

Enabling Accessibility Monitoring — Full User Journey

End-to-end walkthrough of turning on automatic scan monitoring for a site in the WebAbility dashboard, captured against the merged #486 + #488 + #489 build with real production data.

TEST ACCOUNT

sidharth.dogfood@webability. dogfood-sumo-b.com

DOMAIN UNDER TEST

BUILD UNDER TEST

PR #486 + #488 + #489
merged, local dev server
proxied to prod API

API TARGET

https://api.webability.io (real prod data,
real DB writes)

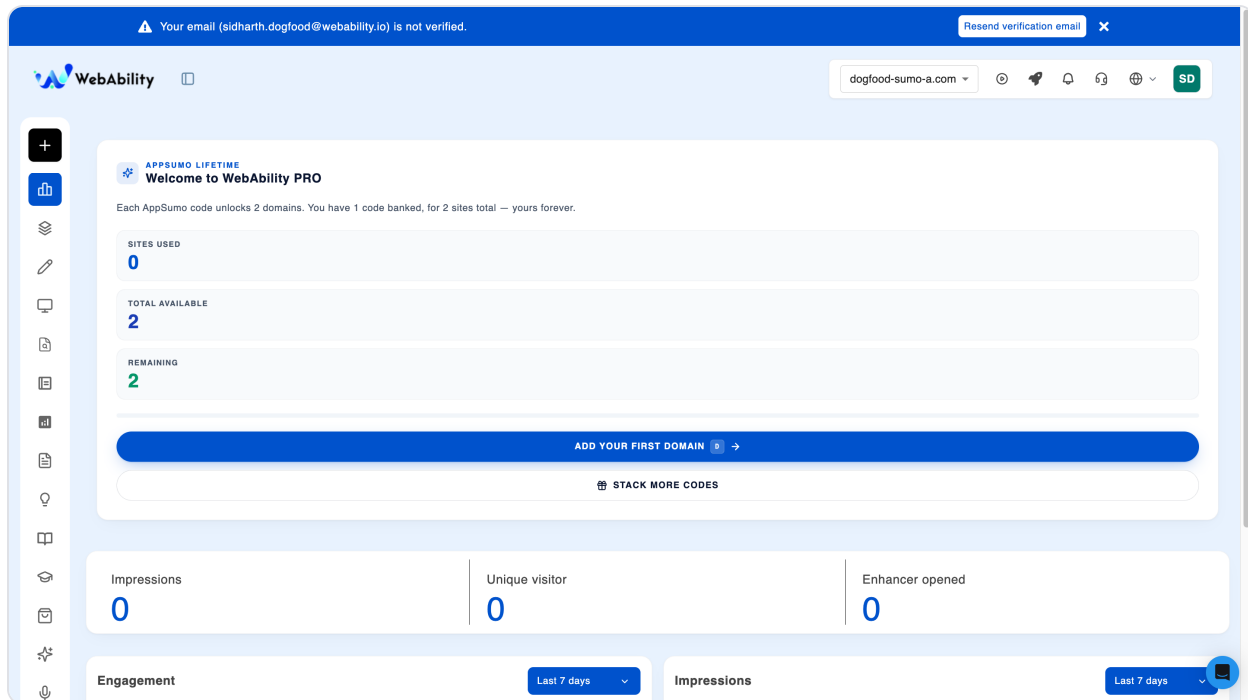
CAPTURED

2026-07-18/19, real browser screenshots, no
mockups

The Journey — Step by Step

1 Dashboard landing page

The user's first view after logging in — before touching anything. The active site selector shows dogfood-sumo-a.com; there is no visible entry point to monitoring settings from this screen.



2 Sidebar navigation to "My sites"

The user clicks "My sites" in the left sidebar (now shown active/highlighted) and lands on the sites list, which shows both licensed domains for this account.

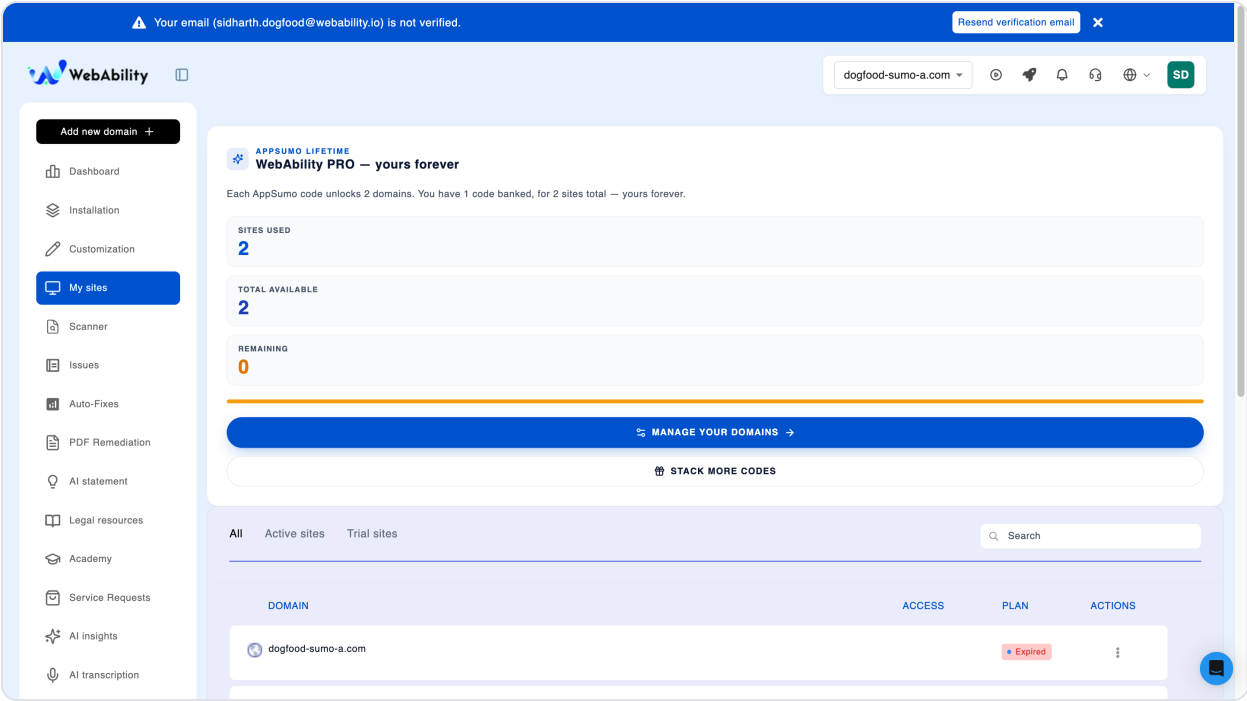
The screenshot displays the WebAbility dashboard. At the top, a blue header bar contains a notification: "Your email (sidharth.dogfood@webability.io) is not verified." with a "Resend verification email" link. The left sidebar features a "My sites" button, which is highlighted in blue. The main content area shows the "WebAbility PRO — yours forever" section, indicating that each AppSumo code unlocks 2 domains. It displays "SITES USED: 2", "TOTAL AVAILABLE: 2", and "REMAINING: 0". Below this, there are buttons for "MANAGE YOUR DOMAINS" and "STACK MORE CODES". A table at the bottom lists domains, with the first entry being "dogfood-sumo-a.com" under the "DOMAIN" column, and "Expired" under the "ACCESS" column. The table has columns for "DOMAIN", "ACCESS", "PLAN", and "ACTIONS".

DOMAIN	ACCESS	PLAN	ACTIONS
dogfood-sumo-a.com	Expired		

3

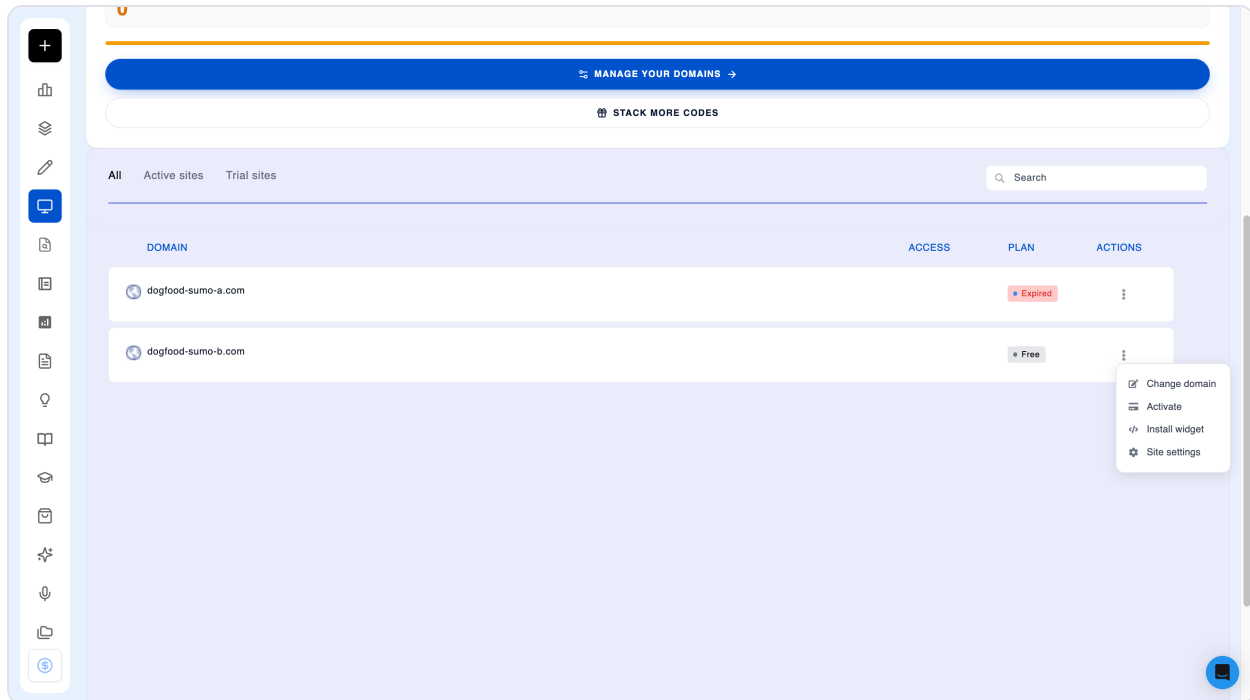
My Sites list — dogfood-sumo-b.com row, actions menu closed

The default state of the site row for dogfood–sumo–b.com in the sites table. The : actions menu button is visible but not yet opened.



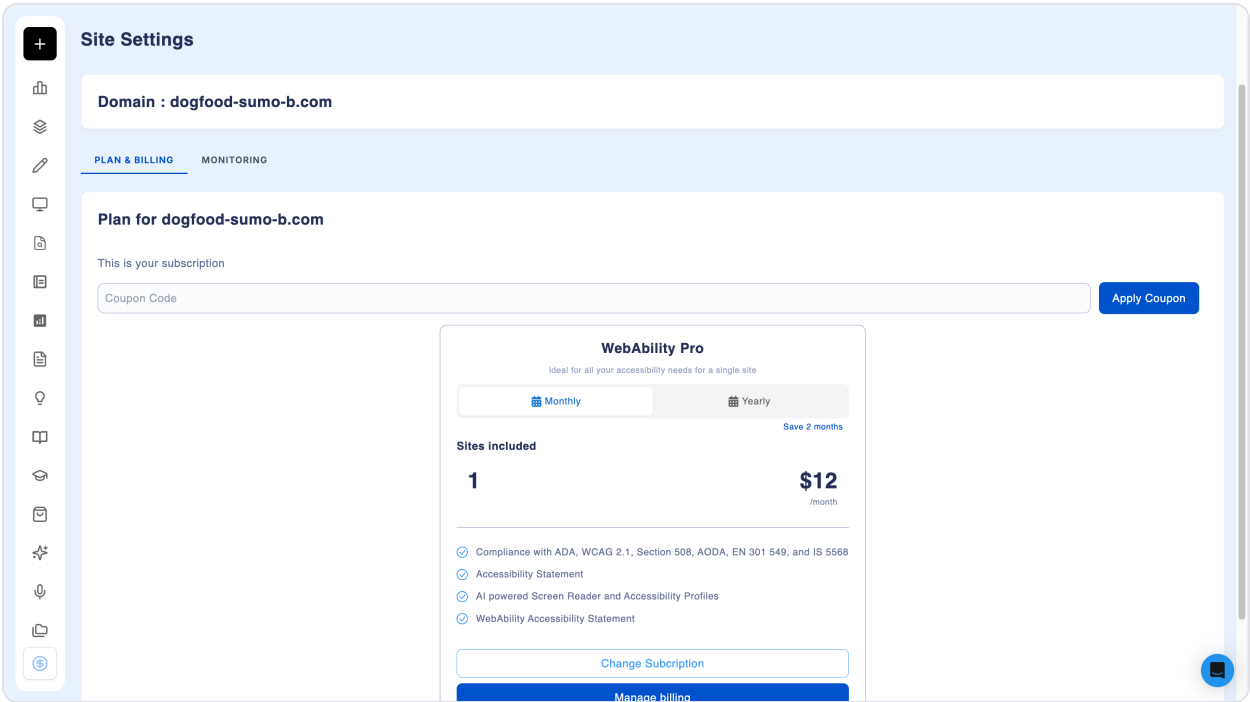
4 Actions menu opened

Clicking the  button opens the actions menu. "Site settings" is one option among several (Change domain / Activate or Buy License / Install widget) — this is the only entry point into monitoring configuration from the sites list. PR #488's change is exactly this menu item.



5 Site Settings → "Plan & Billing" tab (default)

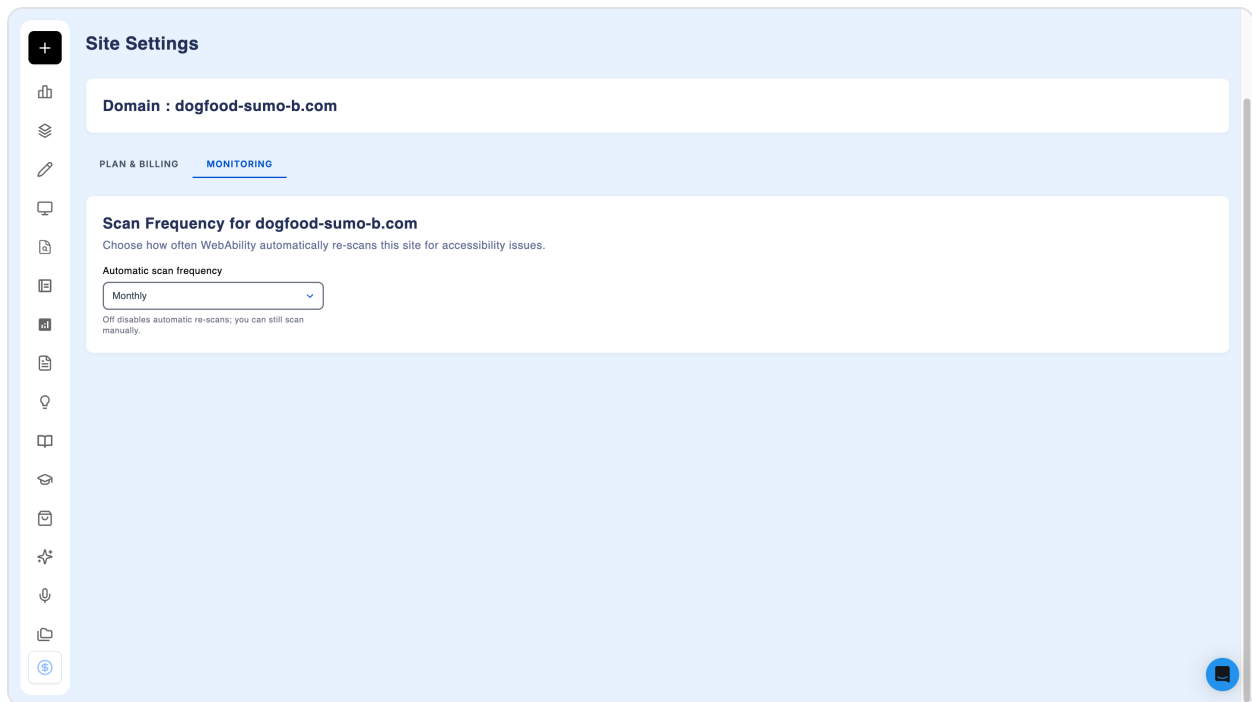
Clicking "Site settings" lands on the Site Settings page for dogfood-sumo-b.com. "Plan & Billing" is the default selected tab — subscription/checkout details, not monitoring.



6

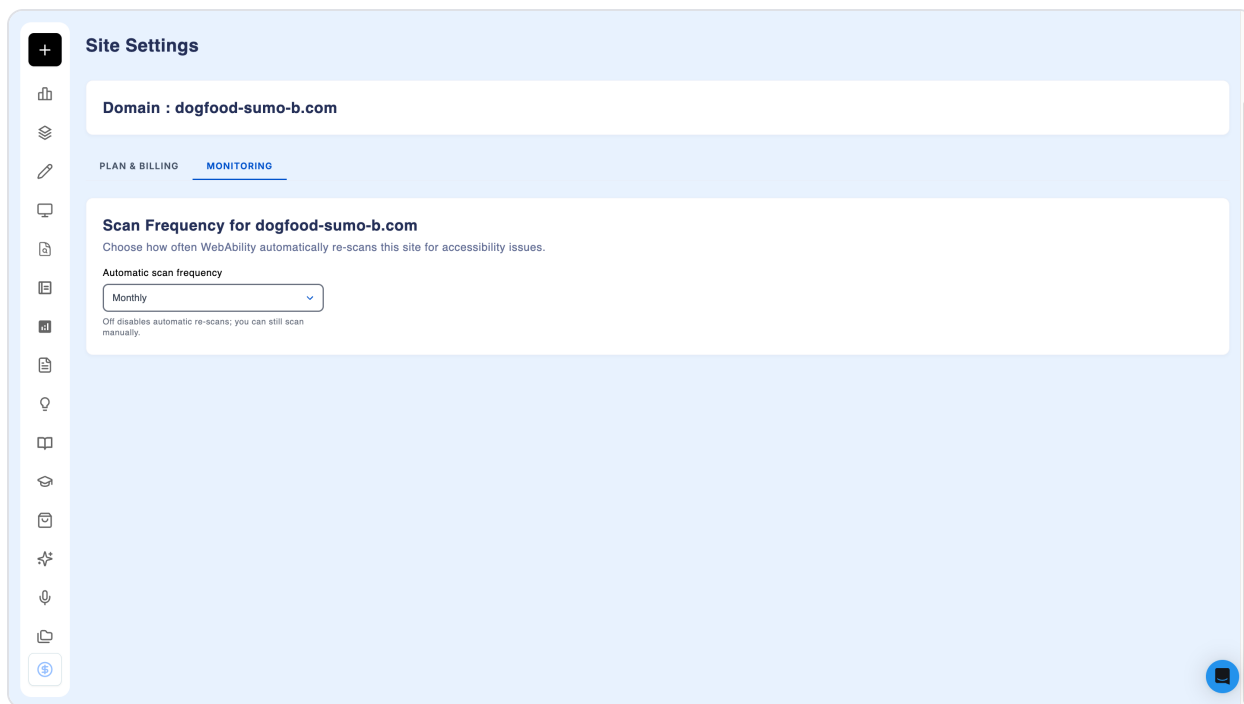
Clicking the "Monitoring" tab

The user switches to the second tab, "Monitoring" (added by PR #489, which split what used to be one combined Plan+Monitoring page into two distinct tabs).



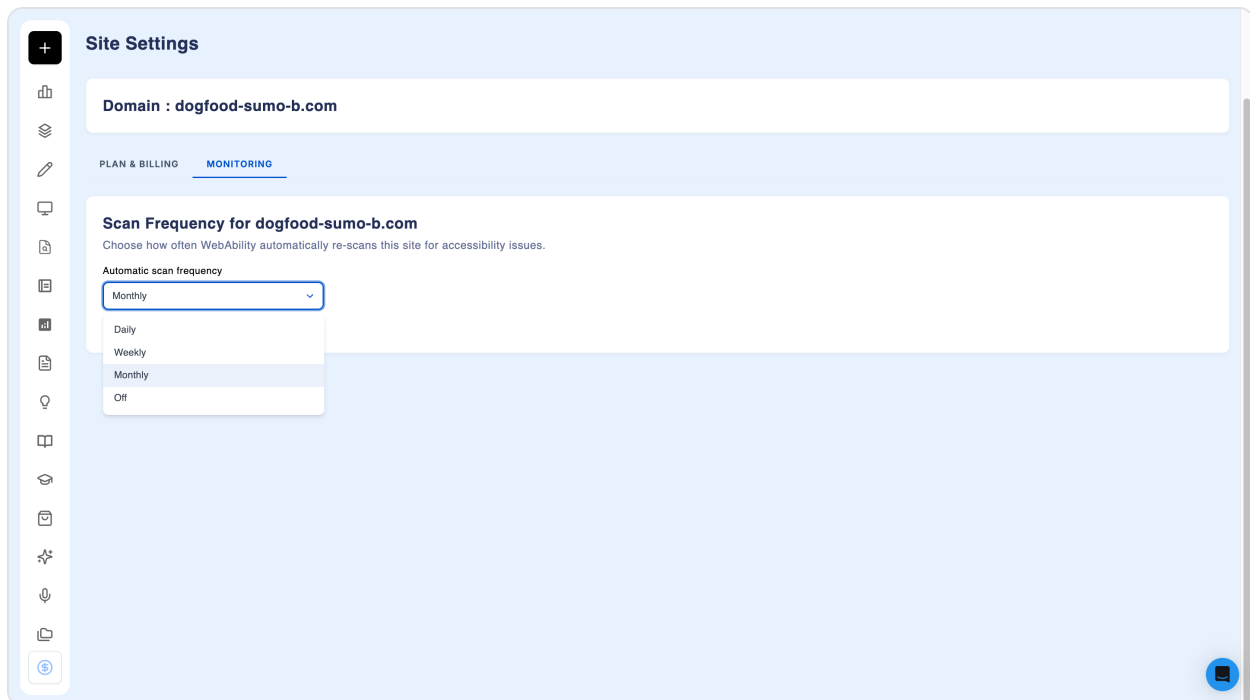
7 Monitoring tab — state before any change

The current (pre-change) state of the Monitoring tab for dogfood-sumo-b.com. The frequency dropdown already shows **Monthly** as the live setting for this account.



8 Opening the frequency dropdown

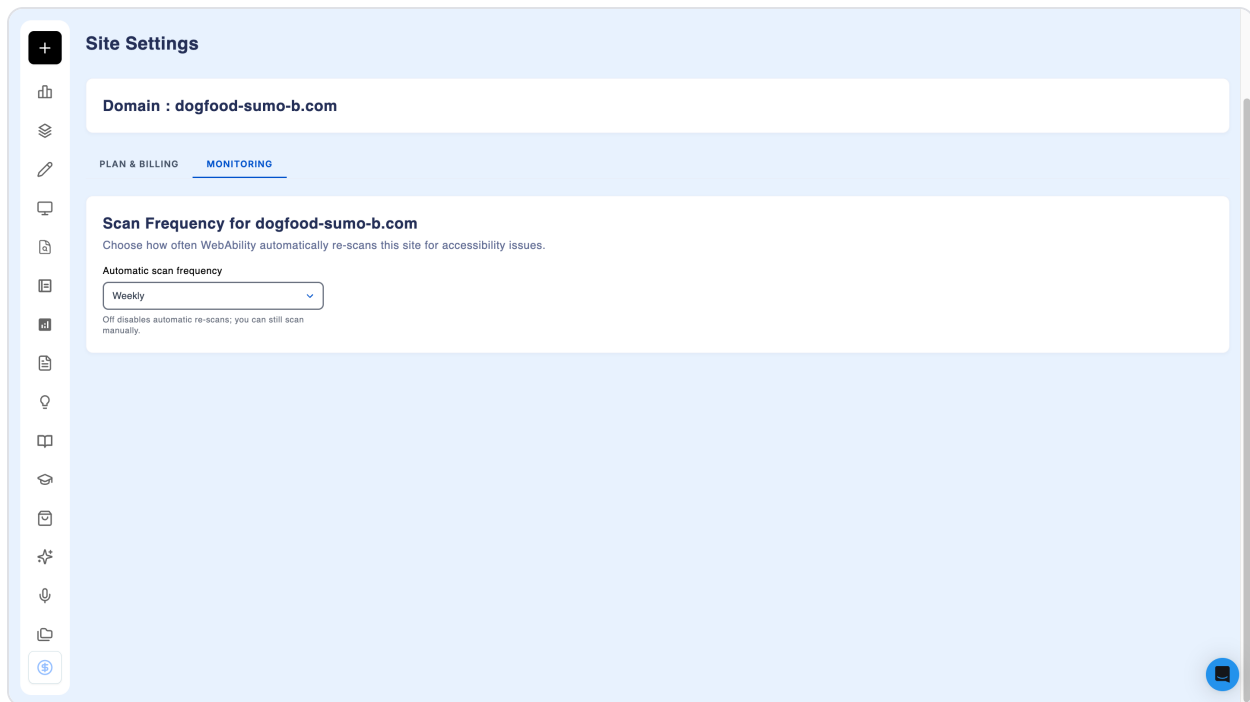
The dropdown opens to show all four available options: Daily, Weekly, Monthly, Off. Note there is no "Hourly" option (removed in commit 02ebc9bc, folded into #486) and — relevant to the open question below — no option anywhere in this menu that mentions page count or crawl scope.



9

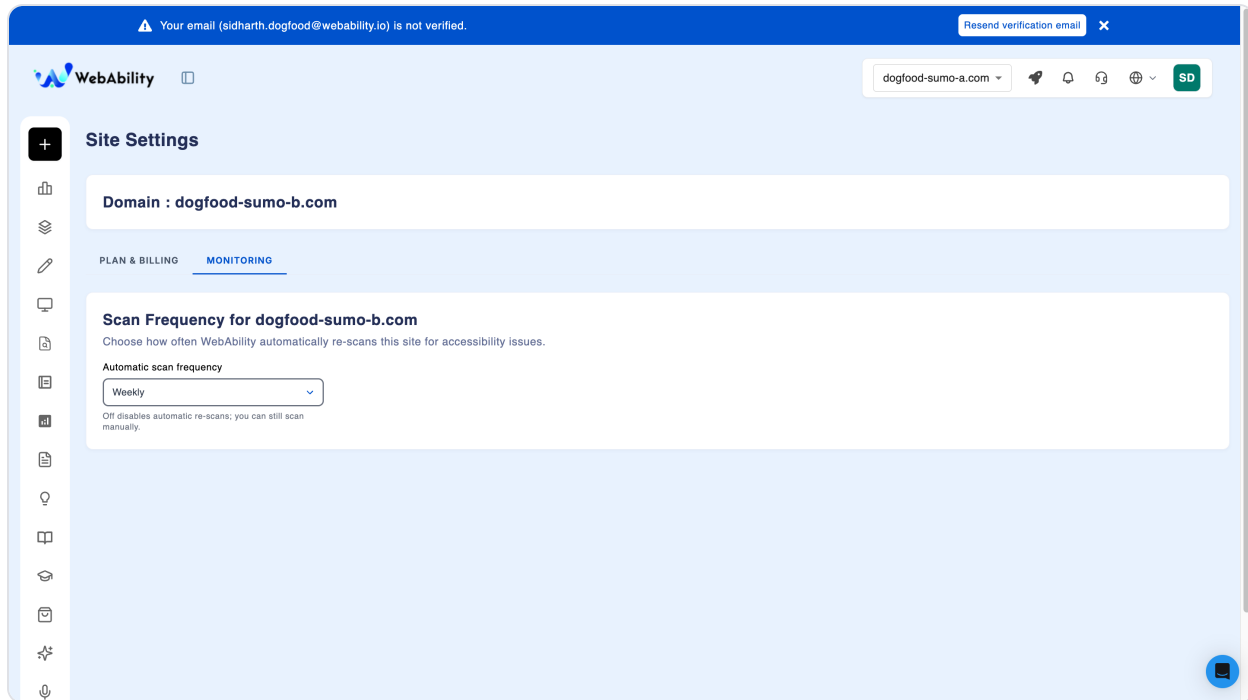
Selecting "Weekly" — save-success toast

Selecting Weekly triggers the `updateSiteMonitoringFrequency` mutation against the real production API. A toast confirms: "Scan frequency set to weekly for this site."



10 Reload — new value persisted VERIFIED REAL STATE

After a full page reload (fresh network-only query, not a cached client state), the Monitoring tab still shows **Weekly** — confirming the frequency is a real row in the database, not just optimistic UI feedback. The value was reset back to Monthly afterward to leave the account in its original state.



Open Product Question: Whole-Site vs. Single-Page Monitoring

FACTUAL FINDING — CURRENT DEPLOYED BEHAVIOR

Automatic monitoring scans only ONE page — the site's root/homepage URL — not the whole site. This is a full-crawl-vs-single-page distinction that exists in the codebase (a `fullSiteScan` boolean flag), and the scheduled monitoring job does not set it.

Code path, traced end to end:

```
api/jobs/scheduledMonitoring.ts:136
  const report: any = await fetchAccessibilityReport({ url: site.url, prior
  // no `fullSiteScan` passed → undefined → falsy

api/services/accessibilityReport/accessibilityReport.service.ts:920
  export const fetchAccessibilityReport = async (options: { url: string; us

api/services/accessibilityReport/accessibilityReport.service.ts:635
  const accessibilityResult = await getAccessibilityInformationPally(format
```

```
api/helpers/accessibility.helper.ts:541
  const scanType = fullSiteScan ? 'full-site' : 'single-page'
  // scheduledMonitoring.ts's call resolves scanType = 'single-page'

api/helpers/accessibility.helper.ts:590
  console.log(`... ${fullSiteScan ? 'Full-site axe scan (discover URLs, par
```

The `fullSiteScan` flag and the whole-site crawler it activates (`api/services/axe/fullSiteAxeScan.service.ts`, URL discovery + parallel workers) do exist in the codebase and are used elsewhere in the product (e.g. on-demand scans from the Scanner page). They are simply never invoked by the scheduled monitoring cron — `site.url` is scanned as a single page, every time, on whatever cadence (Daily/Weekly/Monthly) the user picks.

UI SURFACE CHECK

Confirmed by reading

`app/src/containers/SiteDetail/MonitoringFrequencySetting.tsx`: the only control exposed is the frequency dropdown (Daily/Weekly/Monthly/Off, screenshot Step 8). The panel's copy is: *"Choose how often WebAbility automatically re-scans this site for accessibility issues."* There is no mention of "page" vs. "site", no scope toggle, no page-count indicator, and no link to what URL(s) actually get scanned. **This is not currently a user-facing choice anywhere in the UI** — confirmed true by direct inspection of the merged component and the live dropdown captured above.

FOR SIDHARTH TO DECIDE — NOT A RECOMMENDATION BEING MADE HERE

The current behavior (homepage-only, every cadence) may or may not match what "monitoring a site" should mean to a paying customer. Open questions, surfaced without a proposed answer:

- Should scheduled monitoring use `fullSiteScan: true` (multi-page crawl) instead of a single-page scan, at least for some frequencies/plans?
- If the scope stays single-page, should the UI say so explicitly (e.g. "we re-check your homepage every [frequency]") so customers aren't assuming whole-site coverage they aren't getting?
- If whole-site monitoring becomes an option, should it be a toggle on this same panel, a plan-tier gate, or a separate feature entirely — and what's the cost/scan-time tradeoff given full-site scans use a materially longer timeout and parallel workers?

