**INDEX**

[Preface 2](#_Toc115275281)

[ACKNOWLEDGEMENT 3](#_Toc115275282)

[Project Profile 4](#_Toc115275283)

[Hardware & Software Specification (SRS) 6](#_Toc115275284)

[Design 7](#_Toc115275285)

[Use Case Diagram 7](#_Toc115275286)

[Class diagram 9](#_Toc115275287)

[Data Flow Diagram 11](#_Toc115275288)

[Activity diagram 13](#_Toc115275289)

[Testing 15](#_Toc115275290)

[Implementing 16](#_Toc115275291)

[User Interface Designs (Snapshots) & Source Code 16](#_Toc115275292)

[Limitation & Feature Enhancement 43](#_Toc115275293)

[Conclusion 44](#_Toc115275294)

[Reference 45](#_Toc115275295)

[Bibliography 46](#_Toc115275296)

# Preface

The main objective of the project is to get the practical

experience of the current B.C.A. field.

We feel that such training for students like us is carrier-maker and also life-maker. As a part of system development,

got a valuable experience of various business processes, system

development methodology and strategies. This project is a part of fifth semester study, the final step toward the completion of B.C.A fifth semester.

In the end we like to say that by this project development we have tried to feel the gap between the theoretical and practical knowledge.

We are pleased to present the project report; proper care has been taking while organizing the report that is easy to comprehend.

# ACKNOWLEDGEMENT

It is out great pleasure to present our project report on **Calculator** which we conceived during the BCA 5th SEM affiliated to **Bhakta Kavi Narsinh Mehta University, Junagadh (Gujarat).**

We take this opportunity to express our sincere gratitude and we feel immense pleasure to thank our faculty, philosopher and guide **Dr. Bhavesh Lukka** who helped us and gave full support in each and every way to fulfill and accomplish our project of **Calculator.**

We are indebted to our college **Dr. V. R. Godhaniya College of I.T** our principal **Dr. Sanjay Agal** and Director **Mr. Dhaval Kher** for given us an excellent chance to prove our best work and efforts.

We are mostly thankful to them, for giving us the inspiration and dedication to successfully complete this project. Finally, we thanks to all who help us directly or indirectly in our project.

- Darshil Solanki

- Manoj Salet

# Project Profile

**Project Title:**

**Calculator** Application

**Project Information:**

This Project is used for Calculation of any simple and Scientific Calculation. With Amazing and Powerful Graphic User Interface.

**Institution:**

Dr V. R. Godhaniya IT College, Porbandar

**Programing Language:**

 C#

**Project Time Duration:**

three months

**Summary:**

Calculator Application is Provide to Calculation of Simple and Scientific Equation. This application uses a Graphical user interface so, this feature is provided user friendly interface. User can use this application in Popup Window with Transparent Controller. User can see history of own calculation with history feature.

So, there are three (3) sub modules:

 **Popup**

 **History**

 **Help**

**Guided By:**

**Dr. Bhavesh Lukka**

**Prepared By:**

Salet Manoj Ramjibhai

Solanki DarshilKumar Mansukhbhai

# Hardware & Software Specification (SRS)

* Hardware Specification

- 1 GHz or faster processor

- 50 MB of RAM

- 50 MB of hard disk

* Software Specification
* .NET Framework 4.5 or higher version
* Adobe Acrobat DC
* Windows Vista SP2 or higher version

# Design

## Use Case Diagram

* **What is use case diagram?**

A use case diagram is a simplest representation of User’s interaction with the system that shows the relationship between the user and the different use case in which the user is involved.

A use case diagram is a methodology used in system analysis to identify, clarify and organize system requirements.

In this context, the term “SYSTEM” refers to something being developed or operated such as a Calculation or PopUp Window etc.

Use case diagram are employed in UML (Unified Modeling Language) a standard notation for the modeling of real-world objects and system.



## Class diagram

* **What is Class Diagram?**

A class diagram is used to support functional requirement of system.

It contains sets of classes, interfaces, relationships, dependency (need), and generalization (overview) and association (link) relationship.

The notation for classes and the relationship between classes is shown as:





**: FileInfo**

## Data Flow Diagram

DFD graphically representing the functions, or processes, which capture, manipulate, store, and distribute data between a system and its environment and between components of a system.

The visual representation makes it a good communication tool between User and System designer. Structure of DFD allows starting from a broad overview and expand it to a hierarchy of detailed diagrams. DFD has often been used due to the following reasons:

* Logical information flow of the system
* Determination of physical system construction requirements
* Simplicity of notation
* Establishment of manual and automated systems requirements

**Symbols:** 

Data Flow Diagram



## Activity diagram

* **What is Activity Diagram?**

* Activity diagram is basically a flow chart to represent the flow form one activity to another activity. The activity can be described as an operation of the system.
* The control flow is drawn from one operation to another. This flow can be sequential Branched or concurrent.
* Activity diagram deals with all types of flow control by using different element like fork join etc.
* The basic purposes of activity diagrams are similar to other four diagram. It captures the dynamic behavior of the system.



# Testing

Before actually implementing the new system into operations, a test run of the system is done removing all the bugs, if any. It is an important phase of a successful system. After codifying the whole programs of the system, a test plan should be developed and run-on a given set of test data. The output of the test run should match the expected results. Sometimes, system testing is considered as a part of implementation process.

**Using the test data following test run are carried out:**

* **Program test:**

When the programs have been coded and compiled and brought to working conditions, they must be individually tested with the prepared test data. All verification and validation be checked and any undesirable happening must be noted and debugged (error Corrected)

* **System test:**

After carrying out the program test for each of the programs of the system and errors removed, then system test is done.

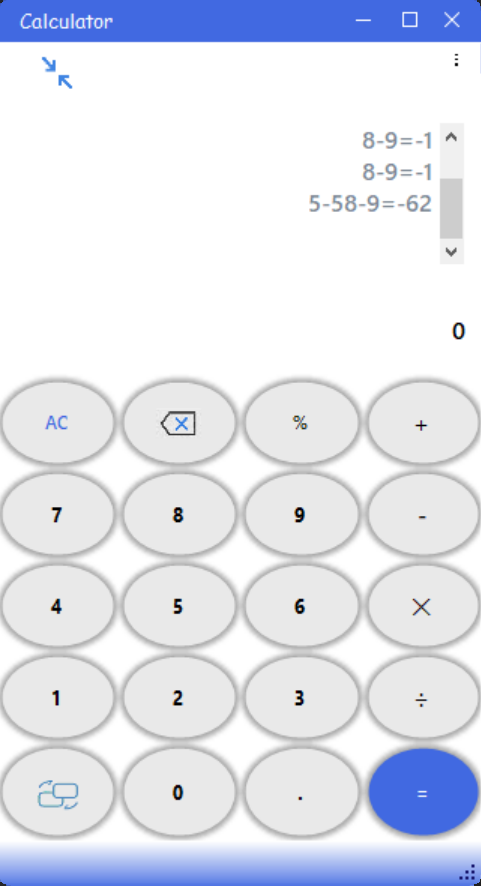
* **Peer testing:**

After testing of all the test case, we have created we gave our project to our classmates for peer testing that help us to remove run time exception and bugs

# Implementing

# User Interface Designs (Snapshots) & Source Code

**Home.cs**



using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.IO;

using org.mariuszgromada.math.mxparser;

using System.Configuration;

using Guna.UI2.WinForms;

namespace Calculator

{

public partial class Home : Form

{

//Declaration

private static StreamReader sr;

private static StreamWriter sw;

private static FileInfo file;

public static string data;

private int f; //flag for change button

private int fInv = 1; //flag for inverse and radian

public Home()

{

InitializeComponent();

}

private void fileread()

{

try

{

file = new FileInfo(ConfigurationSettings.AppSettings["Path"]);

sr = file.OpenText();

if (file.Length != 0)

{

txtres.AppendText(sr.ReadToEnd());

}

sr.Dispose();

}

catch (FileNotFoundException e)

{

}

catch (IOException e)

{

MessageBox.Show(e.ToString());

}

}

private void filewrite()

{

try

{

if (txtres.Text != null)

{

if (txtshow.Text.Contains("="))

{

txtres.Text += txtshow.Text + Environment.NewLine;

txtshow.Text = "0";

}

sw = file.CreateText();

string[] data = txtres.Text.Split(Environment.NewLine.ToCharArray(), StringSplitOptions.RemoveEmptyEntries);

string hist = "";

int l = data.Length;

if (l > 5)

{

for (int i = l - 5; i < l; i++)

{

hist += data[i] + "\r\n";

}

}

else

{

for (int i = 0; i < l; i++)

{

hist += data[i] + "\r\n";

}

}

sw.Write(hist);

}

else if (txtres.Text == null && file.Length != 0)

{

sw.Write("");

}

sw.Dispose();

}

catch (IOException)

{

}

}

private void Home\_Load(object sender, EventArgs e)

{

fileread();

tblpanel.ColumnStyles[4].SizeType = SizeType.Absolute;

tblpanel.ColumnStyles[4].Width = 0;

tblpanel.RowStyles[3].SizeType = SizeType.Absolute;

tblpanel.RowStyles[3].Height = 0;

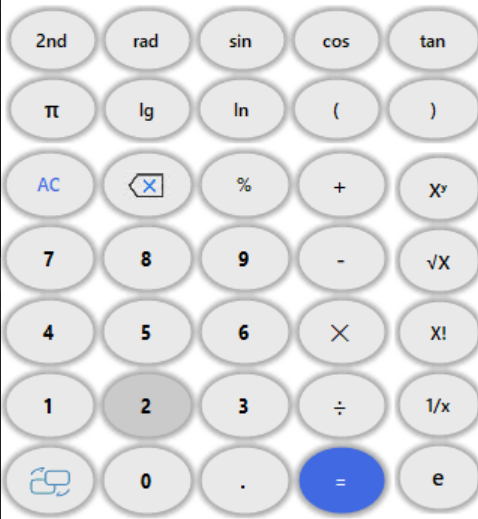
tblpanel.RowStyles[4].SizeType = SizeType.Absolute;

tblpanel.RowStyles[4].Height = 0;

f = 1;

}

**Scientific Button Click:**

** **

private void btnchange\_Click(object sender, EventArgs e)

{

if (f == 0)

{

tblpanel.RowStyles[3].SizeType = SizeType.Absolute;

tblpanel.RowStyles[3].Height = 0;

tblpanel.RowStyles[4].SizeType = SizeType.Absolute;

tblpanel.RowStyles[4].Height = 0;

uptblpanel.Visible = false;

tblpanel.ColumnStyles[4].SizeType = SizeType.Absolute;

tblpanel.ColumnStyles[4].Width = 0;

sidetblpanel.Visible = false;

f = 1;

}

else

{

tblpanel.ColumnStyles[4].SizeType = SizeType.Percent;

tblpanel.ColumnStyles[4].Width = 20;

sidetblpanel.Visible = true;

tblpanel.RowStyles[4].SizeType = SizeType.Percent;

tblpanel.RowStyles[4].Height = 10.0f;

tblpanel.RowStyles[5].SizeType = SizeType.Percent;

tblpanel.RowStyles[5].Height = 10.0f;

uptblpanel.Visible = true;

for (int i = 0; i < 10; i++)

{

tblpanel.RowStyles[i].SizeType = SizeType.Percent;

tblpanel.RowStyles[i].Height = 10.00f;

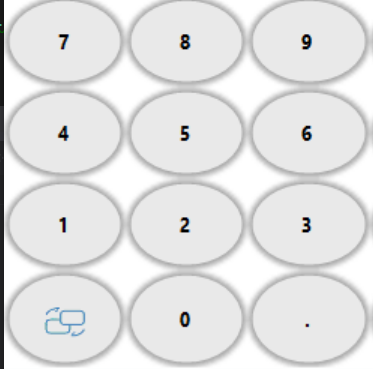
}

f = 0;

}

}

**Number Click:**

****

private void number\_click(object sender, EventArgs e)

{

Guna2CircleButton b = (Guna2CircleButton)sender;

Functions.settext(b.Text,this.txtshow,this.txtres);

}

**Arithmetic Click:**



private void arop\_click(object sender, EventArgs e)

{

Guna2CircleButton b = (Guna2CircleButton)sender;

string c = b.Text;

if (c == "X!")

Functions.setarop("!",this.txtshow,this.txtres);

else if (c == "Xʸ")

Functions.setarop("^", this.txtshow, this.txtres);

else if (c == "√X")

{

if (txtshow.Text == "0")

txtshow.Text = "";

Functions.setarop("√", this.txtshow, this.txtres);

}

else if (c == "1/x")

Functions.setarop("^(-1)", this.txtshow, this.txtres);

else if (c == "%")

{

if (txtshow.Text != "0")

Functions.setarop(c,this.txtshow,this.txtres);

}

else

Functions.setarop(c, this.txtshow, this.txtres);

}

**TextBox Text:**

****

private void txtshow\_TextChanged(object sender, EventArgs e)

{

Functions.txtchanged(this.txtshow,this.btnac);

}

**Clear Button Click:**

****

private void btnc\_Click(object sender, EventArgs e)

{

Functions.clear(this.txtshow);

}

**Clear All Button Click:**



private void btnac\_Click(object sender, EventArgs e)

{

Functions.allclear(this.txtshow, this.txtres, this.btnac);

}

**Equal Button Click:**



private void btnequal\_Click(object sender, EventArgs e)

{

Functions.result(this.txtshow,this.txtres);

}

**Trigno Click:**

****

private void trigno\_click(object sender, EventArgs e)

{

Guna2CircleButton b = (Guna2CircleButton)sender;

if (txtshow.Text == "0")

{

txtshow.Text = b.Text + "(";

}

else if (txtshow.Text.Contains("="))

{

txtres.Text += txtshow.Text + "\r\n";

txtshow.Text = b.Text + "(";

}

else

{

txtshow.Text += b.Text + "(";

}

}

**Brackte Button Click:**



private void brackate\_click(object sender, EventArgs e)

{

Guna2CircleButton b = (Guna2CircleButton)sender;

if (txtshow.Text == "0")

{

txtshow.Text = b.Text;

}

else if (!txtshow.Text.Contains("="))

{

txtshow.Text += b.Text;

}

}

**Const Button Click:**



private void const\_click(object sender, EventArgs e)

{

Guna2CircleButton b = (Guna2CircleButton)sender;

if (txtshow.Text == "0")

{

txtres.Text += b.Text;

if (b.Text == "π")

txtshow.Text = "=" + Math.PI.ToString();

else

txtshow.Text = "=" + Math.E.ToString();

}

else if (txtshow.Text.Contains("="))

{

txtres.Text += txtshow.Text + Environment.NewLine;

txtshow.Text = b.Text;

}

else

{

txtshow.Text += b.Text;

}

}

**Red Button Click:**



private void btnraddeg\_Click(object sender, EventArgs e)

{

Guna2CircleButton b = (Guna2CircleButton)sender;

if (b.Text == "rad")

{

btnraddeg.Text = "deg";

mXparser.setDegreesMode();

btninv.Enabled = false;

}

else

{

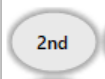
btnraddeg.Text = "rad";

btninv.Enabled = true;

}

}

**Inv Button Click:**



private void btninv\_Click(object sender, EventArgs e)

{

if (fInv == 1)

{

btnraddeg.Enabled = false;

mXparser.setRadiansMode();

btnsin.Text = "asin";

btncos.Text = "acos";

btntan.Text = "atan";

fInv = 0;

}

else

{

btnraddeg.Enabled = true;

btnsin.Text = "sin";

btncos.Text = "cos";

btntan.Text = "tan";

fInv = 1;

}

}

**Keypress From External Keyboard:**



private void form\_keypress(object sender, KeyPressEventArgs e)

{

if (e.KeyChar == 27)

{

this.Close();

}

else if ((e.KeyChar >= 48 && e.KeyChar <= 57) || e.KeyChar == 46)

{

Functions.settext(e.KeyChar.ToString(),this.txtshow,this.txtres);

}

else if (e.KeyChar == 42 || e.KeyChar == 43 || e.KeyChar == 45 || e.KeyChar == 47)

{

if (e.KeyChar == 42)

{

Functions.setarop("⨉", this.txtshow, this.txtres);

}

else if (e.KeyChar == 47)

{

Functions.setarop("÷", this.txtshow, this.txtres);

}

else

{

Functions.setarop(e.KeyChar.ToString(), this.txtshow, this.txtres);

}

}

else if (e.KeyChar == 40 || e.KeyChar == 41)

{

if (txtshow.Text == "0")

{

txtshow.Text = e.ToString();

}

else if (!txtshow.Text.Contains("="))

{

txtshow.Text += e.ToString();

}

}

else if (e.KeyChar == 61)

{

Functions.result(this.txtshow,this.txtres);

}

else if (e.KeyChar == 8)

{

Functions.clear(this.txtshow);

}

}

private void form\_close(object sender, FormClosingEventArgs e)

{

filewrite();

}

**Popup Button Click:**



private void btnpop\_Click(object sender, EventArgs e)

{

this.Hide();

if (txtshow.Text.Contains("="))

{

txtres.AppendText(txtshow.Text + Environment.NewLine);

txtshow.Text = "0";

}

data = txtres.Text;

Popup d = new Popup();

d.ShowDialog();

d.Dispose();

txtres.Text=data;

txtres.Select(txtres.TextLength-1, 0);

txtres.ScrollToCaret();

this.Show();

}

**Menu Button Click:**

****

private void tab\_click(object sender, EventArgs e)

{

topgradientpanel.Size = new Size(this.Width, 95);

toptoppanel.Size = new Size(this.Width, 29);

topbottompanel.Size = new Size(this.Width, 65);

topbottompanel.Dock = DockStyle.None;

topbottomtblpanel.RowStyles[0].Height = 50.0f;

topbottomtblpanel.RowStyles.Add(new RowStyle(SizeType.Percent,50.00f));

topbottomtblpanel.SetRowSpan(guna2TabControl1, 2);

guna2TabControl1.Size = new Size(47, 68);

}

private void tab\_leave(object sender, EventArgs e)

{

topbottomtblpanel.SetRowSpan(guna2TabControl1, 1);

guna2TabControl1.Size = new Size(47, 20);

topbottomtblpanel.RowStyles[0].Height = 100.0f;

topbottompanel.Size = new Size(this.Width, 40);

topbottompanel.Dock = DockStyle.Bottom;

topgradientpanel.Size = new Size(this.Width, 70);

}

**When you click the help link in menu item:**

private void help\_click(object sender, EventArgs e){

if (txtshow.Text.Contains("=")){

txtres.AppendText(txtshow.Text + Environment.NewLine);

txtshow.Text = "0";

}

Functions.gotohelp(this);

}

**When you click the hist link in menu item:**

private void hist\_click(object sender, EventArgs e){

this.Hide();

if (txtshow.Text.Contains("="))

{

txtres.AppendText(txtshow.Text + Environment.NewLine);

txtshow.Text = "0";

}

data = txtres.Text;

history h = new history();

h.ShowDialog();

h.Dispose();

this.Show();

}

private void frm\_keydown(object sender, KeyEventArgs e){

if (Convert.ToInt16(e.KeyCode) == 112){

if (txtshow.Text.Contains("=")){

txtres.AppendText(txtshow.Text + Environment.NewLine);

txtshow.Text = "0";

}

Functions.gotohelp(this);

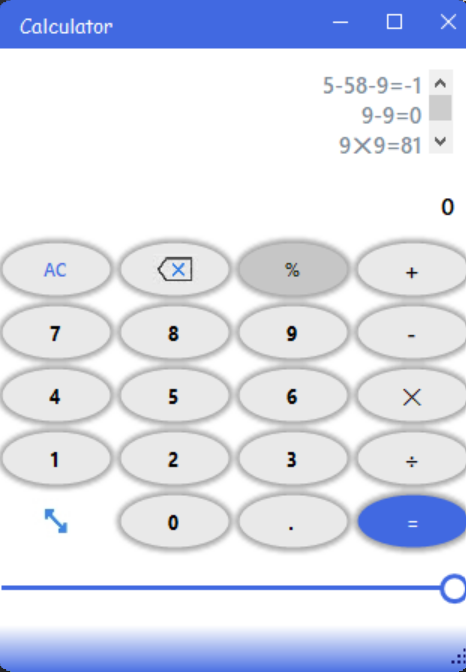
}

}

}

}

**Popup.cs**

****

using Guna.UI2.WinForms;

using org.mariuszgromada.math.mxparser;

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.IO;

using System.Configuration;

namespace Calculator

{

public partial class Popup : Form

{

public Popup()

{

InitializeComponent();

}

private void number\_click(object sender, EventArgs e)

{

Guna2CircleButton b = (Guna2CircleButton)sender;

Functions.settext(b.Text,this.txtshow,this.txtres);

}

private void btnequal\_Click(object sender, EventArgs e)

{

Functions.result(this.txtshow,this.txtres);

}

private void btnc\_Click(object sender, EventArgs e)

{

Functions.clear(this.txtshow);

}

private void btnac\_Click(object sender, EventArgs e)

{

Functions.allclear(this.txtshow,this.txtres,this.btnac);

}

private void arop\_click(object sender, EventArgs e)

{

Guna2CircleButton b = (Guna2CircleButton)sender;

string c = b.Text;

if (c == "%")

{

if (txtshow.Text != "0")

Functions.setarop(c,this.txtshow,this.txtres);

}

else

Functions.setarop(c,this.txtshow,this.txtres);

}

**Popup Close Button:**



private void btnpop\_Click(object sender, EventArgs e)

{

this.Close();

}

**Transparent Window Trackbar:**



private void guna2TrackBar1\_Scroll(object sender, ScrollEventArgs e)

{

this.Opacity=(double)transparencybar.Value/100.0;

}

private void txtshow\_TextChanged(object sender, EventArgs e)

{

Functions.txtchanged(this.txtshow, this.btnac);

}

private void Popup\_Load(object sender, EventArgs e)

{

txtres.AppendText(Home.data);

}

private void Popup\_KeyPress(object sender, KeyPressEventArgs e)

{

if (e.KeyChar == 27)

{

this.Close();

}

else if ((e.KeyChar >= 48 && e.KeyChar <= 57) || e.KeyChar == 46)

{

Functions.settext(e.KeyChar.ToString(), this.txtshow, this.txtres);

}

else if (e.KeyChar == 42 || e.KeyChar == 43 || e.KeyChar == 45 || e.KeyChar == 47)

{

if (e.KeyChar == 42)

{

Functions.setarop("⨉",this.txtshow,this.txtres);

}

else if (e.KeyChar == 47)

{

Functions.setarop("÷",this.txtshow,this.txtres);

}

else

{

Functions.setarop(e.KeyChar.ToString(), this.txtshow, this.txtres);

}

}

else if (e.KeyChar == 61)

{

Functions.result(this.txtshow, this.txtres);

}

else if (e.KeyChar == 8)

{

Functions.clear(this.txtshow);

}

}

private void demo\_FormClosing(object sender, FormClosingEventArgs e)

{

if (txtshow.Text.Contains("="))

{

txtres.AppendText(txtshow.Text + Environment.NewLine);

txtshow.Text = "0";

}

Home.data = txtres.Text;

}

private void moved(object sender, EventArgs e)

{

this.Opacity = (double)transparencybar.Value / 100.0;

}

private void popup\_keydown(object sender, KeyEventArgs e)

{

if (Convert.ToInt16(e.KeyCode) == 112)

{

if (txtshow.Text.Contains("="))

{

txtres.AppendText(txtshow.Text + Environment.NewLine);

txtshow.Text = "0";

}

Functions.gotohelp(this);

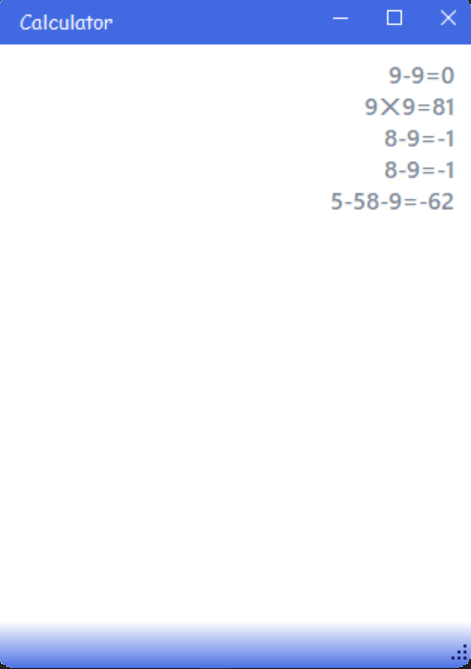
}

}

}

}

**History.cs**



using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Calculator

{

public partial class history : Form

{

public history()

{

InitializeComponent();

}

private void history\_load(object sender, EventArgs e)

{

txtres.AppendText(Home.data);

}

private void history\_keypress(object sender, KeyPressEventArgs e)

{

if(e.KeyChar==27)

this.Close();

}

}

}

**help.cs**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Configuration;

using System.IO;

namespace Calculator

{

public partial class help : Form

{

FileInfo file;

public help()

{

InitializeComponent();

}

private void help\_load(object sender, EventArgs e)

{

file = new FileInfo(ConfigurationSettings.AppSettings["Pathpdf"]);

axAcroPDF1.src = file.FullName + "#toolbar=0";

}

private void help\_keypress(object sender, KeyPressEventArgs e)

{

if (e.KeyChar == 27)

this.Close();

}

}

}

Limitation & Feature Enhancement

* Limitation

-Dependency on adobe

-Requires a good graphic processing power

* Feature Enhancement

-creating own pdf reader remove dependency on adobe.

-creating and replacing high graphic component with low graphic component with same functionality and appearance.

* Scalability

-Dark mode

-Converter ex. Length converter

-providing local unit of measurement as per region

-Finance related calculation module ex. Simple interest calculation, FD Calculator etc..

-More better user interface and accessibility

# Conclusion

It was a wonderful learning experience for us while working on this project. This project took us through the various phases of project development and gave us real insight into the world of software engineering. The joy of working and the thrill involved while tackling the various problems and challenges gave us a feel of the developers’ industry.

It was due to this project we came to know how professional software is designed. An experience of

practical project gave us idea of project management, resource, human resource, time and cost management.

# Reference

* <https://www.nuget.org/packages/Guna.UI2.WinForms/>
* <https://www.nuget.org/packages/MathParser.org-mXparser/>
* <https://www.revenera.com>
* Physical Project Documentation

# Bibliography

* Guna UI
* Math Parser
* Adobe
* InstallShield 2022
* Xiaomi Calculator
* Windows Calculator
* <https://bootcamp.uxdesign.cc/how-i-designed-published-subtle-a-gorgeous-calculator-3f482d11a80a>
* Physical Project Documentation
* <https://www.tutorialspoint.com/csharp/csharp_text_files.htm>