

1) هل تقرير الـ Route Table اللي بعته “صحيح”؟

أيوه، بنسبة كبيرة جدًا مطابق للكود الموجود في الـ ZIP (نفس المسارات ونفس أماكن الملفات تقريبًا).

لكن وأنا براجع الكود نفسه لقيت شوية حاجات زيادة / أخطاء ربط مهمين (مش ظاهرين في التقرير أو محتاجين توضيح):

✓ أخطاء ربط (Navigation) إضافية لازم تتسجل

1. Redirect لصفحة مش موجودة

- في: `src/app/property/[id]/booking/confirmation/page.tsx`
- فيه:

◦ `('router.push('/profile/bookings`

- بس مفيش **route** اسمه `profile/bookings/` في المشروع الموجود: `profile/` و `bookings/` فقط
→ الحل: يا إما تغيّر ها لـ `bookings/`، يا إما تنشئ route جديد `profile/bookings/`.

2. `add-property/` مش محمي فعليًا

- الملف: `src/app/add-property/page.tsx`

- مستورد `ProtectedRoute` ... بس مش مستخدم
→ يعني أي حد يقدر يفتح صفحة إضافة عقار.

3. `bookings/` فيه لينك غلط

- الملف: `src/app/bookings/page.tsx`

- عامل لينك: `{properties/${id}/`

- الصح في مشروعك: `{property/${id}/`

4. خطر/غلط في **Unlock Logic**

- في: `src/services/supabaseService.ts` داخل `unlockProperty()`
- عامل:

◦ `('from('unlocked_properties').select('id.`

- بس حسب `schema.sql` جدول `unlocked_properties` مافيهوش عمود `id` (هو Composite (PK: `user_id`, `property_id` → ده غالبًا هيكسر الاستعلام إلا لو DB عندك اتغيرت عن `schema.sql`.

5. Mismatch كبير في نظام الـ Auth

- عندك مسارين Auth مختلفين:

○ `login/` و `register/` بيستخدموا `login/register` من `AuthContext`

○ `auth/` بيستخدم `components/auth/LoginForm.tsx` و `components/auth/SignUpForm.tsx` و دول بينادوا على `useAuth()` بس متوقعين دوال مش موجودة في `AuthContext` زي: `signIn` / `signUp` / `signInWithGoogle` / `signInWithFacebook`

- ده معناه غالبًا المشروع مش هيبني TypeScript بنجاح إلا لو أنت عامل bypass بطريقة ما.
- الأفضل: توحيد `auth flow` (إما تعتمد `auth/` وتكمل `AuthContext`، أو تشيل `auth/` وتخليك على `login/` و `register/`).

2) “مين الملفات المسؤولة عن الصفحات والربط بينهم؟” (Cheat Sheet سريع)

(A) ملفات الصفحات نفسها (Routes)

أي صفحة في Next App Router بتبقى هنا:

- `src/app/**/page.tsx` → صفحة Route
- `src/app/**/layout.tsx` → Layout بيغلف Routes تحته
- `.../[Dynamic routes: src/app/property/[id`

(B) الملفات اللي “بتربط” الصفحات ببعض (Navigation)

دي أهم الملفات اللي بتعمل Links/Buttons بين الصفحات:

1. Bottom Navigation

- `src/components/BottomNav.tsx`
- `favorites /add-property /bookings /profile/ /` يربط:

2. Header

- `src/components/Header.tsx`
- يربط لحساب المستخدم + تسجيل الدخول/التسجيل (وفيهِ notifications بس حالياً معتمدة على `lib/storage`)

3. Property Cards / Lists

- غالباً:
- `src/components/PropertyCard.tsx` (لو موجود عندك)
- `<Link href="/property/${id}>` component يعمل أو أي

4. تفاصيل العقار

- `src/app/property/[id]/client.tsx`
- فيه توجيه لـ `auth/` لو المستخدم مش مسجل وعائز `unlock`

5. تدفق الحجز

- `src/app/property/[id]/booking/client.tsx`
- يعمل:
- `router.push(/property/${id}/booking/confirmation?bookingI`
- `(...=d`

6. صفحة الحجوزات

- `src/app/bookings/page.tsx`
- فيها `links` للرسائل + (لينك `property` غلط زي ما قلنا)

(C) ملفات "مصادر الداتا" اللي كل الصفحات راکبة عليها

دي أهم 3 ملفات بتحدد الداتا هتيجي منين:

1. Supabase Client

- `src/lib/supabase.ts`

2. **Supabase Service** (المصدر الأساسي للاستعلامات)

- `src/services/supabaseService.ts`

3. **Mock/Local Storage Layer**

- `src/lib/storage.ts`

(ودي سبب إن صفحات زي Home/Bookings/Notifications لسه Mock في أجزاء)

D) Guards / Protection

- `src/components/ProtectedRoute.tsx` (بيروح `...=login?redirect/`)

- `src/components/auth/AdminGuard.tsx` (بيروح `auth/` أو `/`)

- `src/middleware.ts` (حاليًا **Security headers** فقط)

3) برومبتات جاهزة تديها لـ “جينا” علشان تطلعك خريطة الملفات والربط بينهم

انسخ/الصق أي Prompt من دول زي ما هو:

Route Table — Prompt 1 تلقائي + Server/Client

افحص مشروع (Next.js App Router) داخل `src/app`.
اعمل جدول Routes شامل لكل `page.tsx` و `layout.tsx` و dynamic segments مثل `[id]`.
لكل Route اكتب:

1. route path

2. file path

3. هل الملف فيه `'use client'` ولا Server Component

4. أهم (components/services imports)

5. أهم (supabaseService data source ؟storage ؟mock array)

(Prompt 2 — Navigation Graph (Link + router.push

اعمل تحليل للـ navigation داخل المشروع:

- استخرج كل `<"..."=Link href>`
- واستخرج كل `(...)router.push` و `(...)router.replace`
- وحدد الصفحة/الملف الذي عامل التوجيه.
في الآخر اطبع Graph على شكل:
`(FROM_ROUTE -> TO_ROUTE (file:line or file`
وكمان اعمل Mermaid diagram لو تقدر.

Prompt 3 — كشف الروابط المكسورة (Broken Routes)

- راجع كل `"..."=href` وكل `(...)router.push` (`".../"`) في المشروع.
قارنها فعليًا بالـ `routes` الموجودة في `src/app`.
اطبع قائمة بالروابط التي بتشير لمسارات غير موجودة (404).
واقترح التصحيح الأقرب (مثال: `/property/[id]` -> `[properties/[id]`).

Prompt 4 — Mock vs Supabase Coverage

- صنّف كل صفحة في `src/app` إلى واحدة من:
 - Supabase fully
 - Hybrid (جزء supabase وجزء mock/localStorage)
 - Mock only
- مع ذكر الدوال/المصادر المستخدمة في كل صفحة (مثلاً:
`supabaseService.getProperties` أو `arrays` ثابتة).

Prompt 5 — Booking Flow Compliance مع V1

- راجع تدفق الحجز في:
 - `*/src/app/property/[id]/booking`
 - `src/services/supabaseService.ts`
(`createBooking/checkAvailability/calculateTotalPrice`)
 - `(functions.sql` (`create_atomic_booking`)
 - `schema.sql` + migrations الخاصة بالحجوزات
وقارنهم مع مواصفات (booking statuses) V1 + منع التداخل `confirmed+active` فقط +
(payment flow).
اطبع:

1. أين عدم التوافق بالضبط (file + snippet)

2. patch مقترح (minimal) SQL + TS.

Prompt 6 — Auth System Unification

راجع كل الصفحات المتعلقة بـ `AuthContext` + `/auth`, `/login`, `/register` استخرج كل الدوال التي يتم استدعاؤها من `useAuth()` في المشروع. قارنها بـ `methods` المتاحة فعلياً في `AuthContextType`. اطلع `mismatches` وقدم خطة توحيد:

- خيار A: دعم `auth/` بالكامل
- خيار B: حذف `auth/` وتوحيد `login/` و `register/` مع قائمة بالتعديلات المطلوبة.

4) لو عايز "أقفل الدائرة" بسرعة: 6 Fixes أولوية (مباشرين)

لو هدفك إن المشروع يبقى متماسك بسرعة قبل أي تطوير كبير:

1. تصليح لينك `{properties/{id}} → /property/{id}`
2. تصليح `/bookings → /profile/bookings` (أو إنشاء route جديد) `redirect`
3. حماية `add-property/` بـ `<ProtectedRoute>` فعلياً
4. توحيد `Auth`: إمّا تكمل `auth/` + تعدّل `AuthContext`، أو تلغي `auth/`
5. تحديث RLS bookings من `guest_id` → `user_id` (عندك migration عمل `rename`)
6. تعديل `availability check` من `('confirmed', 'pending')` → `('confirmed', 'active')` (زي تقريرك)

Gamasa Properties — Codebase Analysis Report #

1. Route Table

| Server / Client | نوع الملف | Route | File Path |

```

|-----|-----|-----|-----|
| **src/app/page.tsx` | page | **Server` | `/` |
| **layout` (root) | `src/app/layout.tsx` | layout | **Server/` |
add-property` | `src/app/add-property/page.tsx` | page | Client/` |
| (`('use client
| admin` | `src/app/admin/page.tsx` | page | Client/` |
admin` (layout) | `src/app/admin/layout.tsx` | layout | **Server**/` |
| ((wraps AdminGuard
admin/payments` | `src/app/admin/payments/page.tsx` | page | Client/` |
|
admin/properties` | `src/app/admin/properties/page.tsx` | page | `/` |
| Client
| admin/users` | `src/app/admin/users/page.tsx` | page | Client/` |
| auth` | `src/app/auth/page.tsx` | page | Client/` |
bookings` | `src/app/bookings/page.tsx` | page | Client (`'use/` |
| (`'client
| **favorites` | `src/app/favorites/page.tsx` | page | **Server/` |
favorites` (client) | `src/app/favorites/FavoritesClient.tsx` | /` |
| client component | Client
| login` | `src/app/login/page.tsx` | page | Client/` |
| messages` | `src/app/messages/page.tsx` | page | Client/` |
| messages/[id]` | `src/app/messages/[id]/page.tsx` | page | Client/` |
| my-properties` | `src/app/my-properties/page.tsx` | page | Client/` |
| notifications` | `src/app/notifications/page.tsx` | page | Client/` |
| profile` | `src/app/profile/page.tsx` | page | Client/` |
property/[id]` | `src/app/property/[id]/page.tsx` | page | /` |
| ****Server
property/[id]` (details) | `src/app/property/[id]/client.tsx` | /` |
| client component | Client
property/[id]/booking` | `src/app/property/[id]/booking/page.tsx` | /` |
| **page | **Server
property/[id]/booking` (form) | /` |
`src/app/property/[id]/booking/client.tsx` | client component | Client
| (`('use client
property/[id]/booking/confirmation` | /` |
`src/app/property/[id]/booking/confirmation/page.tsx` | page | Client
| (`('use client
| register` | `src/app/register/page.tsx` | page | Client/` |
| **search` | `src/app/search/page.tsx` | page | **Server/` |
search` (client) | `src/app/search/client.tsx` | client component | /` |
| Client
robots.ts` | `src/app/robots.ts` | route handler (robots.txt) | ` |
| Server

```

```

sitemap.ts` | `src/app/sitemap.ts` | route handler (sitemap.xml) | ` |
                                                    | Server
providers.tsx` | `src/app/providers.tsx` | special (Provider` |
                                                    | wrapper) | Client
                                                    ---

(Route Tree (Layout Nesting .2 ##


...

RootLayout (src/app/layout.tsx) - lang="ar", dir="rtl", wraps Providers
+ IOSInstallPrompt
page.tsx (Server - mock data, no → / —|
(Supabase
search → page.tsx (Server) → SearchClient.tsx/ —|
property/[id] → page.tsx (Server, fetches from/ —|
(Supabase
(client.tsx (Client component for interactivity —| |
booking → page.tsx (Server, fetches property from/ —L |
(Supabase
client.tsx (Client, handles booking form + —| |
(createBooking RPC
confirmation → page.tsx (Client, fetches booking +/ —L |
(receipt upload
(add-property → page.tsx (Client/ —|
bookings → page.tsx (Client - MOCK DATA ONLY, no/ —|
(Supabase
favorites → page.tsx (Server) → FavoritesClient.tsx/ —|
(my-properties → page.tsx (Client/ —|
(profile → page.tsx (Client/ —|
(notifications → page.tsx (Client/ —|
(messages → page.tsx (Client/ —|
(messages/[id] → page.tsx (Client/ —L |
(auth → page.tsx (Client/ —|
(login → page.tsx (Client/ —|
(register → page.tsx (Client/ —|
<AdminLayout (src/app/admin/layout.tsx) - wraps <AdminGuard —L
(admin → page.tsx (Client/ —|
(admin/payments → page.tsx (Client/ —|
(admin/properties → page.tsx (Client/ —|
(admin/users → page.tsx (Client/ —L
...

```


Route Guards & Redirects .3

Location	الآلية	المسار المحمي	يعمل كيف؟
****auth guard** لا يوجد ****** فقط - src/middleware.ts	Security Headers		
جميع المسارات	يضيف CSP + X-Frame	ثم `()NextResponse.next` إلخ.	
src/app/admin/layout.tsx	<AdminGuard> component	`/admin/**`	
Client-side: useEffect → redirect to `/auth` if not logged in, redirect			
to `/` if not admin			
src/components/auth/AdminGuard.tsx	`useAuth()` + `		
, `('/')router.push` → إذا لم يكن مدير	`**/isAdmin(user?.role)`	`/admin	
`('router.push('/auth` → إذا لم يكن مسجلاً			
src/app/property/[id]/booking/client.tsx | L108 | `user?.id` | |
`'guest'` | `/property/[id]/booking
ك`guest` بدون تسجيل دخول |

< ******  نقص أمني: ****** ال middleware لا يحمي أي مسار. add-property`,/,
.Server بدون حماية على مستوى ال Server
الحماية تعتمد فقط على ال Client (ProtectedRoute/AdminGuard components) مما
يعني أن Server-Side Rendering يحدث دائماً حتى للمسارات المحمية.

Page → Dependencies → Data Sources → Navigation .4

(Home) `/\` ###
Imports: ****** `Link`, `BottomNav`, `Header`, `PropertyCard`, ****** -
`CategoryFilter`, `JsonLd
hardcoded arrays) ****** فقط Data Sources: ******  ****Mock data**** -
(`featuredProperties`, `recentProperties
****Navigation FROM**** -
(`"عرض الكل" + Link href="/search">` (filter button>` -
(section مميز) <"Link href="/search?featured=true">` -
(section حديث) <"Link href="/search?recent=true">` -
(Link href="/add-property">` (CTA>` -
BottomNav, Header من ****Navigation TO**** -

`[property/[id/` (تفاصيل العقار)

```

Imports:** `supabaseService.getPropertyById`,`** -
`supabaseService.incrementPropertyViews`,` `ClientPropertyDetails`,`
`JsonLd`
Data Sources:** Supabase → `properties` table (via** -
(``getPropertyById`
**::Navigation FROM** -
Link href="/property/[id]/booking">` (from ClientPropertyDetails>` -
(presumably
(booking/client.tsx (L184 داخل `())router.back` -
PropertyCard components من **::Navigation TO** -
---

(property/[id]/booking/` ###
Imports:** `supabaseService.getPropertyById`,`** -
`supabaseService.checkAvailability`,` `supabaseService.createBooking`,`
`supabaseService.calculateTotalPrice`,` `useAuth`,` `DateSelector`,`
`TenantForm`,` `PaymentMethods`,` `PriceBreakdown`
**::Data Sources** -
(Supabase: `supabaseService.getPropertyById(id)` (server-side -
Supabase: `supabase.rpc('create_atomic_booking')` (via -
(createBooking
(Supabase: `bookings` table (checkAvailability query -
**::Navigation FROM** -
-
router.push('/property/${propertyId}/booking/confirmation?bookingId=${`
(booking.id}')` (L169
(router.back())` (L184` -
---

(property/[id]/booking/confirmation/` ###
Imports:** `supabaseService.getBookingById`,`** -
`supabaseService.uploadPaymentReceipt`,` `useAuth`,` `useRouter`,`
`useSearchParams`
**::Data Sources** -
Supabase: `bookings` join `properties` join `profiles` (via -
(``getBookingById`
(Supabase: Storage bucket `payment-receipts` (receipt upload -
**::Navigation FROM** -
(router.push('/profile/bookings')` (L283` -
(router.push('/')` (L286` -

```

```

        (الحجوزات) `bookings/` ###
`Imports:** `useState`, `Link`, `BottomNav** -
INITIAL_MY_BOOKINGS`,` - **فقط Data Sources:** ⚠️ **Mock data** -
(`INITIAL_INCOMING_REQUESTS` (hardcoded
**Navigation FROM** -
⚠️ رابط خاطئ! يجب أن يكون `[property/[id/` -
(مراسلة المالك) `<`{Link href="/messages/${booking.id}>` -

```

```

`**/admin/` ###
Layout Guard:** `AdminGuard` → checks `isAdmin(user?.role)` from** -
`AuthContext
Data Sources:** Supabase via `supabaseService` (getAllProfiles,** -
(getProperties, payment_requests
Navigation FROM:** `

```

.5 Navigation Graph (نص)

```

...
(Home) /
(search (Link:L184, L203, L224 in page.tsx/ →|
(add-property (Link:L247 in page.tsx/ →|
(property/[id] (PropertyCard → Link internally/ →|
/ ,BottomNav → /bookings, /favorites, /messages, /profile →L
[property/[id/
(property/[id]/booking (client.tsx → Link/button/ →|
(messages/[id] (via conversation/ →L
property/[id]/booking/
property/[id]/booking/confirmation?bookingId=xxx (client.tsx:L169/ →|
(router.push
router.back() →L
((client.tsx:L184
property/[id]/booking/confirmation/

```

```

(profile/bookings (confirmation/page.tsx:L283 router.push/ →-|
(confirmation/page.tsx:L286 router.push) / →-L

bookings/
(messages/[bookingId] (bookings/page.tsx:L193-L196 Link/ →-L
([Link /properties/[id] (WRONG path - should be /property/[id] ⚠️

(admin (layout/
admin/properties/ →-|
admin/payments/ →-|
admin/users/ →-L

auth | /login/
redirect to / after login (AuthContext logout → →-L
('/'=window.location.href

(* /AdminGuard (all /admin
(auth (if !isAuthenticated/ →-|
(if !isAdmin) / →-L
...

```

6. تحليل قواعد v1 للحجز

القاعدة 1: Availability لا تُخزن داخل properties، وتُشتق فقط من property_unavailability و confirmed+active

```

| الحالة | التطبيق في الكود |
|-----|-----|
| **متوافق جزئياً** | `checkAvailability` في
`supabaseService.ts:L1236-1248` يستعلم من `bookings` table مباشرة |
| **خرق** | `checkAvailability` يفحص `status IN ('confirmed',
'pending')` - يجب أن يكون `confirmed+active` فقط |
| **خرق** | لا يوجد جدول `property_unavailability` في أي من المخططات |
| **خرق** | `create_atomic_booking` في `functions.sql:L131` يستخدم
`status IN ('confirmed', 'pending')` للتحقق من التعارض |
| **خرق** | صفحة `/` تعرض عقارات بـ `status: 'available'` من mock data، لا
تُشتق من الحجوزات |

```

****إصلاح مقترح:****

sql`

-- أضف جدول property_unavailability

```

        ) CREATE TABLE property_unavailability
        ,id UUID DEFAULT gen_random_uuid() PRIMARY KEY
        ,property_id UUID REFERENCES properties(id) ON DELETE CASCADE
        ,start_date DATE NOT NULL
        ,end_date DATE NOT NULL
        ,reason TEXT -- 'maintenance', 'blocked', etc
    ; (

    عدّل checkAvailability لتشمل هذا الجدول
    SELECT COUNT(*) FROM bookings
    WHERE property_id = p_property_id
    AND status IN ('confirmed', 'active') -- V1 compliant
    ; (AND (p_start_date <= end_date AND p_end_date >= start_date
    ...

    ---

    ### القاعدة 2: حالات الحجز المسموحة

    requested / approved / payment_pending / `**المطلوبة:**`
    payment_uploaded / confirmed / active / completed / rejected / expired
    `/ cancelled

    | الحالة | الكود/DB |
    |-----|-----|
    `fix_enum_values.sql:L36_20260208` و `schema.sql:L102` | **خرق كبير** |
    | `pending, confirmed, cancelled, completed` ب: |
    | `bookings/page.tsx:L8` | **خرق** | تعرّف `BookingStatus` محليًا كـ
    | `pending` \| `confirmed` \| `completed` \| `cancelled`
    | `updateBookingStatus` في `supabaseService.ts:L1490` يقبل فقط
    | `pending / confirmed / cancelled`
    | لا يوجد `requested, approved, payment_pending,`
    | `payment_uploaded, active, rejected, expired`

    **إصلاح مقترح (minimal):**
    sql```
    ;ALTER TABLE bookings DROP CONSTRAINT bookings_status_check
    ALTER TABLE bookings ADD CONSTRAINT bookings_status_check
    ) CHECK (status IN
    , 'requested', 'approved', 'payment_pending', 'payment_uploaded'
    'confirmed', 'active', 'completed', 'rejected', 'expired', 'cancelled'
    ; (
    ...

```

```

:TypeScript وفي
typescript``
src/types/index.ts //
= export type BookingStatus
'requested' | 'approved' | 'payment_pending' | 'payment_uploaded' |
  confirmed' | 'active' | 'completed' | 'rejected' | 'expired' | ' |
; 'cancelled
...

---

### القاعدة 3: منع التداخل فقط على confirmed+active

| الحالة | التطبيق |
|-----|-----|
status IN` يمنع التداخل على `create_atomic_booking` | **خرق** |
(pending) الانتظار فترة خلال جوازات جديدة مما يمنع جوازات جديدة خلال فترة الانتظار (pending)
|
| `confirmed + active` فقط | **خرق** |

**إصلاح في `functions.sql`**
sql``
-- في `create_atomic_booking` السطر 131
confirmed',') من -- بدلاً من ('AND status IN ('confirmed', 'active
('pending
...

وكذلك في `checkAvailability` في `supabaseService.ts:L1240`
typescript``
confirmed','] من بدلاً // ([ 'in('status', ['confirmed', 'active.
['pending
...

---

### القاعدة 4: تحصيل العمولة فقط بعد الانتقال لـ active

| الحالة | التطبيق |
|-----|-----|
**غير مطبق** | لا يوجد أي منطق لتحصيل العمولة في الكود |
**خرق** | `calculateTotalPrice` في `supabaseService.ts:L1199` يحسب
`serviceFee = basePrice * 0.10` ويضيفها للإجمالي فور الحجز (pending) |

```



```
create_atomic_booking(...) | `functions.sql:L104` | Booking form → |
  `confirmed+pending` | يمنع التعارض على | لكن يجب
  | state machine transitions فقط. لا يتحقق من
`RLS: `Public profiles are viewable by everyone` | `schema.sql:L189` |
  | | أي استعلام على profiles
RLS: `Users can update own profile` | `schema.sql:L192` | Profile
  | | page
RLS: `Approved properties are viewable by everyone` | |
  | | `schema.sql:L199` | Property listings
RLS: `Users can insert own properties` | `schema.sql:L203` | Add
  | | property page
RLS: `Admins can do everything on properties` | `schema.sql:L308` | |
  | | Admin panel
RLS: `Users can view own bookings` | `schema.sql:L213` | Bookings |
- migration | `user_id` | rename | لكن العمود تم |
  | `guest_id` | يفحص | في |
  | | قد يكون مكسوراً |
RLS: `Users can create bookings` | `schema.sql:L216` | Booking form |
  | | نفس المشكلة - يستخدم `guest_id` القديم |
RLS: `Users can view own payment requests` | `schema.sql:L220` | |
  | | Payment flow
RLS: `Admins can manage payment requests` | `schema.sql:L318` | |
  | | Admin payments page
  idx_payment_requests_property_user_active` (UNIQUE INDEX) | `
  | `20260208000001_add_payment_consumed.sql:L10` | unlockProperty flow
  | | يمنع دفعيتين نشطتين لنفس العقار/المستخدم |
is_consumed` column | `20260208000001_add_payment_consumed.sql:L6` | `
  | | `unlockProperty | يمنع إعادة استخدام نفس الدفع |
```

```
supabase.rpc() ` calls` خريطة ## 8
```

```

| الدالة | يُستدعى من | يتحقق من كذا |
|-----|-----|-----|
| `rpc('increment_views', { property_id })` | `supabaseService.ts:L384` |
| | UPDATE فقط | لا شيء - فقط |
| | `rpc('create_atomic_booking', {...})` | `supabaseService.ts:L1279` |
| | فحص تعارض التواريخ (جزئي - pending مشمول) |
| | `rpc('unlock_property_with_payment', {...})` |
| | `supabaseService.ts:L592` | دفع معتمد + غير مستهلك (✓ حماية كاملة) |

```


9. ملخص المخاطر والإصلاحات

الأولوية	المشكلة	الملف	الإصلاح
● **عالية**	Booking status constraint لا يطابق 4 states V1 فقط بدلاً من		
(9	`schema.sql:L102`, `20260208_fix_enum_values.sql:L34`	تحديث CHECK	
constraint لإضافة `requested, approved, payment_pending,`			
`payment_uploaded, active, rejected, expired`			
● **عالية**	RLS policies على `bookings` تستخدم `guest_id` القديم بعد		
rename إلى `schema.sql:L213,L216`	`user_id`	تحديث الـ policies لاستخدام	
`user_id`			
● **متوسطة**	`create_atomic_booking` + `checkAvailability` يفحصان		
`pending` كتعارض (يجب `confirmed+active` فقط)			
`supabaseService.ts:L1240`, `functions.sql:L131`	تغيير IN		
`('confirmed', 'pending` إلى `('confirmed', 'active`			
● **متوسطة**	العمولة (service_fee) تُحسب وتُضاف للإجمالي فور الحجز وليس عند		
`supabaseService.ts:L1199`	`active`	فصل `service_fee` عن	
`total_amount` حتى transition لـ `active`			
● **متوسطة**	`bookings` page/` تستخدم mock data فقط - لا تستدعي		
`Supabase`	تكامل		
`supabaseService.getUserBookings`			
● **منخفضة**	رابط خاطئ في `properties/[id]`	`bookings/page.tsx:L182`	تصحيح الـ href
● **منخفضة**	home page `/` تستخدم mock data فقط (8 عقارات hardcoded)		
`supabaseService.getProperties`	تكامل		
● **منخفضة**	Middleware لا يحمي أي route على مستوى Server (Add		
`Property, My Properties, Bookings, Profile`	`middleware.ts`	إضافة	
middleware في الـ check			
● **منخفضة**	لا يوجد جدول `property_unavailability`	غير موجود	إضافة
الجدول وتحديث `checkAvailability`			
● **منخفضة**	`AuthContext` لا يستعيد الـ Supabase session في non-mock		
`AuthContext.tsx:L38-44`	`mode`	مزامنة `supabase.auth.getUser` عند	
التحميل |