OOD Report

Arian Atapour

In the following table the issue, fix and which part of the software is fixed is explained.

|  |  |  |
| --- | --- | --- |
| **Issue** | **Fix** | **Part fixed** |
| 1 – Go to, can go over the limit of the presentation slide wise and will stay stuck on the slide selected previously. Recreate the error by inputting a bigger number than the slide counter limit. | If the number of the Go to input dialog box is over the total size of slides inside the presentation (getSize() method) then display the latest slide. This is done inside the MenuController class where the Go To menu item resides in. More accurately the fix is inside the actionPerfomed() method, inside the if statement.  This bug fix is the most logical one because it does not affect the overall look and feel of the app and it also improves its functionality, reliability and usability. | Functionality  Reliability and Usability |
| 2 – By using Go to the slide counter goes over the limit. To recreate this error just input in the Go to input box a number bigger than the limit of the slides. You should get for example 6 out of 3 if you inputted 6 as a number. | If the number of the Go to input dialog box is over the total size of slides (getSize() method) then set the slide counter selected on the left to the total size since it displays the last slide. So, if we have a presentation with 3 slides and we input the number 6, the last slide is going to appear (as seen in fix 1) and the counter will display Slide 3 of 3. This is done inside the MenuController class where the Go To menu item resides in. More accurately the fix is inside the actionPerfomed() method, inside the if statement.  This bug fix is the most logical one because it does not affect the overall look and feel of the app and it also improves its functionality, reliability and usability. | Functionality  Reliability and Usability |
| 3 – Too much text inside one method show() (About box text) | Breaking down the text into a separate method called text() and the utilization of StringBuilder.  This bug fix is the most logical one because it does not affect the overall look and feel of the app and it also improves its functionality and adaptability as well as testability. | Functionality  Adaptability  Testability |
| 4 – Unused variables (dead code) found in the Accessor class (DEMO\_NAME and DEFAULT\_EXTENSION never used). | Removed dead code which was never used.  This bug fix is the most logical one because it does not affect the overall look and feel of the app and it also improves its functionality. | Functionality |
| 5 – Unused constructor (dead code) found in the BitmapItem class (BitmapItem constructor never used). | Removed dead code which was never used.  This bug fix is the most logical one because it does not affect the overall look and feel of the app and it also improves its functionality. | Functionality |
| 6 – Unused method (dead code) found in the Slide class (getSlideItem() method never used). | Removed dead code which was never used.  This bug fix is the most logical one because it does not affect the overall look and feel of the app and it also improves its functionality. | Functionality |
| 7 – Unused method (dead code) found in the SlideItem class (SlideItem constructor never used). | Removed dead code which was never used.  This bug fix is the most logical one because it does not affect the overall look and feel of the app and it also improves its functionality. | Functionality |
| 8 – Unused constructor (dead code) found in the TextItem class (TextItem constructor never used). | Removed dead code which was never used.  This bug fix is the most logical one because it does not affect the overall look and feel of the app and it also improves its functionality. | Functionality |
| 9 – Unused variables (dead code) found in the XMLAccessor class (DEFAULT\_API\_TO\_USE never used). | Removed dead code which was never used.  This bug fix is the most logical one because it does not affect the overall look and feel of the app and it also improves its functionality. | Functionality |
| 10 – imageName variable from BitmapItem class is not final. | Made the imageName variable from BitMapItem final since no further value overriding is required.  This bug fix is the most logical one because it does not affect the overall look and feel of the app and it also improves its functionality. | Functionality |
| 11 – presentation variable from KeyController class is not final. | Made the presentation variable from KeyController class final since no further value overriding is required.  This bug fix is the most logical one because it does not affect the overall look and feel of the app and it also improves its functionality. | Functionality |
| 12 – Switch statement from keyPressed method taking too many lines of code from KeyController class | Made switch statement inline to save code lines.  This bug fix is the most logical one because it does not affect the overall look and feel of the app and it also improves its functionality. | Functionality |
|  |  |  |