Food Delivery System

Schema Documentation

Data Dictionary • Index Specifications • User Role Definitions

1. Data Dictionary

All tables, columns, data types and constraints for the Food Delivery System database.

Table	Column	Data Type	Constraints	Description
Person	person_id	INT	PK, AUTO_INCREMENT	Unique identifier for all users
	name	VARCHAR(100)	NOT NULL	Full name of the person
	email	VARCHAR(100)	UNIQUE, NOT NULL	Email used for login
	phone	VARCHAR(15)	NOT NULL, CHECK (LENGTH(phone) >= 10)	Contact phone number
	password_hash	VARCHAR(255)	NOT NULL	Bcrypt-hashed password
Customer	customer_id	INT	PK, AUTO_INCREMENT	Unique identifier for customers
	person_id	INT	FK (Person.person_id), NOT NULL, UNIQUE	Links to parent Person record (1:1)
Admin	admin_id	INT	PK, AUTO_INCREMENT	Unique identifier for admins
	person_id	INT	FK (Person.person_id), NOT NULL, UNIQUE	Links to parent Person record (1:1)
	role	ENUM('SuperAdmin','Manager','Support')	NOT NULL, DEFAULT 'Support'	Admin permission level
Rider	rider_id	INT	PK, AUTO_INCREMENT	Unique identifier for riders

Table	Column	Data Type	Constraints	Description
	person_id	INT	FK (Person.person_id), NOT NULL, UNIQUE	Links to parent Person record (1:1)
	vehicle_type	ENUM('Bike','Car','Scooter')	NOT NULL	Vehicle type for delivery
	license_plate	VARCHAR(20)	NOT NULL, UNIQUE	Vehicle registration plate
	is_available	TINYINT(1)	NOT NULL, DEFAULT 1	1 = available, 0 = busy/offline
Address	address_id	INT	PK, AUTO_INCREMENT	Unique identifier for addresses
	customer_id	INT	FK (Customer.customer_id), NOT NULL	References the owning customer
	street	VARCHAR(255)	NOT NULL	Street name and number
	city	VARCHAR(100)	NOT NULL	City name
	postal_code	VARCHAR(10)	NOT NULL	Postal / ZIP code
	country	VARCHAR(50)	NOT NULL, DEFAULT 'Pakistan'	Country (defaults to Pakistan)
	is_default	TINYINT(1)	NOT NULL, DEFAULT 0	1 = customer default address
Restaurant	restaurant_id	INT	PK, AUTO_INCREMENT	Unique identifier for restaurants
	name	VARCHAR(100)	NOT NULL	Restaurant display name
	location	VARCHAR(255)	NOT NULL	Physical address / location
	phone	VARCHAR(15)	NOT NULL	Restaurant contact phone

Table	Column	Data Type	Constraints	Description
	email	VARCHAR(100)	UNIQUE, NOT NULL	Restaurant email
	rating	DECIMAL(3,2)	CHECK (rating BETWEEN 0 AND 5)	Average customer rating
	is_active	TINYINT(1)	NOT NULL, DEFAULT 1	1 = visible to customers
Category	category_id	INT	PK, AUTO_INCREMENT	Unique identifier for categories
	restaurant_id	INT	FK (Restaurant.restaurant_id), NOT NULL	References owning restaurant
	name	VARCHAR(50)	NOT NULL	Category name (e.g. Appetizers)
MenuItem	item_id	INT	PK, AUTO_INCREMENT	Unique identifier for menu items
	category_id	INT	FK (Category.category_id), NOT NULL	References parent category
	name	VARCHAR(100)	NOT NULL	Menu item name
	description	TEXT	NULL	Optional item description
	price	DECIMAL(10,2)	NOT NULL, CHECK (price > 0)	Price in PKR
	is_available	TINYINT(1)	NOT NULL, DEFAULT 1	1 = currently orderable
Order	order_id	INT	PK, AUTO_INCREMENT	Unique identifier for orders
	customer_id	INT	FK (Customer.customer_id), NOT NULL	Customer who placed the order
	restaurant_id	INT	FK (Restaurant.restaurant_id), NOT NULL	Restaurant fulfilling the order
	address_id	INT	FK (Address.address_id), NOT NULL	Delivery address

Table	Column	Data Type	Constraints	Description
	rider_id	INT	FK (Rider.rider_id), NULL	Assigned rider (null until assigned)
	order_date	DATETIME	NOT NULL, DEFAULT CURRENT_TIMESTAMP	Date and time order was placed
	status	ENUM('Pending','Preparing','Out for Delivery','Delivered','Cancelled')	NOT NULL, DEFAULT 'Pending'	Order lifecycle status
	total_amount	DECIMAL(10,2)	NOT NULL, CHECK (total_amount >= 0)	Total cost in PKR
OrderItem	orderitem_id	INT	PK, AUTO_INCREMENT	Unique identifier for order line items
	order_id	INT	FK (Order.order_id), NOT NULL	References parent order
	item_id	INT	FK (MenuItem.item_id), NOT NULL	References the menu item ordered
	quantity	INT	NOT NULL, CHECK (quantity > 0)	Units ordered
	unit_price	DECIMAL(10,2)	NOT NULL	Price per unit at time of order
Payment	payment_id	INT	PK, AUTO_INCREMENT	Unique identifier for payments
	order_id	INT	FK (Order.order_id), NOT NULL, UNIQUE	References order (1:1)
	amount	DECIMAL(10,2)	NOT NULL, CHECK (amount > 0)	Payment amount in PKR
	payment_date	DATETIME	NOT NULL	Date and time of payment
	method	ENUM('Credit Card','Cash on Delivery','JazzCash','EasyPaisa')	NOT NULL	Payment method used
	status	ENUM('Paid','Failed','Refunded')	NOT NULL	Payment status

Table	Column	Data Type	Constraints	Description
Review	review_id	INT	PK, AUTO_INCREMENT	Unique identifier for reviews
	customer_id	INT	FK (Customer.customer_id), NOT NULL	Customer who wrote the review
	restaurant_id	INT	FK (Restaurant.restaurant_id), NOT NULL	Restaurant being reviewed
	rating	INT	NOT NULL, CHECK (rating BETWEEN 1 AND 5)	Star rating 1–5
	comment	TEXT	NULL	Optional written comment
	review_date	DATE	NOT NULL, DEFAULT CURDATE()	Date review was submitted

PK = Primary Key | FK = Foreign Key | UNIQUE = must be unique | CHECK = validation rule

2. Index Specifications

Indexes defined to improve query performance for common access patterns and join conditions.

Index Name	Table	Column(s)	Rationale
idx_person_email	Person	email	Enforce email uniqueness and speed up login lookups.
idx_customer_person	Customer	person_id	Speed up JOIN between Customer and Person on profile load.
idx_admin_person	Admin	person_id	Speed up JOIN between Admin and Person; enforce one admin per person.
idx_rider_person	Rider	person_id	Speed up JOIN between Rider and Person; enforce one rider per person.
idx_rider_plate	Rider	license_plate	Unique index prevents duplicate plates; supports vehicle lookup.
idx_address_customer	Address	customer_id	Fast retrieval of all addresses for a customer during checkout.
idx_category_rest	Category	restaurant_id	Instant category listing when a restaurant menu page is loaded.
idx_menuitem_cat	MenuItem	category_id	Fast item listing per category filtered by is_available.
idx_order_customer	Order	customer_id	Fast order history queries for a given customer.

Index Name	Table	Column(s)	Rationale
`idx_order_restaurant`	Order	`restaurant_id`	Restaurant dashboard queries for incoming and past orders.
`idx_order_rider`	Order	`rider_id`	Rider app fetches all orders assigned to them quickly.
`idx_order_status`	Order	`status`	Admin dashboards filtering orders by status.
`idx_orderitem_order`	OrderItem	`order_id`	Build order receipts — retrieve all line items for an order.
`idx_payment_order`	Payment	`order_id`	Unique index enforces one payment per order; supports status lookup.
`idx_review_rest`	Review	`restaurant_id`	Compute average ratings and list reviews per restaurant.
`idx_review_cust`	Review	`customer_id`	List all reviews submitted by a customer on their profile page.

3. User Role Definitions

Three user roles are defined. Access is enforced at the application layer via scoped queries.

Role	Permissions	Description
Admin	- Full CRUD on all tables - Manage restaurants (add/edit/delete) - Manage user accounts - View all orders and payments - Assign and reassign riders - Generate system reports	Platform staff with elevated access. Sub-roles: SuperAdmin (all permissions), Manager (no user deletion), Support (read-only + order management). All admin actions are timestamped via managed_at.
Customer	- Read: Restaurants, Menus, Categories - Full CRUD on own Addresses - Create and read own Orders - Create and read own Reviews - Read own Payment records	Registered end-users who place orders. Strictly scoped to their own data via customer_id. Cannot access other customers' orders, addresses, or payment details.
Rider	- Read: Assigned orders only - Update: Order status on own orders - Update: Own availability flag - Read: Delivery address for active order	Delivery personnel. Access is scoped to orders assigned to their rider_id. Cannot access customer profiles, payment data, or restaurant management.

4. Relationship Summary

All foreign key relationships and their cardinalities across the system.

Parent Table	Child Table	Cardinality	FK Column
Person	Customer	1:1	customer.person_id
Person	Admin	1:1	admin.person_id
Person	Rider	1:1	rider.person_id
Customer	Address	1:N	address.customer_id
Customer	Order	1:N	order.customer_id
Customer	Review	1:N	review.customer_id
Restaurant	Category	1:N	category.restaurant_id
Restaurant	Order	1:N	order.restaurant_id
Restaurant	Review	1:N	review.restaurant_id
Category	MenuItem	1:N	menuitem.category_id
Order	OrderItem	1:N	orderitem.order_id
Order	Payment	1:1	payment.order_id
MenuItem	OrderItem	1:N	orderitem.item_id
Rider	Order	1:N	order.rider_id
Address	Order	1:N	order.address_id
Admin	Restaurant	1:N	via MANAGES relationship