

ShieldSense AI

Architecture Diagrams

Screen Hierarchy | User Flow | Onboarding | MaxDiff | C4 Scoring
Data Flow | Component Tree | Rider Policy | Auth | API Map

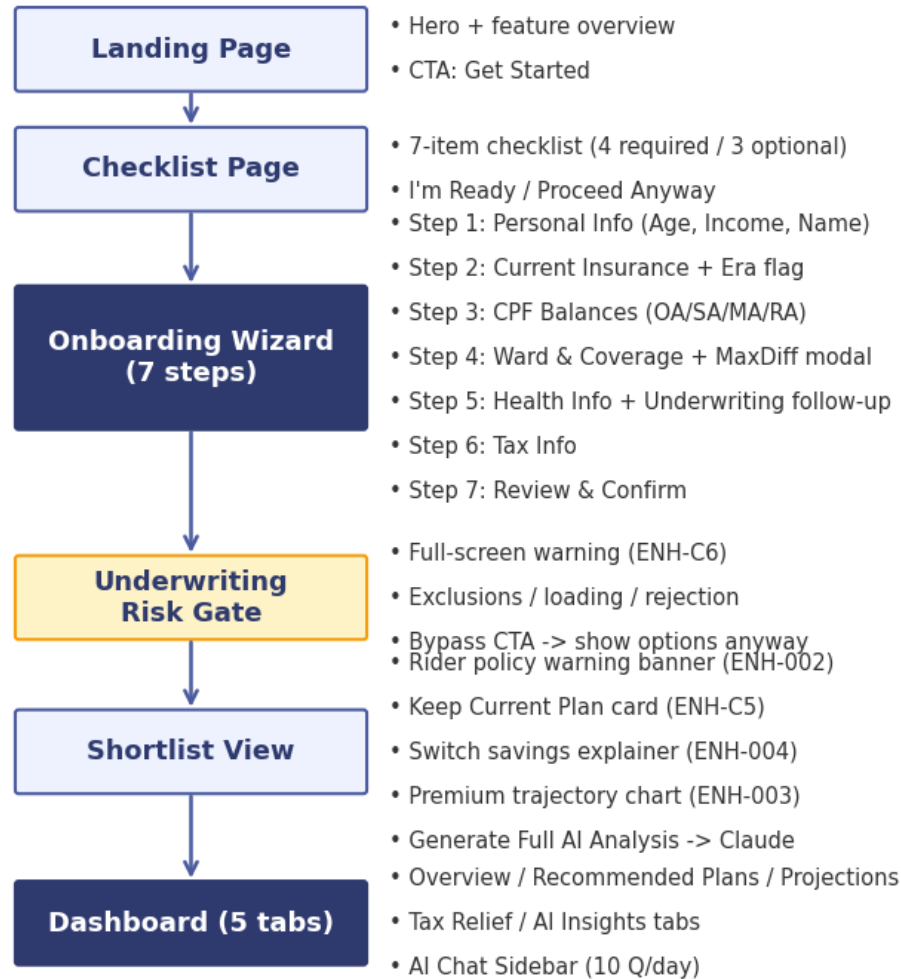
Current as of: 2026-04-26

ENH-C6 (underwriting gate) | ENH-C5 (keep-current-plan) | ENH-C4 (costSavings)
ENH-006/007 (era-aware onboarding) | ENH-009 (Singlife Health Plus full rider data)
pre-Apr 13 riders | post-Apr corrected | Raffles Shield Standard Plan (Post-Apr)
PRUExtra Plus / Lite grandfathered riders (Post-Apr)

1 | Information Architecture — Screen Hierarchy

Every screen and its key contents, arranged in user-journey order (top = first seen).

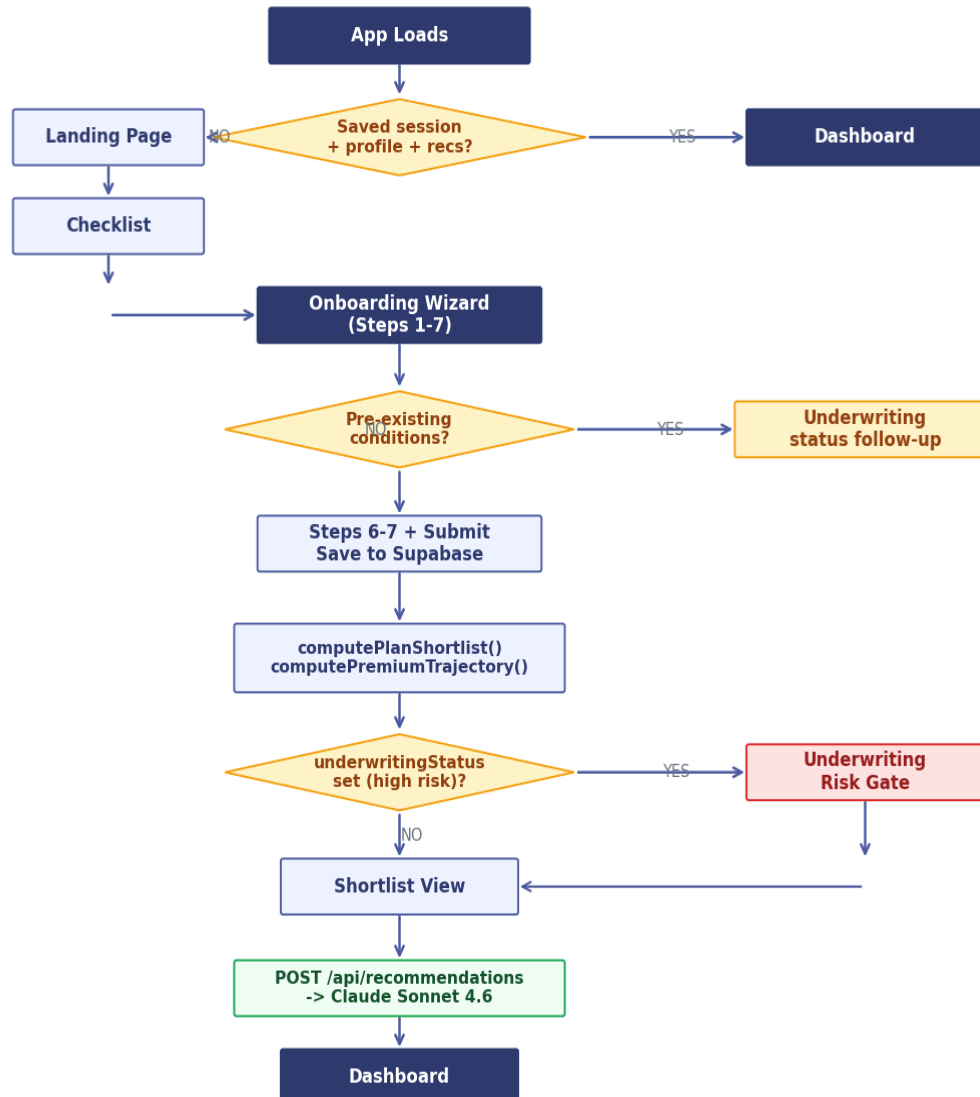
1 · Information Architecture — Screen Hierarchy



2 | User Flow — End-to-End Journey

Complete path from first visit to AI dashboard, including all branching decisions.

2 · User Flow — End-to-End Journey



3 | Onboarding Wizard — Step-by-Step Detail

Each step's inputs, output data object, and conditional logic. All steps auto-save to localStorage on every change.

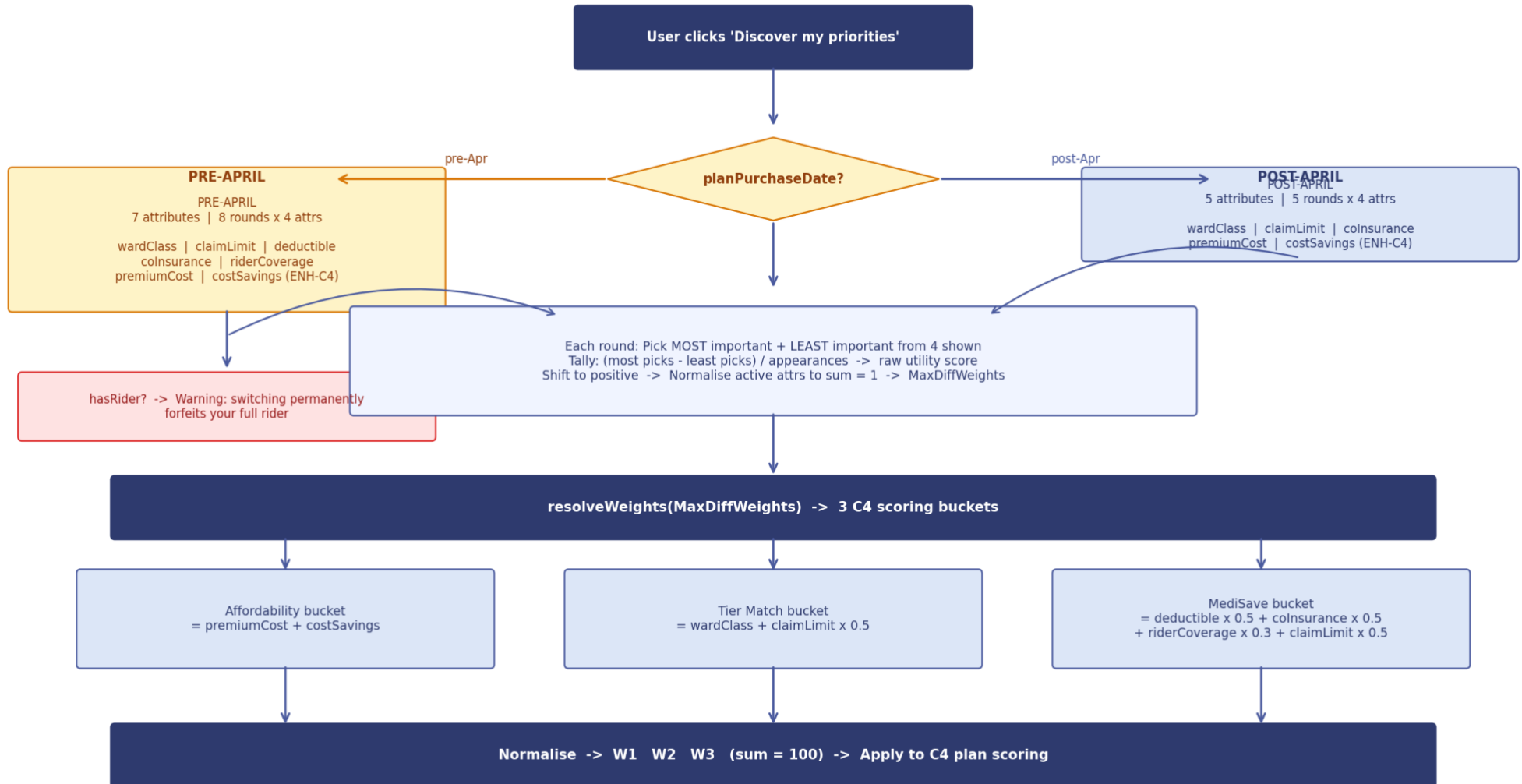
3 · Onboarding Wizard — Step-by-Step Detail

Step 1	Personal Info Age 18-90 Monthly income Name (optional) Output: PersonalInfo
Step 2	Current Insurance Insurer dropdown Era-filtered plan dropdown MediSave + Cash premiums Rider name + cash premium Plan purchase date -> era flag Output: CurrentInsurance + planPurchaseDate
Step 3	CPF Balances OA SA MA RA (optional) Live projection charts update as you type Output: CPFBalances
Step 4	Ward & Coverage Preferences Ward class Coverage level Risk tolerance Optional MaxDiff Survey (5 or 7 attrs depending on era) Full-rider warning banner if pre-April + hasRider Output: Preferences + MaxDiffWeights
Step 5	Health Info Smoker y/n Pre-existing conditions grid If conditions: underwriting status (4 options) (ENH-C6) Output: HealthInfo + conditionUnderwritingStatus
Step 6	Tax Info Life insurance premium Voluntary MediSave top-up Output: TaxInfo
Step 7	Review & Confirm Read-only summary of all sections Inline Edit buttons Submit -> save to Supabase -> run C4 shortlist Output: Complete UserProfile

4 | MaxDiff Survey — Era-Aware Logic & Weight Mapping

How the survey adapts to the user plan era, and how resulting weights flow into C4 scoring buckets.

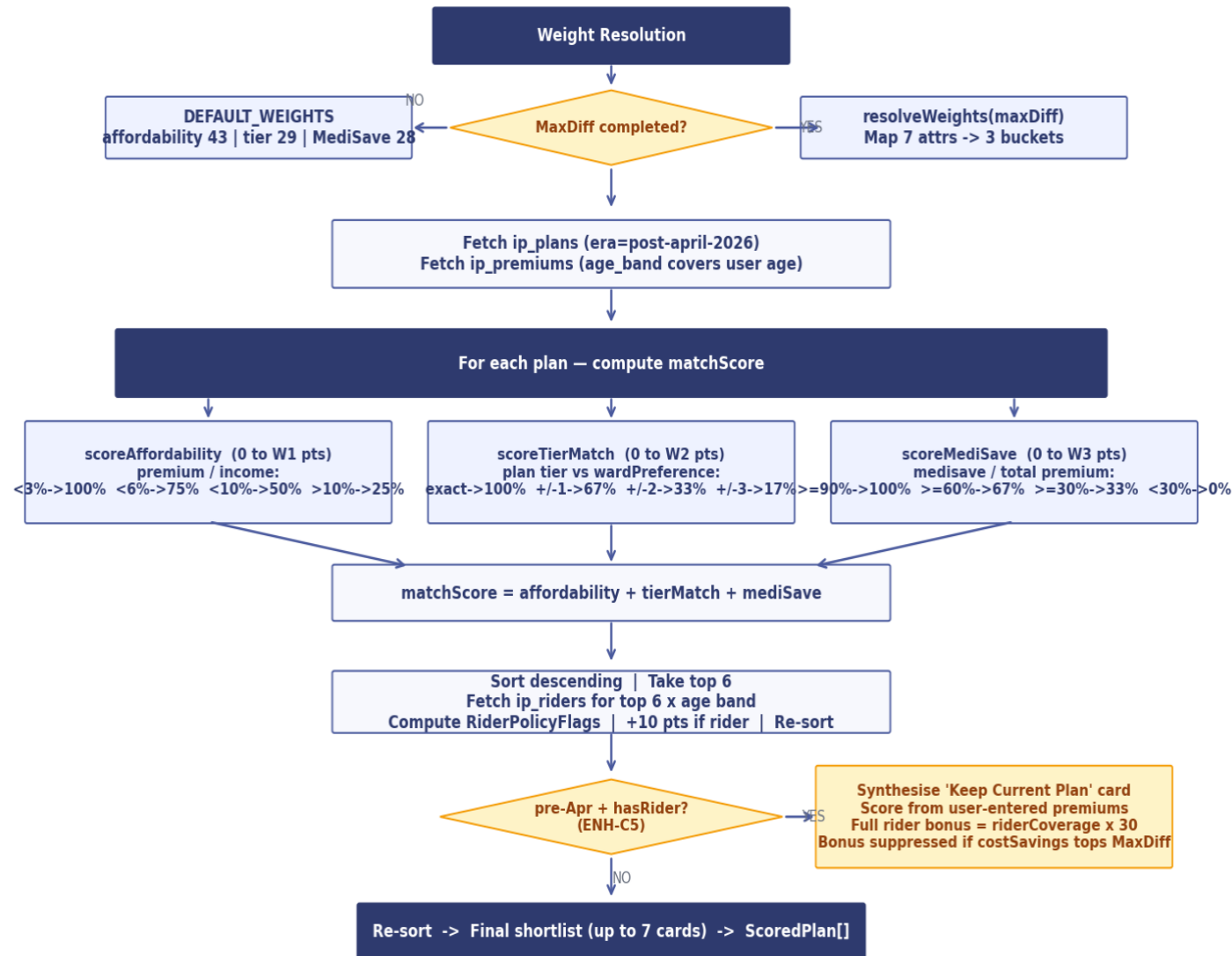
4 · MaxDiff Survey — Era-Aware Logic & Weight Mapping



5 | C4 Pre-computation Scoring Engine

Runs entirely in-browser against Supabase KB — no Claude involved. Produces a ranked shortlist in seconds.

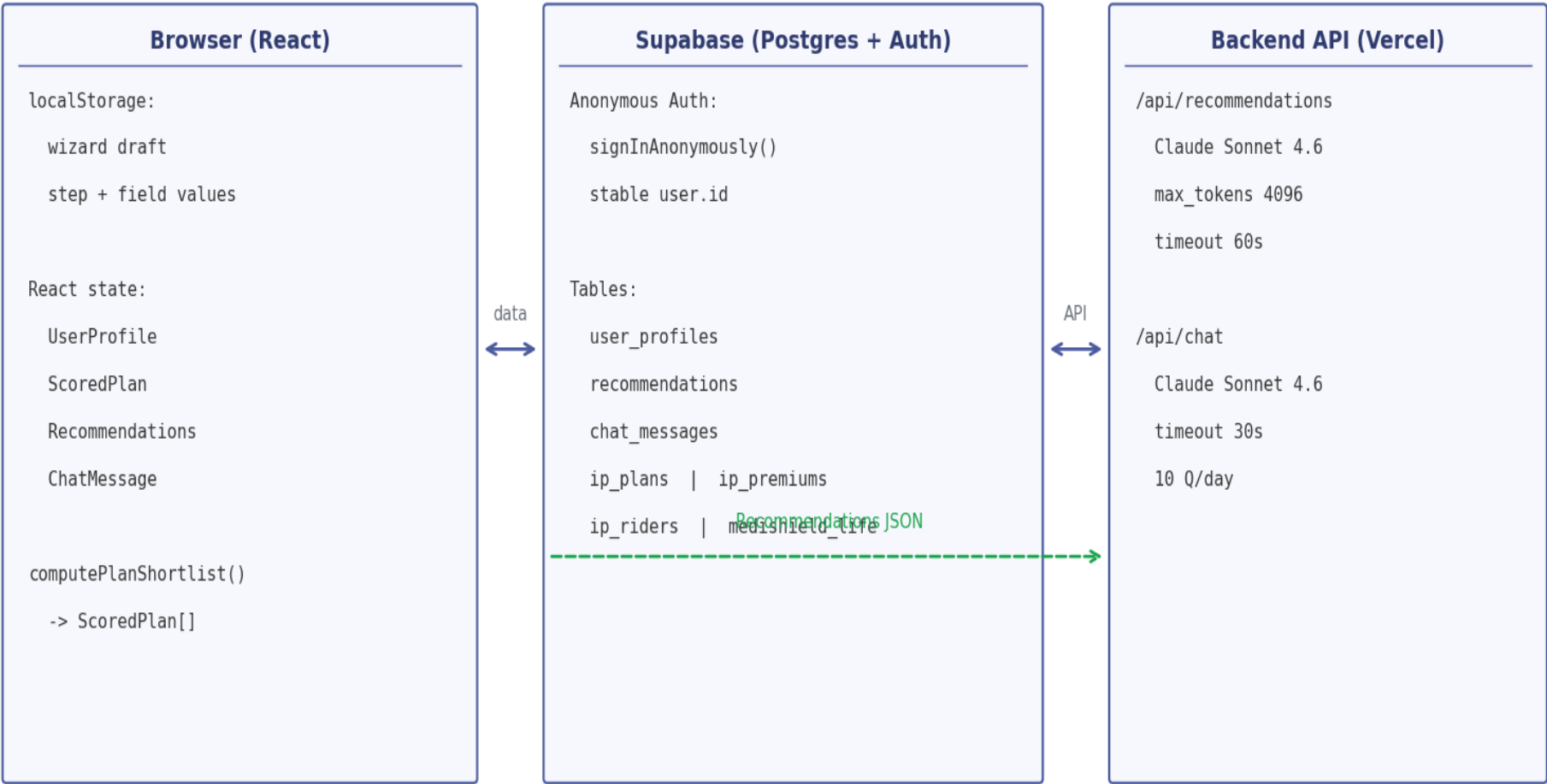
5 · C4 Pre-computation Scoring Engine



6 | Data Flow & Persistence

How data moves between the browser, Supabase, and the Claude API backend.

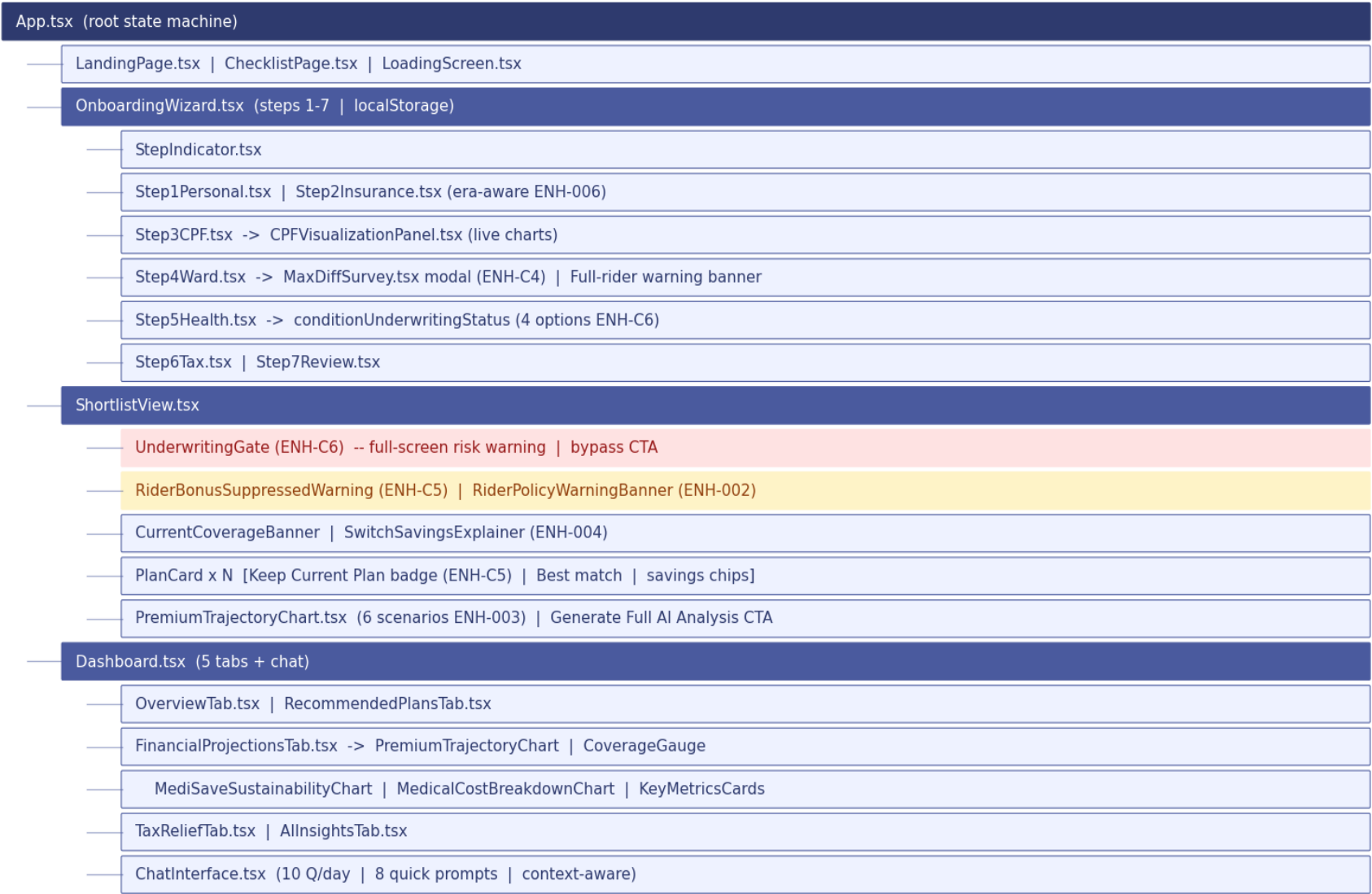
6 · Data Flow & Persistence



7 | Component Tree

React hierarchy from App.tsx downward. Generic UI primitives omitted. Arranged in user-journey order.

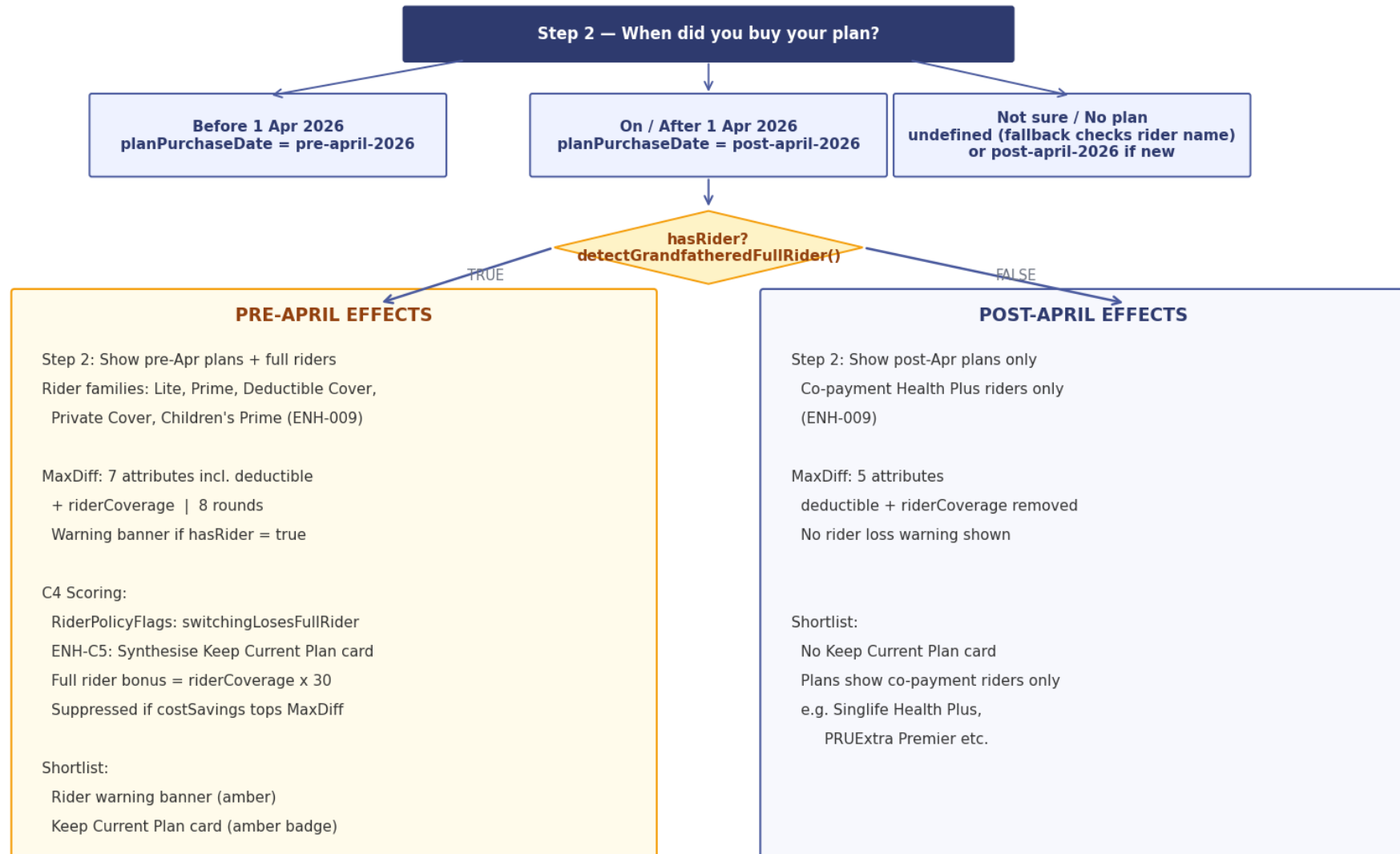
7 · Component Tree



8 | April 2026 Rider Policy Logic

How planPurchaseDate flows through the app — onboarding, MaxDiff, C4 scoring, and shortlist display.

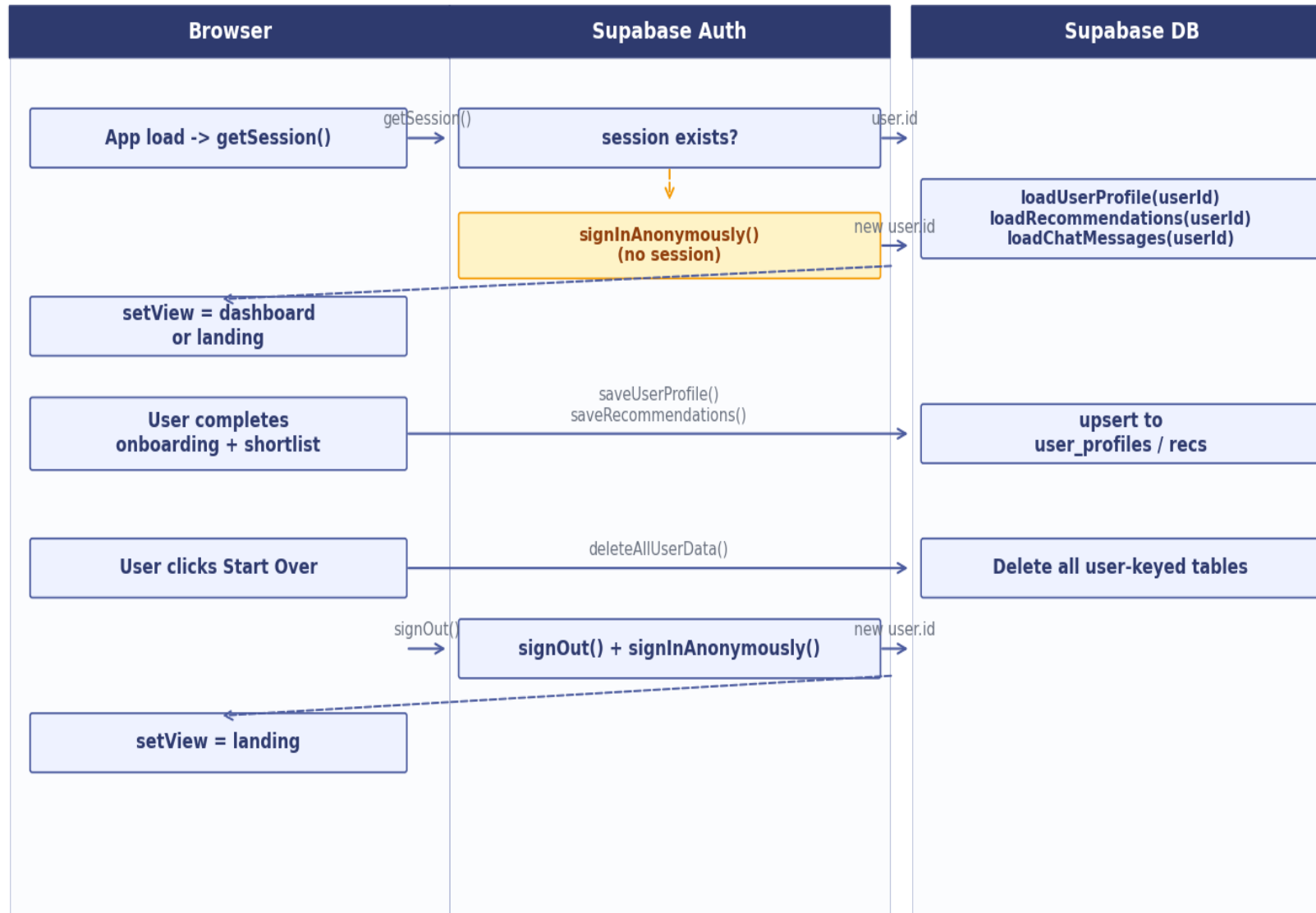
8 · April 2026 Rider Policy Logic



9 | Authentication & Session Lifecycle

Anonymous-first auth — no email or login required. Supabase provides a stable `user.id` that persists across reloads.

9 · Authentication & Session Lifecycle



10 | API & External Service Call Map

Every external call the app makes. Triggers flow downward to Supabase calls and Claude API calls.

10 · API & External Service Call Map

Trigger	Supabase Calls	Claude API
App mount	auth.getSession() or signInAnonymously()	--
Step 2 insurer selected	ip_plans.select() WHERE insurer = X AND era = Y	--
Step 2 plan selected	ip_riders.select() WHERE plan_id = X AND age band	--
Step 7 submit C4 shortlist	ip_plans.select() era = post-april-2026 ip_premiums.select() plan_id + age band ip_riders.select() top 6 x age band ip_premiums.select() all age bands (trajectory) medishield_life.select() all age bands user_profiles.upsert()	--
Shortlist Analyse button	recommendations.upsert()	POST /api/recommendations Claude Sonnet 4.6 Payload: UserProfile + ScoredPlan[] max_tokens 4096 timeout 60s Response: Recommendations JSON
Chat message sent	chat_messages.insert() x2 (user + assistant) chat_messages.select() today count	POST /api/chat Claude Sonnet 4.6 Payload: message + history + profile + recs timeout 30s 10 Q/day
Dashboard Start Over	deleteAllUserData() all user-keyed tables auth.signOut() then signInAnonymously()	--