**✅ Final Summary – TDA-124 (Admin Login & Auth)**

All acceptance criteria and DoD are now fully met and verified in production.

**Implemented & verified features**

* 🔐 **Supabase Auth integration (email/password)** — complete end-to-end flow with secure JWT validation.
* ⚙️ **Backend /api/v1/admin/whoami** — verifies Supabase JWT using SUPABASE\_JWT\_SECRET and checks ALLOWED\_ADMIN\_EMAILS.
* 🧭 **Frontend login & admin protection**
  + Added LoginPage with Supabase signInWithPassword and improved UX (autoComplete attributes, proper error handling).
  + Added useAuth hook for session state, hash-token bootstrap, and persistent login.
  + Added RequireAdmin guard that renders protected routes only when authenticated.
* 🌐 **Routing & deployment**
  + Migrated to HashRouter (no basename) to support GitHub Pages SPA routing — deep links (#/login, #/admin) now work without 404.
  + GH Actions injects Supabase + API env vars into .env.production.
* 🪄 **Magic-link & password recovery** — handled by useAuth, which detects #access\_token / refresh\_token hashes, sets the session via supabase.auth.setSession(...), and cleans the URL to #/.
* 🧭 **Race condition fix** — introduced isLoading state in useAuth and delayed redirects in RequireAdmin until hydration completes. This resolved the post-login redirect issue on production.

**Current production status**

* App deployed at: <https://kulmetehan.github.io/turkish-diaspora-app/#/login>
* Login → redirect → /admin flow confirmed working.
* Console clean, no errors or warnings.
* Refreshing /admin with a valid session keeps the user logged in.
* Magic link and password recovery also work correctly.

**Result**

“Admin Area – Auth OK” now loads after successful login on production.

The authentication system is stable, secure, and matches all acceptance criteria for TDA-124.

**Status:** ✅ *Done / Closed*

**Next:** integrate admin dashboard content in upcoming story (TDA-125).