One booking. Every next step.

I built a customer booking flow and business workspace that connect the appointment to its schedule changes, deposits and messages.

A booking is the beginning.

A customer chooses a time. The business still has to honor it, collect the right deposit, handle changes and communicate clearly.

The problem

Scheduling becomes harder when the original appointment, a later change, its payment and its messages are treated as unrelated records.

My role

I designed and developed Chime across the customer interface, Studio, application logic and documentation.

The solution

Connect the appointment lifecycle while showing what happened, what remains pending and what someone can do next.

This case combines reviewed source, a fictional customer flow and an authenticated workspace viewed without changes. The functioning public website was not matched to the inspected source revision.

Visit Chime

Follow the illustrated guide

Watch the booking walkthrough

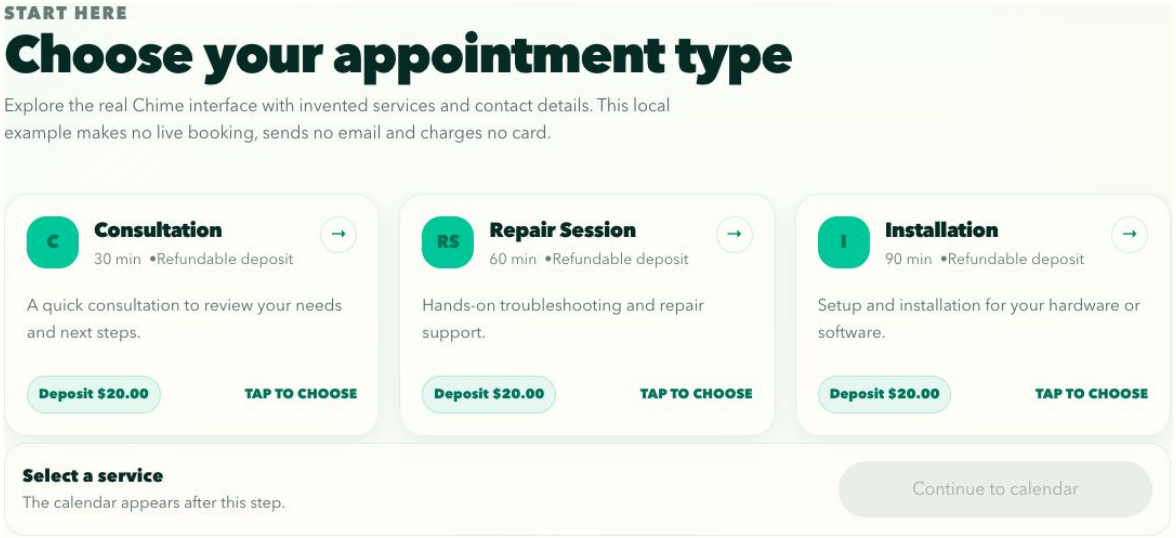


FIG 01 Begin with the service the customer needs.

Make the next choice obvious.

A short sequence keeps the appointment visible as the customer makes each decision.

Keep the context close

Service supplies duration and deposit information. Month orientation leads to a focused day view. Terms sit beside the appointment summary, followed by contact and any required deposit.

Make progress recognizable

A reading gate unlocks agreement; valid required fields unlock continuation. Saving, retry and result have their own states. Jordan's fictional 30 minute Consultation keeps the same date, time and USD 20 sample deposit throughout.

Preserve a useful retry

In the reviewed implementation, a payment identifier survives a failed booking save. Retrying that save does not send the customer back to start another payment.

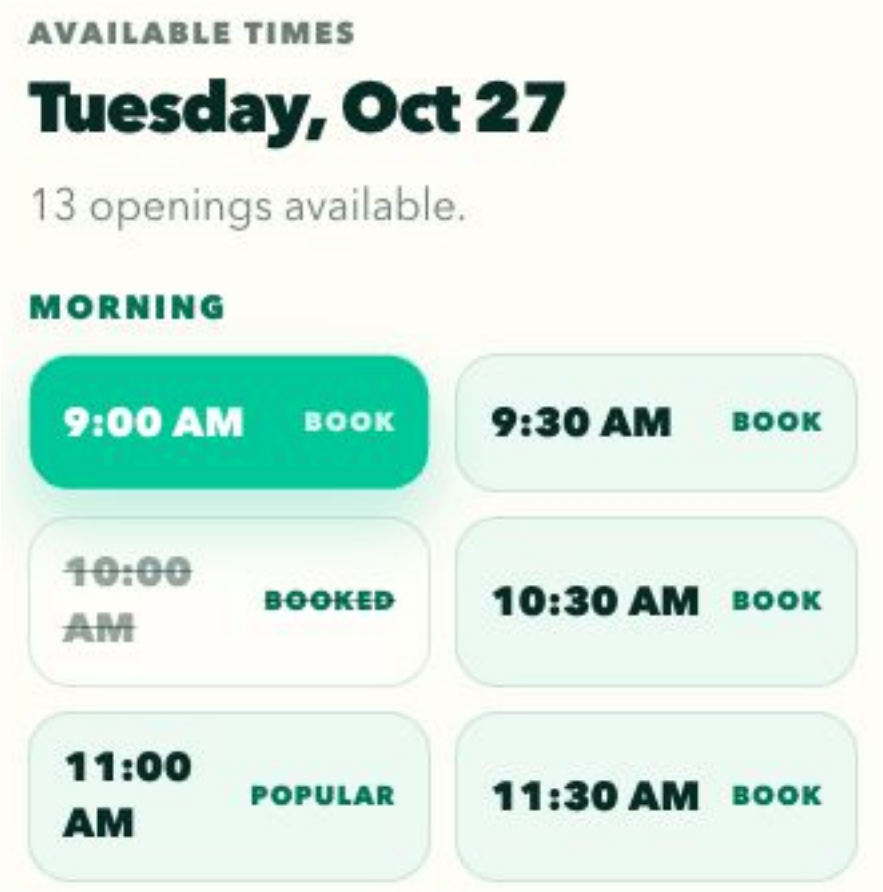


FIG 02 A focused day view separates available, booked and popular times.

Check the choice before accepting it.

An older calendar screen should not decide whether the business can still accept an appointment.

Check before saving

The reviewed version checks whether the business, service and opening can still accept the request. It uses the latest capacity, notice, advance window and deposit rules rather than trusting an older calendar screen.

Keep the records together

The accepted appointment connects the customer, calendar and payment references. Staff access is checked against the current business membership and role.

REVIEWED IMPLEMENTATION

One acceptance transaction

Business + service

Current rules + capacity

Linked booking, customer, calendar and payment references

The engineering behind that decision

The reviewed TypeScript and Express service uses PostgreSQL transactions and a slot row lock to recheck capacity and write linked records together. Signed sessions are checked against current membership, role and business status. Shared appointment contracts connect the frontend and backend. These are inspected controls, not a security certification or proof of every simultaneous scheduling case.

A proposal is not an applied change.

Staff need to see both the appointment still in force and the alternative being discussed.

Review the change once

In the reviewed version, dragging and typed edits reach the same comparison of time, duration, person and location. The review names the recipient and the action allowed by the rules. An unchanged record is not sent again.

Keep pending work visible

A customer decision keeps the original active and allows withdrawal. Approval rechecks the proposal, so the alternative time is not guaranteed reserved.

Bound the source claim

Random decision tokens are stored as hashes, expire and handle replay. That revision's staff related modes fall back to customer approval because its staff channel is unfinished. This is not asserted as a deployed limit; live templates contain newer team messages and the revisions were not matched.

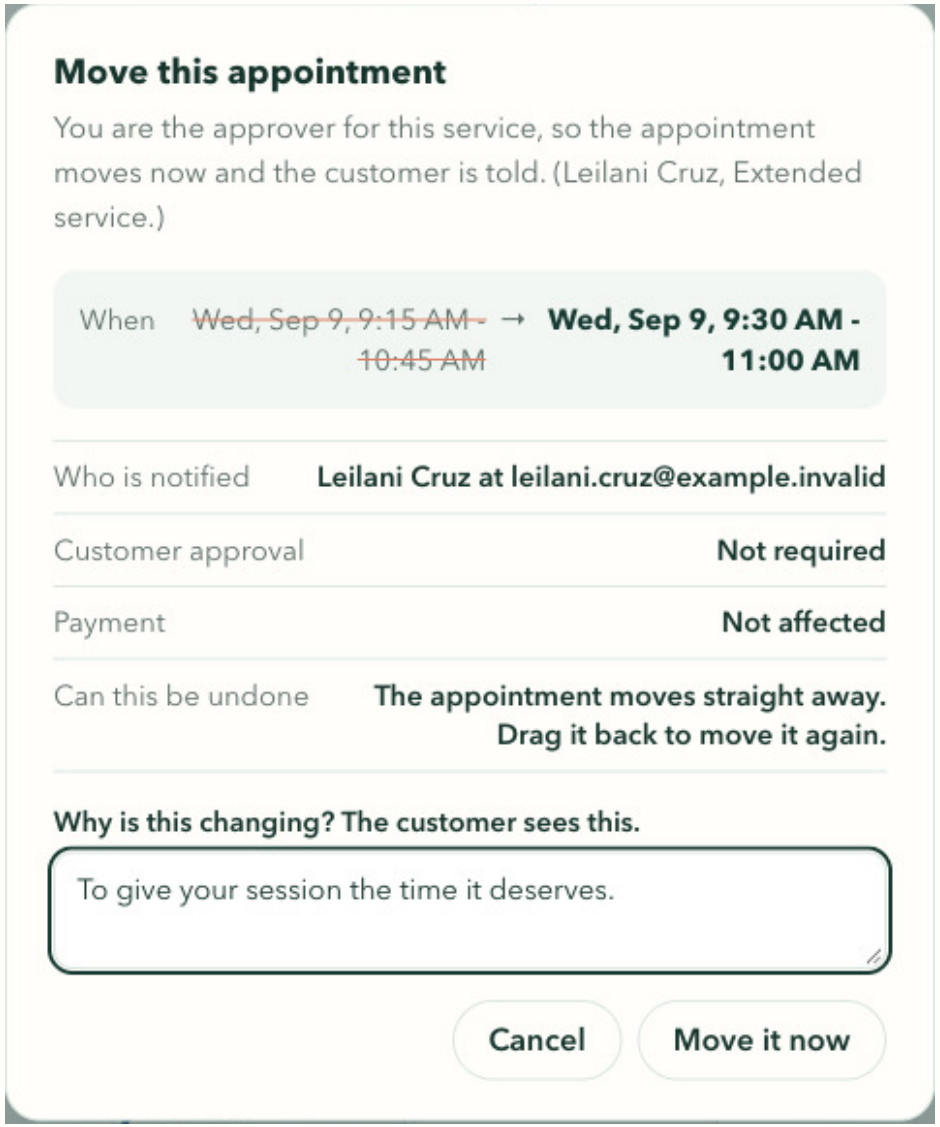


FIG 03 Earlier populated example: compare what stays and what is proposed.

Name the money state.

A deposit can be held, collected, released or returned. Those are different events.

Keep the origin with the payment

A deposit can begin as a hold rather than money already collected. The reviewed version keeps the originating account with a new connected payment, so later actions use the right payment context.

Make actions reviewable

Staff can distinguish collection, release and refund. The system checks permission, the current record and how much remains refundable, then records who acted and the payment service's reported result.

The payment mechanics

The inspected Stripe route uses manual capture, version preconditions and idempotency keys. Refunds are bounded by captured money less prior returns. Historical records without a connected account use the platform branch.

What the illustration proves

The live Deposits workspace was empty and no money action was performed. The fictional customer used a local simulated form. Neither establishes real collection, settlement or universal provider recovery.

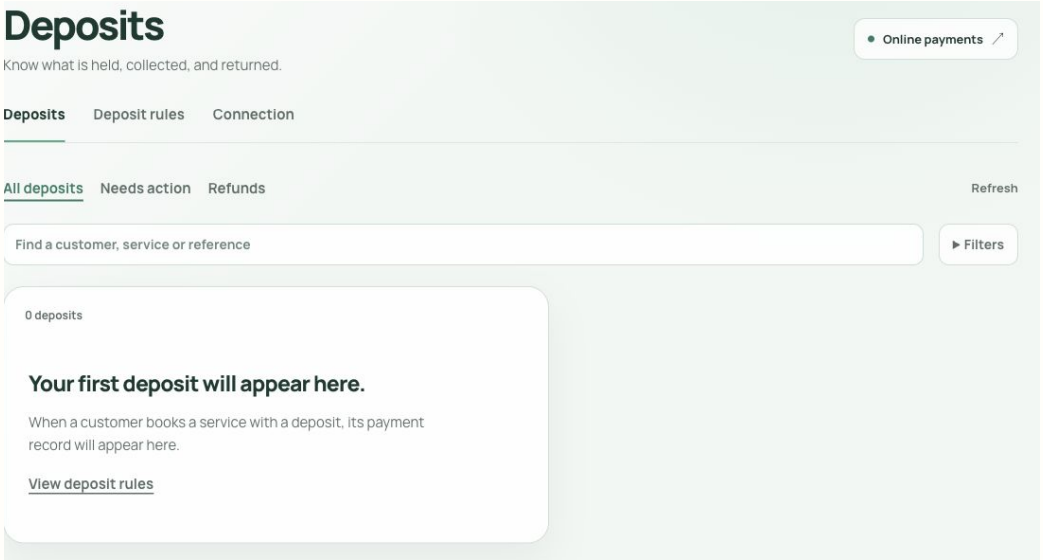


FIG 04 Current Deposits separates the record list from rules and connection setup.

Keep delivery separate from booking.

A prepared confirmation, provider acceptance and inbox arrival are different states.

Give messages their own work area

The live workspace separates Activity, Templates and Sending setup. A published sample can be viewed without sending. Send preview email is an actual send.

Show the missing evidence

The sender reported Running, but Last message accepted was None recorded. The screen states that acceptance does not confirm inbox delivery.

Background mechanics

The reviewed worker uses leases, locked queue selection, attempts, retry backoff and business turns. Provider idempotency supports repeated handoff attempts. This source inspection did not run a live provider or establish complete failure recovery.

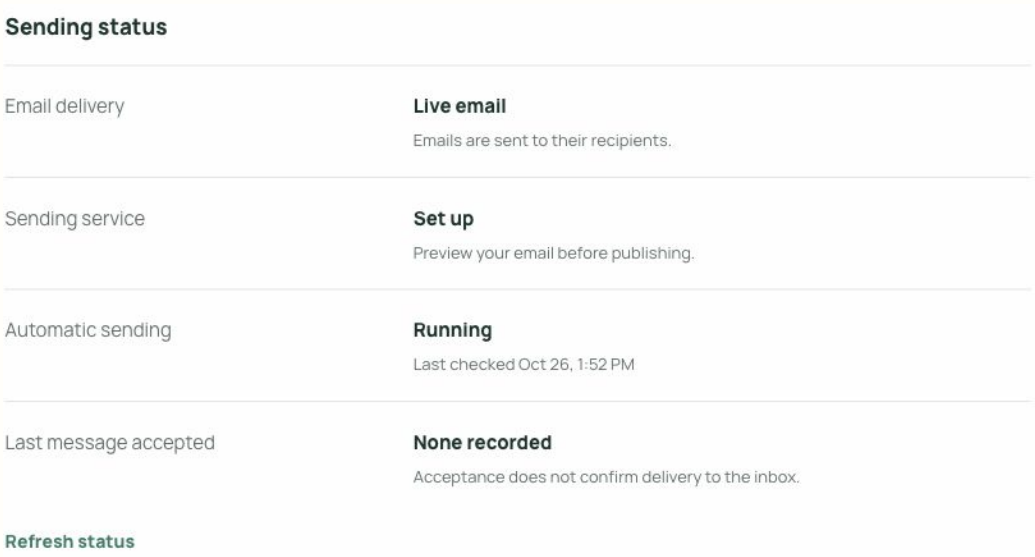


FIG 05 A running service can still have no recorded accepted message.

Separate design from being ready.

A working preview does not establish a published customer channel.

Let the business judge appearance in context

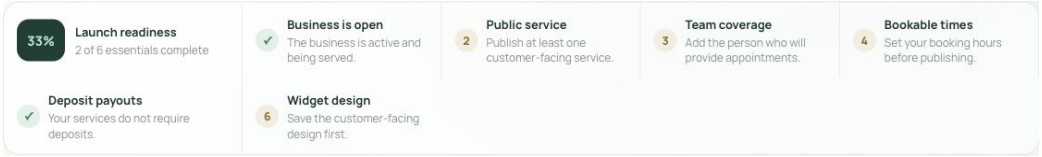
Widget designer uses Your brand, Customer welcome, Color and finish and Saved design, with Wide, Inline and Phone previews.

Make readiness specific

Launch separates hosted booking from a website widget. In the observed workspace, the hosted link control was checked while the website widget was Not published yet and blocked by incomplete readiness. The broad Live badge did not settle every channel's state.

A shared customer experience

The reviewed project separates customer, Studio, embed and approval builds while sharing relevant models and components. Public connection code carries routing and display configuration, never administrator credentials.



READINESS DETAIL

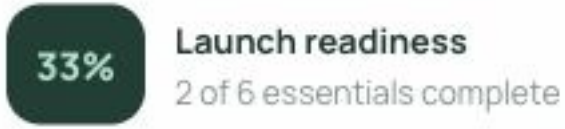


FIG 06 Two of six essentials are complete; four setup items remain.

Teach the job, then offer depth.

The guide uses one fictional booking, clear staff routes and recognizable results.

Help someone act

Six customer tasks give the next action and what should appear. Staff can enter at the task they need. Exact labels, nearby pictures and a compact recovery table reduce searching.

Keep the explanation maintainable

Extra explanation stays available without interrupting the task. One content source supplies web and PDF, so correcting a control name or condition does not create competing procedures.

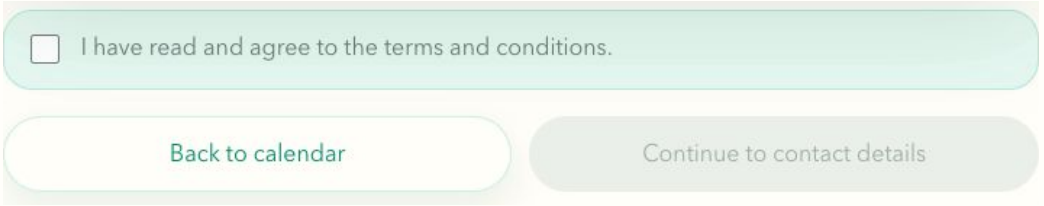


FIG 07 A focused state detail explains when the next action becomes available.

State what was actually checked.

Reviewed October 26, 2026.

Available evidence	Boundary
Actual synthetic customer flow	No saved booking, charge or email
Authenticated live Studio screens	Viewed without saved changes or financial and message actions
Inspected private source and regression scripts	Not deployment matched; scripts not executed for this review

What the checks do not establish

The capacity script exercises the last place sequentially, not a simultaneous race. Stripe routing uses a recording stand in, not settlement. No adoption, time saving, security certification, accessibility conformance or representative reader study is claimed.

Read the complete illustrated guide

Explore Chime