APPENDIX N

Mobile Application Architecture

Progressive Web App — Native iOS — Native Android — Single API Layer

Confidential — For Informational Purposes Only — Patent Pending

This appendix describes the complete mobile application architecture of the DCXchange.net platform. The platform delivers full exchange functionality across three access points — a Progressive Web App (PWA) accessible from any browser on any device, a native iOS application, and a native Android application — all powered by a single shared API layer that guarantees complete data consistency and real-time synchronization regardless of how any participant chooses to access the platform.

The three-access-point architecture is not three separate platforms. It is one platform delivered three ways. Every participant sees the same data, the same listings, the same bids, and the same market activity regardless of whether they are on a desktop browser, an iPhone, or an Android tablet. This uniformity is a function of architecture, not of maintenance effort.

SECTION 1 — THREE-TIER ACCESS MODEL

DCXchange.net is accessible through three distinct but architecturally unified access points. Each access point serves a different participant preference and device context. Together they ensure that no participant is excluded from full platform participation by device type, operating system, or willingness to download an application.

Access Point	Device Coverage	Capability	Installation	Audience
PWA (Browser)	Any device with a modern browser — no download or installation	Full feature parity with desktop web platform	Immediate — visit DCXchange.net on any mobile browser	All participants
Native iOS App	iPhone and iPad — iOS 15 and above	Full feature parity plus biometric auth, push notifications, offline saved searches	App Store download — one tap install	All verified participants
Native Android App	Android phone and tablet — Android 10 and above	Full feature parity plus biometric auth, push notifications, offline saved searches	Google Play download — one tap install	All verified participants

SECTION 2 — PROGRESSIVE WEB APP (PWA)

The Progressive Web App is the DCXchange.net website itself, built to the W3C Progressive Web App specification. It requires no installation, no App Store approval, and no separate development effort beyond the web platform. Any participant who visits DCXchange.net on a mobile device is using the PWA. The distinction between the desktop website and the mobile PWA is purely the screen it renders on — the underlying application, data layer, and feature set are identical.

The term Progressive Web App describes a set of technical standards — a service worker, a web app manifest, and HTTPS — that together enable a website to behave like a native application. The

DCXchange.net platform satisfies all three requirements by design. The result is a platform that a participant on an iPhone can add to their home screen with a single tap, launch from a branded icon, and use with the same full functionality as a downloaded app — without visiting the App Store, without creating an Apple ID, and without consuming device storage beyond the app icon and cached data.

PWA Technical Specifications

Component	Specification and Implementation
Technology Standard	W3C Progressive Web App specification — supported natively by all modern browsers including Safari, Chrome, Firefox, and Edge
Responsive Framework	Single responsive codebase renders correctly at all screen sizes from 320px mobile to 4K desktop without separate mobile site or device detection
Service Worker	Background script enables offline capability for saved searches and recently viewed listings, push notification delivery without app store, and faster load times through intelligent caching
Web App Manifest	JSON configuration file defines app name, icon set, theme colors, display mode, and start URL — enables Add to Home Screen functionality on iOS and Android
HTTPS Requirement	PWA specification requires HTTPS throughout — DCXchange.net's TLS-enforced data transmission architecture satisfies this requirement by design
Installation Prompt	Browser-native install prompt appears automatically on qualifying devices after participant engagement criteria are met — no manual configuration required
Update Mechanism	Service worker update cycle ensures all participants receive platform updates automatically without manual app update through an app store
Icon and Splash Screen	Full icon set across all required sizes for iOS and Android home screens, plus branded splash screens on launch — visually indistinguishable from a native app icon

PWA Participant Experience

A participant accessing DCXchange.net from a mobile browser for the first time experiences a fully responsive interface that adapts to their screen size without horizontal scrolling, pinching, or layout degradation. Navigation is touch-optimized with appropriately sized tap targets. The market activity ticker, bid history charts, and analytics dashboard all render correctly on mobile screen dimensions. After a defined engagement threshold — typically a second or subsequent visit — the browser presents an Add to Home Screen prompt. The participant taps it once. DCXchange.net appears on their home screen as a branded icon. On subsequent launches, the platform opens directly to the exchange interface, full-screen, with no browser chrome, indistinguishable in appearance from a native application.

SECTION 3 — NATIVE IOS AND ANDROID APPLICATIONS

The native iOS and Android applications provide the full DCXchange.net platform experience with the addition of capabilities that browser-based applications cannot access: native push notifications delivered to the device lock screen, biometric authentication processed exclusively by the device's secure hardware, background data synchronization without requiring the app to be open, and native camera and file system integration for document upload workflows.

The native applications are purpose-built for their respective platforms — not wrappers around the web interface. A wrapper app loads a web page inside a native shell. A purpose-built native app

communicates directly with the platform API and renders every screen using native UI components. The difference is performance, reliability, and the depth of integration with device capabilities. DCXchange.net's native applications fall into the latter category. They are real applications, built in Swift for iOS and Kotlin for Android, that treat the DCXchange.net API as their data layer and the device as their rendering environment.

Native Application Technical Specifications

Component	Specification and Implementation
iOS Platform	Swift — native iOS application targeting iOS 15 and above, covering 98%+ of active iPhone and iPad devices in the United States
Android Platform	Kotlin — native Android application targeting Android 10 (API level 29) and above, covering 95%+ of active Android devices in the United States
API Layer	Both native applications communicate exclusively with the DCXchange.net REST API — the same API layer that serves the web application, ensuring complete data consistency and real-time synchronization across all access points
WebSocket Connection	Native socket implementation maintains persistent real-time connection to the platform event stream, powering live auction bid chart updates, market ticker events, and message delivery without polling
Biometric Authentication	Face ID and Touch ID on iOS, fingerprint and face unlock on Android — processed exclusively by the device's native secure enclave. No biometric data is transmitted to or stored on any DCXchange.net server at any time
Push Notifications	Apple Push Notification Service (APNs) for iOS, Firebase Cloud Messaging (FCM) for Android. Notification categories include new listing alerts matching saved search criteria, auction bid events on watched listings, incoming messages, and auction closing alerts
Document Handling	Native file system integration for due diligence document download, in-app PDF viewer for immediate document review, and native camera and photo library access for document upload from mobile devices
Offline Capability	Saved searches, recently viewed listings, and active message threads cached locally for access without network connectivity. Bid submission and listing creation require live connection and are clearly indicated as requiring online status
App Store Compliance	iOS application submitted and maintained in compliance with Apple App Store Review Guidelines. Android application submitted and maintained in compliance with Google Play Developer Policy Center requirements
Background Processing	Lightweight background refresh delivers updated saved search results and new message counts to notification badges without requiring the app to be open

SECTION 4 — SINGLE API LAYER — THE ARCHITECTURE THAT UNIFIES ALL THREE

The architectural decision that makes three access points function as one platform is the single shared API layer. Every request made by the web browser, the iOS app, and the Android app goes to the same API endpoints. Every response comes from the same data layer. Every event — a new bid, a new listing, an incoming message, an auction closing — is broadcast to all connected clients simultaneously through the same event stream.

This architecture is not merely a technical convenience. It is the platform commitment described in the main pro forma: one platform, one data layer, one consistent verified experience regardless of how any participant chooses to access it. The following table describes the specific ways the single API layer enforces this commitment.

Principle	How the Single API Layer Enforces It
Data Consistency	A listing created on the desktop web platform is immediately visible on mobile. A bid submitted from a native iOS app is immediately reflected in the web browser bid history chart. There is one data layer. There are no synchronization delays between access points.
Feature Parity	Every platform function accessible on the desktop web platform is accessible on mobile. There are no features reserved for desktop only and no features that exist only on mobile. The API layer enforces this uniformity by design.
Authentication	Session tokens issued at login are valid across all access points simultaneously. A participant logged in on desktop and mobile at the same time maintains separate authenticated sessions on each device through the same API authentication layer.
Tier Enforcement	Subscription tier permissions are evaluated at the API layer on every request, regardless of which access point initiates the request. A Free tier seller attempting to access Professional tier features through a modified mobile client receives the same rejection as through the web interface.
Real-Time Events	The WebSocket event stream that powers the market activity ticker, live bid charts, and push notifications is a single server-side stream consumed by web clients via browser WebSocket and by native clients via native socket implementations. All clients receive the same events simultaneously.

SECTION 5 — FEATURE PARITY MATRIX

The following table documents feature availability across all three access points. Features marked as Native in the mobile columns indicate capabilities that are technically superior on native apps due to device integration, but functionally equivalent from a platform participation standpoint.

Platform Feature	PWA (Browser)	Native iOS	Native Android
Full listing browse and search	✔	✔	✔
Complete listing detail view	✔	✔	✔
Due diligence document access	✔	✔	✔
Internal messaging system	✔	✔	✔
Seller listing creation and management	✔	✔	✔
Auction bid submission	✔	✔	✔
Live auction bid history chart	✔	✔	✔
Market activity ticker	✔	✔	✔
Analytics dashboard	✔	✔	✔
Saved searches and alerts	✔	✔	✔

Account verification and profile	✔	✔	✔
Push notifications — new listings	Browser only	✔ Native	✔ Native
Push notifications — bid events	Browser only	✔ Native	✔ Native
Push notifications — messages	Browser only	✔ Native	✔ Native
Biometric authentication (Face/Touch ID)	No	✔ Native	✔ Native
Offline saved search access	No	✔	✔
Home screen installation	✔ Add to Home Screen	✔ App icon	✔ App icon
App store presence	No	✔ Apple App Store	✔ Google Play Store
Camera access for document upload	✔ Browser camera	✔ Native camera	✔ Native camera
Background data sync	No	✔	✔

SECTION 6 — SECURITY AND PRIVACY ARCHITECTURE ACROSS ALL ACCESS POINTS

The security architecture of the DCXchange.net mobile platform reflects the same principles that govern the desktop web platform, applied to the specific capabilities and constraints of mobile devices.

Data Transmission

All data transmitted between any access point and the DCXchange.net API layer is encrypted in transit using Transport Layer Security (TLS). This applies equally to the PWA, the native iOS application, and the native Android application. No unencrypted data transmission occurs on any access point under any circumstance.

Authentication and Session Management

Participants authenticate with email and password credentials transmitted over TLS to the API authentication endpoint. Passwords are processed through cryptographic one-way hashing algorithms and are never stored in any readable form on any server or device. Session tokens issued at login are cryptographically signed, time-limited, and stored in secure storage on the device — the iOS Keychain for iOS, the Android Keystore for Android, and HTTPS-only secure cookies for the PWA. Biometric authentication on native apps delegates all biometric processing to the device's native secure enclave. No biometric data is ever transmitted to or stored on any DCXchange.net server.

Local Data Storage

The native applications cache a limited set of data locally for offline access and performance optimization — specifically saved search criteria, recently viewed listing summaries, and active message thread previews. All locally cached data is encrypted at rest using the device's native encryption framework — iOS Data Protection and Android Encrypted SharedPreferences. Sensitive data fields including due diligence documents, full listing financial details, and participant identity information are never cached locally and require a live authenticated API connection to access.

Device Permission Model

The native applications request only the device permissions required for the features that use them. Camera and photo library access is requested only when a participant initiates a document upload

workflow. Push notification permission is requested during onboarding with a clear explanation of notification types. Location permission is not requested or used. No contact list access, no microphone access, and no background location tracking is requested or implemented.

SECTION 7 — MOBILE ARCHITECTURE VALUE IN THE LICENSING AND ACQUISITION CONTEXT

The three-access-point mobile architecture represents material commercial value in the context of all three DCXchange.net commercial arrangements. The following table addresses mobile architecture specifically as it relates to each arrangement.

Commercial Arrangement Consideration	Mobile Architecture Position
No Development Required	The PWA is the DCXchange.net website itself — no separate development effort, no additional infrastructure, no app store account. Any licensee who operates the web platform automatically operates a fully functional PWA.
Native Apps as Platform Asset	The native iOS and Android applications are included in all three commercial arrangements — License to Operate, Platform Business Sale, and Technology/IP Acquisition. A licensee receives working, store-ready native applications as part of the platform, not as a separate development engagement.
App Store Accounts	Under a License to Operate arrangement, the native apps are distributed under TooziT's App Store and Google Play accounts with the licensee's branding applied. Under a Platform Business Sale or Technology Acquisition, all App Store and Google Play developer accounts and application assets transfer to the acquiring party.
White-Label Capability	The app name, icon, splash screen, color scheme, and branding elements in both the PWA manifest and native app configurations are parameterized and can be updated for white-label deployment under a licensee's brand identity without code changes to the underlying application logic.
Update Responsibility	Under a License to Operate, TooziT maintains responsibility for all iOS and Android OS compatibility updates, App Store policy compliance updates, and security patches. Under a Platform Business Sale or Technology Acquisition, this responsibility transfers to the new owner along with the development infrastructure required to manage it.

A platform that delivers full exchange functionality on desktop, browser-based mobile, iOS, and Android is a platform that reaches every participant regardless of their preferred device. For a licensing candidate or acquirer evaluating market reach, the three-access-point architecture eliminates a category of objection — participants cannot be excluded by device type — and adds a category of credibility. An exchange that exists only on desktop is a desktop tool. An exchange that exists on every device a participant carries is infrastructure.

One Platform. One Data Layer. Every Device.