

Práctica Modelado de Datos & Persistencia de Datos

Modelado de Datos (desde el modelo datos hacia el modelo de clases)

1. Descargar el Template Base del Aula Virtual.
2. Leer el dominio asignado al equipo.
3. Diseñar y generar:
 - a. **Modelo de Datos** que permita cumplir con los requerimientos especificados en el Dominio.
 - b. **Modelo de Clases** a partir del modelado de Datos generado en el punto anterior.
4. Detallar las decisiones más significativas tomadas en los modelos del punto anterior.

Enunciados

- Enunciado 1: Recomendaciones de lecturas: [2021/12/18](#)
- Enunciado 2: Sistema de Logística: [2021/12/22](#)
- Enunciado 3: Cooperativa de Internet Rural: [2022/03/05](#)

Persistencia de Datos (desde el modelo de clases hacia el modelo de datos)

1. Leer el dominio asignado al equipo.
2. Implementar la persistencia del modelo planteado, en Java, con Hibernate como ORM.
3. Generar el Modelo de Datos a partir del modelo persistido.

Enunciados

Enunciado 1: Audiencias Públicas

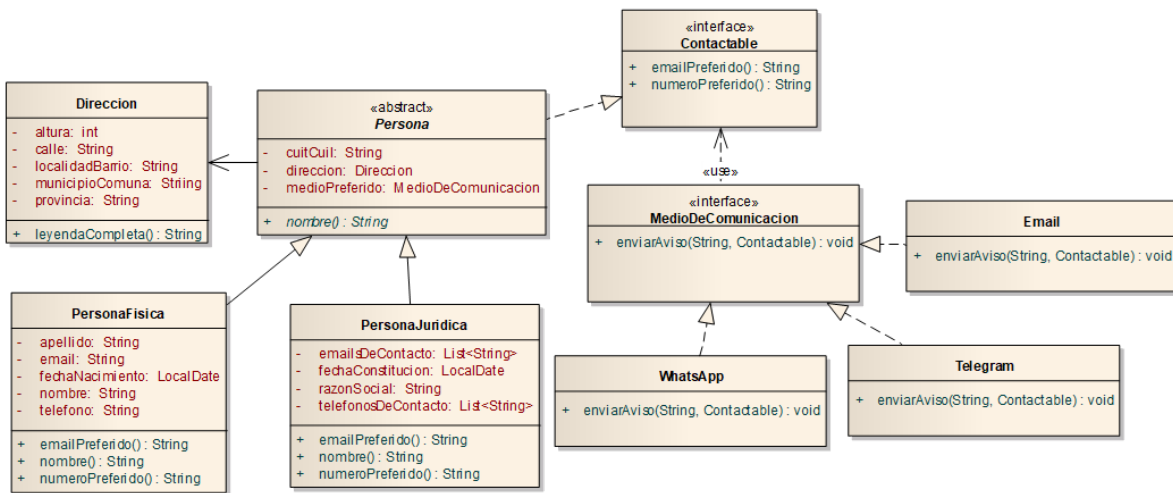
Nos han solicitado el diseño y desarrollo de un sistema de Audiencias Públicas.

Tanto los sujetos obligados como los participantes de las audiencias pueden ser personas físicas o jurídicas, de las cuales siempre interesa mostrar su nombre en el resumen de la audiencia, independientemente de su tipo.

Además, cada persona (física o jurídica) debe poseer un medio preferido de comunicación por el cual se le enviarán avisos de distintos tipos desde la plataforma.

Nos han dicho que la dirección debe ser meramente informativa y que no es obligatorio completar todos los campos, excepto la provincia y municipio/comuna.

Nuestro equipo ya estuvo trabajando y generó el siguiente diagrama de clases que, por cierto, consideramos correcto:



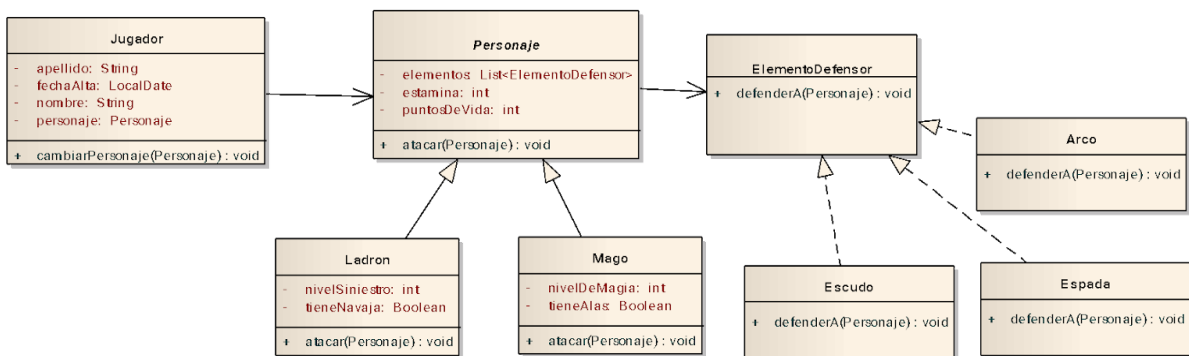
Enunciado 2: Gloranta

Nos han solicitado el diseño y desarrollo de un juego de roles.

Los distintos jugadores pueden jugar y poseer un solo personaje a la vez.

Los personajes a contemplar en esta primera iteración son los ladrones y los magos. Se sabe que cada uno de estos personajes ataca de una forma diferente a sus oponentes y que poseen diferentes cualidades. Además, los jugadores pueden acumular distintos elementos defensores para usarlos cuando su personaje sea atacado. Cada elemento defensor se comporta de una forma diferente y se sabe que no posee ninguna característica que lo deteriore en el tiempo.

Nuestro equipo ya estuvo trabajando y generó el siguiente diagrama de clases que, por cierto, consideramos correcto:



Enunciado 3: Módulo de Stock

Nos han solicitado el diseño y desarrollo de un módulo que se encargue de calcular el stock de productos de cualquier local comercial, así como también el cálculo de precios de los mismos.

Se debe tener en cuenta que no solamente existen y se venden productos simples, sino que también existen combos. Un combo está formado por productos que se venden juntos. También pueden existir combos de combos.

Por ejemplo, un local de motos vende motos, cascos, guantes, chalecos, pilotos, entre otros productos; pero también vende algunos combos como por ejemplo guantes + casco + chaleco.

Cada producto tiene un precio en particular. El precio del combo es el resultado de la suma de los productos que contiene.

Además, se debe permitir aplicar descuentos a los distintos productos/combo, los cuales podrían ser acumulables.

Por último, nos han avisado que los productos/combo podrían ser vendidos con distintos packagings, cada uno de los cuales tiene un precio particular que se debería sumar al precio final del producto.

Nuestro equipo ya estuvo trabajando y generó el siguiente diagrama de clases que, por cierto, consideramos correcto:

