OpenSmartFill Pro — Complete Technical and User Documentation

1. Problem Statement

The default Safari autofill handles only basic, hardcoded field types (name, address, credit card). It fails for academic or institutional forms like 'ASU ID', 'Employee Number', or 'Project Code'. Users repeatedly retype this data across multiple sites.

2. Solution Overview

OpenSmartFill Pro is an intelligent, on-device, privacy-first autofill extension that semantically understands form fields, adapts to user context, learns from corrections, and securely syncs encrypted data via Firebase. It uses Transformers.js to compute embeddings locally and AES-GCM for end-to-end encryption.

3. User Perspective

Users install the Safari extension and add their personal fields (e.g., ASU ID, Work Email) into categorized profiles (Academic, Work, Travel). When they open a form, the extension highlights recognized fields with confidence chips. Users can confirm or auto-fill all fields with ■■F. Over time, the extension learns from accepted and corrected suggestions to improve accuracy. All data can be backed up or restored using a passphrase for encryption.

4. Developer Architecture Overview

The system consists of a Safari Web Extension built with Manifest v3, a local ML engine using Transformers.js, and Firebase for encrypted cloud sync. The architecture is fully offline-capable with optional cloud backup.

Frontend Components

- Content Script: scans DOM, extracts labels, computes embeddings, shows chips, and fills inputs. - Background Worker: handles hotkeys, sync, model preload, and caching. - Options UI (React or plain HTML): manages profiles, vault data, backup, and preferences. - Local ML: quantized MiniLM or E5-small model via Transformers.js (WASM backend). - IndexedDB: Dexie-based local storage for user fields and embeddings. - Encryption: Web Crypto API (PBKDF2 → AES-GCM 256).

Backend Components (Firebase, Free Tier)

- Firebase Auth: anonymous authentication. - Firestore: encrypted vault (user data), non-PII telemetry, and public synonym packs. - Cloud Functions: optional stats aggregation, synonym pack publishing, and validation. - Firestore Rules: restrict data per user and ensure only encrypted writes.

5. Processing Pipeline (End-to-End)

1. **Extract labels:** Identify text from , aria-label, or placeholder attributes. 2. **Embed:** Convert normalized label text to a vector using the local MiniLM model. 3. **Match:** Compute cosine similarity with saved field embeddings. 4. **Boost:** Apply domain-specific and rule-based confidence boosts. 5. **Decision:** - ≥ 0.85 → Auto-fill. - 0.72–0.85 → Suggest (chip confirmation). - < 0.72 → Ignore. 6. **Learn:** Update alias vectors and field centroids based on user actions. 7. **Sync:** Encrypt vault and sync to Firebase when connected.

6. Key Performance Indicators (KPIs)

• Top-1 Field Recognition Accuracy: ≥ 90% • p95 Fill-All Latency: ≤ 120 ms (30 fields) • Model Cold Start: ≤ 500 ms • Encrypted Sync Success Rate: ≥ 99% • Correction Rate: ≤ 8% • Auto-Fill Acceptance Rate: ≥ 85% • Confidence Precision@0.85: ≥ 0.9 • User Keystroke Reduction: ≥ 70% • Offline Operation Coverage: 100% • Local Storage Footprint: ≤ 50 MB (model + Dexie DB)

7. Security Model

- End-to-End encryption (AES-GCM, PBKDF2 key derivation). - Passphrase never leaves device; loss = irreversible vault lock. - No plaintext PII stored or transmitted. - Cloud Functions access only hashed/aggregated telemetry. - Safe Mode: blocks autofill on HTTP or unverified domains.

8. Differentiators vs Native Autofill

- Understands non-standard institutional fields. - Semantic similarity, not static mappings. - Transparent confidence indicators and explanations. - On-device learning from user corrections. - Offline-first with optional encrypted sync. - Multiple profiles for contextual autofill.

9. User Benefits and Trade-offs

Advantages: Faster, smarter autofill for complex forms, fully private, context-sensitive, and adaptive. **Disadvantages:** Initial setup required, slightly higher memory footprint, lost passphrase means unrecoverable vault, occasional SPA edge cases.

10. Summary

OpenSmartFill Pro is a complete semantic autofill ecosystem. It outperforms built-in Safari autofill by combining local embeddings, contextual confidence scoring, encrypted cloud sync, and user learning. It provides measurable improvements in accuracy, latency, and convenience while ensuring privacy through a zero-knowledge architecture.