as deployed on 2026-10-06 (`448f06a0`), before group tickets existed

Base URL: `https://api.nexgate.co/api/v1` (staging: `https://dev.api.nexgate.co/api/v1`)

Short Description: Event tickets can now be bought **together at a group price** ("3 tickets at 15,000 instead of 20,000"). Ticket responses and the event checkout gained fields for it. Product group buys changed too: moving and leaving a group now take a **whole payment**, there is a new **Leave group** action, and filled groups now really create their orders (before, they never did).

Audience: mobile team · web team

Related docs — the full flows, with every new endpoint, live in their own docs (not repeated here):

- `event_group_tickets_api_doc.md` — event tickets "buy together"
- `product_group_purchase_api_doc.md` — product group buys
- `docs/changelog/frontend_changelog_2026-10-25.md` §7.1 — group checkout with pickup / delivery

How to read this

Badge	Means	App must act?
NEW	a new endpoint, field or notification	only to use the new feature
CHANGED	an existing endpoint gained fields or behaves differently; old calls still work	should update
FIX	a bug fixed; same shape, now correct values	check screens that relied on the wrong value
BREAKING	an old call can now be refused or means something different	must update

One change is breaking: moving a product group payment (§2.2). Everything else keeps working.

0. At a glance

#	Service	New	Changed	Breaking	App work
1	Event tickets	7	3	—	"Buy together" for organisers and buyers; waiting state with no QR until the group is full
2	Product group buys	1	1	1	pick a payment, Move sends <code>checkoutSessionId</code> , new Leave button
3	Notifications	4	—	—	4 new types, 1 new target

1. Event tickets

What changed in this service: an organiser can switch on group buying for a **paid** ticket type. One buyer starts a group, friends join with a code, and when the group is full everyone gets their tickets at once. If it does not fill in time, everyone is refunded automatically.

Example: VIP is 20,000. The organiser sets group price 15,000, size 3.

- Asha starts a group "Yoyo Crew" with 2 tickets → pays 30,000 → **no booking yet**, she sees "2 of 3".
- Baraka joins with the code with 1 ticket → pays 15,000 → group is full → Asha gets 2 tickets, Baraka 1, with QR codes.
- ☐ If nobody joins before the deadline → Asha gets her 30,000 back.

Rules the app must show:

- Size counts **tickets, not people**.
- **No ticket and no QR code until the group is full.**
- Paid tickets only (not free, donation or door sales).
- A group closes at the earliest of: its time limit, the end of ticket sales, or **2 hours before the event**. `expiresAt` is the deadline.
- Each friend buys with their own account: a group checkout cannot buy for `otherAttendees`.

Change	Method	Endpoint	One line	App
NEW	PATCH	/e-events/tickets/{ticketId}/group-buying	organiser switches group buying on / off	to use
CHANGED	GET	ticket details, ticket list, event tickets	groupBuying object	should update
CHANGED	POST	/e-events/checkout	optional group in, group out	to use
CHANGED	POST	/e-events/checkout/{sessionId}/payment	can fill a group; new "just filled" error	should update
CHANGED	PATCH	/e-events/booking-orders/{bookingId}/cancel	also works for bookings from a group	no change
NEW	GET	/e-events/ticket-groups/ticket/{ticketTypeId}/open	open groups for a ticket	to use
NEW	GET	/e-events/ticket-groups/code/{groupCode}	look up a shared code (TG- + 6)	to use
NEW	GET	/e-events/ticket-groups/{groupId}	one group	to use
NEW	GET	/e-events/ticket-groups/my	my groups	to use
NEW	POST	/e-events/ticket-groups/{groupId}/transfer	move my tickets to another group	to use
NEW	POST	/e-events/ticket-groups/{groupId}/withdraw	leave with my tickets, full refund	to use

1.1 NEW Organiser: switch group buying on

Endpoint: PATCH {base_url}/e-events/tickets/{ticketId}/group-buying

```
{ "enabled": true, "groupPrice": 15000, "groupSize": 3, "timeLimitHours": 24 }
```

Field	Rule
enabled	required; false switches it off, groups already running still finish
groupPrice	required to switch on; above 0 and below the ticket price
groupSize	required to switch on; 2 up to the ticket's total quantity

Field	Rule
timeLimitHours	required to switch on; 1 to 720

Changing price or size later does not touch groups already running. Errors are 400, show the message as it is (for example "Group price must be lower than the ticket price (20000.00)").

1.2 CHANGED Ticket responses — groupBuying

Endpoint: GET {base_url}/e-events/tickets/{eventId}/{ticketId} · the event's ticket list · the PATCH in §1.1

Field	Before	After
groupBuying	—	added; missing when group buying is off and no group is running

```
"groupBuying": { "enabled": true, "groupPrice": 15000.0, "groupSize": 3, "timeLimitHours": 24,
                  "availableToBuy": 14, "heldInOpenGroups": 3, "paidInOpenGroups": 2,
                  "openGroups": 1, "sold": 0 }
```

Field	Meaning
availableToBuy	tickets anyone can still buy now
heldInOpenGroups	kept back for groups still filling (a whole group's worth from the moment it starts)
paidInOpenGroups	how many of those are paid
openGroups	groups running now
sold	real bookings, including from filled groups

App to do

- ☐ Organiser ticket screen: "Buy together" switch with price / size / time, and the counts above.
- ☐ Ticket page: show "Buy together at 15,000" when groupBuying.enabled.

1.3 CHANGED Event checkout — group

Endpoint: POST {base_url}/e-events/checkout

Request

Field	Before	After
group	—	added , optional: { "groupName": "Yoyo Crew" } starts a group, { "groupId": "..." } joins one

```
{ "eventId": "1635...", "ticketTypeId": "8a31...", "ticketsForMe": 1, "group": { "groupId": "3a6d..." } }
```

Response

Field	Before	After
price / totals	ticket price	the group price when group was sent
group	—	added : { "groupName": "Yoyo Crew", "startsNewGroup": true } or { "groupId": "...", "startsNewGroup": false }; missing on a normal checkout

Errors (400, show as is)

- Group name is required when starting a group
- An open group named 'yoyo crew' already exists for this ticket (names are unique per ticket, case ignored)
- Only 1 tickets left in this group · This group is no longer open · This group is for a different ticket
- Group buying is not available for this ticket · It is too close to the event to start a group
- Group tickets are bought for yourself; each friend joins the group with their own account
- You can hold at most 6 of this ticket; 2 are already waiting in groups — the per-person limit counts waiting tickets

1.4 CHANGED Pay — may fill the group

Endpoint: POST {base_url}/e-events/checkout/{sessionId}/payment — same request

After paying a group checkout	What the user has
-------------------------------	-------------------

started a group	group <code>OPEN</code> , no booking
joined, group not full	in the group, no booking
this payment filled the group	every member gets bookings and QR codes at once, in My bookings

New error to handle: `This group just filled up or closed. You have not been charged.` — two people paid for the last place at the same moment and the other got it. Nothing was taken; send the user back to the group list.

1.5 CHANGED Cancel a booking

Endpoint: PATCH `{base_url}/e-events/booking-orders/{bookingId}/cancel` — unchanged

Also works for a booking that came from a group: it refunds exactly those tickets and puts them back on sale.

1.6 NEW Ticket groups — read, move, leave

Full shapes and errors: `event_group_tickets_api_doc.md` Parts 2 and 4. The parts the screens need:

Field on a group	Meaning
<code>status</code>	<code>OPEN</code> , <code>COMPLETED</code> (full, tickets issued), <code>FAILED</code> (ended, refunded)
<code>seatsOccupied</code> / <code>totalSeats</code>	the "2 of 3"
<code>endReason</code>	when ended: <code>FILLED</code> , <code>EXPIRED</code> , <code>EMPTIED</code> , <code>EVENT_CANCELLED</code> , <code>TICKET_DELETED</code>
<code>myTickets</code>	the viewer's tickets in it; missing when none
<code>participants</code>	who is in, with how many <code>tickets</code> each — no money shown; <code>isMe: true</code> on the viewer; empty once <code>FAILED</code>

- **Move** `{ "targetGroupId": "...", "quantity": 1 }` — any number of my tickets, same event and same ticket type, no money moves. If it fills the target, everyone there gets tickets at once.
- **Leave** `{ "quantity": 1 }` — any number of my tickets, refunded in full at once; the places open up again.

App to do

- ☐ Ticket page: open groups list, "Join with a code", "Start a group".
- ☐ My groups: "Waiting for group: 2 of 3" with `expiresAt` — **never a ticket or QR** while `OPEN`.
- ☐ Group screen: participants (highlight `isMe`), Move and Leave.

2. Product group buys

What changed in this service: moving and leaving now take **one whole payment** — its items, delivery and tax stay together. Each payment a user makes in a group becomes **its own order**.

Example: Asha pays twice in a group: payment A (1 item, 900) and payment B (1 item, 900).

- ☐ Move payment A to another group → A's item and money go there, B stays.
- ☐ Leave with payment B → 900 (plus its delivery) back at once.
- ☐ Before: "move 1 item" did not say which payment, so the wrong money could be refunded, and a full move left Asha with no order.

Change	Method	Endpoint	One line	App
CHANGED	GET	<code>/group-purchases/{groupId}, /code/{groupCode}, /my-participations</code>	<code>purchaseHistory</code> = the payments to pick from	should update
BREAKING	POST	<code>/group-purchases/transfer</code>	<code>checkoutSessionId</code> instead of <code>quantity</code>	must update
NEW	POST	<code>/group-purchases/{groupId}/withdraw</code>	leave with one payment, full refund	to use
FIX	—	group fills / expires	orders created, every payment refunded	check screens

2.1 CHANGED Group reads — `purchaseHistory`

Endpoint: `GET` `{base_url}/e-commerce/group-purchases/{groupId} · /code/{groupCode} · /my-participations`

Field	Before	After
-------	--------	-------

<code>purchaseHistory</code>	only the first payment for some members	every payment (one entry = one payment); only on the viewer's own entry — missing on others and when logged out
<code>purchaseCount</code>		how many payments

```
"purchaseHistory": [
  { "checkoutSessionId": "dd9b...", "quantity": 1, "amountPaid": 900.00, "purchasedAt": "2026-10-07T12:40:11" },
  { "checkoutSessionId": "28bc...", "quantity": 1, "amountPaid": 900.00, "purchasedAt": "2026-10-07T12:41:03" }
]
```

Missing `purchaseHistory` means "not mine", never an error.

2.2 BREAKING Move a payment to another group

Endpoint: POST `{base_url}/e-commerce/group-purchases/transfer`

Request

Field	Before	After
<code>quantity</code>	how many items to move	ignored (an old body is not refused)
<code>checkoutSessionId</code>	—	added: which payment moves; may be left out when the user paid only once

Before	After
<pre>{ "sourceGroupId": "34e5...", "targetGroupId": "5604...", "quantity": 1 }</pre>	<pre>{ "sourceGroupId": "34e5...", "targetGroupId": "5604...", "checkoutSessionId": "dd9b..." }</pre>

Why it is breaking: an old app that sends `quantity: 1` for a payment of 2 items now moves **both**, and a user who paid twice is refused until the app sends `checkoutSessionId`. Only between groups of the same product and shop at the same group price. To move only part of a payment: leave with it (\$2.3) and buy again in the other group.

Errors (400)

- You paid 2 times in this group; choose which payment with checkoutSessionId
- That payment is not in this group · The target group does not have 3 free seats
- Payments can only move out of a group that is still open · Source and target groups must be different

2.3 NEW Leave a group — full refund

Endpoint: POST {base_url}/e-commerce/group-purchases/{groupId}/withdraw

```
{ "checkoutSessionId": "28bc..." }
```

Body may be empty when the user paid once. Only while the group is OPEN. The whole payment (items, delivery, tax) goes back to the wallet at once. The response is the user's entry: status: REFUNDED when nothing is left, still ACTIVE when another payment remains. A user who left can join again.

Errors (400): You can only leave a group that is still open, You have no active payment in this group.

2.4 FIX Filling, expiring, racing

What	Before	After
a group fills	no orders were created — buyers paid and got nothing	one order per payment, within seconds; a job every 15 min creates any that were missed
a buyer paid twice	one order, one refund	an order / refund for each payment
a group expires	refunds failed; checked only at midnight	every payment refunded; checked every 15 min
two people pay for the last place	both could get in	one gets it; the other sees Checkout failed during post-payment processing. Please contact support. and is not charged
everyone moved out	stock stayed held	stock back on sale, group DELETED
"You saved" on the order	today's product price	the price the group started with

App to do

- ☐ My groups: list the payments from `purchaseHistory` ("1 item · 900"); skip the choice when there is one.
- ☐ Move: send the chosen `checkoutSessionId`, stop sending `quantity`.
- ☐ New **Leave group** button while `OPEN`.
- ☐ The last-place error: send the user back to the group list.

3. Notifications

All four open the group: `targetType: "EVENT_TICKET_GROUP"`, `targetId` = group id → `GET /e-events/ticket-groups/{targetId}`. An app that does not know the target yet opens the bell list.

Change	Type	To	When	App
NEW	<code>EVENT_TICKET_GROUP_CREATED</code>	the buyer who started it	group opened — prompt to share the code	
NEW	<code>EVENT_TICKET_GROUP_JOINED</code>	everyone in the group	someone joined or moved tickets in	refresh "2 of 3"
NEW	<code>EVENT_TICKET_GROUP_COMPLETED</code>	everyone in the group	full — tickets are in My bookings	refresh bookings
NEW	<code>EVENT_TICKET_GROUP_FAILED</code>	everyone still in it	ended without filling — refunded	

A refund also sends the existing `ESCROW_REFUNDED`, once per person (not once per ticket).

4. App checklist by screen

Screen	What to change	Sections
Ticket settings (organiser)	"Buy together" switch, price / size / time, <code>groupBuying</code> counts	§1.1, §1.2
Ticket page / event checkout	buy-together price, open groups, join by code, send <code>group</code> ; "just filled" error	§1.2-§1.4
My ticket groups	waiting "2 of 3" with no QR; participants; Move / Leave	§1.6
Product group — my groups	pick a payment; Move with <code>checkoutSessionId</code> ; Leave button	§2.1-§2.3
Bell list / push	4 new types, target <code>EVENT_TICKET_GROUP</code>	§3

Revision #1

Created 31 October 2026 04:58:36 by Admin Qbit

Updated 31 October 2026 05:13:27 by Admin Qbit