NeoNatal 36

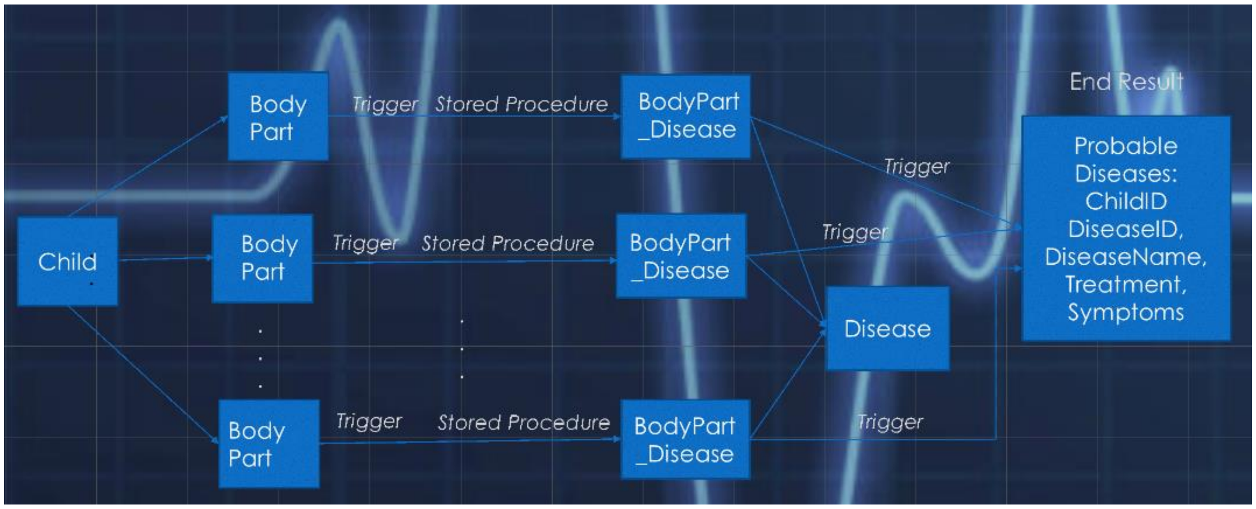
1 Abstract:

1.1 Overview of the Project

In medical contexts, newborn or neonate is an infant who is only hours, days, or up to a few weeks old. So, our project, NeoNatal-36 is all about health care for neonates. The first 36 hours for them is the most critical time, as they are prone to many fatal diseases like Bronchitis, Pneumonia, and Pyloric Stenosis etc. Our project will help the pediatric or NICU department in the hospitals and clinics to overcome the possible diseases and conditions of a neonate. The procedure involves continuous inputs of vital signs on an hourly or regular intervals and comparing it with the ideal vital signs. Thus pro-active actions can save the neonates by providing them with the correct treatment on time.

1.2 Proposed System

Our database will filter the probable diseases by judging from various symptoms like heart rate, respiratory rate, skin color, blood levels, organ conditions etc. On selecting the correct disease it will return the diagnosis, causes, and treatments of it. Thus, the audience for this database would be the nurses, doctors and hospital management. As the time passes by, we can figure out how much the baby is out of danger. Hopefully we wish to revolutionize the system for the pediatric and NICU departments of the hospitals with NeoNatal-36.



2 Project Objective

The objective of this project was to create a one-stop database to track medical conditions of a child for the first 36 hours from the time of birth. Real time data feed was recorded in the database which would help diagnose the most probable disease a child is prone to. It also consists of records of symptoms, causes, treatments and list of diseases which an infant is susceptible to immediately after birth. The database also maintains and manages information about premature babies.

3 Scope

The scope of this project is to record and track data which will be fed in the database to filter the probable diseases a child could be prone to. Real time data feed of vital signs has been recorded in an hourly and secondly basis. Vital signs such as respiratory rate, heart rate, temperature and blood pressure are tracked every second. Blood tests and urine tests of an infant were performed every hour. The data obtained was then tracked and fed in the database which was compared to the normal range of vital signs for an infant. This helped the doctors and nurses in NICU to find the most likely disease the child suffers from. Stored procedures and triggers are used to inform the doctors in NICU in case there is an emergency or there is a drop or spike in the vital signs drastically. After the completion of the 36 hours period, an event scheduler is called which deletes the data from probable diseases table and stores this data in an archival table.

4 Roles and Responsibilities

4.1 User Roles:

1. Cardiologist
   Roles – Select on View_Cardiologist
2. Gastroenterologist
   Role – Select on View_Gastro
3. General Doctor
   Role – Modify on Diseases table
4. NicuAdmin
   Role – All privileges on all the tables in the schema
5. Nurse
   Role – Select on organ tables for the baby.
6. Father/Mother

Role – Select on Doctor, Child and Hospital tables.

7. Ophthalmologist
   Role - Select on View_ Ophthalmologist
8. Hematologist
   Role - Select on View_ Hematologist
9. Pulmonologist
   Role - Select on View_Pulmonologist
10. Neurologist
    Role - Select on View_Neurologist
11. Urologist
    Role - Select on View_Urologist

Users	**Privileges**	**Tables/Views**
Admin	ALL	All tables
Nurse	Select	Vital Signs Tables-(Cardiac, Lungs, Skin, etc.)
Father	Select	Doctor, Child and Hospital
Mother	Select	Doctor, Child and Hospital
Cardiologist	Select	Cardiac View
Gastro	Select	Gastro view
Hematologist	Select	Hematologist view
Ophthalmologist	Select	Ophthalmologist view
Pulmonologist	Select	Pulmonologist view
General	Select and Modify	Diseases table

5 Assumptions and Constraints:

For our project, we need to monitor the vital signs of an infant periodically and feed the data to our database which will then compare it with the normal range of vital signs for an infant. We achieved this by making use of a java program which we connected to our database using JDBC/ ODBC connection. This program generates live data feed and then this data is stored in the database. The following machines are used to monitor these vital signs in NICU.

- Bililights- These are bright blue lights that are placed over the infant's incubator to monitor jaundice.

- Blood pressure monitor- This is a machine that is connected to a small blood pressure cuff which is wrapped around the infant's arm or leg. This machine periodically measures the infant’s blood pressure and displays it on a monitor.

- Cardiopulmonary monitor- This machine tracks the infant’s heart rate and breathing rate. Small adhesive monitoring pads are placed on the infant’s chest. The machine triggers an alarm if the heart rate or breathing rate becomes too slow or too fast.

- Central line- This is a small plastic tube that is placed in a large vessel. Doctors administer medicine and fluids through this tube. They can draw blood from this tube as well.

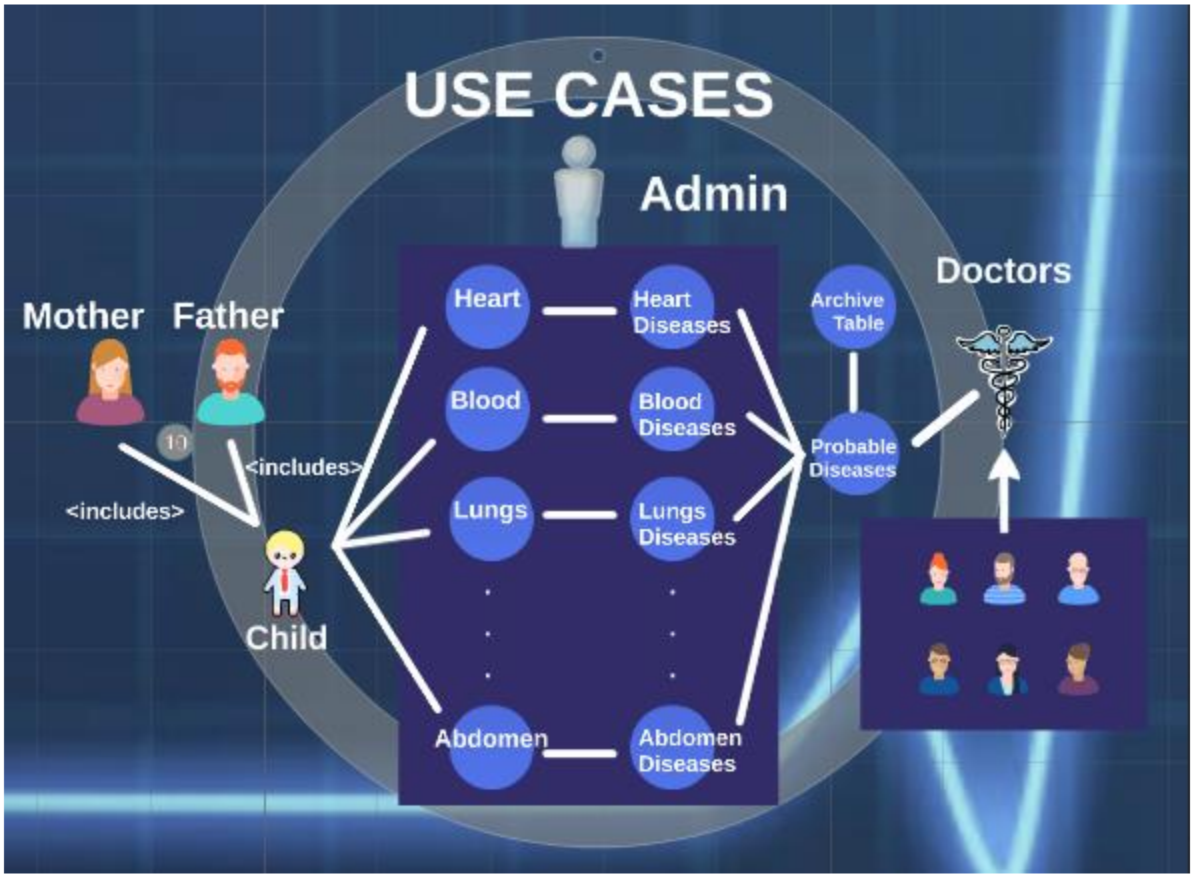
- C-PAP (continuous positive airway pressure) - This machine is used to deliver air to the baby’s lung to help him/her breathe.

- Incubator- This is a clear plastic bed, which regulates the temperature.

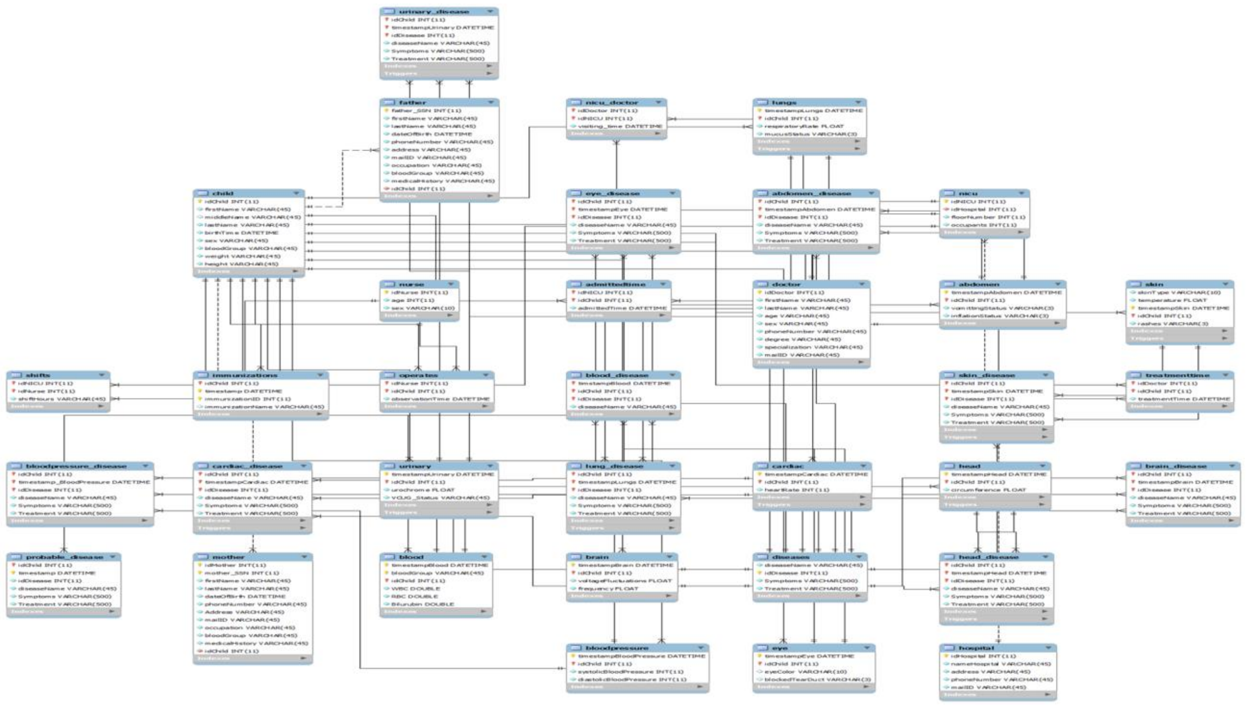
- Pulse oximeter- This is a small device that is taped to the infant’s leg or arm. This machine makes use of light sensor to determine if the baby has enough oxygen in his/her blood.

- Umbilical catheter- An infant’s umbilical cord has two arteries and a vein. A catheter is inserted into one of these vessels and threaded to the aorta, through which the doctors and nurses can painlessly draw blood. They can also administer fluid, nutrition, blood and medication through this catheter.

6 UML Diagram



7 EER Diagram



7.1 Organ Tables:

1. Our organ tables are Head, Eyes, Lungs, Cardiac, Abdomen, Blood, Blood Pressure, Skin and Brain

2. These body organs have various vital signs.

3. For Example, in case of Blood body organ, we measure the baby's Blood Group, WBC, RBC, Bilirubin.

7.2 Organ Disease Tables:

1. Our Organ Disease Tables are Head_Disease, Eyes_Disease, Lungs_Disease, Cardiac_Disease, Abdomen_Disease, Blood_Disease, Blood Pressure_Disease, Skin_Disease and Brain_Disease.

2. In these tables we store ID_Disease, ID_Child, timestamp_vitalSign, DiseaseName, Symptoms and Treatment.

7.3 Probable Disease Table:

1. When a particular vital sign is above an acceptable range, we use stored procedures and triggers to find out the cause of a disease.

2. This probable disease is stored in the Probable_Disease Table.

3. Columns in the Probable_Disease table - ID_Child, ID_Disease, timestamp, diseaseName, Symptoms and Treatment.

7.4 Archival Table

1. After the baby has been monitored for 36 hours, the dedicated data is then transferred into the archival table, by invoking an event scheduler.

2. Columns in the Archival table - ID_Child, ID_Disease, timestamp, diseaseName, Symptoms and Treatment

8. Stored Procedures

1. When data is inserted into the body organ tables, a probable disease is identified in a trigger and then its respective stored procedure is called from the trigger.

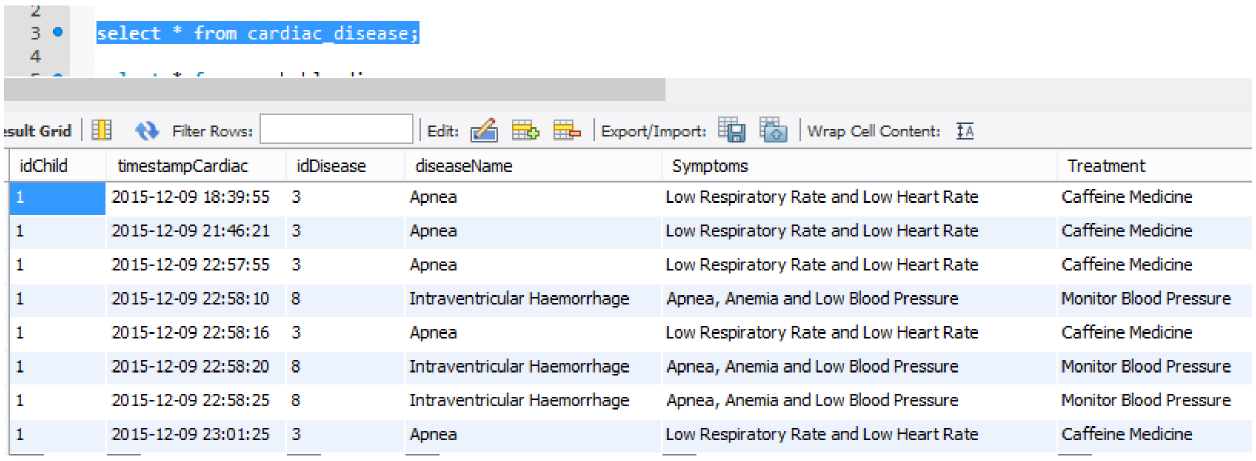
2. Various Stored Procedures created – Anemia_Procedure, Apnea_Procedure, Bronchitis_Procedure, Epilepsy_Procedure, Hemolytic_Procedure, Hydrocephalus_Procedure, Intraventricular_Haemorrhage_Procedure, Necrotizing_Enterocolitis_Procedure, Patentductusarteriosis_Procedure and Pyloric_Stenosis_Procedure.

8.1 Example of the Stored Procedure

```
-- --------------------------------------------------------------------------------
-- Routine DDL
-- Note: comments before and after the routine body will not be stored by the server
-- --------------------------------------------------------------------------------
DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `anemia_procedure`()
BEGIN

#DECLARE CONTINUE HANDLER FOR 1062 SELECT 'Duplicate keys error encountered';
```

```
INSERT into
cardiac_disease(idChild,timestampCardiac,idDisease,diseaseName,symptoms,treatment)
SELECT idChild,NOW(),
(SELECT idDisease from diseases where diseaseName='Anemia'),
(select diseaseName from diseases where diseaseName = 'Anemia'),
(select symptoms from diseases where diseaseName = 'Anemia'),
(select treatment from diseases where diseaseName = 'Anemia')
from cardiac
group by idChild;

END
```

Output:



```
select * from cardiac_disease;
```

idChild	timestampCardiac	idDisease	diseaseName	Symptoms	Treatment
1	2015-12-09 18:39:55	3	Apnea	Low Respiratory Rate and Low Heart Rate	Caffeine Medicine
1	2015-12-09 21:46:21	3	Apnea	Low Respiratory Rate and Low Heart Rate	Caffeine Medicine
1	2015-12-09 22:57:55	3	Apnea	Low Respiratory Rate and Low Heart Rate	Caffeine Medicine
1	2015-12-09 22:58:10	8	Intraventricular Haemorrhage	Apnea, Anemia and Low Blood Pressure	Monitor Blood Pressure
1	2015-12-09 22:58:16	3	Apnea	Low Respiratory Rate and Low Heart Rate	Caffeine Medicine
1	2015-12-09 22:58:20	8	Intraventricular Haemorrhage	Apnea, Anemia and Low Blood Pressure	Monitor Blood Pressure
1	2015-12-09 22:58:25	8	Intraventricular Haemorrhage	Apnea, Anemia and Low Blood Pressure	Monitor Blood Pressure
1	2015-12-09 23:01:25	3	Apnea	Low Respiratory Rate and Low Heart Rate	Caffeine Medicine

9 Trigger

There are two types of triggers that we have used in our project.

A. When data is inserted into the body organ table, a trigger is called that identifies the probable disease and calls the respective stored procedure.

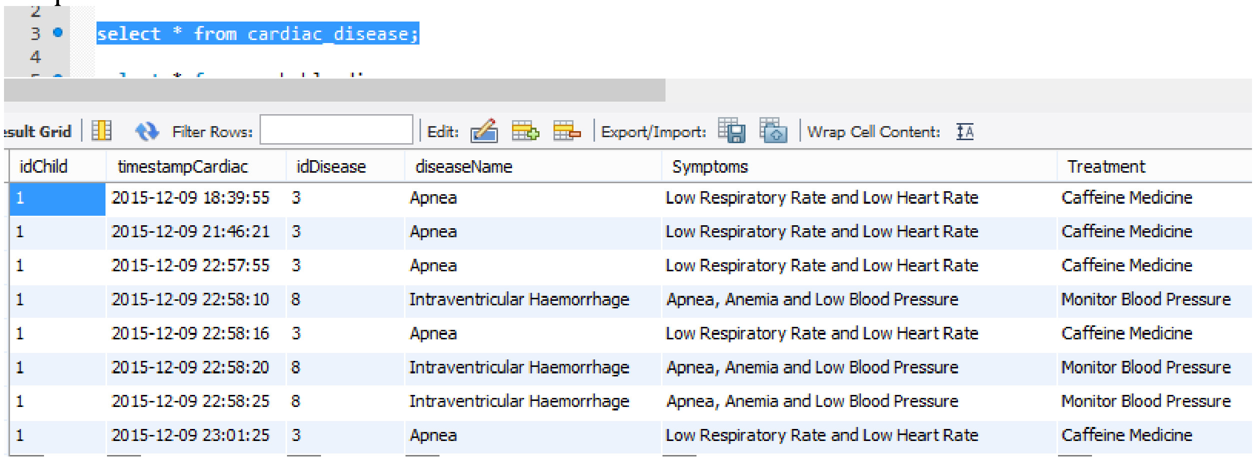
Sample Trigger:

```
USE `neonatal`;
DELIMITER $$
CREATE
DEFINER=`root`@`localhost`
TRIGGER `neonatal`.`cardiacdetails_AINS`
AFTER INSERT ON `neonatal`.`cardiac`
```

```
FOR EACH ROW
begin

IF
(SELECT max(Cardiac.timestampCardiac)
FROM Lungs JOIN  Cardiac
ON Lungs.idChild = Cardiac.idChild
WHERE Lungs.respiratoryRate<30 AND Cardiac.heartRate<65 AND
cardiac.timestampCardiac = (select max(cardiac.timestampCardiac) from cardiac
order by cardiac.timestampCardiac desc limit 1)
ORDER BY Cardiac.timestampCardiac desc
) is not null

THEN
call apnea_procedure;
end if;

if
(SELECT max(cardiac.timestampCardiac)
FROM cardiac JOIN skin
ON cardiac.idChild=skin.idChild
JOIN BLOOD ON(skin.idChild=blood.idChild)
JOIN lungs ON(blood.idChild=lungs.idChild)
where skin.skinType = 'Pale' AND blood.RBC < 4700 AND cardiac.heartRate >190 AND
cardiac.heartRate <230
 AND lungs.respiratoryRate < 12 AND cardiac.timestampCardiac = (select
max(cardiac.timestampCardiac) from cardiac
order by cardiac.timestampCardiac desc limit 1)
order by cardiac.timestampCardiac desc
) is not null


THEN
call anemia_procedure;
end if;

if
(SELECT max(Cardiac.timestampCardiac)
FROM (Cardiac JOIN BloodPressure
ON Cardiac.idChild=BloodPressure.idChild)
JOIN Blood ON(Cardiac.idChild=blood.idChild)
JOIN lungs ON(blood.idChild=lungs.idChild)
where BloodPressure.diastolicBloodPressure <75 AND blood.RBC < 4700 AND
cardiac.heartRate >140 AND cardiac.heartRate <190
AND lungs.respiratoryRate < 30  AND cardiac.timestampCardiac = (select
max(cardiac.timestampCardiac) from cardiac
```

```
order by cardiac.timestampCardiac desc limit 1)
order by cardiac.timestampCardiac desc
) is not null
then
call intraventricular_haemorrhage_procedure;

end if;

if
(SELECT max(cardiac.timestampCardiac)
FROM (cardiac JOIN lungs
ON cardiac.idChild=lungs.idChild)
JOIN ABDOMEN ON(lungs.idChild=abdomen.idChild)
where cardiac.heartRate > 65 and cardiac.heartRate < 100 AND lungs.respiratoryRate < 10
AND abdomen.inflationStatus = 'Yes'  AND cardiac.timestampCardiac = (select
max(cardiac.timestampCardiac) from cardiac
order by cardiac.timestampCardiac desc limit 1)
order by cardiac.timestampCardiac desc
) is not null

then
call necrotizing_enterocolitis_procedure;
end if;

if
(SELECT max(Abdomen.timestampAbdomen)
FROM (Cardiac JOIN Abdomen
ON Cardiac.idChild = Abdomen.idChild)
JOIN Blood ON(Cardiac.idChild = blood.idChild)
JOIN lungs ON(blood.idChild = lungs.idChild)
JOIN Skin ON(lungs.idChild = Skin.idChild)
where Skin.skinType='Pale' AND blood.Bilurubin >1.2 AND cardiac.heartRate > 100
and  cardiac.heartRate < 140
AND lungs.respiratoryRate < 30 AND abdomen.inflationStatus='Yes' and
cardiac.timestampCardiac = (select max(cardiac.timestampCardiac) from cardiac
order by cardiac.timestampCardiac desc limit 1)
order by cardiac.timestampCardiac desc
) is not null

then
call hemolytic_procedure;
end if;

if
(SELECT max(cardiac.timestampCardiac)
FROM (cardiac JOIN skin
ON cardiac.idChild=skin.idChild)
JOIN BloodPressure ON(skin.idChild=BloodPressure.idChild)
JOIN lungs ON(BloodPressure.idChild=lungs.idChild)
where skin.skinType = 'Pale' AND bloodPressure.systolicBloodPressure >120 AND
cardiac.heartRate >230 AND cardiac.heartRate <270
AND lungs.respiratoryRate < 12 and cardiac.timestampCardiac = (select
max(cardiac.timestampCardiac) from cardiac
order by cardiac.timestampCardiac desc limit 1)
order by cardiac.timestampCardiac desc
) is not null
then
call patentductusarteriosis_procedure;
end if;

if
(SELECT max(cardiac.timestampCardiac)
FROM (cardiac JOIN abdomen
ON cardiac.idChild=abdomen.idChild)
where cardiac.heartRate >230 and abdomen.vomittingStatus = 'Yes' and
cardiac.timestampCardiac = (select max(cardiac.timestampCardiac) from cardiac
order by cardiac.timestampCardiac desc limit 1)
order by cardiac.timestampCardiac desc
) is not null
then
call pyloric_stenosis_procedure;
end if;

end
```

Output



```
select * from cardiac_disease;
```

idChild	timestampCardiac	idDisease	diseaseName	Symptoms	Treatment
1	2015-12-09 18:39:55	3	Apnea	Low Respiratory Rate and Low Heart Rate	Caffeine Medicine
1	2015-12-09 21:46:21	3	Apnea	Low Respiratory Rate and Low Heart Rate	Caffeine Medicine
1	2015-12-09 22:57:55	3	Apnea	Low Respiratory Rate and Low Heart Rate	Caffeine Medicine
1	2015-12-09 22:58:10	8	Intraventricular Haemorrhage	Apnea, Anemia and Low Blood Pressure	Monitor Blood Pressure
1	2015-12-09 22:58:16	3	Apnea	Low Respiratory Rate and Low Heart Rate	Caffeine Medicine
1	2015-12-09 22:58:20	8	Intraventricular Haemorrhage	Apnea, Anemia and Low Blood Pressure	Monitor Blood Pressure
1	2015-12-09 22:58:25	8	Intraventricular Haemorrhage	Apnea, Anemia and Low Blood Pressure	Monitor Blood Pressure
1	2015-12-09 23:01:25	3	Apnea	Low Respiratory Rate and Low Heart Rate	Caffeine Medicine

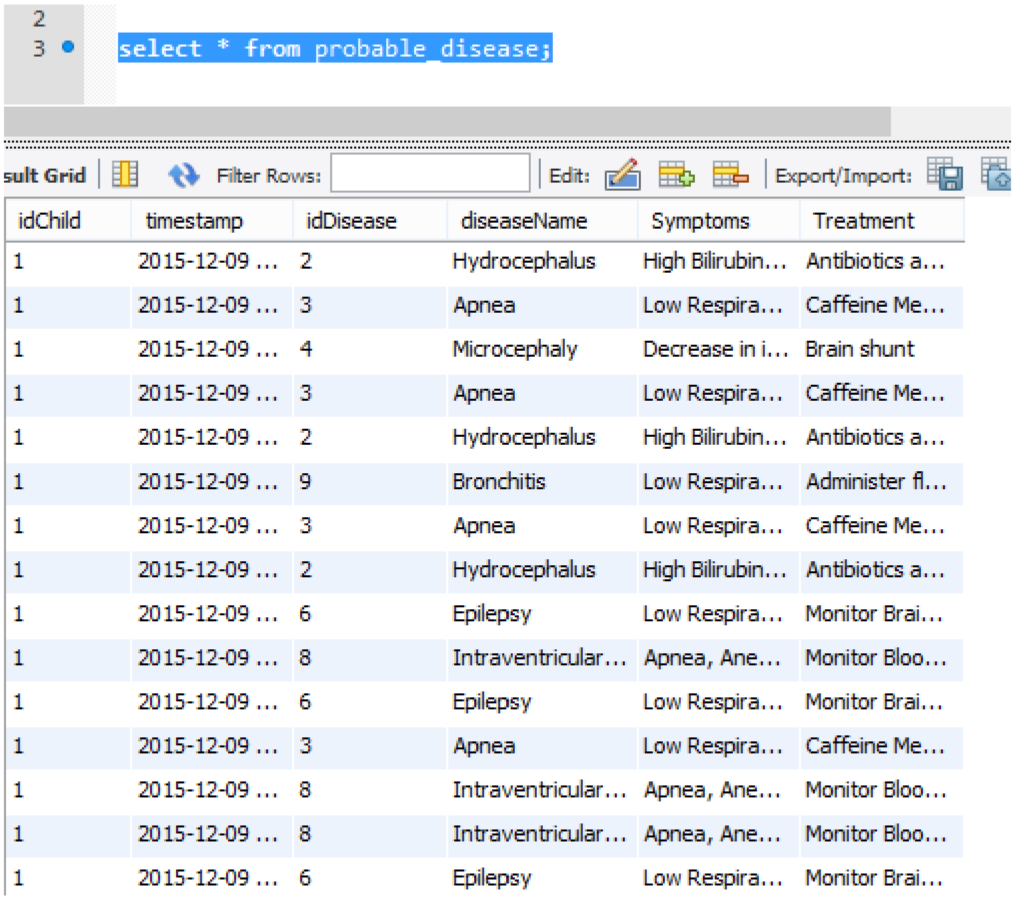
B. When data is inserted into bodyorgan_disease table, a trigger is called that inserts data into the ProbableDisease table.

Sample Trigger

```
USE `neonatal`;
DELIMITER $$
CREATE TRIGGER `cardiac_disease_AINS` AFTER INSERT ON `cardiac_disease` FOR EACH
ROW
begin
if
(Select cardiac_disease.diseaseName from cardiac_disease
order by cardiac_disease.timestampCardiac
desc limit 1) = 'Anemia'
then
insert into Probable_Disease(idChild,
timestamp,idDisease,diseaseName,symptoms,treatment)
(Select idChild, timestampCardiac,idDisease,diseaseName,Symptoms,Treatment from
cardiac_disease order by
cardiac_disease.timestampCardiac desc limit 1);

else if
(Select cardiac_disease.diseaseName from cardiac_disease
order by cardiac_disease.timestampCardiac
desc limit 1) = 'Apnea'
then
insert into Probable_Disease(idChild,
timestamp,idDisease,diseaseName,symptoms,treatment)
(Select idChild, timestampCardiac,idDisease,diseaseName,Symptoms,Treatment from
cardiac_disease
order by cardiac_disease.timestampCardiac desc limit 1);

else if
(Select cardiac_disease.diseaseName from cardiac_disease
order by cardiac_disease.timestampCardiac
desc limit 1) = 'Intraventricular Haemorrhage'
then
insert into Probable_Disease(idChild,
timestamp,idDisease,diseaseName,symptoms,treatment)
(Select idChild, timestampCardiac,idDisease,diseaseName,Symptoms,Treatment from
cardiac_disease
order by cardiac_disease.timestampCardiac desc limit 1);

else if
(Select cardiac_disease.diseaseName from cardiac_disease
order by cardiac_disease.timestampCardiac
desc limit 1) = 'NecrotizingEntorocolitis'
```

```
then
insert into Probable_Disease(idChild,
timestamp,idDisease,diseaseName,symptoms,treatment)
(Select idChild, timestampCardiac,idDisease,diseaseName,Symptoms,Treatment from
cardiac_disease
order by cardiac_disease.timestampCardiac desc limit 1);

else if
(Select cardiac_disease.diseaseName from cardiac_disease
order by cardiac_disease.timestampCardiac
desc limit 1) = 'Hemolytic'
then
insert into Probable_Disease(idChild,
timestamp,idDisease,diseaseName,symptoms,treatment)
(Select idChild, timestampCardiac,idDisease,diseaseName,Symptoms,Treatment from
cardiac_disease
order by cardiac_disease.timestampCardiac desc limit 1);

else if
(Select cardiac_disease.diseaseName from cardiac_disease
order by cardiac_disease.timestampCardiac
desc limit 1) = 'PatentDuctusArteriosis'
then
insert into Probable_Disease(idChild,
timestamp,idDisease,diseaseName,symptoms,treatment)
(Select idChild, timestampCardiac,idDisease,diseaseName,Symptoms,Treatment from
cardiac_disease
order by cardiac_disease.timestampCardiac desc limit 1);

else if
(Select cardiac_disease.diseaseName from cardiac_disease
order by cardiac_disease.timestampCardiac
desc limit 1) = 'Pyloric Stenosis'
then
insert into Probable_Disease(idChild,
timestamp,idDisease,diseaseName,symptoms,treatment)
(Select idChild, timestampCardiac,idDisease,diseaseName,Symptoms,Treatment from
cardiac_disease order by
cardiac_disease.timestampCardiac desc limit 1);

end if;
end if;
end if;
end if;
end if;
end if;
end if;
end;
```

Output



```
select * from probable_disease;
```

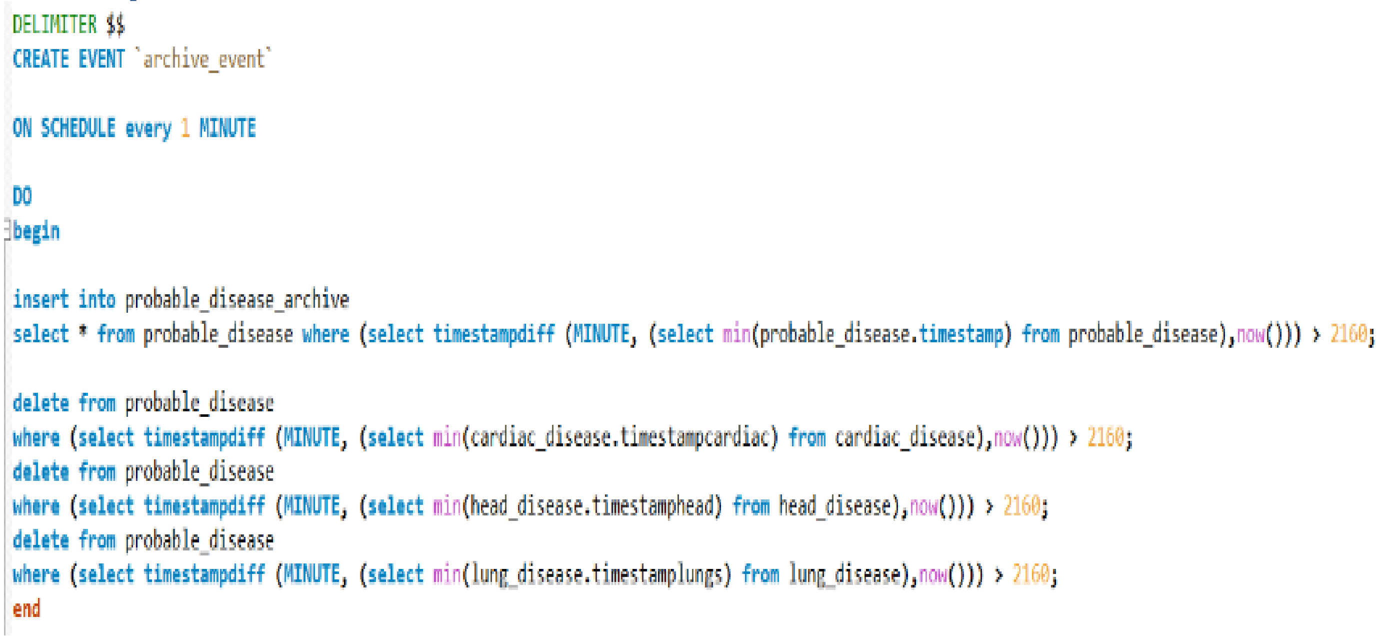
sult Grid | Filter Rows: | Edit: | Export/Import:

idChild	timestamp	idDisease	diseaseName	Symptoms	Treatment
1	2015-12-09 ...	2	Hydrocephalus	High Bilirubin...	Antibiotics a...
1	2015-12-09 ...	3	Apnea	Low Respira...	Caffeine Me...
1	2015-12-09 ...	4	Microcephaly	Decrease in i...	Brain shunt
1	2015-12-09 ...	3	Apnea	Low Respira...	Caffeine Me...
1	2015-12-09 ...	2	Hydrocephalus	High Bilirubin...	Antibiotics a...
1	2015-12-09 ...	9	Bronchitis	Low Respira...	Administer fl...
1	2015-12-09 ...	3	Apnea	Low Respira...	Caffeine Me...
1	2015-12-09 ...	2	Hydrocephalus	High Bilirubin...	Antibiotics a...
1	2015-12-09 ...	6	Epilepsy	Low Respira...	Monitor Brai...
1	2015-12-09 ...	8	Intraventricular...	Apnea, Ane...	Monitor Bloo...
1	2015-12-09 ...	6	Epilepsy	Low Respira...	Monitor Brai...
1	2015-12-09 ...	3	Apnea	Low Respira...	Caffeine Me...
1	2015-12-09 ...	8	Intraventricular...	Apnea, Ane...	Monitor Bloo...
1	2015-12-09 ...	8	Intraventricular...	Apnea, Ane...	Monitor Bloo...
1	2015-12-09 ...	6	Epilepsy	Low Respira...	Monitor Brai...

10 Event Scheduler

After monitoring the baby for 36 hours, data from the probableDisease table is purged, and inserted into the Probable_Disease_Archive table.

10.1 Sample event scheduler



```
DELIMITER $$
CREATE EVENT `archive_event`

ON SCHEDULE every 1 MINUTE

DO
begin

insert into probable_disease_archive
select * from probable_disease where (select timestampdiff (MINUTE, (select min(probable_disease.timestamp) from probable_disease),now())) > 2160;

delete from probable_disease
where (select timestampdiff (MINUTE, (select min(cardiac_disease.timestampcardiac) from cardiac_disease),now())) > 2160;
delete from probable_disease
where (select timestampdiff (MINUTE, (select min(head_disease.timestamphead) from head_disease),now())) > 2160;
delete from probable_disease
where (select timestampdiff (MINUTE, (select min(lung_disease.timestamplungs) from lung_disease),now())) > 2160;
end
```

11 Views

1. We have created views for the specialized doctors to view the data inside Probable_Disease table.
2. If the baby is suffering from a cardiac disease like Apnea, then the cardiologist can view this information from the Probable_Disease table.

11.1 Sample View:

CREATE VIEW `neonatal`.`view_neurologist` as
SELECT * from probable_disease
where probable_disease.diseaseName = 'Epilepsy' or probable_disease.diseaseName = 'Hydrocephalus'
or probable_disease.diseaseName='Microcephaly';

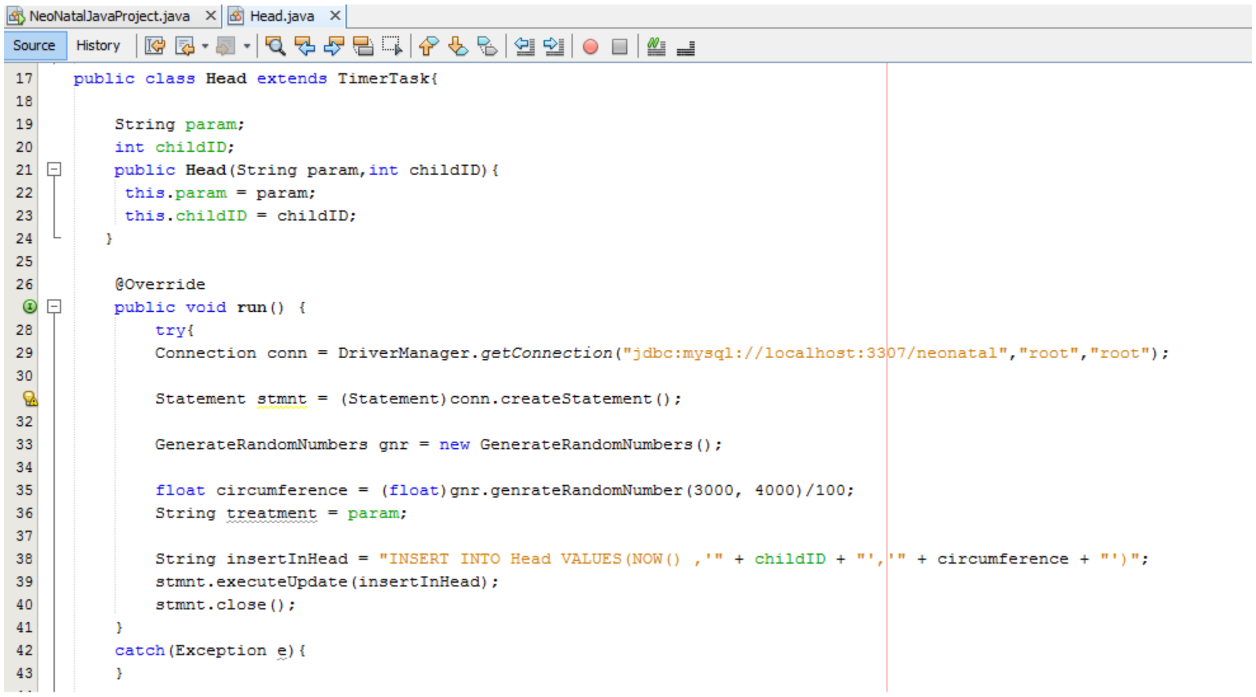
12 Normalization

We are using associative tables to provide normalization and flexibility to our project. When we defined the logical model for our database, we noted that the organ to probable disease relationship is many-to-many. So we translated that relationship into something which is functional. To do this, we created associative tables whose main purpose is to store foreign keys. Another advantage of associative tables in our case is that we are doing comparison of vital signs where the data is coming from different organ tables so we needed a common staging area where the data can be stored and passed on to the next table.

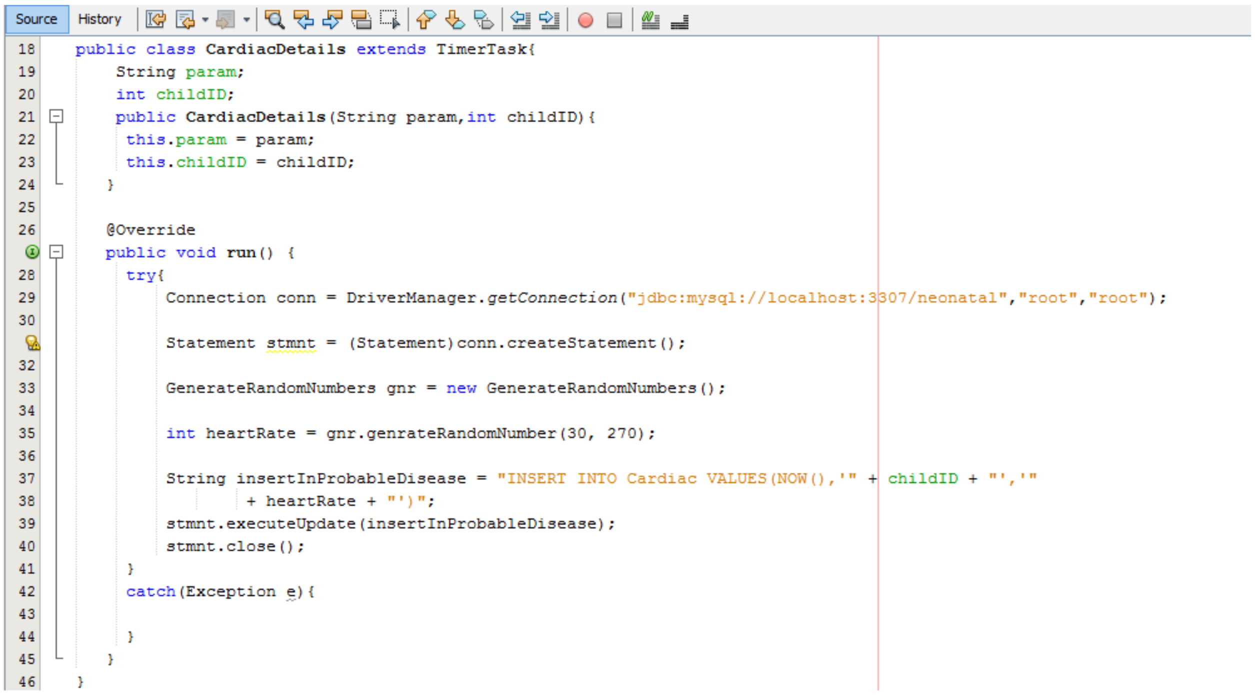
13 Java Code for random data generation

1. We created a Java project which inserts data into the body organ tables like Head, Lungs, Blood, Blood Pressure, etc.
2. This code mimics the actions of the sensors connected to the baby's head, lungs etc.
3. This code inserts data in the range that can be manipulated. For example, if the valid range of the body temperature is between 97 degree and 100.7 degree Fahrenheit, then the java code can be manipulated to generate random data between or outside this range.
4. Using this code we can generate random data at fixed interval.

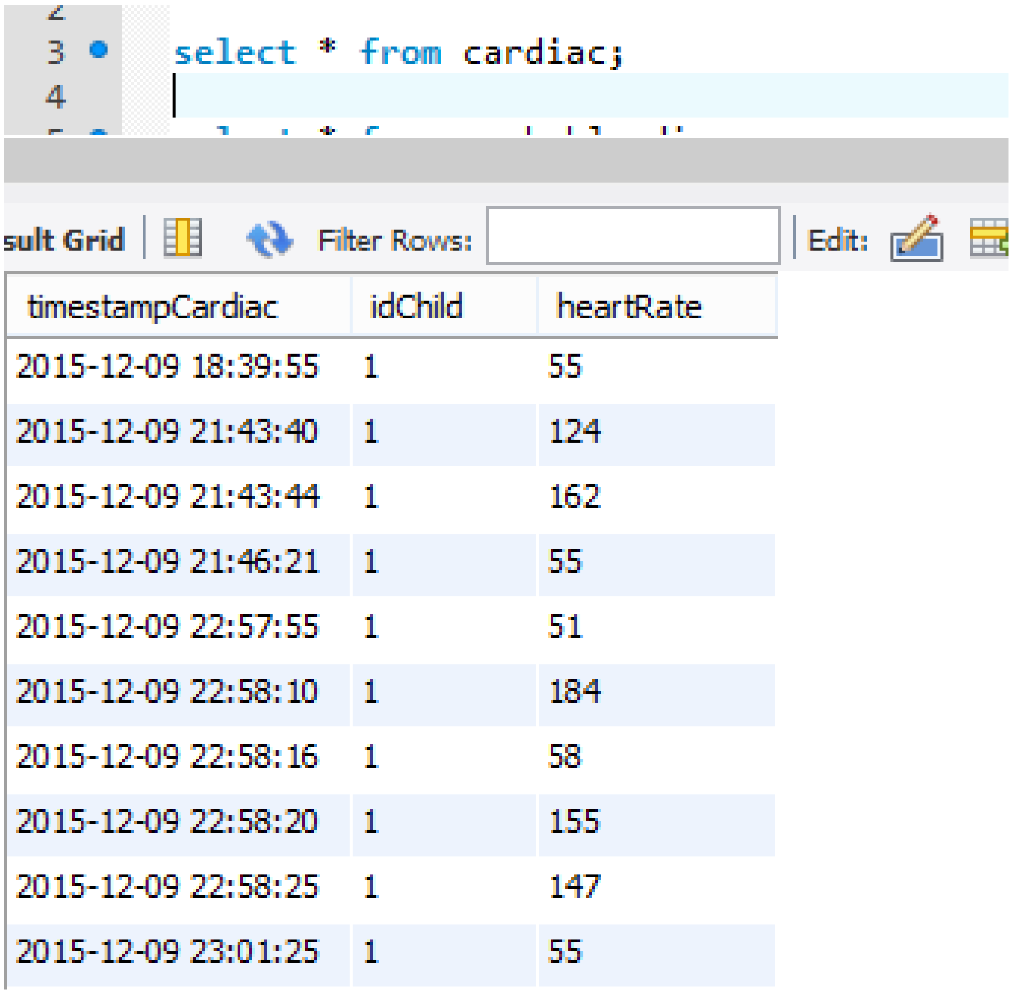
13.1 Sample code



```java
public class Head extends TimerTask{

    String param;
    int childID;
    public Head(String param,int childID){
     this.param = param;
     this.childID = childID;
   }

    @Override
    public void run() {
        try{
        Connection conn = DriverManager.getConnection("jdbc:mysql://localhost:3307/neonatal","root","root");

        Statement stmnt = (Statement)conn.createStatement();

        GenerateRandomNumbers gnr = new GenerateRandomNumbers();

        float circumference = (float)gnr.genrateRandomNumber(3000, 4000)/100;
        String treatment = param;

        String insertInHead = "INSERT INTO Head VALUES(NOW() ,'" + childID + "','" + circumference + "')";
        stmnt.executeUpdate(insertInHead);
        stmnt.close();
    }
    catch(Exception e){
    }
```



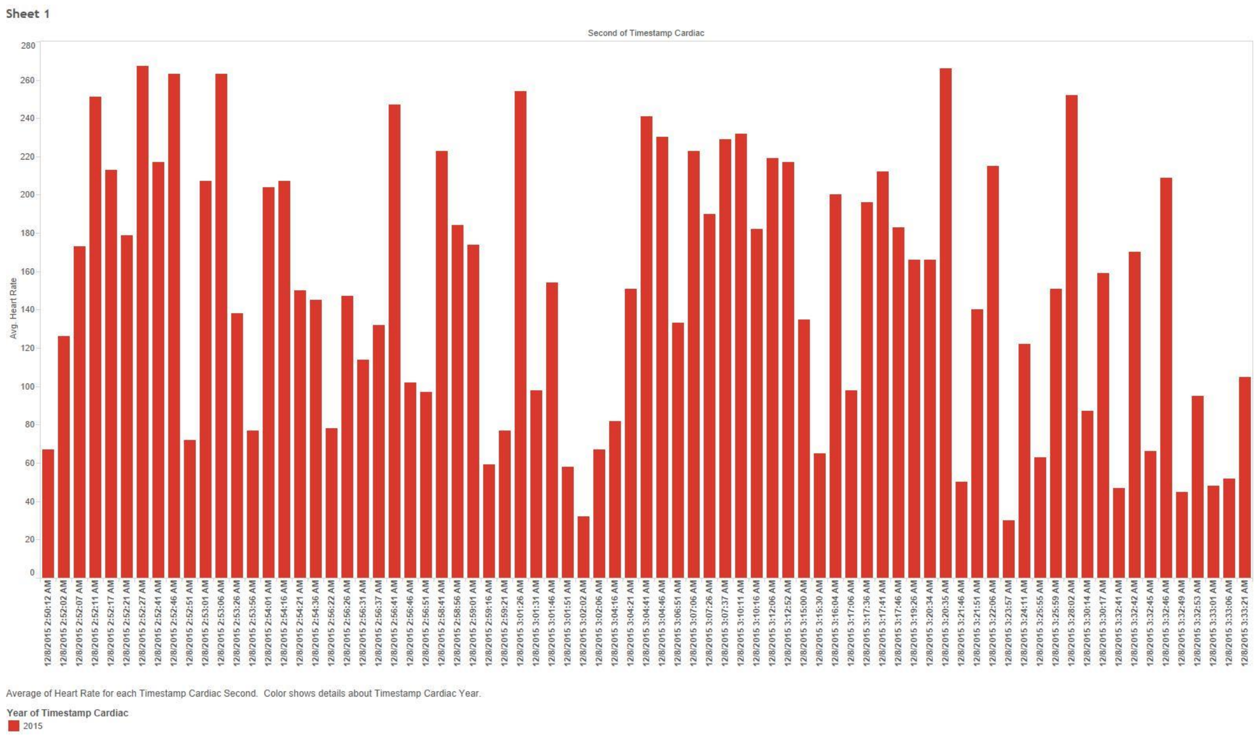
```
public class CardiacDetails extends TimerTask{
    String param;
    int childID;
    public CardiacDetails(String param,int childID){
     this.param = param;
     this.childID = childID;
    }

   @Override
   public void run() {
     try{
         Connection conn = DriverManager.getConnection("jdbc:mysql://localhost:3307/neonatal","root","root");

         Statement stmnt = (Statement)conn.createStatement();

         GenerateRandomNumbers gnr = new GenerateRandomNumbers();

         int heartRate = gnr.genrateRandomNumber(30, 270);

         String insertInProbableDisease = "INSERT INTO Cardiac VALUES(NOW(),'" + childID + "','"
                 + heartRate + "')";
         stmnt.executeUpdate(insertInProbableDisease);
         stmnt.close();
     }
     catch(Exception e){

     }
   }
}
```



```
select * from cardiac;
```

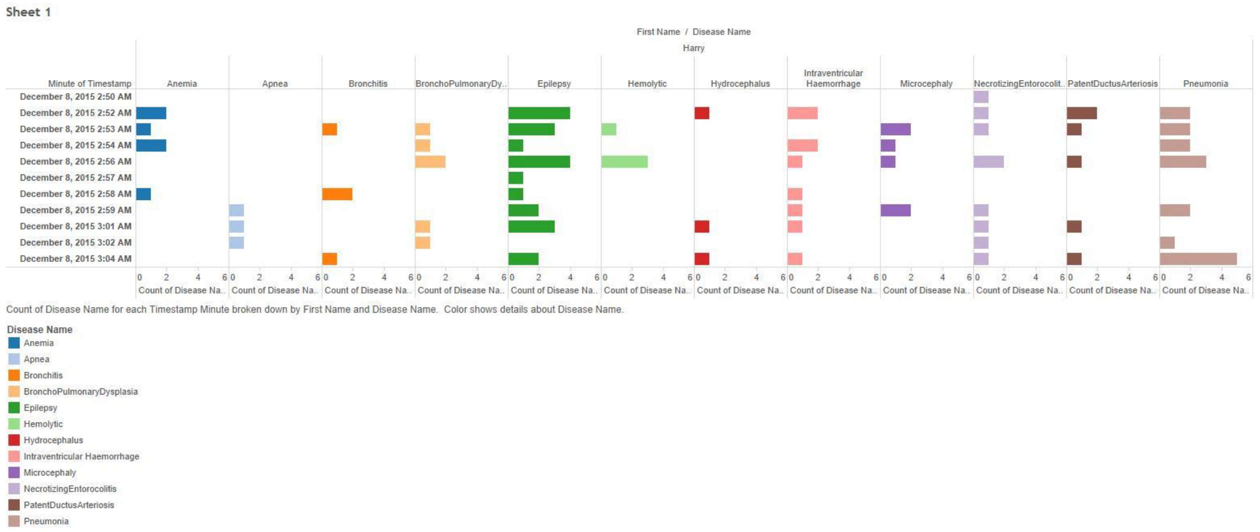
timestampCardiac	idChild	heartRate
2015-12-09 18:39:55	1	55
2015-12-09 21:43:40	1	124
2015-12-09 21:43:44	1	162
2015-12-09 21:46:21	1	55
2015-12-09 22:57:55	1	51
2015-12-09 22:58:10	1	184
2015-12-09 22:58:16	1	58
2015-12-09 22:58:20	1	155
2015-12-09 22:58:25	1	147
2015-12-09 23:01:25	1	55

14 Tableau Implementation



14.1 Description:

The above graph denotes the average heart rate per minute of a child.



14.2 Description:

The above graph denotes the minute wise disease occurrence in an infant and the likelihood of the occurrence of the disease.

15 Deliverables

The deliverables that we achieved in this project are:

1. To monitor the child on an hourly and secondly basis
2. To obtain real time data feed, for which we made use of a java program and connected it to our database using JDBC/ ODBC connection
3. To record all the vital signs in the database
4. To find the most probable disease a child is suffering from
5. To generate triggers and stored procedures to caution doctors and nurses
6. Insert data into an archival table after 36 hours using event scheduler
7. Made use of BI tool, Tableau to generate reports related to most probable diseases observed in neonates.

To create a one stop database consisting of records of symptoms, causes, treatments and list of diseases.

16 Integrated Master Schedule/Milestones:

Following are some key schedule tasks and milestones which are the building blocks of this project:

1. Final ER and Enhanced ER diagram of the whole Neonatal 36 system was completed by 10/20/2015.
2. Searching and creation of some actual and sample data which was populated in database and its tables. This task was accomplished by 10/27/2015.
3. We started creating the database and its structure in the last week of October and populated them with dummy data to check the queries.
4. We started working on the daily auto backup of the data and the retention policy.
5. We additionally kept up the database security and access control lists by making use of user privileges.
6. Made use of triggers, event schedulers, and stored procedures to obtain the required results.
7. Coordinated the database with reporting Tableau by 11/30/2015.

17 References:

Some references which are being followed:

1. http://www.parents.com/baby/care/newborn/baby-first-hours/
2. http://www.babycenter.com/0_first-24-hours-at-home-with-your-baby_10345806.bc
3. https://www.nlm.nih.gov/medlineplus/ency/article/007233.htm
4. https://www.nlm.nih.gov/medlineplus/uncommoninfantandnewbornproblems.html
5. https://www.nlm.nih.gov/medlineplus/prematurebabies.html
6. http://www.marchofdimes.org/baby/common-nicu-equipment.aspx