**Design Report: Word To Powerpoint converter**

**Submitted to Vishwakarma University, Pune**

**Made by:-Ayush katekhaye**

**Roll no.- 22**

**Srn: 202101420**

**Div: D**

**Third Year Engineering**

**Department of Computer Engineering**

**Faculty of Science and Technology**

**Academic Year**

**2023-2024**

**Overview:**

The Word to PowerPoint Converter is an application designed to facilitate the seamless conversion of Word documents (.docx) into corresponding PowerPoint presentations (.pptx). Developed using Python and the Tkinter library for the graphical user interface, this tool offers users a straightforward method to transform textual content into a slideshow format.

**Objectives:**

The primary objective of the project was to create a user-friendly interface that allows users to:

Select a Word document for conversion.

Specify the destination for the converted PowerPoint file.

Automate the conversion process while maintaining the original content's formatting and structure.

**Features Implemented:**

File Selection and Conversion

Browse Functionality: Enables users to browse and select a Word document for conversion.

Conversion Process: Transforms the content of the Word document into a PowerPoint presentation.

**Presentation Structure:**

Slide Layouts: Utilizes predefined slide layouts for consistency in the generated PowerPoint slides.

Content Formatting: Structures text content into slides with a maximum of 15 lines per slide, wrapped within text boxes.

**User Interface:**

Simple Interface: Offers a clear and easy-to-navigate interface using Tkinter widgets.

Information Prompts: Utilizes message boxes to provide feedback on the success or absence of content during conversion.

**Code Implementation:**

The application is developed in Python and leverages several libraries, including:

Tkinter: Used for creating the graphical user interface.

python-docx: Facilitates the extraction of content from Word documents.

python-pptx: Enables the creation and manipulation of PowerPoint presentations.

The conversion process involves iterating through the paragraphs of the selected Word document, organizing the content into slides, and populating them within a predefined slide layout with a background image.

**Final code:**

import tkinter as tk

from tkinter import filedialog

from tkinter import messagebox

from docx import Document

from pptx import Presentation

from pptx.util import Inches

from textwrap import fill

def open\_word\_file():

    file\_path = filedialog.askopenfilename(filetypes=[("Word files", "\*.docx")])

    if file\_path:

        ppt\_file\_path = filedialog.asksaveasfilename(defaultextension=".pptx")

        if ppt\_file\_path:

            word\_to\_ppt(file\_path, ppt\_file\_path)

def word\_to\_ppt(word\_file\_path, ppt\_file\_path):

    ppt = Presentation()

    doc = Document(word\_file\_path)

    slide\_layout = ppt.slide\_layouts[5]

    background\_image\_path = "7243-01-low-poly-background-16x9-1.jpg"

    current\_slide = None

    text\_frame = None

    line\_count = 0

    lines\_per\_slide = 15

    for paragraph in doc.paragraphs:

        if current\_slide is None or line\_count >= lines\_per\_slide:

            current\_slide = ppt.slides.add\_slide(slide\_layout)

            current\_slide.shapes.add\_picture(background\_image\_path, Inches(0), Inches(0), Inches(10), Inches(7.5))

            text\_frame = current\_slide.shapes.add\_textbox(Inches(1), Inches(1), Inches(8), Inches(4)).text\_frame

            line\_count = 0

        p = text\_frame.add\_paragraph()

        text = fill(paragraph.text, width=40)

        p.text = text

        line\_count += len(text.split('\n'))

    if ppt.slides:

        ppt.save(ppt\_file\_path)

        messagebox.showinfo("Conversion Complete", "Word to PowerPoint conversion successful!")

    else:

        messagebox.showwarning("No Content", "No content to convert!")

window = tk.Tk()

window.title("Word to PowerPoint Converter")

window.geometry("800x500")

window.resizable(False, False)

label = tk.Label(window, text="Select a Word document to convert to PowerPoint:")

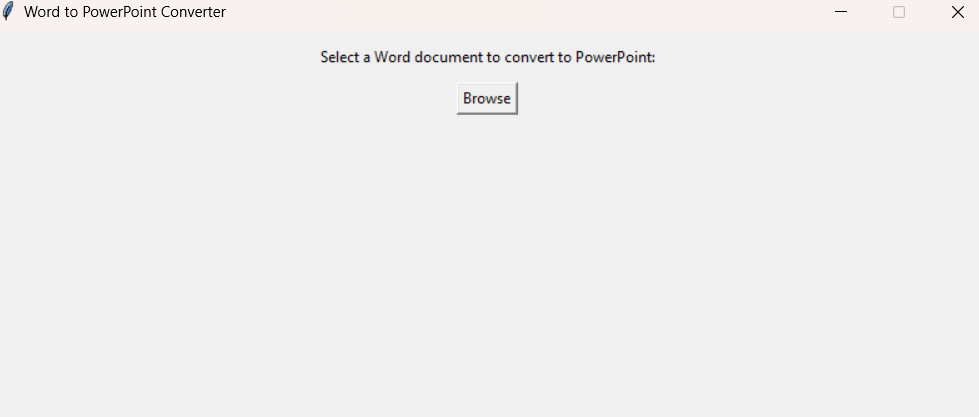
label.pack(pady=10)

browse\_button = tk.Button(window, text="Browse", command=open\_word\_file)

browse\_button.pack()

window.mainloop()

**Output:**



**Future Enhancements:**

The project has potential for further improvements and additions, such as:

Customization Options: Introducing options for customizing slide layouts, background images, and text formatting within the slides.

Error Handling: Implementing comprehensive error handling mechanisms to address potential issues during conversion.

Performance Optimization: Optimizing code performance for faster conversion, especially with larger documents.

**Conclusion:**

The Word to PowerPoint Converter provides a basic yet functional solution for converting textual content from Word documents into PowerPoint presentations. Its simplicity and ease of use make it a valuable tool for users seeking a quick and efficient way to transform documents.

**Reference:**

Tkinter Documentation: https://docs.python.org/3/library/tkinter.html

Googletrans Documentation: https://pypi.org/project/googletrans