

[• YOUR DATA](#)

Where does SumizAI store your notes?

The question behind this one is usually “what happens to my notes if this application goes away”. **The answer is nothing**, and the reason is structural rather than a promise: the application was built with the files as the source of truth and the database as a cache of what is in them.

Updated: 2026-08-14 8 questions 4 min read

Q. What exactly is written to disk?

Markdown files, in a directory you chose when you set the application up.

The layout is one directory per vault. Inside it, `base.md` holds the generated table of contents, and a `notes/` directory holds one `.md` file per note. Each filename is the slug of the note's title.

That is the whole scheme. There is no hidden metadata directory, no sidecar database in the folder, no proprietary container. Open the folder in Finder and you can read your library.

Q. What is in an individual note file?

Front matter and the note body: the title, the summary, the expansion, and the source material — the original question and the original answer.

The file is deliberately **complete on its own**. Everything the application's index knows about a note can be recovered from the file, which is what makes the index disposable.

Links to other notes appear inside the body as relative Markdown links, so a note is readable and navigable in any editor.

Q. If the files hold everything, what is the database for?

Speed. It knows titles, summaries, the storage key for each file and the derived columns that make search fast, and it answers questions the files cannot answer cheaply — such as which notes link to this one.

It is explicitly an index, not a second copy. There is a reindex operation that rebuilds it entirely from the contents of the storage, and **in any disagreement the file is correct**.

This is a genuine architectural commitment, not a marketing line. It is why the note view reads the raw Markdown straight from the file by name without touching the database at all.

Q. How do I take a vault with me?

There is a *Download vault* button on the vault page. It gives you a ZIP containing `base.md` and the `notes/` directory — the vault exactly as it is.

Unzip it and open it in Obsidian, VS Code or a GitHub repository. The links work immediately, because they are relative paths between files that are all in the archive.

No import step, no conversion, no plugin. That is the test of whether an export is real: does the thing you downloaded work without the application that produced it.

Q. What should I back up?

The folder. That is genuinely all — it holds the notes and the table of contents, in an open format.

If you are running the server version yourself rather than the desktop application, note that a database dump is no longer sufficient on its own: the note content lives in object storage, so you need both the database volume and the storage volume. The database alone would restore an index pointing at nothing.

On the desktop the question does not arise in the same way, because the folder is the whole story and your normal backup already covers it.

Q. Do filenames change?

Yes, when a title changes — and that is a deliberate reversal of an earlier decision.

Filenames used to be permanent, which made renames free and produced folders full of names that no longer described their contents. Now the filename is the slug of the title, so renaming a note moves the file and rewrites every link to it across the vault.

Two consequences worth knowing. Collisions get a numeric suffix — `(1)`, `(2)` — decided by asking the storage what already exists rather than trusting the index. And an archive you exported before a rename keeps the old names; that cost was accepted knowingly in exchange for folders that read correctly.

Q. Where does my AI provider key live?

In the device's keychain on the desktop application, and encrypted at rest in the server version.

In the server version the key is encrypted with AES-256-GCM, decrypted only for the duration of a request, and never returned to the browser in full or written to logs. The masking in the log configuration is written greedily on purpose — a lazy pattern masks the beginning of a secret and leaves the rest in plain text, which is the kind of bug that only shows up in an incident.

On the desktop the calls go out from your machine directly to the provider, which is a genuinely different security model and worth stating plainly rather than glossing.

Q. What is not stored, and what is not synced?

Nothing is stored that you did not create: notes, the conversations that produced them, your vault settings and your encrypted provider key.

There is **no synchronisation between devices**. Your laptop and your phone hold two separate sets of notes. The data model carries the revision, change timestamp and per-device change log that a future sync would need, but the feature does not exist today.

We would rather say that clearly than let you discover it after two weeks of writing notes on the wrong machine.