

A decorative graphic on the left side of the slide consisting of two overlapping parallelograms. The front one is blue and the back one is a light green. They are positioned diagonally, with the blue one partially covering the green one.

# Object Oriented Design

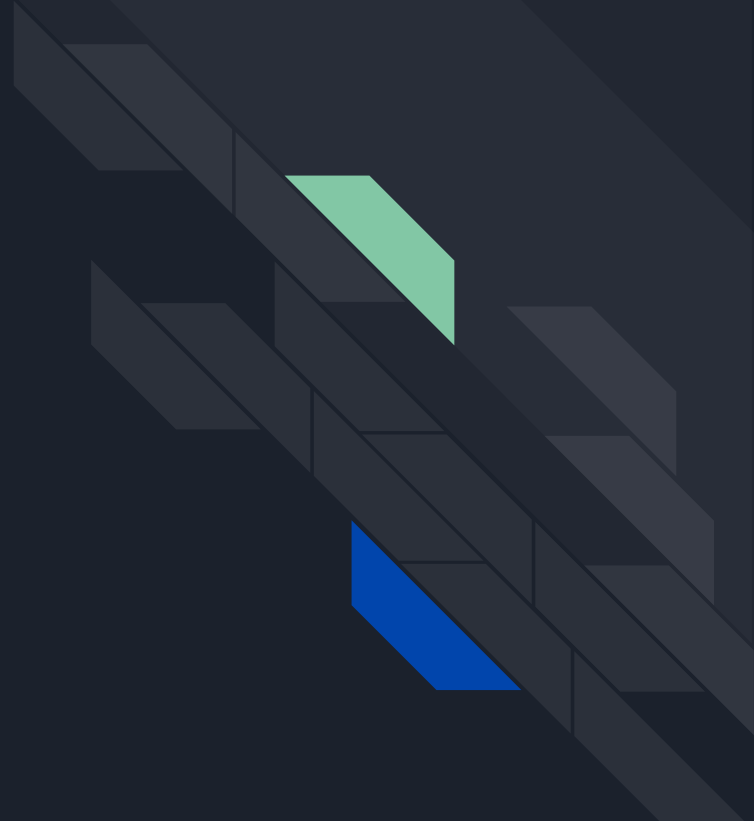
IIC2143 - 2020-1

# Índice

¿Qué es un modelo de dominio?

Estrategias para identificar entidades

Ejercicios “prácticos” 🎉



# Modelos de dominio






# Estrategias para identificar entidades

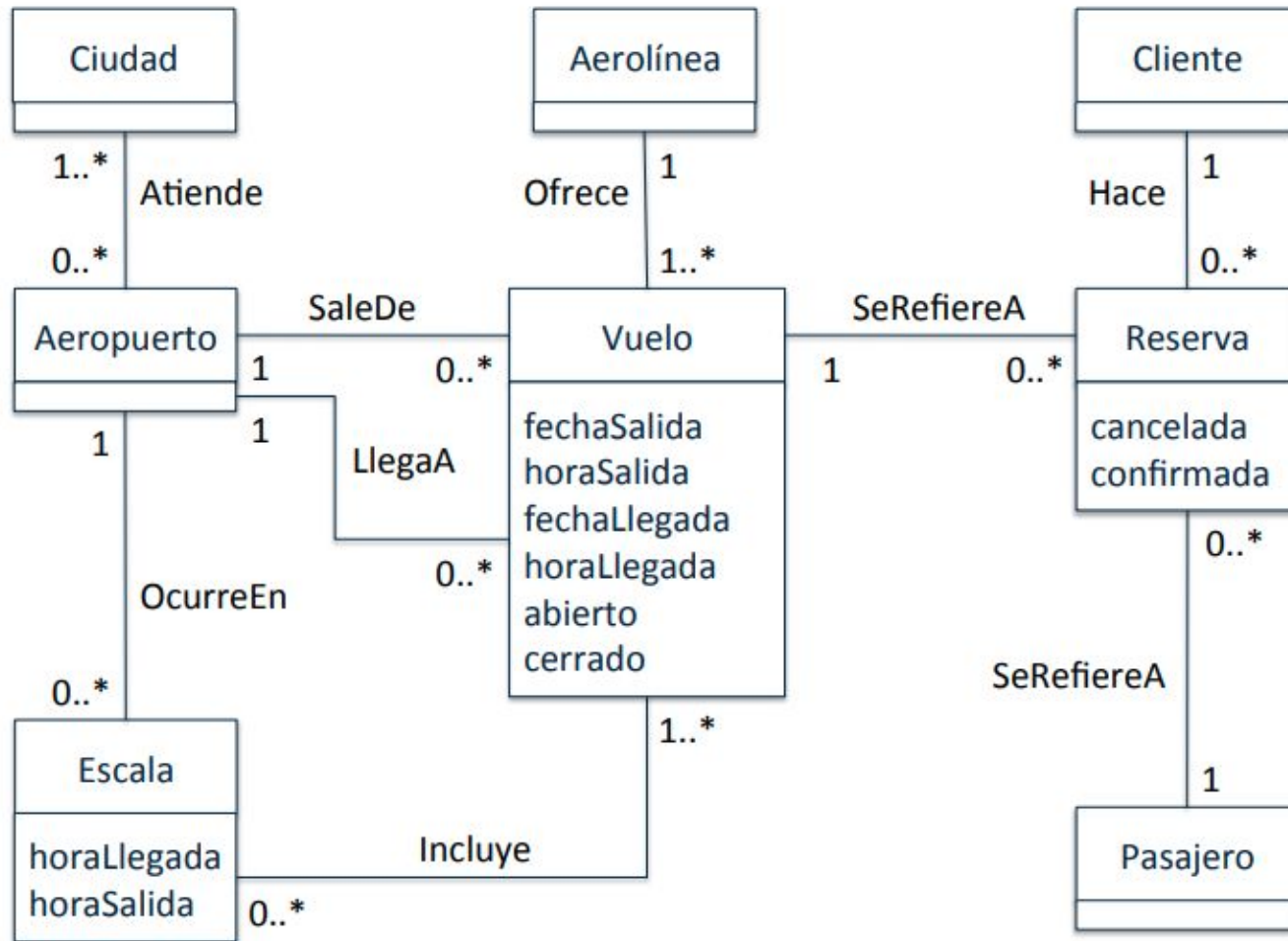
- Preguntarse “¿qué?” más que “¿cómo?”
- Identificar los contenedores de información (frases como “tiene un nombre y un apellido”, “posee un tamaño y un color” y similares denotan a estos elementos).
- Identificar elementos que realizan acciones (frases como “puede pedir una hora”, “ordena los productos” y similares denotan a estos elementos).
- Identificar elementos auxiliares.


# Ejemplos prácticos





Considera un sistema de reserva de vuelos para una agencia de viajes, con las siguientes características. Las líneas aéreas ofrecen varios vuelos. Un vuelo es abierto para hacer reservas y nuevamente cerrado por orden de la compañía. Un cliente puede reservar uno o más vuelos y para diferentes pasajeros. Una reserva se refiere a un único vuelo y a un único pasajero, y puede ser cancelada o confirmada. Los vuelos tienen aeropuerto, fecha y hora de salida y de llegada, y pueden incluir escalas en aeropuertos; cada escala tiene una hora de llegada y una hora de salida. Cada aeropuerto atiende a una o más ciudades.






Una empresa de arriendo de habitaciones de un complejo residencial se encuentra desarrollando una aplicación web para ofrecer sus servicios a través del mercado digital. La aplicación debe permitir que administradores, moderadores y arrendatarios puedan ingresar para realizar distintos tipos de operaciones. Todos deben definir un email, el cual debe ser único, y un nombre para poder identificarlos. Solo los arrendatarios pueden arrendar habitaciones.

Todo arriendo debe obligatoriamente pasar a través de un contrato de arriendo, el cual define una fecha de inicio y una fecha de término. Dado un contrato de arriendo vigente, la aplicación debe permitir a los arrendatarios efectuar los respectivos pagos de sus arriendos en cuotas mensuales, las cuales vencen en una determinada fecha. El valor de la cuota depende del monto acordado al momento de firmar el contrato, la habitación y servicios adicionales que el cliente pudo haber solicitado. Estos incluyen servicios como muebles, pago de cuentas (agua, luz, basura) incluido, y servicio de limpieza de la habitación. Cada uno de estos servicios debe contar con un valor para poder ser considerado por el cliente. La habitación por su lado debe contar con un número y una superficie. Cada pago efectivamente realizado por un cliente debe figurar en el sistema; sin embargo, nótese que el monto cancelado por el cliente no necesariamente corresponde al monto de la cuota ya que en caso de atraso deben agregarse intereses.





Todo cliente cuenta adicionalmente con una hoja de vida en la cual figura el registro de todas las transgresiones cometidas por él, incluyendo atrasos en los pagos, violación de las reglas de la comunidad y delitos comunes de los que se tenga conocimiento. Esto le permite a la agencia tomar estos hechos en consideración en futuras solicitudes de arriendo efectuados por el cliente.

Por último, con el fin de promover la nueva aplicación, se está pensando en implementar un sistema de referencias en el que arrendatarios puedan invitar a amigos para así conseguir descuentos para ambos. Como está concebida la funcionalidad, si un arrendatario se registra al sistema luego de ser referido por un amigo, la aplicación debe generarle un voucher de descuento a ambos por un determinado monto para ser usado cuando estimen conveniente.

Dibuje un modelo de dominio (usando la notación del diagrama de clases de UML) que ilustre las principales clases y relaciones (asociaciones, agregaciones, composiciones y generalizaciones) entre clases que se desprenden del enunciado. En las clases, incluya todos los atributos que se desprendan del enunciado; para las relaciones, especifique etiquetas y multiplicidades; añada generalizaciones donde sea pertinente. Especificar visibilidad y tipos de los atributos, además de navegabilidad de las relaciones es opcional.

