USER INTERFACE DESIGN EXERCISE-3

Roll no:230701211

Difference Between CLI (Command Line Interface) and

GUI (Graphical User Interface) and VUI (Voice User Interface)

CLI

- **Definition**: A text-based interface where users interact with the system by typing commands.
- **Interaction**: Users type commands into a terminal or console window to execute tasks**.**
- **Input/Output**: Input is given via text commands, and output is displayed in text format.
- **Advantages:** Fast and efficient for advanced users, consumes fewer system resources, and offers powerful features.
- **Disadvantages:** Requires knowledge of specific commands, not user-friendly for beginners.

Example: Linux terminal, Windows Command Prompt.

IMPLEMENTATION:

```python
import os
import sys

def  rename_file(old_name, new_name):
    try:
        os.rename(old_name, new_name)
        print(f"File renamed from {old_name} to {new_name}")
    except FileNotFoundError:
        print(f"Error: {old_name} not found.")
    except Exception as e:
        print(f"An error occurred: {e}")

if __name__ == "__main__":
    if len(sys.argv) != 3:
        print("Usage: python rename_file_cli.py <old_filename> <new_filename>")
    else:
        rename_file(sys.argv[1], sys.argv[2])
```

OUTPUT:

```
PS C:\Users\gvmani\OneDrive\Desktop\Python Project> python rename_file_cli.py nikitha.txt niki.txt
File renamed from nikitha.txt to niki.txt
```

GUI:

. **Definition**: A visual interface that allows users to interact with software through icons, buttons, and other graphical elements.

. **Interaction:** Users interact with visual elements such as windows, icons, and menus, often using a mouse or touchscreen.

. **Input/Output**: Input is given by clicking, dragging, and typing, while output is shown through visual components.

. **Advantages:** Intuitive, easy for beginners, and supports multimedia content.

. **Disadvantages**: Can be slower than CLI, consumes more system resources.

Example: Windows, macOS, Android, iOS.

IMPLEMENTATION:

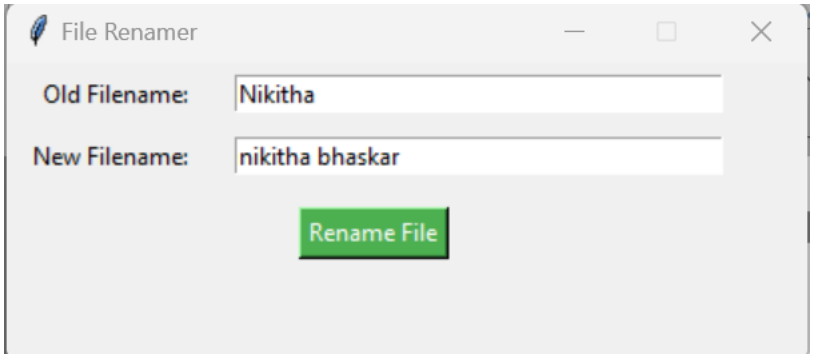
Program

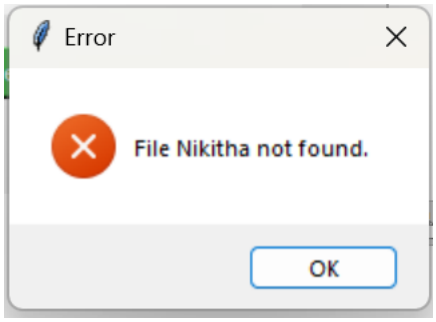
GUI_LAB3.py > ...

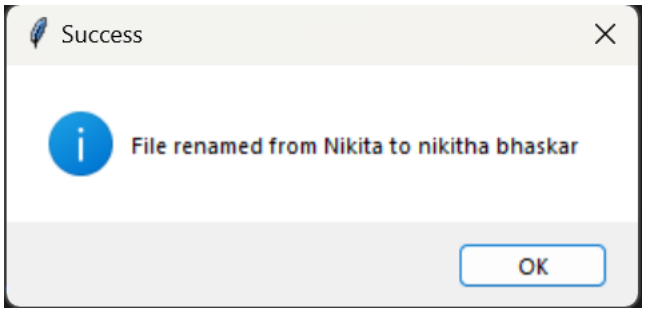
```python
import tkinter as tk
from tkinter import messagebox
import os

def rename_file():
    old_name = old_filename_entry.get().strip()
    new_name = new_filename_entry.get().strip()

    if not old_name or not new_name:
        messagebox.showwarning("Warning", "Both filename fields must be filled.")
        return

    if os.path.exists(new_name):
        messagebox.showwarning("Warning", f"File {new_name} already exists. Overwriting is not recommended
        return

    try:
        os.rename(old_name, new_name)
        messagebox.showinfo("Success", f"File renamed from {old_name} to {new_name}")
    except FileNotFoundError:
        messagebox.showerror("Error", f"File {old_name} not found.")
    except Exception as e:
        messagebox.showerror("Error", f"An error occurred: {e}")

# Set up the main window
root = tk.Tk()
root.title("File Renamer")
root.geometry("400x150")
root.resizable(False, False)
```

```python
# Create and place labels, entries, and buttons
tk.Label(root, text="Old Filename:").grid(row=0, column=0, padx=10, pady=5, sticky="e")
tk.Label(root, text="New Filename:").grid(row=1, column=0, padx=10, pady=5, sticky="e")

old_filename_entry = tk.Entry(root, width=40)
old_filename_entry.grid(row=0, column=1, padx=10, pady=5)

new_filename_entry = tk.Entry(root, width=40)
new_filename_entry.grid(row=1, column=1, padx=10, pady=5)

default_button_color = "#4CAF50"  # Green
rename_button = tk.Button(root, text="Rename File", command=rename_file, bg=default_button_color, fg="whit
rename_button.grid(row=2, column=0, columnspan=2, pady=10)

# Start the Tkinter event loop
root.mainloop()
```

Output:







VUI:

. Definition: An interface where users interact with the system using voice commands and receive feedback via speech or sound.

. Interaction: Users give voice commands, and the system responds with spoken words or other auditory feedback.

. Input/Output: Input is through speech (microphone), and output is through speech (speakers).

. Advantages: Hands-free, natural interaction, accessible for users with disabilities.

. Disadvantages: Limited by recognition accuracy, challenges in noisy environments, and may not support complex tasks.

Example: Amazon Alexa, Google Assistant, Apple Siri.

IMPLEMENTATION:

```
import speech_recognition as sr
import os

def rename_file_from_voice_command(command):
    # Extracting old and new filename from the command
    try:
        words = command.split(" ")
        old_name = words[1]
        new_name = words[3]

        os.rename(old_name, new_name)
        print(f"File renamed from {old_name} to {new_name}")
    except Exception as e:
        print(f"Error: {e}")

def listen_for_command():
    recognizer = sr.Recognizer()
    mic = sr.Microphone()

    print("Listening for command to rename a file...")
    with mic as source:
        recognizer.adjust_for_ambient_noise(source)
        audio = recognizer.listen(source)

    try:
        command = recognizer.recognize_google(audio)
        print(f"Command received: {command}")
        rename_file_from_voice_command(command)
    except sr.UnknownValueError:
        print("Sorry, I couldn't understand the command.")
    except sr.RequestError as e:
        print(f"Could not request results from Google Speech Recognition service; {e}")

if __name__ == "__main__":
    listen_for_command()
```

OUTPUT:

```
Listening for command to rename a file...
Command received: Nikita
Error: list index out of range
```