A PROJECT REPORT ON

**GAMING WORLD**

BY

M.Abhishek 1402256

Soumya Mishra 1405169

UNDER THE GUIDANCE OF

Mr. Srinadh Babu (CSE Dept)

KIIT UNIVERSITY

BHUBANESWAR-24

*Certificate*

This is to certify that the project on **“Gaming World”** is carried out by M. Abhishek and Soumya Mishra, in partial fulfilment for the award of degree of Bachelor of Technology at KIIT University, Bhubaneswar during the academic year 2015-2016 under my supervision. The matter embodied in this project is original and has not been submitted for the award of any other degree.

**Srinadh Babu**

**(School of Comp Sc. & Engg.)**

*Acknowledgement*

It is our privilege to thank Mr.Srinadh Babu for suggesting that we take on this project and his valuable comments. His involvement and enthusiasm egged us on to take up and work on this project.

M.Abhishek (1402256)

Soumya Mishra (1405169)

Table Of Contents

***Abstract***

JAVA based project on Gaming World is a collection of famous games like Hangman, Tic-Tac-Toe and Stone-Paper-Scissor. The basic concepts of java like loops, switch cases, etc. are involved in this project. All these games are single player games where the player will be playing against the computer. Certain in-built functions of java have been used to make the game more responsive.

***Objective***

* Involve people in gaming.
* Enhance vocabulary.
* Developing analysing skills.

***Problem Definition***

The games like stone-paper-scissor & tic-tac-toe required presence of a gaming partner which was not always suitable. The purpose of this project is to facilitate gaming being a single player. The game like hangman had heavy graphics which required a lot of loading time because a lot of time was wasted in bringing this game full of graphics from secondary memory to the primary memory. This modified version of hangman keep the purpose of the game intact and at the same time reduces the loading time because of the absence of graphics.

***Technologies to be used***

The technical specification document provides an overall note of the software that has been used to design the project. It provides a complete description of how the software has been used and what areas of the project has it been implemented.

|  |  |
| --- | --- |
| **JAVA :** | The language has been used to perform all the coding required. |
| **BlueJ :** | An integrated development environment(IDE) for the Java programming language |

***Functional Requirement***

Functional requirement captures the intended behaviour of the system. This behaviour may be expressed as service tasks or functions the system is required to perform.

**Precondition** :There should be a single user.

**Brief Description :**  The user needs to choose the game he/she wants to play.

**Input :** The user need to enter the corresponding game serial number and follow the instructions.

**Output :** A message displaying the result of the game is displayed.

***Codes and Snapshots***

import java.io.\*;

import java.math.\*;

public class GAME

{

public static char a[][] = new char[3][3];

public static char temp[] = new char[9];

int k=0,x=0,y=0;

int user=0, comp=0;

int u[]=new int[100];

int c[]=new int[100];

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

{

BufferedReader in = new BufferedReader(new InputStreamReader(System.in));

System.out.println("\t\t\tWELCOME TO THE WORLD OF GAMING");

System.out.println();

System.out.println("What would you like to play?");

System.out.println("1. TIC-TAC-TOE");

System.out.println("2. HANGMAN");

System.out.println("3. STONE-PAPER-SCISSOR");

System.out.println();

int ch;

do

{

System.out.println("Please enter the number of your choice.");

ch = Integer.parseInt(in.readLine());

System.out.println();

GAME ob = new GAME();

switch(ch)

{

case 1:

ob.tic\_tac\_toe();

break;

case 2:

ob.hangman();

break;

case 3:

ob.sps();

break;

default:

System.out.println("Oooopsss... I didn't get you..!! Please re-enter.");

}

}

while(ch>=4);

}

public void tic\_tac\_toe()throws IOException

{

BufferedReader in = new BufferedReader(new InputStreamReader(System.in));

for(int i=0; i<3; i++)

{

for(int j=0; j<3; j++)

{

a[i][j] = ' ';

}

}

display(a);

int k=0, count=0;

for(int i=0; i<3; i++)

{

for(int j=0; j<3; j++)

{

temp[k++] = a[i][j];

}

}

do

{

System.out.println("Your turn");

int n = Integer.parseInt(in.readLine());

if(temp[n-1]==' ')

{

temp[n-1] = 'X';

count++;

}

else

{

System.out.println("Space occupied. Re-try.");

continue;

}

put\_array();

display(a);

if(count<9)

{

System.out.println("Computer's turn");

int s = comp\_input1();

if(s==1)

count++;

put\_array();

display(a);

}

if(count<9)

{

if(check1()==1)

{

System.out.println("You Win !!! ");

break;

}

else if(check1()==2)

{

System.out.println("Computer Wins !!!");

break;

}

else

System.out.println("Continue playing.");

}

else

{

if(check1()==1)

{

System.out.println("You Win !!! ");

break;

}

else if(check1()==2)

{

System.out.println("Computer Wins !!!");

break;

}

else

{

System.out.println("\*\*\*\*\*GAME DRAWN\*\*\*\*\*");

break;

}

}

}while(count<9);

}

public int comp\_input1()

{

int flag;

do

{

flag=0;

int rnd = (int)(Math.random()\*8);

if(temp[rnd]==' ')

{

temp[rnd]='O';

flag = 1;

}

}while(flag==0);

return flag;

}

public void put\_array()

{

int k=0;

for(int i=0; i<3; i++)

{

for(int j=0; j<3; j++)

{

a[i][j] = temp[k++];

}

}

}

public void display(char a[][])

{

for(int i=0; i<3; i++)

{

for(int j=0; j<3; j++)

{

if(j<2)

System.out.print(a[i][j]+" | ");

else

System.out.println(a[i][j]);

}

if(i<2)

System.out.println("\_ \_ \_");

}

}

public int check1()

{

int f=0;

for(int i=0;i<3;i++)

{

if(a[i][i]=='X')

{

f++;

}

}

if(f==3)

{

return 1;

}

f=0;

for(int i=0;i<3;i++)

{

for(int j=0;j<3;j++)

{

if((i+j)==2)

{

if(a[i][j]=='X')

{

f++;

}

}

}

}

if(f==3)

{

f=0;

return 1;

}

f=0;

for(int i=0;i<3;i++)

{

f=0;

for(int j=0; j<3; j++)

{

if(a[i][j]=='X')

f++;

}

if(f==3)

return 1;

}

f=0;

for(int i=0;i<3;i++)

{

f=0;

for(int j=0; j<3; j++)

{

if(a[j][i]=='X')

f++;

}

if(f==3)

return 1;

}

f=0;

for(int i=0;i<3;i++)

{

if(a[i][i]=='O')

{

f++;

}

}

if(f==3)

{

f=0;

return 2;

}

f=0;

for(int i=0;i<3;i++)

{

for(int j=0;j<3;j++)

{

if((i+j)==2)

{

if(a[i][j]=='O')

{

f++;

}

}

}

}

if(f==3)

{

f=0;

return 2;

}

f=0;

for(int i=0;i<3;i++)

{

f=0;

for(int j=0; j<3; j++)

{

if(a[i][j]=='O')

f++;

}

if(f==3)

return 2;

}

f=0;

for(int i=0;i<3;i++)

{

f=0;

for(int j=0; j<3; j++)

{

if(a[j][i]=='O')

f++;

}

if(f==3)

return 2;

}

return 0;

}

public void hangman()throws IOException

{

BufferedReader in = new BufferedReader(new InputStreamReader(System.in));

String s = "cat bat rat sat mat mad gun fan ton fly fun nice kiss wise like love pray hiss hoof prom prop drop crop right flight write left soft deaf hard wet dry cuddle mug cup disgust attention honest loyal comfortable knife building chair table jug light plate glass phone food shoes top dress jeans charger mobile finite laptop library computer point line queue weight wait race horse round monkey donkey elephant fuse bulb tube tyre wheel seat metal bomb leather ear silence happiness child affection parents husband wife family son daughter father mother house sweet home warmth human memory ring operation kid delivery hospital food nutrition college machines gym school tution coaching dot caught cot short skirt shot thought dart pierce heart cricket music valentine hostel court indoor stadium walk campus twenty three march memorable day cherished forever missed laughter cry yell scream shout shoot boot boon economic better economy books ";

int count=0, k=0, st=0;

for(int i=0; i<s.length(); i++)

{

if(s.charAt(i)==' ')

{

count++;

}

}

String arr[] = new String[count];

for(int i=0; i<s.length(); i++)

{

if(s.charAt(i)==' ')

{

arr[k++] = s.substring(st, i);

st=i+1;

}

}

int r = (int)(Math.random() \* count);

String rnd = arr[r];

System.out.println(rnd);

int l = rnd.length();

char word[] = new char[l];

char found[] = new char[l];

for(int i=0; i<l; i++)

{

found[i] = '\_';

}

int x=0;

while(l>0)

{

word[x] = rnd.charAt(x++);

l--;

}

System.out.print("The word for today is : ");

for(int i=0; i<rnd.length(); i++)

{

System.out.print("\_ ");

}

System.out.println();

int correct=0;

int wrong=0;

do

{

System.out.println();

System.out.print("Is there a letter....");

char c = (char)in.read();

System.out.println();

int flag=0;

for(int i=0; i<rnd.length(); i++)

{

if(word[i]==c)

{

word[i]=' ';

found[i]=c;

correct++;

flag=1;

}

}

for(int i=0; i<rnd.length(); i++)

{

System.out.print(found[i]+" ");

}

System.out.println();

}while(correct<rnd.length());

if(correct==rnd.length())

System.out.println("\*\*\*\*\*YOU GOT IT CORRECT\*\*\*\*\*");

}

public void sps()throws IOException

{

InputStreamReader read = new InputStreamReader(System.in);

BufferedReader in = new BufferedReader(read);

System.out.println("Enter the number of rounds you want to play");

int r=Integer.parseInt(in.readLine());

while(r>0)

{

user\_inp();

comp\_inp();

check2();

r--;

}

if(user>comp)

System.out.println("CONGRATULATIONS!! YOU WIN!!");

else if(comp>user)

System.out.println("Oooops..!! THE COMPUTER WINS!!");

else

System.out.println("GAME DRAWN. PLAY AGAIN!!");

}

void user\_inp()throws IOException

{

InputStreamReader read = new InputStreamReader(System.in);

BufferedReader in = new BufferedReader(read);

System.out.println("Enter the corresponding number of your choice :)");

System.out.println("1. Stone");

System.out.println("2. Paper");

System.out.println("3. Scissor");

int n1 = Integer.parseInt(in.readLine());

u[x++] = n1;

}

void comp\_inp()

{

int min=1;

int max=3;

int r1=(int)(min+Math.random()\*max);

c[y++] = r1;

if(r1==1)

System.out.println("The computer entered Stone");

else if(r1==2)

System.out.println("The computer entered Paper");

else

System.out.println("The computer entered Scissor");

}

void check2()

{

if(u[k]==1)

{

switch(c[k])

{

case 1:

{

u[k]=9;

c[k]=9;

break;

}

case 2:

{

u[k]=0;

c[k]=1;

comp++;

break;

}

case 3:

{

u[k]=1;

c[k]=0;

user++;

break;

}

}

}

else if(u[k]==2)

{

switch(c[k])

{

case 1:

{

u[k]=1;

c[k]=0;

user++;

break;

}

case 2:

{

u[k]=9;

c[k]=9;

break;

}

case 3:

{

u[k]=0;

c[k]=1;

comp++;

break;

}

}

}

else

{

switch(c[k])

{

case 1:

{

u[k]=0;

c[k]=1;

comp++;

break;

}

case 2:

{

u[k]=1;

c[k]=0;

user++;

break;

}

case 3:

{

u[k]=9;

c[k]=9;

break;

}

}

}

k++;

}

}

***Conclusion***

***References***

* JAVA – [www.javatpoint.com](http://www.javatpoint.com)