

WEBSITE REVIEW

getvie.com, read closely

Your site exactly as it stands, with every finding written the same way: what I found, why it matters, how I would fix it, and the result we would measure afterwards. Nothing here is a guess dressed as a fact.

WHAT THIS IS FOR

Vie already has the hard part, which is three thousand recipes and a matching engine that works. This review is about the distance between that and a product people can find, trust and pay for.

At a glance

6

to fix first

6

next up

2

polish

5

growth moves

Links

Design

UX

Engineering

Mobile

Forms

Accessibility

SEO

AEO

Copy

Performance

MEASURED

found and reproduced on this build, with the method recorded

SAMPLE NOTE

an illustration of the format, not a claim about this site

4 of the 14 notes below are measured. The rest are marked sample on every surface that shows them, including this page, the pins on the site, and the PDF.

Fix first

The ones that cost something every day they stay.

1

Every link you share lands on the homepage, whatever you linked to

Links

2

The first time here card sits on top of the pills

Design

3

The homepage opened halfway down, with the hero scrolled away

UX

5

The shopping list does not reflow on a phone

Mobile

6

The signup form accepts an address and says nothing back

Forms

7

A footer link points at a page that was never built

Links

Five growth moves

Not fixes. The things that would change the size of this.

1

Make the archive visible, because it is the moat

Three thousand recipes you wrote, cooked, styled and shot is the one thing a competitor cannot buy or copy. Right now a search engine cannot see any of it: no Recipe markup, no recipe pages indexed, nothing eligible for a rich result. The archive is doing none of the work it could be doing.

GENERATE Emit Recipe structured data per recipe from recipes.json at build time. The fields are already parsed: ingredients, instructions, times, yield. Skip any recipe missing a time rather than inventing one.

PUBLISH Ship the recipe box as real indexable pages rather than a feather marked soon. Start with the 608 that already have a photograph.

MEASURE Impressions and clicks on recipe pages in Search Console, monthly. This is the number that tells you whether the archive is earning.

RESULT The asset you already own starts bringing people in, instead of sitting behind a card that says coming soon.

2

Answer the five questions that stop a signup

The page argues well and answers nothing. There is no FAQ, no question shaped heading, and no short self contained answer anywhere. That costs twice: it leaves the real objections unaddressed for a human reader, and it leaves nothing for an assistant to quote when someone asks it how to plan meals around sale prices.

WRITE Five questions, answered in forty words each, complete without the paragraph above them. Which stores. What if mine is not covered. Do I have to cook all five. What does it cost. What happens to my data.

MARK UP FAQPage structured data so the answers are machine readable.

BE HONEST Name the current store coverage rather than implying all of them. The waitlist for uncovered stores is worth more than the pretence.

RESULT Fewer people leave with an unanswered objection, and the page becomes quotable by the assistants a growing share of people now ask first.

3

Treat the Wednesday email as the product

The site sells a weekly email, and the weekly email is where the relationship actually lives. Today the signup path ends at an address in a list, with the welcome built but inert and no sending domain verified, so a real signup currently receives nothing at all.

UNBLOCK Verify getvie.com in Resend and set RESEND_FROM. Until that is done mail only reaches your own verified address, which means every real signup so far has had silence.

WELCOME Turn on the welcome, with an unsubscribe link, which CAN-SPAM requires and which the current draft does not have.

ROUTE Send every link in the email through the site so clicks are exact and a stale link stays editable after the send.

RESULT A signup becomes a subscriber instead of a row, and you can see which parts of the plan people actually open.

4

Close the loop between sale prices and the plan

The Kroger connection works and the matching engine is good: given only your own sale flagged ingredients it finds your recipe in the top three of 860. But the plan on the site still runs on a static sample, and the sales writer is failing on a Blobs permission error, so the live pipeline stops short of the thing the whole product promises.

FIX Resolve the Netlify Blobs 401 so fetch-weekly-sales can write sales-state. Check the token scopes and the site id.

CONNECT Point the plan engine at the real recipe data rather than the static RECIPES sample in plan.js.

PROVE Publish the weekly saving against the same basket at full price. It is the only number that makes the claim in the headline checkable.

RESULT The sentence on the homepage becomes demonstrably true, with a number attached, which is a different kind of marketing entirely.

5

Decide what is free and build the door

The site gates the plan behind an email and says the first week is free, but there is no paid tier behind it. The pricing history in the old database runs from 9.99 a month to 399 a year, and the last real signups were at 15 a month in 2024. Right now there is nowhere for a willing customer to pay.

CHOOSE One price and one promise. The archive plus the weekly plan is a stronger package than the plan alone.

BUILD Checkout, then gate the printable plan and the shared list behind it. Keep the first week genuinely free.

HONOR The old list has roughly four hundred to seven hundred real people on it with no outstanding obligations. They are the warmest launch audience available and they have not heard from you since the host went down.

RESULT A product that can take money, launched to people who already paid for it once and liked it.

Every note, in order

Numbered the same here, on the site, and in the PDF, because all three read from one list.

1

Every link you share lands on the homepage, whatever you linked to

MEASURED

Fix first

Links

WHAT

A script at the top of index.html strips the fragment from the URL whenever the visitor did not arrive from getvie.com. So `getvie.com/#plan` sent by text, email, Instagram or a QR code opens the homepage instead of the plan. Clicking the same link inside the site works perfectly, which is why it survived a month without being spotted.

WHY

This is the one that costs money. Every link in a newsletter, every link in a post, every link you paste into a DM to a prospective subscriber points at a specific thing and delivers the front door instead. The visitor does not report it. They assume they tapped the wrong thing, or that the page is not there, and they leave.

HOW

Delete the script. It was written to stop the page opening below the fold, and it never did that, because the scroll came from `settle()` inside `showHome()` rather than from the fragment. Fixing the real cause lets the fragment stay, so deep links work again.

RESULT

Shared links reach what they name. Already fixed in v26, which is the build this review sits on.

HOW IT WAS CHECKED

Chromium, three builds side by side. v24 opens `#plan` correctly. v25 strips it and shows the homepage. v26 opens it again. The hash is gone before `DOMContentLoaded` fires, which is how the script was traced.

The first time here card sits on top of the pills

MEASURED

Fix first

Design

WHAT

The card is `position:absolute`, but it lived in the left column of the hero while `.hero` was the only positioned ancestor above it. Every comment in the stylesheet assumes it anchors to `.herostage`, the photo column. It never did. Anchored to the whole hero, `bottom:0` parks it across the pill row.

WHY

The pills are the only navigation on the homepage that names what Vie does. Covering them hides the five things a first visitor most needs to see, and it makes the page look broken rather than busy, which is a harder thing to come back from on a site whose whole argument is care.

HOW

Move the card inside `.herostage` so the positioning resolves against the element it was written for, and reserve enough room under the photo for it to hang without clipping the byline.

RESULT

The card cannot reach the pills at any width, because they now sit in different columns. Fixed in v26.

HOW IT WAS CHECKED

Measured at thirteen widths from 360 to 1440. Before: overlapping. After: 52px clear side by side above 880, stacked 588px below the pills under 880, and 24px clear of the byline rather than clipping it by 14.

3

The homepage opened halfway down, with the hero scrolled away

MEASURED

Fix first

UX

WHAT

`boot()` finished by calling `showHome()`, and `showHome()` always called `settle()`, which glides to the top of the enclosing band. On a first load that scrolled past the headline, the photograph and the pills, and opened the site on the Wednesday section.

WHY

The hero carries the whole proposition: meal plans that save you money, the sentence about reading the sale prices, and the photograph that tells someone a real person cooks this. Skipping it means a first visitor starts in the middle of an argument whose opening they never saw.

HOW

Treat the first paint as what it is. There is nowhere to scroll back to before anyone has navigated, so settle only on a real navigation, and set `scrollRestoration` to `manual` so a reload starts at the top too.

RESULT

The site opens at the top on every width tested, and clicking through to a feather still scrolls to it. Fixed in v26.

HOW IT WAS CHECKED

`scrollY` on load was 927 before and is 0 after, at all thirteen widths. Clicking a pill still scrolls, so the fix did not flatten navigation.

4

The deploy folder carries build files Netlify will try to act on

MEASURED

Next up

Engineering

WHAT

The folder that gets dragged to Netlify contains package.json with a real dependency, a src/ directory, add_plausible.py, check-preview.py, two .bak files and three markdown handoffs. The functions are already bundled, so none of it is needed at deploy time.

WHY

Your own Glenaire notes record this one: a package.json or a src/ directory in a drag and drop folder makes Netlify run an install it was not asked to run. It is deploying cleanly today, so this is a trap rather than a fire, but it is the kind that goes off during a launch rather than on a quiet Tuesday. The .bak files are also a second copy of the site sitting next to the site.

HOW

Move the sources and the handoffs up one level into vie/, the way glenaire-dev already does it, and leave only what is served in the deploy folder.

RESULT

A deploy folder where everything present is something a browser asks for, and no install step that can surprise you mid launch.

HOW IT WAS CHECKED

Read from the v25 deploy zip. Left alone in v26 on purpose: it is a restructure, not a bug fix, and it did not belong in a patch aimed at two visual faults.

The shopping list does not reflow on a phone

SAMPLE NOTE**Fix first****Mobile****WHAT**

A week's list is laid out as a fixed table. On a phone the columns keep their widths, so the quantity column slides under the thumb and the aisle column leaves the screen entirely.

WHY

The shopping list is the one screen that gets used standing up, in a store, one handed, on a phone. If any part of Vie has to be perfect on a small screen it is this one, and a layout that works everywhere except the aisle is a layout that fails at the only moment that counts.

HOW

Drop the table below 600px and stack each line as its own row: item first, quantity underneath in the lighter weight, aisle as a small tag. Keep the tick target at 44px so it can be hit while walking.

RESULT

A list that can be worked through in a store without pinching or scrolling sideways, which is the difference between a tool someone uses weekly and one they screenshot once.

This note is an illustration of the format, not a finding about this site. It shows how a note of this kind reads before it is pointed at a real client.

6

The signup form accepts an address and says nothing back

SAMPLE NOTE

Fix first

Forms

WHAT

Submitting the email form clears the field and leaves the status line empty. Nothing confirms the address was taken, and nothing appears if it was not.

WHY

Silence after a form reads as failure. People submit again, or they assume it is broken and go. On the band that exists purely to collect addresses, a quiet success is indistinguishable from a dropped one, and you cannot tell the difference either when you look at the numbers afterwards.

HOW

Read the real response and say what actually happened. On success, name the next thing: the plan lands Wednesday. On failure, keep what they typed, say so plainly, and offer a mailto fallback so the address is not lost.

RESULT

Every submission ends in a sentence, and the ones that fail are recoverable instead of silently gone.

This note is an illustration of the format, not a finding about this site. It shows how a note of this kind reads before it is pointed at a real client.

7

A footer link points at a page that was never built

SAMPLE NOTE

Fix first

Links

WHAT

One of the links in the footer resolves to a path with nothing behind it and returns the 404.

WHY

A dead link in the footer is a small thing that does a specific kind of damage: it tells a visitor the site is not maintained. On a subscription product, where the whole ask is that someone trusts you to turn up every Wednesday, looking unmaintained is expensive.

HOW

Either build the page or remove the link today, and add a link check to the pre ship list so the next one is caught before it ships rather than after.

RESULT

No dead ends, and a check that keeps it that way without anyone having to remember.

This note is an illustration of the format, not a finding about this site. It shows how a note of this kind reads before it is pointed at a real client.

8

Alt text still describes the photograph that used to be there

SAMPLE NOTE

Next up

Accessibility

WHAT

Two images carry alt text written for earlier photographs. The words describe a dish that is no longer in the frame.

WHY

Wrong alt text is worse than none. A screen reader announces a confident description of something that is not on the page, and a search engine indexes the same mistake. It also undoes the thing these photographs are for, which is proving that you cooked and shot every one.

HOW

Rewrite both against what is actually in the frame, in the same register as the rest: what the dish is, what it is on, what is around it. Then make alt text part of the image swap, so it moves when the photo moves.

RESULT

Descriptions that match the pictures, and a habit that stops them drifting apart again.

This note is an illustration of the format, not a finding about this site. It shows how a note of this kind reads before it is pointed at a real client.

Card titles disappear in dark mode

[SAMPLE NOTE](#)[Next up](#)[Design](#)

WHAT

The feather card headings sit inside link and button wrappers and never set their own color. They inherit the browser default, which is near black, and the cards go dark when the system is in dark mode.

WHY

Dark text on a dark card is not hard to read, it is invisible. The card looks empty, so the page looks broken, and the person it happens to has no way to tell you what they saw beyond saying it did not work.

HOW

Set an explicit palette color on every heading rather than letting it inherit. This is already written down as a house rule, so it is a compliance fix rather than a new decision.

RESULT

Every card readable in both modes, enforced by the contrast suite so it cannot come back.

This note is an illustration of the format, not a finding about this site. It shows how a note of this kind reads before it is pointed at a real client.

The meta description is a tagline where an answer belongs

[SAMPLE NOTE](#)[Next up](#)[SEO](#)

WHAT

The description tag repeats the brand line rather than saying what the page gives someone. Search results show it verbatim.

WHY

The description is the only sentence you get to write in a search result. A tagline asks the reader to work out whether this is for them. A plain answer tells them, and the click rate follows the difference.

HOW

Write it as the answer to the question that brought them: a weekly meal plan built from your own store's sale prices, with the shopping list done. Keep it under 155 characters so it is not cut, and make it different from the title rather than a restatement of it.

RESULT

A result that explains itself in the list, which lifts clicks without touching rankings at all.

This note is an illustration of the format, not a finding about this site. It shows how a note of this kind reads before it is pointed at a real client.

Nothing on the page is shaped like a question, so assistants cannot quote it

SAMPLE NOTE

Next up

AEO

WHAT

There is no FAQ block, no question shaped heading and no short self contained answer anywhere on the page.

WHY

A growing share of the people who would like Vie will never see a list of blue links. They will ask an assistant how to plan meals around what is on sale, and it will answer out of pages that have a question as a heading and a complete answer in the next forty words. Pages written only as marketing prose do not get quoted, because there is nothing liftable in them.

HOW

Add five real questions with real answers, each answer complete on its own without the paragraph above it. How does Vie know what is on sale. Which stores. What if my store is not covered. Do I have to cook all five. What does it cost. Mark them up as FAQPage so the structure is machine readable, and keep the answers honest about current limits.

RESULT

A page an assistant can quote accurately, and a set of answers that also removes the five objections that stop a signup.

This note is an illustration of the format, not a finding about this site. It shows how a note of this kind reads before it is pointed at a real client.

Three thousand recipes and no structured data on any of them

[SAMPLE NOTE](#)[Next up](#)[SEO](#)

WHAT

The recipe pages carry no Recipe markup: no ingredients, no times, no yield, no ratings in a form a search engine reads.

WHY

Recipe markup is the single largest lever this archive has, and it is the one piece of SEO where the format is fixed and the payoff is visible. Without it the recipes compete as plain pages against sites whose entries show a photo, a time and a star rating in the results. The archive is the moat and it is currently invisible.

HOW

Generate the markup from the parsed recipe data rather than writing it by hand. The fields are already in `recipes.json`: ingredients, instructions, times, yield. Emit it per recipe at build time, and skip any recipe missing a time rather than guessing one.

RESULT

Recipes eligible for rich results, generated from the data you already have rather than maintained as a second copy by hand.

This note is an illustration of the format, not a finding about this site. It shows how a note of this kind reads before it is pointed at a real client.

An apostrophe in the rotator that should not be there

[SAMPLE NOTE](#)[Polish](#)[Copy](#)

WHAT One of the rotating lines uses it's where its belongs.

WHY On a page selling care and craft, a misplaced apostrophe is read as a signal about everything else. It is a small thing that gets noticed by exactly the sort of reader who would otherwise be impressed.

HOW Fix the line, then add the pair to the spelling check that already runs in the suite, alongside the British spellings it is catching now.

RESULT One fewer thing to notice, and a check that keeps it that way.

This note is an illustration of the format, not a finding about this site. It shows how a note of this kind reads before it is pointed at a real client.

14

The photographs ship at the size they came off the camera

SAMPLE NOTE

Polish

Performance

WHAT

Several images are served at their full dimensions and scaled down by the browser, rather than being resized and served in a modern format.

WHY

These photographs are the proof that a real person cooks this, so they have to stay beautiful. But on a phone on a weak connection the hero can be most of what loads first, and a hero that arrives late undoes the impression it was there to make.

HOW

Serve each one at the size it is actually displayed, in AVIF with a JPEG fallback, and keep the explicit width and height so nothing jumps while it loads. The hero keeps fetchpriority high. The rest stay lazy.

RESULT

The same photographs, arriving fast enough to do their job on the connection most people will be on.

This note is an illustration of the format, not a finding about this site. It shows how a note of this kind reads before it is pointed at a real client.

What I need from you

1

The Resend domain verification

SPF, DKIM and DMARC on getvie.com. The records are yours to add and DNS has lead time, so this one starts first.

2

A decision on price

One number. Everything downstream of the payroll waits on it.

3

The Netlify token scopes

So the sales writer can store a week's prices.

4

Which stores you will promise

Kroger and Harris Teeter are confirmed working. Anything beyond that is a claim we cannot currently keep.

the other half

The Vie Coop

The site is the shop window. The Coop is the back room: the week, the list, the archive, the subscribers, and the pieces that are still being built, all on one page with no menus to hunt through.

Open the Coop

This week

the plan, the list, the swaps

The archive

3,000 recipes, searchable

The list

subscribers and what they open

The build

what is live, what is next