# Privacy Advisor – PRD v3.0 (Finalized for MVP Build)

## 1. Overview

Privacy Advisor is a consumer-facing web app that lets users check the privacy and security posture of websites (MVP), and later apps and crypto wallets. It provides a clear score (0–100) with simple labels: Safe, Caution, High Risk.

## 2. Scope (MVP)

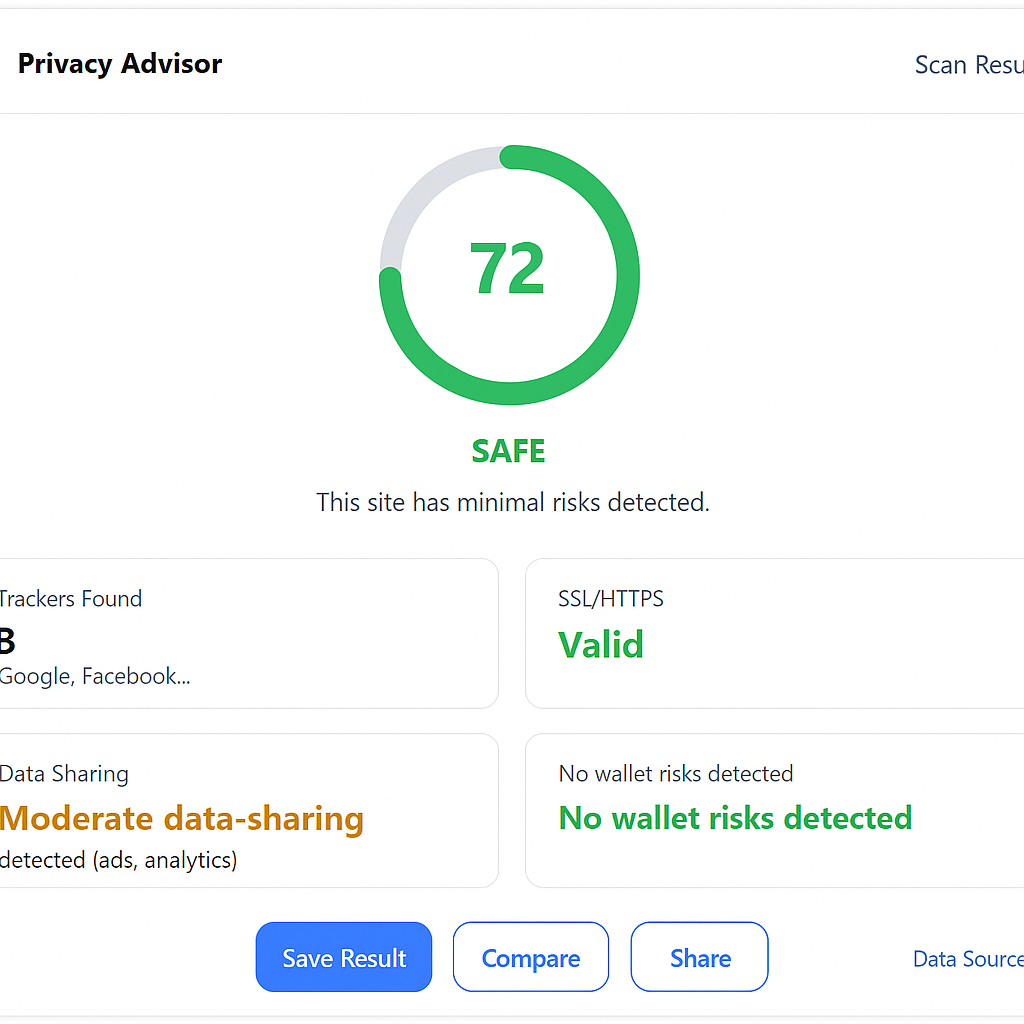
• Input: Website URL (apps + crypto postponed to later phases)  
• Output: Score + explanation  
• Risk Factors displayed:  
 1. Trackers Found (list key trackers)  
 2. SSL/HTTPS status  
 3. Data Sharing (Minimal / Moderate / High with context)  
 4. Wallet Risk (shown only if relevant)

## 3. Scoring Framework

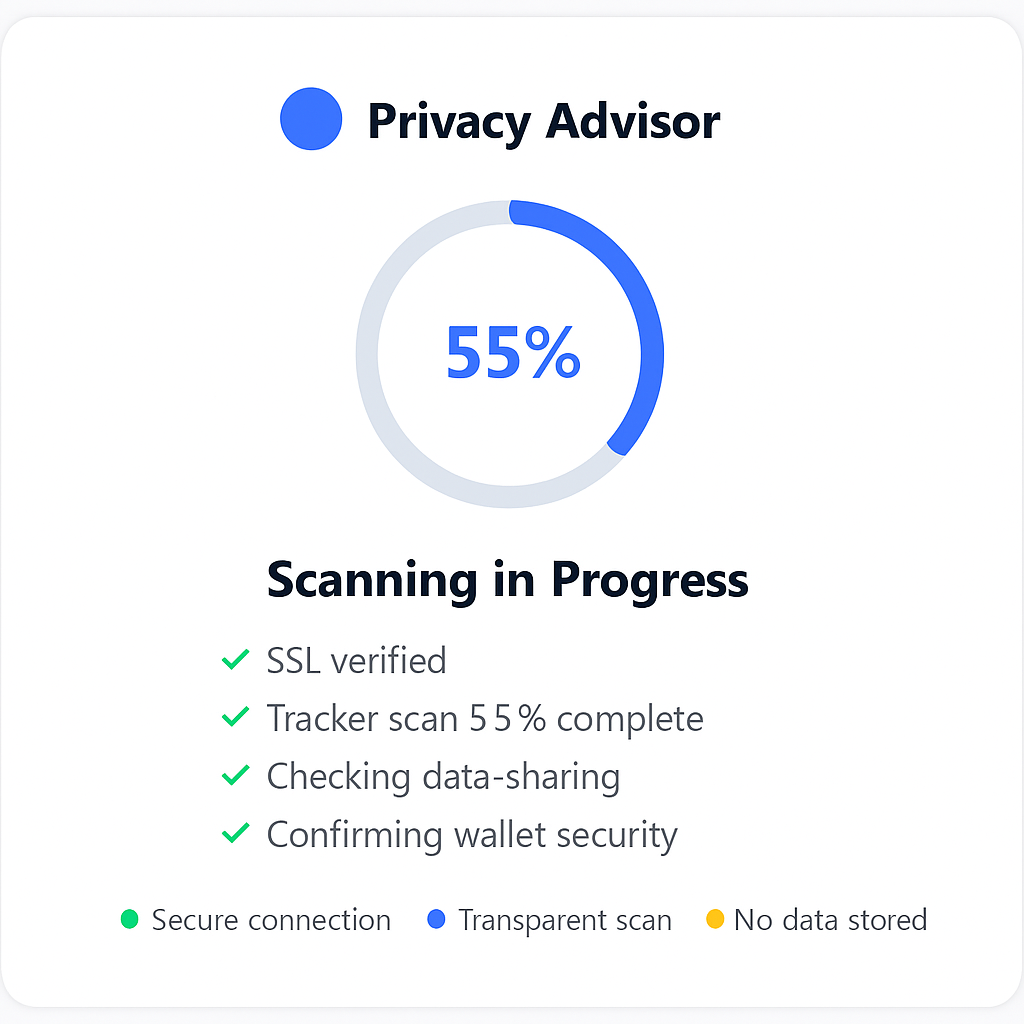
• Safe (Green, 70–100): Low trackers, valid SSL, minimal data-sharing  
• Caution (Yellow, 40–69): Multiple trackers, weak SSL, moderate data-sharing  
• High Risk (Red, 0–39): Expired/invalid SSL, many trackers, high data-sharing, scam flagged

## 4. UX (Finalized from Approved Mockups)

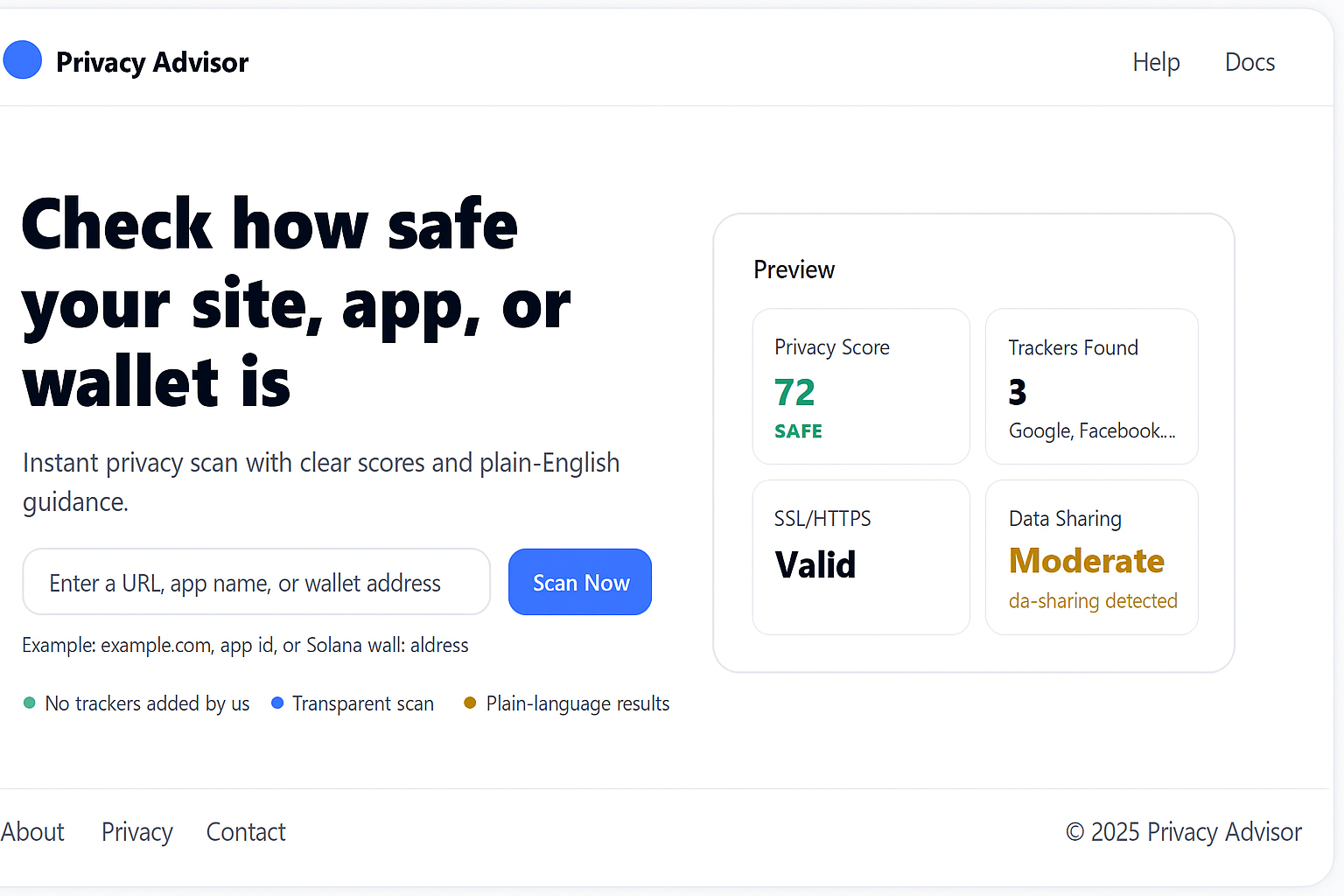
Screens refined and approved for MVP build:



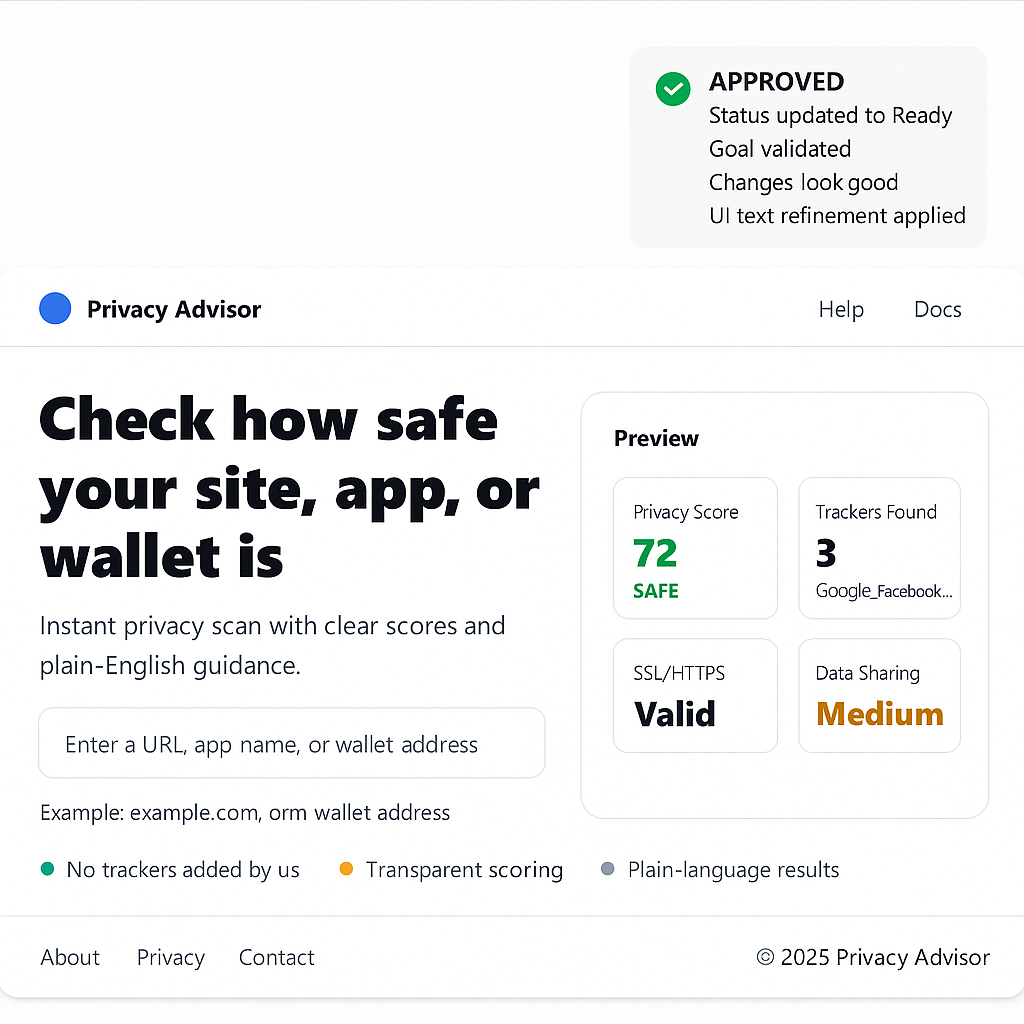
Scan Result Screen – Score ring with SAFE label, risk factors, and footer actions.



Scanning in Progress – Progress ring with animated checklist and trust legends.



Landing Page – Headline, input box, sample scan preview, footer with Data Sources.



Comparison Screen – Side-by-side site scores with reasons and icons.

## 5. Non-Functional Requirements

• Response Time: <3s scan result  
• Transparency: Data sources always shown  
• Privacy: No logs stored unless user opts-in  
• Accessibility: Colors + icons for clarity

## 6. Risks & Mitigation

• False positives/negatives → Validated scoring dataset  
• Legal issues → Disclaimer: 'Automated advisory only, based on open-source data.'  
• Credibility → Transparent scoring methodology page

## 7. Timeline

• Sprint 1 → Core Scoring Engine + UI skeleton  
• Sprint 2 → UX polish, trust layer, comparison feature  
• Beta Release → After Sprint 2

## 8. Phased Implementation Roadmap

### Phase 0 – Foundation (Week 0–1)

- Finalize PRD v3.0 ✅  
- Approve UX/UI ✅  
- Setup infra (Coolify + Hetzner baseline)  
- Define data ingestion pipelines (EasyPrivacy, DuckDuckGo Tracker Radar, WhoTracks.me)

### Phase 1 – MVP (Websites Only) [~6 Weeks]

Goal: Deliver fast, simple web scanner for website URLs only.  
- Input: URL  
- Output: Score + Risk Factors (Trackers, SSL, Data Sharing)  
- UX:  
 • Landing page + scan flow  
 • Progress with checklist  
 • Result screen (score + reasons)  
 • Comparison view  
- Deliverable: Public Beta (Web)

### Phase 1.5 – Crypto Extension (Websites + dApps/Wallets) [+4 Weeks]

Goal: Add crypto/Web3 trust scanning.  
- dApp/wallet contract verification via Solscan/Etherscan  
- Rugpull DB integrations (TokenSniffer, RugDoc)  
- Scoring extended with Web3-specific risks (liquidity lock, honeypot flags)  
- UX: show Wallet Risk section in results  
- Deliverable: Beta with Web + Crypto

### Phase 2 – Mobile Apps [~6–8 Weeks]

Goal: Expand beyond websites → Android/iOS apps.  
- Integrate with App Store + Play Store APIs  
- Parse app permissions → score vs baseline  
- Show privacy policies (if available)  
- Deliverable: Multi-platform scanning (Web + Apps + Crypto)

### Phase 3 – Browser Extension [~8 Weeks]

Goal: Real-time, inline scanning.  
- Extension auto-checks site privacy risk  
- Pop-up UI: score + quick explanation  
- Pro Tier: alerts when site classification changes  
- Deliverable: Privacy Advisor Extension (Chrome, Firefox, Brave, Edge)

### Phase 4 – Pro Tier & Monetization [~6 Weeks]

Goal: Convert into SaaS revenue.  
- Free tier: 3 scans/day  
- Pro tier: Unlimited scans, history, alerts, deeper insights  
- Payment: $3–5/mo or token-gated (align with $PRICKO ecosystem)  
- Deliverable: Monetized SaaS product

### Phase 5 – Full Ecosystem Integration [~3 Months+]

Goal: Integrate Privacy Advisor with Pricko ecosystem.  
- Integrate with Pricko Radar (in-house tracker DB)  
- Connect with Pricko Guard extension and Pricko Browser long-term  
- Potential API for third-party developers (B2B opportunities)