

Roadmap zur Umsetzung des Affiliate-Features

Tap n' Track Entwicklungsteam

22. April 2025

Zielsetzung

Entwicklung eines vollumfänglichen Affiliate-Programms in der *Tap n' Track*-App. Coaches und Partner-Studios können Angebote listen, Nutzer klick- und konversions-basiert tracken, Umsätze automatisiert splitten und reporten.

1 Übersicht der Phasen

Phase 1: Analyse & Anforderungen

Phase 2: Architektur & Datenmodell

Phase 3: Backend-Implementierung

Phase 4: Frontend-UI/UX

Phase 5: Tracking-Logik

Phase 6: Reporting & Abrechnung

Phase 7: Rechtliche Integration

Phase 8: Testing & Qualitätssicherung

Phase 9: Dokumentation & Rollout

2 Detaillierte Aufgaben

Phase 1: Analyse & Anforderungen

- **Use Cases** erfassen:
 - Studios hinterlegen Angebote (z. B. Mitgliedschaft, Merchandise).
 - Nutzer sehen Katalog, klicken auf Links.
 - Klicks und Conversions (z. B. Kaufabschluss) tracken.
 - Monatliche Provisionsabrechnung.
- **Provisionsmodell** definieren:

- Standard-Split (z. B. 30 % App / 70 % Studio).
- Flexible Raten pro Offer.
- **Technologie-Stack** festlegen:
 - Firestore für Datenspeicherung.
 - Cloud Functions für Automatisierung.
 - Flutter-Packages: `url_launcher`, `firebase_analytics`.

Phase 2: Architektur & Datenmodell

- **Firestore Collections:**
 - `affiliateOffers`
 - * *offerId*: Dokument-ID
 - * `name`, `imageUrl`, `targetUrl`
 - * `commissionRate`, `studioId`, `description`
 - `affiliateClicks`
 - * `offerId`, `userId?`, `clickedAt`
 - `affiliateConversions`
 - * `offerId`, `studioId`, `amount`, `ownerShare`, `studioShare`, `timestamp`
- **API-Schichten** in `api_services.dart`:
 - `getAffiliateOffers()`
 - `trackClick(offerId, userId?)`
 - `reportConversion(offerId, amount)`
 - `getAffiliateStats(studioId, timeRange)`
- **Cloud Function Design:**
 - On-Write Trigger für `affiliateConversions` zur Berechnung von Shares.
 - Scheduled Function für monatliche Aggregation & Email-Report.

Phase 3: Backend-Implementierung

- 1) **Firestore Regeln** anpassen:
 - Studios dürfen nur eigene `affiliateOffers` anlegen.
 - Klick-/Conversion-Logs nur anonym schreiben.
- 2) **API-Service Funktionen** implementieren.
- 3) **Cloud Functions:**
 - `onConversionCreate`:

- Provision berechnen: $ownerShare = amount \times commissionRate$ $studioShare = amount - ownerShare$
- Dokument updaten.
- **monthlyReport:**
 - Aggregation: Klicks, Conversions, Umsätze pro Studio.
 - CSV generieren und per Email verteilen.

Phase 4: Frontend-UI/UX

- **AffiliatePage** erstellen:
 - Kachel-Grid mit `affiliateOffers`.
 - Studio-Badge und Provisionsrate anzeigen.
- **OfferDetailPage:**
 - Großes Bild, Beschreibung, CTA-Button Zum Angebot.
- **StudioDashboard** (optional):
 - Stats-Übersicht (Klicks, Conversions, Umsätze).
 - Filter nach Zeiträumen.

Phase 5: Tracking-Logik

1) OnTap „Zum Angebot“:

- `await api.trackClick(offerId, userId);`
- `launchUrl(Uri.parse("$targetUrl?ref=$appIdstudio=$studioId"));`

2) Conversion-Handling:

- Manuelle Bestätigung via Backend-Callback (z. B. Webhook).
- Automatisierte Integration mit Zahlungs-API für automatische Conversion-Meldung.

Phase 6: Reporting & Abrechnung

- **Monatliche CSV-Reports:**
 - Cloud Function erstellt CSV mit Spalten: `studioId`, `offerId`, `clicks`, `conversions`, `totalAmount`, `ownerShare`, `studioShare`.
 - Versand per Email & optional als Download im StudioDashboard.
- **Live-Dashboard:**
 - Aggregate Zahlen in Echtzeit (Firestore Query + Charts).
 - Export-Button für CSV.

Phase 7: Rechtliche Integration

- **Transparenz in AGB/Impressum:**
 - Affiliate-Hinweis, Provisionsverteilung.
- **Datenschutz:**
 - Opt-In Checkbox für Affiliate-Tracking.
 - Anonymisierung: Keine personenbezogenen Daten ohne Zustimmung.

Phase 8: Testing & Qualitätssicherung

- **Unit Tests:**
 - API-Stubs: Klick-/Conversion-Tracking.
 - Cloud Functions Logik (Provisionsermittlung, Report-Export).
- **Integrationstests:**
 - E2E: Klick aus App → Firestore → Redirect.
 - Report-Generierung Email-Versand.
- **UX-Tests** mit Pilot-Studios.

Phase 9: Dokumentation & Rollout

- **Entwickler-Docs:**
 - API-Endpunkte, Cloud Function Specs, DB-Schema.
- **User Guide:**
 - Kurzanleitung für Coaches & Studios.
 - Beispiel AGB-Text.
- **Deployment:**
 - Release-Plan, Feature-Flag für stufenweise Ausrollung.
 - Monitoring Alerting (z. B. Crashlytics, Logs).

Zeitplan (Beispiel)

Woche 1–2	Analyse, Datenmodell, Architektur
Woche 3–4	Backend-Implementierung (affiliateOffers, Klick/Conversion Tracking)
Woche 5	Frontend UI: Katalog & Detailseite
Woche 6	Tracking-Logik & URL-Handling
Woche 7	Reporting-Feature & Cloud Functions
Woche 8	Rechtliche Integration & AGB
Woche 9	Testing, Pilot-Rollout
Woche 10	Dokumentation, Vollständiger Live-Rollout