Assignment 7

Title: Prepare a State Model for NBA Attainment System

Problem Statement:

- Identify States and events for the NBA Attainment System.
- Study state transitions and identify Guard conditions.
- Draw State chart diagram with advanced UML 2 notations. Implement the state model with a suitable object-oriented language for the NBA Attainment System.

Objective:

- To identify state transitions, events in the system flow.
- Draw a state diagram and implement a state diagram.

Theory:

A **state diagram** is used to represent the condition of the system or part of the system at finite instances of time. It's a **behavioral** diagram and it represents the behavior using finite state transitions. State diagrams are also referred to as **State machines** and **State-chart Diagrams**. These
terms are often used interchangeably. So simply, a state diagram is used to model the dynamic behavior of a class in response to time and changing external stimuli. We can say that each and every class has a state but we don't model every class using State diagrams. We prefer to model the states with three or more states.

Uses of statechart diagram –

- We use it to state the events responsible for change in state (we do not show what processes cause those events).
- We use it to model the dynamic behavior of the system .
- To understand the reaction of objects/classes to internal or external stimuli.

Basic components of a statechart diagram –

1. **Initial state –** We use a black filled circle to represent the initial state of a

System or a class.

Figure – initial state notation

2. **Transition –** We use a solid arrow to represent the transition or change of control from one state to another. The arrow is labelled with the event which causes the change in state.

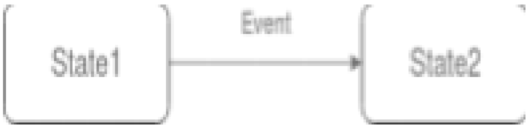


Figure – transition

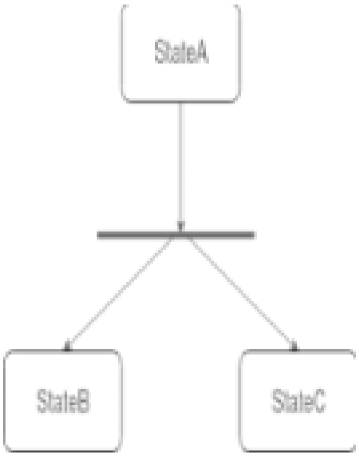
3. **State –** We use a rounded rectangle to represent a state. A state represents the conditions or circumstances of an object of a class at an instant of time.

Figure – state notation

These are the major states of NBA attainment system :

- CO Displayed
- PO Displayed
- CO-PO Displayed
- CO Updated
- PO Updated
- Target Attainment Displayed
- Target Attainment Updated
- Test Displayed
- Test Updated
- Marks Displayed
- Marks Updated
- Report Generated

- 4. **Fork –** We use a rounded solid rectangular bar to represent a Fork notation with incoming arrows from the parent state and outgoing arrows towards the newly created states. We use the fork notation to represent a state splitting



into two or more concurrent states.

Figure – a diagram using the fork notation

In the NBA Attainment system , the overall system ,manage marks state, manage test state and manage co-po state uses fork to split different states .

5. **Join –** We use a rounded solid rectangular bar to represent a Join notation with incoming arrows from the joining states and outgoing arrow towards the common goal state. We use the join notation when two or more states concurrently converge into one on the occurrence of an event or events.

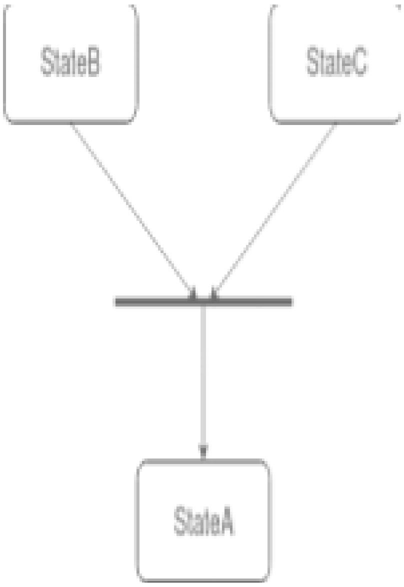


Figure – a diagram using join notation

In the NBA Attainment system , the overall system ,manage marks state, manage test state and manage co-po state uses ‘join’ to merge different states .

6. **Self transition –** We use a solid arrow pointing back to the state itself to represent a self transition. There might be scenarios when the state of the

object does not change upon the occurrence of an event. We use self transitions to represent such cases.

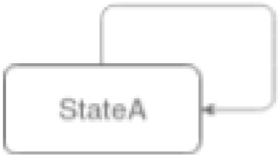


Figure – self transition notation

7. **Composite state –** We use a rounded rectangle to represent a composite state also.We represent a state with internal activities using a composite state.

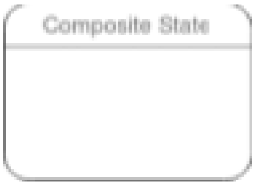


Figure – a state with internal activities

In login state of NBA attainment 2 composite states are used.One for the general login verification and other for detailed user verification.Both the state contains several other internal activities and states.

8. **Final state –** We use a filled circle within a circle notation to represent the final state in a state machine diagram.

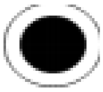


Figure – final state notation

Steps to draw a state diagram –

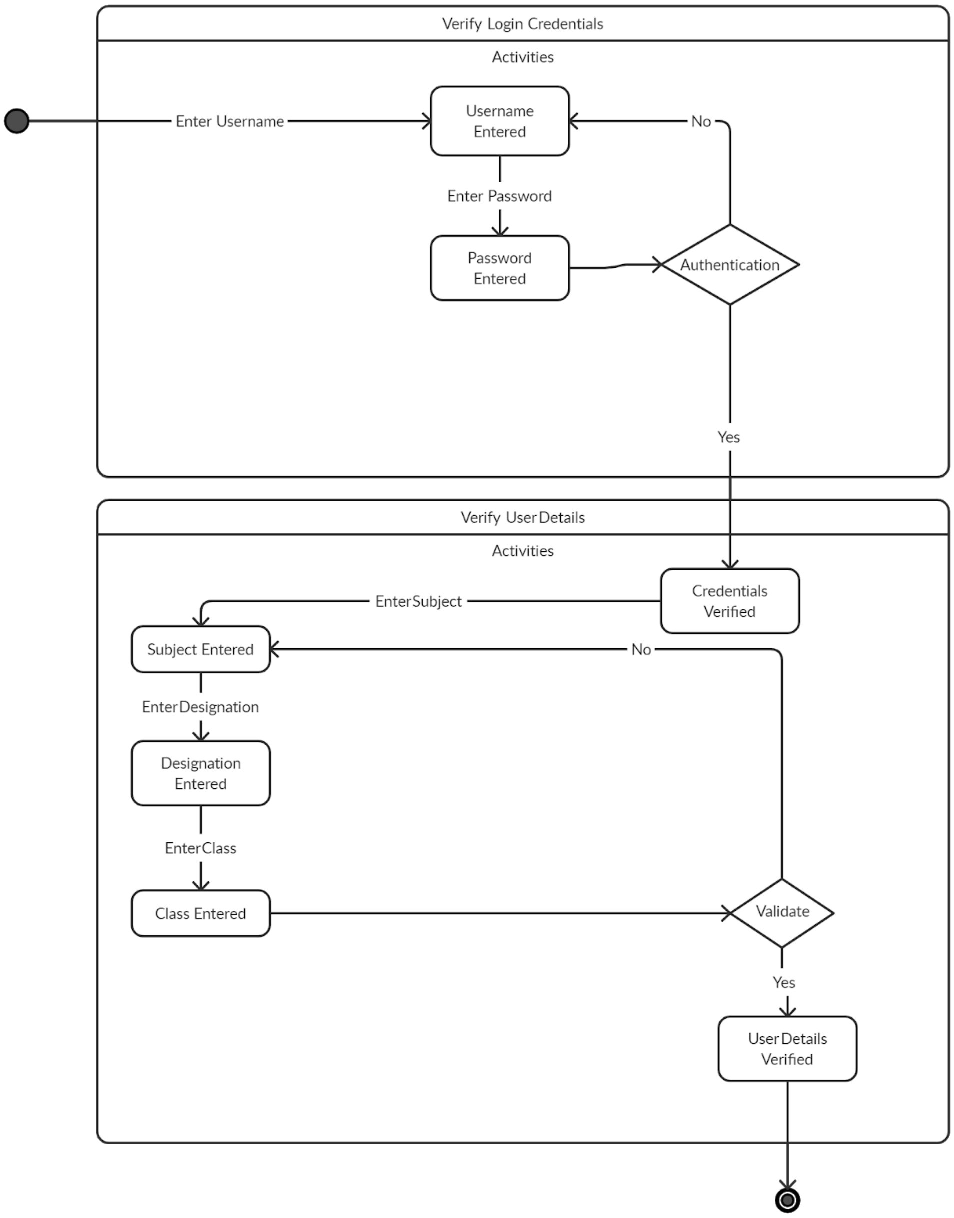
1. Identify the initial state and the final terminating states.
2. Identify the possible states in which the object can exist (boundary values corresponding to different attributes guide us in identifying different states).
3. Label the events which trigger these transitions.

STATE DIAGRAM for NBA Attainment System

State diagram 1 : Login

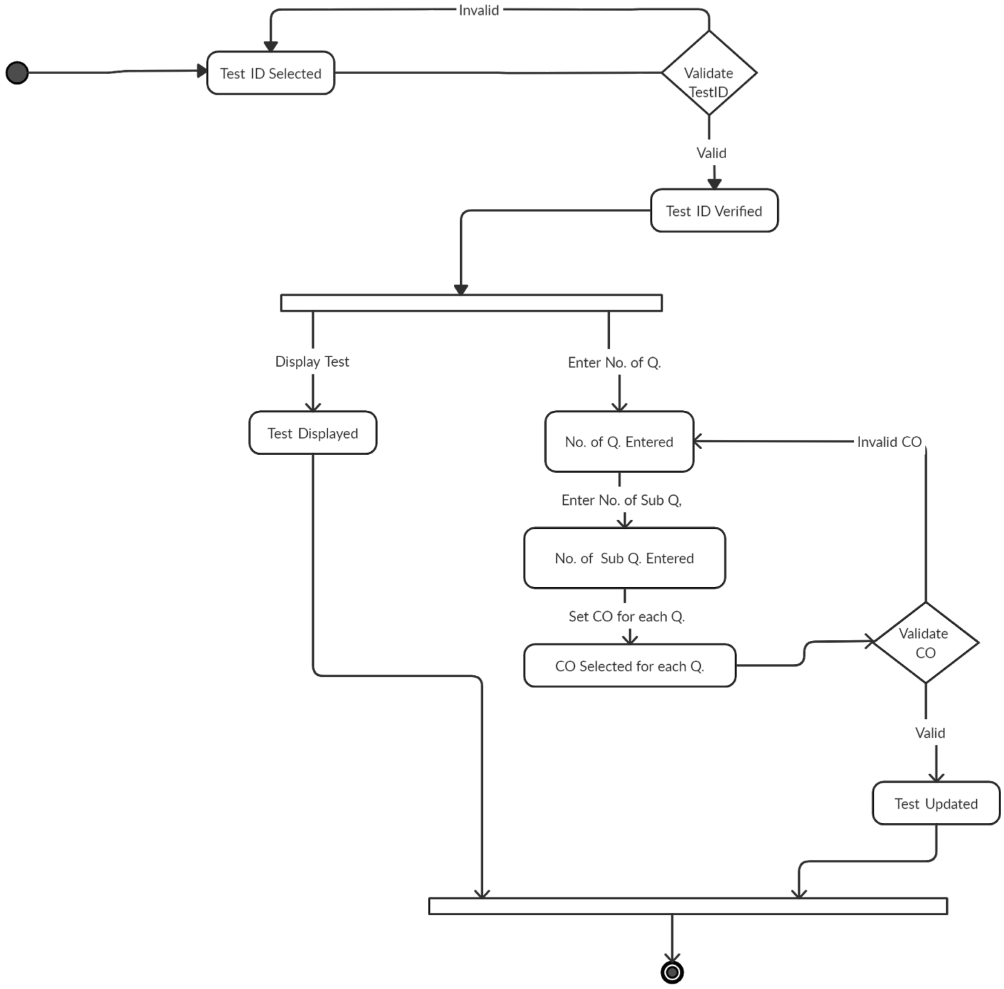
Basic state diagram components like initial state, state, transitions,final state are used in this diagram.

There are two composite states i.e. state with internal activities ‘verify login credentials’ and ‘verify user details’ in this diagram.



State diagram 2 : Set Test

Basic state diagram components like initial state, state, transitions,final state are used in this diagram.
We also use a fork notation to represent the state splitting. And we have also joined various states to converge them into new state.

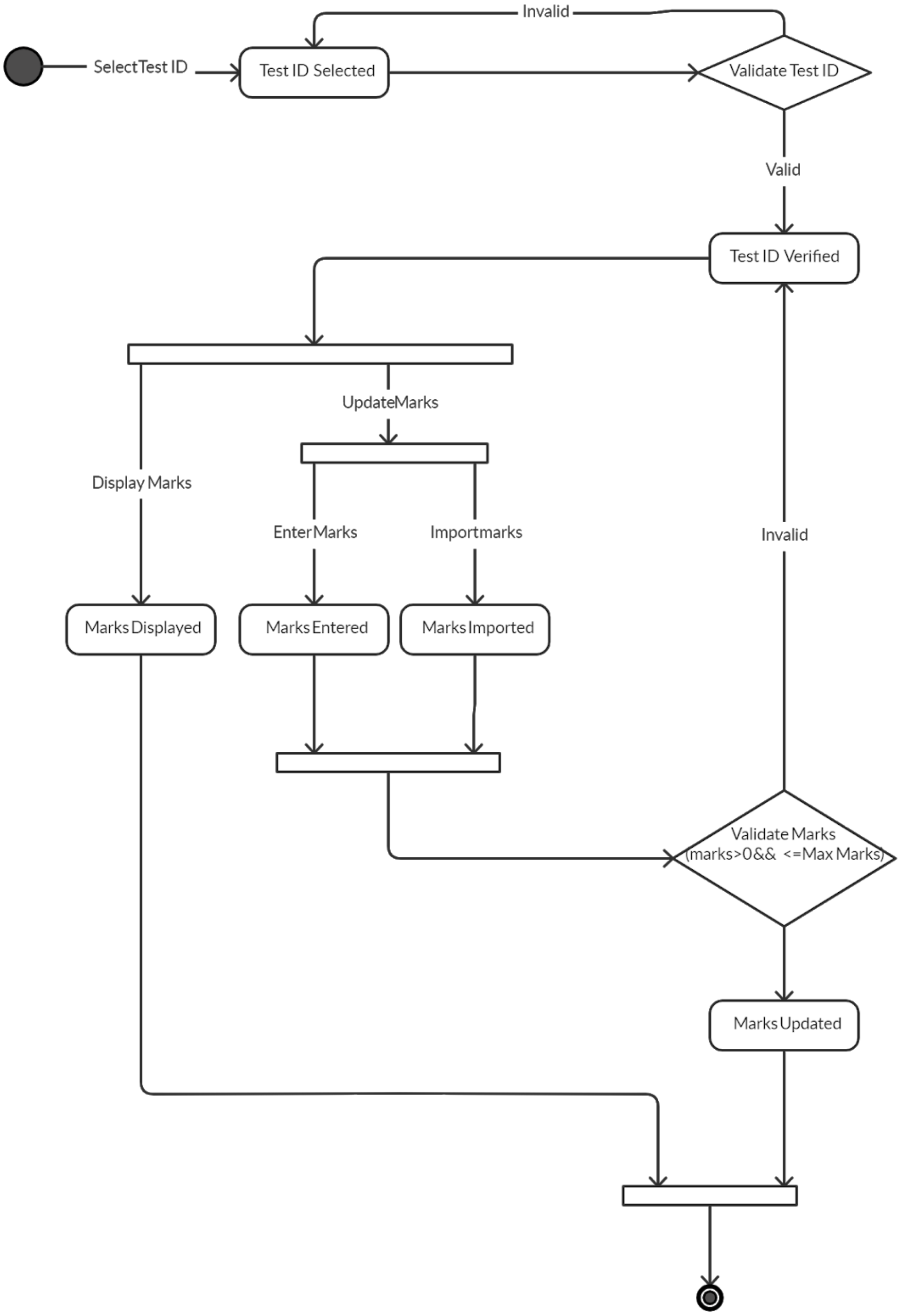


State diagram 3 : Enter Marks

Basic state diagram components like initial state, state, transitions,final state are used in this diagram.

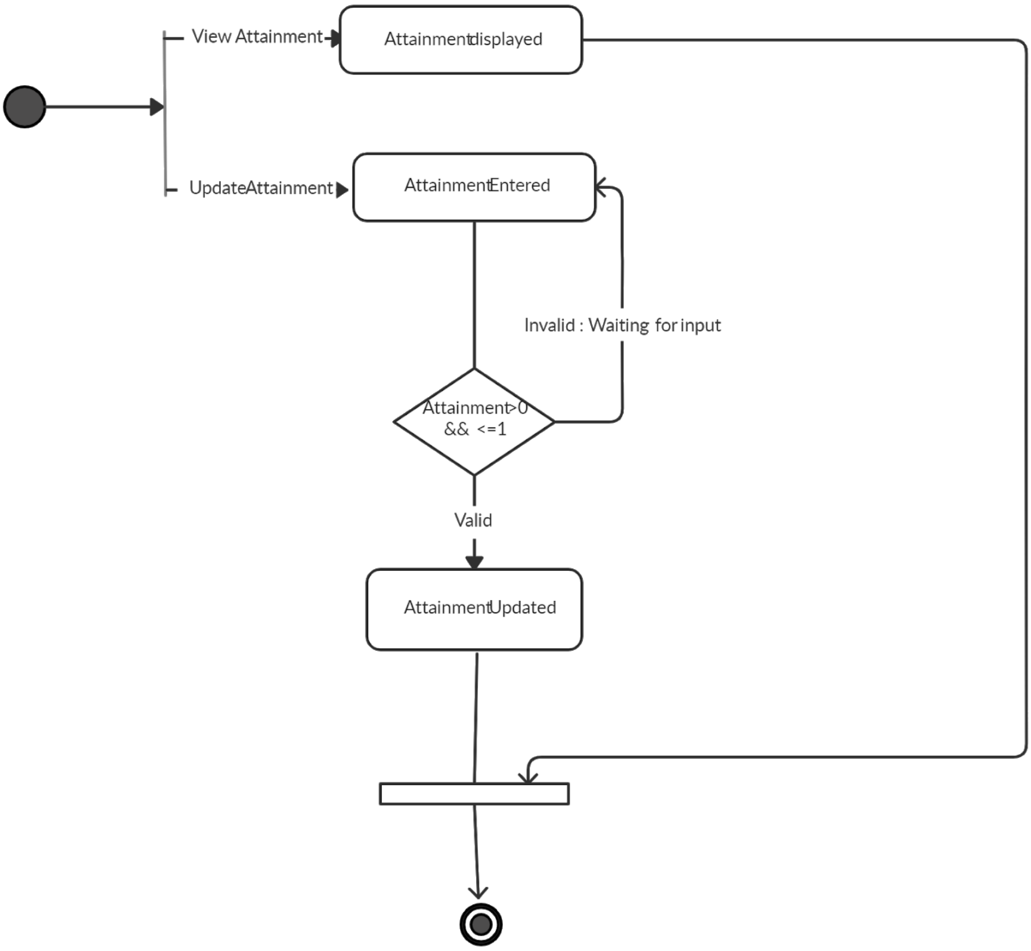
This diagram also uses a fork notation to represent a state splitting into 2 different states.

It also makes use of the join notation to converge 2 different states into a new state.



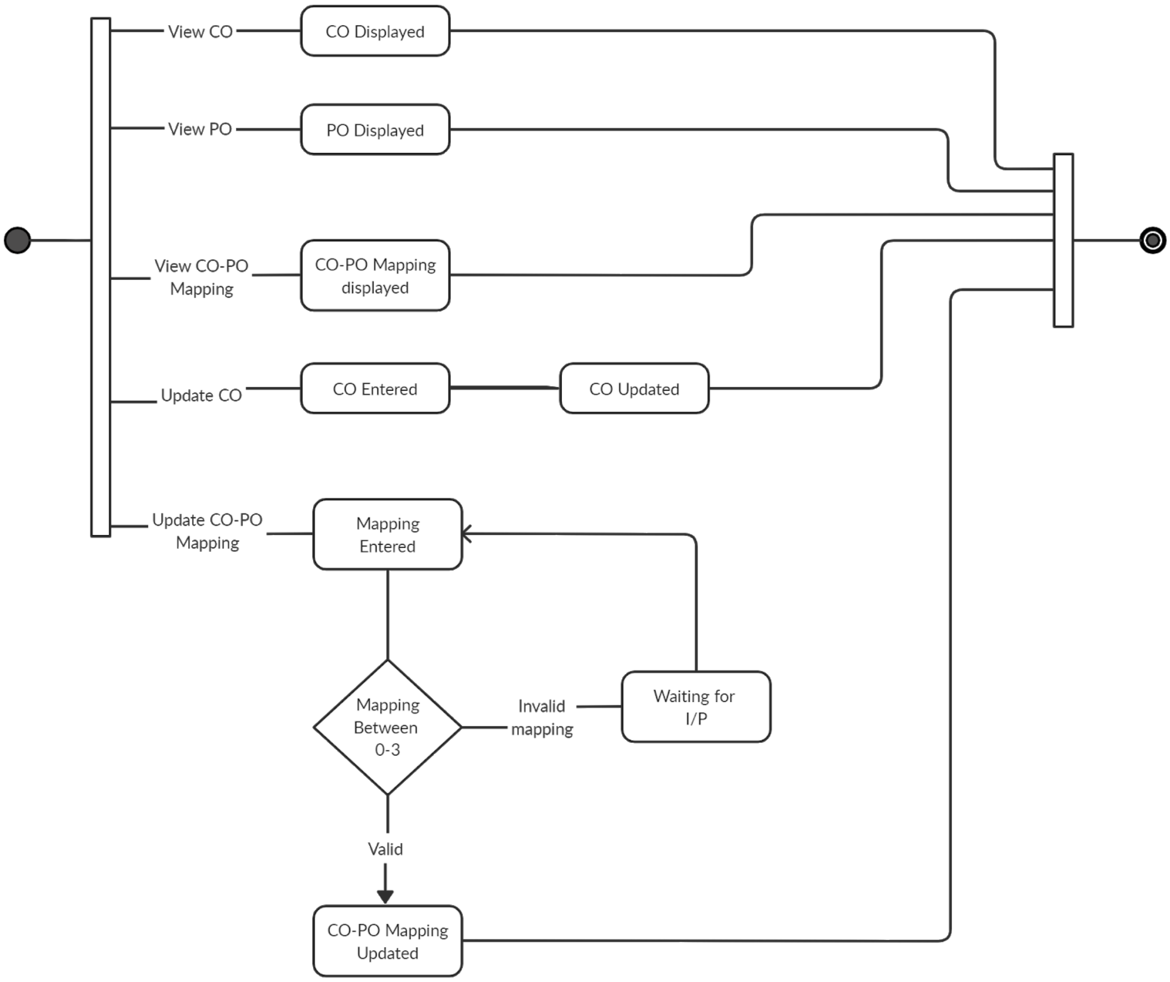
State diagram 4 : Set Target Attainment

Basic state diagram components like initial state, state, transitions,final state are used in this diagram.
This diagram also uses a fork notation to represent a state splitting into 2 different states.
It also makes use of the join notation to converge 2 different states into a new state.



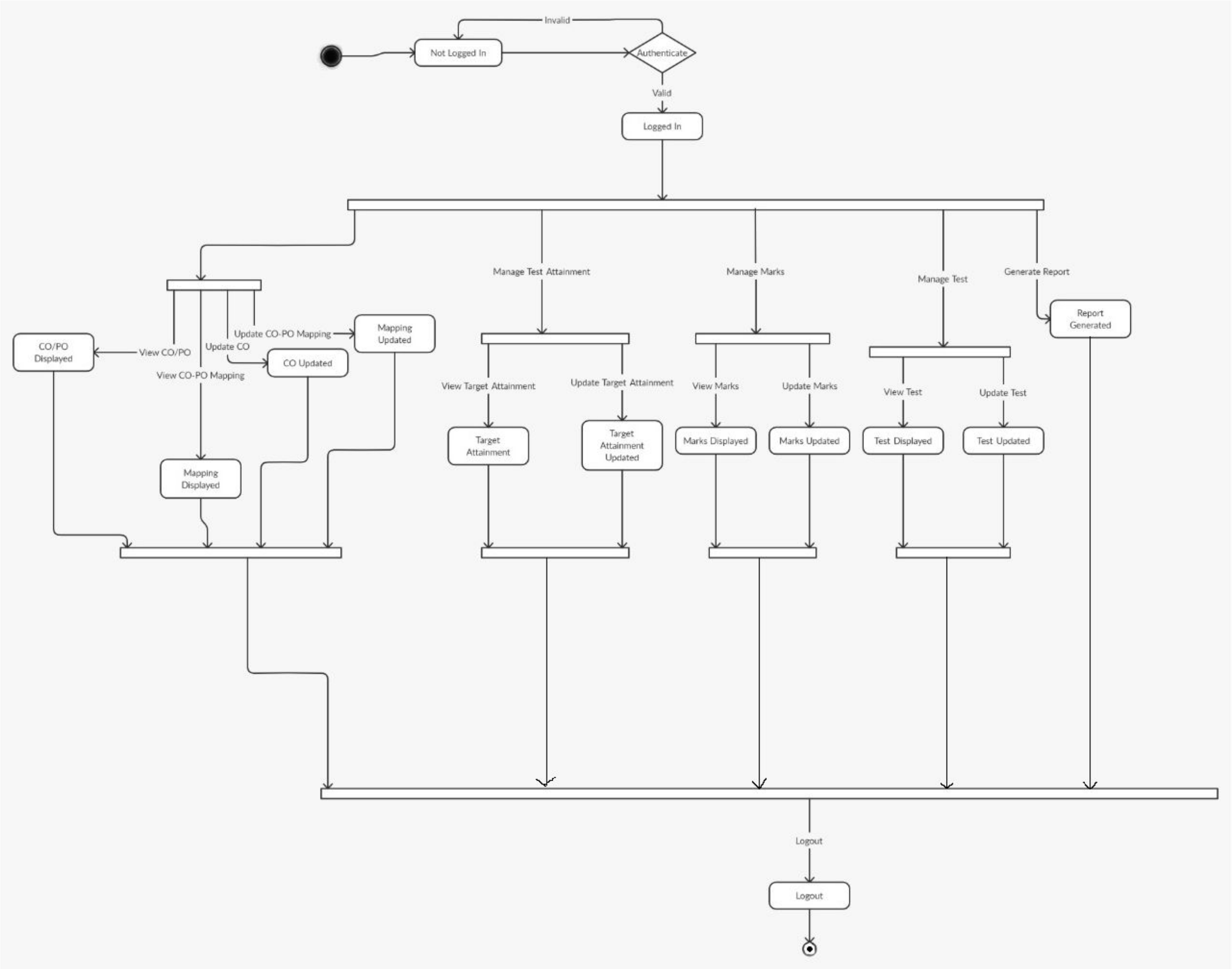
State diagram 5 : CO’s PO’s and Mapping

Basic state diagram components like initial state, state, transitions,final state are used in this diagram.
This diagram also uses a fork notation to represent a state splitting into 5 different states.
It also makes use of the join notation to converge 5 different states into a new state.



Overall System

Basic state diagram components like initial state, state, transitions,final state are used in this diagram.
This diagram also uses a fork notation to represent a state splitting into 2 different states and self transition also.
It also makes use of the join notation to converge 2 different states into a new state.



Conclusion :

Thus we identified the states transitions and events of the NBA Attainment System and implemented state chart diagrams for each of the events.