



FUNCTIONAL REQUIREMENT DOCUMENT

Project Name: EcoEats Local Sourcing Initiative

Version: 1.0

Prepared By: Riya Thorve (Business Analyst)

Company : CityLights Software Solutions Inc

Date: 09 th October 2025



1. INTRODUCTION

1.1 Purpose

This document specifies the functional requirements for the **EcoEats Local Sourcing Initiative**, which enables restaurants to tag dishes as *locally sourced*, users to filter sustainable options, and the system to validate sourcing details.

It serves as a blueprint for developers, testers, and business stakeholders to ensure consistent understanding of the system's behavior.

1.2 Intended Audience

- **Business Stakeholders** – For feature understanding and validation.
- **Developers** – For implementation details.
- **QA/Test Team** – For creating test cases.
- **Product/Operations** – For rollout and reporting setup.

1.3. References

- Project Charter – EcoEats Local Sourcing Initiative (v1.0)
- Product Requirement Document (PRD) – EcoEats (Oct 2025)
- EcoEats Design System
- Google Maps API Documentation

1.4.Glossary

Term	Definition
Locally Sourced Dish	Dish prepared using ingredients sourced within 100 miles of the restaurant.
Tagging	Process by which restaurants mark menu items as “Locally Sourced.”
Local Filter	A toggle option allowing users to view dishes tagged as locally sourced.
CMS	EcoEats Content Management System used by restaurant owners to manage menus.
Verification API	Internal API that checks the supplier location and validates it as local.



2. Overall Description

2.1 Product Perspective

This feature will be an enhancement to the existing EcoEats ecosystem. It introduces new backend APIs, CMS tagging modules for restaurants, and front-end updates for end-users to discover locally sourced dishes.

2.2 Product Features (Feature List)

Sr	Feature	Priority
1	Restaurant Dish Tagging	High
2	Local Filter & Discovery	High
3	Checkout & Impact Confirmation	High

2.3. Project Scope

In Scope:

- Tagging system for restaurants.
- Local sourcing verification using Google Maps API.
- Filter integration for user app.
- Sustainability analytics dashboard.

Out of Scope:

- Blockchain integration for supply chain validation.
- Physical supplier verification visits.

2.4. Operating Environment

1. **Browser:** Chrome, Safari, Edge
2. **Mobile App:** iOS and Android

2.5. Assumptions & Dependencies

1. Active Internet connection required.
2. Supplier location and ingredient data available.
3. EcoEats CMS and APIs are functional during rollout.



3. System Features

Feature 1. Restaurant Dish Tagging (FR-1)

Requirement ID: FR-1

Title: Restaurant Dish Tagging

Actors:

- Restaurant Owner
- EcoEats CMS System
- Verification API

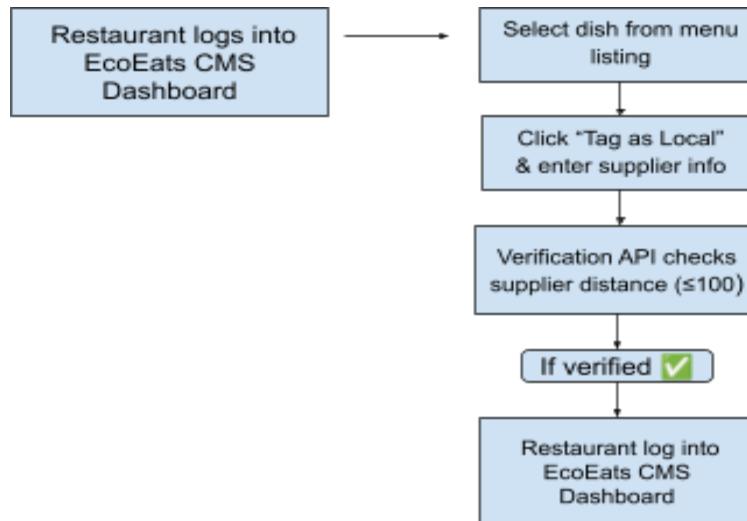
Preconditions:

1. Restaurants must be logged in to EcoEats CMS.
2. Menu data already uploaded.
3. Verification API connection active.

Acceptance Criteria:

- Restaurant can tag any dish as *Locally Sourced*.
- System validates supplier location (≤ 100 miles).
- Badge  appears on approved dishes in CMS and user app.
- Audit data logged successfully.

Business Flow Diagram :



Feature 2. Local Filter & Discovery (FR-2)

Requirement ID: FR-2

Title: Local Filter & Discovery

Actors:

- End User
- EcoEats Mobile/Web App
- Backend Search API

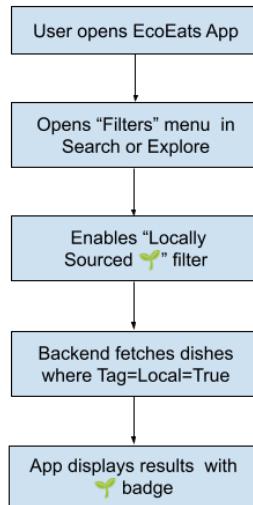
Preconditions:

1. User must be logged in to the EcoEats app.
2. Local dishes are already tagged and synced in the database.

Acceptance Criteria:

- User can enable “Locally Sourced” filter.
- Search results show only tagged dishes.
- Load time < 2 seconds.
- Filter state persists until cleared.

Business Workflow Diagram (Textual Format)



Feature 3. Checkout & Impact Confirmation (FR-3)

Requirement ID: FR-3

Title: Checkout and Impact Confirmation

Actors:

- Customer
- Payment Gateway
- Order Management System

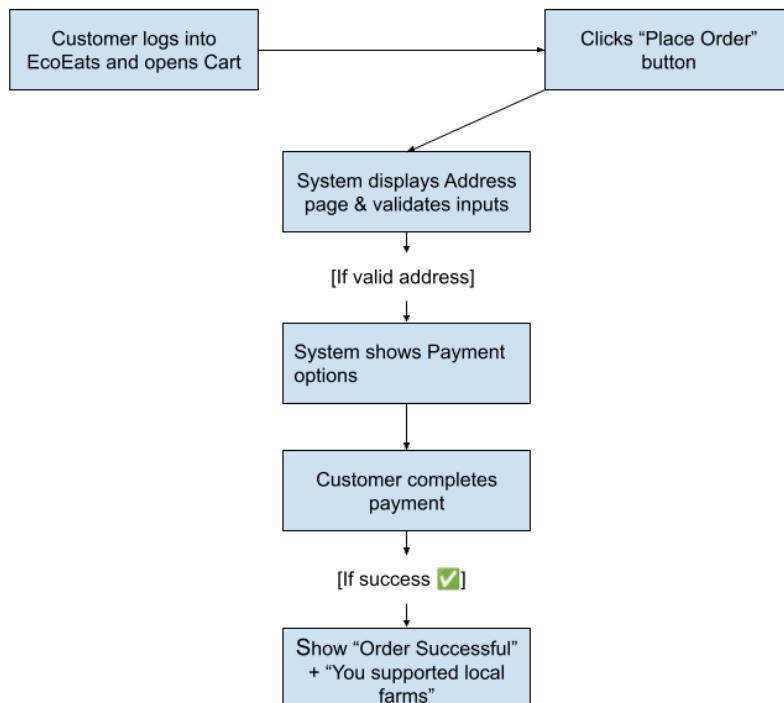
Preconditions:

1. Customer logged into EcoEats app.
2. Cart contains at least one item.

Acceptance Criteria:

- Customer completes checkout successfully.
- Payment gateway response captured.
- Confirmation message displays sustainability impact.

Business Workflow Diagram (Textual Format)



User Roles & Permissions

- Only **logged-in customers** can place orders.
- **Admins** can access audit and transaction logs.

Notifications & Alerts

Type	Trigger	Message
Success	Order confirmation	Dear <Customer Name>, your order #<Order No> was successful.
Failure	Payment declined	Dear <Customer Name>, your order #<Order No> failed. Refund (if any) within 7 days.

Audit Trail & Logging

Captured details:

- Customer ID
- Ordered items (Dish ID, Quantity)
- Address details
- Timestamp and payment mode

Cases & Error Messages

Case	System Action	Message
Items out of stock	Block checkout	“One or more items are out of stock.”
Quantity > 10	Block checkout	“Maximum 10 units allowed per dish.”
Invalid Full Name	Halt progress	“Full Name is invalid or blank.”
Invalid Address Line 1	Halt progress	“Address Line 1 is invalid or blank.”
Payment timeout (> 3 min)	Abort transaction	“Payment in progress. Please check status later.”

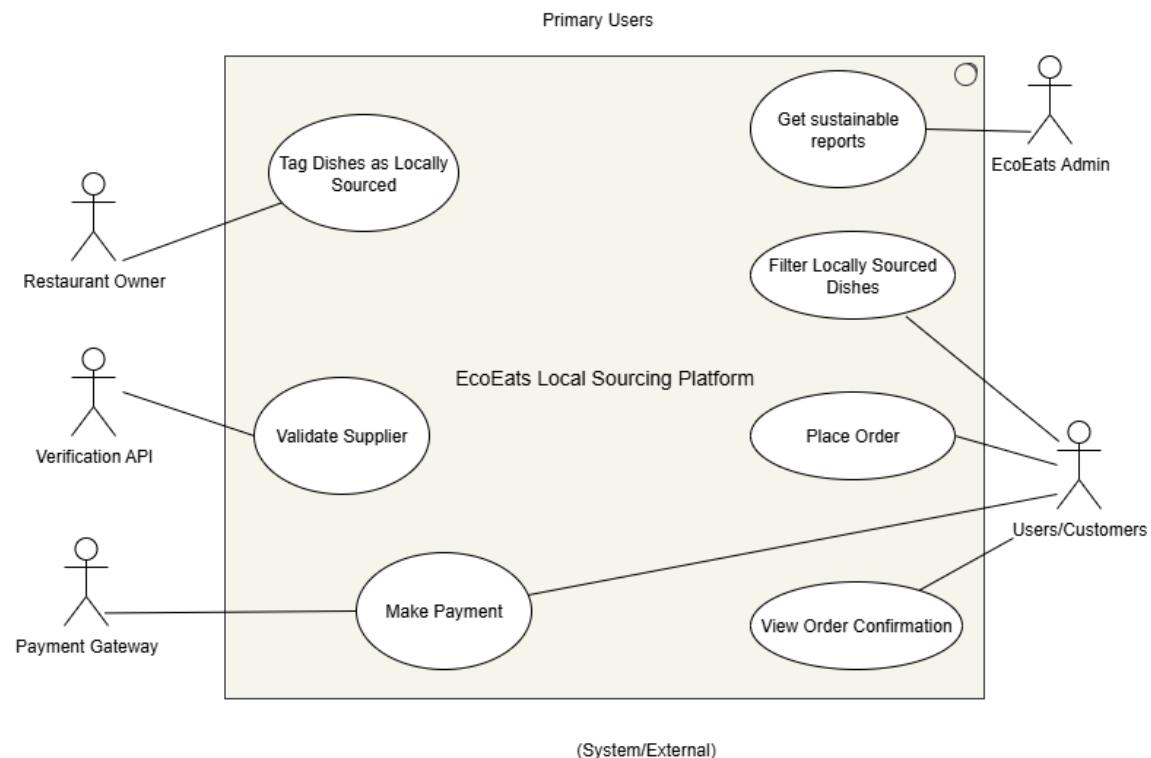
Data Entry & Field Validation (Address Screen)

Field Name	Purpose	M/O/C	Validation
Full Name	Recipient Name	M	Min 3 chars, Max 50, A-Z only
Address Line 1	Address	M	Min 1 char, Max 50, Alphanumeric
Address Line 2	Additional Address Info	O	Min 1 char, Max 50

Exception / Alternate Flows

- If payment gateway response is not received within 3 minutes → display: “Payment is in progress. Please check the order status later.”
- If verification API fails → allow tagging in pending verification state.
- If network timeout → prompt user to retry.

Use Case Diagram





4. Non-Functional Requirements

Category	Requirement
Performance	Response time < 2 seconds; supports 50 K concurrent users.
Security	OAuth 2.0 authentication; AES-256 encryption; role-based access
Usability	Uniform across web & mobile; 2-click filter experience.
Reliability	99.5 % uptime SLA during peak hours.



5. Appendix

1. Success Metrics (from PRD):

- 30% of restaurants adopt local tagging within 6 months.
- 20% of users use “Local” filters regularly.
- 4.5+/5 user satisfaction (NPS) for sustainability.

2. Future Enhancements:

- Blockchain traceability for supplier verification.
- Loyalty program for users who frequently order local dishes.