**Application Programming with JAVA**

**Course Work**

**Word Count**

**Student ID: TM 221-390**

**2021-2022**

**Abstract**

* **Word Count is considered as one of the most crucial function in modern life as so many online messengers need this one or typing programs also need to count words and “Word Count” will handle this case**
* **IT expert has to run the program and Users on the other hand must download and Login in order to use**
* **In addition it will be helpful in exams based on writing to identify how many words were written**

**Motivation:**

**That program will save a lot of time by calculating in authorized and automatic way**

**Introduction:**

* **The program will allow us to save the time and it is useful for limiting the maximum limit of words in addition it serves for encouraging the writer to think of enough ideas in order to make ideas in a limited way**

**Software Requirements:**

* **Windows 7, 8, 9 and 10 is also acceptable**
* **Notepad**
* **Eclipse IDE**

**Hardware Requirement:**

* Intel i3 processor
* RAM 4 GB
* **500 GB HDD**
* **256 GB CDD**

**Code explanation:**

**import java.awt.Color;**

**//This code is used in order to change the color of appearance interface //**

**import java.awt.event.ActionEvent;**

**// This is needed to call function or method below so the program will run**

**import java.awt.event.ActionListener;**

**// I used this code for (JButton) to handle all action related to (JButton)**

**import javax.swing.JButton;**

**// This code exists here because of creating Button that will be pushed**

**import javax.swing.JColorChooser;**

**//This code is for Choosing the color of typing text so the typed text into the frame can be changed to any kind of color**

**import javax.swing.JFrame;  
// This code is used to create a Frame in which I can put further functions**

**import javax.swing.JLabel;**

**//This class is responsible to show short string or a image**

**import javax.swing.JTextArea;**

**// A JTextArea is a place in which characters and words can be written by the user**

**JLabel lb1, lb2;**

**// JLabel lb1 and lb2 are the frames in which “Character” and “Word” will be displayed**

**JTextArea ta;**

**//”Ta” needs to be used for the TextArea**

**JButton b;**

**//For doing further operations**

**JFrame f;**

**// This code is for Frame which will be appeared the next stages of writing code**

**JButton pad, text;**

**//I used this code in order to give a specific function with JButton**

**public void actionPerformed(ActionEvent e) {**

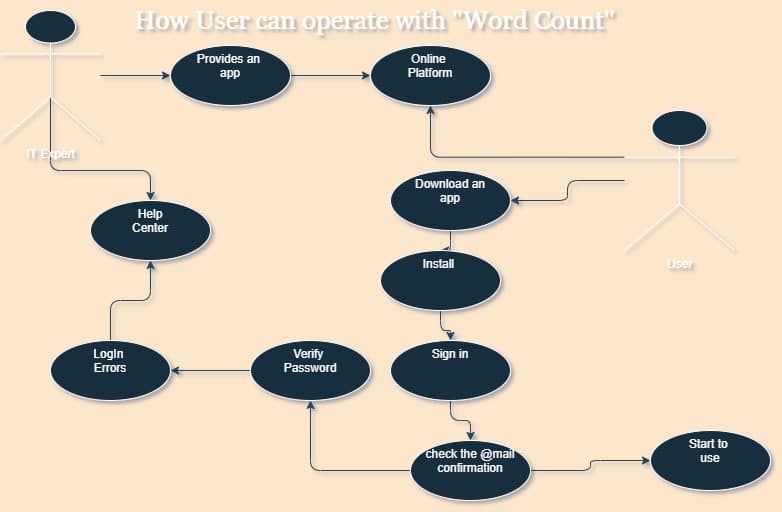
**if(e.getSource()==b) {**

**String text=ta.getText();**

**lb1.setText("Charecter" + text.length());**

**//it allows me to have a access of ActionEvent in a word this method is called to associate objects**

**Work Breakdown Structure**

****

**Figure 1: Work Breakdown**

**Modules:**

* **IT expert**
* **User**

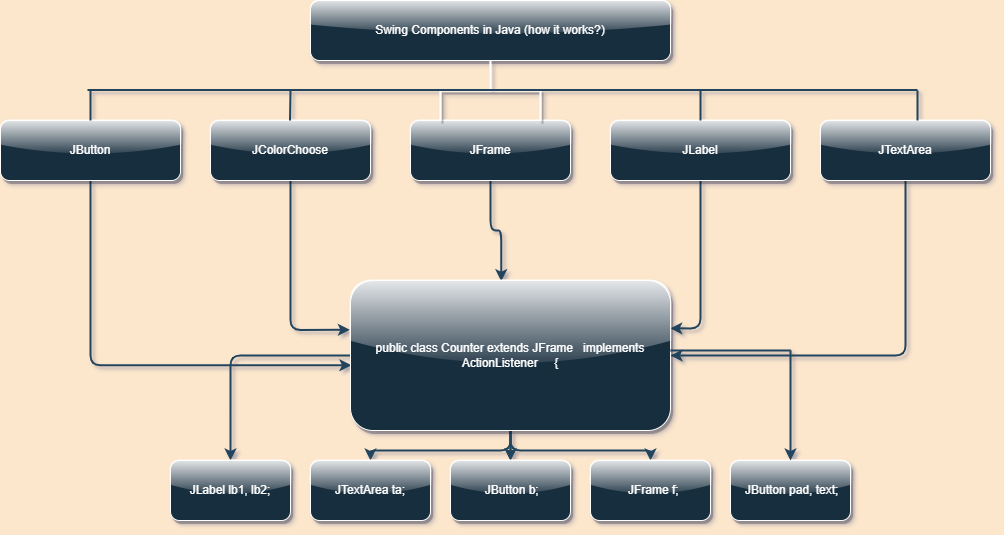
**IT expert module:**

* **The IT expert is the most obvious module of application**
* **Expert or owner of application creates and uploads an application on online platform so user can download**

**User module:**

* **The user has to go platform and download already uploaded application**
* **The next step is to install on**
* **At the same time the user can address help center if there any problem with signing in**
* **Consequently the user is able to start working**

**Here is Swing components that I have used or order explanations in diagram below**

****

**package johon;**

**import java.awt.Color;**

**import java.awt.event.ActionEvent;**

**import java.awt.event.ActionListener;**

**import javax.swing.JButton;**

**import javax.swing.JColorChooser;**

**import javax.swing.JFrame;**

**import javax.swing.JLabel;**

**import javax.swing.JTextArea;**

**public class Counter extends JFrame implements ActionListener {**

**JLabel lb1, lb2;**

**JTextArea ta;**

**JButton b;**

**JFrame f;**

**JButton pad, text;**

**Counter(){**

**super("Charecter And Word Counter");**

**lb1 =new JLabel("Charecter" );**

**lb1.setBounds(50, 50, 100, 20);**

**lb2 = new JLabel("Word:");**

**lb2.setBounds(50, 80, 100, 20);**

**ta = new JTextArea();**

**ta.setBounds(50, 110, 300, 200);**

**b = new JButton ("click" );**

**b.setBounds(50, 320, 80, 30);**

**b.addActionListener(this);**

**pad = new JButton("pad color");**

**pad.setBounds(140, 320, 80, 30);**

**pad.addActionListener(this);**

**text = new JButton ("Text Color");**

**text.setBounds(260,320 ,110 ,30 );**

**text.addActionListener(this);**

**add(lb1); add(lb2); add(ta); add(b); add(pad); add(text);**

**setSize(400,400);**

**setLayout(null);**

**setVisible(true);**

**setDefaultCloseOperation(EXIT\_ON\_CLOSE);**

**setLocation(320,190);**

**}**

**@Override**

**public void actionPerformed(ActionEvent e) {**

**if(e.getSource()==b) {**

**String text=ta.getText();**

**lb1.setText("Charecter" + text.length());**

**String words[]=text.split("//s");**

**lb2.setText("Words:"+text.length());**

**} else if(e.getSource()==pad) {**

**Color c = JColorChooser.showDialog(this,"Choose color", Color.BLACK);**

**ta.setBackground(c);**

**}else if (e.getSource()==text) {**

**Color c =JColorChooser.showDialog(this, "Choose color", Color.black );**

**ta.setForeground(c);**

**}**

**}**

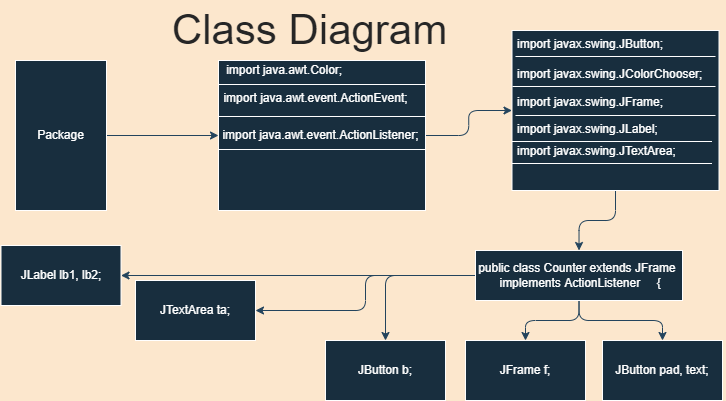
**public static void main(String[] args) {**

**new Counter();**

**}**

**}**

**Here is a Class Diagram which explanation goes deeper**

****

**Advantages of using Word count:**

1. **Saves time. It means people do not need to count word as they did in the past**
2. **Will be handful in many kind of exam held by university, official councils and to name but a few**
3. **An application can be used offline**

**Disadvantages:**

1. **Someone will not like the simple interface**
2. **It requires some of the software**
3. **It is needed to be launched by the user and user must sign first**

**Conclusion:**

* **To sum up, using word count will be beneficial not only in a saving time but also comfortable interface is easy to understand quickly without much ado**
* **This project will be utilized in exams such as IELTS especially in (Writing) sections**

**Reference**

[**https://github.com/JakhongirAken/Application-with-Java-.git**](https://github.com/JakhongirAken/Application-with-Java-.git)