



Name: Bokang Tjamela

Matriculation: 102401207

Tutor : Sharam Dadashnia

Introduction

- The Implementation Phase focuses on converting the system design into a fully functioning and operational database solution. After completing the conceptual and logical design, this phase translates all the models such as the ERD, data dictionaries, and business rules into actual SQL structures, test data, and operational database components. First, all 23 database tables were created using SQL CREATE TABLE statements, including primary keys, foreign keys, constraints, and data types that support data integrity and system rules. This ensures that the relationships designed in the ERD are accurately enforced within the database. Next, the database was populated with realistic test data, allowing the system to be tested with meaningful scenarios. The test data includes users, hosts, guests, listings, bookings, payments, amenities, reviews, notifications, and more. This enables full end-to-end testing of the platform's core features such as booking processes, payments, messaging, reporting, and user interactions. A fully connected ERD diagram was also generated to visually represent the database structure, showing how all tables relate to one another. This serves as a reference point for validating the implementation and ensuring alignment with the logical design. Finally, the entire database was exported into SQL format, which can be deployed, shared, or reused for future development stages. Overall, the Implementation Phase ensures that the system is correctly built, functional, test-ready, and aligned with the initial requirements.

DATA DICTIONARY

1. User:

UserID	INTEGER	Unique identifier for the user
Name	STRING	Name of the user
Email	STRING	Email address of the user
Phone	STRING	Phone number of the user
ProfilePicture	STRING	URL of the profile picture
VerifiedStatus	BOOLEAN	Verification status of the user

2. Guest:

GuestID	INTEGER	Unique identifier for the host
UserID	INTEGER	Foreign key to user
GuestRating	FLOAT	Rating of the guest

3. Host:

HostID	INTEGER	Unique identifier for the host
UserID	INTEGER	Foreign key to user
HostRating	FLOAT	Rating of the guest

4. Listing:

ListingID	INTEGER	Unique identifier for the listing
HostID	INTEGER	Foreign key to host
Title	STRING	Title of the listing
Description	STRING	Description of the listing
Location	STRING	Location of the listing
Price	FLOAT	Price per night
PropertyID	INTEGER	Type of the property (Foreign key)
MaxGuests	INTEGER	Maximum number of guests allowed
NumberOfBedRooms	INTEGER	Number of bedrooms
Rules	STRING	Rules for the listing

5. Booking:

BookingID	INTEGER	Unique identifier for the booking
GuestID	INTEGER	Foreign key to guest
ListingID	INTEGER	Foreign key to listing
CheckInDate	DATE	Check-in date
CheckOutDate	DATE	Check-out date
Status	STRING	Status of the booking
PaymentID	INTEGER	Foreign key to payment

6. Payment:

PaymentID	INTEGER	Unique identifier for the payment
BookingID	INTEGER	Foreign key to booking
Amount	Float	Amount paid
PaymentDate	DATE	Date of payment
PaymentMethodID	INTEGER	Foreign key to payment method

7. PaymentMethod:

MethodID	INTEGER	Unique identifier for the method
MethodName	STRING	Name of the payment
PaymentAccount	STRING	Account used for payment

8. Transaction:

TransactionID	INTEGER	Unique identifier for the transaction
UserID	INTEGER	Foreign key to user
BookingID	INTEGER	Foreign key to booking
Amount	FLOAT	Transaction amount
Date	DATE	Transaction date
PaymentID	INTEGER	Foreign key references to payment

9. Property:

PropertyID	INTEGER	Unique identifier for the property
.PropertyType	VARCHAR	Type of property
CountryID	INTEGER	Foreign key references to country
CityID	INTEGER	Foreign key references to city
Bedrooms	INTEGER	Number of bedrooms
Bathrooms	INTEGER	Number of bathrooms
Accommodates	INTEGER	
Size	FLOAT	Size of the property
Amenities	VARCHAR	Amenities
ImageURL	VARCHAR	URL to the image
Rating	FLOAT	Rating of the property

10. Tax:

TaxID	INTEGER	Unique identifier
BookingID	INTEGER	Foreign key to booking
Amount	FLOAT	Tax amount

11. Review:

ReviewID	INTEGER	Unique identifier for the review
ReviewerID	INTEGER	Foreign key to the user
ListingID	INTEGER	Foreign key to listing
Rating	FLOAT	Rating given
Comments	STRING	Review comment
ReviewDate	DATE	Date of the review

12. Report:

ReportID	INTEGER	Unique identifier for the report
ReporterID	INTEGER	Foreign key to the reporter
ListingID	INTEGER	Foreign key to listing
ReportReason	STRING	Reason for reporting
ReportDate	DATE	Date for reporting

ReportStatus	STRING	Status of the report
--------------	--------	----------------------

13. Message:

MessageID	INTEGER	Unique identifier for the message
SenderId	INTEGER	Foreign key to the user
ReceiverID	INTEGER	Foreign key to user
Content	STRING	Content of the message

14. Notification:

NotificationID	INTEGER	Unique identifier for the notification
UserID	INTEGER	Foreign key to user
SenderId	INTEGER	Foreign key to the sender
ReceiverID	INTEGER	Foreign key to the receiver
TextMessage	STRING	Notification text
Date	DATE	Date of the notification

15. Policy:

PolicyID	INTEGER	Unique identifier for the policy
Description	STRING	Description of the policy

16. Amenity:

AmenityID	INTEGER	Description
Name	STRING	Name of the amenity
ListingID	INTEGER	Foreign key to the listing
AmenityID	INTEGER	Foreign key to the amenity

17. Experience:

ExperienceID	INTEGER	Unique identifier for the experience
HostID	INTEGER	Foreign key to the host
Rating	FLOAT	Rating of the experience

18. Image:

ImageID	INTEGER	Unique identifier for the image
URL	STRING	URL to the image
MediaType	STRING	Type of the media

19. Complaint:

ComplaintID	INTEGER	Unique identifier for the complaint
BookingID	INTEGER	Foreign key to booking
ComplaintText	STRING	Text of the complaint
ComplaintDate	DATE	Date of complaint

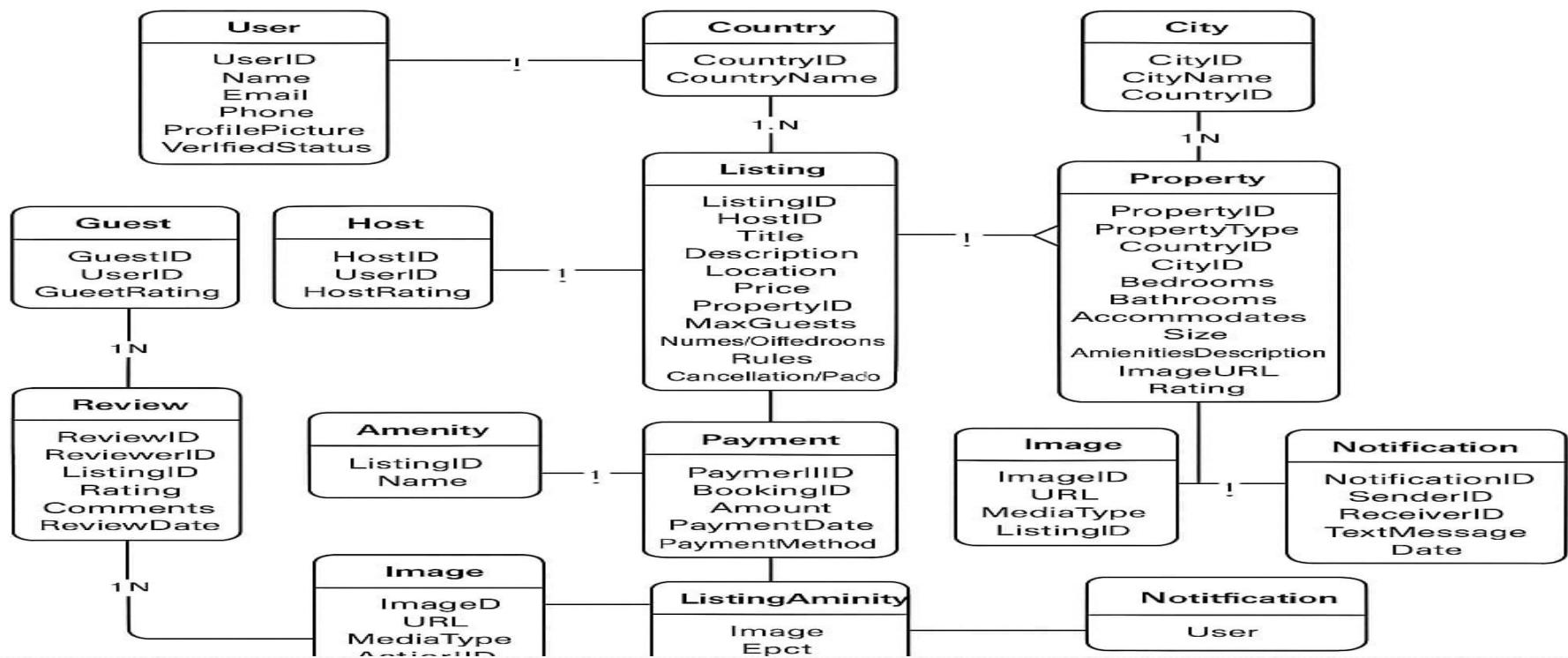
20. Social media platform:

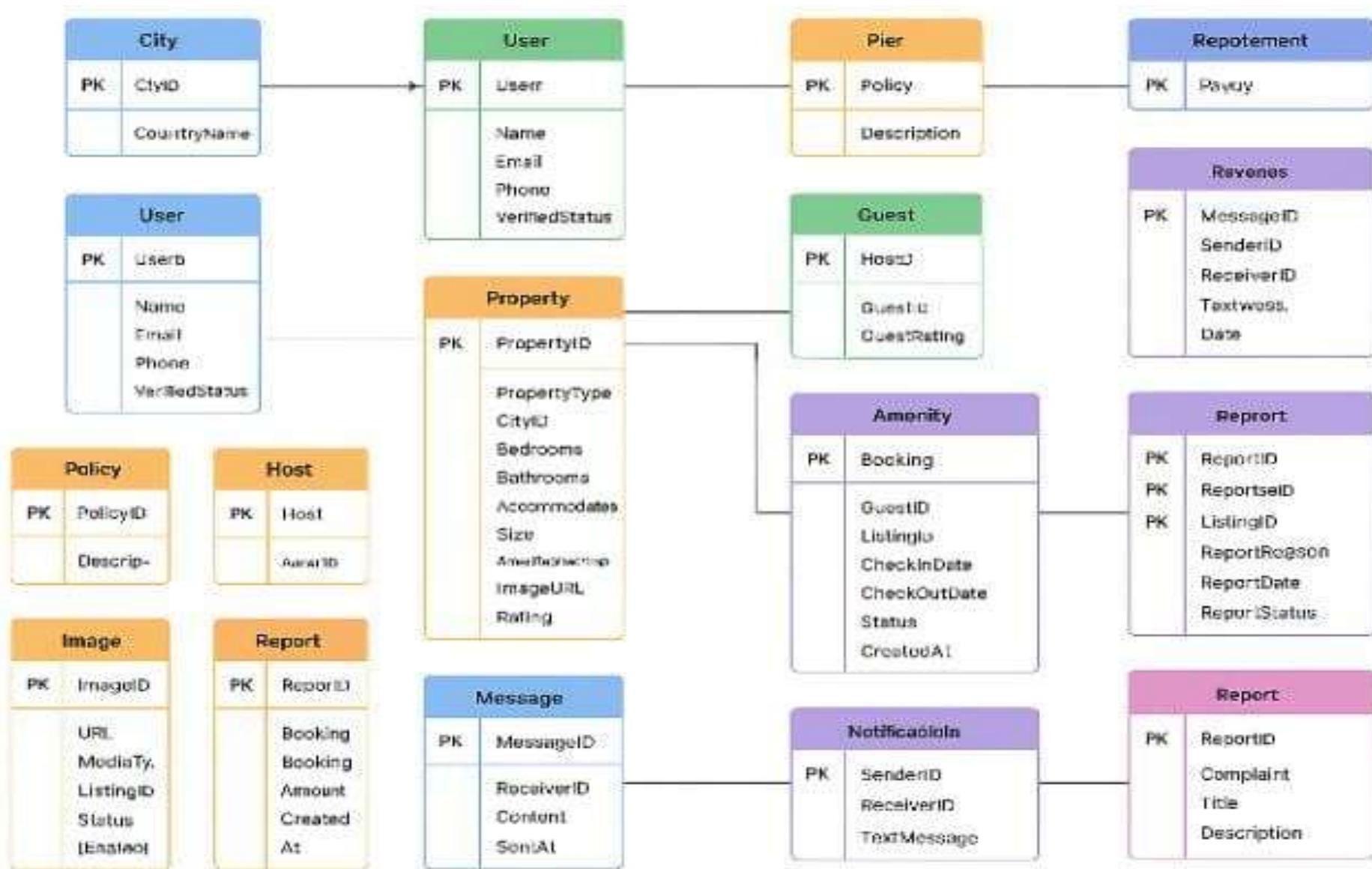
Platform_id	INTEGER	Unique identifier for the platform
Platform_name	STRING	Name of the platform
Website_url	STRING	URL to the platform

21. User social account:

user_social_media_id	INTEGER	Unique identifier for the account
User_id	INTEGER	Foreign key references the user
Platform_id	INTEGER	Foreign key references the platform
Profile_url	STRING	URL to user account
Username	STRING	Username of the user

ERD MODEL





SQL STATEMENTS



server version

Anfrage GET(
Server
E = null
if fo

def doMain()

```
CREATE TABLE Country (
    CountryID INT PRIMARY KEY,
    CountryName VARCHAR(100)
);
```

```
INSERT INTO Country (CountryID, CountryName) VALUES
(1, 'South Africa'),
(2, 'United States');
```

TABLE: Country

Column	Type	Constraints
CountryID	INT	PK, AUTO_INCREMENT
CountryName	VARCHAR(100)	NOT NULL

CountryID	CountryName
1	South Africa
2	United States

```
CREATE TABLE City (
    CityID INT PRIMARY KEY,
    CityName VARCHAR(100),
    CountryID INT,
    FOREIGN KEY (CountryID) REFERENCES Country(CountryID)
);
```

```
INSERT INTO City (CityID, CityName, CountryID) VALUES
(1, 'Johannesburg', 1),
(2, 'Cape Town', 1),
(3, 'New York', 2);
```

TABLE: City

Column	Type	Constraints
CityID	INT	PK, AUTO_INCREMENT
CityName	VARCHAR(100)	NOT NULL
CountryID	INT	FK → Country(CountryID)

CityID	CityName	CountryID
1	Johannesburg	1
2	Cape Town	1
3	New York	2

```
CREATE TABLE User (
    UserID INT PRIMARY KEY,
    Name VARCHAR(100),
    Email VARCHAR(255),
    Phone VARCHAR(20),
    ProfilePicture VARCHAR(255),
    VerifiedStatus BOOLEAN
);
```

02:45

```
INSERT INTO `User` (UserID, Name, Email, Phone,
ProfilePicture, VerifiedStatus) VALUES
(1, 'John Mokoena', 'john.mokoena@example.com',
'0723456789', 'profiles/john.jpg', 1),
(2, 'Mary Smith', 'mary.smith@example.com', '0839876543',
'profiles/mary.jpg', 1),
(3, 'Thabo Nkosi', 'thabo.nkosi@example.com',
'0612345678', 'profiles/thabo.jpg', 0),
(4, 'Emily Jacobs', 'emily.jacobs@example.com',
'0746582390', 'profiles/emily.jpg', 1),
(5, 'Sipho Dlamini', 'sipho.dlamini@example.com',
'0785559012', 'profiles/sipho.jpg', 0);
```

02:46

TABLE: User

Column	Type	Constraints
UserID	INT	PK, AUTO_INCREMENT
Name	VARCHAR(100)	NOT NULL
Email	VARCHAR(255)	UNIQUE
Phone	VARCHAR(20)	
ProfilePicture	VARCHAR(255)	
VerifiedStatus	TINYINT(1)	DEFAULT 0
CreatedAt	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP

02:47

UserID	Name	Email	Phone
1	John Mokoena	john@example.com	0723456789

02:48

```
CREATE TABLE SocialMediaPlatform (
    PlatformID INT PRIMARY KEY,
    PlatformName VARCHAR(100),
    WebsiteUrl VARCHAR(255)
);
```

02:23

```
INSERT INTO SocialMediaPlatform (PlatformID,
PlatformName, WebsiteUrl) VALUES
(1, 'Facebook', 'https://facebook.com'),
(2, 'Instagram', 'https://instagram.com'),
(3, 'Twitter', 'https://twitter.com');
```

02:25

TABLE: SocialMediaPlatform

Column	Type	Constraints
PlatformID	INT	PK, AUTO_INCREMENT
PlatformName	VARCHAR(100)	NOT NULL
WebsiteURL	VARCHAR(512)	

02:26

PlatformID	PlatformName	WebsiteUrl
------------	--------------	------------

1	Facebook	https://facebook.com
---	----------	----------------------

02:29

```

CREATE TABLE UserSocialAccount (
    UserSocialAccountId INT PRIMARY KEY,
    UserID INT,
    PlatformID INT,
    ProfileUrl VARCHAR(255),
    Username VARCHAR(255),
    FOREIGN KEY (UserID) REFERENCES User(UserID),
    FOREIGN KEY (PlatformID) REFERENCES SocialMediaPlatform(PlatformID)
);

```

02:31

```

INSERT INTO UserSocialAccount (UserSocialAccountId, UserID, PlatformID, ProfileUrl, Username) VALUES
(1, 1, 1, 'https://facebook.com/john.mokoena', 'john.mokoena'),
(2, 2, 1, 'https://facebook.com/mary.smith', 'mary.smith'),
(3, 4, 2, 'https://instagram.com/emily.j', 'emily.j'),
(4, 7, 2, 'https://instagram.com/zanele_k', 'zanele_k'),
(5, 9, 3, 'https://twitter.com/karabo_m', 'karabo_m')

```

02:32

Column	Type	Constraints
UserSocialAccountId	INT	PK, AUTO_INCREMENT
UserID	INT	FK → User(UserID)
PlatformID	INT	FK → SocialMediaPlatform
ProfileURL	VARCHAR(512)	
Username	VARCHAR(150)	UNIQUE per-user/platform

02:33

UserSocialAccountId	UserID	PlatformID	Username
1	1	1	john.mokoena

02:33

```

CREATE TABLE Policy (
    PolicyID INT PRIMARY KEY,
    Description TEXT
);

```

02:36

```

INSERT INTO Policy (PolicyID, Description) VALUES
(1, 'Flexible: full refund 1 day prior to arrival.'),
(2, 'Moderate: 50% refund up to 7 days before arrival')

```

02:37

TABLE: Policy

Column	Type	Constraints
PolicyID	INT	PK, AUTO_INCREMENT
Description	TEXT	

02:38

PolicyID Description

1	Full refund 1 day before arrival
---	----------------------------------

02:38

```

CREATE TABLE Property (
    PropertyID INT PRIMARY KEY,
    PropertyType VARCHAR(100),
    CountryID INT,
    CityID INT,
    Bedrooms INT,
    Bathrooms INT,
    Accommodates INT,
    Size DECIMAL(10,2),
    AmenitiesDescription TEXT,
    ImageURL VARCHAR(255),
    Rating DECIMAL(3,1)
);

```

02:45

```

INSERT INTO Property (PropertyID, PropertyType,
CountryID, CityID, Bedrooms, Bathrooms, Accommodates,
Size, AmenitiesDescription, ImageURL, Rating) VALUES
(1, 'Apartment', 1, 1, 2, 1, 4, 75.00, 'Close to
transport, small balcony', 'properties/apt1.jpg', 4.6),
(2, 'House', 1, 2, 3, 2, 6, 180.50, 'Garden, parking, sea
view', 'properties/house1.jpg', 4.9),
(3, 'Loft', 2, 3, 1, 1, 2, 45.00, 'Urban loft, modern
appliances', 'properties/loft1.jpg', 4.4),
(4, 'Cottage', 1, 2, 2, 1, 4, 90.00, 'Quiet
neighbourhood, fireplace', 'properties/cottage1.jpg',
4.5);

```

02:46

TABLE: Property

Column	Type	Constraints
PropertyID	INT	PK, AUTO_INCREMENT
PropertyType	VARCHAR(100)	
CountryID	INT	FK → Country
CityID	INT	FK → City
Bedrooms	INT	DEFAULT 0
Bathrooms	INT	DEFAULT 0
Accommodates	INT	DEFAULT 1

02:48

PropertyID	Type	CityID	Bedrooms
------------	------	--------	----------

```

CREATE TABLE Host (
    HostID INT PRIMARY KEY,
    UserID INT,
    HostRating DECIMAL(3,1),
    PhoneNumber VARCHAR(20),
    FOREIGN KEY (UserID) REFERENCES User(UserID)
);

```

02:51

```

INSERT INTO Host (HostID, UserID, HostRating,
PhoneNumber) VALUES
(1, 2, 4.9, '0839876543'),
(2, 5, 4.6, '0785559012'),
(3, 6, 4.3, '0659087772'),
(4, 8, 4.7, '0846653321'),
(5, 10, 4.8, '0792234001');

```

02:52

TABLE: Host

Column	Type	Constraints
HostID	INT	PK, AUTO_INCREMENT
UserID	INT	FK (Unique, 1:1 with User)
HostRating	DECIMAL	
PhoneNumber	VARCHAR	

02:53

HostID	UserID	HostRating
--------	--------	------------

1	2	4.9
---	---	-----

02:53

```
CREATE TABLE Guest (
    GuestID INT PRIMARY KEY,
    UserID INT,
    GuestRating DECIMAL(3,1),
    FOREIGN KEY (UserID) REFERENCES User(UserID)
);
```

02:55

```
INSERT INTO Guest (GuestID, UserID, GuestRating) VALUES
(1, 1, 4.5),
(2, 3, 4.2),
(3, 4, 3.9),
(4, 7, 4.8),
(5, 9, 4.7),
(6, 2, 4.6); -- Mary also as guest
```

02:56

TABLE: Guest

Column	Type	Constraints
GuestID	INT	PK, AUTO_INCREMENT
UserID	INT	FK (Unique, 1:1 with User)
GuestRating	DECIMAL	

02:57

GuestID UserID GuestRating

1 1 4.5

02:57

```
CREATE TABLE Listing (
    ListingID INT PRIMARY KEY,
    HostID INT,
    Title VARCHAR(255),
    Description TEXT,
    Location VARCHAR(255),
    Price DECIMAL(10,2),
    PropertyID INT,
    MaxGuests INT,
    NumberOfBedrooms INT,
    Rules TEXT,
    CancellationPolicyID INT,
    CreatedAt DATETIME
);
```

02:59

```
INSERT INTO Listing (ListingID, HostID, Title, Description, Location, Price, PropertyID, MaxGuests, NumberOfBedrooms, Rules, CancellationPolicyID, CreatedAt) VALUES
(1, 1, 'Sunny Apartment in Sandton', 'Bright 2-bed apt near malls and business district.', 'Sandton, Johannesburg', 850.00, 1, 4, 2, 'No smoking', 1, '2025-10-01 09:00:00'),
(2, 2, 'Luxury Sea View House', 'Spacious house with ocean view and private garden.', 'Sea Point, Cape Town', 2500.00, 2, 8, 3, 'No parties', 2, '2025-09-15 12:00:00'),
(3, 3, 'Modern City Loft', 'Stylish loft in downtown Manhattan.', 'Manhattan, New York', 1200.00, 3, 2, 1, 'No pets', 1, '2025-08-20 15:00:00');
```

TABLE: Listing

Column	Type	Constraints
ListingID	INT	PK, AUTO_INCREMENT
HostID	INT	FK → Host
Title	VARCHAR(255)	NOT NULL
Description	TEXT	
Location	VARCHAR(255)	
Price	DECIMAL	NOT NULL

ListingID	Title	Price	HostID
1	Sunny Apartment in Sandton	850	1

```
CREATE TABLE Amenity (
    AmenityID INT PRIMARY KEY,
    Name VARCHAR(255)
);
```

04:41

```
CREATE TABLE ListingAmenity (
    ListingAmenityID INT PRIMARY KEY,
    ListingID INT,
    AmenityID INT
);
```

04:45

```
INSERT INTO Amenity (AmenityID, Name) VALUES
(1, 'WiFi'),
(2, 'Parking'),
(3, 'Air Conditioning'),
(4, 'Kitchen'),
(5, 'Washer'),
(6, 'Free toiletries');
```

04:41

```
INSERT INTO ListingAmenity (ListingAmenityID, ListingID,
AmenityID) VALUES
(1, 1, 1),
(2, 1, 4),
(3, 1, 5),
(4, 2, 2),
(5, 2, 3),
(6, 2, 4),
(7, 3, 1),
(8, 3, 3),
(9, 4, 4),
(10, 5, 1),
(11, 5, 6),
(12, 6, 1);
```

04:45

TABLE: Amenity

Column	Type	Constraints
AmenityID	INT	PK, AUTO_INCREMENT
Name	VARCHAR(150)	NOT NULL

04:42

TABLE: ListingAmenity

Column	Type	Constraints
ListingAmenityID	INT	PK, AUTO_INCREMENT
ListingID	INT	FK → Listing(ListingID)
AmenityID	INT	FK → Amenity(AmenityID)

04:46

AmenityID	Name
1	WiFi

04:43

ListingAmenityID	ListingID	AmenityID
1	1	1

04:47

```
CREATE TABLE Image (
    ImageID INT PRIMARY KEY,
    URL VARCHAR(255),
    MediaType VARCHAR(50),
    ListingID INT,
    UploadedAt DATETIME
);
```

```
INSERT INTO Image (ImageID, URL, MediaType, ListingID,
UploadedAt) VALUES
(1, 'images/listing1_1.jpg', 'image/jpeg', 1, '2025-09-01
08:00:00'),
(2, 'images/listing1_2.jpg', 'image/jpeg', 1, '2025-09-01
08:05:00'),
(3, 'images/listing2_1.jpg', 'image/jpeg', 2, '2025-09-16
12:10:00'),
(4, 'images/listing3_1.jpg', 'image/jpeg', 3, '2025-08-21
09:00:00'),
(5, 'images/listing4_1.jpg', 'image/jpeg', 4, '2025-11-05
10:05:00'),
(6, 'images/listing5_1.jpg', 'image/jpeg', 5, '2025-10-12
11:40:00');
```

TABLE: Image

Column	Type	Constraints
ImageID	INT	PK, AUTO_INCREMENT
URL	VARCHAR(512)	NOT NULL
MediaType	VARCHAR(50)	
ListingID	INT	FK → Listing
UploadedAt	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP

ImageID	ListingID	URL
1	1	images/listing1_1.jpg

```
CREATE TABLE Booking (
    BookingID INT PRIMARY KEY,
    GuestID INT,
    ListingID INT,
    CheckInDate DATE,
    CheckOutDate DATE,
    Status VARCHAR(50),
    CreatedAt DATETIME
);
```

```
INSERT INTO Booking (BookingID, GuestID, ListingID,
CheckInDate, CheckOutDate, Status, CreatedAt) VALUES
(1, 1, 1, '2025-12-01', '2025-12-05', 'Confirmed',
'2025-11-10 09:00:00'),
(2, 2, 3, '2025-11-20', '2025-11-25', 'Completed',
'2025-10-15 10:00:00'),
(3, 3, 5, '2025-12-10', '2025-12-15', 'Pending',
'2025-11-01 14:00:00'),
(4, 4, 4, '2025-12-02', '2025-12-04', 'Cancelled',
'2025-10-20 13:00:00'),
(5, 5, 2, '2025-12-18', '2025-12-22', 'Confirmed',
'2025-11-12 11:00:00'),
(6, 6, 6, '2025-12-20', '2025-12-23', 'Confirmed',
'2025-11-15 12:00:00');
```

TABLE: Booking

Column	Type	Constraints
BookingID	INT	PK, AUTO_INCREMENT
GuestID	INT	FK → Guest
ListingID	INT	FK → Listing
CheckInDate	DATE	NOT NULL
CheckOutDate	DATE	NOT NULL
Status	VARCHAR(50)	DEFAULT 'Pending'
CreatedAt	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP

BookingID	GuestID	ListingID	Checkin	Status
-----------	---------	-----------	---------	--------

```
CREATE TABLE PaymentMethod (
    PaymentMethodID INT PRIMARY KEY,
    MethodName VARCHAR(100),
    PaymentAccount VARCHAR(255)
);
```

09:32

```
-- PaymentMethod (3)
-- 
INSERT INTO PaymentMethod (PaymentMethodID, MethodName,
PaymentAccount) VALUES
(1, 'Credit Card', 'VISA **** 4242'),
(2, 'Debit Card', 'MASTERCARD **** 1122'),
(3, 'PayPal', 'paypal@payments.example');
```

09:33

TABLE: PaymentMethod

Column	Type	Constraints
PaymentMethodID	INT	PK, AUTO_INCREMENT
MethodName	VARCHAR(100)	NOT NULL
PaymentAccount	VARCHAR(255)	

09:34

PaymentMethodID MethodName

1	Credit Card
---	-------------

09:34

```
CREATE TABLE Transaction (
    TransactionID INT PRIMARY KEY,
    UserID INT,
    BookingID INT,
    PaymentID INT,
    Amount DECIMAL(10,2),
    TransactionType VARCHAR(50),
    Date DATETIME,
    RefundProcessedDate DATETIME
);
```

09:53

```
INSERT INTO 'Transaction' (TransactionID, UserID,
BookingID, PaymentID, Amount, TransactionType, Date,
RefundProcessedDate) VALUES
(1, 1, 1, 1, 3400.00, 'Payment', '2025-11-10 09:10:00',
NULL),
(2, 3, 2, 2, 6000.00, 'Payment', '2025-10-15 10:05:00',
NULL),
(3, 4, 3, 3, 7500.00, 'Payment', '2025-11-02 08:30:00', '09:54
NULL);
```

TABLE: Transaction

Column	Type	Constraints
TransactionID	INT	PK, AUTO_INCREMENT
UserID	INT	FK → User
BookingID	INT	FK → Booking
PaymentID	INT	FK → Payment
Amount	DECIMAL	NOT NULL
TransactionType	VARCHAR(100)	
Date	DATETIME	DEFAULT CURRENT_TIMESTAMP
RefundProcessedDate	DATETIME	NULL

09:54

TransactionID UserID BookingID Amount Type

1	1	1	3400	Payment
---	---	---	------	---------

09:54

```
CREATE TABLE Tax (
    TaxID INT PRIMARY KEY,
    BookingID INT,
    Amount DECIMAL(10,2)
);
```

09:58

```
-- Tax (4)
INSERT INTO Tax (TaxID, BookingID, Amount) VALUES
(1, 1, 340.00),
(2, 2, 600.00),
(3, 5, 900.00),
(4, 6, 210.00);
```

09:59

TABLE: Tax

Column	Type	Constraints
TaxID	INT	PK, AUTO_INCREMENT
BookingID	INT	FK → Booking
Amount	DECIMAL	NOT NULL

09:59

TaxID	BookingID	Amount
1	1	340

09:59

```
CREATE TABLE Review (
    ReviewID INT PRIMARY KEY,
    ReviewerID INT,
    ListingID INT,
    Rating DECIMAL(3,1),
    Comments TEXT,
    ReviewType VARCHAR(50),
    ReviewDate DATETIME
);
```

10:05

```
INSERT INTO Review (ReviewID, ReviewerID, ListingID,
Rating, Comments, ReviewType, ReviewDate) VALUES
(1, 1, 1, 4.5, 'Great apartment, very clean and close to
everything.', 'GuestToListing', '2025-12-06 10:00:00'),
(2, 2, 3, 5.0, 'Excellent stay, host was very helpful.',
'GuestToListing', '2025-11-26 09:00:00'),
(3, 4, 4, 4.0, 'Lovely cottage; peaceful neighbourhood.',
'GuestToListing', '2025-12-05 14:00:00'),
(4, 5, 2, 4.8, 'Stunning views and great amenities.',
'GuestToListing', '2025-12-23 12:00:00'),
(5, 6, 6, 4.2, 'Compact but comfortable for business
trip.', 'GuestToListing', '2025-12-24 09:30:00'),
(6, 2, 1, 4.0, 'Good location but parking limited.',
'GuestToListing', '2025-12-07 16:00:00');
```

10:06

TABLE: Review

Column	Type	Constraints
ReviewID	INT	PK, AUTO_INCREMENT
ReviewerID	INT	FK → User
ListingID	INT	FK → Listing
Rating	DECIMAL	NOT NULL
Comments	TEXT	
ReviewType	VARCHAR(50)	
ReviewDate	DATETIME	DEFAULT CURRENT_TIMESTAMP

10:06

ReviewID	ReviewerID	ListingID	Rating
1	1	1	4.5

10:07

```
CREATE TABLE Message (
    MessageID INT PRIMARY KEY,
    SenderID INT,
    ReceiverID INT,
    Content TEXT,
    SentAt DATETIME
);
```

10:09

```
INSERT INTO Message (MessageID, SenderID, ReceiverID,
Content, SentAt) VALUES
(1, 1, 2, 'Hi Mary – is the Sandton apartment available
for December?', '2025-11-01 09:05:00'),
(2, 2, 1, 'Yes John, it is available. I will hold the
dates for you.', '2025-11-01 09:10:00'),
(3, 3, 2, 'Hi, can I check in early on the 20th?',
'2025-10-10 08:00:00'),
(4, 2, 3, 'Sure Thabo, early check-in is possible for a
small fee.', '2025-10-10 08:20:00'),
(5, 4, 5, 'Thanks for the quick response!', '2025-11-10
12:00:00'),
(6, 5, 4, 'Are pets allowed at the cottage?', '2025-11-11
13:20:00');
```

10:10

Column	Type	Constraints
MessageID	INT	PK, AUTO_INCREMENT
SenderId	INT	FK → User
ReceiverID	INT	FK → User
Content	TEXT	
SentAt	DATETIME	DEFAULT CURRENT_TIMESTAMP

10:10

MessageID	SenderId	ReceiverID	Content
1	1	2	Is the apartment free?

10:10

```
CREATE TABLE Notification (
    NotificationID INT PRIMARY KEY,
    SenderID INT,
    ReceiverID INT,
    TextMessage TEXT,
    Date DATETIME,
    IsRead BOOLEAN
);
```

10:12

```
INSERT INTO Notification (NotificationID, SenderID,
ReceiverID, TextMessage, Date, IsRead) VALUES
(1, 2, 1, 'Your booking has been confirmed.', '2025-11-10
09:12:00', 0),
(2, NULL, 2, 'New message from John Mokoena.', '2025-11-01 09:06:00', 1),
(3, 1, 3, 'Your reservation is pending payment.', '2025-11-02 08:35:00', 0),
(4, 5, 4, 'Booking cancelled by guest.', '2025-10-20
13:05:00', 1),
(5, 2, 6, 'New review received for your listing.', '2025-12-07 16:05:00', 0),
(6, 3, 5, 'Host replied to your message.', '2025-11-10
12:05:00', 0);
```

10:12

Column	Type	Constraints
NotificationID	INT	PK, AUTO_INCREMENT
SenderId	INT	FK → User (nullable)
ReceiverID	INT	FK → User
TextMessage	VARCHAR(1000)	
Date	DATETIME	DEFAULT CURRENT_TIMESTAMP
IsRead	TINYINT(1)	DEFAULT 0

10:13

NotificationID	ReceiverID	TextMessage	IsRead
1	1	Booking confirmed	No

10:13

```
CREATE TABLE Report (
    ReportID INT PRIMARY KEY,
    ReporterID INT,
    ListingID INT,
    ReportReason TEXT,
    ReportDate DATETIME,
    ReportStatus VARCHAR(50)
);
```

```
INSERT INTO `Report` (ReportID, ReporterID, ListingID,
ReportReason, ReportDate, ReportStatus) VALUES
(1, 1, 2, 'Listing description inaccurate about view.',
'2025-11-26 10:00:00', 'Open'),
(2, 4, 3, 'Host did not provide agreed amenities.',
'2025-11-27 11:00:00', 'Under review');
```

TABLE: Report

Column	Type	Constraints
ReportID	INT	PK, AUTO_INCREMENT
ReporterID	INT	FK → User
ListingID	INT	FK → Listing (nullable)
ReportReason	TEXT	
ReportDate	DATETIME	DEFAULT CURRENT_TIMESTAMP
ReportStatus	VARCHAR(50)	DEFAULT 'Open'

ReportID	ReporterID	ListingID	Status
1	1	2	Open

```
CREATE TABLE Complaint (
    ComplaintID INT PRIMARY KEY,
    BookingID INT,
    ComplaintText TEXT,
    ComplaintDate DATETIME
);
```

```
INSERT INTO Complaint (ComplaintID, BookingID,
ComplaintText, ComplaintDate) VALUES
(1, 4, 'Host double-booked the cottage for our dates.',
'2025-10-21 11:30:00'),
(2, 3, 'Noise from neighboring property very loud at
night.', '2025-12-12 07:45:00');
```

TABLE: Complaint

Column	Type	Constraints
ComplaintID	INT	PK, AUTO_INCREMENT
BookingID	INT	FK → Booking
ComplaintText	TEXT	
ComplaintDate	DATETIME	DEFAULT CURRENT_TIMESTAMP

ComplaintID	BookingID	ComplaintText	Date
1	4	Host double-booked property	10:38

```
CREATE TABLE Experience (
    ExperienceID INT PRIMARY KEY,
    HostID INT,
    Title VARCHAR(255),
    Description TEXT,
    Rating DECIMAL(3,1),
    CreatedAt DATETIME
);
```

10:39

```
INSERT INTO Experience (ExperienceID, HostID, Title,
Description, Rating, CreatedAt) VALUES
(1, 1, 'City Food Tour', 'Guided food tasting around
Sandton.', 4.8, '2025-09-01 09:00:00'),
(2, 2, 'Coastal Kayaking', 'Kayak with an experienced
guide along the coast.', 4.9, '2025-09-20 11:00:00'),
(3, 3, 'Yoga on the Beach', 'Sunrise beach yoga
sessions.', 4.6, '2025-10-05 06:30:00');
```

10:39

```
TABLE: Experience
+-----+-----+-----+
| Column      | Type       | Constraints
+-----+-----+-----+
| ExperienceID | INT        | PK, AUTO_INCREMENT
| HostID       | INT        | FK → Host
| Title         | VARCHAR(255) |
| Description   | TEXT       |
| Rating        | DECIMAL    |
| CreatedAt     | DATETIME   | DEFAULT CURRENT_TIMESTAMP
+-----+-----+-----+
```

10:40

ExperienceID	HostID	Title	Rating
1	1	City Food Tour	4.8

10:40



thank
you