**📄 Final Report Template**

**1. Title Page**

* **Title**: Wordcount Tools in Java
* **Submitted by**: [Purushottam Phuyal]
* **Course**: [Mini-Project]
* **Date**: [March 28, 2025]

**2. Abstract**  
The **Wordcount Tools in Java** is a tool designed to count the number of lines, words, and characters in .txt, .docx, and .pdf files. The application is built using Java, Apache POI, and PDFBox libraries to handle the reading of different file formats. It features a simple **Swing-based GUI** that allows users to select a file, process it, and display the word count statistics in a popup window. The results are also saved in a text file, making it easier for users to access the data later.

**3. Background**  
Word counting is a fundamental task in data analysis, content creation, and academic writing. Various tools are available to count words, lines, and characters, but they are often limited to specific formats like .txt. This project aims to develop a **versatile word-counting tool** that supports multiple file formats, ensuring that users can count text content across different document types, including .txt, .docx, and .pdf.

**4. Problem Statement**  
While many word-counting tools exist, they often do not support all common file formats (e.g., .pdf or .docx). This limits the tool's usability, especially for users working with documents in various formats. Therefore, the need arises for a simple yet effective tool that can handle multiple file types and provide real-time feedback.

**5. Research and Methodology**  
The project was developed using **Java** with the following technologies:

* **Apache POI** for reading and processing .docx files.
* **PDFBox** for reading and processing .pdf files.
* **Swing** for building a user-friendly GUI.

The approach involved:

1. Selecting the appropriate libraries for different file types.
2. Writing functions to count lines, words, and characters.
3. Implementing the GUI using **JFileChooser** to allow users to select files.
4. Using **JOptionPane** to display results in a popup.

The tool was tested on several documents with varying content, and the output was verified to be accurate.

**6. Objectives**  
The objectives of the **Wordcount Tools in Java** project are as follows:

1. To build a word-counting tool that supports .txt, .docx, and .pdf files.
2. To create an intuitive GUI for easy user interaction.
3. To provide output both in a popup and a saved text file.
4. To ensure the tool is easy to use by non-technical users.

**7. Results**  
The **Wordcount Tools in Java** successfully met the objectives:

* Users can select a file from their system and get word, line, and character counts.
* The results are displayed in a **popup dialog** and saved in the same folder as the selected file.
* The tool supports multiple file types, including .txt, .docx, and .pdf.

**8. Conclusion**  
The Wordcount Tool provides a simple, efficient, and user-friendly way to count the words, lines, and characters of text in various formats. The project demonstrates the ability to work with file I/O in Java and showcases a **clean graphical user interface**. The tool can be further expanded to support additional formats or advanced features like **text analysis** or **statistics tracking**.

**9. Future Work**

* Support additional formats such as .rtf, .odt.
* Implement text analysis features (e.g., frequency analysis, sentiment analysis).
* Add the ability to **export results** in different formats (e.g., CSV, JSON).

**10. References**

* Apache POI: <https://poi.apache.org/>
* PDFBox: <https://pdfbox.apache.org/>
* Java Swing Documentation: <https://docs.oracle.com/javase/tutorial/uiswing/>

**📖 README File Template**

**Wordcount Tools in Java**

**Overview**

Wordcount Tools in Java is a desktop application that allows users to count the number of lines, words, and characters in .txt, .docx, and .pdf files. The application is built using **Java**, **Apache POI** for reading .docx files, and **PDFBox** for reading .pdf files. The tool features an easy-to-use **Swing-based GUI** that allows users to select a file and get word count statistics.

**Features**

* **Support for multiple file types**: .txt, .docx, and .pdf.
* **GUI interface**: Built using Java Swing to allow easy file selection.
* **Results display**: Shows word, line, and character counts in a popup window.
* **Output file**: Saves the results in an output.txt file in the same folder as the input file.

**Requirements**

* **Java 17 or later**
* **Libraries**:
  + Apache POI for .docx support (version 5.2.3)
  + PDFBox for .pdf support (version 2.0.27)

**How to Run the Application**

1. **Download the .jar file**: [WordcountTools.jar](https://chatgpt.com/g/g-p-67dda7e26870819183a5668c02b9c429-wordcount-tools-in-java/c/path-to-your-jar)
2. **Ensure Java is installed**: [Download Java](https://www.oracle.com/java/technologies/javase-jdk17-downloads.html)
3. **Run the .jar file**:
   * Double-click the .jar file.
   * The file chooser will appear; select a .txt, .docx, or .pdf file.
   * The word count, line count, and character count will be displayed in a popup, and the results will be saved in an output.txt file in the same folder as the selected file.

**How it Works**

1. Select a file using the **JFileChooser**.
2. The tool reads the file content.
3. It counts the **number of lines**, **words**, and **characters**.
4. A popup window displays the result.
5. The results are saved in output.txt.

**Contact**

If you have any questions or feedback, please contact:  
[Your Name]  
[Your Email Address]

**✅ What to Submit:**

1. **WordcountTools.jar** file.
2. **Final report** (structured as described above).
3. **README** file.
4. **Screenshots** of the working app (including file selection and popup).
5. **Source code** (if required).

Let me know if you'd like me to assist with the **final check**, **packing files**, or **submission details**!