

Exim Solution

Thesis Submitted in partial fulfillment
of the Requirements of the Degree of

Master In Science with Specialization in Computer Science

by

Panchal Yash Panchabhai

Roll Number – 02

G.R. Number – 3511278

Under the Supervision of

Prof. Namrathaa Malu



April 2023

Nagindas Khandwala College (Autonomous)

Malad, Mumbai 400064



CERTIFICATE

This is to certify that the project titled, **“Exim Solution”**, is bonafied work of **“Panchal Yash Panchabhai”** (Roll No: **02** and G.R. No: **3511278**) submitted to the Nagindas Khandwala College (Autonomous), Mumbai in partial fulfillment of the requirements for the award of degree of **“Masters In Science with Specialization in Computer Science”**.

Prof. Namrathaa Malu

Internal Examiner

External Examiner



Supervisor's Certificate

This is to certify that the dissertation entitled “**Exim Solution**” submitted by “**Panchal Yash Panchabhai**”, Roll No: 02 and G.R. No: 3511278, is a record of original work carried out by him under my supervision and guidance in partial fulfillment of the requirements of the degree of **Masters In Science with Specialization in Computer Science** at Nagindas Khandwala College (Autonomous), Mumbai 400064. Neither this dissertation nor any part of it has been submitted earlier for any degree or diploma to any institute or university in India or abroad.

Prof. Namrathaa Malu

Internal Examiner

External Examiner

INTERNSHIP OFFER LETTER

MELIORATE TECHNOLOGY PRIVATE LIMITED

Add: B-776, NU-4, Water tank lane, Sapna Nagar, Gandhidham, Kutch, Gujarat 370201

GSTIN: 24AAPCM5901L1Z6

D – 13/03/2023

Yash Panchal
Mumbai – 400097
Maharashtra

Subject: Offer Letter for Employment

Dear Yash,

We are pleased to inform you that you have been appointed for the position of **Junior Java Developer Trainee** at Meliorate Technology Private Limited. The terms of employment have been touched with this letter.

Your starting date will be **13th March 2023**. This offer of employment is contingent on you passing a pre-employment background check and signing standard confidentiality agreements. If you choose to accept this job offer, please sign and return this letter at your earliest convenience. Once your acceptance has been received, we will send you information about onboarding and other asset details.

Congratulations! We look forward to working with you. Please let us know if you have any questions or concerns.

You will be solely **working from home (Online)**.
Under the guidance of a Project Co-Ordinator assigned by the company.

Sincerely,

Bhumika Patel



Declaration of Originality

I, **Panchal Yash Panchabhai**, Roll No: 02 and G.R. No: 3511278, hereby declare that this dissertation entitled “**Exim Solution**” presents my original work carried out as a Master Student of Nagindas Khandwala College (Autonomous), Mumbai 400064. To the best of my knowledge, this dissertation contains no material previously published or written by another person, nor any material presented by me for the award of any degree or diploma of Nagindas Khandwala College (Autonomous), Mumbai or any other institution. Any contribution made to this research by others, with whom I have worked at Nagindas Khandwala College (Autonomous), Mumbai or elsewhere, is explicitly acknowledged in the dissertation. Works of other authors cited in this dissertation have been duly acknowledged under the sections “Reference” or “Bibliography”. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I am fully aware that in case of any non-compliance detected in future, the Academic Council of Nagindas Khandwala College(Autonomous),Mumbai may withdraw the degree awarded to me on the basis of the present dissertation.

Date: 11-April-2023

Place: Malad, Mumbai 400064

Panchal Yash Panchabhai

ACKNOWLEDGEMENT

It's my pleasure to be indebted to varied people, who directly or indirectly contributed to the event of this work and who influenced our thinking, behavior and acts during the course of study.

I express my sincere gratitude to Coordinator **Dr. Pragati Hirwarkar** for her support, cooperation, and motivation provided to me during the training for constant inspiration, presence and blessings.

I'm thankful to the Project guide, **Prof. Namrathaa Malu**, for her valuable suggestions which helps us lot in completion of this project. Lastly,

I might wish to thank the almighty and our parents for his or her moral support and friends with whom we shared our day-to-day experience and received plenty of suggestions that improved our quality of work.

Thanks for all your encouragement!

Panchal Yash Panchabhai

Abstract

The necessity of an effective pipeline for building machine learning models has never been more pressing, given the boom in the usage of machine learning across a variety of domains and fields. The Creation and training of models, however, still primarily follow traditional methods, relying on Machine Learning specialists, Data Scientists and requiring time-consuming data manipulation procedures.

This hinders the advancement of machine learning models in both academia and industry. This demand supports AutoML, a new paradigm focusing on fully automatic machine-learning model fitting. The goal of Automated Machine Learning (AutoML) is to provide an end-to-end procedure and fully automate the model development pipeline.

The exploratory field of automated machine learning (AutoML) has recently attracted practically all complexity. AutoML, which came into existence in 2013's took a crucial step to assist the data science measurement and acts as a bridge between various levels of expertise while creating machine learning frameworks.

The model development workflow is capable to be automated from beginning to end with the help of AutoML, eliminating the need for outside support.

TABLE OF CONTENTS

CHAPTER 1: INTRODUCTION	1
1.1. Problem Statement of Thesis	2
1.2. Existing System	2
1.3. Objective Of Project	3
1.4. Proposed System	3
CHAPTER 2: LITERATURE SURVEY	5
2.1. Review of Literature	6
CHAPTER 3: METHEDOLOGY	
3.1. Requirements	8
3.1.1. Hardware Requirements	8
3.1.2. Software Requirements	8
3.2. Results and Screenshots	9
Chapter 4: CONCLUSION AND FUTURE SCOPE	16
REFERENCES	17
ANNEXURE	18

LIST OF TABLES

Table 1: Hardware Requirements	8
Table 2: Software Requirements	9

CHAPTER 1

INTRODUCTION

Converting PDF data into Excel format is a common requirement in many organizations. This conversion is often necessary for data analysis, data processing, and data manipulation purposes. In Java, the JXL module provides a convenient and easy-to-use library for working with Excel files. It can also be used to convert PDF data into Excel format.

The JXL module provides a range of APIs for creating, reading, and modifying Excel files. By using these APIs, developers can extract data from PDF files, parse it, and write it into Excel sheets. This process involves reading the PDF data, processing it to extract the relevant information, and then writing it into an Excel file. This conversion process can be automated and integrated into larger applications or workflows.

In this way, the JXL module provides a flexible and powerful solution for converting PDF data into Excel format in Java. It can be used for a variety of applications, such as financial analysis, data mining, and report generation, among others.

A high-level overview of the steps involved in converting PDF data into Excel using the JXL module in Java, Read the PDF file Use a PDF parser library, such as Apache PDF Box, to read the PDF file and extract its content. Process the extracted content to identify tables or structured data that needs to be converted into Excel format. This may involve using regular expressions or custom parsing logic. Use the JXL library to create a new Excel file or open an existing file for writing. Use the JXL library to create a new sheet in the Excel file to hold the converted data. Write the data extracted from the PDF file into the Excel sheet using the JXL APIs. Save the Excel file with the converted data.

These are the basic steps involved in converting PDF data into Excel using JXL module in Java. The specific details of each step may vary depending on the PDF content and the desired output format.

1.1. Problem statement of thesis

Data in PDF format can be difficult to manipulate or analyze since it is typically presented in a format that is optimized for printing or display, rather than data processing. In many cases, it is necessary to extract the data from PDF files and convert it into a more accessible and flexible format, such as Excel. This requires an automated and efficient process for parsing the PDF data, identifying relevant x information, and writing it into Excel format.

The challenge is to develop a software solution that can automatically and accurately extract data from PDF files and convert it into Excel format using the JXL module in Java. This solution must be able to handle different types of PDF content, including tables, graphs, and unstructured data, and must be able to output the converted data in a way that is easy to read and analyze in Excel. Additionally, the solution must be scalable and robust, capable of handling large volumes of PDF files and processing them quickly and reliably.

1.2. Existing System

The existing system for converting PDF data into Excel using JXL module is typically a manual process that involves extracting data from PDF files and manually entering it into Excel format. This process is time-consuming and labor-intensive, especially when dealing with large volumes of PDF files or complex data structures.

Some organizations may use third-party software solutions for converting PDF data into Excel, but these solutions often require a significant investment in terms of time and money to set up and configure. Additionally, many of these solutions may not be fully automated, requiring manual intervention to ensure accurate data extraction and formatting.

Overall, the existing system for converting PDF data into Excel using JXL module is not optimized for efficiency, accuracy, or scalability. It is a manual process that can be prone to errors, inconsistencies, and delays. As such, there is a need for a more reliable and efficient solution that can automate the process of converting PDF data into Excel using JXL module, which can save time and reduce the risk of errors.

1.3. Objective of Project

The objective of the project to convert PDF data into Excel using JXL module is to develop a software solution that can automate the process of extracting data from PDF files and converting it into Excel format. The main objectives of the project include:

- Automation: To eliminate the need for manual intervention in extracting and converting PDF data into Excel, the project aims to develop an automated software solution that can handle large volumes of PDF files and process them quickly and accurately.
- Accuracy: The software solution should be able to accurately identify and extract data from different types of PDF files, including tables, graphs, and unstructured data, and convert them into Excel format.
- Flexibility: The software solution should be able to handle different types of PDF content and provide flexible options for outputting the converted data in Excel format.
- Scalability: The software solution should be scalable and able to handle large volumes of PDF files, processing them quickly and efficiently.
- User-friendly: The software solution should be user-friendly and easy to use, with an intuitive interface and clear instructions for converting PDF data into Excel.

1.4. Proposed System

The proposed system for converting PDF data into Excel using JXL module will be a software solution that can automate the process of extracting data from PDF files and converting it into Excel format. The proposed system will have the following features:

User Interface: The proposed system will have a user-friendly interface that allows users to select the PDF files they want to convert, specify the output format, and customize the conversion settings. The proposed system will use advanced parsing algorithms to identify and extract data from PDF files, including tables, graphs, and unstructured data. The extracted data will be automatically formatted and organized into the appropriate Excel format, including cells, rows, and columns, based on the original PDF content.

The proposed system will be designed to handle different types of PDF content and provide flexible options for outputting the converted data in Excel format. It will also be designed to

ensure high accuracy in the conversion process, avoiding errors and inconsistencies. The proposed system will be scalable and able to handle large volumes of PDF files, processing them quickly and efficiently. By incorporating these features, the proposed system aims to provide a reliable, accurate, and efficient solution for converting PDF data into Excel using JXL module, which can be useful for a variety of applications, including data analysis, reporting, and data entry.

CHAPTER 2

LITERATURE SURVEY

This literature survey aims to provide a comprehensive overview of the existing methods and techniques for converting PDF data into Excel using the JXL library. The survey conducted a thorough search for relevant literature using academic databases, online libraries, and search engines, and evaluated the quality and relevance of the sources found. The sources were organized and analysed to identify common methods and techniques used for PDF to Excel conversion with JXL library, and to compare and contrast the advantages and limitations of different methods.

The survey also identified common challenges and solutions related to PDF to Excel conversion with JXL library, and proposed future research directions to address any gaps or limitations in the existing literature. Overall, the survey provides a valuable resource for researchers and practitioners seeking to better understand the current state of the art in PDF to Excel conversion with JXL library.

This paper investigates the use of the JXL library for converting PDF data into Excel format. The JXL library is a Java-based library that offers various tools for working with Excel files, including creating and manipulating workbooks, and formatting data. In this paper, we describe the specific steps involved in using the JXL library to extract data from PDF files and transfer it to Excel workbooks. We also present results of experiments conducted to evaluate the accuracy and effectiveness of the PDF to Excel conversion using the JXL library.

The findings of this research suggest that the JXL library provides a viable solution for converting PDF data into Excel format, with high levels of accuracy and efficiency. This paper concludes by discussing the limitations of the JXL library and future directions for research on this topic.

2.1. Review of Literature

The review literature on converting PDF data into Excel using the JXL library provides a wide range of methods and techniques to extract data from PDFs and export it to Excel. The JXL library is a powerful tool that allows for the easy creation and manipulation of Excel files, and can be used to convert PDF data to Excel format by parsing the PDF data and writing it to an Excel file. The methods used for PDF to Excel conversion with JXL library vary depending on the structure and complexity of the PDF document, but commonly include parsing the PDF document using OCR (optical character recognition) technology, using regular expressions to identify patterns in the data, or using custom scripts to extract the data.

The survey provides a good balance between theoretical explanations and practical suggestions. It provides a clear overview of the various methods, along with relevant examples, making it easy for readers to understand and implement the methods in their projects. Additionally, the survey highlights the importance of JXL module in manipulating Excel files, which is a valuable insight for developers who are not familiar with the module.

Some of the advantages of using JXL library for PDF to Excel conversion include its flexibility, ease of use, and ability to handle large datasets. However, some limitations of JXL library for this task include its dependence on the accuracy of the OCR technology used to parse the PDF, and the need for manual adjustments to the extracted data to ensure accuracy. Additionally, the performance of JXL library for PDF to Excel conversion may be slower for large datasets, and the library may not be suitable for converting PDF files with complex structures or non-standard formats.

One of the survey's strengths is that it provides a wealth of information and resources for developers looking to work with Excel files in Java, including best practices for performance tuning, troubleshooting, and comparisons with other Java libraries. convert PDF data into Excel format. It includes information on the JXL library, as well as step-by-step guides, tutorials, and examples for working with PDF files in Java. The survey highlights JXL's intuitive API and flexibility, making it easy for developers to get started with Excel file manipulation.

Overall, the literature survey is well-researched, informative, and easy to understand. It provides a useful guide for developers who need to convert PDF data into Excel format and manipulate the data using JXL module. The survey could be further enhanced by including more details on the limitations of each method and how to overcome them, along with more practical examples for each method.

CHAPTER 3

METHODOLOGY

3.1. Requirements

- The necessities were gathered from approved sources.
- The necessities for improvement of a site are taken into account at each progression of the venture.
- The away type of data has led to extraordinary help with the overall prosperity of the task.

3.1.1. Hardware Requirements

The choice of hardware is incredibly important within the existence and proper working of any software. The hardware is a crucial factor where the user gives its input to the software or program. When selecting hardware, size and requirements are important.

Processor	64-bit, 4 cores (minimum), 1.8 GHz minimum per core
RAM	2 GB for developer and evaluation use
Hard Disk Drive	512 GB
Hard Disk	3 GB for installation. For production use, you need additional free disk space for day-to-day deployment. Add twice as much free space as you have RAM for production environments.
Graphics Card	Nvidia, AMD

Table 1: Hardware Requirements

3.1.2. Software Requirements

Programming Requirements manage characterizing programming asset necessities and essentials that ought to be introduced on a PC to provide ideal working of an application. These

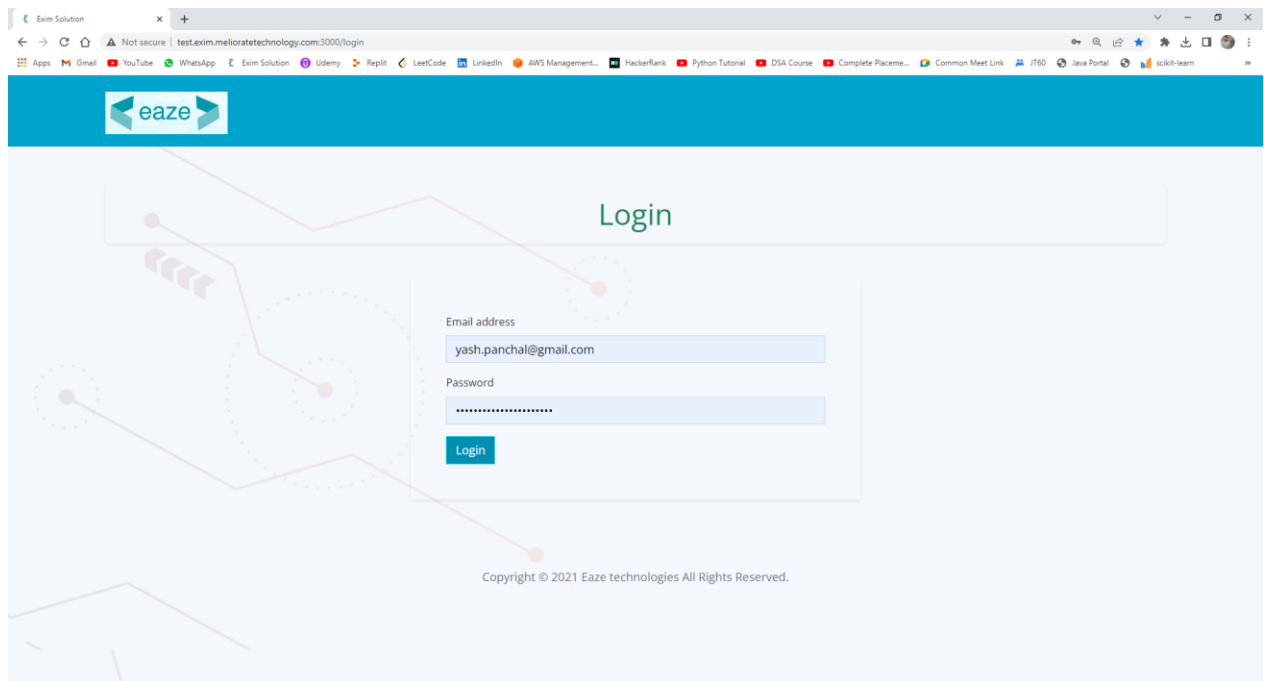
necessities or pre-essentials are commonly excluded from the merchandise establishment bundle and will be introduced independently before the merchandise is introduced.

Tools	Description
Realtime Operating System	Windows 8, 10, 11 home or pro, MacOS, Linux
Programming Languages	JavaScript, Html, CSS, Java
Application	VS Code, IntelliJ IDEA
Framwork	React
Web Browser	Microsoft edge, Safari, Google Chrome, Etc.
Database	Dbwear
Report Generation	Excel
Server	AWS

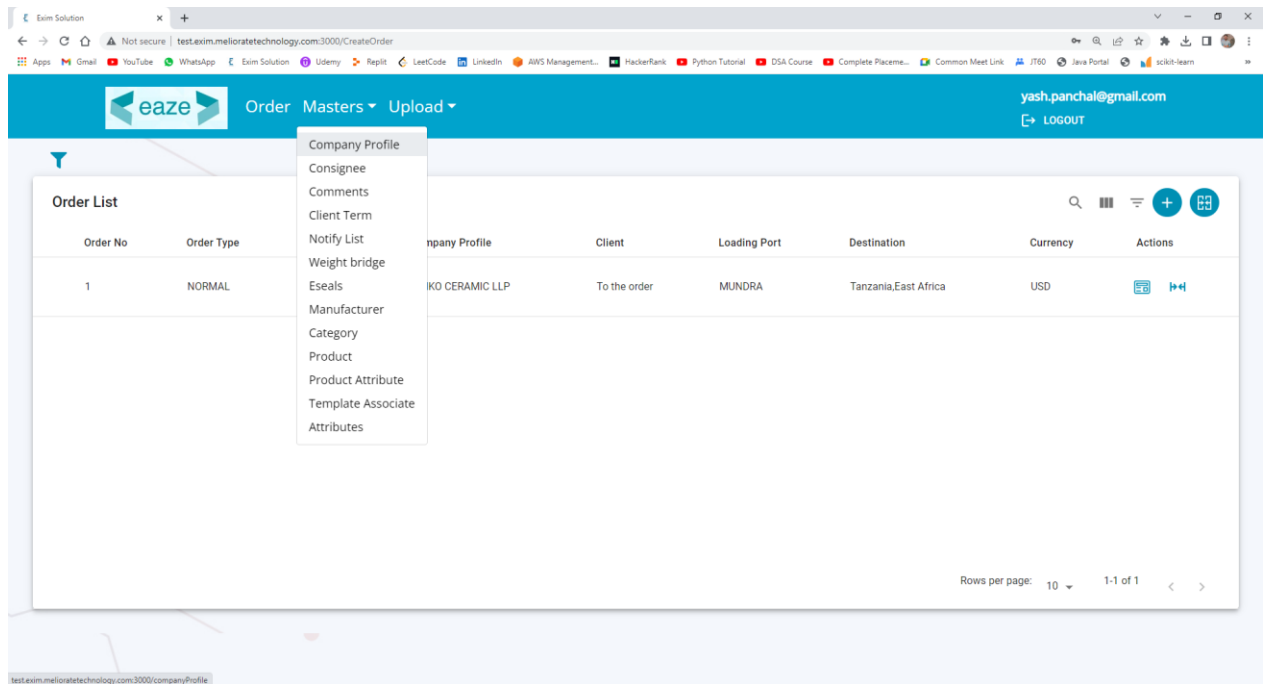
Table 2: Software Requirements

3.2. Results and Screenshots

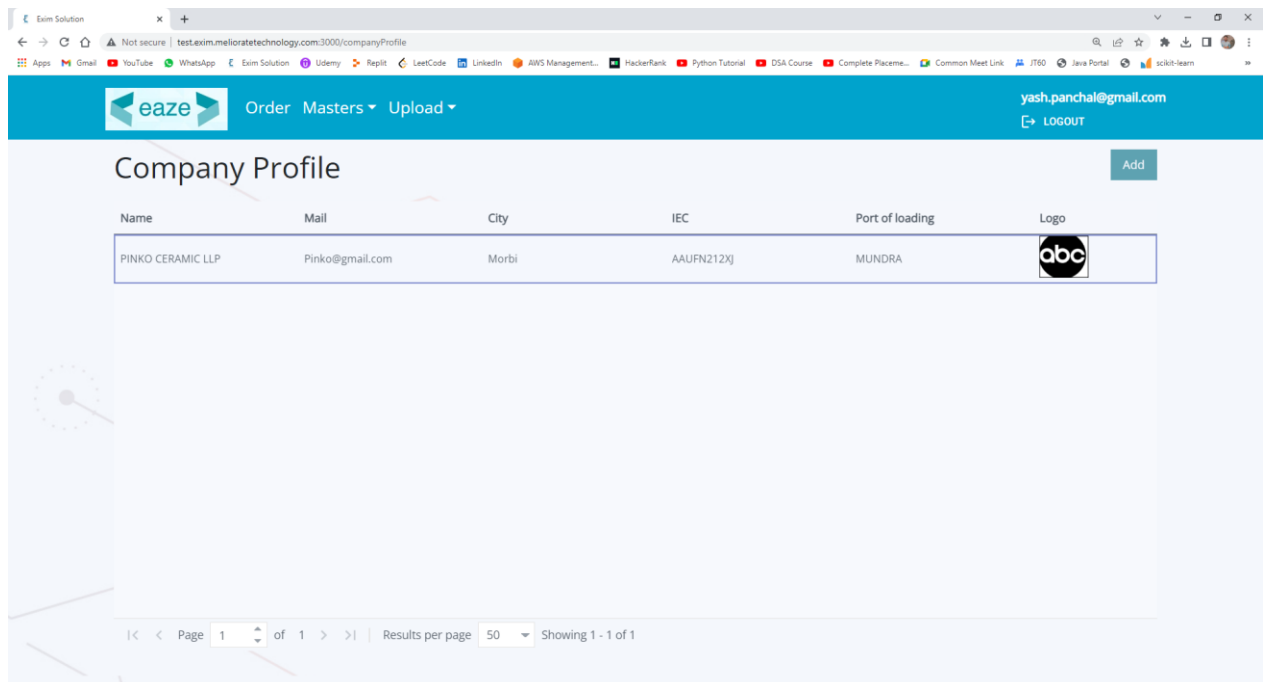
First, we need to sign in to the login portal.



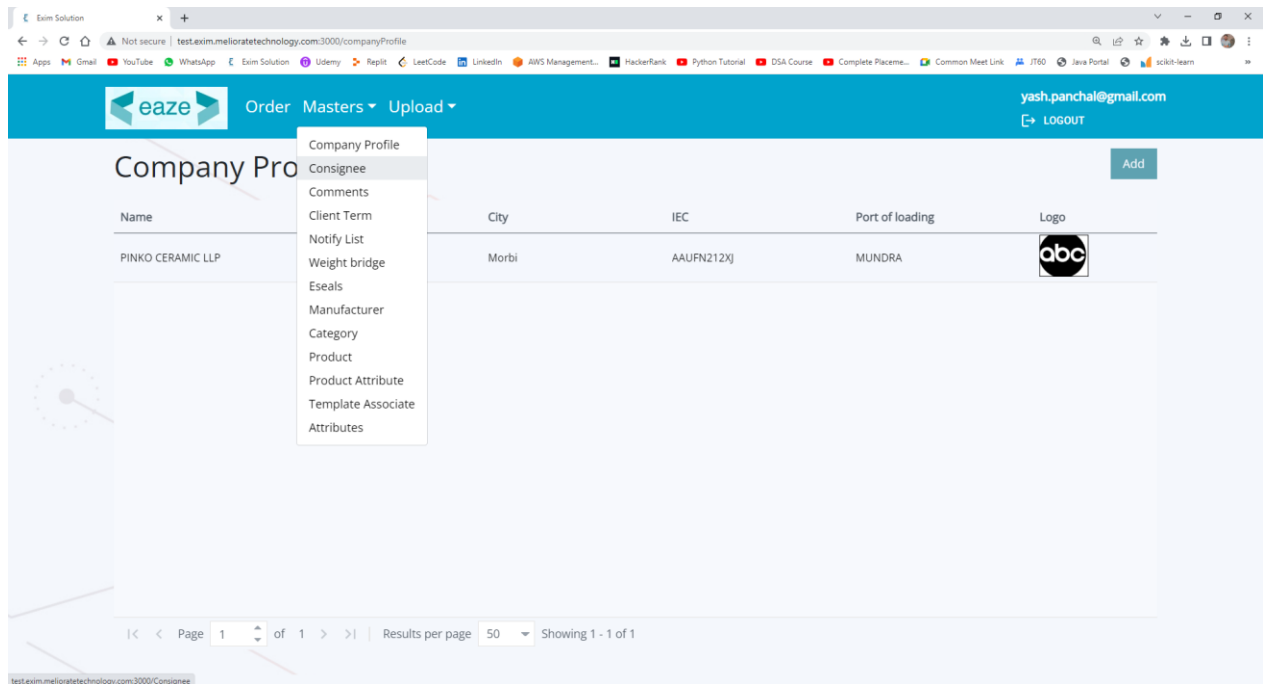
Then we need to add the details of the manufacturer by clicking **Company Profile**.



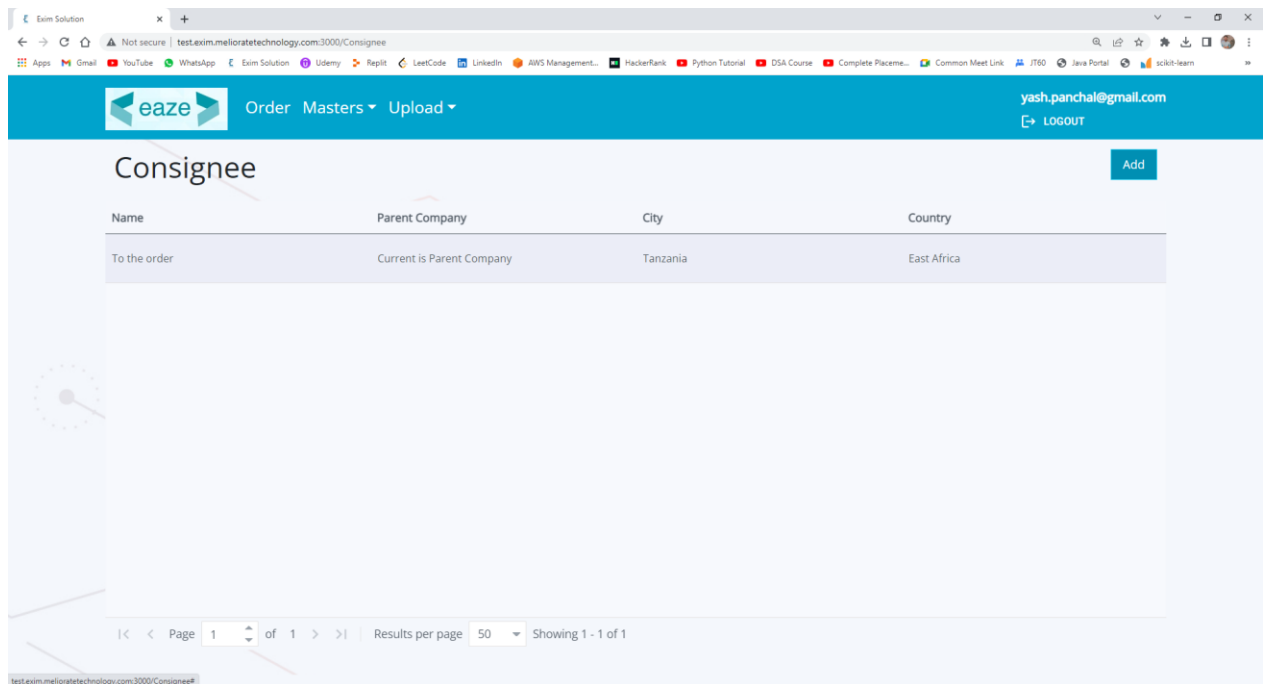
Here we added one company **Pinko Ceramic LLP**.



After that click on consignee means add the details of the client.



Here we have added the details of a client company.



We can add the comment for the order.

The screenshot shows a web browser window with the URL `test.exim.meliortatechnology.com:3000/Comments`. The application header is blue with the 'eaze' logo, navigation links 'Order', 'Masters', and 'Upload', and a user profile 'yash.panchal@gmail.com' with a 'LOGOUT' button. The 'Comments' modal is open, displaying a table with the following data:

Entity Type	Entity	Template Category	Comments	Image	Actions
CLIENT	To the order	Perfoma	Delivery must in 15 days		

At the bottom right of the modal, there is a pagination control: 'Rows per page: 10' and '1-1 of 1'.

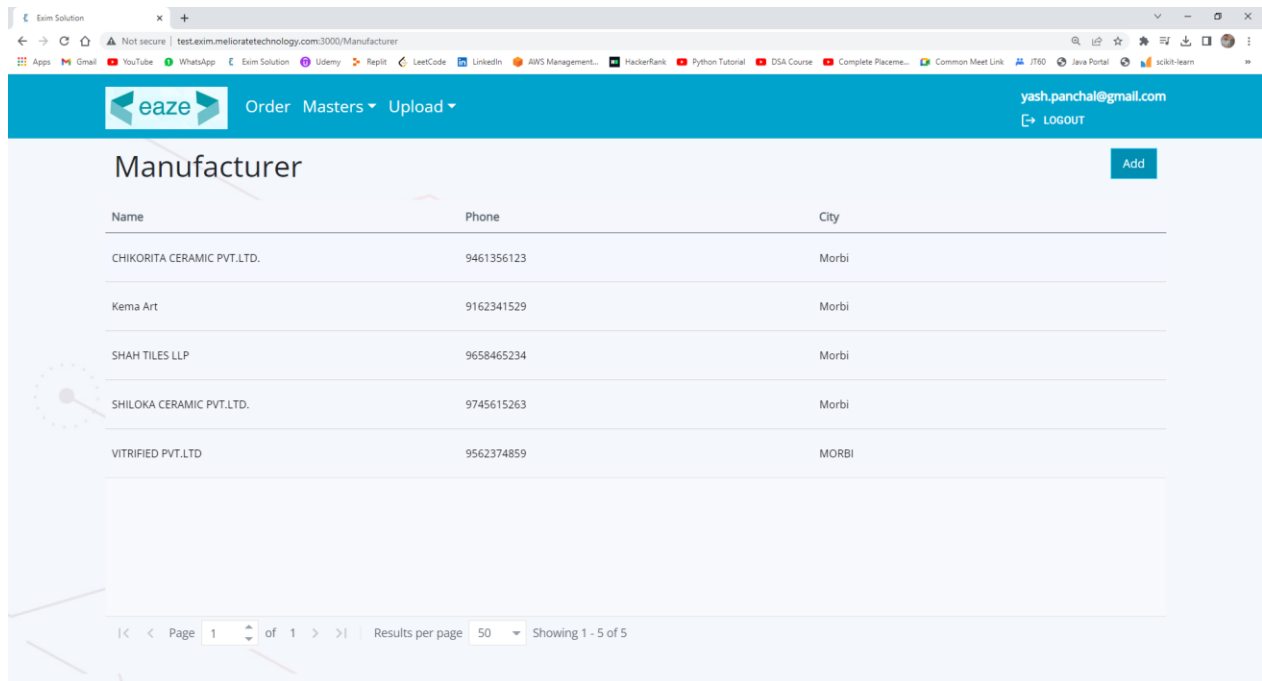
We can notify the client's manager for delivery details.

The screenshot shows the 'Notify Party List' page in the eaze application. The header is the same as the previous screenshot. The page title is 'Notify Party List' with an 'Add' button. Below the title is a table with the following data:

Name	City	Pincode	Country	Phone	Mail
Rahul	CA	56	USA	9562374859	rahul123@gmail.com

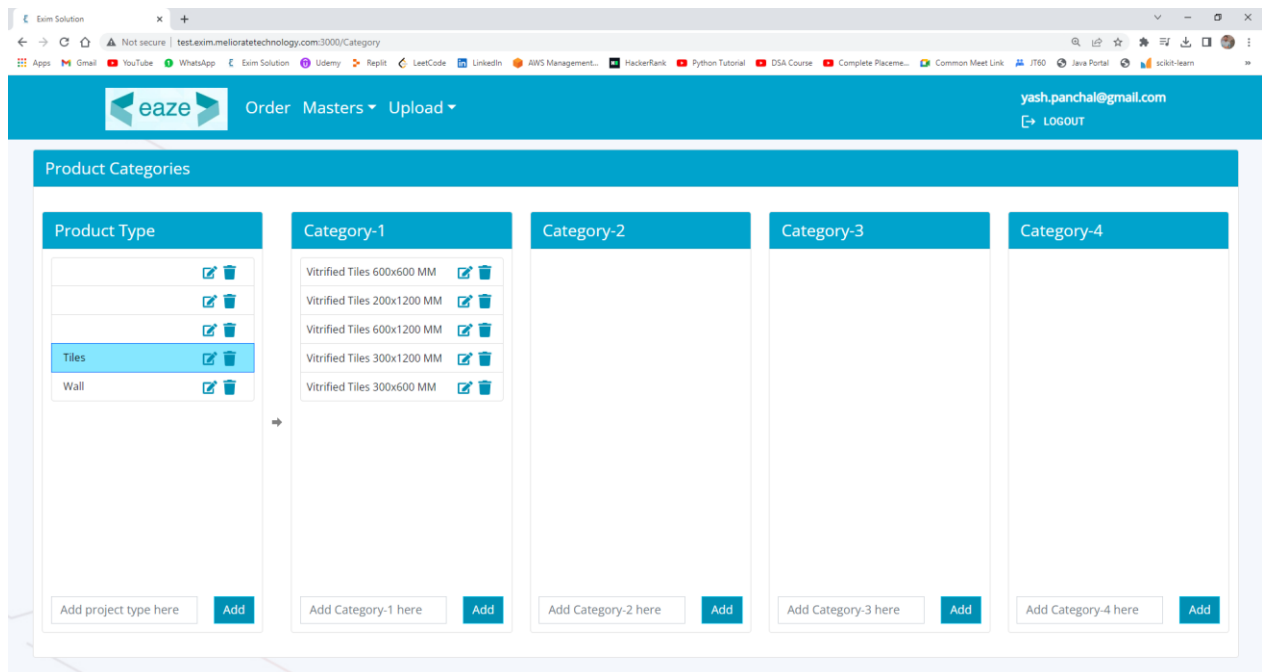
At the bottom of the page, there is a pagination control: 'Page 1 of 1', 'Results per page 50', and 'Showing 1 - 1 of 1'.

List of all Manufacturers



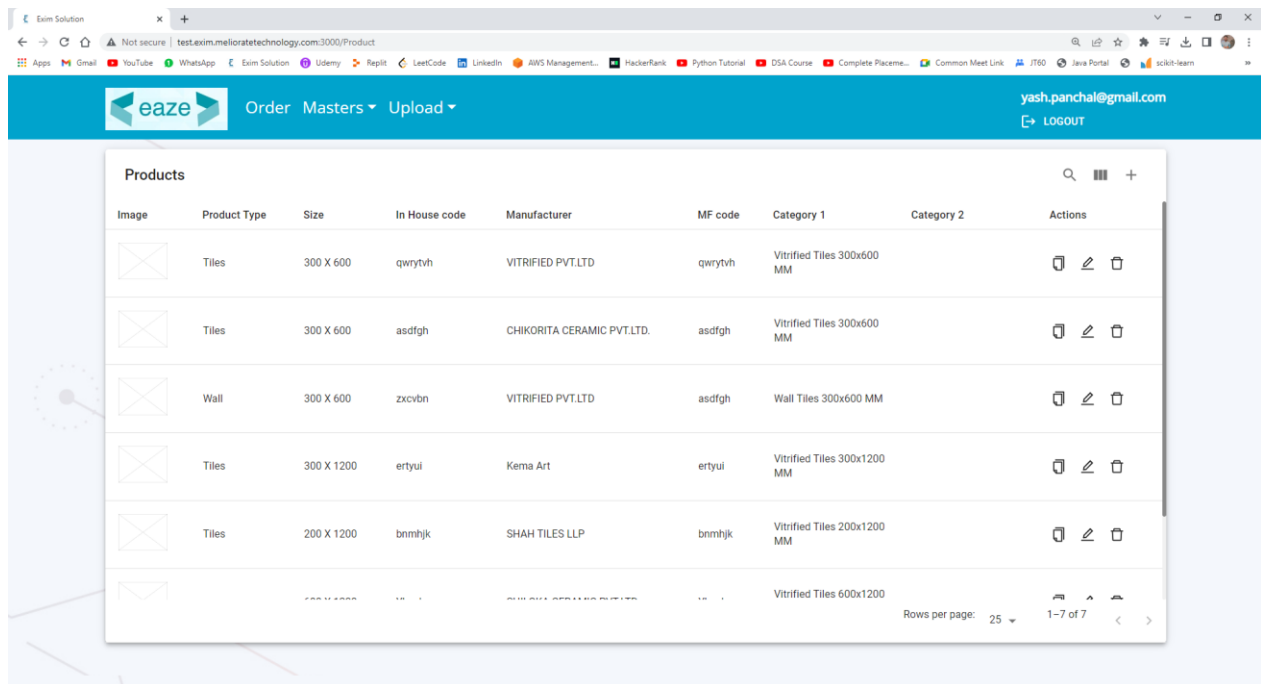
Name	Phone	City
CHIKORITA CERAMIC PVT.LTD.	9461356123	Morbi
Kema Art	9162341529	Morbi
SHAH TILES LLP	9658465234	Morbi
SHILOKA CERAMIC PVT.LTD.	9745615263	Morbi
VITRIFIED PVT.LTD	9562374859	MORBI

Here I have to add the all details tiles and walls.

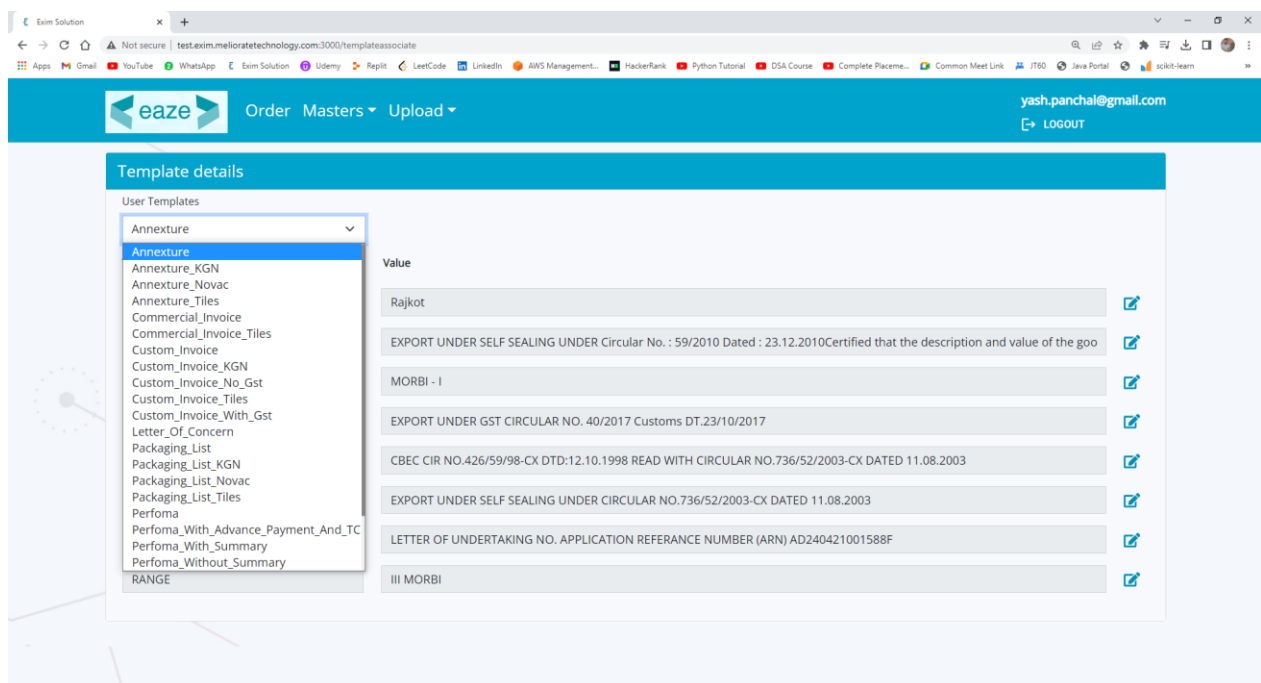


Product Type	Category-1	Category-2	Category-3	Category-4
Vitrified Tiles 600x600 MM				
Vitrified Tiles 200x1200 MM				
Vitrified Tiles 600x1200 MM				
Vitrified Tiles 300x1200 MM				
Vitrified Tiles 300x600 MM				

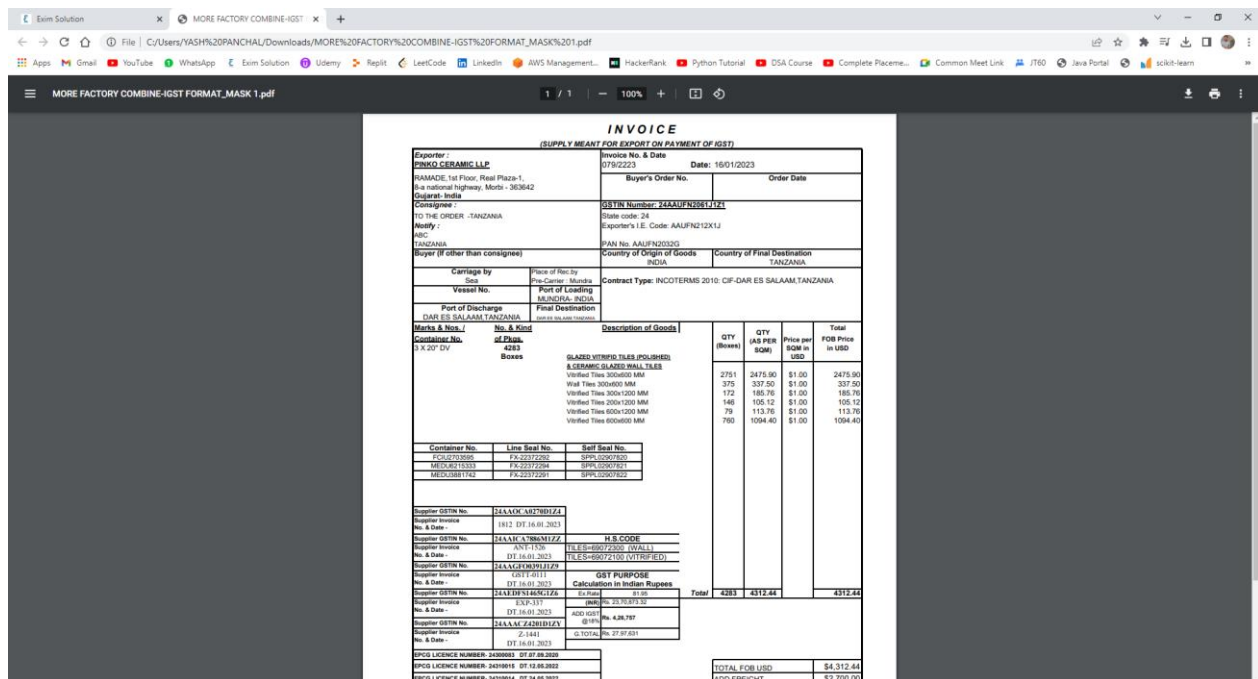
Here I need to add the details of tiles like size, product type, size, In house code, Manufacturer, MF code, Category, etc.



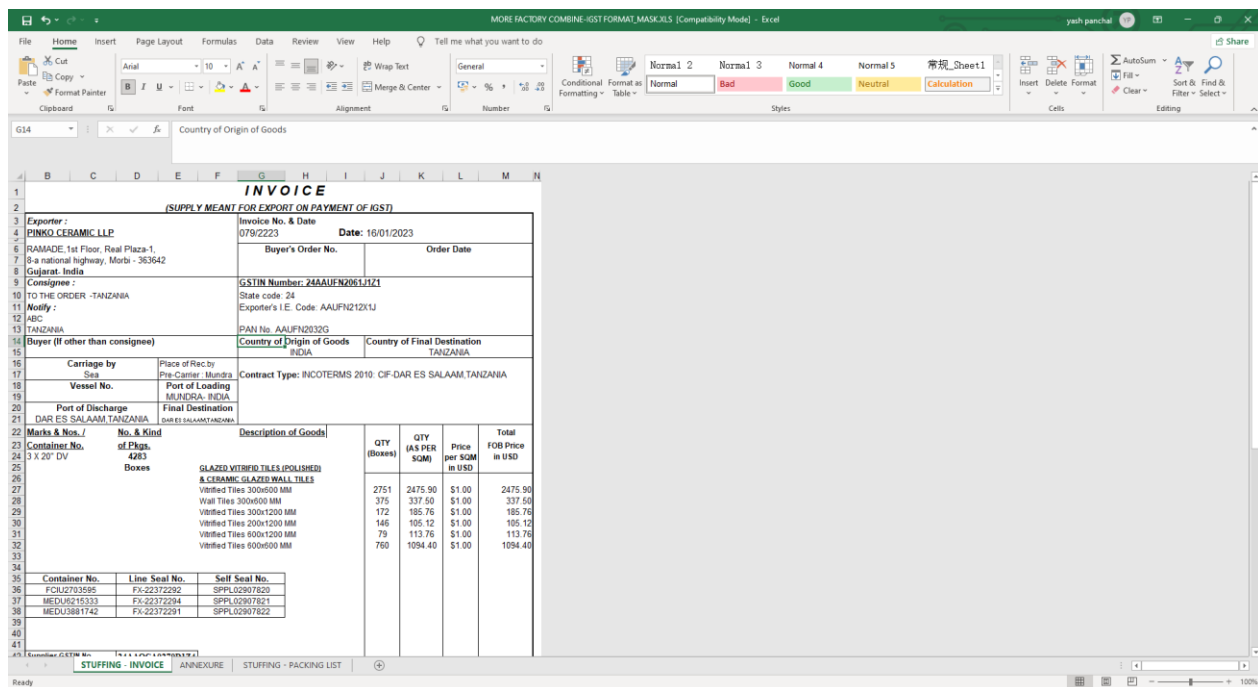
Then I have to download all the documentation part.



Then all the document is downloaded in pdf format so here at the same time it will be converted into an excel sheet.



Excel report generation.



CHAPTER 4

CONCLUSION AND FUTURE SCOPE

Improved data extraction: As PDF files come in various formats and layouts, there is scope for further development of data extraction methods to enhance the accuracy and efficiency of the process. Techniques such as machine learning and natural language processing could be employed to improve data extraction from PDF files.

Integration with other libraries and tools: JXL library can be integrated with other libraries and tools to further enhance its functionality. For example, integration with Apache POI (another Java library for working with Microsoft Office files) can provide more extensive Excel features.

Cross-platform compatibility: JXL library currently only supports Windows and Linux operating systems. Future developments could focus on making the library compatible with other operating systems such as macOS.

Cloud-based conversion: With the increasing use of cloud-based computing, there is scope for developing a cloud-based solution for converting PDF data to Excel using JXL library. This would allow users to access the conversion process from anywhere and at any time, without the need for installing the JXL library or any other software on their local machines.

User interface development: While JXL library is a powerful tool for developers, non-technical users may find it challenging to use. Future development could focus on developing a user-friendly interface for the conversion process, allowing non-technical users to convert PDF data into Excel easily.

REFERENCES

- [1] JXL Official Website: <http://jexcelapi.sourceforge.net/>
- [2] JXL GitHub Repository: <https://github.com/centic9/jexcelapi>
- [3] JXL JavaDocs: <http://jexcelapi.sourceforge.net/resources/javadocs/current/>
- [4] TutorialsPoint JXL Tutorial: <https://www.tutorialspoint.com/jexcel/index.htm>
- [5] JXL Tutorial by Avajava: <https://www.avajava.com/tutorials/categories/jexcel/index.html>
- [6] JavaBeat JXL Tutorial: <https://javabeat.net/jxl-tutorial/>
- [7] Mkyong JXL Tutorial: <https://www.mkyong.com/java/jexcel-api-read-write-excel-file-in-java/>
- [8] Convert PDF to Excel in Java using Apache PDFBox and JXL:
<https://www.roytuts.com/convert-pdf-to-excel-in-java-using-apache-pdfbox-and-jxl/>
- [9] How to extract table data from PDF and export to Excel using Java:
<https://www.codingame.com/playgrounds/2290/how-to-extract-table-data-from-pdf-and-export-to-excel-using-java>
- [10] PDF to Excel in Java: Convert PDF data to Excel using JExcel API:
<https://www.tutorialspoint.com/pdf-to-excel-in-java-convert-pdf-data-to-excel-using-jexcel-api>
- [11] Convert PDF to Excel in Java using iText and JExcel API: <https://www.roytuts.com/convert-pdf-to-excel-in-java-using-itext-and-jexcel-api/>

ANNEXURE

(Note: All entries of the proforma of approval should be filled up with appropriate and complete information. Incomplete proforma of approval in any respect will be summarily rejected.)

GR No.: **3511278**.....

Roll no: **02**.....

1. Name of the Student: **PANCHAL YASH PANCHABHAI**
2. Title of the Project: **Exim Solution**
3. Name of the Guide: **Prof. Namrathaa Malu**
4. Teaching/Industry experience of the Guide: **04**
5. Is this your first submission? **Yes**



Signature of the Student:

Signature of the Guide:

Date:

Date:

Signature of the Coordinator:

Date: