Mediator Pattern

A Mediator Pattern says that "to define an object that encapsulates how a set of objects interact".

I will explain the Mediator pattern by considering a problem. When we begin with development, we have a few classes and these classes interact with each other producing results. Now, consider slowly, the logic becomes more complex when functionality increases. Then what happens? We add more classes and they still interact with each other but it gets really difficult to maintain this code now. So, Mediator pattern takes care of this problem.

Mediator pattern is used to reduce communication complexity between multiple objects or classes. This pattern provides a mediator class which normally handles all the communications between different classes and supports easy maintainability of the code by loose coupling.

Benefits:

* It decouples the number of classes.
* It simplifies object protocols.
* It centralizes the control.
* The individual components become simpler and much easier to deal with because they don't need to pass messages to one another.

The components don't need to contain logic to deal with their intercommunication and therefore, they are more generic.

Usage:

* It is commonly used in message-based systems likewise chat applications.
* When the set of objects communicate in complex but in well-defined ways.

UML for Mediator Pattern:

