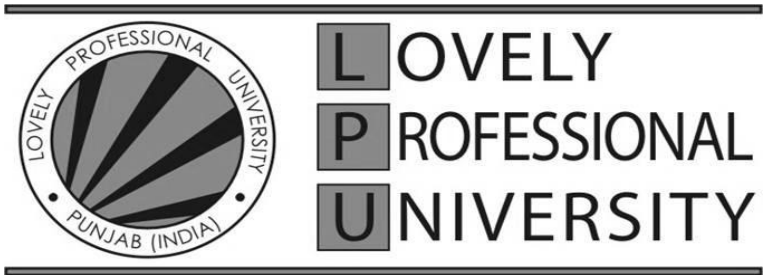
JAVA PROJECT REPORT

(Project Term January-May 2023)

GUESSING THE NUMBER



FACULTY – DR. RANJITH KUMAR SIR

GROUP MEMBER

ABHISHEK PATHAK	**REG NO :12111136**	**ROLL NO: RK21STB69**
PRADYUMN TRIPATHI	**REG NO :12110326**	**ROLL NO:RK21STA25**
SAHAJ SACHDEVA	**REG NO :12110280**	**ROLL NO:RK21STB70**

"School of Computer Science and Engineering"
Lovely Professional University
Phagwara, Punjab

DECLARATION

I hereby declare that the project work entitled “Guessing The Number” is an authentic record of our own work carried out as requirements of Java Project for the award of B.Tech degree in Computer Science and Engineering from Lovely Professional University, Phagwara, under the guidance of Dr. Ranjith Kumar, August to November 2022. All the information furnished in this capstone project report is based on our own intensive work and is genuine.

ABHISHEK PATHAK – 12111136

PRADYUMN TRIPATHI- 12110326

SAHAJ SACHDEVA- 12110280

TABLE OF CONTENTS

INTRODUCTION

This is a Java program called "Computer Guessing Game" that implements a guessing game where the computer tries to guess a number chosen by the user. The program uses a GUI (graphical user interface) created with the Swing framework.

The game is implemented using Java's Swing library to create a graphical user interface (GUI). The GUI includes a text area to display messages to the player, three buttons for the player to select whether their number is higher, lower, or correct, and a text field for the player to enter their name.

The code starts by creating a JFrame to prompt the player to enter their name. Once the player enters their name, the game starts by generating the first guess and displaying it to the player. The player can then select one of the three buttons to indicate whether their number is higher, lower, or correct.

PROPOSED TECHNIQUE

The program starts by prompting the user to enter their name in a separate window. Once the user enters their name and submits it, the Shown . The computer generates a random guess between 1 and 100, and displays it on the screen along with a message asking the user whether their number is higher or lower than the guess. The user can then click on one of three buttons: "Higher", "Lower", or "Correct", depending on whether the guess is too high, too low, or correct.

If the user clicks on the "Higher" button, the program adjusts the range of possible numbers to exclude all numbers less than or equal to the current guess, and generates a new guess within that range. Similarly, if the user clicks on the "Lower" button, the program adjusts the range to exclude all numbers greater than or equal to the current guess. This process continues until the user clicks on the "Correct" button, at which point the program displays a message saying that the computer has guessed the user's number correctly.

The program implements the ActionListener interface to handle button clicks, and also uses try-catch blocks to handle any exceptions that may occur.

Overall, this program provides a simple and fun game for users to play, while also demonstrating the use of Swing components and basic programming in java.

MODULES

3.1 PLAYER REGISTRATION

```
public ComputerGuessingGame() {
    // Get player's name
    JFrame nameFrame = new JFrame(title:"Enter Your Name");
    JPanel namePanel = new JPanel();
    JLabel nameLabel = new JLabel(text:"Enter Your Name:");
    textField = new JTextField(columns:20);
    JButton submitButton = new JButton(text:"Submit");
    submitButton.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent e) {
            playerName = textField.getText();
            nameFrame.dispose();
            startGame();
        }
    });
    namePanel.add(nameLabel);
    namePanel.add(textField);
    namePanel.add(submitButton);
    nameFrame.add(namePanel);
    nameFrame.pack();
    nameFrame.setLocationRelativeTo(c:null);
    nameFrame.setVisible(b:true);
}
```

This code defines a constructor for a class called ComputerGuessingGame. When an object of this class is created, the constructor is called and it creates a GUI (graphical user interface) window that prompts the player to enter their name.

The GUI window is created using the Swing library, which is part of Java's standard class library. The window is created as a JFrame object with the title "Enter Your Name". The user interface components such as the label, text field, and button are created as JLabel, JTextField, and JButton objects

respectively. These components are added to a JPanel object, which is then added to the JFrame object.

The submit button has an action listener attached to it that listens for when the button is clicked. When the button is clicked, the actionPerformed() method of the action listener is executed. This method gets the text from the text field and stores it in a variable called playerName. Then, it closes the nameFrame window and calls the startGame() method to start the game.

START GAME MODULE

```
private void startGame() {
    setTitle(title:"Computer Guessing Game");
        setLayout(new GridBagLayout());


        guess = (int) (Math.random() * 100) + 1;


        add(new JLabel("Is your number higher or lower than " + guess + "?"),
            createGridBagConstraints(x:0, y:0, width:3, height:1, GridBagConstraints.CENTER));

        textArea = new JTextArea(rows:10, columns:30);
        textArea.setEditable(b:false);
        JScrollPane scrollPane = new JScrollPane(textArea);
        add(scrollPane, createGridBagConstraints(x:0, y:1, width:3, height:1, GridBagConstraints.CENTER));

        higherButton = new JButton(text:"Higher");
        higherButton.addActionListener(this);
        add(higherButton, createGridBagConstraints(x:0, y:2, width:1, height:1, GridBagConstraints.CENTER));

        lowerButton = new JButton(text:"Lower");
        lowerButton.addActionListener(this);
        add(lowerButton, createGridBagConstraints(x:1, y:2, width:1, height:1, GridBagConstraints.CENTER));

        correctButton = new JButton(text:"Correct");
        correctButton.addActionListener(this);
        add(correctButton, createGridBagConstraints(x:2, y:2, width:1, height:1, GridBagConstraints.CENTER));


        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        pack();
        setLocationRelativeTo(c:null);
        setVisible(b:true);
}
```

The code is for a simple GUI-based game called "Computer Guessing Game". It creates a game window with a flexible grid layout using a GridBagLayout manager. The game generates a random number between 1 and 100, which serves as the computer's initial guess. The interface includes a JLabel displaying the initial guess, and three JButton components labeled "Higher", "Lower", and "Correct". The user interacts with the game by clicking these buttons to provide feedback on whether the computer's guess is higher or lower than their number, or if the guess is correct. The game also includes a JTextArea to display the game log. Overall, this code sets up the basic interface of the "Computer Guessing Game" and creates the necessary components for the game to function.

This is a Java code for a simple GUI-based game called "Computer Guessing Game". The code starts by defining a method called startGame().

The first line in the startGame() method sets the title of the game window to "Computer Guessing Game".

The second line uses the setLayout() method to set the layout manager of the game window to a GridBagLayout. This layout manager allows the game components to be arranged in a flexible grid of rows and columns.

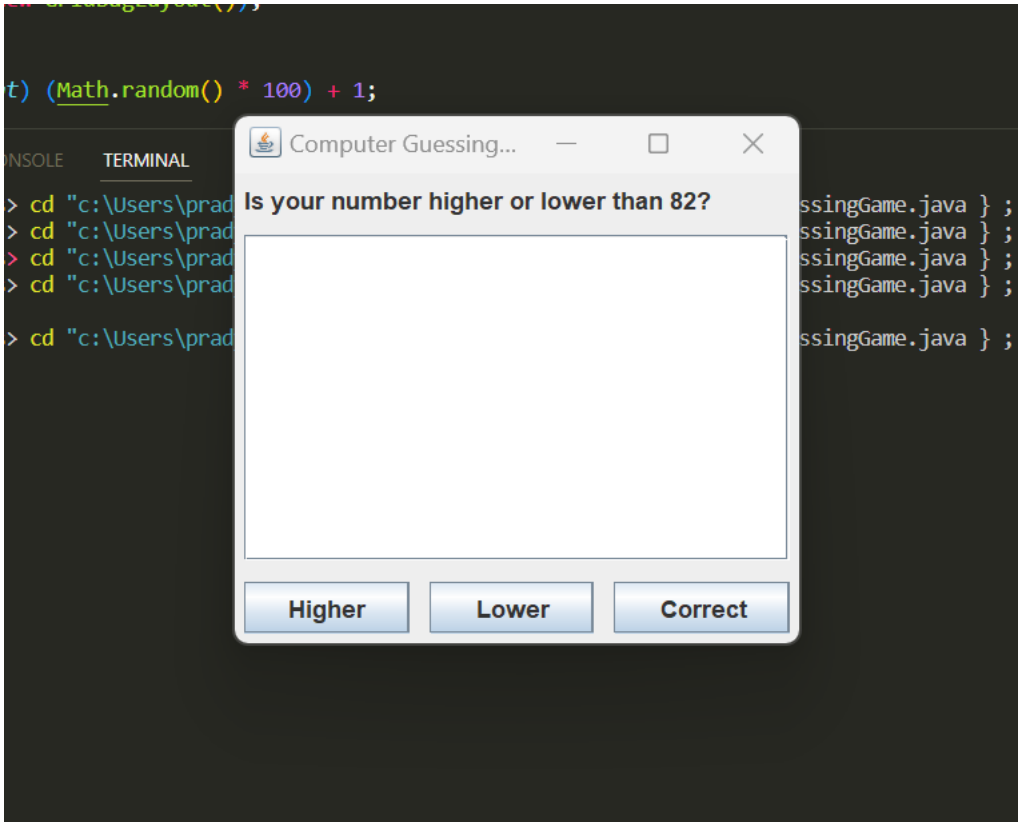
Next, the code generates a random number between 1 and 100 (inclusive) using the Math.random() method. This number will be used as the initial guess made by the computer.

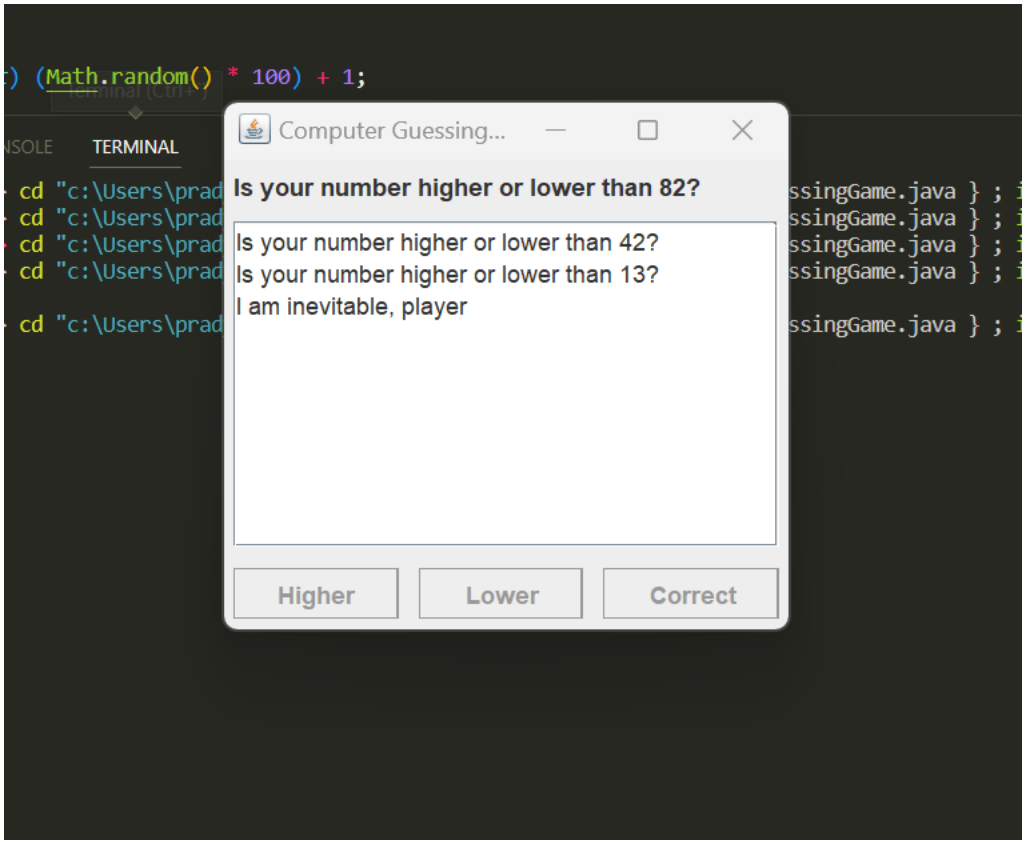
After that, the code adds a JLabel to the game window using the add() method. The label displays a message asking the user if their number is higher or lower than the initial guess made by the computer. The createGridBagConstraints() method is used to create the constraints for the label's position within the grid layout.

The code then creates a JTextArea and a JScrollPane to display the game log. The JTextArea is added to the game window using the add() method, and the JScrollPane is added using the createGridBagConstraints() method.

Next, the code creates three JButton components for the user to interact with. The buttons are labeled "Higher", "Lower", and "Correct". Each button is added to the game window using the add() method, and the createGridBagConstraints() method is used to create the constraints for their position within the grid layout.

Finally, the code sets some basic properties of the game window using setDefaultCloseOperation(), pack(), setLocationRelativeTo(), and setVisible() methods.


t) (Math.random() * 100) + 1;
NSOLE
TERMINAL
> cd "c:\Users\prad
> cd "c:\Users\prad
> cd "c:\Users\prad
> cd "c:\Users\prad
> cd "c:\Users\prad
ssingGame.java } ;
ssingGame.java } ;
ssingGame.java } ;
ssingGame.java } ;
ssingGame.java } ;
Computer Guessing...
Is your number higher or lower than 82?
Higher
Lower
Correct


(Math.random() * 100) + 1;
TERMINAL
cd "c:\Users\prad
Computer Guessing...
Is your number higher or lower than 82?
Is your number higher or lower than 42?
Is your number higher or lower than 13?
I am inevitable, player
Higher
Lower
Correct

Code Snippet

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class ComputerGuessingGame extends JFrame implements
ActionListener {
    private JTextField textField;
    private JTextArea textArea;
    private JButton higherButton;
    private JButton lowerButton;
    private JButton correctButton;
    private int min = 1;
    private int max = 100;
    private int guess;
    private String playerName;

    public ComputerGuessingGame() {
        // Get player's name
        JFrame nameFrame = new JFrame("Enter Your Name");
        JPanel namePanel = new JPanel();
        JLabel nameLabel = new JLabel("Enter Your Name:");
        textField = new JTextField(20);
        JButton submitButton = new JButton("Submit");
        submitButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                playerName = textField.getText();
                nameFrame.dispose();
                startGame();
            }
        });
        namePanel.add(nameLabel);
        namePanel.add(textField);
        namePanel.add(submitButton);
        nameFrame.add(namePanel);
        nameFrame.pack();
        nameFrame.setLocationRelativeTo(null);
        nameFrame.setVisible(true);
    }

    private void startGame() {
        setTitle("Computer Guessing Game");
```

```
            setLayout(new GridBagLayout());

            // Generate first guess
            guess = (int) (Math.random() * 100) + 1;

            // Add UI components
            add(new JLabel("Is your number higher or lower than " + guess
+ "?"),
                createGridBagConstraints(0, 0, 3, 1,
GridBagConstraints.CENTER));

            textArea = new JTextArea(10, 30);
            textArea.setEditable(false);
            JScrollPane scrollPane = new JScrollPane(textArea);
            add(scrollPane, createGridBagConstraints(0, 1, 3, 1,
GridBagConstraints.CENTER));

            higherButton = new JButton("Higher");
            higherButton.addActionListener(this);
            add(higherButton, createGridBagConstraints(0, 2, 1, 1,
GridBagConstraints.CENTER));

            lowerButton = new JButton("Lower");
            lowerButton.addActionListener(this);
            add(lowerButton, createGridBagConstraints(1, 2, 1, 1,
GridBagConstraints.CENTER));

            correctButton = new JButton("Correct");
            correctButton.addActionListener(this);
            add(correctButton, createGridBagConstraints(2, 2, 1, 1,
GridBagConstraints.CENTER));


            setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
            pack();
            setLocationRelativeTo(null);
            setVisible(true);
    }

    // Utility method to create GridBagConstraints
    private GridBagConstraints createGridBagConstraints(int x, int y, int
width, int height, int anchor) {
        GridBagConstraints gbc = new GridBagConstraints();
```

```
        gbc.gridx = x;
        gbc.gridy = y;
        gbc.gridwidth = width;
        gbc.gridheight = height;
        gbc.anchor = anchor;
        gbc.insets = new Insets(5, 5, 5, 5);
        gbc.fill = GridBagConstraints.BOTH;
        gbc.weightx = 1.0;
        gbc.weighty = 1.0;
        return gbc;
    }

    public void actionPerformed(ActionEvent e) {
        try{
            if (e.getSource() == higherButton) {
                // User's number is higher, adjust range and guess again
                min = guess + 1;
                guess = (int) (Math.random() * (max - min + 1)) + min;
                textArea.append("Is your number higher or lower than " +
guess + "?\n");
            } else if (e.getSource() == lowerButton) {
                // User's number is lower, adjust range and guess again
                max = guess - 1;
                guess = (int) (Math.random() * (max - min + 1)) + min;
                textArea.append("Is your number higher or lower than " +
guess + "?\n");
            } else if (e.getSource() == correctButton) {
                // User's number was guessed correctly
                textArea.append("I am inevitable, " + playerName + "\n");
                higherButton.setEnabled(false);
                lowerButton.setEnabled(false);
                correctButton.setEnabled(false);
            }
        }
        catch(Exception exception){
            System.out.println("Error: "+exception);
        }
    }


    public static void main(String[] args) {
        new ComputerGuessingGame();
    }
```

CONCLUSION

- In conclusion, the Computer Guessing Game is a simple yet entertaining Java program that demonstrates the use of graphical user interfaces (GUI) and event-driven programming. The program prompts the user to enter their name before the game begins, and then asks the user to think of a number between 1 and 100. The computer then tries to guess the user's number by asking if it is higher or lower than its current guess, and adjusting its range accordingly.

- The program uses the Java Swing library to create the GUI components, such as text fields, buttons, and labels, and the ActionListener interface to handle user events, such as button clicks. The program also uses the Math library to generate random numbers for the computer's guesses.

- Overall, the Computer Guessing Game is a fun and educational project for beginners to learn about GUI programming and event-driven programming in Java. With some modifications and enhancements, it can be turned into a more sophisticated game or tool for educational purposes.