EXPERIMENT - 6

Software Engineering Lab

Aim

To draw the behavioral view diagram: State-chart diagram, Activity diagram.

Syeda Reeha Quasar
14114802719
4C7

EXPERIMENT – 6

Aim:

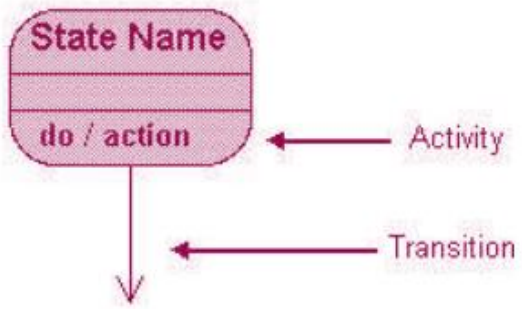
To draw the behavioural view diagram: State-chart diagram, Activity diagram.

Theory:

behavioral diagrams visualize, specify, construct, and document the dynamic aspects of a system. The behavioral diagrams are categorized as follows: use case diagrams, interaction diagrams, state–chart diagrams, and activity diagrams.

State–Chart Diagrams

State chart diagrams represent the behavior of entities capable of dynamic behavior by specifying its response to the receipt of event instances.



A state–chart diagram shows a state machine that depicts the control flow of an object from one state to another. A state machine portrays the sequences of states which an object undergoes due to events and their responses to events.

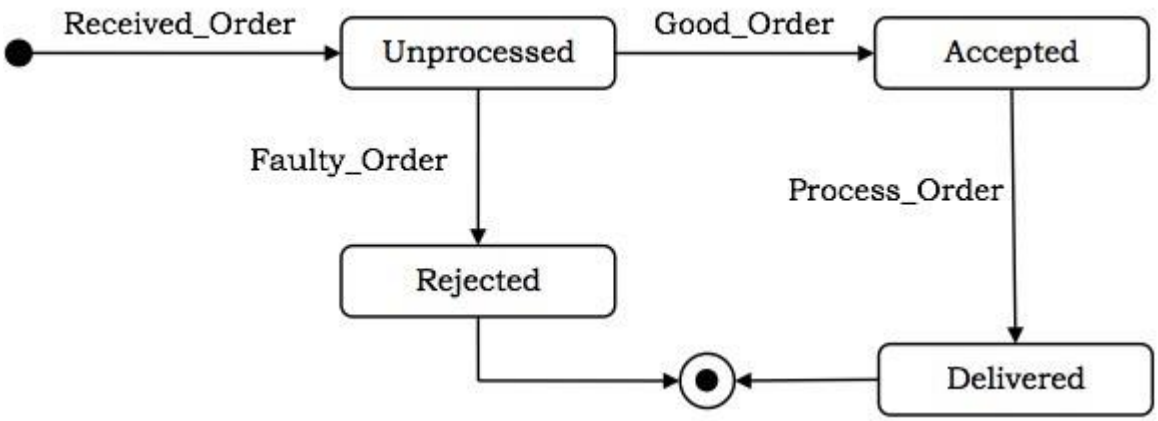
State–Chart Diagrams comprise of −

- States: Simple or Composite
- Transitions between states
- Events causing transitions
- Actions due to the events

State-chart diagrams are used for modeling objects which are reactive in nature.

Example

In the Automated Trading House System, let us model Order as an object and trace its sequence. The following figure shows the corresponding state–chart diagram.



Activity Diagrams

An activity diagram depicts the flow of activities which are ongoing non-atomic operations in a state machine. Activities result in actions which are atomic operations.

Activity diagrams describe the workflow behavior of a system. Activity diagrams are similar to state diagrams because activities are the state of doing something. The diagrams describe the state of activities by showing the sequence of activities performed. Activity diagrams can show activities that are conditional or parallel. Activity diagrams show the flow of activities through the system. Diagrams are read from top to bottom and have branches and forks to describe conditions and parallel activities. A fork is used when multiple activities are occurring at the same time. The diagram below shows a fork after activity1. This indicates that both activity2 and activity3 are occurring at the same time. After activity2 there is a branch. The branch describes what activities will take place based on a set of conditions. All branches at some point are followed by a merge to indicate the end of the conditional behavior started by that branch. After the merge all of the parallel activities must be combined by a join before transitioning into the final activity state.

Activity diagrams comprise of −

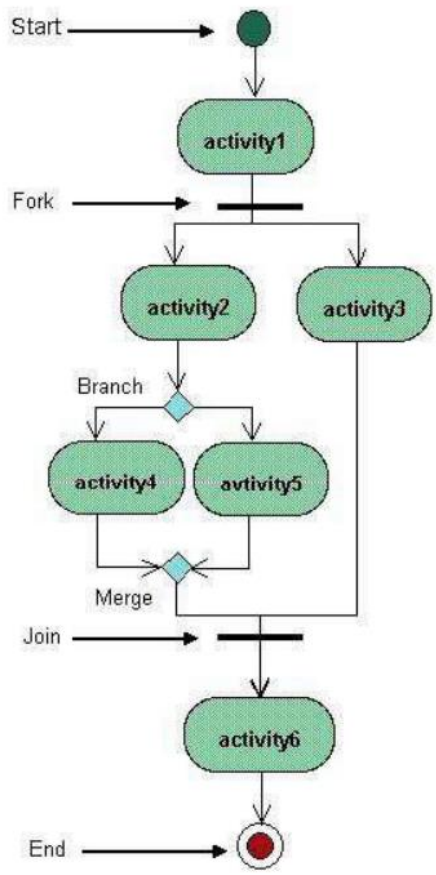
- Activity states and action states
- Transitions
- Objects

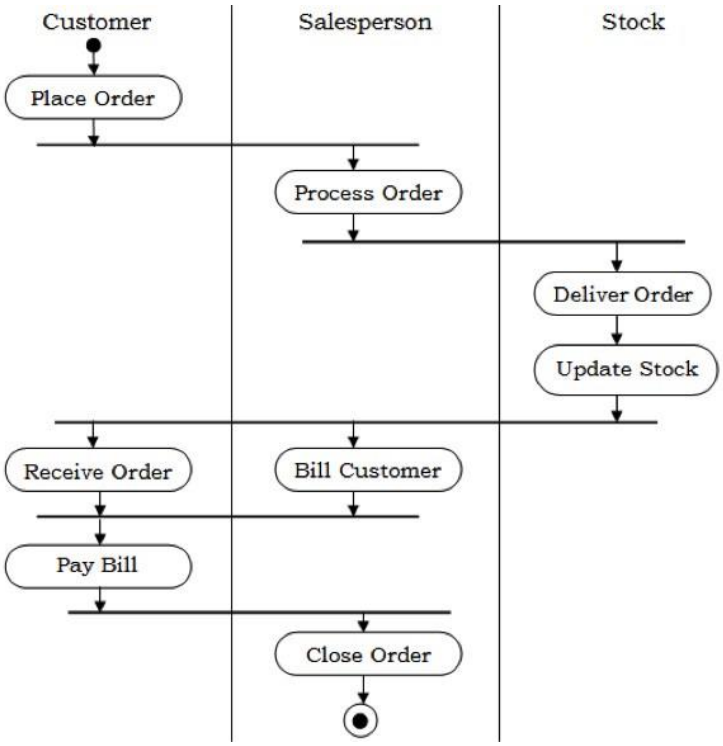
Activity diagrams are used for modeling −

- workflows as viewed by actors, interacting with the system.
- details of operations or computations using flowcharts.

Example

The following figure shows an activity diagram of a portion of the Automated Trading House System.





Performance Instruction:

To draw state chart diagram

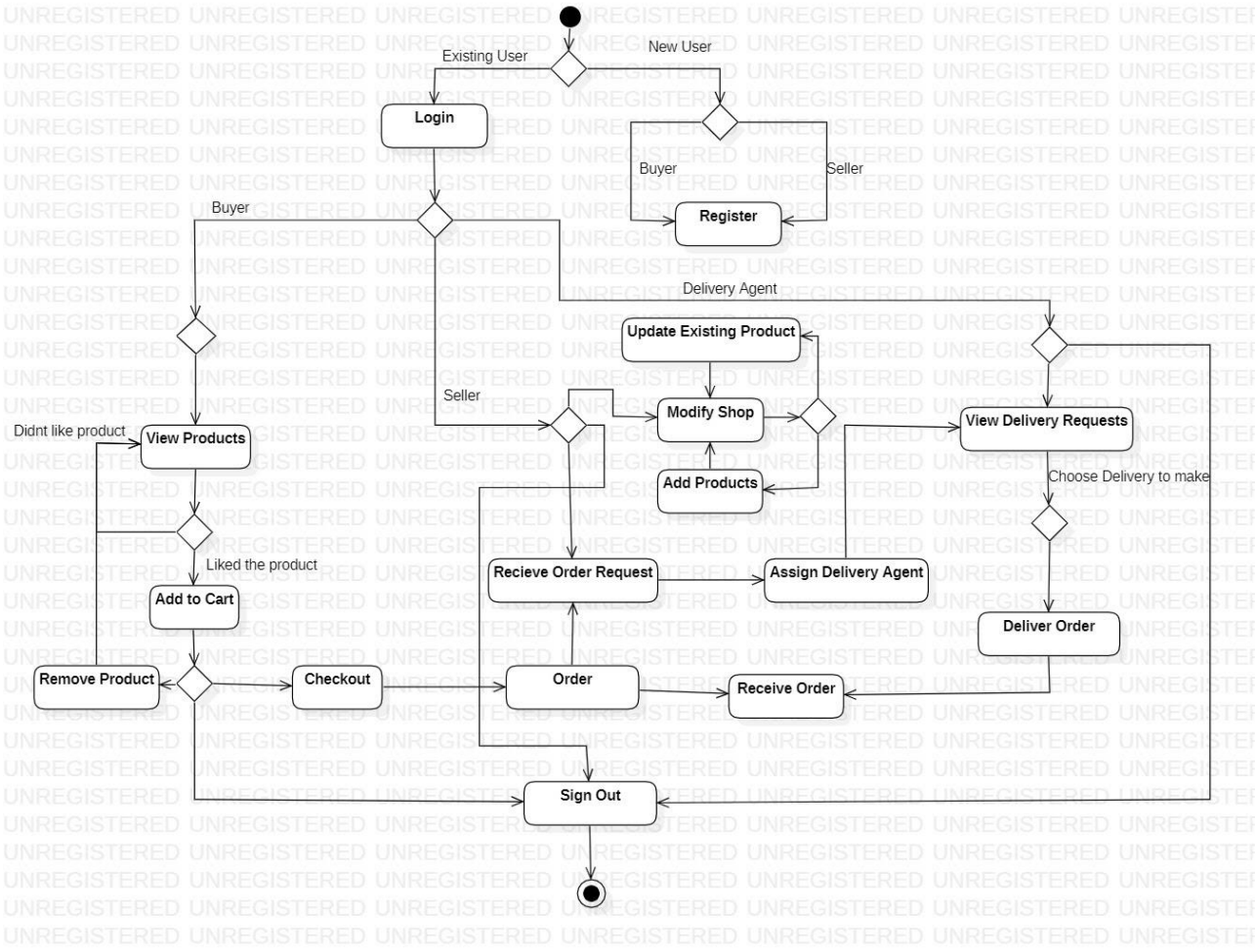
1. Identify various elements states and their different transition of the state-chart diagram.
2. Draw the state-chart diagram as per the norms.

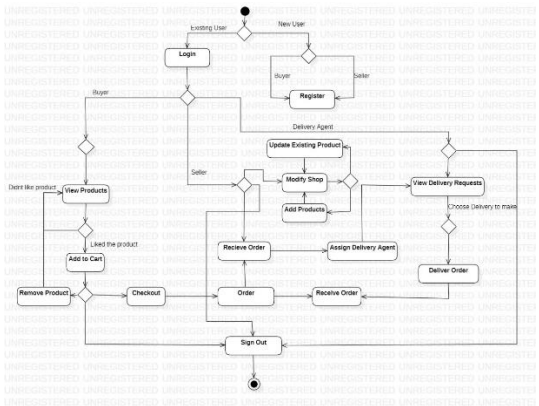
To draw activity diagram

1. Identify various elements such as different activity their boundaries etc. of the activity diagram.
2. Draw the activity diagram as per the norms.

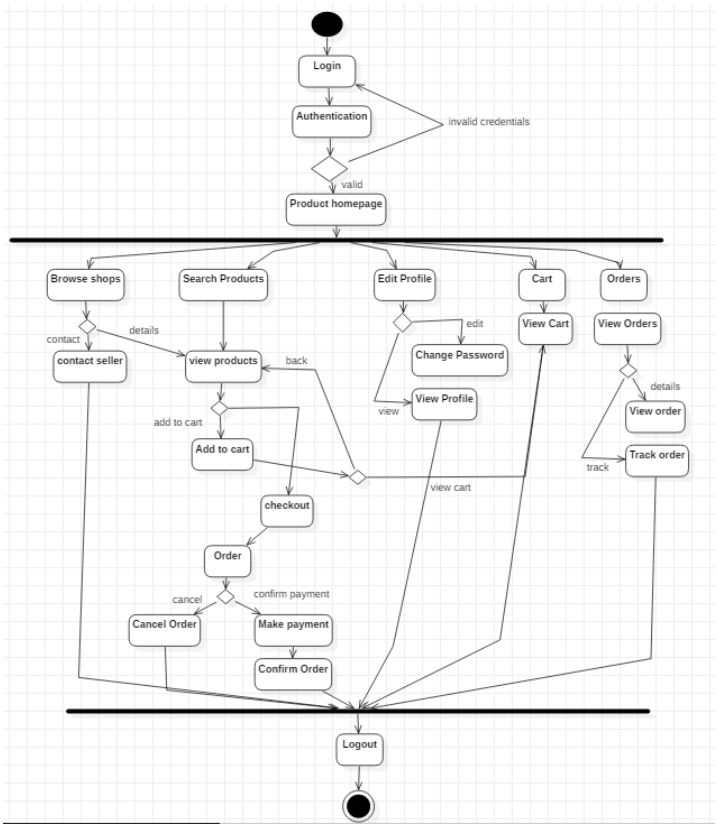
Output:

State Chart Diagram

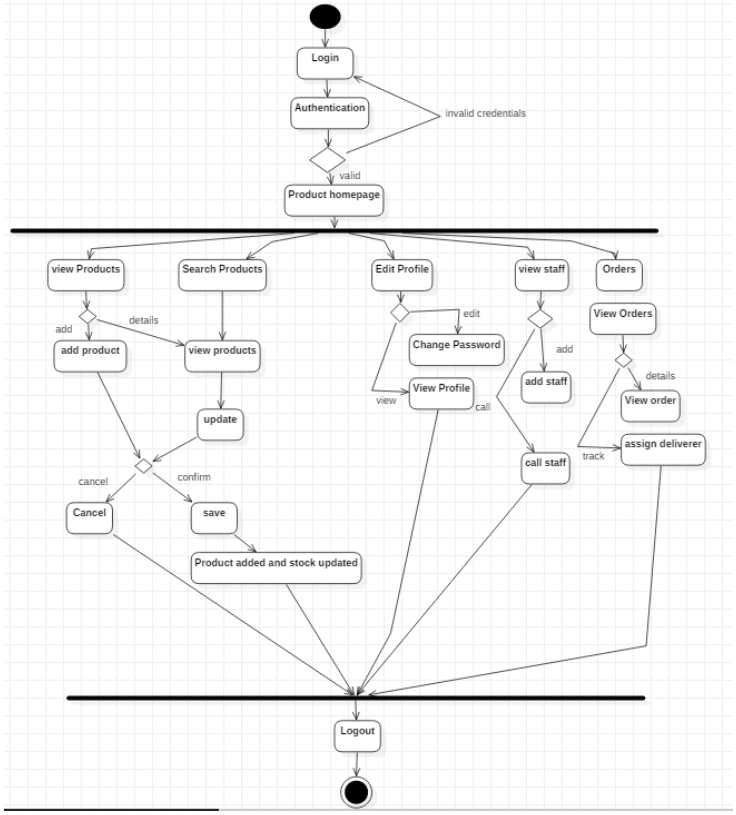




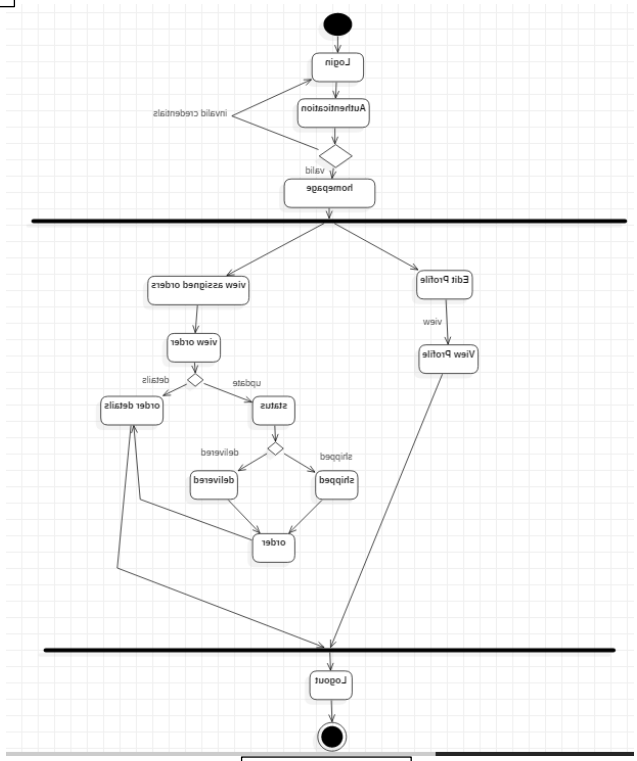
Activity Diagram



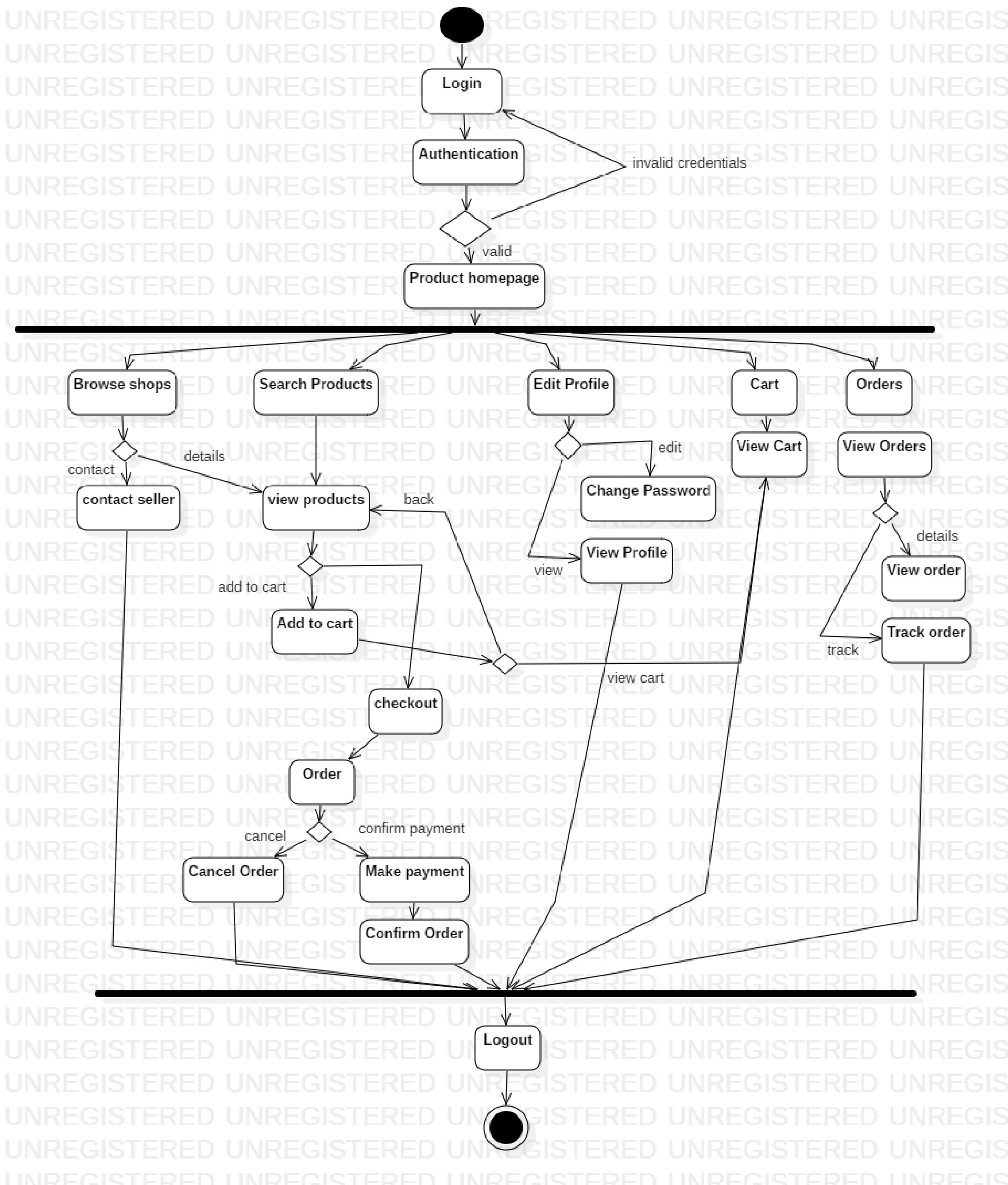
Buyer



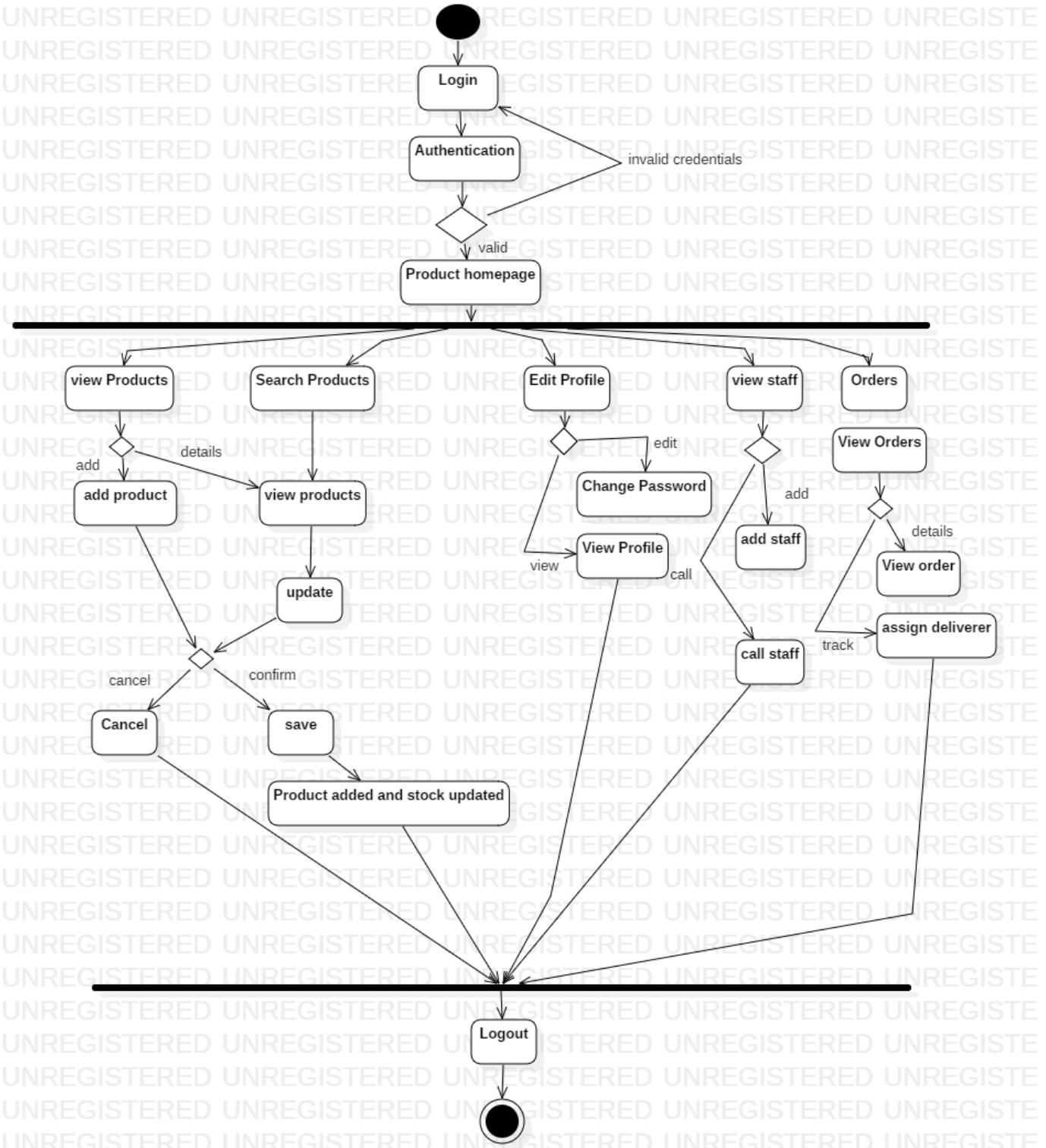
Seller



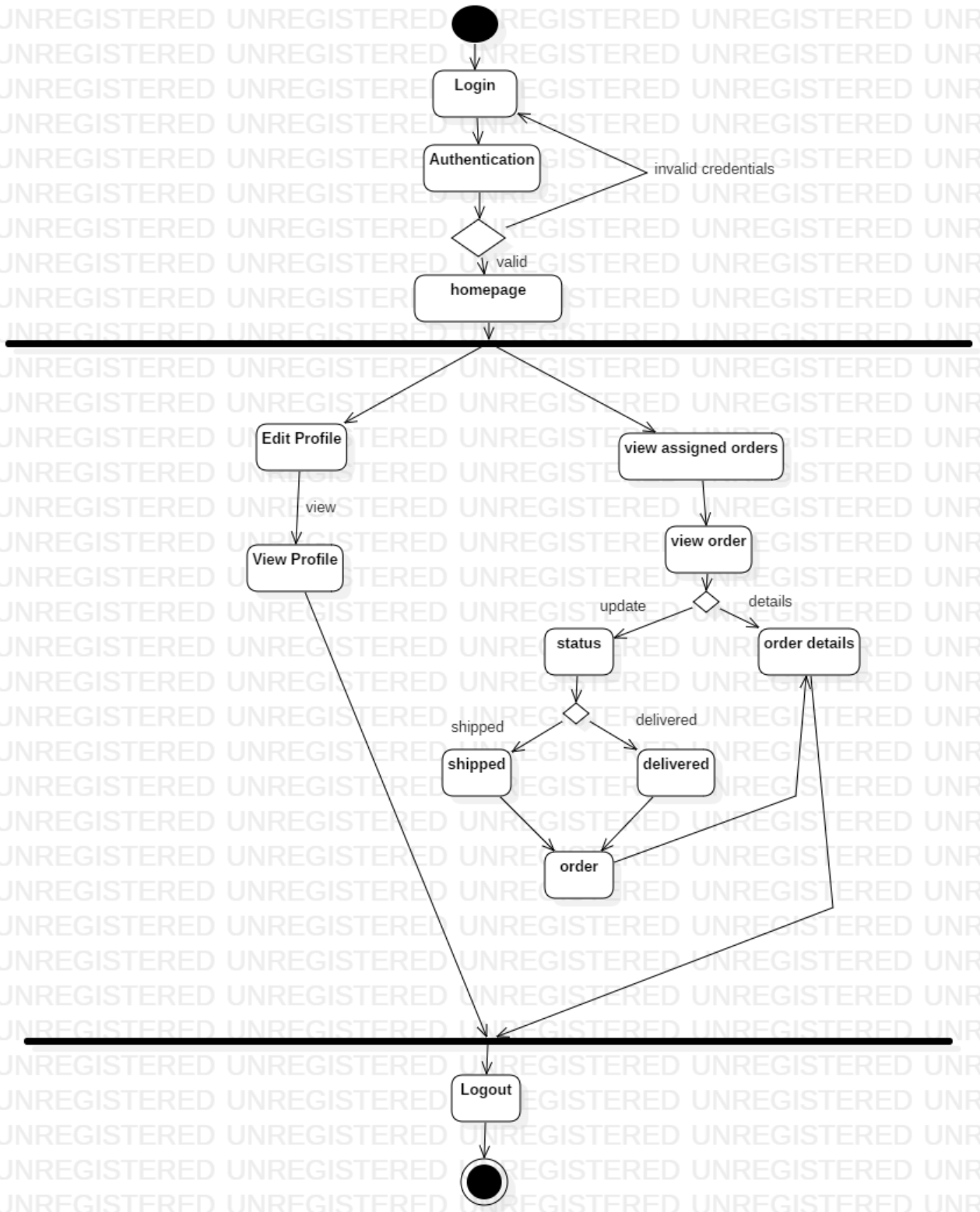
Deliverer


Login
Authentication
invalid credentials
valid
Product homepage
Browse shops
Search Products
Edit Profile
Cart
Orders
details
contact
contact seller
view products
back
edit
Change Password
View Cart
View Orders
View Profile
view
details
View order
add to cart
Add to cart
Track order
track
view cart
checkout
Order
cancel
confirm payment
Cancel Order
Make payment
Confirm Order
Logout

Buyer


Login
Authentication
invalid credentials
valid
Product homepage
view Products
Search Products
Edit Profile
view staff
Orders
add
details
edit
View Orders
add product
view products
Change Password
add
add staff
details
update
view
View Profile
call
View order
assign deliverer
track
call staff
cancel
confirm
Cancel
save
Product added and stock updated
Logout

Seller


Login
Authentication
invalid credentials
valid
homepage
Edit Profile
view assigned orders
view
View Profile
view order
update
details
status
order details
shipped
delivered
shipped
delivered
order
Logout

Deliverer

Conclusion:

State Chart and Activity diagram were made successfully by following above steps.

Viva Questions

1. How activity diagram explains behavioral view of system?

Ans.

An activity diagram is a behavioral diagram i.e. it depicts the behavior of a system. An activity diagram portrays the control flow from a start point to a finish point showing the various decision paths that exist while the activity is being executed.

2. Explain steps followed to construct activity diagram?

Ans.

Step 1: Figure out the action steps from the use case

Here you need to identify the various activities and actions your business process or system is made up of.

Step 2: Identify the actors who are involved

If you already have figured out who the actors are, then it's easier to discern each action they are responsible for.

Step 3: Find a flow among the activities

Figure out in which order the actions are processed. Mark down the conditions that have to be met in order to carry out certain processes, which actions occur at the same time and whether you need to add any branches in the diagram. And do you have to complete some actions before you can proceed to others?

Step 4: Add swimlanes

You have already figured out who is responsible for each action. Now it's time to assign them a swimlane and group each action they are responsible for under them.

3. How state chart diagram explains behavioral view of system?

Ans.

Statechart diagram defines the states of a component and these state changes are dynamic in nature. Its specific purpose is to define the state changes triggered by events. Events are internal or external factors influencing the system.

4. Explain steps followed to construct state chart diagram?

Ans.

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.

5. Explain symbols used to construct these diagrams?

Ans.

STATE CHART DIAGRAM

States
States represent situations during the life of an object. You can easily illustrate a state in SmartDraw by using a rectangle with rounded corners.

A simple state

A state with internal activities

Transition

A solid arrow represents the path between different states of an object. Label the transition with the event that triggered it and the action that results from it. A state can have a transition that points back to itself.

Transition

Initial State

A filled circle followed by an arrow represents the object's initial state.

Initial state

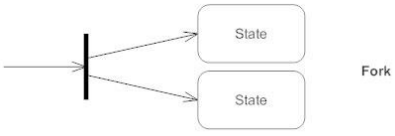
Final State

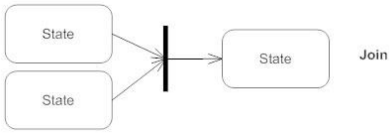
An arrow pointing to a filled circle nested inside another circle represents the object's final state.

Final state

Synchronization and Splitting of Control

A short heavy bar with two transitions entering it represents a synchronization of control. The first bar is often called a fork where a single transition splits into concurrent multiple transitions. The second bar is called a join, where the concurrent transitions reduce back to one.

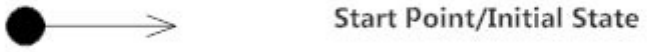




ACTIVITY DIAGRAM

Initial State or Start Point

A small filled circle followed by an arrow represents the initial action state or the start point for any activity diagram. For activity diagram using swimlanes, make sure the start point is placed in the top left corner of the first column.



Activity or Action State

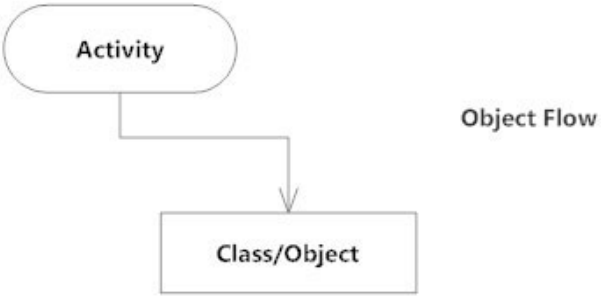
An action state represents the non-interruptible action of objects. You can draw an action state in SmartDraw using a rectangle with rounded corners.

Action Flow

Action flows, also called edges and paths, illustrate the transitions from one action state to another. They are usually drawn with an arrowed line.

Object Flow

Object flow refers to the creation and modification of objects by activities. An object flow arrow from an action to an object means that the action creates or influences the object. An object flow arrow from an object to an action indicates that the action state uses the object.



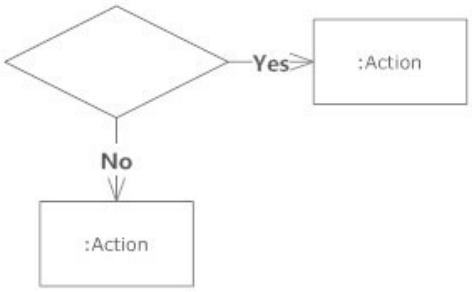
Decisions and Branching

A diamond represents a decision with alternate paths. When an activity requires a decision prior to moving on to the next activity, add a diamond between the two activities. The outgoing alternates should be labeled with a condition or guard expression. You can also label one of the paths "else."

Decision Symbol

Guards

In UML, guards are a statement written next to a decision diamond that must be true before moving next to the next activity. These are not essential, but are useful when a specific answer, such as "Yes, three labels are printed," is needed before moving forward.



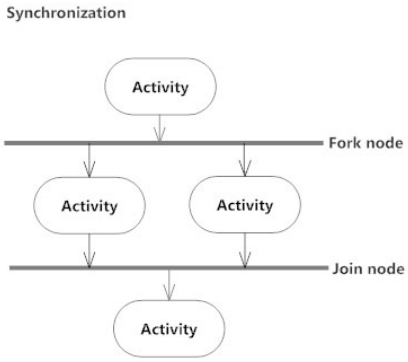
Guard Symbols

Synchronization

A fork node is used to split a single incoming flow into multiple concurrent flows. It is represented as a straight, slightly thicker line in an activity diagram.

A join node joins multiple concurrent flows back into a single outgoing flow.

A fork and join mode used together are often referred to as synchronization.

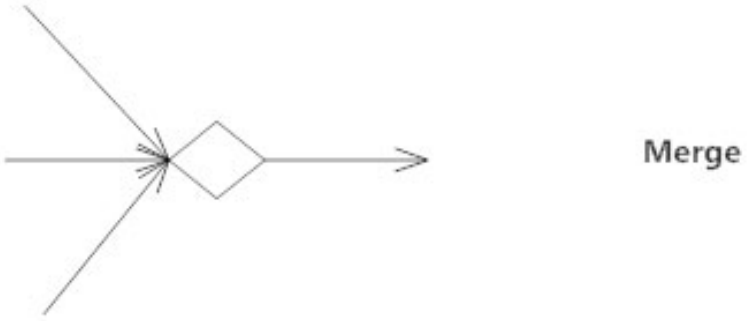


Time Event

This refers to an event that stops the flow for a time; an hourglass depicts it.

Merge Event

A merge event brings together multiple flows that are not concurrent.

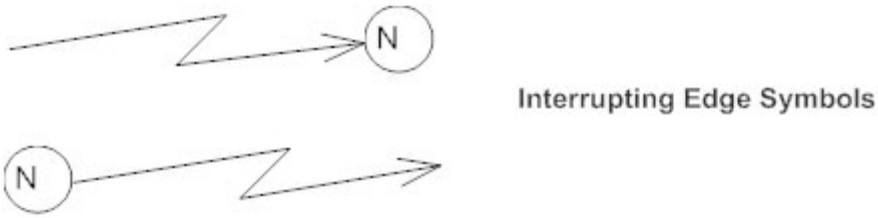


Sent and Received Signals

Signals represent how activities can be modified from outside the system. They usually appear in pairs of sent and received signals, because the state can't change until a response is received, much like synchronous messages in a sequence diagram. For example, an authorization of payment is needed before an order can be completed.

Interrupting Edge

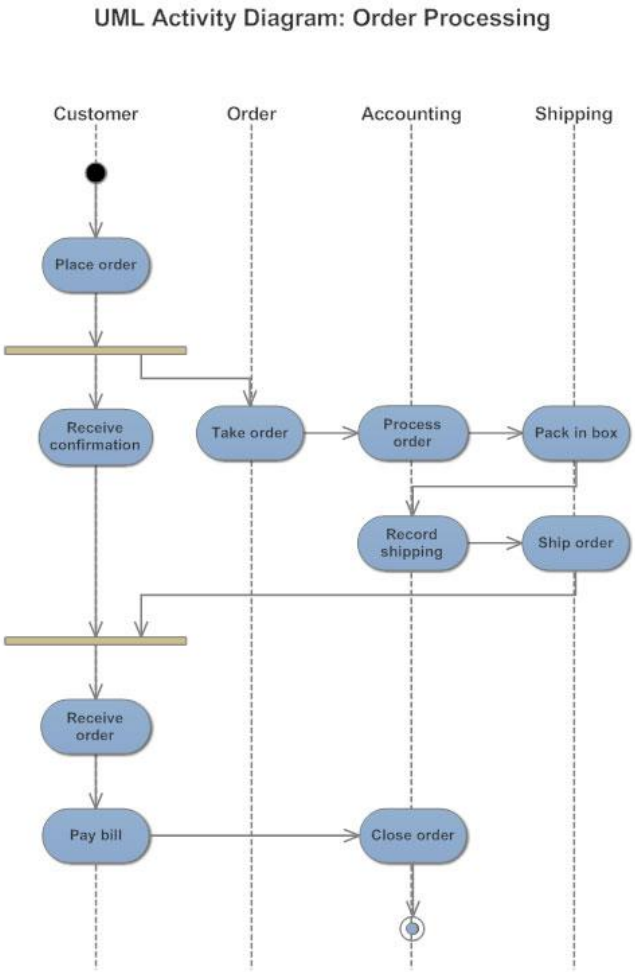
An event, such as a cancellation, that interrupts the flow denoted with a lightning bolt.



Interrupting Edge Symbols

Swimlanes

Swimlanes group related activities into one column.



Final State or End Point

An arrow pointing to a filled circle nested inside another circle represents the final action state.

End Point Symbol