

Testing checklist: messaging app

Source: <https://howtotestaproduct.com/how-to-test/messaging-app/>

Generated: 2026-10-07. Licence: CC BY 4.0.

☐ 1. release smoke tests

Done when: On the release build, two test users on different platforms exchange a text, a photo and a voice note, each item shows as delivered, and report and block appear in the chat.

Risks: store-rejection. Tool class: mobile automation. Usually run by: QA engineer.

☐ 2. end-to-end functional tests

Done when: A child account cannot be added to a group, call or chat by anyone outside its approved contacts, including through a group invite from an approved friend.

Risks: user-harm. Tool class: mobile automation. Usually run by: QA engineer.

☐ 3. compliance checks

Done when: A reviewer confirms the privacy notice states what happens to uploaded address books and what is shared with group companies, and report and block appear in every one-to-one and group chat.

Risks: regulatory-penalty, store-rejection. Tool class: test management. Usually run by: product manager.

☐ 4. security scanning

Done when: The scan report shows no open finding on message, attachment and contact-sync endpoints, and the scan of each app build flags no chat data written to shared storage or left in unprotected backups.

Risks: data-breach. Tool class: DAST scanner. Usually run by: security specialist.

☐ 5. penetration testing

Done when: A tester with one account cannot fetch another user's attachment by guessing its link, cannot read another user's chat history from the server, and finds no chat content in the phone's shared storage or unprotected backup.

Risks: data-breach. Tool class: proxy. Usually run by: security specialist.

☐ 6. resilience and failure testing

Done when: With the network gateway or cache tier throttled, clients back off, connected users keep sending messages, and the service recovers without manual restarts.

Risks: dependency-failure, outage-under-load. Tool class: chaos tool. Usually run by: developer.

☐ 7. integration tests

Done when: With the cloud network gateway in front of the chat backend switched off in staging and then restored, phones reconnect without the app freezing, every message sent during the outage arrives once and each recipient gets one notification.

Risks: dependency-failure. Tool class: API client. Usually run by: developer.

☐ 8. performance and load testing

Done when: A reconnect of all simulated clients at once, with empty caches, completes without errors and without new logins being refused.

Risks: outage-under-load. Tool class: load generator. Usually run by: developer.