**Lab # 9**

**STATE TRANSITION DIAGRAM**

**Objectives:**

* Deeper understanding of UML state transition diagrams (STD).

1. **Outline**

* UML state diagrams.
* UML state diagram notation
* UML state details

1. **Background**

Mainly, we use interaction diagrams to study and model the behavior of objects in our

system. Sometimes, we need to study the behavior of a specific object that shows

complex behavior to better understand its dynamics. For that sake, UML provides

state transition diagrams used to model the behavior of objects of complex behavior.

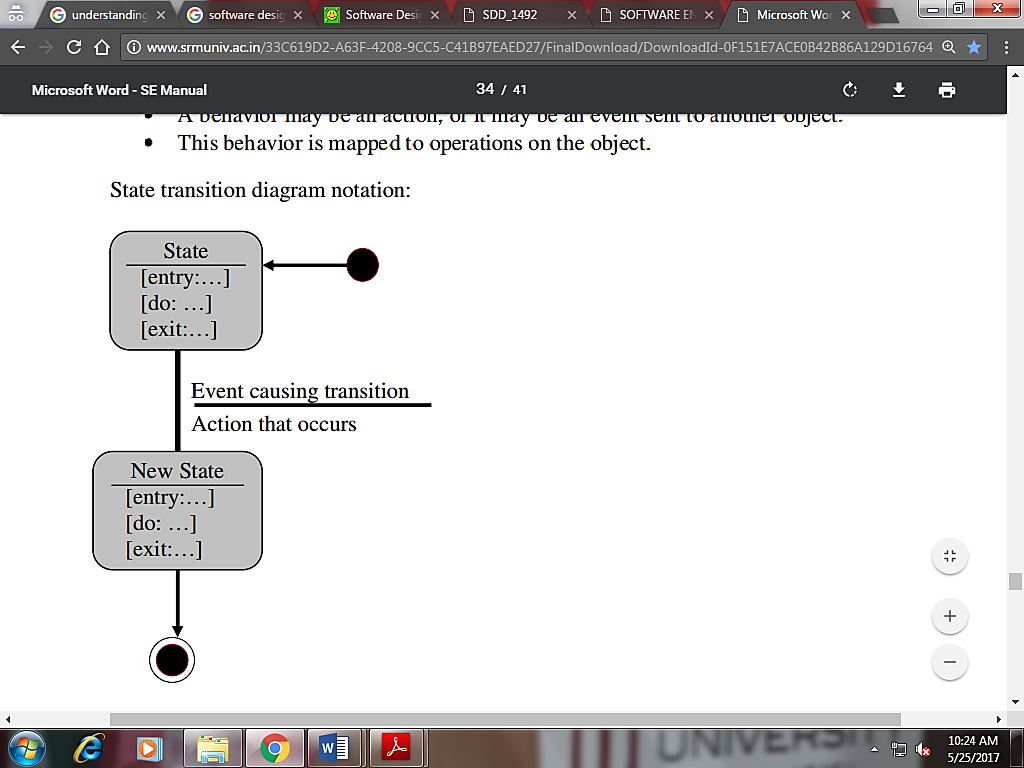
In this Lab, UML state transition diagrams will be introduced. We will study their

notation and how can we model them using Rational Rose.

* 1. **UML State Diagrams**

State diagrams show how one specific object changes state as it receives and processes messages:

* Since they are very specific, they are used for analyzing very specific situations if we compare them with other diagrams.
* A state refers to the set of values that describe an object at a specific moment in time.
* As messages are received, the operations associated with the object’s parent class are invoked to deal with the messages.
* These messages change the values of these attributes.
* There is no need to prepare a state diagram for every class you have in the system.
  1. **Creating State Transition Diagrams**
* States are represented by rectangles with rounded corners with an attribute name with a values associated with it.
* The name of the state is placed within the box.
* Events are shown by arrows.
* An event occurs when at an instant in time when a value is changed.
* A message is data passed from one object to another.
* The name of a state usually refers to the name of the attribute and the values associated to it.
* Example, a student object may receive a message to change its name. The state of that object changes from the first name state to the new state name.
* The name of the state is placed in the top compartment.
* State variables are placed in the next compartment.
* The operations associated with the state are listed in the lowest compartment of the state box.
* In the operations part, we usually use one of the following reserved words:
* Entry: a specific action performed on the entry to the state.
* Do: an ongoing action performed while in the state.
* On: a specific action performed while in the state.
* Exit: a specific action performed on exiting the state.
* There are two special states added to the state transition diagram- start state and end state.
* Notation of start state is a solid black circle and for the end state a bull’s eye is used.
  1. **State Transition Details**
* A state transition may have an action and/or guard condition associated with it and it may also trigger an event.
* An action is the behavior that occurs when the state transition occurs.
* An event is a message that is sent to another object in the system.
* A guard condition is a Boolean expression of attribute values that allows a state transition only if the condition is true.
* Both actions and guards are behaviors of the object and typically become operations. Also they are usually private operations (used by the object itself)
* Actions that accompany all state transitions into a state may be placed as an entry action within the state.
* Actions that accompany all state transitions out of a state may be placed as exit actions within the state
* A behavior that occurs within the state is called an activity.
* An activity starts when the state is entered and either completes or is interrupted by an outgoing state transition.
* A behavior may be an action, or it may be an event sent to another object.
* This behavior is mapped to operations on the object.



**Lab Task:**

1. Draw State Diagram for ATM machine
2. Draw State Diagram for Library Management