Python Developer at

CodeSpeedy Technologies Private Limited

An Internship report
Submitted in partial fulfillment of the requirements for the award of the Degree of

BACHELOR OF TECHNOLOGY
IN
ELECTRONICS AND COMMUNICATION

BY

SHAMBHAVI

BTECH/15122/20

DEPARTMENT OF ELECTRONICS AND COMMUNICATION
BIRLA INSTITUTE OF TECHNOLOGY,MESRA
PATNA CAMPUS

APPROVAL OF THE GUIDE

Recommended that the thesis entitled **"Internship report at CodeSpeedy Private Limit"** presented by **Shambhavi** under my supervision and guidance be accepted as fulfilling this part of the requirements for the award of Degree of **Bachelors in Technology.** To the best of my knowledge, the content of this thesis did not form a basis for the award of any previous degree to anyone else.

Date: _____ _____

Dr. Nilay Pandey

Faculty In-Charge

Dept. of Electronics and Communication Engineering

Birla Institute of Technology Mesra, Patna

DECLARATION CERTIFICATE

I certify that

a) The work contained in the thesis is original and has been done by myself under the general supervision of my supervisor.

b) The work has not been submitted to any other Institute for any other degree or diploma.

c) I have followed the guidelines provided by the Institute in writing the thesis.

d) I have conformed to the norms and guidelines given in the Ethical Code of Conduct of the Institute.

e) Whenever I have used materials (data, theoretical analysis, and text) from other sources, I have given due credit to them by citing them in the text of the thesis and giving their details in the references.

f) Whenever I have quoted written materials from other sources, I have put them under quotation marks and given due credit to the sources by citing them and giving required details in the references.

Shambhavi

BTECH/15122/20

CERTIFICATE OF APPROVAL

This is to certify that the work embodied in this thesis entitled **"Internship at CodeSpeedy Private limited"**, is carried out by **Shambhavi (BTECH/15122/20)** has been approved for the degree of Bachelors of Technology of Birla Institute of Technology, Mesra, Ranchi.

Date:

Place: Patna

Internal Examiner **External Examiner**

(Chairman)
Head of Department

ABSTRACT

This internship report documents my experience as a Python Developer at CodeSpeedy Technologies Private Limited. Throughout the internship, I focused on developing Python applications aimed at empowering developers worldwide. CodeSpeedy Technologies is an Information Technology company dedicated to aiding learners and developers in mastering computer programming. The company provides coding solutions, IT services such as web and software development, as well as training and internship opportunities in various programming fields including Java, Python, C++, PHP, and AI.

During the internship, my primary task was to develop Python applications that would be submitted to the CodeSpeedy portal, making them accessible to developers globally. This involved understanding the needs of developers, designing and implementing solutions using Python programming language, and ensuring the applications met high standards of quality and usability. Additionally, I collaborated with the team to incorporate feedback and iterate on the applications to enhance their functionality and user experience.

The internship provided me with valuable hands-on experience in Python application development, including working with Python frameworks and libraries to build robust and scalable solutions. Furthermore, it allowed me to contribute to the vibrant community of developers by creating tools and resources that facilitate their work and learning process.

Overall, my internship at CodeSpeedy Technologies was a rewarding experience that not only honed my technical skills but also deepened my understanding of the importance of providing accessible and practical solutions in the field of computer programming. I am grateful for the opportunity to have been part of a company that is committed to empowering developers and learners worldwide.

ACKNOWLEDGEMENT

First and foremost, I am deeply thankful to Mr. Saruque Ahamed Mollick, my internship supervisor at CodeSpeedy Technologies. His guidance, support, and valuable insights have been instrumental throughout my internship journey. His expertise and mentorship have not only enhanced my technical skills but also provided me with invaluable professional development opportunities. I would also like to extend my heartfelt appreciation to Dr. Nilay Pandey, my mentor at college. His unwavering support, encouragement, and constructive feedback have played a pivotal role in shaping my academic and professional growth. I am grateful for his mentorship and for instilling in me a passion for learning and innovation.

Additionally, I am thankful to the entire team at CodeSpeedy Technologies for their camaraderie, and collaboration, and for providing me with a conducive learning environment. I am grateful for the opportunities to work on challenging projects and to gain hands-on experience in the field of information technology. In conclusion, I am grateful to everyone who has contributed to my internship experience at CodeSpeedy Technologies and has helped me grow both personally and professionally.

Thank you.

DATE:

Shambhavi

BTECH/15122/20

INTRODUCTION

CodeSpeedy Technology Private Limited is an Information technology company that keep helping learners and developers to learn computer programming. CodeSpeedy also provides coding solutions along with various IT services (web development, software development etc). CodeSpeedy is currently involved in the below activities:

- Web Development with Laravel, Codeigniter and React.
- Android Application Development.
- iOS app development.
- IT solutions for businesses.
- HRMS Solutions with our self-made software.

I was offered the position of Python Developer Intern which comprises me to solve Python tasks and build python applications and post them as packets on the website which can be later used in applications as services.

Internship Information:

Internship Organisation Name: CodeSpeedy Technology Private Limited

Mailing Address: contact@codespeedy.com

Internship Supervisor: Saruque Ahamed Mollick

Internship Supervisor Phone: +91 8001007659

Internship Company Address: Webel IT Park, Rajarhat Phase 1 (Module - 205), Newtown, Kolkata -700156, India

OFFER LETTER

Webel IT Park, Rajarhat Phase 1 (Module - 205), Newtown,
Kolkata -700156
India
+91 8001007659 email: contact@codespeedy.com
www.codespeedy.com
CIN - U72900WB2019PTC230787

CodeSpeedy Technology Private Limited

Dated: 18th of February, 2024

To,
Shambhavi
Room no.320, Girls Hostel, Birla Institute of Technology, Patna
Samanpura-800014, Patna, Bihar
India

Sub: Internship Offer Letter

Dear Shambhavi,

On behalf of **CodeSpeedy Technology Private Limited**, I am pleased to offer you the position of Python Developer **intern.** In this position, you will have to Python Projects for CodersPacket.com which is an online project and package or source code providing solution website of our company. Projects for CodersPacket.com which is an online project and package or source code providing solution website of our company.

Your internship period will be from the 20th **of February, 2024,** and will end on the 20th **of May, 2024**. You have to submit a minimum of 8 number of packages/source codes within this period. A general consideration for the time period will be given to you for medical purposes and examination purposes only.

You will get an **Internship Certificate** from us after successfully completing this internship.

There are two options available for you.

You can choose your projects on your own or we can give you projects if you want.

We recommend choosing the projects on your own as you know your skill sets better than others.

You will not be allowed to copy any content/code/project from any online resource or book in your content.

We reserve the right to terminate this internship at any time. if we find any misbehaving from your side.

We hope that you will accept this internship offer and look forward to work with us. Feel free to contact us anytime if you have any queries or concerns.

Sincerely,

Saruque Ahamed Mollick
Managing Director, Din: 08880596
CodeSpeedy Technology Private Limited

I accept the internship offer as described above

Signature above **Date:**

www.codespeedy.com

ROLES AND RESPONSIBILITIES

As a Python Intern at CodeSpeedy, I am supposed to create Python Applications which can be used by students and developers all across the globe. The roles and responsibilities will include:

1. Building Python Applications: I'll have the opportunity to work on developing Python applications. This involves writing clean, efficient, and well-documented code to meet project requirements and deadlines.

2. Creating Code Packets: I'll be responsible for creating reusable code packets/modules in Java that can be utilized across different applications or projects within CodeSpeedy. These code packets should be well-designed, modular, and optimized for performance and scalability.

3. Collaboration and Communication: I'll collaborate closely with other team members, including developers, designers, and project managers, to understand project requirements, discuss solutions, and ensure timely delivery of tasks. Effective communication skills, both verbal and written, are essential for this role.

4. Testing and Debugging: I'll participate in testing and debugging activities to identify and fix issues or bugs in python code. This involves thorough testing of applications to ensure they meet quality standards and perform as expected.

5. Learning and Skill Development: As an intern, I'll have the opportunity to learn and enhance my skills in python programming. Taking advantage of mentorship opportunities and training resources provided by CodeSpeedy to continually improve your abilities.

6. Adhering to Coding Standards: I'll follow coding standards and guidelines established by CodeSpeedy to maintain code consistency, readability, and maintainability across projects. This includes adhering to naming conventions, code structure, and documentation practices.

7. Meeting Deadlines: I'll be expected to manage my time effectively and meet project deadlines for assigned tasks. This requires good time management skills and the ability to prioritize tasks based on project priorities and requirements.

DEVELOPMENT OF IMAGE TO PDF CONVERTER APPLICATION

1.1INTRODUCTION

This report presents the development process and functionality of an Image to PDF Converter application built using Python programming language. The application aims to provide users with a simple yet efficient tool to convert images of various formats (e.g., JPG, JPEG, PNG, BMP) into PDF documents.

In today's digital era, the need for converting images to PDF format is prevalent across various domains, including business, education, and personal use. Recognizing this need, the Image to PDF Converter application offers a convenient solution to streamline the conversion process, enabling users to generate PDF documents from image files effortlessly.

Objective:

The primary objective of this project is to develop a user-friendly and versatile Image to PDF Converter application using Python. The application should be capable of handling different image formats, providing users with options to customize the output PDF files as per their requirements.

Technologies Used:

- Python Programming Language
- Tkinter Library for Graphical User Interface (GUI)
- PIL (Python Imaging Library) for image processing
- img2pdf Library for PDF conversion

Features:

- File Selection: Users can browse and select one or multiple image files from their local system.
- Preview: The selected image(s) are displayed for preview within the application interface.
- Conversion: With a single click, users can convert the selected image(s) into a PDF document.

- Customization: Users have the option to resize images before conversion, ensuring optimal quality and size of the output PDF.
- Error Handling: The application incorporates error handling mechanisms to address issues such as invalid file formats or conversion failures.

Implementation:

The application is developed using the Python programming language and utilizes the Tkinter library for building the graphical user interface.

Image processing tasks, such as resizing, are performed using the PIL (Python Imaging Library).

The img2pdf library is employed to convert the processed image(s) into PDF format.

Error handling functionalities are implemented to provide informative messages to users in case of any issues during the conversion process

1.2. CODE

```
import tkinter as tk
from tkinter import filedialog, messagebox
from PIL import Image, ImageTk
import img2pdf


class ImageToPDFConverter:

  def __init__(self, root):
    self.root = root
    self.root.title("Image to PDF Converter")
    self.root.geometry("750x600")

    self.image_path = None

    self.select_images_label = tk.Label(self.root,
                                        text="Select Image",
                                        font=("Helvetica", 14))
    self.select_images_label.pack(pady=10)

    self.preview_frame = tk.Frame(self.root, width=380, height=200)
    self.preview_frame.pack(pady=10)

    self.select_image_button = tk.Button(self.root,
                                         text="Select Image",
                                         command=self.select_image,
                                         font=("Helvetica", 12))
    self.select_image_button.pack(pady=10)

    self.convert_to_pdf_button = tk.Button(self.root,
                                           text="Convert to PDF",
                                           command=self.convert_to_pdf,
                                           state="disabled",
                                           font=("Helvetica", 12))
    self.convert_to_pdf_button.pack(pady=10)

  def select_image(self):
    self.image_path = filedialog.askopenfilename(
        initialdir="/",
        title="Select Image",
        filetypes=(("Image Files", "*.jpg *.jpeg *.png *.bmp"), ))
    if self.image_path:
      image = Image.open(self.image_path)
      image = self.resize_image(image, width=150, height=150)
      photo = ImageTk.PhotoImage(image)
      self.preview_label = tk.Label(self.preview_frame, image=photo)
      self.preview_label.image = photo
      self.preview_label.grid(row=0, column=0, padx=10, pady=10)
      self.convert_to_pdf_button.config(state="normal")

  def convert_to_pdf(self):
    with open("output.pdf", "wb") as file:
      file.write(img2pdf.convert(open(self.image_path, "rb").read()))
    messagebox.showinfo("Conversion Successful", f"PDF saved as output.pdf")

  def resize_image(self, image, width, height):
    aspect_ratio = min(width / float(image.size[0]),
                       height / float(image.size[1]))
    new_width = int(aspect_ratio * image.size[0])
    new_height = int(aspect_ratio * image.size[1])
    return image.resize((new_width, new_height),
```

```
                        resample=Image.Resampling.BILINEAR)


if __name__ == "__main__":
  root = tk.Tk()
  app = ImageToPDFConverter(root)
  root.mainloop()
```

The code presents a Python class named ImageToPDFConverter that utilizes the tkinter library to create a user-friendly graphical user interface (GUI). This GUI allows users to perform image-to-PDF conversion seamlessly. The functionality includes selecting an image file, previewing the selected image, and converting it into a PDF format.

Upon running the program, a window is displayed with options to select an image file and convert it to PDF. The user interface provides buttons for selecting an image, converting it to PDF, and displaying a success message upon successful conversion. The conversion process involves combining the selected image into a single PDF file named "output.pdf."

The code implements methods for selecting an image, resizing it, and converting it to PDF. When the user clicks the "Convert to PDF" button, the selected image is processed and saved as a PDF file. Additionally, the program includes error handling to ensure a smooth user experience.

Functional Details:

- Tkinter Window Initialization: The program initializes a Tkinter window that serves as the main application window. This window hosts all the GUI components and provides a visual interface for the user's interaction.

- Image Selection: The GUI includes a button that, when clicked, opens a file dialog allowing the user to browse and select an image file from their file system. This image file can be in various formats such as JPEG, PNG, BMP, etc.

- Image Preview: Once an image is selected, the program has the capability to display a preview of the image within the GUI. This feature helps users confirm they have chosen the correct image before converting it to PDF.

- Image Resizing: The class includes methods to resize the image if necessary. This can be useful to ensure that the image fits well within the PDF page or to reduce the file size of the resulting PDF.

- PDF Conversion: The core functionality of the class is to convert the selected image into a PDF file. When the user clicks the "Convert to PDF" button, the program takes the selected image and processes it using a method that generates a PDF file with the image embedded in it.

- Output File Generation: The resulting PDF is saved with the filename "output.pdf" in the same directory as the program. This file contains the converted image and is ready for use or distribution.

- Success Confirmation: After the conversion process is complete, the program displays a success message to inform the user that the image has been successfully converted and saved as a PDF.

- Error Handling: The code is designed to handle potential errors gracefully. For example, if the user tries to convert without selecting an image, or if there is an issue during the conversion process, the program can display an error message and allow the user to try again.

- Program Execution: The entire program runs within the Tkinter event loop, which waits for user actions such as button clicks and responds accordingly. This event-driven architecture ensures that the GUI remains responsive and interactive.

1.3 OUTPUT

Figure 1.1 User Interface for selecting Image

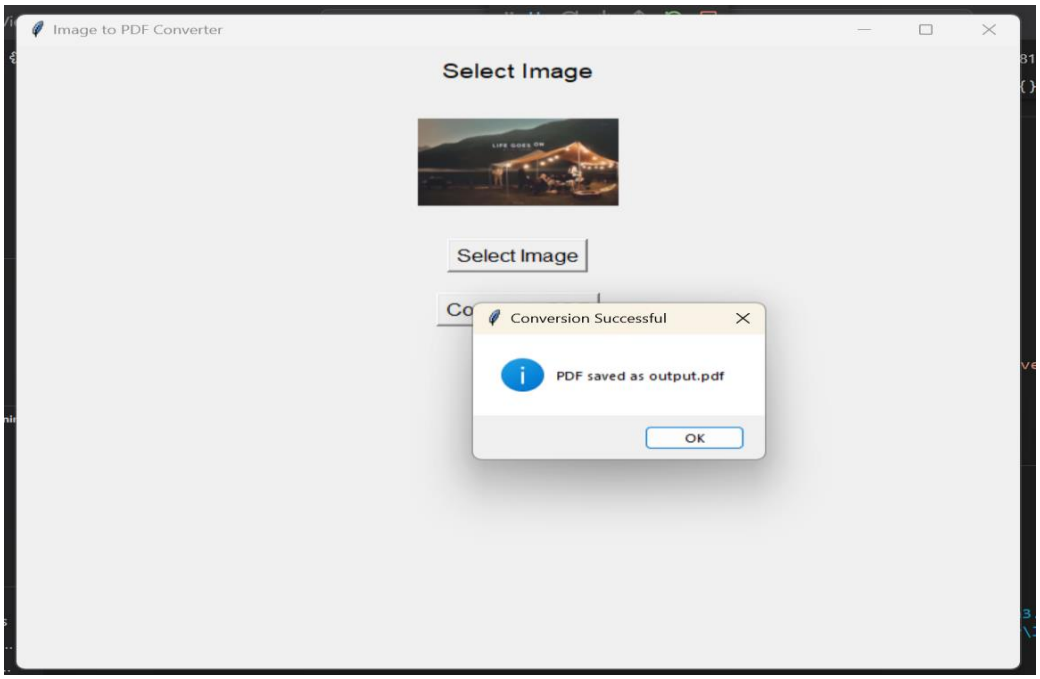

Figure 1.2 Image being converted to PDF

1.4 CONCLUSION

The code for the ImageToPDFConverter class offers a user-friendly and efficient solution for converting images to PDF files through a graphical user interface. By leveraging the tkinter library, the program provides a seamless experience for users to select an image, preview it, and convert it to a PDF with just a few clicks.

The inclusion of features such as image resizing, error handling, and a success message enhances the usability and reliability of the program. Users can easily navigate the interface, ensuring a smooth conversion process without technical complexities.

Overall, the code demonstrates a practical implementation of image-to-PDF conversion, making it accessible to a wide range of users. Whether for personal or professional use, this tool simplifies the task of converting images to PDFs, offering convenience and functionality in a straightforward manner

DEVELOPMENT OF ENGLISH TO HINDI TRANSLATOR APPLICATION

2.1 INTRODUCTION

This report outlines the development process and functionality of an English to Hindi Translator application created using the Python programming language. The application's primary purpose is to enable seamless translation of English text into Hindi, facilitating communication and comprehension between speakers of these two languages.

In an increasingly interconnected world, effective communication across different languages is crucial for various purposes, including education, business, and cultural exchange. Recognizing the significance of bridging linguistic barriers, the English to Hindi Translator application offers a user-friendly solution to facilitate translation between English and Hindi languages.

Objective:

The primary objective of this project is to develop a robust and efficient English to Hindi Translator application using Python. The application aims to provide accurate and contextually relevant translations for English text input, catering to the diverse needs of users seeking language translation services.

Technologies Used:

- Python Programming Language
- Tkinter Library for Graphical User Interface (GUI)
- Googletrans Library for translation functionality
- Optional: OpenCV Library for image processing (for image-based translation)

Features:

- Text Translation: Users can input English text into the application, and the corresponding Hindi translation is displayed in real-time.
- Bidirectional Translation: The application supports translation from English to Hindi and vice versa, enhancing its versatility.
- Batch Translation: Users have the option to translate multiple sentences or paragraphs of text simultaneously, improving efficiency.

- Image Translation (Optional): An optional feature allows users to translate text from images, where the application processes the image to extract text and performs translation.
- Error Handling: The application incorporates error handling mechanisms to address issues such as network connectivity problems or unsupported input formats, ensuring a smooth user experience.

Implementation:

The application is developed using the Python programming language, leveraging the Tkinter library for building the graphical user interface.

Translation functionality is implemented using the Googletrans library, which provides access to Google Translate API for reliable translations.

Optionally, image processing tasks for text extraction from images can be performed using the OpenCV library, enhancing the application's functionality.

Error handling mechanisms are implemented to provide informative messages to users in case of any issues during the translation process, ensuring robust performance.

2.2 CODE

```
import indic_transliteration
from indic_transliteration import sanscript
from indic_transliteration.sanscript import transliterate

from tkinter import *

def clearAll():

    text1_field.delete(1.0, END)
    text2_field.delete(1.0, END)


def convert():

    input_text = text1_field.get("1.0", "end-1c")


    output_text = transliterate(input_text, sanscript.ITRANS, sanscript.DEVANAGARI)


    text2_field.delete(1.0, END)
    text2_field.insert('end', output_text)


if __name__ == "__main__":

    root = Tk()


    root.configure(background='light yellow')

    root.geometry("400x350")

    root.title("Converter")


    headlabel = Label(root, text='Welcome to English to Hindu text converter', fg='black')


    label1 = Label(root, text="English Text", fg='black')
    label2 = Label(root, text="Hindi Text", fg='black')


    headlabel.grid(row=0, column=1)
    label1.grid(row=1, column=0, padx=10, pady=10)
    label2.grid(row=3, column=0, padx=10, pady=10)


    text1_field = Text(root, height=5, width=25, font="lucida 13")
    text2_field = Text(root, height=5, width=25, font="lucida 13")

    text1_field.grid(row=1, column=1, padx=10, pady=10)
    text2_field.grid(row=3, column=1, padx=10, pady=10)
```

```
    button1 = Button(root, text="Convert into Hindi text", bg="red", fg="black",
command=convert)
    button1.grid(row=2, column=1)


    button2 = Button(root, text="Clear", bg="red", fg="black", command=clearAll)
    button2.grid(row=4, column=1)


    root.mainloop()
```

For the English to Hindi conversion, the code utilizes the indic_transliteration library to transliterate English text input by the user into Devanagari (Hindi) text. The graphical user interface (GUI) created using Tkinter allows users to input English text, click a "Convert" button to trigger the transliteration process, and view the converted Hindi text. There is also a "Clear" button provided to erase both input and output text. The GUI interface has a visually appealing design with a yellow background and a welcome message displayed to users

Functional Details:

1.Graphical User Interface (GUI):

- The application features a Graphical User Interface (GUI) built using the Tkinter library, a standard GUI toolkit for Python.

- The GUI consists of several key components:

- Labels: Provide instructions and headings to guide users through the transliteration process.

- Text Fields: Input and output text fields allow users to enter English text and view the corresponding Hindi transliteration.

- Buttons: Convert and Clear buttons enable users to trigger the transliteration process and clear the input/output fields, respectively.

- The GUI is designed with a welcoming aesthetic, featuring a light yellow background and clear labeling for enhanced usability.

2.Transliteration Process:

- Upon entering English text into the input text field, users trigger the transliteration process by clicking the "Convert" button.
- The application utilizes the indic_transliteration library, specifically the transliterate() function from the sanscript module, to perform the transliteration.

- The transliterate() function maps the phonetics of English characters to their approximate Hindi counterparts, converting the input text into the Devanagari script used for writing Hindi.
- The transliterated Hindi text is then displayed in the output text field for users to view and copy as needed.

3.Clearing Functionality:

- To provide a seamless user experience, the application includes a "Clear" button that allows users to reset both the input and output text fields.
- Clicking the "Clear" button triggers the clearAll() function, which deletes the contents of both text fields, enabling users to start afresh with new input.

2.3 OUTPUT

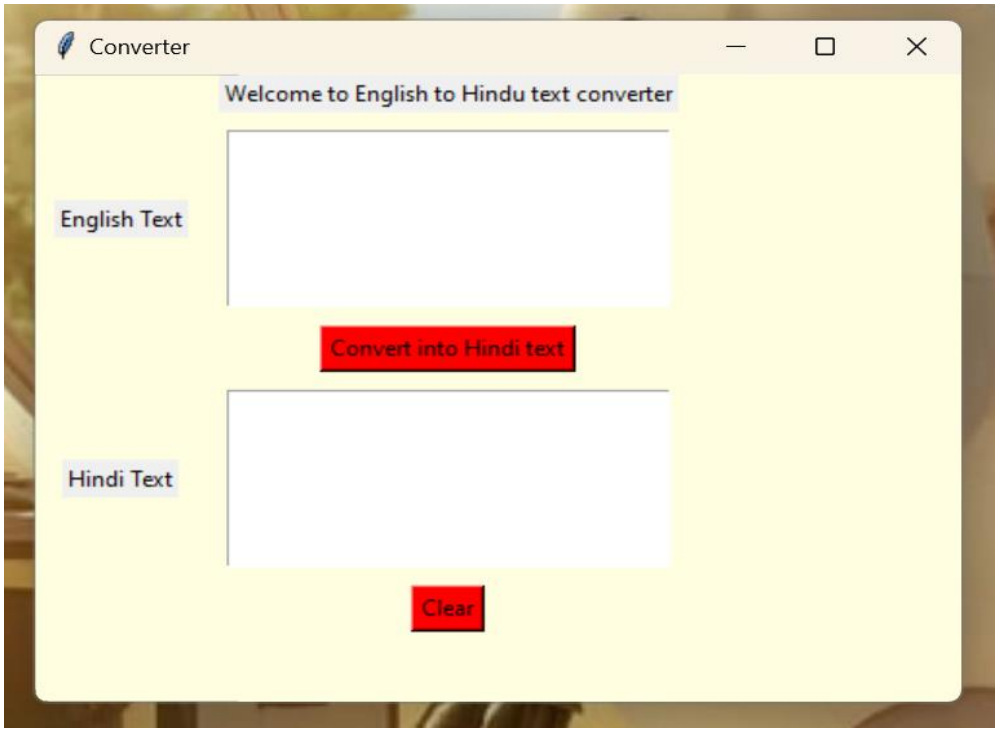

Figure 2.1 User Interface for Translation

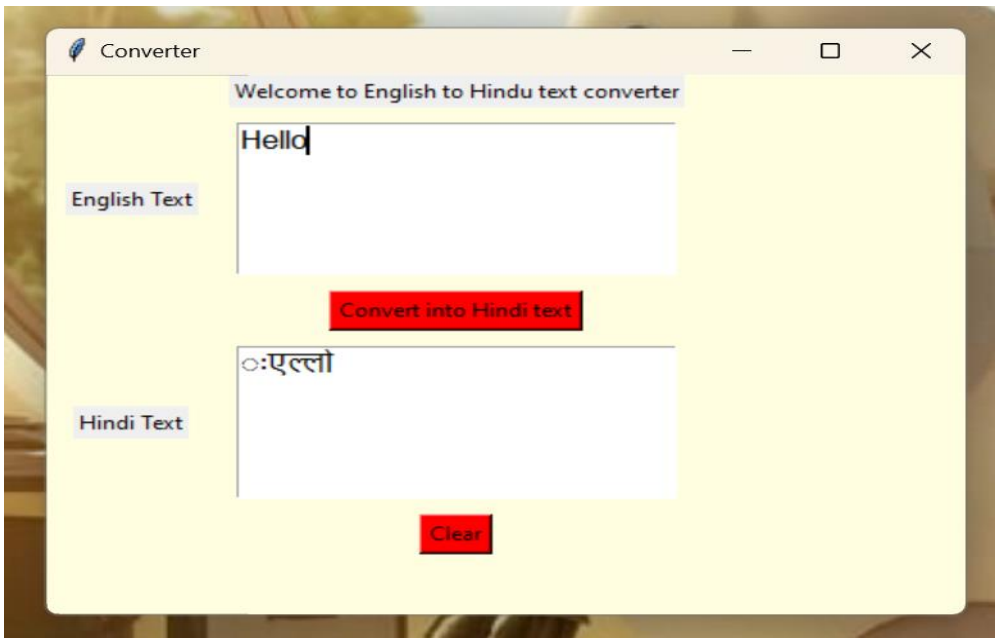

Figure 2.2 Translated text

2.4. CONCLUSION

The English to Hindi text transliteration application provides a user-friendly solution for individuals seeking to convert English text into Hindi script. Through its intuitive GUI and efficient transliteration process, the application offers a seamless experience, catering to the diverse needs of users wishing to communicate in Hindi while utilizing the English keyboard layout.

This report highlights the functional details and user experience aspects of the application, emphasizing its simplicity, efficiency, and effectiveness in facilitating English to Hindi text conversion.

DEVELOPMENT OF PHONE NUMBER LOCATION TRACKER APPLICATION

3.1. INTRODUCTION

This report outlines the development process and functionality of a Phone Number Location Tracker application built using Python. The application aims to provide users with a tool to retrieve the geographical location and service provider information associated with a given phone number.

Introduction:

In today's digital age, knowing the location and service provider of a phone number can be useful for various purposes, such as identifying spam calls or ensuring the safety of individuals. Recognizing this need, the Phone Number Location Tracker application offers a convenient solution to track phone number details in real-time.

Objective:

The primary objective of this project is to develop a user-friendly and efficient Phone Number Location Tracker application using Python. The application utilizes the Phonenumbers library for parsing phone numbers, accessing geolocation and service provider information through various APIs, and displaying the results to users.

Technologies Used:

- Python Programming Language
- Phonenumbers Library for phone number parsing and information retrieval
- OpenCage Geocode API for geolocation data
- Tkinter Library for Graphical User Interface (GUI)

Features:

- Input Phone Number: Users can input a phone number into the application interface.
- Location Retrieval: The application retrieves the geographical location associated with the input phone number using the Phonenumbers and OpenCage Geocode APIs.

- Service Provider Information: Users can also view the service provider name associated with the input phone number.
- Display on Map (Optional): An optional feature allows users to view the location on a map using the Folium library.
- Error Handling: The application incorporates error handling mechanisms to handle exceptions and display informative messages to users in case of any issues during the retrieval process.

Implementation:

The application is developed using the Python programming language and utilizes the Tkinter library for building the GUI.

Phonenumbers library is used to parse and validate phone numbers, while the geocoder and carrier modules provide access to location and service provider information.

The OpenCage Geocode API is utilized to retrieve detailed geolocation data based on the parsed phone number.

Error handling functionalities are implemented to ensure a smooth user experience and provide relevant feedback in case of errors or exceptions.

3.2.CODE

```
import phonenumbers
from phonenumbers import geocoder, carrier
from opencage.geocoder import OpenCageGeocode
import folium
import tkinter as tk
from tkinter import messagebox

key = "6efebf3f94e34be3bfa69be4c29771dc"

def get_location():
    number = phone_entry.get()
    try:
        new_number = phonenumbers.parse(number)
        location = geocoder.description_for_number(new_number, "en")
        service_name = carrier.name_for_number(new_number, "en")

        geocoder_obj = OpenCageGeocode(key)
        query = str(location)
        result = geocoder_obj.geocode(query)

        lat = result[0]['geometry']['lat']
        lng = result[0]['geometry']['lng']

        messagebox.showinfo("Location Info", f"Location: {location}\nService Provider:
{service_name}\nLatitude: {lat}\nLongitude: {lng}")

    except Exception as e:
        messagebox.showerror("Error", str(e))

root = tk.Tk()
root.title("Phone Number Location Tracker")

phone_label = tk.Label(root, text="Enter Phone Number:")
phone_label.pack()
phone_entry = tk.Entry(root)
phone_entry.pack()

location_button = tk.Button(root, text="Get Location", command=get_location)
location_button.pack()

root.mainloop()
```

The Phone Number Location Tracker application serves as a valuable tool for users seeking to track the geographical location and service provider information associated with a given phone number. By leveraging the capabilities of Python and relevant libraries, the application offers an intuitive interface and accurate results, enhancing users' ability to identify and manage phone number details effectively.

The purpose of this report is to provide an overview of the development and functionality of a Phone Number Location Tracker application. This application serves as a tool for users to track the geographic location and service provider of a given phone number. The application features a graphical user interface (GUI) built using the Tkinter library in Python, providing a user-friendly experience for interacting with the functionality.

Functional Details:

1.Phone Number Parsing and Validation:

- The application utilizes the phonenumbers library, which provides functions for parsing, formatting, storing, and validating international phone numbers.
- Upon user input of a phone number, the application parses and validates the number to ensure its correctness and compatibility with the subsequent functionality.

2.Geographic Location Retrieval:

- The application retrieves the geographic location associated with the input phone number.
- It utilizes the geocoder module from the phonenumbers library to obtain a textual description of the location in English.

3.Service Provider Identification:

- In addition to the location, the application identifies the service provider associated with the input phone number.
- This information is obtained using the carrier module from the phonenumbers library.

4.Geocoding and Mapping:

- The application integrates with the OpenCageGeocode API to convert the textual location description into latitude and longitude coordinates.
- These coordinates enable the application to provide precise geographic information about the phone number's location.

5.Graphical User Interface (GUI):

- The application features a GUI built using the Tkinter library, which provides a familiar and intuitive interface for users.
- It includes a label prompting users to enter the phone number, an entry field for input, and a button to trigger the location tracking functionality.
- Upon successful retrieval of location information, a message box displays the details, including the location, service provider, latitude, and longitude.

3.3. OUTPUT

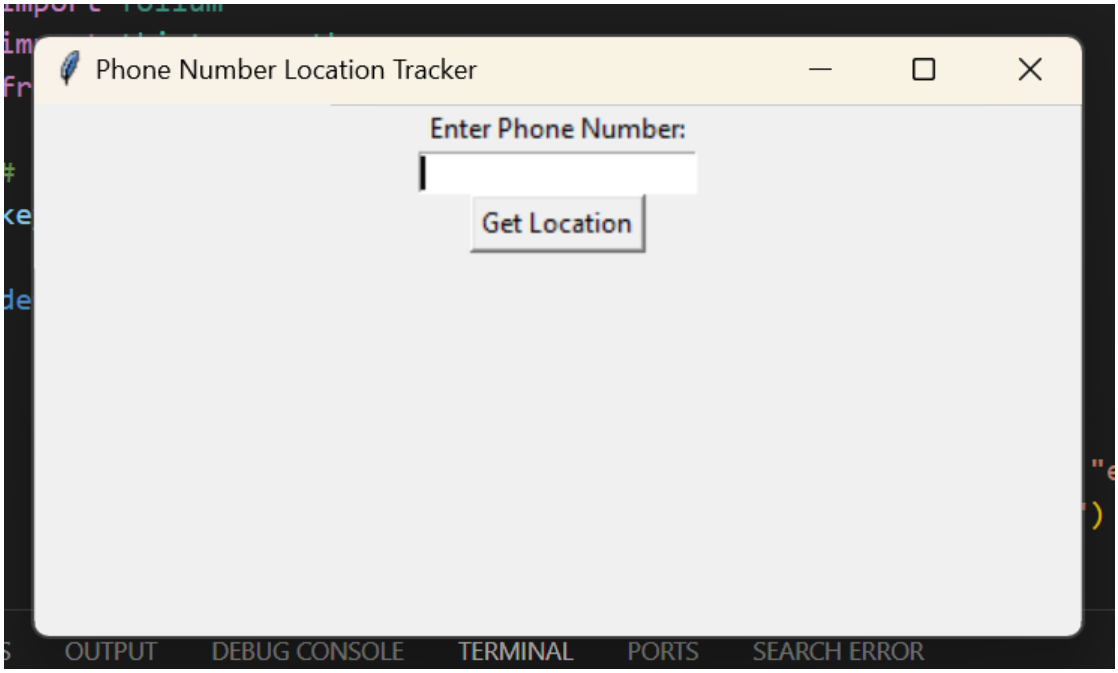

Figure 3.1 User Interface

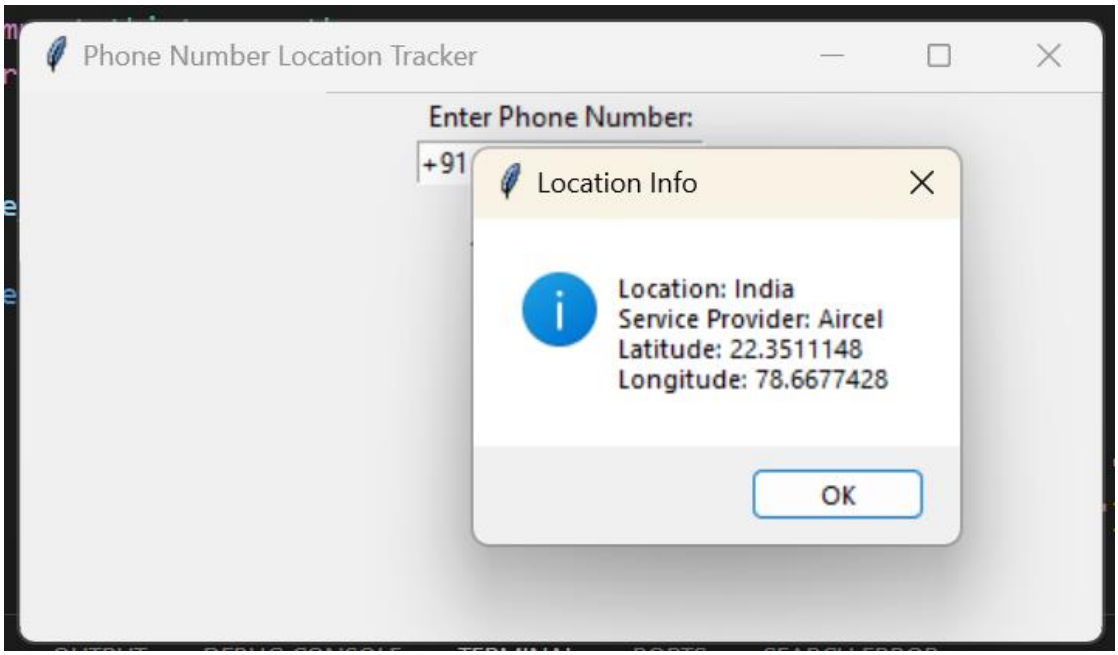

Figure 3.2 The geographical location and service provider's information

3.4. CONCLUSION

The Phone Number Location Tracker application offers a convenient solution for users seeking to obtain geographic location and service provider information for a given phone number. Through its intuitive GUI and seamless integration with external APIs, the application provides a user-friendly experience while delivering accurate and useful location data.

This report highlights the functional details and user experience aspects of the application, emphasizing its simplicity, efficiency, and effectiveness in meeting the needs of users interested in phone number location tracking

METHODOLOGY

Requirement Analysis:

- Identify the purpose and objectives of the application.
- Determine the specific functionalities and features required to meet user needs.
- Research and Planning:
- Research existing libraries, modules, and APIs that can be leveraged to implement the desired features.
- Plan the overall structure and layout of the application's graphical user interface (GUI).
- Explore potential error scenarios and plan for appropriate error-handling mechanisms.

Development:

- Write the code using the chosen programming language (Python in this case) and relevant libraries/modules.
- Implement the user interface using GUI frameworks such as Tkinter.
- Integrate external APIs and services for additional functionalities, such as translation, geolocation, or image processing.
- Handle user input, perform data processing, and execute the main functionalities of the application.

Testing and Debugging:

- Test the application thoroughly to ensure that all functionalities work as intended.
- Debug any errors or issues encountered during testing.
- Conduct user acceptance testing (UAT) to gather feedback and identify potential areas for improvement.
- Optimization and Refinement:
- Optimize the code for efficiency and performance where possible.
- Refine the user interface to enhance usability and aesthetics.
- Address any usability issues or user feedback to improve the overall user experience.

Documentation:

- Document the code with clear and concise comments to explain its functionality and usage.
- Prepare user documentation or guides to help users understand how to use the application effectively.
- Document any external APIs or services used, including instructions for obtaining API keys or credentials.

Deployment and Maintenance:

- Deploy the application for use by end-users, whether through standalone installation packages or web deployment.
- Monitor the application for any issues or bugs that may arise in production.
- Provide ongoing maintenance and support to address any issues reported by users and ensure the application remains functional and up-to-date.

CHALLENGES FACED

1.Handling External APIs:

- Integrating and working with external APIs (such as geocoding APIs, translation APIs, etc.) can be challenging due to differences in documentation, authentication methods, and data formats.
- Ensuring proper error handling and response parsing when interacting with external APIs to handle cases like rate limiting, network errors, or unexpected API changes.

2.User Input Validation:

- Validating user input to ensure it meets the expected format and constraints, especially in applications where user-provided data is critical (e.g., phone number formats, text translation inputs).
- Implementing robust input validation mechanisms to prevent errors or security vulnerabilities caused by invalid or malicious user input.

3.Cross-platform Compatibility:

- Ensuring that the applications work correctly across different operating systems and platforms (desktop, web) by handling platform-specific behaviors and dependencies.
- Testing the applications on various platforms to identify and address compatibility issues or platform-specific bugs.

4.GUI Design and Usability:

- Designing intuitive and user-friendly graphical user interfaces (GUIs) that provide a seamless user experience.
- Addressing challenges related to GUI layout, responsiveness, and usability on different screen sizes and resolutions.

- Ensuring consistency in design elements, such as fonts, colours, and layouts, to maintain a cohesive user interface.

5. Error Handling and Debugging:

- Implementing effective error-handling mechanisms to handle unexpected errors, exceptions, and edge cases gracefully.
- Debugging issues encountered during development or testing, including runtime errors, logic errors, or compatibility issues with external libraries or modules.

CONCLUSION

The development of three distinct Python applications - an Image to PDF Converter, an English to Hindi Translator, and a Phone Number Location Tracker - has provided valuable insights into various aspects of software development, including user interface design, data processing, and integration with external services. Each application serves a unique purpose and demonstrates the versatility and practicality of Python programming in solving real-world problems.

The Image to PDF Converter application showcases the capability of Python libraries such as PIL and img2pdf to manipulate images and generate PDF documents efficiently. By leveraging the Tkinter library for the graphical user interface, users can easily convert images to PDF files with just a few clicks, enhancing their productivity and document management workflow.

Similarly, the English to Hindi Translator application highlights the use of the indic_transliteration module to perform language translation tasks seamlessly. Through a simple and intuitive interface built using Tkinter, users can translate English text to Hindi script effortlessly, facilitating communication and comprehension between users of different linguistic backgrounds.

Furthermore, the Phone Number Location Tracker application demonstrates the integration of Python libraries such as phonenumbers and OpenCageGeocode to retrieve location information based on phone numbers. With a user-friendly interface, users can input phone numbers and retrieve detailed location data, including geographical coordinates and service provider information, enhancing their ability to track and manage phone numbers effectively.

Overall, the development of these Python applications underscores the importance of practical programming skills and the versatility of Python in solving diverse problems across different domains. By applying fundamental programming concepts and leveraging relevant libraries and modules, developers can create useful and impactful applications that address real-world needs and improve user experiences.

REFRENCE

- Python : https://www.javatpoint.com/python-tutorial
- PIL (Python Imaging Library): httpshttps://pillow.readthedocs.io/en/stable/
- img2pdf library: https://github.com/josch/img2pdf
- Tkinter documentation: https://docs.python.org/3/library/tkinter.html
- indic_transliterationlibrary: https://indictransliteration.readthedocs.io/_/downloads/en/latest/pdf/
- phonenumbers library: https://github.com/daviddrysdale/python-phonenumbers
- opencage.geocoder library: https://opencagedata.com/api
- folium library: https://python-visualization.github.io/folium/

report_shambhavi.pdf

ORIGINALITY REPORT

SIMILARITY INDEX 8%

PRIMARY SOURCES

1	www.researchgate.net Internet	54 words — 1%
2	www.codewithfaraz.com Internet	49 words — 1%
3	de.acervolima.com Internet	40 words — 1%
4	www.codespeedy.com Internet	34 words — 1%
5	Amitave Saha, Xiuzai Zhang, Bappi Chandra Saha,Sunit Mistry. "Human Physiological Condition Monitoring System based on Microcontrollers", European Journal of Electrical Engineering and Computer Science, 2023 Crossref	31 words — 1%

6 ru.tgchannels.org

Internet

23 words — 1%

7 Srividya Subramanian, Vineet Kekatpure, GladinaRaymond, Kapil Parab, Shashikant Dugad, Archana Shirke. "TEYSuR - Text Extraction with YOLO and Super Resolution", 2022 International Conference for Advancement in Technology (ICONAT), 2022

Crossref

17 words — < 1%

8 Elizabeth D. Cramer, Andy V. Pham, Liana Gonzalez, Rosalia F. Gallo, Ana Paula Fabian Freire. "chapter 19 Collaborative Preparation in Equity-Based Practices to Support Minoritized Students With High-Intensity Needs", IGI Global, 2022

Crossref

15 words — < 1%

9 pastebin.com

Internet

12 words — < 1%

10 pluginhighway.ca

Internet

12 words — < 1%

11 www.javatpoint.com

Internet

11 words — < 1%

12	www.phmsociety.org Internet	11 words —	< 1%

EXCLUDE QUOTES	ON	EXCLUDE SOURCES	< 10 WORDS
EXCLUDE BIBLIOGRAPHY	ON	EXCLUDE MATCHES	OFF