

# Testing checklist: telemedicine app

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

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

## ☐ 1. release smoke tests

Done when: On the release build, a test patient books a visit, enters the waiting room and sees and hears the test clinician once the clinician admits them.

Risks: store-rejection, outage-under-load. Tool class: mobile automation. Usually run by: QA engineer.

## ☐ 2. end-to-end functional tests

Done when: A test patient completes intake, a video visit and a prescription to the chosen pharmacy, and a stimulant refill request is held until a clinician documents a new assessment.

Risks: user-harm, regulatory-penalty. Tool class: browser automation. Usually run by: QA engineer.

## ☐ 3. payment and money-flow testing

Done when: After a test patient cancels the membership in the app, the payment sandbox shows no further charge on the next billing date.

Risks: money-loss. Tool class: API client. Usually run by: QA engineer.

## ☐ 4. compliance checks

Done when: A reviewer confirms that every controlled-substance prescription record shows the prescriber's registration and the patient's state, and that the HIPAA risk analysis covers the video and chat services.

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

## ☐ 5. analytics and event tracking tests

Done when: A network capture of sign-up, symptom intake, booking and visit screens shows no symptom, drug name or visit reason sent to any advertising or analytics domain.

Risks: data-breach, regulatory-penalty. Tool class: proxy. Usually run by: QA engineer.

## ☐ 6. accessibility testing

Done when: A screen reader user books a visit, joins the video room and reads the clinician's chat messages without sighted help.

Risks: accessibility-liability. Tool class: accessibility checker. Usually run by: accessibility specialist.

## ☐ 7. security scanning

Done when: The scan report for the app, API and video signalling service lists no open high-severity finding on the release date.

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

## ☐ 8. penetration testing

Done when: A tester logged in as patient A cannot open patient B's visit notes, prescriptions or chat, and a deactivated clinician account cannot open any chart.

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

## ☐ 9. performance and load testing

Done when: With as many test patients as join in the busiest appointment slot, every one reaches the waiting room and the clinician admits each without an error.

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