StoreFlow — A1.2 Data Model & SQL Specification (Part 3)

15. Customer Domain – Overview

The Customer Domain handles identification, communication, loyalty tracking, and relationship management for individuals placing orders on storefronts.

Customers may interact with StoreFlow in two ways:

• Guest Checkout — minimal data collected

• Registered Account — optional, provides access to order history and loyalty

Requirements:

• Customer identity must always be scoped by merchant_id.

• A customer may appear in multiple stores but always under the same merchant.

• The system ensures simple UX flows for registration or guest checkout.

• Loyalty accounts are tied to customers at the merchant level (not store level).

16. Customer Table – Full Specification

TABLE: customers

---

id BIGINT PK

merchant_id BIGINT FK → merchants.id

first_name VARCHAR(255)

last_name VARCHAR(255)

email VARCHAR(255)

mobile VARCHAR(50)

password_hash VARCHAR(255) NULL (guest accounts have NULL)

created_at TIMESTAMP

updated_at TIMESTAMP

Important Behaviors:

• Email is NOT globally unique—only unique within merchant_id.

• A customer without password_hash is treated as a guest.

• When a guest later registers, the existing record is upgraded.

Indexes:

• idx_customers_merchant (merchant_id)

• idx_customers_email (merchant_id, email)

• idx_customers_mobile (merchant_id, mobile)

17. Guest Checkout Model

Guest checkout is the fastest path to placing an order.

Guest Flow:

1. Customer enters first name, last name, email, mobile.

2. StoreFlow checks if a customer exists for that email/mobile.

3. If found → reuse existing record.

4. If not found → create new customer with password_hash = NULL.

Advantages:

• Zero friction for customers.

• Loyal customers can reuse record without logging in.

• Upgrading to full account is seamless.

Security Note:

• Guest accounts cannot log in.

• Access to order tracking is provided via a secure public_id token per order.

18. Registered Account Model

Registered accounts allow customers to:

• View past orders

• Earn & redeem loyalty points

• Update account details

• Faster checkout experience

Registration workflow:

1. Guest customer is identified by email/mobile.

2. Customer chooses to create password.

3. password_hash is set.

4. Account becomes active.

Password Rules:

• bcrypt hashing

• Min length 8–10

• Optional MFA future support

Login Rules:

• Email/mobile + password

• merchant_id scoping required

19. Loyalty Accounts – Full Specification

Loyalty accounts belong to a customer at a merchant level.

TABLE: loyalty_accounts

---

id BIGINT PK

merchant_id BIGINT FK

customer_id BIGINT FK → customers.id

points_balance INT DEFAULT 0

created_at, updated_at

Notes:

• One loyalty account per merchant per customer.

• Points accrue from paid orders only.

• Redemptions subtract from points_balance.

• Loyalty events logged in audit system.

Indexes:

• UNIQUE (merchant_id, customer_id)

20. Customer-to-Order Relationship

Customers do not need accounts to place orders.

Orders table stores customer_id:

• Guest or registered customers always have a record.

• Orders remain static after creation.

Relationship:

CUSTOMER 1---N ORDERS

Important:

Order contact details (email, mobile, name) must also be stored on the order itself.

Reason:

• Customer may update or delete their account.

• Order must remain immutable for auditing and tax records.

Thus, the order table stores:

• shipping_name

• shipping address fields

• contact email

• contact mobile

21. Customer Identity Rules

Identity Resolution Rules:

• Email + mobile combined provide strong identification.

• merchant_id is always part of the identity context.

• No two customers under the same merchant may share both email AND mobile.

Account De-duplication:

During checkout:

• If email exists → match

• Else if mobile exists → match

• Else create new guest customer

Upgrading guests:

• Guest → registered simply sets password_hash.

Merging customers:

• Manual merge (future admin dashboard item).

22. GDPR & Privacy Considerations

StoreFlow must follow privacy best practices.

Guidelines:

• Right-to-access: Provide customer data export (future).

• Right-to-delete: Allowed only if no completed orders exist.

• Account deletion → anonymize PII, retain order records.

• Encryption at rest for passwords.

• No plaintext email/mobile in logs.

• Audit logs must store redacted metadata only.

These rules ensure compliance in international markets.

23. Customer Domain – Example SQL Queries

Query 1 — Locate customer by email:

```
SELECT * FROM customers
WHERE merchant_id = ?
AND email = ? LIMIT 1;
```

Query 2 — Create guest customer:

```
INSERT INTO customers (merchant_id, first_name, last_name, email, mobile)
VALUES (?, ?, ?, ?, ?);
```

Query 3 — Attach loyalty:

```
SELECT * FROM loyalty_accounts
WHERE merchant_id = ?
AND customer_id = ?;
```

Query 4 — Upgrade guest to registered:

```
UPDATE customers
SET password_hash = ?
WHERE id = ?;
```

Query 5 — List all customers for a store:

```
SELECT DISTINCT c.*
FROM customers c
JOIN orders o ON o.customer_id = c.id
WHERE o.store_id = ?;
```