**Web Application:**

I had created all the tables locally using MySQL database and the updated java code for creating and uploading the data is present in the folder code/java code to load and create tables.

We have total of 32 tables and 2 views. Data in the csv format are present in the data folder.( for details pls check the web interface tables.pdf in documentation folder.

We are explicitly using Jquery and Jquery UI in the project to create the slider and the views. Others things are self explanatory. I am going to go over the few important ones.

**Script.php** and **FirstTry.php** are the two important php scripts that has almost all the backend code to authenticate the user, register the user, take survey and record the doctors response. We are explicitly using GET protocol to exchange data between the pages and the logics uses a lot of time the variables passed through it to perform many functionalities.

**modelResults\_v2.php** is called many times and it will create pages dynamically according to the values passed to it. Here we have used Jquery UI for the showing the linear gradient scale for the prediction values. For the different prediction factors different cutoffs are implemented.

$(function() {

var label = "<?php echo $outcomeTable?>";

startTimeScrn=Date.now();

$("#pbar").progressbar();

$("#pbar").bind('progressbarchange', function(event, ui) {

var value =this.getAttribute( "aria-valuenow" );

//console.log(value);

var selector = "#" + this.id + " > div";

//console.log(this.id);

if(label == "rifle7"){

colorcode(value,1 ,4);

}else if(label == "ICU"){

colorcode(value,35 , 47);

}else if(label== "ventilator"){

colorcode(value,13 ,40);

}else if(label=="cardioVascular"){

colorcode(value, 7,43);

}else if(label == "sepsis"){

colorcode(value, 6,35 );

}else if(label == "mortality"){

colorcode(value, 3, 41)

}else { colorcode(value,25 ,50 );

}

function colorcode( value ,cut1, cut2){

if (value >= cut2){

$(selector).css({ 'background': '#FF0000' });

} else if (value >= cut1){

$(selector).css({ 'background': '#FFFF00' });

}else{

$(selector).css({ 'background': '#00FF00' });

}

}

});

var val=parseInt($('#mcPredict').val());

$("#pbar").progressbar({ "value":val});

});

It also created two views as positive and negative views to store information temporarily and then using joins it fetches the required data to output on the screen.

// mysqli\_query($con,"create or replace view tempViewPositive as select outc.id ID,rank,var,description,ptDt.value value,weight from outcomeRank\_".$outcomeTable." outc, varDef, patientDetails ptDt where outc.id='".$ptID."' and outc.id=ptDt.ID and outc.weight > 0 and outc.var=varDef.id and ptDt.feature=outc.var and rank<=5 limit 5");

// mysqli\_query($con,"create or replace view tempViewNegative as select outc.id ID,rank,var,description,ptDt.value value,outc.weight\*-1 weight from outcomeRank\_".$outcomeTable." outc, varDef, patientDetails ptDt where outc.id='".$ptID."' and outc.id=ptDt.ID and outc.weight < 0 and outc.var=varDef.id and ptDt.feature=outc.var order by rank desc limit 5");

mysqli\_query($con,"create or replace view tempViewPositive as select outc.id ID,rank,var,description,ptDt.value value,weight from outcomeRank\_".$outcomeTable." outc, varDef, patientDetails ptDt where outc.id='".$ptID."' and outc.id=ptDt.ID and outc.weight > 0 and outc.var=varDef.type and ptDt.feature=outc.var and rank<=5 limit 5");

mysqli\_query($con,"create or replace view tempViewNegative as select outc.id ID,rank,var,description,ptDt.value value,outc.weight\*-1 weight from outcomeRank\_".$outcomeTable." outc, varDef, patientDetails ptDt where outc.id='".$ptID."' and outc.id=ptDt.ID and outc.weight < 0 and outc.var=varDef.type and ptDt.feature=outc.var order by rank desc limit 5");

The code marked in yellow was the change I made which was wrong before.

So the code is very modular and well typed so it is very easy to understand and change it.

In my server I have also implemented the SSL security using self signed certificate.