

Experiment 2:

Design and validate login form

Aim:

Implementation of Data Validation of
Login form

Theory:

Form validation is a technique of ensuring that the data entered into a user form meets certain criteria before it is submitted to the server. This process helps to maintain data integrity, enhance user experience, and prevent errors or malicious inputs. Form validation can be divided into two main types: Client-side validation and server-side validation.

Handwritten notes at the top of the page, including the word "stabilization" and other illegible scribbles.

EPARLE KELA
IAI MAF

ENGINEER

Visi
environment

Miss
tools and
developme

nts in con

moral and

ducation

es (PEOs)
lve comp

ng by in

ork and

n Specif

ndament
gy, Micro
l develop
ge Open
dent gre

A hand-drawn login form on lined paper. It features two input fields: one labeled "Username" and one labeled "Password". Below these fields are two buttons: "Submit" and "Cancel". A red diagonal line is drawn across the bottom right portion of the form, crossing over the "Cancel" button.

Procedure:

Experiment 2A

```

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 Exp2A
```

```

{
    public partial class Form1: Form
    {
        public Form1()
        {
            InitializeComponent();
        }
        private void button1_Click(object sender
            Event args e)
        {
            string username;
            string password;
            username = textbox1.Text;
            Password = textbox2.Text;

```

```
if (string.IsNullOrEmpty(username))  
{
```

```
    MessageBox.Show("Please input proper  
    name");
```

```
    Return;
```

```
}
```

```
if (username.Contains(" ")) {
```

```
    MessageBox.Show("Username cannot  
    contain whitespace");
```

```
}
```

```
if (password.Contains(" "))
```

```
{
```

```
    MessageBox.Show("Password cannot  
    contain whitespace");
```

```
}
```

```
if (string.IsNullOrEmpty(password))  
{
```

```
    MessageBox.Show("Enter proper password");  
    Return;
```

```
}
```

```
if (password.Length < 14 || password.Length > 8) {
```

```
    MessageBox.Show("Password cannot be  
    more than 14 characters and less than  
    8 characters");
```

```
    return;
```

```
}
```

```
if (!password.Any(char.IsUpper)) {
```

```
    MessageBox.Show("At least one character  
    should be upper case"); return; }
```

if (!password.Any (char. IsLower))
{

Message Box.Show ("Atleast one character
should be lower case");

return;

}

Message Box.Show ("Login Successful");

}

}

}

(password: admin) (username: admin)

Username:

Password:

Image:

Experiment 2B

```

namespace Exp2B
{
    public partial class Form1: Form
    {
        static int attempt = 3;
        public Form1()
        {
            InitializeComponent();
        }
        private void button1_Click(object sender,
            EventArgs e)
        {
            if (this.textBox1.Text == "Rashi" &&
                this.textBox2.Text == "Rashi@123")
            {
                attempt = 0;
                pictureBox1.SizeMode = PictureBoxSizeMode.
                    StretchImage;
                pictureBox1.Image = Image.FromFile("C:\
                    users\desktop\
                    Login\success.png");
                label3.Text = "Access Granted";
                MessageBox.Show("You are granted with
                    access");
            }
            else if (attempt <= 3 && attempt > 0)
            {

```

```
pictureBox1.SizeMode = PictureBoxSizeMode.  
StretchImage;
```

```
pictureBox1.Image = Image.FromFile(@"C:\Users\  
Desktop\Retry.png");
```

```
MessageBox.Show("You have only " +  
Convert.ToString(Attempt) +  
" attempt left to try");
```

```
-- attempt;
```

```
}
```

```
else
```

```
{
```

```
pictureBox1.SizeMode = PictureBoxSizeMode.  
StretchImage;
```

```
pictureBox1.Image = Image.FromFile(@"C:\  
Users\Desktop>LoginFailed.png");
```

```
MessageBox.Show("You are not granted  
with access");
```

```
}
```

```
}
```

```
}
```

```
}
```

Conclusion:

Hence, we have learned and implemented to design and data validation of login form.

✓
17/27