CS23A34

USER INTERFACE DESIGN

EXPERIMENT-3

ROLL NO: 240701194

NAME: ILAKKIYA P

To design and implement a **Smart Queue ManagementSystem** that efficiently manages waiting queues using Command Line Interface (CLI), Graphical User Interface (GUI), and Voice User Interface (VUI).

I. COMMAND LINE INTERFACE

CLI Python Code:

cli_app.py:

```
from queue_logic import SmartQueue
queue = SmartQueue()
print("\nSMART QUEUE MANAGEMENT SYSTEM (CLI)")
print("Commands:")
print(" generate  -> Generate new token")
print(" next      -> Call next token")
print(" status    -> View current queue")
print(" exit      -> Exit system\n")
while True:
    command = input("Enter command: ").lower()
    if command == "generate":
        token = queue.generate_token()
```

```
            print(f"Token generated: {token}")
        elif command == "next":
            token = queue.call_next()
            if token:
                print(f"Now serving token: {token}")
            else:
                print("Queue is empty")
        elif command == "status":
            print("Current Queue:", queue.get_status())
        elif command == "exit":
            print("Exiting system...")
            break
        else:
            print("Invalid command")
```

OutPut:

```
PS C:\Users\jahna\Downloads\SmartQueue-CLI> python cli_app.py

SMART QUEUE MANAGEMENT SYSTEM (CLI)
Commands:
 generate  -> Generate new token
 next      -> Call next token
 status    -> View current queue
 exit      -> Exit system

Enter command: generate
Token generated: 1
Enter command: generate
Token generated: 2
Enter command: generate
Token generated: 3
Enter command: generate
Token generated: 4
Enter command: generate
Token generated: 5
Enter command: next
Now serving token: 1
Enter command: next
Now serving token: 2
Enter command: status
Current Queue: [3, 4, 5]
Enter command: exit
Exiting system...
```

II. GRAPHICAL USER INTERFACE

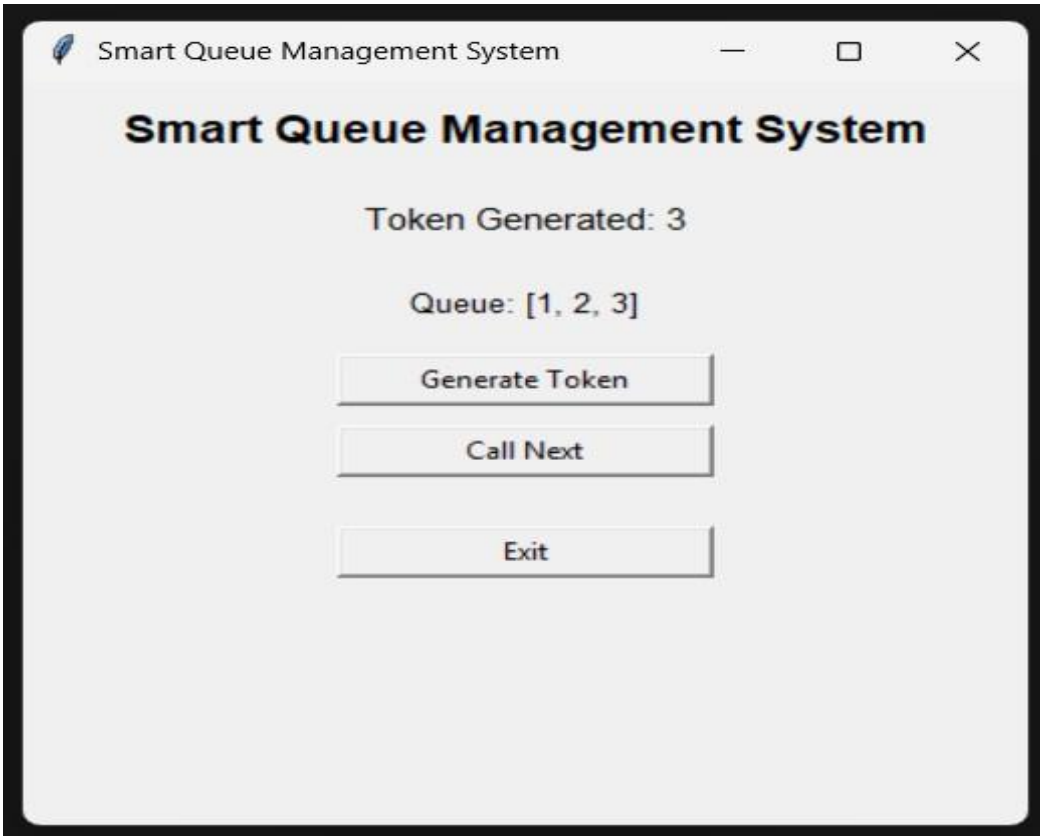
GUI Python Code:

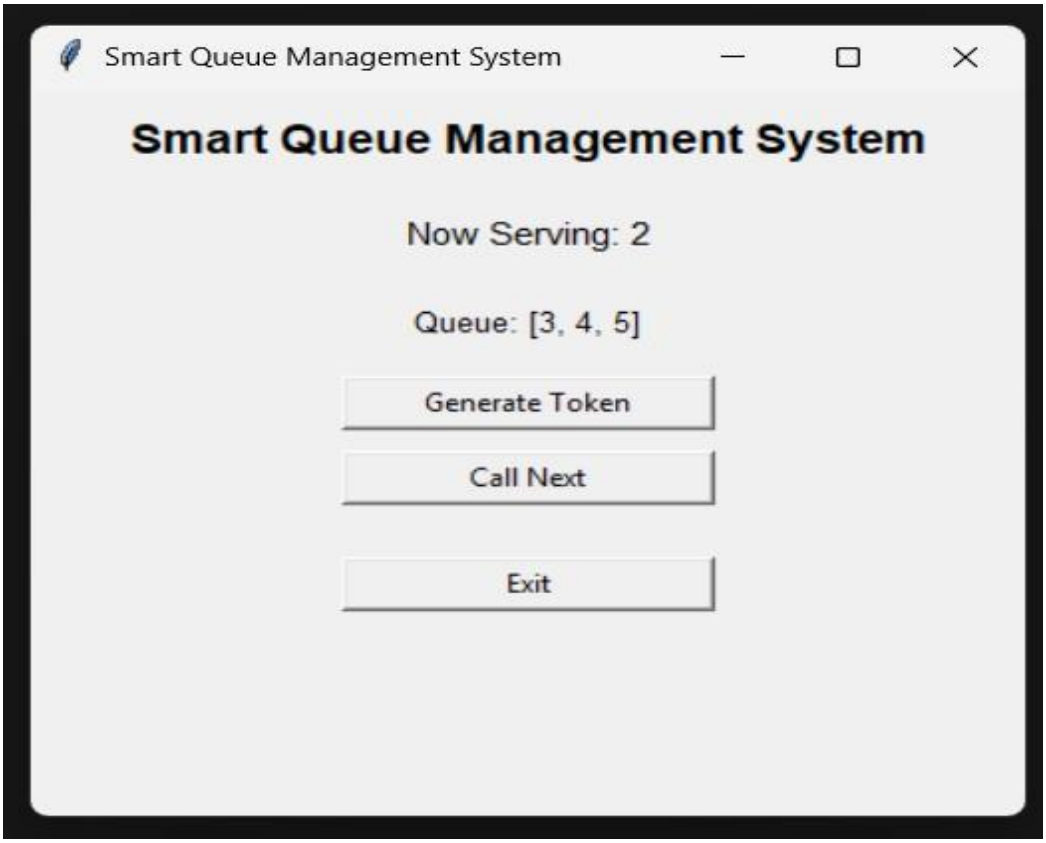
gui_app.py:

```
import tkinter as tk
from queue_logic import SmartQueue
# Create queue object
queue = SmartQueue()
# Main window
window = tk.Tk()
window.title("Smart Queue Management System")
window.geometry("400x350")
# Title label
title = tk.Label(window, text="Smart Queue Management
System", font=("Arial", 14, "bold"))
title.pack(pady=10)
# Output label
output = tk.Label(window, text="", font=("Arial", 11))
output.pack(pady=10)
# Queue status box
queue_box = tk.Label(window, text="Queue: []",
font=("Arial", 10))
queue_box.pack(pady=10)
# Functions
def generate_token():
    token = queue.generate_token()
    output.config(text=f"Token Generated: {token}")
    update_queue()
def call_next():
```

```
    token = queue.call_next()
    if token:
        output.config(text=f"Now Serving: {token}")
    else:
        output.config(text="Queue is empty")
    update_queue()
def update_queue():
    queue_box.config(text=f"Queue: {queue.get_status()}")
# Buttons
btn_generate = tk.Button(window, text="Generate Token",
width=20, command=generate_token)
btn_generate.pack(pady=5)
btn_next = tk.Button(window, text="Call Next", width=20,
command=call_next)
btn_next.pack(pady=5)
btn_exit = tk.Button(window, text="Exit", width=20,
command=window.destroy)
btn_exit.pack(pady=20)
# Run GUI
window.mainloop()
```

OutPut:


Smart Queue Management System
Smart Queue Management System
Token Generated: 3
Queue: [1, 2, 3]
Generate Token
Call Next
Exit


Smart Queue Management System
Smart Queue Management System
Now Serving: 2
Queue: [3, 4, 5]
Generate Token
Call Next
Exit

III. VOICE USER INTERFACE

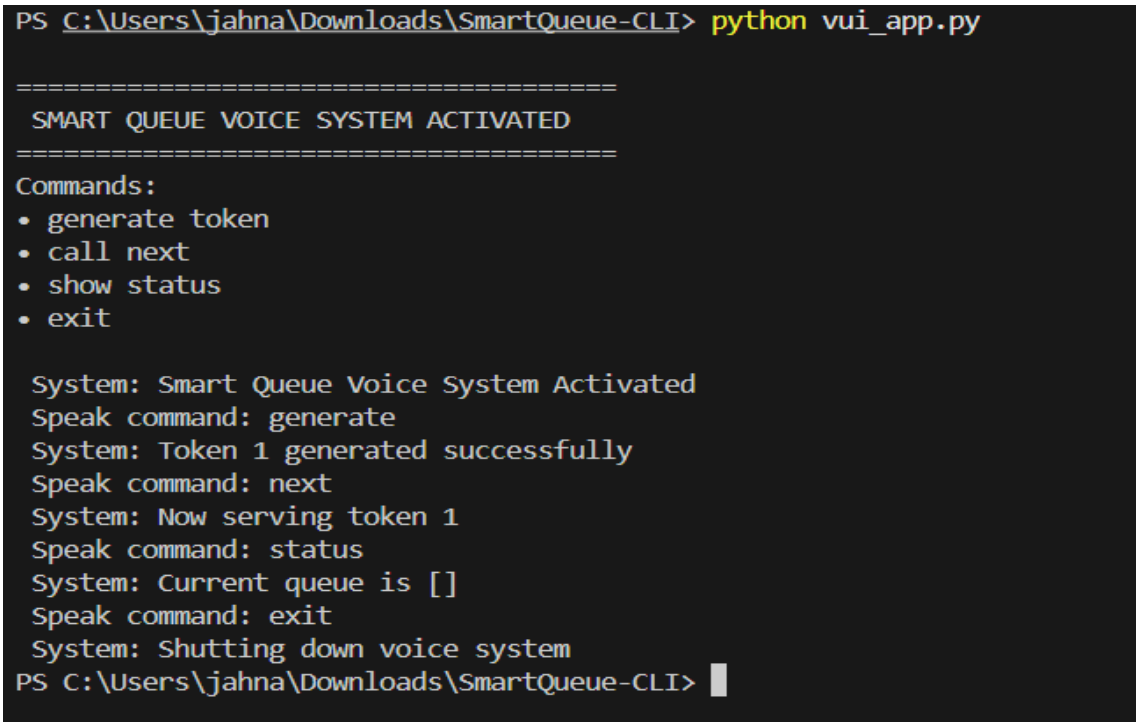
VUI Python Code:

vui_app.py:

```
from queue_logic import SmartQueue
import pyttsx3
import time
# Initialize text-to-speech engine
engine = pyttsx3.init()
engine.setProperty('rate', 170)   # speed of voice
engine.setProperty('volume', 1.0) # volume
def speak(text):
    print(f" System:", text)
    engine.say(text)
    engine.runAndWait()
def listen():
    # Simulated voice input using keyboard
    return input(" Speak command: ").lower()
def main():
    queue = SmartQueue()
    print("\n====================================")
    print(" SMART QUEUE VOICE SYSTEM ACTIVATED")
    print("====================================")
    print("Commands:")
    print("• generate token")
    print("• call next")
    print("• show status")
    print("• exit\n")
    speak("Smart Queue Voice System Activated")
    while True:
        command = listen()
        if "generate" in command:
```

```
            token = queue.generate_token()
            speak(f"Token {token} generated successfully")
        elif "next" in command:
            token = queue.call_next()
            if token:
                speak(f"Now serving token {token}")
            else:
                speak("Queue is empty")
        elif "status" in command:
            status = queue.get_status()
            speak(f"Current queue is {status}")
        elif "exit" in command:
            speak("Shutting down voice system")
            break
        else:
            speak("Sorry, command not recognized")
        time.sleep(0.3)
if __name__ == "__main__":
    main()
```

OutPut:


PS C:\Users\jahna\Downloads\SmartQueue-CLI> python vui_app.py
SMART QUEUE VOICE SYSTEM ACTIVATED
Commands:
• generate token
• call next
• show status
• exit
System: Smart Queue Voice System Activated
Speak command: generate
System: Token 1 generated successfully
Speak command: next
System: Now serving token 1
Speak command: status
System: Current queue is []
Speak command: exit
System: Shutting down voice system
PS C:\Users\jahna\Downloads\SmartQueue-CLI>

OutPut:

```
PS C:\Users\jahna\Downloads\SmartQueue-CLI> python satisfaction_comparision.py

USER SATISFACTION COMPARISON SYSTEM
----------------------------------

Rate the CLI (1 = Poor, 5 = Excellent)
Ease of Use: 4
Speed: 5
User Friendliness: 3
Overall Satisfaction: 4

Rate the GUI (1 = Poor, 5 = Excellent)
Ease of Use: 5
Speed: 5
User Friendliness: 5
Overall Satisfaction: 5

Rate the VUI (1 = Poor, 5 = Excellent)
Ease of Use: 3
Speed: 2
User Friendliness: 3
Overall Satisfaction: 2

--- SATISFACTION SCORES ---
CLI Score: 16/20
GUI Score: 20/20
VUI Score: 10/20

 BEST USER EXPERIENCE
GUI Interface gives highest user satisfaction
PS C:\Users\jahna\Downloads\SmartQueue-CLI>
```