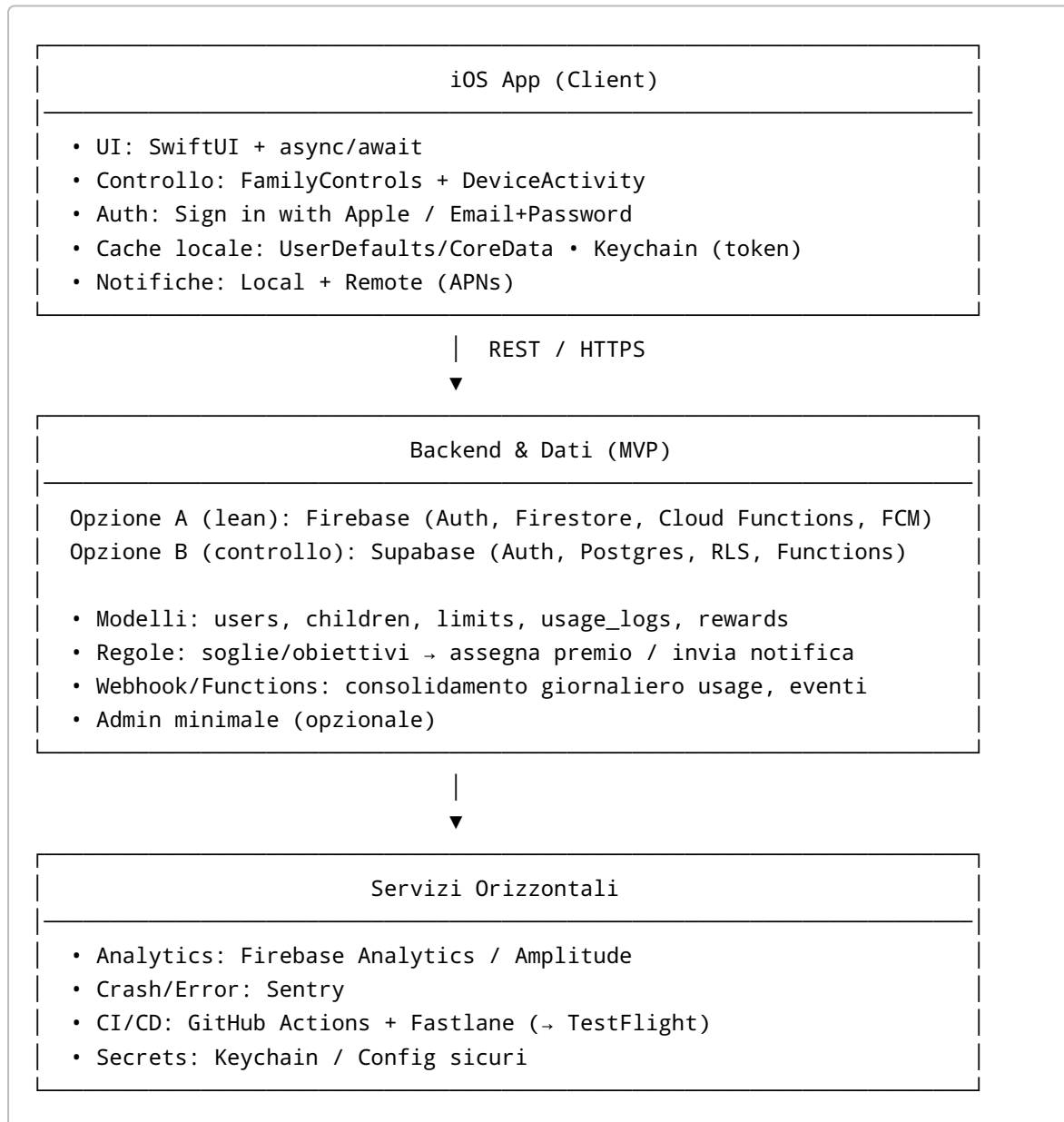


TREATS — Architettura (MVP)

Slide 1 — Vista logica ad alto livello



Flussi chiave (MVP)

1. **Setup:** Genitore crea account → aggiunge figli → seleziona app → imposta limiti.
2. **Monitoraggio:** DeviceActivity emette eventi → aggiornamento `usage_logs`.
3. **Regole & premi:** rispetto obiettivo/settimana virtuosa → `rewards` → notifica.
4. **Notifiche:** figlio (tempo residuo) / genitore (limite raggiunto, premio sbloccato).

Schema dati minimo

- **users:** id, email/appleId, ruolo=parent
- **children:** id, user_id, nome, età
- **limits:** child_id, bundle_id, daily_minutes, weekly_minutes, active
- **usage_logs:** child_id, bundle_id, date, minutes_used
- **rewards:** id, user_id, titolo, descrizione, criterio, stato
- **events** (opz.): type, payload, ts

Tabella — Tech Stack (MVP)

Componente	Scelta consigliata	Perché (breve)	Alternative	Note
Mobile UI	SwiftUI	Rapidità, nativo, mantenibilità	UIKit	iOS-first coerente con profilo ingegnere
Parental APIs	FamilyControls + DeviceActivity	Screen Time integrato	(—)	Richiede corretti permessi/UX
Auth	Sign in with Apple	UX senza password, privacy	Email+Password	Mix possibile
Cache locale	UserDefaults/ CoreData	Semplicità MVP	SQLite	Solo dati leggeri
Token/ Secrets	Keychain	Sicurezza locale	(—)	—
Notifiche	APNs (via FCM opz.)	Reach affidabile	OneSignal	In MVP bastano APNs
Backend (lean)	Firebase (Auth, Firestore, CF)	Zero DevOps, veloce e cheap	Supabase	Lock-in accettabile per MVP
DB (controllo)	Supabase (Postgres + RLS)	SQL, policy robuste	Managed Postgres	Più setup
Serverless	Cloud Functions	Regole & cron facili	Vercel/Netlify Fn	—
Analytics	Firebase Analytics	Gratis, integrato	Amplitude	In seguito coesistenza
Crash/Error	Sentry	Debug efficace	Firebase Crashlytics	Sentry + Crashlytics ok
CI/CD	GitHub Actions + Fastlane	Pipeline semplice fino a TestFlight	Bitrise	—
QA dist.	TestFlight	Beta testing Apple	(—)	—

Rischi tecnici principali

- Permessi Screen Time e messaggi chiari a minori.
 - Accuratezza logging e consolidamento giornaliero.
 - Batteria/performance con attività in background.
 - Privacy (minimizzare dati; trasparenza e consenso).
-

Domande di allineamento (da chiudere in meeting)

1. Partiamo **Firestore** per velocità o **Supabase** per controllo? (MVP vs. medio termine)
 2. Solo **Sign in with Apple** o affianchiamo **email+password**?
 3. **Admin panel** nel MVP o differito a M+1?
 4. Quale **metrica Nord Star** per il primo trimestre? (es. % giorni entro limite)
 5. **Roadmap Android**: quando e con quale stack (nativo/KMM/RN)?
-

Bozza timeline tecnica (8 settimane)

- **W1-2**: Auth + modelli dati + UI setup (figli, limiti)
- **W3-4**: Integrazione FamilyControls/DeviceActivity + logging
- **W5**: Regole premi + notifiche essenziali
- **W6**: Dashboard genitore (riepilogo)
- **W7**: Analytics/Crash + polishing
- **W8**: TestFlight, test con 5-10 famiglie