Information Systems and Data Modeling – IT1090

Assignment

Title: Life Insurance Management System	
Batch Number: Group 04.02	Group Number: MLB_04.02_08

Declaration:

We hold a copy of this assignment that we can produce if the original is lost or damaged.

We hereby certify that no part of this assignment has been copied from any other group's work or from any other source. No part of this assignment has been written / produced for our group by another person except where such collaboration has been authorized by the subject lecturer/tutor concerned.

Group Members:

IT23228276	Aaqib A.R	………………………… signature
IT23223530	Thakkif Ahamed I.M.	……………………… signature
IT23220942	Thanis Ahamed A.R.	………………………… signature
IT23222236	Erathnage T.N.	………………………… signature
IT23222786	Budara V.P.R.	………………………… signature

Submitted on: <03/05/2024>

Contents

1. Hypothetical Scenario

Ceylon Insurance is a company providing various life insurance policies to customers (individuals, families) Ceylon Insurance has been operating successfully for 10 years.

The life Insurance management system provides the ability to browse details about the company and guest customers can view insurance packages. After registering with the system customers can make a payment, make a claim, and select an insurance package. Update Beneficially and contact the company for any issues.

The system admin has access to manage FAQs, manage user accounts, and update Company details, The Manager can guide the admin and generate reports. The agent can approve the claim and select the correct claim for the customer.

The life insurance management system will store all the important data in the database for usage. (Details about Insurance package, claims, customers, beneficiaries.)

2. Requirements Analysis Document

2.1. Overview

A database is a data structure that holds structured, meaningful, and linked information. A database is a critical component of a system. A database is used to store enormous amounts of information. It is capable of establishing associations between data points. In databases, primary and foreign keys. Support data retrieval from many database tables.

The life insurance management system requires a database to hold significant amounts of data. The database will contain all data relating to clients, claims, and packagers.

The life insurance Management System makes use of a database to provide a better user experience to customers. Customers can also make decisions based on their saved data. The database system offers robust security for data collection. To prevent illegal entry.

2.2. Functional Requirements

1. Guest customers can visit the page without registering with the system.

2. Guest customers can register in the system.

3. Registered customers can apply for a claim.

4. Registered customers can apply for a package.

5. Registered customers should be able to make payments with cash, PayPal, Master, or VISA.

6. Registered customers can add feedback.

7. Admin should manage FAQs.

8. The admin manages accounts.

9. The manager can generate the reports.

10. The manager can guide the admin.

11. The agent can approve claims.

12. The agent can choose the claims.

13. Registered customer can choose their beneficially.

2.3. Non-functional Requirements

1. Usability:

The system should be simple to use and understand. This can be viewed from several perspectives.

▪ Execution time.

▪ Task completion without assistance.

▪ User interface simplicity and clarity.

2. Security:

The system must be secure against unauthorized access. This also includes antivirus and malware protection.

3. Performance:

The system must be able to manage the needed number of users without compromising performance.

4. Scalability:

The system structure must be able to scale up and down as needed.

5. Availability:

The system must be accessible when needed. A good system delivers services around all of the time.

3. Data Requirement

- Registered_Customer
    - Customer ID (customer_ID)
    - First Name (F_name)
    - Last Name (L_name)
    - Password (C_password)
    - Date of Birth (DOB)
    - Customer Address (C_Address)
    - Gender(Gender)

- ContactNo
    - Contact number (Contact_No)
    - Customer ID (customer_ID)

- Email
    - Email (Email)
    - Customer ID (customer_ID)

- Administer
    - Admin ID (Admin_ID)
    - First Name (F_name)
    - Last Name (L_name)
    - Password (Admin_Password)
    - Date of Birth (DOB)
    - Tele phone Number (phone_number)
    - Address (Ad_Address)
    - Gender(Gender)
    - Email (Email)

- Agent
    - ➢ Agent ID (Agent_ID)
    - ➢ First Name (F_name)
    - ➢ Last Name (L_name)
    - ➢ Password (Agent_password)
    - ➢ Date of Birth (DOB)
    - ➢ Tele phone Number (phone_number)
    - ➢ Address (Address)
    - ➢ Gender(Gender)
    - ➢ Email (Email)

- Manager
    - ➢ Manager ID(Manager_ID)
    - ➢ First Name (F_name)
    - ➢ Last Name (L_name)
    - ➢ Password (password)
    - ➢ Date of Birth (DOB)
    - ➢ Tele phone Number (phone_number)
    - ➢ Address (Address)
    - ➢ Gender(Gender)
    - ➢ Email (Email)

- Packagers
    - ➢ Package ID (Package_ID)
    - ➢ Package Type (Package_Type)
    - ➢ Amount (Amount)

- Reports
    - ➢ Report ID (Report_ID)
    - ➢ Report Type (Report_type)
    - ➢ Report submitted date (S_date)
    - ➢ Manager ID (Manager_ID)

- Beneficially
  - Beneficially Id (B_ID)
  - First Name (F_name)
  - Last Name (L_name)
  - Date of Birth (DOB)
  - Tele phone Number (phone_number)
  - Gender(Gender)
  - Email (Email)
  - Customer ID (customer_ID)

- Payment
  - Payment code(payment_code)
  - Payment amount (Amount)
  - Payment date (payment_date)
  - Customer ID (customer_ID)

- Claim
  - Claim ID (Claim_id)
  - Claim Type (Claim_type)
  - Claim Number (Claim_NO)
  - Post date (Post_date)
  - End date (End_date)
  - Customer ID (customer_ID)
  - Manager ID (Manager_ID)
  - Agent ID (Agent_ID)

- Method
  - Method ID (M_ID)
  - Method Type(M_type)

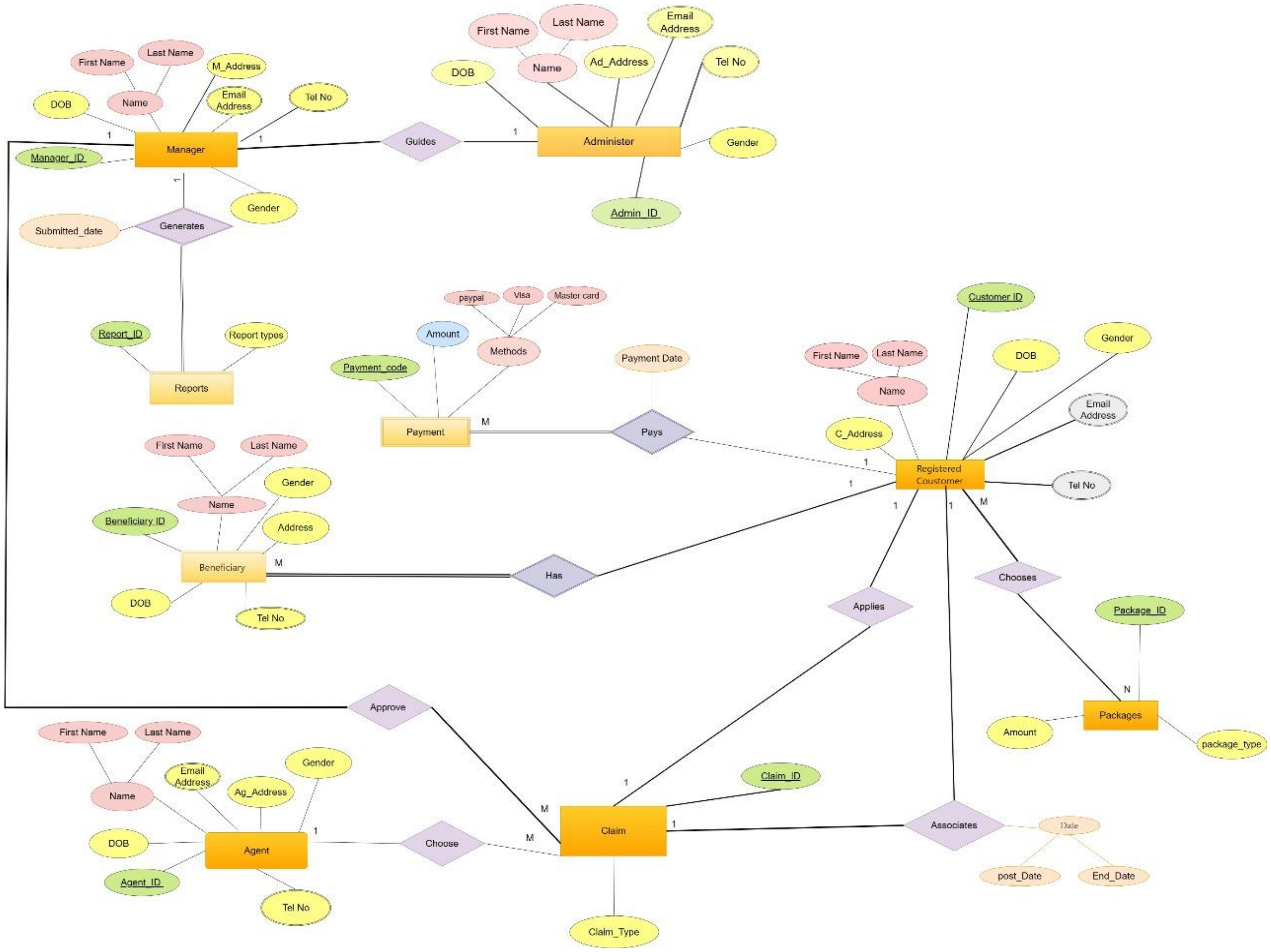
- Transactions
  - Transaction ID (Transaction_ID)

- Transaction Method (Transaction_Method)
- Amount (Amount)
- Transaction Date (Transaction_Date)

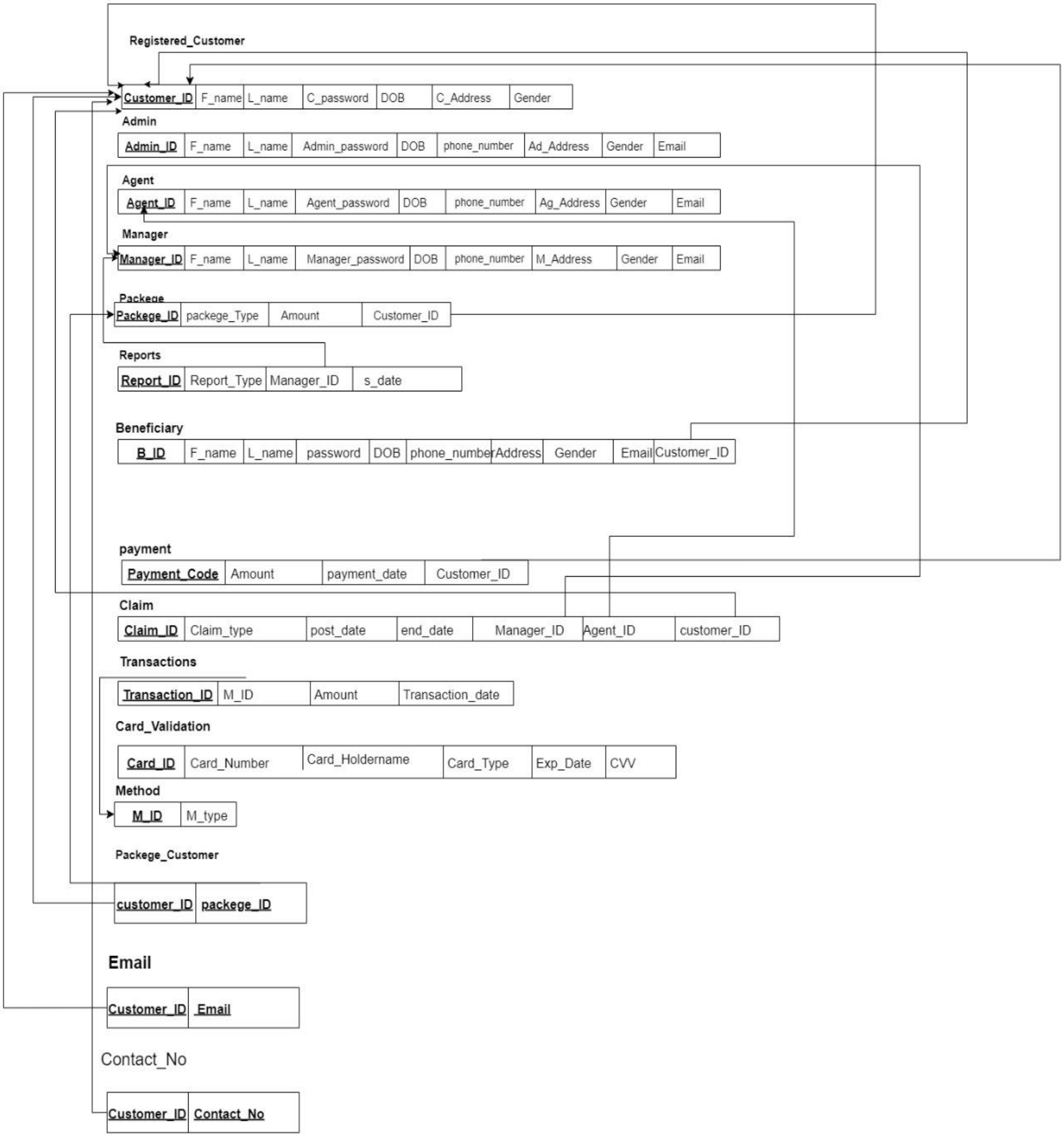
- Card Validation
  - Card ID (Card_ID)
  - Card Number (Card_Number)
  - Card Holder’s name (Card_Holdername)
  - Card Name (Card_Type)
  - Expire Date (Exp_Date)
  - CVV_Number(CVV)

- Payment_Cutomer
  - Customer ID (customer_ID)
  - Package ID (Package_ID)

4. ER Diagram


First Name
Last Name
M_Address
DOB
Name
Email Address
Tel No
1
1
Manager
Manager_ID
1
Gender
Submitted_date
Generates
Guides
1
First Name
Last Name
Email Address
DOB
Name
Ad_Address
Tel No
Administer
Gender
Admin_ID
Report_ID
Report types
Reports
paypal
Visa
Master card
Amount
Methods
Payment_code
Payment
M
Payment Date
Pays
Customer ID
First Name
Last Name
DOB
Gender
Name
Email Address
C_Address
1
Registered Coustomer
1
Tel No
1
1
M
First Name
Last Name
Gender
Name
Beneficiary ID
Address
Beneficiary
M
Has
DOB
Tel No
Chooses
Applies
Package_ID
N
Packages
Amount
package_type
Approve
First Name
Last Name
Gender
Email Address
Ag_Address
Name
1
M
Claim_ID
DOB
Agent
1
Choose
M
Claim
1
Associates
Date
Agent_ID
Tel No
post_Date
End_Date
Claim_Type

5. Relational Schema



Registered_Customer

Customer_ID	F_name	L_name	C_password	DOB	C_Address	Gender

Admin

Admin_ID	F_name	L_name	Admin_password	DOB	phone_number	Ad_Address	Gender	Email

Agent

Agent_ID	F_name	L_name	Agent_password	DOB	phone_number	Ag_Address	Gender	Email

Manager

Manager_ID	F_name	L_name	Manager_password	DOB	phone_number	M_Address	Gender	Email

Packege

Packege_ID	packege_Type	Amount	Customer_ID

Reports

Report_ID	Report_Type	Manager_ID	s_date

Beneficiary

B_ID	F_name	L_name	password	DOB	phone_number	Address	Gender	Email	Customer_ID

payment

Payment_Code	Amount	payment_date	Customer_ID

Claim

Claim_ID	Claim_type	post_date	end_date	Manager_ID	Agent_ID	customer_ID

Transactions

Transaction_ID	M_ID	Amount	Transaction_date

Card_Validation

Card_ID	Card_Number	Card_Holdername	Card_Type	Exp_Date	CVV

Method

M_ID	M_type

Packege_Customer

customer_ID	packege_ID

Email

Customer_ID	Email

Contact_No

Customer_ID	Contact_No

6. SQL Queries

6.1 Data Base Create

```
   /*Table registered Customer*/

CREATE TABLE Registered_Customer
 (
 customer_ID INT PRIMARY KEY IDENTITY (1,1),
 F_name varchar(40) not null,
 L_name varchar(40) not null,
 C_Password varchar(15) not null,
 DOB varchar(20) not null,
 C_Address Varchar (50) not null,
 Gender varchar(10) not null,

  );

     /* Contact TABLE */

 CREATE TABLE Contact_No
 (
   customer_ID INT NOT NULL,
   Contact_No Varchar (15) NOT NULL,
   PRIMARY KEY (customer_ID,Contact_No),

   CONSTRAINT FK__Contact_No_Registered_Customer FOREIGN KEY (customer_ID)
   REFERENCES Registered_Customer(customer_ID)

 );

     /*Email Table */

CREATE TABLE Email
 (
 customer_ID INT NOT NULL,
 Email VARCHAR (40) NOT NULL,
 PRIMARY KEY (customer_ID,Email),

 CONSTRAINT FK_Email_Registered_Customer FOREIGN KEY (customer_ID)
 REFERENCES Registered_Customer(customer_ID)

 );
 /*Admin Table */

CREATE TABLE Administer(

    Admin_ID  INT PRIMARY KEY  IDENTITY (1,1),
    F_name varchar(40) not null,
    L_name varchar(40) not null,
    Admin_Password varchar(15) not null,
    DOB varchar (10) not null,
    phone_number varchar (12) not null,
    Ad_Address Varchar (50) not null,
    Gender varchar(10) not null,
    Email varchar(35) not null,

);

      /* Table Agent */

CREATE TABLE  Agent (
Agent_ID INT PRIMARY KEY IDENTITY (1,1),
F_name varchar (40) not null,
L_name varchar (40) not null,
Agent_password varchar (15) not null,
DOB varchar(20) not null,
phone number varchar (12) not null,
Ag_Address varchar (50) not null,
Gender varchar (10) not null,
Email varchar (35) not null,

);


   /* Table Manager */

CREATE TABLE  Manager (
Manager_ID INT PRIMARY KEY  IDENTITY (1,1),
F_name varchar (40) not null,
L_name varchar (40) not null,
Manager_password varchar (15) not null,
DOB varchar(15) not null,
phone_number varchar (12) not null,
M_Address varchar (50) not null,
Gender varchar (10) not null,
Email varchar (35) not null,
);

    /* Table packege */

CREATE TABLE Packege (
packege_ID INT PRIMARY KEY  IDENTITY(1,1),
packege_Type varchar (30) not null,
Amount varchar(50) not null,
);
```

```
       /* Report Table*/

CREATE TABLE Reports (
Report_ID INT PRIMARY KEY IDENTITY(1,1),
Report_type varchar(20) not null,
Manager_ID INT  not null,
S_date varchar(20) not null,

CONSTRAINT FK_Reports_Manager FOREIGN KEY (Manager_ID)
REFERENCES Manager(Manager_ID)
  );

     /* Beneficiary Table */


   CREATE TABLE Beneficiary (
   B_ID INT PRIMARY KEY IDENTITY(1,1),
   F_name varchar (40) not null,
   L_name varchar (40) not null,
   DOB varchar (15) not null,
   phone_number varchar (12) not null,
   Gender varchar (10) not null,
   Email varchar (35) not null,
   customer_ID INT  not null,

  CONSTRAINT FK_Beneficiary_Registered_Customer FOREIGN KEY (customer_ID)
  REFERENCES Registered_Customer(customer_ID)
  );


    /* Payment Table */

  CREATE TABLE payment (
  payment_code INT PRIMARY KEY IDENTITY(1,1),
  Amount Varchar (20) not null,
  payment_date varchar (30) not null,
  customer_ID INT not null,

 CONSTRAINT FK_payment_Registered_Customer FOREIGN KEY (customer_ID)
 REFERENCES Registered_Customer(customer_ID)

);
```

```sql
/* Claim Table */

 CREATE TABLE Claim(
 Claim_ID  INT PRIMARY KEY IDENTITY(1,1),
 Claim_type varchar (30) not null,
 customer_ID INT not null,
 Post_date Varchar (30) ,
 End_date Varchar (30),
 Manager_ID INT  not null,
 Agent_ID INT not null,

CONSTRAINT FK_Claim_Registered_Customer FOREIGN KEY (customer_ID)
REFERENCES Registered_Customer(customer_ID),
CONSTRAINT FK_Claim_Manager FOREIGN KEY (Manager_ID)
REFERENCES Manager(Manager_ID),
CONSTRAINT FK_Claim_Agent FOREIGN KEY (Agent_ID)
REFERENCES Registered_Customer(customer_ID)

 );

/* Method data */

 CREATE TABLE Method
(
   M_ID INT  PRIMARY KEY IDENTITY(1,1),
   M_type VARCHAR (20) NOT NULL,

);

 /*Transactions Table */

    CREATE TABLE Transactions (
    Transaction_ID INT PRIMARY KEY IDENTITY(1,1),
    Transaction_Method VARCHAR(15) NOT NULL,
    Amount varchar (20) NOT NULL,
    M_ID INT  NOT NUll,
    Transaction_Date VARCHAR(10) NOT NULL,

    CONSTRAINT FK_Transactions_Method FOREIGN KEY (M_ID)
    REFERENCES Method (M_ID)
);

/*Card_validation Table*/

CREATE TABLE Card_Validation (
    Card_ID INT PRIMARY KEY IDENTITY(1,1),
    Card_Number VARCHAR(20) NOT NULL,
    Card_Holdername VARCHAR(50) NOT NULL,
    Card_Type VARCHAR(20) NOT NULL,
    Exp_Date VARCHAR(10) NOT NULL,
    CVV  VARCHAR(5) NOT NULL
);
```

```
/*Packege_Customer Table*/

 CREATE TABLE Payment_Customer
 (
  customer_ID INT NOT NULL,
  packege_ID INT NOT NULL ,
  PRIMARY KEY (customer_ID,packege_ID),

 CONSTRAINT FK_Payment_Customer_Registered_Customer FOREIGN KEY (customer_ID)
 REFERENCES Registered_Customer(customer_ID),
 CONSTRAINT FK_payment_Customer_Packege FOREIGN KEY (packege_ID)
 REFERENCES Packege(packege_ID) ,

                                            );
```

6.2 Data Store in Data Base

```
             /* Enter Registered Customer Data*/

INSERT INTO Registered_Customer( F_name , L_name ,C_Password , DOB  ,C_Address , Gender)
VALUES('Aaquib','Rasmy','110011','2003/02/05','No 10,Rajagiriya,Colombo','Male' ),

  ('Thanis','Ahamad','226508','2000/02/05','No 12,T.B jaya road,colombo 7','Male'),

 ('Thakkif','Illiyas','123456','2002/02/12','442/Prince road,Trinco','Male'),

 ('Tharuki',Nethmini','121250','2001/06/21','446/A/1 Flower road,Rathnapura','Female'),

 ('Rishini','Budhara','112357','2003/04/20','No 50,Vihara road,Bomiriya','Female'),

 ('Kanthi','Daias','126752','1967/05/12','167/B/3 Kingston road,Maharagama','Fmale'),

 ('Sunil','Rathwaththa','112340','1966/12/5','No 20,William road,Colombo 3','Male' ),

 ('Mangala','Dissanayaka','177450','1976/02/3','No 2,New town,Rathnapura','Male' ),

 ('Sushila','Damayanthi','177165','1962/01/25','No 5 ,Kudaligas road,Malabe','Female' ),

 ('Neeta','Rodrigo','124567','1977/11/8','No 12,Pannipitiya road,Mahabage','Female');



             /* Enter Contact Number Table */

INSERT INTO Contact_No(customer_ID,Contact_No)
VALUES ('1','0771245685'),
       ('2','0771650470'),
       ('3','0711245881'),
       ('4','07416204271'),
       ('5','0771651282'),
       ('6','0771256484'),
       ('7','0771280213'),
       ('8','0762114532'),
       ('9','0765362500'),
       ('10','0784567894');
```

```
                 /* Enter Email Table */


INSERT INTO Email(customer_ID,Email)
VALUES ('1','aaquib@gmail.com'),
       ('2','thanis12@gmail.com'),
       ('3','iliyas23@gmail.com'),
       ('4','tharukierathna@gmail.com'),
       ('5','budarapvr@gmail.com'),
       ('6','kanthi@gmail.com'),
       ('7','sunil22@gmail.com'),
       ('8','mangala@gmail.com'),
       ('9','damayanthi99@gmail.com'),
       ('10','neeta@gmail.com');



           /*Enter Admin Data */

INSERT INTO
Administer(F_name,L_name,Admin_Password,DOB,phone_number,Ad_Address,Gender,Email)
VALUES('Janaka','Dissanayaka','134679','1977/12/7','0771397774','22/7 Prince
road,Kaluaggala','Male','janaka5@gmail.com'),

('Wasana','Kalubowila','113469','1987/05/12','0771369781','453/5 Senanayaka
road,Horana','Female','wasanakalubowila@gmail.com'),

('Kumari','Wijesinghe','117290','1890/11/12','0771399942','No 10,Mirigama,colombo
5','Female','wijesinghekumari@gmail.com');



/*Enter Agent Data */

INSERT INTO Agent(F_name,L_name,Agent_password,DOB,phone_number,Ag_Address,Gender,Email)
VALUES ('Malki','Karunasiri','221315','1988/12/10','0712034678','No 20,Dikhena,ja-
ela','Female','Malki@gmail.com'),

('Susila','Kumari','155579','1969/04/29','0712034678','N0 22/8 Vidyaloka mawatha,Colombo
3','Female','Susila@gmail.com'),

('Mohomad','Nazar','449768','1990/07/6','0713944678','No 2,Ramadan
road,Bambalapitiya','Male','mohomad@gmail.com'),


('Kanmani','Ramasami','134685','1991/08/9','07125034678','No 5,Colombo
road,Mahabage','Female','Kanmani@gmail.com'),

('Samantha','Silva','123469','1967/09/6','0712037878','12/8 Colombo
rpad,Kiribathgoda','Female','Samntha@gmail.com');
```

```
             /* Enter Manager Data */

INSERT INTO
Manager(F_name,L_name,Manager_password,DOB,phone_number,M_Address,Gender,Email)
VALUES('Namal','Balasooriya','447589','1988/04/4','0714448758','No 5,Chandrika
road,Malabe','Male','namalbalasooriya77@gmail.com'),

('Kumari','Thennakoon','114577','1988/12/8','0771620427','No
10,Pannipitiya,Maharagama','Female','kumari200@gmail.com');



           /*Enter Packge Type */

INSERT INTO Packege(packege_Type,Amount)
VALUES('Protection','100000'),
      ('Helth Care','250000'),
      ('Protection','100000'),
      ('Retirement','220000'),
      ('Retirement','220000'),
      ('Helth Care','250000'),
      ('Protection','120000');



/* Enter data of Reports  */

INSERT INTO Reports(Report_type , Manager_ID , S_date)
VALUES('Anual','1','2024/05/2'),
      ('Monthly','2','2024/05/3'),
      ('benifisary','1','2024/05/3'),
      ('monthly','2','2024/03/5'),
      ('monthly','1','2024/03/6'),
      ('monthly','1','2024/02/5'),
      ('monthly','2','2024/02/6');



/* Enter Benifisary data  */

INSERT INTO Beneficiary( F_name, L_name, DOB,phone_number,Gender,Email,customer_ID)

VALUES ('Aaquib','Rasmy','2003/02/05','0771245685','Male','aaquib@gmail.com','1'),

('Rishini','Budhara','2003/04/20','0771651282','Female','budarapvr@gmail.com','5'),

('Sushila','Damayanthi','1962/01/25','0765362500','Female','damayanthi99@gmail.com','9'),

('Neeta','Rodrigo','1977/11/8','0784567894', 'Female','neeta@gmail.com','10');
```

```sql
          /*Enter payment details*/

INSERT INTO payment( Amount , payment_date , customer_ID)
VALUES('100000','2024/02/05','1'),
      ('250000','2024/02/05','2'),
      ('100000','2024/05/28','2'),
      ('220000','2024/03/1','4'),
      ('220000','2024/04/2','5' ),
      ('250000','2024/04/2','3'),
      ('120000','2024/06/21','1');



         /*Enter Claim details*/

INSERT INTO Claim( Claim_type, customer_ID ,Post_date,End_date,Manager_ID,Agent_ID)
VALUES('Atype',1,'2024-05-03','2026-06-04',1,1),
      ('Btype',2,'2024-02-15','2026-05-07',1,2),
      ('Ctype',3,'2024-03-11','2026-05-07',1,2),
      ('Dtype',4,'2024-04-03','2026-05-07',1,2);



        /* Enter Method Table */

INSERT INTO Method(M_type )
VALUES('Cardpay'),
      ('Cardpay'),
      ('Cardpay'),
      ('Cardpay'),
      ('Cardpay');


       /* Enter Transactions Table */

INSERT INTO Transactions(Transaction_Method,Amount,M_ID,Transaction_date)
VALUES ('Cardpay','2000','1','24/02'),
       ('Cardpay','3000','2','12/04'),
       ('Cardpay','4000','1','23/04'),
       ('Cardpay','5000','2','09/04');



        /*Enter card validation*/

INSERT INTO Card_Validation(Card_Number,Card_Holdername,Card_type,Exp_Date,CVV )
VALUES('234','Namal','visa','23/25','2345'),
      ('234','Kamal','Master','23/25','2345'),
      ('234','Wimal','Paypal','23/25','2345'),
      ('234','Anil','visa','23/25','2345'),
      ('294','Sandun','Master','23/25','2845');
```

```
/*Packege_Customer Table*/

         INSERT INTO Payment_Customer(Customer_ID,packege_ID)
         VALUES ('1','1'),
                ('2','2'),
                ('3','3'),
                ('4','4'),
                ('5','5');
```

7. Performance Requirement

- The system must be available 24 hours every day and without any inconvenience.
- The system couldn’t limit the user's time using the system and the user can log in to his account anytime.
- Users can use any device to access the website.
- Loading time must be radiused and must load in a few seconds.
- Registered Customers can view package details.
- User can edit, delete, and renew their details at any time.
- Admin can add, edit, and remove feedback.
- The developer can add new features and update the systems.
- The developer can be able to fix errors.
- Managers can approve and delete claims.
- Customers can request to change the beneficiary at any time.

8. Security Requirements

- Users' private details should be protected before being sent to the database.
- There’s only one email for one account.
- Illegal users should not be allowed to gain access to restricted functionality.
- The database server must be managed using a backup server.
- Only the admin can access and modify the user’s data.
- Accounts for users need a strong password containing uppercase, lowercase, numerals, and special characters.