**CafeNet Backend Documentation** **System Architecture: Phone Number-Based Check-in + Referral + Cafe Validation**

## 📊 OVERVIEW

This backend powers the CafeNet platform, a local cafe discovery and loyalty system that uses phone numbers as a core identifier. It supports:

* User registration and referral chain system
* Cafe account and staff dashboard
* Manual reward entry by phone number
* Secure point allocation and leaderboard tracking
* Obfuscated secure phone number storage for admin-only access

## 👤 USER MODULE (Updated & Extended)

### 🔍 Core Features

* Phone number-based registration & login password and otp
* Admin-only secure suffix stored with phone number
* Editable user profile with profile image
* Referral system with automatic point allocation (150 points on first visit by referred user)
* Points system based on spending (₹500 = 50 points default logic)
* Complete visit and transaction history (earned & redeemed)

### 📚 API Endpoints

| Method | Endpoint | Description |
| --- | --- | --- |
| POST | /api/users/register | Register user with phone number, password, referral code |
| POST | /api/users/login | Login with phone + password or OTP |
| GET | /api/users/profile/:phone | Get profile data by phone |
| PUT | /api/users/profile/:phone | Edit name or profile image |
| GET | /api/users/referral-chain/:phone | Retrieve full referral tree |
| GET | /api/users/history/:phone | Fetch visit history |
| GET | /api/users/rewards/:phone | Fetch earned and redeemed point history |

### 📂 Model: User

{  
 name: String,  
 phone: { type: String, unique: true }, // visible to cafe and user  
 securePhoneId: { type: String, unique: true }, // phone + 3-digit suffix, admin-only  
 password: String, // hashed  
 profilePic: String,  
 points: { type: Number, default: 0 },  
 referredBy: String, // referral code of parent  
 referralCode: String, // auto-generated code  
 referralChildren: [String], // phone numbers referred by this user  
 createdAt: Date,  
 visitLogs: [ObjectId], // VisitLog refs  
 rewardLogs: [ObjectId] // RewardTransaction refs  
}

### 🔁 Referral Logic

* During registration, if a valid referralCode is provided:
  + It maps to an existing user.
  + That user is stored in referredBy.
  + Once the new user completes their **first cafe order**, the referrer gets **150 points** via RewardTransaction.

### 📖 Visit History

* Available at /api/users/history/:phone
* Retrieved from VisitLog collection
* Includes:
  + Cafe name
  + Amount spent
  + Points earned
  + Timestamp

### 💰 Reward History (Earned & Redeemed)

* Available at /api/users/rewards/:phone
* Retrieved from RewardTransaction

{  
 userPhone: String,  
 type: "earn" | "redeem",  
 points: Number,  
 description: String,  
 source: String, // Cafe or system  
 timestamp: Date  
}

### 🧰 Middleware

* validatePhoneNumber: Checks phone number format
* authenticateUserJWT: Protect routes requiring user auth
* checkReferralEligibility: Ensures referrer gets points only once

### 🧭 Routes (Folder Structure)

/routes  
 users.js // user routes  
/controllers  
 userController.js  
/models  
 User.js  
 RewardTransaction.js  
 VisitLog.js  
/middleware  
 auth.js  
 validate.js

### 🔐 Admin-only Logic

* Every phone number stored with a hidden suffix:
  + e.g., 9876543210 → 9876543210-491 in securePhoneId
  + Not exposed to cafes or end users
* Used by:
  + /api/admin/users
  + Internal logs & fraud audit

### ✅ Sample Reward Logic (Earned)

When cafe logs ₹500 bill:

* System gives **50 points** based on business rule (₹10 = 1 point)
* Logs it to RewardTransaction

If referred user is making **first visit**:

* System checks checkReferralEligibility
* Gives **150 points** to referrer
* Logs separate transaction for that

### ✅ Sample Reward Logic (Redeem)

When user redeems 300 points:

* System checks if points >= 300
* Deducts 300 from user.points
* Logs redemption in RewardTransaction
* Sends SMS confirmation (optional)

Let me know if you’d like this exported as a .docx or .pdf file!