# 🧩 Module 7: Customer Upload + Mapping Engine

## 🎯 Overview

This module allows each brand to securely upload their existing customer base (name, phone, email, address, previous order data) and automatically match these users when they: - Comment on a Facebook/Instagram campaign - Message in inbox/DMs - Participate in UGC - Place orders or engage on the brand’s website/app

The goal is to centralize identity across all channels to: - Enable personalized replies - Send follow-up Email/SMS campaigns - Auto-fill user profiles in leaderboards - Improve reporting & attribution

## 🗂️ Core Features

### 1. CSV Upload Interface

* Brands can upload a CSV file with fields: name, email, phone, address, order history, user ID (optional).
* Real-time column mapping UI with preview
* Validation for email/phone format
* Upload limit settings (configurable per plan)

### 2. Identity Normalization & Indexing

* Phone/email is normalized (e.g., removing +880, 0-prefix)
* Addresses tokenized to allow approximate match
* Hash map & inverted index to speed up lookup during future mapping

### 3. Cross-Channel Mapping

* When new UGC, comment, or inbox data comes in:
  + Match commenter’s name with known name + fuzzy rules
  + Match phone/email (if exposed via Meta API) directly
  + Infer connection based on past inbox chats, shared address, city, zone

### 4. Matching Confidence Scoring

* Score is generated based on exact match, fuzzy name match, behavioral signal overlap (like area, product interest)
* Admins can define minimum score threshold to auto-map
* UI to approve/reject edge cases

### 5. Profile Enrichment

* Matched user data is linked to existing or new system profile
* Used to:
  + Auto-fill leaderboard profiles
  + Enable reward dispatch
  + Add to email/SMS flows
  + Run user-specific reporting

### 6. GDPR + Permissions

* Consent field available per user (“allow marketing communication”)
* Deletion API for GDPR right to erasure
* Uploaded data is siloed per brand with encryption at rest

## 🔧 How to Build It

### Stack

* Backend: Node.js + PostgreSQL + Redis
* Frontend: Next.js + Tailwind
* File Parsing: PapaParse for frontend; csv-parser or fast-csv in backend
* Identity Match: Fuse.js or custom fuzzy engine for names; phone-lib for phone normalization

### DB Tables

* customer\_uploads: upload metadata per brand
* customer\_profiles: master profile with UID, email, phone, last\_seen
* customer\_mappings: tracks cross-platform connections
* consents: GDPR, opt-in, communication permission

### API Endpoints

* POST /upload-customers → Upload and validate
* GET /uploads/:brand\_id → Show previous uploads
* POST /match-customer → Attempt to resolve identity
* POST /delete-customer/:uid → GDPR compliance

## ⚙️ Superadmin Controls

* Enable/disable module for brand
* Set upload quota
* Set mapping confidence threshold
* Logs of all customer matches (audit trail)
* Force full re-matching across brands

## 🧠 AI Assist Add-ons (Optional)

* AI Identity Matcher: GPT powered identity suggestion with reason
* Address Normalizer: AI-based city/locality correction from partial/incomplete address
* GPT-powered UGC reader: Suggest matching customer from UGC or inbox message (“Hi, I ordered a pink serum last month”)

## 🔗 Connected Modules

* Email/SMS Campaign Builder → audience segments, abandoned cart flows
* UGC/Gamification → map comments to user
* Inbox/Messenger → recognize repeat users and personalize
* Analytics → accurate cohort reporting, per-customer lifetime value

## 📈 Usage Benefits

* Higher personalization & engagement
* Unified customer identity across sales channels
* Enhanced marketing attribution
* Lower manual effort for mapping DMs to users

✅ This module will function as a standalone add-on and also empower nearly every other module when enabled.