**App report 1 & 2**

**Construcción de aplicaciones móviles**

**Integrantes:**

**Juan José Hurtado Cantin 202224472**

**Juan Manuel Gordillo Sastoque 202222259**

**Juan José Ramírez 202013228**

**Universidad de los Andes**

**Bogotá, 30 de septiembre del 2025**

# App Report Part 1

Repository Link: <https://github.com/signalapp/Signal-Android>

## 1. App Description

Signal is a messaging application focused on privacy that offers services such as text messages, voice calls, video calls, sharing photos, videos, and files, all protected with end-to-end encryption that prevents third parties, including the app itself, from accessing the content. It also allows the creation of large groups, sharing stories that disappear after 24 hours, contact verification through QR codes, automatic face blurring in images, and, in some countries, cryptocurrency payments with MobileCoin.

Signal does not show ads or sell user data, since it operates as a nonprofit organization sustained by donations from users, foundations, and philanthropic contributions, which ensures its independence and commitment to security. Since its launch, supported in part by a loan from WhatsApp co-founder Brian Acton, it has managed to stay afloat with millions of dollars in annual contributions without compromising user privacy.

By 2024, the app had reached more than 220 million downloads and around 70 million monthly active users (TechCrunch, 2022; Signal Transparency Report, 2023). These figures tend to rise especially during times of distrust in other platforms like WhatsApp. What makes Signal most interesting is that it proves that an app can be widely used and effective without relying on advertising or data tracking, since its open-source code assures transparency and regular audits. This makes it one of the most reliable tools for activists, journalists, and people concerned with protecting their communications.

## 2. Repository Description

Signal’s repository on GitHub is the official open-source client of Signal for Android devices, maintained under the AGPL-3.0 license. It is a very active repository with more than 17,000 commits, around 27.5k stars, and 6.5k forks (GitHub, 2025).

The main languages used are Kotlin (~59%) and Java (~40%), with small portions of Handlebars, JavaScript, HTML, and Python. This mix reflects its evolution, with newer features in Kotlin while the older base remains in Java.

The project is structured into multiple components that keep functionalities separate and maintainable. For example, the app module contains the Android application logic, libsignal-service manages communication with the backend, and other modules focus on contacts, paging, image editing, video, and QR scanning.

There are also specialized libraries such as billing/donations, device-transfer, and debuglogs-viewer. A key highlight is reproducible builds, which allow anyone to verify the integrity of the distributed APKs. Over 140 releases are tagged, with frequent updates.

Branch analysis: Most development occurs in the main branch, complemented by feature/release branches. Community forks (6k+) suggest high engagement. Lines of code are approximately: Kotlin ~1.1M, Java ~800k, Other ~50k.

## 3. Business Questions

Here is the corrected set of Business Questions according to types:

• Type 1 (Telemetry): What percentage of calls (voice/video) fail due to network quality or encryption handshake errors?

• Type 2 (Direct UX Improvement): How often do users activate disappearing messages in chats, and what is the average timer duration?

• Type 3 (Feature Analysis): What fraction of monthly active users post at least one story per week?

• Type 4 (Benefits from Data): Which donation methods (credit card, Google Pay, MobileCoin) generate the highest contributions across different regions?

# App Report Part 2

## Architecture and Design Patterns

Singleton Pattern: Signal uses a singleton like design in parts of its encryption / session management infrastructure. For example, the cryptographic key manager or session manager typically must ensure only one instance is used per user/device to avoid key mismatches or duplication.

Builder Pattern: Signal could use the Builder Pattern when creating complex messages. A message may include text, images, videos, voice notes, and encryption data, but not all fields are always needed. A Message Builder would let developers add only the parts they need in a clear and chainable way. This would make the code easier to read, avoid having many different constructors, and allow future extensions if new message types are added.

Strategy Pattern: The Strategy Pattern would be useful in Signal for managing different ways of handling connections or encryption. For example, the app could switch between retry policies for weak networks or choose different encryption/compression methods depending on the device. With strategies separated into different classes, developers could change or add new behaviors without touching the main code. This makes the app more flexible, easier to test, and adaptable to different conditions.

## 2. UI/UX Analysis

Signals use a design focused on privacy and clarity. It uses a white/blue color palette, Roboto font, and familiar chat metaphors.

View analysis:

Chat View: It’s a simple layout that contains the chat list, with circles as profile pictures for each chat, also with the name of the chat in the upper right part of the profile picture, below the name of the chat you can see the last message sent or received and also de message status at the right part of the screen in each chat.

Call View: It’s an interface with two options above where you got all the calls or just the missed ones, you can also make new calls by looking for a new contact by their number or name.

Story View: The story view is very similar to the WhatsApp status view, where you can see what your contacts post by seeing their profile picture in a circle frame.

I liked that the interface in general is simple and intuitive, the colors are not too heavy to the user’s sight, and it accomplishes its purpose in a very good way. I think it could improve in the call view by adding the option to have your whole contacts in a list, without having the need to have called them recently.

## 3. Quality Attributes

## Security: Signal’s strongest attribute is security. All communications are protected with end-to-end encryption using the Signal Protocol, which ensures that only the sender and the receiver can access the content. The app also uses reproducible builds so that anyone can verify the integrity of the code published in the Play Store, and it undergoes constant security audits by the community. These practices guarantee transparency and reduce the possibility of malicious alterations.

## Reliability: The application is constantly updated and maintained, which improves its stability and ensures compatibility with new Android versions. Signal also includes robust error handling in functions like calls and file transfers, and it collects anonymous telemetry (only with user consent) to detect failures and connection problems. This contributes to a reliable service even in situations with low connectivity.

## Usability: Signal has an intuitive interface that makes it easy to use even for new users. Privacy indicators, such as the clock icon for disappearing messages or the padlock for encrypted calls, are clear and easy to understand. The learning curve is very short compared to other apps, and users can quickly adopt its features without confusion.

## 4. Libraries and Dependencies

libsignal-protocol: This is the core cryptographic library that implements the Signal Protocol. It is responsible for managing encryption keys, secure sessions, and message authentication, guaranteeing that all communications remain private and protected.

OkHttp & Retrofit: These libraries are used for network communication. OkHttp manages HTTP connections in a reliable and efficient way, while Retrofit simplifies the creation of APIs and data transfers between the client and the server. Together, they ensure stable and secure exchanges of information.

Glide: An image loading and caching library that allows Signal to handle profile pictures, attachments, and multimedia content in a fast and optimized way. Glide reduces memory consumption and improves performance, which is key in messaging apps where images are constantly loaded.

WorkManager: A library that allows background task scheduling, such as synchronizing messages, updating conversations, or sending logs for debugging. Its integration ensures that tasks are executed efficiently, even if the app is closed or the device is restarted.

Play Billing: This dependency is used to manage donations directly from the application. It integrates with Google Play’s billing system to allow users to contribute financially in a safe and transparent way, reinforcing Signal’s nonprofit sustainability model.