

Making Gemini CLI extensions easier to use

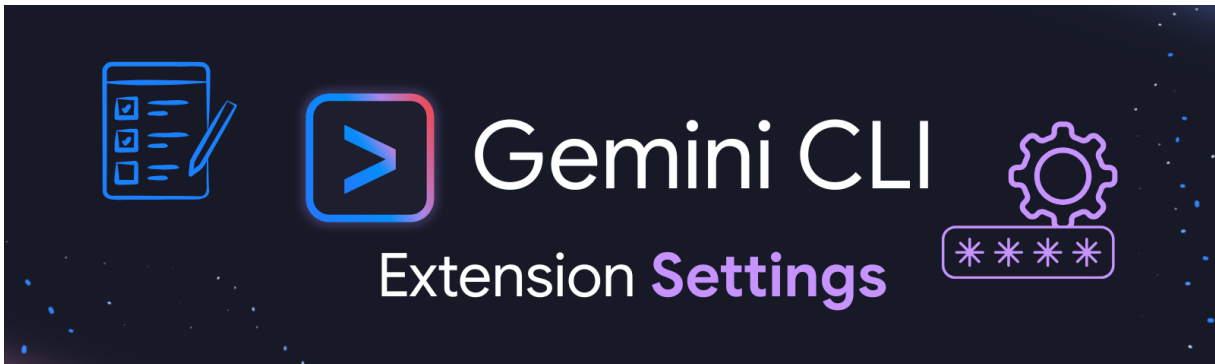
FEB. 11, 2026

Jack Wotherspoon
Developer Advocate

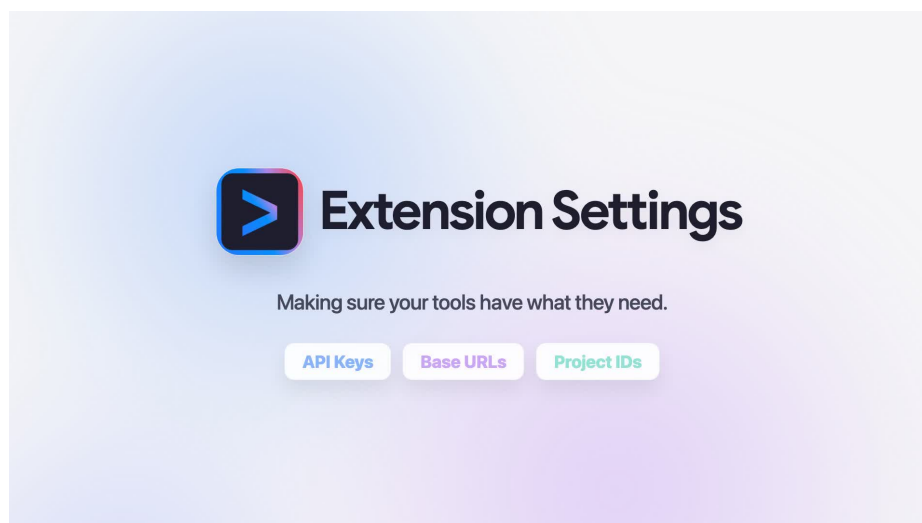
Christine Betts
Software Engineer
Gemini CLI

Bala Narasimhan
Group Product Manager, Google Cloud

[Share](#)



Stop debugging ambiguous startup failures caused by missing API keys, hidden environment variables, or a crashing MCP server. AI tools are most effective when they're tailored to your specific environment and data but manual configuration is often brittle and confusing. This is why we are launching **extension settings**. Gemini CLI extensions can now define settings that the user will be prompted to provide upon installation, ensuring extensions have exactly what they need to function from the moment you install them.



A first-class experience for extension configuration

Extension settings provide a structured approach that ensures your experience is painless and straightforward. Extension authors can now define exactly what information their tool needs to function; whether it's an API key, a base URL, or a project identifier. No more digging through README's or trying to decode error messages.

manifest text file.

- **Command-line management:** The new gemini extensions config command provides a centralized way to view and update settings.
- **Scoped overrides:** Support for both global user and project-specific workspace scopes ensures the right configuration is used in the right context.

Practical example: The AlloyDB extension

The [AlloyDB extension](#) is a perfect example of why structured settings matter. To connect with your database and interact with your data, the extension requires several pieces of information: a *project ID*, *region*, *cluster ID*, *instance ID*, and *database*.

In the past, you had to manually export these as environment variables. Now, the extension defines them in its manifest. When you install it, Gemini CLI guides you through the setup:

```
Shell
1 ? Project ID: Your GCP project ID
2 ? Region: us-central1
3 ? Cluster ID: my-cluster
4 ? Instance ID: my-primary-instance
5
6
7 ✓ Project ID
8 ID of the Google Cloud project: my-google-cloud-project
9 ✓ Location
10 Region of the AlloyDB cluster: us-central1
11 ✓ Cluster ID
12 ID of the AlloyDB cluster: my-cluster
13 ✓ Instance ID
14 ID of the AlloyDB instance: my-instance
15 ✓ Database
16 Name of the database: postgres
17 ✓ User
18 (Optional) Username of the database user (Default: IAM user): postgres
19 ✓ Password
20 (Optional) Password of the database user (Default: IAM user): *****
21 ✓ IP Type
22 (Optional) Type of the IP address: PUBLIC, PRIVATE, or PSC (Default: Public):
23 Extension "alloydb" installed successfully and enabled.
```

These values are saved securely and provided to the extension's MCP server automatically, so you can start querying your data immediately. This setup provides a reliable and secure method for database connection. It ensures you have constant visibility into which database is active and prevents the agent from altering your configuration.

Check out these updates for all **Data Cloud extensions**: [bigquery-data-analytics](#), [cloud-sql-mysql](#), [cloud-sql-postgresql](#), [cloud-sql-sqlserver](#), [firestore-native](#), [looker](#), [spanner](#).

How to define settings as an author

To add configuration to your extension, include a settings array in your gemini-extension.json file. Each setting specifies a user-friendly name, a description, and the environment variable it should be mapped to.

```
JSON
1 {
2   "name": "my-cool-extension",
3   "version": "1.0.0",
4   "settings": [
5     {
6       "name": "API Key",
7       "description": "Your secret API key for the service.",
8       "envVar": "MY_API_KEY",
9       "sensitive": true
10    },
11  ]
12 }
```

When a user installs this extension, Gemini CLI guides them through the setup process. If a setting is marked as sensitive, the input is obfuscated during entry and stored securely in the system keychain.

Managing settings with the config command

The `gemini extensions config` command is your primary tool for managing extension configuration after installation. Use it to update settings interactively or target specific values directly.

Update all settings for an extension

To walk through every setting defined by an extension, run:

Shell

```
1 | gemini extensions config <extension-name>
```



Update a specific setting

If you know exactly which value needs to change, specify the environment variable name:

Shell

```
1 | gemini extensions config <extension-name> <env-var-name>
```



Use workspace overrides

For settings that vary between projects, use the `--scope workspace` flag to save the value in your current directory's configuration rather than globally:

Shell

```
1 | gemini extensions config alloydb ALLOYDB_POSTGRES_CLUSTER --scope workspace
```



Debugging the configuration

When troubleshooting issues with extensions, it is often helpful to verify that all configuration settings are correct. A simple typo in a *project ID* or a *missing API key* can prevent an extension from functioning properly.

You can inspect the currently installed extensions and their active settings using the `extensions list` command. This functionality is available via the **command line** and as a **built-in command**. Both will output a summary of your installed extensions, their version status, and their current configuration.

Command Line

Shell

```
1 | gemini extensions list
2
3 # ✓ alloydb (0.1.10)
4 # Source: https://github.com/gemini-cli-extensions/alloydb (Type: github-release)
5 # Release tag: 0.1.10
6 # ...
7 # Settings:
8 #   Project ID: my-gcp-project
9 #   Location: us-central1
10 #   Cluster ID: my-alloy-db
11 #   Instance ID: my-primary
12 #   Database: postgres
```



Built-in Command

Shell

```
1 /extensions list
2
3 # Installed extensions:
4 # alloydb (v0.1.10) - active (up to date)
5 #   settings:
6 #     - Project ID: my-gcp-project
7 #     - Location: us-central1
8 #     - Cluster ID: my-alloy-db
9 #     - Instance ID: my-primary
10 #     - Database: postgres
11 #     - User: postgres
12 #     - Password: ****
13 #     - IP Type: public
```

Best practices for extension settings

- **Provide clear descriptions:** Help users understand exactly where to find the values they need (e.g., "Find this in your GCP Console under IAM").
- **Use sensitive for secrets:** Always set "sensitive": true for API keys, passwords, or any data that should not be stored in plain text.
- **Keep names concise:** Use short, user-friendly names that fit well within terminal prompts.
- **Leverage workspace scope:** Use workspace-scoped settings for project-specific IDs to prevent global configuration pollution.

Get started today

Extension settings and the gemini extensions config command are available as of **v0.28.0+**. Update to the latest version by running:

Shell

```
1 | npm install -g @google/gemini-cli@latest
```

Check out our updated documentation to learn more:

- [Extensions reference](#)
- [Writing extensions guide](#)

Try out extension settings today when building Gemini CLI extensions and if you run into any issues let us know on our [GitHub repository](#) or on socials!

You can also follow [Gemini CLI on X](#) to stay up to date with the latest news and announcements.

POSTED IN:

[AI](#)[Announcements](#)[Learn](#)[PREVIOUS](#)[NEXT](#)

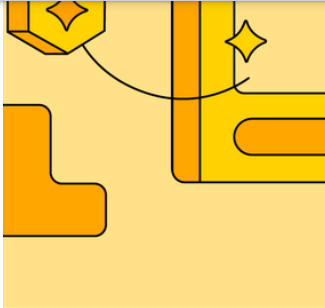
Related Posts



AI CLOUD ANNOUNCEMENTS SOLUTIONS

Expanding Choice in Gemini Enterprise Agent Platform: Introducing Grounding with Parallel Web Search

JULY 16, 2026



AI BEST PRACTICES INDUSTRY TRENDS

Building scalable AI agents

JULY 16, 2026



Connect

- Blog
- Bluesky
- Instagram
- LinkedIn
- X (Twitter)
- YouTube

Programs

- Google Developer Program
- Google Developer Groups
- Google Developer Experts
- Accelerators
- Women Techmakers
- Google Cloud & NVIDIA

Developer consoles

- Google API Console
- Google Cloud Platform Console
- Google Play Console
- Firebase Console
- Actions on Google Console
- Cast SDK Developer Console
- Chrome Web Store Dashboard
- Google Home Developer Console

- Android
- Chrome
- Firebase
- Google Cloud Platform
- All products