

DjVu++

The DjVu complete solution



Faculty of Computers and Informatics

Graduation Project

2015

Project name: DjVu++

(Complete System to manipulate Djvu files)

Subervised by:

Dr. Ahmed Sallam

Team Work:

- | | |
|-----------------------|---------------------|
| 1. Alaa Ahmed. | 7. Germen Fayez. |
| 2. Ahmed Ali. | 8. Hussein Mohamed. |
| 3. Ahmed Beder. | 9. Marina Botros. |
| 4. Amr Sayed. | 10. Salah Sayed. |
| 5. Essam AbdelGhafar. | 11. Usama Ahmed. |
| 6. Fathi Hossam. | |

Contents

1. Abstract.....	6
2. Introduction	8
2.1. When To Use DjVu	9
2.2. DjVu Products	10
2.3. Handling Scanned Document	10
2.4. DjVu VS. PDF format	12
2.5. DjVu readers/editors.....	16
3. Stories & Views.....	18
3.1. Start Page	18
3.2. Ribbon Menu	20
3.3. Creating Ribbon Menu	20
3.4. DJVU Ribbon Menu	20
3.5. Home Tasks	20
3.5.1. Tool Band (JRibbon Band):.....	20
3.5.2. View Band (JFlow Ribbon Band):	21
3.5.3. Comment Band (JRibbonBand):	23
3.5.4. Convert Band (JRibbonBand):	24
3.6. Convert Tasks	25
3.6.1. Tool Band (JRibbon Band):.....	25
3.6.2. Convert Band (JRibbonBand):	26
3.7. Edit Tasks.....	27
3.7.1. Tool Band (JRibbon Band):.....	27
3.7.2. Font Band (JFlow Ribbon Band):	28
3.7.3. Paragraph Band (JFlow Ribbon Menu):	31
3.8. Comment Tasks:	33
3.8.1. Tool Band (JRibbon Band):.....	33
3.8.2. Text Makeup (JFlow Ribbon Band):.....	34
3.8.3. Type Writer Band (JRibbon Band):.....	36
3.8.4. Drawing Band (JRibbon Band):	36
3.9. View Tasks:	37

3.9.1. Tool Band (JRibbon Band):.....	38
3.9.2. Go To Band (JFlow Ribbon Band):.....	39
3.9.3. Page Display (JRibbon Band):.....	41
3.9.4. View Setting (JFlow Ribbon Band):	42
3.10. Task Bar	44
3.11. Ribbon Tasks:.....	48
3.12. Ribbon Conceptual Task Group:	49
3.13. Status Bar	50
3.14. DJVU++ Button.....	55
3.15. Help Icon:	62
3.16. Put Bands into Tasks:	63
3.17. Put Tasks into Ribbon Menu:.....	64
3.18. Show Ribbon in DjVu++ Frame:	65
3.19. Show Status bar in DjVu++ Frame:	65
3.20. Tabs:.....	65
3.20.1. Creating of tabs:.....	65
3.21. OCR	66
3.22. Print.....	67
3.23. Outline Tabbed Panel:.....	69
3.24. Full Book continuous View:	72
4. System Architecture	73
4.1. System services.....	74
5. Technical implementation	76
5.1. DJVU++:	76
5.1.1. Implementation:	76
5.1.2. Supplementary resources.....	77
5.2. OCR	78
5.2.1. Implementation:	78
5.2.2. Supplementary resources:.....	79
6. User Guide.....	80
6.1. Inception	80
6.2. Start Page	80

6.3. Top Bar	81
6.4. DJVU++ Button.....	83
6.5. Home.....	85
6.6. Convert.....	89
6.7. Edit	91
6.8. Comment.....	94
6.9. View	96
6.10. Out Line	98
6.11. Thumbnails	99
6.12. Status Bar	100
7. Conclusion & Future Work.....	102
7.1. inception	102
7.2. Future Work	103
References	106

1. Abstract

DjVu++ is a user-friendly program that used to manipulate DjVu file formats such as eBooks with a penalty of editing features. The program introduce a free replacement for the property PDF format with similar resolution and smaller file size

DjVu++ also support OCR to handle text in scanned books and images. The program shows good performance for English. In addition to the Arabic language to lead free and commercial software in this area.

The main features of DjVu++ program are:

- Manipulate DjVu files.
- Support smaller size than PDF with the same performance.
- DjVu++ supports two languages in the OCR technique (Arabic and English).
- Read multiple documents at the same time with the new tabs feature.
- DjVu++ supports multiple formats:
 - Convert PDF document into DjVu format with smaller file size and the same performance.
 - Convert DjVu into PDF format.
 - Combine images to a single DjVu document.

- Perform OCR operations on multiple image formats (png, jpg, bmp and tif).

2. Introduction

Many products and technologies are available today for storing and distributing digital images. However, most of these technologies are slow and inconvenient for distributing high-resolution color photos, and woefully impractical for delivering high-resolution scanned documents in B&W and color, because the files they produce are prohibitively large, and the viewing software memory hungry [1].

With its ability to obtain very high compression ratios while preserving crisp and legible text, DjVu has no rival when it comes to distributing scanned documents, particularly color documents. For this kind of applications, there is quite simply no practical alternative to DjVu.

DjVu is an image compression technique, a Document format and a Software platform for delivering documents images over the internet.

A DjVu document may contain one or several pages, Plug-ins and viewers provide basic mechanisms for navigating between the pages of a multipage document. Multipage documents come in two flavors: BUNDLED, where the whole document is packed in a single file, and INDIRECT, where individual pages are stored in separate files in a directory.

Each DjVu image or page may contain additional chunks of data that include such information as the resolution of the image, hyperlink definitions, highlighted area definitions, default zoom factors, frame color, and display mode.

A DjVu image can also include a compressed text chunk which contains the text on the page in a computer readable format (produced by an OCR software for example). When present, this text chunk is used by viewers and plug-ins to search and highlight words and phrases in DjVu documents.

2.1. When To Use DjVu

The short answer to the question "when to use DjVu" is: Use DjVu for:

- Any image with more than 1 million pixels (high resolution photo or picture, scanned document).
- Any image with text on it (scanned document in color or black and white, digital document).
- Any multipage content, such as a multipage document or a photo.

DjVu is also an excellent format for distributing high-resolution photos, because its wavelet-based continuous tone image compression technology produces small files with very fast progressive display, seamless zooming and panning, and requires minimal memory in the client.

2.2. DjVu Products

Here are some companies that offer DjVu products and services

- LizardTech, a leading developer of imaging solutions for Internet applications, purchased the DjVu technology from AT&T in March 2000.
- Expervision produces and distributes end-user software and development toolkits for Optical Character Recognition. Their latest OCR engine supports DjVu-3.x.
- Princeton Imaging offers document scanning and conversion services. Princeton Imaging has helped create several high profile DjVu websites.
- ACD Systems, makers of the popular image viewer ACDsee have integrated support for DjVu-2.0 in their latest version of ACDsee.
- IrfanView is a freeware image viewing software for Windows that supports many formats. It includes limited support of the DjVu-3.9 format through a plug-in.

2.3. Handling Scanned Document

To handle single-page or multi-page scanned document in color or black and white, and high-resolution photos and pictures, DjVu wraps three compression formats into one.

- DjVuText (also known as JB2): a compression format for black and white (bitonal) image, or for images containing objects of mostly uniform colors.
- DjVuPhoto (also known as IW44): a progressive compression format for color or greyscale photos, paintings, and other continuous-tone images. It is based on the mathematical theory of wavelets [2].
- DjVuLayered (or simply DjVu): applies to documents scanned in color or grayscale that contain mixtures of text, drawings, pictures, and background textures.

2.4. DjVu VS. PDF format

Advantages and disadvantages of DjVu format compared to pdf [3]

Comparison	Advantages	Disadvantages
DjVu	<ul style="list-style-type: none"> • Tried and tested: DjVu have been in use on Wikisource for longer than PDFs. It is more likely that any problems that can occur have already occurred and have been solved. DjVu files are less likely to cause problems with the ProofreadPage Extension or Wikisource in general. • Smaller files: DjVu files are generally smaller than an equivalent PDF 	<ul style="list-style-type: none"> • Lower resolution: DjVu files have a lower resolution than PDFs. For the most part, this is not a problem for proofreading as long as text is legible. It can be a problem with smaller text of text on the borderline of legibility. This might be a problem for illustrations and other images, if the images are being extracted from the file.[1] • Glyphs: Due to the compression system used in DjVu files some glyphs (letters, numbers and other symbols) may be incorrect. The pixels of each page are divided into symbols and a dictionary of these symbols is then created. The pages are then

	<p>file. With exceptionally large documents, this can help fit within Wikimedia Commons' 100MB file size limit. (This was a stronger advantage in the past when the size limit was much smaller).</p> <ul style="list-style-type: none"> • Compatible philosophy: The DjVu format is and always has been an open format, while the PDF was original a proprietary format owned by Adobe. This can give DjVu the edge in the opinion of some Wikimedians whose personal philosophies lean that way. 	<p>encoded by describing which symbols (from the dictionary) appear where. Therefore, the page image may not always be an exact representation of the original work. Note that this should only happen with poor quality, lossy compression.</p> <ul style="list-style-type: none"> • Less external support: The DjVu format is not as widely supported as PDF. There is less software available for creating and editing files in this format.
--	---	--

<p>PDF</p>	<ul style="list-style-type: none"> • Higher resolution: PDFs have a higher resolution than DjVu. This may not make much difference with text which just needs to be legible to be used for transcription by Wikisource, unless the text is particularly small or on the borderline of legibility, in which case PDFs are clearer. This might be an advantage for illustrations and other images, if the images are being extracted from the file.[1] • Wider external support: PDFs are more widely known and supported than DjVu files. Many users can find it easier to create and edit 	<ul style="list-style-type: none"> • Bugs: As a later addition to the original software, PDF-support is more likely to exhibit "buggy" behavior and less likely to be as fully tested as DjVu support. Known bugs include: <ul style="list-style-type: none"> • The inability of the ProofreadPage Extension to recognize accents and diacritics in PDFs. • Some recent versions of PDF cannot be read properly by Wikimedia's Ghost script software. In these cases, all pages appear blank when viewed on a Wikimedia website. • Larger file size: PDFs are generally larger than an equivalent DjVu file. Wikimedia Commons has a 100MB file size limit, which may be a problem with very large documents (and certainly was a problem in the past when this limit was much smaller). • Expensive software: Editing a PDF can still require more
-------------------	---	---

	PDFs for this reason and PDFs may be easier to acquire than DjVu files.	expensive, proprietary software, whereas DjVu is free in both senses of the word.
--	---	---

2.5. DjVu readers/editors

There are many different versions of DjVu readers/editors [4]:

- DjVuLibre
- DjVu Viewer
- WinDjView [5]
- DjVuReader
- STDU Viewer

DjVu Editors	Features
DjVuLibre	DjVuLibre is a free, open source software, which allows you to open and read DjVu files. It's a feature-rich DjVu Viewer which provides dozens of useful tools to view, zoom, rotate, export, and print DjVu documents.
DjVu Viewer	The program window consists of two sections: the left section contains thumbnails of all the pages in DjVu document, and the right section shows full preview of selected thumbnail image. You can easily zoom in or zoom out images in DjVu Viewer.
WinDjView	Provides a tool for seamless zooming of scanned images. Apart from that, it uses various modes to display images, including color mode and black and white mode. It can also display backgrounds of scanned images [6].
DjVuReader	It allows you to zoom in or zoom out pages, and view them in different modes (colored, mask, background, and foreground). Apart from that, it provides custom sliders to easily adjust brightness or contrast of the pages. It also allows to select a rectangular portion of the page and copy it to the clipboard. You can also copy hidden text to clipboard. DjVuReader is

	simple, nice, and a handy program, which is appropriate for both professional and novice users.
STDUviewer	Provides various options to adjust paper settings, brightness/contrast, page width, page layout, and other such settings, print images or search text within the document, create your own bookmarks, so that you can access your favorite pages quickly and easily.
Easy DJVU Reader for iPhone (\$)	(commercial)Open files of DJVU format and read them on your iPhone.

3. Stories & Views

3.1. Start Page

It is the first tab that opened when the program is start, it

Consists of some operations.

- Package: com.lizardtech.djview.frame
- Class: DjVuStart
- Method: new StartPage.ContentPane
- Recent Files:
 - Package: recent file
 - Class: FilesavePath
 - It takes one parameter (path of book).
 - Add book in jlist by get path of book and save it in file
 - Check if file found or not.
- Select Task:
 - Package: main page

It consists of:

- Convert DjVu To PDF:
 - Package: main Page
 - Class: DjVuToPdf

- Convert PDF To DjVu:
 - Package: main Page
 - Class: PdfToDjVu
- Convert DjVu To Image:
 - Package: main Page
 - Class: DjVuToImage
- Convert Image To DjVu:
 - Package: main Page
 - Class: ImageToDjVu
- Convert PDF To Image:
 - Package: main Page
 - Class: PdfToImage
- Convert Image To PDF:
 - Package: main Page
 - Class: ImageToPdf
- DjVu++ Online:

It is a button that connect a link to the main Website of DjVu++.

- Package: com.lizardtech.djview.frame

3.2. Ribbon Menu

It uses four Java Archives (jar):

- Flamingo.
- Trident.
- Substance.
- Substance flamingo.

3.3. Creating Ribbon Menu

It consists of three parts:

- Tasks: Each task have group of Bands (JRibbon Band and JFlow Ribbon Band).
- Task Bar.
- Ribbon App Menu.

3.4. DJVU Ribbon Menu

It consists of four tasks (Home, Convert, Edit and View).

3.5. Home Tasks

It consists of five bands:

3.5.1. Tool Band (JRibbon Band):

- Package: com.lizardtech.DjVubean.RibbonMenu
- Class: DjVu Component that implement Ribbon get icon
- Methods:

- getHomertoolsband that return tools band
- AddCommandButton that take two parameters
(JcommandButton&RibbonElementPeriority (top or medium))
- Hand Tool: is an object from (jcommand button) class whose constructor with two parameters (Name and Icon of this object).
- Select Tool: is an object from (jcommand button) Class whose constructor with two parameters (Name and Icon of this object).
- Zoom Area: is an object from (jcommand button) Class whose constructor with two parameters (Name and Icon of this object).
- Snapshot: is an object from (jcommand button) Class whose constructor with two parameters (Name and Icon of tis object).
- Clipboard: is an object from (jcommand button) Class whose constructor with two parameters (Name and Icon of this object) that make pop up only having JcommandMenuButtons are (Copy, Cut, Paste, Select all) that the previous Buttons are objects from (jcommand button) Class whose constructor with two parameters (Name and Icon of this object) and to Make Clipboard as a kind of pop up only command button should use (SetCommandButton) that take an object from type (JcommandButton.CommandButtonKind.popuponly).
- To add any component into this JRibbonBand you should identify (JRibbonBandResizablePolicy): identify (ArrayList) ,then use this array list to determine the resizable policy by using ADD function that taking object from (CoreRibbonResizablePolicy.Mirror(JBandControlPanel)) and to apply the resizable policy in band you should call (SetResizablePolicy that take one Parameter (ArrayList)).

3.5.2. View Band (JFlow Ribbon Band):

- Package: com.lizardtech.DjVuBean.RibbonMenu
- Class: DjVu Component that implement Ribbon get icon
- Methods:
 - Gethomeviewband that return view band
 - Addflowcomponent that take one parameter (JRibbonComponent or JcommandButton)
- Actual Size: shown in code as (ASize) object, can determine its form by (SetDisplayState) that take one parameter (commandButtonDisplayState.Medium)
- Fit Page: shown in code as (FitPage) Object , can determine its form by (SetDisplayState) that take one parameter (commandButtonDisplayState.Medium)
- Fit Weights: shown in code as (Fit Weights) Object, can determine its form by (SetDisplayState) that take one parameter (commandButtonDisplayState.Medium)
- Fit Visible: shown in code as (Fit Visible) Object , can determine its form by (SetDisplayState) that take one parameter (commandButtonDisplayState.Medium)
- Rotate Lift: shown in code as “ RotateLift ” Object , can determine its form by (SetDisplayState) that take one parameter (commandButtonDisplayState.Medium)
- Rotate Right: shown in code as (RotateRight) Object , can determine its form by (SetDisplayState) that take one parameter (commandButtonDisplayState.Medium)
- JComboBox: shown in code as (ZoomB) Object, identified By (JComboBox) and make it (Set Editable) to write on it, in ZoomB it

will identify an object from (JRibbonComponent) that take that ZoomB.

- Zoom in and Zoom out: Shown in Code as (Command Buttons)
Then identify object named as (Strip) that its type is (JCommandButtonStrip) and make its State by (SetDisplayState) then adding (zoom in & zoom out) on that Strip to shown into their shape.
- To add any component into this JRibbonBand you should identify (JRibbonBandResizablePolicy): identify (ArrayList) ,then use this array list to determine the resizable policy by using ADD function that taking object from (CoreRibbonResizablePolicy.Mirror(JBandControlPanel)) and to apply the resizable policy in band you should call (SetResizablePolicy that take one Parameter (ArrayList)).

3.5.3. Comment Band (JRibbonBand):

- Package: com.lizardtech.DjVuBean.RibbonMenu
- Class: DjVu Component that implement Ribbon get icon
- Methods:
 - getHomecommentband that return comment band
 - AddCommandButton that take two parameters (JcommandButton&RibbonElementPeriority (top or medium)
- Typewriter: is an object from (jcommand button) class whose constructor with two parameters (name of this object and icon of this object)
- Note: is an object from (jcommand button) class whose constructor with two parameters (name of this object and icon of this object)

- Highlight: is an object from (jcommand button) class whose constructor with two parameters (name of this object and icon of this object)
- Strikeout: is an object from (jcommand button) class whose constructor with two parameters name of this object and icon of this object)
- Underline: is an object from (jcommand button) class whose constructor with two parameters (name of this object and icon of this object)
- To add any component into this JRibbonBand you should identify (JRibbonBandResizablePolicy): identify (ArrayList) ,then use this array list to determine the resizable policy by using ADD function that taking object from
(CoreRibbonResizablePolicy.Mirror(JBandControlPanel))
and to apply the resizable policy in band you should call
(SetResizablePolicy that take one Parameter (ArrayList)).

3.5.4. Convert Band (JRibbonBand):

- Package: com.lizardtech.DjVuBean.RibbonMenu
- Class: DjVu Component that implement Ribbon get icon
- Methods:
 - getHomeconvertband that return Convert band
 - AddCommandButton that take two parameters
(JcommandButton and RibbonElementPeriority (top or medium)
- OCR: is an object from (jcommand button) class whose constructor with two parameters (name of this object and icon of this object) that make pop up only having JcommandMenuButtons are (Ocr

Arabic and Ocr English) that the previous Buttons are objects from (jcommand button) class whose constructor with two parameters (name of this object and icon of this object)

- To make OCR as a kind of pop up only command button should use (SetCommandButton) that take an object from type (JcommandButton.CommandButtonKind.popuponly).
- To add any component into this JRibbonBand you should identify (JRibbonBandResizablePolicy): identify (ArrayList) ,then use this array list to determine the resizable policy by using ADD function that taking object from (CoreRibbonResizablePolicy.Mirror(JBandControlPanel)) and to apply the resizable policy in band you should call (SetResizablePolicy that take one Parameter (ArrayList)).

3.6. Convert Tasks

It consists of two bands:

3.6.1. Tool Band (JRibbon Band):

- Package: com.lizardtech.DjVuBean.RibbonMenu
- Class: DjVu Component that implement Ribbon get icon
- Methods:
 - getHometoolsband that return tools band
 - AddCommandButton that take two parameters (JcommandButton&RibbonElementPeriority (top or medium))
- Hand Tool: is an object from (jcommand button) class whose constructor with two parameters (Name and Icon of this object).

- Select Tool: is an object from (Jcommand button) Class whose constructor with two parameters (Name and Icon of this object).
- Zoom Area: is an object from (Jcommand button) Class whose constructor with two parameters (Name and Icon of this object).
- Snapshot: is an object from (Jcommand button) Class whose constructor with two parameters (Name and Icon of tis object).
- Clipboard: is an object from (Jcommand button) Class whose constructor with two parameters (Name and Icon of this object) that make pop up only having JcommandMenuButtons are (Copy, Cut, Paste, Select all) that the previous Buttons are objects from (Jcommand button) Class whose constructor with two parameters (Name and Icon of this object) and to Make Clipboard as a kind of pop up only command button should use (SetCommandButton) that take an object from type (JcommandButton.CommandButtonKind.popuponly).
- To add any component into this JRibbonBand you should identify (JRibbonBandResizablePolicy): identify (ArrayList) ,then use this array list to determine the resizable policy by using ADD function that taking object from (CoreRibbonResizablePolicy.Mirror(JBandControlPanel)) and to apply the resizable policy in band you should call (SetResizablePolicy that take one Parameter (ArrayList)).

3.6.2. Convert Band (JRibbonBand):

- Package: com.lizardtech.DjVuBean.RibbonMenu
- Class: DjVu Component that implement Ribbon get icon
- Methods:
 - getHomeconvertband that return Convert band

- AddCommandButton that take two parameters
(JcommandButton and RibbonElementPeriority (top or medium))
- OCR: is an object from (jcommand button) class whose constructor with two parameters (name of this object and icon of this object) that make pop up only having JcommandMenuButtons are (Ocr Arabic and Ocr English) that the previous Buttons are objects from (jcommand button) class whose constructor with two parameters (name of this object and icon of this object)
- To make OCR as a kind of pop up only command button should use (SetCommandButton) that take an object from type (JcommandButton.CommandButtonKind.popuponly).
- To add any component into this JRibbonBand you should identify (JRibbonBandResizablePolicy): identify (ArrayList) ,then use this array list to determine the resizable policy by using ADD function that taking object from (CoreRibbonResizablePolicy.Mirror(JBandControlPanel)) and to apply the resizable policy in band you should call (SetResizablePolicy that take one Parameter (ArrayList)).

3.7. Edit Tasks

It consists of three bands:

3.7.1. Tool Band (JRibbon Band):

- Package: com.lizardtech.DjVubean.RibbonMenu
- Class: DjVu Component that implement Ribbon get icon
- Methods:
 - getToolForEditBand

- AddCommandButton that take two parameters
(JcommandButton&RibbonElementPeriority (top or medium))
- Hand Tool: is an object from (jcommand button) class whose constructor with two parameters (Name and Icon of this object).
- Select Tool: is an object from (jcommand button) Class whose constructor with two parameters (Name and Icon of this object).
- Zoom Area: is an object from (jcommand button) Class whose constructor with two parameters (Name and Icon of this object).
- Snapshot: is an object from (jcommand button) Class whose constructor with two parameters (Name and Icon of tis object).
- Clipboard: is an object from (jcommand button) Class whose constructor with two parameters (Name and Icon of this object) that make pop up only having JcommandMenuButtons are (Copy, Cut, Paste, Select all) that the previous Buttons are objects from (jcommand button) Class whose constructor with two parameters (Name and Icon of this object) and to Make Clipboard as a kind of pop up only command button should use (SetCommandButton) that take an object from type
(JcommandButton.CommandButtonKind.popuponly).
- To add any component into this JRibbonBand you should identify (JRibbonBandResizablePolicy): identify (ArrayList) ,then use this array list to determine the resizable policy by using ADD function that taking object from
(CoreRibbonResizablePolicy.Mirror(JBandControlPanel))
and to apply the resizable policy in band you should call
(SetResizablePolicy that take one Parameter (ArrayList)).

3.7.2. Font Band (JFlow Ribbon Band):

- Package: com.lizardtech.DjVubean.RibbonMenu
- Class: DjVu Component that implement Ribbon get icon
- Methods:
 - getFontBand
 - Addflowcomponent that take one parameter (JRibbonComponent or JcommandButton)
- Font combobox: JComboBox that takes an array of type of font. To add the combobox in it you should identify an object from (JRibbon component), this object is called (FontComboWrapper), then function (Addflowcomponent) that take the (JRibbon Component).
- Size combobox: JComboBox that takes an array of type of font. To add the combobox in it you should identify an object from (JRibbon component), this object is called (SizeComboWrapper), then function (Addflowcomponent) that take the (JRibbon Component).
- Style Strip:

It is JCommandButtonStrip.

- To add more than one JcommandToggleButton, identify object (StyleBoldButton).
- JcommandToggleButton take two parameter (Name and Icon of this object).
- Note that: any Icon get from interface (JRibbonIcon) that have one method (getResizableIcon).

- To add StyleBoldButton in style strip you should use addFunction that take one parameter.
- Style Italic Button:

It is JCommandButtonStrip.

- To add more than one JcommandToggleButton, identify object (StyleItalicButton).
- JcommandToggleButton take two parameter (Name and Icon of this object).
- Note that: any Icon get from interface (JRibbonIcon) that have one method (getResizableIcon).
- To add StyleItalicButton in style strip you should use addFunction that take one parameter.
- Style Underline Button:

It is JCommandButtonStrip.

- To add more than one JcommandToggleButton, identify object (StyleUnderlineButton).
- JcommandToggleButton take two parameter (Name and Icon of this object).
- Note that: any Icon get from interface (JRibbonIcon) that have one method (getResizableIcon).
- To add StyleUnderlineButton in style strip you should use addFunction that take one parameter.
- Color Combobox:

- Get from (getCommandButtonColor).
 - It is in the same package and in class (ColorSelector)
 - It popup only from (JColorSelectorPopupMenu):
 - It have JcommandMenuButton (automatic color and more colors).
 - It take two parameters (Name and Icon).
 - Add recent color from method (addColorSection).
 - Add themes color from method (addColorSectionWithDerived).
 - Those methods added into result from method (addMenuButton).
- To add any component into this JRibbonBand you should identify (JRibbonBandResizablePolicy): identify (ArrayList) ,then use this array list to determine the resizable policy by using ADD function that taking object from (CoreRibbonResizablePolicy.Mirror(JBandControlPanel)) and to apply the resizable policy in band you should call (SetResizablePolicy that take two Parameter (ArrayList)).

3.7.3. Paragraph Band (JFlow Ribbon Menu):

- Package: com.lizardtech.DjVuBean.RibbonMenu
- Class: DjVu Component that implement Ribbon get icon
- Methods:
 - getParagraphBand
 - Addflowcomponent that take one parameter (JRibbonComponent or JcommandButton)
 - It have (intent Strip) and JCommandButtonStrip:
 - To add more than one JcommandToggleButton, identify object (StyleBoldButton).

- JcommandToggleButton take two parameter (Name and Icon of this object).
 - Note that: any Icon get from interface (JRibbonIcon) that have one method (getResizableIcon).
 - To add StyleBoldButton in style strip you should use addFunction that take one parameter.
- Intent Left Button: is an object from (jcommand button) class whose constructor with two parameters (Name and Icon of this object).
 - Intent Right Button: is an object from (jcommand button) class whose constructor with two parameters (Name and Icon of this object).
 - Align Strip:

It is JCommandButtonStrip.

- To add more than one JcommandToggleButton, identify object (StyleBoldButton).
 - JcommandToggleButton take two parameter (Name and Icon of this object).
 - Note that: any Icon get from interface (JRibbonIcon) that have one method (getResizableIcon).
 - To add StyleBoldButton in style strip you should use addFunction that take one parameter.
- Identify align group (CommandToggleButtonGroup) that have a group of toggle button:
 - Align Left Button: it is a (Jcommand toggled button) and its constructor take two parameters (Name and Icon).

- Align Center Button: it is a (Jcommand toggled button) and its constructor take two parameters (Name and Icon).
- Align Right Button: it is a (Jcommand toggled button) and its constructor take two parameters (Name and Icon).
- They added into strip by add () that take one parameter.
- This group is created to use it like (JRadio Button) in an action.
- Add align strip and intent strip by (addFlowComponent).
- To add any component into this JRibbonBand you should identify (JRibbonBandResizablePolicy): identify (ArrayList) ,then use this array list to determine the resizable policy by using ADD function that taking object from
(CoreRibbonResizablePolicy.Mirror(JBandControlPanel))
and to apply the resizable policy in band you should call
(SetResizablePolicy that take two Parameter (ArrayList)).

3.8. Comment Tasks:

It consists of three bands:

3.8.1. Tool Band (JRibbon Band):

- Package: com.lizardtech.DjVubean.RibbonMenu
- Class: DjVu Component that implement Ribbon get icon
- Methods:
 - getToolForCommentBand
 - AddCommandButton that take two parameters
(JcommandButton&RibbonElementPeriority (top or medium))

- Hand Tool: is an object from (JCommandButton) class whose constructor with two parameters (Name and Icon of this object).
- Select Tool: is an object from (JCommandButton) Class whose constructor with two parameters (Name and Icon of this object).
- Zoom Area: is an object from (JCommandButton) Class whose constructor with two parameters (Name and Icon of this object).
- Snapshot: is an object from (JCommandButton) Class whose constructor with two parameters (Name and Icon of tis object).
- Clipboard: is an object from (JCommandButton) Class whose constructor with two parameters (Name and Icon of this object) that make pop up only having JCommandMenuButtons are (Copy, Cut, Paste, Select all) that the previous Buttons are objects from (JCommandButton) Class whose constructor with two parameters (Name and Icon of this object) and to Make Clipboard as a kind of pop up only command button should use (SetCommandButton) that take an object from type (JCommandButton.CommandButtonKind.popuponly).
- To add any component into this JRibbonBand you should identify (JRibbonBandResizablePolicy): identify (ArrayList) ,then use this array list to determine the resizable policy by using ADD function that taking object from (CoreRibbonResizablePolicy.Mirror(JBandControlPanel)) and to apply the resizable policy in band you should call (SetResizablePolicy that take one Parameter (ArrayList)).

3.8.2. Text Makeup (JFlow Ribbon Band):

- Package: com.lizardtech.DjVubean.RibbonMenu
- Class: RibbonMenuCreation

- Highlight (Jcommand Button): it take two parameters (Name and Icon) and Change display state from method (SetDisplayState) that take parameter (CommandButtonDisplayState.Small)
- Underline (Jcommand Button): it take two parameters (Name and Icon) and Change display state from method (SetDisplayState) that take parameter (CommandButtonDisplayState.Small)
- Underline two (Jcommand Button): it take two parameters (Name and Icon) and Change display state from method (SetDisplayState) that take parameter (CommandButtonDisplayState.Small)
- Strikeout (Jcommand Button): it take two parameters (Name and Icon) and Change display state from method (SetDisplayState) that take parameter (CommandButtonDisplayState.Small)
- Strikeout two (Jcommand Button): it take two parameters (Name and Icon) and Change display state from method (SetDisplayState) that take parameter (CommandButtonDisplayState.Small)
- Strikeout three (Jcommand Button): it take two parameters (Name and Icon) and Change display state from method (SetDisplayState) that take parameter (CommandButtonDisplayState.Small)
- Note that: to identify JFlow Ribbon Band: identify from class (JFlowRibbonBand), its constructor take two parameters (Name and Icon).
- To add any component into this JRibbonBand you should identify (JRibbonBandResizablePolicy): identify (ArrayList) ,then use this array list to determine the resizable policy by using ADD function that taking object from (CoreRibbonResizablePolicy.Mirror(JBandControlPanel)) and to apply the resizable policy in band you should call (SetResizablePolicy that take one Parameter (ArrayList)).

3.8.3. Type Writer Band (JRibbon Band):

- Package: com.lizardtech.DjVubean.RibbonMenu
- Class: RibbonMenuCreation
- To make separator between groups using set resize policies (CoreRibbonResizePolicies.getCorePoliciesRestrictive (Type Writer)).
- Note Two: it is Jcommand Button added in Band by (AddCommandButton) and take (Jcommand Button and Ribbon element priority) and It takes (RibbonElementPriority.Top)
- TWriter: it is Jcommand Button added in Band by (AddCommandButton) and take (Jcommand Button and Ribbon element priority) and It takes (RibbonElementPriority.Top)
- Text Box: it is Jcommand Button added in Band by (AddCommandButton) and take (Jcommand Button and Ribbon element priority) and It takes (RibbonElementPriority.Medium)
- Call Out: it is Jcommand Button added in Band by (AddCommandButton) and take (Jcommand Button and Ribbon element priority) and It takes (RibbonElementPriority.Medium)
- To add any component into this JRibbonBand you should identify (JRibbonBandResizablePolicy): identify (ArrayList) ,then use this array list to determine the resizable policy by using ADD function that taking object from (CoreRibbonResizablePolicy.Mirror(JBandControlPanel)) and to apply the resizable policy in band you should call (SetResizablePolicy that take one Parameter (ArrayList)).

3.8.4. Drawing Band (JRibbon Band):

- Package: com.lizardtech.DjVubean.RibbonMenu
- Class: RibbonMenuCreation

- Method: `getDrawingBand`
- Rect:
 - Jcommand button that take two parameter (Name and Icon).
 - Display State Changed from (`SetDisplayState`) and take object from (`CommandButtonDisplayState.Small`).
 - It is added in drawing band by (`addCommandButton`) and it takes two parameters (`JcommandButton` and `RibbonElementPriority`)
 - It takes `RibbonElementPriority.Medum`
- Area Highlight:
 - It is added in drawing band by (`addCommandButton`) and it takes two parameters (`JcommandButton` and `RibbonElementPriority`)
 - It takes `RibbonElementPriority.Medum`
- To add any component into this `JRibbonBand` you should identify (`JRibbonBandResizablePolicy`): identify (`ArrayList`) ,then use this array list to determine the resizable policy by using `ADD` function that taking object from (`CoreRibbonResizablePolicy.Mirror(JBandControlPanel)`) and to apply the resizable policy in band you should call (`SetResizablePolicy` that take one Parameter (`ArrayList`)).

3.9. View Tasks:

It consists of four bands:

3.9.1. Tool Band (JRibbon Band):

- Package: com.lizardtech.DjVuBean.RibbonMenu
- Class: DjVu Component that implement Ribbon get icon
- Methods:
 - getToolForViewBand
 - AddCommandButton that take two parameters
(JcommandButton&RibbonElementPeriority (top or medium))
- Hand Tool: is an object from (jcommand button) class whose constructor with two parameters (Name and Icon of this object).
- Select Tool: is an object from (jcommand button) Class whose constructor with two parameters (Name and Icon of this object).
- Zoom Area: is an object from (jcommand button) Class whose constructor with two parameters (Name and Icon of this object).
- Snapshot: is an object from (jcommand button) Class whose constructor with two parameters (Name and Icon of tis object).
- Clipboard: is an object from (jcommand button) Class whose constructor with two parameters (Name and Icon of this object) that make pop up only having JcommandMenuButtons are (Copy, Cut, Paste, Select all) that the previous Buttons are objects from (jcommand button) Class whose constructor with two parameters (Name and Icon of this object) and to Make Clipboard as a kind of pop up only command button should use (SetCommandButton) that take an object from type
(JcommandButton.CommandButtonKind.popuponly).
- To add any component into this JRibbonBand you should identify (JRibbonBandResizablePolicy): identify (ArrayList) ,then use this array list to determine the resizable policy by using ADD function

that taking object from
(CoreRibbonResizablePolicy.Mirror(JBandControlPanel))
and to apply the resizable policy in band you should call
(SetResizablePolicy that take one Parameter (ArrayList)).

3.9.2. Go To Band (JFlow Ribbon Band):

- Package: com.lizardtech.DjVubean.RibbonMenu
- Class: DjVu Component that implement Ribbon get icon
- Method: getGoToBand
- Panel:
 - Package: com.lizardtech.djview.frame
 - Class: PageSelect
 - Method: goToPanel
 - Go To First Page:
 - It is a Jcommand Button.
 - Its constructor takes two parameters (Name and Icon).
 - It added into panel using (add (JComponent)).
 - The panel that contains it is added into Go to Band using (addFlowComponent ()).
 - Go To Last Page:
 - It is a Jcommand Button.
 - Its constructor takes two parameters (Name and Icon).
 - It added into panel using (add (JComponent)).
 - The panel that contains it is added into Go to Band using (addFlowComponent ()).
 - Go To Next Page:
 - It is a Jcommand Button.

- Its constructor takes two parameters (Name and Icon).
- It added into panel using (add (JComponent)).
- The panel that contains it is added into Go to Band using (addFlowComponent ()).
- Go To Prev Page:
- It is a Jcommand Button.
- Its constructor takes two parameters (Name and Icon).
- It added into panel using (add (JComponent)).
- The panel that contains it is added into Go to Band using (addFlowComponent ()).
- Go To Area:
- JText Field.
- Its constructor takes one parameter (Size of text field).
- It added into panel using (add (JComponent)).
- The panel that contains it is added into Go to Band using (addFlowComponent ()).
- Next View:
 - It is a Jcommand Band.
 - Display State Changed from (setDisplayState) and take object from (CommandButtonDisplayState.Small).
 - It is added into the panel using (addFlowComponent ()).
- Prev View:
 - It is a Jcommand Band.
 - Display State Changed from (setDisplayState) and take object from (CommandButtonDisplayState.Small).

- It is added into the panel using (addFlowComponent ()).
- To add any component into this JRibbonBand you should identify (JRibbonBandResizablePolicy): identify (ArrayList) ,then use this array list to determine the resizable policy by using ADD function that taking object from (CoreRibbonResizablePolicy.Mirror(JBandControlPanel)) and to apply the resizable policy in band you should call (SetResizablePolicy that take two Parameter (ArrayList)).

3.9.3. Page Display (JRibbon Band):

- Package: com.lizardtech.djview.frame
- Class: PageSelect
- Method: getPageDisplay
- Rotate Page2:
 - It is Jcommand Button.
 - It change its kind using (SetCommandButtonKind (JcommandButton.CommandButtonKind.Action_and_popup_main_action)).
 - It is popup using object (JpopupMenu) and the object is called Menu.
 - The Menu have two JcommandMenuButton.
 - It added in band using (addCommandButton ()) that takes (JcommandButton and RibbonElementPriority.Top).
- Rotate Left:

- It is Jcommand Menu Button.
 - Its constructor takes two parameters (Name and Icon)
 - Display State Changed from (setDisplayState) and take object from (CommandButtonDisplayState.Medium).
 - It is added in menu using (addMenuButton ()).
- Rotate Right:
 - It is Jcommand Menu Button.
 - Its constructor takes two parameters (Name and Icon)
 - Display State Changed from (setDisplayState) and take object from (CommandButtonDisplayState.Medium).
 - It is added in menu using (addMenuButton ()).
- Single:
 - Jcommand Button.
 - Its constructor takes two parameters (Name and Icon).
 - It added in band using (addCommandButton ()) that takes (JcommandButton and RibbonElementPriority.Medium).
- To add any component into this JRibbonBand you should identify (JRibbonBandResizablePolicy): identify (ArrayList) ,then use this array list to determine the resizable policy by using ADD function that taking object from (CoreRibbonResizablePolicy.Mirror(JBandControlPanel)) and to apply the resizable policy in band you should call (SetResizablePolicy that take two Parameter (ArrayList)).

3.9.4. View Setting (JFlow Ribbon Band):

- Package: com.lizardtech.djview.frame

- Class: RibbonMenuCreation
- Method: getViewSetting
- App Menu Vis:
 - It is a JCheckBox.
 - Its constructor takes one parameter (Name).
 - It is added by creating panel (JCheckBox) and its type is JPanel.
 - It is put in panel using add ().
 - The panel contains it put into Band using (addFlowComponent ()).
- Task Bar Vis:
 - It is a JCheckBox.
 - Its constructor takes one parameter (Name).
 - It is added by creating panel (JCheckBox) and its type is JPanel.
 - It is put in panel using add ().
 - The panel contains it put into Band using (addFlowComponent ()).
- Show Status:
 - It is a JRadio Button.
 - It is put in a group called (Button Group) using add ().
 - Then put it in JPanel using (StatusRadioButton ()).
- Hidden Status:
 - It is a JRadio Button.

- It is put in a group called (Button Group) using add ().
- Then put it in Jpanel using (StatusRadioButton ()).
- Auto Hide:
 - It is a JRadio Button.
 - It is put in a group called (Button Group) using add ().
 - Then put it in Jpanel using (StatusRadioButton ()).

3.10. Task Bar

- Package: com.lizardtech.djview.frame
- Class: RibbonMenuCreation
- Method: configureTaskBarComponents
- Open:
 - It is a Jcommand Button.
 - Its Constructor takes two parameters (Name and Icon).
 - It is added in Taskbar using
(DjVu.getRibbon.addTaskBarComponent ()).
 - Its Constructor takes one parameter (Component).
 - It is shown in Ribbon using Function (SetRibbonMenu), then call the method (configureTaskBarComponents ()).
- Save:
 - It is a Jcommand Button.
 - Its Constructor takes two parameters (Name and Icon).
 - It is added in Taskbar using
(DjVu.getRibbon.addTaskBarComponent ()).

- Its Constructor takes one parameter (Component).
- It is shown in Ribbon using Function (SetRibbonMenu), then call the method (configureTaskBarComponents ()).
- Undo:
 - It is a Jcommand Button.
 - Its Constructor takes two parameters (Name and Icon).
 - It is added in Taskbar using (DjVu.getRibbon.addTaskBarComponent ()).
 - Its Constructor takes one parameter (Component).
 - It is shown in Ribbon using Function (SetRibbonMenu), then call the method (configureTaskBarComponents ()).
- Redo:
 - It is a Jcommand Button.
 - Its Constructor takes two parameters (Name and Icon).
 - It is added in Taskbar using (DjVu.getRibbon.addTaskBarComponent ()).
 - Its Constructor takes one parameter (Component).
 - It is shown in Ribbon using Function (SetRibbonMenu), then call the method (configureTaskBarComponents ()).
- TpPrinter:
 - It is a Jcommand Button.
 - Its Constructor takes two parameters (Name and Icon).
 - It is added in Taskbar using (DjVu.getRibbon.addTaskBarComponent ()).

- Its Constructor takes one parameter (Component).
- It is shown in Ribbon using Function (SetRibbonMenu), then call the method (configureTaskBarComponents ()).
- TpBlank:
 - It is a Jcommand Button.
 - Its Constructor takes two parameters (Name and Icon).
 - It is added in Taskbar using (DjVu.getRibbon.addTaskBarComponent ()).
 - Its Constructor takes one parameter (Component).
 - It is shown in Ribbon using Function (SetRibbonMenu), then call the method (configureTaskBarComponents ()).
- Search:
 - It is a Jcommand Button.
 - Its Constructor takes two parameters (Name and Icon).
 - It is added in Taskbar using (DjVu.getRibbon.addTaskBarComponent ()).
 - Its Constructor takes one parameter (Component).
 - It is shown in Ribbon using Function (SetRibbonMenu), then call the method (configureTaskBarComponents ()).
- Search Two:
 - It is a Jcommand Button.
 - Its Constructor takes two parameters (Name and Icon).
 - It is added in Taskbar using (DjVu.getRibbon.addTaskBarComponent ()).

- Its Constructor takes one parameter (Component).
- It is shown in Ribbon using Function (SetRibbonMenu), then call the method (configureTaskBarComponents ()).
- Find Text:
 - It is a Jcommand Button.
 - Its Constructor takes two parameters (Name and Icon).
 - It is added in Taskbar using (DjVu.getRibbon.addTaskBarComponent ()).
 - Its Constructor takes one parameter (Component).
 - It is shown in Ribbon using Function (SetRibbonMenu), then call the method (configureTaskBarComponents ()).
- Find Prev:
 - It is a Jcommand Button.
 - Its Constructor takes two parameters (Name and Icon).
 - It is added in Taskbar using (DjVu.getRibbon.addTaskBarComponent ()).
 - Its Constructor takes one parameter (Component).
 - It is shown in Ribbon using Function (SetRibbonMenu), then call the method (configureTaskBarComponents ()).
- Area Search:
 - It is a Jcommand Button.
 - Its Constructor takes two parameters (Name and Icon).
 - It is added in Taskbar using (DjVu.getRibbon.addTaskBarComponent ()).

- Its Constructor takes one parameter (Component).
- It is shown in Ribbon using Function (SetRibbonMenu), then call the method (configureTaskBarComponents ()).

3.11. Ribbon Tasks:

- Highlight:
 - It is identified by Class (RibbonTask).
 - Its constructor takes two parameters (Task Name and Array of Bands).
- Underline:
 - It is identified by Class (RibbonTask).
 - Its constructor takes two parameters (Task Name and Array of Bands).
- Strikeout:
 - It is identified by Class (RibbonTask).
 - Its constructor takes two parameters (Task Name and Array of Bands).
- Squiggly:
 - It is identified by Class (RibbonTask).
 - Its constructor takes two parameters (Task Name and Array of Bands).
- Replace:
 - It is identified by Class (RibbonTask).

- Its constructor takes two parameters (Task Name and Array of Bands).
- Insert:
 - It is identified by Class (RibbonTask).
 - Its constructor takes two parameters (Task Name and Array of Bands).

3.12. Ribbon Conceptual Task Group:

- HighUnder:
 - Its constructor takes three Parameters (Title, Color and Tasks).
 - It Appears when need to add action using change property of it.
 - It is added into Ribbon using Method (addConceptualTaskGroup()).
 - It is shown by (DjVu.getRibbon.setVisible (Ribbon conceptual task Group and Boolean (is Visible))).
- Strike:
 - Its constructor takes three Parameters (Title, Color and Tasks).
 - It Appears when need to add action using change property of it.
 - It is added into Ribbon using Method (addConceptualTaskGroup()).
 - It is shown by (DjVu.getRibbon.setVisible (Ribbon conceptual task Group and Boolean (is Visible))).
- Squiggly:
 - Its constructor takes three Parameters (Title, Color and Tasks).

- It Appears when need to add action using change property of it.
 - It is added into Ribbon using Method (addConceptualTaskGroup ()).
 - It is shown by (DjVu.getRibbon.setVisible (Ribbon conceptual task Group and Boolean (is Visible))).
- Replace:
 - Its constructor takes three Parameters (Title, Color and Tasks).
 - It Appears when need to add action using change property of it.
 - It is added into Ribbon using Method (addConceptualTaskGroup ()).
 - It is shown by (DjVu.getRibbon.setVisible (Ribbon conceptual task Group and Boolean (is Visible))).
- Insert:
 - Its constructor takes three Parameters (Title, Color and Tasks).
 - It Appears when need to add action using change property of it.
 - It is added into Ribbon using Method (addConceptualTaskGroup ()).
 - It is shown by (DjVu.getRibbon.setVisible (Ribbon conceptual task Group and Boolean (is Visible))).

3.13. Status Bar

- Package: com.lizardtech.djview.frame
- Class: StatusBar
- Method: getStatusBar that returns JPanel (Status Panel)
- At the Center of the panel:
- Page Nav:

Jpanel get from Class (PageSelect).

- First Page: it is a Jcommand Button and its constructor takes one parameter (Icon).
- Prev Page: it is a Jcommand Button and its constructor takes one parameter (Icon).
- Next Page: it is a Jcommand Button and its constructor takes one parameter (Icon).
- Last Page: it is a Jcommand Button and its constructor takes one parameter (Icon).
- Prev: it is a Jcommand Button and its constructor takes one parameter (Icon).
- Next: it is a Jcommand Button and its constructor takes one parameter (Icon).
- Book Mode Panel:
 - Package: com.lizardtech.djview.frame
 - Class: StatusBar
 - Single Page Mode:
 - Jcommand Button.
 - Its Constructor takes one parameter (Icon).
- Continuous Mode:
 - Jcommand Button.
 - Its Constructor takes one parameter (Icon).
- Facing Mode:
 - Jcommand Button.

- Its Constructor takes one parameter (Icon).
- Zooms:
 - Jcommand Button.
 - Its constructor takes one parameter (Name).
 - The Name changing depending on the chosen scale, using (setText) Method.
 - The type is changed using (SetCommandButtonKind).
 - The (SetCommandButtonKind) takes one parameter (JCommandButton.CommandButtonKind.PopUpOnly).
 - Display State Changed from (SetDisplayState) and take object from (CommandButtonDisplayState.Small).
 - It pops (JCommandPopUpMenu).
 - This Menu Contains
(z25,z50,z75,z125,z100,z200,z400,z600,z800,as[actualSize],fb,fw,fv).
- Z25:
 - It is a scale of Zooming.
 - It is a JcommandMenuButton.
 - Its constructor takes two parameters (Name and Icon).
- Z50:
 - It is a scale of Zooming.
 - It is a JcommandMenuButton.
 - Its constructor takes two parameters (Name and Icon).
- Z75:

- It is a scale of Zooming.
- It is a JcommandMenuButton.
- Its constructor takes two parameters (Name and Icon).
- Z100:
 - It is a scale of Zooming.
 - It is a JcommandMenuButton.
 - Its constructor takes two parameters (Name and Icon).
- Z125:
 - It is a scale of Zooming.
 - It is a JcommandMenuButton.
 - Its constructor takes two parameters (Name and Icon).
- Z200:
 - It is a scale of Zooming.
 - It is a JcommandMenuButton.
 - Its constructor takes two parameters (Name and Icon).
- Z400:
 - It is a scale of Zooming.
 - It is a JcommandMenuButton.
 - Its constructor takes two parameters (Name and Icon).
- Z600:
 - It is a scale of Zooming.
 - It is a JcommandMenuButton.

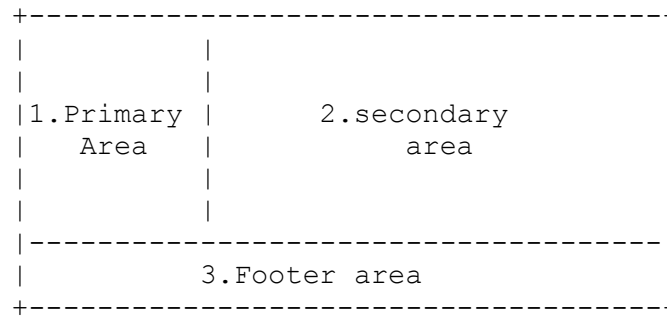
- Its constructor takes two parameters (Name and Icon).
- Z800:
 - It is a scale of Zooming.
 - It is a JcommandMenuButton.
 - Its constructor takes two parameters (Name and Icon).
- AS (Actual Size):
 - It is a scale of Zooming.
 - It is a JcommandMenuButton.
 - Its constructor takes two parameters (Name and Icon).
- FP (Fit Page):
 - It is a scale of Zooming.
 - It is a JcommandMenuButton.
 - Its constructor takes two parameters (Name and Icon).
- FW (Fit Width):
 - It is a scale of Zooming.
 - It is a JcommandMenuButton.
 - Its constructor takes two parameters (Name and Icon).
- FV (Fit Visible):
 - It is a scale of Zooming.
 - It is a JcommandMenuButton.
 - Its constructor takes two parameters (Name and Icon).

- Slider Panel:
 - Minus:
 - It is a Jcommand Button.
 - Its Constructor takes one parameter (Icon).
 - Plus:
 - It is a Jcommand Button.
 - Its Constructor takes one parameter (Icon).
 - JsldHort:
 - It is a JSlider.
 - Its orientation is horizontal.
 - It Determine the size using (setPreferedSize).
 - It takes one parameter (Name [type]).
 - Its dimensional Constructor takes two parameters (Height and Width).
 - It determine Maximum using (setMaximum).
 - It determine Minimum using (setMinimum).
 - Those Constructor takes one parameter.
 - They added onto the panel using the function (add ()).

3.14. DJVU++ Button

Metadata description of the application menu of the {@link JRibbon}

- Component. The ribbon application menu has three parts:



- The entries in the primary area are always visible.
- The secondary area entries are shown based on the currently active element in the primary area.
- There are three different types of primary entries:
 - Associated {@link ActionListener} passed to the constructor of the {@link RibbonApplicationMenuEntryPrimary}. When this entry is armed (with mouse rollover or via keyboard navigation), the contents of the secondary area are cleared. The menu item is an example of such a primary menu entry.
 - Associated {@link PrimaryRolloverCallback} set by the {@link RibbonApplicationMenuEntryPrimary#setRolloverCallback(PrimaryRolloverCallback)}. When this entry is armed (with mouse rollover or via keyboard navigation), the contents of the secondary area are populated by the application callback implementation of {@link PrimaryRolloverCallback#menuEntryActivated(javax.swing.JPanel)}.
 - The menu item is an example of such a primary menu entry, showing a list of recently opened files.
 - Associated list of {@link RibbonApplicationMenuEntrySecondary}s added with the {@link

`RibbonApplicationMenuEntryPrimary#addSecondaryMenuGroup`
(String, RibbonApplicationMenuEntrySecondary...)

API. When this entry is armed (with mouse rollover or via keyboard navigation), the secondary area shows menu buttons for the registered secondary menu entries. The menu item is an example of such a primary menu item, showing a list of default save formats.

- At runtime, the application menu entries are implemented as{@link JCommandMenuButton}, but the application code does not operate on that level. Instead, the application code creates metadata-driven description of the ribbon application menu, and that description is used to create and populate the "real" controls of the application menu popup.
- Note that once a {@link RibbonApplicationMenu} is set on the {@link JRibbon} with the {@link JRibbon#setApplicationMenu(RibbonApplicationMenu)}, its
- Contents cannot be changed. An {@link IllegalStateException} will be thrown from {@link #addMenuEntry (RibbonApplicationMenuEntryPrimary)} and
- {@link #addFooterEntry (RibbonApplicationMenuEntryFooter)}.
- Package: com.lizardtech.DjVuBean.RibbonMenu
- Class: RibbonMenuCreation
- Method: CreateAppMenuButton
- Creating Button Menu using (RibbonApplicationMenu).
- In code there is an object called appMenu from PopupOnly.
- PopupOnly:
 - It is a JRibbonApplicationPopup Panel.

- Its takes two objects of type (JRibbonApplicationMenuButton [take an object of type JRibbon] and RibbonApplicationButton).
- Identify an object called Exit (Close DjVu):
 - It is a RibbonApplicationMenuEntryFooter.
 - It takes three parameters (Name, Text and Action).
 - This action is a type of ActionListener.
- Create (RibbonApplicationMenuPrimary):
 - It takes four parameters (Name, Icon, Action and Type).
 - Its type is Popup Only.
 - It is determined by
(JCommandButton.CommandButtonKind.PopupOnly).
- In Creation, it pops up (appBlank) :
 - It is an object, its type is (RibbonApplicationMenuSecondary).
 - AppBlank pops up (JPopPanel).
 - JPopPanel contains Label and Jcommand Button.
 - F_Clip (Object of Secondary type).
- Open (RibbonApplicationMenuPrimary):
 - It takes four parameters (Name, Icon, Action and Type).
 - Its type is Popup Only.
 - It is determined by
(JCommandButton.CommandButtonKind.PopupOnly).
- Refile (Recent File or New Open):

- It is an object, its type is (RibbonApplicationMenuSecondary).
- AppBlank pops up (JPopPanel).
- JPopPanel contains Label and Jcommand Button.
- F_Clip (Object of Secondary type).
- Save (RibbonApplicationMenuPrimary):
 - It takes four parameters (Name, Icon, Action and Type).
 - Its type is Popup Only.
 - It is determined by
(JCommandButton.CommandButtonKind.ActionOnly).
- Save As:
 - It is RibbonApplicationMenuPrimary.
 - It takes four parameters (Name, Icon, Action and Type).
 - Its type is Popup Only.
 - It is determined by
(JCommandButton.CommandButtonKind.PopupOnly).
 - It pops (DjVu, AppPDF and Word):
 - DjVu:
 - It is an object, its type is (Action Only).
 - AppBlank pops up (JPopPanel).
 - JPopPanel contains Label and Jcommand Button.
 - F_Clip (Object of Secondary type).
- Print (RibbonApplicationMenuPrimary):
 - It takes four parameters (Name, Icon, Action and Type).
 - Its type is Popup Only.

- It is determined by
(JCommandButton.CommandButtonKind.ActionOnly).
- About DjVu (RibbonApplicationMenuPrimary):
 - It takes four parameters (Name, Icon, Action and Type).
 - Its type is Popup Only.
 - It is determined by
(JCommandButton.CommandButtonKind.ActionOnly).
- Exit [Close] (RibbonApplicationMenuPrimary):
 - It takes four parameters (Name, Icon, Action and Type).
 - Its type is Popup Only.
 - It is determined by
(JCommandButton.CommandButtonKind.ActionOnly).
- In Create:
 - Add secondary buttons using (addSecondaryMenuGroup).
 - It takes two parameters (Name of the group and RibbonApplicationMenuSecondaryButton).
 - To Add Create on the panel of AppMenu using (addMenuEntry).
 - It takes one parameter (object of type RibbonApplicationMenuPrimary).
 - Adding separator Between Primary Buttons using function (addMenuSeperator).
 - To make Description for Secondary Buttons use (setDescriptionText).

- Set Default call back for (RibbonApplicationMenu) using (setDefaultCallBack).
 - (setDefaultCallBack) take an object with type of (RibbonApplicationEntryPrimary.PrimaryRollOverCallBack).
 - RollOverCallBack: is an interface implements a method (MenuEnrtryActive).
- In Open:
 - Add secondary buttons using (addSecondaryMenuGroup).
 - It takes two parameters (Name of the group and RibbonApplicationMenuSecondaryButton).
 - To Add Open on the panel of AppMenu using (addMenuEntry).
 - It takes one parameter (object of type RibbonApplicationMenuPrimary).
 - Adding separator Between Primary Buttons using function (addMenuSeperator).
 - To make Description for Secondary Buttons use (setDescriptionText).
 - Set Default call back for (RibbonApplicationMenu) using (setDefaultCallBack).
 - (setDefaultCallBack) take an object with type of (RibbonApplicationEntryPrimary.PrimaryRollOverCallBack).
 - RollOverCallBack: is an interface implements a method (MenuEnrtryActive).
 - In Save as:
 - Add secondary buttons using (addSecondaryMenuGroup).

- It takes two parameters (Name of the group and RibbonApplicationMenuSecondaryButton).
 - To Add Save as on the panel of AppMenu using (addMenuEntry).
 - It takes one parameter (object of type RibbonApplicationMenuPrimary).
 - Adding separator Between Primary Buttons using function (addMenuSeperator).
 - To make Description for Secondary Buttons use (setDescriptionText).
 - Set Default call back for (RibbonApplicationMenu) using (setDefaultCallBack).
 - (setDefaultCallBack) take an object with type of (RibbonApplicationEntryPrimary.PrimaryRollOverCallBack).
 - RollOverCallBack: is an interface implements a method (MenuEnrtryActive).
- In (DeafultCallBack) page: it contains:
 - About us: it is a Jcommand Button.
 - Contact us: it is a Jcommand Button.
 - Help: it is a Jcommand Button.
 - To add icon for (RibbonApplicationMenu) use (setApplicationIcon) it takes one parameter (Icon).
 - To add it on Ribbon use (DjVu.getRibbon.setApplicationMenu) it takes an object (RibbonApplicationMenu) that takes (appMenu).

3.15. Help Icon:

- It is created using Function (ConfigureHelp) and it takes two parameters (Name and Icon).

3.16. Put Bands into Tasks:

- Home (THome):
 - Package: com.lizardtech.djview.frame
 - Class: RibbonMenuCreation
 - Method: SetRibbonMenu
 - It is identified by Class (RibbonTask).
 - It takes two Parameters (Home and Array of Bands).
 - The Array of bands Contains (getHometoolsband, getHomeViewBand, getHomecommentband and getHomeconvertband).
- Convert (Task Convert):
 - Package: com.lizardtech.djview.frame
 - Class: RibbonMenuCreation
 - Method: SetRibbonMenu
 - It is identified by Class (RibbonTask).
 - It takes two Parameters (Convert and Array of Bands).
 - The Array of bands Contains (getHometoolsband and getHomeconvertband).
- Edit (Task Edit):
 - Package: com.lizardtech.djview.frame
 - Class: RibbonMenuCreation
 - Method: SetRibbonMenu
 - It is identified by Class (RibbonTask).
 - It takes two Parameters (Edit and Array of Bands).

- The Array of bands Contains (getToolForEditBand, getFontBand and getParagraphBand).
- Comment (Task Comment):
 - Package: com.lizardtech.djview.frame
 - Class: RibbonMenuCreation
 - Method: SetRibbonMenu
 - It is identified by Class (RibbonTask).
 - It takes two Parameters (Comment and Array of Bands).
 - The Array of bands Contains (getToolForCommentBand and getDrawingBand).
- View (Task View):
 - Package: com.lizardtech.djview.frame
 - Class: RibbonMenuCreation
 - Method: SetRibbonMenu
 - It is identified by Class (RibbonTask).
 - It takes two Parameters (View and Array of Bands).
 - The Array of bands Contains (getToolForViewBand, getGoToBand, getPageDisplay, getViewSetting).

3.17. Put Tasks into Ribbon Menu:

- Package: com.lizardtech.djview.frame
- Class: RibbonMenuCreation
- Method: setRibbonMenu
- Ribbon Menu:
 - It is put on JRibbon Frame.

- It always be in North (Ribbon).
- Function (getRibbon) is used to get this Ribbon.
- The tasks are added in it by JRibbonFrame.getRibbon.addTask()
- In Code:

DjVu.getRibbon.addTask ()

and take one parameter (The name of any task), note that: DjVu is the Ribbon Frame.

3.18. Show Ribbon in DjVu++ Frame:

- Package: com.lizardtech.djview.frame
- Class: DjVuStart
- Method: initGUI
- Then call for (setRibbonMenu) take an object (object from DjVuStart).

3.19. Show Status bar in DjVu++ Frame:

- Package: com.lizardtech.djview.frame
- Class: DjVuStart
- Method: initGUI
- Then call for (add (StatusPanel, BorderLayout.South)).

3.20. Tabs:

It uses one java archives (jar):

- JchromeTab

3.20.1. Creating of tabs:

- Package: com.lizardtech.djview.frame
- Class: DjVuStart that implemented from a lot of interfaces:

- ISexyTabsDemo: it has one method named as (Start) That initialize main window of the application
 - ITabbedPaneWindow: it have two methods: Method from type (JTabbedPane) named as (GetTabbedPane) that return tab on the application.
And the other method from type (Window) named as (GetWindow) That return main window of application.
 - ITabbedPaneWindowFactory: it have one method from type (ITabbedPaneWindow) named as (CreateWindow) that create main window of application.
 - ITabFactory: it have two methods: Method from type (Tab) named as (CreateTab) that create tabs individually and the other Method from type (Tab) named as (CreateTabWithContent) that put contents in the opening tab.
- The previous methods we implemented it on the (DjVuStart).
 - Opening Book or a lot of Books from Arguments:
 - If we open any book with extension (.DjVu) from computer without opening DjVu program, we should select it and pressed on it, after that the program is opened and tab is created for that book.
 - We take URL of some of books in array Called (Args) in main of DjVuStart and take each URL of these books from this array to open it in program in Tabs.

3.21. OCR

Using Tesseract we implement the following in the program

- Package: OCR
- Class: OCROnTiFImage
- Using new Process:

Process p = Runtime.getRuntime.exec ();

- It have three Parameters:
- Dir path of tesseract.exe
- Image to do OCR by select mode:
- Package: com.lizardtech.DjVubean
- Class: SelectMode

Draw rectangle to crop image and save on disk.

- File to output:

Implement in DjVu by button on Ribbon Menu.

- Package: com.lizardtech.DjVubean.RibbonMenu
- Class: DjVuRibbonComponents – JCommandButton Ocr – JcommandMenuButton OCREnglish – JcommandMenuButton OCRArabic (they doing action on the class)
- OCREnglish:
- Method:
 - getHomeconvertband ()
 - its type is JRibbonBand

3.22. Print

- Package: Printer

- Class:

- PrintDjVu ()

Using this class to select pages you want to print by (drawingimage ())

- Method: print (Graphics, Page Format, Page Index)

Use method print to get the pages you want to print by array of image.

- GetBook ()
- Method: GetBookImage ()

Use it to read book on the current panel and get the image of the book stored on array of image.

- Method: getPrintBookImage ()

Return array of pages with current open book.

- PrintBook ()

This is the class of the GUI for Printing.

Use object from PrinterJob job = PrinterJob.getPrinterJob () to set the Printer.

Use PrintRequestAttributeSet asset = new HashPrintRequestAttributeSet () to get the properties of page (font, quality and position).

And put it in page format PageFormat pf = job.getPageFormat (aset);

job.setPrintable (new PrintDjVu (), pf); to print page with the proprieties
Selected.

- Implement the printer in DjVu using Button:
- Package: com.lizardtech.DjVubean.RibbonMenu
- Class: RibbonMenuCreation

JCommandButton TBprint;

- Do an action on TBprint to print book by PrinterBook.PrintBook()

3.23. Outline Tabbed Panel:

- Package: com.lizardtech.DjVubean.outline
- Class: Outlinetabbedpane
- Methods:

- createPagesTab ():

Creates panel with the tree of the book outline then add it to the
tabbedpanel.

- createThumblainsTab():

Creates a panel with a thumbnails of all book pages then add it to the
tabbedpanel.

- `createCommentTab()`:

Creates a panel to add comments per page in a tab added within a tab.

- `createTabs()`:

Adds the tabs to the tabbedpanel.

- `RemoveTabs ()`:

Remove the tabs from the tabbedpanel.

- `GettopPanel ()`:

Return the Top Panel which contains the tapped panel.

- `switchFlag ()`:

Reflect the Boolean value.

- Package: `com.lizardtech.DjVubean.outline`
- Class: `ImagelistCellRender`
- Methods:

Has no methods it's just an adaptive overridden class to match the cell rendering of thumbnails in jlist.

- Package: `com.lizardtech.DjVubean.outline`
- Class: `createthumbnails`

- Methods:

- `getScreenShot (component):`

Takes a screenshot image of the component and return it as buffered image.

- `Resizeimage ():`

Takes an image with the needed width and height and return a image with the needed size.

- `getCroppedImage ():`

Corps an image of the boundaries noise.

- `generateThumbnail ():`

Generates an image of a certain page and returns it as buffered image.

- `switchFlag():`

Reflect the Boolean value.

- `RotateImage (buffered image , degree):`

Rotate image with corresponding degree and returns a buffered image.

3.24. Full Book continuous View:

- Package: com.lizardtech.djview
- Class: FullBookView
- Methods:
 - `getTopPanel()`:

Return the Top Panel which contains the full book pages in continuous view.

- `createImages ()`:

Renders the book pages into one panel.

`Switch_FullBookView ()`:

Switches between the DjVuBean panel and the full view panel.

- `Rotate (degree)`:

Rotate the book pages with the given degree

4. System Architecture

The main window

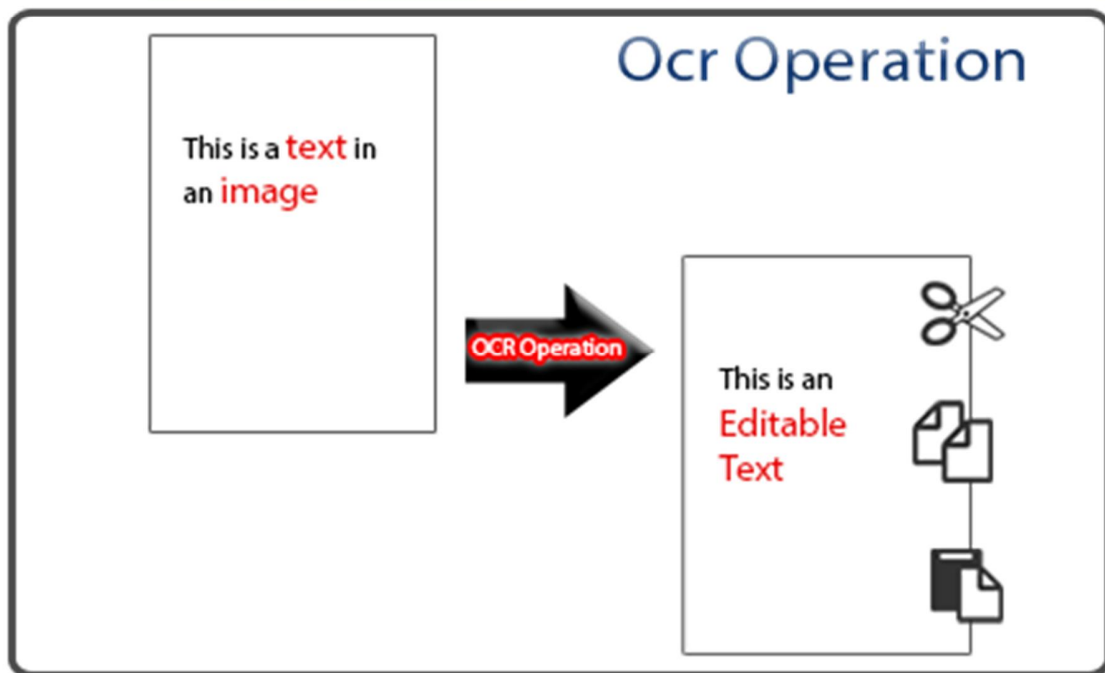
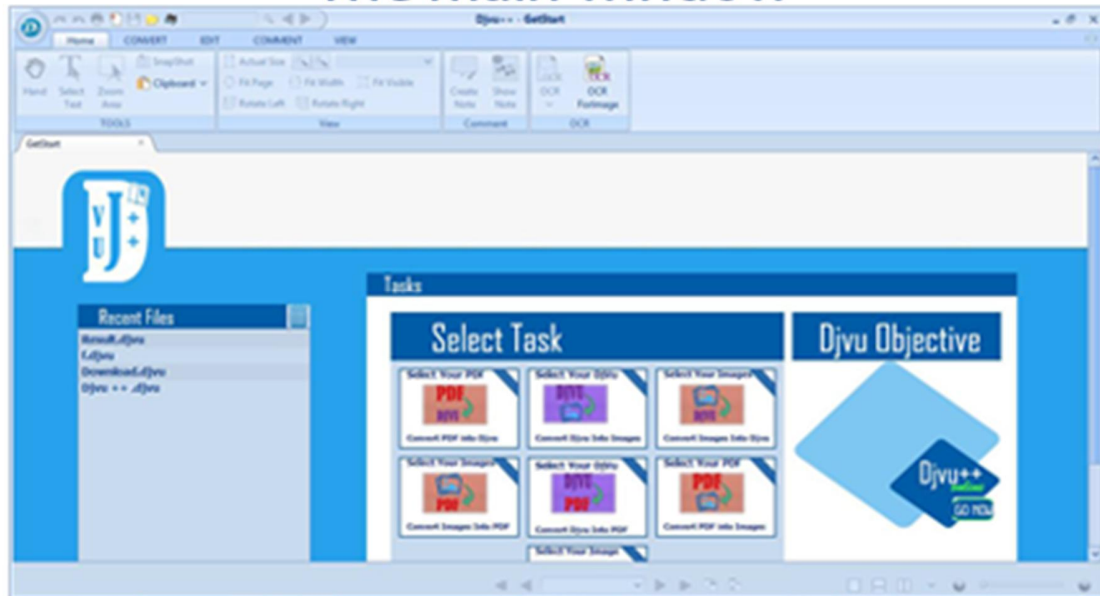


Figure 5-1: System components

4.1. System services

To acquire most of the benefits of this system we divide the desired services into two groups:

- View DJVU and PDF files.
- Annotations.
- OCR: English and Arabic.
- PDF to DJVU and DJVU to PDF.
- Combine images to a DJVU file.

This make the application more easy and powerful.

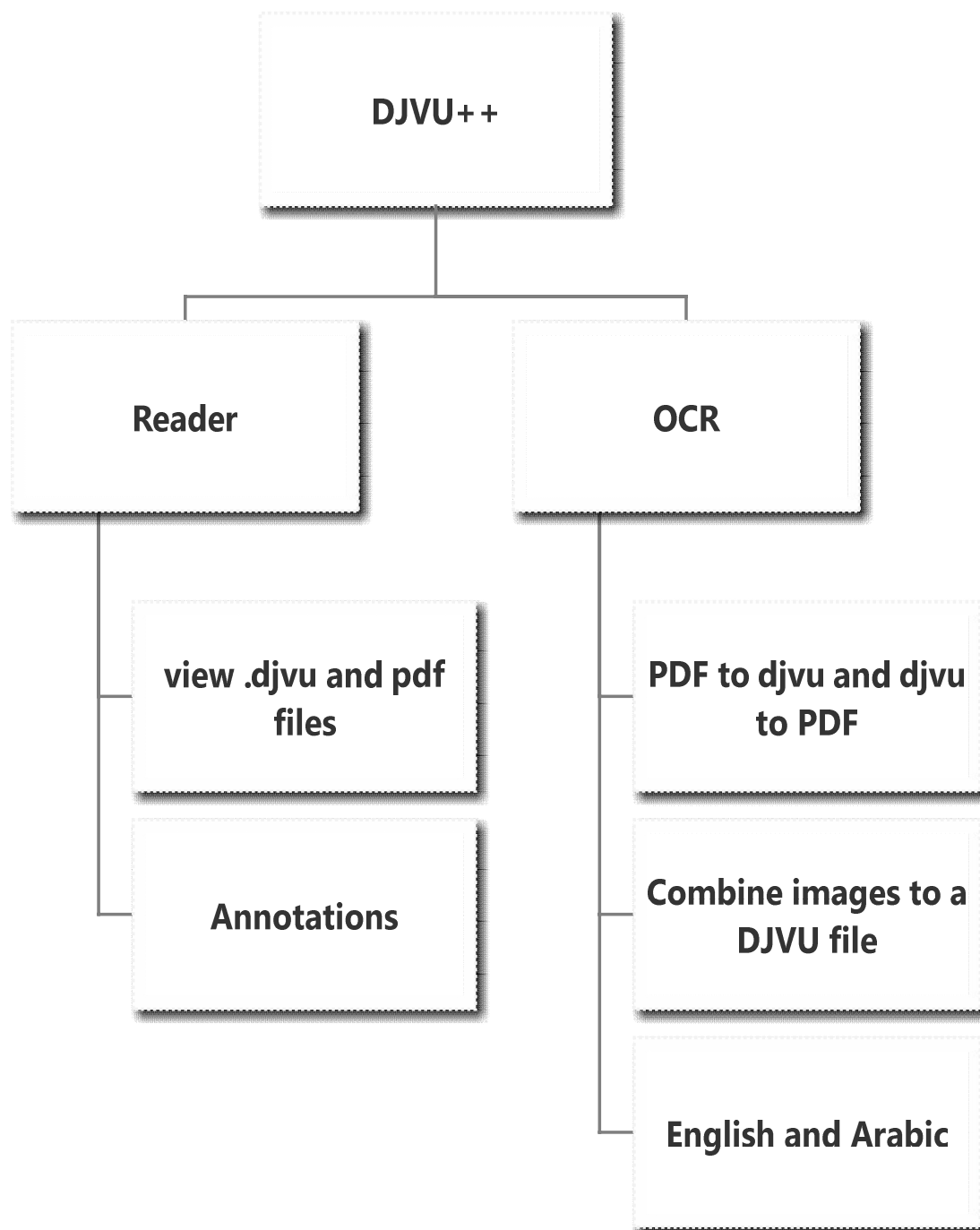


Figure 5-2: System Services

5. Technical implementation

5.1. DJVU++:

DjVu++ includes the following services: read, copy, cut, delete and make annotations ...etc.

5.1.1. Implementation:

The DJVU++ contains a Great GUI that make the user interact with this software easily and can read any kind of books. It also make the user able to edit in the text books he read.

- Book Outline: A tab in outline bar that shows the book outline.
- Book Outline Thumbnails: A tab in outline bar that shows the book pages in thumbnails.
- Create classic menu bar: Create all classic menu bar items for the DjVu++ reader contains these main menus (File, Edit, View, Tools, and Help).
- Simple action in classic menu: Implement simple action in the classic menu.
- Link between classic menu and ribbon menu: Create small frame that's open the DjVu++ reader with ribbon menu or classic menu.
- Implement the actions: Implement all actions that will be in classic and ribbon menus.
- MaaDesign ribbon menu: add it into the program and do action and implementation on it.

- J chrome tab: design tabbedpanel in the DjVu like tabs in Google chrome.
- Store bean: each tab has a bean (information of the current book).
- Status Bar: Book navigation, zooming and slider.
- Start Page: Group of converters (some actions to convert, Example: DjVu book to PDF book).
- Recent File: Store the opening book in a list, and the user can open that book from the recent book by its name.
- Program Argument (Args): open book from args without open the program window.

5.1.2. Supplementary resources

We have a ready good start with an open source called [Java DjVu Viewer \[7\]](#)

5.2. OCR

5.2.1. Implementation:

It is a technique in the DJVU++ that make it possible to edit text on images either Arabic or English text.

- Implement OCR English on various type OF images: Extract English text from image. (Aspose, ocr java and tesseract)
- Implement OCR Arabic on various type OF images: extract Arabic text from image.
- Image processing: convert any image, extract image from pdf, convert image to pdf.
- Edit on OCR:
 - Snapshot for image and do OCR about it.
 - Crop the region of image and do OCR about it.
 - OCR on All and Current Page For DjVu++.
- Conversion Module:
 - Convert DjVu to PDF: Convert using exe command line program.
 - Convert PDF to DjVu: Convert using DjVuLibre exe command line and dll program.
 - Convert DjVu to Tif: Convert using DjVuLibre exe command line and dll program
- GUI OCR Module:
 - implement OCR Arabic on TIF images :

Using tesarct exe version3.03 exe command line with Arabic Lang.

- implement OCR English ON tif images :

Using tesarct exe version3.03 exe command line with English Lang.

- implement Recent Books in start page in DjVu++ GUI :

Open recent books that open from users later.

- implement Crop on DjVu current page OCR :

Show notification for current crop image and can OCR on it.

- Implement OCR on All and Current Page for DjVu++.
- Apply Ocr Arabic & English on sample page of DjVu Desc :

Using tesseract for extracting text from DjVu image and show that text in the same place or at least show text in another area beside original image.

5.2.2. Supplementary resources:

We have a ready good start with an open source library called [Tesseract](#)

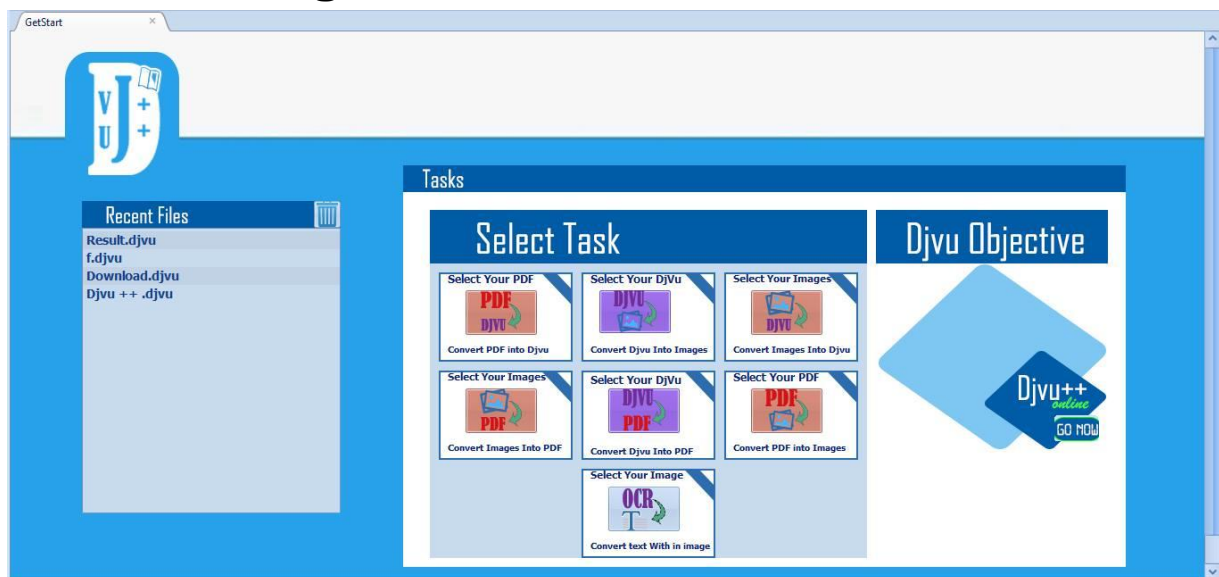
[8]

6. User Guide

6.1. Inception

DjVu++ contains some components that help to read eBooks.

6.2. Start Page



- Recent Files:

It is used to store the last opened books in the program.

- Select Task:

It is used to select a convert style to apply it on the book.

- DjVu++ online:

It is used to go online to know more about the DjVu++ Program.

6.3. Top Bar



In this bar there are some shortcuts that can help the reader.

- Redo:

It is used to go forward from an action that done and have an undo action.

- Undo:

It is used to go back from an action that just done.

- Print:

It is used to print a document using the printer.

- Blank:

It is used to create a new blank document.

- Save:

It is used to save any changes have done on the opening book.

- Open:

It is used to open a book from the drive.

- Search area:

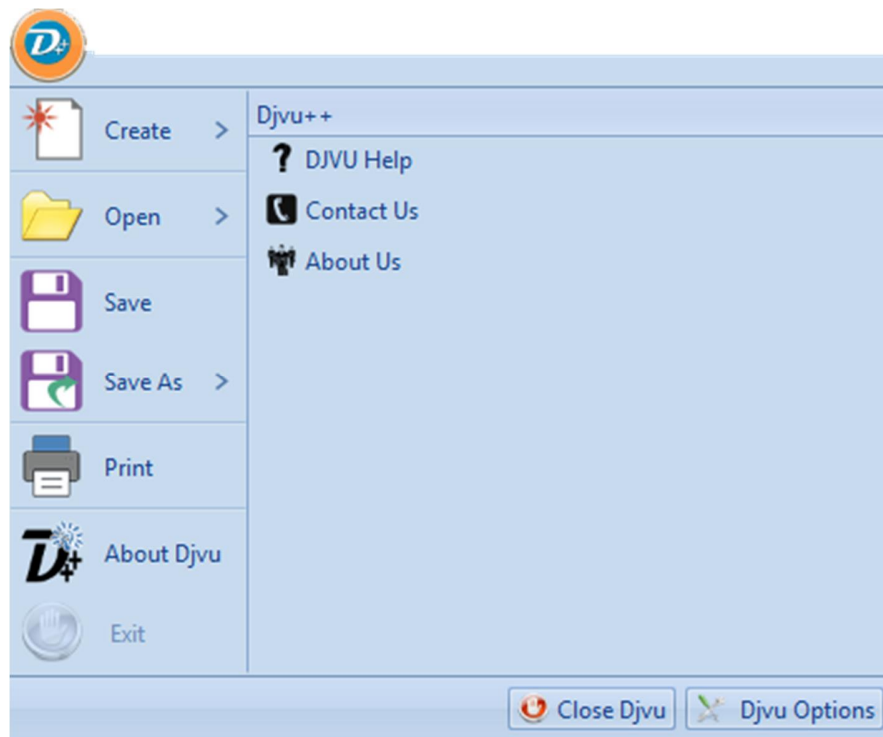
It is used to search about any word in the whole document.

- Book's name area:

In this the opening book's name appears.

- Minimize to task bar.
- Minimize and maximize window.
- Exit.

6.4. DJVU++ Button



- Create:

It is used to create a new blank document.

- Open:

It is used to open document from the drive.

- Save:

It is used to save any changes that done on the opening book with the same name.

- Save as:

It is used to save with a new name.

- Print:

It is used to print the document.

- Exit:

It is used to close the program.

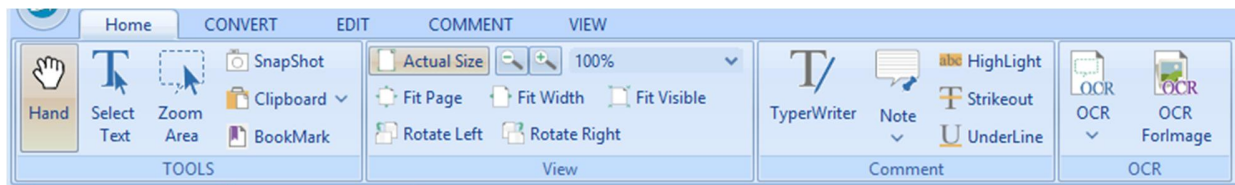
- Close all:

It is used to close all the opening tabs.

- DJVU++ options:

Make you can change the setting of the program to be easy to use.

6.5. Home



It have:

- Hand Tool:

It used to move the pages up and down.

- Select Text:

It is used to select some text from the page.

- Select Annotation:

It is used to select and make an annotation on any area of the page.

- Snap Shot:

It is used to take a picture of an area of the page.

- Clipboard:

It is used to make copy, cut and paste a text.

- Bookmark:

It is used to make a mark on some place and name it with a name.

- Actual size:

It is used to view pages in actual size.

- Fit page:

It is used to fit the page with width and height.

- Fit width:

It is used to fit the page with width.

- Fit visible:

It is used to fit the page to view the text.

- Rotate left:

It is used to rotate page to the left side.

- Rotate right:

It is used to rotate page to the right side.

- Zoom in and zoom out:

It is used to make page smaller or larger.

- Edit text:

It is used to make any edit in the text selection.

- Edit object:

It is used to edit an object in the page.

- Typewriter:

It is used to write text in any place in the page.

- Note:

It is used to make any note in the page.

- Highlight:

It is used to make colored background to the text.

- Strikeout:

It is used to put line on some words.

- Underline:

It is used to put line under some words.

- Rotate pages:

It is used to choose one of rotate actions.

- Insert:

It is used to insert an object in the page.

- Delete:

It is used to delete an object from the page.

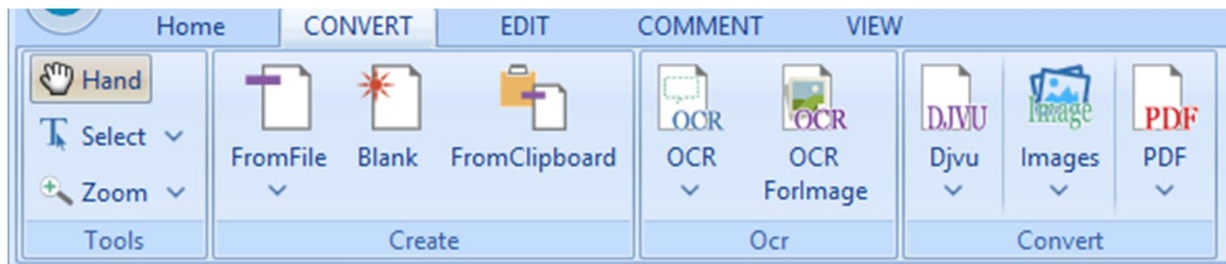
- OCR:

It is used to choose between Arabic and English to do the OCR.

- DJVU Converts:

It is used to choose between PDF and DJVU to convert the document.

6.6. Convert



- Hand Tool:

It used to move the pages up and down.

- Select Text:

It is used to select some text from the page.

- Zoom in and zoom out:

It is used to make page smaller or larger.

- From file:

It is used to get a text from file.

- Blank:

It is used to get new blank.

- From clipboard:

It is used to get anything from clipboard.

- OCR:

It is used to choose between Arabic and English to do the OCR.

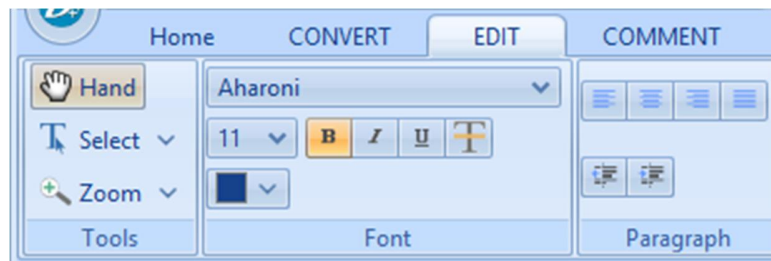
- DJVU Converts:

It is used to choose between PDF and DJVU to convert the document.

- PDF converts:

It is used to convert DJVU to PDF document.

6.7. Edit



- Hand Tool:

It used to move the pages up and down.

- Select Text:

It is used to select some text from the page.

- Zoom in and zoom out:

It is used to make page smaller or larger.

- Edit text:

It is used to edit in text.

- Edit object:

It is used to edit in object.

- Join & spilt text:

It is used to join and spilt in text.

- Add text:

It is used to add new text in the original text.

- Add image:

It is used to add an image in the text.

- Add shapes:

It is used to add shape in the text.

- B:

It is used to make line bold.

- I:

It is used to make line italic.

- Underline:

It is used to put line under the word.

- Strike out:

It is used to put line above word or text.

- Font face area:

It is used to select the name of the writing.

- Font size area:

It is used to select font size in writing.

- Font color area:

It is used to select font color in writing.

- Align text left:

It is used to align text to left.

- Align text right:

It is used to align text to right.

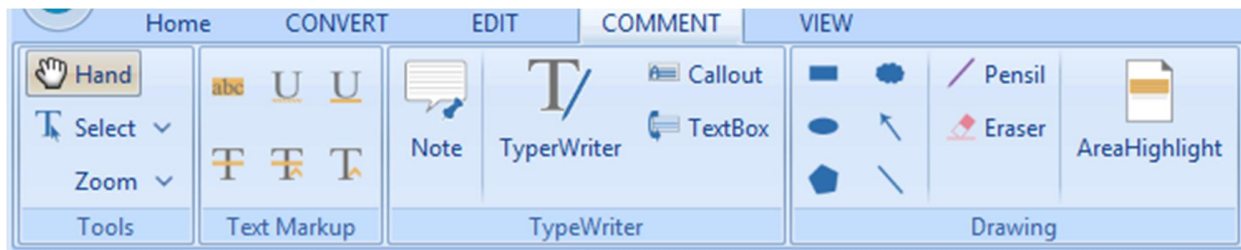
- Center:

It is used to put text in the center of the page.

- Justify:

It is used to align text to left and right.

6.8. Comment



- Hand Tool:

It used to move the pages up and down.

- Select Text:

It is used to select some text from the page.

- Zoom in and zoom out:

It is used to make page smaller or larger.

- Highlight:

It is used to put a highlight on the words as a mark.

- Underline:

It is used to put a line under the words.

- Strikeout:

It is used to put a line on the words.

- Note:

It is used to put a note behind a specific line you want.

- Type Writer:

It is used to add text in the book you read.

- Drawing:

It is used to draw a shape on the words.

- Pencil:

It is used to write as you use a hand pencil.

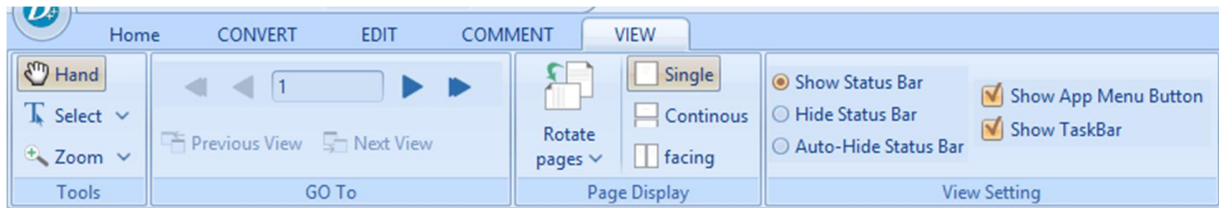
- Eraser:

It is used to delete the drawing.

- Area Highlight:

It is used to put a highlight on a whole area of lines.

6.9. View



- Hand Tool:

It used to move the pages up and down.

- Select Text:

It is used to select some text from the page.

- Zoom in and zoom out:

It is used to make page smaller or larger.

- Previous View:

It is used to preview the previous page.

- Next View:

It is used to preview the next page.

- Rotate Pages:

It is used to rotate pages (90,-90,180 and -180).

- Single:

It is used to show a single page in the opened tab.

- Continues:

It is used to show the pages continuously.

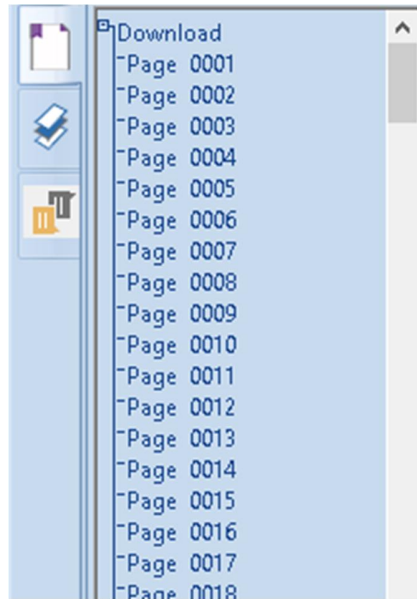
- Facing:

It is used to preview two pages in the opened tab.

- View Setting:

It is used to show some properties or hidden it using (radio buttons and checkbox).

6.10. Out Line



- Book Mark:

It is used to put a bookmark in the outline for a specific page as a mark.

- Multi Layers:

It is used to preview the pages in the outline.

- Comments:

It is used to put comments on pages.

6.11. Thumbnails



- Book Mark:

It is used to put a bookmark in the outline for a specific page as a mark.

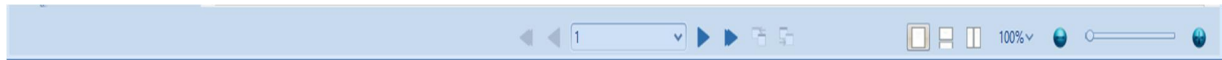
- Multi Layers:

It is used to preview the pages in the outline.

- Comments:

It is used to put comments on pages.

6.12. Status Bar



- First previous:

It is used to get the first previous page.

- Previous:

It is used to get the previous page.

- Combo box:

It is used to get the number of the page.

- Next:

It is used to get the next page.

- First next:

It is used to get the first next page.

- View first:

It is used to view the first page.

- View previous:

It is used to view the previous page.

- Single page:

It is used to select single page.

- Continuous:

It is used to continuous the text.

- Facing:

It is used to open in two pages in the screen.

- Zoom in:

It is used to small the text.

- Zoom out:

It is used to large the text.

7. Conclusion & Future Work

7.1. inception

We create a user-friendly system to read and edit eBooks with a penalty of features.

DjVu++ also support OCR to handle text in scanned books and images. The program show good performance for English. In addition to the Arabic language to lead the free and commercial software in this area.

The main features of DjVu++ program are:

- Manipulate DjVu files.
- Support smaller size than PDF with the same performance.
- DjVu++ supports two languages in the OCR technique (Arabic and English).
- Read multiple documents at the same time with the new tabs feature.
- DjVu++ supports multiple formats:
 - Convert PDF document into DjVu format with smaller file size and the same performance.
 - Convert DjVu into PDF format.
 - Combine images to a single DjVu document.
 - Perform OCR operations on multiple image formats (png, jpg, bmp and tif).

7.2. Future Work

- Enhancing Design:
 - We worked on the classic menu but it is not added to the program so you have to work on the Classic Menu to add it with the ribbon menu to enhance and improve the interface of the program.
 - Improve the interface design by make a new version of the ribbon menu and the icons in the whole program, you can also change the style of the start page to be more attractive.
- Enhancing some tools code:
 - Enhance comment on DjVu++ document as we added comment on DjVu++ document but from storing its data on text file you will see x.txt file in your work space, this file stores document name and page number that Comment Find It. The Solution for this problem is: we already do more Search about Comment on document and we found a solution to insert data for Comment Using Meta data of DjVu++ document.
 - Apply insert and delete text on DjVu++ document: as we wanted to insert text over document like the updated Version of Foxit Reader. The solution for this problem is: from Meta data you can insert and delete text from document (store text position, page number that user wrote a text in it).
 - Annotations: we wanted to add annotation over DjVu++ document to make user select easily from text and highlight the text. The solution for this problem is: also Meta data (store

annotation's position and page number that the user has Draw in it and using the highlight color.

- Draw line, square, rectangle and save it on DjVu++ document: we wanted to add Rectangle over DjVu++ document and Draw multi shapes. The solution for this problem is: also Meta data.

- In Developing you must begin with:

Understanding the Meta data of DjVu++ document to be able to make comment and annotations.

- Making change on DjVu++ document and save it: if the user draw Rectangle and close the document, when the user Open it again, the user must see the Rectangle that he has drew in a specific position.

- In OCR:

You need to apply preprocessing operation on images before using tesseract to apply OCR, this operation can improve the chances of successful recognition using some techniques:

- De-skew:

If the document was not aligned properly when scanned, it may need to be tilted a few degrees clockwise or counterclockwise in order to make lines of text perfectly horizontal or vertical.

- Despeckle:

Remove positive and negative spots and smoothing edges.

- Binarization:

Convert an image from color or grayscale to black-and-white (called a binary image because there are two colors). The task of Binarization is performed as a simple way of separating the text (or any other desired image component) from the background. The task of Binarization itself is necessary since most commercial recognition algorithms work only on binary images since it proves to be simpler to do so.

- Line removal:

Cleans up non-glyph boxes and lines.

- Layout analysis (Zoning):

Identifies columns, paragraphs, captions, etc. as distinct blocks, especially important in multi-column layouts and tables.

- After apply character recognition on image user will have the ability to update words on image and save it with its update.

References

- [1] Djvu, "Djvu: A Tutorial," [Online]. Available: <http://www.djvuzone.org/support/tutorial/chapter-intro.html>.
- [2] Djvu, "Djvu: Applications," [Online]. Available: <http://www.djvuzone.org/demos/appli/>.
- [3] n. PDF, "Convert Djvu to PDF," [Online]. Available: <http://www.novapdf.com/kb/convert-djvu-to-pdf-215.html>.
- [4] S. Kardam, "5 free Djviewer Applications," [Online]. Available: <http://www.ilovefreesoftware.com/05/featured/5-free-djvu-viewer-applications.html>.
- [5] A. Zhezherun, "WinDjView: Home," [Online]. Available: <http://windjview.sourceforge.net/>.
- [6] G. Arghire, "WinDjView," [Online]. Available: <http://www.softpedia.com/get/Office-tools/Other-Office-Tools/WinDjView.shtml>.
- [7] profshadoko, "Java Djvu Viewer," [Online]. Available: <http://sourceforge.net/projects/javadjvu/files/Java%20DjVu%20Viewer/>.
- [8] theraysmith, "Tesseract," [Online]. Available: <http://sourceforge.net/projects/tesseract-ocr/>.