

**Ministerio de Producción**  
**Secretaría de Emprendedores y de la**  
**Pequeña y Mediana Empresa**  
**Dirección Nacional de Servicios Basados en el**  
**Conocimiento**



**Programadores**

# Guía de Ejercicios Prácticos para el Módulo

## Base de Datos



## Tabla de Contenido

|   |           |
|---|-----------|
| <b>INTRODUCCIÓN .....</b>   | <b>5</b>  |
| <b>ENUNCIADOS DE LOS EJERCICIOS A DESARROLLAR EN LA GUÍA .....</b>  | <b>7</b>  |
| <b>EJERCICIO 1: CASO PRÁCTICO – CINE .....</b>  | <b>7</b>  |
| PRESENTACIÓN DEL CASO DE ESTUDIO.....   | 7         |
| <b>EJERCICIO 2: CASO PRÁCTICO – PIZZERÍA .....</b>  | <b>8</b>  |
| PRESENTACIÓN DEL CASO DE ESTUDIO.....   | 8         |
| <b>EJERCICIO 3: CASO PRÁCTICO – ESTACIONAMIENTO DE UNIVERSIDAD .....</b>  | <b>8</b>  |
| PRESENTACIÓN DEL CASO DE ESTUDIO.....   | 8         |
| <b>EJERCICIO 4: CASO PRÁCTICO – MERCADO DE ABASTO .....</b>   | <b>9</b>  |
| PRESENTACIÓN DEL CASO DE ESTUDIO.....   | 9         |
| <b>EJERCICIO 5: CASO PRÁCTICO – PANADERÍA .....</b>   | <b>10</b> |
| PRESENTACIÓN DEL CASO DE ESTUDIO.....   | 10        |
| <b>EJERCICIO 6: CASO PRÁCTICO – CONSULTORIO ODONTOLÓGICO .....</b>  | <b>10</b> |
| PRESENTACIÓN DEL CASO DE ESTUDIO.....   | 10        |
| <b>SOLUCIONES PROPUESTAS .....</b>  | <b>11</b> |
| <b>EJERCICIO 1: CASO PRÁCTICO – CINE .....</b>  | <b>11</b> |
| PRESENTACIÓN DEL CASO DE ESTUDIO.....   | 11        |
| MODELO DE DOMINIO.....  | 12        |
| CONSIGNAS: .....  | 13        |
| SOLUCIÓN PROPUESTA PARA EL DIAGRAMA ENTIDAD-RELACIÓN QUE MAPEA LAS CLASES DEL MODELO DE DOMINIO DEL COMPLEJO DE CINES: .....              | 14        |
| <b>EJERCICIO 2: CASO PRÁCTICO – PIZZERÍA .....</b>  | <b>15</b> |
| PRESENTACIÓN DEL CASO DE ESTUDIO.....   | 15        |
| MODELO DE DOMINIO.....  | 16        |
| CONSIGNAS: .....  | 17        |
| SOLUCIÓN PROPUESTA PARA EL DIAGRAMA ENTIDAD-RELACIÓN QUE MAPEA LAS CLASES DEL MODELO DE DOMINIO DE LA PIZZERÍA:.....                      | 19        |
| <b>EJERCICIO 3: CASO PRÁCTICO – ESTACIONAMIENTO DE UNIVERSIDAD .....</b>  | <b>20</b> |
| PRESENTACIÓN DEL CASO DE ESTUDIO.....   | 20        |
| MODELO DE DOMINIO.....  | 21        |
| CONSIGNAS: .....  | 22        |
| SOLUCIÓN PROPUESTA PARA EL DIAGRAMA ENTIDAD-RELACIÓN QUE MAPEA LAS CLASES DEL MODELO DE DOMINIO DEL ESTACIONAMIENTO DE UNIVERSIDAD: ..... | 23        |
| <b>EJERCICIO 4: CASO PRÁCTICO – MERCADO DE ABASTO .....</b>   | <b>24</b> |
| PRESENTACIÓN DEL CASO DE ESTUDIO.....   | 24        |
| MODELO DE DOMINIO.....  | 25        |
| DESCRIPCIÓN CON RESUMEN ESENCIAL DEL CASO DE USO 15. GENERAR REPORTE DE PUESTOS POR ESTADO.....   | 26        |
| CONSIGNAS: .....  | 27        |
| DIAGRAMA ENTIDAD-RELACIÓN .....   | 28        |
| <b>EJERCICIO 5: CASO PRÁCTICO – PANADERÍA .....</b>   | <b>29</b> |

|   |           |
|---|-----------|
| PRESENTACIÓN DEL CASO DE ESTUDIO.....   | 29        |
| MODELO DE DOMINIO.....  | 30        |
| CONSIGNAS: .....  | 30        |
| SOLUCIÓN PROPUESTA PARA EL DIAGRAMA ENTIDAD-RELACIÓN QUE MAPEA LAS CLASES DEL MODELO DE DOMINIO DE LA PANADERÍA:.....               | 31        |
| <b>EJERCICIO 6: CASO PRÁCTICO – CONSULTORIO ODONTOLÓGICO .....</b>  | <b>32</b> |
| PRESENTACIÓN DEL CASO DE ESTUDIO.....   | 32        |
| MODELO DE DOMINIO.....  | 33        |
| CONSIGNAS: .....  | 34        |
| SOLUCIÓN PROPUESTA PARA EL DIAGRAMA ENTIDAD-RELACIÓN QUE MAPEA LAS CLASES DEL MODELO DE DOMINIO DEL CONSULTORIO ODONTOLÓGICO: ..... | 35        |
| FUENTES DE INFORMACIÓN .....  | 36        |

## Introducción

---

La guía práctica del Módulo Base de Datos incluye los casos de estudio vinculados a los contenidos desarrollados en el apunte teórico del módulo. El objetivo de esta guía es brindar una herramienta de apoyo, que facilite el desarrollo de los temas y posibilite aplicar los conocimientos adquiridos, mostrando casos prácticos y su resolución propuesta.

En el apunte teórico se trabajó alrededor de los siguientes ejes temáticos: sistema administrador de bases de datos, tecnología de bases de datos y sus diferentes tipos, el modelo de entidad-relación y el diagrama de entidad-relación como herramienta para modelar. Luego se trabajó el modelo lógico relacional, los motores de bases de datos relacionales en general y en particular MySQL. También se desarrollaron conceptos sobre SQL, tales como lenguaje de consulta y modelos de persistencia y cómo se mapea objetos-relacional utilizando Hibernate.

Esta guía utiliza los mismos casos que se presentaron por primera vez en la Guía de Prácticos del Módulo de Programación Orientada a Objetos, dado que la intención es dar un cierre a esos mismos casos, agregando la parte de persistencia que no se trató antes.

Entonces, se toman los diagramas de clase que modelan el dominio, esos diagramas se utilizan para mapear creando primero el diagrama de entidad-relación correspondiente y luego creando la base de datos y la persistencia con Hibernate.

En dos de los casos de estudio, Pizzería y Mercado de Abasto, se agregó un caso de uso de generación de reportes, para poder mostrar el uso de una herramienta de reportes llamada Jasper Reports.

Para la instalación de la aplicación necesaria para trabajar con el módulo de Base de Datos, está disponible el siguiente instructivo:

- Instructivo: **Generación de reportes con Jasper Reports**, disponible en el repositorio, en la siguiente dirección: <https://github.com/111milprogramadores/bd-instructivos>

Al igual que en el módulo de Programación Orientada a Objetos, las soluciones propuestas y los instructivos, junto con el resto del material, está a disposición en el repositorio.

El repositorio se encuentra en: <https://github.com/111milprogramadores>


[Personal](#) [Open source](#) [Business](#) [Explore](#)
[Pricing](#) [Blog](#) [Support](#)

[Sign in](#)
[Sign up](#)


## 111mil Programadores

111milprogramadores

Programa de Estímulo a la  
Formación de Analistas del  
Conocimiento

[Block or report user](#)

[Argentina](#)

[111milprogramadores@gmail...](mailto:111milprogramadores@gmail.com)

<https://www.argentina.gob.ar...>

Overview Repositories **10** Stars **0** Followers **23** Following **0**

### Popular repositories

#### 01-cine

Ejercicio 1 de Programación Orientada a Objetos y Bases de Datos

● Java ★ 1

#### presentaciones

Presentaciones utilizadas en las capacitaciones del programa 111mil programadores.

★ 1 🍴 1

#### 02-pizzeria

Ejercicio 2 de Programación Orientada a Objetos y Bases de Datos

● Java

#### poo-instructivos

Instructivos correspondientes al módulo Programación Orientada a Objetos.

#### bd-instructivos

Instructivos correspondientes al módulo Bases de Datos

#### apuntes

Apuntes Teóricos y Prácticos del Programa

Finalmente, en el repositorio también hay un instructivo para la descarga del proyecto desde el repositorio público de 111mil creado en la herramienta GitHub<sup>1</sup>.

<sup>1</sup> GitHub: Es una plataforma de desarrollo colaborativo de software para alojar proyectos utilizando el sistema de control de versiones Git. El código se almacena de forma pública, aunque también se puede hacer de forma privada, creando una cuenta de pago.

## Enunciados de los Ejercicios a Desarrollar en la Guía

---

### Ejercicio 1: Caso Práctico – Cine

#### Presentación del Caso de Estudio

Este caso se encuentra mencionado en el material teórico del módulo: se trata de un Sistema de Gestión de Ventas y Reservas de Entradas para un Complejo de Cines. El funcionamiento del negocio se describe a continuación:

Un complejo de cines está integrado por varios cines ubicados principalmente en los centros comerciales de la ciudad. Cada cine cuenta con una cantidad de salas, que son las que exhiben las películas en las distintas funciones cinematográficas. La programación de las salas se renueva en forma semanal, existiendo la posibilidad de que algunas salas queden sin uso. Cabe mencionar que no todas las salas tienen la misma capacidad (cantidad de butacas).

La programación es la que determina qué películas van a proyectarse y los horarios para cada función de cada una de las salas, para todos los cines. Esta programación se realiza en forma centralizada, desde la administración del Complejo, tomándose como base la información de las películas próximas a estrenar, que envía el INCAA (Instituto Nacional de Cines y Artes Audiovisuales). La programación implica el diseño de las funciones y sus horarios en forma anticipada, debiendo el responsable de la misma, habilitar cada función en el momento que desee permitir la reserva y/o venta de entradas para la misma.

La entrada que se le entrega al cliente representa el comprobante de venta y como tal debe cumplir con lo reglamentado en la Ley de Facturación vigente, debiendo contener como datos: nro. de venta, fecha de venta, número de función, sala en la que se proyecta la película, el nombre de la película, fecha y hora de la función, el precio, el tipo de entrada (si es mayor, menor, jubilado) y la calificación de la película, que según especificaciones de la Ley de Cine Nro. 17.741, debe ser informada tanto en la entrada como al inicio de la película. Es importante destacar que la entrada es válida únicamente para la fecha, hora y función indicadas en la misma.

Los tipos de entradas y los días y horarios de proyección son los que determinan el precio de la entrada, que también pueden variar en cada cine del complejo. Las funciones admiten ciertos tipos de entradas y otros no, dependiendo de factores como: horarios, calificación de las películas, etc. Por ejemplo: si una película está calificada como para mayores de 16 años, para esa función no se pueden vender entradas de TIPO = MENOR. Cada función tiene asociado un tipo de función, que determina si la función es un pre-estreno, un estreno o una función normal.

---

## Ejercicio 2: Caso Práctico – Pizzería

### Presentación del Caso de Estudio

Una pizzería de la ciudad ofrece a sus clientes una amplia variedad de pizzas de fabricación propia, de varios tamaños (8, 10 y 12 porciones). Los clientes tienen a disposición un menú que describe para cada una de las variedades, el nombre, los ingredientes y el precio según el tamaño y el tipo (a la piedra, a la parrilla, de molde) de la pizza. Los clientes realizan sus pedidos en el mostrador.

El pedido debe contener el nombre del Cliente, para llamarlo cuando su pedido está listo; la cantidad de pizzas, el tamaño, la variedad, la fecha del pedido, la hora en la que el pedido debe entregarse y la demora estimada informada al cliente.

El pedido va a la cocina y cuando está preparado se informa al que lo tomó para que se genere la factura correspondiente y se le entregue el pedido al cliente.

El dueño de la pizzería ha manifestado la necesidad de acceder al menos a la siguiente información:

- Variedades y tipos de pizzas más pedidas por los clientes.
- Ingresos (recaudaciones) por períodos de tiempo.
- Pedidos (cantidad y monto) por períodos de tiempo.

## Ejercicio 3: Caso Práctico – Estacionamiento de Universidad

### Presentación del Caso de Estudio

Se describe a continuación el funcionamiento de la playa de estacionamiento de la Universidad Tecnológica y del sistema de información que le da soporte.

- ⇒ Pueden estacionar distintos tipos de vehículos (motos/automóviles), cada uno de ellos con una tarifa de ingreso diferente. Si tiene abono, el precio es menor.
- ⇒ Se puede ingresar a la playa de estacionamiento por varios portones de ingreso diferentes.
- ⇒ No se asignan lugares específicos para los vehículos; las personas que ingresan al estacionamiento deberán ubicar su vehículo en algún lugar que se encuentre disponible.
- ⇒ Los interesados pueden comprar un abono de estacionamiento, de pago anticipado, que hace que el valor de cada estacionamiento sea más económico que si paga cada vez que ingresa a la playa. Debe informar su DNI y la cantidad de dinero que desea acreditar.
- ⇒ Si es la primera vez que estaciona, debe registrar sus datos personales (apellido, nombre, dni), y los datos del o los vehículos (marca, modelo, dominio), con los cuales desea ingresar a la playa de estacionamiento.
- ⇒ Una vez registrado el propietario, cada vez que necesite acreditar dinero informa su DNI y la cantidad de dinero y se le cobra entregándole un comprobante donde consta: apellido y nombre, dni, fecha de la transacción, monto acreditado y monto disponible en su cuenta.
- ⇒ El comprobante (ticket) que se entrega como constancia del cobro tiene los siguientes datos: apellido y nombre del propietario, dni, fecha y hora de la transacción, monto acreditado y monto disponible en



su cuenta, los números de dominio de todos los vehículos registrados de ese propietario y un número único de identificación del comprobante.

- ⇒ Puede tener hasta dos ingresos sin crédito, es decir saldo negativo, que se descontarán de la siguiente vez que acredite dinero en su cuenta.
- ⇒ Mientras tenga crédito, la persona puede ingresar a la playa con cualquiera de los vehículos registrados.
- ⇒ La persona puede en cualquier momento agregar y/o cambiar los vehículos con los que ingresará a la playa de estacionamiento.
- ⇒ El valor del estacionamiento es por el día completo, sin límite de tiempo ni inferior ni superior; es decir se paga un ingreso diario, que es válido independientemente de la cantidad de ingresos que haga durante el mismo día y del tiempo que permanezca en la playa.
- ⇒ Al ingresar se le entrega a la persona un comprobante que contiene: dominio del vehículo, apellido y nombre del dueño del vehículo, el valor del ingreso, la fecha de ingreso y el saldo disponible. También se informa el número de ingreso del día. El portón por el que ingresa y el usuario logueado.
- ⇒ Si el vehículo no está registrado, se guarda en el ingreso el número de dominio del vehículo y se informa como observación que no está registrado.
- ⇒ El primer ingreso del día se cobra, descontando del saldo disponible. A partir del segundo ingreso del día en adelante, el monto debe figurar en cero y se debe informar cuál es el número de ingreso, por ejemplo: "Segundo ingreso del día".
- ⇒ A las personas que desean ingresar a la playa de estacionamiento sin tener el abono de pago anticipado, se les cobra en el momento del ingreso, registrando como observación el número de dominio del vehículo, entregándoles un comprobante con el monto cobrado. Los datos del comprobante en ese caso son: dominio del vehículo, monto, fecha de ingreso, número de vez que ingresa a la playa de estacionamiento, usuario logueado, fecha y hora y portón por el que ingresa.
- ⇒ Si la persona tiene abono, puede tener hasta el valor de dos estacionamientos como saldo negativo, que se descontarán de la siguiente vez que acredite dinero en su cuenta.

## Ejercicio 4: Caso Práctico – Mercado de Abasto

### Presentación del Caso de Estudio

El Mercado de Abasto de Frutas y Verduras de una ciudad de la región necesita un Sistema de Información que brinde soporte a las actividades que allí se realizan.

El mercado está organizado en sectores. Cada sector contiene puestos, los cuales son alquilados a empresas y quinteros (genéricamente clientes) para que allí realicen sus ventas. Existen distintos tipos de puestos (con techo, sin techo, con cámara refrigerante, etc.) y distintas dimensiones para cada puesto (10m<sup>2</sup>, 15m<sup>2</sup>, etc.), para poder ajustarse mejor a las necesidades de cada cliente.

El precio del alquiler depende del sector en el que se encuentre el puesto, el tipo de puesto y sus dimensiones, y está predefinido.

Cuando un cliente desea alquilar uno o más puestos, se verifica la disponibilidad del tipo de puesto que requiere. Si existe disponibilidad y el cliente está de acuerdo con el precio, se realiza un contrato de alquiler por cada puesto que se alquile. En el contrato se especifica la fecha de inicio y fin del alquiler, el monto mensual del alquiler y tiene además un número que identifica el contrato que es único y el nombre del

responsable por parte del Mercado que intervino en la firma del contrato y el responsable de la registración del mismo. Además, cada puesto cuenta con un medidor para el consumo de energía eléctrica. Mensualmente se registran las lecturas de cada medidor, ya que el consumo de cada puesto es facturado al cliente que está alquilando ese puesto. En el momento de efectuar el alquiler, se registra en el contrato la última lectura del medidor del puesto que se está alquilando. Los aspectos vinculados a la facturación quedan excluidos del alcance del sistema, como así también la gestión de cobro de los alquileres.

---

## Ejercicio 5: Caso Práctico – Panadería

### Presentación del Caso de Estudio

La Panadería que se describe en este caso de estudio, pertenece a la Fundación Brisas de Cambio, ubicada en el interior de la provincia de Córdoba. La Fundación tiene el propósito fundamental de contener laboralmente a un grupo numeroso de jóvenes y adultos con discapacidades intelectuales y físicas. Su objetivo es desarrollar proyectos productivos que les permita desempeñarse en un oficio para sentirse útiles y adquirir a diario el conocimiento necesario para desempeñarse en esta actividad dentro de un ambiente laboral sano.

En este contexto, la panadería está atendida por este grupo de personas con capacidades especiales y la intención es desarrollar un producto de software que asista a las personas en el proceso de venta y cobro de los productos que la panadería vende.

---

## Ejercicio 6: Caso Práctico – Consultorio Odontológico

### Presentación del Caso de Estudio

En este consultorio odontológico trabajan varios profesionales que brindan sus servicios. Cuando un paciente necesita atención, debe solicitar un turno previamente. No se atienden pacientes que no tienen turno. El odontólogo para el que el paciente solicita el turno es el que lo va a atender. Cada odontólogo tiene una agenda con los días y horarios en los que puede atender, que se crea mensualmente en función de la disponibilidad que el odontólogo informa, con turnos de 30 minutos de duración. Esta agenda genérica representa los días y horarios de atención que tiene disponible ese odontólogo en términos generales y la duración de su consulta. Esta información se tomará como base para crear la agenda cada mes, considerando para cada mes los días y/u horarios que en ese mes no podrá atender.

Cuando el paciente llama por teléfono, se le pregunta el motivo de la consulta y en función de eso se le asigna uno o más turnos. Por ejemplo, si lo que debe hacerse es un tratamiento de conducto, se le asignan dos turnos de media hora, consecutivos.

## Soluciones Propuestas

---

### Ejercicio 1: Caso Práctico – Cine

#### Presentación del Caso de Estudio

Este caso se encuentra mencionado en el material teórico del módulo: se trata de un Sistema de Gestión de Ventas y Reservas de Entradas para un Complejo de Cines. El funcionamiento del negocio se describe a continuación:

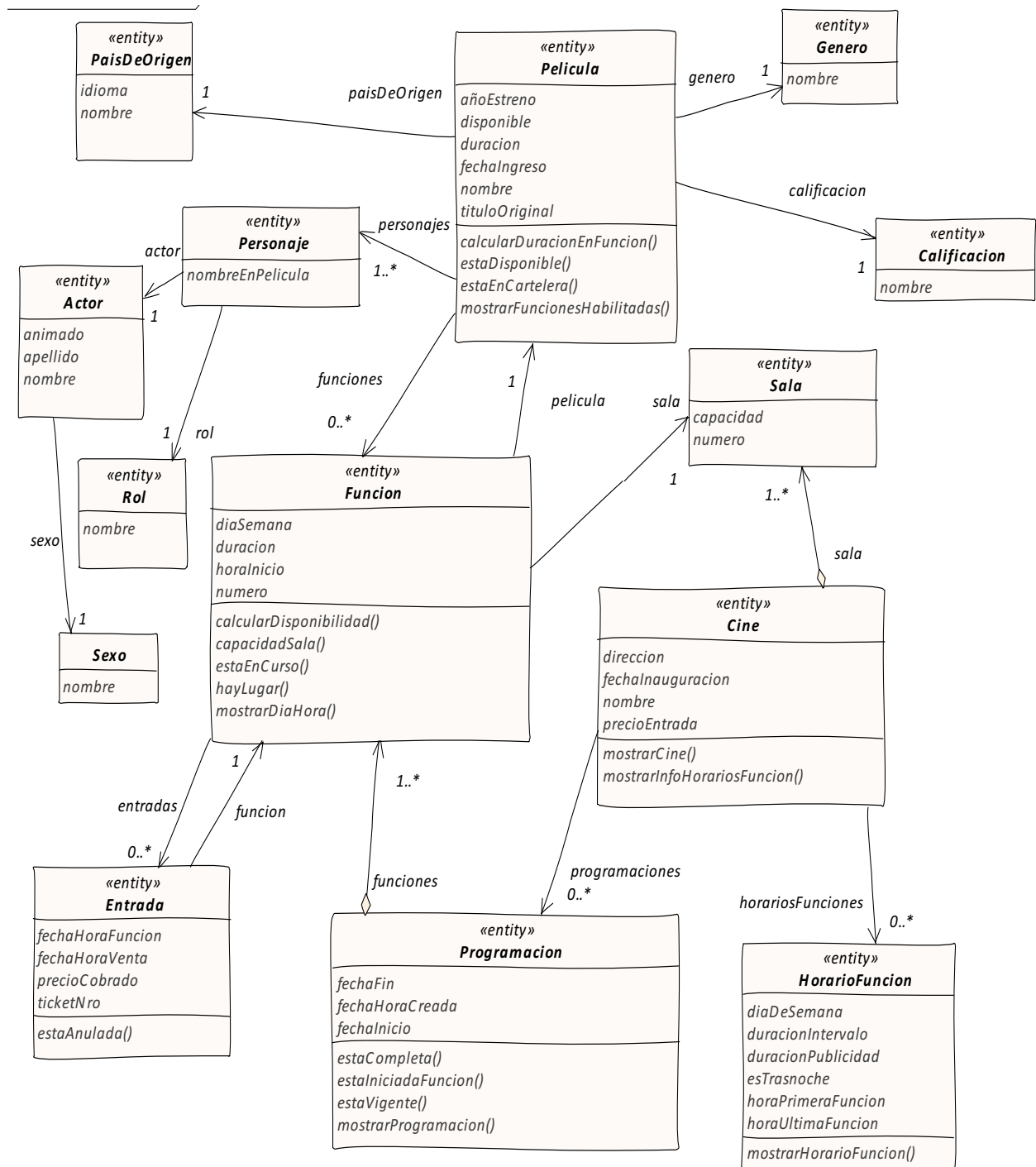
Un complejo de cines está integrado por varios cines ubicados principalmente en los centros comerciales de la ciudad. Cada cine cuenta con una cantidad de salas, que son las que exhiben las películas en las distintas funciones cinematográficas. La programación de las salas se renueva en forma semanal, existiendo la posibilidad de que algunas salas queden sin uso. Cabe mencionar que no todas las salas tienen la misma capacidad (cantidad de butacas).

La programación es la que determina qué películas van a proyectarse y los horarios para cada función de cada una de las salas, para todos los cines. Esta programación se realiza en forma centralizada, desde la administración del Complejo, tomándose como base la información de las películas próximas a estrenar, que envía el INCAA (Instituto Nacional de Cines y Artes Audiovisuales). La programación implica el diseño de las funciones y sus horarios en forma anticipada, debiendo el responsable de la misma, habilitar cada función en el momento que desee permitir la reserva y/o venta de entradas para la misma.

La entrada que se le entrega al cliente representa el comprobante de venta y como tal debe cumplir con lo reglamentado en la Ley de Facturación vigente, debiendo contener como datos: nro. de venta, fecha de venta, número de función, sala en la que se proyecta la película, el nombre de la película, fecha y hora de la función, el precio, el tipo de entrada (si es mayor, menor, jubilado) y la calificación de la película, que según especificaciones de la Ley de Cine Nro. 17.741, debe ser informada tanto en la entrada como al inicio de la película. Es importante destacar que la entrada es válida únicamente para la fecha, hora y función indicadas en la misma.

Los tipos de entradas y los días y horarios de proyección son los que determinan el precio de la entrada, que también pueden variar en cada cine del complejo. Las funciones admiten ciertos tipos de entradas y otros no, dependiendo de factores como: horarios, calificación de las películas, etc. Por ejemplo: si una película está calificada como para mayores de 16 años, para esa función no se pueden vender entradas de TIPO = MENOR. Cada función tiene asociado un tipo de función, que determina si la función es un pre-estreno, un estreno o una función normal.

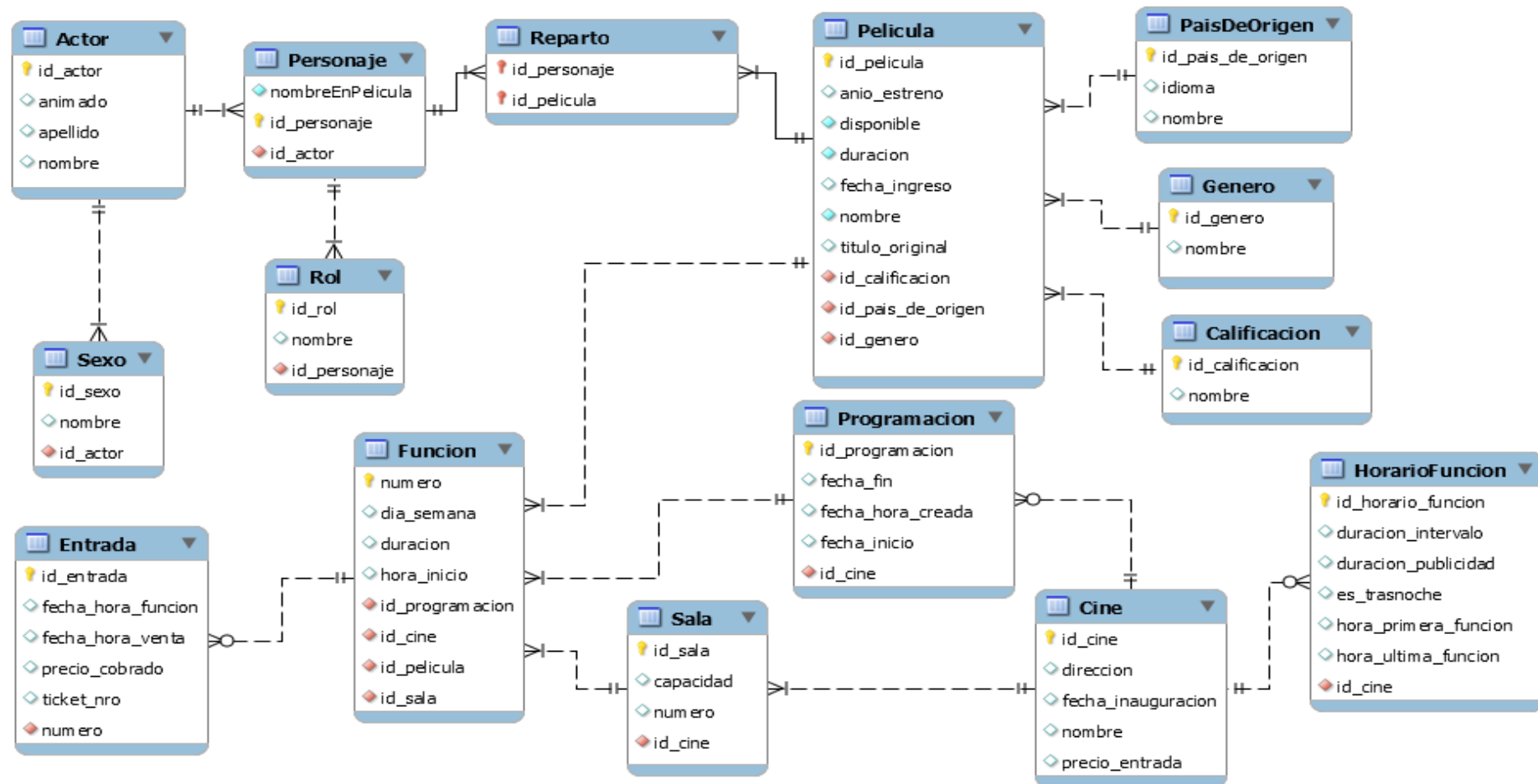
## Modelo de Dominio



Consignas:

- Construir el diagrama de entidad-relación correspondiente al Modelo de Dominio presentado, especificando claves primarias y foráneas.
- Anotar las clases de entidad dentro de la aplicación Spring, utilizando el estándar JPA; indicando los mapeos de relación correspondientes y la especificación de columnas, si fuera requerido.
- Implementar la capa de acceso a datos, utilizando repositorios de Spring. Para más información consultar en el apunte teórico de Base de Datos en la sección **Repositorios con Spring Data JPA**.
- Una propuesta de solución está disponible en el repositorio, en la siguiente dirección: <https://github.com/111milprogramadores>. Para obtener asistencia sobre cómo descargar los archivos de este repositorio, está disponible el “Instructivo de descarga de proyecto del repositorio público de 111mil en GitHub”.

Solución propuesta para el Diagrama Entidad-Relación que mapea las clases del Modelo de dominio del Complejo de Cines:



## Ejercicio 2: Caso Práctico – Pizzería

### Presentación del Caso de Estudio

Una pizzería de la ciudad ofrece a sus clientes una amplia variedad de pizzas de fabricación propia, de varios tamaños (8, 10 y 12 porciones). Los clientes tienen a disposición un menú que describe para cada una de las variedades, el nombre, los ingredientes y el precio según el tamaño y el tipo (a la piedra, a la parrilla, de molde) de la pizza. Los clientes realizan sus pedidos en el mostrador.

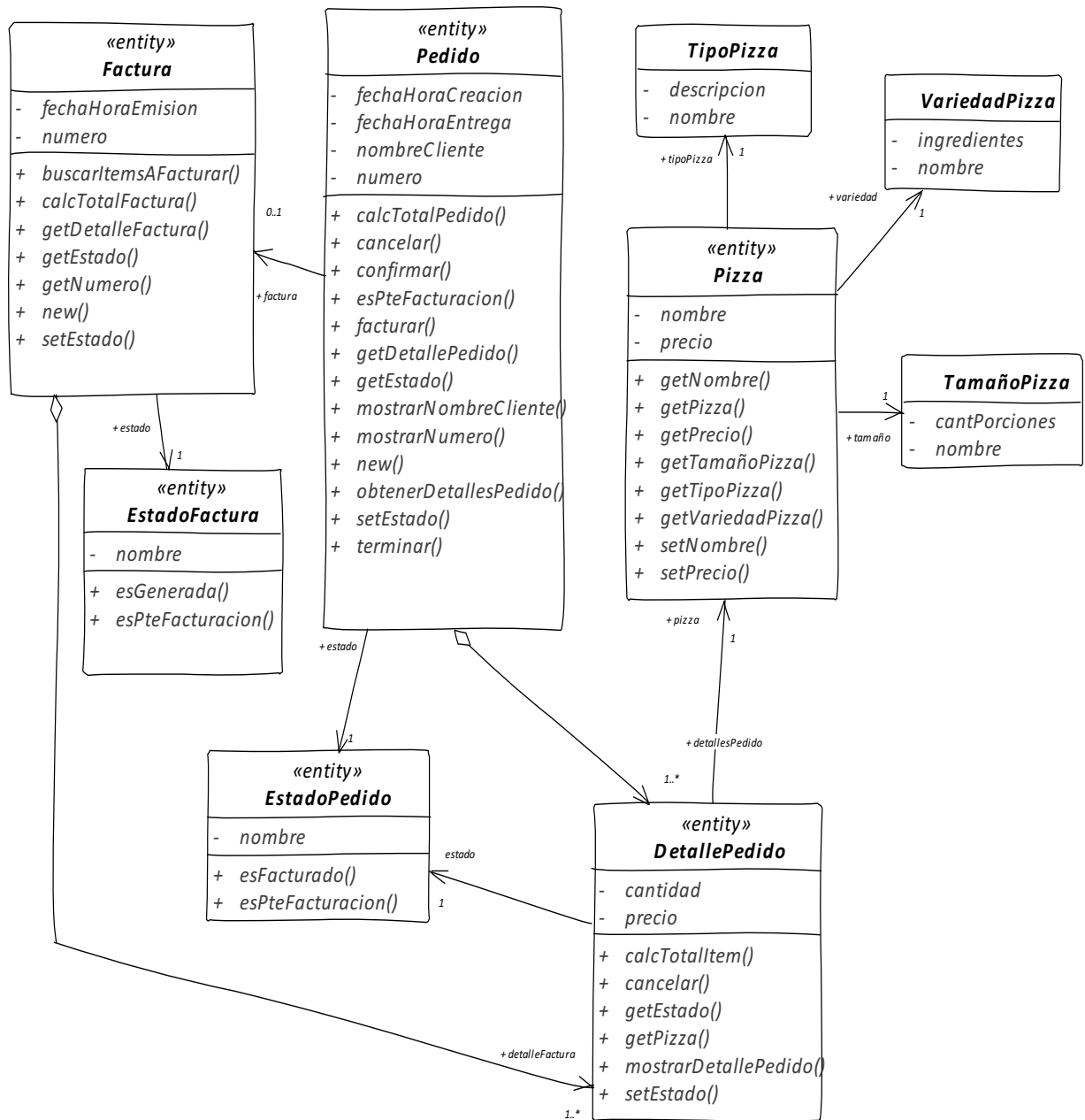
El pedido debe contener el nombre del Cliente, para llamarlo cuando su pedido está listo; la cantidad de pizzas, el tamaño, la variedad, la fecha del pedido, la hora en la que el pedido debe entregarse y la demora estimada informada al cliente.

El pedido va a la cocina y cuando está preparado se informa al que lo tomó para que se genere la factura correspondiente y se le entregue el pedido al cliente.

El dueño de la pizzería ha manifestado la necesidad de acceder al menos a la siguiente información:

- Variedades y tipos de pizzas más pedidas por los clientes.
- Ingresos (recaudaciones) por períodos de tiempo.
- Pedidos (cantidad y monto) por períodos de tiempo.

## Modelo de Dominio





**Generar Reporte de pizzas más demandadas**

Como Administrador de la Pizzería quiero obtener un reporte de pizzas más pedidas para decidir que variedades de pizza comercializar

3

Nota:

- Incluir tabla, que muestre para cada variedad de pizza, de las 10 más vendidas, el nombre y el monto en números, para el período seleccionado.
- El período, es una fecha desde y una fecha hasta, con formato DD/MM/AAAA.
- La forma de visualización del reporte (impreso en papel, en pantalla o en pdf)

**Pruebas de Usuario**

- ☐ Probar ingresar un período válido, (fecha de inicio menor que la fecha de fin y ambas menores a la fecha del día de la generación del reporte) (pasa)
- ☐ Probar ingresar un período inválido (falla)
- ☐ Probar generar un reporte impreso en papel (pasa)
- ☐ Probar generar un reporte en pantalla. (pasa)
- ☐ Probar generar un reporte en pdf. (pasa)
- ☐ Probar no confirmar generación del reporte y cancela. (pasa)

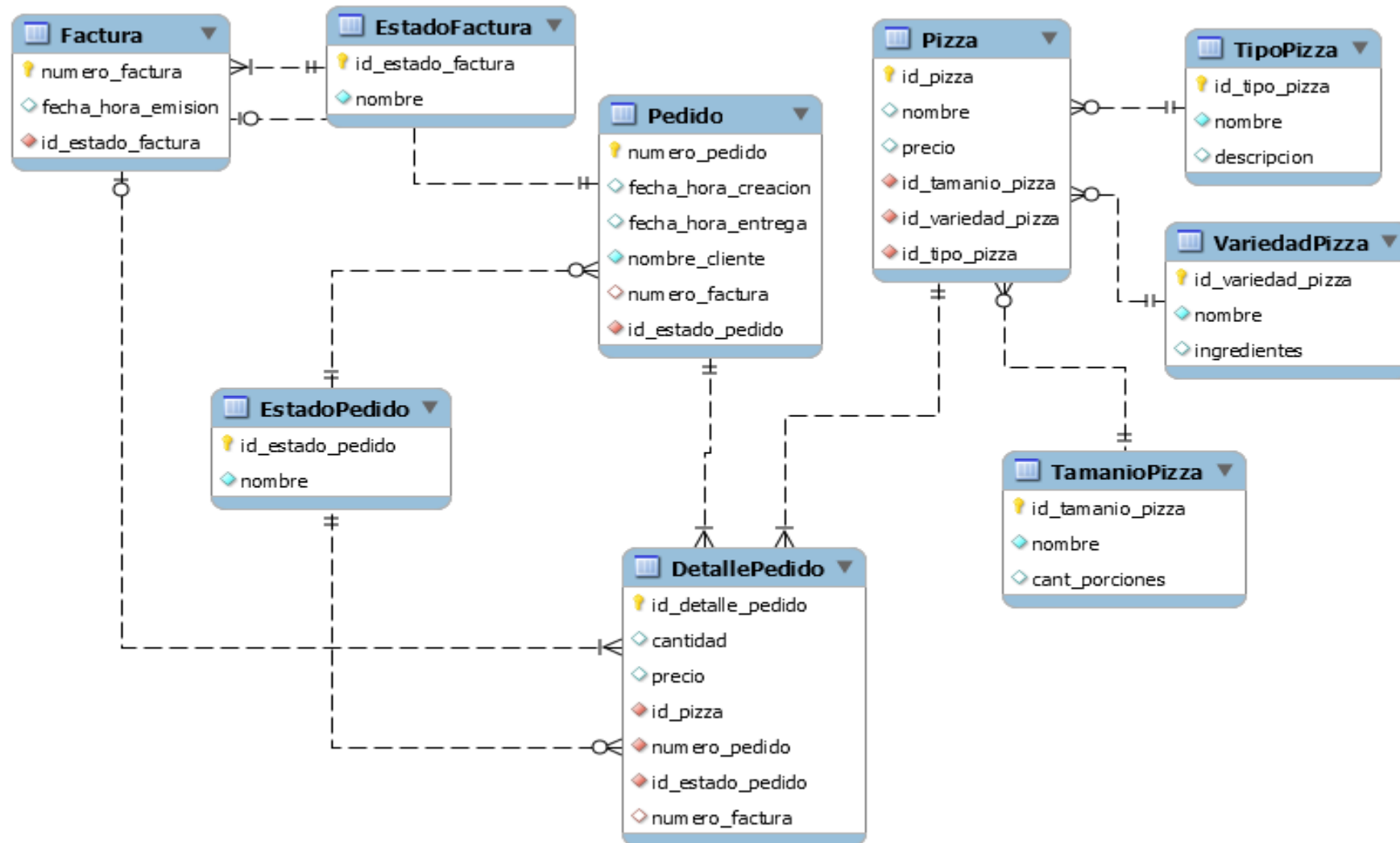
## Consignas:

- Construir el diagrama de entidad-relación correspondiente al Modelo de Dominio presentado, especificando claves primarias y foráneas.
- Anotar las clases de entidad dentro de la aplicación Spring, utilizando el estándar JPA; indicando los mapeos de relación correspondientes y la especificación de columnas, si fuera requerido.
- Implementar la capa de acceso a datos, utilizando repositorios de Spring. Para más información consultar en el apunte teórico de Base de Datos en la sección **Repositorios con Spring Data JPA**.
- Una propuesta de solución está disponible en el repositorio, en la siguiente dirección: <https://github.com/111milprogramadores>. Para obtener asistencia sobre cómo descargar los archivos de este repositorio, está disponible el “Instructivo de descarga de proyecto del repositorio público de 111mil en GitHub”.
- Para la implementación de la Historia de Usuario “Generar reporte de pizzas más demandadas”:
  - Configurar la conexión de la fuente de datos creada con Hibernate con la biblioteca de Jasper Reports.
  - Crear la plantilla de reporte utilizando la herramienta Jasper Studio
  - Implementar el controlador que muestra el reporte generado.

Para más información ver el instructivo: “Generación de reportes con Jasper Reports”, que está disponible en el repositorio, en la siguiente dirección:

<https://github.com/111milprogramadores/bd-instructivos>

Solución propuesta para el Diagrama Entidad-Relación que mapea las clases del Modelo de dominio de la Pizzería:



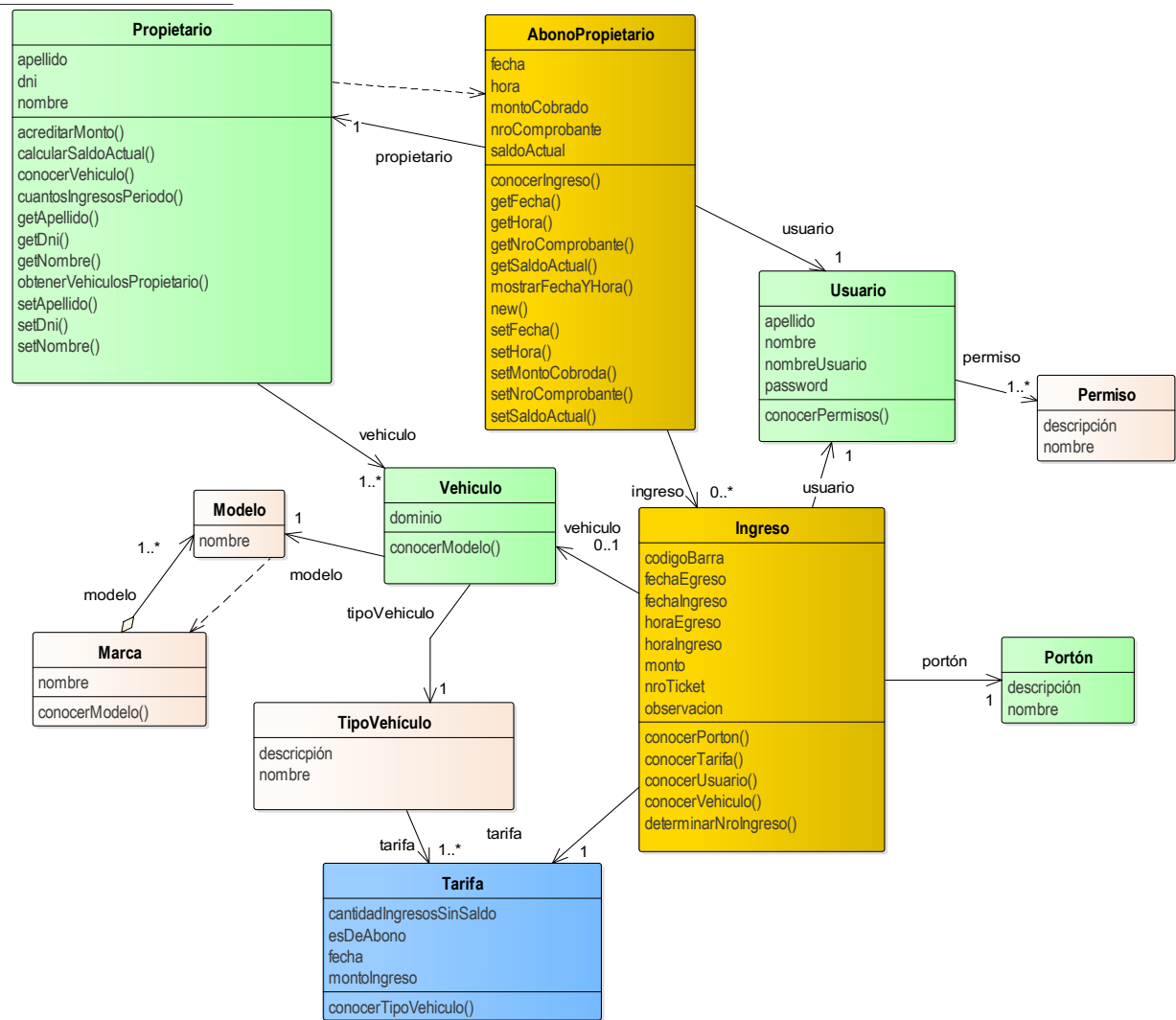
## Ejercicio 3: Caso Práctico – Estacionamiento de Universidad

### Presentación del Caso de Estudio

Se describe a continuación el funcionamiento de la playa de estacionamiento de la Universidad Tecnológica y del sistema de información que le da soporte.

- ⇒ Pueden estacionar distintos tipos de vehículos (motos/automóviles), cada uno de ellos con una tarifa de ingreso diferente. Si tiene abono, el precio es menor.
- ⇒ Se puede ingresar a la playa de estacionamiento por varios portones de ingreso diferentes.
- ⇒ No se asignan lugares específicos para los vehículos; las personas que ingresan al estacionamiento deberán ubicar su vehículo en algún lugar que se encuentre disponible.
- ⇒ Los interesados pueden comprar un abono de estacionamiento, de pago anticipado, que hace que el valor de cada estacionamiento sea más económico que si paga cada vez que ingresa a la playa. Debe informar su DNI y la cantidad de dinero que desea acreditar.
- ⇒ Si es la primera vez que estaciona, debe registrar sus datos personales (apellido, nombre, dni), y los datos del o los vehículos (marca, modelo, dominio), con los cuales desea ingresar a la playa de estacionamiento.
- ⇒ Una vez registrado el propietario, cada vez que necesite acreditar dinero informa su DNI y la cantidad de dinero y se le cobra entregándole un comprobante donde consta: apellido y nombre, dni, fecha de la transacción, monto acreditado y monto disponible en su cuenta.
- ⇒ El comprobante (ticket) que se entrega como constancia del cobro tiene los siguientes datos: apellido y nombre del propietario, dni, fecha y hora de la transacción, monto acreditado y monto disponible en su cuenta, los números de dominio de todos los vehículos registrados de ese propietario y un número único de identificación del comprobante.
- ⇒ Puede tener hasta dos ingresos sin crédito, es decir saldo negativo, que se descontarán de la siguiente vez que acredite dinero en su cuenta.
- ⇒ Mientras tenga crédito, la persona puede ingresar a la playa con cualquiera de los vehículos registrados.
- ⇒ La persona puede en cualquier momento agregar y/o cambiar los vehículos con los que ingresará a la playa de estacionamiento.
- ⇒ El valor del estacionamiento es por el día completo, sin límite de tiempo ni inferior ni superior; es decir se paga un ingreso diario, que es válido independientemente de la cantidad de ingresos que haga durante el mismo día y del tiempo que permanezca en la playa.
- ⇒ Al ingresar se le entrega a la persona un comprobante que contiene: dominio del vehículo, apellido y nombre del dueño del vehículo, el valor del ingreso, la fecha de ingreso y el saldo disponible. También se informa el número de ingreso del día. El portón por el que ingresa y el usuario logueado.
- ⇒ Si el vehículo no está registrado, se guarda en el ingreso el número de dominio del vehículo y se informa como observación que no está registrado.
- ⇒ El primer ingreso del día se cobra, descontando del saldo disponible. A partir del segundo ingreso del día en adelante, el monto debe figurar en cero y se debe informar cuál es el número de ingreso, por ejemplo: "Segundo ingreso del día".
- ⇒ A las personas que desean ingresar a la playa de estacionamiento sin tener el abono de pago anticipado, se les cobra en el momento del ingreso, registrando como observación el número de dominio del vehículo, entregándoles un comprobante con el monto cobrado. Los datos del comprobante en ese caso son: dominio del vehículo, monto, fecha de ingreso, número de vez que ingresa a la playa de estacionamiento, usuario logueado, fecha y hora y portón por el que ingresa.
- ⇒ Si la persona tiene abono, puede tener hasta el valor de dos estacionamientos como saldo negativo, que se descontarán de la siguiente vez que acredite dinero en su cuenta.

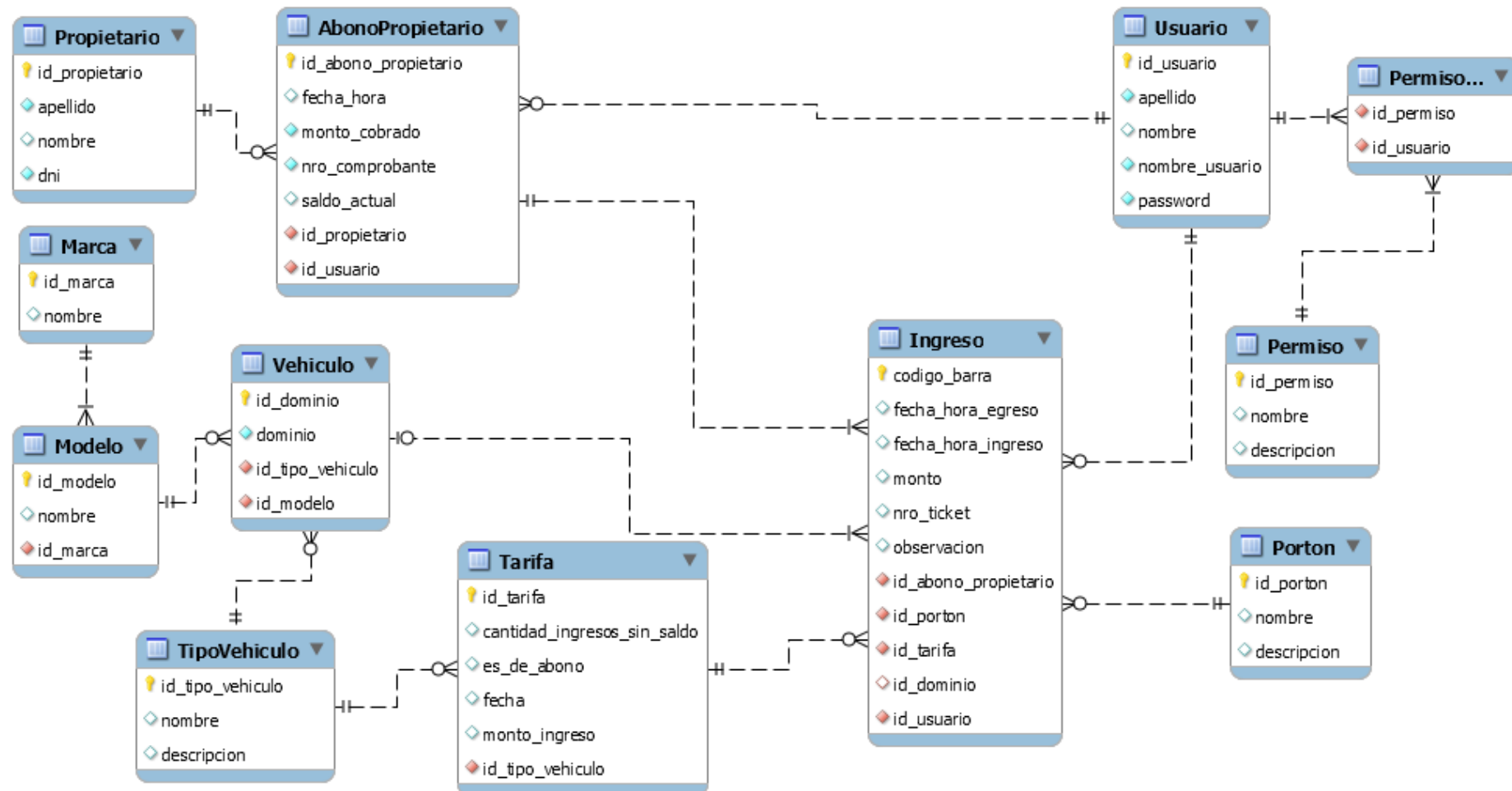
## Modelo de Dominio



Consignas:

- Construir el diagrama de entidad-relación correspondiente al Modelo de Dominio presentado, especificando claves primarias y foráneas.
- Anotar las clases de entidad dentro de la aplicación Spring, utilizando el estándar JPA; indicando los mapeos de relación correspondientes y la especificación de columnas, si fuera requerido.
- Implementar la capa de acceso a datos, utilizando repositorios de Spring. Para más información consultar en el apunte teórico de Base de Datos en la sección **Repositorios con Spring Data JPA**.
- Una propuesta de solución está disponible en el repositorio, en la siguiente dirección: <https://github.com/111milprogramadores>. Para obtener asistencia sobre cómo descargar los archivos de este repositorio, está disponible el “Instructivo de descarga de proyecto del repositorio público de 111mil en GitHub”.

Solución propuesta para el Diagrama Entidad-Relación que mapea las clases del Modelo de dominio del Estacionamiento de Universidad:



## Ejercicio 4: Caso Práctico – Mercado de Abasto

### Presentación del Caso de Estudio

El Mercado de Abasto de Frutas y Verduras de una ciudad de la región necesita un Sistema de Información que brinde soporte a las actividades que allí se realizan.

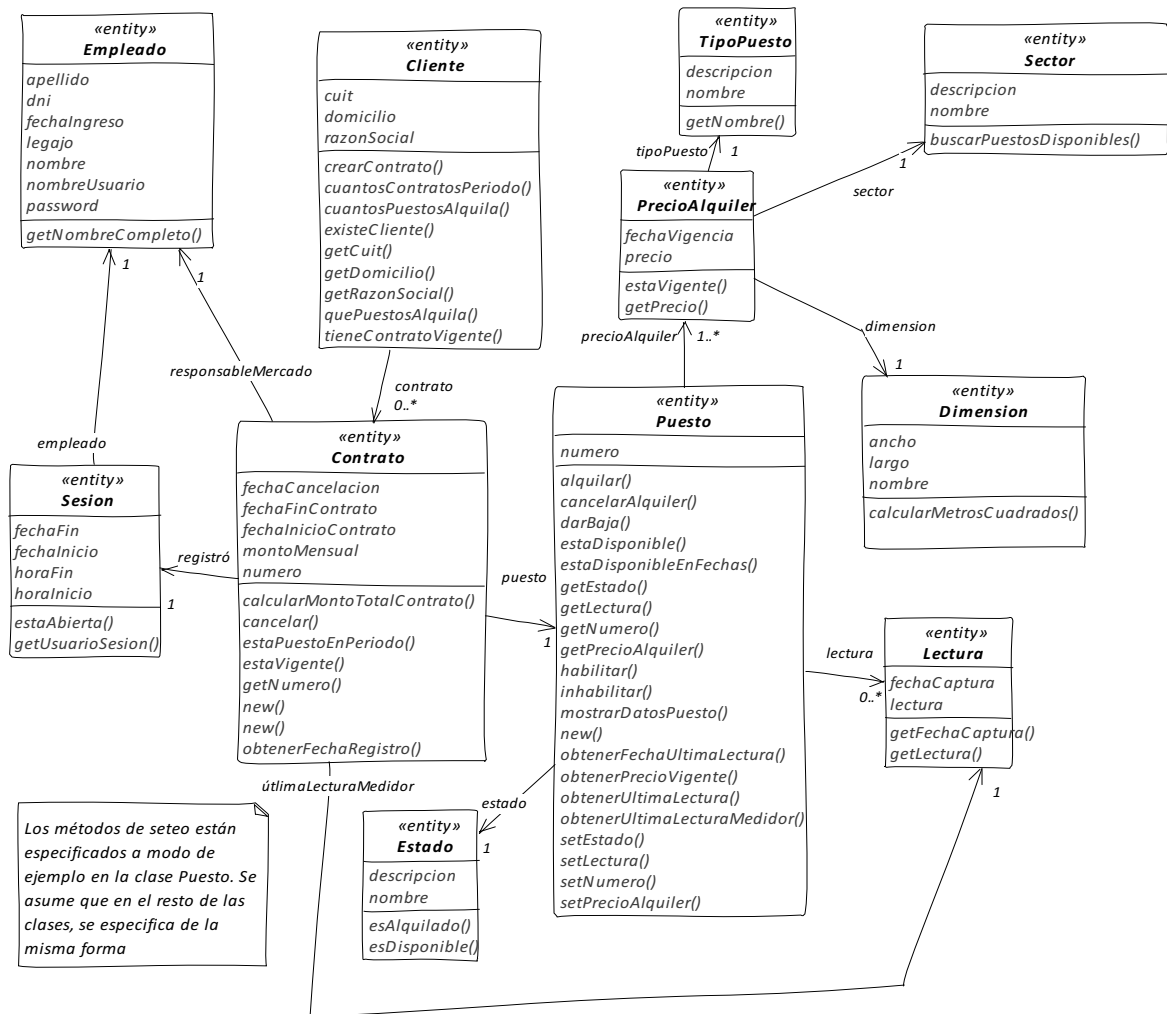
El mercado está organizado en sectores. Cada sector contiene puestos, los cuales son alquilados a empresas y quinteros (genéricamente clientes) para que allí realicen sus ventas. Existen distintos tipos de puestos (con techo, sin techo, con cámara refrigerante, etc.) y distintas dimensiones para cada puesto ( $10\text{m}^2$ ,  $15\text{m}^2$ , etc.), para poder ajustarse mejor a las necesidades de cada cliente.

El precio del alquiler depende del sector en el que se encuentre el puesto, el tipo de puesto y sus dimensiones, y está predefinido.

Cuando un cliente desea alquilar uno o más puestos, se verifica la disponibilidad del tipo de puesto que requiere. Si existe disponibilidad y el cliente está de acuerdo con el precio, se realiza un contrato de alquiler por cada puesto que se alquile. En el contrato se especifica la fecha de inicio y fin del alquiler, el monto mensual del alquiler y tiene además un número que identifica el contrato que es único y el nombre del responsable por parte del Mercado que intervino en la firma del contrato y el responsable de la registración del mismo. Además, cada puesto cuenta con un medidor para el consumo de energía eléctrica. Mensualmente se registran las lecturas de cada medidor, ya que el consumo de cada puesto es facturado al cliente que está alquilando ese puesto. En el momento de efectuar el alquiler, se registra en el contrato la última lectura del medidor del puesto que se está alquilando. Los aspectos vinculados a la facturación quedan excluidos del alcance del sistema, como así también la gestión de cobro de los alquileres.



## Modelo de Dominio



Historia de Usuario: Generar reporte de puestos por estado

|   |          |
|---|----------|
| <p><b>Generar Reporte de puestos por estado</b></p> <p>Como Administrador del Mercado quiero obtener un reporte de puestos por estado para analizar situación del mercado y disponibilidad para alquilar</p> <p>Nota:</p> <ul style="list-style-type: none"> <li>• Permitir seleccionar puestos de una lista</li> <li>• Permitir seleccionar estados de una lista</li> <li>• Mostrar mensaje explicando, si no encuentra datos que cumplan la selección</li> <li>• Incluir tabla, que muestre para cada puesto: tipo de puesto, dimensión, sector, número y estado.</li> <li>• La forma de visualización del reporte (impreso en papel, en pantalla o en pdf)</li> </ul>  | <p>3</p> |
| <p><b>Pruebas de Usuario</b></p> <ul style="list-style-type: none"> <li><input type="checkbox"/> Probar seleccionar un puesto (pasa)</li> <li><input type="checkbox"/> Probar seleccionar todos los puestos (pasa)</li> <li><input type="checkbox"/> Probar seleccionar varios puestos (pasa)</li> <li><input type="checkbox"/> Probar seleccionar puestos y no seleccionar estado (falla)</li> <li><input type="checkbox"/> Probar generar un reporte impreso en papel (pasa)</li> <li><input type="checkbox"/> Probar generar un reporte en pantalla. (pasa)</li> <li><input type="checkbox"/> Probar generar un reporte en pdf. (pasa)</li> <li><input type="checkbox"/> Probar no confirmar generación del reporte y cancela. (pasa)</li> <li><input type="checkbox"/> Probar que no encuentre datos para las selecciones y no muestre mensaje (falla)</li> <li><input type="checkbox"/> Probar que no encuentre datos para las selecciones y muestra mensaje (pasa)</li> </ul> |          |

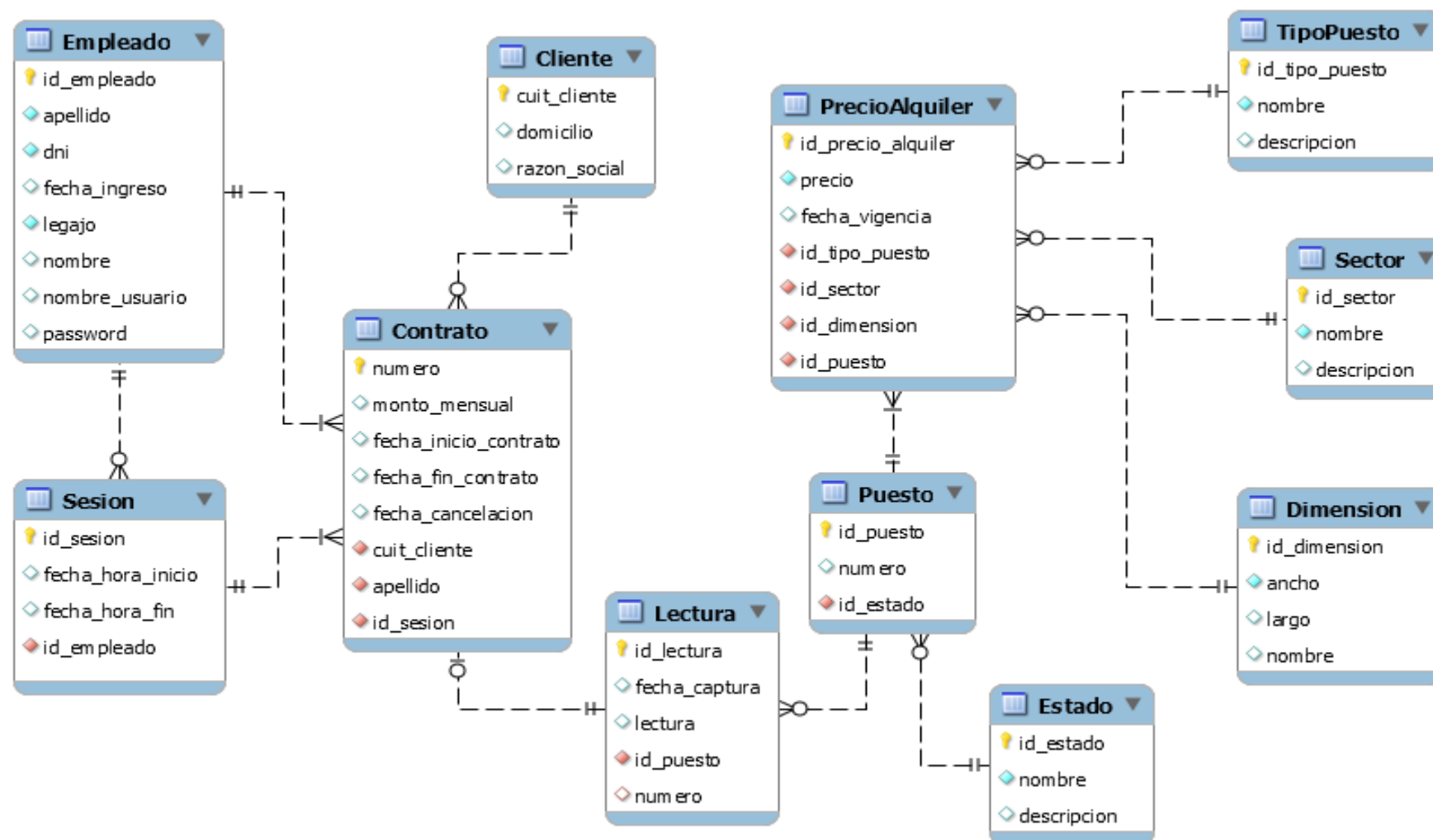
## Consignas:

- Construir el diagrama de entidad-relación correspondiente al Modelo de Dominio presentado, especificando claves primarias y foráneas.
- Anotar las clases de entidad dentro de la aplicación Spring, utilizando el estándar JPA; indicando los mapeos de relación correspondientes y la especificación de columnas, si fuera requerido.
- Implementar la capa de acceso a datos, utilizando repositorios de Spring. Para más información consultar en el apunte teórico de Base de Datos en la sección **Repositorios con Spring Data JPA**.
- Una propuesta de solución está disponible en el repositorio, en la siguiente dirección: <https://github.com/111milprogramadores>. Para obtener asistencia sobre cómo descargar los archivos de este repositorio, está disponible el “Instructivo de descarga de proyecto del repositorio público de 111mil en GitHub”.
- Para la implementación de la Historia “Generar reporte de puestos por estado”:
  - Configurar la conexión de la fuente de datos creada con Hibernate con la librería de Jasper Reports.
  - Crear la plantilla de reporte utilizando la herramienta Jasper Studio
  - Implementar el controlador que muestra el reporte generado.

Para más información ver el instructivo: “Generación de reportes con Jasper Reports”, que está disponible en el repositorio, en la siguiente dirección:

<https://github.com/111milprogramadores/bd-instructivos>

## Diagrama Entidad-Relación



## Ejercicio 5: Caso Práctico – Panadería

### Presentación del Caso de Estudio

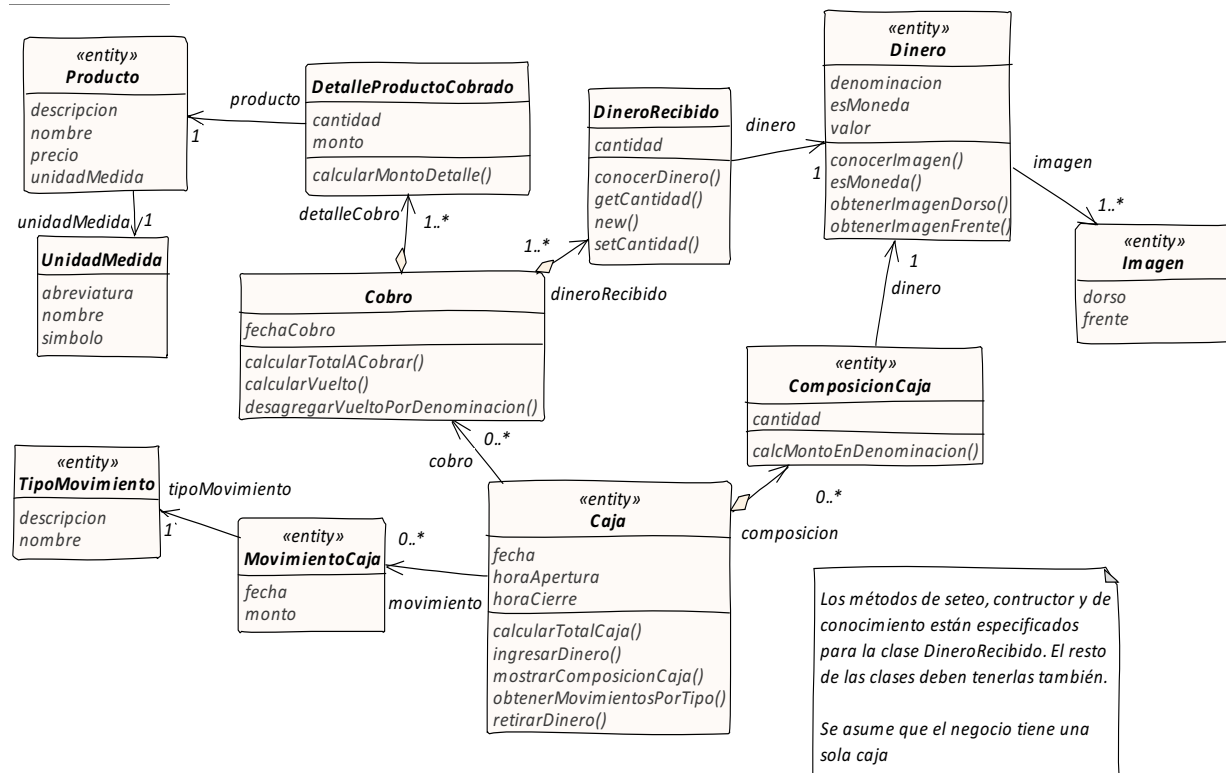
La Panadería que se describe en este caso de estudio, pertenece a la Fundación Brisas de Cambio, ubicada en el interior de la provincia de Córdoba. La Fundación tiene el propósito fundamental de contener laboralmente a un grupo numeroso de jóvenes y adultos con discapacidades intelectuales y físicas. Su objetivo es desarrollar proyectos productivos que les permita desempeñarse en un oficio para sentirse útiles y adquirir a diario el conocimiento necesario para desempeñarse en esta actividad dentro de un ambiente laboral sano.

En este contexto, la panadería está atendida por este grupo de personas con capacidades especiales y la intención es desarrollar un producto de software que asista a las personas en el proceso de venta y cobro de los productos que la panadería vende.

Toda la interacción con el producto debe ser basada en imágenes y muy simple, para lo cual, se presentan a continuación una serie de prototipos que ayudarán a visualizar lo que se pretende construir.

El producto de software esencialmente realizará las siguientes funcionalidades. En este caso haremos foco en el desarrollo del caso de uso que está remarcada, que se presenta a continuación:

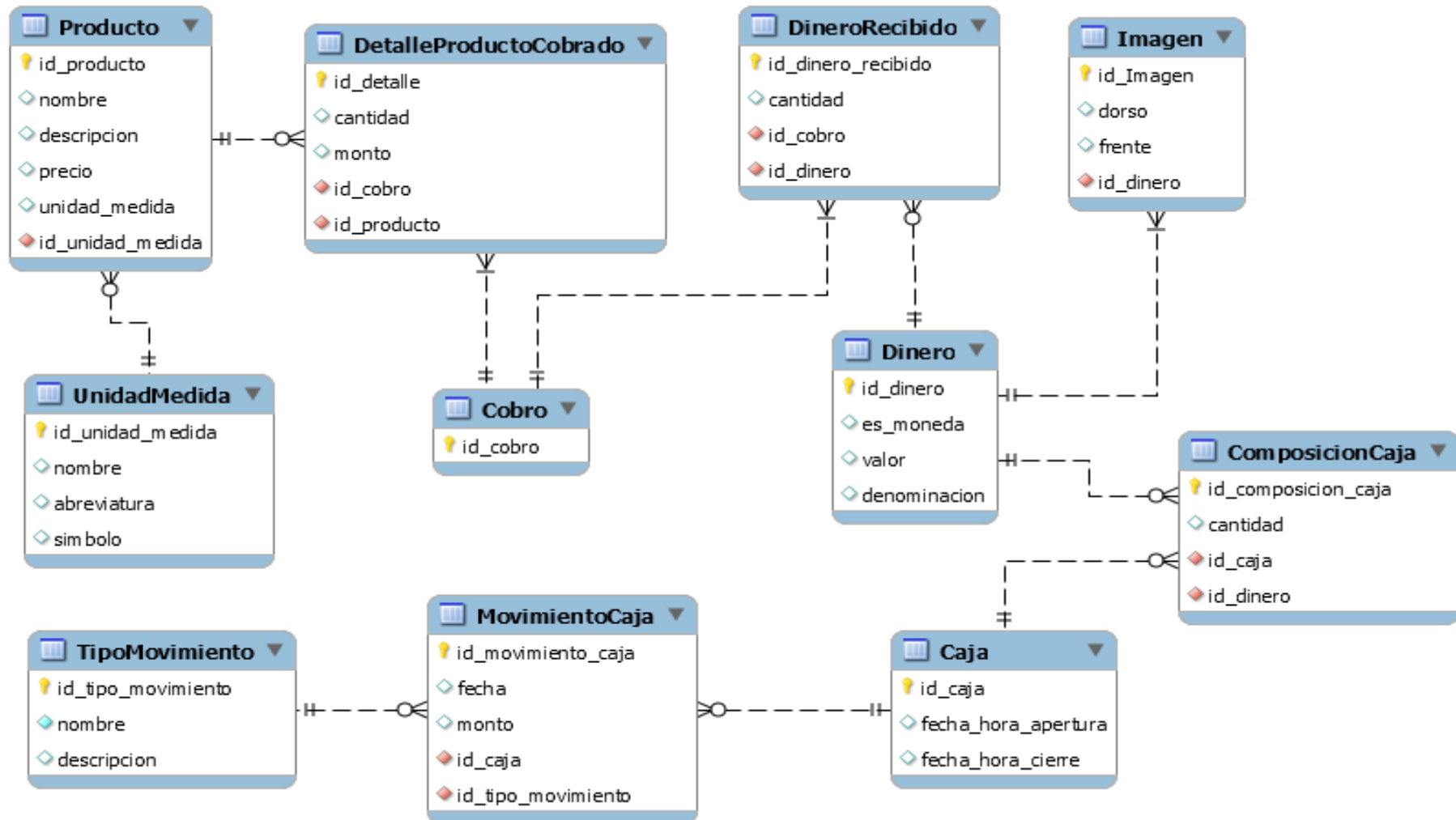
## Modelo de Dominio



## Consignas:

- Construir el diagrama de entidad-relación correspondiente al Modelo de Dominio presentado, especificando claves primarias y foráneas.
- Anotar las clases de entidad dentro de la aplicación Spring, utilizando el estándar JPA; indicando los mapeos de relación correspondientes y la especificación de columnas, si fuera requerido.
- Implementar la capa de acceso a datos, utilizando repositorios de Spring. Para más información consultar en el apunte teórico de Base de Datos en la sección **Repositorios con Spring Data JPA**.
- Una propuesta de solución está disponible en el repositorio, en la siguiente dirección: <https://github.com/111milprogramadores>. Para obtener asistencia sobre cómo descargar los archivos de este repositorio, está disponible el "Instructivo de descarga de proyecto del repositorio público de 111mil en GitHub".

Solución propuesta para el Diagrama Entidad-Relación que mapea las clases del Modelo de dominio de la Panadería:



## Ejercicio 6: Caso Práctico – Consultorio Odontológico

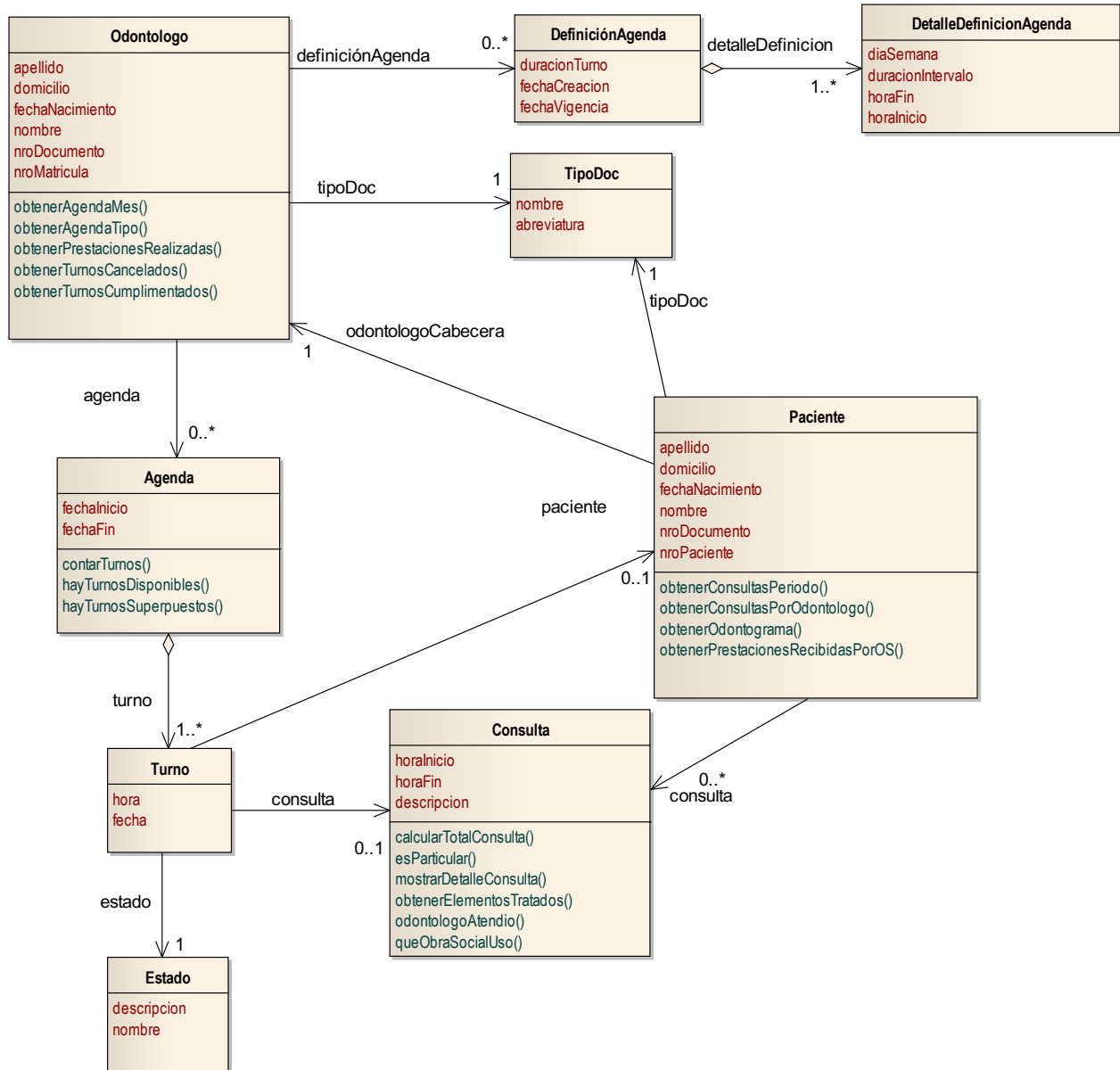
### Presentación del Caso de Estudio

En este consultorio odontológico trabajan varios profesionales que brindan sus servicios. Cuando un paciente necesita atención, debe solicitar un turno previamente. No se atienden pacientes que no tienen turno. El odontólogo para el que el paciente solicita el turno es el que lo va a atender. Cada odontólogo tiene una agenda con los días y horarios en los que puede atender, que se crea mensualmente en función de la disponibilidad que el odontólogo informa, con turnos de 30 minutos de duración. Esta agenda genérica representa los días y horarios de atención que tiene disponible ese odontólogo en términos generales y la duración de su consulta. Esta información se tomará como base para crear la agenda cada mes, considerando para cada mes los días y/u horarios que en ese mes no podrá atender.

Cuando el paciente llama por teléfono, se le pregunta el motivo de la consulta y en función de eso se le asigna uno o más turnos. Por ejemplo, si lo que debe hacerse es un tratamiento de conducto, se le asignan dos turnos de media hora, consecutivos.



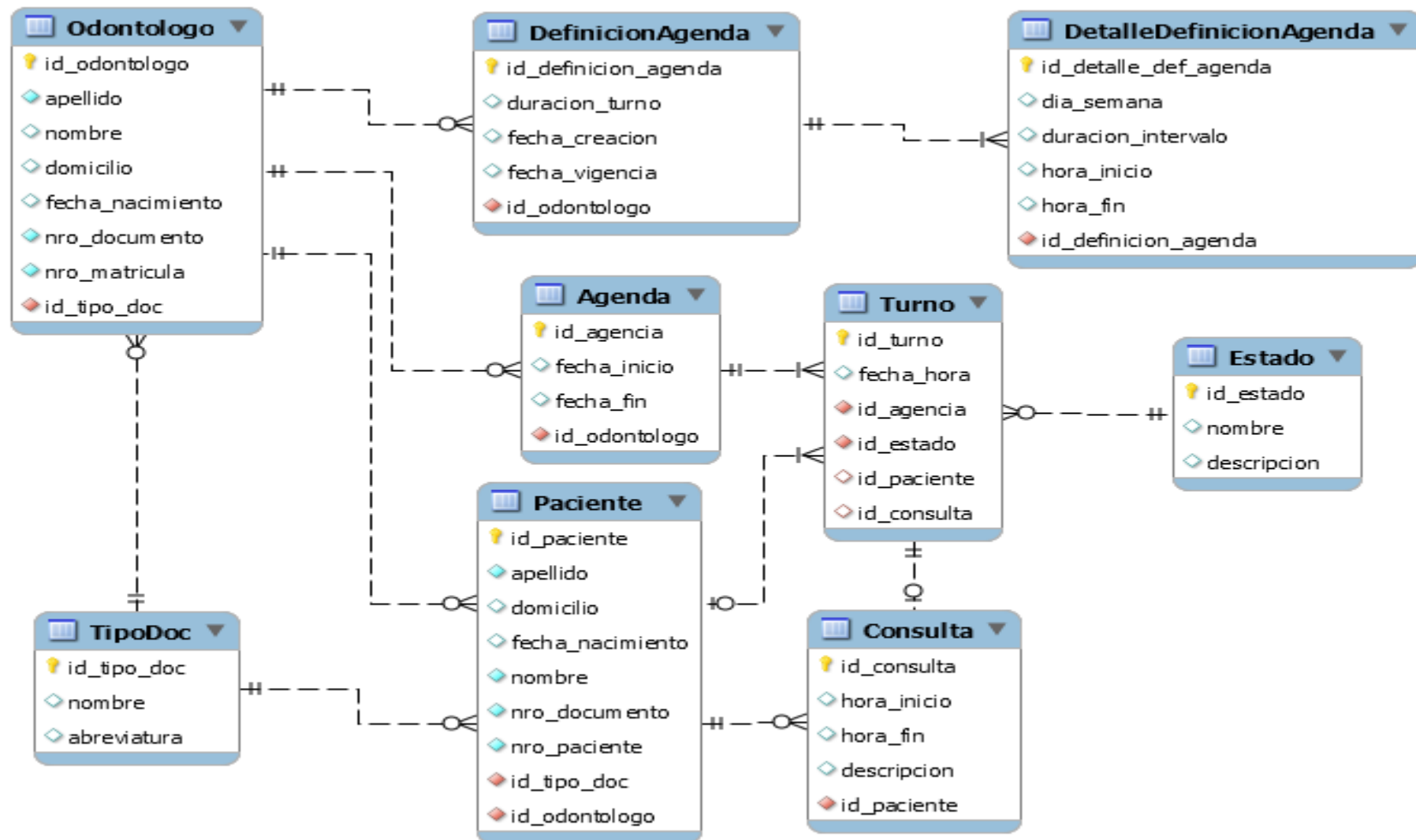
## Modelo de Dominio



Consignas:

- Construir el diagrama de entidad-relación correspondiente al Modelo de Dominio presentado, especificando claves primarias y foráneas.
- Anotar las clases de entidad dentro de la aplicación Spring, utilizando el estándar JPA; indicando los mapeos de relación correspondientes y la especificación de columnas, si fuera requerido.
- Implementar la capa de acceso a datos, utilizando repositorios de Spring. Para más información consultar en el apunte teórico de Base de Datos en la sección **Repositorios con Spring Data JPA**.
- Una propuesta de solución está disponible en el repositorio, en la siguiente dirección: <https://github.com/111milprogramadores>. Para obtener asistencia sobre cómo descargar los archivos de este repositorio, está disponible el “Instructivo de descarga de proyecto del repositorio público de 111mil en GitHub”.

Solución propuesta para el Diagrama Entidad-Relación que mapea las clases del Modelo de dominio del Consultorio Odontológico:



## Fuentes de Información

- Todos los casos de estudio planteados son elaboración del equipo de Formadores que preparó el material. (Meles, Judith /Robles Joaquín / Fey Candelaria).