# La Tortuga Digital Intake App — Software Design Specification (SDS)

Version: v1.0 Date: October 2025

Prepared By: Jose Rodriguez (Project Lead / Developer)

## 1. System Overview

The La Tortuga app is a bilingual (English/Spanish) offline-first medical and dental intake system for rural missions in Chiapas, MX. It operates on Android tablets, syncs data to Firebase Realtime Database, and visualizes anonymized statistics in a dashboard for both clinicians and community leaders.

## 2. Architecture Diagram

[Tablet App - React Native] → Sync Manager → Firebase Auth → Firebase Realtime Database → Cloud Function → Firebase Storage → BigQuery Export → Dashboard Web App

## 3. Technology Stack

Front-End: React Native + Expo  
Local Storage: SQLite / AsyncStorage  
Backend: Firebase Realtime Database  
Auth: Firebase Auth  
Analytics: BigQuery  
Dashboard: React + Chart.js  
CI/CD: GitHub Actions

## 4. Data Model Overview

Entities: Patients, Visits, MedicalIntake, DentalIntake, Vitals, Medications, Procedures, Communities, EducationSessions, Users.  
Relationships: One Patient → many Visits; One Visit → one MedicalIntake and/or DentalIntake; Visits linked to UserID and CommunityID.

## 5. Example JSON Schema

{  
 "Visits": {  
 "visit\_id": "UUID",  
 "patient\_id": "UUID",  
 "community": "SanMiguel",  
 "date": "2025-02-26",  
 "medical": {"neuro": ["Headache", "Anxiety"], "pain\_level": 5},  
 "dental": {"painful\_teeth": [3,14], "procedures": {"cleanings": 3}},  
 "created\_by": "user123",  
 "synced": true  
 }  
}

## 6. Module Design

Authentication: Firebase Auth with PIN login.  
Sync Manager: Offline detection, queued uploads, conflict resolution.  
MedicalIntake.js: Checkbox + 'Other' text fields.  
DentalIntake.js: Numeric grid (1–32) + procedure counters.  
Vitals.js: Pain slider, BP, HR, Temp.  
Dashboard: React + Chart.js with Firebase Admin SDK.

## 7. UI/UX Design

Bilingual toggle top-bar switch.  
Tab navigation per section.  
Large buttons for gloved input.  
Autosave every 30s.  
Stylus and touch supported.

## 8. Security & Privacy

Encryption: AES-256 at rest, TLS 1.3 in transit.  
Local storage encrypted.  
Role-based Firebase rules.  
PHI removed from external reports.  
Backup retention 90 days.

## 9. API Contracts

POST /Visits/{id} → Upload new record  
GET /Visits?community=XYZ → Dashboard data  
PATCH /Visits/{id}/synced → Update sync flag  
GET /Reference/Medications → Fetch list

## 10. Testing Plan

Unit: Jest + React Testing Library  
Integration: Expo + Firebase Emulator  
UAT: On-site pilot  
Performance: Android Profiler  
Security: Firebase Rules Testing

## 11. Deployment & CI/CD

Branches: main → staging → production  
Pipeline: GitHub Actions + Expo EAS  
Staging: la-tortuga-dev  
Production: la-tortuga-prod  
Release: Monthly or per mission.

## 12. Risk Management (FMEA)

Data loss during sync → Mitigation: retry queue.  
Device theft → Mitigation: PIN + short token TTL.  
Offline limits → Full local caching.  
Language issues → Manual toggle.

## 13. Acceptance Criteria

All forms save offline and sync correctly.  
Dashboard data accurate (±1%).  
Bilingual UI verified.  
>80% testing coverage.

## 14. Maintenance Plan

Quarterly DB rule review.  
Annual tablet updates.  
GitHub issue tracking.  
Monthly analytics check.