Protocol

I - CONOCIMIENTOS Y HABILIDADES DE LOS MESEROS

2. Etiqueta del Mesero:
- Saludar al cliente
- Dar la bienvenida
- Orientar su ingreso y ubicación
- Ayudar a la acomodación
- Respetar la forma de pensar del cliente
- Escuchar activamente
- Respetar la forma de ser de las personas
- Hablar con cortesía
- Respetar la forma de actuar del cliente
- Usar vocabulario adecuado
- Identificar las necesidades del cliente

A protocol defines a **blueprint of methods, properties, and other requirements** that suit a particular task or piece of functionality. The protocol can then be adopted by a class, structure, or enumeration to provide an actual implementation of those requirements. Any type that satisfies the requirements of a protocol is said to **conform to that protocol**.

Delegation

Delegation is a **design pattern** that enables a class or structure to hand off (or delegate) some of its responsibilities to an instance of another type

According to Apple

Delegation is a simple and powerful pattern in which one object in a program **acts on behalf of, or in coordination with, another object.** The delegating object keeps a reference to the other object–the delegate–and at the appropriate time sends a message to it. The message informs the delegate of an event that the delegating object is about to handle or has just handled. The delegate may respond to the message by updating the appearance or state of itself or other objects in the application, and in some cases it can return a value that affects how an impending event is handled. The main value of delegation is that it allows you to easily customize the behavior of several objects in one central object.

This design pattern is implemented by defining a protocol that encapsulates the delegated responsibilities, such that a conforming type (known as a delegate) is guaranteed to provide the functionality that has been delegated. Delegation can be used to respond to a particular action, or to retrieve data from an external source without needing to know the underlying type of that source.

Diagram Delegate

responsibilities

Protocol

Class A acts on behalf of, or in coordination with Class B

Reference

Class B has an attribute of type protocol call delegate

Class A: Protocol

Class A has an attribute of type Class B

Class B

Must conform methods and attributes of the Protocol

Delegates via attribute