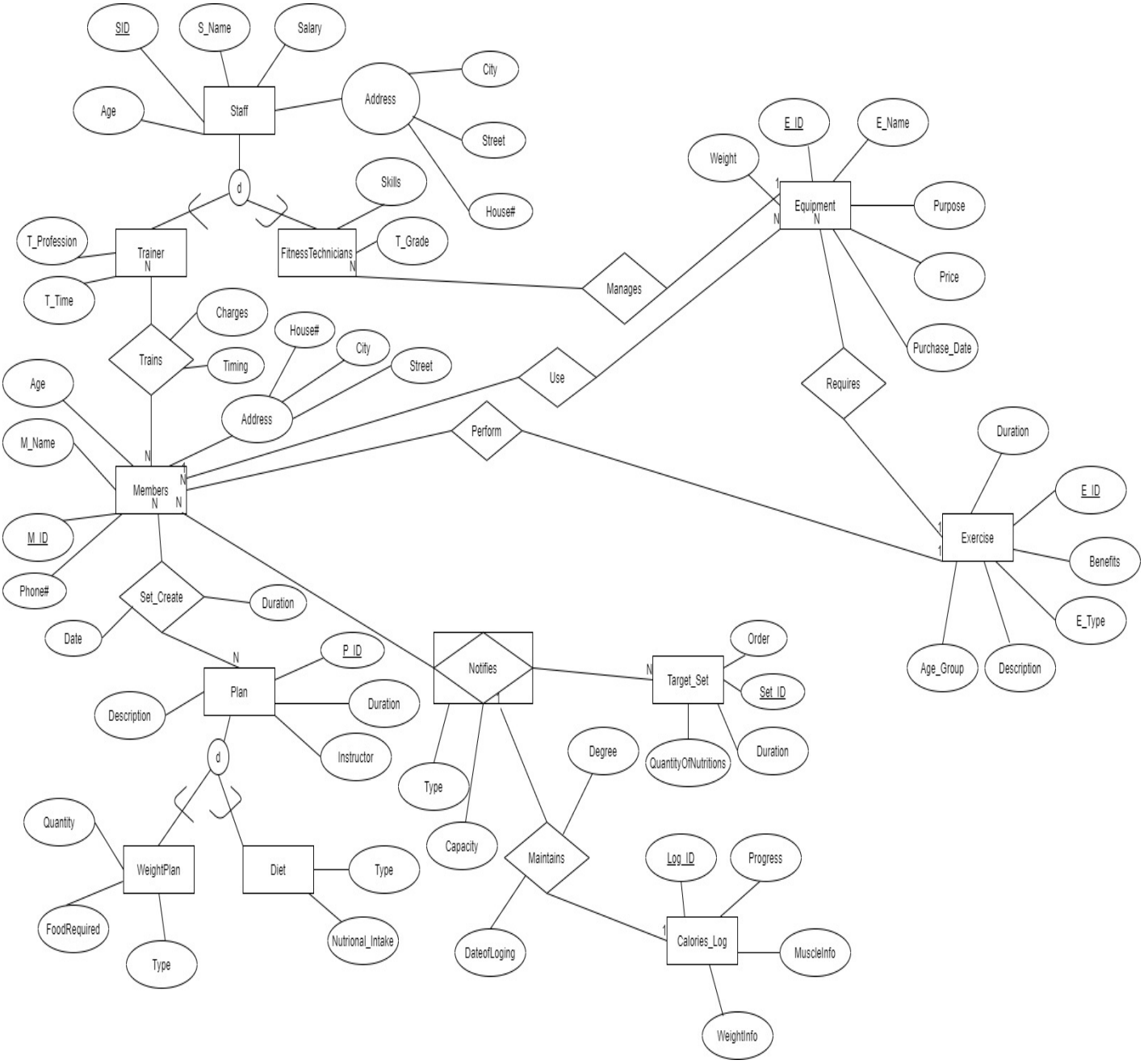
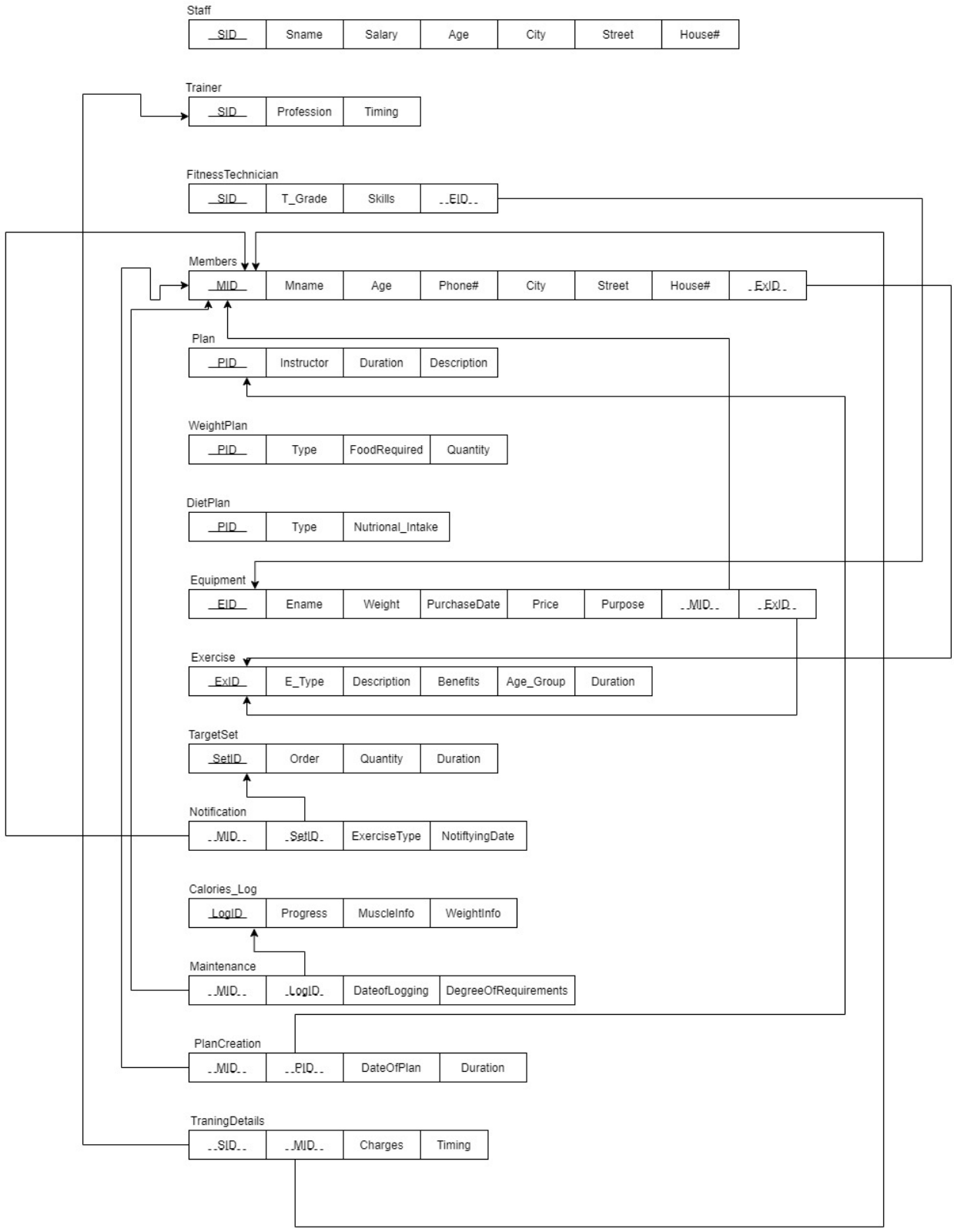
FIT-ME Application Database Development Report

Abdul Manan (19I-0500).

1). ERD


SID
S_Name
Salary
Age
Staff
Address
City
Street
House#
d
Skills
T_Profession
Trainer
N
FitnessTechnicians
N
T_Grade
T_Time
Manages
E_ID
E_Name
Weight
1
Equipment
N
N
Purpose
Price
Purchase_Date
Charges
Trains
Timing
House#
City
Street
Use
Requires
Age
Address
M_Name
Perform
N
Members
N
N
1
N
Duration
E_ID
Benefits
E_Type
Exercise
1
1
M_ID
Phone#
Set_Create
Duration
Date
N
P_ID
Plan
Description
Duration
Instructor
Notifies
N
Target_Set
Order
Set_ID
Duration
QuantityOfNutritions
Degree
Type
Capacity
Maintains
1
Log_ID
Progress
Calories_Log
MuscleInfo
WeightInfo
DateofLoging
Quantity
WeightPlan
FoodRequired
Type
Diet
Type
Nutrional_Intake
Age_Group
Description

2). Relational Schema



3). Table Description

a). Staff:

This table is basically the parent class. As, there are more than one staff members in the Gym center. So, that's why this table is created.

i). Trainer:

This table is basically the sub-class of table Staff. As, the purpose of trainers in the Gym is to train the members or user according to their particular exercise and they are also part of staff members, similarly, as in this covid situation, trainer can train the members online. That's why there table is included.

ii). Fitness Technician:

As, in Gym centers there also some staff members who are responsible for managing overall fitness and Equipment's things. So, this table is created to manage equipment and members related issues.

b). Plan:

Another important table is of Plan. Where each member creates their plans in order to fulfill their requirements.

i). WeightPlan:

Now, Plan class or table is further divided into weight plan sub-class. In which user creates their plan of weight losing/Gaining and then according to this plan they maintain their food and exercise on daily basis.

ii) DietPlan:

This table is also a sub-class of Plan, in which user creates their plan of maintaining diet by taking nutritional intakes on daily basis. And, each Diet plan have some instructor, for instructing the user their requirements on daily basis.

c). Members:

Now, the main core table in this application is of Members. Which are basically the real evaluators of the services provided by fit-me application. Members are the one who create plans, perform exercises, and maintain their log profile on daily bases, in order to see to what extent they were able to meet the requirements.

d). Exercise:

This is also an important table which basically tells which types of exercises to perform by which member. And, the purpose or description of each exercise. Similarly, which type of exercise requires which equipment are all fall under this table.

e). Equipment

As, Exercise requires equipment. So, that's why this kind of table was also important to manage in this case. As, each type of equipment require exercise to perform. And, each equipment have certain price and the date in which it was purchased.

f). TargetSet:

This is basically the table in which user set their Targets for exercise and their duration of that Target in which they want to complete it. And, also the Quantity of each target set.

g). Notification:

This table is basically the Add-on table as Target-set and Member table were having Many-Many relationship so, in order to manage it, Notification table is being created, which notifies the user of exercise type they have to perform in the particular Date.

h). Calories_Log:

This table basically tells user's progress till now. Whenever user want to see their progress related to muscles and weight information, they log into this table to find out this particular information.

i). Maintenance:

This is another Add-on table, as Members and Calories Log were having many-many relationship. So, in order to manage it, Maintenance table is created which takes both of the table's primary key and make it foreign key in its table. Also, it tells the user of their Log Date and Percentage of Degree requirements managed by user up till now.

j). Plan_Creation:

Another Add-on table to manage the relationship between member and Plan having N-N relationship. This table tells the information related to Date of creating the plan to user and similarly, Duration of a particular plan in which it should be completed.

h). Training_Details:

This is the table for managing N-N relationship between training and Member table. This table which basically tells the timing and charges of each Trainer in the institution. So that they can see the trainer best suitable for them for a particular exercise.

4). DDL

```
create Table Staff(
     St_ID      Number(8),
     Sname      varchar2(50),
     Salary     Number(25),
     Age        Number(3),
     City       varchar2(40),
     Street     Number(15),
     HouseNo    varchar2(20),
      constraints  PK_Staff PRIMARY KEY(St_ID)
      );

 create Table Trainer(
     St_ID      Number(8),
     Sname      varchar2(50),
     Salary     Number(25),
     Age        Number(3),
     City       varchar2(40),
     Street     Number(15),
     HouseNo    varchar2(20),
     Profession     varchar2(40),
     Timing         varchar2(45),
     constraints  PK_Trainer PRIMARY KEY(St_ID)
 );

 create Table Plan(

     PID     Number(8),
     Instrcutor   varchar2(50),
     Duration     varchar2(20),
     Description  varchar2(100),
     constraints  PK_Plan PRIMARY KEY(PID)
     );
```

```
create Table WeightPlan(

     PID     Number(8),
     Instrcutor   varchar2(50),
     Duration     varchar2(20),
     Description  varchar2(100),
     Type         varchar2(40),
     FoodRequired varchar2(50),
     Quantity     varchar2(15),
     constraints  PK_Weight PRIMARY KEY(PID)
     );

create Table DietPlan(

     PID     Number(8),
     Instrcutor   varchar2(50),
     Duration     varchar2(20),
     Description  varchar2(100),
     Type         varchar2(40),
     NutritionalIntake varchar2(50),
     constraints  PK_Diet PRIMARY KEY(PID)
     );

create Table Exercise(

      ExID     Number(8),
      E_Type   varchar2(40),
      Description   varchar2(100),
      Benefits      varchar2(50),
      Age_Group     varchar2(15),
      DurationE      varchar2(15),
      constraints  PK_Exercise PRIMARY KEY(ExID)
      );
```

```
create Table Members(

       MID       Number(8),
       Mname     varchar2(40),
       Age       Number(3),
       PhoneNo   int,
       City       varchar2(40),
       Street     Number(15),
       HouseNo    varchar2(20),
       ExID       Number(8),
      constraints  PK_Member PRIMARY KEY(MID),
      constraints  FK_Member Foreign KEY(ExID) REFERENCES Exercise(ExID)
      );

create Table Equipment(

     EID      Number(8),
     Ename    varchar2(35),
     Weight   varchar2(10),
     PurchaseDate    Date,
     Price          Number(25),
     Purpose        varchar2(35),
     MID            Number(8),
     ExID           Number(8),
     constraints  PK_Equipment PRIMARY KEY(EID),
     constraints  FK_Equipment Foreign KEY(ExID) REFERENCES Exercise(ExID),
     constraints  FK_EquiMember FOREIGN KEY(MID) REFERENCES Members(MID)
     );
```

```
create Table FitnessTechnician(
    St_ID     Number(8),
    Sname     varchar2(50),
    Salary    Number(25),
    Age       Number(3),
    City      varchar2(40),
    Street    Number(15),
    HouseNo   varchar2(20),
    T_Grade    varchar2(5),
    Skills        varchar2(45),
    EID         Number(8),
    constraints  PK_Technique PRIMARY KEY(St_ID),
     constraints  FK_Technique FOREIGN KEY(EID) REFERENCES Equipment(EID)

);

Create Table TargetSet(

    SetID     Number(8),
    Order1     varchar2(15),
    Quantity   VARCHAR2(10),
    Duration    varchar2(15),
     constraints  PK_Set PRIMARY KEY(SetID)
    );

 create Table Notification(

    MID      Number(8),
    SetID    Number(8),
    ExerciseType   varchar2(30),
    NotifyingDate       Date,
   constraints  FK_Notification FOREIGN KEY(MID) REFERENCES Members(MID),
   constraints  FK_Notification2 FOREIGN KEY(SetID) REFERENCES TargetSet(SetID)
   );
```

```
create Table CaloriesLog(

    LogID       NUMBER(8),
    Progress    varchar2(30),
    MuscleInfo    varchar2(40),
    WeightInfo    varchar2(40),
    constraints   PK_LOG    PRIMARY KEY(LogID)
);

create Table Maintenance (

     MID     NUMBER(8),
     LogID   NUMBER(8),
     DateOFLoging    Date,
     DegreeOFRequirements     varchar2(30),
      constraints   FK_Maintain1 FOREIGN KEY(MID) REFERENCES Members(MID),
      constraints   FK_Maintain2 FOREIGN KEY(LogID) REFERENCES CaloriesLOG(LogID)

 );

 create Table PlanCreation(

       MID     NUMBER(8),
       PID     NUMBER(8),
       DateOFPlan     Date,
       Duration       varchar2(15),
        constraints   FK_PlanCreate FOREIGN KEY(MID) REFERENCES Members(MID),
       constraints   FK_PlanCreate2 FOREIGN KEY(PID) REFERENCES Plan(PID)
       );
```

5). DML

```
insert into staff
values(23,'Hamza',20000,26,'Islamabad',11,'F-10');

/*insert into Trainer
values(24,'Kashif',15000,24,'Rawalpindi',20,'DhokeSyedan','CheerLeader','25 minutes');*/

 insert into trainer
  values(18,'Kashif',15000,24,'Rawalpindi',20,'Dhoke','Cheerleader','25minutes');

   insert into trainer
   values(21,'ALi',16000,26,'Islamabad',17,'F-11','DrillSergeant','30minutes');

    insert into trainer
    values(22,'Hamza',17000,28,'Islamabad',21,'G-11','CardioQueen','27minutes');

     insert into plan
    values(341,'Jameel','3 months','Facilities and Equipments');

 insert into WeightPlan
    values(341,'Jameel','3 months','Facilities and Equipments','Weight Gaining Plan','Milk, Rice, RedMeat','2-3 times a day');

     insert into WeightPlan
    values(345,'Qasim','6 months','Physical Fitness','Weight Loss Plan','Dark Chocolate,Beans','3-4 times a day');


   insert into WeightPlan
    values(350,'Usama','5 months','Muscle Building','Weight Gain Plan','Fish, Protein Supplements','5-6 times a day');

     insert into DietPlan
    values(233,'Khalid','3 months','A low-fat diet is one that restricts fat, and often saturated fat','Low-fat diet','vagetables, fruits, grains');


     insert into DietPlan
    values(244,'Asghar','2 months','DASH diet is a lower sodium version of the diet','DASH-Diet','nuts, seeds, beans 4-5 times daily');
```

```
 insert into Exercise
values(121,'Strength Training','Strength training exercises work your muscles by applying a resistance','Protect Bone Health and Muscle mass','25+','5-10 s

 insert into Exercise
values(125,'Aerobic Exercise','Aerobic exercise is any type of cardiovascular conditioning','Improves cardiovascular conditioning.','20+','5-10 minutes');

 insert into Exercise
values(128,'Stretching','Stretching is a form of physical exercise in which a specific muscle or tendon is flexed ','Increases blood flow to your muscles',


  insert into Members
values(1832, 'Qumail', 20, 02341, 'Wah',12,'4th',128);

 insert into Members
values(1751, 'Saim', 23, 08712, 'Taxila',8,'7th',121);

insert into Members
values(1743, 'Saad', 28, 03421, 'Rawalpindi',5,'10th',121);

insert into Members
values(1643, 'Daniyal', 25, 04512, 'Islamabad',15,'16th',125);


insert into Equipment
values(451,'Treadmill','25 kg','12-Mar-2020',35000,'Allows user to walk or run',1743,128);

 insert into Equipment
values(444,'Stationary Bicycle','30 kg','23-June-2020',45000,'Build Strength in legs',1751,125);

insert into Equipment
values(433,'Ellipticals','23 kg','05-November-2020',33000,'To get a good aerobic workout',1643,125);

insert into Equipment
values(420,'Barbell Set','28 kg','27-December-2020',22000,'Used for free weight training',1743,121);
```

```
 insert into FitnessTechnician
values(24,'Usman',10000,31,'Islamabad',23,'20th','9th','Membership Sales',420);

insert into FitnessTechnician
 values(29,'Mubashir',13000,27,'Tarnol',27,'31st','7th','Fitness Environment',420);

 insert into FitnessTechnician
 values(34,'Qasim',11000,32,'Rawalpindi',33,'9th','8th','Customer Service',433);

 insert into FitnessTechnician
 values(39,'Sajjad',16000,36,'Dhoke Syedan',35,'34th','11th','Manages Stationary Bicycles',444);



  insert into TargetSet
 values(5412,'2nd','3','4 months');

 insert into TargetSet
 values(6513,'3rd','5','3 months');

 insert into TargetSet
 values(7231,'4th','7','2 months');


  insert into Notification
 values(1832,6513,'Strength Training','31-Dec-2020');

  insert into Notification
 values(1743,7231,'Aerobic Exercise','23-September-2020');

 insert into Notification
 values(1751,5412,'Stretching','15-Jan-2021');
```

```
    insert into CaloriesLog
values(1241,'30%','227g Increased','5 kg Decreased');

    insert into CaloriesLog
values(1236,'60% Completed','250g Increased','3 kg Increased');

    insert into CaloriesLog
values(1222,'90% Completed','270g Increased','7 kg Decreased');



     insert into Maintenance
values(1751,1241,'17-Feb-2021','Requirements meet 5% till now');

     insert into Maintenance
values(1832,1236,'23-Mar-2021','Requirements meet 15% till now');

      insert into Maintenance
values(1643,1222,'5-April-2021','Requirements meet 13% till now');

      insert into Maintenance
values(1743,1236,'28-April-2021','Requirements meet 20% till now');


       insert into PlanCreation
values(1743,341,'13-Jan-2021','5 months');


       insert into TrainingDetails
values(21,1743, 500, '5-5:30 Pm Daily');

        insert into TrainingDetails
values(18,1751, 800, '4:30-5:30 Pm Daily');
```

6). PL/SQL

Procedures:

```
/----------------------------PL/SQL-----------------------------------------------------/

create PROCEDURE PROC_TARGETSET(p_Set IN VARCHAR2) IS
    v_DURATION      TARGETSET.ORDER1%TYPE;
    BEGIN
         select duration
         into v_DURATION
         from TargetSet
         where SetID=p_Set;
         DBMS_OUTPUT.PUT_LINE('Duration fot this TargetSet Plan is'||v_DURATION);
    EXCEPTION
        WHEN OTHERS THEN
        DBMS_OUTPUT.PUT_LINE('ERROR...');
   END;
   /


 create PROCEDURE PROC_Req(p_MID IN VARCHAR2) IS
    v_Req   Maintenance.DegreeOFREQUIREMENTS%TYPE;
    BEGIN
              select degreeofrequirements  d into v_Req
              from maintenance d JOIN members m
              on d.MID=m.MID
              where d.MID=p_MID;
              DBMS_OUTPUT.PUT_LINE('Degree Requirements for this Member='||v_Req);
     EXCEPTION
               WHEN OTHERS THEN
               DBMS_OUTPUT.PUT_LINE('ERROR...');
   END;
   /
```

Triggers:

```
CREATE TRIGGER tr_Equip
 2  After insert on Equipment
 3  for each row
 4  enable
 5  Declare
 6  v_user varchar2(15);
 7  BEGIN
 8          select user into v_user from dual;
 9          DBMS_OUTPUT.PUT_LINE('One Equipment Inserted'||v_user);
10  END;
11  /


create TRIGGER tr_Notification
   After update on Notification
   for each row
   enable
   declare
   v_user Varchar2(15);
   Begin
        select user into v_user from dual;
        DBMS_OUTPUT.PUT_LINE('New Exercise: '||:NEW.ExerciseType||'Old Exercise: '||:OLD.ExerciseType);
  End;
  /
```

Queries with Outputs

1)

```
SQL> select Sname
  2  from trainer
  3  where st_ID=(select SID_T
  4               from TrainingDetails
  5               where MID=1832);

SNAME
--------------------------------------------------
Hamza
```

Description: This Query basically returns trainer name who is training the employee “1832”. As, these both were in different tables so, Nested Query is used in this case.

2).

```
SQL> select Description
  2  from Exercise
  3  where ExID=(select ExID
  4              from Equipment
  5              where Ename='Stationary Bicycle');

DESCRIPTION
--------------------------------------------------------------------------------
Aerobic exercise is any type of cardiovascular conditioning
```

Description: This Query returns the Description of exercise that can be performed from the Equipment “Stationary Bicycle” by matching both foreign keys on one anothers table.

3).

```
SQL> select DegreeOfRequirements
  2  from Maintenance
  3  where MID=(select MID
  4             from Notification
  5             where SetID=(select setID
  6                          from TargetSet
  7                          where Duration='3 months')
  8               );

DEGREEOFREQUIREMENTS
------------------------------
Requirements meet 15% till now
```

Description: This Query is basically a triple nested Query that returns the Degree of Requirements meet till now of a particular Member who have chosen the Duration plan of 3 months. So, three tables are being linked here with the help of foreign keys and relationships.

4).

```
SQL> select E_TYPE E, MAX(PRICE) P
  2  from Exercise E JOIN Equipment P
  3  ON E.ExID=P.ExID
  4  where rownum<=2
  5  Group by E_Type
  6  Order by Max(Price) DESC;

E                                          P
---------------------------------------- ----------
Aerobic Exercise                          45000
Strength Training                         22000
```

Description:

This Query returns top 2 Exercises done with the highest price of Equipments by joining exercise and Equipment table together.

5).

```
SQL> select MuscleInfo M
  2  from CaloriesLog M JOIN Maintenance mt
  3  ON M.LogID=mt.LogID
  4  where mt.LogID=1222;

M
----------------------------------------
270g Increased
```

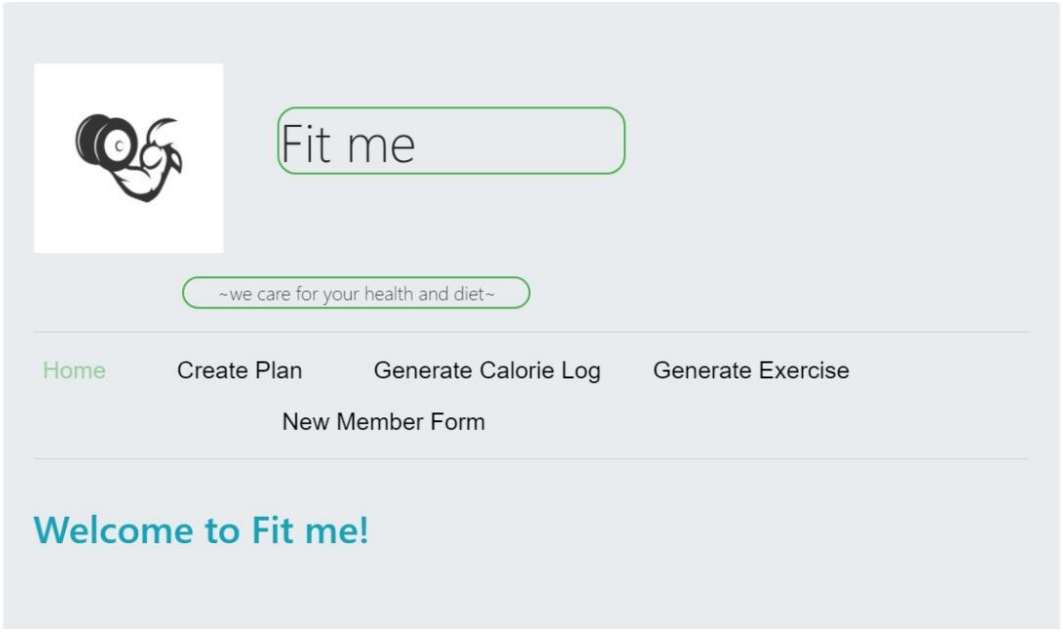
Description: This Query returns Muscles Information to the user 1222 or in other words progress of muscles till now by joining Log and Maintenance table together.

6).

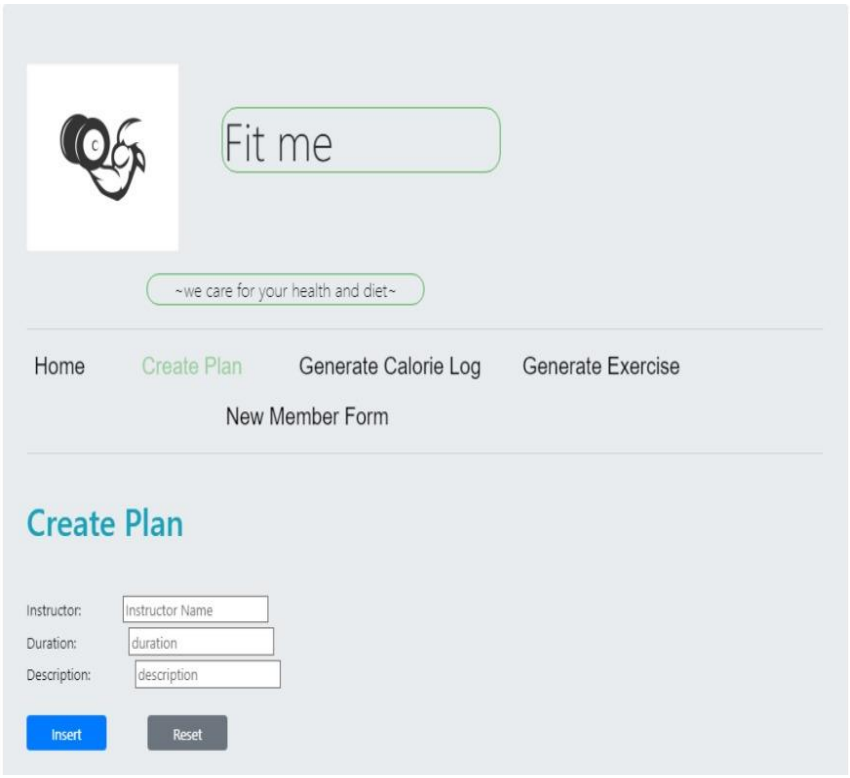
```
SQL> select MIN(Charges) "MINIMUM CHARGES FOR TRAINER"
  2  from Trainer T JOIN TrainingDetails D
  3  on T.st_ID=D.SID_T;

MINIMUM CHARGES FOR TRAINER
---------------------------
                        500
```

Description: This Query returns Minimum charges required for a particular trainer in the gym, by joining Trainer and Trainning_Details (Add On table) together.

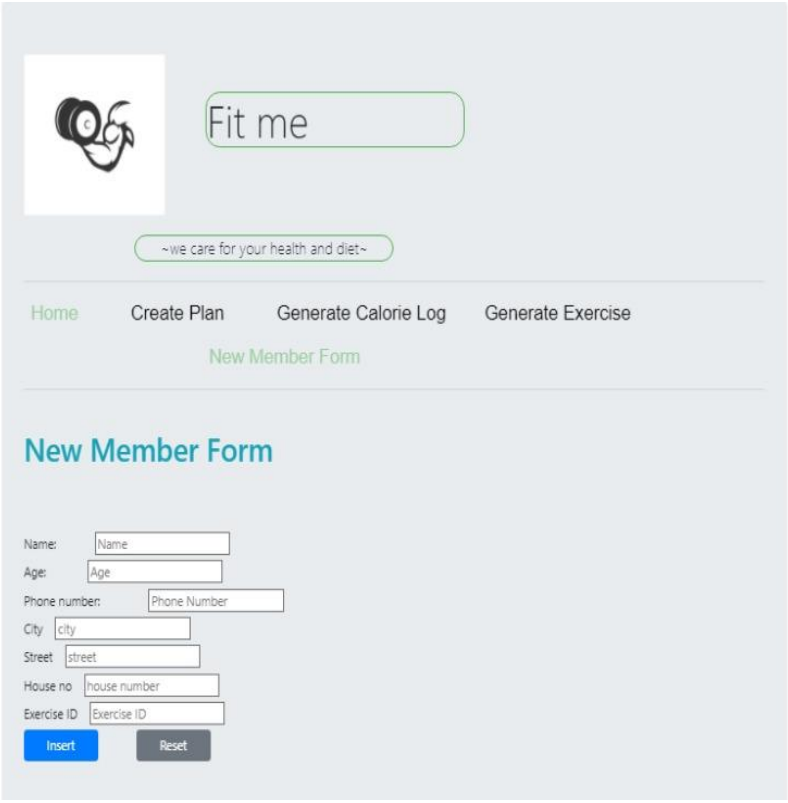
6). Front End Development



Plan Creation Form:



Member Entry Form



Report Exercise: