**Alarm Clock Operation in Mobile Phones**

**A PROJECT REPORT**

***Submitted by***

***in partial fulfillment for the award of the degree***

***of***

**BACHELOR OF ENGINEERING**

***in***

**--------**

****

**NOORUL ISLAM CENTRE FOR HIGHER EDUCATION**

**NOORUL ISLAM UNIVERSITY**

**Kumaracoil-629180**

**NOVEMBER 2015**

**NOORUL ISLAM CENTRE FOR HIGHER EDUCATION**

**NOORUL ISLAM UNIVERSITY**

****

**BONAFIDE CERTIFICATE**

Certified that this project report **“An alarm clock operation in Mobile Phones”**, is the bonafide work of----------- who carried out the project work under my supervision

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Project Guide**   |  | | --- | | **----------------** | | **Assistant Professor** | | **Department of IT** | | **N.I University** | | **Head of the Department**   |  | | --- | | **----------------------** | | **Associate Professor & Head** | | **Department of IT** | | **N.I University** | |
|  |  |

Submitted for the MiniProject examination held on\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**INTERNAL EXAMINER EXTERNAL EXAMINER**

**NOORUL ISLAM CENTRE FOR HIGHER EDUCATION**

**NOORUL ISLAM UNIVERSITY**

**DECLARATION**

I Affirm That The Mini Project Title**“An alarm clock operation in Mobile Phones”**, Being Submitted in partial fulfillment for the award of B.Tech(Information Technology) degree is the original work carried out by me. It has not formed the part of any other project work submitted for award of any degree, either in this or any other University.

Name:

Register No:

(Signature of candidate)

I certify that the declaration made above by the candidate is true.

Signature of the Guide

Name:

Designation:

**ABSTRACT**

The Real-time clock (RTC) is a computer clock that keeps track of the current time. Although the term often refers to the devices in personal computers, servers and embedded systems, RTCs are present in almost any electronic device which needs to keep accurate time. In some digital cellular systems like phones, real-time clock is synchronized to the precise local time given by the base station to which the phone is communicating with current. This feature is known as automatic local time adjustment. To implement automated and programmable functions, a real time clock within the wireless device is generally required. The real time clock provides a precise representation of time, which can be compared to stored values in registers to provide automated functions. The clock is typically powered from batteries. For setting Alarm in mobile phones, first set the time manually or automatically done by the base station. Then set the time when to make an alarm. After setting the alarm, it waits for reaching the trigger. When reach the trigger, it starts buzzer.

**ACKNOWLEDGEMENT**

All praise, glory and honor be to the Lord Almighty , source of all acknowledgement for his gracious presence and guidance that enable us to complete this in due time.

We take the chance of this moment to thank our Chancellor **Janab Dr. A.P. Majeed Khan,**for providing us an opportunity to do the project. We also express our hearty thanks to our pro-vice- chancellor**Dr. N. Chandrasheker, B.E, Ph.D.,** for encouraging us in the project.

We also express our sense of gratitude to our Head of the Department **Dr. R. MathusoothanaS.Kumar, M.E, Ph.D.,** for providing us all the facilities and encouraging for developing the project.

We would like to express our gratitude to our Project Coordinator **Mr--------------------**Assistant Professor ---------------Department for their enthusiastic support shown towards our project.

With deep sense of gratitude, we extent our earnest and sincere thanks to our guide **Mr. --------------**Assistant Professor --------------Department for his kind guidance and encouragement during this project.

With great enthusiasm we express our thanks to all our department faculty members and technical staff members for providing necessary information and their sustained interest in our fruitful completion. Our heartfelt gratitude and thanks to our parents and friends for their perpetual support.

**Table of Content**

|  |  |  |
| --- | --- | --- |
|  | **TITLE** | **PAGE NO** |
| **1** | Introduction |  |
|  | 1.1 Overview |  |
|  | 1.2 Need for Project |  |
|  | 1.3 Objectives |  |
|  | 1.4 System Requirements |  |
|  | 1.4.1 Operating System |  |
|  | 1.4.2 Hardware Requirements |  |
|  | 1.4.3 Software Requirements |  |
| **2** | Existing System |  |
| **3** | Proposed System |  |
| **4** | Block Diagram |  |
|  | 4.1`Time Setting |  |
|  | 4.2 Setting Alarm |  |
|  | 4.3 Waiting for alarm Time |  |
|  | 4.4 Waiting for Set |  |
|  | 4.5 Alarming |  |
|  | 4.6 Snoozing |  |
| **5** | Test and Implementation |  |
|  | 5.1 Time Display Module |  |
|  | 5.2 Initial Module |  |
|  | 5.3 Alarm Setting Module |  |
| **6** | Result and Discussion |  |
| **7** | Software Codes |  |
| **8** | Conclusion and Future Work |  |
|  | 8.1 Conclusion |  |
|  | 8.2 Future Work |  |

**1. INTRODUCTION**

**1.1 OVERVIEW**

An alarm clock is a clock that is designed to make a sound, or some other signal, at a specific time. The primary utility of these clocks is to awaken people from their night's sleep or short naps; they are sometimes used for other reminders as well. Most use sound; some use light or vibration. Some have sensors to identify when a person is in a lighter stage of sleep, in order to avoid waking someone who is deeply asleep, which causes tiredness, even if the person has had adequate sleep. To stop the sound or light, a button or handle on the clock is pressed and clocks automatically stop the alarm if left unattended long enough.The mobile phones feature the ability for the user to set the alarm's [ringtone](https://en.wikipedia.org/wiki/Ringtone), and in some cases music can be [downloaded](https://en.wikipedia.org/wiki/Downloaded) to the phone and then chosen to play for an alarming

**1.2 NEED FOR PROJECT**

The design of an alarm clock, which will be useful for almost all demographics including deaf, blind, and over sleepers. The deaf can feelthe Vibration of the device when it is attached with human body . The visually challenged person may utilize this system by activating the voice in the mobile. Due to the accelerated pace of life, a clock is necessary to get up on time and tell the time when a person needs it. This design posed interesting problems. Generally devices for the deaf are unusually loud and could annoy the sensitive ears of the blind or for that matter of normal hearing people.

**1.3 OBJECTIVES**

􀂃Timing and alarming.

􀂃Equipped with additional features for the deaf, blind, and over sleepers.

􀂃Voice activation.

􀂃Color code AM/PM.

**1.4. SYSTEM REQIREMENTS**

**1.4.1 Operating System**

Windows Xp/Windows7

**1.4.2 HARDWARE REQUIREMENTS**

* Cellular Phones
* Intel® Pentium® CPU [B940@2.00GHz](mailto:B940@2.00GHz) 2.00 GHz
* RAM 512MB and more

**1.4.3 SOFTWARE REQIREMENTS**

* Net Beans IDE6.1
* Internet Explorer

**2.Existing System**

A classic analog alarm clock has an extra hand or inset dial that is used to specify the time at which to activate the alarm. Alarm clocks are also found on mobile phones. Traditional mechanical alarm clocks have one or two bells that ring by means of a mainspring that powers a gear to propel a hammer back and forth between the two bells or between the interior sides of a single bell. In some models, the back encasement of the clock itself acts as the bell. In an electric bell-style alarm clock, the bell is rung by an electromagnetic circuit and armature that turns the circuit on and off repeatedlyand point to a location on the dial that represents the approximate time. On the analog clock, the time is represented by hands that spin around a dial.



Digital clocks typically use the 50 or 60 [hertz](https://en.wikipedia.org/wiki/Hertz) oscillation of [AC power](https://en.wikipedia.org/wiki/AC_power) or a 32,768 hertz [crystal oscillator](https://en.wikipedia.org/wiki/Crystal_oscillator) as in a [quartz clock](https://en.wikipedia.org/wiki/Quartz_clock) to keep time. Most [digital](https://en.wikipedia.org/wiki/Digital_data) clocks display the hour of the day in [24-hour format](https://en.wikipedia.org/wiki/24-hour_clock); in the United States and a few other countries, a more commonly used hour sequence option is [12-hour format](https://en.wikipedia.org/wiki/12-hour_clock)(with some indication of AM or PM). Emulations of analog-style faces often use an [LCD](https://en.wikipedia.org/wiki/LCD) screen, and these are also sometimes described as "digital".

Digital clocks that run on mains electricity and have no battery must be reset every time the power is cut off or if they are moved. Even if power is cut off for a second, most clocks will still have to be reset. This is a particular problem with [alarm clocks](https://en.wikipedia.org/wiki/Alarm_clock) that have no "battery" backup, because even a very brief [power outage](https://en.wikipedia.org/wiki/Power_outage) during the night usually results in the clock failing to trigger the alarm in the morning.



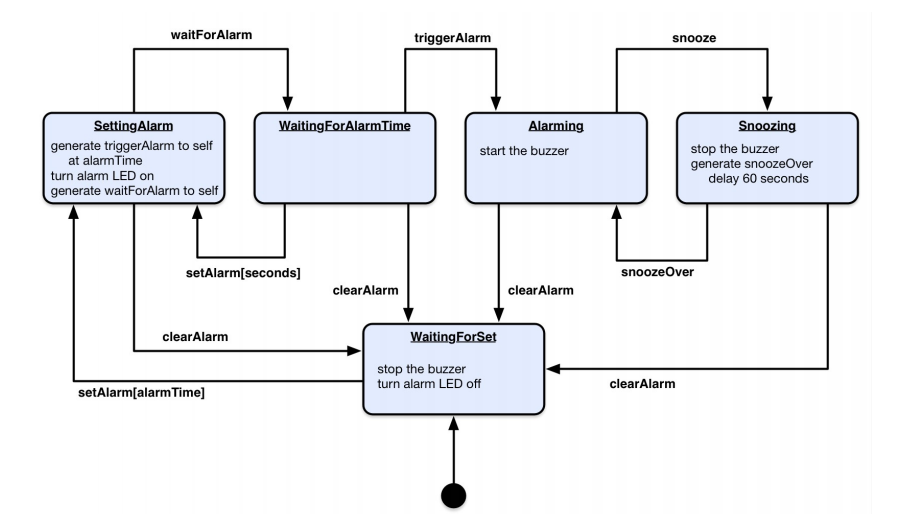
Many choices exist but this brun is hard to pass up for its classic good looks. Many people less in love with the pip pip noise it makes but perhaps you will eventually find yourself walking up just before it goes off.

**3. Proposed System**

The mobile alarm clocks can make variety of noises. Simple battery-powered alarm clocks make a loud buzzing or beeping sound to wake a sleeper, while novelty alarm clocks can speak, laugh, sing, or play sounds from nature. Some alarm clocks can set multiple alarms, a useful feature for couples who have different waking up schedules. For setting Alarm in mobile phones, first set the time manually or automatically done by the base station.On a digital clock, a numeric display indicates the exact time. You can set up multiple alarms, set some active and disable others, and the app doesn’t have to be in the foreground for it to wake you up when the time comes. It’s not particularly feature rich, but if you just want something to go off to wake you up or remind you to do a thing, this will work just fine and not cost you a cent.Then set the time when to make an alarm.It still has all of the things that made it great originally—as many alarms as you want to set, puzzles or challenges to snooze or disable the alarm, a tablet-friendly version and an updated UI, auto-snooze and auto-dismiss, a built-in timer, the option to wake to music instead of sounds, and so on.

* To make alarm in mobile is easy.
* In this, alarm time is accurate
* Do not fails to make buzzer even though the mobile is switched off.
* We can set as many alarms as you want, at any time of day, in both 12 and 24 hour formats
* Create pre-alarm, alarm, and snooze “profiles” where you can customize the alarm sound, volume, fade-in volume, duration of the alarm, even display brightness. Comes with three profiles by default: “workday,” “day off,” and “nap.”
* Can be configured to play music, playlists, or even an internet radio stream as an alarm, so you can wake up to your favorite song or playlist, and can even be configured to only play songs by specific genres or artists
* Can be configured to sound an alarm that slowly increases in volume or vibration intensity until you dismiss it or after successive snoozes
* Can schedule multiple alarms, each with different songs or playlists, or to play a random song or playlist
* Can be configured to launch a specific application when the alarm goes off
* Can display a countdown timer on-screen until the alarm sounds, making the app useful for a timer as well as an alarm clock
* Has a built-in “dock” mode and a “night mode,” where the clock and alarm settings are displayed on-screen or the current time is displayed on a dimmed display, making it useful as a desk clock or bedside clock, respectively
* Supports speech input for alarms, the same way the default clock and alarm does
* Allows you to auto-dismiss or auto-snooze alarms
* Allows you to customize the color and background of the alarm display
* Has a downloadable home screen widget, and downloadable extra alarm sounds if you don’t like the default selection
* Has a built-in flashlight

**4.BLOCK DIAGRAM**



**4.1 Time Setting**

* First set the time of the mobile phone for the current time.
* The time must be the time of the current place, then only it make alert the exact time period.

**4.2 Setting Alarm**

* Then set the time for making an alarm.
* Trigger the alarm to set at time to turn on the alarm.
* After the time setting, it go for either the waiting for alarm time or it go to waiting for set block.

**4.3 Waiting for alarm Time**

* Exists in this block till the triggering condition satisfied.
* This goes to the Alarming block, when it reaches the trigger.
* Any failure in this function, it goes to the “setting alarm” block or it reset the alarm and goes to the “waiting for set” block.

**4.4 Waiting for set**

* Either failure in the alarm setting or the alarm ringing comes to “waiting for set block”.
* It clear the time that is previously set or stops alarm and reset the alarm time.

**4.5 Alarming**

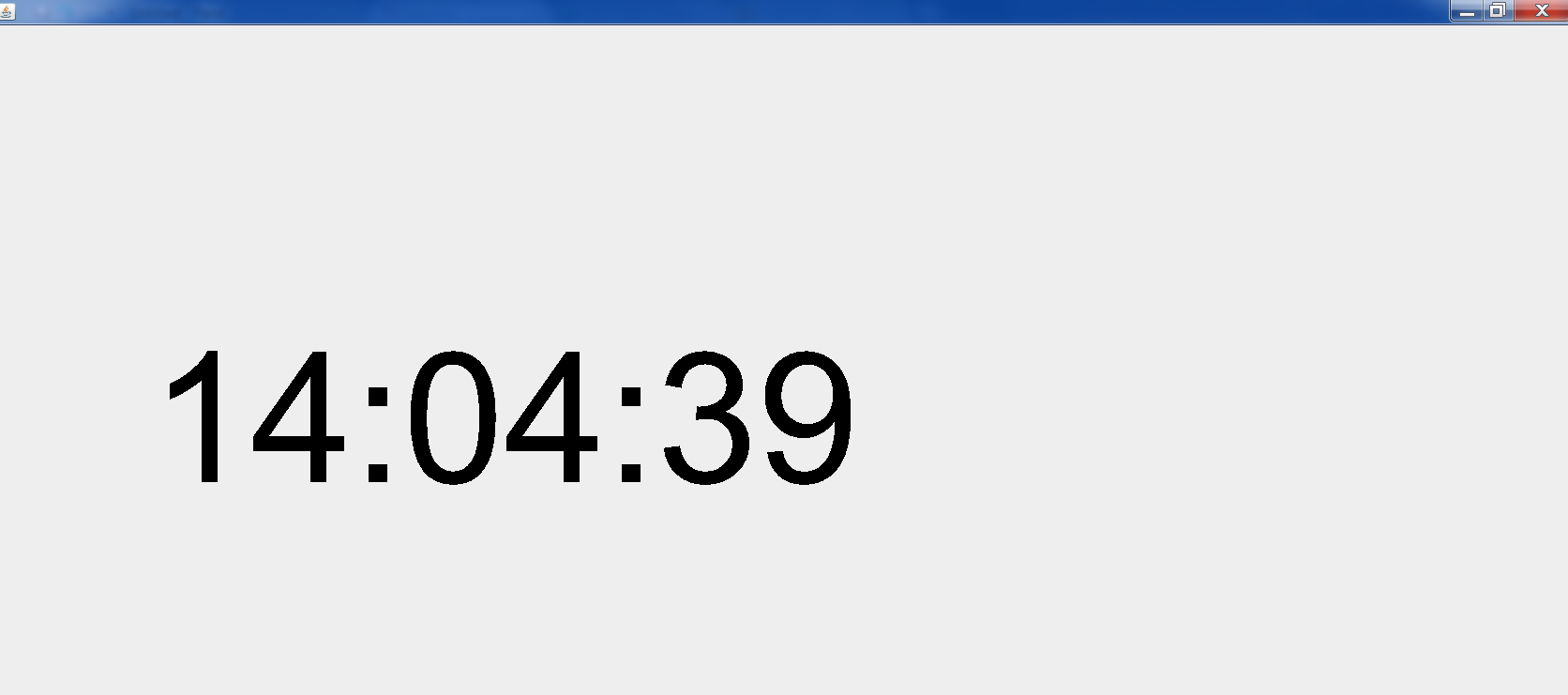
* Processor scheduling delays may prevent the process from handling the signal as soon as it is generated.
* When reaches the trigger, it starts buzzer.

**4.6 Snoozing**

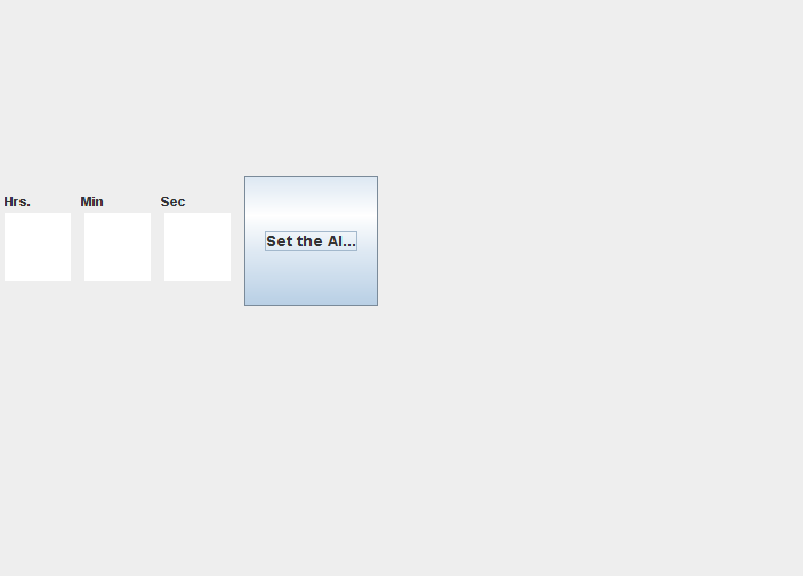
* The 'snooze' function no longer works.
* It stops alarm and snooze over delay 60 seconds after buzzing and it go for the waiting for set to reset the alarm.

**5. TEST AND IMPLEMENTATION**

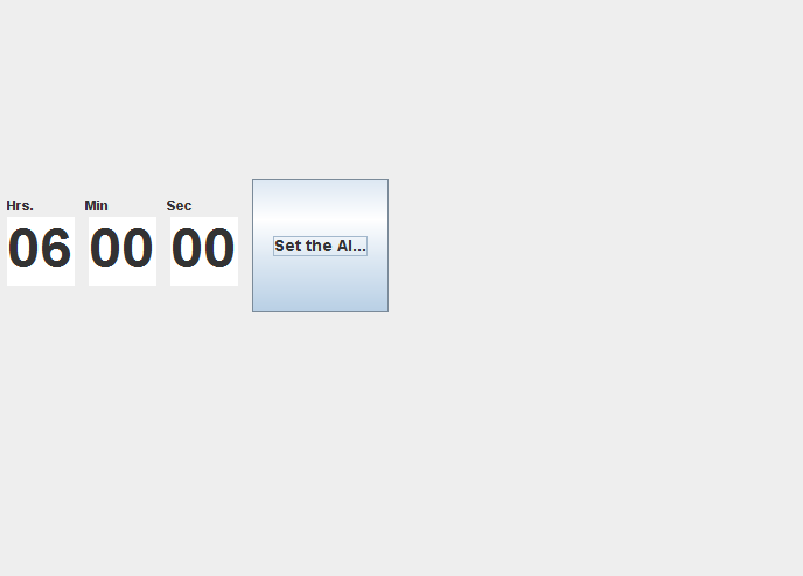
**5.1 TIME DISPLAY MODULE**

****

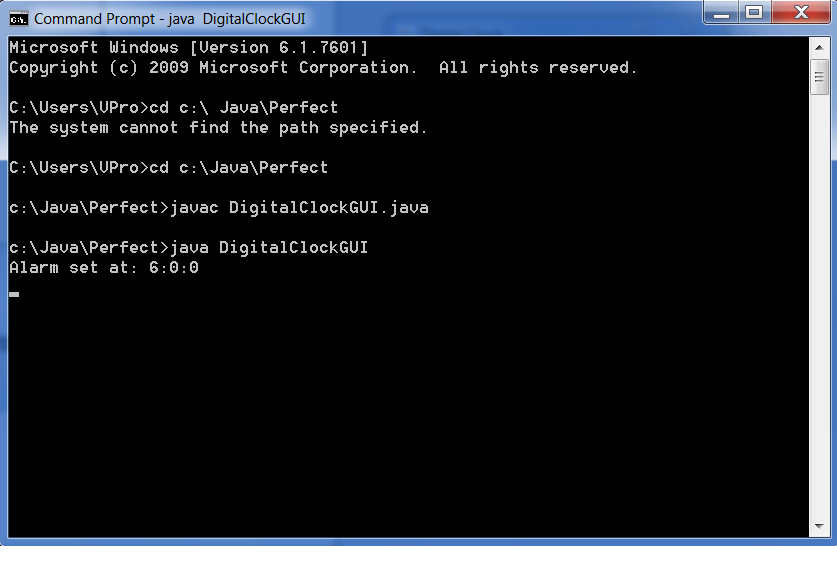
**5.2 INITIAL MODULE**

****

**5.3 ALARM SETTING MODULE**

****

**6. RESULT AND DISCUSSION**

****

**7. SOFTWARE CODES**

**SimpleDigitalClock.java**

importjava.awt.Color;

importjava.awt.Dimension;

importjava.awt.Font;

importjava.awt.Graphics;

importjava.awt.event.ActionEvent;

importjava.awt.event.ActionListener;

importjava.util.Calendar;

importjavax.swing.JFrame;

importjavax.swing.JPanel;

importjavax.swing.Timer;

importjavax.swing.WindowConstants;

public class SimpleDigitalClock {

public static void main(String[] args) {

JFrame f = new JFrame();

f.setDefaultCloseOperation(WindowConstants.EXIT\_ON\_CLOSE);

DigitalClockmyClock = new DigitalClock();

f.add(myClock);

f.pack();

f.setVisible(true);

}

static class DigitalClock extends JPanel {

String stringTime;

int hour, minute, second;

String correctionHour = "";

String correctionMinute = "";

String correctionSecond = "";

public void setStringTime(String xyz) {

this.stringTime = xyz;

}

publicintfindMinimumBetweenTwoNumbers(int a, int b) {

return (a <= b) ? a : b;

}

DigitalClock() {

Timer t1 = new Timer(1000, new ActionListener() {

public void actionPerformed(ActionEvent e) {

repaint();

}

});

t1.start();

}

@Override

public void paintComponent(Graphics g) {

super.paintComponent(g);

Calendar now = Calendar.getInstance();

hour = now.get(Calendar.HOUR\_OF\_DAY);

minute = now.get(Calendar.MINUTE);

second = now.get(Calendar.SECOND);

if (hour < 10) {

this.correctionHour = "0";

}

if (hour >= 10) {

this.correctionHour = "";

}

if (minute < 10) {

this.correctionMinute = "0";

}

if (minute >= 10) {

this.correctionMinute = "";

}

if (second < 10) {

this.correctionSecond = "0";

}

if (second >= 10) {

this.correctionSecond = "";

}

setStringTime(correctionHour + hour + ":" + correctionMinute+ minute + ":" + correctionSecond + second);

g.setColor(Color.BLACK);

int length = findMinimumBetweenTwoNumbers(this.getWidth(),this.getHeight());

Font myFont = new Font("SansSerif", Font.PLAIN, length / 5);

g.setFont(myFont);

g.drawString(stringTime, (int) length/6, length/2);

}

@Override

public Dimension getPreferredSize() {

return new Dimension(200, 200);

}

}

}

**DigitalClockGUI.java**

importjava.awt.Color;

importjava.awt.EventQueue;

importjava.awt.Font;

importjava.awt.event.ActionEvent;

importjava.awt.event.ActionListener;

importjavax.swing.JButton;

importjavax.swing.JFrame;

importjavax.swing.JLabel;

importjavax.swing.JPanel;

importjavax.swing.SwingConstants;

importjavax.swing.border.EmptyBorder;

importjavax.swing.JTextArea;

public class DigitalClockGUI extends JFrame {

/\*\*

\*

\*/

private static final long serialVersionUID = 1L;

privateJPanelcontentPane;

privateJLabelclockLabel;

private String time;

private String \_hours;

private String \_minutes;

private String \_seconds;

private String myHour;

private String myMin;

private String mySec;

intalarmHour;

intalarmMinute;

intalarmSecond ;

//private Clock myclock;

privateJTextAreatextArea\_hrs;

privateJTextAreatextArea\_min;

privateJTextAreatextArea\_sec;

/\*\*

\* Launch the application.

\*/

public static void main(String[] args) {

EventQueue.invokeLater(new Runnable() {

public void run() {

try {

DigitalClockGUI frame = new DigitalClockGUI();

frame.setVisible(true);

} catch (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the frame.

\*/

publicDigitalClockGUI() {

//myclock = new Clock(DigitalClockGUI.this);

//myclock.execute();//execute the Clock

setTitle("Digital Clock");

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setBounds(100, 100, 339, 324);

contentPane = new JPanel();

contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));

setContentPane(contentPane);

contentPane.setLayout(null);

JButtonbtnAlarm = new JButton("Set the Alarm");

btnAlarm.setFont(new Font("Arial", Font.BOLD, 14));

btnAlarm.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent arg0) {

/\*\*

\* Gets 3 different values from the textAreas

\*/

myHour = new String (textArea\_hrs.getText());

myMin = new String (textArea\_min.getText());

mySec = new String (textArea\_sec.getText());

/\*\*

\* Transform the 3 values from the textAreas to Integer

\*/

int \_alarmHour = Integer.parseInt(myHour);

int \_alarmMinute = Integer.parseInt(myMin);

int \_alarmSecond = Integer.parseInt(mySec);

/\*\*

\* Set the alarm from the textArea values

\*/

alarmHour = \_alarmHour;

alarmMinute = \_alarmMinute;

alarmSecond = \_alarmSecond;

//myclock.setBoolean(true);

//myclock.setAlarm(alarmHour, alarmMinute, alarmSecond); // set the alarm

System.out.println("Alarm set at: " + alarmHour +":"+ alarmMinute + ":" + alarmSecond); // shows on the console what time the alarm has been set up

}

});

btnAlarm.setBounds(216, 172, 117, 124);

contentPane.add(btnAlarm);

clockLabel = new JLabel("");

clockLabel.setHorizontalAlignment(SwingConstants.CENTER);

clockLabel.setFont(new Font("Arial", Font.BOLD, 50));

clockLabel.setBounds(30, 25, 277, 92);

contentPane.add(clockLabel);

textArea\_sec = new JTextArea();

textArea\_sec.setFont(new Font("Arial", Font.BOLD, 50));

textArea\_sec.setBounds(146, 207, 58, 65);

contentPane.add(textArea\_sec);

textArea\_min = new JTextArea();

textArea\_min.setFont(new Font("Arial", Font.BOLD, 50));

textArea\_min.setBounds(76, 207, 58, 65);

contentPane.add(textArea\_min);

textArea\_hrs = new JTextArea();

textArea\_hrs.setFont(new Font("Arial", Font.BOLD, 50));

textArea\_hrs.setBounds(6, 207, 58, 65);

contentPane.add(textArea\_hrs);

JLabellblHrs = new JLabel("Hrs.");

lblHrs.setBounds(6, 188, 61, 16);

contentPane.add(lblHrs);

JLabellblMin = new JLabel("Min");

lblMin.setBounds(73, 188, 61, 16);

contentPane.add(lblMin);

JLabellblSec = new JLabel("Sec");

lblSec.setBounds(143, 188, 61, 16);

contentPane.add(lblSec);

}

public void setTime(int hour, int minutes, int seconds){

/\*\*

\* this method makes the clock work on the ClockLabel

\*/

\_hours=Integer.toString(hour);

\_minutes=Integer.toString(minutes);

\_seconds=Integer.toString(seconds);

/\*\*

\* the if help to put a 0 if it is necessary

\*/

if(hour < 10){

\_hours = "0" + \_hours;

}

if(minutes < 10)

{

\_minutes = "0" + \_minutes;

}

if(seconds < 10)

{

\_seconds = "0" + \_seconds;

}

time = \_hours + ":" + \_minutes + ":" + \_seconds;

clockLabel.setText(time); //Paint the clock on the ClockLabel

}

public void invokeAlarm(){

/\*\*

\* Changes the color of the background when the alarm starts

\*/

Color d = new Color((float) Math.random(), (float) Math.random(),(float) Math.random());

contentPane.setBackground(d);

textArea\_hrs.setText("");

textArea\_min.setText("");

textArea\_sec.setText("");

}

}

**8. CONCLUSIONAND FUTURE WORK**

**8.1 CONCLUSION**

In this project we have described the specification and architecture of “Alarm Clock operation in Mobile Phones”. In previous analog alarm clock have some disadvantages comparing with our project. In our project, we overcome the drawbacks in the existing system. This system should be very useful to everyone.

**8.2 FUTURE WORK**

In future by improving the performance and efficiency may design any other type of alarm clocks.