

Feedbox Technical Specification v2.0

Mobile-First Client-Side Intelligence Engine for Blogger, WordPress, and GitHub Pages

System Engineering Specification

August 2026

1 System Overview

A **Feedbox** is a lightweight, asynchronous, zero-dependency client-side RSS aggregation dashboard designed to run in site footers. It ingests 100 domain-specific RSS/Atom feeds via client-side cross-origin proxying (`rss2json`), rendering real-time headline streams organized into 8 thematic pillars.

1.1 Core Architectural Guardrails

- **Core Web Vitals Integrity:** Execution is gated by an `IntersectionObserver` (200px bottom threshold) to eliminate unneeded API calls on initial page load.
- **Mobile Viewport Strict Containment:** Prevents horizontal overflow blowout and iOS Safari auto-zoom via CSS containment (`max-width: 100% !important, flex-wrap: wrap, font-size: 16px` on inputs).
- **SEO Link Equity Preservation:** External hyperlinking enforces `rel="nofollow noopener noreferrer"`.

2 Universal Code Template (Blogger, GitHub Pages & Static Web)

This single-file solution embeds directly into Blogger HTML/JavaScript gadgets, static static sites, or GitHub Pages (`index.html`).

```
1 <!-- UNIVERSAL FEEDBOX CONTAINER -->
2 <style>
3   :root {
4     --fb-bg: #090a0f;
5     --fb-card: #121520;
6     --fb-border: #1e2638;
7     --fb-accent: #3b82f6;
8     --fb-text: #f8fafc;
9     --fb-sub: #94a3b8;
10  }
11
12  #feedbox-app, #feedbox-app * {
13    box-sizing: border-box !important;
14    max-width: 100% !important;
15  }
16
17  #feedbox-app {
18    background: var(--fb-bg);
19    color: var(--fb-text);
20    font-family: -apple-system, BlinkMacSystemFont, "Segoe UI", Roboto, sans-serif;
21    padding: 12px;
22    border-radius: 8px;
23    border: 1px solid var(--fb-border);
```

```

24     width: 100% !important;
25     margin: 16px 0;
26     overflow: hidden !important;
27 }
28
29 .fb-header {
30     display: flex;
31     justify-content: space-between;
32     align-items: center;
33     border-bottom: 1px solid var(--fb-border);
34     padding-bottom: 8px;
35     margin-bottom: 10px;
36 }
37
38 .fb-title {
39     font-size: clamp(0.75rem, 3.5vw, 1rem);
40     font-weight: 700;
41     margin: 0;
42     display: flex;
43     align-items: center;
44     gap: 6px;
45 }
46
47 .fb-dot {
48     width: 8px;
49     height: 8px;
50     background: var(--fb-accent);
51     border-radius: 50%;
52     box-shadow: 0 0 6px var(--fb-accent);
53 }
54
55 .fb-controls { display: flex; flex-direction: column; gap: 8px; margin-bottom: 10px;
56 }
57
58 .fb-search {
59     width: 100% !important;
60     background: var(--fb-card);
61     border: 1px solid var(--fb-border);
62     color: var(--fb-text);
63     padding: 8px 10px;
64     border-radius: 6px;
65     font-size: 16px !important; /* Disables iOS auto-zoom */
66     min-height: 40px;
67 }
68
69 .fb-pillars { display: flex !important; flex-wrap: wrap !important; gap: 5px !
70             important; }
71
72 .fb-pill {
73     background: var(--fb-card);
74     border: 1px solid var(--fb-border);
75     color: var(--fb-sub);
76     padding: 5px 9px;
77     border-radius: 12px;
78     font-size: 0.7rem;
79     cursor: pointer;
80     flex-grow: 1;
81     text-align: center;
82 }
83
84 .fb-pill.active { background: var(--fb-accent); color: #fff; font-weight: 700; }
85
86 .fb-scroll-area { max-height: 55vh; overflow-y: auto; overflow-x: hidden; }
87
88 .fb-grid { display: grid !important; grid-template-columns: 100% !important; gap: 8
89           px; }

```

```

87
88 .fb-card {
89     background: var(--fb-card);
90     border: 1px solid var(--fb-border);
91     border-radius: 6px;
92     padding: 10px;
93     display: flex;
94     flex-direction: column;
95     justify-content: space-between;
96 }
97
98 .fb-tag { font-size: 0.6rem; color: var(--fb-accent); text-transform: uppercase;
99     margin-bottom: 4px; }
100 .fb-card-title {
101     font-size: 0.82rem;
102     color: var(--fb-text);
103     text-decoration: none;
104     font-weight: 600;
105     line-height: 1.3;
106     display: -webkit-box;
107     -webkit-line-clamp: 2;
108     -webkit-box-orient: vertical;
109     overflow: hidden;
110 }
111
112 .fb-meta {
113     font-size: 0.65rem;
114     color: var(--fb-sub);
115     display: flex;
116     justify-content: space-between;
117     margin-top: 8px;
118     padding-top: 6px;
119     border-top: 1px solid rgba(255,255,255,0.05);
120 }
121
122 @media (min-width: 640px) {
123     .fb-pill { flex-grow: 0; }
124     .fb-grid { grid-template-columns: repeat(auto-fill, minmax(260px, 1fr)) !important
125         ; gap: 12px; }
126 }
127 </style>
128
129 <div id="feedbox-app">
130     <div class="fb-header">
131         <h2 class="fb-title"><span class="fb-dot"></span> FEEDBOX // LIVE INTELLIGENCE
132         STREAM</h2>
133     </div>
134     <div class="fb-controls">
135         <input type="text" id="fb-search" class="fb-search" placeholder="Search 100 feeds
136         by keyword...">
137         <div class="fb-pillars" id="fb-pillars">
138             <button class="fb-pill active" data-pillar="ALL">ALL PILLARS</button>
139             <button class="fb-pill" data-pillar="PILLAR_1">PILLAR ONE</button>
140             <button class="fb-pill" data-pillar="PILLAR_2">PILLAR TWO</button>
141             <button class="fb-pill" data-pillar="PILLAR_3">PILLAR THREE</button>
142             <button class="fb-pill" data-pillar="PILLAR_4">PILLAR FOUR</button>
143             <button class="fb-pill" data-pillar="PILLAR_5">PILLAR FIVE</button>
144             <button class="fb-pill" data-pillar="PILLAR_6">PILLAR SIX</button>
145             <button class="fb-pill" data-pillar="PILLAR_7">PILLAR SEVEN</button>
146             <button class="fb-pill" data-pillar="PILLAR_8">PILLAR EIGHT</button>
147         </div>
148     </div>
149     <div class="fb-scroll-area">
150         <div class="fb-grid" id="fb-grid">
151             <div style="text-align:center; padding: 16px;" id="fb-status">Scroll to footer
152             to connect live feeds...</div>

```

```

148     </div>
149   </div>
150 </div>
151
152 <script>
153 (function() {
154   // 100 FEED ENDPOINTS ARRAY PLACEHOLDER
155   const FEED_SOURCES = [
156     { name: "Source 001", url: "https://example.com/feed1.xml", category: "PILLAR_1"
157     },
158     { name: "Source 002", url: "https://example.com/feed2.xml", category: "PILLAR_2"
159     },
160     { name: "Source 100", url: "https://example.com/feed100.xml", category: "PILLAR_8"
161     }
162   ];
163
164   let articlesAcc = [];
165   let currentPillar = "ALL";
166   let initialized = false;
167
168   const grid = document.getElementById('fb-grid');
169   const searchInput = document.getElementById('fb-search');
170   const status = document.getElementById('fb-status');
171   const app = document.getElementById('feedbox-app');
172
173   async function fetchFeed(src) {
174     try {
175       const r = await fetch('https://api.rss2json.com/v1/api.json?rss_url=${
176         encodeURIComponent(src.url)}');
177       const d = await r.json();
178       if (d.status === 'ok') {
179         return d.items.map(i => ({
180           title: i.title, link: i.link, pubDate: new Date(i.pubDate), source: src.name
181           , category: src.category
182         }));
183       }
184     } catch(e) {}
185     return [];
186   }
187
188   async function streamAll() {
189     const BATCH_SIZE = 4;
190     for (let i = 0; i < FEED_SOURCES.length; i += BATCH_SIZE) {
191       const batch = FEED_SOURCES.slice(i, i + BATCH_SIZE);
192       const res = await Promise.all(batch.map(s => fetchFeed(s)));
193       articlesAcc = [...articlesAcc, ...res.flat()].sort((a,b) => b.pubDate - a.
194         pubDate);
195       if (status) status.innerText = `Streaming... (${Math.min(i + BATCH_SIZE,
196         FEED_SOURCES.length)}/${FEED_SOURCES.length})`;
197       render();
198     }
199     if (status) status.style.display = 'none';
200   }
201
202   function render() {
203     const q = searchInput.value.toLowerCase();
204     const filtered = articlesAcc.filter(a =>
205       (currentPillar === "ALL" || a.category === currentPillar) &&
206       (a.title.toLowerCase().includes(q) || a.source.toLowerCase().includes(q))
207     );
208
209     grid.innerHTML = '';
210     filtered.forEach(a => {
211       const card = document.createElement('div');
212       card.className = 'fb-card';

```

```

206     const dt = isNaN(a.pubDate) ? 'Recent' : a.pubDate.toLocaleDateString(undefined,
{month:'short', day:'numeric'});
207     card.innerHTML = `
208         <div>
209             <div class="fb-tag">${a.category} // ${a.source}</div>
210             <a href="${a.link}" class="fb-card-title" target="_blank" rel="nofollow
noopener noreferrer">${a.title}</a>
211         </div>
212         <div class="fb-meta"><span>${dt}</span><span>Read &rarr;</span></div>
213     `;
214     grid.appendChild(card);
215 });
216 }
217
218 const observer = new IntersectionObserver((entries) => {
219     if (entries[0].isIntersecting && !initialized) {
220         initialized = true;
221         streamAll();
222     }
223 }, { rootMargin: "200px" });
224
225 observer.observe(app);
226
227 document.getElementById('fb-pillars').addEventListener('click', (e) => {
228     if (e.target.classList.contains('fb-pill')) {
229         document.querySelectorAll('.fb-pill').forEach(b => b.classList.remove('active'))
230         ;
231         e.target.classList.add('active');
232         currentPillar = e.target.getAttribute('data-pillar');
233         render();
234     }
235 });
236
237 searchInput.addEventListener('input', render);
238 })();
</script>

```

Listing 1: Universal Feedbox Implementation Template

3 WordPress Deployment Architecture

To deploy a Feedbox seamlessly on WordPress without third-party plugins, register a custom shortcode in the active theme's `functions.php` file.

```

1 // Register [feedbox] Shortcode for WordPress
2 function render_feedbox_shortcode() {
3     ob_start();
4     ?>
5     <!-- Paste Universal Feedbox HTML/CSS/JS Code Here -->
6     <div id="feedbox-app">...</div>
7     <?php
8     return ob_get_clean();
9 }
10 add_shortcode('feedbox', 'render_feedbox_shortcode');

```

Listing 2: WordPress Shortcode Implementation (`functions.php`)

3.1 WordPress Usage

1. Paste the shortcode code block into your theme's `functions.php` or a snippet plugin.
2. Insert `[feedbox]` into any Gutenberg Custom HTML Block, Footer Widget, or Elementor container.

4 GitHub Pages Standalone Deployment

Feedboxes can be hosted on GitHub Pages as standalone widget endpoints or iFrames.

1. Create a new repository (e.g., `user/intel-feedbox`).
2. Save the Universal Feedbox code as `index.html`.
3. Enable GitHub Pages via **Settings** → **Pages** → **Branch: main / root**.
4. Embed the hosted URL anywhere via an responsive iFrame:

```
1 <iframe
2   src="https://username.github.io/intel-feedbox/"
3   style="width: 100%; height: 600px; border: none; overflow: hidden;"
4   scrolling="no"
5   loading="lazy">
6 </iframe>
```

Listing 3: iFrame Embed Specification for External Sites

5 Prompt Engineering Blueprint: Automated Feedbox Generation via Gemini

To generate a fully populated, 100-source Feedbox tailored to any specific blog URL using Gemini's live web search capabilities, execute the prompt below.

5.1 Copy-Pasteable Gemini Prompt Specification

```
1 ACT AS AN EXPERT FRONT-END DEVELOPER AND OSINT RESEARCHER.
2
3 TASK:
4 1. Perform web searches to thoroughly analyze the target blog URL: [
5   INSERT_BLOG_URL_HERE].
6 2. Identify the core niche, target audience, and 8 distinct thematic pillars relevant
7   to this site.
8 3. Conduct web searches to discover 100 ACTIVE, HIGH-AUTHORITY RSS/ATOM FEED URLs
9   across these 8 pillars (approx. 12-13 working feed endpoints per pillar).
10 4. Output a single, complete, production-ready Feedbox code block (HTML + CSS + JS)
11    based on the Feedbox Specification.
12
13 STRICT CODE CONSTRAINTS:
14 - CSS Containment: Set max-width: 100% !important, box-sizing: border-box !important
15   on all elements.
16 - Input Zoom Fix: Search inputs must have font-size: 16px !important to prevent iOS
17   Safari auto-zoom.
18 - Mobile Wrapping: Category buttons (.fb-pillars) MUST use flex-wrap: wrap !important
19   to prevent horizontal page overflow.
20 - Core Web Vitals: Wrap execution in an IntersectionObserver with lazy-loading.
21 - Link Equity Protection: External links MUST use rel="nofollow noopener noreferrer".
22 - Data Array: Populate the FEED_SOURCES JavaScript array with all 100 verified RSS
23   sources categorized accurately under the 8 pillars.
24
25 OUTPUT FORMAT:
26 Provide the full, un-truncated, self-contained HTML/CSS/JS block ready to paste
27 directly into Blogger HTML/JS gadget or WordPress Gutenberg HTML block.
```

Listing 4: Master AI Prompt Template