



# ParentSync

Your family's WhatsApp and email messages, automatically  
organized into calendar events

v1.0.2 · Desktop App for Linux, Windows & macOS

Created by [Shahar Bar-Moshe](#)


# What is ParentSync?

A desktop app that reads your children's school WhatsApp groups and teacher emails, uses AI to find events, and puts them on your Google Calendar.

- Monitors WhatsApp parent groups and Gmail automatically
- AI-powered parsing extracts dates, times, and event details
- Events sync directly to your family Google Calendar
- Optional approval workflow via WhatsApp reactions (👍 approve / 😞 reject)
- All data stored locally on your machine — no cloud, no sign-up

# Get the App


Click your platform for a direct download from GitHub Releases v1.0.2. Per-platform run instructions later in the deck (Quick Start section).



**macOS**


↓ Apple Silicon arm64.dmg

↓ Intel .dmg



**Windows 10/11**

↓ Installer Setup-1.0.2.exe



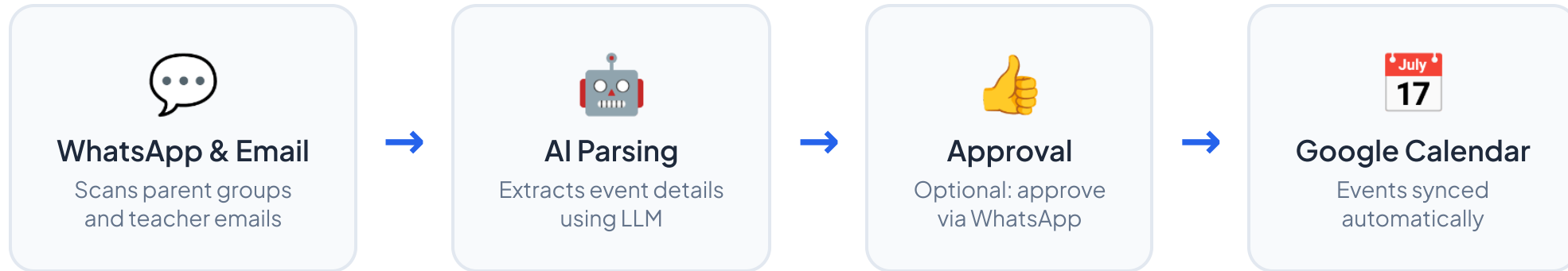
**Linux**

↓ Portable .ApImage

↓ Debian/Ubuntu .deb

No account needed. All assets: [github.com/ShaharBarMoshe/ParentSync/releases/tag/v1.0.2](https://github.com/ShaharBarMoshe/ParentSync/releases/tag/v1.0.2)

# How It Works



# Dashboard

The dashboard provides a comprehensive overview of the user's sync status and history. It includes a navigation bar with 'ParentSync', 'Dashboard', 'Calendar', 'Monitor', and 'Settings'. The main content area is divided into several sections:

- Dashboard Overview:** A summary of messages, events, and sync status, with a 'Sync Now' button.
- SYNC STATUS:** A table showing the current sync status, including the number of messages processed, events created, the last sync time, duration, and status.
- SYNC HISTORY:** A list of previous sync attempts, including the date, time, duration, and number of messages.
- RECENT MESSAGES:** A list of the most recent messages received.
- UPCOMING EVENTS (7 DAYS):** A list of events scheduled for the next seven days.

MESSAGES PROCESSED	EVENTS CREATED	LAST SYNC	DURATION	STATUS
1004	0	30/04/2026, 22:00:12	12s	Success

STATUS	DATE/TIME	DURATION	MESSAGES
Success	30/04/2026, 22:00:00	12s	37 msgs
Success	30/04/2026, 21:07:03	15s	242 msgs
Success	30/04/2026, 21:03:00	14s	242 msgs
Success	30/04/2026, 21:00:00	14s	242 msgs
Success	30/04/2026, 20:00:00	13s	241 msgs

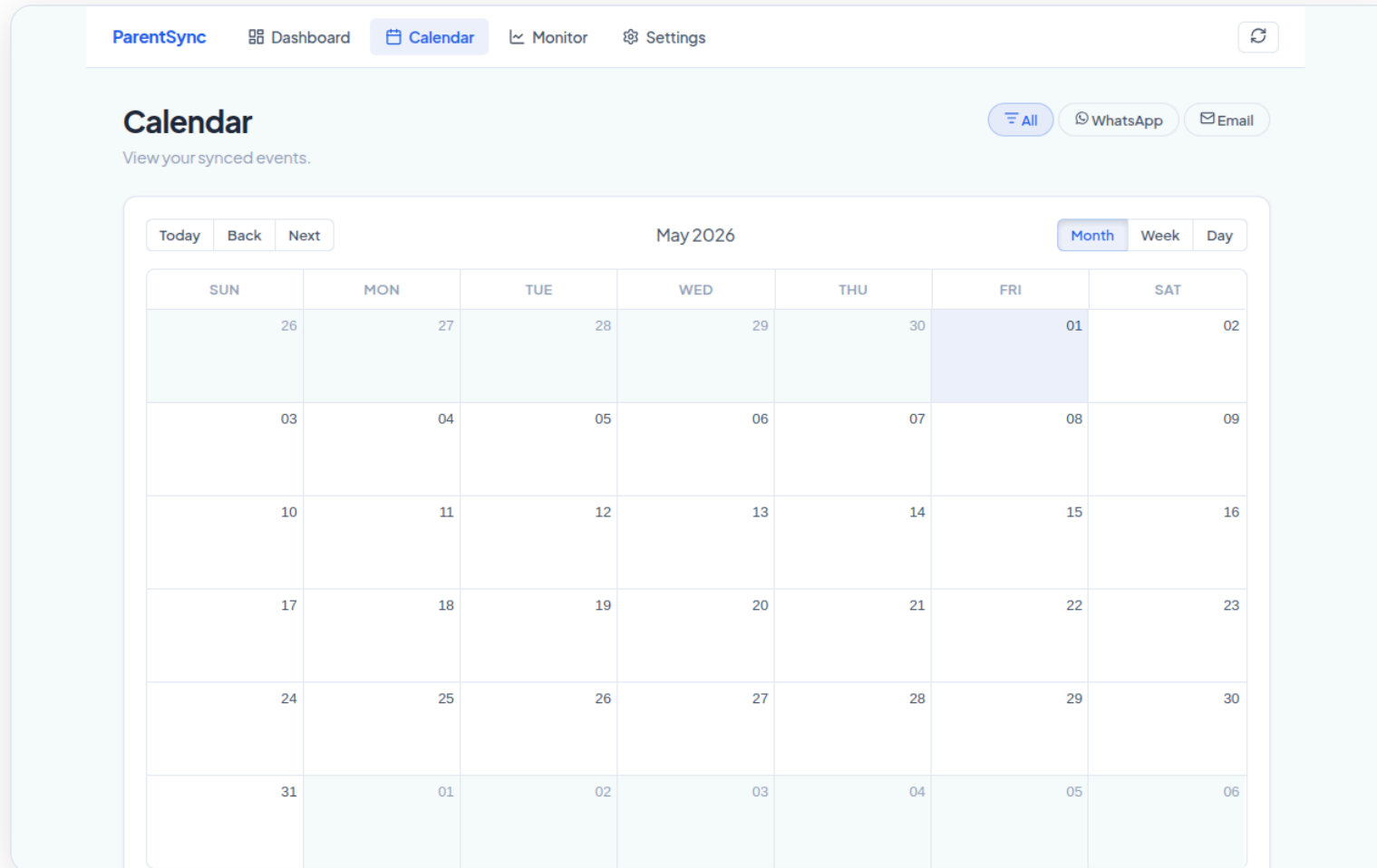
MESSAGE	DATE
כיתהד' 2 תשפ"ו	30/04/2026

EVENT	STATUS
אג: יום הולדת	Pending

Sync status, recent messages, upcoming events, and sync history at a glance

# Calendar View



Full month/week/day calendar with events color-coded by source (WhatsApp or Email)

# Settings & Setup

## Connections

The screenshot shows the 'Connections' settings page in ParentSync. The navigation bar includes 'ParentSync', 'Dashboard', 'Calendar', 'Monitor', and 'Settings'. The main heading is 'Settings' with a subtitle 'Configure your children, sync schedule, and integrations.' There are two main sections: 'WHATSAPP' and 'GOOGLE ACCOUNTS'. Under 'WHATSAPP', there is a 'WhatsApp Web' card showing 'Connected' status and a 'Manage Connection' button. Under 'GOOGLE ACCOUNTS', there are two cards: 'Email Scanning' with email 'shbmosh@gmail.com' and 'Calendar' with email 'bar.moshe.family@gmail.com', both showing 'Connected' status and 'Disconnect' buttons.

## Children & Schedule

The screenshot shows the 'Children & Schedule' settings page in ParentSync. The navigation bar includes 'ParentSync', 'Dashboard', 'Calendar', 'Monitor', and 'Settings'. At the top, there is a time selection interface with buttons for 4p, 5p, 6p, 7p, 8p, 9p, 10p, and 11p, and a summary 'All None 16 hours selected'. Below this is the 'EVENT APPROVAL' section with an 'Approval Channel' input field containing 'אישור אירועים'. The 'AI EXTRACTION PROMPT' section contains a text area with the following content: 'You are a calendar event extractor. Extract calendar events from messages, which may be WhatsApp group chat logs or emails. CRITICAL RULES: - ONLY extract information that is EXPLICITLY stated in the message text. NEVER invent, guess, or hallucinate details (names, times, locations, descriptions) that are not present in the text. - If a detail is not mentioned, DO NOT include that field. Leave optional fields out entirely rather than guessing. - For a single message: return a JSON array of events. If no actionable event is detected, return an empty array: []. - For multiple numbered messages: return a JSON object where each'.

# Dashboard — In Detail

## Sync Status (top card)

- **Messages Processed** — total messages collected across all recent syncs
- **Events Created** — how many calendar events the AI extracted
- **Last Sync** — when the last sync happened
- **Duration** — how long it took (e.g., 18s)
- **Status badge** — green "Success", orange "Partial", or red "Failed"

💡 Click "**Sync Now**" to trigger an immediate scan of all your configured WhatsApp groups and email addresses.

## Sync History (expandable list)




- Shows the **last 5 syncs** with status, time, duration, message count
- **Click any entry** to expand — see per-channel breakdown (which child, which group, how many messages)
- Skipped channels show the **reason** (e.g., "no new messages")
- Click "**View synced messages**" to see exact messages that were collected

# Dashboard — Messages & Events

## Recent Messages (bottom left)

- All messages collected from WhatsApp and Gmail
- Source icon: **WhatsApp icon** = WhatsApp group, **Mail icon** = email
- **Click a message** to expand and read full content
- Click "**Show more**" to load additional messages (20 at a time)

## Upcoming Events (bottom right)

- Events in the **next 7 days** parsed from messages
- Coloured pill:  **Pending**,  **Approved**,  **Rejected**
- Pending events have inline **Approve / Reject** buttons — same effect as  /  in WhatsApp
- **Checkmark icon** = synced to Google Calendar

## How to get data flowing

- 1 Make sure **WhatsApp is connected** (Settings → WhatsApp → "Connected")
- 2 Add at least one **child with WhatsApp channels** (Settings → Children)
- 3 Set your **Gemini API key** (Settings → Gemini AI), or use OpenRouter as alternative
- 4 Click "**Sync Now**" on the Dashboard
- 5 Wait 15–30 seconds — messages and events appear

 **Refresh button** (top-right ↻ icon) reloads all data on the current page without triggering a new sync.

# Getting & Refreshing Data


## Automatic Sync

ParentSync automatically scans at the hours you configured in Settings → Sync Schedule. For example, if you selected 8am–10pm, it scans once per hour during those times.


## Manual Sync

Click "**Sync Now**" on the Dashboard to trigger an immediate scan. Useful after adding a new child or channel.

## Refresh Display

The  **refresh button** (top-right corner of the nav bar) reloads all data on the current page from the database. It does NOT trigger a new sync — it just refreshes the view.

## Data Freshness Tips

- 1 After connecting WhatsApp for the first time, click "**Sync Now**" to populate initial data
- 2 Add all your children's group names **before** the first sync so all channels are scanned
- 3 The Monitor page filters (7d/30d/90d) affect all charts — use **30 days** for a good overview
- 4 If Dashboard shows old data, click  **refresh** — if it still doesn't update, click "**Sync Now**"

 The app keeps running and syncing even when minimized to the system tray.



# Setup: Gemini (default AI)

The AI engine that reads messages and extracts calendar events

## What is Gemini?

Google's Gemini family of LLMs — fast, cheap, and very good at multilingual extraction. ParentSync uses it to read WhatsApp and email messages and identify dates, events, and details.

💡 The default model `gemini-2.5-flash-lite` is fast and cheap — typical extraction costs are negligible. A free tier is available too.

🔄 Don't want Gemini? **OpenRouter** is supported as an alternative provider — see next slide.

## How to get your API key

- 1 Go to [aistudio.google.com/apikey](https://aistudio.google.com/apikey)
- 2 Sign in with your Google account
- 3 Click "**Create API key**"
- 4 Copy the key
- 5 Paste in ParentSync → Settings → Gemini AI → API Key
- 6 Optionally pick a different model in the same section

📺 **Video walkthrough** — [Gemini API Key in Google AI Studio \(YouTube\)](#)

# Alternative: OpenRouter

One-stop shop for many LLM providers behind a single API key


## What is OpenRouter?

A gateway to many AI models (GPT, Claude, Llama, Qwen, etc.) through a single API key. Useful if you already have credits there, or if you want to A/B test models without changing keys.

💡 Free models are available — typically labelled : f ree. They have rate limits but cost nothing.

## How to get your API key

- 1 Go to [openrouter.ai/keys](https://openrouter.ai/keys)
- 2 Sign up or log in (Google/GitHub)
- 3 Click "**Create Key**"
- 4 Copy the key (starts with sk - or -)
- 5 Paste in ParentSync → Settings → OpenRouter → API Key

 **Video walkthrough** — [Get your OpenRouter API Key \(YouTube\)](#)

You can fill both keys; the active provider is whichever the build is wired to (Gemini by default).

# Setup: Google Account

Connect Gmail and Google Calendar to read emails and sync events

## What is Google OAuth?

OAuth lets ParentSync access your Gmail and Calendar **without storing your Google password**. You create a "Client ID" in Google Cloud Console — think of it as a permission slip that tells Google "this app is allowed to access my data."

 You can use **separate Google accounts** for email scanning and calendar — or the same account for both.

## How to set it up

- 1** Go to [console.cloud.google.com](https://console.cloud.google.com)  
Create a project if you don't have one
- 2** Enable **Gmail API** and **Calendar API**  
APIs & Services → Enable APIs
- 3** Create **OAuth consent screen**  
Add your email as a Test user
- 4** Create **OAuth Client ID (Web app)**  
Redirect URI: `http://localhost:41932/api/auth/google/callback`
- 5** Copy **Client ID** and **Client Secret** into ParentSync Settings

 **Tip:** switch the consent screen to "In production" — Test-mode tokens expire after 7 days.



**Video  
walkthrough**

— [Create OAuth 2.0 Client ID in Google Console \(YouTube\)](#)

# Connecting Google Accounts

## After entering Client ID & Secret

- 1 Click "**Save Settings**"
- 2 Click "**Sign in with Google**" under Email Scanning
- 3 Google's consent screen opens in your browser
- 4 Select your account and click "**Allow**"
- 5 You're redirected back — status shows **Connected**
- 6 Repeat for **Calendar** (same or different account)

## Troubleshooting

### ⚠ "Error 403: access\_denied"

Your Google app is in "Testing" mode. Fix: Go to OAuth consent screen → either **add your email as a Test user** or click "**Publish App**".

### ⚠ "Failed to exchange authorization code"


The Client Secret may be wrong, or the redirect URI doesn't match. Double-check both in Google Cloud Console and in ParentSync Settings.


# Setup: WhatsApp

Connect your WhatsApp to scan parent group messages

## How it works

ParentSync connects to WhatsApp Web — the same way you'd use WhatsApp on your computer. It runs its own browser in the background to read messages from the groups you specify.

 Your WhatsApp session **persists across app restarts**. You only need to scan the QR code once. If the session expires, you'll be prompted to re-scan.

 Your phone must stay connected to the internet for WhatsApp Web to work (WhatsApp requirement).

## How to connect

- 1 Open ParentSync → **Settings**
- 2 Click "**Connect WhatsApp**"
- 3 A **QR code** appears in the app
- 4 On your phone:  
WhatsApp → Settings → Linked Devices → Link a Device
- 5 Scan the QR code with your phone camera
- 6 Status changes to "**Connected**"

Then go to **Children** in Settings, add a child, and type the exact WhatsApp group name.

# Under the Hood

## DESKTOP SHELL

Electron (single executable)

## BACKEND

NestJS (REST API, embedded)

## AI

Gemini (default) · OpenRouter swap-in

## FRONTEND

React + TypeScript

## DATABASE

SQLite (local, no server)

## WHATSAPP

whatsapp-web.js (Chromium)

All data stored locally on your machine. No cloud servers, no accounts to create. Just download, configure, and run.

# Run on macOS

↓ Download — Apple Silicon ParentSync-1.0.2-arm64.dmg

↓ Download — Intel ParentSync-1.0.2.dmg

## Step by step

- 1 Click the right download above (M-series Mac → Apple Silicon; older Intel Mac → Intel)
- 2 **Double-click** the .dmg in your Downloads folder
- 3 **Drag** the ParentSync icon into the **Applications** folder
- 4 Open **Applications** and double-click ParentSync
- 5 First time only: macOS says "Apple cannot check it for malicious software."  
**Right-click** the icon → **Open** → **Open** in the dialog

## What you'll see

- The app window opens — click **Settings** (top bar) to start configuring
- An icon appears in your menu bar (top right) — right-click for Sync Now / Quit
- Closing the window keeps it running; quit fully from the menu bar

 The build is unsigned (no Apple Developer certificate). The right-click → Open dance is only needed the first time.

# Run on Windows

 **Download Installer** ParentSync-Setup-1.0.2.exe

## Step by step

- 1 Click the download button above
- 2 **Double-click** the installer in your Downloads folder
- 3 Windows SmartScreen says "Windows protected your PC."  
Click "**More info**" → "**Run anyway**"
- 4 Click through the installer (Next → Install → Finish)
- 5 The app launches automatically and adds itself to the Start menu

## What you'll see

- A desktop shortcut and a Start-menu entry are created
- An icon appears in the system tray (bottom right) — right-click for Sync Now / Quit
- Closing the window minimizes to the tray; right-click → Quit to exit fully
- Auto-starts on login by default

 The installer is unsigned (no Authenticode certificate). The SmartScreen "Run anyway" is only needed the first time.

# Run on Linux

↓ **AppImage (any distro)** ParentSync-1.0.2.AppImage

↓ **.deb (Debian/Ubuntu)** parentsync\_1.0.2\_amd64.deb

## AppImage (any distro)

- 1 Click the AppImage download above
- 2 Files need to be marked runnable first. Two ways:  
**Mouse:** Right-click → **Properties** → **Permissions** → check "**Allow executing as a program**".  
**Terminal:** `chmod +x ParentSync-1.0.2.AppImage`
- 3 **Double-click** it (or `./ParentSync-1.0.2.AppImage`)

## Debian / Ubuntu: .deb

- 1 Click the .deb download above
- 2 Double-click → opens in Software Center → **Install**
- 3 Or terminal: `sudo dpkg -i parentsync_1.0.2_amd64.deb`
- 4 Launch from your application menu

## Auto-start on login (optional)

Clone the repo and run `npm run install:local` — registers the AppImage as a systemd user service.

# Platform Support & Building From Source

Platform	Format	How to Run
<b>Linux</b>	.ApplImage (portable)	chmod +x, double-click or run from terminal
<b>Linux</b>	.deb (Debian/Ubuntu)	dpkg -i, run from app menu
Windows	.exe (NSIS installer)	Double-click installer
macOS	.dmg	Drag to Applications

```
# Build it yourself (any platform)
git clone https://github.com/ShaharBarMoshe/ParentSync.git
cd ParentSync && ./setup.sh
npm run package:linux # or :mac / :win
```

Built with Electron | All data stored locally | Private use

# Uninstalling

An in-app **Settings** → **Danger Zone** → **Uninstall ParentSync** button is planned ([Phase 19](#)) — until then, here's the manual cleanup per platform. Add `--purge` / the data-removal step to also wipe your database, OAuth tokens, WhatsApp session, and logs.

## Linux

```
# 1. systemd autostart
systemctl --user stop parentsync.service
systemctl --user disable parentsync.service
rm -f
~/config/systemd/user/parentsync.service

# 2. binary + versions
rm -f ~/.local/bin/ParentSync.AppImage
rm -rf ~/.local/share/parentsync
rm -f
~/local/share/applications/parentsync.desktop

# 3. (.deb only)
sudo apt remove parentsync

# 4. data – IRREVERSIBLE
rm -rf ~/.config/parentsync
```

## macOS

```
# 1. Quit ParentSync (menu bar)
# 2. Drag /Applications/ParentSync.app to Trash

# 3. data – IRREVERSIBLE
rm -rf "$HOME/Library/Application Support/ParentSync"
rm -rf "$HOME/Library/Logs/ParentSync"
rm -rf "$HOME/Library/Caches/com.parentsync.app"
rm -f
"$HOME/Library/Preferences/com.parentsync.app.plist"
rm -f
"$HOME/Library/LaunchAgents/com.parentsync.app.plist"
```

## Windows

```
# 1. Settings → Apps → ParentSync → Uninstall
# (Handles binary, Start menu, registry)

# 2. data – IRREVERSIBLE (PowerShell)
Remove-Item -Recurse -Force "$env:APPDATA\ParentSync"
Remove-Item -Recurse -Force
"$env:LOCALAPPDATA\ParentSync"
Remove-ItemProperty
"HKCU:\Software\Microsoft\Windows\CurrentVersion\Run"
"ParentSync"
```

 After removing user data, open WhatsApp on your phone → **Settings** → **Linked Devices** and remove the ParentSync entry to fully unlink.

# Event Reminders

ParentSync sends a WhatsApp reminder ~24 hours before an event begins, but only for events that have settled in the system long enough to be trusted.

- Hourly cron scans the calendar for upcoming events
- Only events added more than **24 hours ago** are eligible — protects against last-minute parsing mistakes
- Only events already **synced to Google Calendar** qualify
- Event existence is **verified against Google Calendar** before sending — deleted events are silently skipped
- Reminder is delivered to the configured approval/notification WhatsApp chat with full details: title, date, time, location, description, and source channel
- Each event is reminded at most once (reminderSent flag)

See docs/`EVENT-REMINDERS.md` for the full specification.

# You're in control of the AI 🧠

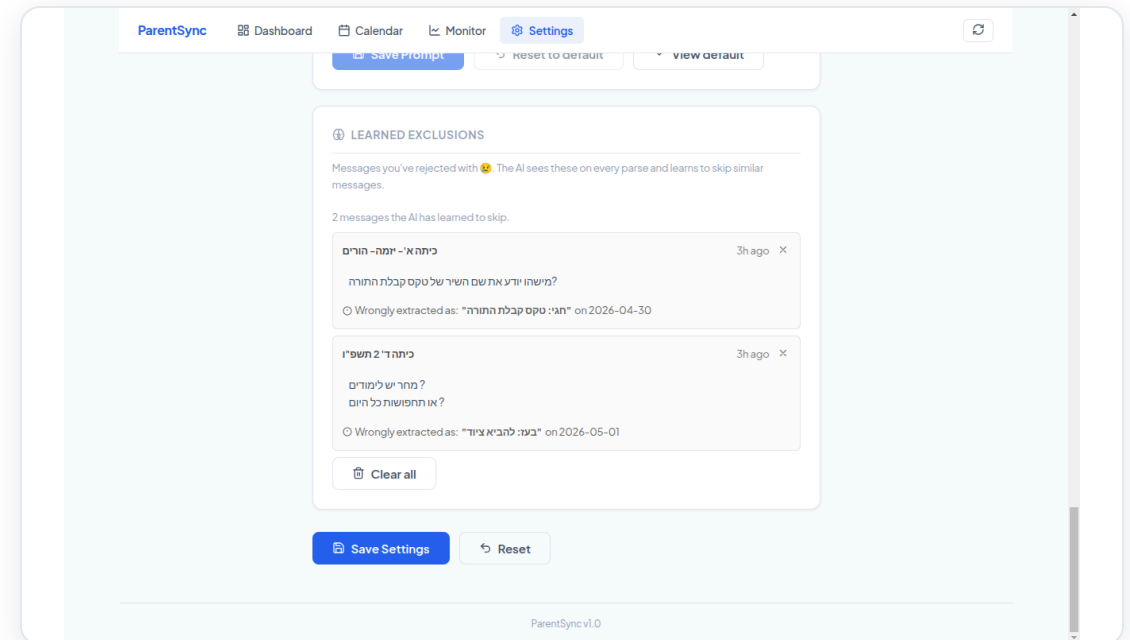
The AI never gets the last word — **you do**, in two complementary ways. The prompt sets the rules; your reactions tune the edge cases. Both feed into the same LLM call, and changes take effect on the next sync.

## 1. Edit the rules



**Settings → AI Extraction Prompt.** The whole system prompt is a textarea. Add an example, change wording, or hit *Reset to default*. Effective on the next sync.

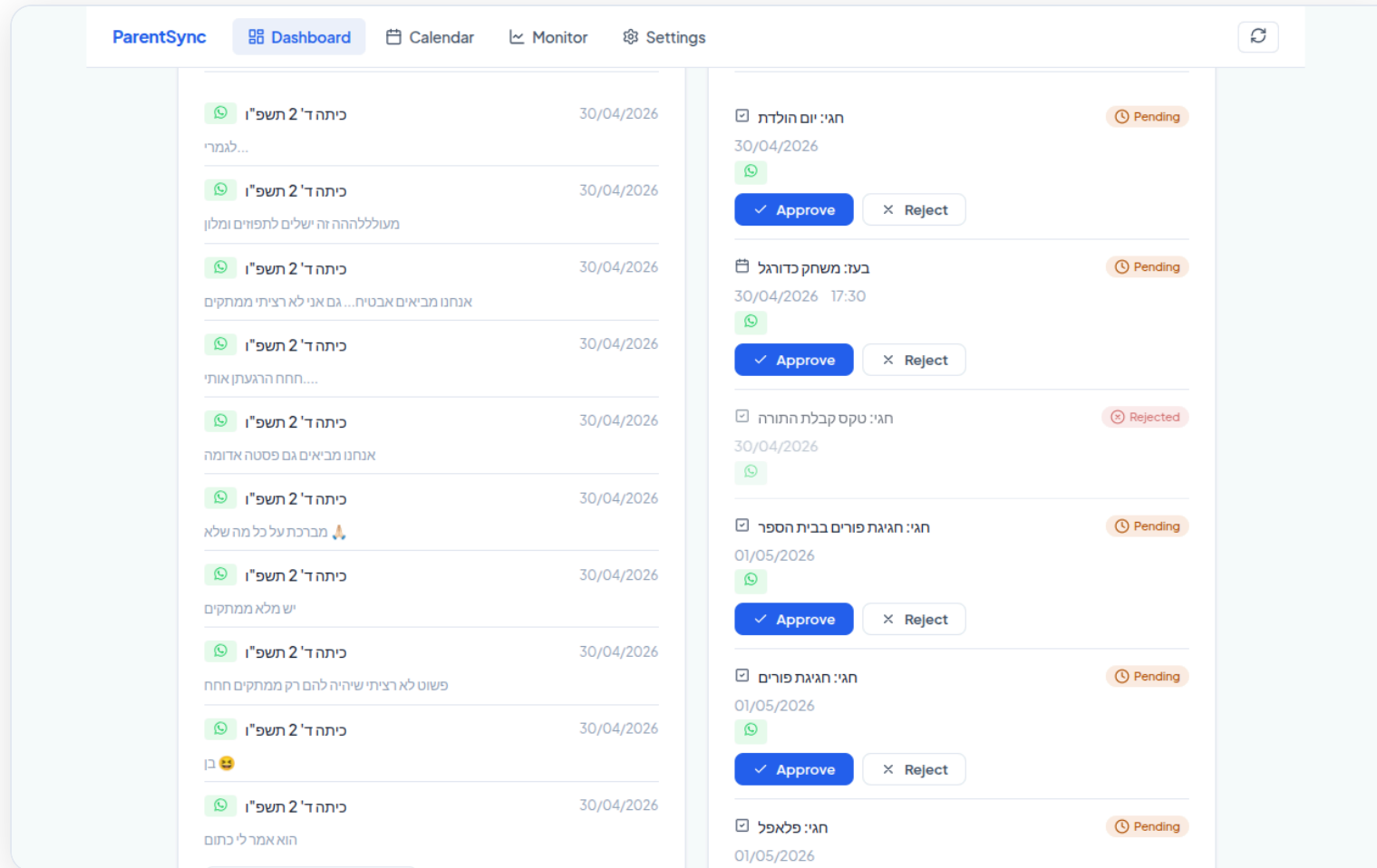
## 2. React to the results



👍 / 🙅 in WhatsApp — or the inline buttons on the Dashboard. Every 🙅 captures the source as a *Learned Exclusion* (most recent 50 appended to the prompt). Take back a reaction to undo.

The parse cache keys include a hash of the prompt + exclusions — so feedback closes on the **next** sync, not 24 h later.

# In-App Approval



Each upcoming event shows its status as a pill —  Pending,  Approved,  Rejected. Pending events have inline Approve / Reject buttons. Same effect as 👍 / 🙄 in WhatsApp; reactions are reversible too — take back a 👍 to unsync from Google, take back a 🙄 to clear the learned exclusion.

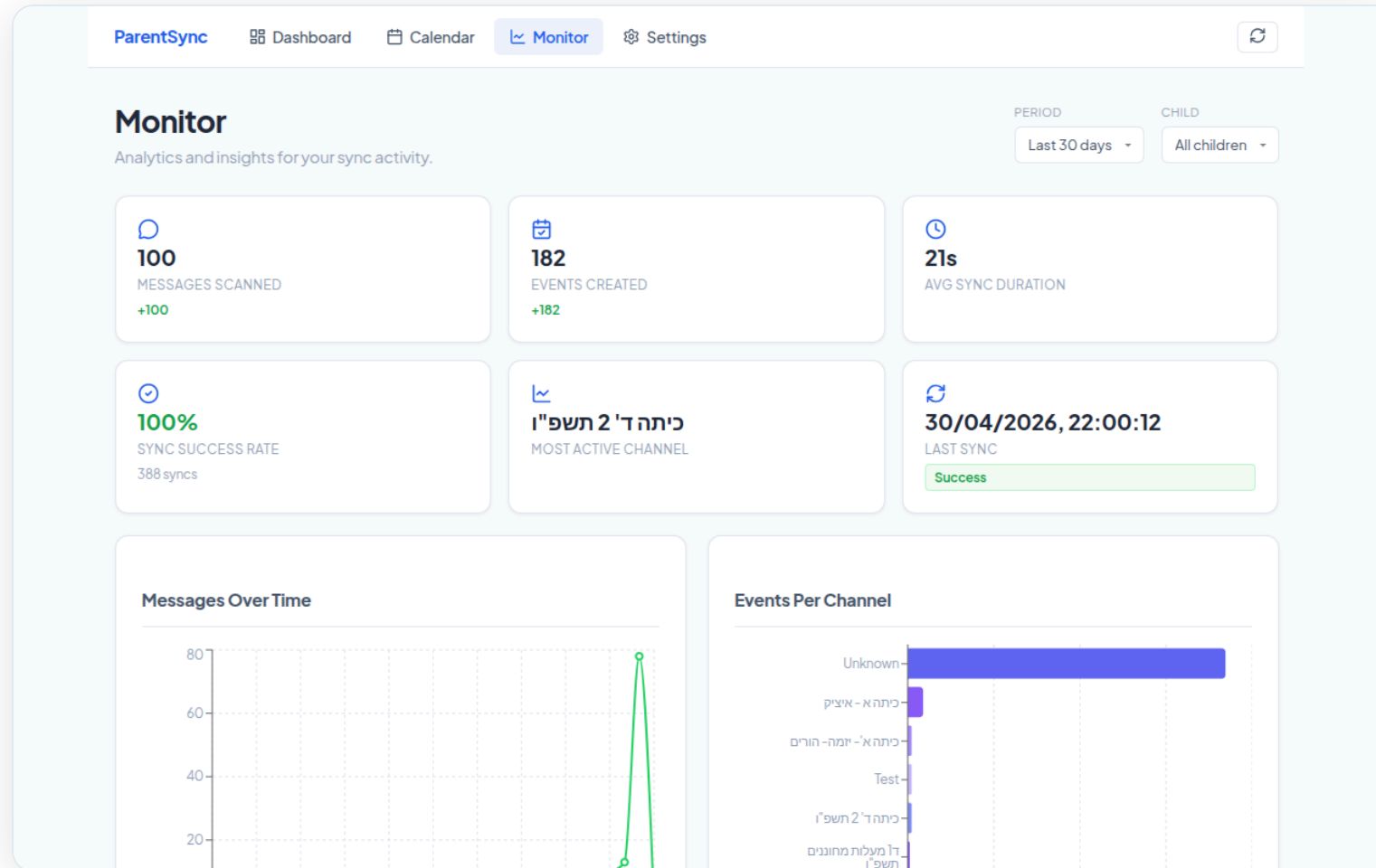
# Duplicate Suppression

One real-world WhatsApp message can describe the same gathering from multiple angles — a birthday is also a "meetup," a class trip is also a "ceremony." The LLM used to extract them as *separate* events at the same date+time. Now:

- Before sending an event for approval, the backend looks up other non-rejected events for the same child at the same date+time slot.
- For each match, a focused LLM call asks "**are these two events the same gathering?**" The prompt is tiny and cheap.
- If the answer is yes, the new event is silently rejected and never reaches your approval channel — the existing event remains the canonical record.
- Date-only tasks (no time) skip the check entirely — the false-positive risk is too high without a time slot.
- LLM hiccups default to "different" — a transient error never accidentally swallows a real event.

💡 The check pairs with the negative-feedback loop: even if a duplicate slips through, one 🙄 reaction puts it in the learned-exclusions pool so it's gone for good.

# Monitor & Analytics



Messages scanned, events created, sync success rate, and channel activity charts

# Monitor — Summary Cards

Six cards at the top give you a quick health check. Use the **Period** dropdown (7/30/90 days) and **Child** filter to narrow the view.

## Messages Scanned

Total WhatsApp + email messages collected. Green trend arrow shows change vs. previous period.

## Avg Sync Duration

Average time per sync cycle (scanning + parsing). Under 30s is normal.

## Most Active Channel

The WhatsApp group or email source that generated the most messages in this period.

## Events Created

How many calendar events the AI extracted from messages. Trend arrow compares to previous period.

## Sync Success Rate

Percentage of syncs that completed without errors. Green  $\geq 80\%$ , orange  $\geq 50\%$ , red below 50%.

## Last Sync

Timestamp and status badge of the most recent sync. Tells you when data was last refreshed.

# Monitor — Charts & Graphs

## Messages Over Time

**Line chart** with two lines:

● **Green** = WhatsApp messages

● **Red** = Email messages

X-axis: dates, Y-axis: message count. Shows daily volume trends — helps spot when groups are most active.

## Sync History

**Combo chart** — bars + line overlay.

**Bars:** message count per sync (**green** = success, **orange** = partial, **red** = failed)

**Line:** sync duration in seconds (right axis)

Spot slow syncs or recurring failures at a glance.

## Events Per Channel

**Horizontal bar chart** showing event count by WhatsApp group name.

Each channel gets a different purple shade. Hover to see percentage of total. Identifies which groups generate the most calendar events.

## Channel Activity Heatmap

**Grid** with channels as rows, dates as columns. Cell color intensity = message count. Darker purple = more messages. Hover any cell for exact count. Reveals which days and channels are busiest.

# Common Errors & Solutions

## ⚠️ "Sync failed"

WhatsApp may be disconnected, or the LLM API key is missing/expired. Check Settings → WhatsApp status and Gemini / OpenRouter API key.

## ⚠️ "0 Events Created" after sync

The AI didn't find any events in the messages. This is normal if messages are just chat. Only messages with dates/times get parsed into events.

## ⚠️ "Google OAuth not configured"

You haven't entered the Google Client ID and Secret yet. Go to Settings → Google OAuth and follow the setup steps (slide 17).

## ⚠️ "WhatsApp is not connected"

Session expired. Go to Settings → click "Connect WhatsApp" → scan QR code again with your phone.

## ⚠️ "Failed to load dashboard data"

Backend may still be starting. Wait a few seconds and click the **refresh button** (🔄 top-right). If it persists, restart the app.

## ⚠️ Events not appearing on Google Calendar

Check: 1) Google Calendar account is connected (Settings), 2) If using approval channel, react 👍 to approve the event in WhatsApp.