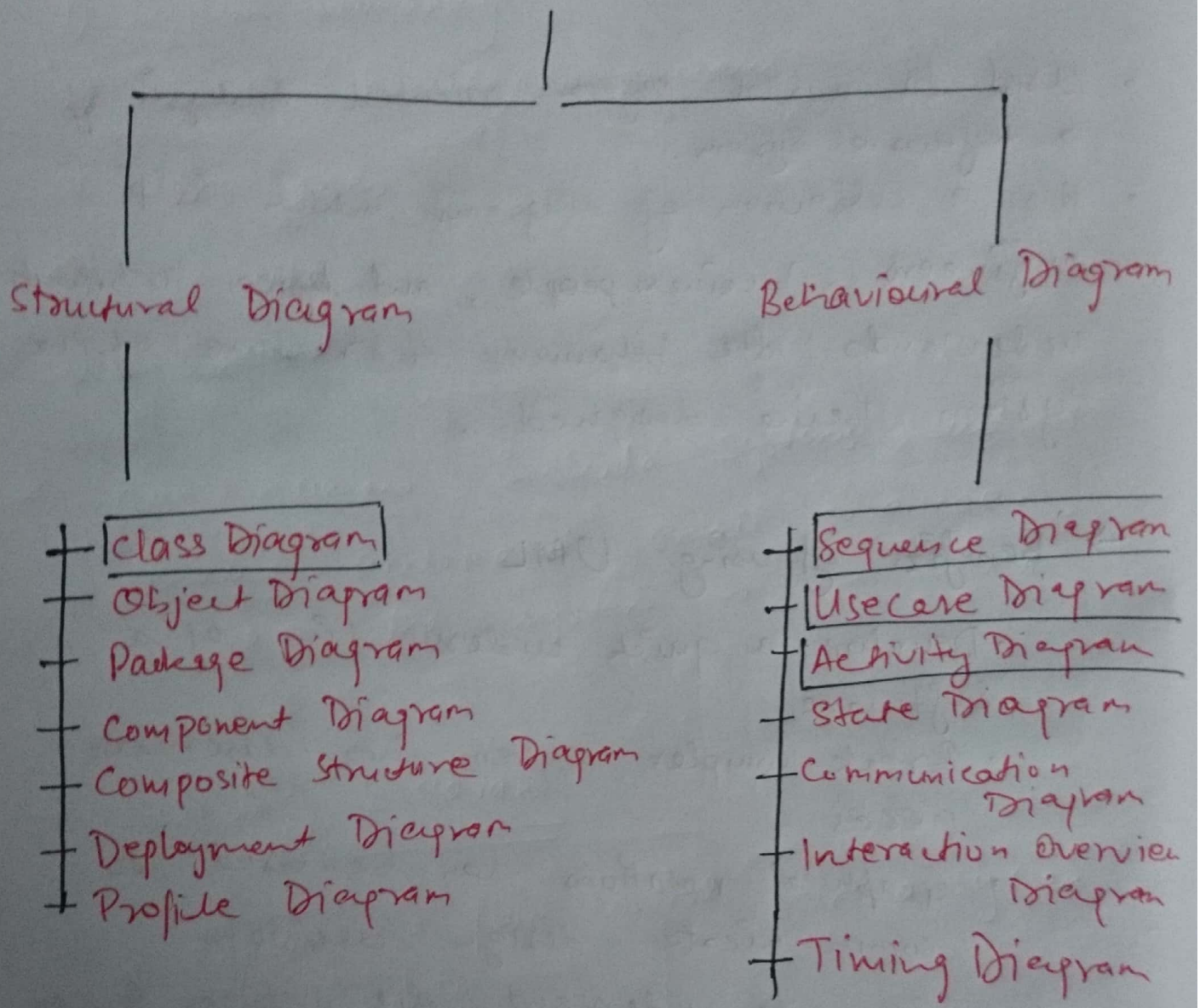
# UML DIAGRAMS

- Used to model object-oriented Analysis of a software System.
- His a collection of diagrams, which helps engineers, business people, and system architects understands the behaviour & structure of the system being designed.

→ Benefits of using UML

- Develop a quick understanding of a software
- Break complex system into discreate pieces
- Graphical notations can be used to communicate design decisions
- Easy to understand the system & can be handed the system to next team.

# UML Diagrams

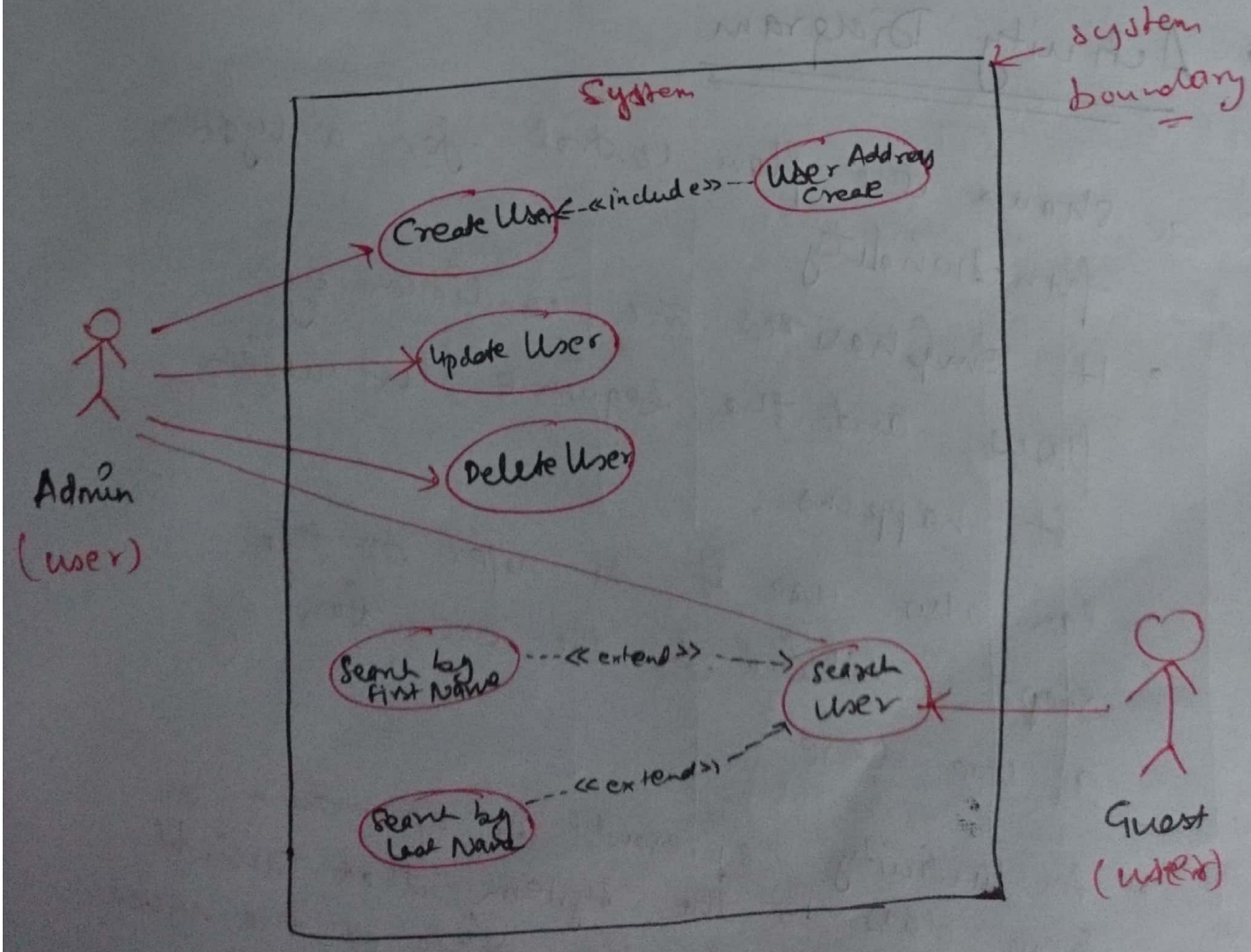



* Most commonly used UML Diagrams

# → Use Case Diagram

- Used to analyze system's high - level requirements. These requirements are expressed through different use cases.
- Each use case should provide some observable & valuable results to the users

- High level functional behaviour
- What system does from the user point of view
- What system will do and will not do.



→ Different components of use case diagrams :→

1) System boundary
2) Users (humans / non-humans)
3) Use Cases
4) Include
5) Extend

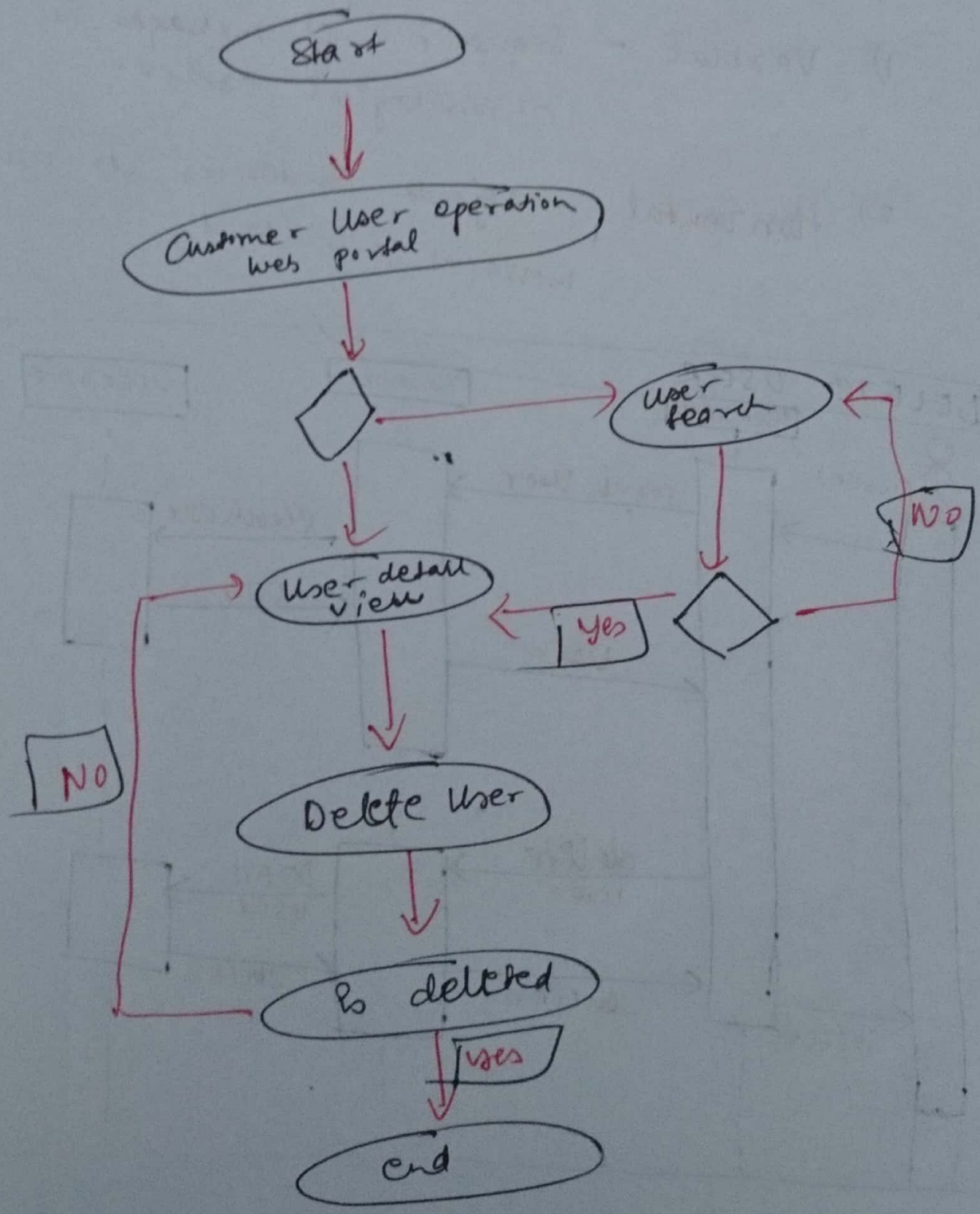
* non-humans include systems, such as sending automated email to a user. So here the system is the user.

## → Activity Diagram

- shows the flow control for a system functionality
- It emphasizes the condition of flow and the sequence in which it happens.
- Can also use it to refer to the steps involved in the execution of a use case.
- An activity represents an operation on some class in the system that results in a change in the state of the system.

- activity diagrams are used to model work flow or business processes and internal operation

for eg. delete User from previous Use case example.



→ Sequence diagram

- Interactions among classes in terms of an exchange of messages over time.
- It's a call different Objects in their ~~sequence~~ sequence.

Sequence diagram has two dimensions :→

1) Vertical – Sequence of messages in chronological order

2) Horizontal – Objects instances to which messages are sent.