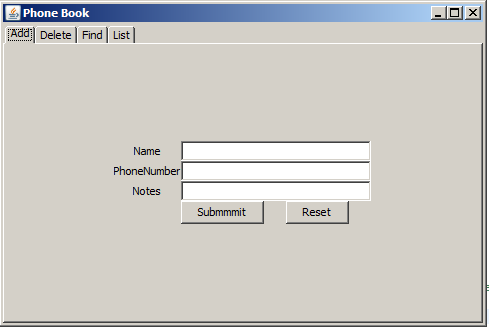
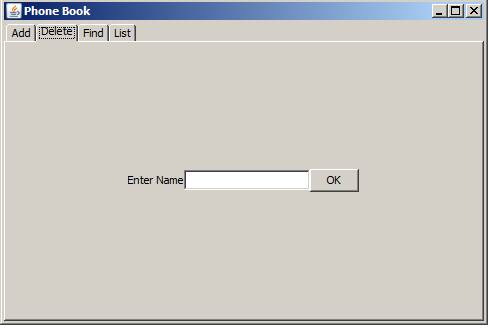
**Ye, Zhengkun**

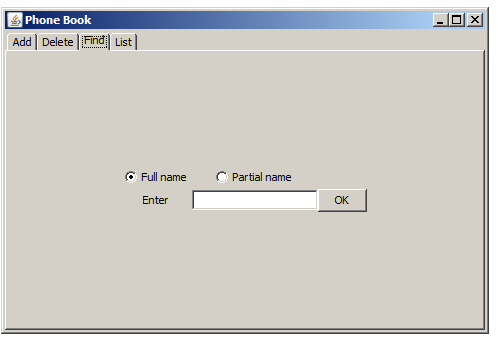
**EECS 1510**

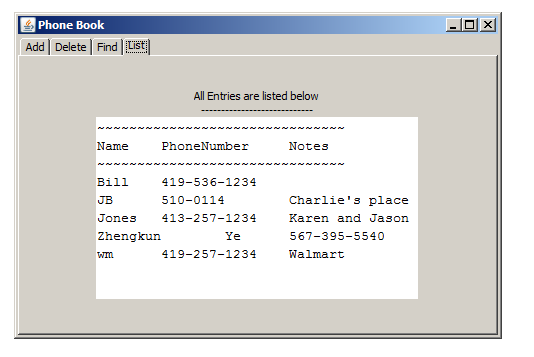
**Final Project**

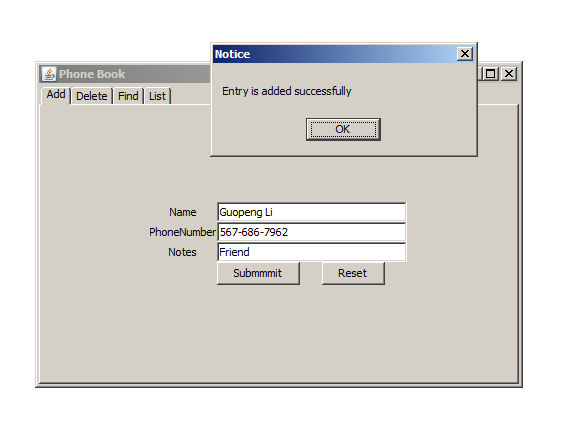
**04/23/2014**

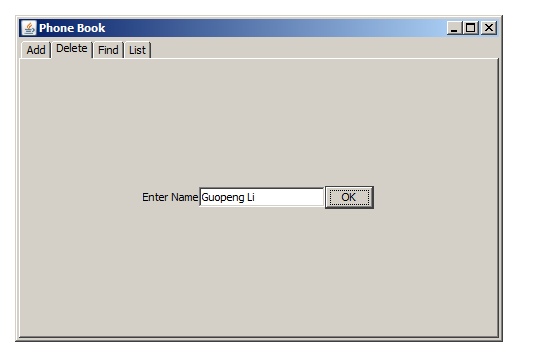


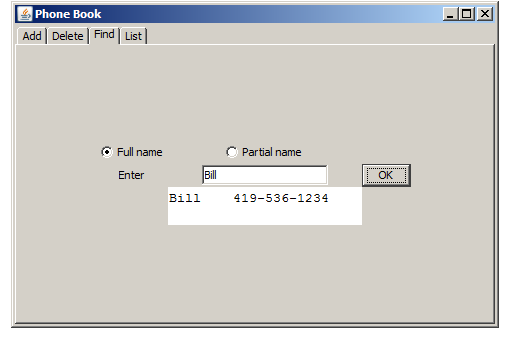


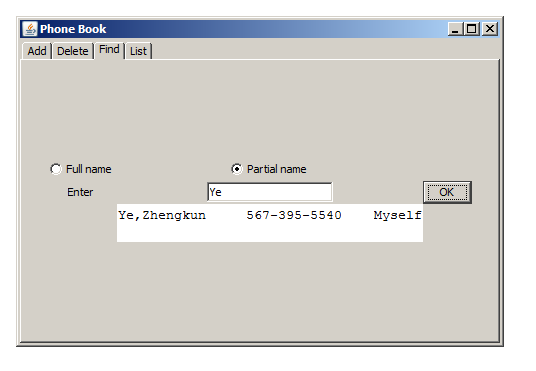


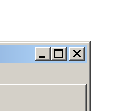










 The cross is just my “ Quit” .

package GUIPhoneBook;

import java.io.File;

import java.io.FileNotFoundException;

import java.io.FileOutputStream;

import java.io.IOException;

import java.io.InputStreamReader;

import java.io.PrintStream;

import java.util.Iterator;

import java.util.Scanner;

import java.util.Vector;

import java.util.regex.Matcher;

import java.util.regex.Pattern;

import javax.swing.JOptionPane;

public class PhoneBook {

private static Vector<User> userlist = new Vector<User>();

public PhoneBook()

{}

public void init(String FileName) throws Exception

{

File F;

F = new File(FileName);

Scanner S = new Scanner(F);

if(S == null ) return;

while (S.hasNextLine())

{

User temp = new User();

temp.setName(S.next());

temp.setPhoneNumber(S.next());

String notes = S.nextLine();

if(notes.charAt(0) == '\t') //delete the '\t' character

notes = notes.substring(1);

temp.setNotes(notes);

userlist.add(temp);

}

S.close();

}

public boolean add(User src) throws IOException

{

int i;

Object[] options = {"Yes","No"};

for(i = 0; i < userlist.size(); i++)

{

if(src.getName().compareTo(userlist.get(i).getName()) == 0) //warning: same name

{

int result = JOptionPane.showOptionDialog(null,"Same name exists, still want to add?","Warning",JOptionPane.YES\_NO\_OPTION,JOptionPane.PLAIN\_MESSAGE,null,options,options[0]);

if(0 == result)

continue;

else

break;

}

else if(src.getName().compareTo(userlist.get(i).getName())< 0)

{

userlist.insertElementAt(src, i);

return true;

}

}

if(i == userlist.size()) //the new entry is located in the last index or the entry list is null

{

userlist.add(src);

return true;

}

return false;

}

public Vector<User> list()

{

// Iterator<User> it = userlist.iterator();

// while(it.hasNext())

// {

// User tmp = new User();

// tmp = it.next();

// System.out.println(tmp.getName());

// System.out.println(tmp.getPhoneNumber());

// System.out.println(tmp.getNotes()+ "\n");

// }

return userlist;

}

public Vector<User> find(String name)

{

Vector<User> temp = new Vector<User>();

Iterator<User> it = userlist.iterator();

while(it.hasNext())

{

User u = new User();

u = it.next();

if(u.getName().compareToIgnoreCase(name) == 0)

{

temp.add(u);

}

}

return temp;

}

public Vector<User> partialfind(String name)

{

Vector<User> temp = new Vector<User>();

Iterator<User> it = userlist.iterator();

while(it.hasNext())

{

User u = new User();

u = it.next();

if(u.getName().toLowerCase().indexOf(name.toLowerCase()) >= 0)

{

temp.add(u);

}

}

return temp;

}

public boolean delete(String name)

{

Iterator<User> it = userlist.iterator();

boolean flag = false;

while(it.hasNext())

{

if(it.next().getName().equals(name))

{

flag = true;

it.remove();

}

}

if(flag) return true;

else return false; //delete fails

}

public void save(String FileName) throws FileNotFoundException

{

FileOutputStream out = new FileOutputStream(FileName);

PrintStream P = new PrintStream( out );

for (int i=0; i < userlist.size() ; i++)

{

P.println(userlist.get(i).getName() + "\t" + userlist.get(i).getPhoneNumber() +

"\t" + userlist.get(i).getNotes());

}

try {

out.close();

} catch (IOException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

}

}

**package** GUIPhoneBook;

**public** **class** User{

**private** String name, phoneNumber, notes;

**public** User()

{

}

**public** User(String name, String phoneNumber, String notes)

{

**this**.name = name;

**this**.phoneNumber = phoneNumber;

**this**.notes = notes;

}

**public** **void** setName(String name)

{

**this**.name = name;

}

**public** **void** setPhoneNumber(String phoneNumber)

{

**this**.phoneNumber = phoneNumber;

}

**public** **void** setNotes(String notes)

{

**this**.notes = notes;

}

**public** String getName()

{

**return** **this**.name;

}

**public** String getPhoneNumber()

{

**return** **this**.phoneNumber;

}

**public** String getNotes()

{

**return** **this**.notes;

}

}

package GUIPhoneBook;

import javax.swing.\*;

import javax.swing.event.ChangeEvent;

import javax.swing.event.ChangeListener;

import java.awt.\*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.io.FileNotFoundException;

import java.io.IOException;

import java.util.Iterator;

import java.util.Vector;

import java.util.regex.Matcher;

import java.util.regex.Pattern;

public class UserLogin

{

static final int WIDTH = 480;

static final int HEIGHT = 300;

final static String FILEPATH = "info.txt";

public UserLogin()

{

try

{

UIManager.setLookAndFeel(UIManager.getSystemLookAndFeelClassName());

}

catch(Exception e){}

final PhoneBook pb = new PhoneBook();

try {

pb.init(FILEPATH);

} catch (Exception e2) {}

JFrame frame = new JFrame("Phone Book");

Toolkit kit = Toolkit.getDefaultToolkit();

Dimension screenSize = kit.getScreenSize();

int width = screenSize.width;

int height = screenSize.height;

int x = (width - WIDTH)/2;

int y = (height - HEIGHT)/2 - 50;

frame.setLocation(x,y);

frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

final JTabbedPane tp = new JTabbedPane();

frame.setContentPane(tp);

JPanel panel1 = new JPanel();

JPanel panel2= new JPanel();

JPanel panel3 = new JPanel();

JPanel panel4 = new JPanel();

tp.addTab("panel1", panel1);

tp.setEnabledAt(0,true);

tp.setTitleAt(0,"Add");

tp.addTab("panel2", panel2);

tp.setEnabledAt(1,true);

tp.setTitleAt(1,"Delete");

tp.addTab("panel3", panel3);

tp.setEnabledAt(2,true);

tp.setTitleAt(2,"Find");

tp.addTab("panel4", panel4);

tp.setEnabledAt(3,true);

tp.setTitleAt(3, "List");

tp.setPreferredSize(new Dimension(480,300));

tp.setTabPlacement(JTabbedPane.TOP);

tp.setTabLayoutPolicy(JTabbedPane.SCROLL\_TAB\_LAYOUT);

//frame.pack();

panel1.setLayout(new GridBagLayout());

panel1.setOpaque(true);

panel2.setLayout(new GridBagLayout());

panel2.setOpaque(true);

panel3.setLayout(new GridBagLayout());

panel3.setOpaque(true);

panel4.setLayout(new GridBagLayout());

panel4.setOpaque(true);

final JLabel l1 = new JLabel("Name");

final JLabel l2 = new JLabel("PhoneNumber");

final JLabel l3 = new JLabel("Notes");

final JTextField t1 = new JTextField(23);

final JTextField t2 = new JTextField(23);

final JTextField t3 = new JTextField(23);

JButton b1 = new JButton("Submmmit");

JButton b2 = new JButton("Reset");

GridBagConstraints c = new GridBagConstraints();

// c.fill = GridBagConstraints.NONE;

// c.anchor = GridBagConstraints.CENTER;

// c.weightx = 1;

// c.weighty = 1;

//add(t1,c,0,0,1,1);

c.gridx = 0;

c.gridy = 0;

panel1.add(l1,c);

c.gridwidth = 2;

c.gridheight = 1;

c.gridx = 1;

panel1.add(t1,c);

c.gridx = 0;

c.gridy = 1;

c.gridwidth = 1;

c.gridheight = 1;

panel1.add(l2,c);

c.gridx = 1;

c.gridwidth = 2;

c.gridheight = 1;

panel1.add(t2,c);

c.gridx = 0;

c.gridy = 2;

c.gridwidth = 1;

c.gridheight = 1;

panel1.add(l3,c);

c.gridx = 1;

c.gridwidth = 2;

c.gridheight = 1;

panel1.add(t3,c);

c.gridx = 1;

c.gridy = 3;

c.gridwidth = 1;

c.gridheight = 1;

panel1.add(b1,c);

c.gridx= 2;

c.gridwidth = 1;

c.gridheight = 1;

panel1.add(b2,c);

frame.setVisible(true);

frame.pack();

t1.grabFocus(); //t1 grab the focus

b1.addActionListener(new ActionListener(){

public void actionPerformed(ActionEvent e)

{

User newUser = new User();

if(t1.getText().trim().length() == 0)

{

JOptionPane.showMessageDialog(null,"Name cannot be empty.","Error",JOptionPane.ERROR\_MESSAGE);

return;

}

newUser.setName(t1.getText());

if(!validatePhoneNumber(t2.getText().trim()))

{

//System.out.println("illegal phone number");

JOptionPane.showMessageDialog(null,"illegal phone number","Error",JOptionPane.ERROR\_MESSAGE);

return;

}

newUser.setPhoneNumber(t2.getText());

newUser.setNotes(t3.getText());

pb.list();

boolean ac = false;

try {

ac = pb.add(newUser);

} catch (IOException e1) {

// TODO Auto-generated catch block

e1.printStackTrace();

}

try {

pb.save(FILEPATH);

} catch (FileNotFoundException e1) {

// TODO Auto-generated catch block

e1.printStackTrace();

}

pb.list();

if(ac)

{

JOptionPane.showMessageDialog(null,"Entry is added successfully","Notice",JOptionPane.INFORMATION\_MESSAGE|JOptionPane.DEFAULT\_OPTION);

t1.setText("");

t2.setText("");

t3.setText("");

}

}

});

b2.addActionListener(new ActionListener(){

public void actionPerformed(ActionEvent e)

{

t1.setText("");

t2.setText("");

t3.setText("");

}

});

//---------------Pane2-------------------------------//

final JLabel ll1 = new JLabel("Enter Name");

final JTextField tt1 = new JTextField(15);

final JButton bb1 = new JButton("OK");

panel2.add(ll1);

panel2.add(tt1);

panel2.add(bb1);

bb1.addActionListener(new ActionListener(){

public void actionPerformed(ActionEvent e)

{

boolean f = pb.delete(tt1.getText().trim());

if(!f)

JOptionPane.showMessageDialog(null,"No such name","Notice",JOptionPane.WARNING\_MESSAGE);

else

{

JOptionPane.showMessageDialog(null,"Entry is deleted successfully","Notice",JOptionPane.INFORMATION\_MESSAGE|JOptionPane.DEFAULT\_OPTION);

try {

pb.save(FILEPATH);

} catch (FileNotFoundException e1) {

// TODO Auto-generated catch block

e1.printStackTrace();

}

}

//tt1.setText("");

}

});

final JLabel ll3 = new JLabel("Enter");

final JTextField tf3 = new JTextField(15);

final JTextArea ja3= new JTextArea(2,2);

ja3.setVisible(false);

ja3.setEditable(false);

final JButton bb3 = new JButton("OK");

final JRadioButton r1 = new JRadioButton("Full name");

final JRadioButton r2 = new JRadioButton("Partial name");

ButtonGroup bg = new ButtonGroup();

r1.setSelected(true); //set the r1 selected

bg.add(r1);

bg.add(r2);

c.gridx = 0;

c.gridy = 0;

panel3.add(r1,c);

c.gridx = 1;

panel3.add(r2,c);

c.gridx = 0;

c.gridy = 1;

panel3.add(ll3,c);

c.gridx= 1;

panel3.add(tf3,c);

c.gridx = 2;

panel3.add(bb3,c);

c.gridx = 1;

c.gridy = 2;

panel3.add(ja3,c);

bb3.addActionListener(new ActionListener(){

public void actionPerformed(ActionEvent e)

{

ja3.setVisible(true);

Vector<User> result ;

if(r1.isSelected())

result = pb.find(tf3.getText().trim());

else

result = pb.partialfind(tf3.getText().trim());

if(0 == result.size())

ja3.setText("No such name");

else

{

Iterator<User> it = result.iterator();

ja3.setText("");

while(it.hasNext())

{

User ui = new User();

ui = it.next();

ja3.append(ui.getName()+"\t" + ui.getPhoneNumber() + "\t"

+ ui.getNotes() + "\n");

}

}

//tt1.setText("");

}

});

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*panel 4\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

final JLabel j4 = new JLabel("All Entries are listed below");

final JLabel j5 = new JLabel("----------------------------");

final JTextArea tf4 = new JTextArea(10,30);

// tf4.append("Name" + "\t" + "PhoneNumber" + "\t" + "Notes\n");

// tf4.append("~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~");

tf4.setEditable(false);

c.gridx = 0;

c.gridy = 0;

c.gridwidth = 1;

c.gridheight = 1;

panel4.add(j4,c);

c.gridy = 1;

panel4.add(j5,c);

c.gridy = 2;

panel4.add(tf4,c);

tp.addChangeListener(new ChangeListener(){

public void stateChanged(ChangeEvent e){

int selectedIndex = tp.getSelectedIndex();

// String title = tp.getTitleAt(selectedIndex);

if(3 == selectedIndex)

{

tf4.setText("~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~\n");

tf4.append("Name" + "\t" + "PhoneNumber" + "\t" + "Notes\n");

tf4.append("~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~");

Vector<User> lists = pb.list();

Iterator<User> it = lists.iterator();

while(it.hasNext())

{

User tmp = new User();

tmp = it.next();

tf4.append("\n");

tf4.append(tmp.getName()+"\t" + tmp.getPhoneNumber() + "\t"

+ tmp.getNotes());

}

}

}

});

}

public static boolean validatePhoneNumber(String phoneNumber)

{

String regex = "(^\\d{3}-\\d{3}-\\d{4}$)|(^\\(\\d{3}\\)\\d{3}-\\d{4}$)|(^\\d{3}-\\d{4}$)" ;

Pattern pat = Pattern.compile(regex);

Matcher ma = pat.matcher(phoneNumber);

boolean rs = ma.find();

return rs;

}

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

{

new UserLogin();

}

}

