

## Table of Contents

PDFSam – v4.0.5 .....	2
Code Structure.....	2
Directories .....	<b>Error! Bookmark not defined.</b>
Features Request.....	8
Application/Functional Bug .....	9
Bug 1: Module: Split by bookmarks.....	9
Bug 2: Page Extraction module: .....	11
JEdit .....	14
Code Structure.....	14
Features Request.....	14
Application/Functional Bug .....	14
Bug 1: File Not Found Exception.....	14
Bug 1: GUI .....	17
Other Miscellaneous bugs found while going over the code and using the applications.....	18

## How to Start

Ideas:

1. Plan to understand the code (apart from just reading it!) via the test cases and the documentation.
2. Plan to understand the build requirement of a package from just its pom.xml
3. Build class(UML) diagram of each package(very long job – lots of subclasses)

## PDFSam – v4.0.5

### Introduction

PDFSam is an open source software developed for manipulating pdf files. It offers several features allowing the user to merge, split, extract, mix and even rotate pages in a given file. It does not offer an editor, or even a pdf viewer as part of its functionality, those being offered in its paid version 'PDFsam Enhanced'. Several other premium features such as pdf signing, and attachments are offered in the Enhanced version. We built PDFSam from source on Eclipse as specified in instructions present in A1.pdf.

### Version

V4.0.5

### Code Structure

15 directories:

Reactor Build Order:	
PDFsam	[pom]
PDFsam i18n	[jar]
PDFsam core	[jar]
PDFsam JavaFx views	[jar]
PDFsam service	[jar]
PDFsam merge module	[jar]
PDFsam simple split module	[jar]
PDFsam split by size module	[jar]
PDFsam split by bookmarks module	[jar]
PDFsam alternate mix module	[jar]
PDFsam rotate module	[jar]
PDFsam extract module	[jar]
PDFsam desktop client application	[jar]
PDFsam Basic Edition application	[jar]
PDFsam docs	[pom]

## Modules

1. pdfsam-docs: among other files this folder contains
  - a. icons
  - b. A font package
  - c. A code file template
  - d. A latest news/changes document in the form of an html file.
  - e. It is stated in the pdfsam-parent folder/contributing.md that

*Code format and templates for the Eclipse IDE are included in the pdfsam-docs module as well as the checkstyle plugin configuration, use them.*

- f. Dependency graph for pdfsam-docs:

```
org.pdfsam:pdfsam-docs:pom:4.0.5
+- junit:junit:jar:4.12:test
| \- org.hamcrest:hamcrest-core:jar:1.3:test
+- ch.qos.logback:logback-classic:jar:1.2.3:test
| \- ch.qos.logback:logback-core:jar:1.2.3:test
+- org.slf4j:slf4j-api:jar:1.7.28:compile
+- org.mockito:mockito-core:jar:3.0.0:test
| +- net.bytebuddy:byte-buddy:jar:1.9.10:test
| +- net.bytebuddy:byte-buddy-agent:jar:1.9.10:test
| \- org.objenesis:objenesis:jar:2.6:test
+- org.hamcrest:hamcrest-all:jar:1.3:test
\-- com.google.code.tempus-fugit:tempus-fugit:jar:1.1:test
```

2. pdfsam-fx and pdfsam-gui: Contain code and classes for:
  - a. GUI for the whole application (Dashboard, notification window, log window)
  - b. custom animations: Affects that are invoked during actions (e.g. Clicking the what's new icon in the top right of the window shifts the content left to make space for a column in which a custom window appears.
  - c. custom controls(ones not generally found in swing or javafx) (e.g. ).
  - d. Dependency graph for pdfsam-fx:

```

org.pdfsam:pdfsam-fx:jar:4.0.5
+- org.apache.commons:commons-lang3:jar:3.9:compile
+- commons-io:commons-io:jar:2.6:compile
+- org.sejda:sejda-model:jar:3.2.85:compile
| \- javax.validation:validation-api:jar:1.0.0.GA:compile
+- org.sejda:eventstudio:jar:1.0.6:compile
+- de.jensd:fontawesomefx-fontawesome:jar:4.7.0-9.1.2:compile
| \- de.jensd:fontawesomefx-commons:jar:9.1.2:compile
+- de.jensd:fontawesomefx-materialdesignfont:jar:2.0.26-9.1.2:compile
+- org.pdfsam:pdfsam-i18n:jar:4.0.5:compile
| \- com.googlecode.gettext-commons:gettext-commons:jar:0.9.8:compile
+- org.pdfsam:pdfsam-core:jar:4.0.5:compile
| \- org.sejda:sejda-conversion:jar:3.2.85:compile
+- ch.qos.logback:logback-classic:jar:1.2.3:compile
| \- ch.qos.logback:logback-core:jar:1.2.3:compile
+- org.sejda:sejda-injector:jar:1.0.2:compile
+- javax.inject:javax.inject:jar:1:compile
+- org.openjfx:javafx-controls:jar:13.0.1:compile
| +- org.openjfx:javafx-controls:jar:win:13.0.1:compile
| \- org.openjfx:javafx-graphics:jar:13.0.1:compile
|   \- org.openjfx:javafx-graphics:jar:win:13.0.1:compile
+- jdepend:jdepend:jar:2.9.1:test
+- org.testfx:testfx-core:jar:4.0.15-alpha:test
| +- org.hamcrest:hamcrest-core:jar:1.3:test
| \- org.assertj:assertj-core:jar:3.11.1:test
+- org.testfx:testfx-junit:jar:4.0.15-alpha:test
+- org.testfx:openjfx-monocle:jar:jdk-12.0.1+2:test
| \- org.openjfx:javafx-base:jar:12.0.1:compile
|   \- org.openjfx:javafx-base:jar:win:12.0.1:compile
+- org.openjfx:javafx-swing:jar:13.0.1:test
| \- org.openjfx:javafx-swing:jar:win:13.0.1:test
+- org.pdfsam:pdfsam-core:test-jar:tests:4.0.5:test
+- junit:junit:jar:4.12:test
+- org.slf4j:slf4j-api:jar:1.7.28:compile
+- org.mockito:mockito-core:jar:3.0.0:test
| +- net.bytebuddy:byte-buddy:jar:1.9.10:test
| +- net.bytebuddy:byte-buddy-agent:jar:1.9.10:test
| \- org.objenesis:objenesis:jar:2.6:test
+- org.hamcrest:hamcrest-all:jar:1.3:test
\-- com.google.code.tempus-fugit:tempus-fugit:jar:1.1:test

```

3. pdfsam-alternate-mix: Contains the code containing the functionality and tests for the same for the alternate mix module, with a few custom icons and tests for the code.
  - a. Dependency graph for pdfsam-alternate-mix:

```

org.pdfsam:pdfsam-alternate-mix:jar:4.0.5
+- org.apache.commons:commons-lang3:jar:3.9:compile
+- commons-io:commons-io:jar:2.6:compile
+- org.sejda:sejda-model:jar:3.2.85:compile
| \- javax.validation:validation-api:jar:1.0.0.GA:compile
+- org.pdfsam:pdfsam-fx:jar:4.0.5:compile
| +- org.sejda:eventstudio:jar:1.0.6:compile
| +- de.jensd:fontawesomefx-fontawesome:jar:4.7.0-9.1.2:compile
| | \- de.jensd:fontawesomefx-commons:jar:9.1.2:compile
| +- de.jensd:fontawesomefx-materialdesignfont:jar:2.0.26-9.1.2:compile
| \- org.pdfsam:pdfsam-i18n:jar:4.0.5:compile
|   \- com.googlecode.gettext-commons:gettext-commons:jar:0.9.8:compile
+- org.pdfsam:pdfsam-core:jar:4.0.5:compile
| \- org.sejda:sejda-conversion:jar:3.2.85:compile
+- org.sejda:sejda-injector:jar:1.0.2:compile
+- javax.inject:javax.inject:jar:1:compile
+- org.openjfx:javafx-controls:jar:13.0.1:compile
| +- org.openjfx:javafx-controls:jar:win:13.0.1:compile
| \- org.openjfx:javafx-graphics:jar:13.0.1:compile
|   +- org.openjfx:javafx-graphics:jar:win:13.0.1:compile
|   \- org.openjfx:javafx-base:jar:13.0.1:compile
|     \- org.openjfx:javafx-base:jar:win:13.0.1:compile
+- org.pdfsam:pdfsam-core:test-jar:tests:4.0.5:test
+- jdepend:jdepend:jar:2.9.1:test
+- org.openjfx:javafx-swing:jar:13.0.1:test
| \- org.openjfx:javafx-swing:jar:win:13.0.1:test
+- junit:junit:jar:4.12:test
| \- org.hamcrest:hamcrest-core:jar:1.3:test
+- ch.qos.logback:logback-classic:jar:1.2.3:test
| \- ch.qos.logback:logback-core:jar:1.2.3:test
+- org.slf4j:slf4j-api:jar:1.7.28:compile
+- org.mockito:mockito-core:jar:3.0.0:test
| +- net.bytebuddy:byte-buddy:jar:1.9.10:test
| +- net.bytebuddy:byte-buddy-agent:jar:1.9.10:test
| \- org.objenesis:objenesis:jar:2.6:test
+- org.hamcrest:hamcrest-all:jar:1.3:test
\- com.google.code.tempus-fugit:tempus-fugit:jar:1.1:test

```

4. pdfsam-i18n: This module provides services and string translations to handle different locales of language in the application.

a. Dependency tree for pdfsam-i18n:

```

org.pdfsam:pdfsam-i18n:jar:4.0.5
+- org.apache.commons:commons-lang3:jar:3.9:compile
+- org.sejda:eventstudio:jar:1.0.6:compile
+- com.googlecode.gettext-commons:gettext-commons:jar:0.9.8:compile
+- junit:junit:jar:4.12:test
| \- org.hamcrest:hamcrest-core:jar:1.3:test
+- ch.qos.logback:logback-classic:jar:1.2.3:test
| \- ch.qos.logback:logback-core:jar:1.2.3:test
+- org.slf4j:slf4j-api:jar:1.7.28:compile
+- org.mockito:mockito-core:jar:3.0.0:test
| +- net.bytebuddy:byte-buddy:jar:1.9.10:test
| +- net.bytebuddy:byte-buddy-agent:jar:1.9.10:test
| \- org.objenesis:objenesis:jar:2.6:test
+- org.hamcrest:hamcrest-all:jar:1.3:test
\- com.google.code.tempus-fugit:tempus-fugit:jar:1.1:test

```



5. Pdfsam-merge: This module contains the following:

- a. Code and classes for the merge-pdf functionality.
- b. A custom icon
- c. Tests to be run for this functionality
- d. Dependency graph for pdfsam-merge:

```
org.pdfsam:pdfsam-merge:jar:4.0.5
+- org.apache.commons:commons-lang3:jar:3.9:compile
+- commons-io:commons-io:jar:2.6:compile
+- org.sejda:sejda-model:jar:3.2.85:compile
| \- javax.validation:validation-api:jar:1.0.0.GA:compile
+- org.pdfsam:pdfsam-fx:jar:4.0.5:compile
| +- org.sejda:eventstudio:jar:1.0.6:compile
| +- de.jensd:fontawesomefx-fontawesome:jar:4.7.0-9.1.2:compile
| | \- de.jensd:fontawesomefx-commons:jar:9.1.2:compile
| +- de.jensd:fontawesomefx-materialdesignfont:jar:2.0.26-9.1.2:compile
| \- org.pdfsam:pdfsam-i18n:jar:4.0.5:compile
|   \- com.googlecode.gettext-commons:gettext-commons:jar:0.9.8:compile
+- org.pdfsam:pdfsam-core:jar:4.0.5:compile
| \- org.sejda:sejda-conversion:jar:3.2.85:compile
+- org.sejda:sejda-injector:jar:1.0.2:compile
+- javax.inject:javax.inject:jar:1:compile
+- org.openjfx:javafx-controls:jar:13.0.1:compile
| +- org.openjfx:javafx-controls:jar:win:13.0.1:compile
| \- org.openjfx:javafx-graphics:jar:13.0.1:compile
|   \- org.openjfx:javafx-graphics:jar:win:13.0.1:compile
+- org.pdfsam:pdfsam-core:test-jar:tests:4.0.5:test
+- jdepend:jdepend:jar:2.9.1:test
+- org.testfx:testfx-core:jar:4.0.15-alpha:test
| +- org.hamcrest:hamcrest-core:jar:1.3:test
| \- org.assertj:assertj-core:jar:3.11.1:test
+- org.testfx:testfx-junit:jar:4.0.15-alpha:test
+- org.testfx:openjfx-monocle:jar:jdk-12.0.1+2:test
| \- org.openjfx:javafx-base:jar:12.0.1:compile
|   \- org.openjfx:javafx-base:jar:win:12.0.1:compile
+- org.openjfx:javafx-swing:jar:13.0.1:test
| \- org.openjfx:javafx-swing:jar:win:13.0.1:test
+- junit:junit:jar:4.12:test
+- ch.qos.logback:logback-classic:jar:1.2.3:test
| \- ch.qos.logback:logback-core:jar:1.2.3:test
+- org.slf4j:slf4j-api:jar:1.7.28:compile
+- org.mockito:mockito-core:jar:3.0.0:test
| +- net.bytebuddy:byte-buddy:jar:1.9.10:test
| +- net.bytebuddy:byte-buddy-agent:jar:1.9.10:test
| \- org.objenesis:objenesis:jar:2.6:test
+- org.hamcrest:hamcrest-all:jar:1.3:test
\- com.google.code.tempus-fugit:tempus-fugit:jar:1.1:test
```

6. Pdfsam-simple-split, pdfsam-split-by-bookmarks and pdfsam-split-by-size: Grouping all these similar modules, we can see that they handle the functionalities of splitting the pdf according to certain features, such as:

- a. size(MB,KB),
- b. bookmarks(with different levels of bookmarks or certain fixed bookmarks, like 'Section' included)

- c. regular splitting by pages (page numbers, odd/even pages and splitting after fixed intervals.)

Interestingly, we can see in the dependencies that they don't explicitly need the other modules to be built.

- d. Dependency graph for pdfsam-simple-split:

```
org.pdfsam:pdfsam-simple-split:jar:4.0.5
+- org.apache.commons:commons-lang3:jar:3.9:compile
+- commons-io:commons-io:jar:2.6:compile
+- org.sejda:sejda-model:jar:3.2.85:compile
| \- javax.validation:validation-api:jar:1.0.0.GA:compile
+- org.pdfsam:pdfsam-fx:jar:4.0.5:compile
| +- org.sejda:eventstudio:jar:1.0.6:compile
| +- de.jensd:fontawesomefx-fontawesome:jar:4.7.0-9.1.2:compile
| | \- de.jensd:fontawesomefx-commons:jar:9.1.2:compile
| +- de.jensd:fontawesomefx-materialdesignfont:jar:2.0.26-9.1.2:compile
| \- org.pdfsam:pdfsam-i18n:jar:4.0.5:compile
|   \- com.googlecode.gettext-commons:gettext-commons:jar:0.9.8:compile
+- org.pdfsam:pdfsam-core:jar:4.0.5:compile
| \- org.sejda:sejda-conversion:jar:3.2.85:compile
+- org.sejda:sejda-injector:jar:1.0.2:compile
+- javax.inject:javax.inject:jar:1:compile
+- org.openjfx:javafx-controls:jar:13.0.1:compile
| +- org.openjfx:javafx-controls:jar:win:13.0.1:compile
| \- org.openjfx:javafx-graphics:jar:13.0.1:compile
|   \- org.openjfx:javafx-graphics:jar:win:13.0.1:compile
+- org.pdfsam:pdfsam-core:test-jar:tests:4.0.5:test
+- jdepend:jdepend:jar:2.9.1:test
+- org.testfx:testfx-core:jar:4.0.15-alpha:test
| +- org.hamcrest:hamcrest-core:jar:1.3:test
| \- org.assertj:assertj-core:jar:3.11.1:test
+- org.testfx:testfx-junit:jar:4.0.15-alpha:test
+- org.testfx:openjfx-monocle:jar:jdk-12.0.1+2:test
| \- org.openjfx:javafx-base:jar:12.0.1:compile
|   \- org.openjfx:javafx-base:jar:win:12.0.1:compile
+- junit:junit:jar:4.12:test
+- ch.qos.logback:logback-classic:jar:1.2.3:test
| \- ch.qos.logback:logback-core:jar:1.2.3:test
+- org.slf4j:slf4j-api:jar:1.7.28:compile
+- org.mockito:mockito-core:jar:3.0.0:test
| +- net.bytebuddy:byte-buddy:jar:1.9.10:test
| +- net.bytebuddy:byte-buddy-agent:jar:1.9.10:test
| \- org.objenesis:objenesis:jar:2.6:test
+- org.hamcrest:hamcrest-all:jar:1.3:test
\- com.google.code.tempus-fugit:tempus-fugit:jar:1.1:test
```

- e. Dependency graph for pdfsam-split-size:

```

org.pdfsam:pdfsam-split-by-size:jar:4.0.5
+- org.apache.commons:commons-lang3:jar:3.9:compile
+- commons-io:commons-io:jar:2.6:compile
+- org.sejda:sejda-model:jar:3.2.85:compile
| \- javax.validation:validation-api:jar:1.0.0.GA:compile
+- org.pdfsam:pdfsam-fx:jar:4.0.5:compile
| +- org.sejda:eventstudio:jar:1.0.6:compile
| +- de.jensd:fontawesomefx-fontawesome:jar:4.7.0-9.1.2:compile
| | \- de.jensd:fontawesomefx-commons:jar:9.1.2:compile
| +- de.jensd:fontawesomefx-materialdesignfont:jar:2.0.26-9.1.2:compile
| \- org.pdfsam:pdfsam-i18n:jar:4.0.5:compile
|   \- com.googlecode.gettext-commons:gettext-commons:jar:0.9.8:compile
+- org.pdfsam:pdfsam-core:jar:4.0.5:compile
| \- org.sejda:sejda-conversion:jar:3.2.85:compile
+- org.sejda:sejda-injector:jar:1.0.2:compile
+- javax.inject:javax.inject:jar:1:compile
+- org.openjfx:javafx-controls:jar:13.0.1:compile
| +- org.openjfx:javafx-controls:jar:win:13.0.1:compile
| \- org.openjfx:javafx-graphics:jar:13.0.1:compile
|   \- org.openjfx:javafx-graphics:jar:win:13.0.1:compile
+- org.pdfsam:pdfsam-core:test-jar:tests:4.0.5:test
+- jdepend:jdepend:jar:2.9.1:test
+- org.testfx:testfx-core:jar:4.0.15-alpha:test
| +- org.hamcrest:hamcrest-core:jar:1.3:test
| \- org.assertj:assertj-core:jar:3.11.1:test
+- org.testfx:testfx-junit:jar:4.0.15-alpha:test
+- org.testfx:openjfx-monocle:jar:jdk-12.0.1+2:test
| \- org.openjfx:javafx-base:jar:12.0.1:compile
|   \- org.openjfx:javafx-base:jar:win:12.0.1:compile
+- org.openjfx:javafx-swing:jar:13.0.1:test
| \- org.openjfx:javafx-swing:jar:win:13.0.1:test
+- junit:junit:jar:4.12:test
+- ch.qos.logback:logback-classic:jar:1.2.3:test
| \- ch.qos.logback:logback-core:jar:1.2.3:test
+- org.slf4j:slf4j-api:jar:1.7.28:compile
+- org.mockito:mockito-core:jar:3.0.0:test
| +- net.bytebuddy:byte-buddy:jar:1.9.10:test
| +- net.bytebuddy:byte-buddy-agent:jar:1.9.10:test
| \- org.objenesis:objenesis:jar:2.6:test
+- org.hamcrest:hamcrest-all:jar:1.3:test
\- com.google.code.tempus-fugit:tempus-fugit:jar:1.1:test

```

7. [Pdfsam-merge](#)
8. [Pdfsam-rotate](#)
9. [Pdfsam-extract](#)
- 10.

## Features Request

1. Ability to find the table of contents and extract chapters. Use Case: If a user is reading a book or Ph D dissertation, they are really long (some dissertations can be up to 200 pages in length). Maybe the user does not want to print the whole thing out like a textbook, but instead read it one chapter at a time. In this case, maybe this functionality will be helpful.



2. Feature to rip images or tables from a given file. Use Case: If a user is reading a research paper and wants to extract the tables or images from a pdf in the correct formatting (tables) or proper resolution (image). Then having this functionality would be helpful.

## Application/Functional Bug

### Bug I: Module: Split by bookmarks

#### *Summary:*

When using the split by bookmarks module, running it on a .pdf file (with bookmarks) and entering a regular expression searching for a heading with at least one space in front of it. It fails silently, with just the status and an application log being dropped. Leaving me to guess why this is happening.

#### *Version:*

V4.0.5

#### *Steps to reproduce:*

1. Open the 'split by bookmarks' module.
2. Selected a pdf with any number of bookmarks
3. Enter the regex: `^. * +$`

#### *Expected Behavior:*

1. Message telling me that file does not include given regex
2. Message stating that it is an invalid regex.

#### *Observed Behavior:*

1. Failed status displayed above progress bar.
2. Application Logs (Log register) icon blinking.
3. Stack trace dumped in logs.
4. No other functionality

#### *Stack Trace:*

```
ERROR [21:38:32]: Task (org.sejda.impl.sambox.SplitByOutlineLevelTask@311fc502) execution failed.  
org.sejda.model.exception.TaskExecutionException: Unable to split, no page number given.  
at org.sejda.model.split.SplitPages.ensureIsValid(SplitPages.java:59)  
at org.sejda.model.split.PageDestinationsSplitPages.ensureIsValid(PageDestinationsSplitPages.java:44)  
at org.sejda.impl.sambox.component.split.AbstractPdfSplitter.split(AbstractPdfSplitter.java:71)  
at org.sejda.impl.sambox.SplitByOutlineLevelTask.execute(SplitByOutlineLevelTask.java:79)  
at org.sejda.impl.sambox.SplitByOutlineLevelTask.execute(SplitByOutlineLevelTask.java:46)  
at org.sejda.core.service.DefaultTaskExecutionService.actualExecution(DefaultTaskExecutionService.java:148)  
at org.sejda.core.service.DefaultTaskExecutionService.execute(DefaultTaskExecutionService.java:71)  
at org.sejda.core.service.DefaultTaskExecutionService.execute(DefaultTaskExecutionService.java:58)
```

at org.pdfsam.task.TaskExecutionController.lambda\$request\$0(TaskExecutionController.java:87)

at java.base/java.util.concurrent.ThreadPoolExecutor.runWorker(Unknown Source)

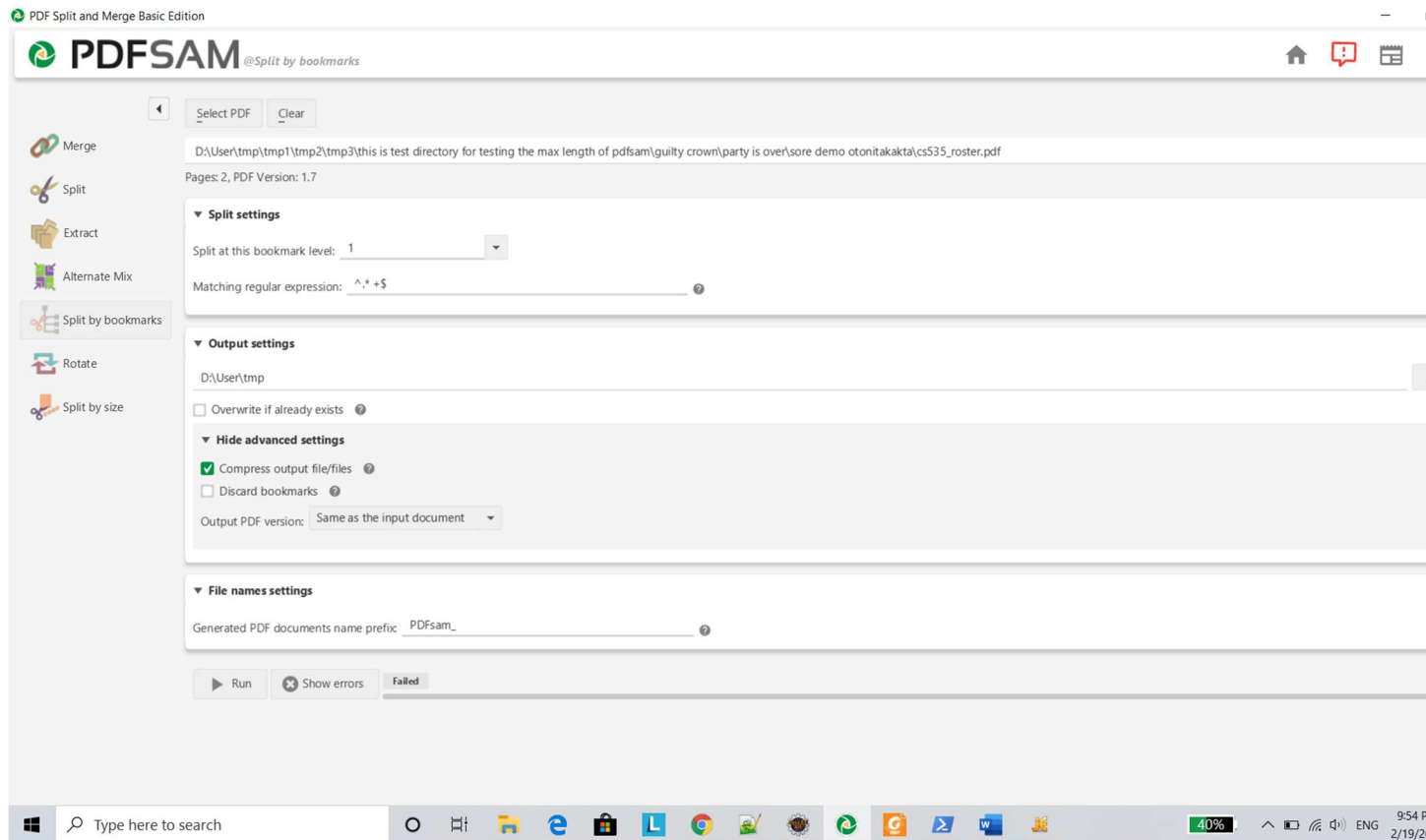
at java.base/java.util.concurrent.ThreadPoolExecutor\$Worker.run(Unknown Source)

at java.base/java.lang.Thread.run(Unknown Source)

## Screenshot

Log register

```
INFO [21:08:24]: Starting PDFsam
INFO [21:08:25]: Configuring Sejda 3.2.85
INFO [21:08:26]: Started in 1 second
INFO [21:08:27]: PDFsam 4.1.1 is available for download
INFO [16:52:12]: Workspace saved
INFO [21:35:31]: cs535_roster.pdf loaded
INFO [21:38:09]: Final_submitted_CS_533_Database_Temporality_Metrics(10).pdf loaded
INFO [21:38:12]: cs535_roster.pdf loaded
INFO [21:38:32]: Starting task (org.sejda.impl.sandbox.SplitByOutlineLevelTask@311fc502) execution.
ERROR [21:38:32]: Task (org.sejda.impl.sandbox.SplitByOutlineLevelTask@311fc502) execution failed.
org.sejda.model.exception.TaskExecutionException: Unable to split, no page number given.
    at org.sejda.model.split.SplitPages.ensureIsValid(SplitPages.java:59)
    at org.sejda.model.split.PageDestinationsSplitPages.ensureIsValid(PageDestinationsSplitPages.java:44)
    at org.sejda.impl.sandbox.component.split.AbstractPdfSplitter.split(AbstractPdfSplitter.java:71)
    at org.sejda.impl.sandbox.SplitByOutlineLevelTask.execute(SplitByOutlineLevelTask.java:79)
    at org.sejda.impl.sandbox.SplitByOutlineLevelTask.execute(SplitByOutlineLevelTask.java:46)
    at org.sejda.core.service.DefaultTaskExecutionService.actualExecution(DefaultTaskExecutionService.java:148)
    at org.sejda.core.service.DefaultTaskExecutionService.execute(DefaultTaskExecutionService.java:71)
    at org.sejda.core.service.DefaultTaskExecutionService.execute(DefaultTaskExecutionService.java:58)
    at org.pdfsam.task.TaskExecutionController.lambda$request$0(TaskExecutionController.java:87)
    at java.base/java.util.concurrent.ThreadPoolExecutor.runWorker(Unknown Source)
    at java.base/java.util.concurrent.ThreadPoolExecutor$Worker.run(Unknown Source)
    at java.base/java.lang.Thread.run(Unknown Source)
```



## Bug 2: Page Extraction module:

### Summary:

After adding a pdf file with only one page, I accidently typed in the page number to extract as 2 and the execution silently failed, dropping stack trace logs but I didn't exactly understand what was going on.

### Version:

V4.0.5

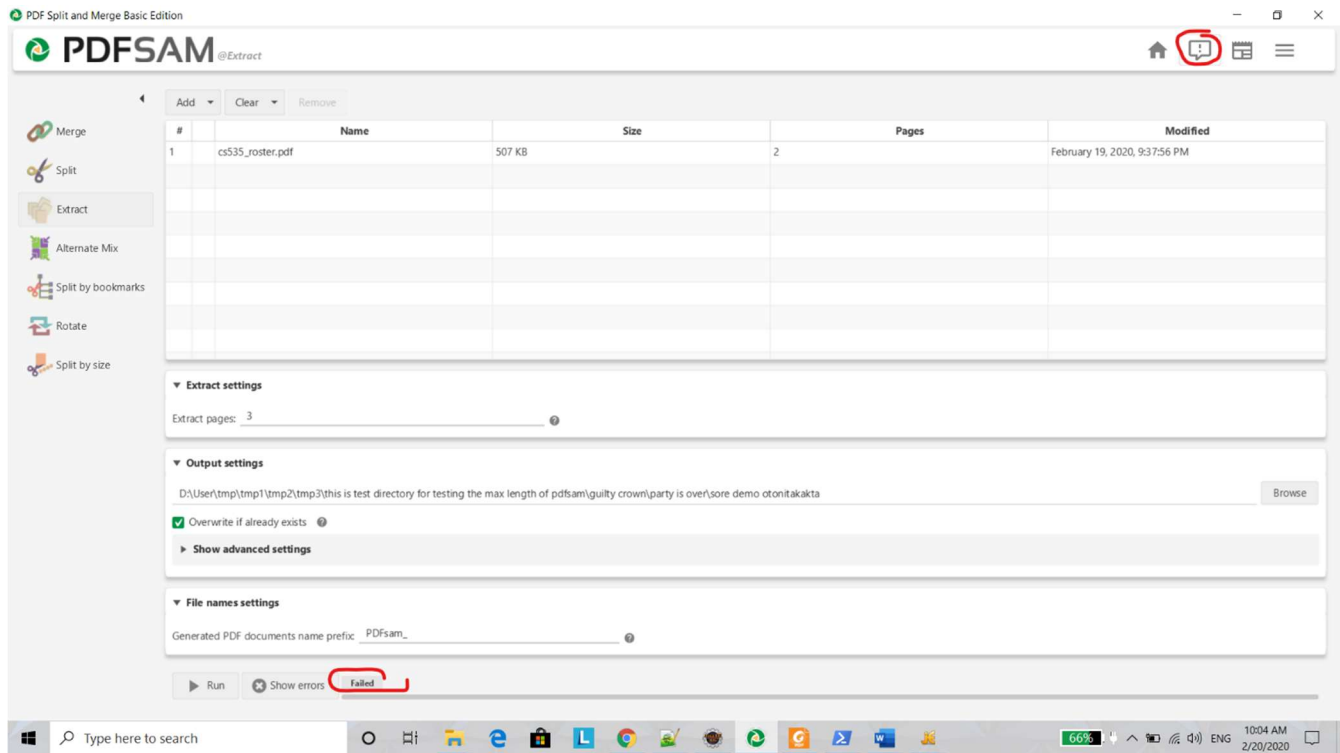
### Expected Behavior:

1. A message telling me that the page number that I have entered does not exist in the pdf.
2. OR a message telling me that the page number is invalid.

### Observed Behavior:

1. Failed status badge above the progress bar.
2. Application logs icon becomes highlighted.
3. Stack trace contains the message "No page selected for extraction"

## Screenshots





```

at org.sejda.model.split.PageDestinationsSplitPages.ensureIsValid(PageDestinationsSplitPages.java:44)
at org.sejda.impl.sambox.component.split.AbstractPdfSplitter.split(AbstractPdfSplitter.java:71)
at org.sejda.impl.sambox.SplitByOutlineLevelTask.execute(SplitByOutlineLevelTask.java:79)
at org.sejda.impl.sambox.SplitByOutlineLevelTask.execute(SplitByOutlineLevelTask.java:46)
at org.sejda.core.service.DefaultTaskExecutionService.actualExecution(DefaultTaskExecutionService.java:148)
at org.sejda.core.service.DefaultTaskExecutionService.execute(DefaultTaskExecutionService.java:71)
at org.sejda.core.service.DefaultTaskExecutionService.execute(DefaultTaskExecutionService.java:58)
at org.pdfsam.task.TaskExecutionController.lambda$request$0(TaskExecutionController.java:87)
at java.base/java.util.concurrent.ThreadPoolExecutor.runWorker(Unknown Source)
at java.base/java.util.concurrent.ThreadPoolExecutor$Worker.run(Unknown Source)
at java.base/java.lang.Thread.run(Unknown Source)
INFO [21:56:13]: cs535_roster.pdf loaded
INFO [21:56:35]: Starting task (org.sejda.impl.sambox.ExtractPagesTask@1297228c) execution.
INFO [21:56:35]: Task (org.sejda.impl.sambox.ExtractPagesTask@1297228c) executed in 0 seconds
INFO [21:56:40]: Starting task (org.sejda.impl.sambox.ExtractPagesTask@1235457b) execution.
INFO [21:56:40]: Task (org.sejda.impl.sambox.ExtractPagesTask@1235457b) executed in 0 seconds
INFO [21:56:43]: Starting task (org.sejda.impl.sambox.ExtractPagesTask@2ca4b0be) execution.
INFO [21:56:43]: Task (org.sejda.impl.sambox.ExtractPagesTask@2ca4b0be) executed in 0 seconds
INFO [21:57:31]: PDFsam_cs535_roster.pdf loaded
INFO [21:57:34]: Starting task (org.sejda.impl.sambox.ExtractPagesTask@509c98ba) execution.
ERROR [21:57:34]: Task (org.sejda.impl.sambox.ExtractPagesTask@509c98ba) execution failed.
org.sejda.model.exception.TaskExecutionException: No page has been selected for extraction.
    at org.sejda.impl.sambox.ExtractPagesTask.execute(ExtractPagesTask.java:86)
    at org.sejda.impl.sambox.ExtractPagesTask.execute(ExtractPagesTask.java:54)
    at org.sejda.core.service.DefaultTaskExecutionService.actualExecution(DefaultTaskExecutionService.java:148)
    at org.sejda.core.service.DefaultTaskExecutionService.execute(DefaultTaskExecutionService.java:71)
    at org.sejda.core.service.DefaultTaskExecutionService.execute(DefaultTaskExecutionService.java:58)
    at org.pdfsam.task.TaskExecutionController.lambda$request$0(TaskExecutionController.java:87)
    at java.base/java.util.concurrent.ThreadPoolExecutor.runWorker(Unknown Source)
    at java.base/java.util.concurrent.ThreadPoolExecutor$Worker.run(Unknown Source)
    at java.base/java.lang.Thread.run(Unknown Source)

```



Type here to search



38%



## JEdit

### Introduction

jEdit is a programmer's text editor, which was originally developed by Slava Pestov available under the GNU GPL v 2.0. It allows itself to be customized using macros written in BeanShell and has a flexible plugin architecture allowing its features to be extended by other programs. We built jEdit from source on Eclipse using Apache Ant as specified in instructions present in A1.pdf.

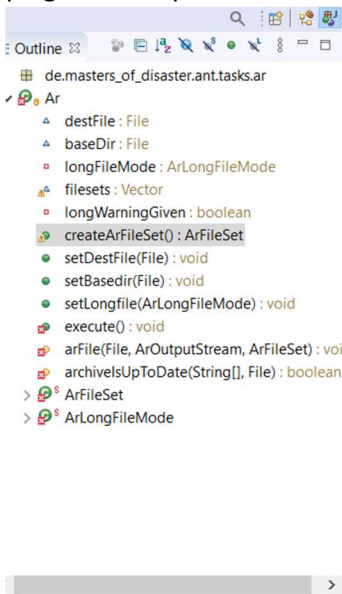
### Version

v5.5.0

### Code Structure

### Features Request

1. An outline for a given code file. As implemented in Eclipse below, It would be useful to scan a code file to detect its classes and methods. With jEdit calling itself a programmers text editor, it feels only natural it would have something similar to this as an inbuilt feature. I couldn't find any developed plugin that implements this feature either.



2.

### Application/Functional Bug

#### Bug I: File Not Found Exception

- *Summary*

The Help tab in Jedit has three options. One of these options is JEdit Help which providing documents for different purposes such as invoked document. When user clicks on these documents, the File not found

Exception message is shown. What expected is that the HTML files with the required details should be presented when these documents' links are invoked.

- *Steps to reproduce*
  - a. Click on "Help" tab.
  - b. Select "JEdit Help" or press F1.
  - c. Select one of the following documents
    - General Information
    - Detailed Change Log
    - To Do List and Known Bugs
    - GNU General public License
    - GNU Free Documentation License
    - Apache License
    - Plugin Licensing Amendment

- *Expected Behavior*

When pressing on any of those documents, HTML documents is shown for each of them in order to provide the required information.

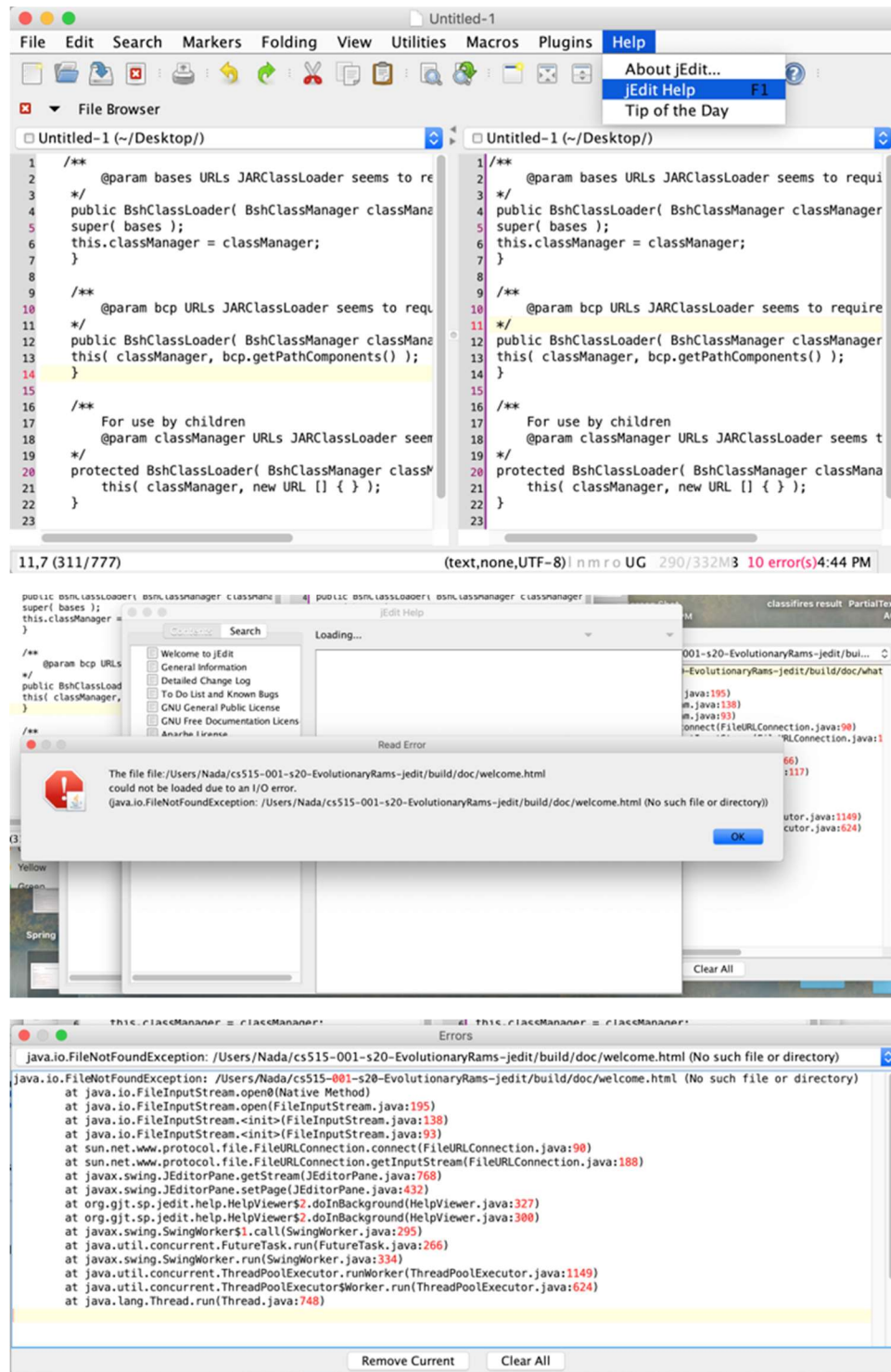
- *Observed Behavior*

File not found Exception message is shown when clicking on these links of documents even though all documents are found in the project.

- *Stack Trace*

```
java.io.FileNotFoundException: /Users/Nada/cs515-001-s20-EvolutionaryRams-jedit/build/doc/users-guide/toc.xml (No such file or directory)
at java.io.FileInputStream.open0(Native Method)
at java.io.FileInputStream.open(FileInputStream.java:195)
at java.io.FileInputStream.<init>(FileInputStream.java:138)
at java.io.FileInputStream.<init>(FileInputStream.java:93)
at sun.net.www.protocol.file.FileURLConnection.connect(FileURLConnection.java:90)
at sun.net.www.protocol.file.FileURLConnection.getInputStream(FileURLConnection.java:188)
at java.net.URL.openStream(URL.java:1057)
at org.gjt.sp.jedit.help.HelpTOCLoader.loadTOC(HelpTOCLoader.java:166)
at org.gjt.sp.jedit.help.HelpTOCLoader.createTOC(HelpTOCLoader.java:118)
at org.gjt.sp.jedit.help.HelpTOCPanel$2.run(HelpTOCPanel.java:105)
at org.gjt.sp.util.TaskManager$MyTask._run(TaskManager.java:258)
at org.gjt.sp.util.Task.run(Task.java:65)
at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1149)
at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:624)
at java.lang.Thread.run(Thread.java:748)
```

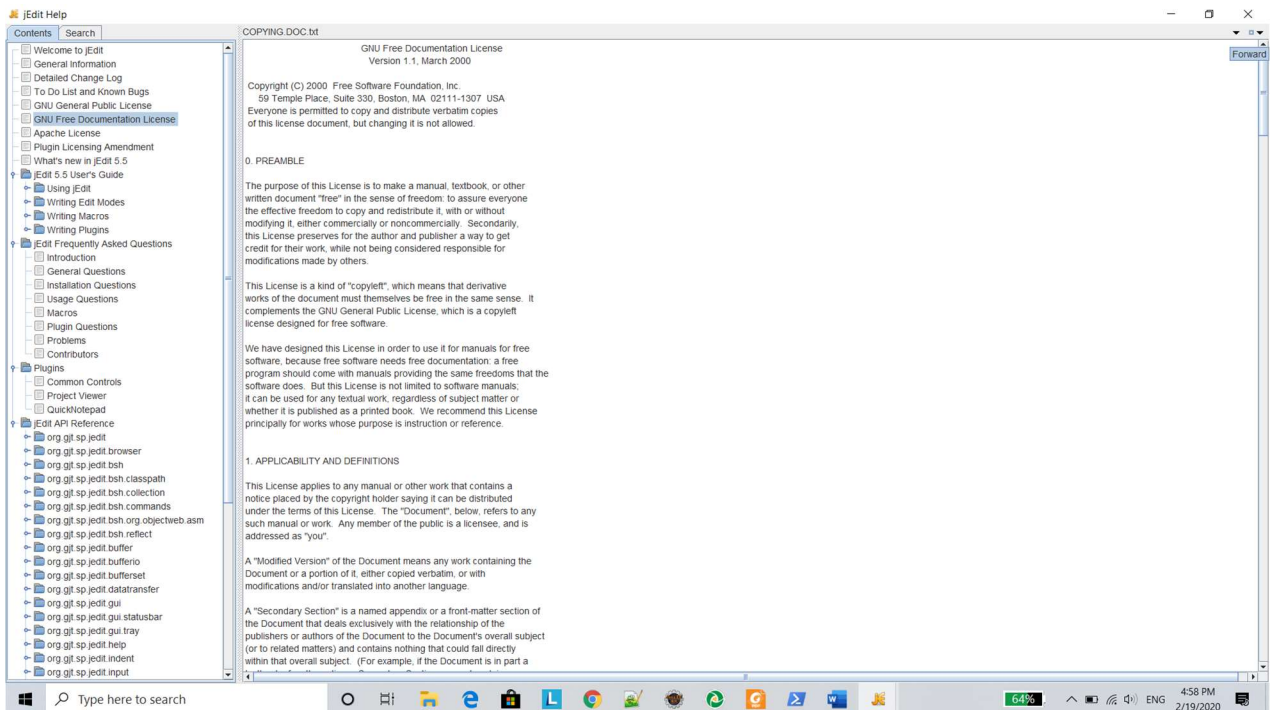
- *Screen Shots*

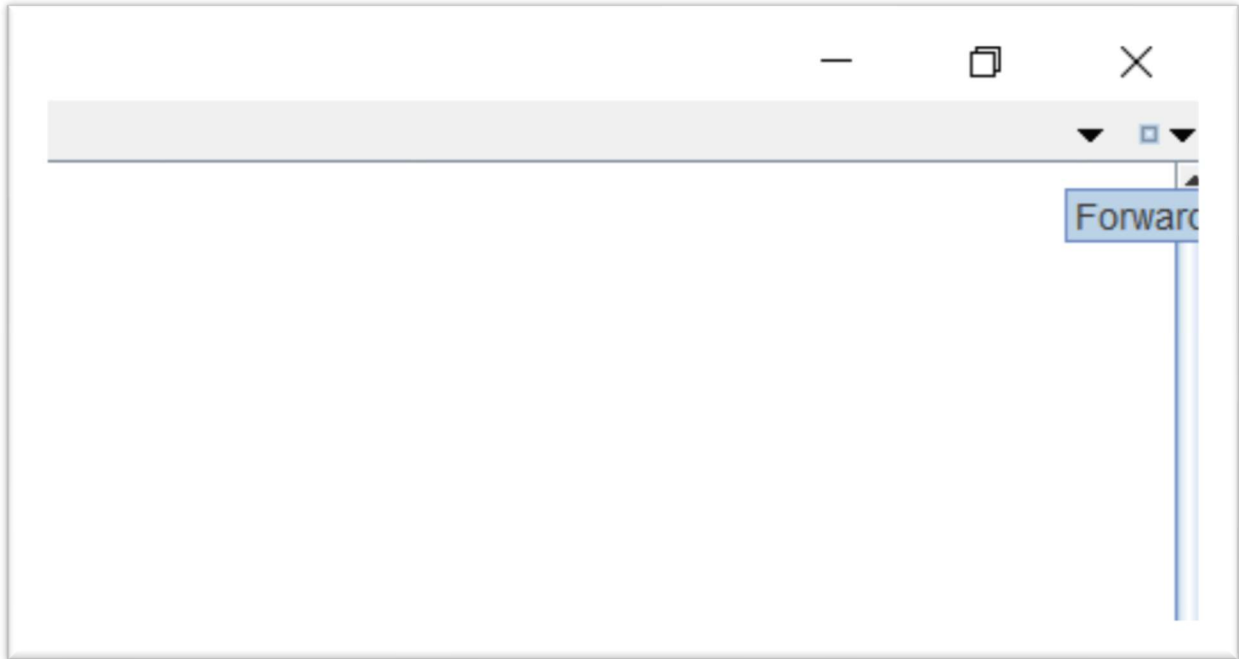




## Bug 2: GUI

- *Summary:*
  - Open jEdit help (Toolbar: 'Help' -> jEdit Help OR press F1)
  - Select any page other than the welcome page (e.g. 'General Information')
  - The buttons in the top right will
- 
- *Steps to reproduce*
  - 1. Open jEdit help (Toolbar: 'Help' -> jEdit Help OR press F1)
  - 2. Select any page other than the welcome page (e.g. 'General Information')
  - 3. The buttons in the top right will
- 
- *Expected Behavior*
  - *Observed Behavior*
  - *Stack Trace*
  - *Screen shots*





## Other Miscellaneous bugs found while going over the code and using the applications

### jEdit – Bug

While using the application, I installed a recommended plugin called Project Viewer.

Opened Toolbar Menu item Plugins -> ProjectViewer -> Projects -> New project here.

This opened up a window which was sized incorrectly. Not all the fields were visible.

### PDFSam – Code Template Issue

Even code files in other functionalities start with the code comment

*This file is part of the PDF Split And Merge source code...*

This seems like a template applied globally.