Project Title:- File Encryption Decryption using symmetric key cryptography

Group Members:

1) Shaikh Sujan Karim (2020BCS505)
2) Shinde Akshata Karabhari (2019BCS134)
3) Nilewar Pratiksha Datta (2020BCS508)

Project Code:

```
import java.awt.FlowLayout;
import java.awt.Font;
import java.awt.Label;
import java.io.File;
import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.IOException;
import javax.imageio.ImageIO;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JFileChooser;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JTextField;

public class ImageOperation {
   public static void operate(int key) {
```

```
        // to select the file
        JFileChooser fileChooser = new JFileChooser();
        fileChooser.showOpenDialog(null);

        File file = fileChooser.getSelectedFile();
        // file FileInputStream to convert file into bytes of data
        try {

            FileInputStream fis = new FileInputStream(file);
            byte[] data = new byte[fis.available()];
            fis.read(data);
            int i = 0;
            for (byte b : data) {
                System.out.println(b);
                // perform XOR operation to encrypt the file
                data[i] = (byte) (b ^ key);
                i++;
            }

            FileOutputStream fos = new FileOutputStream(file);
            fos.write(data);
            fos.close();
            fis.close();
            JOptionPane.showMessageDialog(null, "Done! Go and Check your
selected image path!");

        } catch (Exception e) {
```

```
            e.printStackTrace();
        }
    }

    public static void main(String[] args) throws IOException {

        JFrame f = new JFrame();
        f.setTitle("File Encryption/Decryption Tool");
        f.setSize(400, 500);
        f.setLocationRelativeTo(null);
        f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        f.setContentPane(new JLabel(new ImageIcon(ImageIO.read(new
File("back.png")))));

        Font fonts = new Font("Bembo", Font.BOLD, 18);
        JLabel label = new JLabel();
        label.setText("Enter Key To Encrypt/ Decrypt File");
        label.setFont(fonts);
        label.setBounds(100, 200, 100, 50);

        JTextField textField = new JTextField(20);
        textField.setBounds(100, 150, 20, 30);
        textField.setFont(fonts);

        Font font = new Font("Roboto", Font.BOLD, 18);
        JButton button = new JButton();
        button.setText("Select File");
```

```
        button.setFont(font);


        Label l1, l2;
        l1 = new Label("To Decrypt file give the same key as given on Encryption
time!");
        l1.setBounds(50, 250, 100, 30);
        l2 = new Label("And select the same file!");
        l2.setBounds(50, 150, 100, 30);


        button.addActionListener(e -> {
           System.out.println("button clicked");
           String text = textField.getText();
           int temp = Integer.parseInt(text);
           operate(temp);
        });


        f.setLayout(new FlowLayout());
        f.add(label);
        f.add(textField);
        f.add(button);
        f.add(l1);
        f.add(l2);
        f.setVisible(true);
}
    }
```