|  |  |
| --- | --- |
| Course | Advanced Software Design – CS525 |
| Assignment | Lab 7 |
| Week | 07 |
| Due | 8/10/2025 |
| Student name | Toe Toe Aung |
| Student ID | 618090 |

1. **UML Class Diagram**

A computer screen shot of a computer

AI-generated content may be incorrect.

1. **Sequence diagram of Camera Application**

A diagram of a diagram

AI-generated content may be incorrect.

1. **Implementation of Camera Application Design in Java**

**package** CameraApp;

**public** **class** CameraRecord {

String licensePlate;

String cameraId;

**double** speed;

**public** CameraRecord(String cameraId, String licensePlate, **double** vehicleSpeed) {

**this**.licensePlate = licensePlate;

**this**.cameraId = cameraId;

**this**.speed = speed;

}

**public** **void** checkStatus() {

System.***out***.println(**this**.toString());

DetectionHandler handler = **new** UnregisterDetectionHandler(**this**);

handler.detect();

System.***out***.println();

}

@Override

**public** String toString() {

**return** "Camera=" + **this**.cameraId + "\tplate=" + **this**.licensePlate + "\tspeed=" + **this**.speed;

}

}

**package** CameraApp;

**public** **abstract** **class** DetectionHandler {

CameraRecord record;

**public** DetectionHandler(CameraRecord record) {

**this**.record = record;

}

**abstract** **void** detect();

}

**package** CameraApp;

**public** **class** SpeedingDetectionHandler **extends** DetectionHandler {

**public** SpeedingDetectionHandler(CameraRecord record) {

**super**(record);

}

@Override

**void** detect() {

**if** (**this**.record.speed >= 65) {

System.***out***.println("Speeding detected!");

} **else** {

System.***out***.println("Normal");

}

}

}

**package** CameraApp;

**public** **class** StolenDetectionHandler **extends** DetectionHandler {

**public** StolenDetectionHandler(CameraRecord record) {

**super**(record);

}

@Override

**public** **void** detect() {

**if** (record.licensePlate.contains("ABC1234")) {

System.***out***.println("Stolen detected!");

} **else** {

DetectionHandler nextHandle = **new** SpeedingDetectionHandler(record);

nextHandle.detect();

}

}

}

**package** CameraApp;

**import** java.util.Arrays;

**public** **class** UnpaidDetectionHandler **extends** DetectionHandler {

String[] unpaid = { "XYZ0987", "XZY5678" };

**public** UnpaidDetectionHandler(CameraRecord record) {

**super**(record);

}

@Override

**void** detect() {

**if** (Arrays.*asList*(unpaid).contains(**this**.record.licensePlate)) {

System.***out***.println("Unpaid ticket whose owner detected!");

} **else** {

DetectionHandler nextHandle = **new** StolenDetectionHandler(**this**.record);

nextHandle.detect();

}

}

}

**package** CameraApp;

**import** java.util.Arrays;

**public** **class** UnregisterDetectionHandler **extends** DetectionHandler {

String[] unregistered = { "XYZ1234", "ZYX4321" };

**public** UnregisterDetectionHandler(CameraRecord record) {

**super**(record);

}

@Override

**void** detect() {

**if** (Arrays.*asList*(unregistered).contains(**this**.record.licensePlate)) {

System.***out***.println("Unregistered detected!");

} **else** {

DetectionHandler nextHandle = **new** UnpaidDetectionHandler(**this**.record);

nextHandle.detect();

}

}

}

**d. Class Diagram of Counter App**

A diagram of a computer

AI-generated content may be incorrect.

**e. Sequence Diagram of the Counter App**

A screenshot of a computer program

AI-generated content may be incorrect.

**f. Java Program of the Counter App**

**package** Counter;

**import** java.awt.Dimension;

**import** java.awt.Toolkit;

**import** java.awt.Rectangle;

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

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

**import** javax.swing.JButton;

**import** javax.swing.JFrame;

**public** **class** JFrameCounter **extends** JFrame {

/\*\*

\* Generated UID

\*/

**private** **static** **final** **long** ***serialVersionUID*** = -8652729993765133587L;

**private** JButton jButtonIncrement = **new** JButton();

**private** JButton jButtondecrement = **new** JButton();

**private** JButton jButtonundo = **new** JButton();

**private** JButton jButtonredo = **new** JButton();

**private** Counter counter;

**private** History hist;

**public** JFrameCounter() {

**try** {

jbInit();

counter = **new** Counter();

TextFrame textframe = **new** TextFrame();

textframe.setVisible(**true**);

RectFrame rectframe = **new** RectFrame();

rectframe.setVisible(**true**);

OvalFrame ovalframe = **new** OvalFrame();

ovalframe.setVisible(**true**);

counter.registerObserver(textframe);

counter.registerObserver(rectframe);

counter.registerObserver(ovalframe);

hist = **new** History(counter);

} **catch** (Exception e) {

e.printStackTrace();

}

}

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

JFrameCounter frame = **new** JFrameCounter();

Dimension screenSize = Toolkit.*getDefaultToolkit*().getScreenSize();

Dimension frameSize = frame.getSize();

**if** (frameSize.height > screenSize.height) {

frameSize.height = screenSize.height;

}

**if** (frameSize.width > screenSize.width) {

frameSize.width = screenSize.width;

}

frame.setLocation((screenSize.width - frameSize.width) / 2, (screenSize.height - frameSize.height) / 2);

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

frame.setVisible(**true**);

}

**private** **void** jbInit() **throws** Exception {

**this**.getContentPane().setLayout(**null**);

**this**.setSize(**new** Dimension(297, 169));

jButtonIncrement.setText("+");

jButtonIncrement.setBounds(**new** Rectangle(30, 25, 73, 22));

jButtonIncrement.setActionCommand("increment");

jButtonIncrement.addActionListener(**new** ActionListener() {

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

jButtonIncrement\_actionPerformed(e);

}

});

jButtondecrement.setText("-");

jButtondecrement.setBounds(**new** Rectangle(155, 25, 73, 22));

jButtondecrement.setActionCommand("decrement");

jButtondecrement.addActionListener(**new** ActionListener() {

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

jButtondecrement\_actionPerformed(e);

}

});

jButtonundo.setText("undo");

jButtonundo.setBounds(**new** Rectangle(30, 80, 73, 22));

jButtonundo.addActionListener(**new** ActionListener() {

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

jButtonundo\_actionPerformed(e);

}

});

jButtonredo.setText("redo");

jButtonredo.setBounds(**new** Rectangle(155, 80, 73, 22));

jButtonredo.addActionListener(**new** ActionListener() {

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

jButtonredo\_actionPerformed(e);

}

});

**this**.getContentPane().add(jButtonredo, **null**);

**this**.getContentPane().add(jButtonundo, **null**);

**this**.getContentPane().add(jButtondecrement, **null**);

**this**.getContentPane().add(jButtonIncrement, **null**);

}

**private** **void** jButtonIncrement\_actionPerformed(ActionEvent e) {

ICounterCommand incCommand = **new** IncrementCommand(counter);

incCommand.execute();

hist.addCommand(incCommand);

}

**private** **void** jButtondecrement\_actionPerformed(ActionEvent e) {

ICounterCommand decCommand = **new** DecrementCommand(counter);

decCommand.execute();

hist.addCommand(decCommand);

}

**private** **void** jButtonundo\_actionPerformed(ActionEvent e) {

hist.undo();

}

**private** **void** jButtonredo\_actionPerformed(ActionEvent e) {

hist.redo();

}

}

A screenshot of a computer

AI-generated content may be incorrect.