



***IRideMoto***  
**Database Design Document (DDD)**  
**Version 1.0**

Prepared by: Diaz, Ven Jabez  
*Aguila, Dreig Rashid*  
Pano, Ronen Kenneth  
Pascual, Tyrone Gabriel

## Revision History

Date	Version	Description	Author
28/02/2025	1.0		Pano, Ronen Kenneth S.

# Table of Contents

<b>1</b>	<b>Introduction .....</b>	<b>1</b>
1.1	Document Objectives .....	1
1.2	Intended Audiences .....	1
<b>2</b>	<b>Entity Relationship Diagram (ERD).....</b>	<b>2</b>
<b>3</b>	<b>Detailed Database Design .....</b>	<b>3</b>
3.1.1.1	Data dictionary for Element: Customer Table .....	3
3.1.1.2	Data dictionary for Element: Rider Table.....	3
3.1.1.3	Data dictionary for Element: Booking Table.....	4
3.1.1.4	Data dictionary for Element: Payment Table.....	5
3.1.1.5	Data dictionary for Element: Promo Table .....	5
3.2	MySQL database design (Relational database) .....	6
3.2.1	<i>Conceptual diagram</i> .....	6
3.2.2	<i>Description</i> .....	6
3.2.3	<i>Purpose of Tables</i> .....	7
3.2.3.1	Purpose of Customer .....	7
3.2.3.2	Purpose of Rider.....	7
3.2.3.3	Purpose of Booking.....	7
3.2.3.4	Purpose of Payment.....	7
3.2.3.5	Purpose of Promo .....	7
3.2.4	<i>Relations</i> .....	8
<b>4</b>	<b>References .....</b>	<b>8</b>

# 1 Introduction

In today's busy world, where every moment is precious and every urban commute poses numerous challenges, the IRideMoto Motorcycle Booking App has emerged as a reliable solution, providing a smart and practical approach to navigating the daily commute. IRideMoto connects users with certified motorcycle riders at the tap of a button, providing affordable, dependable, and rapid transportation. IRideMoto guarantees that all people, as well as goods, move efficiently through all congested streets. It doesn't matter if it's every commuter rushing to work, every traveler navigating a new city, or every business needing urgent package delivery.

This revolutionary app has a database system that is both forceful and quite well-structured. IRideMoto relies on this database. Passengers and riders can easily interact with the platform because of it. The database is absolutely key to a truly satisfying user experience since it capably manages user accounts, efficiently processes ride requests, accurately tracks real-time locations, and securely handles payments.

It is the plan for building and maintaining the database used by the IRideMoto app. It confirms the database fully meets all the app's requirements and consistently remains scalable, secure, and efficient through the complete outlining of the entire structure, full functionality, and all technical details.

## 1.1 Document Objectives

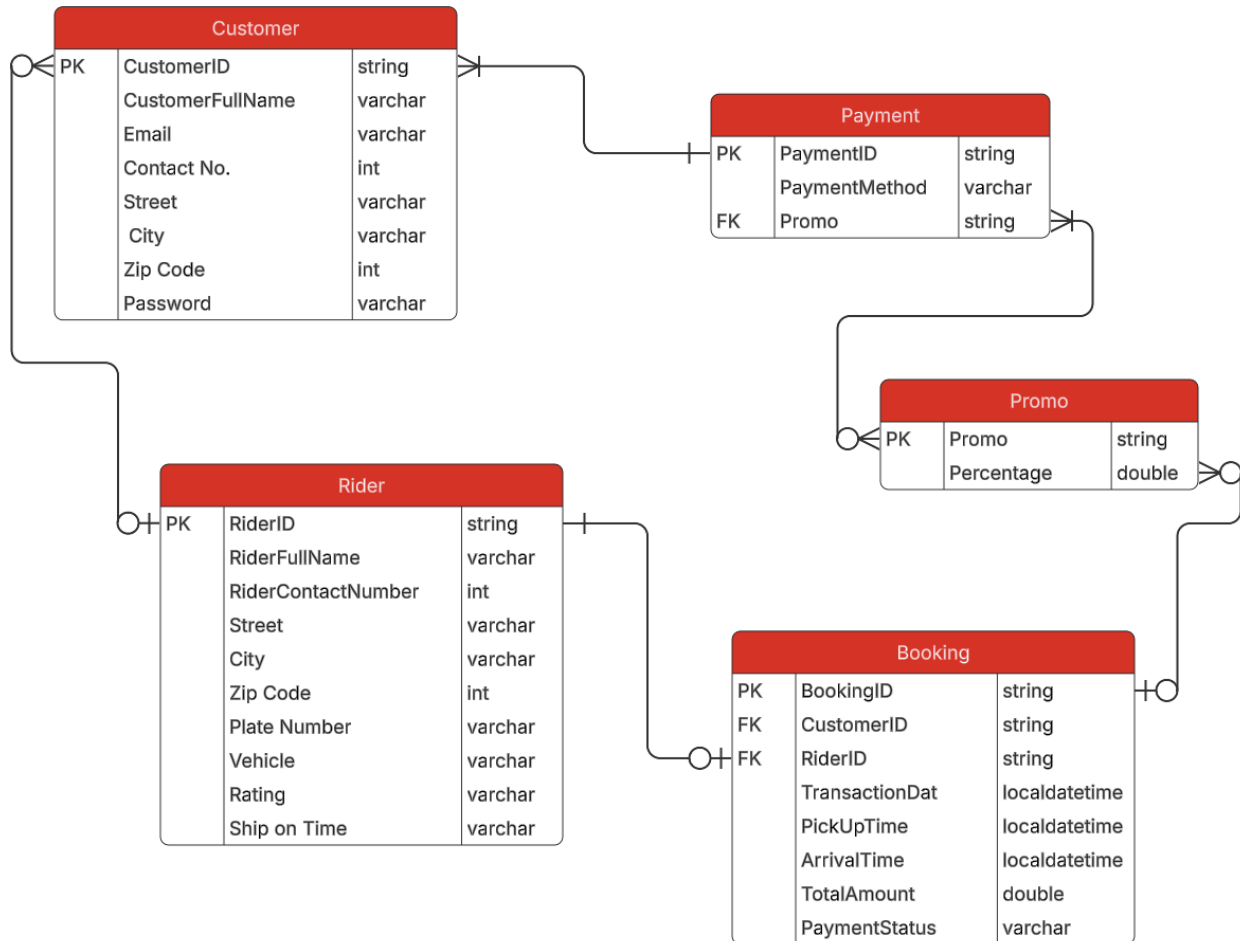
- It describes the tables, relationships, and rules that ensure data is stored efficiently and can be accessed or updated easily by the app. This overview also highlights the tools and techniques (like queries, triggers, and stored procedures) used to work with the data.
- This serves as a guide for building the database. It gives developers, database administrators, and stakeholders a clear picture of how everything fits together, making it easier to develop, maintain, and support the app as it grows.

## 1.2 Intended Audiences

- **Passengers:** Individuals looking for a fast, affordable, and reliable way to book motorcycle rides for personal travel or commuting.
- **Riders:** Licensed motorcycle drivers who want to earn income by providing ride-hailing or delivery services through the platform.
- **Businesses:** Small businesses or individuals needing quick and efficient delivery services for packages or goods.
- **Developers and Database Administrators (DBAs):** Technical teams responsible for building, maintaining, and optimizing the app's backend and database systems.

## 2 Entity Relationship Diagram (ERD)

### Entity Relationship Diagram



### 3 Detailed Database Design

#### 3.1.1.1 Data dictionary for Element: Customer Table

Name	Data Type	Constrain	Description
<b>Customer ID (primary key)</b>	String	Min: 4, Max:7	Unique identifier for each customer
<b>Customer Full Name</b>	String		Complete Name of the user
<b>Email</b>	String		Customer's email address for communication and log in
<b>Contact Number</b>	Integer	Starts at (09) Max:11	Customer's phone number
<b>Street</b>	String		More Information about the exact location of the customer
<b>City</b>	String		The city where the customer resides
<b>Zip Code</b>	Integer	Max: 4	The postal code of the customer's location
<b>Password</b>	String		The password of the user

#### 3.1.1.2 Data dictionary for Element: Rider Table

Name	Data Type	Constrain	Description
<b>Rider ID (primary key)</b>	String	Min :4, Max:7	The identifier for each rider
<b>Rider Full Name</b>	String		Complete Name of the Rider
<b>Rider Contact No.</b>	String	Starts at (09) Max:11	Contact No. of the rider
<b>Zip</b>	String	Max: 4	The postal code of the rider's registered location

<b>City</b>	String		The City where the rider operates
<b>Street</b>	String		To determine exact location of the Rider
<b>Plate Number</b>	String		The license plate number of the rider's vehicle
<b>Vehicle</b>	String		Name unit of the Vehicle
<b>Rating</b>	Int		Feedback about the ride
<b>Ship on Time</b>	String		A performance metric that tracks how often the rider arrives on time

#### 3.1.1.3 Data dictionary for Element: Booking Table

<b>Name</b>	<b>Data Type</b>	<b>Constrain</b>	<b>Description</b>
<b>Booking ID</b>	String	Min :4, Max:7	Unique reference number assigned to a booking
<b>Customer ID</b>	String	Min :4, Max:7	ID assigned to a customer by the service provider
<b>Rider ID</b>	String	Min :4, Max:7	To identify the rider assigned to a passenger
<b>Transaction Date</b>	LocalDateTime		The date when the booking was made and the payment was processed
<b>Pick up time</b>	LocalDateTime		Scheduled time at which a service provider or expected to pick a customer
<b>Arrival Time</b>	LocalDateTime		Expected time when a customer reaches its destination

<b>Total Amount</b>	Double		The total amount the customer has paid for the booking
<b>Payment Status</b>	String		Indicates the current status of the payment (“pending”, “Paid”&”Failed”)

#### 3.1.1.4 Data dictionary for Element: Payment Table

Name	Data Type	Constrain	Description
<b>Payment ID (primary key)</b>	String	Min: 4, Max:7	Unique identifier for each payment
<b>Payment Method</b>	String		Options for customer of how they are going to pay.
<b>Promo</b>	String		Codes that can be used to avail specific discounts.

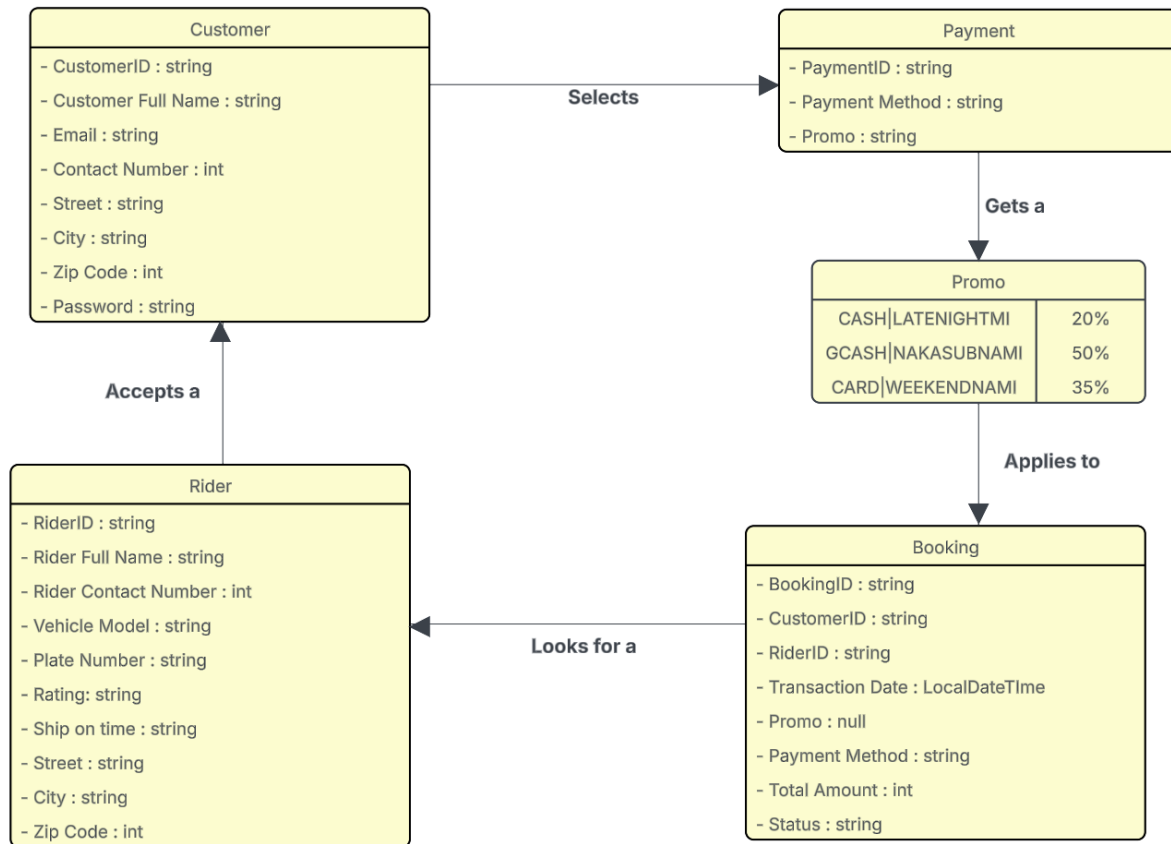
#### 3.1.1.5 Data dictionary for Element: Promo Table

Name	Data Type	Constrain	Description
<b>Promo</b>	String		Codes that can be used to avail specific discounts.
<b>Percentage (%)</b>	String		Refers to the discount value offered by a promotional code. It is represented as a numerical value (e.g., 20%, 50%)



## 3.2 MySQL database design (Relational database)

### 3.2.1 Conceptual diagram



### 3.2.2 Description

This diagram displays the conceptual model of the **IRideMoto Motorcycle Booking App** database. This database will be created to manage ride bookings, customer and rider information, promotional discounts, and feedback ratings. The system will store user details, including customers, who are currently logged in. Each user will have a unique ID, contact information, and a password for authentication. Customers can apply promotional discounts to their bookings, and riders will be assigned to fulfill those bookings.

The database will store information such as ride details (pickup location, destination, fare, and status), promotional codes, and rider ratings. Each booking will be linked to a customer and a rider, and after completing a booking the customer has the option to rate and give feedbacks to the rider. Selecting the payment option will give promotional discounts that can be applied to bookings, and the system will track payment methods and transaction details.

All sensitive data, such as passwords and payment information, will be encrypted for security. The database contents will be decrypted only when necessary, such as when a user makes a request or when the system

processes a transaction. This ensures data privacy and security while providing a seamless experience for customers and riders.

### **3.2.3 Purpose of Tables**

#### **3.2.3.1 Purpose of Customer**

This table stores information regarding customers who use the IRideMoto app to book rides. It includes personal details such as full name, email, contact number, and address (street, city, zip code). The table also contains a password field for secure authentication. Each customer can have multiple bookings over time but is limited to one active booking at a time. Customers can apply promo codes to their rides to receive discounts.

#### **3.2.3.2 Purpose of Rider**

This table stores all information related to the riders who fulfil bookings. It includes the rider's full name, contact details, vehicle model, and license plate number. The table also tracks rider performance through a rating system, ensuring accountability and service quality. While a rider can be assigned multiple booking requests over time, they can only accept and complete one booking at a time before taking another request. This ensures efficient ride management and prevents overloading riders with multiple simultaneous tasks.

#### **3.2.3.3 Purpose of Booking**

This table records all ride transactions made within the system. It stores details such as the booking ID, the customer who initiated the request, and the rider assigned to fulfil it. The table also keeps track of the transaction date, pickup and destination locations, total fare, payment method used, and booking status (e.g., "Pending", "Completed"). A booking may have a promo code applied, which reduces the total fare.

#### **3.2.3.4 Purpose of Payment**

This table stores information about payment methods used by customers for their bookings. It includes details such as the payment ID, payment method (Cash, GCash, Card), and any promo applied to the payment. The table ensures that payments are correctly linked to bookings and that promotional discounts are applied accurately. Each payment is associated with a single booking, ensuring proper tracking of transactions.

#### **3.2.3.5 Purpose of Promo**

This table stores promotional discount codes that can be applied to bookings (for one time use only). Each promo code has a corresponding discount percentage, allowing customers to receive fare reductions based on specific conditions, such as time of day or promotional events. The table ensures that promo codes are correctly mapped to bookings for accurate fare calculation.

### 3.2.4 Relations

From Table	To Table	Relation
Customer	Payment	A customer selects a Payment mode for the fare.
Payment	Promo	A payment gets a promo depending on the selected method.
Promo	Booking	A can be applied to a booking for a certain amount of discount.
Booking	Rider	A booking looks for an available rider in the area.
Rider	Customer	A rider can accept or decline a customer for the booking.

## 4 References

[1] "Lucidchart," [Online]. Available: <http://lucidchart.com/>.