Practical 6

Development of State Transition Diagram for Software project definition

State Transition Diagram

A **State Transition Diagram** is a type of behavioral diagram in the Unified Modeling Language (UML) that represents the different states of a system or an object and how it transitions from one state to another due to various events. It is particularly useful for modeling the dynamic behavior of a system and understanding how it responds to inputs over time. Importance of Sequence Diagrams

Importance of State Transition Diagrams

- **Understanding System States**: Helps in visualizing the different states an object or system can be in and how it changes based on inputs.
- **Clarifying System Behavior**: Provides a clear representation of how a system responds to different events, ensuring a well-defined execution flow.
- **Enhancing Communication**: Helps stakeholders, developers, and designers understand system behavior, reducing ambiguities and misunderstandings.
- **Identifying Edge Cases**: Aids in discovering unexpected transitions or missing conditions, ensuring system reliability.
- **Supporting Software Development**: Acts as a reference for developers, allowing them to implement state-based logic more effectively.

Components of a State Transition Diagram

- **States**: Represent the various conditions an object or system can be in at any given time.
- **Transitions**: Arrows that depict how the system moves from one state to another due to events or conditions.
- **Events**: External or internal occurrences that trigger state changes.
- **Initial State**: The starting point of the system or object before any interaction occurs.
- **Final State**: The end condition of the system after completing all processes.
- **Guards/Conditions**: Logical expressions that must be true for a transition to take place.
- **Actions**: Tasks executed when transitioning from one state to another.

State Transition Diagram for Movie Booking Management System

A State Transition Diagram represents the various states a system goes through and how it transitions from one state to another based on user actions and system events. Below is a breakdown of the state transitions in an online movie ticket booking system.

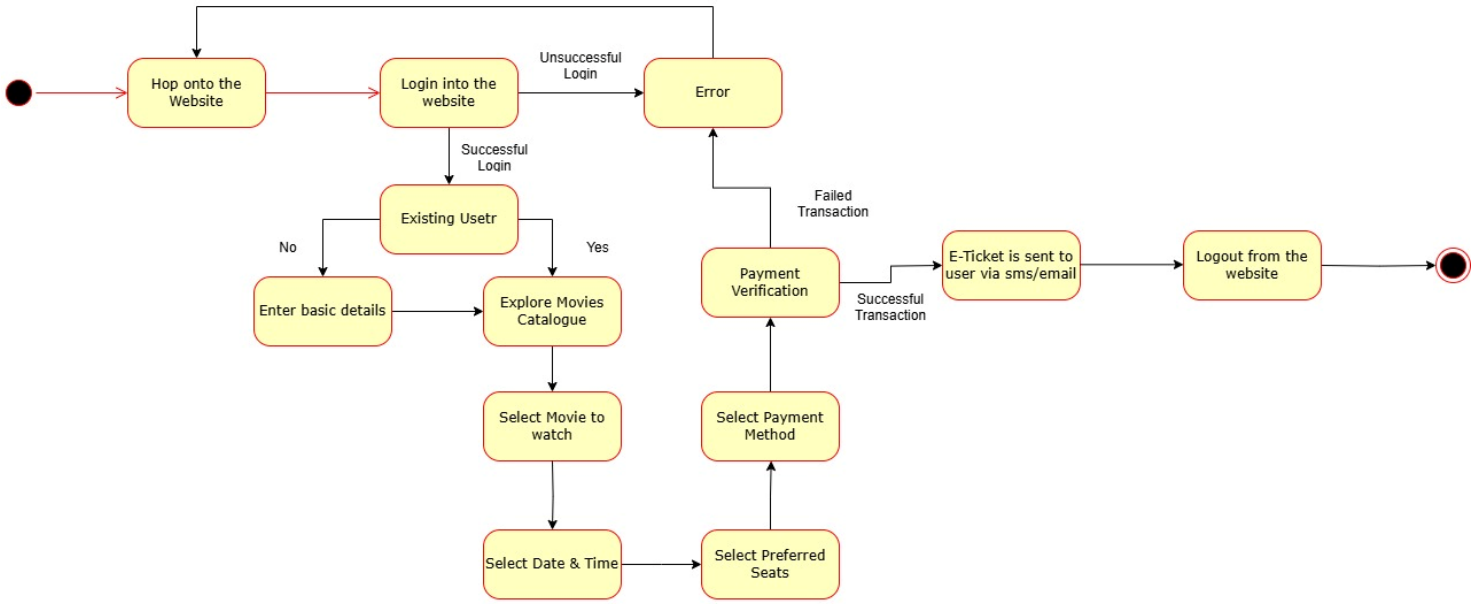
Process Breakdown:

1. **Hop onto the Website (Initial State)**
    - The user starts interacting with the movie booking system by visiting the website.
2. **Login into the Website**
    - The user enters login credentials.
    - If **login is successful**, the system moves to the **Existing User** state.
    - If **login fails**, the system transitions to an **Error** state.

3. **Existing User Check**
    - If the user is new, they must **enter basic details** before proceeding.
    - If they are an existing user, they can directly **explore the movie catalogue**.
4. **Movie Selection Process**
    - The user browses the movie catalogue.
    - Selects a **movie to watch**.
    - Selects a **date & time** for the show.
5. **Seat and Payment Selection**
    - The user selects **preferred seats**.
    - Chooses a **payment method**.
    - The system moves to **Payment Verification**.
6. **Payment Verification**
    - If the **payment fails**, the system moves to the **Error** state.
    - If the **payment is successful**, the system sends an **E-Ticket via SMS/Email**.
7. **Logout from the Website (Final State)**
    - Once the ticket is confirmed, the user can **log out** from the website, completing the process.

Error Handling

- Failed login attempts lead to an **Error** state.
- Payment failures return to the **Payment Verification** state for a retry or cancellation.



State Transition Diagram

.