Change request log

1   Team

Team Name: Deep Mind
Team Members and Roles:

Ujwal Srinivasa- Concept location, Impact analysis, actualization, Reverse engineering and validation

Nisha Rani - Concept Location and Reverse engineering.

2   Change Request

#ps2
Description: The Merge Module throws an error when you try to merge pages that intersect. For eg: if you specify the range as 1-4,2. The range 1-4 contains the page 2, and if you want to merge 2 again after 4, PDFsam will throw an error saying the input parameters are invalid. This change request will fix this issue and allow intersection while merging.

3   Concept Location

Step #	Description	Rationale
1	We ran the application PDFsam.	
2	We interacted with the merge feature of the PDFsam	We had to test the issue of intersection in merge. We did get an exception when testing the issue as stated in the description.
3	We navigated into the folder within the PDFmerge that contains all the related java files.	To Begin Concept location
4	We wanted to know where the user input is considered so we began checking the mergeparameterbuilder.java file	
5	We examined the class mergeparameterbuilder	To check where the input for range is considered.
6	We found a method called addinput, by using the search tool in IDE.	
7	We then ran Debugger tool in Eclipse.	To check how the input is considered.
8	Upon examination, we could see that a "set" data structure has been used to take in the input	Reason why intersection not happening.
9	Upon Debugging we found that this class mergeparameterbuilder is responsible for the issue, and we marked the class as "located"	We found that this class needs to be modified.

Time spent (in minutes): 180

4   Impact Analysis

Step #	Description	Rationale
1	The change is made in the method addinput in the class MergeParameterBuilder.	The function/method responsible for taking range and page numbers as input .
2	This method is used in the class MergeParameterBuilder.	
3	This method is only used in the class MergeParameterBuilder. This method is not used anywhere outside of the class. The Class MergeParameterBuilder extends to an abstract class, and implements parameters from the class SingleOutputTaskParametersBuilder. .	None of the extended or inherited class use the addinput method.
4	There is no impact by this change.	

Time spent (in minutes): 40

5   Actualization

Step #	Description	Rationale
1	There is only one method within the class mergeparameterbuilder that takes a range or page numbers as input.	
2	The void addinput method needs to be modified.	The method responsible for taking the input. The addinput function used to take input as "set" due to which only a single occurrence of input was allowed. This is why intersection of the same page was not allowed when merging.
3	The addinput method is modified to take and store input in an array and not set.	By doing this we can store many occurrences of the same element.
4	Two For loops are implemented.	One for loop is used to loop for the different values mentioned in the input(ranges,page numbers..etc)

		Another for loop to loop within the ranges and page numbers, and push all occurrences of the page numbers into the array list.This way the array will allow intersection. Now this change allows merging with intersection.

Time spent (in minutes): 120

x

6 Validation

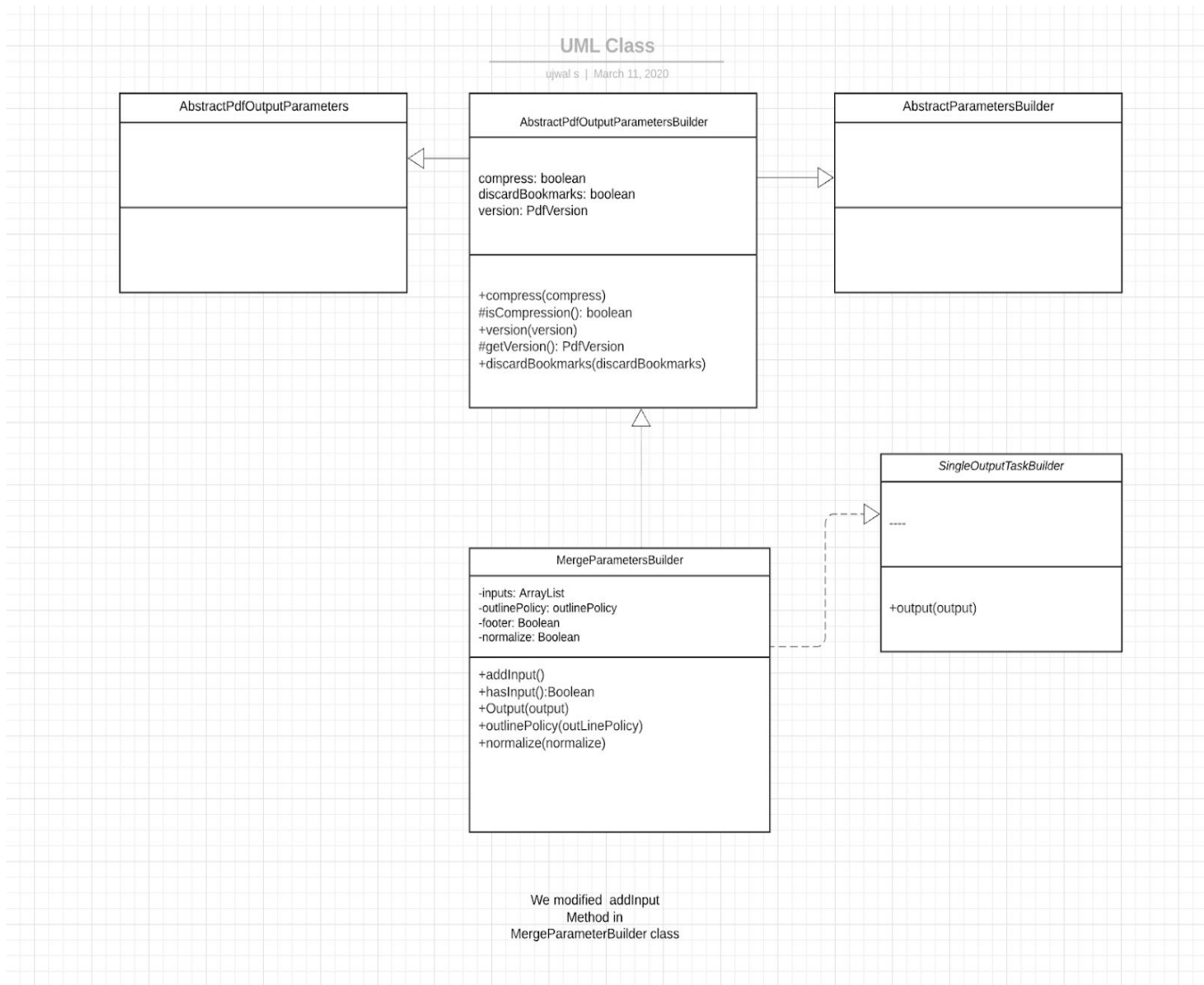| Step #	Description	Rationale
1	Manual Testing: -Build and Run Test case 1: Input: 1-25,4 Output: a pdf document with 1-25 pages with page 4 appended in the end	The test passed.no exception raised
2	Manual Testing: -Build and Run Test case 2: Input: 1-25,4-20,9,11 Output: a pdf document with all the pages from 1-25 pages+4-20pages+9+11	The Test case passed.no exception raised
3	Manual Testing: -Build and Run Test case 3: Input: Document 1- 40-45,41 Document 2- 1-10,5 Output: a pdf document with all the pages 40-45+41+1-10+5	The Test case passed.no exception raised
4	Manual Testing: -Build and Run Test case 4: Input: Document 1- 30,4,15,6 Document 2- 5-8,5,8 Output: a pdf document with all the pages 1-30+4+15+6+5-8+5+8	The Test case passed.no exception raised

Time spent (in minutes): 45

7 Timing

Summarize the time spent on each phase.

Phase Name	Time (in minutes)

Concept location	180
Impact Analysis	40
Prefactoring	-
Actualization	120
Postfactoring	100
Verification	45
Total	485

8 Reverse engineering

UML Class
ujwal s | March 11, 2020

AbstractPdfOutputParameters

AbstractPdfOutputParametersBuilder

compress: boolean
discardBookmarks: boolean
version: PdfVersion

+compress(compress)
#isCompression(): boolean
+version(version)
#getVersion(): PdfVersion
+discardBookmarks(discardBookmarks)

AbstractParametersBuilder

SingleOutputTaskBuilder

+output(output)

MergeParametersBuilder

-inputs: ArrayList
-outlinePolicy: outlinePolicy
-footer: Boolean
-normalize: Boolean

+addInput()
+hasInput():Boolean
+Output(output)
+outlinePolicy(outLinePolicy)
+normalize(normalize)

We modified addInput
Method in
MergeParameterBuilder class

9   Conclusions

- The concept location was very challenging for this change request. The debugger tool and the search tool in Eclipse IDE was very useful. The debug and trace method helped us find the right class that needed modification. Impact analysis was easy to determine, as the change made in the method of the class RotateParameterBuilder did affect any other modules.
- The change made is within the else statement of the method, just to consider both even/odd choice and the page range.
- Testing was done by various unique combinations of inputs and the change passed all cases.
- class and method change:
    - o   The addInput Method in the class MergeParameterBuilder-
    - o   File location- pdfsam-merge/src/main/java/org.pdfsam.merge/MergeParametersBuilder.java