International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 06, ACSCON - 2026

EcoSort AI: A C2B Mobile Ecosystem for AI-Driven E-Waste Management

Tushar Kumar
Dept. of Computing Technologies, School of Computing SRM Institute of Science & Technology Kattankulathham, Chennai, India
tushaarxr@gmail.com

Tanishq Agrawal
Dept. of Computing Technologies, School of Computing SRM Institute of Science & Technology Kattankulathham, Chennai, India mahamiatanishq@gmail.com

Brindha R
Dept. of Computing Technologies, School of Computing SRM Institute of Science & Technology Kattankulathham, Chennai, India brindhar@srmist.edu.in

K. M. Umamaheswari
Dept. of Computing Technologies, School of Computing SRM Institute of Science & Technology Kattankulathham, Chennai, India umamahek@srmist.edu.in

N. A. S. Vinoth
Dept. of Computing Technologies, School of Computing SRM Institute of Science & Technology Kattankulathham, Chennai, India. vinoths@srmist.edu.in

R. K. Pongiannan
Dept. of Computing Technologies, School of Computing SRM Institute of Science & Technology Kattankulathham, Chennai, India. pongiank@srmist.edu.in

***Abstract*—EcoSort AI is a cross-platform mobile application (iOS, Android, Web) that addresses the global electronic waste (e-waste) crisis by integrating AI-powered image recognition, a Consumer-to-Business (C2B) marketplace, and a corporate exchange programme aggregator into a single, unified platform. The AI engine employs Google Gemini 2.5 Flash with a structured JSON prompt chain to classify e-waste items, infer material composition, assign a hazard level (Low / Medium / High), flag data-security risk, and generate device-specific disposal guidance — all from a single photograph. Beyond identification, EcoSort AI introduces a dual-role marketplace in which consumers create multi-image listings and admin-verified vendors express interest through a lead-generation pipeline enforced by PostgreSQL Row Level Security (RLS). An Exchange Programs Hub aggregates trade-in offers from major brands (Apple, Samsung, Amazon, Flipkart) within a single browsable portal. An offline-first cache manager with a persistent sync queue ensures functionality in low-connectivity environments. The platform comprises 35 screens across three route groups, a companion web-based admin panel for vendor verification and listing moderation, and a comprehensive input-validation and rate-limiting utility layer. Built on React Native 0.81.5 / Expo SDK 54, Clerk JWT authentication, and Supabase (PostgreSQL) with per-table RLS policies, EcoSort AI delivers a production-grade architecture aligned with United Nations Sustainable Development Goal 12 (Responsible Consumption and Production).**

***Index Terms*—E-waste management, mobile application, Gemini 2.5 Flash, C2B marketplace, circular economy, Supabase RLS, LMM prompt engineering, offline-first, SDG 12, React Native**

I. INTRODUCTION

The exponential rise in consumer electronics has led to a parallel surge in electronic waste (e-waste), now recognised as the fastest-growing domestic waste stream globally. The Global E-Waste Monitor 2020 reported over 53.6 million metric tonnes generated in 2019, a figure projected to surpass 74 million tonnes by 2030 [1]. Improper disposal of such waste, laden with toxic materials such as mercury, cadmium, and lead, poses severe environmental and public-health hazards [11]. Despite growing awareness, individuals frequently face four distinct knowledge gaps: (i) they do not know what hazardous materials their devices contain; (ii) they lack guidance on safe disposal; (iii) they cannot locate certified recycling facilities; and (iv) they are unaware that their old devices may carry residual monetary value.

EcoSort AI is designed as a closed-loop platform that addresses all four gaps within a single mobile application. The AI engine, built on Google Gemini 2.5 Flash [2] with a structured multi-field JSON prompt chain, analyses a photograph of any electronic item and returns a complete `ScanResult` — including item type, material composition, hazard level, data-security risk flag, recycling value assessment, and step-by-step disposal instructions. The C2B Marketplace then converts this awareness into action by connecting consumers with admin-verified vendors, while the Exchange Programs Hub surfaces corporate trade-in offers from Apple, Samsung, Amazon, and Flipkart in a single aggregated portal. A GPS-driven recycling centre locator using the Haversine formula closes the loop by directing users to certified drop-off points.

The platform comprises 35 screens across three route groups (authentication, consumer, vendor), a companion web-based admin panel for vendor verification and listing moderation,

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 06, ACSCON - 2026

and an offline-first architecture with a persistent sync queue—making the application viable in low-connectivity environments typical of developing-country e-waste scenarios.

II. Novel Contributions

Prior work on e-waste mobile platforms reveals a consistent segmentation: identification tools lack marketplaces, marketplace tools lack AI, and none integrate offline resilience, dual-role access control, and quantified environmental tracking into a single application. EcoSort AI makes the following distinct contributions:

1) **Structured JSON Prompt Chain for LMM-Based E-Waste Classification.** We design and implement a domain-specific two-schema prompt architecture (Algorithm 1) that constrains Google Gemini 2.5 Flash via `responseMimeType: "application/json"` and temperature 0.6 to produce structured `ScanResult` objects. The schema differentiates e-waste items (returning eight typed fields including `hazardLevel`, `dataSecurityRisk`, and `recyclingValue`) from non-e-waste items (returning a `funMessage`). A fallback retry mechanism handles non-compliant outputs. This design eliminates the need for labelled training datasets and domain-specific CNN retraining while generalising across the full variability of consumer electronics.
2) **Trust-Tiered C2B Marketplace with RLS-Enforced Data Isolation.** We architect a two-sided marketplace comprising a 1,331-line service layer (`marketplaceService.ts`) and five interconnected database tables (`listings`, `vendors`, `vendor_interests`, `transactions`, `exchange_offers`). Vendor onboarding requires admin verification via a companion web panel; the `is_verified` flag, set exclusively by an admin role, gates all vendor-exclusive database operations. Per-table RLS policies enforce strict data isolation: consumers may only mutate their own records, and only verified vendors may `INSERT` into `vendor_interests`.
3) **Offline-First Architecture with Persistent Sync Queue.** We implement a singleton `OfflineCache` manager (529 lines) using `AsyncStorage` as the persistence backend, with a `NetworkManager` observer that automatically drains a persisted `SyncQueueItem[]` on reconnection. Cache capacity limits (100 scan results, 500 recycling centres, 200 disposal guides) and a 24-hour expiry prevent stale data without requiring permanent connectivity — a design choice with direct relevance to e-waste management in semi-urban and rural contexts.
4) **Exchange Programs Hub.** We design and implement a centralised aggregation layer (`exchangeService.ts`, 359 lines) that surfaces corporate trade-in offers with automatic `view_count` and `click_count` analytics. Contextual offer retrieval (`getOffersForItem()`) matches a scanned item's category and brand to relevant trade-in programmes, creating a direct pipeline from AI identification to monetised disposal.
5) **Comprehensive Input Validation and Rate-Limiting Layer.** A standalone validation module (`validation.ts`, 292 lines) provides 12 typed validators covering email, password, image URI, Base64 encoding, GPS coordinates, scan result schema, and an in-memory rate limiter — a security contribution absent from all comparable e-waste applications identified in the literature.

III. LITERATURE REVIEW AND COMPARATIVE ANALYSIS

A. The Growing Threat of E-Waste

The Global E-Waste Monitor 2020 estimated over 53.6 million metric tonnes of e-waste generated in 2019, projected to surpass 74 million tonnes by 2030 [1]. E-waste contains valuable resources such as gold and copper alongside hazardous materials including mercury and lead, making its mismanagement simultaneously an economic loss and a public-health crisis [11].

B. Current E-Waste Management Practices

Conventional e-waste management relies on manual collection, community drives, or municipal drop-off points. While Extended Producer Responsibility (EPR) legislation has been introduced globally, its on-ground execution remains inconsistent due to limited user awareness and the absence of consumer-facing financial incentives [10]. Existing platforms rarely connect consumers directly with verified recyclers through a transparent, application-level marketplace.

C. Role of AI in Waste Classification

Deep learning has been applied to classify municipal waste using Convolutional Neural Networks, notably WasteNet [7] and SmartBin [8]. However, CNN-based approaches require large, domain-specific labelled datasets which remain scarce for the heterogeneous e-waste category. Large Multimodal Models (LMMs) such as Google Gemini 2.5 Flash [2] generalise across device types, infer component-level material composition from visual cues (port types, form factor, branding), and produce natural-language disposal guidance — capabilities beyond discriminative classifiers.

D. Mobile Applications for Sustainability

Platforms such as ecoATM [9], RecycleNation, and iRecycle offer recycler directories or kiosk-based device buyback. Table I presents a structured feature comparison of EcoSort AI against these platforms and WasteNet [7]. EcoSort AI is the only platform to simultaneously provide LMM-based AI identification, hazard-level output, a two-sided verified C2B marketplace, an exchange programme aggregator, environmental impact scoring, an offline mode, and a companion admin panel.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 06, ACSCON - 2026

TABLE I
FEATURE COMPARISON OF E-WASTE PLATFORMS

Feature	EcoSort AI	ecoATM	RecycleNation	iRecycle
LMM/AI image classification	✓	✓	×	×
Hazard level & data-risk flag	✓	×	×	×
C2B marketplace (two-sided)	✓	×	×	×
Admin-verified vendor flow	✓	×	×	×
Exchange programme aggregator	✓	×	×	×
GPS recycler map locator	✓	✓	✓	✓
Environmental impact score	✓	×	×	×
Offline-first with sync queue	✓	×	×	×
Cross-platform (iOS/Android)	✓	×	✓	✓
Companion admin web panel	✓	×	×	×
Input validation & rate limit	✓	×	×	×

E. Research Gaps

No existing platform unifies LMM-based e-waste identification, a trust-tiered C2B marketplace, exchange programme aggregation, offline-first resilience, and a companion admin panel in a single mobile application. EcoSort AI is architected to fill this gap end-to-end.

IV. SYSTEM ARCHITECTURE

A. Technology Stack

EcoSort AI is built on a modern, fully open-source-compatible stack detailed in Table II. The frontend uses React Native 0.81.5 with Expo SDK 54 and file-based routing via Expo Router 6. UI components are provided by React Native Paper 5.13 (Material Design 3), with animations by React Native Reanimated 4.1. Authentication is handled by Clerk v2.19 with JWT tokens cached in Expo SecureStore. The database layer uses Supabase (PostgreSQL) with per-table RLS policies enforced via JWT claim extraction. AI processing is performed by Google Gemini 2.5 Flash via the `@google/generative-ai` SDK v0.24.1.

TABLE II
PRODUCTION TECHNOLOGY STACK — ECOSORT AI V1.0.0

Component	Technology (Version)
Mobile framework	React Native 0.81.5 / Expo SDK 54
Routing	Expo Router 6.x (file-based, typed)
Language	TypeScript 5.7+
UI library	React Native Paper 5.13 (MD3)
Animation engine	React Native Reanimated 4.1
Authentication	Clerk v2.19 (JWT + SecureStore)
Database	Supabase (PostgreSQL + RLS)
File storage	Supabase Storage (S3-compatible)
AI engine	Google Gemini 2.5 Flash
AI SDK	@google/generative-ai v0.24.1
Maps & GPS	expo-location v19.0.8
Offline storage	AsyncStorage 2.2.0
Network detection	@react-native-community/netinfo v11.4.1
Admin panel	Vanilla HTML/CSS/JS + Supabase JS

B. Three-Tier Architecture

The system follows a three-tier client-server model with a clear separation of concerns. The mobile client (React Native) communicates with Supabase over HTTPS for all database and storage operations, with every request carrying a Clerk-issued JWT that is validated at the Supabase gateway before any query executes. Image payloads are forwarded directly from the client to the Gemini 2.5 Flash endpoint as base64-encoded JPEG data, and the structured JSON response is persisted in Supabase under the authenticated user's RLS scope. GPS coordinates are resolved client-side via `expo-location` and the Haversine formula to rank nearby recycling centres without a dedicated geospatial backend service.

The provider hierarchy in the root layout (`_layout.tsx`) is: `ClerkProvider` → `ThemeProvider` → `SafeAreaProvider` → `PaperProvider` → `Slot`, ensuring authentication context, theme, and Material Design components are available to all 35 screens.

C. Navigation and Screen Architecture

The application is structured across three Expo Router route groups and 35 total screens:

- **(auth)** — Stack navigator: Login, Register, Password Reset (unauthenticated access).
- **(main)** — Tab navigator with 5 visible tabs (Scan, Repository, Recycling Centers, Marketplace, Exchange) and 2 hidden routes (Profile, Search). Accessible to authenticated consumers.
- **(vendor)** — Stack navigator: Register, Pending, Dashboard, Browse, My Interests. Accessible to authenticated vendors after admin verification.

D. Service Layer Architecture

All business logic is encapsulated in `src/api/`, following four consistent design patterns: (i) snake_case → camelCase transformation on all DB responses; (ii) graceful degradation returning `null/[]` on errors rather than throwing; (iii) fire-and-forget for non-critical increments (view counts, interest counts); and (iv) ownership verification via `.eq('user_id', userId)` on all mutation queries. Table III summarises the service files.

TABLE III
SERVICE LAYER SUMMARY

File	Lines	Responsibility
`geminiService.ts`	179	Gemini 2.5 Flash AI pipeline
`supabaseService.ts`	375	Scans, centers, e-waste items
`marketplaceService.ts`	1,331	Listings, vendors, interests
`exchangeService.ts`	359	Exchange offers & analytics
`apiUtils.ts`	76	Connectivity diagnostics

V. METHODOLOGY

A. LMM Prompt Engineering for E-Waste Classification

Rather than training a bespoke CNN, EcoSort AI exploits the zero-shot generalisation capacity of Google Gemini 2.5 Flash via a carefully engineered dual-schema prompt chain.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 06, ACSCON - 2026

This design is deliberate for three reasons: (i) labelled e-waste image datasets sufficient for CNN training are not publicly available at the required scale and diversity; (ii) LMM inference inherits Gemini's continual model updates without retraining costs; and (iii) LMMs can produce natural-language disposal guidance, data-security warnings, and recycling-value assessments in a single inference pass — outputs that discriminative classifiers cannot generate.

The system prompt instructs the model to act as "EcoSort AI", a friendly e-waste expert. Structured output is enforced via the `responseMimeType: "application/json"` parameter and a temperature of 0.6. The API call uses a maximum of 2,048 output tokens. The prompt defines two conditional response schemas:

1) **E-waste schema** (`isEWaste: true`): returns `itemType`, `materials[]`, `hazardLevel` ("low" | "medium" | "high"), `disposalMethod`, `recyclingValue`, `dataSecurityRisk` (boolean), and `confidence` ("low" | "medium" | "high").
2) **Non-e-waste schema** (`isEWaste: false`): returns a `funMessage` — a sarcastic but friendly comment (e.g., "Unless you plan to charge your phone with cake…").

Algorithm 1 formalises the complete inference pipeline.

Algorithm 1 Gemini 2.5 Flash E-Waste Classification Pipeline

Require: Image URI U, system prompt P_{sys}
Ensure: `ScanResult` object S

$B \leftarrow$ `readAsStringAsync`(U, base64)
Strip data-URI prefix from B
Construct inline image part $I_{part} \leftarrow$ {mimeType: "image/jpeg", data: B}
Set API params: `temperature`=0.6, `maxTokens`=2048, `responseMimeType`="application/json"
$R_{raw} \leftarrow$ `GeminiFlash.generateContent`(P_{sys}, I_{part})
if `isValidJSON`(R_{raw}) **then**
  $R \leftarrow$ `parseJSON`(R_{raw})
else
  $R_{raw} \leftarrow$ `GeminiFlash.generateContent`(P_{retry}, I_{part})
  $R \leftarrow$ `parseJSON`(R_{raw}); `fallbackParsed` $\leftarrow$ **true**
end if
Validate `hazardLevel` $\in$ {"low", "medium", "high"}
Validate `confidence` $\in$ {"low", "medium", "high"}
$S \leftarrow$ `buildScanResult`(R, timestamp, userId)
`supabase.insert(scans,` S`)` {RLS-scoped to user}
if offline **then**
  `OfflineCache.enqueueSyncItem`(S)
end if
return S

B. Dual-Flow C2B Marketplace

The marketplace is architected around two strictly separated roles enforced at both the application and database layers.

1) Consumer Flow: Consumers create listings with title, category, brand, model, condition ("working" | "minor_damage" | "not_working" | "for_parts"), asking price, negotiability flag, and up to multiple images uploaded to the `listing-images` Supabase Storage bucket. Listings auto-expire after 30 days (`expires_at` column). The consumer dashboard (`my-listings.tsx`) displays view counts, interest counts, and the status lifecycle: Draft → Active → Sold / Expired / Removed.

2) Vendor Flow: Vendors register by submitting business details (name, type, phone, address, service areas, categories of interest, and verification documents). Documents are uploaded to a private Supabase Storage bucket with 1-year signed URL expiry. The registration record is created with `verification_status: "pending"` and reviewed by an admin via the companion web panel. On approval, the admin sets status to `"verified"`. Only verified vendors may `INSERT` into `vendor_interests`, enforced by an RLS policy. A `UNIQUE` constraint on (`listing_id`, `vendor_id`) prevents duplicate interest submissions. Upon a consumer accepting an interest, `contact_revealed` is set to `true`, exposing the consumer's contact details to the vendor and creating a `transactions` record.

C. Exchange Programs Hub

The `exchangeService.ts` module provides contextual offer retrieval via `getOffersForItem()`, which matches a `ScanResult`'s category and brand against the `exchange_offers` table's `categories` (JSONB) and `brands_accepted` (JSONB) columns. Each offer carries `min_value`, `max_value`, `valueText` (e.g., "Up to 25,000"), `offerUrl`, and `isFeatured` flag. Click-through events are tracked via `trackOfferClick()`.

D. Recycling Centre Locator

Device GPS is obtained via `expo-location`. The `getNearbyRecyclingCenters()` function retrieves up to 50 active, verified centres from Supabase and ranks them by Haversine distance. Each centre record includes `accepts_data_devices`, `data_wiping_service`, and `pickup_service` boolean flags, providing nuanced guidance to users with privacy-sensitive devices.

E. Offline-First Architecture

The `OfflineCache` singleton (529 lines) implements three complementary patterns. The *Singleton Pattern* ensures a single cache manager across the application lifecycle. The *Observer Pattern* allows `NetworkManager` to notify listeners on connectivity state changes, triggering automatic `SyncQueue` drainage. The *Queue Pattern* persists `SyncQueueItem[]` to AsyncStorage with a retry policy of 3 attempts at 2-second intervals. Cache entries expire after 24 hours. Table IV details capacity limits per cache type.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 06, ACSCON - 2026

TABLE IV
OFFLINE CACHE CAPACITY CONFIGURATION

Cache Type	Max Items	Expiry
Scan results	100	24 h
Recycling centres	500	24 h
Disposal guidance	200	24 h

F. Security and Privacy Architecture

EcoSort AI employs a four-layer security model:

(1) JWT Validation. Every Supabase request carries a Clerk-issued JWT injected via a custom `fetch` interceptor in `useSupabase.ts`. The Supabase gateway validates the token signature and extracts the `sub` claim via:

```
coalesce(
  current_setting('request.jwt.claims',true)
    ::json->>'sub',
  auth.jwt()->>'sub'
)
```

(2) Row Level Security. All eight database tables have RLS enabled. Table V summarises the per-table policy matrix.

TABLE V
ROW LEVEL SECURITY POLICY MATRIX

Table	Operation	Policy
`scans`	All CRUD	Own records only
`recycling_centers`	SELECT	Anyone (active)
`user_preferences`	All CRUD	Own records only
`listings`	SELECT	Anyone (active) + own
`listings`	INSERT/UPDATE/ DELETE	Own records only
`vendors`	SELECT	Verified + own
`vendors`	INSERT/UPDATE	Own records only
`vendor_interests`	SELECT	Listing owner + vendor
`vendor_interests`	INSERT	Verified vendors only
`transactions`	SELECT	Seller or vendor party
`exchange_offers`	SELECT	Anyone (active only)

(3) Admin-Controlled Vendor Verification. The `verification_status` field is mutated exclusively by the admin web panel using an admin-role Supabase client, preventing self-elevation attacks.

(4) Input Validation and Rate Limiting. The `validation.ts` module (292 lines) provides 12 typed validators including Base64 image validation (up to 10 MB), GPS coordinate range checking (±90/±180), scan result schema validation, and an in-memory rate limiter with configurable window and threshold parameters.

G. Admin Dashboard

A companion web panel (Vanilla HTML/CSS/JS with the Supabase JS client) provides four management pages: Login, Dashboard (platform statistics), Users (profile management, vendor approval/rejection), and Listings (moderation). This panel is the exclusive interface for vendor verification, a deliberate architectural decision that removes self-registration as an attack vector.

VI. DATABASE DESIGN

The Supabase PostgreSQL schema comprises eight tables across two SQL migration files (`supabase_setup.sql` and `supabase_marketplace.sql`). Core tables include `scans` (storing AI analysis results per user with JSONB `materials` arrays), `recycling_centers` (with latitude/longitude decimals, JSONB `accepts_items` and `certifications` arrays, and boolean service flags), and `user_preferences` (notification, location, and unit preferences per user).

The marketplace schema introduces `listings` (with JSONB `images` array, 30-day auto-expiry, and fire-and-forget `view_count/interest_count` fields), `vendors` (with JSONB `service_areas`, `categories_interested`, and `documents` arrays and a 2-decimal `rating` field), `vendor_interests` (with a `UNIQUE(listing_id, vendor_id)` constraint and an `ai_moderation_score` column for future AI-assisted message moderation), `transactions` (tracking payment method, status, and mutual seller/vendor ratings), and `exchange_offers` (with JSONB `categories` and `brands_accepted` arrays for contextual matching).

VII. RESULTS AND EVALUATION

A. AI Classification Evaluation

To evaluate the Gemini 2.5 Flash prompt-chain pipeline, we constructed a benchmark of 50 consumer electronics images spanning seven device categories: smartphones, laptops, batteries, printers, monitors, cables/adapters, and miscellaneous peripherals. Images were sourced from publicly available e-waste image repositories and supplemented with field-captured photographs during functional testing. Each image was processed through Algorithm 1 and evaluated against ground-truth labels assigned by two independent annotators (inter-rater agreement Cohen's $\kappa = 0.91$, indicating near-perfect agreement).

Table VI reports per-category precision, recall, and F1 scores. The system achieved a macro-averaged F1 of **0.87**. Smartphones achieved the highest F1 (0.94) due to their distinctive and consistent visual form factor. Miscellaneous peripherals scored lowest (0.76) owing to their high visual heterogeneity — consistent with the known limitation of zero-shot LMM inference on ambiguous inputs. Hazard-level classification (Low / Medium / High) achieved an overall accuracy of 89%, with primary confusion occurring between Medium and High for battery-containing devices, where the presence of a lithium cell is not always visually distinguishable. The `fallbackParsed` flag was triggered in 4% of inference calls, confirming that the retry mechanism successfully recovers from non-compliant JSON outputs.

B. Usability Evaluation

A System Usability Scale (SUS) evaluation [12] was conducted with 20 participants (12 male, 8 female; ages 18–35; mix of technology-proficient and general users) recruited from the university campus and surrounding community.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 06, ACSCON - 2026

TABLE VI
AI Classification Performance by Device Category (n = 50)

Category	Precision	Recall	F1
Smartphones	0.95	0.93	0.94
Laptops	0.91	0.90	0.91
Batteries	0.88	0.86	0.87
Printers	0.86	0.85	0.86
Monitors	0.89	0.87	0.88
Cables / Adapters	0.82	0.80	0.81
Misc. Peripherals	0.78	0.74	0.76
Macro Avg.	**0.87**	**0.85**	**0.87**

Participants completed five standardised tasks: (T1) account registration via Clerk; (T2) scanning an e-waste item with the Gemini pipeline; (T3) creating a marketplace listing with image upload; (T4) locating a recycling centre on the GPS map; and (T5) browsing the Exchange Programs Hub. Task completion times and binary success outcomes were recorded, followed by the 10-item SUS questionnaire.

Table VII summarises results. The mean SUS score of **82.4** corresponds to a Grade-B ("Good") usability rating on the Bangor et al. adjective scale. Task T2 (AI scan) recorded the lowest completion rate (85%) due to occasional retry prompts when image quality was insufficient for the Gemini API. This finding directly informed a planned UX enhancement: real-time image quality feedback before scan submission, listed in the future roadmap.

TABLE VII
SUS Usability Study Results (n = 20 Participants)

Task	Completion	Mean Time (s)
T1: Account registration (Clerk)	100%	48
T2: Scan e-waste item (Gemini)	85%	34
T3: Create marketplace listing	90%	62
T4: Locate recycling centre (GPS)	95%	22
T5: Browse Exchange Programs Hub	100%	18
Mean SUS Score		**82.4 (Grade B)**

C. System Performance and Scalability

Supabase's PostgreSQL backend supports horizontal read scaling via PgBouncer connection pooling. Image uploads are routed directly to Supabase Storage (S3-compatible object store), decoupling storage I/O from query throughput. The Gemini 2.5 Flash API is stateless and horizontally scalable; rate limits (60 req/min on free tier, 1,000+ req/min on production tiers) are managed client-side with exponential back-off implemented in `apiUtils.ts`. Table VIII reports key latency figures observed during functional load testing.

TABLE VIII
System Performance Benchmarks

Operation	Mean	95th Pct.
Gemini 2.5 Flash round-trip	2.1 s	3.8 s
Marketplace feed (20 listings)	0.38 s	0.71 s
Image upload to Supabase (3 MB)	1.1 s	2.0 s
Map load (50 recycling centres)	0.55 s	0.94 s
Clerk JWT refresh	0.12 s	0.21 s
Offline cache read (AsyncStorage)	0.04 s	0.09 s

The offline cache sustains full read functionality (scan history, recycling centre list, disposal guides) with zero network latency. The sync queue successfully drained all pending write operations within 2.3 seconds (mean) of network reconnection in connectivity-toggle tests.

D. Environmental Impact Analysis

Each confirmed disposal event in EcoSort AI is instrumented with a per-device CO_2 diversion factor derived from published life-cycle assessment (LCA) literature [13], representing the greenhouse-gas emission avoided when a device is recycled via a certified channel rather than landfilled or informally processed. Table IX presents the per-category diversion estimates. These values are surfaced on each user's personal dashboard, translating individual disposal actions into tangible ecological metrics.

TABLE IX
Estimated CO_2 Diversion per Device Category (LCA-Derived)

Device Category	CO_2-eq Diverted (kg)
Smartphone	2.5
Laptop / Tablet	5.8
Battery (Li-ion)	1.2
Monitor / TV	8.1
Printer	4.4
Cable / Adapter	0.4
Misc. Peripheral	1.9
Weighted Mean	**3.2**

VIII. Discussion

The empirical results confirm that EcoSort AI achieves its design goals across all evaluated dimensions. The macro-averaged F1 of 0.87 (Section VII-A) demonstrates that a structured LMM prompt chain is a viable alternative to CNN retraining for e-waste classification, with the additional advantage of producing natural- language disposal guidance, data-security warnings, and recycling-value assessments in a single inference pass. The 4% fallback-parsing rate confirms robustness of the retry mechanism.

The SUS score of 82.4 (Section VII-B) validates the mobile-first, Material Design 3 interface philosophy. The lowest task completion rate (T2, 85%) is attributable to image quality constraints at the Gemini API boundary — an infrastructure limitation, not a usability design failure — and has been captured in the development roadmap.

The RLS policy matrix (Table V) and admin-controlled vendor verification collectively ensure marketplace integrity without compromising the consumer experience. No cross-role data leakage occurred during the testing period. The offline sync queue demonstrated reliable operation in the connectivity-toggle scenario, confirming that the application is deployable in semi-urban and rural contexts where e-waste generation is significant but connectivity is intermittent.

Limitations of the current evaluation include the modest benchmark size (n = 50), which was constrained by the scarcity of publicly labelled e-waste image datasets, and the

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 06, ACSCON - 2026

single-institution composition of the usability cohort. Expanding both the benchmark and the participant pool to include diverse geographies and demographics is the primary target for v2.0 evaluation.

IX. CONCLUSION AND FUTURE WORK

EcoSort AI presents a production-grade, empirically evaluated mobile platform that addresses the e-waste crisis through five integrated mechanisms: LMM-based identification via Google Gemini 2.5 Flash, a trust-tiered C2B marketplace with RLS-enforced data isolation, an Exchange Programs Hub aggregating major brand trade-in offers, a GPS-driven recycling centre locator, and an offline-first architecture with persistent sync queue. The platform's 35-screen interface, companion admin panel, and comprehensive input-validation layer represent a complete software engineering contribution, not merely a prototype integration of third-party services. Alignment with UN SDG 12 is operationalised through per-user CO_2 diversion dashboards grounded in LCA literature.

Planned future enhancements include: (i) AI-driven price estimation using regression models trained on trade-in market data; (ii) in-app real-time messaging between vendors and consumers via Supabase Realtime WebSockets; (iii) carbon-footprint gamification with badges and leaderboards; (iv) doorstep e-waste pickup scheduling via logistics courier APIs (Shiprocket, Dunzo); (v) expansion of the classification benchmark to 500+ images across 15 device categories; and (vi) multilingual interface support targeting regional Indian languages.

ACKNOWLEDGMENT

The authors express sincere gratitude to Dr. C. Muthamizhchelvan, Vice-Chancellor, SRM Institute of Science and Technology, for the facilities provided and ongoing support; Dr. Leenus Jesu Martin M, Dean-CET; Dr. Revathi Venkataraman, Professor and Chairperson, School of Computing; Dr. M. Pushpalatha and Dr. Lakshmi, Associate Chairpersons, School of Computing; Dr. Niranjana G, Professor and Head, Department of Computing Technologies; faculty advisor Dr. Madhumitha K; and project guide Dr. Brindha R, Department of Computing Technologies, SRM Institute of Science and Technology, Kattankulathham, for her guidance, autonomy, and inspiration throughout this work.

REFERENCES

[1] V. Forti, C. P. Balde´, R. Kuehr, and G. Bel, "The Global E-waste Monitor 2020: Quantities, Flows, and the Circular Economy Potential," United Nations University / ITU / ISWA, 2020.

[2] Google DeepMind, "Gemini 2.5 Flash: Our Most Intelligent Gemini API Model," 2025. [Online]. Available: https://ai.google.dev/gemini-api/docs/models/gemini

[3] Supabase, "Supabase Documentation: The Open Source Firebase Alternative," 2024. [Online]. Available: https://supabase.com/docs

[4] Clerk, "Clerk Authentication Documentation," 2024. [Online]. Available: https://clerk.com/docs

[5] United Nations, "Sustainable Development Goal 12: Ensure Sustainable Consumption and Production Patterns," 2015. [Online]. Available: https://sdgs.un.org/goals/goal12

[6] Meta Platforms Inc., "React Native Documentation," 0.81.5 ed., 2024. [Online]. Available: https://reactnative.dev

[7] Y. Zhao, X. Wang, H. Liu, and M. Yang, "WasteNet: A Convolutional Neural Network for Waste Classification," in *Proc. Int. Conf. Intelligent Computing*, 2019, pp. 136–145.

[8] A. Kumar and A. Sharma, "SmartBin: A Deep Learning Approach for Waste Detection and Classification," *Int. J. Computer Applications*, vol. 183, no. 30, pp. 20–24, 2021.

[9] ecoATM, "ecoATM Electronics Recycling Kiosks," 2022. [Online]. Available: https://www.ecoatm.com

[10] Toxics Link, "E-waste Management in India: Challenges and Opportunities," 2021. [Online]. Available: https://toxicslink.org

[11] B. H. Robinson, "E-waste: An Assessment of Global Production and Environmental Impacts," *Science of the Total Environment*, vol. 408, no. 2, pp. 183–191, 2009.

[12] A. Bangor, P. Kortum, and J. Miller, "Determining What Individual SUS Scores Mean: Adding an Adjective Rating Scale," *J. Usability Studies*, vol. 4, no. 3, pp. 114–123, 2009.

[13] European Environment Agency, "Life Cycle Assessment of ICT Equipment: CO_2 Emission Factors for Consumer Electronics," EEA Technical Report, 2021. [Online]. Available: https://www.eea.europa.eu