

Ambito de decision (Objetos / Arquitectura / Persistencia / Otro)	Componente/s impactado/s	Decisión	Otras Alternativas	Justificación de la decisión
Objeto	Colaborador, PersonaJuridica, PersonaHumana	Clase abstracta con clases concretas	clase	Representar personas humanas y jurídicas, y no repetir los atributos que comparten ambas entidades. Además nos permite desacoplar a la persona jurídica de la humana.
Objeto	MedioDeContacto, Telefono, Whatsapp, CorreoElectrónico	Strategy	Clase abstracta con clases concretas	Optamos por un strategy ya que necesitamos distintas formas de ejecutar un mismo comportamiento de distintas formas. Además nos permite abstraer el código y nos aporta mayor flexibilidad y extensibilidad. Por otro lado, no nos decidimos por una clase abstracta ya que solo tendría comportamiento y ningún atributo en común entre los medios de contacto.
Objeto	Contribución, DonacionDeDinero, DonacionViandas, DistribucionViandas, HacerseCargoDeUnaHeladera, RegistrarPersonasEnSituacionVulnerable, OfrecerProductos	Clase abstracta con clases concretas	Interfaz con clases que la implementen	Decidimos hacer una clase abstracta ya que, para todos los tipos de contribuciones, a la hora de verificar quién la puede efectuar y quién no, la interfaz nos dificultaba este apartado. De esta forma, con el método "puedeRealizarContribucion()", podemos verificar, sin repetir lógica, qué clase de persona puede realizar determinada colaboración. Además, nos permite desacoplar las diferentes formas de contribución.
Objeto	TipoDePersona	Enum	-	Funciona en conjunto con la clase abstracta Contribucion, su utilidad es para detallar qué tipo de persona debería tener permiso para realizar una determinada colaboración.
Objeto	Documento, TipoDocumento	Clase	-	Value Object. Para agrupar el tipo de documento con el número de documento. Además, el TipoDocumento nos permite modelar los distintos tipos de Documentos que podrían agregarse más a futuro.
Objeto	VerificadorContrasenia	Clase	-	Para desacoplarlo de Registrar Usuario y usarlo en Usuario para verificar la contraseña en caso de que se la cambie.
Objeto	ValidacionContrasenia, LongitudMinimaContrasenia, LongitudMaximaContrasenia, CaracterRepetido, ValidacionTopPeoresContrasenia y FormatoContrasenia	Clase abstracta con clases concretas	Interfaz	Para extraer la lógica en común entre las diferentes restricciones de la contraseña.
Objeto	ValidarUsuario	Clase	-	Para separar la responsabilidad de validar si el nombre y la contraseña son correctas dentro de la clase IniciarSesion.
Objeto	Uso	Clase	atributo de TarjetaHeladera	Debido a que para cada uso se necesita saber la fecha y la heladera en la que se usó dicha tarjeta. Además, tienen un límite de tarjetas por lo que no sería indefinido.
Objeto	RegistroDePersonasEnSituacionVulnerable	Clase	-	Relaciona a la persona en situación vulnerable con el colaborador que lo registró en el sistema.
Objeto	TemperaturaHeladera	Clase	atributos en Heladera	Porque la temperatura mínima y máxima están relacionadas entre sí y que ambas son temperaturas. Además, evitamos acoplarlo todo en la heladera.
Objeto	Sensor, SensorDeTemperatura, SensorDeMovimiento	Observer	-	El Sensor es una clase abstracta para abstraer los métodos del Sujeto dentro del patrón observer, ya que son iguales para todos los sensores. La única diferencia con el patrón es que en este caso tenemos un solo Observador que es la heladera en la que está el sensor. Por otro lado, los métodos de agregar y quitar la heladera serían de utilidad sólo por si se quiere sacar un sensor de una heladera y ponerlo en otra.
Objeto	CalculadorDePuntos, RegistroContribucion, Coeficiente	Template method	método en Colaborador	Nos permite separar la lógica de obtener los diferentes atributos que varían del cálculo del puntaje. Además, en caso de que no se puedan obtener algunos atributos como los meses de la heladera activa, se pueden setear como 0 o algún valor por defecto, sin afectar al cálculo. También nos permite desacoplar los coeficientes de los atributos del cálculo del puntaje.
Objeto	IAdapter, APIAdapter	Adapter	-	Para adaptar la interfaz de la API externa con nuestra clase Ubicacion en la respuesta de la API.
Objeto	estadoHeladeraActual historialEstadoHeladera	atributos	-	Para poder acceder rápidamente al estado actual de la heladera, es necesario el atributo estadoHeladeraActual y luego para el cálculo de los meses activos de la heladera es necesario llevar un historial del estado de la heladera con las fechas del cambio de estado de inicio a fin.