

Group name: SCIENCE

Group members:

1. Fabrice Irazi (F17040123)
2. Ridita Islam (F17040106)
- 3.Elizabeth Lisa Mondol (F17040114)
- 4.Wallace-Bruce Naa Koshie (F17040122)
- 5.Mukhriddin Ziyodullaev(F17040110)

Project name: LOTTERY MACHINE

LOTTERY MACHINE

INTRODUCTION

A lottery is a game of chance in which people pay for the opportunity to win prizes. Part of the money taken in by a lottery is used to award the winners and to pay the costs of administering the lottery. The money left over is profit. Lotteries are extremely popular and legal in more than a hundred countries.

Our goal is to make a software lottery machine which is very faithful and reliable.

The players just need to get a number by paying online .

PROJECT DESCRIPTION

Our software lottery machine will be such us software easy to use so that an institution can be able to use it.it must be also built with some fun action inside for making the activity of lottery not boring and interesting.

The major features of our software are given below:

- There will be some buttons as in START, STOP, RESET to make it clear and easy for the user.
- Some moving pictures and colorful background will be added to the software.
- There will be emoticons with message as in THE WINNER IS or CONGRATULATIONS etc.
- There will be some interesting music during the lottery time
- QR code scanning will be the most interesting and updated feature to make it more accessible.

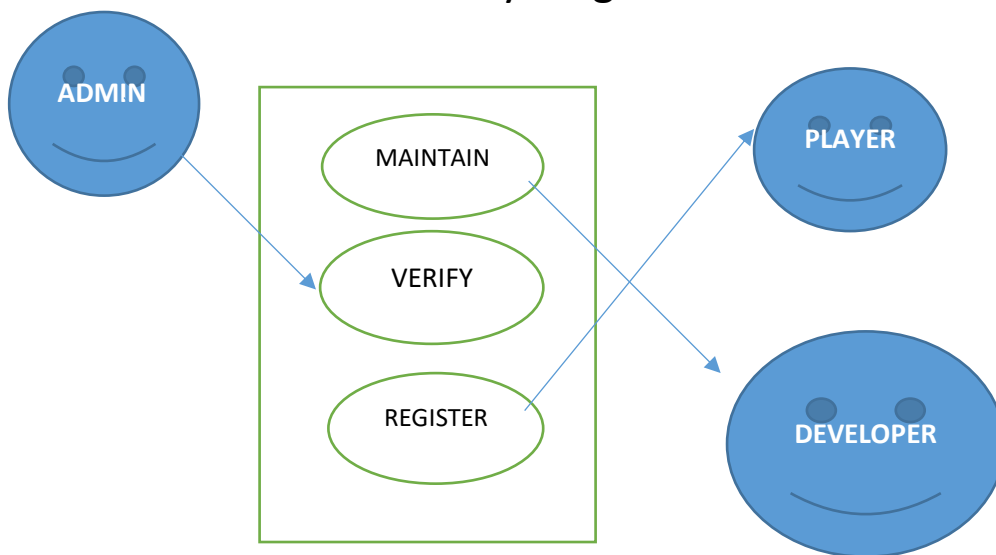
User cases:

We have three user cases: Software developer, the admin and the player.

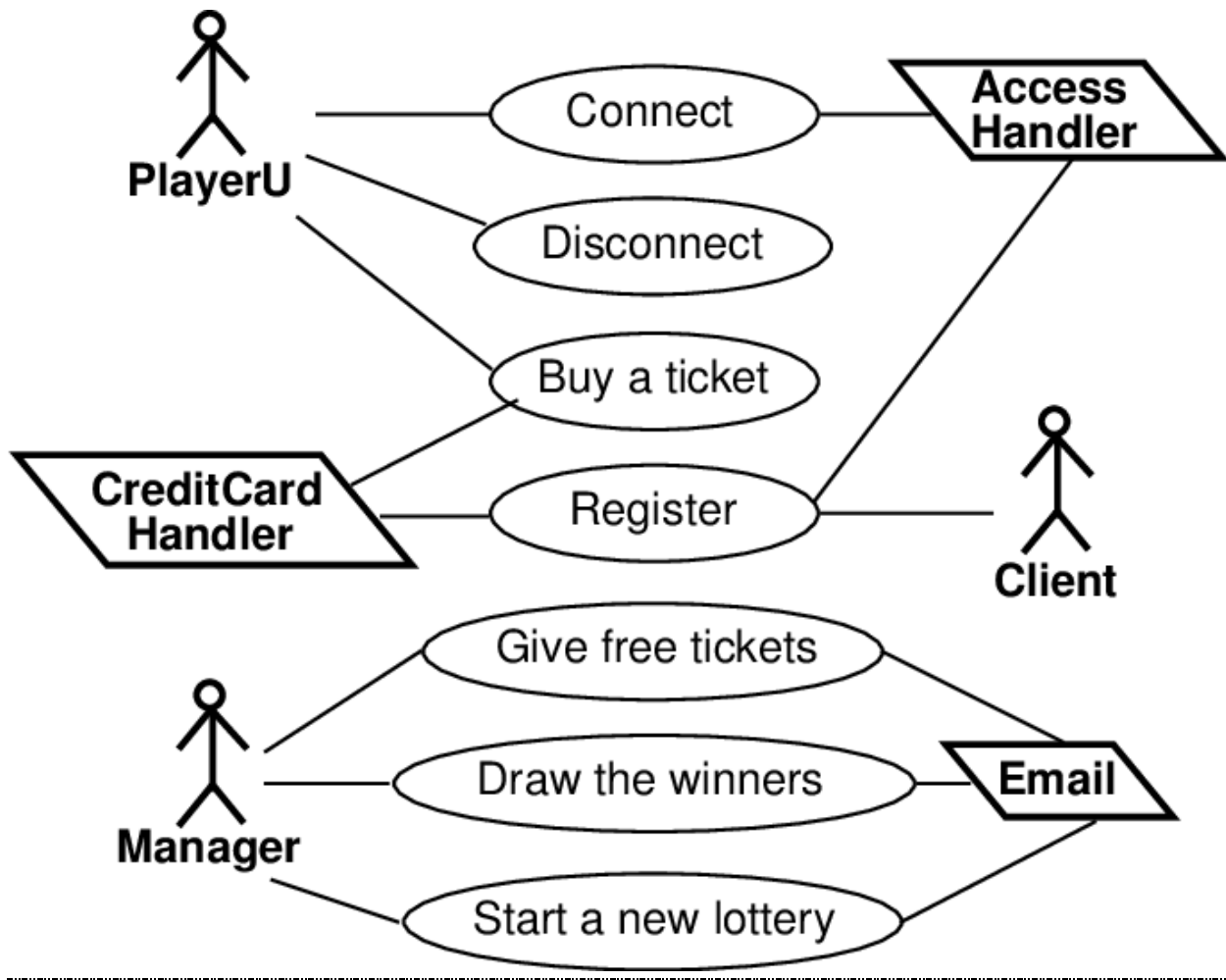
1. The software developer need the system to be maintained.
2. The admin need to verify all the information and let the player register into the system.
3. The player need to register by scanning the QR code.

ACTORS	GOALS
Software developer	Maintain the system
Admin	Verify the information
Player	Register by scanning the QR code.

User cases summary diagrams:



We can have also this diagram:



Primary and Secondary Actors

Φ Primary Actors

- Actors who initiate a scenario (use case) causing the system to achieve a goal
- recording action example the “player or admin” is a primary actor.

Φ Secondary Actors

- Actors supporting the system so primary user’s goals can be completed.

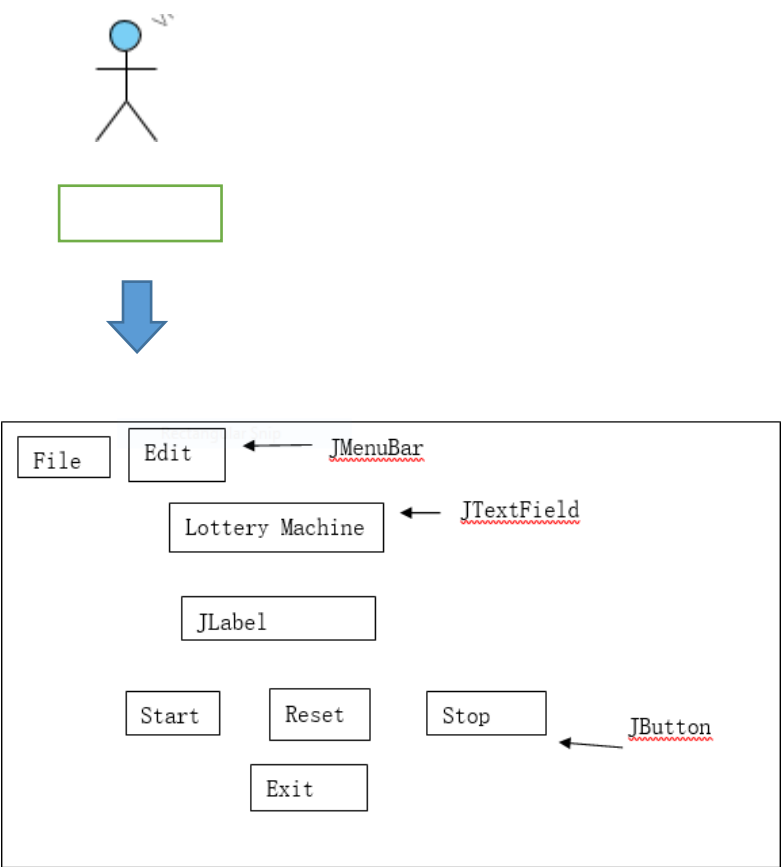
- recording action example Software Developer is a secondary actor.

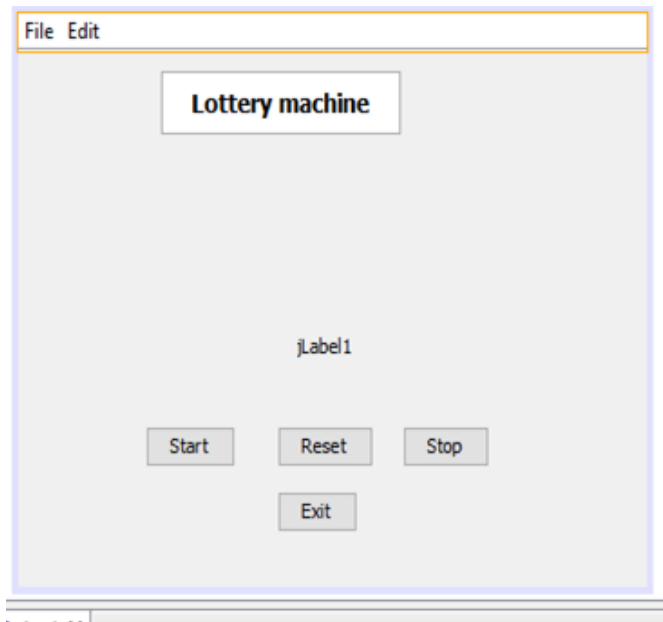
Main success scenario	1.admin turn on the software to start the game. 2. System presents the QR code . 3. System provides some music . 4. System provides a background with images moving . 5.The player scan the QR code to register . 6.The admin check the registration . 7. The admin press the start button to run the game. 8. The software developer maintain the system .
Extensions (error scenarios)	5a. the player forget to put his ID for registration . 5a.1 : system send him or her a message (please put your ID) 5a.2:the player tries again 1a.The system is not running 1a.1: the software developer check what's wrong with the

	system . 1a.2: The software developer bring a solution by finding a way to make the software running.
Variations (alternative scenarios)	The admin can fix the number of players for one game.

-

SOFTWARE'S DEMO





we can have an interface like this:



#286386312

CODE

Here is the basic source code of our project. The whole program will be uploaded in Github.

Database Connection:

In this Experiment we have created two class. One is “LotteryMachine” class and another is “Music” class.

1. LotteryMachine class
2. Music class

In the LotteryMachine class we have design a JFrame like this

1. One JMenuBar.
2. JFileChooser
3. Two JMenu
4. Two JMenuItem
5. One JTextField
6. Two JPanel
7. One JLabel
8. Four JButton

JMenuBar: We use JMenuBar to hold JMenu .

JFileChooser: We put text from file into “array” type variable. The variable is used in the thread. To get students list we used “File Chooser”. File choosers provide a GUI for navigating the file system, and then either choosing a file or directory from a list, or entering the name of a file or directory. To display a file chooser, we used the JFileChooser API to show a modal dialog containing the file chooser. The component of the “File Chooser” is located in “Menu”. There is information about our team as well.

JMenu: We use JMenu to add JMenuItem.

JMenuItem: By using JMenuItem we can access our student list.

JTextField: In this experiment we use JTextField to show text “LotteryMachine”.

JButton: After choosing file you are able to start the lottery by clicking “START” button. For generating winner we have used thread with while loop inside. The names comes from the file and displays in “text field” randomly.

To reset the machine, click button “RESET”. After that, the “text field” and list of students will be cleaned.

The purpose of “STOP” button is to stop machine and display winner name. Once you click the button, you will see winner’s name and animation

We have implement code for “LottertMachine” class and “Music” class.

Now we are going to describe those code

LotteryMachine

First of all we used thread in our code.

Thread: A thread is a program's path of execution. Most programs written today run as a single thread, causing problems when multiple events or actions need to occur at the same time. Let's say, for example, a program is not capable of drawing pictures while reading keystrokes. The program must give its full attention to the keyboard input lacking the ability to handle more than

```
import java. Lang.*;

public class Counter extends Thread

{

    public void run()
```

one event at a time. The ideal solution to this problem is the seamless execution of two or more sections of a program at the same time. Threads allows us to do this.

Coading for JMenuItem: First of all we design a JMenuItem and then we insert the following code in source file

Here is the Source code:

```
jFileChooser = new JFileChooser();  
  
    jFileChooser.showOpenDialog(this);  
  
    file = jFileChooser.getSelectedFile();  
  
    studentlist = loadFile(file);  
  
    fileloaded = true;
```

Coading for “Start” Button (JButton1): In this experiment we need a “start” button to start the machine . So we insert some code in start button to perform some action.

```
if (fileloaded){           //if statement  
  
    flag = true;  
  
    InputStream in;  
  
    try {                  //try block  
  
        Startmusic = new Music("2.wav"); //Music File name  
    } catch (MalformedURLException ex) {  
  
        Logger.getLogger(LotteryMachine.class.getName()).log(Level.SEVERE, null, ex);  
    }  
  
    Startmusic.play(); //Method to play music  
  
    myThread = new Thread((Runnable) LotteryMachine.this);  
  
    myThread.start();  
  
    jButton1.setEnabled(false); //set function  
  
}else {  
  
    JOptionPane.showMessageDialog(jPanel2, "Please, load a file from the input"); //to  
show message on jPanel to load file from project folder
```

Coading for “Stop” Button (JButton3): we need a “stop” to get a winner name. so we design a stop button and in stop we insert some code

```
flag = false;

    Startmusic.stop();//to stop the music

    jButton1.setEnabled(true);

    if(fileloaded){

        jLabel1.setIcon(new
    javax.swing.ImageIcon(getClass().getResource("win.gif")));//to show a animation
    when we will get a winner name
```

Coading for “Reset” Button (JButton2): we need a “Reset” button to clean the student list. So we design a reset button and insert some code in source of reset.

```
fileloaded = false;

    jLabel1.setText("Reset done");

    jButton1.setEnabled(true);
```

Coading for “Exit” Button (JButton4): we need to exit the program, so we also design an exit button to exit the program.

```
System.exit(0);
```

Define some variable:

```
private JFileChooser jFileChooser;
private File file;
```

```
private String[] studentlist;  
private Thread myThread;  
boolean fileloaded;  
private boolean flag = false;  
private Music Startmusic;  
private Music Stopmusic;
```

Here we used “private” access modifier

Try block: In this experiment we used two try blocks, which is given below

```
try{  
    initComponents() ;  
} catch (Exception e) {  
    ,
```

```
try{  
    try (FileReader fileReader = new FileReader(inputFile)) {  
        BufferedReader bufferedReader = new BufferedReader(fileReader);  
  
        while ((itemperline = bufferedReader.readLine()) != null){  
            itemstring.addElement(itemperline);  
        }  
  
        bufferedReader.close();  
    }  
} catch (FileNotFoundException e) {  
} catch (IOException e) {  
}  
  
int item_len = itemstring.size();  
  
String itemstring_out[] = new String[item_len];  
  
for (int i=0;i<item_len;i++){  
    itemstring_out[i] = (String)itemstring.elementAt(i);
```

Run Method:

```
public void run() {  
    int len = studentlist.length;           //declare variable for  
the length of array  
    while (fileloaded == true && flag == true){ //loop while  
variables fileloaded and flag are true  
        int i = (int)(Math.random()*len);  
                                           //choose random element  
from array  
        System.out.println(studentlist[i]);  
        jLabel1.setText(studentlist[i]);  
                                           //display the result in jLabel2.  
studentlist - is variable for the file. [i] - is element of the file.  
        try {  
            sleep(100);  
            //set the speed of loop  
        } catch (InterruptedException ex) {}  
    }  
}
```

Project Schedule and Team Structure:

SCHEDULE	We made a routine to meet at least 3 days per week from 7pm to 9 pm.
tasks for team members	We divided the work between each member. One of us is working with coding part, one is helping with researches and rest of us are preparing the reports and uploading in github.

Fabrice and Lisa did the coding part while Ridita, Bruce and Mukhrridin were helping them to find errors. As we all have less knowledge about java so we utilized each and every member's idea and put it all together to run the code. On the other hand fabrice, lisa and ridita prepared the pdf files and bruce , mukhrridin made powerpoint file for presentation and they also helped to upload the files on github. Every member contributed their hand and worked really hard. We fixed a daily schedule to meet at least three days in evening so that we can discuss our ideas and research more about our project. We made sure that everyone is working equally in this project and getting chance to share their ideas as well. During the discussion we also learnt new things together which helped us a lot to build our software successfully.

RISK ASSESSMENT

This is our first project to make a software about lottery machine. So it might be really tough and ofcourse a challenge for us. The first challenge is to design how the lottery machine will look like and to make it clear for the user. The second challenge is to correct the errors when we'll be coding for example we can have a new idea and then we don't get quickly how to put the new idea in the code by using some functions so it'll take us a long time to find the correct function. The next challenge is to check If we have all necessary tools to run the program. For example, If the IDE field does have JDK and JRE then we can run it properly.

CONCLUSION

The topic we chose was quite challenging. Sometimes we had many ideas but to execute we had to work on it so hard. We were not so good at java so the project helped us a lot to learn new things in java because we had to do a lot of researches. During our coding part we were facing some errors, from which we worked on different ideas to build the software successfully. There were some lacking on our project so our respected Laoshi gave us some good ideas and suggestions to make it look more interesting. So we hope that we'll end up with a good software and utilize these ideas on future.