SynchroUI User Guide

Version 1.0.134

3 September 2026

Contents

SynchroUI User Guide

Documented against	SynchroServer `2.7.9`
SynchroAPI	`v1` (`/api/v1/`)

Last updated	2026-09-03

SynchroUI is the React-based dashboard for SynchroServer. It connects to a SynchroServer instance over the SynchroAPI (`/api/v1/`), browses your time series store, displays charts and alarm state, and lets you save reusable dashboards as **Views**.

This guide walks through the UI from the user's perspective: every button, every modal, every concept. For the underlying REST API see the SynchroAPI reference next to this file.

Logging in

Open SynchroUI in a browser (e.g. `http://<your-synchroserver-host>/SynchroUI/`). You'll land on the login page.

Login form: - **Username** — your SynchroServer account username (default admin: `synchro`). - **Password** — your SynchroServer account password. - **Sign in** — submits credentials. The button shows a loading state while the request is in flight.

If credentials are wrong, or SynchroServer is unreachable, an inline error message appears under the form.

> ⚠️ **The default `synchro` account ships with the password `synchro`.** It is an administrator, and the password is published in the product documentation. Change it as soon as you first log in, using the **Password** button in the header (changing your password).

Too many failed attempts

From SynchroServer 2.7.9, repeated failed logins are rate-limited per client address: 5 failures in a minute, or 10 in fifteen minutes, and further attempts are refused for a while. The error message says how long to wait. Successful logins are never delayed and reset the counter, so a few typos cost you nothing.

The version popup

Hover the **Synchro logo** in the upper-left corner of the login page (and later, of every page). A small black popup appears showing three lines:

```
SynchroServer:  SynchroServer-2.7.9
SynchroAPI:     2.7.9.052
SynchroUI:      1.0.134
```

This is the easiest way to confirm which build of each component you're talking to — handy when debugging or filing a bug report.

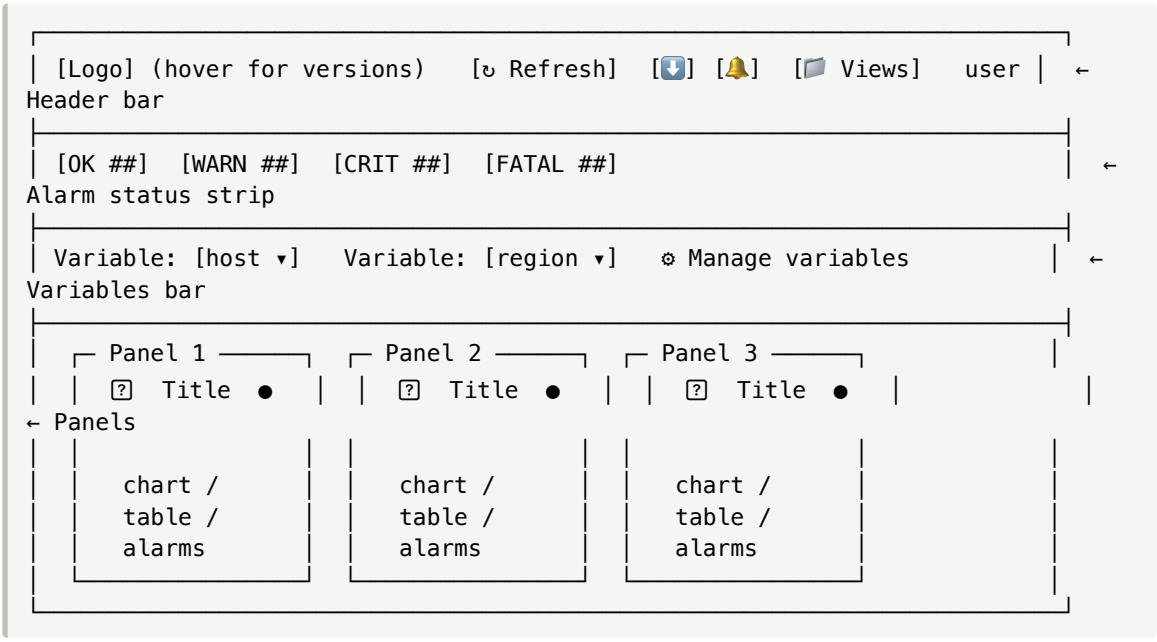
Session length

The login token (a JWT) is valid for 8 hours by default. After it expires, the next API call returns 401 and the UI redirects you back to the login page.

> 🔓 **Guest mode**: shared "guest" links bypass the login form entirely. See Guest view mode below.

Anatomy of the screen

After login, the main dashboard page is laid out as follows, top to bottom:



The variables bar and the alarm status strip are hidden in **guest view mode**. Everything else behaves the same.

The header bar

From left to right.

Logo (with version popup)

Hover the Synchro logo to see the SynchroServer / SynchroAPI / SynchroUI versions, exactly as on the login page. The popup is purely informational — nothing happens on click.

Refresh control

Two parts:

- ↻ **Refresh now** — invalidates every panel's query cache and refetches immediately. Useful if you've just made a change in SynchroServer and want the UI to reflect it without waiting for the next auto-refresh tick.
- **Dashboard refresh** dropdown — applies a single refresh interval to *all* panels at once, overriding their individual settings:
  - **Per-panel** *(default)* — each panel uses its own interval (set in its Search tab).
  - **Paused** — auto-refresh stops everywhere. The control turns red so you can't miss it. Manual ↻ Refresh still works.
  - **5 s, 10 s, 30 s, 1 m, 5 m, 15 m** — fixed interval applied to every panel until you switch back to **Per-panel**.

> 💡 The dashboard override is the right tool when you're presenting on a TV wall and want every panel ticking in lockstep, or when you're investigating something and don't want refreshes to wipe out your scroll position.

Releases / downloads icon (⬇️)

Opens the **Releases modal**. From there an admin can list the release files in `data/releases/` on the server, generate a signed time-limited download link, and revoke previously issued links. See Releases and download links below.

The icon is hidden in guest view mode.

Alarm bell (🔔)

Opens the **Alarms management dropdown** — a focused panel for adding, editing, and toggling alarm rules without leaving the dashboard. See Alarms for the full walkthrough.

The bell is hidden in guest view mode.

Views menu (📁)

Opens the **Views dropdown**, listing every view you can access:

- Your own views.
- Views shared with you by other users (`sharing: ["alice", "bob"]`).
- Views the owner has marked as `public`.

Click any row to load that view. The current view's name is shown in the header next to the menu icon and is truncated if it's very long.

For each row in the list:

- **Globe icon** *(your own views only)* — toggles `public / private`. Public views can be browsed by anyone with a guest link.
- **Copy guest link** *(public views only)* — copies a `?##<base64(viewname)>` URL to the clipboard. Recipients open the link and see the view in read-only guest mode (no login).
- **Delete** *(your own views only)* — asks for confirmation, then removes the view from SynchroServer.

Below the list:

- **Save** — overwrites the currently loaded view with the panels and variables you have on screen right now. Only enabled if a view is currently loaded (i.e. you have something to overwrite).
- **Save As…** — prompts for a new view name and creates `<your-username>/<that-name>`. The new view becomes the "current view."

Current view name

Once a view is loaded, its name is shown next to the views icon. If you modify panels without saving, the name stays the same — there's no "modified" indicator yet, so be deliberate about hitting **Save** when you mean to.

Username, password, logout

The far right of the header shows your logged-in username, a **Password** button, and a **Logout** button that clears the JWT and returns you to the login page. Both buttons are hidden in guest view mode.

Changing your password

Click **Password** to open the change-password dialog. It asks for your current password and the new one twice.

Rules enforced by the server, reported inline if you trip them:

- at least 8 characters by default (the server's configured minimum is shown in the field label — an administrator can change or disable it);
- must differ from your current password;
- must not be your username, and must not contain a colon;
- must not be one of the well-known Synchro default passwords.

Your session stays valid after a change — you are not logged out.

The cleartext password warning

Before SynchroServer 2.7.9, passwords were stored in cleartext in the server's login file. If your account has not been converted yet, two things appear:

- the header button reads ⚠️ **Password** instead of **Password**;
- an amber banner sits below the header: *"Your password is still stored in cleartext in the login file. Changing it now converts it to a salted hash."*

Changing your password clears both, because the new value is stored hashed. The banner can be dismissed with ? for the current session, and the warning reflects storage only — it does not mean your password is known to anyone.

Administrators can convert every account at once without anyone changing their password; see `PasswordManagement-<version>.md`.

The alarm status strip

Right under the header, SynchroUI shows a horizontal strip with four counters:

```
[ OK 142 ]   [ WARNING 7 ]   [ CRITICAL 2 ]   [ FATAL 0 ]
```

Each badge:

- Counts how many series across the entire SynchroServer are currently in that alarm level.
- Is color-coded: **green** / **yellow** / **orange** / **red**.
- Refreshes every **15 seconds**.

Click a badge to expand it inline and see the list of series names that are currently in that level. Click again to collapse.

The strip is hidden in guest view mode.

The variables bar

If a view defines **variables**, they appear as dropdowns just above the panels:

```
host: [affa5 ▾]   region: [na ▾]   ⚙
```

A **variable** is a named placeholder you can substitute into series selections. For example a panel selection of `{component: "$host", type: "HOST", metric: "CPU", target: "HOST"}` will re-query whenever you change the `host` dropdown — no need to edit each panel.

Each variable has: - A **label** shown to the user (falls back to the variable name). - A **field to populate from**: `component`, `type`, `metric`, or `target`. - Up to four **regex filters** that narrow the candidate values that show up in the dropdown.

The dropdown values are auto-discovered by querying the SynchroAPI index endpoint with those filters; you don't enter them by hand.

Click the ⚙ **gear icon** to open the **Manage Variables modal**. See Variables below.

The variables bar is hidden in guest view mode.

Panels

Each **panel** is one self-contained mini-dashboard: it has its own series selection, time range, refresh interval, period filter, and active display tab. A view is just a list of panels (plus the variables they share).

Panel header

```
[?]  My CPU Panel        ●     12 series   [+] [↓ ▾]  ×
```

- **☰ Hamburger** — opens the panel **overlay** with all the tabs (where you edit selections and switch what's displayed). See Panel tabs.
- **Title** — click to rename. Press **Enter** to save, **Esc** to cancel.
- **● Status dot** — color tells you the data state:
  - **green** — data complete
  - **yellow** — query loading
  - **red** — query returned no data
- **Series count** — how many time series the current selection resolves to.
- **+ Duplicate** — clones the panel into a new one right next to it.
- **↓ Export ▾** — download the current panel state. See Exporting data.

- ⍰ **Remove** — deletes the panel from the view (with no undo — re-create it if you change your mind).

Panel size

Each panel can be sized at **25 %, 33 %, 50 %, 66 %, 75 %, or 100 %** of the window width. Panels at smaller sizes wrap into multiple rows automatically. Set the size from the **Search tab** of the panel overlay.

Reordering panels

Drag a panel by its header to reorder it. The drop position is reflected in real time.

Empty panels

A brand-new panel shows the message **"Query result is empty"** until you go into the panel overlay's **Search tab** and pick at least one series.

Panel tabs in detail

Click the ☰ on any panel to open the **panel overlay**. At the top of the overlay is a tab row with all 10 tabs. Apart from **Search**, every tab is a way of *displaying* the same selected series — switching tabs never changes what data is selected.

Tab	What it shows
Search	Pick the series, time range, refresh, periods, functions, panel size, Y-axis scale, sort order
Time Series Graph	Interactive multi-line uPlot chart with optional alarm threshold zones
Heatmap	One color row per series — values normalized across all series
Histogram	Distribution of values for the selected series
Statistics	Filterable table with one row per (series × period), one column per function
Alarms	Current alarm level per (series × period), color-coded
Data	Raw time series records (most recent ~200)
H-Bar Graph	Horizontal bar chart of stats per period per series
V-Bar Graph	Vertical version of the same

Tab	What it shows
Card	One large value-card per series, background colored by alarm level

At the bottom of the overlay:

- **Apply** — commit your changes and close the overlay.
- **Cancel** *(or Esc)* — discard everything you changed in this overlay session and revert to the snapshot from when it opened.

Search tab

This is where you choose what the panel will look at and how. It's grouped into the following sections:

Selected series. A list of removable chips. Click a chip's ? to drop that series from the panel.

Add series. A search box that hits the SynchroAPI **index** endpoint. Type a few letters to narrow the list, then click a result to add it as a new chip.

Time range. Two ways to set it:

- **Presets** — *Last 5 min*, *30 min*, *1 hour*, *6 hours*, *24 hours*, *7 days*. Picking a preset puts the panel into **sliding window** mode: the range follows wall-clock time on every refresh.
- **Custom From/To** — one date input + one time input for each end. Editing any of them switches the panel to **manual** mode: the range stops sliding and stays exactly as you set it.

> 🔁 **Panel 0 sync.** When you change the time range or preset of the *first* panel, every other panel in the view picks up the same range. This is the convention for "synchronized investigation." If you want one panel independent, change *its* range first (so it isn't panel 0) — or just edit a non-first panel and it stays local.

Refresh interval (per panel). *Off, 5 s, 30 s, 1 m, 5 m*. The dashboard refresh override (in the header) takes precedence — see Time range and refresh.

Periods. A multi-select of the period names defined in SynchroServer (`00@24`, `08@17`, etc.). Leave it empty to query all periods; pick one or more to filter.

Functions. A multi-select of statistical functions: *DTime, Min, Max, Average, Sum, Count, Last*. Each function you select is rendered as its own line/bar/column in the display tabs.

Panel size. *25 %, 33 %, 50 %, 66 %, 75 %, 100 %.*

Y-axis scale. *Min* and *Max* numeric inputs. Leave both empty for auto-scaling.

Alarm zones toggle. *On / Off.* When on, the Time Series Graph overlays the alarm thresholds (OK / Warning / Critical / Fatal bands) behind the data lines so you can see at a glance whether the trace is in danger.

Sort by. *Selection Order, TS Name, DTime, Last, Min, Max, Average, Sum, Count*. Controls the order rows/lines appear in the display tabs.

Sort reverse. *Ascending / Descending* toggle.

Time Series Graph

The default display tab. Multi-line chart powered by **uPlot**.

- One line per `(series × selected function)` combination.
- Cursor tooltip shows the value at the hovered timestamp for every line.
- Alarm threshold bands behind the lines if **Alarm zones** is on in the Search tab.
- Two queries are issued in parallel: a **historical** query (full range, cached for 30 s) and a **live tail** query (last few seconds, refetched every refresh interval). They merge seamlessly so the chart updates smoothly without re-fetching the whole range every tick.

Click and drag on the chart to zoom; double-click to reset.

Heatmap

For comparing many series at once (think 50 hosts).

- One color row per series.
- X-axis is time, color intensity encodes the value.
- Values are normalized to the *(min, max)* across all series in the panel, so anomalies stand out instead of being drowned out by big absolute differences.

Histogram

The frequency distribution of values for the selected series. Useful for spotting bimodal load patterns or outliers.

Statistics

A spreadsheet-style table:

- One row per `(series × period)`.
- One column per function (DTime / Min / Max / Average / Sum / Count / Last).
- A **filter input** in every column header — type a regex to narrow rows.
- Row background colored to match the alarm level if a sister `*-ALARM` series exists.

Alarms

A focused table for the alarm tab specifically. Since SynchroUI 1.0.117 the **Status** column appears first, with rows colored by alarm level (green / yellow / orange / red). Columns:

```
Status | Date | Time | DTime | Period | Component | Type | Metric | Target |
Last
```

Status values: **OK**, **WARNING**, **CRITICAL**, **FATAL**.

The data comes from sister `*-ALARM` series (their **LAST** value encodes the level: 0 = OK, 1 = Warning, 2 = Critical, 3 = Fatal).

Data

Raw time series records — the **most recent ~200** entries for the panel's selection. Use this when you want to inspect actual numbers rather than trends. Columns: Date, Time, then one per selected function per series.

H-Bar / V-Bar Graph

Horizontal (H-Bar) or vertical (V-Bar) bar chart of statistics per period per series. Each function gets its own bar color (up to 10 colors); period filtering happens client-side so toggling periods is instant.

Card

A "wallboard" view: each series is rendered as one big card with the series name and its current value in extra-large font. The card background is colored to match the alarm level, so a screen full of cards reads like a status board.

Time range and refresh

A few rules worth memorizing:

1. **Per-panel time range** is set in each panel's Search tab.
2. **Panel 0's range syncs to all other panels** as soon as you change it. To opt a panel out, change its range *after* the panel-0 sync, not before.

3. **Manual ranges are persisted in the URL** as `from` / `to` query params, so you can copy the URL and share an exact frozen view.
4. **Preset (sliding) ranges are NOT persisted in the URL.** Reloading the page recomputes them against the *current* wall clock, which is exactly what you want for "always show the last hour" dashboards.
5. **Per-panel refresh** lives in the same Search tab.
6. **Dashboard refresh override** (in the header) overrides every per-panel setting. Use this to pause everything at once or force a uniform tick.

Views: save, load, share

A **View** is a snapshot of: - the list of panels (titles, selections, time ranges, sizes, active tabs, display options) - the list of variables and their current values - the sharing setting

It's stored on the server as `<username>/<viewname>` and saved as JSON in SynchroServer.

Loading a view

Three ways:

- **From the Views menu** — pick any row.
- **From a URL** — `?<viewname>` on the page URL loads that view at startup.
- **From localStorage** — if you don't pass `?view`, SynchroUI auto-loads the last view you had open in this browser (key `synchroui:lastView`).

Saving

- **Save** — overwrites the currently loaded view (the one whose name is shown next to the views icon) with whatever you have on screen.
- **Save As…** — prompts for a new name and creates `<you>/<thatname>`. After saving, that becomes the "current" view.

> ⚠️ There's currently **no "modified" indicator** when you change panels after loading a view. If you want to keep your changes, hit **Save** deliberately.

Sharing

Each view has a `sharing` field with three possible values:

- **Private** *(default)* — only you can see and load it.

- **Public** *(globe icon turns blue)* — anyone with a guest link can open it without logging in. Use the **Copy guest link** button to put a `?##<base64(viewname)>` URL on your clipboard.
- **Shared with specific users** *(set via the API today)* — visible to the named users in their Views menu, but not public.

Deleting

The trash icon in the views menu deletes the view from SynchroServer (your own views only). It asks for confirmation first; deletion is permanent.

Alarms

Alarms in SynchroServer are organized into **groups** (a name + a list of **alarm items**). Each item is a rule that matches some series and defines the OK / Warning / Critical / Fatal value bands plus optional timing controls. SynchroUI lets you manage all of this from the alarm bell dropdown without editing JSON by hand.

Opening the alarms manager

Click the 🔔 **alarm bell** in the header. A dropdown opens showing:

1. **Status row** — OK / Warning / Critical / Fatal counts (the same data as the alarm status strip).
2. **Global toggle** — *Alarms enabled / disabled.* Flipping this off pauses ALL alarm evaluation server-side immediately.
3. **Group list** — every alarm group, expandable.
4. **Add new group** — input + button at the bottom.

Working with groups

- Click a group row to expand it and see its rules.
- The **Delete group** button removes the group and every rule in it (with confirmation).
- The **Add group** input at the bottom of the dropdown creates a new empty group. Pick a name that's filesystem-safe — anything outside `[A-Za-z0-9._-]` is replaced by _ when SynchroServer writes the file. See "Storage" below.

Working with alarm rules

Inside a group:

- **Click a rule** to open the **Edit alarm modal**.
- The **Delete rule** button removes just that rule.

The edit-alarm modal

Every alarm rule has the following fields:

Field	Purpose
Description	Free-text label shown in the rule list.
Component / Type / Metric / Target	Regex filters defining which series this rule applies to. Default `.*`.
Date / Period	Regex filters scoping by date and period. Default `.*`.
Function	Which DQM function to evaluate the rule against — *Last, Average, Min, Max, DTime, Count, Sum*.
Raise criteria	*Both / Rise / Fall* — whether the rule fires on rising values, falling values, or both.
Raise level	The minimum severity the rule will report — *OK / Warning / Critical / Fatal*.
OKMin / OKMax	The "everything is fine" band.
WarnMin / WarnMax	The Warning band.
CriticalMin / CriticalMax	The Critical band.
FatalMin / FatalMax	The Fatal band.
Delay (s)	Wait this many seconds in the threshold-breached state before raising.
Snooze (s)	After raising, suppress new triggers for this many seconds.
Repeat (s)	After snooze, re-trigger every N seconds while still breached.
Exemption	Number of breaches to ignore before the rule starts counting.
UseByTime	If on, the rule respects period exemption windows.

Hit **Save** to commit, **Cancel** to discard.

Storage on the server

When you save the alarm dropdown, SynchroUI sends the whole alarm tree to `POST /api/v1/admin/alarms`. Since SynchroServer **2.7.8.048**, the server splits the request into **one file per group** under `conf/alarm-json.d/`:

```
conf/alarm-json.d/api-alarms-DEFAULT.json
conf/alarm-json.d/api-alarms-Production.json
conf/alarm-json.d/api-alarms-ExampleCPU.json
…
```

The server cleans up files for any group you remove, so the on-disk state always matches the dropdown. The legacy combined `api-alarms.json` (from older SynchroServer builds) is removed automatically the first time you save through this UI.

After every save the in-memory alarm engine is reloaded automatically — you never need to hit a separate "reload" button.

What does an alarm look like once it fires?

When SynchroServer evaluates a rule and finds a series in a non-OK band, it creates a **sister time series** with `-ALARM` appended to the type. For example, an alarm on `affa5/HOST/CPU/HOST` produces `affa5/HOST-ALARM/CPU/HOST` whose **Last** value is the level (0..3). That sister series is what the **Alarms tab** queries to color rows, and what the alarm status strip aggregates.

Releases and download links

SynchroServer can publish release tarballs (or in fact any file) from `data/releases/` via signed time-limited URLs. SynchroUI exposes this feature through the **Releases modal**, opened by the ⬇️ icon in the header.

> 🔒 The Releases modal is admin-only. The icon is hidden in guest view mode and only effective when you're logged in as an admin user.

What you can do in the modal

List release files. Files in `data/releases/` are listed with their name, size, and modification time. Hidden files and `.sha256` companion files are excluded.

Generate a signed download link.

1. Pick a file from the dropdown.
2. Optionally set:
   - **TTL** *(default 7 days)* — how long the link is valid. Range 1–365 days.

- **Max downloads** *(default 0 = unlimited)* — hard cap on the number of times the link can be used.
- **Label** — a free-form note for your own audit log (e.g. *"Customer A — Apr release"*).

3. Click **Generate**.

The modal returns a complete download URL — just the path `/api/v1/download/<filename>?token=<jwt>` against your SynchroServer host. **Copy** it and send it to whoever needs the file. The recipient does NOT need a SynchroUI account — the token gates access on its own.

List active links. Every link you've issued is listed with its filename, creation time, expiry, max-downloads cap, current download count, the label you gave it, and a *revoked* flag. You can expand a link to see its **audit log** (timestamp + remote IP for every successful download).

Revoke a link. The trash button on a link sets it to *revoked*. The link's record is kept (so the audit history is preserved), but any further download attempt with that token returns HTTP 403.

How clients use the link

```
GET /api/v1/download/<filename>?token=<jwt>
```

This is a **public** endpoint (no `Authorization` header), but it validates the token, checks that it hasn't been revoked or hit `max_downloads`, then streams the file with a `Content-Disposition: attachment; filename="..."` header. The response also includes `X-Checksum-Sha256` and an RFC 3230 `Digest: sha-256=<base64>` header so the recipient can verify integrity.

Variables

A **variable** is a placeholder you can substitute into series selections. Once defined, a variable shows up as a dropdown in the Variables bar above the panels. Changing it re-queries every panel that references it — without you editing each panel by hand.

The Manage Variables modal

Click the ⚙ in the variables bar.

You see a list of every variable defined for the current view. For each one:

Field	Purpose
Name	Identifier used in selections (`alphanumeric_underscore`). Reference it in selections as `$name`.
Label	Human label shown above the dropdown in the variables bar. Optional — falls back to **Name**.
Populate from	Which axis to query: *component, type, metric,* or *target*.
Filter — Component / Type / Metric / Target	Four regex filters that narrow what shows up in the dropdown.
Current value	Whatever you've selected in the dropdown right now.

Buttons:

- **Add Variable** — appends a new empty row.
- **Remove** — drops a variable from the view (panel selections that referenced it become literal `$name` strings until you fix them).
- **Save** — commits all changes. Variables and their current values are stored as part of the view, so they persist when you save the view.
- **Cancel** — discards everything you changed in this modal session.

How substitution works

If a panel selection has `component: "$host"` and the `host` variable is currently set to `affa5`, the actual API query goes out as `component=affa5`. Substitution happens at query time, so if a user changes the dropdown, every dependent panel re-queries on the next tick (or immediately if they hit *Refresh now*).

Variable values in the URL

The variables bar mirrors current values into the URL as `var-<name>=<value>` parameters. That means:

- Refresh the page → values are restored.
- Copy the URL to share → recipient sees the same values.
- Override one value via URL → useful for "view this same dashboard but scoped to host X" deep links.

Guest view mode

Guest view mode is how you share a dashboard with someone who **doesn't have a SynchroServer login**.

How a guest link is generated. In the Views menu, mark a view as *public* (globe icon) and click **Copy guest link**. The clipboard now has a URL of the form:

```
https://<your-host>/SynchroUI/?##<base64(viewname)>
```

The `##` after the `?` is the marker SynchroUI uses to recognize a guest load. The base64 portion is the view name (so the recipient never sees the raw `username/viewname` string).

What a guest can do. Open the link → SynchroUI loads the public view in **read-only mode**. The guest can:

- See all the panels and their data.
- Switch panel tabs to look at the same series in different views.
- Scrub the time range *visually* on the chart (drag-zoom).
- Watch the data update on the panels' refresh schedule.

What a guest CANNOT do.

- Log in or see the username/logout area.
- Save, save-as, or delete views.
- Edit panels (no Search tab access).
- Add or remove panels.
- Change variables (the variables bar is hidden).
- Manage alarms (the bell is hidden).
- Open the Releases modal (the icon is hidden).
- Pause or change refresh from the header.

The current view name appears in the header where the views menu would be, so the guest at least knows what they're looking at.

> 🔒 **Public is public.** A guest link is unauthenticated. Anyone who has the link can view the data. Treat the marker `?##...` as a "view this is on the open internet" decision, not a secret.

Deep links and persistence

SynchroUI keeps a surprising amount of state in the URL so links are shareable and refreshable.

URL fragment	Meaning
`?<viewname>`	Load this view at startup.
`?##<base64(viewname)>`	Load this view in guest mode.
`?from=<unix>&to=<unix>`	Override the panels' time range to a manual window. Only meaningful for views with at least one manual-mode panel.
`?var-<name>=<value>`	Override the value of variable `<name>`.

Multiple parameters combine — you can deep-link to "this view, this time range, with `host=affa5` and `region=na`" in one URL.

localStorage is used as a fallback for "the last thing you were looking at":

Key	Purpose
`synchroui:lastView`	Name of the last view you loaded. Auto-loaded on next page open if no `?viewname` is in the URL.

There's no other client-side state of note. Clearing your browser cache or cookies signs you out (the JWT lives in `sessionStorage`).

Exporting data

The ↓ **export menu** in each panel header offers four formats:

Format	Notes
PNG	Snapshot of the current Time Series Graph (only works when the Graph tab is the active display). Captures the uPlot canvas as-is, including alarm zones if enabled.
JSON	Full panel data — both the time series points and the statistics — as a single JSON document.

Format	Notes
CSV	Tabular export of the same data as comma-separated values. Suitable for Excel / Google Sheets.
TSV	Same as CSV but tab-separated. Often handier for command-line tools.

The file is downloaded directly via the browser; nothing is uploaded to SynchroServer.

Troubleshooting

The login page rejects my correct credentials. - Check the version popup on the logo. If `SynchroServer:` shows `unknown`, the UI cannot reach `GET /api/v1/health` — your reverse proxy or firewall is blocking the API path. Fix the proxy and reload.

Every panel shows "Query result is empty" right after load. - Open the Search tab on one panel and confirm series are selected. Views saved against an old SynchroServer may reference series that no longer exist.

The alarm strip says 0 / 0 / 0 / 0 even though I know there are alarms. - Click the alarm bell and check the **Alarms enabled** toggle at the top. If it's off, the server-side alarm engine is paused. - If it's on, check that there's at least one **alarm group with at least one rule**. An empty alarm config evaluates to "everything OK." - Check the version popup — if SynchroServer is older than `2.7.8.048`, alarms saved through the UI may not have been wired up correctly. Upgrade.

A view loads with no panels. - The view is corrupt or was saved from a much older UI. Delete it from the views menu and re-create it.

My guest link returns "view is not public." - The owner toggled the view back to private, or copied the link from a view that was never marked public. Have them re-publish via the globe icon and re-send the link.

The "Refresh now" button does nothing visible. - Per-panel refresh might still be running its own queries with cached results. Bump the panel's refresh interval down to *5 s* in the Search tab, or set the dashboard refresh dropdown to **5 s** to force a uniform re-fetch.

The download link I just generated returns 403. - Check the link in the Releases modal — it may have hit its **max downloads** cap, or you (or another admin) may have **revoked** it. Generate a new link.

The page is blank after I update SynchroUI. - Hard-refresh the browser (`Cmd+Shift+R` on macOS, `Ctrl+F5` on Windows/Linux). The previous build's JavaScript may be cached.

Events tab

The **Events** panel tab is a real-time dashboard of the day's *scheduled events* — reports, application start/stop, phase changes and anything else you define through the Events API. Unlike the other tabs it does not use the panel's series selections; it shows every event from `GET /api/v1/events/state`, refreshing on the panel's refresh interval.

One row per event, with these columns: Name, Type, Status, Planned, Delay, Late, Est. Compl., To Alarm, Level, Comments. Times are shown as `HH:MM:SS`; delays/lateness as durations.

Row color follows the alarm level:

Color	Meaning
Grey	Not started yet and not late (nothing to worry about)
Green	OK (started / completed on time)
Yellow	Warning — late past the warning threshold
Orange	Critical
Red	Fatal

Inactive events are dimmed. **Click any row** to open its details panel: all timing/calculated fields, comments, and today's last metric values for that event.

View modes (buttons at the top of the tab) control which events are shown and how they are ordered:

- **Time** *(default)* — the next event expected to complete is first, then the remaining events ordered by estimated completion, then planned/start time.
- **Scheduled** — active events planned for today.
- **Planned** — all events planned for today, **including inactive** ones.
- **All** — every defined event, regardless of day.

To add an Events dashboard, create a panel and switch its tab to **Events** (the tab selector lists it alongside Graph, Stats, Alarms, Card, …).

Where to go next

- **REST API reference:** SynchroAPI-2.7.9.055.md
- **Server installation and operations:** see `install.sh` inside the release tarball at `live5:/home/synchro/SynchroServer/data/releases/SynchroServer-2.7.9.051.tar.gz`.