CS 1120 – Computer Science II (with Java)

Instructor: <Bob Hardin>, Western Michigan University

Lab TA: <Nishant Gupta>

<Sam Stanchak, Blake Wrege>

Repository Name: pp4

SOFTWARE LIFE CYCLE REPORT – FOR LAB ASSIGNMENT <6>

**PHASE 1: SPECIFICATION (“What do we build?”)**

This program is a GUI application that works like a checkpoint system for a country named Atlantis. A person entering the country can enroll specifying their passport details and personal information. An enrollment officer can see all of the enrollments made via loading a text file in a GUI that displays the information in the file. GUI code and data should be in separate classes, there should be getters for all file names, and all GUI elements should be properly labeled.

**PHASE 2: DESIGN**

**Basic Structure:**

This program is split into four classes:

1. AtlantisMenu - First GUI that’s initialized when the program starts. This allows you to select between loading enrollments or entering a new enrollee.

2. EnrollmentForm - For new enrollments. Allows the user to enter their information and save it to the text file.

3. LoadEnrollment - For the enrollment officer to load the data in the text file into the GUI to see who has enrolled and their corresponding information.

4. MainTest - Contains the main method which initializes AtlantisMenu and contains the ArrayList which contains an enrollees information.

**Pseudocode:**

AtlantisMenu:

-Create GUI using WindowBuilder

-Add ActionListeners to buttons

-In ActionListeners, initialize EnrollmentForm or LoadEnrollment and set AtlantisMenu invisible

EnrollmentForm:

-Create GUI using WindowBuilder

-Accept information ArrayList in constructor

-Create getters for information

-Create checkValues method to test if values have been entered into fields and if they have, enable “Save” button

-Add ActionListeners to:

1. Initialize AtlantisMenu and set EnrollmentForm invisible when clicking “Menu”

2. Add to the information ArrayList using the getters, use a FileWriter to write to the file “data.txt” using try-catch blocks for exceptions when clicking “Save”

-Add KeyListeners to all relevant TextFields which call the checkValues method every time a key is released

-Add ItemListeners to all relevant ComboBoxes which call the checkValues method whenever an item has been changed

LoadEnrollment:

-Create GUI using WindowBuilder

-Accept information ArrayList in constructor

-Create checkValues method to test if values have been entered into the filename TextField and if there has been, enable “Load” button

-Add KeyListener to filename TextField which calls the checkValues method every time a key is released

-Add ActionListener to:

1. Initialize AtlantisMenu and set LoadEnrollment invisible when clicking “Menu”

2. Use a scanner to read inputted filename line by line and print it in the TextArea when clicking “Load”

MainTest:

-Create ArrayList of strings for information

-Initialize AtlantisMenu and pass information to it

**PHASE 3: RISK ANALYSIS (“What can go wrong, and how bad can it be?”)**

There is no security on this program. A enrollee without supervision from an enrollment officer could find the text file which contains everyone’s information and steal it or edit it without permission.

**PHASE 4: VERIFICATION (“Are the algorithms correct?”)**

The program has one sequential logic path and it has been verified by us.

**PHASE 5: CODING**

package lab6;

import javax.swing.JFrame;

/\*\*

\* This class contains the AtlantisMenu GUI which allows you to select between EnrollmentForm and LoadEnrollment.

\* @author Sam, Blake

\* @version 1.0

\*/

public class AtlantisMenu extends JFrame {

/\*\*

\* This constructor initializes the main menu GUI and contains ActionListeners for "Load Enrollment" and "New Enrollment".

\* @param information Arraylist which contains information entered in by the user.

\*/

public AtlantisMenu(final ArrayList<String> information){

btnLoadEnrollment.addActionListener(new ActionListener() {

/\*\*

\* This method sets the AtlantisMenu GUI invisible, sets LoadEnrollment GUI visible, and passes information

\* when "Load Enrollment" is pressed.

\*/

public void actionPerformed(ActionEvent arg0) {

}

});

btnNewEnrollment.addActionListener(new ActionListener() {

/\*\*

\* This method sets the AtlantisMenu GUI invisible, sets EnrollmentForm GUI visible, and passes information

\* when "New Enrollment" is pressed.

\*/

public void actionPerformed(ActionEvent e) {

}

});

}

private void buildPanel() {

}

}

----------------------------------------------------------------------------------------------------------------------------

package lab6;

import javax.swing.\*;

/\*\*

\* This class contains the EnrollmentForm GUI and writes information to data.txt when saved.

\* @author Sam, Blake

\* @version 1.0

\*/

public class EnrollmentForm extends JFrame {

/\*\*

\* This consturctor initializes the GUI and contains ActionListeners for the buttons "Menu" and "Save".

\* @param information The ArrayList where information entered by the user is put.

\* @param parent References AtlantisMenu.

\*/

public EnrollmentForm(final ArrayList<String> information, final AtlantisMenu parent) {

btnMenu.addActionListener(new ActionListener() {

/\*\*

\* This method sets the EnrollmentForm GUI invisible and sets the parent AtlantisMenu GUI visible when "Menu" is pressed.

\*/

public void actionPerformed(ActionEvent arg0) {

}

});

btnSave.addActionListener(new ActionListener() {

/\*\*

\* This method saves the information the user entered/selected in the fields into the ArrayList information.

\* The ArrayList information is saved into "data.txt" using a FileWriter.

\*/

public void actionPerformed(ActionEvent e) {

}

});

}

private void buildPanel() {

}

/\*\*

\* Getter method for passport type.

\* @return Passport type as a string.

\*/

public String getPassportType() {}

/\*\*

\* Getter method for passport number.

\* @return Passport number.

\*/

public String getPassportNo() {}

/\*\*

\* Getter method for first name.

\* @return First name.

\*/

public String getFirstName() {}

/\*\*

\* Getter method for last name.

\* @return Last name.

\*/

public String getLastName() {}

/\*\*

\* Getter method for country.

\* @return Country as a string.

\*/

public String getCountry() {}

/\*\*

\* Getter method for birth place.

\* @return Birth place.

\*/

public String getBirthPlace() {}

/\*\*

\* Getter method for visa type.

\* @return Visa type as a string.

\*/

public String getVisaType() {}

/\*\*

\* Getter method for visa number.

\* @return Visa number.

\*/

public String getVisaNo() {}

/\*\*

\* Getter method for remarks.

\* @return Remarks.

\*/

public String getRemarks() {}

/\*\*

\* This method checks the values where you input your informationm and if any are empty, the "Save" button is disabled. KeyListeners are used

\* to call this method every time a key is released. ItemListeners are used for the ComboBoxes.

\*/

public void checkValues() {

}

}

----------------------------------------------------------------------------------------------------------------------------

package lab6;

import javax.swing.JFrame;

/\*\*

\* This class contains the LoadEnrollment GUI and loads information from data.txt.

\* @author Sam, Blake

\* @version 1.0

\*/

public class LoadEnrollment extends JFrame {

/\*\*

\* This consturctor initializes the GUI and contains ActionListeners for the buttons "Menu" and "Load".

\* @param information The ArrayList where information entered by the user is put.

\* @param parent References AtlantisMenu.

\*/

public LoadEnrollment(final ArrayList<String> information, final AtlantisMenu parent) {

btnLoad.addActionListener(new ActionListener() {

/\*\*

\* This method loads data from "data.txt" using a scanner and sets the text to the textArea when the "Load" button is selected.

\*/

public void actionPerformed(ActionEvent e) {

}

});

btnMenu.addActionListener(new ActionListener() {

/\*\*

\* This method sets the LoadEnrollment GUI invisible and sets the parent AtlantisMenu GUI visible when "Menu" is pressed.

\*/

public void actionPerformed(ActionEvent arg0) {

}

});

}

private void buildPanel() {

}

/\*\*

\* This method checks the value where you input the file name and if it is empty, the "Load" button is disabled. KeyListeners are used

\* to call this method every time a key is released.

\*/

public void checkValues() {

}

}

----------------------------------------------------------------------------------------------------------------------------

package lab6;

import java.util.ArrayList;

/\*\*

\* This class contains the main method.

\* @author Sam, Blake

\* @version 1.0

\*/

public class MainTest {

/\*\*

\* The main method contains the ArrayList with information needed and also initializes the AtlantisMenu.

\* @param args Command line arguments.

\*/

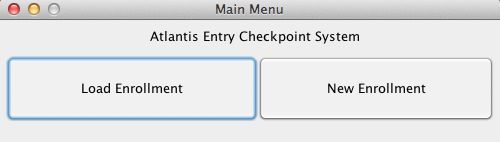
public static void main(String[] args) {

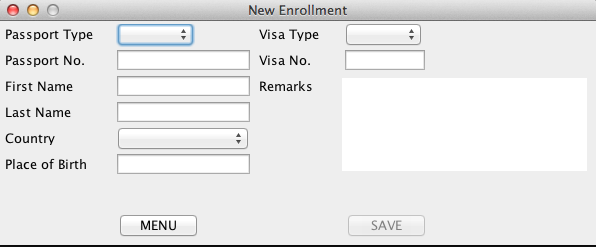
}

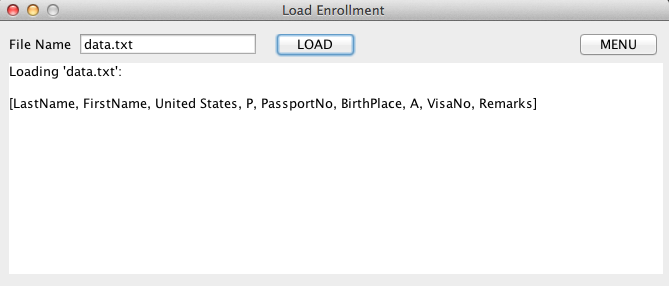
}

**PHASE 6: TESTING (“Did we build it correctly?”)**

**Example Output:**







**PHASE 7: REFINING THE PROGRAM (“Add bells and whistles to the program”)**

Features Added:

Enrollees cannot click the save button until they have provided input for all the fields necessary. Enrollment officers cannot click the load button until they have entered a filename to load.

**PHASE 8: PRODUCTION**

We have prepared this SLC and the program itself for my TA for grading. Each team member’s individual contributions can be found in the readme file in our repository.

**PHASE 9: MAINTENANCE**

We shall use my TA’s feedback to perform maintenance if necessary.