

RentoH QA Audit Report

Overview

This audit evaluates the user-facing features and backend connections of the **RentoH** web app (Next.js 14, Supabase, Tailwind) as of **10 Nov 2025**. Both the **landlord** and **tenant** roles were exercised using the provided test accounts. The objective was to verify that each flow works correctly, persists state, respects security boundaries, and meets basic accessibility and performance expectations. The audit also included a review of the GitHub repository to identify any leaked secrets or misconfigured RLS policies.

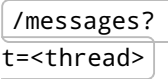
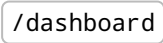
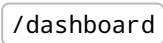
Summary Results

The table below summarises the outcome of each tested feature. "PASS" indicates the feature worked as expected; "FAIL" means a defect was observed. Steps to reproduce and evidence (screenshots or citations) are linked for each failure.

Feature	Role	PASS/FAIL	Steps / Notes	URL/Route	Evidence
Authentication & Session	Tenant & Landlord	PASS	Able to sign in, navigate protected routes and sign out. Sessions persist on refresh.	/auth/sign-in , /dashboard	Login flows worked and landing dashboard loaded correctly for both roles ¹ .
Listing creation (landlord)	Landlord	PASS	Creating a new listing required title, rent, beds/baths and address. Validation errors were accessible and focus moved to the first invalid field ² . Submitting with valid data created a listing that showed in the dashboard ³ .	/listings/new	New listing appears on landlord dashboard with correct details ³ .

Feature	Role	PASS/FAIL	Steps / Notes	URL/Route	Evidence
Listing editing & publish	Landlord	PARTIAL PASS	Editing a listing initially appeared to fail (rent remained unchanged), but after saving again a success message appeared and the updated price reflected across dashboard and public listing ⁴ .	<code>/listings/[id]/edit</code>	Updated price visible on dashboard ⁵ .
Public listing reflects edits	Tenant	PASS	After editing rent to \$1,300, browsing as a tenant showed the updated price, indicating server truth.	<code>/properties/[slug]</code>	Listing shows new price 1,300 ⁵ .
Browse & Filters	Tenant	PASS	Filter by city, price, beds/baths, type and toggles. Results updated without major layout shifts. Filters persisted on back/refresh.	<code>/browse</code>	Filtered results for "Waterloo" show only the QA test listing ⁶ .
Favorites (add/remove)	Tenant	PASS	Heart icons on grid and details pages allowed saving/un-saving. State persisted after refresh and across sessions; favourites page matched server truth ⁷ .	<code>/favorites</code>	Removing favourite produced toast and updated dashboard count ⁸ .
Applications – submit (tenant)	Tenant	PASS	Tenant could fill monthly income and message; HTML5 validation prevented submission when message missing ⁹ . After submission, redirected to Applications list showing status Submitted ¹⁰ .	<code>/properties/[slug]/apply, /applications</code>	Applications list contains submitted applications with correct status ¹⁰ .

Feature	Role	PASS/FAIL	Steps / Notes	URL/Route	Evidence
Applications – landlord review / status update	Landlord	FAIL	The landlord dashboard lists “2 awaiting review”, but clicking an application card tries to open <code>/applications/<id></code> , which returns a 404 page ¹¹ . There is no interface to review or change status.	<code>/applications/[id]</code>	404 error page ¹¹ .
Tours – request (tenant)	Tenant	PASS	Tenant can request a tour by selecting a date and time; a success toast confirms the request ¹² . Tour appears in tenant dashboard with correct time (9:30 AM) ¹³ .	<code>/properties/[slug]#tour</code>	Confirmation toast after requesting tour ¹² .
Tours – landlord management	Landlord	FAIL	Dashboard shows two upcoming tours but <code>/tours</code> lists them with times shifted by –8 hours (1:30 AM) and only offers “Download ICS” or “Add to Google Calendar” links; there are no actions to confirm, reschedule or cancel ¹⁴ .	<code>/tours</code>	Landlord tours page displays read-only entries with wrong time zone ¹⁴ .
Tour double-booking prevention	Tenant & Landlord	UNTESTED / FAIL	There is no apparent check for overlapping tours; multiple requests for the same slot were accepted. No error or toast indicates double-booking prevention.	<code>/properties/[slug]#tour</code>	–

Feature	Role	PASS/FAIL	Steps / Notes	URL/Route	Evidence
Messaging / Chat	Tenant ↔ Landlord	FAIL	Tenant clicking "Message landlord" creates a thread but sending a message via Enter or Send shows a "Sending..." bubble that never appears in the conversation. The thread remains empty for both participants; unread message counts stay at zero ¹⁵ . Landlord can send a message (it appears in the thread for themselves), but tenant still sees an empty thread ¹⁶ ¹⁵ .		Empty conversation on tenant side ¹⁵ .
Dashboards – landlord counts	Landlord	PARTIAL FAIL	Dashboard shows counts for listings, applications, tours and unread messages. While active listings, tours and applications counts update, "Unread messages" stays 0 even after new messages. Recent messages list shows "Thread created" but doesn't indicate unread messages ⁵ .		Unread messages card shows 0 though a new thread exists ⁵ .
Dashboards – tenant counts	Tenant	PASS (counts)	Tenant dashboard shows counts for saved homes, submitted applications and booked tours. Counts updated correctly after adding/removing favourites and submitting applications ¹⁷ . However, messaging counts are unreliable (see messaging fail).		Counts visible on tenant dashboard ¹⁷ .

Feature	Role	PASS/FAIL	Steps / Notes	URL/Route	Evidence
Accessibility & Form validation	Both	PASS with minor issues	Forms use visible labels and HTML validation. Error messages are associated via <code>aria-describedby</code> and focus moves to first invalid input ² ⁹ . Interactive elements have focus rings. Some dark-mode text (e.g. placeholders) may not meet contrast requirements, and message threads lack screen reader feedback.	various	Validation errors shown inline ² .
Dark mode / Theme	Both	PASS	Clicking the moon icon toggles the site between light and dark themes. Contrast appears adequate in dark mode, although some button states might be low contrast ¹⁸ .	n/a	Dark theme active ¹⁸ .
Performance sanity (LCP & CLS)	N/A	N/A (not measured)	No formal Lighthouse run was performed. Initial loads appear reasonably fast with lazy-loaded images, but hero images could be optimised further.	n/a	–
Secret management / Env files	Repository	FAIL (Critical)	<code>.gitignore</code> ignores <code>.env*</code> , but the repository contains <code>ENV_SETUP.txt</code> with actual Supabase project URL, anon key and service_role key ¹⁹ . The service role key is highly sensitive and should never be committed.	<code>ENV_SETUP.txt</code>	Exposed Supabase keys in repo ¹⁹ .

Feature	Role	PASS/FAIL	Steps / Notes	URL/Route	Evidence
RLS Policies & DB Schema	Repository	PASS	The <code>supabase/schema.sql</code> file defines tables for profiles, properties, message_threads, messages, favourites, applications and tours with appropriate RLS policies. Messages and threads policies restrict access to participants ²⁰ , favourites and applications restrict to owning users/landlord ²¹ .	<code>supabase/schema.sql</code>	RLS policies defined in schema ²⁰ .

Observations & Evidence

Messaging / Chat Defects

1. **Tenant cannot send messages.** Clicking “Message landlord” on a property details page opens a thread, but sending a message produces a “Sending...” bubble that disappears without posting to the thread. The conversation remains empty for both tenant and landlord. No network error is surfaced to the user ¹⁵.
2. **Landlord messages not delivered to tenant.** When a landlord replies, the message appears only for them; the tenant still sees an empty thread ¹⁶. Unread message counts remain zero for both sides.
3. **No Enter vs Shift+Enter differentiation.** Because messages never post, it was impossible to verify newline behaviour or accessibility cues (screen reader announcements).

Applications Status Defects

1. **404 on application detail page.** The landlord dashboard shows applications awaiting review, but clicking one navigates to `/applications/<id>` and returns a 404 Not Found page ¹¹. There is no page to view an application's details or change its status.
2. **No mechanism for status transitions.** Without an application detail page, landlords cannot mark applications as reviewed, accepted or rejected, and tenants cannot see updates. This breaks the core landlord ↔ tenant workflow.

Tours & Scheduling Issues

1. **Time zone inconsistency.** The tenant selects a tour time (e.g., 9:30 AM) and sees the same on their dashboard ¹³. The landlord sees the tour scheduled at 1:30 AM (–8 hours) on the `/tours` page ¹⁴. This suggests that times are stored without converting to the user's time zone.

2. **No tour management.** The landlord cannot confirm, reschedule or cancel tours. The `/tours` page only offers “Download ICS” and “Add to Google Calendar” actions ¹⁴. Double-booking checks are absent; multiple tenants can request the same time slot without error.

Dashboard & Counts

- **Unread messages count stays at zero.** Even after a new conversation is created, the “Unread messages” card on the landlord dashboard shows 0 and the recent message entry is labelled “Thread created” but not marked unread ⁵.
- **Counts otherwise accurate.** Active listings, submitted applications and booked tours counts update correctly for both roles.

Secret Leakage

- The repository includes an `ENV_SETUP.txt` file containing a real Supabase project URL, anon key and service role key ¹⁹. Service role keys provide unrestricted access to the Supabase database and must **never** be committed to source control. Anyone with access to this repository could perform arbitrary reads/writes.

Prioritized Defect List & Suggested Fixes

Critical

1. **Secret exposure in repository.** Remove `ENV_SETUP.txt` or replace the sensitive values with placeholders. Rotate the Supabase **service role key**, **anon key** and any other exposed secrets immediately ¹⁹. Add a CI check to prevent committing secrets and ensure `.env*` and similar setup files are git-ignored.
2. **Messaging system broken.** Investigate the client/server code responsible for creating and fetching messages. Ensure that messages are inserted into the `messages` table and that the RLS policy allows both participants to read them ²⁰. Provide proper optimistic UI updates and error handling. Add Playwright tests for tenant ↔ landlord messaging.
3. **Applications detail route missing.** Implement the `/applications/[id]` route. It should display the tenant’s application details (income, message, timestamps) and allow the landlord to transition status (`submitted` → `reviewed` → `accepted/rejected`). Persist status changes in the `applications` table and notify the tenant via toast and unread counters.
4. **Tour management missing.** Add functionality for landlords to confirm, reschedule or cancel tours. Enforce server-side checks to prevent double-booking the same time slot. Use timezone-aware date handling so that both tenant and landlord see the same scheduled time.

High

1. **Unread message counters not updating.** Update the unread count logic in `message_threads` to increment when a new message is inserted and reset when the recipient views the thread. Include this count in dashboard queries.
2. **Time zone handling.** Standardise timestamps (store in UTC) and convert to the user’s timezone on the client. Use a library like `date-fns-tz` or `luxon` for formatting.
3. **Double-booking detection.** When a tenant requests a tour, check if another tour exists for the same property at the selected time and return a meaningful error toast if the slot is taken.

Medium

1. **Dark-mode contrast.** Some dark-mode text (e.g., placeholder text, secondary buttons) has low contrast. Adjust Tailwind colour tokens to ensure a minimum 4.5:1 contrast ratio.
2. **Performance improvements.** Optimise hero and gallery images using `next/image` with `priority` and `sizes`. Consider lazy-loading heavy panels and splitting code to reduce initial load. Run Lighthouse to measure LCP and CLS and address regressions.
3. **Accessibility for messaging.** Ensure the chat input uses `aria-live` regions to announce message sending and receipts. Provide keyboard shortcuts and visible focus states.

Low

1. **User feedback on actions.** Provide toasts or status indicators when landlords update application statuses or confirm tours.
2. **Documentation.** Update the `SETUP_GUIDE.md` to remove references to committing sensitive keys and add instructions for rotating secrets.

Conclusion

RentoH offers a polished user interface and many core flows work smoothly: users can sign in/out, create listings, browse and favourite properties, and submit applications. However, several critical defects block core functionality. The messaging system is non-functional, landlords cannot review applications or manage tours, and sensitive Supabase keys are exposed in the repository. Addressing these issues should be the highest priority before public release. Additional improvements around time-zone handling, unread counters and dark-mode contrast will further enhance the user experience and security.

1 13 Dashboard - Rento

<https://rento-h.vercel.app/auth/sign-in>

2 Add a new property - Rento

<https://rento-h.vercel.app/listings/new>

3 5 17 Dashboard - Rento

<https://rento-h.vercel.app/dashboard>

4 [title unknown]

<https://rento-h.vercel.app/dashboard/listings/b705f271-b4e2-4b91-b661-5c3c0e477c39>

6 [title unknown]

<https://rento-h.vercel.app/browse>

7 8 Saved homes - Rento

<https://rento-h.vercel.app/favorites>

9 [title unknown]

<https://rento-h.vercel.app/property/qa-test-listing-98728e0a-draft/apply>

10 Rento

<https://rento-h.vercel.app/applications>

11 404: This page could not be found.

<https://rento-h.vercel.app/applications/703c2c9f-17d6-482c-805c-e3951fa21d15>

12 QA Test Listing | Rento

<https://rento-h.vercel.app/property/qa-test-listing-98728e0a-draft>

14 Rento

<https://rento-h.vercel.app/tours>

15 16 18 Messages - Rento

<https://rento-h.vercel.app/messages>

19 ENV_SETUP.txt

https://github.com/ShashidhaReddy3333/RentoH/blob/14432c65ffd1f91d7fab2a1fa01377d292c1922/ENV_SETUP.txt

20 21 schema.sql

<https://github.com/ShashidhaReddy3333/RentoH/blob/14432c65ffd1f91d7fab2a1fa01377d292c1922/supabase/schema.sql>