SQUIRREL.AI

COMPANY BRAIN BLUEPRINT

Fernhill Bikes

Prepared for Hannah Ashworth · 26 November 2026 · Drafted automatically from public information by Squirrel AI

What we understood about Fernhill Bikes

Fernhill Bikes designs and sells bicycles, both direct through its own website and through independent bike shops. New models are developed in-house and launched to retailers each season, and the company handles warranty claims itself.

Your work seems to run in **product launches and retailer accounts**. A company brain is organised around that unit, not around departments, because a project, deal or account gives every page a natural owner and every action somewhere to belong.

Where we would start: Product Launch Pages

A launch has a fixed date, a single owner and many small promises made on calls that are easy to forget. It is a contained place to prove the record can be trusted before widening it.

Weeks 1 to 2 Choose one upcoming model with its product manager and write down what counts as a decision, an action and a launch risk.

Weeks 3 to 4 Compile the page for that model from the last two months of launch calls, chat and the product tracker, with every line traced to its source.

Weeks 5 to 6 Review the page with the product manager, correct what is wrong, count what was missed and decide whether to add a second model.

> The test at the end of six weeks: does the owner trust the page enough to stop keeping their own notes?

The pages we would build for Fernhill Bikes

A brain is a small set of short pages. Each answers one question, has one accountable owner, and is compiled from the sources listed, never written by hand.

1. Product Launch Pages

For each model, where does the launch stand, what has been decided and what is still outstanding?

OWNER

Product Manager for the model

FED BY

launch sync calls · Slack channels for each model · Airtable product tracker · Google Workspace email threads with test labs

KEEPS CURRENT

- Decisions on launch date, price and specification
- Actions agreed on launch calls, with owner and due date
- Checklist items such as safety certificate, first stock and retailer briefing

A QUESTION IT ANSWERS

Are we still on for the launch date, and what is left to do before it?

2. Retailer Account Pages

What have we agreed with each bike shop, and what do we owe them?

OWNER

Head of Retail Sales

FED BY

retailer calls · Google Workspace email threads with each retailer · Slack threads about retailer requests

KEEPS CURRENT

- Commitments on stock, demo bikes and margins
- Requests from the retailer that are still open
- Decisions on terms agreed for the season

A QUESTION IT ANSWERS

What did we promise this shop for the new model, and has it been sent?

3. Warranty Claims Rules

What are our rules for accepting a warranty claim, and is the team following them?

OWNER

Head of Customer Service

FED BY

Zendesk tickets · Slack threads where exceptions are discussed · the written warranty policy

KEEPS CURRENT

- The rules as currently stated
- Exceptions granted and who approved them
- Places where what people say in chat differs from the stated rule

A QUESTION IT ANSWERS

Do we replace a frame after two years, and who can approve an exception?

4. Supplier Commitments

What has each component supplier agreed to deliver, and by when?

OWNER

Head of Operations

FED BY

supplier calls · Google Workspace email threads with suppliers · Airtable supplier tracker

KEEPS CURRENT

- Delivery dates and quantities promised
- Risks to supply raised in conversation
- Actions to chase or re-plan

A QUESTION IT ANSWERS

When did the supplier say the frames would arrive, and has that moved?

5. Launch Calendar

Which models launch when, and do any dates conflict?

OWNER

Head of Marketing

FED BY

Product Launch Pages · Airtable product tracker · Shopify product listings

KEEPS CURRENT

- Launch date for each model
- Changes to dates and who decided them
- Launches whose checklist is behind

A QUESTION IT ANSWERS

What is launching this quarter, and is anything at risk of slipping?

Definitions to settle first

This is the step most projects skip. No model captures a decision or a commitment consistently until the company has written down what one is, in its own terms.

- **Launch date.** Retailers plan around it, so the company needs one meaning: the day stock is in shops, not the day of the announcement.
- **Retailer commitment.** A remark on a call can become a promise a shop relies on, so you need a rule for when something said counts as owed.
- **Warranty exception.** If exceptions are granted in chat, the brain can only flag drift from the rules once an exception is defined and has a named approver.

Tools we would ask you about

These appear in your domain's public DNS records or on your website, which is how we could name them. That tells us a tool is connected, not how you use it, so treat this as a list of questions.

TOOL	WHAT IT HOLDS	PAGES IT WOULD FEED
Google Workspace	email threads, shared documents and meeting invites	Product Launch Pages, Retailer Account Pages, Supplier Commitments
Slack	day-to-day decisions and requests made in channels and threads	Product Launch Pages, Retailer Account Pages, Warranty Claims Rules
Airtable	the trackers teams keep for launches, suppliers or pipelines	Product Launch Pages, Supplier Commitments, Launch Calendar
Zendesk	support tickets and what customers were told	Warranty Claims Rules

Shopify	orders, products and customer records	Launch Calendar

How your information would be handled

Nothing in this blueprint needed access to your systems. If you went ahead:

- **Read-only.** The brain connects to your tools to read them. It never writes to them.
- **Only where you add it.** It reads the spaces you add it to, at a level your administrator sets for each tool.
- **Access carries through.** A line on a page is visible only to people who could already see all the evidence behind it.
- **Agreed before anything is connected.** Where it is hosted, on your cloud or ours, and which AI model providers are used are settled with you first.

What we assumed

We have only seen what is public, so these are guesses to correct.

- Fernhill Bikes is a made-up company we use for testing, so this sample was written from our test data, not from a website.
- The tools listed are illustrative. In a real blueprint they come from public records for your domain.
- Owner roles are our best guess at who would approve each page.

What agents could do once the pages are trusted

An agent is only as good as the context it is given, so the order matters: record first, agents second.

- Before each launch sync, an agent could list the actions still open for the model and who owes them.
- When a retailer asks about a new model, an agent could answer from the approved launch page and cite where each fact came from.
- When a chat message contradicts a stated warranty rule, an agent could flag it to the page owner instead of letting it pass.

Three questions to ask any vendor, including us

1. When the system gives a wrong answer, who is responsible for correcting it, and does the correction stick?
2. Can I click from any statement to the exact source words behind it?
3. How do you measure what the system missed, and will you show me the number?

> You can put the second question to us now. A real page from our engine, where every line opens to the exact words it came from, is at squirrelai.co.uk/company-brain/demo

> This blueprint was drafted from public information, so parts of it will be wrong. A 30 minute working session with a founder would correct it and turn it into a plan: squirrelai.co.uk/contact

APPENDIX

A company brain in one minute

Most of what a company knows was never written down as a document. It was said on a call, agreed in a thread or settled in a reply to an email. A company brain compiles that into a small set of short pages. Each has a named owner who approves it, and every line traces back to the exact words it came from. People read it. AI assistants and agents work from it.

The idea is **owned truth, not search**. Search finds what was said. A brain states what is true now, shows how it knows, and has someone accountable for each part.

How it differs from what you have

APPROACH	HOW IT WORKS	WHERE IT FALLS SHORT
Shared drive or wiki	People write it by hand	It goes out of date the week people get busy
Search or an AI assistant	Looks things up when asked	Nobody owns the answer, and old and new look alike
Company brain	Compiled from the work itself	It needs owners and agreed definitions to start

How a brain is built

STAGE 1

Keep the originals

Calls, email, chat and documents are read where they live and kept as they were.

›

STAGE 2

Pull out facts, with quotes

Each fact carries the exact words it came from. Code checks the quote exists, or the fact is dropped.

›

STAGE 3

Work out what is true now

New facts are matched against old. History is kept, and conflicts go to a person.

›

STAGE 4

Compile pages owners approve

Short pages, one question each. The owner sees what changed and signs it off.

Pages are compiled, not typed. An owner can still correct anything: the correction is recorded as part of the evidence, and a person's decision is never overwritten by a model.

Six mistakes to avoid

These apply whoever you build with, including if you build it yourselves.

1. **Connecting everything first.** Start with one area where lost knowledge costs money. Prove the pages can be trusted, then widen.
2. **Skipping the definitions.** If nobody has written down what counts as a decision or a commitment, no model will capture them consistently.
3. **Reading one message at a time.** A line in a thread only makes sense beside the lines around it. Knowledge is a web, so conversations must be read whole.

4. **Treating every fact as equal.** The biggest facts must be captured. Rate importance from the start, using your own priorities.
5. **Trusting it without a number.** A wrong entry gets caught by its owner. A missed one nobody sees. Ask how misses are measured.
6. **Starting with the agents.** Agents acting on an unreliable record make confident mistakes faster. Build the record first.