

1. TABLE DESIGN

USERS

Field	Type
user_id	INT AUTO_INCREMENT PRIMARY KEY
name	VARCHAR(100)
email	VARCHAR(100) UNIQUE
phone	VARCHAR(20)
password_hash	VARCHAR(255)
role	ENUM('Customer','Owner','Delivery','Admin')
address	TEXT
rating	DECIMAL(3,2)
created_at	DATETIME

USER_VERIFICATION

Field	Type
verification_id	INT AUTO_INCREMENT PRIMARY KEY
user_id	INT (FK)
id_type	VARCHAR(50)
id_number	VARCHAR(50)
status	ENUM('Pending','Verified','Rejected')
verified_at	DATETIME

CATEGORIES

Field	Type
category_id	INT AUTO_INCREMENT PRIMARY KEY
category_name	VARCHAR(100)
description	TEXT

ITEMS

Field	Type
item_id	INT AUTO_INCREMENT PRIMARY KEY
owner_id	INT (FK → USERS.user_id)
category_id	INT (FK)
item_name	VARCHAR(150)
condition	ENUM('New','Like New','Used')
description	TEXT
item_count	INT
price_per_day	DECIMAL(10,2)
availability_status	ENUM('Available','Rented','Unavailable')
created_at	DATETIME

INVENTORY

Field	Type
inventory_id	INT AUTO_INCREMENT PRIMARY KEY
item_id	INT (FK → ITEMS.item_id)
total_quantity	INT
available_quantity	INT

Field	Type
rented_quantity	INT
last_updated	DATETIME

ITEM_IMAGES

Field	Type
image_id	INT AUTO_INCREMENT PRIMARY KEY
item_id	INT (FK)
Category_id	INT(FK)
image_url	TEXT

RENTAL_BOOKINGS

Field	Type
rental_id	INT AUTO_INCREMENT PRIMARY KEY
item_id	INT (FK → ITEMS.item_id)
customer_id	INT (FK → USERS.user_id)
quantity	INT
start_date	DATE
end_date	DATE
rental_days	INT
price_per_day	DECIMAL(10,2)
total_rent	DECIMAL(10,2)
rental_status	ENUM('Pending','Confirmed','Cancelled','Completed')
created_at	DATETIME

PURCHASE_ORDERS

Field	Type
purchase_id	INT AUTO_INCREMENT PRIMARY KEY
item_id	INT (FK → ITEMS.item_id)
customer_id	INT (FK → USERS.user_id)
quantity	INT
price_per_unit	DECIMAL(10,2)
total_price	DECIMAL(10,2)
order_status	ENUM('Pending','Confirmed','Cancelled','Delivered')
purchased_at	DATETIME

PAYMENTS

Field	Type
transaction_id	INT AUTO_INCREMENT PRIMARY KEY
reference_type	ENUM('Rental','Purchase')
reference_id	INT
customer_id	INT (FK → USERS.user_id)
amount	DECIMAL(10,2)
payment_method	ENUM('UPI','Card','NetBanking','Wallet','Cash')
payment_status	ENUM('Pending','Success','Failed','Refunded')
paid_at	DATETIME

DELIVERY

Field	Type
delivery_id	INT AUTO_INCREMENT PRIMARY KEY
booking_id	INT (FK → BOOKINGS.booking_id)
partner_id	INT (FK → DELIVERY_PARTNERS.partner_id)
transaction_id	INT(FK→PAYMENTS.transaction_id)
delivery_type	ENUM('Pickup','Return')
delivery_status	ENUM('Assigned','In Transit','Completed','Failed')
pickup_address	TEXT
drop_address	TEXT
scheduled_date	DATE
scheduled_time	TIME
delivered_at	DATETIME
created_at	DATETIME

DELIVERY_PARTNERS

Field	Type
partner_id	INT AUTO_INCREMENT PRIMARY KEY
user_id	INT (FK → USERS.user_id)
phone	VARCHAR(20)
address	TEXT
vehicle_type	VARCHAR(50)
availability	ENUM('Available','Busy')
created_at	DATETIME

SUPPORT_TICKETS

Field	Type
ticket_id	INT AUTO_INCREMENT PRIMARY KEY
booking_id	INT (FK)
raised_by	INT (FK → USERS.user_id)
issue_type	VARCHAR(100)
description	TEXT
status	ENUM('Open','In Review','Resolved')
created_at	DATETIME

REVIEWS

Field	Type
review_id	INT AUTO_INCREMENT PRIMARY KEY
item_id	INT (FK)
user_id	INT (FK → USERS.user_id)
rating	INT
comment	TEXT
created_at	DATETIME

2. TABLE NORMALIZATION (1NF → 2NF → 3NF)

UNNORMALIZED FORM (UNF)

Initially, all information related to users, items, bookings (rental/purchase), payments, delivery, reviews, and support issues were stored in a single table.

Problems:

- Data redundancy
- Update and deletion anomalies
- Multiple values stored in single attributes

FIRST NORMAL FORM (1NF)

- Atomic values ensured
- Repeating groups removed
- Separate tables created for:

USERS , ITEMS , BOOKINGS / RENTAL_BOOKINGS /
PURCHASE_ORDERS , PAYMENTS , DELIVERY

SECOND NORMAL FORM (2NF)

- Partial dependencies removed
- All tables use single-column primary keys
- Non-key attributes fully depend on their respective primary keys:
 - User details depend only on user_id
 - Item details depend only on item_id
 - Booking details depend only on booking_id / rental_id / purchase_id
 - Payment details depend only on transaction_id

THIRD NORMAL FORM (3NF)

- Transitive dependencies removed
- Categories separated into CATEGORIES
- Delivery details separated into DELIVERY and DELIVERY_PARTNERS
- Review and support handling separated into REVIEWS and SUPPORT_TICKETS