Can you suggest me how ecommerce companies like Urban Company or Flipkart retarget a user based on visit. Or say if someone has made an entry into google form or Instagram form with mobile no and then after 6 months came to app and registered using OTP in App. Please remember 6 months back he/she did not verify OTP, it was a pure form submission. Can all entries be store in single database?

**1. Retargeting Based on Visit:**

E-commerce platforms use a combination of **cookies, device fingerprinting, pixels**, and **backend identifiers** to track user visits and retarget them later via:

* **Display Ads (Google Display Network, Facebook, etc.)**
* **Push Notifications**
* **SMS / WhatsApp (if mobile number is known)**
* **Email Campaigns**

**2. Someone filled a form (e.g., Google Form / Insta Lead Ad) with mobile number (6 months ago), but didn't verify via OTP back then — and now installs app and registers via OTP.**

This scenario is a classic case of **anonymous-to-known user mapping**. Here's how it can work:

**🧠 Smart Retargeting Logic:**

You **can** unify all these records (form submission, app registration) **into a single database**, but you need to design it to **handle identity resolution** across sessions/channels.

**📍 Key Concepts:**

1. **Unified Customer Profile (UCP):**
   * Maintain a centralized UserIdentity table.
   * Store multiple identity types: mobile number, email, device ID, ad click ID (e.g., gclid, fbclid), etc.
   * Tag each event or entry (form fill, visit, app install) with a reference to this table — or with a temporary ID that can later be linked.
2. **Deferred Matching (Soft Linking):**
   * When someone fills a Google Form or Insta Lead Ad, store that entry in a Lead table with mobile number, source, timestamp, and a status = unverified.
   * When they later sign up on the app (OTP verified), **match the verified mobile number** against historical lead entries.

This enables a **retrospective merge** of user journeys.

1. **Retargeting Engines:**
   * Tools like **MoEngage, WebEngage, Clevertap, Segment, Amplitude**, etc., allow you to track and stitch user journeys even across months — especially when you unify based on **mobile number**.

Database Design(Simplified Pseudo Logic):

TABLE: Leads

- id (UUID)

- mobile\_number (VARCHAR)

- source (ENUM: 'google\_form', 'instagram', 'website', 'app')

- entry\_timestamp (DATETIME)

- verified (BOOLEAN)

TABLE: Users

- user\_id (UUID)

- mobile\_number (VARCHAR)

- signup\_source (ENUM: 'app', 'web')

- signup\_timestamp (DATETIME)

TABLE: UserEvents

- event\_id (UUID)

- user\_id (nullable, for verified users)

- temp\_id (used before OTP verified)

- event\_type (ENUM: 'visit', 'click', 'form\_submit', 'install', etc.)

- device\_id / gclid / fbclid

- event\_timestamp

**Practical Workflow:**

1. User fills Google Form (mobile: 99800XXXXX) → Store in Leads.
2. Later installs app and signs up → OTP verified.
3. Mobile number matches → Merge into Users.
4. Retargeting engine sees full journey → Can now send personalized nudges: *"Hey! Still thinking about getting your car serviced at home?"*

**✅ Yes, You Can Store All Entries in One Database — If:**

* You **normalize** identity resolution.
* You **log anonymous events** with temp IDs.
* You have a **backend that can do retrospective user stitching**.

How to communicate via InApp or Push Notification or WhatsApp using Amplitude, twillo, clevertap, webengage, moengage to those users who have entered mobile no?

**Yes — but with conditions.**  
Just because you have a user's mobile number from a form doesn't mean you can **freely message them via Push, In-App, or WhatsApp** unless certain conditions are met. Let's break it down by channel and platform:

**1. Push Notifications**

✅ You can **only** send push notifications if:

* The **user has installed your app**
* They've **opted in** for notifications (Android: auto; iOS: needs user permission)
* You have **their device token / push token**

🛑 So even if you have their mobile number from a form — you **can't push** unless they've installed your app *and* given permission.

**Platform Support:**

* ✅ Clevertap, MoEngage, WebEngage — full support
* ❌ Amplitude — *does not send* push (analytics only)
* ❌ Twilio — not for push (messaging & OTP infra)

**🧩 2. In-App Notifications**

✅ You can only send if:

* User has opened the app and has an active session
* You have a way to identify them (via mobile, device ID, or internal ID)

So again, a form entry alone ≠ in-app communication.

**Platform Support:**

* ✅ Clevertap, MoEngage, WebEngage — yes
* ❌ Amplitude — no direct messaging
* ❌ Twilio — not relevant

**🧩 3. WhatsApp Messaging**

✅ You **can send WhatsApp messages** via Twilio, Gupshup, etc. *IF*:

* You have user’s **valid consent** (opt-in) as per WhatsApp Business Policy.
* You use **pre-approved message templates** (e.g., lead nurturing, appointment reminders)
* You’re sending messages within the **24-hour customer service window** or using **template-based notifications** outside it.

🛑 You can’t use WhatsApp like a free marketing channel unless you're compliant.

**Platform Support:**

* ✅ Twilio — excellent WhatsApp integration
* 🟡 Clevertap / MoEngage — can integrate with WhatsApp providers (via Gupshup/Twilio)
* 🟡 WebEngage — same, uses partner integrations
* ❌ Amplitude — not messaging focused

**🧩 4. SMS**

✅ If you’ve collected mobile numbers via a form, you **can send SMS**, but ideally:

* You disclose during form submission that you may contact them via SMS.
* You honor DND and unsubscribe norms.

**Platform Support:**

* ✅ Twilio — very good
* ✅ Clevertap / MoEngage / WebEngage — can integrate SMS providers

**🚦Summary Table:**

| **Channel** | **Clevertap** | **MoEngage** | **WebEngage** | **Amplitude** | **Twilio** |
| --- | --- | --- | --- | --- | --- |
| Push | ✅ | ✅ | ✅ | ❌ | ❌ |
| In-App | ✅ | ✅ | ✅ | ❌ | ❌ |
| WhatsApp | 🟡 (via API) | 🟡 | 🟡 | ❌ | ✅ |
| SMS | ✅ (via API) | ✅ | ✅ | ❌ | ✅ |
| Analytics Only | ✅ | ✅ | ✅ | ✅ | ❌ |

🟡 = requires integration with third-party tools like Twilio/Gupshup/MessageBird