Activity diagram

Activity diagram is another important behavioral diagram in UML diagram to describe dynamic aspects of the system. Activity Diagrams describe how activities are coordinated to provide service. It is also suitable for modeling how a collection of use cases coordinate or how the events in a single use case relate to one another, in particular, use cases where activities may overlap and require coordination (Model workflows between/within use cases and Identify pre- and post-conditions for use cases)

Activity diagrams show the steps involved in how a system works, helping us understand the flow of control. They display the order in which activities happen and whether they occur one after the other (sequential) or at the same time (concurrent).

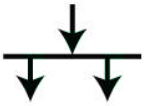
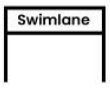
Activity Diagram vs Sequence Diagram:

The choice between using a sequence diagram or an activity diagram depends on the nature of the use case and what aspects you want to emphasize.

Typically, Activity Diagrams are created during early analysis and design to understand workflows. Sequence Diagrams follow during detailed design and implementation to capture specific object interactions.

Sequence Diagram	Activity Diagram
• Provides detailed, step-by-step interactions, good for precise behavior analysis. • Limited support for branching, usually with simple alternatives or loops. • Minimal support for parallel processes, can be complex to visualize. • Can become cluttered in complex systems with many interactions.	• Offers a high-level overview of processes, suitable for visualizing workflows without implementation details. • Supports concurrent activities clearly, including forks and joins • Handles decisions explicitly using decision nodes for clear workflow paths. • Typically, easier to read, providing a simplified overview for stakeholders.

Activity Diagram Notations :

Type and notation	Notation Description
 Activity State	Represent individual steps or tasks in the process (used to represent a set of actions)
 Control Flow	Show the direction of the workflow between actions (Shows the sequence of execution)
 Initial State	Define where the process begins.
 Final State	Define where the process ends.
 Decision node	Indicate branching points where choices are made.
 Fork	Split behavior into a set of parallel or concurrent flows of activities (or actions)
 Join	Bring back together a set of parallel or concurrent flows of activities (or actions).
 Swimlane	Organize activities by the responsible actor or department (optional)