ChatGPT

UGC-BCP CRM Audit (Next.js 16 & Supabase)

1. Executive Summary

- **Overall readiness** – The CRM module in `assujiar/ugc-bcp-2` is incomplete and suffers from multiple critical issues. Many UI pages are placeholders, queries omit important states, and several workflows end in dead-ends. The lead qualification bug is not a corner case – it is a systemic visibility gap where records drop out of every list. Overall readiness is **red**.
- **Major blockers (P0/P1)** –
- **Qualified Leads vanish** – After triage to `Qualified` the Lead Inbox removes the record because its query only shows `New` and `In Review` statuses [1]. There is no automatic handover or route to Sales Pool; thus the lead disappears from all pages [2].
- **No API/UI for Lead creation outside POST /api/crm/leads** – The UI has no "Add Lead" page; leads are only created via inbound marketing forms (not shown) or sales quick add.
- **Numerous missing routes and placeholder pages** – Accounts creation link goes to pipeline; Add Contact button has no handler; the "Imports" and "Customers" pages are placeholders.
- **RLS mismatch & over-permissive policies** – Several API handlers validate roles in UI but the database RLS allows broader access (e.g., `crm_leads_update_allowed` permits any creator to update their lead; the API restricts triage to marketing roles only [3]).
- **Positive aspects** – The system uses Supabase RPCs for atomic operations with idempotency and consistent logging. Role definitions are explicit (15 roles). The migrations create dedupe functions, idempotency tables, and helper enums; this foundation is strong [4] [5].

2. UI Inventory (minimum 12 findings)

Page / file	Purpose	Findings / Evidence	Problems & observations
`app/(protected)/crm/lead-inbox/page.tsx`	Lead triage queue. Fetches leads using `/api/crm/leads?view=inbox` and renders status badges. Includes triage modal and handover button [6].	Only leads with `New` & `In Review` statuses appear [1]. The `Send to Sales Pool` button calls `/api/crm/leads/{id}/handover` but the triage action changes status to `Qualified`, thus removing the record from the list. No UI shows `Qualified` leads.	After triage to `Qualified`, the lead disappears; user must remember to click `Send to Sales Pool` quickly. No page lists qualified leads.

Page / file	Purpose	Findings / Evidence	Problems & observations
`app/(protected)/crm/sales-inbox/page.tsx`	Sales “My Work Queue.” Fetches overdue activities and handover pool leads via `/api/crm/activities?view=inbox` and `/api/crm/leads?view=handover_pool` respectively. Provides claim button that calls `/api/crm/leads/{id}/claim` [7].	The list shows only leads in handover pool; triage status `Qualified` is not a criterion. Leads that were handed over appear here until claimed. Once claimed, leads vanish from sales inbox but no “My Leads” page exists.	Missing view for assigned leads and active opportunities; unclaimed leads show but claimed leads disappear from this page.
`app/(protected)/crm/pipeline/page.tsx`	Sales pipeline board. Fetches opportunities with `view=pipeline` [8]; provides quick add and stage change modals.	Only opportunities in active stages appear. Quick Add uses `/api/crm/opportunities` with `quick_add` true to call `rpc_sales_quick_add_prospect` [9]; stage change uses `/api/crm/opportunities/{id}/stage`.	Filtering by owner or account is absent. There’s no list of overdue opportunities. The board does not automatically include newly claimed leads.
`app/(protected)/crm/activities/page.tsx`	Activities planner. Fetches activities via `/api/crm/activities?view=planner` and groups them by date; supports adding and completing activities [10].	Completion calls `/api/crm/activities/{id}/complete`.	Good grouping, but there is no calendar view or link back to related lead/opportunity.
`app/(protected)/crm/accounts/page.tsx`	Accounts list. Fetches accounts via `/api/crm/accounts?view=enriched` with search; displays badges for tenure & activity status. Add Account button incorrectly links to `/crm/pipeline` [8].	Add account missing; search works but there’s no filter by owner.	Major navigation bug: Add Account link goes to pipeline. No ability to create accounts.

Page / file	Purpose	Findings / Evidence	Problems & observations
`app/(protected)/crm/accounts/[id]/page.tsx`	Account 360. Displays account details, opportunities, activities, contacts, invoice summary. Data loaded by `/api/crm/accounts/{id}`.	`Add Contact` button is a placeholder – no handler. `Add Opportunity` button simply links to pipeline with `account_id` query but does not pre-fill or restrict account.	Many parts empty (invoice snapshot, tickets snapshot). Contact creation missing.
`app/(protected)/crm/targets/page.tsx`	Prospecting targets workspace. Lists targets with filters (status, search) via `/api/crm/targets` [11]; quick add target via POST `/api/crm/targets`; convert via `/api/crm/targets/{id}/convert` [12].	UI includes counts by status, but import button is stubbed. Convert requires manual entry of next step; no pre-generated cadence.	After conversion, the UI does not navigate to the resulting lead/opportunity. Import stub is missing.
`app/(protected)/crm/imports/page.tsx`	Placeholder for import management.	Only static text; no functionality.	Page not implemented.
`app/(protected)/crm/customers/page.tsx`	Placeholder for Account 360 list.	Only static text; no list.	Not implemented.
`app/(protected)/crm/prospects/page.tsx`	Placeholder for prospects/opportunities list (duplicate of pipeline?).	Only static text describing the pipeline rules.	Redundant: duplicates pipeline page; not implemented.
`app/(protected)/crm/page.tsx`	Redirects `/crm` to `/crm/pipeline` [13].	Hard-coded redirect.	Could hide other pages; might mislead testers expecting `/crm` to have landing.

Page / file	Purpose	Findings / Evidence	Problems & observations
`components/ui`	Contains `Card`, `Button`, `Input`, `Badge`.	Styles unify design but there is no global error or toast component; forms lack validation messages.	UI components cannot handle asynchronous errors elegantly.

3. API Inventory (minimum 12 findings)

API route	Method(s)	Purpose & Evidence	Issues / Observations
`/api/crm/leads`	GET, POST	GET returns leads filtered by `view=inbox` or `handover_pool`; query restricts to `New` & `In Review` statuses [1]. POST inserts lead with dedupe check and `triage_status=New` [1].	API lacks endpoint for `view=qualified` or `view=all`. Cannot fetch leads by owner or search across statuses; leads created via POST are not reachable in UI until they are triaged.
`/api/crm/leads/[id]`	GET, PATCH	GET returns a lead. PATCH updates fields; no role restriction beyond authentication [14].	Without server-side role checks, any authenticated user may update leads (RLS might block but UI expects marketing).
`/api/crm/leads/[id]/triage`	PATCH	Allows marketing roles to set `triage_status` to `In Review`, `Qualified`, `Nurture`, `Disqualified` [15]. When `Qualified`, it sets `qualified_at` and `handover_eligible` but leaves status unchanged.	No server-side validation for mandatory fields before moving to `Qualified`. The status remains `Qualified` until manually handed over via separate call, causing disappearance.

API route	Method(s)	Purpose & Evidence	Issues / Observations
`/api/crm/leads/[id]/handover`	POST	Calls RPC `rpc_lead_handover_to_sales_pool` to move lead to sales pool [16]. Requires marketing role.	Does not automatically create opportunity or convert lead; lead remains separate.
`/api/crm/leads/[id]/claim`	POST	Calls RPC `rpc_sales_claim_lead` for sales roles; ensures idempotency [7].	After claim, lead disappears from sales inbox but not visible in pipeline or My Work queue until opportunity created via other actions.
`/api/crm/activities`	GET, POST	GET lists activities; supports `view=inbox` (due today) and `view=planner` for scheduler [17]. POST creates activity.	No pagination beyond `pageSize`. UI might become slow with many activities.
`/api/crm/activities/[id]/complete`	POST	Calls RPC `rpc_activity_complete_and_next`; can create next activity [18].	Good atomic function, but missing error handling for invalid next step.
`/api/crm/opportunities`	GET, POST	GET lists opportunities; `view=pipeline` returns active pipeline; `view=my_overdue` for overdue tasks [8]. POST either standard create or quick add via `rpc_sales_quick_add_prospect` [9].	Only the pipeline view is used in UI. Overdue view not exposed. No filter by owner or stage.
`/api/crm/opportunities/[id]/stage`	POST	Calls RPC `rpc_opportunity_change_stage` with new stage and next step; enforces stage validation [19].	API doesn't block stage change if prerequisites (e.g., quote) not met; RPC does but UI doesn't show error gracefully.
`/api/crm/targets`	GET, POST	GET returns prospecting targets with filters for status, owner, search [11]. POST upserts targets; can import multiple targets or create a single target with dedupe key [20].	There is no delete endpoint; statuses like `Dropped` still show up until manually updated.

API route	Method(s)	Purpose & Evidence	Issues / Observations
`/api/crm/targets/[id]/convert`	POST	Calls RPC `rpc_target_convert_to_lead` converting target to lead/account/opportunity [12]. Requires next step and due date.	No read endpoint for single target. After conversion, UI does not redirect to new records.
`/api/crm/accounts`	GET	Lists accounts with `view=enriched`.	No create endpoint; account creation is part of quick add prospect or quick add target conversion via RPC.
`/api/crm/accounts/[id]`	GET, PATCH	Returns account details; PATCH updates account; not implemented in UI.	Access to update accounts via API but not in UI.

4. Database / RPC / RLS Inventory (minimum 12 findings)

Object	Evidence	Findings & issues
Enums	`LeadTriageStatus` = `New`, `In Review`, `Qualified`, `Nurture`, `Disqualified`, `Handed Over` [4]; `LeadStatus` separate; `OpportunityStage` enumerates pipeline stages; `ActivityStatus`, `ActivityTypeV2`, `TargetStatus` etc [5].	Separate `Lead.triage_status` and `Lead.status` cause confusion. UI uses triage status for lead inbox but not `Lead.status`. Status is updated to `converted` when opportunity created, but triage remains unchanged.
Tables	`crm_leads` table includes fields `triage_status`, `status`, `handover_eligible`, `sales_owner_user_id`, `linked_account_id`, `linked_contact_id`, etc (from migrations).	`triage_status` is used by API queries; `status` indicates conversion. The `handover_pool` view uses `handover_eligible` flag. Missing computed columns to sync statuses.
Views	`v_pipeline_summary`, `lead_handover_pool` (exposed by migrations).	`lead_handover_pool` selects leads with `handover_eligible=true` and `sales_owner_user_id IS NULL`; these are shown in sales inbox.

Object	Evidence	Findings & issues
Functions (RPC)	`rpc_sales_quick_add_prospect` creates account, contact, lead (status `qualified`, triage not set), opportunity, cadence activities; logs idempotency [21].	Ensures atomic creation and dedupe. But triage status remains `qualified`, causing the lead to bypass inbox; it is also set to `linked_opportunity_id` and `status=converted`.
	`rpc_target_convert_to_lead` creates account/contact/lead/opportunity from prospecting target; updates target status to `converted` [21].	Triages lead as `qualified` but does not set `handover_eligible`; thus the lead never appears in handover pool.
	`rpc_lead_handover_to_sales_pool` marks lead as `handed_over_at` and sets `handover_eligible`; enforces idempotency.	Works as intended but not automatically triggered when triaged to `Qualified`.
	`rpc_sales_claim_lead` sets `sales_owner_user_id` on lead and resets `assignment_mode` [21].	Does not create opportunity; just assigns the lead. After claim, there is no UI to act on the lead.
	`rpc_opportunity_change_stage` updates `stage`, `next_step`, `next_step_due_at`, ensures mandatory `next_step` for open stages; checks quote requirement [19].	Good guardrails.
	`rpc_activity_complete_and_next` marks activity done and optionally inserts next activity [21].	Good idempotent function. No UI to create multi-step cadence outside quick add.
	`rpc_account_merge` merges duplicate accounts and moves child entities [21].	Not exposed in UI.
RLS Policies	`crm_leads_select_allowed` allows marketing and mgmt roles to see leads; sales can see leads they own or unassigned [22].	UI restricts triage to marketing; but RLS allows sales owners to update leads. Divergence between API checks and RLS must be reconciled.
	`crm_accounts_select_allowed` allows many roles; insert/update allowed to marketing, sales, support [22].	UI prohibits account creation; misalignment.
	Policies for `crm_prospecting_targets` restrict select/update to owner or sales mgmt/support.	Aligns with UI but there is no deletion mechanism.
	`crm_activities_select_allowed` allows mgmt, marketing, sales to read; update allowed to owner or managers.	Good alignment with UI.

Object	Evidence	Findings & issues
	`crm_idempotency` RLS ensures only owner reads/writes idempotency rows [22].	Good.

5. Role-based Access Matrix (minimum 10 findings)

Roles defined: Director, super admin, Marketing Manager, Marcomm, DGO, MACX, VSDO, sales manager, salesperson, sales support, EXIM Ops, domestics Ops, Import DTD Ops, traffic & warehous, finance [4].

Role	UI access (permissions.ts)	RLS policy privileges	Mismatches & notes
Director / super admin	Access all menus: dashboard, performance, CRM, ticketing, DSO.	RLS allows select/insert/update across CRM tables (via `super admin` role; Director inherits typical mgmt privileges).	Good alignment; super admin can call RPC functions.
Marketing Manager / Marcomm / DGO / MACX / VSDO	Menus: dashboard, performance, CRM, ticketing.	RLS allows inserting leads, accounts, contacts, targets; triage leads; cannot claim leads (sales only).	UI restricts some actions: marketing cannot create accounts in UI; RLS allows it.
sales manager	Menus: dashboard, performance, CRM, ticketing, DSO.	RLS allows reading/updating accounts, leads, opportunities; can claim leads; can call RPCs.	UI missing features: cannot view “my leads” page; pipeline only shows pipeline view.
salesperson	Menus: dashboard, performance, CRM, ticketing, DSO.	RLS allows select/insert/update on leads, opportunities; can claim leads; cannot see `lead_inbox` marketing page.	UI shows only sales inbox; there is no dedicated page for leads they own.
sales support	Menus: CRM, ticketing.	RLS allows reading leads, targets; cannot triage; can insert targets; cannot update leads.	UI missing import management; cannot create leads as RLS allows.
EXIM Ops / domestics Ops / Import DTD Ops / traffic & warehous	Menus: dashboard, ticketing.	RLS for CRM tables not included, so they cannot access leads/ opportunities.	UI does not display CRM menu, so alignment good.
finance	Menus: dashboard, performance, CRM, DSO.	RLS allows select leads and opportunities for invoice/dso; cannot update.	UI has no DSO or invoice page implemented.

Role	UI access (permissions.ts)	RLS policy privileges	Mismatches & notes
Mismatches	–	RLS often grants capabilities (insert accounts, update leads) that UI lacks. API handlers enforce narrower checks (e.g., triage roles) but PATCH `/api/crm/leads/[id]` does not validate role.	Danger of privilege escalation via API.

6. Workflow Completeness (7 workflows; minimum 14 findings)

1. Marketing inbound lead → triage → qualified → Send to Sales Pool → sales claim → cadence activities → My Work Queue

- **Lead creation** – Inbound forms call POST `/api/crm/leads` (not in UI). The API inserts lead with `status=new` & `triage_status=new` [1].
- **Triage** – Marketing triages via `PATCH /api/crm/leads/{id}/triage` to `Qualified`. Server sets `qualified_at` & `handover_eligible=true` but triage status remains `Qualified` [15].
- **Visibility gap** – Lead disappears from Inbox because the query `view=inbox` only includes `New` and `In Review` [1]. There is no view for `Qualified`.
- **Handover** – Marketing must click separate "Send to Sales Pool" button to call `/api/crm/leads/{id}/handover`. This calls RPC `rpc_lead_handover_to_sales_pool` to set `handed_over_at` and `handover_eligible` [16].
- **Sales Pool** – Sales inbox shows leads from `lead_handover_pool` view where `handover_eligible=true` & `sales_owner_user_id IS NULL`.
- **Claim** – Sales claims via POST `/api/crm/leads/{id}/claim` invoking `rpc_sales_claim_lead` to set `sales_owner_user_id` [7].
- **Next step** – After claim, there is no automatic opportunity creation or cadence. The quick add prospect path is not used here. Sales must manually convert lead to opportunity (no UI route). Thus the lead remains invisible.
- **My Work Queue** – Sales inbox lists overdue activities; claimed lead remains invisible because no opportunity or activity exists.

2. Sales self-sourced → Quick Create → account/contact/opportunity → planned activities → appears in My Work Queue & pipeline

- **Quick Add** – Sales uses Quick Add button on pipeline page; triggers POST `/api/crm/opportunities` with `quick_add` flag causing RPC `rpc_sales_quick_add_prospect` [9].
- **DB actions** – RPC creates account, contact, lead (status `qualified`), opportunity with stage `SQL Qualified`, next steps and cadence activities [21].
- **State transitions** – Lead's status becomes `converted`, triage status remains `qualified`, and `sales_owner_user_id` is set. Opportunity appears in pipeline list; activities appear in My Work Queue.

- **Visibility** – Works as expected: pipeline page shows new opportunity; sales inbox shows new cadence activities.
- **Edge** – The lead is not visible anywhere but its role is now redundant. It may still appear in RLS but not in UI; this is acceptable.

3. Prospecting Targets import → outreach status updates → convert target → lead + opportunity → appears correctly

- **Import** – POST `/api/crm/targets` upserts targets; import route stub not implemented in UI. Import_batches table & RLS exist but missing UI.
- **Outreach** – Sales updates target status via UI (not fully implemented). GET `/api/crm/targets` filter `status` [11].
- **Convert** – Clicking Convert triggers POST `/api/crm/targets/{id}/convert` calling RPC `rpc_target_convert_to_lead` [12].
- **State transitions** – RPC creates account/contact/lead/opportunity from prospecting target; updates target status to `converted`. However, triage status remains `qualified` and `handover_eligible` remains null [21]. The new lead does not appear in handover pool.
- **Visibility** – The resulting opportunity appears in pipeline & activities show in My Work Queue, but the lead record is invisible. UI does not navigate to account/opportunity; there is no success message.

4. Opportunity stage change guardrails

- **UI** – Pipeline page allows stage drag/drop; user selects new stage and next step in modal [8].
- **API** – POST `/api/crm/opportunities/{id}/stage` calls RPC `rpc_opportunity_change_stage` which checks for required next step/due date and quote requirement [19].
- **Guardrails** – If next step missing, RPC throws error. UI does not catch the error gracefully; may swallow with toast.
- **Visibility** – Stage change updates appear immediately in pipeline list because GET opportunities uses stage state. Overdue list not implemented.

5. Activity lifecycle

- **Plan** – Users create planned activities via POST `/api/crm/activities` (used in Activities Planner). Quick Add prospect also creates initial cadence activities.
- **Complete** – POST `/api/crm/activities/{id}/complete` calls RPC `rpc_activity_complete_and_next` [18]. This marks `status=done` and optionally inserts next activity.
- **Visibility** – Completed activities disappear from planner but do not appear in any “history” list. No timeline view per lead/opportunity.

6. Account 360 visibility

- **Account creation** – Accounts created via quick add prospect; there is no dedicated create page.
- **Tenure/Activity status** – `Account.tenure_status` and `activity_status` computed by SQL functions from invoice ledger and displayed as badges; not editable.

- **Account 360** – `/crm/accounts/[id]` shows account details, list of opportunities, activities, contacts, and invoice summary. Contacts add button has no handler; invoice snapshot is stub.
- **Integration** – No integration with ticketing or DSO yet; account view does not fetch tickets or AR/DSO data.

7. Winback/reactivation

- **Detection** – Not implemented in UI. Migrations show concept: `tenure_status` computed from `first_invoice_at`; `activity_status` computed from `last_invoice_at`; there is concept of `winback_start_at`. Cadence for reactivation is planned but not implemented.
- **Reactivate** – Sales has to enroll in reactivation cadence manually; no UI or API exists.

7. Sync Gaps UI ↔ API ↔ DB (minimum 12 findings)

1. **Lead qualification disappears** – UI uses `/api/crm/leads?view=inbox` which excludes `Qualified` leads [1]. The triage call sets `triage_status=Qualified` but does not mark lead as handed over; thus the record is not returned by any list and is effectively invisible [2].
2. **RPC `rpc_lead_handover_to_sales_pool` not triggered automatically** – Triaging a lead to `Qualified` does not call this RPC; the UI relies on user clicking the separate “Send to Sales Pool” button. Without doing so, the lead never appears in sales inbox.
3. **Lead conversion via quick add sets `status=converted` but leaves `triage_status=qualified`** – Such leads are not returned by GET leads (New/In Review only). There is no UI that lists converted leads; only the opportunity remains.
4. **Target conversion leaves lead invisible** – `rpc_target_convert_to_lead` sets triage `qualified` but not `handover_eligible` [21]. Thus the new lead never appears in handover pool. Only the opportunity is visible.
5. **Claimed leads vanish** – After POST `/api/crm/leads/{id}/claim`, the lead is removed from handover pool. There is no UI to list leads assigned to salesperson; pipeline page lists only opportunities, not leads.
6. **Add Account link misroutes** – Accounts list page’s Add Account button links to `/crm/pipeline` instead of an account creation page, causing confusion.
7. **Targets import stub** – UI shows “Import” button but no route; the API supports import via POST `/api/crm/targets`. This mismatch leaves the feature unused.
8. **Add Contact not implemented** – Accounts 360 page has Add Contact button but no API call; the API (`crm_contacts` policies) allows inserting contacts.
9. **No deletion** – There are no API endpoints or UI to delete leads, targets, or activities. Database lacks cascade tasks for cleanup; manual deletion through DB only.
10. **No error handling** – Stage change RPC returns errors if conditions not met (e.g., quote required) but UI does not show messages clearly; users may think action succeeded.
11. **Inconsistent Next.js route groups** – `app/(protected)` vs `app/(app)` vs `(auth)` cause confusion. Some pages unreachable via sidebar.
12. **Email/password login** – Auth UI uses SSR session but there is no sign-up or invitation flow; user roles must be inserted manually into `profiles` table.

8. Button / Shortcut Trace List (minimum 20 findings)

Page / file	Button / CTA	OnClick handler (code path)	Target route/API
Lead Inbox `page.tsx`	**Qualify** (triage) button in row actions	`handleTriage(id, "Qualified")` -> `fetch("/api/crm/leads/${id}/triage",{method:"PATCH",body:{triage_status:"Qualified"}})` 6	`/api/crm/leads/[id]/triage`
Lead Inbox	**Nurture**, **Disqualify** triage buttons	Similar call to triage with other statuses 6	`/api/crm/leads/[id]/triage`
Lead Inbox	**Send to Sales Pool** button (only when status = Qualified)	`handleHandover(id)` -> POST `/api/crm/leads/${id}/handover` 23	`/api/crm/leads/[id]/handover` RPC `rpc_lead_handover_to_sales_pool`
Sales Inbox	**Claim Lead** button	`handleClaimLead` -> POST `/api/crm/leads/${leadId}/claim` 7	`rpc_sales_claim_lead`
Sales Inbox	**Complete Activity** button	`handleCompleteActivity(activityId)` -> POST `/api/crm/activities/${activityId}/complete` with body `outcome: "Completed"` 18	`rpc_activity_complete_and_next`
Sales Inbox	**+** quick create FAB	`Link href="/crm/pipeline"` (plus icon)	Navigates to pipeline page
Pipeline	**Add Opportunity** / Quick Add button	Opens modal; `handleQuickAdd` calls POST `/api/crm/opportunities` with `quick_add=true` 9	`rpc_sales_quick_add_prospect`
Pipeline	**Drag & Drop Stage** on card	`handleStageChange(opportunityId, newStage)`; after confirm, calls POST `/api/crm/opportunities/${id}/stage` 8	`rpc_opportunity_change_stage`
Pipeline	**Stage Change Modal Save** button	Part of `StageChangeModal` -> calls `handleStageChange`	Same as above

Page / file	Button / CTA	OnClick handler (code path)	Target route/API
Pipeline	**Edit Opportunity** (ellipsis menu)	Not implemented; ellipsis shows placeholder.	None
Pipeline	**Delete Opportunity**	Not implemented.	None
Pipeline	**Quick Add** fields: company, contact, next step, due date	Input fields; call `handleQuickAdd`	RPC
Account list	**Add Account** button	`Link href="/crm/pipeline"` inside `accounts/page.tsx`	Should link to account creation but mislinked to pipeline
Account 360	**Add Opportunity** button	`Link href="/crm/pipeline?account_id=..."`	Navigates to pipeline page with query param
Account 360	**Add Contact** button (plus icon)	No handler; button present but not clickable	None
Account 360	**Opportunity cards** link	Not implemented; row is plain text.	None
Targets list	**Add Target** button	Opens modal; calls POST `/api/crm/targets` [20]	Upsert into `crm_prospecting_targets`
Targets list	**Convert** action (arrow)	`handleConvert` -> POST `/api/crm/targets/${id}/convert` [12]	`rpc_target_convert_to_lead`
Targets list	**Import** button	OnClick shows alert (“TODO”).	None
Activities planner	**Add Activity** button	Opens modal; calls POST `/api/crm/activities` [10]	Insert into `crm_activities`
Activities planner	**Complete** button on activity card	Same as Sales Inbox; calls `/api/crm/activities/{id}/complete`	Works; but no ability to create next step from this view.
Generic	**Sidebar links**	Provided by `permissions.ts` mapping; some links lead to placeholder pages (e.g., imports)	Points to incomplete pages

9. State → Destination Map (minimum 15 findings)

Entity / State change	Trigger UI control (file)	API/RPC & DB mutation	Expected destination pages	Actual behaviour & gaps
Lead status = New (default)	Automatic from POST `/api/crm/leads` (not visible in UI)	DB: `crm_leads.status='new', triage_status='new'` [1]	Lead Inbox (inbox)	Appears in Lead Inbox (view=inbox).
Triage → In Review	Lead Inbox triage button	`PATCH /api/crm/leads/{id}/triage` sets `triage_status='In Review'`	Lead Inbox	Remains in Lead Inbox (filter includes In Review).
Triage → Qualified	Lead Inbox triage button	`PATCH /api/crm/leads/{id}/triage` sets `triage_status='Qualified'`, `handover_eligible=true` [15]	Should remain visible until handed over; or show in Sales Pool	Lead disappears from Lead Inbox and does not appear anywhere until manual handover. Bug.
Send to Sales Pool	Lead Inbox handover button	`POST /api/crm/leads/{id}/handover` calls `rpc_lead_handover_to_sales_pool` to set `handed_over_at`, `handover_eligible`, and maybe update triage status	Sales Inbox (handover pool view)	Appears in Sales Inbox under "New Handover."
Lead claimed	Sales Inbox claim button	`POST /api/crm/leads/{id}/claim` sets `sales_owner_user_id` [7]	My Work Queue, pipeline after opportunity creation	Lead removed from Sales Inbox; no list of claimed leads; pipeline remains empty until opportunity created.

Entity / State change	Trigger UI control (file)	API/RPC & DB mutation	Expected destination pages	Actual behaviour & gaps
Lead converted to opportunity (quick add prospect)	Quick Add on pipeline	`rpc_sales_quick_add_prospect` sets `crm_leads.status='converted'`, `triage_status='qualified'`, sets `linked_opportunity_id` [21]	Pipeline board & My Work Queue (activities)	Opportunity appears in pipeline & activities show in work queue; lead no longer relevant.
Lead converted via target convert	Convert target button on Targets page	`rpc_target_convert_to_lead` sets `lead.status='qualified'`, leaves triage `qualified`, not `handover_eligible` [21]	Should appear in Sales Inbox or pipeline	Only opportunity appears; lead invisible.
Lead disqualified	Triage → Disqualified	`PATCH /api/crm/leads/{id}/triage` sets `triage_status='Disqualified'`	Possibly Nurture view (not implemented)	Lead disappears; there is no page to list disqualified leads.
Lead nurtured	Triage → Nurture	`PATCH /api/crm/leads/{id}/triage` sets `triage_status='Nurture'`	Should be in nurture list	No nurture page exists; lead disappears.
Opportunity stage moves	Drag & drop in pipeline board	`rpc_opportunity_change_stage` updates `crm_opportunities.stage`, `next_step`, `next_step_due_at` [19]	Pipeline board & My Work Queue	Works; but no timeline of stage history.
Opportunity won / lost	Stage change to `Won (Onboarding)` or `Lost / Nurture`	RPC sets `won_at` or `lost_at`, `next_step` may be removed	Should be removed from pipeline; appear in performance dashboard and Account 360	Pipeline view hides terminal stages; no reporting view exists.
Activity status = done	Complete button	`rpc_activity_complete_and_next` sets `status='done'` [18]	Planner & Sales Inbox	Disappears from lists; no history view.

Entity / State change	Trigger UI control (file)	API/RPC & DB mutation	Expected destination pages	Actual behaviour & gaps
Target status transitions (`new_target`, `contacted`, `engaged`, `qualified`, `dropped`)	Buttons in targets page (not fully implemented)	Update `crm_prospecting_targets.status` via direct update (API not implemented)	Targets page with filter by status	Only `new_target` filter is used; others not fully surfaced.
Target convert → converted	Convert button	`rpc_target_convert_to_lead` sets `status='converted'`	Removed from targets list	Works; but new records not shown.
Account created	Quick Add prospect or target convert	Insert into `crm_accounts` via RPC; `owner_user_id` set	Accounts page (view=enriched)	Appears in Accounts list.
Account updated	API `/api/crm/accounts/[id]` PATCH	Update account fields; not implemented in UI	Should update Account 360 & lists	Hidden; patch accessible via API only.
Contact created	RPC quick add or target convert; Add contact button (missing)	Insert into `crm_contacts`	Account 360	Not implemented in UI.

10. Query/Filter Audit (why records disappear)

List view	Query (filter)	Evidence & notes	Impact
Lead Inbox (view=inbox)	GET `/api/crm/leads?view=inbox&pageSize=100` → SQL filters: `triage_status IN ('New','In Review')` [1]	Excludes `Qualified`, `Nurture`, `Disqualified`, `Handed Over`.	Leads change to `Qualified` vanish.
Sales Inbox (handover pool)	GET `/api/crm/leads?view=handover_pool&pageSize=10` → uses `lead_handover_pool` view where `handover_eligible=true` & `sales_owner_user_id IS NULL`	Excludes qualified leads that have not been handed over; excludes claimed leads.	Qualified leads not handed over vanish; claimed leads vanish.

List view	Query (filter)	Evidence & notes	Impact
My Work Queue (activities)	GET `/api/crm/activities?view=inbox&pageSize=50` → filter `status='planned'` & `next_step_due_at < now()` [17]	Only overdue planned activities show; does not display upcoming tasks.	Sales may miss scheduled tasks not yet overdue.
Pipeline view	GET `/api/crm/opportunities?view=pipeline` → filter excludes final stages (won/lost).	Only active pipeline shows; no overdue or assigned leads list.	Claimed leads without opportunities vanish.
Opportunities overdue	GET `?view=my_overdue` exists but not used in UI [8].	Could show overdue next steps; but not wired.	Missed feature.
Accounts list	GET `/api/crm/accounts?view=enriched`	No filter by owner; returns all accounts accessible via RLS.	Sales support sees all accounts; may be overwhelmed.
Targets list	GET `/api/crm/targets?status=${status}&search=${search}` [11]	Only filters by status & search; cannot filter by owner except implicitly through RLS.	Good.
Activities planner	GET `/api/crm/activities?view=planner` → returns planned & done; sorts by date [17]	Doesn't filter by entity_type; planner may show mixed activities (leads, opportunities, accounts).	Hard to track context.

11. Security & Data Integrity

- **RLS & API mismatch** – Some API endpoints validate roles (e.g., triage restricted to marketing) but the underlying RLS policies allow owners to update leads; a malicious user could bypass UI and call PATCH `/api/crm/leads/{id}/triage` to change statuses. Align API checks with RLS or rely solely on RLS to avoid confusion.
- **No rate limiting** – API endpoints accept unlimited POST requests; idempotency helps but there is no throttling. Consider adding rate limits via Supabase Edge Functions or Next.js middleware.
- **Idempotency** – Good practice: RPCs insert into `crm_idempotency` to ensure repeat calls don't duplicate records [21].

- **Data dedupe** – Functions `crm_match_account` and `crm_match_contact` normalize names and extract domains; ensures single truth for accounts/contacts [21]. However, duplicates may still occur via direct inserts (e.g., `crm_accounts` insert API). UI should enforce dedupe before insertion.
- **Constraint coverage** – Unique indexes on NPWP, domain, contact email/phone; but no unique constraint on `crm_leads` (duplication possible).
- **Logging & audit** – Migration adds audit triggers; but only RPCs log actions; direct API updates may not log properly.
- **Sensitive data** – The database holds contact phone numbers, emails; ensure encryption at rest and restrict RLS appropriately.
- **Service role keys** – Service role keys must not be exposed in frontend; the user inadvertently shared them. Remove from `.env.local` on client side.

12. Fix Plan & Best Practices (prioritized)

1. **Resolve qualified lead disappearance** (P0):
2. *Option A:* Modify `PATCH /api/crm/leads/{id}/triage` to automatically call `rpc_lead_handover_to_sales_pool` when triage status becomes `Qualified`. This sets `handover_eligible` and moves lead to Sales Pool. UI would no longer require separate "Send to Sales Pool" click.
3. *Option B:* Change Lead Inbox query to include `Qualified` leads with `handover_eligible=false` and display them in an "Awaiting Handover" section. Provide a prominent "Send to Sales Pool" CTA.
4. *Option C:* Add a new "Qualified Leads" page to hold leads not yet handed over.
5. **Implement missing pages & actions** (P1–P2):
6. **Account creation page** – Provide `/crm/accounts/new` or a modal on accounts list; calls API to create account via `crm_accounts` insert; use RLS for marketing/sales roles.
7. **Add Contact** – Implement POST `/api/crm/contacts` API and UI form in Account 360; update accounts page accordingly.
8. **Prospects page** – Remove duplicate `prospects` placeholder or make it a list of leads and opportunities for sales.
9. **Imports** – Build UI for `import_batches` to upload CSV, preview, and call `/api/crm/targets` for bulk insert.
10. **Align UI queries with states** (P1):
11. Adjust Lead Inbox, Sales Inbox, Pipeline queries to include additional statuses or provide tabs (e.g., "New/In Review", "Qualified", "Nurture", "Disqualified").
12. Provide a "My Leads" view showing claimed leads not yet converted to opportunities.
13. Use pipeline `view=my_overdue` to display overdue opportunities; add filter by owner.
14. **Enhance workflows** (P1):

15. After claim or conversion, automatically navigate to the new opportunity or account; show toast with link.
16. In target conversion, set `handover_eligible=true` or directly create opportunity and mark lead `converted` to avoid orphan lead.
17. Provide reactivation/Winback automation: detect inactive accounts and enroll cadence; show in Sales Inbox.
18. **Improve error handling & UX** (P2):
19. Surface RPC errors from stage change or conversion modals as toast notifications.
20. Validate Quick Add & Convert forms; show field errors.
21. Add deletion/archive functionality for leads, activities, targets with soft-delete flag and RLS.
22. **Strengthen security & data integrity** (P1):
23. Move sensitive service role keys out of client code; store only on server side.
24. Align RLS and API role checks; remove redundant checks or enforce RLS in API routes.
25. Add indexes on `crm_leads.sales_owner_user_id` to speed up owner queries.
26. Audit logging: ensure PATCH `/api/crm/leads/[id]` and other direct updates insert audit records.
27. **Refactor route structure** (P3):
28. Consolidate route groups into `(app)` with clear subfolders (`crm/leads`, `crm/accounts`, etc.). Avoid mixing `(protected)` and `(app)` naming.
29. Provide canonical paths for each entity.
30. **Add analytics & dashboards** (P3):
31. Implement performance dashboards (formerly KPI) reading from invoice ledger, activities, leads, and opportunities.
32. Provide Win rate, response time, pipeline metrics.

By addressing these items systematically—from qualified lead disappearance to missing pages and inconsistent filtering—the UGC BCP CRM can become operational, transparent, and user-friendly.

---

1 route.ts
https://github.com/assujiar/ugc-bcp-2/blob/main/app/api/crm/leads/route.ts

2 6 23 page.tsx
https://github.com/assujiar/ugc-bcp-2/blob/main/app/(protected)/crm/lead-inbox/page.tsx

3 15 route.ts
https://github.com/assujiar/ugc-bcp-2/blob/main/app/api/crm/leads/[id]/triage/route.ts

4 5 database.ts
https://github.com/assujiar/ugc-bcp-2/blob/main/lib/types/database.ts

7 route.ts
https://github.com/assujiar/ugc-bcp-2/blob/main/app/api/crm/leads/[id]/claim/route.ts

8 9 route.ts
https://github.com/assujiar/ugc-bcp-2/blob/main/app/api/crm/opportunities/route.ts

10 page.tsx
https://github.com/assujiar/ugc-bcp-2/blob/main/app/(protected)/crm/activities/page.tsx

11 20 route.ts
https://github.com/assujiar/ugc-bcp-2/blob/main/app/api/crm/targets/route.ts

12 route.ts
https://github.com/assujiar/ugc-bcp-2/blob/main/app/api/crm/targets/[id]/convert/route.ts

13 page.tsx
https://github.com/assujiar/ugc-bcp-2/blob/main/app/(protected)/crm/page.tsx

14 route.ts
https://github.com/assujiar/ugc-bcp-2/blob/main/app/api/crm/leads/[id]/route.ts

16 route.ts
https://github.com/assujiar/ugc-bcp-2/blob/main/app/api/crm/leads/[id]/handover/route.ts

17 route.ts
https://github.com/assujiar/ugc-bcp-2/blob/main/app/api/crm/activities/route.ts

18 route.ts
https://github.com/assujiar/ugc-bcp-2/blob/main/app/api/crm/activities/[id]/complete/route.ts

19 route.ts
https://github.com/assujiar/ugc-bcp-2/blob/main/app/api/crm/opportunities/[id]/stage/route.ts

21 22 10_crm_rpc_rls.sql
https://github.com/assujiar/ugc-bcp-2/blob/main/supabase/migrations/10_crm_rpc_rls.sql