

Lab 4: Dialogue System Design using FSM & State Chartn

Ishat Shivhare
U23AI071

August 20, 2025

Aim

Design a dialogue system model (Flight Ticket Booking Chatbot) using a Finite State Machine (FSM) and a UML State Chart, and implement a minimal, testable text-based prototype.

Objectives

1. Select a system: **Flight Ticket Booking Chatbot**.
2. Identify states and transitions.
3. Draw the FSM diagram.
4. Create a UML State Chart showing normal flow, error handling, and loops.
5. Implement a simple prototype.
6. Test transitions with sample interactions and capture screenshots.

Tools / Software

- Diagramming: draw.io, Mermaid Chart
- Prototype: Python 3
- L^AT_EX on Overleaf for documentation

System Description

Modelled a conversational agent that books a flight for a single passenger. The chatbot elicits destination and date, confirms details, collects payment approval, and issues a ticket. It gracefully handles invalid inputs and cancellation.

FSM Design

States

- Start (implicit initial)
- Greeting
- Ask_Destination
- Ask_Date
- Confirm_Booking
- Payment
- Ticket_Issued
- Error (invalid input handler)
- End

FSM Diagram

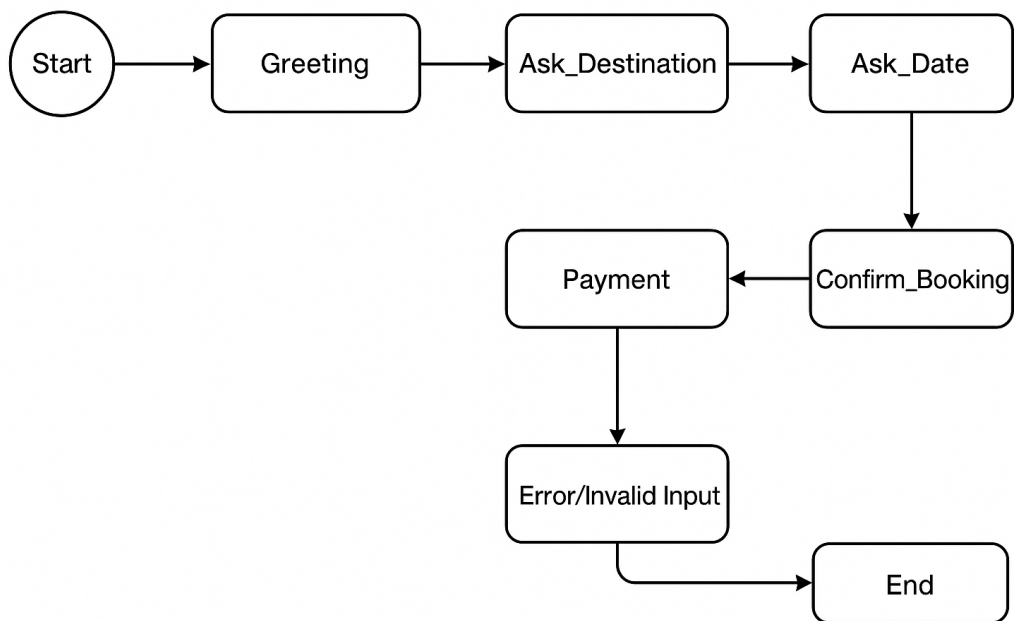


Figure 1: Finite State Machine for Flight Ticket Booking Chatbot

State–Transition Table

| From | Event / Guard | Action | To |
|------------------------|---------------------|-------------------------------------|-----------------------------|
| Start | user greets | send welcome prompt | Greeting |
| Greeting | any | ask destina- tion | Ask_Destination |
| Ask_Destination | valid destination | store desti- nation; ask date | Ask_Date |
| Ask_Destination | invalid/empty | notify error | Error |
| Ask_Date | valid date | store date; show sum- mary | Confirm_Booking |
| Ask_Date | invalid/empty | notify error | Error |
| Confirm_Booking | user says yes | proceed to payment | Payment |
| Confirm_Booking | user says no | cancel flow | End |
| Payment | user approves | issue ticket | Ticket_Issued |
| Payment | user declines/fails | cancel flow | End |
| Error | after message | re-ask rele- vant field | Ask_Destination or Ask_Date |
| Ticket_Issued | done | thank user | End |

UML State Chart

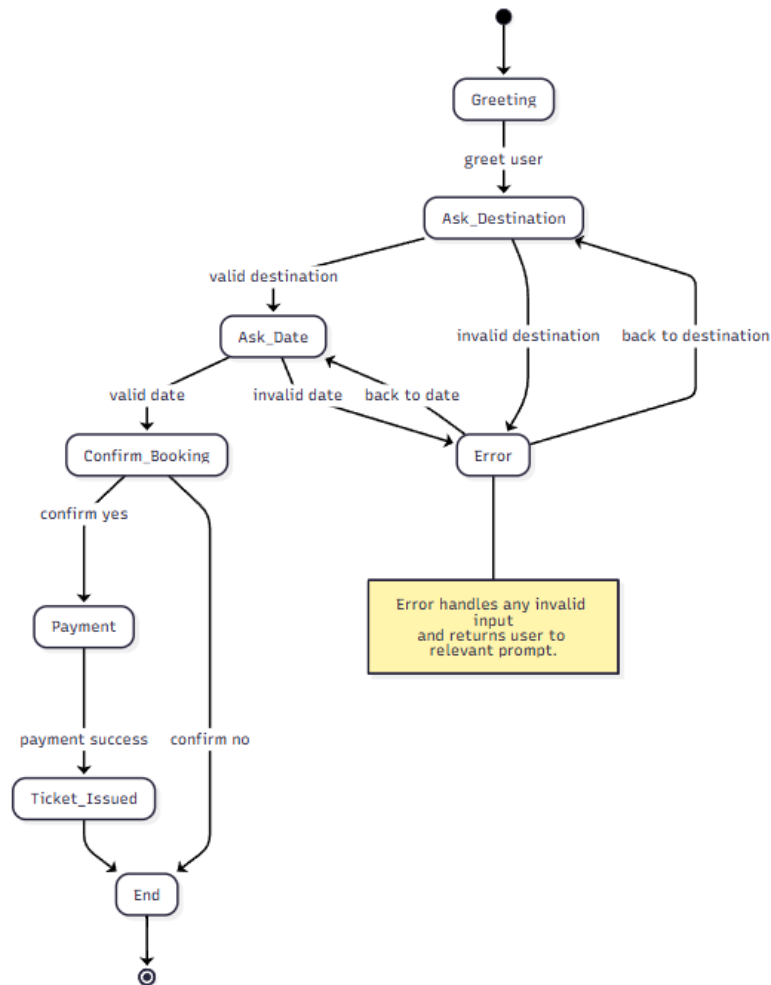


Figure 2: UML State Chart with initial/final nodes, guards, and error loops

Prototype Implementation (Python)

How to run:

```
python src/chatbot.py
```

Source Code:

Listing 1: Prototype code

```
1 def chatbot():
2     print("Chatbot: Hello! Welcome to Flight Ticket Booking
      System.")
3
4     # Ask destination
5     destination = input("Chatbot: Where would you like to travel?
      ")
6     if not destination.strip():
```

```

7         print("Chatbot: Invalid input! Please enter a valid
            destination.")
8         return
9
10    # Ask date
11    date = input("Chatbot: Please enter your travel date (DD/MM/
        YYYY): ")
12    if not date.strip():
13        print("Chatbot: Invalid input! Please enter a valid date.
            ")
14        return
15
16    # Confirm booking
17    confirm = input(f"Chatbot: You want to book a ticket to {
        destination} on {date}? (yes/no): ").lower()
18    if confirm != "yes":
19        print("Chatbot: Booking cancelled. Goodbye!")
20        return
21
22    # Payment
23    payment = input("Chatbot: Proceed with payment? (yes/no): ").
        lower()
24    if payment != "yes":
25        print("Chatbot: Payment cancelled. Goodbye!")
26        return
27
28    print("Chatbot: Payment successful! Your ticket has been
        booked.")
29    print("Chatbot: Thank you for using our system. Have a safe
        journey!")
30
31    # Run chatbot
32    if __name__ == "__main__":
33        chatbot()

```

Testing and Screenshots

```
Chatbot: Hello! Welcome to Flight Ticket Booking System.  
Chatbot: Where would you like to travel?  
New Delhi  
Chatbot: Please enter your travel date (DD/MM/YYYY):  
12/09/2025
```

Figure 3: Greeting and destination input

```
Chatbot: Hello! Welcome to Flight Ticket Booking System.  
Chatbot: Where would you like to travel?  
New Delhi  
Chatbot: Please enter your travel date (DD/MM/YYYY):  
12/09/2025  
Chatbot: You want to book a ticket to New Delhi on 12/09/2025? (yes/no):  
Yes  
Chatbot: Proceed with payment? (yes/no):  
 
```

Figure 4: Date entry and booking confirmation

```
Chatbot: Hello! Welcome to Flight Ticket Booking System.  
Chatbot: Where would you like to travel?  
New Delhi  
Chatbot: Please enter your travel date (DD/MM/YYYY):  
12/09/2025  
Chatbot: You want to book a ticket to New Delhi on 12/09/2025? (yes/no):  
Yes  
Chatbot: Proceed with payment? (yes/no):  
yes  
Chatbot: Payment successful! Your ticket has been booked.  
  
Chatbot: Thank you for using our system. Have a safe journey!
```

Figure 5: Payment and ticket issued

Results

The FSM and UML State Chart model the dialogue flow, including error handling and cancellation. The prototype traverses the designed states based on user input, producing expected outcomes.

Conclusion

We designed and implemented a dialogue system using FSM and UML State Chart techniques. The design can be extended with additional features like multiple passengers or payment options.