**RED TEAMS DELIVERABLES FEEDBACK “blue print” UML Modeling with actual examples.**

**MODELING** **Definitions:**

* **“The Requirements”** - “are contained on pages 1, 2, …. and 11 starting with Line Number 1 and ending with Line Number 147”. Please think of it as a .pdf file that cannot be modified.
* **“The Requirements”** – are not repeated for Step 1: Identify Actor, then copied again, thus here there are again on pages 12, 13, 14, and 23 for Step 2: Identify Use Cases; they stay on pages 1, 2, …. and 11 with Line Numbers 1 to 147 only. You apply Step 1 on **“The Requirements”**, and then you apply Step 2 on **“The Requirements”**. Think of **“The Requirements”** as being a .pdf 11 pages document. I can scan it if you desire so. If you Requirements have less than or more than “**The Requirements”** you lost up to 10 points.

**1. Feedback “SE Team Project with Line Numbers for UML USE CASE DIAGRAM.doc”**

**USE CASE MODELING:**

* **Step 1** template does not have a box around the **A Label** like this:

**A1: User**



A1: User



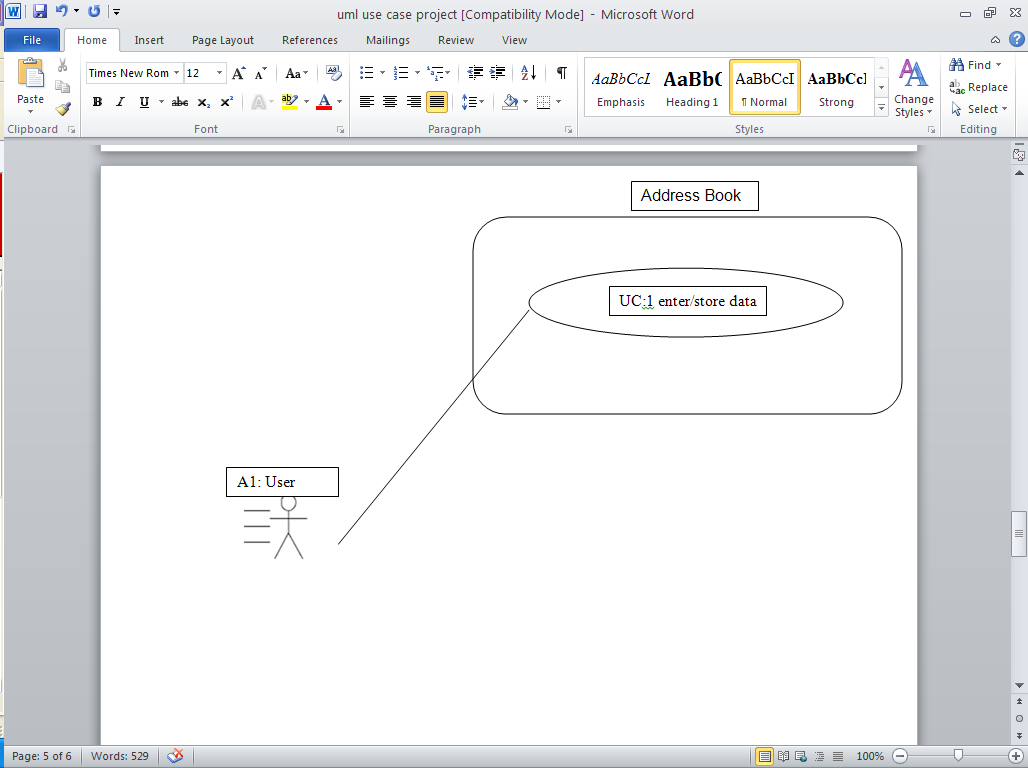
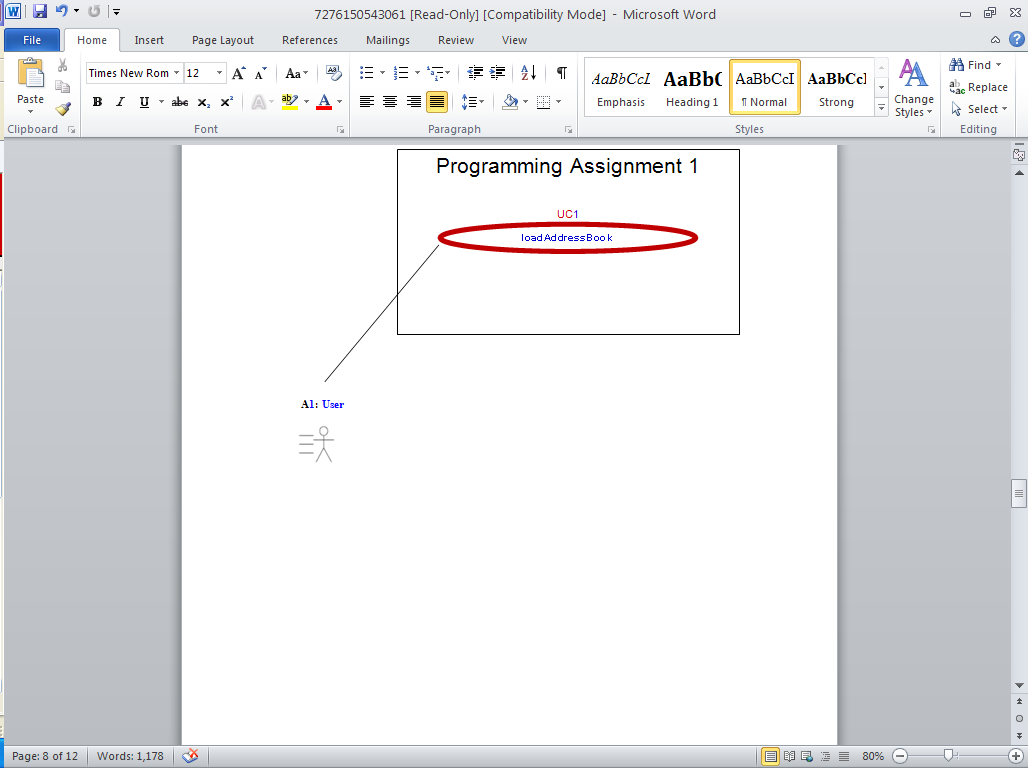
* **Step 2** template does not have a box around the **UC Label** like this:

UC:1 Store/enter data

UC1: **loadAddressBook**

If you did not use the UML USE CASE DIAGRAM templates you lost up to 20 points.

* **Step 3** is the “Select All Templates you placed/annotated on **“The Requirements”** i.e., pages 1, 2, …. and 11 “, “Cut” , and “Paste” them on a fresh page 12, then rearrange according to the UML USE CASE DIAGRAM notation:

NOTE: There is no need that on page 12 you present a summary of the UC s the only place they exist is as annotations you placed on **“The Requirements”** i.e., pages 1, 2, …. and 11 as part of Steps 1 & 2. What goes on page 12 is the UML USE CASE DIAGRAM in one place not a different one per each actor where UCs are repeated.

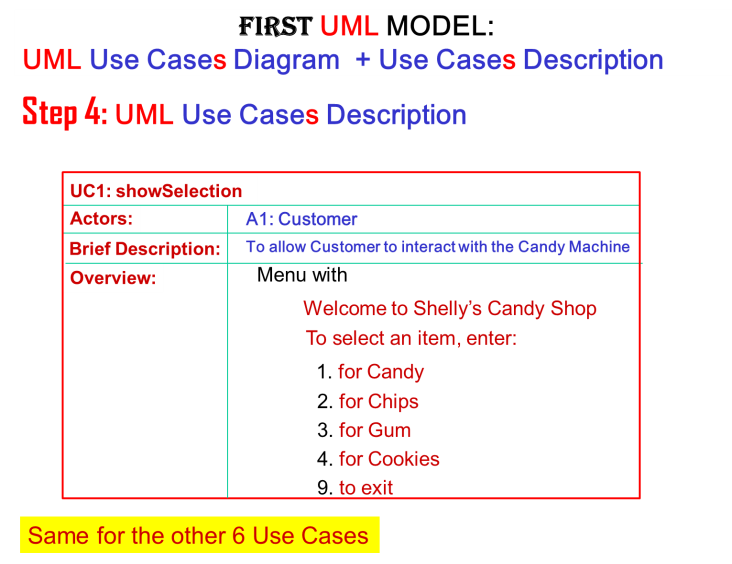
NOTE: UML USE CASE DIAGRAM must be created in Word. If you did not use Word you lost up to 20 points.

If the UML USE CASE DIAGRAM does not match 100% the annotations you placed on the **“The Requirements”** i.e., pages 1, 2, …. and 11 you lost up to 30 points.

* **Step 4** is the “Scenario” Description of each Labeled Use Case:

UC:1 This prompts the use to enter that data and then stores it in the appropriate places.

Must use a Template (see MRT 1 Tutorial for one or just something like this):



NOTE: The UC s become **methods** in the **UserController** Class of the **MVC** Design Pattern. In this class you are required to use **MVC.** Thus, use the “C” compilable module convention.

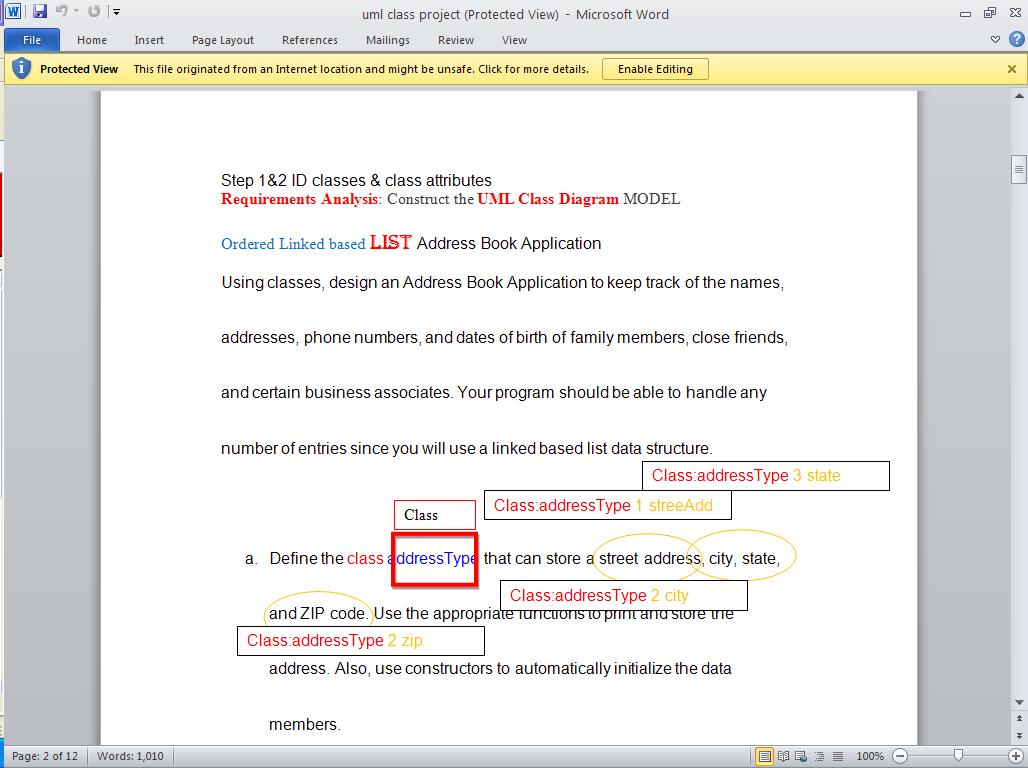
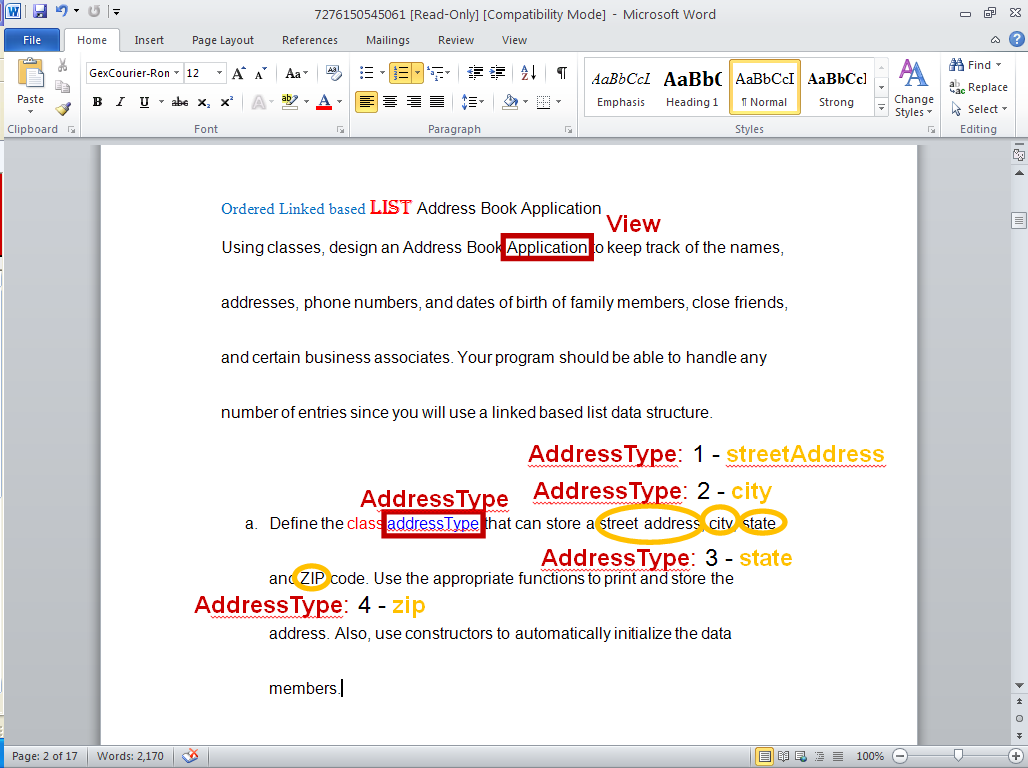
enter/store data **searchByLastName**

NOTE: Unless your UML Use Case Model is **UML “compilable”** you will be considered as having no “**blue print**” for the purpose of building the “**house**” . If you did not use the UML USE CASE Description templates you lost up to 30 points.

**2. Feedback “SE Team Project with Line Numbers for UML CLASS DIAGRAM.doc”**

**USE CLASS MODELING:**

* **Steps 1, 2**  templates do not have a box around the **Class or Attribute Labels** like this:

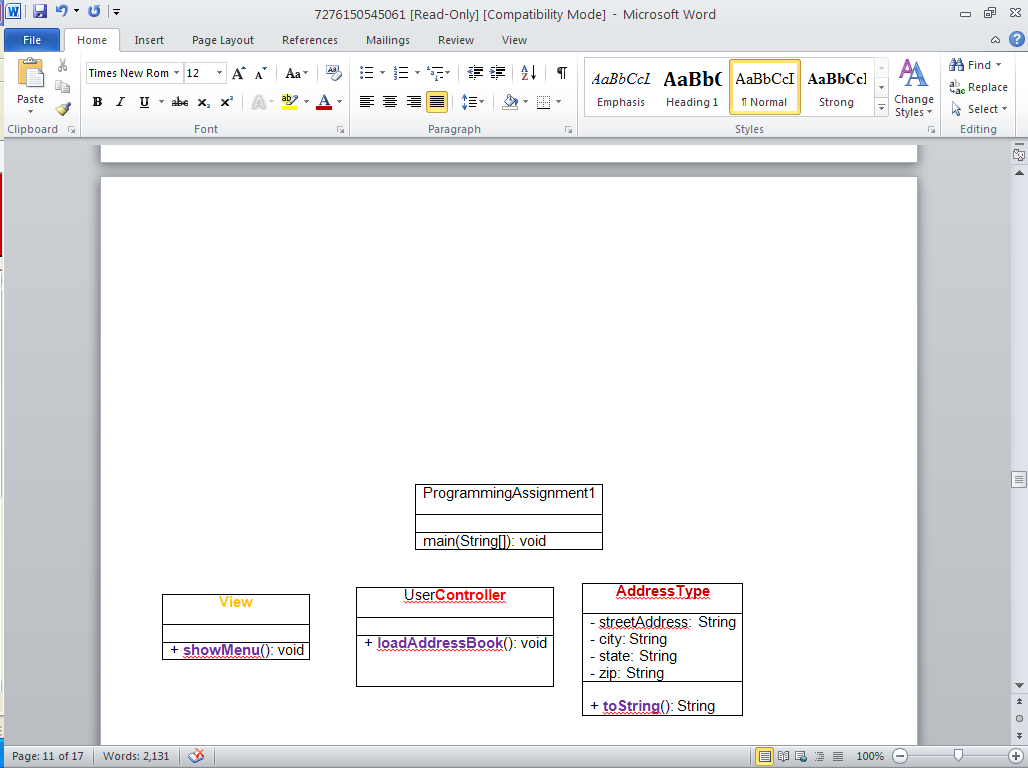
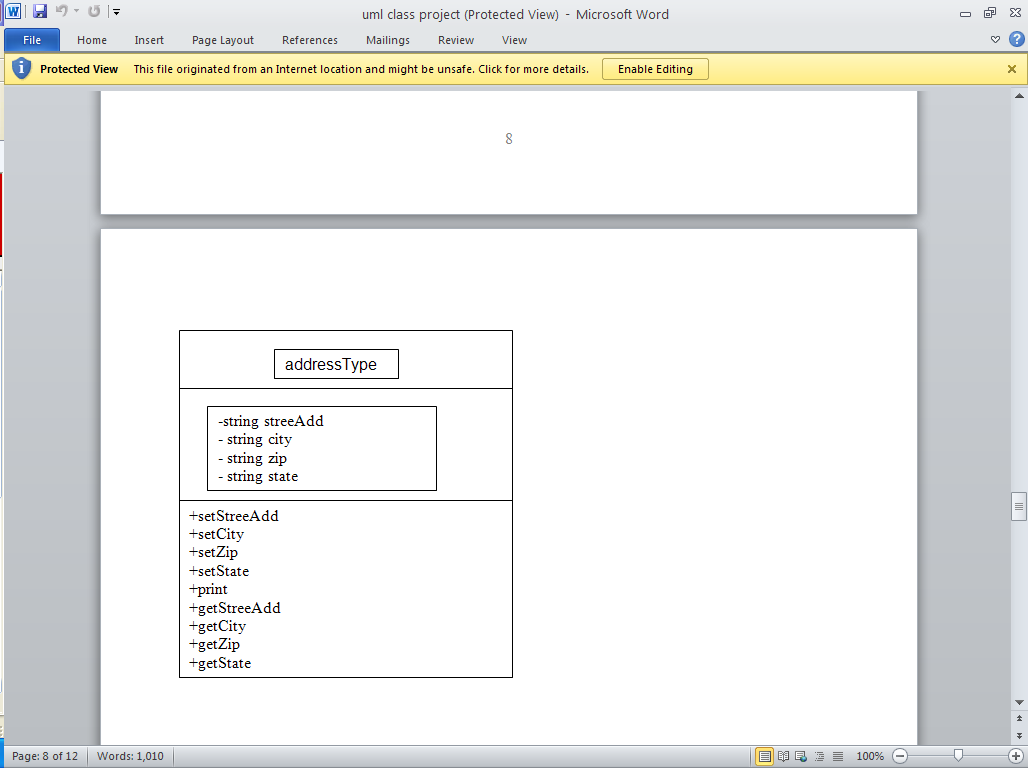
 

**NOTE: Classes start with UpperCase Letter by Convention.**

**NOTE: Attributes start with lowerCase letter by Convention.**

If you did not use the UML CLASS DIAGRAM templates you lost up to 50 points.

* **Step 4** is the “Select All Templates you placed on **“The Requirements”** , “Cut” , and “Paste” them on a fresh page 12, then rearrange according to the UML CLASS DIAGRAM notation. If the UML CLASS DIAGRAM does not match 100% the annotations you placed on the **“The Requirements”** you lost up to 40 points. If the UML CLASS DIAGRAM is not MVC you lost up to 20 points.



NOTE: There is no need that on page 12 you present a summary of the **Classes**, **Attributes**. The only place they exist is as annotations you placed on **“The Requirements”** as part of Steps 1, 2. What goes on page 12 is the UML CLASS DIAGRAM.

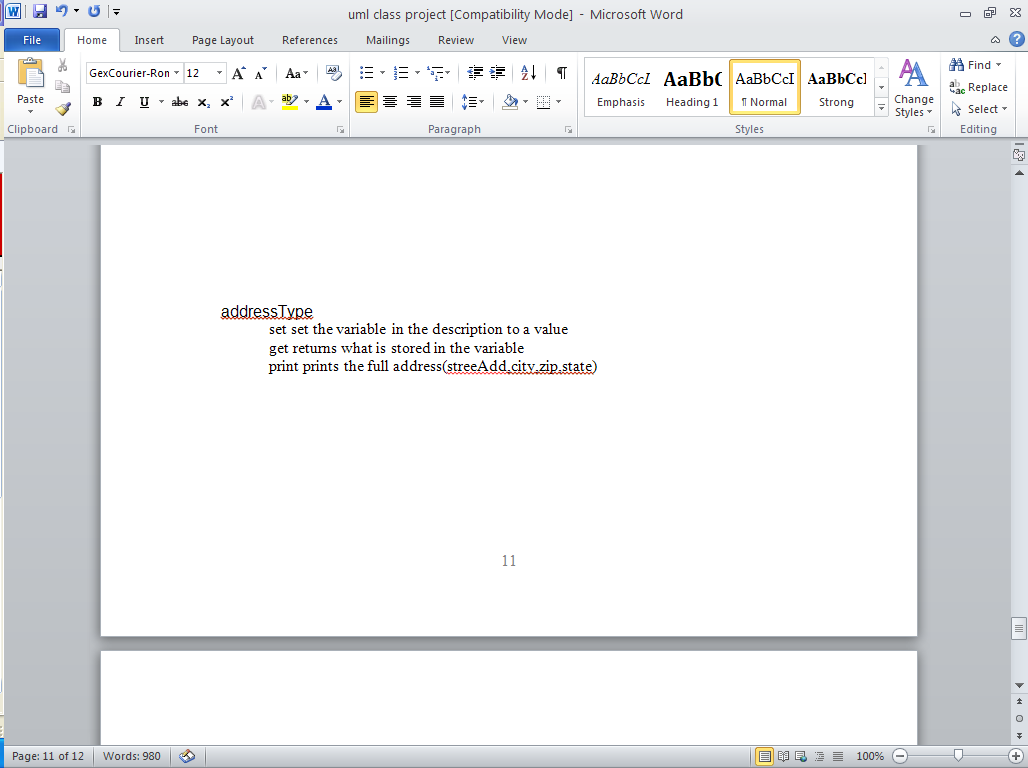
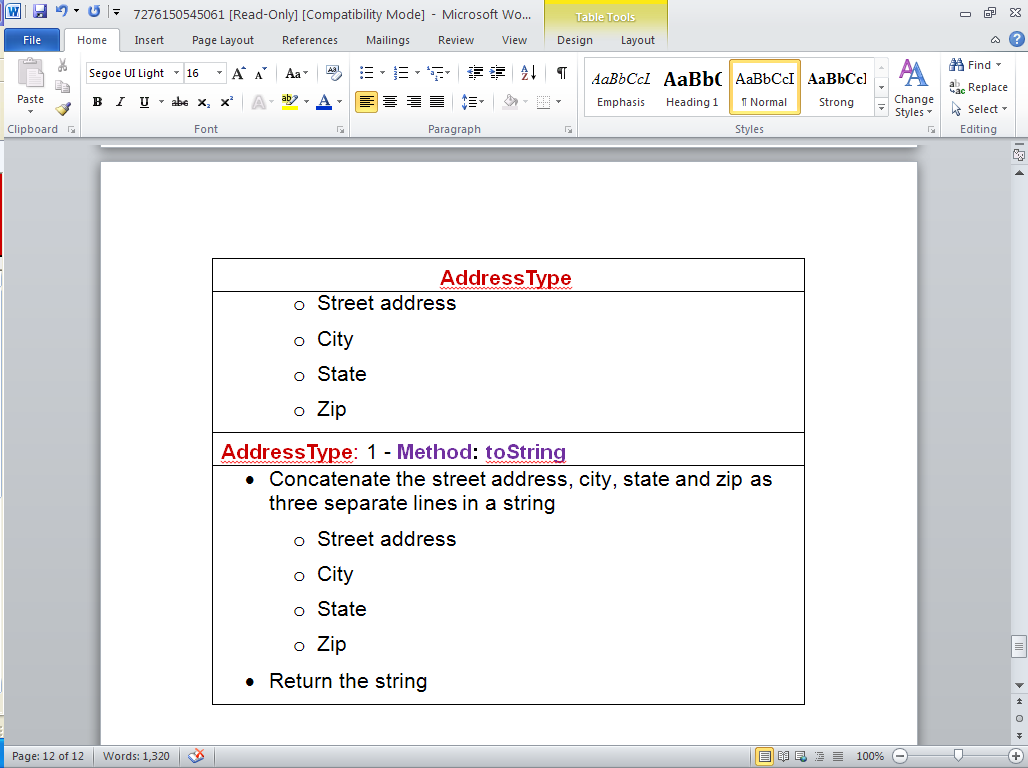
NOTE: UML CLASS DIAGRAM must be **MVC**, i.e., you need to add the **UserController** Class (UML USE CASE DIAGRAM, “right click” or “copy and paste” 100% match) and the **View** Classes.

NOTE: UML CLASS DIAGRAM must be created in Word.

If the UML CLASS DIAGRAM is not created using Word you lost up to 20 points.

This is Feedback for when you will do OOD:

* **Step 5** is the “Pseudocode” is for **methods** not for **classes**. **Methods** must have parameters and return types. Think of “Right Click” on the UML Class DIAGRAM “blue print” to “auto” generate the “house” skeleton classes needed for TDD.

NOTE: Think of “Right Click” on the UML Class DIAGRAM “blue print” to “auto” generate the “house” skeleton classes needed for TDD.

This is Feedback for when you will do TDD:

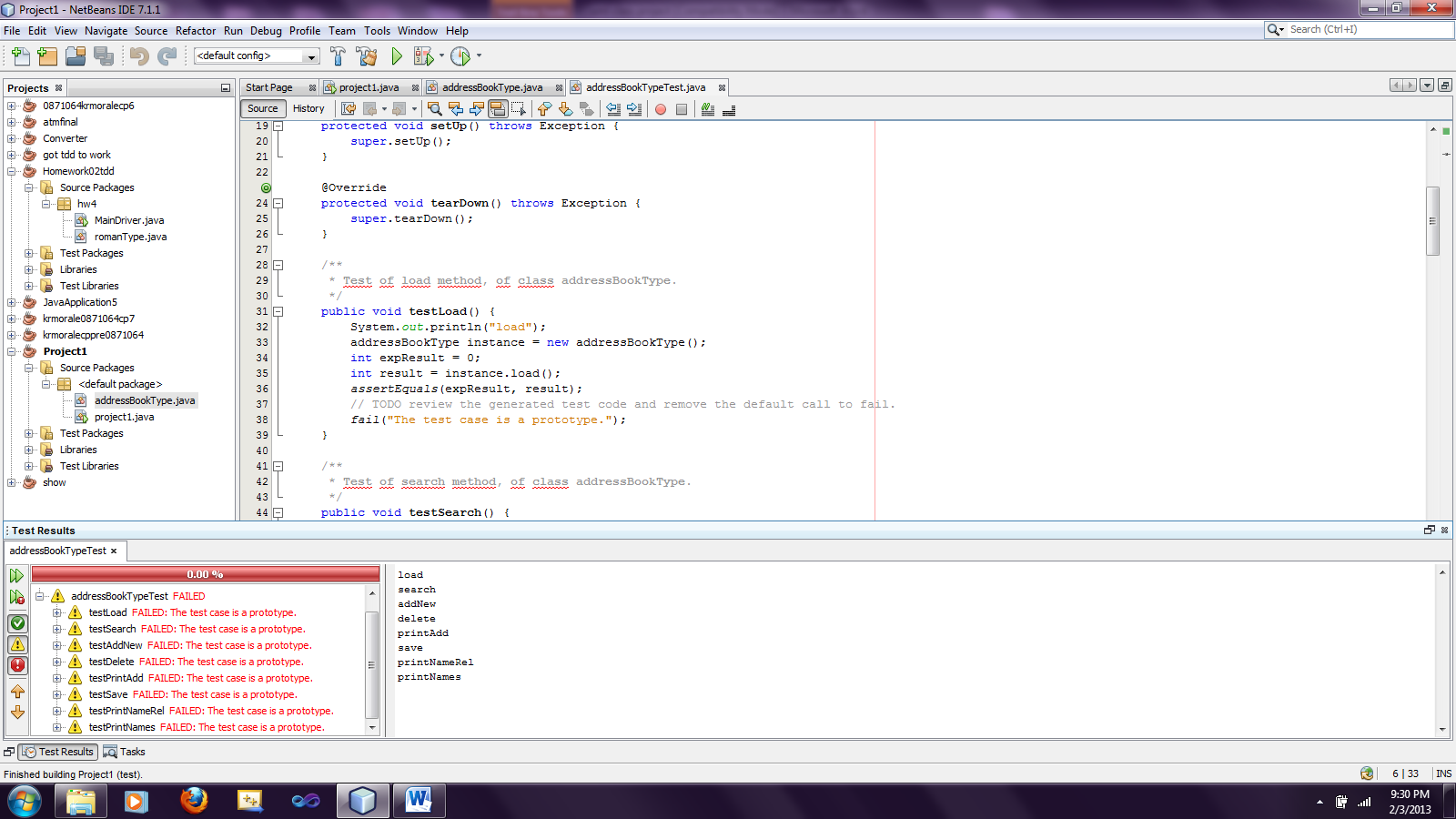
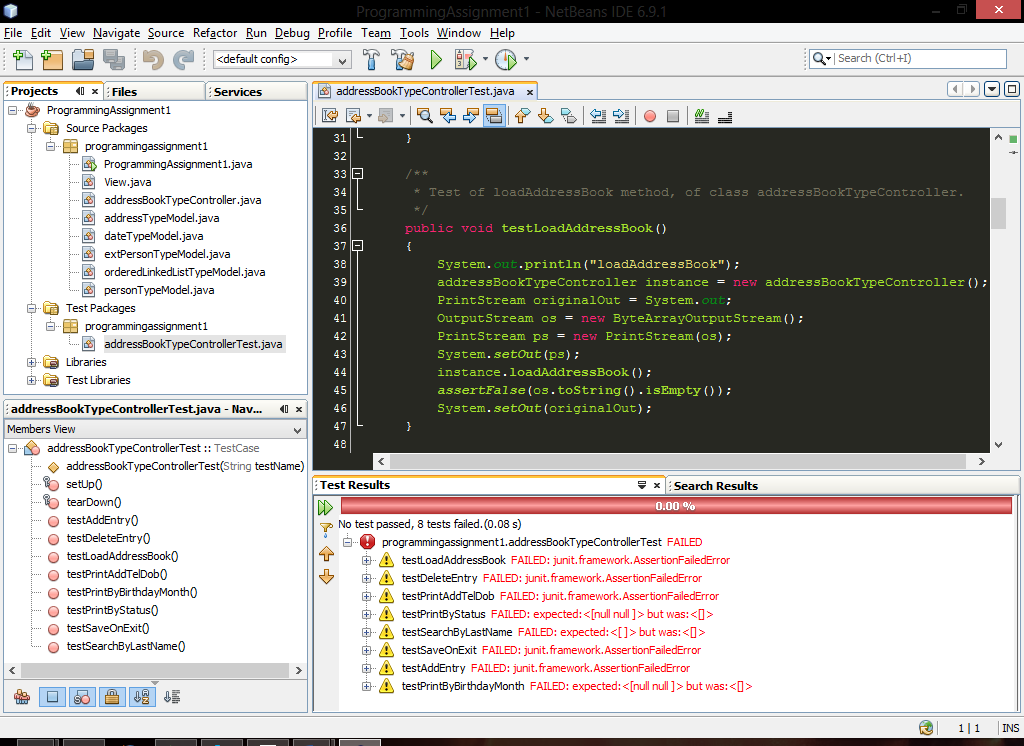
**USE TDD Testing** Definitions**:**

NOTE: Think of “Right Click” on the UML Class DIAGRAM “blue print” to “auto” generate the “house” skeleton classes needed for TDD in JAVA/C#/PYTHON/ RUBY/PHP.

**- Step 1** is to take a screenshot of a method. Lots of you did not scroll down enough to see any test cases thus.

TDD using JUnit (JAVA) will autogenerate test cases for the class you specify, in your case the **AddressBookType** Class that will have 8 **methods**. At this stage you are to write the code for the test cases but not have code for the **AddressBookType** Class 8 **methods (they will only be skeletons).**

NOTE: JUnit has a default fail statement. You must remove it or replace it with your code for that test case, just as presented in class.

You must write the code and the tests fail because there test case fails and not because of the fail statement.

**- Step 2** is not to take a screenshot but to copy and paste AddressBookType.java Class which will contain the methods skeletons and it will have your 4 lines at the beginning of the .java file.

**- Step 3** is not to take a screenshot but to copy and paste AddressBookTypeTest.java Class which will contain the test methods coded and it will have your 4 lines at the beginning of the .java file.