EXPERIMENT – 3

NAME: Roshini S **ROLL NO:** 240701442

Design and Implement Different User Interfaces to Create a File (CLI, GUI, VUI). Evaluate the usability and interaction differences among interfaces.

---

1. COMMAND LINE INTERFACE (CLI):

Analysis of the Command Line Interface – File Creation

The Command Line Interface represents a **text-based interaction model**, where users type commands manually to perform actions. This interface focuses on simplicity and direct execution.

1. Purpose of the Interface

- Enables users to create files using typed commands.
- Demonstrates fundamental interaction between user and system.
- Provides fast execution without graphical elements.

2. Key Elements and Functional Components

Input Prompt
- Requests the user to enter the file name.
- Allows flexible naming based on user preference.

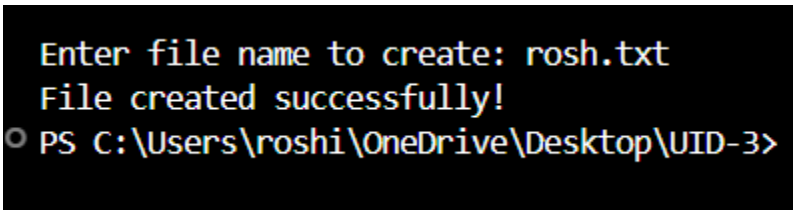
File Creation Logic
- Uses Python's open() function in write mode.
- Automatically generates a new text file.

Error Handling
- Displays error message if file creation fails.

3. Cognitive and UX Benefits

- Minimal interface reduces distraction.
- Enhances technical familiarity and typing accuracy.
- Suitable for advanced users who prefer speed over visuals.

OUTPUT:



---

2. GRAPHICAL USER INTERFACE (GUI):

Analysis of the Graphical User Interface – File Creation

The Graphical User Interface provides a **visual interaction model**, allowing users to perform file creation through buttons and text fields instead of typed commands.

1. Purpose of the Interface

• Offers a beginner-friendly and interactive environment.
• Eliminates the need for technical command knowledge.
• Improves accessibility through visual components.

2. Key Elements and UI Components

Window Layout
• Displays a titled window labeled “Create File – GUI”.
• Fixed geometry ensures consistent appearance.

Text Entry Field
• Allows users to type the file name easily.
• Prevents empty submission using validation.

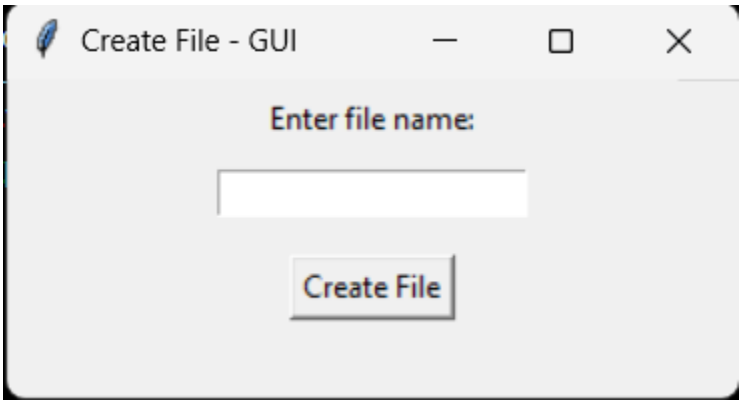
Action Button – “Create File”
• Executes file creation when clicked.
• Enhances user control and clarity.

Message Boxes
• Warning alert for empty input.
• Success alert upon file creation.

3. Cognitive and UX Benefits

• Visual elements improve comprehension.
• Reduces cognitive effort for non-technical users.
• Provides instant feedback through pop-up messages.

OUTPUT:



---

3. VOICE USER INTERFACE (VUI):

Analysis of the Voice User Interface – File Creation

The Voice User Interface represents a **speech-based interaction model**, where users perform actions using voice commands instead of typing or clicking.

1. Purpose of the Interface

• Enables hands-free interaction.
• Demonstrates modern AI-based user communication.
• Improves accessibility for users with physical limitations.

2. Key Elements and Functional Components

Speech Recognition Module
• Captures audio input using a microphone.
• Converts spoken words into text using Google recognition.

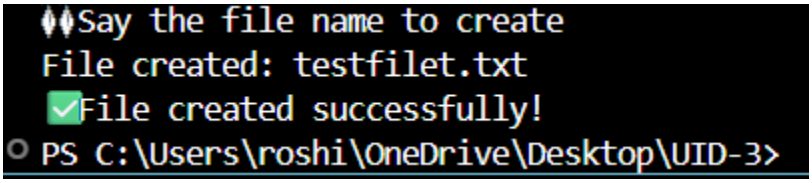
Automatic File Naming
• Removes spaces and appends ".txt" extension.
• Ensures valid file format.

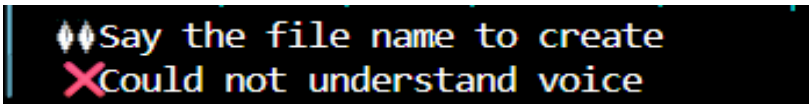
Voice Feedback Messages
• Confirms successful creation.
• Displays error message if voice is unclear.

3. Cognitive and UX Benefits

• Natural interaction style improves comfort.
• Reduces manual effort.
• Encourages inclusive and assistive technology usage.

OUTPUT:





CONCLUSION:

This experiment demonstrates the practical differences between **CLI, GUI, and VUI interaction models** in file creation tasks.

• **CLI** emphasizes speed and technical control.
• **GUI** enhances usability through visuals and feedback.
• **VUI** promotes accessibility and natural interaction.

The comparison highlights how interface design directly influences **user experience, efficiency, and accessibility**, proving that diverse interaction styles cater to different user needs and preferences.