

campi cli

Table of contents

1 campi-cli 3
1.1 Build 3
1.2 Development 3
1.3 Roadmap 4
1.4 Documentation 5
1.5 Project 5
1.6 License 5

1 campi-cli

campi-cli will connect to many marketing platforms directly; Meta, TikTok and such, so you can manage campaigns as code; reviewable, version controlled and reproducible.



Figure 1: demo gif

1.1 Build

```
zig build
```

1.2 Development

The dev shell provides zig, clang, and the shared treefmt setup:

```
just             # list recipes
zig build        # build
just lint        # treefmt lint
just format      # treefmt format
```

To build and install the binary:

```
zig build --prefix ~/.local
```

1.3 Roadmap

- ☒ project scaffold
  - ☒ devshells
  - ☒ justfile
  - ☒ docs; quarto
  - ☒ branding
  - ☒ man pages
  - ☒ nix
  - ☒ ci
- ☒ schemas
  - ☒ config; the base config file
  - ☒ manifest; the campaign manifest
- ☐ cli
  - ☒ verbosity; debug logs
  - ☒ help command; print tool and commands help text
  - ☒ version; print the tool version
  - ☒ init; scaffold a new project
  - ☐ fmt; format manifest files
  - ☒ validate; validate the config and manifests
  - ☒ state; read and compare the state and lock files
  - ☒ config; manage the config file
  - ☒ plan; compute what would change
  - ☒ apply; apply the changes
  - ☒ import; import hand-created campaigns into state
  - ☐ platform; fetch the platform marketing API data
    - ☒ meta
    - ☒ tiktok
    - ☐ x
    - ☒ google; read-only
    - ☐ reddit
    - ☐ linkedin
- ☐ file handling
  - ☐ state reads and compares the state and lock files
  - ☒ plan and apply read the campaign manifest
  - ☒ init writes the base config file and prepares the working directory
- ☐ tests
  - ☐ cli
    - ☒ parser
    - ☒ runner
    - ☒ env
  - ☐ platforms
    - ☒ meta; payload and schema

  - ☒ tiktok; schema
  - ☒ google; schema and client
- ☐ artifacts
  - ☒ source resolution and caching

1.4 Documentation

- Installation
- Usage
- Commands
- Platform guides
- Development

1.5 Project

> [!NOTE]
>
> This project is not vibe-coded. AI is a coding companion, not a replacement for judgement. Everything is reviewed, critiqued, and guided; clean-code and architecture standards are enforced. This project wouldn't exist without the time AI saves. Locally hosted, energy-efficient models are being used when possible.

- Contributing
- Support
- Security

1.6 License

See LICENSE.