

1. Project Overview

Project Name: Internal Ads Management Assistant

Developer: Maxim Shumer

Purpose: This tool is an internal-use application designed to automate reporting, perform deep data analysis, and assist in the optimization of Google Ads campaigns for accounts owned by the developer. It operates through a local Model Context Protocol (MCP) interface, ensuring that all data processing remains private and within the local environment.

2. Target Audience

The tool is strictly for internal use only. There is no public access, no multi-tenant login, and no commercial distribution. The application interacts only with the accounts under our direct management.

3. System Architecture

The tool follows a modular architecture:

- * Interface Layer: Gemini CLI / MCP (Model Context Protocol). This acts as the command-line interface for triggering tasks.
- * Execution Layer: Python 3.x using the official google-ads client library.
- * Data Layer: Local file storage for configuration (google-ads.yaml) and output data (CSV/JSON).
- * Authentication: OAuth 2.0 Desktop Flow for secure, user-initiated authorization.

4. Key Features & Functionality

The application performs the following operations:

- * Automated Reporting: Executing Google Ads Query Language (GAQL) searches to retrieve performance metrics (CTR, Conversion Rate, ROAS).
- * Data Analysis: Using local AI processing to identify underperforming keywords or high-cost ad groups.
- * Optimization Recommendations: Generating data-driven suggestions for budget reallocation and bid adjustments.
- * Bid Management: (Planned) Applying bid changes based on predefined internal ROI thresholds.

5. Data Handling & Security

- * Authentication: We use OAuth 2.0 with the refresh_token stored locally. Developer tokens are never shared or hardcoded.
- * Privacy: All API responses are processed locally. No Google Ads data is sent to third-party servers.
- * Audit Logging: All API calls are logged locally for monitoring and compliance.
- * Scope: We only request the <https://www.googleapis.com/auth/adwords> scope necessary for managing our ads.

6. Workflow Example

1. User Request: The user asks for a report on campaigns with a ROAS below 200%.
2. API Call: The tool generates and executes a GAQL query via the GoogleAdsService.search method.
3. Local Processing: The data is saved to a local CSV file.
4. Analysis: The tool analyzes the CSV and provides a summary of recommended actions.

Коли подасте заявку:

Google зазвичай надсилає автоматичний лист із уточнюючими запитаннями. Просто відповідайте на них, підтверджуючи, що ви єдиний користувач і розробник.

Як тільки ви збережете цей документ і подасте його, ми зможемо продовжити з налаштуванням аналітичних скриптів у вашому **Test Account**, поки чекаємо на схвалення. Створимо тестовий акаунт?

? for shortcuts

auto-accept edits Shift+Tab to plan

1 GEMINI.md file · 2 MCP servers · 4 skills

> Type your message or @path/to/file

workspace (/directory)

sandbox

/model

~/Projects/Others/TestsAI

no sandbox

Auto (Gemini 3)

```
pmi7on9qjocg8bigrl7lots.apps.googleusercontent.com" --client_secret "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"
```

Please visit this URL to authorize this application: https://accounts.google.com/o/oauth2/auth?response_type=code&client_id=969563964616-gsd4korjrpmi7on9qjocg8bigrl7lots.apps.googleusercontent.com&redirect_uri=http%3A%2F%2Flocalhost%3A49773%2F&scope=https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fadwords&state=ntHYDlSIQ78TYsS7t7xAkP3RaAoZWWh&code_challenge=8dcZVoARtKNp1TAjASbfmMN153f5yzwD-0ZSM6sqoOY&code_challenge_m

html ☐ Ready (TestsAI)