

Set up a Zapier Developer account and push Verae Time

Set up a Zapier Developer account and push Verae Time (tomorrow)

This is the only document you need in the morning to **create a Zapier developer account**, **push Verae Time**, and **run Add Numbers** (`number1 + number2 → sum`). No Verae servers, no NATS, no public URL.

App path: `packages/verae-activate`

0. What you will have when this works

1. A Zapier developer account (free).
2. A private integration named **Verae Time**.
3. A Zap: any trigger (e.g. Schedule) → **Verae Time** → **Add Numbers**.
4. A test that shows `2 + 3 = 5`.

After that, we grow the same integration into timestamping. Do **not** mix this with `zapier-sdk` (that consumes other people's apps). We use `zapier-platform` to *publish* ours.

1. Create the Zapier account (browser, ~5 minutes)

1. Open <https://zapier.com/sign-up> and create an account (Google/SSO is fine).
2. Confirm email if asked.
3. Open the Platform: <https://developer.zapier.com>
Same login. This is **Zapier Platform**, not the 8,000-app consumer directory yet.
4. You should land on **My Integrations** (empty is fine).

If you already have a Zapier account from using Zaps, skip sign-up and go straight to `developer.zapier.com`.

2. Install the CLI on this Mac (Terminal)

The CLI is already on this machine as `zapier-platform` (v19.1.0) in `~/.npm-global/bin`. In a new terminal:

```
export PATH="$HOME/.npm-global/bin:$PATH"
zapier-platform --version
```

You want **19.x**. If the command is missing:

```
npm install -g zapier-platform-cli
export PATH="$HOME/.npm-global/bin:$PATH"
zapier-platform --version
```

Do **not** use the old `zapier` binary if it is a different major from `zapier-platform-core` in the app (19.1.0).

3. Log the CLI into your account

```
zapier-platform login
```

A browser window opens. Approve the deploy key. This writes `~/.zapierrc` (never commit that file).

SSO-only org:

```
zapier-platform login --sso
```

Check:

```
zapier-platform whoami
```

4. Register and push Verae Time

```
cd /Users/marchon/apps/zapier/packages/verae-activate
npm install
npm test
zapier-platform validate
zapier-platform register "Verae Time"
zapier-platform push
```

`register` creates the integration and writes `.zapierapprc` (app id). **Do commit** `.zapierapprc` after register so the next push knows which app to update. **Do not commit** `~/.zapierrc`.

`push` uploads a **private** version. It is only visible to your account until you invite others.

If `register` says the name is taken, use "Verae Time Dev" or "VeraeTime".

5. Enable the integration in the Zap editor

1. Open <https://zapier.com/app/zaps> → **Create**.
2. **Trigger:** *Schedule by Zapier* → Every hour (or *Zapier Chrome extension* / any trigger you like). You only need something that can fire a test.
3. **Action:** search **Verae Time**. If it does not appear:
 - In the Zap editor app search, open **Change account** / custom integrations.
 - Or visit <https://developer.zapier.com> → your app → **Sharing** and confirm it is available to your user.
 - Hard refresh. Private apps can take a minute after the first push.
4. **Connect** Verae Time:
 - **API Key** — leave **blank**.
 - **API base URL** — leave **blank**.
 - Test connection → should succeed (*Verae Time (Add Numbers)* / local mode).
5. Choose action **Add Numbers**, **Echo Text**, **SHA256 Hash Text**, or **Create Timestamp** (local mock if API fields blank).
6. Set **Number 1** = 2, **Number 2** = 3.
7. **Test step**. You want:

```
{
  "number1": 2,
  "number2": 3,
  "sum": 5,
  "mode": "local"
}
```

Turn the Zap on if you want it to keep running. Failures on Zapier are free; successes count as tasks.

6. Optional: point Add Numbers at your hosted zappier API

Only after `packages/zappier` is reachable on the public internet (ngrok, fly, etc.):

1. Sign up in `packages/zappier` portal (`/portal`) → copy API key.
2. In the Zapier connection, set:
 - API Key = that key
 - API base URL = `https://<your-host>` (no trailing path)
3. Re-test. mode becomes "hosted" and POST `/v1/add` is metered (list price **free**).

Until then, leave both fields empty.

7. If something fails

Symptom	Fix
<code>zapper-platform: command not found</code>	<code>export</code> <code>PATH="\$HOME/.npm-global/bin:\$PATH"</code>
Not logged in	<code>zapper-platform login</code>
validate schema error	Stay on <code>zapper-platform-core@19.1.0</code> matching CLI 19.x
App not in Zap editor	Push succeeded? Check <code>developer.zapper.com</code> versions. Search "Verae".
Connection test fails with blank keys	You are on an old push; <code>git pull</code> / push again. Local test does not HTTP.
ENOTFOUND / timeout on test	You filled API base URL. Clear it for Add Numbers.
Invite a teammate	<code>developer.zapper.com</code> → app → Sharing → invite email

8. What this is *not*

- This is **not** `zapper-sdk` and **not** Zapier MCP.
 - This does **not** yet timestamp on `api.veraetime.net`. That is `packages/veraie-zapper` + middleware, next after Add Numbers is green on your account.
 - Do not put Verae JWTs in Zapier. Keys, when you use them, are zapper portal keys.
-

9. Commands cheat sheet

```
export PATH="$HOME/.npm-global/bin:$PATH"
cd /Users/marchon/apps/zapper/packages/veraie-activate
```

```
npm test
zapper-platform whoami
zapper-platform validate
```

```
zapier-platform push          # after the first register
zapier-platform versions
zapier-platform logs
```

Official CLI docs: <https://docs.zapier.com/platform/reference/cli-docs>