Tourism Management System

Group No: 7

TA: Kashyap Nirmal (202011031)

Group members:

- Riya Dineshkumar Soni (202012025)
- Kakkan Anurag Kishor (202012026)
- Gandhi Viral Ashok (202012027)
- Sukhadia Rutvi Kumarpal (202012028)
- Shah Siddhant Alkeshbhai (202018013)
- Shah Nihar Shaileshbhai (202018014)

Functional Dependencies, Constraints And Normal Forms

1)

Table:

User (UserAadharNo, Fname, Lname, Email, Phone, Gender, Age)

Functional Dependencies:

UserAadharNo → Fname
UserAadharNo → Lname
UserAadharNo → Email
UserAadharNo → Phone
UserAadharNo → Gender
UserAadharNo → Age

Email → UserAadharNo
Email → Fname
Email → Lname
Email → Phone
Email → Gender
Email → Age

Phone → UserAadharNo
Phone → Fname
Phone → Lname
Phone → Email
Phone → Gender
Phone → Age

Closure sets:

$\{UserAadharNo\}^+ = \{UserAadharNo, Fname, Lname, Email, Phone, Gender, Age\}$

$\{Email\}^+ = \{UserAadharNo, Fname, Lname, Email, Phone, Gender, Age\}$

$\{Phone\}^+ = \{UserAadharNo, Fname, Lname, Email, Phone, Gender, Age\}$

Constraints:

a) Primary Key: UserAadharNo
b) Foreign Key: None
c) Referential: None
d) Candidate Key: {UserAadharNo, Email, Phone}
d) Domain:

UserAadharNo: BIGSERIAL PRIMARY KEY
Fname: VARCHAR(10) NOT NULL,
Lname: VARCHAR(10) NOT NULL,
Email: VARCHAR(20) NOT NULL,
Phone: Numeric(10) NOT NULL,
IsActive: BOOLEAN DEFAULT TRUE,
Gender: CHAR(1) CHECK (GENDER IN('M','F','O')),
AGE: int

- Here, as we have {UserAadharNo, Email, Phone} as candidate key which defines all the attributes, so our relation is in BCNF Form.

2)

Table:

COPASSENGER **(UserAadharNo, CoPassID, Fname, Lname, Email, Phone, Gender, Age)**

Functional Dependencies:

{UserAadharNo, CoPassID} → Fname
{UserAadharNo, CoPassID} → Lname
{UserAadharNo, CoPassID} → Email
{UserAadharNo, CoPassID} → Phone
{UserAadharNo, CoPassID} → Gender
{UserAadharNo, CoPassID} → Age

Constraints:

a) Primary Key: (UserAadharNo, CoPassID)
b) Foreign Key: UserAadharNo
c) Referential: User Table
d) Candidate Key: {UserAadharNo, CoPassID}
e) Domain:

UserAadharNo: BIGSERIAL NOT NULL
CoPassID: int NOT NULL
Fname: VARCHAR(10) NOT NULL,
Lname: VARCHAR(10) NOT NULL
Email: VARCHAR(20)
Phone: DECIMAL(12,0)
Gender: CHAR(1) CHECK (GENDER IN('M','F','O'))
AGE: int

- Here, as we have (UserAadharNo, CoPassID) as candidate key which defines all the attributes, so our relation is in BCNF Form.

3)

Table:

Tourist_spots (spotid, Name, season, ratings, address, pincode)

Functional Dependencies:

spotid → Name
spotid → season
spotid → ratings
spotid → address
spotid → pincode

Closure sets:

{spotid} + = {Name, season, ratings, address, pincode}

Constraints:

a) Primary Key: spotid
b) Foreign Key: PINCODE
c) Referential: LOCATION
c) Candidate Key: spotid

d) Domain:

SPOTID: INT PRIMARY KEY GENERATED ALWAYS AS IDENTITY
Name: VARCHAR (20) NOT NULL
SEASON: VARCHAR (10)
RATINGS: FLOAT CHECK (RATINGS>=0 AND RATINGS<=5)
ADDRESS: VARCHAR (100)
PINCODE: Numeric (6)

- Here, as we have spotid as candidate key which defines all the attributes, so our relation is in BCNF Form.

4)

Table:

Guide (GuideAadharNo, fname, lname, email, phone, gender, age, address, pincode)

Functional Dependencies:

GuideAadharNo→ fname
GuideAadharNo→ lname
GuideAadharNo→ email
GuideAadharNo→ phone
GuideAadharNo→ gender
GuideAadharNo→ age
GuideAadharNo→ address
GuideAadharNo→ pincode

email → GuideAadharNo
email → fname
email → lname
email → phone
email → gender
email → age
email → address
email → pincode

phone → GuideAadharNo
phone → fname

phone → lname
phone → email
phone → gender
phone → age
phone → address
phone → pincode

Closure sets:

{GuideAadharNo}+ = {GuideAadharNo, fname, lname, email, phone, gender, age, address, pincode}

{email}+ = {GuideAadharNo, fname, lname, email, phone, gender, age, address, pincode}

{Phone}+ = {GuideAadharNo, fname, lname, email, phone, gender, age, address, pincode}

Constraints:

e) Primary Key: GuideAadharNo
f) Foreign Key: PINCODE
g) Referential: LOCATION
d) Candidate Key: (GuideAadharNo, email, phone)
h) Domain:

GuideAadharNo: BIGSERIAL PRIMARY KEY
Fname: VARCHAR(10) NOT NULL
Lname: VARCHAR(10) NOT NULL
Email: VARCHAR(20)
Phone: Numeric(10) NOT NULL
Gender: CHAR(1) CHECK (GENDER IN('M','F','O'))
AGE: int
ADDRESS: VARCHAR(100)
PINCODE: Numeric(6)

- Here, as we have (GuideAadharNo, email, phone) as candidate key which defines all the attributes, so our relation is in BCNF Form.

5)

Table:

Location (pincode, city, state)

Functional Dependencies:

pincode → city
pincode → state

{pincode }+ = {pincode , city , state }

Constraints:

a) Primary Key: PINCODE
b) Foreign Key: None
c) Referential: None
d) Candidate Key: Pincode
e) Domain:

PINCODE Numeric (6) PRIMARY KEY
CITY VARCHAR (30)

- Here, as we have Pincode as candidate key which defines all the attributes, so our relation is in BCNF Form.

6)

Table:

Hotel (hotelid, Name, phone, foodtype, ratings, address, isactive, pincode)

Functional Dependencies:

hotelid → Name
hotelid → phone
hotelid → foodtype
hotelid → ratings
hotelid → address

hotelid → isactive
hotelid → pincode

phone → hotelid
phone → Name
phone → foodtype
phone → ratings
phone → address
phone → isactive
phone → pincode

Closure Sets:

{hotelid }+ = {hotelid, Name, phone, foodtype, ratings, address, isactive, pincode}

{phone }+ = {hotelid, Name, phone, foodtype, ratings, address, isactive, pincode}

Constraints:

a) Primary Key: (hotelid)
b) Foreign Key: (PINCODE)
c) Referential: LOCATION
d) Candidate Key: (hotelid, phone)
d) Domain:

HOTELID: int PRIMARY KEY
Name: VARCHAR(50) NOT NULL15)
PHONE: Numeric(10)
FOODTYPE: VARCHAR(20) CHECK (FOODTYPE IN('VEG','NON-VEG','BOTH'))
RATINGS: float CHECK (RATINGS>=0 AND RATINGS<=5)
ADDRESS: VARCHAR(100)
IsActive: BOOLEAN DEFAULT TRUE
PINCODE: Numeric(6)

- Here, as we have (hotelid, phone) as candidate key which defines all the attributes, so our relation is in BCNF Form.

7)

Table:

Hotel_services (hotelid, services)

Functional Dependencies:

{hotelid, services} → hotelid
{hotelid, services} → services

Closure Sets:

{hotelid, services}+ = {hotelid, services}

Constraints:

a) Primary Key: (HOTELID, SERVICES)
b) Foreign Key: (HOTELID)
c) Referential: Hotel
d) Candidate Key: (hotelid, services)
e) Domain:

HOTELID: int NOT NULL
SERVICES: VARCHAR (50) NOT NULL

- Here, as we have (hotelid, services) as candidate key which defines all the attributes, so our relation is in BCNF Form.

8)

Table:

Room (hotelid, room_no, Type, beds, capacity, rate, status)

Functional Dependencies:

{hotelid, room_no} → hotelid
{hotelid, room_no} → room_no
{hotelid, room_no} → Type
{hotelid, room_no} → beds
{hotelid, room_no} → capacity
{hotelid, room_no} → rate
{hotelid, room_no} → status

Closure Sets:

{hotelid, room_no}+ = {hotelid, room_no, Type, beds, capacity, rate, status}

Constraints:

a) Primary Key: (hotelid, room_no)
b) Foreign Key: (HOTELID)
c) Referential: HOTEL
d) Candidate Key: (hotelid, room_no)
d) Domain:

HOTELID: int NOT NULL
ROOM_NO: DECIMAL(3,0) NOT NULL
Type: VARCHAR(6) CHECK("Type" IN('AC','NON-AC','HEATER'))
BEDS: int CHECK(BEDS>0)
CAPACITY: int CHECK(CAPACITY>0)
RATE : FLOAT
STATUS: VARCHAR(15) CHECK (STATUS IN('AVAILABLE','NOT-AVAILABLE','BOOKED'))

- Here, as we have (hotelid, room_no) as candidate key which defines all the attributes, so our relation is in BCNF Form.

9)

Table:

Room_facilities (hotelid, roomno, facility)

Functional Dependencies:

{hotelid, roomno, facility} → hotelid
{hotelid, roomno, facility} → roomno

{hotelid, roomno, facility} → facility

Closure Sets:

{hotelid, roomno, facility}+ = {hotelid, roomno, facility}

Constraints:

a) Primary Key: (hotelid, roomno, facility)
b) Foreign Key: (HOTELID, ROOM_NO)
c) Referential: ROOM
d) Candidate Key: (hotelid, roomno, facility)
e) Domain:

HOTELID: int NOT NULL
ROOM_NO: int NOT NULL
FACILITY: VARCHAR (50) NOT NULL

• Here, as we have (hotelid, roomno, facility) as candidate key which defines all the attributes, so our relation is in BCNF Form.

10)

Table:

Restaurant (rid, Name, phone, foodtype, ratings, address, pincode)

Functional Dependencies:

rid → Name
rid → phone
rid → foodtype
rid → ratings
rid → address
rid → pincode
phone → rid
phone → Name
phone → foodtype
phone → ratings
phone → address
phone → pincode

Closure Sets:

{rid}+ = {rid, Name, phone, foodtype, ratings, address, pincode}
{phone}+ = {rid, Name, phone, foodtype, ratings, address, pincode}

Constraints:

a) Primary Key: rid
b) Foreign Key: PINCODE
c) Referential: LOCATION
d) Candidate Key: (rid, phone)
d) Domain:

RID: INT PRIMARY KEY
Name: VARCHAR(50) NOT NULL
PHONE: Numeric(10)
FOODTYPE: VARCHAR(20) CHECK (FOODTYPE IN('VEG','NON-VEG','BOTH')
RATINGS: float CHECK (RATINGS>=0 AND RATINGS<=5)
ADDRESS: VARCHAR(100)
PINCODE: Numeric(6)

- Here, as we have (rid, phone) as candidate key which defines all the attributes, so our relation is in BCNF Form.

11)

Table:

Restaurant_cuisines (rid, cuisines)

Functional Dependencies:

rid, cuisines → rid
rid, cuisines → cuisines

Closure Sets:

{rid, cuisines}+ = {rid, cuisines}

Constraints:

a) Primary Key: (rid, cuisines)
b) Foreign Key: rid
c) Referential: RESTAURANT

d) Candidate Key: (rid, cuisines)
e) Domain:
RID: int NOT NULL
CUISINES: VARCHAR(20)

- Here, as we have (rid, cuisines) as candidate key which defines all the attributes, so our relation is in BCNF Form.

12)

Table:

Package (PackageID, Title, Duration, No_Of_people, Amount)

Functional Dependencies:

PackageID → Title
PackageID → Duration
PackageID → No_Of_People
PackageID → Amount

Closure Sets:

{PackageID }+ = {Title, Duration , No_Of_People, Amount}

Constraints:

a) Primary Key: PackageID
b) Foreign Key: -
c) Referential: -
d) Candidate Key: PackageID
e) Domain:
PACKAGEID: INT PRIMARY KEY
TITLE: VARCHAR (15) NOT NULL
DURATION: INT
NO_OF_PEOPLE: INT
IsActive: BOOLEAN DEFAULT TRUE
AMOUNT: float

- Here, as we have PackageID as candidate key which defines all the attributes, so

our relation is in BCNF Form.

13)

Table:

Package_includes_spots (packageid, spotid)

Functional Dependencies:

packageid, spotid → packageid
packageid, spotid → spotid

Closure Sets:

{packageid, spotid }+ = {packageid, spotid}

Constraints:

a) Primary Key: (PACKAGEID, SPOTID)
b) Foreign Key: (PACKAGEID, SPOTID)
c) Referential: (PACKAGE, TOURIST_SPOTS)
d) Candidate Key: (PACKAGEID, SPOTID)
d) Domain:
PACKAGEID: INT NOT NULL
SPOTID: INT NOT NULL

- Here, as we have (PACKAGEID, SPOTID) as candidate key which defines all the attributes, so our relation is in BCNF Form.

14)

Table:

Package_includes_guides (packageid, GuideAadharNo)

Functional Dependencies:

{packageid, GuideAadharNo } → packageid
{packageid, GuideAadharNo } → GuideAadharNo

Closure Sets:

{packageid, GuideAadharNo }+ = {packageid, GuideAadharNo}

Constraints:

a) Primary Key: (PACKAGEID, GuideAadharNo)
b) Foreign Key: (PACKAGEID, GuideAadharNo)
c) Referential: Package, Guide
d) Candidate Key: (PACKAGEID, GuideAadharNo)
d) Domain:

PACKAGEID int NOT NULL
GuideAadharNo BIGSERIAL NOT NULL

- Here, as we have (PACKAGEID, GuideAadharNo) as candidate key which defines all the attributes, so our relation is in BCNF Form.

15)

Table:

Package_includes_hotels (packageid, hotelid, roomno)

Functional Dependencies:

{packageid, hotelid, roomno} → packageid
{packageid, hotelid, roomno} → hotelid
{packageid, hotelid, roomno} → roomno

Closure Sets:

{packageid, hotelid, roomno}+ = {packageid, hotelid, roomno}

Constraints:

a) Primary Key: (packageid, hotelid, roomno)
b) Foreign Key: (HOTELID, ROOM_NO)
c) Referential: ROOM
d) Candidate Key: (packageid, hotelid, roomno)

e) Domain:

PACKAGEID: int NOT NULL
HOTELID: int NOT NULL
ROOM_NO: int NOT NULL

- Here, as we have (packageid, hotelid, roomno) as candidate key which defines all the attributes, so our relation is in BCNF Form.

16)

Table:

Booking (BID, UserAdhar_No, BookingDate, TripStartDate, TripEndDate, Amount)

Functional Dependencies:

BID → UserAdhar_No
BID → BookingDate
BID → TripStartDate
BID → TripEndDate
BID → Amount

Closure Sets:

{BID}+ = {BID, UserAdhar_No, BookingDate, TripStartDate, TripEndDate, Amount}

Constraints:

a) Primary Key: {Bid}
b) Foreign Key: UserAdhar_No
c) Candidate Key: {Bid}
d) Referential: User Table
e) Domain:

BID: Int UNIQUE,
UserAdhar_No: BIGSERIAL NOT NULL,
BookingDate: TIMESTAMP DEFAULT CURRENT_TIMESTAMP
TripStartDate: Date NOT NULL,
TripEndDate: Date NOT NULL,
Amount: FLOAT,

- Here, as we have {Bid} as candidate key which defines all the attributes, so our relation is also in BCNF Form.

17)

Table:

Booking_for_package (BID, PackageID)

Functional Dependencies:

BID → PackageID

Closure Sets:

{BID}+ = {BID, PackageID}

Constraints:

a) Primary Key: BID
b) Foreign Key: (BID, PACKAGEID)
c) Referential: BOOKING Table, PACKAGE Table,
d) Candidate Key: BID
e) Domain:
BID: INT Primary Key
PACKAGEID: INT NOT NULL

- Here, as we have BID as candidate key which defines all the attributes, so our relation is in BCNF Form.

18)

Table:

Booking_CoPassenger (BID, UserAadharNo, CoPassID)

Functional Dependencies:

BID → UserAadharNo
BID → CoPassID

Closure Sets:

{BID}+ = {BID, UserAadharNo, CoPassID}

Constraints:

a) Primary Key: (BID)
b) Foreign Key: (BID, UserAadharNo, CoPassID)
c) Referential: Booking, COPASSENGER
d) Candidate Key: BID
e) Domain:

BID: int NOT NULL
UserAadharNo: BIGSERIAL NOT NULL
CoPassID: int NOT NULL

- Here, as we have BID as candidate key which defines all the attributes, so our relation is in BCNF Form.

19)

Table:

Booking_for_hotel (bid, hotelid, roomno)

Functional Dependencies:

{bid, hotelid, roomno} → bid
{bid, hotelid, roomno} → hotelid
{bid, hotelid, roomno} → roomno

Closure Sets:

{bid, hotelid, roomno}+ = {bid, hotelid, roomno}

Constraints:

a) Primary Key: (bid, hotelid, roomno)
b) Foreign Key: (BID, HOTELID, ROOM_NO)
c) Referential: Booking, ROOM
d) Candidate Key: (bid, hotelid, roomno)

e) Domain:

BID: int NOT NULL
HOTELID: int NOT NULL
ROOM_NO: int NOT NULL

• Here, as we have (bid, hotelid, roomno) as candidate key which defines all the attributes, so our relation is in BCNF Form.