NIC/IISD/Website/Accessibility/2020/1
15695/2020/INDUSTRY INFORMATICS DIVISION

PROCESS & OBSERVATIONS ON ACCESSIBILITY OF PDF DOCUMENTS ON INDIAN GOVERNMENT WEBSITES

STANDARD OPERATING PROCEDURE

NIC-IID-PDF-ACC-2020-01

August 2020

Version 2.0

INDUSTRY INFORMATICS DIVISION
NATTIONAL INFORMATICS CENTRE
MINISTRY OF ELECTRONICS & INFORMATION TECHNOLOGY
GOVT. OF INDIA

Disclaimer

Adobe Acrobat professional, Microsoft Word and (Libre Office Writer Open Source Software), Tesseract, Imagemagick, PAC3 and FreeOCR are copy right of respective Owners / Open Source Community. The users of templates indemnify NIC for their use and are requested to comply with licensing of the Propriety and OSS tools used. In no event will the Government or NIC be liable for any expense, loss or damage including, without limitation, indirect or consequential loss or damage, or any expense, loss or damage whatsoever arising from use, or loss of use, of data, arising out of or in connection with the use of these templates. The information provided in this document are/is as basis without warranty.

NOTICE: This document is received in confidence and its contents cannot be disclosed or copied without the prior written consent of National Informatics Centre (NIC). Nothing in this document constitutes a guaranty, warranty, or license, express or implied. NIC disclaims all liability for all such guaranties, warranties, and licenses, including but not limited to: Fitness for a particular purpose; merchant-ability; not infringement of intellectual property or other rights of any third party or of NIC; indemnity; and all others. The reader is advised that third parties can have intellectual property rights that can be relevant to this document and the technologies discussed herein, and is advised to seek the advice of competent legal counsel, without obligation of NIC. NIC retains the right to make changes to this document at any time, without notice. NIC makes no warranty for the use of this document and assumes no responsibility for any errors that can appear in the document nor does it make a commitment to update the information contained herein.

Amendment Log

Version	Release Date	Description	Section(s) Modified	Prepared By	Reviewed by	Approved By
0.1	-	Process & Observations on Accessibility of PDF Documents on Indian Government Websites	All Sections	Yatindra Saxena , Scientist-F	R Vijay Raghavan, Scientist-F Girish Chandra (Scientist-F & HoD)	-
0.2	-	Process & Observations on Accessibility of PDF Documents on Indian Government Websites	All Sections	Yatindra Saxena , Scientist-F	R Vijay Raghavan, Scientist-F Girish Chandra (Scientist-F & HoD) Narinder Singh Arneja (Scientist-G & HoG)	-

Table of Contents

Foreword

Congratulations! To Mr. Yatindra Saxena, Scientist-F and Mr. R Vijay Raghavan, Scientist-F of NIC, Industry Informatics Division for making extensive research on making accessible PDF documents available on the Indian Government websites as per the mandatory Policy given in the Guidelines for Indian Government Websites issued in 2009 (Web Content Accessibility Guidelines (WCAG) 2.0.)

This report describes in detail the process/tools available for making accessible documents, limitations of various tools and how to achieve 100% compliant accessible documents.

NARINDER SINGH ARNEJA
Deputy Director General, NIC

Preface

Why Accessible document? As per the Govt. of India Guidelines for Indian government Websites, Web Content Accessibility Guidelines WCAG 2.0 are the mandatory policy documents all Indian Govt. website needs to be compliant.

After making lot of research on the various Govt. Ministries/Departments/Attached offices websites both at Central level and at State Level, it is noticed that only 5-10% documents are in accessible format. The reasons for this are summarized below:

a. Lack of technical knowledge to the user departments.
b. Casual approach in publishing of documents
c. Efforts and time required in making accessible documents.
d. Old documents published only in image format requires lot of efforts/time and money to make them compliant accessible documents.
e. Lack of will of the user organization.
f. Govt. of India Guidelines for Indian Govt. websites is not being fully followed.

This document has been written to educate the user on making accessible documents using various open Source/propriety tools, Issues/challenges and efforts required in this activity.

The activity was started with the letter on the subject from the MeitY to all the Administrative Secretaries of the Govt. of India. We noticed that even the documents being published on the DPIIT website are not accessible. The matter was discussed with my senior NIC colleagues and Shri N.S. Arneja, DDG & HOG. It was decided to find out the necessary tools required to publish all documents in the accessible format as per the mandate given.

During the period of study, we discussed the subject with many domain experts in NIC, officers at Open technology Centre, NIC Chennai and got lot of useful input. By using those inputs and performing actual runs of various tools on the different types of documents, we have come up with this document for the use and benefit of all the users who wish to publish accessible documents on their website.

This document shall serve as our contribution to society and especially to the visual challenged persons who would like to visit Government websites to know the different policies, acts, orders

and notification etc. prevailing in different Government Organizations. During recent times, many new initiatives such as Skill India, Digital India, Make in India, Start-up India etc. have been launched by the Government of India and most of the contents related to such schemes have been published in PDF document on different Government Website. An accessible PDF document shall help those intellects, who in spite of being visually challenged can contribute by availing and contributing in such schemes and for that they should have a mechanism to understand what is written in such documents.

However, we were working on different aspects of accessibility of PDF file during the start of the year 2020 but momentum was gained after July 2020 when we received comments from various domain expert groups of NIC and then this document was authored with practical example illustrated in it. While authoring this document, we initially faced problems when practically demonstrating the accessibility of PDF using different tools mentioned in the report but the help from online documentation really helped to complete it. We feel that the steps and techniques to make PDF document accessible shall look easy once content managers will start working on the accessibility guidelines on their initial website documents to make them accessible.

Content managers of different website in Government domain may consult this document to make pdf documents published on their respective website accessible to visually challenged persons.

Acknowledgement

I would like to express my sincere thanks to my NIC colleagues Mr. Pramod Kumar, Mr. Anil Kumar Awasthi, Mr. Mohinder Kumar and Mrs. Anju Sondhi who have always been supportive to me while authoring this document.

I am highly thankful to Mr. Piyush Chandra Dubey (Sr. Programmer) from M/s Velocis who helped me a lot during testing of tools mentioned in this document.

I would also like to extend my gratitude to Mr. Girish Chandra, Scientist-F & HoD who persuaded to me to work on accessibility of documents.

I would express my deep and since gratitude to Mr. Narinder Singh Arneja, Scientist-G & HoG for providing me invaluable guidance to work on this document. His dynamism, vision and motivation have deeply inspired me.

Finally, I would like to acknowledge with gratitude the support & love of my family. They all kept me going and this work would not have been possible without their support.

Yatindra Saxena
Scientist-F

Abstract

The Process & Observations on Accessibility of PDF Documents on Indian Government Websites is offered to create accessible PDF file for websites and to convert existing PDF Documents available on all Indian Government Websites to an accessible format.

WCAG 2.0: Web Content Accessibility Guidelines (WCAG) 2.0 covers a wide range of recommendations for making Web content more accessible. Following these guidelines will make content accessible to a wider range of people with disabilities, including blindness and low vision, deafness and hearing loss, learning disabilities, cognitive limitations, limited movement, speech disabilities, photosensitivity and combinations of these. Following these guidelines will also often make your Web content more usable to users in general.

WCAG 2.0 success criteria is written as testable statements that are not technology-specific. Guidance about satisfying the success criteria in specific technologies, as well as general information about interpreting the success criteria, is provided in separate documents. See Web Content Accessibility Guidelines (WCAG) Overview for an introduction and links to WCAG technical and educational material.

For more information on these, please read PDF Techniques for WCAG 2.0 .

Intended Audience

All Indian Government Website Stakeholders including Web Information Managers, Hired Resources, and Operational Staff & Content Managers who intend to manage PDF documents on respective websites.

Prerequisite Skills

Knowledge on MS Word, Libre Office, Adobe Acrobat Pro.

Guidance for Users

NIC, Industry Informatics Division, while formulating this document have used **MS Word 2010**, **Acrobat 9 Pro** and **Libre Office 6.4.5.2** to achieve accessibility of PDF documents/files. However, it is advisable to check for latest versions of MS Word and Acrobat Pro for more accessibility feature such as direct PDF/UA compliance and in such cases, the use of **PDF Accessibility Checker** (**PAC3**) can be ignored. Users of this document can take a view on this by referring the manual and features of the latest versions of software, if being used by them. Users are free to use any other Proprietary / Open Source Tool to achieve accessibility of PDF documents as per available standards.

References

- Users of this document are advised to visit hyperlinks to different websites for reference that have been added in this document for further details.
- Many Important Instructions, Results and Information have been marked in **BOLD** for the users of this document.

Executive Summary

This document describes techniques for creation of Accessible PDF documents to be published on Indian Government Websites which complies to PDF/UA and WCAG2.0 standards using Proprietary software (MS Word 2010 and Acrobat 9 Pro Extended), Open Source (Libre Office 6.4.5.2) and Shareware PDF Accessibility Checker (PAC3) . This document also highlight techniques for using Open Source Tools such as Imagemagick and Tesseract, to convert an image scanned PDF to an OCR PDF document.

This document is intended for content managers of Government websites having knowledge of MS Word, Libre Office and Acrobat Pro.

The process described in this document to create accessible PDF have been formulated based upon using the source document in word processors and scanned images of text in PDF documents that had been published on Government websites.

It is to inform the users of this document that foremost requirement of any PDF to be declared as accessible is by Meticulous Tagging its contents. A tagged PDF includes hidden accessibility mark-ups that, when properly applied, help to optimize the reading experience of those who use screen readers and other assistive technology (AT). A properly tagged PDF can also re-flow to adapt its presentation to different screen sizes, for example to provide a high-quality experience to users of smart mobile devices.

To start with, MS Word has an in-built checker, which could be used to first check the accessibility of MS Word file itself. This document lists recommended techniques, which should be implemented in a MS Word document before saving it as Tagged PDF. It is pertinent to mention that a Tagged PDF generated from MS Word may not be PDF/UA compliant, if the version of MS Word being used does not support it. In such cases, Acrobat Pro Full Check feature could be used to check the Accessibility of PDF document where by the Acrobat Pro lists errors and helpful hints for repairing it. Acrobat advanced editing tool such as Touch-up Object tool and Touch-up Reading Order Feature of accessibility could be used to fix most of the issues listed during checking of accessibility in Acrobat Pro.

Since different users of this document may have different version of Acrobat Pro and some of those version may not be checking the PDF/UA criteria, therefore to help such users, a shareware PDF Accessibility Checker (PAC3) could be used to audit PDF document for accessibility. If required, a few iteration could be made between Acrobat Pro and PAC3 to make PDF document PDF/UA compliant. While checking a PDF document using PAC3 for accessibility, errors and warning which are listed up could be fixed using Acrobat Pro. If the user is not very well conversant with Acrobat Pro, then use of online documentation available could be used to fix the error and warnings to make it PDF/UA compliant.

The same process used for making MS Word generated tagged PDF accessible could be used if the native source document has been authored using Open Source Libre Office.

The Image scanned PDF document available on Government websites could be made accessible by using OCR Text Recognition and Adding Tags to the document to make it Tagged PDF. A similar approach of few iteration between Acrobat Pro and PAC3 could be made to make PDF document PDF/UA compliant.

Based upon the observations of OTG, NIC the use of open source tools such as Tesseract, Imagemagick and FreeOCR have been defined in the document. Imagemagick could be used to convert an image scanned PDF to an image which further could be fed to Tesseract to generate an OCR based PDF. Generation of Tagged PDF is a limitation in Tesseract.

It is pertinent to mention that the Industry Informatics Division, NIC have used the common tools such as MS Word, Libre Office, and Acrobat Pro in this document. However, users of this document are free to use any other Proprietary / Open Source Tool to achieve accessibility of PDF documents subject to the conforming of standards prescribed in Government website guidelines.

The document shall help content managers of Government websites in complying with one of the objective of Accessible India Campaign launched by Government of India where Enhancing proportion of accessible and usable public documents and websites that meet internationally recognized accessibility standards are defined.

1. Introduction

1.1. Portable Document Format (PDF)

The Portable Document Format (PDF) is a file format developed by Adobe in the 1990s to present documents, including text formatting and images, in a manner independent of application software, hardware, and operating systems.

1.2. What is Accessible PDF

An accessible PDF is a PDF document that can be read and accessed by people with disabilities, primarily for the persons with impaired visions. They may use assistive technology to read the file through text-to-speech or a Braille printout of an accessible pdf document. A PDF document is considered to be accessible only if it meets a set of accessibility guidelines

1.3. Standards

- **Web Content Accessibility Guidelines (WCAG) 2.0**
- **PDF/UA (PDF/Universal Accessibility)**, formally ISO 14289, is an ISO standard for accessible PDF technology

1.4. What is Tagged PDF

A tagged PDF includes hidden accessibility mark-ups that, when properly applied, help to optimize the reading experience of those who use screen readers and other assistive technology (AT). Meticulous tagging is a crucial component of achieving a truly accessible PDF. A properly tagged PDF can also re-flow to adapt its presentation to different screen sizes, for example to provide a high-quality experience to users of smart mobile devices.

1.5. Background

Department of Empowerment of Persons with Disabilities (DEPwD) had Accessible India Campaign (Sugamya Bharat Abhiyan) as a nation-wide Campaign for achieving universal accessibility for Persons with Disabilities (PwDs). One of the objectives of Accessible India Campaign (Sugamya Bharat Abhiyan) is enhancing proportion of accessible and usable public documents and websites that meet internationally recognized accessibility standards. This target will ensure conversion of public documents published as of a specified year and all current websites meeting the relevant International Organization for Standardization (ISO) criteria that are found in ISO / IEC 40500: 2012, Information Technology – W3C Web Content Accessibility Guidelines (WCAG) 2.0. Public documents refer to all documents issued by the national government as well as all subnational documents. These include all publications such as laws, regulations, reports, forms and informational brochures. The target includes conducting accessibility audit of 50% of all government (both Central and State Governments) websites and converting them into fully accessible websites and ensuring that at least 50% of all public documents issued by the Central Government and the State Governments meet accessibility standards

MeitY OM No 18(3)/2018-E-Infra (Pt.) dated 10th December, 2019 (Annexure -'A') requesting Secretaries of all Central Ministries/Departments and Chief Secretaries of States/UTs to make the public documents accessible on Government websites. It has been suggested that the all the Government notifications/ orders uploaded on the website should be digitally signed and in ePub or OCR based PDF only along with a technical write-up regarding conversion. Their OM also mentions the procedure of making OCR based PDF files, W3C guidelines .

Subsequently, DEPWD in their OM dated 26th February, 2020 (Annexure -'B') has requested all Ministries / Department that document to be published on website should be in accessible format i.e. ePub or OCR based PDF formats.

1.6. Present Status

Most of the documents such as Acts, Orders, Notification, and Reports available on Government websites are in PDF files. They have been created mostly by using Microsoft Office Suite and saved as PDF without keeping accessibility in view. Moreover, large number of documents whose native source is not available has been imaged scanned and published on websites without performing Optical Character Recognition (OCR). Such documents pose inability to visually challenged persons to know their contents.

2. How to Make PDF Documents Accessible

Two main issues related to accessibility of PDF files on Government websites for Persons with Special Abilities (Divyangans) are as follows:-

i. **How to create accessible PDFs from source document like Word processors, Spreadsheet, Presentations etc.**
ii. **Which process is to be adopted to ensure accessibility of scanned images of text in PDF file that had been published on Government websites in the past?**

2.1. Software/Tools used

The following Propriety / Open Source / Shareware were used while drafting the procedures and observations in this document.

S.No	Name & URL	Version
1.	Microsoft Word (available with NIC-DPIIT)	Version 2010
2.	(*) **PDF Accessibility Checker 3.** **Download PAC3** **Please read License**	PAC 3
3.	(*) **Libre Office** **Download Libre Office** **License : MPLv2.0 (secondary license GPL, LGPLv3+ or Apache License 2.0)**	6.4.5.2

S.No	Name & URL	Version
4.	Acrobat 9 Pro Extended (available with NIC-DPIIT)	9
5.	(*) OCR Engine libtesseract and a command line - Tesseract Download Tesseract	Apache License 2.0 Tesseract open source OCR Engine v5.0.0-alpha.20200328
6.	(*) Free OCR Download FreeOCR	GNU AGPL V3 (a9t9) Free OCR for Windows Desktop V1.08
7.	Imagicmagick Download Imagemagick	Version Imagemagick 7.0.10-26 Q16 x64 IMDisplay Version 1.0
8.	Nonvisual Desktop Access(NVDA) Screen Reader Download NVDA	2020.2

2.2. Create accessible PDF documents from source documents

Overview of Accessible Documents

Few basic steps to assure that document are readable by individuals with disabilities.

- ✓ **Use Headings**

Headings and subheadings should to be identified as such using the built-in heading features of the authoring tool. Headings should form an outline of the page content (Heading 1 for the main heading, Heading 2 for the first level of sub-headings, Heading 3 for the next level of sub-headings, etc.). This enables screen reader users to understand how the page is organized, and to quickly navigate to content of interest. Most screen readers have features that enable users to jump quickly between headings with a single key-stroke. Virtually every document authoring format includes support for headings and subheadings.

- ✓ **Use Lists**

Any content that is organized as a list should be created using the list controls that are provided in document authoring software. Most authoring tools provide one or more controls for adding unordered lists (with bullets) and ordered lists (with numbers). When lists are explicitly created as lists, this helps screen readers to understand how the content is organized. When screen reader users enter a list, their screen reader informs them that they're on a list and may also inform them of how many items are in the list, which can be very helpful information when deciding whether to continue reading.

✓ **Use Meaningful Hyperlinks**

Links presented in an electronic document should convey clear and accurate information about the destination. Most authoring tools allow the creator to assign a hyperlink to text. For documents that will be circulated as print material, use a URL shortening service to create a customized and meaningful link name.

✓ **Add Alternate Text for Images**

Users who are unable to see images depend on content authors to supplement their images with alternate text, which is often abbreviated "alt text". The purpose of alt text is to communicate the content of an image to people who can't see it. The alt text should be succinct, just enough text to communicate the idea without burdening the user with unnecessary detail. When screen readers encounter an image with alt text, they typically announce the image then read the alt text. Most authoring tools provide a means of adding alternate text to images, usually in a dialog that appears when an image is added, or later within an image properties dialog. If images are purely decorative and contain no informative content, they do not require a description. However, they may still require specific mark-up so screen readers know to skip them. The methods for hiding decorative images from screen reader users is described in more detail in the format-specific pages within this section of the website. Also, images that require a more lengthy description, such as charts and graphs, may require additional steps beyond adding alt text.

✓ **Identify Document Language**

Leading screen reader software is multilingual, and can read content in English, Spanish, French, and a wide variety of other languages. In order to ensure that screen readers will read a document using the appropriate language profile, the language of the document must be identified. Identification of the language of any content written in a language other than the document's default language should be mentioned. With this information, supporting screen readers will switch between language profiles as needed on the fly. Most document authoring tools provide a means of identifying the document language as well the language of specific parts.

✓ **Use Tables Wisely**

Tables in documents are useful for communicating relationships between data, especially when those relationships can be best expressed in a matrix of rows and columns. Tables should not be used to control layout. Authoring tools have other means of doing this, including organizing content into columns. If the data is best presented in a table, try to keep the table simple. If the table is complex, consideration should be done to divide it into multiple smaller tables with a heading above each. A key to making data tables accessible to screen reader users is to clearly identify column and row headers. Also, if there are nested columns or rows with multiple headers for each cell, screen readers need to be explicitly informed as to which headers relate to which cells.

- ✓ **When Exporting to PDF, Understand How to Preserve Accessibility**

In order for an Adobe PDF document to be accessible, it must be a "tagged" PDF, with an underlying tagged structure that includes all of the features already described on this page. There are right ways and wrong ways to export documents to PDF. Some authoring tools don't support tagged PDF at all while others provide multiple ways of exporting to PDF. Some produce tagged PDF but some do not.

2.2.1. Using Proprietary Software Microsoft Word 2010

A sample MS Word file comprising of Text, Images, Bullets, Number List and Table have been used here. Following should be applied to make MS Word document accessible.

- ✓ Use MS Word to correct the Errors and Warning mentioned in the Accessibility Checker Pane before saving as PDF File
- ✓ Whenever possible, please return to the source document file and add accessibility features in the authoring application such as MS Word
- ✓ Best Practices for making Word document accessible (Refer Microsoft Support)
- ✓ Preserve Fidelity while sharing this document by embedding the Fonts (File>>Options>>Save>>Embed Fonts in the File)
- ✓ List formatting

  Check to ensure that bulleted, numbered, outline and multi-level lists are formatted properly. Improper formatting makes it difficult for non-sighted users to find a list, navigate through a list, identify the list type, and identify when there are multiple levels within a list.

- ✓ **Language settings**

  Check to ensure the language setting is defined properly for passages of text. Improper language settings result is mispronounced words and impaired comprehension by non-sighted users.

- ✓ **Document properties**

  Check to ensure that the document title, author, subject, and keywords are provided under document properties. Missing information will make it difficult for non-sighted users to discern this important information about the document.

- ✓ **Colour and contrast**

  Check to ensure that all text is readable and distinguishable from background colours, watermarks, and background images, and that all text is readable in High Contrast mode. This will help user with partial visual impairments read the document more easily.

- ✓ **Complex table**

  Check to determine if the document contains complex tables. If it does, move on to complete Step 2, and then convert the document to an accessible PDF document.

- ✓ **Unclear hyperlink text**

  Hyperlink text, which is not meaningful, descriptive, and unique, needs to be appropriately labelled. For example, a link titled click here does not provide enough information to a non-sighted user to understand the link's destination or purpose.

- ✓ **Unstructured document**

  Documents, which are not, formatted using styles and heading levels may not contain enough structure to enable a non-sighted user to navigate through a document as quickly as a sighted user.

- ✓ **Skipped heading level**

  Skipped heading levels exist when heading levels are defined in the document but in an inconsistent logical reading order (for example, a heading formatted as level 3 follows a heading formatted as level 1). Skipped heading levels make it difficult for non-sighted users to navigate a document.

- ✓ **Repeated blank characters**

  Blank spaces used for formatting purposes (for example, multiple carriage returns, and the use of tabs and spaces to align text) create reading issues for non-sighted users.

- ✓ **Object not inline**

  Objects, which are not 'inline' with text (also called floating objects), cannot be found by a non-sighted user and should not be used.

- ✓ **No header row specified**

  When heading rows are not defined, non-sighted users may have difficulty identifying the meaning of data cells and how they relate to other data in the table.

- ✓ **Blank table rows or columns**

  When tables contain blank rows or columns, it is difficult for non-sighted users to understand and navigate through the table.

- ✓ **Missing alt text (table)**

  Titles or summaries should be added to tables so non-sighted users can comprehend the purpose and design of the table without going through the entire table.

✓ **Missing alt text (picture, text box, other elements)**

Picture, text boxes, and other non-decorative images require text descriptions (also called alternative text or "Alt Text"), to convey information to non-sighted users.

✓ **Heading is too long**

This issue can be ignored. Please avoid long headings, but this is not a requirement of Section 508, and often unavoidable with government documents. Use plain and concise language for headers and otherwise ignore this test result.

✓ **Infrequent headings**

This issue can be ignored. It is safe to ignore this test; it is a redundant test already covered by the 'Unstructured Document' test.

✓ **Merged or split cells**

This issue can be ignored. It is safe to ignore this test; it is a redundant test already sufficiently covered by the 'Unstructured Document' test.

✓ **Use image watermark**

This issue can be ignored. See the manual check titled "Colour and Contrast" for more relevant guidance for SSA.

✓ **Check reading order**

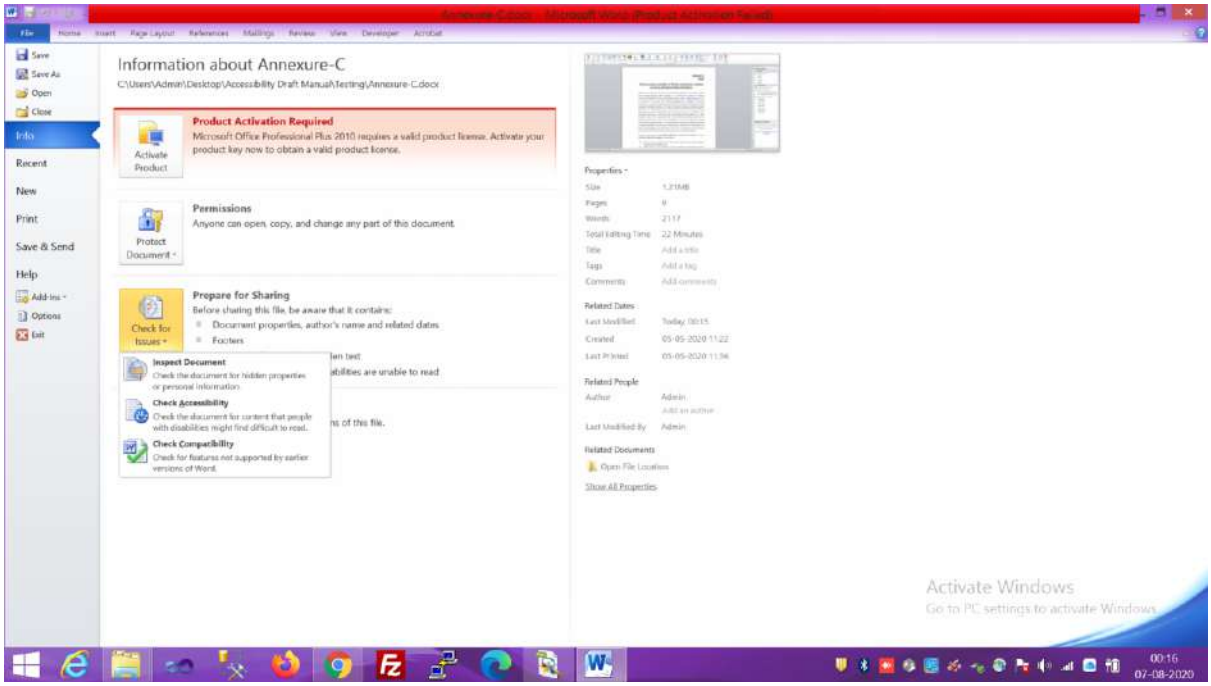
This issue can be ignored. Microsoft's automated checker suggests using tables to create a logical reading order structure within a document. However, SSA discourages the use of tables for page formatting because layout tables create many accessibility challenges. If using tables for formatting/layout complete all the manual and automated tests and convert the document to an accessible PDF.

2.2.1.1. Verify Accessibility

2.2.1.1.1. Using MS Word 2010

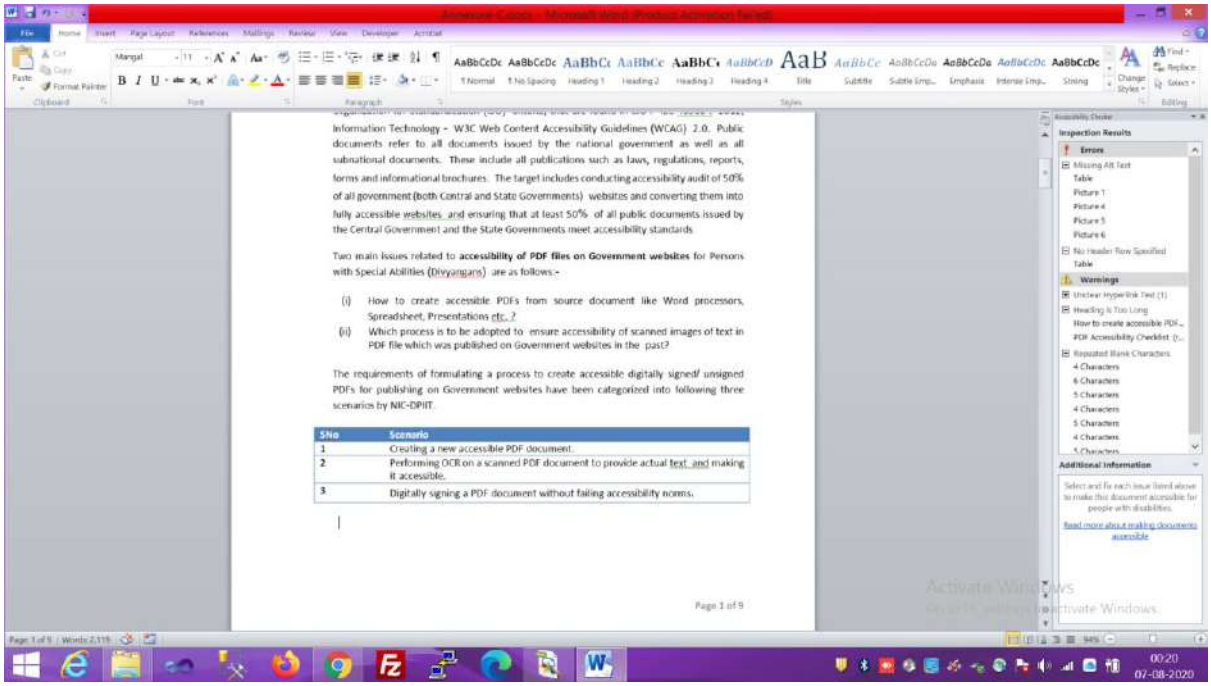
Before saving as PDF, the accessibility of the word document itself can be checked as follows:-

- **Go To File >>Info>>Check for Issues>>Check Accessibility** (refer Figure-1)



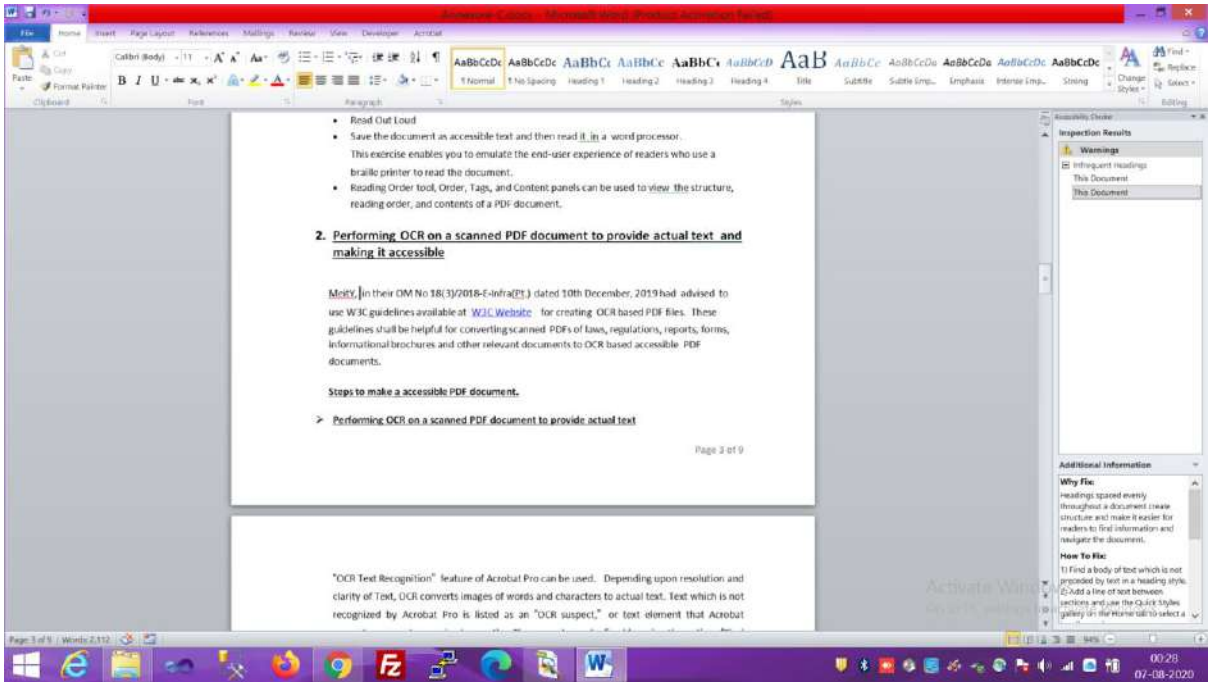
(Figure-1)

- Accessibility Checker Pane on the right list inspection Results which comprises of Error and Warnings (Refer Figure-2)



(Figure-2)

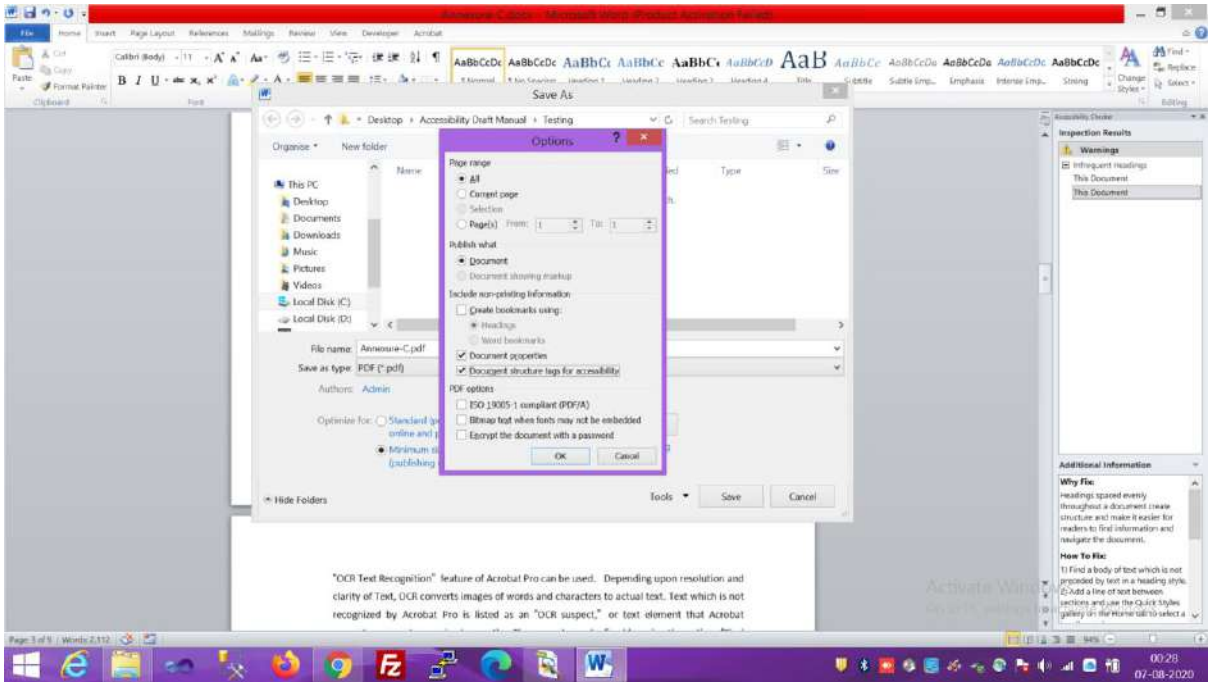
- Refer (Figure-3) – Most of the error and warning have been fixed using MS Word Accessibility Checker pane.



(Figure-3)

- Save as PDF Document

  File >>Save As >>Select PDF Type>>Option>>Document Structure Tags for Accessibility (Refer Figure-4)



(Figure-4)

- **Please Note: - If the PDF/A compliance is selected then for editing it in Acrobat Pro, PDF/A compliance has to be disabled.**

2.2.1.1.2. Using Acrobat 9 Extended

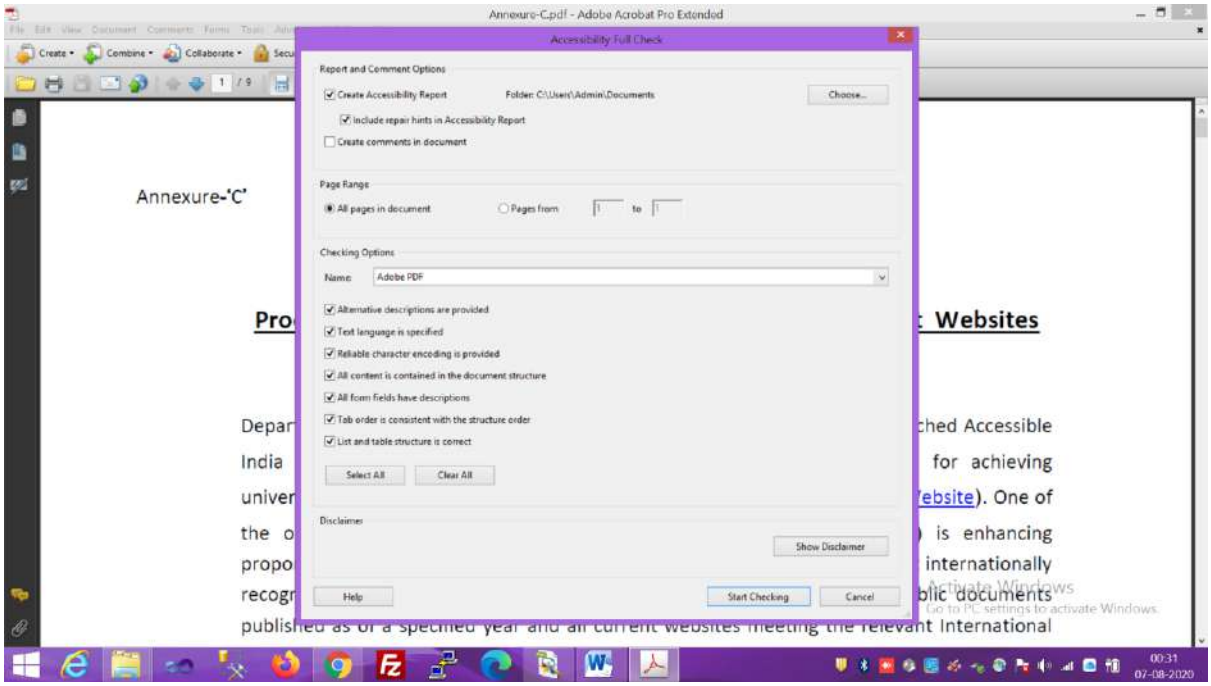
Sometimes even if MS Word Accessible converted, PDF is opened in Acrobat Pro, it highlights some error and those can be corrected using Acrobat Pro. In order to check the accessibility of PDF files, **Full Check** Feature of Acrobat Pro under Accessibility can be used .The results are displayed in the Accessibility Checker panel on the left, which also has helpful links and hints for repairing issues such as Adding Tags, Character Encodings, Alternate text, Language Attributes etc.

Please Note -:

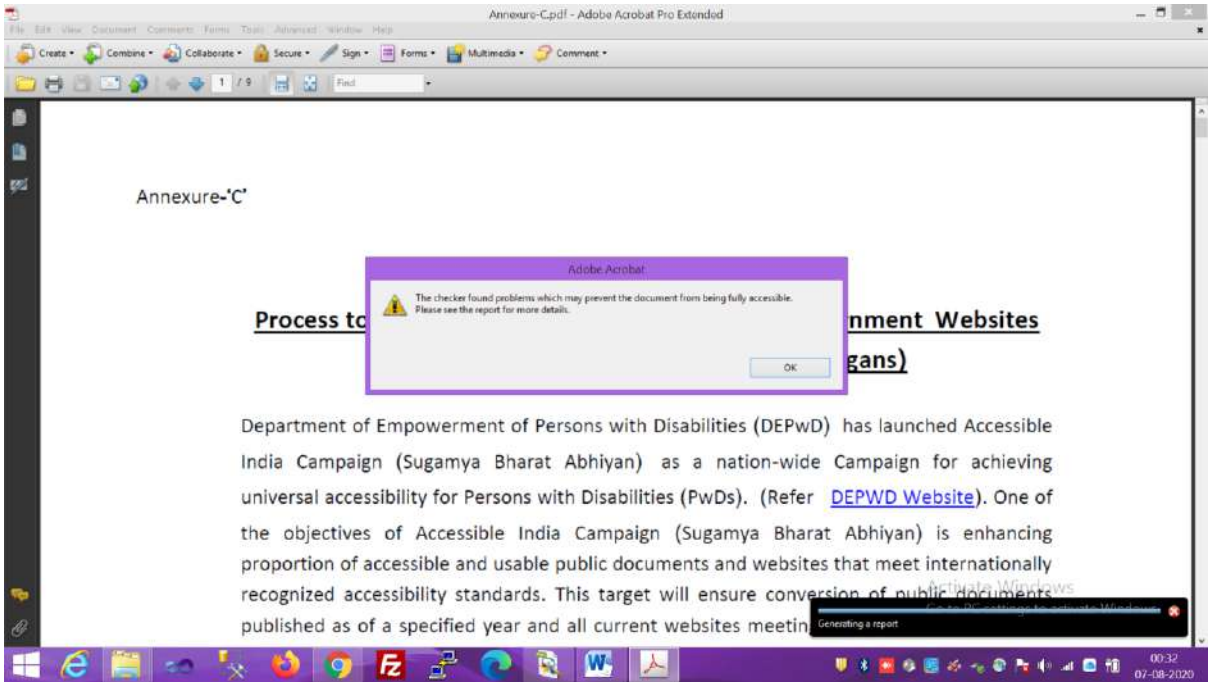
Other methods in Acrobat can be used to check PDF accessibility:

- Reflow view to check the reading order.
- Read Out Loud
- Save the document as accessible text and then read, it in a word processor.This exercise enables to emulate the end-user experience of readers who use a braille printer to read the document.
- Reading order Tool, Order, Tags, and Content panels can be used to view the structure, reading order, and contents of a PDF document.

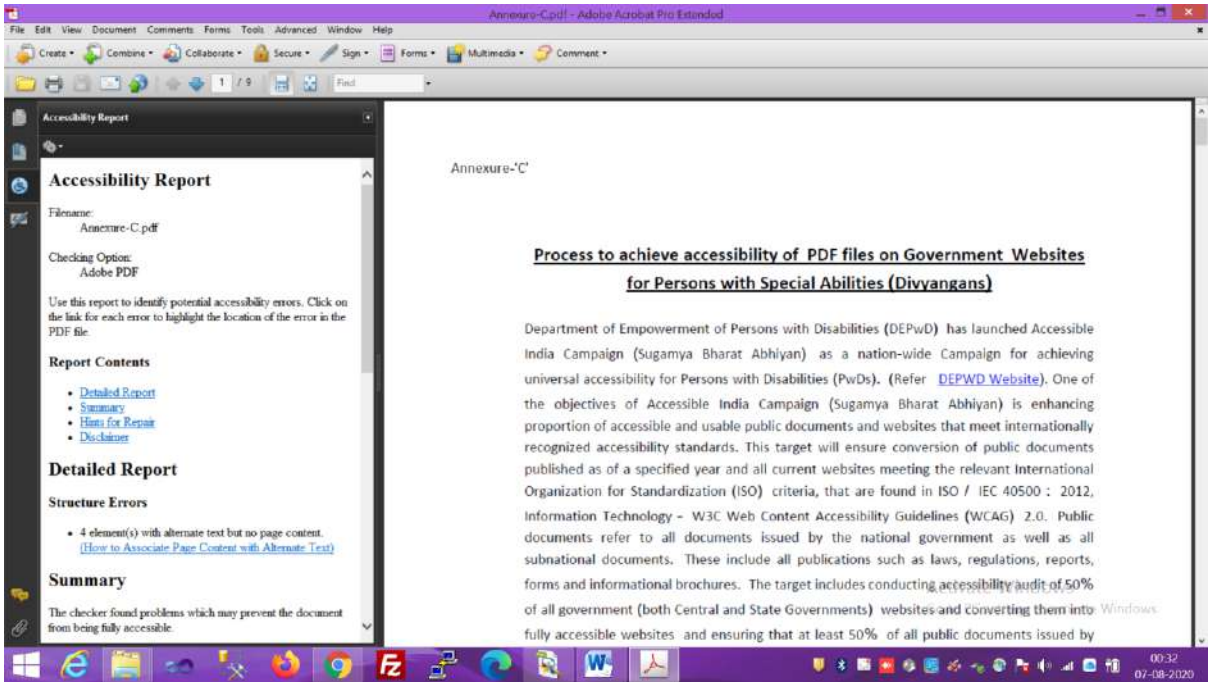
The PDF document created using MS Word in section 2.2.1.1.1 is checked for accessibility as illustrated above and it fails accessibility criteria (Refer Figure-5, Figure-6 and Figure-7).



(Figure-5)



(Figure-6)

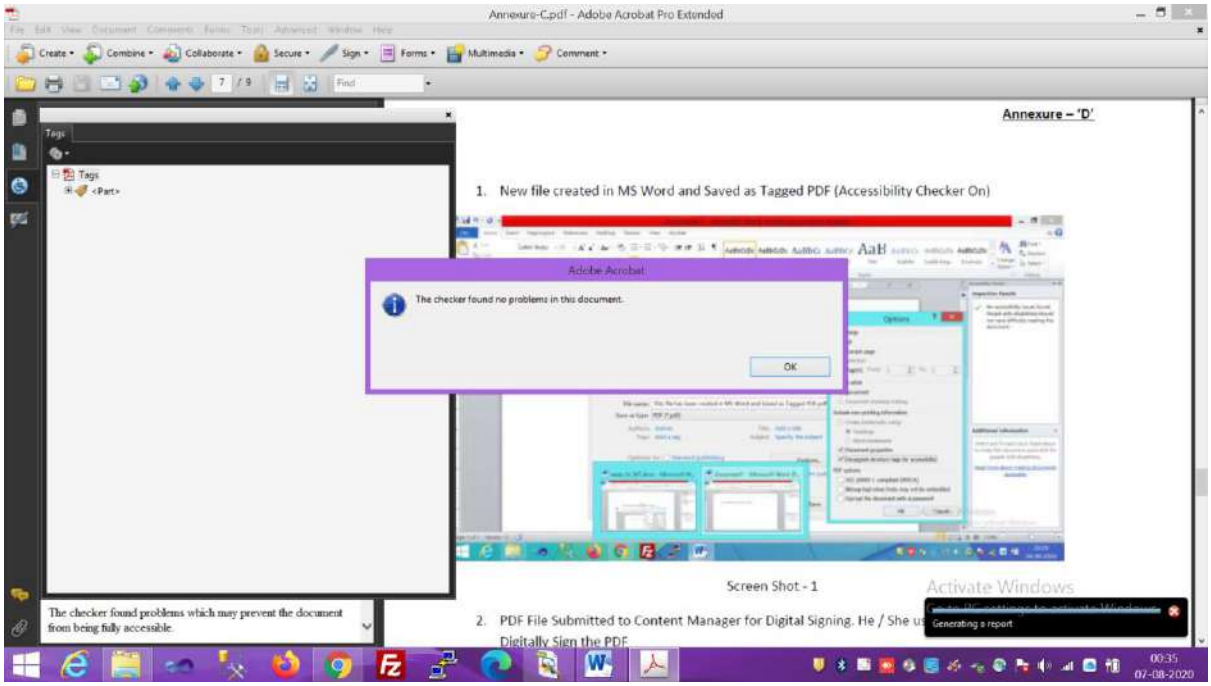


(Figure-7)

The error listed in the Accessibility Report on the left pane of Acrobat Pro can be resolved with the help of Hints for Repair mentioned there and the document can be made accessible (Refer Figure-8).

Advanced Editing tools such as "Touch-up Object Tool" are used for some type of listed repairs. The "Touch-up Reading Order Tool" provides the easiest and quickest method to fix reading order and tagging issues. The Reading Order tool is intended for repairing PDFs that were tagged using Acrobat, not for repairing PDFs that were tagged during conversion from an authoring application. Whenever possible, return to the source file and add accessibility features in the authoring application. Repairing the original file ensures that repeatedly touch up future iterations of the PDF in Acrobat will not be required (Refer here).

- ✓ **Sometimes it is quite possible that native source document of PDF in MS Word etc. is not available and in such case editing of PDF can be done using Acrobat Pro and accessibility can also be achieved by using the Full Check Feature of Acrobat Pro.**

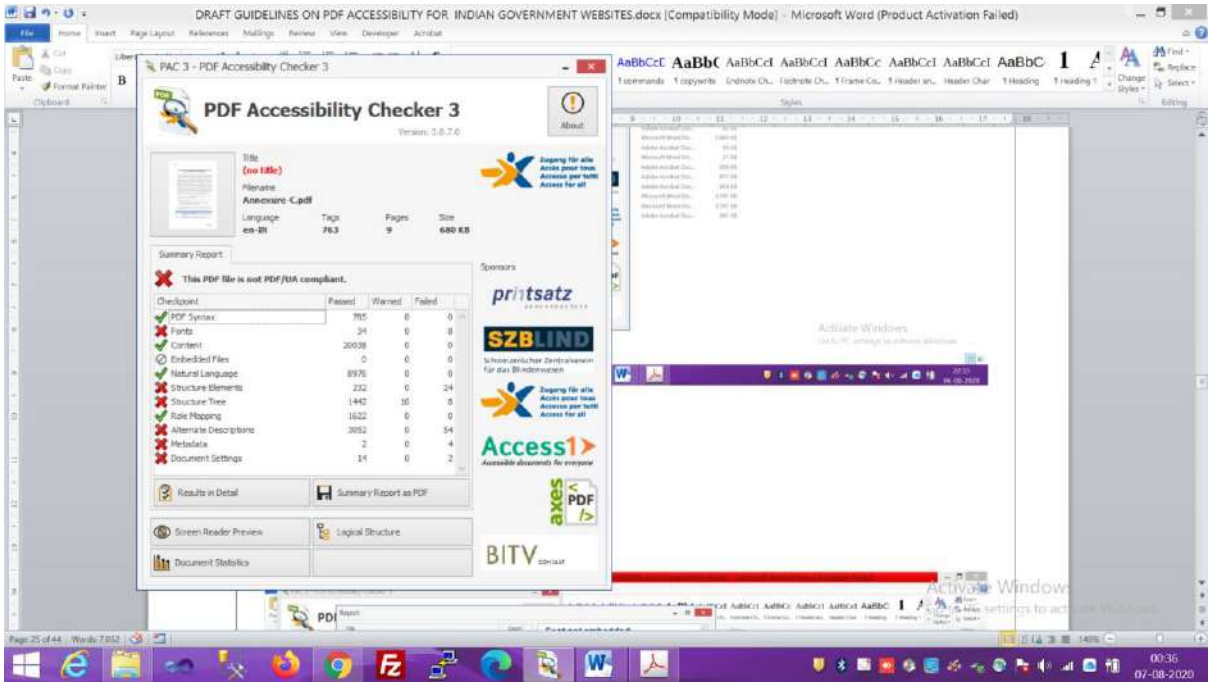


(Figure-8)

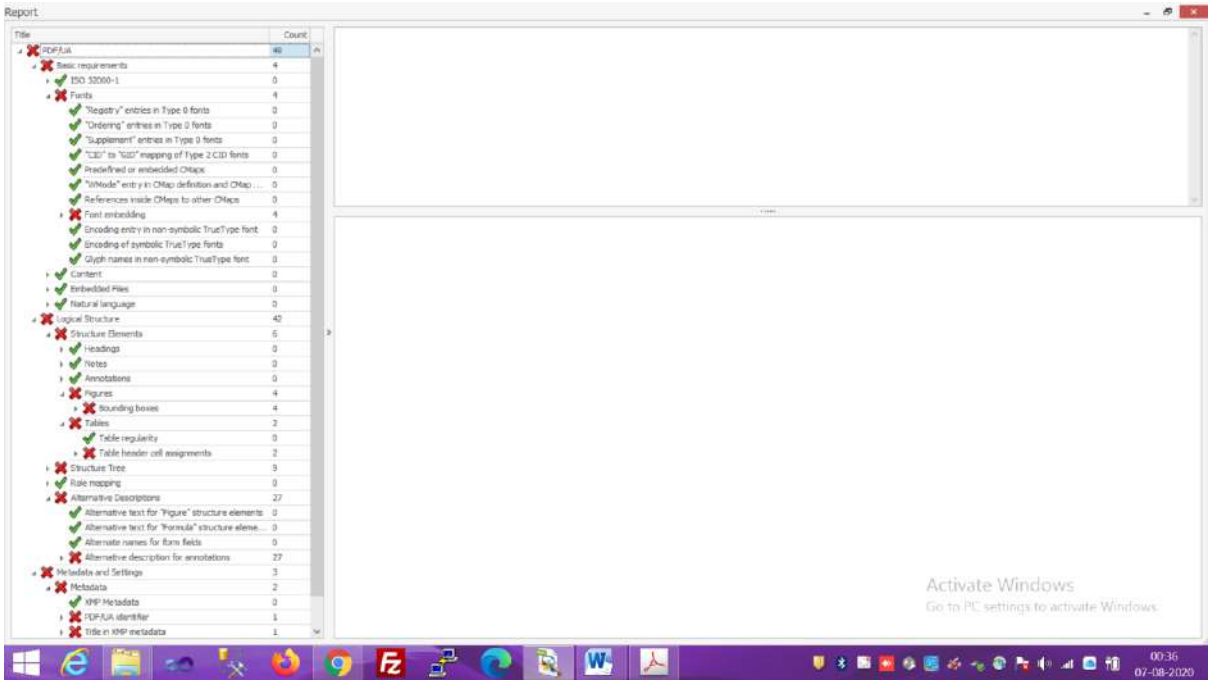
2.2.1.1.3. Using PDF Accessibility Checker (PAC3)

A shareware PDF Accessibility Checker (PAC3) can be downloaded subject to their Licence Agreement to check the accessibility of PDF documents. This checker lists the errors but it does not display hints to resolve the issue. This checker, which confirms to standards of PDF/UA (also mentioned by OTG, NIC in their observation) can be used to identify the accessibility issues.

In the following Figure-9 and Figure-10, the document, which was declared accessible by Acrobat Pro, is rechecked using PAC 3 to find out whether the document confirms to PDF/UA standards.



(Figure-9)



(Figure-10)

It can be seen in Figure-9 & Figure-10 that the document which was declared accessible by Acrobat 9 Pro is not PDF/UA complaint as per PDF Accessibility Checker (PAC3).

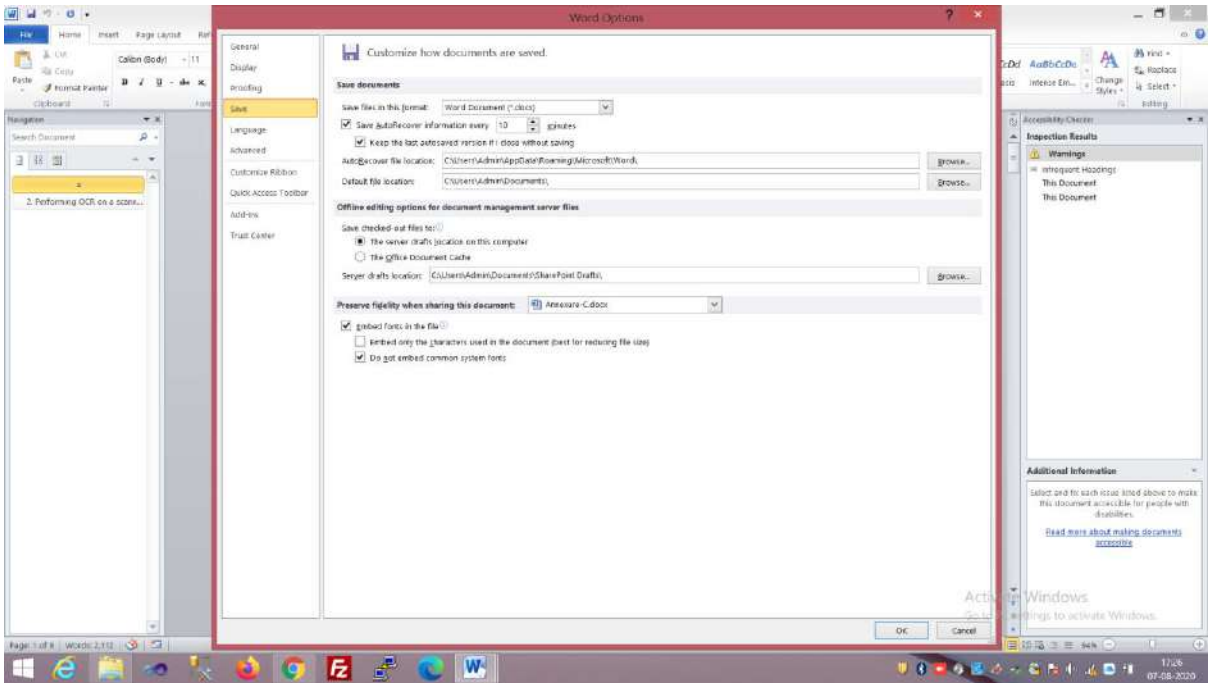
- ✓ **In the test case mentioned above, we used Microsoft 2010 and Acrobat 9 Pro. The latest version of MS Word or Acrobat Pro DC may exhibit PDF/UA compliance, however, we have not**

explored these latest versions for the conformance of PDF/UA compliance using PAC3

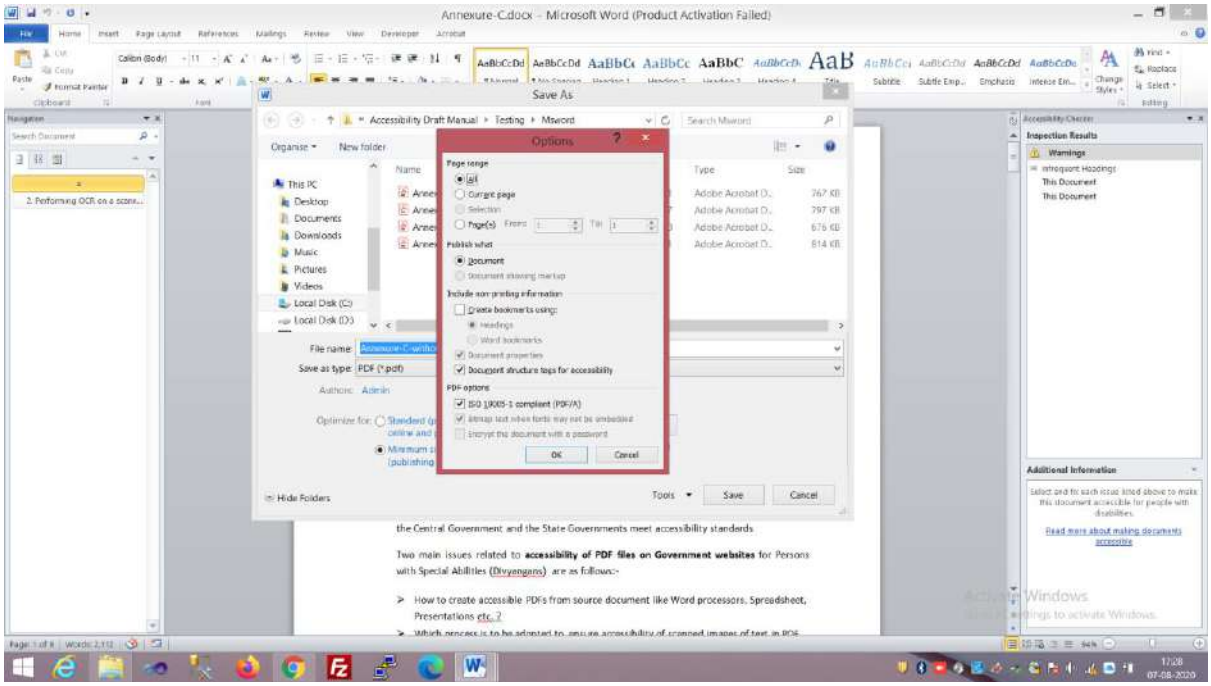
2.2.1.2. Repairing to make it accessible per PDF/UA Compliant

2.2.1.2.1. Using MS Word

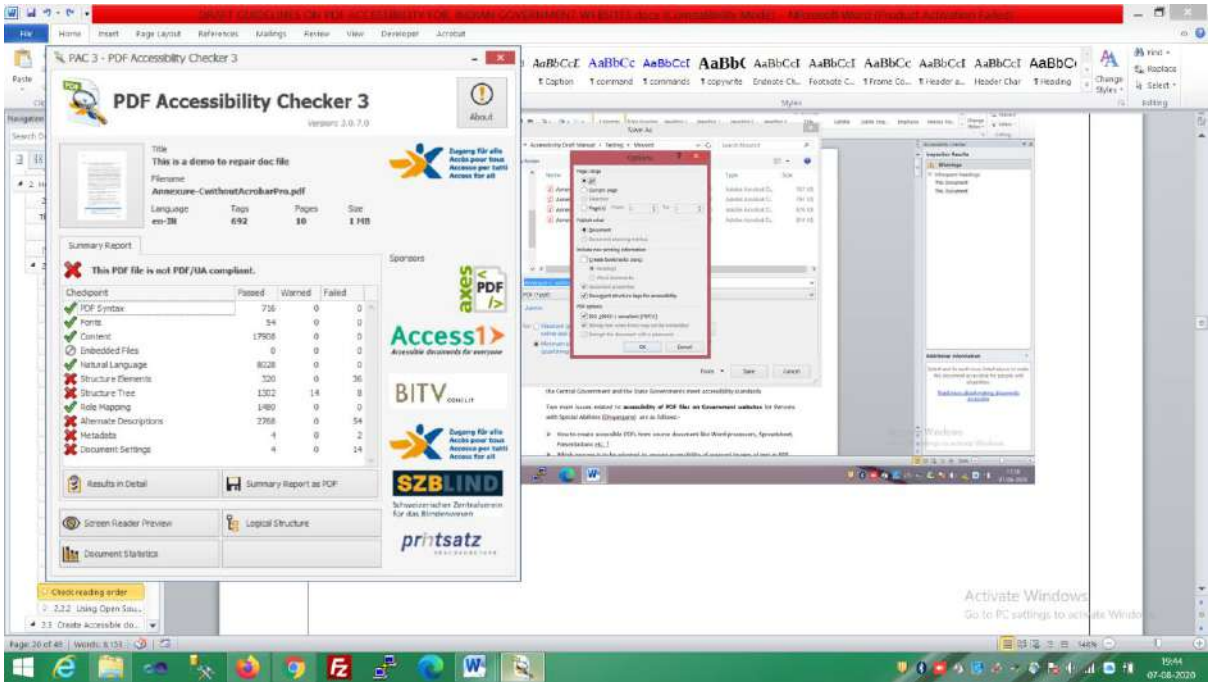
In this section, an attempt has been made to repair the native MS Word Files iteratively and saving it as tagged pdf document by checking its PDF/UA compliance using PDF Accessibility Checker PAC3. **Acrobat Pro has not been used in this process**.



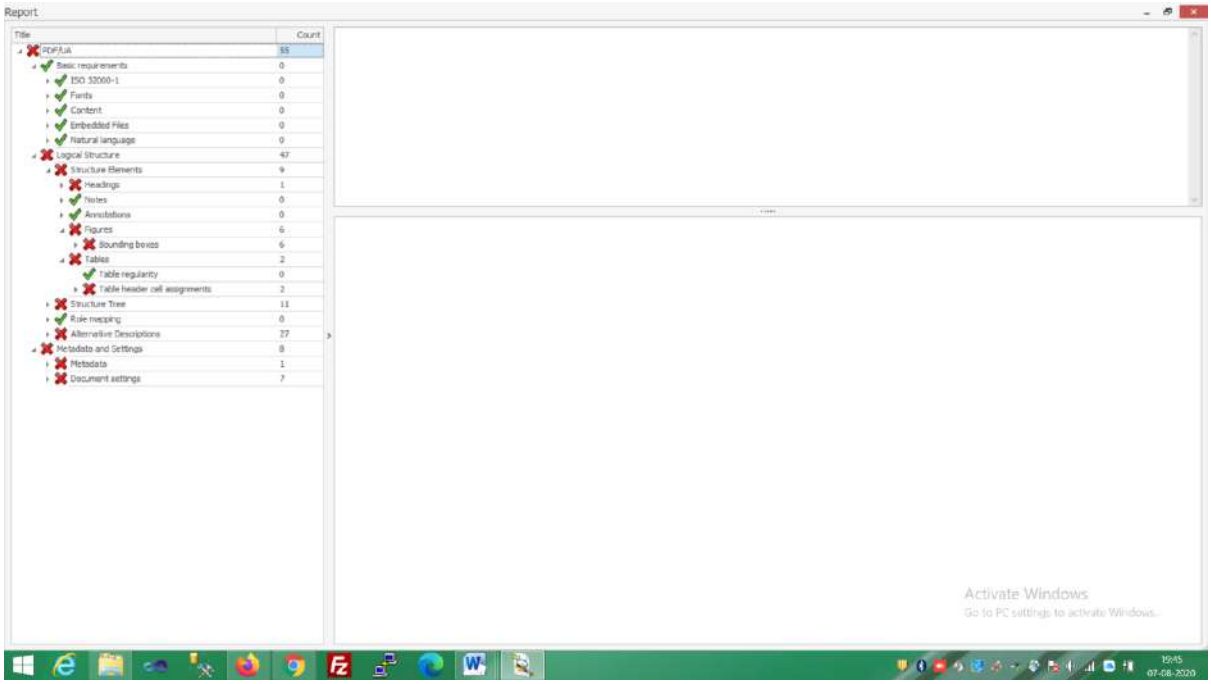
(Figure-11)



(Figure-12)



(Figure-13)



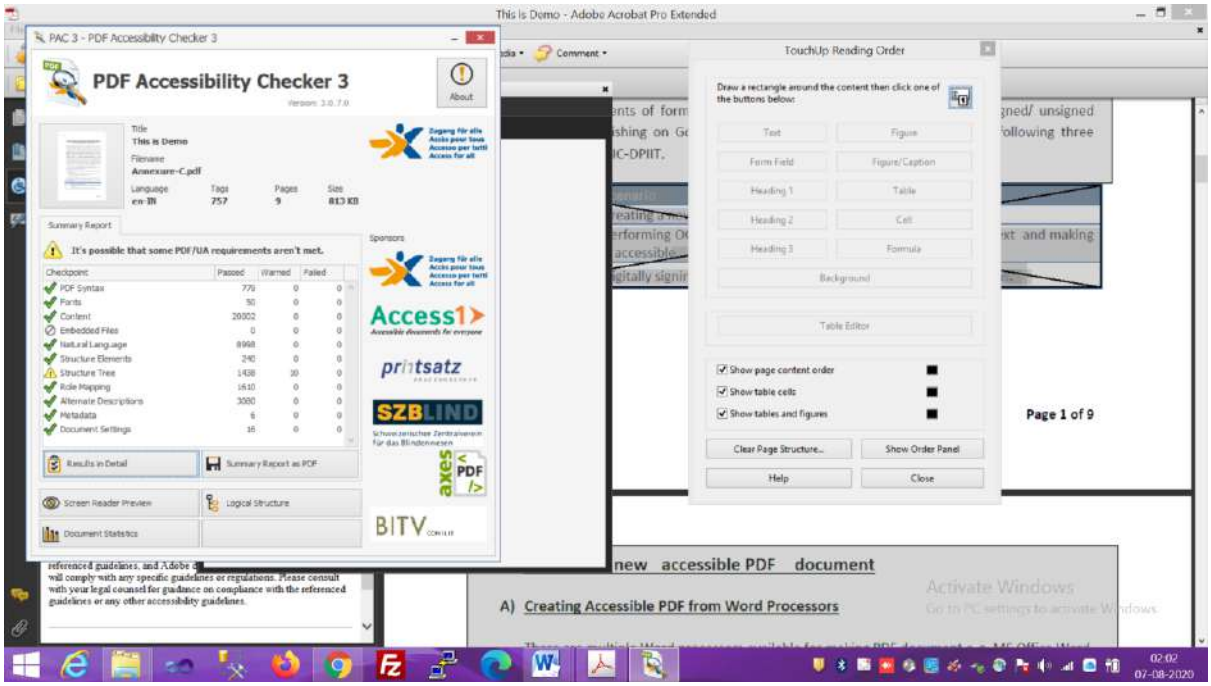
(Figure-14)

Now it was attempted to resolve the errors with the help of MS Word. The MS Word document confirms to most of the accessibility criteria such as Document Title, Alt Text, Table Header, Hyperlink, Creating list etc. as mentioned in Section 2.2.1.1.1 but PAC3 report has listed accessibility issues (Refer Figure-11, Figure-12, and Figure-13 & Figure-14). Hence, a good knowledge of MS Word may help to achieve PDF/UA compliance although we were not able to achieve this compliance using MS Word 2010 alone.

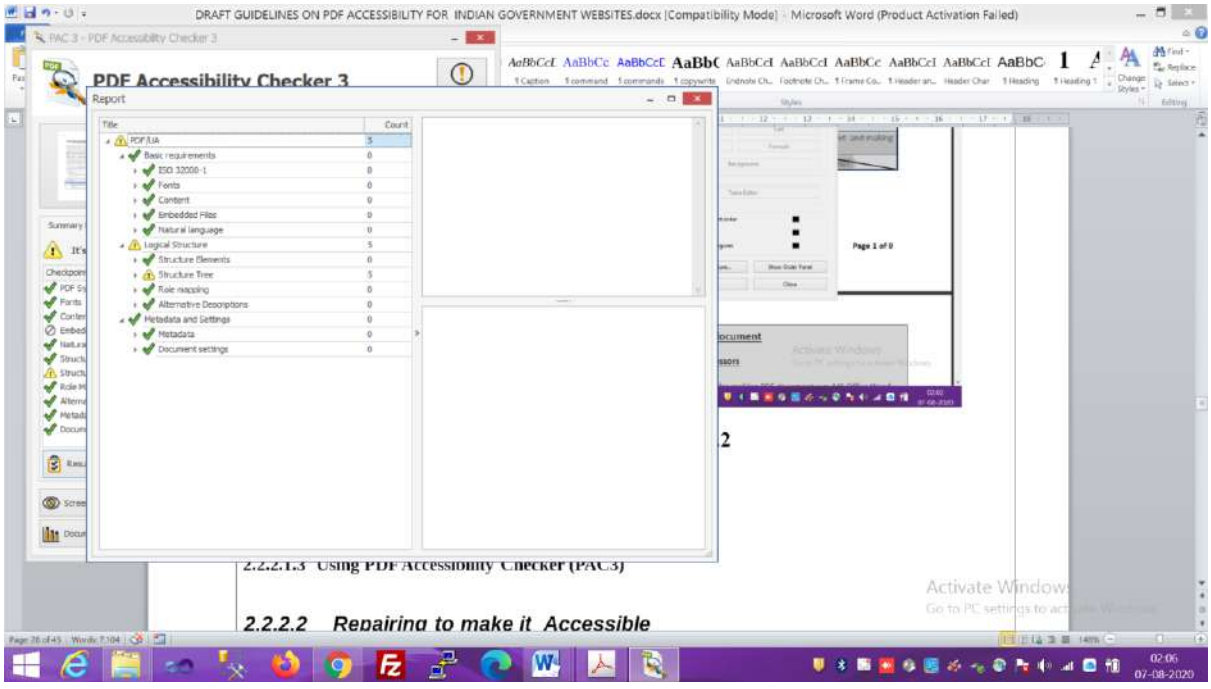
2.2.1.2.2. Using Acrobat 9 Pro

The Error listed in PAC3 were resolved using Acrobat Pro and the PAC3 results are shown in Figure-15 & Figure-16

- ✓ **A Good Knowledge of Acrobat Pro is essential to achieve PDF/UA compliance. The user can also seek the help from Internet in finding the solution and fixing the issued pointed by PAC3 but this process could be time consuming.**
- ✓ **We were able to achieve PDF/UA compliance using the old Acrobat 9 Pro Extended (only warnings are left which can also be removed).**



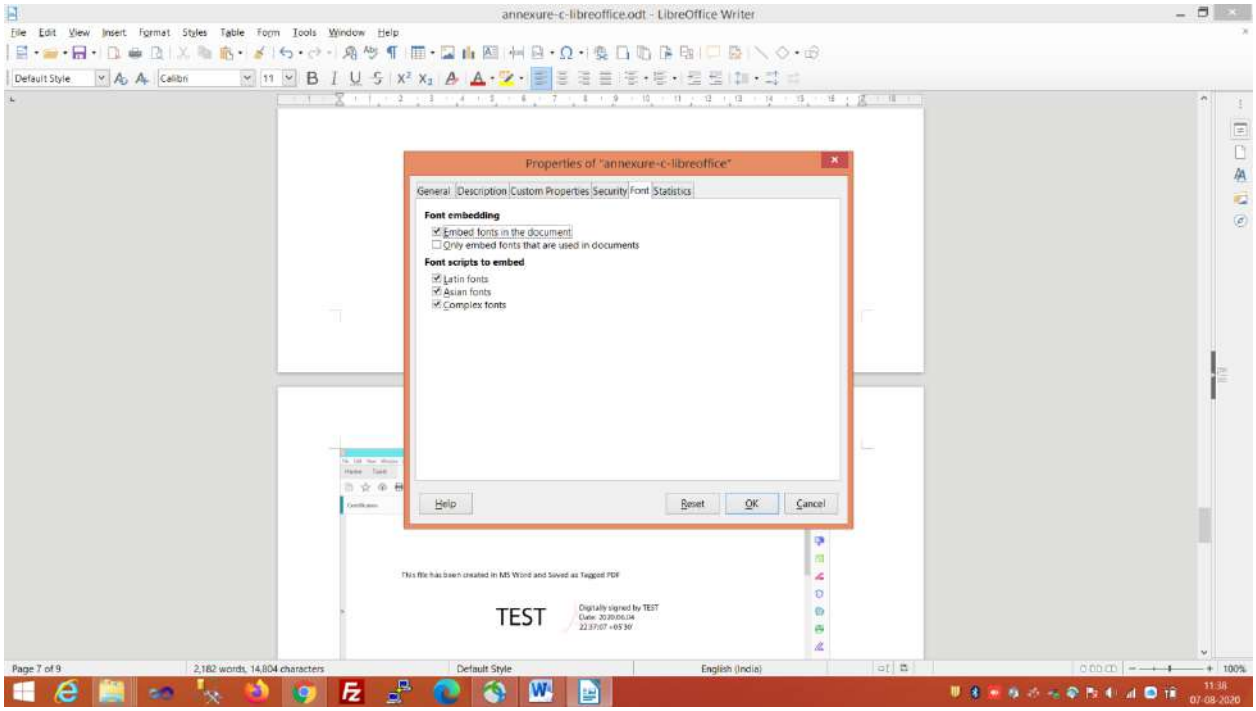
(Figure-15)



(Figure-16)

2.2.2. Using Open Source Libre Office Write 64.5.2

Same contents, which were earlier edited in MS Word, are entered in Libre Office Write, which comprises of Text, Images, Bullets, Number List and Table etc. has been used here. Fonts that were used in this document were embedded (Refer Figure-17).



(Figure-17)

Following should be taken into consideration to make Libre Office Write document accessible.

- ✓ Use Accessible Templates
- ✓ Specify Document Language
- ✓ Provide Text Alternatives for Images and Graphical Objects
- ✓ Avoid "Floating" Elements
- ✓ Use Headings
- ✓ Use Named Styles
- ✓ Use Built-In Document Structuring Features
- ✓ Create Accessible Charts
- ✓ Make Content Easier to See
- ✓ Make Content Easier to Understand
- ✓ Check Accessibility
- ✓ Use Accessibility Features when Saving/Exporting to Other Formats
- ✓ Consider Using Accessibility Support Applications/Plugins

  Some of the best practices to make a Libre Office Document accessible can be found at (LibreOffice documentation-1 and LibreOffice documentation-2).

2.2.2.1. Verify Accessibility

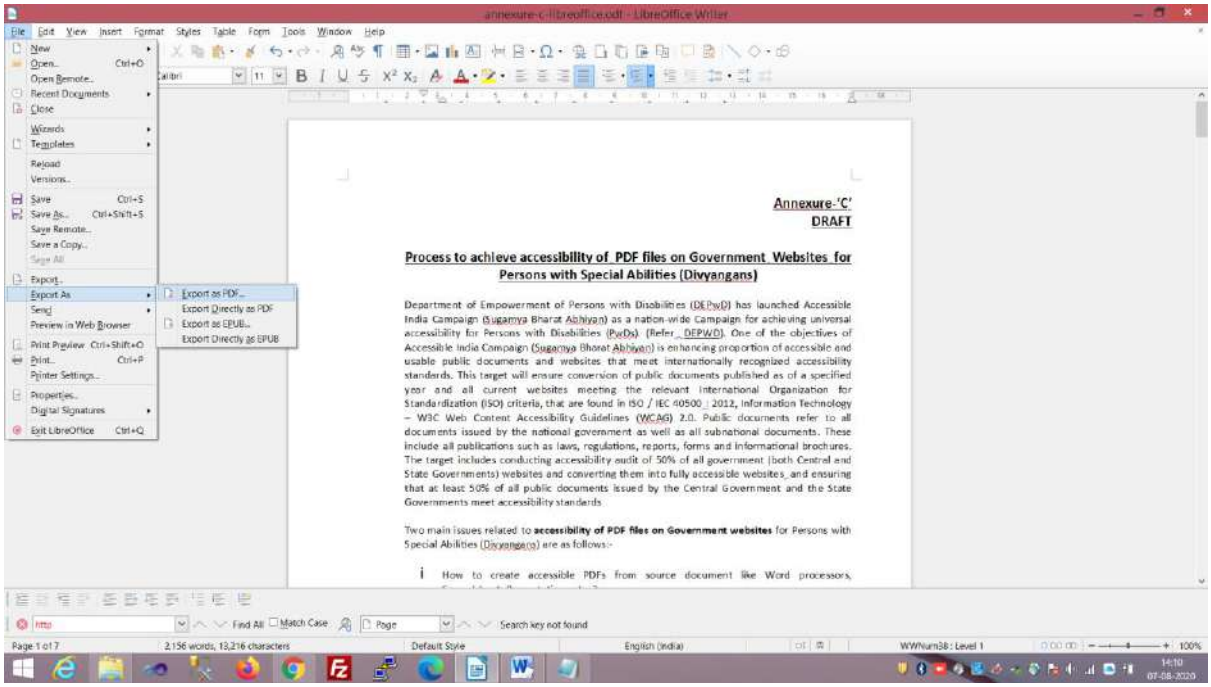
2.2.2.1.1. Use Libre Office

- AccessODF extension (.odt) can be downloaded and In LibreOffice, go to Tools > Extensions, and browse to the OXT file to add the extension.
- After restarting LibreOffice, please locate a new "Accessibility evaluation" item in the Tools menu.
- Using AccessODF

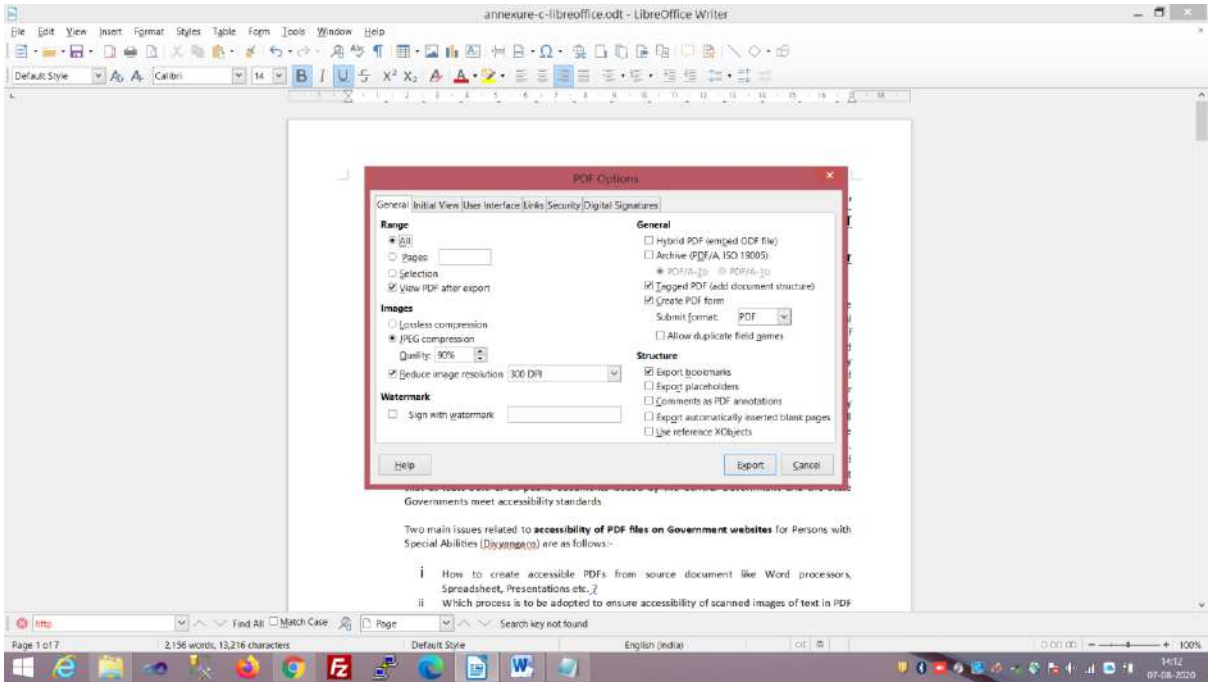
Go to "Accessibility evaluation" in the Tools menu. This opens a task panel to the right of the editing area. Press the "Check/Recheck" button at the bottom of the task panel and wait for the evaluation to complete. AccessODF will create a tree-like list of errors and warnings. Review each error and warning. For each issue, AccessODF displays its name, a description (what the problem is and why) and repair suggestions. For some issues, the Repair button will become active; pressing this button will either repair the issue automatically or open the dialog where the issues can be fixed. For other issues, it is needed to follow the instructions in the repair suggestions. If AccessODF erroneously flagged something as an issue, press the Ignore button. When ready, press the Check/Recheck button again for a new evaluation. If all issues have been solved, AccessODF will display a success dialog.

- ✓ **Please Note that AccessODF 0.1.0 is not compatible with the sidebar in LibreOffice 4.0. The sidebar panel where the AccessODF user interface should appear remains empty. This will be fixed in a later release of AccessODF (refer LibreOffice Extension).**
- ✓ Export LibreOffice File to tagged PDF as depicted in Figure-18 & Figure-19.
- ✓ Save as PDF Document

File >>Export As >>Export As PDF >>Select Tagged PDF Type



(Figure-18)



(Figure-19)

2.2.2.1.2. Using Acrobat Pro

In order to check the accessibility of PDF files exported from Libre Officer Write, **Full Check** Feature of Acrobat Pro under Accessibility can be use. The results are displayed in the Accessibility Checker

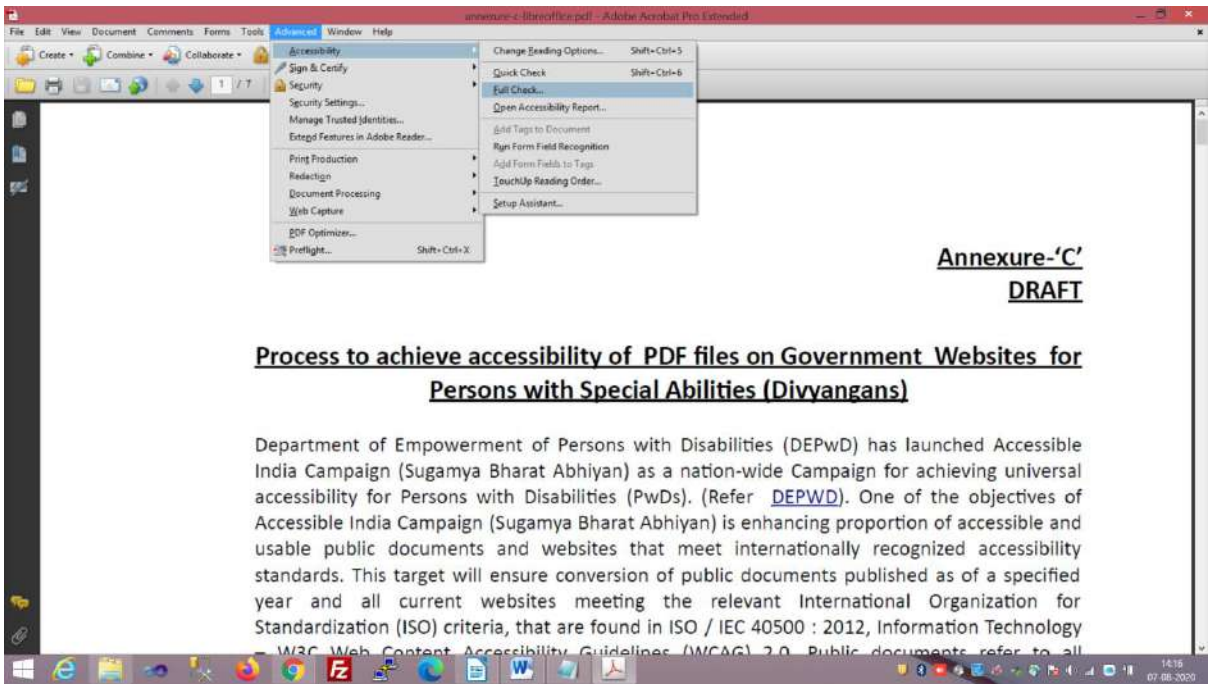
panel on the left, which also has helpful links and hints for repairing issues such as Adding Tags, Character Encodings, Alternate text, Language Attributes etc.

Please Note -:

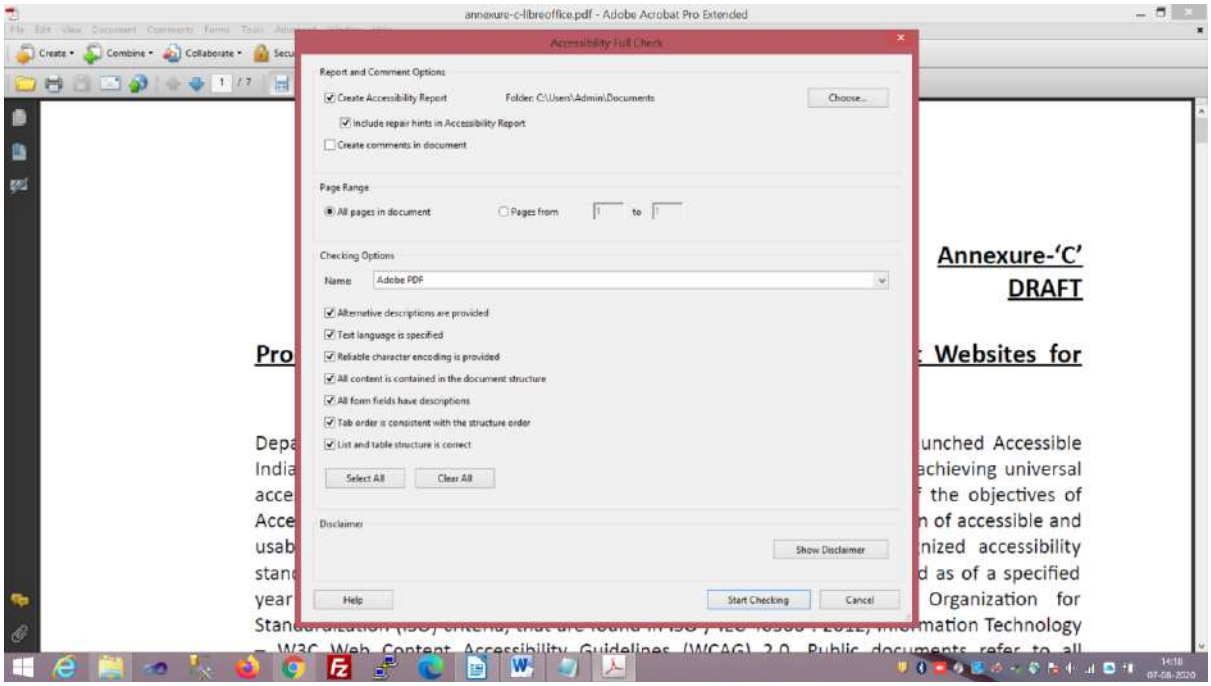
Other methods in Acrobat can be used to check PDF accessibility:

- Reflow View to check the reading order.
- Read Out Loud
- Save the document as accessible text and then read, it in a word processor.This exercise enables to emulate the end-user experience of readers who use a braille printer to read the document.
- Reading Order Tool, Order, Tags, and Content panels can be used to view the structure, reading order, and contents of a PDF document.

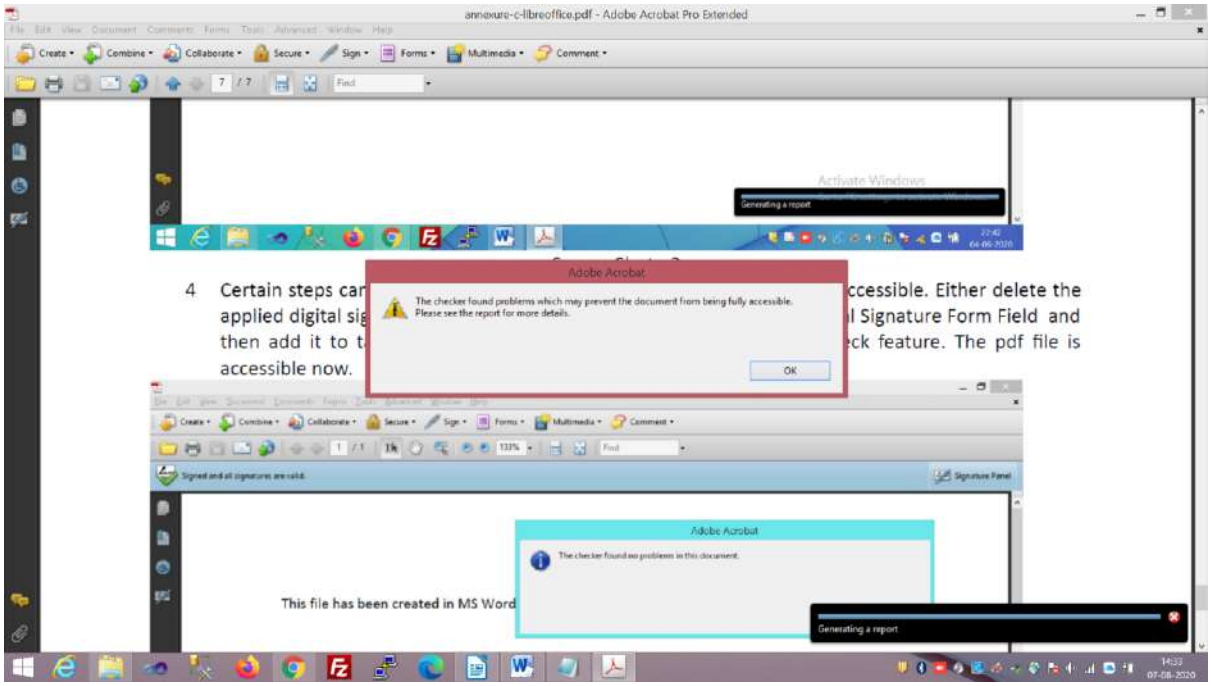
The pdf document created using Libre Office in section 2.2.2.1.1 is checked for accessibility as illustrated above and it fails accessibility criteria (Refer Figure-20, Figure-21, Figure-22 and Figure-23).



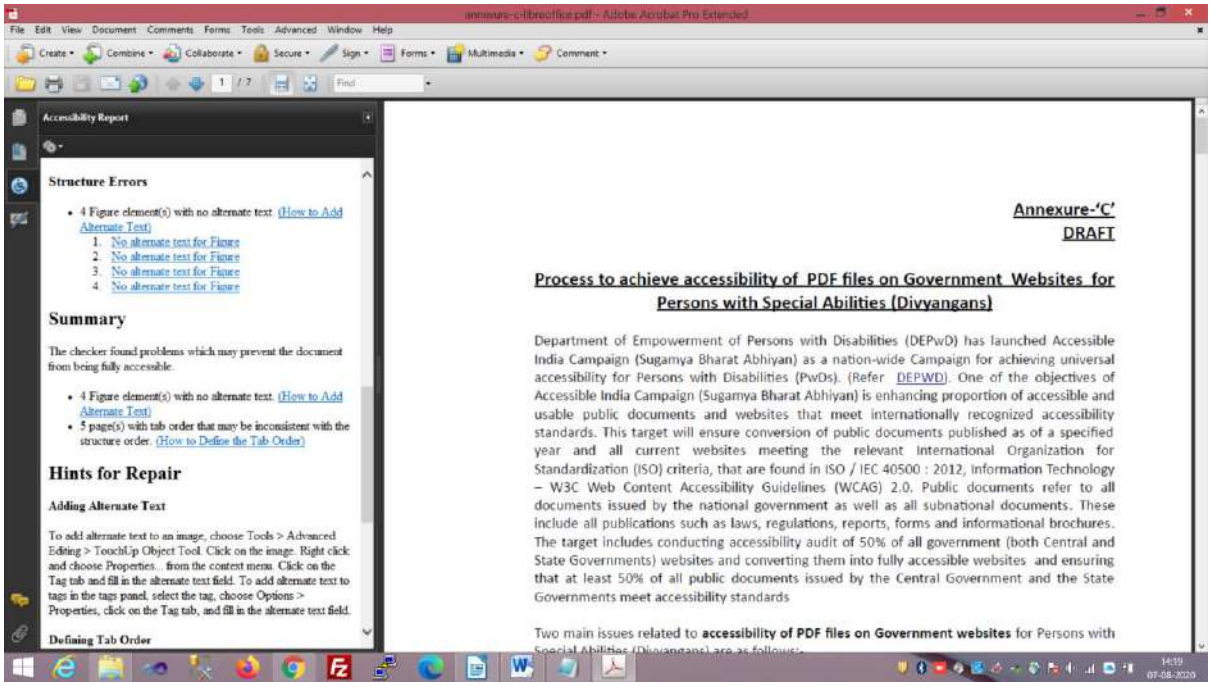
(Figure-20)



(Figure-21)



(Figure-22)

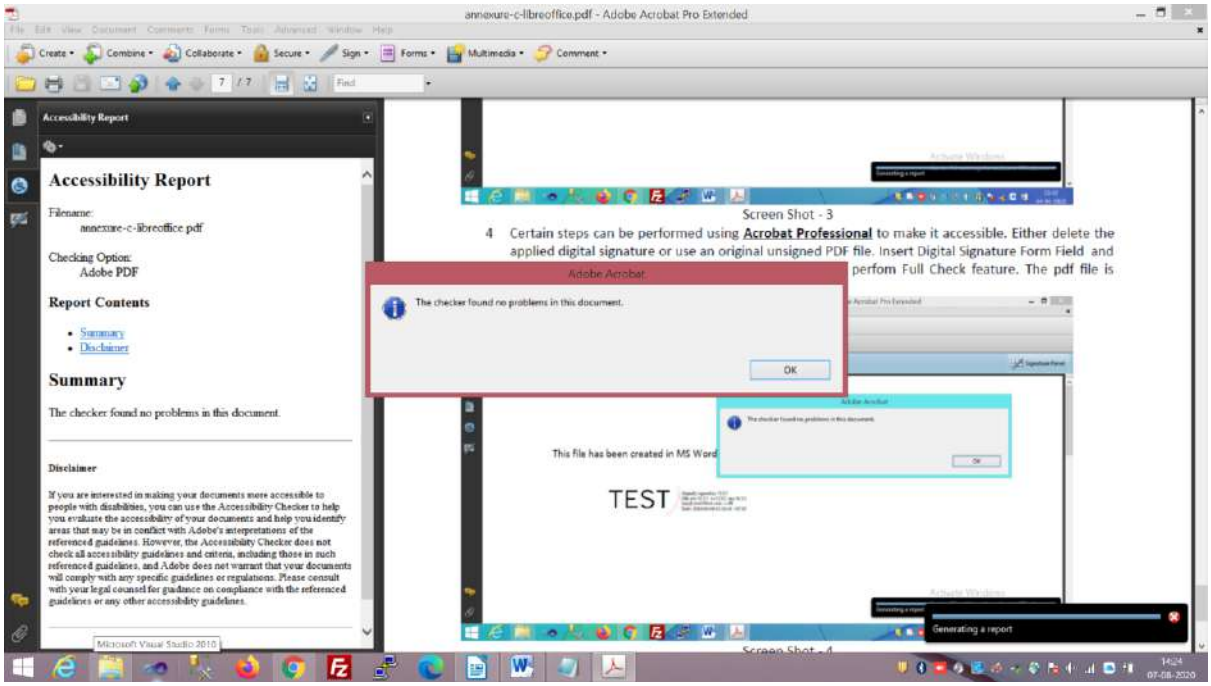


(Figure-23)

The error listed in the Accessibility Report on the left pane of Acrobat Pro can be resolved with the help of Hints for Repair mentioned there and the document can be made accessible (Refer Figure-24).

Advanced Editing tools such as "Touch-up Object Tool" are used for some type of listed repairs. The "Touch-up Reading Order Tool" provides the easiest and quickest method to fix reading order and tagging issues. The Reading Order tool is intended for repairing PDFs that were tagged using Acrobat, not for repairing PDFs that were tagged during conversion from an authoring application. Whenever possible, return to the source file and add accessibility features in the authoring application. Repairing the original file ensures that repeatedly touch up future iterations of the PDF in Acrobat will not be required (Refer here).

- ✓ **Sometimes it is quite possible that native source document of PDF in Libre Office etc. is not available and in such case editing of PDF can be done using Acrobat Pro and accessibility can be achieved by using the Full Check Feature of Acrobat Pro.**

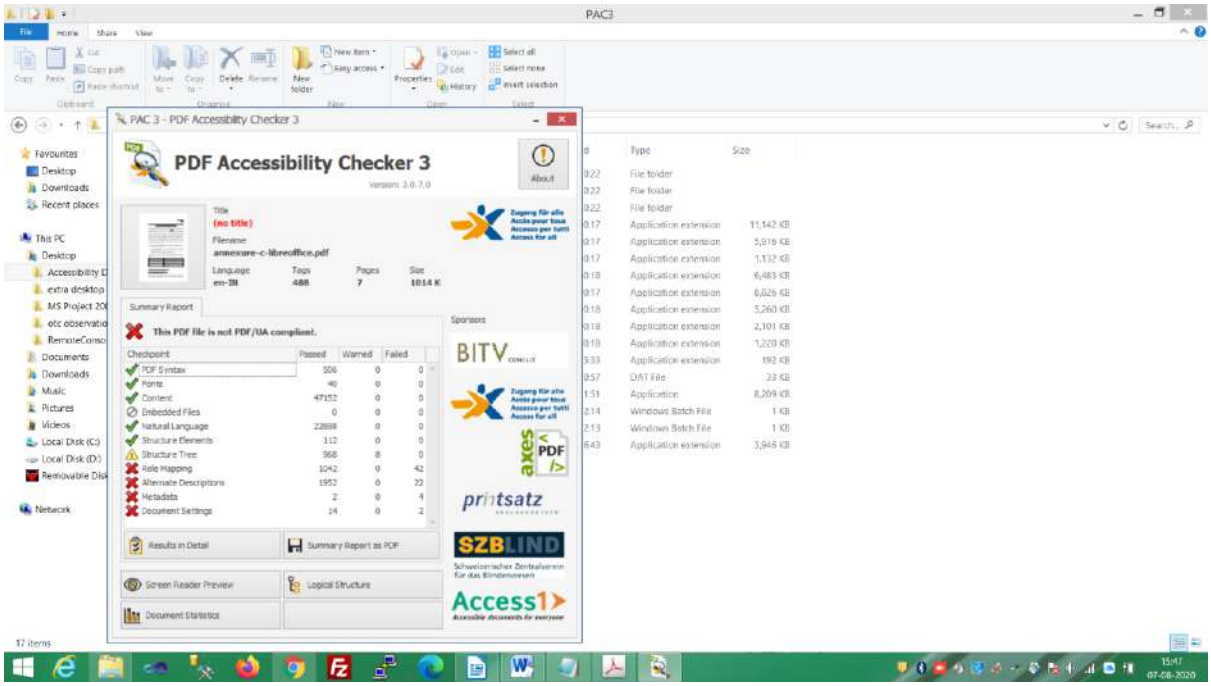


(Figure-24)

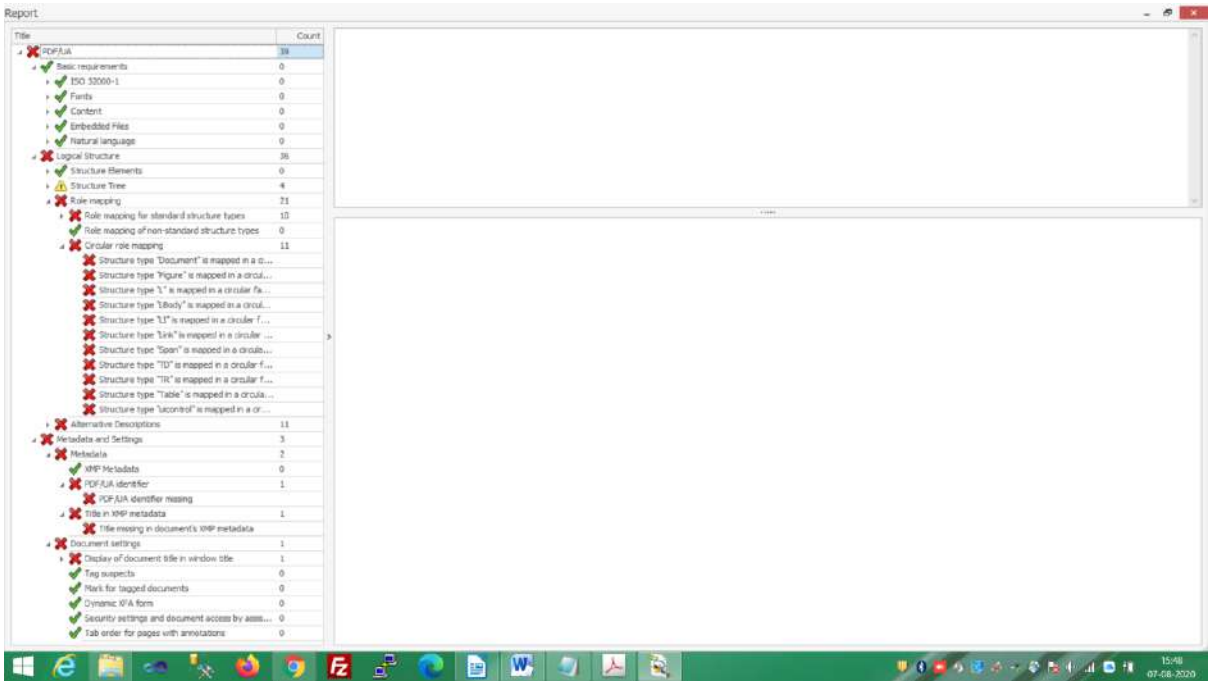
2.2.2.1.3. Using PDF Accessibility Checker (PAC3)

A shareware PDF Accessibility Checker (PAC3) can be downloaded subject to their Licence Agreement to check the accessibility of PDF documents. This checker lists the errors but it does not display hints to resolve the issue. Hence, such checker, which confirms to standards of PDF/UA (also mentioned by OTG, NIC in their observation) can be used to identify the accessibility issues.

In the following example, the document, which was declared accessible by Acrobat Pro, is rechecked using PAC 3 to find out whether the document confirms to PDF/UA standards.



(Figure-25)



(Figure-26)

It can be seen in Figure-25 & Figure-26 that the document, which was declared accessible by Acrobat 9 Pro, is not PDF/UA complaint as per PDF Accessibility Checker (PAC3).

- ✓ **In the test case mentioned above, this division of NIC. Used Libre Office 6.4.5.2 and Acrobat 9 Pro. The latest version of Acrobat Pro may exhibit PDF/UA compliance, however, we**

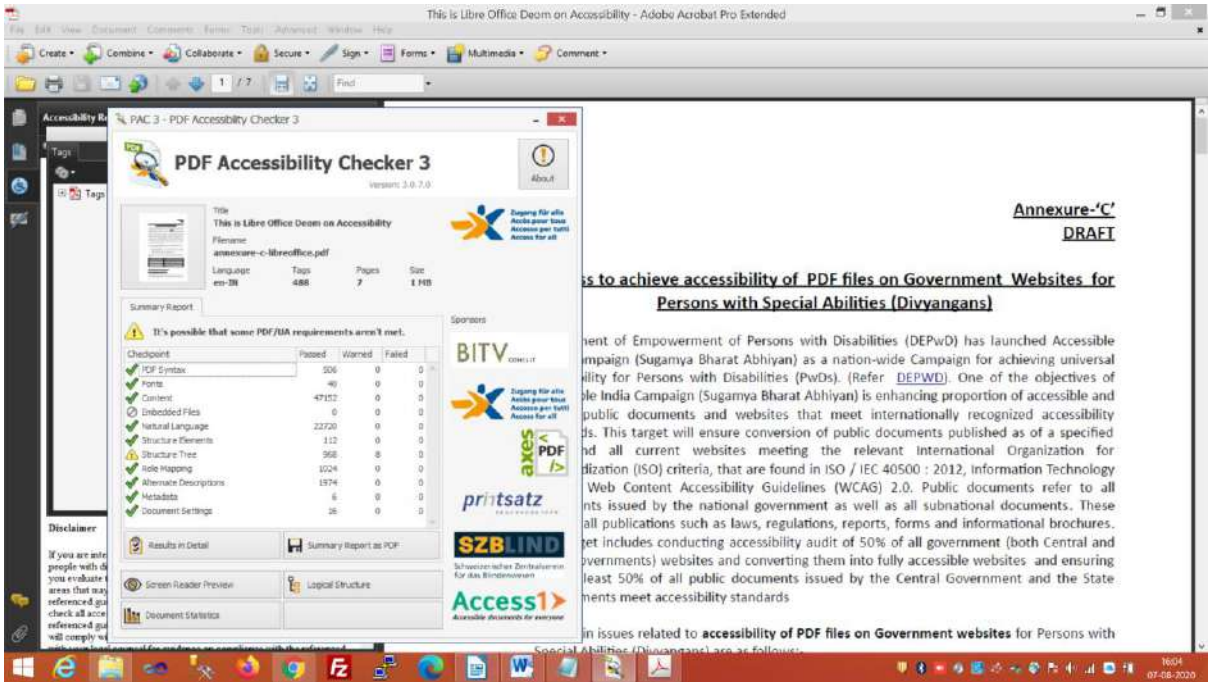
have not explored these latest version for the conformance of PDF/UA compliance using PAC3

2.2.2.2. Repairing to make it Accessible

2.2.2.2.1. Use Libre Office

Since there was no mechanism, found in Libre Office to view the accessibility issue because AccessODF extension is not compatible with the latest versions of Libre Office, so the repairing in Libre Office has been ignored.

2.2.2.2.2. Using Acrobat Pro



(Figure-27)

- ✓ **It has been observed that tagged PDF file exported from Libre Office listed fewer errors in PAC3 as compared to tagged pdf saved from MS Word although the native source document was same in both the cases.**

  The Error listed in PAC3 were resolved as shown in Figure-27

- ✓ **A Good Knowledge of Acrobat Pro is essential to achieve PDF/UA compliance. The user can also seek the help from Internet in finding the solution and fixing the issued pointed by PAC3 but this process could be time consuming.**

- ✓ **We were able to achieve PDF/UA compliance using the old Acrobat 9 Pro Extended.**

2.3. Create Accessible document from scanned images PDF Files

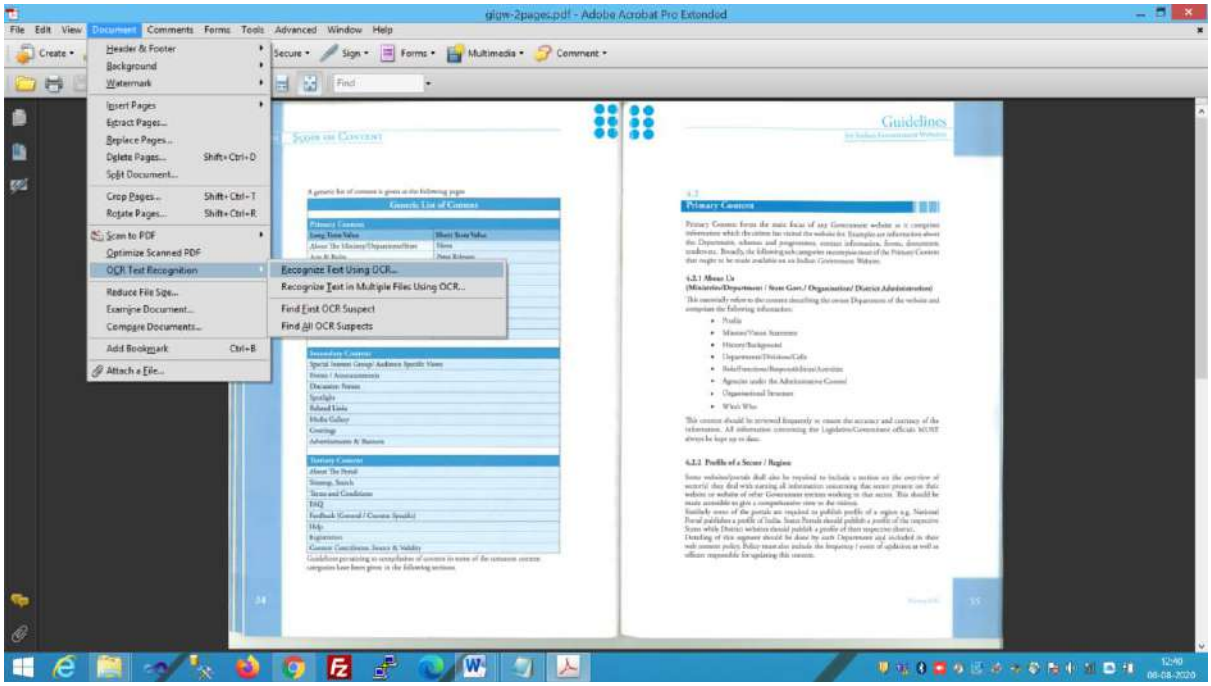
2.3.1. Using Acrobat 9 Pro

It has been observed that most of the PDF documents available on the Indian Government Websites are image scanned which have been uploaded by Content Managers after image scanning the hard copy of the documents available with them. Visually Challenged persons cannot access such documents by using Assistive Technologies.

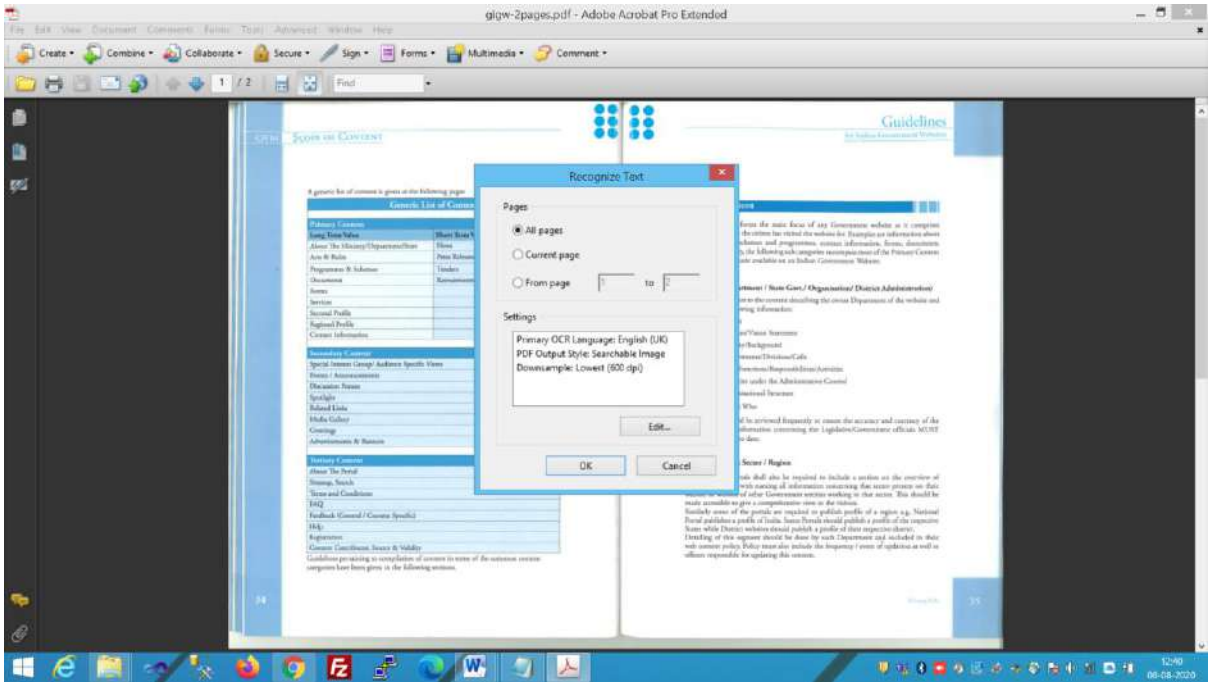
Such documents can be easily made accessible by using the Acrobat Pro. As an example, two scanned pages of GIGW manual have been used to illustrate the process.

i. **Performing OCR on a scanned PDF document to provide actual text**

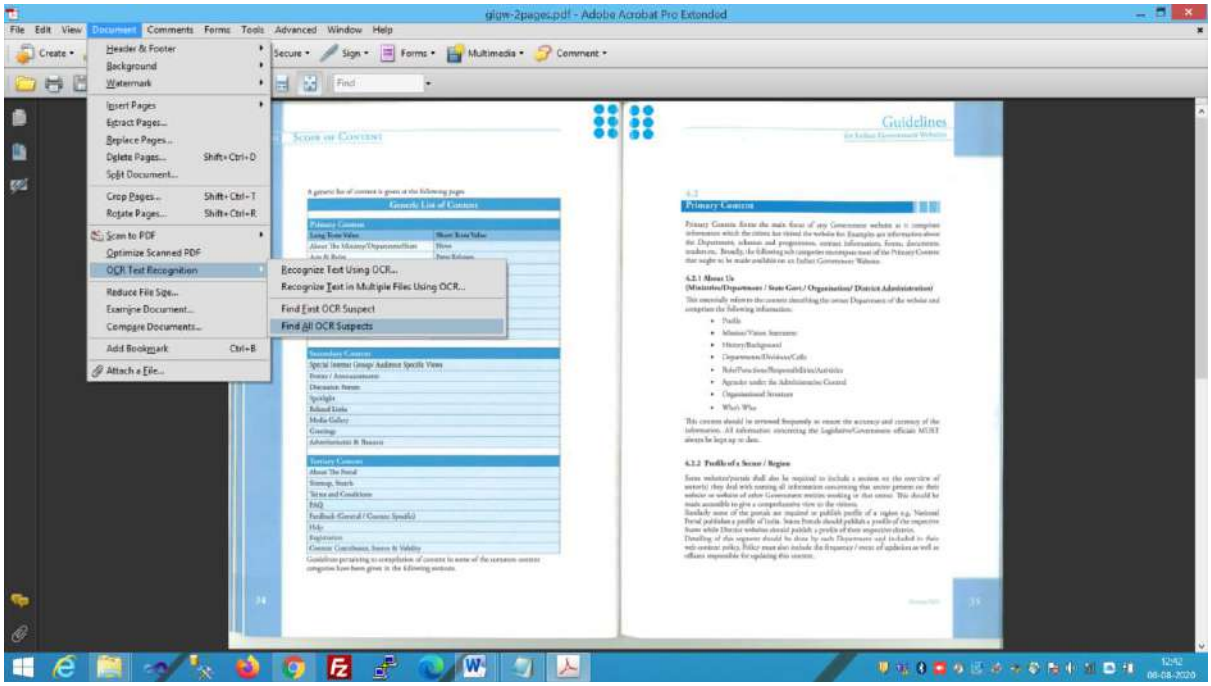
"OCR Text Recognition" feature of Acrobat Pro can be used. Depending upon resolution and clarity of Text, OCR converts images of words and characters to actual text. Text, which is not recognized by Acrobat Pro, is listed as an "OCR suspect," or text element that Acrobat suspects were not recognized correctly. The suspects can be fixed by using the options **"Find First OCR Suspects"**. These suspects are presented one at a time, which can be corrected use Acrobat Pro touch-up tools. Alternatively, **"Find All OCR Suspects"** can be used to display all OCR suspects at the same time for faster editing (Refer Figure-28, Figure-29 & Figure-30).



(Figure-28)



(Figure-29)

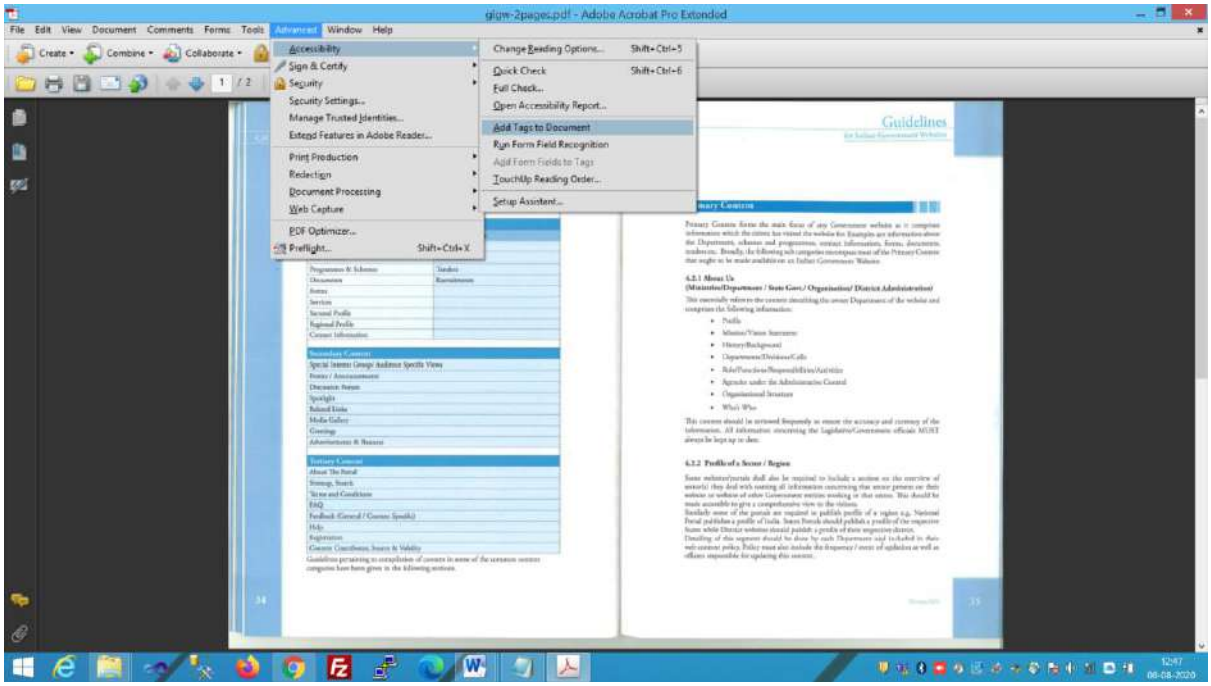


(Figure-30)

ii. **Adding Tags to document**

Tags can be added to untagged documents using Adobe Acrobat Pro. There are several ways to do this:

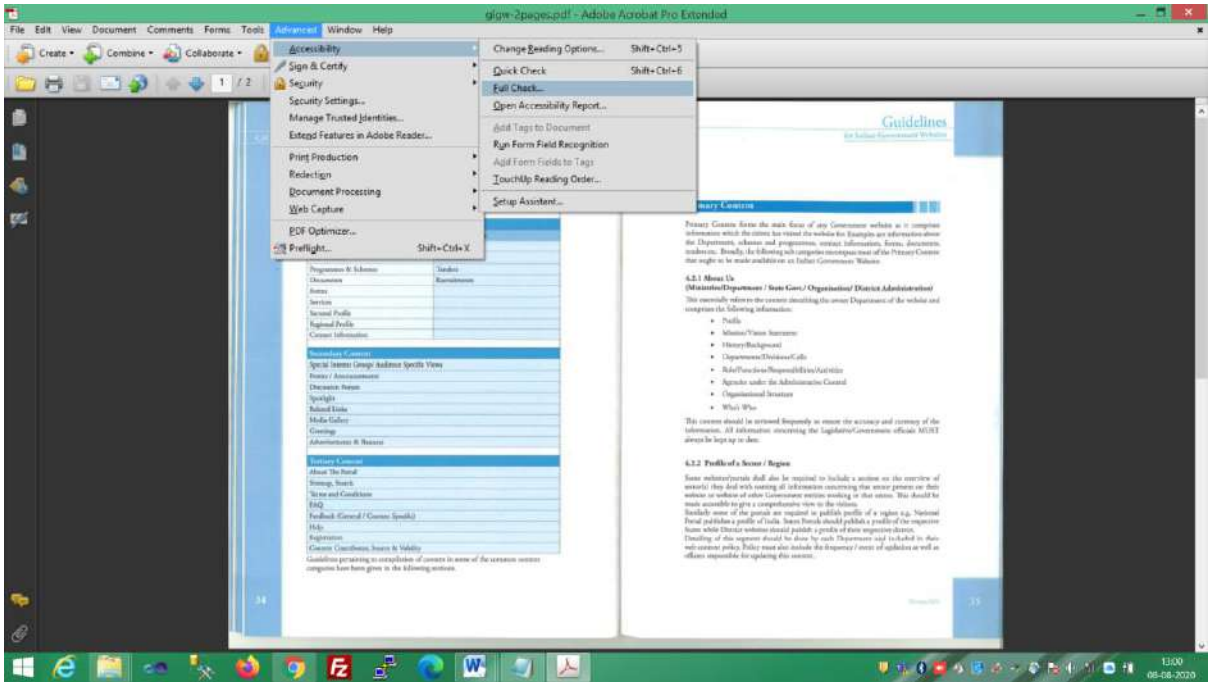
- Add Tags from the Make Accessible Action Wizard (Acrobat Pro Latest Versions).
- Add Tags from the Accessibility Checker results.
- Add Tags Manually via the Tags panel: - For example, "Add tags to the Document" feature of Acrobat Pro can be used to add tagging to the OCR document (Refer Figure-31).



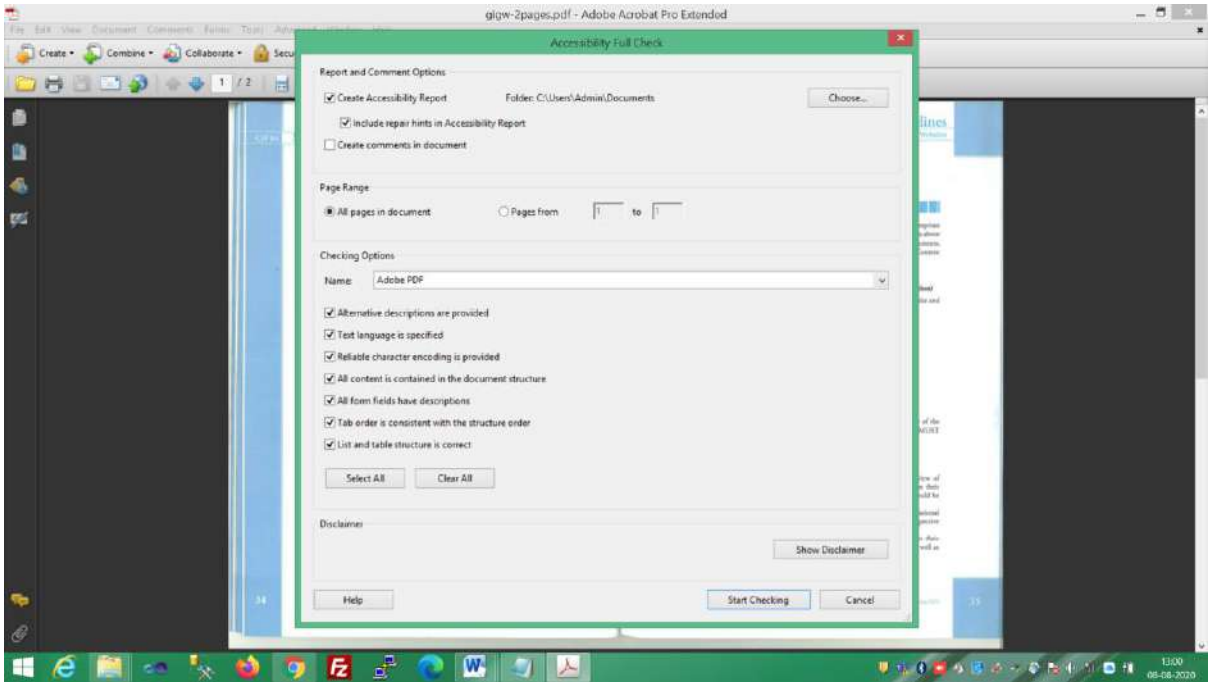
(Figure-31)

iii. **Checking the accessibility of PDF File using Acrobat Pro**

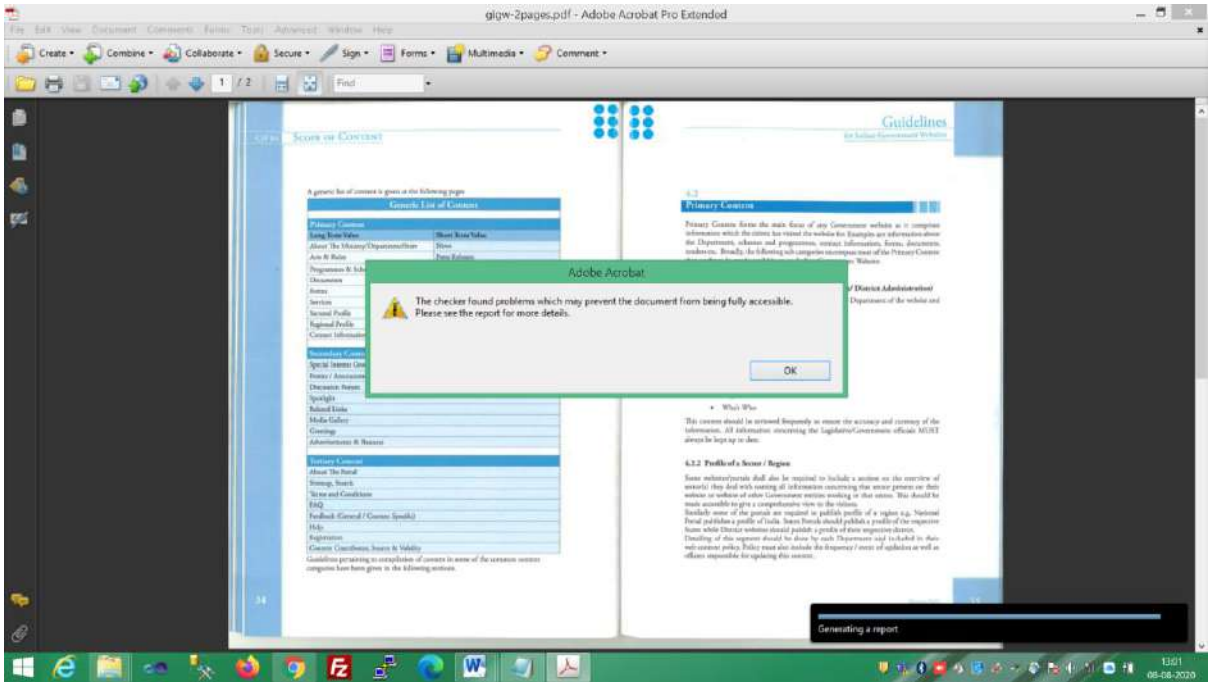
In order to check the accessibility of PDF files, **Full Check** Feature of Acrobat Pro under Accessibility can be used. The results are displayed in the Accessibility Checker panel on the left, which also has helpful links and hints for repairing issues such as Adding Tags, Character Encodings, and Alternate text, Language Attributes etc. (Refer Figure-32, Figure-33, and Figure-34 & Figure-35).



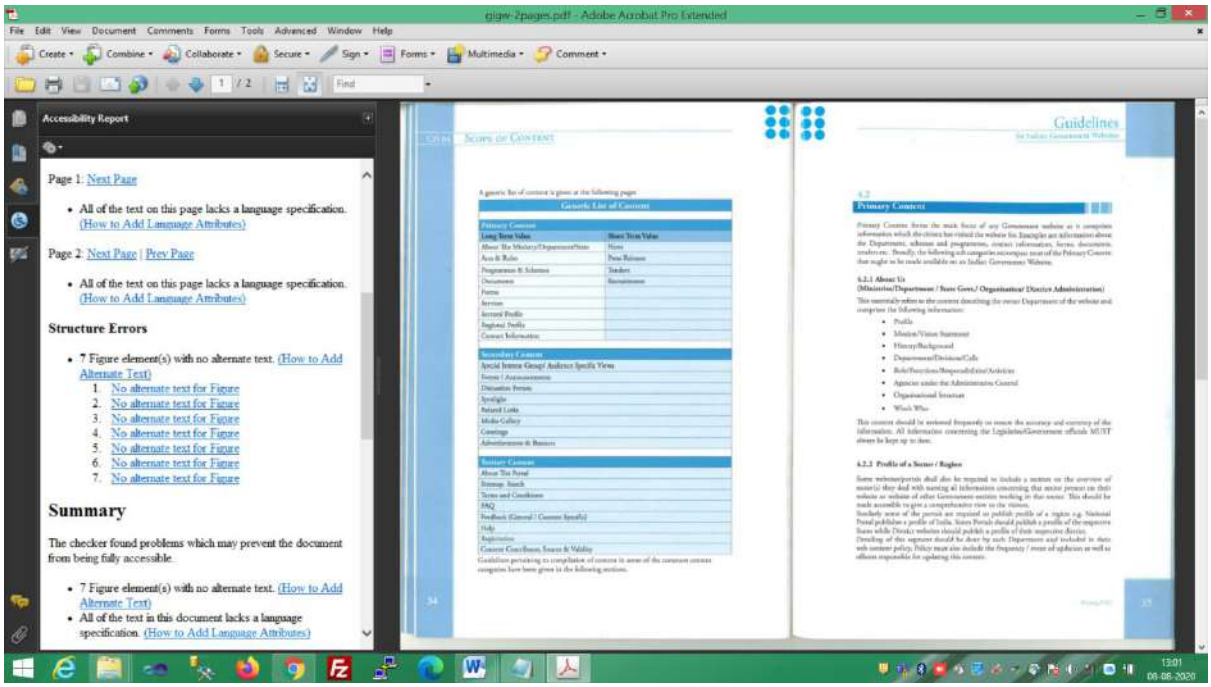
(Figure-32)



(Figure-33)



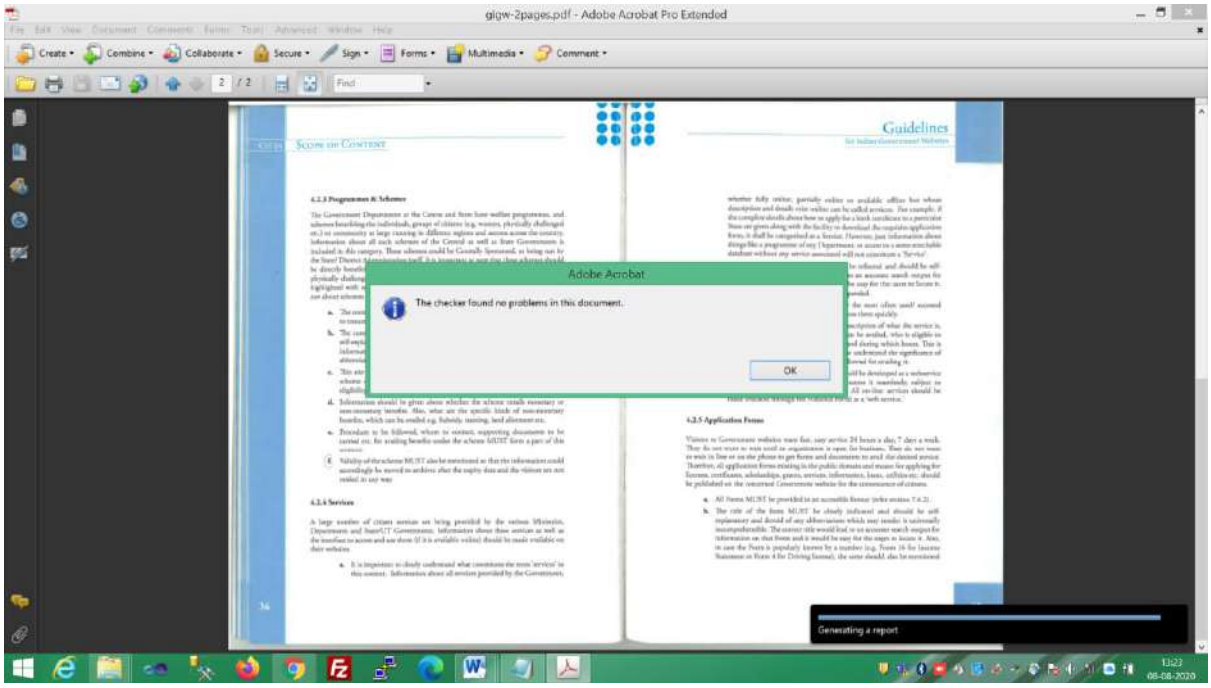
(Figure-34)



(Figure-35)

Advanced Editing tools such as "Touch-up Object Tool" are used for some type of listed repairs. The "Touch-up Reading Order Tool" provides the easiest and quickest method to fix reading order and tagging issues. The Reading Order tool is intended for repairing PDFs that were tagged using Acrobat, not for repairing PDFs that were tagged during conversion from

an authoring application. The document, which was made accessible using Acrobat Pro, can be seen in Figure-36.



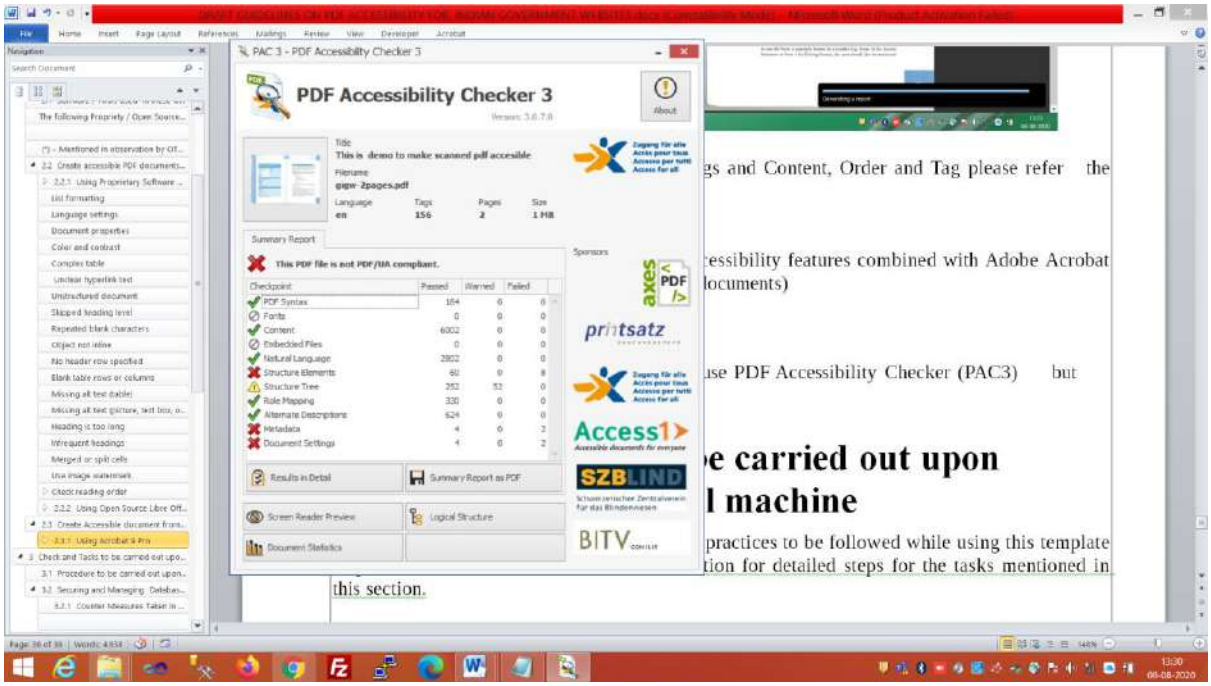
(Figure-36)

For more information on reading order of tags and Content, Order and Tag please refer the following:-

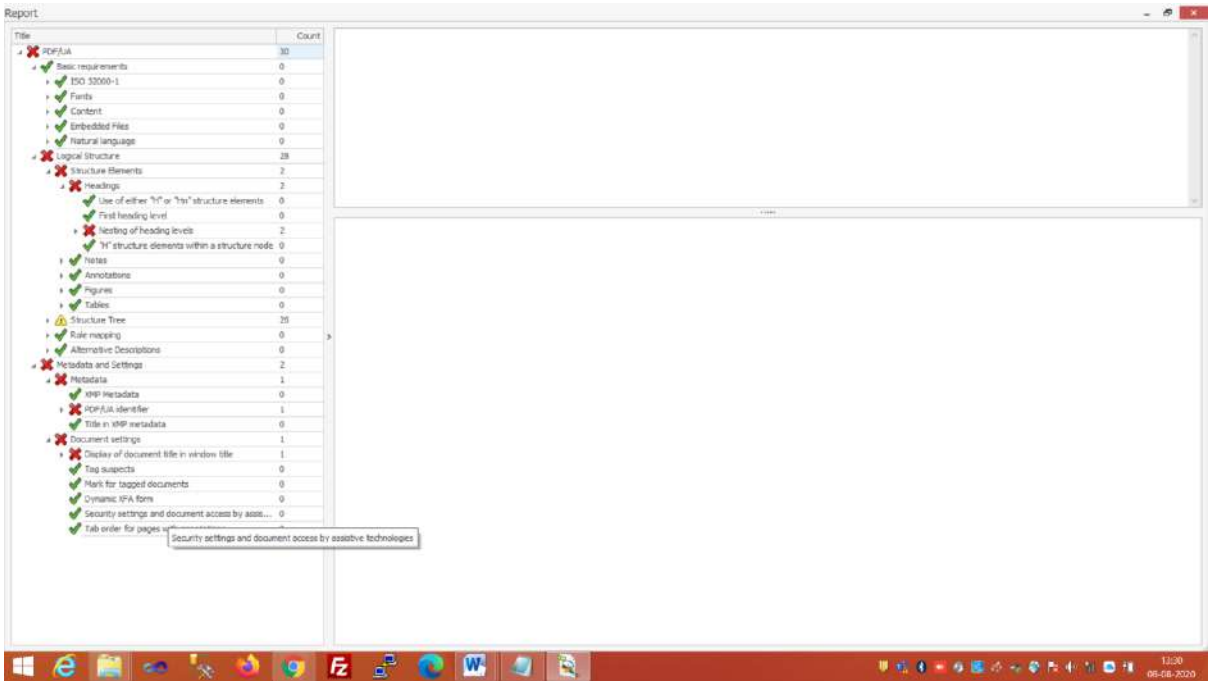
- W3C Website
- Adobe Website Help
- Adobe Accessibility (PDF file format accessibility features combined with Adobe Acrobat and Adobe Reader allow universal access to documents).

2.3.1.1. Verifying Accessibility

The PDF declared accessible by Acrobat Pro is verified use PDF Accessibility Checker (PAC3) but PDF/UA accessibility norms fails (Refer Figure-37 & Figure-38)

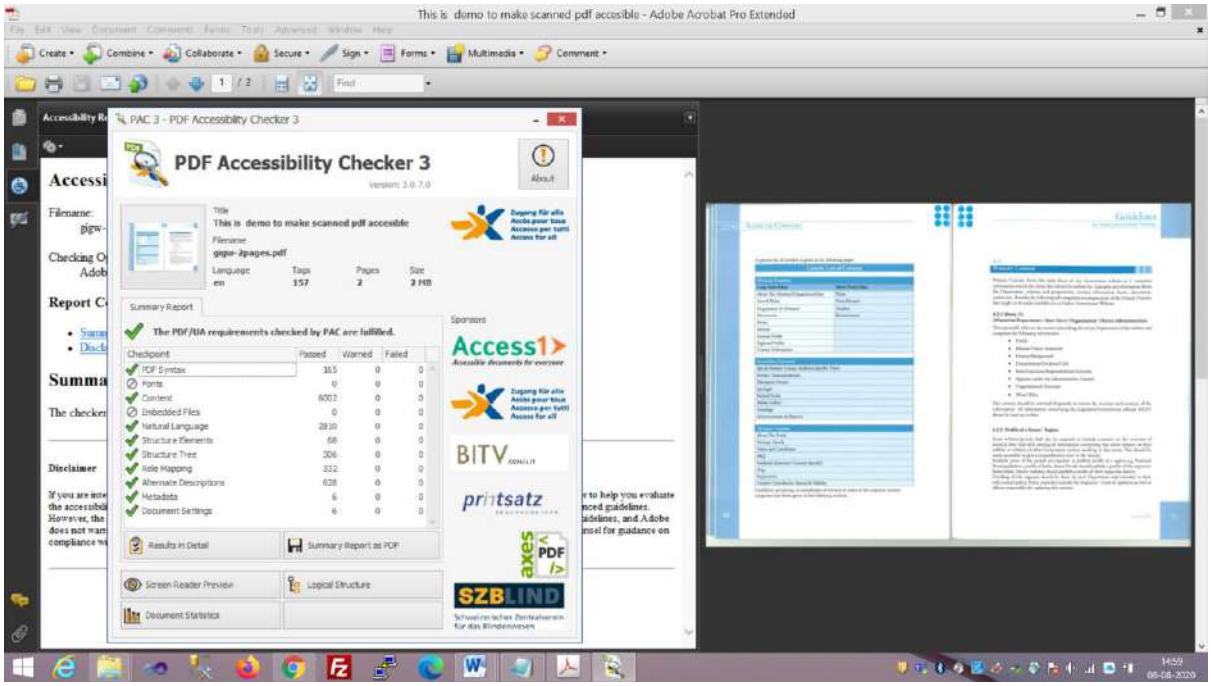


(Figure-37)



(Figure-38)

Certain steps were performed in Acrobat based on error listed by PAC3 and the pdf document was made PDF/UA complaint.



(Figure-39)

- ✓ **A Good Knowledge of Acrobat Pro is essential to achieve PDF/UA compliance. The user can also seek the help from Internet in finding the solution and fixing the issued pointed by PAC3 but this process could be time consuming.**
- ✓ **We were able to achieve PDF/UA compliance using the old Acrobat 9 Pro Extended (Refer Figure-39).**

2.4. Errors Requiring Human Inspection

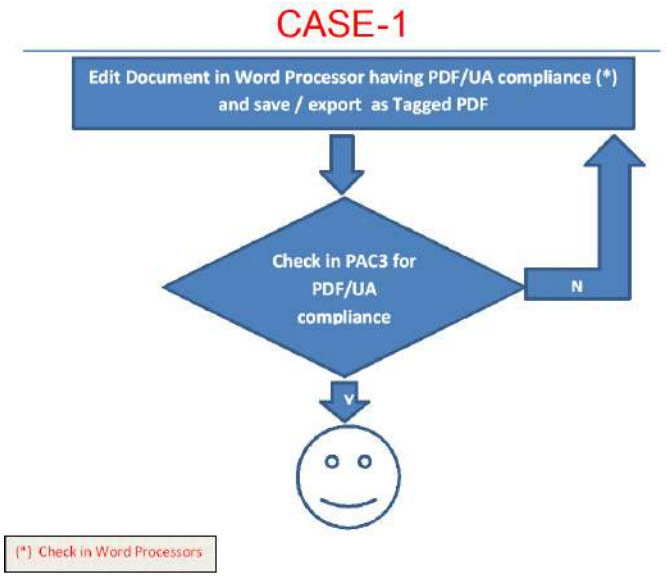
It has been observed in this document that MS Word 2010, PAC3 and Adobe Acrobat Pro are used to verify the accessibility of the PDF, however a need of a human expert to find the errors that could not be detected by the automated tools is required. Accessibility checking requires manual inspection and some human judgement (e.g. "What is meaningful alternative text for an image?"

PAC does not detect the following common PDF accessibility errors and so a human inspection is required.

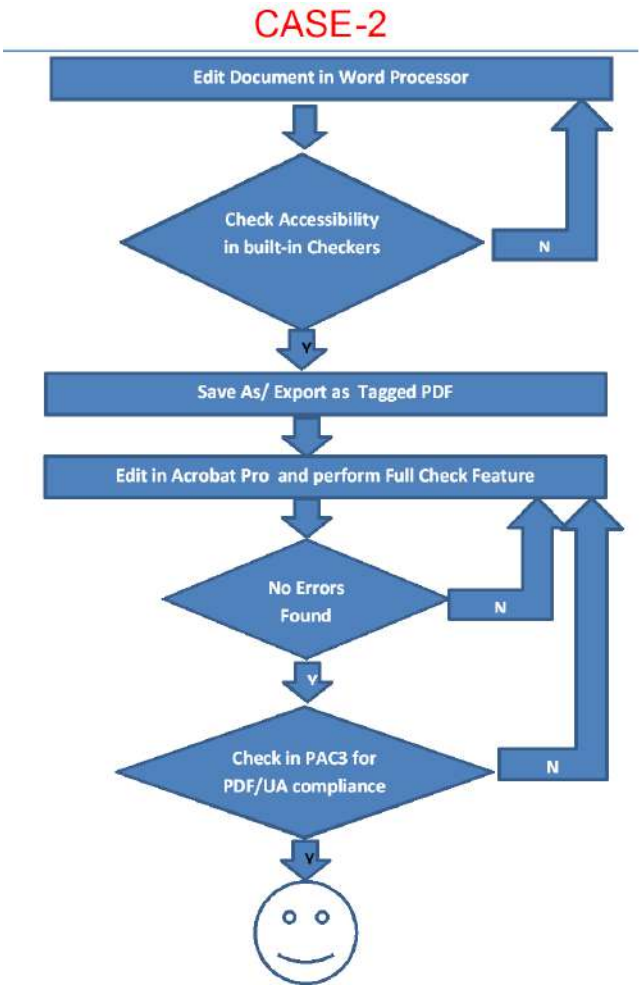
- ✓ List Numbering attribute for ordered list
- ✓ Header and footer artefacts
- ✓ Table header cell not tagged as a header
- ✓ Non-table content tagged as a table
- ✓ Complex table header IDs
- ✓ Z order problem
- ✓ Actual Text with null string
- ✓ Actual Text – Alt Text – Expansion Text – Contents Key
- ✓ I ran out of heading levels!
- ✓ Insufficient contrast for text

2.5. Draft Process to obtain Accessible PDF

2.5.1. Using Word Processors

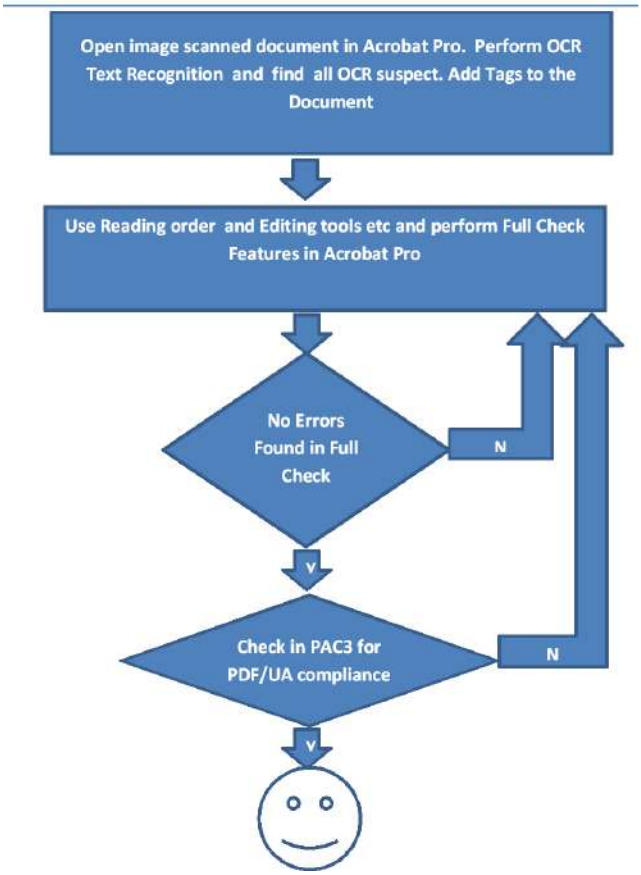


(Process Flow Chart-1)



(Process Flow Chart-2)

2.5.2. Using Image Scanned PDF



(Process Flow Chart-3)

2.5.3. Using latest versions of Word Processors & Acrobat Pro

- **Please Note: - We have used MS Word 2010, Acrobat 9 Pro and Libre Office 6.4.5.2 in this document to achieve accessibility of PDF files. However, it is advisable to check for latest versions of MS Word and Acrobat Pro for more accessibility feature such as direct PDF/UA compliance and in such cases, the use of PAC3 can be ignored. Users of this document can take a view on this by referring the manual and features of the latest versions of software if being used by them.**

3. Using Other Open Source Tools

This section provides information on some of the open source tools mentioned in the observations of OTG, NIC and the best practices that can be followed.

3.1. PyPDF and pytesseract

Both PyPDF and pytesseract requires python platform and therefore we have not evaluated its use, because it has been observed that most of content managers do not have an expertise to work on Python Platform.

3.2. OCRFeeder

OCRFeeder works on UNIX and therefore its use has not been evaluated by us, as most of content manager's works on Windows / MAC platforms.

3.3. VietOCR.Net

VietOCR.Net required .NET platform and therefore we have not evaluated its use, as most of content managers do not have such technical expertise.

3.4. Tesseract

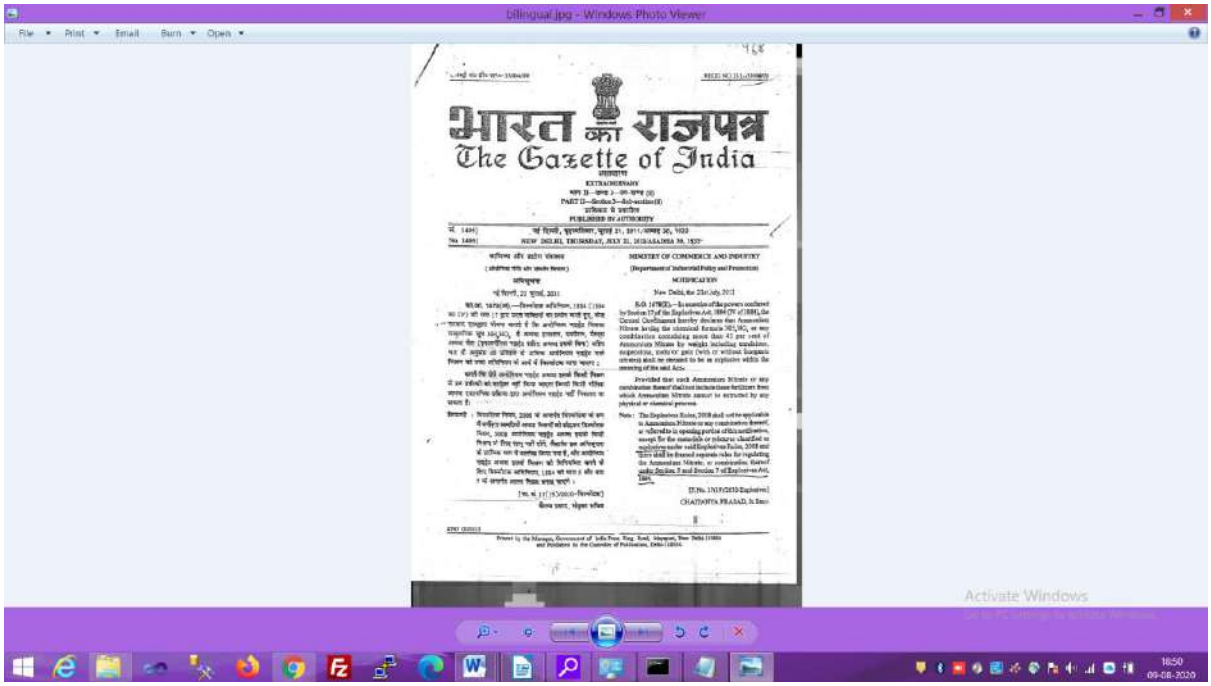
This package contains an OCR engine - libtesseract and a command line program - tesseract. Tesseract 4 adds a new neural net (LSTM) based OCR engine which is focused on line recognition, but also still supports the legacy Tesseract OCR engine of Tesseract 3 which works by recognizing character patterns. Compatibility with Tesseract 3 is enabled by using the Legacy OCR Engine mode (--OEM 0). It also needs trained data files which support the legacy engine, for example those from the tessdata repository.

Tesseract has Unicode (UTF-8) support, and can recognize more than 100 languages "out of the box".

Tesseract supports various output formats: plain text, hOCR (HTML), PDF, invisible-text-only PDF, TSV. The master branch also has experimental support for ALTO (XML) output.

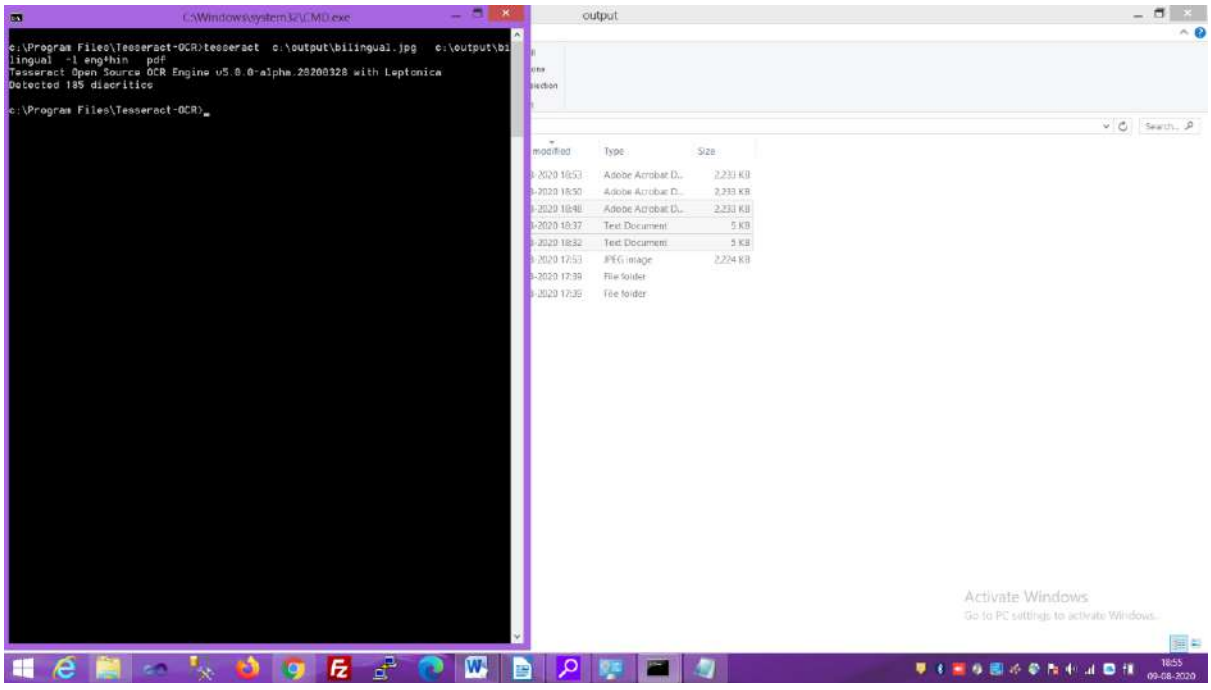
It should be noted that in many cases, in order to get better OCR results, there is a need to improve the quality of the image given to Tesseract.

Tesseract command line takes images as an input. In case, if available information is available in PDF file, then some tool such as Imagemagick may be used to convert it into an image file. (**Learn Tesseract**). Refer Figure-40 where an image file has been shown in Windows Photo Viewer.



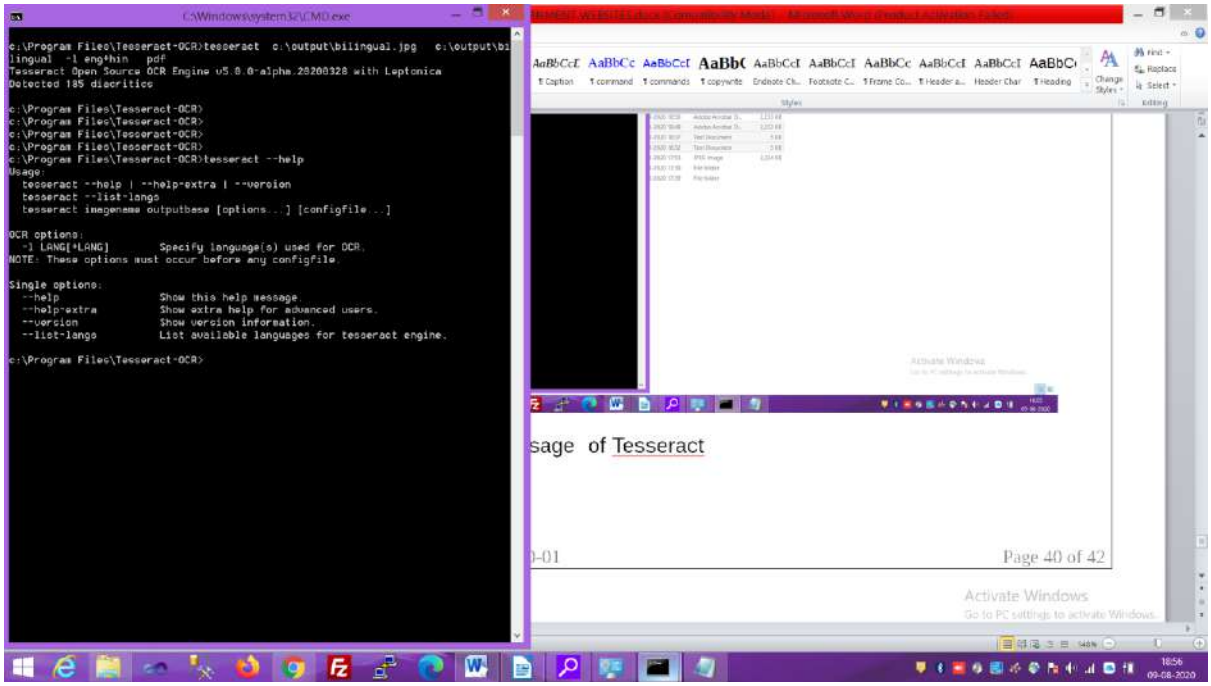
(Figure-40)

It can be converted to PDF file using tesseract command line as shown in the following Figure-41.



(Figure-41)

The command line usage/help of Tesseract to convert an image to PDF file is shown below in Figure-42.



(Figure-42)

- ✓ **Though Tesseract was able to create searchable PDF but when checked with Acrobat Pro and PAC3, Tesseract did not tag the document therefore for PDF/UA compliance, editing with Acrobat and verifying with PAC3 is required as explained in section 2.3**
- ✓ **It is to mention that an Image file can also be converted to Text file using Tesseract and the generated Text file can be accessed using Assistive Technologies.**

Tips for better recognition results:

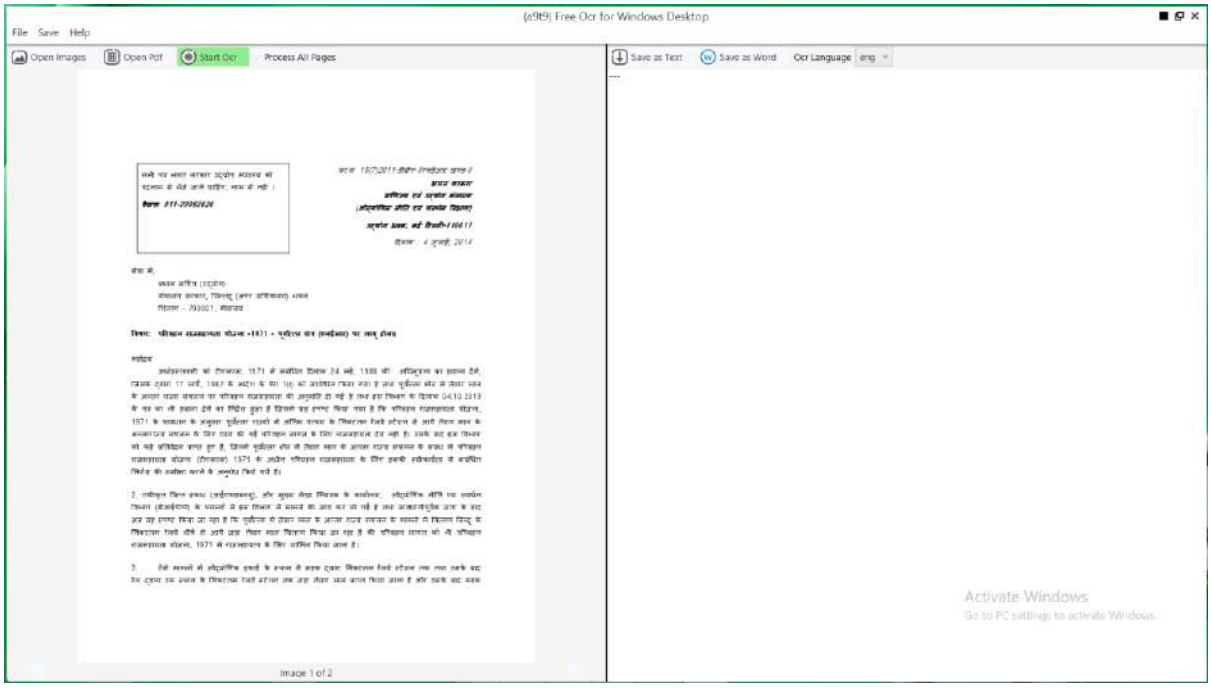
- Tesseract's output will be very poor quality if the input images are not pre-processed to suit it:
- Images (especially screenshots) must be scaled up such that the text height is at least 20 pixels.
- Any rotation or skew must be corrected or no text will be recognized,

Dark borders must be manually removed, or they will be misinterpreted as characters.

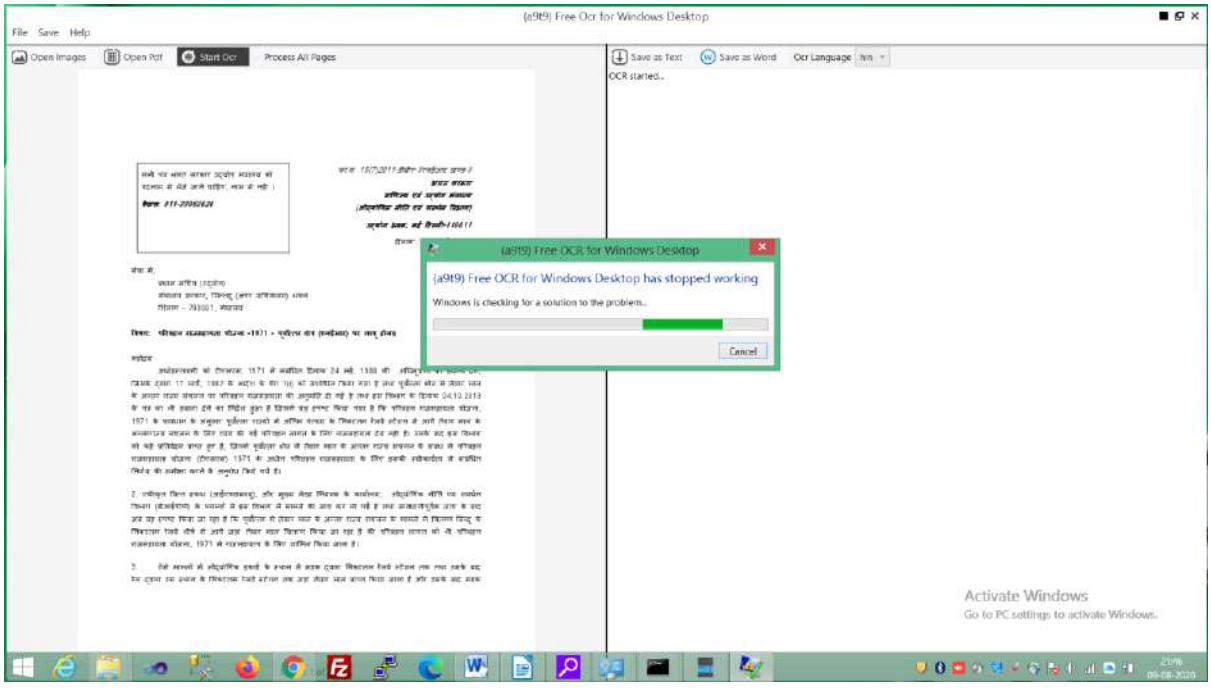
3.5. FreeOCR (a9t9)

Free OCR application for the Windows Desktop - Essentially a graphical user interface (GUI) for the Tesseract OCR engine. The application also includes support for reading and doing OCT of PDF files. It has been observed that the conversion quality is not very good. This software takes Image or PDF as an input and after OCR, the output can be saved as a text or MS Word file. The output can then be converted to accessible PDF using Word Processor, PAC3 and Acrobat Pro, if required.

We could not achieve OCR of Hindi and English+Hindi bilingual documents even though Hindi trained data file was added in the tessdata language folder (Refer Figure-43 and Figure-44).

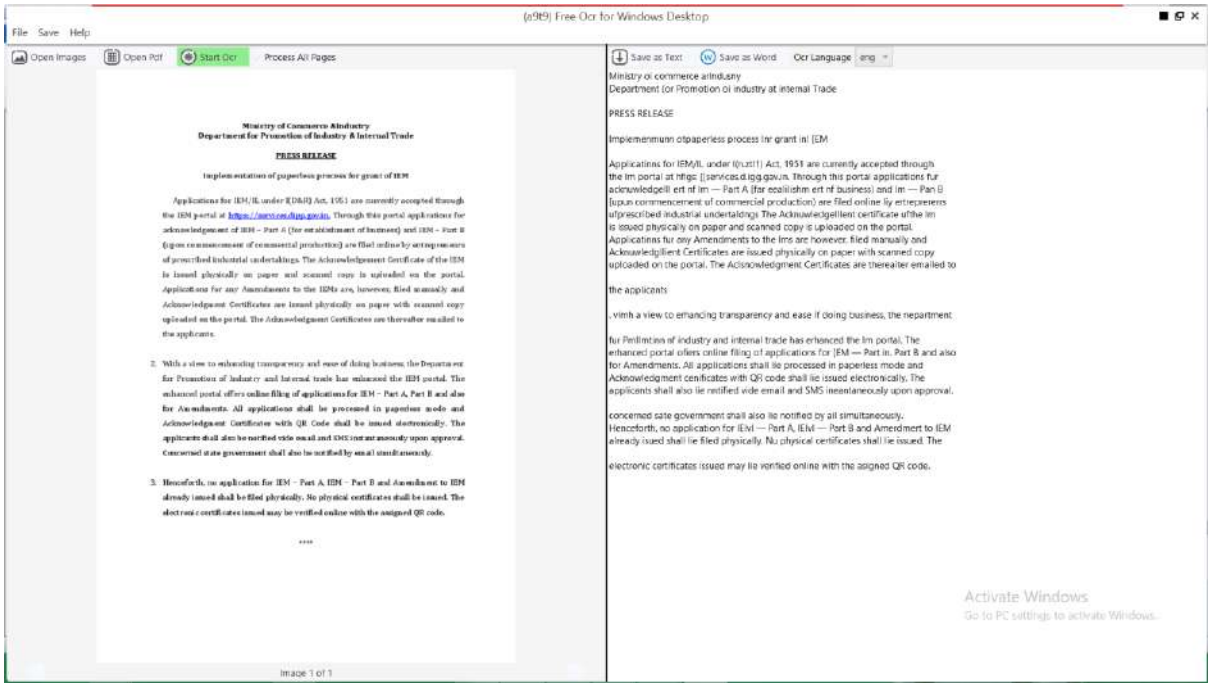


(Figure-43)

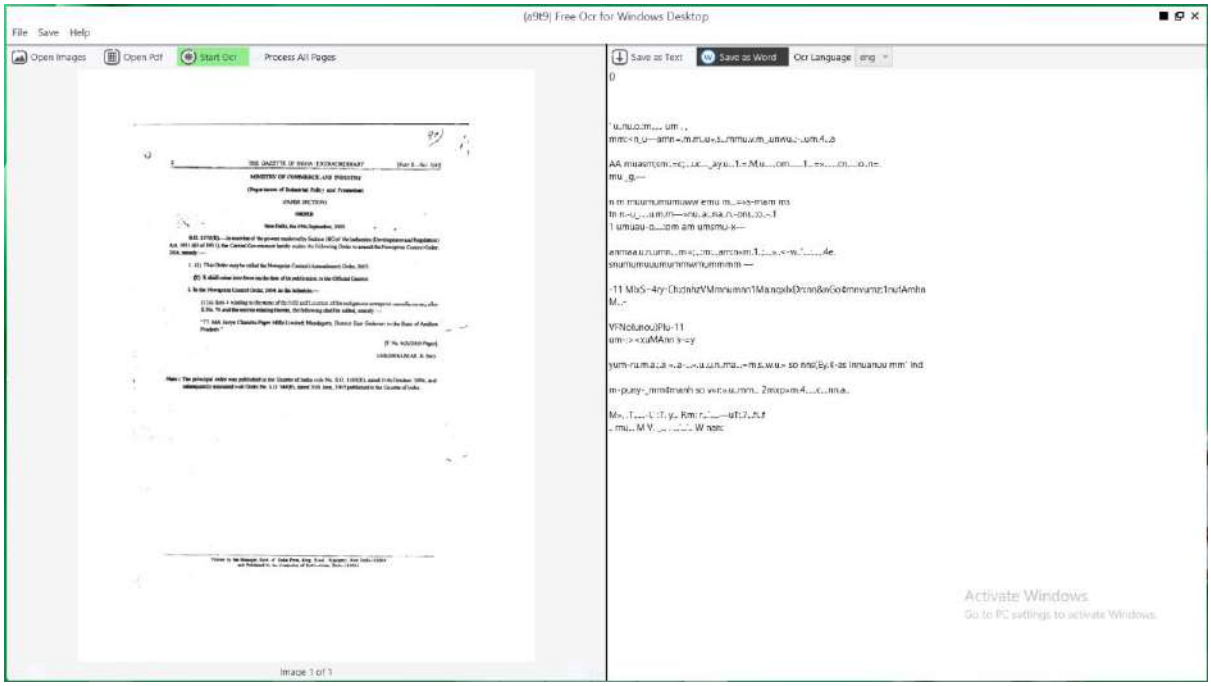


(Figure-44)

On doing OCR a good quality and resolution, scanned PDF with English text resulted in a text file with little distortion. OCR of a bad quality and resolution scanned PDF with English text resulted in a text file with complete distortion (Refer Figure-45 and Figure-46)

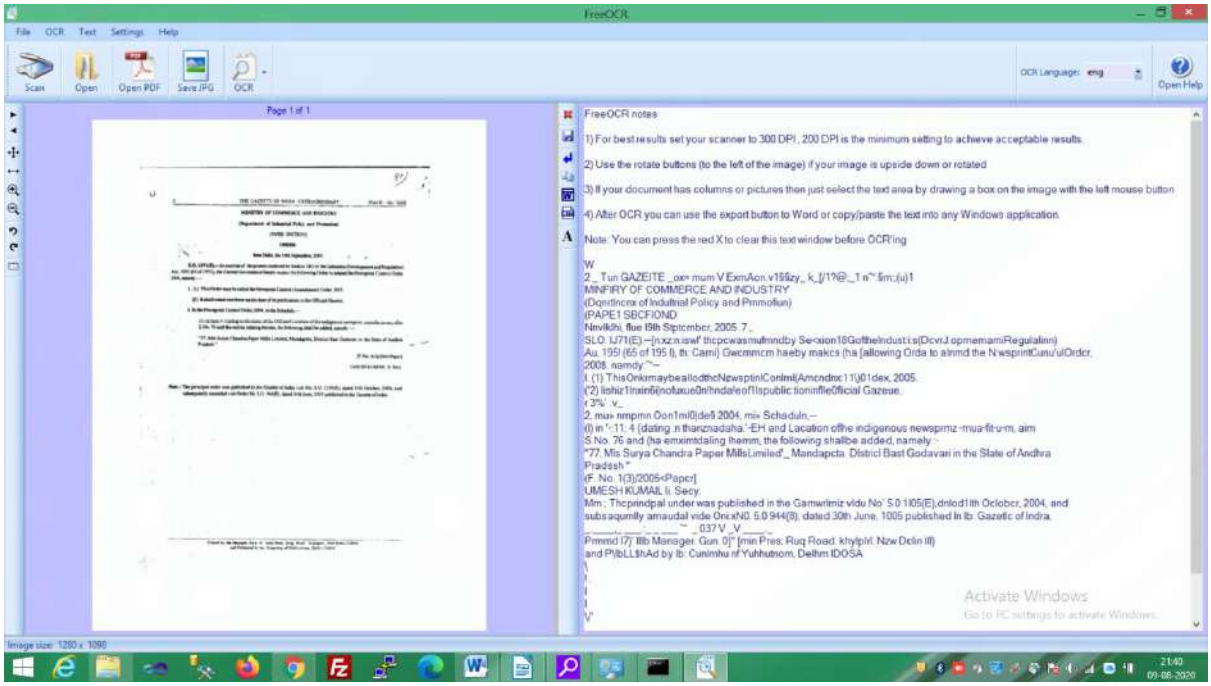


(Figure-45)

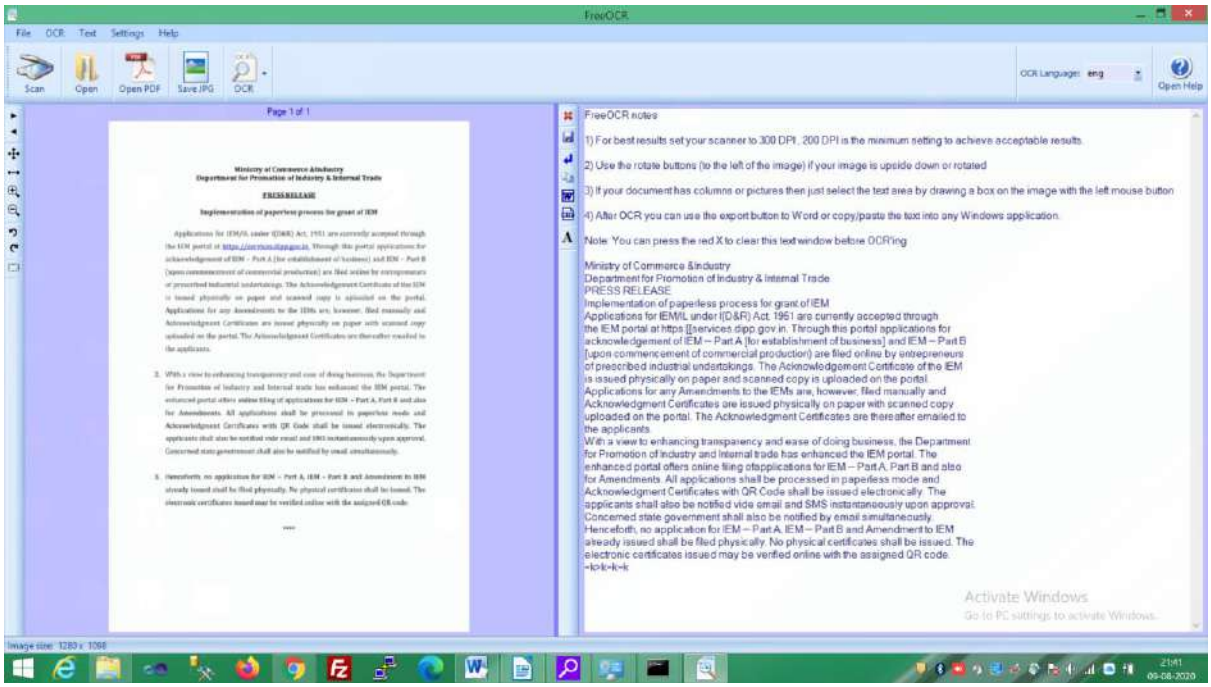


(Figure-46)

One more FreeOCR downloaded from **FreeOCR** was able to OCR a bad and good quality image scanned PDF to a partial distorted and better quality text respectively (Refer Figure-47 & Figure-48).



(Figure-47)



(Figure-48)

3.6. Imagemagick

Imagemagick can be used to convert a PDF document to image. The resultant output of multiple pages PDF can be appended in a single file or as separate image file for each page of PDF file

3.6.1. Convert an entire PDF to an single image

Command

Convert -density 150 -antialias "input_file_name.pdf" -append -resize 1024x -quality 100 "output_file_name.png"

Key parameter here is -append which actually makes a difference if PDF is converted to a single image or to a series of images.

3.6.2. Convert a PDF document to a series of enumerated images.

Command

convert -density 150 -antialias "input_file_name.pdf" -resize 1024x -quality 100 "output_file_name.png"

Because of this command, a series of image files named output_file_name-0.png, output_file_name-1.png, output_file_name-2.png .... etc., will be created in the working directory. If having more than 10 pages, it would come in handy to have those enumerated file names with multiple digits, for the convenience of easy sorting. If including a C-style integer format string, for example if adding %03d to the end of output file name, the result will be output_file_name-001.png, output_file_name-002.png, output_file_name-003.png, etc.

3.6.3. Convert only specified pages to images:

Command

convert "input_file_name.pdf[1]" "output_file_name.png"

This will actually convert page 2 of PDF to PNG, since numbering starts with 0. To convert range of pages, from i to j, use this command:

convert "input_file_name.pdf[i-j]" "output_file_name.png"

- ✓ **It is to mention that the output generated from Imagemagick can be fed to tesseract to generate resultant OCR based PDF file which can be converted to PDF/UA complaint using additional software such as Word Processor, Acrobat Pro and PAC3**

4. Digitally Signing a Document

MeitY OM No 18(3)/2018-E-Infra (Pt.) dated 10th December 2019 (Annexure -'A') requesting Secretaries of all Central Ministries/Departments and Chief Secretaries of States/UTs to make the

public documents accessible on Government websites. It has been suggested that the all the Government notifications/ orders uploaded on the website should be digitally signed and in ePub or OCR based PDF only along with a technical write-up regarding conversion.

The document can be signed either in the native source document such as MS Word or Libre Office or in the PDF document using Acrobat Pro.

4.1. Using native source document in MS Word or Libre Office

4.1.1. Using Ms Word

- Open Document.
- Insert Signature Line.
- Fill the Signature setup and sign after selecting the certificate.
- Create an accessible PDF and check its accessibility as explained in Section 2.2

4.1.2. Using Libre Office

- Go to File
- Select Digital Signature
- Select Sign Document
- Select the Certificate from list
- Provide Description, if any
- Click Sign
- Create an accessible PDF and check its accessibility as explained in Section 2.2

4.2. Using Acrobat Pro

- First remove all tags
- Add Digital Signature Form Field
- Add tags to Documents
- Remove Errors
- Apply Signature
- Perform Full Check Feature

OR

- Add Digital Signature Form Field
- Add Form fields to tags(Check for unsigned annotations and tag the element then Choose form field type while tagging and give title)
- Remove Errors
- Apply Signature
- Perform Full Check Feature

5. Using Assistive Technologies

5.1. Using Read Loud Feature of Acrobat Reader/ Pro

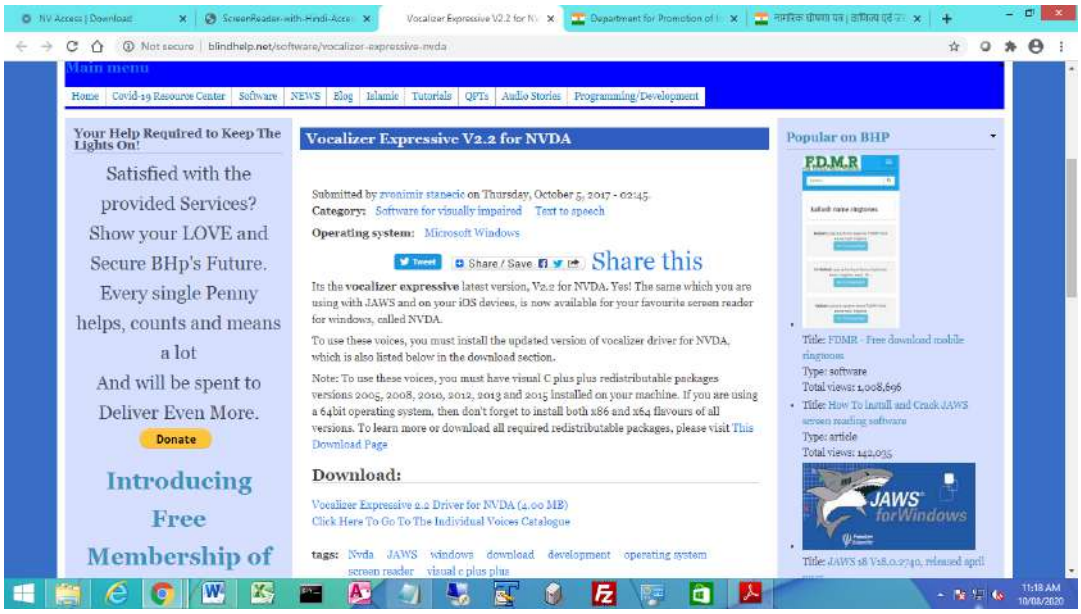
It has been observed that Acrobat Pro / Reader is suitable for reading English text. **Although a /Lang entry can be set in Acrobat Pro to specify the language for a paragraph or for a specific word or phrase etc. However, we could not listen to Hindi text using Read Loud feature of Acrobat Reader or Pro**. As far as English is concerned, the Read Loud feature works perfectly. Refer W3C Webpage. Since Adobe includes 16 languages so for Hindi, please use its ISO 639 code instead of name.

5.2. Using NVDA

This Screen Reader is able to read English or Hindi contents. To invoke Hindi language for reading, please follow the following procedure (Refer Figure-49 to Figure-58).

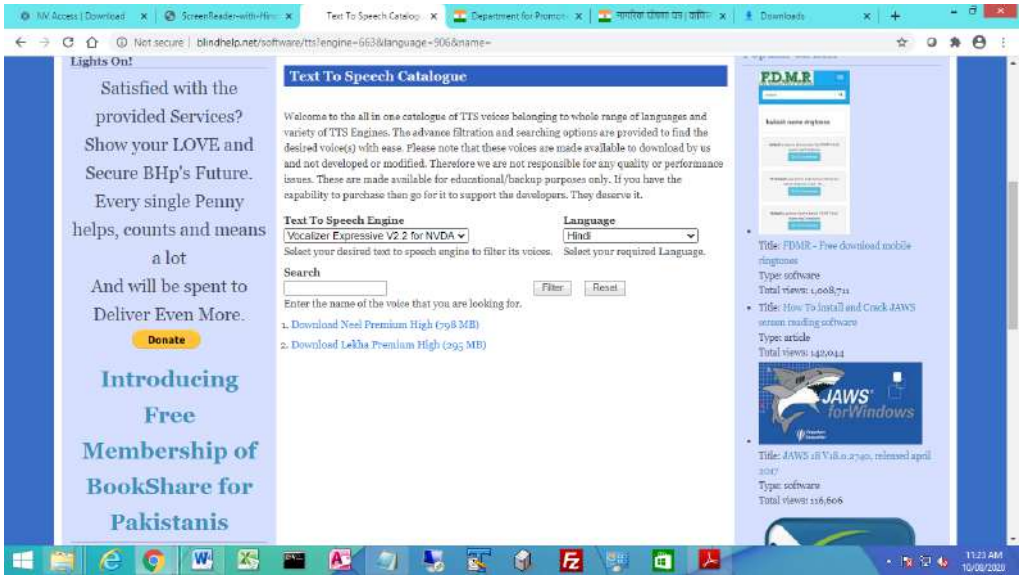
- Download two Driver to support Hindi language

  Download from given first line link



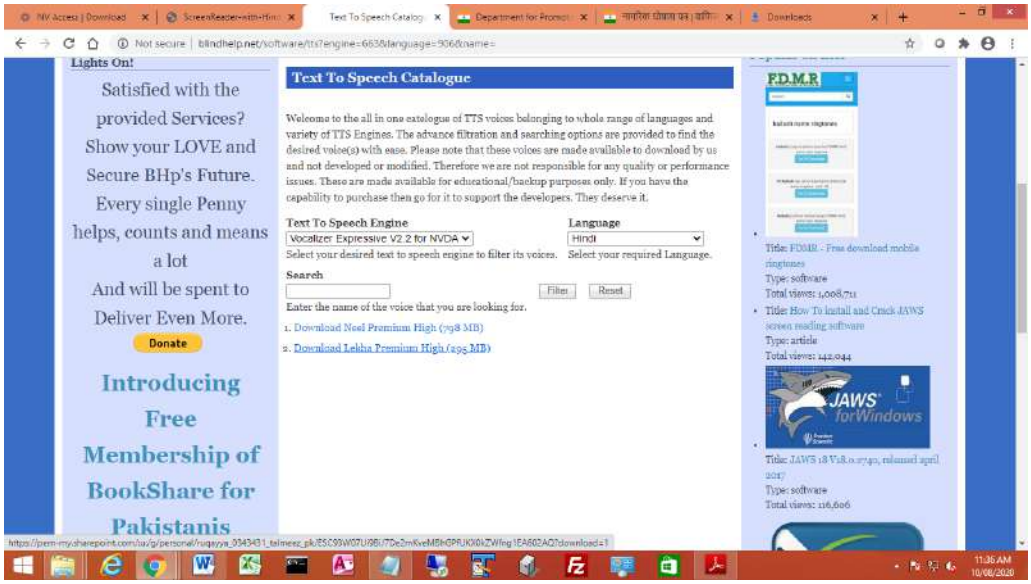
(Figure-49)

- Click To Go to The Individual Catalogue on above figure.



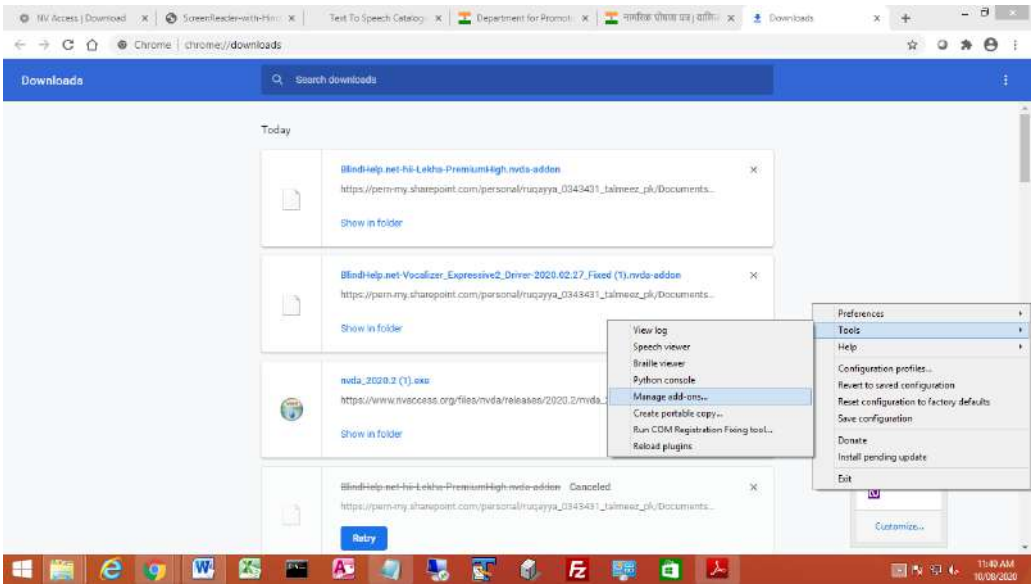
(Figure-50)

- Choose Hindi and Filter from above Figure.



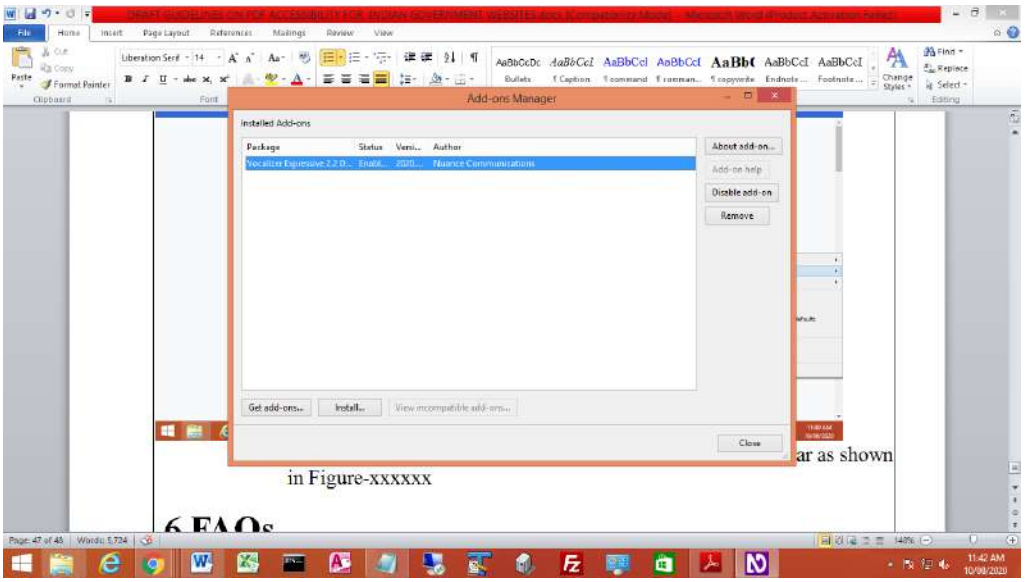
(Figure-51)

- Download Lekha Premium



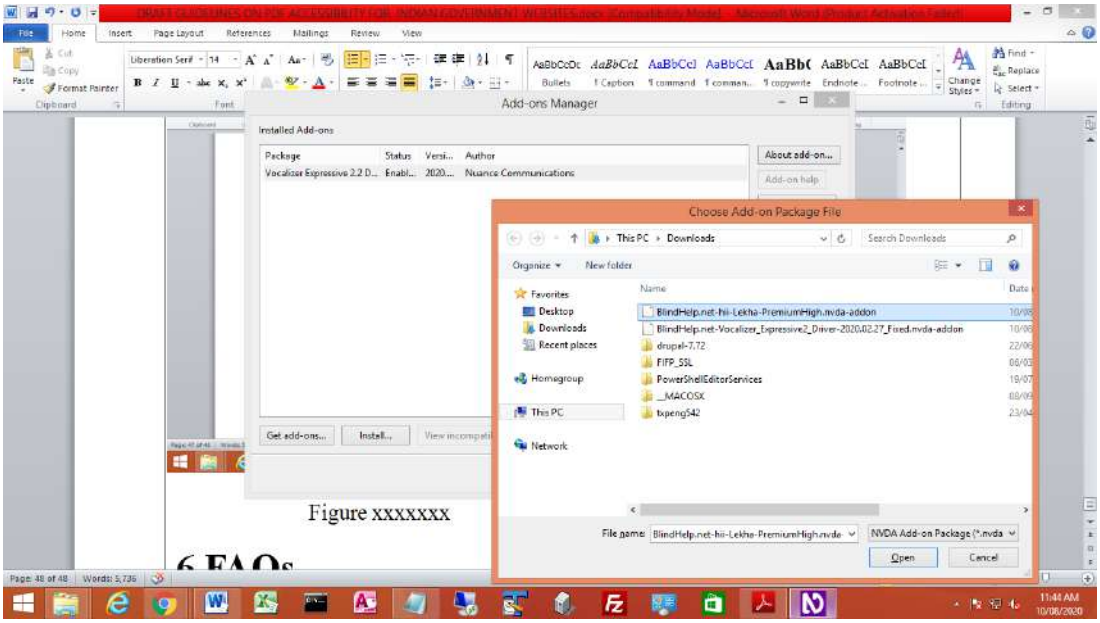
(Figure-52)

- Select Manage add-ons from the NVDA hidden icon in task bar



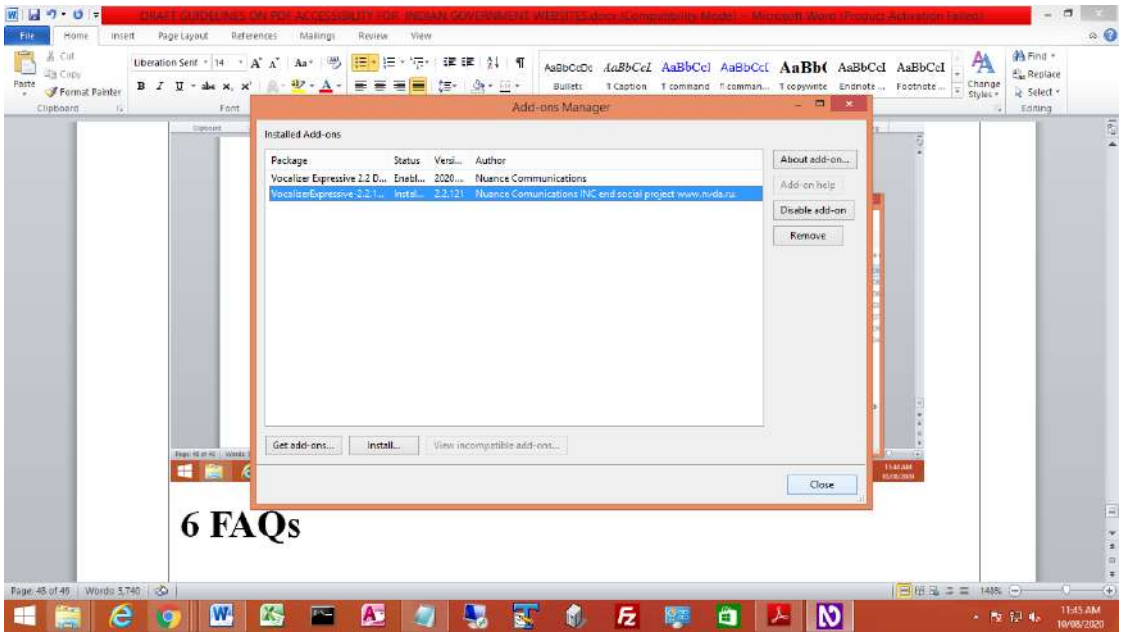
(Figure-53)

- Select Install and then Choose-Add-on Package (2 file) one by one

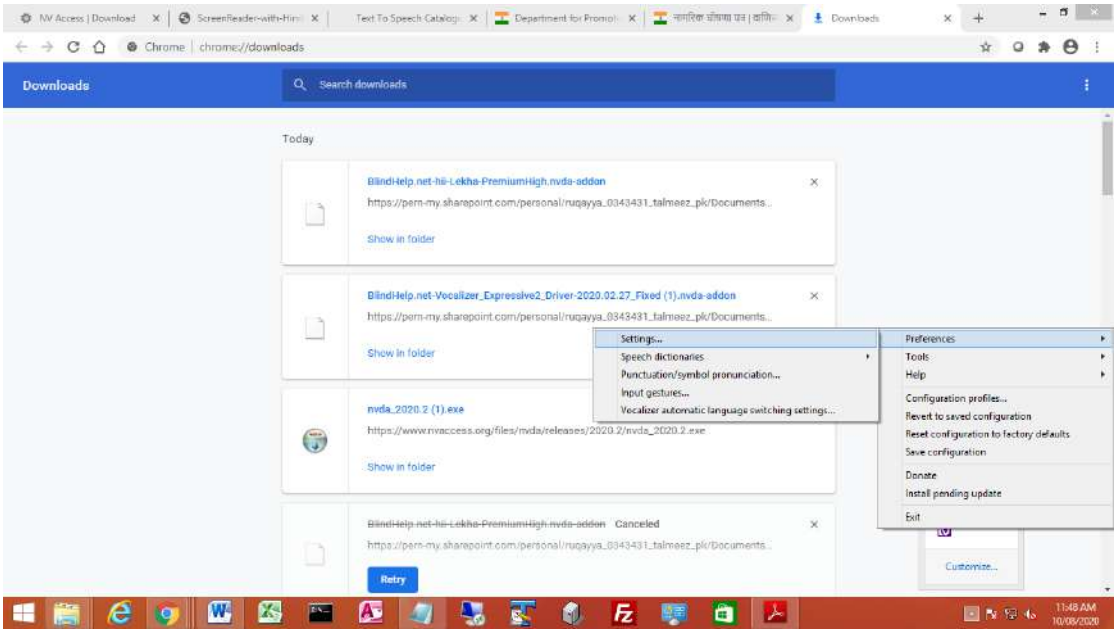


(Figure-54)

- Select Manage add-ons from the NVDA hidden icon in task bar
- Both the Add-on package files have been installed.

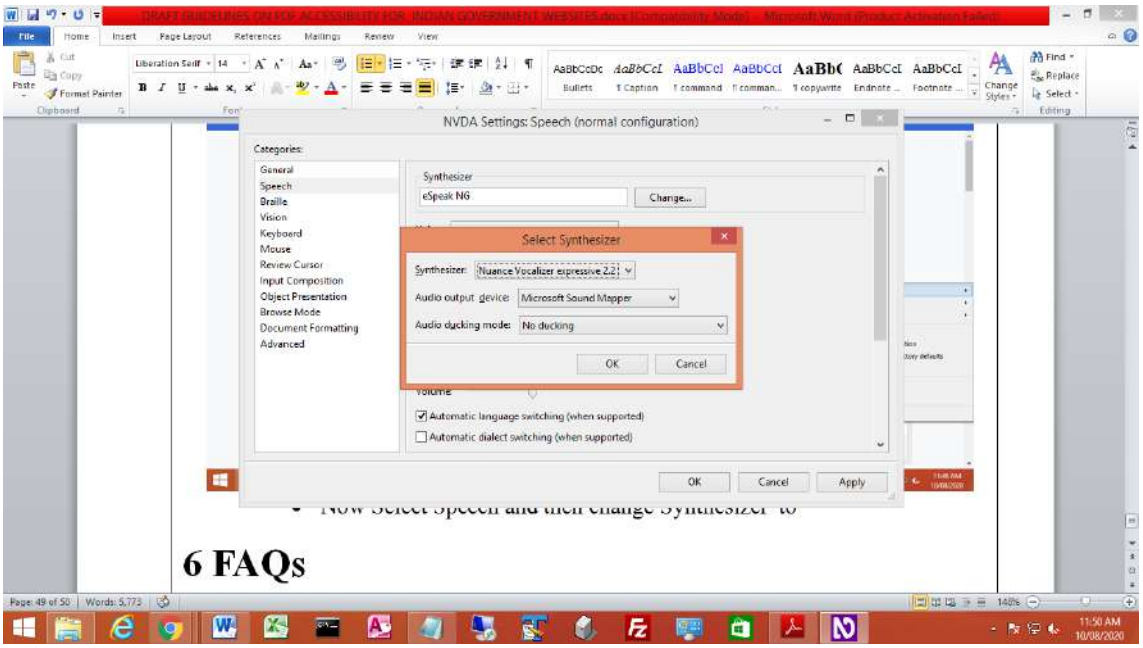


(Figure-55)

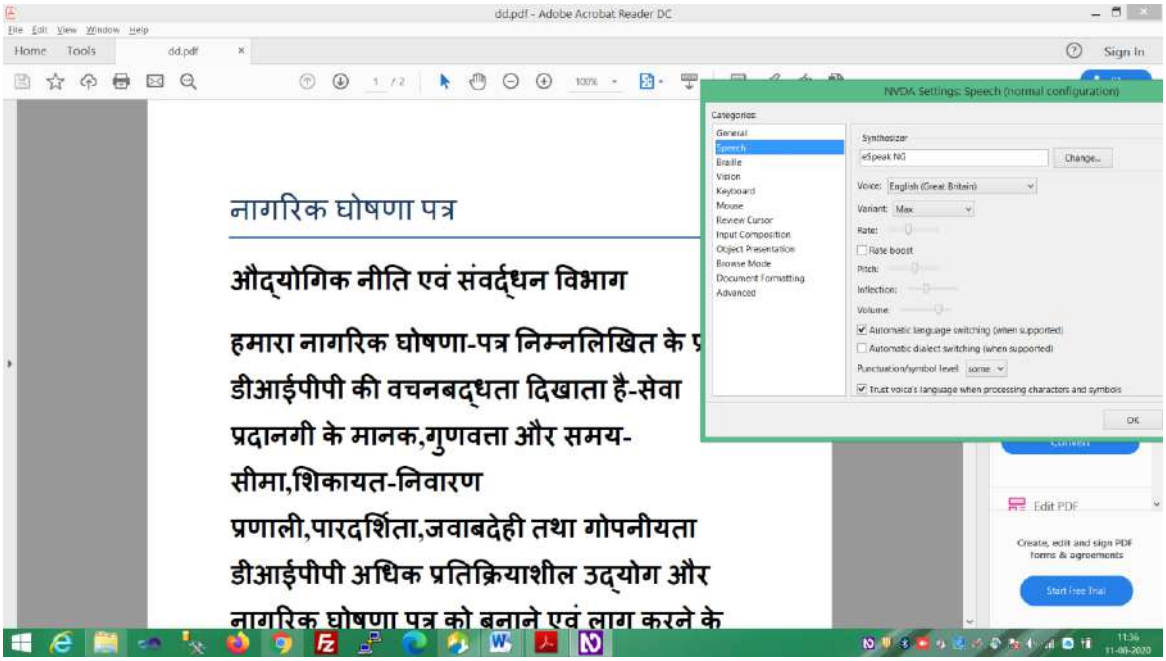


(Figure-56)

- Now Select Speech and then change Synthesizer to Nuance Vocalizer expressive 2.2 and press OK. Rate, Pitch, Inflection and Volume can be changed as desired.



(Figure-57)



(Figure-58)

6. Considerations for STQC Certification Benchmark

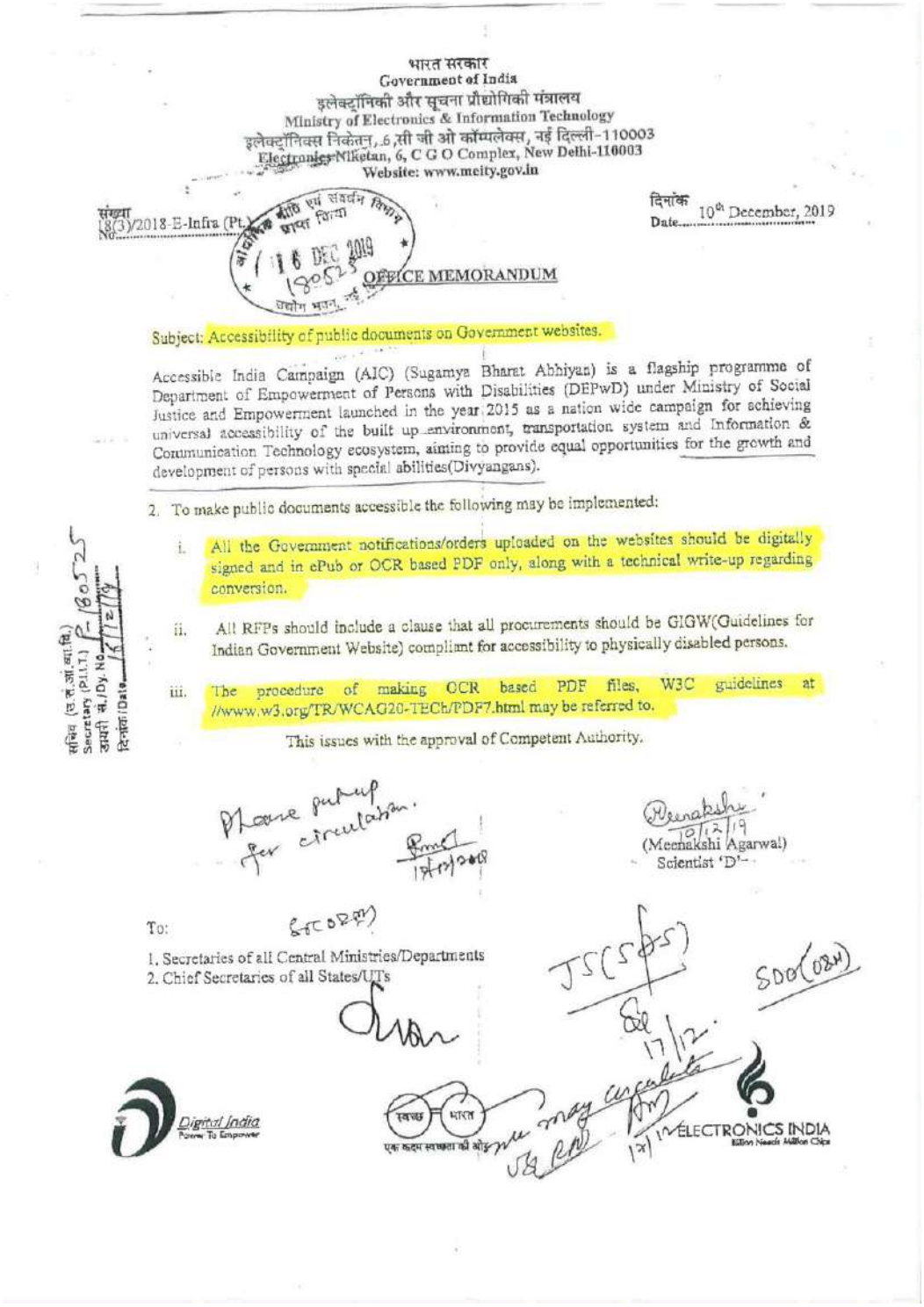
- **If the STQC certification on GIGW 2.0 requires a PDF document on website to be PDF/UA compliant then Industry Division, NIC is of the view that in order to make a document PDF/UA compliant, content managers of the website should be well versed with the operations of the software such as MS Word, Libre Office, and Acrobat Pro etc.**
- **However, if STQC compliance on PDF documents can be relaxed to an OCR tagged PDF in case of Image scanned document and Tagged PDF saved and exported from MS Word and Libre Office respectively, then the efforts of content managers could be reduced.**

7. Suggestions from CDAC

Since most of available PDF documents on Government websites are published in Hindi and other Regional languages, therefore CDAC may be approached to provide the technology guidelines on accessibility of documents in Hindi and other regional languages including bi-lingual documents while using Assistive Technologies.

8. Annexures

8.1. MeitY Office Memorandum dated 10th December, 2019



भारत सरकार
Government of India
इलेक्ट्रॉनिकी और सूचना प्रौद्योगिकी मंत्रालय
Ministry of Electronics & Information Technology
इलेक्ट्रॉनिक्स निकेतन, 6,सी जी ओ कॉम्प्लेक्स, नई दिल्ली-110003
Electronics Niketan, 6, C G O Complex, New Delhi-110003
Website: www.meity.gov.in

संख्या No. 18(3)/2018-E-Infra (Pt.)

दिनांक Date 10th December, 2019

OFFICE MEMORANDUM

Subject: Accessibility of public documents on Government websites.

Accessible India Campaign (AIC) (Sugamya Bharat Abhiyan) is a flagship programme of Department of Empowerment of Persons with Disabilities (DEPwD) under Ministry of Social Justice and Empowerment launched in the year 2015 as a nation wide campaign for achieving universal accessibility of the built up environment, transportation system and Information & Communication Technology ecosystem, aiming to provide equal opportunities for the growth and development of persons with special abilities(Divyangans).

2. To make public documents accessible the following may be implemented:

i. All the Government notifications/orders uploaded on the websites should be digitally signed and in ePub or OCR based PDF only, along with a technical write-up regarding conversion.

ii. All RFPs should include a clause that all procurements should be GIGW(Guidelines for Indian Government Website) compliant for accessibility to physically disabled persons.

iii. The procedure of making OCR based PDF files, W3C guidelines at //www.w3.org/TR/WCAG20-TECh/PDF7.html may be referred to.

This issues with the approval of Competent Authority.

Please put up for circulation.

(Meenakshi Agarwal)
Scientist 'D'

To:

1. Secretaries of all Central Ministries/Departments
2. Chief Secretaries of all States/UTs

8.2. DEPWD Office Memorandum dated 26th Feb, 2020

F.No 34-07/2019-DD-III(Pt-C)
भारत सरकार / Government of India
दिव्यांगजन सशक्तिकरण विभाग
Department of Empowerment of Persons with Disabilities (Divyangjan)
सामाजिक न्याय और अधिकारिता मंत्रालय/ Ministry of Social Justice & Empowerment
पाँचवा तल, बी विंग, पंडित दीनदयाल अंत्योदय भवन, सी जी ओ कॉम्प्लेक्स, लोधी रोड, नई दिल्ली -110003
5th Floor, B Wing, Pt. Deendayal Antyodaya Bhawan, CGO Complex, New Delhi-110003

Date: 26th Feb 2020

OFFICE MEMORANDUM

Subject: - Accessibility of Website - reg

The undersigned is directed to refer to the subject mentioned above and to say that Rule 15 of the RPwD Rules, 2017 mandates that all the websites need to be made accessible as per the guidelines adopted by M/o Administrative Reforms and Public Grievances and all the documents to be uploaded are in Electronic Publication (ePUB) or Optical Character Reader (OCR) based pdf forma.

2. All Ministries/Departments are requested to ensure that:

i. the website is accessible as per the Guidelines for Indian Government Websites adopted by D/o Administrative Reforms and Public Grievances.

ii. documents to be placed on websites shall be in accessible format i.e. ePUB or OCR based pdf forma.

(KVS Rao)
Director
Tel: 24369054
Email: kvs.rao13@nic.in

All Ministries/Departments

8.3. Observations of OTG, NIC

Annexure E: The observations on open source tools

The following tools/apis that were examined to analyze the features required towards the accessibility requirements namely creation of PDF with accessibility including Digital Signature , Auditing of PDF documents for conformance and using OCR for converting existing scanned documents to accessible PDF documents.

S.No	Name & URL	Purpose	License	Remarks
1	Libre Office https:// www.libreoffice.org/	Create Accessible PDF Documents	MPLv2.0 (secondary license GPL, LGPLv3+ or Apache License 2.0)	Sample Documents were tried. Could create tagged and signed PDF files. However certain compliance requirements could not be met as per the PDF Accessibility Checker 3. Please see the sample Document and The compliance Report.
2	PDF Accessibility Checker 3. https://www.access-for-all.ch/en/pdf-lab/pdf-accessibility-checker-pac.html	Auditing PDF Documents for Accessibility	Shareware	Not a Open Source Tool
3	Pypdf and pytesseract https://pypi.org/project/PyPDF2/1.26.0/ https://pypi.org/project/pytesseract/	API that can used to process scanned PDF to text using OCR	BSD and Apache License	APIs that can used to convert scanned PDF to text
4	OCR engine - libtesseract and a command line program – tesseract https://github.com/tesseract-ocr/tesseract	Convert scanned PDF documents to text documents	Apache license 2.0	A command line tool to convert images to texts and use standard text processing tools to convert them to PDF with tagging.
5	OCRFeeder https://wiki.gnome.org/action/show/Apps/OCRFeeder	Import PDF and Images to convert to text using OCR (Tesseract based)	GNU GPL V3.0	The extracted text can be exported in txt, pdf and odt formats.
6	FreeOCR https://github.com/A9T9/Free-Ocr-Windows-Desktop/releases	Import PDF and Images to convert to text using OCR (Tesseract based)	GNU AGPL V3	The OCR conversion from multi-page image has better conversion efficiency than the direct PDF conversion. However, the accessibility compliance need manual efforts in adding the required tags and format setting
7	VietOCR.NET http:// vietocr.sourceforge.net/	Covert from PDF, TIFF, JPEG, GIF, PNG, BMP image formats to text using OCR (Tesseract based)	Apache 2.0	Bulk & batch operations Java & .NET GUI frontends for Tesseract OCR engine

References
https://www.w3.org/WAI/ER/tools/
https://wiki.documentfoundation.org/Accessibility

END OF DOCUMENT