|  |
| --- |
|  |
| Assignment #4 |
| CMPT 270 – Intersession 2013 |
|  |
| **Submitted by: Bo Dong (bod458) and Tyler Spink (tws167)** |
| **Due: June 24, 2013** |

|  |
| --- |
|  |

Table of Contents

**1. UML Classification Model2**

**2. UML State Transition Diagram3**

**3. UML Interaction Diagram 3**

**4. Source Code5**

**5. XML Data Storage 5**

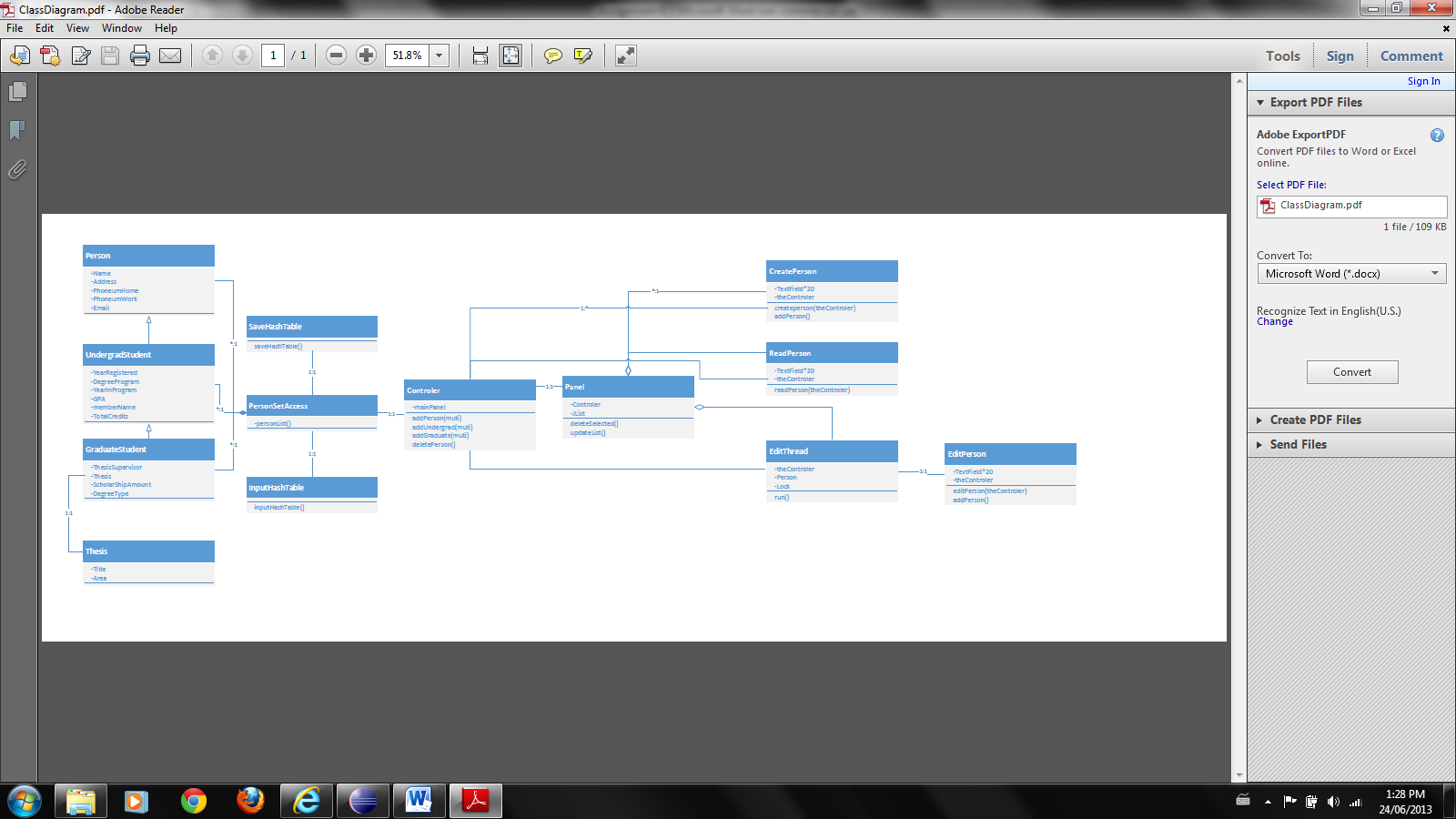
**6. Documentation of GUI 6**

**7. Limitations 9**

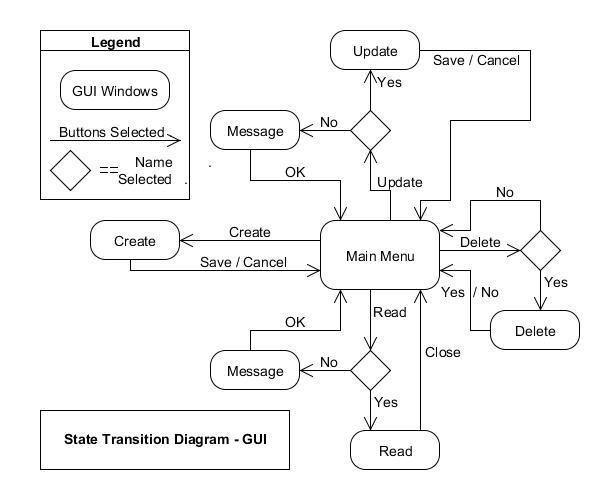
**8. Assumptions 9**

**9. Testing 9**

1. UML Classification Model

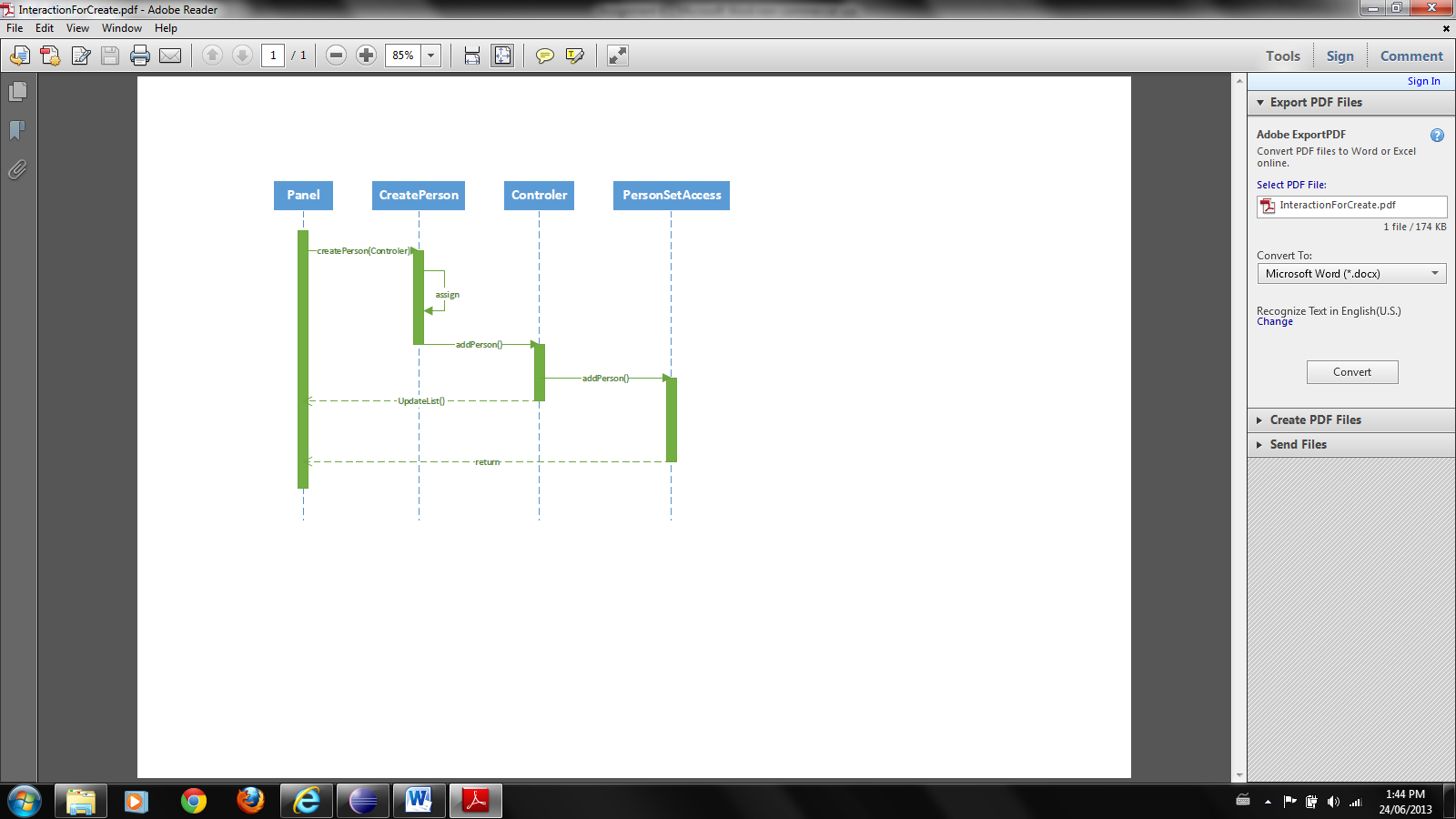


2. UML State Transition Diagram

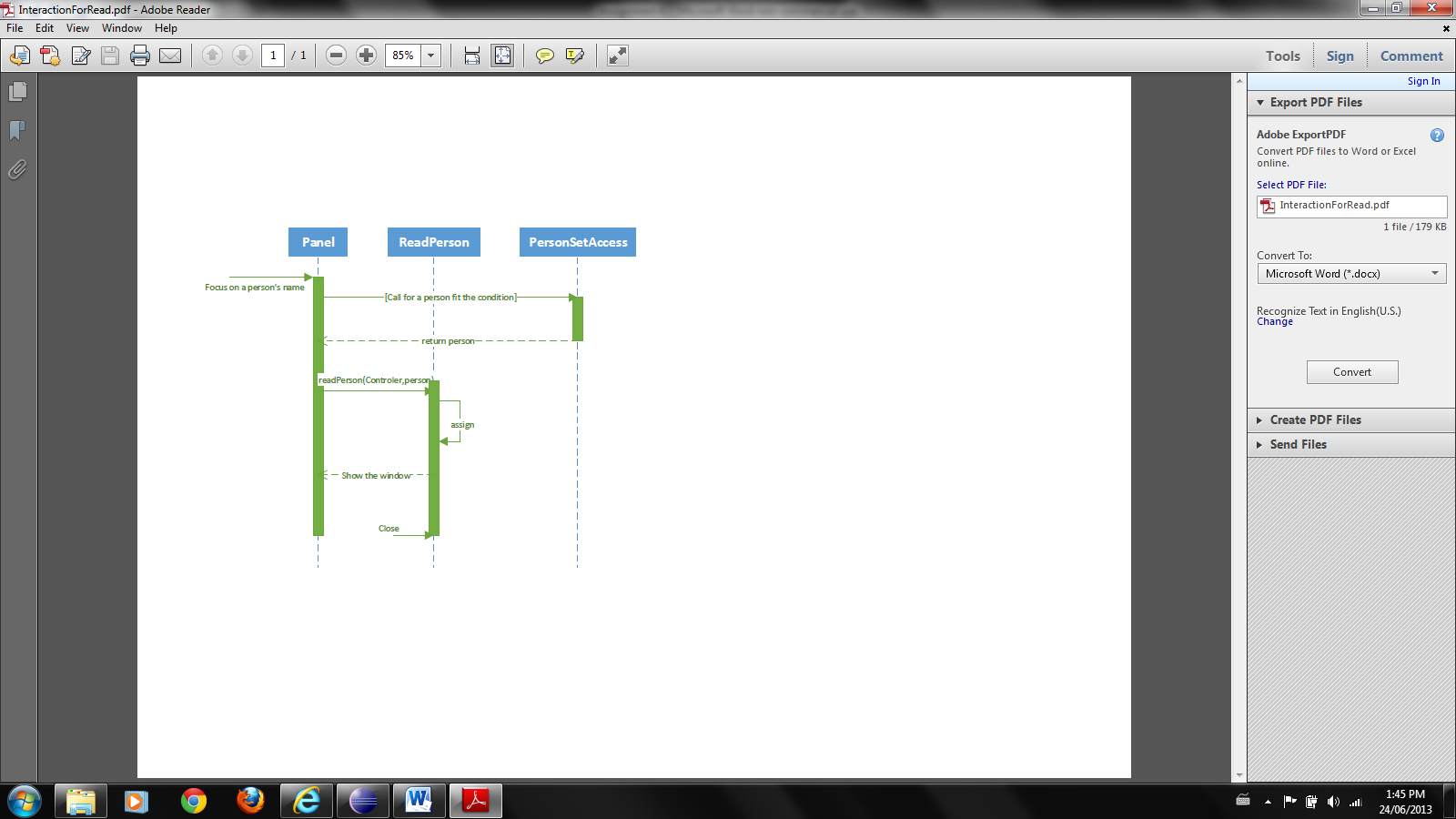


3. UML Interaction Diagrams

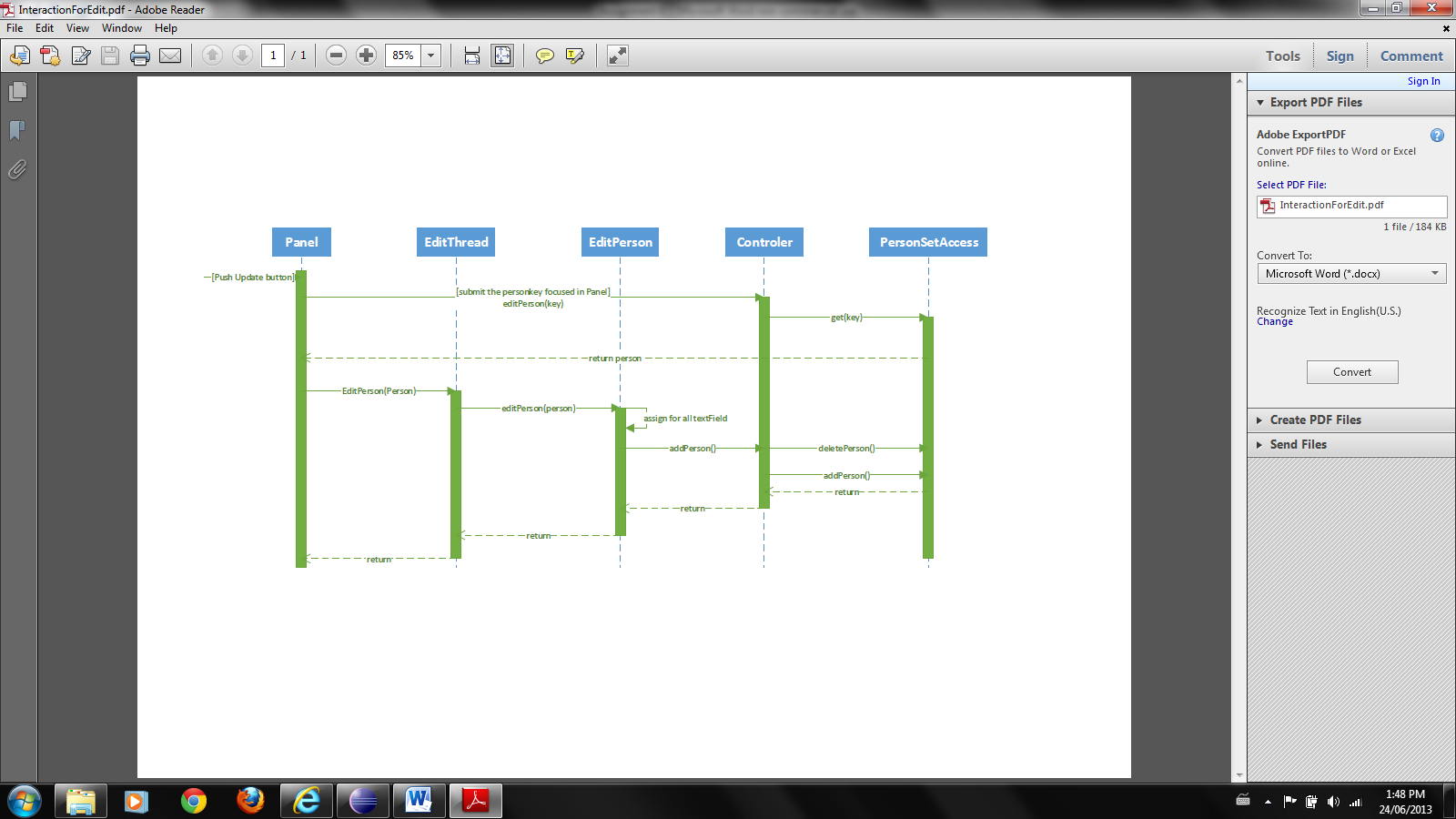
Creating a new person:



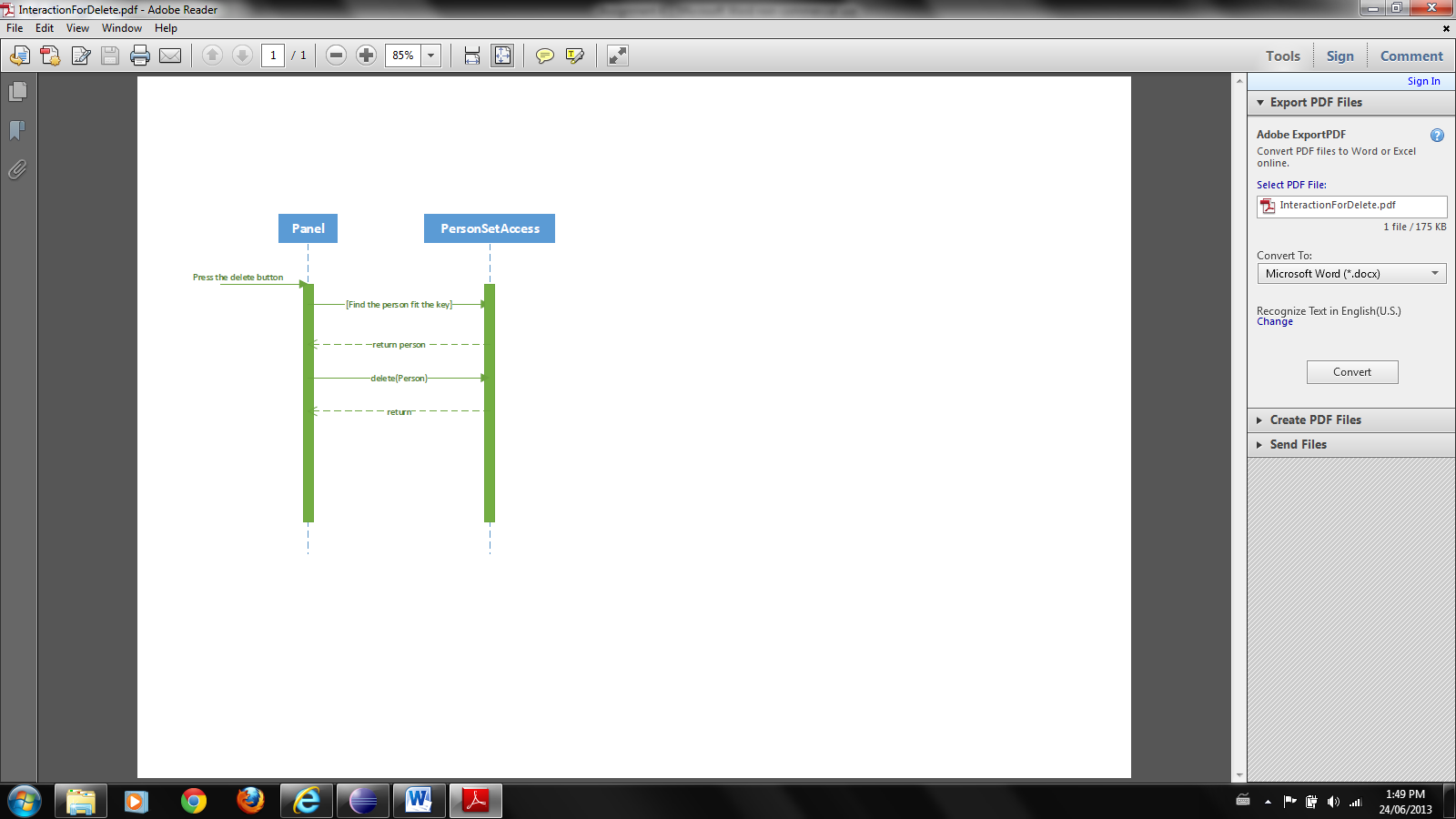
Viewing a selected person:



Updating a selected person:

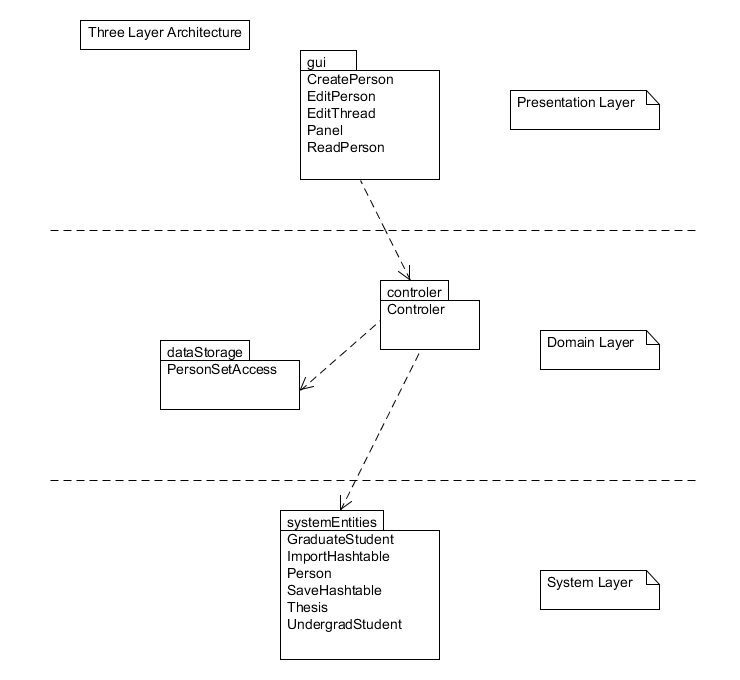


Deleting a selected person:



4. Source code

See included files for the source code of all the classes shown below:



5. File created for storing data between sessions

Name: Hashtable.xml

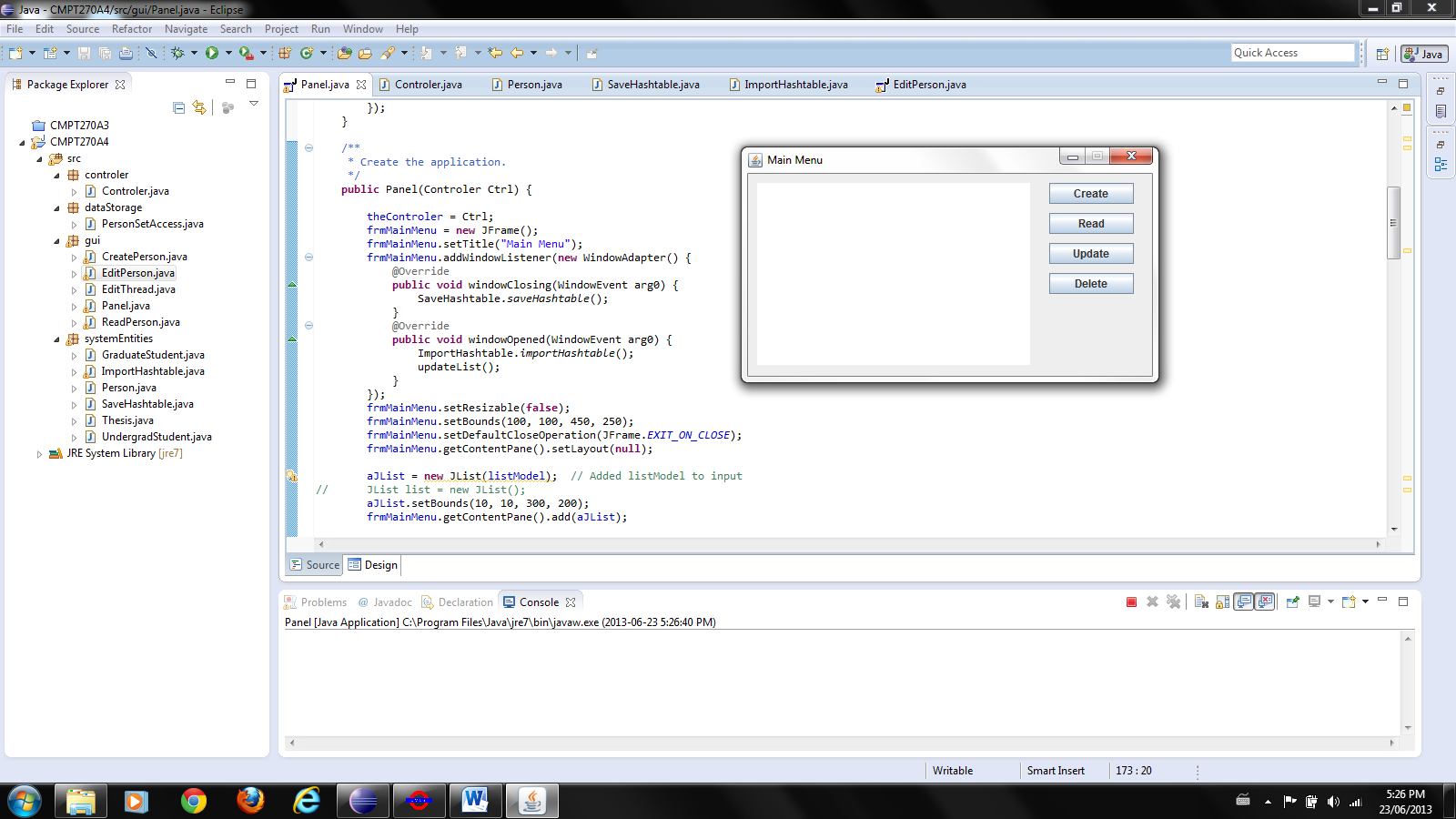
Type: XML format

Stores: Hashtable entries (People types and fields)

Note: See included file ‘Hashtable.xml’ for example of three different types of people stored.

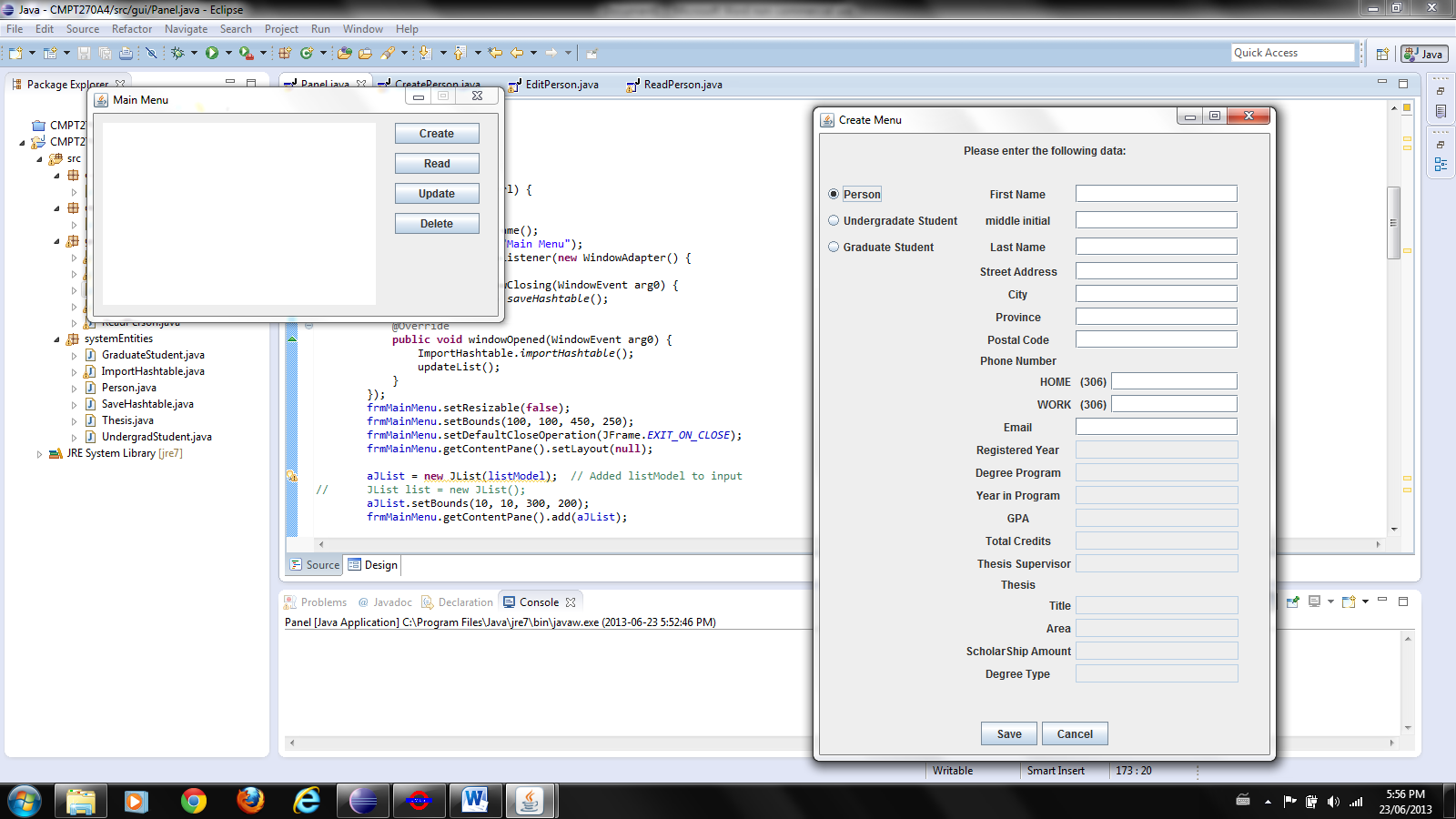
6. Documentation of the GUI

Once the program has started up the main menu appears as follows:



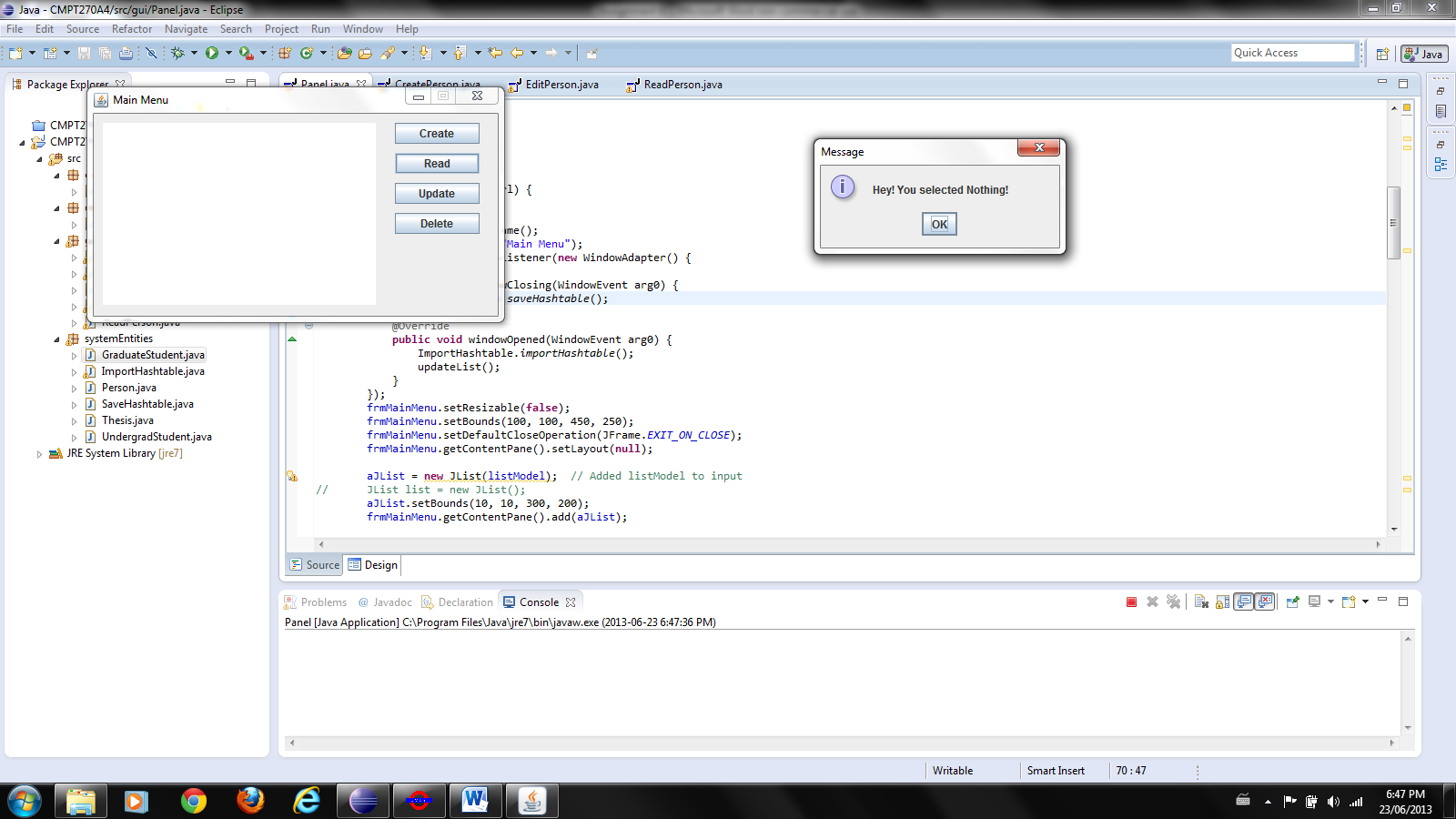
The user has four main options for manipulating the data as well as an option to exit the program by selecting the ‘x’ in the top right hand corner of the main menu.

The first of the four main options is ‘Create’, selecting this option pops up the ‘Create Menu’ which appears as follows:



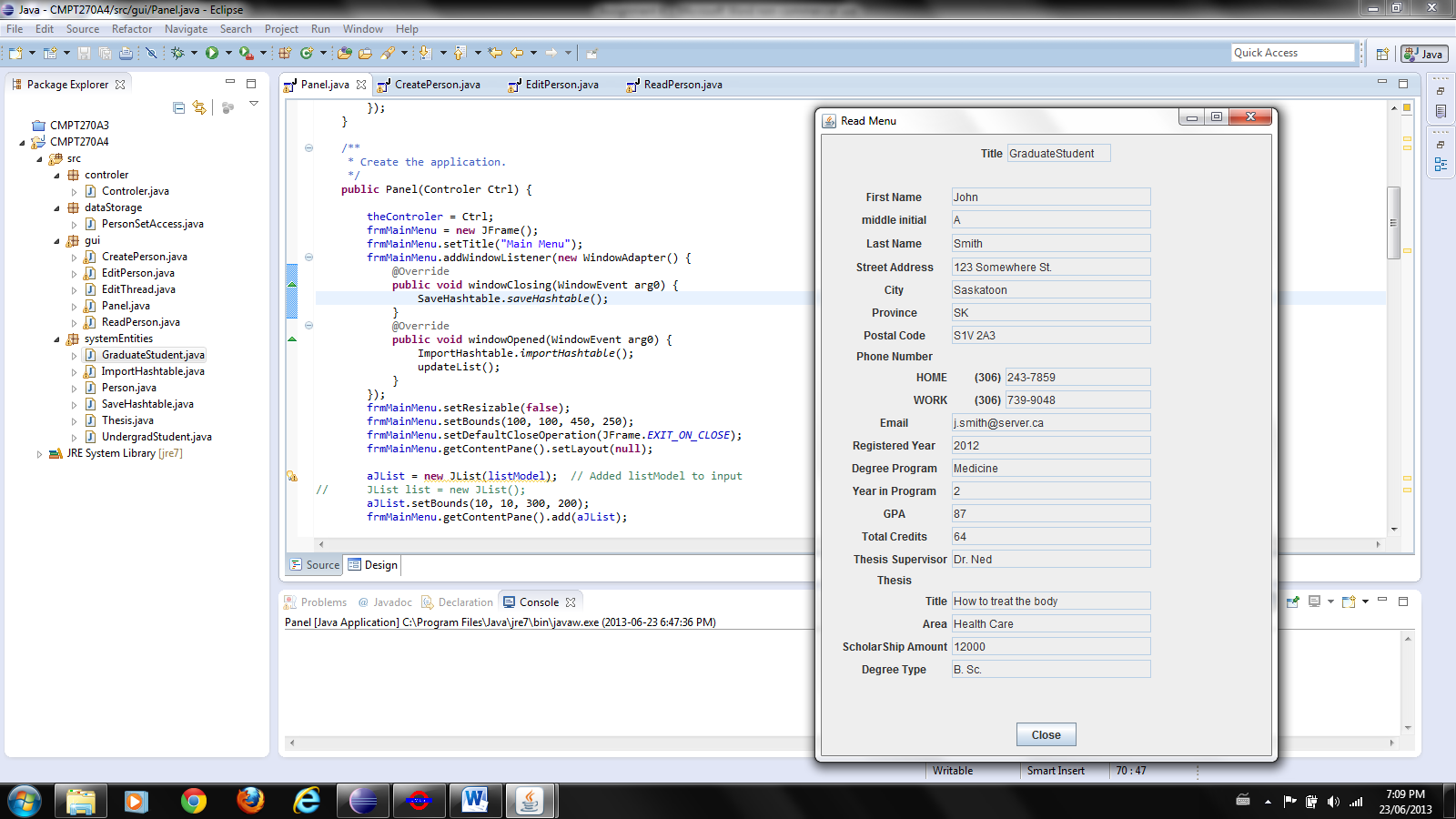
From here the user can select which type of person (non-student, undergrad, or grad student) they want to create by selecting the corresponding radio button. Once that has been selected the corresponding fields will be made editable for the user to enter the respective data. After the minimum amount of data has been entered (this includes a first name, middle initial, and last name), the user may click the ‘Save’ button to save any data they have entered and return to the main menu. The user also has the option to click the ‘Cancel’ button if they wish to return to the main menu without saving their data.

The next main menu options is ‘Read’, if this button is selected without selecting a person the following dialog message will appear:



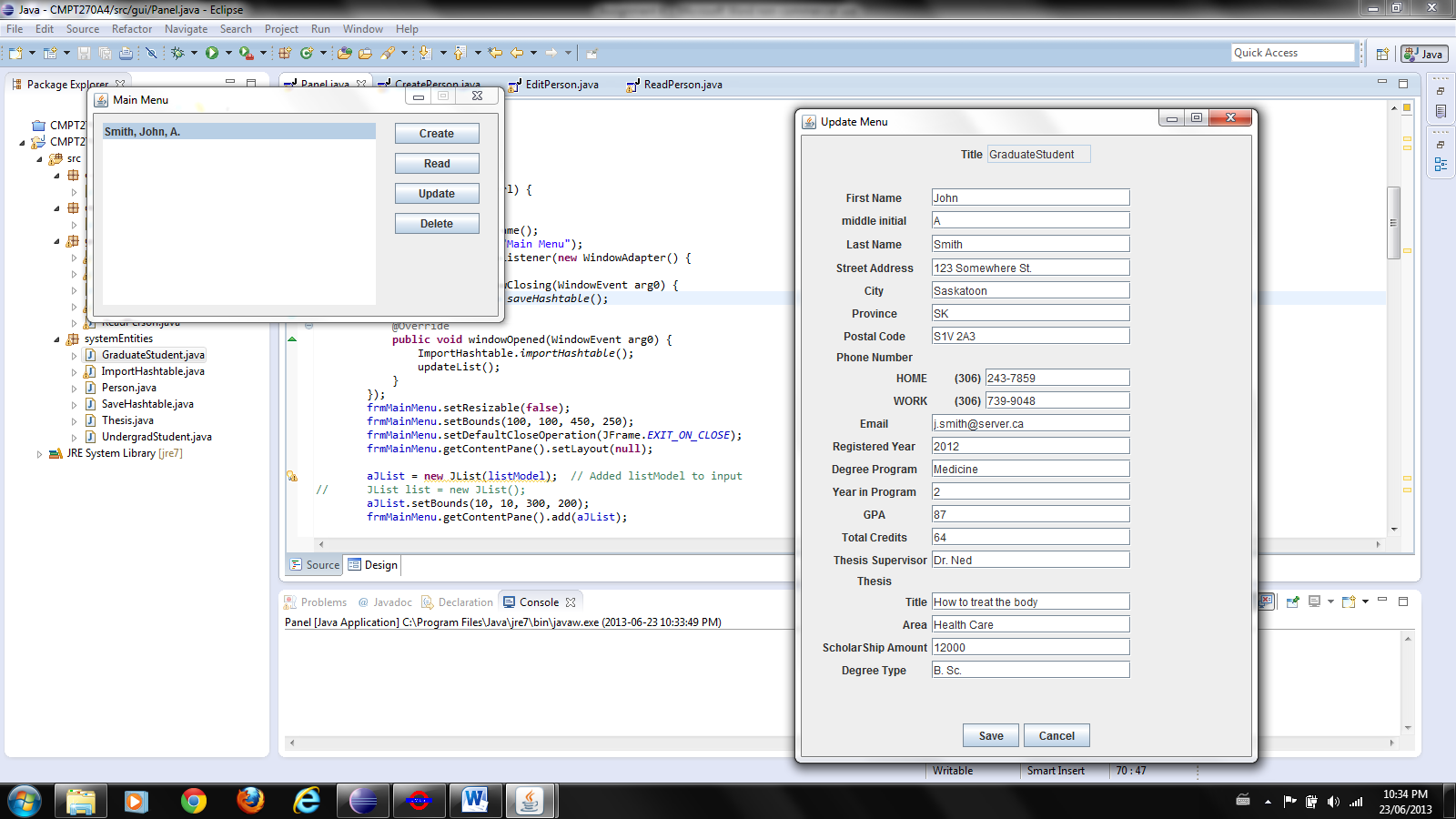
This message informs the user that a selection was not made before selecting the read option. Selecting the ‘OK’ button will return the user to the main menu.

If the ‘Read’ button is selected when a person is selected, then the ‘Read Menu’ will pop up as follows:



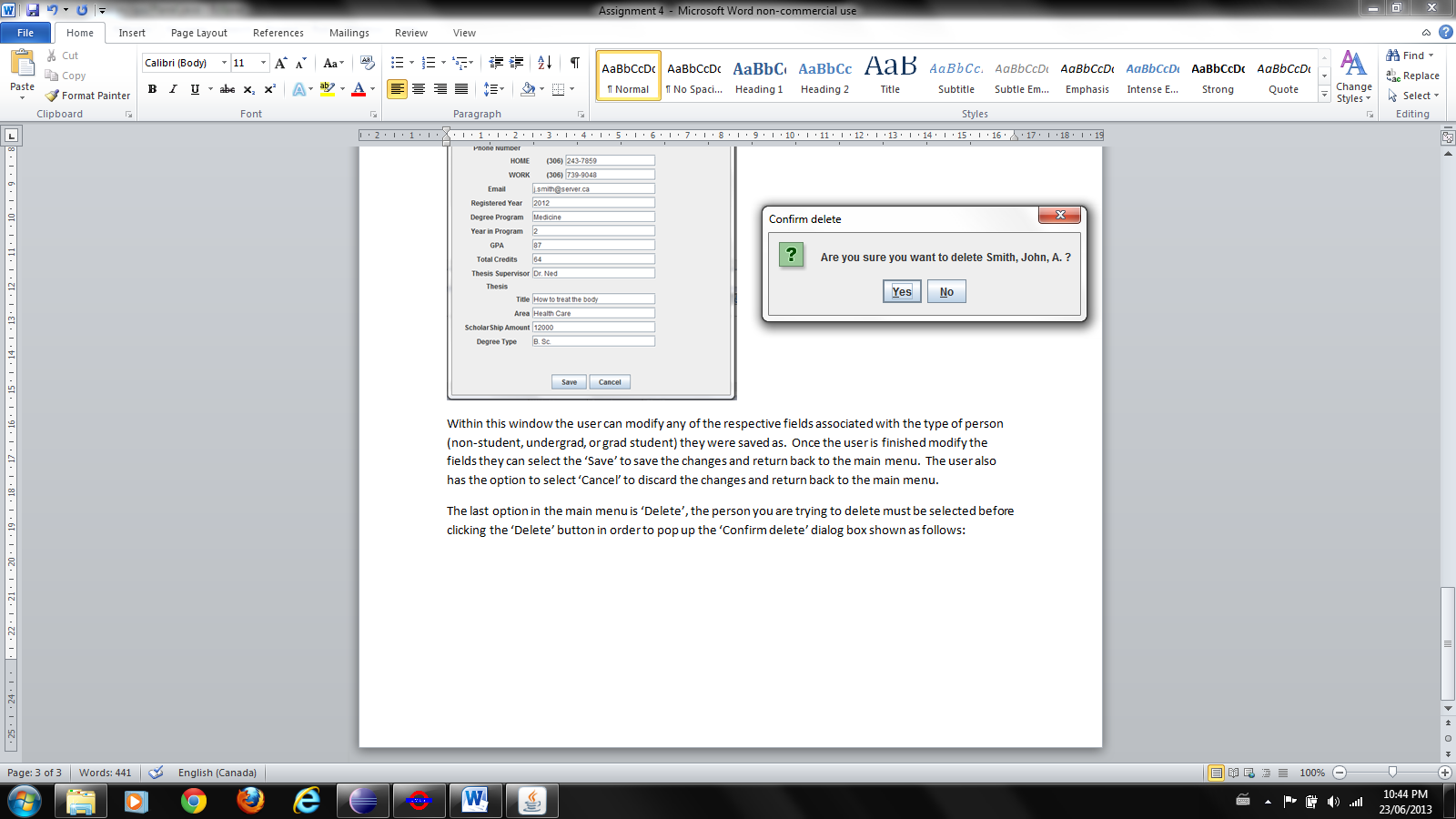
The user can review the information saved for that selected person in this window. Once the user is done viewing, they can select the ‘Close’ button to take them back to the main menu.

The third option in the main menu is ‘Update’. If this is selected without having a person selected the same dialog window will pop up as it did for selecting the ‘Read’ option. When ‘Update’ is selected with a person highlighted it pop up the ‘Update Menu’ window seen as follows:



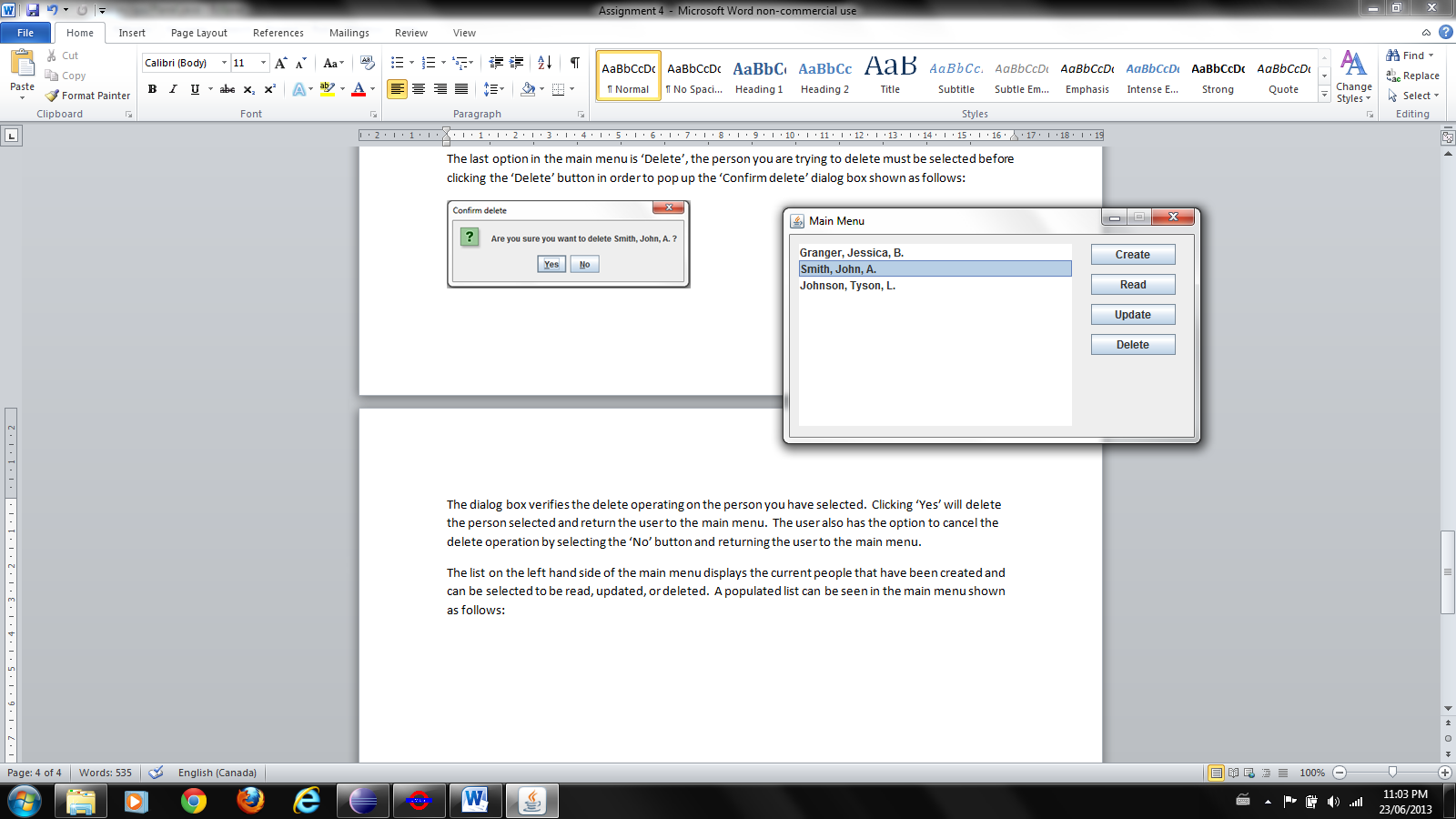
Within this window the user can modify any of the respective fields associated with the type of person (non-student, undergrad, or grad student) they were saved as. Once the user is finished modify the fields they can select the ‘Save’ to save the changes and return back to the main menu. The user also has the option to select ‘Cancel’ to discard the changes and return back to the main menu.

The last option in the main menu is ‘Delete’, the person you are trying to delete must be selected before clicking the ‘Delete’ button in order to pop up the ‘Confirm delete’ dialog box shown as follows:



The dialog box verifies the delete operating on the person you have selected. Clicking ‘Yes’ will delete the person selected and return the user to the main menu. The user also has the option to cancel the delete operation by selecting the ‘No’ button and returning the user to the main menu.

The list on the left hand side of the main menu displays the current people that have been created and can be selected to be read, updated, or deleted. A populated list can be seen in the main menu shown as follows:



The main menu shows three different names with three different types of people (non-student, undergrad, or grad student) in the list with ‘Smith, John, A.’ being currently selected. The list is automatically refreshed every time a modification is made to the list such as create, update, or delete to reflect the respective change.

7. Limitations

One of the limitations in our program is that once you create a person and select their type (non-student, undergrad, or graduate student) you cannot change their type. We made this choice as we made the assumption that the data would be stored yearly and deleted at the end of each year and therefore wouldn’t have the need to change types during the year. If a type does need to be changed, the user can create a new person of the new type and open up the old data to reference and transfer over data that remains unchanged and delete the old data once the new person has been created.

8. Assumptions

An assumption we made with our program is that the user will input the correct data in a format that will be readable and in the correct field. This allows the user to customize the fields as they see fit and allows for more flexibility. An example is that we don’t force the user to enter data into a field, if a person doesn’t have a work phone number then that field can be left blank or a note such as ‘Same as home phone’ can be written in that field.

9. Testing

We used a hybrid of testing methods at different stages when creating the program. When creating the program we used a hybrid of TDD, bottom-up testing, human testing, and regression testing. Once the program had been created we used a combination of black box testing and boundary-value testing.