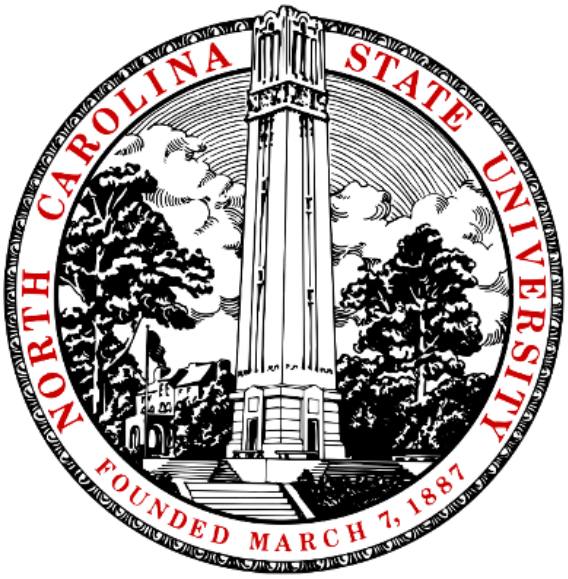
Wearable System for Health and Environmental Tracking

NORTH CAROLINA STATE UNIVERSITY
DATABASE APPLICATIONS IN INDUSTRIAL & SYSTEMS ENGINEERING
ISE 519 – SPRING 2022

Final Project Team 6



Authors:
Anirudh Nori
Bhavya Sai Volepu
Venkata Mahesh Punugu
Nandhini Ravi

Instructor:
Dr. Michael Spano

25 April 2022

Table of Contents

1 Abstract

The scope of this project is to build a web-user interface on MS Access that can return results of the followings: -

1. When and for how much time users are wearing the HET system.
2. Some simple indicators of the device's signal quality to determine whether it is operating as expected.

The project will help the client determine the patterns for predicting Asthma at an early stage. From the data analysis, this project can correlate environmental factors and physiological parameters in individuals with asthma compared with those who do not. Using an integrated analytics dashboard, the client can check the progress of their research work.

2 Project Definition

The ASSIST lab of North Carolina State University focuses on creating self-powered sensing, computing, and communication systems to enable data-driven insights for a bright and healthy world. The Health and Environmental Tracking system supports those suffering from chronic respiratory disease. The HET system contains a chest patch and a wrist-worn band. Both devices provide real-time monitoring of the user's pulse rate, electrocardiogram, pulse rate activity, and environmental ozone exposure in this system. The goal of the HET system is to determine if these devices can predict exacerbated asthma conditions hours before they occur.

3 Potential next steps:

1. **Data Extraction**: The readings from the sensors are in the NumPy format, which will be cleaned and converted into CSV using the python libraries such as NUMPY and Pandas, which are then loaded into the MS ACCESS, a cleansed database on which we would be performing the given below queries.
2. **MS Access Database:** Creating SQL Queries for assessing signal quality, usage tracking, and operational performance and querying the time stamp during which the maximum number of users wears the tracker and querying the cough rate, the date obtained in the audio recordings, which are in the binary label format.
3. **Statistical Analysis**: Sampling Time series data to perform analysis on signal quality, usage, and operational performance of the wearable device.
4. **Analytics Dashboard**: Integrating Access database with Microsoft Power BI to create compelling descriptive visual charts to understand information and generate reports quickly.

4 Data Extraction

The data recorded for both devices is in the NumPy format. NumPy format is the standard binary file format in NumPy for persisting a single arbitrary NumPy array on a disk. The format stores l the shape and dtype information necessary to reconstruct the array correctly even on another machine with a different architecture. To extract the data from NumPy into tabular format, python library NumPy is used to convert the array data into NumPy series format. Further, Python library Pandas used to convert the series format into tabular format. From pandas' data frame, we have used the group by sampling function to reduce the size of data into equivalent size of all device's measurements. The data frames are converted into excel files then those excel files are uploaded in MS Access Database. Additionally, the data has been collected on different days. We have concatenated all the data day wise readings into one data frame.

```
ECG_65 = np.load("C:\\Users\\nsani\\Desktop\\Courses\\ISE 519\\Project\\sample_preprocessed_data.exdir\\sample_preprocessed_data.
```

```
acc_x_65 = np.load("C:\\Users\\nsani\\Desktop\\Courses\\ISE 519\\Project\\sample_preprocessed_data.exdir\\sample_preprocessed_dat
acc_y_65 = np.load("C:\\Users\\nsani\\Desktop\\Courses\\ISE 519\\Project\\sample_preprocessed_data.exdir\\sample_preprocessed_dat
acc_z_65 = np.load("C:\\Users\\nsani\\Desktop\\Courses\\ISE 519\\Project\\sample_preprocessed_data.exdir\\sample_preprocessed_dat
```

```
hr_65 = np.load("C:\\Users\\nsani\\Desktop\\Courses\\ISE 519\\Project\\sample_preprocessed_data.exdir\\sample_preprocessed_data.e
```

The code used for loading the NumPy data

```
ECG_final = ECG.groupby('Day', group_keys=False).apply(lambda x: x.sample(frac=0.0077218))
```

```
Acc_final = ACC.groupby('Day', group_keys=False).apply(lambda x: x.sample(frac=0.00496857))
```

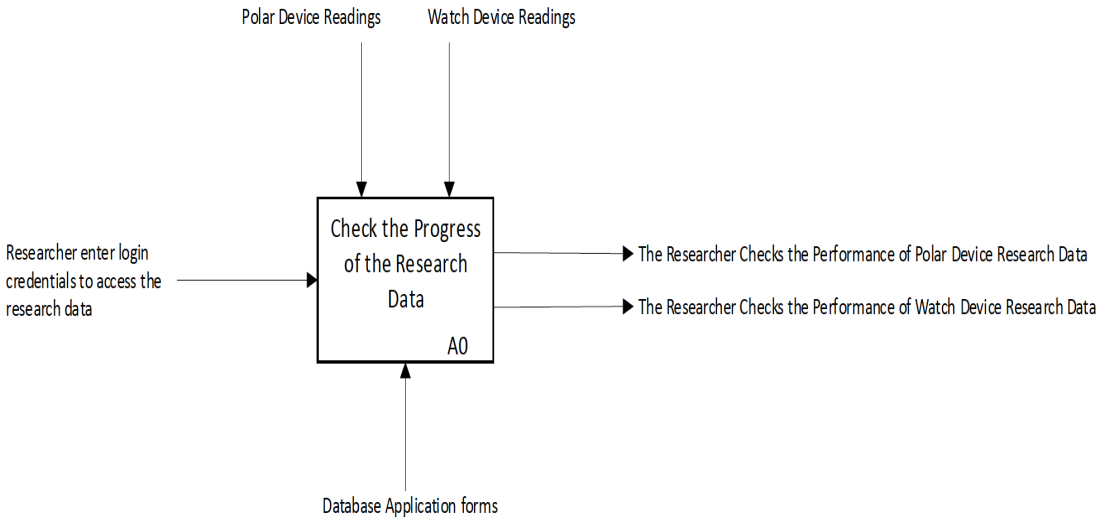
```
Gyro['Device'] = pd.Series(["Watch" for x in range(len(Gyro.index))])
```

Group By sampling the data for reducing the dimension of the data

```
# to excel file
Gyro.to_excel("C:\\Users\\nsani\\Desktop\\Courses\\ISE 519\\Excel\\Gyro.xlsx")
```

Converting the data into excel format

5 IDEF0



Context:

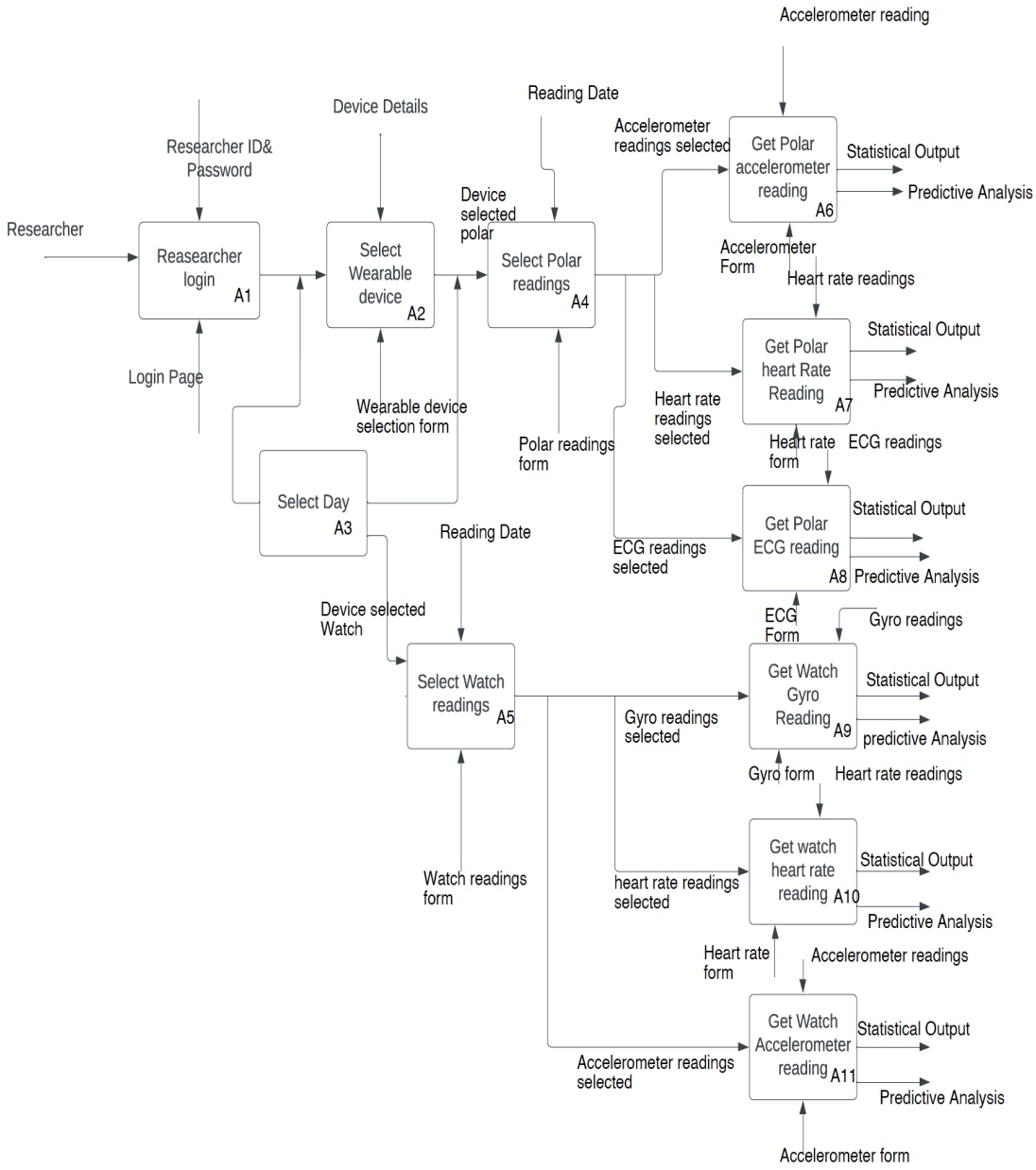
In this project, Researcher will be able to check the statistical output (Averages) of the readings collected from the wearable devices for developing inference useful for future development of research

Viewpoint:

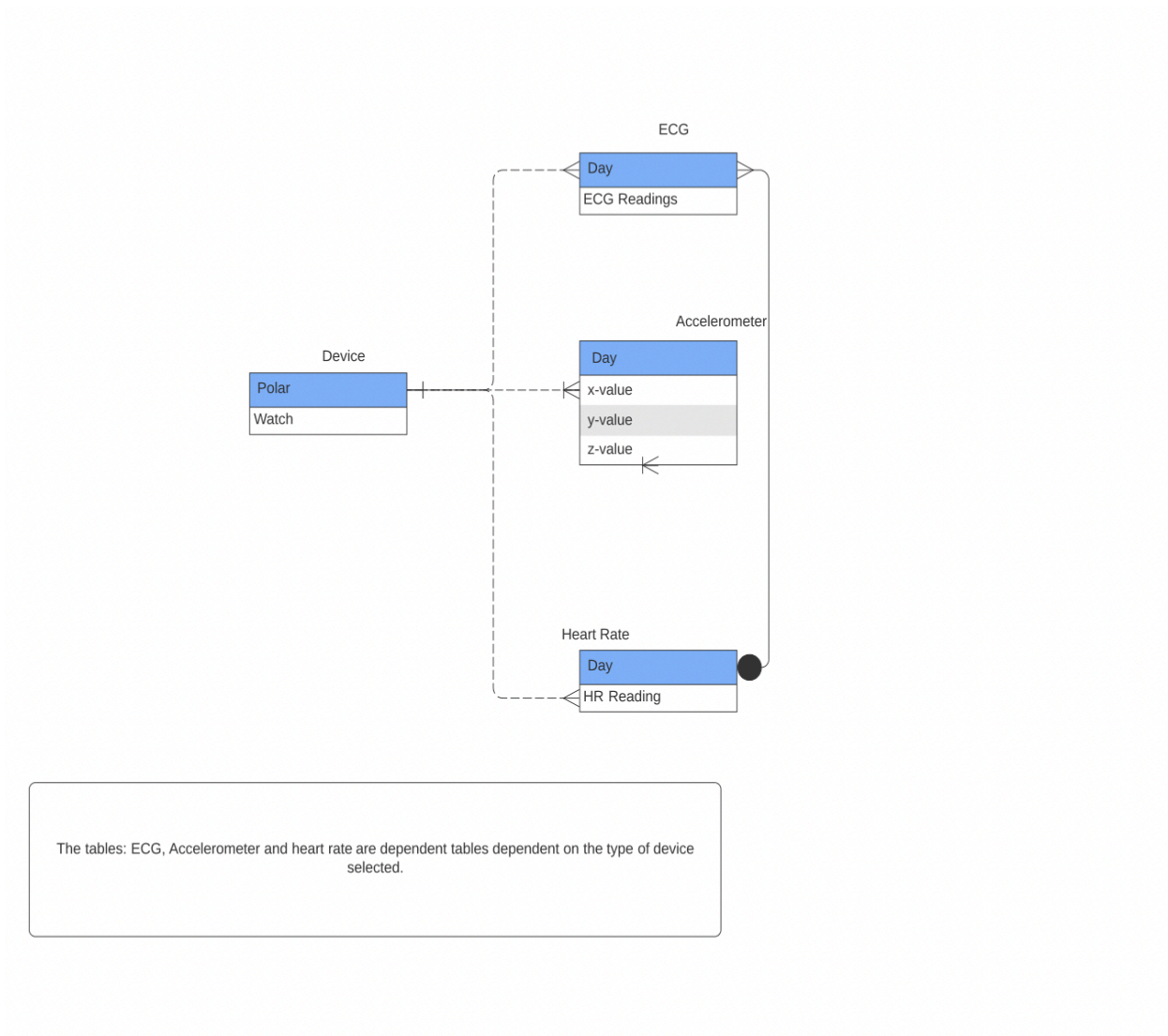
The viewpoint is from the researcher who will want to check the readings not from the view of the database developer .

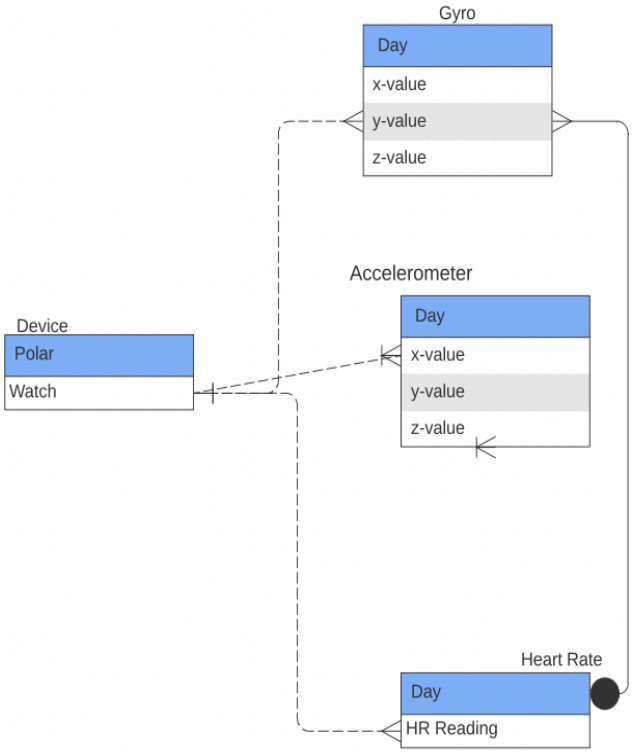
Purpose:

The model will show the necessary steps that needs to be followed to successfully retrieve statistical output of different device reading for the research. It will indicate activities, nodes and the forms that support the different activities.


Accelerometer reading
Reading Date
Device Details
Researcher ID& Password
Accelerometer readings selected
Get Polar accelerometer reading A6
Statistical Output
Predictive Analysis
Device selected polar
Researcher
Reasearcher login A1
Select Wearable device A2
Select Polar readings A4
Accelerometer Form
Heart rate readings
Get Polar heart Rate Reading A7
Statistical Output
Predictive Analysis
Login Page
Wearable device selection form
Heart rate readings selected
Polar readings form
Heart rate form
ECG readings
Select Day A3
Get Polar ECG reading A8
Statistical Output
Predictive Analysis
Reading Date
ECG readings selected
Device selected Watch
ECG Form
Gyro readings
Select Watch readings A5
Get Watch Gyro Reading A9
Statistical Output
predictive Analysis
Gyro readings selected
Gyro form
Heart rate readings
Get watch heart rate reading A10
Statistical Output
Predictive Analysis
Watch readings form
heart rate readings selected
Heart rate form
Accelerometer readings
Get Watch Accelerometer reading A11
Statistical Output
Predictive Analysis
Accelerometer readings selected
Accelerometer form

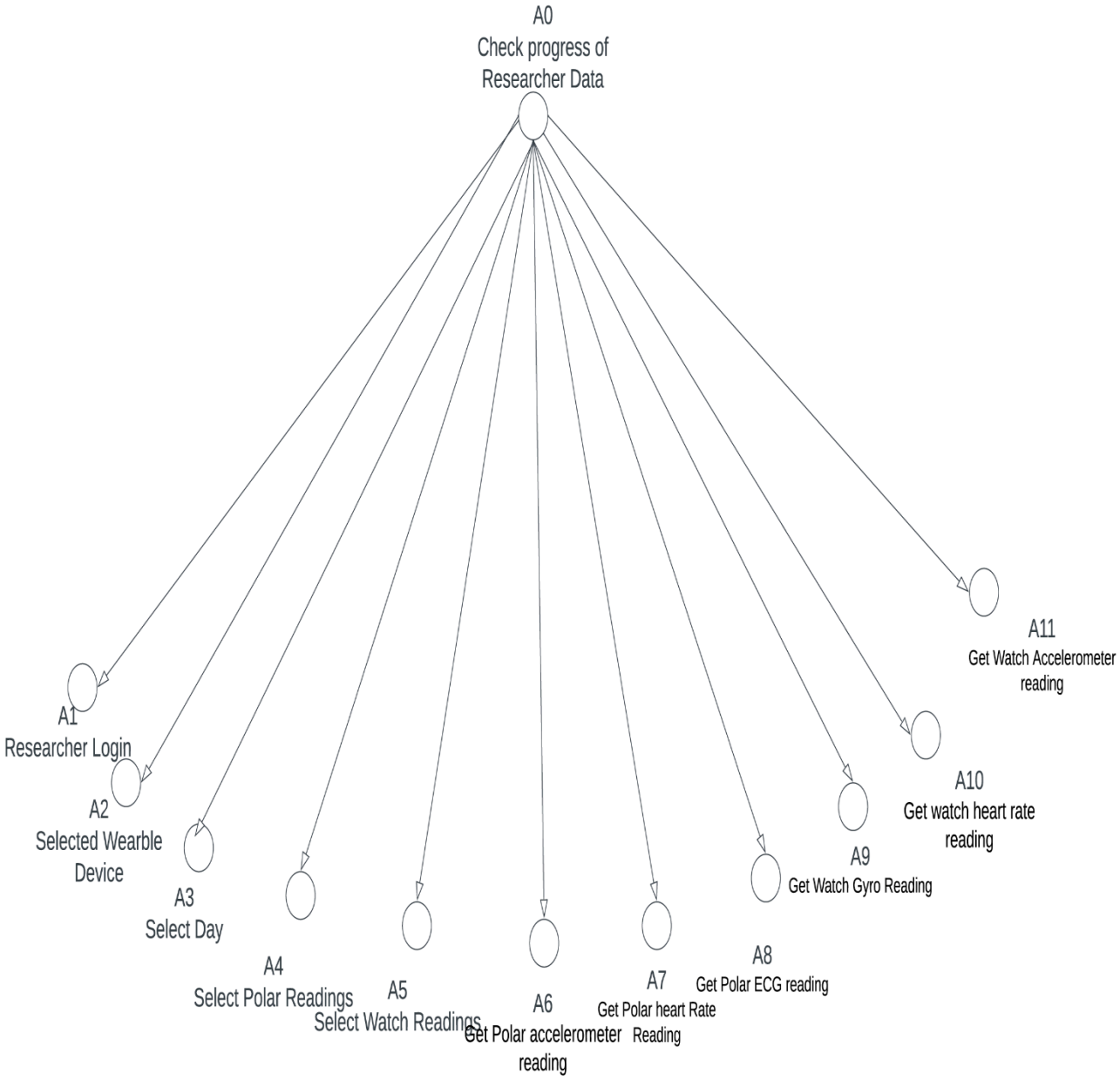
6 IDEF1X


ECG
Day
ECG Readings
Accelerometer
Device
Day
Polar
x-value
Watch
y-value
z-value
Heart Rate
Day
HR Reading
The tables: ECG, Accelerometer and heart rate are dependent tables dependent on the type of device selected.



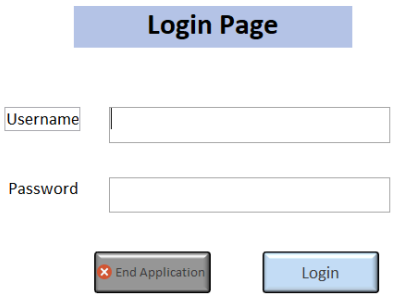
The tables: Gyro,Accelerometer and Heart rate are dependent tables that depends on the type of device selected.

7 Node Diagram



8 Forms

8.1 Login Form



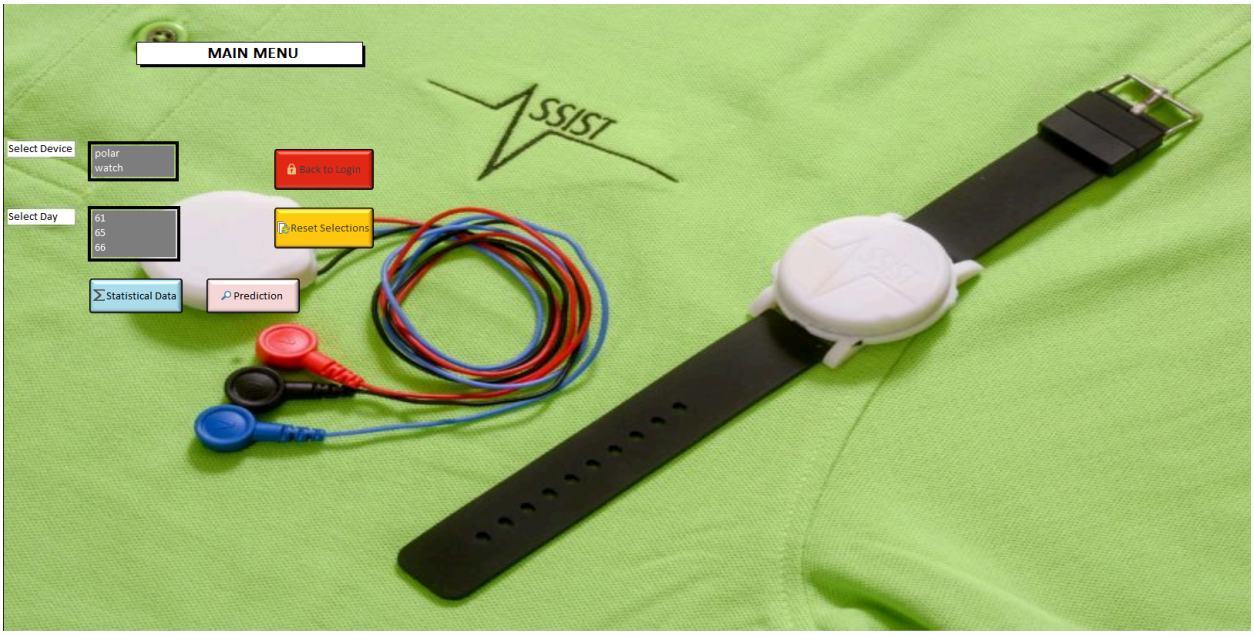
SSIST
Transformative technologies for personalized, vigilant health monitoring

Object Name	Query	Macro	Event Property
Username_txt	N/A	N/A	N/A
password_txt	N/A	N/A	N/A
login_btn	N/A	VBA code	On click

8.1.1 VBA Code for Login Form:

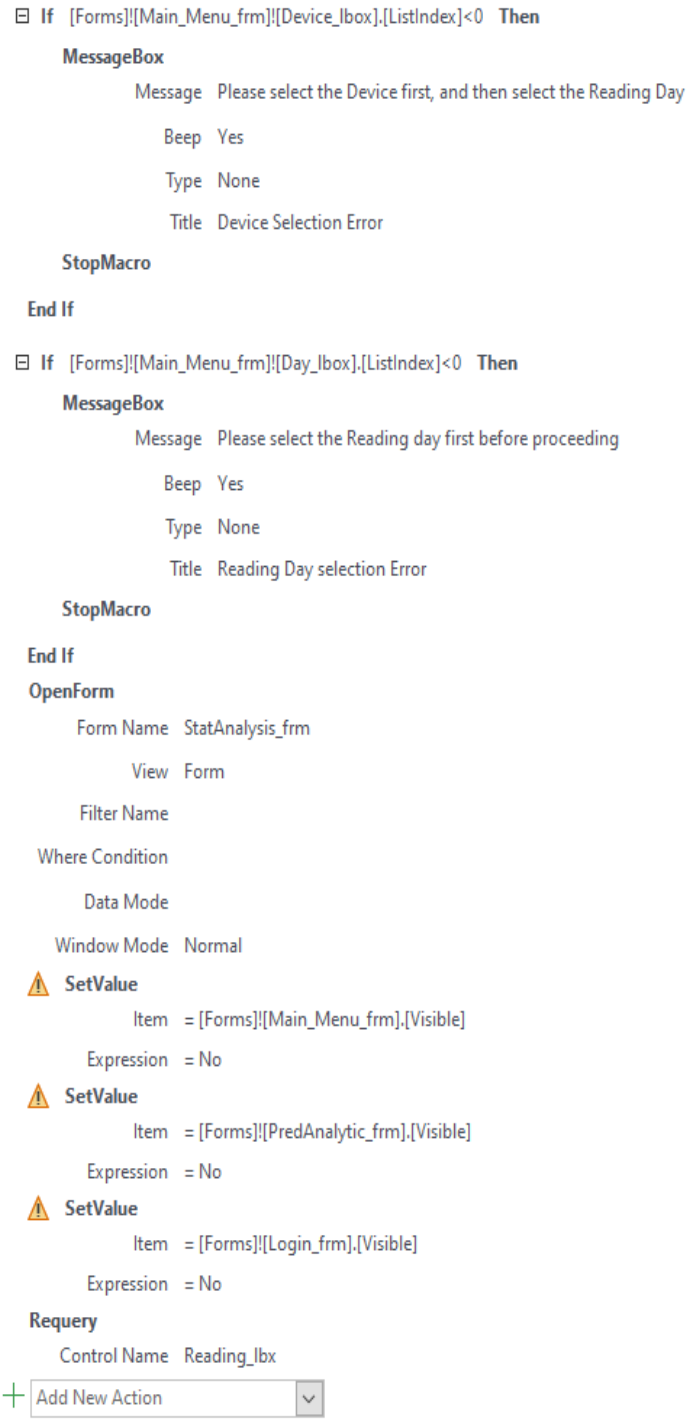
```
Option Compare Database

Private Sub Command4_Click()

If pass1 = DLookup("password", "tblUser", "username='" & [user1] & " '") Then

DoCmd.OpenForm "Main_Menu_frm"
Forms!Main_Menu_frm.Visible = True

DoCmd.OpenForm "PredAnalytic_frm"
Forms!PredAnalytic_frm.Visible = False

DoCmd.OpenForm "StatAnalysis_frm"
Forms!StatAnalysis_frm.Visible = False


Else

MsgBox " Error username or passoword"

End If



End Sub
```

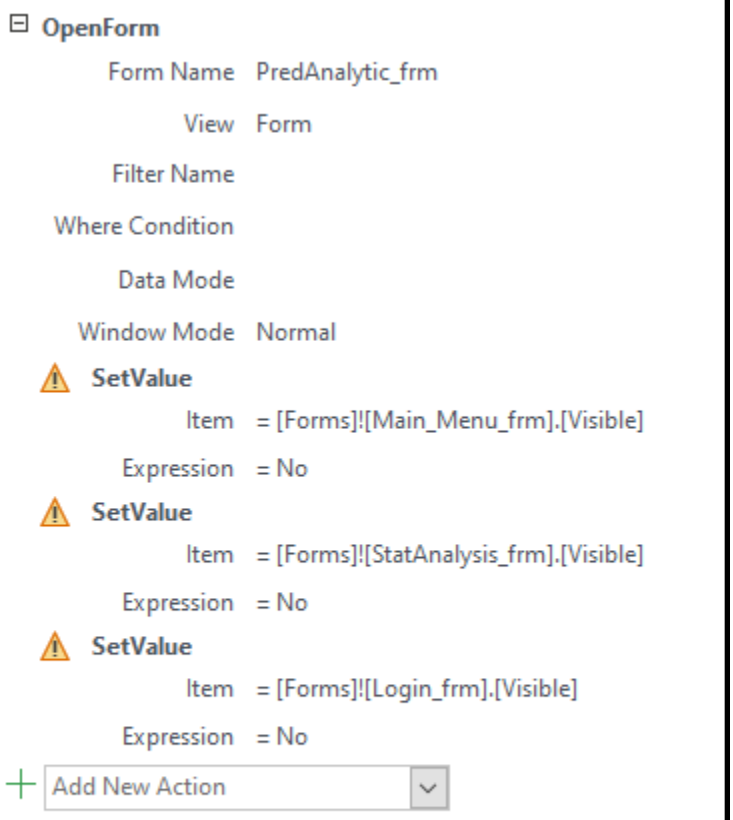
8.2 MAIN MENU FORM:

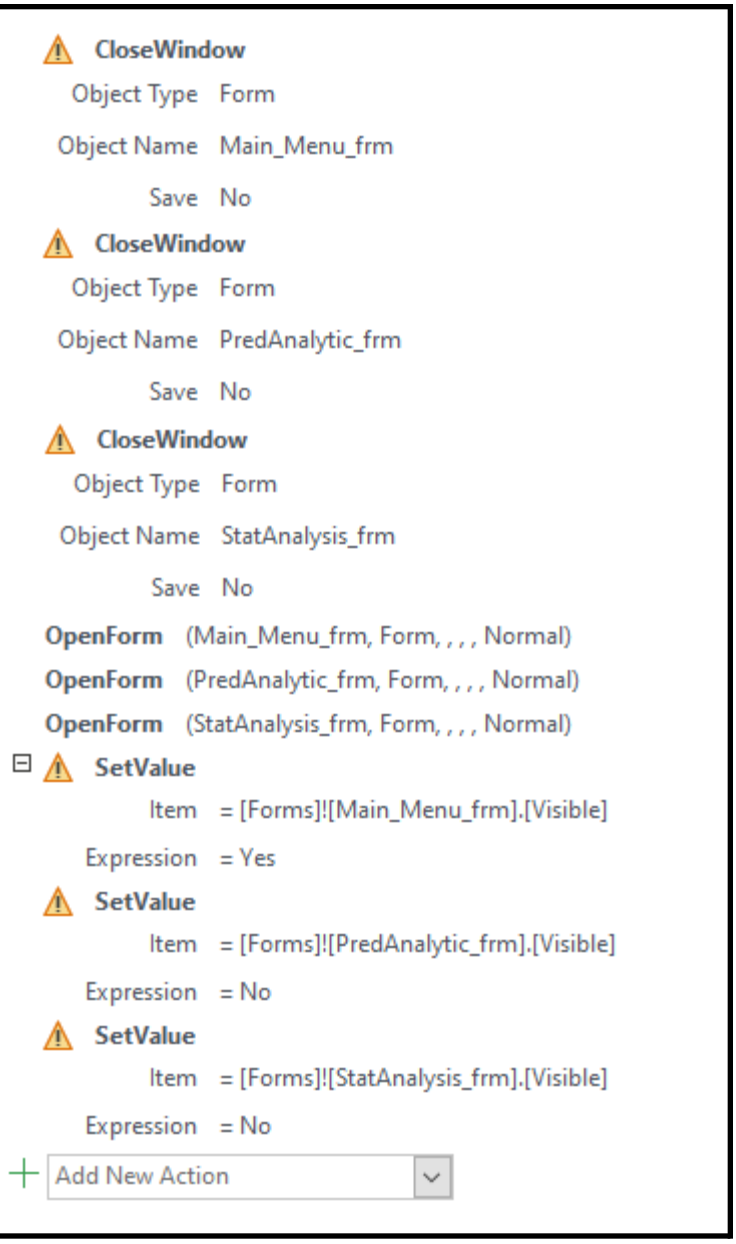


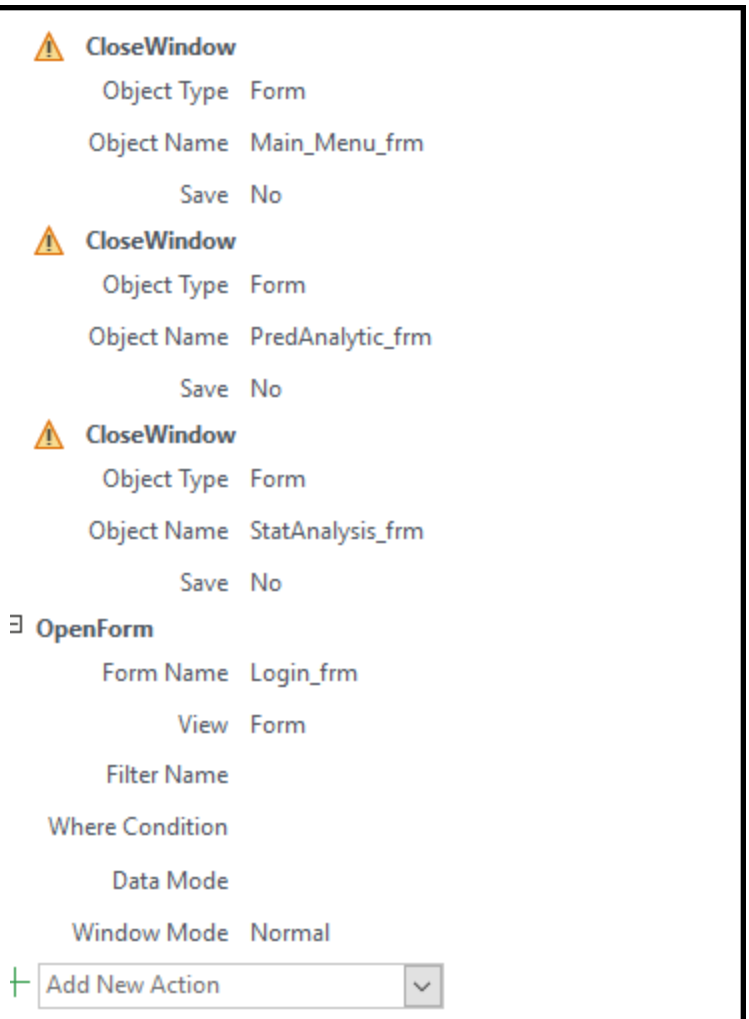
Object Name	Query/Row Source	Macro	Event Property
Device_lbox	SELECT DISTINCT Device FROM Device_tbl;	N/A	N/A
Day_lbox	SELECT DISTINCT Reading_day FROM Device_tbl;	N/A	N/A
Statistical data_btn	N/A	GoToStatfrm_macro	On Click
Prediction_btn	N/A	GoToPredfrm_macro	On Click
Reset Selection_btn	N/A	Reset_Mainfrm_macro	On Click
Back_to_Login_btn	N/A	ReturnfromMainfrm_macro	On Click

8.2.1 Main Menu Macros:

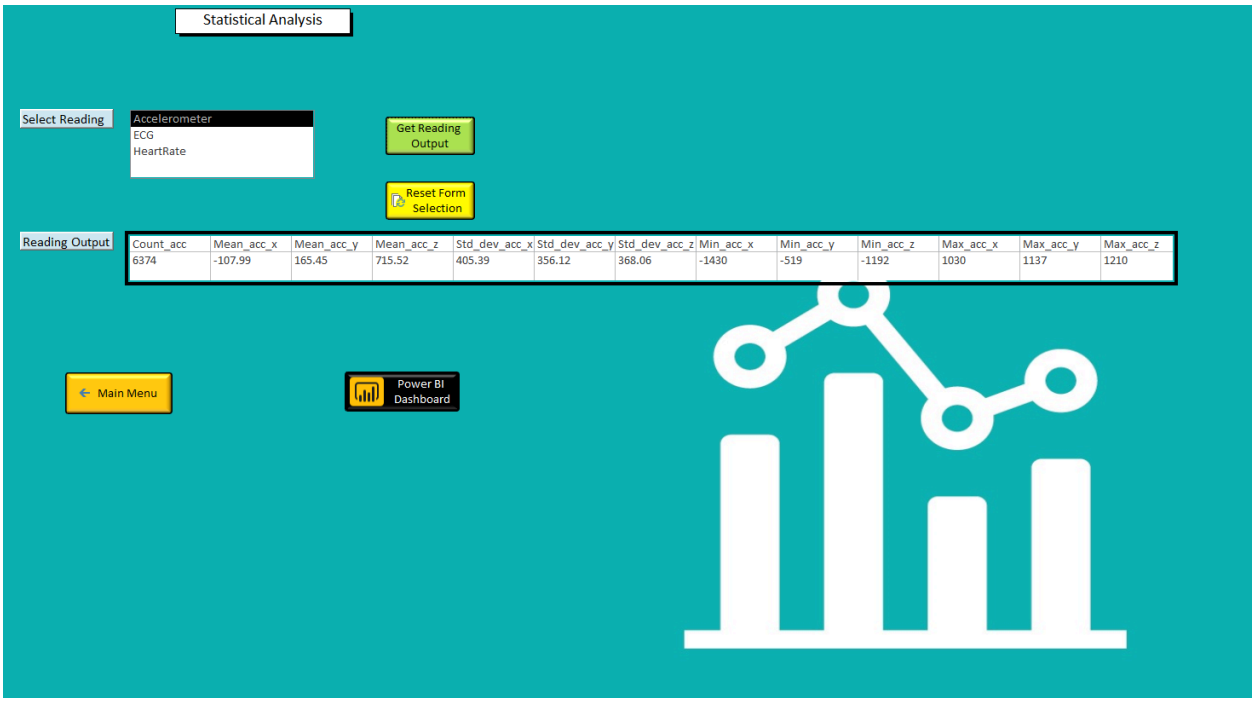
Macro Name	Object Name	Macro
GoToStatfrm_macro	Statistical data_btn	

GoToPredfrm_macro	Prediction_btn	

Reset_Mainfrm_macro	Reset Selection_btn	

ReturnfromMainfrm_macro	Back_to_Login_btn	

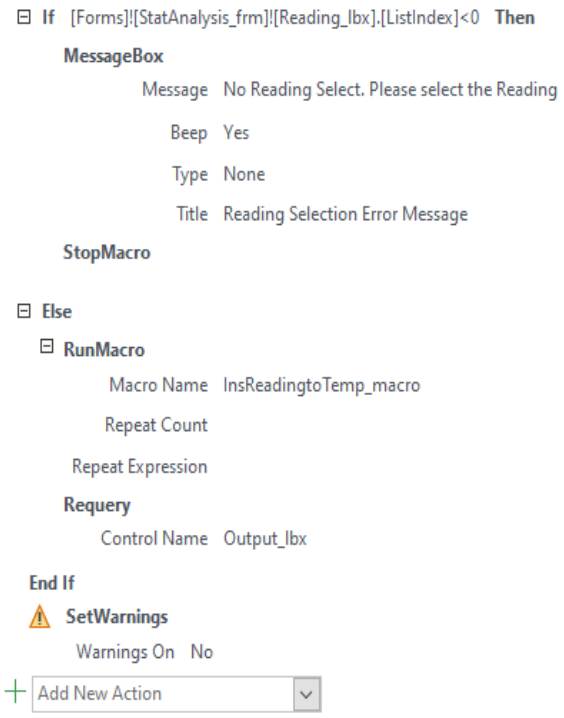
8.3 Statistical Analysis Form

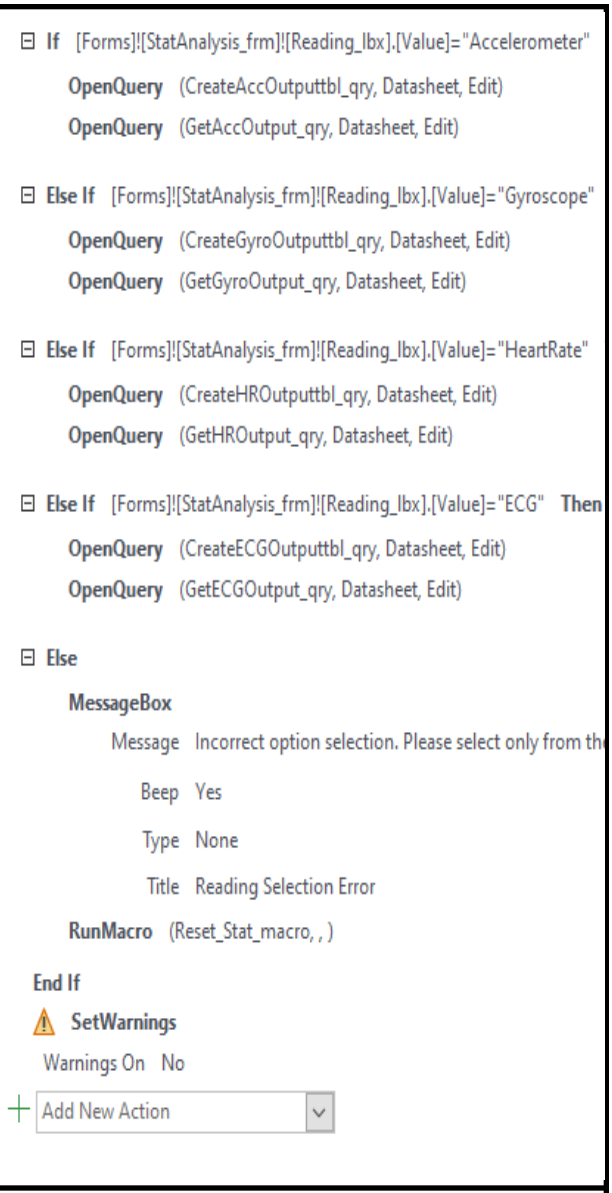


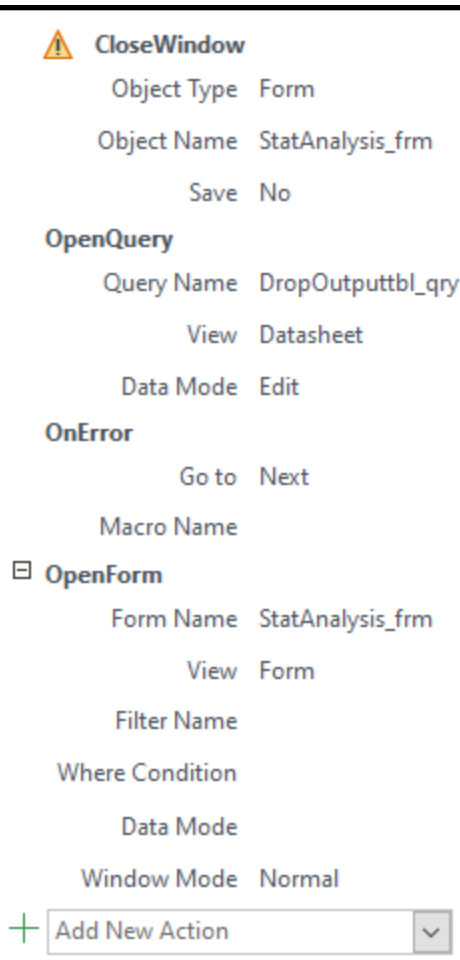
Object Name	Query/Row Source	Macro	Event Property
Reading_lbx	SELECT Reading FROM Device_tbl WHERE [Forms]![Main_Menu_frm]![Device_lbox].Value = [Device_tbl].[Device] AND [Forms]![Main_Menu_frm]![Day_lbox].Value = [Device_tbl].[Reading_day];	N/A	N/A
Output_lbx	Output_tbl	N/A	N/A
Get_Reading_Output_btn	N/A	GetReadingOutput_macro (Macro within above Macro - InsReadingtoTemp_macro)	On Click

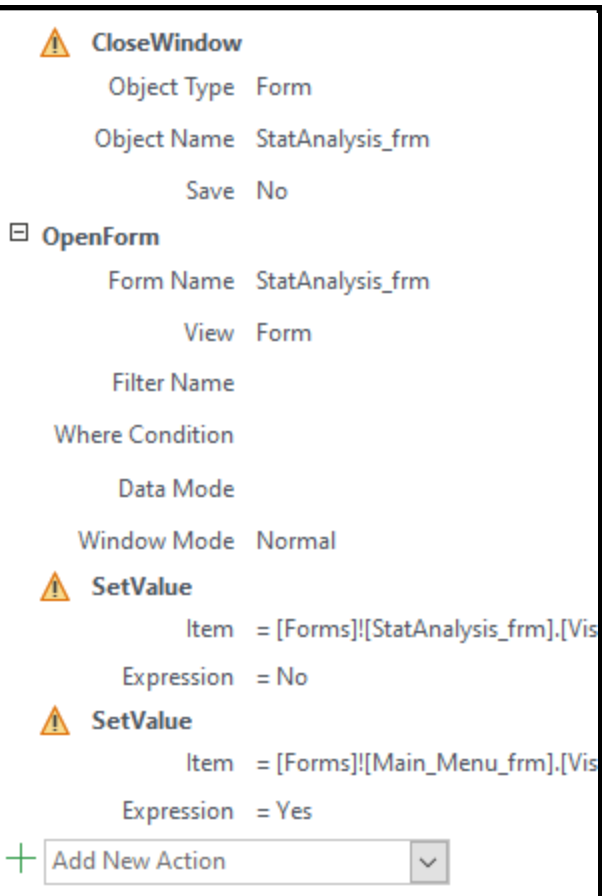
Dashboard_btn	N/A	VBA	On Click
Reset_statfrm_selection_btn	N/A	Reset_Stat_macro	On Click
Return_MainMenu_btn	N/A	VBA	On Click

8.3.1 Statistical Menu Form Macros:

Macro Name	Object Name	Macro
GetReadingEventOutput_macro	Get Reading Output_btn	

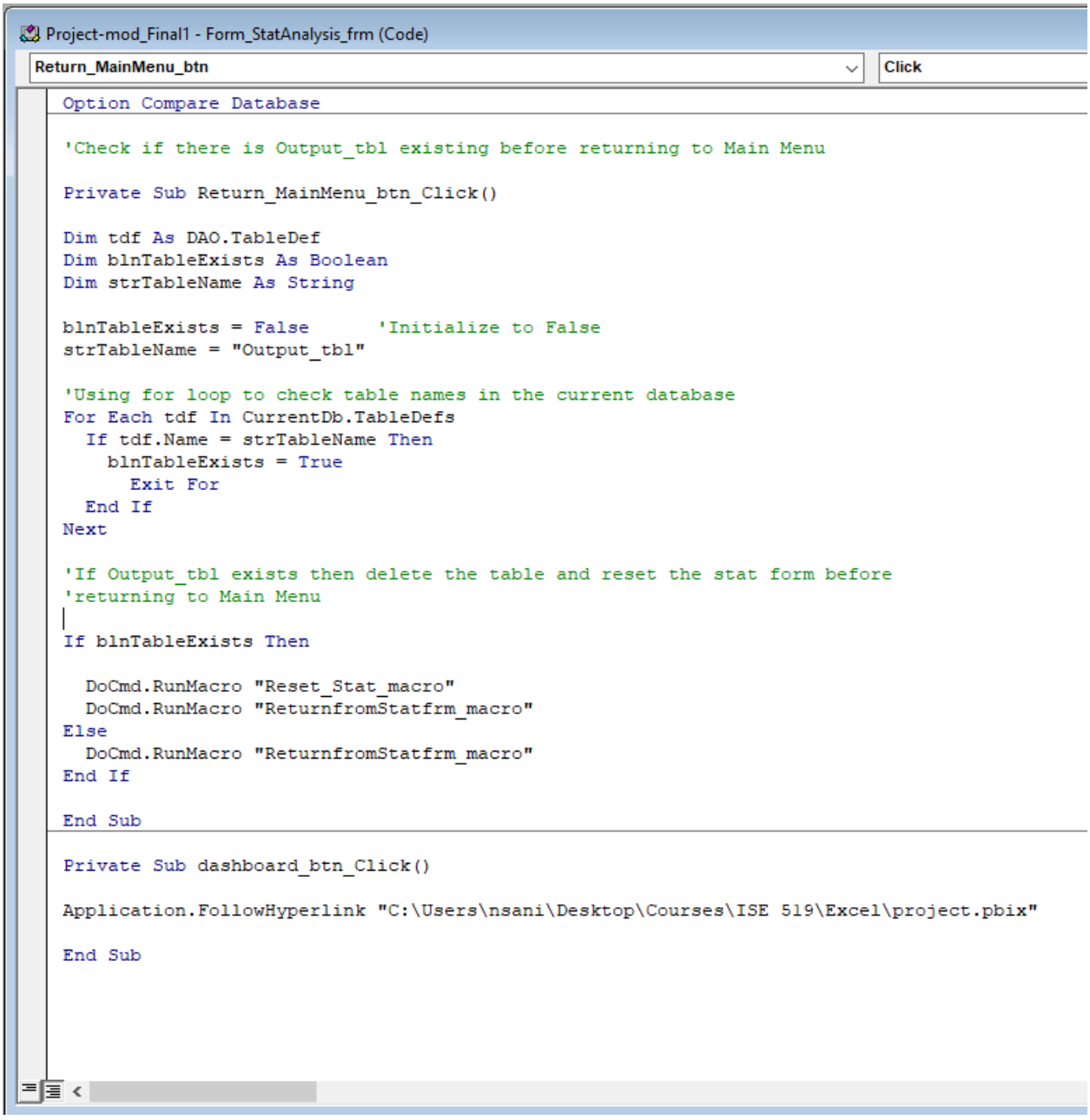
InsReadingtoTemp_macro	GetReadingEventOutput_macro	
Reset_Stat_macro	Reset Form Selection_btn	

		
ReturnfromStatfrm_macro	Return_MainMenu_btn	

		 CloseWindow Object Type Form Object Name StatAnalysis_frm Save No OpenForm Form Name StatAnalysis_frm View Form Filter Name Where Condition Data Mode Window Mode Normal SetValue Item = [Forms]![StatAnalysis_frm].[Vis Expression = No SetValue Item = [Forms]![Main_Menu_frm].[Vis Expression = Yes Add New Action

8.4 Statistical Menu Form VBA Code:

VBA Code for opening PowerBI dashboard for visualizing data upon clicking dashboard_btn and for returning to Main Menu upon clicking the Return_MainMenu_btn

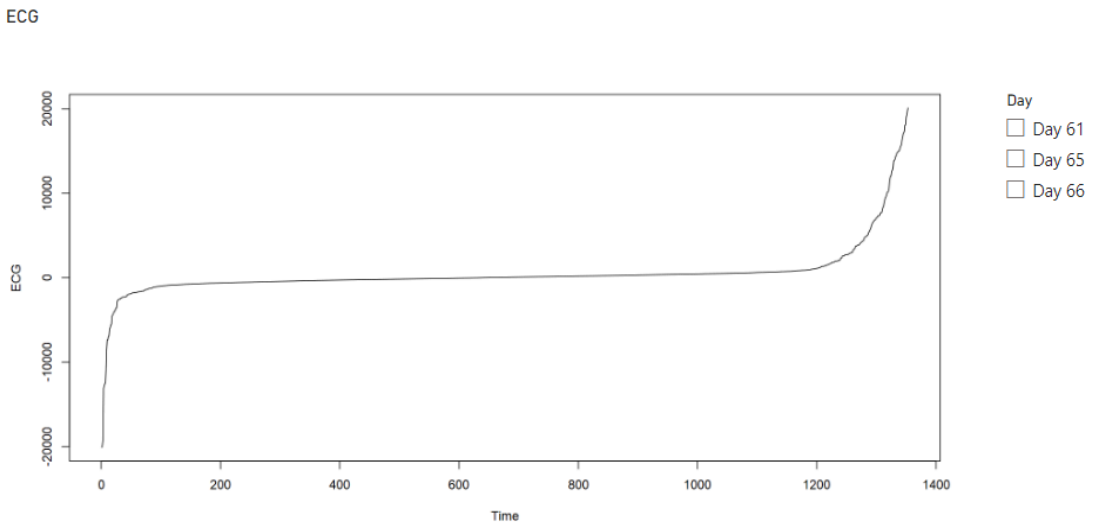



```
Option Compare Database

'Check if there is Output_tbl existing before returning to Main Menu

Private Sub Return_MainMenu_btn_Click()

Dim tdf As DAO.TableDef
Dim blnTableExists As Boolean
Dim strTableName As String

blnTableExists = False      'Initialize to False
strTableName = "Output_tbl"

'Using for loop to check table names in the current database
For Each tdf In CurrentDb.TableDefs
  If tdf.Name = strTableName Then
    blnTableExists = True
      Exit For
  End If
Next

'If Output_tbl exists then delete the table and reset the stat form before
'returning to Main Menu

If blnTableExists Then

  DoCmd.RunMacro "Reset_Stat_macro"
  DoCmd.RunMacro "ReturnfromStatfrm_macro"
Else
  DoCmd.RunMacro "ReturnfromStatfrm_macro"
End If

End Sub

Private Sub dashboard_btn_Click()

Application.FollowHyperlink "C:\Users\nsani\Desktop\Courses\ISE 519\Excel\project.pbix"

End Sub
```

9 PowerBI Dashboard

The PowerBI Dashboard is used to visualize the data; the data is extracted from the MS Access Database into PowerBI Dashboard. For each measurement, the data is visualized with filter which corresponds to the day and device options. The researchers can keep track of the readings according to their convenience. The dashboard contains R and Python Script, where it can use the libraries related to those scripts.

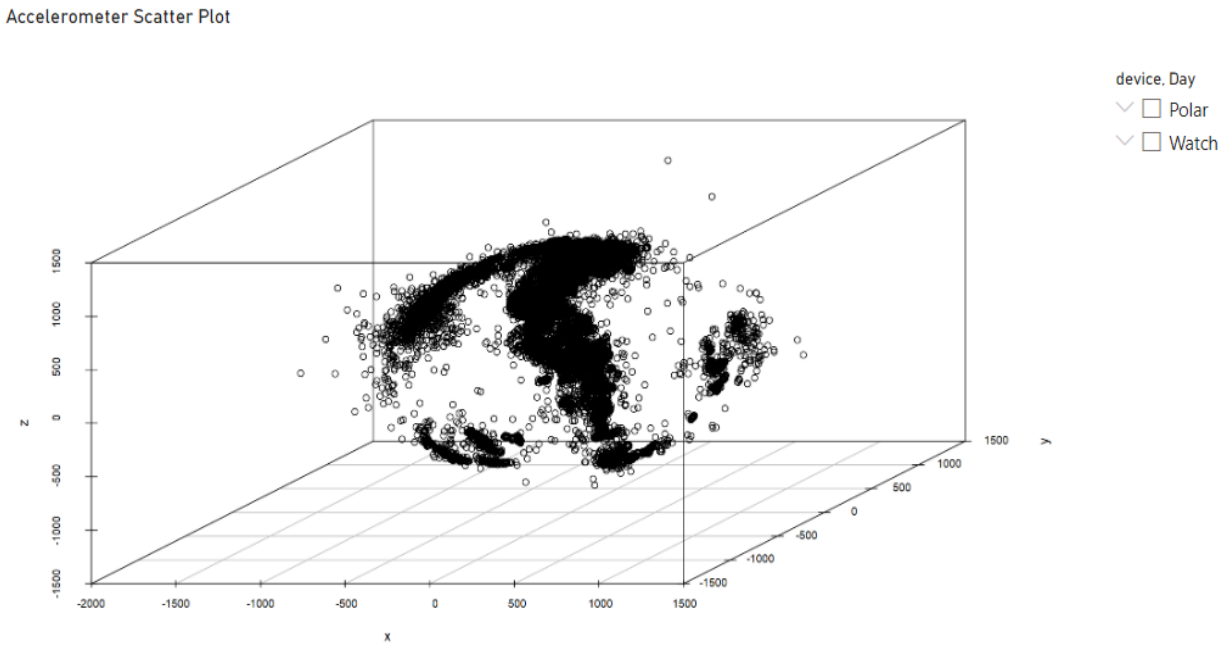
9.1 ECG



The R script program for ECG measurement:

```r
# The following code to create a dataframe and remove duplicated rows is always executed and acts as a preamble for your script:

# dataset <- data.frame(ECG)
# dataset <- unique(dataset)

# Paste or type your script code here:
plot.ts(dataset)
```

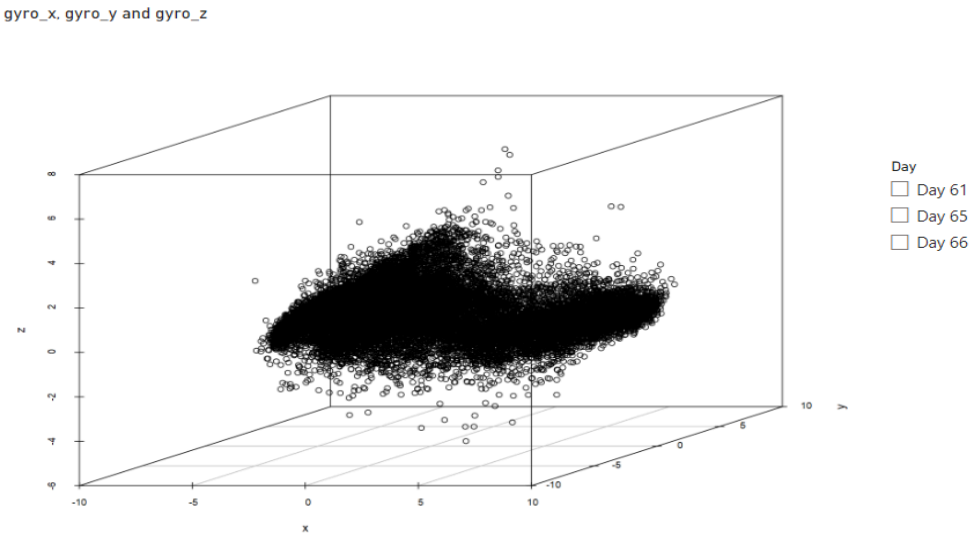
9.2 Accelerometer



The R script program for this accelerometer measurement:

```
# dataset <- data.frame(x)
# dataset <- data.frame(y)
# dataset ,- data.frame(z)
# dataset <- unique(dataset)

# Paste or type your script code here:
library(scatterplot3d)
x <- dataset$x
y <- dataset$y
z <- dataset$z

scatterplot3d(x,y,z)
```

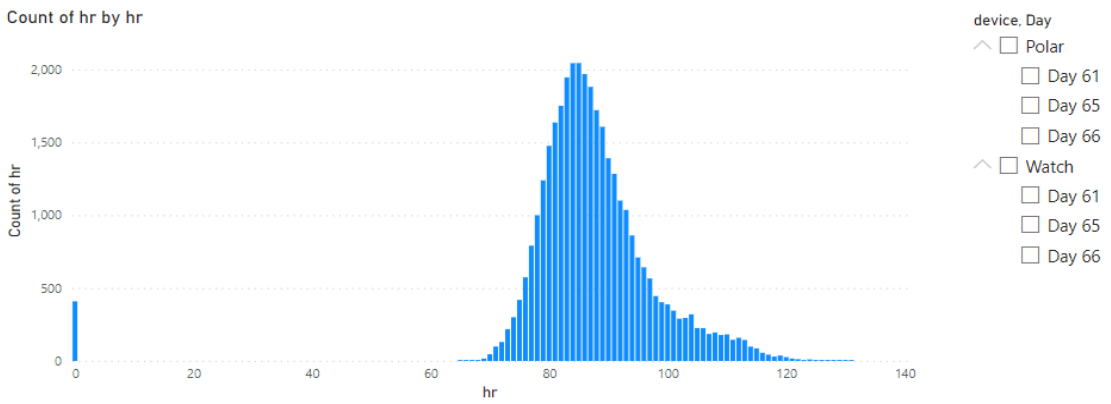
9.3 Gyroscope



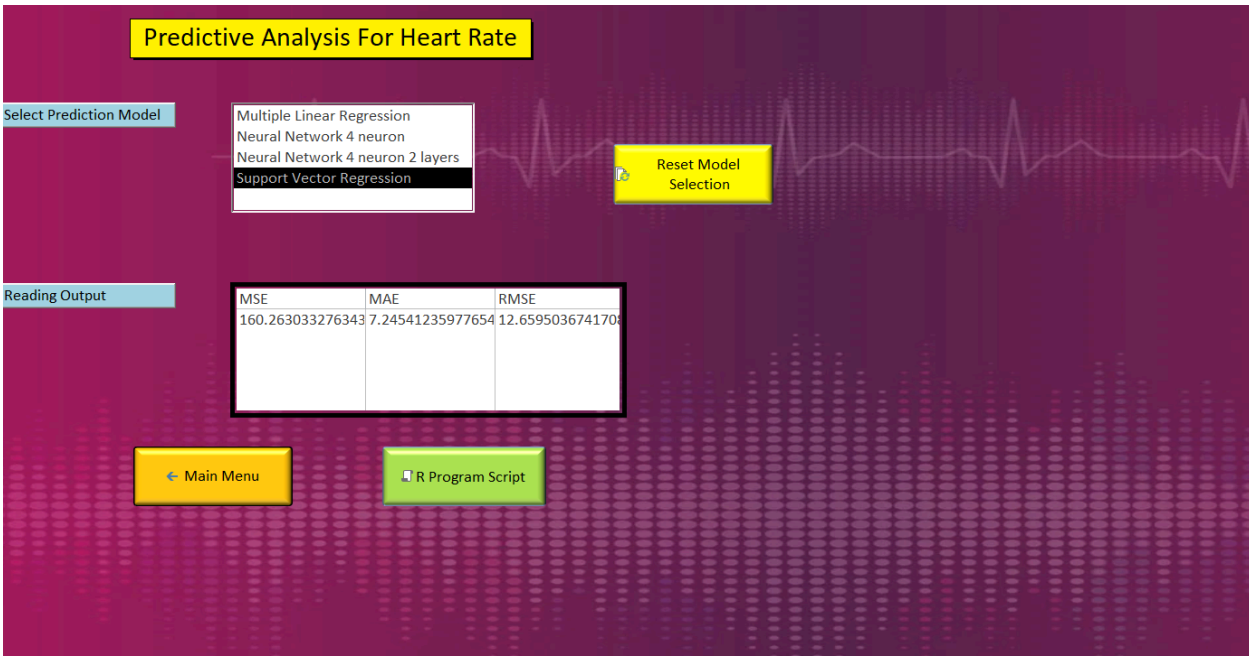
The R script program for Gyroscope measurement:

```
# The following code to create a dataframe and remove duplicated rows is always executed and acts as a preamble for your script:

# dataset <- data.frame(gyro_x, gyro_y, gyro_z)
# dataset <- unique(dataset)

# Paste or type your script code here:
library(scatterplot3d)
x <- dataset$gyro_x
y <- dataset$gyro_y
z <- dataset$gyro_z

scatterplot3d(x,y,z)
```

9.4 Heart Rate



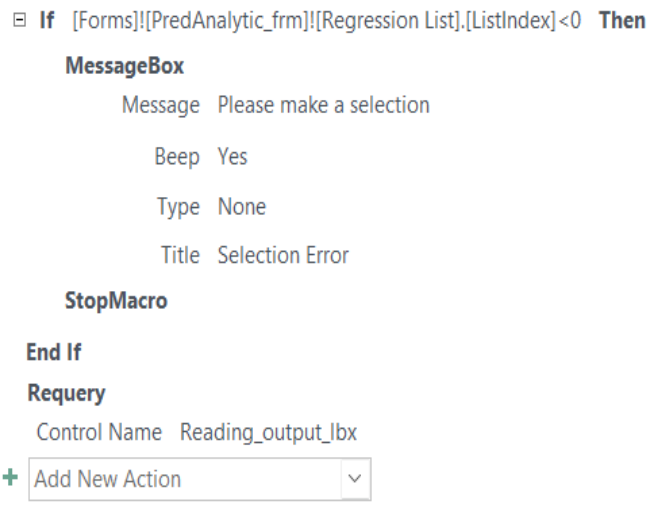
10 Predictive Analysis Form



Object Name	Query/Row Source	Macro	Event Property
Regression List	SELECT [Prediction_tbl].[Algorithm] FROM Prediction_tbl ORDER BY [Algorithm];	Pred_reading_macro	After Update

Reading_output_lbx	SELECT MSE, MAE, RMSE FROM Prediction_tbl WHERE Algorithm = forms![PredAnalytic_frm]![Regression List].Value;	N/A	N/A
Command8	N/A	Reset_Pred_Macro	On Click
Command10	N/A	ReturnfromPredAfrm_macro	On Click
R_btn	N/A	VBA	On Click

10.1 Predictive Analysis Form Macros

Macro Name	Object Name	Macro
Pred_reading_macro	Regression List	

Reset_Pred_Macro	Command 8	**CloseWindow** Object Type Form Object Name PredAnalytic_frm Save No **OpenForm** Form Name PredAnalytic_frm View Form Filter Name Where Condition Data Mode Window Mode Normal Add New Action
ReturnfromPredAfrm_ macro	Command 10	**CloseWindow** Object Type Form Object Name PredAnalytic_frm Save Prompt **OpenForm** Form Name PredAnalytic_frm View Form Filter Name Where Condition Data Mode Window Mode Normal **SetValue** Item = [Forms]![PredAnalytic_frm].[Visible] Expression = No **SetValue** Item = [Forms]![Main_Menu_frm].[Visible] Expression = Yes Add New Action

10.2 Predictive Analysis Form VBA Code

11 R Programming for Machine Learning

In the machine learning predictive analysis of data, we would be training and testing the following readings to determine the heart rate of the patient to check for the abnormality in heart rate conditions. We associate all the device measurements with the person's heart rate, so that we can find the perfect algorithm which can be able to correlate those readings with the heart rate. Therefore, to check for the perfect algorithm, we will use regression analysis of main metrics I.e., MSE (Mean Square Error), MAE (Mean Absolute Error), and RMSE (Root Mean Square Error). For this project, we have used 4 algorithms which are of the following: -

1. Multiple Linear Regression
2. Neural Network 4-neuron Single Layer
3. Neural Network 4-neuron Double Layer
4. Support Vector Regression

Multiple Linear Regression:

Multiple linear regression (MLR), often known as multiple regression, is a statistical technique that predicts the result of a response variable by combining numerous explanatory variables.
Multiple regression is a variant of linear (OLS) regression in which just one explanatory variable is used.

Neural Network Regression:
Neural networks are a set of algorithms that simulate the functions of a brain to recognize patterns in large volumes of data. As a result, they resemble the connections between neurons and synapses in the brain.

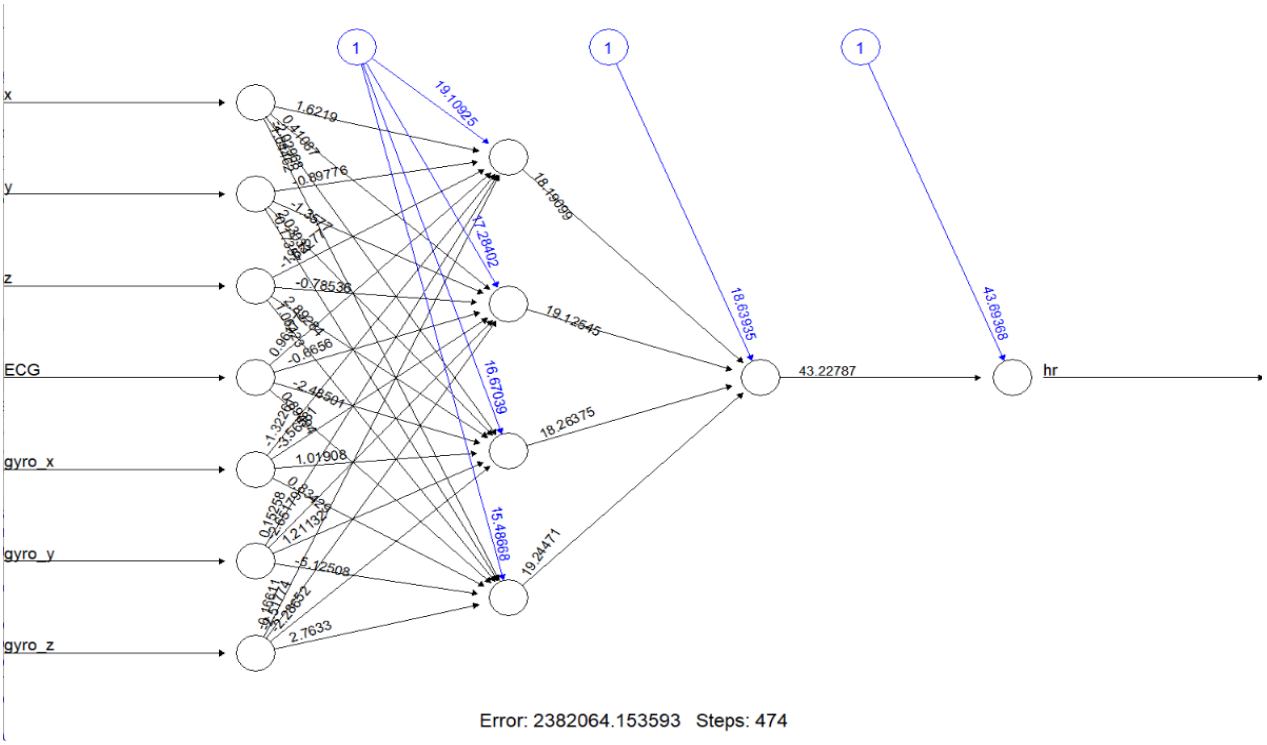
Support Vector Regression:
SVR allows us to indicate how many errors in our model are acceptable, and it will locate an appropriate line (or hyperplane in higher dimensions) to fit the data.

11.1 Database Connection

In R language, we use library obdc which will be able to establish connection with the MS Access database. From those databases, we can create data frames where we can run all machine algorithm regression.

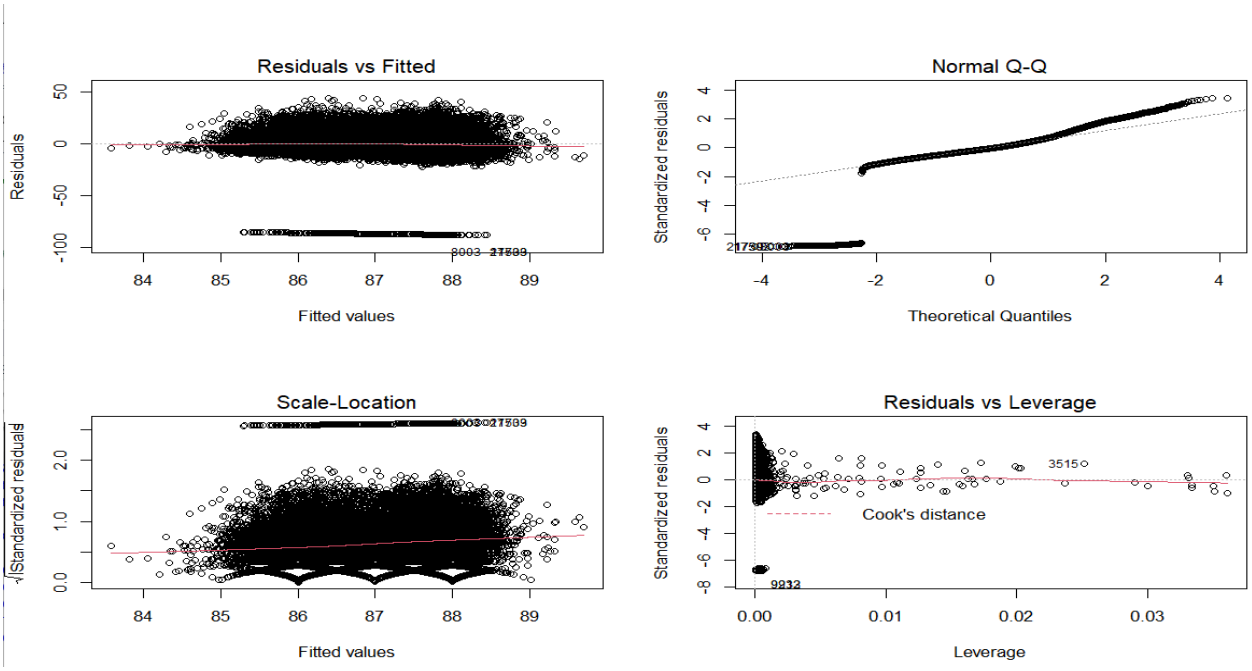
Plots of Regression Analysis

Neural Network Regression



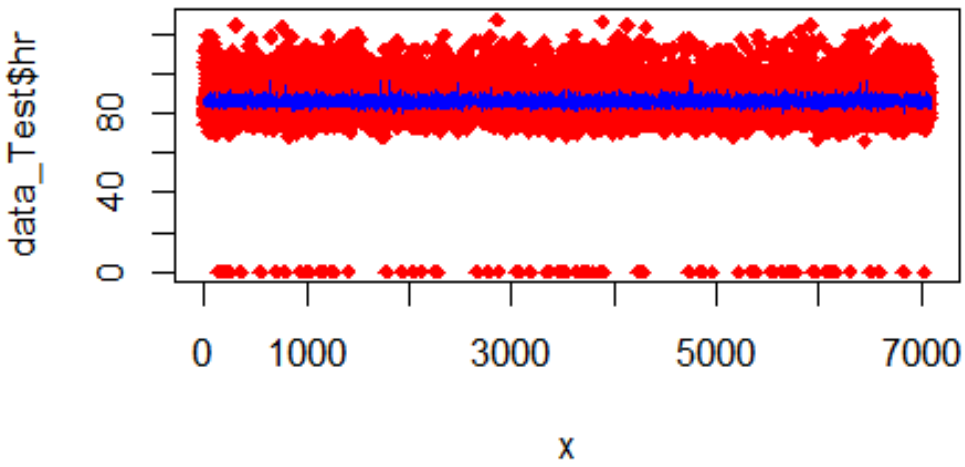
This 4 Neurons and 2-hidden layers. The epoch steps for the analysis are 474.

Multiple Linear Regression Analysis



The plots give us the understanding that how true values are closely related to predicted values. It also points out the outliers on the dataset.

Support Vector Regression Analysis



11.2 Statistical Significance of the Dataset for Machine Learning

Multiple Linear Regression Analysis

```
Residuals:
    Min      1Q  Median      3Q     Max
-88.441  -4.954  -0.540   5.214  44.599

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) 86.8629043  0.0913085 951.312  < 2e-16 ***
x           -0.0013488  0.0002135  -6.319 2.67e-10 ***
y            0.0002746  0.0001923   1.428   0.1534
z           -0.0001713  0.0001397  -1.227   0.2199
ECG          0.0000205  0.0001217   0.168   0.8663
gyro_x       0.0732552  0.0307656   2.381   0.0173 *
gyro_y      -0.0522955  0.0546358  -0.957   0.3385
gyro_z       0.5524075  0.1396314   3.956 7.63e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 12.93 on 28382 degrees of freedom
Multiple R-squared:  0.004332,  Adjusted R-squared:  0.004086
F-statistic: 17.64 on 7 and 28382 DF,  p-value: < 2.2e-16
```

Here we can see that the P-value of y, z, ECG, gyro_y is less than 0.05 which means that these variables do not form significant relation with the dependent variable Heart Rate. Therefore, from these values we can recommend to the researchers about the issues on the data collection process.

11.3 Prediction Data table

MSE	MAE	RMSE	Algorithm
152.7847	7.371819	12.36061	Neural Network 4 neuron
152.1750	7.338417	12.33592	Neural Network 4 neuron 2 layers
151.5925	7.315051	12.31229	Multiple Linear Regression
151.9013	7.161368	12.32482	Support Vector Regression

The following values are regression metrics on the testing data, so that we can know which algorithm can predict the testing data well and how well the values are closely related to the true values of the testing data.

12 Reference

https://bytes.com/topic/access/answers/800392-check-if-table-exists

13 Future Enhancements

- This model can further be developed for tracking other vital information of the patients like the diabetes and hypertension level and electrocardiogram rates of the heart.
- With the PowerBI dashboard functionality built this model can be used for capturing the live dashboard for any new entries added to the data model.
- To make timestamp series better, track device readings with timestamp Integrating MS Access dashboard with MS SQL server.
- Classification Analysis of data can be an extended scope of this model.