Events

Group sessions, workshops, webinars, info nights, and any public event that takes registrations. Free or paid (Stripe Connect), online or in-person or both, single attendee or multi-attendee per registration, QR-coded ticket emails, the public check-in page, 24-hour reminders, the discount-code feature dedicated to Events, edit + unpublish + cancel + archive lifecycle, and how Events differ from Bookings and Q&A.

Up to date as of v1.29.0

CONTENTS

Overview

Events is the module for one-off group sessions: a webinar on the new IRCC processing rules, a workshop for prospective Express Entry applicants, an info session for parents considering sponsoring a child, a CPD-style session for fellow consultants. Each Event has a single date and time, a single public registration page (separate from your individual booking page at /book/<slug>), an optional ticket price collected through Stripe Connect, a capacity cap, an optional cover image, and a list of attendees who register and receive QR-coded ticket emails. The module is open to every tenant on every tier (Basic and Premium); the Stripe Connect requirement is the same as for paid Bookings.

Where the Events page lives

Sidebar -> Events. The page title reads 'Events' with the subtitle 'Publish one-off webinars, workshops, or info sessions. Clients register through a public event page, free or paid.' A maroon '+ New Event' button sits in the top-right. Below the header, a two-tab strip switches between Active (the default; every event whose archived flag is off) and Archived (every event you have archived). Below the tabs, the events list itself: one card per event with title, status pill (Free / Paid), date and time, registered count, Public page link (deep-links to the public registration URL), Attendees link (jumps into the attendee list), Edit button, Unpublish button (when published) or Publish button (when draft), and Archive button.

- Public registration page at /event/<event-slug>: the URL you share with prospects via your website, social media, email blasts, or partner referrals.
- Multi-attendee per registration: a single registrant can sign up themselves PLUS 1 to 9 additional people in the same transaction (a family of four buys four tickets together).
- QR-coded ticket emails: each attendee gets their own ticket with a unique QR code that scans at the door for check-in.
-

Three meeting modes: Online (Google Meet auto-attached when you have Google Calendar connected; Teams auto-attached for Microsoft 365 with Teams Business), In-person (you provide an address), Both (the registrant picks at registration).

- Capacity: a per-event maximum total attendee count, enforced live. When the cap is hit, the public page shows 'Sold out' and the registration form refuses new registrations.
- Dedicated discount codes: separate from the Bookings discount codes; live under Events sidebar -> Discounts. Promo codes prospects type on the registration form to get a percentage or fixed-amount discount on this event (or every event).
- 24-hour reminder cron: a daily 12:00 UTC sweep emails every confirmed attendee a reminder for any event happening in the next 24 hours.

> **Important**: Two prerequisites apply to paid Events: Stripe Connect must be connected (see the Bookings chapter for the walkthrough) and at least one calendar must be connected if you want auto-Google-Meet / Teams links on online events. Free events with no calendar still work; the public page accepts registrations and the platform sends ticket emails, but no calendar event is created and no Meet / Teams link is auto-attached.

Events vs Bookings vs Q&A: when to pick which

The platform has three modules that look superficially similar: Bookings (the booking-engine for one-to-one meetings), Events (this module, for public group sessions), and Q&A (a shared-session-per-date Q&A block where multiple prospects register for the same date and the first registrant sets the start time). Pick the right one based on the SHAPE of the engagement, not on the price.

Pick Bookings when

-

The meeting is one-on-one between you and a single prospect (or a single prospect + their family on a multi-attendee booking).

- The prospect picks their own time from your availability; you do not pre-publish a fixed date.
- Each meeting is its own engagement with its own potential follow-up (a Service Agreement, a Transfer Room, etc.).

Pick Events when

- The session is one-to-many: you present or facilitate, multiple unrelated prospects attend.
- The date and time are FIXED in advance; you publish a specific session and prospects sign up for it.
- Each attendee gets their own ticket (with QR code) and the check-in is a moment-of-event verification.
- Capacity is a meaningful limit (you have a physical venue with 30 seats, or you want to cap your Zoom webinar attendees).

Pick Q&A when

- You want to publish a service (e.g., 'Express Entry Q&A') that prospects register against, and the FIRST registrant on a given date sets the start time for that date (subsequent registrants for the same date join the same session).
- You want this format to be repeatable: every date that anyone registers for becomes a Q&A session; you do not have to manually publish each session.
- The natural pattern is 'multiple prospects join the same session and ask questions live' rather than 'you present a fixed agenda to a fixed audience' (the latter is Events).

> **Note**: When in doubt between Events and Q&A: 'I want to control which dates exist as sessions' is Events. 'I want every date a prospect picks to become a session' is Q&A. The two modules can co-exist; many tenants

> use Events for once-a-quarter scheduled webinars AND Q&A for the ongoing 'anyone can ask anytime' service.

Creating an Event: the form overview

Click '+ New Event' on the Events page. A side panel opens with the multi-section event form. The form is divided into logical groups: Basics (title, description, language), When (date and time, duration, timezone), Meeting (mode, location or video link, capacity), Image (cover image cropper), Pricing (price + Stripe), and Notifications (anonymous tickets toggle, tenant notification toggle). At the bottom: 'Save as draft' (the event is created but not yet visible on the public page; capable of further editing without affecting prospects) and 'Publish' (the event goes live immediately on the public page and is visible at its registration URL). The next several sections walk through each field.

Editing later

Editing a published event from the event card's Edit button opens the same side panel pre-filled with the current values. Most fields are editable post-publish; some have safety guards (changing the date or time on an event with confirmed registrations prompts a confirmation modal because the change emails every attendee with the new time). The calendar event on your calendar is updated in lockstep with date/time/location changes so attendees on the calendar invite see the new time too.

Event fields: title, description, language

Title (required)

The headline that appears on the public registration page, on the event card in your dashboard, on the calendar event title, on the ticket emails, and on the 24-hour reminder. Keep it concise + descriptive: 'Parya Q&A Session 2026', 'Express Entry Workshop for Indian Applicants', 'PFL Response Strategy Webinar', 'Sponsorship

Information Session (in Mandarin)'. The title shows on every prospect-facing surface; pick one that prospects will recognize when they see the reminder email two days after they registered.

Description (required)

A multi-paragraph free-text description that renders on the public registration page. This is where you sell the event: who is it for, what will attendees learn, what is the agenda, what should they bring. Treat it like a marketing landing page. Markdown is not supported; you get plain text with line breaks. Aim for 2 to 5 paragraphs (too short and prospects bounce; too long and they skip).

Primary language (required)

Which language the event will be conducted in. Defaults to English. The picker lists English and Quebec French; on Premium tenants with Phase 10C runtime translation, the picker offers the eight additional languages (Spanish, Mandarin, Tagalog, Punjabi, Arabic, Hindi, Urdu, Persian / Farsi). The public registration page chrome flips to the selected language; the description text stays in whatever language you typed (the platform does not translate your description text). Use this field to set the language a prospect should expect at the actual event; a 'Sponsorship Information Session in Mandarin' set to Mandarin tells the prospect they will hear Mandarin during the event.

Event fields: when (date, time, duration)

Date (required)

A date picker. Pick the calendar date the event will happen on. Future dates only on a NEW event (the validator refuses past dates on creation; on edit, past dates are accepted but you would only do that for a back-dated administrative record of an event you already held). The picker honors your tenant timezone; the date you pick is interpreted in YOUR tenant timezone (so a Toronto tenant picking June 12 means June 12 Toronto time).

Start time (required)

A time picker accepting any 15-minute increment from 00:00 to 23:45 in your tenant timezone. Pick the wall-clock start time of the event. Combined with the date, the platform stores the start_time as a timestamptz computed via toUtcMs (the same wall-clock-to-UTC helper used in Bookings) so daylight saving transitions are handled correctly.

Duration (required)

How long the event runs, in minutes. Typical values: 30, 60, 90, 120. Used to compute the end_time (start_time + duration), which renders on the public page ('Saturday, June 12, 2026 from 6:00 PM to 7:30 PM Toronto time') and on the calendar event you create. The freebusy guard on calendar conflict checks (see below) uses this duration to know how long to mark you busy.

Timezone display

The public registration page renders the event time in TWO timezones: your tenant timezone (the canonical one, since the event happens at YOUR location or YOUR Zoom room) AND the prospect's browser-detected timezone if it differs. A prospect in Vancouver registering for a Toronto event sees both '6:00 PM Toronto time' and '3:00 PM your time'. The ticket email + the 24-hour reminder email use the same dual-timezone format. Calendar invites use the canonical timezone with each attendee's calendar app converting to local on display.

Event fields: meeting mode and location

An event runs in one of three modes: Online (video conference), In-person (a physical venue), or Both (the registrant picks at registration). The mode picker is a radio group on the event form.

Online

The event runs over video conference. When the event is created, the platform calls your connected calendar's API to create a calendar event with conferenceData.createRequest (Google Calendar) or isOnlineMeeting=true (Microsoft 365 with Teams Business). The auto-attached Google Meet or Teams link lands on the event's meeting_url field and is included in every ticket email + every 24-hour reminder. Each attendee can click the link directly from their inbox to join. If your calendar provider returns the create response WITHOUT the hangoutLink populated (Google does this occasionally on Workspace domains where conferences provision asynchronously), the platform defends with a sleep + events.get re-fetch to pull the link a moment later; the link almost always lands within 1.5 seconds.

What if you want to bring your own video link instead of the auto-attached Meet / Teams? The Online mode also offers a 'Bring your own video link' radio sub-option: tick it and a text field appears where you paste a custom URL (a Zoom link, a Webex link, a custom URL). The platform then uses your provided URL as the meeting_url and skips the conferenceData.createRequest step. The custom URL is what shows up on tickets, reminders, and the calendar event location field.

In-person

The event happens at a physical address. Picking In-person reveals a Location address field where you type the venue (e.g., '123 Bay Street, Toronto, ON M5J 2N9, Conference Room B'). The address appears on the public registration page, on every ticket email (so attendees can pull up directions on the way), and on the calendar event's location field. No video link is created. Online attendance is not offered to registrants.

Both

The event runs as a HYBRID: you have a physical venue AND a video link, and each registrant picks which side they will attend on at registration time. Picking Both reveals BOTH the Location address field AND the video sub-options (auto-attach Meet/Teams or bring your own). The public registration form then carries a radio 'How will you attend?' (In-person at <address> / Online via video link); the registrant's choice is stored on their event_registration row and the ticket email mentions only the side they picked. Useful for events where the venue has a hard physical cap but you want unlimited online attendance.

Event fields: cover image (3:1 cropper)

Every event can carry a cover image that renders large at the top of the public registration page and as a small thumbnail on the dashboard card. The image is OPTIONAL (events without an image just show a maroon-bordered placeholder), but a thoughtful cover image dramatically improves registration conversion. The form's Image section carries a drag-and-drop file picker that accepts PNG, JPG, and WebP up to 5 MB.

The 3:1 cropper

When you drop an image, an inline cropper opens. The crop area is locked to a 3:1 aspect ratio (three units wide for every one unit tall). Drag the crop window or pinch to zoom on touch screens to frame your image; the platform crops to your selection and exports a 1500 x 500 pixel JPEG at quality 85%, typically ~150 to 300 KB. The 3:1 ratio matches the standard banner shape used across the platform (it is the same ratio as social-media cover photos on Facebook, LinkedIn, Twitter). Output is JPEG always (even when you uploaded a PNG with transparency); the platform flattens transparency against white. Use this knowing transparent logos will get a white background.

Storage

The cropped image lands in the event-images Supabase Storage bucket (public read, 5 MB cap, allowlist PNG / JPG / WebP). The public URL is stored on events.image_url and used directly in ticket emails + on the public registration page; no per-request decryption (the image is public by design, so it can render anywhere a URL appears). To swap a cover image on an existing event, drop a new one in the cropper; the platform overwrites the previous file and cache-busts the URL with a query-string version stamp so attendees who already received their ticket with the old image see the new image when they re-open the email.

> **Note**: For best results, upload an image that is ALREADY close to 3:1 (a banner you designed in Canva, a photo you cropped intentionally). Uploading a tall portrait photo and cropping to 3:1 will lose most of the photo's content; consider re-shooting or finding a better source image. The cropper does NOT upscale; if you drop a small image (under 1500 pixels wide), the export will be the source resolution, which may look blurry on retina displays.

Event fields: capacity

Every event has a Capacity field: the maximum total attendee count across all registrations. The default is empty (which the platform treats as 'unlimited'; useful for purely-online events with no physical seat constraint). For events with a real cap (a venue with 30 seats, a Zoom meeting with a 100-attendee license tier, an exclusive workshop deliberately limited to 12), type the number. The platform enforces the cap at registration time.

How capacity is computed

Capacity is checked as SUM(event_registrations.attendee_count) across every non-cancelled registration. This is critical for multi-attendee registrations: a family of four buying four tickets at once consumes 4 capacity slots, not 1. If your event has capacity 20 and three families have registered with 4 attendees each (12 total), there are 8 slots left and the next registration form caps the 'How many attendees?' input at 8. A registration that includes attendee_count larger than the remaining slots is rejected by the server-side validator with a clean error.

Sold out

When the running total hits the capacity, the public registration page replaces the registration form with a 'Sold out' message and the event card on your dashboard shows a charcoal 'Sold out' badge next to the registered count. The event stays Active (it has not ended yet; the calendar event is still on your calendar; existing attendees still expect to show up). If a registered attendee cancels later (freeing slots), the page automatically reopens to new registrations until the cap is hit again. Capacity is dynamic; you can edit it up or down at any time.

> **Important**: Lowering capacity BELOW the current registered count does NOT cancel anyone. The platform refuses the lower value with a 'cannot reduce capacity below current registrations; cancel attendees first or pick a higher number' error. Use this on purpose: if you discover at the last minute that your venue can only fit 25 instead of 30, you need to decide who to disinvite (typically the most-recently-registered five attendees), cancel them via the Attendees list (which refunds them automatically for paid events), then lower the capacity.

Event fields: pricing and Stripe Connect

An event is either Free or Paid. The Pricing section of the form carries a single price field; type 0 for Free, type a positive number for Paid. The platform stores cents internally (the same Rule 41 money-input pattern as Bookings), so '$50' types as 50 and renders as 'CA$50.00' on the public registration page. The currency is your tenant's configured currency (CAD by default). Per-event currency override is not supported in v1.

Per-registration vs per-attendee

The price is PER-ATTENDEE, not per-registration. A multi-attendee registration multiplies: a family of four registering for a $25 event pays $100 ($25 x 4). The form makes this explicit ('CA$25.00 per attendee'). The public registration page also shows the multiplication explicitly: as the prospect types the attendee count, the displayed total updates live ('4 attendees x CA$25.00 = CA$100.00').

Stripe Connect requirement

Paid events require Stripe to be connected (Settings -> Payments). If Stripe is not connected, the form refuses to save a paid event with an inline 'Connect Stripe first' notice + a deep link to the Payments settings page. Existing paid events on a tenant who later disconnects Stripe (rare) stay published but the public registration page returns a 'this event is temporarily unavailable for online payment' message; prospects cannot register until Stripe is reconnected. The platform fee mechanics are identical to Bookings: Basic tier pays 1% on the first CA$5,000 of monthly Stripe-processed volume + 0% above the cap; Premium pays 0% always.

Changing price after registrations exist

On an event with at least one registration, changing the price is restricted. You CAN lower the price (an existing-attendee-fairness consideration: lowering means past attendees overpaid, which prompts a goodwill refund-difference path you handle manually outside the platform if you wish). You CANNOT raise the price (past attendees would have paid less for the same event than future attendees, which damages trust). The form refuses a raise with an error explaining the constraint and suggesting you create a NEW event at the new price tier and let existing-event registrations finish at the old price.

Event fields: anonymous tickets and tenant notifications

Anonymous tickets toggle

An optional toggle in the Notifications section of the event form. When OFF (the default), each attendee on a multi-attendee registration receives their OWN ticket email at the email address they provided (per-attendee mode). When ON, only ONE bundled ticket email goes to the registrant (the primary contact), and the email contains a single ticket valid for the full attendee_count. Use anonymous tickets when: (a) the registrant is paying on behalf of friends or family and the other attendees do not necessarily have their own email, (b) the event is a single-entry door check where one ticket per registration is operationally simpler, (c) the registrants would prefer not to share their guests' email addresses with the platform.

The platform also auto-detects bundled mode when EVERY attendee email on a multi-attendee registration matches the registrant email (a registrant types their own email for every attendee row): the platform treats those as the same person and sends a single bundled email regardless of the toggle. The decision lives in decideEmailMode (lib/event-tickets.ts) and both the free-path register route and the Stripe webhook's paid-path handler use the same function.

Tenant notification toggle

A second optional toggle in the Notifications section. When ON, the platform emails YOU (the tenant) every time a new registration lands on the event. The notification email carries the registrant's contact details, the attendee count, the payment status (Confirmed / Paid / Pending), and a deep link to the event's Attendees list. When OFF (the default), no per-registration emails go to you; you check the Attendees list on the dashboard when you want to see who has registered. Useful to turn ON for low-volume premium events where you want real-time signal, and OFF for high-volume webinars where 200 notification emails would drown your inbox.

Calendar conflict check on save

When you click Save / Publish on an Online or Both event (any event that will result in a calendar event being created on your calendar), the platform runs a calendar conflict check BEFORE actually creating the event row. The platform queries your connected calendar's freebusy API for the event's date + time + duration window; if your calendar shows you as busy in that window (an existing meeting, an external event you accepted, another Event you already published), the save returns a 409 with a structured response listing the conflicting events.

The conflict modal

The form's save handler catches the 409 and opens a confirmation modal listing the conflicts. Each conflict row shows the conflicting event's title, the conflict's start + end times, the conflict's overlap percentage (50% means the conflicting event spans half of your new event's window; 100% means total overlap), and which side it sits on (Before, During, After, Spans). At the bottom: two buttons. 'Change time' returns you to the form to pick a different time. 'Proceed anyway' re-fires the save with a bypass_calendar_check=true flag; the platform creates the event despite the conflict, your calendar will have overlapping events, and you accept the consequences (you may end up double-booked at the meeting time).

When the check is skipped

- In-person-only events do NOT run the check. They do not create a calendar event either way; the venue is yours independently of your calendar state.
- Tenants without a connected calendar skip the check (there is no calendar to query). The platform creates the event without checking; if you have external commitments at the same time, you handle the conflict manually.
- Edits to an existing event that do NOT change the time (a description tweak, a capacity bump, a price change) skip the check; only date+time edits trigger it.

Publishing and unpublishing

An event has a publishing state independent of its lifecycle (active vs archived) and its sold-out state (capacity reached vs slots remaining). The publishing state controls whether the public registration page is reachable.

Draft (Save as draft)

Clicking 'Save as draft' creates the event in the database at is_published=false. The event card appears on your dashboard with an amber Draft badge but the public registration page returns a 404 (the URL exists in the slug space but resolves only when is_published=true). The calendar event on your calendar IS created (even on Draft) so you can see the event in your own schedule planning; the calendar event has no attendees because no registrations have come in. Drafts are useful for: setting up an event days in advance and reviewing the description multiple times, getting a teammate to proofread the copy before going live, holding the event in private review until you have finalized the agenda.

Publish

Clicking Publish flips is_published to true. The public registration URL immediately resolves with the event details and the registration form. The event card on your dashboard loses the Draft badge and becomes a regular Active event. Existing calendar event on your calendar is unchanged; the platform does not re-fire any notifications on publish since no attendees exist yet.

Unpublish

Unpublish flips is_published back to false. The public registration URL returns a 404 again. EXISTING registrations are PRESERVED; their tickets are still valid; their calendar invites still work; the event is still on your calendar. Unpublishing is a way to STOP accepting new registrations without cancelling the event (a typical use case: you have 47 of your 50 capacity sold and want to stop the public flow even though you have not formally sold out; unpublish gives you that control). To resume accepting registrations, click Publish again. Unpublish does NOT email existing attendees; they are unaffected.

The public event page

Every published event has its own public URL at /event/<event-slug>. The slug is auto-generated from the event title (lowercase, hyphenated, deduplicated if needed). Share the URL via your website, social media, email blasts, partner referrals, or print it on a flyer with a QR code. The page is fully self-serve; prospects land on it, read the details, and register without your intervention.

Page layout

- Hero band: the cover image (3:1 banner) renders large at the top, full-width. Event title overlaid in white text on the bottom-left corner.
- Meta strip: a horizontal band under the hero with the date + time (both timezones), the meeting mode pill, the price pill (Free / CA$25.00 per attendee), and the registered count (e.g., '43 of 100 registered'). Sold-out events show 'Sold out' in place of the count.
- Description block: your full description text, rendered as multi-paragraph prose.
- Registration form: name + email + phone + 'How many attendees?' (capped at min(9, remaining capacity)). On multi-attendee, per-attendee name + email fields appear dynamically. On Both mode, a 'How will you attend?' radio. An optional discount-code field below ('I have a discount code'). For paid events, a 'Proceed to payment' button; for free events, a 'Register' button.
- Language picker: top-right, lets the prospect switch the page chrome between English and Quebec French. On Premium tenants with Phase 10C runtime translation, the picker offers the eight additional languages.
- Footer: tenant logo + 'Powered by RCIC App' attribution. The footer is intentionally minimal; the event details are what should hold the prospect's attention.

Social-share card

The page sets Open Graph + Twitter Card metadata so a link shared on LinkedIn, Twitter, Facebook, or WhatsApp renders with a rich preview: the cover image as the banner, the event title as the headline, the description's first paragraph as the snippet. If you do not have a cover image, the platform falls back to a generic 'RCIC App event' card; with a cover image the share is dramatically more engaging.

The registration flow, step by step

A prospect on the public event page goes through a multi-step registration flow. Free events have one less step (no payment); paid events have the full flow.

Step 1: Fill the form

Type the primary registrant's name, email, and phone. Set the attendee count (1 to min(9, remaining capacity)). Type each additional attendee's name + email if the count is greater than 1 AND the event has anonymous tickets OFF. Pick the attendance mode (online or in-person) if the event is in Both mode. Optionally type a discount code and click Apply to see the price drop in real time. The running total ('4 attendees x CA$25.00 = CA$100.00 minus CA$10.00 discount = CA$90.00') updates live as the prospect types.

Step 2: Payment (paid events only)

Click 'Proceed to payment'. The platform creates an event_registration row at status='pending_payment' (so the slots are held), creates a Stripe Checkout Session via Stripe Connect on your tenant's connected Stripe account, and redirects the prospect to Stripe's hosted checkout page. The prospect enters card or bank details, completes 3D Secure if their card requires it, and on successful capture is redirected back to the event page's confirmation step. If the prospect closes the Stripe page without paying, the registration sits at pending_payment for a 15-minute grace window; if no payment lands in that window, the cleanup cron purges the row and the slots are freed back into capacity.

Step 3: Confirmation page

A success page showing the registration reference number (EVT-YYYY-NNNN format), the event title, the date + time in both timezones, the attendee list, the meeting mode + location or video link, the price paid, and a 'check your inbox for your tickets' note. The page is the redirect target after Stripe Checkout completes for paid events AND the direct confirm for free events.

Ticket emails: per-attendee vs bundled, with QR codes

The moment a registration confirms, the platform emails tickets. The exact email shape depends on the bundled-vs-per-attendee decision (see the Anonymous tickets toggle section above): per-attendee mode sends one email PER attendee to that attendee's email address; bundled mode sends one email to the registrant with a single ticket covering all attendees.

Ticket email contents

- From: '<Your company name> via RCIC App' on the RCIC App branded transporter (noreply@rcicapp.ca). Reply-To: your tenant's primary email.
- Subject: 'Your ticket for <event title>'. On bundled mode, 'Your N tickets for <event title>'.
- The cover image at the top (the same 1500x500 banner you uploaded), the event title and date / time, the meeting mode + location or video link, the attendee's name (per-attendee) or the registrant's name + attendee count (bundled), and most importantly the QR code.
- A 'Cancel my registration' link at the bottom for the registrant (the link is an HMAC-signed token; clicking it auto-cancels the registration and triggers a refund per your refund policy). Available on the registrant's email only, not on each attendee's per-attendee email.

The QR code

Each ticket carries a QR code generated via the qrcode npm package at email-render time. The QR encodes a one-time-readable check-in URL with a per-ticket token: https://rcicapp.ca/event/checkin/<ticket-token>. The token is unique per attendee (per-attendee mode) or per registration (bundled mode), HMAC-signed, and stored hashed on the event_attendees row. Scanning the QR with any QR reader (a phone camera, a barcode scanner, an iPad-mounted scanner at your venue door) opens the check-in URL in the browser.

The check-in page

When you (the tenant) scan an attendee's QR at the door, the check-in URL opens on your phone or tablet and shows: the attendee's name (or registrant + attendee count on bundled), the event title, the registration reference, the registered timestamp, and a big maroon 'Mark attended' button. Tap it; the attendee_marked_attended_at timestamp is recorded on the event_attendees row, the page flashes a green checkmark with the message 'Welcome to <event title>, <attendee name>'. Subsequent scans of the same QR show 'Already checked in at <time>' instead of the button, so an attendee cannot accidentally check in twice or pass the ticket to someone else after they have already used it.

> **Note**: The check-in URL works without authentication. Anyone with the URL (or the QR that encodes it) can check the attendee in. This is intentional: at the door, you (or any staff member or volunteer you have helping) just need to scan and tap. There is no login friction. The single-use enforcement (the second scan reads as Already checked in) plus the HMAC token (a fake URL with a guessed token will not validate) provides the security. If you need to UN-check-in someone (an accidental tap), open the Attendees list in your dashboard and click 'Mark not attended' on that attendee's row.

24-hour reminder emails

A daily cron at /api/cron/event-reminders runs at 12:00 UTC and identifies every confirmed event_registration on every event starting in the next 24 hours (more precisely, between the cron tick and 24 hours later). For each, the platform sends a reminder email to the registrant (and to each per-attendee email if the event is in per-attendee mode). The reminder is a courtesy nudge so attendees do not forget; calendar providers also fire their own per-attendee reminders based on each attendee's calendar defaults, but a fair number of attendees never opened the original calendar invite at all, so the platform reminder catches them too.

Reminder email contents

- From + Reply-To: same as the original ticket email.
- Subject: 'Reminder: <event title> tomorrow at <time>'. The 'tomorrow' framing is in your tenant timezone; if the event is at 8 AM Toronto time and the reminder sends at 8 AM UTC, the recipient in Vancouver still gets it called 'tomorrow' relative to the event's tenant timezone.
- Body: a condensed version of the ticket email. Event title + date + time in both timezones, meeting mode + location or video link, the QR code re-rendered (so the attendee can check in directly from the reminder email without digging up the original ticket), the cancel link (registrant only).

Idempotency: a registration is reminded only once

Each event_registration row carries an event_reminder_sent_at timestamp (added in mig 137). The cron's SELECT filters for registrations where event_reminder_sent_at IS NULL; once the platform sends the reminder, it stamps the timestamp atomically with the send, so a re-run of the cron later in the day (e.g., a manual cron trigger by an operator) does not send a duplicate. If the SMTP soft-fails during the send, the row keeps event_reminder_sent_at NULL and the next cron run retries; the retry behaviour is identical to other reminder crons on the platform.

What about events more than 24 hours out

The cron sends only at the 24-hour mark. Events more than 24 hours out get no early reminder; events less than ~5 minutes out (the cron's MIN window edge) get the reminder when the next 12:00 UTC tick fires. The single 24-hour reminder is intentional; multiple reminders on the same event would annoy attendees and the platform deliberately does not implement a 7-day or 1-hour reminder to keep the inbox-respect promise.

The Attendees list

Click the 'Attendees' link on any event card on your dashboard. The Attendees page opens with the full registration list for that event, sortable + searchable + filterable.

Page layout

- Header: event title, date + time, capacity status (registered count / total capacity, or 'unlimited' if no cap), 'Back to Events' breadcrumb.
- Filter strip: All / Confirmed / Pending Payment / Cancelled / Attended / No-show. Click any pill to narrow the list.
- Search box: searches across registrant name, registrant email, attendee name, attendee email, and registration reference. Useful when you have 100+ attendees and need to find a specific one fast.
- Bulk-action toolbar at the top: appears when you tick the per-row checkboxes. Three bulk actions: Resend tickets (re-fires the ticket email + QR to selected rows), Mark attended (bulk check-in), Cancel registrations (with optional refund toggle for paid events).
- Per-registration rows: each registration is one expandable row showing the registrant's name + email + phone, the registration reference, the status, the attendee_count + a list of attendee names (per-attendee mode), the payment amount + status, the registered timestamp, and a per-row action menu (Resend ticket, Mark attended, Mark not attended, Cancel registration, Refund without cancelling).

Resend tickets

An attendee says they lost their ticket email. Click Resend ticket on their row. The platform re-renders the ticket email with the SAME QR code (the QR token does not rotate; the original ticket from the original email still works at check-in, AND the re-sent ticket works at check-in, because they encode the same token). Useful when a ticket lands in spam, gets accidentally deleted, or the attendee forwarded it to a friend and now needs their own copy. Bulk resend works the same: tick multiple rows, click the bulk Resend button, the platform fans out per-row resends.

Mark attended / Mark not attended

The most common attended-marking path is the QR check-in flow at the door (described above). But if you ran the event without a door-check (a Zoom webinar where you trust your attendance list, or a small in-person event where you knew everyone by face), you can mark people attended manually after the event. Tick the per-row checkboxes for everyone who showed up, click the bulk Mark attended button; the platform stamps attendee_marked_attended_at on every selected attendee. To undo a wrong attendance mark, click Mark not attended on the row (clears the timestamp).

Editing a published event

Click Edit on the event card. The same side panel as Add opens, pre-filled with the current values. Most fields are editable post-publish; the platform's behaviour on each field differs based on whether the change affects existing attendees.

Title, description, image: free to edit any time

These changes update the public registration page + the dashboard card immediately. Existing attendees do NOT get re-emailed; their original ticket retains the original title in its subject line, but if they click any link back to the event page they see the new content. The calendar event title on your calendar IS updated to match.

Date and time: prompts an existing-attendees confirmation

When you change date or time on an event with at least one confirmed registration, the Save click opens a confirmation modal: 'This event has N confirmed registrations. Changing the date / time will email every attendee with the new time. Continue?' Confirm: the platform updates the event row, updates the calendar event on your calendar (which re-fires the calendar invite to every attendee from your calendar provider), AND sends a custom

'Event time changed' platform email to every attendee with both the old time and the new time spelled out. The dual update guarantees every attendee notices.

Meeting mode and location: similar guardrail

Switching from Online to In-person (or vice versa) on an event with confirmed registrations triggers the same modal because attendees showed up expecting one mode and now need to know it changed. The custom email's body spells out the old mode + new mode + new location-or-link.

Capacity: live, no email

Raising capacity is trivial: type the new number, save, the public page now accepts more registrations. Lowering capacity below the current registered count is refused with an inline error; cancel attendees first (which refunds them) to free slots, then lower. No email goes to existing attendees on a capacity change; they neither need nor benefit from knowing the cap moved.

Cancelling an event (with refund cascade)

If you need to cancel an event entirely (a venue closure, a personal emergency, a regulatory change that obviates the topic), the dashboard surfaces a Cancel action on the event card. Cancellation is the heaviest operation in the module because it cascades to every confirmed registration.

The cancel modal

Click Cancel on the event card. A modal opens carrying: the registered count, the total paid amount across all paid registrations, an OPTIONAL free-text 'cancellation reason' field (whatever you type lands in the cancellation email to every attendee; useful for explaining 'venue closure' vs 'rescheduling to a later date you will announce'), and a confirm-by-typing-the-event-title safety check (you must type the event title verbatim in a confirm field before the Cancel button enables). The check is the platform's defense against fat-finger clicks on irreversible operations.

What happens on cancel

1.

The event row's status flips to cancelled and is_published is forced to false. The public registration URL returns a 'this event has been cancelled' message; new registrations are refused.

2. The calendar event on your calendar is DELETED (the calendar provider sends a 'meeting cancelled' notification to every attendee on the calendar invite as part of the delete; this is separate from the platform's cancellation email).
3. For every paid registration, the platform issues a FULL refund through Stripe Connect (the original payment amount, including the prospect's share of any GST/HST your tenant collected). The refund includes the platform fee that Stripe took at capture (proportionally refunded automatically). Refunds take 5-10 business days to appear on the prospect's statement.
4. Every confirmed registration's status flips to cancelled. The event_attendees rows are preserved; their attendance markers stay (a cancelled-event attendance does not auto-clear, in case you need the record for some reason).
5. Every attendee receives a custom cancellation email: 'The <event title> on <date> has been cancelled. <Your cancellation reason if you provided one>. <Refund amount and timing>.' The email is sent through the RCIC App branded transporter; Reply-To is your tenant email so attendees who have questions reach you directly.

> **Important**: Cancellation is irreversible at the application layer. You cannot un-cancel an event via the dashboard. If you cancelled by mistake and need to bring the event back, the only path is to create a NEW event with the same title + date + time (and ask the previously-registered attendees, who you can find in your dashboard's Cancelled list, to re-register on the new event). The refunds already issued cannot be reversed; you would need to take fresh payment on the new event.

Archiving an event

Archive is a SOFT lifecycle action that hides an event from the Active tab. Use it for events that have already happened and that you no longer need cluttering the Active list. The Archived tab on the Events page surfaces every archived event, sorted by event date descending (most recent first). The event card on the Archived tab carries an Unarchive button that flips the event back to the Active tab.

Two archive paths depending on the event state

- Past event with zero confirmed registrations (or all registrations are cancelled / past): the Archive button is a simple flag flip. The event moves to the Archived tab with no email fan-out + no calendar work. Use this for events that ran without anyone showing, or events you cancelled long ago and want off the active list.

- Past event with active attendance markers (people showed up): same simple flag flip. The historical attendance record is preserved on event_attendees rows.

- UPCOMING event with active confirmed registrations: clicking Archive cascades through the full Cancel flow described above (refund every paid registration, send cancellation emails, delete the calendar event) and THEN archives the event. The platform combines the two actions into a single Cancel-and-archive operation with a single confirmation modal. The cancellation reason field is mandatory in this flow (cancelling and archiving an active upcoming event is a meaningful decision; attendees deserve a reason).

Unarchive

On the Archived tab, click Unarchive on an event card to flip it back to Active. The public registration URL becomes resolvable again ONLY if is_published is also true. If the event was archived via the cancel-and-archive flow (so is_published was forced to false at cancellation), unarchiving alone does NOT re-publish; you also need to click Publish if you want the public page back. For a regularly-archived past event, unarchive is rare (you usually do not need past events in your active list); the action exists primarily so that an accidental archive does not have to be redone via a fresh event creation.

Discount codes (dedicated to Events)

Events has its OWN discount-code feature, separate from the Bookings discount codes. The Discounts page lives at Events sidebar -> Discounts (a sibling of the Events list). The shape is similar to Bookings discounts but the codes are independent: a Bookings code does NOT apply to Events registrations, and an Events code does NOT apply to Bookings reservations. The separation is intentional: the two modules have different pricing semantics (per-meeting vs per-attendee) and different campaign cadences (typical Bookings campaigns target prospects entering the funnel; typical Events campaigns target a specific event's promotion).

Where the Events Discounts page lives

Events sidebar -> Discounts (the page reachable from the Events module sidebar; do not confuse with Bookings sidebar -> Discounts). The page title reads 'Discounts' with the subtitle 'Create promo codes attendees can enter on the event registration form'. A maroon '+ New discount code' button sits in the top-right. Below the header, every existing code renders as a card with its code (uppercase + monospace), discount type and value, applies-to summary, validity window, usage counter, and Pause / Edit / Delete buttons.

Add a discount: the form

- Code (required): uppercase letters + digits + hyphens, 3 to 40 characters. The string attendees type on the registration form. Unique within your tenant per module (you could have a 'LAUNCH50' on both Bookings and Events without conflict).
- Description (optional): your internal label.
- Discount type: Percentage off (1-100) or Fixed amount off (dollars). Percentage applies to the per-attendee price; on a multi-attendee registration, the percentage discount applies BEFORE the attendee multiplication (a 10% discount on a $25 event for 4 attendees yields ($25 x 0.9) x 4 = $90, NOT $25 x 4 x 0.9 which is mathematically equivalent but conceptually different). Fixed amount also applies BEFORE multiplication: a $5 fixed discount on a $25 event for 4 attendees yields ($25 - $5) x 4 = $80.
-

Applies to (event picker): a checkbox list of every event in your dashboard (Active AND Archived, in case you want to retro-apply a code to a past event for record purposes; rare but possible). Leave all UNCHECKED to apply to every event including future ones; check specific events to restrict.

- Validity window (Valid from + Valid until): the same dates-driven gate as Bookings discounts. Validated at the EVENT START time, not the discount-application time.
- Maximum redemptions (optional): integer cap on total uses. Useful for 'first 20 sign-ups get 50% off' campaigns; the dashboard card shows '14 of 20 used' badge.

How attendees use a code on the registration form

Below the attendee fields, an 'I have a discount code' link reveals a small input + Apply button. The attendee types the code (case-insensitive), clicks Apply, the platform validates server-side: the code exists in YOUR tenant's Events codes, is not paused, is within its validity window, has remaining redemptions, applies to THIS event (per the applies-to picker). On valid, an inline confirmation appears: 'Discount applied: 10% off' or 'Discount applied: CA$5.00 off'. The running total below the attendee fields updates immediately to reflect the discount. On invalid (wrong code, paused, expired, cap reached, not applicable to this event), an inline error appears: the platform deliberately returns the same generic 'this code is not valid for this event' message regardless of the underlying reason, to avoid leaking signal that helps a malicious user enumerate valid codes.

> **Note**: Pause vs Delete are the same as on Bookings discounts: Pause temporarily disables; Delete is allowed only on codes with ZERO redemptions. Code + discount type + value are NOT editable on an existing code; description + applies-to + validity window + max redemptions + paused state are. The restrictions protect the audit trail of past redemptions.

Owner-creator routing (Team Mode)

On Team Mode tenants, every event carries an owner_user_id field set to the user_id of the team member who CREATED the event (no 'Assigned member' picker on the event form, unlike services). The owner is the creator; events do not get reassigned. The calendar event for an Online or Both event lands on the CREATOR's connected calendar (not on the company Owner's, unless those are the same person). The Edit / Cancel / Archive / Resend ticket actions on the event row enforce a 'staff can mutate own events; owner+admin can mutate any' rule.

Implications

- An assistant on Team Mode who creates an event owns it; they can edit + cancel + archive it without admin approval. They can also send the event-related notifications, manage attendees, and mark check-ins. Owner + admin always have full access regardless of who created the event.
- When a team member leaves the firm and you want to reassign their events to someone else: the platform does NOT support owner_user_id reassignment via the UI. The operator-handled SQL escape valve is the only path; email info@investatech.com with the team member's email + the new owner's email and we coordinate the migration. In practice this is rare; most tenants either let the events run their course under the original owner's calendar OR archive the events when the original owner leaves.
- The 24-hour reminder cron and the calendar conflict check both use owner_user_id to know whose calendar to query + whose freebusy to consult.

Troubleshooting

Attendees say their tickets did not arrive

First check the platform side: open the Attendees list for the event, find the registration, look at the per-row ticket-status indicator. If it reads 'Pending' the SMTP send failed; click Resend ticket. If it reads 'Sent', the email left our infrastructure successfully and the attendee should check their spam folder (the RCIC App branded domain is reputable but spam filters are aggressive on a transactional email with a QR image attached). Bulk-resend tickets on a multi-attendee registration: tick the row, click Resend ticket; per-attendee mode sends a fresh ticket to each attendee's email.

The QR code is not scanning at the door

Usually a camera issue (low light, an extremely matte phone screen, an image whose contrast got destroyed by the attendee's email client's dark-mode rendering). Two manual workarounds: (a) ask the attendee to read the URL under the QR (the check-in URL is text-printed below every QR specifically for this case); type it into your phone, the same check-in page opens, click Mark attended. (b) On your dashboard's Attendees list, find the attendee by name + email, click the per-row Mark attended button. The audit ledger records a manual-check-in event differently from a QR-scan event (both stamp attendee_marked_attended_at but the actor differs).

Discount code is not applying

Verify on your dashboard: Events sidebar -> Discounts. Find the code, check that (a) it is not Paused, (b) the validity window covers TODAY + the event's start time, (c) the applies-to picker either includes the event in question OR is empty (applies to all), (d) the maximum-redemptions cap has not been reached. A discount that fails all four of these still shows on your dashboard but the registration form returns the generic 'not valid' message. If everything looks right, ask the attendee to copy the code verbatim from your email or campaign asset (a typo in a partner-shared code is the most common cause; the form is case-insensitive but typos in the middle of the string still fail).

The event is not showing the Google Meet / Teams link

This happens when the calendar event create-call to Google or Microsoft returned WITHOUT the conference link populated (Google occasionally provisions conferences asynchronously on Workspace domains; the platform's defensive 1.5-second wait + events.get re-fetch usually catches this, but sometimes Google takes longer). Open the calendar event on your calendar (Google Calendar or Outlook web); inspect the conference section. If the link is there, the platform may have missed the re-fetch; on the event card in your dashboard, click Edit, save without changing anything, and the platform re-reads the calendar event and stamps the link. If the link is genuinely not there on your calendar either, the conference provisioning failed (sometimes a Google Workspace domain policy disables Meet for external users); in that case, switch to 'Bring your own video link' mode and paste a Zoom URL or your own Meet room URL.

An attendee paid but their registration is still Pending Payment

The Stripe webhook delivery may have been delayed (rare but possible during a Stripe-side outage). The platform's defensive retry is at the Stripe side: Stripe re-delivers failed webhooks for up to 3 days at exponential backoff. Wait 5-10 minutes; the webhook usually lands. If the registration is still Pending Payment after 30 minutes AND you can see the charge in your Stripe dashboard, the webhook may have routed to the wrong URL (verify the Stripe Connect webhook endpoint in your Stripe Dashboard is https://rcicapp.ca/api/webhooks/stripe). Email info@investatech.com with the Stripe payment_intent_id and we can manually flip the registration to Confirmed and ship the ticket.

> **Note**: Most Events issues trace back to one of three sources: Stripe Connect health (for paid event payment flow), calendar connection health (for Meet / Teams link generation), or SMTP transient failures (for ticket and reminder delivery). The platform surfaces an amber banner on the affected event card when any of these three are in a degraded state. Click the banner to land on the relevant settings page or to retry the specific action.