A Mini Project Report

On

“The Hangman Game”

Submitted in partial fulfilment of the requirements for the award of degree of

Bachelor of Engineering

in

Computer Science and Engineering

Submitted by

Aayush Gour	1608-16-733-062
Srividya Kotagiri	1608-16-733-083
Ahmed Rizwan	1608-16-733-084

Under the Guidance of

Mr. P. Ravindra

Asst. Professor

Dept. of Computer Science and Engineering

Department of Computer Science and Engineering

Matrusri Engineering College

(Affiliated to Osmania University, Approved by AICTE)

Saidabad, Hyderabad - 500059

(2017-2018)

Department of Computer Science and Engineering

Matrusri Engineering College

(Affiliated to Osmania University, Approved by AICTE)
Saidabad, Hyderabad - 500059

CERTIFICATE

This is to Certify that A Mini Project report entitled **“The Hangman Game**” is being submitted by Aayush Gour (1608-16-733-062), Srividya Kotagiri (1608-16-733-083), Ahmed Rizwan (1608-16-733-084) in partial fulfilment of the requirement of the award for the degree of Bachelor of Engineering in “Computer Science and Engineering” O.U., Hyderabad during the year 2017-2018 is a record of bonafide work carried out by him/her under my guidance. The results presented in this mini project report have been verified and are found to be satisfactory.

Project Coordinator	Project Guide	HOD
Mrs. K. Shalini **Assistant Professor** **Dept. of CSE**	**Mr. P. Ravindra** **Assistant Professor** **Dept. of CSE**	**Dr. P. Vijayapal Reddy** **Professor & Head** **Dept. of CSE**

External Examiner(s)

ACKNOWLEDGEMENT

It is our privilege and pleasure to express our profound sense of respect, gratitude and indebtedness to our guide Mr. **P. Ravindra**, Assistant professor, Department of Computer Science and Engineering, Matrusri Engineering College, for his/her indefatigable inspiration, guidance, cogent discussion, constructive criticisms and encouragement throughout this dissertation work.

We express our sincere thanks to mini project coordinator **Mrs. K. Shalini**, Assistant professor, Department of Computer Science and Engineering, Matrusri Engineering College, for her valuable suggestions and constant help in completing the work.

We express our sincere gratitude to **Dr. P. Vijayapal Reddy**, Professor & Head, Department of Computer Science and Engineering, Matrusri Engineering College, for his precious suggestions, motivation and co-operation.

We extend our sincere thanks to **Dr. D. Hanumantha Rao**, Principal, Matrusri Engineering College, Saidabad, Hyderabad, for his encouragement and constant help.

We extend our sincere thanks to all the teaching and non-teaching staff of CSE Department for their support and encouragement.

Last but not least, we wish to acknowledge our friends and family members for giving moral strength and helping us to complete this dissertation.

CONTENTS

ABSTRACT

The hangman game has amused people of all age groups, throughout the world. It is one of the most popular games in the world. Since it is very enjoyable, we decided to make our own hangman game.

The objective of this game is to guess the word correctly. The player will be given 5 chances to guess it right.

In our project we have made 5 levels with increasing difficulty so that the player will be amused and also know new words. We have designed the code in such a way that the player gets a new word every time he plays the game.

Every level has one question. If the player is able to guess the word within 5 chances, then he moves to the next level and he gets 5 chances to guess the next word. If the player fails to guess the word correctly, he gets a message about the correct word.

LIST OF FIGURES

Chapter 1
Introduction

Hangman is a popular word game in which one player (the "chooser") chooses a secret word and another player (the "guesser") attempts to guess the word one letter at a time. If a guessed letter appears in the word, all instances of it are revealed. If not, the guesser loses a chance.

The word to guess is represented by a row of dashes, representing each letter of the word. In most variants, proper nouns, such as names, places, and brands, are not allowed. Slang words, sometimes referred to as informal or shortened words, are also not allowed. If the guessing player suggests a letter which occurs in the word, the other player writes it in all its correct positions. If the suggested letter or number does not occur in the word, the other player draws one element of a hanged man stick figure as a tally mark.

The player guessing the word may, at any time, attempt to guess the whole word. If the word is correct, the game is over and the guesser wins. Otherwise, the other player may choose to penalize the guesser by adding an element to the diagram. On the other hand, if the other player makes enough incorrect guesses to allow his opponent to complete the diagram, the game is also over, this time with the guesser losing. However, the guesser can also win by guessing all the letters or numbers that appears in the word, thereby completing the word, before the diagram is completed.

1.1 – Existing System

There are a lot of systems and algorithms for the hangman game.

Given below are a few.

- **Backward Propagation Algorithm**

Backward propagation is a method used in artificial neural networks to calculate a gradient that is needed in the calculation of the weights to be used in the network. It is commonly used to train deep neural networks, a term referring to neural networks with more than one hidden layer.

Backward propagation is a special case of an older and more general technique called automatic differentiation. In the context of learning, backward propagation is commonly used by the gradient descent optimization algorithm to adjust the weight of neurons by calculating the gradient of the loss function. This technique is also sometimes called backward propagation of errors, because the error is calculated at the output and distributed back through the network layers.

- **Runtime Value Selection (RVS)**

If there is some value, such as importance of given letters vs. general letter frequency, it may be smart to adjust it at runtime based on how well the current value is doing. There is more than one way to adjust it.

Just choose whichever one is best. a.k.a. Compile-time Value Selection. Keep a list of different values and how well they each do. If the score starts to fall too low, switch to another value that has a record of good performance. Keep a list of different values and how well they each do. Before each round, select whichever value is doing best. Calculate the average for how well different values do, then choose a value to start with. If the score ever drops below average, switch to a different value. Have an array with just a default value, and if it falls below a certain ratio for too long, add another value to the array.

1.2 - Disadvantages

Though these are the best algorithms, they also come with some disadvantages.

- **Backward Propagation**

The gradient descent algorithm is generally very slow because it requires small learning rates for stable learning. The momentum variation is usually faster than simple gradient descent, since it allows higher learning rates while maintaining stability, but it is still too slow for many practical applications. These two methods would normally be used only when incremental training is desired. You would normally use Levenberg-Marquardt training for small and medium size networks, if you have enough memory available. If memory is a problem, then there are a variety of other fast algorithms available. For large networks you will probably want to use *trainscg* or *trainrp.*

- **Runtime Value Selection**

This algorithm can only be applied at runtime. Which means that if any input given does not match the datatype of the variable, then the compiler throws an exception and the program comes to a halt.

1.3 – Proposed System

The proposed system consists of the following algorithm.

- Retrieving data from the database and storing it in a String variable.
- Splitting the string into its respective word and hint.
- Converting the String to an array to ease computing.
- Creating another array with the same size of the word and placing a few random alphabets in the array.
- Accepting input from the user and comparing it with the alphabets of the extracted word.
- Computing the input and displaying the respective output.

1.4 – Advantages

This proposed system has the following advantages.

- Uses strings, so extraction of data is easy and faster.
- Converts strings to arrays, so computation becomes easier.
- A random word is chosen each time the game is played.
- Interactive Gameplay

1.5 – System Requirements

Hardware Requirements

Processor	-	PENTIUM® (II) or higher.
Processor speed	-	Minimum of 1 GHz.
RAM	-	Minimum of 500 GB.
Memory	-	Minimum of 256 GB.

Software Requirements

Operating System	-	Windows® XP, 7, 8, 8.1, 10.
IDE	-	NetBeans or JVM.

1.6 – Technologies Used

Java is a general-purpose computer-programming language that is concurrent, class-based, object-oriented, and specifically designed to have as few implementation dependencies as possible. It is intended to let application developers "write once, run anywhere" (WORA), meaning that compiled Java code can run on all platforms that support Java without the need for recompilation. Java applications are typically compiled to bytecode that can run on any Java virtual machine (JVM) regardless of computer architecture. As of 2016, Java is one of the most popular programming languages in use, particularly for client-server web applications, with a reported 9 million developers. Java was originally developed by James Gosling at Sun Microsystems(which has since been acquired by Oracle Corporation) and released in 1995 as a core component of Sun Microsystems' Java platform. The language derives much of its syntax from C and C++, but it has fewer low-level facilities than either of them.

Fig. 1.6.1 – Logo of Java

1.6.1 – Concepts Used

The following concepts were used for the execution of this project.

- **Packages**

  A package as the name suggests is a pack (group) of classes, interfaces and other packages. In java we use packages to organize our classes and interfaces. We have two types of packages in Java:

  - Built-in packages.
  - User defined package.

In this project, predefined packages were used for type conversion, calculation, comparison of strings and to display the output onto the console. User defined packages were used to display an image based on the number of lives the player has.

- **BufferedReader and FileReader**

  BufferedReader is a class present in java.io package. It is used to read text from a character-input stream, buffering characters so as to provide for the efficient reading of characters, arrays, and lines. The buffer size may be specified, or the default size may be used. The default is large enough for most purposes. In general, each read request made of a Reader causes a corresponding read request to be made of the underlying character or byte stream. It is therefore advisable to wrap a BufferedReader around any Reader whose read() operations may be costly, such as FileReaders and InputStreamReaders. Programs that use DataInputStreams for textual input can be localized by replacing each DataInputStream with an appropriate BufferedReader.

- **ProcessBuilder**

  ProcessBuilder is a class in java.lang package. It is used to create operating system processes. Each ProcessBuilder instance manages a collection of process attributes. The start() method creates a new Process instance with those attributes. The start() method can be istanvoked repeatedly from the same instance to create new subprocesses with identical or related attributes. ProcessBuilder can be used to help create operating system process. Before JDK 5.0, the only way to create a process and execute it was to use Runtime.exec() method. It extends class Object. This class is not synchronised.

  Since Java is platform independent, ProcessBuilder was used to clear the console screen by accessing the internal “cls” command of the Command Prompt.

- ## StringTokenizer

  StringTokenizer class is present in java.util package. This class breaks the string into tokens based on the separator specified. This class was used to split the line read from the database into its respective word and hint.

- ## Exception Handling

  An exception is an unwanted or unexpected event, which occurs during the execution of a program i.e at runtime, that disrupts the normal flow of the program’s instructions. All exception are sub classes of class **Throwable**, which is base class of hierarchy. One branch is headed by **Exception**. This class is used for exceptional conditions that user programs should catch. NullPointerException is an example of such an exception.

  In this project we have used exception handling in order to avoid checked exceptions like FileNotFoundException in case of BufferedReader and IOException & InterruptedException in case of ProcessBuilder.

- ## Event handling

  Event handling is used to handle events generated by various devices while executing a Java program. In this project, event handling was implemented using ActionListener method form java.awt.event.* package. This method provides event handling for buttons, checkboxes and menu items.

Chapter 2
Architecture and Implementation

Chapter 2.1 – Modules

Following are the modules used in this project.

- **Hangman()**

  This function is to initialize the frame, to set the layout, to display the initial dialog box, to set size, location and font for the game and to add an ActionListener.

- **init()**

  This method is used to accept input and initialize the buttons for the input and to display level information.

- **compare()**

  This method compares the user's input with the actual word in order to determine whether the input given was correct or not. It also displays whether the given input was correct or not.

- **actionPerformed()**

  This method is used for event handling. The method stores the input given by the user into a variable which is then compared with the word in the database.

- **paint()**

  This method is used to display the interface. It accesses the image files and displays the image according to the level and score of the player. It is also used to display information like the player's score, level and the words to be guessed.

Chapter 2.2 – Code

```
import java.io.*;
import java.util.*;
import java.awt.*;
import java.applet.*;
import java.awt.event.*;
import javax.swing.*;
//import game.hangman;

 public class Hangman extends Applet implements ActionListener
     {String[] hint={"A person who sells and arranges cut 
flowers.","A system for transmitting voices over a distance using 
wire or radio.","A sum of money due spread over  a period of 
time","Loose wet sand that yields easily to pressure and sucks in 
anything falling into it.","Designed to aid the memory, related to 
the power of memory.","Writing or drawings scribbled or sprayed 
illicitly on a wall in a public place.","Surprise attack by people 
lying in wait in a concealed position.","Belief maintained despite 
being contradicted by reality or rational argument.","Destroy 
utterly; obliterate.","Extremely large, massive, huge, 
colossal.","Too zealous in one's attitude or behavior, excessively 
enthusiastic.","Deliberately destroy, damage or obstruct.","A huge, 
powerful, and overwhelming force.","Able to adapt to many different 
functions or activities.","A person with whom one works in a 
profession or business.","The elected head of a republican 
state.","A ghost.","An event causing great and usually sudden damage 
or suffering or a disaster.","A frightening or unpleasant dream.","A 
stupid or careless mistake."};
     String [] 
answer={"F_OR__T","T_LE_HO_E","I_S__LM_NT","Q_ICKS___","M__MO_IC","G
R__FI_I","A_BUS_","D_LU__ON","A_NI_ILA__","G_RG__TU_N","OV_R_E_LO_S"
,"'_A_OTA_E","J_G__RNA_T","V_R__TI_E","_OL_EA_U_","_R_S_DE_T","_HA_T
_M","CA_AS_R_P_E","_I_H_MA_E","B_UN__R"};
     Font f1;
     String [] 
question={"FLORIST","TELEPHONE","INSTALMENT","QUICKSAND","MNEMONIC",
"GRAFFITI","AMBUSH","DELUSION","ANNIHILATE","GARGANTUAN","OVERZEALOU
S","'SABOTAGE","JUGGERNAUT","VERSATILE","COLLEAGUE","PRESIDENT","PHA
NTOM","CATASTROPHE","NIGHTMARE","BLUNDER"};
     char hintarr[],answerarr[],questionarr[];char in;
     Button 
b1,b2,b3,b4,b5,b6,b7,b8,b9,b10,b11,b12,b13,b14,b15,b16,b17,b18,b19,b
20,b21,b22,b23,b24,b25,b26;
     String s1,s2;
     Image img,logo;
     static Dialog d,end;
     int random,life,level=0,pass=0,co=0;

     public Hangman() throws Exception
            {Frame f= new Frame();
            d = new Dialog(f,"Welcome To Hangman", true);
            d.setLayout(new FlowLayout());
            Button b = new Button ("Start Game");
            b.addActionListener ( new ActionListener()
                   {
                          public void actionPerformed( ActionEvent e 
)
                                 {
                          Hangman.d.setVisible(false);}
                   });
            d.add( new Label ("Welcome to The Hangman Game"));
            d.add(b);
            d.setSize(250,100);
            d.setLocation (626,368);
            d.setVisible(true);
            f1 = new Font("Arial",Font.BOLD,18);


            }

    public void init()
            {
            s2="life1.jpg";
            life=5;
            level++;
            if(level>5)
                   {destroy();setVisible(false);

    JOptionPane.showMessageDialog(null,"CONGRATULATIONS!!! YOU
WON.");System.exit(0); }
            Random r=new Random();
            random=r.nextInt(19);

            answerarr=answer[random].toCharArray();
            questionarr=question[random].toCharArray();
            for(int i=0;i<question[random].length();i++)
                   {if(answerarr[i]=='_')
                          co++;
                   }


            setLayout(null);

            b1=new Button("A");
            b2=new Button("B");
            b3=new Button("C");
            b4=new Button("D");
            b5=new Button("E");
            b6=new Button("F");
            b7=new Button("G");
            b8=new Button("H");
            b9=new Button("I");
            b10=new Button("J");
            b11=new Button("K");
            b12=new Button("L");
            b13=new Button("M");
            b14=new Button("N");
            b15=new Button("O");
            b16=new Button("P");
            b17=new Button("Q");
            b18=new Button("R");
            b19=new Button("S");
            b20=new Button("T");
            b21=new Button("U");
            b22=new Button("V");
            b23=new Button("W");
            b24=new Button("X");
            b25=new Button("Y");
            b26=new Button("Z");
            b1.setFont(f1);
```

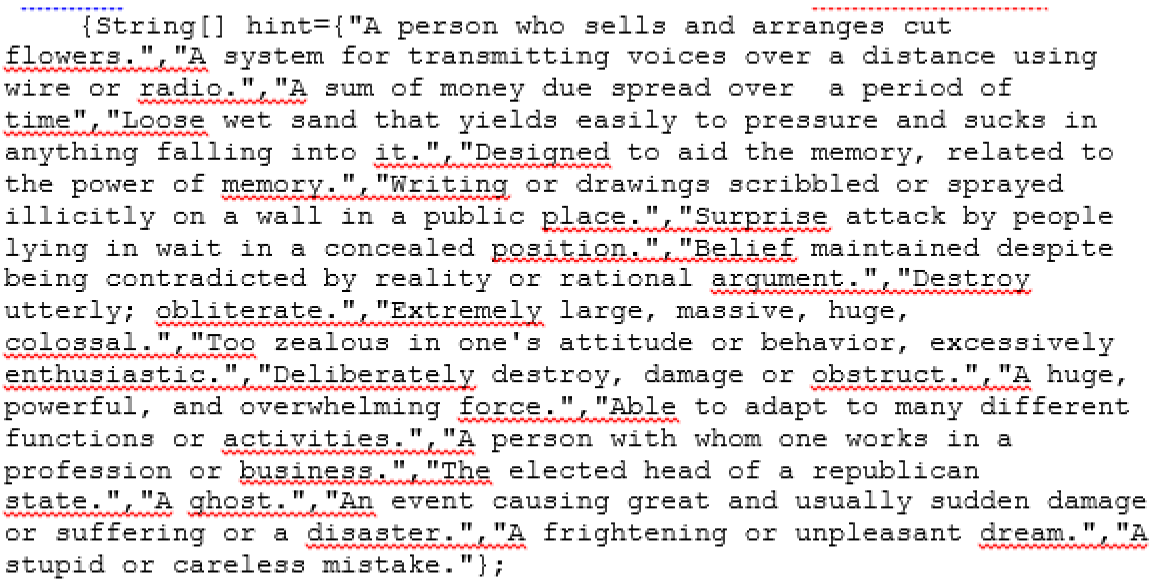
```
b2.setFont(f1);
b3.setFont(f1);
b4.setFont(f1);
b5.setFont(f1);
b6.setFont(f1);
b7.setFont(f1);
b8.setFont(f1);
b9.setFont(f1);
b10.setFont(f1);
b11.setFont(f1);
b12.setFont(f1);
b13.setFont(f1);
b14.setFont(f1);
b15.setFont(f1);
b16.setFont(f1);
b17.setFont(f1);
b18.setFont(f1);
b19.setFont(f1);
b20.setFont(f1);
b21.setFont(f1);
b22.setFont(f1);
b23.setFont(f1);
b24.setFont(f1);
b25.setFont(f1);
b26.setFont(f1);

b1.addActionListener(this);
b2.addActionListener(this);
b3.addActionListener(this);
b4.addActionListener(this);
b5.addActionListener(this);
b6.addActionListener(this);
b7.addActionListener(this);
b8.addActionListener(this);
b9.addActionListener(this);
b10.addActionListener(this);
b11.addActionListener(this);
b12.addActionListener(this);
b13.addActionListener(this);
b14.addActionListener(this);
b15.addActionListener(this);
b16.addActionListener(this);
b17.addActionListener(this);
b18.addActionListener(this);
b19.addActionListener(this);
b20.addActionListener(this);
b21.addActionListener(this);
b22.addActionListener(this);
b23.addActionListener(this);
b24.addActionListener(this);
b25.addActionListener(this);
b26.addActionListener(this);
add(b1);
add(b2);
add(b3);
add(b4);
add(b5);
add(b6);
add(b7);
add(b8);
add(b9);
```

```
        add(b10);
        add(b11);
        add(b12);
        add(b13);
        add(b14);
        add(b15);
        add(b16);
        add(b17);
        add(b18);
        add(b19);
        add(b20);
        add(b21);
        add(b22);
        add(b23);
        add(b24);
        add(b25);
        add(b26);

        b1.setBounds(5,500,40,40);
        b2.setBounds(50,500,40,40);
        b3.setBounds(95,500,40,40);
        b4.setBounds(140,500,40,40);
        b5.setBounds(185,500,40,40);
        b6.setBounds(230,500,40,40);
        b7.setBounds(275,500,40,40);
        b8.setBounds(320,500,40,40);
        b9.setBounds(365,500,40,40);
        b10.setBounds(410,500,40,40);
        b11.setBounds(455,500,40,40);
        b12.setBounds(500,500,40,40);
        b13.setBounds(545,500,40,40);
        b14.setBounds(5,545,40,40);
        b15.setBounds(50,545,40,40);
        b16.setBounds(95,545,40,40);
        b17.setBounds(140,545,40,40);
        b18.setBounds(185,545,40,40);
        b19.setBounds(230,545,40,40);
        b20.setBounds(275,545,40,40);
        b21.setBounds(320,545,40,40);
        b22.setBounds(365,545,40,40);
        b23.setBounds(410,545,40,40);
        b24.setBounds(455,545,40,40);
        b25.setBounds(500,545,40,40);
        b26.setBounds(545,545,40,40);
        repaint();
        JOptionPane.showMessageDialog(null,"Level "+level);
        }
public void compare()
        {
        for(int i=0;i<question[random].length();i++)
                if(in==questionarr[i])
                        {
                        if(in==answerarr[i])
                                {repaint();
                                }
                        else if(in!=answerarr[i])
                                {repaint();answerarr[i]=in;--co;
                                return;
                                }
                        }
```

```
                   --life;



            if(life==0)
                   {
                   ImageIcon icon=new ImageIcon("lost.gif");

                   JOptionPane.showMessageDialog(null,"","The 
Hangman Game",JOptionPane.INFORMATION_MESSAGE,icon);
                   setVisible(false);}

            if(co==0)
                   init();





            return;
            }
      public void actionPerformed(ActionEvent e)
            {
            String s1=e.getActionCommand();
            in=s1.charAt(0);


            compare();

            int count=0;
            for(int i=0;i<question[random].length();i++)
                   {if(answerarr[i]!='_')
                          count++;
                   if(count==question[random].length())
                          {init();}
                   }

            repaint();


            }
      public void paint(Graphics g)
            {
            g.setFont(f1);
            logo=getImage(getDocumentBase(),"logo.jpg");
            s2="life"+(6-life)+".jpg";
            img=getImage(getDocumentBase(),s2);
            g.drawImage(logo,523,5,this);
            g.drawImage(img,780,130,this);

            g.drawString("Level = "+Integer.toString(level),5,130);
            g.drawString("Life = "+Integer.toString(life),100, 130);

            g.drawString("Guess the word: 
"+String.valueOf(answerarr),5,170);
            g.drawString("Hint: "+hint[random],5,190);
            }
      }
/*<applet code="Hangman.class" width =1920 height=955></applet>*/
```

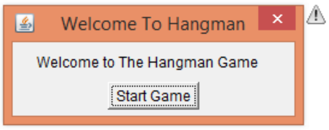
Chapter 3

Output Screenshots



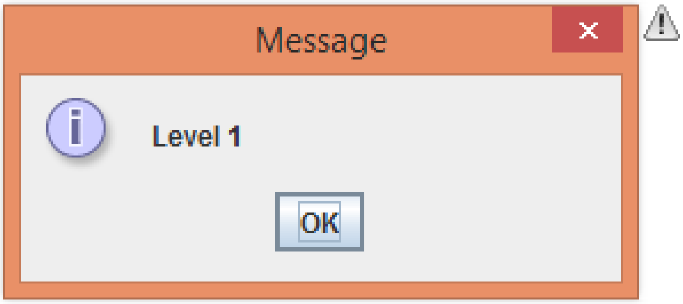
```
    {String[] hint={"A person who sells and arranges cut
flowers.","A system for transmitting voices over a distance using
wire or radio.","A sum of money due spread over  a period of
time","Loose wet sand that yields easily to pressure and sucks in
anything falling into it.","Designed to aid the memory, related to
the power of memory.","Writing or drawings scribbled or sprayed
illicitly on a wall in a public place.","Surprise attack by people
lying in wait in a concealed position.","Belief maintained despite
being contradicted by reality or rational argument.","Destroy
utterly; obliterate.","Extremely large, massive, huge,
colossal.","Too zealous in one's attitude or behavior, excessively
enthusiastic.","Deliberately destroy, damage or obstruct.","A huge,
powerful, and overwhelming force.","Able to adapt to many different
functions or activities.","A person with whom one works in a
profession or business.","The elected head of a republican
state.","A ghost.","An event causing great and usually sudden damage
or suffering or a disaster.","A frightening or unpleasant dream.","A
stupid or careless mistake."};
```

Fig. 3.1 – A part of the Database.



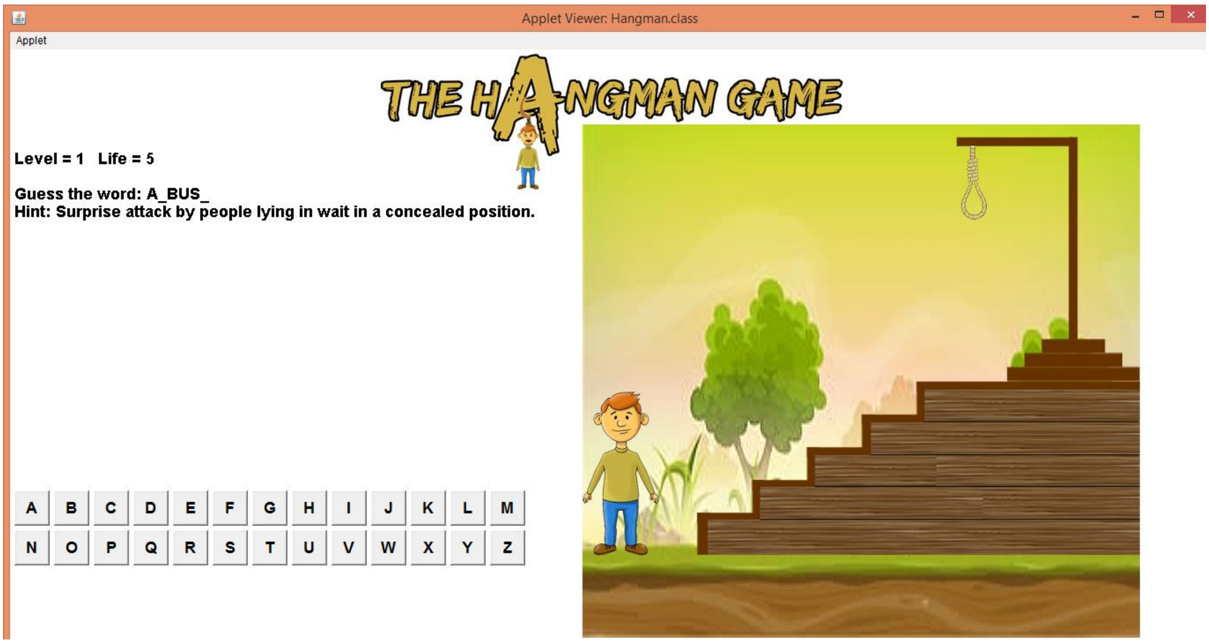
Welcome to The Hangman Game

**Fig. 3.2 – The first screen,
Welcome to hangman**



Level 1

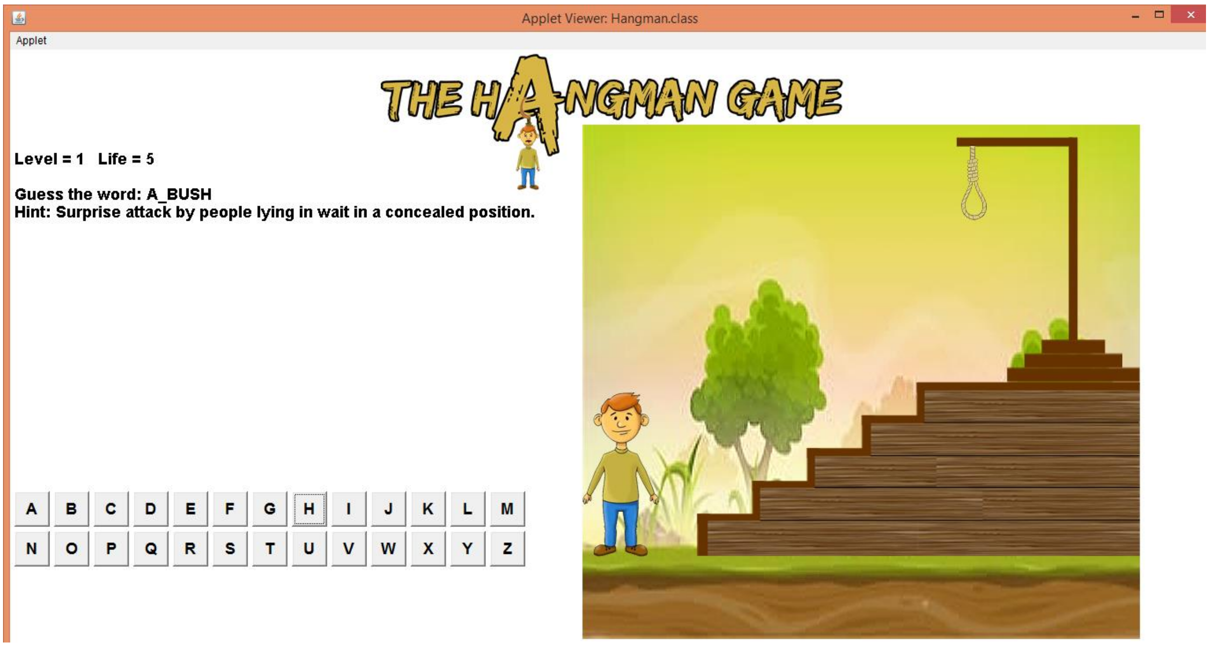
Fig. 3.3 – Level 1 dialog box.


Applet Viewer: Hangman.class
Applet
THE HANGMAN GAME
A B C D E F G H I J K L M
N O P Q R S T U V W X Y Z

Level = 1 Life = 5

Guess the word: A_BUS_
Hint: Surprise attack by people lying in wait in a concealed position.

Fig. 3. 4 – Level 1.

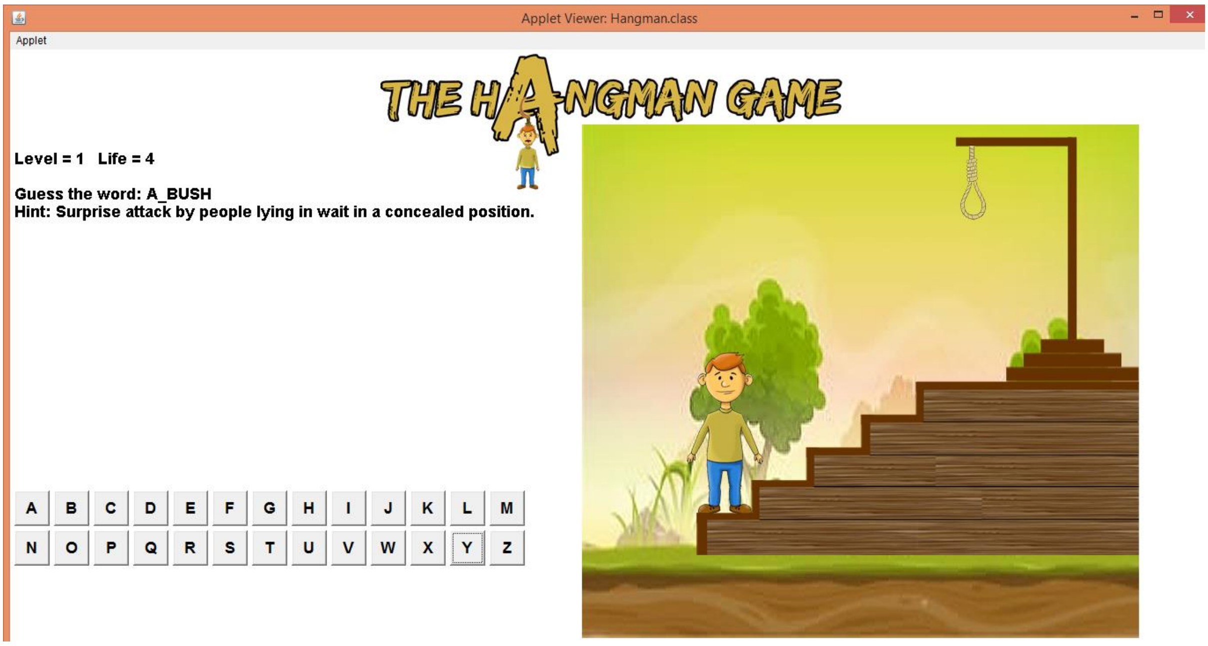

Applet Viewer: Hangman.class
Applet
THE HANGMAN GAME

Level = 1 Life = 5

Guess the word: A_BUSH
Hint: Surprise attack by people lying in wait in a concealed position.

A B C D E F G H I J K L M
N O P Q R S T U V W X Y Z

Fig. 3.5 – After a correct input is given.

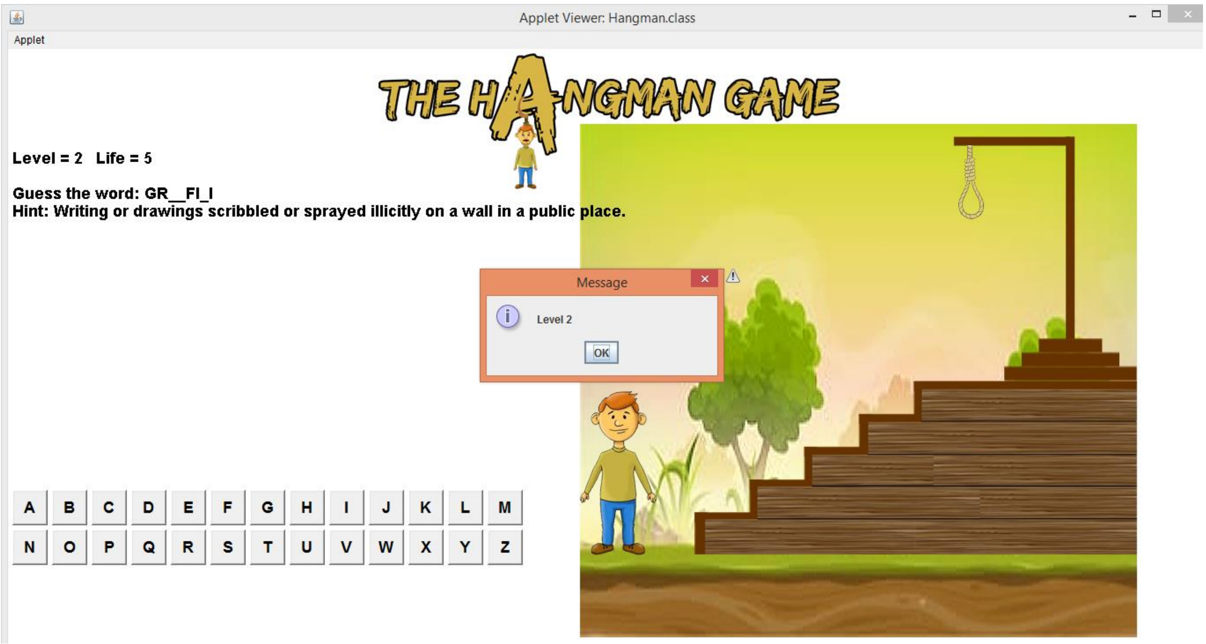

Applet Viewer: Hangman.class
Applet
THE HANGMAN GAME

Level = 1 Life = 4

Guess the word: A_BUSH
Hint: Surprise attack by people lying in wait in a concealed position.

A B C D E F G H I J K L M
N O P Q R S T U V W X Y Z

Fig. 3.6 – After one wrong input.



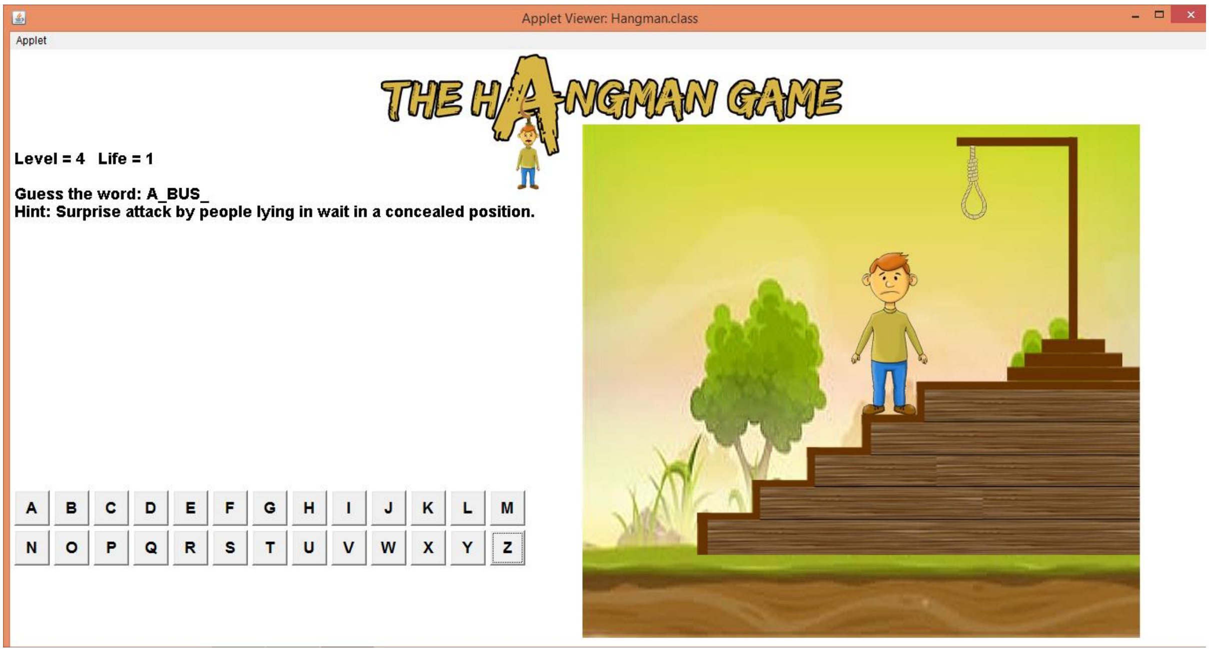
Applet Viewer: Hangman.class
Applet
THE HANGMAN GAME
Message
OK
A B C D E F G H I J K L M
N O P Q R S T U V W X Y Z

Level = 2 Life = 5

Guess the word: GR__FI_I
Hint: Writing or drawings scribbled or sprayed illicitly on a wall in a public place.

Level 2

Fig. 3.7 – Level 2.

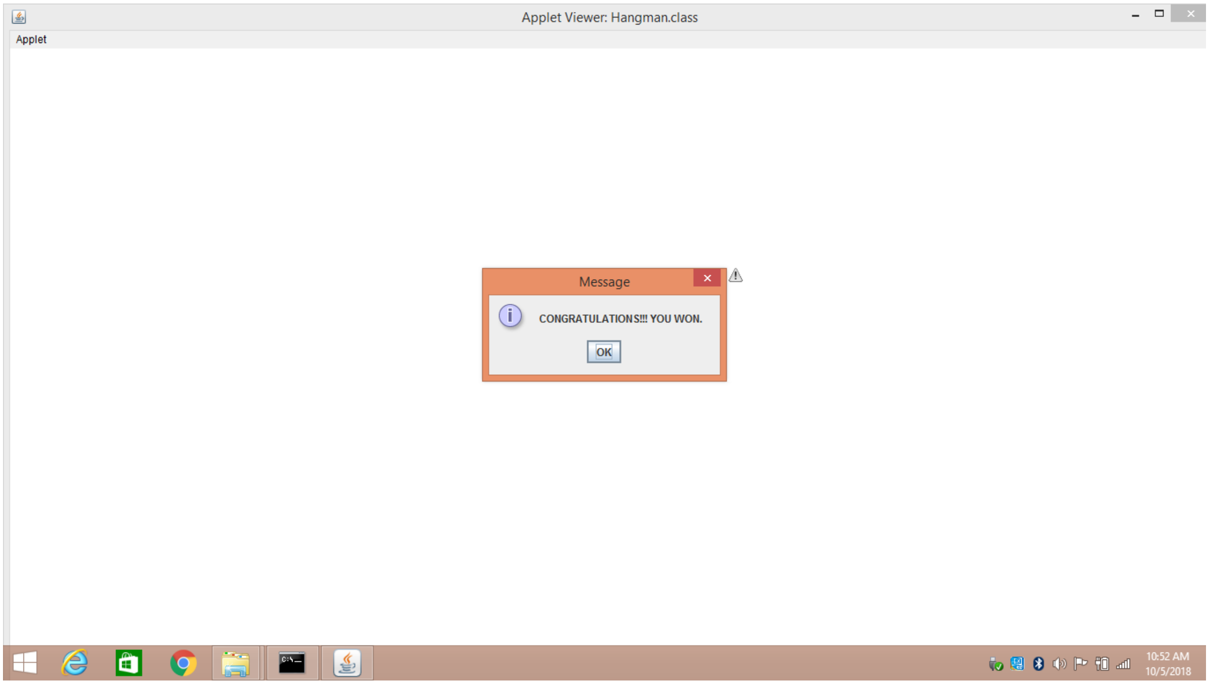


Applet Viewer: Hangman.class
Applet
THE HANGMAN GAME
A B C D E F G H I J K L M
N O P Q R S T U V W X Y Z

Level = 4 Life = 1

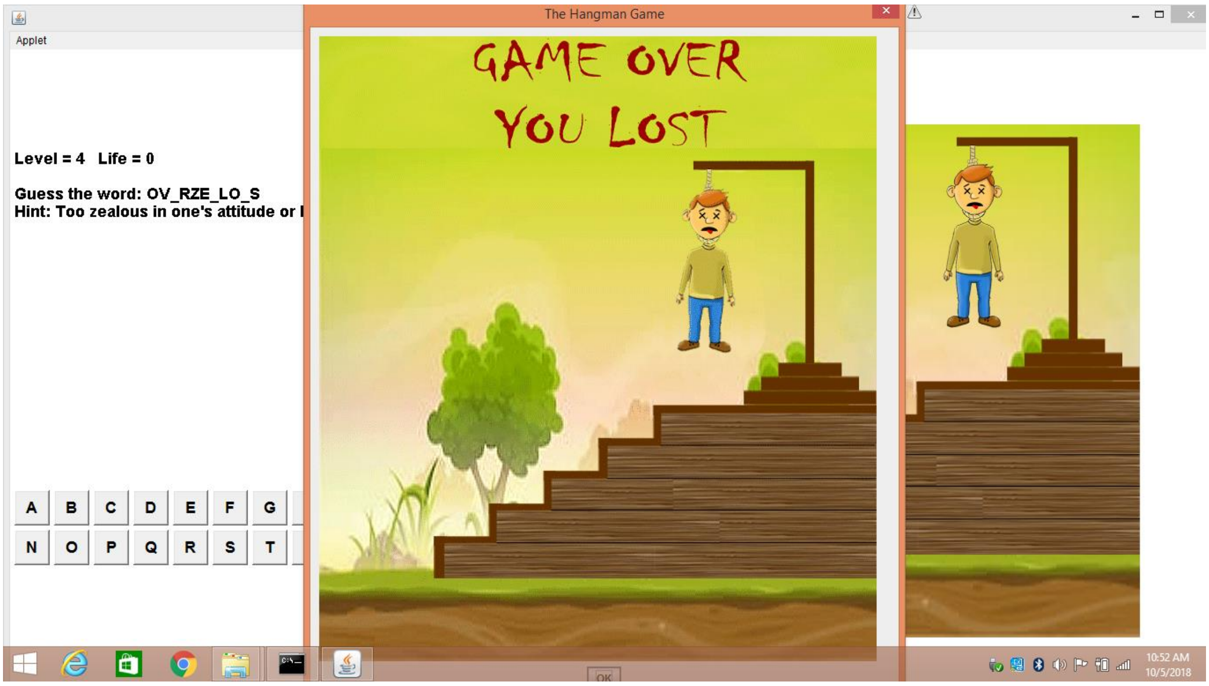
Guess the word: A_BUS_
Hint: Surprise attack by people lying in wait in a concealed position.

Fig. 3.8 – After 4 wrong inputs.



CONGRATULATIONS!!! YOU WON.

Fig. 3.9 – After winning the game.



Level = 4 Life = 0

Guess the word: OV_RZE_LO_S
Hint: Too zealous in one's attitude or l

Fig 3.10 – After losing the game.

Chapter 4

Conclusion

The Hangman game helps the players to develop their vocabulary. If it is clubbed with different fields of Science and Technology, History & Civics, etc., people could gain knowledge in an interactive way. This improves learning experience and also encourages students to understand the subject matter in context and deal with problems from a different perspective.

This game can also be used for entertainment purposes like game shows, public speeches to make them more interactive and knowledge imparting. It also helps people socialize and understand other peoples idea in a much better and diverse way i.e. it promotes lateral thinking.

Chapter 5

Future Enhancements

The game can be modified to a very large extent. Nothing is perfect in this world. So, we are also no exception. Although, we have tried our best to present the information effectively, yet, there can be further enhancements in the project. Here are a few ideas.

- It can be made online so that people from all over the world access it.
- The database can be extended to different fields and more levels can be added.
- Graphics can be added to the game in order to make it much appealing and fun to play.
- A leaderboard could be added to the game so that people from all around the world could know their ranking.

Chapter 6

References

World Wide Web

[1] https://en.wikipedia.org/wiki/Hangman_(game)

[2] https://tophat.com/blog/gamified-learning/

[3] https://www.javatpoint.com/

[4] https://www.geeksforgeeks.org/