Problem 1: Page Counter

The unusual behavior occurs in the Menu Controller Class. In order to trigger this behavior the user has to go through the Menu Bar and select “Go To”.

After that if the user selects a number grater then the number of slides the counter will be out of range.

The way that it should behave is to not allow the user to go over or under the number of slides that the presentation has.

In order to tackle this issue a check has been made in order to verify the number input it.

Problem 2: Creating fonts in SlideViewerComponent

The issue relies in SlideViewerComponent class. The class is also responsible for creating a font which is used for painting.

The solution for this issue is to implement an interface which is responsible for creating the font. This way we avoid having to many responsibilities in SlideViewerComponent class.

Problem 3: Incorrect execution of the steps in Presentation

The issue occurs in the Presentation class. When setting up a slide a check is being made rather if the number is different then null or not

However, the order of the check is quite wrong because the currentSlide gets the number without even checking if it’s null or not.

The solution to this issue is to make the check and then assign the value of the currentSlide after checking if the value was null or not.

Problem 5: AboutBox long method code smell

In this class there is one method which returns a string that is going to be displayed on the frame. However, this is done in a inefficient way due to the fact of long lines of codes and zero maintainability.

This is done by concatenating strings in order to form the text which also influences the executing time.

A solution for this issue would be the use of String Builders. It is efficient due to the fact that each string added can be manipulated later in project, without having to create a new object.

Another argument would be due to the fact that String Builders are much more efficient in terms of time executing being much more faster then using the normal string concatenation.

Problem 6: Xml accessor

The issue occurs in the XML accessor class. Here we can observe that his class is an abstract class but with no particular fields. It has an empty constructor and it only has methods.

With that having in mind we should turn the Xml Acessor into an interface. This will only hold methods and for the maintainability it can be always implemented in a class for feature development or bug fixing.

By having it as an interface there is no need to instantiate the class in order to access it’s methods.

Problem 7: Slide Viewer Component and Presentation double relationship

The main problem is the double relationship between the SlideViewerComponent class and Presentation. Both of them contain each other which results to a double linked relationship.

A possible solution for this would be to make one of the classes god class , in our case SlideViwerComponent and make it responsible for manipulating a presentation.

This way the double linked relationship is eliminated. It is safe to say this is not the optimal solution considering that our SlideViewerComponent class now is a god class and it has a lot of responsibility, but our double linked relationship issue has been resolved