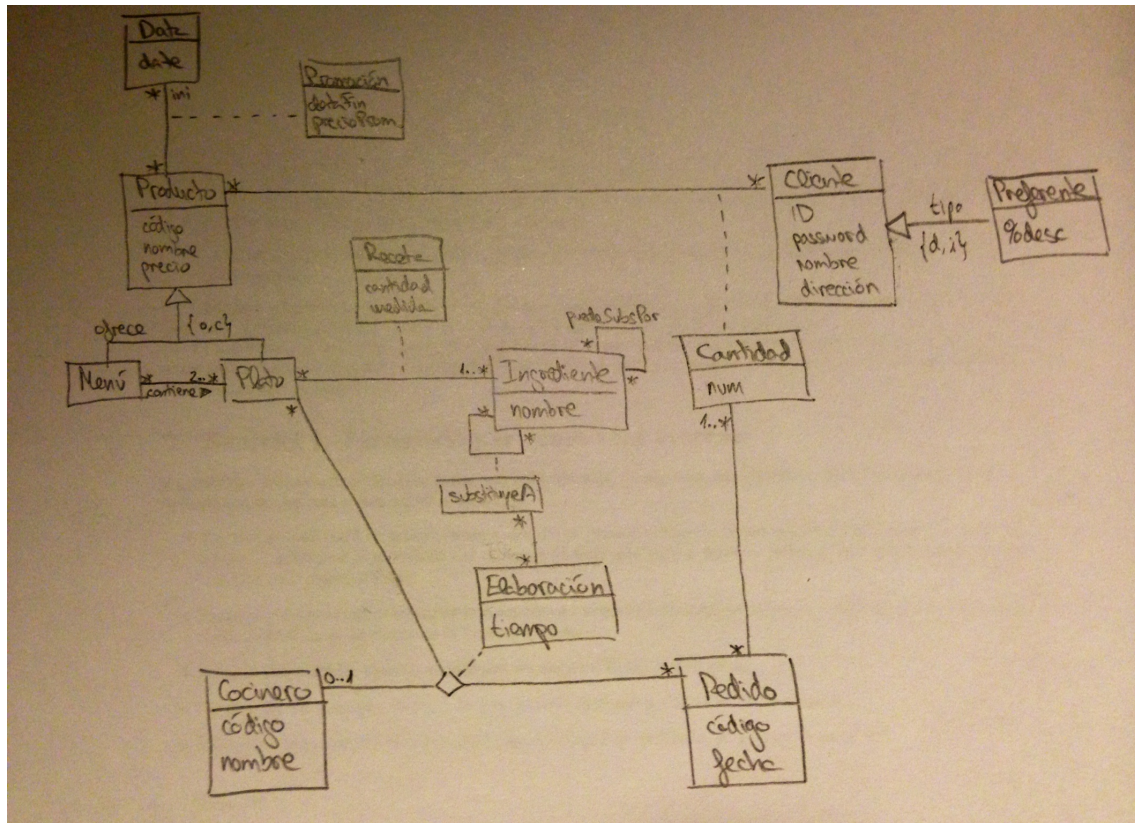


## IES - 5ª ENTREGA



### Restricciones de Integridad:

**Claves Externas:** (Producto, código), (Data, date), (Cliente, usuario + contraseña), (Pedido, código), (Ingrediente, nombre), (Cocinero, código)

### Restricciones Textuales:

- RT1: La fecha de finalización de una promoción ha de ser superior o igual a la fecha de inicio de la misma.
- RT2: El precio de una Promoción ha de ser inferior al del Producto promocionado.
- RT3: Las promociones de un mismo Producto no pueden solaparse temporalmente.
- RT4: Todas las Cantidades de Productos de un Pedido, han de ser escogidas por un mismo Cliente.
- RT5: El Plato Elaborado de la mano de un Cocinero, ha de ser uno de los Platos del Pedido de la Elaboración.
- RT6: El Ingrediente a sustituir ha de ser uno de los que forman la Receta del Plato.
- RT7: El Ingrediente usado como sustituto en una Elaboración, ha de ser uno de los que pueden sustituir al Ingrediente sustituido.

## **ESPECIFICACIÓN:**

Las clases Menú y Plato las he puesto como subclases de Producto porque comparten el identificador (código) y los atributos (nombre y precio).

La relación “contiene” entre Menú y Plato tiene cardinalidad 2..\* en el lado de Plato puesto que un Menú consta, como mínimo, de dos platos.

Como una Promoción está denotada por un Producto y una fecha de inicio, he hecho una asociación entre éstas para identificarla. Y puesto que la Promoción se lleva a cabo en un periodo de tiempo, he puesto la fecha de finalización como atributo de la clase Promoción.

La relación entre un Producto y un Cliente la he llamado Cantidad, ya que un Cliente compra un número concreto de cada producto.

Puesto que un Pedido es un conjunto de Cantidades de Productos escogidos por un Cliente, la clase Pedido la he relacionado con la clase Cantidad.

Debido a que la existencia de un Pedido viene marcada por la necesidad de tener al menos una Cantidad de Productos, en el lado de la relación de Cantidad he puesto cardinalidad mínima de 1.

Debido a que la Receta de un Plato está formada al menos por un Ingrediente, la cardinalidad al lado del Ingrediente, en la relación entre estos dos, he puesto que es como mínimo de uno.

Como un Ingrediente es sustituido por otro Ingrediente, he hecho una relación recursiva en esta clase a la que he llamado puedeSubstituirsePor.

Puesto que en la Elaboración de un Plato, el Cocinero encargado de hacerlo no tiene por qué ser el mismo para todos los Pedidos de ese Plato, he hecho una ternaria entre Plato, Cocinero y Pedido. En la rama del Cocinero he puesto cardinalidad mínima de 0 debido a que en un momento concreto puede no haber ningún cocinero asignado a la elaboración de un plato. Y he puesto cardinalidad máxima de 1 porque la Elaboración de cada Plato de cada Pedido es llevada a cabo por un solo Cocinero.

Como al hacer esto se crea un bucle, he tenido que añadir la RT5.

Para reflejar que, en la Elaboración de un Plato, un Cocinero puede no disponer de uno de los Ingredientes de la Receta, he relacionado Ingrediente consigo mismo y lo he llamado SubstituyeA, y esto lo he relacionado directamente con la Elaboración.

Puesto que al hacerlo se crea un bucle, he tenido que añadir la RT6.