

Data Validation Task

GR12 Assignment

Full name: Milaan Kassie



Table of contents

Table of contents	2
1. Project Summary	4
Fields	4
Student ID – String	4
Date of Absence – Date	4
Online Attendance – Boolean	4
Number of lessons attended – Integer	4
Reason – String	4
2. Interface Design	5
AbsenteeForm	5
AbsenteeForm Components	5
TicketFrame	7
TicketFrame Components	7
HelpFrame	8
HelpFrame Components	8
3. Fields tables	9
4. Test Plan	10
5. Evidence of Testing	11
Student ID – Length Check	11
Standard Data Test Cases: Accepted as valid	11
Extreme Data Test Cases: Rejected as invalid	11
Student ID – Format Check	12
Standard Data Test Cases: Accepted as valid	12
Extreme Data Test Cases: Rejected as invalid	12
Abnormal Data Test Cases: Rejected as invalid	12
Date of Absence – Date Range (component side validation)	13
Standard Data Test Cases: Accepted as valid	13
Component side validation - Minimum	13
Component side validation – Maximum	13
Online Attendance – Boolean value cant be validated	14
Standard Data Test Cases: Accepted as valid	14
Number of Lessons Attended – Data Type Check	15
Standard Data Test Cases: Accepted as valid	15
Abnormal Data Test Cases: Rejected as invalid	15
Number of Lessons Attended – Range Check	16
Standard Data Test Cases: Accepted as valid	16
Extreme Data Test Cases: Rejected as invalid	16
Abnormal Data Test Cases: Rejected as invalid	16
Reason – Length Check	17
Standard Data Test Cases: Accepted as valid	17
Extreme Data Test Cases: Accepted as valid	17

5. Programming code	18
DVTFrame code	18
TicketFrame code	32
HelpFrame code	36

1. Project Summary

The AbsenteeApp is a program designed to allow students to quickly and easily log their absences. The absence will be logged as a ticket which can either be printed out and handed in physically or sent digitally to the school, in which case all school staff will be notified immediately. The application allows students to easily and conveniently notify the school when they are unable to attend school. Absence can only be logged for the last week and cannot be made in advance.

Fields

Student ID – String

11 character long string value made of the prefix “SID” followed by 8 numbers

The primary key of the school student database. By including this in the absentee ticket the school can easily identify the student in their database and record the absence on their system.

Date of Absence – Date

The date of the absence in the format dd/MM/yy. This date cannot be more than a week ago. An absence longer than a week ago should be dealt with through and email or in person interaction with a school administrator or grade coordinator.

Online Attendance – Boolean

A yes/no value that indicates whether the student was able to attend some, or all of the lessons missed online.

Number of lessons attended – Integer

This value may only be entered(changed from max 8 lessons) if the student has indicated that they were able to attend online. A student can attend from 1 to 8 lessons online

Reason – String

The reason for the absence such as illness, family emergencies or other such reasons. Students should explain themselves briefly in detail so that the school is fully aware why they were unable to attend school

2. Interface Design

AbsenteeForm

The screenshot shows the 'Absentee Form' interface. It features a title bar with a 'Help' button (3). Below the title, a layered pane (4) contains the form content. The form includes labels for 'Student ID' (5), 'Date of absence' (6), 'Online attendance' (7), 'Number of lessons attended' (8), and 'Reason of absence:' (9). The 'Student ID' field (10) contains 'SID10002110' and has an 'entry valid' (20) message. The 'Date of absence' field (11) contains '05/08/2024' and has an 'entry valid' (21) message. The 'Online attendance' field (12) is a checkbox labeled 'present?' and has an 'entry valid' (22) message. The 'Number of lessons attended' field (13) contains '1' and has an 'entry valid' (23) message. The 'Reason of absence:' field (14) is a text area containing 'I was sick' and has an 'entry valid' (24) message. At the bottom, there is a 'Submit' button (25) and a 'Ticket sent successfully' message (26).

AbsenteeForm Components

NO	COMPONENT	INPUT	EVENT
0	Frame		
1	Layered Pane		
2	Label		
3	Button	Mouse	Brings up Help Screen
4	Layered Pane		
5	Label		
6	Label		
7	Label		
8	Label		
9	Label		
10	Text Field	Mouse / Keyboard	
11	Date Chooser	Mouse / Keyboard	
12	Checkbox	Mouse	
13	Text Field	Mouse / Keyboard	
14	Scroll Pane & Text Area		
15	Label		
16	Label		
17	Label		

18	Label		
19	Label		
20	Label		
21	Label		
22	Label		
23	Label		
24	Label		
25	Button		
26	Label		

TicketFrame

1 **2** Ticket

3 Absentee ticket:

Student ID: SID10002110
Date of Absence: 04/08/2024
Online Attendance: Yes
Lessons Attended: 8
Lessons Missed: 0

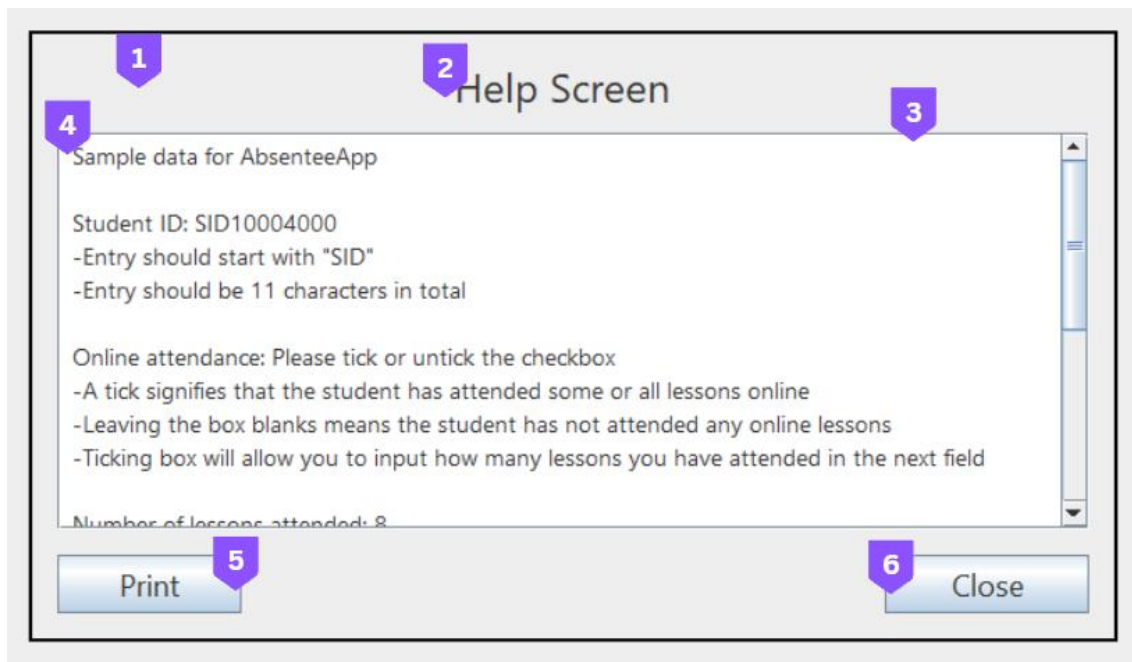
Reason for Absence:
I was unable to attend due to a grave illness

4 Send Ticket **5** Print Ticket

TicketFrame Components

NO	COMPONENT	INPUT	EVENT
0	Frame		
1	Layered Pane		
2	Label		
3	Scroll Pane & Text Area	Keyboard	
4	Button	Mouse	Brings up a dialog box to indicate that the ticket has been sent
5	Button	Mouse	Brings up print settings for printing the contents of the text area, the ticket

HelpFrame



HelpFrame Components

NO	COMPONENT	INPUT	EVENT
0	Frame		
1	Layered Pane		
2	Label		
3	Scroll Pane & Text Area	Keyboard	
4	Button	Mouse	Brings up print settings for printing the contents of the text area, the ticket
5	Button	Mouse	Closes the help screen

3. Fields tables

Field Name	Data Type	GUI component	Reason for GUI component choice	Validation Types	Reason for Validation Types
Student ID	String	Text Field	Precision: Ideal for capturing precise, short text entries Validation: Easy to validate and restrict input length	Format check Length check	Format check: first 3 letters should be "SID" for <u>Student ID</u> Length check: entry should be 11 characters long
Date of Absence	Date	Date Chooser	Accuracy: Ensures the user selects a valid date, reducing errors. Ease of use: Provides a calendar view, making date selection quicker and easier Formatting: Automatically handles date formatting, preventing user errors and reducing validation needs	N/A	
Online Attendance	Boolean	Checkbox	Simplicity: Clearly represents the yes/no option required. Easy to understand Ease of Use: Quick to toggle, making it user-friendly	N/A	
Number of Lessons Attended	Integer	Text field	Ease of use: Provides a straightforward way to input numbers quickly and easily Minimal interaction: no need to interact with component many times, just enter data using the keyboard	Data Type check Range check	Data Type check: number of lessons must be an integer Range check: number of lessons must be between 1 and 8 because there are only 8 lessons in a day
Reason	String	Text Area	Space: Allows for multiple lines of text, if users have a lengthy reason	Range check	Range check: entry should be a full detailed sentence between 1 and 50 characters

4. Test Plan

Field	Type of Check	Standard Data	Extreme Data	Abnormal Data
Student ID	Format Check	SID10003000	00010002110	N/A
Student ID	Length Check	SID10003000	ABC10002110	00010002110
Date of Absence	Date Range Check (component side check)	06/08/2024 (today's date)	N/A	N/A
Online Attendance	N/A	true/false	N/A	N/A
Number of Lessons Attended	Data Type Check	1	N/A	"one"
Number of Lessons Attended	Range Check	1	9	"nine"
Reason	Range Check	I had to attend a family emergency	I could not attend due to an unforeseen family emergency	N/A

5. Evidence of Testing

Student ID – Length Check

Standard Data Test Cases: Accepted as valid

Input: SID10003000 (11 characters)

Results: System accepts SID10003000 and proceeds without error.

Student ID	<input type="text"/>	*entry required -
------------	----------------------	----------------------

Student ID	SID10003000	entry valid -
------------	-------------	------------------

Ticket

Absentee ticket:

Student ID:	SID10003000
Date of Absence:	05/08/2024
Online Attendance:	No
Lessons Attended:	0
Lessons Missed:	8

Reason for Absence:
I was sick

Send TicketPrint Ticket

Extreme Data Test Cases: Rejected as invalid

Input: SID100030000 (12 characters)

Results: System accepts SID10003000 and proceeds without error.

Student ID	<input type="text"/>	*entry required -
------------	----------------------	----------------------

Student ID	SID100030001	*entry required Entry must be 11 characters in length
------------	--------------	--

Input: SID1000300 (10 characters)

Results: System accepts SID1000300 and proceeds without error.

Student ID	<input type="text"/>	*entry required -
------------	----------------------	----------------------

Student ID	SID1000300	*entry required Entry must be 11 characters in length
------------	------------	--

Student ID – Format Check

Standard Data Test Cases: Accepted as valid

Input: SID10002110

Results: System accepts SID10002110 and proceeds without error.

Student ID	<input type="text"/>	*entry required -
------------	----------------------	----------------------

Student ID	SID10002110	entry valid -
------------	-------------	------------------

Ticket

Absentee ticket:

Student ID:	SID10002110
Date of Absence:	05/08/2024
Online Attendance:	No
Lessons Attended:	0
Lessons Missed:	8

Reason for Absence:
I was sick

Send TicketPrint Ticket

Extreme Data Test Cases: Rejected as invalid

Input: ABC10002110

Results: System displays error message saying that the entry must begin with the characters: "SID"

Student ID	<input type="text"/>	*entry required -
------------	----------------------	----------------------

Student ID	ABC10002110	*entry required Entry must begin with the characters: 'SID'
------------	-------------	--

Abnormal Data Test Cases: Rejected as invalid

Input: 00010002110

Results: System displays error message saying that the entry must begin with the characters: "SID"

Student ID	<input type="text"/>	*entry required -
------------	----------------------	----------------------

Student ID	00010002110	*entry required Entry must begin with the characters: 'SID'
------------	-------------	--

Date of Absence – Date Range (component side validation)

Standard Data Test Cases: Accepted as valid

Expected Input: Any date within the last week

Component side validation prevents dates longer than a week ago from the current day from being chosen.

Input: 06/08/2024 (today's date)

Results: System accepts 06/08/2024 and proceeds without error.

Date of absence: 06/08/2024 entry valid

Online attendance: entry valid

Number of: *entry required

Reason of: *entry required

Date of absence: 06/08/2024 entry valid

Online attendance: entry valid

Number of: *entry required

Reason of: *entry required

Component side validation - Minimum

`MinSelectableDate = new Date(new Date().getTime() - 7L * 24 * 60 * 60 * 1000)`

```
jdcDateOfAbsence.setMinSelectableDate(new Date(new Date().getTime() - 7L * 24 * 60 * 60 * 1000));
```

Component side validation – Maximum

`MaxSelectableDate = new Date()`

```
jdcDateOfAbsence.setMaxSelectableDate(new Date());
```

Online Attendance – Boolean value cant be validated

Standard Data Test Cases: Accepted as valid

Input: true/false

Results: System accepts false, deactivates Number of Lessons Attended, and proceeds without error.

OR

Results: System accepts true, activates Number of Lessons Attended, and proceeds without error

Online attendance	<input type="checkbox"/> present?	entry valid -
Number of lessons attended	<input type="text" value="0"/>	entry valid -

Online attendance	<input checked="" type="checkbox"/> present?	entry valid -
Number of lessons attended	<input type="text" value="0"/>	entry valid -

Number of Lessons Attended – Data Type Check

Standard Data Test Cases: Accepted as valid

Input: 1

Results: System accepts 1 proceeds without error.

Number of lessons attended	<input type="text"/>	*entry required -
----------------------------	----------------------	----------------------

Number of lessons attended	<input type="text" value="1"/>	entry valid -
----------------------------	--------------------------------	------------------

Ticket

Absentee ticket:

Student ID:	SID10002110
Date of Absence:	05/08/2024
Online Attendance:	Yes
Lessons Attended:	1
Lessons Missed:	7

Reason for Absence:
I was sick

Send TicketPrint Ticket

Abnormal Data Test Cases: Rejected as invalid

Input: “one”

Results: System displays error message saying that the entry must be an integer value

Number of lessons attended	<input type="text"/>	*entry required -
----------------------------	----------------------	----------------------

Number of lessons attended	<input type="text" value="one"/>	*entry required Entry must be an integer value
----------------------------	----------------------------------	---

Number of Lessons Attended – Range Check

Standard Data Test Cases: Accepted as valid

Input: 1

Results: System accepts 1 proceeds without error.

Number of lessons attended	<input type="text"/>	*entry required -
----------------------------	----------------------	----------------------

Number of lessons attended	<input type="text" value="1"/>	entry valid -
----------------------------	--------------------------------	------------------

Ticket

Absentee ticket:

Student ID:	SID10002110
Date of Absence:	05/08/2024
Online Attendance:	Yes
Lessons Attended:	1
Lessons Missed:	7

Reason for Absence:
I was sick

Send TicketPrint Ticket

Extreme Data Test Cases: Rejected as invalid

Input: 9

Results: System displays an error message saying that the entry must be within the range 1 - 8

Number of lessons attended	<input type="text"/>	*entry required -
----------------------------	----------------------	----------------------

Number of lessons attended	<input type="text" value="9"/>	*entry required Entry must fall between the range: 1 and 8
----------------------------	--------------------------------	---

Abnormal Data Test Cases: Rejected as invalid

Input: "nine"

Results: System displays error message saying that the entry must be a integer value

Number of lessons attended	<input type="text"/>	*entry required -
----------------------------	----------------------	----------------------

Number of lessons attended	<input type="text" value="nine"/>	*entry required Entry must be an integer value
----------------------------	-----------------------------------	---

Reason – Length Check

Standard Data Test Cases: Accepted as valid

Input: "I had to attend a family emergency" (34 characters)

Results: System accepts "I had to attend a family emergency" and proceeds without error.

Reason of absence:

*entry required
-

Reason of absence:

I had to attend a family emergency

entry valid
-

Ticket

Absentee ticket:

Student ID:

SID10002110

Date of Absence:

05/08/2024

Online Attendance:

No

Lessons Attended:

0

Lessons Missed:

8

Reason for Absence:

I had to attend a family emergency

Send Ticket

Print Ticket

Extreme Data Test Cases: Accepted as valid

Input: "I could not attend due to an unforeseen family emergency" (56 characters)

Results: System accepts "I could not attend due to an unforeseen family emergency." and proceeds without error.

Reason of absence:

*entry required
-

Reason of absence:

I could not attend due to an unforeseen
family emergency

*entry required

Entry must be 1 - 50 characters in length

5. Programming code

DVTFrame code

```
package guiPkg;

import com.toedter.calendar.JTextFieldDateEditor;
import java.awt.Color;
import java.text.SimpleDateFormat;
import java.util.Date;
import javax.swing.JOptionPane;
import javax.swing.JSpinner;
import javax.swing.JSpinner.DefaultEditor;
import javax.swing.JTextField;
import javax.swing.SpinnerNumberModel;
import dataPkg.DVT_Validation;
import java.time.LocalDate;

public class DVTFrame extends javax.swing.JFrame
{

    //private int numLessonsMissed;
    //private Date dateOfAbsence;
    public DVTFrame()
    {
        initComponents();

        txfStudentID.requestFocus();

        //sets the default value to today's date.
        jdcDateOfAbsence.setDate(new Date());

        //sets the max to todays date
        jdcDateOfAbsence.setMaxSelectableDate(new Date());

        //now you cant type in to date chooser
        JTextFieldDateEditor editor = (JTextFieldDateEditor) jdcDateOfAbsence.getDateEditor();
        editor.setEditable(false);

        //makes it so that the spinner value can only be changed with arrows, no typing into it
        //spnNumLessonsAttended.setEditor(new JSpinner.DefaultEditor(spnNumLessonsAttended));
        txfNumLessonsAttended.setText("0");
    }

    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">
```

```

private void initComponents()
{

    lpBackground = new javax.swing.JLayeredPane();
    lblTitle = new javax.swing.JLabel();
    lpDetails = new javax.swing.JLayeredPane();
    lblStudentIDCheck = new javax.swing.JLabel();
    jdcDateOfAbsence = new com.toedter.calendar.JDateChooser();
    lblNumLessonsAttended = new javax.swing.JLabel();
    lblStudentID = new javax.swing.JLabel();
    lblOnlineAttendance = new javax.swing.JLabel();
    txfStudentID = new javax.swing.JTextField();
    lblDateOfAbsence = new javax.swing.JLabel();
    cbOnlineAttendance = new javax.swing.JCheckBox();
    lblNumLessonsAttendedCheck = new javax.swing.JLabel();
    lblDateOfAbsenceCheck = new javax.swing.JLabel();
    lblOnlineAttendance1 = new javax.swing.JLabel();
    spReason = new javax.swing.JScrollPane();
    txaReason = new javax.swing.JTextArea();
    lblReasonCheck = new javax.swing.JLabel();
    lblStudentIDErrorMessage = new javax.swing.JLabel();
    lblDateOfAbsenceErrorMessage = new javax.swing.JLabel();
    lblDateOfAbsenceCheck1 = new javax.swing.JLabel();
    lblOnlineAttendanceErrorMessage = new javax.swing.JLabel();
    lblReasonErrorMessage = new javax.swing.JLabel();
    lblNumberOfLessonsAttendedErrorMessage = new javax.swing.JLabel();
    txfNumLessonsAttended = new javax.swing.JTextField();
    btnHelp = new javax.swing.JButton();
    lblSubmissionCheck = new javax.swing.JLabel();
    btnSubmit = new javax.swing.JButton();

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
    setResizable(false);

    lpBackground.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

    lblTitle.setFont(new java.awt.Font("Segoe UI", 0, 38)); // NOI18N
    lblTitle.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
    lblTitle.setText("Absentee Form");

    lpDetails.setBorder(javax.swing.BorderFactory.createTitledBorder(javax.swing.BorderFactory.createLineBo
rder(new java.awt.Color(0, 0, 0)), "Please enter attendance details",
javax.swing.border.TitledBorder.DEFAULT_JUSTIFICATION,
javax.swing.border.TitledBorder.DEFAULT_POSITION, new java.awt.Font("Segoe UI", 0, 24))); // NOI18N

    lblStudentIDCheck.setFont(new java.awt.Font("Segoe UI", 0, 14)); // NOI18N

```

```

lblStudentIDCheck.setForeground(new java.awt.Color(255, 0, 0));
lblStudentIDCheck.setText("*entry required");

jdcDateOfAbsence.setDateFormatString("dd/MM/yyyy");
jdcDateOfAbsence.setFont(new java.awt.Font("Segoe UI", 0, 20)); // NOI18N
jdcDateOfAbsence.setMaxSelectableDate(new Date());
jdcDateOfAbsence.setMinSelectableDate(new Date(new Date().getTime() - 7L * 24 * 60 * 60 * 1000));

lblNumLessonsAttended.setFont(new java.awt.Font("Segoe UI", 0, 20)); // NOI18N
lblNumLessonsAttended.setText("Number of lessons attended");

lblStudentID.setFont(new java.awt.Font("Segoe UI", 0, 20)); // NOI18N
lblStudentID.setText("Student ID");
lblStudentID.setMaximumSize(new java.awt.Dimension(93, 26));
lblStudentID.setMinimumSize(new java.awt.Dimension(93, 26));
lblStudentID.setPreferredSize(new java.awt.Dimension(93, 26));

lblOnlineAttendance.setFont(new java.awt.Font("Segoe UI", 0, 20)); // NOI18N
lblOnlineAttendance.setText("Online attendance");

txfStudentID.setFont(new java.awt.Font("Segoe UI", 0, 20)); // NOI18N
txfStudentID.setPreferredSize(new java.awt.Dimension(68, 35));

lblDateOfAbsence.setFont(new java.awt.Font("Segoe UI", 0, 20)); // NOI18N
lblDateOfAbsence.setText("Date of absence");

cbOnlineAttendance.setFont(new java.awt.Font("Segoe UI", 0, 20)); // NOI18N
cbOnlineAttendance.setText("present?");
cbOnlineAttendance.setPreferredSize(new java.awt.Dimension(99, 35));
cbOnlineAttendance.addActionListener(new java.awt.event.ActionListener()
{
    public void actionPerformed(java.awt.event.ActionEvent evt)
    {
        cbOnlineAttendanceActionPerformed(evt);
    }
});

lblNumLessonsAttendedCheck.setFont(new java.awt.Font("Segoe UI", 0, 14)); // NOI18N
lblNumLessonsAttendedCheck.setForeground(new java.awt.Color(255, 0, 0));
lblNumLessonsAttendedCheck.setText("*entry required");

lblDateOfAbsenceCheck.setFont(new java.awt.Font("Segoe UI", 0, 14)); // NOI18N
lblDateOfAbsenceCheck.setForeground(new java.awt.Color(60, 179, 113));
lblDateOfAbsenceCheck.setText("entry valid");

lblOnlineAttendance1.setFont(new java.awt.Font("Segoe UI", 0, 20)); // NOI18N
lblOnlineAttendance1.setText("Reason of absence:");

```

```

txaReason.setColumns(20);
txaReason.setFont(new java.awt.Font("Courier New", 0, 18)); // NOI18N
txaReason.setLineWrap(true);
txaReason.setRows(5);
txaReason.setMargin(new java.awt.Insets(2, 8, 2, 6));
spReason.setViewportView(txaReason);

lblReasonCheck.setFont(new java.awt.Font("Segoe UI", 0, 14)); // NOI18N
lblReasonCheck.setForeground(new java.awt.Color(255, 0, 0));
lblReasonCheck.setText("*entry required");

lblStudentIDErrorMessage.setFont(new java.awt.Font("Segoe UI", 0, 14)); // NOI18N
lblStudentIDErrorMessage.setForeground(new java.awt.Color(255, 0, 0));
lblStudentIDErrorMessage.setText("-");

lblDateOfAbsenceErrorMessage.setFont(new java.awt.Font("Segoe UI", 0, 14)); // NOI18N
lblDateOfAbsenceErrorMessage.setForeground(new java.awt.Color(60, 179, 113));
lblDateOfAbsenceErrorMessage.setText("-");

lblDateOfAbsenceCheck1.setFont(new java.awt.Font("Segoe UI", 0, 14)); // NOI18N
lblDateOfAbsenceCheck1.setForeground(new java.awt.Color(60, 179, 113));
lblDateOfAbsenceCheck1.setText("entry valid");

lblOnlineAttendanceErrorMessage.setFont(new java.awt.Font("Segoe UI", 0, 14)); // NOI18N
lblOnlineAttendanceErrorMessage.setForeground(new java.awt.Color(60, 179, 113));
lblOnlineAttendanceErrorMessage.setText("-");

lblReasonErrorMessage.setFont(new java.awt.Font("Segoe UI", 0, 14)); // NOI18N
lblReasonErrorMessage.setForeground(new java.awt.Color(255, 0, 0));
lblReasonErrorMessage.setText("-");

lblNumberOfLessonsAttendedErrorMessage.setFont(new java.awt.Font("Segoe UI", 0, 14)); // NOI18N
lblNumberOfLessonsAttendedErrorMessage.setForeground(new java.awt.Color(255, 0, 0));
lblNumberOfLessonsAttendedErrorMessage.setText("-");

txfNumLessonsAttended.setFont(new java.awt.Font("Segoe UI", 0, 20)); // NOI18N
txfNumLessonsAttended.setEnabled(false);
txfNumLessonsAttended.setPreferredSize(new java.awt.Dimension(68, 35));

lpDetails.setLayer(lblStudentIDCheck, javax.swing.JLayeredPane.DEFAULT_LAYER);
lpDetails.setLayer(jdcDateOfAbsence, javax.swing.JLayeredPane.DEFAULT_LAYER);
lpDetails.setLayer(lblNumLessonsAttended, javax.swing.JLayeredPane.DEFAULT_LAYER);
lpDetails.setLayer(lblStudentID, javax.swing.JLayeredPane.DEFAULT_LAYER);
lpDetails.setLayer(lblOnlineAttendance, javax.swing.JLayeredPane.DEFAULT_LAYER);
lpDetails.setLayer(txfStudentID, javax.swing.JLayeredPane.DEFAULT_LAYER);
lpDetails.setLayer(lblDateOfAbsence, javax.swing.JLayeredPane.DEFAULT_LAYER);

```



```

        .addComponent(lblStudentIDErrorMessage, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)))
        .addGroup(lpDetailsLayout.createSequentialGroup())

        .addGroup(lpDetailsLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING, false)
            .addComponent(txfNumLessonsAttended, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addComponent(cbOnlineAttendance, javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addComponent(jdcDateOfAbsence, javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.DEFAULT_SIZE, 162, Short.MAX_VALUE))
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

        .addGroup(lpDetailsLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(lblDateOfAbsenceCheck, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addComponent(lblDateOfAbsenceCheck1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addComponent(lblOnlineAttendanceErrorMessage,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addGroup(lpDetailsLayout.createSequentialGroup())

            .addGroup(lpDetailsLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(lblNumLessonsAttendedCheck,
javax.swing.GroupLayout.PREFERRED_SIZE, 110, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(lblNumberOfLessonsAttendedErrorMessage,
javax.swing.GroupLayout.PREFERRED_SIZE, 271, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(lblDateOfAbsenceErrorMessage,
javax.swing.GroupLayout.PREFERRED_SIZE, 268, javax.swing.GroupLayout.PREFERRED_SIZE))
                .addGap(0, 30, Short.MAX_VALUE))))))
        .addGroup(lpDetailsLayout.createSequentialGroup())
            .addComponent(lblOnlineAttendance1, javax.swing.GroupLayout.PREFERRED_SIZE, 203,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(0, 0, Short.MAX_VALUE))
        .addGroup(lpDetailsLayout.createSequentialGroup())
            .addComponent(spReason, javax.swing.GroupLayout.PREFERRED_SIZE, 445,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

        .addGroup(lpDetailsLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(lblReasonCheck, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addComponent(lblReasonErrorMessage, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)))
        .addContainerGap())
    );
    lpDetailsLayout.setVerticalGroup(

```

```

lpDetailsLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
.addGroup(lpDetailsLayout.createSequentialGroup()
.addGap(18, 18, 18)
.addGroup(lpDetailsLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING,
false)
.addComponent(lblStudentID, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
.addComponent(txfStudentID, javax.swing.GroupLayout.PREFERRED_SIZE, 35,
javax.swing.GroupLayout.PREFERRED_SIZE)
.addGroup(lpDetailsLayout.createSequentialGroup()
.addComponent(lblStudentIDCheck)
.addGap(0, 0, 0)
.addComponent(lblStudentIDErrorMessage)))
.addGap(25, 25, 25)
.addGroup(lpDetailsLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
.addGroup(lpDetailsLayout.createSequentialGroup()
.addComponent(lblDateOfAbsenceCheck)
.addGap(0, 0, 0)
.addComponent(lblDateOfAbsenceErrorMessage))
.addGroup(lpDetailsLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING,
false)
.addComponent(lblDateOfAbsence, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
.addComponent(jdcDateOfAbsence, javax.swing.GroupLayout.PREFERRED_SIZE, 35,
javax.swing.GroupLayout.PREFERRED_SIZE)))
.addGap(25, 25, 25)
.addGroup(lpDetailsLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
.addGroup(lpDetailsLayout.createSequentialGroup()

.addGroup(lpDetailsLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
.addComponent(cbOnlineAttendance, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
.addComponent(lblOnlineAttendance, javax.swing.GroupLayout.PREFERRED_SIZE, 35,
javax.swing.GroupLayout.PREFERRED_SIZE))
.addGap(0, 0, Short.MAX_VALUE))
.addGroup(lpDetailsLayout.createSequentialGroup()
.addGap(9, 9, 9)
.addComponent(lblDateOfAbsenceCheck1)
.addGap(0, 0, Short.MAX_VALUE)
.addComponent(lblOnlineAttendanceErrorMessage)))
.addGroup(lpDetailsLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
.addGroup(lpDetailsLayout.createSequentialGroup()
.addGap(21, 21, 21)

.addGroup(lpDetailsLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
.addComponent(txfNumLessonsAttended, javax.swing.GroupLayout.PREFERRED_SIZE, 35,
javax.swing.GroupLayout.PREFERRED_SIZE)

```



```

        .addGroup(lpDetailsLayout.createSequentialGroup()
            .addComponent(lblNumLessonsAttendedCheck)
            .addGap(0, 0, 0)
            .addComponent(lblNumberOfLessonsAttendedErrorMessage)))
        .addGap(9, 9, 9))
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
lpDetailsLayout.createSequentialGroup()
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addComponent(lblNumLessonsAttended, javax.swing.GroupLayout.PREFERRED_SIZE, 35,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(18, 18, 18)))
        .addComponent(lblOnlineAttendance1, javax.swing.GroupLayout.PREFERRED_SIZE, 35,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGroup(lpDetailsLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(spReason, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGroup(lpDetailsLayout.createSequentialGroup()
                .addComponent(lblReasonCheck)
                .addGap(0, 0, 0)
                .addComponent(lblReasonErrorMessage)))
            .addContainerGap(16, Short.MAX_VALUE))
    );

    btnHelp.setFont(new java.awt.Font("Segoe UI", 0, 24)); // NOI18N
    btnHelp.setText("Help");
    btnHelp.addActionListener(new java.awt.event.ActionListener()
    {
        public void actionPerformed(java.awt.event.ActionEvent evt)
        {
            btnHelpActionPerformed(evt);
        }
    });

    lblSubmissionCheck.setFont(new java.awt.Font("Segoe UI", 0, 16)); // NOI18N
    lblSubmissionCheck.setForeground(new java.awt.Color(255, 0, 0));
    lblSubmissionCheck.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
    lblSubmissionCheck.setText("Please fill in all required fields");

    btnSubmit.setFont(new java.awt.Font("Segoe UI", 0, 24)); // NOI18N
    btnSubmit.setText("Submit");
    btnSubmit.addActionListener(new java.awt.event.ActionListener()
    {
        public void actionPerformed(java.awt.event.ActionEvent evt)
        {
            btnSubmitActionPerformed(evt);
        }
    });

```

```

lpBackground.setLayer(lblTitle, javax.swing.JLayeredPane.DEFAULT_LAYER);
lpBackground.setLayer(lpDetails, javax.swing.JLayeredPane.DEFAULT_LAYER);
lpBackground.setLayer(btnHelp, javax.swing.JLayeredPane.DEFAULT_LAYER);
lpBackground.setLayer(lblSubmissionCheck, javax.swing.JLayeredPane.DEFAULT_LAYER);
lpBackground.setLayer(btnSubmit, javax.swing.JLayeredPane.DEFAULT_LAYER);

javax.swing.GroupLayout lpBackgroundLayout = new javax.swing.GroupLayout(lpBackground);
lpBackground.setLayout(lpBackgroundLayout);
lpBackgroundLayout.setHorizontalGroup(
    lpBackgroundLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(lpBackgroundLayout.createSequentialGroup()
            .addGap(15, 15, 15)

.addGroup(lpBackgroundLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
    .addComponent(lpDetails, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
    .addGroup(lpBackgroundLayout.createSequentialGroup()

.addGroup(lpBackgroundLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(lpBackgroundLayout.createSequentialGroup()
        .addComponent(btnSubmit, javax.swing.GroupLayout.PREFERRED_SIZE, 160,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(83, 83, 83)
        .addComponent(lblSubmissionCheck, javax.swing.GroupLayout.PREFERRED_SIZE, 248,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
    .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
lpBackgroundLayout.createSequentialGroup()
        .addGap(0, 0, Short.MAX_VALUE)
        .addComponent(lblTitle, javax.swing.GroupLayout.PREFERRED_SIZE, 363,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(118, 118, 118)))
    .addComponent(btnHelp)))
    .addGap(19, 19, 19))
);
lpBackgroundLayout.setVerticalGroup(
    lpBackgroundLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(lpBackgroundLayout.createSequentialGroup()
            .addGap(15, 15, 15)
            .addComponent(lblTitle)
            .addComponent(btnHelp)
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 7, Short.MAX_VALUE)

```

```

        .addComponent(lpDetails, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(lpBackgroundLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(btnSubmit)
        .addComponent(lblSubmissionCheck))
        .addGap(15, 15, 15))
);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, layout.createSequentialGroup()
            .addContainerGap(15, Short.MAX_VALUE)
            .addComponent(lpBackground, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(15, 15, 15))
        );
layout.setVerticalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addGap(15, 15, 15)
            .addComponent(lpBackground, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
            .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
        );

pack();
setLocationRelativeTo(null);
} // </editor-fold>

```

```

private void btnHelpActionPerformed(java.awt.event.ActionEvent evt)
{
    new guiPkg.HelpFrame("Help.txt").setVisible(true);
}

```

```

private void btnSubmitActionPerformed(java.awt.event.ActionEvent evt)
{

```

```

    //StudentID
    String studentID = txfStudentID.getText().trim();

    //DateOfAbsence as a Date value
    Date dateOfAbsence = jdcDateOfAbsence.getDate();
    //Date formatting line

```

```

SimpleDateFormat df = new SimpleDateFormat("dd/MM/yyyy");
//DateOfAbsence as a String for the ticket
String dateOfAbsenceString = df.format(dateOfAbsence);

//OnlineAttendance as a boolean value
boolean onlineAttendance = false;
if (cbOnlineAttendance.isSelected())
{
    onlineAttendance = true;
}

//OnlineAttendance as a String value
String onlineAttendanceString = "No";
if (onlineAttendance)
{
    onlineAttendanceString = "Yes";
}
String numLessonsAttended = txfNumLessonsAttended.getText().trim();

//Reason as a String
String reason = txaReason.getText().trim();

//will be used to check variable validity in the final if
boolean isStudentIDValid = false;
boolean isNumLessonsAttendedValid = false;
boolean isReasonValid = false;

//Validation for Reason
if (DVT_Validation.vStringRange(reason, 1, 50, "Reason of Absence", lblReasonErrorMessage))
{
    //lblReasonCheck.setForeground(Color.GREEN);
    lblReasonCheck.setForeground(new Color(60, 179, 113));
    lblReasonCheck.setText("entry valid");
    isReasonValid = true;
} else
{
    txaReason.setText("");
    txaReason.requestFocus();

    lblReasonCheck.setForeground(Color.RED);
    lblReasonCheck.setText("*entry required");
}

//Validation for NumLessonsAttended - Its logical. I'm not removing this, it only makes sense
//DVT_Validation.vInteger(numLessonsAttended, "Number of Lessons Attended",
lblNumberOfLessonsAttendedErrorMessage)

```

```

if (cbOnlineAttendance.isSelected())
{
    if (DVT_Validation.vInteger(numLessonsAttended, "Number of Lessons Attended",
lblNumberOfLessonsAttendedErrorMessage) && DVT_Validation.vIntegerRange(numLessonsAttended, 1,
8, "Number of Lessons Attended", lblNumberOfLessonsAttendedErrorMessage))
    {
        lblNumLessonsAttendedCheck.setForeground(new Color(60, 179, 113));
        lblNumLessonsAttendedCheck.setText("entry valid");
        isNumLessonsAttendedValid = true;
    } else
    {
        txfNumLessonsAttended.setText("");
        txfNumLessonsAttended.requestFocus();

        lblNumLessonsAttendedCheck.setForeground(Color.RED);
        lblNumLessonsAttendedCheck.setText("*entry required");
    }
} else
{
    lblNumLessonsAttendedCheck.setForeground(new Color(60, 179, 113));
    lblNumLessonsAttendedCheck.setText("entry valid");
    isNumLessonsAttendedValid = true;
}

//Validation of StudentID - DONE
//if (DVT_Validation.vStringPresenceCheck(studentID, "StudentID", lblStudentIDErrorMessage) &&
DVT_Validation.vInitialsCheck(studentID, 3, "SID", "Student ID", lblStudentIDErrorMessage) &&
DVT_Validation.vStringLengthCheck(studentID, 11, "StudentID", lblStudentIDErrorMessage))
    if (DVT_Validation.vStringLengthCheck(studentID, 11, "StudentID", lblStudentIDErrorMessage) &&
DVT_Validation.vInitialsCheck(studentID, 3, "SID", "Student ID", lblStudentIDErrorMessage))
    {
        //lblStudentIDCheck.setForeground(Color.GREEN);
        lblStudentIDCheck.setForeground(new Color(60, 179, 113));
        lblStudentIDCheck.setText("entry valid");
        isStudentIDValid = true;
    } else
    {
        txfStudentID.setText("");
        txfStudentID.requestFocus();

        lblStudentIDCheck.setForeground(Color.RED);
        lblStudentIDCheck.setText("*entry required");
    }

//final check for all validations
if (isStudentIDValid && isNumLessonsAttendedValid && isReasonValid)

```

```

{
    lblSubmissionCheck.setForeground(new Color(60, 179, 113));
    lblSubmissionCheck.setText("Ticket sent successfully");

    int numLessonsMissed = 8 - Integer.parseInt(numLessonsAttended);
    Date dateOfAbsenceRaw = jdbcDateOfAbsence.getDate();

    String ticket = "Absentee ticket:\n\n";

    ticket += "Student ID:" + "\t\t" + studentID + "\n";
    ticket += "Date of Absence:" + "\t" + dateOfAbsenceString + "\n";
    ticket += "Online Attendance:" + "\t" + onlineAttendanceString + "\n";
    ticket += "Lessons Attended:" + "\t" + numLessonsAttended + "\n";
    ticket += "Lessons Missed:" + "\t" + numLessonsMissed + "\n";
    ticket += "\n" + "Reason for Absence: " + "\n" + reason;

    new guiPkg.TicketFrame(ticket).setVisible(true);

} else
{
    //at least one input is invalid, set appropriate message
    lblSubmissionCheck.setForeground(Color.RED);
    lblSubmissionCheck.setText("Please fill in all required fields");
    //btnTicket.setEnabled(false); //disable ticket button
}

}

private void cbOnlineAttendanceActionPerformed(java.awt.event.ActionEvent evt)
{
    if (cbOnlineAttendance.isSelected())
    {
        txNumLessonsAttended.setEnabled(true);

    } else
    {
        //lblStudentIDCheck.setForeground(new Color(60, 179, 113))
        txNumLessonsAttended.setEnabled(false);
        txNumLessonsAttended.setText("0");

        lblNumLessonsAttendedCheck.setForeground(new Color(60, 179, 113));
        lblNumLessonsAttendedCheck.setText("entry valid");

        lblNumberOfLessonsAttendedErrorMessage.setForeground(new Color(60, 179, 113));
        lblNumberOfLessonsAttendedErrorMessage.setText("-");
    }
}

```

```

}

public static void main(String args[])
{
    new DVTFrame().setVisible(true);
}

// Variables declaration - do not modify
private javax.swing.JButton btnHelp;
private javax.swing.JButton btnSubmit;
private javax.swing.JCheckBox cbOnlineAttendance;
private com.toedter.calendar.JDateChooser jdcDateOfAbsence;
private javax.swing.JLabel lblDateOfAbsence;
private javax.swing.JLabel lblDateOfAbsenceCheck;
private javax.swing.JLabel lblDateOfAbsenceCheck1;
private javax.swing.JLabel lblDateOfAbsenceErrorMessage;
private javax.swing.JLabel lblNumLessonsAttended;
private javax.swing.JLabel lblNumLessonsAttendedCheck;
private javax.swing.JLabel lblNumberOfLessonsAttendedErrorMessage;
private javax.swing.JLabel lblOnlineAttendance;
private javax.swing.JLabel lblOnlineAttendance1;
private javax.swing.JLabel lblOnlineAttendanceErrorMessage;
private javax.swing.JLabel lblReasonCheck;
private javax.swing.JLabel lblReasonErrorMessage;
private javax.swing.JLabel lblStudentID;
private javax.swing.JLabel lblStudentIDCheck;
private javax.swing.JLabel lblStudentIDErrorMessage;
private javax.swing.JLabel lblSubmissionCheck;
private javax.swing.JLabel lblTitle;
private javax.swing.JLayeredPane lpBackground;
private javax.swing.JLayeredPane lpDetails;
private javax.swing.JScrollPane spReason;
private javax.swing.JTextArea txaReason;
private javax.swing.JTextField txfNumLessonsAttended;
private javax.swing.JTextField txfStudentID;
// End of variables declaration
}

```

TicketFrame code

```
package guiPkg;

import java.awt.Desktop;
import java.io.File;
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.util.Scanner;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.swing.JOptionPane;

public class TicketFrame extends javax.swing.JFrame
{

    public TicketFrame(String ticket)
    {

        initComponents();

        try
        {
            txaDisplay.setText(ticket);

        } catch (Exception e)
        {
            System.out.println(e.toString());
        }

    }

    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">
    private void initComponents()
    {

        lpBackground = new javax.swing.JLayeredPane();
        spDisplay = new javax.swing.JScrollPane();
        txaDisplay = new javax.swing.JTextArea();
        btnPrint = new javax.swing.JButton();
        lblHelpScreen = new javax.swing.JLabel();
        btnSendTicket = new javax.swing.JButton();

        setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE_ON_CLOSE);
        setTitle("Ticket");
        setResizable(false);
```



```
lpBackground.setBackground(new java.awt.Color(0, 0, 0));
lpBackground.setBorder(javax.swing.BorderFactory.createEtchedBorder());
```

```
txaDisplay.setEditable(false);
txaDisplay.setColumns(20);
txaDisplay.setFont(new java.awt.Font("Segoe UI", 0, 14)); // NOI18N
txaDisplay.setRows(5);
txaDisplay.setMargin(new java.awt.Insets(2, 8, 2, 6));
spDisplay.setViewportView(txaDisplay);
```

```
btnPrint.setFont(new java.awt.Font("Segoe UI", 0, 24)); // NOI18N
btnPrint.setText("Print Ticket");
btnPrint.addActionListener(new java.awt.event.ActionListener()
{
    public void actionPerformed(java.awt.event.ActionEvent evt)
    {
        btnPrintActionPerformed(evt);
    }
});
```

```
lblHelpScreen.setFont(new java.awt.Font("Segoe UI", 0, 28)); // NOI18N
lblHelpScreen.setText("Ticket");
```

```
btnSendTicket.setFont(new java.awt.Font("Segoe UI", 0, 24)); // NOI18N
btnSendTicket.setText("Send Ticket");
btnSendTicket.addActionListener(new java.awt.event.ActionListener()
{
    public void actionPerformed(java.awt.event.ActionEvent evt)
    {
        btnSendTicketActionPerformed(evt);
    }
});
```

```
lpBackground.setLayer(spDisplay, javax.swing.JLayeredPane.DEFAULT_LAYER);
lpBackground.setLayer(btnPrint, javax.swing.JLayeredPane.DEFAULT_LAYER);
lpBackground.setLayer(lblHelpScreen, javax.swing.JLayeredPane.DEFAULT_LAYER);
lpBackground.setLayer(btnSendTicket, javax.swing.JLayeredPane.DEFAULT_LAYER);
```

```
javax.swing.GroupLayout lpBackgroundLayout = new javax.swing.GroupLayout(lpBackground);
lpBackground.setLayout(lpBackgroundLayout);
lpBackgroundLayout.setHorizontalGroup(
    lpBackgroundLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(lpBackgroundLayout.createSequentialGroup()
            .addGroup(lpBackgroundLayout.createParallelGroup()
                .addGap(15, 15, 15)
```

```
                .addGroup(lpBackgroundLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING, false)
```

```

        .addGroup(lpBackgroundLayout.createSequentialGroup())
        .addComponent(btnSendTicket)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(btnPrint))

    .addGroup(lpBackgroundLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(lpBackgroundLayout.createSequentialGroup())
        .addGap(131, 131, 131)
        .addComponent(lblHelpScreen))
        .addComponent(spDisplay, javax.swing.GroupLayout.PREFERRED_SIZE, 340,
javax.swing.GroupLayout.PREFERRED_SIZE)))
        .addContainerGap(15, Short.MAX_VALUE))
    );
    lpBackgroundLayout.setVerticalGroup(
        lpBackgroundLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(lpBackgroundLayout.createSequentialGroup())
        .addGap(9, 9, 9)
        .addComponent(lblHelpScreen)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
        .addComponent(spDisplay, javax.swing.GroupLayout.PREFERRED_SIZE, 315,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(15, 15, 15)

    .addGroup(lpBackgroundLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(btnPrint)
        .addComponent(btnSendTicket))
        .addContainerGap(15, Short.MAX_VALUE))
    );

    javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
    getContentPane().setLayout(layout);
    layout.setHorizontalGroup(
        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup())
        .addGap(15, 15, 15)
        .addComponent(lpBackground, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
        .addContainerGap(16, Short.MAX_VALUE))
    );
    layout.setVerticalGroup(
        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup())
        .addGap(15, 15, 15)
        .addComponent(lpBackground, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
        .addContainerGap(52, Short.MAX_VALUE))

```

```

);

setSize(new java.awt.Dimension(421, 526));
setLocationRelativeTo(null);
} // </editor-fold>

private void btnPrintActionPerformed(java.awt.event.ActionEvent evt)
{
    try
    {
        txaDisplay.print();

    } catch (Exception e)
    {
        JOptionPane.showMessageDialog(null, "Printing Failed", "Error", JOptionPane.ERROR_MESSAGE);
    }
}

private void btnSendTicketActionPerformed(java.awt.event.ActionEvent evt)
{
    JOptionPane.showMessageDialog(null, "All relevant subject staff will be notified. Thank you for your submission", "Ticket sent successfully", JOptionPane.DEFAULT_OPTION);
}

public static void main(String args[])
{
    new TicketFrame("").setVisible(true);
}

// Variables declaration - do not modify
private javax.swing.JButton btnPrint;
private javax.swing.JButton btnSendTicket;
private javax.swing.JLabel lblHelpScreen;
private javax.swing.JLayeredPane lpBackground;
private javax.swing.JScrollPane spDisplay;
private javax.swing.JTextArea txaDisplay;
// End of variables declaration
}

```

HelpFrame code

```
package guiPkg;

import java.awt.Desktop;
import java.io.File;
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.util.Scanner;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.swing.JOptionPane;

public class HelpFrame extends javax.swing.JFrame
{

    public HelpFrame(String fileName)
    {

        initComponents();

        try
        {
            //Scanner inputStream = new Scanner(new FileInputStream(filename));
            Scanner inputStream = new Scanner(new FileInputStream(fileName));

            while (inputStream.hasNextLine())
            {

                txaDisplay.append(inputStream.nextLine() + "\n");
            }

            inputStream.close();

        } catch (Exception e)
        {
            System.out.println(e.toString());
        }

    }

    // public Help(String filename)
    // {
    //     try
    //     {
    //         initComponents();
    //     }
    // }
```

```
//      Scanner inputStream = new Scanner(new FileInputStream(filename));
//
//      while (inputStream.hasNextLine())
//      {
//          txaDisplay.append(inputStream.nextLine() + "\n");
//      }
//
//      inputStream.close();
//  } catch (Exception e)
//  {
//      JOptionPane.showMessageDialog(null, "Text file not found", "Error", HEIGHT);
//  }
// }
```

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">

```
private void initComponents()
{
```

```
    lpBackground = new javax.swing.JLayeredPane();
    spDisplay = new javax.swing.JScrollPane();
    txaDisplay = new javax.swing.JTextArea();
    btnPrint = new javax.swing.JButton();
    btnClose = new javax.swing.JButton();
    lblHelpScreen = new javax.swing.JLabel();
```

```
    setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE_ON_CLOSE);
    setTitle("Help");
    setResizable(false);
```

```
    lpBackground.setBackground(new java.awt.Color(0, 0, 0));
    lpBackground.setBorder(javax.swing.BorderFactory.createEtchedBorder());
```

```
    txaDisplay.setEditable(false);
    txaDisplay.setColumns(20);
    txaDisplay.setFont(new java.awt.Font("Segoe UI", 0, 14)); // NOI18N
    txaDisplay.setRows(5);
    txaDisplay.setMargin(new java.awt.Insets(2, 8, 2, 6));
    spDisplay.setViewportView(txaDisplay);
```

```
    btnPrint.setFont(new java.awt.Font("Segoe UI", 0, 18)); // NOI18N
    btnPrint.setText("Print");
    btnPrint.addActionListener(new java.awt.event.ActionListener()
    {
        public void actionPerformed(java.awt.event.ActionEvent evt)
        {
            btnPrintActionPerformed(evt);
        }
    });
```



```

        lpBackgroundLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(lpBackgroundLayout.createSequentialGroup()
            .addGap(15, 15, 15)
            .addComponent(lblHelpScreen)
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
            .addComponent(spDisplay, javax.swing.GroupLayout.PREFERRED_SIZE, 237,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(15, 15, 15)

        .addGroup(lpBackgroundLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
            .addComponent(btnPrint)
            .addComponent(btnClose))
        .addContainerGap(15, Short.MAX_VALUE))
    );

    javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
    getContentPane().setLayout(layout);
    layout.setHorizontalGroup(
        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addGap(15, 15, 15)
            .addComponent(lpBackground, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
            .addContainerGap(15, Short.MAX_VALUE))
    );
    layout.setVerticalGroup(
        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addGap(15, 15, 15)
            .addComponent(lpBackground, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
            .addContainerGap(45, Short.MAX_VALUE))
    );

    setSize(new java.awt.Dimension(696, 434));
    setLocationRelativeTo(null);
} // </editor-fold>

```

```

private void btnCloseActionPerformed(java.awt.event.ActionEvent evt)
{
    this.dispose();
}

```

```

private void btnPrintActionPerformed(java.awt.event.ActionEvent evt)
{
    try

```

```

    {
        txaDisplay.print();

    } catch (Exception e)
    {
        JOptionPane.showMessageDialog(null, "Printing Failed", "Error", JOptionPane.ERROR_MESSAGE);
    }
}

public static void main(String args[])
{
    new HelpFrame("testHelp.txt").setVisible(true);
}

// Variables declaration - do not modify
private javax.swing.JButton btnClose;
private javax.swing.JButton btnPrint;
private javax.swing.JLabel lblHelpScreen;
private javax.swing.JLayeredPane lpBackground;
private javax.swing.JScrollPane spDisplay;
private javax.swing.JTextArea txaDisplay;
// End of variables declaration
}

```