

# Search for Organics: Complete System Specification & Reference Implementation Manual

Autonomous Replication Blueprint for AI-Grounded Organic Marketplaces, Auditing, and Verification Engines

Architecture and Engineering Group

Version 4.2.0 — August 2026

## Abstract

This specification details the end-to-end architecture, computational models, database schemas, API contracts, AI grounding prompts, and UI workflows for **Search for Organics**. It covers all 24 application sub-modules, including the Organic Purity Scoring Engine, OCR Label Ingredient Scanner, Supply Chain Route Optimizer, Farm ROI Calculator, Zero-Leakage Dietary & Allergen Guard, Community Hub, and Automated Business/Marketing Intelligence Generators.

## 1 System Architecture & Runtime Environment

### 1.1 Execution Topology

The platform operates as a hybrid Single-Page Application (SPA) backed by an Express 4 middleware layer compiled to a single CommonJS bundle via **esbuild** and deployed to Google Cloud Run containers.

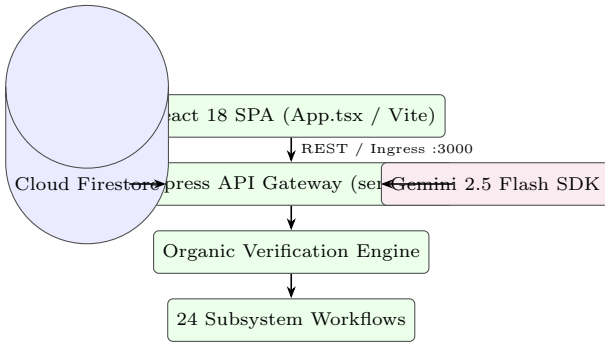


Figure 1: System Subsystem Topology

### 1.2 Port Binding & Container Proxy Rules

- **Ingress Port:** Port 3000 exclusively.
- **Host Binding:** 0.0.0.0 (required for Cloud Run reverse proxies).
- **Build Target:** vite build && esbuild server.ts --bundle --platform=node --format=cjs --packages=external --sourcemap --outfile=dist/server.cjs.

## 2 Core Modules & Tab Navigation Matrix

The application state machine is controlled by the primary enumeration:

```
type TabType = 'search' | 'community' | 'scanner' | 'audit' | 'calc' | 'workshops' | 'route' | 'businessPlan' | 'marketingKit' | 'seoReport' | 'intelligence' | 'info' | 'faq' | 'chat' | 'privacy' | 'ownership' | 'sales' | 'affiliate' | 'terms' | 'sitemap' | 'about' | 'contact' | 'certification' | 'partnership';
```

### 2.1 Module Breakdown

1. **Search (search):** Real-time organic product finder with geolocated distance filters, vegan alternatives, and certification audits.
2. **Community Hub (community):** Social forum, organic recipe catalog, local produce swap marketplace, and farmer workshops.
3. **OCR Label Scanner (scanner):** Optical character recognition on packaging labels to detect hidden synthetic additives.
4. **Brand Audit Tool (audit):** Deep corporate ownership and greenwashing evaluation pipeline.
5. **ROI Transition Calculator (calc):** 5-year financial modeling for conventional farms transitioning to organic soil management.
6. **Workshops & Events (workshops):** Educational webinars, permaculture masterclasses, and RSVP ticketing.
7. **Supply Chain Route Planner (route):** Multi-stop logistics planner calculating fuel savings and localized delivery waypoints.
8. **Business Plan Generator (businessPlan):** Automated financial projections and grant drafting for organic startups.
9. **Marketing Kit Builder (marketingKit):** Social media graphics, copy generators, and USDA label templates.
10. **SEO Health Report (seoReport):** Semantic analysis of agricultural e-commerce stores.
11. **Market Intelligence (intelligence):** Live pricing indices, wholesale commodity trends, and consumer demand trackers.
12. **Consumer AI Chat (chat):** Interactive certified nutritionist and agronomist chatbot.

13. **Standards Directory** (info): Comprehensive registry of 45+ international certification bodies.
14. **Ownership Radar** (ownership): Parent conglomerate mapping showing mega-corporation food acquisitions.
15. **Affiliate Engine** (affiliate): Referral link generator and commission payout calculator.
16. **Certification Portal** (certification): Step-by-step audit preparation tool for USDA/CCOF certification.

### 3 Mathematical Formulations

#### 3.1 Organic Purity Score Algorithm

For any product  $x$ , the Purity Score  $\mathcal{P}(x) \in [0, 100]$  is computed deterministically:

$$\mathcal{P}(x) = \text{clamp}_{[0,100]} \left( \mathcal{B} + \sum_{i=1}^{N_c} w_i C_i - \sum_{j=1}^{N_a} \delta_j A_j - \sum_{k=1}^{N_p} \gamma_k P_k \right) \quad (1)$$

where:

- $\mathcal{B} = 50.0$  (Baseline neutral score).
- $C_i \in \{0, 1\}$  denotes verified certifications:  
 $w_{\text{USDA 100\% Organic}} = 30, \quad w_{\text{Demeter Biodynamic}} = 35$   
 $w_{\text{Regenerative Organic}} = 35, \quad w_{\text{Non-GMO Verified}} = 15$   
 $w_{\text{EU Organic}} = 28, \quad w_{\text{Fair Trade USA}} = 10$
- $A_j$  denotes banned synthetic additives (e.g., Titanium Dioxide, Sodium Benzoate, BHT, High Fructose Corn Syrup) with penalty  $\delta_j \in [15, 30]$ .
- $P_k$  denotes plastic packaging index ( $\gamma_k = 10$  for non-biodegradable plastics).

#### 3.2 Farm Transition ROI Model

The 5-year net profit  $V(t)$  for a transitioning farm with acreage  $A$  is:

$$V(t) = A \cdot [Y(t) \cdot (P_{\text{base}} \cdot (1 + \mu(t))) - (C_{\text{input}}(t) + C_{\text{cert}}(t))] \quad (2)$$

where:

- $Y(t)$ : Crop yield at year  $t$ .  $Y(1) = 0.75Y_0$ ,  $Y(2) = 0.85Y_0$ ,  $Y(5) = 1.10Y_0$  (soil microbiome recovery curve).
- $\mu(t)$ : Organic price premium ( $\mu(1.3) = 0$  during transition;  $\mu(4.5) \in [0.25, 0.45]$  post-certification).
- $C_{\text{input}}(t)$ : Input costs, declining by  $\sim 35\%$  over 5 years as synthetic fertilizer is phased out.

#### 3.3 Supply Chain Route Haversine Optimization

For local multi-farm sourcing, distance  $D$  between coordinate pairs  $(\phi_1, \lambda_1)$  and  $(\phi_2, \lambda_2)$  is calculated via the Haversine metric:

$$a = \sin^2 \left( \frac{\Delta \phi}{2} \right) + \cos(\phi_1) \cos(\phi_2) \sin^2 \left( \frac{\Delta \lambda}{2} \right) \quad (3)$$

$$D = 2R \cdot \arcsin \left( \min(1, \sqrt{a}) \right), \quad R = 6371 \text{ km} \quad (4)$$

## 4 TypeScript Domain Entities

```

1 // --- Product & Marketplace Entity ---
2 export interface Product {
3   id: string;
4   name: string;
5   brand: string;
6   description: string;
7   certifications: string[];
8   price: string;
9   shippingPrice: string;
10  directBuyUrl: string;
11  vendor: string;
12  isAvailable: boolean;
13  score: number;
14  confidenceScore: number;
15  verificationSource: string;
16  detectedAdditives?: string[];
17  containsAllergens?: string[];
18  isVegan?: boolean;
19  isRegenerative?: boolean;
20  purityExplanation?: string;
21  coordinates?: { lat: number; lng: number };
22  farmOrigin?: {
23    name: string;
24    location: string;
25    distanceMiles?: number;
26  };
27 }
28
29 // --- Vegan Alternative Item ---
30 export interface VeganAlternative {
31   name: string;
32   brand: string;
33   category: string;
34   matchReason: string;
35   healthBenefit: string;
36   score: number;
37   certifications: string[];
38 }
39
40 // --- Zero-Leakage Private Dietary Profile ---
41 export interface UserPrivateSettings {
42   userId: string;
43   isVegan: boolean;
44   suggestVeganAlternatives: boolean;
45   allergies: string[]; // 14 Major Allergens
46   customAllergens: string[];
47   avoidIngredients: string[];
48   dietaryNotes: string;
49   updatedAt: string;
50 }
51
52 // --- Community Post & Recipe Entity ---
53 export interface CommunityPost {
54   id: string;
55   authorId: string;
56   authorName: string;
57   authorAvatar?: string;
58   title: string;
59   content: string;
60   category: 'general' | 'recipe' | 'swap' | 'question' | 'alert';
61   upvotes: number;
62   commentCount: number;
63   tags: string[];
64   createdAt: string;
65   recipeMeta?: {
66     prepTime: string;
67     cookTime: string;
68     servings: number;
69     ingredients: string[];
70     isVegan: boolean;
71     certificationsUsed: string[];
72   };
73 }

```

| Method | Endpoint                    | Auth Required | Rate Limit | Functional Description                                           |
|--------|-----------------------------|---------------|------------|------------------------------------------------------------------|
| GET    | /api/health                 | No            | None       | Container readiness and DB connection probe.                     |
| POST   | /api/search                 | Optional      | 30 req/min | AI search grounding with location and dietary filters.           |
| POST   | /api/scan-label             | Optional      | 15 req/min | OCR image analysis and chemical additive detection.              |
| POST   | /api/audit-brand            | Optional      | 20 req/min | Parent corporation lookup and greenwashing audit.                |
| POST   | /api/plan-route             | Optional      | 20 req/min | Multi-waypoint farm logistics and carbon footprint optimization. |
| POST   | /api/generate-business-plan | User          | 5 req/min  | Generates full financial, operational, and marketing plan.       |
| POST   | /api/generate-marketing-kit | User          | 5 req/min  | Generates campaign copy, social captions, and flyer assets.      |
| POST   | /api/chat                   | Optional      | 20 req/min | Streaming nutritional advice and agronomist guidance.            |
| GET    | /api/intelligence/trends    | No            | 60 req/min | Aggregated commodity pricing and search demand indices.          |

Table 1: Express API Gateway Route Specification

## 5 Express Backend Gateway Endpoints

## 6 AI Grounding & Prompt Specifications

### 6.1 Product Search Grounding Contract

```

1 const prompt = `
2 System: You are the Search For Organics AI Grounding
  Engine.
3 Query: "${query}"
4 Context: Location=${location.city}, Lat/Lng=(${
  location.lat}, ${location.lng})
5 Preferences: VeganOnly=${isVegan}, FlaggedAllergens=[${
  allergies.join(',')}]
6
7 Generate a strictly valid JSON object conforming to:
8 {
9   "products": [
10     {
11       "id": "string",
12       "name": "string",
13       "brand": "string",
14       "description": "string",
15       "certifications": ["USDA Organic", ...],
16       "price": "$X.XX",
17       "shippingPrice": "string",
18       "directBuyUrl": "https://...",
19       "vendor": "string",
20       "isAvailable": true,
21       "score": 95,
22       "confidenceScore": 0.98,
23       "verificationSource": "string",
24       "containsAllergens": [],
25       "isVegan": true,
26       "isRegenerative": true,
27       "purityExplanation": "string"
28     }
29   ],
30   "isNonVeganSearch": boolean,
31   "nonVeganReason": "string",
32   "veganAlternatives": [
33     {
34       "name": "string",
35       "brand": "string",
36       "category": "string",
37       "matchReason": "string",
38       "healthBenefit": "string",
39       "score": 98,
40       "certifications": ["USDA Organic"]
41     }
42   ]
43 }
```

### 6.2 OCR Label Scanner Prompt Contract

```

1 const ocrPrompt = `
2 Analyze this product label image:
3 1. Extract complete text including ingredient list and
  nutrition facts.
4 2. Flag any synthetic additives, high-risk pesticides,
  synthetic dyes, artificial sweeteners, or hidden
  GMO derivatives.
5 3. Identify official certification seals (USDA Organic
  seal, Non-GMO Project butterfly, CCOF, Demeter).
6 4. Compute the Organic Purity Score (0-100) with
  detailed line-by-line ingredient breakdown.
7 `;
```

## 7 Cloud Security Rules Specification

```

1 rules_version = '2';
2 service cloud.firestore {
3   match /databases/{database}/documents {
4     function isAuthenticated() { return request.auth
      != null; }
5     function isOwner(uid) { return isAuthenticated()
      && request.auth.uid == uid; }
6
7     match /users/{userId} {
8       allow read: if true;
9       allow write: if isOwner(userId);
10
11       // Strict Zero-Leakage Dietary Settings
12       match /private_settings/dietary {
13         allow read, write: if isOwner(userId);
14       }
15     }
16
17     match /posts/{postId} {
18       allow read: if true;
19       allow create: if isAuthenticated() && request.
      resource.data.authorId == request.auth.uid;
20       allow update, delete: if isOwner(resource.data.
      authorId);
21
22       match /comments/{commentId} {
23         allow read: if true;
24         allow create: if isAuthenticated();
25         allow update, delete: if isOwner(resource.data
      .authorId);
26       }
27     }
28   }
29 }
```

```

26     }
27   }
28
29   match /workshops/{workshopId} {
30     allow read: if true;
31     allow write: if isAuthenticated() && request.
32       auth.token.admin == true;
33   }
34 }

```

## 8 Replication & Bootstrap Protocol

### 8.1 Package Dependencies (package.json)

```

1 {
2   "name": "search-for-organics",
3   "version": "4.2.0",
4   "dependencies": {
5     "@google/genai": "^1.27.0",
6     "canvas-confetti": "^1.9.4",
7     "clsx": "^2.1.1",
8     "d3": "^7.9.0",
9     "express": "^4.21.2",
10    "firebase": "^10.14.1",
11    "lucide-react": "^0.474.0",
12    "motion": "^12.4.7",
13    "react": "^18.3.1",
14    "react-dom": "^18.3.1",
15    "react-markdown": "^9.0.3",
16    "recharts": "^2.15.1",
17    "tailwind-merge": "^3.0.1"
18  },
19  "devDependencies": {
20    "@types/node": "^22.13.1",
21    "@types/react": "^18.3.18",
22    "esbuild": "^0.25.0",
23    "tailwindcss": "^4.0.0",
24    "tsx": "^4.19.2",
25    "typescript": "^5.7.3",
26    "vite": "^6.1.0"
27  }
28 }

```

### 8.2 Compilation Commands

```

1 # Step 1: Install exact package hierarchy
2 npm install --legacy-peer-deps
3
4 # Step 2: Build frontend static bundle and server.cjs
5 npm run build
6
7 # Step 3: Run the verified CommonJS server
8 node dist/server.cjs

```