



Your Lean Do IT offer

This transactional email contains the estimate generated after completing the Lean Do IT questionnaire. Offer reference: LDI-20260703-CD5047ED.

Offer code: LDI-20260703-CD5047ED

Offer valid until July 10, 2026, 10:33 UTC. After this date, the preliminary analysis fee (30% of the offered price) can no longer be paid.

Offered price

2 253 USD

If you wish to order the software, complete the form below and send it in reply to this offer (the offer must remain visible in your reply).

"I accept the scope of work and the amount shown as the Offered price. I have paid a non-refundable fee of 676 USD (30% of the amount shown as the Offered price) for the preliminary project analysis. Please proceed with the order. Name of the ordering party: Billing address: ..."

Preliminary analysis fee

To pay the preliminary project analysis fee for offer LDI-20260703-CD5047ED, use the payment page below.

Fee amount: 676 USD

[Payment page](#)

https://app.leandoit.com/pay/a3motUAQkQ_3u6Vko6Shf7iIMGWhO3usarGk28UwwWY

Payment is handled by Stripe.

Questionnaire and user description

Solution type

Mobile application

Target platforms

iOS (iPhone, iPad), Android

Mobile store publishing

Store publishing by the client (App Store / Google Play)

Mobile application technology

Flutter mobile app (cross-platform iOS/Android)

Features and requirements

User login and accounts, Online payments, Notifications (email, push, SMS), Integrations with other systems, Data management (create, edit, delete), AI features, Offline mode (works without internet)

Language scope

Full multilingual support (6+ languages, content translation)

Number of users

Up to 1,000 users

How often do users use the application?

Daily

Security level

Enhanced security (GDPR, encryption, audit)

GDPR requirements (personal data protection)

Standard GDPR compliance (policies, consents, processing register)

Cryptocurrencies and NFT

—

Pre-production testing

Client-side acceptance testing before production

Vendor share of sales revenue

No – full upfront development fee

Application name

fluent

Project and functionality description

Language learning app.

Features:

- creating a user account
- logging in via Google and Apple
- deleting a user account
- generating flashcards
- translations
- simultaneous translation
- using the phone's local AI
- translated texts can be saved for offline use
- there should be an option to shuffle flashcards
- Ability to add flashcards to favorites
- Support for the following languages: Polish, English, German, French, Spanish
- The app should include incentives to learn—gamification

The app should have a nice, modern design. The app should support subscription-based payments. Monthly and annual subscriptions.

There should be limits on flashcard generation. For the free account: 2 flashcards per day and 30 per month; for the Pro version: 100 per day and 1,000 per month.

Notifications should be available for inactivity and when the daily point limit is reached.

The app should award points for studying.

Users should be able to listen to phrases, record their own speech, and compare it to the original.

Detailed software specification

Software Architecture and Technical Specification: fluent

1. Executive Summary

This document defines the complete technical architecture and implementation specification for "fluent", a cross-platform language learning mobile application. The system shall be developed using Flutter to provide a unified codebase for iOS and Android platforms. The application shall feature on-device AI processing, offline data persistence, subscription management, and standard GDPR compliance. The architecture is designed to support up to 1,000 daily active users without requiring a dedicated backend system, utilizing on-device processing and direct third-party SDK integrations.

2. Business Context

The "fluent" application provides a language learning platform focused on flashcard generation, translation, and pronunciation comparison. The system incentivizes daily usage through a gamification engine that awards points for studying. The application operates primarily offline once content is generated, utilizing the mobile device's local AI capabilities for translations and flashcard creation.

3. Functional Scope

The system shall implement the following functional requirements:

- * **User Account Management:** The application shall allow users to create accounts, authenticate via Google and Apple, and delete their accounts.
- * **Flashcard Generation:** The application shall generate flashcards based on user input, subject to daily and monthly limits based on the user's subscription tier.
- * **Translation & AI:** The application shall provide simultaneous translation and flashcard generation using the phone's local AI. Translated texts shall be saved locally for offline use.
- * **Flashcard Operations:** The application shall provide an option to shuffle flashcards and allow users to add flashcards to a favorites list.
- * **Language Support:** The application shall support content and translation for Polish, English, German, French, and Spanish.
- * **Gamification:** The application shall award points for studying and enforce a daily point limit.
- * **Notifications:** The application shall trigger notifications for user inactivity and when the daily point limit is reached. Notification channels

shall include email, push, and SMS.

- * **Audio & Speech:** The application shall allow users to listen to phrases, record their own speech, and compare the recording to the original phrase.

- * **Payments:** The application shall support subscription-based payments with monthly and annual billing cycles.

- * **Flashcard Generation Limits:**

- * Free Account: 2 flashcards per day, 30 per month.

- * Pro Version: 100 flashcards per day, 1,000 per month.

4. User Roles

The system shall implement Role-Based Access Control (RBAC) with the following roles:

- * **Free User:** Has access to standard application features. Flashcard generation is limited to 2 per day and 30 per month.

- * **Pro User:** Has access to premium application features. Flashcard generation is limited to 100 per day and 1,000 per month.

5. System Modules

1. **Authentication Module:** Manages Google and Apple OAuth flows, session persistence, and account deletion.

2. **Flashcard & Data Management Module:** Manages CRUD operations for flashcards, favorites management, and shuffle logic.

3. **Local AI & Translation Module:** Interfaces with the mobile device's local AI models for simultaneous translation and flashcard generation.

4. **Audio Comparison Module:** Handles audio playback of phrases, microphone recording, and comparative analysis of user speech to original audio.

5. **Gamification Module:** Tracks study points, enforces daily point limits, and monitors user activity for inactivity tracking.

6. **Notification Module:** Integrates with mobile OS push notification systems and third-party providers for email/SMS to deliver inactivity and limit alerts.

7. **Subscription Module:** Interfaces with App Store and Google Play billing systems to validate monthly and annual subscriptions.

8. **Localization & i18n Module:** Manages translation files, date, currency, and format localization for supported languages.

9. **Security & Compliance Module:** Manages local data encryption, GDPR consents, and the processing activity register.

6. Architecture

The system shall be architected as a standalone mobile application without a dedicated backend system.

- * **Client Application:** A Flutter application compiled for iOS (iPhone, iPad) and Android.
- * **Local Storage:** The application shall utilize an encrypted local database (e.g., SQLite with SQLCipher) for offline persistence of flashcards, user preferences, and audit logs.
- * **Secure Storage:** The application shall utilize OS-level secure storage (iOS Keychain, Android Keystore) for authentication tokens and encryption keys.
- * **On-Device AI:** The application shall execute translation and flashcard generation models entirely on the mobile device's hardware.
- * **Third-Party Integrations:** The application shall communicate directly with external services via their respective mobile SDKs.

7. Data Flow

1. **Authentication Flow:** The user initiates login via Google or Apple SDK -> OAuth token is received by the application -> Token is stored in OS-level secure storage -> User session is established.
2. **Flashcard Generation Flow:** User requests flashcard generation -> Application checks local storage for daily/monthly limit compliance -> If within limits, input is passed to the local AI model -> AI returns translation/flashcard data -> Data is encrypted and saved to the local database -> Gamification module awards points.
3. **Payment Flow:** User selects monthly/annual subscription -> Application initiates purchase via App Store/Google Play billing SDK -> Store validates transaction -> Application unlocks Pro User role locally.
4. **Notification Flow:** Gamification module detects inactivity or daily point limit reached -> Application triggers local push notification or calls third-party Email/SMS SDK -> User receives notification.

8. Data Model

The application shall maintain the following local data structures:

- * **User:** `id`, `auth\_provider`, `provider\_user\_id`, `role` (Free/Pro), `total\_points`, `daily\_points`, `subscription\_expiry\_date`.
- * **Flashcard:** `id`, `user\_id`, `source\_text`, `translated\_text`, `source\_language`, `target\_language`, `is\_favorite`, `created\_at`.
- * **UsageCounter:** `id`, `user\_id`, `date`,

```
`daily_flashcards_generated`, `monthly_flashcards_generated`.
* **AuditLog:** `id`, `action_type`, `timestamp`, `data_hash`.
* **ConsentRecord:** `id`, `consent_type`, `granted`, `timestamp`.
```

9. APIs

As no backend system is included in the scope, the application shall not expose any external APIs. The application shall consume the following external APIs via integrated SDKs:

- * **Google Identity Services API:** For Google authentication.
- * **Apple Sign In API:** For Apple authentication.
- * **StoreKit API:** For iOS in-app purchases.
- * **Google Play Billing API:** For Android in-app purchases.
- * **Push/Email/SMS Provider APIs:** Integrated directly via SDKs for notification delivery.

10. Integrations

The application shall integrate with the following external systems:

- * **Google SDK:** For user authentication and identity verification.
- * **Apple SDK:** For user authentication and identity verification.
- * **Apple App Store / Google Play Store:** For subscription billing and validation.
- * **Notification Services:** Third-party SDKs for Email and SMS notifications (Push notifications handled via native OS APIs).

11. Security

- * **Data Encryption:** The application shall encrypt all locally stored user data and translated texts using AES-256 encryption.
- * **GDPR Compliance:** The application shall implement standard GDPR compliance mechanisms including the display of privacy policies, capture of explicit user consents, and maintenance of a local processing activity register.
- * **RBAC:** The application shall enforce role-based access control to differentiate feature availability between Free and Pro users.
- * **Audit:** The application shall log security-relevant actions (login, account deletion, consent changes) to a local audit log.
- * **Account Deletion:** The application shall provide a function to permanently delete all local user data, consents, and audit logs upon account termination.

12. Non-Functional Requirements

- * **Performance:** The local AI translation and flashcard generation shall execute within 5 seconds on supported devices.
- * **Scalability:** The architecture shall support up to 1,000 concurrent standalone users without degradation, as all processing is client-side.
- * **Availability:** The application shall remain fully functional in offline mode for reviewing saved flashcards, listening to phrases, and recording speech.
- * **Usability:** The application shall feature a modern UI design as specified in the project requirements.
- * **Localization:** The application shall utilize an i18n framework to support full multilingual UI and content formatting.

13. Infrastructure & Deployment

- * **Environments:** A staging environment shall be configured for client-side acceptance testing prior to production go-live.
- * **Deployment:** The Client shall be responsible for publishing the application to the App Store and Google Play.
- * **Infrastructure Costs:** Infrastructure costs for third-party services (if required by the Client) are borne by the Client and excluded from the development estimate.
- * **Dependencies:** The Client shall provide all required libraries, drivers, and access keys for third-party services.

14. Compliance

The application shall adhere to Standard GDPR compliance requirements:

- * Display of privacy policies within the application.
- * Mechanisms for capturing and storing user consents.
- * Maintenance of a processing activity register accessible within the application.
- * Right to erasure implemented via the account deletion function.

15. Required Launch Artifacts

The Client shall supply the following artifacts prior to launch:

- * App store listing descriptions and assets for App Store and Google Play.
- * Terms of service and privacy policy documents.
- * API access keys for selected third-party services (e.g., notification providers, Google/Apple OAuth)

providers, Google/Apple OAuth).

- * Any required proprietary libraries or drivers for local AI models.

16. Not Feasible — Out of Scope

- * **Backend System / API:** A dedicated backend server or API is out of scope. All processing is client-side.
- * **Website or Web Dashboard:** Any web-based administration or user portal is out of scope.
- * **Desktop App:** Desktop platforms are out of scope.
- * **Additional Languages:** Support for languages beyond the five specified (Polish, English, German, French, Spanish) is excluded due to contradictory requirements in the source materials (questionnaire mentions 6+, description specifies 5).
- * **Infrastructure Costs:** All infrastructure and third-party service usage costs are excluded from the offer.
- * **Deployment Services:** Publishing to app stores and deployment services are excluded from the development scope.
- * **Peripheral Integration Testing:** Integration testing with peripheral devices is the responsibility of the Client.

Important notes

- The estimate does not include infrastructure costs, which must be provided by the Client.
- The estimate assumes that all required artifacts will be supplied by the Client. Examples include app store listing descriptions, terms of service, privacy policies, access keys, etc.
- Costs related to the use of third-party services such as maps, AI models, email, hosting, etc. are borne by the Client.
- The estimate covers only software development within the scope described by the user. Deployment services are additional services.
- The Client is obliged to provide all required libraries, drivers, etc. Integration testing with peripheral devices is ALWAYS on the Client's side.
- This offer is valid until July 10, 2026, 10:33 UTC. After this date, the preliminary analysis fee (30% of the offered price) can no longer be paid.

This transactional email was sent by Lean Do IT.

© 2026 Lean Do IT