

AuroraStack Calendar Web Pro User Manual

AuroraStack Calendar Web Pro User Manual

AuroraStack Calendar Web Pro is the standalone Pro calendar for non-WordPress websites and custom applications. It provides the same display surface as the WordPress Pro plugin, plus a browser-based Web Manager for preparing event data, calendar profiles, Hall of Fame entries, and JSON files.

Web Pro is suitable for static websites, React/Vue/Angular applications, custom portals, membership sites, and website builders that allow custom HTML, CSS, and JavaScript.

Secure workflows such as licence checks, registrations, paid-event checkout, protected downloads, reminders, WhatsApp Cloud API messaging, and advanced imports use backend endpoints. The component provides the UI and calls those configured endpoints.

Table Of Contents

1. [What Web Pro Includes](#what-web-pro-includes)
2. [Package Contents](#package-content)
3. [Install The Web Component](#install-the-web-component)
4. [Use The Web Manager](#use-the-web-manager)
5. [Create And Manage Calendars](#create-and-manage-calendars)
6. [Add, Edit, Delete, And Import Events](#add-edit-delete-and-import-events)
7. [Customise Calendar Settings](#customise-calendar-settings)
8. [Calendar View](#calendar-view)
9. [List View](#list-view)
10. [Image And Calendar View](#image-and-calendar-view)
11. [Registrations And Payments](#registrations-and-payments)
12. [Calendar Sync And ICS Feeds](#calendar-sync-and-ics-feeds)
13. [Reminders And WhatsApp](#reminders-and-whatsapp)
14. [Hall Of Fame](#hall-of-fame)
15. [Import Hall Of Fame Entries](#import-hall-of-fame-entries)
16. [Licence And Plan Entitlements](#licence-and-plan-entitlements)
17. [Backend Endpoints](#backend-endpoints)
18. [Publish The Finished Data](#publish-the-finished-data)
19. [Mobile And Tablet Behaviour](#mobile-and-tablet-behaviour)
20. [Security And Privacy](#security-and-privacy)
21. [Troubleshooting](#troubleshooting)

What Web Pro Includes

Web Pro includes:

- Month calendar, list, and image plus calendar views.
- Web Manager for settings, calendars, event data, and Hall of Fame data.
- Multiple named calendars.
- Per-calendar visual profiles, so different calendars can have different looks.
- CSV event import.
- Event icon URL support.
- Event details links.
- Registration and payment buttons.
- Calendar subscription and ICS download controls.
- Post-event file links.
- Full-description popovers and mobile bottom sheets.
- Multi-day events shown on each covered day.
- Multiple season definitions.
- Hall of Fame year tabs and shiny gold, silver, or bronze plaques.
- Hall of Fame CSV import.
- Licence verification hook.
- Backend hooks for secure registrations, payments, reminders, WhatsApp jobs, protected downloads, and admin imports.

Package Contents

The Pro ZIP contains:

- `aurorastack-calendar.css`: visual styles.
- `aurorastack-calendar.js`: the Web Component.
- `edition.js`: marks the package as Pro.
- `edition.json`: package metadata.
- `manager/index.html`: browser-based Web Manager.
- `manager/aurorastack-web-manager.js`: manager logic.
- `manager/aurorastack-web-manager.css`: manager styles.
- `examples/index.html`: basic example page.
- `examples/events.json`: sample data.
- `docs/PRO-DOCUMENTATION.md`: this manual.
- `docs/AuroraStack-Calendar-Web-Pro-User-Manual.pdf`: PDF copy of this manual.
- `docs/screenshots/`: Web Pro screenshots used by the manual.

Install The Web Component

Upload these files to your website:

- aurorastack-calendar.css
- edition.js
- aurorastack-calendar.js
- events.json

Add the stylesheet and scripts:

```
<link rel="stylesheet" href="/calendar/aurorastack-calendar.css">
<script src="/calendar/edition.js"></script>
<script src="/calendar/aurorastack-calendar.js" defer></script>
```

Add the calendar where it should appear:

```
<aurorastack-calendar src="/calendar/events.json" view="calendar"></aurorastack-calendar>
```

Embed a named calendar:

```
<aurorastack-calendar
  src="/calendar/events.json"
  view="list"
  calendar="members">
</aurorastack-calendar>
```

Embed the Hall of Fame:

```
<aurorastack-calendar
  src="/calendar/events.json"
  view="hall_of_fame">
</aurorastack-calendar>
```

Use The Web Manager

Open `manager/index.html` from the package. If your browser blocks local file access, serve the package from a local web server and open the manager through `http://localhost`.

The manager stores draft data in the current browser's local storage until you export JSON. Export before clearing browser data, changing computers, or publishing.

Data Summary

Events: 7
Calendars: 3
Hall entries: 6
Edition: pro-test

Preview view: Calendar
Preview calendar: All calendars

Changes are saved in this browser until you export the JSON data file.
[Reset local data](#)

Settings
Global settings apply to all calendars unless a calendar profile overrides them.

General

Calendar title: AuroraStack Test Pro
List title: Upcoming Test Event
Default view: Calendar
Week starts: Monday
Force centre: ☒
Show sync controls: ☒
Show WhatsApp: ☐

Calendar Layout

Calendar width: 1180px
Calendar height: auto
Cell width: 150px
Cell height: 170px
Event icon size: 64px
Icon background: transparent
Image position:
Hero image URL: /test-site/assets/bac

List View

Card width: 520
Card height: 260
Columns: 2
Fit width to cards: ☒

Colours

Accent:
Text colour:
Muted colour:
Cell background: rgba(18, 50, 59, 0.82)
Empty cell background: rgba(31, 41, 55, 0.62)
Event background: rgba(0, 0, 0, 0.58)
Event text colour:
Button background:
Second button colour:
Button text:

Sync and Endpoints

Webcal URL: webcal://localhost:81
Feed URL: http://localhost:8091
Registration endpoint: /api/register
Payment endpoint: /api/payments/stripe
Licence endpoint: /api/license/verify
Public site key: test-site-key

Hall of Fame

Title: AuroraStack Hall of F
Width: 1180px
Card width: 520px
Card height: 180px
Name font size: 28px
Detail font size: 16px
Layout: Horizontal cards

[Save settings](#)

Web Manager settings

The manager provides:

- Settings tabs for general, calendar, list, colour, season, sync, registration/payment, and Hall of Fame options.
- Separate event calendars.
- Per-calendar visual profiles.
- Event add/edit/delete screens.
- Event CSV import.
- Hall of Fame add/edit/delete screens.
- Hall of Fame CSV import.
- Live preview using the same Web Component customers embed on their websites.
- JSON export for publishing.

Recommended workflow:

1. Open the Web Manager.
2. Create or select a calendar.
3. Configure the global settings.
4. Add per-calendar visual overrides if required.
5. Import or add events.
6. Add Hall of Fame entries if required.
7. Use Live Preview to test calendar, list, image-month, and Hall of Fame views.
8. Export JSON.
9. Upload the exported file as events.json.
10. Embed the calendar on the target website.

Create And Manage Calendars

Web Pro supports multiple named calendars. This is useful when one website needs separate public events, members-only programmes, competitions, training sessions, or different club sections.

In the manager:

1. Open the Calendars section.
2. Add a calendar name, such as Public Events.
3. Confirm the slug, such as public-events.
4. Optionally set a per-calendar title and visual profile.
5. Save the data.

Assign an event to a calendar:

1. Open Events.
2. Add or edit an event.
3. Select the target calendar.
4. Save.

Import into a specific calendar:

1. Open the import screen.
2. Choose the destination calendar.
3. Upload or paste the CSV.
4. Confirm the imported rows.

Import using a calendar column:

```
title,start,end,calendar,location,description
Opening Night,2026-09-04 19:00,2026-09-04 21:30,public-events,Main Hall,Season launch evening
Print Competition,2026-09-16 20:00,2026-09-16 22:00,competitions,Club Room,Monthly print competition
```

The component accepts `calendar`, `calendar_slug`, `calendar_id`, `event_calendar`, `calendars`, `calendarSlugs`, or `calendar_ids`.

To display only one calendar:

```
<aurorastack-calendar src="events.json" calendar="competitions"></aurorastack-calendar>
```

Add, Edit, Delete, And Import Events

The Web Manager event screen gives a WordPress-like workflow for standalone sites.

Data Summary

Events	7
Calendars	3
Hall entries	6
Edition	pro-test

Preview view
Calendar

Preview calendar
All calendars

Changes are saved in this browser until you export the JSON data file.

Reset local data

Events

Add, modify, cancel, or assign events to a named calendar.

Filter calendar All calendars

TITLE	CALENDAR	START	STATUS
Opening Night	public-programme	4 Sept 2026, 19:00	Paid
Photography Outing Weekend	members	12 Sept 2026, 10:00	Registration
Speaker: Modern Landscapes	public-programme	18 Sept 2026, 20:00	Registration
Digital Competition	competitions	24 Sept 2026, 18:30	Paid
Print Competition	competitions	8 Oct 2026, 19:30	Open
Workshop: Editing Clinic	members	17 Oct 2026, 09:30	Paid
Annual Exhibition	public-programme	5 Nov 2026, 10:00	Open

opening-nigh

public-progr

04/09/20:

/test-site/ass

Opening Nigh

04/09/20:

Main Hall

/test-site/eve

Season launch with a welcome talk, refreshments, a walkthrough of the annual programme, and a short showcase of what the calendar can display when an event has a longer description. Hover, focus, or tap

Registration

Amount

Post-event files

Manage URL

Payment required

Currency

Programme PDF | /test-site/downloads/opening-night-programme.pdf

/test-site/mai

Save event

Delete event

Web Manager event editing

Add an event:

1. Open Manager > Events.
2. Select the calendar the event belongs to.
3. Click Add Event.
4. Enter title, start date/time, end date/time, location, and description.
5. Add an optional event icon URL.
6. Add an optional details URL.
7. Enable registration or payment if this event needs it.
8. Add post-event file links if files should be shown after the event has finished.
9. Save.

Edit an event:

1. Open Manager > Events.
2. Find the event.
3. Click Edit.
4. Change title, description, dates, calendar, icon, details URL, registration, payment, or file links.
5. Save.

Delete an event:

1. Open Manager > Events.
2. Find the event.
3. Click Delete.
4. Export JSON after confirming the event should no longer appear.

CSV import columns:

title,start,end,location,description,calendar,icon_url,details_url,registration_enabled,payment_required,amount
Opening Night,2026-09-04 19:00,2026-09-04 21:30,Main Hall,Season launch,public-events,https://example.com/icon

Accepted event icon columns:

- icon_url
- event_icon_url
- event_icon
- image_url
- event_image
- image
- icon

Accepted details URL columns:

- details_url
- post_url
- url

Accepted file format in CSV:

Programme PDF | <https://example.com/programme.pdf>; Recording | <https://example.com/recording.mp4>

Multi-day events:

- Use one row with a start date and an end date.
- Calendar view shows the event on each day it spans.
- List view creates a visible card for each covered day in the selected month.
- The same event icon is shown on the start day and all subsequent displayed days.

Customise Calendar Settings

The manager separates settings into logical sections so it is easier to see what affects the whole calendar, the grid, list cards, colours, sync, and Hall of Fame.

Main setting groups:

- General: title, list title, default view, week start, alignment, sync visibility, WhatsApp visibility.
- Calendar layout: width, height, cell width, cell height, image placement, hero image, background image.
- List view: card width, card height, column count, and auto-fit behaviour.
- Colours: accent, text, muted text, day cells, empty cells, event cards, buttons, arrows, sync buttons.
- Fonts: general font, day number font, event font, selected event font, button font.
- Images: background fit, repeat, position, event background images, monthly calendar backgrounds, monthly event backgrounds.
- Borders: border image, border width, and custom border URLs.
- Seasons: one or more season ranges.
- Sync: webcal URL, feed URL, and subscription controls.

- Registrations and payments: endpoint URLs and checkout behaviour.
- Hall of Fame: heading text, width, border width, plaque size, card layout, font sizes, and metal style.
- Calendar profiles: per-calendar visual overrides.

Per-calendar profile example:

```
{
  "settings": {
    "calendarProfiles": {
      "competitions": {
        "enabled": true,
        "settings": {
          "title": "Competitions Calendar",
          "accent": "#f97316",
          "eventBackground": "rgba(20, 20, 20, 0.88)",
          "eventTextColor": "#ffffff",
          "listCardWidth": 620,
          "listColumns": 1
        }
      }
    }
  }
}
```

Calendar View

Calendar view shows one month at a time with previous/next arrows and month/year jump controls.

Use:

```
<aurorastack-calendar src="events.json" view="calendar"></aurorastack-calendar>
```

Important settings:

- width: calendar width.
- height: optional calendar height.
- cellWidth: minimum cell width.
- cellHeight: minimum cell height.
- forceCenter: centre the calendar even when the host page tries to align it differently.
- showWhatsApp: show or hide WhatsApp share buttons.
- showSync: show or hide sync controls.
- arrowBackground: previous/next arrow colour.
- arrowText: previous/next arrow text colour.

The calendar expands to honour the configured cell width where the browser has room. On small screens it switches to a stacked mobile layout to avoid unreadable squeezed cells.

List View

List view displays equal-sized cards for one month at a time.

Use:

```
<aurorastack-calendar src="events.json" view="list"></aurorastack-calendar>
```


List card settings:

- `listCardWidth`: width of each card.
- `listCardHeight`: fixed height, or 0 for automatic height.
- `listColumns`: number of columns.
- `fitListWidthToCards`: sizes the list shell around the configured card width and column count.

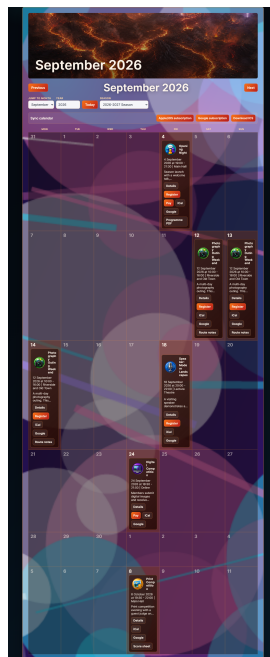
List cards use:

- Event icon and title at the top.
- Compact action buttons in the top-right corner.
- Date/time in the left column.
- Location and description in the right column.
- Justified description text.
- Full-description popup only when the description is clipped.

One-column list cards can be up to around 1200px wide. Multi-column cards are capped lower so the layout remains usable.

Image And Calendar View

Image-month view adds a visual feature image to the calendar.



Web Pro image calendar

Use:

```
<aurorastack-calendar  
  src="events.json"  
  view="image_month"  
  image-position="top">  
</aurorastack-calendar>
```

Image placement options:

- top
- left
- right

For a lava or volcano design:

1. Set backgroundImage to a lava or volcano image.
2. Set heroImage to the same or related image.
3. Use dark transparent cells.
4. Use bright orange accents.
5. Use pale event text.
6. Use a contrasting event background image or dark event card.

Example:

```
"settings": {
  "title": "Volcanic Season",
  "backgroundImage": "assets/backgrounds/realistic-volcano-lightning.jpg",
  "heroImage": "assets/backgrounds/realistic-volcano-lightning.jpg",
  "backgroundSize": "cover",
  "cellBackground": "rgba(21, 4, 4, 0.68)",
  "eventBackground": "rgba(18, 18, 18, 0.88)",
  "eventTextColor": "#fff7ed",
  "accent": "#fb923c",
  "buttonBackground": "#f97316",
  "buttonText": "#ffffff"
}
```

Registrations And Payments

Web Pro can display registration and payment buttons. The browser component does not store registrations or collect card details itself; it calls the configured endpoints.

Registration settings:

- registrationEndpoint: endpoint that receives registration submissions.
- registrationEnabled: can be set per event.
- registrationFields: optional extra field configuration.

Payment settings:

- paymentEndpoint: endpoint that creates a checkout session.
- paymentRequired: can be set per event.
- amount: event price.
- currency: event currency, such as gbp.
- paymentUrl: optional hosted checkout URL for simple handoff.

Example event:

```
{
```

```

    "id": "workshop-1",
    "title": "Portrait Workshop",
    "start": "2026-10-12T10:00:00",
    "end": "2026-10-12T16:00:00",
    "registrationEnabled": true,
    "paymentRequired": true,
    "amount": 35,
    "currency": "gbp"
}

```

Stripe Checkout can provide Apple Pay and Google Pay when those payment methods are enabled in Stripe and the visitor's device/browser supports them.

Calendar Sync And ICS Feeds

Sync controls can show:

- Apple/iOS subscription.
- Google subscription.
- Download ICS.

Set the active feed:

```

"settings": {
  "subscriptionUrl": "webcal://calendar.example.com/events/feed.ics",
  "feedUrl": "https://calendar.example.com/events/feed.ics",
  "showSync": true
}

```

Useful aliases:

- subscriptionUrl
- webcalUrl
- calendarSubscriptionUrl
- feedUrl

Per-calendar feeds:

```

"settings": {
  "feedUrl": "https://calendar.example.com/events/{calendar}/feed.ics"
}

```

If the URL has {calendar} or {calendar_slug}, Web Pro substitutes the active calendar slug. Otherwise it appends calendar=slug automatically where possible.

Subscribed calendars can refresh when the calendar app re-reads the feed. Downloaded ICS files are snapshots and do not auto-update.

Reminders And WhatsApp

Web Pro provides UI hooks for reminders and WhatsApp jobs. Actual sending happens on a backend service.

Common reminder configuration:

```

"settings": {

```

```

    "reminderEndpoint": "https://api.example.com/license-admin/web-pro/reminders",
    "defaultLeadMinutes": 1440,
    "showWhatsApp": false
}

```

Manual WhatsApp share buttons are hidden by default. To show them:

```

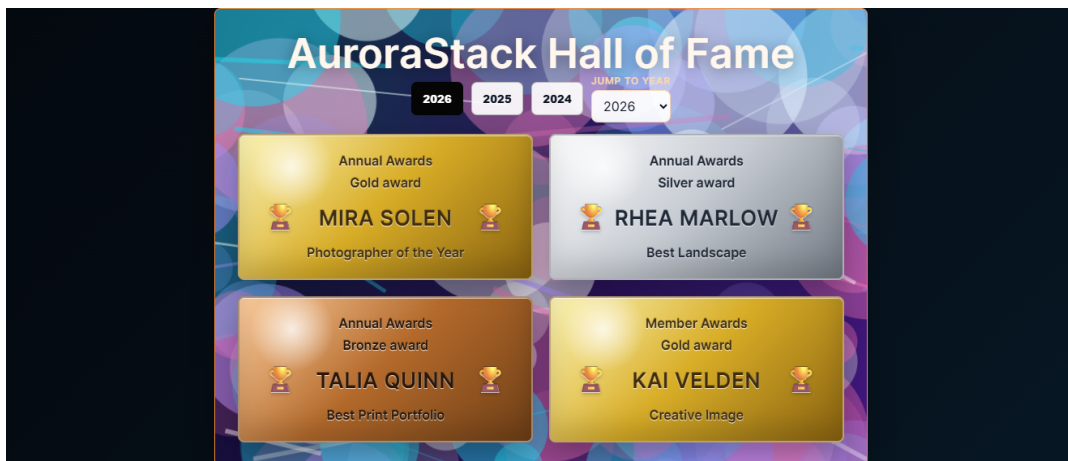
"settings": {
  "showWhatsApp": true,
  "whatsappButtonBackground": "#16a34a",
  "whatsappButtonText": "#ffffff"
}

```

For automatic WhatsApp community or group reminders, configure the customer's WhatsApp Cloud API phone number ID and token on the backend licence configuration. Do not place WhatsApp access tokens in browser JSON.

Hall Of Fame

Hall of Fame view displays entries under year tabs. It is designed for awards, competition winners, achievements, alumni recognition, sports honours, and club plaques.



Web Pro Hall of Fame

Use:

```
<aurorastack-calendar src="events.json" view="hall_of_fame"></aurorastack-calendar>
```

Hall of Fame settings:

- hallOfFameTitle: heading text.
- hallOfFameTitleFontFamily: heading font.
- hallOfFameTitleFontSize: heading size.
- hallOfFameWidth: overall width.
- hallOfFameBorderWidth: outer border width.
- hallOfFameCardWidth: plaque width.
- hallOfFameCardHeight: plaque height.
- hallOfFameLayout: rows or horizontal.
- hallOfFamePersonFontSize: person-name size.
- hallOfFameDetailFontSize: event and achievement text size.

- hallOfFameMetalStyle: default metal colour.

Position controls plaque colour:

- 1: gold.
- 2: silver.
- 3: bronze.
- Any other value uses the default metal style.

If no image URL is provided, the image section is hidden.

On phones, long person names are fitted onto one line and all visible plaques keep a consistent size.

Import Hall Of Fame Entries

CSV columns:

```
year,position,image_url,person_name,achievement,event_name,event_description
2026,1,https://example.com/winner.jpg,A. Winner,Best Portfolio,Annual Awards,Gold award
2026,2,,B. Finalist,Best Print,Annual Awards,Silver award
```

In the manager:

1. Open Hall of Fame.
2. Choose Import.
3. Paste or upload the CSV.
4. Review the parsed entries.
5. Save.
6. Preview the Hall of Fame view.
7. Export JSON.

Manual Hall of Fame entry:

1. Open Hall of Fame.
2. Click Add Entry.
3. Enter year, position, optional image URL, person name, achievement, event name, and event description.
4. Save.

Edit or delete Hall of Fame entries from the same screen before exporting JSON.

Licence And Plan Entitlements

When a licence endpoint and licence key are configured, Web Pro verifies the licence and disables features that are not included in the active plan.

Plan feature guide:

| Plan | Included Web Pro features |

| --- | --- |

| Personal | Advanced imports and calendar sync |

| Professional | Personal plus registrations, payments, Hall of Fame, post-event downloads, and multiple named calendars |

| Business | Professional plus advanced visual designs, monthly backgrounds, webhook reminders, and higher limits |

| Agency | Business plus WhatsApp Cloud API workflows and highest site/origin limits |

Example licence settings:

```
"settings": {
  "licenseEndpoint": "https://license.aurorastackltd.co.uk/public/web-pro/license/verify",
  "licenseKey": "AUR-....",
  "licenseEmail": "customer@example.com",
  "publicSiteKey": ""
}
```

Expired, invalid, or over-limit licences do not unlock paid features. Backend endpoints should also enforce the same entitlement checks.

Backend Endpoints

Public browser endpoints:

```
POST /public/web-pro/license/verify
POST /public/web-pro/registrations
POST /public/web-pro/payments/stripe-checkout
GET /public/web-pro/downloads/{token}
```

Admin/server endpoints:

```
POST /license-admin/web-pro/imports/events
POST /license-admin/web-pro/reminders
POST /license-admin/web-pro/whatsapp/jobs
POST /license-admin/web-pro/downloads/tokens
POST /license-admin/web-pro/jobs/process-due
GET /license-admin/web-pro/items?kind=registration
GET /license-admin/web-pro/items?kind=payment
GET /license-admin/web-pro/items?kind=import
GET /license-admin/web-pro/items?kind=reminder
GET /license-admin/web-pro/items?kind=whatsapp
GET /license-admin/web-pro/items?kind=download
GET /license-admin/licenses/{id}/web-pro-config
PUT /license-admin/licenses/{id}/web-pro-config
```

Admin calls require:

X-Admin-API-Key: <admin key>

Registration request example:

```
{
  "licenseKey": "AUR-....",
  "licenseEmail": "customer@example.com",
  "eventId": "meeting-1",
  "eventTitle": "Community Meeting",
  "name": "Visitor Name",
  "email": "visitor@example.com",
  "phone": "",
  "notes": "",
  "paymentRequired": true,
  "amount": 12.5,
}
```

```
"currency": "gbp",
"siteUrl": "https://example.com"
}
```

Payment checkout request:

```
{
  "licenseKey": "AUR-...",
  "licenseEmail": "customer@example.com",
  "registrationId": "4f6d...",
  "eventId": "meeting-1",
  "eventTitle": "Community Meeting",
  "email": "visitor@example.com",
  "amount": 12.5,
  "currency": "gbp",
  "successUrl": "https://example.com/payment-success",
  "cancelUrl": "https://example.com/payment-cancel",
  "siteUrl": "https://example.com"
}
```

Stripe event-payment webhook:

POST /webhooks/web-pro/stripe

Subscribe it to `checkout.session.completed` in Stripe and store the signing secret as `AURORA_WEBPRO_STRIPE_WEBHOOK_SECRET`.

Publish The Finished Data

1. In the manager, confirm the Live Preview looks correct.
2. Export JSON.
3. Save the file as `events.json`.
4. Upload it to the folder that contains the component files.
5. Open the public page and test calendar, list, image-month, and Hall of Fame views.
6. Test all links, downloads, registration buttons, payment buttons, and sync buttons.

Production embed:

```
<aurorastack-calendar
  src="/calendar/events.json"
  view="calendar"
  calendar="public-events"
  force-center="true"
  show-whatsapp="false">
</aurorastack-calendar>
```

Mobile And Tablet Behaviour

Web Pro includes phone and tablet-specific layout behaviour:

- Month grids change to a readable stacked layout on phone portrait screens.
- Year fields remain wide enough for all digits.
- Event icons remain contained and close to their configured size.
- List cards collapse to one column.
- Full descriptions open as a mobile bottom sheet with a close button.

- Manage, register, pay, sync, and download actions remain touch-friendly.
- Hall of Fame plaques keep consistent sizing and fit person names onto one line on phones.

Test at common widths:

- 390px phone portrait.
- 430px larger phone portrait.
- 768px tablet portrait.
- 1024px tablet landscape.
- 1366px desktop.

Security And Privacy

Do:

- Serve the component and JSON over HTTPS.
- Keep API secrets on the backend.
- Store Stripe secrets only in server configuration.
- Store WhatsApp Cloud API tokens only in backend licence configuration.
- Use allowed origins/domains on the licence server.
- Provide privacy wording for registrations and payments.
- Use protected download endpoints for files that should not be public.

Do not:

- Place Stripe secret keys in `events.json`.
- Place WhatsApp access tokens in browser-accessible files.
- Put private file URLs in public JSON unless they are intended to be public.
- Rely only on browser-side licence checks for paid features.

Troubleshooting

The manager opens but the home preview does not reflect settings:

- Export the latest JSON from the manager.
- Confirm the page is loading the exported file rather than an old `events.json`.
- Clear the browser cache or change the file URL with a version query string.

The calendar looks clipped:

- Reduce `cellWidth` or overall width.
- Confirm the host page is not forcing a narrow parent container.
- Use `force-center="true"`.
- For very wide calendars, place the component inside a full-width section.

Google subscription opens login but does not subscribe:

- Confirm `feedUrl` is a public HTTPS URL.
- Confirm the feed returns valid ICS content.
- Google cannot subscribe to local `localhost` feeds or private URLs.

Apple/iOS subscription opens Edge instead of Calendar:

- Use a `webcal://` subscription URL for Apple/iOS.
- Provide a matching public `https://` feed URL for browsers that do not handle `webcal://`.

Registrations or payments fail:

- Check `registrationEndpoint` and `paymentEndpoint`.
- Confirm the browser can call the endpoint from the site origin.
- Confirm CORS/allowed origins on the backend.
- Confirm the licence plan includes registrations/payments.

Hall of Fame plaques look inconsistent:

- Set `hallOfFameCardWidth` and `hallOfFameCardHeight`.
- Use `hallOfFameLayout: "horizontal"` for multi-card rows.
- Keep `hallOfFamePersonFontSize` larger than `hallOfFameDetailFontSize`.
- Avoid extremely narrow card widths for long names.