JAVA PROJECT REPORT

(Project Term January-May 2023)

BRICK BREAKER A CHALLENGING ARCADE ACTION GAME IN JAVA

Submitted by:

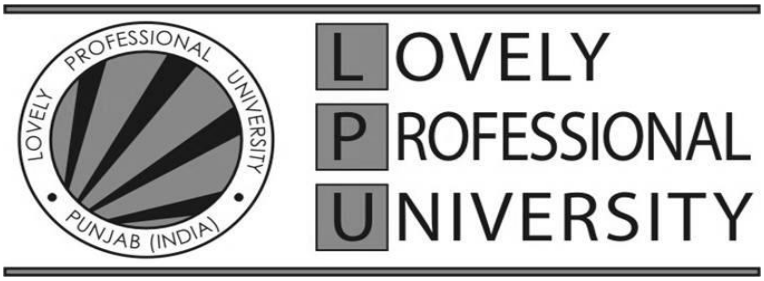
NAME: PRINCE KUMAR
REGI NO: **12112403**
ROLL NO: RK21WYA02

Course Code CSE-310

Under the Guidance of:

DR.A.RANJITH KUMAR

School of Computer Science and Engineering

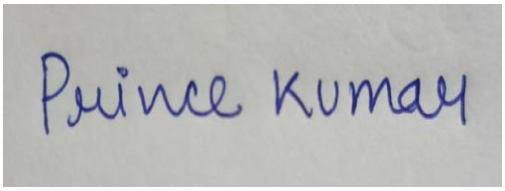


DECLARATION

We hereby declare that the project work entitled ("BRICK-BREAKER GAME") is an authentic record of our own work carried out as requirements of Capstone Project for the award of B.Tech degree in CSE from Lovely Professional University, Phagwara, under the guidance of (DR.A.RANJITH KUMAR), during August to November 2022. All the information furnished in this capstone project report is based on our own intensive work and is genuine.

NAME: PRINCE KUMAR

REGI NO: 12112403



Date:11-04-2023

TABLE OF CONTENTS

1.INTRODUCTION

Brick Breaker is a classic game involving breaking a brick wall with a ball and a paddle. The game was popularized in the 1970s and has since been adapted for various platforms, including mobile devices and personal computers.I create this Brick Breaker game using various techniques such as Java Swing, JavaFX. Java Swing is a powerful toolkit for building graphical user interfaces (GUIs) in Java. It provides a wide range of components, including buttons, labels, text fields, and more, that I used to create the login page and register page. In this project I used JavaFX to draw the game board, bricks, paddle, ball, and other visual elements of the game.In this game, the player controls a paddle at the bottom of the screen using the left-arrow key and right-arrow key and when the ball collides with the paddle ball bounces towards a wall of bricks at the top of the screen if the ball collides with brick, the brick will be removed from the screen, and the player's score will be increased. The objective of the game is to break all the bricks without letting the ball fall off the bottom of the screen.

2.Proposed Technique

2.1.Module 1

MODULE 1 contains a login page and a register page to allow users to create an account and sign in to access the game.For creating login page and register page I mainly used Java Swing.Java Swing is a powerful toolkit for building graphical user interfaces (GUI's) in Java. It provides a wide range of components, including buttons, labels, text fields, and more, that I used to create your login page and register page.

2.1.1.LOGIN PAGE

The login page allows users who have already registered an account to authenticate their identity by entering their Username and password. Once the user's credentials are verified against the information stored in the database(I used file to store the information), they are granted access to their account and game if not matched it will display an invalid username and password.

IMG-01

CODE SNIPPET FOR LOGIN PAGE:

```
class logincod extends JFrame{
JTextField t1,t2;
JButton b1,b2;
JLabel l1,l2;
    logincod(){

        setLayout(null);
        setDefaultCloseOperation(EXIT_ON_CLOSE);
        l1=new JLabel("Login");
        l1.setFont(new Font("Times New Roman",Font.BOLD,30));
        l1.setForeground(Color.BLUE);
        l1.setBounds(300,10,300,40);
        add(l1);
        ImageIcon i1=new ImageIcon(ClassLoader.getSystemResource("login.jpg"));
        JLabel img1=new JLabel(i1);

        img1.setBounds(0, 0, 200, 300);
        add(img1);
        t1=new JTextField(60);
        t2=new JPasswordField(60);
        b1=new JButton("SignIn");
        b2=new JButton("Register");
        JLabel username=new JLabel("User_name");
        username.setBounds(210,60,100,25);
        username.setFont(new Font("Verdana",Font.BOLD,14));
        username.setForeground(Color.BLACK);

        add(username);
        JLabel Password=new JLabel("Password");
        Password.setBounds(210,100,100,25);
        Password.setFont(new Font("Verdana",Font.BOLD,14));
        Password.setForeground(Color.BLACK);    add(Password);
```

2.1.2.REGISTER PAGE

The register page in this game serves the primary function of allowing new users to create an account and register their information. Specifically, the register page prompts the user to enter their username, and password. Once the user's information is submitted, it is stored in a database and associated with their account after that user can login to the game by entering their username and password and then user can play the game.

IMG-02

CODE FOR REGISTER PAGE:

```
class Register extends JFrame{
    JTextField t1,t2;
    JButton b1;
    JLabel l1;
    Register(){
```

```
        getContentPane().setBackground(Color.WHITE);
        setLayout(null);
        t1=new JTextField(60);
        t2=new JTextField(60);
        ImageIcon i1=new ImageIcon(ClassLoader.getSystemResource("l4.jpg"));
        JLabel img1=new JLabel(i1);

        img1.setBounds(0, 0, 200, 300);
        add(img1);
        b1=new JButton("Submit");
        l1=new JLabel("Register");
        l1.setFont(new Font("Times New Roman",Font.BOLD,30));
        l1.setForeground(Color.BLUE);
        l1.setBounds(300,10,300,40);
        add(l1);

        b1.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent ae){
                try{
                    FileWriter f=new FileWriter("login.txt",true);
                    f.write(t1.getText()+"\t"+t2.getText()+"\n");
                    f.close();
                    JFrame con =new JFrame();
                    JOptionPane.showMessageDialog(con,"Registration scucessfull");
                    dispose();
                }catch(Exception e){}

            }
        } );

        add(b1);
        }
}
```

2.2.Module 2

Module 2 mainly contain BRICK-BREAKER GAME (GUI).

The interface of a brick breaker game typically includes several key elements:

Game board: The game board is where the bricks are located, and it's where the player will bounce the ball around to destroy them.

Paddle: The paddle is controlled by the player and is used to bounce the ball back up towards the bricks.

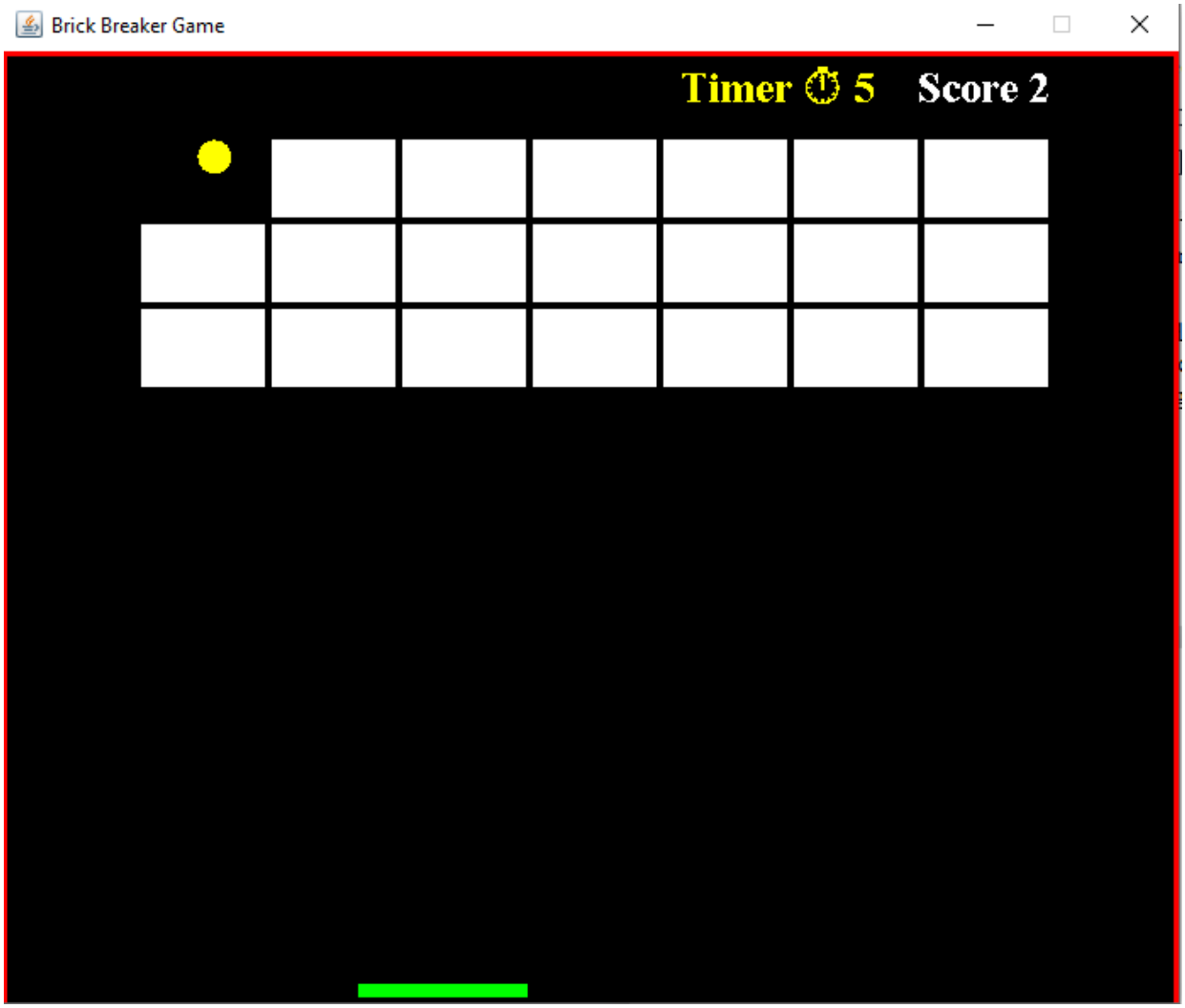
Ball: The ball is what the player uses to hit the bricks and destroy them. It bounces off the walls and paddle, and must be kept in play to continue the game.

Bricks: The bricks are arranged on the game board in various patterns, and the player must hit them with the ball to destroy them.

Score: The player's score is typically displayed on the screen, and it updates by +2 as the player destroys bricks.

Timer: The time spent Playing the game it increases every second.

Overall, the interface of this game is designed to be intuitive and easy to use, with simple controls and clear feedback to the player about their progress and performance.



IMG-03

CODE SNIPPET FOR MODULE 2

```
public class Gameplay extends JPanel implements KeyListener,ActionListener {
  private boolean play=false;
  public int score=0;
  private int totalbrick=21;
  private Timer timer;
  private Timer s;
  private int delay=9;
  public int second=0;
  private int playerx=310;
  private int ballposX=120;
  private int ballposY=350;
```

```
private int ballxdir=-1;
private int ballydir=-2;
private mapgenerator map;
Gameplay(){
  map=new mapgenerator(3,7);
  addKeyListener(this);
  setFocusable(true);
  setFocusTraversalKeysEnabled(false);
  timer=new Timer(delay,this);
  showtimer();
  s.start();
  timer.start();}

public void showtimer(){
      s=new Timer(1000,new ActionListener() {
      public void actionPerformed(ActionEvent e){
      second++;
      }
  });
}
```

3.Conclusion

In conclusion, building a Brick Breaker game in Java requires a solid understanding of Java programming concepts, such as object-oriented programming, event-driven programming,JFrames, and graphics programming. The game involves designing and implementing game mechanics, such as collision detection, ball movement, and brick destruction.

4.Future Enhancements

There are several potential enhancements that could be made to this Brick Breaker game to improve its gameplay, graphics, and overall user experience.

Adding more Levels: Adding more levels can make this game more interesting and competitive for the users.

Improving graphics : Updating the game's graphics could make it more visually appealing and modern-looking. This could involve adding more detailed backgrounds, smoother animations, or special effects for power-ups or explosions.

Powers-up: Adding power-ups to the game would give players new abilities or bonuses that could help them break bricks more easily or increase their score. Examples of power-ups could include extra lives, larger paddle size, multiple balls, or the ability to shoot lasers.